

**Δημοκρίτειο Πανεπιστήμιο Θράκης Πολυτεχνική Σχολή**

**Τμήμα Μηχανικών Παραγωγής και Διοίκησης**

Τομέας Συστημάτων Παραγωγής

Επιβλέπων: Αναπληρωτής Καθηγητής Κατσαβούνης Στέφανος

**Διαδικτυακή εφαρμογή παρακολούθησης & διαχείρισης έξυπνων συσκευών με σκοπό την μετατροπή ενός συμβατικού σπιτιού σε έξυπνο**

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 με σκοπό την μετατροπή ενός συμβατικού σπιτιού σε έξυπνο**

Διπλωματική Εργασία

**Μησσήν Στέφανος**

Αρ. Μητρώου: 91432

Διπλωματική εργασία η οποία υποβλήθηκε τον Οκτώβριο 2018

για την απόκτηση του διπλώματος του Μηχανικού Παραγωγής και Διοίκησης

*Αφιερωμένο στους γονείς μου,*

***Κίμωνα*** *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 την 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 την δημιουργία ενός συστήματος παρακολούθησης & διαχείρισης έξυπνων συσκευών με σκοπό την 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 την 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 την 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 την 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 την διεπαφή του χρήστη με τις έξυπνες συσκευές αλλά και την διαχείριση 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 βελτίωσης της.

Αναλυτικότερα στο 1ο κεφάλαιο παρουσιάζεται το πρόβλημα, η σημασία του για την ανθρωπότητα, 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.

Έπειτα 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 3ο κεφάλαιο.

Στο κεφάλαιο 4 επεξηγείται λεπτομερώς η σχεδίαση και η υλοποίηση της διαδικτυακής εφαρμογής. Πιο συγκεκριμένα, αναλύονται οι απαιτήσεις του συστήματος, γίνεται λόγος για την ασφάλεια 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, όπου χωρίς αυτούς θα ήταν πολύ δυσκολότερος και δύσβατος ο δρόμος που έχω διανύσει για να φτάσω ως εδώ. Συνέβαλλαν όλοι με τον τρόπο τους στην διαμόρφωση του χαρακτήρα και της νοοτροπίας μου. Με βοήθησαν να πετύχω πολλά από τα όνειρα 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 διαθεσιμότητα του σε ότι χρειαζόμουν, με βοήθησε να ξεπεράσω τις δυσκολίες που αντιμετώπισα στην 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 κομμάτι των γνώσεων & και των εμπειριών του όσο αφορά επιστημονικούς κλάδους, που ανταποκρίνονται στα ενδιαφέροντα 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 αυτά 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 στην μνήμη μου ο καθένας ξεχωριστά την δική του εικόνα & προσωπικότητα.

***Μησσήν Στέφανος,***

***Οκτώβριος 2018***

# Πίνακας περιεχομένων

[Πρόλογος i](#_Toc526802622)

[Ευχαριστίες iii](#_Toc526802623)

[Πίνακας περιεχομένων v](#_Toc526802624)

[Περιεχόμενα Πινάκων και Σχημάτων v](#_Toc526802625)

[Περίληψη vi](#_Toc526802626)

[Abstract vii](#_Toc526802627)

[**1.1** **ΤΟ ΠΡΟΒΛΗΜΑ ΚΑΙ Η ΣΗΜΑΣΙΑ ΤΟΥ** 1](#_Toc526802628)

[**1.2** **INTERNET OF THINGS** 3](#_Toc526802629)

[**1.3** **ΣΤΟΧΟΙ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ** 6](#_Toc526802630)

[**1.4** **ΔΟΜΗ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ** 7](#_Toc526802631)

[**2.1** **ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΔΙΑΔΙΚΤΥΑΚΗΣ ΕΦΑΡΜΟΓΗΣ** 9](#_Toc526802633)

[**2.1.1.** **ΓΛΩΣΣΑ ΣΗΜΑΝΣΗΣ ΥΠΕΡΚΕΙΜΕΝΟΥ - HTML** 10](#_Toc526802636)

[**2.1.2.** **ΓΛΩΣΣΑ ΜΟΡΦΟΠΟΙΗΣΗΣ – CASCADING STYLES SHEETS (CSS)** 16](#_Toc526802637)

[**2.1.3.** **ΓΛΩΣΣΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΧΡΗΣΤΗΣ – ΔΙΑΚΟΜΙΣΤΗ PHP** 19](#_Toc526802638)

[**2.1.4.** **ΓΛΩΣΣΑ SCRIPTING - JAVASCRIPT** 23](#_Toc526802639)

[**2.1.5.** **ΓΛΩΣΣΑ ΣΧΕΣΙΑΚΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ MySQL** 25](#_Toc526802640)

[**2.2** **ΥΛΙΚΟΣ ΕΞΟΠΛΙΣΜΟΣ - HARDWARE** 27](#_Toc526802641)

[**2.2.1.** **ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO UNO REV3** 28](#_Toc526802646)

[**2.2.2.** **ΜΙΚΡΟΕΛΕΓΚΤΗΣ WEMOS D1 R2** 33](#_Toc526802647)

# 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

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 Tρόπος 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

# Περίληψη

Κείμενο περίληψης

# Abstract

Κείμενο abstract

Κεφάλαιο 1

**Εισαγωγή**

* 1. **ΤΟ ΠΡΟΒΛΗΜΑ ΚΑΙ Η ΣΗΜΑΣΙΑ 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/εξυπηρετητή (client/server) που χρησιμοποιεί ένα φυλλομετρητή (browser). Οι εφαρμογές αυτές συνήθως εκτελούνται σε ισχυρές υπολογιστικές μηχανές οι οποίες έχουν τον ρόλο του σταθμού εξυπηρέτησης 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, παραδείγματος χάρη 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 χρήσης (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 που περνάνε μεγάλο μέρος της ημέρας όπως είναι τα σπίτια τους. Τα σημαντικότερα ζητήματα ενέργειας που έρχονται να καλύψουν οι έξυπνες συσκευές είναι θέματα που αφορούν την εξοικονόμηση ρεύματος από ηλεκτρικές συσκευές του σπιτιού, που χρησιμοποιούνται για πολλές ώρες ανά την ημέρα όπως τα κλιματιστικά και τα καλοριφέρ, ακόμη για την θέρμανση του νερού όπου χρησιμοποιείται στο μπάνιο, στο πλύσιμο των πιάτων καθώς και στο 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 μίας διαδικτυακής εφαρμογής η οποία θα είναι εύχρηστη, θα μπορεί να παρέχει τις απαραίτητες πληροφορίες που ζητάει ο χρήστης, στο σωστό χρόνο και χωρίς να υπάρχουν προβλήματα στην παραμετροποίηση και τροποποίηση αυτών των δεδομένων για την καλύτερη εμπειρία χρήσης.

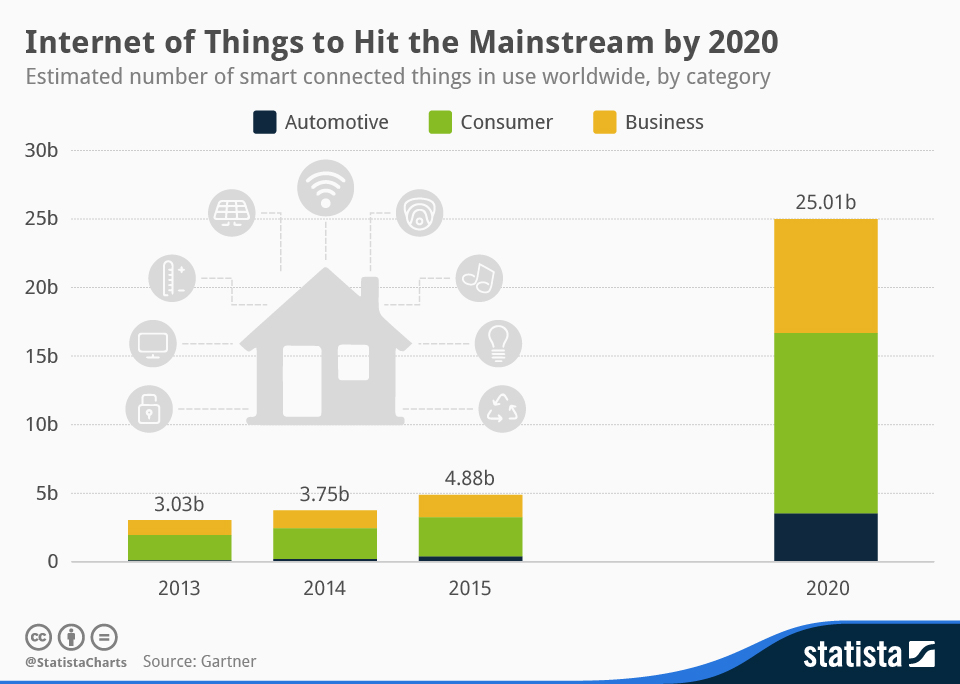
Σκοπός της παρούσας διπλωματικής εργασίας είναι η δημιουργία μιας διαδικτυακής εφαρμογής για την ανάπτυξη ενός συστήματος παρακολούθησης & διαχείρισης έξυπνων συσκευών με σκοπό την μετατροπή ενός συμβατικού σπιτιού σε έξυπνο. Λαμβάνονται ως δεδομένα, πρωτογενείς μετρήσεις θερμοκρασίας και υγρασίας εντός και εκτός του σπιτιού που χρησιμοποιήθηκε για την 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. **INTERNET OF THINGS**

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][3].

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 [4].

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 [5].



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

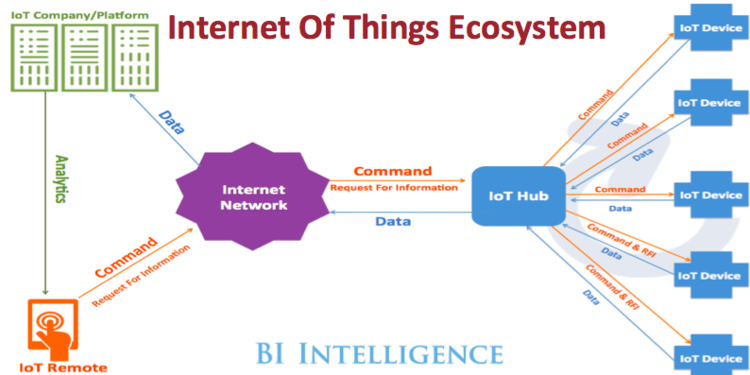
Περαιτέρω, το IoT βοηθάει στην σύνδεση αντικειμένων ή διαδικασιών οι οποίες επιτρέπουν την καλύτερη παρακολούθηση και αλληλεπίδραση. Υπάρχουν τέσσερα στρώματα της IoT αρχιτεκτονικής:

* Το στρώμα ανίχνευσης ή αντίληψης, το οποίο συλλέγει και μεταδίδει δεδομένα με την 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 την 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, επεξεργάζονται μέσω συγκεκριμένων τεχνικών που χρησιμοποιούνται για ειδικά σχέδια. Επιπλέον, αφορά τη διαχείριση αντικειμένων και πραγμάτων [6].

*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 του Διαδικτύου των Πραγμάτων (IoT Ecosystem)

* 1. **ΣΤΟΧΟΙ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ**

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 την 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 την 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 την δημιουργία της διαδικτυακής εφαρμογής. Σε αυτή 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 στην χρήση του σπιτιού.

* 1. **ΔΟΜΗ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ**

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 την 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. Παραθέτονται 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 και γίνεται λόγος για την τοπική λειτουργία της εφαρμογής & 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 την διαδικτυακή εφαρμογή. Επιπλέον, γίνεται μία λεπτομερή 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 προσπάθεια της να δημιουργήσει μία διαδικτυακή εφαρμογή η οποία θα προσφέρει την καλύτερη δυνατή εμπειρία χρήσης για την διαχείριση έξυπνων συσκευών, είναι απαραίτητος ο διαχωρισμός της σε δύο μέρη καθώς και η εναρμόνιση αυτών.

Αρχικά, το υλικό μέρος (Hardware), το οποίο περιλαμβάνει τα αισθητήρια όργανα καθώς και την συνδεσμολογία αυτών για την λήψη των απαραίτητων δεδομένων. Το δεύτερο μέρος, αποτελείται από την κωδικοποίηση/παραμετροποίηση των αισθητηρίων καθώς και την ανάπτυξη του λογισμικού (Software) με την βοήθεια του οποίου συμπληρώνεται η διαδικτυακή εφαρμογή. Για την ευκολότερη κατανόηση του 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 την διασφάλιση της ποιότητας της εφαρμογής.

1. 1. **ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ ΔΙΑΔΙΚΤΥΑΚΗΣ ΕΦΑΡΜΟΓΗΣ**

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 [7].

2. * 1. **ΓΛΩΣΣΑ ΣΗΜΑΝΣΗΣ ΥΠΕΡΚΕΙΜΕΝΟΥ - 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.



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 μία ένα έγγραφο HTML, το οποίο μετά παραδίδεται στον χρήστη και εμφανίζεται σε μία συσκευή [8].

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



Σχήμα 2.2 Βασικό κομμάτι κώδικα για την εκκίνηση ενός εγγράφου HTML

Στην προσπάθεια που γίνεται για την 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 κομμάτι 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 την σύνδεση μίας σελίδας προς μία άλλη. Τo πιο σημαντικό στοιχείο αυτής της ετικέτας είναι το *href* ,το οποίο υποδεικνύει τον προορισμό του συνδέσμου |
| <article> | Η ετικέτα <article> καθορίζει ένα ανεξάρτητο, αυτό-συντηρούμενο περιεχόμενο. |
| <aside> | Η ετικέτα <aside> καθορίζει περιεχόμενο , πέρα από το ήδη υπάρχων περιεχόμενο των άλλων ετικετών. Το περιεχόμενο αυτό πρέπει να συσχετίζεται με τα υπόλοιπα περιεχόμενα. |
| <blockquote> | Η ετικέτα <blockquote> 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 κομματιών κώδικα. |
| <body> | Η ετικέτα <body> ορίζει το σώμα του εγγράφου. Όλα τα περιεχόμενα ενός εγγράφου HTML, όπως κείμενο, υπερσύνδεσμοι, εικόνες, πίνακες, λίστες, κτλ. Βρίσκονται μέσα σε αυτό. |
| <button> | Η ετικέτα <button> 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 <button> και με την ετικέτα <input>. |
| <canvas> | Η ετικέτα <canvas> χρησιμοποιείται για την δημιουργία δυναμικών γραφικών (on the fly), μέσω του scripting (συνήθως με την γλώσσα JavaScript) |
| <div> | Η ετικέτα <div> καθορίζει ένα τμήμα σε ένα έγγραφο HTML. Χρησιμοποιείται για την ομαδοποίηση των στοιχείων, ώστε να δοθούν κοινές μορφοποιήσεις με τη χρήση του Id. |
| <footer> | Η ετικέτα <footer> καθορίζει το υποσέλιδο για ένα αντικείμενο ή μία περιοχή. Η ετικέτα, θα πρέπει να περιέχει να περιέχει πληροφορίες για τα στοιχεία που περιβάλει. |
| <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 περιέχει, είτε περιέχει κομμάτια κώδικα, είτε συνδέει το παρών έγγραφο με ένα εξωτερικό αρχείο script μέσω του στοιχείου *src.* Η βασική γλώσσα scripting είναι ηJavaScript. |
| <table> | Η ετικέτα <table> ορίζει έναν πίνακα HTML. Ο πίνακας αποτελείται από ένα ή περισσότερα <tr>, <td> ή <th>. |

* + 1. **ΓΛΩΣΣΑ ΜΟΡΦΟΠΟΙΗΣΗΣ – 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 [9]. 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.

H 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.

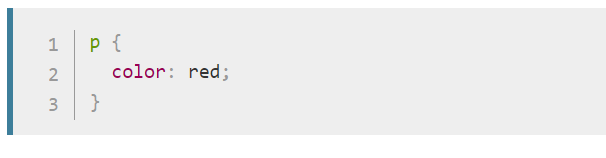
Σύνταξη της CSS

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, για αυτό τον 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.

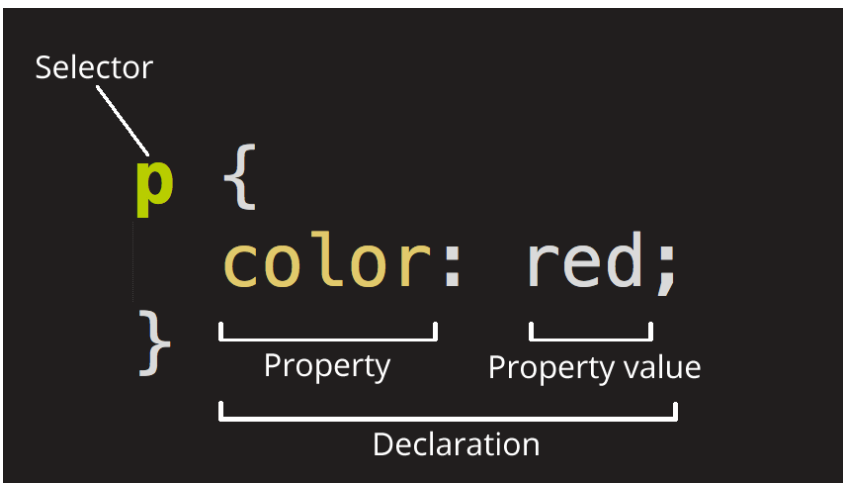


Σχήμα 2.3 Κώδικας σύνδεσης ενός αρχείου CSS σε ένα έγγραφο HTML

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 κομμάτι κώδικα CSS:



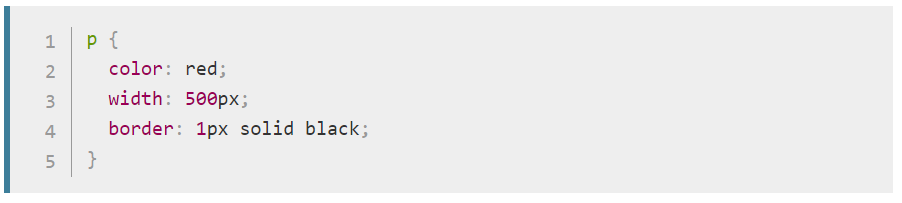
Σχήμα 2.4 Κομμάτι κώδικα CSS για αλλαγή του χρώματος των παραγράφων σε κόκκινο

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.

Σχήμα 2.5 Ανατομία ενός κώδικα CSS

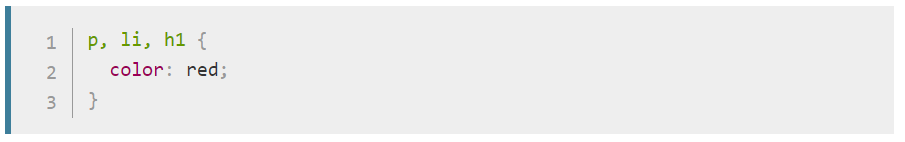
Ο κανόνας χωρίζεται σε κομμάτια για την καλύτερη ανάγνωση του από τους ανθρώπους αλλά και για την εύρυθμη λειτουργία μίας ιστοσελίδας αλλά και μίας διαδικτυακής εφαρμογής. Ο διαχωρισμός γίνεται ως εξής:

* **Selector:** Ο selector είναι το όνομα του στοιχείου HTML και βρίσκεται στην αρχή του σετ κανόνων. Είναι υπεύθυνος για την επιλογή των στοιχείων των οποίων 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.

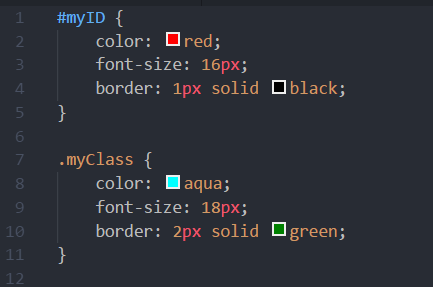
Ακόμη, για να τροποποιήσουμε πολλαπλές τιμές με μία προσπάθεια, απλά χρειάζεται να τις γράψουμε, την μία κάτω από την άλλη και χωρισμένες με ερωτηματικό, όπως φαίνεται παρακάτω.

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 [10] .



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’ αυτή την επιλογή στην CSS χρησιμοποιείται το σύμβολο “.”. Στο παρακάτω σχήμα παρουσιάζεται ο τρόπος επιλογής ενός στοιχείου προς μορφοποίηση.

Σχήμα 2.8 Tρόπος επιλογής ενός στοιχείου προς μορφοποίηση - CSS

* + 1. **ΓΛΩΣΣΑ ΕΠΙΚΟΙΝΩΝΙΑΣ ΧΡΗΣΤΗΣ – ΔΙΑΚΟΜΙΣΤΗ 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 [11]. 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 την χρησιμοποιήσουμε. Μπορεί να χρησιμοποιηθεί για την 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 (κομμάτια 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 την σύνδεση τους με τον εξυπηρετητή, για την κρυπτογράφηση και αποστολή δεδομένων σε εξυπηρετητές καθώς και για προγραμματισμό αυτόματων διαδικασιών στον εξυπηρετητή.

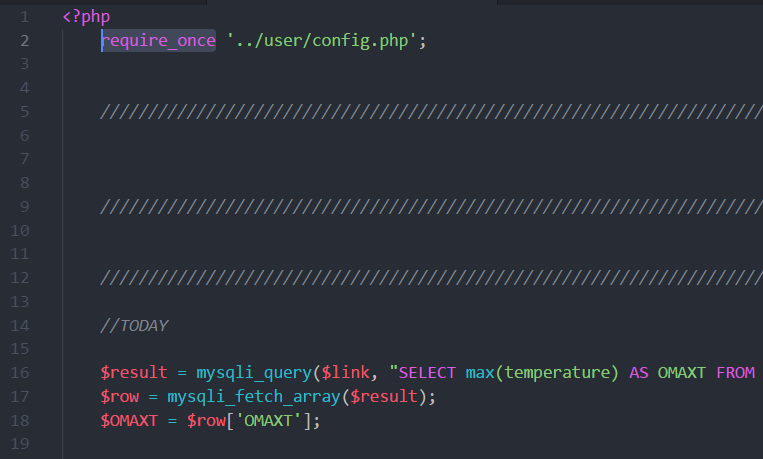
Την παρούσα στιγμή, κατά την οποία συγγράφεται αυτή η διπλωματική εργασία η γλώσσα PHP βρίσκεται στην έκδοση *7.3.0* [12]*.* Είναι συμβατή με τους πιο συχνά χρησιμοποιούμενους διακομιστές παγκόσμιου ιστού όπως ο ISS της Microsoft και ο Apache. Μια σελίδα PHP υπόκειται σε επεξεργασία μέσω του συγκεκριμένου 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 [13].

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 ο διαχωριστής και διαχωρίζει την μια εντολή από την επόμενή της. Μέσα σε ένα αρχείο php μπορεί να ανοίξει και να κλείσει όσες φορές είναι απαραίτητο κώδικας. Ακόμη, σε περίπτωση που κομμάτια κώδικα πρέπει να επαναληφθούν πολλές φορές μέσα στο ίδιο αρχείο, μπορεί να γίνει ξεχωριστή συγγραφή ενός php αρχείο 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.



Σχήμα 2.9 Tρόπος σύνταξης της συνάρτησης require\_once στην PHP

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



Σχήμα 2.10 Tρόπος σύνταξης της PHP

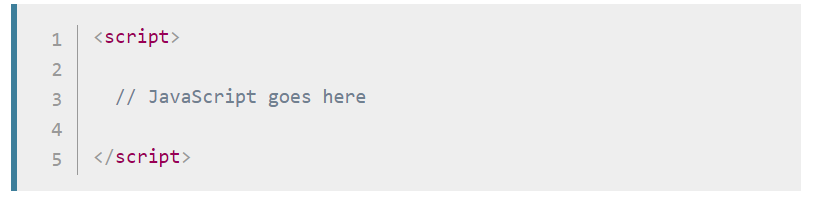
Στην εικόνα διακρίνουμε πώς η PHP γράφεται ανάμεσα στα στοιχεία <body> </body> της HTML. Στην PHP δεν είναι απαραίτητη η δήλωση των μεταβλητών πριν την ανάθεση μιας τιμής σε αυτές, όπως επίσης και η δήλωση του τύπου αυτών. Η PHP αυτόματα αποφασίζει τον τύπο ανάλογα με την τιμή που έχει τεθεί στην εκάστοτε 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.

* + 1. **ΓΛΩΣΣΑ 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 δε 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 [14].

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 ο 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 JavaScipt [15].

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.

Σχήμα 2.11 Tρόπος 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.

Σχήμα 2.12 Σύνδεση αρχείου JavaScript σε αρχείου HTML/PHP

Τα εξωτερικά αυτά αρχεία έχουν την κατάληξη .js και δεν πρέπει να περιέχουν τις ετικέτες <script>…</script>. Για την 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.

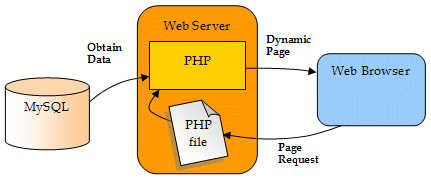


Σχήμα 2.13 Κομμάτι κώδικα σε JavaScript για την λειτουργικότητα του μενού της διαδικτυακής εφαρμογής Smart Homie

* + 1. **ΓΛΩΣΣΑ ΣΧΕΣΙΑΚΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ MySQL**

Η MySQL, η οποία είναι 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 [16]. Η MySQL είναι η πιο δημοφιλής βάση δεδομένων ανοιχτού κώδικα στον κόσμο, 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 στην βάση δεδομένων (ΒΔ) λειτουργούν αξιόπιστα. Η ιδέα του ACID είναι να αξιολογεί την δομή βάσεων δεδομένων και εφαρμογών. Όσον αφορά τις βάσεις δεδομένων μια λογική ενέργεια που έχει να κάνει με τα δεδομένα της αποκαλείται συναλλαγή [17]. 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 βοηθήσουν τους προγραμματιστές και τα DBA να δημιουργήσουν και να διαχειριστούν 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 να αναπτύξουμε πολλαπλά λειτουργικά συστήματα.
* Διαλειτουργικότητα μεγάλου όγκου δεδομένων μέσω της MySQL [18].

 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
    1. **ΥΛΙΚΟΣ ΕΞΟΠΛΙΣΜΟΣ - 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 κομμάτι συμπληρώνει το εγχείρημα που πραγματοποιήθηκε. Γίνεται εκτενής αναφορά στις δυνατότητες καθώς και στην επεκτασιμότητα που δέχεται το κάθε υλικό στοιχείο.

2. 2. 1. **ΜΙΚΡΟΕΛΕΓΚΤΗΣ ARDUINO UNO REV3**

*Ιστορία του Arduino*

Το 2005, με σκοπό την ευκολότερη εκμάθηση των ηλεκτρονικών για τους μαθητές, ο καθηγητής Massimo Banzi στο Ινστιστούτο 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 Cuatrielles, αποφάσισαν να φτιάξουν ένα μικροελεγκτή, ο οποίος ήταν πιο προσιτός ως προς τη χρήση του. Ονόμασαν το έργο τους Arduinο 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» [19].

*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 Machinstoh 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**: To 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**: To 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 [20].

*Διάφορες πλακέτες Arduino*

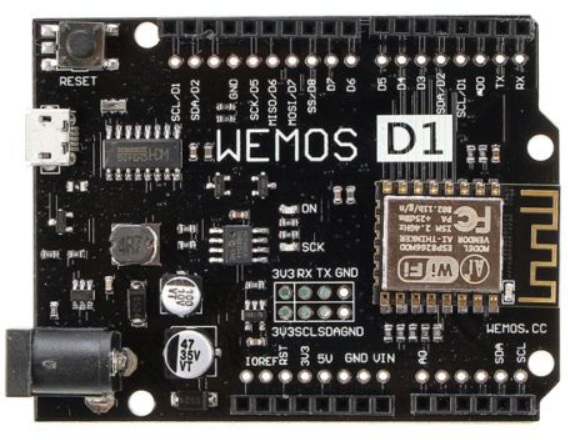
Στην παρούσα διπλωματική εργασία για λόγο ευκολίας αλλά και μεγάλης παραμετροποίησης σύμφωνα με τις ανάγκες της διαδικτυακής εφαρμογής επιλέχθηκε το arduino uno. Αυτό δεν σημαίνει πώς δεν υπάρχουν και άλλες πλακέτες arduino που είναι εξίσου αξιόλογες ανάλογα με την χρήση στην οποία υπόκεινται. Στον παρακάτω πίνακα παρατίθενται διάφορες ακόμα πλακέτες arduino καθώς και τα τεχνικά χαρακτηριστικά τους [21].

***Πίνακας 2.2: Είδη Arduino***

|  |  |
| --- | --- |
| **Πλακέτα Arduino** | **Τεχνικά Χαρακτηριστικά** |
| **Arduino Uno Rev3** | * 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 |
| **Arduino Ethernet** | * Μικροελεγκτής: ATMega328 * Τάση λειτουργίας: 5V * Ψηφιακοίείσοδοι/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 |
| **Arduino Fio** | * 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 |
| **Arduino Mega ADK** | * 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 |
| **Arduino Due** | * 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 32ΚΒ) * Ταχύτητα ρολογιού: 84MHz |
| **Arduino Leonardo** | * 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 |
| **Arduino Yun** | * 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 |
| **Arduino Micro** | * 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 |

* + 1. **ΜΙΚΡΟΕΛΕΓΚΤΗΣ 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 τον 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, την σύνδεση και τον χώρο που έχουμε στην διάθεση μας ανάλογα με το project μπορούμε να επιλέξουμε διαφορετικό μικροελεγκτή. Στο επόμενο πίνακα επεξηγούνται οι υπόλοιπες πλακέτες WeMos D1[22].

|  |  |
| --- | --- |
| **ΠΛΑΚΕΤΑ** | **ΤΕΧΝΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ** |
| **WeMos D1 Mini Pro** | * 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** | * 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) * Χρονισμόςπυρήνα: 80Mhz/160Mhz * Μνήμη Flash: 1MBytes * Μήκος: 34.2mm * Πλάτος: 25.6mm * Βάρος 10g |

* + 1. **ΔΡΟΜΟΛΟΓΗΤΗΣ - ROUTER**

Για την εκπόνηση της παρούσας διπλωματικής εργασίας καθώς και για την ανάπτυξη της διαδικτυακής εφαρμογής ήταν αναγκαία η τοποθέτηση ενός δρομολογητή (router) στην σύνδεση των μικροελεγκτών με τον εξυπηρετητή. δρομολογητής (αγγλ. 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 η διαδικασία μεταφοράς δεδομένων από ένα σημείο σε ένα άλλο 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 router είναι να συνδέει πολλαπλά δίκτυα και να προωθεί τα πακέτα που προορίζονται είτε στα δίκτυά του, είτε σε άλλα δίκτυα. Η προεπιλεγμένη (η 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 (η 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 [23].

Για τις ανάγκες της παρούσας διπλωματικής εργασίας χρησιμοποιήθηκε το δυναμικό 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.