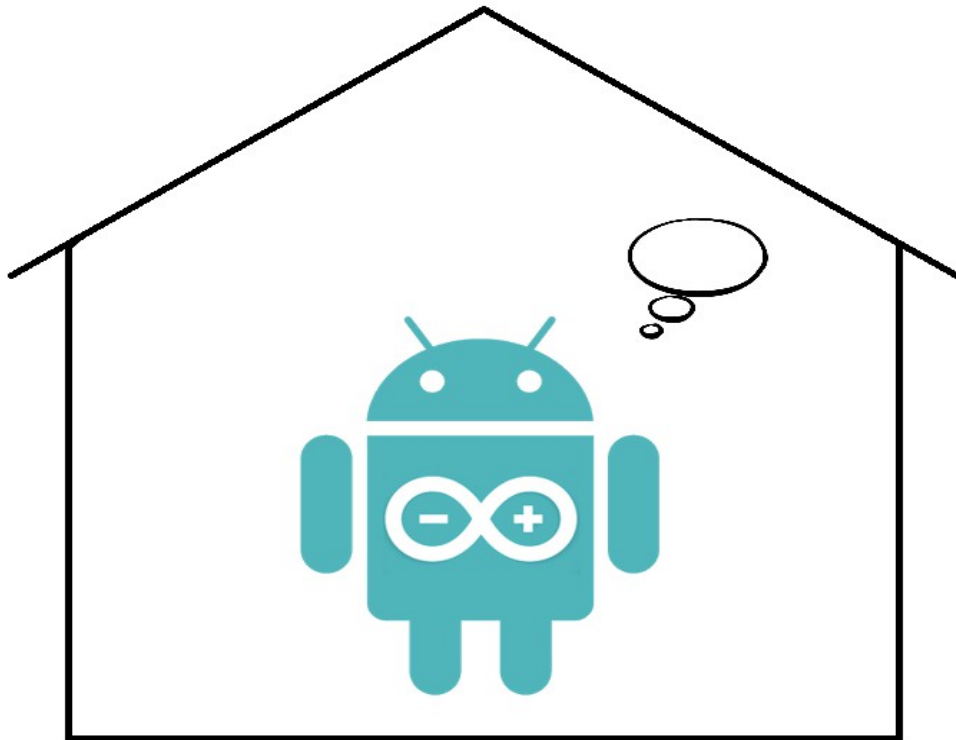




Πανεπιστήμιο Πειραιώς

Τμήμα Ψηφιακών Συστημάτων

Ανάπτυξη Υλικού και Λογισμικού για συστήματα ασφαλείας και απομακρυσμένου ελέγχου κατοικίας σε περιβάλλον Arduino



Επιβλέπον Καθηγητής : Μηλιώνης Απόστολος

Ονοματεπώνυμο φοιτητή : Φωτεινογιαννόπουλος Ανδρέας

Περιεχόμενα

Κεφάλαιο 1 :Εισαγωγή.....	4
1.1 Πρόλογος.....	4
1.2 Δομή.....	5
Κεφάλαιο 2 : Εισαγωγή Arduino-Έξυπνο Σπίτι.....	6
2.1 Περιγραφή Arduino.....	6
2.1.1 SainSmart.....	6
2.1.2 Αισθητήρες – Συσκευές.....	7
2.1.3 Arduino Software	11
2.2 Έξυπνο Σπίτι.....	12
2.2.1 Εισαγωγή.....	12
2.2.2 Παρουσίαση συστήματος.....	13
Κεφάλαιο 3 : Αναλυτική περιγραφή εργασίας.....	15
3.1 Προγραμματισμός Arduino-Κατασκευή.....	15
3.2 Λειτουργίες Server -Βάση δεδομένων.....	17
3.3 Interface χρήστη / Παρατηρητή.....	18
Κεφάλαιο 4 : Θεωρητικό υπόβαθρο.....	20
4.1 Προγραμματισμός Arduino / Κατασκευή.....	20
4.1.1 Παράδειγμα 1 : Απλό LED που αναβοσβήνει.....	20
4.1.2 Παράδειγμα 2 : Άναμμα LED με την πίεση ενός κουμπιού	22
4.1.3 Παράδειγμα 3 : Άναμμα LED με τον εντοπισμό κίνησης και ενημέρωση βάσης.....	24
4.2 Επεξήγηση στήσιμου βάσης/Server.....	28
4.3 Επεξήγηση κώδικα διασύνδεσης Arduino-Βάσης-Interface.....	30

Κεφάλαιο 5 : Σενάρια Χρήσης.....	32
5.1 Ψευδοκώδικας.....	32
5.1.1 Arduino-Ψευδοκώδικας.....	32
5.1.2 Interface-Ψευδοκώδικας.....	37
5.2 Περιπτώσεις Χρήσης.....	38
Διάγραμμα DFD	38
5.2.1 DFD Επίπεδο 0.....	39
5.2.2 DFD Επίπεδο 1.....	40
5.2.2.1 DFD 1.1.....	41
5.2.2.2 DFD 1.2.....	42
5.2.2.3 DFD 1.3.....	43
5.2.3 DFD Επίπεδο 2.....	44
5.2.3.1 DFD 2.1.....	44
Κεφάλαιο 6 : Μελλοντικές Επεκτάσεις / Ρεαλιστική Εφαρμογή.....	46
6.1 Μελλοντικές Επεκτάσεις.....	46
6.2 Ρεαλιστική Εφαρμογή.....	48
Κεφάλαιο 7 : Εγχειρίδιο χρήσης.....	49
7.1 Εντολές KeyPad.....	49
7.2 Εντολές IRControl.....	50
7.3 Επεξήγηση Interface.....	51
Βιβλιογραφία.....	58

Κεφάλαιο 1

Εισαγωγή

1.1 Προλογος

Σε αυτήν την πτυχιακή εργασία σκοπός μας είναι να δημιουργήσουμε ένα σύστημα ασφάλειας και απομακρυσμένης επέμβασης-παρακολούθησης της κατάστασης μιας οικίας με την βοήθεια του Arduino. Να φτιάξουμε δηλαδή ένα Έξυπνο Σπίτι , ένα σπίτι που θα αντιλαμβάνεται κινδύνους και θα προσπαθεί να τους αποτρέπει. Οι αισθητήρες και οι συσκευές που θα χρησιμοποιήσουμε είναι καθαρά για λόγους παρουσίασης του προγράμματος (σε μακέτα), το οποίο σημαίνει ότι με κάποιες αλλαγές στον κώδικα (που θα αναπτυχθούν σε ψευδοκώδικα στο κεφάλαιο 6.2) να μπορεί το σύστημά μας να λειτουργήσει σε ένα κανονικό σπίτι .Η εργασία μας αποτελείται από 3 βασικά κομμάτια

1. Προγραμματισμός Arduino
2. Server
3. Interface χρήστη

Οι λόγοι που μας οδήγησαν σε αυτή την εργασία είναι ότι :

- Όπως έχουν δείξει έρευνες οι περισσότερες ληστείες συμβαίνουν κατά την διάρκεια απουσίας των ενοίκων .Συνήθως τα σπίτια-στόχοι παρακολουθούνται μέχρι να είναι βέβαιο ότι δεν υπάρχει κάποιος μέσα
- Επίσης η αργή αντίληψη των κινδύνων σε ένα σπίτι οδηγούν απλά ατυχήματα να εξελιχθούν σε δυστυχήματα , όπως για παράδειγμα μία πυρκαγιά στην κουζίνα ή ένας σεισμός την ώρα που κοιμούνται οι ένοικοι .

Σκοπός λοιπόν της εργασίας μας είναι να δημιουργήσουμε ένα μηχανισμό-σύστημα ο οποίος να αναστέλλει και να καταπολεμεί αυτούς τους κινδύνους . Με την βοήθεια του Arduino και διαφόρων ελεγκτών θα κατασκευάσουμε

- ένα μηχανισμό ο οποίος θα προσομοιώνει την συμπεριφορά των ενοίκων ώστε όταν αυτοί είναι απόντες να μην γίνεται αντιληπτό
- θα ανταποκρίνεται σε εξωτερικά ερεθίσματα με τον ίδιο σκοπό
- θα ειδοποιεί έγκαιρα τους ενοίκους σε περίπτωση κινδύνων όπως φυσικές καταστροφές, ατυχήματα κλπ
- θα επιτρέπει στον χρήστη να ελέγχει κάποιες λειτουργίες του σπιτιού από το ίντερνετ
- θα επιτρέπει στον χρήστη να επιβλέπει τις λειτουργίες που συμβαίνουν στο σπίτι σε πραγματικό χρόνο μέσω ίντερνετ

1.2 Δομή

Στο **1ο** κεφάλαιο όπως είδαμε υπάρχει ο πρόλογος και η δομή της εργασίας μας

Στο **2ο** κεφάλαιο θα κάνουμε μία περιγραφή του τι είναι το Arduino και οι αισθητήρες που χρησιμοποιούμε καθώς και τα τεχνικά τους χαρακτηριστικά . Επίσης θα εξηγήσουμε τον όρο Έξυπνο Σπίτι

Στο **3ο** κεφάλαιο θα κάνουμε μία περιγραφή του συστήματός μας , της βάσης δεδομένων καθώς και του interface μας.

Στο **4ο** κεφάλαιο θα υπάρχει ένα “tutorial” , ένας οδηγός δηλαδή που θα εξηγεί μερικές βασικές λειτουργίες του Arduino . Με παραδείγματα θα προσπαθήσουμε να δείξουμε σε κάποιον που δεν έχει ασχοληθεί με το Arduino μερικές από τις δυνατότητές του με σκοπό όποιος το διαβάσει να ενδιαφερθεί και να καταλάβει ότι αν το δεις σε κομμάτια δεν είναι τόσο δύσκολο όσο μπορεί να φαντάζεται , ώστε να τον τραβήξουμε να ασχοληθεί με αυτό.

Στο **5ο** κεφάλαιο θα αναπτύξουμε σε ψευδοκώδικα το βασικό κομμάτι του κώδικα μας στο Arduino(δηλαδή το κομμάτι που παίρνει τις αποφάσεις)όπως και του κώδικα του Interface . Επίσης θα υπάρχουν κάποια σχήματα που θα κάνουν πιο κατανοητά τα σενάρια χρήσης και την ροή των δεδομένων μας .

Στο **6ο** κεφάλαιο θα παρουσιάσουμε με ψευδοκώδικα μελλοντικές επεκτάσεις που θα μπορούσαν να γίνουν στο σύστημά μας καθώς επίσης και τον τρόπο με τον οποίο το ίδιο σύστημα θα μπορούσε να έχει πραγματική εφαρμογή σε ένα σπίτι και όχι σε μακέτα.

Στο **7ο** κεφάλαιο υπάρχει ένα εγχειρίδιο χρήσης για οποιαδήποτε αλληλεπίδραση επιτρέπουμε στο χρήστη είτε μέσω του Key Pad είτε μέσω του Interface μας .

Κεφάλαιο 2

Εισαγωγή Arduino-Έξυπνο σπίτι

1. Περιγραφή Arduino
2. Έξυπνο Σπίτι

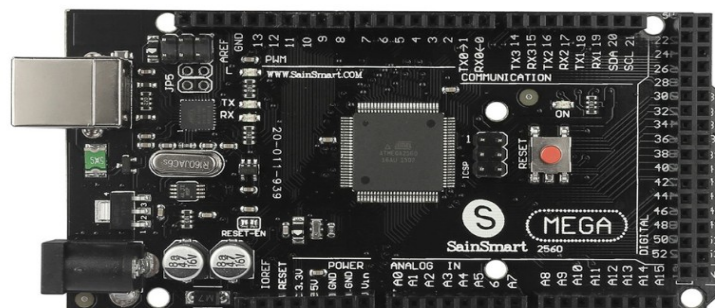
2.1.Περιγραφή Arduino

2.1.1 SainSmart

Το Arduino είναι ένα εργαλείο το οποίο χρησιμοποιείται για να φτιαχτεί ένας υπολογιστής ο οποίος να μπορεί να αλληλεπιδρά με τον φυσικό κόσμο . Δηλαδή να μπορεί να αντιλαμβάνεται τις συνθήκες που επικρατούν μέσω αισθητήρων και να επιδρά σε αυτές μέσω κάποιων συσκευών (Led, Camera, Moter κτλ). Είναι μια open-source πλατφόρμα βασισμένη σε έναν μικροελεγκτή και ένα προγραμματιστικό περιβάλλον για να αναπτύσσεται κώδικας. Τα project του Arduino μπορεί να εκτελούνται εξολοκλήρου από το Arduino και τις συνδεδεμένες σε αυτό συσκευές και αισθητήρες ή μπορούν να αλληλεπιδρούν με άλλα προγράμματα που εκτελούνται σε κάποιον υπολογιστή. Το προγραμματιστικό κομμάτι του Arduino βασίζεται στην γλώσσα Processing η οποία μοιάζει αρκετά στη C++ .

Η συσκευή SainSmart MEGA2560 R3 που θα χρησιμοποιήσουμε έχει τα παρακάτω στοιχεία :

- Μικροελεγκτή Atmega2560
- Ρυθμιστή τάσης 5V
- Κρυσταλλικό ταλαντωτή 16 Mhz (Αντίληψη Χρόνου)
- Είσοδο ρεύματος (7-12V)
- 54 ψηφιακές εισόδους (15 μπορούν να παράγουν αναλογική έξοδο)
- 16 αναλογικές εισόδους
- 256KB Flash Memory (Από τα οποία τα 8 χρησιμοποιούνται για Bootloader)
- SRAM 8KB
- EEPROM 4KB



2.1.2 Αισθητήρες – Συσκευές

Led : Τα φωτάκια που αναβοσβήνουν ανάλογα με το πρόγραμμά μας. Υπάρχει ένας θετικός(μακρύς) και ένας αρνητικός πόλος. Ο θετικός ενώνεται με τα Pin μας που του δίνουν τάση και ο αρνητικός με την γείωση GND .



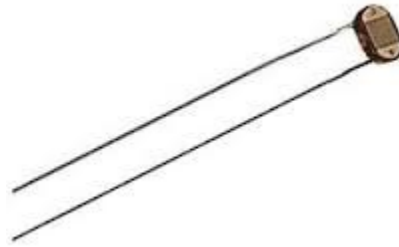
Buzzer : Το Buzzer μπορεί να παράγει ήχους ανάλογα με την συχνότητα που επιθυμούμε. Έχει 2 pins , ένα για GND και ένα για την σύνδεσή μας που του παρέχει τάση .



TiltSensor: Περιέχει μία μπάλα που αν ξεπεραστεί μια συγκεκριμένη γωνία κλίσης ενεργοποιεί τον αισθητήρα μας. Έχει 2 Pins, μία σύνδεση και ένα GND



Photoresistor :Αντιλαμβάνεται την ένταση φωτεινότητας στο χώρο. Έχει 2 pins, ένα GND και ένα για την σύνδεση



Temperature Sensor :Αντιλαμβάνεται την θερμοκρασία στο χώρο και ανάλογα αυξομειώνει την τάση εξόδου του .Έχει 3 pins όπως το βλέπουμε από κάτω αριστερά ρεύμα , στο κέντρο την έξοδο , δεξιά το GND



Move Sensor PIR : εξετάζει την Infrared ακτινοβολία που εκπέμπεται από όλα τα αντικείμενα με αποτέλεσμα αν περάσει κάποιος, αυτή να αλλάξει και να “ενεργοποιηθεί “.Έχει 3 pins, για ρεύμα(αριστερά από πίσω) , για γείωση(δεξιά) και για σύνδεση(μέση) .



Keypad : Το Keypad μας επιτρέπει να έχουμε 4X4 εντολές για το σύστημά μας, έχει 8 pins που τα συνδέουμε με εισόδους (ρυθμίζουμε το χάρτη στο σύστημά μας)



HanRun Ethernet : Η συγκεκριμένη συσκευή μας προσφέρει πρόσβαση στο internet για εκτέλεση απλών εντολών . Διαθέτει πολλά pins για ξεχωριστές εισόδους-εξόδους.



GSM Shield : Η συσκευή αυτή μας δίνει την δυνατότητα να χρησιμοποιούμε με το Arduino διάφορες λειτουργίες που θα έκανε ένα τηλέφωνο . Αποστολή Λήψη μηνυμάτων , GPS κ άλλα.



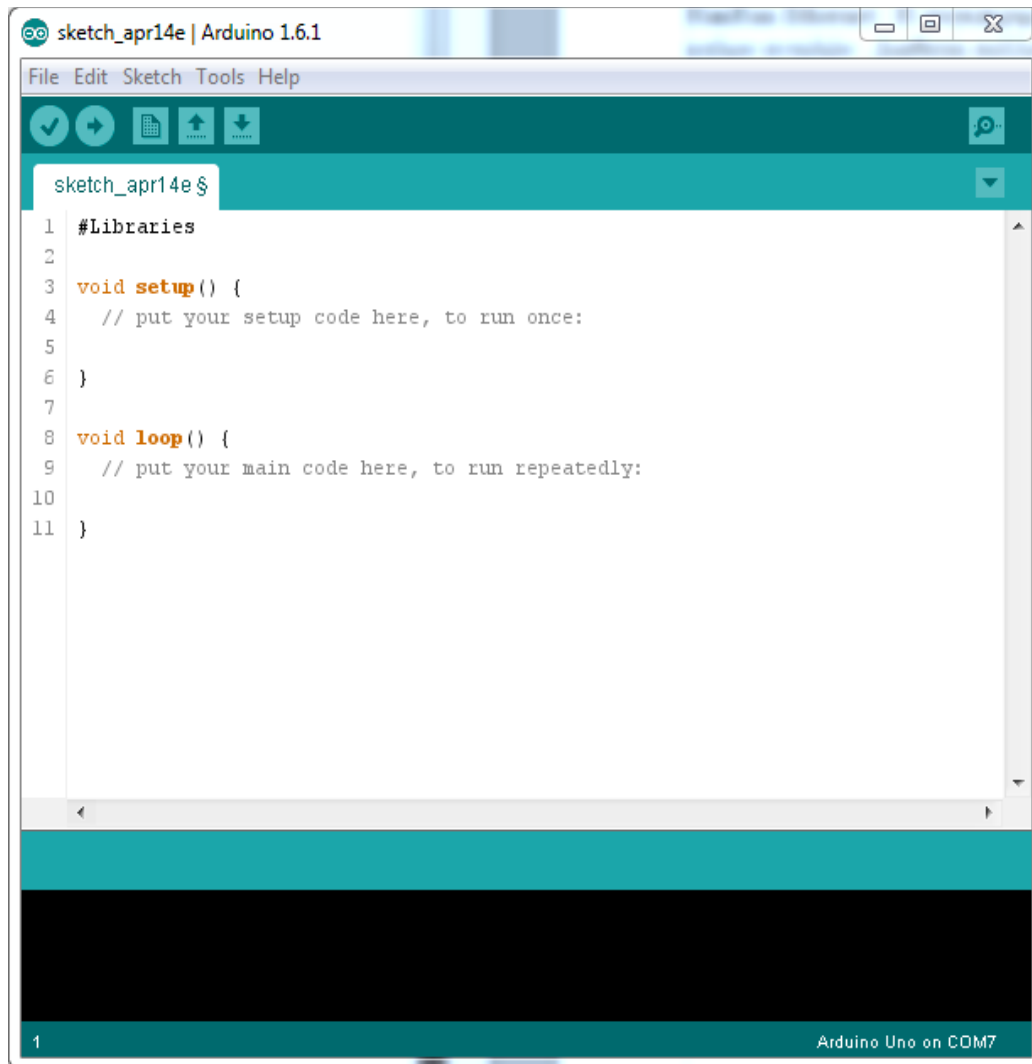
Infrared Control: Η συσκευή αυτή μας επιτρέπει με το πάτημα ενός κουμπιού να δίνουμε εντολές στο σύστημά μας από απόσταση . Η κεραία έχει 3 pins (Πίσω Αριστερά Ρεύμα, GND, Σύνδεση)



2.1.3 Arduino Software

Η γλώσσα στην οποία γράφουμε μοιάζει αρκετά με τις C/C++ . Το πρόγραμμα είναι ελεύθερο και μπορούμε να το κατεβάσουμε από το site του Arduino .

Αυτό είναι το πρόγραμμα στο οποίο γράφουμε τον κώδικα μας . Συνήθως θα υπάρχουν τρία κομμάτια στον κώδικα .



```
sketch_apr14e $
1 #Libraries
2
3 void setup() {
4 // put your setup code here, to run once:
5
6 }
7
8 void loop() {
9 // put your main code here, to run repeatedly:
10
11 }
```

1. Δηλώνουμε και εισάγουμε τις βιβλιοθήκες που χρησιμοποιούμε (#όνομα βιβλιοθήκης). Βιβλιοθήκη είναι ένα σύνολο συναρτήσεων έτοιμες για χρήση .
2. `void setup () {`
Εδώ γίνονται οι δηλώσεις των μεταβλητών μας και τα κομμάτια του προγράμματος που τρέχουν 1 φορά , για να “σετάρουμε” κάποιους αισθητήρες κλπ
}
3. `void loop () {`
Εδώ υπάρχει το κομμάτι του κώδικα που τρέχει συνεχώς σε επανάληψη.
}

2.2 Έξυπνο Σπίτι

2.2.1 Εισαγωγή

Με τον όρο έξυπνο σπίτι εννοούμε τον έλεγχο οικιακών συσκευών εξ αποστάσεως ή αυτόματα. Αυτό μπορεί να περιλαμβάνει το καλοριφέρ , τα παντζούρια , το θερμοσίφωνο , κάμερες και οποιαδήποτε ηλεκτρονική συσκευή . Αυτά μπορεί να λειτουργούν αυτόματα με βάση τις εντολές που έχει δώσει ο ιδιοκτήτης ή μπορεί ο ίδιος μέσω internet να τα ενεργοποιεί και να τους δίνει εντολές . Τα πιο δημοφιλή συστήματα είναι ο έλεγχος θερμοκρασίας και η προσαρμογή της , συστήματα ασφαλείας, έλεγχος ηλεκτρικών συσκευών κλπ .

Όσο περνάει ο καιρός βλέπουμε το έξυπνο σπίτι και τις έξυπνες συσκευές να μπαίνουν όλο και περισσότερο στην καθημερινότητα μας . Από τους συναγερμούς που πλέον υπάρχουν στα περισσότερα σπίτια , τα αυτόματα καλοριφέρ που ανάβουν όταν η θερμοκρασία πέσει κάτω από κάποιο επίπεδο μέχρι τα “έξυπνα” ψυγεία τα οποία αντιλαμβάνονται τι τρόφιμα έχουν έλλειψη και ενημερώνουν τον ιδιοκτήτη ή ακόμα παραγγέλνουν μόνα τους !

Με βάση αυτά ήταν λογικό η επόμενη ιδέα να είναι ένα έξυπνο σπίτι που να ταιριάζει στις ανάγκες μας , να μας κάνει την ζωή πιο εύκολη και ασφαλή .

Εμείς στην εργασία μας θα παράγουμε ένα σύστημα πρόληψης- ασφάλειας το οποίο θα αντιλαμβάνεται κινδύνους και θα προσπαθεί να τους αποτρέψει . Παράλληλα θα μας δίνει την δυνατότητα απομακρυσμένα να έχουμε “εικόνα” του τι συμβαίνει στην οικία μας αλλά και να επεμβαίνουμε σε κάποιους τομείς .

2.2.2 Παρουσίαση Συστήματος

Σε αυτό το σημείο θα παρουσιάσουμε ένα από τα πιο γνωστά συστήματα της αγοράς. Το σύστημα είναι της εταιρίας HomeSeer .

Το εν λόγω σύστημα χρησιμοποιεί αισθητήρες στις πόρτες και στα παράθυρα , στην πόρτα του γκαράζ , περιλαμβάνει κάμερες ασφαλείας εξωτερικού χώρου καθώς κ ελέγχει κάποια ποτιστικά για τα λουλούδια. Ακόμα έχει αισθητήρες κίνησης και συσκευές που επιτρέπουν τον έλεγχο άλλων συσκευών πχ Infrared Transmitter . Το σύστημα όχι μόνο ελέγχει τις συσκευές αλλά μας επιτρέπει να παρακολουθούμε τι γίνεται και να ελέγχουμε κάποιες από τις συσκευές από απόσταση .

Συμβατές συσκευές :

- **Hardware Controller** : Ένα κοντρόλερ που επιτρέπει να αλληλεπιδρούμε με το σύστημα.
- **Energy management** : Μπορεί να παρακολουθεί την κατανάλωση ενέργειας από τις συσκευές .
- **Lighting Systems** : Το σύστημα μπορεί να ελέγχει τα φώτα είτε αυτόματα είτε να δίνει τη δυνατότητα στον χρήστη να ελέγχει από απόσταση .
- **Thermostats** : Μπορεί να μετράει τη θερμοκρασία του σπιτιού και να την κρατάει σταθερή χρησιμοποιώντας τις ανάλογες συσκευές .
- **Windows Coveringw** : Μπορεί να ρυθμίζει τη φωτεινότητα στο σπίτι μέσω των παντζουριών , κουρτινών κλπ
- **Enviromental Sensors** : Μπορεί να μετράει εξωτερική θερμοκρασία , να παρατηρεί τον καιρό κλπ.
- **Door Locks** : Επιβλέπει τις κλειδαριές .
- **Garage Door Openers** :Επιτρέπει στον χρήστη αν ανοιγοκλείνει την πόρτα του γκαράζ από απόσταση
- **Security Cameras** :Επιβλέπει τις κάμερες και δίνει την δυνατότητα ο χρήστης να παρακολουθεί το σπίτι από απόσταση
- **Entry Sensors** : Παρακολουθεί αν ανοίγουν οι πόρτες και τα παράθυρα .

Λειτουργικότητα :

- **Remote Access** : Απομακρυσμένη επίβλεψη του σπιτιού και των συσκευών καθώς και απομακρυσμένος έλεγχός τους.
- **Randomize Programs**: Τυχαία ενεργοποίηση συσκευών για προσομοίωση ύπαρξης ενοίκων σε περίπτωση που λείπουν
- **Touchscreen** :Κοντρόλερ για τις συσκευές με λειτουργία αφής .
- **Voice Control** :Χρήση των συσκευών μέσω φωνητικών εντολών
- **Wired System** : Διασύνδεση επιπλέον συσκευών

Παραδείγματα χρήσης

Αυλή : Δυνατότητα ρύθμισης αυτόματου ποτίσματος με πρόγραμμα ώστε να ποτίζονται τα φυτά βάση αυτού ή ακόμα και δυνατότητα να ποτίζουμε εμείς τα φυτά με μία εντολή μέσω κινητού .

Σαλόνι :Δυνατότητα να ανάβουμε/ σβήνουμε τα φώτα με μία λέξη ! Ακόμα μπορούμε να επιλέξουμε τραγούδι ή ταινία στο MP3 μας ή την τηλεόρασή μας απλά και μόνο λέγοντας το .

Κουζίνα : Ελεγκτές κίνησης στα παράθυρα για ασφάλεια . Ακόμα μέσω ελεγκτών κίνησης στον διάδρομο μπορούν να ανάβουν τα φώτα της κουζίνας όταν εμείς κινούμαστε προς αυτήν . Κουνώντας το χέρι μας προς την καφετιέρα αυτή ανάβει και μας ετοιμάζει τον καφέ μας . Σε περίπτωση που έχουμε φύγει και αναρωτιόμαστε αν έχουμε ξεχάσει ανοιχτό το ψυγείο μπορούμε να το δούμε μέσω του τηλεφώνου μας.

Μπάνιο : Μπορούμε μέσω του τηλεφώνου μας να ανάψουμε το θερμοσίφωνο όταν είμαστε στο δρόμο για το σπίτι ή να έχουμε προγραμματίσει πότε θέλουμε να υπάρχει ζεστό νερό και το σύστημα να φροντίζει για αυτό μόνο του .

Application : Παρακολούθηση και έλεγχος των συσκευών μας όταν δεν βρισκόμαστε στο σπίτι .

Παραπάνω είδαμε μερικές λειτουργίες που μας προσφέρει ένα γνωστό σύστημα . Εμείς βάση των αναγκών μας έχουμε επιλέξει κάποιες από αυτές τις λειτουργίες και τις έχουμε αναπτύξει στο δικό μας σύστημα μαζί με κάποιες άλλες που μας φαίνονται χρήσιμες .Έτσι με αρκετά λιγότερα χρήματα μπορούμε να φτιάξουμε το δικό μας έξυπνο σπίτι το οποίο να ταιριάζει απολύτως στις ανάγκες μας για ασφάλεια και “καλοπέραση”.

Κεφάλαιο 3

Αναλυτική περιγραφή εργασίας

Η εργασία μας αποτελείται από 3 βασικά μέρη

1. Προγραμματισμός Arduino – Κατασκευή
2. Λειτουργίες Server – Βάση δεδομένων
3. User Interface

3.1 Προγραμματισμός Arduino – Κατασκευή

Σε αυτή την παράγραφο θα εξηγήσουμε αναλυτικά τις λειτουργίες που αναπτύξαμε στο Arduino .

Το Σύστημα μας αποτελείται από :

- Ελεγκτές Κίνησης
- Ελεγκτές Θερμότητας
- Ελεγκτές Επιπέδου Φωτεινότητας
- Ελεγκτές Κλίσης/Tilt
- Πληκτρολόγιο
- Μοτέρ
- Λαμπάκια
- Μπάζερ
- Πύλη ethernet
- GSM Shield
- Infrared Control

- Δύο Ελεγκτές κίνησης .

Ο **ένας ελεγκτής** βρίσκεται στο ένα παράθυρο .Όταν εντοπιστεί κίνηση τίθεται σε λειτουργία για μικρό χρονικό διάστημα το μπάζερ ,ώστε να παράγει θόρυβο, και ανάβει το ένα λαμπάκι . Με αυτόν τον τρόπο προσομοιώνουμε την ύπαρξη ενοίκων στην οικία .

Ο **δεύτερος ελεγκτής** κίνησης βρίσκεται μέσα στην οικία . Όταν ενεργοποιηθεί (Πράγμα που σημαίνει ότι κάποιος έχει εισβάλει στην οικία) αποστέλλει ένα e-mail και ένα SMS στον ιδιοκτήτη ώστε να προβεί σε όποια ενέργεια αποφασίσει και ενεργοποιεί τον συναγερμό (φώτα και μπάζερ) ώστε να ειδοποιηθεί ο ιδιοκτήτης σε περίπτωση που είναι εντός της οικίας (κοιμάται κλπ)

- Ο ελεγκτής **θερμότητας** ενεργοποιείται όταν η θερμοκρασία του σπιτιού ανέβει πάνω από ένα επίπεδο . Σε αυτήν την περίπτωση έχουμε ενεργοποίηση του συναγερμού ώστε ο ένοικος να αντιληφθεί άμεσα τον κίνδυνο (πυρκαγιά κλπ)
- Ο ελεγκτής **φωτεινότητας** ενεργοποιείται όταν το επίπεδο φωτεινότητας πέσει κάτω από κάποια ορισμένη τιμή . Τότε ανάβει ένα φως στο σπίτι ώστε να φαίνεται από το εξωτερικό του σπιτιού ότι κάποιος είναι μέσα (κάθε “βράδυ” ανάβει ένα φως ώστε σε περίπτωση διακοπών πχ να μην φαίνεται ότι το σπίτι δεν κατοικείται τα βράδια)
- Ο ελεγκτής **κλίσης** ενεργοποιείται με το τράνταγμα , σε περίπτωση σεισμού ενεργοποιεί τον συναγερμό ώστε να αφυπνιστεί ο ιδιοκτήτης .
- Το **πληκτρολόγιο** δίνει δυνατότητα κλειδώματος της πόρτας με κωδικό και ανοίγματος της (**μοτέρ**). Όπως επίσης και κάποιες εντολές του χρήστη προς την βάση.
- Η πύλη **Ethernet** μας επιτρέπει να επικοινωνούμε με την βάση δεδομένων μας , να στέλνουμε και να παίρνουμε δεδομένα
- Η **GSM Shield** μας δίνει την δυνατότητα με μία κάρτα SIM να στέλνουμε SMS σε κάποιο κινητό.
- Το **IRControl** μας επιτρέπει από κοντινή απόσταση να επεμβαίνουμε στο σύστημα , να αναβοσβήνουμε τα φώτα , να ανοιγοκλείνουμε την πόρτα και να αλλάζουμε την λειτουργία στο KeyPad

Λίγα λόγια για την βασική δομή του συστήματος :

Το σύστημά μας διαβάζει τη βάση ανά τακτά χρονικά διαστήματα ώστε να βλέπει τις εντολές για τα LED και το τηλέφωνο χρήστη . Έπειτα το σύστημα έχει 2 μέρη

1. Χρήση IRControl . Όταν είναι ενεργό το IRControl οι αισθητήρες μας είναι ανενεργό και μπορούν να γίνουν οι ενέργειες που αναπτύξαμε παραπάνω μέσω του τηλεχειριστηρίου.
2. Χρήση KeyPad . Όταν το keyPad είναι ενεργό λειτουργούν όλοι οι αισθητήρες μας όπως εξηγούμε παραπάνω. Εμείς μπορούμε να απενεργοποιήσουμε τον εσωτερικό αισθητήρα κίνησης ώστε να κινούμαστε ελεύθερα μέσα στο σπίτι ή να τον ενεργοποιήσουμε για να έχουμε πλήρη ασφάλεια όταν λείπουμε

3.2 Λειτουργίες Server – Βάση δεδομένων

Ο Server μας χρειάζεται ώστε να μπορούμε να αποθηκεύουμε τα δεδομένα μας και να επικοινωνούμε με το σύστημα. Έχουμε νοικιάσει χώρο σε έναν σέρβερ ο οποίος μας προσφέρει απομακρυσμένες εντολές καθώς επίσης και διάφορα εργαλεία για παρασκευή βάσης, αποθήκευση αρχείων κλπ

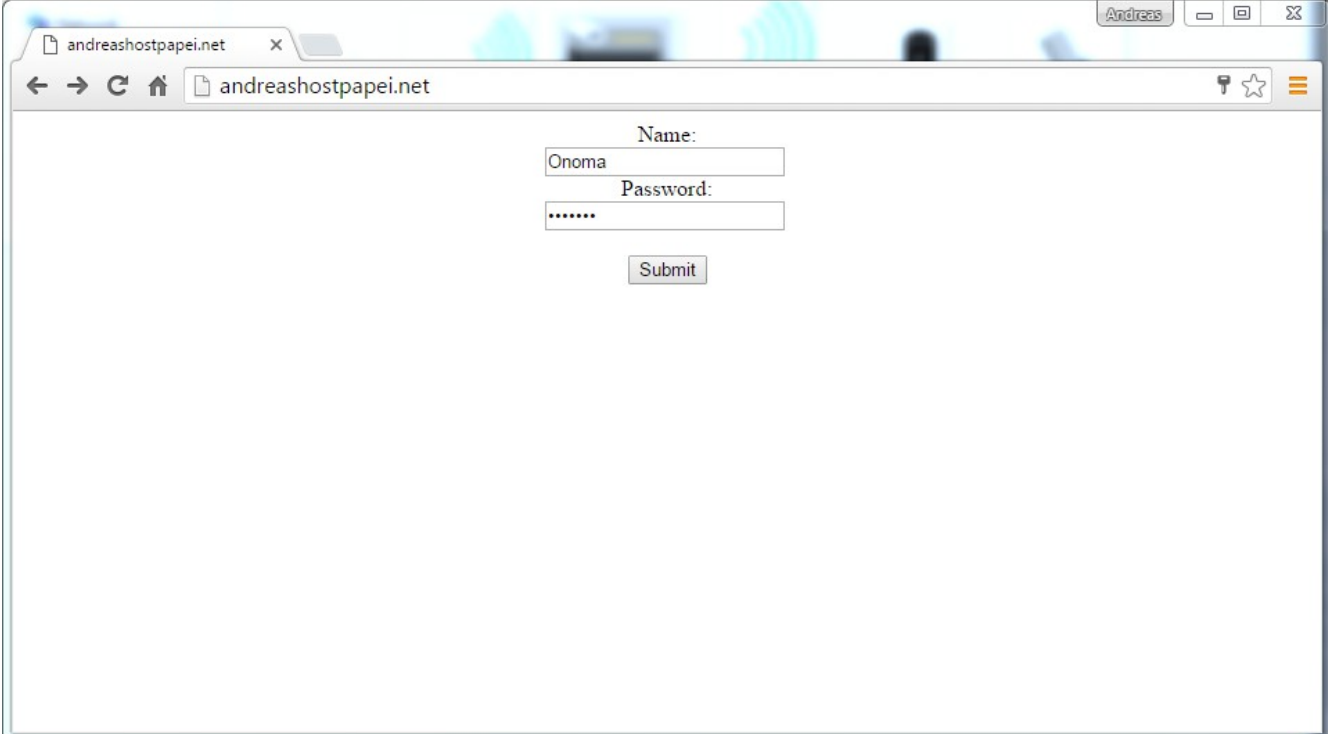
Βάση δεδομένων : Δημιουργήσαμε την βάση μας (MySQL) μέσω του “εργαλείου” phpMyAdmin που προσφέρεται από το server .

Η βάση μας περιέχει τις εγγραφές

- Control (ID,Mode,Time):Δέχεται τιμές από την διεπαφή χρήσης και οι τιμές της χρησιμοποιούνται για να α
- ανάβει ή να σβήνει μόνιμα κάποιο φως .
- KeyPad (Mode,Time,Times):Δέχεται τις τιμές από τα πατήματα στο πληκτρολόγιο .
- Led1(Mode,Time,Times):Καταγράφει πότε ανάβει το φως.
- Led2(Mode,Time,Times):Καταγράφει πότε ανάβει το φως.
- MoveSensorOne(Mode,Time,Times):Δέχεται τις τιμές όποτε ενεργοποιείται ο πρώτος ελεγκτής .κίνησης .
- MoveSensorTwo(Mode,Time,Times):Δέχεται τις τιμές όποτε ενεργοποιείται ο δεύτερος ελεγκτής κίνησης .
- ServoSensor(Mode,Time,Times):Αποθηκεύει τις κινήσεις του μοτέρ (άνοιγμα – κλείσιμο πόρτας).
- TempSensor(Fire,Temp,Time,Times):Δέχεται τις τιμές της θερμοκρασίας.
- TiltSensor(Time,Times):Δέχεται τις τιμές του αισθητήρα κλίσης (τρέμουλο).
- Number (ID , Telephone): Δέχεται τον αριθμό κλήσης που εισάγουμε μέσω του Web Page μας

3.3 Interface Χρήστη / Παρατηρητή

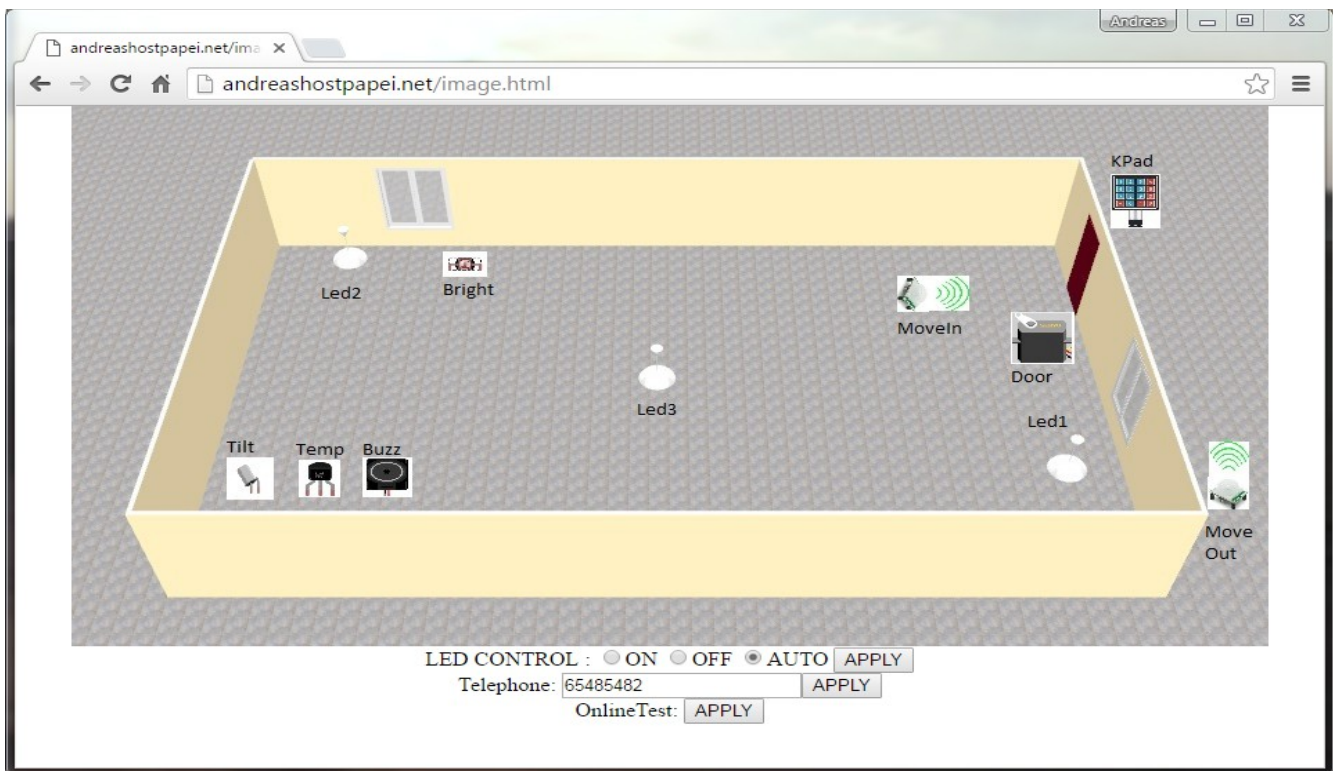
1 Με το που μπαίνουμε στην σελίδα μας , εμφανίζεται ένα πεδίο πιστοποίηση στοιχείων για να αποκτήσουμε πρόσβαση στις λειτουργίες που έχουμε αναπτύξει .



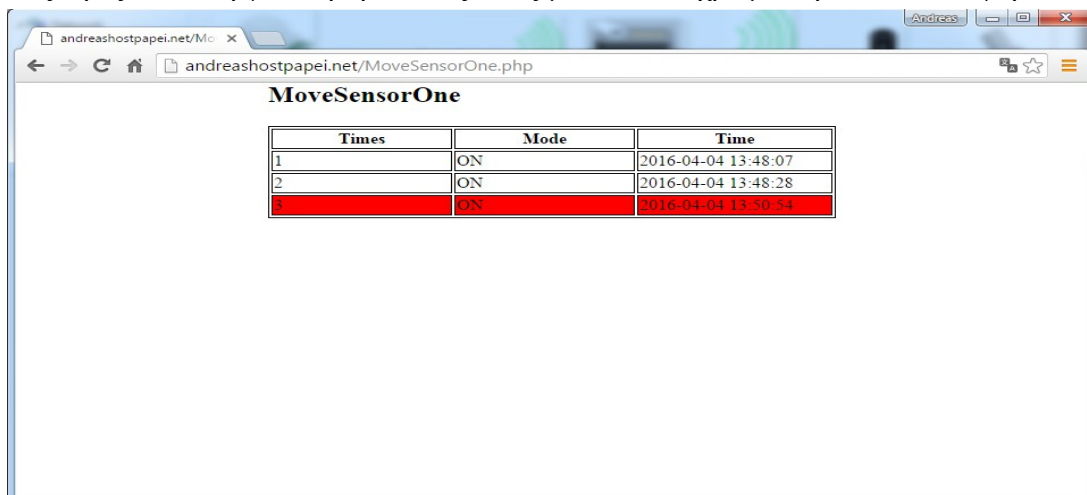
The screenshot shows a web browser window with the address bar displaying 'andreashostpapei.net'. The page content is a simple login form with the following elements:

- A label 'Name:' above a text input field containing the text 'Onoma'.
- A label 'Password:' above a password input field containing seven dots '.....'.
- A 'Submit' button located below the password field.

Με την επιτυχημένη εισαγωγής στοιχείων μας εμφανίζεται η κύρια οθόνη μας , η οποία είναι μια κάτοψη του χώρου στον οποίο έχει τοποθετηθεί το σύστημά μας και λειτουργεί σαν χάρτης ,δηλαδή μπορούμε να πατήσουμε πάνω στα περισσότερα εικονίδια και να δούμε πότε έχουν ενεργοποιηθεί . Έτσι μπορούμε από οπουδήποτε να έχουμε μια εικόνα του τι συμβαίνει στο σπίτι μας .



Παράδειγμα : Όταν κλικάρουμε πάνω στον Εσωτερικό ελεγκτή κίνησης , θα μας παρουσιαστεί μία σελίδα με τις ώρες που ενεργοποιήθηκε τονίζοντας με κόκκινο χρώμα την τελευταία φορά .



1. Στο πεδίο Telephone βάζουμε τον αριθμό στον οποίο θέλουμε να αποστέλλει το μήνυμα το Arduino και πατάμε **Apply** .
2. Ακόμα μπορούμε να παρέμβουμε στον φωτισμό του σπιτιού . Μπορούμε να ανάψουμε/σβήσουμε τα φώτα ή να τα αφήσουμε στον αυτόματο (δηλαδή να επιλέγει το πρόγραμμά μας τι θα κάνει ,όπως το έχουμε προγραμματίσει).Επιλογή στο κάτω μέρος της οθόνης μας **LED CONTROL :ON OFF AUTO**
3. Μπορούμε να ελέγχουμε αν το πρόγραμμά μας είναι **Online** πατώντας στο OnlineTest

Κεφάλαιο 4

Θεωρητικό υπόβαθρο

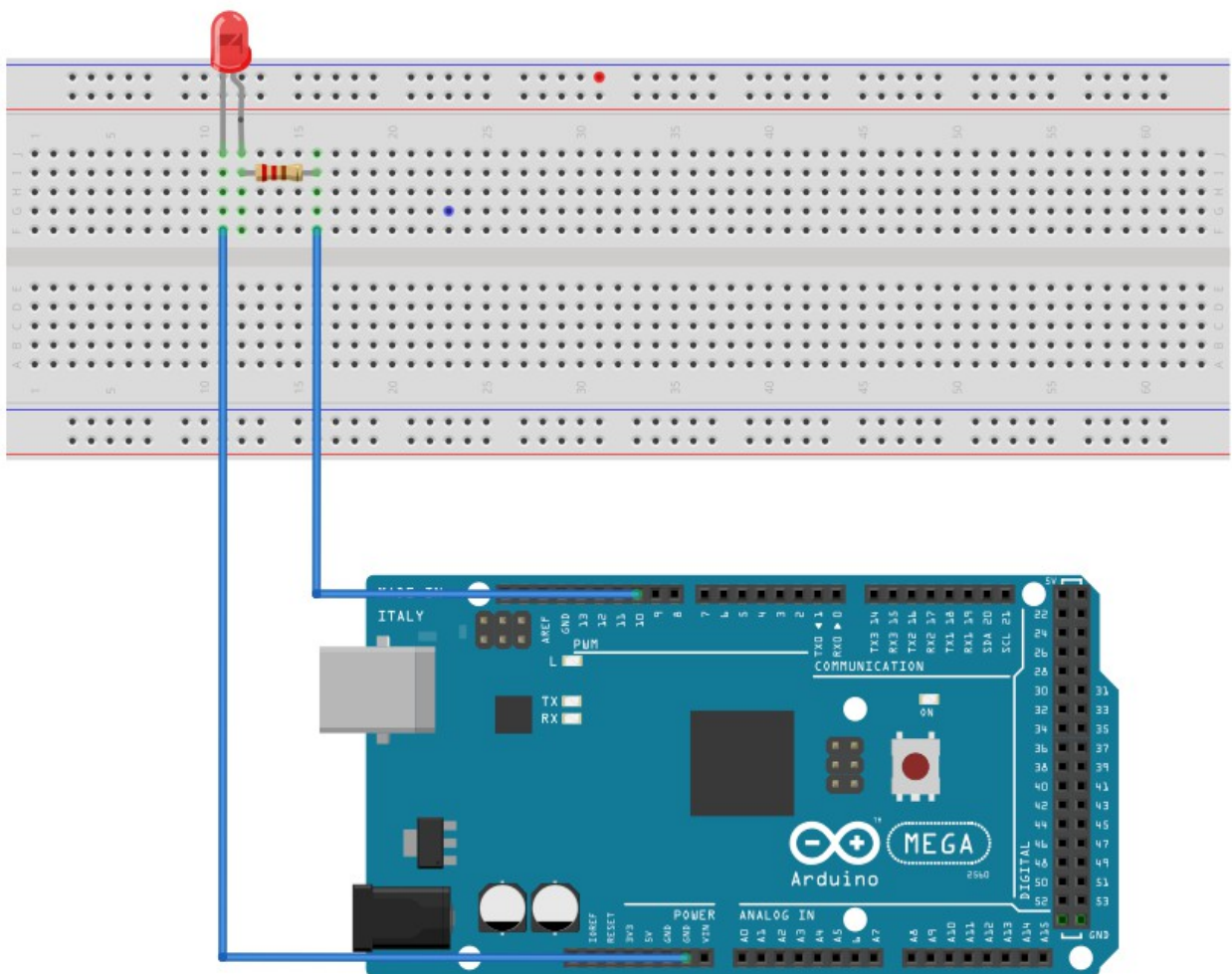
Σε αυτό το κεφάλαιο θα προσπαθήσουμε με μερικά απλά παραδείγματα να σας βάλουμε στον κόσμο του Arduino . Να σας δείξουμε δηλαδή ελάχιστες από τις διεργασίες που μπορεί να εκτελέσει .

4.1 Προγραμματισμός Arduino / Κατασκευή(Λειτουργίες)

4.1.1 Παράδειγμα 1 : Απλό LED που αναβοσβήνει

Σύνδεση

Συνδέουμε τη θέση 10 μέσω μια αντίστασης στο θετικό πόλο (πιο κοντό καλώδιο) του LED , και τον αρνητικό πόλο στο GND (γείωση)



Κώδικας



```
Blink | Arduino 1.6.1
File Edit Sketch Tools Help
Blink
1
2
3 int LedPin = 10 ; //Σε αυτήτη θέση βρίσκεται το λαμπάκι μας , το δηλώνουμε σαν μεταβλητη
4
5 void setup() {
6     // Δειχνουμε οτι στη θέση 10 βρίσκεται μια έξοδος (LED)
7     pinMode(LedPin, OUTPUT);
8 }
9
10 // Η παρακάτω διεργασία είναι το πρόγραμμα μας κ τρέχει σε επανάληψη συνέχεια
11
12 void loop() {
13     digitalWrite(LedPin, HIGH); // Ανάβει το λαμπάκι
14     delay(5000); // Περιμενει 5" , δηλαδή το λαμπάκι μένει αναμμένο 5 δευτερόλεπτα
15     digitalWrite(LedPin, LOW); // Σβήνει το λαμπάκι
16     delay(2000); // Περιμενει 2" , δηλαδή το λαμπάκι μένει σβηστό 2 δευτερόλεπτα
17 }
Done compiling.
Global variables use 11 bytes (0%) of dynamic memory, leaving 8,181 bytes for local variables. Maximum is 8,192 bytes.
16 Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3
```

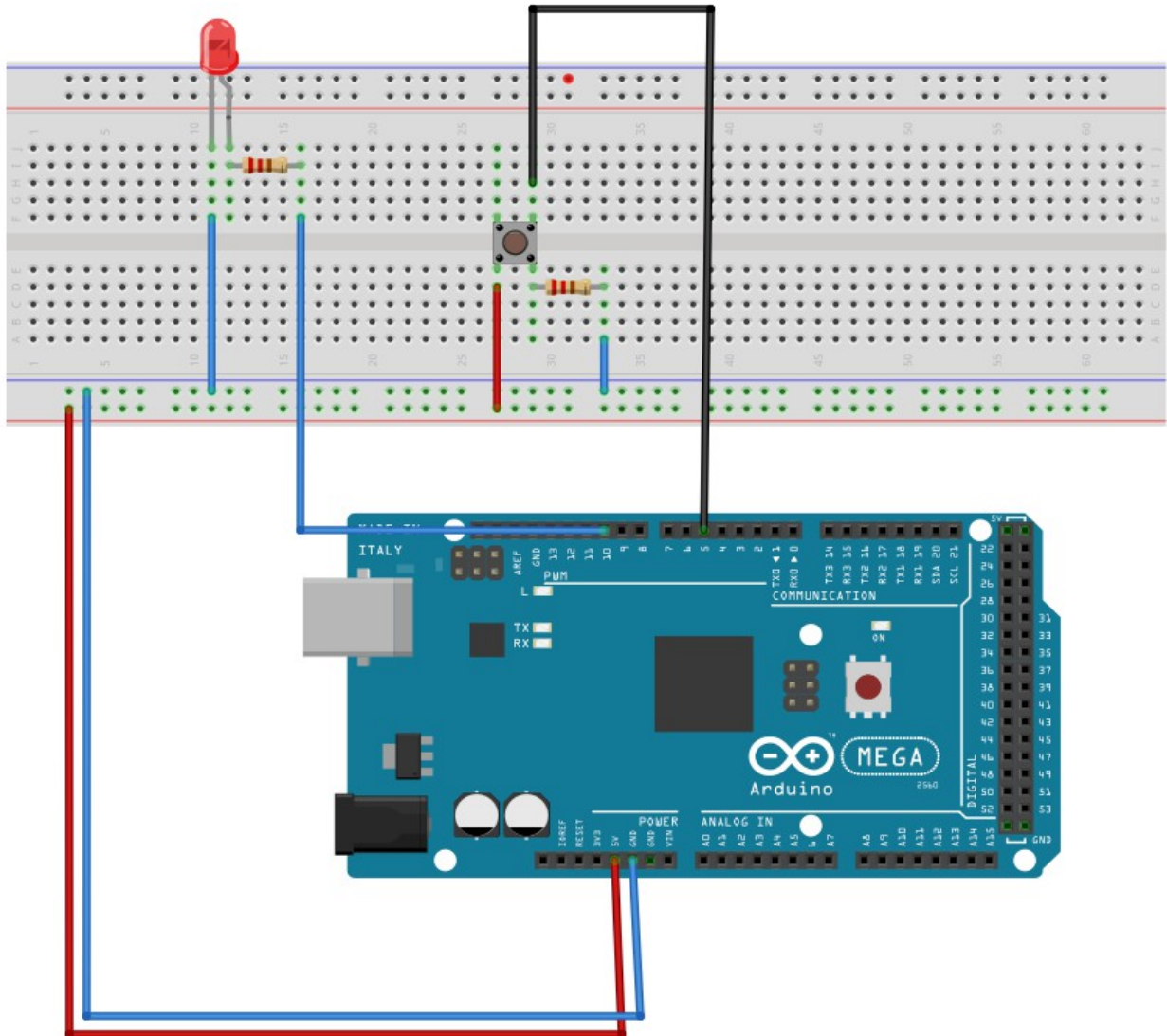
Αυτό ήταν ! Κάνοντας Upload Θα δούμε ότι σε μερικά δευτερόλεπτα το λαμπάκι μας θα αρχίσει να αναβοσβήνει με συχνότητα 5/2 (5" αναμμένο / 2" σβηστό)

Επεξήγηση κώδικα : Το καλύπτουν τα comments (//comment)

4.1.2 Παράδειγμα 2 : Αναμα LED με την πίεση ενός κουμπιού

Σύνδεση

Συνδέουμε το κουμπί και το LED όπως βλέπουμε στο παρακάτω σχήμα (Η αντίσταση στο κουμπί είναι 10Ω)



Κώδικας

```
1
2
3
4 const int buttonPin = 5;    // Η θέση του διακόπτη
5 const int ledPin = 10;     // Η θέση του LED
6
7 // variables will change:
8 int buttonState = 0;       // Μεταβλητή για την κατάσταση του κουμπιού
9
10 void setup() {
11   // Δείχνουμε ότι στη θέση ledPin βρίσκεται μια έξοδος (LED)
12   pinMode(ledPin, OUTPUT);
13   // Δείχνουμε ότι στη θέση buttonPin βρίσκεται μια είσοδος (Button)
14   pinMode(buttonPin, INPUT);
15 }
16
17 void loop() {
18   // Διαβάζουμε την κατάσταση του button
19   buttonState = digitalRead(buttonPin);
20
21   // Όταν το κουμπί πατιέται ανάβει το λαμπάκι (Μένει αναμένο για όση διάρκεια πατάμε το κουμπί)
22
23   if (buttonState == HIGH) {
24     // Ανάβει το λαμπάκι
25     digitalWrite(ledPin, HIGH);
26   }
27   else {
28     // Σβήνει το λαμπάκι
29     digitalWrite(ledPin, LOW);
30   }
31 }
```

Done Saving.

Global variables use 11 bytes (0%) of dynamic memory, leaving 8,181 bytes for local variables. Maximum is 8,192 bytes.

28 Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3

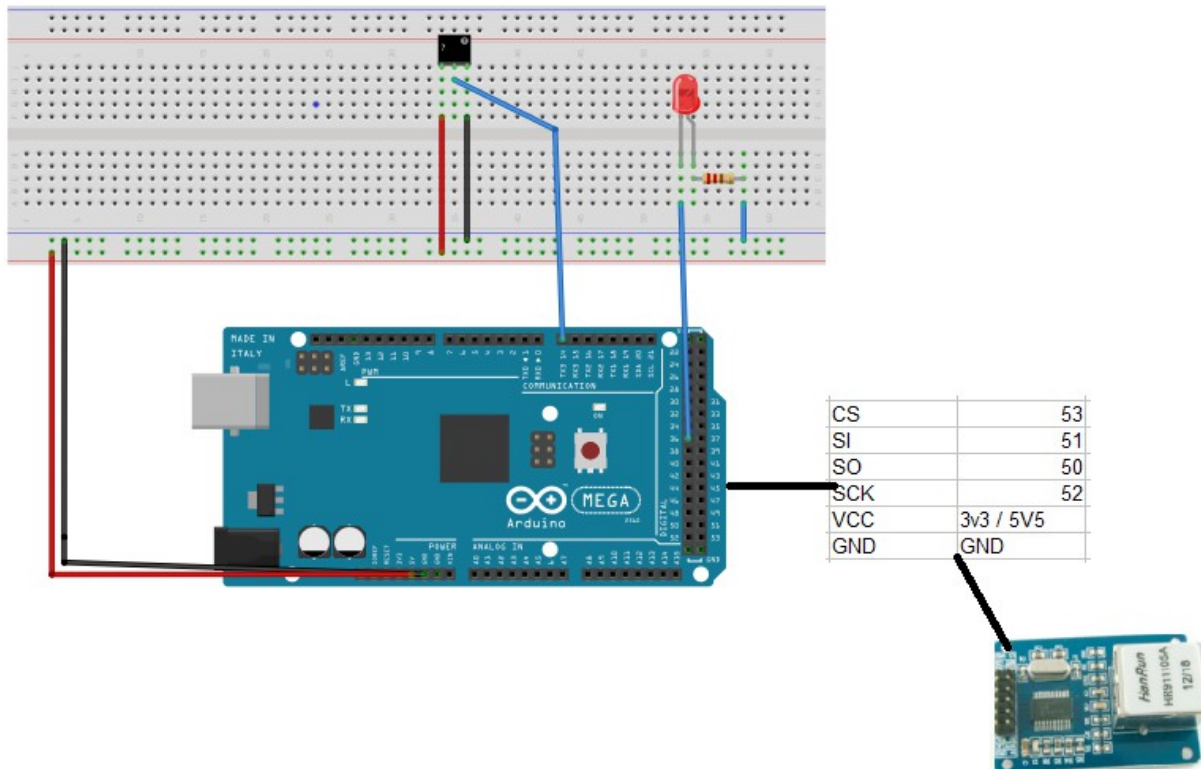
Κάνοντας Upload , θα δούμε ότι όσο διαρκεί το πάτημα του κουμπιού , το LED μένει αναμμένο !

Επεξήγηση κώδικα : Το καλύπτουν τα comments (//comment)

4.1.3 Παράδειγμα 3 : Άναμμα LED με τον εντοπισμό κίνησης και ενημέρωση βάσης δεδομένων

Σύνδεση

Συνδέουμε τον αισθητήρα κίνησης όπως παρακάτω . (πίσω όψη : VCC,PIN,GND) και την θύρα ethernet όπως στον πίνακα (Οι θύρες αυτές έχουν οριστεί από την βιβλιοθήκη που χρησιμοποιούμε)



Κώδικας

```
datawriteandblink | Arduino 1.6.1
File Edit Sketch Tools Help
datawriteandblink
1 #include <UIPEthernet.h> //Βιβλιοθήκη Ethernet
2 byte mac[] = { 0x54, 0x34, 0x41, 0x30, 0x30, 0x31 };
3 EthernetClient client; //Δηλώνουμε ότι το client είναι τύπου ethernet client
4
5 char server[] = "andreashostpapei.net"; // Διευθυνση IP ή ονομα server
6 int interval = 5000; // Άναμνη μετὰξυ επαναλήψεων
7 int calibrationTime = 30; // Η ώρα που χρειάζεται ο σενσορας για να "αντιληφθει" το περιβαλλον
8
9 int i;
10 int pirPin = 14; //Το PIN συνδεσης του σενσορα
11 int ledPin = 36; //Το PIN συνδεσης του LED
12 int γ;
13
14 void setup(){
15   Serial.begin(9600); //Ετοι ανοιγομαι τη συνδεση για οπτική επικοινωνία με το Arduino
16   Ethernet.begin(mac); //Εντολή ενεργης ethernet , δίνουμε μια mac address στ Arduino
17   Serial.print("IP Address : "); // Τυπώνουμε τα στοιχεία που παίρνει το Arduino
18   Serial.println(Ethernet.localIP());
19   Serial.print("Subnet Mask : ");
20   Serial.println(Ethernet.subnetMask());
21   Serial.print("Default Gateway IP: ");
22   Serial.println(Ethernet.gatewayIP());
23   Serial.print("DNS Server IP : ");
24   Serial.println(Ethernet.dnsServerIP());
25
26   Serial.print("Go Away You Have "); //Δίνουμε χρόνο στο χρήστη να απομακρυνθει
27   for(int i = 5; i > 0; i--){
28     Serial.print(i);
29     Serial.print(" ");
30     delay(1000);
31   }
32
33   pinMode(pirPin, INPUT); // Δήλωση εισόδου-εξοδου
34   pinMode(ledPin, OUTPUT);
35   digitalWrite(pirPin, LOW); //Δίνουμε τιμή LOW στον ελεγκτή κίνησης για να ξεκινήσει η μέτρηση
36
37   Serial.println("calibrating sensor for 30 sec"); //Δίνουμε χρόνο στον σενσορα να κάνει τις μετρήσεις
38   for(int i = calibrationTime; i >0; i--){
39     Serial.print(i);
40     Serial.print(".");
41     delay(1000);
42   }
43   Serial.println(" done");
44   Serial.println("SENSOR ACTIVE");
45   delay(50);
46
47
48
49
50
51
52
53
54
55
56
Done uploading.
Global variables use 1,517 bytes (18%) of dynamic memory, leaving 6,675 bytes for local variables. Maximum is 8,192 bytes.
Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3
```

```
datawriteandblink | Arduino 1.6.1
File Edit Sketch Tools Help
datawriteandblink
46 }
47
48 //Δημιουργούμε μια κλάση ενημέρωσης της βάσης
49 void addData(int y){
50 if (client.connect(server,80)) {
51   Serial.println("--> Connected");
52   client.print("GET /add_dataexample.php?");//επιλέγουμε το αρχείο που θέλουμε να τρέξουμε , αυτο στην ουσία θα κάνει την ενη
53   client.print("pass=XXXXXXXXXX");//Κωδικος
54   client.print("&&");
55   client.print("val=");//Τιμή μεταβλητης
56   client.print(y);
57
58   client.println( " HTTP/1.1");
59   client.print( "Host: " );
60   client.println(server);
61   client.println( "Connection: close" );
62   client.println();
63   client.println();
64   client.stop();
65 }
66 else {
67   Serial.println("--> connection failed/n");
68 }
69
70 delay(1000);
71 }
72
73 //LOOP
74 void loop(){
75
76
77   if(digitalRead(pirPin) == HIGH){
78     digitalWrite(ledPin, HIGH);
79
80     addData(4);// Τρέχουμε την κλάση addData με τιμή μεταβλητης γ = 4 ,λογο ενός προβλήματος στην σύδεση το τρεχουμε X2
81     addData(4);
82     delay(5000);
83   }
84
85   if(digitalRead(pirPin) == LOW){
86     digitalWrite(ledPin, LOW);
87     delay(2000);
88   }
89 }
90
58 Arduino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3
```

Επεξήγηση κώδικα :

Κλάση addData : Στην κλάση αυτή χρησιμοποιούμε τον ethernetclient “client” που έχουμε δηλώσει αρχικά για να εκτελέσουμε μια απλή εντολή GET .

GET /add_dataexample.php?pass=YOURPASSWORD&&val=y

Η εντολή GET πηγαίνει στα public files της IP/name που έχουμε δώσει και εκτελεί το αρχείο add-dataexample.php με μεταβλητή y (η y δηλώνεται κατά την εκτέλεση της κλάσης και δίνει την τιμή της στο **val**). Παρακάτω θα εξηγήσουμε πως λειτουργεί το php αρχείο μας.

```
<?php
```

```
$servername = "localhost"; // Localhost καθώς το αρχείο βρίσκεται στον server
```

```
$username = "yourusername";
```

```
$dbname = "yourdbname";
```

```
$conn = new mysqli($servername,$username, $_GET["pass"], $dbname); // Δημιουργία σύνδεσης
```

```
date_default_timezone_set("Europe/Athens"); //Ορίζουμε την timezone μας
```

```
$dt = date('YmdHis'); //Δίνουμε στην μεταβλητή dt την τιμή της Ημ/νίας με την μορφή
```

```
(EEEE,MM,Μερα,Ω,Λ,Δ)
```

```
$sql = "INSERT INTO example (Move,Date) //Απλή εντολή SQL για εισαγωγή στις στήλες
```

```
//Move,Date
```

```
VALUES ("$_GET["val"].", $dt)";
```

```
//Τις μεταβλητές Val (την παίρνουμε απ την Get εντολή)
```

```
//και dt που ορίσαμε παραπάνω
```

```
if ($conn->query($sql) === TRUE) {
```

```
//Ελεγχος επιτυχούς προσθήκης για εμφάνιση σε Client
```

```
    echo "New record created successfully";
```

```
} else {
```

```
    echo "Error: " . $sql . "<br>" . $conn->error;
```

```
}
```

```
$conn->close();
```

```
//Κλείσιμο σύνδεσης
```

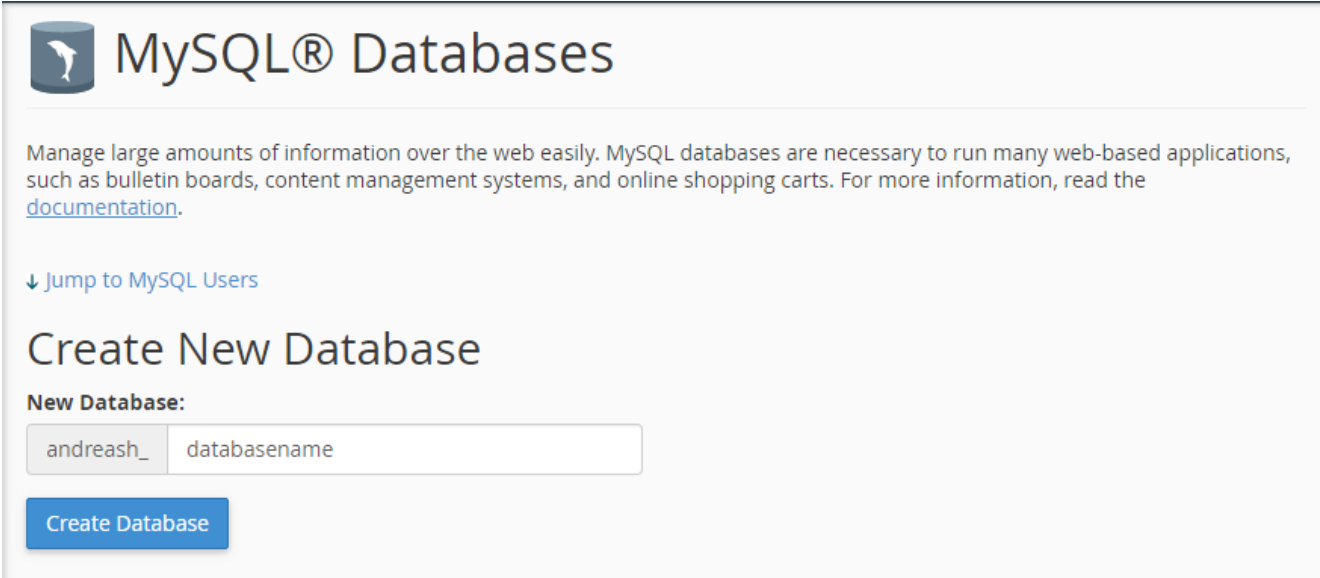
```
?>
```

4.2 Επεξήγηση στήσιμου βάσης/Server

Για να στήσουμε την βάση μας χρησιμοποιούμε τα εργαλεία που μας δίνει ο πάροχός μας . Κατά βάση το phpMyAdmin , το οποίο είναι ελεύθερο εργαλείο και μπορεί να χρησιμοποιηθεί για στήσιμο

Home Server . Θα δείξουμε με απλά βήματα πώς φτιάχνεται μία βάση .

Αρχικά πρέπει να δημιουργήσουμε την βάση μας. Στην περίπτωση μας επιλέγουμε την εφαρμογή MySQL Database , βάζουμε το όνομα που θέλουμε και πατάμε **create database** (Όπου databasename βάζουμε database).

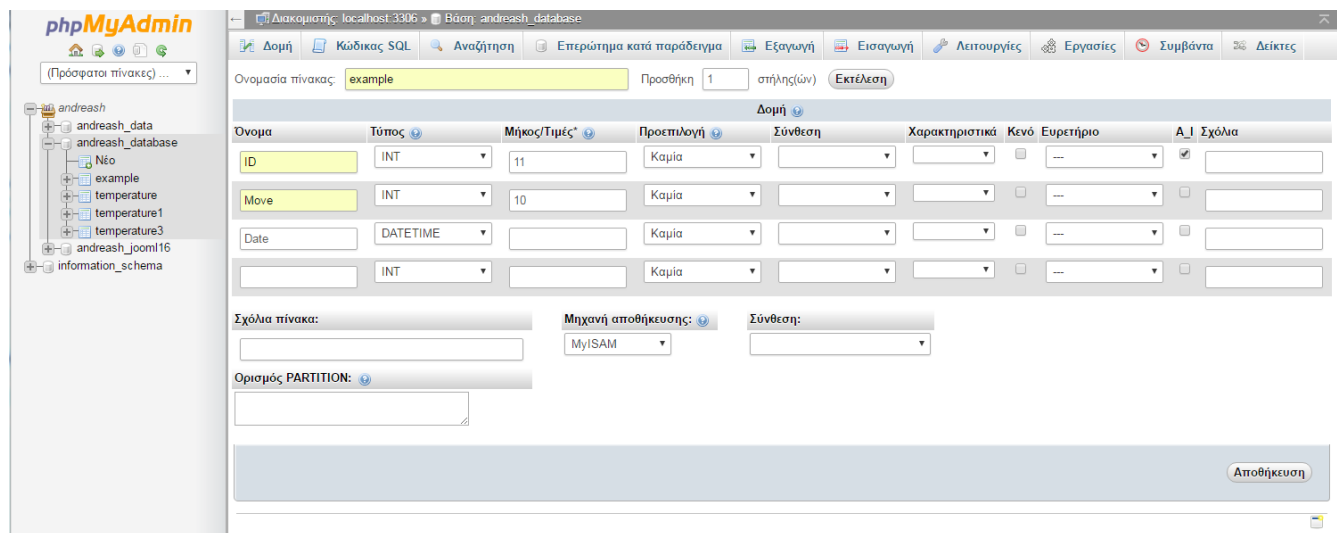


The screenshot shows the MySQL Databases management interface. At the top left is the MySQL logo, a blue circle with a white horse. To its right is the text 'MySQL® Databases'. Below this is a paragraph of text: 'Manage large amounts of information over the web easily. MySQL databases are necessary to run many web-based applications, such as bulletin boards, content management systems, and online shopping carts. For more information, read the [documentation](#).' Below the text is a blue link '↓ Jump to MySQL Users'. The main heading is 'Create New Database'. Underneath is the label 'New Database:' followed by a text input field containing 'andreash_' and 'databasename'. At the bottom left of the form is a blue button labeled 'Create Database'.

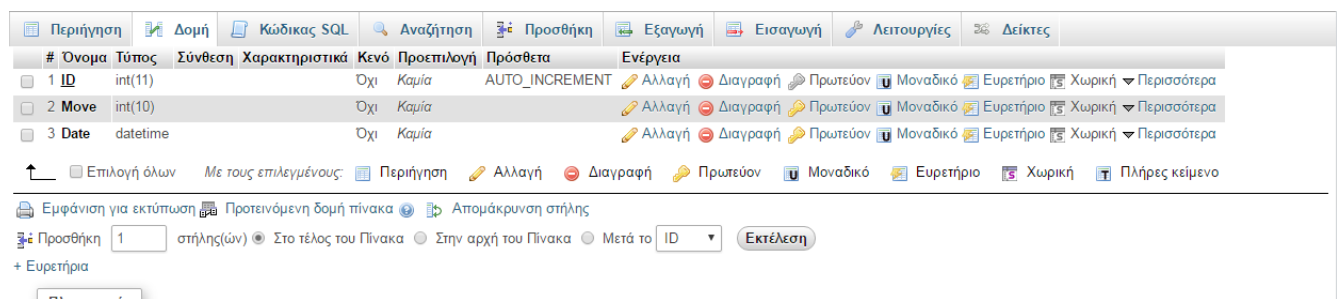
Έπειτα ανοίγουμε την εφαρμογή phpMyAdmin και βλέπουμε αριστερά την βάση μας .Εδώ φτιάχνουμε τους επιμέρους πίνακές μας, πατάμε στο Νέο και μας εμφανίζεται αυτή η σελίδα όπου βάζουμε τα στοιχεία του πίνακά μας. Εμείς θέλουμε έναν πίνακα example με 3 στήλες .

- Μία στήλη ID τύπου INT (ακέραιος) με μήκος (11) και την επιλογή A_I (auto increment) δηλαδή αυτόματα η βάση μας σε κάθε καταχώρηση αυξάνει αυτό τον αριθμό.
- Μία στήλη Move τύπου INT (ακέραιος) με μήκος (10) .
- Μία στήλη Date τύπου DateTime .Εδώ θα αποθηκεύεται η ακριβής ημερομηνία και ώρα της καταγραφής .

Και πατάμε **Εκτέλεση**



Τέλος πρέπει να ορίσουμε ένα κλειδί . Το κλειδί μπορεί να είναι μία στήλη ή ένα σύνολο στηλών . Πρέπει η κάθε καταγραφή να έχει ένα δικό της κλειδί . Δύο Καταχωρίσεις δεν μπορούν να έχουν το ίδιο κλειδί οπότε αν σε κάποια στήλη μας θα χρησιμοποιούνται ίδια στοιχεία, καλό είναι να επιλέγουμε ένα σύνολο στηλών που δεν μπορεί να είναι όλες ίδιες κάθε φορά. Εμείς εδώ έχουμε επιλέξει ως κλειδί την ID (πατάμε πάνω στο **Πρωτεύον**) αφού κάθε καταχώρηση παίρνει το δικό της ID.

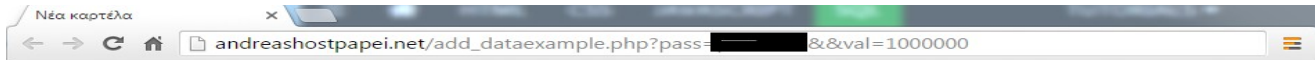


Η βάση μας έχει ολοκληρωθεί ! Μπορούμε τώρα να κάνουμε καταχωρήσεις .

4.3 Επεξήγηση κώδικα διασύνδεσης Arduino-Βασης-Interface

Στο Κεφάλαιο 4.1 (Παράδειγμα 3) είπαμε πως για να κάνουμε μία καταχώρηση εκτελούμε την εντολή `andreashostpapei.net/add_dataexample.php?pass=Password&&val=1000000` σε έναν Browser .Εδώ θα δείξουμε τον κώδικα που κρύβεται σε αυτό το αρχείο (`add_dataexample.php`).

Όταν τρέξουμε την εντολή σε έναν Browser



θα δούμε ότι θα υπάρξει καταχώρηση στην βάση μας .

The screenshot shows a database management interface. On the left, a tree view displays the database structure: `andreash` (server) contains a `[redacted].database`, which includes a `Νέο` (New) button, an `example` table, and an `information_schema` table. The `example` table has a `Ευρετήρια` (Index) section with a `Νέο` button and a `PRIMARY` key, and a `Στήλες` (Columns) section with `temperature`, `temperature1`, and `temperature3`.

The main area shows a SQL query: `SELECT * FROM `example` LIMIT 0, 30`. Below the query, there are options for `Δημιουργία προφίλ` (Profile creation), `Εσωτερικό` (Internal), `Επεξεργασία` (Edit), `Ανάλυση SQL` (SQL analysis), `Δημιουργία κώδικα PHP` (Generate PHP code), and `Ανανέωση` (Refresh).

Below the query, there are input fields for `Εμφάνιση` (Display): `Εγγραφή έναρξης` (Start record) set to 0, `Αριθμός εγγραφών` (Number of records) set to 30, and `Κεφαλίδες κάθε` (Rows per page) set to 100.

The results table shows one record:

ID	Move	Date
20	1000000	2016-04-08 13:10:57

Below the table, there are options for `Επιλογές` (Options): `Επιλογή όλων` (Select all), `Με τους επιλεγμένους` (With selected), `Αλλαγή` (Change), `Διαγραφή` (Delete), and `Εξαγωγή` (Export).

Κώδικας PHP

```
<?php //Έτσι αρχίζει ένα αρχείο PHP
$servername = "localhost"; //Όνομα server
$username = "UsernameFake"; //UserName
$dbname = "My_database_name"; //UserDatabaseName

$conn = new mysqli($servername,$username, $_GET["pass"], $dbname); // Δημιουργία Σύνδεσης

//$_GET["pass"] με αυτή την εντολή το αρχείο μας ψάχνει στην εντολή με την οποία εκτελέστηκε
//andreashostpapei.net/add_dataexample.php?pass=Password&&val=1000000
//το pass=something και χρησιμοποιεί την τιμή της για password στην σύνδεση .

date_default_timezone_set("Europe/Athens"); //Ορίζουμε την timezone μας καθώς
//server μας έχει διαφορετική default

$dt = date('YmdHis');
// Ορίζουμε την μεταβλητή dt και της δίνουμε την μορφή (Έτος,
//Μήνας,Μέρα,Ωρα,Λεπτό,Δευτερόλεπτο) καθώς έτσι αποθηκεύεται στη βάση μας.

//εντολή sql για καταχώρηση σε βάση
//"INSERT INTO όνομα_πίνακα (Όνομα_Στήλης1,Όνομα_Στήλης2 ...)
// VALUES(Πρώτο, Δεύτερο κλπ)"

$sql = "INSERT INTO example (Move,Date) VALUES ('".$_GET["val"]."', $dt)";
//Το αρχείο μας ψάχνει στην εντολή που το κάλεσε για το Val και δίνει την
//τιμή του στο Move .

if ($conn->query($sql) === TRUE) { //Έλεγχος επιτυχούς καταχώρησης
    echo "New record created successfully";
} else {
    echo "Error: " . $sql . "<br>" . $conn->error;
}
$conn->close();
?>
```

5 Σενάρια Χρήσης

Πρόλογος

Στο σύστημά μας υπάρχουν κάποιοι αισθητήρες και κάποιες συσκευές (ΚΕΦ3). Ανάλογα με τις μετρήσεις των αισθητήρων εκτελούνται ανάλογες ενέργειες. Στα σενάρια χρήσης φαίνονται οι ενέργειες που εκτελούνται και είναι ο πιο απλός τρόπος περιγραφής όλης της διαδικασίας

5.1 Ψευδοκώδικας

Σε αυτό το κεφάλαιο θα αναπτύξουμε σε ψευδοκώδικα το λειτουργικό κομμάτι του κώδικά μας , δηλαδή το κομμάτι κώδικα που τρέχει συνεχώς κ παίρνει τις αποφάσεις για τις ενέργειες του συστήματός μας.

5.1.1 Arduino-Ψευδοκώδικας

```
#  
Φόρτωση Βιβλιοθηκών  
#  
Δήλωση μεταβλητών  
#  
Δημιουργία κλάσεων  
#  
Έναρξη Επαναλαμβανόμενου προγράμματος
```

//Έλεγχο Απομακρυσμένης Εντολής

```
Διάβασε Βάση  
Αν Κ=7  
  Άναψε Led3 {  
  Ενημέρωσε Βάση}  
Αν Κ=8 {  
  Σβήσε Led3}
```

//Έλεγχος Τηλεφώνου

```
Διάβασε Βάση  
Τηλέφωνο καταχώρησε αριθμός
```


Av safe ON {

//Πρώτος αισθητήρας κίνησης (Μέσα)

Av κίνηση μέσα = Αλήθεια {
Αύξησε μετρητή κίνησης
Ενημέρωσε Βάση

Άναψε τα φώτα 1,2,3
Εκτέλεσε την κλάση : Ηχείο
Εκτέλεσε την κλάση :Αποστολή mail στον ένοικο
Εκτέλεσε την κλάση :Αποστολή SMS στον ένοικο
Av κατάσταση φώτων = 0 {
 Σβήσε τα φώτα}
}

//Δεύτερος αισθητήρας κίνησης (Έξω)

Av κίνηση έξω= Αλήθεια {
Αύξησε μετρητή κίνησης
Ενημέρωσε βάση
Άναψε τα φώτα 1
Εκτέλεσε την κλάση : Ηχείο
Av κατάσταση φώτων = 0 {
 σβήσε τα φώτα}
}

//Πληκτρολόγιο

Αν πατηθεί κουμπί {

Αν κουμπί = C {

Αύξησε Μετρητή Πατήματος κουμπιού

Κλείσε την πόρτα

Ενημέρωσε Βάση }

Αν κουμπί = D {

Εκτέλεσε την κλάση : Διαγραφή Βάσης Δεδομένων }

Αν κουμπί = A {

Περίμενε εισόδους

Δημιουργία κωδικού (P1,P2,P3,P4)

Αν πατηθεί κουμπί ανέθεσε την τιμή του στο P1 {

Αν πατηθεί κουμπί ανέθεσε την τιμή του στο P2 {

Αν πατηθεί κουμπί ανέθεσε την τιμή του στο P3 {

Αν πατηθεί κουμπί ανέθεσε την τιμή του στο P4 {

Απόδοση Κωδικός = (P1,P2,P3,P4) }

}

}

}

Αν κουμπί = B {

Αν Κωδικός(P1,P2,P3,P4) = Κωδικό (Δεδομένος){

Αύξησε μετρητή Πληκτρολογίου

Ενημέρωσε Βάση Πληκτρολογίου

Άνοιξε Πόρτα

Ενημέρωση Βάση Πόρτα}

Αλλιώς (Λάθος Κωδικός) {

Αύξησε μετρητή Λάθος εισόδου

Ενημέρωσε Βάση Πληκτρολογίου

}

}

Αν κουμπί = A

Αν Κωδικός(P1,P2,P3,P4) = Κωδικό (Δεδομένος){

Ενεργοποίησε Control}

Αν κουμπί = C

Αν Κωδικός(P1,P2,P3,P4) = Κωδικό (Δεδομένος){

Απενεργοποίησε MoveIn και αυτόματα LED}

Αν κουμπί = D

Αν Κωδικός(P1,P2,P3,P4) = Κωδικό (Δεδομένος){

Ενεργοποίησε MoveIn και αυτόματα LED

SAFE ON }

}

//Αισθητήρας Φωτεινότητας

Διάβασε Αισθητήρα Φωτεινότητας

Αν Φωτεινότητα < Όριο και Κατάσταση Φώτων = 0 {

 Όρισε Κατάσταση Φώτων = 1

 Άναψε Φως 1,2

 Αύξησε μετρητή Φως 1

 Ενημέρωση Βάσης Φως 1

 Αύξησε μετρητή Φως 2

 Ενημέρωση Βάσης Φως 2

}

Αν Φωτεινότητα > Όριο και Κατάσταση Φώτων = 1 {

 Όρισε Κατάσταση Φώτων = 0

 Σβήσε Φώτα 1,2

}

//Αισθητήρας Θερμοκρασίας

Αν Ωρα – Ωρα Τελευταίας Μέτρησης > Όριο {

 Διάβασε Θερμοκρασία

 Μετατροπή Θερμοκρασίας

 Όρισε Ωρα Τελευταίας Μέτρησης = Ωρα

 Αν Απόλυτη Διαφορά Προσωρινής Θερμοκρασία – Τελευταία Θερμοκρασία

 > Όριο {

 Όρισε Ωρα Τελευταίας Μέτρησης = Ωρα

 Όρισε Τελευταία θερμοκρασία = Προσωρινή Θερμοκρασία

 Αύξησε Μετρητή Θερμοκρασίας

 Ενημέρωσε Βάση Θερμοκρασίας

 }

 Αν Προσωρινή Θερμοκρασία > Όριο και Απόλυτη Διαφορά Προσωρινής
 Θερμοκρασία – Τελευταία Θερμοκρασία > Όριο {

 Όρισε Ωρα Τελευταίας Μέτρησης = Ωρα

 Όρισε Τελευταία θερμοκρασία = Προσωρινή Θερμοκρασία

 Εκτέλεση Κλάσης Συναγερμός

 Ενημέρωση Βάσης Θερμοκρασίας

 }

 }

//Αισθητήρας Τιλτ (Τρέμουλο)

Διάβασε Τιλτ

Αν Τιλτ > Όριο {

 Ενημέρωσε Βάση Τιλτ

 Εκτέλεση Κλάσης Συναγερμός

 }

}

Av SafeOff {

//Τηλεκοντρόλ

Διάβασε Είσοδο Τηλεκοντρόλ

Αν Είσοδος = Αλήθεια {

Διάβασε είσοδο

Κουμπί καταχώρησε είσοδο

Περίπτωση (68): {Άνοιξε Πόρτα}

Περίπτωση (69): {Κλείσε Πόρτα}

Περίπτωση (21): {Αναψε LED3}

Περίπτωση (9): {Σβήσε LED3}

Περίπτωση (13): {Ενεργοποίησε Keypad }

}

}

5.1.2 Interface-Ψευδοκώδικας

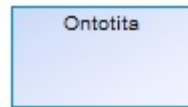
```
Εμφάνιση Σελίδας Log In
Εισαγωγή Ονόματος – Κωδικού
Αν Επιλογή Submit {
    Έλεγχος Ονόματος – Κωδικού}
Αν Όνομα – Κωδικός = Αλήθεια {
    Εμφάνιση Βασικής Σελίδας
    Αν Επιλογή Led 1 {
        Εμφάνιση Βάσης Led 1}
    Αν Επιλογή Led 2 {
        Εμφάνιση Βάσης Led 2}
    Αν Επιλογή Led 3 {
        Εμφάνιση Βάσης Led 3}
    Αν Επιλογή Door {
        Εμφάνιση Βάσης Door}
    Αν Επιλογή MoveIn {
        Εμφάνιση Βάσης MoveIn}
    Αν Επιλογή MoveOut {
        Εμφάνιση Βάσης MoveOut}
    Αν Επιλογή KPad {
        Εμφάνιση Βάσης KPad}
    Αν Επιλογή Tilt {
        Εμφάνιση Βάσης Tilt}
    Αν Επιλογή Temp {
        Εμφάνιση Βάσης Temp}
    Αν Επιλογή Apply {
        Έλεγε Κουμπί
        Ενημέρωση Βάσης Led Control}
    Αν Επιλογή Apply 2 {
        Διάβασε Τηλέφωνο
        Ενημέρωση Βάσης Number}}
```

5.2 Περιπτώσεις Χρήσης

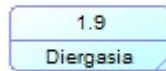
Διάγραμμα DFD

Το διάγραμμα DFD(Data Flow Datagramme) μας βοηθά να κατανοήσουμε την ροή των δεδομένων στο σύστημά μας . Είναι μία σχηματική επεξήγηση για το πώς κυλάνε τα δεδομένα μέσα στο σύστημα .Στα DFD έχουμε :

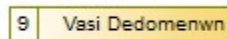
Εξωτερικές οντότητες : Όποια οντότητα ή συσκευή εισάγει ή δέχεται δεδομένα από το σύστημα μας .



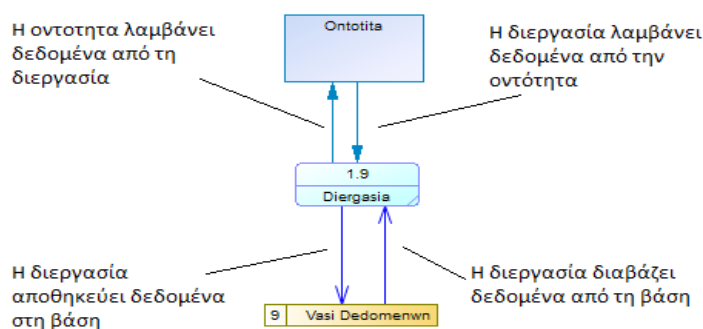
Διεργασίες : Διεργασίες ονομάζουμε όλες τις μεταβολές – χρήση που κάνει το πρόγραμμά μας στα δεδομένα που δέχεται.



Βάση Δεδομένων : Οι βάσεις δεδομένων του συστήματός μας .



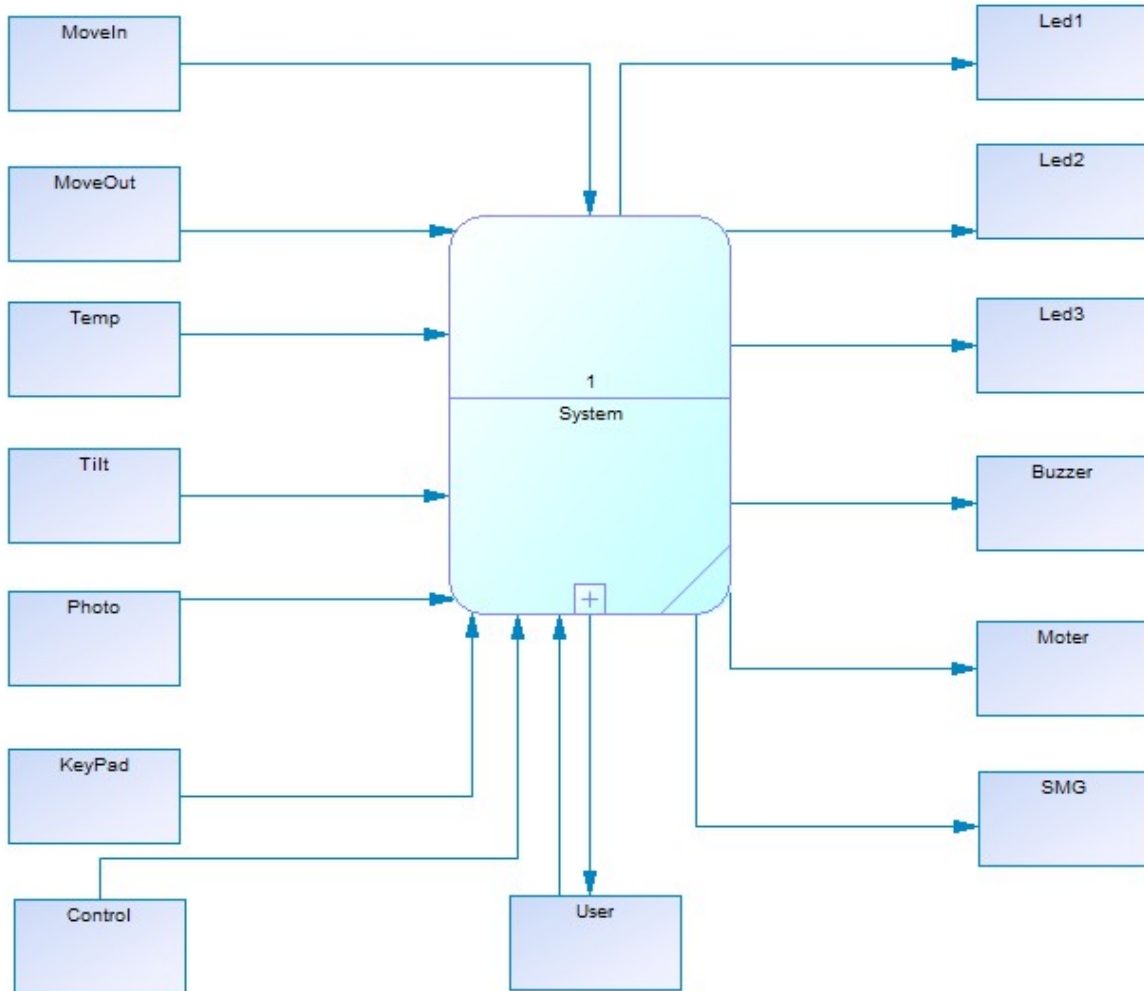
Με βελάκια ανάμεσα στις διεργασίες-οντότητες-βάσεις μας απεικονίζουμε την ροή των δεδομένων μας



Όσο προχωράμε στα επίπεδα (0,1,2...) αναλύουμε όλο και πιο βαθιά τι συμβαίνει στο σύστημά μας

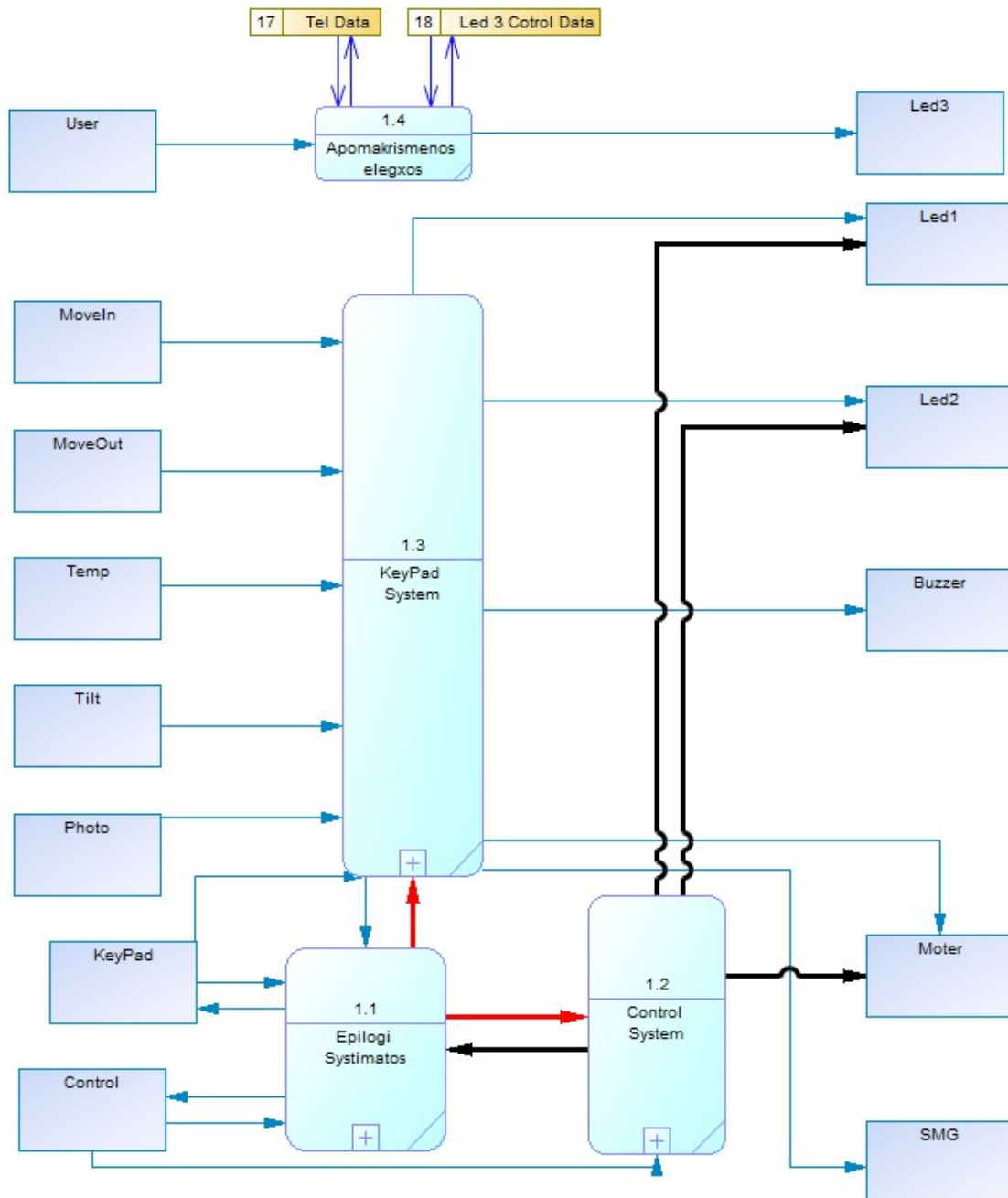
5.2.1 DFD: Επίπεδο 0

Σε αυτό το επίπεδο βλέπουμε όλους τους αισθητήρες που είναι συνδεδεμένοι με το πρόγραμμά μας καθώς επίσης και τις οντότητες και τις συσκευές που αλληλεπιδρούν με αυτό.



5.2.2 DFD:Επίπεδο 1

Σε αυτό το επίπεδο βλέπουμε πως λειτουργεί το πρόγραμμά μας γενικά . Γίνεται η επιλογή (1.1) για το αν θα λειτουργούμε το IRControl (1.2) ή το Keypad (1.3)και αναλόγως επιλέγονται οι διεργασίες που θα είναι ενεργές. Ακόμα βλέπουμε τον έλεγχο στοιχείων (1.4) που τρέχει ανά τακτά χρονικά διαστήματα ανεξάρτητων της λειτουργίας που έχουμε επιλέξει.

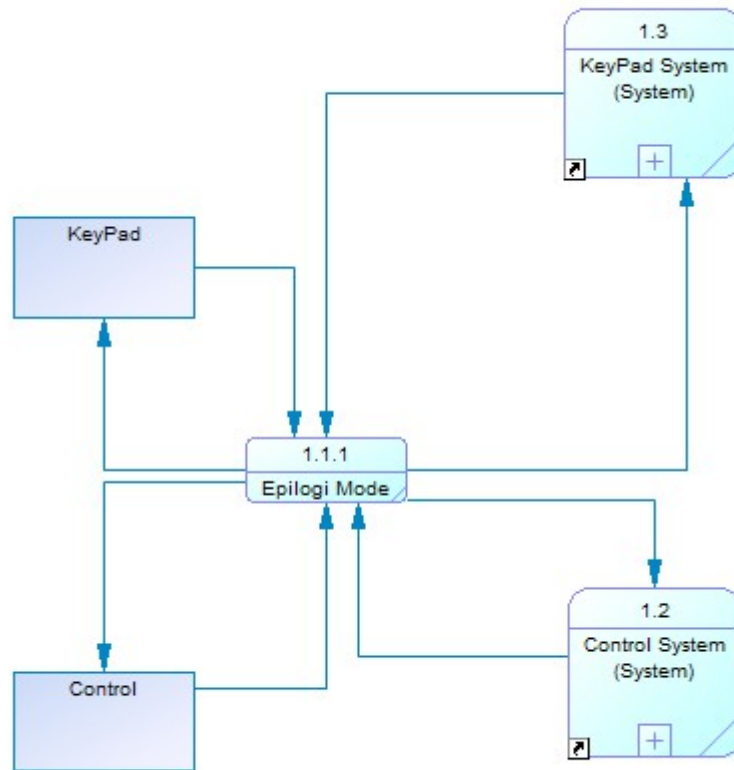


Το 1.4 τρέχει ανά τακτά χρονικά διαστήματα και ενημερώνει τη βάση για το τηλέφωνο που θα αποστέλλονται τα στοιχεία και ανάβει/σβήνει το φως ανάλογα την τιμή που βρίσκει στο Led Control.

5.2.2.1 DFD1.1

Επιλογή Συστήματος 1.1

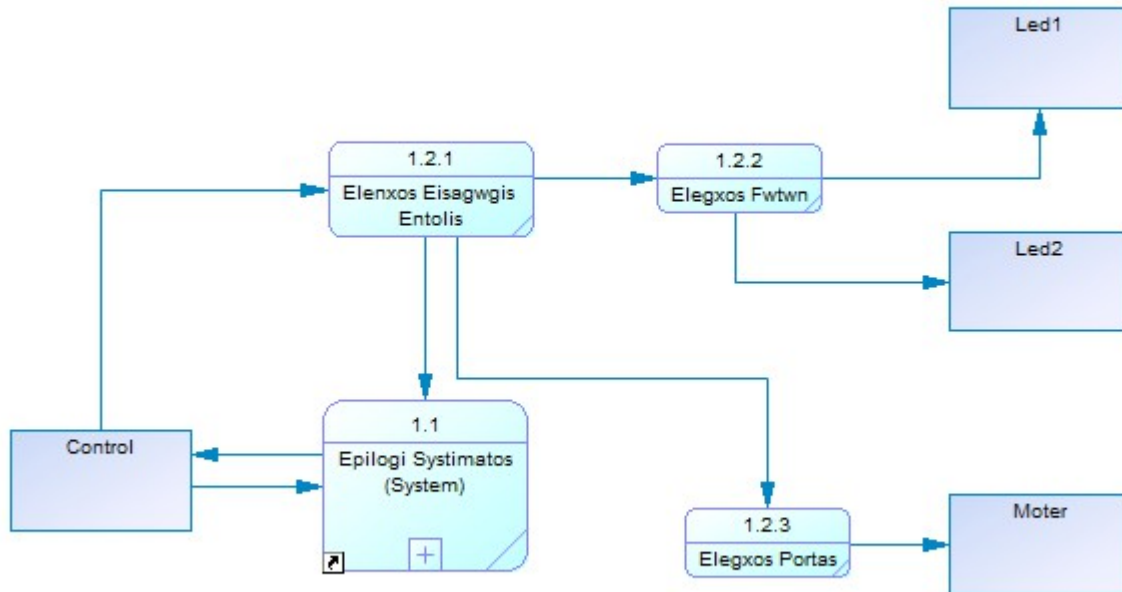
Όταν το σύστημά μας ξεκινά είναι ενεργό το Keypad , εμείς μπορούμε να επιλέξουμε να λειτουργεί το IRControl. Ανάλογα την επιλογή μας ενεργοποιούνται άλλες διεργασίες . Όταν είναι ενεργό το IRControl μπορούμε να ελέγξουμε τις συσκευές μας από κει . Όταν είναι ενεργό το Keypad μπορούμε να ελέγξουμε από εκεί τον συναγερμό και τις συσκευές .



5.2.2.2 DFD1.2

Control System 1.2

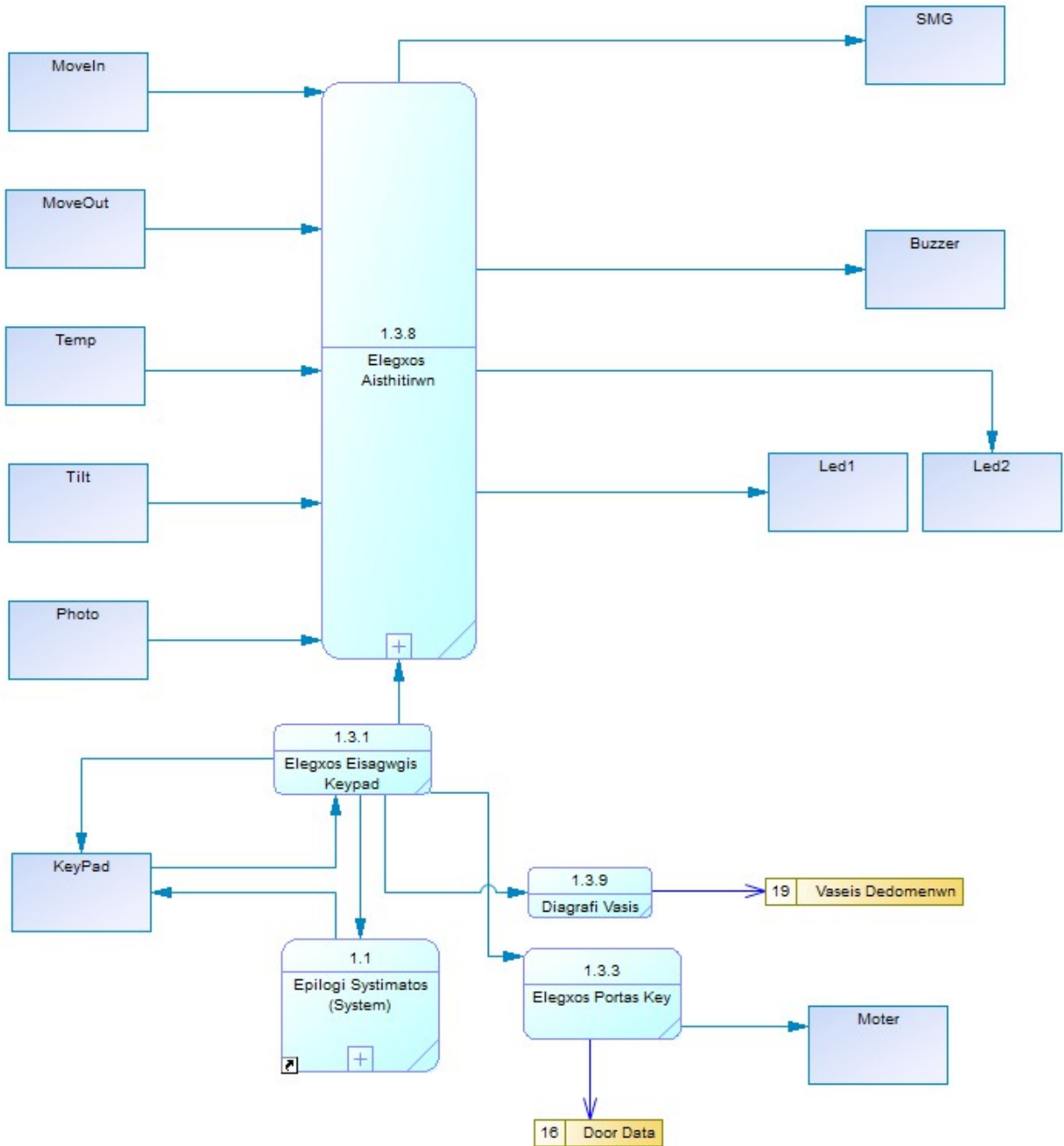
Όταν έχουμε ενεργό το IRControl μπορούμε να ανάψουμε/σβήσουμε τα φώτα , να ανοίξουμε/κλείσουμε την πόρτα ή να δώσουμε εντολή να λειτουργήσει το KeyPad .



5.2.2.3 DFD1.3

KeyPad System 1.3

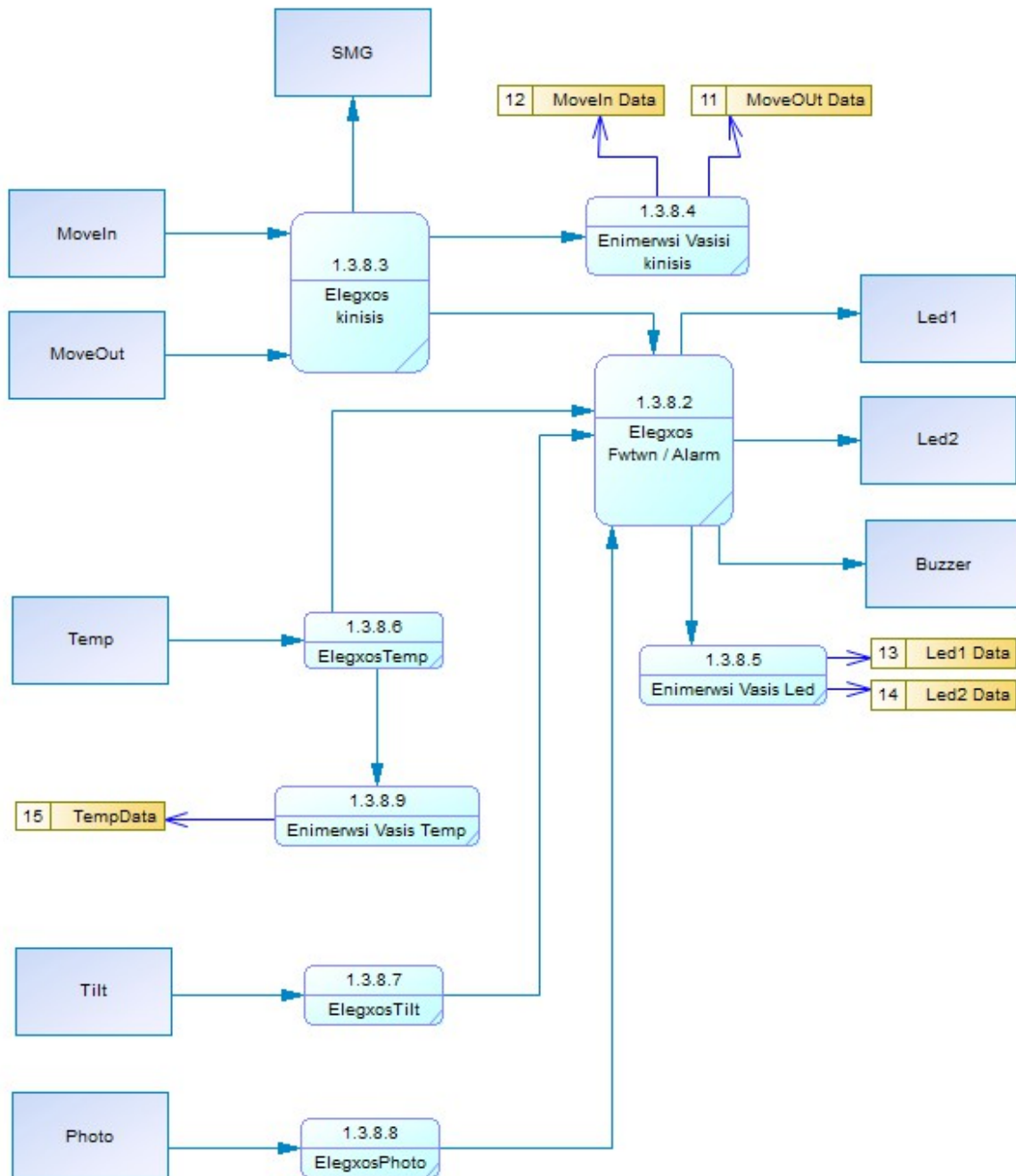
Από το KeyPad έχουμε τις επιλογές να ανοίξουμε/κλείσουμε την πόρτα (1.3.3) , να διαγράψουμε τη βάση (1.3.9) , να επιλέξουμε την λειτουργία IRControl (1.1) και να ενεργοποιήσουμε ή απενεργοποιήσουμε τον αισθητήρα MoveIn .



5.2.3 Επίπεδο 2

5.2.3.1 DFD 2.1

Διεργασία Ελεγχος Αισθητηρων 1.3.8



Όταν οι αισθητήρες μας είναι ενεργοί λειτουργεί ο συναγερμός μας . Δηλαδή :

Διεργασία 1.3.8.3 Έλεγχος κίνησης

Αν υπάρξει κίνηση έξω εκτελείται το Alarm μας (Φώτα και Ήχος) και ενημερώνεται η βάση

Αν ο MoveIn είναι ενεργός , αν υπάρξει κίνηση εκτελείται το Alarm και αποστέλλεται mail και sms στον ιδιοκτήτη , επίσης ενημερώνεται η βάση .

Διεργασία 1.3.8.6 Έλεγχος Θερμοκρασίας

Η θερμοκρασία ελέγχεται ανά τακτά χρονικά διαστήματα και ενημερώνεται η βάση . Σε περίπτωση απότομης αλλαγής ή πολύ μεγάλης θερμοκρασίας εκτελείται το Alarm και ενημερώνεται η βάση .

Διεργασία 1.3.8.8 Έλεγχος Τιλτ

Σε περίπτωση σεισμού εκτελείται το Alarm .

Διεργασία 1.3.8.8 Έλεγχος Φωτεινότητας

Σε περίπτωση που πέσει η φωτεινότητα στο χώρο (εξωτερικά) ανάβουν τα φώτα και ενημερώνεται η βάση .

Κεφάλαιο 6

Μελλοντικές Επεκτάσεις /Ρεαλιστική εφαρμογή

6.1 Μελλοντικές Επεκτάσεις

Σε αυτό το κεφάλαιο θα εξηγήσουμε πως θα μπορούσαν να εισαχθούν κάποιες επιπλέον επεκτάσεις στο πρόγραμμά μας . Με λίγα λόγια για το τι θα έκανε στην ουσία η καθεμιά τους και ένα ψευδοκώδικα για το πως .

- **IP Κάμερα**

Μια IP κάμερα θα μπορεί να λειτουργεί με 2 τρόπους , είτε αν ο πάροχος μας παρέχει χώρο σε server και δυνατότητα live οπότε θα έχουμε στην βασική μας σελίδα απλά ένα hyperlink ,είτε θα συνδέεται με το Arduino οπότε θα την χειριζόμαστε εμείς

IP Camera – Arduino

Αν κίνηση μέσα = Αλήθεια {

 Άναψε Φώτα

 Εκτέλεσε Φωτογραφία

 Σβήσε Φώτα

 Εκτέλεσε Φόρτωση Φωτογραφίας σε Μνήμη

 Ενημέρωση Βάσης Φωτογραφία

 Αποστολή mail με link φωτογραφίας }

IP Camera – Online

Στην αρχική μας σελίδα προσθέτουμε ένα κουμπί “LIVE” . Στον κώδικά μας στο LIVE έχουμε ένα hyperlink με την διεύθυνση που μας παρέχει ο πάροχος της κάμερας στην οποία υπάρχει αναπαραγωγή εικόνας από τον χώρο μας.

Αν LIVE= Αλήθεια {

 Εμφάνισε σελίδα Live Feed}

- **Μοτέρ Παράθυρα**

Μια εφαρμογή μοτέρ στα παράθυρα θα μπορούσε να λειτουργεί είτε μέσω του KeyPad για άμεσο έλεγχο από τον χρήστη είτε να ρυθμίσουμε το Arduino να τα ανοίγει το πρωί σαν μια “έξτρα” λειτουργία ξυπνητηριού.

Αν Πάτημα Κουμπιού = Αλήθεια {

Κουμπί καταχώρησε = Είσοδος // Καταχωρούμε στο κουμπί την όποια είσοδο

Διάβασε Βάση Παράθυρο

Αν Είσοδος = 9 και ΠαραθυροΑνοιχτο = 0 {

Άνοιξε παράθυρα // Θέτει σε λειτουργία το μοτέρ κάνοντας 10 κύκλους .

Ενημέρωσε Βάση Παράθυρο //ΠαράθυροΑνοιχτο = 1 }

Αν Είσοδος = 8 και ΠαράθυροΑνοιχτο = 1 {

Κλείσε παράθυρα // Θέτει σε λειτουργία το μοτέρ κάνοντας -10 κύκλους .

Ενημέρωσε Βάση Παράθυρο //ΠαράθυροΑνοιχτο = 0 }

}

Διάβασε Φωτεινότητα

Αν Φωτεινότητα > Όριο {

Διάβασε Βάση Παράθυρο

Αν ΠαράθυροΑνοιχτο = 0 {

Άνοιξε παράθυρα // Θέτει σε λειτουργία το μοτέρ κάνοντας 10
//κύκλους .

Ενημέρωσε Βάση Παράθυρο //ΠαράθυροΑνοιχτο = 1 }

}

6.2 Ρεαλιστική εφαρμογή

Σε αυτό το κομμάτι θα εξηγήσουμε πώς θα εφαρμοστεί το πρόγραμμα σε πραγματικές συνθήκες δηλαδή με λάμπες αντί για LED και με κάποιο σύστημα ήχου αντί για BUZZER. Όπως και παραπάνω θα τα αναπτύξουμε σε ψευδοκώδικα .

- **Relay για λάμπες-ηχοσύστημα**

Επειδή τα LED μας κάνουν διαφορά στην μακέτα αλλά στον πραγματικό κόσμο δεν θα φαινόταν , είτε θα πρέπει να πάρουμε κάποιες λάμπες που να μπορούν να συνδεθούν με το Arduino είτε μπορούμε να λειτουργήσουμε με Relay . Δηλαδή το Relay είναι συνδεδεμένο με την πρίζα και την λάμπα η οποία είναι πάντα αναμμένη και το Arduino ελέγχει το πότε το Relay θα επιτρέψει να περνάει το ρεύμα και πότε όχι . Το ίδιο και με ένα ηχοσύστημα ή τηλεόρασή για παραγωγή ήχου .

Στα σημεία του κώδικα που υπάρχει άναψε Led ή Buzzer

Αν συνθήκη {

 Άνοιξε Relay }

- **Μοτέρ Πόρτας**

Το μοτέρ που χρησιμοποιήσαμε για την πόρτα είναι για λόγους επίδειξης . Για μία κανονική πόρτα θα χρειαζόταν κάτι μεγαλύτερο (θα λειτουργούσε περίπου σαν τις αυτόματες πόρτες γκαράζ που όλοι γνωρίζουμε) . Θα υπάρχει αισθητήρας κίνησης στην πόρτα για αποφευχθούν ατυχήματα

Αν Κουμπί = A και PassCheck = Αλήθεια {

 Άνοιξε πόρτα // Το μοτέρ γυρνάει 10 φορές ώστε να ανοίξει η πόρτα

 M κατοχύρωσε 10

 Περίμενε 15000 // 10"

 Όσο M > 0 {

 Διάβασε MoveSensorDoor

 Αν MoveSensorDoor = Ψέμα {

 Κλείσε Πόρτα // Το μοτέρ γυρνάει 1 φορά

 M-- // M= M-1}

 }

 Αλλιώς Περίμενε 3000 // 3"

 }

Αν Κουμπί = B {

N κατοχύρωσε 10

 Όσο N > 0 {

 Διάβασε MoveSensorDoor

 Αν MoveSensorDoor = Ψέμα

 Κλείσε Πόρτα // Το μοτέρ γυρνάει 1 φορά

 N-- // N= N-1} }

Κεφάλαιο 7

Εγχειρίδιο χρήσης

7.1 Εντολές Keypad

Μέσω του Keypad ο χρήστης μπορεί να έχει μια αλληλεπίδραση με το σύστημα την οποία να αντιλαμβάνεται άμεσα . Θα μπορεί να ανοίξει , να κλείσει την πόρτα και να αδειάσει την βάση δεδομένων .



Κλείσιμο Πόρτας

Αν θέλουμε να κλείσουμε την πόρτα πατάμε το **C** .

Διαγραφή βάσης δεδομένων

Αν θέλουμε να διαγράψουμε τη βάση δεδομένων μας πατάμε το **D** . (Προτείνεται πάντα να γίνεται διαγραφή στην έναρξη του προγράμματος)

A → **Pass** → ?

Λειτουργία Control A

Αν θέλουμε να ελέγχουμε το σύστημα από το IRControl Πατάμε το **A** . Έπειτα εισάγουμε τον κωδικό μας . Τέλος πατάμε το **A** . Αν ο κωδικός μας είναι σωστός τότε θα ενεργοποιηθεί το IRControl και θα απενεργοποιηθούν οι αισθητήρες μας .

Άνοιγμα Πόρτας **B**

Αν θέλουμε να ανοίξουμε την Πόρτα Πατάμε το **A** . Έπειτα εισάγουμε τον κωδικό μας . Τέλος πατάμε το **B** . Αν ο κωδικός μας είναι σωστός τότε θα ανοίξει η πόρτα .

Συναγερμός Μέσα και Αυτόματα Led **C/D**

Πατάμε το **A** . Έπειτα εισάγουμε τον **κωδικό μας** . Τέλος πατάμε το

- **C** Αν ο κωδικός μας είναι σωστός τότε απενεργοποιείται ο συναγερμός
- **D** Αν ο κωδικός μας είναι σωστός τότε ενεργοποιείται ο συναγερμός

7.2 Εντολές IRControl



Άνοιγμα Πόρτας

Αν θέλουμε να ανοίξουμε την πόρτα πατάμε το



Κλείσιμο Πόρτας

Αν θέλουμε να κλείσουμε τη πόρτα πατάμε το



Άναμμα Led3

Αν θέλουμε να ανάψουμε το LED πατάμε το



Σβήσιμο Led3

Αν θέλουμε να σβήσουμε το LED πατάμε το



Λειτουργία Keypad

Αν θέλουμε να λειτουργεί το Keypad πατάμε το



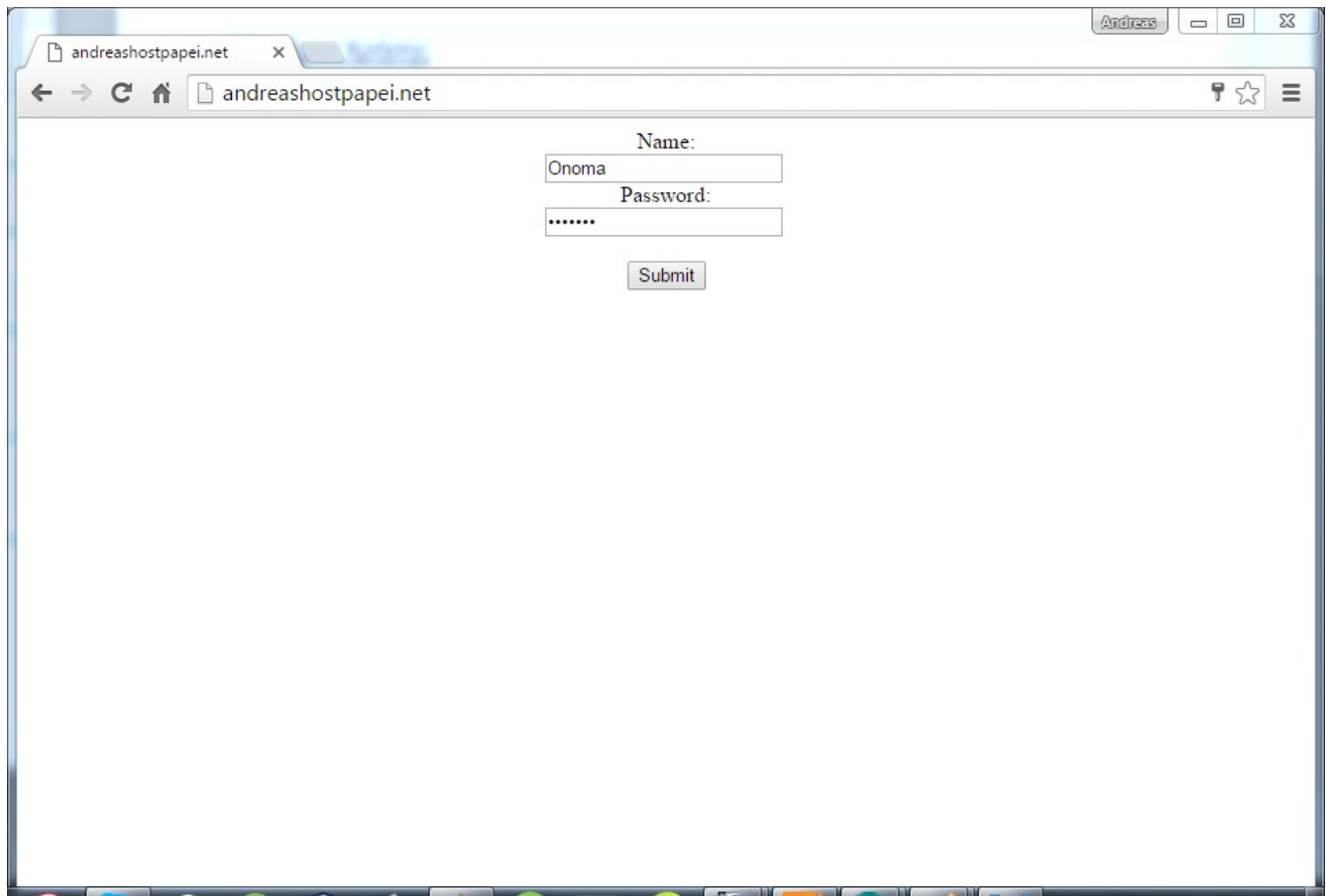
7.3 Επεξήγηση Interface

Το σύστημά μας προσφέρει στον χρήστη τη δυνατότητα να ελέγχει τι γίνεται στο σπίτι ανά πάσα στιγμή καθώς και να επεμβαίνει , ανάβοντας ή σβήνοντας ένα LED με το πάτημα ενός κουμπιού από το κινητό του ή τον υπολογιστή του .

Αρχικά ανοίγουμε έναν browser και πληκτρολογούμε την εξής διεύθυνση :

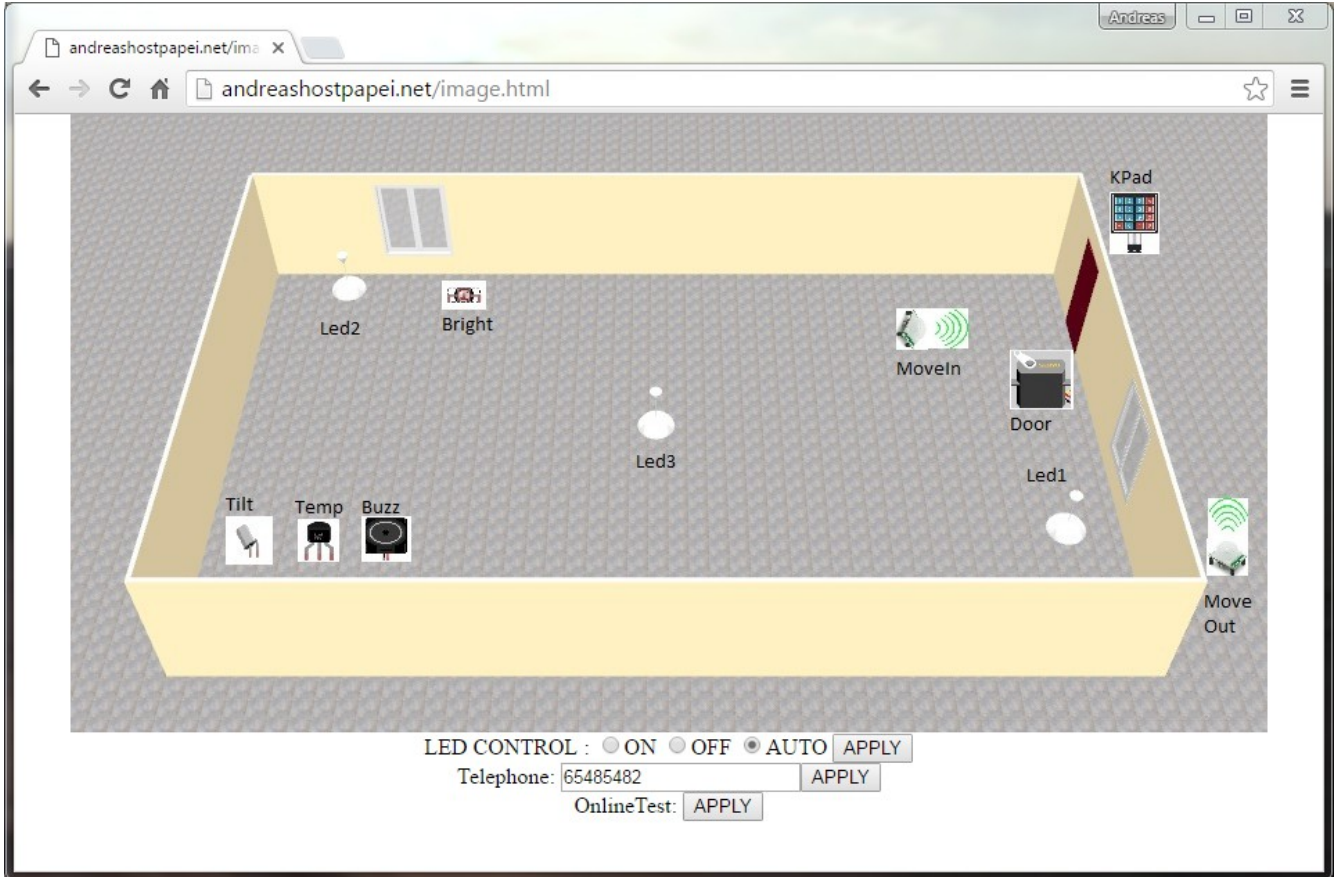
andreashostpapei.net

Μας εμφανίζεται η παρακάτω σελίδα



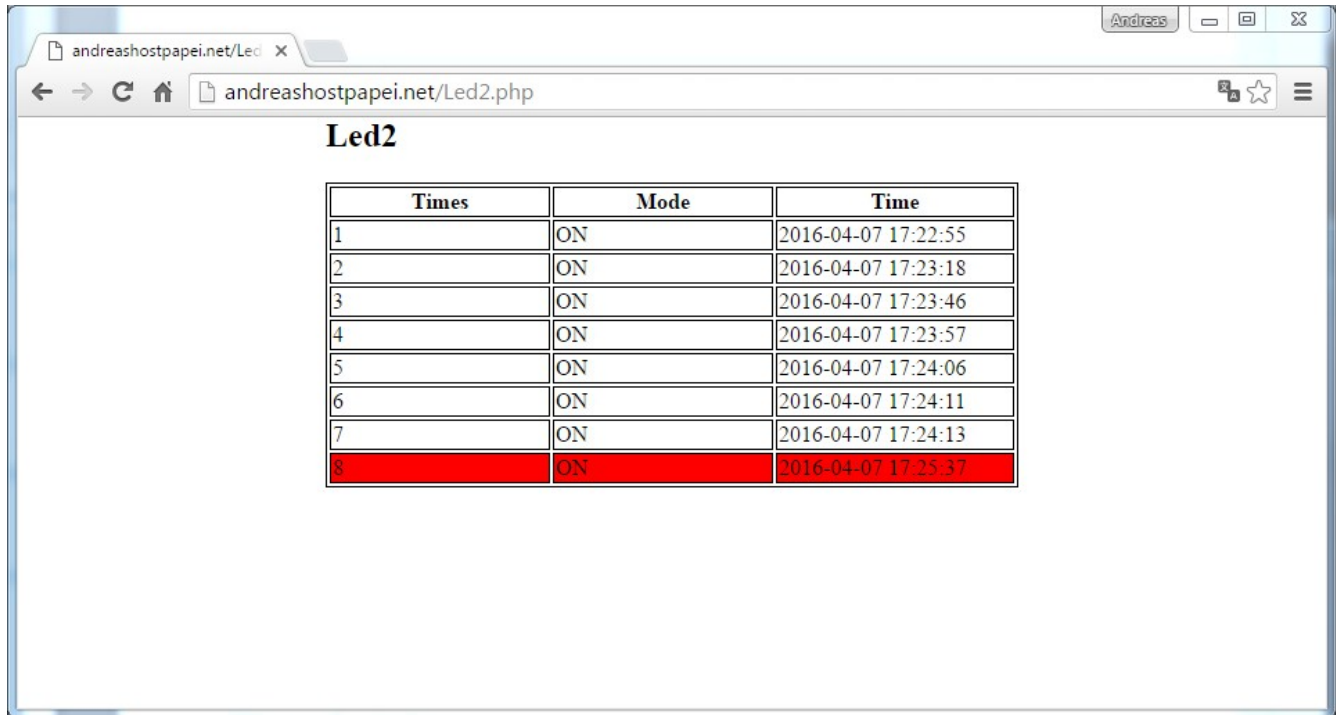
Για να κάνουμε Log In εισάγουμε τα στοιχεία που μας έχουν δοθεί και πατάμε το Submit .

Αν τα στοιχεία μας είναι σωστά τότε μας εμφανίζεται η βασική μας σελίδα. Η σελίδα μας είναι ένας “χάρτης”. Αυτό σημαίνει ότι μπορούμε να πατήσουμε πάνω σε όποιον αισθητήρα-συσσκευή θέλουμε (περισσότερους) και να δούμε τις εναλλαγές της κατάστασής τους.



Παραδείγματα Εμφάνισης Βάσεων Δεδομένων

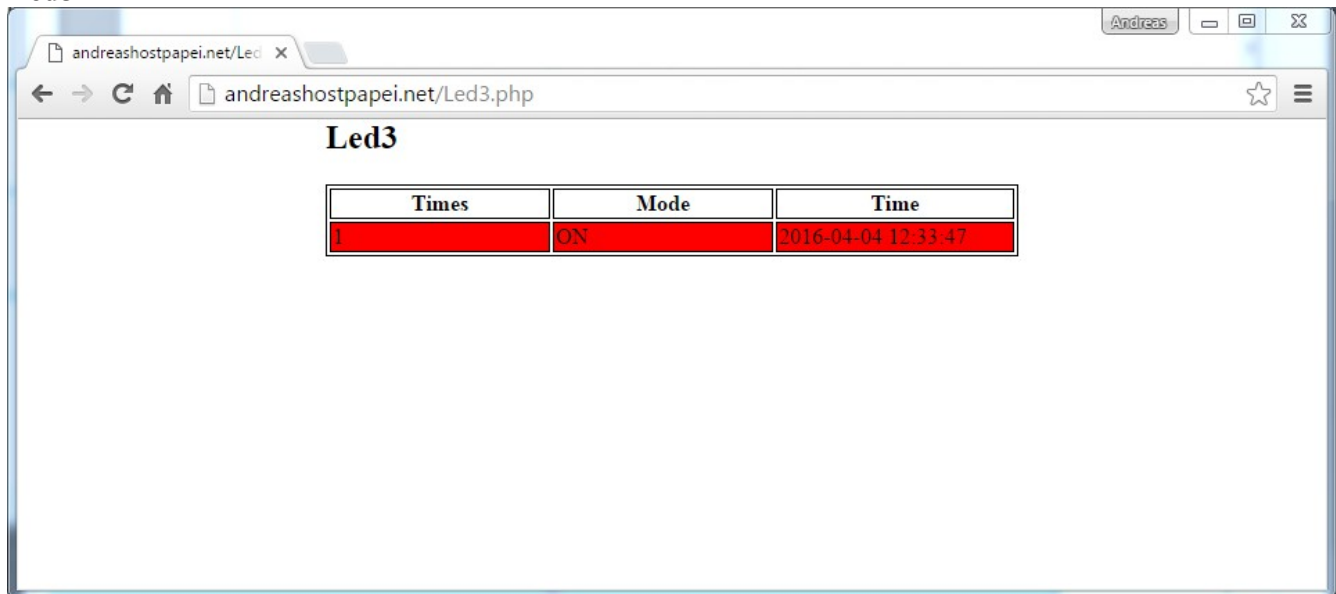
Led2



The screenshot shows a web browser window with the address bar containing 'andreashostpapei.net/Led2.php'. The page title is 'Led2'. Below the title is a table with three columns: 'Times', 'Mode', and 'Time'. The table contains 8 rows of data, with the last row (Times: 8) highlighted in red.

Times	Mode	Time
1	ON	2016-04-07 17:22:55
2	ON	2016-04-07 17:23:18
3	ON	2016-04-07 17:23:46
4	ON	2016-04-07 17:23:57
5	ON	2016-04-07 17:24:06
6	ON	2016-04-07 17:24:11
7	ON	2016-04-07 17:24:13
8	ON	2016-04-07 17:25:37

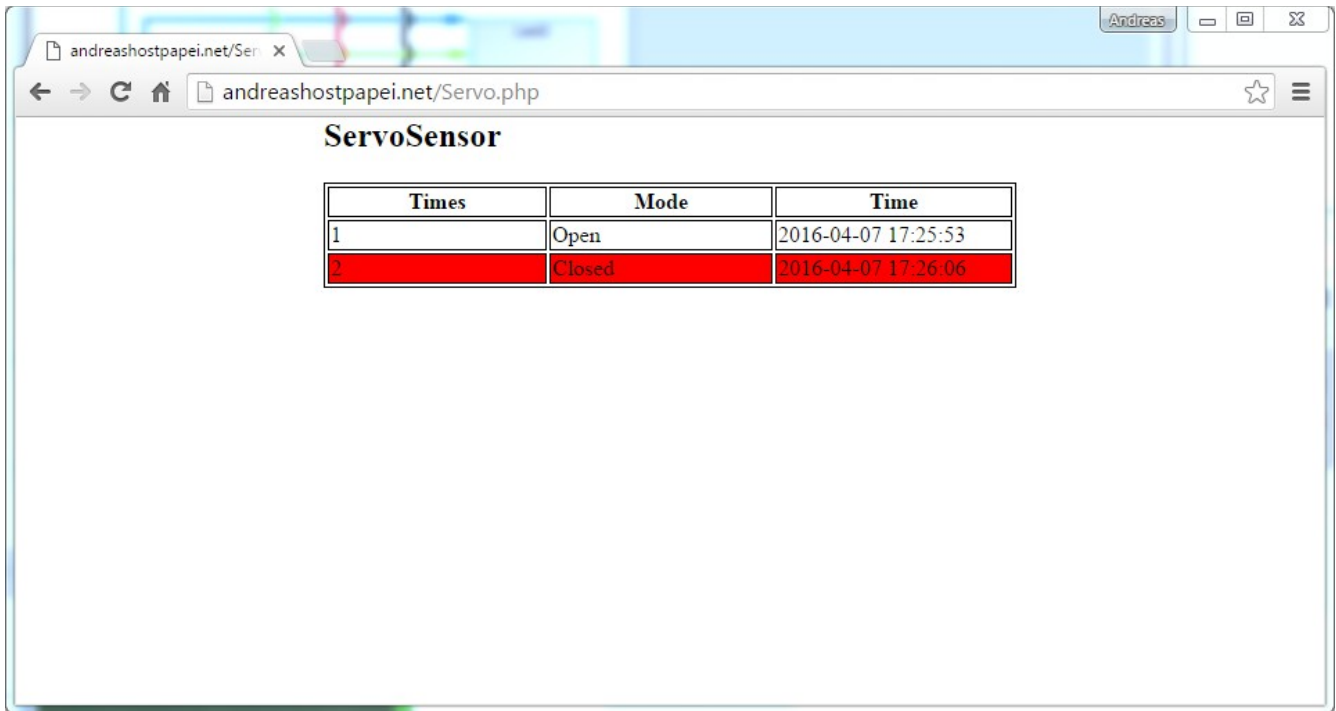
Led3



The screenshot shows a web browser window with the address bar containing 'andreashostpapei.net/Led3.php'. The page title is 'Led3'. Below the title is a table with three columns: 'Times', 'Mode', and 'Time'. The table contains 1 row of data, which is highlighted in red.

Times	Mode	Time
1	ON	2016-04-04 12:33:47

Door



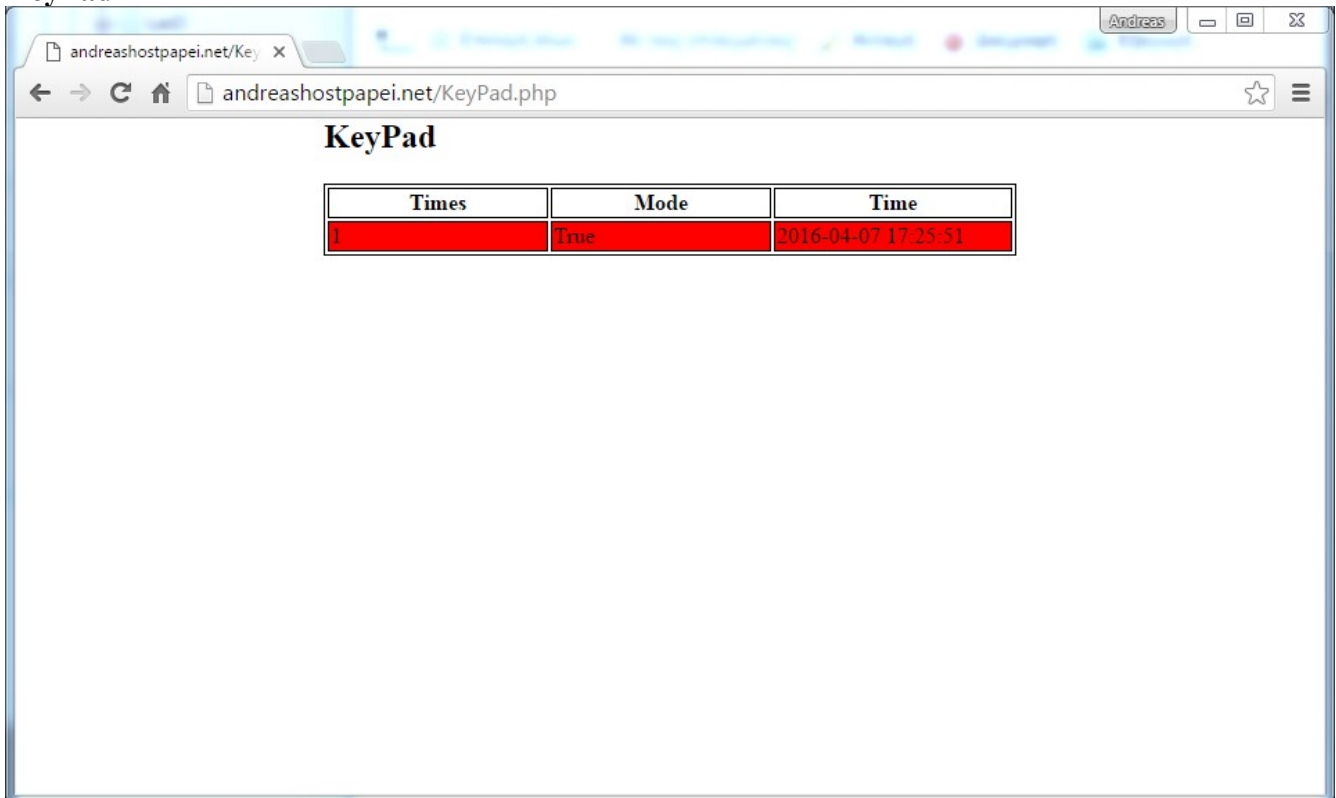
andreashostpapei.net/Ser x

andreashostpapei.net/Servo.php

ServoSensor

Times	Mode	Time
1	Open	2016-04-07 17:25:53
2	Closed	2016-04-07 17:26:06

KeyPad



andreashostpapei.net/Key x

andreashostpapei.net/KeyPad.php

KeyPad

Times	Mode	Time
1	True	2016-04-07 17:25:51

Temp

andreashostpapei.net/Ten x

andreashostpapei.net/Temp.php

TempSensor

Times	Temperature	Fire	Time
1	92		2016-04-07 17:19:57
2	-66		2016-04-07 17:19:58
3	103		2016-04-07 17:20:03
4	-124		2016-04-07 17:26:25
5	102		2016-04-07 17:26:39

Tilt

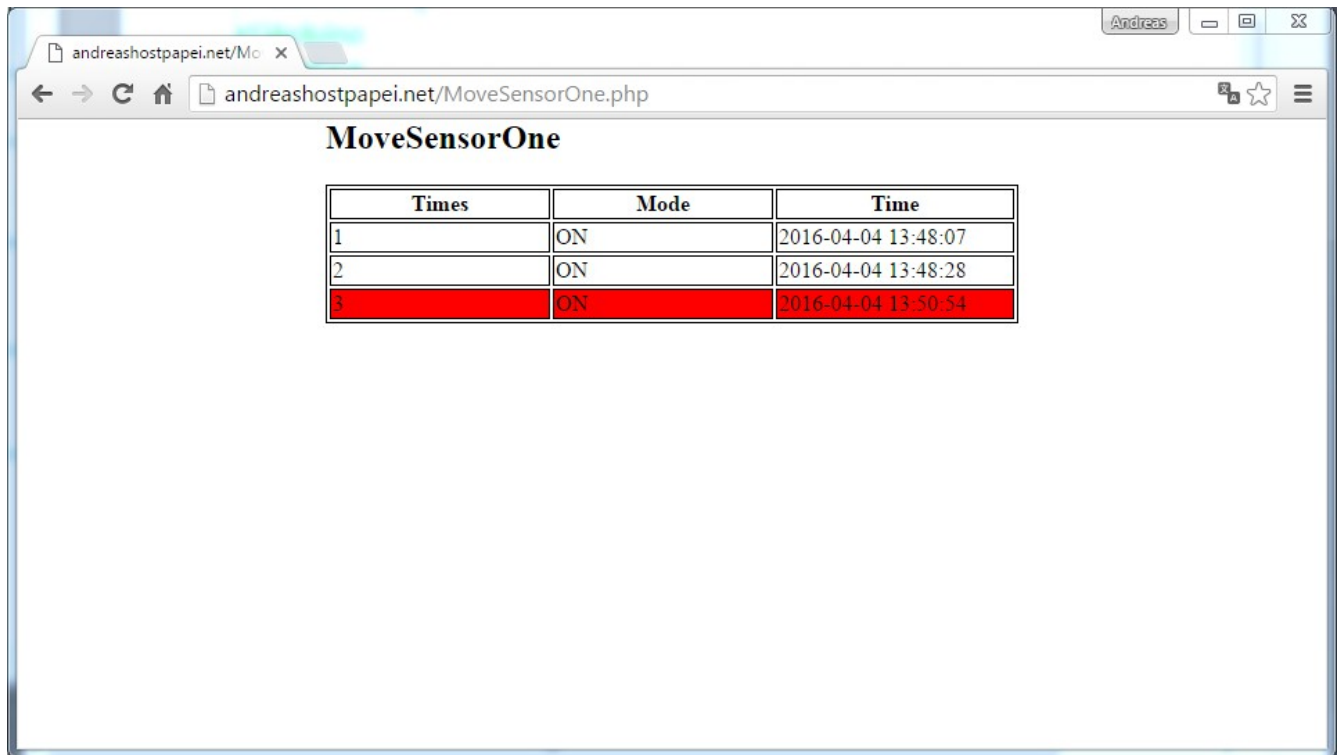
andreashostpapei.net/Tilt x

andreashostpapei.net/Tilt.php

TiltSensor

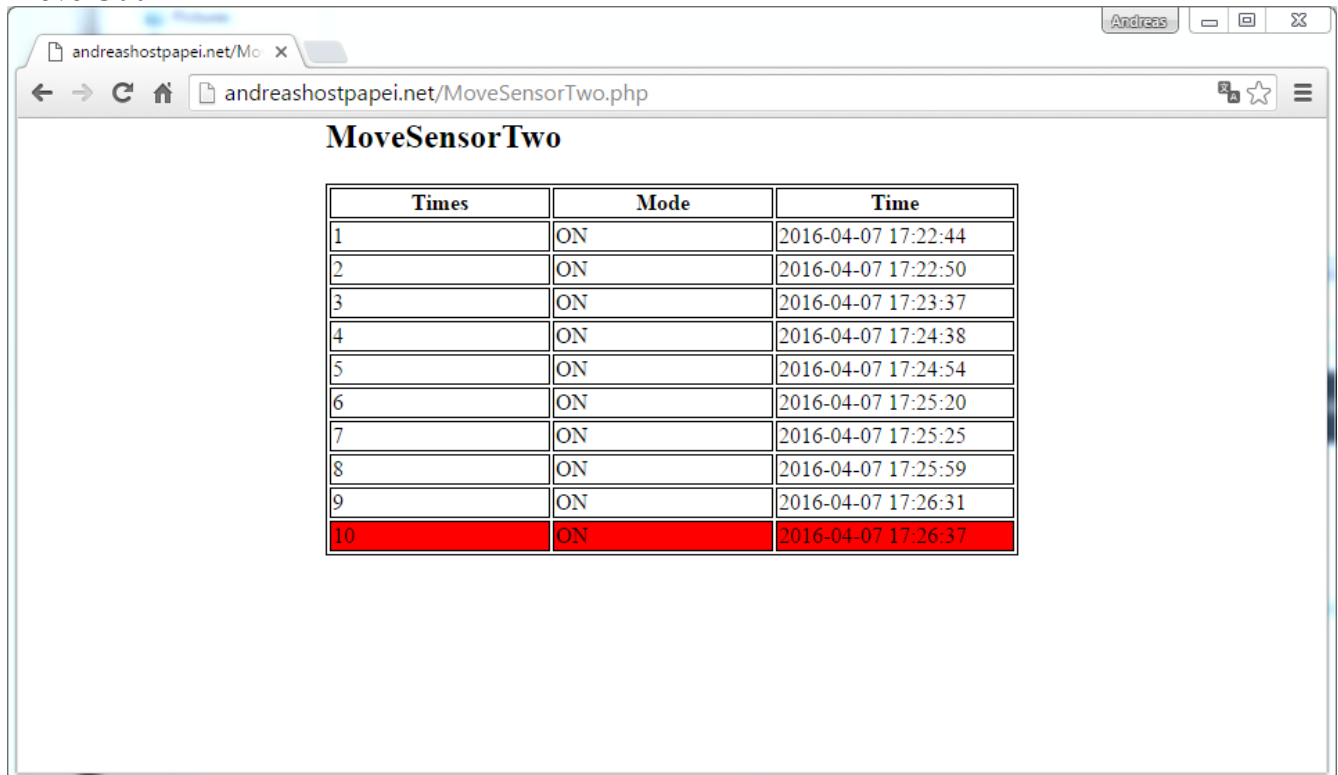
Times	Time
1	2016-04-07 17:26:21

MoveIn



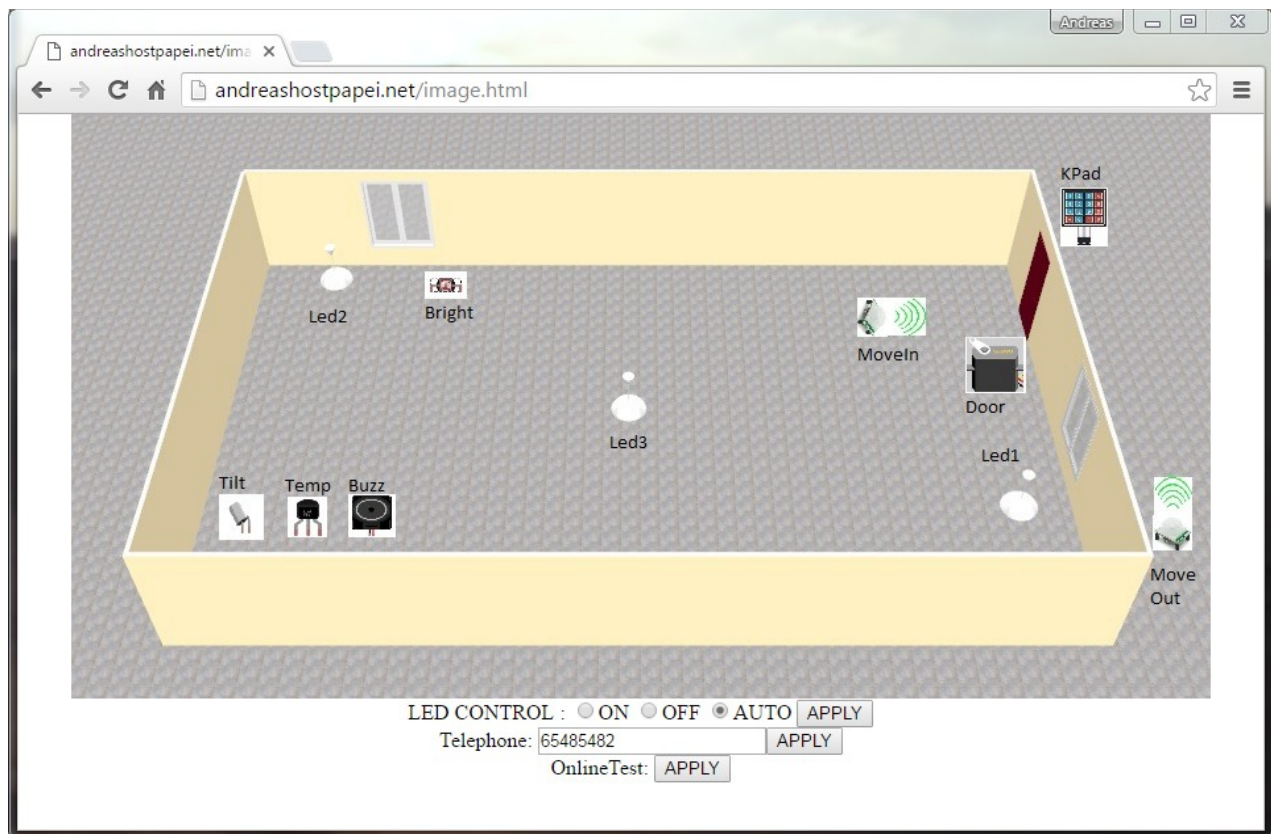
Times	Mode	Time
1	ON	2016-04-04 13:48:07
2	ON	2016-04-04 13:48:28
3	ON	2016-04-04 13:50:54

Move Out



Times	Mode	Time
1	ON	2016-04-07 17:22:44
2	ON	2016-04-07 17:22:50
3	ON	2016-04-07 17:23:37
4	ON	2016-04-07 17:24:38
5	ON	2016-04-07 17:24:54
6	ON	2016-04-07 17:25:20
7	ON	2016-04-07 17:25:25
8	ON	2016-04-07 17:25:59
9	ON	2016-04-07 17:26:31
10	ON	2016-04-07 17:26:37

Σε κάθε περίπτωση πατώντας το **BACK** (βελάκι αριστερά) επιστρέφουμε στην αρχική μας οθόνη.



Ακόμα στο κάτω μέρος της οθόνης μας υπάρχει το LED control .

Επιλέγοντας το **ON** και πατώντας το **APPLY** μέσα σε 1 λεπτό θα ανάψει το LED 3 στην οικία μας

Επιλέγονταςτο **OFF** και πατώντας το **APPLY** μέσα σε 1 λεπτό θα σβήσει το LED 3 στην οικία μας

Επιλέγονταςτο **AUTO** και πατώντας το **APPLY** μέσα σε 1 λεπτό το σύστημα θα παίρνει την απόφαση για το αν θα ανάψει η θα σβήνει το το LED 3 στην οικία μας

Ενημέρωση Βάσης Τηλεφώνου

Στο κάτω μέρος βρίσκεται το πεδίο Telephone , σε αυτό το πεδίο βάζουμε τον τηλεφωνικό αριθμό που θέλουμε να ενημερωνόμαστε από το σύστημα μας (Προσοχή **στη αρχή βάζουμε +30'Το νούμερο μας**).Έπειτα πατάμε το **Apply** , μέσα στο επόμενο λεπτό το σύστημα μας θα έχει ενημερωθεί για τον αριθμό μας.

Έλεγχος Σύνδεσης Συστήματος

Πατώντας στο **Apply** δίπλα στο Online Test μας εμφανίζεται ένας πίνακας με 5 γραμμές . Αυτές είναι οι 5 τελευταίες φορές που το Arduino μας ενημέρωσε ότι λειτουργεί . Αυτό γίνεται ανά 2 λεπτά οπότε αν δούμε ότι η τελευταία ενημέρωση είναι πχ πριν 10 λεπτά ξέρουμε ότι το σύστημά μας είναι offline .

Βιβλιογραφία

- [Www.arduino.cc](http://www.arduino.cc) πολλά tutorial
- www.github.com libraries
- <https://www.arduino.cc/en/Main/Standalone>
- <http://www.tweaking4all.com/hardware/arduino/arduino-ethernet-data-push/> Ethernet tutorial
- http://www.tutorialspoint.com/android/android_php_mysql.htm MySql tutorial
- <http://www.w3schools.com/> Html,css tutorial
- <http://www.instructables.com/id/Arduino-Infrared-Remote-tutorial/?ALLSTEPS> IRControl tutorial
- <https://el.wikipedia.org/wiki/Arduino>
- <http://home-automation-systems-review.toptenreviews.com> πληροφορίες για συστήματα
- <http://www.instructables.com/id/Arduino-Web-LED> απομακρυσμένος έλεγχος Arduino
- <https://www.youtube.com/watch?v=n-RkWRUw62g&list=LLL0n1jfFSGZRrUPkP88KIjA&index=4> tutorial για GSM
- <https://www.youtube.com/channel/UCfF3jNfZdJ8H3sOaZ0B4iDw> Βασικές γνώσεις database
- C Προγραμματισμός εισαγωγή στην C++ και Java
- Ανάλυση και Σχεδιασμός συστημάτων με τη UML 2.0
- Σχεδιασμός Βάσεων Δεδομένων ,Βασιλακόπουλος Γ
- Σημειώσεις Διδάσκοντα, Θεμιστοκλέους Μ (εργαστήρια DFD)

Ευχαριστώ τον κύριο Μηλιώνη για την συνεργασία και την ανεκτικότητα του .
Ευχαριστώ επίσης τον συνάδελφο Φίλιο Κωνσταντίνο χωρίς του οποίου την προτροπή δεν θα είχα
ασχοληθεί με αυτόν τον τομέα.