



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ Η/Υ

**Έξυπνο Σπίτι: Αυτοματισμοί με arduino για εξοικονόμηση
ενέργειας και απομακρυσμένος έλεγχος**

**Smart House: Automations with arduino for energy saving and
remote control**

Διπλωματική Εργασία

Νικόλαος Μακαρίτης

Επιβλέπων Καθηγητής: Αλκιβιάδης Ακρίτας, Καθηγητής ΠΘ

Συν επιβλέπων Καθηγητής: Γεώργιος Σταμούλης, Καθηγητής ΠΘ

Ευχαριστίες

Στο σημείο αυτό της ολοκλήρωσης των προπτυχιακών μου σπουδών θα ήθελα να ευχαριστήσω ιδιαίτερος τον καθηγητή και επιβλέπων της Διπλωματικής μου εργασίας κ. Αλκιβιάδη Ακρίτα καθώς και τον συν επιβλέπων καθηγητή κ. Γεώργιο Σταμούλη για την εμπιστοσύνη τους στο πρόσωπό μου και την συνεργασία τους.

Τους διδάσκοντες του τμήματος για την συμβολή στα εφόδια που απαιτήθηκαν ώστε να ολοκληρώσω τις σπουδές μου, αλλά κυρίως να νοιώσω την απαραίτητη αυτοπεποίθηση στον εργασιακό χώρο επαγγελματικά.

Επίσης θα ήθελα να ευχαριστήσω ιδιαίτερος την οικογένεια μου που με την υποστήριξη της όλα αυτά τα χρόνια, τόσο ψυχολογική όσο και οικονομική, κατάφερα να ολοκληρώσω τις σπουδές μου. Επίσης θα ήθελα να ευχαριστήσω τους συμφοιτητές-φίλους που και αυτοί από τη μεριά τους, με την βοήθεια και την ενθάρρυνση τους συνέβαλλαν στην έως τώρα πορεία μου.

Περίληψη

Τα τελευταία χρόνια υπάρχει έντονο ερευνητικό ενδιαφέρον γύρω από την ανάπτυξη έξυπνων περιβαλλόντων, τα οποία μπορούν να υποστηρίξουν προηγμένες υπηρεσίες διάχυτου υπολογισμού (pervasive services) οι οποίες έχουν επίγνωση του περιβάλλοντος και αλληλοεπιδρούν μαζί του. Βασικό στοιχείο κάθε τέτοιου έξυπνου χώρου, είναι η ικανότητά του να εκτελεί λογικούς συλλογισμούς (reasoning capability) και να συντονίζει τις προσφερόμενες υπηρεσίες σύμφωνα με τις συνθήκες, καθώς και τις προτιμήσεις και τις ανάγκες των χρηστών, χωρίς συνήθως την άμεση εμπλοκή τους.

Η συγκεκριμένη διπλωματική εργασία έχει ως στόχο να περιγράψει την υλοποίηση και εφαρμογή ενός «έξυπνου σπιτιού» με την χρήση αυτοματισμών και απομακρυσμένου ελέγχου από οποιαδήποτε συσκευή μπορεί να συνδεθεί στο διαδίκτυο και έχει πρόσβαση σε οποιονδήποτε φυλλομετρητή. Η διευκόλυνση των ανθρώπων και η εξοικονόμηση ενέργειας είναι που με ενέπνευσαν γ αυτήν την εργασία. Τρόπους ώστε το σπίτι να μπορεί από μόνο του μέσω αισθητήρων να κάνει συγκεκριμένες ενέργειες καθώς και το πώς θα διαμεσολαβήσει ένας απομακρυσμένος εξυπηρετητής, ανάμεσα σε μια συσκευή ελέγχου και την συσκευή προς έλεγχο, έτσι ώστε να καταστήσει δυνατή την επικοινωνία μεταξύ τους.

Πιο συγκεκριμένα το κείμενο θα είναι χωρισμένο σε τμήματα. Το ένα θα είναι το τμήμα της προετοιμασίας, σχεδίασης και υλοποίησης από άποψη υλικού και από άποψη λογισμικού, το επόμενο θα είναι το τμήμα της εγκατάστασης και το τελευταίο θα είναι το τμήμα των προτάσεών μου για μελλοντική ανάπτυξη. Όσον αφορά το τελευταίο τμήμα, οι προτάσεις θα αφορούν είτε βελτίωση του υπάρχοντος είτε προσθήκη επιπλέον λειτουργιών.

Λέξεις-Κλειδιά

Arduino, Ethernet shield, Έξυπνο Σπίτι, Αυτοματισμοί, προγραμματισμός, ιστοσελίδα, ασφάλεια, προσβασιμότητα

Abstract

In recent years there has been intense research interest around the development of smart environments, which can support advanced pervasive computing services (pervasive services) which are aware of the environment and interact with it. A key element of any such smart space, is the ability to perform logical reasoning (reasoning capability) and coordinate the services according to the conditions and the preferences and needs of users, usually without their direct involvement.

This thesis aims to describe the development and implementation of a "smart home" with the use of automation and remote control from any device can connect to the Internet and is accessible to any browser. Facilitating people and saving energy is what inspired this work. Ways that the house can itself via sensors make specific actions and how to mediate a remote server, between a control device and the device to be controlled so as to enable communication between them.

More specifically, the text is divided into sections. One will be part of the preparation, design and implementation in terms of hardware and in terms of software, the next will be the part of the plant and the last will be the part of this Opinion for future growth. What concerns the last part, the proposals involving either improve existing or add additional functions.

Key Words

Arduino, Ethernet shield, Smart House, automations, programming, webpage, security, accesibility

Περιεχόμενα

Εισαγωγή	1
Υλοποίηση	3
Ο μικροελεγκτής Arduino Mega 2560	4
Pin Mapping	6
Ρεύμα	7
Μνήμη.....	7
Επικοινωνία.....	8
Γλώσσα Προγραμματισμού	8
Ethernet Shield.....	9
Pin Mapping	10
Arduino Mega + Ethernet Shield.....	11
Arduino Relay Board	11
Αισθητήρες	12
Προγραμματισμός.....	13
Δημιουργία της ιστοσελίδας	13
Κώδικας HTML	18
Προσβασιμότητα.....	19
Πόρτες (Ports)	21
Port Forwarding	22
Ορίζουμε σταθερή IP στον υπολογιστή μας.....	22
Το άνοιγμα θύρας στο router	23
Είσοδος στο web interface	24
Εύρεση της ενότητας για το άνοιγμα θύρας	24
Δημιουργία κανόνα για το άνοιγμα θύρας	25
Άνοιγμα θύρας στο firewall των Windows	27
Ασφάλεια	31
Η εφαρμογή στην οικία.....	36
Σύνοψη.....	39
Βιβλιογραφία	40

Εισαγωγή

Σήμερα λόγω του επιβαρυσμένου προγράμματος και την ανάγκη εκπλήρωσης πολλών αναγκών και υποχρεώσεων σε περιορισμένο χρόνο, ο άνθρωπος κατέστησε απαραίτητη την ανάγκη να κάποιες εργασίες να γίνονται πιο εύκολα και πιο γρήγορα απ' ότι στο παρελθόν. Έτσι λοιπόν χάρη στην τεχνολογία και τα επιτεύγματά της δημιουργήθηκαν οι αυτοματισμοί προς διευκόλυνσή μας.

Οι αυτοματισμοί και οι αυτοματοποιημένες διεργασίες είναι πλέον αναπόσπαστο κομμάτι της ζωής μας καθώς όπως αναφέραμε υπάρχει σε πολλές περιπτώσεις μεγάλο κέρδος χρόνου και κόπου χάρη σ αυτούς. Υπάρχει σε πολλούς διαφορετικούς τομείς ήδη και γίνεται συνεχώς προσπάθεια ένταξης και σε νέους.

Κάποια αυτοματοποιημένα συστήματα υπάρχουν ήδη στα μέσα μεταφοράς, σε βιομηχανικές κατασκευές, σε κτήρια κλπ. όπως είναι ο αυτόματος φωτισμός, το αυτόματο πότισμα και ο συναγερμός. Όλες αυτές λοιπόν οι έξυπνες λύσεις παίζουν κυρίαρχο ρόλο στην καθημερινότητά μας όπου ο τρόπος ζωής μας πλέον τις καθιστά απαραίτητες.

Αυτές λοιπόν οι αυτοματοποιημένες συσκευές είναι σχεδιασμένες άλλες να λειτουργούν μεμονωμένα και άλλες να πραγματοποιούν πιο πολύπλοκες διαδικασίες όταν βρίσκονται σε συνεργασία με άλλους μηχανισμούς ή σένσορες. Το θέμα όμως είναι να μπορεί κάποιος να προσαρμόσει αυτές τις λειτουργίες στις προσωπικές του ανάγκες και τα θέλω του καθώς και να μπορεί να τις ελέγξει ακόμα με τη χρήση του διαδικτύου.

Εδώ λοιπόν έρχεται το Ίντερνετ των Πραγμάτων (Internet Of Things) να προσφέρει νέα αξία στις ζωές των καταναλωτών, μεταμορφώνοντας τα καθημερινά αντικείμενα σε έξυπνες συνδεδεμένες συσκευές που ουσιαστικά κατανοούν τους καταναλωτές και τους προσφέρουν πλεονεκτήματα και υπηρεσίες που καλύπτουν τις ανάγκες και τις επιθυμίες τους. Το IoT είναι ότι αποτελείται από 'έξυπνα' (δηλ. εξοπλισμένα με υπολογιστή) 'πράγματα' (δηλ. συσκευές) που συνδέονται τόσο μεταξύ τους, όσο και με βάσεις δεδομένων (δηλ. servers) με στόχο να παρέχουν πλήθος υπηρεσιών που αξιοποιούν και βασίζονται στα δεδομένα που παρέχουν οι συσκευές που έχουν στην κατοχή τους και χρησιμοποιούν οι καταναλωτές καθημερινά.

Και ενώ ακούγεται επαναστατικό και καινοτόμο ως concept, το IoT δεν είναι κάτι καινούριο στην πράξη, καθώς έχει βασιστεί στις εξελίξεις της τεχνολογίας μέσα στις τελευταίες δεκαετίες. Μία από αυτές τις εξελίξεις αφορά στο κομμάτι των ηλεκτρονικών υπολογιστών – στην εξέλιξη τους δηλαδή από τους κεντρικούς υπολογιστές στους ηλεκτρονικούς υπολογιστές (PC), από τα PC στα notebooks και από τα notebooks στις φορητές συσκευές (όπως τα tablets, smartphones και τα wearables). Κατά τη διάρκεια αυτής της εξέλιξης, οι συσκευές έγιναν μικρότερες – πια υπάρχουν ακόμα και συσκευές που φοριούνται - καθώς όλα τα εξαρτήματα περιορίστηκαν σε μέγεθος, ενώ οι επιδόσεις τους βελτιώθηκαν - τόσο που αρκετές συσκευές σήμερα ξεπερνούν σε επιδόσεις τους ογκώδεις κεντρικούς υπολογιστές του παρελθόντος.

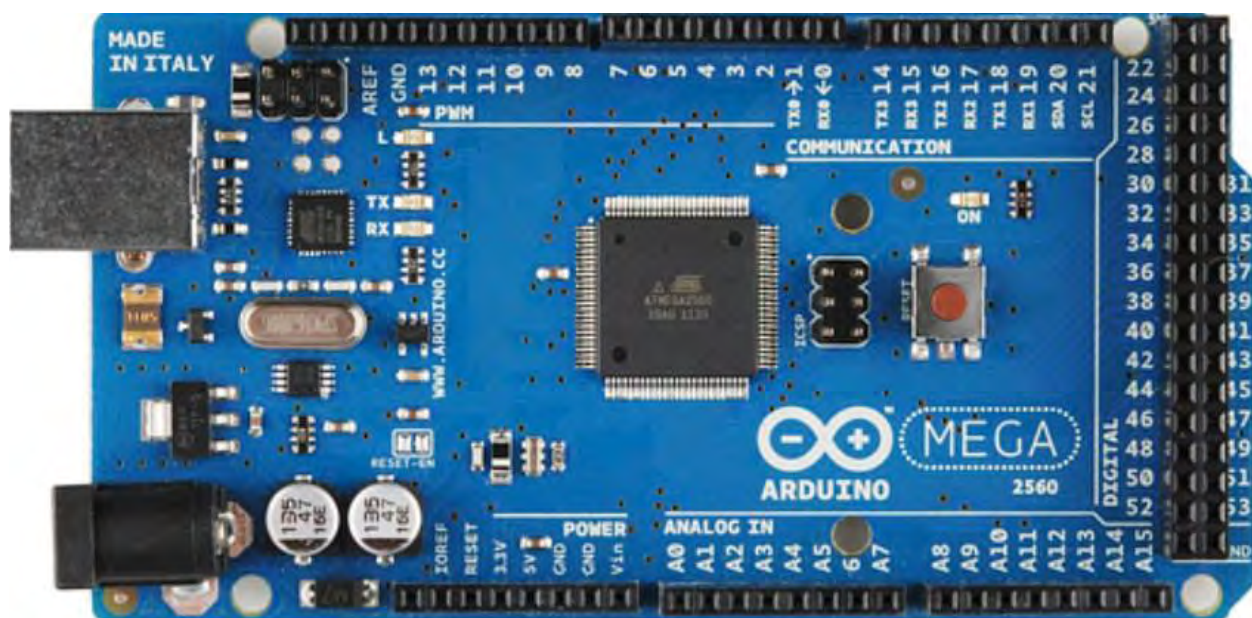
Υλοποίηση

Σκοπός μας είναι να συνδέσουμε έναν μικροελεγκτή στο διαδίκτυο μέσω ενός Ethernet Modude. Στη συνέχεια να ορίσουμε μία IP στην οποία θα πραγματοποιείται αυτή η σύνδεση και να δημιουργήσουμε ένα γραφικό περιβάλλον σε αυτήν την ιστοσελίδα. Μέσα από αυτήν την ιστοσελίδα θα μπορούμε να ελέγξουμε διάφορους μηχανισμούς μέσα σ ένα σπίτι καθώς επίσης και να παρατηρούμε τιμές από μετρητές που θα έχουμε ορίσει σε συγκεκριμένα σημεία για συγκεκριμένους λόγους.

Τα υλικά που θα χρειαστούν γ αυτήν την υλοποίηση είναι τα παρακάτω:

- Arduino Mega 2560
- Arduino Ethernet Shield
- Ethernet Cable Cat5e RJ45
- Μετασχηματιστής 5V DC
- Arduino Relay Board 5V Trigger
- Αισθητήρας Θερμοκρασίας DS1820
- Αισθητήρας Βροχής
- Αισθητήρας Κίνησης
- Καλώδια ενώσεων
- Αντιστάσεις

Ο μικροελεγκτής Arduino Mega 2560



Το Mega Arduino είναι μια πλακέτα μικροελεγκτή με βάση την ATmega1280. Έχει 54 ψηφιακές καρφίτσες εισόδου / εξόδου (εκ των οποίων 14 μπορούν να χρησιμοποιηθούν ως έξοδοι PWM), 16 αναλογικές εισόδους, 4 UARTs (hardware σειριακές θύρες), ένα 16 MHz κρύσταλλο ταλαντωτή, μια σύνδεση USB, μια υποδοχή τροφοδοσίας, μια κεφαλή ICSP, και ένα κουμπί επαναφοράς. Περιέχει όλα όσα χρειάζονται για τη στήριξη του μικροελεγκτή.

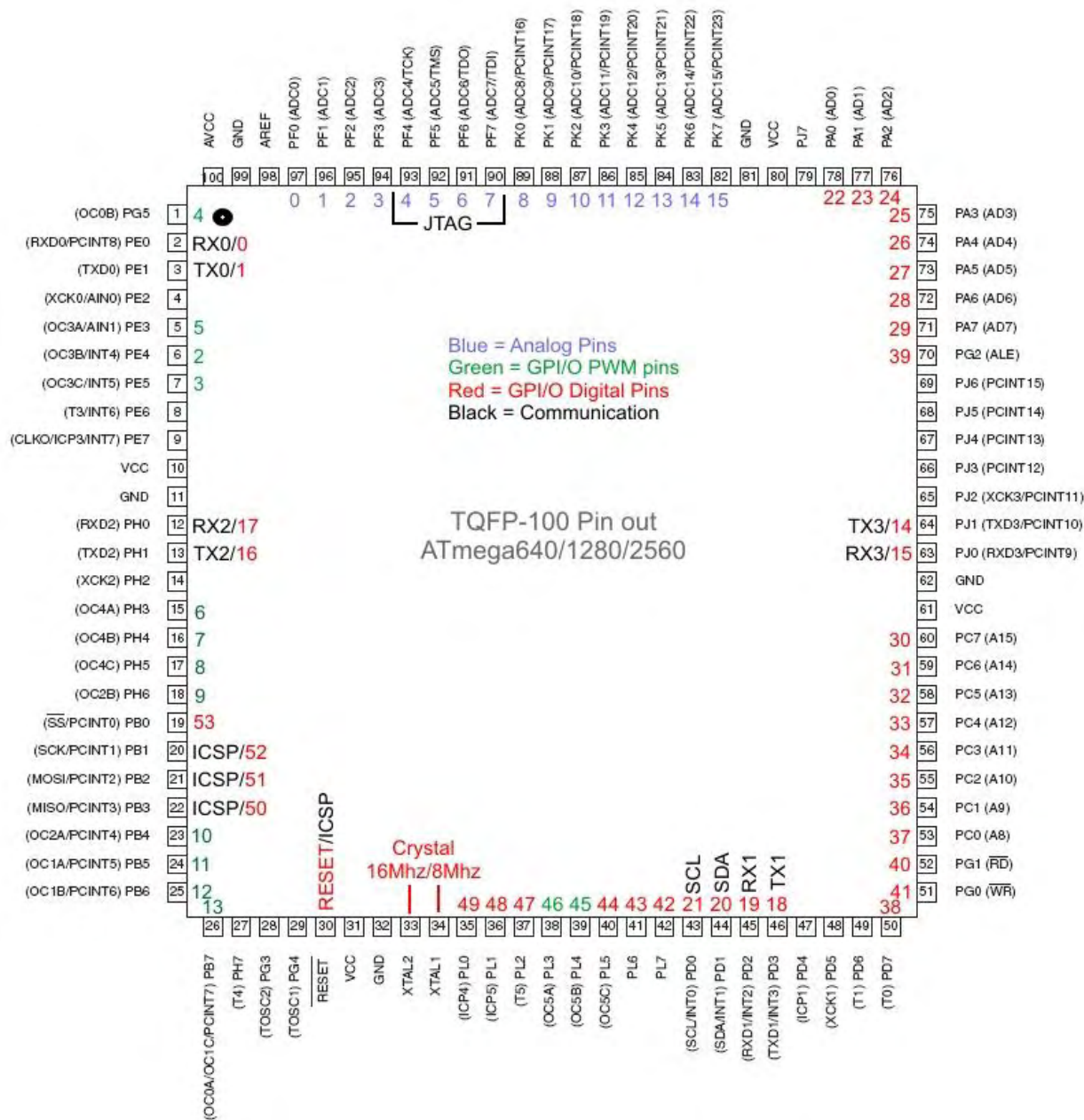
Ένας μικροελεγκτής έχει ενσωματωμένη μνήμη και θύρες Εισόδου/Εξόδου (I/O), ενώ αντίθετα σε ένα μικροεπεξεργαστή χρειάζεται να συνδέσουμε τα παραπάνω σαν πρόσθετα περιφερειακά κυκλώματα. Οι μικροελεγκτές είναι το κύριο στοιχείο σε πολλά είδη ηλεκτρονικού εξοπλισμού. Ένα τυπικό σπίτι στο δυτικό κόσμο είναι πιθανό να περιλαμβάνει μόνο έναν ή δύο γενικού σκοπού μικροεπεξεργαστές, αλλά περισσότερους από είκοσι μικροελεγκτές. Μικροελεγκτές απαντώνται σε οποιονδήποτε τύπο ηλεκτρικής συσκευής, πλυντήρια ρούχων, φούρνους μικροκυμάτων, τηλέφωνα κ.λπ.

- Τα βασικά χαρακτηριστικά του συγκεκριμένου μικροελεγκτή είναι:

- Αναβαθμισμένη RISC αρχιτεκτονική
- 131 πολύ ισχυρές εντολές – οι περισσότερες απαιτούν μόνο ένα κύκλο ρολογιού για την εκτέλεση τους
- 54 καταχωρητές μεγέθους 8-bit γενικής χρήσης
- Μέχρι 20 MIPS στα 16MHz
- On chip πολλαπλασιαστής 2 κύκλων
- Μνήμη προγράμματος και δεδομένων
- 32 KB Αυτοπρογραμματιζόμενη μνήμη flash
- 1 KB EEPROM μνήμη
- 2 KB εσωτερική SRAM
- Πλήθος περιφερειακών
- 2 timers 8-bit με ξεχωριστά ρολόγια
- 1 timer 16-bit με ξεχωριστό ρολόι
- Μετρητής πραγματικού χρόνου με εξωτερικό κρύσταλλο
- 48 γραμμές για ψηφιακή είσοδο/έξοδο (6 εκ των οποίων υποστηρίζουν PWM)
- 6 γραμμές για αναλογική είσοδο (με 10-bit αναλογο-ψηφιακό μετατροπέα ADC)
- Master/Slave SPI λειτουργία
- Σειριακή θύρα με δυνατότητες σύγχρονης και ασύγχρονης λειτουργίας
- Watchdog timer με ξεχωριστό κρύσταλλο
- On-chip αναλογικό συγκριτή

- Ειδικά χαρακτηριστικά – Reset αυτόματα με την τροφοδότηση
- Εσωτερικό ρολόι
- Εσωτερικές και εξωτερικές διακοπές
- 6 sleep modes για εξοικονόμηση ενέργειας
- Ταχύτητα έως 16MHz

Pin Mapping



Ρεύμα

Το Mega Arduino μπορεί να τροφοδοτείται μέσω της σύνδεσης USB ή με εξωτερική τροφοδοσία ρεύματος. Η πηγή ενέργειας επιλέγεται αυτόματα.

Εξωτερικό (μη-USB) ρεύμα μπορεί να προέλθει είτε από έναν προσαρμογέα AC σε DC (τοίχο-κονδυλωμάτων) ή μπαταρία. Ο προσαρμογέας μπορεί να συνδεθεί με τη σύνδεση ενός 2,1 χιλιοστά κέντρο θετικό βύσμα στην υποδοχή ρεύματος της πλακέτας.

Η πλακέτα μπορεί να λειτουργήσει με εξωτερική πηγή 6 έως 20 βολτ. Εάν τροφοδοτείται με λιγότερο από 7V, εντούτοις, η υποδοχή 5V μπορεί να παρέχει λιγότερο από πέντε βολτ και η πλακέτα μπορεί να είναι ασταθής. Εάν χρησιμοποιείτε περισσότερα από 12V, ο ρυθμιστής τάσης μπορεί να υπερθερμανθεί και να προκαλέσει ζημιά στην πλακέτα. Η συνιστάμενη εμβέλεια είναι 7 έως 12 βολτ.

Οι ακίδες ρεύματος είναι ως εξής:

VIN. Η τάση εισόδου στην πλακέτα Arduino όταν χρησιμοποιείται μια εξωτερική πηγή ισχύος (σε αντιδιαστολή με 5 βολτ από τη σύνδεση USB ή άλλες οργανωμένες πηγή ενέργειας). Μπορείτε να παρέχετε τάση μέσω αυτής της υποδοχής.

5V. Η ρυθμιζόμενη παροχή ηλεκτρικού ρεύματος που χρησιμοποιείται για να τροφοδοτήσει το μικροελεγκτή και άλλα συστατικά στην πλακέτα. Αυτό μπορεί να προέλθει είτε από VIN μέσω ενός ρυθμιστή επί της πλακέτας, ή να τροφοδοτείται από USB ή άλλη ρυθμιζόμενη παροχή 5V.

3V3. Η παροχή 3,3 βολτ που παράγεται από το chip επί της πλακέτας FTDI. Η μέγιστη κατανάλωση ρεύματος είναι 50 mA.

GND. Καρφίτσα γείωσης.

Μνήμη

Η ATmega1280 έχει 128 KB μνήμη flash για την αποθήκευση κώδικα (εκ των οποίων 4 KB χρησιμοποιείται για τον φορτωτή εκκίνησης), 8 KB SRAM και των 4 KB EEPROM του (το οποίο μπορεί να διαβαστεί και να γραφτεί με τη βιβλιοθήκη EEPROM).

Επικοινωνία

Το Mega Arduino έχει μια σειρά από εγκαταστάσεις για την επικοινωνία με έναν υπολογιστή, ένα άλλο Arduino, ή άλλους μικροελεγκτές. Η ATmega1280 παρέχει τέσσερις UARTs υλικού για TTL (5V) της σειριακής επικοινωνίας. Μια FTDI FT232RL στα κανάλια της πλακέτας μία από αυτές μέσω USB και τους οδηγούς FTDI (συμπεριλαμβάνεται με το λογισμικό Arduino) παρέχει μια εικονική θύρα COM για το λογισμικό στον υπολογιστή. Το λογισμικό Arduino περιλαμβάνει μια σειριακή οθόνη η οποία επιτρέπει την απλά δεδομένα κειμένου που θα αποσταλούν προς και από την πλακέτα Arduino. Οι RX και TX λυχνίες LED στο ταμπλό αναβοσβήνει όταν γίνεται μετάδοση δεδομένων μέσω του FTDI τσιπ και USB σύνδεση με τον υπολογιστή (όχι όμως για σειριακή επικοινωνία στις ακίδες 0 και 1).

Η ATmega1280 υποστηρίζει επίσης I2C (TWI) και την επικοινωνία SPI. Το λογισμικό Arduino περιλαμβάνει μια βιβλιοθήκη καλωδίων για να απλοποιήσει τη χρήση του διαύλου I2C.

Γλώσσα Προγραμματισμού

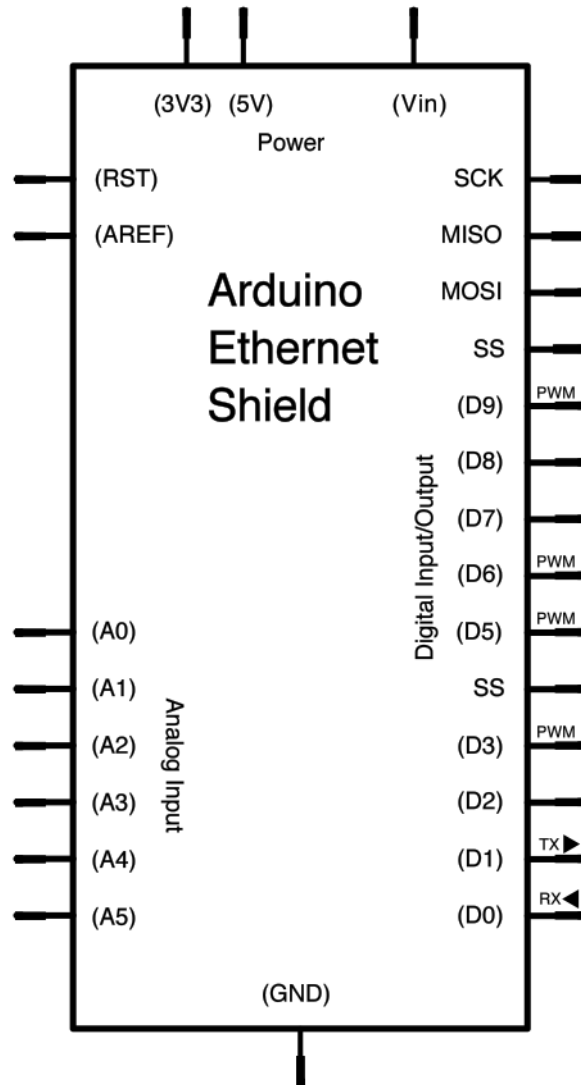
Η γλώσσα του Arduino βασίζεται στη γλώσσα Wiring μια παραλλαγή C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C καθώς και μερικά χαρακτηριστικά της C++. Για compiler χρησιμοποιείται ο AVR gcc και ως βασική βιβλιοθήκη C χρησιμοποιείται η AVR libc. Λόγω της καταγωγής της από τη C, στη γλώσσα του Arduino, μπορούν να χρησιμοποιηθούν ουσιαστικά οι ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπων δεδομένων και τους ίδιους τελεστές όπως και στη C. Πέρα από αυτές όμως, υπάρχουν κάποιες ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν για τη διαχείριση του ειδικού hardware του Arduino. Τα προγράμματα του Arduino διαιρούνται σε τρία μέρη: δομή (structure), τιμές (values) και συναρτήσεις (functions). Οι βασικές και απαραίτητες συναρτήσεις ενός κώδικα για Arduino είναι οι void setup() και void loop(). Η πρώτη χρησιμοποιείται για την αρχικοποίηση των απαραίτητων δεδομένων και εκτελείται μια φορά κατά την εκκίνηση. Η δεύτερη, όπως προδίδει και το όνομά της εκτελείται επαναληπτικά για όσο χρονικό διάστημα είναι ενεργοποιημένη η συσκευή.

Ethernet Shield



Το Arduino Ethernet Shield Arduino μας συνδέει με το διαδίκτυο σε λίγα λεπτά. Απλά συνδέουμε αυτό το module στο Arduino μας, συνδεόμαστε με το δίκτυό μας με ένα καλώδιο RJ45. Όπως πάντα με Arduino, κάθε στοιχείο της πλατφόρμας είναι ελεύθερα διαθέσιμο. Αυτό σημαίνει ότι μπορούμε να μάθουμε πώς ακριβώς έχει κατασκευαστεί και χρησιμοποιούμε το σχεδιασμό του ως σημείο εκκίνησης για τα δικά μας κυκλώματα.

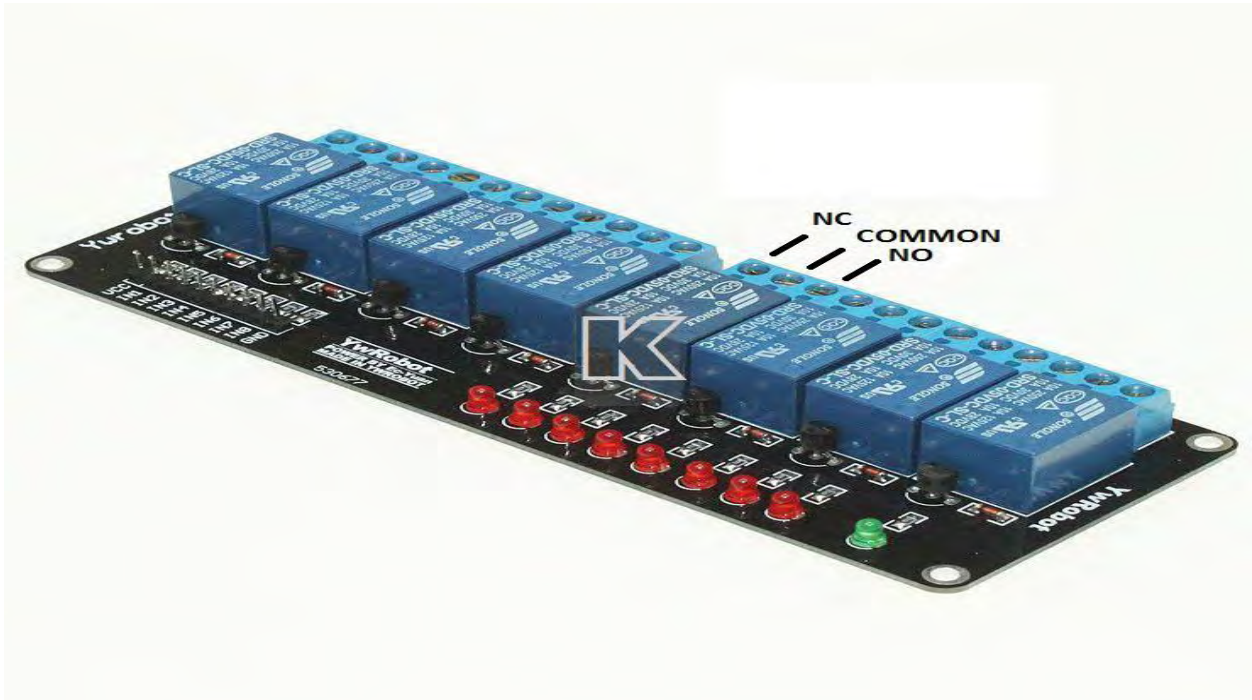
Pin Mapping



Arduino Mega + Ethernet Shield



Arduino Relay Board



Αισθητήρες

Θερμοκρασίας



Βροχής



Κίνησης



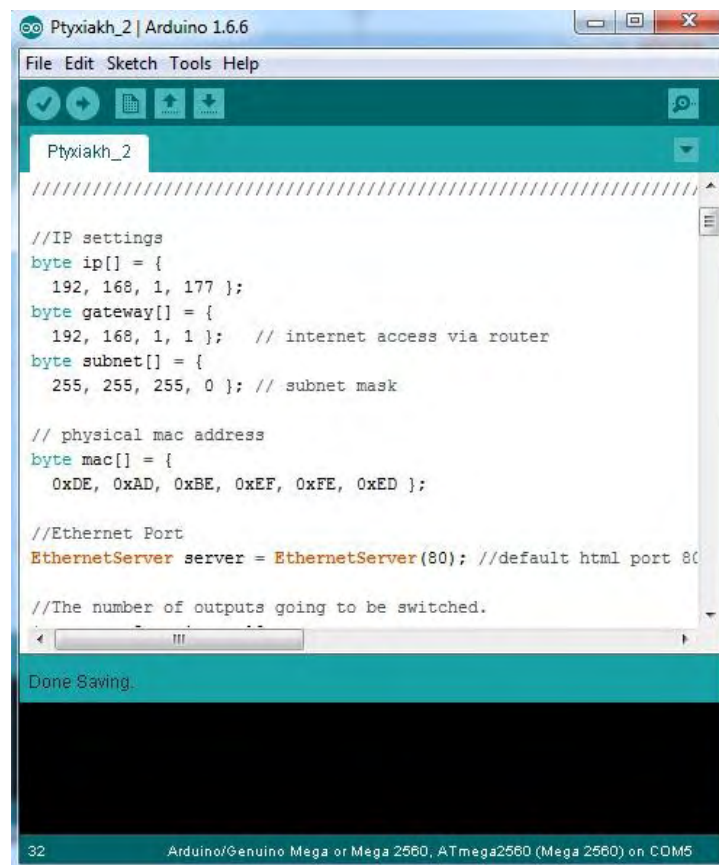
Υγρασίας



Προγραμματισμός

Δημιουργία της ιστοσελίδας

Σκοπός μου είναι να θέσω τον μικροελεγκτή ως σέρβερ σε μία IP που θα του ορίσω με την προϋπόθεση βέβαια ότι αυτή δεν είναι δεσμευμένη από κάποια άλλη συσκευή στο τοπικό μου δίκτυο. Αν είναι την ξαναορίζω έτσι ώστε να μην αντιστοιχεί σε καμία άλλη παρά μόνο στο arduino. Δημιουργώντας λοιπόν τον σέρβερ μου μέσα στον ελεγκτή ορίζω την IP διεύθυνση, την MAC φυσική διεύθυνση του arduino, το gateway για να μπορεί να συνδεθεί μέσω router, την μάσκα υποδικτύου (subnet mask) καθώς και το Port. Στη συνέχεια αφού κάνω ελέγχους ότι η σύνδεση έχει πραγματοποιηθεί με επιτυχία εμφανίζεται στο Serial Port και το αντίστοιχο μήνυμα, ενώ αν υπήρξε κάποιο πρόβλημα κατά την σύνδεσή του στο διαδίκτυο ενημερωνόμαστε και πάλι μέσω μηνυμάτων.



```

Ptyxiakh_2 | Arduino 1.6.6
File Edit Sketch Tools Help
Ptyxiakh_2
////////////////////////////////////
//IP settings
byte ip[] = {
  192, 168, 1, 177 };
byte gateway[] = {
  192, 168, 1, 1 }; // internet access via router
byte subnet[] = {
  255, 255, 255, 0 }; // subnet mask

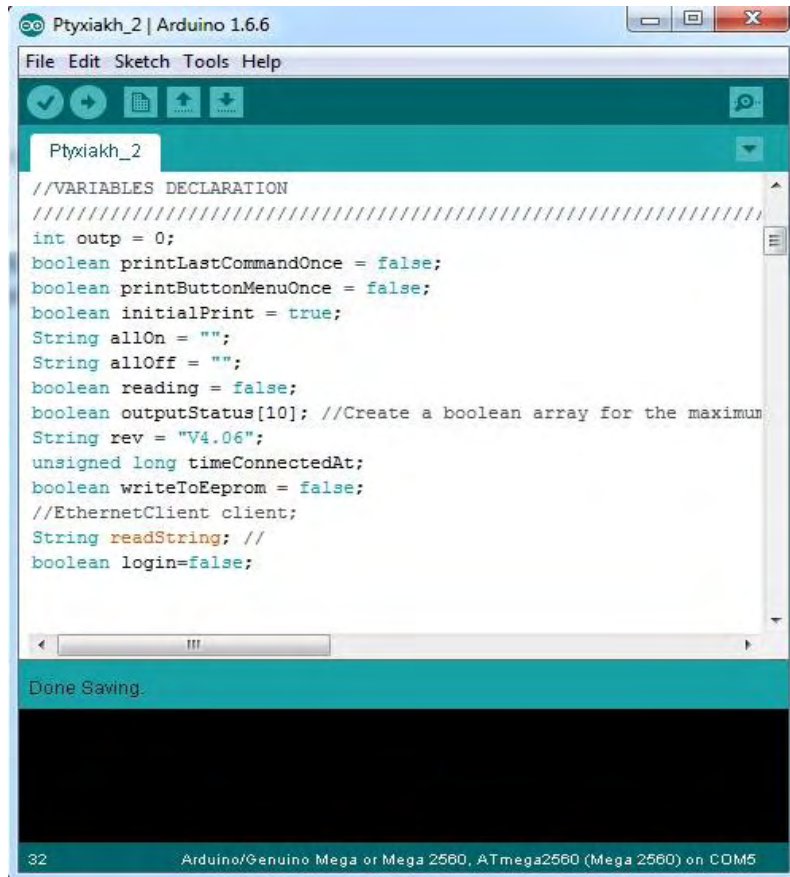
// physical mac address
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };

//Ethernet Port
EthernetServer server = EthernetServer(80); //default html port 80

//The number of outputs going to be switched.
Done Saving.
32 Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM5

```

Στην συνέχεια και εφόσον έχει εξασφαλιστεί ότι δουλεύει σωστά ο σέρβερ μας προχωρώ στο παρακάτω βήμα το οποίο είναι ο προγραμματισμός της πλακέτας για την δημιουργία της ιστοσελίδας. Έτσι λοιπόν ορίζω ποια pins της πλακέτας θα χρησιμοποιηθούν ως είσοδοι και ποια ως έξοδοι. Ύστερα δηλώνω και αρχικοποιώ όλες τις μεταβλητές που θα χρησιμοποιηθούν παρακάτω στο πρόγραμμα.



```

Ptyxiakh_2 | Arduino 1.6.6
File Edit Sketch Tools Help
Ptyxiakh_2
//VARIABLES DECLARATION
////////////////////////////////////
int outp = 0;
boolean printLastCommandOnce = false;
boolean printButtonMenuOnce = false;
boolean initialPrint = true;
String allOn = "";
String allOff = "";
boolean reading = false;
boolean outputStatus[10]; //Create a boolean array for the maximum
String rev = "V4.06";
unsigned long timeConnectedAt;
boolean writeToEeprom = false;
//EthernetClient client;
String readString; //
boolean login=false;

Done Saving.

32 Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM5

```

Κατόπιν δημιουργώ το γραφικό περιβάλλον της ιστοσελίδας μου καλώ τις συναρτήσεις `printHtmlHeader()`, `printLoginTitle()`, `printHtmlButtonTitle()` και `printHtmlFooter`. Η συγκεκριμένη ιστοσελίδα αποτελείται από διάφορους διακόπτες που αντιστοιχούν σε κάποια λειτουργία η οποία αναφέρεται δίπλα τους. Επίσης στα δεξιά των διακοπών υπάρχουν και ενδείξεις πράσινου led που παρουσιάζει την κατάσταση του συγκεκριμένου διακόπτη εκείνη την στιγμή, έτσι ώστε να υπάρχει πάντα πλήρης επίγνωση του τι λειτουργεί και τι όχι. Ακόμη υπάρχει μία ένδειξη που δείχνει την θερμοκρασία του Θερμοσίφωνα έτσι ώστε να μπορεί να αποφασίσει ο χρήστης αν επιθυμεί να το θέσει σε λειτουργία ή όχι. Όλα αυτά παρουσιάζονται στις φωτογραφίες παρακάτω.

Home Automation

Switch the required output.

Temperature 73.20 °C

01. Gate	ON	OFF	●
02. Garage	ON	OFF	●
03. Outside Lights	ON	OFF	●
04. A/C	ON	OFF	●
05. Main Heater	ON	OFF	●
06. Water Heater	ON	OFF	●
07. Garden Watering	ON	OFF	●
08. Kitchen	ON	OFF	●
09. Tents	ON	OFF	●
10. MAIN SWITCH	ON	OFF	●

Taskbar: Home Automation - ..., Untitled3 - Paint, Ptiyakh_2, Ptiyakh_2 | Arduino...

Home Automation

Switch the required output.

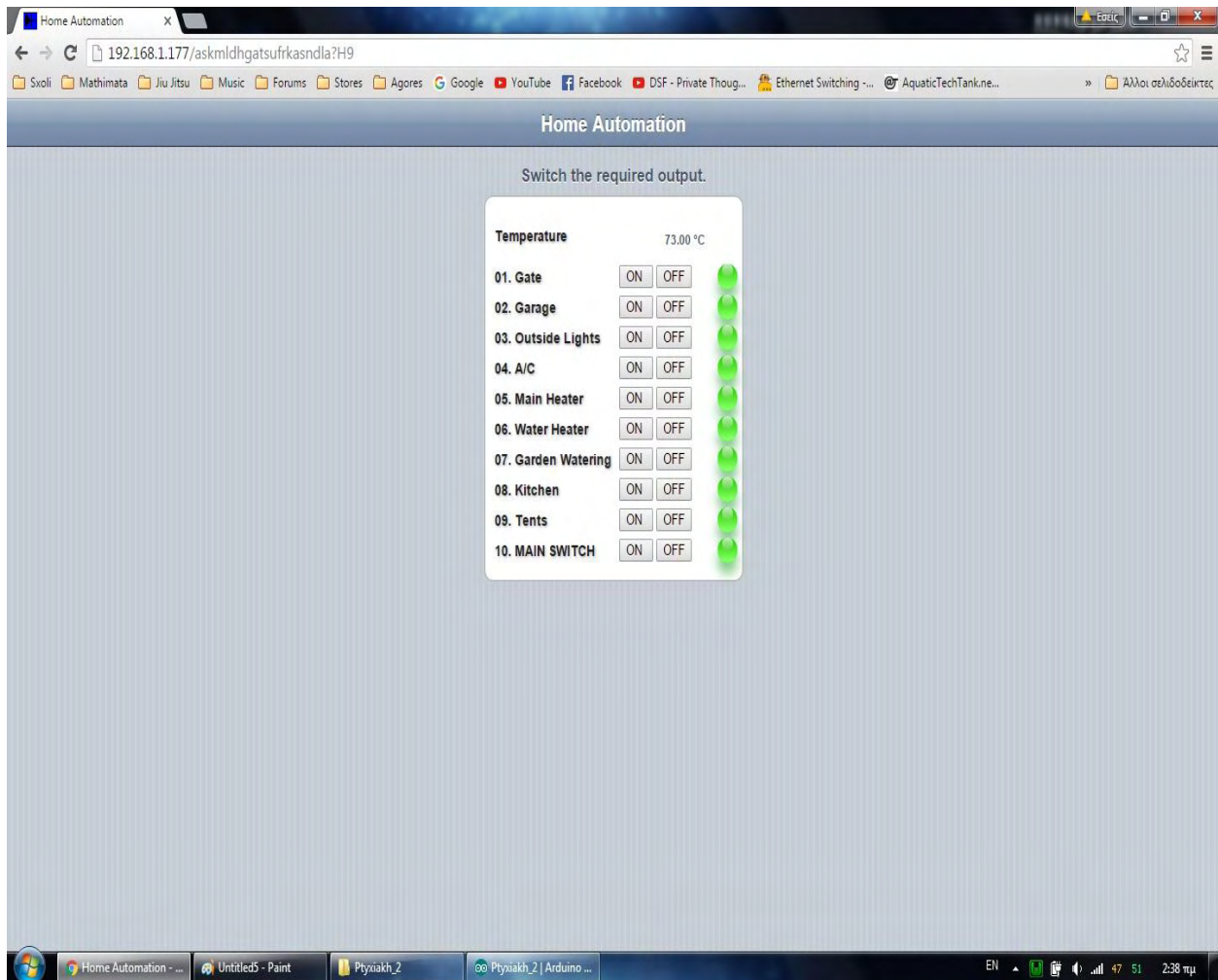
Temperature 72.50 °C

01. Gate	ON	OFF	●
02. Garage	ON	OFF	●
03. Outside Lights	ON	OFF	●
04. A/C	ON	OFF	●
05. Main Heater	ON	OFF	●
06. Water Heater	ON	OFF	●
07. Garden Watering	ON	OFF	●
08. Kitchen	ON	OFF	●
09. Tents	ON	OFF	●
10. MAIN SWITCH	ON	OFF	●

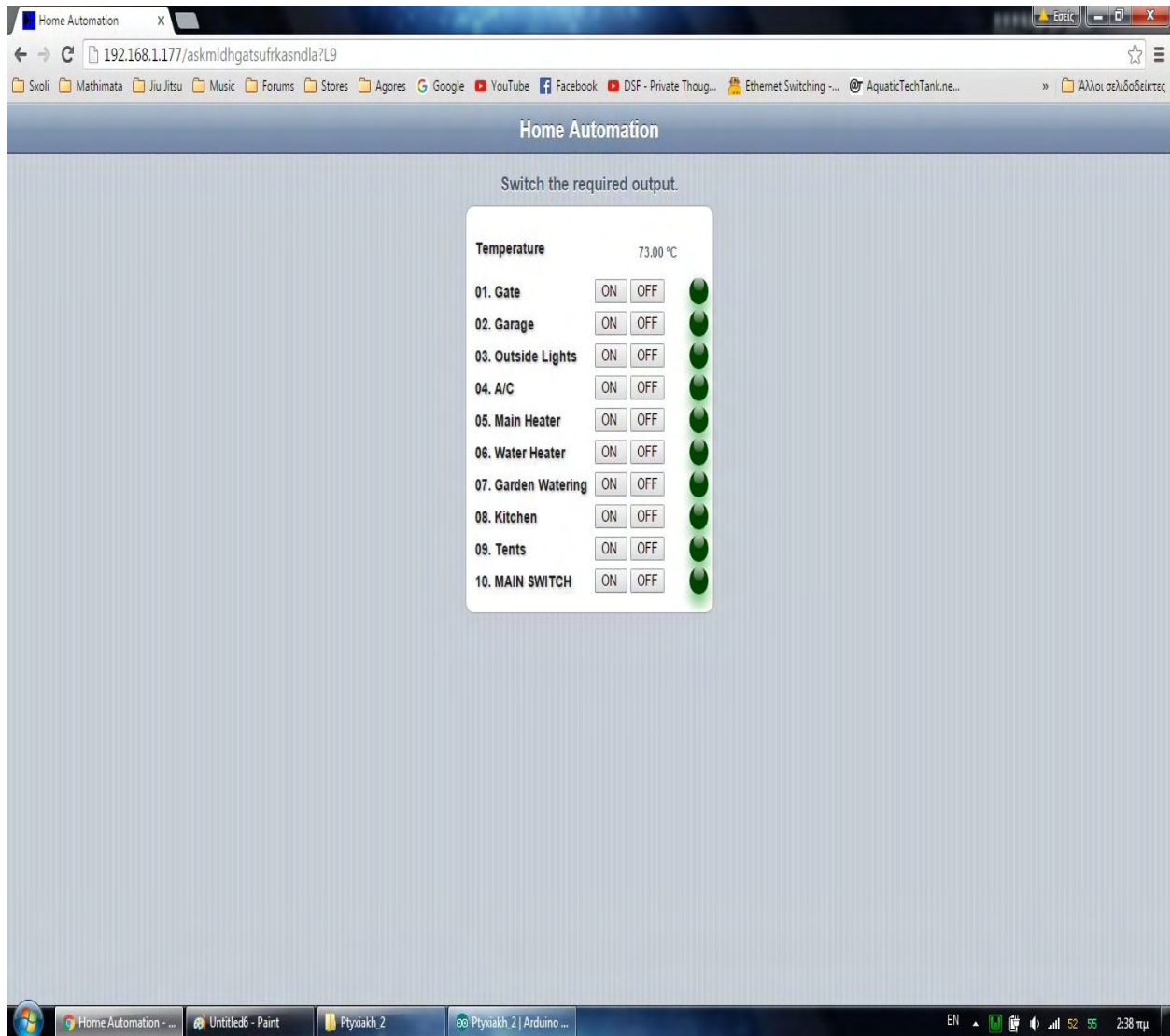
Taskbar: Home Automation - ..., Untitled4 - Paint, Ptiyakh_2, Ptiyakh_2 | Arduino...

Εδώ αξίζει να σημειωθεί επίσης ότι δημιούργησα έναν γενικό διακόπτη ο οποίος ελέγχει όλες τις εξόδους έτσι ώστε να μπορώ να τις ανάψω ή να τις σβήσω όλες μαζί.

MAIN SWITCH “ON”



MAIN SWITCH “OFF”



Κώδικας HTML

```

Ptyxiakh_2 | Arduino 1.6.6
File Edit Sketch Tools Help

Ptyxiakh_2

client.println("<!DOCTYPE HTML>");
client.println("<head>");

// add page title
client.println("<title>Home Automation</title>");
client.println("<meta name=\"description\" content=\"Home Automation\"/>");

// add a meta refresh tag, so the browser pulls again every x seconds:
client.print("<meta http-equiv=\"refresh\" content=\"\"");
client.print(refreshPage);
client.println("; url=/askmldhgatsufrkasndla\">");

// add other browser configuration
client.println("<meta name=\"apple-mobile-web-app-capable\" content=\"yes\">");
client.println("<meta name=\"apple-mobile-web-app-status-bar-style\" content=\"default\">");
client.println("<meta name=\"viewport\" content=\"width=device-width, user-scalable=no\">");

//inserting the styles data, usually found in CSS files.
client.println("<style type=\"text/css\">");
client.println("");

//This will set how the page will look graphically
client.println("html { height:100%; }");

client.println("  body {");
client.println("    height: 100%;");
client.println("    margin: 0;");
client.println("    font-family: helvetica, sans-serif;");
client.println("    -webkit-text-size-adjust: none;");
client.println("  }");
client.println("");
client.println("body {");
client.println("  -webkit-background-size: 100% 21px;");
client.println("  background-color: #c5ccd3;");
client.println("  background-image:");
client.println("  -webkit-gradient(linear, left top, right top,");
client.println("  color-stop(.75, transparent),");
client.println("  color-stop(.75, rgba(255,255,255,.1)) );");
client.println("  -webkit-background-size: 7px;");
client.println("}");
client.println("");
client.println(".view {");
client.println("  min-height: 100%;");
client.println("  overflow: auto;");
client.println("}");
client.println("");

```


Παραπάνω φαίνεται ένα απόσπασμα του κώδικα html που χρησιμοποίησα για το interface της ιστοσελίδας, τροποποιημένο έτσι ώστε να αναγνωρίζεται από τον compiler του προγράμματος και να μπορεί να τρέξει. Σε αυτόν λοιπόν τον κώδικα έχουν δημιουργηθεί όλα τα κείμενα, χρώματα, διακόπτες και τα led που παρουσιάζονται μέσα στην σελίδα. Επίσης φαίνεται και ένας αριθμός που δείχνει την θερμοκρασία του boiler και ο οποίος ανανεώνεται κάθε φορά που έχω ρυθμίσει να κάνει refresh η ιστοσελίδα μου και συγκεκριμένα κάθε 15 δευτερόλεπτα έτσι ώστε να είναι και λειτουργική αλλά να παρέχει και αρκετά μεγάλη ακρίβεια ταυτόχρονα. Τέλος να αναφέρω ότι η σελίδα έχει προγραμματιστεί με τέτοιο τρόπο ώστε να μπορεί να εμφανίζεται σε όλους του browsers με τον ίδιο τρόπο όπως επίσης και να προσαρμόζεται στις διαστάσεις κινητών τηλεφώνων (smartphones).

Προσβασιμότητα

Όπως είναι λογικό δεν είναι καλή λύση να περιοριστούμε μόνο σε τοπικό επίπεδο δηλαδή μόνο στο δίκτυο της οικίας μας μέσω LAN (Local Area Network) γιατί αυτό μας περιορίζει σε σημαντικό βαθμό. Στόχος μου είναι να μπορούμε να έχουμε πρόσβαση στον server-arduino και να μπορούμε να παρακολουθούμε τις τιμές και να παρεμβαίνουμε όπου και όποτε εμείς θεωρούμε πως χρειάζεται. Χρειαζόμαστε δηλαδή μεγαλύτερη προσβασιμότητα από αυτήν που μας παρέχει απλά ένα τοπικό δίκτυο καθώς για να μπορέσουμε να συνδεθούμε θα πρέπει η συσκευή μας να είναι συνδεδεμένη στο ίδιο δίκτυο που είναι συνδεδεμένο και το arduino μας. Οπότε η λύση για αυτό το θέμα δεν είναι τίποτε άλλο παρά το άνοιγμα θύρας στο router μας δηλαδή το λεγόμενο port forwarding.

Αρχικά το NAT (Network Address Translation, δηλαδή **Μετάφραση Διευθύνσεων Δικτύου**) αναπτύχθηκε σαν ένας τρόπος για να αντιμετωπιστεί η **έλλειψη διευθύνσεων IP (IPv4)**. Έχει γίνει πλέον καθιερωμένο χαρακτηριστικό σε όλους τους δρομολογητές (routers). Τα περισσότερα συστήματα χρησιμοποιούν το NAT για να δώσουν την δυνατότητα σε πολλούς χρήστες σε ένα ιδιωτικό δίκτυο να έχουν πρόσβαση στο διαδίκτυο χρησιμοποιώντας **μια μοναδική IP**.

Ένα **ιδιωτικό** δίκτυο σε μια τυπική ρύθμισή του, χρησιμοποιεί "ιδιωτικές" διευθύνσεις IP όπως 192.168.x.x ή 10.x.x.x και ο δρομολογητής (router) συνήθως έχει την IP 192.168.1.1.

Ο δρομολογητής ωστόσο, είναι συνδεδεμένος και στο **διαδίκτυο** χρησιμοποιώντας μία και μοναδική δημόσια IP διεύθυνση.

Η κίνηση διεκπεραιώνεται από το ιδιωτικό δίκτυο προς το διαδίκτυο μέσω του δρομολογητή και η διεύθυνση αποστολέα κάθε πακέτου μεταφράζεται από **ιδιωτική** σε **δημόσια**. Ο δρομολογητής καταγράφει τις βασικές πληροφορίες για κάθε ενεργή σύνδεση (ιδιαίτερα την διεύθυνση προορισμού και την θύρα).

Όταν επιστρέφεται απάντηση από το διαδίκτυο στον δρομολογητή, ο τελευταίος χρησιμοποιεί το ιστορικό καταγραφής κατά την διάρκεια της εξερχόμενης επικοινωνίας για να αποφασίσει σε ποιο σύστημα του εσωτερικού ιδιωτικού δικτύου θα προωθήσει την απάντηση.

Για ένα σύστημα που βρίσκεται στο διαδίκτυο, ο δρομολογητής παρουσιάζεται σαν να είναι αυτός ο αποστολέας και ο αποδέκτης της διαδικτυακής κίνησης.

Μία από τις μορφές / τύπους NAT που χρησιμοποιείται ευρέως, είναι το **NAPT** (Port-level NAT). Το NAPT μεταφράζει ένα σύνολο ιδιωτικών IP διευθύνσεων σε μία δημόσια IP διεύθυνση αλλά σε **διαφορετικές πόρτες (Ports)**.

Μία **πόρτα (port)** είναι ένα εικονικό μονοπάτι από το οποίο γίνεται δρομολόγηση δεδομένων προς την συσκευή / εφαρμογή που έχει ρυθμιστεί να την χρησιμοποιεί. Για να γίνουν πιο κατανοητά τα παραπάνω, δείτε το παρακάτω σχήμα:



Για παράδειγμα, βλέπουμε στην παραπάνω εικόνα ότι η **κάμερα** με ιδιωτική διεύθυνση IP "ακούει" εισερχόμενα http requests στην πόρτα **8080**. Αυτό σημαίνει ότι όταν ένα πακέτο (που έχει σταλεί στη διεύθυνση **94.75.154.23:8080**) φτάσει στον **δρομολογητή με δημόσια IP 94.75.154.23**, αυτός θα **δρομολογήσει** το πακέτο στην πόρτα **8080**, δηλαδή στη συσκευή με ιδιωτική IP διεύθυνση **192.168.1.10** που έχει ρυθμιστεί να "ακούει" στην πόρτα **8080**.

Συνοπτικά, η διαδικασία αυτή υλοποιείται επειδή ο αποστολέας του πακέτου **δεν είναι σε θέση να γνωρίζει** την εσωτερική / ιδιωτική διεύθυνση της συσκευής έτσι ώστε να δρομολογήσει το πακέτο απευθείας σε αυτήν και γι' αυτό χρησιμοποιείται το **NAPT**. Το NAPT επίσης χρησιμοποιείται πολλές φορές για λόγους ασφαλείας.

Πόρτες (Ports)

- Οι πόρτες (ports) είναι ένας **16-bit αριθμός**. Αυτό σημαίνει ότι το εύρος τους είναι από **1** έως **65535**.
- Χωρίζονται σε **TCP** και **UDP** πόρτες ανάλογα με την κεφαλίδα του πακέτου που καθορίζεται από την εφαρμογή που δημιουργεί το πακέτο.
- Οι πόρτες που χρησιμοποιεί κάθε εφαρμογή ορίζονται συνήθως από τον δημιουργό της εφαρμογής.
- Κάθε πόρτα μπορεί να χρησιμοποιηθεί μόνο από μία εφαρμογή κάθε χρονική στιγμή.
- Οι πόρτες από **0** έως **1023** έχουν οριστεί από την **IANA (Internet Assigned Numbers Authority)** ως πόρτες που εξυπηρετούν διαδικασίες συστήματος και παρέχουν υπηρεσίες δικτύου ευρέως διαδεδομένες.

Παρακάτω παρουσιάζεται σχετικός πίνακας με κάποιες από αυτές:

PORT	PROTOCOL	ΠΕΡΙΓΡΑΦΗ
1	TCP & UDP	TCP Port Service Multiplexer (TCPMUX)
20	TCP & UDP	FTP Data Transfer
22	TCP & UDP	Secure Shell (SSH) - used for secure logins, file transfers (scp, sftp) and port forwarding
25	TCP	Simple Mail Transfer Protocol (SMTP) - Used for e-mail routing between mail servers
53	TCP & UDP	Domain Name System
80	TCP	Hypertext Transfer Protocol (HTTP)
110	TCP	Post Office Protocol v3 (POP3)
118	TCP & UDP	SQL (Structured Query Language) Services
158	TCP & UDP	SQL Service
194	TCP & UDP	Internet Relay Chat (IRC)
220	TCP & UDP	Internet Message Access Protocol (IMAP), version 3
389	TCP & UDP	Lightweight Directory Access Protocol (LDAP)
443	TCP	HTTPS (Hypertext Transfer Protocol over SSL/TLS)
554	TCP & UDP	Real Time Streaming Protocol (RTSP)
587	TCP	e-mail message submission (SMTP)
843	TCP	Adobe Flash (Unofficial)

Port Forwarding

Το Port Forwarding είναι η διαδικασία που πρέπει να γίνει για να είναι ένας υπολογιστής που βρίσκεται πίσω από έναν δρομολογητή (Router), προσβάσιμος από άλλους υπολογιστές στο διαδίκτυο. Η διαδικασία αυτή γίνεται σε όλους τους δρομολογητές με τον ίδιο τρόπο σε 3 απλά βήματα:

- 1) Ρυθμίζουμε την **εφαρμογή** που θέλουμε, να "ακούει" στην συγκεκριμένη πόρτα (στο παράδειγμά μας είναι η 8080).
- 2) Ρυθμίζουμε το **Router** να επιτρέπει την κίνηση στη συγκεκριμένη πόρτα και να την δρομολογεί στην συσκευή που θέλουμε, δηλώνοντας **external** και **internal port** (στην περίπτωση απλών δικτύων και εφαρμογών που εξυπηρετούμε, αυτές συμπίπτουν πάντα. Στο παράδειγμά μας, θα είναι **internal port: 8080**, **external port: 8080** για την ιδιωτική / εσωτερική IP **192.168.1.10** της κάμερας στην οποία θέλουμε να δρομολογείται η κίνηση.
- 3) Κάνουμε **reboot** τον δρομολογητή και την συσκευή (πχ κάμερα).

Ορίζουμε σταθερή IP στον υπολογιστή μας

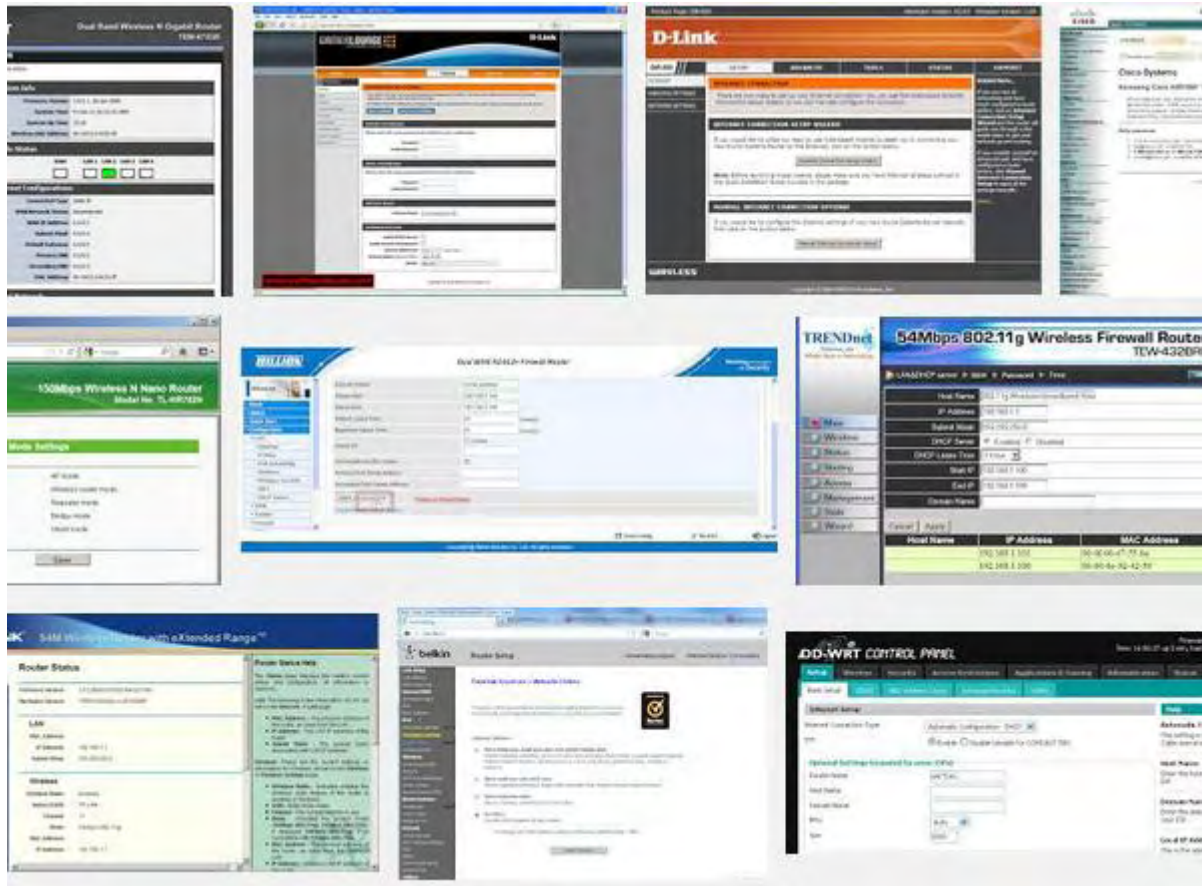
Για να λειτουργήσει το άνοιγμα θύρας για τις εφαρμογές του υπολογιστή μας, είναι απαραίτητο να έχουμε ορίσει στο λειτουργικό σύστημα μια σταθερή IP για το τοπικό δίκτυο, και όχι να παίρνει αυτόματα IP από τον DHCP Server.

Ο λόγος είναι πως το NAT στα περισσότερα τα router θα κάνει port forwarding μόνο για μία συγκεκριμένη IP στο δίκτυο, όχι για όλες τις συνδεδεμένες συσκευές.

Το άνοιγμα θύρας στο router εξ' ορισμού μειώνει την ασφάλεια του δικτύου. Θα ήταν παράλογο λοιπόν το port forwarding να ισχύει για όλους τους υπολογιστές του δικτύου.

Το άνοιγμα θύρας στο router

Διαφορετικά router συνήθως έχουν εντελώς διαφορετικό διαχειριστικό σύστημα.

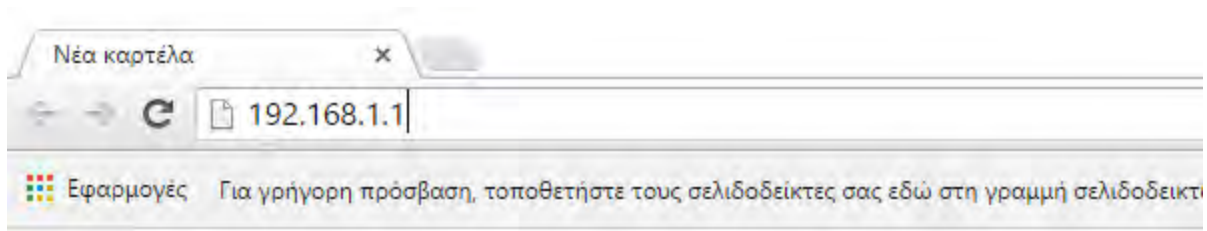


Συχνά, μάλιστα, έχουν διαφορετική ονομασία για το άνοιγμα θύρας: Port Forwarding, Address Translation, NAT, NAT table, virtual server, και άλλες.

Είσοδος στο web interface

Όταν ορίσαμε τη σταθερή IP στον υπολογιστή μας, βρήκαμε τη διεύθυνση IP της προεπιλεγμένης πύλης, που σε ένα οικιακό δίκτυο είναι η διεύθυνση IP του Router. Συνήθως είναι 192.168.1.1 ή 192.168.0.1, αλλά μπορεί επίσης να είναι 192.168.0.254 ή και 10.0.0.xxx.

Γράφουμε αυτή τη διεύθυνση στον browser της επιλογής μας.

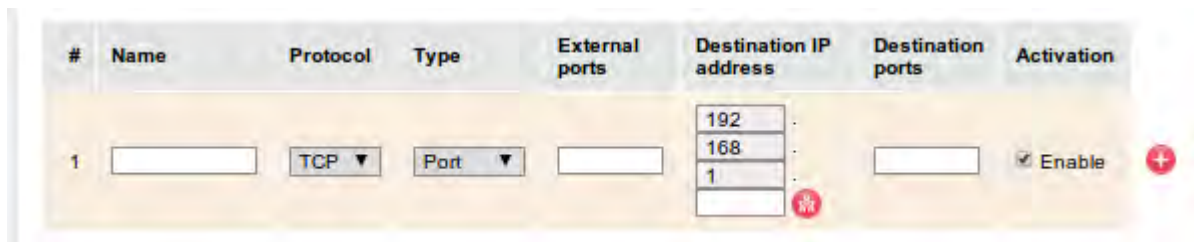


Η συντριπτική πλειοψηφία των router θα μας ζητήσει ένα username και ένα Password για να συνδεθούμε.

Εύρεση της ενότητας για το άνοιγμα θύρας

Εδώ είναι που τα πράγματα δυσκολεύουν όσον αφορά το άνοιγμα θύρας, το να βρούμε πώς ονομάζεται στο δικό μας router. Μπορεί να το βρούμε στο Network -> NAT -> Port translation.

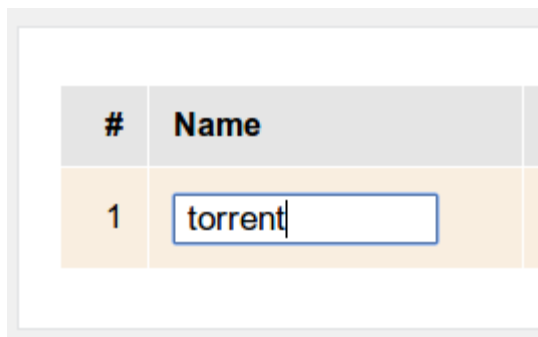
Σε κάθε περίπτωση, ψάχνουμε μια ενότητα στην οποία να μπορούμε να ορίσουμε πρωτόκολλο, external και destination port, και Destination IP address (καμιά φορά αναφέρεται και σαν Server IP address).



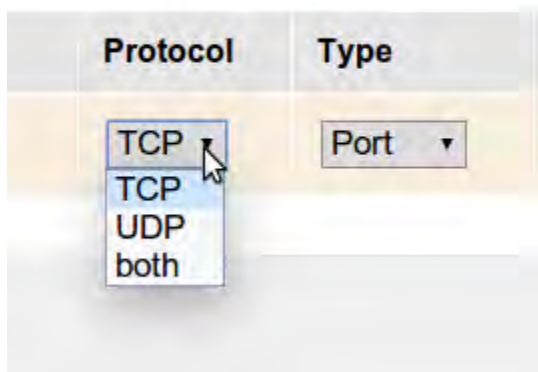
Δημιουργία κανόνα για το άνοιγμα θύρας

Έχοντας βρει την κατάλληλη ενότητα, αρκεί απλά να φτιάξουμε ένα κανόνα για το άνοιγμα θύρας.

Αρχικά ορίζουμε ένα όνομα για τον κανόνα. Αυτό το όνομα είναι για δική μας χρήση, ώστε να γνωρίζουμε για ποια εφαρμογή κάναμε το port forwarding.



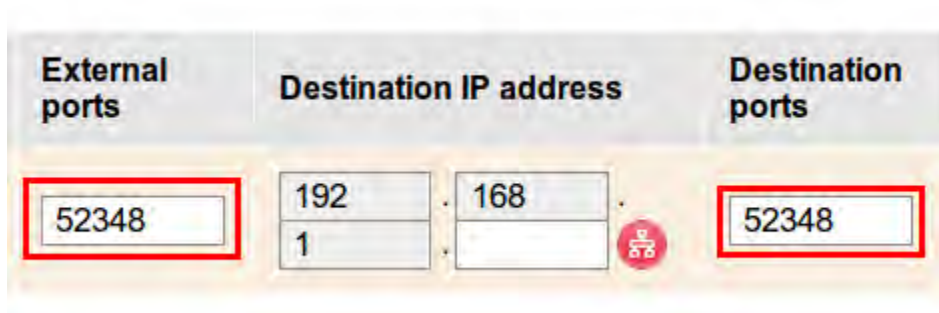
Στο πρωτόκολλο υπάρχουν συνήθως τρεις επιλογές: TCP, UDP ή και τα δύο μαζί. Το ποια είναι η σωστή επιλογή εξαρτάται από την εφαρμογή για την οποία χρειαζόμαστε το άνοιγμα θύρας.



Για παράδειγμα, οι torrent client όπως το utorrent/μutorrent συνήθως θέλουν και τα δύο μαζί, TCP και UDP, ενώ πχ ένας FTP Server θέλει μόνο TCP.

Ορισμένα router έχουν μόνο επιλογή TCP ή UDP, όχι όμως και για τα δύο. Σε αυτά τα router, αν χρειαζόμαστε και τα δύο πρωτόκολλα, θα χρειαστεί να δημιουργήσουμε δύο κανόνες, έναν για το TCP και έναν ξεχωριστό για το UDP.

Στις ports βάζουμε την ίδια θύρα για την external και τη destination port. Επειδή ορισμένες ports είναι κρατημένες από το σύστημα ή για συγκεκριμένες εφαρμογές, καλό είναι να επιλέξουμε έναν αριθμό μεταξύ του 50000 και του 65536, που είναι και το μέγιστο.



Τέλος, στο destination IP address βάζουμε τη σταθερή IP που ορίσαμε στον υπολογιστή μας.

Έχοντας συμπληρώσει όλα τα πεδία, αποθηκεύουμε το νέο κανόνα.

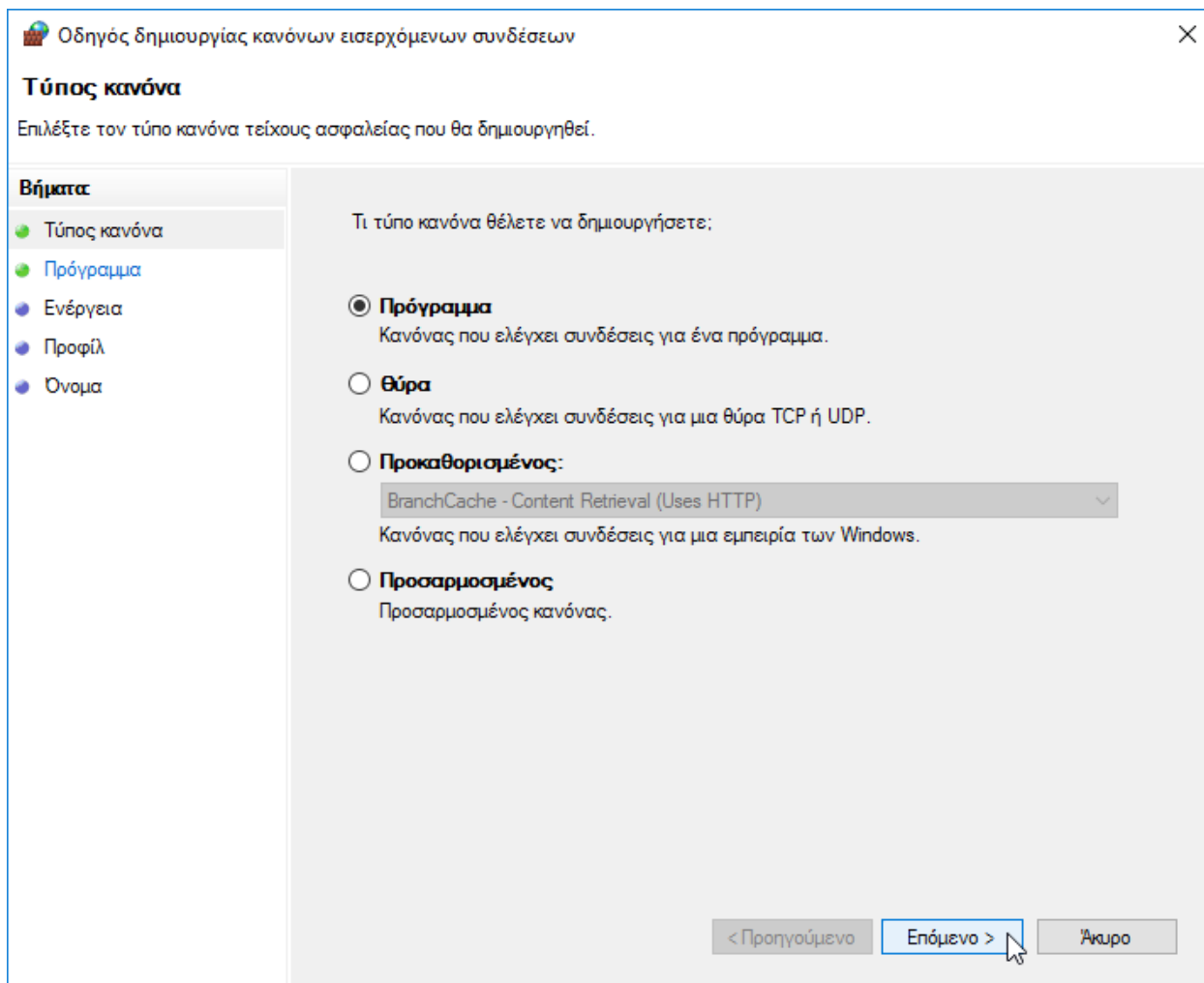
Στα περισσότερα router το port forwarding λειτουργεί άμεσα. Σε κάποια όμως ίσως χρειαστεί επανεκκίνηση του router από το web interface, ή να το κλείσουμε για 10 δευτερόλεπτα και να το ξανανοίξουμε.

Άνοιγμα θύρας στο firewall των Windows

Για να είναι πλήρες το άνοιγμα θύρας, δεν αρκεί να κάνουμε τη διαδικασία μόνο στο router, πρέπει να την κάνουμε και στο firewall των Windows.

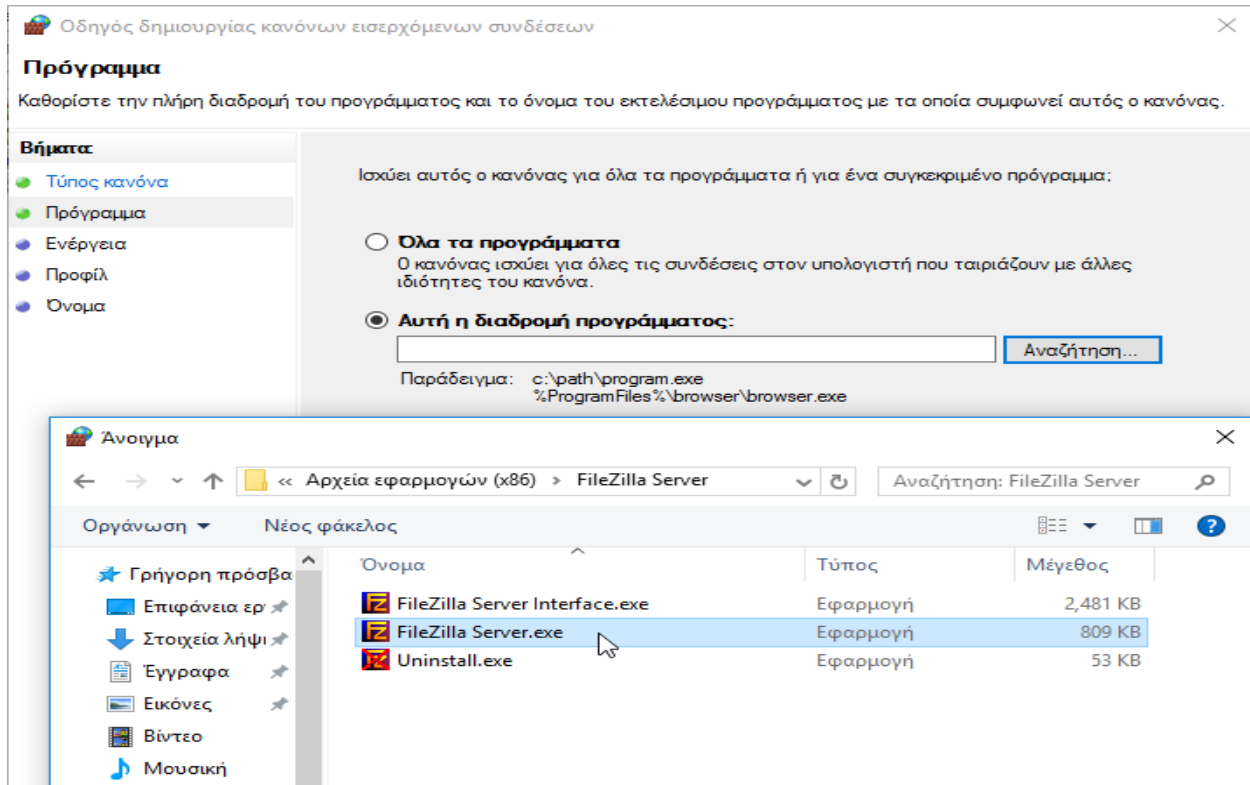
Προφανώς αν χρησιμοποιούμε κάποιο τρίτο firewall, είτε αυτόνομο είτε ενσωματωμένο σε μια σουίτα Internet Security, θα πρέπει να ρυθμίσουμε εκεί το port forwarding.

Η ευκολότερη μέθοδος για το άνοιγμα θύρας στο Windows Firewall είναι να αφήσουμε την προεπιλογή "Πρόγραμμα"

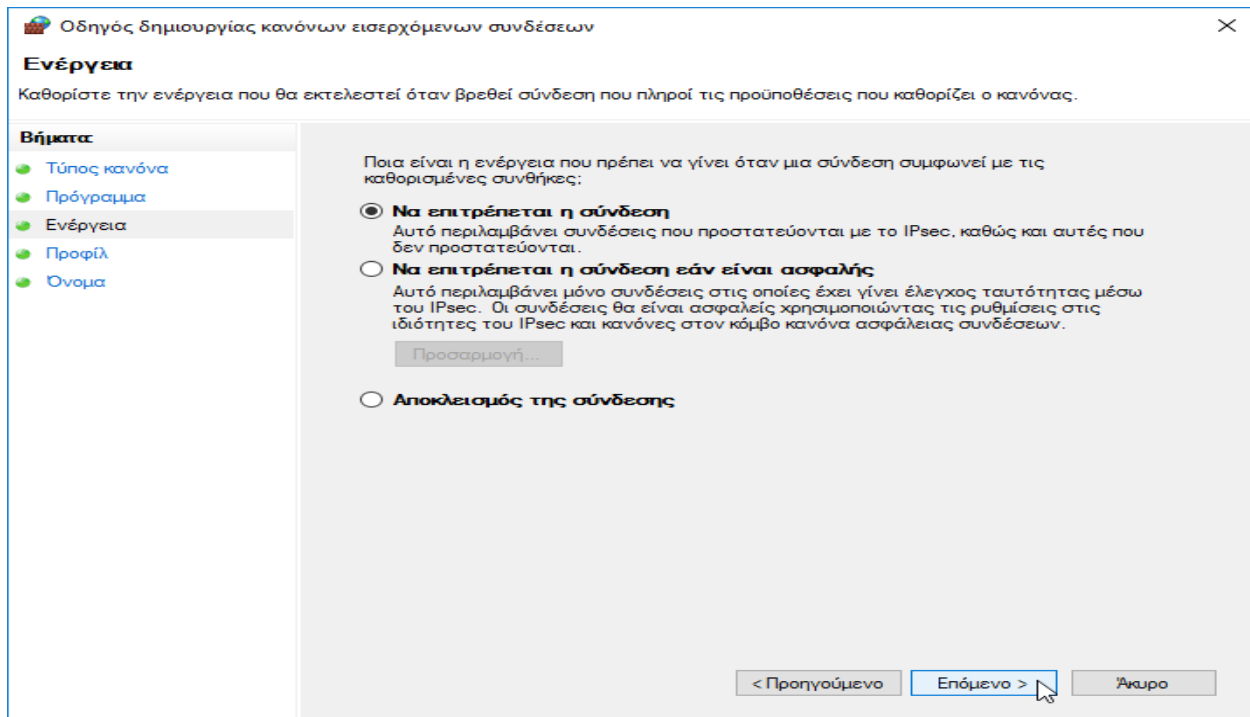


Κάνουμε κλικ στην Αναζήτηση και βρίσκουμε το εκτελέσιμο αρχείο του προγράμματος για το οποίο θέλουμε το άνοιγμα θύρας.

Τα περισσότερα προγράμματα βρίσκονται στο C:\Αρχεία εφαρμογών ή στο C:\Αρχεία εφαρμογών (x86). Στο παράδειγμα επιλέγουμε τον FileZilla Server.



Στην επόμενη οθόνη αφήνουμε το "Να επιτρέπεται η σύνδεση".



Μπορούμε να ορίσουμε ο κανόνας να ισχύει μόνο σε ιδιωτικό δίκτυο, όχι πχ αν έχουμε συνδεθεί σε κάποιο δημόσιο WiFi.

Οδηγός δημιουργίας κανόνων εισερχόμενων συνδέσεων

Προφίλ

Καθορίστε τα προφίλ για τα οποία ισχύει αυτός ο κανόνας.

Βήματα

- Τύπος κανόνα
- Πρόγραμμα
- Ενέργεια
- Προφίλ
- Όνομα

Πότε ισχύει αυτός ο κανόνας:

Τομέας
Ισχύει όταν ένας υπολογιστής είναι συνδεδεμένος στον εταιρικό τομέα του.

Ιδιωτικό
Ισχύει όταν ένας υπολογιστής είναι συνδεδεμένος με μια ιδιωτική θέση δικτύου, όπως οικιακό ή χώρο εργασίας.

Δημόσιο
Ισχύει όταν ένας υπολογιστής είναι συνδεδεμένος με δημόσια θέση δικτύου.

< Προηγούμενο Επόμενο > Άκυρο

Τέλος, βάζουμε ένα όνομα για τον κανόνα.

Οδηγός δημιουργίας κανόνων εισερχόμενων συνδέσεων

Όνομα

Καθορίστε το όνομα και την περιγραφή αυτού του κανόνα.

Βήματα

- Τύπος κανόνα
- Πρόγραμμα
- Ενέργεια
- Προφίλ
- Όνομα

Όνομα:
Filezilla Server

Περιγραφή (προαιρετικό):

< Προηγούμενο Τέλος Άκυρο

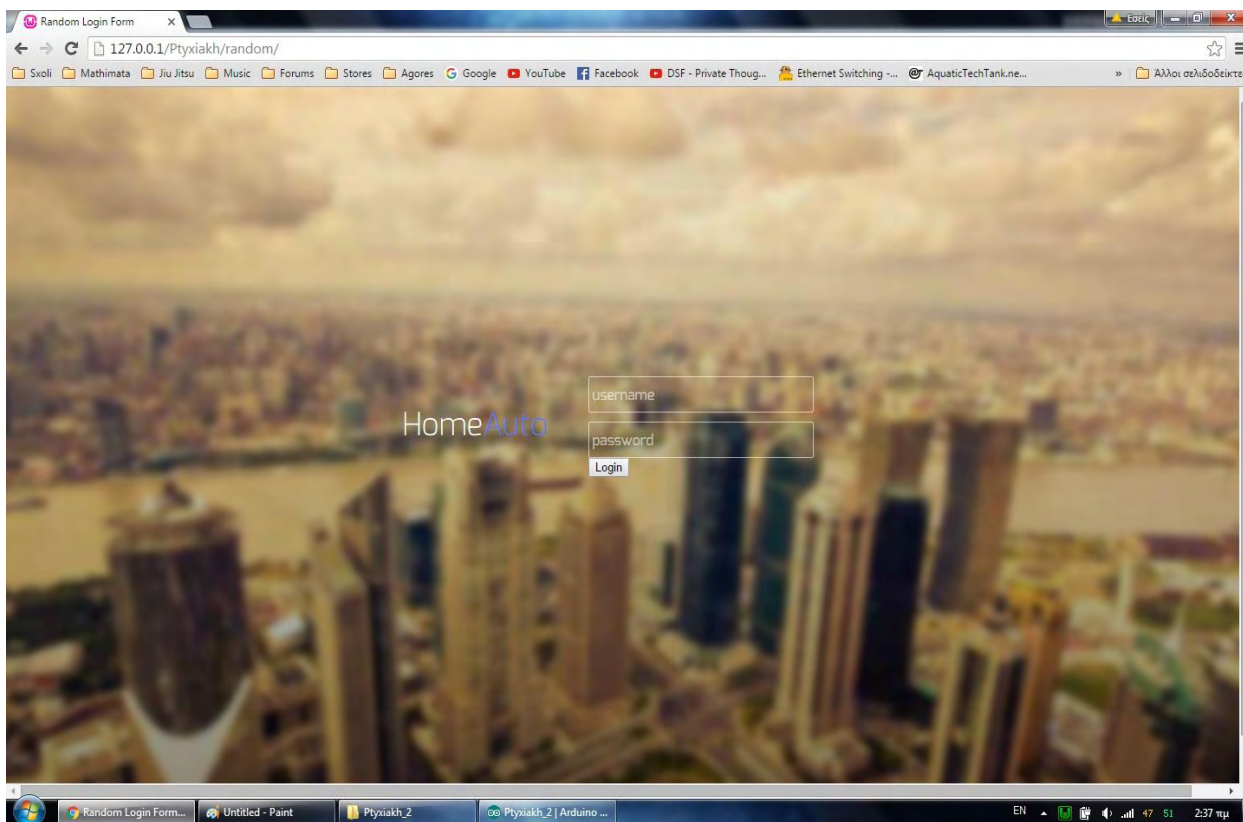
Προφανώς αυτή τη διαδικασία θα την κάνουμε μόνο για προγράμματα που εμπιστευόμαστε, όπως και το άνοιγμα θύρας γενικότερα. Θεωρητικά, το Port Forwarding έχει ολοκληρωθεί.

Ασφάλεια

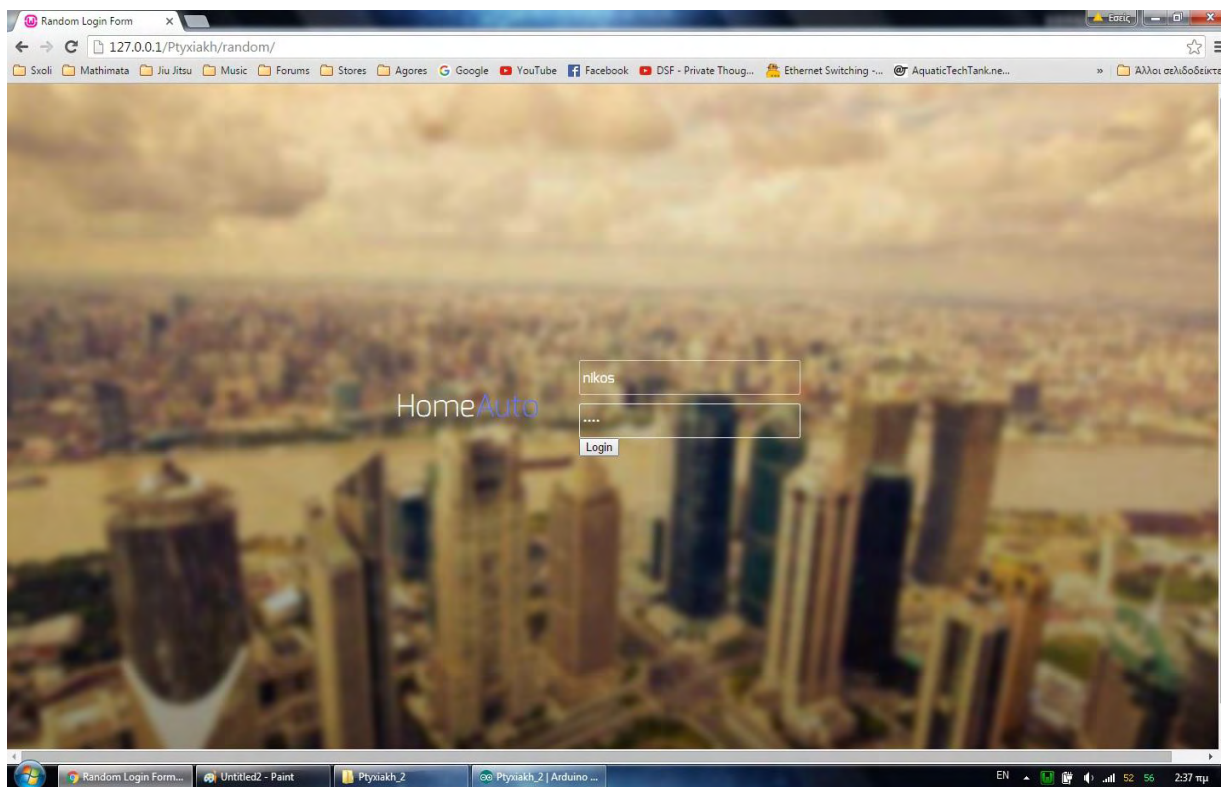
Αφού λοιπόν λύσαμε το πρόβλημα της προσβασιμότητας δημιουργείται ταυτόχρονα και ένα νέο πρόβλημα. Αυτό το πρόβλημα είναι η ασφάλεια που έχει το δίκτυό μας και κατά συνέπεια η εφαρμογή μας. Με το Port Forwarding που έκανα προηγουμένως μπορεί πλέον ο καθένας μαθαίνοντας με κάποιον τρόπο την IP μου να έχει αυτόματα πρόσβαση και στην ιστοσελίδα του server μου αποκτώντας αμέσως πρόσβαση στις λειτουργίες της οικίας μου. Αυτό όπως καταλαβαίνουμε είναι επικίνδυνο και σε καμία περίπτωση δεν πρέπει να αμεληθεί.

Για αυτόν τον λόγο λοιπόν δημιούργησα μία σελίδα η οποία εμφανίζεται πρώτη, και ο χρήστης που προσπαθεί να συνδεθεί στον server μου θα πρέπει να δώσει το σωστό όνομα χρήστη (username) και τον κωδικό (password) που έχω ορίσει εγώ, για να μπορέσει να έχει πρόσβαση στο control interface της εφαρμογής. Με αυτόν τον τρόπο προφυλάσσω την εφαρμογή από χρήστες που θελήσουν παράνομα να έχουν πρόσβαση στον server μου.

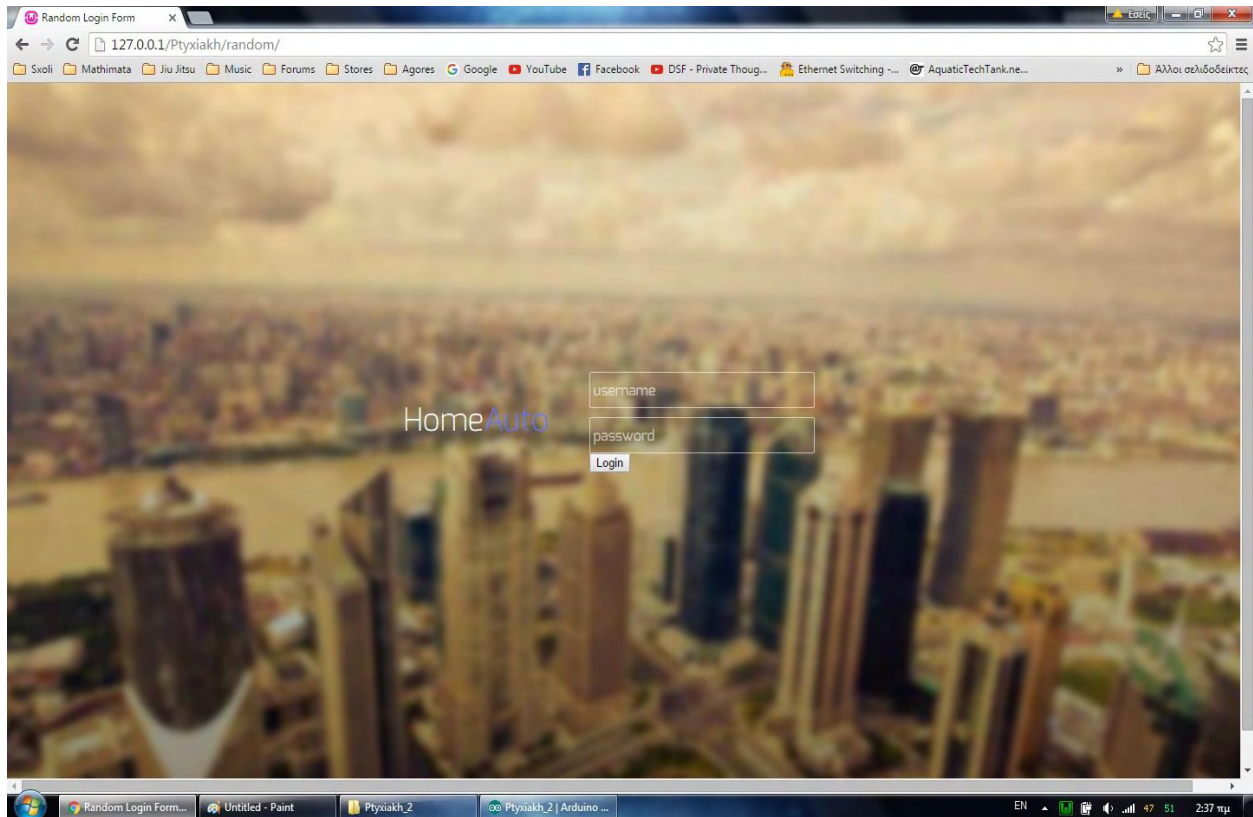
Η φόρμα σύνδεσης χρήστη είναι η εξής που φαίνεται παρακάτω:



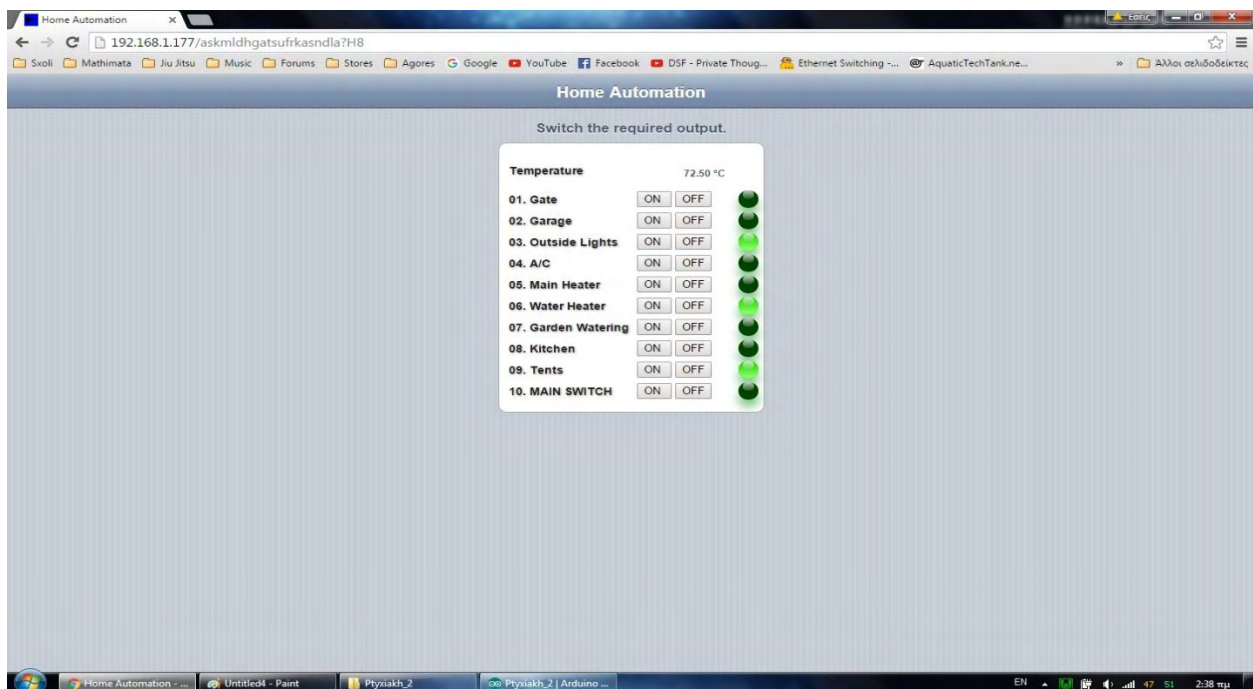
Συμπληρώνοντας τα στοιχεία



Αν δεν συμπληρωθούν σωστά τα στοιχεία που έχω ορίσει τότε παραμένει προφανώς στην ίδια σελίδα.

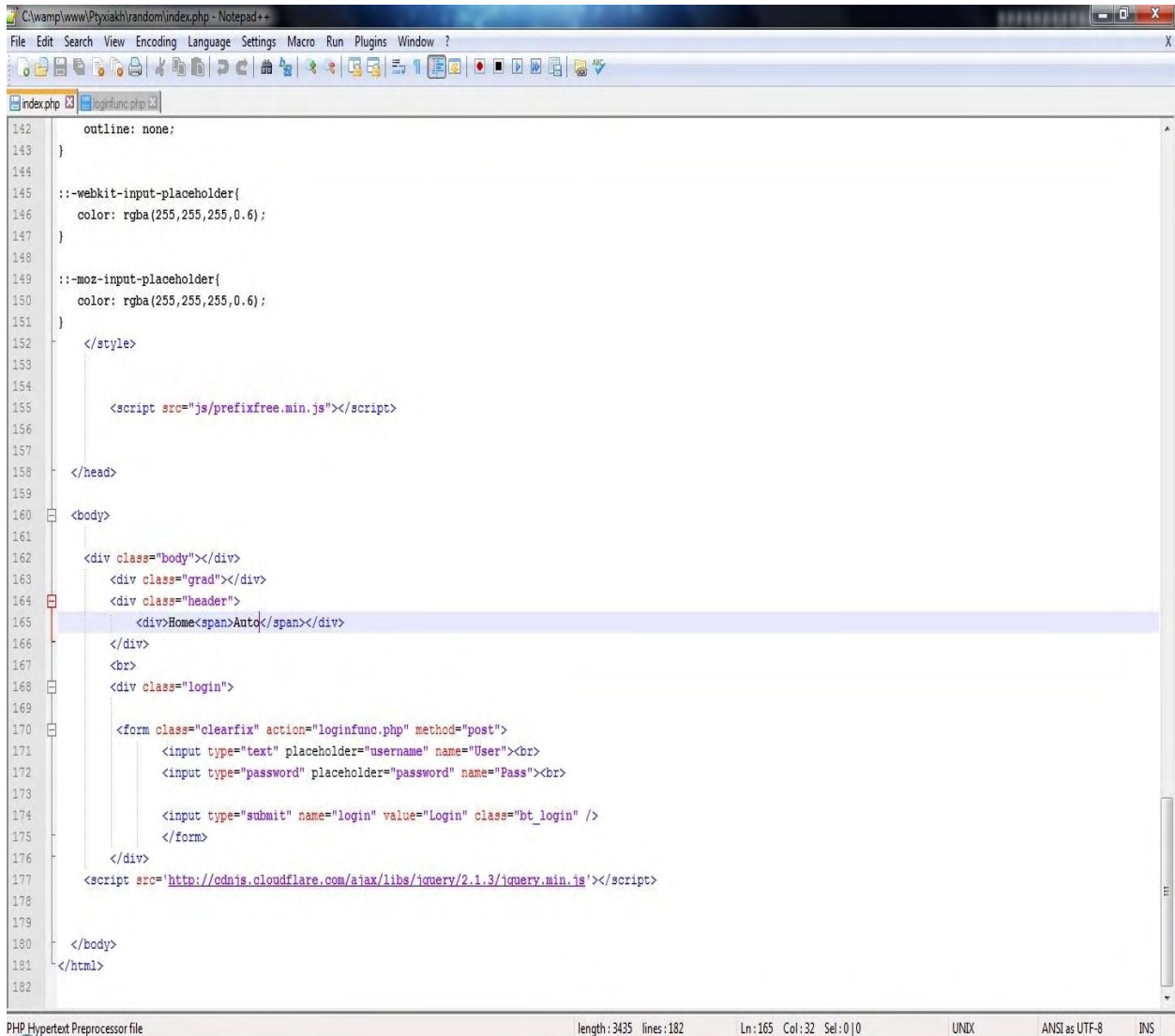


Αν τα στοιχεία είναι τα σωστά τότε και μόνο έχουμε πρόσβαση



Για την ιστοσελίδα αυτήν χρησιμοποίησα τον Wamp server για να μπορώ να την φορτώνω από τον τοπικό μου server όπου την έχω αποθηκευμένη..

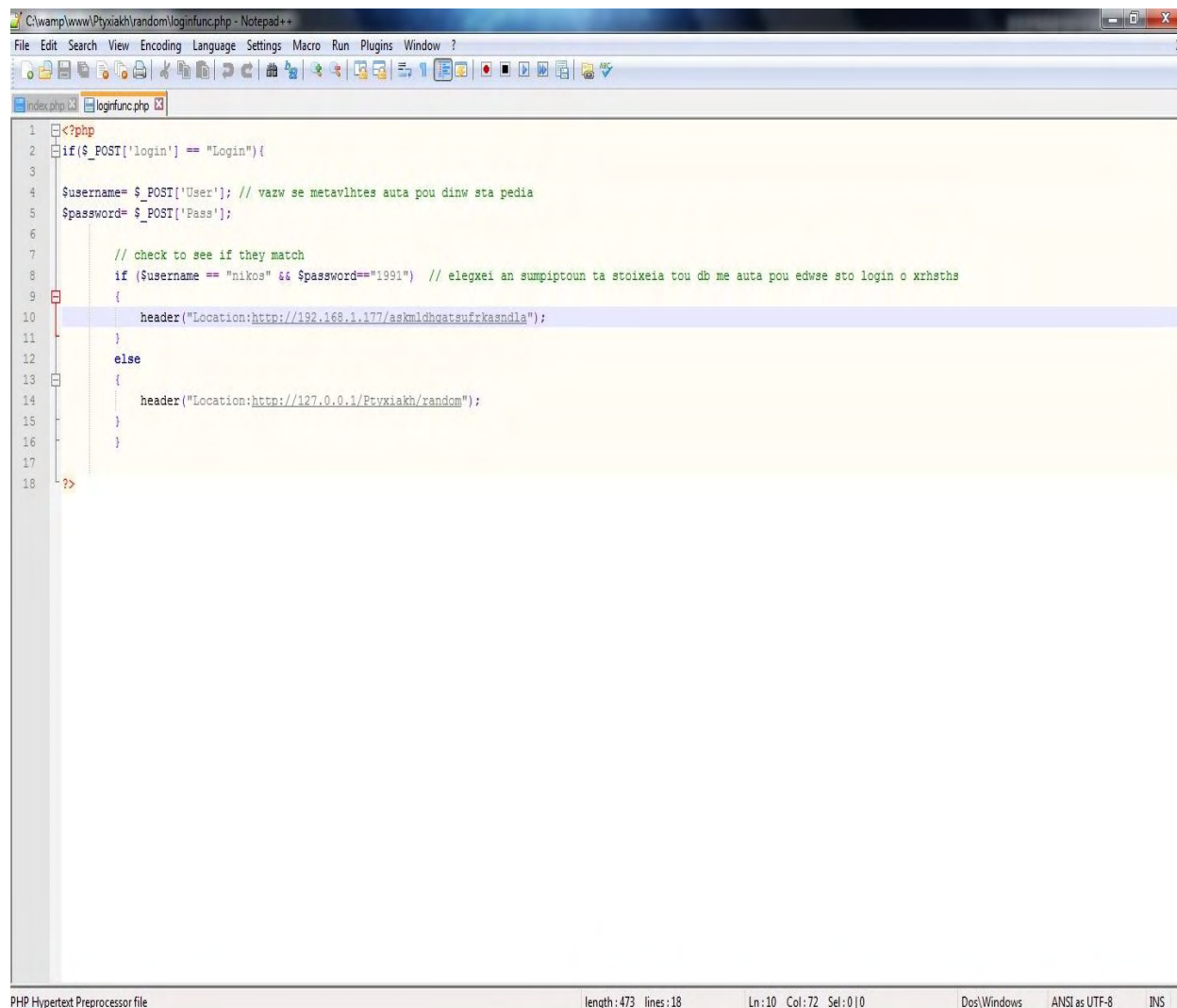
Παρακάτω φαίνεται ο html κώδικας που έγραψα για την σελίδα



```
142     outline: none;
143 }
144
145 ::-webkit-input-placeholder{
146     color: rgba(255,255,255,0.6);
147 }
148
149 ::-moz-input-placeholder{
150     color: rgba(255,255,255,0.6);
151 }
152 </style>
153
154
155     <script src="js/prefixfree.min.js"></script>
156
157
158 </head>
159
160 <body>
161
162     <div class="body"></div>
163     <div class="grad"></div>
164     <div class="header">
165         <div>Home<span>Autq</span></div>
166     </div>
167     <br>
168     <div class="login">
169
170         <form class="clearfix" action="loginfunc.php" method="post">
171             <input type="text" placeholder="username" name="User"><br>
172             <input type="password" placeholder="password" name="Pass"><br>
173
174             <input type="submit" name="login" value="Login" class="bt_login" />
175         </form>
176     </div>
177     <script src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>
178
179
180 </body>
181 </html>
182
```

PHP Hypertext Preprocessor file length: 3435 lines: 182 Ln: 165 Col: 32 Sel: 0 | 0 UNIX ANSI as UTF-8 INS

Παρακάτω φαίνεται ο κώδικας php που έγραψα για την σελίδα και ο οποίος είναι υπεύθυνος για τον έλεγχο των στοιχείων που δίνει ο χρήστης.



```
1 <?php
2 if($_POST['login'] == "Login"){
3
4 $username= $_POST['User']; // vazw se metavlhtes auta pou dinw sta pedia
5 $password= $_POST['Pass'];
6
7     // check to see if they match
8     if ($username == "nikos" && $password=="1991") // elegxei an sumptoun ta stoixeia tou db me auta pou edwse sto login o xrhsths
9     {
10         header("Location:http://192.168.1.177/askmidhoatsufrkasndla");
11     }
12     else
13     {
14         header("Location:http://127.0.0.1/Prvxiakh/random");
15     }
16 }
17
18 ?>
```

PHP Hypertext Preprocessor file length: 473 lines: 18 Ln: 10 Col: 72 Sel: 0 | 0 Dos: Windows ANSI as UTF-8 INS

Η εφαρμογή στην οικία

Αφού λοιπόν δημιούργησα την εφαρμογή το επόμενο βήμα είναι να συνδέσω το arduino μαζί με το Ethernet Shield με το Ethernet καλώδιο στο ίντερνετ και να του δώσω τροφοδοσία 5 Volts με από έναν μετασχηματιστή. Στην συνέχεια συνδέω με καλώδια τις εξόδους του arduino με το relay board το οποίο είναι σχεδιασμένο να ελέγχει 230 Volt και να οπλίζει με 5 Volts όσο είναι δηλαδή και το ρεύμα που δίνουν οι εξόδοι της πλακέτας. Με αυτόν τον τρόπο θα μπορώ να ελέγξω οποιαδήποτε ηλεκτρική συσκευή της οικίας μου επιθυμώ.

Έχοντας λοιπόν κάνει τις απαραίτητες συνδέσεις το επόμενο βήμα είναι να εγκαταστήσω τους διάφορους σένσορες που περιέγραψα προηγουμένως. Δηλαδή τον αισθητήρα θερμοκρασίας στο Boiler, τον αισθητήρα βροχής στο μπαλκόνι, τους αισθητήρες κίνησης σε κάποια συγκεκριμένα δωμάτια και τους αισθητήρες υγρασίας στον κήπο. Αυτοί οι σέσορες είναι προγραμματισμένοι να ελέγχουν κάποιες τιμές που έχω ορίσει εγώ.

Πιο συγκεκριμένα ο αισθητήρας θερμοκρασίας του boiler μου δείχνει την θερμοκρασία του νερού εκείνη την στιγμή και το έχω προγραμματίσει έτσι ώστε όταν είναι σε λειτουργία να σβήνει αυτόματα με το που φτάσει την συγκεκριμένη θερμοκρασία των 75 βαθμών Κελσίου έτσι ώστε να αποφεύγεται η άσκοπη χρήση της συσκευής ειδικά όταν η συγκεκριμένη είναι από αυτές με το μεγαλύτερο κόστος ανά ώρα λειτουργίας.

Ο αισθητήρας βροχής στο μπαλκόνι είναι ρυθμισμένος να επιστρέφει μία τιμή και να καταλαβαίνει από μόνο του το σύστημα ότι εκείνη την στιγμή αρχίζει να βρέχει έτσι ώστε αυτόματα να ανοίξει ή να κλείσει τις τέντες ή και τα ηλεκτρικά στόρια ακόμα αν υπάρχουν για να αποφευχθούν τυχόν ζημιές που μπορεί να προκαλέσει μία κακοκαιρία όταν δεν βρίσκεται κάποιος εκείνη την ώρα στο σπίτι.

Οι αισθητήρες κίνησης είναι εγκατεστημένοι σε κάποια δωμάτια του υπογείου όπου είναι προγραμματισμένοι με τέτοιο τρόπο ώστε να δίνεται εντολή να ανάψουν αυτόματα τα φώτα του συγκεκριμένου χώρου όταν εντοπιστεί κίνηση έτσι ώστε να αποφεύγεται και σε αυτή την περίπτωση άσκοπη χρήση ηλεκτρικής ενέργειας.

Οι αισθητήρες υγρασίας είναι εγκατεστημένοι σε κάποια σημεία στον κήπο όπου μετράνε ανά κάποιες ώρες τις τιμή της υγρασίας του εδάφους έτσι ώστε να μπορεί το σύστημα με τον προγραμματισμό που του έχω φορτώσει να δίνει εντολή για πότισμα του κήπου όταν και για όση ώρα χρειάζεται.

Με αυτούς του τρόπους λοιπόν εκτός από ευκολία υπάρχει και εξοικονόμηση ενέργειας σε μια οικία όπου οι συσκευές είναι προγραμματισμένες να απενεργοποιούνται μόνες τους όταν τελειώνει η δουλειά τους αλλά και να ενεργοποιούνται μόνο όταν εμείς το θελήσουμε και το έχουμε ανάγκη. Φυσικά όλες οι τιμές που έχω δώσει στον προγραμματισμό τις πλακέτας μπορούν πάντοτε να τροποποιηθούν έτσι ώστε να βολεύουν καλύτερα τις δικές μου προσωπικές ανάγκες ανάλογα πάντα με την οικία που γίνεται χρήση αυτού του συστήματος.

Παρακάτω φαίνεται ένας πίνακας που εμφανίζει τις διάφορες καταναλώσεις των συσκευών που χρησιμοποιούμε καθημερινά στις οικίες μας.

Καταναλώσεις Οικιακών Ηλεκτρικών Συσκευών		
ΠΕΡΙΓΡΑΦΗ ΣΥΣΚΕΥΗΣ	ΙΣΧΥΣ ΣΕ ΚΩΗ	ΚΟΣΤΟΣ ΑΝΑ ΩΡΑ ΛΕΙΤΟΥΡΓΙΑΣ
ΗΛΕΚΤΡΙΚΟΣ ΦΟΥΡΝΟΣ	2.5	0.35
ΜΕΓΑΛΟ ΜΑΤΙ ΚΟΥΖΙΝΑΣ	2	0.28
ΦΟΥΡΝΟΣ ΜΙΚΡΟΚΥΜΑΤΩΝ	1	0.14
ΗΛΕΚΤΡΙΚΟΣ ΘΕΡΜΟΣΙΦΩΝΑΣ	4	0.56
ΗΛΙΑΚΟΣ ΘΕΡΜΟΣΙΦΩΝΑΣ (ΧΕΙΜΕΡΙΝΗ ΛΕΙΤΟΥΡΓΙΑ)	4	0.56
ΗΛΕΚΤΡΙΚΟ ΣΙΔΕΡΟ	2	0.28
ΒΡΑΣΤΗΡΑΣ	2	0.28
ΚΑΦΕΤΙΕΡΑ	0.8	0.11
ΦΡΥΓΑΝΙΕΡΕΣ	0.9	0.13
ΤΟΣΤΙΕΡΑ ΜΙΚΡΗ	0.75	0.11
ΣΤΕΓΝΩΤΗΡΕΣ ΜΑΛΛΙΩΝ (ΣΕΣΟΥΑΡ)	2	0.28
ΗΛΕΚΤΡΙΚΟ ΨΥΓΕΙΟ ΠΑΛΑΙΟΥ ΤΥΠΟΥ	0.2	0.03
ΗΛΕΚΤΡΙΚΟ ΨΥΓΕΙΟ ΝΕΟΥ ΤΥΠΟΥ	0.04	0.01
ΗΛΕΚΤΡΙΚΟ ΠΛΥΝΤΗΡΙΟ ΡΟΥΧΩΝ	2.2	0.31
ΗΛΕΚΤΡΙΚΟ ΣΙΔΕΡΩΤΗΡΙΟ (ΠΡΕΣΣΑ)	2	0.28
ΠΛΥΝΤΗΡΙΟ ΠΙΑΤΩΝ	2	0.28
ΗΛΕΚΤΡΙΚΗ ΣΚΟΥΠΑ	1.50	0.21
ΤΗΛΕΟΡΑΣΗ ΝΕΟΥ ΤΥΠΟΥ 40"	0.14	0.03
ΑΕΡΟΘΕΡΜΟ	2	0.28
ΗΛΕΚΤΡΟΝΙΚΟΣ ΥΠΟΛΟΓΙΣΤΗΣ ΣΤΑΘΕΡΟΣ ΜΕ ΘΘΟΝΗ	0.22	0.03
ΦΟΡΗΤΟΣ ΗΛΕΚΤΡΟΝΙΚΟΣ ΥΠΟΛΟΓΙΣΤΗΣ (ΙΑΡΤΟΡ)	0.10	0.01
ΕΝΙΣΧΥΤΗΣ ΚΕΡΑΙΑΣ	0,03	0.005
ΡΟΥΤΕΡ (ΣΥΣΚΕΥΗ ΓΡΗΓΟΡΟΥ ΙΝΤΕΡΝΕΤ)	0.015	0.003
ΛΑΜΠΤΗΡΑΣ ΠΑΛΑΙΟΥ ΤΥΠΟΥ	0.1	0.004
ΛΑΜΠΤΗΡΑΣ ΕΞΟΙΚΟΝΟΜΗΣΗΣ ΕΝΕΡΓΕΙΑΣ (ΑΝΤΙΣΤΟΙΧΗΣ ΦΩΤΕΙΝΗΣ ΡΟΗΣ ΤΟΥ ΠΑΛΑΙΟΥ ΤΩΝ 0.1 ΚΥΥ)	0.02	0.001
ΚΛΙΜΑΤΙΣΤΙΚΟ ΙΝΒΕΡΤΕΡ (9000ΒΤII)	0.75	0.11

Οπότε όπως φαίνεται και από τον παραπάνω πίνακα μπορούμε εύκολα να καταλάβουμε ότι κάποιες ηλεκτρικές συσκευές έχουν μεγάλο κόστος λειτουργίας και θεωρώ ότι είναι καλό αφού υπάρχει η δυνατότητα να μπορούμε να τις ελέγξουμε όσο το δυνατόν περισσότερο, στην συγκεκριμένη περίπτωση ακόμα και χωρίς να βρισκόμαστε καν στην οικία.

Σύνοψη

Όπως είδαμε μία τέτοια εφαρμογή θα μπορούσε να λύσει τα χέρια πολλών ανθρώπων. Μία ιδέα η οποία ξεκίνησε σε απλή μορφή σε τοπικό δίκτυο και στη συνέχεια εξελίχθηκε να έχει πολύ μεγαλύτερη προσβασιμότητα ακόμη και ασφάλεια. Η εφαρμογή έχει ακόμα αρκετά περιθώρια βελτίωσης και αναβάθμισης των ήδη υπαρχόντων λειτουργιών της. Για παράδειγμα μία βελτίωση θα ήταν να μπορείς να δώσεις τις εντολές φωνητικά ή η εγκατάσταση μίας live κάμερας ή ακόμα και το στήσιμο ολόκληρου αυτοματοποιημένου συναγερμού για την οικία. Το arduino και γενικότερα οι microcontrollers έχουν πολύ μεγάλες δυνατότητες και πραγματικά οι συνδυασμοί και οι διαφορετικές ιδέες που μπορούν να υλοποιηθούν είναι αμέτρητες. Αφού λοιπόν έχουμε στα χέρια μας τέτοια εργαλεία είναι προς όφελός μας να μάθουμε να τα χειριζόμαστε και να τα αξιοποιούμε όσο περισσότερο γίνεται ώστε να κάνουμε εμείς οι ίδιοι τη ζωή μας πιο άνετη και πιο διασκεδαστική.

Βιβλιογραφία

1. <https://www.arduino.cc/en/Main/Products>
2. <http://stackoverflow.com/>
3. <http://www.insomnia.gr/>
4. <https://www.cosmote.gr/fixed/help-and-support/faq/terms/-/support/article/1212852>
5. <http://www.pcsteps.gr/>