



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
Τμήμα Μηχανικών
Πληροφορικής και τηλεπικοινωνιών

Διπλωματική Εργασία

Σχεδιασμός και υλοποίηση μιας έξυπνης
ενσωματωμένης κεντρικής μονάδας συναγερμού IP

Δημητρίου Βησσαρίων

Επιβλέπων Καθηγητής: Δρ. Μηνάς Δασυγένης
Εργαστήριο Ψηφιακών Συστημάτων και Αρχιτεκτονικής
Υπολογιστών

Κοζάνη, Ιούνιος 2016

Περίληψη

Στην παρούσα διπλωματική εργασία αναπτύχθηκε μία έξυπνη κεντρική μονάδα συναγερμού ελεγχόμενη μέσω διαδικτύου, καθώς επίσης και μία ασύρματη μονάδα με πληκτρολόγιο και RFID δέκτη για τον χειρισμό της.

Ο χρήστης αφού εγκαταστήσει τις συσκευές εισόδου και εξόδου μπορεί να προγραμματίσει την κεντρική μονάδα σύμφωνα με τις ανάγκες του. Μπορεί λόγω χάρη, να ρυθμίσει τις μαγνητικές επαφές στα παράθυρα να ενεργοποιούν άμεσα τον συναγερμό, ενώ η επαφή στην κεντρική είσοδο να έχει μία μικρή καθυστέρηση. Εκτός από τις σειρήνες, συσκευή εξόδου μπορεί να είναι οποιαδήποτε συσκευή μπορεί να ελεγχθεί από την έξοδο 3,3 V της πλακέτας, ή κάποια συσκευή 220 V με χρήση ρελέ, ή η εκτέλεση κάποιου κώδικα shell script.

Παρέχεται επιπλέον η δυνατότητα στον χρήστη να αποθηκεύσει διάφορα σενάρια χρήσης, έτσι ώστε να οπλίζει εκείνο που του παρέχει μεγαλύτερη ασφάλεια. Έτσι, ένα σενάριο μπορεί να χρησιμοποιεί μόνο τις μαγνητικές επαφές και να το οπλίζει όταν βρίσκετε στον χώρο στον οποίο είναι εγκατεστημένος ο συναγερμός, κι ένα άλλο, το οποίο θα χρησιμοποιεί και τους αισθητήρες κίνησης και θα το οπλίζει όταν θα λείπει.

Μέσα από τον ιστοχώρο επίσης μπορεί να ενεργοποιεί τις συσκευές εξόδου, να χρησιμοποιεί τον εβδομαδιαίο χρονοπρογραμματισμό των σεναρίων και των συσκευών εξόδου, να παρακολουθεί ζωντανά τη κατάσταση των συσκευών εισόδου και της κάμερας, και να προγραμματίζει την λειτουργία των μονάδων απομακρυσμένου ελέγχου. Επίσης μπορεί να ορίζει τις ενέργειες που θα εκτελεστούν σε περίπτωση επίθεσης brute-force και να επισκέπτεται την καταγραφή συμβάντων.

Τέλος, η μονάδα ασύρματου ελέγχου οπλίζει και αφοπλίζει τον συναγερμό σύμφωνα με τον προγραμματισμό που έχει γίνει στην κεντρική μονάδα, εισάγοντας τον κωδικό pin ή την κάρτα RFID του χρήστη.

Λέξεις κλειδιά: Ενσωματωμένα Συστήματα, Προγραμματισμός Διαδικτύου, Συναγερμός, Arduino, BeagleBone, PHP, MySQL

Abstract

Thesis topic: Design and implementation of a smart IP embedded alarm system unit.

In this thesis, it has developed a smart alarm system unit which is controlled through the internet. Also, it is developing a wireless unit with a keypad and a RFID reader in order to control the alarm system unit.

The user, after having installed the input and output devices, can program the system unit according to his needs. For instance, he can set the magnetic contacts on the windows to activate immediately the alarm, while the magnetic contact of the main entrance to activate it after a small delay. Besides sirens, an output device can be any device controlled by the 3.3 V output of the board, or any 220 V device with the use of a relay, or the execution of a shell script.

The user also has the capability to save various usage scenarios in order to activate the one which provides the most security. Thereby, one scenario could use only the magnetic contacts and be used when the user is at the place where the alarm unit is installed, and another, which could use also the motion sensors and be used when the user is away.

Furthermore, through the website the user can activate the output devices, schedule the automatic activation of scenarios and output devices, monitor the state of the input devices and the place by using the camera, and to program the function of the remote control units. He can also set the actions to be executed in case of brute-force attack, and visit the event log.

Finally, the wireless alarm unit can activate and deactivate the alarm system according to the settings in the system unit, by entering the pin number or the RFID card of the user.

Keywords: Embedded systems, Web Development, Alarm System, Arduino, BeagleBone, PHP, MySQL

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας, κ. Μηνά Δασυγένη, για την καθοδήγηση και ουσιαστική βοήθεια που μου παρείχε κατά τη διάρκεια εκπόνησης αυτής της διπλωματικής εργασίας. Επίσης, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στην οικογένεια μου, για την ηθική και οικονομική υποστήριξη που μου προσέφεραν όλα αυτά τα χρόνια.

Δήλωση πνευματικών δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο “Σχεδιασμός και υλοποίηση μιας έξυπνης ενσωματωμένης κεντρικής μονάδας συναγερμού IP” καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Μηνά Δασυγένη αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Copyright (C) Βησσαρίων Δημητρίου – Μηνάς Δασυγένης, 2016, Κοζάνη

Περιεχόμενα

Περίληψη	3
Abstract.....	5
Ευχαριστίες.....	6
Δήλωση πνευματικών δικαιωμάτων	7
Περιεχόμενα	9
Κατάλογος εικόνων	13
Κατάλογος πινάκων.....	15
Κατάλογος κωδίκων	17
Κεφάλαιο 1 – Εισαγωγή	19
1.1 Αντικείμενο διπλωματικής εργασίας.....	19
1.2 Σύνοψη διπλωματικής εργασίας.....	22
Κεφάλαιο 2 – Θεωρητικό υπόβαθρο	23
2.1 Προγραμματισμός διαδικτύου	23
2.1.1 HTML	24
2.1.2 CSS	25
2.1.3 Bootstrap.....	26
2.1.4 JavaScript	26
2.1.5 Ajax	27
2.1.6 PHP.....	28
2.1.7 MySQL	28
2.1.8 jQuery Mobile	29
2.2 Αρχιτεκτονική Arduino	30
2.2.1 Είδη Arduino	30
2.2.2 Πλακέτες επέκτασης Arduino (Arduino Shields).....	31
2.2.3 Τεχνικά χαρακτηριστικά Arduino Uno	32
2.2.4 Προγραμματισμός Arduino Uno	35
2.3 Αρχιτεκτονική BeagleBoard.....	35
2.3.1 Είδη BeagleBoard.....	36
2.3.2 Πλακέτες επέκτασης BeagleBone (BeagleBone Capes)	37
2.3.3 Τεχνικά χαρακτηριστικά BeagleBone Black.....	37

2.3.4 Προγραμματισμός BeagleBone Black	39
2.4 Πλατφόρμες ανάπτυξης λογισμικού	40
2.4.1 Notepad++	40
2.4.2 Fritzing	41
2.4.3 Arduino IDE	42
2.4.4 PuTTY	42
2.4.5 Apache HTTP Server	43
2.4.6 phpMyAdmin	43
2.4.7 Fsw webcam	44
2.4.8 GIMP	44
Κεφάλαιο 3 – Υλικό μέρος συστήματος	45
3.1 Υλικό μέρος – BeagleBone	46
3.1.1 Μαγνητικές επαφές συναγερμού	46
3.1.2 Αισθητήρες κίνησης PIR	49
3.1.3 Σειρήνες συναγερμού	51
3.1.4 Strobe συναγερμού	53
3.1.5 Ενεργητικός υπέρυθρος ανιχνευτής (Infrared Beam)	54
3.1.6 Web Camera	55
3.1.7 Βομβητής	55
3.1.8 LED	57
3.2 Υλικό μέρος – Arduino	59
3.2.1 Πληκτρολόγιο	59
3.2.2 Ανάγνωσης RFID	60
3.2.3 Σειριακός πομποδέκτης Wi-Fi	61
3.2.4 Βομβητής	63
3.2.5 LEDs	63
3.3 Υπολογισμός κόστους υλικών	65
Κεφάλαιο 4 – Λογισμικό μέρος συστήματος	67
4.1 Απαιτήσεις συστήματος	67
4.1.1 Λειτουργικές απαιτήσεις της κεντρικής μονάδας	68
4.1.2 Μη λειτουργικές απαιτήσεις της κεντρικής μονάδας	70
4.1.3 Λειτουργικές απαιτήσεις της μονάδας απομακρυσμένου ελέγχου	70
4.1.4 Μη λειτουργικές απαιτήσεις της μονάδας απομακρυσμένου ελέγχου	70

4.2 Μεθοδολογία σχεδιασμού βάσης δεδομένων.....	71
4.2.1 Πίνακας users	71
4.2.2 Πίνακας rfids	72
4.2.3 Πίνακας devices.....	73
4.2.4 Πίνακας deviceOn	74
4.2.5 Πίνακας gpsetup	75
4.2.6 Πίνακας groups.....	75
4.2.7 Πίνακας <group_name>_group.....	76
4.2.8 Πίνακας scripts	76
4.2.9 Πίνακας scenario_list	77
4.2.10 Πίνακας <scenario_name>_scenario.....	78
4.2.11 Πίνακας <scenario_name>_andSensors.....	80
4.2.12 Πίνακας scenarioOnOff.....	81
4.2.13 Πίνακας keypads	82
4.2.14 Πίνακας banned_channels	84
4.2.15 Πίνακας bruteforce_actions.....	85
4.2.16 Πίνακας log	86
4.2.17 Πίνακας actions	87
4.2.18 Πίνακας schedule.....	88
4.3 Λειτουργίες του ιστοχώρου	89
4.3.1 Λειτουργίες απλού χρήστη	90
4.3.1 Λειτουργίες διαχειριστή	95
4.4 Λειτουργίες ιστοχώρου για κινητές συσκευές.....	101
4.5 Λειτουργίες webservice.....	102
4.6 Το λογισμικό της κεντρικής μονάδας.....	103
4.6.1 Ανάλυση λειτουργίας της κεντρικής μονάδας.....	103
4.6.2 Παρουσίαση βιβλιοθηκών και συναρτήσεων	104
4.7 Το λογισμικό της μονάδας απομακρυσμένου ελέγχου.....	106
4.7.1 Ανάλυση λειτουργίας της μονάδας απομακρυσμένου ελέγχου.....	106
4.7.2 Παρουσίαση βιβλιοθηκών και συναρτήσεων	107
4.7.3 Επικοινωνία Arduino - BeagleBone.....	109
4.7.4 Παράλληλη εκτέλεση εργασιών	109
4.8 Ασφάλεια συστήματος	110

4.9 Ανάλυση κώδικα	111
4.9.1 Ενεργοποίηση σεναρίου.....	111
4.9.2 Καταγραφή εικόνας	113
4.9.3 Συνάρτηση loop της μονάδας απομακρυσμένου ελέγχου.....	113
4.9.4 Σύνδεση του Arduino στο Access Point	115
4.10 Σύνοψη μετρικών κώδικα	116
4.10.1 Μετρική κώδικα Arduino.....	116
4.10.2 Μετρική κώδικα back-end λογισμικού	116
4.10.3 Μετρική κώδικα ιστοχώρου.....	116
4.11 Ανάλυση SWOT	117
4.12 Δοκιμές.....	118
Κεφάλαιο 5 – Συμπεράσματα και μελλοντικές επεκτάσεις.....	121
5.1 Σύνοψη και συμπεράσματα.....	121
5.2 Προβλήματα που ανέκυψαν κατά την υλοποίηση	122
5.3 Μελλοντικές επεκτάσεις	123
Βιβλιογραφία.....	125

Κατάλογος εικόνων

Εικόνα 1: Η σύνταξη της CSS.....	25
Εικόνα 2: Παράδειγμα ιστοσελίδας με χρήση Bootstrap	26
Εικόνα 3: Παράδειγμα κώδικα JavaScript.....	27
Εικόνα 4: Τεχνολογία Ajax	27
Εικόνα 5: Παράδειγμα ιστοσελίδας με χρήση jQuery Mobile	29
Εικόνα 6: Arduino Uno	30
Εικόνα 7: Arduino Ethernet.....	30
Εικόνα 8: Arduino Leonardo.....	30
Εικόνα 9: Arduino Yun	31
Εικόνα 10: Arduino Micro	31
Εικόνα 11: Arduino Mega Adk	31
Εικόνα 12: Arduino Due.....	31
Εικόνα 13: Ethernet Shield.....	32
Εικόνα 14: Wi-Fi Shield.....	32
Εικόνα 15: GSM Shield.....	32
Εικόνα 16: Motor Shield	32
Εικόνα 17: Ακροδέκτες τροφοδοσίας Arduino	33
Εικόνα 18: Ψηφιακοί ακροδέκτες, γείωσης και τάσης αναφοράς Arduino	34
Εικόνα 19: Αναλογικοί ακροδέκτες Arduino	34
Εικόνα 20: Παράδειγμα προγράμματος Arduino	35
Εικόνα 21: BeagleBoard.....	36
Εικόνα 22: BeagleBoard-xM.....	36
Εικόνα 23: BeagleBone	36
Εικόνα 24: BeagleBone Black.....	36
Εικόνα 25: LCD7 Cape	37
Εικόνα 26: Tracking Cape	37
Εικόνα 27: Battery Cape.....	37
Εικόνα 28: BeagleBone Headers	38
Εικόνα 29: Ακροδέκτες BeagleBone.....	39
Εικόνα 30: Παράδειγμα προγράμματος BeagleBone σε Python.....	40
Εικόνα 31: Επεξεργαστής κειμένου Notepad++	41
Εικόνα 32: Το περιβάλλον εργασίας του Fritzing.....	41
Εικόνα 33: Το περιβάλλον εργασίας του Arduino IDE.....	42
Εικόνα 34: Το περιβάλλον εργασίας του PuTTY.....	43
Εικόνα 35: Το περιβάλλον εργασίας του phpMyAdmin.....	44
Εικόνα 36: Μαγνητική επαφή συναγερμού.....	46
Εικόνα 37: Συνδεσμολογία pull-up αντίστασης.....	47
Εικόνα 38: Συνδεσμολογία pull-down αντίστασης	48
Εικόνα 39: Κύκλωμα μαγνητικής επαφής.....	49
Εικόνα 40: Αισθητήρες κίνησης PIR	50

Εικόνα 41: Κύκλωμα αισθητήρων κίνησης PIR.....	51
Εικόνα 42: Σειρήνες συναγερμού	51
Εικόνα 43: Κύκλωμα σειρήνων συναγερμού.....	52
Εικόνα 44: Strobe συναγερμού	53
Εικόνα 45: Κύκλωμα Strobe	53
Εικόνα 46: Ενεργητικός υπέρυθρος ανιχνευτής	54
Εικόνα 47: Κύκλωμα υπέρυθρου ανιχνευτή	55
Εικόνα 48: Web Camera	55
Εικόνα 49: Βομβητής.....	56
Εικόνα 50: Κύκλωμα βομβητή BeagleBone	56
Εικόνα 51: LED	57
Εικόνα 52: Κύκλωμα LED BeagleBone	57
Εικόνα 53: Κύκλωμα BeagleBone	58
Εικόνα 54: Αριθμητικό πληκτρολόγιο.....	59
Εικόνα 55: Κύκλωμα αριθμητικού πληκτρολόγιου	60
Εικόνα 56: Αναγνώστης RFID.....	61
Εικόνα 57: Κύκλωμα αναγνώστη RFID	61
Εικόνα 58: ESP8266	62
Εικόνα 59: Κύκλωμα ESP8266	62
Εικόνα 60: Κύκλωμα βομβητή Arduino	63
Εικόνα 61: Κύκλωμα LEDs Arduino.....	64
Εικόνα 62: Κύκλωμα Arduino	64
Εικόνα 63: Κύκλωμα Arduino, προβολή Breadboard	65
Εικόνα 64: Αρχική σελίδα	90
Εικόνα 65: Διαχείριση σεναρίων	91
Εικόνα 66: Χειρισμός συσκευών εξόδου.....	92
Εικόνα 67: Χρονοπρογραμματισμός.....	92
Εικόνα 68: Επεξεργασία χρήστη.....	93
Εικόνα 69: Παρακολούθηση συσκευών	94
Εικόνα 70: Κάμερα	94
Εικόνα 71: Καταγραφή συμβάντων.....	95
Εικόνα 72: Επεξεργασία χρηστών	96
Εικόνα 73: Προσθήκη συσκευών.....	97
Εικόνα 74: Προσθήκη ομάδων	97
Εικόνα 75: Επεξεργασία μονάδων απομακρυσμένου ελέγχου	98
Εικόνα 76: Διαχείριση scripts	99
Εικόνα 77: Ασφάλεια.....	100
Εικόνα 78: Αρχική σελίδα και διαχείριση σεναρίων από κινητή συσκευή	101
Εικόνα 79: Διαχείριση συσκευών εξόδου και καταγραφή συμβάντων από κινητή συσκευή	102
Εικόνα 80: Η κεντρική μονάδα.....	119
Εικόνα 81: Η μονάδα απομακρυσμένου ελέγχου	119

Κατάλογος πινάκων

Πίνακας 1: Ετικέτες HTML	25
Πίνακας 2: Είδη Arduino.....	31
Πίνακας 3: Πλακέτες επέκτασης Arduino.....	32
Πίνακας 4: Είδη BeagleBoard	36
Πίνακας 5: Πλακέτες επέκτασης BeagleBone	37
Πίνακας 6: Κόστος υλικών.....	66
Πίνακας 7: Η δομή του πίνακα users.....	71
Πίνακας 8: Η δομή του πίνακα rfids	72
Πίνακας 9: Η δομή του πίνακα devices.....	73
Πίνακας 10: Η δομή του πίνακα deviceOn	74
Πίνακας 11: Η δομή του πίνακα gpiosetup	75
Πίνακας 12: Η δομή του πίνακα groups.....	75
Πίνακας 13: Η δομή του πίνακα <group_name>_group.....	76
Πίνακας 14: Η δομή του πίνακα scripts	77
Πίνακας 15: Η δομή του πίνακα scenario_list.....	77
Πίνακας 16: Η δομή του πίνακα <scenario_name>_scenario	78
Πίνακας 17: Η δομή του πίνακα <scenario_name>_andSensors	80
Πίνακας 18: Η δομή του πίνακα scenarioOnOff	81
Πίνακας 19: Η δομή του πίνακα keypads.....	82
Πίνακας 20: Η δομή του πίνακα banned_channels	84
Πίνακας 21: Η δομή του πίνακα bruteforce_actions	85
Πίνακας 22: Η δομή του πίνακα log.....	86
Πίνακας 23: Η δομή του πίνακα actions	87
Πίνακας 24: Η δομή του πίνακα schedule.....	88
Πίνακας 25: Μετρική κώδικα Arduino	116
Πίνακας 26: Μετρική κώδικα back-end λογισμικού.....	116
Πίνακας 27: Μοντέλο SWOT.....	117

Κατάλογος κωδίκων

Κώδικας 1: Ενεργοποίηση σεναρίου	112
Κώδικας 2: Καταγραφή εικόνας.....	113
Κώδικας 3: Συνάρτηση loop του Arduino.....	114
Κώδικας 4: Σύνδεση Arduino με Access Point	115

Κεφάλαιο 1 – Εισαγωγή

Σε αυτό το κεφάλαιο παρουσιάζεται το αντικείμενο της διπλωματικής εργασίας και οι λόγοι, τα κίνητρα, και οι προκλήσεις που οδήγησαν στην ανάληψη της παρούσας εργασίας. Επίσης, αναφέρονται κάποιες παρόμοιες υλοποιήσεις από άλλους ερευνητές και τέλος, γίνεται μία συνοπτική παρουσίαση της δομής της εργασίας.

1.1 Αντικείμενο διπλωματικής εργασίας

Τα τελευταία χρόνια, η ραγδαία διάδοση των ευρυζωνικών συνδέσεων έχει δημιουργήσει τη τάση για τη διασύνδεση ποικίλων συσκευών και εξαρτημάτων με το διαδίκτυο. Με αυτόν τον τρόπο ενισχύεται η λειτουργικότητα και ο έλεγχος των συσκευών αυτών, προσδίδοντας επίσης νέα χαρακτηριστικά και τρόπους χρήσης που απουσίαζαν από τον αρχικό σχεδιασμό τους.

Η τάση αυτή είναι γνωστή με τον όρο “Διαδίκτυο Αντικειμένων” (Internet of Things) με πιο συνηθισμένες εφαρμογές της σήμερα να είναι οι έξυπνες τηλεοράσεις (Smart TVs), οι διαδικτυακές κάμερες (IP Cameras) και οι διαδικτυακές μονάδες αποθήκευσης (Network Attached Storage).

Η παρούσα διπλωματική εργασία καλείται να αναπτύξει μία έξυπνη κεντρική μονάδα συναγερμού, η οποία μπορεί να χειριστεί μέσω διαδικτύου, υλοποιώντας μια λύση “Διαδικτύου Αντικειμένων”. Η κεντρική μονάδα αποτελεί την καρδιά του συναγερμού. Σε αυτήν συνδέονται όλες οι συσκευές του συστήματος, όπως οι αισθητήρες, οι σειρήνες και οι κάμερες. Η μονάδα αυτή, λαμβάνει τα σήματα από τους αισθητήρες και ενεργοποιεί τις συσκευές εξόδου, ανάλογα με το προγραμματισμό που έχει γίνει από το χρήστη.

Ταυτόχρονα, κατασκευάστηκε μία μονάδα απομακρυσμένου ελέγχου, η οποία διαθέτει αριθμητικό πληκτρολόγιο και RFID δέκτη, και μπορεί να ενεργοποιεί και να απενεργοποιεί τον συναγερμό.

Η εγκατάσταση μονάδων συναγερμού τόσο σε οικίες, όσο και σε καταστήματα και επιχειρήσεις έχει γίνει απαραίτητη τα τελευταία χρόνια λόγω της αυξανόμενης εγκληματικότητας και των διαρρήξεων. Ο επίδοξος ληστής θα τραπεί σε φυγή όταν ο συναγερμός ενεργοποιήσει τις σειρήνες υπό τον φόβο πως θα γίνει αντιληπτός από τους περίοικους. Παράλληλα, η ύπαρξη κάποιου συστήματος συναγερμού μπορεί να λειτουργήσει αποτρεπτικά για τον υποψήφιο διαρρήκτη, ο οποίος θα προτιμήσει να εισβάλει σε κάποιον απροστάτευτο χώρο.

Ο τομέας των συστημάτων ασφαλείας είναι αρκετά διαδομένος στην επιστημονική κοινότητα τα τελευταία χρόνια. Στη συνέχεια παρουσιάζονται μερικές υλοποιήσεις από άλλους ερευνητές.

Στη διπλωματική εργασία με θέμα “Σχεδιασμός και κατασκευή ενσωματωμένου συστήματος συναγερμού στην αρχιτεκτονική Arduino με υποστήριξη δυναμικού ιστοχώρου διαχείρισης” [1] υλοποιήθηκε ένα σύστημα συναγερμού αποτελούμενο από ένα έως εννιά μικροελεγκτές Arduino, με κάθε μικροελεγκτή να μπορεί να συνδεθεί με έως και οχτώ αισθητήρια. Η διαχείριση του όλου συστήματος γίνεται μέσω διαδικτύου, επισκέπτοντας την ανάλογη ιστοσελίδα.

Στην δημοσίευση “Implementation of ZigBee-GSM based home security monitoring and remote control system” [2] κατασκευάστηκε ένα σύστημα ασφαλείας και ελέγχου στο οποίο συνδέονται οι αισθητήρες, χρησιμοποιώντας τη τεχνολογία ZigBee [3]. Η επικοινωνία με τη κεντρική μονάδα γίνεται μέσω SMS ή μέσω GSM κλήσης. Το σύστημα αυτό προσφέρει μία χαμηλού κόστους και κατανάλωσης, εύκολη στη χρήση, αξιόπιστη, φορητή λύση ασφάλειας και ελέγχου ενός χώρου.

Στη δημοσίευση “The Design and Implementation of a guard against theft and alarm system with embedded telephone access and KEELOQ technique” [4] κατασκευάστηκε ένα σύστημα για την εποπτεία ενός χώρου κατά της κλοπής, το οποίο χρησιμοποιεί τη τηλεφωνική γραμμή και τη τεχνική KEELOQ [5]. Όταν ανιχνεύσει παραβίαση, θα ενεργοποιηθεί ο συναγερμός και θα καταγράψει εικόνα του χώρου, την οποία θα ανεβάσει στον εξυπηρετητή του συστήματος. Το σύστημα αυτό έχει υψηλή ανοχή στις παρεμβολές, υψηλή απόδοση, και χαμηλές απαιτήσεις σε υλικό.

Στη δημοσίευση “Intelligent residential security alarm and remote control system based on single chip computer” [6] υλοποιήθηκε ένα σύστημα οικιακού αντιδιαρρηκτικού

συναγερμού, συναγερμού πυρανίχνευσης και διαρροής αερίου και ένα σύστημα απομακρυσμένου ελέγχου, το οποίο βασίζεται στο μικροελεγκτή 89C51 [7]. Το σύστημα μπορεί να κάνει κλήση σε κάποιον αριθμό και να τροφοδοτηθεί από τη τηλεφωνική γραμμή.

Εκτός από τις ερευνητικές εφαρμογές όπως αυτές που αναφέρθηκαν, στο εμπόριο υπάρχει πληθώρα μονάδων συναγερμού που αναλαμβάνουν την προστασία της περιουσίας του χρήστη από προσπάθειες διάρρηξης. Ωστόσο, οι φτηνές λύσεις δεν αποτελούν αξιόπιστη λύση, καθώς μπορούν εύκολα να αντιμετωπιστούν από τους διαρρήκτες ή και να μην ενεργοποιηθούν όταν θα απαιτηθεί. Επιπλέον, η λανθασμένη ενεργοποίηση του συναγερμού όταν δεν υπάρχει παραβίαση προκαλεί αναστάτωση στους ενοίκους και τους περίοικους.

Στα πιο ακριβά συστήματα, η εγκατάσταση αυτών απαιτεί εξειδικευμένες γνώσεις ηλεκτρονικών, ή ακόμα και την εγκατάσταση τους από εξειδικευμένο συνεργείο τεχνικών. Επίσης, ο αριθμός των συσκευών εξόδου που μπορούν να συνδεθούν περιορίζεται συνήθως στους δύο, ενώ ο προγραμματισμός τους είναι αρκετά πολύπλοκος και καθόλου ευέλικτος.

Αν και η τεχνολογία των ενσωματωμένων συστημάτων έχει κάνει μεγάλα άλματα τα τελευταία χρόνια, οι κεντρικές μονάδες συναγερμού που υποστηρίζουν διασύνδεση με το διαδίκτυο σήμερα έχουν ιδιαίτερα υψηλό κόστος το οποίο ξεπερνά συνήθως τα 300 ευρώ. Επιπλέον, η λειτουργία τους εξαρτάται από τον κατασκευαστή καθώς ο εξυπηρετητής δεν βρίσκεται ενσωματωμένος στην μονάδα συναγερμού, αλλά σε κάποιον εξωτερικό εξυπηρετητή του κατασκευαστή. Τέλος, οι λειτουργίες που προσφέρουν περιορίζονται στην απλή όπλιση και αφόπλιση του συναγερμού, χωρίς να παρέχουν επιπλέον δυνατότητες.

Τα συγκεκριμένα προβλήματα καλείται να επιλύσει η παρούσα διπλωματική εργασία, αναπτύσσοντας μια κεντρική μονάδα συναγερμού χαμηλού κόστους, η οποία ενσωματώνει τον εξυπηρετητή ώστε να δύναται να λειτουργήσει και σε τοπικά δίκτυα τα οποία δεν είναι συνδεδεμένα με το διαδίκτυο.

Ταυτόχρονα, η κεντρική μονάδα που κατασκευάστηκε παρέχει επιπλέον λειτουργίες, τόσο για τους απλούς χρήστες, όπως ο χρονοπρογραμματισμός και η παρακολούθηση των αισθητήρων σε πραγματικό χρόνο, όσο και για τους προγραμματιστές, όπως η εκτέλεση κώδικα shell script στη περίπτωση ενεργοποίησης του συναγερμού.

1.2 Σύνοψη διπλωματικής εργασίας

Το παρών κείμενο είναι δομημένο σε πέντε κεφάλαια στα οποία αναλύεται πλήρως τόσο το θεωρητικό, όσο και το πρακτικό μέρος ανάπτυξης της διπλωματικής εργασίας. Παρουσιάζονται επίσης οι λειτουργίες που προσφέρει το σύστημα, τα προβλήματα που παρουσιάστηκαν, και πιθανές μελλοντικές βελτιώσεις.

Στο πρώτο κεφάλαιο παρουσιάζεται το αντικείμενο της εργασίας και οι λόγοι που οδήγησαν στην ανάληψη αυτής. Αναφέρονται επίσης κάποιες παρόμοιες υλοποιήσεις από άλλους ερευνητές, και τέλος παρουσιάζεται η σύνοψη της δομής της εργασίας.

Στο δεύτερο κεφάλαιο αναλύεται το θεωρητικό υπόβαθρο των τεχνολογιών που χρησιμοποιήθηκαν για την ανάπτυξη του συστήματος, παρουσιάζονται οι πλακέτες Arduino Uno και BeagleBone Black και οι πλατφόρμες ανάπτυξης λογισμικού.

Στο τρίτο κεφάλαιο παρουσιάζεται το υλικό μέρος της εργασίας, τα τεχνικά χαρακτηριστικά κάθε εξαρτήματος, ο τρόπος λειτουργίας τους, καθώς επίσης και αναλυτικά σχεδιαγράμματα των κυκλωμάτων που υλοποιήθηκαν.

Στο τέταρτο κεφάλαιο αναλύεται το λογισμικό μέρος της εργασίας και οι απαιτήσεις του συστήματος. Παρουσιάζεται η μεθοδολογία σχεδιασμού της βάσης δεδομένων, οι λειτουργίες του ιστοχώρου, και ο προγραμματισμός του BeagleBone και του Arduino.

Στο πέμπτο, και τελευταίο, κεφάλαιο παρουσιάζονται τα συμπεράσματα από την ανάπτυξη του συστήματος, τα προβλήματα που προέκυψαν κατά την υλοποίηση, και πιθανές μελλοντικές επεκτάσεις και βελτιώσεις.

Κεφάλαιο 2 – Θεωρητικό υπόβαθρο

Στο κεφάλαιο αυτό παρουσιάζονται οι τεχνολογίες που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας. Αποσαφηνίζονται θεωρητικές έννοιες που αφορούν τον προγραμματισμό διαδικτύου, της αρχιτεκτονικής Arduino και BeagleBone, και αναλύονται οι πλατφόρμες ανάπτυξης λογισμικού που χρησιμοποιήθηκαν.

2.1 Προγραμματισμός διαδικτύου

Προγραμματισμό διαδικτύου (web development) ορίζουμε την εργασία ανάπτυξης μιας ιστοσελίδας για το διαδίκτυο ή για ένα τοπικό δίκτυο [8]. Στις εργασίες αυτές περιλαμβάνεται ο προγραμματισμός υπολογιστή-πελάτη (client-side scripting) και υπολογιστή-εξυπηρετητή (server-side scripting), καθώς επίσης και το σύστημα διαχείρισης της βάσης δεδομένων.

Ως προγραμματισμό υπολογιστή-πελάτη ορίζουμε την κατηγορία των προγραμμάτων στο διαδίκτυο που εκτελούνται στην πλευρά του πελάτη από τον φυλλομετρητή [9]. Με αυτό τον τρόπο το περιεχόμενο της ιστοσελίδας μπορεί να αλλάζει δυναμικά αναλόγως των δεδομένων που εισάγει ο χρήστης.

Ως προγραμματισμό υπολογιστή-εξυπηρετητή ορίζουμε την τεχνική η οποία ενσωματώνει σενάρια στον διακομιστή τα οποία παράγουν απόκριση προσαρμοσμένη στο αίτημα του πελάτη. Η απόκριση εκτελείται στην πλευρά του διακομιστή πριν σταλεί στον πελάτη [10].

Σύστημα διαχείρισης βάσης δεδομένων είναι μία εφαρμογή λογισμικού που αλληλεπιδρά είτε με τον χρήστη, είτε με άλλες εφαρμογές, προκειμένου να συλλάβει και να αναλύσει δεδομένα. Ένα γενικής χρήσης σύστημα διαχείρισης βάσης δεδομένων, είναι ένα σύστημα λογισμικού που έχει σχεδιαστεί για να επιτρέπει τον ορισμό, τη δημιουργία, την επερώτηση, την ενημέρωση και τη διαχείριση των στοιχείων των βάσεων δεδομένων [11].

2.1.1 HTML

Η HTML (HyperText Markup Language, Γλώσσα Σήμανσης Υπερκειμένου) είναι η πιο διαδεδομένη γλώσσα σήμανσης ιστοσελίδων [12].

Η HTML γράφεται υπό μορφή στοιχείων HTML, τα οποία αποτελούνται από ετικέτες (tags), οι οποίες περικλείονται μέσα σε σύμβολα “<” και “>” (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, εικόνες κλπ.

Η HTML αποτελεί υλοποίηση προγραμματισμού υπολογιστή-πελάτη, και συνεπώς αποτελεί ευθύνη του φυλλομετρητή να διαβάσει τα έγγραφα HTML και να τα συνθέτει σε σελίδες. Ο φυλλομετρητής δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας.

Οι σημαντικότερες ετικέτες που χρησιμοποιήθηκαν για την κατασκευή του ιστοχώρου της παρούσας διπλωματικής παρουσιάζονται στον πίνακα 1.

Ετικέτα HTML	Περιγραφή [13]
<!DOCTYPE>	Πληροφορεί τον φυλλομετρητή για την έκδοση HTML που χρησιμοποιείται.
<html>	Ορίζει την αρχή του HTML κώδικα.
<head>	Ορίζει την αρχή του head, το οποίο περιλαμβάνει πληροφορίες για την κωδικοποίηση των χαρακτήρων, αρχεία μορφοποίησης κ.α.
<script>	Ορίζει την αρχή ενός σεναρίου που χρησιμοποιεί κώδικα υπολογιστή-πελάτη.
<title>	Καθορίζει τον τίτλο του ιστοχώρου.
<body>	Ορίζει την αρχή του σώματος της σελίδας.
<div>	Ορίζει ένα τμήμα της σελίδας με κοινή μορφοποίηση.
<form>	Ορίζει την αρχή μιας φόρμας για την εισαγωγή δεδομένων από τον χρήστη
<input>	Ορίζει ένα στοιχείο υποβολής της φόρμας. Το χαρακτηριστικό type δηλώνει τον τύπο του στοιχείου.
<select>	Ορίζει την αρχή ενός μενού επιλογών.
<table>	Ορίζει την αρχή ενός πίνακα.
<button>	Ορίζει ένα κουμπί.
<h1> έως <h6>	Ορίζει μια επικεφαλίδα.

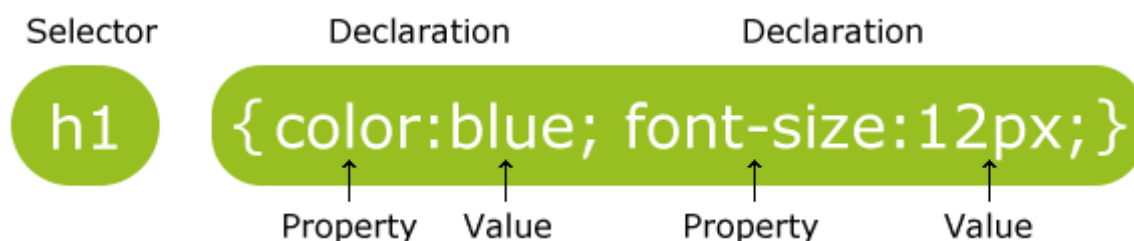
<p>	Ορίζει την αρχή μιας παραγράφου.
 	Ορίζει μία καινούρια γραμμή.

Πίνακας 1: Ετικέτες HTML

2.1.2 CSS

Η CSS (Cascading Style Sheets-Διαδοχικά Φύλλα Στυλ) ή (Αλληλουχία Φύλλων Στυλ) είναι μια γλώσσα υπολογιστή, η οποία ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου, το οποίο έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στυλιστικά μια ιστοσελίδα, δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και να δίνει περισσότερες δυνατότητες σε σχέση με την HTML. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα, η χρήση της CSS κρίνεται ως απαραίτητη [14].

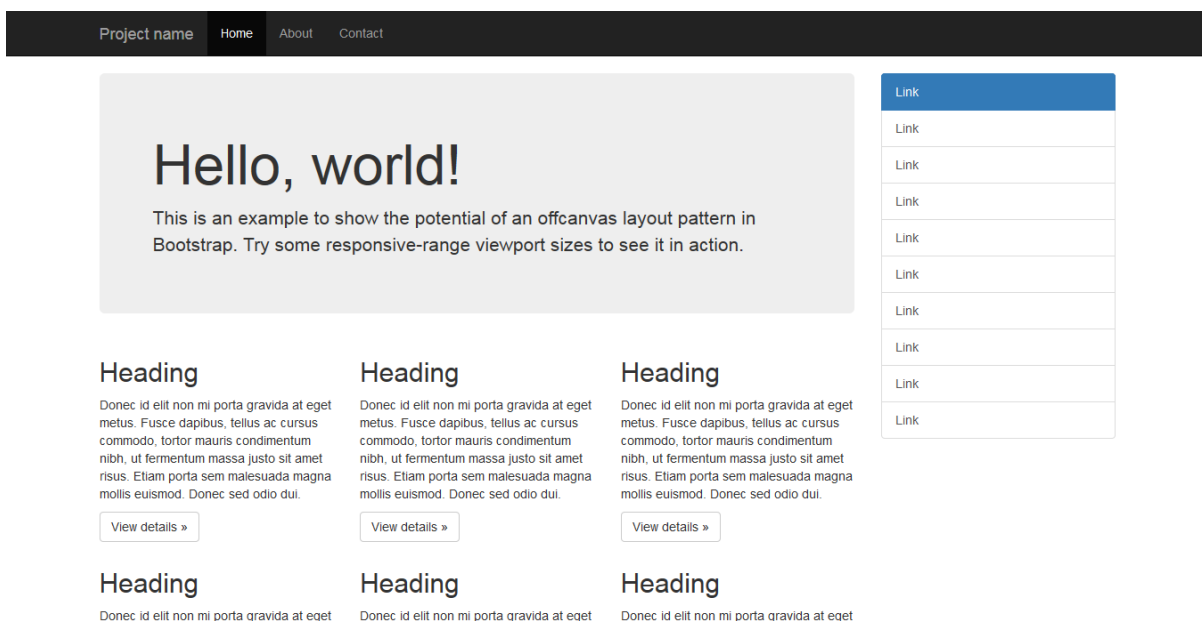
Ένα φύλλο στυλ απαρτίζεται από διαδοχικούς κανόνες. Ένας κανόνας CSS αποτελείται από δύο μέρη: τον επιλογέα (selector), ο οποίος καθορίζει ποια στοιχεία θα μορφοποιηθούν, και από το τμήμα της δήλωσης (declaration). Η δήλωση αποτελείται από τις ιδιότητες (properties), οι οποίες καθορίζουν το πώς θα μορφοποιηθεί το συγκεκριμένο στοιχείο, και από τις τιμές (values) που παίρνει. Στην παρακάτω εικόνα παρουσιάζεται ένα παράδειγμα κανόνα CSS (εικόνα 1).



Εικόνα 1: Η σύνταξη της CSS

2.1.3 Bootstrap

Το bootstrap είναι μία δωρεάν και ανοιχτού κώδικα συλλογή από εργαλεία για την δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών [15]. Περιλαμβάνει έτοιμες σχεδιαστικές προτάσεις για φόρμες, κουμπιά, πλοήγηση και άλλα στοιχεία διεπαφής, καθώς επίσης και προαιρετικά JavaScript σενάρια. Εκτελείται στην πλευρά του πελάτη και συνεπώς κατατάσσεται στον προγραμματισμό υπολογιστή-πελάτη. Ένα παράδειγμα ιστοσελίδας με χρήση Bootstrap παρουσιάζεται στην εικόνα 2.



Εικόνα 2: Παράδειγμα ιστοσελίδας με χρήση Bootstrap

2.1.4 JavaScript

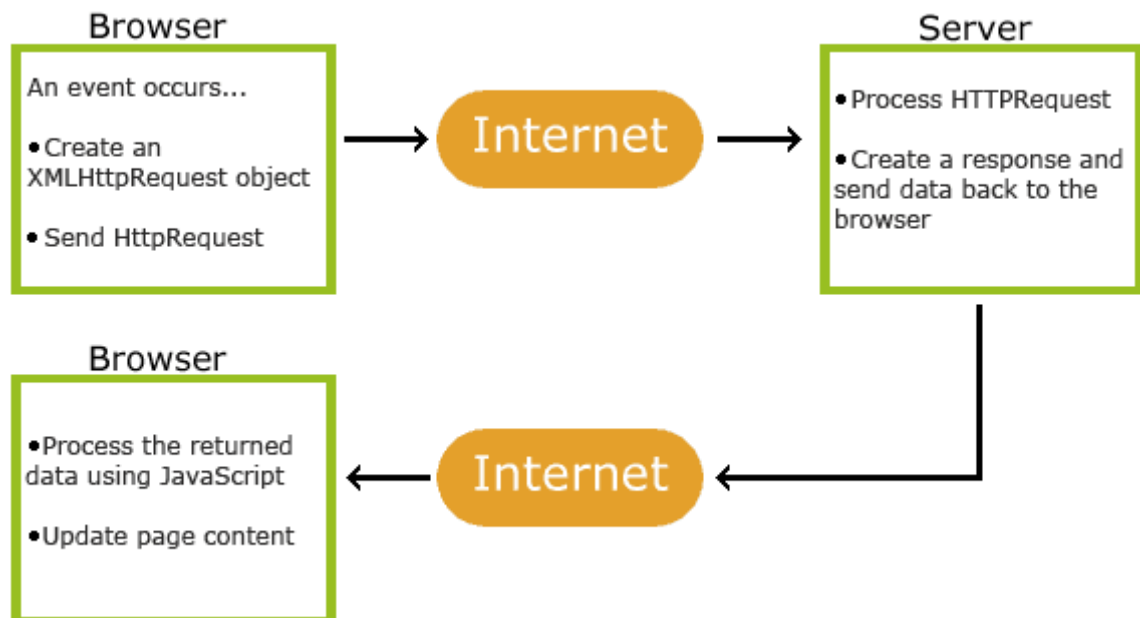
Η JavaScript είναι η πιο δημοφιλής αντικειμενοστραφής γλώσσα σεναρίων. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται [16]. Το γεγονός αυτό επιτρέπει στον προγραμματιστή να εισάγει ελέγχους στα δεδομένα που εισάγει ο χρήστης, οι οποίοι εκτελούνται στην πλευρά του πελάτη και δεν επιβαρύνουν τον εξυπηρετητή. Η εικόνα 3 παρουσιάζει ένα παράδειγμα κώδικα σε JavaScript.

```
function value_changed() {
    var value=document.getElementById('demo').value;
    alert("you typed: "+value);
}
```

Εικόνα 3: Παράδειγμα κώδικα JavaScript

2.1.5 Ajax

Η Ajax (Asynchronous JavaScript and XML, Ασύγχρονη JavaScript και XML) είναι ένα σύνολο τεχνικών ανάπτυξης ιστοχώρων που χρησιμοποιούνται στην μεριά του πελάτη με σκοπό την δημιουργία ασύγχρονων διαδικτυακών εφαρμογών [17]. Με αυτόν τον τρόπο μια ιστοσελίδα μπορεί να στέλνει και να λαμβάνει δεδομένα από ένα διακομιστή ασύγχρονα (στο παρασκήνιο), χωρίς να επηρεαστεί η εμφάνιση και η συμπεριφορά της σελίδας. Τα δεδομένα ανακτώνται χρησιμοποιώντας το αντικείμενο XMLHttpRequest. Στην παρακάτω εικόνα παρουσιάζεται ο τρόπος με τον οποίο λειτουργεί η τεχνολογία Ajax (εικόνα 4).



Εικόνα 4: Τεχνολογία Ajax

2.1.6 PHP

Η PHP είναι μια γλώσσα σεναρίων υπολογιστή-εξυπηρετητή η οποία σχεδιάστηκε για διαδικτυακές εφαρμογές, αλλά χρησιμοποιείται και ως μια γενικής χρήσης γλώσσα προγραμματισμού [18]. Ο κώδικας PHP εκτελείται από ένα διερμηνευτή κώδικα (interpreter) σε πραγματικό χρόνο, ο οποίος τρέχει ως μέρος της εφαρμογής του διακομιστή ώστε να σταλεί στο πρόγραμμα περιήγησης σε μορφή HTML.

Η PHP περιέχει κάποιες καθολικές μεταβλητές (superglobals) οι οποίες βοηθούν στην ανάπτυξη διαδικτυακών εφαρμογών [19]. Οι βασικότερες μεταβλητές είναι οι εξής:

- `$_GET`: Χρησιμοποιείται για τη συλλογή τιμών από μια φόρμα (method="GET") ή από την διεύθυνση URL.
- `$_POST`: Χρησιμοποιείται για τη συλλογή τιμών από μια φόρμα (method="POST").
- `$_REQUEST`: Περιέχει όλες τις τιμές των μεταβλητών `$_POST`, `$_GET` και `$_COOKIE`.
- `$_COOKIE`: Ένα cookie είναι ένα μικρό αρχείο που ενσωματώνει ο διακομιστής στον υπολογιστή του χρήστη. Η συνάρτηση `setcookie()` χρησιμοποιείται για να ορίσει ένα cookie.
- `$_SESSION`: Περιέχει μεταβλητές συνεδρίας διαθέσιμες στο τρέχον σενάριο.

2.1.7 MySQL

Η MySQL είναι ένα ανοιχτού κώδικα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων, η οποία από τον Ιούνιο του 2013 αποτελεί το δεύτερο πιο διαδεδομένο σύστημα διαχείρισης βάσεων δεδομένων [20].

Τα δεδομένα στη MySQL αποθηκεύονται σε αντικείμενα βάσης δεδομένων τα οποία ονομάζονται πίνακες. Ένας πίνακας αποτελεί μια συλλογή από σχετικές καταχωρήσεις δεδομένων και αποτελείται από γραμμές και στήλες. Η διαχείριση αυτών των δεδομένων πραγματοποιείται μέσω των SQL εντολών. Οι βασικότερες εντολές SQL είναι οι εξής [21]:

- Select: Εξάγει δεδομένα από την βάση δεδομένων.
- Update: Ενημερώνει δεδομένα στην βάση δεδομένων.
- Delete: Διαγράφει δεδομένα από την βάση δεδομένων.
- Insert into: Εισάγει δεδομένα στην βάση δεδομένων.
- Create database: Δημιουργεί μία βάση δεδομένων.
- Alter database: Τροποποιεί μία βάση δεδομένων.
- Create table: Δημιουργεί έναν πίνακα.
- Alter table: Τροποποιεί έναν πίνακα.
- Drop table: Διαγράφει έναν πίνακα.
- Create index: Δημιουργεί έναν δείκτη (κλειδί αναζήτησης).
- Drop index: Διαγράφει έναν δείκτη.

2.1.8 jQuery Mobile

Το jQuery Mobile είναι ένα πλαίσιο για την κατασκευή διαδικτυακών εφαρμογών σε κινητές συσκευές [22]. Δημιουργεί ιστοχώρους με μία ελκυστική και “εύκολη στη χρήση” εμφάνιση (εικόνα 5), η οποία λειτουργεί με τον ίδιο τρόπο σε όλες τις συσκευές, καθώς χρησιμοποιεί μόνο HTML, CSS και JavaScript.



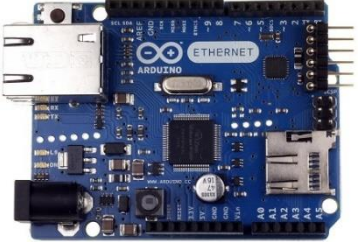
Εικόνα 5: Παράδειγμα ιστοσελίδας με χρήση jQuery Mobile

2.2 Αρχιτεκτονική Arduino

Το Arduino είναι μία ανοιχτού λογισμικού και υλικού εταιρία και μία κοινότητα χρηστών, η οποία σχεδιάζει και κατασκευάζει μητρικές πλακέτες με ενσωματωμένο μικροελεγκτή με σκοπό την δημιουργία ψηφιακών συσκευών, οι οποίες μπορούν να αισθανθούν και να ελέγξουν αντικείμενα του φυσικού κόσμου [23]. Ο προγραμματισμός του Arduino γίνεται με την γλώσσα Wiring η οποία είναι μία παραλλαγή της C/C++.

2.2.1 Είδη Arduino

Συνολικά έχουν κατασκευαστεί 16 πλατφόρμες Arduino. Στον πίνακα 2 παρουσιάζονται τα πιο διαδεδομένα είδη Arduino με τα βασικά χαρακτηριστικά τους. Στην παρούσα διπλωματική επιλέχτηκε το Arduino Uno.

Είδος Arduino	Τεχνικά Χαρακτηριστικά
	Μικροελεγκτής: ATmega328 Τάση λειτουργίας: 5V Ψηφιακές εισοδοι / έξοδοι: 14 (6 με PWM) Αναλογικές εισοδοι: 6 Μνήμη Flash/SRAM/EEPROM: 32KB/2KB/1KB Συχνότητα ρολογιού: 16 MHz
 Εικόνα 7: Arduino Ethernet [25]	Μικροελεγκτής: ATmega328 Τάση λειτουργίας: 5V Ψηφιακές εισοδοι / έξοδοι: 14 (4 με PWM) Αναλογικές εισοδοι: 6 Μνήμη Flash/SRAM/EEPROM: 32KB/2KB/1KB Συχνότητα ρολογιού: 16 MHz W5100 TCP/IP Embedded Ethernet Controller
 Εικόνα 8: Arduino Leonardo [26]	Μικροελεγκτής: ATmega32u4 Τάση λειτουργίας: 5V Ψηφιακές εισοδοι / έξοδοι: 20 (7 με PWM) Αναλογικές εισοδοι: 12 Μνήμη Flash/SRAM/EEPROM: 32KB/2.5KB/1KB Συχνότητα ρολογιού: 16 MHz



Εικόνα 9: Arduino Yun [27]



Εικόνα 10: Arduino Micro [28]



Εικόνα 11: Arduino Mega Adk [29]



Εικόνα 12: Arduino Due [30]

Μικροελεγκτής: ATmega32u4

Μικροεπεξεργαστής: Atheros AR9331

Τάση λειτουργίας: 5V

Ψηφιακές εισοδοι / έξοδοι: 20 (7 με PWM)

Αναλογικές εισοδοι: 12

Μνήμη Flash/SRAM/EEPROM: 32KB/2.5KB/1KB

Μνήμη RAM/Flash μικροεπεξεργαστή: 64 MB
DDR2/16MB

Συχνότητα ρολογιού: 16 MHz

Συχνότητα ρολογιού μικροεπεξεργαστή: 400MHz

Ethernet: IEEE 802.3 10/100Mbit/s

Wi-Fi: IEEE 802.11b/g/n

Card Reader: Micro-SD only

Μικροελεγκτής: ATmega32u4

Τάση λειτουργίας: 5V

Ψηφιακές εισοδοι / έξοδοι: 20 (7 με PWM)

Αναλογικές εισοδοι: 12

Μνήμη Flash/SRAM/EEPROM: 32KB/2.5KB/1KB

Συχνότητα ρολογιού: 16 MHz

Μικροελεγκτής: ATmega2560

Τάση λειτουργίας: 5V

Ψηφιακές εισοδοι / έξοδοι: 54 (15 με PWM)

Αναλογικές εισοδοι: 16

Μνήμη Flash/SRAM/EEPROM: 256KB/8KB/4KB

Συχνότητα ρολογιού: 16 MHz

Μικροελεγκτής: AT91SAM3X8E

Τάση λειτουργίας: 3.3V

Ψηφιακές εισοδοι / έξοδοι: 54 (12 με PWM)

Αναλογικές εισοδοι: 12

Αναλογικές έξοδοι: 2

Μνήμη Flash/SRAM: 512KB/96KB

Συχνότητα ρολογιού: 84 MHz

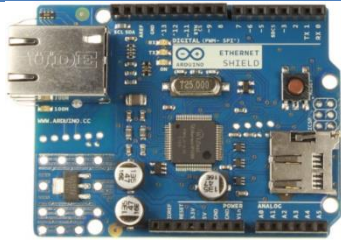
Πίνακας 2: Είδη Arduino

2.2.2 Πλακέτες επέκτασης Arduino (Arduino Shields)

Οι πλακέτες επέκτασης είναι πλακέτες που τοποθετούνται πάνω από το Arduino και του προσθέτουν νέες δυνατότητες. Στον πίνακα 3 παρουσιάζονται μερικά από τα πιο διαδεδομένα Shields.

Arduino Shield

Περιγραφή



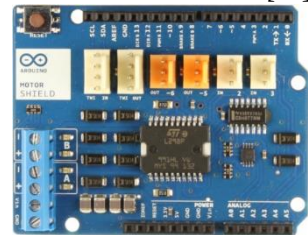
Εικόνα 13: Ethernet Shield [31]



Εικόνα 14: Wi-Fi Shield [32]



Εικόνα 15: GSM Shield [33]



Εικόνα 16: Motor Shield [34]

Προσφέρει ενσύρματη επικοινωνία μέσω Ethernet. Επίσης περιλαμβάνει και μία θύρα υποδοχής SD.

Προσφέρει ασύρματη επικοινωνία μέσω Wi-Fi. Επίσης περιλαμβάνει και μία θύρα υποδοχής SD.

Προσφέρει πρόσβαση στο διαδίκτυο μέσω δικτύου GPRS.

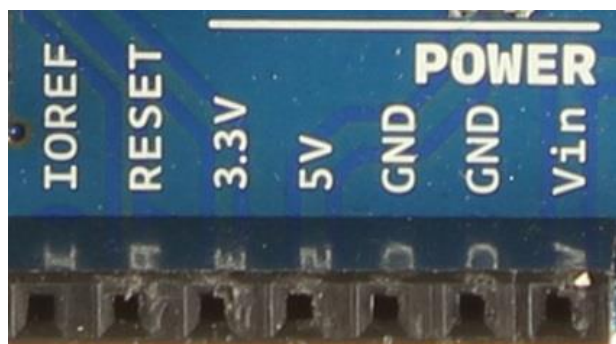
Επιτρέπει την οδήγηση δύο DC κινητήρων ελέγχοντας την ταχύτητα και την κατεύθυνση του καθενός ξεχωριστά.

Πίνακας 3: Πλακέτες επέκτασης Arduino

2.2.3 Τεχνικά χαρακτηριστικά Arduino Uno

Όπως προαναφέρθηκε, η πλακέτα που χρησιμοποιήθηκε στην παρούσα διπλωματική είναι η Arduino Uno, λόγω του χαμηλού κόστους απόκτησης και του αριθμού των ακροδεκτών που διαθέτει και επαρκούν για τις ανάγκες της εργασίας.

Η τροφοδοσία του Arduino μπορεί να πραγματοποιηθεί είτε μέσω σύνδεσης USB, είτε μέσω εξωτερικής τροφοδοσίας 7-12 V που καταλήγει σε βύσμα διαμέτρου 2,1 mm με θετικό κέντρο, ή στους ακροδέκτες Vin και GND. Στην παρακάτω εικόνα εμφανίζονται οι είσοδοι και οι έξοδοι τροφοδοσίας τους Arduino Uno (εικόνα 17).



Εικόνα 17: Ακροδέκτες τροφοδοσίας Arduino

Οι ακροδέκτες τροφοδοσίας (εικόνα 17) είναι οι εξής:

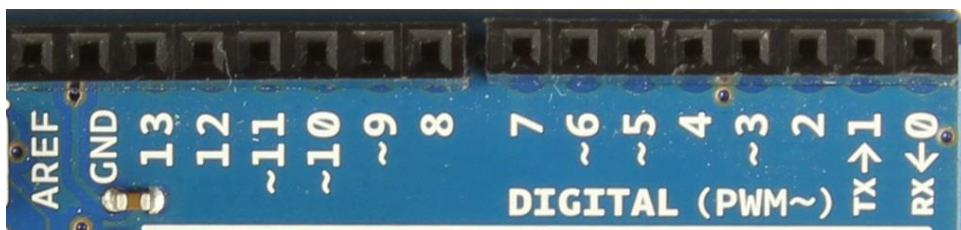
- Vin: Η τάση εισόδου της πλακέτας όταν χρησιμοποιεί εξωτερική τροφοδοσία.
- 5V: Σταθεροποιημένη έξοδος 5 V.
- 3V3: Σταθεροποιημένη έξοδος 3,3 V. Η μέγιστη κατανάλωση ρεύματος δεν πρέπει να υπερβαίνει τα 50 mA.
- GND: Ακροδέκτης γείωσης.
- IOREF: Παρέχει την τάση αναφοράς στην οποία λειτουργεί ο μικροελεγκτής, στις πλακέτες επέκτασης.

Ο μικροελεγκτής έχει τρεις τύπους μνήμης. Διαθέτει μνήμη Flash, SRAM (Static Random Access Memory, Στατική Μνήμη Τυχαίας Προσπέλασης) και EEPROM (Electrically Erasable Programmable Read-Only Memory, Ηλεκτρικός Διεγράψιμη Επανεγγράψιμη Μνήμη Μόνο για Ανάγνωση). Αναλυτικότερα:

- 32 KB μνήμη Flash: 0,5 KB χρησιμοποιείται από τον Bootloader. Στην υπόλοιπη μνήμη αποθηκεύονται τα προγράμματα του χρήστη, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash δεν χάνει τα δεδομένα της σε απώλεια τροφοδοσίας ή επανεκκίνησης.
- 2 KB μνήμη SRAM: Η μνήμη στην οποία αποθηκεύονται οι μεταβλητές όταν το πρόγραμμα βρίσκεται σε λειτουργία. Η συγκεκριμένη μνήμη χάνει τα δεδομένα της σε απώλεια τροφοδοσίας.
- 1 KB μνήμη EEPROM: Μνήμη η οποία δεν χάνει τα δεδομένα της σε απώλεια τροφοδοσίας και χρησιμοποιείται για την εγγραφή ή ανάγνωση δεδομένων από το πρόγραμμα.

Ο μικροελεγκτής διαθέτει 14 ψηφιακούς ακροδέκτες εισόδου/εξόδου (εικόνα 18), οι οποίοι λειτουργούν στα 5 V και μπορούν να διαχειριστούν μέγιστο ρεύμα 20 mA. Επιπλέον διαθέτουν pull-up αντιστάσεις μεγέθους 20-50 kOhm. Κάποιοι ακροδέκτες επίσης μπορούν να έχουν επιπλέον λειτουργίες:

- Pins 0 και 1: Μπορούν να λειτουργήσουν ως σειριακή διεπαφή.
- Pins 2 και 3: Λειτουργούν ως εξωτερικά interrupt ώστε να σταματάει η κανονική ροή του προγράμματος και να εκτελούνται συγκεκριμένες ρουτίνες.
- Pins 3, 5, 6, 9, 10 και 11: Μπορούν να χρησιμοποιηθούν ως ψευδο-αναλογικοί είσοδοι χρησιμοποιώντας την διαμόρφωση PWM (Pulse Width Modulation, Παλμοευρική Διαμόρφωση).
- Pins 10, 11, 12 και 13: Υποστηρίζουν SPI (Serial Peripheral Interface, Σειριακή Διεπαφή Περιφερειακών) επικοινωνία.



Εικόνα 18: Ψηφιακοί ακροδέκτες, γείωσης και τάσης αναφοράς Arduino

Ο μικροελεγκτής επίσης διαθέτει 6 αναλογικές εισόδους (εικόνα 19) με ακρίβεια 10 bit (λαμβάνει δηλαδή 1024 διαφορετικές τιμές). Προεπιλεγμένη ανώτερη τάση είναι τα 5 V, η οποία μπορεί να αλλάξει τροφοδοτώντας τον ακροδέκτη AREF με την επιθυμητή τάση αναφοράς.



Εικόνα 19: Αναλογικοί ακροδέκτες Arduino

2.2.4 Προγραμματισμός Arduino Uno

Ο προγραμματισμός του Arduino Uno επιτυγχάνεται μέσω της γλώσσας προγραμματισμού Wiring, μιας παραλλαγής της C/C++ για μικροελεγκτές αρχιτεκτονικής AVR. Η δομή του προγράμματος αποτελείται από δύο βασικές συναρτήσεις, την `setup()`, η οποία εκτελείται μία φορά στην αρχή του προγράμματος, και την `loop()`, η οποία εκτελείται συνεχώς μέχρι να διακοπεί η τροφοδοσία της πλακέτας. Ένα παράδειγμα προγράμματος Arduino παρουσιάζεται στην εικόνα 20.

```
/*  
  Turns on an LED on for one second, then off for one second, repeatedly.  
  */  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin 13 as an output.  
  pinMode(13, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(13, HIGH);  // turn the LED on (HIGH is the voltage level)  
  delay(1000);             // wait for a second  
  digitalWrite(13, LOW);   // turn the LED off by making the voltage LOW  
  delay(1000);             // wait for a second  
}
```

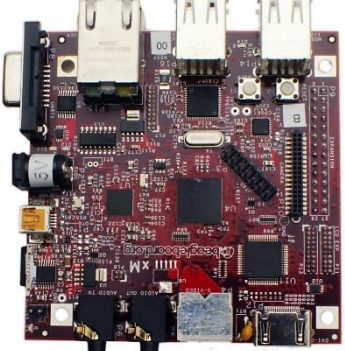
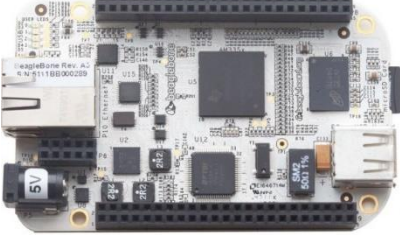
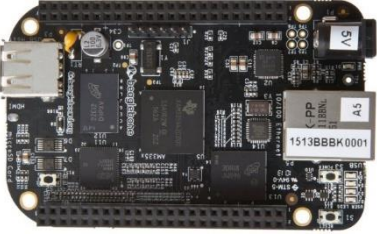
Εικόνα 20: Παράδειγμα προγράμματος Arduino

2.3 Αρχιτεκτονική BeagleBoard

Το BeagleBoard είναι μία χαμηλής κατανάλωσης, ανοιχτού κώδικα και υλικού υπολογιστική πλατφόρμα, η οποία κατασκευάζεται από την Texas Instruments σε συνεργασία με την Digi-Key και την Newark element14 [35]. Χρησιμοποιεί τον ARM Cortex-A8 επεξεργαστή και μπορεί να τρέξει Linux, Minix, FreeBSD, OpenBSD, Symbian και Android.

2.3.1 Είδη BeagleBoard

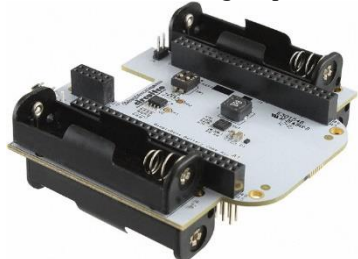
Στον παρακάτω πίνακα παρουσιάζονται τα BeagleBoard που κυκλοφορούν στο εμπόριο, μαζί με τα βασικά χαρακτηριστικά τους (πίνακας 4). Στην παρούσα διπλωματική επιλέχτηκε το BeagleBone Black.

Είδος BeagleBoard	Τεχνικά Χαρακτηριστικά
	Επεξεργαστής: Cortex-A8 + 2xPRU(200Mhz) Μνήμη Ram: 128MB DDR (rev B)/256MB DDR (rev C+) Μνήμη Flash: 256MB Συχνότητα επεξεργαστή: 720Mhz Συνδεσιμότητα: DVI-D, S-Video, USB, SD/MMC card slot, audio jack
	Επεξεργαστής: Cortex-A8 + 2xPRU(200Mhz) Μνήμη Ram: 512MB DDR2 Συχνότητα επεξεργαστή: 1000Mhz Συνδεσιμότητα: DVI-D, S-Video, 4 x USB, Ethernet, SD/MMC card slot, audio jack, camera port
	Επεξεργαστής: Cortex-A8 + 2xPRU(200Mhz) Μνήμη Ram: 256MB DDR2 Συχνότητα επεξεργαστή: 720Mhz Συνδεσιμότητα: 2 x 46 Pin Headers, USB, Ethernet, microSD card slot
	Επεξεργαστής: Cortex-A8 + 2xPRU(200Mhz) Μνήμη Ram: 512MB DDR3 Μνήμη eMMC: 2GB/4GB Συχνότητα επεξεργαστή: 1000Mhz Συνδεσιμότητα: 2 x 46 Pin Headers, USB, Ethernet, microSD card slot, micro-HDMI

Πίνακας 4: Είδη BeagleBoard

2.3.2 Πλακέτες επέκτασης BeagleBone (BeagleBone Capes)

Οι πλακέτες επέκτασης είναι πλακέτες που τοποθετούνται στο BeagleBone/BeagleBone Black και του προσθέτουν νέες δυνατότητες. Στον πίνακα 5 παρουσιάζονται μερικά από τα πιο διαδεδομένα Capes.

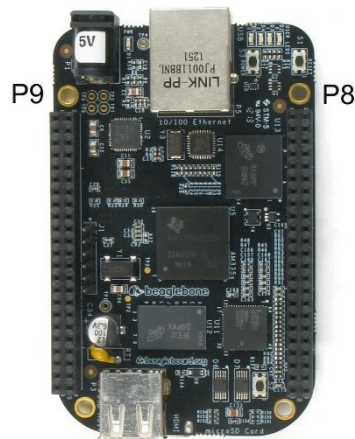
BeagleBone Cape	Περιγραφή
 <i>Εικόνα 25: LCD7 Cape [40]</i>	Πρόκειται για μία TFT LCD οθόνη αφής 7” με μέγιστη ανάλυση 800x400. Το BeagleBone τοποθετείται στο πίσω μέρος της οθόνης.
 <i>Εικόνα 26: Tracking Cape [41]</i>	Επιτρέπει τη λήψη του γεωγραφικού στίγματος καθώς επίσης και GSM λειτουργίες.
 <i>Εικόνα 27: Battery Cape [42]</i>	Παρέχει φορητή λύση τροφοδοσίας χρησιμοποιώντας 4 μπαταρίες λιθίου AA.

Πίνακας 5: Πλακέτες επέκτασης BeagleBone

2.3.3 Τεχνικά χαρακτηριστικά BeagleBone Black

Όπως προαναφέρθηκε, η πλακέτα που επιλέχθηκε για την ανάπτυξη της εργασίας είναι το BeagleBone Black λόγω του χαμηλού κόστους απόκτησης του, και του μεγάλου αριθμού ψηφιακών ακροδεκτών που διαθέτει.

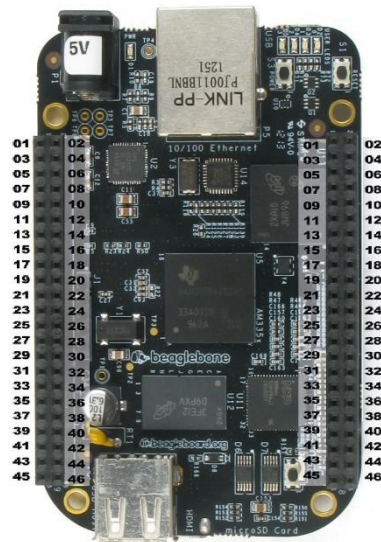
Η τροφοδοσία της πλακέτας πραγματοποιείται, είτε μέσω της θύρας USB, είτε μέσω εξωτερικής τροφοδοσίας 5 V που καταλήγει σε βύσμα διαμέτρου 5,5 mm με θετικό πόλο.



Εικόνα 28: BeagleBone Headers

Διαθέτει δύο Header, P8 και P9 (εικόνα 28), με το κάθε ένα να περιλαμβάνει 46 ακροδέκτες (εικόνα 28), οι οποίοι μπορούν να διαμορφωθούν ως εξής:

- Έως 65 ψηφιακοί ακροδέκτες εισόδου/εξόδου οι οποίοι λειτουργούν στα 3,3 V και μπορούν να διαχειριστούν μέγιστο ρεύμα 6 mA.
- Έως 8 μπορούν να λειτουργήσουν χρησιμοποιώντας την διαμόρφωση PWM .
- Έως 7 αναλογικοί είσοδοι με ακρίβεια 12 bit (λαμβάνει δηλαδή 4096 διαφορετικές τιμές), με μέγιστη επιτρεπτή τάση τα 1,8 V.
- Έως 5 ακροδέκτες Uart (Universal Asynchronous Receiver/Transmitter, Καθολικού Ασύγχρονου Πομποδέκτη).
- Έως 2 ακροδέκτες I2C (Inter-Integrated Circuit, Ενδιάμεσο Ολοκληρωμένο Κύκλωμα).
- Έως 2 ακροδέκτες SPI.
- Έως 25 ακροδέκτες χαμηλής καθυστέρησης PRU (Programmable Real-time Unit, Προγραμματιζόμενες Μονάδες Πραγματικού Χρόνου).



Εικόνα 29: Ακροδέκτες BeagleBone

Η πλακέτα επίσης ενσωματώνει μνήμη eMMC (embedded Multi-Media Controller, ενσωματωμένο Ελεγκτή Πολυμέσων) μεγέθους 4 GB και θήρα micro-HDMI. Ωστόσο για να γίνει χρήση αυτών, δεσμεύονται 10 ακροδέκτες από το BeagleBone για την eMMC και 20 ακροδέκτες για την micro-HDMI, καθώς θεωρούνται εικονικές πλακέτες επέκτασης. Για αυτόν τον λόγο απενεργοποιήθηκαν και προστέθηκε μία κάρτα microSD στην οποία αποθηκεύτηκε το λειτουργικό σύστημα. Τέλος, την σύνδεση του BeagleBone με το διαδίκτυο αναλαμβάνει η ενσωματωμένη κάρτα δικτύου 10/100 Mbps που διαθέτει.

2.3.4 Προγραμματισμός BeagleBone Black

Καθώς το BeagleBone Black αποτελεί μία υπολογιστική πλατφόρμα που εκτελεί λειτουργικό σύστημα Linux, υπάρχει πληθώρα επιλογών για τον προγραμματισμό του. Στην παρούσα εργασία χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python, λόγω του εύκολου χειρισμού των ψηφιακών ακροδεκτών μέσω μιας έτοιμης βιβλιοθήκης για τον σκοπό αυτό. Ένα παράδειγμα προγράμματος BeagleBone σε γλώσσα Python παρουσιάζεται στην εικόνα 30.

```
# Turns on an LED on for one second, then off for one second, repeatedly.
import Adafruit_BBIO.GPIO as GPIO
import time

GPIO.setup("P8_10", GPIO.OUT)

while True:
    GPIO.output("P8_10", GPIO.HIGH)
    time.sleep(1)
    GPIO.output("P8_10", GPIO.LOW)
    time.sleep(1)
```

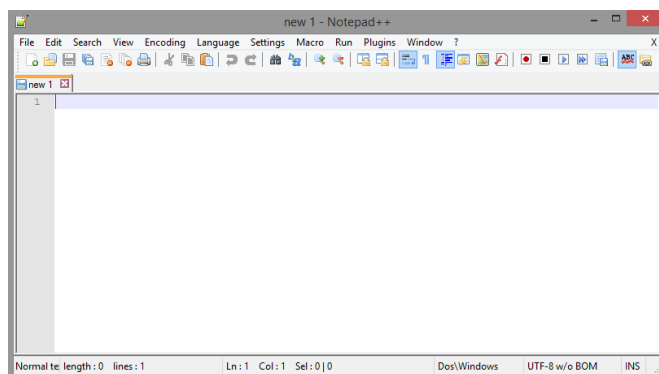
Εικόνα 30: Παράδειγμα προγράμματος BeagleBone σε Python

2.4 Πλατφόρμες ανάπτυξης λογισμικού

Στην ενότητα αυτή παρουσιάζονται οι εφαρμογές που χρησιμοποιήθηκαν για την ανάπτυξη του λογισμικού μέρους της εργασίας. Όλα τα εργαλεία είναι ανοιχτού κώδικα και διανέμονται δωρεάν, κάτι που συνέβαλε στον περιορισμό του κόστους ανάπτυξης της εργασίας.

2.4.1 Notepad++

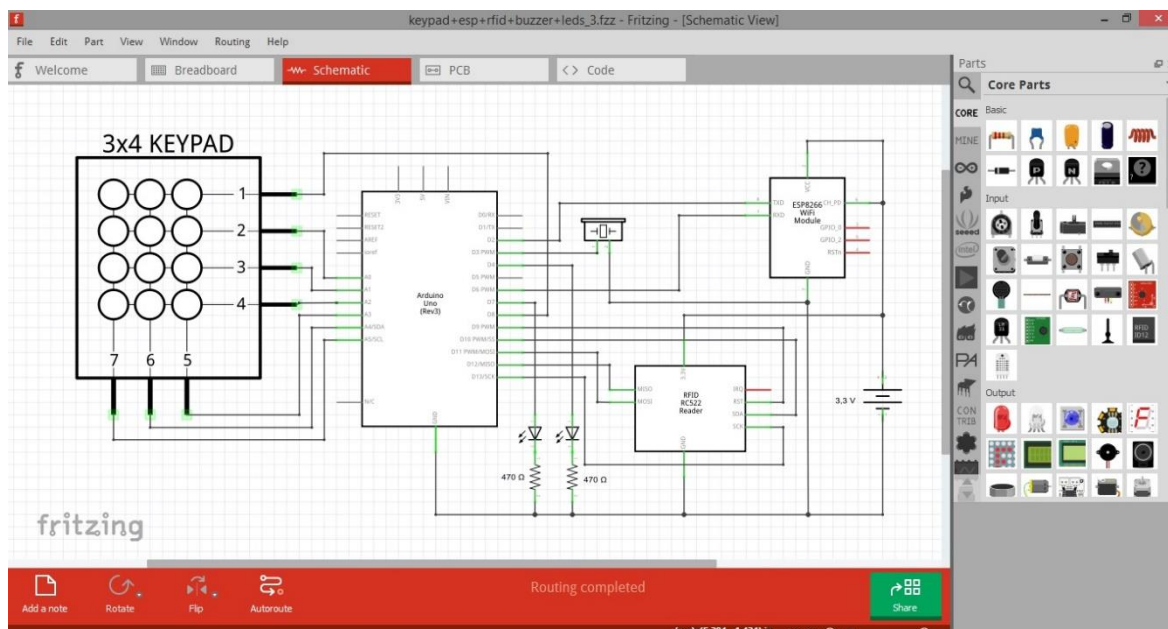
Το Notepad++ (εικόνα 31) είναι ένας ανοιχτού κώδικα επεξεργαστής κειμένου [43], ιδιαίτερα δημοφιλής στους προγραμματιστές καθώς προσφέρει διάφορες ευκολίες, όπως αυτόματη στοίχιση και χρωματισμό. Στην παρούσα διπλωματική χρησιμοποιήθηκε για την συγγραφή του κώδικα.



Εικόνα 31: Επεξεργαστής κειμένου Notepad++

2.4.2 Fritzing

Πρόκειται για μία ανοιχτού κώδικα πλατφόρμα σχεδίασης ηλεκτρονικών κυκλωμάτων για Windows, Mac OS και Linux [44]. Υποστηρίζει ένα μεγάλο πλήθος ηλεκτρονικών στοιχείων και μικροελεγκτών και, καθώς το περιβάλλον εργασίας του είναι ιδιαίτερα απλό, θεωρείται ένα από τα πιο δημοφιλή προγράμματα σχεδιασμού ηλεκτρονικών κυκλωμάτων. Στην εικόνα 32 παρουσιάζεται το περιβάλλον εργασίας του Fritzing.



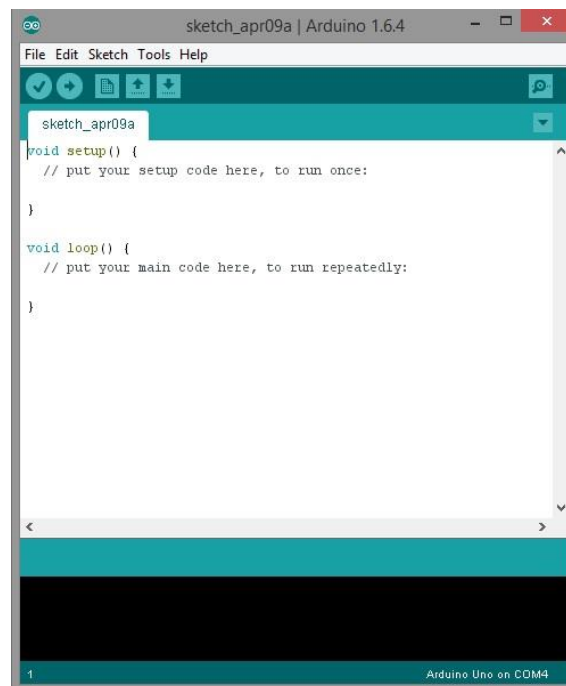
Εικόνα 32: Το περιβάλλον εργασίας του Fritzing

2.4.3 Arduino IDE

Το Arduino IDE (εικόνα 33) είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης που χρησιμοποιείται για την συγγραφή και αποστολή του κώδικα στο Arduino [45].

Συγκεκριμένα, το Arduino IDE παρέχει:

- Ένα περιβάλλον για τη συγγραφή των προγραμμάτων
- Διαχείριση βιβλιοθηκών
- Compiler (μεταγλωττιστής) για τη μεταγλώττιση των προγραμμάτων
- Μία σειριακή οθόνη (serial monitor)
- Την επιλογή για μεταφορά των μεταγλωττισμένων προγραμμάτων στο Arduino

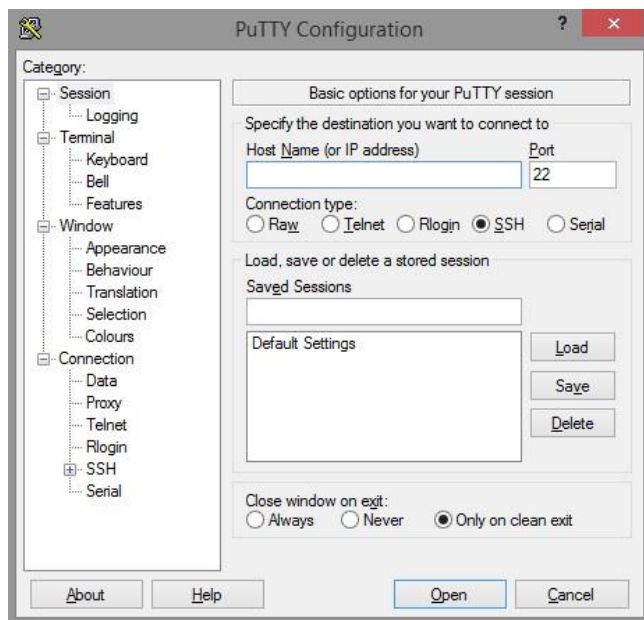


Εικόνα 33: Το περιβάλλον εργασίας του Arduino IDE

2.4.4 PuTTY

Το PuTTY (εικόνα 34) είναι ένας δωρεάν και ανοιχτού κώδικα εξομοιωτής τερματικού, σειριακή κονσόλα και εφαρμογή μεταφοράς αρχείων σε δίκτυα. Υποστηρίζει διάφορα

δικτυακά πρωτόκολλα όπως SCP, SSH, Telnet και rlogin και διατίθεται για Windows, Mac OS και Linux [46]. Χρησιμοποιήθηκε για τον χειρισμό του BeagleBone Black μέσω SSH.



Εικόνα 34: Το περιβάλλον εργασίας του PuTTY

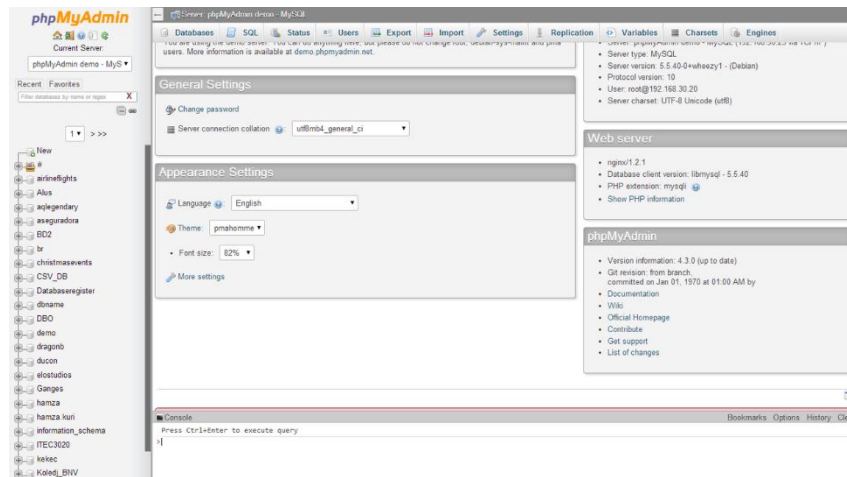
2.4.5 Apache HTTP Server

Ο apache είναι ο πιο δημοφιλής εξυπηρετητής του παγκόσμιου ιστού (web). Από το Νοέμβριο του 2015, εκτιμάτε πως εξυπηρετεί το 50% όλων των ενεργών ιστοσελίδων και το 37% των κορυφαίων ιστοτόπων παγκοσμίως [47]. Σε αυτόν φιλοξενήθηκε ο ιστοχώρος τους συστήματος.

2.4.6 phpMyAdmin

Το phpMyAdmin (εικόνα 35) είναι ένα δωρεάν και ανοιχτού κώδικα εργαλείο γραμμένο σε PHP, το οποίο αναλαμβάνει τη διαχείριση βάσεων δεδομένων MySQL και MariaDB μέσω του φυλλομετρητή. Μπορεί να εκτελέσει διάφορες ενέργειες, όπως την

δημιουργία, επεξεργασία ή διαγραφή βάσεων δεδομένων, πινάκων και πεδίων, την εκτέλεση SQL κώδικα και την διαχείριση χρηστών και δικαιωμάτων [48].



Εικόνα 35: Το περιβάλλον εργασίας του phpMyAdmin

2.4.7 Fsw webcam

Το fsw webcam είναι μια απλή εφαρμογή χειρισμού Web Cameras. Λαμβάνει εικόνες από μία V4L1/V4L2 συμβατή συσκευή ή αρχείο, αφαιρεί τον θόρυβο, προσθέτει λεζάντα με την ημερομηνία και τις αποθηκεύει σε αρχεία PNG ή JPEG [49]. Χρησιμοποιήθηκε για την λήψη διαδοχικών εικόνων από την Web Camera για την παρακολούθηση του χώρου.

2.4.8 GIMP

Το GIMP (GNU Image Manipulation Program) είναι ένα δωρεάν και ανοιχτού κώδικα πρόγραμμα επεξεργασίας εικόνας το οποίο διατίθεται για Windows, Mac Os και Linux [50]. Χρησιμοποιήθηκε για την επεξεργασία των εικόνων του ιστοχώρου και αυτού του κειμένου.

Κεφάλαιο 3 – Υλικό μέρος συστήματος

Σε αυτό το κεφάλαιο παρουσιάζεται και αναλύεται το υλικό μέρος της εργασίας. Η υλοποίηση της βασίστηκε στο BeagleBone Black, το οποίο αναλαμβάνει τον ρόλο της κεντρικής μονάδας του συναγερμού, και του Arduino Uno, το οποίο αποτελεί μέρος της ασύρματης μονάδας ελέγχου. Επιπλέον, χρησιμοποιήθηκαν αισθητήρια και συσκευές εξόδου, καθώς επίσης και άλλα ηλεκτρονικά στοιχεία για την σωστή λειτουργία του συστήματος.

Αναλυτικότερα στο BeagleBone Black συνδέθηκαν:

- Μαγνητικές επαφές συναγερμού
- Αισθητήρες PIR
- Σειρήνες συναγερμού
- Strobe συναγερμού
- Ενεργητικός υπέρυθρος ανιχνευτής (Infrared Beam)
- Web Camera
- Βομβητής
- LED

Στο Arduino Uno συνδέθηκαν:

- Πληκτρολόγιο
- Αναγνώστης RFID
- Σειριακός πομποδέκτης Wi-Fi
- Βομβητής
- LEDs

Στη συνέχεια παρουσιάζονται όλα τα παραπάνω εξαρτήματα, αναλύεται ο τρόπος λειτουργίας τους, τα τεχνικά χαρακτηριστικά τους και ο τρόπος με τον οποίο συνδέονται με το BeagleBone ή το Arduino.

3.1 Υλικό μέρος – BeagleBone

Το BeagleBone αποτελεί την κεντρική μονάδα του συναγερμού. Συνεπώς σε αυτό συνδέονται όλες οι συσκευές εισόδου και εξόδου, η Web Camera και ο βομβητής που μαζί με το LED ενημερώνουν τον χρήστη για την κατάσταση του συναγερμού. Χρησιμοποιήθηκαν κάποιοι ενδεικτικοί αισθητήρες, δύο σειρήνες και ένα Strobe. Δυνατή είναι επίσης και η προσθήκη πιο ειδικευμένων συσκευών, όπως ρελέδων και ανιχνευτών καπνού, ή συσκευών USB, όπως περισσότερων καμερών ή κάποιου 3g modem για την ενημέρωση του χρήστη μέσω SMS.

3.1.1 Μαγνητικές επαφές συναγερμού

Η μαγνητική επαφή συναγερμού (εικόνα 36) τοποθετείται σε πόρτες και παράθυρα, και ειδοποιεί την κεντρική μονάδα όταν αυτά είναι ανοιχτά. Λειτουργεί σαν ένας απλός διακόπτης ο οποίος ελέγχεται από ένα εφαρμοσμένο μαγνητικό πεδίο. Όταν τα δύο άκρα της μαγνητικής επαφής βρίσκονται κοντά, τότε ο διακόπτης είναι κλειστός, ενώ όταν απομακρυνθούν ο διακόπτης ανοίγει.

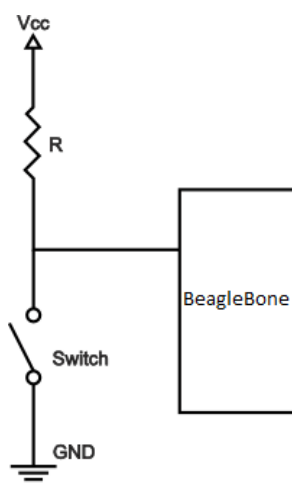


Εικόνα 36: Μαγνητική επαφή συναγερμού

Κάθε ακροδέκτης του BeagleBone που λειτουργεί ως είσοδος, θα πρέπει να βρίσκεται είτε στα 3,3 V (high), είτε στα 0 V (low). Εάν δεν συμβαίνει αυτό και ο ακροδέκτης είναι ασύνδετος, τότε το BeagleBone μπορεί να ερμηνεύσει την είσοδο είτε ως high, είτε ως low. Για

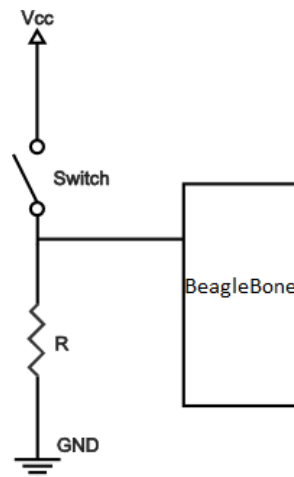
να αποτρέψουμε αυτό το γεγονός, μπορούμε να προσθέτουμε μία αντίσταση pull-up ή pull-down, η οποία εξασφαλίζει πως η είσοδος θα βρίσκεται πάντα σε κατάσταση high ή low.

Η συνδεσμολογία της αντίστασης pull-up παρουσιάζεται στην εικόνα 37 . Με αυτό τον τρόπο, όταν ο διακόπτης είναι ανοιχτός, η είσοδος του BeagleBone θα βρίσκεται στο λογικό high, ενώ όταν κλείσει, στο λογικό low. Η τιμή της αντίστασης θα πρέπει να ικανοποιεί δύο συνθήκες. Θα πρέπει να είναι μεγάλη, ώστε να περιορίζει το ρεύμα που διαρρέεται από αυτή όταν ο διακόπτης είναι κλειστός, και να μην είναι τόσο μεγάλη, κάτι που θα οδηγούσε το BeagleBone να μην αναγνωρίζει το λογικό high όταν ο διακόπτης είναι ανοιχτός. Γενικά, μία αντίσταση 3 KΩ-10 KΩ είναι ιδανική.



Εικόνα 37: Συνδεσμολογία pull-up αντίστασης

Αντίστοιχα με την pull-up, η pull-down αντίσταση κρατά χαμηλά την είσοδο του BeagleBone όταν ο διακόπτης είναι ανοιχτός, ενώ όταν κλείσει, η είσοδος θα είναι high. Η τιμή της θα πρέπει να είναι μεγάλη, ώστε να διαρρέεται από χαμηλής έντασης ρεύμα όταν ο διακόπτης είναι κλειστός, αλλά όχι τόσο μεγάλη ώστε η είσοδος να μην αναγνωρίζει το λογικό low όταν ο διακόπτης είναι ανοιχτός. Η συνδεσμολογία της pull-down αντίστασης παρουσιάζεται στην παρακάτω εικόνα (εικόνα 38).

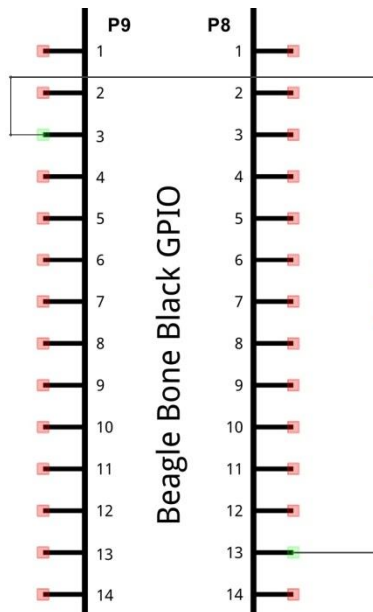


Εικόνα 38: Συνδεσμολογία pull-down αντίστασης

Καθώς το BeagleBone διαθέτει μερικούς ακροδέκτες με ενσωματωμένες pull-up και pull-down αντιστάσεις, δεν είναι απαραίτητη η προσθήκη τέτοιας αντίστασης σε αυτούς τους ακροδέκτες. Ο χρήστης μπορεί να ενημερωθεί εάν κάποιος ακροδέκτης διαθέτει ενσωματωμένη αντίσταση pull-up ή pull-down, επιλέγοντας τον ακροδέκτη στην προσθήκη, ή επεξεργασία, συσκευών.

Το σταθερό μέρος της μαγνητικής επαφής διαθέτει δύο επαφές. Η μία επαφή συνδέεται με τον ακροδέκτη του BeagleBone (μαζί με pull-up ή pull-down αντίσταση εάν δεν υπάρχει ενσωματωμένη) και η δεύτερη με τον ακροδέκτη 3,3 V (για pull-down αντίσταση) ή την γείωση (για pull-up αντίσταση).

Στην εικόνα 39 παρουσιάζεται το ηλεκτρονικό κύκλωμα μιας μαγνητικής επαφής, η οποία συνδέεται σε ακροδέκτη με ενσωματωμένη pull-down αντίσταση.



Εικόνα 39: Κύκλωμα μαγνητικής επαφής

3.1.2 Αισθητήρες κίνησης PIR

Αισθητήρας PIR (Passive InfraRed, Παθητικός Υπέρυθρος) είναι ένας ηλεκτρονικός αισθητήρας ο οποίος μετράει την υπέρυθη ακτινοβολία που εκπέμπεται από τα αντικείμενα γύρω του [51]. Ο όρος παθητικός αναφέρεται στο γεγονός ότι ο αισθητήρας δεν παράγει, ούτε ακτινοβολεί κάποια ενέργεια για την μέτρηση αυτή. Αντίθετα, χρησιμοποιεί την υπέρυθη ακτινοβολία η οποία εκπέμπεται ή αντανακλάται από τα άλλα αντικείμενα.

Η κύρια χρήση των αισθητήρων PIR είναι στους ανιχνευτές κίνησης τύπου PIR ή αλλιώς PID (Passive Infrared Detector, Παθητικός Υπέρυθρος Ανιχνευτής). Σε αυτή την περίπτωση, η αλλαγή στην τιμή της λαμβανόμενης ακτινοβολίας που μετράει ο αισθητήρας ενεργοποιεί την έξοδο του.

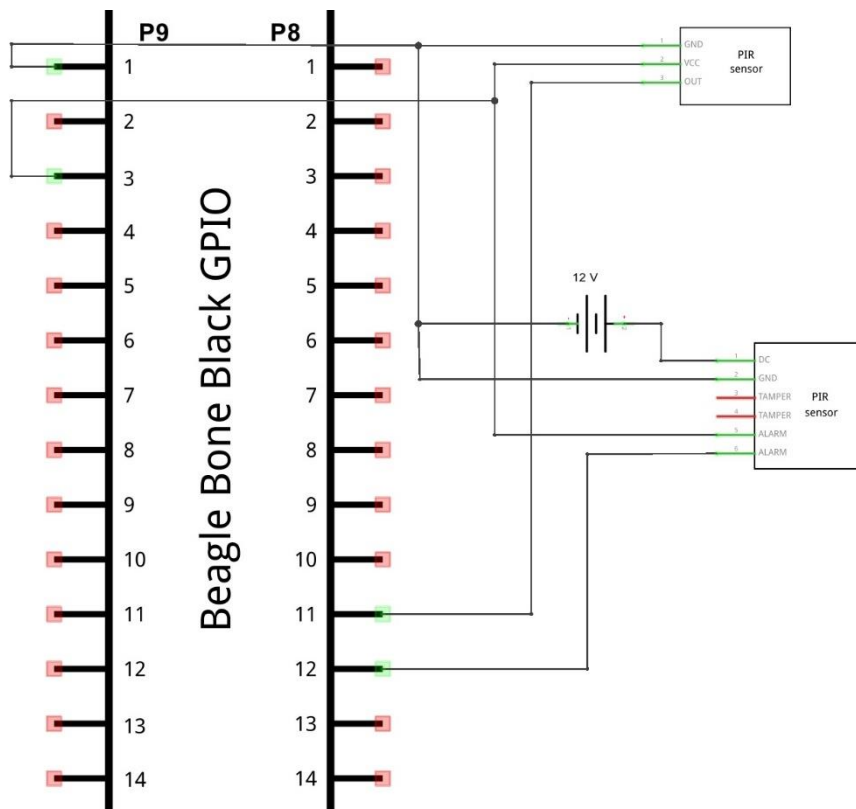
Χρησιμοποιήθηκαν δύο αισθητήρες κίνησης PIR (εικόνα 40). Ο πρώτος λειτουργεί με τάση 3,3 V και τροφοδοτείται από την έξοδο 3,3 V του BeagleBone (Pin 1 του Header P9). Σε κατάσταση αδράνειας η έξοδος του είναι στα 0 V, ενώ όταν ανιχνευτεί κίνηση στον χώρο γίνεται 3,3 V.



Εικόνα 40: Αισθητήρες κίνησης PIR

Ο δεύτερος αισθητήρας λειτουργεί στα 12 V και τροφοδοτείται από εξωτερική πηγή 12 V. Ως έξοδος υπάρχει μια ηλεκτρική επαφή [52], συμπεριφέρεται δηλαδή σαν ένας διακόπτης. Σε κατάσταση αδράνειας, οι επαφές εξόδου θα είναι ανοιχτό (NO) ή κλειστό κύκλωμα (NC), ανάλογα με την θέση του jumper [53] που διαθέτει, ενώ στην περίπτωση που ανιχνεύσει κίνηση, οι επαφές του θα έχουν την αντίθετη συμπεριφορά. Η συνδεσμολογία των επαφών είναι ανάλογη με την σύνδεση της μαγνητικής επαφής που αναλύθηκε προηγουμένως.

Στην εικόνα 41 παρουσιάζεται το ηλεκτρονικό κύκλωμα με τους δύο αισθητήρες. Είσοδοι στο BeagleBone είναι οι ακροδέκτες 11 και 12 του P8 Header, ενώ οι 3 και 1 του P9 είναι η παροχή 3,3 V και η γείωση, αντίστοιχα.



Εικόνα 41: Κύκλωμα αισθητήρων κίνησης PIR

3.1.3 Σειρήνες συναγερμού

Χρησιμοποιήθηκαν δύο σειρήνες συναγερμού (εικόνα 42), μία εσωτερικού χώρου και μία εξωτερικού, τροφοδοτούμενες και οι δύο από εξωτερική πηγή 12 V.

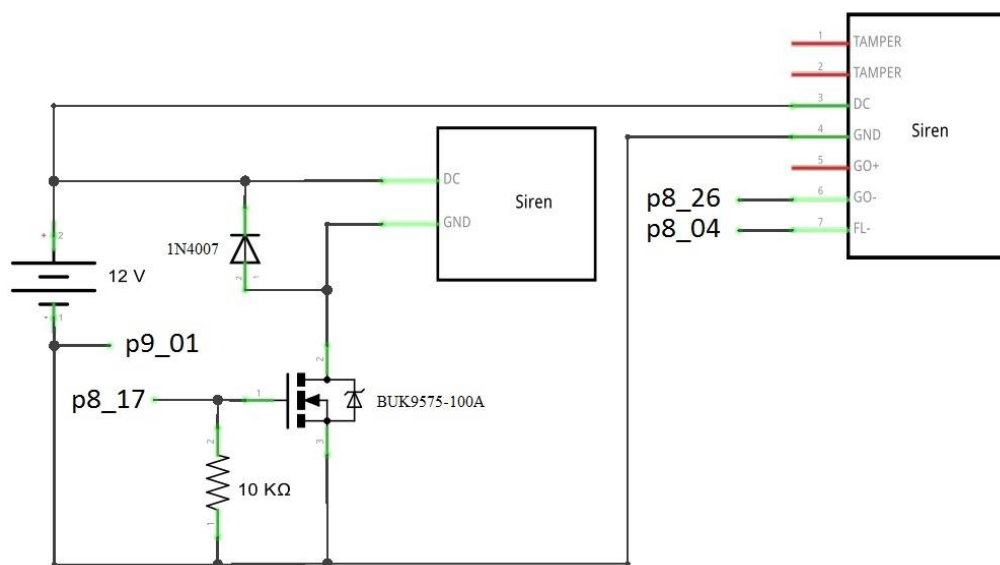


Εικόνα 42: Σειρήνες συναγερμού

Η σειρήνα εσωτερικού χώρου διαθέτει μόνο δύο ακροδέκτες (για την τροφοδοσία της), και συνεπώς ενεργοποιείται όταν εφαρμοστεί τάση 12 V σε αυτούς. Για αυτόν τον λόγο, χρησιμοποιήθηκε ένα N-MOSFET [54] τρανζίστορ και συγκεκριμένα το BUK9575-100A [55], για τον έλεγχο των 12 V από τον ακροδέκτη του BeagleBone. Επίσης, προστέθηκε η διάοδος 1N4007 [56], η οποία χρησιμοποιείται ως διάοδος σβέσης [57] για να προστατεύει το κύκλωμα από πιθανή ανάστροφη ηλεκτρεγερτική δύναμη. Τέλος, η αντίσταση 10 KΩ στο G του τρανζίστορ, κρατά χαμηλή την τάση στο G του τρανζίστορ όταν δεν στέλνει σήμα το BeagleBone.

Η σειρήνα εξωτερικού χώρου τροφοδοτείται επίσης από εξωτερική πηγή 12 V, διαθέτει όμως ηλεκτρονικό κύκλωμα για τον χειρισμό της. Μπορεί να ενεργοποιηθεί (μαζί με την λειτουργία φλας) εφαρμόζοντας τάση 0 V ή 12V στον ανάλογο ακροδέκτη, ή να ενεργοποιηθεί μόνο η λειτουργία φλας εφαρμόζοντας τάση 0 V σε τρίτο ακροδέκτη.

Στην εικόνα 43 παρουσιάζεται το κύκλωμα με τις δύο σειρήνες και τους ακροδέκτες του BeagleBone. Ο ακροδέκτης 26 του P8 Header όταν βρίσκεται στα 0 V ενεργοποιεί την εξωτερική σειρήνα (μαζί με την λειτουργία φλας), ενώ ο ακροδέκτης 4 του ίδιου Header, όταν βρεθεί στα 0 V ενεργοποιεί μόνο την λειτουργία φλας. Τέλος, η εσωτερική σειρήνα ενεργοποιείται εφαρμόζοντας τάση 3,3 V στον ακροδέκτη 17 του P8 Header.



Εικόνα 43: Κύκλωμα σειρήνων συναγερμού

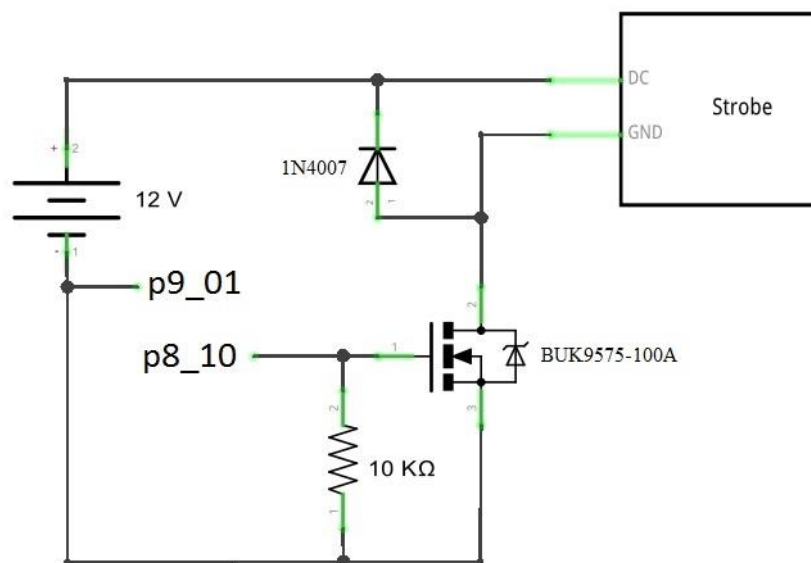
3.1.4 Strobe συναγερμού

Το strobe (εικόνα 44) είναι μια συσκευή που παράγει διαδοχικές λάμπσεις φωτός [58] η οποία μας προσφέρει οπτική ειδοποίηση σε περίπτωση συναγερμού. Η τροφοδοσία του γίνεται από εξωτερική πηγή 12 V και η συνδεσμολογία του είναι ανάλογη της σειρήνας εσωτερικού χώρου.



Εικόνα 44: Strobe συναγερμού

Στην εικόνα 45 παρουσιάζεται το κύκλωμα του Strobe μαζί με τους ακροδέκτες του BeagleBone. Ο ακροδέκτης 10 του P10 Header αποτελεί τον ακροδέκτη που ενεργοποιεί το Strobe, ενώ ο ακροδέκτης 1 του P9 Header είναι η γείωση.



Εικόνα 45: Κύκλωμα Strobe

3.1.5 Ενεργητικός υπέρυθρος ανιχνευτής (Infrared Beam)

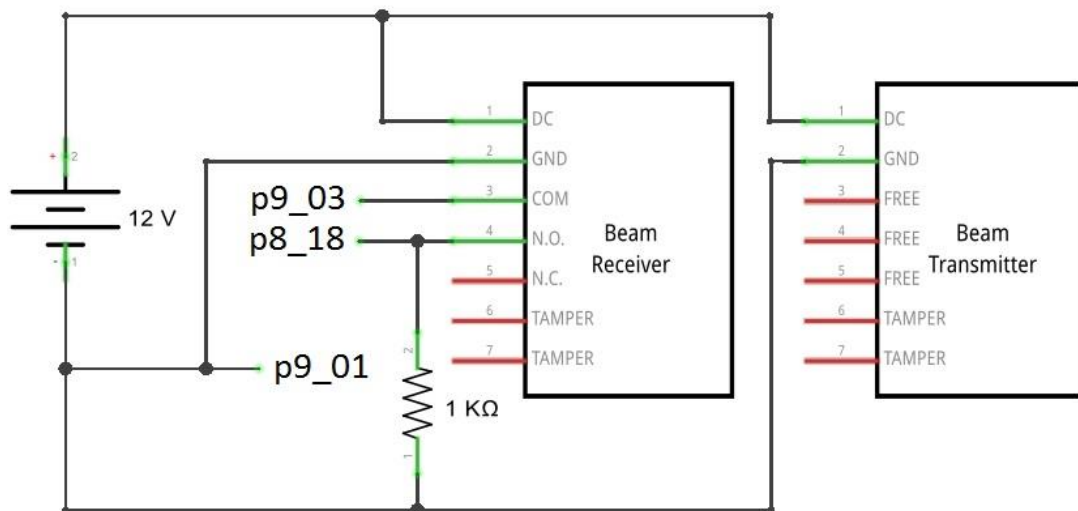
Αποτελείται από δύο μονάδες, την μονάδα εκπομπού και τον δέκτη. Ο εκπομπός στέλνει μία παλμική δέσμη υπέρυθρου φωτός στον δέκτη, η οποία αν διακοπεί από κάποιο αντικείμενο ενεργοποιεί την έξοδο του. Χρησιμοποιείται κυρίως σε εξωτερικούς χώρους, λόγω της μεγάλης απόστασης που μπορεί να καλύψει (100m στο Beam που χρησιμοποιήθηκε). Λειτουργεί στα 12 V και έχει δύο επαφές για την έξοδο του (NO και NC). Η συνδεσμολογία του είναι ανάλογη του αισθητήρα κίνησης PIR που αναλύθηκε προηγουμένως.

Ο υπέρυθρος ανιχνευτής που χρησιμοποιήθηκε στην παρούσα εργασία παρουσιάζεται στην παρακάτω εικόνα (εικόνα 46).



Εικόνα 46: Ενεργητικός υπέρυθρος ανιχνευτής

Στην εικόνα 47 παρουσιάζεται το ηλεκτρονικό κύκλωμα του υπέρυθρου ανιχνευτή, με τους ακροδέκτες 3 και 1 του P9 Header του BeagleBone να είναι η παροχή 3,3V και η γείωση αντίστοιχα, και τον ακροδέκτη 18 του P8 Header να είναι η είσοδος.



Εικόνα 47: Κύκλωμα υπέρυθρου ανιχνευτή

3.1.6 Web Camera

Η Web Camera (εικόνα 48) συνδέεται μέσω της θήρας USB του BeagleBone και προσφέρει στον χρήστη την δυνατότητα να παρακολουθεί ζωντανά τον χώρο στον οποίο είναι εγκατεστημένη.



Εικόνα 48: Web Camera

3.1.7 Βομβητής

Ο Βομβητής [59] χρησιμοποιείται για να ενημερώνει τον χρήστη ηχητικά όταν οπλίζεται ή αφοπλίζεται ο συναγερμός. Λόγω του ότι λειτουργεί στα 5 V και η έξοδος των

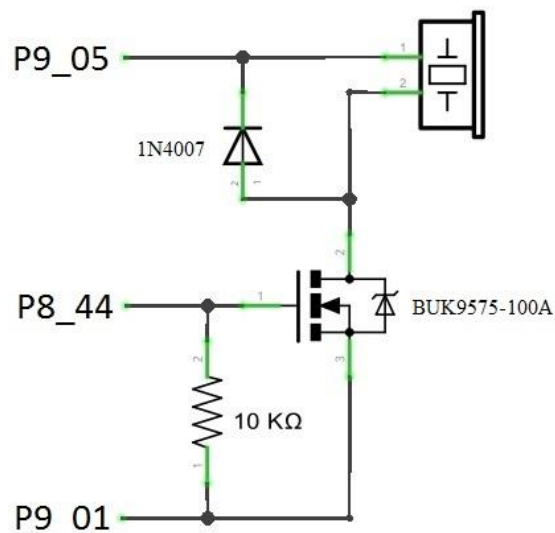
ψηφιακών ακροδεκτών του BeagleBone είναι 3,3 V, προστέθηκε ένα N-MOSFET τρανζίστορ για τον έλεγχο του, όπως αναλύθηκε στην σειρήνα εσωτερικού χώρου προηγουμένως. Τάση 5 V μας παρέχει το BeagleBone μέσω του ακροδέκτη 5 του P9 Header.

Ο βομβητής που χρησιμοποιήθηκε παρουσιάζεται στην παρακάτω εικόνα (εικόνα 49).



Εικόνα 49: Βομβητής

Στην εικόνα 50 παρουσιάζεται το ηλεκτρονικό κύκλωμα με τον βομβητή και τους ακροδέκτες του BeagleBone.



Εικόνα 50: Κύκλωμα βομβητή BeagleBone

3.1.8 LED

Το LED [60] ενημερώνει οπτικά τον χρήστη όταν είναι οπλισμένος ο συναγερμός. Χρησιμοποιείται επίσης και μία αντίσταση 470 Ω σε σειρά με το LED (εικόνα 51) για να περιορίζει το ρεύμα που το διαρρέει και να προστατεύει το BeagleBone και το LED.



Εικόνα 51: LED

Στην εικόνα 52 παρουσιάζεται το ηλεκτρονικό κύκλωμα με το Led και τους ακροδέκτες του BeagleBone.



Εικόνα 52: Κύκλωμα LED BeagleBone

Στην παρακάτω εικόνα παρουσιάζεται το τελικό κύκλωμα του BeagleBone μαζί με όλα τα εξαρτήματα (εικόνα 53).

3.2 Υλικό μέρος – Arduino

Το Arduino αποτελεί την απομακρυσμένη μονάδα ελέγχου με την οποία ο χρήστης μπορεί να οπλίζει και να αφοπλίζει τον συναγερμό. Αυτό μπορεί να το πραγματοποιηθεί χρησιμοποιώντας το πληκτρολόγιο ή τον αναγνώστη RFID που διαθέτει, ενώ η επικοινωνία με την κεντρική μονάδα πραγματοποιείται ασύρματα μέσω του σειριακού πομποδέκτη Wi-Fi. Τέλος, ο βομβητής και τα LED ενημερώνουν τον χρήστη για την κατάσταση του συναγερμού.

3.2.1 Πληκτρολόγιο

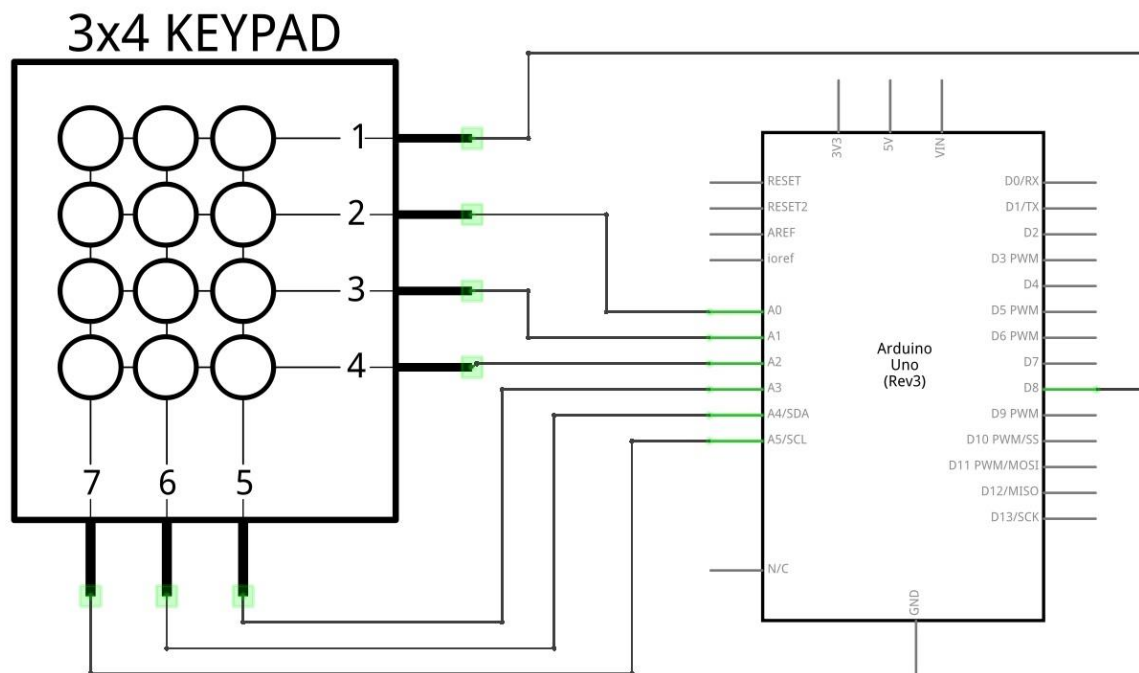
Εισάγοντας τον κωδικό pin στο αριθμητικό πληκτρολόγιο (εικόνα 54) του Arduino, ο χρήστης μπορεί να οπλίζει και να αφοπλίζει τον συναγερμό. Αποτελείται από δέκα πλήκτρα για τους πρώτους δέκα φυσικούς αριθμούς (0-9) και δύο επιπλέον για το αστεράκι (*) και την δέση (#).



Εικόνα 54: Αριθμητικό πληκτρολόγιο

Χρησιμοποιεί επτά ακροδέκτες, έναν για κάθε γραμμή και έναν για κάθε στήλη. Συνεπώς, με το πάτημα ενός πλήκτρου θα ενεργοποιηθούν δύο ακροδέκτες, αυτός που αντιστοιχεί στην γραμμή και αυτός που αντιστοιχεί στην στήλη.

Η σύνδεσή του με το Arduino γίνεται μέσω του ψηφιακού ακροδέκτη 8, και των αναλογικών ακροδεκτών 0, 1, 2, 3, 4 και 5. Το κύκλωμα του πληκτρολογίου, μαζί με το Arduino, παρουσιάζεται στην εικόνα 55.

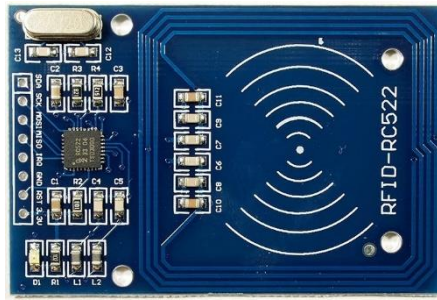


Εικόνα 55: Κύκλωμα αριθμητικού πληκτρολογίου

3.2.2 Ανάγνωσης RFID

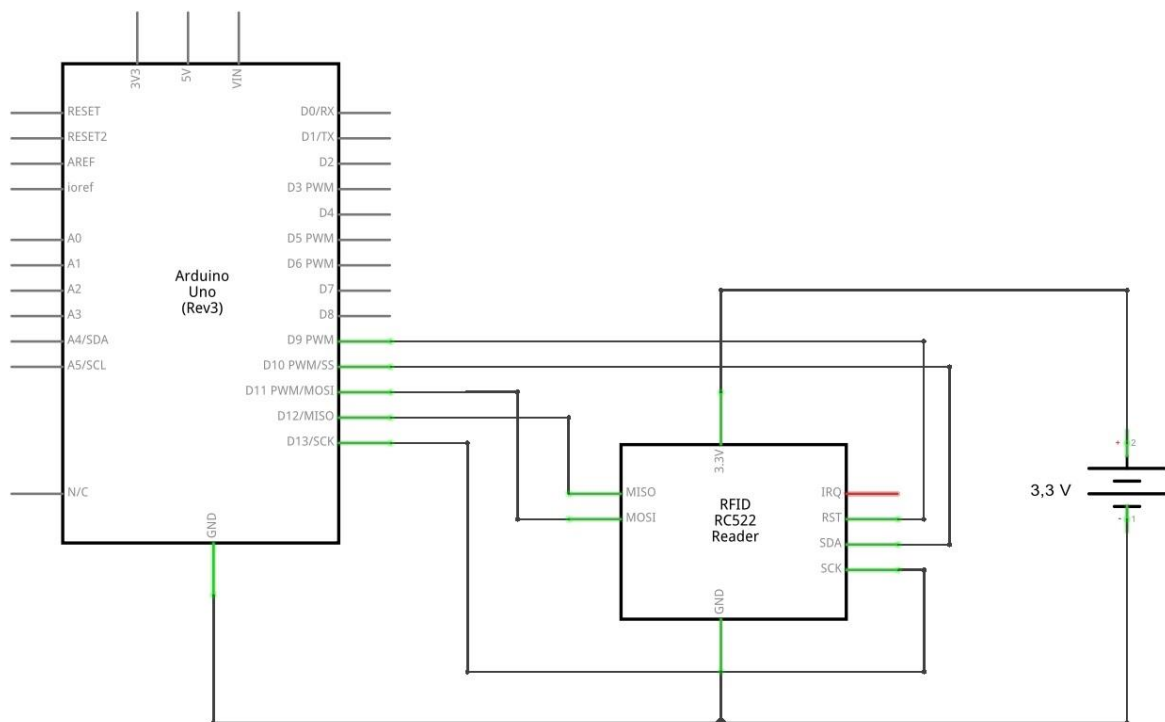
RFID (Radio Frequency IDentification, Ταυτοποίηση μέσω Ραδιοσυχνοτήτων) είναι η τεχνολογία που χρησιμοποιεί ραδιοκύματα για την ταυτοποίηση ανθρώπων ή αντικειμένων [61]. Χρησιμοποιεί μια συσκευή η οποία διαβάζει τα δεδομένα που λαμβάνει από μια ετικέτα RFID (RFID Tag) από απόσταση και χωρίς να απαιτείται οπτική επαφή. Η ετικέτα αυτή μπορεί να τροφοδοτείται από μπαταρία (ενεργό Tag) ή να λαμβάνει ενέργεια από τα ραδιοκύματα που εκπέμπει ο αναγνώστης RFID (παθητικό Tag).

Στην παρούσα εργασία επιλέχτηκε ο αναγνώστης MFRC522 [62], ο οποίος λειτουργεί στα 13,56 MHz και επικοινωνεί με παθητικά Tag τύπου ISO/IEC 14443 A/MIFARE, ενώ η σύνδεση με τον Arduino γίνεται μέσω της διεπαφής SPI στους ακροδέκτες 10, 11, 12 και 13, και του ακροδέκτη 9.



Εικόνα 56: Αναγνώστης RFID

Το κύκλωμα του αναγνώστη RFID (εικόνα 56), μαζί με το Arduino, παρουσιάζεται στην εικόνα 57.



Εικόνα 57: Κύκλωμα αναγνώστη RFID

3.2.3 Σειριακός πομποδέκτης Wi-Fi

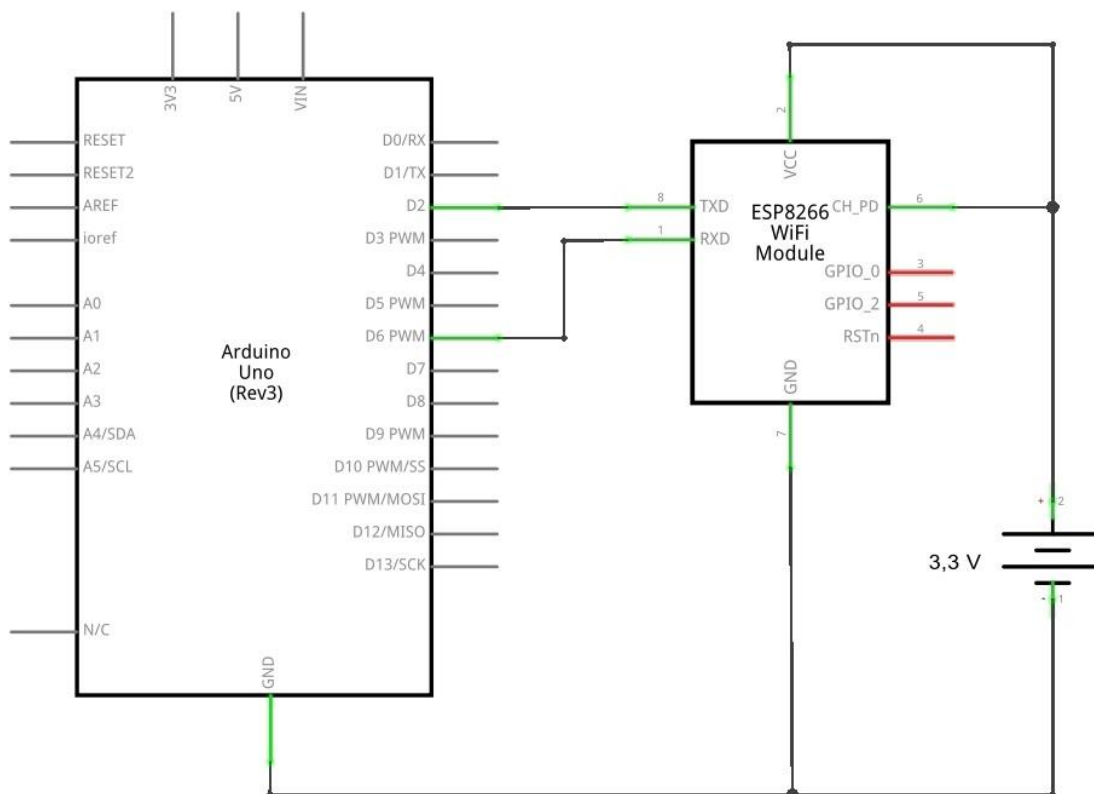
Την επικοινωνία του Arduino με την κεντρική μονάδα αναλαμβάνει ο σειριακός πομποδέκτης Wi-Fi. Χρησιμοποιήθηκε ο ESP8266 [63], ο οποίος μας προσφέρει ικανοποιητική εμβέλεια και χαμηλό κόστος απόκτησης. Η σύνδεση του με το Arduino γίνεται

σειριακά στους ακροδέκτες 2 και 6 (χρησιμοποιώντας την βιβλιοθήκη SoftwareSerial [64]), και η τροφοδοσία του γίνεται από εξωτερικό τροφοδοτικό 3,3 V, καθώς το Arduino δεν παρείχε ικανοποιητική ένταση ρεύματος για την ορθή λειτουργία του.



Εικόνα 58: ESP8266

Το κύκλωμα του πομποδέκτη Wi-Fi (εικόνα 58), μαζί με το Arduino, παρουσιάζεται στην εικόνα 59.

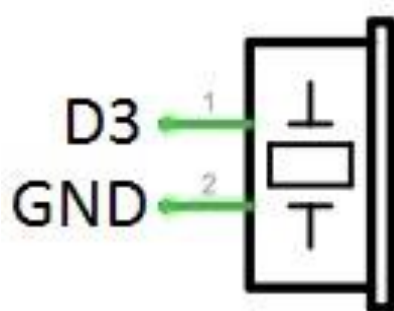


Εικόνα 59: Κύκλωμα ESP8266

3.2.4 Βομβητής

Αντίστοιχα με το BeagleBone, προστέθηκε ένας βομβητής 5 V για την ηχητική ενημέρωση του χρήστη όταν οπλίζεται ή αφοπλίζεται ο συναγερμός. Σε αυτήν την περίπτωση, ο ακροδέκτης του Arduino λειτουργεί στα 5 V, παρέχοντας ικανή ένταση ρεύματος για την λειτουργία του. Συνεπώς συνδέθηκε απευθείας στο Arduino, στον ακροδέκτη 3, χωρίς την χρήση κάποιου τρανζίστορ.

Στην εικόνα 60 παρουσιάζεται το ηλεκτρονικό κύκλωμα με τον βομβητή και τους ακροδέκτες του Arduino.

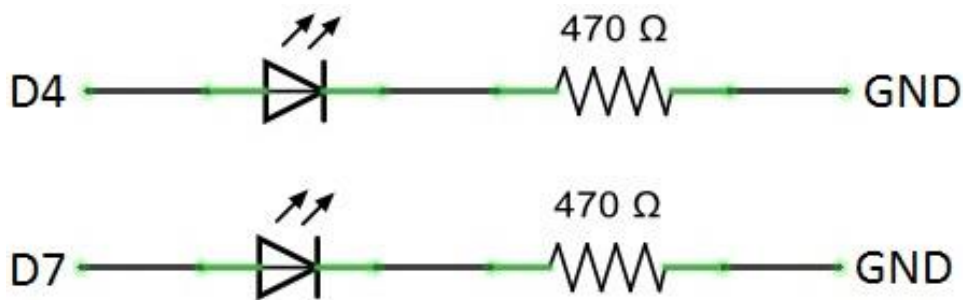


Εικόνα 60: Κύκλωμα βομβητή Arduino

3.2.5 LEDs

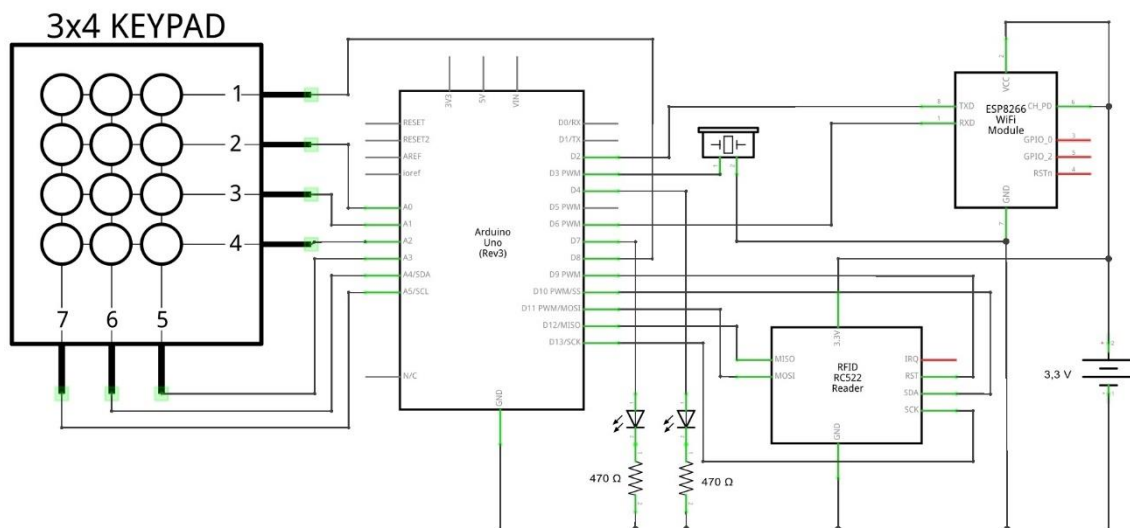
Χρησιμοποιήθηκαν δύο LED, ένα πράσινο και ένα κόκκινο, για την οπτική ενημέρωση του χρήστη της κατάστασης του συναγερμού. Το πρώτο είναι αναμμένο όταν υπάρχει επικοινωνία του Arduino με την κεντρική μονάδα του συναγερμού, και το δεύτερο όταν ο συναγερμός είναι οπλισμένος. Το πράσινο συνδέθηκε στο ακροδέκτη 4 του Arduino, και το κόκκινο στον ακροδέκτη 7. Όπως και με το BeagleBone, προστέθηκαν δύο αντιστάσεις 470 Ω σε σειρά για να περιορίσουν το ρεύμα που διαρρέει τα LED.

Στην εικόνα 61 παρουσιάζεται το ηλεκτρονικό κύκλωμα με τα LED και τους ακροδέκτες του Arduino.



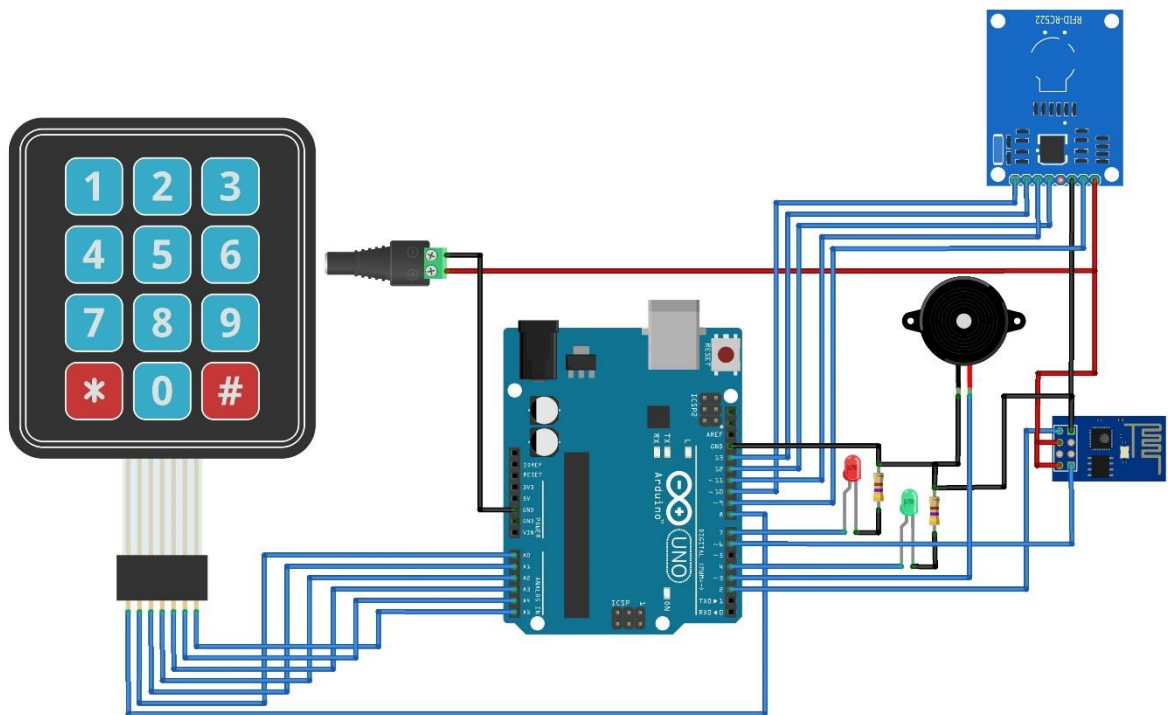
Εικόνα 61: Κύκλωμα LEDs Arduino

Στην παρακάτω εικόνα παρουσιάζεται το τελικό κύκλωμα του Arduino με όλα τα εξαρτήματα (εικόνα 62).



Εικόνα 62: Κύκλωμα Arduino

Στην εικόνα 63 παρουσιάζεται το τελικό κύκλωμα του Arduino με όλα τα εξαρτήματα σε προβολή Breadboard.



Εικόνα 63: Κόκλωμα Arduino, προβολή Breadboard

3.3 Υπολογισμός κόστους υλικών

Στο τέλος αυτού του κεφαλαίου, παρουσιάζονται όλα τα υλικά που χρησιμοποιήθηκαν μαζί με το κόστος τους (πίνακας 6).

Όνομα	Ποσότητα	Τιμή μονάδας (€)
BeagleBone Black	1	65,00
Κλώνος Arduino Uno R3	1	6,00
Αισθητήρας PIR - Parallax 555-28027	1	13,00
Αισθητήρας PIR - PIR-661	1	9,40
Μαγνητική Επαφή	5	0,65
Ενεργητικός Υπέρυθρος Ανιχνευτής	1	18,00
Σειρήνα - MARS-250	1	28,50
Σειρήνα - PS-913R	1	4,00
Strobe – SL-30R	1	9,90
Web Camera	1	10,00
Αναγνώστης RFID - MFRC522	1	2,70
Πομποδέκτης Wi-Fi - ESP8266	1	2,30
Αριθμητικό πληκτρολόγιο	1	2,00
Τροφοδοτικό 3,3V	1	1,30

Τροφοδοτικό 5V	1	5,00
Τροφοδοτικό 12V	1	8,00
Διάτρητη Πλακέτα 16,5cm x 5,6cm	2	5,00
Διάτρητη Πλακέτα 4,57cm x 3,56cm	1	2,00
Βομβητής 5V	2	0,50
LED 5mm	3	0,10
Κάρτα microSD 4GB	1	5,00
Καλώδιο Ethernet 2m	1	1,00
Καλώδιο Jumper 30cm πακέτο 30 τμχ.	1	3,60
Δίκλωνο καλώδιο 2 x 0.35mm 3m	1	0,60
Τρανζίστορ BUK9575-100A	3	0,90
Δίοδος 1N4007	3	0,02
Αντίσταση 470Ω, ¼ Watt	4	0,02
Αντίσταση 1KΩ, ¼ Watt	2	0,02
Αντίσταση 10KΩ, ¼ Watt	3	0,02
Συνολικό κόστος: 214,79 €		

Πίνακας 6: Κόστος υλικών

Κεφάλαιο 4 – Λογισμικό μέρος συστήματος

Σκοπός αυτού του κεφαλαίου είναι η ανάλυση του λογισμικού μέρους του συστήματος. Παρουσιάζεται το back-end λογισμικό που αναπτύχθηκε για την κεντρική μονάδα και την μονάδα απομακρυσμένου ελέγχου, ο ιστοχώρος, ο οποίος αποτελεί διεπαφή χρήστη του συστήματος, και η βάση δεδομένων.

Αναλυτικότερα, παρουσιάζονται:

- Οι απαιτήσεις του συστήματος.
- Η μεθοδολογία σχεδιασμού της βάσης δεδομένων.
- Οι λειτουργίες του ιστοχώρου, του ιστοχώρου για κινητές εφαρμογές και του webservice.
- Το λογισμικό της κεντρικής μονάδας και της μονάδας απομακρυσμένου ελέγχου.
- Μετρικά στοιχεία του κώδικα.
- Ανάλυση SWOT

4.1 Απαιτήσεις συστήματος

Σκοπός της παρούσας διπλωματικής εργασίας είναι η κατασκευή μιας έξυπνης κεντρικής μονάδας συναγερμού. Για τον σωστό σχεδιασμό του λογισμικού αρχικά καθορίστηκαν οι λειτουργικές, και μη λειτουργικές απαιτήσεις που πρέπει να ικανοποιούνται από το σύστημα.

Λειτουργικές απαιτήσεις είναι οι εργασίες που θα πρέπει να παρέχει το σύστημα στον χρήστη, ενώ μη λειτουργικές είναι τα χαρακτηριστικά και οι ιδιότητες που θα πρέπει να ικανοποιούνται από αυτό.

4.1.1 Λειτουργικές απαιτήσεις της κεντρικής μονάδας

Οι λειτουργικές απαιτήσεις της κεντρικής μονάδας είναι οι εξής:

- **Υποστήριξη πολλαπλών χρηστών.** Ο κάθε χρήστης θα έχει ένα username/password για τον έλεγχο της μονάδας συναγερμού μέσω διαδικτύου, έναν κωδικό pin και προαιρετικά ένα ή περισσότερα RFID Tag για τον έλεγχο μέσω των μονάδων απομακρυσμένου ελέγχου.
- **Θα πρέπει να υπάρχει η δυνατότητα για προσθήκη/επεξεργασία/διαγραφή συσκευής.** Ο χρήστης θα μπορεί να προσθέσει τις συσκευές εισόδου (αισθητήρια) ή τις συσκευές εξόδου (δράσης) που επιθυμεί, καθώς επίσης και να τα τροποποιήσει στο μέλλον, ή να τα αφαιρέσει.
- **Θα πρέπει να υπάρχει η δυνατότητα για προσθήκη/επεξεργασία/διαγραφή ομάδας συσκευών.** Θα υπάρχει η δυνατότητα για δημιουργία ομάδων συσκευών εισόδου και συσκευών εξόδου, ώστε να είναι ευκολότερη η διαχείρισή τους.
- **Όπλιση/αφόπλιση του συναγερμού και ενεργοποίηση/απενεργοποίηση των συσκευών εξόδου μέσω του ιστοχώρου:** Ο χρήστης θα μπορεί να χειριστεί τα σενάρια και να ενεργοποιεί τις συσκευές εξόδου μέσω του ιστοχώρου.
- **Όπλιση/αφόπλιση συναγερμού μέσω webservice αρχείου PHP.** Αυτό σημαίνει πως θα μπορεί να οπλιστεί ή να αφοπλιστεί κάποιο σενάριο επισκέπτοντας μια σελίδα με ορίσματα το διαπιστευτήριο του χρήστη, το όνομα του σεναρίου και την ενέργεια προς εκτέλεση.
- **Λειτουργία χρονοπρογραμματισμού.** Θα δίνεται η δυνατότητα όπλισης του συναγερμού ή ενεργοποίησης κάποιας συσκευής εξόδου επιλέγοντας συγκεκριμένη ημέρα και ώρα ενεργοποίησης και απενεργοποίησης.
- **Σελίδα για κινητά τηλέφωνα.** Θα δημιουργηθεί μια μινιμαλιστική σελίδα για κινητά τηλέφωνα με δυνατότητα όπλισης/αφόπλισης του συναγερμού, ενεργοποίησης/απενεργοποίησης των συσκευών εξόδου, προβολής της κατάστασης των συσκευών εισόδου, παρακολούθησης του χώρου μέσω κάμερας και προβολής των τελευταίων γεγονότων.
- **Ένδειξη στο website και τοπικά πότε είναι οπλισμένος ο συναγερμός.** Ο χρήστης θα μπορεί να πληροφορηθεί για την κατάσταση του συναγερμού είτε επισκέπτοντας το website, είτε μέσω του LED που θα υπάρχει στην κεντρική μονάδα. Επιπλέον, ηχητική ένδειξη θα τον ενημερώνει για αλλαγές στην κατάσταση του συναγερμού.

- **Δημιουργία σεναρίων αναλόγως των συνδυασμών των αισθητηρίων και των συνδυασμών των συσκευών δράσης.** Ο χρήστης θα δημιουργεί τα σενάρια λειτουργίας με τους δικούς του συνδυασμούς ανάλογα με τις ανάγκες του. Θα μπορεί να επιλέξει για κάθε συσκευή εισόδου, ή ομάδα συσκευών εισόδου διαφορετική καθυστέρηση ενεργοποίησης, καθυστέρηση δράσης και διάρκεια δράσης, καθώς επίσης και διαφορετική συσκευή εξόδου ή ομάδα συσκευών εξόδου ή εκτέλεση κάποιου script.
- **Δυνατότητα εκτέλεσης script αντί για ενεργοποίηση συσκευής εξόδου.** Θα υπάρχει η δυνατότητα για την εκτέλεση κάποιου shell script που ο χρήστης θα έχει δημιουργήσει, αντί για την ενεργοποίηση κάποιας συσκευής εξόδου. Το script θα έχει ως παραμέτρους την ώρα, το σενάριο και το αισθητήριο που το κάλεσε.
- **Δυνατότητα για επίσκεψη μιας σελίδας <http://> αντί για ενεργοποίηση συσκευής εξόδου.** Αυτό μπορεί πραγματοποιηθεί με το κατάλληλο shell script που ο χρήστης θα δημιουργήσει, και θα έχει ως παραμέτρους την ώρα, το σενάριο και το αισθητήριο που το κάλεσε.
- **Δυνατότητα για σιωπηλό συναγερμό.** Θα δίνεται η δυνατότητα στον χρήστη να δημιουργήσει κάποιο σενάριο το οποίο δεν θα ενεργοποιεί κάποια συσκευή εξόδου, αλλά θα εκτελεί τα script που έχει δηλώσει.
- **Δυνατότητα για σύνδεση οθόνης LCD με πληροφορίες της κατάστασης του συναγερμού.** Θα δημιουργηθεί μία ιδιόμορφη διαμορφωμένη σελίδα η οποία θα πληροφορεί τον χρήστη για την κατάσταση του συναγερμού. Η σελίδα αυτή θα μπορεί πολύ εύκολα να εμφανιστεί σε μία οθόνη LCD.
- **Χρήση κάμερας για παρακολούθηση του χώρου.** Ο χρήστης θα μπορεί να συνδεθεί μέσω διαδικτύου και να παρακολουθεί ζωντανά τον χώρο.
- **Προσωρινή απενεργοποίηση.** Ο χρήστης θα μπορεί να αφοπλίζει προσωρινά κάποιο σενάριο ή συγκεκριμένη συσκευή εξόδου για 0,5, 1, 2 και 5 λεπτά .
- **Δυνατότητα όπλισης για stay in.** Ο χρήστης θα μπορεί να δημιουργήσει κάποιο σενάριο το οποίο θα χρησιμοποιεί μόνο τα εξωτερικά αισθητήρια και τις μαγνητικές επαφές σε πόρτες και παράθυρα, έτσι ώστε να μπορεί να οπλίζει αυτό το σενάριο όταν βρίσκεται στον χώρο στον οποίο είναι εγκατεστημένος ο συναγερμός.

4.1.2 Μη λειτουργικές απαιτήσεις της κεντρικής μονάδας

Οι μη λειτουργικές απαιτήσεις της κεντρικής μονάδας είναι οι εξής:

- **Αδιάλειπτη λειτουργία.** Το σύστημα θα πρέπει να λειτουργεί απροβλημάτιστα όλο το χρόνο.
- **Ασφάλεια.** Μόνο εγγεγραμμένοι χρήστες θα πρέπει να έχουν πρόσβαση στις λειτουργίες του συναγερμού. Το σύστημα θα απαγορεύει την είσοδο χρηστών σε συγκεκριμένη διεύθυνση ip για πέντε λεπτά όταν υπάρξουν πέντε επαναλαμβανόμενες αποτυχημένες προσπάθειες εισόδου. Επιπλέον, θα εκτελούνται οι ενέργειες που ο χρήστης έχει ορίσει για τον σκοπό αυτό.
- **Υψηλή απόδοση.** Το σύστημα θα πρέπει να διεκπεραιώνει τις εργασίες άμεσα και χωρίς καθυστέρηση.

4.1.3 Λειτουργικές απαιτήσεις της μονάδας απομακρυσμένου ελέγχου

Οι λειτουργικές απαιτήσεις της μονάδας απομακρυσμένου ελέγχου είναι οι ακόλουθες:

- **Όπλιση/αφόπλιση του συναγερμού.** Ο χρήστης θα μπορεί να οπλίζει κάποιο σενάριο, ή να αφοπλίζει κάποιο ή όλα τα σενάρια εισάγοντας τον κωδικό pin ή την κάρτα RFID.
- **Ένδειξη πότε είναι οπλισμένος ο συναγερμός.** Μια ενδεικτική λυχνία και ένας βομβητής θα ενημερώνουν τον χρήστη όταν οπλίζεται ή αφοπλίζεται ο συναγερμός.
- **Ένδειξη όταν υπάρχει επικοινωνία με τη κεντρική μονάδα:** Ενδεικτική λυχνία θα είναι ενεργοποιημένη όταν υπάρχει σύνδεση με τη κεντρική μονάδα.

4.1.4 Μη λειτουργικές απαιτήσεις της μονάδας απομακρυσμένου ελέγχου

Οι μη λειτουργικές απαιτήσεις της μονάδας απομακρυσμένου ελέγχου είναι οι εξής:

- **Αδιάλειπτη λειτουργία.** Το σύστημα θα πρέπει να λειτουργεί απροβλημάτιστα όλο το χρόνο. Στην περίπτωση που η μονάδα χάσει τη σύνδεση της με το Access Point, θα επιχειρήσει άμεσα να επανασυνδεθεί.

- **Ασφάλεια.** Μόνο εγγεγραμμένοι χρήστες θα πρέπει να έχουν πρόσβαση στις λειτουργίες του συναγερμού. Όταν υπάρξουν πέντε επαναλαμβανόμενες αποτυχημένες προσπάθειες εισόδου, η συγκεκριμένη μονάδα απομακρυσμένου ελέγχου θα απενεργοποιείται για πέντε λεπτά.

4.2 Μεθοδολογία σχεδιασμού βάσης δεδομένων

Μετά την ανάλυση των απαιτήσεων, εξαιρετικής σημασίας είναι ο σωστός σχεδιασμός της βάσης δεδομένων, έτσι ώστε να είναι αποδοτική και να αποφεύγονται τα άσκοπα ερωτήματα. Στη συνέχεια παρουσιάζονται οι πίνακες της βάσης δεδομένων, καθώς και μια σύντομη περιγραφή των πεδίων τους.

4.2.1 Πίνακας users

Ο πίνακας users περιλαμβάνει όλα τα στοιχεία των εγγεγραμμένων χρηστών, εκτός των RFID Tags, τα οποία βρίσκονται στον πίνακα rfids. Ο πίνακας 7 παρουσιάζει την δομή του πίνακα users.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
user_id	int(11)		Όχι	καμία	auto_increment
username	varchar(20)	utf8_unicode_ci	Όχι	καμία	
password	varchar(11)	utf8_unicode_ci	Όχι	καμία	
type	tinyint(1)		Όχι	καμία	
pin	int(11)		Όχι	καμία	

Πίνακας 7: Η δομή του πίνακα users

user_id: Αποτελεί τον αναγνωριστικό αύξων αριθμό του χρήστη και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό χρηστών.

username: Το όνομα χρήστη, το οποίο είναι μοναδικό για κάθε χρήστη. Ο τύπος varchar(20) μας δίνει την δυνατότητα για ονόματα χρήστη μέγιστου μήκους 20 χαρακτήρων.

password: Ο κωδικός του χρήστη. Ο τύπος varchar(11) μας δίνει την δυνατότητα για κωδικούς μέγιστου μήκους 11 χαρακτήρων.

type: Το είδος του χρήστη. Η τιμή 1 υποδηλώνει απλό χρήστη, ενώ για διαχειριστή παίρνει την τιμή 2. Ο τύπος tinyint(1) επιλέχτηκε καθώς είναι ο μικρότερος σε μέγεθος τύπος ακεραίου.

pin: Ο κωδικός pin του κάθε χρήστη. Ο κωδικός αποτελείται από 4 αριθμούς και είναι μοναδικός για κάθε χρήστη. Για αυτόν το λόγο επιλέχτηκε ο τύπος int(4).

4.2.2 Πίνακας rfids

Ο πίνακας rfids περιλαμβάνει όλα τα RFID Tags των εγγεγραμμένων χρηστών. Ο πίνακας 8 παρουσιάζει την δομή του πίνακα rfids.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
user_id	int(11)		όχι	Καμία	
rfid	varchar(8)	utf8_unicode_ci	όχι	Καμία	
rfid_id	tinyint(1)		ναι	NULL	
in_use	tinyint(1)		όχι	Καμία	

Πίνακας 8: Η δομή του πίνακα rfids

id: Αποτελεί τον αναγνωριστικό αύξων αριθμό του RFID Tag και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό RFID Tag.

user_id: Το αναγνωριστικό του χρήστη το οποίο συνδέεται με το RFID Tag. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη user_id του πίνακα users.

rfid: Ο αναγνωριστικός αριθμός του RFID Tag. Επιλέχτηκε ο τύπος varchar(8) καθώς κάθε RFID Tag αποτελείται από 8 χαρακτήρες.

rfid_id: Αποτελεί έναν αναγνωριστικό αριθμό, ο οποίος συνδέεται με το συγκεκριμένο RFID Tag, έτσι ώστε ο χρήστης να μπορεί να διαχειρίζεται το RFID Tag γνωρίζοντας αυτόν τον

αριθμό. Έχει επιλεγεί ο τύπος `tinyint(1)` και το χαρακτηριστικό `unsigned` έτσι ώστε κάθε χρήστης να μπορεί να έχει 256 διαφορετικούς αναγνωριστικούς αριθμούς RFID Tag.

in_use: Παίρνει τιμή 0 όταν το συγκεκριμένο RFID Tag είναι ανενεργό και 1 όταν είναι ενεργό.

4.2.3 Πίνακας devices

Ο πίνακας `devices` περιλαμβάνει όλες τις εγκατεστημένες συσκευές εισόδου και εξόδου. Στον πίνακα 9 παρουσιάζεται η δομή του πίνακα `devices`.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
device_id	<code>int(11)</code>		όχι	καμία	<code>auto_increment</code>
device_name	<code>varchar(20)</code>	<code>utf8_unicode_ci</code>	όχι	καμία	
pin	<code>varchar(5)</code>	<code>utf8_unicode_ci</code>	όχι	καμία	
type	<code>tinyint(1)</code>		όχι	καμία	
normal_state	<code>tinyint(1)</code>		όχι	καμία	
live	<code>tinyint(1)</code>		ναι	NULL	

Πίνακας 9: Η δομή του πίνακα `devices`

device_id: Αποτελεί τον αναγνωριστικό αύξων αριθμό της συσκευής και το πρωτεύων κλειδί του πίνακα. Ο τύπος `int(11)` μας προσφέρει έναν μεγάλο αριθμό συσκευών.

device_name: Το όνομα της συσκευής. Ο τύπος `varchar(20)` μας δίνει την δυνατότητα για ονόματα μέγιστου μήκους 20 χαρακτήρων.

pin: Ο ακροδέκτης στον οποίο είναι εγκατεστημένη η συσκευή. Είναι αποθηκευμένο με την μορφή `<header>_<pin>`, με το `<header>` να παίρνει τιμές “P8” και “P9”, και το `<pin>` ένα διψήφιο αριθμό που αντιστοιχεί στο `pin` του BeagleBone. Για αυτόν τον λόγο επιλέχτηκε ο τύπος `varchar(5)` στον οποίο μπορεί να αποθηκεύσει αλφαριθμητικό μέγιστου μήκους 5 χαρακτήρων.

type: Παίρνει τιμή 0 όταν η συσκευή είναι συσκευή εισόδου και 1 όταν η συσκευή είναι συσκευή εξόδου.

normal_state: Η κατάσταση, στην οποία βρίσκετε η συσκευή σε κατάσταση αδράνειας. Παίρνει τιμή 0 όταν είναι `normal low` και 1 όταν είναι `normal high`.

live: Χρησιμοποιείται για την προσωρινή αποθήκευση της παρούσας κατάστασης των συσκευών εισόδου. Σε κατάσταση αδράνειας παίρνει τιμή 0, διαφορετικά την τιμή 1. Οι συσκευές εξόδου έχουν τιμή null σε αυτό το πεδίο.

4.2.4 Πίνακας deviceOn

Ο πίνακας deviceOn χρησιμοποιείται για την ενεργοποίηση και την απενεργοποίηση των συσκευών εξόδου ύστερα από επιλογή του χρήστη ή από τον χρονοπρογραμματισμό. Στον πίνακα 10 παρουσιάζεται η δομή του πίνακα deviceOn.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
device_id	int(11)		όχι	Καμία	
pid	int(11)		όχι	Καμία	
status	tinyint(1)		όχι	Καμία	

Πίνακας 10: Η δομή του πίνακα deviceOn

id: Αποτελεί τον αναγνωριστικό αύξων αριθμό της συσκευής εξόδου και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό συσκευών.

device_id: Το αναγνωριστικό της συσκευής εξόδου στην οποία αναφέρεται αυτή η εγγραφή. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη device_id του πίνακα devices.

pid: Σε αυτό το πεδίο αποθηκεύεται το PID (Process Identifier, Αναγνωριστικό Διεργασίας) της διεργασίας, η οποία δημιουργείται όταν ενεργοποιήσουμε την συγκεκριμένη συσκευή. Ο τύπος int(11) μας εξασφαλίζει την απροβλημάτιστη αποθήκευση αυτής της τιμής.

status: Η κατάσταση της συσκευής εξόδου. Παίρνει τιμή 0 όταν είναι απενεργοποιημένη, 1 όταν είναι ενεργοποιημένη, 2 όταν επιλέγουμε την απενεργοποίηση της συσκευής και 3 όταν επιλέγουμε την ενεργοποίηση της.

4.2.5 Πίνακας gpsetup

Ο πίνακας gpsetup (πίνακας 11) χρησιμοποιείται για την προσωρινή αποθήκευση της συσκευής που προσθέτουμε ή επεξεργαζόμαστε, έτσι ώστε το back-end λογισμικό να ορίσει τον ακροδέκτη της συσκευής ως είσοδο ή ως έξοδο. Στην συνέχεια, η εγγραφή για την συγκεκριμένη συσκευή αφαιρείται από τον πίνακα.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
gpio	varchar(5)	utf8_unicode_ci	όχι	Καμία	
type	tinyint(1)		όχι	Καμία	

Πίνακας 11: Η δομή του πίνακα gpsetup

id: Ο αναγνωριστικός αριθμός της συσκευής και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό συσκευών.

gpio: Ο ακροδέκτης στον οποίο είναι εγκατεστημένη η συσκευή. Είναι αποθηκευμένο με την μορφή <header>_<pin>, με το <header> να παίρνει τιμές “P8” και “P9”, και το <pin> έναν διψήφιο αριθμό που αντιστοιχεί στο pin του BeagleBone. Για αυτόν τον λόγο επιλέχτηκε ο τύπος varchar(5) στον οποίο μπορεί να αποθηκεύσει αλφαριθμητικό μέγιστου μήκους 5 χαρακτήρων.

type: Παίρνει τιμή 0 όταν η συσκευή είναι συσκευή εισόδου και 1 όταν η συσκευή είναι συσκευή εξόδου.

4.2.6 Πίνακας groups

Ο πίνακας groups περιλαμβάνει τα ονόματα των ομάδων συσκευών εισόδου και εξόδου. Στον πίνακα 12 παρουσιάζεται η δομή του πίνακα groups.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
group_id	int(11)		όχι	καμία	auto_increment
group_name	varchar(20)	utf8_unicode_ci	όχι	καμία	
type	tinyint(1)		όχι	καμία	

Πίνακας 12: Η δομή του πίνακα groups

group_id: Ο αναγνωριστικός αριθμός της ομάδας συσκευών και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό ομάδων.

group_name: Το όνομα της ομάδας. Ο τύπος varchar(20) μας δίνει την δυνατότητα για ονόματα μέγιστου μήκους 20 χαρακτήρων.

type: Ο τύπος της ομάδας. Παίρνει τιμή 0 για ομάδα συσκευών εισόδου και 1 για ομάδα συσκευών εξόδου.

4.2.7 Πίνακας <group_name>_group

Δημιουργείται δυναμικά όταν προσθέτουμε καινούρια ομάδα και διαγράφεται όταν την αφαιρούμε. Περιέχει τις συσκευές που περιλαμβάνονται στην συγκεκριμένη ομάδα.

Παρακάτω παρουσιάζεται η δομή του πίνακα (πίνακας 13).

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
<u>id</u>	int(11)		όχι	καμία	auto_increment
device_id	int(11)		όχι	καμία	

Πίνακας 13: Η δομή του πίνακα <group_name>_group

id: Ο αναγνωριστικός αύξων αριθμός της συσκευής που αποτελεί μέρος της ομάδας και το πρωτεύων κλειδί του πίνακα.

device_id: Το αναγνωριστικό της συσκευής η οποία ανήκει στην συγκεκριμένη ομάδα συσκευών. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη device_id του πίνακα devices.

4.2.8 Πίνακας scripts

Ο πίνακας scripts περιλαμβάνει τα ονόματα των αποθηκευμένων scripts. Στον πίνακα 14 παρουσιάζεται η δομή του πίνακα scripts.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
script_id	int(11)		όχι	καμία	auto_increment
script_name	varchar(20)	utf8_unicode_ci	όχι	καμία	

Πίνακας 14: Η δομή του πίνακα scripts

script_id: Ο αναγνωριστικός αύξων αριθμός του script και το πρωτεύων κλειδί του πίνακα.

script_name: Το όνομα του script. Ο τύπος varchar(20) μας δίνει την δυνατότητα για ονόματα μέγιστου μήκους 20 χαρακτήρων.

4.2.9 Πίνακας scenario_list

Ο πίνακας scenario_list περιλαμβάνει τα ονόματα των αποθηκευμένων σεναρίων χρήσης. Στον πίνακα 15 παρουσιάζεται η δομή του πίνακα.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
scenario	varchar(20)	utf8_unicode_ci	όχι	Καμία	
status	tinyint(1)		όχι	Καμία	
temp_disable	int(20)		όχι	Καμία	
change	tinyint(1)		όχι	Καμία	

Πίνακας 15: Η δομή του πίνακα scenario_list

id: Ο αναγνωριστικός αριθμός του σεναρίου και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό σεναρίων.

scenario: Το όνομα του σεναρίου. Ο τύπος varchar(20) μας δίνει την δυνατότητα για ονόματα μέγιστου μήκους 20 χαρακτήρων.

status: Η κατάσταση λειτουργίας του συγκεκριμένου σεναρίου. Παίρνει τιμή 0 όταν είναι αποπλισμένο και 1 όταν είναι οπλισμένο.

temp_disable: Παίρνει την τιμή 1 όταν έχουμε αποπλίζει προσωρινά όλο ή κάποια συσκευή εισόδου του σεναρίου. Διαφορετικά, έχει την τιμή 0.

change: Παίρνει την τιμή 1 όταν οπλίζουμε ή αποπλίζουμε όλο. ή κάποια συσκευή εισόδου του σεναρίου. Διαφορετικά, έχει την τιμή 0.

4.2.10 Πίνακας <scenario_name>_scenario

Δημιουργείται δυναμικά όταν προσθέσουμε νέο σενάριο και διαγράφεται όταν το αφαιρέσουμε. Περιλαμβάνει όλες τις πληροφορίες που σχετίζονται με το συγκεκριμένο σενάριο (πίνακας 16).

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
activation_delay	int(11)		όχι	Καμία	
action_delay	int(11)		όχι	Καμία	
action_time	int(11)		όχι	Καμία	
input_type	tinyint(1)		όχι	Καμία	
input_device_id	int(11)		ναι	NULL	
input_group_id	int(11)		ναι	NULL	
and_sensors	tinyint(1)		όχι	Καμία	
output_type	tinyint(1)		όχι	Καμία	
output_device_id	int(11)		ναι	NULL	
output_group_id	int(11)		ναι	NULL	
script_id	int(11)		ναι	NULL	
pid	int(11)		όχι	Καμία	
status	tinyint(1)		όχι	Καμία	
temp_disable_time	int(11)		ναι	NULL	

Πίνακας 16: Η δομή του πίνακα <scenario_name>_scenario

id: Ο αναγνωριστικός αριθμός της συγκεκριμένης εγγραφής και το πρωτεύων κλειδί του πίνακα.

activation_delay: Ο χρόνος σε δευτερόλεπτα για την ενεργοποίηση της συγκεκριμένης εγγραφής, από την στιγμή που θα επιλέξουμε την όπλιση της.

action_delay: Ο χρόνος σε δευτερόλεπτα για την ενεργοποίηση της συσκευής εξόδου, από την στιγμή που θα ενεργοποιηθεί η συσκευή εισόδου.

action_time: Ο χρόνος σε δευτερόλεπτα που θα παραμείνει ενεργή η συσκευή εξόδου, ύστερα από την ενεργοποίηση της συσκευής εισόδου της συγκεκριμένης εγγραφής.

input_type: Το είδος της συσκευής εισόδου. Παίρνει τιμή 0 όταν πρόκειται για απλή συσκευή εισόδου, 1 για ομάδα συσκευών εισόδου και 2 όταν επιλέξουμε όλες τις συσκευές εισόδου.

input_device_id: Το αναγνωριστικό της συσκευής εισόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εισόδου κάποια ομάδα ή όλες τις συσκευές παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη device_id του πίνακα devices.

input_group_id: Το αναγνωριστικό της ομάδας των συσκευών εισόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εισόδου κάποια συσκευή ή όλες τις συσκευές παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη group_id του πίνακα groups.

and_sensors: Παίρνει τιμή 1 όταν έχουμε συνδυασμό περισσότερων συσκευών εισόδου με το λογικό “και“, οι οποίες αποθηκεύονται στον πίνακα <scenario_name>_andSensors. Διαφορετικά παίρνει την τιμή 0.

output_type: Το είδος της συσκευής εξόδου. Παίρνει τιμή 0 όταν πρόκειται για απλή συσκευή εξόδου, 1 για ομάδα συσκευών εξόδου, 2 για εκτέλεση script, και 3 όταν επιλέξουμε όλες τις συσκευές εξόδου.

output_device_id: Το αναγνωριστικό της συσκευής εξόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια ομάδα, εκτέλεση script, ή όλες τις συσκευές, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη device_id του πίνακα devices.

output_group_id: Το αναγνωριστικό της ομάδας συσκευών εξόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια συσκευή, εκτέλεση script, ή όλες τις συσκευές, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη group_id του πίνακα groups.

script_id: Το αναγνωριστικό του script προς εκτέλεση. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια συσκευή, ομάδα συσκευών, ή όλες τις συσκευές, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη script_id του πίνακα scripts.

pid: Σε αυτό το πεδίο αποθηκεύεται το PID της διεργασίας η οποία δημιουργείται όταν ενεργοποιήσουμε την συγκεκριμένη εγγραφή του σεναρίου. Ο τύπος int(11) μας εξασφαλίζει την απροβλημάτιστη αποθήκευση αυτής της τιμής.

status: Η κατάσταση λειτουργίας της συγκεκριμένης εγγραφής. Παίρνει τιμή 0 όταν είναι αφοπλισμένη, 1 όταν είναι οπλισμένη, 2 όταν επιλέγουμε την αφόπλιση της, 3 όταν επιλέγουμε την όπλιση της, 4 όταν επιλέγουμε την αφόπλιση όλων των εγγραφών του σεναρίου και 5 όταν επιλέγουμε την όπλιση όλων των εγγραφών.

temp_disable_time: η χρονική στιγμή σε UNIX time κατά τον οποίο η εγγραφή θα οπλιστεί, ύστερα από την προσωρινή αφόπλιση της.

4.2.11 Πίνακας <scenario_name>_andSensors

Δημιουργείται δυναμικά όταν προσθέσουμε νέο σενάριο και διαγράφεται όταν το αφαιρούμε. Περιλαμβάνει τις συσκευές εισόδου οι οποίες συνδέονται με το λογικό “και” με κάποια συσκευή από τον πίνακα <scenario_name>_name. Η δομή του παρουσιάζεται στον πίνακα 17.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
scenario_id	int(11)		όχι	Καμία	
input_type	tinyint(1)		όχι	Καμία	
input_device_id	int(11)		ναι	NULL	
input_group_id	int(11)		ναι	NULL	
window_time	int(11)		όχι	Καμία	

Πίνακας 17: Η δομή του πίνακα <scenario_name>_andSensors

id: Ο αναγνωριστικός αριθμός της συγκεκριμένης εγγραφής και το πρωτεύων κλειδί του πίνακα.

scenario_id: Ο αναγνωριστικός αριθμός του πίνακα <scenario_name>_scenario στον οποίο αναφέρεται η συγκεκριμένη εγγραφή.

input_type: Το είδος της συσκευής εισόδου. Παίρνει τιμή 0 όταν πρόκειται για απλή συσκευή εισόδου και 1 για ομάδα συσκευών εισόδου.

input_device_id: Το αναγνωριστικό της συσκευής εισόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εισόδου κάποια ομάδα, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη device_id του πίνακα devices.

input_group_id: Το αναγνωριστικό της ομάδας των συσκευών εισόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εισόδου κάποια συσκευή, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη group_id του πίνακα groups.

window_time: Το χρονικό παράθυρο σε δευτερόλεπτα μέσα στο οποίο αν ενεργοποιηθούν όλες οι συσκευές εισόδου θα ενεργοποιηθούν την συσκευή εξόδου.

4.2.12 Πίνακας scenarioOnOff

Ο πίνακας scenarioOnOff περιλαμβάνει όλες τις ενέργειες που εκτελούνται όταν οπλίσουμε ή αφοπλίσουμε κάποιο σενάριο. Στον πίνακα 18 παρουσιάζεται η δομή του πίνακα scenarioOnOff.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
scenario_id	int(11)		όχι	Καμία	
status	tinyint(1)		όχι	Καμία	
action_delay	int(11)		όχι	Καμία	
output_type	tinyint(1)		όχι	Καμία	
output_device_id	int(11)		ναι	NULL	
output_group_id	int(11)		ναι	NULL	
script_id	int(11)		ναι	NULL	
action_time	int(11)		όχι	Καμία	

Πίνακας 18: Η δομή του πίνακα scenarioOnOff

id: Ο αναγνωριστικός αριθμός της συγκεκριμένης εγγραφής και το πρωτεύων κλειδί του πίνακα.

scenario_id: Ο αναγνωριστικός αριθμός του σεναρίου στο οποίο αναφέρεται η συγκεκριμένη εγγραφή.

status: Το είδος της ενέργειας προς εκτέλεση. Λαμβάνει τιμή 0 για εκτέλεση της ενέργειας όταν οπλίσουμε το σενάριο, και 1 για εκτέλεση κατά την αφοπλισή του σεναρίου.

action_delay: Ο χρόνος σε δευτερόλεπτα για την ενεργοποίηση της συσκευής εξόδου, από την στιγμή που θα οπλίσουμε ή αφοπλίσουμε το σενάριο.

output_type: Το είδος της συσκευής εξόδου. Παίρνει τιμή 0 όταν πρόκειται για απλή συσκευή εξόδου, 1 για ομάδα συσκευών εξόδου, 2 για εκτέλεση script, και 3 όταν επιλέξουμε όλες τις συσκευές εξόδου.

output_device_id: Το αναγνωριστικό της συσκευής εξόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια ομάδα, εκτέλεση script, ή όλες τις συσκευές, παίρνει τιμή NULL.

output_group_id: Το αναγνωριστικό της ομάδας συσκευών εξόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια συσκευή, εκτέλεση script, ή όλες τις συσκευές, παίρνει τιμή NULL.

script_id: Το αναγνωριστικό του script προς εκτέλεση. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια συσκευή, ομάδα συσκευών, ή όλες τις συσκευές, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη script_id του πίνακα scripts.

action_time: Ο χρόνος σε δευτερόλεπτα που θα παραμείνει ενεργή η συσκευή εξόδου.

4.2.13 Πίνακας keypads

Ο πίνακας keypads περιλαμβάνει όλες τις αποθηκευμένες μονάδες απομακρυσμένου ελέγχου. Στον πίνακα 19 παρουσιάζεται η δομή του πίνακα keypads.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
keypad_name	varchar(20)	utf8_unicode_ci	όχι	Καμία	
type	tinyint(1)		όχι	Καμία	
scenario_id_enable	int(11)		ναι	NULL	
scenario_id_disable	int(11)		ναι	NULL	
inUse	int(1)		όχι	Καμία	
status	varchar(3)	utf8_unicode_ci	όχι	Καμία	

Πίνακας 19: Η δομή του πίνακα keypads

id: Ο αναγνωριστικός αριθμός της μονάδας απομακρυσμένου ελέγχου και το πρωτεύων κλειδί του πίνακα. Ο τύπος int(11) μας προσφέρει έναν μεγάλο αριθμό μονάδων.

keypad_name: Το όνομα της μονάδας. Ο τύπος varchar(20) μας δίνει την δυνατότητα για ονόματα μέγιστου μήκους 20 χαρακτήρων.

type: Παίρνει μία τιμή 0-7 ανάλογα με την ενέργεια προς εκτέλεση, ύστερα από τη ταυτοποίηση του χρήστη. Συγκεκριμένα:

- 0 όταν έχουμε επιλέξει όπλιση και αφοπλιση κάποιου σεναρίου (ίδιου ή διαφορετικού), με προτεραιότητα στην εκτέλεση της αφοπλίσης.
- 1 όταν έχουμε επιλέξει όπλιση κάποιου σεναρίου και αφοπλιση όλων, με προτεραιότητα στην εκτέλεση της αφοπλίσης. Σε αυτήν την περίπτωση, αν υπάρχει κάποιο οπλισμένο σενάριο θα το αφοπλίζει, διαφορετικά θα οπλίζει το σενάριο που έχουμε επιλέξει.
- 2 όταν έχουμε επιλέξει όπλιση και αφοπλιση κάποιου σεναρίου (ίδιου ή διαφορετικού), με προτεραιότητα στην εκτέλεση της όπλίσης.
- 3 όταν έχουμε επιλέξει όπλιση κάποιου σεναρίου και αφοπλιση όλων, με προτεραιότητα στην εκτέλεση της όπλίσης. Σε αυτήν την περίπτωση, αν το σενάριο που έχουμε επιλέξει είναι αφοπλισμένο θα το οπλίζει, διαφορετικά θα αφοπλίζει όλα τα οπλισμένα σενάρια.
- 4 όταν έχουμε επιλέξει μόνο να οπλίζει κάποιο σενάριο.
- 5 όταν έχουμε επιλέξει μόνο να αφοπλίζει κάποιο σενάριο.
- 6 όταν έχουμε επιλέξει να αφοπλίζει όλα τα σενάρια.
- 7 όταν δεν έχουμε επιλέξει κάποια ενέργεια.

scenario_id_enable: Το αναγνωριστικό του σεναρίου προς όπλιση. Στην περίπτωση που δεν έχουμε επιλέξει κάποιο σενάριο προς όπλιση, παίρνει την τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη id του πίνακα scenario_list.

scenario_id_disable: Το αναγνωριστικό του σεναρίου προς αφοπλιση. Στην περίπτωση που δεν έχουμε επιλέξει κάποιο σενάριο προς αφοπλιση ή έχουμε επιλέξει την αφοπλιση όλων των σεναρίων, παίρνει την τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη id του πίνακα scenario_list.

inUse: Παίρνει τιμή 1 όταν η μονάδα απομακρυσμένου ελέγχου είναι ενεργοποιημένη, διαφορετικά παίρνει την τιμή 0. Στην περίπτωση που είναι απενεργοποιημένη, δεν θα εκτελεί καμία ενέργεια.

status: Λαμβάνει την τιμή 'on' όταν υπάρχει κάποιο οπλισμένο σενάριο, διαφορετικά έχει την τιμή 'off'.

4.2.14 Πίνακας `banned_channels`

Ο πίνακας `banned_channels` περιλαμβάνει όλες τις αποτυχημένες προσπάθειες εισόδου χρηστών, από κάθε κανάλι αυθεντικοποίησης. Κανάλι αυθεντικοποίησης είναι είτε η διεύθυνση IP του χρήστη, είτε κάποια ασύρματη μονάδα ελέγχου. Αν υπάρξουν πέντε αποτυχημένες προσπάθειες εισόδου, τότε απενεργοποιείται το συγκεκριμένο κανάλι για πέντε λεπτά και εκτελούνται οι ενέργειες που έχουν οριστεί για τον σκοπό αυτό. Αυτό συμβαίνει για την αποτροπή επιθέσεων brute-force [65] στο σύστημα. Όταν εισέλθει επιτυχώς ο χρήστης από το συγκεκριμένο κανάλι, τότε η εγγραφή του αφαιρείται από τον πίνακα. Στον πίνακα 20 παρουσιάζεται η δομή του πίνακα `banned_channels`.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
channel	varchar(45)	utf8_unicode_ci	όχι	Καμία	
type	tinyint(1)		όχι	Καμία	
time_banned	int(20)		όχι	Καμία	
count_attempts	int(1)		όχι	Καμία	

Πίνακας 20: Η δομή του πίνακα `banned_channels`

id: Ο αναγνωριστικός αριθμός των αποτυχημένων προσπαθειών εισόδου και το πρωτεύων κλειδί του πίνακα.

channel: Το κανάλι στο οποίο υπήρξε αποτυχημένη προσπάθεια εισόδου.

type: Παίρνει τιμή 0 όταν το κανάλι αυθεντικοποίησης είναι διεύθυνση ip, και 1 όταν είναι μονάδα απομακρυσμένου ελέγχου.

time_banned: Η χρονική στιγμή σε UNIX time [66] κατά την οποία επιβλήθηκε απαγόρευση εισόδου στο συγκεκριμένο κανάλι αυθεντικοποίησης, ύστερα από 5 αποτυχημένες προσπάθειες εισόδου. Αν δεν υπάρχει απαγόρευση εισόδου έχει τιμή 0. Επιλέχτηκε ο τύπος `int(20)` καθώς στο πεδίο αποθηκεύετε ένας μεγάλος αριθμός.

count_attempts: Ο αριθμός των αποτυχημένων προσπαθειών εισόδου από το συγκεκριμένο κανάλι αυθεντικοποίησης. Επιλέχτηκε ο τύπος `int(1)` καθώς λαμβάνει μία τιμή 1 έως 5.

4.2.15 Πίνακας bruteforce_actions

Ο πίνακας bruteforce_actions (πίνακας 21) περιλαμβάνει τις ενέργειες που θα εκτελεστούν στην περίπτωση που ανιχνευτεί επίθεση brute-force από κάποιο κανάλι αυθεντικοποίησης.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
action_delay	int(11)		όχι	Καμία	
output_type	tinyint(1)		όχι	Καμία	
output_device_id	int(11)		ναι	NULL	
output_group_id	int(11)		ναι	NULL	
script_id	int(11)		ναι	NULL	
action_time	int(11)		όχι	Καμία	

Πίνακας 21: Η δομή του πίνακα bruteforce_actions

id: Ο αναγνωριστικός αριθμός της συγκεκριμένης εγγραφής και το πρωτεύων κλειδί του πίνακα.

action_delay: Ο χρόνος σε δευτερόλεπτα για την ενεργοποίηση της συσκευής εξόδου από την στιγμή που θα επιβληθεί απαγόρευση εισόδου σε κάποιο κανάλι αυθεντικοποίησης.

output_type: Το είδος της συσκευής εξόδου. Παίρνει τιμή 0 όταν πρόκειται για απλή συσκευή εξόδου, 1 για ομάδα συσκευών εξόδου, 2 για εκτέλεση script και 3 για όλες τις συσκευές εξόδου.

output_device_id: Το αναγνωριστικό της συσκευής εξόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια ομάδα, script, ή όλες τις συσκευές, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη device_id του πίνακα devices.

output_group_id: Το αναγνωριστικό της ομάδας συσκευών εξόδου. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια συσκευή, script, ή όλες τις συσκευές, παίρνει τιμή NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη group_id του πίνακα groups.

script_id: Το αναγνωριστικό του script προς εκτέλεση. Στην περίπτωση που έχουμε επιλέξει ως συσκευή εξόδου κάποια συσκευή, ομάδα συσκευών, ή όλες τις συσκευές, παίρνει τιμή

NULL. Ο τύπος int(11) επιλέχτηκε με τα ίδια κριτήρια που επιλέχτηκε στην στήλη script_id του πίνακα scripts.

action_time: Ο χρόνος σε δευτερόλεπτα που θα παραμείνει ενεργή η συσκευή.

4.2.16 Πίνακας log

Ο πίνακας log (πίνακας 22) περιλαμβάνει το αρχείο καταγραφής των συμβάντων.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
date	date		όχι	Καμία	
time	time		όχι	Καμία	
event	tinyint(1)		όχι	Καμία	
scenario	varchar(20)	utf8_unicode_ci	όχι	Καμία	
id_scenario	int(11)		όχι	Καμία	
user	varchar(20)	utf8_unicode_ci	όχι	Καμία	
device	varchar(50)	utf8_unicode_ci	όχι	Καμία	

Πίνακας 22: Η δομή του πίνακα log

id: Ο αναγνωριστικός αριθμός της συγκεκριμένης εγγραφής και το πρωτεύων κλειδί του πίνακα.

date: Η ημερομηνία που συνέβη το συμβάν. Επιλέχτηκε ο τύπος date, και η αποθήκευση της γίνεται με την μορφή YY-MM-DD.

time: Η ώρα που συνέβη το συμβάν. Επιλέχτηκε ο τύπος time, και η αποθήκευση της γίνεται με την μορφή hh:mm:ss.

event: Παίρνει μία τιμή ανάλογα με το συμβάν στο οποίο αναφέρεται η εγγραφή. Συγκεκριμένα:

- 1 όταν ενεργοποιείται ο συναγερμός.
- 2 όταν οπλίζεται κάποιο σενάριο.
- 3 όταν αφοπλίζεται κάποιο σενάριο.
- 4 όταν ενεργοποιείται κάποια συσκευή εξόδου από τον χρήστη.
- 5 όταν απενεργοποιείται κάποια συσκευή εξόδου από τον χρήστη.
- 6 όταν εισάγεται λανθασμένος κωδικός.

- 7 όταν εισάγεται λανθασμένο RFID Tag.
- 8 όταν επιβάλλεται απαγόρευση εισόδου σε κάποιο κανάλι αυθεντικοποίησης.
- 9 όταν κάποιος χρήστης συνδέεται στο σύστημα.
- 0 όταν κάποιος χρήστης αποσυνδέεται από το σύστημα.

scenario: Το όνομα του σεναρίου στην περίπτωση όπλισης/αφόπλισης ή ενεργοποίησης του σεναρίου, ή της συσκευής εξόδου, όταν ο χρήστης ενεργοποιεί ή απενεργοποιεί τη συσκευή.

id_scenario: Ο αριθμός της εγγραφής του σεναρίου, όταν υπάρχει όπλιση/αφόπλιση ή ενεργοποίηση αυτού. Διαφορετικά έχει τιμή 0.

user: Το όνομα χρήστη, όταν συνδέεται/αποσυνδέεται στο σύστημα, οπλίζει/αφοπλίζει κάποιο σενάριο ή όταν ενεργοποιεί/απενεργοποιεί κάποια συσκευή εξόδου.

device: Λαμβάνει τιμή 'website', 'mobile' και 'webservice', όταν ο χρήστης οπλίζει/αφοπλίζει κάποιο σενάριο ή ενεργοποιεί/απενεργοποιεί κάποια συσκευή εξόδου από τον ιστοχώρο, από τον ιστοχώρο για κινητές συσκευές ή από το webservice, αντίστοιχα. Όταν συνδέεται/αποσυνδέεται κάποιος χρήστης στο σύστημα, ή εισάγεται λανθασμένος κωδικός ή RFID Tag λαμβάνει την διεύθυνση IP ή το όνομα της απομακρυσμένης μονάδας ελέγχου.

4.2.17 Πίνακας actions

Στον πίνακα actions (πίνακας 23) αποθηκεύεται η ενέργεια που ο χρήστης έχει επιλέξει, ώστε το back-end λογισμικό να την εκτελέσει.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(1)		όχι	Καμία	auto_increment
action	varchar(20)	utf8_unicode_ci	όχι	Καμία	
user	varchar(20)	utf8_unicode_ci	όχι	Καμία	
device	varchar(10)	utf8_unicode_ci	όχι	Καμία	

Πίνακας 23: Η δομή του πίνακα actions

id: Το πρωτεύων κλειδί του πίνακα. Έχει πάντα την τιμή 1, γι' αυτό επιλέχθηκε ο τύπος int(1).

action: Λαμβάνει τις εξής τιμές:

- scenario: Όταν ο χρήστης επιλέξει την όπλιση/αφόπλιση κάποιου σεναρίου.

- deviceOn: Όταν ο χρήστης επιλέξει την ενεργοποίηση/απενεργοποίηση κάποιας συσκευής εξόδου.
- allOff: Όταν ο χρήστης επιλέξει την αφόπλιση όλων των σεναρίων.
- gpriosetup: Όταν προστίθεται ή τροποποιείται κάποια συσκευή.
- bruteforce_actions: Όταν ανιχνευτεί επίθεση brute-force.
- restore: Όταν ενεργοποιείται το σύστημα.

user: Το όνομα του χρήστη που εκτελεί την ενέργεια. Όταν εκτελείται από τον χρονοπρογραμματισμό παίρνει την τιμή 'scheduled'.

device: Η συσκευή από την οποία ο χρήστης εκτελεί την ενέργεια. Λαμβάνεις τις εξής τιμές:

- Website: Όταν εκτελείται από την ιστοσελίδα.
- Mobile: Όταν εκτελείται από την ιστοσελίδα για κινητές συσκευές.
- Webservice: Όταν εκτελείται από το webservice.

4.2.18 Πίνακας schedule

Στον πίνακα schedule (πίνακας 24) αποθηκεύεται ο χρονοπρογραμματισμός των σεναρίων και των συσκευών εξόδου.

Όνομα	Τύπος	Σύνθεση	Κενό	Προεπιλογή	Πρόσθετα
id	int(11)		όχι	Καμία	auto_increment
date	varchar(10)	utf8_unicode_ci	όχι	Καμία	
starttime	varchar(8)	utf8_unicode_ci	όχι	Καμία	
stoptime	varchar(8)	utf8_unicode_ci	όχι	Καμία	
status	tinyint(1)		όχι	Καμία	
scenario_id	int(11)		ναι	NULL	
device_id	int(11)		ναι	NULL	
type	tinyint(1)		όχι	Καμία	

Πίνακας 24: Η δομή του πίνακα schedule

id: Ο αναγνωριστικός αριθμός της συγκεκριμένης εγγραφής και το πρωτεύων κλειδί του πίνακα.

date: Η ημέρα εκτέλεσης του χρονοπρογραμματισμού.

starttime: Η ώρα εκτέλεσης του χρονοπρογραμματισμού.

stoptime: Η ώρα τερματισμού του χρονοπρογραμματισμού.

status: Λαμβάνει μία τιμή 0-2. Συγκεκριμένα:

- 0, όταν ο χρονοπρογραμματισμός είναι απενεργοποιημένος.
- 1, όταν ο χρονοπρογραμματισμός είναι ενεργοποιημένος και το σενάριο είναι οπλισμένο.
- 2, όταν ο χρονοπρογραμματισμός είναι ενεργοποιημένος και το σενάριο είναι αφοπλισμένο

scenario_id: Ο αναγνωριστικός αριθμός του σεναρίου στο οποίο αναφέρεται η συγκεκριμένη εγγραφή. Στην περίπτωση που ο χρονοπρογραμματισμός είναι για κάποια συσκευή εξόδου, λαμβάνει τιμή NULL.

device_id: Ο αναγνωριστικός αριθμός της συσκευής εξόδου στην οποία αναφέρεται η συγκεκριμένη εγγραφή. Στην περίπτωση που ο χρονοπρογραμματισμός είναι για κάποια σενάριο, λαμβάνει τιμή NULL.

type: Το είδος του προγραμματισμού. Παίρνει τιμή 0 για χρονοπρογραμματισμό σεναρίου και 1 για χρονοπρογραμματισμό συσκευής εξόδου.

4.3 Λειτουργίες του ιστοχώρου

Στην ενότητα αυτή παρουσιάζονται όλες οι λειτουργίες που προσφέρει το σύστημα μέσω του ιστοχώρου, το οποίο αποτελεί διεπαφή χρήστη του συστήματος. Για οποιαδήποτε ενέργεια, θα πρέπει πρώτα ο πιστοποιημένης χρήστης να συνδεθεί εισάγοντας το όνομα χρήστη και τον κωδικό του.

Οι χρήστες κατηγοριοποιούνται σε απλούς χρήστες και διαχειριστές. Στους απλούς χρήστες προσφέρονται λειτουργίες για την χρήση του συστήματος, και στους διαχειριστές λειτουργίες για την διαχείριση του.

Αρχική
Διαχείριση
Σενάρια χρήσης
Χρονοπρογραμματισμός
Παρακολούθηση συσκευών
Κάμερα
Καταγραφή συμβάντων

12 Μαΐου 2016 16:33:54
χρήστης: admin [Αποσύνδεση]

Αρχική σελίδα

Σενάρια

σενάριο	παύση ▾	ON
scenario2		OFF
away		OFF
stayIn		OFF

Συσκευές εξόδου

led1	OFF
led2	OFF
led3	OFF
strobe	OFF
σειρήνα	OFF

Καταγραφή συμβάντων

σύνδεση χρήστη	admin	16:33:36 12-05-2016
αποσύνδεση χρήστη	admin	16:33:26 12-05-2016
λανθασμένο rfid	keypad1	16:32:46 12-05-2016
λανθασμένος κωδικός	192.168.1.65	16:28:22 12-05-2016
off	led3	16:27:25 12-05-2016
on	led3	16:27:07 12-05-2016
off	σενάριο	16:26:48 12-05-2016
alarm	σενάριο(6)	16:26:48 12-05-2016
alarm	σενάριο(3)	16:26:46 12-05-2016
on	σενάριο	16:26:40 12-05-2016

Εικόνα 64: Αρχική σελίδα

4.3.1 Λειτουργίες απλού χρήστη

Οι λειτουργίες του απλού χρήστη περιορίζονται στον χειρισμό των σεναρίων, των συσκευών εξόδου και του χρονοπρογραμματισμού αυτών, χωρίς την δυνατότητα τροποποίησής αυτών. Μπορούν δηλαδή να οπλίζουν και να αποπλίζουν σενάρια, να ενεργοποιούν τις συσκευές εξόδου και τον χρονοπρογραμματισμό, αλλά δεν έχουν την δυνατότητα επεξεργασίας ή διαγραφής αυτών. Επιπλέον, μπορούν να τροποποιήσουν τα στοιχεία του λογαριασμού τους, να παρακολουθούν ζωντανά την κατάσταση των συσκευών εισόδου, της κάμερας και να επισκέπτονται την καταγραφή συμβάντων.

➤ Χειρισμός σεναρίων

Ο χρήστης μπορεί να χειρίζεται τα σενάρια είτε από την αρχική σελίδα του συστήματος (εικόνα 64), είτε από την διαχείριση σεναρίων (εικόνα 65). Στην πρώτη περίπτωση, μπορεί να οπλίσει ή να αποπλίσσει όλο το σενάριο, ενώ στην δεύτερη περίπτωση, ο χρήστης μπορεί

επιπλέον να επιλέξει συγκεκριμένη εγγραφή του σεναρίου που επιθυμεί να οπλίσει ή να αφοπλίσει. Σε κάθε περίπτωση, ο χρήστης μπορεί να επιλέξει τον προσωρινό αφοπλισμό του σεναρίου ή κάποιας εγγραφής για 0.5, 1, 2 και 5 λεπτά.

Διαχείριση σεναρίων

12 Μαΐου 2016 14:56:33
χρήστης: user1 [Αποσύνδεση]

Επιλογή σεναρίου:

Όνομα σεναρίου:

A/A	Συσκευή εισόδου	Καθυστέρηση ενεργοποίησης (sec)	Χρόνος για δράση (sec)	Συσκευή εξόδου	Διάρκεια δράσης (sec)		
1	magneticTrap +	20	5	σειρήνα	60	☒	παύση
	pir +						
	beam -						
2	magneticTrap -	10	10	omada2	30	☒	παύση
	χρονικό παράθυρο (sec): 3						
3	omada1 +	5	0	sendMail	5	☐	
	magneticTrap +						
	pir2 -						
4	omada1 -	9	5	σειρήναΕξωτερ	4	☐	
	χρονικό παράθυρο (sec): 5						

Εικόνα 65: Διαχείριση σεναρίων

➤ Χειρισμός συσκευών εξόδου

Ο χρήστης μπορεί να ενεργοποιεί και να απενεργοποιεί τις συσκευές εξόδου από την κεντρική σελίδα του συστήματος. Οι συσκευές εξόδου (εικόνα 66) βρίσκονται κάτω από τα διαθέσιμα σενάρια.

Συσκευές εξόδου	
led1	☐ OFF
led2	☐ OFF
led3	☒ ON
strobe	☐ OFF
σειρήνα	☐ OFF
σειρήναΕξωτερStrobe	☐ OFF
σειρήναΕξωτερική	☐ OFF

Εικόνα 66: Χειρισμός συσκευών εξόδου

➤ Χειρισμός χρονοπρογραμματισμού

Αντίστοιχα με τον χειρισμό σεναρίων και συσκευών εξόδου, ο χρήστης μπορεί να ενεργοποιεί και να απενεργοποιεί τον χρονοπρογραμματισμό σεναρίων, συσκευών εξόδου ή ομάδων συσκευών εξόδου, χωρίς όμως να έχει την δυνατότητα επεξεργασίας των επιλογών του χρονοπρογραμματισμού.

Η σελίδα του χρονοπρογραμματισμού παρουσιάζεται στην παρακάτω εικόνα (εικόνα 67).

Χρονοπρογραμματισμός

12 Μαΐου 2016 15:14:01
χρήστης: user1 [Αποσύνδεση]

Ημέρα	Ώρα ενεργοποίησης	Ώρα απενεργοποίησης	Σενάριο/Συσκευή	
Τρίτη ▼	08 ▼ : 30 ▼	14 ▼ : 00 ▼	σενάριο ▼	☐ OFF
Δευτέρα ▼	12 ▼ : 05 ▼	13 ▼ : 45 ▼	led2 ▼	☐ OFF
Παρασκευή ▼	18 ▼ : 35 ▼	23 ▼ : 25 ▼	scenario2 ▼	☒ ON
Παρασκευή ▼	14 ▼ : 50 ▼	20 ▼ : 55 ▼	σενάριο ▼	☐ OFF

Εικόνα 67: Χρονοπρογραμματισμός

➤ Επεξεργασία χρήστη

Ο χρήστης μπορεί να επεξεργαστεί τα στοιχεία του λογαριασμού του (εικόνα 68) επιλέγοντας τον λογαριασμό του από την επεξεργασία χρηστών. Μπορεί να τροποποιήσει το όνομα χρήστη, τον κωδικό χρήστη, τον pin του, και να προσθέσει ή να τροποποιήσει κάποιο RFID Tag που διαθέτει. Εάν δεν γνωρίζει τον κωδικό κάποιου RFID Tag, μπορεί να το εισάγει μία φορά από τον αναγνώστη RFID κάποιας μονάδας απομακρυσμένου ελέγχου, και αυτός θα καταγραφεί στο αρχείο καταγραφής συμβάντων.

Επεξεργασία χρηστών

Όνομα χρήστη:	user1
Κωδικός χρήστη:	
Pin:	
Rfid:	65a171d5 1 ☒ +
Είδος χρήστη:	Απλός χρήστης
Αποθήκευση αλλαγών Άκυρο	

Εικόνα 68: Επεξεργασία χρήστη

➤ Παρακολούθηση συσκευών

Δίνεται η δυνατότητα στον χρήστη να παρακολουθεί ζωντανά την κατάσταση των συσκευών εισόδου (εικόνα 69). Μπορεί να παρακολουθεί είτε όλες τις συσκευές, είτε μία ομάδα συσκευών εισόδου, είτε μία μόνο συσκευή.

Παρακολούθηση συσκευών

Επιλογή συσκευής:

Όνομα συσκευής	Κατάσταση συσκευής
beam	on
diakoptis	off
magneticTrap	off
pir	on
pir2	on

Εικόνα 69: Παρακολούθηση συσκευών

➤ Κάμερα

Ο χρήστης μπορεί να παρακολουθεί με ζωντανή εικόνα τον χώρο στον οποίο είναι τοποθετημένη η Web Camera. Στην παρακάτω εικόνα παρουσιάζεται η σελίδα της κάμερας (εικόνα 70).

Κάμερα



Εικόνα 70: Κάμερα

➤ Καταγραφή συμβάντων

Ο χρήστης μπορεί να ενημερωθεί για τα τελευταία συμβάντα από την αρχική σελίδα ή, με περισσότερες λεπτομέρειες, από την καταγραφή συμβάντων (εικόνα 71). Στο αρχείο καταγράφονται οι συνδέσεις/αποσυνδέσεις των χρηστών, οι οπλίσεις/αφοπλίσεις και ενεργοποιήσεις των σεναρίων, οι ενεργοποιήσεις/απενεργοποιήσεις των συσκευών εξόδου από τον χρήστη, οι εισαγωγές λανθασμένων κωδικών και RFID Tag, και οι περιπτώσεις επιβολής απαγόρευσης εισόδου σε κάποιο κανάλι αυθεντικοποίησης.

Ο απλός χρήστης δεν έχει την δυνατότητα διαγραφής του αρχείου συμβάντων.

Καταγραφή συμβάντων

12 Μαΐου 2016 16:47:25
χρήστης: admin [Αποσύνδεση]

Γεγονός	Σενάριο/Συσκευή	Ωρα	Ημερομηνία	Χρήστης	Συσκευή
σύνδεση χρήστη		16:33:36	12-05-2016	admin	192.168.1.65
αποσύνδεση χρήστη		16:33:26	12-05-2016	admin	192.168.1.65
λανθασμένο rfid		16:32:46	12-05-2016	65a171d5	keypad1
λανθασμένος κωδικός		16:28:22	12-05-2016		192.168.1.65
off	led3	16:27:25	12-05-2016	admin	website
on	led3	16:27:07	12-05-2016	admin	website
off	σενάριο	16:26:48	12-05-2016	admin	website
alarm	σενάριο(6)	16:26:48	12-05-2016		
alarm	σενάριο(3)	16:26:46	12-05-2016		
on	σενάριο	16:26:40	12-05-2016	admin	website
alarm	σενάριο(6)	16:26:40	12-05-2016		
alarm	σενάριο(3)	16:26:40	12-05-2016		
σύνδεση χρήστη		15:37:57	12-05-2016	admin	192.168.1.65
αποσύνδεση χρήστη		15:37:49	12-05-2016	user1	192.168.1.65

Εικόνα 71: Καταγραφή συμβάντων

4.3.1 Λειτουργίες διαχειριστή

Ο διαχειριστής του συστήματος είναι ο αρμόδιος για την δημιουργία και την επεξεργασία των χρηστών, των συσκευών, των ομάδων συσκευών, των μονάδων απομακρυσμένου ελέγχου, των scripts, για την διαχείριση της ασφάλειας του συστήματος, των σεναρίων και του

χρονοπρογραμματισμού. Στον διαχειριστή επίσης παρέχονται οι λειτουργίες του απλού χρήστη.

➤ Δημιουργία και επεξεργασία χρηστών

Η λειτουργία αυτή παρέχει την δυνατότητα δημιουργίας και επεξεργασίας των χρηστών του συστήματος. Συγκεκριμένα, ο διαχειριστής μπορεί να δημιουργήσει νέους χρήστες (απλούς και διαχειριστές), να τροποποιήσει τα στοιχεία τους, να τους αναβαθμίσει ή να τους υποβαθμίσει σε διαχειριστή ή απλό χρήστη, και τέλος, να τους διαγράψει.

Η σελίδα της επεξεργασίας χρηστών παρουσιάζεται στην παρακάτω εικόνα (εικόνα 72).

Επεξεργασία χρηστών

Όνομα χρήστη	Είδος χρήστη	
admin	διαχειριστής	 
user1	απλός χρήστης	 

Εικόνα 72: Επεξεργασία χρηστών

➤ Δημιουργία και επεξεργασία συσκευών

Ο διαχειριστής είναι υπεύθυνος για την δημιουργία των συσκευών εισόδου και εξόδου. Στην εικόνα 73 φαίνεται η φόρμα στην οποία ο διαχειριστής εισάγει τα στοιχεία της συσκευής. Ο διαχειριστής επίσης, μπορεί να τροποποιήσει τα στοιχεία των συσκευών ή να τις διαγράψει.

Προσθήκη συσκευών

Όνομα συσκευής:	
Header: ?	
Αριθμός pin: ?	
Είδος συσκευής:	Συσκευή εισόδου ▼
Φυσιολογική κατάσταση:	low ▼
Προσθήκη Επαναφορά	

Εικόνα 73: Προσθήκη συσκευών

➤ Δημιουργία και επεξεργασία ομάδων συσκευών

Δίνεται η δυνατότητα στον διαχειριστή να δημιουργήσει ομάδες συσκευών εισόδου ή εξόδου (εικόνα 74), έτσι ώστε να είναι ευκολότερη η διαχείριση τους. Επιπλέον, μπορεί να τροποποιήσει το όνομα της ομάδας, να αλλάξει το είδος της (εισόδου ή εξόδου), να προσθέσει ή να αφαιρέσει συσκευές από αυτή, και να την διαγράψει.

Προσθήκη ομάδων

Όνομα ομάδας:	
Είδος ομάδας:	Συσκευών εισόδου ▼
Επιλογή συσκευών:	☐ pir ☐ diakoptis ☐ magneticTrap ☐ pir2 ☐ beam
Προσθήκη Επαναφορά	

Εικόνα 74: Προσθήκη ομάδων

➤ Δημιουργία και επεξεργασία μονάδων απομακρυσμένου ελέγχου

Ο διαχειριστής μπορεί να προσθέσει, να τροποποιήσει και να διαγράψει τις μονάδες απομακρυσμένου ελέγχου. Τα στοιχεία που αποθηκεύονται στο σύστημα για κάθε μονάδα είναι το όνομα της, ένα μοναδικό αναγνωριστικό (id) το οποίο εισάγεται και στον κώδικα της μονάδας απομακρυσμένου ελέγχου, έτσι ώστε η κεντρική μονάδα του συναγερμού να μπορεί να αναγνωρίσει την κάθε μονάδα απομακρυσμένου ελέγχου. Επίσης, η κατάσταση ορίζει αν η συγκεκριμένη μονάδα ελέγχου είναι ενεργοποιημένη ή απενεργοποιημένη, και τέλος στις ενέργειες δηλώνεται τι θα πραγματοποιεί η κεντρική μονάδα όταν κάποιος χρήστης εισάγει τον κωδικό pin ή το RFID Tag του.

Στην εικόνα 75 παρουσιάζεται η επεξεργασία μίας μονάδας απομακρυσμένου ελέγχου.

Επεξεργασία μονάδων απομακρυσμένου ελέγχου

Όνομα:	keypad1
Id:	5
Κατάσταση:	ενεργοποιημένο
Ενέργειες:	☒ ενεργοποίηση σενάριο ☒ απενεργοποίηση Όλων προτεραιότητα στην ενεργοποίηση
Αποθήκευση αλλαγών Άκυρο	

Εικόνα 75: Επεξεργασία μονάδων απομακρυσμένου ελέγχου

➤ Δημιουργία και επεξεργασία scripts

Η λειτουργία αυτή παρέχει την δυνατότητα δημιουργίας, επεξεργασίας και διαγραφής κώδικα σε shell script. Αυτά τα script συμπεριφέρονται σαν συσκευές εξόδου και εκτελούνται είτε από κάποιο σενάριο, είτε στην περίπτωση ανίχνευσης επίθεσης brute-force.

Η σελίδα της επεξεργασίας των shell script παρουσιάζεται στην παρακάτω εικόνα (εικόνα 76).

Διαχείριση scripts

Επιλογή script: makeFile

Όνομα script: makeFile

```
#!/bin/bash
# Για εκτέλεση από σενάριο/ενέργειες bruteforce:
# $1: Σενάριο που καλεί το script
# $2: Id του σεναρίου
# $3: Ημερομηνία
# $4: Ώρα
# Για εκτέλεση δράσης κατά την ενεργοποίηση/απενεργοποίηση σεναρίου:
# $1: Σενάριο που ενεργοποιείται/απενεργοποιείται
# $2: Id του σεναρίου
# $3: Ημερομηνία
# $4: Ώρα
# $5: Ενεργοποίηση/απενεργοποίηση

touch /home/scripts/testfile
echo "scenario: $1 id:$2 hmerominia: $3 ora: $4 $5" >> /home/scripts/testfile
```

αποθήκευση

Εικόνα 76: Διαχείριση scripts

➤ Διαχείριση ασφάλειας

Για την αποτροπή επιθέσεων brute-force το σύστημα, όταν ανιχνεύσει πέντε συνεχόμενες εισαγωγές λανθασμένου κωδικού ή μη καταχωρημένου RFID Tag, περιορίζει την πρόσβαση στην συγκεκριμένη διεύθυνση IP ή μονάδα απομακρυσμένου ελέγχου για πέντε λεπτά και εκτελεί τις δράσεις που έχουν οριστεί από τον διαχειριστή για τον σκοπό αυτό. Ο διαχειριστής επίσης μπορεί να άρει την απαγόρευση εισόδου για κάποιο κανάλι αυθεντικοποίησης και να μηδενίσει τον μετρητή.

Η σελίδα της διαχείρισης της ασφάλειας του συστήματος παρουσιάζεται στην εικόνα 77.

Συσκευή	Αποτυχημένες προσπάθειες εισόδου	Κατάσταση	
keypad1	2	ενεργό	✗
192.168.1.65	1	ενεργό	✗

Δράσεις στην περίπτωση επίθεσης

Καθυστέρηση δράσης (sec):	Συσκευή εξόδου:	Διάρκεια δράσης (sec):	
0	sendMail	5	⊖
0	strobe	20	⊖
0	σειρήνα	10	⊖

+

Εικόνα 77: Ασφάλεια

➤ Διαχείριση σεναρίων

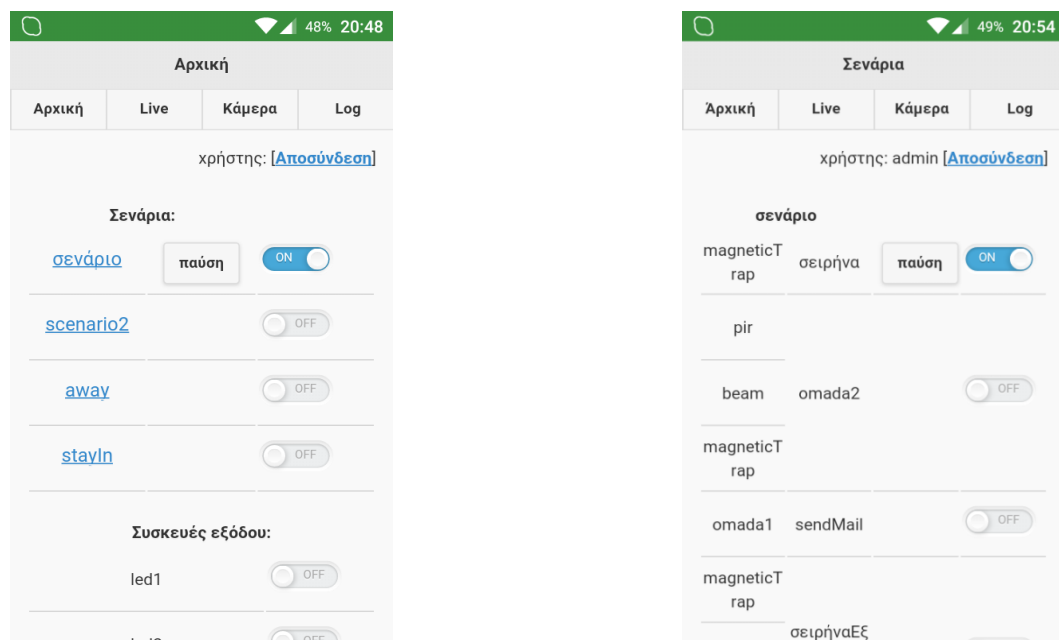
Η λειτουργία αυτή παρέχει την δυνατότητα δημιουργίας και επεξεργασίας των σεναρίων λειτουργίας του συστήματος. Κάθε σενάριο είναι ανεξάρτητο από τα υπόλοιπα και μπορεί να εκτελείται παράλληλα με κάποιο άλλο. Ο διαχειριστής ορίζει για κάθε συσκευή εισόδου, ομάδα συσκευών εισόδου, ή συνδυασμό συσκευών εισόδου με το λογικό “και” (μέσα σε χρονικό παράθυρο), τον χρόνο στον οποίο θα τεθεί σε λειτουργία από την στιγμή που θα οπλιστεί. Επίσης, ορίζει τον χρόνο που θα ενεργοποιηθεί η έξοδος από την στιγμή που θα ενεργοποιηθεί η είσοδος, την έξοδο, η οποία μπορεί να είναι μία συσκευή ή ομάδα συσκευών εξόδου ή εκτέλεση shell script, και την διάρκεια δράσης της εξόδου. Τέλος, μπορεί να ορίσει την ενεργοποίηση κάποιας συσκευής ή ομάδας συσκευών εξόδου ή την εκτέλεση κάποιου script κατά την όπλιση ή την αφόπλιση του σεναρίου.

➤ Διαχείριση χρονοπρογραμματισμού

Παρέχεται η δυνατότητα στον διαχειριστή να προγραμματίσει την όπλιση κάποιου σεναρίου ή την ενεργοποίηση κάποιας συσκευής εξόδου, επιλέγοντας την ημέρα και την ώρα ενεργοποίησης και απενεργοποίησης αυτών.

4.4 Λειτουργίες ιστοχώρου για κινητές συσκευές

Ο ιστοχώρος για κινητές συσκευές παρέχει στους χρήστες τις βασικές λειτουργίες για τον έλεγχο και χειρισμό του συναγερμού. Συγκεκριμένα ο χρήστης μπορεί να ενεργοποιήσει και να απενεργοποιήσει σενάρια και συσκευές εξόδου (εικόνα 78), να ελέγξει την κατάσταση των συσκευών εισόδου (εικόνα 79), να παρακολουθήσει τον χώρο μέσω της Web Camera και να ελέγξει το αρχείο καταγραφής συμβάντων (εικόνα 79).



Εικόνα 78: Αρχική σελίδα και διαχείριση σεναρίων από κινητή συσκευή

49% 20:56

Συσκευές

Αρχική

Live

Κάμερα

Log

χρήστης: admin [\[Αποσύνδεση\]](#)

Όνομα συσκευής

Κατάσταση συσκευής

beam

on

diakoptis

off

magneticTrap

off

pir

off

pir2

on

Δημητρίου Βησσαρίων

51% 20:59

Log

Αρχική

Live

Κάμερα

Log

χρήστης: admin [\[Αποσύνδεση\]](#)

Γεγονός

Ημερομηνία

σύνδεση
χρήστη

user1

14-05-2016
20:58:51

λανθασμένος
κωδικός

192.168.1.65

14-05-2016
20:58:24

λανθασμένος
κωδικός

192.168.1.65

14-05-2016
20:57:23

αποσύνδεση
χρήστη

admin

14-05-2016
20:56:37

off

σενάριο(1)

14-05-2016
20:55:07

on

σενάριο(1)

14-05-2016
20:54:34

14-05-2016

Εικόνα 79: Διαχείριση συσκευών εξόδου και καταγραφή συμβάντων από κινητή συσκευή

4.5 Λειτουργίες webservice

Δίνεται η δυνατότητα στον χρήστη να οπλίσει και να αφοπλίσει σενάρια μέσω ενός αρχείου webservice. Συγκεκριμένα, ο χρήστης επισκέπτοντας την σελίδα `webservice.php?user=username&pass=password&action=action_type&scenario=scenario_name`, ή την `webservice.php?pin=pin_num&action=action_type&scenario=scenario_name`, όπου `username`, `password` και `pin_num` το όνομα χρήστη, ο κωδικός, και ο `pin` του χρήστη, `action_type`: `enable` για όπλιση και `disable` για αφοπλίση, και `scenario_name` το όνομα του σεναρίου, μπορεί να χειρίζεται τα σενάρια. Το webservice απαντάει με “OK” αν είναι επιτυχημένη η ενέργεια ή με «FAIL» και HTTP error 500 [67] αν είναι αποτυχημένη.

4.6 Το λογισμικό της κεντρικής μονάδας

Στην ενότητα αυτή παρουσιάζεται το λογισμικό μέρος που αναπτύχθηκε για την κεντρική μονάδα. Το λογισμικό αυτό αναλαμβάνει να εκτελέσει όλες τις ενέργειες που ο χρήστης έχει ορίσει μέσω του ιστοχώρου του συστήματος.

4.6.1 Ανάλυση λειτουργίας της κεντρικής μονάδας

Η κεντρική μονάδα φιλοξενεί τον διακομιστή διαδικτύου μαζί με την βάση δεδομένων, τα οποία αποτελούν τη διεπαφή ελέγχου και προγραμματισμού. Παράλληλα, εκτελείται το back-end λογισμικό το οποίο είναι υπεύθυνο για όλες τις λειτουργίες του συστήματος.

Μόλις ο χρήστης συνδέσει την κεντρική μονάδα στην πρίζα ρεύματος η συσκευή επανέρχεται στην κατάσταση στην οποία ήταν πριν αποσυνδεθεί από την παροχή ρεύματος. Έτσι, αν ο συναγερμός ήταν οπλισμένος, θα εξακολουθεί να βρίσκεται οπλισμένος. Για να μπορέσει ο χρήστης να συνδεθεί με τον ιστοχώρο, ή να χρησιμοποιήσει κάποια συσκευή απομακρυσμένου ελέγχου, θα πρέπει να συνδέσει την κεντρική μονάδα με κάποιο τοπικό δίκτυο μέσω της θύρας Ethernet που διαθέτει.

Η κεντρική μονάδα διαθέτει επίσης ένα LED και ένα Beeper τα οποία ειδοποιούν τον χρήστη σχετικά με την κατάσταση του συναγερμού. Όταν κάποιο σενάριο οπλισθεί, τότε ανάβει το LED συνοδευόμενο από έναν διακοπτόμενο ήχο που παράγει το Beeper. Σε περίπτωση που αφοπλισθεί κάποιο σενάριο και δεν υπάρχει κάποιο άλλο οπλισμένο, το LED θα σβήσει και το Beeper θα παράξει ένα συνεχή ήχο. Τέλος, στην περίπτωση που αφοπλισθεί κάποιο σενάριο, ενώ υπάρχει κάποιο άλλο οπλισμένο, το LED θα παραμένει αναμμένο και ο χρήστης θα ενημερωθεί μόνο από το Beeper.

4.6.2 Παρουσίαση βιβλιοθηκών και συναρτήσεων

Για την ανάπτυξη του λογισμικού της κεντρικής μονάδας χρησιμοποιήθηκαν έξι έτοιμες βιβλιοθήκες οι οποίες αναλύονται στη συνέχεια.

MySQLdb

Η βιβλιοθήκη MySQLdb αποτελεί τη διεπαφή για τη σύνδεση με τη βάση δεδομένων MySQL. Στη συνέχεια παρουσιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν στην παρούσα διπλωματική:

- `MySQLdb.connect("localhost","username","password","database_name")`: Συνδέεται με τη βάση δεδομένων.
- `db.cursor()`: Δημιουργεί ένα αντικείμενο τύπου `cursor`.
- `cursor.execute(sql)`: Εκτελεί το SQL ερώτημα.
- `cursor.fetchall()`: Λαμβάνει όλες τις γραμμές από ένα ερώτημα προς τη βάση δεδομένων.
- `cursor.fetchone()`: Λαμβάνει μία γραμμή από ένα ερώτημα προς τη βάση δεδομένων.
- `db.commit()`: Αποθηκεύει τις αλλαγές στη βάση δεδομένων.

Adafruit_BBIO.GPIO

Η βιβλιοθήκη `Adafruit_BBIO.GPIO` αναλαμβάνει την επικοινωνία με τους ψηφιακούς ακροδέκτες του BeagleBone. Στη συνέχεια παρουσιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν στην παρούσα διπλωματική:

- `GPIO.setup("P8_10", GPIO.OUT)`: Ορίζει τον ακροδέκτη ως έξοδο.
- `GPIO.setup("P8_10", GPIO.IN)`: Ορίζει τον ακροδέκτη ως είσοδο.
- `GPIO.output("P8_10", GPIO.HIGH)`: Θέτει την τιμή του ακροδέκτη `high`.
- `GPIO.output("P8_10", GPIO.LOW)`: Θέτει την τιμή του ακροδέκτη `low`.
- `GPIO.input("P8_10")`: Λαμβάνει την τιμή του ακροδέκτη.

Multiprocessing

Η βιβλιοθήκη Multiprocessing αναλαμβάνει την παράλληλη εκτέλεση πολλών διεργασιών από το back-end λογισμικό. Στη συνέχεια παρουσιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν στην παρούσα διπλωματική:

- multiprocessing.Process(target=func, args=(arg1,)): Δημιουργεί ένα αντικείμενο τύπου process.
- Process.start(): Ξεκινά το process.

Os

Η βιβλιοθήκη os παρέχει πρόσβαση σε λειτουργίες του λειτουργικού συστήματος, όπως στον χειρισμό αρχείων, φακέλων και των process. Χρησιμοποιήθηκε η συνάρτηση os.system(arg) η οποία εκτελεί την εντολή που βρίσκεται ως όρισμα της στο τερματικό.

Time

Η βιβλιοθήκη Time παρέχει λειτουργίες που σχετίζονται με την μέτρηση του χρόνου. Χρησιμοποιήθηκε η συνάρτηση time.sleep(x) η οποία αναστέλλει την εκτέλεση του προγράμματος x δευτερόλεπτα.

Datetime

Η βιβλιοθήκη Datetime παρέχει κλάσεις και συναρτήσεις για τον χειρισμό ημερομηνιών και ωρών. Χρησιμοποιήθηκε η συνάρτηση datetime.now() η οποία επιστρέφει την τωρινή ημερομηνία και ώρα.

4.7 Το λογισμικό της μονάδας απομακρυσμένου ελέγχου

Η μονάδα απομακρυσμένου ελέγχου χρησιμοποιείται για την όπλιση και την αφόπλιση κάποιου σεναρίου από απόσταση, εισάγοντας τον κωδικό pin ή το RFID Tag του χρήστη. Η λειτουργία που θα εκτελέσει η συγκεκριμένη μονάδα προγραμματίζεται από τον χρήστη στην κεντρική μονάδα του συστήματος.

4.7.1 Ανάλυση λειτουργίας της μονάδας απομακρυσμένου ελέγχου

Αρχικά η μονάδα απομακρυσμένου ελέγχου επιχειρεί να συνδεθεί στο τοπικό δίκτυο μέσω του Wi-Fi module που διαθέτει. Στην περίπτωση που δεν επιτευχθεί σύνδεση με το Access Point μετά από τέσσερις συνεχόμενες προσπάθειες, γίνεται αυτόματη επανεκκίνηση, διαφορετικά η μονάδα απομακρυσμένου ελέγχου θα ξεκινήσει την επικοινωνία της με την κεντρική μονάδα. Με την κεντρική μονάδα επικοινωνεί κάθε 1,5 δευτερόλεπτο και λαμβάνει την κατάσταση των σεναρίων, ώστε να ενεργοποιήσει ή να απενεργοποιήσει το αντίστοιχο LED. Όπως και με την σύνδεση με το Access Point, αν δεν υπάρξει σύνδεση με την κεντρική μονάδα σε τέσσερις συνεχόμενες προσπάθειες, γίνεται αυτόματη επανεκκίνηση.

Παράλληλα με την επικοινωνία με την κεντρική μονάδα, η μονάδα απομακρυσμένου ελέγχου, αναμένει την εισαγωγή κωδικού pin ή RFID Tag. Στην περίπτωση εισαγωγής κωδικού pin στο πληκτρολόγιο, το σύστημα θα διακόψει την επικοινωνία με την κεντρική μονάδα μέχρι την εισαγωγή και των τεσσάρων ψηφίων του κωδικού, με μέγιστο χρονικό διάστημα μεταξύ του κάθε ψηφίου που εισάγει ο χρήστης να είναι τα τρία δευτερόλεπτα. Σε περίπτωση που ξεπεραστεί, θα διαγραφεί ότι έχει εισαχθεί και θα επανέλθει η επικοινωνία με την κεντρική μονάδα του συστήματος. Ο κωδικός pin ή ο αριθμός του RFID Tag αποστέλλονται στην κεντρική μονάδα για να γίνει η αυθεντικοποίηση του χρήστη και η εκτέλεση της ενέργειας που έχει προγραμματιστεί.

Η μονάδα απομακρυσμένου ελέγχου διαθέτει επίσης δύο LED, ένα πράσινο και ένα κόκκινο, και ένα Beeper για την οπτική και ηχητική ενημέρωση του χρήστη για την κατάσταση του συναγερμού. Το πράσινο LED είναι αναμμένο όταν υπάρχει επικοινωνία με την κεντρική

μονάδα, ενώ το κόκκινο όταν υπάρχει κάποιο οπλισμένο σενάριο. Το Beeper παράγει έναν στιγμιαίο ήχο κατά την εισαγωγή ψηφίων στον πληκτρολόγιο ή την εισαγωγή RFID Tag. Επίσης, παράγει έναν διακοπτόμενο ήχο όταν οπλίζεται κάποιο σενάριο από την συγκεκριμένη μονάδα ελέγχου, και έναν συνεχή ήχο όταν απενεργοποιείται κάποιο σενάριο από την συγκεκριμένη μονάδα ελέγχου.

4.7.2 Παρουσίαση βιβλιοθηκών και συναρτήσεων

Για την ανάπτυξη του λογισμικού της μονάδας απομακρυσμένου ελέγχου χρησιμοποιήθηκαν τέσσερις έτοιμες βιβλιοθήκες οι οποίες αναλύονται στη συνέχεια.

MFRC522.h

Πρόκειται για την βιβλιοθήκη που αναλαμβάνει την επικοινωνία με τον αναγνώστη RFID. Στη συνέχεια παρουσιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν στην παρούσα διπλωματική:

- MFRC522::PCD_Init(): αρχικοποιεί το MFRC522.
- MFRC522::PICC_IsNewCardPresent(): επιστρέφει true όταν ανιχνεύσει εισαγωγή νέου RFID Tag.
- MFRC522::PICC_ReadCardSerial(): επιστρέφει true όταν μπορεί να διαβαστεί το μοναδικό αναγνωριστικό (uid-unique identification) του RFID Tag. Το αναγνωσθέν UID αποθηκεύεται στην δομή uid.
- MFRC522::PICC_HaltA(): Οδηγεί το RFID Tag σε κατάσταση παύσης.

ESP8266.h

Πρόκειται για την βιβλιοθήκη που αναλαμβάνει την επικοινωνία με το Wi-Fi module. Στη συνέχεια παρουσιάζονται οι συναρτήσεις που χρησιμοποιήθηκαν στην παρούσα διπλωματική:

- ESP8266::getVersion(): επιστρέφει την έκδοση του λογισμικού του ESP8266.

- ESP8266::setOprToStation(): ορίζει τη λειτουργία του ESP8266 ως πελάτη (client).
- ESP8266::joinAP(String ssid, String pwd): συνδέεται με το Access Point.
- ESP8266::getLocalIP(): επιστρέφει την διεύθυνση IP.
- ESP8266::disableMUX(): οδηγεί το ESP8266 σε λειτουργία μιας σύνδεσης (TCP ή UDP).
- ESP8266::createTCP(String addr, uint32_t port): δημιουργεί μία TCP σύνδεση στην διεύθυνση IP και στην θύρα που υπάρχουν ως ορίσματα στην κλήση της συνάρτησης
- ESP8266::send(const uint8_t * buffer, uint32_t len): στέλνει δεδομένα στην TCP ή UDP σύνδεση που έχει δημιουργηθεί.
- ESP8266::recv(uint8_t * buffer, uint32_t buffer_size, uint32_t timeout = 1000): λαμβάνει δεδομένα από μια σύνδεση TCP ή UDP.

Keypad.h

Η βιβλιοθήκη keypad.h αναλαμβάνει τον χειρισμό του αριθμητικού πληκτρολογίου. Χρησιμοποιήθηκε η συνάρτηση Keypad::getKey() η οποία επιστρέφει το πλήκτρο που πατήθηκε.

SoftwareSerial.h

Η βιβλιοθήκη SoftwareSerial.h υποστηρίζει σειριακή επικοινωνία μέσω λογισμικού στους ψηφιακούς ακροδέκτες του Arduino, πέραν της ενσωματωμένης υποστήριξης σειριακής επικοινωνίας των ακροδεκτών 0 και 1. Η κλήση της SoftwareSerial serial1(rxPin, txPin) δημιουργεί ένα νέο αντικείμενο (serial1) στους ακροδέκτες που ορίζονται στην παρένθεση. Στην παρούσα διπλωματική χρησιμοποιήθηκε για το χειρισμό του Wi-Fi module.

SPI.h

Η συγκεκριμένη βιβλιοθήκη αναλαμβάνει την επικοινωνία με άλλες συσκευές που χρησιμοποιούν διεπαφή SPI. Σειριακή περιφερική διεπαφή (SPI) είναι ένα σύγχρονο σειριακό πρωτόκολλο που χρησιμοποιείται από μικροελεγκτές για την επικοινωνία με μία ή περισσότερες περιφερειακές συσκευές σε μικρές αποστάσεις. Μπορεί επίσης να

χρησιμοποιηθεί για την επικοινωνία μεταξύ δύο μικροελεγκτών. Στην παρούσα διπλωματική χρησιμοποιήθηκε για το χειρισμό του αναγνώστη RFID.

4.7.3 Επικοινωνία Arduino - BeagleBone

Η μονάδα απομακρυσμένου ελέγχου επικοινωνεί με την κεντρική μονάδα κάθε 1,5 δευτερόλεπτο προκειμένου να λάβει την κατάσταση του συναγερμού και να ενεργοποιήσει ή να απενεργοποιήσει το ανάλογο LED. Επίσης, όταν ο χρήστης εισάγει κωδικό pin ή RFID TAG, αποστέλλονται στην κεντρική μονάδα για την ταυτοποίησή του.

Σε κάθε μήνυμα προς την κεντρική μονάδα, η μονάδα απομακρυσμένου ελέγχου περιλαμβάνει τον αναγνωριστικό αριθμό του, καθώς και τον κωδικό pin ή τον αριθμό του RFID Tag αν έχουν εισαχθεί από τον χρήστη.

Η κεντρική μονάδα θα ελέγξει τον αναγνωριστικό αριθμό του Arduino και στην περίπτωση που έχει εισαχθεί κωδικός pin ή RFID Tag πιστοποιημένου χρήστη, τότε θα εκτελέσει την ενέργεια που είναι προγραμματισμένο. Σε κάθε περίπτωση, η κεντρική μονάδα θα επιστρέψει έναν αριθμό που υποδηλώνει την κατάσταση του συναγερμού, το οποίο θα το διαβάσει το Arduino και αναλόγως θα ενεργοποιήσει ή θα απενεργοποιήσει το LED και τον βομβητή.

4.7.4 Παράλληλη εκτέλεση εργασιών

Η μονάδα απομακρυσμένου ελέγχου αναμένει την εισαγωγή κωδικού pin ή RFID Tag από τον χρήστη, ενώ παράλληλα επικοινωνεί με την κεντρική μονάδα κάθε 1,5 δευτερόλεπτο για να ενημερωθεί για την κατάσταση των σεναρίων. Δυστυχώς, το Arduino Uno δεν παρέχει την δυνατότητα για παράλληλη εκτέλεση δύο νυμάτων, και για αυτόν τον λόγο υλοποιήθηκε ένα σύστημα ψευδοπρογραμματισμού, χρησιμοποιώντας τον εσωτερικό χρονομετρητή του μικροελεγκτή.

Μέσω της συνάρτησης `millis()`, η οποία επιστρέφει τον χρόνο που πέρασε από την εκκίνηση του Arduino, μπορούμε να μετρήσουμε τον χρόνο που πέρασε από την προηγούμενη επικοινωνία του Arduino με το BeagleBone. Έτσι, η συνάρτηση `loop` εκτελεί τον κώδικα που είναι υπεύθυνος για την εισαγωγή κωδικού `pin` και RFID Tag, ενώ μια δομή επιλογής `if` ελέγχει αν ο χρόνος αυτός έχει υπερβεί το 1,5 δευτερόλεπτο, για να καλέσει την συνάρτηση για την επικοινωνία με την κεντρική μονάδα.

4.8 Ασφάλεια συστήματος

Μεγάλη έμφαση δόθηκε στην προστασία του συστήματος από κακόβουλους χρήστες. Για αυτόν τον λόγο πάρθηκαν ορισμένα μέτρα τα οποία αποτρέπουν τέτοιες ενέργειες, τόσο στην κεντρική μονάδα και στην μονάδα απομακρυσμένου ελέγχου, όσο και στην μεταξύ τους επικοινωνία.

Η πρόσβαση στον ιστοχώρο, η χρήση του webservice και των μονάδων απομακρυσμένου ελέγχου γίνεται μόνο από πιστοποιημένους χρήστες. Διαφορετικά, ο χρήστης προωθείται στην σελίδα σύνδεσης, ή λαμβάνει σφάλμα (στο webservice και στις μονάδες απομακρυσμένου ελέγχου).

Όλα τα δεδομένα που εισάγει ο χρήστης φιλτράρονται και ελέγχονται από κανονικές εκφράσεις (regular expressions). Με αυτόν τον τρόπο, τα δεδομένα “καθαρίζονται” από ανεπιθύμητους χαρακτήρες που θα μπορούσαν να εκτελέσουν κώδικα ενός κακόβουλου χρήστη.

Σε κάθε σελίδα του ιστοχώρου υπάρχει έλεγχος των δικαιωμάτων του χρήστη. Έτσι, αν κάποιος απλός χρήστης επισκεφτεί μία σελίδα που προορίζεται για διαχειριστές, δεν θα του επιτραπεί η είσοδος, ενώ ένα μήνυμα θα τον ενημερώσει πως δεν έχει τα απαραίτητα δικαιώματα για να συνεχίσει.

Στην επικοινωνία μεταξύ της μονάδας απομακρυσμένου ελέγχου και της κεντρικής μονάδας χρησιμοποιείται ένας αναγνωριστικός αριθμός ο οποίος προσφέρει ένα είδος ταυτοποίησης. Στην περίπτωση που ο αριθμός αυτός δεν ταιριάζει με κάποιον που έχει εισαχθεί στην κεντρική μονάδα, η συσκευή αποκλείεται από το σύστημα και δεν εκτελεί καμία ενέργεια.

Τέλος, η απαγόρευση εισόδου για πέντε λεπτά στο κανάλι αυθεντικοποίησης που υπήρξαν πέντε συνεχόμενες ανεπιτυχείς προσπάθειες εισόδου, και η εκτέλεση των ανάλογων ενεργειών, συμβάλει σημαντικά στην ασφάλεια του συστήματος από επιθέσεις brute-force.

4.9 Ανάλυση κώδικα

Στη συνέχεια, παρουσιάζονται κάποια ενδεικτικά κομμάτια του κώδικα, τόσο του BeagleBone, όσο και του Arduino, και περιγράφεται η λειτουργία τους.

4.9.1 Ενεργοποίηση σεναρίου

Όταν ο χρήστης επιλέξει την όπλιση ενός σεναρίου, τότε για κάθε μία εγγραφή του σεναρίου θα δημιουργηθεί μία διεργασία, η οποία θα εκτελείται μέχρι την αφόπλιση του. Παρακάτω, παρουσιάζεται ο κώδικας που θα εκτελείται από τη διεργασία, όταν στην συγκεκριμένη εγγραφή δεν υπάρχει συνδυασμός συσκευών εισόδου με το λογικό 'και' (κώδικας 1).

```
1. def on_without_and_sensors(id,input_list,output_list,action_delay,action_time,in
   put_state_list,scenario,output_type,output_state_list,activation_delay):
2.     time.sleep(activation_delay)
3.
4.     while True:
5.         for i in range(0,len(input_list)):
6.             if (GPIO.input(input_list[i]) and input_state_list[i]==0) or (not GPI
   O.input(input_list[i]) and input_state_list[i]==1):
7.                 time.sleep(action_delay)
8.                 t=multiprocessing.Process(name=row[0],target=log,args=(scenario,
   str(id),str(1),"",),)
9.                 t.start()
10.
11.             if (output_type!=2):#device
12.                 timeout=time.time()+action_time
13.                 exit=0
14.                 while True:
15.                     for j in range(0,len(output_list)):
16.                         if (output_state_list[j]==0):
17.                             GPIO.output(str(output_list[j]), GPIO.HIGH)
18.                         else:
```

```

19.         GPIO.output(str(output_list[j]), GPIO.LOW)
20.         if time.time()>=timeout:
21.             for k in range(0,len(output_list)):
22.                 if(output_state_list[k]==0):
23.                     GPIO.output(str(output_list[k]), GPIO.LO
W)
24.             else:
25.                 GPIO.output(str(output_list[k]), GPIO.HI
GH)
26.
27.                 exit=1
28.                 break
29.                 time.sleep(0.1)
30.                 if(exit==1):
31.                     break
32.
33.                 elif(output_type==2):#script
34.                     t=subprocess.Popen(["/home/scripts/"+output_list[0],scenario
,str(id),str(datetime.now().strftime('%d-%m-
%Y')),str(datetime.now().strftime('%H:%M:%S'))])
35.                     time.sleep(action_time)
36.                     time.sleep(1)

```

Κώδικας 1: Ενεργοποίηση σεναρίου

Αρχικά, αναστέλλεται η εκτέλεση του για όσο χρόνο έχει οριστεί ως “καθυστέρηση ενεργοποίησης”. Στη συνέχεια, ένας ατέρμων βρόγχος εξετάζει αν έχει ενεργοποιηθεί κάποια συσκευή εισόδου (μία συσκευή, ομάδα, ή όλες οι συσκευές), λαμβάνοντας υπόψη την φυσιολογική κατάσταση της συσκευής όταν βρίσκεται σε αδράνεια.

Όταν υπάρξει ενεργοποίηση κάποιας συσκευής, τότε αναστέλλεται η εκτέλεση για χρόνο “καθυστέρησης δράσης”. Ακολούθως, το σύστημα καταγράφει το γεγονός στην καταγραφή συμβάντων, και εξετάζεται το είδος της εξόδου προς ενεργοποίηση. Εάν είναι συσκευή εξόδου (μία συσκευή, ομάδα, ή όλες οι συσκευές), τότε θα την ενεργοποιήσει και, όταν παρέλθει ο χρόνος που έχει οριστεί ως “διάρκεια δράσης”, θα την απενεργοποιήσει.

Τέλος, εάν ως έξοδος έχει οριστεί η εκτέλεση κάποιου shell script, τότε αυτό θα εκτελεστεί ως μία νέα διεργασία, ενώ θα διακοπεί η εκτέλεση του προγράμματος για το χρονικό διάστημα της “διάρκειας δράσης”.

4.9.2 Καταγραφή εικόνας

Το σύστημα λαμβάνει μία εικόνα, μέσω της κάμερας, κάθε δύο δευτερόλεπτα και την αποθηκεύει στην μνήμη RAM, ώστε να αποφεύγεται η συνεχής εγγραφή δεδομένων στην κάρτα μνήμης του BeagleBone. Ο χρήστης, επισκέπτοντας την σελίδα παρακολούθησης της κάμερας, λαμβάνει αυτήν την εικόνα, η οποία ανανεώνεται όταν υπάρχει νέα λήψη.

Στον κώδικα 2, παρουσιάζεται το shell script που υλοποιεί αυτή τη λειτουργία.

```
1. #!/bin/bash
2. while true
3. do
4.     fswebcam -r 320x176 --jpeg 85 -D 1 /var/www/ramdisc/image.jpeg
5.     sleep 2s
6. done
```

Κώδικας 2: Καταγραφή εικόνας

4.9.3 Συνάρτηση loop της μονάδας απομακρυσμένου ελέγχου

Όπως προαναφέρθηκε, η συνάρτηση loop εκτελείται στο Arduino συνεχώς, μέχρι να διακοπεί η τροφοδοσία του. Στη συνέχεια, παρουσιάζεται ο κώδικας της συνάρτησης loop που εκτελείται στη μονάδα απομακρυσμένου ελέγχου (κώδικας 3).

```
1. void loop(void)
2. {
3.
4.     keyPress = keypad.getKey();
5.     delay(100);
6.     if(keyPress){
7.         pin+=keyPress;
8.         previousMillis = millis();
9.         digitalWrite(3, HIGH);delay(100);digitalWrite(3, LOW);
10.        if(pin.length()==4){
11.            wireless_process();
12.        }
13.    }
14.
15.    notrfid=false;
16.    // Look for new cards
17.    if ( ! mfrc522.PICC_IsNewCardPresent() ) {
18.        notrfid=true;
```

```

19.  }
20.
21.  // Select one of the cards
22.  if ( ! mfrc522.PICC_ReadCardSerial() ) {
23.      notrfid=true;
24.  }
25.
26.  if(notrfid==false){ //insert rfid
27.      digitalWrite(3, HIGH);delay(100);digitalWrite(3, LOW);
28.
29.      for (byte i = 0; i < mfrc522.uid.size; i++) {
30.          rfid+=String(mfrc522.uid.uidByte[i], HEX);
31.      }
32.
33.      mfrc522.PICC_HaltA(); // Halt PICC
34.      wireless_process();
35.
36.  } //notrfid=false
37.
38.  if ((unsigned long)(millis() - previousMillis) >= interval && pin.length()==0
  || pin.length()==4) {
39.      previousMillis = millis();
40.      wireless_process();
41.  }
42.  else if (((unsigned long)(millis() - previousMillis) >= interval ) && ( (unsigned long)(millis() - previousMillis) >= interval2 )) {
43.      pin="";
44.      previousMillis = millis();
45.      wireless_process();
46.  }
47. } //loop

```

Κώδικας 3: Συνάρτηση loop του Arduino

Αρχικά, ελέγχεται εάν ο χρήστης έχει πατήσει κάποιο πλήκτρο, και εάν είναι το τέταρτο ψηφίο που εισάγεται, τότε εκτελείται η συνάρτηση `wireless_process()`, η οποία στέλνει τον κωδικό στο BeagleBone και λαμβάνει την ανάλογη απόκριση. Για κάθε ψηφίο που πληκτρολογείται, ο βομβητής παράγει έναν στιγμιαίο ήχο.

Στη συνέχεια, εξετάζεται εάν υπάρχει εισαγωγή κάποιου RFID Tag. Αντίστοιχα με προηγουμένως, θα παραχθεί ένας στιγμιαίος ήχος και θα εκτελεστεί η `wireless_process()` για να αποσταλεί ο αριθμός του RFID Tag στην κεντρική μονάδα.

Τέλος, το Arduino επικοινωνεί με την κεντρική μονάδα εάν έχει να επικοινωνήσει μαζί της πάνω από 1,5 δευτερόλεπτο, και διαγράφει όσα ψηφία έχει εισάγει ο χρήστης, εάν το προηγούμενο ψηφίο το είχε εισάγει περισσότερο από 3 δευτερόλεπτα νωρίτερα.

4.9.4 Σύνδεση του Arduino στο Access Point

Παρακάτω παρουσιάζεται ο κώδικας του Arduino, της συνάρτησης setup(), που είναι υπεύθυνο για την σύνδεση του Arduino με το Access Point μέσω του ESP8266 (κώδικας 4).

```
1.  if (wifi.setOprToStation()) {
2.    Serial.print("to station ok\r\n");
3.  }
4.  else {
5.    Serial.print("to station err\r\n");
6.  }
7.
8.  int i=0;
9.  while(i<maxretries && status == 0 ){
10.   if (wifi.joinAP(SSID, PASSWORD)) {
11.     Serial.print(F("Join AP success\r\n"));
12.     Serial.print("IP: ");
13.     Serial.println(wifi.getLocalIP().c_str());
14.     status=1;
15.   }
16.   else {
17.     i++;
18.     Serial.print(F("Join AP failure\r\n"));
19.     status=0;
20.   }
21. } //end while
22.
23. if ( i == maxretries ) { software_Reset(); }
24.
25.
26. if (wifi.disableMUX()) {
27.   Serial.print("single ok\r\n");
28. }
29. else {
30.   Serial.print("single err\r\n");
31. }
32.
33. Serial.print("setup end\r\n");
```

Κώδικας 4: Σύνδεση Arduino με Access Point

Η μονάδα απομακρυσμένου ελέγχου επιχειρεί να συνδεθεί με το Access Point. Εάν αυτό δεν επιτευχθεί για τέσσερις συνεχόμενες προσπάθειες, τότε γίνεται αυτόματη επανεκκίνηση του Arduino.

4.10 Σύνοψη μετρικών κώδικα

Σε αυτή την ενότητα αναφέρονται κάποια στοιχεία σχετικά με τον πυγαίο κώδικα που υλοποιήθηκε για την ανάπτυξη του λογισμικού. Πιο συγκεκριμένα, παρουσιάζεται πληροφορίες για τον κώδικα του Arduino, του back-end κώδικα του BeagleBone και του ιστοχώρου.

4.10.1 Μετρική κώδικα Arduino

Στον παρακάτω πίνακα παρουσιάζονται τα στοιχεία του κώδικα που αναπτύχθηκε για το Arduino (πίνακας 25).

Πλήθος αρχείων	1
Γραμμές κώδικα	260
Μέγεθος (bytes)	6614
Αριθμός συναρτήσεων	5
Αριθμός βιβλιοθηκών	5

Πίνακας 25: Μετρική κώδικα Arduino

4.10.2 Μετρική κώδικα back-end λογισμικού

Στον παρακάτω πίνακα παρουσιάζονται τα στοιχεία του κώδικα του back-end λογισμικού του BeagleBone που υλοποιήθηκε κυρίως στην γλώσσα Python (πίνακας 26).

Πλήθος αρχείων	3
Γραμμές κώδικα	935
Μέγεθος (bytes)	27657
Αριθμός συναρτήσεων	7
Αριθμός βιβλιοθηκών	6

Πίνακας 26: Μετρική κώδικα back-end λογισμικού

4.10.3 Μετρική κώδικα ιστοχώρου

Ο ιστοχώρος αποτελείται από 83 αρχεία, συνολικού μεγέθους 2046 KB. Αναλυτικότερα:

- 62 αρχεία PHP, συνολικού μεγέθους 349 KB.
- 5 αρχεία CSS, συνολικού μεγέθους 354 KB.
- 5 αρχεία JavaScript, συνολικού μεγέθους 553 KB.
- 11 αρχεία εικόνων, συνολικού μεγέθους 792 KB.

4.11 Ανάλυση SWOT

Το ακρόνυμο SWOT προκύπτει από τις αγγλικές λέξεις Strengths, Weaknesses, Opportunities, Threats (δυνατά σημεία, αδύνατα σημεία, ευκαιρίες, απειλές), και αποτελεί μία μέθοδο στρατηγικού σχεδιασμού που αξιολογεί αυτά τα τέσσερα στοιχεία ενός έργου ή ενός επιχειρηματικού εγχειρήματος [68].

Εσωτερικοί παράγοντες στο μοντέλο SWOT χαρακτηρίζονται τα δυνατά και τα αδύνατα σημεία του έργου, και εξωτερικοί παράγοντες, οι ευκαιρίες και οι απειλές.

Στη συνέχεια παρουσιάζεται το μοντέλο SWOT για την παρούσα εργασία (πίνακας 27).

Δυνατά σημεία Χαμηλό κόστος Καινοτόμες λειτουργίες	Αδύνατα σημεία Εμφάνιση ιστόχωρου Έλλειψη εφαρμογής για κινητά τηλέφωνα
Ευκαιρίες Βελτίωση λειτουργίας μονάδων απομακρυσμένου ελέγχου Δημιουργία αυτόνομου συστήματος μέσω 3g	Απειλές Μεγάλος ανταγωνισμός Ραγδαίες τεχνολογικές αλλαγές

Πίνακας 27: Μοντέλο SWOT

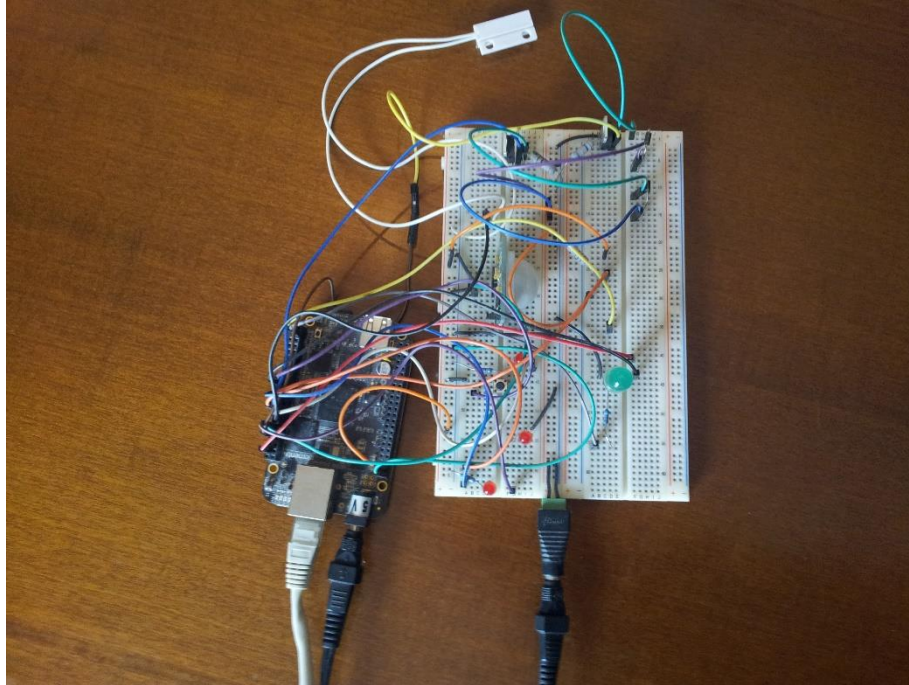
4.12 Δοκιμές

Μετά την ανάπτυξη του συστήματος, ακολούθησε μία περίοδος δοκιμαστικής λειτουργίας. Στην κεντρική μονάδα προστέθηκαν πέντε μαγνητικές επαφές και δύο αισθητήρες κίνησης PIR ως συσκευές εισόδου, ενώ οι συσκευές εξόδου που χρησιμοποιήθηκαν ήταν δύο σειρήνες, ένα strobe και τρία LED. Επίσης, χρησιμοποιήθηκε μία Web-Camera και μία μονάδα απομακρυσμένου ελέγχου, η οποία εγκαταστάθηκε σε άλλο δωμάτιο.

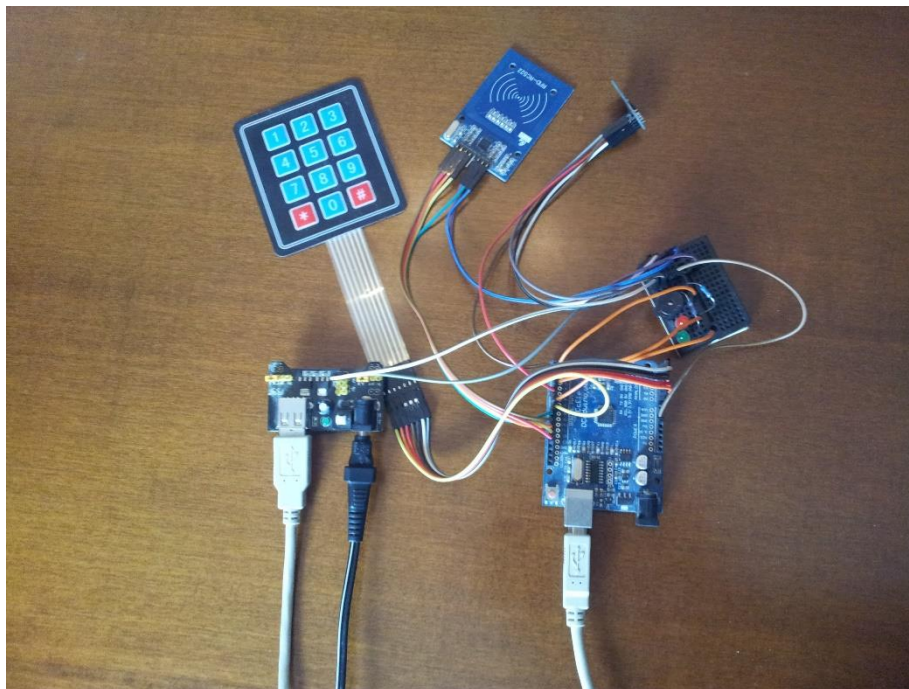
Το σύστημα δοκιμάστηκε για δέκα ημέρες συνεχής λειτουργίας όπου εξετάστηκαν τα εξής:

- Λειτουργία σεναρίων χρήσης: Διαπιστώθηκε η ομαλή λειτουργία των σεναρίων χρήσης. Το σύστημα ανταποκρινόταν σε όλες τις ενεργοποιήσεις των συσκευών εξόδου.
- Λειτουργία χρονοπρογραμματισμού: Ελέγχθηκε η ορθή λειτουργία του χρονοπρογραμματισμού, τόσο για τα σενάρια, όσο και για τις συσκευές εξόδου.
- Χειρισμός συσκευών εξόδου: Οι συσκευές εξόδου ενεργοποιούνταν ύστερα από εντολή του χρήστη.
- Παρακολούθηση συσκευών εισόδου και της Web-Camera: Εξετάστηκε η σωστή λειτουργία της παρακολούθησης της κατάστασης των συσκευών εισόδου και της Web-Camera.
- Καταγραφή συμβάντων: Το σύστημα κατέγραφε ορθά όλα τα γεγονότα.
- Εκτέλεση Scripts: Εξετάστηκε η σωστή εκτέλεση των shell scripts.
- Λειτουργία της μονάδας απομακρυσμένου ελέγχου: Διαπιστώθηκε η ομαλή λειτουργία της μονάδας απομακρυσμένου ελέγχου.
- Διαχείριση ασφαλιμάτων: Εκλέχθηκε η συμπεριφορά του συστήματος σε εισαγωγή μη ορθών δεδομένων από το χρήστη.
- Πρόσβαση σε μη εξουσιοδοτημένους χρήστες: Η πρόσβαση σε οποιαδήποτε σελίδα του ιστοχώρου δεν επιτρέπεται σε μη εξουσιοδοτημένους χρήστες.
- Προστασία από επιθέσεις brute-force: Το σύστημα, όταν ανίχνευε πέντε συνεχόμενες αποτυχημένες προσπάθειες εισόδου χρήστη, απαγόρευε την είσοδο για πέντε λεπτά από την συγκεκριμένη συσκευή και εκτελούσε τις ενέργειες που είχαν οριστεί για αυτή τη περίπτωση.

Στις παρακάτω εικόνες, παρουσιάζεται η τελική κατασκευή της κεντρικής μονάδας και της μονάδας απομακρυσμένου ελέγχου (εικόνα 80, εικόνα 81).



Εικόνα 80: Η κεντρική μονάδα



Εικόνα 81: Η μονάδα απομακρυσμένου ελέγχου

Κεφάλαιο 5 – Συμπεράσματα και μελλοντικές επεκτάσεις

Σκοπός της διπλωματικής εργασίας ήταν η κατασκευή μιας κεντρικής μονάδας συναγερμού ελεγχόμενη μέσω διαδικτύου, και μίας μονάδας απομακρυσμένου ελέγχου για τον χειρισμό της. Στη συνέχεια αναλύονται κάποια συμπεράσματα από την υλοποίηση του συστήματος, τα προβλήματα που παρουσιάστηκαν και πως αντιμετωπίστηκαν, και κάποιες μελλοντικές επεκτάσεις που θα πρόσδιδαν στο σύστημα νέα χαρακτηριστικά.

5.1 Σύνοψη και συμπεράσματα

Η κεντρική μονάδα συναγερμού που κατασκευάστηκε αποτελεί ένα ευέλικτο, σύγχρονο σύστημα συναγερμού το οποίο ανταγωνίζεται τα υπάρχοντα εμπορικά συστήματα πολλαπλάσιου κόστους. Ο χειρισμός του γίνεται είτε μέσω του διαδικτύου, είτε μέσω των μονάδων απομακρυσμένου ελέγχου.

Σε αντίθεση με άλλες κεντρικές μονάδες συναγερμού οι οποίες είναι προγραμματισμένες με δύο προγράμματα λειτουργίας, παραμονής (stay) και απών (away), ο χρήστης μπορεί να αποθηκεύσει τα δικά του σενάρια χρήσης και να ρυθμίσει διαφορετικούς χρόνους ενεργοποίησης, καθυστέρησης δράσης και διάρκεια δράσης για κάθε συσκευή εισόδου και εξόδου.

Επιπλέον, μπορούν να υπάρξουν πολλές καταστάσεις συναγερμού. Για παράδειγμα, ένας αισθητήρας κίνησης που βρίσκεται εκτός οικίας θα μπορούσε να ενεργοποιεί ένα strobe, ενώ η μαγνητική επαφή της εισόδου να ενεργοποιεί μία σειρήνα.

Άλλο ένα πλεονέκτημα του συστήματος είναι ο μεγάλος αριθμός συσκευών και μονάδων απομακρυσμένου ελέγχου που υποστηρίζει. Έτσι, μπορούν να εγκατασταθούν και συσκευές που δεν προορίζονται για την αντιδιαρρηκτική προστασία του χώρου, όπως

αισθητήρες πυρανίχνευσης ή διαρροής υγραερίου. Ο χρήστης μπορεί να παρακολουθεί μέσω διαδικτύου την κατάσταση των συσκευών εισόδου και να ενεργοποιεί τις συσκευές εξόδου.

Τέλος, λειτουργίες όπως ο χρονοπρογραμματισμός των σεναρίων και των συσκευών εξόδου, η εκτέλεση shell script, η προσωρινή παύση σεναρίων ή συγκεκριμένων συσκευών, η υποστήριξη κάμερας και η αυξημένη προστασία από επιθέσεις brute-force, καθιστούν την κεντρική μονάδα συναγερμού που κατασκευάστηκε καινοτόμα και πλήρως ανταγωνιστική των υπαρχόντων υλοποιήσεων.

5.2 Προβλήματα που ανέκυψαν κατά την υλοποίηση

Κατά την διάρκεια υλοποίησης της διπλωματικής εργασίας προέκυψαν κάποια προβλήματα, τόσο στο υλικό μέρος, όσο και στο λογισμικό, τα οποία έπρεπε να επιλυθούν ώστε το σύστημα να λειτουργεί ορθά και αποδοτικά.

Στο υλικό μέρος, υπήρξε μία δυσκολία στην χρήση όλων των διαθέσιμων ακροδεκτών του BeagleBone. Λόγω του ότι κάποιοι από τους ακροδέκτες χρησιμοποιούνται από το BeagleBone για την λειτουργία της μνήμης eMMC και του micro-HDMI, δεν μπορούσαν να λειτουργήσουν. Για αυτόν το λόγο, απενεργοποιήθηκαν αυτές οι λειτουργίες και προστέθηκε μία κάρτα microSD στην οποία εγκαταστάθηκε το λειτουργικό σύστημα.

Ένα άλλο ζήτημα που προέκυψε ήταν η λειτουργία του ESP8266. Αρχικά δοκιμάστηκε η λειτουργία του χρησιμοποιώντας εντολές AT [69], αλλά οι αποκρίσεις που λαμβάνονταν από το Arduino δεν έβγαζαν κάποιο νόημα. Ύστερα από πολλές δοκιμές, το πρόβλημα αντιμετωπίστηκε προσθέτοντας ένα εξωτερικό τροφοδοτικό 3,3 V αποκλειστικά για το ESP8266.

Στο λογισμικό μέρος, ένα πρόβλημα που έπρεπε να αντιμετωπιστεί ήταν αυτό του ελέγχου των επιλογών που κάνει ο χρήστης. Για παράδειγμα, έπρεπε να υπάρχει κάποιος μηχανισμός που να αποτρέπει την διαγραφή κάποιας συσκευής που χρησιμοποιείται από κάποιο σενάριο. Για αυτόν τον λόγο διαμορφώθηκε η βάση δεδομένων με τις κατάλληλες συσχετίσεις και προστέθηκαν οι απαραίτητοι έλεγχοι στον κώδικα.

Ένα τέταρτο ζήτημα που προέκυψε ήταν η ταυτόχρονη εκτέλεση δύο εργασιών από το Arduino. Για την αντιμετώπιση αυτού υλοποιήθηκε ένα σύστημα ψευδοπαράλληλισμού χρησιμοποιώντας τον εσωτερικό χρονομετρητή, όπως αναλύθηκε στο προηγούμενο κεφάλαιο.

5.3 Μελλοντικές επεκτάσεις

Σε αυτή την ενότητα παρουσιάζονται κάποιες επεκτάσεις που θα βελτιώναν το σύστημα και θα του πρόσθεταν νέα χαρακτηριστικά.

Μια χρήσιμη επέκταση που θα μπορούσε να υλοποιηθεί, η οποία θα διασφάλιζε την αδιάλειπτη λειτουργία του συστήματος, είναι η προσθήκη μπαταρίας ή συστοιχίας μπαταριών για την τροφοδοσία της κεντρικής μονάδας και των συσκευών, σε περίπτωση διακοπής ρεύματος. Επιπλέον, ο χρήστης θα μπορεί να προγραμματίσει κάποιες ενέργειες προς εκτέλεση, όταν υπάρξει διακοπή ρεύματος και όταν επανέλθει η τροφοδοσία.

Μία εξίσου σημαντική επέκταση είναι η χρήση ενός 3g modem, το οποίο θα αναλαμβάνει την επικοινωνία της κεντρικής μονάδας με το διαδίκτυο, σε περίπτωση διακοπής της σύνδεσης της κεντρικής μονάδας. Αντίστοιχα με την διακοπή της παροχής ρεύματος, ο χρήστης θα μπορεί να ορίσει κάποιες ενέργειες που θα εκτελούνται όταν η κεντρική μονάδα χάσει τη συνδεσιμότητα με το διαδίκτυο, και όταν την ανακτήσει.

Άλλη μία χρήσιμη βελτίωση που θα μπορούσε να γίνει σύστημα, είναι η χρήση του πρωτοκόλλου SSL [70] για την ασφαλή επικοινωνία του χρήστη με τον ιστοχώρο μέσω HTTPS [71]. Έτσι, τα δεδομένα που θα διακινούνται μεταξύ του χρήστη και του συναγερμού θα είναι κρυπτογραφημένα, αποτρέποντας κάποιον κακόβουλο χρήστη να τα υποκλέψει.

Τέλος, ο χειρισμός του συναγερμού μέσω μηνυμάτων SMS θα ήταν μια χρήσιμη βελτίωση, σε περίπτωση που ο χρήστης δεν διαθέτει σύνδεση με το διαδίκτυο.

Βιβλιογραφία

- [1] “Σχεδιασμός και κατασκευή ενσωματωμένου συστήματος συναγερμού στην αρχιτεκτονική Arduino με υποστήριξη δυναμικού ιστοχώρου διαχείρισης”, Μερτ Σοζούμπεκ, Μηνάς Δασυγένης, τμήμα μηχανικών πληροφορικής και τηλεπικοινωνιών, πανεπιστήμιο δυτικής Μακεδονίας, Ιούλιος 2012
- [2] A. W. Ahmad, N. Jan, S. Iqbal and C. Lee, "Implementation of ZigBee-GSM based home security monitoring and remote control system," *2011 IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Seoul, 2011, pp. 1-4
- [3] ZigBee Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/ZigBee>, Ιανουάριος 2016
- [4] Y. Xu and L. Shun, "The Design and Implementation of a guard against theft and alarm system with embedded telephone access and KEELOQ technique," *The 26th Chinese Control and Decision Conference (2014 CCDC)*, Changsha, 2014, pp. 4060-4065
- [5] KeeLoq Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/KeeLoq>, Ιανουάριος 2016
- [6] Z. y. Liu, Z. d. Wang, R. Chen and X. f. Wu, "Intelligent residential security alarm and remote control system based on single chip computer," *2008 3rd IEEE Conference on Industrial Electronics and Applications*, Singapore, 2008, pp. 159-161
- [7] AT89C51, <http://www.atmel.com/images/doc0265.pdf>, Ιανουάριος 2016
- [8] Web development Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Web_development, Ιανουάριος 2016
- [9] Client-side scripting Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Client-side_scripting, Ιανουάριος 2016
- [10] Server-side scripting Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Server-side_scripting, Ιανουάριος 2016
- [11] Database Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/Database>, Ιανουάριος 2016
- [12] HTML Βικιπαίδεια – Wikipedia, <https://el.wikipedia.org/wiki/HTML>, Ιανουάριος 2016
- [13] HTML Element Reference, <http://www.w3schools.com/tags/default.asp>, Ιανουάριος 2016
- [14] CSS Βικιπαίδεια – Wikipedia, <https://el.wikipedia.org/wiki/CSS>, Ιανουάριος 2016

- [15] Bootstrap Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Bootstrap_%28front-end_framework%29, Φεβρουάριος 2016
- [16] JavaScript Βικιπαίδεια – Wikipedia, <https://el.wikipedia.org/wiki/JavaScript>, Φεβρουάριος 2016
- [17] Ajax Βικιπαίδεια – Wikipedia, [https://en.wikipedia.org/wiki/Ajax_\(programming\)](https://en.wikipedia.org/wiki/Ajax_(programming)), Φεβρουάριος 2016
- [18] PHP Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/PHP>, Φεβρουάριος 2016
- [19] PHP 5 Global Variables – Superglobals, http://www.w3schools.com/php/php_superglobals.asp, Φεβρουάριος 2016
- [20] MySQL Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/MySQL>, Φεβρουάριος 2016
- [21] SQL Syntax, http://www.w3schools.com/sql/sql_syntax.asp, Φεβρουάριος 2016
- [22] jQuery Mobile Introduction, http://www.w3schools.com/jquerymobile/jquerymobile_intro.asp, Φεβρουάριος 2016
- [23] Arduino Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/Arduino>, Φεβρουάριος 2016
- [24] Arduino UNO, <https://www.arduino.cc/en/Main/ArduinoBoardUno>, Φεβρουάριος 2016
- [25] Arduino Ethernet, <http://www.arduino.cc/en/Main/ArduinoBoardEthernet>, Φεβρουάριος 2016
- [26] Arduino Leonardo, <http://www.arduino.cc/en/Main/ArduinoBoardLeonardo>, Φεβρουάριος 2016
- [27] Arduino Yun, <http://www.arduino.cc/en/Main/ArduinoBoardYun>, Φεβρουάριος 2016
- [28] Arduino Micro, <http://www.arduino.cc/en/Main/ArduinoBoardMicro>, Φεβρουάριος 2016
- [29] Arduino Mega Adk, <http://www.arduino.cc/en/Main/ArduinoBoardMegaADK>, Φεβρουάριος 2016
- [30] Arduino Due, <http://www.arduino.cc/en/Main/ArduinoBoardDue>, Φεβρουάριος 2016
- [31] Arduino Ethernet Shield, <http://www.arduino.cc/en/Main/ArduinoEthernetShield>, Φεβρουάριος 2016
- [32] Arduino Wi-Fi Shield, <http://www.arduino.cc/en/Main/ArduinoWiFiShield>, Φεβρουάριος 2016
- [33] Arduino GSM Shield, <http://www.arduino.cc/en/Main/ArduinoGSMShield>, Φεβρουάριος 2016
- [34] Arduino Motor Shield, <http://www.arduino.cc/en/Main/ArduinoMotorShieldR3>, Φεβρουάριος 2016

- [35] BeagleBoard Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/BeagleBoard>, Φεβρουάριος 2016
- [36] BeagleBoard, <http://beagleboard.org/beagleboard>, Φεβρουάριος 2016
- [37] BeagleBoard-xM, <http://beagleboard.org/beagleboard-xm>, Φεβρουάριος 2016
- [38] BeagleBone, <http://beagleboard.org/bone-original>, Φεβρουάριος 2016
- [39] BeagleBone Black, <http://beagleboard.org/black>, Φεβρουάριος 2016
- [40] LCD7 Cape, http://elinux.org/CircuitCo:BeagleBone_LCD7, Φεβρουάριος 2016
- [41] Tracking Cape, <http://elinux.org/CiudadOscura:TrackingCape>, Φεβρουάριος 2016
- [42] Battery Cape, http://elinux.org/CircuitCo:BeagleBone_Battery, Φεβρουάριος 2016
- [43] Notepad++, <https://notepad-plus-plus.org/>, Φεβρουάριος 2016
- [44] Fritzing, <http://fritzing.org/>, Φεβρουάριος 2016
- [45] Arduino IDE, <https://www.arduino.cc/en/Guide/Environment>, Φεβρουάριος 2016
- [46] PuTTY Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/PuTTY>, Φεβρουάριος 2016
- [47] Apache Server Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Apache_HTTP_Server, Φεβρουάριος 2016
- [48] PHPMyAdmin Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/PhpMyAdmin>, Φεβρουάριος 2016
- [49] Fsw webcam, <http://www.firestorm.cx/fswebcam/>, Φεβρουάριος 2016
- [50] Gimp, <https://www.gimp.org/>, Φεβρουάριος 2016
- [51] Passive infrared sensor Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Passive_infrared_sensor, Μάρτιος 2016
- [52] Electrical contacts Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Electrical_contacts, Μάρτιος 2016
- [53] Jumper Βικιπαίδεια – Wikipedia, [https://en.wikipedia.org/wiki/Jumper_\(computing\)](https://en.wikipedia.org/wiki/Jumper_(computing)), Μάρτιος 2016
- [54] MOSFET Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/MOSFET>, Μάρτιος 2016

- [55] BUK9575-100A, http://www.nxp.com/documents/data_sheet/BUK9575-100A.pdf, Μάρτιος 2016
- [56] 1N4007, http://www.diodes.com/_files/datasheets/ds28002.pdf, Μάρτιος 2016
- [57] Flyback diode Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Flyback_diode, Μάρτιος 2016
- [58] Strobe light Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Strobe_light, Μάρτιος 2016
- [59] Buzzer Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/Buzzer>, Μάρτιος 2016
- [60] Light-emitting diode Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Light-emitting_diode, Μάρτιος 2016
- [61] Radio-frequency identification Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Radio-frequency_identification, Μάρτιος 2016
- [62] MFRC522, http://www.nxp.com/documents/data_sheet/MFRC522.pdf, Μάρτιος 2016
- [63] ESP8266, https://cdn-shop.adafruit.com/datasheets/ESP8266_Specifications_English.pdf, Μάρτιος 2016
- [64] SoftwareSerial, <https://www.arduino.cc/en/Reference/SoftwareSerial>, Μάρτιος 2016
- [65] Brute-force attack Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Brute-force_attack, Μάρτιος 2016
- [66] Unix time Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Unix_time, Μάρτιος 2016
- [67] HTTP Status Codes, <https://www.w3.org/Protocols/HTTP/HTRESP.html>, Μάρτιος 2016
- [68] SWOT analysis Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/SWOT_analysis, Μάρτιος 2016
- [69] ESP8266 AT Command Set, http://www.pridopia.co.uk/pi-doc/ESP8266AT_CommandsSet.pdf, Μάρτιος 2016
- [70] Transport layer security Βικιπαίδεια – Wikipedia, https://en.wikipedia.org/wiki/Transport_Layer_Security, Μάρτιος 2016
- [71] HTTPS Βικιπαίδεια – Wikipedia, <https://en.wikipedia.org/wiki/HTTPS>, Μάρτιος 2016