



Δημοκρίτειο Πανεπιστήμιο Θράκης
Πολυτεχνική Σχολή
Τμήμα Μηχανικών Παραγωγής και Διοίκησης
Τομέας Συστημάτων Παραγωγής

Επιβλέπων: Αναπληρωτής Καθηγητής Κατσαβούνης Στέφανος

**Διαδικτυακή Εφαρμογή Παρακολούθησης & Διαχείρισης
Έξυπνων Συσκευών:
Σχεδιασμός & Υλοποίηση για Έξυπνα Σπίτια**

Διπλωματική Εργασία

Μησσήν Στέφανος

Αρ. Μητρώου: 91432



Ξάνθη, Οκτώβριος 2018



Δημοκρίτειο Πανεπιστήμιο Θράκης
Πολυτεχνική Σχολή
Τμήμα Μηχανικών Παραγωγής και Διοίκησης
Τομέας Συστημάτων Παραγωγής

Επιβλέπων: Αναπληρωτής Καθηγητής Κατσαβούνης Στέφανος

Διαδικτυακή Εφαρμογή Παρακολούθησης & Διαχείρισης
Έξυπνων Συσκευών:
Σχεδιασμός & Υλοποίηση για Έξυπνα Σπίτια

Διπλωματική Εργασία

Μησσήν Στέφανος

Αρ. Μητρώου: 91432

Αφιερωμένο στους γονείς μου,

Kíμωνα και Μαρία.

Πρόλογος

Οι έννοιες των έξυπνων συσκευών (Smart Devices), των έξυπνων σπιτιών (Smart Homes) αλλά και των έξυπνων συμπλεγμάτων (Smart Clusters), συνδέονται με την έννοια των Πραγμάτων του Διαδικτύου (Internet of Things – IoT). Όλες οι παραπάνω έννοιες αποτελούν επίκεντρο της τεχνολογικής έρευνας. Έτσι, για τη μετατροπή ενός συμβατικού σπιτιού σε έξυπνο, κρίνεται αναγκαία η δημιουργία ενός συστήματος επικοινωνίας, μεταξύ των έξυπνων συσκευών που είναι εγκατεστημένες στο σπίτι, αλλά και του ανθρώπου, έτσι ώστε ο δεύτερος, να διαχειρίζεται τις λειτουργίες που προσφέρουν οι έξυπνες συσκευές ανάλογα με τις απαιτήσεις του.

Η παρούσα διπλωματική εργασία μελετά τη δημιουργία ενός συστήματος παρακολούθησης & διαχείρισης έξυπνων συσκευών με σκοπό τη μετατροπή ενός συμβατικού σπιτιού σε έξυπνο. Πιο συγκεκριμένα, εξετάζονται οι εσωτερικές & εξωτερικές θερμοκρασίες του χώρου μέσω αισθητήριων οργάνων, γίνεται η αποθήκευση των μετρήσεων σε κατάλληλες βάσεις δεδομένων και έτσι ο χρήστης μπορεί να κρίνει πότε είναι αναγκαία η ενεργοποίηση ή η απενεργοποίηση των έξυπνων συσκευών εντός του χώρου του σπιτιού (στην παρούσα διπλωματική έχουν χρησιμοποιηθεί ως έξυπνες συσκευές, το κλιματιστικό και ο αφυγραντήρας). Η διαχείριση των έξυπνων συσκευών γίνεται μέσω από τη διαδικτυακή εφαρμογή (Web Application) Smart Homie που δημιουργήθηκε για τις ανάγκες της παρούσας διπλωματικής εργασίας. Επιπλέον, εξετάζονται τα πρωτόκολλα αποστολής δεδομένων από τα αισθητήρια όργανα προς τις βάσεις δεδομένων, που είναι η ασύρματη μεταφορά δεδομένων (Wi-Fi) καθώς και η ενσύρματη μεταφορά δεδομένων (Ethernet), οι τρόποι αποφυγής απωλειών δεδομένων, καθώς και η εύρυθμη επικοινωνία των έξυπνων συσκευών με τη διαδικτυακή εφαρμογή.

Το πρόβλημα προσεγγίστηκε από διαφορετικές διαστάσεις. Αρχικά διαπιστώθηκε το κατά πόσο είναι εφικτό, η δημιουργία μίας γέφυρας επικοινωνίας μεταξύ των έξυπνων συσκευών και του ανθρώπου. Έπειτα αναπτύχθηκε ο κατάλληλος αλγόριθμος για τη συλλογή πρωτογενών δεδομένων θερμοκρασίας και υγρασίας εντός και εκτός του σπιτιού που χρησιμοποιήθηκε για την παρούσα εργασία, και τέλος αναπτύχθηκε μία διαδικτυακή εφαρμογή με σκοπό τη διεπαφή του χρήστη με τις έξυπνες συσκευές αλλά και τη διαχείρισή τους.

Ακολούθησε μια εκτενής ανάλυση των αποτελεσμάτων. Διαπιστώθηκαν οι περιορισμοί και οι υποθέσεις της εργασίας που αφορούν το κατά πόσο εύχρηστη θα μπορεί να είναι μία τέτοιου είδους διαδικτυακή εφαρμογή για τον άνθρωπο, καθώς και οι τρόποι βελτίωσής της.

Αναλυτικότερα στο 1ο κεφάλαιο παρουσιάζεται το πρόβλημα, η σημασία του για τις σύγχρονες κοινωνίες, οι στόχοι της διπλωματικής εργασίας καθώς και η δομή αυτής.

Στο 2ο κεφάλαιο εξετάζεται το θεωρητικό υπόβαθρο, αναλύονται οι τεχνολογίες καθώς και ο εξοπλισμός που χρησιμοποιήθηκε για την υλοποίηση της διαδικτυακής εφαρμογής.

Στο 3^ο κεφάλαιο αναλύεται το υλικό μέρος, η συνδεσμολογία του συστήματος για την άντληση δεδομένων και τέλος, η σύνδεση αυτών με τον τοπικό διακομιστή και τις βάσεις δεδομένων.

Στο 4^ο κεφάλαιο επεξηγείται λεπτομερώς η σχεδίαση και η υλοποίηση της διαδικτυακής εφαρμογής. Πιο συγκεκριμένα, αναλύονται οι απαιτήσεις του συστήματος, προσεγγίζεται το θέμα της ασφάλειας των δεδομένων του χρήστη και πώς προστατεύονται τα παραπάνω. Ακόμη, αναλύονται οι βάσεις δεδομένων και οι λειτουργίες της εφαρμογής.

Στο 5ο κεφάλαιο συνοψίζονται τα συμπεράσματα και τα προβλήματα που αντιμετωπίστηκαν κατά την υλοποίηση της διπλωματικής εργασίας και αναλύονται περαιτέρω τρόποι βελτίωσης της διαδικτυακής εφαρμογής.

Ευχαριστίες

Στα πλαίσια της παρούσας διπλωματικής εργασίας, αισθάνομαι την υποχρέωση να ευχαριστήσω ορισμένους ανθρώπους που γνώρισα, χωρίς αυτούς θα ήταν πολύ δυσκολότερος και δύσβατος ο δρόμος που έχω διανύσει για να φτάσω ως εδώ. Συνέβαλαν όλοι με τον τρόπο τους στη διαμόρφωση του χαρακτήρα και της νοοτροπίας μου. Με βοήθησαν να πετύχω πολλά από τα όνειρά μου αλλά και να δημιουργήσω νέα, βάζοντας έτσι στόχο, στο προσεχές μέλλον να κατορθώσω και να εκπληρώσω όλες τις φιλοδοξίες μου.

Αρχικά οφείλω να ευχαριστήσω τον κ. Κατσαβούνη Στέφανο, Αναπληρωτή Καθηγητή του Τμήματος Μηχανικών Παραγωγής και Διοίκησης της Πολυτεχνικής Σχολής του Δημοκριτείου Πανεπιστημίου Θράκης, επιβλέποντα της παρούσας διπλωματικής εργασίας μου. Η καθοδήγηση, οι προτάσεις για βελτίωση καθώς και η συνεχής διαθεσιμότητά του σε ότι χρειαζόμουν, με βοήθησε να ξεπεράσω τις δυσκολίες που αντιμετώπισα στη διπλωματική μου εργασία και να κάνω τις ιδέες μου πραγματικότητα. Από την πρώτη κιόλας μέρα της γνωριμίας μας, κατάφερε να μου κεντρίσει το ενδιαφέρον, μεταδίδοντας μου έτσι μέρος των γνώσεων & και των εμπειριών του όσον αφορά επιστημονικούς κλάδους, που ανταποκρίνονται στα ενδιαφέροντά μου.

Τους Καθηγητές του Τμήματος Μηχανικών Παραγωγής και Διοίκησης, που με την πολυετή πείρα τους κατάφεραν να μεταδώσουν με επιτυχία πολύτιμες γνώσεις και εξελίξεις της επιστήμης μας, χρήσιμες για την περαιτέρω επαγγελματική μου σταδιοδρομία. Τους φίλους και συναδέλφους μου, για την υποστήριξη και τη συνέργεια μας τα τελευταία αντά 5 χρόνια. Τους ευχαριστώ θερμά για τον ευχάριστο χρόνο, τις χαρές αλλά και τις λύπες που περάσαμε μαζί, και πάνω από όλα που καταφέραμε να είμαστε φίλοι και συνοδοιπόροι όλο αυτό τον καιρό χωρίς διαλλείματα και παρεξηγήσεις.

Τέλος, έχω αφήσει τους σημαντικότερους ανθρώπους όλης μου της ζωής, την οικογένεια μου, χωρίς τους οποίους δεν θα έγραφα αυτές τις σελίδες της παρούσας διπλωματικής εργασίας. Τους ευχαριστώ με όλη μου την καρδιά για την απεριόριστη υποστήριξη, δύναμη αλλά και επιμονή που μου ενέπνευσαν ώστε να φτάσω μέχρι εδώ.

Τον πατέρα μου και την μητέρα μου, Κίμωνα και Μαρία, που με στήριξαν σε κάθε ξεχωριστό βήμα της ζωής μου, που μου έδωσαν την οικονομική αλλά και ψυχολογική υποστήριξη να σπουδάσω και πάνω από όλα στην αγάπη που μου έχουν δείξει καθ' όλη την διάρκεια της ζωής μου.

Έπειτα τον παππούδες μου, Νικόλαο & Ωριγένη καθώς και τις γιαγιάδες μου, Μαρία & Άννα, με τους οποίους αν και δεν έχω περάσει αρκετές στιγμές, έχουν χαράξει στη μνήμη μου ο καθένας ξεχωριστά τη δική του εικόνα & προσωπικότητα.

Μησσήν Στέφανος,

Οκτώβριος 2018

Πίνακας Περιεχομένων

Πρόλογος	i
Ευχαριστίες	iii
Πίνακας περιεχομένων	v
Περιεχόμενα Πινάκων και Σχημάτων	v
Περίληψη	ix
Abstract	xi
1 Εισαγωγή	1
1.1 Το πρόβλημα και η σημασία του	1
1.2 Internet of Things (IoT)	3
1.3 Στόχοι της διπλωματικής εργασίας	6
1.4 Δομή της διπλωματικής εργασίας	7
2 Θεωρητικό υπόβαθρο εννοιών και εξοπλισμού	9
2.1 Γλώσσες Προγραμματισμού Διαδικτυακής Εφαρμογής	9
2.1.1. Γλώσσα Σήμανσης Υπερκειμένου - HTML	10
2.1.2. Γλώσσα Μορφοποίησης – CASCADING STYLES SHEETS (CSS)	15
2.1.3. Γλώσσα Επικοινωνίας Χρήστη – Διακομιστή PHP	18
2.1.4. Γλώσσα Scripting - JavaScript	22
2.1.5. Γλώσσα Σχεσιακών Βάσεων Δεδομένων MySQL	24
2.2 Υλικός Εξοπλισμός - Hardware	26
2.2.1. Μικροελεγκτής Arduino uno Rev3	26

2.2.2. Μικροελεγκτής WeMos D1 R2	32
2.2.3. Δρομολογητής - Router.....	36
2.3 Πλατφόρμες Ανάπτυξης Λογισμικού.....	37
2.3.1. Visual Studio Code	38
2.3.2. Brackets	40
2.3.3. XAMPP.....	41
2.3.4. Πλατφόρμα ανάπτυξης λογισμικού Arduino IDE	42
3 Υλικό μέρος και συνδεσμολογία συστήματος	49
3.1 Hardware Υλοποίησης Συστήματος.....	49
3.1.1. Arduino Uno Rev3 – Arduino Ethernet Shield.....	49
3.1.2. WeMos D1 R2	51
3.1.3. Πλακέτα δοκιμών (Breadboard 400)	52
3.1.4. Ακρυλική βάση τοποθέτησης μικροελεγκτή – πλακέτας δοκιμών	53
3.1.5. Αισθητήριο θερμοκρασίας & υγρασίας DHT22/AM2302	54
3.1.6. Καλώδια Jumper	58
3.1.7. Καλώδια σύνδεσης πλακετών με H/Y	59
3.2 Συνδεσμολογία Συστήματος (Hardware Wiring).....	60
3.3 Σύνδεση υλικού με τοπικό εξυπηρετητή.....	66
3.3.1. Προγραμματισμός αισθητηρίων για λήψη δεδομένων	66
3.3.2. Διάγραμμα ροής λειτουργίας αισθητήρα DHT22/Am2302	71
3.4 Υπολογισμός κόστους εξαρτημάτων	73
4 Σχεδίαση και υλοποίηση διαδικτυακής εφαρμογής	77
4.1 Απαιτήσεις Συστήματος.....	75
4.2 Διαδικτυακή ασφάλεια & Προστασία προσωπικών δεδομένων	77
4.3 Δημιουργία και περιγραφή των βάσεων δεδομένων	78
4.3.1. Εργαλείο δημιουργίας βάσεων δεδομένων phpMyAdmin	78
4.3.2. Αρχιτεκτονική βάσης δεδομένων διαδικτυακής εφαρμογής.....	80
4.4 Περιγραφή λειτουργίας διαδικτυακής εφαρμογής Smart Homie.....	90
4.4.1. Λειτουργία εγγραφής και λειτουργία σύνδεσης (Register & Log In).....	90
4.4.2. Λειτουργία αρχικής καρτέλας Dashboard.....	93
4.4.3. Λειτουργία καρτέλας Live Monitoring	94
4.4.4. Λειτουργία καρτέλας Analytics	96
4.4.5. Λειτουργία καρτέλας Profile.....	99
4.4.6. Λειτουργία καρτέλας Users	101
4.4.7. Λειτουργία καρτέλας Register User.....	102

4.4.8. Λειτουργία λοιπών σημείων διαδικτυακής εφαρμογής	103
5 Συμπεράσματα και περαιτέρω έρευνα	111
5.1 Σύνοψη και συμπεράσματα.....	108
5.2 Προβλήματα που προέκυψαν κατά την υλοποίηση του συστήματος	109
5.3 Περαιτέρω Έρευνα.....	110
Δικτυογραφία.....	112
Βιβλιογραφία	114
Παράρτημα A	119
Παράρτημα B	122

Περιεχόμενα Πινάκων και Σχημάτων

Πίνακας 2.1: Βασικές ετικέτες HTML

Πίνακας 2.2: Είδη Arduino

Πίνακας 2.3: Τεχνικά Χαρακτηριστικά WeMos D1 R2

Πίνακας 2.4: Σύνδεση - Λειτουργία των pins του WeMos D1 R2

Πίνακας 2.5: Σύνδεση - Πλακέτες WeMos D1

Πίνακας 2.6: Εργαλεία ανάπτυξης Arduino IDE

Πίνακας 2.7: Βασικές δομές & εντολές - Arduino IDE

Πίνακας 3.1: Τεχνικά Χαρακτηριστικά του DHT22/AM2302

Πίνακας 3.2: Περιγραφή λειτουργίας ακροδεκτών DHT22/AM2302

Πίνακας 3.3: Υπολογισμός κόστους υλικών μερών

Πίνακας 4.1: Πίνακας Users

Πίνακας 4.2: Πίνακας user_log

Πίνακας 4.3: Πίνακας devices_states

Πίνακας 4.4: Πίνακας Indoor_data

Πίνακας 4.5: Πίνακας outdoor_data

Σχήμα 1.1 Εκτιμώμενος αριθμός συνδεδεμένων συσκευών σε παγκόσμια κλίμακα έως το 2020

Σχήμα 1.2 Οικοσύστημα του Διαδικτύου των Πραγμάτων (IoT Ecosystem)

Σχήμα 2.1 Η ιστορία εκδόσεων της γλώσσας σήμανσης υπερκειμένου HTML

Σχήμα 2.2 Βασικό κομμάτι κώδικα για την εκκίνηση ενός εγγράφου HTML

Σχήμα 2.3 Κώδικας σύνδεσης ενός αρχείου CSS σε ένα έγγραφο HTML

Σχήμα 2.4 Κομμάτι κώδικα CSS για αλλαγή του χρώματος των παραγράφων σε κόκκινο

- Σχήμα 2.5 Ανατομία ενός κώδικα CSS
- Σχήμα 2.6 Κώδικας CSS τροποποίησης πολλαπλών ιδιοτήτων ενός στοιχείου
- Σχήμα 2.7 Κώδικας CSS τροποποίησης πολλαπλών στοιχείων ταυτόχρονα
- Σχήμα 2.8 Τρόπος επιλογής ενός στοιχείου προς μορφοποίηση - CSS
- Σχήμα 2.9 Τρόπος σύνταξης της συνάρτησης require_once στην PHP
- Σχήμα 2.10 Τρόπος σύνταξης της PHP
- Σχήμα 2.11 Τρόπος σύνταξης της JavaScript
- Σχήμα 2.12 Σύνδεση αρχείου JavaScript σε αρχείου HTML/PHP
- Σχήμα 2.13 Κομμάτι κώδικα σε JavaScript για την λειτουργικότητα του μενού της διαδικτυακής εφαρμογής Smart Homie
- Σχήμα 2.14 Λειτουργία και αλληλεπίδραση της MySQL
- Σχήμα 2.15 WeMos D1 R2
- Σχήμα 2.16 Δρομολογητής – Μόντεμ TG585 V7
- Σχήμα 2.17 Περιβάλλον Προγραμματισμού Visual Studio Code
- Σχήμα 2.18 Περιβάλλον Προγραμματισμού Brackets
- Σχήμα 2.19 Πίνακας Ελέγχου του XAMPP
- Σχήμα 2.20 Λογισμικό Arduino IDE
- Σχήμα 3.1 Διάγραμμα ροής συνδεσμολογίας εξαρτημάτων
- Σχήμα 3.2 Arduino Uno Rev3 και Ethernet Shield
- Σχήμα 3.3 WIZnet Ethernet W5100 Chipset & Schematic Diagram
- Σχήμα 3.4 WeMos D1 R2
- Σχήμα 3.5 ESP8266 microchip Schematic
- Σχήμα 3.6 Breadboard 400 Schematic
- Σχήμα 3.7 Ακρυλική Βάση
- Σχήμα 3.8 Σχηματική Απεικόνιση & Διαστάσεις του αισθητηρίου DHT22/AM2302
- Σχήμα 3.9 Διαδικασία επικοινωνίας DHT22/AM2302 με Arduino
- Σχήμα 3.10 Σχηματική Απεικόνιση ακροδεκτών DHT22/AM2302
- Σχήμα 3.11 Καλώδια Jumper
- Σχήμα 3.12 Καλώδιο micro USB σε USB
- Σχήμα 3.13 Καλώδιο USB Type-B σε USB
- Σχήμα 3.14 Καλώδιο Ethernet RJ45
- Σχήμα 3.15 Σχηματική απεικόνιση σύνδεσης Arduino Uno Rev3 και DHT22

- Σχήμα 3.16 Σύνδεση Arduino Uno Rev3 και Arduino Ethernet Shield
- Σχήμα 3.17 Ηλεκτρολογική σχηματική απεικόνιση συνδεσμολογίας
- Σχήμα 3.18 Τελική μορφή υποσυστήματος
- Σχήμα 3.19 Σχηματική απεικόνιση σύνδεσης WeMos D1 R2 και DHT22
- Σχήμα 3.20 Ηλεκτρολογική σχηματική απεικόνιση συνδεσμολογίας δεύτερου υποσυστήματος
- Σχήμα 3.21 Τελική μορφή δεύτερου υποσυστήματος
- Σχήμα 3.22 Διάγραμμα ροής λειτουργίας αισθητήρα DHT22/AM2302
- Σχήμα 4.1 Γραφικό περιβάλλον phpMyAdmin
- Σχήμα 4.2 Συσχετισμός πινάκων της βάσης δεδομένων
- Σχήμα 4.3 Αρχική σελίδα εφαρμογής
- Σχήμα 4.4 Καρτέλα Εγγραφής (Register)
- Σχήμα 4.5 Καρτέλα Εισόδου (Log In)
- Σχήμα 4.6 Τμήμα ένδειξης κατάστασης οικιακών συσκευών
- Σχήμα 4.7 Τμήμα εναλλαγής κατάστασης οικιακών συσκευών
- Σχήμα 4.8 Τμήμα εμφάνισης δραστηριότητα χρήστη
- Σχήμα 4.9 Καρτέλα Dashboard
- Σχήμα 4.10 Διαγράμματα παρακολούθησης τιμών σε ζωντανό χρόνο
- Σχήμα 4.11 Καρτέλα Live Monitoring
- Σχήμα 4.12 Καρτέλα Analytics: Επιλογή χώρων
- Σχήμα 4.13 Καρτέλα Analytics: Εναλλαγή σε διαφορετικό χώρο για εμφάνιση στατιστικών
- Σχήμα 4.14 Καρτέλα Analytics: Χρονικές περίοδοι μέτρησης
- Σχήμα 4.15 Καρτέλα Analytics: Αναλυτικός πίνακας μετρήσεων
- Σχήμα 4.16 Καρτέλα Profile: Πληροφορίες λογαριασμού
- Σχήμα 4.17 Καρτέλα Profile: Φόρμα ανανέωσης στοιχείων λογαριασμού
- Σχήμα 4.18 Καρτέλα Users: Πίνακας εγγεγραμμένων χρηστών
- Σχήμα 4.19 Καρτέλα Users: Μήνυμα επιτυχημένης διαγραφής χρήστη
- Σχήμα 4.20 Καρτέλα Users: Τροποποίηση δικαιωμάτων χρήστη
- Σχήμα 4.21 Καρτέλα Register User: Φόρμα δημιουργίας νέου χρήστη
- Σχήμα 4.22 Κεντρική μπάρα πληροφοριών
- Σχήμα 4.23 Αναδυόμενο παράθυρο αποσύνδεσης χρήστη

Σχήμα 4.24 Μπάρα μονοπατιού (Directory Path)

Σχήμα 4.25 Κεντρικό μενού περιήγησης

Σχήμα 4.26 Βοηθητικές λειτουργίες διαδικτυακής εφαρμογής

Σχήμα 4.27 Φόρμα τεχνικής υποστήριξης

Περίληψη

**Διαδικτυακή Εφαρμογή Παρακολούθησης & Διαχείρισης Έξυπνων Συσκευών:
Σχεδιασμός & Υλοποίηση για Έξυπνα Σπίτια**

Οι έξυπνες συσκευές (Smart Devices), τα έξυπνα σπίτια (Smart Homes) και η πληθώρα των ερευνητικών εξελίξεων στον τομέα του Διαδικτύου των πραγμάτων (IoT) καθιστούν επιτακτική την ανάγκη για νέες διαδικτυακές εφαρμογές. Η σύνδεση μικροελεγκτών και αισθητήρων για τη συλλογή δεδομένων αλλά και τον πλήρη έλεγχο των έξυπνων συσκευών γίνεται ολοένα και πιο σύνηθες φαινόμενο μέσα στα έξυπνα σπίτια.

Όλα αυτά ωθούν την ακαδημαϊκή και την επιχειρηματική ερευνητική προσπάθεια στην αναζήτηση νέων και έξυπνων τρόπων διαχείρισης των πληροφοριών που συλλέγονται από τους αισθητήρες καθώς και την ορθή χρήση των αποτελεσμάτων για εξοικονόμηση ενέργειας στις κατοικίες.

Στην παρούσα διπλωματική εργασία παρουσιάζεται η ανάπτυξη ενός συστήματος καταγραφής, αποθήκευσης και παρουσίασης των περιβαλλοντικών συνθηκών θερμοκρασίας και υγρασίας που επικρατούν σε ένα σπίτι. Ακόμη, δίνονται δυνατότητες ελέγχου των εγκατεστημένων έξυπνων συσκευών του σπιτιού με σκοπό την εξοικονόμηση ενέργειας. Η διπλωματική εργασία αποτελείται από ένα σύστημα μικροελεγκτών και αισθητήρων για τη συλλογή δεδομένων καθώς και από μία διαδικτυακή εφαρμογή για την παρουσίαση των αποτελεσμάτων στον χρήστη.

Αρχικά αναλύονται οι απαραίτητες θεωρητικές γνώσεις που αφορούν το υλικό και το λογισμικό και παρουσιάζεται βηματικά η διαδικασία για το σχεδιασμό και τη δημιουργία του τελικού συστήματος.

Ακολουθεί η σχεδίαση, ο προγραμματισμός και η υλοποίηση της διαδικτυακής εφαρμογής. Η εφαρμογή βασίζεται σε τεχνικές ανάπτυξης των πιο σύγχρονων γλωσσών προγραμματισμού καθώς και σε αρκετές βιβλιοθήκες οι οποίες αυξάνουν τις δυνατότητες παραμετροποίησης της εφαρμογής από τον χρήστη.

Η χρήση του συστήματος σε συνεχή βάση και σε πραγματικά περιβάλλοντα καταδεικνύουν τη δυνατότητα χρήσης και περαιτέρω εξέλιξής του.

Λέξεις κλειδιά: Διαδικτυακή εφαρμογή, αισθητήρες, θερμοκρασία, υγρασία, συλλογή δεδομένων, μικροελεγκτής, βάση δεδομένων, Arduino, WeMos, PHP, HTML, JavaScript, MySQL, XAMPP.

Abstract

Web Application for Monitoring & Managing Smart Devices: Design & Implementation for Smart Homes

Thesis submitted to the Department of Production and Management Engineering,
School of Engineering, Democritus University of Thrace, Greece, on October 2018

For the degree

Diploma in Production and Management Engineering (Dip. Eng.)

Supervisor: Assoc. Prof. Katsavounis Stefanos

Smart Devices, Smart Homes, and the plethora of recent advancement in the area of the Internet of Things, creates a need of new web applications. Connecting microcontrollers and sensors for collecting data, but also for complete control of smart devices is becoming a more common phenomenon in residential homes.

All these promote the academic and business research trials, in search of new and smart ways of managing information that are collected by sensors, as well as the correct use for best energy efficiency in houses.

This diploma thesis presents the development of a system to monitor, store, and present the environmental conditions of temperature and humidity that prevail in a home. The capability to control the installed smart devices also gives maximum energy efficiency. The diploma thesis is comprised of a system with microcontrollers and sensors for collecting data, and a web application for visualizing the results to the user.

Primarily, the theoretical knowledge about the hardware and software is analyzed, and the procedure of the design and the assembly of the system is presented.

The implementation, and the programming of the web application follows. The application is based on programming techniques of the most advanced programming languages and framework that increase the parameterizational capabilities.

The daily use of the system in a real environment, demonstrates its advantages, and the development furthermore.

Keywords: Web Application, sensors, temperature, humidity, data collection, microcontroller, database, Arduino, WeMos, PHP, HTML, JavaScript, MySQL, XAMPP.

Κεφάλαιο 1

Εισαγωγή

1.1 Το Πρόβλημα και η Σημασία του

Οι έννοιες των έξυπνων συσκευών (Smart Devices), των έξυπνων σπιτιών (Smart Homes), των έξυπνων συμπλεγμάτων (Smart Clusters), αλλά και του Διαδικτύου των πραγμάτων (Internet of Things – IoT), είναι ευρέως γνωστές και αποτελούν σημαντικό τμήμα στο οποίο εστιάζει η τεχνολογική έρευνα. Καθημερινά ολοένα και περισσότερες έξυπνες συσκευές εγκαθίστανται σε σπίτια, σε επιχειρήσεις, σε οργανισμούς ακόμα και σε δημόσια μέρη. Για την εύρυθμη λειτουργία των συσκευών αυτών, απαιτείται σωστή διαχείριση των δεδομένων, η συλλογή των δεδομένων αυτών σε καλά οργανωμένες βάσεις δεδομένων καθώς και η σωστή γραφική απεικόνιση τους είτε μέσω κάποιο προσαρμοσμένο λογισμικό, είτε μέσω κάποιας διαδικτυακής εφαρμογής.

Ως διαδικτυακή εφαρμογή ορίζεται μια εφαρμογή πελάτη/έξυπηρετητή (client/server) που χρησιμοποιεί ένα φυλλομετρητή (browser). Οι εφαρμογές αυτές συνήθως εκτελούνται σε ισχυρές υπολογιστικές μηχανές οι οποίες έχουν το ρόλο του σταθμού έξυπηρέτησης και παρέχουν τις υπηρεσίες τους σε περισσότερους του ενός χρήστη [1]. Προβάλλουν δυναμικό περιεχόμενο στηριγμένο στις απαιτήσεις του χρήστη. Μια τέτοιου τύπου εφαρμογή συνήθως δεν έχει στατικά στοιχεία προς προβολή, π.χ. μία στατική ιστοσελίδα, αλλά παρέχει δυναμικά στοιχεία προς προβολή και στις περισσότερες περιπτώσεις υπάρχει και η δυνατότητα αλληλεπίδρασης του χρήστη ώστε να επιτυγχάνεται η καλύτερη εμπειρία χρήστης (user experience). Έτσι, οι άνθρωποι για την καθημερινή τους διευκόλυνση καθώς και για την εξοικονόμηση ενέργειας που καταναλώνουν καθημερινά, έχουν αρχίσει να ενσωματώνουν τις έξυπνες συσκευές στους χώρους όπου διαβιούν μεγάλο μέρος της ημέρας. Τα σημαντικότερα ζητήματα ενέργειας που έρχονται να καλύψουν οι έξυπνες συσκευές είναι θέματα που αφορούν την εξοικονόμηση ρεύματος από ηλεκτρικές συσκευές που χρησιμοποιούνται για πολλές ώρες κάθε ημέρα, όπως τα κλιματιστικά, τα καλοριφέρ, η θέρμανση του νερού (μπάνιο, πλύσιμο των πιάτων) καθώς και το πότισμα των εξωτερικών χώρων.

Επίσης, χρησιμοποιούνται αισθητήρες μέτρησης της ποιότητας του αέρα, έτσι ώστε να μην υπάρχει μεγάλο ποσοστό υγρασίας & μόλυνσης του αέρα που αναπνέουμε στο σπίτι.

Όλες αυτές οι μετρήσεις αλλά και η σωστή απεικόνιση αυτών στον τελικό χρήστη ενθαρρύνουν σημαντικά τη δημιουργία μιας διαδικτυακής εφαρμογής η οποία θα είναι εύχρηστη, θα μπορεί να παρέχει τις απαραίτητες πληροφορίες που ζητά ο χρήστης, στο σωστό χρόνο και χωρίς να υπάρχουν προβλήματα στην παραμετροποίηση και τροποποίηση αυτών των δεδομένων για την καλύτερη εμπειρία χρήσης.

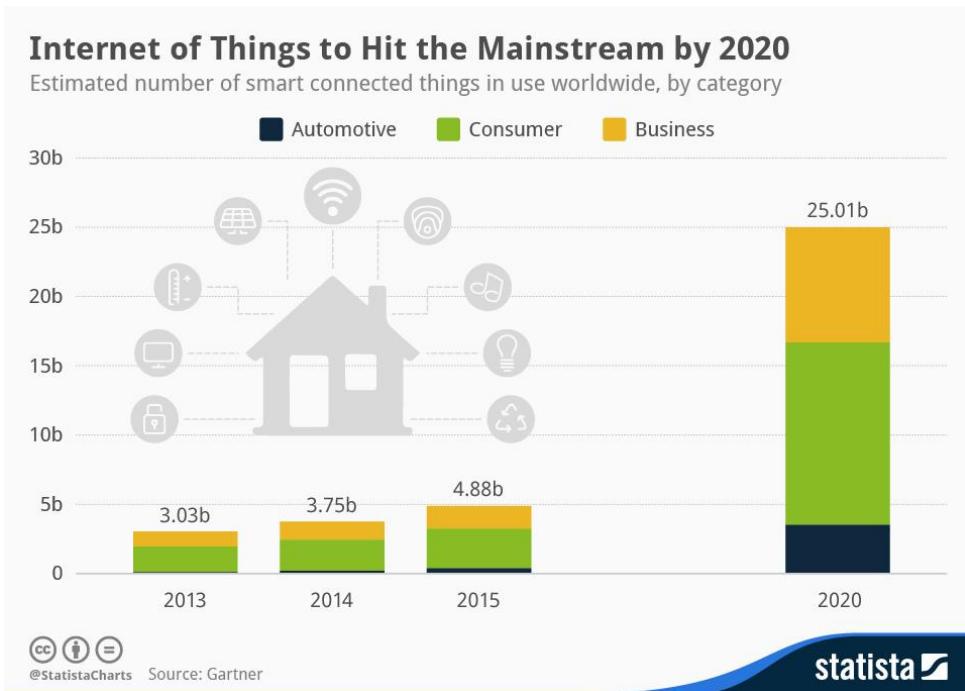
Σκοπός της παρούσας διπλωματικής εργασίας είναι η δημιουργία μιας διαδικτυακής εφαρμογής για την ανάπτυξη ενός συστήματος παρακολούθησης & διαχείρισης έξυπνων συσκευών με σκοπό τη μετατροπή ενός συμβατικού σπιτιού σε έξυπνο. Λαμβάνονται ως δεδομένα, πρωτογενείς μετρήσεις θερμοκρασίας και υγρασίας εντός και εκτός του σπιτιού που χρησιμοποιήθηκε για τη δημιουργία της παρούσας διπλωματικής εργασίας. Μέσα από την διαδικτυακή εφαρμογή, ο χρήστης μπορεί να παρακολουθεί σε ζωντανό χρόνο τις μετρήσεις που αφορούν το σπίτι του, να λαμβάνει στατιστικά στοιχεία των τελευταίων μηνών καθώς και να παίρνει αποφάσεις για την ενεργοποίηση/απενεργοποίηση έξυπνων συσκευών που έχουν τοποθετηθεί στο σπίτι.

1.2 Internet of Things (IoT)

Το Διαδίκτυο των Πραγμάτων είναι ένα δίκτυο από φυσικές συσκευές, κινούμενα οχήματα, σπίτια και άλλα ενσωματωμένα συστήματα με ηλεκτρονικά, λογισμικό, αισθητήρια, ενεργοποιητές, τα οποία μπορούν να συνδεθούν μεταξύ τους, να συλλέξουν αλλά και να ανταλλάξουν δεδομένα. Με το IoT τα αντικείμενα αυτά μπορούν να ελέγχονται απομακρυσμένα μέσω της υπάρχουσας δικτυακής υποδομής δημιουργώντας ευκαιρίες άμεσης ενσωμάτωσης του φυσικού κόσμου με τα υπολογιστικά συστήματα. Λόγω του εξοπλισμού με αισθητήρες και ενεργοποιητές το IoT είναι μέρος των έξυπνων συστημάτων της καθημερινότητας όπως είναι τα έξυπνα σπίτια και πόλεις. Λόγω του ενσωματωμένου υπολογιστικού συστήματος κάθε αντικείμενο αναγνωρίζεται μοναδικά και μπορεί να λειτουργεί αυτόνομα αλλά και σε συνεργασία με την υπόλοιπη Διαδικτυακή υποδομή [2][27].

Ο όρος “Διαδίκτυο των Πραγμάτων” η αλλιώς Internet of Things (IoT) χρησιμοποιήθηκε πρώτη φορά το 1999 από τον Βρετανό πρωτοπόρο στην τεχνολογία Kevin Ashton, για την περιγραφή ενός συστήματος στο οποίο, τα αντικείμενα του φυσικού κόσμου θα μπορούσαν να συνδεθούν με το Διαδίκτυο μέσω αισθητήρων. Σήμερα το IoT έχει γίνει ένας δημοφιλής όρος για την περιγραφή σεναρίων στα οποία η σύνδεση στο Διαδίκτυο και οι δυνατότητες των υπολογιστών επεκτείνονται σε μια ποικιλία από αντικείμενα όπως συσκευές, αισθητήρες και ήδη καθημερινής χρήσης. Παρά το γεγονός ότι, ο όρος IoT είναι σχετικά νέος, η έννοια του συνδυασμού υπολογιστών και των δικτύων για την παρακολούθηση και των έλεγχο των συσκευών υπάρχει εδώ και δεκαετίες [29].

Υπολογίζεται, σύμφωνα με την Statista, η ανάπτυξη του IoT έχει ξεκινήσει αρχίσει να αναπτύσσεται ραγδαία και θα φτάσει σε πολύ μεγάλα χρηματικά ποσά μέχρι το 2020, σύμφωνα με το οποίο και οι επιχειρήσεις αλλά και οι καταναλωτές ενσωματώνουν τις έξυπνες τεχνολογίες σε μεγάλη κλίμακα [3].



Σχήμα 1.1 Εκτιμώμενος αριθμός συνδεδεμένων συσκευών σε παγκόσμια κλίμακα έως το 2020

Περαιτέρω, το IoT βοηθά στη σύνδεση αντικειμένων ή διαδικασιών οι οποίες επιτρέπουν την καλύτερη παρακολούθηση και αλληλεπίδραση. Υπάρχουν τέσσερα στρώματα της IoT αρχιτεκτονικής:

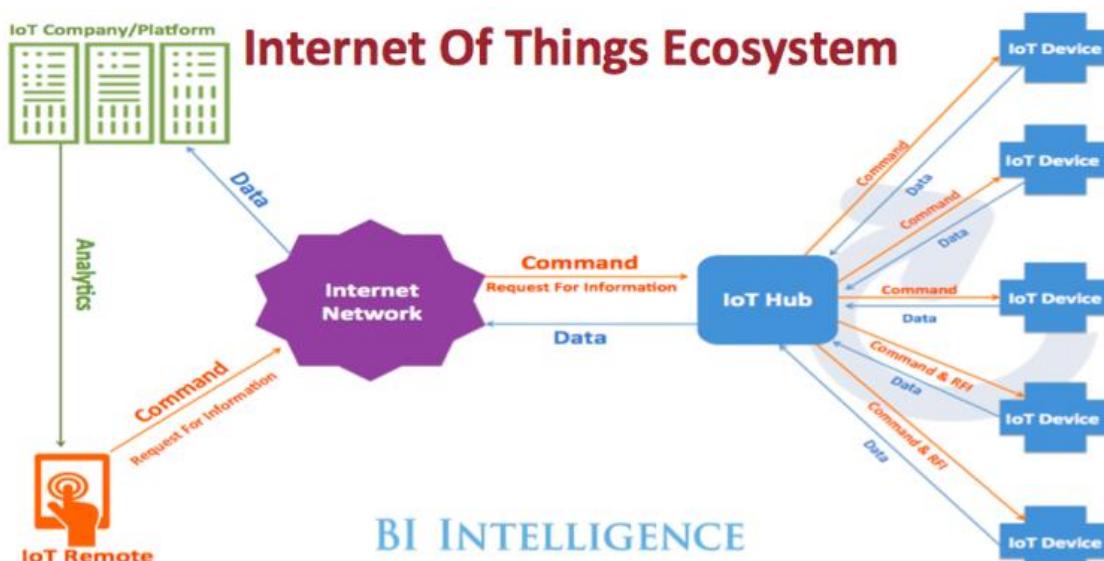
- Το στρώμα ανίχνευσης ή αντίληψης, το οποίο συλλέγει και μεταδίδει δεδομένα με τη βοήθεια ιδιοτήτων εντοπισμού αντικειμένων και εργαλείων ανάγνωσης.
- Το στρώμα δικτύου και πύλης, το οποίο συνδέει αντικείμενα ή πράγματα και τους επιτρέπει να κοινοποιούν και να ανταλλάσσουν πληροφορίες με τη βοήθεια ειδικών πυλών, όπως ένα εσωτερικό δίκτυο ή ένα τοπικό δίκτυο και ένα εξωτερικό δίκτυο ή ένα δίκτυο ευρείας περιοχής.
- Το στρώμα διαχείρισης υπηρεσιών, το οποίο είναι υπεύθυνο για τις αναλύσεις των πληροφοριών, τον έλεγχο της ασφάλειας, τη μοντελοποίηση της διαδικασίας και τη διαχείριση των συσκευών.
- Το στρώμα εφαρμογών, το οποίο διατηρεί τα δεδομένα που συλλέγονται και μεταφέρονται, υπόκεινται σε επεξεργασία μέσω συγκεκριμένων τεχνικών που χρησιμοποιούνται για ειδικά σχέδια. Επιπλέον, αφορά τη διαχείριση αντικειμένων και πραγμάτων [30].

Παράδειγμα IoT Εφαρμογής: Το έξυπνο σπίτι:

Τα μελλοντικά έξυπνα σπίτια θα έχουν επίγνωση του τι συμβαίνει μέσα σε ένα κτίριο, επηρεάζοντας κυρίως τρεις τομείς:

- Την κατανάλωση πόρων, που αφορούν τη διατήρηση και κατανάλωση ενέργειας
- Την Ασφάλεια
- Την Άνεση

Στόχος αυτών, είναι η επίτευξη των καλύτερων επιπέδων άνεσης μειώνοντας ταυτόχρονα τις συνολικές δαπάνες. Επιπλέον, τα έξυπνα σπίτια αντιμετωπίζουν επίσης ζητήματα ασφάλειας μέσω σύνθετων συστημάτων ασφαλείας για την ανίχνευση κλοπής, πυρκαγιάς ή μη εξουσιοδοτημένης εισόδου. Οι φορείς που εμπλέκονται σε αυτό το σενάριο αποτελούν μια πολύ ετερογενή ομάδα. Διαφορετικοί φορείς θα συνεργαστούν στο σπίτι του χρήστη, όπως εταιρείες του διαδικτύου, κατασκευαστές συσκευών, τηλεπικοινωνιακοί φορείς, υπηρεσίες οπτικοακουστικών μέσων προμηθευτές, εταιρείες ασφάλειας, εταιρείες ηλεκτρισμού.[3]



Σχήμα 1.2 Οικοσύστημα των Διαδικτύου των Πραγμάτων (IoT Ecosystem)

1.3 Στόχοι της Διπλωματικής Εργασίας

Ο στόχος της παρούσας διπλωματικής εργασίας είναι η παρακολούθηση των εσωτερικών αλλά και εξωτερικών θερμοκρασιών ενός σπιτιού και η δημιουργία μιας διαδικτυακής εφαρμογής έτσι ώστε ο χρήστης να μπορεί να βγάλει συμπεράσματα ανάλογα με τα δεδομένα που συλλέγονται από τα αισθητήρια όργανα, πότε είναι η κατάλληλη στιγμή να ενεργοποιήσει ή να απενεργοποιήσει τις έξυπνες συσκευές που υπάρχουν στο σπίτι του έτσι ώστε να πετύχει τη μέγιστη ωφέλιμη ενεργειακή κατανάλωση. Μέσω της εφαρμογής ο χρήστης θα είναι σε θέση να παρακολουθήσει στατιστικά των μετρήσεων για τον τελευταίο μήνα καθώς και να παρακολουθεί τις ελάχιστες και τις μέγιστες τιμές ανά ημέρα.

Για την εκπλήρωση αυτού του στόχου είναι απαραίτητες δύο κύριες δραστηριότητες.

Η πρώτη είναι η αγορά των αισθητηρίων, των μικροελεγκτών καθώς και ενός Η/Υ ο οποίος θα χρησιμοποιηθεί ως τοπικός εξυπηρετητής (Local Host Server). Για τη μείωση του όγκου αλλά και λόγω των περιορισμένων απαιτήσεων που χρειάζεται η διαδικτυακή εφαρμογή μπορεί να γίνει αντικατάσταση του Η/Υ με ένα μικρότερο μονής μονάδας υπολογιστή όπως είναι το Raspberry Pi.

Η δεύτερη δραστηριότητα αφορά τη δημιουργία της διαδικτυακής εφαρμογής. Σε αυτήν τη φάση, εφόσον έχουν ελεγχθεί όλα τα κομμάτια του υλικού (Hardware) που απαιτούνται και πληρούν τις προτεινόμενες απαιτήσεις συστήματος πρέπει να ξεκινήσει η ανάπτυξη της εφαρμογής. Αναπτύσσονται οι απαραίτητοι αλγόριθμοι συλλογής των δεδομένων από τα αισθητήρια και αποθηκεύονται στις βάσεις δεδομένων. Έπειτα γίνεται η γραφική αναπαράσταση των δεδομένων προς τον χρήστη και έτσι του δίνεται η δυνατότητα επιλογής του τι θέλει να κάνει με τις έξυπνες συσκευές στο σπίτι του.

Τέλος σκοπός είναι να δοθεί μία απάντηση στο ερώτημα εάν μία διαδικτυακή εφαρμογή όπως η Smart Homie κρίνεται αναγκαία για την παρακολούθηση τέτοιων συστημάτων όπως είναι οι έξυπνες συσκευές μέσα στο σπίτι του καθημερινού χρήστη. Ακόμη πρέπει να εξακριβωθούν οι συνθήκες στις οποίες ενδείκνυται μία τέτοια διαδικτυακή εφαρμογή και κατά πόσο εναρμονίζεται με τις υπόλοιπες συσκευές του σπιτιού χωρίς να δημιουργεί πρόβλημα στη χρήση του σπιτιού.

1.4 Δομή της Διπλωματικής Εργασίας

Η δομή της διπλωματικής εργασίας είναι η εξής:

Στο Κεφάλαιο 1 παρουσιάζεται το πρόβλημα που εξετάζει η διπλωματική εργασία καθώς και πληροφορίες για τον τρόπο λειτουργίας της διαδικτυακής εφαρμογής καθώς και για τα ζητήματα που είναι απαραίτητα να λυθούν πριν την εγκατάσταση και χρήση αυτής.

Στο Κεφάλαιο 2 παρουσιάζονται οι βασικές έννοιες καθώς και το υλικό μέρος (Hardware) που θα χρησιμοποιηθεί για την παρούσα διπλωματική εργασία. Πιο συγκεκριμένα, αναλύονται οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για να ανάπτυξη και κωδικοποίηση της διαδικτυακής εφαρμογής, οι μικροελεγκτές και ο δρομολογητής-μόντεμ που χρησιμοποιήθηκαν για τη δημιουργία του τοπικού συστήματος & δικτύου και τέλος αναλύονται οι πλατφόρμες πάνω στις οποίες αναπτύχθηκε η διαδικτυακή εφαρμογή.

Στο Κεφάλαιο 3 αναλύονται εκτενώς το υλικό μέρος καθώς και η συνδεσμολογία τους συστήματος. Παρατίθενται τα σχεδιαγράμματα του κάθε μικροελεγκτή, των αισθητηρίων καθώς και οι βάσεις τοποθέτησης των παραπάνω υλικών για την εύρυθμη λειτουργία του συστήματος. Ακόμη, αναλύεται ο τρόπος σύνδεσης όλων των παραπάνω υλικών έτσι ώστε να είναι σε θέση να δεχθούν την κατάλληλη παραμετροποίηση και τους κατάλληλους αλγορίθμους ώστε να λειτουργήσουν, να συλλέξουν αλλά και να αποστείλουν δεδομένα. Έπειτα, γίνεται η σύνδεση με τον τοπικό εξυπηρετητή. Τέλος αναλύεται το κόστος των εξαρτημάτων που χρησιμοποιήθηκαν στην παρούσα διπλωματική εργασία.

Στο Κεφάλαιο 4 γίνεται ανάλυση της σχεδίασης και της υλοποίησης τη διαδικτυακής εφαρμογής. Αρχικά, εξετάζονται οι απαιτήσεις του συστήματος, η τοπική λειτουργία της εφαρμογής και η προστασία των προσωπικών δεδομένων του χρήστη. Έπειτα γίνεται περιγραφή του τρόπου δημιουργίας της βάσης δεδομένων, της αρχιτεκτονικής της, καθώς και ο τρόπος επεξεργασίας της από τη διαδικτυακή εφαρμογή. Επιπλέον, γίνεται μία λεπτομερής ανάλυση για καθεμία από τις καρτέλες που απαρτίζουν την εφαρμογή.

Στο Κεφάλαιο 5 γίνεται μία ανασκόπηση της λειτουργικότητας της διαδικτυακής εφαρμογής, εξαγωγή και ανάλυση των συμπερασμάτων που προέκυψαν κατά την διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας. Αναφέρονται τα προβλήματα που αντιμετωπίστηκαν όπως και προτάσεις για μελλοντική έρευνα.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο Εννοιών και Εξοπλισμού

Η παρούσα διπλωματική εργασία, στην προσπάθειά της να δημιουργήσει μία διαδικτυακή εφαρμογή η οποία θα προσφέρει την καλύτερη δυνατή εμπειρία χρήσης για την διαχείριση έξυπνων συσκευών, διαχωρίζεται καταρχήν σε δύο μέρη, ενώ στη συνέχεια ακολουθεί η διασύνδεσή τους.

Αρχικά, το υλικό μέρος (Hardware), το οποίο περιλαμβάνει τα αισθητήρια όργανα καθώς και τη συνδεσμολογία αυτών για τη λήψη των απαραίτητων δεδομένων. Το δεύτερο μέρος, αποτελείται από την κωδικοποίηση/παραμετροποίηση των αισθητηρίων καθώς και την ανάπτυξη του λογισμικού (Software) με τη βοήθεια του οποίου συμπληρώνεται η διαδικτυακή εφαρμογή. Για την ενκολότερη κατανόηση των τρόπου σύνδεσης των δύο αυτών μερών, στο κεφάλαιο αυτό παρουσιάζονται βασικές θεωρητικές γνώσεις και έννοιες που θα χρησιμοποιηθούν στη συνέχεια. Γίνεται εκτενής αναφορά στις γλώσσες προγραμματισμού που θα χρησιμοποιηθούν, στον υλικό εξοπλισμό που είναι απαραίτητος καθώς και στις πλατφόρμες στις οποίες αναπτύχθηκε η εφαρμογή, και στις υπηρεσίες που παρέχουν για τη διασφάλιση της ποιότητας της εφαρμογής.

2.1 Γλώσσες Προγραμματισμού Διαδικτυακής Εφαρμογής

Μια γλώσσα προγραμματισμού είναι μια επίσημη γλώσσα, η οποία περιλαμβάνει ένα σύνολο οδηγιών που χρησιμοποιούνται για την παραγωγή διαφόρων ειδών εξόδου. Οι γλώσσες προγραμματισμού χρησιμοποιούνται για τη δημιουργία προγραμμάτων που υλοποιούν συγκεκριμένους αλγόριθμους.

Οι περισσότερες γλώσσες προγραμματισμού αποτελούνται από οδηγίες για υπολογιστές, παρόλο που υπάρχουν προγραμματιζόμενες μηχανές που χρησιμοποιούν περιορισμένο σύνολο συγκεκριμένων οδηγιών αντί για τις γενικές γλώσσες προγραμματισμού των σύγχρονων υπολογιστών [4].

2.1.1. Γλώσσα Σήμανσης Υπερκειμένου - HTML

Η HTML (αρχικοποίηση του αγγλικού όρου HyperText Markup Language) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων. Η HTML γράφεται υπό μορφή στοιχείων, τα οποία αποτελούνται από ετικέτες (tags), οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (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, πίνακες, εικόνες κλπ. Ο σκοπός ενός φυλλομετρητή είναι να διαβάζει τα έγγραφα HTML και να τα συνθέσει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο φυλλομετρητής δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να παρουσιάσει το περιεχόμενο της σελίδας.

Στις μέρες μας το πρότυπο που χρησιμοποιείται για την HTML είναι το HTML5.2 το οποίο ανακοινώθηκε από την κοινοπραξία World Wide Web (W3C) το 2017. Παρακάτω παρατίθεται η ιστορία της HTML.

HTML History

Year	Version
1989	Tim Berners-Lee invented www
1991	Tim Berners-Lee invented HTML
1993	Dave Raggett drafted HTML+
1995	HTML Working Group defined HTML 2.0
1997	W3C Recommendation: HTML 3.2
1999	W3C Recommendation: HTML 4.01
2000	W3C Recommendation: XHTML 1.0
2008	WHATWG HTML5 First Public Draft
2012	WHATWG HTML5 Living Standard
2014	W3C Recommendation: HTML5
2016	W3C Candidate Recommendation: HTML 5.1
2017	W3C Recommendation: HTML5.1 2nd Edition
2017	W3C Recommendation: HTML5.2

Σχήμα 2.1 Η ιστορία εκδόσεων της γλώσσας σήμανσης υπερκειμένου HTML

Η HTML κυρίως παράγεται ως έγγραφο. Μπορεί οι προγραμματιστές να κάνουν λόγο για κομμάτια κώδικα HTML, αλλά στις περισσότερες περιπτώσεις, μία ιστοσελίδα θα απαντήσει σε ένα αίτημα που έγινε από τον διακομιστή ή τον χρήστη με ένα έγγραφο HTML, το οποίο μετά παραδίδεται στον χρήστη και εμφανίζεται σε μία συσκευή [28].

Παρακάτω εμφανίζεται ένα παράδειγμα κώδικα (HTML snippet) για το βασικό κώδικα εκκίνησης της HTML.

```

1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>Document</title>
8  </head>
9  <body>
10
11 </body>
12 </html>

```

Σχήμα 2.2 Βασικό τμήμα κώδικα για την εκκίνηση ενός εγγράφου HTML

Στην προσπάθεια που γίνεται για τη σωστή ανάπτυξη του κώδικα και την εύρυθμη λειτουργία των HTML εγγράφων, είναι απαραίτητο να χρησιμοποιούνται οι κατάλληλες ετικέτες (tags) ανάλογα με τον τρόπο που θέλει ο προγραμματιστής να επεξεργασθεί τα δεδομένα που είναι προς εμφάνιση στον τελικό χρήστη. Οι σημαντικότερες ετικέτες html που χρησιμοποιούνται για την δημιουργία των HTML εγγράφων την παρούσας διπλωματικής εργασίας, παρουσιάζονται στον παρακάτω πίνακα.

Πίνακας 2.1: Βασικές ετικέτες HTML

Στοιχείο (tag)	Λειτουργία
<!---->	Η ετικέτα <!----> αφορά τα σχόλια τα οποία γράφει ο προγραμματιστής. Με τα σχόλια μπορεί να γίνει ευκολότερη η δόμηση ενός εγγράφου σε μικρότερες υποενότητες καθώς και γίνει επεξήγηση ενός τμήματος κώδικα ή και να τοποθετηθούν σημειώσεις για την μετέπειτα επεξεργασία του εγγράφου.
<!DOCTYPE>	Η δήλωση του DOCTYPE πρέπει να είναι η πρώτη σε ένα έγγραφο HTML. Είναι μια εντολή στο πρόγραμμα περιήγησης για το ποια είναι η έκδοση της HTML στο έγγραφο.
<a>	Η ετικέτα <a> καθορίζει έναν υπερσύνδεσμο, ο οποίος χρησιμοποιείται για τη σύνδεση μίας σελίδας προς μία άλλη. Το πιο σημαντικό στοιχείο αυτής της ετικέτας είναι το href, το οποίο υποδεικνύει τον προορισμό του συνδέσμου.

<code><article></code>	Η ετικέτα <code><article></code> καθορίζει ένα ανεξάρτητο, αυτό-συντηρούμενο περιεχόμενο. 
<code><aside></code>	Η ετικέτα <code><aside></code> καθορίζει περιεχόμενο , πέρα από το ήδη υπάρχον περιεχόμενο των άλλων ετικετών. Το περιεχόμενο αυτό πρέπει να συσχετίζεται με τα υπόλοιπα περιεχόμενα. 
<code><blockquote></code>	Η ετικέτα <code><blockquote></code> καθορίζει μία περιοχή η οποία αναφέρθηκε από μία άλλη πηγή. Οι φυλλομετρητές συνήθως δημιουργούν μία εσοχή όταν βρίσκουν την ετικέτα αυτή. Είναι σύνηθες φαινόμενο η συγκεκριμένη ετικέτα να χρησιμοποιείται για την παρουσίαση τμημάτων κώδικα.
<code><body></code>	Η ετικέτα <code><body></code> ορίζει το σώμα του εγγράφου. Όλα τα περιεχόμενα ενός εγγράφου HTML, όπως κείμενο, υπερσύνδεσμοι, εικόνες, πίνακες, λίστες, κτλ. Βρίσκονται μέσα σε αυτό.
<code><button></code>	Η ετικέτα <code><button></code> καθορίζει ένα κουμπί το οποίο μπορεί να πατηθεί. Μέσα στην ετικέτα, μπορούμε να προσθέσουμε περιεχόμενο, όπως κείμενο ή φωτογραφίες. Αυτή είναι η διαφορά μεταξύ των κουμπιών που δημιουργούνται με την ετικέτα <code><button></code> και με την ετικέτα <code><input></code> .
<code><canvas></code>	Η ετικέτα <code><canvas></code> χρησιμοποιείται για τη δημιουργία δυναμικών γραφικών (on the fly), μέσω του scripting (συνήθως με την γλώσσα JavaScript) 
<code><div></code>	Η ετικέτα <code><div></code> καθορίζει ένα τμήμα σε ένα έγγραφο HTML. Χρησιμοποιείται για την ομαδοποίηση των στοιχείων, ώστε να δοθούν κοινές μορφοποιήσεις με τη χρήση του Id.
<code><footer></code>	Η ετικέτα <code><footer></code> καθορίζει το υποσέλιδο για ένα αντικείμενο ή μία περιοχή. Η ετικέτα, θα πρέπει να περιέχει να περιέχει πληροφορίες για τα στοιχεία που περιβάλλει.

<form>	Η ετικέτα <form> χρησιμοποιείται για τη δημιουργία φορμών, με τις οποίες αλληλοεπιδρούν οι χρήστες. Χρησιμοποιείται επίσης για να μεταβιβάζει δεδομένα σε ένα διακομιστή.
<h1> έως <h6>	Οι ετικέτες <h1> έως <h6> χρησιμοποιούνται για τον καθορισμό του μεγέθους των γραμμάτων για τις επικεφαλίδες.
<head>	Η ετικέτα <head> είναι βασική, καθώς στο πεδίο της ορίζονται δεδομένα όπως το είδος των χαρακτήρων που χρησιμοποιούνται (UTF-8, ANSI κτλ), εξωτερικές πηγές αρχείων κτλ.
<header> 	Η ετικέτα <header> αναπαριστά μία περιοχή για εισαγωγικά περιεχόμενα ή ένα μενού.
<input>	Η ετικέτα <input> χρησιμοποιείται για την εισαγωγή πληροφοριών από τον χρήστη. Οι ετικέτες αυτές, περιλαμβάνονται σε μια <form>. Το χαρακτηριστικό type δηλώνει τη μορφή της ετικέτας input (type=text/ password/ button/ radio/ checkbox/ number/ email/ date).
<script>	Η ετικέτα script χρησιμοποιείται για να καθορίσει ένα script στην πλευρά του χρήστη. Η ετικέτα, είτε περιέχει τμήματα κώδικα, είτε συνδέει το παρόν έγγραφο με ένα εξωτερικό αρχείο script μέσω του στοιχείου src. Η βασική γλώσσα scripting είναι η JavaScript.
<table>	Η ετικέτα <table> ορίζει έναν πίνακα HTML. Ο πίνακας αποτελείται από ένα ή περισσότερα <tr>, <td> ή <th>.

2.1.2. Γλώσσα Μορφοποίησης – CASCADING STYLES SHEETS (CSS)

Η γλώσσα μορφοποίησης CSS (Cascading Style Sheets - Διαδοχικά Φύλλα Ύφους) ή (αλληλουχία φύλλων ύφους) είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων ύφους που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης [5]. Σύμφωνα με τα πρότυπα αυτής της γλώσσας γίνονται οι απαραίτητες μορφοποιήσεις στις ετικέτες html του εγγράφου έτσι ώστε να παραχθεί το επιθυμητό για τον χρήστη αποτέλεσμα στον φυλλομετρητή. Στις μέρες μας δεν νοείται η δημιουργία ιστοσελίδων ή διαδικτυακών εφαρμογών χωρίς την γλώσσα CSS καθώς οι δυνατότητες που προσφέρει αυτή η γλώσσα παράγουν ένα πολύ όμορφο αποτέλεσμα.

Η CSS είναι προορισμένη να αναπτύσσει τη διαμόρφωση (στυλ) μιας ιστοσελίδας, δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και να δίνει περισσότερες δυνατότητες σε σχέση με την HTML και δημιουργήθηκε για να λύσει ένα μεγάλο πρόβλημα που παρουσίαζε η HTML.

Σύνταξη της CSS

Η σύνταξη της CSS μπορεί να ξεκινήσει από πολύ απλούς κανόνες αλλά και να γίνει τόσο περίπλοκη που μπορεί να μην μπορεί εύκολα να κατανοηθεί από έναν αρχάριο. Η σύνταξη των κανόνων μορφοποίησης των ετικετών μπορεί να διαφέρουν από ένα έγγραφο σε ένα άλλο, γι' αυτό το λόγο δημιουργούνται περισσότερα από ένα αρχεία με την κατάληξη .css. Για να το πετύχουμε αυτό, αρκεί να εισάγουμε στο έγγραφο HTML, ανάμεσα στις ετικέτες <head> και </head> την απαραίτητη σύνδεση του αρχείο .css που έχουμε δημιουργήσει. Πρέπει να αναφέρουμε πώς τηρείται η σειρά προτεραιότητας με την οποία συνδέουμε στο έγγραφο HTML μας τα εξωτερικά αρχεία CSS, με αυτό να επιφέρει αλλαγές στον τρόπο με τον οποίο επηρεάζονται οι ετικέτες. Η σύνδεση φαίνεται στην παρακάτω εικόνα.

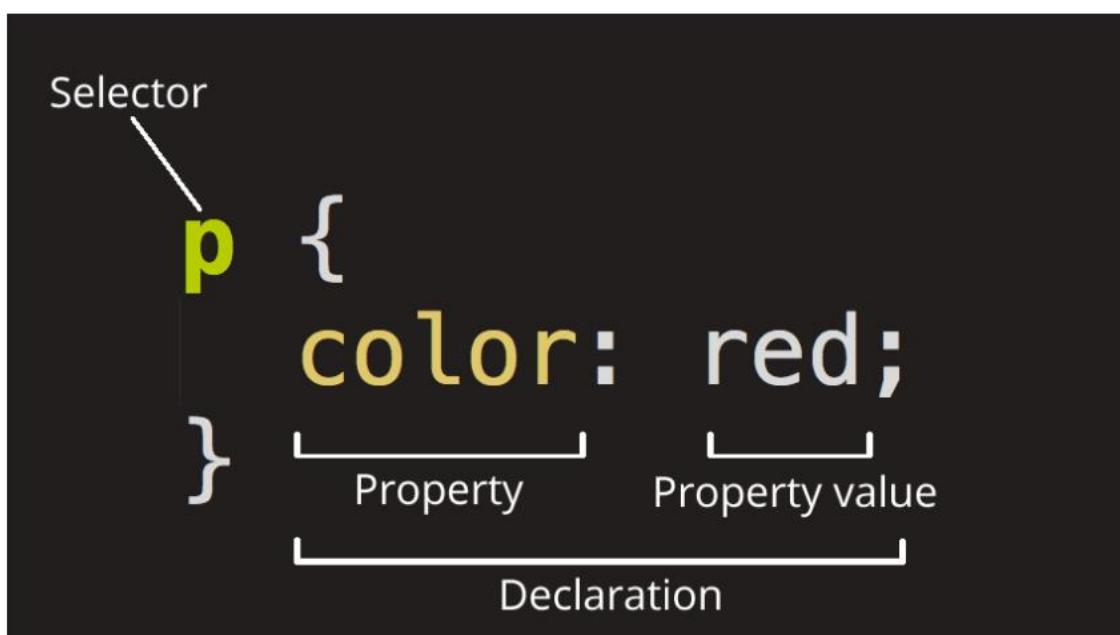
```
1 | <link href="styles/style.css" rel="stylesheet" type="text/css">
```

Σχήμα 2.3 Κώδικας σύνδεσης ενός αρχείου CSS σε ένα έγγραφο HTML

Για παράδειγμα, για να επιλέξουμε όλα τα στοιχεία παραγράφου σε μια σελίδα HTML και να μετατρέψουμε το κείμενο μέσα σε αυτά κόκκινο, θα γράψουμε αυτό το τμήμα κώδικα CSS:

```
1 | p {  
2 |   color: red;  
3 | }
```

Σχήμα 2.4 Κομμάτι κώδικα CSS για αλλαγή του χρώματος των παραγράφων σε κόκκινο



Σχήμα 2.5 Ανατομία ενός κώδικα CSS

Κρίνεται αναγκαίο στο συγκεκριμένο σημείο, να κάνουμε λόγο για την ανατομία ενός κανόνα CSS. Κάθε ετικέτα HTML την οποία θέλουμε να επεξεργασθούμε και να διαμορφώσουμε το στιλ της με τέτοιο τρόπο, ώστε να παραχθεί το επιθυμητό αποτέλεσμα πρέπει να γράφει με τέτοιο τρόπο, όπως φαίνεται στην παρακάτω εικόνα.

Ο κανόνας χωρίζεται σε τμήματα για την καλύτερη ανάγνωση του από τους ανθρώπους αλλά και για την εύρυθμη λειτουργία μίας ιστοσελίδας αλλά και μίας διαδικτυακής εφαρμογής. Ο διαχωρισμός γίνεται ως εξής:

- **Selector:** Ο selector είναι το όνομα του στοιχείου HTML και βρίσκεται στην αρχή του σετ κανόνων. Είναι υπεύθυνος για την επιλογή των στοιχείων τα οποία θα διαμορφωθούν με το κατάλληλο στιλ (στην περίπτωση μας, τα στοιχεία <p>). Για να αποδώσουμε στιλ σε άλλο στοιχείο, αρκεί να αλλάξουμε τον selector.
- **Declaration:** Ένας απλός κανόνας όπως `color: red;` καθορίζει σε ποιες ιδιότητες θα αποδώσουμε στιλ, από το στοιχείο που επιλέξαμε.
- **Properties:** Είναι οι τρόποι με τους οποίους μπορούμε να δώσουμε στιλ στα HTML στοιχεία. Στην CSS, μπορούμε να διαλέξουμε ποιες ιδιότητες θα επηρεαστούν από τον κανόνα που θέτουμε.
- **Property Value:** Στα δεξιά της ιδιότητας και μετά τα εισαγωγικά, έχουμε την τιμή της ιδιότητας, η οποία μας δίνει την επιθυμητή εμφάνιση.

Ακόμη, για να τροποποιήσουμε πολλαπλές τιμές με μία προσπάθεια, απλά χρειάζεται να τις γράψουμε, την μία κάτω από την άλλη και χωρισμένες με ερωτηματικό, όπως φαίνεται παρακάτω.

```

1 | p {
2 |   color: red;
3 |   width: 500px;
4 |   border: 1px solid black;
5 |

```

Σχήμα 2.6 Κώδικας CSS τροποποίησης πολλαπλών ιδιοτήτων ενός στοιχείου

Μπορούμε επίσης να επιλέξουμε πολλαπλούς τύπους στοιχείων και να τους αποδώσουμε έναν κανόνα συνολικά. Μπορούμε να εισάγουμε πολλαπλούς selectors χωρισμένους με κόμμα όπως φαίνεται στο Σχήμα 2.7 [6] .

```

1 | p, li, h1 {
2 |   color: red;
3 |

```

Σχήμα 2.7 Κώδικας CSS τροποποίησης πολλαπλών στοιχείων ταυτόχρονα

Τέλος, για την καλύτερη παραμετροποίηση αλλά και των διαχωρισμό των στοιχείων HTML, η CSS έχει την δυνατότητα να προσδίδει στα στοιχεία ένα διαχωρισμό στον

τρόπο που εφαρμόζει τους κανόνες μορφοποίησης. Υπάρχουν δύο τρόποι με τους οποίους μπορούμε να διαλέξουμε ένα στοιχείο, είτε προσδίδοντας του ένα ID είτε μία κλάση. Η επιλογή με ID, χρησιμοποιείται για να μορφοποιηθεί ένα συγκεκριμένο και μοναδικό στοιχείο html. Αυτή η επιλογή χρησιμοποιεί την ιδιότητα id του στοιχείου και ορίζεται με ένα #. Η απόδοση κλάσης, χρησιμοποιείται για να μορφοποιήσει μια ομάδα από στοιχεία της HTML. Αυτό επιτρέπει την χρήση της ίδιας μορφοποίησης σε πολλά στοιχεία ταυτόχρονα που έχουν την ίδια ιδιότητα class. Γι' αυτήν την επιλογή στην CSS χρησιμοποιείται το σύμβολο “.”. Στο παρακάτω Σχήμα παρουσιάζεται ο τρόπος επιλογής ενός στοιχείου προς μορφοποίηση.

```

1  #myID {
2      color: red;
3      font-size: 16px;
4      border: 1px solid black;
5  }
6
7  .myClass {
8      color: aqua;
9      font-size: 18px;
10     border: 2px solid green;
11 }
12

```

Σχήμα 2.8 Τρόπος επιλογής ενός στοιχείου προς μορφοποίηση - CSS

2.1.3. Γλώσσα Επικοινωνίας Χρήστη – Διακομιστή PHP

Η γλώσσα PHP το οποίο είναι ακρολέξιο του Personal Home Page: Hypertext Preprocessor, αποτελεί μία scripting [βασισμένη σε scripts (σενάρια)] γλώσσα υπολογιστή - εξυπηρετητή. Είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που είτε θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML ή θα επεξεργασθεί τις εισόδους δίχως να προβάλλει την έξοδο στο χρήστη, αλλά θα τις μεταβιβάσει σε κάποιο άλλο PHP script [7]. Η γλώσσα αυτή εκτελείται από την πλευρά του διακομιστή, όπως η ASP της Microsoft με τη διαφορά ότι είναι ανοιχτού λογισμικού. Υποστηρίζει πολλές βάσεις δεδομένων όπως η Oracle, η PostgreSQL και η MySQL η οποία χρησιμοποιήθηκε για τη συγκεκριμένη διπλωματική εργασία. Τα αρχεία php έχουν την δυνατότητα να περιλαμβάνουν κείμενο, ετικέτες html και εκτελέσιμο κώδικα. Επιστρέφουν στον φυλλομετρητή (web server), από τον οποίο ερμηνεύονται, σαν απλό HTML.

Η χρήση της PHP ποικίλει ανάλογα με τον τρόπο με τον οποίο θέλουμε να τη χρησιμοποιήσουμε. Μπορεί να χρησιμοποιηθεί για τη συλλογή δεδομένων μέσω ειδικής φόρμας που συμπληρώνεται από τον χρήστη και με την μέθοδο POST μπορεί να σταλεί στον εξυπηρετητή server και να διαχειριστεί ανάλογα. Μπορεί να χρησιμεύσει στην αποστολή και λήψη των cookies (τμήματα κώδικα που αφορούν τις προτιμήσεις και άλλες ενέργειες του χρήστη), μπορεί να τροποποιήσει δεδομένα σε βάσεις δεδομένων, για παράδειγμα: 1) προσθήκη, 2) διαγραφή, 3) αλλαγή τιμής, 4) άδειασμα ολόκληρου πίνακα στην βάση δεδομένων. Ακόμη, η PHP χρησιμοποιείται για τον περιορισμό των χρηστών σε σελίδες ανάλογα με τους κωδικούς που έχουν εισάγει στις απαραίτητες φόρμες για τη σύνδεσή τους με τον εξυπηρετητή, για την κρυπτογράφηση και αποστολή δεδομένων σε εξυπηρετητές καθώς και για προγραμματισμό αυτόματων διαδικασιών στον εξυπηρετητή.

Την παρούσα στιγμή, κατά την οποία συγγράφεται αυτή η διπλωματική εργασία η γλώσσα PHP βρίσκεται στην έκδοση 7.3.0 [8]. Είναι συμβατή με τους πιο συχνά χρησιμοποιούμενους διακομιστές παγκόσμιου ιστού όπως ο ISS της Microsoft και ο Apache. Μια σελίδα PHP υπόκειται σε επεξεργασία μέσω του συγκεκριμένου διακομιστή, ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο.

Για την ορθή γραφή του κώδικα σε γλώσσα PHP είναι απαραίτητες οι βασικές μεταβλητές. Η μεταβλητή στην PHP, είναι μία θέση προσωρινής μνήμης στην οποία αποθηκεύονται πληροφορίες οι οποίες διαχειρίζονται από τον εξυπηρετητή και μπορούν να αλλάξουν ριζικά τα παραγόμενα αποτελέσματα. Ακολουθεί μία ανάλυση των βασικών μεταβλητών.

1. **\$GLOBALS:** Μια καθολική μεταβλητή που χρησιμοποιείται για πρόσβαση σε καθολικές μεταβλητές από οπουδήποτε στο PHP script. Η PHP αποθηκεύει όλες τις μεταβλητές αυτού του τύπου σε έναν πίνακα που ονομάζεται \$GLOBALS[index]. Ο δείκτης index διατηρεί το όνομα της εκάστοτε μεταβλητής.
2. **\$_SERVER:** Μια καθολική μεταβλητή που περιέχει πληροφορίες σχετικά με τις κεφαλίδες, τις διαδρομές των φακέλων που αφορούν τη συγκεκριμένη ιστοσελίδα και τις τοποθεσίες κάθε script.
3. **\$_REQUEST:** Μια καθολική μεταβλητή που περιέχει τις τιμές \$_GET, \$_POST και \$_COOKIE. Χρησιμοποιείται για τη συλλογή των δεδομένων μετά την υποβολή μιας φόρμας HTML. Όταν ένας χρήστης υποβάλλει τα στοιχεία κάνοντας κλικ στο αντίστοιχο κουμπί της φόρμας, τα δεδομένα αυτής στέλνονται στο αρχείο που ορίζεται στην ετικέτα html <form>. Ο τρόπος με τον οποίο χρησιμοποιείται, χρειάζεται ιδιαίτερη προσοχή, λόγω της ευαισθησίας των δεδομένων.
4. **\$_GET:** Μια προκαθορισμένη μεταβλητή που χρησιμοποιείται ευρέως για τη συλλογή δεδομένων σε μια φόρμα html (με τη method="GET"). Επίσης μπορεί να συλλέξει τα δεδομένα που αποστέλλονται στη διεύθυνση URL. Οι πληροφορίες

που αποστέλλονται από μια φόρμα με τη μέθοδο GET, είναι αόρατες σε όλους, και εμφανίζονται στη γραμμή διευθύνσεων του φυλλομετρητή, επίσης υπάρχει όριο στην ποσότητα των πληροφοριών προς αποστολή με αυτήν τη μέθοδο.

5. **\$_POST:** Μια προκαθορισμένη μεταβλητή που χρησιμοποιείται ευρέως για τη συλλογή δεδομένων σε μια φόρμα html (με τη method="POST"). Οι πληροφορίες που αποστέλλονται από μια φόρμα με τη μέθοδο POST, είναι αόρατες στους άλλους και δεν υπάρχουν όρια για τον όγκο των πληροφοριών κατά την αποστολή.
6. **\$_FILES:** Μια καθολική μεταβλητή που χρησιμοποιείται για να γίνει μεταφόρτωση αρχείων στο διακομιστή (server).
7. **\$_COOKIE:** Ένα μικρό αρχείο που ενσωματώνει ο διακομιστής (server) στον υπολογιστή του χρήστη. Για να οριστεί ένα cookie χρησιμοποιείται η συνάρτηση setcookie().
8. **\$_SESSION:** Μια μεταβλητή συνεδρίας που χρησιμοποιείται για την αποθήκευση πληροφοριών, ή για την αλλαγή των ρυθμίσεων, για μια συνεδρία χρήστη. Οι μεταβλητές συνεδρίας είναι διαθέσιμες σε όλες τις σελίδες σε μια εφαρμογή. Κατά την ενασχόληση με μια εφαρμογή, μπορούν να εκτελεστούν ενέργειες και αλλαγές δεδομένων. Η κατάσταση αυτή είναι μια συνεδρία. Μια συνεδρία PHP, επιτρέπει την αποθήκευση πληροφοριών του χρήστη στο διακομιστή, για μελλοντική χρήση (δηλαδή όνομα, στοιχεία αγορών, κλπ). Ωστόσο, οι πληροφορίες συνεδρίας, είναι προσωρινές και θα διαγράφονται, αφού ο χρήστης έχει αποχωρήσει από το διαδικτυακό τόπο. Για μόνιμη αποθήκευση, μπορεί να χρησιμοποιηθεί μια βάση δεδομένων [9].

Σύνταξη της PHP

Για να ξεκινήσει ένας κώδικας της PHP είναι απαραίτητη η χρήση των ετικετών <?php στην εισαγωγή και ?> και στο τέλος του κώδικα. Μέσα σε αυτές τις ετικέτες περιλαμβάνεται όλος ο κώδικας. Κάθε εντολή της PHP πρέπει να τελειώνει με το σύμβολο ";" το οποίο καλείται διαχωριστής εντολών και υποδηλώνει το τέλος μίας εντολής. Μέσα σε ένα αρχείο php μπορεί να ανοίξει και να κλείσει όσες φορές είναι απαραίτητο κώδικας. Ακόμη, σε περίπτωση που τμήματα κώδικα πρέπει να επαναληφθούν πολλές φορές μέσα στο ίδιο αρχείο, μπορεί να γίνει ξεχωριστή συγγραφή ενός php αρχείου που περιέχει τον επαναλαμβανόμενο κώδικα και έπειτα, στα υπόλοιπα αρχεία να εκτελεστεί η συνάρτηση require_once και έπειτα να μπει ανάμεσα σε μονά εισαγωγικά το μονοπάτι στο οποίο βρίσκεται το αρχείο με τον επαναλαμβανόμενο κώδικα. Στην παρακάτω εικόνα μπορεί να διαπιστωθεί ο τρόπος γραφής της συνάρτησης require_once.

```

1 <?php
2     |require_once '../user/config.php';
3
4
5     ///////////////////////////////////////////////////////////////////
6
7
8     ///////////////////////////////////////////////////////////////////
9
10
11    ///////////////////////////////////////////////////////////////////
12
13    ///////////////////////////////////////////////////////////////////
14    //TODAY
15
16    $result = mysqli_query($link, "SELECT max(temperature) AS OMAXT FROM
17    $row = mysqli_fetch_array($result);
18    $OMAXT = $row['OMAXT'];
19

```

Σχήμα 2.9 Τρόπος σύνταξης της συνάρτησης require_once στην PHP

Ο τρόπος σύνταξης είναι πολύ απλός και κατανοητός σε έναν αρχάριο χρήστη. Παρακάτω δίνεται μία εικόνα ενός τμήματος κώδικα σε για ένα αρχείο PHP.

```

1  <!DOCTYPE html>
2 ▼ <html>
3 ▼ <body>
4
5  <?php
6  echo "My first PHP script!";
7  ?>
8
9  </body>
10 </html>

```

Σχήμα 2.10 Τρόπος σύνταξης της PHP

Στην εικόνα διακρίνουμε πώς η PHP γράφεται ανάμεσα στα στοιχεία `<body>` `</body>` της HTML. Στην PHP δεν είναι απαραίτητη η δήλωση των μεταβλητών πριν την ανάθεση μιας τιμής σε αυτές, όπως επίσης και η δήλωση του τύπου αυτών. Η PHP αυτόματα αποφασίζει τον τύπο ανάλογα με την τιμή που έχει τεθεί στην εκάστοτε μεταβλητή. Η εντολή `echo` είναι υπεύθυνη για την εκτύπωση του συνόλου των χαρακτήρων που βρίσκονται μέσα στα εισαγωγικά. Γενικά στην γλώσσα PHP ισχύουν οι τελεστές και οι εντολές εκτέλεσης υπό συνθήκη που υπάρχουν και σε άλλες γλώσσες.

2.1.4. Γλώσσα Scripting - JavaScript

Η γλώσσα scripting (σεναρίων) που χρησιμοποιήθηκε στην παρούσα διπλωματική εργασία είναι η JavaScript. Είναι μία δυναμική γλώσσα προγραμματισμού ηλεκτρονικών υπολογιστών και η πιο δημοφιλής γλώσσα scripting ανά τον κόσμο. 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. Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δεν σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστρεφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού. Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων — τέτοια παραδείγματα είναι τα έγγραφα PDF, οι εξειδικευμένοι φυλλομετρητές (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets). Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js) έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side). Το πρότυπο της γλώσσας κατά τον οργανισμό τυποποίησης ECMA ονομάζεται ECMAScript [10].

Χρησιμοποιείται από εκατομμύρια ιστοσελίδες και προσθέτει χρηστικότητα, επικύρωση φορμών και την επικοινωνία με διακομιστές. Γίνεται ευρεία χρήση της ακόμα και για δημιουργία desktop εφαρμογών αλλά και για εφαρμογές κινητής τηλεφωνίας. Υποστηρίζεται από όλους τους σύγχρονους φυλλομετρητές όπως o Chrome, o Safari και o Firefox. Γνώστες αλλά και χρήσιμες εφαρμογές οι οποίες έχουν δημιουργηθεί με την γλώσσα προγραμματισμού JavaScript είναι το Slack (εφαρμογή η οποία έχει βοηθήσει πολύ στην οργάνωση και επικοινωνία των εργαζομένων – ομάδων των επιχειρήσεων) και το WhatsApp (εφαρμογή μέσα από την οποία οι άνθρωποι μπορούν να επικοινωνήσουν μεταξύ τους, με μηνύματα κειμένου, εικόνας αλλά και να αποστείλουν αρχεία). Και οι δύο αυτές εφαρμογές βασίστηκαν στο framework (βιβλιοθήκη) Electron το οποίο είναι γραμμένο για την JavaScript [11].

Η υλοποίησης της JavaScript είναι client-side (υπολογιστή - πελάτη). Με το χαρακτηριστικό αυτό προσδίδεται ασφάλεια για τα δεδομένα που εισάγονται, καθώς γίνεται έλεγχος για την εγκυρότητά τους πριν αυτά αποσταλούν στο διακομιστή (server). Έτσι επιτυγχάνεται η αποφυγή κακόβουλων επιθέσεων από τους χρήστες.

Για να προσθέσουμε λειτουργικότητα στην σελίδα μας χρησιμοποιούμε την JavaScript. Η γλώσσα αυτή λειτουργεί με παρόμοια λογική όπως και η CSS. Όπως η CSS χρησιμοποιεί

τα στοιχεία <link> και <style>, η JavaScript χρειάζεται μόνο ένα στοιχείο, το <script>. Για να ξεκινήσουμε να γράφουμε κώδικα σε JavaScript απλά ανοίγουμε τον επεξεργαστή κειμένου μας και μέσα στο στοιχείο <head> ανοίγουμε το στοιχείο <script> και είμαστε έτοιμοι να γράψουμε τον κώδικα μας, όπως φαίνεται στο παρακάτω παράδειγμα.

```
1 | <script>
2 |
3 |   // JavaScript goes here
4 |
5 | </script>
```

Σχήμα 2.11 Τρόπος σύνταξης της JavaScript

Ένας άλλος τρόπος για να προσθέσουμε κώδικα JavaScript στην σελίδα είναι με την σύνδεση ενός εξωτερικού αρχείου. Στην προκειμένη περίπτωση η σύνδεση γίνεται όπως φαίνεται παρακάτω.

```
1 | <script src="script.js" defer></script>
```

Σχήμα 2.12 Σύνδεση αρχείου JavaScript σε αρχείον HTML/PHP

Τα εξωτερικά αυτά αρχεία έχουν την κατάληξη .js και δεν πρέπει να περιέχουν τις ετικέτες <script>...</script>. Για τη συγγραφή της παρούσας διπλωματικής εργασίας, δημιουργήθηκαν αρκετά αρχεία σε κώδικα JavaScript καθώς υπάρχει μεγάλη παραμετροποίηση στον τρόπο αλλά και στο είδος εμφάνισης των δεδομένων προς τον χρήστη.

```

$(function() {
    "use strict";
    $(function() {
        $(".preloader").fadeOut();
    });

    // =====
    // This is for the top header part and sidebar part
    // =====
    var set = function() {
        var width = (window.innerWidth > 0) ? window.innerWidth : this.screen.width;
        var topOffset = 70;
        if (width < 1170) {
            $("body").addClass("mini-sidebar");
            $('.navbar-brand span').hide();
            $(".scroll-sidebar, .slimScrollDiv").css("overflow-x", "visible").parent().css("overflow", "hidden");
            $(".sidbartoggler i").addClass("ti-menu");
        } else {
            $("body").removeClass("mini-sidebar");
            $('.navbar-brand span').show();
            //$(".sidbartoggler i").removeClass("ti-menu");
        }
    }

    var height = ((window.innerHeight > 0) ? window.innerHeight : this.screen.height) - 1;
    height = height - topOffset;
    if (height < 1) height = 1;
    if (height > topOffset) {
        $(".page-wrapper").css("min-height", (height) + "px");
    }
};

$(window).ready(set);
$(window).on("resize", set);

```

Σχήμα 2.13 Τμήμα κώδικα σε JavaScript για τη λειτουργικότητα των μενού της διαδικτυακής εφαρμογής Smart Homie

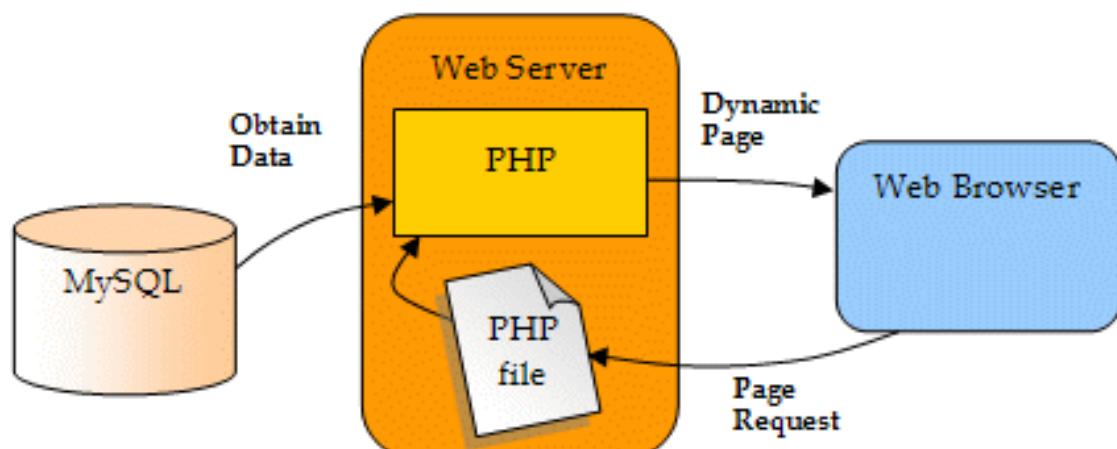
2.1.5. Γλώσσα Σχεσιακών Βάσεων Δεδομένων - MySQL

Η MySQL, ακρολέξιο του (My Structured Query Language), είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων που μετρά περισσότερες από 11 εκατομμύρια εγκαταστάσεις. Έλαβε το όνομά της από την κόρη του Μόντυ Βιντένιους. Το πρόγραμμα τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων [12]. Η MySQL είναι η πιο δημοφιλής βάση δεδομένων ανοιχτού κώδικα παγκοσμίως, επιτρέποντας οικονομικά αποδοτική παράδοση αξιόπιστων, υψηλής απόδοσης και κλιμακούμενων εφαρμογών και ενσωματωμένων βάσεων δεδομένων. Πρόκειται για μια ολοκληρωμένη βάση δεδομένων, συμβατή με το ACID, με πλήρη δέσμευση, επαναφορά, αποκατάσταση σφαλμάτων και δυνατότητες κλειδώματος σε επίπεδο σειράς. Στην πληροφορική, το ACID (ατομικότητα, συνέπεια, απομόνωση, μονιμότητα) είναι ένα σύνολο ιδιοτήτων το οποίο εγγυάται ότι οι συναλλαγές στη βάση δεδομένων (ΒΔ) λειτουργούν αξιόπιστα. Η ιδέα του ACID είναι να αξιολογεί τη δομή βάσεων δεδομένων και εφαρμογών. Όσον αφορά τις βάσεις δεδομένων μια λογική ενέργεια που αφορά τα δεδομένα της αποκαλείται συναλλαγή [13]. Η MySQL παρέχει την ευκολία χρήσης, την επεκτασιμότητα και την υψηλή απόδοση, καθώς και ένα πλήρες

πακέτο προγραμμάτων οδήγησης βάσεων δεδομένων και οπτικά εργαλεία για να βοηθούν τους προγραμματιστές και τα DBA να δημιουργούν και να διαχειρίζονται τις κρίσιμες για την επιχείρησή τους εφαρμογές με την βοήθεια της MySQL. Αναπτύσσεται, διανέμεται και υποστηρίζεται από την Oracle. Παρέχει της ακόλουθες δυνατότητες:

- Υψηλή απόδοση και δυνατότητα κλιμάκωσης για την κάλυψη των απαιτήσεων των εκθετικά αυξανόμενων φορτίων δεδομένων και των χρηστών
- Αυτο-θεραπευόμενες ομάδες ανάπλασης δεδομένων για τη βελτίωση της επεκτασιμότητας, της απόδοσης και της διαθεσιμότητας
- SQL και NoSQL πρόσβαση για την εκτέλεση πολύπλοκων ερωτημάτων και απλών, γρήγορων λειτουργιών για τιμές - κλειδιά
- Ανεξαρτησία πλατφόρμας που δίνει την ευελιξία ανάπτυξης σε πολλαπλά λειτουργικά συστήματα.
- Διαλειτουργικότητα μεγάλου όγκου δεδομένων μέσω της MySQL [14].

Τα δεδομένα στη MySQL αποθηκεύονται σε αντικείμενα βάσης δεδομένων τα οποία ονομάζονται πίνακες. Ένας πίνακας είναι μια συλλογή από σχετικές καταχωρήσεις δεδομένων και αποτελείται από γραμμές και στήλες. Για την αποθήκευση πληροφοριών σε κατηγορίες οι βάσεις δεδομένων είναι πολύ χρήσιμες.



Σχήμα 2.14 Λειτουργία και αλληλεπίδραση της MySQL

Μέσω των SQL εντολών πραγματοποιείται η διαχείριση των δεδομένων στις βάσεις δεδομένων. Η πιο σημαντική ίσως κατηγορία εντολών που χρησιμοποιείται είναι τα queries. Ένα query είναι ένα ερώτημα ή ένα αίτημα. Με τη MySQL, επιτρέπεται η διερεύνηση σε μια βάση δεδομένων και η επιστροφή ζητηθέντων πληροφοριών. Παρακάτω παρουσιάζονται οι βασικότερες εντολές SQL με τη σημασία τους.

1. SELECT: Εξάγει δεδομένα από μια βάση δεδομένων
2. UPDATE: Ενημερώνει δεδομένα σε μια βάση δεδομένων
3. DELETE: Διαγράφει δεδομένα από μια βάση δεδομένων
4. INSERT INTO: Εισάγει δεδομένα σε μια βάση δεδομένων
5. CREATE DATABASE: Δημιουργεί μια βάση δεδομένων
6. ALTER DATABASE: Τροποποιεί μια βάση δεδομένων
7. CREATE TABLE: Δημιουργεί έναν νέο πίνακα
8. ALTER TABLE: Τροποποιεί έναν πίνακα
9. DROP TABLE: Διαγράφει έναν πίνακα
10. CREATE INDEX: Δημιουργεί έναν δείκτη (κλειδί αναζήτησης)
11. DROP INDEX: Διαγράφει έναν δείκτη

2.2 Υλικός Εξοπλισμός - Hardware

Για την παρούσα διπλωματική εργασία, είναι απαραίτητος ο υλικός εξοπλισμός (Hardware) διότι χωρίς αυτόν είναι αδύνατη η συλλογή δεδομένων, η οπτική απεικόνιση στο χρήστη καθώς και η δυνατότητα που δίνεται σε αυτόν να ελέγχει τις έξυπνες συσκευές του. Παρακάτω γίνεται μία αναφορά στον εξοπλισμό που χρησιμοποιήθηκε και πώς το κάθε εξάρτημα συμπληρώνει το εγχείρημα που πραγματοποιήθηκε. Γίνεται εκτενής αναφορά στις δυνατότητες καθώς και στην επεκτασιμότητα που δέχεται το κάθε υλικό στοιχείο.

2.2.1. Μικροελεγκτής Arduino Uno Rev3

Iστορία του Arduino

Το 2005, με σκοπό την ευκολότερη εκμάθηση των ηλεκτρονικών για τους μαθητές, ο καθηγητής Massimo Banzi στο Ινστιτούτο Σχεδιασμού Αλληλεπίδρασης στην Ivrea της Ιταλίας, επιχείρησε τον σχεδιασμό μιας νέας συσκευής για τον έλεγχο προγραμμάτων και διαδραστικών σχεδίων από τους ίδιους τους μαθητές, με χαμηλότερο κόστος. Μαζί με έναν μηχανικό από το Πανεπιστήμιο Malmo, τον David Cuartielles, αποφάσισαν να φτιάξουν ένα μικροελεγκτή, ο οποίος ήταν πιο προσιτός ως προς τη χρήση του.

Ονόμασαν το έργο τους Arduino of Ivrea «Arduino». Το πρώτο μοντέλο περιλάμβανε ATmega8 με άμεση σύνδεση RS-232 με το μικροελεγκτή και όλα τα στοιχεία του και ονομάστηκε «Serial Arduino» [15].

Πληροφορίες για το Arduino

Το Arduino βασίζεται στην αρχή του ελεύθερου λογισμικού και υλικού. Είναι ουσιαστικά ένας μικροελεγκτής μέσω του οποίου μπορεί να πραγματοποιηθεί έλεγχος και τροποποίηση παραμέτρων σε φυσικές συσκευές. Το περιβάλλον ανάπτυξης του λογισμικού είναι βασισμένο στις γλώσσες προγραμματισμού Processing και Wiring, η οποία είναι στην ουσία η C++ και ένα σύνολο από βιβλιοθήκες υλοποιημένες στην C++. Τόσο η Wiring όσο και η Processing είναι ανοιχτού κώδικα (open source) και μπορεί κάποιος να τις “κατεβάσει δωρεάν”. Το Arduino χρησιμοποιεί έναν ειδικά προγραμματισμένο ATmega328P αντί του FTDI chip ώστε να επιτρέπεται τόσο η πιο γρήγορη ταχύτητα μεταφοράς όσο και η γρήγορη σειριακή επικοινωνία.

Πλεονεκτήματα επιλογής Arduino

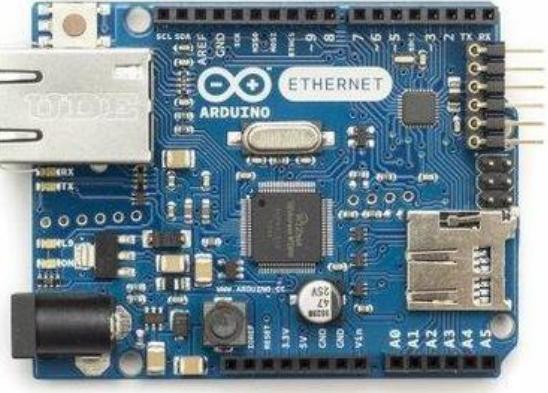
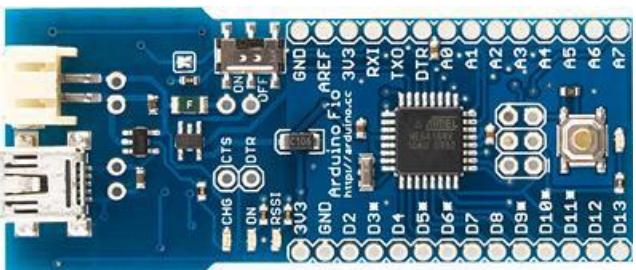
- **Χαμηλό κόστος:** Οι πλακέτες του Arduino είναι εξαιρετικά οικονομικές σε σύγκριση με άλλες πλατφόρμες μικροελεγκτών. Με τις σχηματικές αναπαραστάσεις που κυκλοφορούν στο Διαδίκτυο δεν μπορεί να κατασκευάσει κάποιος την φθηνότερη εκδοχή ενός Arduino. Ωστόσο αν προμηθευτεί την έτοιμη (μονταρισμένη πλακέτα) αυτή θα κοστίσει το μέγιστο 50 €. Το κόστος διαμορφώνεται ανάλογα με την έκδοση.
- **Πληθώρα Λειτουργικών Συστημάτων:** Οι μηχανικοί λογισμικού, ανέπτυξαν το περιβάλλον προγραμματισμού του Arduino όχι μόνο για Windows αλλά και Macintosh OSX καθώς και λειτουργικά συστήματα Linux.
- **Απλό και ξεκάθαρο περιβάλλον προγραμματισμού:** Το περιβάλλον προγραμματισμού του Arduino είναι ευέλικτο για προχωρημένους χρήστες αλλά ταυτόχρονα ενδείκνυται και για αρχάριους.
- **Ανοιχτού λογισμικού που επεκτείνεται και παραμετροποιείται:** Το Software του Arduino διανέμεται με τη μορφή εργαλείων ανοιχτού λογισμικού και μπορεί να επεκταθεί από έμπειρους προγραμματιστές. Η επέκταση της γλώσσας προγραμματισμού του γίνεται μέσω των βιβλιοθηκών της C++ και οι άνθρωποι που ενδιαφέρονται για περαιτέρω ενασχόληση με τους μικροελεγκτές μπορούν να μεταβούν από το Arduino στην AVR C, η γλώσσα στην οποία βασίστηκε το λογισμικό του Arduino, για προγραμματισμό των μικροελεγκτών Atmel. Ομοίως μπορεί κάποιος να προσθέσει κώδικα της AVR C στο πρόγραμμα που έχει γράψει για το Arduino του.

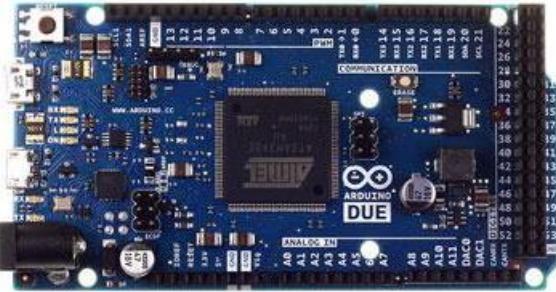
- **Ανοιχτού υλικού το οποίο μπορεί να επεκταθεί:** Το Arduino βασίζεται στους μικροελεγκτές της Atmel. Οι σχηματικές αναπαραστάσεις για τα αναπτυξιακά είναι υπό την άδεια της Creative Commons, επιτρέποντας την κατασκευή αναπτυξιακού σε έμπειρους σχεδιαστές, εξελίσσοντας το ήδη υπάρχον χωρίς να δημιουργηθεί κάποιο νομικό πρόβλημα. Ακόμα, επιτρέπεται στους όχι και τόσο έμπειρους χρήστες να επιδιώξουν να αντιγράψουν και να κατασκευάσουν την πλακέτα για να καταλάβουν τη λειτουργία ενός Arduino [16].

Διάφορες πλακέτες Arduino

Στην παρούσα διπλωματική εργασία για λόγους ευκολίας αλλά και μεγάλης παραμετροποίησης σύμφωνα με τις ανάγκες της διαδικτυακής εφαρμογής επιλέχθηκε το arduino uno. Αυτό δεν σημαίνει πώς δεν υπάρχουν και άλλες πλακέτες arduino που είναι εξίσου αξιόλογες ανάλογα με την χρήση στην οποία υπόκεινται. Στον παρακάτω πίνακα παρατίθενται διάφορες ακόμα πλακέτες arduino καθώς και τα τεχνικά χαρακτηριστικά τους [31].

Πίνακας 2.2: Είδη Arduino

Πλακέτα Arduino	Τεχνικά Χαρακτηριστικά
 <p>Arduino Uno Rev3</p>	<ul style="list-style-type: none"> Μικροελεγκτής : ATMega328P Τάση λειτουργίας: 5V Ψηφιακοί είσοδοι/έξοδοι: 14(6 παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 6 Μνήμη Flash: 32KB (ATMega328P) SRAM: 2KB (ATMega328P) EEPROM: 1KB (ATMega328P) Ταχύτητα ρολογιού: 16MHz
 <p>Arduino Ethernet</p>	<ul style="list-style-type: none"> Μικροελεγκτής: ATMega328 Τάση λειτουργίας: 5V Ψηφιακοί είσοδοι/έξοδοι: 14(64παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 6 Μνήμη Flash: 32KB (ATMega328P) SRAM: 2KB (ATMega328P) EEPROM: 1KB (ATMega328P) Ταχύτητα ρολογιού: 16MHz W5100 TCP/IP Embedded Ethernet Controller
 <p>Arduino Fio</p>	<ul style="list-style-type: none"> Μικροελεγκτής: ATMega328P Τάση λειτουργίας: 3.3V Ψηφιακοί είσοδοι/έξοδοι: 14(6 παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 8 Μνήμη Flash: 32KB SRAM: 2KB EEPROM: 1KB Ταχύτητα

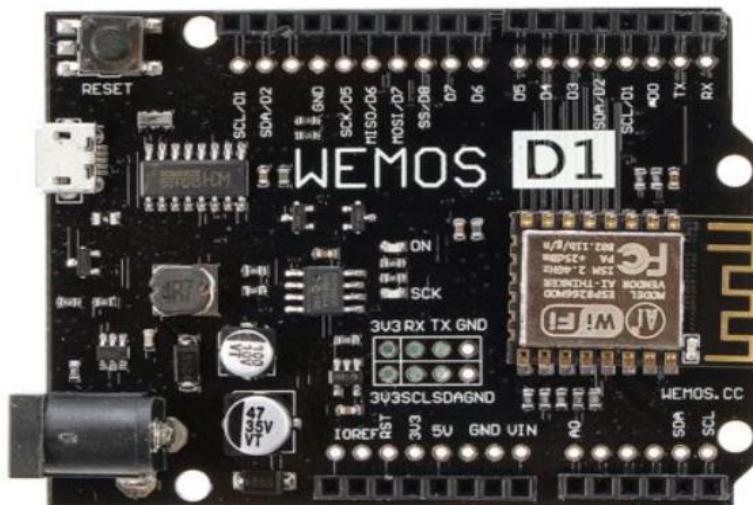
	ρολογιού: 8MHz
 <p>Arduino Mega ADK</p>	<ul style="list-style-type: none"> Μικροελεγκτής: ATMega2560 Τάση λειτουργίας: 5V Ψηφιακοί είσοδοι/έξοδοι: 54(6 παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 16 Μνήμη Flash: 256KB SRAM: 8KB EEPROM: 4KB Ταχύτητα ρολογιού: 16MHz
 <p>Arduino Due</p>	<ul style="list-style-type: none"> Μικροελεγκτής: AT91SAM3X8E Τάση λειτουργίας: 3.3V Συνολικό ρεύμα Dc εξόδου: 130mA Ψηφιακοί είσοδοι/έξοδοι: 54(12 παρέχουν PWM έξοδο) Αναλογικές είσοδοι: 12 Αναλογικές έξοδοι: 2 (DAC) Μνήμα Flash: 512KB SRAM: 96KB (64KB και 32KB) Ταχύτητα ρολογιού: 84MHz
 <p>Arduino Leonardo</p>	<ul style="list-style-type: none"> Μικροελεγκτής: ATMega32U4 Τάση λειτουργίας: 5V Ψηφιακοί είσοδοι/έξοδοι: 20(7 παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 12 Μνήμη Flash: 32KB (ATMega32U4) SRAM: 2KB (ATMega32U4) EEPROM: 1KB (ATMega32U4) Ταχύτητα ρολογιού: 16MHz

 <p>Arduino Yun</p>	<ul style="list-style-type: none"> Μικροελεγκτής: ATMega32U4 Τάση λειτουργίας: 5V Ψηφιακοί είσοδοι/έξοδοι: 20(7 παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 12 Μνήμη Flash: 32K SRAM: 2.5KB EEPROM: 1KB Ταχύτητα ρολογιού: 16MHz Μικροεπεξεργαστής: Linux Processorv Arheros AR9331 Αρχιτεκτονική: MIPS @400MHz Τάση λειτουργίας: 3.3V Ethernet: IEEE 802.3 10/100Mbit/s Wi-Fi: IEE 802.11b/g/n Card Reader: 64 MB DDR2 Μνήμη Flash: 16MB
 <p>Arduino Micro</p>	<ul style="list-style-type: none"> Μικροελεγκτής: ATMega32U4 Τάση λειτουργίας: 5V Ψηφιακοί είσοδοι/έξοδοι: 20(7 παρέχουν PWM έξοδο) Ρεύμα Dc ανά είσοδο/ έξοδο: 40mA Αναλογικές είσοδοι: 12 Μνήμη Flash: 32KB (ATMega32U4) SRAM: 2.5KB (ATMega32U4) EEPROM: 1KB (ATMega32U4) Ταχύτητα ρολογιού: 16MHz

2.2.2. Μικροελεγκτής WeMos D1 R2

Πληροφορίες για το WeMos D1

Ο μικροελεγκτής WeMos D1 είναι ένας μικροελεγκτής ο οποίος είναι συμβατός με το arduino uno, και είναι μία πλακέτα WiFi βασισμένη στο ESP8266EX. Διαθέτει 11 ψηφιακά pins εισόδου/εξόδου, όλα τα pins υποστηρίζουν τις λειτουργίες interrupt/pwm/I2C/one-wire εκτός από pin D0. Διαθέτει μία αναλογική είσοδο όπου η μέγιστη τάση εισόδου ανέρχεται στα 3.2V. Η σύνδεση και ο προγραμματισμός του γίνεται μέσω μίας micro USB θύρας, διαθέτει ένα βύσμα ρεύματος τύπου barrel jack, το οποίο επιτρέπει στο WeMos D1 να περάσει τάση εισόδου από 9 έως 24 volt. Είναι συμβατό με διάφορες πλακέτες arduino και nodemcu.



Σχήμα 2.15 WeMos D1 R2

Τεχνικά χαρακτηριστικά του WeMos D1

Στο παρακάτω πίνακα εμφανίζονται τα τεχνικά χαρακτηριστικά για τον μικροελεγκτή WeMos D1 R2 που χρησιμοποιήθηκε για την παρούσα διπλωματική εργασία.

Πίνακας 2.3: Τεχνικά Χαρακτηριστικά WeMos D1 R2

Μικροελεγκτής	ESP-8266EX
Τάση Λειτουργίας	3.3V
Ψηφιακά Pins I/O	11
Αναλογικά pins εισόδου	1 (Μέγιστη τάση εισόδου: 3.2V)
Χρονισμός πυρήνα	80Mhz / 160Mhz
Μνήμη Flash	4MBytes
Μήκος	68.6mm
Πλάτος	53.4mm
Βάρος	25g

Pins – Λειτουργίες του WeMos D1

Κάθε pin στον μικροελεγκτή WeMos έχει σχεδιαστεί έτσι ώστε να συμφωνεί με τους μικροελεγκτές με τους οποίους έχει επίσημη συμβατότητα. Για το λόγο αυτό παρατίθεται ο πίνακας με τα pins καθώς και τις λειτουργίες του μικροελεγκτή.

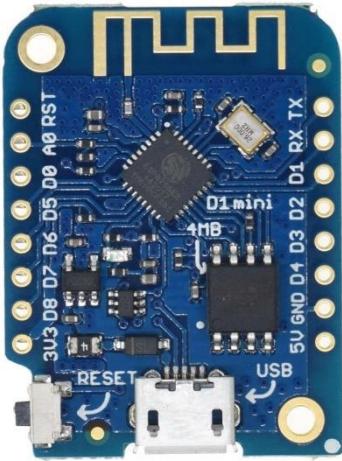
Πίνακας 2.4: Σύνδεση - Λειτουργία των pins του WeMos D1 R2

Pin	Function	ESP-8266 Pin
TX	TXD	TXD
RX	RXD	RXD
A0	Analog input, max 3.3V input	A0
D0	IO	GPIO16
D1	IO, SCL	GPIO5
D2	IO, SDA	GPIO4
D3	IO, 10k Pull-up	GPIO0
D4	IO, 10k Pull-up, BUILTIN_LED	GPIO2
D5	IO, SCK	GPIO14
D6	IO, MISO	GPIO12
D7	IO, MOSI	GPIO13
D8	IO, 10k Pull-down, SS	GPIO15
G	Ground	GND
5V	5V	-
3V3	3.3V	3.3V
RST	Reset	RST

Διάφορες Πλακέτες WeMos D1

Η WeMos εκτός από τον μικροελεγκτή WeMos D1 R2 διαθέτει και άλλες πλακέτες WeMos D1. Ανάλογα με την χρήση, τη σύνδεση και τον χώρο που έχουμε στην διάθεση μας, και ανάλογα με την περίπτωση που υλοποιούμε, μπορούμε να επιλέξουμε διαφορετικό μικροελεγκτή. Στον επόμενο πίνακα επεξηγούνται οι υπόλοιπες πλακέτες WeMos D1[17].

Πίνακας 2.5: Σύνδεση - Πλακέτες WeMos D1

Πλακέτα	Τεχνικά Χαρακτηριστικά
 WeMos D1 Mini Pro	<ul style="list-style-type: none"> Μικροελεγκτής: ESP-8266EX Τάση Λειτουργίας: 3.3V Ψηφιακά Pins I/O: 11 Αναλογικά pins εισόδου: 1 (Μέγιστη τάση εισόδου: 3.2V) Χρονισμόςπυρήνα: 80Mhz/160Mhz Μνήμη Flash: 16MBytes Μήκος: 48mm Πλάτος: 25.4mm Βάρος 4.5g
 WeMos D1 Mini	<ul style="list-style-type: none"> Μικροελεγκτής: ESP-8266EX Τάση Λειτουργίας: 3.3V Ψηφιακά Pins I/O: 11 Αναλογικά pins εισόδου: 1 (Μέγιστη τάση εισόδου: 3.2V) Χρονισμόςπυρήνα: 80Mhz/160Mhz Μνήμη Flash: 4MBytes Μήκος: 34.2mm Πλάτος: 25.6mm Βάρος 3g



WeMos D1 Mini Lite

- Μικροελεγκτής: ESP-8266EX
- Τάση Λειτουργίας: 3.3V
- Ψηφιακά Pins I/O: 11
- Αναλογικά pins εισόδου: 1 (Μέγιστη τάση εισόδου: 3.2V)
- Χρονισμός πυρήνα: 80Mhz/160Mhz
- Μνήμη Flash: 1MBytes
- Μήκος: 34.2mm
- Πλάτος: 25.6mm
- Βάρος 10g

2.2.3. Δρομολογητής - Router

Για την εκπόνηση της παρούσας διπλωματικής εργασίας καθώς και για την ανάπτυξη της διαδικτυακής εφαρμογής ήταν αναγκαία η τοποθέτηση ενός δρομολογητή (router) στη σύνδεση των μικροελεγκτών με τον εξυπηρετητή. Δρομολογητής (router) είναι μια ηλεκτρονική συσκευή η οποία αναλαμβάνει την αποστολή και λήψη πακέτων δεδομένων μεταξύ ενός ή περισσοτέρων διακομιστών, άλλων δρομολογητών και πελατών, κατά μήκος πολλαπλών δικτύων (δρομολόγηση). Η δρομολόγηση, δηλαδή είναι η διαδικασία μεταφοράς δεδομένων από ένα σημείο σε ένα άλλο και αποτελεί κεντρική λειτουργία του επιπέδου δικτύου, γίνεται με βάση διάφορα κριτήρια και τελικώς επιλέγεται μία ανάμεσα σε διάφορες πιθανές διαδρομές. Κάθε δρομολογητής χρησιμοποιεί ένα ή περισσότερα πρωτόκολλα δρομολόγησης. Με βάση αυτά τα πρωτόκολλα ο δρομολογητής καθορίζει ποιος ή ποιοι διακομιστές ή δρομολογητές είναι οι καταλληλότεροι κάθε χρονική στιγμή και δρομολογεί τα πακέτα δεδομένων προς αυτούς.

Forwarding – Προώθηση

Για τις διαδικασίες προώθησης αμιγώς του Internet Protocol (IP), ο δρομολογητής (router) έχει κατασκευαστεί ώστε να ελαχιστοποιεί την κατάσταση της πληροφορίας που σχετίζεται με το κάθε πακέτο. Ο κύριος σκοπός ενός δρομολογητή είναι να συνδέει πολλαπλά δίκτυα και να προωθεί τα πακέτα που προορίζονται είτε στα δίκτυα του, είτε σε άλλα δίκτυα. Η προεπιλεγμένη (στατική) δρομολόγηση (default or static route) είναι αυτή που χρησιμοποιείται όταν η δρομολόγηση όλων των προορισμών δεν διαφοροποιείται στον πίνακα (routing table). Αυτή χρησιμοποιείται στα μικρά δίκτυα όπως είναι π.χ. τα οικιακά η τα δίκτυα μικρών επιχειρήσεων, όπου η προεπιλεγμένη (η

στατική) δρομολόγηση απλά στέλνει όλη την κίνηση στον πάροχο - Internet Service Provider (ISP). Η προεπιλεγμένη (στατική) δρομολόγηση μπορεί να ρυθμιστεί “χειροκίνητα” ως static route, η μπορεί να αποκτηθεί από το δυναμικό πρωτόκολλο δρομολόγησης (dynamic routing protocol), η από το DHCP. Ο router μπορεί να χρησιμοποιηθεί τόσο ως DHCP client, όσο και ως DHCP server [18].

Για τις ανάγκες της παρούσας διπλωματικής εργασίας χρησιμοποιήθηκε το πρωτόκολλο δρομολόγησης DHCP, με αποτέλεσμα την απόδοση στατικών IP στους μικροελεγκτές, στις έξυπνες συσκευές καθώς και στον εξυπηρετητή δημιουργώντας έτσι ένα μικρό τοπικό δίκτυο για την ταχύτερη, ασφαλέστερη και χωρίς απώλειες αποστολή δεδομένων μεταξύ των μικροελεγκτών και του εξυπηρετητή. Τον ρόλο του δρομολογητή στην παρούσα διπλωματική εργασία κατέχει ένα Thomson TG585 V7.



Σχήμα 2.16 Δρομολογητής – Μόντεμ TG585 V7

2.3 Πλατφόρμες Ανάπτυξης Λογισμικού

Για την ανάπτυξη λογισμικού, διαδικτυακών εφαρμογών καθώς και για αυτοματισμούς οι προγραμματιστές χρησιμοποιούν διάφορες πλατφόρμες πάνω στις οποίες επεξεργάζονται τον πηγαίο κώδικα ενός project, προσομοιώνουν τοπικούς διακομιστές ή χρήστες αλλά και εξετάζουν τα θέματα ασφαλείας που μπορεί να έχει ο κώδικας που έχουν αναπτύξει. Έτσι λοιπόν, οι πλατφόρμες ανάπτυξης λογισμικού είναι αναπόσπαστο μέρος της δουλειάς ενός προγραμματιστή. Αυτό δεν σημαίνει σε καμία περίπτωση πώς οι προγραμματιστές είναι αναγκασμένοι να επιλέξουν μία πλατφόρμα πάνω στην οποία θα βασίσουν το project τους και να μην χρησιμοποιήσουν οποιαδήποτε άλλη. Αντίθετα, η

πολυπλοκότητα αλλά και οι αυξημένες απαιτήσεις που έχουν οι πελάτες, οδηγούν τους προγραμματιστές στο να χρησιμοποιήσουν και μερικές φορές να ενοποιήσουν διάφορες πλατφόρμες με σκοπό την ομαλή λειτουργία του project τους.

Στα πλαίσια της παρούσας διπλωματικής εργασίας, χρησιμοποιήθηκαν 4 διαφορετικές πλατφόρμες ανάπτυξης λογισμικού. Επιλέχθηκαν ανάλογα με τα χαρακτηριστικά τους, τις δυνατότητες που έχουν καθώς και το πεδίο εφαρμογής.

2.3.1. Visual Studio Code

Το Visual Studio Code είναι ένας επεξεργαστής πηγαίου κώδικα ο οποίος αναπτύχθηκε από την Microsoft και υποστηρίζεται σε λειτουργικά συστήματα όπως είναι τα Windows, τα Linux αλλά και τα macOS. Περιλαμβάνει υποστήριξη για debugging, ενσωματωμένο έλεγχο του Git, επισήμανση συντακτικού, έξυπνη αυτοσυμπλήρωση κώδικα και αποσπάσματα κώδικα. Η δημιουργία του Visual Studio Code έχει βασιστεί στην βιβλιοθήκη Electron που έχουμε αναφέρει παραπάνω και παρουσιάστηκε πρώτη φορά από την Microsoft τον Απρίλιο του 2015. Παρέχει υποστήριξη σε πάρα πολλές γλώσσες προγραμματισμού, οι πιο γνωστές εκ των οποίων είναι οι C και C++, C#, Go, HTML, CSS, JavaScript, TypeScript, PHP, Python, R και Swift [19]. Το Visual Studio Code, αποτέλεσε τον βασικό κινητήριο μοχλό της παρούσας διπλωματικής εργασίας καθώς υποστηρίζει όλες τις γλώσσες προγραμματισμού που χρησιμοποιήθηκαν και αρκετά τμήματα από τον κώδικα αναπτύχθηκαν στο συγκεκριμένο επεξεργαστή κειμένου.

Βασικά πλεονεκτήματα που διαθέτει ο συγκεκριμένος επεξεργαστής κειμένου σε σχέση με άλλα λογισμικά είναι:

- **Διαθεσιμότητα σε γνωστά λειτουργικά:** Το Visual Studio Code υποστηρίζει λειτουργικά όπως macOS, Linux, και Windows. Καθίσταται λοιπόν μία σημαντική επιλογή για κάποιον ο οποίος θέλει να αναπτύξει εφαρμογές σε cross-platform επίπεδο.
- **Επεξεργασία, ανάπτυξη και αποσφαλμάτωση με ευκολία:** Με την ταχεία αλλά και την ογκώδη υποστήριξη πολλών γλωσσών προγραμματισμού, το Visual Studio Code δίνει την δυνατότητα στους χρήστες του να είναι πιο παραγωγικοί με τα χαρακτηριστικά που προσφέρει ανάλογα με τη γλώσσα που επεξεργάζεται εκείνη την στιγμή. Διαθέτει αρκετές συντομεύσεις μέσα από τις οποίες μπορεί να αυξηθεί σε μεγάλο βαθμό η παραγωγικότητα και να μειωθεί ο χρόνος εκτέλεσης, με αποτέλεσμα τη μείωση τους κόστους.
- **Εύκολη παραμετροποίηση:** Κάθε χρήστης – προγραμματιστής ανάλογα με την εμπειρία αλλά και τις προτιμήσεις του, είναι πιθανό να χρειάζεται ένα

διαφορετικό περιβάλλον εργασίας. Το Visual Studio Code, δίνει λύση σε αυτό το ζήτημα με τις απεριόριστες παραμετροποίησεις που επιτρέπει στον χρήστη να κάνει καθώς και με τα διάφορα πρόσθετα που δημιουργούνται καθημερινά από την κοινότητα των προγραμματιστών.

- **Εστίαση στην ανάπτυξη περιεχομένου στο Διαδίκτυο:** Επιλέγεται κυρίως ως πλατφόρμα ανάπτυξη περιεχομένου από web developers. Αυτό γίνεται επειδή η Microsoft επενδύει καθημερινά στην out-of-the-box υποστήριξη του Node.js και των γλωσσών JavaScript και TypeScript που απασχολούν την πλειοψηφία των διαδικτυακών τόπων & εφαρμογών [20].

```

devices_controller.ino - MyThesis - Visual Studio Code
File Edit Selection View Go Debug Tasks Help
EXPLORER devices_controller.ino
OPEN EDITORS
MYTHESIS
vscode
Arduino
controller
devices_controller
status_update.php
indoor_data
libraries
outdoor_data
LiveSite
css
dashboard
assets
images
plugins
charts
css
js
scss
static
head.html
lowerbody.php
upperbody.php
OUTLINE
ARDUINO EXAMPLES
24 int AC = 8;
25 int DH = 7;
26
27 //-----
28 //-----
29 void setup()
30 {
31 //-----
32
33 // Extra Set up code:
34 pinMode(AC, OUTPUT); //pin selected to control
35 pinMode(DH, OUTPUT); //pin selected to control
36
37 //-----
38 //-----
39 //enable serial data print
40 Serial.begin(9600);
41
42 //start Ethernet
43 Ethernet.begin(mac, ip, gateway, subnet);
44 server.begin();
45 Serial.print("Server is at ");
46 Serial.println(Ethernet.localIP());
47 Serial.println("LED Controller Test 1.0");
48 }
49 //-----
50 //-----
51

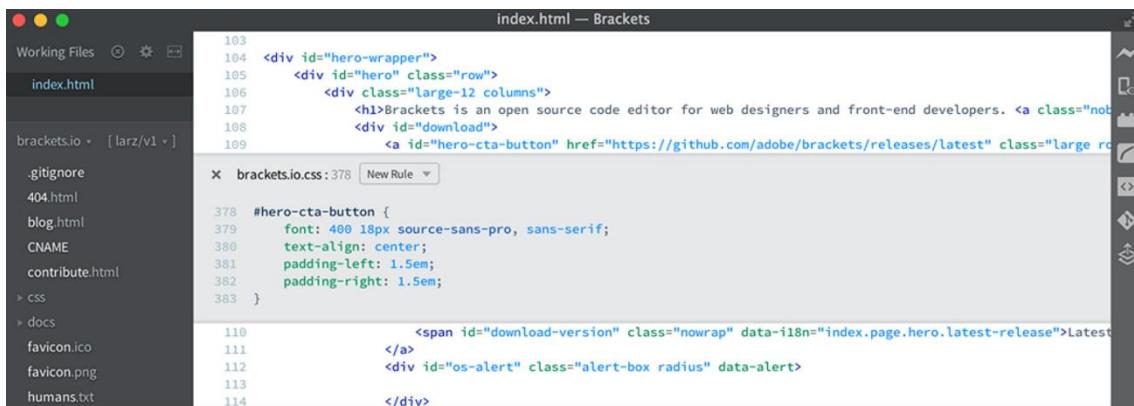
```

master* Go Live Ln 30, Col 2 Spaces: 2 UTF-8 LF Arduino <Select Programmer> <Select Board Type>

Σχήμα 2.17 Περιβάλλον Προγραμματισμού Visual Studio Code

2.3.2. Brackets

Το Brackets είναι ένα νέος, ανοικτού κώδικα επεξεργαστής κειμένου ο οποίος αντιλαμβάνεται τα ‘κενά’ στην ανάπτυξη εφαρμογών του Διαδικτύου και προσφέρει λύσεις. Την παρούσα στιγμή κατά την οποία συγγράφεται η διπλωματική εργασία το Brackets βρίσκεται στην έκδοση 1.13 και οι δυνατότητες που προσφέρει το καθιστούν αξιόλογο για την ανάπτυξη του κώδικα μια διαδικτυακής εφαρμογής επάνω σε αυτό [21]. Δημιουργήθηκε από την Adobe Systems το 2014 και μέχρι σήμερα έχει γνωρίσει αρκετή ανάπτυξη. Πλέον το Brackets υποστηρίζει όλα τα γνωστά λειτουργικά συστήματα και ο πηγαίος κώδικας του συντηρείται σε ένα open source project στο GitHub.



Σχήμα 2.18 Περιβάλλον Προγραμματισμού Brackets

Βασικά πλεονεκτήματα που διαθέτει ο συγκεκριμένος επεξεργαστής κειμένου σε σχέση με άλλα λογισμικά είναι:

- Εσωτερικοί επεξεργαστές:** Το Brackets δίνει την δυνατότητα να υπάρχει ένα παράθυρο ανοικτό μέσα σε άλλα παράθυρα για την καλύτερη επεξεργασία του κώδικα. Διαθέτει συγκεκριμένες συντομεύσεις μέσω των οποίων αυξάνεται η παραγωγικότητα του προγραμματιστή.
- Ζωντανή προεπισκόπηση:** Με αυτό το χαρακτηριστικό μπορεί να υπάρξει ζωντανή σύνδεση με τον φυλλομετρητή (browser). Μπορούν να εκτελεστούν δυναμικές αλλαγές οι οποίες αυτόματα αναπαράγονται στην οθόνη χωρίς να χρειάζεται να κάνουμε χειροκίνητα ανανέωση στον φυλλομετρητή.
- Υποστήριξη προεπεξεργαστών:** Το Brackets παρέχει υποστήριξη για προεπεξεργαστές κώδικα όπως είναι οι LESS και SCSS δίνοντας έτσι τη δυνατότητα στους προγραμματιστές να έχουν ένα ενοποιημένο λογισμικό για την ανάπτυξη του κώδικά τους.

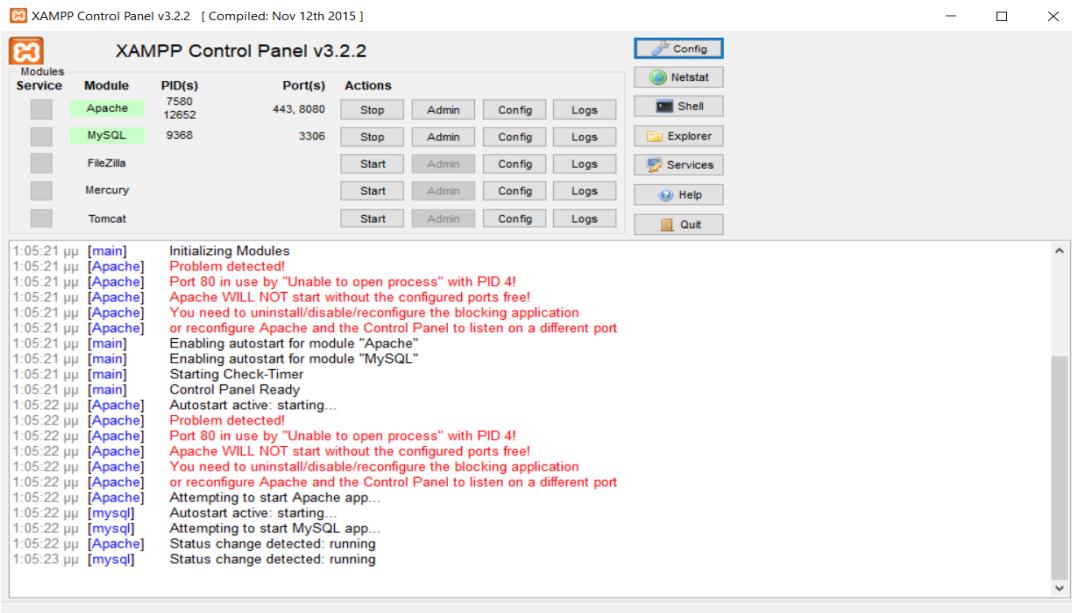
2.3.3. XAMPP

Το Xampp είναι ένα τοπικός διακομιστής που χρησιμοποιήθηκε για την ανάπτυξη του ιστοχώρου – διαδικτυακής εφαρμογής. Πρόκειται για ένα πακέτο προγραμμάτων ελεύθερου λογισμικού. Η ονομασία του οφείλεται στις γλώσσες προγραμματισμού και τον διακομιστή των βάσεων δεδομένων που χρησιμοποιούνται για την προσομοιωμένη φιλοξενία ενός ιστόχωρου σε web server (διακομιστής διαδικτύου) [22].

Η παρόντα διπλωματική βασίστηκε στο λειτουργικό Windows, και με τη βοήθεια του XAMPP προσομοιώθηκε ο τοπικός web server. Λόγω περιορισμών, ήταν αναγκαία η αλλαγή της τοπικής θύρας για τη λειτουργία της διαδικτυακής εφαρμογής από τη θύρα :80 που είναι η προεπιλεγμένη στη θύρα :8080. Το πρωτόκολλο ασφαλείας SSL (Secure Sockets Layer) τοποθετήθηκε στη θύρα :443 που είναι και η προεπιλεγμένη και η βάση δεδομένων MySQL τοποθετήθηκε στη θύρα :3306. Δεν ήταν απαραίτητη η χρήση των πλήρων χαρακτηριστικών του XAMPP και έτσι δεν χρησιμοποιήθηκαν οι προσομοιώσεις του FileZilla, του Tomcat και του Mercury.

Το Xampp δημιουργείται από τα αρχικά:

- X: Cross platform- Σημαίνει ότι υποστηρίζει οποιοδήποτε λογισμικό από OS, W (Windows), M (Macintosh) και L (Linux).
- A: Apache server. Το Xampp περιλαμβάνει Apache Server.
- M: Maria-DB. Το Xampp υποστηρίζει MySQL.
- P: PHP. Το Xampp υποστηρίζει PHP.
- P: Perl. Το Xampp υποστηρίζει Perl.



Σχήμα 2.19 Πίνακας Ελέγχου των XAMPP

2.3.4. Πλατφόρμα Ανάπτυξης Λογισμικού Arduino IDE

To Arduino IDE (Integrated Development Environment) είναι πολλαπλού λειτουργικού (multi-platform) και αποτελεί ένα ολοκληρωμένο περιβάλλον ανάπτυξης, το οποίο σε αντίθεση με άλλα παρόμοια περιβάλλοντα ανάπτυξης όπως το Brackets και το Visual Studio είναι απλούστερο και πιο φιλικό προς το χρήστη.

Περιέχει έναν επεξεργαστή πηγαίου κώδικα, έναν μεταγλωττιστή, εργαλεία αυτόματης παραγωγής κώδικα, αποσφαλμάτωση, συνδέτη, σύστημα ελέγχου εκδόσεων και εργαλεία κατασκευής γραφικών διασυνδέσεων χρήστη για τις υπό ανάπτυξη εφαρμογές. Sketch ονομάζεται το σχεδιάγραμμα που χρησιμοποιεί το Arduino για το εκάστοτε πρόγραμμα. Είναι η μονάδα κώδικα που φορτώνεται και εκτελείται στην πλακέτα Arduino.

Συνδέεται το hardware μέρος του Arduino για να φορτώσει-μεταφέρει προγράμματα και να επικοινωνεί μαζί τους. Η σύνδεση αυτή πραγματοποιείται με μια από τις θύρες USB του υπολογιστή και λόγω του ελεγκτή Serial-over-USB, θα πρέπει να αναγνωριστεί από το λειτουργικό σύστημα ως εικονική σειριακή θύρα.

```
devices_controller | Arduino 1.8.5
File Edit Sketch Tools Help
devices_controller

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

// Enter a MAC address and IP address for your controller below.
byte mac[] = {0x90, 0xA2, 0xDA, 0x0D, 0x48, 0xD3};

// The IP address will be dependent on your local network:
// assign an IP address for the controller:

IPAddress ip(192,168,1,75);
IPAddress gateway(192,168,1,254);
IPAddress subnet(255, 255, 255, 0);

// Initialize the Ethernet server library with the port you want to use.
EthernetServer server(8080);
String readString;
//-----
//-----
// Any extra codes for Declaration :

// Declare Pin 8 as an LED because that's what we will be connecting the LED to. You could use any other pin
int AC = 8;
int DH = 7;

//-----
//-----
void setup()
{
//-----

// Extra Set up code:
```

Σχήμα 2.20 Λογισμικό Arduino IDE

Οι συντομεύσεις αλλά και τα κουμπιά που περιέχει το περιβάλλον εργασίας του Arduino IDE είναι αρκετά εύχρηστα και προσφέρουν γρήγορες λύσεις στα προβλήματα που προκύπτουν κατά την διαδικασία του προγραμματισμού. Στον παρακάτω πίνακα επισημαίνονται οι επίσημες ορολογίες των κουμπιών αλλά και η περιγραφή της λειτουργίας τους.

Πίνακας 2.6: Εργαλεία ανάπτυξης Arduino IDE

Εργαλείο	Περιγραφή λειτουργίας
 Verify	Ελέγχει για συντακτικά λάθη στον κώδικα.
 Upload	Μεταγλωττίζει τον κώδικα και τον φορτώνει στο Arduino. Αν δεν είναι συντακτικά σωστός δεν μπορεί να γίνει η φόρτωση.
 New	Δημιουργεί ένα νέο sketch.
 Open	Παραθέτει ένα μενού με όλα τα sketch. Ενεργοποιώντας ένα από αυτά, θα ανοίξει αυτόματα στο τρέχον παράθυρο.
 Save	Αποθηκεύει ένα sketch.
 Serial Monitor	Ανοίγει τη σειριακή οθόνη και μέσω αυτής παρακολουθείται η ανταλλαγή δεδομένων που γίνεται στην σειριακή θύρα.

Η γλώσσα συγγραφής κώδικα του Arduino, βασίζεται στη γλώσσα Wiring, μια παραλλαγή C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C, καθώς και μερικά χαρακτηριστικά της C++. Ως compiler χρησιμοποιείται ο AVR gcc και ως βασική βιβλιοθήκη C χρησιμοποιείται η AVR libc. Ακόμη υπάρχουν ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν τη διαχείριση του ειδικού hardware του Arduino. Στον παρακάτω πίνακα παρουσιάζονται οι βασικές δομές και εντολές του IDE.

Πίνακας 2.7: Βασικές δομές & εντολές - Arduino IDE

Βασικές δομές /εντολές	Περιγραφή λειτουργίας
if...else	Δομή επιλογής
switch case	Δομή επιλογής που εξαρτάται από την τιμή μιας μεταβλητής. Σε κάθε περίπτωση χρησιμοποιούμε την εντολή break για να τερματίσει η δομή.
for	Δομή επανάληψης με γνωστό αριθμό επαναλήψεων που ορίζεται στην αρχή του βρόγχου.
while	Δομή επανάληψης, ο βρόγχος εκτελείται συνέχεια έως ότου η συνθήκη που ακολουθεί τη while να γίνει ψευδής (false).
do...while	Δομή επανάληψης, ο βρόγχος εκτελείται τουλάχιστον μια φορά έως να γίνει έλεγχος της συνθήκης στο τέλος του βρόγχου.
break	Χρησιμοποιείται για την έξοδο από ένα βρόγχο (for, while,do...while) ή από μια switch...case παρακάμπτοντας την κανονική εκτέλεση των εντολών.
continue	Χρησιμοποιείται για την παράκαμψη μιας επανάληψης ενός βρόγχου και μεταπήδηση στην επόμενη επανάληψη.
return	Τερματίζει μια συνάρτηση ή επιστρέφει μια τιμή σε μια συνάρτηση που έχει κληθεί (τερματίζοντας την).
.goto	Μεταφέρει τη ροή του προγράμματος σε ένα άλλο σημείο χρησιμοποιώντας ετικέτες.
;	Χρησιμοποιείται στο τέλος μιας εντολής.
{}	Περικλείουν ένα σύνολο εντολών. Ένα άγκιστρο ({}) όταν ανοίγει θα πρέπει κάπου να κλείνει.
//	Εισάγεται από τον προγραμματιστή για σχολιασμούς και ευκολία ανάγνωσης του sketch.
#define	Χρησιμοποιείται για τη δήλωση μεταβλητών στην αρχή του sketch. Οι τιμές αυτών των μεταβλητών δεν θα αλλάξουν σε όλη τη διάρκεια της εκτέλεσης του sketch.
#include	Χρησιμοποιείται για τη δήλωση διάφορων βιβλιοθηκών στο sketch.
delay(x)	Σταματά προσωρινά τη ροή του προγράμματος για time ms. Η παράμετρος time είναι unsigned long (από 0 ως 232). Παρά την προσωρινή παύση, συναρτήσεις των οποίων η εκτέλεση ενεργοποιείται από Interrupt θα εκτελεστούν κανονικά κατά τη διάρκεια μιας delay.
millis	Μετρητής που επιστρέφει το χρονικό διάστημα σε ms από την στιγμή που άρχισε η εκτέλεση του προγράμματος. Λόγω του τύπου μεταβλητής (unsigned long δηλ. 32bit) θα συμβεί υπερχείλιση σε 232 ms δηλαδή περίπου σε 50 μέρες, οπότε ο μετρητής θα ξεκινήσει πάλι από το μηδέν.

Κεφάλαιο 3

Υλικό Μέρος και Συνδεσμολογία Συστήματος

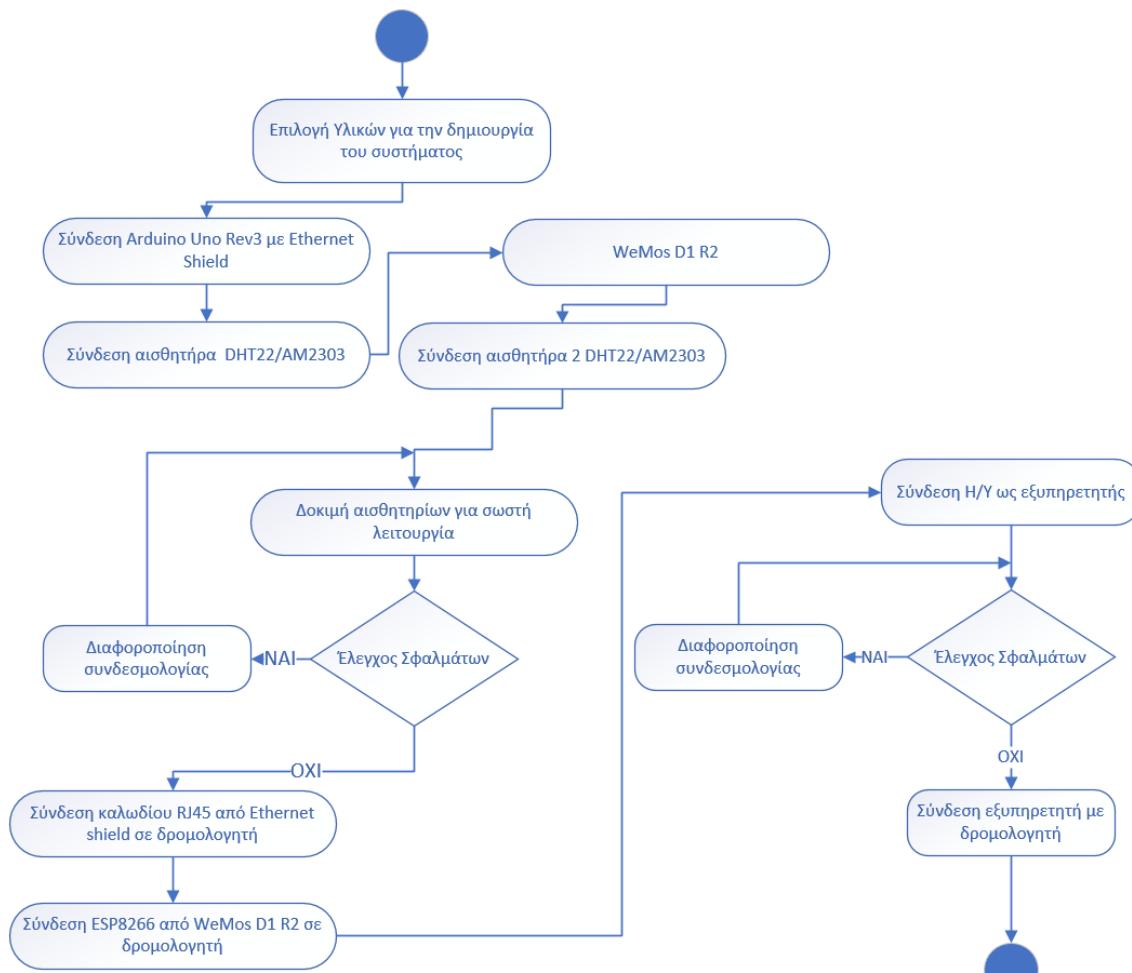
Ένα από τα δύο σημαντικά μέρη της διπλωματικής εργασίας, όπως αναφέρθηκε παραπάνω είναι τα υλικά μέρη που χρησιμοποιήθηκαν. Απαραίτητη προϋπόθεση για την εγκυρότητα και αμεσότητα των δεδομένων προς εμφάνιση στην διαδικτυακή εφαρμογή έχει ο τρόπος με τον οποίο γίνεται η σύνδεση των υλικών καθώς και ο σωστός προγραμματισμός τους. Η συνδεσμολογία της διαδικτυακής εφαρμογής βασίστηκε στους μικροελεγκτές Arduino Uno Rev3 και στον WeMos D1 R2. Η αρχιτεκτονική και τα χαρακτηριστικά των οποίων αναλύθηκαν στο προηγούμενο κεφάλαιο. Έπειτα έγινε η απαραίτητη δρομολόγηση των στατικών IP και αργότερα έγινε η λήψη των δεδομένων από τους αισθητήρες. Εξαιτίας της αρχιτεκτονικής των μικροελεγκτών που χρησιμοποιήθηκαν η οποία είναι ανοικτού κώδικα, με τις κατάλληλες τροποποιήσεις στον προγραμματισμό είναι ικανή η σύνδεση με πληθώρα συσκευών όπως είναι διάφορες πλακέτες, αισθητήρες αλλά και κινητήρες. Έτσι, με την σωστή επιλογή των συσκευών αυτών μπορούν να επιτευχθούν διάφορα project.

Στην παρούσα διπλωματική εργασία, χρησιμοποιήθηκε ένα όχι και τόσο πολύπλοκο σύστημα το οποίο προσφέρει αρκετές επιλογές καθώς και επεκτασιμότητα, εάν αυτή κριθεί αναγκαία. Οι αισθητήρες που χρησιμοποιήθηκαν μαζί με τους μικροελεγκτές, μας δίνουν την δυνατότητα παρακολούθησης του τρόπου με τον οποίο τα μετεωρολογικά φαινόμενα επηρεάζουν τις συνθήκες ενός σπιτιού (στους εσωτερικούς αλλά και εξωτερικούς χώρους) και μας παρέχουν τη δυνατότητα εναλλαγής των καταστάσεων στις έξυπνες συσκευές του σπιτιού (ενεργοποίηση/απενεργοποίηση). Σημαντικά χαρακτηριστικά της διαδικτυακής εφαρμογής είναι, η ζωντανή σε χρόνο αναμετάδοση δεδομένων από τους αισθητήρες προς τον χρήστη, η δημιουργία στατιστικών για την περαιτέρω λήψη αποφάσεων, η δυνατότητα εναλλαγής κατάστασης των συσκευών σε ζωντανό χρόνο καθώς και η ασφαλής λήψη των δεδομένων από τρίτους.

Στις ενότητες που ακολουθούν γίνεται μία εκτενής παρουσίαση όλων των εξαρτημάτων που συνετέλεσαν στην δημιουργία του συστήματος, την ανάλυση των τεχνικών

χαρακτηριστικών και τον τρόπο λειτουργίας τους αλλά και στην ορθή σύνδεση αυτών. Παρατίθενται εικόνες και σχηματικές απεικονίσεις για τον τρόπο σύνδεσης των μικροελεγκτών και των αισθητηρίων οργάνων. Έπειτα αναλύεται και εξηγείται η λειτουργία του κώδικα βάσει του οποίου πραγματοποιείται η λήψη δεδομένων από τη συσκευή. Τέλος, υπολογίζεται το κόστος της ολοκληρωμένης κατασκευής.

Το διάγραμμα ροής της διαδικασίας της συνδεσμολογίας που ακολουθεί στο Σχήμα 3.1 περιγράφει συνοπτικά τα βήματα που εκτελέστηκαν για την υλοποίηση του συστήματος.



Σχήμα 3.1 Διάγραμμα ροής συνδεσμολογίας εξαρτημάτων

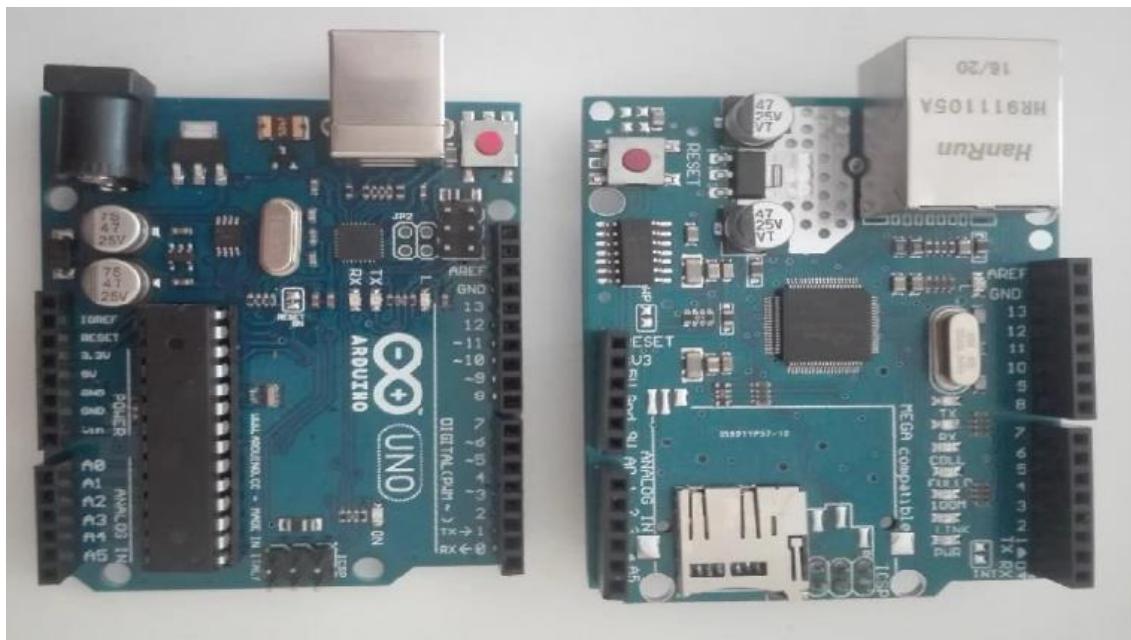
3.1 Hardware Υλοποίησης Συστήματος

Η παρούσα διπλωματική εργασία έχει σκοπό, την δημιουργία ενός συστήματος μέσω του οποίου ο κάθε άνθρωπος θα μπορεί να έχει τον έλεγχο των έξυπνων συσκευών που έχει τοποθετήσει στο σπίτι του αλλά ταυτόχρονα με όσο το δυνατό μικρότερο κόστος. Έτσι λοιπόν, ο χρήστης θα πρέπει να προμηθευτεί τα βασικά εξαρτήματα που είναι τα εξής:

- Μικροελεγκτής Arduino Uno Rev3
- Ασπίδα Arduino Ethernet Shield
- Μικροελεγκτής WeMos D1 R2
- 2x Αισθητήρας θερμοκρασίας DHT22/AM2302
- 2x Πλακέτα δοκιμών (Breadboard 400)

Το σύνολο του συστήματος θα πρέπει να συμπληρωθεί από καλώδια σύνδεσης (jumper cables) καθώς και έναν δρομολογητή (router). Στις επόμενες ενότητες εξηγούνται πιο αναλυτικά το κάθε υλικό και ο τρόπος λειτουργίας αυτών.

3.1.1. Arduino Uno Rev3 – Arduino Ethernet Shield

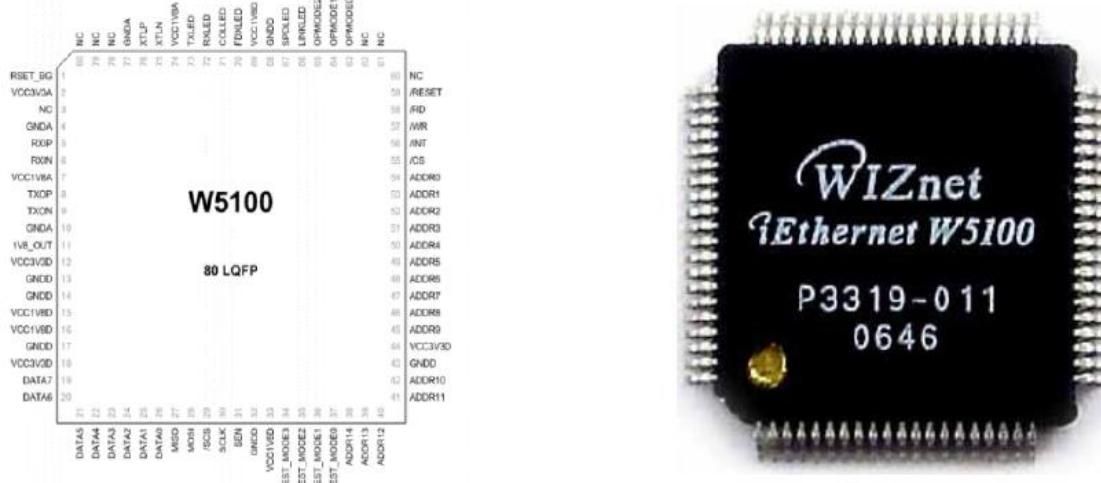


Σχήμα 3.2 Arduino Uno Rev3 και Ethernet Shield

Ξεκινώντας την συλλογή δεδομένων θερμοκρασίας και υγρασίας εντός του χώρου του σπιτιού κρίνεται απαραίτητη η δημιουργία ενός υποσυστήματος. Βασικά υλικά μέρη αυτού είναι ο μικροελεγκτής Arduino Uno Rev3 καθώς και το Ethernet Shield τα οποία φαίνονται στο Σχήμα 3.2. Το υποσύστημα συμπληρώνουν ένας αισθήτηρας θερμοκρασίας/υγρασίας DHT22/AM2302 και μία πλακέτα δοκιμών Breadboard 400 οι οποίες θα περιγραφούν σε παρακάτω κεφάλαιο. Η χρήση αυτού του υποσυστήματος αποσκοπεί στην ενσύρματη μεταφορά των συλλεγόμενων δεδομένων στον τοπικό εξυπηρετητή και στη βάση δεδομένων μέσω καλωδίου Ethernet.

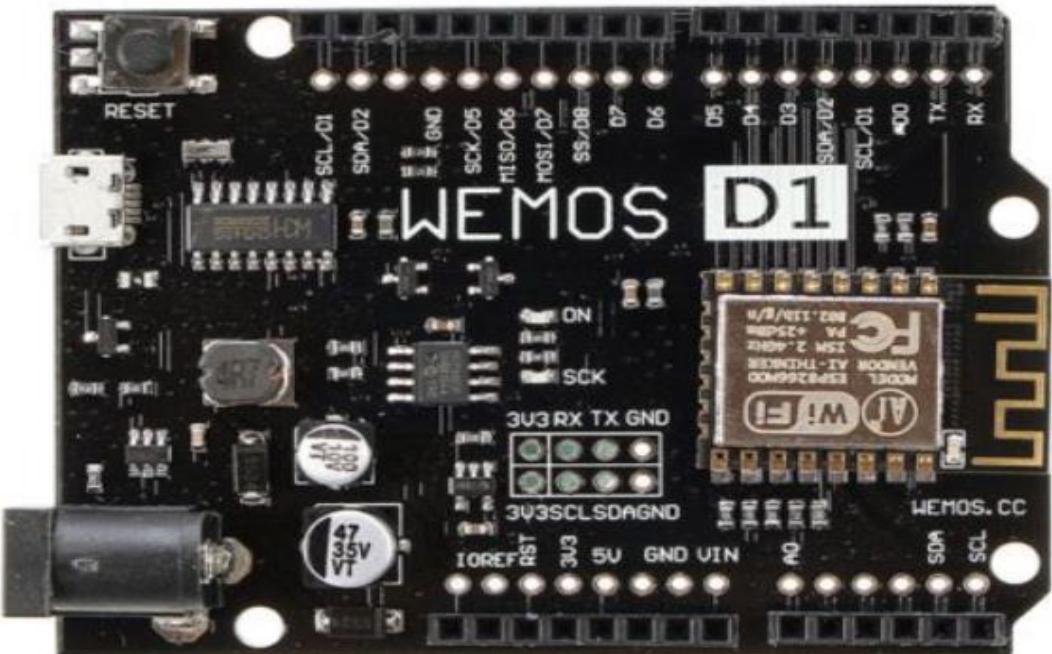
Τα τεχνικά χαρακτηριστικά των δύο αυτών πλακετών έχουν παρουσιαστεί αναλυτικά στο προηγούμενο κεφάλαιο. Στην ενότητα αυτή πραγματοποιείται η περιγραφή της λειτουργίας τους.

Η συλλογή των δεδομένων από τον μικροελεγκτή Arduino Uno Rev3 γίνεται μέσω της συνδεσμολογίας που έχει πραγματοποιηθεί μεταξύ αυτού και του αισθητήρα θερμοκρασίας/υγρασίας. Έπειτα ο μικροελεγκτής καθώς έχει συνδεθεί μέσω του δρομολογητή στον τοπικό εξυπηρετητή στέλνει τα δεδομένα που έχει συλλέξει μέσω του Ethernet Shield. Το τελευταίο, επιτελεί έναν πολύ σημαντικό ρόλο, καθώς μέσω αυτού γίνεται η ενσύρματη αποστολή δεδομένων στον τοπικό εξυπηρετητή μέσω του καλωδίου RJ45 Ethernet. Μπορεί να χρησιμοποιηθεί είτε σαν server είτε σαν client και υποστηρίζεται άμεσα από την επίσημη Ethernet βιβλιοθήκη. Χρησιμοποιώντας τη βιβλιοθήκη αυτή, είναι δυνατή η συγγραφή των sketches για την σύνδεση της κατασκευής στον τοπικό εξυπηρετητή. Όπως αναφέρθηκε και προηγουμένως βασίζεται στον ethernet ελεγκτή W5100 της Wiznet.



Σχήμα 3.3 WIZnet Ethernet W5100 Chipset & Schematic Diagram

3.1.2. WeMos D1 R2



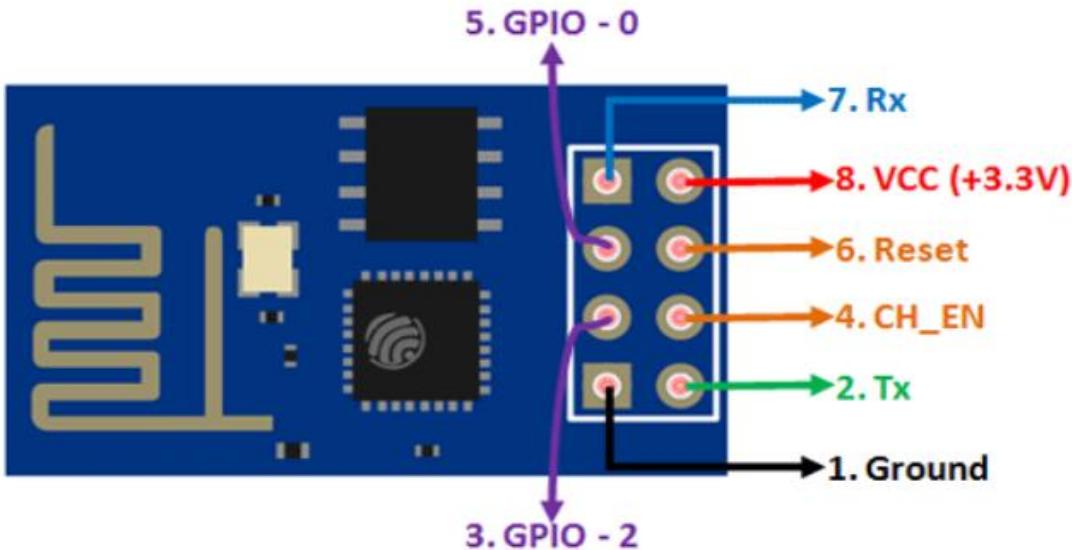
Σχήμα 3.4 WeMos D1 R2

Συνεχίζοντας την συλλογή δεδομένων θερμοκρασίας και υγρασίας εκτός του χώρου του σπιτιού κρίνεται απαραίτητη η δημιουργία ενός δεύτερου υποσυστήματος. Βασικά υλικά μέρη αυτού είναι ο μικροελεγκτής WeMos D1 R2 και το microchip ESP8266 το οποίο είναι ενσωματωμένο στην πλακέτα του μικροελεγκτή. Η πλακέτα φαίνεται στο Σχήμα 3.3. Το υποσύστημα συμπληρώνουν ένας αισθήτηρας θερμοκρασίας/υγρασίας DHT22/AM2302 και μία πλακέτα δοκιμών Breadboard 400 οι οποίες προαναφέραμε πως θα περιγραφούν σε παρακάτω κεφάλαιο. Η χρήση αυτού του υποσυστήματος αποσκοπεί στην ασύρματη μεταφορά των συλλεγόμενων δεδομένων στον τοπικό εξυπηρετητή και στη βάση δεδομένων μέσω πρωτόκολλο WiFi.

Τα τεχνικά χαρακτηριστικά της πλακέτας έχουν παρουσιαστεί αναλυτικά στο προηγούμενο κεφάλαιο. Στην ενότητα αυτή πραγματοποιείται η περιγραφή της λειτουργίας τους.

Η συλλογή των δεδομένων από τον μικροελεγκτή WeMos D1 R2 γίνεται μέσω της συνδεσμολογίας που έχει πραγματοποιηθεί μεταξύ αυτού και του αισθητήρα θερμοκρασίας/υγρασίας. Έπειτα ο μικροελεγκτής καθώς έχει συνδεθεί μέσω του δρομολογητή στον τοπικό εξυπηρετητή στέλνει τα δεδομένα που έχει συλλέξει μέσω του WiFi. Το microchip υποστηρίζεται από μεγάλη κοινότητα καθώς και από αρκετές βιβλιοθήκες οι οποίες έχουν συγγραφεί για τις διαφορετικές λειτουργίες του ESP8266. Χρησιμοποιώντας τις βιβλιοθήκες αυτές, είναι δυνατή η συγγραφή των sketches για τη

σύνδεση της κατασκευής στον τοπικό εξυπηρετητή.



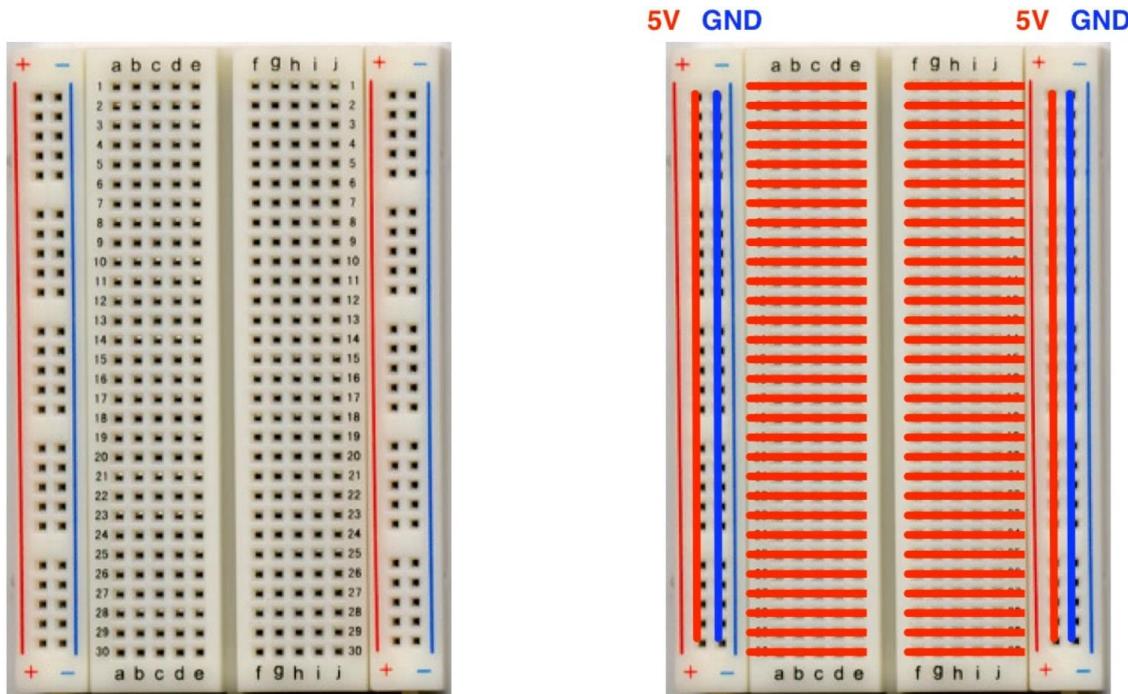
Σχήμα 3.5 *ESP8266 microchip Schematic*

3.1.3. Πλακέτα Δοκιμών (Breadboard 400)

Η πλακέτα δοκιμών είναι ένα από τα βασικά κομμάτια και των δύο υποσυστημάτων που χρησιμοποιήθηκαν για την παρούσα διπλωματική εργασία, καθώς πάνω σ' αυτήν βασίζεται η δημιουργία του κυκλώματος και η ταχεία και ασφαλής αποστολή των δεδομένων από τον αισθητήρα προς τον μικροελεγκτή. Τονίζουμε πως η πλακέτα δοκιμών που χρησιμοποιήσαμε είναι μια πλακέτα solderless breadboard η οποία διαθέτει 400 pins δίνοντας μας έτσι την δυνατότητα να μπορούν να γίνουν εύκολα τροποποιήσεις στη συνδεσμολογία, καθώς δεν υπάρχουν κολλημένα μέρη.

Τα υλικά μέρη πέραν του μικροελεγκτή, τοποθετούνται επάνω στην πλακέτα δοκιμών, στην παρούσα περίπτωση ο αισθητήρας και τα καλώδια σύνδεσης, εισάγοντας τους ακροδέκτες τους στις ειδικές οπές που υπάρχουν στην επιφάνειά του. Οι οπές αυτές απέχουν η κάθε μία μεταξύ τους κατά 2,54mm. Κάτω από κάθε “γραμμή-σειρά” της πλαστικής επιφάνειας, στο εσωτερικό του breadboard, βρίσκονται μεταλλικά κλιπ που επιτρέπουν τη ροή ρεύματος και επιτυγχάνουν την αγωγιμότητά της. Κάθε ένα από αυτά τα κλιπ αντιστοιχεί σε πέντε οριζόντιες οπές που σημαίνει ότι σε κάθε ξεχωριστό τμήμα μπορούν να συνδεθούν μέχρι πέντε εξαρτήματα. Πέρα από τις οριζόντιες γραμμές, στις πλαϊνές πλευρές βρίσκονται και κάθετες γραμμές πανομοιότυπες με τις οριζόντιες με τη μόνη διαφορά, ότι αυτές συνδέονται σε όλο το μήκος και δίνουν ρεύμα οπουδήποτε χρειαστεί στο κύκλωμα. Οι ετικέτες “+” και “-” υποδεικνύουν τις θετικές και αρνητικές πλευρές.

Στο παρακάτω Σχήμα εφιανίζεται η πλακέτα δοκιμών καθώς και οι συνδέσεις αυτής.



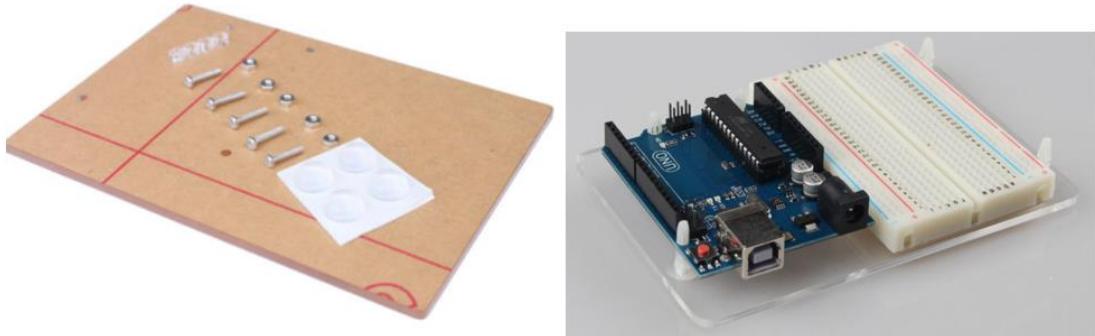
Σχήμα 3.6 Breadboard 400 Schematic

3.1.4. Ακρυλική Βάση Τοποθέτησης Μικροελεγκτή – Πλακέτας Δοκιμών

Για την ομαλότερη τοποθέτηση των υποσυστημάτων στους εξωτερικούς χώρους του σπιτιού ήταν προαιρετική η ακρυλική βάση πάνω στην οποία τοποθετήθηκαν τα δύο υποσυστήματα. Η ακρυλική βάση είναι ουσιαστικά ένα κομμάτι από διαφανές πλαστικό πάνω στο οποίο έχουν δημιουργηθεί οπές σύμφωνα με τα πρότυπα του Arduino Uno Rev3 έτσι ώστε να υπάρχει η σωστή εφαρμογή του επάνω στην ακρυλική βάση. Υπάρχουν οι κατάλληλοι κοχλίες οι οποίοι βιδώνουν από το κάτω μέρος της ακρυλικής βάσης, διαπερνάνε τον μικροελεγκτή και ασφαλίζονται με παξιμάδια. Επίσης υπάρχει χώρος, ικανός να φιλοξενήσει επάνω της η βάση και μία πλακέτα δοκιμών breadboard 400.

Σκοπός της βάσης είναι ασφαλής και σταθερή τοποθέτηση του κυκλώματος. Λόγω της φύσης της βάσης (πλαστικό) προστίθεται και ένα επιπλέον μονωτικό επίπεδο έτσι ώστε ο μικροελεγκτής να μην εφάπτεται σε περιοχές που μπορεί να προκαλέσουν βραχυκύκλωμα.

Η ακρυλική βάση, οι κοχλίες και τα παξιμάδια καθώς και το ολοκληρωμένο οπτικό αποτέλεσμα παρουσιάζονται στο επόμενο Σχήμα.

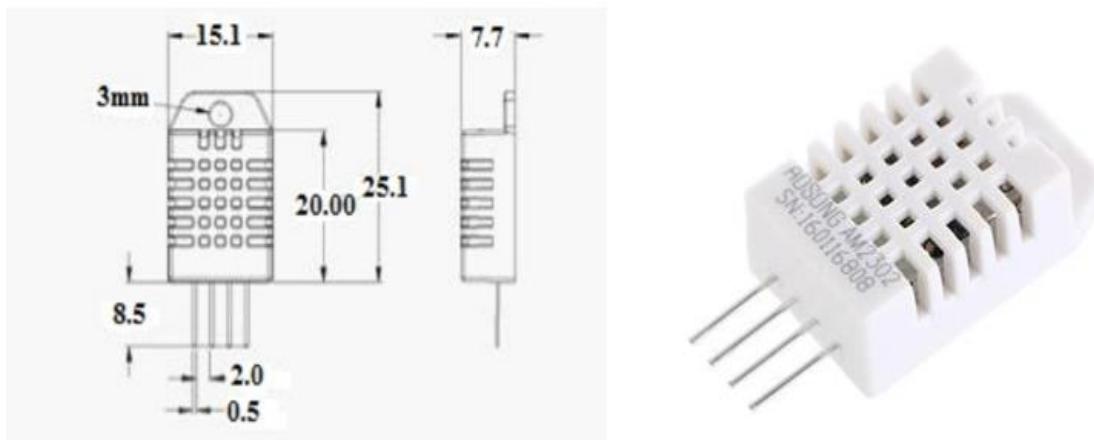


Σχήμα 3.7 Ακρυλική Βάση

3.1.5. Αισθητήριο Θερμοκρασίας & Υγρασίας DHT22/AM2302

Για την συλλογή των δεδομένων στους χώρους που τοποθετήκαν τα υποσυστήματα χρησιμοποιήθηκε ο ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας DHT22/AM2302. Όσο αφορά την ποιότητα κατασκευής, ο DHT22/AM2302 είναι από πλαστικό σώμα, είναι διάτρητος σε αρκετά σημεία και στο κάτω μέρος του υπάρχουν 4 ακροδέκτες. Λόγω του μικρού μεγέθους του αλλά και του μικρής πλακέτας που φέρει, δικαιολογείται και η χαμηλή σε κόστος αγοράς, τιμή του.

Αποτελεί ένα σύνθετο αισθητήρα που περιέχει μια βαθμολογημένη παραγωγή ψηφιακών σημάτων. Έχει μικρή κατανάλωση ρεύματος, μπορεί να λαμβάνει τάση εισόδου την οποία μετατρέπει σε θερμοκρασία και την υγρασία του χώρου με ακρίβεια σε απόσταση 20 m και να μεταδίδει δεδομένα στην πλακέτα με την οποία έχει συνδεθεί κάθε 2 sec. Βάσει αυτών των ιδιαιτεροτήτων του χρησιμοποιείται σε πολλές εφαρμογές και υπάρχει πληθώρα βιβλιοθηκών στο Διαδίκτυο.



Σχήμα 3.8 Σχηματική Απεικόνιση & Διαστάσεις του αισθητηρίου DHT22/AM2302

Σημαντικό είναι να αναφέρουμε τα τεχνικά χαρακτηριστικά του εν λόγω αισθητήρα καθώς πληρεί τα τεχνικά χαρακτηριστικά που έχει θέσει ως πρότυπο ο παγκόσμιος μετεωρολογικός οργανισμός (WMO) και έτσι ο DHT22/AM2302 καθίσταται ιδανικός αισθητήρας για τις μετρήσεις που χρειάζονται στην παρούσα διπλωματική εργασία. Τα πρότυπα του παγκόσμιου μετεωρολογικού οργανισμού είναι:

- Εύρος τιμών περιβαλλοντικής καταγραφής θερμοκρασίας: -30°C έως +50°C
- Εύρος τιμών περιβαλλοντικής καταγραφής υγρασίας: 5% έως 100%

Τα αναλυτικά τεχνικά χαρακτηριστικά του αισθητήρα αναλύονται στον παρακάτω πίνακα.

Πίνακας 3.1: Τεχνικά Χαρακτηριστικά του DHT22/AM2302

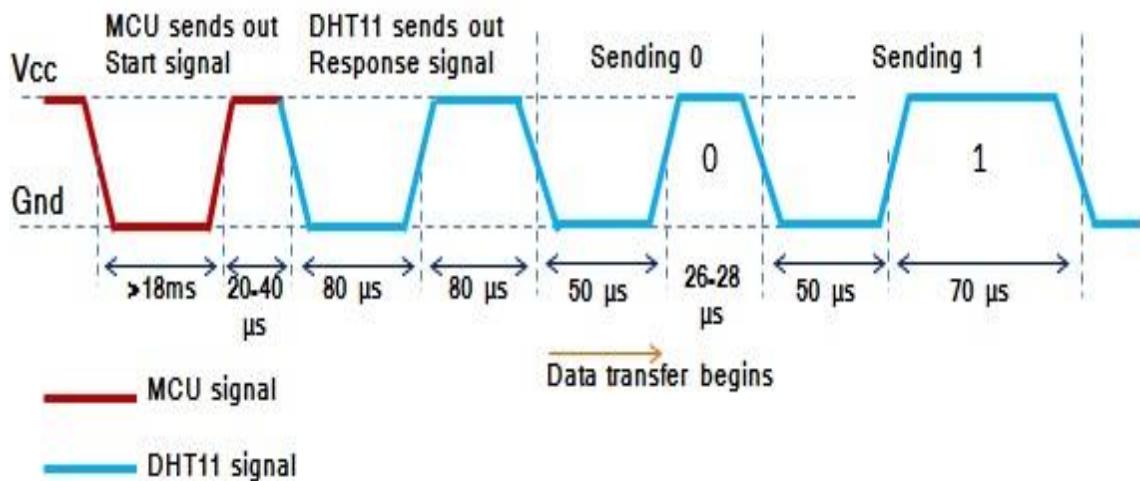
Μοντέλο	DHT22	
Τάση λειτουργίας	3.3-6V DC	
Σήμα εξόδου	Ψηφιακό σήμα μέσω μονού-bus	
Στοιχείο αίσθησης	Πολυμερής Πυκνωτής	
Εύρος λειτουργίας	Υγρασία: 0-100% RH	Θερμοκρασία: -40 – 80°C
Ακρίβεια	Υγρασία: ±2-5%RH	Θερμοκρασία: ±0,5°C
Ανάλυση/Εναισθησία	Υγρασία: 0,1%RH	Θερμοκρασία: 0,1°C
Επαναληψιμότητα	Υγρασία: ±1%RH	Θερμοκρασία: ±0,2°C
Υστέρηση υγρασίας	±0,3%RH	
Μακροσταθερότητα	+0.5%RH/χρόνο	
Ανταλλαξιμότητα	Πλήρως έναλλαξιμος	
Δειγματοληψία	2 δευτερόλεπτα	

Η λειτουργία του αισθητήρα κατά την επικοινωνία του με το Arduino περιγράφεται με τον ακόλουθο τρόπο. Αρχικά, το Arduino μέσω της ψηφιακής του εξόδου στέλνει ένα σήμα έναρξης της τάξης των 500μs περίπου και αναμένει απάντηση από τον αισθητήρα. Στη συνέχεια, ο αισθητήρας μειώνει το σήμα στα 80μs και στέλνει εκ νέου ένα παρόμοιο σήμα. Ταυτόχρονα, στέλνει ένα ακόμα σήμα της τάξης των 50μs (διαδικασία χειραψίας). Τα επόμενα σήματα που ακολουθούν είναι ψηφιακά και κάθε 28μs για λογικό 0 και 70μs για λογικό 1. Κάθε δύο δευτερόλεπτα, ο αισθητήρας εκπέμπει 40 bit πληροφορίας τα οποία αντιστοιχούν σε:

- 8 bit για την τιμή της υγρασίας
- 8 bit για την ακέραια τιμή της υγρασίας

- 8 bit για την τιμή της θερμοκρασίας
- 8 bit για την ακέραια τιμή της θερμοκρασίας
- 8 bit για το bit ελέγχου ισοτιμίας των δεδομένων.

Η διαδικασία επικοινωνίας του αισθητήρα με τον μικροελεγκτή παρουσιάζεται στο παρακάτω Σχήμα.

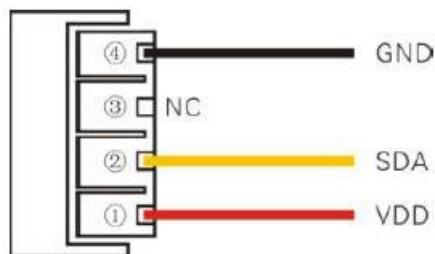


Σχήμα 3.9 Διαδικασία επικοινωνίας DHT22/AM2302 με Arduino

Οι ακροδέκτες (pins) που διαθέτει είναι 4 και οι 3 από αυτούς χρησιμοποιούνται για την καταγραφή των απαραίτητων δεδομένων. Η αντιστοιχία τους καθώς και η λειτουργία καθενός περιγράφεται στον πίνακα και το Σχήμα που ακολουθούν.

Πίνακας 3.2: Περιγραφή λειτουργίας ακροδεκτών DHT22/AM2302

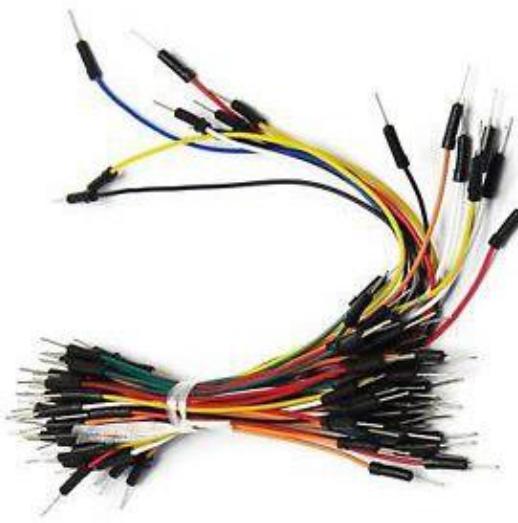
Ακροδέκτης (Pin)	Όνομα	Περιγραφή
(1)	VDD	Τροφοδοσία (3.3-6V DC)
(2)	SDA	Θύρα σειριακών δεδομένων
(3)	NC	Κενός
(4)	GND	Γείωση



Σχήμα 3.10 Σχηματική Απεικόνιση ακροδεκτών DHT22/AM2302

3.1.6. Καλώδια Jumper

Τα καλώδια Jumper είναι κυλινδρικά ηλεκτρικά καλώδια με βύσματα (συνδέτες) ή αγωγούς στην κάθε άκρη τους. Χρησιμοποιούνται για τη σύνδεση εξαρτημάτων μεταξύ τους με ή χωρίς την βοήθεια κάποιου breadboard. Ανάλογα με τις άκρες τους τα καλώδια αυτά μπορεί να είναι αρσενικά (συνδέτες) ή θηλυκά (αγωγοί). Για να πραγματοποιηθούν οι συνδέσεις πάνω στο breadboard στη συγκεκριμένη περίπτωση χρησιμοποιήθηκαν αρσενικά καλώδια.



Σχήμα 3.11 Καλώδια Jumper

3.1.7. Καλώδια Σύνδεσης Πλακετών με H/Y

Για την σωστή λειτουργία του συστήματος της παρούσας διπλωματικής εργασίας, είναι απαραίτητη η χρήση ορισμένων καλωδίων για τη σύνδεση των πλακετών και των ασπίδων με τον H/Y και τον τοπικό εξυπηρετητή. Η σύνδεση των μικροελεγκτών Arduino Uno Rev3 καθώς και του WeMos D1 R2 με τον H/Y γίνεται με τα καλώδια:

- **WeMos D1 R2 – H/Y:** micro USB σε USB
- **Arduino Uno Rev3 – H/Y:** USB Type-B σε USB

Τα καλώδια που αναφέρθηκαν παραθέτονται στα παρακάτω Σχήματα.



Σχήμα 3.12 Καλώδιο micro USB σε USB



Σχήμα 3.13 Καλώδιο USB Type-B σε USB

Ακόμα, ένα καλώδιο Ethernet RJ45 για τη σύνδεση της πλακέτας Ethernet shield με το router του εκάστοτε χρήστη, έτσι ώστε να είναι εφικτή η διαδικτυακή παρακολούθηση των δεδομένων που συλλέγονται.

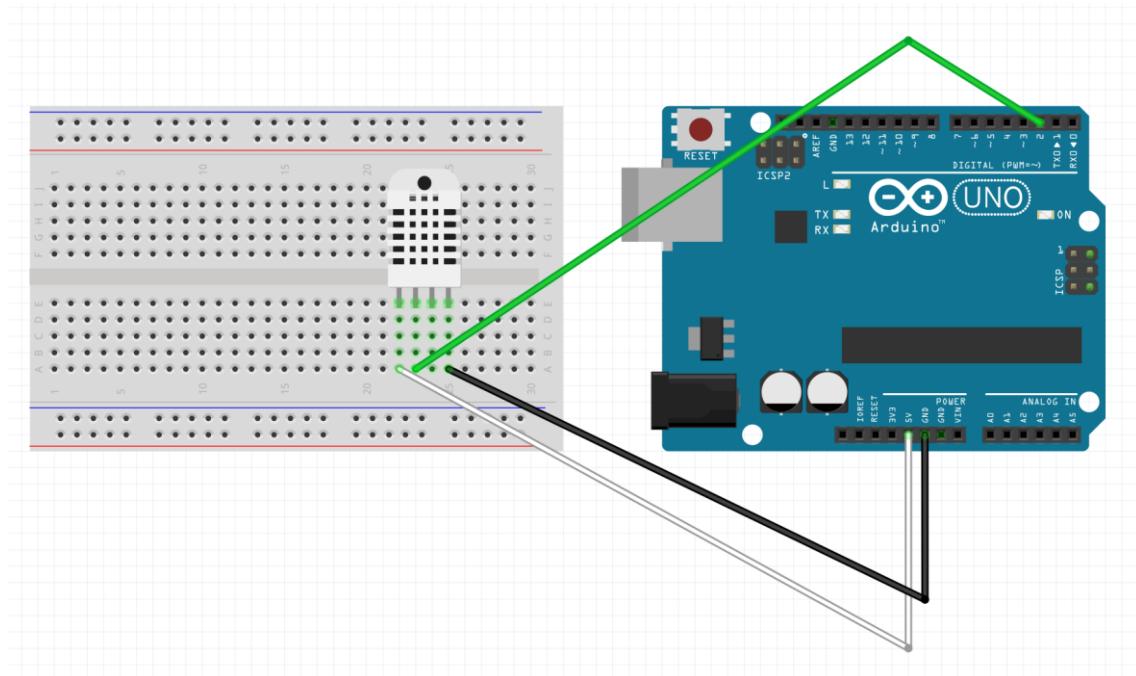


Σχήμα 3.14 Καλώδιο Ethernet RJ45

3.2 Συνδεσμολογία Συστήματος (Hardware Wiring)

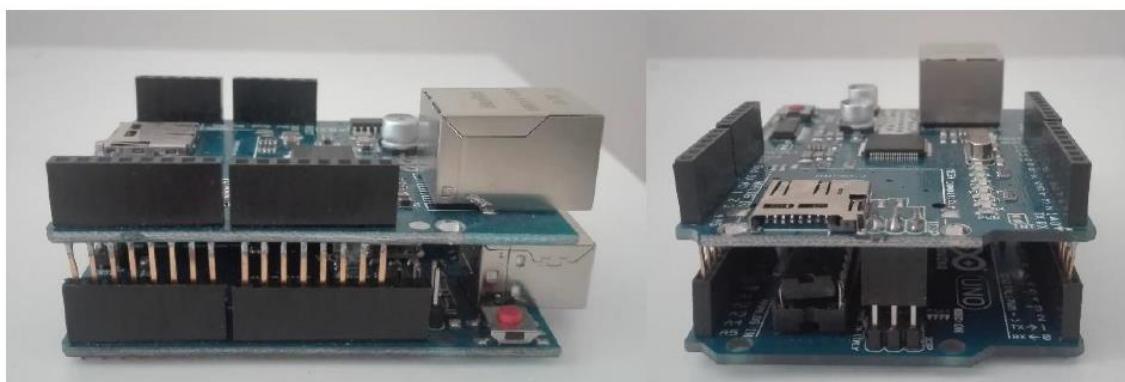
Για την λήψη των δεδομένων, όπως προαναφέρθηκε δημιουργήθηκαν δύο υποσυστήματα, στα οποία υπήρχε από ένας μικροελεγκτής και ένα αισθητήρας θερμοκρασίας/υγρασίας. Στην ενότητα αυτή παρουσιάζεται η βήμα προς βήμα διαδικασία που απαιτείται για τη σύνδεση όλων των επιμέρους υλικών που προαναφέρθηκαν.

Συνδεσμολογία Υποσυστήματος με Arduino Uno Rev3



Σχήμα 3.15 Σχηματική απεικόνιση σύνδεσης Arduino Uno Rev3 και DHT22

Αρχικά, στον μικροελεγκτή συνδέεται η πλακέτα Ethernet Shield. Οι ακροδέκτες των δύο συνεργαζόμενων προαναφερθέντων συσκευών έχουν ίδια τοποθέτηση στις πλακέτες, με αποτέλεσμα αυτοί του Ethernet Shield να εφάπτονται με απόλυτη ακρίβεια στις αντίστοιχες υποδοχές του Arduino Uno Rev3 όπως φαίνεται και στο Σχήμα 3.15. Για την λειτουργία της πλακέτας Ethernet Shield απαιτείται η σύνδεση των ακροδεκτών 11,12 και 13 του Arduino με τους αντίστοιχους τις πλακέτας. Είναι οι ακροδέκτες που παρέχουν τη σύγχρονη σειριακή επικοινωνία δεομένων (SPI) μεταξύ των μικροελεγκτών που βρίσκονται στην κάθε μια πλακέτα αντίστοιχα. Αξίζει να σημειωθεί ότι, οι ακροδέκτες αυτοί δεν μπορούν να χρησιμοποιηθούν ως είσοδοι ή έξοδοι. Το Ethernet Shield δεν απαιτεί επιπλέον τροφοδοσία, τροφοδοτείται μέσω της πλακέτας στην οποία έχει ενσωματωθεί. Στην παρούσα κατασκευή από 5V dc/ 0,7mA που της παρέχει το Arduino Uno Rev3.



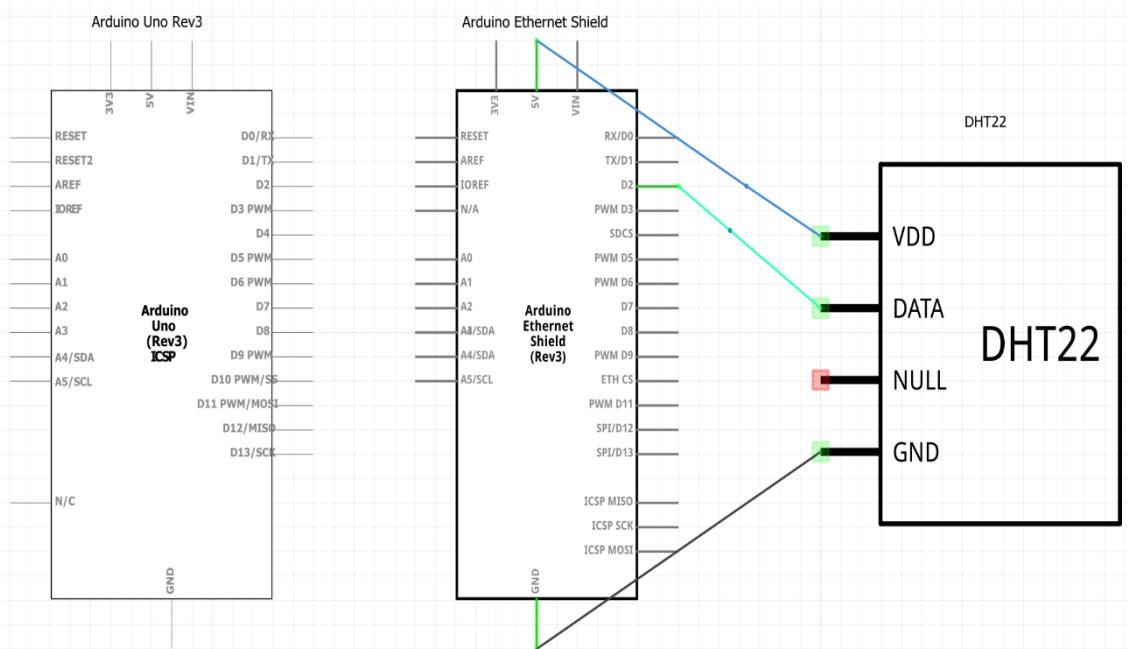
Σχήμα 3.16 Σύνδεση Arduino Uno Rev3 και Arduino Ethernet Shield

Σειρά έχει η τοποθέτηση του αισθητήρα DTH22/AM2302 πάνω στην πλακέτα δοκιμών καθώς και η καλωδίωσή του μέσω των jumper cables και η σύνδεση αυτού με το ethernet shield έτσι ώστε να πραγματοποιηθεί η δημιουργία του πρώτου υποσυστήματος. Όπως τονίσαμε παραπάνω, τρεις από τους τέσσερεις ψηφιακούς ακροδέκτες του αντιστοιχούν σε υποδοχές, καθώς ο ένας είναι κενός. Η σύνδεση του με τον μικροελεγκτή Arduino γίνεται μέσω των υποδοχών της πλακέτας Ethernet Shield χωρίς να υπάρχει καμία διαφορά με την απευθείας σύνδεση πάνω στον πρώτο.

Είναι απαραίτητο να κατανοήσουμε την σωστή συνδεσμολογία που απαιτείται στον αισθητήρα DHT22/AM2302 έτσι ώστε να πάρουμε τα επιθυμητά αποτελέσματα. Πρέπει λοιπόν, να συνδέσουμε σωστά τον κάθε ακροδέκτη του αισθητήρα με κάποιο υποδοχή του μικροελεγκτή. Ξεκινώντας την αρίθμηση των ακροδεκτών από αριστερά προς τα δεξιά όπως φαίνεται στο Σχήμα 3.9, ο πρώτος ακροδέκτης χρησιμοποιείται για την τροφοδοσία των 5V και συνδέεται, με το άσπρο καλώδιο, στην υποδοχή με την αντίστοιχη υπόδειξη. Ο δεύτερος ακροδέκτης είναι υπεύθυνος για τη μεταφορά των συλλεγόμενων δεδομένων. Αποτελεί ουσιαστικά την ταυτότητα του αισθητήρα στην πλακέτα, στην παρούσα κατασκευή συνδέεται, με το πράσινο καλώδιο, στη ψηφιακή είσοδο της πλακέτας η οποία είναι αριθμημένη ως 2. Τέλος, η χρησιμότητα του τέταρτου ακροδέκτη αφορά τη γείωση και συνδέεται, με το μαύρο καλώδιο, σε μια από τις

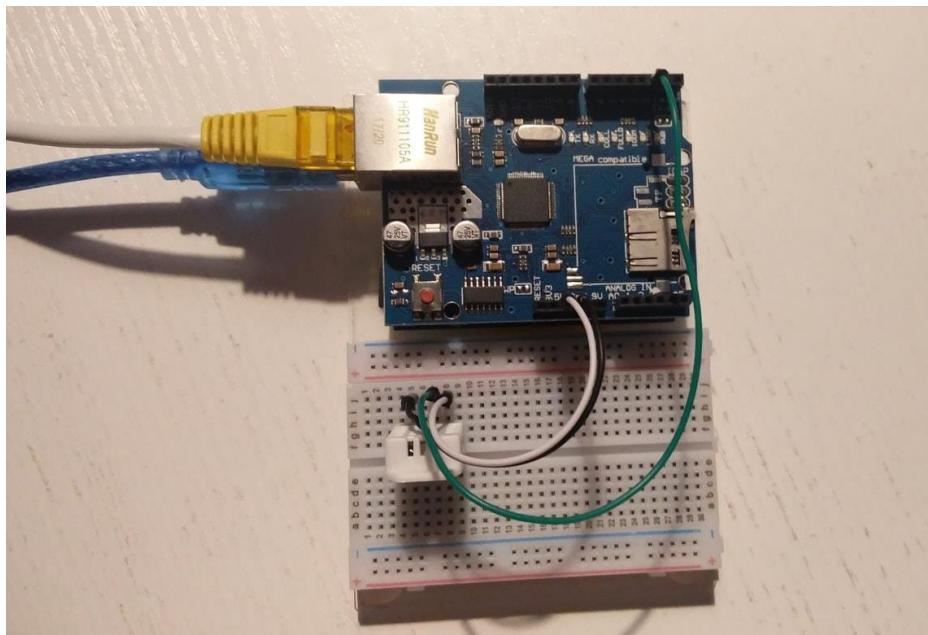
αναγραφόμενες GND υποδοχές. Αξίζει να σημειωθεί σε αυτό το σημείο πώς το χρώμα των καλωδίων που χρησιμοποιήθηκαν στην παρούσα διπλωματική εργασία δεν παίζουν κάποιο συγκεκριμένο ρόλο και πως ανά πάσα στιγμή μπορεί να γίνει η αλλαγή των καλωδίων με άλλα καλώδια διαφορετικού χρώματος.

Στο παρακάτω Σχήμα φαίνεται το ηλεκτρολογικό σχέδιο σύνδεσης του υποσυστήματος που αναλύσαμε παραπάνω. (Λόγω τεχνικών προβλημάτων στο πρόγραμμα που σχεδιάστηκαν οι απεικονίσεις των συστημάτων, έχει γίνει η αλλαγή του άσπρου καλωδίου με μπλε καλώδιο)



Σχήμα 3.17 Ηλεκτρολογική σχηματική απεικόνιση συνδεσμολογίας

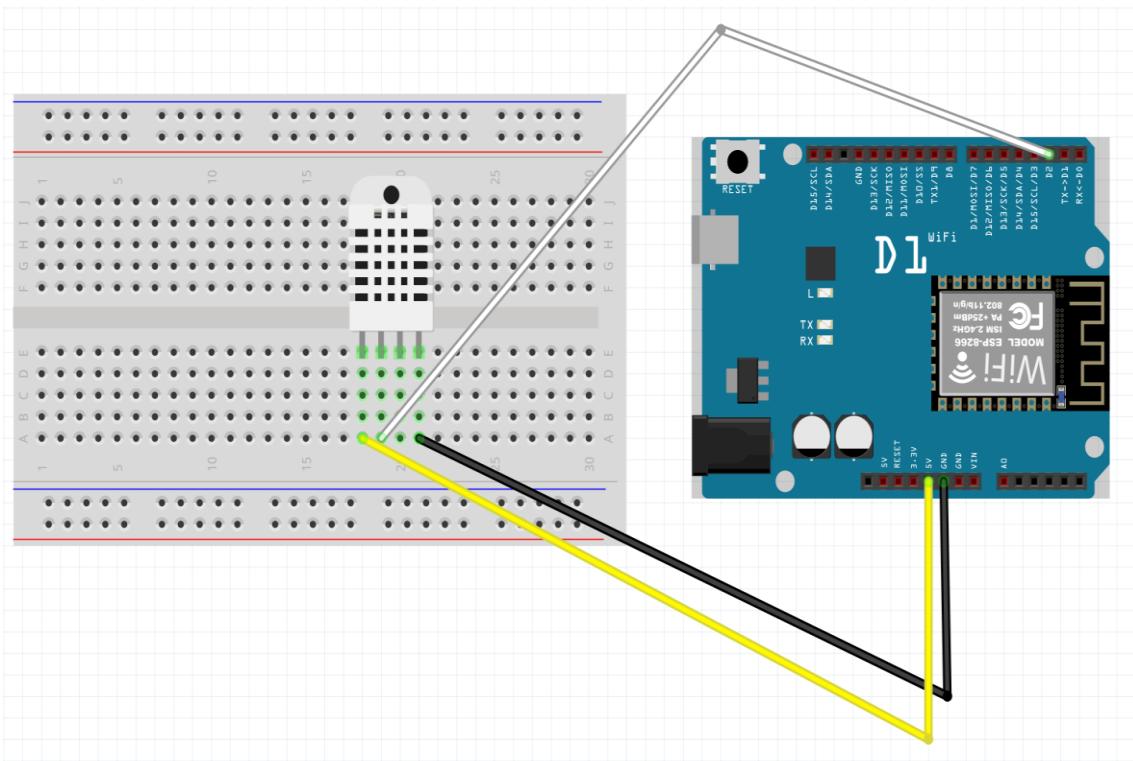
Τέλος, είναι απραίτητη η σύνδεση του μικροελεγκτή Arduino Uno Rev3 με τον H/Y, μέσω ενός USB καλωδίου, τόσο για τροφοδοσία όσο και για την μεταφορά και συλλογή των δεδομένων, και το ανέβασμα του κώδικα στην πλακέτα. Επίσης πραγματοποιείται η σύνδεση του Ethernet Shield με το router αποτελούν τα τελικά αλλά αρκετά σημαντικά βήματα για την ολοκλήρωση της κατασκευής του πρώτου υποσυστήματος. Στο Σχήμα 3.17, παρουσιάζεται η τελική μορφή της συνδεσμολογίας που περιγράφηκε.



Σχήμα 3.18 Τελική μορφή υποσυστήματος

Συνδεσμολογία Υποσυστήματος με WeMos D1 R2

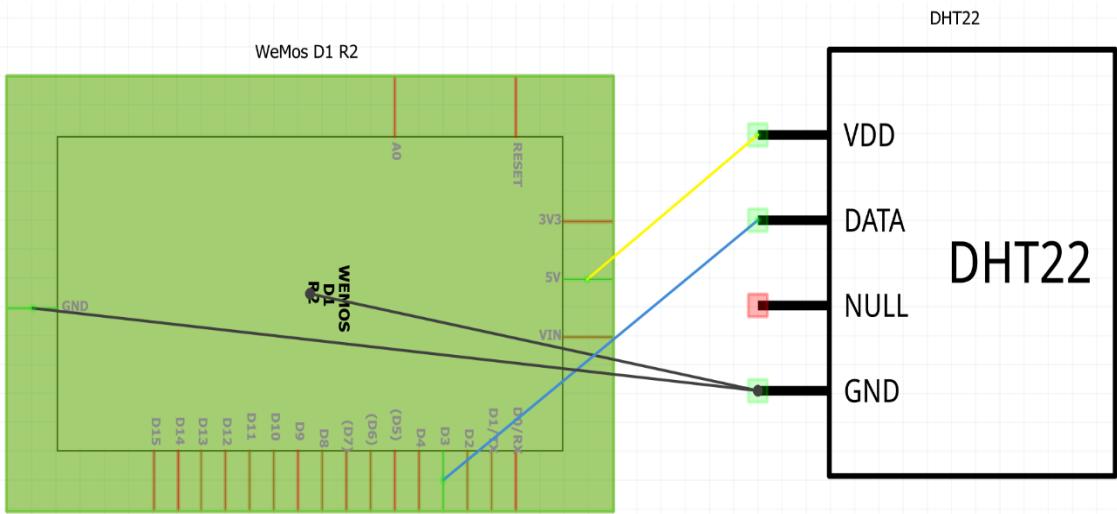
Η συνδεσμολογία του δεύτερου υποσυστήματος που έχει ως μικροελεγκτή το WeMos D1 R2 διαφέρει σε σχέση με το προηγούμενο υποσύστημα. Σε αυτό το υποσύστημα δεν υπάρχει κάποιο shield με αποτέλεσμα μην υπάρχουν τέσσερα υλικά μέρη αλλά τρία.



Σχήμα 3.19 Σχηματική απεικόνιση σύνδεσης WeMos D1 R2 και DHT22

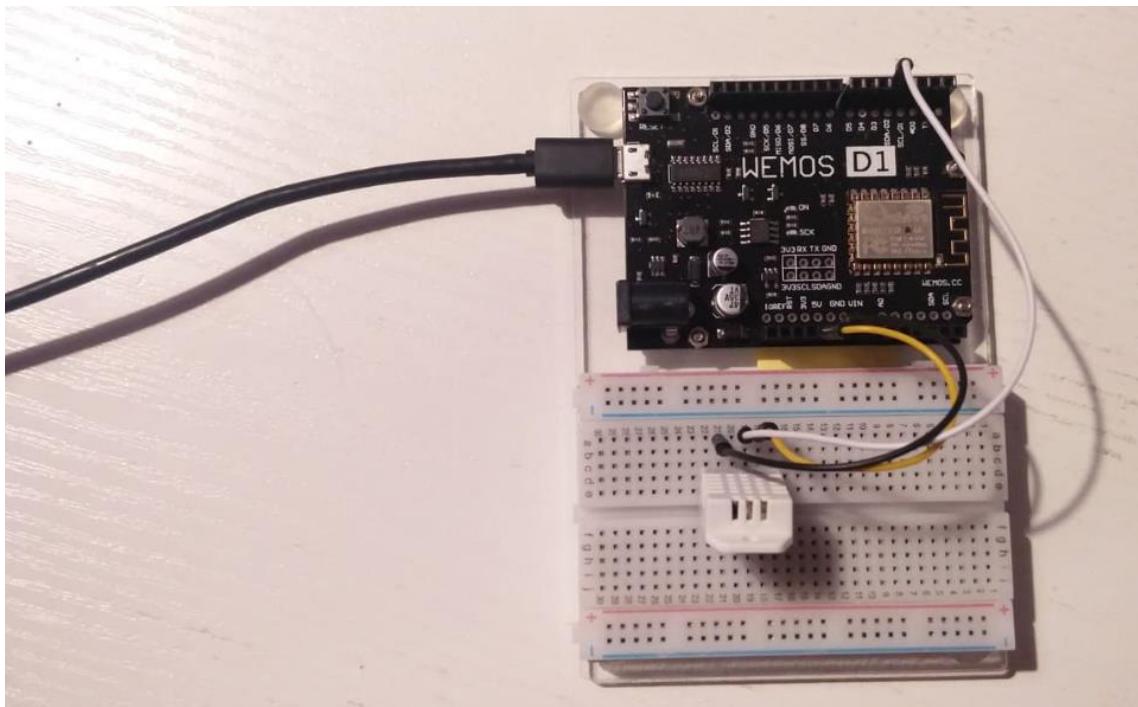
Όσο αφορά τον αισθητήρα θερμοκρασίας & υγρασίας DHT22/AM2302, δεν υπάρχουν τροποποιήσεις σε ότι αφορά τη συνδεσμολογία ή τον κώδικα καθώς οι βιβλιοθήκες που χρησιμοποιούνται είναι κοινές για τους δύο μικροελεγκτές.

Στο παρακάτω Σχήμα φαίνεται το ηλεκτρολογικό σχέδιο σύνδεσης του δεύτερου υποσυστήματος. (Λόγω τεχνικών προβλημάτων στο πρόγραμμα που σχεδιάστηκαν οι απεικονίσεις των συστημάτων, έχει γίνει η αλλαγή του άσπρου καλωδίου με μπλε καλώδιο.)



Σχήμα 3.20 Ηλεκτρολογική σχηματική απεικόνιση συνδεσμολογίας δεύτερου υποσυστήματος

Τέλος, είναι απράίτητη η σύνδεση του μικροελεγκτή WeMos D1 R2 με τον H/Y, μέσω ενός micro USB καλωδίου, τόσο για τροφοδοσία όσο και για την μεταφορά και συλλογή των δεδομένων, και το ανέβασμα του κώδικα στην πλακέτα. Στο Σχήμα 3.20, παρουσιάζεται η τελική μορφή της συνδεσμολογίας που περιγράφηκε.



Σχήμα 3.21 Τελική μορφή δεύτερου υποσυστήματος

3.3 Σύνδεση Υλικού με Τοπικό Εξυπηρετητή

Καθώς έχουν ολοκληρωθεί η συνδεσμολογία στα δύο υποσυστήματα για τη συλλογή και αποστολή των δεδομένων, σειρά έχει η σύνδεση του υλικού με τον τοπικό εξυπηρετητή καθώς και ο προγραμματισμός των υποσυστημάτων για την εύρυθμη αποστολή των δεδομένων που συλλέγονται. Στην παρούσα διπλωματική εργασία, αξίζει να σημειωθεί πώς οι μικροελεγκτές καθώς και ο τοπικός εξυπηρετητής έχουν το δικό του αναγνωριστικό συσκευής MAC Address, κάτι το οποίο έπαιξε σημαντικό ρόλο στον διαχωρισμό του κώδικα, αλλά και στον τρόπο με τον οποίο μεταφέρονται τα δεδομένα στη διαδικτυακή εφαρμογή. Όπως αναφέραμε παραπάνω, το λογισμικό που χρησιμοποιήθηκε για τη συγγραφή του κώδικα που ‘ανέβηκε’ στους μικροελεγκτές είναι το Arduino IDE και το λειτουργικό σύστημα στο οποίο βασίστηκε η συγγραφή της παρούσας διπλωματικής εργασίας είναι τα Windows. Παρακάτω γίνεται μία ανάλυση σε ότι αφορά τον προγραμματισμό των μικροελεγκτών ώστε να γίνει η σωστή αποστολή των δεδομένων στις βάσεις δεδομένων καθώς και η σωστή - ασφαλή δρομολόγηση των δεδομένων μέσω του router στον τελικό τους προορισμό που είναι ο τοπικός εξυπηρετητής (Windows Server).

3.3.1. Προγραμματισμός Αισθητηρίων για Λήψη Δεδομένων

Ξεκινώντας με τον προγραμματισμό των αισθητηρίων, πρέπει να αναφέρουμε πώς υπάρχουν διαφορές στον προγραμματισμό τους, καθώς αναφερόμαστε σε διαφορετικό μικροελεγκτή σε κάθε υποσύστημα. Οι βασικές βιβλιοθήκες παραμένουν ίδιες, αλλάζει όμως μερικώς, ο τρόπος γραφής ορισμένων μεταβλητών όπως είναι τα pins στα οποία είναι τοποθετημένοι οι ακροδέκτες για τα δεδομένα από τους αισθητήρες. Επίσης οι βιβλιοθήκες για τον τρόπο αποστολής των δεδομένων διαφέρουν, καθώς στο υποσύστημα με το μικροελεγκτή Arduino Uno Rev3 η μεταφορά δεδομένων γίνεται ενσύρματα μέσω καλωδίου ethernet και στο υποσύστημα με το μικροελεγκτή WeMos D1 R2 η μεταφορά δεδομένων γίνεται ασύρματα μέσω πρωτόκολλου WiFi. Το αρχεία που περιέχουν τον κώδικα για κάθε μικροελεγκτή είναι διαφορετικά και έχουν κατάληξη .ino έτσι ώστε να μπορούν να ανοίξουν με το λογισμικό Arduino IDE. Παρακάτω γίνεται μία πιο αναλυτική παρουσίαση των δύο αυτών αρχείων καθώς και των διαφορών που έχουν στην σύνταξη του πηγαίου κώδικα.

Υποσύστημα με μικροελεγκτή Arduino Uno Rev3

Για να παραμετροποιήσουμε το περιβάλλον εργασίας ώστε να γράψουμε τον πηγαίο κώδικα για το μικροελεγκτή Arduino Uno Rev3 δημιουργούμε ένα νέο sketch στο Arduino IDE και έπειτα το αποθηκεύουμε ως `indoor_sensor_data.ino`. Επίσης είναι απαραίτητο να επιλεγεί η πλακέτα Arduino και η θύρα σύνδεσής της με το περιβάλλον προγραμματισμού. Τα βήματα που ακολουθούνται για την επιλογή πλακέτας Arduino είναι: Tools → Board → και επιλογή της πλακέτας Arduino Uno Rev3 που χρησιμοποιείται. Τα βήματα που ακολουθούνται για την επιλογή θύρας είναι: Tools →

Serial Port → και επιλογή της σειριακής θύρας (συνήθως COM#) ή θύρας USB που είναι συνδεδεμένο το Arduino.

Ξεκινώντας τον κώδικα, στην αρχή του προγράμματος, γίνονται οι δηλώσεις βιβλιοθηκών, μεταβλητών και έπειτα αυτό χωρίζεται σε δύο απλές συναρτήσεις. Η void `setup()` συνάρτηση χρησιμοποιείται για την αρχικοποίηση της Ethernet σύνδεσης. Τέλος, η void `loop()`, η οποία πραγματοποιεί συνεχείς επαναλήψεις στον κώδικα που έχει αυτή, δίνοντας τη δυνατότητα στο πρόγραμμα να εκτελείται συνεχώς και να στέλνει ή να λαμβάνει δεδομένα. Εκτελείται συνεχώς και καλείται αφού έχει πρώτα κληθεί και ολοκληρωθεί η void `setup()`.

Για τη συγγραφή του αρχείου `indoor_sensor_data.ino` είναι αναγκαία η χρήση των βιβλιοθηκών `<SPI.h>` για την επικοινωνία Arduino-Shield, `<Ethernet.h>` για τη χρήση της πλακέτας Ethernet Shield και `<DHT.h>` για τη χρήση αισθητήρα.

- `#include <SPI.h>`
- `#include <Ethernet.h>`
- `#include <dht.h>`

Με τη λέξη `#include` εισάγεται κάθε βιβλιοθήκη, τιθέμενη σε εισαγωγικά ή αποσιωπητικά όταν αυτή βρίσκεται σε αρχείο. Οι βιβλιοθήκες είτε παρέχονται από το λογισμικό για τις πιο γνωστές λειτουργίες, είτε υπάρχουν στο Διαδίκτυο όπως η `<dht.h>` η λήψη της οποίας έγινε δωρεάν.

Η διασύνδεση του αισθητήρα DHT22/AM2302 γίνεται μέσω της βιβλιοθήκης “`DHT.h`” Στην ψηφιακή είσοδο (pin 2) εισέρχεται τάση η οποία έχει σταθμιστεί ανάλογα με τις τιμές θερμοκρασίας/υγρασίας του περιβάλλοντος και μετατρέπεται σε τιμή η οποία αντιστοιχεί στη θερμοκρασία/υγρασία. Η μετατροπή αυτή επιτυγχάνεται με τη χρήση της βιβλιοθήκης [23].

Με βάση το datasheet της εταιρίας κατασκευής ο αισθητήρας αναγνωρίζει τρεις συναρτήσεις:

1. `readData()`
2. `readHumidity()`
3. `readTemperature()`

Η `readData()`, όπου επιστρέφει την κατάσταση του αισθητήρα. Κατά την επιστροφή,

έχουν οριστεί οι παρακάτω τιμές από την βιβλιοθήκη *<dht.h>*:

- **DHT_ERROR_NONE**: Εάν δεν υπάρχουν σφάλματα
- **DHT_ERROR_CHECKSUM**: Σφάλμα κατά τον έλεγχο με το Bit ισοτιμίας
- **DHT_BUS_HUNG**: Σφάλμα
- **DHT_ERROR_NOT_PRESENT**: Δεν εντοπίστηκε ο αισθητήρας
- **DHT_ERROR_ACK_TIMEOUT**: Το σήμα ACK έληξε
- **DHT_ERROR_SYNC_TIMEOUT**: Σφάλμα κατά το συγχρονισμό
- **DHT_ERROR_DATA_TIMEOUT**: Σφάλμα δεδομένων
- **DHT_ERROR_TOOQUICK**: Ζητήθηκε γρηγορότερα από τον αναμενόμενο χρόνο (< 2 seconds) αίτημα για νέα δεδομένα

Η *readHumidity()*, επιστρέφει την τρέχουσα τιμή του αισθητήρα για την υγρασία σε ποσοστό %

Η *readTemperature()*, επιστρέφει την τρέχουσα τιμή της θερμοκρασίας σε βαθμούς Κελσίου

Στο τμήμα των κώδικα χρησιμοποιούνται οι *readHumidity()* και *readTemperature()* για τη λήψη δεδομένων ως εξής:

```
Void loop () {  
    int readData = DHT.read22(dataPin); // Επιστρέφεται η κατάσταση  
    του αισθητήρα  
    float t = DHT.temperature; // Λαμβάνουμε την τιμή της  
    θερμοκρασίας  
    float h = DHT.humidity; // Λαμβάνουμε την τιμή της υγρασίας
```

Μετά τη δήλωση των βιβλιοθηκών ακολουθεί η δήλωση και αρχικοποίηση των σταθερών μεταβλητών, η τιμή των οποίων δεν μπορεί να αλλάξει μέσα στο τμήμα κώδικα που ακολουθεί.

```

#define datapin 2 // ακροδέκτης για την σύνδεση του αισθητήρα και
μεταφορά των δεδομένων
#define DHTTYPE DHT22 // DHT 22 (AM2302)
DHT dht(DHTPIN, DHTTYPE); // Προετοιμασία αισθητήρα για 16mhz
Arduino

```

Οι μεταβλητές τ και h έχουν δηλωθεί στο αρχικό τμήμα κώδικα μαζί με τον domain της Διαδικτυακής εφαρμογής και τον κωδικό-όνομα της συσκευής στην οποία θα μεταφερθεί ο κώδικας.

```

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
byte ip[] = {192, 168, 1, 76 }; // Ανάθεση στατικής τοπικής IP
byte serv[] = {192, 168, 1, 100} ; // Πληροφορία για την στατική
IP του τοπικού εξυπηρετητή
EthernetClient cliente; // Δημιουργία κλάσης client

```

Η μεταφορά των ληφθέντων τιμών γίνεται με τη χρήση της βιβλιοθήκης Ethernet.h. Αρχικά στο τμήμα του κώδικα με τη δήλωση των μεταβλητών, πραγματοποιείται η δημιουργία μιας κλάσης cliente. Μέσα στη συνάρτηση loop() μέσω της cliente εκτελούνται όλες οι συναρτήσεις και εντολές για τη δημιουργία διασύνδεσης της εκάστοτε συσκευής μετρήσεων με το Διαδίκτυο και κατά συνέπεια τη βάση και τη διαδικτυακή εφαρμογή. Η cliente.connect(server, 8080) συνδέεται με τη διεύθυνση IP και τη θύρα που καθορίστηκε κατά τη δημιουργία. Έπειτα με την μέθοδο cliente.print() δημιουργείται ένα αίτημα GET από τον τοπικό εξυπηρετητή και έτσι οι τιμές θερμοκρασίας και υγρασίας περνάνε στη βάση δεδομένων μέσω του αρχείου data.php με ένα query που δημιουργείται και οι τιμές εισάγονται υπό μορφή JSON Format.

Στην void loop() εκτός από τη λήψη δεδομένων και τη διαχείρισή τους, ορίζεται ταυτόχρονα και ο ρυθμός επαναληψιμότητας της. Στη συγκεκριμένη εφαρμογή έχει επιλεγεί η συλλογή δεδομένων από τον αισθητήρα να γίνεται κάθε πέντε λεπτά.

Μόλις ολοκληρωθεί και ο έλεγχος για τυχόν σφάλματα και συντακτικά λάθη στον κώδικα, μέσω του εργαλείου “upload” γίνεται η μεταγλώττιση και φόρτωση του στην πλακέτα και η συσκευή άμεσα ξεκινά την καταγραφή των δεδομένων. Η τελική μορφή του αρχείου indoor_sensor_data.ino βρίσκεται στο Παράρτημα B.

Υποσύστημα με μικροελεγκτή WeMos D1 R2

Για να παραμετροποιήσουμε το περιβάλλον εργασίας ώστε να γράψουμε τον πηγαίο κώδικα για το μικροελεγκτή WeMos D1 R2 δημιουργούμε ένα νέο sketch στο Arduino

IDE και έπειτα το αποθηκεύουμε ως outdoor_sensor_data.ino. Έπειτα η διαδικασία τροποιείται σε σχέση με το μικροελεγκτή Arduino Uno Rev3 καθώς πρέπει να εγκαταστήσουμε κάποια πακέτα ώστε να υπάρχει υποστήριξη από το Arduino IDE για το μικροελεγκτή WeMos D1 R2. Συγκεκριμένα, τα βήματα είναι: File → Preferences και έτσι μας ανοίγει ένα παράθυρο με τις προτιμήσεις. Στην καρτέλα Settings και στο πεδίο Additional Boards Manager URLs τοποθετούμε τον παρακάτω σύνδεσμο http://arduino.esp8266.com/stable/package_esp8266com_index.json και πατάμε OK. Μετά, πηγαίνουμε Tools → Board → Boards Manager και στο πεδίο αναζήτησης από το παράθυρο που έχει ανοίξει αναζητούμε την λέξη esp8266. Εγκαθιστούμε τα πακέτα που μας έχει προβάλει και πλέον είμαστε έτοιμοι να χρησιμοποιήσουμε το WeMos D1 R2. Τα βήματα που ακολουθούνται για την επιλογή πλακέτας Arduino είναι: Tools → Board → και επιλογή της πλακέτας WeMos D1 R2 & mini που χρησιμοποιείται. Τα βήματα που ακολουθούνται για την επιλογή θύρας είναι: Tools → Serial Port → και επιλογή της σειριακής θύρας (συνήθως COM#) ή θύρας USB που είναι συνδεδεμένο το WeMos D1 R2. Τελευταίο βήμα είναι να αλλάξουμε την ταχύτητα ανεβάσματος, έτσι πηγαίνουμε Tools → Upload Speed → και επιλέγουμε ταχύτητα 115200 διότι έτσι ορίζει ο κατασκευαστής της πλακέτας.

Για τη συγγραφή του αρχείου outdoor_sensor_data.ino είναι αναγκαία η χρήση των βιβλιοθηκών, `<ESP8266WiFi.h>` - `<WiFiClient.h>` - `<ESP8266WebServer.h>` - `<ESP8266mDNS.h>` για την αρχικοποίηση του ESP8266 module και για την χρήση του ως τοπικό υποσύστημα, αλλά και τις βιβλιοθήκες `<DHT.h>` - `<Adafruit_Sensor.h>` για τη χρήση του αισθητήρα.

- `#include <ESP8266WiFi.h>`
- `#include <WiFiClient.h>`
- `#include <ESP8266WebServer.h>`
- `#include <ESP8266mDNS.h>`
- `#include <DHT.h>`
- `#include <Adafruit_Sensor.h>`

Μετά τη δήλωση των βιβλιοθηκών ακολουθεί η δήλωση και αρχικοποίηση των σταθερών μεταβλητών, των οποίων η τιμή δεν μπορεί να αλλάξει μέσα στο τμήμα κώδικα που ακολουθεί.

```
#define DHTPIN D2 // ακροδέκτης για την σύνδεση του αισθητήρα και μεταφορά των δεδομένων
#define DHTTYPE DHT22 // DHT 22 (AM2302), AM2321
DHT dht(DHTPIN, DHTTYPE); //Προετοιμασία αισθητήρα για WeMos D1 R2
const char* ssid = "SmartHomie"; // Απόδοση ονόματος τοπικού δικτύου WiFi (WiFi SSID)
```

```
const char* password = "!smarthomie10!"; // Απόδοση τοπικού  
password για το δίκτυο WiFi
```

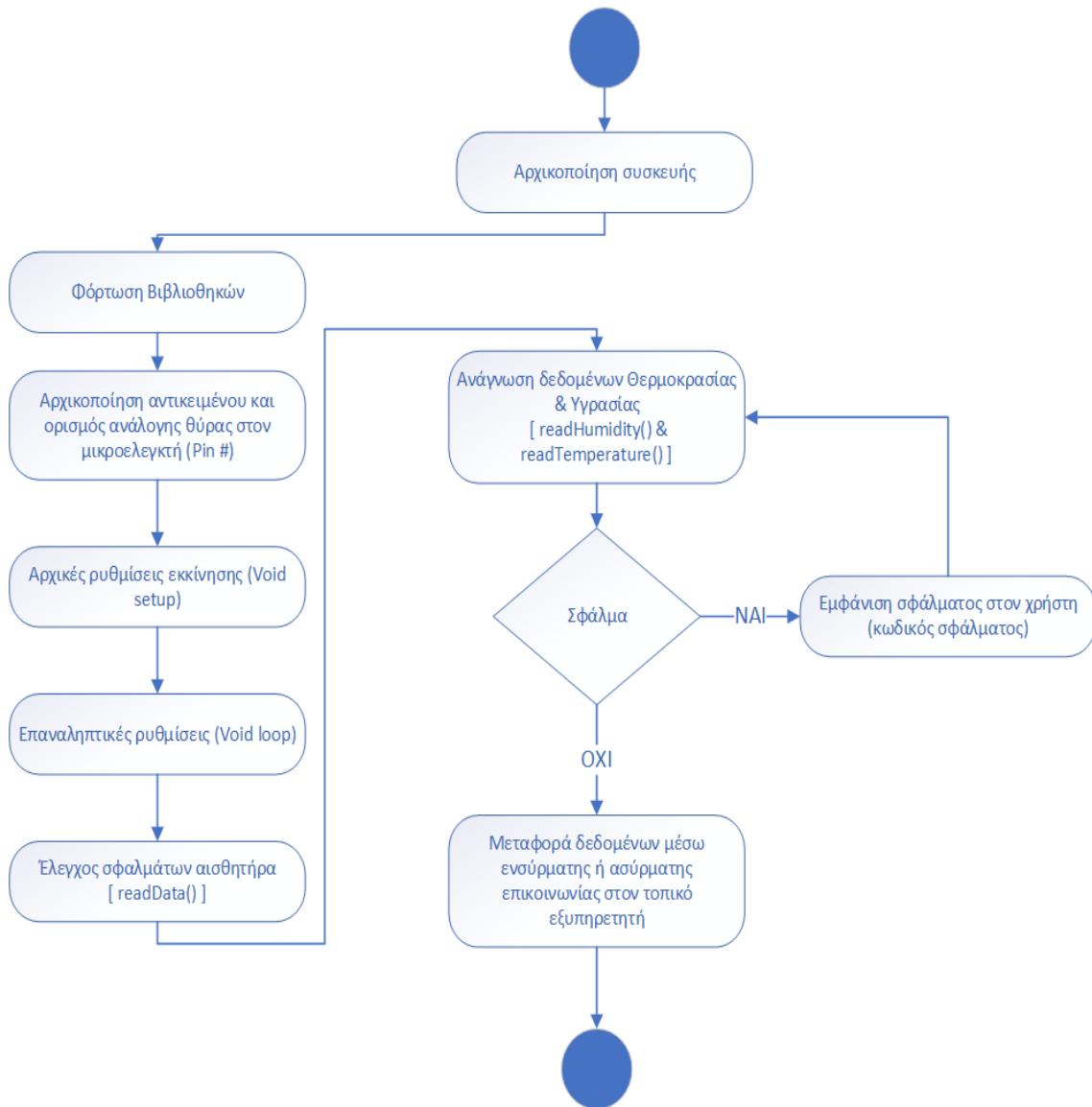
Έπειτα ορίζονται οι στατικές IP για την σωστή δρομολόγηση των πακέτων.

```
IPAddress ip(192, 168, 1, 77); // Ανάθεση τοπικής στατικής IP  
IPAddress gateway(192, 168, 1, 1); // Ανάθεση προεπιλεγμένης  
πύλης  
IPAddress subnet(255, 255, 255, 0); // Ανάθεση τοπικής μάσκας  
δικτύου  
IPAddress dns1(192, 168, 1, 1); //DNS Server 1  
IPAddress dns2(192, 168, 1, 2); //DNS Server 2
```

Όσο αφορά τις ρυθμίσεις του αισθητήρα DHT22/AM2302, δεν υπάρχουν τροποποιήσεις στη λήψη και αποστολή δεδομένων, και η διαδικασία με την μέθοδο GET και τη μορφή JSON Format με οποία αποστέλλονται τα δεδομένα στις βάσεις δεδομένων παραμένει ίδια.

3.3.2. Διάγραμμα Ροής Λειτουργίας Αισθητήρα DHT22/Am2302

Η σωστή λειτουργία του αισθητήρα DHT22/Am2302 βασίζεται τόσο στη σωστή σύνδεσή του με τον μικροελεγκτή όσο και στον ορθό προγραμματισμό του. Η διαδικασία για τον τρόπο λειτουργίας του ως εξάρτημα της συσκευής και βασικό χαρακτηριστικό του κώδικα αναπαριστάται στο διάγραμμα ροής του Σχήματος 3.21.



Σχήμα 3.22 Διάγραμμα ροής λειτουργίας αισθητήρα DHT22/AM2302

3.4 Υπολογισμός Κόστους Εξαρτημάτων

Σ' αυτήν την ενότητα, γίνεται καταγραφή των υλικών μερών που χρησιμοποιήθηκαν για την κατασκευή των υποσυστημάτων καθώς και για το σύνολο τους συστήματος με το οποίο έγινε πραγματικότητα η διαδικτυακή εφαρμογή. Στον παρακάτω πίνακα γίνεται μία ανάλυση κόστους για κάθε τιμή μονάδας προϊόντος, αναλύονται οι ποσότητες και υπολογίζεται το συνολικό κόστος εγκατάστασης των συσκευών.

Πίνακας 3.3: Υπολογισμός κόστους υλικών μερών

Υλικό	Ποσότητα	Τιμή μονάδας (€)
Arduino Uno Rev3 + USB Καλώδιο	1	20,00
WeMos D1 R2 + microUSB καλώδιο	1	12,00
Ethernet Shield W5100	1	16,20
Αισθητήρας DHT22/AM2302	2	7,00
Mini Solderless Breadboard 400	2	3,00
Jumper Wires (65 κομμάτια)	1	3,50
Ethernet RJ45 cable	1	5,00
Ακρυλική βάση	2	2,80
Σύνολο: 79,50€		

Όπως αναφέρθηκε και σε προηγούμενες ενότητες, καθίσταται δυνατό, με τις παραπάνω συσκευές χαμηλού κόστους, να μπορεί κάθε χρήστης να καταγράφει και να παρακολουθεί σε ζωντανό χρόνο την κατάσταση θερμοκρασίας και υγρασίας καθώς και να εναλλάσσει τις καταστάσεις στις έξυπνες συσκευές στο σπίτι του.

Κεφάλαιο 4

Σχεδίαση και Υλοποίηση Διαδικτυακής Εφαρμογής

Έχοντας ολοκληρώσει τη συνδεσμολογία των συστημάτων, το τελικό στάδιο είναι η σχεδίαση και δημιουργία της διαδικτυακής εφαρμογής και παράλληλα η σχεδίαση των βάσεων στις οποίες θα αποθηκεύονται όλα τα απαραίτητα δεδομένα και πληροφορίες οι οποίες εμφανίζονται στο χρήστη. Με την παρούσα διαδικτυακή εφαρμογή, ο χρήστης μπορεί να παρακολουθεί, να αναλύει, αλλά και να κάνει αλλαγές στην υπάρχουσα κατάσταση του χώρου τον οποίο έχει επιλέξει. Η προβολή και παρακολούθηση αυτών των δεδομένων επιτυγχάνεται μέσω της διαδικτυακής εφαρμογής Smart Homie.

Στις ενότητες που ακολουθούν παρουσιάζονται όλες οι απαιτήσεις του συστήματος που κατασκευάστηκε. Περιγράφεται η διαδικασία σχεδίασης και δημιουργίας των βάσεων καθώς και οι λειτουργίες της διαδικτυακής εφαρμογής, που στηρίχτηκαν στις απαιτήσεις του συστήματος. Τέλος, αναλύεται η διαδικασία και ο πηγαίος κώδικας που αναπτύχθηκε για τη δημιουργία της διαδικτυακής εφαρμογής και την επικοινωνία της με τη βάση.

4.1 Απαιτήσεις Συστήματος

Βασικός στόχος της παρούσας διπλωματικής εργασίας είναι η υλοποίηση του συστήματος καταγραφής και παρουσίασης δεδομένων θερμοκρασίας και υγρασίας. Το σύστημα αυτό αποτελείται τόσο από τα υποσυστήματα για τον εσωτερικό και εξωτερικό χώρο, όσο και από τη διαδικτυακή εφαρμογή Smart Homie. Έτσι λοιπόν, κρίνεται αναγκαίο να διαμορφωθούν κάποιες απαιτήσεις από το σύστημα για την αποτελεσματική λειτουργία όλων των χαρακτηριστικών που έχουν αναπτυχθεί. Παρακάτω παρουσιάζεται μία λίστα με τις απαιτήσεις που υπάρχουν από την διαδικτυακή εφαρμογή.

1. Για την σωστή καταγραφή της δραστηριότητας αλλά και για την ασφάλεια της διαδυνκτιακής εφαρμογής είναι απαραίτητη η εγγραφή κάποιου χρήστη κατά την εκκίνηση της διαδικτυακή εφαρμογής για πρώτη φορά. Όταν τοποθετηθεί το σύστημα για πρώτη φορά σε κάποιον οικιακό χώρο, ο πρώτος χρήστης λαμβάνει δικαιώματα διαχειριστή και έπειτα μπορεί να προσθέσει νέους διαχειριστές/χρήστες, ανάλογα με τα δικαιώματα που θέλει να δώσει.
2. Κάθε υποσύστημα καταγραφής, απαιτεί την ύπαρξη του κατάλληλου πηγαίου κώδικα με σκοπό την επικοινωνία του με τον τοπικό εξυπηρετητή. Με αυτόν τον τρόπο, καθορίζεται και η τοποθεσία στην οποία θα εγκατασταθεί, είτε σε εξωτερικό είτε σε εσωτερικό χώρου του σπιτιού. Ακόμη, πρέπει να τοποθετηθεί το ανάλογο αρχείο .ino, ανάλογα με τον τρόπο αποστολής των δεδομένων (ασύρματα ή ενσύρματα).
3. Το σύστημα πρέπει να προσφέρει εξασφάλιση γρήγορης και ακίνδυνης σύνδεσης στην εφαρμογή από τον χρήστη. Κάθε χρήστης κατά την εγγραφή του πληκτρολογεί έναν προσωπικό κωδικό πρόσβασης. Ο κρυπτογραφημένος αυτός κωδικός αποθηκεύεται στη βάση με τη χρήση της εντολής του κώδικα.
4. Το περιβάλλον την εφαρμογής πρέπει να είναι απλό, ξεκάθαρο και φιλικό προς τον χρήστη. Η διαδικτυακή εφαρμογή ενδέχεται να χρησιμοποιηθεί από άτομα με έλλειψη τεχνολογικής εκπαίδευσης. Ο σχεδιασμός της πρέπει να είναι συμβατός με την εύκολη και ομαλή περιήγηση του κάθε χρήστη μέσα σε αυτή.
5. Οι χρήστες πρέπει να μπορούν να επεξεργάζονται τα στοιχεία των προφίλ τους. Κάθε χρήστης θα πρέπει να έχει τη δυνατότητα να τροποποιήσει τα προσωπικά του στοιχεία οποιαδήποτε στιγμή αυτός επιθυμεί (π.χ. αλλαγή κωδικού πρόσβασης για καλύτερη ασφάλεια, αλλαγή του email τους σε περίπτωση που έχουν αλλάξει).
6. Οι χρήστες πρέπει να μπορούν να ενημερώνονται για την κατάσταση των συσκευών σε πραγματικό χρόνο. Πέρα από την καταγραφή των δεδομένων που συλλέγουν οι συσκευές, ο κάθε χρήστης θα πρέπει να μπορεί να παρακολουθεί τις τρέχουσες τιμές υγρασίας και θερμοκρασίας του χώρου που έχει επιλέξει όποια στιγμή επισκεφτεί τη διαδικτυακή εφαρμογή.
7. Οι χρήστες πρέπει να μπορούν να αποθηκεύουν αρχεία από την εφαρμογή. Εκτός του αρχείου καταγραφής δεδομένων στην εφαρμογή ο χρήστης θα μπορεί να αποθηκεύει και να εκτυπώνει τα διαγράμματα που παρακολουθεί στην εφαρμογή.

4.2 Διαδικτυακή Ασφάλεια & Προστασία Προσωπικών Δεδομένων

Στις μέρες μας, μεγάλο μέρος των προσωπικών δεδομένων των ανθρώπων βρίσκεται στο Διαδίκτυο. Άλλοτε γίνεται με τη θέληση τους και άλλοτε εν αγνοίᾳ τους. Προσωπικά δεδομένα είναι κάθε πληροφορία που αναφέρεται στο πρόσωπο του κάθε ατόμου, όπως: το όνομα και το επάγγελμά του, η οικογενειακή του κατάσταση, η ηλικία του, ο τόπος κατοικίας, η φυλετική του προέλευση, τα πολιτικά του φρονήματα, η θρησκεία που πιστεύει, οι φιλοσοφικές του απόψεις, η συνδικαλιστική του δράση, η υγεία του, η ερωτική του ζωή και οι τυχόν ποινικές του διώξεις και καταδίκες. Δεν θεωρούνται προσωπικά δεδομένα πληροφορίες από τις οποίες δεν δύναται να ταυτοποιηθεί ένα συγκεκριμένο άτομο [24].

Σημαντικό μέρος της παρούσας διπλωματικής εργασίας αποτέλεσε ο τρόπος με τον οποίο συλλέγονται τα δεδομένα από τους αισθητήρες, ο τρόπος αποθήκευσής τους στη βάση δεδομένων καθώς και η εμφάνισή τους στον τελικό χρήστη. Ακολουθήθηκαν κάποια βήματα για την ασφαλέστερη λειτουργία της διαδικτυακής εφαρμογής και τα δεδομένα του χρήστη τα οποία είναι:

- **Η διαδικτυακή εφαρμογή δουλεύει σε τοπικό δίκτυο:** Πρακτικά, αυτό σημαίνει ότι ο χρήστης δεν μπορεί κατά την διάρκεια της αρχικής εγκατάστασης να εισέλθει στην εφαρμογή από το Διαδίκτυο. Αυτό μειώνει τον κίνδυνο απώλειας δεδομένων κατά ένα μεγάλο βαθμό, καθώς είναι πρακτικά αδύνατο κάποιος ο οποίος δεν βρίσκεται στο τοπικό δίκτυο της εφαρμογής να υποκλέψει τα δεδομένα.
- **Η χρήση της διαδικτυακής εφαρμογής μπορεί να γίνει μέσω δικτύου:** Με την ως άνω λειτουργία της εφαρμογής ο χρήστης θα πρέπει να βρίσκεται σε μία απόσταση, ώστε να μπορεί να συνδεθεί στο τοπικό δίκτυο με WiFi είτε με καλώδιο Ethernet. Έτσι, η λειτουργικότητα της εφαρμογής δυσκολεύει σε ορισμένες περιπτώσεις τον χρήστη. Λύση σε αυτό το πρόβλημα, έρχεται να δώσει η δυνατότητα του δρομολογητή, να ανοίξει μία διαδικτυακή θύρα στην οποία θα γίνει αντιστοίχιση του τοπικού εξυπηρετητή και έτσι εφόσον ο χρήστης έχει ορίσει το όνομα τομέα (domain name) που αυτός θέλει, καθώς και τη θύρα στο δρομολογητή, θα μπορεί να χρησιμοποιήσει την εφαρμογή από οπουδήποτε μέσω Διαδικτύου. Για τις ανάγκες της παρούσας διπλωματικής εργασίας χρησιμοποιήθηκε ένα τοπικό όνομα τομέα <https://smarthomie.local/> καθώς και η θύρα 61457 στον δρομολογητή.
- **Κατά την εγγραφή του χρήστη, ο κωδικός αποστέλλεται κρυπτογραφημένος:** Η φόρμα εγγραφής αποστέλλει τα στοιχεία του χρήστη στη βάση δεδομένων. Πριν την αποστολή ο κωδικός που έχει εισάγει ο χρήστης μεταβιβάζεται σε μία συνάρτηση της PHP, την `password_hash()`, με τη βοήθεια της οποίας δημιουργείται ένα αλγόριθμος κρυπτογράφησης και έτσι ο κωδικός αποστέλλεται κρυπτογραφημένος [25].

- **Στον τοπικό εξυπηρετητή έχει ενεργοποιηθεί το πρωτόκολλο SSL:** Το πρωτόκολλο SSL (Secure Sockets Layer) αναπτύχθηκε από την εταιρεία Netscape και σχεδιάστηκε για να παρέχει ασφάλεια κατά τη μετάδοση εναίσθητων δεδομένων στο Διαδίκτυο. Το SSL χρησιμοποιεί μεθόδους κρυπτογράφησης των δεδομένων που ανταλλάσσονται μεταξύ δύο συσκευών (συνηθέστερα HY) εγκαθιδρύοντας μία ασφαλή σύνδεση μεταξύ τους μέσω του Διαδικτύου. Το πρωτόκολλο αυτό χρησιμοποιεί το TCP/IP για τη μεταφορά των δεδομένων και είναι ανεξάρτητο από την εφαρμογή που χρησιμοποιεί ο τελικός χρήστης. Για τον λόγο αυτό μπορεί να παρέχει υπηρεσίες ασφαλούς μετάδοσης πληροφοριών σε πρωτόκολλα ανώτερου επιπέδου όπως για παράδειγμα το HTTP [26]. Ετσι για την ασφαλέστερη μεταφορά των δεδομένων προς τον χρήστη, στον πίνακα ελέγχου του XAMPP έχει ενεργοποιηθεί το πρωτόκολλο SSL και έχει ανοιχθεί η απαραίτητη 443 θύρα στον δρομολογητή.

4.3 Δημιουργία και Περιγραφή της Βάσης Δεδομένων

Η βάση δεδομένων που δημιουργήθηκε για τη διαδικτυακή εφαρμογή Smart Homie ονομάστηκε smart_homie και αποτελείται από πέντε πίνακες οι οποίοι δημιουργήθηκαν για να υποστηρίζουν τις απαιτήσεις του συστήματος. Ο σχεδιασμός της έγινε έτσι ώστε να μπορούν να εξυπηρετούνται πολλοί χρήστες με μία ή και περισσότερες συσκευές στην κατοχή τους ταυτόχρονα. Αφού γίνει περιγραφή των πινάκων και των χαρακτηριστικών τους πρέπει να καθοριστούν και οι συσχετίσεις μεταξύ αυτών. Οι πίνακες παρατίθενται στο τέλος της συγκεκριμένης ενότητας.

4.3.1. Εργαλείο Δημιουργίας Βάσεων Δεδομένων - phpMyAdmin

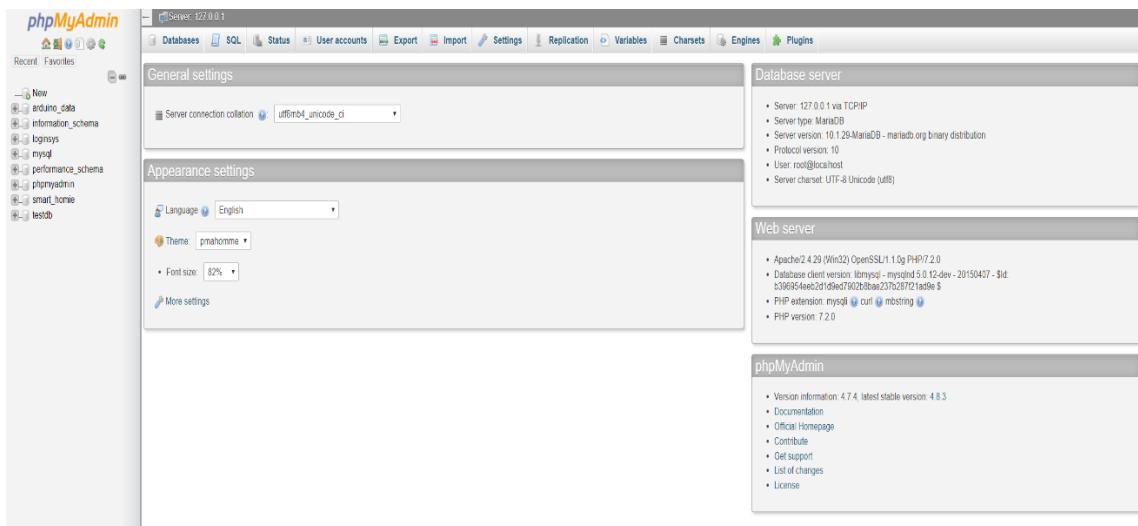
Για την επεξεργασία των βάσεων δεδομένων είναι απαραίτητη η χρήση ενός εργαλείου διαχείρισης βάσεων δεδομένων. Στην παρούσα διπλωματική εργασία επιλέχθηκε το phpMyAdmin, (μια εφαρμογή) που δημιουργήθηκε για τη διαχείριση εξυπηρετητών MySQL. Το εργαλείο phpMyAdmin έρχεται προεγκατεστήμενο στον πίνακα ελέγχου του XAMPP και έτσι καθίσταται αρκετά εύκολη η πρόσβασή του. Το εργαλείο είναι ικανό να διαχειρίζεται από απλές βάσεις δεδομένων μέχρι έναν ολόκληρο mysql server. Είναι εύκολο και κατανοητό στη χρήση και παράλληλα πάροχος πολλών δυνατοτήτων για κάθε χρήστη.

Οι δυνατότητες που προσφέρει το phpMyAdmin, το καθιστούν ένα δυνατό εργαλείο έναντι άλλων, ανταγωνιστικών εργαλείων. Παρακάτω αναλύονται μερικές από τις δυνατότητες που έχει αυτό το εργαλείο.

- Δημιουργεί και διαγράφει βάσεις δεδομένων
- Δημιουργεί, τροποποιεί, διαγράφει, αντιγράφει και μετονομάζει πίνακες
- Κάνει συντήρηση της βάσης

- Προσθέτει, διαγράφει και τροποποιεί πεδία πινάκων
- Εκτελεί ερωτήματα SQL ακόμα και ομαδικά (batch)
- Διαχειρίζεται κλειδιά σε πεδία
- Φορτώνει αρχεία κειμένου σε πίνακες
- Δημιουργεί και διαβάζει πίνακες (που προέρχονται από dump βάσεις)
- Εξάγει δεδομένα σε μορφή CSV, XML και LATEX
- Εισάγει δεδομένα και δομές MySQL από CSV, XML και SQL αρχεία
- Διαχειρίζεται πολλούς διακομιστές
- Προσθέτει, επεξεργάζεται και μετακινεί τους χρήστες MySQL και τα δικαιώματά τους
- Ελέγχει την αναφορική δραστηριότητα των δεδομένων των MyISAM πινάκων
- Δημιουργεί αυτόματα περίπλοκα ερωτήματα για τη σύνδεση των πινάκων χρησιμοποιώντας QBE(Query-by-example)
- Δημιουργεί PDF γραφικών του layout της βάσης δεδομένων
- Εκτελεί αναζητήσεις σε όλη τη βάση δεδομένων ή σε μέρος αυτής
- Εντοπίζει και επισημάνει τυχόν αλλαγές στις βάσεις και τους πίνακες
- Υποστηρίζει πίνακες InnoDB και ξένα κλειδιά
- Υποστηρίζει MySQLi, μια βελτιωμένη επέκταση του MySQL PhpMyAdmin
- Δημιουργεί, επεξεργάζεται, καλεί, εξάγει αποθηκευμένες διαδικασίες και συναρτήσεις
- Επικοινωνεί σε 80 διαφορετικές γλώσσες

Η διαδικασία την οποία ακολουθούμε για να δούμε το γραφικό περιβάλλον αυτού του εργαλείου είναι απλή. Χρειάζεται απλά να γράψουμε το τοπικό όνομα τομέα <http://localhost:8080/phpMyAdmin> το οποίο έχει παραχθεί από τις ρυθμίσεις που χρησιμοποιούμε στο XAMPP και το αποτέλεσμα φαίνεται στο Σχήμα 4.1.



Σχήμα 4.1 Γραφικό περιβάλλον phpMyAdmin

4.3.2. Αρχιτεκτονική Βάσης Δεδομένων Διαδικτυακής Εφαρμογής

Όλες οι βάσεις δεδομένων αποτελούνται από πίνακες όπου αποθηκεύονται τα δεδομένα. Με αυτό τον τρόπο επιτυγχάνεται ένας διαχωρισμός δεύτερου επιπέδου, καθώς πολλά δεδομένα μπορεί να αναφέρονται στην ίδια εφαρμογή αλλά να χρησιμοποιούνται για διαφορετικό σκοπό. Έτσι για την καλύτερη επεξεργασία των δεδομένων που λαμβάνει η διαδικτυακή εφαρμογή έχουν δημιουργηθεί πέντε πίνακες μέσα στη βάση δεδομένων. Αφού γίνει περιγραφή των πινάκων και των γνωρισμάτων τους πρέπει να καθοριστούν και οι συσχετίσεις μεταξύ αυτών. Οι πίνακες παρατίθενται στο τέλος της συγκεκριμένης ενότητας.

1) Πίνακας Users

Στον Πίνακα 4.1: Πίνακας users, αποθηκεύονται όλες οι πληροφορίες και τα στοιχεία κάθε χρήστη που έχει εγγραφεί στην εφαρμογή και έχει δυνατότητα πρόσβασης σε αυτήν.

Ανάλυση των στοιχείων του πίνακα:

- **id:** Ο αναγνωριστικός αύξων αριθμός του χρήστη στον πίνακα, είναι μοναδικός για κάθε εγγραφή και αποτελεί το πρωτεύον κλειδί του. Ο τύπος του int(11) επιλέχθηκε γιατί η τιμή των αναγνωριστικών αριθμών είναι πάντα ακέραια. Με την ιδιότητα UNSIGNED εξασφαλίζονται 2,147,483,647 διαφορετικοί χρήστες για το σύστημα. Η πρόσθετη επιλογή AUTO_INCREMENT δείχνει ότι ο αριθμός αυτός αυξάνεται αυτόματα.
- **username:** Το όνομα χρήστη με το οποίο, ο χρήστης θα εισέρχεται στην εφαρμογή. Ο τύπος varchar(50) χρησιμοποιείται για να καλυφθεί το μήκος των χαρακτήρων που συμφωνούν με τους κανόνες ασφαλείας.
- **email:** Η διεύθυνση ηλεκτρονικού ταχυδρομείου με την οποία ο χρήστης θα κάνει εγγραφή στην ιστοσελίδα. Ο τύπος varchar(255) χρησιμοποιείται για να καλυφθεί το μέγιστο δυνατό όριο χαρακτήρων που μπορεί να έχει μια ηλεκτρονική διεύθυνση.
- **adminuser:** Το αναγνωριστικό με το οποίο κάποιος χρήστης διαχωρίζεται σε διαχειριστή και απλό χρήστη. Τα δικαιώματα που έχει εξαρτώνται από την τιμή του πεδίου adminuser. Ο τύπος είναι TINYINT(1), η τιμή 1 περιγράφει τον χρήστη ως διαχειριστή και η τιμή 0 ως απλό χρήστη.
- **password:** Ο κωδικός με τον οποίο ενεργοποιείται η εγγραφή και πραγματοποιείται η είσοδος του χρήστη στο σύστημα. Για την ασφάλεια του χρήστη επίσης δόθηκε ο

τύπος varchar(255) έτσι ώστε να καλυφθεί και ο πιο σύνθετος κωδικός του χρήστη. Αξίζει να σημειωθεί ότι ο κωδικός κάθε χρήστη στη βάση αποθηκεύεται σε κρυπτογραφημένη μορφή.

- **created_at:** Η ημερομηνία δημιουργίας του χρήστη. Έχει ως προεπιλογή την αυτόματη λήψη ημερομηνίας με το πέρας της διαδικασίας εγγραφής.

2) Πίνακας users_log

Στον Πίνακα 4.2: Πίνακας users_log, αποθηκεύεται η δραστηριότητα του χρήστη κατά την διάρκεια χρήσης της εφαρμογής.

Ανάλυση των στοιχείων του πίνακα:

- **id:** Είναι ο αναγνωριστικός αύξων αριθμός ο οποίος ξεχωρίζει τις δραστηριότητες μεταξύ τους. Ο τύπος int(11) επιλέχθηκε γιατί η τιμή των αναγνωριστικών αριθμών είναι πάντα ακέραια. Με την ιδιότητα UNSIGNED εξασφαλίζεται ένα μεγάλο πλήθος για καταχωρήσεις διαφορετικών συσκευών. Η πρόσθετη επιλογή AUTO_INCREMENT δείχνει ότι ο αριθμός αυτός αυξάνεται αυτόματα.
- **user_id:** Είναι ο αναγνωριστικός αριθμός που έχει τοποθετηθεί για τον συγκεκριμένο χρήστη στον πίνακα users. Έτσι, μπορεί να γίνει ο διαχωρισμός των δραστηριοτήτων ανά χρήστη. Ο τύπος varchar(20) επιλέχθηκε καθώς το αναγνωριστικό εδώ εξετάζεται ως χαρακτήρας και όχι ως κάποιος αριθμός.
- **date:** Η ημερομηνία στην οποία καταγράφηκε κάποια δραστηριότητα από τον χρήστη. Ο τύπος date επιλέχθηκε καθώς το πεδίο έχει μόνο ημερομηνίες με την μορφή YYYY-MM-DD.
- **time:** Η ώρα στην οποία καταγράφηκε κάποια δραστηριότητα από τον χρήστη. Ο τύπος time επιλέχθηκε καθώς το πεδίο έχει μόνο την ώρα με την μορφή HH-MM-SS.
- **log:** Το μήνυμα με το οποίο υποδεικνύει η διαδικτυακή εφαρμογή στον χρήστη, το είδος της δραστηριότητας που καταγράφηκε. Ο τύπος varchar(100) επιλέχθηκε καθώς το πεδίο θα έχει ως τιμές κάποιο μήνυμα το οποίο δεν πρέπει να υπερβαίνει τους 100 χαρακτήρες.

3) Πίνακας devices_states

Στον Πίνακα 4.3: devices_states καταχωρούνται όλες οι έξυπνες συσκευές του χώρου καθώς και η κατάσταση στην οποία βρίσκονται (ενεργή/ανενεργή κατάσταση)

Ανάλυση των στοιχείων του πίνακα:

- **id:** Είναι ο αναγνωριστικός αύξων αριθμός ο οποίος ξεχωρίζει τις συσκευές μεταξύ τους. Ο τύπος int(11) επιλέχθηκε γιατί η τιμή των αναγνωριστικών αριθμών είναι πάντα ακέραια. Με την ιδιότητα UNSIGNED εξασφαλίζεται ένα μεγάλο πλήθος για καταχωρήσεις διαφορετικών συσκευών. Η πρόσθετη επιλογή AUTO_INCREMENT δείχνει ότι ο αριθμός αυτός αυξάνεται αυτόματα.
- **devicename:** Είναι η περιγραφή που δίνεται για την αναγνώριση της συσκευής και τον χώρο όπου είναι τοποθετημένη. Ο τύπος varchar(20) επιλέχθηκε καθώς η περιγραφή περιέχει χαρακτήρες και δεν πρέπει να υπερβαίνει τους 20 χαρακτήρες.
- **status:** Είναι η κατάσταση στην οποία βρίσκεται η συσκευή. Ο τύπος varchar(20) επιλέχθηκε καθώς η κατάσταση περιέχει χαρακτήρες και δεν πρέπει να υπερβαίνει τους 20 χαρακτήρες.

4) Πίνακας indoor_data

Στον πίνακα 4.4: indoor_data καταχωρούνται οι μετρήσεις θερμοκρασίας και υγρασίας που συλλέγονται από τους αισθητήρες στον εσωτερικό χώρο.

Ανάλυση των στοιχείων του πίνακα:

- **id:** Είναι ο αναγνωριστικός αύξων αριθμός ο οποίος ξεχωρίζει τις καταγραφές μεταξύ τους. Ο τύπος int(11) επιλέχθηκε γιατί η τιμή των αναγνωριστικών αριθμών είναι πάντα ακέραια. Με την ιδιότητα UNSIGNED εξασφαλίζεται ένα μεγάλο πλήθος για καταχωρήσεις διαφορετικών δεδομένων. Η πρόσθετη επιλογή AUTO_INCREMENT δείχνει ότι ο αριθμός αυτός αυξάνεται αυτόματα.
- **date:** Η ημερομηνία στην οποία καταγράφηκαν κάποια δεδομένα από τον αισθητήρα. Ο τύπος date επιλέχθηκε καθώς το πεδίο έχει μόνο ημερομηνίες με την μορφή YYYY-MM-DD.
- **time:** Η ώρα στην οποία καταγράφηκαν κάποια δεδομένα από τον αισθητήρα. Ο τύπος time επιλέχθηκε καθώς το πεδίο έχει μόνο την ώρα με την μορφή HH-MM-SS.

- **temperature:** Η τιμή της θερμοκρασίας που καταγράφηκε από τον αισθητήρα. Ο τύπος varchar(10) επιλέχθηκε καθώς ο αριθμός περιέχει και χαρακτήρες όπως είναι το σύμβολο υποδιαστολής.
- **humidity:** Η τιμή της υγρασίας που καταγράφηκε από τον αισθητήρα. Ο τύπος varchar(10) επιλέχθηκε καθώς ο αριθμός περιέχει και χαρακτήρες όπως είναι το σύμβολο υποδιαστολής.

5) Πίνακας outdoor_data

Στον πίνακα 4.45 outdoor_data καταχωρούνται οι μετρήσεις θερμοκρασίας και υγρασίας που συλλέγονται από τους αισθητήρες στον εξωτερικό χώρο.

Ανάλυση των στοιχείων του πίνακα:

- **id:** Είναι ο αναγνωριστικός αύξων αριθμός ο οποίος ξεχωρίζει τις καταγραφές μεταξύ τους. Ο τύπος int(11) επιλέχθηκε γιατί η τιμή των αναγνωριστικών αριθμών είναι πάντα ακέραια. Με την ιδιότητα UNSIGNED εξασφαλίζεται ένα μεγάλο πλήθος για καταχωρήσεις διαφορετικών δεδομένων. Η πρόσθετη επιλογή AUTO_INCREMENT δείχνει ότι ο αριθμός αυτός αυξάνεται αυτόματα.
- **date:** Η ημερομηνία στην οποία καταγράφηκαν κάποια δεδομένα από τον αισθητήρα. Ο τύπος date επιλέχθηκε καθώς το πεδίο έχει μόνο ημερομηνίες με την μορφή YYYY-MM-DD.
- **time:** Η ώρα στην οποία καταγράφηκαν κάποια δεδομένα από τον αισθητήρα. Ο τύπος time επιλέχθηκε καθώς το πεδίο έχει μόνο την ώρα με την μορφή HH-MM-SS.
- **temperature:** Η τιμή της θερμοκρασίας που καταγράφηκε από τον αισθητήρα. Ο τύπος varchar(10) επιλέχθηκε καθώς ο αριθμός περιέχει και χαρακτήρες όπως είναι το σύμβολο υποδιαστολής.
- **humidity:** Η τιμή της υγρασίας που καταγράφηκε από τον αισθητήρα. Ο τύπος varchar(10) επιλέχθηκε καθώς ο αριθμός περιέχει και χαρακτήρες όπως είναι το σύμβολο υποδιαστολής.

Πίνακας 4.1: Πίνακας Users

#	ΟΝΟΜΑ	ΤΥΠΟΣ	ΣΥΝΘΕΣΗ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΚΕΝΟ	ΠΡΟΕΠΙΛΟΓΗ	ΠΡΟΣΘΕΤΑ
1	id	int(11)	-	-	Όχι	Καμία	AUTO_INCREMENT
2	username	varchar(50)	utf8_general_ci	-	Όχι	Καμία	-
3	email	varchar(50)	utf8_general_ci	-	Όχι	Καμία	-
4	adminuser	TINYINT(1)	utf8_general_ci	-	Ναι	yes	-
5	password	varchar(50)	utf8_general_ci	-	Όχι	Καμία	-
6	created_at	datetime	-	-	Ναι	CURRENT_TIMESTAMP	-

Πίνακας 4.2: Πίνακας user_log

#	ΟΝΟΜΑ	ΤΥΠΟΣ	ΣΥΝΘΕΣΗ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΚΕΝΟ	ΠΡΟΕΠΙΛΟΓΗ	ΠΡΟΣΘΕΤΑ
1	id	int(11)	-	-	Όχι	Καμία	AUTO_INCREMENT
2	user_id	varchar(20)	utf8_general_ci	-	Όχι	Καμία	-
3	date	date	-	-	Ναι	Καμία	-
4	time	time	-	-	Ναι	Καμία	-
5	log	varchar(100)	utf8_general_ci	-	Όχι	Καμία	-

Πίνακας 4.3: Πίνακας devices_states

#	ΟΝΟΜΑ	ΤΥΠΟΣ	ΣΥΝΘΕΣΗ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΚΕΝΟ	ΠΡΟΕΠΙΛΟΓΗ	ΠΡΟΣΘΕΤΑ
1	id	int(11)	-	-	Όχι	Καμία	AUTO_INCREMENT
2	devicename	varchar(20)	utf8_general_ci	-	Όχι	Καμία	-
3	status	varchar(10)	utf8_general_ci	-	Όχι	Καμία	-

Πίνακας 4.4: Πίνακας Indoor_data

#	ΟΝΟΜΑ	ΤΥΠΟΣ	ΣΥΝΘΕΣΗ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	KENO	ΠΡΟΕΠΙΛΟΓΗ	ΠΡΟΣΘΕΤΑ
1	id	int(11)	-	-	Όχι	Καμία	AUTO_INCREMENT
2	date	date	-	-	Ναι	Καμία	-
3	time	time	-	-	Όχι	Καμία	-
4	temperature	varchar(10)	utf8_general_ci	-	Ναι	Καμία	-
5	humidity	varchar(10)	utf8_general_ci	-	Όχι	Καμία	-

Πίνακας 4.5: Πίνακας outdoor_data

#	ΟΝΟΜΑ	ΤΥΠΟΣ	ΣΥΝΘΕΣΗ	ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ	ΚΕΝΟ	ΠΡΟΕΠΙΛΟΓΗ	ΠΡΟΣΘΕΤΑ
1	id	int(11)	-	-	Όχι	Καμία	AUTO_INCREMENT
2	date	date	-	-	Όχι	Καμία	-
3	time	time	-	-	Όχι	Καμία	-
4	temperature	varchar(10)	utf8_general_ci	-	Όχι	Καμία	-
5	humidity	varchar(10)	utf8_general_ci	-	Όχι	Καμία	-

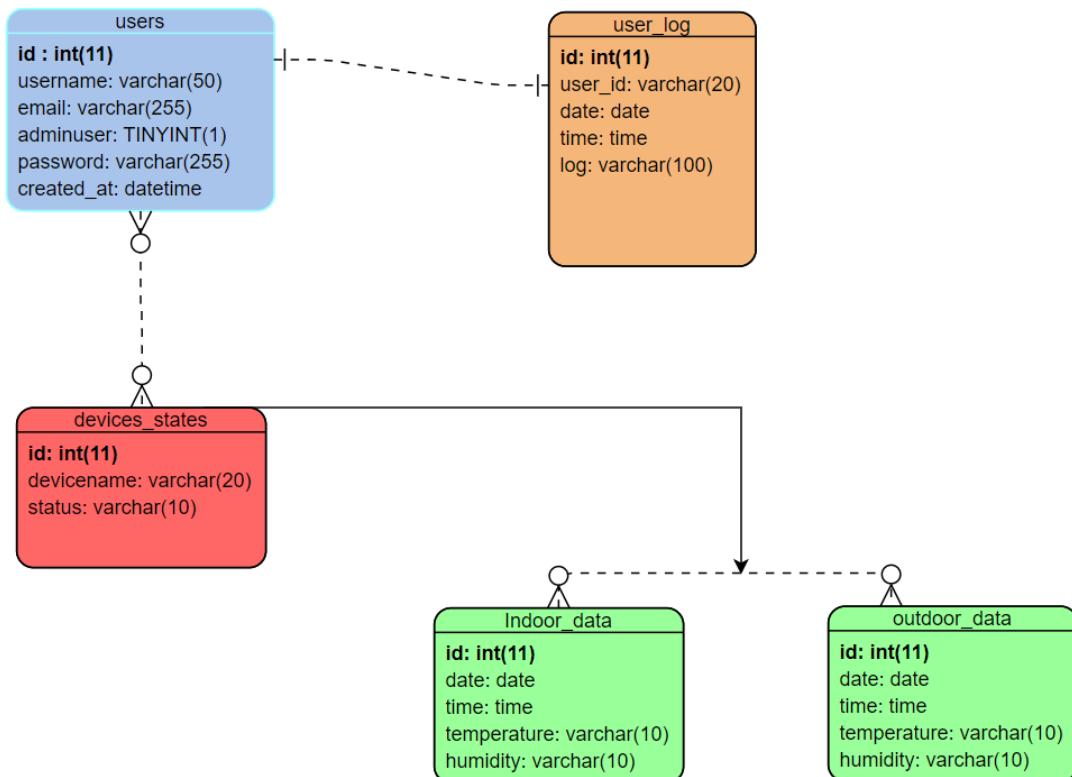
6) Συσχετισμός μεταξύ πινάκων της βάσης δεδομένων

Όπως παρατηρήθηκε και στην προηγούμενη παράγραφο, κάποια στοιχεία πινάκων βρίσκονται με το ίδιο ή διαφορετικό όνομα και σε άλλους πίνακες. Με τον τρόπο αυτό επιτυγχάνεται η συσχέτιση ενός με έναν ή περισσότερους πίνακες.

Ως αρχικός πίνακας, ο πίνακας users δεν περιέχει κανένα στοιχείο κάποιου άλλου. Παράλληλα όμως, ο αναγνωριστικός αριθμός αυτού (στοιχείο id) βρίσκεται στον πίνακα user_log με όνομα user_id, έτσι η κάθε καταγεγραμμένη δραστηριότητα αντιστοιχεί στον χρήστη που ανήκει.

Οσο αφορά τους πίνακες indoor_data & outdoor_data, περιέχουν ίδιου τύπου πληροφορίες και διαχειρίζονται με τον ίδιο τρόπο. Η διαφορά τους είναι πώς προέρχονται από διαφορετικούς χώρους του σπιτιού.

Ακόμη, ο πίνακας devices_states έχει άμεση σχέση με τους δύο πίνακες indoor_data & outdoor_data, καθώς ανάλογα με τις πληροφορίες που έχει ο χρήστης από τους πίνακες θα λάβει την απόφαση να αλλάξει την κατάσταση στις έξυπνες συσκευές. Παράλληλα όμως, θα πρέπει να έχει και τα δικαιώματα τα οποία δίνονται από το πεδίο adminuser του πίνακα users.



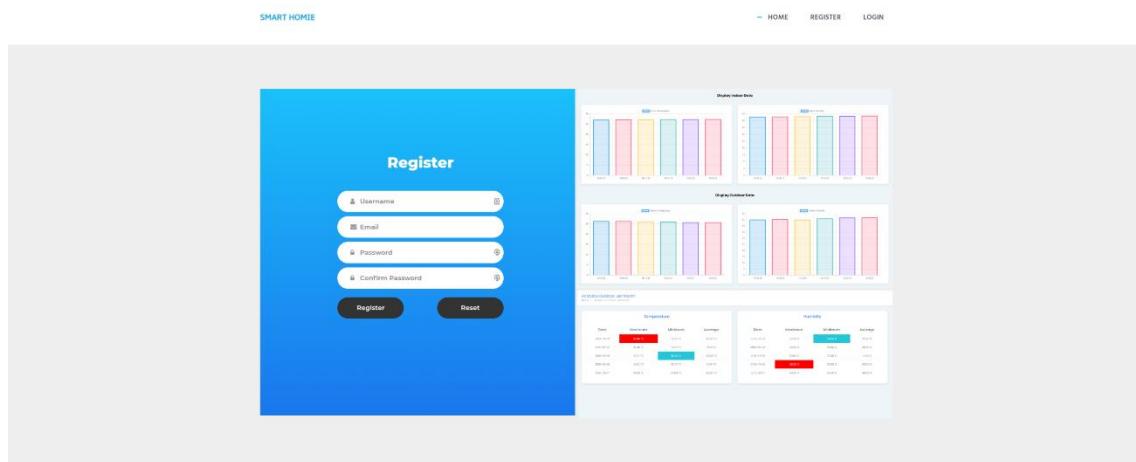
Σχήμα 4.2 Συσχέτισμός πινάκων της βάσης δεδομένων

4.4 Περιγραφή Λειτουργίας Διαδικτυακής Εφαρμογής Smart Homie

Η διαδικτυακή εφαρμογή Smart Homie έχει σκοπό στη παρακολούθηση των έξυπνων συσκευών ενός σπιτιού καθώς και η μέτρηση της θερμοκρασίας και υγρασίας στους εσωτερικούς & εξωτερικούς χώρους του σπιτιού. Οι λειτουργίες αυτής διατυπώθηκαν και υλοποιήθηκαν από τις απαιτήσεις των πιθανών χρηστών και κατ' επέκταση του συστήματος. Η διαδικτυακή εφαρμογή χωρίζεται σε έξι βασικές καρτέλες, οι οποίες εμφανίζονται μετά την είσοδο του χρήστη στην εφαρμογή. Υπάρχουν επίσης δύο καρτέλες οι οποίες αφορούν στην εγγραφή και είσοδο του χρήστη. Αξίζει να σημειωθεί ότι παρά το γεγονός ότι, η γλώσσα της διαδικτυακής εφαρμογής είναι η αγγλική, οι ετικέτες οι τίτλοι, τα διαγράμματα καθώς και τα ενημερωτικά μηνύματα είναι διατυπωμένα και έχουν συνταχθεί με τον πιο δυνατό κατανοητό τρόπο ακόμα και για χρήστης με στοιχειώδη γνώση της αγγλικής γλώσσας.

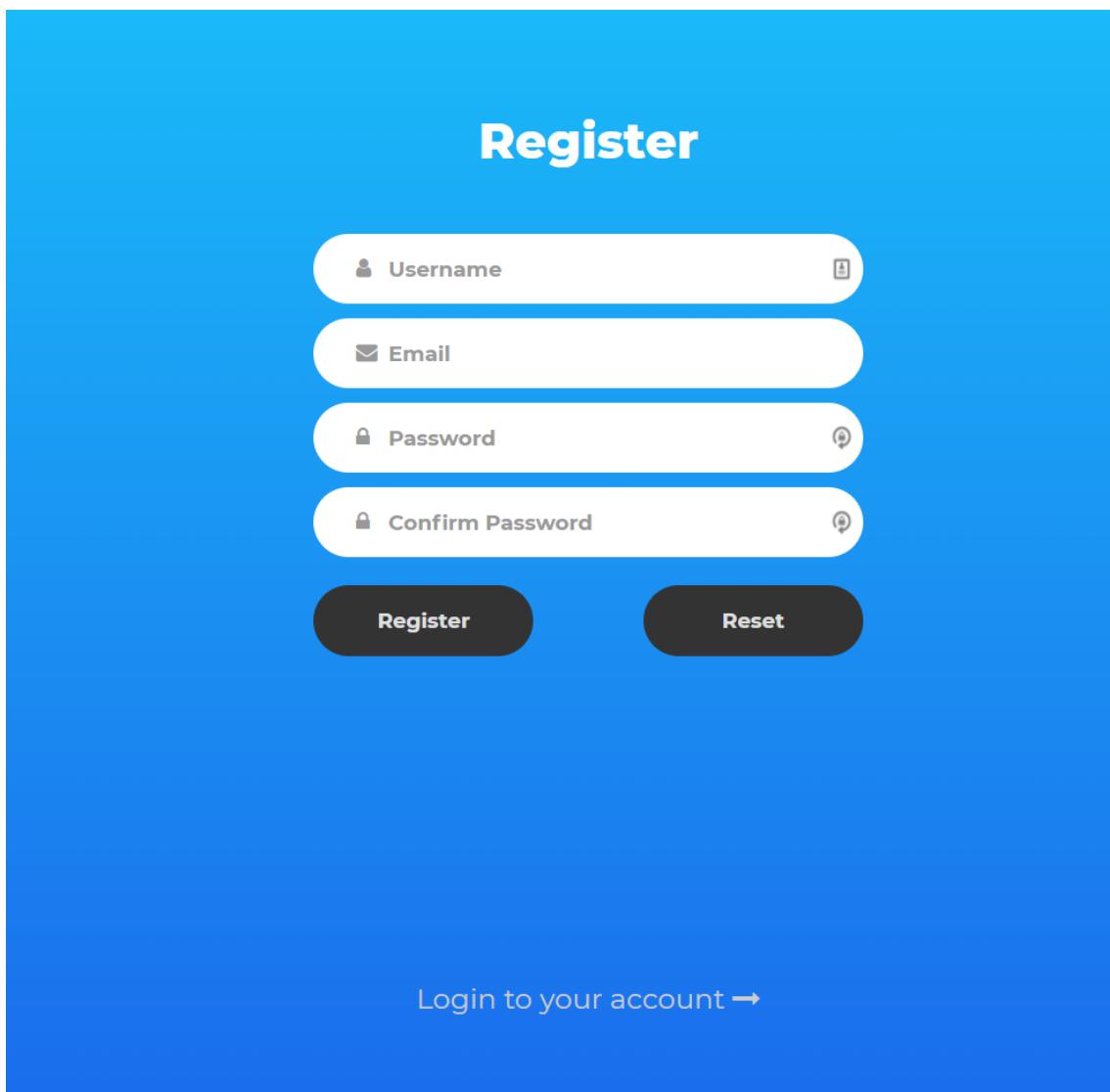
4.4.1. Λειτουργία Εγγραφής και Λειτουργία Σύνδεσης (Register & Log In)

Ξεκινώντας, στην αρχική σελίδα της παρούσας διπλωματικής εργασίας, ο χρήστης μπορεί να διαβάσει για τις δυνατότητες της εφαρμογής καθώς και να δει κάποιες εικόνες από διάφορες εσωτερικές καρτέλες που θα έβλεπε μετά την είσοδο σε αυτή. Στο παρακάτω Σχήμα εμφανίζεται η αρχική σελίδα.



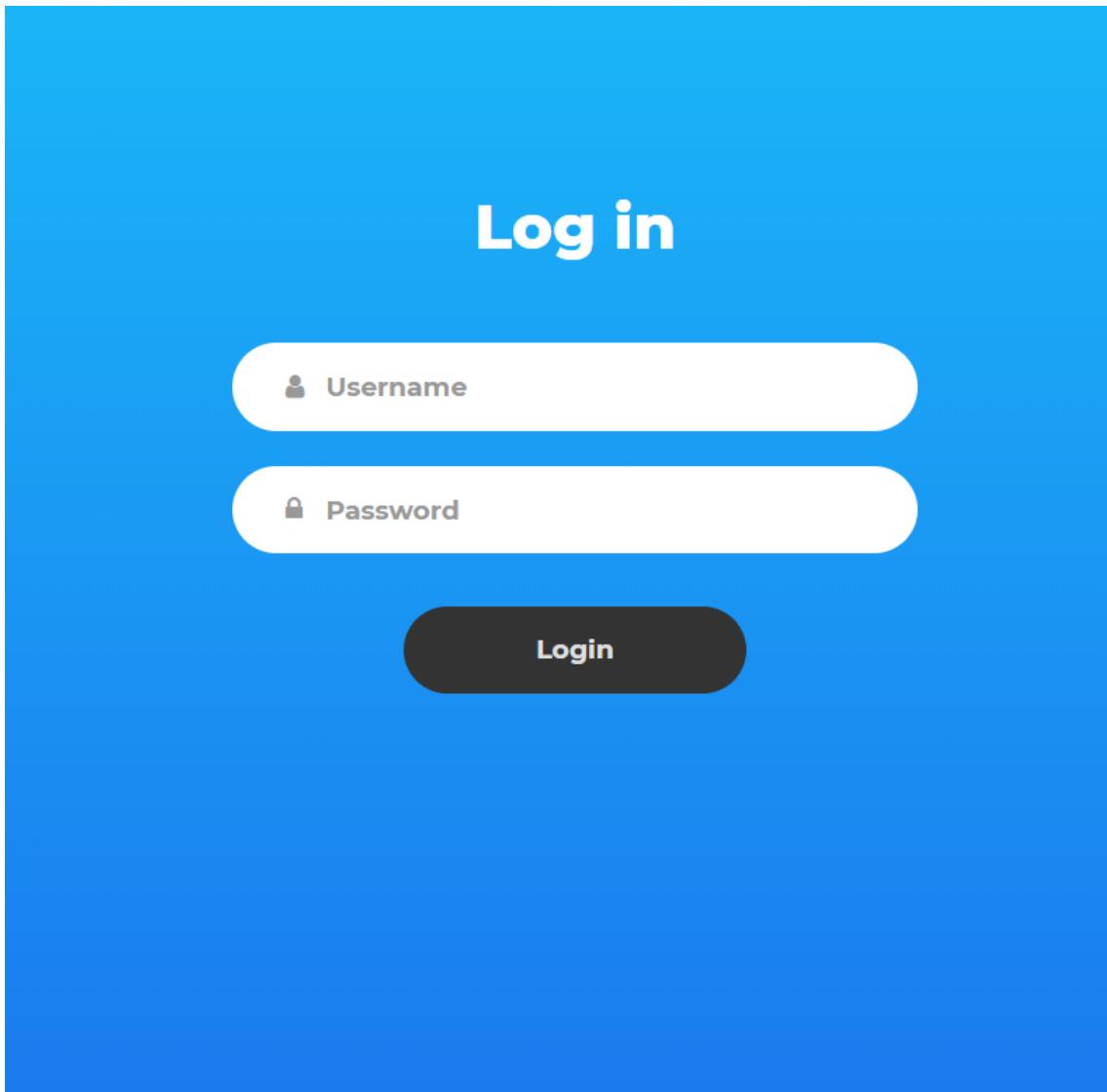
Σχήμα 4.3 Αρχική σελίδα εφαρμογής

Επάνω και δεξιά, βρίσκεται το μενού, στο οποίο εμφανίζονται και οι δύο σημαντικοί σύνδεσμοι μέσω των οποίων ο χρήστης μπορεί να συνεχίσει την περιήγηση του. Είναι ο σύνδεσμος εγγραφής (register) και ο σύνδεσμος είσοδος (Log In). Ξεκινώντας, πατώντας την επιλογή Register συναντάται η φόρμα εγγραφής, και στα πεδία Username, Email, Password, και Confirm Password γίνεται η συμπλήρωση των αντίστοιχων προσωπικών στοιχείων.



Σχήμα 4.4 Καρτέλα Εγγραφής (Register)

Σε περίπτωση που κάποιο από τα πεδία δεν έχει συμπληρωθεί δεν ολοκληρώνεται η διαδικασία της εγγραφής και εμφανίζεται ενημερωτικό μήνυμα ελέγχου, επίσης το ίδιο σφάλμα αντιμετωπίζεται και σε περίπτωση που οι δύο κωδικοί πρόσβασης δεν ταυτίζονται. Μετά την ορθή συμπλήρωση των πεδίων, ολοκληρώνεται η εγγραφή του χρήστη και η εφαρμογή των μεταφέρει αυτόματα στην σελίδα εισόδου (Log In).

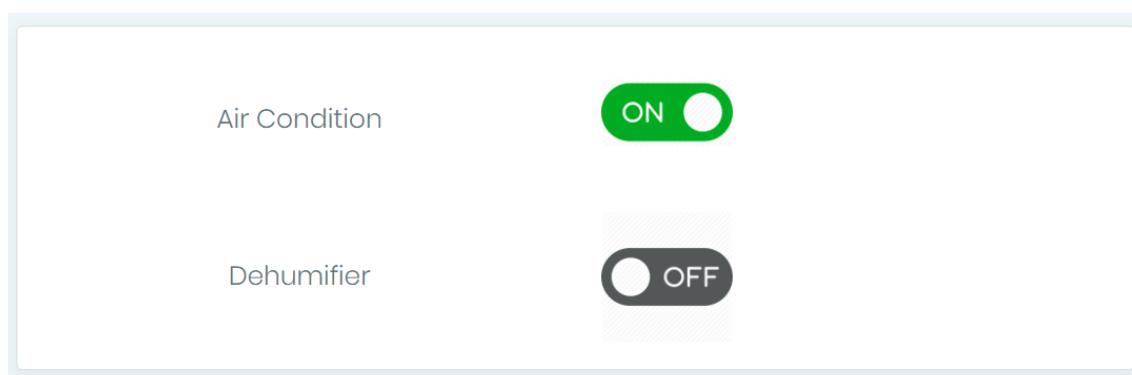


Σχήμα 4.5 Καρτέλα Εισόδου (Log In)

Ακολουθεί η σύνδεση του χρήστη μέσω της Log in φόρμας που εμφανίζεται αυτόματα, χρησιμοποιώντας το όνομα χρήστη και τον κωδικό πρόσβασης με τα οποία πραγματοποιήθηκε η εγγραφή. Στην περίπτωση που κάποιο ή και τα δύο πεδία συμπληρωθούν με λανθασμένα στοιχεία εμφανίζεται μήνυμα προειδοποίησης. Στην αντίθετη περίπτωση, ο χρήστης πραγματοποιεί την είσοδό του στην εφαρμογή και έπειτα μπορεί να πλοηγηθεί σε αυτήν και να πραγματοποιήσει οποιαδήποτε διαδικασία επιθυμεί, όπως η επεξεργασία των προσωπικών του δεδομένων και η παρακολούθηση των δεδομένων που έχουν συλλεχθεί σε περίπτωση που είναι ήδη χρήστης. Οι διαδικασίες αυτές μπορούν να πραγματοποιηθούν επιλέγοντας κάποια από τα tabs (καρτέλες) του μενού της εφαρμογής που βρίσκεται στο αριστερό μέρος αυτής.

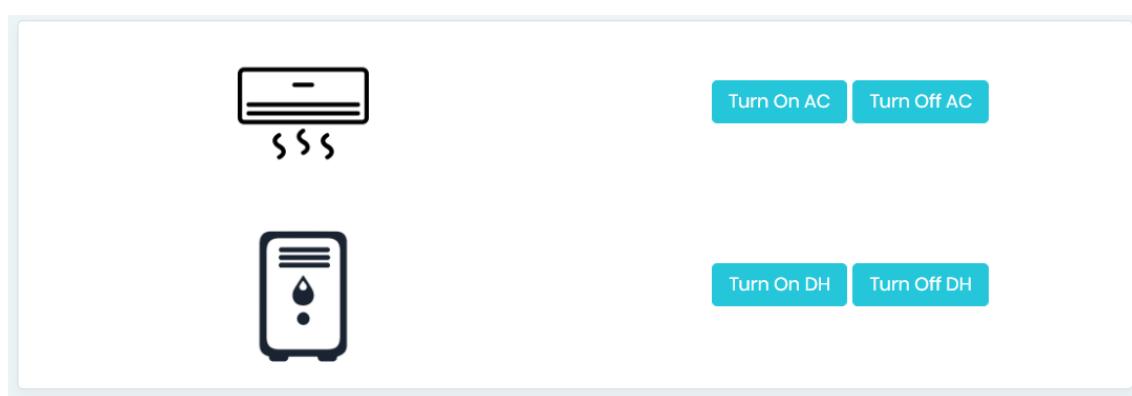
4.4.2. Λειτουργία Αρχικής Καρτέλας Dashboard

Με την επιτυχή είσοδο του χρήστη και την ταυτοποίηση του από την βάση δεδομένων ο χρήστης μεταφέρεται στην σελίδα dashboard όπου είναι και η αρχική σελίδα εντός της διαδικτυακής εφαρμογής. Σε αυτή την καρτέλα περιέχονται βασικές πληροφορίες αλλά και ο κύριος λόγος αλληλεπίδρασης του χρήστη με την διαδικτυακή εφαρμογή. Αρχικά, ο χρήστης αντικρίζει τις υπάρχουσες συνδεδεμένες έξυπνες συσκευές που είναι τοποθετημένες στο σπίτι του. Δίπλα από κάθε συσκευή βρίσκεται μία ενδειξη για την κατάσταση της έξυπνης συσκευής (ενεργοποιημένη/απενεργοποιημένη). Στο παρακάτω Σχήμα απεικονίζεται το τμήμα που μόλις εξετάσαμε.



Σχήμα 4.6 Τμήμα ένδειξης κατάστασης οικιακών συσκευών

Δεξία από το τμήμα που μόλις περιγράψαμε, υπάρχουν τα κουμπιά ενεργοποίησης και απενεργοποίσης των έξυπνων συσκευών καθώς και μία εικόνα για τον διαχωρισμό των συσκευών.



Σχήμα 4.7 Τμήμα εναλλαγής κατάστασης οικιακών συσκευών

Τέλος, στο κάτω μέρος της σελίδας βρίσκεται το τμήμα στο οποίο ο χρήστης μπορεί να ενημερωθεί για την δραστηριότητα του για τον τελευταίες 30 ημέρες. Αρχικά, υπάρχει το μήνυμα καλωσορίσματος, έπειτα ο υπότιτλος που υποδεικνύει τον σκοπό αυτού του τμήματος και μετά ο πίνακας στον οποίο εμφανίζεται η δραστηριότητα.

Hello, Admin		
Your User Activity		
Date	Time	Action
2018-09-21	11:45:08	Logged In
2018-09-21	14:24:26	Logged Out
2018-09-21	14:24:56	Logged In
2018-09-21	14:30:38	Logged Out
2018-09-21	14:30:58	Logged In
2018-09-21	14:57:46	Logged Out

Σχήμα 4.8 Τμήμα εμφάνισης δραστηριότητα χρήστη

Η συνολική εικόνα που αντικρίζει ο χρήστης κατά την είσοδο του στην διαδικτυακή εφαρμογή φαίνεται στο Σχήμα 4.9.

The dashboard interface includes a sidebar with navigation links: Dashboard, Live Monitoring, Analytics, Profile, Users, and Register User. The main area displays a live monitoring feed for an Air Condition unit, which is currently ON. Below it is a Dehumidifier unit, which is currently OFF. Each unit has a 'Turn On' and 'Turn Off' button. At the top, the current time is shown as 19:01:05. A user activity log at the bottom shows the same sequence of events as the previous screenshot, with the addition of a final entry at 19:20:02.

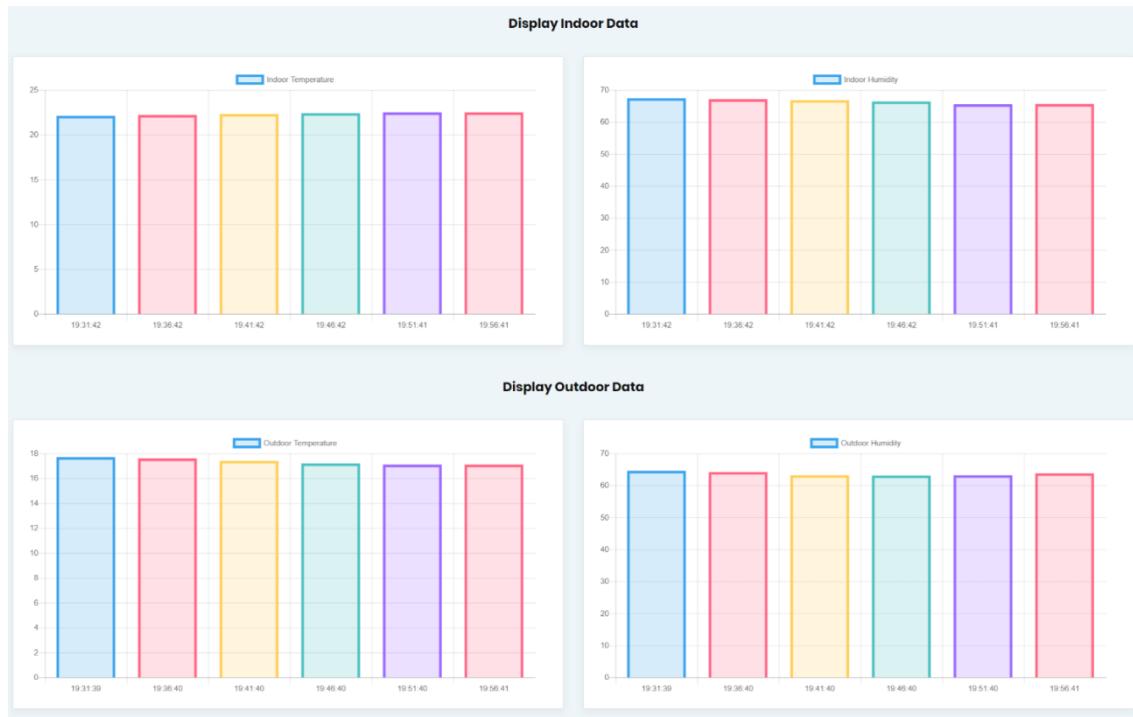
Date	Time	Action
2018-09-21	11:45:08	Logged In
2018-09-21	14:24:26	Logged Out
2018-09-21	14:24:56	Logged In
2018-09-21	14:30:38	Logged Out
2018-09-21	14:30:58	Logged In
2018-09-21	14:57:46	Logged Out
	19:20:02	Logged In

Σχήμα 4.9 Καρτέλα Dashboard

4.4.3. Λειτουργία Καρτέλας Live Monitoring

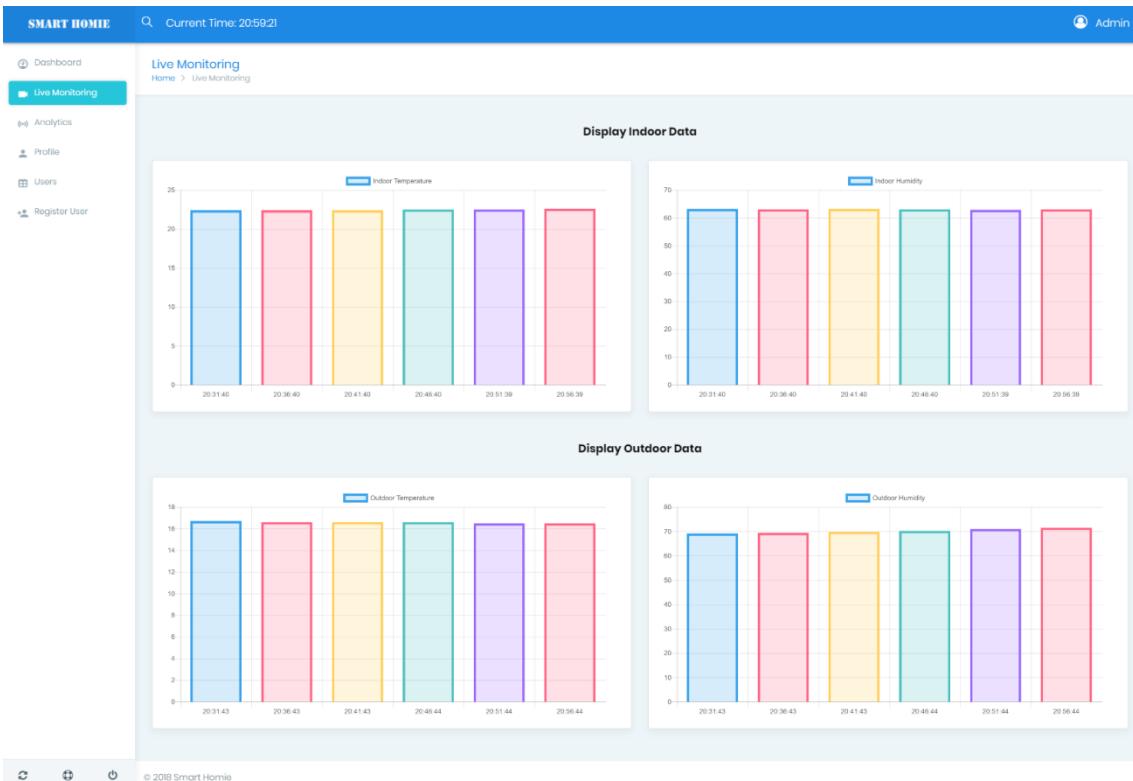
Προχωρώντας στην επόμενη καρτέλα που είναι η Live Monitoring, το δυναμικό τμήμα της εφαρμογής αποτελείται από διαγράμματα. Σε αυτό το σημείο ο χρήστης μπορεί να ενημερωθεί σε πραγματικό χρόνο για τις τιμές που συλλέγονται από τους αισθητήρες που είναι τοποθετημένοι στους εσωτερικούς αλλά και εξωτερικούς χώρους του σπιτιού. Οι αισθητήρες που έχουμε τοποθετήσει στο σπίτι, για τις αναγκές της παρούσας διπλωματικής εργασίας μας παράγουν διαγράμματα θερμοκρασίας και υγρασίας. Στα αριστερά, έχουν τοποθετηθεί τα διαγράμματα θερμοκρασίας και στα δεξιά τα διαγράμματα υγρασίας. Από την άλλη, στο κάθετο προσανατολισμό, τα διαγράμματα που

βρίσκονται στο επάνω μέρος αφορούν τις τιμές των δεδομένων από τους εσωτερικούς χώρους και στο κάτω μέρος τις τιμές από τους εξωτερικούς χώρους. Τα παραγόμενα διαγράμματα φαίνονται στο Σχήμα 4.10.



Σχήμα 4.10 Διαγράμματα παρακολούθησης τιμών σε ζωντανό χρόνο

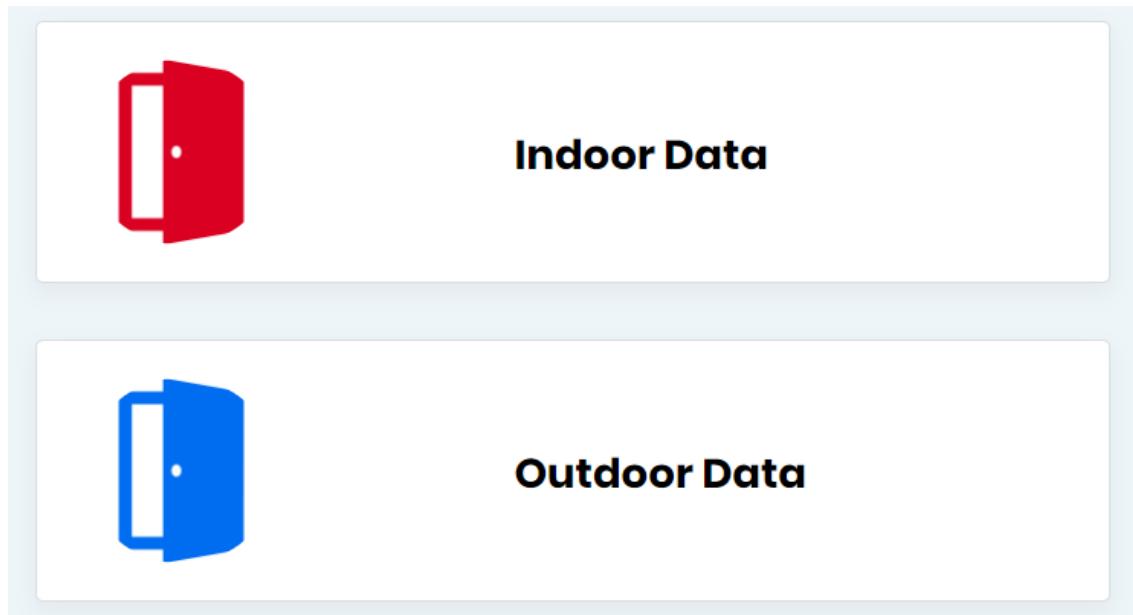
Η συνολική εικόνα που αντικρίζει ο χρήστης κατά την είσοδο του στην καρτέλα Live Monitoring φαίνεται στο Σχήμα 4.11.



Σχήμα 4.11 Καρτέλα Live Monitoring

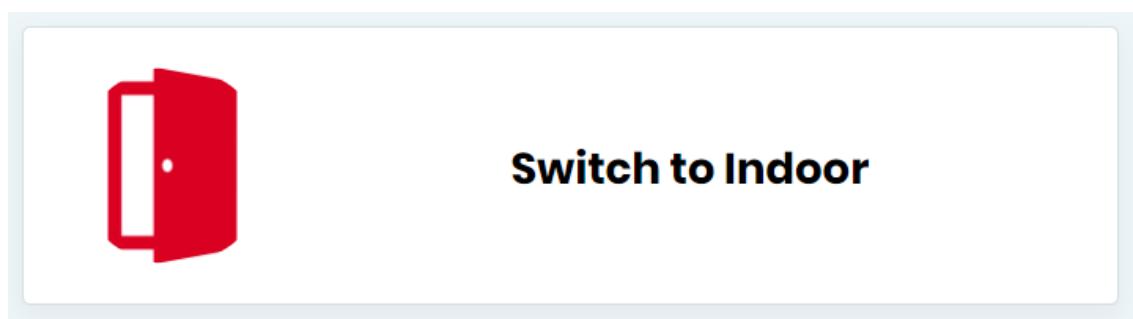
4.4.4. Λειτουργία Καρτέλας Analytics

Η επόμενη καρτέλα στην οποία μπορεί να μεταβεί ο χρήστης είναι η καρτέλα των στατιστικών (Analytics). Πατώντας λοιπόν σε αυτή την καρτέλα, ο χρήστης μεταφέρεται και έτσι στην οθόνη του εμφανίζονται οι διαθέσιμοι χώροι από τους οποίους συλλέγονται τα δεδομένα. Για τις ανάγκες της παρούσας διπλωματικής εργασίας, οι χώροι χωρίστηκαν σε εξωτερικούς και εσωτερικούς. Έτσι, στο παρακάτω Σχήμα φαίνονται οι επιλογές που έχει ο χρήστης για να δει τα στατιστικά.



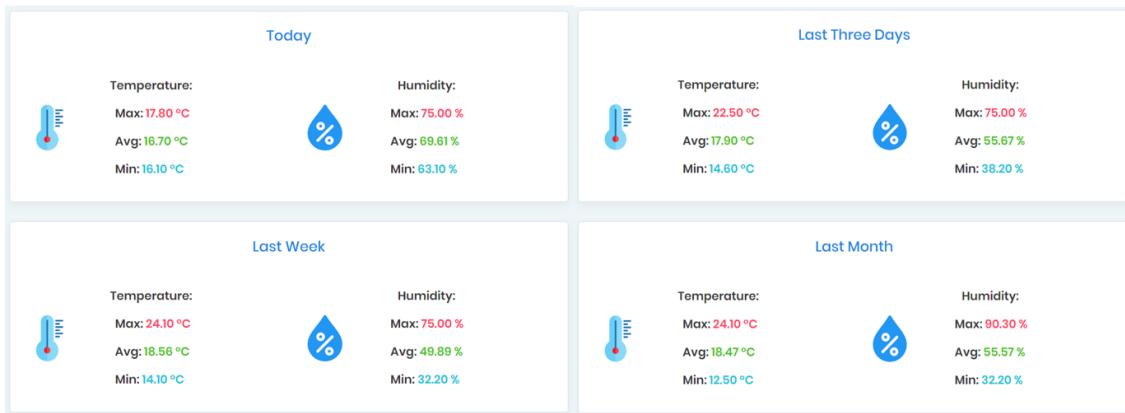
Σχήμα 4.12 Καρτέλα Analytics: Επιλογή χώρων

Εφόσον ο χρήστης επιλέξει για ποιους χώρους θέλει να δει τα στατιστικά, του εμφανίζονται κάποια νέα τμήματα στην οθόνη. Το πρώτο τμήμα, το οποίο βρίσκεται στο επάνω μέρος της σελίδας αφορά τη μετάβαση του χρήστη στους υπόλοιπους διαθέσιμους χώρους όπου υπάρχουν στατιστικά. Για παράδειγμα, εφόσον ο χρήστης κάνει κλικ στην επιλογή Outdoor Data, η σελίδα στην οποία θα μεταβεί, στο επάνω μέρος θα έχει εμφανίσει τμήμα το οποίο θα έχει την επιλογή Switch to Indoor όπως φαίνεται στο Σχήμα.



Σχήμα 4.13 Καρτέλα Analytics: Εναλλαγή σε διαφορετικό χώρο για εμφάνιση στατιστικών

Έπειτα, προχωρώντας προς το κατώτερο μέρος της καρτέλας, εμφανίζονται στο χρήστη 4 τμήματα, τα οποία αφορούν στατιστικά για διαφορετικές χρονικές περιόδους μέτρησης. Οι περίοδοι μέτρησης έχουν χωριστεί σε: 1) Τελευταία Ημέρα 2) Τελευταίες Τρεις Ημέρες 3) Τελευταία Εβδομάδα 4) Τελευταίος Μήνας. Στο ανάλογο τμήμα εμφανίζονται τα δεδομένα θερμοκρασίας & υγρασίας όπως φαίνεται στο Σχήμα 4.14.



Σχήμα 4.14 Καρτέλα Analytics: Χρονικές περίοδοι μέτρησης

Σε κάθε τμήμα, ο χρήστης μπορεί να ενημερωθεί για τη μέγιστη, την ελάχιστη αλλά και την μέση θερμοκρασία και υγρασία για το χρονικό διάστημα της επιλογής του.

Ακόμη μία δυνατότητα που υπάρχει στην παρούσα καρτέλα είναι η εμφάνιση των στατιστικών σε ένα συγκεντρωτικό πίνακα ανάλογα την περίοδο μέτρησης που θέλει ο χρήστης να εξετάσει. Για παράδειγμα, πατώντας στο τμήμα του τελευταίου μήνα, ο χρήστης μεταφέρεται σε μία νέα σελίδα, όπου εκεί εμφανίζονται τα στατιστικά υπό μορφή πίνακα. Ο πίνακας έχει 4 στήλες. Στην πρώτη στήλη εμφανίζεται η ημερομηνία και στις επόμενες 3 στήλες εμφανίζονται αντίστοιχα, η μέγιστη, η ελάχιστη και η μέση τιμή. Υπάρχουν 2 πίνακες, ένας για την θερμοκρασία και ένας για την υγρασία. Επιπλέον, στους πίνακες υπογραμμίζονται οι μέγιστες και οι ελάχιστες τιμές οι οποίες έχουν καταγραφεί για το χρονικό διάστημα που έχουμε επιλέξει. Οι μέγιστες τιμές σημειώνονται με κόκκινο χρώμα, ενώ οι ελάχιστες με μπλε.

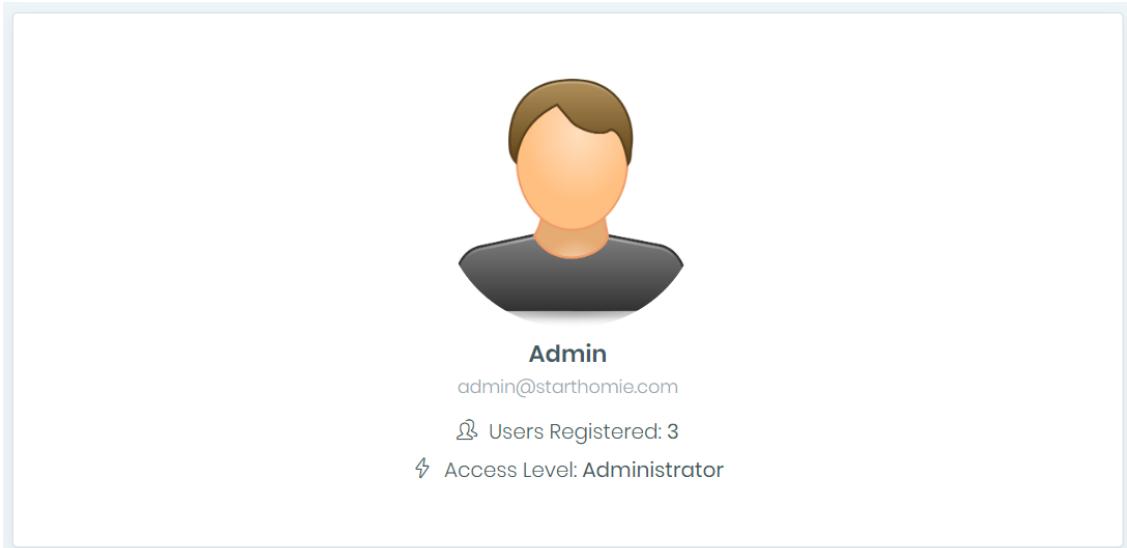
Όλα τα παραπάνω χαρακτηριστικά μπορούν να απεικονιστούν στο παρακάτω Σχήμα.

Temperature			
Date	Maximum	Minimum	Average
2018-10-16	17.80 °C	16.10 °C	16.71 °C
2018-10-15	22.50 °C	14.70 °C	18.45 °C
2018-10-14	19.60 °C	14.60 °C	17.63 °C
2018-10-13	24.10 °C	16.70 °C	19.13 °C
2018-10-12	22.70 °C	18.00 °C	19.74 °C
2018-10-11	22.30 °C	14.20 °C	18.27 °C
2018-10-10	23.80 °C	14.10 °C	18.55 °C
2018-10-09	23.60 °C	14.40 °C	18.66 °C
2018-10-08	23.10 °C	14.20 °C	18.01 °C
2018-10-07	22.10 °C	12.50 °C	16.99 °C
2018-10-06	21.90 °C	13.40 °C	17.24 °C
2018-10-05	23.30 °C	15.70 °C	18.54 °C
2018-10-04	21.60 °C	16.40 °C	19.59 °C
2018-10-03	22.70 °C	19.40 °C	20.96 °C

Σχήμα 4.15 Καρτέλα Analytics: Αναλυτικός πίνακας μετρήσεων

4.4.5. Λειτουργία Καρτέλας Profile

Η επόμενη καρτέλα είναι η καρτέλα του προφίλ (profile). Ο χρήστης μπορεί να χρησιμοποιήσει αυτή την σελίδα ώστε να δει πληροφορίες που αφορούν τον λογαριασμό από τον οποίο έχει εισέλθει στην εφαρμογή. Η σελίδα χωρίζεται σε δύο σημαντικά τμήματα. Στο αριστερό μέρος υπάρχουν σημαντικά στοιχεία που υποδεικνύουν το όνομα χρήστη, το email που έχει δηλώσει, τα δικαιώματα που έχει ο λογαριασμός του, δηλαδή εάν είναι διαχειριστής ή απλός χρήστης καθώς και πόσοι ακόμη χρήστες έχουν εγγραφή συνολικά στην τοπική εγκατάσταση της διαδικτυακής εφαρμογής. Όλα τα στοιχεία που αναφέρθηκαν εμφανίζονται στο παρακάτω Σχήμα.



Σχήμα 4.16 Καρτέλα Profile: Πληροφορίες λογαριασμού

Στο δεξιά τμήμα της σελίδας, εμφανίζεται στον χρήστη μία φόρμα, μέσω της οποίας μπορεί να ανανεώσει τα στοιχεία του λογαριασμού του. Οι προεπιλεγμένες τιμές στα πεδία είναι τα ήδη υπάρχοντα στοιχεία του χρήστη. Ο κωδικός δεν παρουσιάζεται για λόγους ασφαλείας. Ο χρήστης μπορεί να τροποποιήσει οποιοδήποτε από τα στοιχεία του και μόλις πατήσει το κουμπί της υποβολής, δημιουργείται ένα αίτημα στην βάση δεδομένων για αλλαγή. Εφόσον το αίτημα γίνει δεκτό, η διαδικτυακή εφαρμογή αποσυνδέει τον χρήστη για λόγους ασφαλείας και πρέπει να πραγματοποιηθεί εκ νέου σύνδεση. Σε περίπτωση που ο χρήστης δεν τροποποιήσει κάποιο από τα πεδία, το στοιχείο εκείνου του πεδίου παραμένει ίδιο. Στο παρακάτω Σχήμα φαίνεται η φόρμα αλλαγής στοιχείων του χρήστη.

Update Your Credentials

Full Name
Admin

Email
admin@starthomie.com

Password

Update Profile

Σχήμα 4.17 Καρτέλα Profile: Φόρμα ανανέωσης στοιχείων λογαριασμού

4.4.6. Λειτουργία Καρτέλας Users

Συνεχίζοντας, η επομένη καρτέλα που συναντά ο χρήστης είναι η καρτέλα χρήστες (Users). Τονίζεται ότι για να εμφανιστεί η καρτέλα με τους χρήστες, θα πρέπει ο λογαριασμός από τον οποίο έχει εισέλθει ο χρήστης να έχει δικαιώματα διαχειριστή, διαφορετικά δεν θα υπάρχει στο μενού ώστε ο χρήστης να κάνει κλικ σε αυτή. Εφόσον ο χρήστης κάνει κλικ στην επιλογή users από το μενού που θα εξηγηθεί σε επόμενο κεφάλαιο μεταφέρεται σε μία νέα σελίδα. Εκεί, παρουσιάζεται ένας πίνακας όπως φαίνεται στο παρακάτω Σχήμα 4.18.

Registered Users Table				
ID	UserName	Email	Admin Rights	Action
1	Admin	admin@starthomie.com	yes	<button>Delete User</button> <button>Change Admin Rigths</button>
2	User	user@smarthomie.com	no	<button>Delete User</button> <button>Change Admin Rigths</button>
3	Stefanos Missin	stefanosmissin@gmail.com	yes	<button>Delete User</button> <button>Change Admin Rigths</button>

Σχήμα 4.18 Καρτέλα Users: Πίνακας εγγεγραμμένων χρηστών

Ο πίνακας συμπληρώνεται από τον πίνακα users της βάσης δεδομένων της διαδικτυακής εφαρμογής. Κατά την διάρκεια φόρτωσης της σελίδας users η εφαρμογή δημιουργεί ένα αίτημα στη βάση δεδομένων για την άντληση των βασικών πληροφοριών των χρηστών. Στην πρώτη στήλη του πίνακα εμφανίζεται ο αναγνωριστικός αριθμός κάθε χρήστη (ID). Ακολουθεί το όνομα χρήστη (username), το email που έχει δηλώσει ο εκάστοτε χρήστης καθώς και το εάν ο χρήστης έχει δικαιώματα διαχειριστή ή όχι. Τέλος εμφανίζονται δύο κουμπιά ενέργειας επάνω σε κάθε χρήστη. Το πρώτο κουμπί (κόκκινο), μπορεί να διαγράψει ολοκληρωτικά τον χρήστη. Εφόσον η διαγραφή του χρήστη είναι επιτυχημένη, εμφανίζεται το παρακάτω μήνυμα στην οθόνη του χρήστη.



Σχήμα 4.19 Καρτέλα Users: Μήνυμα επιτυχημένης διαγραφής χρήστη

Το δεύτερο κουμπί (μωβ), δίνει τη δυνατότητα στον διαχειριστή να αλλάξει τα δικαιώματα των χρηστών που είναι εγγεγραμμένοι στη διαδικτυακή εφαρμογή. Για παράδειγμα, στο Σχήμα 4.18, παρατηρούμε πώς ο χρήστης User δεν έχει δικαιώματα διαχειριστή. Πατώντας το δεύτερο κουμπί τα δικαιώματα του χρήστη User αλλάζουν όπως φαίνεται στο Σχήμα 4.20.

Registered Users Table					
ID	UserName	Email	Admin Rights	Action	
1	Admin	admin@starthomie.com	yes	<button>Delete User</button>	<button>Change Admin Rigths</button>
2	User	user@smarthomie.com	yes	<button>Delete User</button>	<button>Change Admin Rigths</button>
3	Stefanos Missin	stefanosmissin@gmail.com	yes	<button>Delete User</button>	<button>Change Admin Rigths</button>

Σχήμα 4.20 Καρτέλα Users: Τροποποίηση δικαιωμάτων χρήστη

4.4.7. Λειτουργία καρτέλας Register User

Η τελευταία καρτέλα που υπάρχει στο μενού περιήγησης, είναι η καρτέλα δημιουργίας νέου χρήστη (Register User). Τονίζεται πώς για να εμφανιστεί η καρτέλα δημιουργίας νέου χρήστη, θα πρέπει ο λογαριασμός από τον οποίο έχει εισέλθει ο χρήστης να έχει δικαιώματα διαχειριστή, διαφορετικά δεν θα υπάρχει στο μενού ώστε ο χρήστης να κάνει κλικ σε αυτή. Όταν ο χρήστης μεταβεί στην συγκεκριμένη σελίδα, εμφανίζεται στην οθόνη του μία φόρμα. Η φόρμα αυτή αποσκοπεί στην δημιουργία ενός νέου χρήστη. Τα πεδία αυτής της φόρμας είναι:

- Όνομα χρήστη (username)
- Email
- Επιλογή δικαιωμάτων του χρήστη
- Κωδικός πρόσβασης (password)

Εφόσον συμπληρωθεί σωστά η φόρμα, με το κουμπί submit γίνεται η αίτηση από την διαδικτυακή εφαρμογή προς τη βάση δεδομένων για τη δημιουργία ενός νέου χρήστη. Η φόρμα αυτή εμφανίζεται όπως στο παρακάτω Σχήμα.

Register New User

Username

Email

Give Administrator Rights?

No

Password

Submit

Σχήμα 4.21 Καρτέλα Register User: Φόρμα δημιουργίας νέου χρήστη

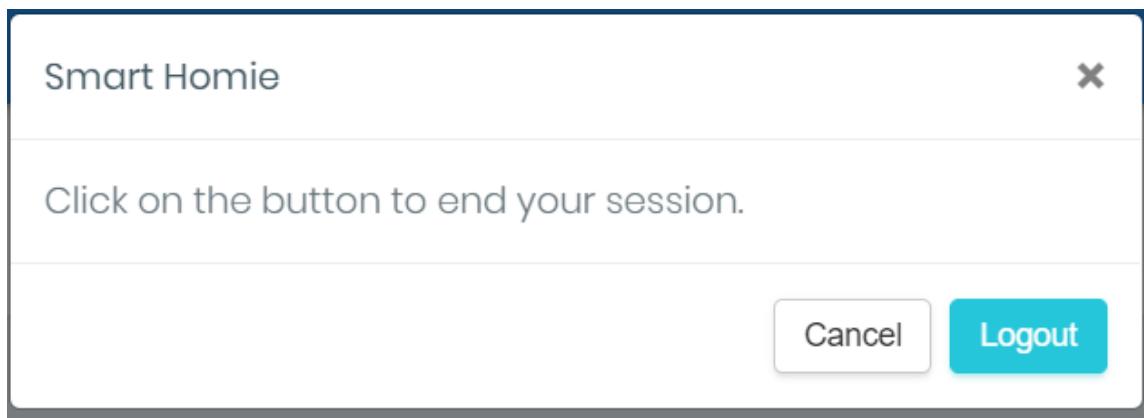
4.4.8. Λειτουργία Λοιπών Σημείων Διαδικτυακής Εφαρμογής

Εκτός από τις καρτέλες που είναι ο κορμός της διαδικτυακής εφαρμογής, υπάρχουν σημεία στο γραφικό περιβάλλον, τα οποία είναι στατικά και αποσκοπούν στην αύξηση της λειτουργικότητας της διαδικτυακής εφαρμογής. Όταν ο χρήστης πραγματοποιήσει την είσοδο στην διαδικτυακή εφαρμογή, στο επάνω μέρος βρίσκεται πάντα η κεντρική μπάρα όπως φαίνεται στο παρακάτω Σχήμα.



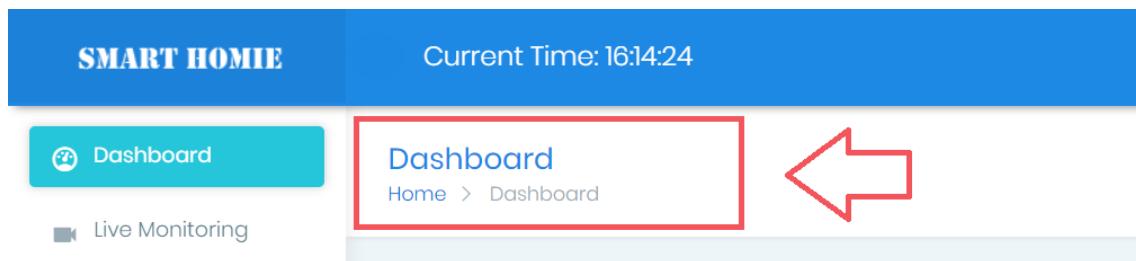
Σχήμα 4.22 Κεντρική μπάρα πληροφοριών

Η μπάρα είναι χρώματος μπλε και περιέχει κάποια σημαντικά στοιχεία για τον χρήστη. Από αριστερά προς τα δεξιά, παρατηρούμε το όνομα της εφαρμογής, την τοπική ώρα (η ώρα αντλείται αυτόματα από την περιοχή στην οποία βρίσκεται ο τοπικός εξυπηρετητής), καθώς και ένα κουμπί το οποίο αφορά την αποσύνδεση του χρήστη από τη διαδικτυακή εφαρμογή. Για την αποσύνδεση από την διαδικτυακή εφαρμογή, ο χρήστης χρειάζεται να κάνει κλικ πάνω στο όνομα χρήστη του και έτσι του εμφανίζεται ένα αναδυόμενο παράθυρο με το κατάλληλο μήνυμα όπως φαίνεται παρακάτω.



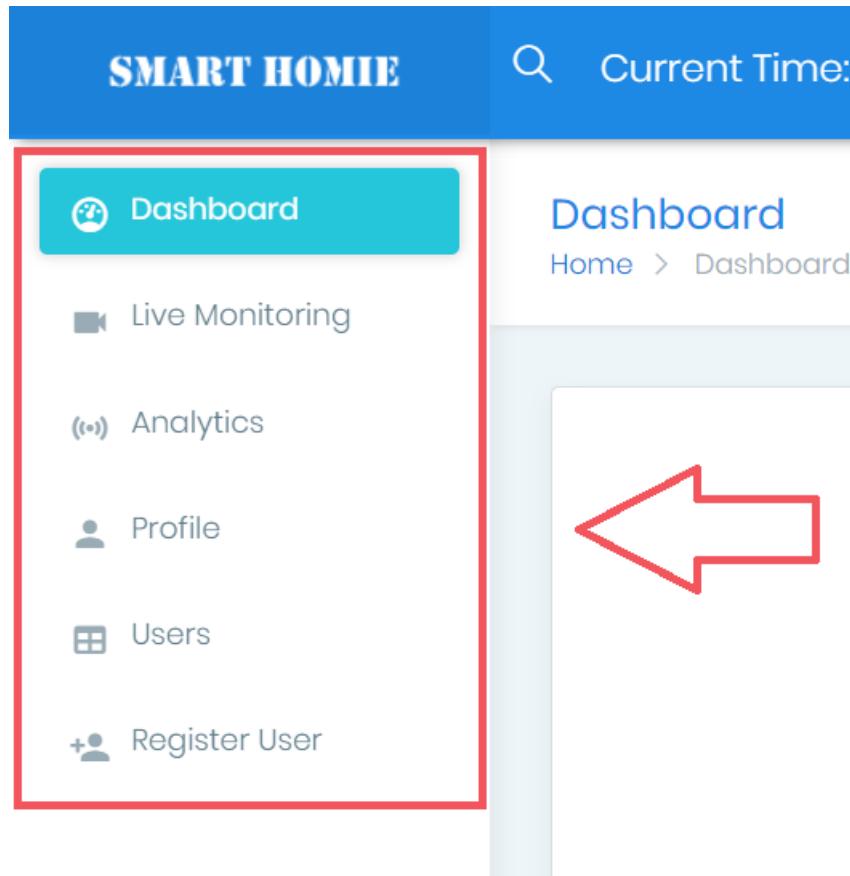
Σχήμα 4.23 Αναδυόμενο παράθυρο αποσύνδεσης χρήστη

Κάτω από την μπάρα που μόλις προαναφέραμε, βρίσκεται μία άλλη μπάρα, η οποία μας υποδεικνύει το μονοπάτι που έχουμε ακολουθήσει ο χρήστης μέσα στη διαδικτυακή εφαρμογή. Είναι ουσιαστικά το σημείο στο οποίο βρισκόμαστε εκείνη την στιγμή ο χρήστης.



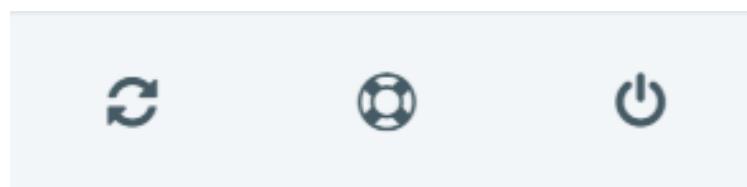
Σχήμα 4.24 Μπάρα μονοπατιού (Directory Path)

Στο αριστερό μέρος της σελίδας, βρίσκεται το κεντρικό μενού, το οποίο περιέχει όλες τις βασικές σελίδες στις οποίες μπορεί να μεταβεί ο χρήστης. Τονίζεται πώς το μενού δεν είναι στατικό και δεν περιέχει πάντα τους ίδιους συνδέσμους. Αυτό εξαρτάται από τα δικαιώματα που έχει ο χρήστης που έχει εισέλθει στην εφαρμογή. Σε προηγούμενο κεφάλαιο έχουν αναφερθεί οι σύνδεσμοι που αποκρύπτονται από τους απλούς χρήστες.



Σχήμα 4.25 Κεντρικό μενού περιήγησης

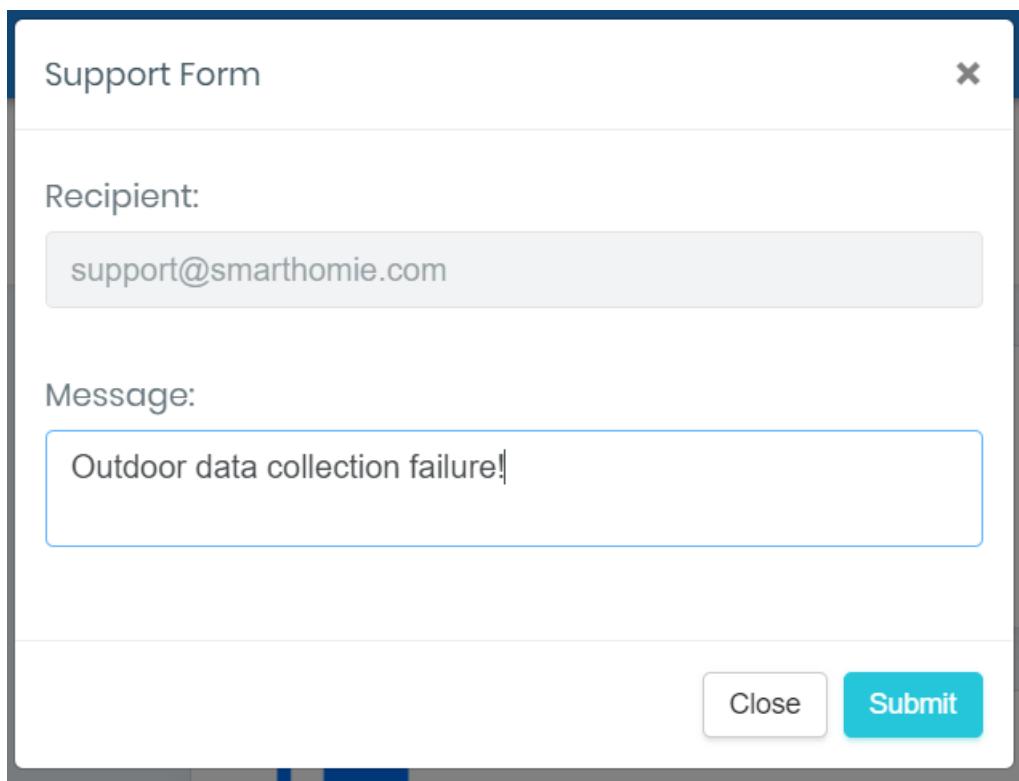
Στο κάτω και αριστερά μέρος της σελίδας υπάρχει ένα ακόμα σημαντικό τμήμα της διαδικτυακής εφαρμογής. Το παρακάτω Σχήμα αφορά το συγκεκριμένο τμήμα της εφαρμογής και έπειτα γίνεται η ανάλυσή του.



Σχήμα 4.26 Βοηθητικές λειτουργίες διαδικτυακής εφαρμογής

Το πρώτο κουμπί στο οποίο απεικονίζονται δύο περιστρεφόμενα βελάκια, αφορά την επί τόπου ανανέωση όλων των συστημάτων της διαδικτυακής εφαρμογής. Εάν για οποιοδήποτε λόγο κολλήσει κάποια λειτουργία της εφαρμογής ο χρήστης με το πάτημα αυτού του κουμπιού μπορεί να πραγματοποιήσει ολική ανανέωση και επαναφόρτωση των συστημάτων της εφαρμογής. Το δεύτερο κουμπί αφορά στην τεχνική υποστήριξη αλλά και στη δήλωση βλαβών στο σύστημα. Πατώντας το συγκεκριμένο κουμπί, ο χρήστης μπορεί να αποστείλει μέσω της φόρμας που αναδύεται το τεχνικό πρόβλημα που αντιμετωπίζει, και έτσι με την υποβολή της φόρμας το μήνυμα αποστέλλεται στον τοπικό

διαχειριστή της εφαρμογής όπως φαίνεται και στο παρακάτω Σχήμα. Το email που αναφέρεται ως παραλήπτης μπορεί να τροποποιηθεί κατάλληλα από τον διαχειριστή του συστήματος.



Σχήμα 4.27 Φόρμα τεχνικής υποστήριξης

Το τρίτο σε σειρά κουμπί από τις βιοηθητικές λειτουργίες, αφορά στην άμεση αποσύνδεση του χρήστη σε περίπτωση που δεν λειτουργεί το κουμπί αποσύνδεσης στην κεντρική μπάρα στο επάνω μέρος της εφαρμογής.

Κεφάλαιο 5

Συμπεράσματα και Περαιτέρω Έρευνα

Στην παρούσα ενότητα γίνεται αναφορά στα συμπεράσματα που προέκυψαν κατά την ανάπτυξη και υλοποίηση της παρούσας διπλωματικής εργασίας, ενώ εξετάζεται η λειτουργικότητα της εφαρμογής καθώς και η σκοπιμότητά της. Αναλύονται τα προβλήματα που προέκυψαν κατά την υλοποίησή της καθώς και οι βελτιώσεις και η περαιτέρω έρευνα που μπορούν να γίνουν στο ήδη υπάρχον αποτέλεσμα.

5.1 Σύνοψη και Συμπεράσματα

Στα πλαίσια της παρούσας διπλωματικής εργασίας ο στόχος που έπρεπε να επιτευχθεί ήταν η υλοποίηση ενός συστήματος. Ο σχεδιασμός και η υλοποίηση του συστήματος χωρίστηκε σε δύο τμήματα, το υλικό και το λογισμικό μέρος. Το πρώτο, αφορά τη δημιουργία και τον προγραμματισμό ενός κυκλώματος βασισμένο στους μικροελεγκτές Arduino Uno Rev3 και WeMos D1 R2, καθώς και τα περιφερειακά εξαρτήματα που συνδέθηκαν σε αυτά. Το δεύτερο, αφορά τη σχεδίαση, τη δημιουργία και τον προγραμματισμό μιας διαδικτυακής εφαρμογής (ιστοσελίδας).

Σκοπός του συστήματος αυτού, είναι η συλλογή δεδομένων θερμοκρασίας και υγρασίας από αισθητήρες οι οποίοι είναι τοποθετημένοι στους εσωτερικούς και εξωτερικούς χώρους ενός σπιτιού. Ακόμη, με την χρήση αυτών των δεδομένων δίνεται η δυνατότητα στον χρήστη για την σωστή χρονικά και οικονομικά ενεργοποίηση/απενεργοποίηση των έξυπνων συσκευών που έχει εγκατεστημένες στο σπίτι του. Η ιστοσελίδα, στην οποία μπορεί να εγγραφεί με ευκολία ένας εν δυνάμει χρήστης, έχει ως κύριες λειτουργίες την καταγραφή, την παρουσίαση και την ανάλυση των δεδομένων θερμοκρασίας και υγρασίας, που συλλέγει το υλικό μέρος του συστήματος, τόσο αριθμητικά όσο και διαγραμματικά. Η σύνδεση του χρήστη στον ιστοχώρο μπορεί να πραγματοποιηθεί από διαφορετικού τύπου συσκευές με την προϋπόθεση ύπαρξης διαδικτυακού περιηγητή. Η πρόσβαση και η πλοήγηση σε αυτόν μπορεί να γίνει εύκολα από υπολογιστή, κινητό τηλέφωνο ή tablet. Κάθε άτομο που συνδέεται στην παρούσα εφαρμογή μπορεί να ρυθμίζει με εύκολο και γρήγορο τρόπο τις υπηρεσίες τις οποίες επιθυμεί να του παρέχει αυτή. Ο χρήστης δεν περιορίζεται ως προς τον αριθμό, τον τύπο και την τοποθεσία των συσκευών καταγραφής δεδομένων που θέλει να εγκαταστήσει, έχοντας παράλληλα την ικανότητα της συγκεντρωτικής εποπτείας αυτών με απλό τρόπο.

Συμπερασματικά, έγινε προσπάθεια για τη δημιουργία ενός οικονομικού συστήματος το οποίο μπορεί να αντικαταστήσει την ανθρώπινη παρουσία σε ένα χώρο που χρήζει επίβλεψης όσον αφορά τις συνθήκες που επικρατούν σε αυτόν. Επίσης, επιχειρήθηκε αποκλειστικά η χρήση τεχνολογιών που στηρίζονται σε πλατφόρμες ανοιχτού λογισμικού. Αυτό επιτεύχθηκε, χρησιμοποιώντας τεχνικές σε PHP, HTML, CSS, JavaScript και Arduino IDE που ακολουθούν αυτήν τη φιλοσοφία.

Καταλήγοντας, πρέπει να σημειωθεί, ότι το σύστημα που αναπτύχθηκε έχει αρκετούς περιορισμούς και ελλείψεις σε σύγκριση με ένα αντίστοιχο εμπορικό. Σημαντικός παράγοντας σε αυτό είναι ότι το κόστος του υλικού μέρους του συστήματος είναι αρκετά μικρότερο από ένα εμπορικό καθώς και το γεγονός ότι δεν χρησιμοποιήθηκαν ειδικά σχεδιασμένες πλακέτες για τις ανάγκες της υλοποίησης της παρούσας διπλωματικής εργασίας. Παράλληλα όμως, αποτελεί έναν ισχυρό πυρήνα πάνω στον οποίο με κάποιες βελτιώσεις, μερικές από τις οποίες αναφέρονται στην ενότητα 5.3, μπορεί να αναπτυχθεί μια μελλοντική αξιόπιστη, και οικονομική λύση.

5.2 Προβλήματα που Προέκυψαν Κατά την Υλοποίηση του Συστήματος

Κατά την εκπόνηση της παρούσας διπλωματικής εργασίας παρουσιάστηκαν διάφορα προβλήματα, τόσο σε επίπεδο υλικού όσο και λογισμικού. Τα περισσότερα εκ των οποίων αντιμετωπίστηκαν με επιτυχία ενώ για μερικά από αυτά δεν βρέθηκε λύση.

Αρχικά, ξεκινώντας με το υλικό μέρος, κατά τη δημιουργία του κυκλώματος και την πρώτη δοκιμαστική λήψη δεδομένων προκλήθηκε βλάβη στον πρώτο αισθητήρα DHT22/AM2302 με αποτέλεσμα να μην εμφανίζονται οι τιμές των δεδομένων στη σειριακή οθόνη του προγραμματιστικού περιβάλλοντος Arduino IDE. Το πρόβλημα αυτό, επιλύθηκε με την αλλαγή στην συνδεσμολογία του αισθητήρα με το μικροελεγκτή Arduino Uno Rev3 καθώς το pin στο οποίο είχε συνδεθεί ο ακροδέκτης των δεδομένων του αισθητήρα ήταν προβληματικός. Με την τροποποίηση αυτή εξασφαλίστηκε η σωστή λειτουργία του αισθητήρα.

Στα πρώτα στάδια δοκιμών λήψης δεδομένων είχε αποφασιστεί η χρήση μίας SD-card ως αποθηκευτικός χώρος των μετρήσεων, μέσα στην αντίστοιχη θύρα που βρίσκεται στην πλακέτα Arduino Ethernet Shield. Στη διάρκεια όμως, των δοκιμών εντοπίστηκε ότι με τη χρήση της κάρτας αυτής δεν μπορούσε να πραγματοποιηθεί σωστά η μεταφορά των ληφθέντων δεδομένων στη βάση δεδομένων, ούτε ήταν δυνατή η εμφάνιση τους στον ιστοχώρο. Έτσι αποφασίστηκε η απευθείας αποστολή των δεδομένων στη βάση με τη χρήση του κατάλληλου κώδικα και η λήψη τους από εκεί με τη χρήση εντολών ερωτημάτων “queries” για την εμφάνισή τους στην ιστοσελίδα.

Ακόμη ένα βασικό πρόβλημα ήταν η σωστή δρομολόγηση των πλακετών έτσι ώστε να μην υπάρχει πρόβλημα κατά την επικοινωνία με τον τοπικό εξυπηρετητή. Το πρόβλημα που παρουσιάστηκε, ήταν πώς κατά την διάρκεια αποστολής των δεδομένων, κάποιες μετρήσεις δεν έφθαναν σωστά στον τελικό προορισμό τους που είναι οι πίνακες indoor_data και outdoor_data της βάσης smart_homie. Για την επίλυση αυτού του προβλήματος χρησιμοποιήθηκε ένα δρομολογητής (router), στον οποίο συνδέθηκαν οι

πλακέτες καθώς και ο τοπικός εξυπηρετητής με σκοπό την απόδοση στατικών IP σε κάθε συσκευή έτσι ώστε να γίνεται η σωστή αναγνώριση τους από το σύστημα.

Τέλος, ένα πρόβλημα για το οποίο δεν μπόρεσε να βρεθεί λύση μέσα στα χρονικά πλαίσια υλοποίησης της διπλωματικής εργασίας είναι εγκατάσταση της εφαρμογής σε ένα διακομιστή (web server), διότι με τον τρόπο που αναπτύχθηκε η εφαρμογή θα ήταν εύκολη η παραβίαση προσωπικών δεδομένων. Το πρόβλημα δεν αποτέλεσε μείζον ζήτημα καθώς η εφαρμογή έχει σχεδιαστεί για τοπική εγκατάσταση και δίνεται η δυνατότητα απομακρυσμένης χρήσης της εφαρμογής μέσω ανοίγματος συγκεκριμένης θύρας (port) στον δρομολογητή (router).

5.3 Περαιτέρω Έρευνα

Λόγω της τεχνολογικής εξέλιξης, συστήματα όπως αυτό που δημιουργήθηκε στη συγκεκριμένη διπλωματική εργασία χρήζουν συνεχούς βελτίωσης. Κατά τη διάρκεια της εκπόνησης της προέκυψαν προβλήματα που βοήθησαν στη δημιουργία ιδεών για προσθήκες ή βελτιώσεις τόσο στο υλικό μέρος του συστήματος όσο και στο λογισμικό.

Η περαιτέρω αναζήτηση τρόπων με τους οποίους θα μπορούσε να υλοποιηθεί το παρόν σύστημα ποικίλουν ανάλογα με τον γνώμονα που τίθεται από εμάς ως βασικότερος έναντι των άλλων. Εάν το χαμηλό κόστος δεν είναι προτεραιότητα, η δημιουργία ειδικά σχεδιασμένων πλακετών καθώς και η αναβάθμιση των αισθητηρίων θα βελτίωνε αρκετά την ασφάλεια, την ταχύτητα καθώς και την αξιοπιστία του συστήματος που κατασκευάσθηκε. Με τον τρόπο αυτό, μπορεί να γίνει αποφυγή ή να εντοπιστούν τυχόν λάθος μετρήσεις από κάποιο αισθητήρα που λειτουργεί εσφαλμένα. Ακόμη, ο τοπικός εξυπηρετητής που χρησιμοποιήθηκε είναι βασισμένος στο λειτουργικό σύστημα Windows. Εάν ο προηγούμενος, αντικατασταθεί με κάποιο πιο φθηνό αλλά και ανοικτού κώδικα όπως είναι το raspberry pi, οι δυνατότητες παραμετροποίησης του συστήματος αυξάνονται σημαντικά.

Βελτίωση επίσης μπορεί να γίνει και στον τρόπο σύνδεσης όλων των υποσυστημάτων. Αυτό μπορεί να επιτευχθεί με την αλλαγή των πρωτόκολλων μεταφοράς δεδομένων. Στην παρούσα διπλωματική εργασία έχει χρησιμοποιηθεί η ενσύρματη σύνδεση με καλώδιο Ethernet για τη σύνδεση του τοπικού εξυπηρετητή καθώς και για τον μικροελεγκτή Arduino Uno Rev3 (μέσω του Ethernet Shield) με τον δρομολογητή (router). Με την αλλαγή της ενσύρματης σύνδεσης σε ασύρματη μέσω πρωτόκολλου WiFi, επιτυγχάνεται η φορητότητα του συστήματος και μπορεί να εγκατασταθεί με περισσότερη ευκολία λόγω των λιγότερων σε αριθμό καλωδίων που απαιτούνται.

Τέλος, παρά το γεγονός ότι η πρόσβαση στην διαδικτυακή εφαρμογή γίνεται πέρα από ηλεκτρονικό υπολογιστή και από κινητό ή tablet μια σημαντική μελλοντική επέκταση θα μπορούσε να αποτελέσει η δημιουργία εφαρμογής λειτουργικών Android ή IOS. Η εφαρμογή αυτή θα μπορεί να εγκατασταθεί σε κινητά ή tablets αντίστοιχων λειτουργικών και θα μπορεί να εξυπηρετεί τις ίδιες απαιτήσεις με αυτές της ιστοσελίδας.

Δικτυογραφία

- [1] Web Application https://en.wikipedia.org/wiki/Web_application, Οκτώβριος 2018
- [2] Internet of Things https://en.wikipedia.org/wiki/Internet_of_things, Οκτώβριος 2018
- [3] What Is the Internet of Things? <https://www.cloudwards.net/what-is-the-internet-of-things/>, Ιούνιος 2018
- [4] Programming Language https://en.wikipedia.org/wiki/Programming_language, Οκτώβριος 2018
- [5] CSS <https://el.wikipedia.org/wiki/CSS>, Οκτώβριος 2018
- [6] CSS Basics
https://developer.mozilla.org/enUS/docs/Learn/Getting_started_with_the_web/CSS_basics, Οκτώβριος 2018
- [7] PHP <https://el.wikipedia.org/wiki/PHP>, Οκτώβριος 2018
- [8] PHP <http://php.net/>, Οκτώβριος 2018
- [9] PHP: Superglobals <http://php.net/manual/en/language.variables.superglobals.php>, Οκτώβριος 2018
- [10] JavaScript <https://el.wikipedia.org/wiki/JavaScript>, Οκτώβριος 2018
- [11] Famous Desktop Apps Using Electron <https://brainhub.eu/blog/7-famous-desktop-apps-using-electron/>, Οκτώβριος 2018
- [12] MySQL <https://en.wikipedia.org/wiki/MySQL>, Οκτώβριος 2018
- [13] ACID <https://el.wikipedia.org/wiki/ACID>, Οκτώβριος 2018
- [14] MySQL <https://www.oracle.com/technetwork/database/mysql/index.html>, Οκτώβριος 2018
- [15] History of Arduino http://creativityprojects.blogspot.com/2013/03/history-of-arduino_4195.html, Οκτώβριος 2018

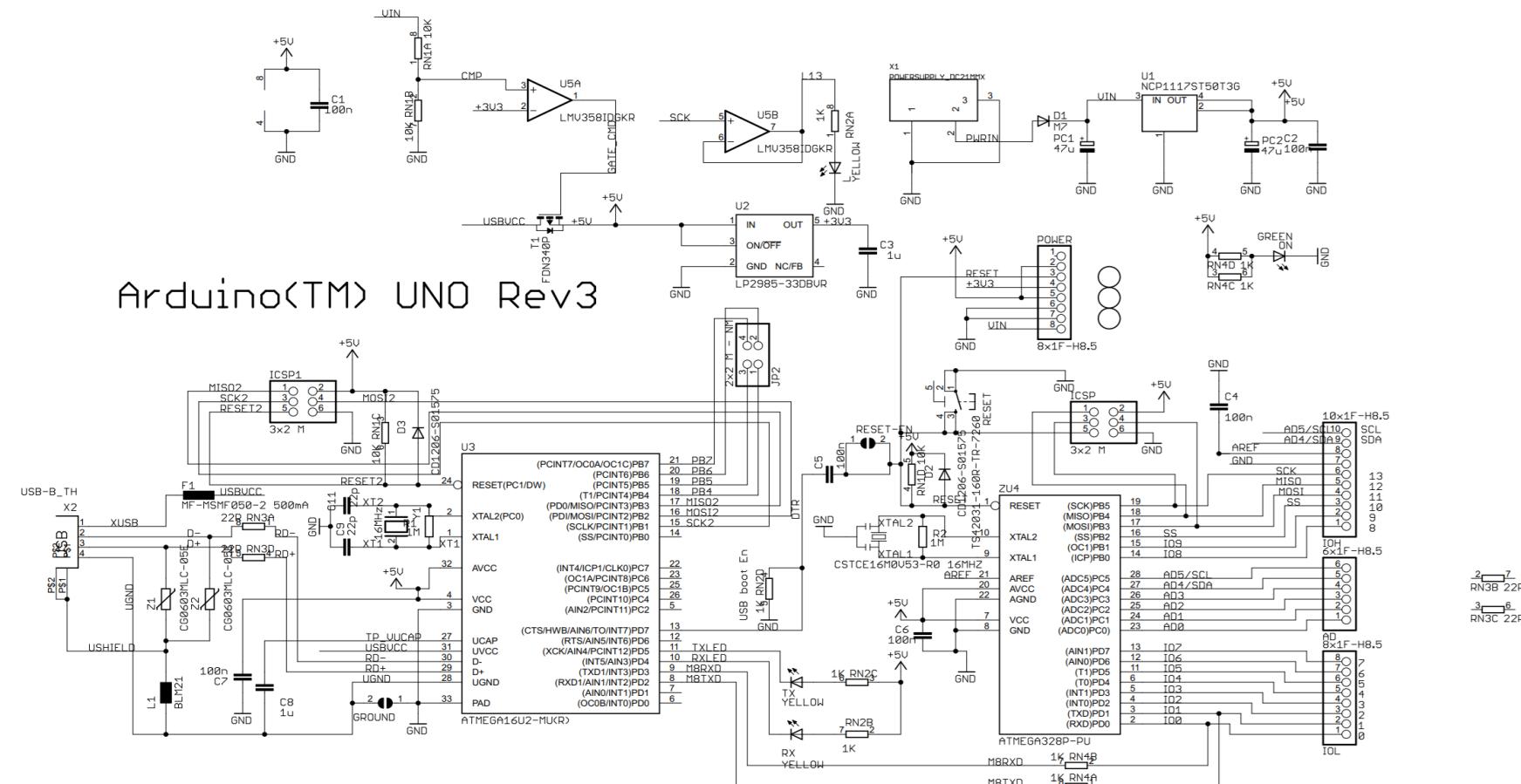
- [16] Arduino Development Kits: Advantages and Features
<https://www.semiconductorstore.com/blog/2014/Arduino-Development-Kits-Advantages-and-Features/811/>, Οκτώβριος 2018
- [17] WeMos Wiki <https://wiki.wemos.cc/start>, Οκτώβριος 2018
- [18] Δρομολογητής
<https://el.wikipedia.org/wiki/%CE%94%CF%81%CE%BF%CE%BC%CE%BF%CE%BB%CE%BF%CE%B3%CE%B7%CF%84%CE%AE%CF%82>, Οκτώβριος 2018
- [19] Visual Studio Code <https://code.visualstudio.com/>, Οκτώβριος 2018
- [20] Why VS Code <https://code.visualstudio.com/docs/editor/whyscode>, Οκτώβριος 2018
- [21] Brackets <http://brackets.io/>, Οκτώβριος 2018
- [22] XAMPP Installers <https://www.apachefriends.org/index.html>, Οκτώβριος 2018
- [23] DHT.lib <https://github.com/ringerc/Arduino-DHT22>, Οκτώβριος 2018
- [24] Προσωπικά Δεδομένα
<https://el.wikipedia.org/wiki/%CE%A0%CF%81%CE%BF%CF%83%CF%89%CF%80%CE%BA%CE%AC%CE%94%CE%B5%CE%B4%CE%BF%CE%BC%CE%AD%CE%BD%CE%B1>, Οκτώβριος 2018
- [25] PHP: password_hash <http://php.net/manual/en/function.password-hash.php>, Οκτώβριος 2018
- [26] SSL <https://el.wikipedia.org/wiki/SSL>, Οκτώβριος 2018

Βιβλιογραφία

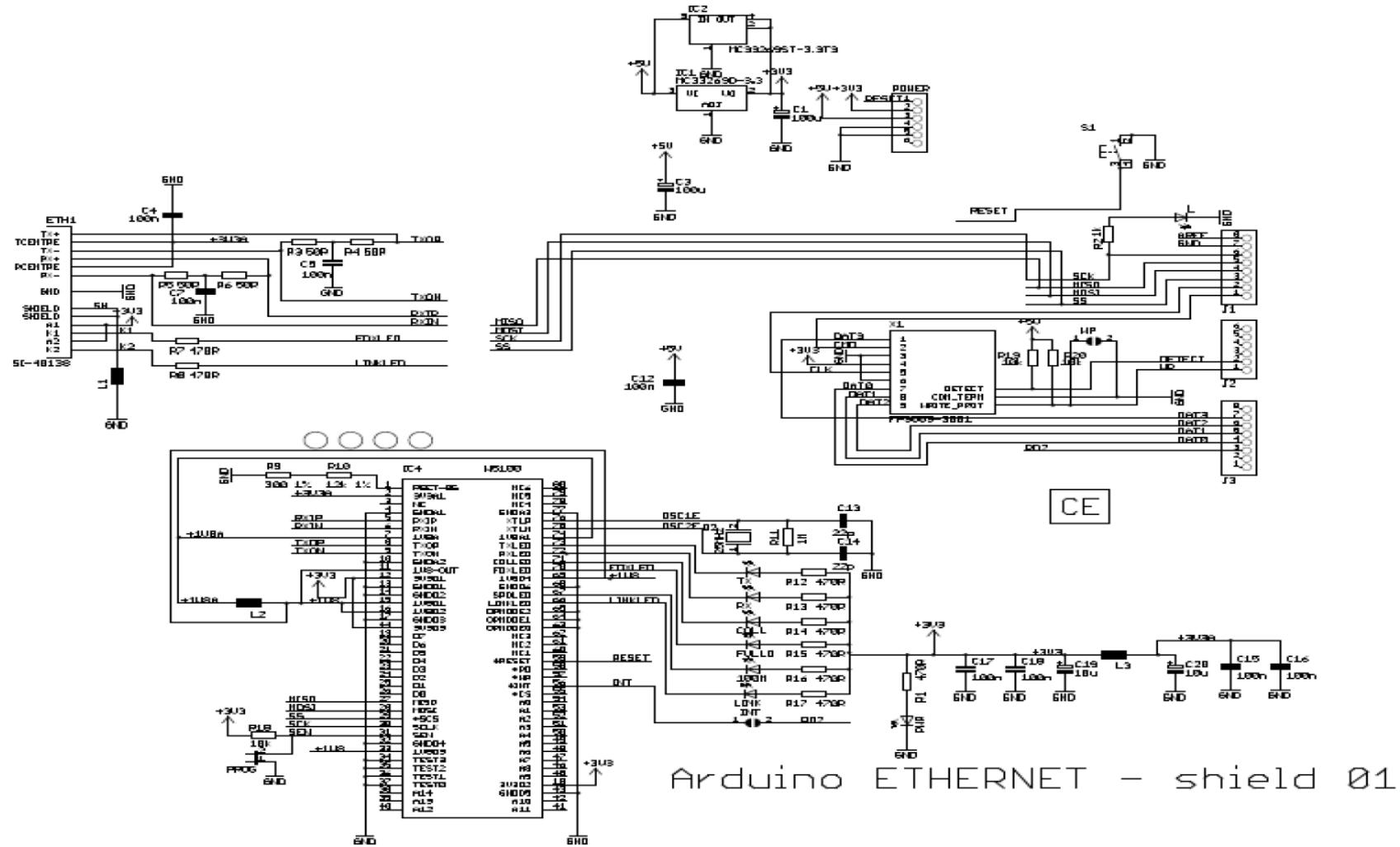
- [27] Bassi A. Bauer M., Fiedler M., Kramp T., Kranenburg R., Lange S., Meissner S. (2013), Enabling Things to Talk: Designing IOT solutions with the IOT architectural Reference Model, New York.
- [28] Mark J. Collins (2017). Pro HTML5 with CSS, JavaScript, and Multimedia: Complete Website Development and Best Practices
- [29] Rose K., Eldrigde S., Chapin L. (2015), The Internet of Things: An Overview | Understanding the Issues and Challenges of a More Connected World
- [30] Saxena, S., & Ali Said Mansour Al-Tamimi, T. (2017). Big data and Internet of Things (IoT) technologies in Omani banks: a case study. *Foresight*, 19(4), 409–420.
- [31] Παυλή Β. (2013), Η Διδασκαλία εκπαιδευτικής ρομποτικής με τη χρήση μικροελεγκτών (π.χ. ARDUINO, PIC), Διπλωματική Εργασία, Τ.Ε.Ι. Λάρισας, Τμήμα Τεχνολογίας Πληροφορικής και Τηλεπικοινωνιών, Λάρισα

Παράρτημα Α

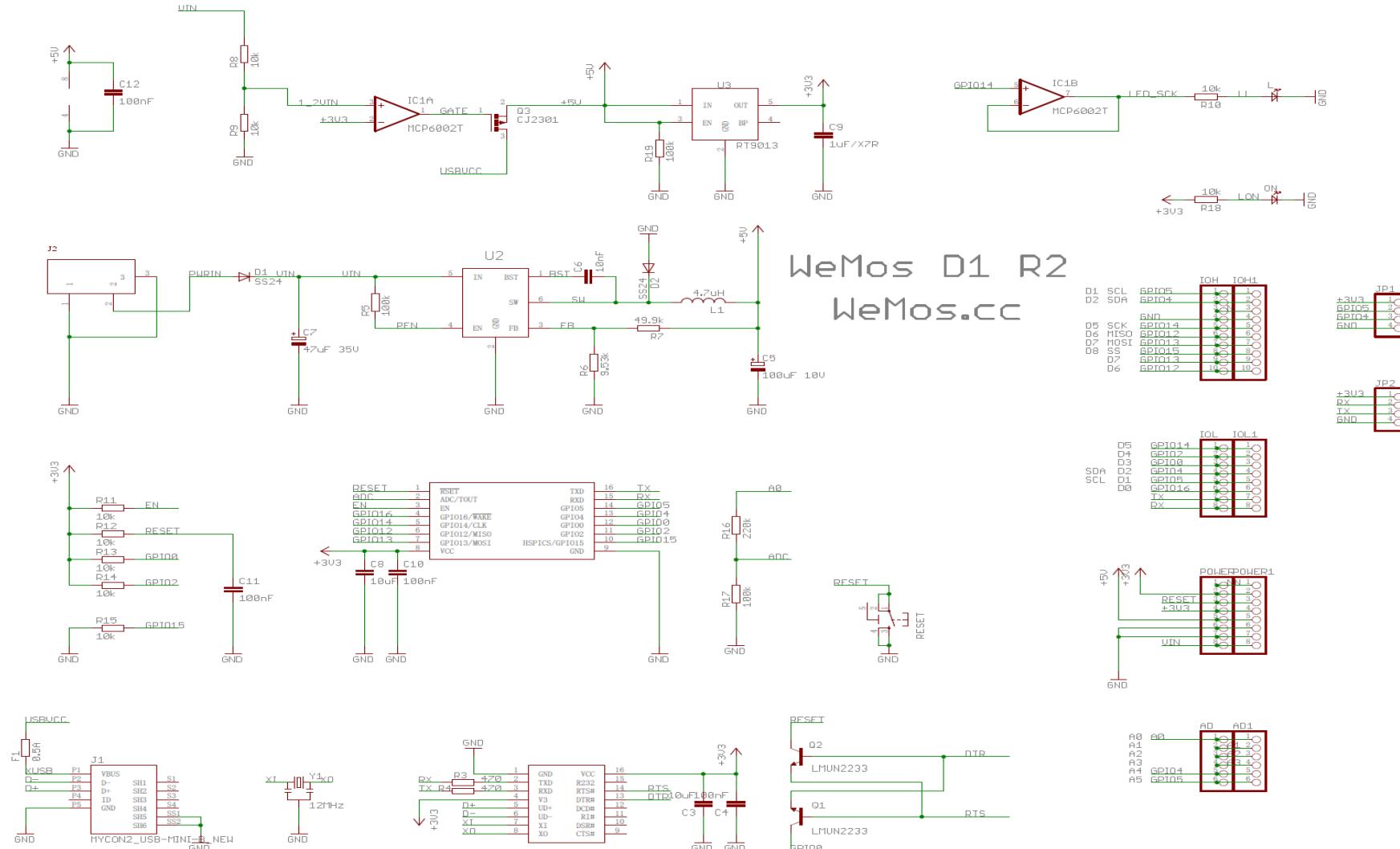
Σχηματική αναπαράσταση Arduino Uno Rev3



Σχηματική αναπαράσταση Arduino Ethernet Shield



Σχηματική αναπαράσταση WeMos D1 R2



Παράρτημα Β

Πηγαίοι Κώδικες

1. Κώδικας Arduino Uno Rev3

`indoor_sensor_data.ino`

```
#include <SPI.h>
#include <Ethernet.h>
#include <dht.h>
#define dataPin 2 // Defines pin number to which the sensor is connected
dht DHT; // Creates a DHT object

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
byte ip[] = {192, 168, 1, 76 }; //Read the code explanation below
byte serv[] = {192, 168, 1, 100} ; //Read the code explanation below
EthernetClient cliente;
void setup()
{
delay(3000);
Serial.begin(9600); //setting the baud rate at 9600
Ethernet.begin(mac, ip);
Serial.println(Ethernet.localIP());
}
void loop() {
//float hum = dht.readHumidity(); //Reading the humidity and storing in
hum
//float temp = dht.readTemperature(); //Reading the temperature as
Celsius and storing in temp
//float fah = dht.readTemperature(true); //reading the temperature in
Fahrenheit
//float heat_index = dht.computeHeatIndex(fah, hum); //Reading the heat
index in Fahrenheit
//float heat_indexC = dht.convertFtoC(heat_index); //Converting the heat
index in Celsius

int readData = DHT.read22(dataPin); // Reads the data from the sensor
float t = DHT.temperature; // Gets the values of the temperature
float h = DHT.humidity; // Gets the values of the humidity
if (cliente.connect(serv, 8080)) { //Connecting at the IP address and
port we saved before
Serial.println("connected");
cliente.print("GET /MyThesis/Arduino/indoor_data/data.php?");
//Connecting and Sending values to database
cliente.print("temperature=");
cliente.print(t);
cliente.print("&humidity=");
cliente.println(h);
//Printing the values on the serial monitor
Serial.print("Temperature= ");
Serial.print("Temperature= ");




```

```

Serial.println(t);
Serial.print("Humidity= ");
Serial.println(h);
cliente.stop(); //Closing the connection
}
else {
// if you didn't get a connection to the server:
Serial.println("connection failed");
}
delay(300000);
}

```

data.php (επικοινωνία arduino με βάση δεδομένων)

```

<?php
/* Database credentials. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_NAME', 'smart_homie');

/* Attempt to connect to MySQL database */
$link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);

$sql = "INSERT INTO indoor_data (date,time,temperature, humidity) VALUES
(CURRENT_DATE(), CURRENT_TIME(), '".$_GET["temperature"]."',
'".$_GET["humidity"]."')";
mysqli_query($link,$sql);

echo "success"
?>

```

2. Κώδικας WeMos D1 R2

outdoor_sensor_data.ino

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
#include <ESP8266mDNS.h>
#include "DHT.h"
#include "Adafruit_Sensor.h"

#define DHTPIN D2      // what digital pin we're connected to
#define DHTTYPE DHT22  // DHT 22 (AM2302), AM2321

DHT dht(DHTPIN, DHTTYPE);

const char* ssid    = "SmartHomie";
const char* password = "!smarthomie10!";

IPAddress ip(192, 168, 1, 77);
IPAddress gateway(192, 168, 1, 1);
IPAddress subnet(255, 255, 255, 0);
IPAddress dns1(192, 168, 1, 1);
IPAddress dns2(192, 168, 1, 2);

ESP8266WebServer server(8080);
WiFiClient client;
MDNSResponder mdns; //multicast Domain Name System

float humidity, temp_f; // Values read from sensor
String webPage = "";

void setup(void)
{
  DHT dht(DHTPIN, DHTTYPE);
  webPage += "<h1>Wemos D1 R2, Outdoor Data Sensor</h1>";
  WiFi.config(ip, gateway, subnet, dns1, dns2);
  Serial.begin(115200); //baude rate .... Serial connection from ESP-01
  via 3.3v console cable
  delay(1000);
  Serial.println("Hii ");
  // Connect to WiFi network
  WiFi.begin(ssid, password);
  Serial.print("\n\r \n\rWorking to connect");

  // Wait for connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.print("Connected to ");
  Serial.println(ssid);
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
  if (mdns.begin("esp8266", WiFi.localIP())) {
    Serial.println("MDNS responder started");
  }
}
```

```

server.on("/", [](){
  server.send(200, "text/html", webPage);
});
server.begin();
Serial.println("HTTP server started");
delay(10000);

}

void loop(void)
{
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  // Wait a few seconds between measurements.

  server.handleClient();

  if (client.connect("192.168.1.100", 8080)) {
    Serial.println("connected to server");

    client.print("GET /MyThesis/Arduino/outdoor_data/data.php?"); // This
    client.print("temperature=");
    client.print(t);
    client.print("&humidity=");
    client.println(h);
    Serial.print("Humidity=");
    Serial.print(h);
    Serial.print(" Temp=");
    Serial.println(t);
    client.println(" HTTP/1.1"); // Part of the GET request
    client.println("Host: 192.168.1.100"); // IMPORTANT: If you are using
    XAMPP you will have to find out the IP address of your computer and put
    it here
    client.println("Connection: close"); // Part of the GET request
    telling the server that we are over transmitting the message
    client.println(); // Empty line
    client.println(); // Empty line
    client.stop(); // Closing connection to server
  }
  else {
    // If Arduino can't connect to the server (your computer or web page)
    Serial.println("--> connection failed\n");
  }

  delay(300000);
}

```

data.php (επικοινωνία arduino με βάση δεδομένων)

```
<?php
/* Database credentials. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_NAME', 'smart_homie');

/* Attempt to connect to MySQL database */
$link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);

$sql = "INSERT INTO outdoor_data (date,time,temperature, humidity)
VALUES (CURRENT_DATE(), CURRENT_TIME(), '".$_GET["temperature"]."',
'".$_GET["humidity"]."')";

mysqli_query($link,$sql);

echo "success"
?>
```

3. Κώδικας Arduino Controller

devices_controller.ino

```
//-----
//-----
#include <SPI.h>
#include <Ethernet.h>

// Enter a MAC address and IP address for your controller below.
byte mac[] = {0x90, 0xA2, 0xDA, 0x0D, 0x48, 0xD3 };

// The IP address will be dependent on your local network:
// assign an IP address for the controller:

IPAddress ip(192,168,1,75);
IPAddress gateway(192,168,1,254);
IPAddress subnet(255, 255, 255, 0);

// Initialize the Ethernet server library with the port you want to use.
EthernetServer server(8080);
String readString;
//-----
//-----
// Any extra codes for Declaration :

// Declare Pin 8 as an LED because thats what we will be connecting the
// LED to. You could use any other pin and would then have to change the pin
// number.
int AC = 8;
int DH = 7;

//-----
//-----
void setup()
{
//-----

// Extra Set up code:
pinMode(AC, OUTPUT); //pin selected to control
pinMode(DH, OUTPUT); //pin selected to control

//-----
//-----
//enable serial data print
Serial.begin(9600);

//start Ethernet
Ethernet.begin(mac, ip, gateway, subnet);
server.begin();
Serial.print("Server is at ");
Serial.println(Ethernet.localIP());
Serial.println("LED Controller Test 1.0");
}
//-----
//-----
```

```

//-----
-----

void loop()
{
    // listen for incoming clients
    EthernetClient client = server.available();
    if (client)

    {
        Serial.println("new client");

        while (client.connected())
        {
            if (client.available())

            {
                char c = client.read();

                //read char by char HTTP request
                if (readString.length() < 100)

                {

                    //store characters to string
                    readString += c;
                    //Serial.print(c);

                    Serial.write(c);
                    // if you've gotten to the end of the line (received a newline
                    // character) and the line is blank, the http request has ended,
                    // so you can send a reply
                    //if HTTP request has ended
                    if (c == '\n') {
                        Serial.println(readString); //print to serial monitor for debugging
                    }
                }
            }
        }
    }
}

// Needed to Display Site:
client.println("HTTP/1.1 200 OK"); //send new page
    client.println("Content-Type: text/html");
    client.println();
    client.println("<HTML>");
    client.println("<HEAD>");

//-----
-----



// what is being Displayed :

    client.println("<TITLE>Home Automation</TITLE>");
    client.println("<center>");
    client.println("<link rel='stylesheet' href='http://192.168.1.100:8080/MyThesis/LiveSite/dashboard/css/arduino_controller.css'>");
        client.println("</HEAD>");
        client.println("<BODY>");
        client.println("<center>");



```

```

        client.println("<div class='cs-loader'>");
        client.println("<div class='cs-loader-inner'>");
        client.println("<label> .</label>");
        client.println("<label> .</label>");
        client.println("<label> .</label>");
        client.println("<label> .</label>");
        client.println("<label> .</label>");
        client.println("<label> .</label>");

        client.println("</div>");
        client.println("</div>");

        client.println("<script>");
        client.println("if (window.location.pathname +
window.location.search == '/?acon') {}");

        client.println("window.location.replace('http://192.168.1.100:8080/MyThe
sis/LiveSite/dashboard/index.php?acon');");
        client.println("{} else {}");
        client.println("console.log(location.pathname +
window.location.search); }");
        client.println("if (window.location.pathname +
window.location.search == '/?acoff') {}");

        client.println("window.location.replace('http://192.168.1.100:8080/MyThe
sis/LiveSite/dashboard/index.php?acoff');");
        client.println("{} else {}");
        client.println("console.log(location.pathname +
window.location.search); }");
        client.println("if (window.location.pathname +
window.location.search == '/?dhon') {}");

        client.println("window.location.replace('http://192.168.1.100:8080/MyThe
sis/LiveSite/dashboard/index.php?dhon');");
        client.println("{} else {}");
        client.println("console.log(location.pathname +
window.location.search); }");
        client.println("if (window.location.pathname +
window.location.search == '/?dhoff') {}");

        client.println("window.location.replace('http://192.168.1.100:8080/MyThe
sis/LiveSite/dashboard/index.php?dhoff');");
        client.println("{} else {}");
        client.println("console.log(location.pathname +
window.location.search); }");
        client.println("</script>");
        client.println("</BODY>");
        client.println("</HTML>");

delay(1);
//stopping client
client.stop();

-----
// Code which needs to be Implemented:
if(readString.indexOf("?acon") >0)//checks for on
{
    digitalWrite(AC, HIGH); // set pin 8 high
}

```

```

    Serial.println("AC On");

}
else{
  if(readString.indexOf("?acoff") >0)//checks for off
  {
    digitalWrite(AC, LOW); // set pin 8 low
    Serial.println("AC Off");
  }
}
if(readString.indexOf("?dhon") >0)//checks for on
{
  digitalWrite(DH, HIGH); // set pin 8 high
  Serial.println("Dehumidifier On");

}
else{
  if(readString.indexOf("?dhoff") >0)//checks for off
  {
    digitalWrite(DH, LOW); // set pin 8 low
    Serial.println("Dehumidifier Off");
  }
}
//clearing string for next read
readString="";

// give the web browser time to receive the data
delay(1);
// close the connection:
client.stop();
Serial.println("client disconnected");

}
}
}
}
}

status_update.php (επικοινωνία controller με βάση δεδομένων)

<?php
/* Database credentials. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_NAME', 'smart_homie');

/* Attempt to connect to MySQL database */
$link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);

$sql = "UPDATE devices_states SET status = 'on' WHERE id = 1";
mysqli_query($link,$sql);

echo "success"
?>

```

4. Κώδικας Διαδικτυακής Εφαρμογής

register.php (εγγραφή χρήστη στη διαδικτυακή εφαρμογή)

```
<?php
// Include config file
require_once 'config.php';

// Define variables and initialize with empty values
$username = $email = $password = $confirm_password = "";
$username_err = $email_err = $password_err = $confirm_password_err = "";

// Processing form data when form is submitted
if($_SERVER["REQUEST_METHOD"] == "POST") {

    // Validate username
    if(empty(trim($_POST["username"]))){
        $username_err = "Please enter a username.";
    } else{
        // Prepare a select statement
        $sql = "SELECT id FROM users WHERE username = ?";

        if($stmt = mysqli_prepare($link, $sql)){
            // Bind variables to the prepared statement as parameters
            mysqli_stmt_bind_param($stmt, "s", $param_username);

            // Set parameters
            $param_username = trim($_POST["username"]);

            // Attempt to execute the prepared statement
            if(mysqli_stmt_execute($stmt)){
                /* store result */
                mysqli_stmt_store_result($stmt);

                if(mysqli_stmt_num_rows($stmt) == 1){
                    $username_err = "This username is already taken.";
                } else{
                    $username = trim($_POST["username"]);
                }
            } else{
                echo "Oops! Something went wrong. Please try again later.";
            }
        }
        // Close statement
        mysqli_stmt_close($stmt);
    }

    // Validate Email
    if(empty(trim($_POST['email']))){
        $email_err = "Please enter an email.";
    } else{
        $email = trim($_POST['email']);
    }

    // Validate password
    if(empty(trim($_POST['password']))){
        $password_err = "Please enter a password.";
    }
}
```

```

elseif(strlen(trim($_POST['password'])) < 6){
    $password_err = "Password must have atleast 6 characters.";
} else{
    $password = trim($_POST['password']);
}

// Validate confirm password
if(empty(trim($_POST["confirm_password"]))) {
    $confirm_password_err = 'Please confirm password.';
} else{
    $confirm_password = trim($_POST['confirm_password']);
    if($password != $confirm_password) {
        $confirm_password_err = 'Password did not match.';
    }
}

// Check input errors before inserting in database
if(empty($username_err) && empty($password_err) &&
empty($confirm_password_err)) {

    // Prepare an insert statement
    $sql = "INSERT INTO users (username, email, password) VALUES (?,?,?,?)";
    if($stmt = mysqli_prepare($link, $sql)){
        // Bind variables to the prepared statement as parameters
        mysqli_stmt_bind_param($stmt, "sss", $param_username,$param_email,
$param_password);

        // Set parameters
        $param_username = $username;
        $param_email = $email;
        $param_password = password_hash($password, PASSWORD_DEFAULT); // Creates a password hash

        // Attempt to execute the prepared statement
        if(mysqli_stmt_execute($stmt)){
            // Redirect to login page
            header("location: login.php");
        } else{
            echo "Something went wrong. Please try again later.";
        }
    }

    // Close statement
    mysqli_stmt_close($stmt);
}

// Close connection
mysqli_close($link);
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <title>Smart Homie - Register</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-

```

```

scale=1">
<!--
=====
=====-->
    <link rel="icon" type="image/png"
href="images/icons/favicon.ico"/>
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css"
href="vendor/bootstrap/css/bootstrap.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css" href="fonts/font-awesome-
4.7.0/css/font-awesome.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css" href="fonts/Linearicons-
Free-v1.0.0/icon-font.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css"
href="vendor/animate/animate.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css" href="vendor/css-
hamburgers/hamburgers.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css"
href="vendor/select2/select2.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css" href="css/util.css">
    <link rel="stylesheet" type="text/css" href="css/main.css">
<!--
=====
=====-->
</head>
<body>

    <div class="limiter">
        <div class="container-login100">
            <div class="wrap-login100 p-t-190 p-b-30">
                <form class="login100-form validate-form"
action="php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?" ?>">
method="post">
                    <span class="login100-form-title p-t-20 p-
b-45">
                        Register
                    </span>

```

```

<div class="wrap-input100 validate-input m-b-10" data-validate =
"Username is required">
    <input class="input100" type="text"
name="username" placeholder="Username" value="<?php echo $username; ?>">
    <span class="focus-input100"></span>
    <span class="symbol-input100">
        <i class="fa fa-user"></i>
    </span>
</div>

<div class="wrap-input100 validate-input m-b-10" data-validate =
= "Email is required">
    <input class="input100" type="text"
name="email" placeholder="Email" value="<?php echo $email; ?>">
    <span class="focus-input100"></span>
    <span class="symbol-input100">
        <i class="fa fa-envelope"></i>
    </span>
</div>

<div class="wrap-input100 validate-input
m-b-10" data-validate = "Password is required">
    <input class="input100"
type="password" name="password" placeholder="Password" value="<?php echo
$password; ?>">
    <span class="focus-input100"></span>
    <span class="symbol-input100">
        <i class="fa fa-lock"></i>
    </span>
</div>

<div class="wrap-input100 validate-input m-b-10" data-validate =
= "Password is required">
    <input class="input100"
type="password" name="confirm_password" placeholder="Confirm Password"
value="<?php echo $confirm_password; ?>">
    <span class="focus-input100"></span>
    <span class="symbol-input100">
        <i class="fa fa-lock"></i>
    </span>
</div>

<div class="container-login100-form-btn p-
t-10 p-b-230">
    <button class="login100-form-btn wi-
40">
        Register
    </button>
    <button class="login100-form-btn wi-40 mal-20" type="reset">
        Reset
    </button>
</div>

<div class="text-center w-full">
    <a class="txt1" href="login.php">
        Login to your account
        <i class="fa fa-long-arrow-
right"></i>
    </a>

```

```

                </div>
            </form>
        </div>
    </div>
</div>

<!--
=====
=====-->
<script src="vendor/jquery/jquery-3.2.1.min.js"></script>
<!--
=====
=====-->
<script src="vendor/bootstrap/js/popper.js"></script>
<script src="vendor/bootstrap/js/bootstrap.min.js"></script>
<!--
=====
=====-->
<script src="vendor/select2/select2.min.js"></script>
<!--
=====
=====-->
<script src="js/main.js"></script>

</body>
</html>

```

login.php (σύνδεση του χρήστη στη διαδικτυακή εφαρμογή)

```

<?php
// Include config file
require_once 'config.php';

// Define variables and initialize with empty values
$username = $email = $adminuser = $password = $id = "";
$username_err = $email_err = $adminuser_err = $password_err = "";

$log_data = " ";
// Processing form data when form is submitted
if($_SERVER["REQUEST_METHOD"] == "POST"){

    // Check if username is empty
    if(empty(trim($_POST["username"]))){
        $username_err = 'Please enter username.';
    } else{
        $username = trim($_POST["username"]);
    }

    // Check if password is empty
    if(empty(trim($_POST['password']))){
        $password_err = 'Please enter your password.';
    } else{
        $password = trim($_POST['password']);
    }
}

```

```

// Validate credentials
if(empty($username_err) && empty($password_err)){
    // Prepare a select statement
    $sql = "SELECT id,username, email, adminuser, password FROM users
WHERE username = ?";

    if($stmt = mysqli_prepare($link, $sql)){
        // Bind variables to the prepared statement as parameters
        mysqli_stmt_bind_param($stmt, "s", $param_username);

        // Set parameters
        $param_username = $username;

        // Attempt to execute the prepared statement
        if(mysqli_stmt_execute($stmt)){
            // Store result
            mysqli_stmt_store_result($stmt);

            // Check if username exists, if yes then verify password
            if(mysqli_stmt_num_rows($stmt) == 1){
                // Bind result variables
                mysqli_stmt_bind_result($stmt, $id,$username, $email,
                $adminuser, $hashed_password);
                if(mysqli_stmt_fetch($stmt)){
                    if(password_verify($password, $hashed_password)){
                        /* Password is correct, so start a new session and
                        save the username to the session */
                        session_start();
                        $_SESSION['username'] = $username;
                        $_SESSION['password'] = $password;
                        $_SESSION['adminuser'] = $adminuser;
                        $_SESSION['email'] = $email;
                        $_SESSION['id'] = $id;

                        // TRACK LOGOUT ACTIVITY

$conn_log=mysqli_connect("localhost","root","","smart_homie");
                    // Check connection
                    if (mysqli_connect_errno())
                    {
                        echo "Failed to connect to MySQL: " .
                    mysqli_connect_error();
                    }

                    // Perform queries
                    mysqli_query($conn_log,"INSERT INTO user_log
(user_id,date,time,log) VALUES ('$id',CURRENT_DATE(),
CURRENT_TIME(),'Logged In')");

                    // close connection
                    mysqli_close($conn_log);

                    header("location: ../dashboard/index.php");
                } else{
                    // Display an error message if password is not valid
                    $password_err = 'The password you entered was not valid.';
                }
            }
        }
    }
}

```

```

        }
    } else{
        // Display an error message if username doesn't exist
        $username_err = 'No account found with that username.';
    }
} else{
    echo "Oops! Something went wrong. Please try again later.";
}
}

// Close statement
mysqli_stmt_close($stmt);
}

$log_data = "INSERT INTO user_log (user_id,date,time,log) VALUES
($username,CURRENT_DATE(), CURRENT_TIME(),'logged in')";
mysqli_query($link,$log_data);

// Close connection
mysqli_close($link);
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <title>Smart Homie - Login</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
<!--
=====
=====-->
    <link rel="icon" type="image/png"
href="images/icons/favicon.ico"/>
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css"
href="vendor/bootstrap/css/bootstrap.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css" href="fonts/font-awesome-
4.7.0/css/font-awesome.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css" href="fonts/Linearicons-
Free-v1.0.0/icon-font.min.css">
<!--
=====
=====-->
    <link rel="stylesheet" type="text/css"
href="vendor/animate/animate.css">
<!--
=====
=====-->

```

```

        <link rel="stylesheet" type="text/css" href="vendor/css-
hamburgers/hamburgers.min.css">
<!--
=====
=====-->
        <link rel="stylesheet" type="text/css"
href="vendor/select2/select2.min.css">
<!--
=====
=====-->
        <link rel="stylesheet" type="text/css" href="css/util.css">
        <link rel="stylesheet" type="text/css" href="css/main.css">
        <link rel="stylesheet" type="text/css" href="css/main2.css">
<!--
=====
=====-->
</head>
<body>

        <div class="limiter">
            <div class="container-login100">
                <div class="wrap-login100 p-t-190 p-b-30">
                    <form class="login100-form validate-form"
action=<?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>">
method="post" >
                    <span class="login100-form-title p-t-20 p-
b-45">
                        Log in
                    </span>

                    <div class="wrap-input100 validate-input
m-b-10" data-validate = "Username is required">
                        <input class="input100" type="text"
name="username" placeholder="Username" value=<?php echo $username; ?>">
                        <span class="focus-input100"></span>
                        <span class="symbol-input100">
                            <i class="fa fa-user"></i>
                        </span>
                    </div>

                    <span style="color:#fff; padding-left:40px; margin-
bottom:10px;"><?php echo $username_err; ?></span>

                    <div class="wrap-input100 validate-input
m-b-10" data-validate = "Password is required">
                        <input class="input100"
type="password" name="password" placeholder="Password">
                        <span class="focus-input100"></span>
                        <span class="symbol-input100">
                            <i class="fa fa-lock"></i>
                        </span>
                    </div>

                    <span style="color:#fff; padding-left:40px; margin-
bottom:10px;"><?php echo $password_err; ?></span>

```

```

        <div class="container-login100-form-btn p-
t-10 p-b-230">
            <button class="login100-form-btn
wid-50">
                Login
            </button>
        </div>

        </form>
    </div>
</div>

<!--
=====
=====-->
<script src="vendor/jquery/jquery-3.2.1.min.js"></script>
<!--
=====
=====-->
<script src="vendor/bootstrap/js/popper.js"></script>
<script src="vendor/bootstrap/js/bootstrap.min.js"></script>
<!--
=====
=====-->
<script src="vendor/select2/select2.min.js"></script>
<!--
=====
=====-->
<script src="js/main.js"></script>

</body>
</html>

```

logout.php (αποσύνδεση του χρήστη από την διαδικτυακή εφαρμογή)

```

<?php
session_start();
// TRACK LOGOUT ACTIVITY
$conn_log=mysqli_connect("localhost","root","","smart_homie");

// Check connection
if (mysqli_connect_errno())
{
    echo "Failed to connect to MySQL: " . mysqli_connect_error();
}

// Perform queries
$test = $_SESSION['id'];
mysqli_query($conn_log,"INSERT INTO user_log (user_id,date,time,log)
VALUES ('$test',CURRENT_DATE(), CURRENT_TIME(),'Logged Out')");

// close connection
mysqli_close($conn_log);
session_destroy();

```

```

// Initialize the session
session_start();

// Unset all of the session variables
$_SESSION = array();

// Destroy the session.
session_destroy();

// Redirect to login page
header("location: ../index.html");
exit;
?>
```

config.php (ρυθμίσεις για την σύνδεση στη βάση δεδομένων)

```

<?php
/* Database credentials. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
define('DB_SERVER', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', '');
define('DB_NAME', 'Smart_Homie');

/* Attempt to connect to MySQL database */
$link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);

// Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
?>
```

4. Κώδικας Διαδικτυακής Εφαρμογής (Βασικές καρτέλες)

Στατικά Αρχεία

head.html

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <!-- Tell the browser to be responsive to screen width -->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- Favicon icon -->

    <link rel="icon" type="image/png"
        href="../../assets/images/icons/favicon.ico"/>

    <!-- Bootstrap Core CSS -->
    <link href="../../assets/plugins/bootstrap/css/bootstrap.min.css"
        rel="stylesheet">

    <!-- Font Awesome 4.7.0 -->
    <link rel="stylesheet" href="../../assets/css/fontawesome-
4.7.min.css">

    <!-- Custom CSS -->
    <link href="css/style.css" rel="stylesheet">
    <link href="css/main.css" rel="stylesheet">

    <!-- You can change the theme colors from here -->
    <link href="css/colors/blue.css" id="theme" rel="stylesheet">

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and
media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via
file:// -->
    <!--[if lt IE 9]>
        <script
            src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
        <script
            src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></scri
pt>
    <![endif]-->
```

upperbody.php

```
<body class="fix-header fix-sidebar card-no-border"
    onload="startTime()">
    <!-- ===== -->
>
```

```

<!-- Preloader - style you can find in spinners.css -->
<!-- ===== -->
>
<div class="preloader">
    <svg class="circular" viewBox="25 25 50 50">
        <circle class="path" cx="50" cy="50" r="20" fill="none" stroke-width="2" stroke-miterlimit="10" /> </svg>
    </div>
<!-- ===== -->
>
<!-- Main wrapper - style you can find in pages.scss -->
<!-- ===== -->
>
<div id="main-wrapper">
    <!-- ===== -->
-->
    <!-- Topbar header - style you can find in pages.scss -->
    <!-- ===== -->
-->
    <header class="topbar">
        <nav class="navbar top-navbar navbar-toggleable-sm navbar-light">
            <!--
=====
            <!-- Logo -->
            <!--
=====
            <div class="navbar-header">
                <a class="navbar-brand" href="index.php">
                    <!-- Logo icon --><b>
                        <!--You can put here icon as well // <i class="wi wi-sunset"></i> //-->
                    <!-- Light Logo icon -->
                    
                </b>
                <!--End Logo icon -->
                <!-- Logo text --><span style="">
                    <!-- Light Logo text -->
                    </span> </a>
                </div>
                <!--
=====
                <!-- End Logo -->
                <!--
=====
                <div class="navbar-collapse">
                    <!--
=====
                    <!-- toggle and nav items -->
                    <!--
=====
                    <ul class="navbar-nav mr-auto mt-md-0">
                        <!-- This is -->
                        <li class="nav-item">
                            <a class="nav-link nav-toggler hidden-md-up text-muted waves-effect waves-dark" href="javascript:void(0)">

```

```

        <i class="mdi mdi-menu"></i>
    </a>
</li>
<!--
=====
-->
<!-- Search -->
<!--
=====
-->
<li class="nav-item hidden-sm-down search-box">
    <a class="nav-link hidden-sm-down text-muted waves-effect waves-dark" href="javascript:void(0)">
        <i class="ti-search disp-not"></i>
    </a>
    <form class="app-search">
        <input type="text" class="form-control" placeholder="Search & enter">
        <a class="srh-btn">
            <i class="ti-close"></i>
        </a>
    </form>
</li>
<!-- Time -->
<!--
=====
-->
<li class="nav-item hidden-sm-down">
    <a class="nav-link hidden-sm-down text-muted waves-effect waves-dark" href="#">
        <div id="clock"></div>
    </a>
</li>

</ul>
<!--
=====
-->
<!-- User profile and search -->
<!--
=====
-->
<ul class="navbar-nav my-lg-0">
    <!--
=====
-->
<!-- Profile -->
<!--
=====
-->
<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle text-muted waves-effect waves-dark" href="" data-toggle="modal" aria-haspopup="true" aria-expanded="false"
        data-target="#LogoutModal">
        <i class="fa fa-user-circle-o user-icon" aria-hidden="true"></i>
        <?php echo htmlspecialchars($_SESSION['username']); ?>
    </a>
</li>
</ul>
</div>
</nav>
</header>
<!-- ===== -->

```

```

-->
    <!-- End Topbar header -->
    <!-- ===== -->
-->
    <!-- ===== -->
-->
    <!-- Left Sidebar - style you can find in sidebar.scss -->
    <!-- ===== -->
-->
<aside class="left-sidebar">
    <!-- Sidebar scroll-->
    <div class="scroll-sidebar">
        <!-- Sidebar navigation-->
        <nav class="sidebar-nav">
            <ul id="sidebarnav">
                <li>
                    <a class="waves-effect waves-dark" href="index.php" aria-expanded="false">
                        <i class="mdi mdi-gauge"></i>
                        <span class="hide-menu">Dashboard</span>
                    </a>
                </li>
                <li>
                    <a class="waves-effect waves-dark" href="livemonitoring.php" aria-expanded="false">
                        <i class="mdi mdi-camcorder"></i>
                        <span class="hide-menu">Live Monitoring</span>
                    </a>
                </li>
                <li>
                    <a class="waves-effect waves-dark" href="analytics.php" aria-expanded="false">
                        <i class="mdi mdi-access-point"></i>
                        <span class="hide-menu">Analytics</span>
                    </a>
                </li>
                <li>
                    <a class="waves-effect waves-dark" href="profile.php" aria-expanded="false">
                        <i class="mdi mdi-account"></i>
                        <span class="hide-menu">Profile</span>
                    </a>
                </li>
            <?php
                if ($_SESSION['adminuser'] == 'yes') {
                    echo " <li>
                        <a class='waves-effect waves-dark'
                            href='userstable.php' aria-expanded='false'>
                            <i class='mdi mdi-table'></i>
                            <span class='hide-menu'>Users</span>
                        </a>
                    </li>
                    <li>
                        <a class='waves-effect waves-dark'
                            href='newuserreg.php' aria-expanded='false'>
                            <i class='mdi mdi-account-plus'></i>
                            <span class='hide-menu'>Register User</span>
                        </a>
                    </li>
                }
            <?php
        }
    </nav>
</div>
</aside>

```

```

        </li>
    ";
}

    ?>
</ul>
</nav>
<!-- End Sidebar navigation --&gt;

&lt;/div&gt;
<!-- End Sidebar scroll--&gt;
<!-- Bottom points--&gt;
&lt;div class="sidebar-footer"&gt;
    <!-- item--&gt;
    &lt;a href="" class="link" data-toggle="tooltip" title="Refresh"
    onclick="window.location.reload(true);"&gt;
        &lt;i class="fa fa-refresh" aria-hidden="true"&gt;&lt;/i&gt;
    &lt;/a&gt;
    <!-- item--&gt;
    &lt;a href="" class="link" data-toggle="modal" title="Support"
    data-target="#SupportModal" data-whatever="support@smarthomie.com"&gt;
        &lt;i class="fa fa-life-ring" aria-hidden="true"&gt;&lt;/i&gt;
    &lt;/a&gt;
    <!-- item--&gt;
    &lt;a href="../user/logout.php" class="link" data-toggle="tooltip"
    title="Logout"&gt;
        &lt;i class="fa fa-power-off" aria-hidden="true"&gt;&lt;/i&gt;
    &lt;/a&gt;
&lt;/div&gt;
<!-- End Bottom points--&gt;
&lt;/aside&gt;
<!-- ======&gt;
--&gt;     <!-- End Left Sidebar - style you can find in sidebar.scss --&gt;
<!-- ======&gt;
--&gt;     <!-- ======&gt;
--&gt;     &lt;!-- Page wrapper --&gt;
<!-- ======&gt;
--&gt;     &lt;!-- Container fluid --&gt;
&lt;!-- ======&gt;
--&gt;     &lt;!-- Bread crumb and right sidebar toggle --&gt;
&lt;!-- ======&gt;
--&gt;     &lt;!-- Row page titles --&gt;
&lt;div class="row page-titles"&gt;
    &lt;div class="col-md-5 col-8 align-self-center"&gt;
        &lt;h3 class="text-themecolor"&gt;
            &lt;?php echo htmlspecialchars($navloctitle); ?&gt;
        &lt;/h3&gt;
</pre>

```

```

<ol class="breadcrumb">
    <li class="breadcrumb-item">
        <a href="javascript:void(0)">Home</a>
    </li>
    <li class="breadcrumb-item active">
        <?php echo htmlspecialchars($navloctitle); ?>
    </li>
</ol>
</div>
<div class="col-md-7 col-4 align-self-center">
</div>
</div>
<!--
=====
-->
<!-- End Bread crumb and right sidebar toggle -->
<!--
=====
-->

<!-- Logout Modal -->
<div class="modal fade" id="LogoutModal" tabindex="-1"
role="dialog" aria-labelledby="LogoutModalTitle" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered"
role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="LogoutModalTitle">Smart
Homie</h5>
                <button type="button" class="close" data-dismiss="modal"
aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                Click on the button to end your session.
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-
dismiss="modal">Cancel</button>
                <button type="button" class="btn btn-success"
onclick="window.location.href='../../user/logout.php'">Logout</button>
            </div>
        </div>
    </div>
</div>

<!-- DeleteUser Modal -->
<div class="modal fade" id="DeleteUserModal" tabindex="-1"
role="dialog" aria-labelledby="DeleteUserModalTitle" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered"
role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="DeleteUserModalTitle">Smart
Homie</h5>
                <button type="button" class="close" data-dismiss="modal"
aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>

```

```

        </div>
        <div class="modal-body">
            Click on the button to end your session.
        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-secondary" data-
dismiss="modal">Cancel</button>
            <button type="button" class="btn btn-success"
onclick="window.location.href='../../user/logout.php'">Logout</button>
        </div>
    </div>
</div>

<!-- Support Modal -->
<div class="modal fade" id="SupportModal" tabindex="-1"
role="dialog" aria-labelledby="SupportModalTitle" aria-hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="SupportModalTitle">New
message</h5>
                <button type="button" class="close" data-dismiss="modal"
aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                <form>
                    <div class="form-group">
                        <label for="recipient-name" class="col-form-
label">Recipient:</label>
                        <input type="text" class="form-control"
id="recipient-name" readonly>
                    </div>
                    <div class="form-group">
                        <label for="message-text" class="col-form-
label">Message:</label>
                        <textarea class="form-control" id="message-
text"></textarea>
                    </div>
                </form>
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-
dismiss="modal">Close</button>
                <button type="button" class="btn btn-
success">Submit</button>
            </div>
        </div>
    </div>
</div>

```

lowerbody.php

```
</div>
<!-- End Container fluid -->
</div>
<!-- ===== -->

<!-- ===== -->
<!-- ===== -->
<!-- footer -->
<!-- ===== -->
<footer class="footer"> © 2018 Smart Homie </footer>
<!-- ===== -->
<!-- End footer -->
<!-- ===== -->
</div>
<!-- ===== -->
<!-- End Page wrapper -->
<!-- ===== -->
</div>
<!-- ===== -->
<!-- End Wrapper -->
<!-- ===== -->

<!-- ===== -->
<!-- All Jquery -->
<!-- ===== -->
<script src="assets/plugins/jquery/jquery.min.js"></script>
<!-- Bootstrap tether Core JavaScript -->
<script src="assets/plugins/bootstrap/js/tether.min.js"></script>
<script src="assets/plugins/bootstrap/js/bootstrap.min.js"></script>
<!-- slimscrollbar scrollbar JavaScript -->
<script src="js/jquery.slimscroll.js"></script>
<!--Wave Effects -->
<script src="js/waves.js"></script>
<!--Menu sidebar -->
<script src="js/sidebarmenu.js"></script>
<!--sticky kit -->
<script src="assets/plugins/sticky-kit-master/dist/sticky-kit.min.js"></script>
<!--Custom JavaScript -->
<script src="js/custom.min.js"></script>

<!-- Chartsjs.org -->
<script src="js/Chart.JS-2.7.2-bundle.js"></script>
<script
src="charts/indoor_data/temperature/indoordata_temperature.js"></script>
<script
src="charts/indoor_data/humidity/indoordata_humidity.js"></script>
<script
src="charts/outdoor_data/temperature/outdoordata_temperature.js"></script>
<script
src="charts/outdoor_data/humidity/outdoordata_humidity.js"></script>
<script src="js/Chart.JS-2.7.2-utils.js"></script>
```

```

<!-- ===== -->
<!-- This page plugins -->
<!-- ===== -->

<!-- Support Modal Script -->
<script>
  $('#SupportModal').on('show.bs.modal', function (event) {
    var button = $(event.relatedTarget) // Button that triggered the modal
    var recipient = button.data('whatever') // Extract info from data-* attributes
    // If necessary, you could initiate an AJAX request here (and then do the updating in a callback).
    // Update the modal's content. We'll use jQuery here, but you could use a data binding library or other methods instead.
    var modal = $(this)
    modal.find('.modal-title').text('Support Form')
    modal.find('.modal-body input').val(recipient)
  })
</script>

<!-- // Current Time Script -->
<script>
  function startTime() {
    var today = new Date();
    var h = today.getHours();
    var m = today.getMinutes();
    var s = today.getSeconds();
    m = checkTime(m);
    s = checkTime(s);
    document.getElementById('clock').innerHTML =
      'Current Time:' + ' ' + h + ":" + m + ":" + s;
    var t = setTimeout(startTime, 500);
  }

  function checkTime(i) {
    if (i < 10) {
      i = "0" + i
    }; // add zero in front of numbers < 10
    return i;
  }
</script>

<!-- Notification Triggers -->
<script language="JavaScript">
  if (window.location.pathname + window.location.search ==
  '/MyThesis/LiveSite/dashboard/index.php?acon') {

    $("#not-acon").removeClass("disp-not");

    $(".close").click(function () {
      $(this).parent().addClass("disp-not");
    });
  }
  if (window.location.pathname + window.location.search ==
  '/MyThesis/LiveSite/dashboard/index.php?acoff')

```

```

{
    $("#not-acoff").removeClass("disp-not");

    $(".close").click(function () {
        $(this).parent().addClass("disp-not");
    });
}
if (window.location.pathname + window.location.search ==
'/MyThesis/LiveSite/dashboard/index.php?dhon') {

    $("#not-dhon").removeClass("disp-not");

    $(".close").click(function () {
        $(this).parent().addClass("disp-not");
    });
}
if (window.location.pathname + window.location.search ==
'/MyThesis/LiveSite/dashboard/index.php?dhoff') {

    $("#not-dhoff").removeClass("disp-not");

    $(".close").click(function () {
        $(this).parent().addClass("disp-not");
    });
}
</script>

<!-- Delete User Notification Script -->
<script language="JavaScript">
    if (window.location.pathname + window.location.search ==
'/MyThesis/livesite/dashboard/userstable.php?USERDELETED') {
        alert('USER DELETED');
    } else {
        console.log(location.pathname + window.location.search);
    }
</script>

<script type="text/javascript">
    function change_state() {
        alert('hello');
        $.get("status_update.php");
        return false;
    }
</script>

</body>

</html>

```

Δυναμικά αρχεία

dashboard.php (Καρτέλα Dashboard)

```
<?php
// Initialize the session
session_start();

// If session variable is not set it will redirect to login page
if(!isset($_SESSION['username']) || empty($_SESSION['username'])) {
    header("location:../user/login.php");
    exit;
}

?>

<!-- Navigation Location Title -->
<?php
    $navloctitle = 'Dashboard';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Dashboard</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<?php
    $log_id = $_SESSION['id'];
?>

<!-- Start Page Content -->
<!-- ===== -->

    <div id="not-acon" class="notification success disp-not">
        <span class="title">Success</span>Air Condition Status:
    ON<span class="close">X</span>
    </div>
    <div id="not-acoff" class="notification success disp-not">
        <span class="title">Success</span>Air Condition Status:
    OFF<span class="close">X</span>
    </div>
    <div id="not-dhon" class="notification success disp-not">
        <span class="title">Success</span>Dehumidifier Status:
    ON<span class="close">X</span>
    </div>
    <div id="not-dhoff" class="notification success disp-not">
        <span class="title">Success</span>Dehumidifier Status:
    OFF<span class="close">X</span>
    </div>

<div class="row">
```

```

<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-7">
    <div class="card">
        <div class="card-block">
            <div class="row">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <p class="device-name">Air Condition</p>
                </div>
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <?php
                        require_once '../user/config.php';
                        $result = mysqli_query($link, "SELECT status AS AC
FROM `devices_states` WHERE id = 1");
                        $row = mysqli_fetch_array($result);
                        $AC = $row['AC'];
                        if ($AC == 'on')
                        {
                            echo "<img
src='assets/images/icons/device_on_green.png' style='margin-left: 5px;
padding-top:20px' width='100' />";
                        }
                        else
                        {
                            echo "<img src='assets/images/icons/device_off.png'
style='margin-left: 5px' width='100' />";
                        }
                    ?>
                </div>
            </div>
            <div class="row pt-39">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <p class="device-name">Dehumifier</p>
                </div>
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <?php
                        require_once '../user/config.php';
                        $result = mysqli_query($link, "SELECT status AS DH FROM
`devices_states` WHERE id = 2");
                        $row = mysqli_fetch_array($result);
                        $DH = $row['DH'];
                        if ($DH == 'on')
                        {
                            echo "<img
src='assets/images/icons/device_on_green.png' style='margin-left: 5px;
padding-top:20px' width='100' />";
                        }
                        else
                        {
                            echo "<img src='assets/images/icons/device_off.png'
style='margin-left: 5px' width='100' />";
                        }
                    ?>
                </div>
            </div>
        </div>
    </div>
</div>
<!-- Column -->

```

```

<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-6">
    <div class="card">
        <div class="card-block" style="text-align: center">
            <div class="row">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    
                </div>
                <div class="col-lg-6 col-xlg-6 col-md-6 pt-25">
                    <a class='btn' style='background-color: #26c6da; border-
color: #26c6da; color: #fff; '
href='devicesstatehandler.php?id=1&status=on'>Turn
                        On AC</a>
                    <a class='btn' style='background-color: #26c6da; border-
color: #26c6da; color: #fff; '
href='devicesstatehandler.php?id=1&status=off'>Turn
                        Off AC</a>
                </div>
            </div>
            <div class="row pt-39">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    
                </div>
                <div class="col-lg-6 col-xlg-6 col-md-6 pt-25">
                    <a class='btn' style='background-color: #26c6da; border-color:
#26c6da; color: #fff; '
href='devicesstatehandler.php?id=2&status=on'>Turn
                        On DH</a>
                    <a class='btn' style='background-color: #26c6da; border-color:
#26c6da; color: #fff; '
href='devicesstatehandler.php?id=2&status=off'>Turn
                        Off DH</a>
                </div>
            </div>
        </div>
        <!-- Column -->
    </div>
</div>

<div class="row">
    <!-- Column -->
    <div class="col-lg-12 col-xlg-12 col-md-12">
        <div class="card">
            <div class="card-block">
                <p class="dash-user-title">Hello,&nbsp;<span>
                    <?php echo $_SESSION['username'] ?></span></p>
                <p style="text-align: center">Your User Activity</p>
                <div class="row">
                    <table class="table">
                        <thead>
                            <tr>
                                <th class='users-table-center'>Date</th>
                                <th class='users-table-center'>Time</th>
                                <th class='users-table-center'>Action</th>

```

```

        </tr>
    </thead>
    <tbody>
        <?php
            $conn = mysqli_connect("localhost", "root", "", "smart_homie");
            // Check connection
            if ($conn->connect_error) {
                die("Connection failed: " . $conn->connect_error);
            }
            $sql = "SELECT date,time,log FROM user_log WHERE user_id = $log_id";
            $result = $conn->query($sql);
            if ($result->num_rows > 0) {
                // output data of each row
                while($row = $result->fetch_assoc()) {
                    echo "<tr>
                        <td class='users-table-center'>" . $row["date"] .
                    "</td>
                    <td class='users-table-center'>" . $row["time"] .
                "</td>
                    <td class='users-table-center'>" . $row["log"] .
                "</td>
                        </tr>";
                }
                echo "</table>";
            } else { echo "0 results"; }
            $conn->close();
        ?>
    </tbody>
    </table>
</div>
</div>
</div>
</div>
<!-- Column -->
</div>

<?php
    function reload()
    {
        header("Refresh:0");
    }
?>

<!-- ===== -->
<!-- End PAge Content -->
<!-- ===== -->

<!-- Close Container -->
<?php include 'static/lowerbody.php';?>
```

livemonitoring.php (Ζωντανή μετάδοση δεδομένων)

```
<?php
// Initialize the session
session_start();

// If session variable is not set it will redirect to login page
if(!isset($_SESSION['username']) || empty($_SESSION['username'])){
    header("location:../user/login.php");
    exit;
}

?>

<?php
    header("Refresh:120");
?>

<!-- Navigation Location Title -->
<?php
    $navloctitle = 'Live Monitoring';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Dashboard</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<!-- Start Page Content -->
<!-- ======>
-->

<!-- START INDOOR DATA DISPLAY -->
<div class="row">
    <!-- Column -->
    <div class="col-lg-12 col-xlg-12 col-md-12">
        <p class="display-data-title">Display Indoor Data</p>
    </div>
    <!-- Column -->
</div>

<div class="row">
    <!-- Column -->
    <div class="col-lg-6 col-xlg-6 col-md-7">
        <div class="card">
            <div class="card-block">
                <canvas id="indoor_data_temp_live"></canvas>
            </div>
        </div>
    </div>
    <!-- Column -->
```

```

<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-7">
    <div class="card">
        <div class="card-block">
            <canvas id="indoor_data_humidity_live"></canvas>
        </div>
    </div>
</div>
<!-- Column -->
</div>

<!-- END INDOOR DATA DISPLAY -->

<!-- START OUTDOOR DATA DISPLAY -->
<div class="row">
    <!-- Column -->
    <div class="col-lg-12 col-xlg-12 col-md-12">
        <p class="display-data-title">Display Outdoor Data</p>
    </div>
    <!-- Column -->
</div>

<div class="row">
    <!-- Column -->
    <div class="col-lg-6 col-xlg-6 col-md-7">
        <div class="card">
            <div class="card-block">
                <canvas id="outdoor_data_temp_live"></canvas>
            </div>
        </div>
    </div>
    <!-- Column -->
    <!-- Column -->
    <div class="col-lg-6 col-xlg-6 col-md-7">
        <div class="card">
            <div class="card-block">
                <canvas id="outdoor_data_humidity_live"></canvas>
            </div>
        </div>
    </div>
    <!-- Column -->
</div>
<!-- END OUTDOOR DATA DISPLAY -->

-->
<!-- ======>
<!-- End PAge Content -->
<!-- ======>
-->

<!-- Close Container -->
<?php include 'static/lowerbody.php';?>

```

```

analytics_indoor.php (Στατιστικά εσωτερικών χώρων)

<?php
// Initialize the session
session_start();

// If session variable is not set it will redirect to login page
if(!isset($_SESSION['username']) || empty($_SESSION['username'])){
    header("location:../user/login.php");
    exit;
}

?>

<!-- Navigation Location Title -->
<?php
    $navloctitle = 'Analytics Indoor';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Analytics Indoor</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<!-- Start Page Content -->
<!-- ===== -->

<?php include 'analytics_indoor_handler.php';?>

<div class="row">
    <div class="col-lg-4 col-xlg-4 col-md-4">
        </div>
        <!-- Column -->
        <div class="col-lg-4 col-xlg-4 col-md-4">
            <div class="card">
                <div class="card-block">
                    <div class="row">
                        <div class="col-lg-3 col-xlg-3 col-md-3">
                            <a href="analytics_outdoor.php">
                                
                            </a>
                        </div>
                        <div class="col-lg-2 col-xlg-2 col-md-2">
                        </div>
                        <div class="col-lg-7 col-xlg-7 col-md-7">
                            <p class="analytics-explanation-text">Switch to Outdoor</p>
                        </div>
                    </div>
                </div>
            </div>
        </div>
        <!-- Column -->
    
```

```

<div class="col-lg-4 col-xlg-4 col-md-4">
</div>
</div>

<div class="row">
<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-6">
<div class="card">
<div class="card-block">
<p class="analytics-table-title t-center">Today</p>
<div class="row">
<div class="col-lg-6 col-xlg-6 col-md-6">
<p class="analytics-table-data t-center">Temperature:</p>
</div>
<div class="col-lg-6 col-xlg-6 col-md-6">
<p class="analytics-table-data t-center">Humidity:</p>
</div>
</div>
<div class="row">

<div class="col-lg-2 col-xlg-2 col-md-2">

</div>
<div class="col-lg-4 col-xlg-4 col-md-4">
<p class="analytics-table-data">Max: <span class="analytics-table-data analytics-data-max"><?php echo $MAXT; ?> &deg;C</span> </p>
<p class="analytics-table-data">Avg: <span class="analytics-table-data analytics-data-avg"><?php echo $AVGT; ?> &deg;C</span> </p>
<p class="analytics-table-data">Min: <span class="analytics-table-data analytics-data-min"><?php echo $MINT; ?> &deg;C</span> </p>
</div>
<div class="col-lg-2 col-xlg-2 col-md-2">

</div>
<div class="col-lg-4 col-xlg-4 col-md-4">
<p class="analytics-table-data">Max: <span class="analytics-table-data analytics-data-max"><?php echo $MAXH; ?> %</span> </p>
<p class="analytics-table-data">Avg: <span class="analytics-table-data analytics-data-avg"><?php echo $AVGH; ?> %</span> </p>
<p class="analytics-table-data">Min: <span class="analytics-table-data analytics-data-min"><?php echo $MINH; ?> %</span> </p>
</div>
</div>
</div>
<!-- Column -->
<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-6">
<div class="card">

```

```

<div class="card-block">
    <p class="analytics-table-title t-center cur-poin"
    onclick="window.location.href='analytics_indoor_threedays.php';">Last
    Three Days</p>
    <div class="row">
        <div class="col-lg-6 col-xlg-6 col-md-6">
            <p class="analytics-table-data t-center">Temperature:</p>
        </div>
        <div class="col-lg-6 col-xlg-6 col-md-6">
            <p class="analytics-table-data t-center">Humidity:</p>
        </div>
    </div>
    <div class="row">

        <div class="col-lg-2 col-xlg-2 col-md-2">
            
        </div>
        <div class="col-lg-4 col-xlg-4 col-md-4">
            <p class="analytics-table-data">Max: <span class="analytics-
            table-data analytics-data-max"><?php echo $MAXT3; ?>
                &deg;C</span> </p>
            <p class="analytics-table-data">Avg: <span class="analytics-
            table-data analytics-data-avg"><?php echo $AVGT3; ?>
                &deg;C</span> </p>
            <p class="analytics-table-data">Min: <span class="analytics-
            table-data analytics-data-min"><?php echo $MINT3; ?>
                &deg;C</span> </p>
        </div>
        <div class="col-lg-2 col-xlg-2 col-md-2">
            
        </div>
        <div class="col-lg-4 col-xlg-4 col-md-4">
            <p class="analytics-table-data">Max: <span class="analytics-
            table-data analytics-data-max"><?php echo $MAXH3; ?> %</span>
            </p>
            <p class="analytics-table-data">Avg: <span class="analytics-
            table-data analytics-data-avg"><?php echo $AVGH3; ?> %</span>
            </p>
            <p class="analytics-table-data">Min: <span class="analytics-
            table-data analytics-data-min"><?php echo $MINH3; ?> %</span>
            </p>
        </div>
    </div>
    <!-- Column -->
</div>

<div class="row">
    <!-- Column -->
    <div class="col-lg-6 col-xlg-6 col-md-6">
        <div class="card">
            <div class="card-block">
                <p class="analytics-table-title t-center cur-poin"
                onclick="window.location.href='analytics_indoor_sevendays.php';">Last

```

```

Week</p>
<div class="row">
    <div class="col-lg-6 col-xlg-6 col-md-6">
        <p class="analytics-table-data t-center">Temperature:</p>
    </div>
    <div class="col-lg-6 col-xlg-6 col-md-6">
        <p class="analytics-table-data t-center">Humidity:</p>
    </div>
</div>
<div class="row">

    <div class="col-lg-2 col-xlg-2 col-md-2">
        
    </div>
    <div class="col-lg-4 col-xlg-4 col-md-4">
        <p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $MAXT7; ?>
&deg;C</span> </p>
        <p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $AVGT7; ?>
&deg;C</span> </p>
        <p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $MINT7; ?>
&deg;C</span> </p>
    </div>
    <div class="col-lg-2 col-xlg-2 col-md-2">
        
    </div>
    <div class="col-lg-4 col-xlg-4 col-md-4">
        <p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $MAXH7; ?> %</span>
    </p>
        <p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $AVGH7; ?> %</span>
    </p>
        <p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $MINH7; ?> %</span>
    </p>
    </div>
    </div>
</div>
<!-- Column -->
<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-6">
    <div class="card">
        <div class="card-block">
            <p class="analytics-table-title t-center cur-poin"
onclick="window.location.href='analytics_indoor_month.php';">Last
Month</p>
            <div class="row">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <p class="analytics-table-data t-center">Temperature:</p>
                </div>
                <div class="col-lg-6 col-xlg-6 col-md-6">

```

```

        <p class="analytics-table-data t-center">Humidity:</p>
    </div>
</div>
<div class="row">

    <div class="col-lg-2 col-xlg-2 col-md-2">
        
    </div>
    <div class="col-lg-4 col-xlg-4 col-md-4">
        <p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $MAXT30; ?>
        &deg;C</span> </p>
        <p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $AVGT30; ?>
        &deg;C</span> </p>
        <p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $MINT30; ?>
        &deg;C</span> </p>
    </div>
    <div class="col-lg-2 col-xlg-2 col-md-2">
        
    </div>
    <div class="col-lg-4 col-xlg-4 col-md-4">
        <p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $MAXH30; ?> %</span>
        </p>
        <p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $AVGH30; ?> %</span>
        </p>
        <p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $MINH30; ?> %</span>
        </p>
    </div>
    </div>
</div>
<!-- Column --&gt;
&lt;/div&gt;

&lt;!-- ===== --&gt;
&lt;!-- End PAge Content --&gt;
&lt;!-- ===== --&gt;

&lt;!-- Close Container --&gt;
&lt;?php include 'static/lowerbody.php';?&gt;
</pre>

```

analytics_outdoor.php (Στατιστικά εξωτερικών χώρων)

```
<?php
// Initialize the session
session_start();

// If session variable is not set it will redirect to login page
if(!isset($_SESSION['username']) || empty($_SESSION['username'])) {
    header("location:../user/login.php");
    exit;
}

?>

<!-- Navigation Location Title -->
<?php
    $navloctitle = 'Analytics Outdoor';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Analytics Outdoor</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<!-- Start Page Content -->
<!-- ===== -->

<?php include 'analytics_outdoor_handler.php';?>

<div class="row">
    <div class="col-lg-4 col-xlg-4 col-md-4">
        </div>
        <!-- Column -->
        <div class="col-lg-4 col-xlg-4 col-md-4">
            <div class="card">
                <div class="card-block">
                    <div class="row">
                        <div class="col-lg-3 col-xlg-3 col-md-3">
                            <a href="analytics_indoor.php">
                                
                            </a>
                        </div>
                        <div class="col-lg-2 col-xlg-2 col-md-2">
                        </div>
                        <div class="col-lg-7 col-xlg-7 col-md-7">
                            <p class="analytics-explanation-text">Switch to Indoor</p>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

</div>
<!-- Column -->
<div class="col-lg-4 col-xlg-4 col-md-4">
</div>
</div>

<div class="row">
<!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-6">
<div class="card">
<div class="card-block">
<p class="analytics-table-title t-center">Today</p>
<div class="row">
<div class="col-lg-6 col-xlg-6 col-md-6">
<p class="analytics-table-data t-center">Temperature:</p>
</div>
<div class="col-lg-6 col-xlg-6 col-md-6">
<p class="analytics-table-data t-center">Humidity:</p>
</div>
</div>
<div class="row">

<div class="col-lg-2 col-xlg-2 col-md-2">

</div>
<div class="col-lg-4 col-xlg-4 col-md-4">
<p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $OMAXT; ?>
&deg;C</span> </p>
<p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $OAVGT; ?>
&deg;C</span> </p>
<p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $OMINT; ?>
&deg;C</span> </p>
</div>
<div class="col-lg-2 col-xlg-2 col-md-2">

</div>
<div class="col-lg-4 col-xlg-4 col-md-4">
<p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $OMAXH; ?> %</span>
</p>
<p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $OAVGH; ?> %</span>
</p>
<p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $OMINH; ?> %</span>
</p>
</div>
</div>
</div>
<!-- Column -->
<!-- Column -->

```

```

<div class="col-lg-6 col-xlg-6 col-md-6">
    <div class="card">
        <div class="card-block">
            <p class="analytics-table-title t-center cur-poin"
                onclick="window.location.href='analytics_outdoor_threedays.php';">Last
                Three Days</p>
            <div class="row">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <p class="analytics-table-data t-center">Temperature:</p>
                </div>
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <p class="analytics-table-data t-center">Humidity:</p>
                </div>
            </div>
            <div class="row">
                <div class="col-lg-2 col-xlg-2 col-md-2">
                    
                </div>
                <div class="col-lg-4 col-xlg-4 col-md-4">
                    <p class="analytics-table-data">Max: <span class="analytics-
                    table-data analytics-data-max"><?php echo $OMAXT3; ?>
                        &deg;C</span> </p>
                    <p class="analytics-table-data">Avg: <span class="analytics-
                    table-data analytics-data-avg"><?php echo $OAVGT3; ?>
                        &deg;C</span> </p>
                    <p class="analytics-table-data">Min: <span class="analytics-
                    table-data analytics-data-min"><?php echo $OMINT3; ?>
                        &deg;C</span> </p>
                </div>
                <div class="col-lg-2 col-xlg-2 col-md-2">
                    
                </div>
                <div class="col-lg-4 col-xlg-4 col-md-4">
                    <p class="analytics-table-data">Max: <span class="analytics-
                    table-data analytics-data-max"><?php echo $OMAXH3; ?> %</span>
                    </p>
                    <p class="analytics-table-data">Avg: <span class="analytics-
                    table-data analytics-data-avg"><?php echo $OAVGH3; ?> %</span>
                    </p>
                    <p class="analytics-table-data">Min: <span class="analytics-
                    table-data analytics-data-min"><?php echo $OMINH3; ?> %</span>
                    </p>
                </div>
            </div>
        </div>
        <!-- Column -->
    </div>

    <div class="row">
        <!-- Column -->
        <div class="col-lg-6 col-xlg-6 col-md-6">
            <div class="card">
                <div class="card-block">

```

```

        <p class="analytics-table-title t-center cur-poin"
        onclick="window.location.href='analytics_outdoor_sevendays.php';">Last
Week</p>
        <div class="row">
            <div class="col-lg-6 col-xlg-6 col-md-6">
                <p class="analytics-table-data t-center">Temperature:</p>
            </div>
            <div class="col-lg-6 col-xlg-6 col-md-6">
                <p class="analytics-table-data t-center">Humidity:</p>
            </div>
        </div>
        <div class="row">

            <div class="col-lg-2 col-xlg-2 col-md-2">
                
            </div>
            <div class="col-lg-4 col-xlg-4 col-md-4">
                <p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $OMAXT7; ?>
                    &deg;C</span> </p>
                <p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $OAVGT7; ?>
                    &deg;C</span> </p>
                <p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $OMINT7; ?>
                    &deg;C</span> </p>
            </div>
            <div class="col-lg-2 col-xlg-2 col-md-2">
                
            </div>
            <div class="col-lg-4 col-xlg-4 col-md-4">
                <p class="analytics-table-data">Max: <span class="analytics-
table-data analytics-data-max"><?php echo $OMAXH7; ?> %</span>
                </p>
                <p class="analytics-table-data">Avg: <span class="analytics-
table-data analytics-data-avg"><?php echo $OAVGH7; ?> %</span>
                </p>
                <p class="analytics-table-data">Min: <span class="analytics-
table-data analytics-data-min"><?php echo $OMINH7; ?> %</span>
                </p>
            </div>
        </div>
    </div>
    <!-- Column -->
    <!-- Column -->
<div class="col-lg-6 col-xlg-6 col-md-6">
    <div class="card">
        <div class="card-block">
            <p class="analytics-table-title t-center cur-poin"
            onclick="window.location.href='analytics_outdoor_month.php';">Last
Month</p>
            <div class="row">
                <div class="col-lg-6 col-xlg-6 col-md-6">
                    <p class="analytics-table-data t-center">Temperature:</p>

```

```

        </div>
        <div class="col-lg-6 col-xlg-6 col-md-6">
            <p class="analytics-table-data t-center">Humidity:</p>
        </div>
    </div>
    <div class="row">

        <div class="col-lg-2 col-xlg-2 col-md-2">
            
        </div>
        <div class="col-lg-4 col-xlg-4 col-md-4">
            <p class="analytics-table-data">Max: <span class="analytics-table-data analytics-data-max"><?php echo $OMAXT30; ?> &deg;C</span> </p>
            <p class="analytics-table-data">Avg: <span class="analytics-table-data analytics-data-avg"><?php echo $OAVGT30; ?> &deg;C</span> </p>
            <p class="analytics-table-data">Min: <span class="analytics-table-data analytics-data-min"><?php echo $OMINT30; ?> &deg;C</span> </p>
        </div>
        <div class="col-lg-2 col-xlg-2 col-md-2">
            
        </div>
        <div class="col-lg-4 col-xlg-4 col-md-4">
            <p class="analytics-table-data">Max: <span class="analytics-table-data analytics-data-max"><?php echo $OMAXH30; ?> %</span> </p>
            <p class="analytics-table-data">Avg: <span class="analytics-table-data analytics-data-avg"><?php echo $OAVGH30; ?> %</span> </p>
            <p class="analytics-table-data">Min: <span class="analytics-table-data analytics-data-min"><?php echo $OMINH30; ?> %</span> </p>
        </div>
    </div>
    <!-- Column -->
</div>

<!-- ===== -->
<!-- End PAge Content -->
<!-- ===== -->

<!-- Close Container -->
<?php include 'static/lowerbody.php';?>
```

profile.php (Προφίλ χρήστη)

```
<?php
    // Initialize the session
    session_start();

    // If session variable is not set it will redirect to login page
    if(!isset($_SESSION['username']) || empty($_SESSION['username'])) {
        header("location:../user/login.php");
        exit;
    }

?>

<!-- Navigation Location Title -->
<?php
    $navloctitle = 'Profile';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Profile</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<!-- Start Page Content -->
<!-- ======>
-->

<!-- Row -->
<div class="row">
    <!-- Column -->
    <div class="col-lg-6 col-xlg-6 col-md-5">
        <div class="card">
            <div class="card-block">
                <center class="m-t-30">
                    
                    <h4 class="card-title m-t-10" style="font-weight: 600">
                        <?php echo htmlspecialchars($_SESSION['username']); ?>
                    </h4>
                    <h6 class="card-subtitle">
                        <?php echo htmlspecialchars($_SESSION['email']); ?>
                    </h6>
                    <div class="row text-center justify-content-md-center">
                        <div class="col-12">
                            <a href="#" class="link">
                                <i class="icon-people" style="padding-right: 5px"></i> Users Registered:
                            <?php
                                require_once '../user/config.php';
                                $result = mysqli_query($link, "SELECT COUNT(*)
```

```

AS `count` FROM `users`);
                $row = mysqli_fetch_array($result);
                $count = $row['count'];
                echo " <b> $count </b> ";
            ?>
        </a>
    </div>
    <div class="col-12" style="padding-top: 5px; padding-
bottom: 27px;">
        <a href="#" class="link">
            <i class="icon-energy" style="padding-right:
5px"></i> Access Level:
        <?php

            if ($_SESSION['adminuser'] == 'yes'){
                echo " <b>Administrator</b> ";
            }else {
                echo " <b>User</b>";
            }
        ?>
        </a>
    </div>
    </div>
    </center>
    </div>
    </div>
</div>
<!-- Column --&gt;

<!-- Column --&gt;
&lt;div class="col-lg-6 col-xlg-6 col-md-7"&gt;
    &lt;div class="card"&gt;
        &lt;div class="card-block"&gt;
            &lt;p class="reg-newuser-title"&gt;Update Your Credentials&lt;/p&gt;
            &lt;form class="form-horizontal form-material"
action="updateprof.php" method="post"&gt;
                &lt;div class="form-group"&gt;
                    &lt;label class="col-md-12"&gt;Full Name&lt;/label&gt;
                    &lt;div class="col-md-12"&gt;
                        &lt;input type="text" placeholder="&lt;?php echo
htmlspecialchars($_SESSION['username']); ?&gt;" class="form-control form-
control-line"
                            name="username-update"&gt;
                    &lt;/div&gt;
                &lt;/div&gt;
                &lt;div class="form-group"&gt;
                    &lt;label class="col-md-12"&gt;Email&lt;/label&gt;
                    &lt;div class="col-md-12"&gt;
                        &lt;input type="email" placeholder="&lt;?php echo
htmlspecialchars($_SESSION['email']); ?&gt;" class="form-control form-
control-line"
                            name="email-update"&gt;
                    &lt;/div&gt;
                &lt;/div&gt;
                &lt;div class="form-group"&gt;
                    &lt;label class="col-md-12"&gt;Password&lt;/label&gt;
                    &lt;div class="col-md-12"&gt;
</pre>

```

```

        <input type="text" placeholder="" class="form-control
form-control-line"
            name="password-update">
        </div>
    </div>
    <div>
        <input style="display:none" type="text"
placeholder=<?php echo htmlspecialchars($_SESSION['username']); ?>" class="form-control form-control-line"
            name="old-username" value=<?php echo
htmlspecialchars($_SESSION['username']); ?>">
        <input style="display:none" type="text"
placeholder=<?php echo htmlspecialchars($_SESSION['email']); ?>" class="form-control form-control-line"
            name="old-email" value=<?php echo
htmlspecialchars($_SESSION['email']); ?>">
        <input style="display:none" type="text"
placeholder=<?php echo htmlspecialchars($_SESSION['password']); ?>" class="form-control form-control-line"
            name="old-password" value=<?php echo
htmlspecialchars($_SESSION['password']); ?>">
        </div>
        <div class="form-group">
            <div class="col-sm-12">
                <button class="btn btn-success" name="update-
btn">Update Profile</button>
            </div>
        </div>
        </form>
    </div>
    </div>
    <!-- Column -->
</div>
<!-- Row -->
<!-- =====-->
-->
<!-- End PAge Content -->
<!-- =====-->
-->

<!-- Close Container -->
<?php include 'static/lowerbody.php';?>
```

userstable.php (Πίνακας εγγεγραμμένων χρηστών)

```
<?php
// Initialize the session
session_start();

// If session variable is not set it will redirect to login page
if(!isset($_SESSION['username']) || empty($_SESSION['username'])){
    header("location:../user/login.php");
    exit;
}

?>

<!-- Navigation Location Title -->
<?php
    $navloctitle = 'Registered Users';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Dashboard</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<!-- ======>
-->
<!-- Start Page Content -->
<!-- ======>
-->

<div class="row">
    <div class="col-12">
        <div class="card">
            <div class="card-block">
                <h4 class="users-table-title">Registered Users Table</h4>
                <div class="table-responsive">
                    <table class="table">
                        <thead>
                            <tr>
                                <th class='users-table-center'>ID</th>
                                <th class='users-table-center'>UserName</th>
                                <th class='users-table-center'>Email</th>
                                <th class='users-table-center'>Admin Rights</th>
                                <th class='users-table-center'>Action</th>
                            </tr>
                        </thead>
                        <tbody>
                            <?php
                                $conn = mysqli_connect("localhost", "root", "", 
"smart_homie");
                                // Check connection
                                if ($conn->connect_error) {
                                    die("Connection failed: " . $conn->connect_error);
                            }
                        </tbody>
                    </table>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

        }
        $sql = "SELECT id, username,email,adminuser FROM
users";
        $result = $conn->query($sql);
        if ($result->num_rows > 0) {
        // output data of each row
        while($row = $result->fetch_assoc()) {
            echo "<tr>
                <td class='users-table-center'>" . $row["id"] .
"("</td>
                <td class='users-table-center'>" . $row["username"] . "</td>
                    <td class='users-table-center'>" . $row["email"] . "</td>
                <td class='users-table-center'>" . $row["adminuser"] . "</td>
                    <td class='users-table-center'>
                        <a class='btn btn-danger'
                            href='deleteuserhandler.php?id=".$row["id"]."'>Delete User</a>
                        <a class='btn btn-primary' style='margin-left:
20px'
                            href='changeadminrightshandler.php?id=".$row["id"]."&adminuser=".$row["a
dminuser"]."'>Change Admin Rigths</a>
                    </td>
                </tr>";
            }
        } else { echo "0 results"; }
        $conn->close();
    ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>

<!-- ======>
-->
<!-- End PAge Content -->
<!-- ======>
-->

<!-- Close Container -->
<?php include 'static/lowerbody.php';?>
```

newuserreg.php (εγγραφή νέου χρήστη)

```
<?php
// Initialize the session
session_start();

// If session variable is not set it will redirect to login page
if(!isset($_SESSION['username']) || empty($_SESSION['username'])) {
    header("location:../user/login.php");
    exit;
}

?>

<!-- Navigation Location Title -->
<?php
$navloctitle = 'Register New User';
?>

<!-- Include head tag -->
<?php include 'static/head.html';?>

<title>Smart Homie - Register New User</title>
</head>

<!-- Start Container -->
<?php include 'static/upperbody.php';?>

<!-- Start Page Content -->
<!-- ======>
-->

<div class="row">
    <div class="col-lg-3 col-xlg-3 col-md-3">
    </div>
    <div class="col-lg-6 col-xlg-6 col-md-6">
        <div class="card">
            <div class="card-block">
                <p class="reg-newuser-title">Register New User</p>
                <form class="form-horizontal form-material"
action="newuserreghandler.php" method="post">
                    <div class="form-group">
                        <label class="col-md-12">Username</label>
                        <div class="col-md-12">
                            <input name="username" type="text" placeholder="">
                        </div>
                    </div>
                    <div class="form-group">
                        <label for="example-email" class="col-md-12">Email</label>
                        <div class="col-md-12">
                            <input name="email" type="email" placeholder="">
                        </div>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>
```

```

<div class="form-group">
    <label class="col-md-12" style="padding-bottom: 10px">Give Administrator Rights?</label>
    <div class="col-md-12">
        <select class="form-control" name="adminrights">
            <option value="no">No</option>
            <option value="yes">Yes</option>
        </select>
    </div>
</div>
<div class="form-group">
    <label class="col-md-12">Password</label>
    <div class="col-md-12">
        <input name="password" type="password" value="" class="form-control form-control-line">
    </div>
</div>
<div class="form-group">
    <div class="col-sm-6 col-md-6">
        <button class="btn btn-success reg-newuser-btn">Submit</button>
    </div>
</div>
</form>
</div>
</div>
</div>

<!-- ======>
--> <!-- End PAge Content -->
<!-- ======>
-->

<!-- Close Container -->
<?php include 'static/lowerbody.php';?>

```