



ΤΕΙ ΠΕΙΡΑΙΑ

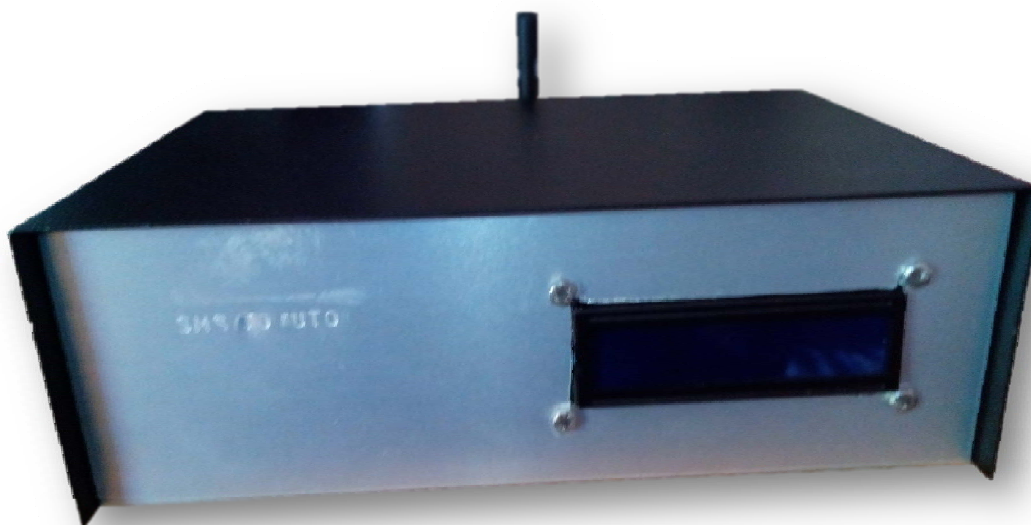
Σχολή

Αυτοματισμού

Πτυχιακή Εργασία

ΤΗΛΕ-ΕΝΗΜΕΡΩΣΗ ΜΕ ΤΗΝ ΒΟΗΘΕΙΑ ΤΗΣ ΣΥΓΧΡΟΝΗΣ
ΤΕΧΝΟΛΟΓΙΑΣ

Live update feed using latest technologies



Στέφανος Βαρδάλος

A.M. : 37442

Επιβλέπων καθηγητής

Ιωάννης Παπαχαραλάμπους

A.M. : 37524

Ι.Δ. Αγγελόπουλος, MSc., PhD.

Αθήνα 2013

Περίληψη

Επικοινωνία ορίζεται ως η διαδικασία αποστολής ενός μηνύματος από έναν πομπό σε ένα δέκτη, χρησιμοποιώντας έναν κοινό κώδικα επικοινωνίας. Στην εποχή μας που οι ρυθμοί της καθημερινότητας έχουν γίνει πολύ έντονοι, υπάρχει μεγάλη ανάγκη για άμεση επικοινωνία. Σκοπός της παρούσας πτυχιακής είναι η δημιουργία ενός μοντέλου επικοινωνίας που θα γεφυρώσει διαφορετικά πρότυπα με στόχο την διευκόλυνση τόσο του πομπού όσο και του δέκτη.

Η πτυχιακή εργασία σχεδιάστηκε με βάση το παράδειγμα της μονόπλευρης επικοινωνίας μεταξύ καθηγητών και φοιτητών. Για τον πομπό (καθηγητής) επιλέχθηκε ένας τρόπος μετάδοσης που είναι καθολικά διαδεδομένος και ανά πάσα στιγμή εύκαιρος, η τεχνολογία Short Message Service (σε συντομία SMS), η οποία επιτρέπει μέσω κινητού τηλεφώνου την αποστολή σύντομων γραπτών μηνυμάτων. Για τον δέκτη (φοιτητής) επιλέχθηκε το πρότυπο του Internet και των Social Media. Πιο συγκεκριμένα, ο δέκτης θα μπορεί να διαβάσει τα μηνύματα του πομπού από ειδικά διαμορφωμένο site αλλά και μέσα από την ευρέως διαδεδομένη σελίδα κοινωνικής δικτύωσης, Facebook.

Εδώ αυτόματα δημιουργείται το πρόβλημα της γεφύρωσης των διαφορετικών προτύπων. Το πρόβλημα αυτό θα το λύσουμε κάνοντας χρήση τελευταίων τεχνολογιών όπως, μεταξύ άλλων, της υπολογιστικής πλατφόρμας Arduino, ενός GSM Module, ενός Ethernet Module και με την των PHP και MySQL. Ο συνδυασμός και προγραμματισμός όλων αυτών θα αναλυθεί στα επόμενα κεφάλαια.

Abstract

Communication is called the procedure through which a transmitter sends a message to a receiver, using some common code for understanding each other. Nowadays that daily rhythm urges us to speed up everything we do, there is an always expanding need for instant communication. The goal of this final paper is to create a model of communication that connects different protocols for the accommodation of both the transmitter and the receiver.

This final paper is designed based on the example of the one-way communication between a teacher and his students. For the transmitter (teacher) we chose a way of transmission that is widespread and practically anytime available, the technology of Short Message Service (or SMS), that gives the ability of sending short messages using a mobile phone. For the receiver (student) we chose the Internet and furthermore the use of Social Media. More specifically, receiver will be able to read transmitter's messages from a well-formed website and through the widely used social networking page, Facebook.

Here rises the obvious problem of bridging those different protocols. This problem will be solved using the all the latest technologies, including, among others, an Arduino board, a GSM Module, an Ethernet Module and the use of PHP and MySQL. The combination and programming of those will be reviewed extensively in the following chapters.

Πίνακας περιεχομένων

Κεφάλαιο 1. Εισαγωγή	-8-
1.1 Στοχοθεσία	-8-
1.2 Ορισμοί	-9-
1.3 Πρόλογος – Εισαγωγή	-10-
Κεφάλαιο 2. Υλικά, Κατασκευή και Προγραμματισμός	-14-
2.1 Καταγραφή χρησιμοποιημένων συσκευών	-14-
2.1.1 GSM Module	-16-
2.1.2 Arduino MEGA2560	-17-
2.1.3 Ethernet Shield	-19-
2.1.4 LCD Screen Module	-20-
2.2 Συνδεσμολογία και Στάδια Κατασκευής	-22-
2.2.1 Σύνδεση του Arduino με τα υπόλοιπα εξαρτήματα	-22-
2.2.2 Λειτουργία του Arduino	-24-
2.3 Δημιουργία web server και εγκατάσταση απαραίτητων προγραμμάτων	-34-
2.3.1 Παρουσίαση του Cloud Computing και της Υπηρεσίας ~Okeanos	-34-

2.3.2 Διαχείριση υπολογιστικού συστήματος και εγκατάσταση εφαρμογών	-39-
2.3.3 Δημιουργία της ιστοσελίδας της εφαρμογής	-43-
2.3.4 Δημιουργία και λειτουργία του Facebook Application	-52-
2.3.5 Διαχείριση Domain Name	-59-
2.4 Εγκατάσταση και ολοκλήρωση κατασκευής	-64-
Κεφάλαιο 3. Εφαρμογές , Προβλήματα και Βελτιώσεις	-70-
3.1 Πηγή αναφοράς και πιθανή εφαρμογή της εργασίας	-64-
3.2 Προβλήματα που παρουσιάστηκαν	-72-
3.3 Πιθανές βελτιώσεις και παραμετροποιήσεις	-75-
Βιβλιογραφία	-77-
Παράρτημα Α΄	-80-
Παράρτημα Β΄	-91-

ΠΙΝΑΚΑΣ ΣΧΗΜΑΤΩΝ ΠΙΝΑΚΩΝ ΚΑΙ ΕΙΚΟΝΩΝ

Σχήμα 1	:	Διαδρομή της πληροφορίας	14
Σχήμα 2	:	Γενική Λειτουργία της εφαρμογής SMStoAuto	14
Σχήμα 3	:	Λειτουργία του μικροελεγκτή Arduino	24
Πίνακας 1	:	Διάγραμμα Συνδέσεων	23
Πίνακας 2	:	Διαστάσεις Εξαρτημάτων	65
Εικόνα 1	:	GSM Module TC35	16
Εικόνα 2	:	Arduino MEGA2560	18
Εικόνα 3	:	Ethernet Shield	19
Εικόνα 4	:	LCD Screen Module	21
Εικόνα 5	:	Arduino Development Environment	25
Εικόνα 6	:	Γραφική απεικόνιση του Cloud Computing	36
Εικόνα 7	:	Βασικές επιλογές δημιουργίας υπολογιστικού συστήματος	38
Εικόνα 8	:	Περιβάλλον PuTTY	39
Εικόνα 9	:	Αρχιτεκτονική LAMP	40
Εικόνα 10	:	Περιβάλλον WinSCP	42
Εικόνα 11	:	Twitter Bootstrap	45
Εικόνα 12	:	Εμφάνιση ιστοσελίδας SMStoAuto	48
Εικόνα 13	:	Περιβάλλον εργασίας του phpMyAdmin	51
Εικόνα 14	:	Πίνακας ελέγχου του Facebook Application	53
Εικόνα 15	:	Διάγραμμα ροής Server-Side Login	54
Εικόνα 16	:	Πρωτόκολλο OAuth 2.0	57

Εικόνα 17	:	Ιεραρχία και δομή DNS	60
Εικόνα 18	:	Godaddy : ο μεγαλύτερος Domain name registrar	61
Εικόνα 19	:	What's My DNS?, χρήσιμο εργαλείο για αλλαγές DNS	64
Εικόνα 20	:	Κουτί κατασκευών	66
Εικόνα 21	:	Εργαλεία παραμετροποίησης κουτιού	67
Εικόνα 22	:	Εμπρόσθιο μέρος της κατασκευής μας	68
Εικόνα 23	:	Οπίσθιο μέρος της κατασκευής μας	69

Κεφάλαιο 1

Εισαγωγή

1.1 Στοχοθεσία

Η παρούσα πτυχιακή αποσκοπεί στην κατασκευή ενός ολοκληρωμένου συστήματος που να επιτρέπει την άμεση επικοινωνία μεταξύ δύο ομάδων που ενδεχομένως είναι εξοικειωμένοι και χρησιμοποιούν διαφορετικά πρότυπα επικοινωνίας.

Κύριος στόχος είναι η διευκόλυνση και των δύο ομάδων ώστε τελικά να επιτυγχάνεται η αναγκαία επικοινωνία και ενημέρωση με τα μέσα που προτιμάει ο καθένας. Για να επιτευχθεί αυτό γίνεται χρήση των τελευταίων τεχνολογιών, αποκλειστικά ελεύθερου λογισμικού, όπως της πλατφόρμας Arduino και των έξτρα εξαρτημάτων της, αλλά και της γλώσσας προγραμματισμού php και της βάσης δεδομένων MySQL.

1.2 Ορισμοί

Arduino. Είναι μια υπολογιστική πλατφόρμα ανοιχτού κώδικα βασισμένη σε μια απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή και εισόδους/εξόδους,

AT Commands. Το σύνολο των εντολών που έχουν θεσπιστεί για την επικοινωνία με ένα GSM Modem ή ένα κινητό τηλέφωνο.

SMS. Πλέον διαδεδομένος τρόπος επικοινωνίας μεταξύ χρηστών κινητής τηλεφωνίας με τον οποίο γίνεται αποστολή μικρών, σύντομων μηνυμάτων.

GSM. Είναι το κοινό Ευρωπαϊκό ψηφιακό σύστημα κινητής τηλεφωνίας, το οποίο χαρακτηρίζει την δομή, την αρχιτεκτονική και τις ζώνες συχνοτήτων του δικτύου.

GSM Modem. Ένα Modem το οποίο με την χρήση κατάλληλης κεραίας μπορεί να συνδεθεί σε δίκτυο GSM.

Arduino Shields. Πλατφόρμες ειδικά σχεδιασμένες ώστε να μπορούν να συνδεθούν με τον Arduino και να αυξήσουν τις δυνατότητές του.

LCD. Ευρέως διαδεδομένος τύπος οθόνης η οποία βασίζεται στην τεχνολογία υγρών κρυστάλλων.

PHP. Γλώσσα προγραμματισμού που χρησιμοποιείται για την δημιουργία δυναμικών ιστοσελίδων.

MySQL. Σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων.

1.3 Πρόλογος - Εισαγωγή

Η πτυχιακή αυτή θα ασχοληθεί με διάφορες τεχνολογίες οπότε κρίνεται αναγκαίος ένας πρόλογος πάνω στις διάφορες τεχνολογίες και στην γενική δομή της κατασκευής, πριν προχωρήσουμε στην μελέτη των βημάτων που ακολουθήθηκαν.

Το ακρωνύμιο SMS προέρχεται από τις λέξεις Short Message Service ή αλλιώς, Υπηρεσία Σύντομου Μηνύματος και, είναι υπηρεσία της κινητής τηλεφωνίας. Με αυτήν ο χρήστης έχει την δυνατότητα να αποστείλει και να παραλάβει ένα σύντομο γραπτό μήνυμα στο κινητό του, το οποίο δεν μπορεί να ξεπερνάει τους 160 χαρακτήρες. Το πρώτο SMS εστάλη το 1992 από την εταιρεία κινητής τηλεφωνίας Vodafone ενώ από τότε το πρότυπο έχει εξαπλωθεί και πια χρησιμοποιείται καθημερινά από εκατομμύρια χρήστες στον κόσμο.

Για την αποστολή και την λήψη ενός τέτοιου μηνύματος απαιτείται ένα μόντεμ που να επικοινωνεί με το δίκτυο της κινητής τηλεφωνίας GSM. Σε αυτό είναι απαραίτητο να προσθέσουμε και μία κάρτα SIM (Subscriber Identity Module), η οποία διατίθεται από τον εκάστοτε πάροχο τηλεπικοινωνίας και αποτελεί την ταυτότητά μας στο δίκτυο. Οι κάρτες αυτές ποικίλουν σύμφωνα με το μέγεθός τους (full , mini , micro , nano) αλλά και τον τύπο τους (USIM, SIM) και αντίστοιχα διαφέρουν στα πρότυπα ασφαλείας, στην χωρητικότητά τους (για αποθηκευμένα νούμερα και SMS) και στα συνδεδεμένα δίκτυα.

Ο Arduino είναι μια ηλεκτρονική πλατφόρμα σχεδιασμένη γύρω από έναν μικροελεγκτή της εταιρείας Atmel, τα χαρακτηριστικά του οποίου διαφέρουν σύμφωνα με το μοντέλο του ενώ, τα σχεδιαγράμματά κατασκευής του υπάρχουν ελεύθερα στο internet ώστε να μπορεί ο καθένας να το κατασκευάσει μόνος του. Ουσιαστικά είναι ένα ολοκληρωμένο σύστημα που παρέχει το απαιτούμενο hardware και software στον οποιοδήποτε με λίγες προγραμματιστικές γνώσεις για να μπορεί να φτιάξει από απλά πράγματα όπως διακοσμητικά (π.χ. Led Cube, δηλαδή έναν κύβο από leds τα οποία δημιουργούν όμορφα φωτορυθμικά) μέχρι πιο εξελιγμένα όπως αυτόνομα ρομπότ.

Οι εκδόσεις του Arduino είναι πολλές και διαφέρουν ως προς τον τύπο του μικροελεγκτή που υποστηρίζουν και το πλήθος των δυνατοτήτων της πλακέτας τους. Γενικά σημαντικές διαφορές που πρέπει να ληφθούν υπόψη στην αρχή ενός project είναι το πλήθος των εισόδων/εξόδων που πρέπει να έχει το Arduino για να επικοινωνεί με εξωτερικά στοιχεία, καθώς και το μέγεθος μνήμης που θα χρειαστούμε για να το προγραμματίσουμε.

Πέρα από την αρχική πλακέτα, υπάρχουν διάφορες έξτρα πλακέτες οι οποίες είναι ειδικά σχεδιασμένες ώστε να “κουμπώνουν” πάνω στην πλατφόρμα του Arduino και να της προσδίδουν έξτρα δυνατότητες, όπως την εύκολη σύνδεση με κινητήρες. Αυτές οι πλακέτες επέκτασης ονομάζονται shields και στην πραγματικότητα κάνουν το εύρος των δυνατοτήτων του Arduino τεράστιο.

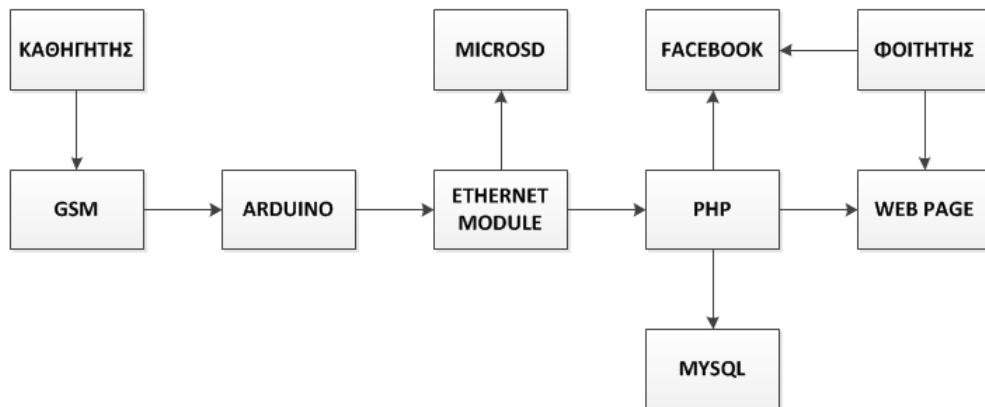
Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων, δηλαδή δεδομένων οργανωμένων σε κατάλληλους πίνακες με την δυνατότητα της εγγραφής νέων τιμών, τροποποίηση και διαγραφή των ήδη υπαρχόντων ή άλλες πιο πολύπλοκες διαδικασίες. Είναι η παγκοσμίως πιο διαδεδομένη βάση δεδομένων και ανήκει στην εταιρεία Oracle. Ο χειρισμός της παραδοσιακά γίνεται μέσω της γραμμής εντολών αλλά τώρα πια ο περισσότερος κόσμος χρησιμοποιεί εφαρμογές συμβατές με την MySQL που επιτρέπουν την γραφική αναπαράσταση και διαχείριση της βάσης δεδομένων τους. Ο προγραμματισμός της είναι απλός και περιλαμβάνει μεταξύ άλλων εντολές για την δημιουργία, προβολή, τροποποίηση και διαγραφή βάσεων, πινάκων και στοιχείων.

Η MySQL διατηρεί τα δεδομένα τα οποία η PHP αναζητά ώστε να δημιουργήσει την ιστοσελίδα που βλέπουμε στον browser μας. Η PHP είναι μια server-side γλώσσα προγραμματισμού (δηλαδή οι εντολές τις εκτελούνται στον server και όχι στον υπολογιστή του client) που χρησιμοποιείται κυρίως για την δημιουργία ιστοσελίδων και διαδικτυακών εφαρμογών και δημιουργήθηκε το 1995. Ο κώδικας της PHP περνάει από έναν διακομιστή ο οποίος παράγει το τελικό περιεχόμενο το οποίο αποστέλλεται τελικά στον browser του χρήστη. Αρκετά συχνά χρησιμοποιείται ο εξυπηρετητής Apache, ενώ η όλη εγκατάσταση μαζί με την PHP και την MySQL γίνεται σε περιβάλλον Linux (όχι αναγκαστικά βέβαια), δημιουργώντας έτσι το γνωστό περιβάλλον υποστήριξης διαδικτυακών εφαρμογών LAMP (Linux – Apache – MySQL – PHP).

Με τον όρο Social Media χαρακτηρίζουμε όλες τις εικονικές κοινότητες και δίκτυα που επιτρέπουν στους χρήστες τους την αλληλοεπίδραση μέσω της δημιουργίας και ανταλλαγής πληροφοριών και ιδεών. Με την γρήγορη εξάπλωσή τους, τα Social Media κατέχουν πλέον εξέχουσα θέση στην καθημερινότητά μας και αποτελούν σημαντικό κομμάτι της επικοινωνίας και της πληροφόρησής μας. Αν και αυτά τα δίκτυα έχουν πολλαπλασιαστεί τον τελευταίο καιρό, με καινούρια να δημιουργούνται κυριολεκτικά καθημερινά, το πιο ευρέως διαδεδομένο είναι το Facebook, το οποίο δημιουργήθηκε το 2004 και σε πρόσφατη μέτρηση βρέθηκε να αριθμεί περίπου 1,1 δισεκατομμύρια εγγεγραμμένους χρήστες.

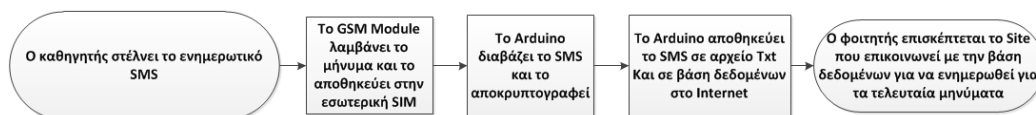
Όλα τα παραπάνω θα συνδυαστούν ώστε να δημιουργηθεί το SMStoAuto, μια ολοκληρωμένη εφαρμογή που έχει στόχο την άμεση ενημέρωση και επικοινωνία με την χρήση των τελευταίων τεχνολογιών. Η εφαρμογή θα παρέχει την δυνατότητα στον αποστολέα να μπορεί με ένα από SMS από το κινητό του τηλέφωνο να μπορεί να απευθυνθεί κατευθείαν σε όσα άτομα χρειάζεται, καθώς το SMStoAuto θα λάβει το SMS και θα το ανεβάσει σε ειδικά διαμορφωμένο site καθώς και στο Facebook, ώστε να μπορούν να ενημερωθούν από εκεί οι παραλήπτες.

Για να γίνει πιο εύκολη η ανάλυση της εφαρμογής, αρκεί να την διαχωρίσουμε στο μηχανικό κομμάτι, το οποίο περιλαμβάνει το κομμάτι του μικροελεγκτή και των εξαρτημάτων του, και στο web κομμάτι, το οποίο περιλαμβάνει στον server, την δημιουργία κατάλληλου site και την επικοινωνία με το Facebook.



Σχήμα 1 Διαδρομή της πληροφορίας

Η ροή της πληροφορίας φαίνεται στο παραπάνω σχήμα ενώ η γενική λειτουργία της εφαρμογής φαίνεται στο παρακάτω σχήμα και θα εξηγηθεί εκτενέστερα στα επόμενα κεφάλαια.



Σχήμα 2 Γενική Λειτουργία της εφαρμογής SMStoAuto

Κεφάλαιο 2

Υλικά, Κατασκευή και Προγραμματισμός

2.1 Καταγραφή χρησιμοποιημένων συσκευών

Σε αυτό το σημείο θα ξεκινήσουμε μία καταγραφή και ανάλυση των συσκευών και εξαρτημάτων που θα χρησιμοποιήσουμε. Στις παρακάτω υποενότητες δηλαδή, θα δούμε το hardware (υλικό) της εφαρμογής μας καθώς και τις ιδιότητες του κάθε κομματιού αυτού. Συγκεκριμένα, ακολουθούν τα GSM Module , Arduino MEGA2560 , Ethernet Module και LCD Module.

2.1.1 GSM Module

Το GSM Module που αγοράστηκε για την κατασκευή είναι ένα ολοκληρωμένο board που βασίζεται πάνω στο GSM MODEM TC35 της εταιρείας SIEMENS.



Εικόνα 1 GSM Module TC35

Όπως μπορούμε να δούμε και από την φωτογραφία του, παρέχει εξωτερική κεραία για την λήψη του σήματος ενώ για να λειτουργήσει χρειάζεται ένα τροφοδοτικό που να μπορεί να αποδώσει τουλάχιστον 9Volt και 1 Ampere. Διαθέτει θύρα RS232 για συριακή επικοινωνία χρησιμοποιώντας είτε κλασικό 9pin καλώδιο μέσω της μπροστινής θύρας είτε με τους ακροδέκτες RX και TX οι οποίοι βρίσκονται πίσω από την πραγματική θύρα και βολεύουν περισσότερο στην σύνδεση με μικροελεγκτές. Το board επίσης έχει ειδικά jumpers τα οποία πρέπει να ρυθμιστούν ανάλογα την σύνδεση που θέλουμε

να επιτύχουμε, δηλαδή με υπολογιστή ή με μικροελεγκτή, πάντα σύμφωνα με το datasheet του.

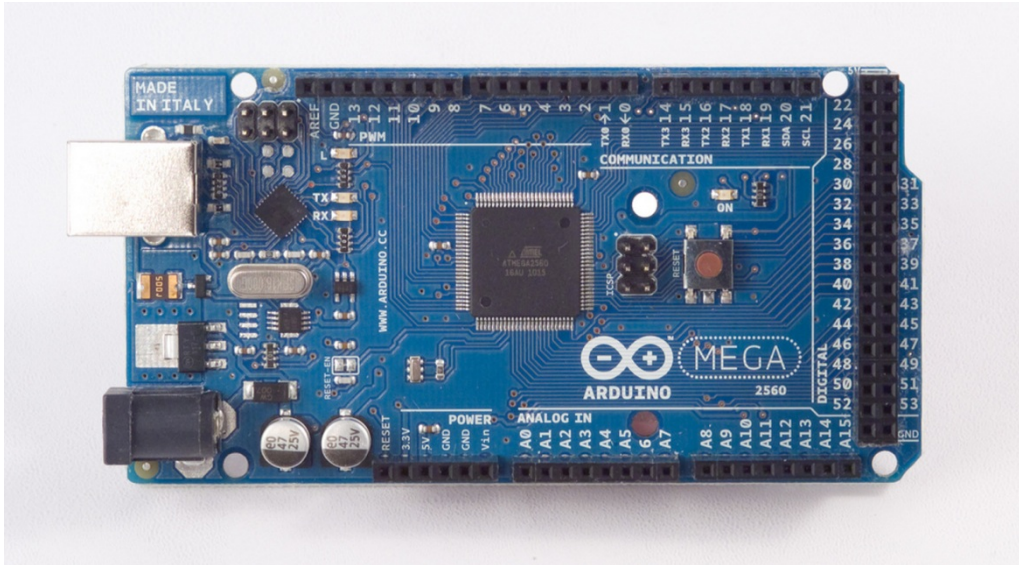
Τέλος, το board παρέχει και ένα κουμπί reset για να μπορούμε χειροκίνητα να το επανεκκινήσουμε και μερικές φωτεινές ενδείξεις που μας ειδοποιούν για την σωστή λειτουργία του board και την ύπαρξη εισερχόμενης κλήσης.

Αναγκαίο βήμα είναι η τοποθέτηση μίας κάρτας SIM στην ειδική υποδοχή που βρίσκεται στην κάτω πλευρά της πλακέτας. Μετά την τοποθέτησή της μπορούμε να χρησιμοποιήσουμε κανονικά το board και δίνοντάς του κατάλληλες εντολές μέσω της σειριακής να μας δώσει τις αντίστοιχες απαντήσεις. Το modem αυτό επικοινωνεί με εντολές AT.

Αυτές είναι ένα σετ εντολών οι οποίες όλες ξεκινάνε με τα γράμματα AT και μετά είτε έχουν άλλο ένα γράμμα που υποδηλώνει την ενέργεια που ζητάμε (π.χ. ATD είναι η εντολή για την κλήση ή αλλιώς Dial) είτε έχουν το σύμβολο + και μετά την εντολή που αντιστοιχεί στην ενέργεια που θέλουμε (π.χ. AT+CMGS και AT+CMGR είναι οι εντολές για την αποστολή και λήψη γραπτών μηνυμάτων ή αλλιώς Send και Receive).

2.1.2 Arduino MEGA2560

Η έκδοση του Arduino που προτιμήθηκε για την εργασία ήταν αυτή του MEGA2560, η οποία όπως προδίδει και το όνομά της βασίζεται στον μικροελεγκτή ATmega2560.



Εικόνα 2 Arduino MEGA2560

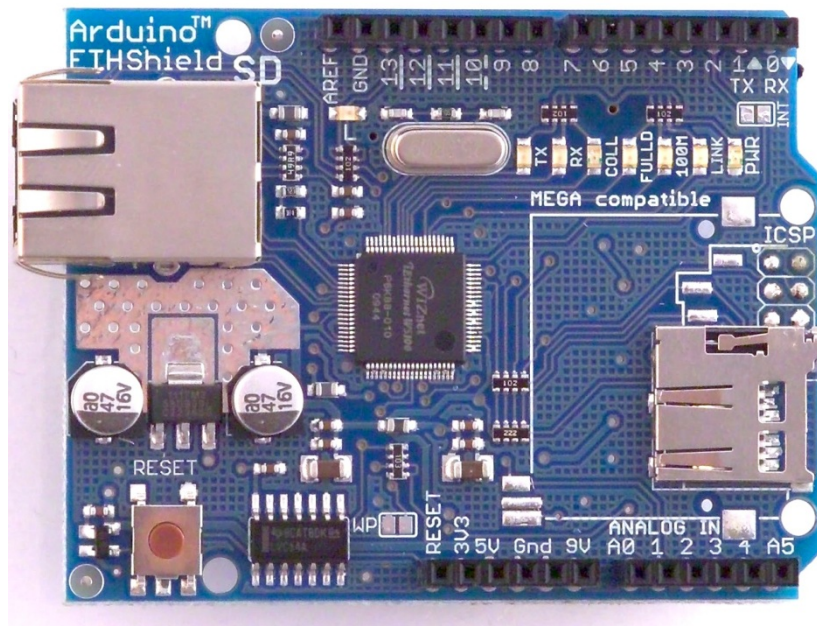
Ο Arduino μπορεί να τροφοδοτηθεί κατευθείαν από την USB ή από εξωτερικό τροφοδοτικό που να υποστηρίζει από 7 έως 12 Volt. Μέσω της θύρας USB γίνεται και ο προγραμματισμός του, χρησιμοποιώντας το προγραμματιστικό περιβάλλον του Arduino (Arduino Development Environment). Προσφέρει 54 ψηφιακές εισόδους/εξόδους, εκ των οποίων οι 15 παρέχουν έξοδο PWM (Pulse-Width Modulation), δηλαδή μπορούν να εξομοιώσουν αναλογική έξοδο, ενώ οι 8 παρέχουν σειριακή επικοινωνία, δηλαδή ανά 2 δημιουργούν ένα ζεύγος RX και TX. Πέρα βέβαια των ψηφιακών, παρέχει άλλες 16 εισόδους αναλογικού σήματος. Διαθέτει 256KB flash μνήμης στα οποία αποθηκεύεται το πρόγραμμα του χρήστη.

Ο σχεδιασμός του και, ειδικά η τοποθέτηση του ICSP header στο κατάλληλο, κεντρικό όπως φαίνεται στην εικόνα, σημείο καθιστά απόλυτα συμβατό με τα shields που έχουν κυκλοφορήσει για τις πιο διαδεδομένες εκδόσεις του Arduino, όπως το Duemilanove και το Uno.

Για να γίνει καλύτερα αντιληπτή η διαφορά της έκδοσης MEGA2560 από τις αυτές με τον μικροελεγκτή ATmega328, όπως τα πιο διαδεδομένα Arduino Uno και Duemilanove, αρκεί να αναφέρουμε πως αυτές παρέχουν μόλις 14 ψηφιακές εισόδους/εξόδους και 6 αναλογικές εισόδους ενώ διαθέτουν λιγότερο από 32KB μνήμης για το πρόγραμμα του χρήστη.

2.1.3 Ethernet Shield

Το Ethernet Shield συνδέεται πάνω στον Arduino και του προσθέτει παραπάνω λειτουργίες. Συγκεκριμένα, του δίνει την δυνατότητα διασύνδεσης με το διαδίκτυο ενώ το συγκεκριμένο μοντέλο, του προσθέτει και έναν έξτρα αποθηκευτικό χώρο μέσω της υποδοχής για κάρτα microSD.



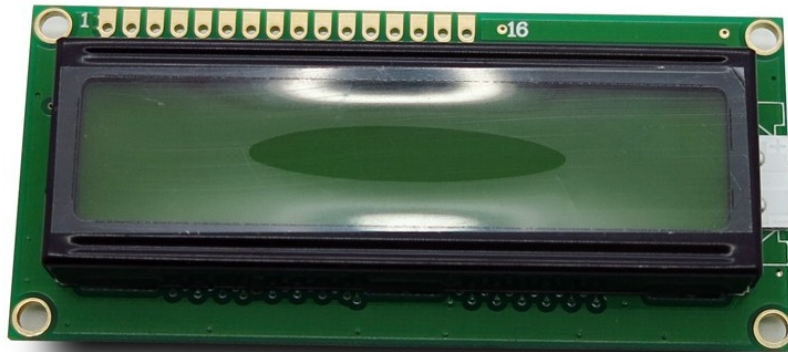
Εικόνα 3 Ethernet Shield

Το shield συνδέεται ακριβώς πάνω από τον Arduino, χωρίς να χρειαστούν πρόσθετες κολλήσεις ή άλλες παραμετροποιήσεις. Με την πρόσθεσή του δεν χάνουμε καμία από τις δυνατότητες του Arduino καθώς μέσω των pins του shield, μεταφέρεται η λειτουργικότητα των θυρών του Arduino πάνω στο shield. Η πλακέτα βασίζεται πάνω στον Ethernet ελεγκτή W5100 της εταιρείας WIZnet ο οποίος παρέχει την επικοινωνία μέσω TCP ή UDP, ενώ μπορεί να χρησιμοποιηθεί τόσο ως client όσο και ως server. Διαθέτει είσοδο για απλό RJ45 καλώδιο με το οποίο πρέπει να συνδεθεί πάνω σε κάποιον router, ενώ έχει και μια θύρα για κάρτα μνήμης microSD, στην οποία μπορούμε να αποθηκεύσουμε αρχεία με χρήσιμες πληροφορίες.

Τέλος, πάνω στο board υπάρχουν διάφορα led τα οποία μας πληροφορούν για την σωστή λειτουργία του shield, καθώς και ένα κουμπί reset το οποίο επανεκκινεί και το shield και τον Arduino.

2.1.4 LCD Screen Module

Η τεχνολογία LCD (Liquid-crystal display), που βασίζεται όπως προδίδει το όνομά της στους υγρούς κρυστάλλους, είναι ευρέως διαδεδομένη και τώρα πια έχει αντικαταστήσει πλήρως την παλαιότερη τεχνολογία καθοδικού σωλήνα.



Εικόνα 4 LCD Screen Module

Το δικό μας εξάρτημα είναι ένα module, δηλαδή μια ολοκληρωμένη πλακέτα που έχει απάνω τόσο την ίδια την οθόνη όσο και το απαραίτητο chip για την λειτουργία της. Η οθόνη μπορεί να απεικονίσει έως 16 χαρακτήρες σε κάθε μία σειρά από τις δύο που διαθέτει. Διαθέτει οπίσθιο φωτισμό για καλύτερη ανάγνωση, χρειάζεται τροφοδοσία της τάσης των 5V και έχει 16 pins τα οποία πρέπει να συνδεθούν καταλλήλως με τον μικροελεγκτή για μπορεί να εμφανίσει ότι θέλει ο χρήστης.

2.2 Συνδεσμολογία και Στάδια Κατασκευής

2.2.1 Σύνδεση του Arduino με τα υπόλοιπα εξαρτήματα

Ο Arduino επικοινωνεί με το GSM module χρησιμοποιώντας μία από τις τέσσερις σειριακές εξόδους του. Χρειάζεται να ενώσουμε δηλαδή τα pin RX και TX του GSM με αντίστοιχα RX και TX του Arduino, στην δικιά μας περίπτωση τα pin 15 και 14 αντίστοιχα. Στην σειριακή επικοινωνία κατά κανόνα η συνδεσμολογία γίνεται χιαστή, δηλαδή RX με TX, αλλά στα GSM Modules είναι αναγκαία η παράλληλη σύνδεση, δηλαδή RX με RX και TX με TX. Τέλος, είναι απαραίτητο οι δύο συσκευές να έχουν κοινή γείωση, οπότε συνδέουμε το pin Ground του GSM με το αντίστοιχο του Arduino. Φυσικά δεν ξεχνάμε να τοποθετήσουμε το jumper επιλογής που υπάρχει στο board σύμφωνα με το datasheet του στην θέση για την σύνδεση με μικροελεγκτή

Το Ethernet Shield είναι σχεδιασμένο σύμφωνα με την τελευταία του έκδοση (R3) και για αυτό δεν χρειάζεται να γίνει καμία παραμετροποίηση στην πλακέτα του καθώς είναι απολύτως συμβατό με τον Arduino Mega2560. Για αυτό το μόνο που έχουμε να κάνουμε είναι να τοποθετήσουμε το Shield πάνω στον Arduino με σχετική προσοχή ώστε να τοποθετηθεί στα σωστά pins και να συνδεθεί και ο ISP Header.

Το LCD Module θέλει αρκετές παραπάνω συνδέσεις, καθώς τουλάχιστον τα 12 από τα 16 pins που έχει πρέπει να συνδεθούν, είτε με ψηφιακές εισόδους/εξόδους του Arduino είτε με τροφοδοσία. Πιο συγκεκριμένα, θα χρειαστούμε σύνδεση για τον οπίσθιο φωτισμό (pins 15 και 16) , την

αντίθεση των χαρακτήρων (pin 3), την λογική λειτουργίας της LCD (pins 1 και 2), την μεταφορά των δεδομένων (pins 11, 12, 13 και 14) και λοιπών επιλογών (pins 4, 5 και 6).

Ένας συνοπτικός πίνακας με τις συνδέσεις που θα χρειαστούν ακολουθεί παρακάτω.

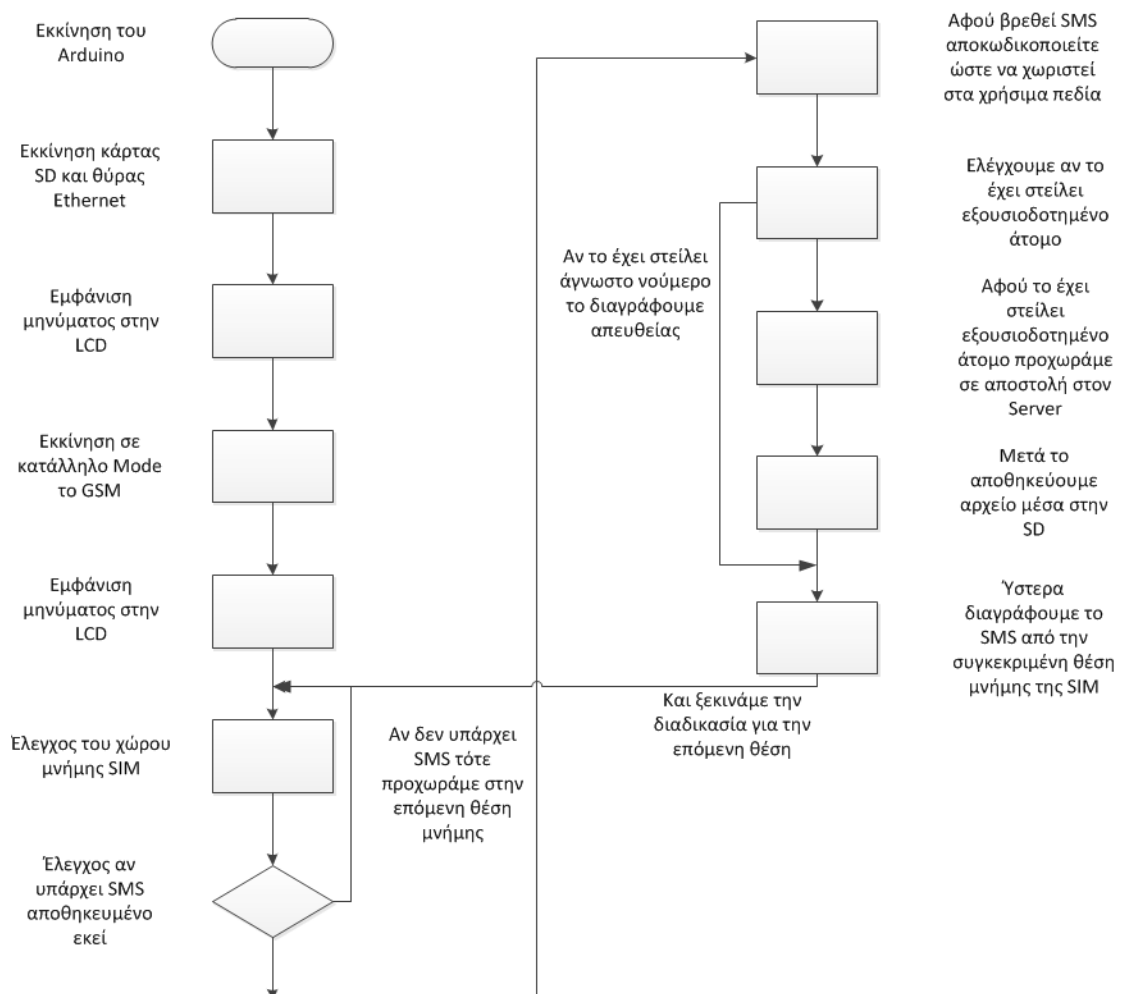
<u>Pin Arduino</u>	<u>Pin GSM</u>
Digital Pin 14	Pin TX
Digital Pin 15	Pin RX
Ground	Pin Ground
<u>Pin Arduino</u>	<u>Pin LCD</u>
Ground	Pin 1
+5V	Pin 2
+5V	Pin 3
Digital Pin 31	Pin 4
Ground	Pin 5
Digital Pin 30	Pin 6
Digital Pin 29	Pin 11
Digital Pin 28	Pin 12
Digital Pin 27	Pin 13
Digital Pin 26	Pin 14
+5V	Pin 15
Ground	Pin 16

Πίνακας 1 Διάγραμμα Συνδέσεων

2.2.2 Λειτουργία του Arduino

Αφού τελειώσαμε με την συνδεσμολογία των εξαρτημάτων ήρθε η ώρα να δούμε την λογική λειτουργίας του μηχανήματός μας. Το κομμάτι αυτό είναι το κεντρικό σημείο της εφαρμογής και είναι αυτό για το οποίο είναι υπεύθυνος ο Arduino.

Πριν ξεκινήσουμε την ανάλυση του προγράμματος, θα ήταν καλύτερο να δούμε συνοπτικά και απλοποιημένα την γενική λειτουργία του προγράμματος του Arduino. Αυτή, απεικονίζεται στο παρακάτω σχήμα.



Σχήμα 3 Λειτουργία του μικροελεγκτή Arduino.

Τώρα θα προχωρήσουμε σε πιο βαθιά ανάλυση τους προγράμματος και των βημάτων του. Για τον προγραμματισμό του Arduino χρησιμοποιήθηκε η τελευταία έκδοση του ADE(Arduino Development Environment) δηλαδή η 1.0.4. Η δομή ενός απλού προγράμματος φαίνεται στην παρακάτω εικόνα.



```
SoftwareSerialExample | Arduino 1.0.4
Αρχείο Επεξεργασία Σχέδιο Εργαλεία Βοήθεια
SoftwareSerialExample $
#include <SoftwareSerial.h>

SoftwareSerial mySerial(10, 11); // RX, TX

void setup()
{
  // Open serial communications and wait for port to open:
  Serial.begin(57600);
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Leonardo only
  }

  Serial.println("Goodnight moon!");

  // set the data rate for the SoftwareSerial port
  mySerial.begin(4800);
  mySerial.println("Hello, world?");
}

void loop() // run over and over
{
  if (mySerial.available())
    Serial.write(mySerial.read());
  if (Serial.available())
    mySerial.write(Serial.read());
}
```

1 Arduino Mega 2560 or Mega ADK on COM20

Εικόνα 5 Arduino Development Environment

Όπως μπορούμε να δούμε, η δομή του προγράμματος ουσιαστικά χωρίζεται σε τρία μέρη, στις αρχικές δηλώσεις, όπου εκεί προσθέτουμε απαραίτητες βιβλιοθήκες έτοιμου κώδικα ή αρχικοποιούμε τιμές μεταβλητών, στο κομμάτι setup, το οποίο τρέχει μόνο μία φορά μόλις τροφοδοτήσουμε τον Arduino και τέλος το κομμάτι loop, στο οποίο γράφουμε συνήθως το μεγαλύτερο κομμάτι του κώδικά μας καθώς είναι αυτό που επαναλαμβάνεται και εκτελείται συνεχώς.

Σε αυτό το σημείο είναι απαραίτητη η αναφορά στην τεράστια ευκολία που προσφέρει η δυνατότητα χρήσης έτοιμων βιβλιοθηκών, καθώς με αυτό τον τρόπο δεν χρειάζεται να ξεκινάς κάθε φορά από το πρόγραμμά σου κυριολεκτικά από την αρχή, αφού μπορείς να πάρεις κομμάτια τα οποία είναι έτοιμα και ήδη, κάτι πολύ σημαντικό, δοκιμασμένα από πολλά άτομα.

Οι βιβλιοθήκες που χρειάστηκαν για την υλοποίηση της εφαρμογής μας ακολουθούν.

- Βιβλιοθήκη SPI.h

Πρόκειται για μια πολύ σημαντική βιβλιοθήκη που επιτρέπει την επικοινωνία του Arduino με συσκευές μέσω του πρωτοκόλλου SPI (Serial Peripheral Interface) με τον Arduino ως την Master συσκευή. Είναι δηλαδή απαραίτητη βιβλιοθήκη για την λειτουργία των διάφορων Shields.

- Βιβλιοθήκη Ethernet.h

Η βιβλιοθήκη αυτή δημιουργήθηκε μαζί με το αντίστοιχο Shield και όπως προδίδει το όνομά της επιτρέπει στον Arduino να συνδεθεί με το Internet. Μπορεί να λειτουργήσει είτε ως server και να δέχεται συνδέσεις είτε ως client για να τις κάνει αυτός.

- Βιβλιοθήκη SD.h

Η βιβλιοθήκη αυτή υποστηρίζει την ανάγνωση και γραφή αρχείων από κάρτα microSD. Η κάρτα microSD πρέπει να είναι διαμορφωμένη σε FAT16 ή FAT32 σύστημα αρχείων ενώ η υποδοχή της microSD μπορεί να έρχεται από το Ethernet Shield ή ακόμα και από άλλο αυτόνομο Shield.

- Βιβλιοθήκη LiquidCrystal.h

Αυτή η βιβλιοθήκη επιτρέπει στον Arduino να χειριστεί μια οθόνη υγρών κρυστάλλων αρκεί να βασίζεται στο chip HD44780 της εταιρείας Hitachi, ή κάποιο αντίστοιχο και συμβατό, πράγμα που ισχύει για τις περισσότερες οθόνες γραμμάτων. Η βιβλιοθήκη λειτουργεί τόσο με την χρήση του 4-bit mode (δηλαδή με την σύνδεση μόνο 4 καλωδίων από το data bus της οθόνης, όπως και στην περίπτωση μας δηλαδή) όσο και με την χρήση 8-bit mode (σύνδεση δηλαδή και των 8 καλωδίων του data bus).

Πέρα από την προσθήκη των βιβλιοθηκών, σε αυτό το σημείο πρέπει να δηλώσουμε όσες μεταβλητές θέλουμε να είναι προσβάσιμες από όλο το πρόγραμμα καθώς και να αρχικοποιήσουμε τις τιμές που θέλουμε να έχουν αυτές στην αρχή του προγράμματος. Τέλος, εδώ πρέπει να προστεθούν και κάποιες γραμμές κώδικα σχετικές με τις διάφορες συνδέσεις που έχουμε κάνει, όπως πχ την MAC του Ethernet ή τα Pin σύνδεσης της LCD. Σημαντική επιλογή σε αυτό το σημείο είναι αν θα αφήσουμε τον DHCP (Dynamic Host Configuration Protocol) του δικτύου μας να παραχωρήσει αυτόματα μια IP διεύθυνση στο Ethernet Module, ή αν θα επιλέξουμε να ρυθμίσουμε μόνοι μας την διεύθυνση IP του όπως και την προεπιλεγμένη πύλη και μάσκα υποδικτύου, σύμφωνα πάντα με τις ρυθμίσεις που έχουμε στο δίκτυό μας.

Μετά προχωράμε στο σημαντικό κομμάτι του setup. Σε αυτό το κομμάτι πρέπει να βάλουμε τον κώδικα που θα τρέξει μία φορά την στιγμή που τροφοδοτηθεί το Arduino. Τα βήματα εδώ είναι σημαντικά καθώς καθορίζουν την σωστή μετέπειτα λειτουργία του Arduino και μπορούν να χωριστούν σε τρία επιμέρους κομμάτια.

- Σωστή εκκίνηση του Ethernet Shield και της κάρτας microSD

Η επικοινωνία με το Ethernet και την κάρτα microSD γίνεται μέσω του ίδιου πρωτοκόλλου SPI και για αυτό μπορούν να δημιουργηθούν προβλήματα ταυτόχρονης μετάδοσης, γιατί προφανώς μόνο ένα από τα δύο μπορεί να το χρησιμοποιεί την ίδια στιγμή. Για αυτό μέσω των διαφορετικών SS pins (Slave Select), τα οποία είναι υπεύθυνα για την εκκίνηση και διακοπή λειτουργίας της αντίστοιχης συσκευής, εκκινούμε τις δύο συσκευές μας.

- Ανάγνωση αρχείου καταλόγου από την κάρτα microSD

Αφού σιγουρευτούμε πως το Ethernet Shield μας λειτουργεί κανονικά, ανοίγουμε το κατάλληλο αρχείο μέσα από την microSD που περιέχει τα νούμερα και τα ονόματα των κατόχων τους που θέλουμε να επιτρέπουμε την χρήση της εφαρμογής. Μετά το σωστό άνοιγμα του αρχείου τραβάμε τα στοιχεία που θέλουμε και τα τοποθετούμε σε κατάλληλο πίνακα του προγράμματος πριν κλείσουμε τελικά το αρχείο.

- Εμφάνιση κατάλληλων μηνυμάτων στην LCD

Το τελευταίο βήμα της setup είναι η εμφάνιση των κατάλληλων μηνυμάτων στην LCD για την ειδοποίηση του χειριστή. Αυτό γίνεται καθ' όλη την διάρκεια της επεξεργασίας του κώδικα της setup, παρέχοντας πληροφορία για την σωστή ή όχι εκκίνηση της εφαρμογής και τελικά την αρχή λειτουργίας της.

Τελευταίο και πιο σημαντικό είναι το κομμάτι loop, το κομμάτι δηλαδή που θα επαναλαμβάνεται συνεχώς και θα εκτελεί το κυρίως μέρος της διαδικασίας. Για να γίνει πιο εύκολος ο προγραμματισμός αυτού του κομματιού, δημιουργήθηκαν κάποιες μέθοδοι (τις οποίες τις προσθέτουμε στο τέλος του προγράμματος, μετά το τέλος της loop) οι οποίες μετά καλούνται μέσα από την loop.

- Μέθοδος λήψης SMS

Με αυτή την μέθοδο στέλνουμε μέσω της κατάλληλης σειριακής θύρας την αντίστοιχη εντολή στο GSM Module για την ανάγνωση συγκεκριμένης θέσης μνήμης.

- Μέθοδος διαγραφής SMS

Με αυτή την μέθοδο στέλνουμε στο GSM Module την κατάλληλη εντολή για την διαγραφή της συγκεκριμένης θέσης μνήμης.

- Μέθοδος λήψης απάντησης

Με αυτή την μέθοδο διαβάζουμε την απάντηση του GSM Module μετά από κάθε δικιά μας εντολή. Η απάντηση του GSM είναι ένα πολύ μεγάλο string το οποίο μέσα έχει χρήσιμη αλλά και άχρηστη πληροφορία.

- Μέθοδος επεξεργασίας απάντησης

Με την μέθοδο αυτή ο Arduino αποκρυπτογραφεί την απάντηση που έλαβε από το GSM Module και διαχωρίζει την χρήσιμη από την άχρηστη πληροφορία. Συγκεκριμένα, από το string εξάγει το νούμερο, την ώρα παραλαβής και το κείμενο του SMS. Μετά ελέγχει αν το νούμερο που έστειλε το SMS υπάρχει μέσα στον πίνακα με τα κατάλληλα νούμερα και, αν υπάρχει το αντικαθιστά με το αντίστοιχο μήνυμα και προχωράει σε αποστολή στον server και στην κάρτα microSD μέσω άλλων μεθόδων και τέλος, το διαγράφει, ενώ αν δεν υπάρχει το διαγράφει κατευθείαν.

- Μέθοδος εγγραφής στην κάρτα microSD

Με την μέθοδο αυτή ο Arduino ανοίγει ή δημιουργεί κατάλληλο αρχείο txt μέσα στην κάρτα microSD, σύμφωνα με την ημερομηνία που στάλθηκε το προς αποθήκευση SMS και, γράφει σε αυτό τον αποστολέα, την ημερομηνία και το κείμενο του μηνύματος.

- Μέθοδος αποστολής στον server

Με την μέθοδο αυτή ο Arduino χρησιμοποιεί το Ethernet Shield για να λειτουργήσει ως client και να στείλει το κατάλληλο GET Request στον server μας ώστε να επικοινωνήσει με κατάλληλο αρχείο php που βρίσκεται αποθηκευμένο εκεί και να του περάσει τα στοιχεία του SMS. Εδώ σημαντικό είναι να προσέξουμε την αποφυγή των επικίνδυνων χαρακτήρων από το κείμενο του SMS, μετατρέποντάς τους στους αντίστοιχους ασφαλείς για URL χαρακτήρες.

- Μέθοδος για επανεκκίνηση του Arduino

Με την μέθοδο αυτή ο Arduino προκαλεί την επανεκκίνηση του εαυτού του και μαζί όλης της εφαρμογής. Αυτό είναι πολύ σημαντικό καθώς ο Arduino έχει μια μεταβλητή millis, η οποία μετράει τον χρόνο που έχει περάσει από την τελευταία ενεργοποίηση του board. Αυτή η μεταβλητή αναγκαστικά σε κάποια στιγμή υπερχειλίζει και επιστρέφει στο 0, πράγμα που εμείς δεν το θέλουμε καθώς την χρησιμοποιούμε μέσα στο πρόγραμμά μας. Επίσης, έχει παρατηρηθεί πως το Ethernet Shield αρχίζει να συμπεριφέρεται περίεργα μετά από μερικές μέρες λειτουργίας. Στην περίπτωση μας έχουμε θέσει την αυτόματη επανεκκίνηση μετά το πέρας 7 ημερών.

Έχοντας δημιουργήσει τις απαραίτητες μεθόδους, το κομμάτι loop του προγράμματος γίνεται μια απλή ρουτίνα διαδοχικής κλήσης της κατάλληλης μεθόδου. Πιο συγκεκριμένα, η loop μπορεί να χωριστεί σε τρία μέρη.

- Έλεγχος τελευταίας εκκίνησης

Εδώ ο Arduino ελέγχει αν η μεταβλητή millis έχει ξεπεράσει την τιμή 604800000, το οποίο αντιστοιχεί σε 7 ημέρες αφού η millis μετράει σε χιλιοστά του δευτερολέπτου. Στην περίπτωση που το νούμερο είναι μεγαλύτερο, καλείται η μέθοδος επανεκκίνησης.

- Σετάρισμα του GSM Module σε κατάλληλο Mode

Εδώ ο Arduino στέλνει την κατάλληλη εντολή στο GSM Module για να το θέσει στην λειτουργία SMS text. Αυτό το βήμα είναι σημαντικό καθώς τα GSM Modules κανονικά δουλεύουν σε SMS PDU mode, στο οποίο είναι πολύ πιο δύσκολη η αποκρυπτογράφηση του εισερχόμενου SMS.

- Έλεγχος, λήψη αποστολή και διαγραφή

Εδώ ο Arduino καλεί τις κατάλληλες μεθόδους για την τελική εκτέλεση της εφαρμογής. Συγκεκριμένα, ξεκινώντας από την πρώτη θέση μνήμης της κάρτας SIM, καλεί τις απαραίτητες μεθόδους για έλεγχο, λήψη, αποστολή, εγγραφή και διαγραφή του μηνύματος και

μετά προχωράει στην επόμενη θέση μνήμης. Προφανώς δεν υπάρχει λόγος να ελεγχθεί ολόκληρη η μνήμη της SIM, οπότε μόλις τελειώσει την διαδικασία και για την 10 θέση μνήμης, επιστρέφει πάλι στην πρώτη.

2.3 Δημιουργία web server και εγκατάσταση απαραίτητων προγραμμάτων

2.3.1 Παρουσίαση του Cloud Computing και της υπηρεσίας ~Okeanos

Το web κομμάτι της εφαρμογής μας θα πρέπει να φιλοξενηθεί σε έναν server στο Internet. Για αυτό τον σκοπό επιλέξαμε την λύση του ~Okeanos, της καινούργιας IaaS (Infrastructure as a Service) υπηρεσίας κατασκευασμένης από την ΕΔΕΤ (Εθνικό Δίκτυο Έρευνας και Τεχνολογίας).

Πριν την ανάλυση της διαδικασίας της εγκατάστασης, θα παρουσιάσουμε με λίγα λόγια την υπηρεσία ~Okeanos και γενικότερα το cloud computing όπως και τα μεγάλα εφόδια που δίνει στους χρήστες της.

Με τον όρο Cloud Computing αναφερόμαστε στην χρήση υπολογιστικών πόρων (υλικό και λογισμικό) που βρίσκονται σε απομακρυσμένη περιοχή και είναι προσβάσιμοι μέσω κάποιου δικτύου, κυρίως του Internet. Σύμφωνα με το Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας της Αμερικής , αυτό το μοντέλο λειτουργίας για να λειτουργήσει σωστά πρέπει να διαθέτει τα παρακάτω χαρακτηριστικά.

- Άμεση εξυπηρέτηση καταπαίτηση (On-demand self-service)
- Ευρεία πρόσβαση δικτύου (Broad network access)
- Συγκέντρωση και ανακατανομή πόρων (Resource pooling)
- Γρήγορη προσαρμοστικότητα (Rapid elasticity)
- Μετρήσιμη υπηρεσία (Measured service)

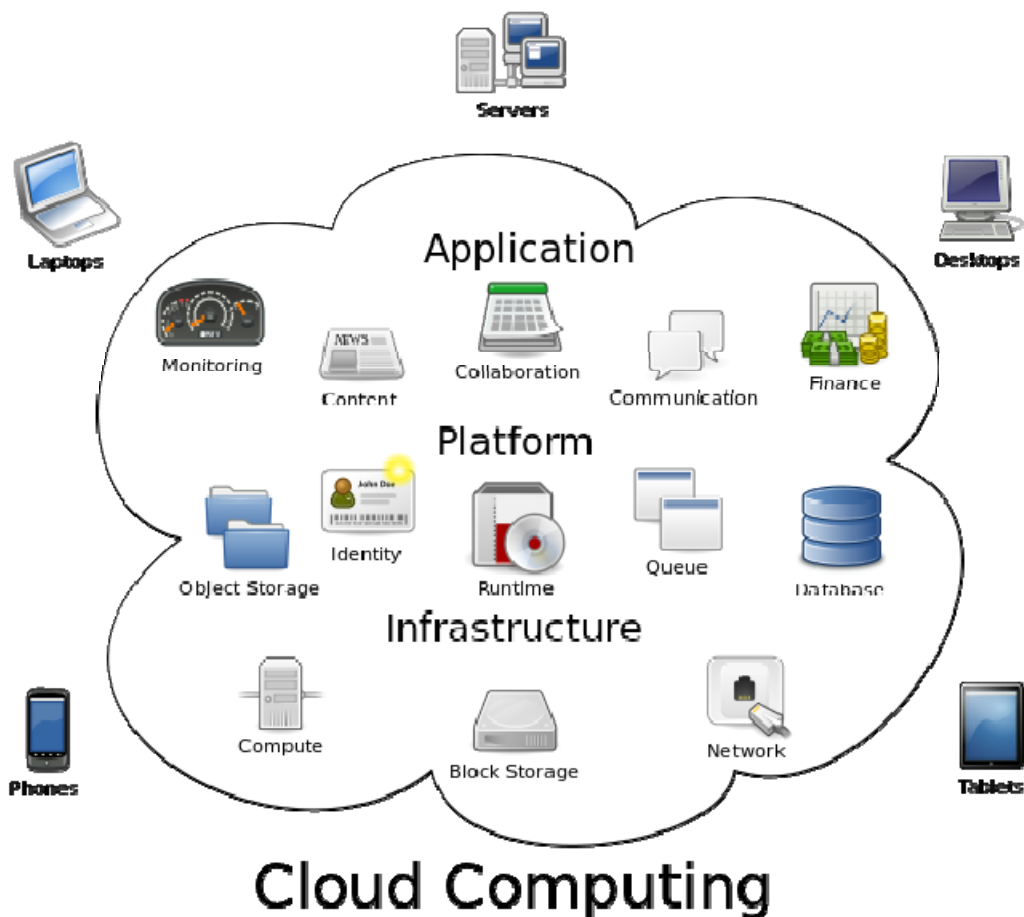
Σύμφωνα με το μοντέλο της υπηρεσίας που παρέχει το Cloud Computing χωρίζεται σε άλλες τρεις κατηγορίες.

- Λογισμικό ως Υπηρεσία (Software as a Service)
- Πλατφόρμα ως Υπηρεσία (Platform as a Service)
- Υποδομή ως Υπηρεσία (Infrastructure as a Service)

Τέλος, υπάρχει και ο διαχωρισμός σύμφωνα με το μοντέλο ανάπτυξης, το οποίο χωρίζεται στις παρακάτω κατηγορίες.

- Ιδιωτικό Cloud (Private Cloud)
- Δημόσιο Cloud (Public Cloud)
- Κοινό Cloud (Common Cloud)
- Υβριδικό Cloud (Hybrid Cloud)

Ο ορισμός βέβαια του Cloud Computing δεν είναι καθολικός, κυρίως λόγω της ταχείας ανάπτυξης του αλλά και του τρόπου με τον οποίο επιλέγουν να τον χρησιμοποιούν οι εταιρείες που τον διαθέτουν. Συνεχώς προστίθενται νέα μοντέλα τα οποία περιγράφουν μία νέα ή πιο συγκεκριμένη μορφή Cloud Computing, όπως τα DaaS (Δεδομένα ως Υπηρεσία - Data as a Service) και NaaS (Δίκτυο ως Υπηρεσία – Network as a Service).



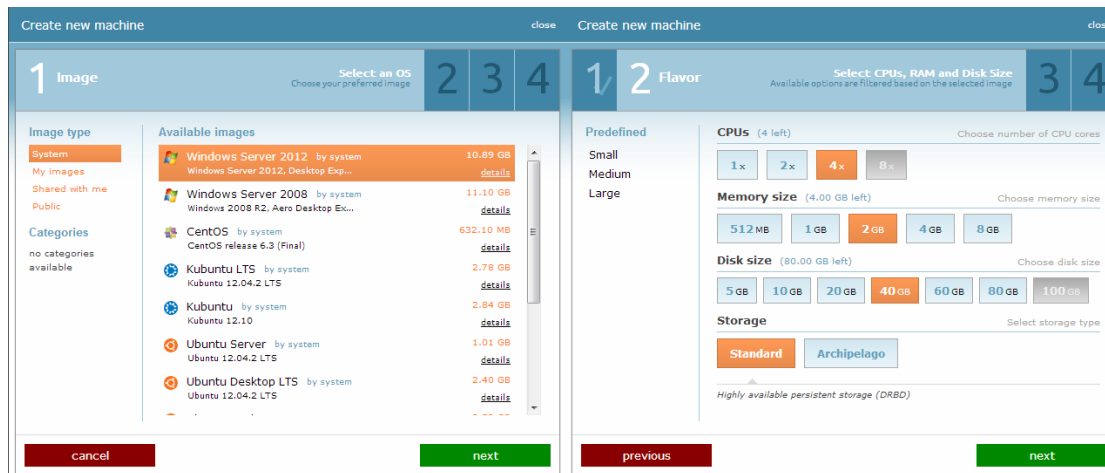
Εικόνα 6 Γραφική απεικόνιση του Cloud Computing

Σε αυτή την εργασία θα ασχοληθούμε με το μοντέλο IaaS, στο οποίο ανήκει και ο ~Οκεανος. Αυτό το μοντέλο παρέχει στον χρήστη του τις κατάλληλες

υπολογιστικές και δικτυακές υποδομές , όπως αποθηκευτικό χώρο και υπολογιστική δύναμη, και του επιτρέπει την εγκατάσταση και χρήση ολοκληρωμένων λειτουργικών συστημάτων. Δηλαδή, ο χρήστης ενός μοντέλου IaaS μπορεί να εγκαταστήσει στον εικονικό του υπολογιστή όποιο λειτουργικό σύστημα θέλει και να το διαχειριστεί με τον τρόπο που επιθυμεί ο ίδιος, για να καλύψει τις ανάγκες του.

Το ~Okeanos είναι μια υπηρεσία τύπου IaaS που έφτιαξε η ΕΔΕΤ και την προσφέρει στην Ελληνική ακαδημαϊκή κοινότητα. Λειτουργεί πάνω στους δικούς της Servers και μπορεί ο κάθε φοιτητής να δημιουργήσει άμεσα έναν λογαριασμό απλά και μόνο με το email που του έχει δώσει η σχολή του. Μέσα από τον ~Okeanos ο χρήστης έχει πρόσβαση σε δύο υπηρεσίες, την Pithos+ και την Cyclades.

Η πρώτη είναι μία cloud hosting υπηρεσία, αντίστοιχη των Dropbox και Google Drive, στην οποία ο χρήστης μπορεί να αποθηκεύσει αρχεία στα οποία θα μπορεί να έχει πρόσβαση από οποιονδήποτε υπολογιστή. Εμείς θα ασχοληθούμε με την δεύτερη υπηρεσία, Cyclades, στην οποία ο χρήστης αποκτά πρόσβαση σε εικονικό δίσκο με χώρο 100 Gigabytes, σε 8 πυρήνες επεξεργαστικής ισχύς και σε μνήμη RAM χωρητικότητας 8 Gigabytes. Αυτά τα στοιχεία ο χρήστης μπορεί να τα διανέμει όπως αυτός θέλει σε μέχρι δύο αυτόνομα υπολογιστικά συστήματα.



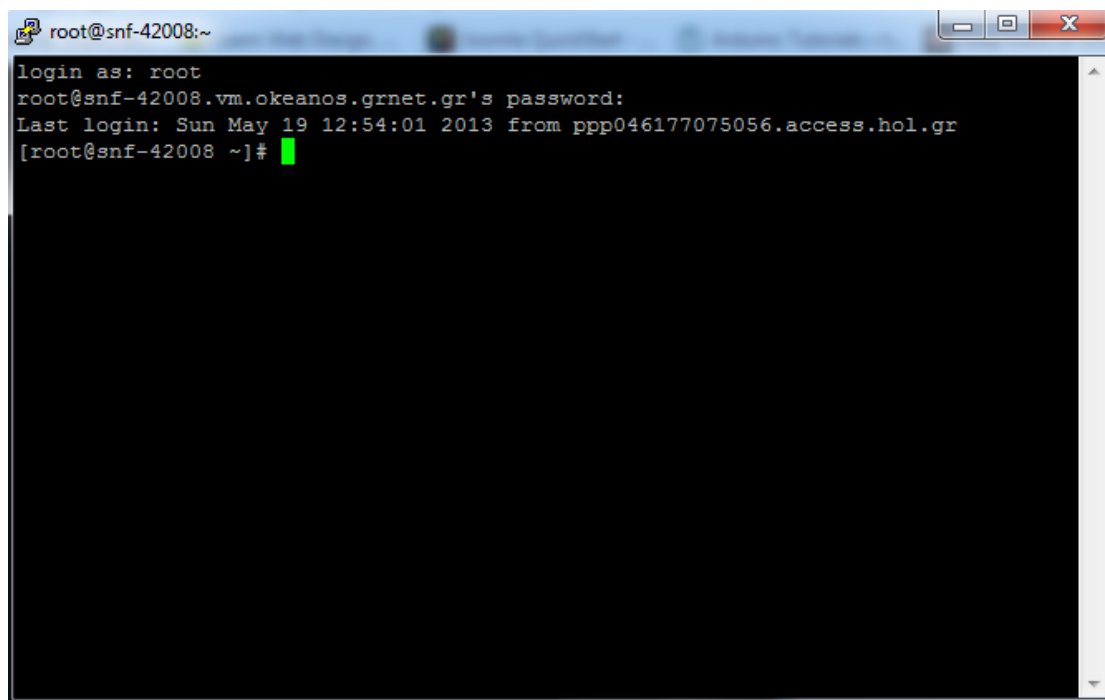
Εικόνα 7 Βασικές επιλογές δημιουργίας υπολογιστικού συστήματος

Στο κάθε υπολογιστικό σύστημα που δημιουργούμε, έχουμε την επιλογή να εγκαταστήσουμε ένα λειτουργικό σύστημα είτε από μία παρεχόμενη λίστα είτε μέσω αρχείου εικονικού δίσκου που θα ανεβάσουμε εμείς.

Μετά ακολουθούν κάποιες επιλογές σχετικά με την ασφάλεια τις σύνδεσης αλλά, είναι ξεκάθαρο πως η δημιουργία ενός τέτοιου μηχανήματος είναι μία πολύ απλή διαδικασία μερικών κλικ. Αφού περιμένουμε μερικά λεπτά για την τελική δημιουργία του μηχανήματος, μας δίνεται η μοναδική IP διεύθυνση του συστήματος και μπορούμε πια να συνδεθούμε απομακρυσμένα σε αυτό και να το διαχειριστούμε. Στην περίπτωση της εφαρμογής μας διαλέξαμε να εγκαταστήσουμε λογισμικό Linux και συγκεκριμένα την διανομή CentOS.

2.3.2 Διαχείριση υπολογιστικού συστήματος και εγκατάσταση εφαρμογών

Για την σύνδεση στον απομακρυσμένο υπολογιστή θα χρησιμοποιήσουμε το PuTTY, ένα πολύ γνωστό και φυσικά open source λογισμικό που μετατρέπει τον τοπικό υπολογιστή σε τερματική συσκευή του απομακρυσμένου υπολογιστή. Το περιβάλλον του PuTTY είναι ένα παράθυρο γραμμής εντολών από το οποίο εμείς μπορούμε να δώσουμε τις εντολές που θέλουμε να τρέξουν στον απομακρυσμένο υπολογιστή.



Εικόνα 8 Περιβάλλον PuTTY

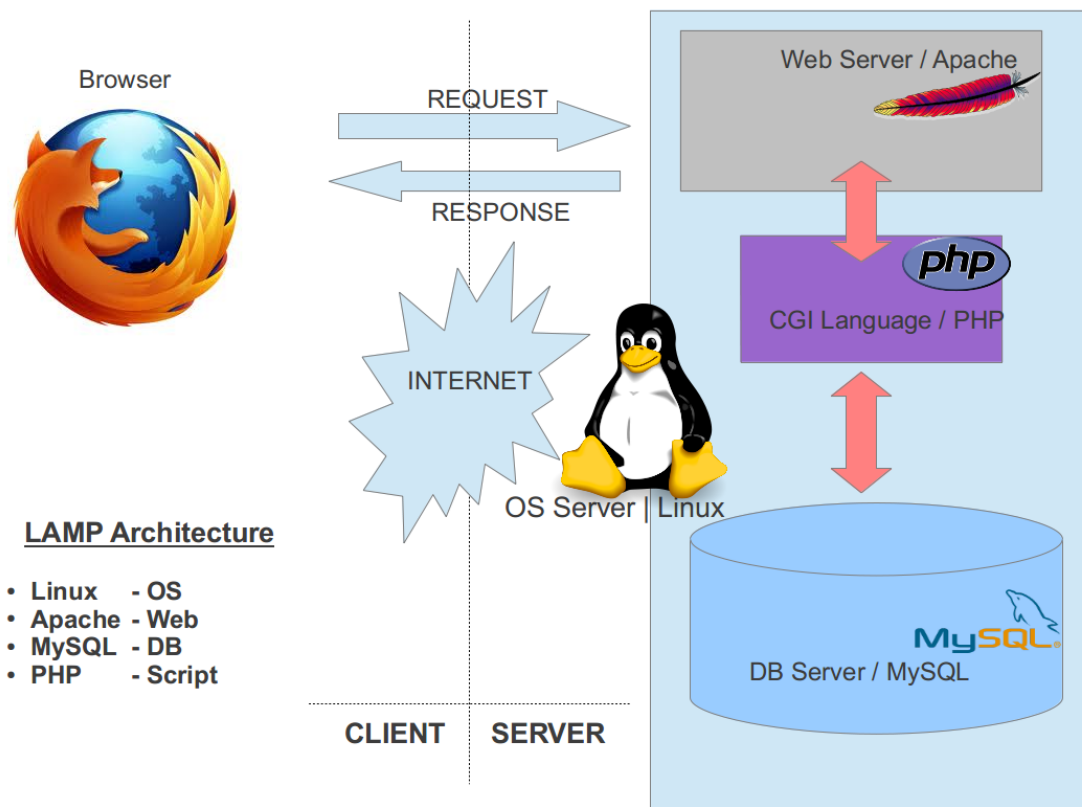
Η διανομή Linux που επιλέξαμε, CentOS, χρησιμοποιεί τον διαχειριστή πακέτων λογισμικού YUM (Yellowdog Updater, Modified). Έτσι, για να εγκαταστήσουμε κάθε πρόγραμμα που χρειαζόμαστε για την εφαρμογή μας

στον απομακρυσμένο υπολογιστή, πρέπει να δώσουμε στο PuTTY κατάλληλη εντολή του τύπου :

```
# yum install mysql-server mysql
```

Η παραπάνω εντολή θα εγκαταστήσει τον MySQL server και τα αντίστοιχα client προγράμματα πάντα με την χρήση του yum.

Απαραίτητα προγράμματα για την εφαρμογή μας, πέρα από την MySQL, είναι ο Apache HTTP Server και φυσικά η PHP, η εγκατάσταση των οποίων γίνεται με αντίστοιχες εντολές πάντα μέσω του YUM. Αφού εγκαταστήσουμε και τις τρεις εφαρμογές, θα έχουμε συμπληρώσει το γνωστό περιβάλλον για server LAMP (Linux , Apache , MySQL , PHP).



Εικόνα 9 Αρχιτεκτονική LAMP

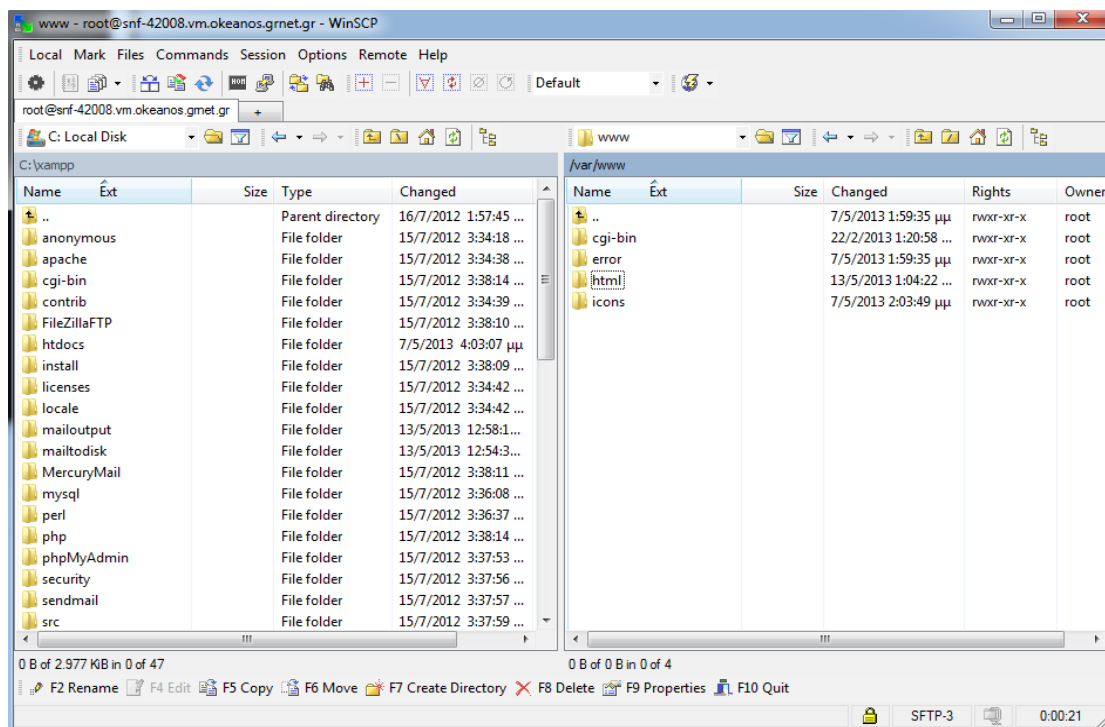
Αυτά είναι τα πλήρως απαραίτητα εργαλεία για την εφαρμογή μας. Φυσικά, στο περιβάλλον του Linux μπορούμε να κάνουμε ότι παραπάνω θέλουμε και να εγκαταστήσουμε ότι μας ενδιαφέρει, με αντίστοιχες εντολές μέσα από το PuTTY.

Τώρα ο απομακρυσμένος υπολογιστής είναι έτοιμος να δεχθεί τα δικά μας αρχεία μας τα οποία θα εμφανίζει, μέσω του Apache, στους χρήστες του διαδικτύου που προσπαθούν να προσπελάσουν την δικτυακή διεύθυνση του υπολογιστή. Τα αρχεία αυτά σύμφωνα με τις προεπιλεγμένες ρυθμίσεις του Apache Server πρέπει να μπουν στον υποκατάλογο του δίσκου **/var/www/html** .

Για την μεταφορά αρχείων από και προς τον απομακρυσμένο υπολογιστή, πρέπει να εγκαταστήσουμε στον τοπικό υπολογιστή ένα πρόγραμμα διαχείρισης και μεταφοράς αρχείων. Η επιλογή μας είναι το open source WinSCP, που υποστηρίζει τόσο το πρωτόκολλο FTP (File Transfer Protocol) όσο και το πρωτόκολλο SFTP (Secure File Transfer Protocol).

Αυτά είναι τα πλήρως απαραίτητα εργαλεία για την εφαρμογή μας. Φυσικά, στο περιβάλλον του Linux μπορούμε να κάνουμε ότι παραπάνω θέλουμε και να εγκαταστήσουμε ότι μας ενδιαφέρει, με αντίστοιχες εντολές μέσα από το PuTTY.

Η σύνδεση στον απομακρυσμένο υπολογιστή μέσω WinSCP γίνεται με παρόμοιο τρόπο με το PuTTY, μόνο που εδώ χρησιμοποιείται το πρωτόκολλο SFTP, ενώ το περιβάλλον εργασίας είναι γραφικό και αρκετά πιο φιλικό. Συγκεκριμένα, ο χρήστης μπορεί να προσπελάσει όλους τους φακέλους του απομακρυσμένου μηχανήματος και να μεταφέρει αρχεία σε αυτούς από τους δικούς του φακέλους.



Εικόνα 10 Περιβάλλον WinSCP

Μετά την εγκατάσταση και του WinSCP έχουμε τελειώσει με την εγκατάσταση του απομακρυσμένου υπολογιστή ο οποίος είναι πια έτοιμος για την χρήση που προορίζεται. Σειρά έχει η δημιουργία στον τοπικό υπολογιστή της κατάλληλης ιστοσελίδας και τέλος η μεταφορά της στον απομακρυσμένο υπολογιστή ώστε να μπορεί να την δει ο καθένας μέσω του διαδικτύου.

2.3.3 Δημιουργία της ιστοσελίδας της εφαρμογής

Σε αυτό το σημείο θα δούμε τον τρόπο και τα μέσα που χρησιμοποιήθηκαν για την δημιουργία της ιστοσελίδας της εφαρμογής μας. Απαραίτητο πρώτο βήμα είναι η εγκατάσταση ενός κατάλληλου, αντίστοιχου με αυτό που βάλαμε στον απομακρυσμένο υπολογιστή μας, περιβάλλον που να μας παρέχει τα εφόδια για να ξεκινήσουμε την δημιουργία της σελίδας. Στον τοπικό υπολογιστή χρησιμοποιούμε λογισμικό Windows, οπότε η αρχιτεκτονική LAMP που δημιουργήσαμε παραπάνω θα πρέπει να παραμετροποιηθεί ελάχιστα.

Για την διευκόλυνσή μας, επιλέχθηκε η εγκατάσταση της έτοιμης πλατφόρμας XAMPP, η οποία εγκαθιστά αυτόματα τον Apache HTTP Server, την MySQL, την PHP και την Perl. Εναλλακτική πλατφόρμα με αντίστοιχες δυνατότητες είναι το WAMPSERVER, ενώ προφανώς υπάρχει πάντα η επιλογή της ανεξάρτητης εγκατάστασης κάθε προγράμματος ξεχωριστά. Μετά την εγκατάσταση και την ενεργοποίηση των επιμέρους στοιχείων του XAMPP, είμαστε έτοιμοι να ξεκινήσουμε τον προγραμματισμό της ιστοσελίδας μας ενώ, με την τοποθέτηση των αρχείων στον κατάλληλο φάκελο, μπορούμε να βλέπουμε απευθείας το αποτέλεσμα της δουλειάς μας μέσω ενός browser.

Για την καλύτερη ανάλυση της διαδικασίας της δημιουργίας της web εφαρμογής μας, πρέπει να την χωρίσουμε στα διακριτά της μέρη τα οποία θα αναλυθούν ξεχωριστά. Αυτά είναι τα παρακάτω.

- Σχεδιαστικό κομμάτι
- Προγραμματιστικό – Διαδραστικό κομμάτι
- Σχεδιασμός βάσης δεδομένων
- Δημιουργία Facebook Application και λειτουργία της

Ξεκινώντας από το σχεδιαστικό κομμάτι λοιπόν, θα πρέπει να παρουσιάσουμε τα προτερήματα της χρήσης ενός έτοιμου Front-End Framework. Το Front-End Development, δηλαδή ο προγραμματισμός που φέρνει το τελικό οπτικό αποτέλεσμα στον χρήστη, παραδοσιακά περιελάμβανε χρήση γλωσσών όπως HTML, CSS και JavaScript. Με τον καιρό όμως παρουσιάστηκαν προβλήματα όπως η συμβατότητα μεταξύ των διαφόρων browser αλλά και φυσικά η μεγάλη γκάμα πιθανών συσκευών, καθώς πια μια ιστοσελίδα πρέπει να σχεδιαστεί με γνώμονα όχι μόνο τον ηλεκτρονικό υπολογιστή αλλά και τα κινητά όπως και τα tablets.

Τελευταία έχουν δημιουργηθεί από μεγάλες εταιρείες του χώρου τα λεγόμενα Front-End Frameworks, τα οποία παρέχουν στον προγραμματιστή ένα σχεδόν έτοιμο, προσχεδιασμένο και αρκετά εύκολα παραμετροποιήσιμο περιβάλλον με έτοιμους κανόνες CSS που ξεπερνάνε τα παραπάνω προβλήματα. Ουσιαστικά, ο προγραμματιστής κατεβάζει τα αρχεία του Framework από το site της εταιρείας, τα περνάει στο δικό του πρόγραμμα και με απλά βήματα έχει έτοιμο ένα πολύ όμορφο αποτέλεσμα που για να

επιτευχθεί διαφορετικά θα ήθελε πολύ χρόνο και η ποιότητα του αποτελέσματος θα ήταν αμφίβολη



Εικόνα 11 Twitter Bootstrap

Τέτοια Frameworks έχουν δημιουργηθεί πολλά, το καθένα με κάποια ξεχωριστά προτερήματα και μειονεκτήματα. Χαρακτηριστικά παραδείγματα είναι το Foundation που έχει δημιουργηθεί από την εταιρεία ZURB και το Kube που έχει δημιουργηθεί από την εταιρεία Impevani. Για την δικιά μας εφαρμογή, χρησιμοποιήθηκε το framework Bootstrap, το οποίο είναι φτιαγμένο από τους προγραμματιστές που φτιάξανε το Twitter. Το Bootstrap παρέχει όλα τα HTML, CSS και JavaScript στοιχεία και περιέχει ένα δωδεκάστηλο grid system που διευκολύνει πολύ στην ανάπτυξη των ιστοσελίδων μας και την παρουσίαση του περιεχομένου σε όλα τα μεγέθη.

Το τελευταίο συγκεκριμένα είναι πολύ σημαντικό, ειδικά στην εφαρμογή μας, καθώς η αμεσότητα της ενημέρωσης σχεδόν επιβάλλει την χρήση κινητού τηλεφώνου αφού προφανώς δεν έχουμε συνέχεια πρόσβαση σε ηλεκτρονικό υπολογιστή.

Μέσα από την ιστοσελίδα του Bootstrap μπορούμε είτε να κατεβάσουμε απευθείας τον κώδικα του και να τον ενσωματώσουμε στην ιστοσελίδα μας είτε, μέσα από ένα ειδικό μενού, να παραμετροποιήσουμε μια πλειάδα διαθέσιμων επιλογών και έτσι να κατεβάσουμε την δική μας έκδοση Bootstrap. Στην εφαρμογή μας ακολουθήθηκε το μοντέλο του Fluid Layout από τα παραδείγματα της σελίδας στο οποίο έγιναν αλλαγές με το προγραμματιστικό κομμάτι στο οποίο θα αναλύσουμε αμέσως μετά.

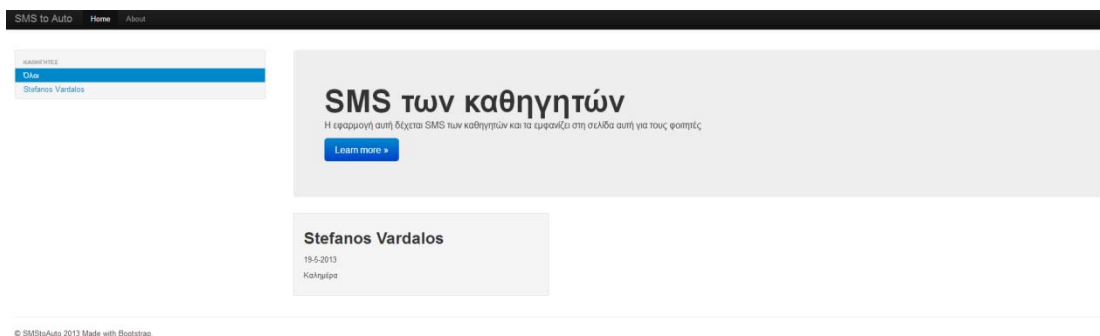
Το προγραμματιστικό κομμάτι της ιστοσελίδας έγινε με την χρήση της γλώσσας PHP. Για τον προγραμματισμό με PHP το μόνο που χρειάζεται είναι ένας text editor, αρκεί το τελικό αρχείο να έχει κατάληξη .php και όχι .txt. Φυσικά, είναι προτιμότερο να χρησιμοποιηθεί κάποιο άλλο πρόγραμμα, πιο ειδικό για προγραμματισμό, το οποίο μπορεί να παρέχει έξτρα λειτουργίες που προφανώς ο απλός text editor των Windows δεν έχει. Στην συγκεκριμένη περίπτωση χρησιμοποιήσαμε το Notepad++, ένα δωρεάν πρόγραμμα επεξεργασίας κειμένου που παρέχει πολλές ευκολίες για προγραμματισμό όχι μόνο σε PHP αλλά και σε άλλες γλώσσες.

Ο σχεδιασμός της σελίδας έγινε δημιουργώντας πέντε διαφορετικά αρχεία .php, το καθένα με την δικιά του λειτουργία. Συγκεκριμένα, έχουμε τα παρακάτω αρχεία.

- Αρχείο απεικόνισης αρχικής σελίδας
- Αρχείο απεικόνισης σελίδας about
- Αρχείο σύνδεσης με βάση δεδομένων
- Αρχείο προσθήκης στοιχείων στην βάση δεδομένων
- Αρχείο επικοινωνίας με Facebook Application

Η σχετική ανάλυση του καθενός θα γίνει παρακάτω, για να αποδοθεί καλύτερα η δομή της σελίδας και της γενικότερης εφαρμογής.

Στο αρχείο της αρχικής σελίδας, στο οποίο έχει προστεθεί και ο κώδικας του Bootstrap, με την χρήση της PHP και ταυτόχρονα με την βοήθεια του αρχείου σύνδεσης με την βάση δεδομένων, δημιουργούμε αριστερά μία λίστα με όλους τους αποστολείς μηνυμάτων, στην οποία ο χρήστης μπορεί να διαλέξει να εμφανιστούν είτε όλα τα μηνύματα είτε μόνο του ανθρώπου που τον ενδιαφέρει ενώ έχουν δημιουργηθεί και τα κατάλληλα links προς την σελίδα About. Τέλος, στο κεντρικό κομμάτι της σελίδας δημιουργούνται δυναμικά κουτιά με τις κατάλληλες πληροφορίες από τα SMS προερχόμενα πάντα από την βάση δεδομένων.



Εικόνα 12 Εμφάνιση ιστοσελίδας SMStoAuto

Το αρχείο για την σελίδα about είναι μια απλή σελίδα που ακολουθεί το κλασικό σχέδιο των σελίδων about, χρησιμοποιώντας ξανά βέβαια των κώδικα του Bootstrap. Εκτός από την μπάρα περιήγησης στο υψηλότερο σημείο της οθόνης, η σελίδα έχει μόνο ένα μεγάλο κουτί στο οποίο με συνοπτικό κείμενο ενημερώνεται ο επισκέπτης για την μορφή της εφαρμογής, την δημιουργία της και την χρήση της.

Το αρχείο σύνδεσης με την βάση δεδομένων έχει μόνο μερικές γραμμές κώδικα, αλλά είναι πολύ σημαντικό καθώς μέσω αυτού η ιστοσελίδα μας ανακτά τα δεδομένα που έχουν φορτωθεί στην βάση δεδομένων από τον Arduino. Ουσιαστικά, το μόνο που κάνει είναι η σύνδεση στην κατάλληλη βάση δεδομένων και στον κατάλληλο πίνακα αυτής, καθώς και στην τοποθέτηση κατάλληλου charset ώστε τα δεδομένα να φαίνονται με σωστούς χαρακτήρες.

Το αρχείο προσθήκης δεδομένων είναι αυτό που καλείται ουσιαστικά από τον Arduino για να προστεθεί κάθε φορά ένα καινούργιο SMS. Ο κώδικάς του μπορεί να χωριστεί σε δύο σημαντικά κομμάτια, σε αυτό που γίνεται η εισαγωγή του SMS στην βάση δεδομένων και σε αυτό που γίνεται η δημοσίευση του SMS σε κατάλληλο profile του Facebook.

Το πρώτο γίνεται αρκετά απλά μέσα από το αρχείο της σύνδεσης με την βάση δεδομένων και των απλών εντολών της MySQL.

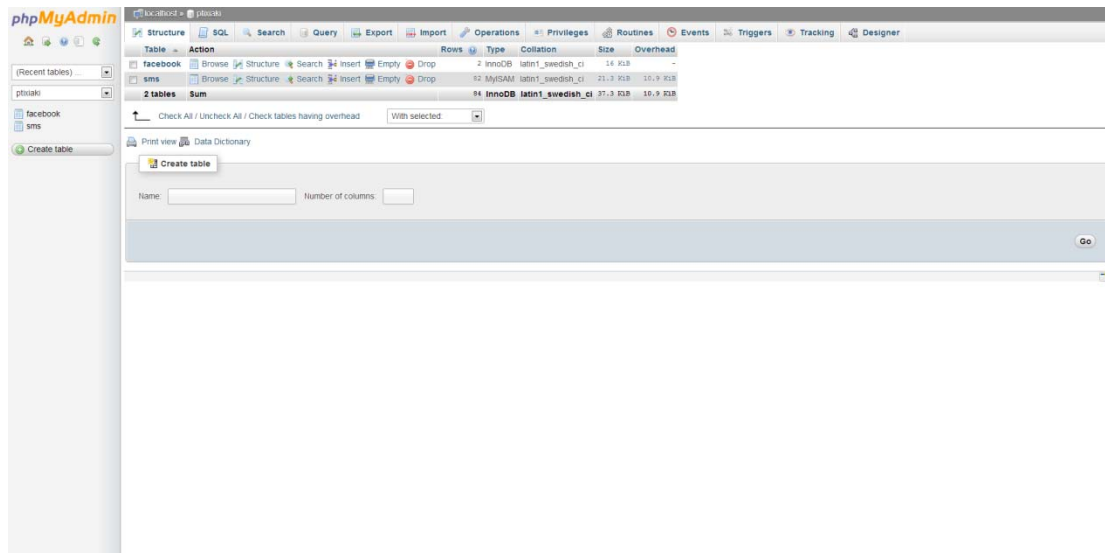
Το δεύτερο και πολύ πιο δύσκολο κομμάτι είναι η σύνδεση με το Facebook και η δημοσίευση του SMS στο κατάλληλο προφίλ. Το Facebook έχει υιοθετήσει το πρωτόκολλο διαπίστευσης χρήστη OAuth 2.0 σύμφωνα με το οποίο πρέπει να ακολουθηθεί μια διαδικασία για την δημοσίευση μηνύματος σε προφίλ χρήστη, την οποία θα την αναλύσουμε παρακάτω, μαζί με την δημιουργία του Facebook Application. Επιπλέον, σε αυτό το αρχείο χρησιμοποιήθηκε και το Facebook PHP SDK, το οποίο περιέχει τις κατάλληλες εντολές για την επικοινωνία με το Facebook Platform.

Το αρχείο επικοινωνίας με την Facebook εφαρμογή μας είναι πολύ σημαντικό για την σωστή λειτουργία του OAuth 2.0, το οποίο όμως θα αναλύσουμε παρακάτω. Η τελική δουλειά αυτού του αρχείου είναι να πιστοποιεί την ταυτότητα του κατόχου του προφίλ του Facebook, δηλαδή προφανώς του διαχειριστή ολόκληρης της εφαρμογής και να αποθηκεύει τα απαραίτητα στοιχεία ώστε να μπορεί η εφαρμογή να χειρίζεται το προφίλ του αυτόματα, ακόμα και όταν δεν είναι αυτός online. Η αποθήκευση των στοιχείων αυτών γίνεται στην βάση δεδομένων για ώστε να χρησιμοποιηθούν αργότερα, τα οποία όμως πρέπει να ανανεώνονται, όπως θα δούμε και

παρακάτω, μετά από κάποιο χρονικό διάστημα, αλλιώς η εφαρμογή θα σταματήσει να δουλεύει. Περισσότερη εμπάθυνση στην διασύνδεση με το Facebook θα γίνει στο κατάλληλο επόμενο τμήμα της πτυχιακής.

Ο σχεδιασμός της βάσης δεδομένων MySQL περιλαμβάνει μία βάση με δύο πίνακες. Ο πρώτος πίνακας είναι αυτός που περιέχει τα στοιχεία των μηνυμάτων SMS, έχει δηλαδή τέσσερις στήλες εκ των οποίων η πρώτη είναι ο χαρακτηριστικός αριθμός αναφοράς της γραμμής και οι άλλες τρεις το όνομα αποστολέα, η ώρα και το κείμενο του μηνύματος. Ο δεύτερος πίνακας χρησιμοποιείται για την σύνδεση με το Facebook και αποτελείται από τρεις στήλες, με την πρώτη να είναι ο χαρακτηριστικός αριθμός αναφοράς της γραμμής και τις άλλες δύο να κρατούν τις απαραίτητες πληροφορίες για την σύνδεση με το Facebook. Στο δοκιμαστικό περιβάλλον του τοπικού υπολογιστή μπορούμε να δημιουργήσουμε την βάση δεδομένων πολύ απλά μέσω του προεγκατεστημένου, λόγω XAMPP, διαχειριστικού εργαλείου phpMyAdmin.

Το phpMyAdmin είναι ένα ευρέως διαδεδομένο εργαλείο γραμμένο σε php που επιτρέπει την εύκολη διαχείριση των βάσεων δεδομένων μας ενώ όλες τις λειτουργίες όπως δημιουργία, προσθήκη και διαγραφή μπορούμε να τις κάνουμε είτε μέσω γραφικού περιβάλλοντος είτε μέσω των κλασικών εντολών της MySQL.



Εικόνα 13 Περιβάλλον εργασίας του phpMyAdmin

Την βάση δεδομένων δεν θα μπορέσουμε να την περάσουμε ως απλό αρχείο μέσω του WinSCP, όπως θα κάνουμε δηλαδή με τα αρχεία της rhr, οπότε θα πρέπει να την φτιάξουμε μόνοι μας αντίστοιχα και στο απομακρυσμένο σύστημα. Εκεί η δημιουργία της βάσης δεδομένων και των πινάκων της γίνεται μέσω του προγράμματος PuTTY, χρησιμοποιώντας της κατάλληλες εντολές της MySQL όπως για παράδειγμα :

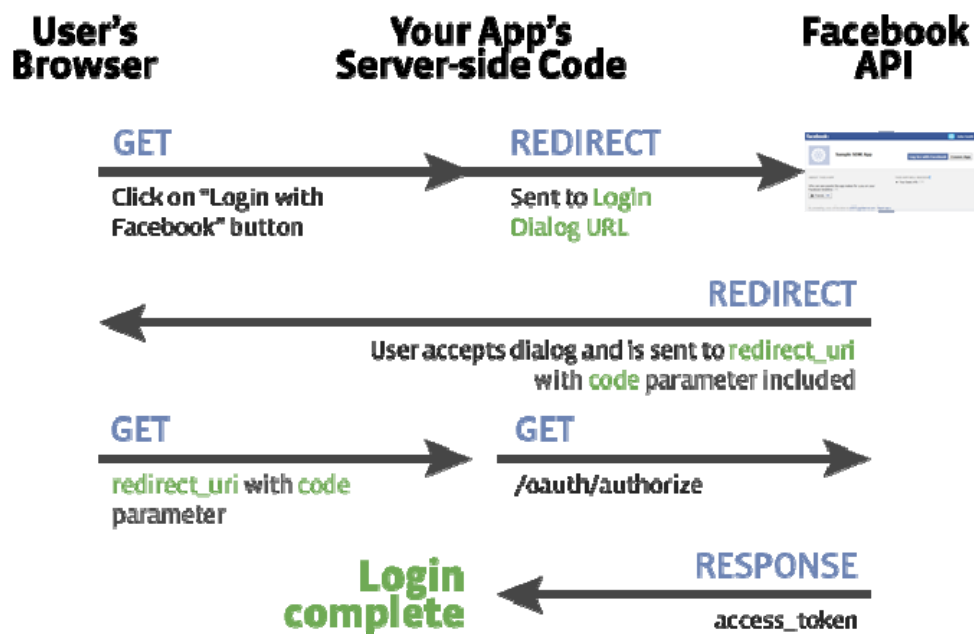
```
create database SMStoAuto
```

2.3.4 Δημιουργία και λειτουργία του Facebook Application

Τελευταίο κομμάτι της Web εφαρμογής μας αφήσαμε την διασύνδεση με το Facebook. Το Facebook είναι το μεγαλύτερο μέσο κοινωνικής δικτύωσης στο Internet και οι χρήστες του συνήθως το χρησιμοποιούν εντατικά καθ' όλη την διάρκεια της ημέρας. Για αυτό είναι ο ευκολότερος τρόπος να επικοινωνήσεις άμεσα με ένα μεγάλο κοινό. Για να μπορέσει η εφαρμογή μας να συνδεθεί με το Facebook όμως, θα πρέπει να δημιουργήσουμε ένα Facebook Application, το οποίο θα είναι, κατά κάποιο τρόπο, υπεύθυνο για την είσοδό μας και την πιστοποίηση του χρήστη μας.

Η δημιουργία ενός Facebook Application είναι μια απλή διαδικασία μερικών μόνο κλικ. Πηγαίνοντας στην κατάλληλη σελίδα του Facebook για τους προγραμματιστές εμφανίζεται η επιλογή της δημιουργίας νέας εφαρμογής. Μετά οι επιλογές είναι απλές όπως για παράδειγμα το όνομα που θέλουμε να έχει η εφαρμογή μας. Με το τέλος της διαδικασίας, μας εμφανίζεται ο πίνακας ελέγχου της εφαρμογής μας.

Τώρα που έχουμε στην κατοχή μας τα App ID και App Secret, μπορούμε να ξεκινήσουμε τον προγραμματισμό της εφαρμογής μας. Συγκεκριμένα, η εφαρμογή θα πρέπει να συνδέεται αυτόματα με τον λογαριασμό στο οποίο το προφίλ θα εμφανίζονται τα μηνύματα και τελικά να δημοσιεύει το κατάλληλο μήνυμα. Αυτό για να γίνει πρέπει να ακολουθηθεί μια διαδικασία που ονομάζεται Server-Side Login και παρακάτω φαίνεται το λογικό της διάγραμμα.



Εικόνα 15 Διάγραμμα ροής Server-Side Login

Όπως βλέπουμε, τα βήματα της διαδικασίας είναι αρκετά διακριτά όμως απαιτούν προσοχή. Αρχικά, ο χρήστης θα πρέπει να επιλέξει κατάλληλο κουμπί τύπου " Login with Facebook " το οποίο θα τον μεταφέρει στο γνωστό κουτί διαλόγου που εμφανίζει το Facebook για την εισαγωγή του χρήστη στο δίκτυό του. Μόλις ο χρήστης δεχτεί την εισαγωγή του στο Facebook, αυτό

επιστρέφει στην εφαρμογή μας μία παράμετρο `code`, η οποία θα μας βοηθήσει στα επόμενα βήματα.

Τώρα, η εφαρμογή μας στέλνει άλλο ένα αίτημα προς το Facebook, αυτή την φορά μαζί με την παράμετρο `code` που μόλις λάβαμε όπως και τα στοιχεία της Facebook Application μας, δηλαδή τα App ID και App Secret. Το Facebook τότε, αν ο χρήστης επισκέπτεται για πρώτη φορά την συγκεκριμένη εφαρμογή, τον ενημερώνει για τις δικαιοδοσίες που απαιτεί και περιμένει την αποδοχή του. Αν ο χρήστης τις έχει ήδη αποδεχθεί προηγούμενη φορά, προχωράει κατευθείαν στην αποστολή πίσω στην εφαρμογή του Access Token, με το οποίο πια η εφαρμογή μας θα μπορεί να διαχειριστεί το προφίλ του χρήστη αυτόματα, πάντα σύμφωνα με τις δικαιοδοσίες που συμφωνήθηκαν. Στην δικιά μας περίπτωση, στο Facebook Application έχει προστεθεί η δικαιοδοσία την αυτόματης δημοσίευσης ενώ εδώ οι επιλογές είναι πάρα πολλές και περιλαμβάνουν όλες τις πτυχές του προφίλ του χρήστη, όπως πχ την πρόσβαση στην λίστα φίλων.

Σημαντική λεπτομέρεια σε αυτό το σημείο είναι πως το Access Token έχει συγκεκριμένη διάρκεια ζωής, η οποία μάλιστα είναι πολύ μικρή. Μέσω ενός προγραμματιστικού κόλπου όμως, αφού έχουμε στην κατοχή μας ένα κανονικό Access Token μπορούμε να το ανταλλάξουμε για ένα αντίστοιχο μακράς διάρκειας, με χρόνο ζωής τις 60 ημέρες. Μετά τις 60 ημέρες, η εφαρμογή θα πάψει να έχει τον έλεγχο του προφίλ του χρήστη και αυτός θα πρέπει να ξανακάνει την διαδικασία από την αρχή.

Συγκεκριμένα όμως, η εφαρμογή μας είναι έτσι φτιαγμένη ώστε το Access Token να αποθηκεύεται σε κατάλληλο πίνακα βάσης δεδομένων, όπως είδαμε

σε προηγούμενο σημείο της εργασίας, έτσι αν ο χρήστης επισκέπτεται την σελίδα της εφαρμογής που εκτελεί όλη την διαδικασία τουλάχιστον μια φορά ανά 60 μέρες, τότε το Access Token θα ανανεώνεται συνεχώς και η εφαρμογή θα συνεχίσει να δουλεύει χωρίς προβλήματα.

Ο παραπάνω διάλογος μεταξύ εφαρμογής του server μας με αυτή του Facebook στηρίζεται πάνω στο πρότυπο OAuth 2.0. Το πρότυπο αυτό δημιουργήθηκε για να καλύψει την ανάγκη για την μη παράδοση των προσωπικών μας κωδικών σε τρίτες εταιρείες ή εφαρμογές. Δηλαδή, τώρα η εφαρμογή που φτιάξαμε στο Facebook θα μπορεί να ελέγχει το profile μας χωρίς ο δημιουργός της (στην συγκεκριμένη περίπτωση το ίδιο άτομο, ο γενικός κανόνας όμως είναι πως ο δημιουργός θα είναι κάποιος ξένος) να γνωρίζει τα προσωπικά στοιχεία πρόσβασής μας. Με σταθερούς ρυθμούς όλες οι μεγάλες ιστοσελίδες έχουν ξεκινήσει ή υλοποιήσει την ενσωμάτωση του OAuth 2.0 πρωτοκόλλου, το οποίο μάλιστα χαρακτηρίζεται από την ευελιξία του ώστε να μπορεί να χρησιμοποιηθεί σε διάφορες βαθμίδες ασφάλειας, πάντα με γνώμονα τις απαιτήσεις της εκάστοτε εφαρμογής.



Εικόνα 16 Πρωτόκολλο OAuth 2.0

Από την στιγμή που παραλάβουμε το Access Token, και για όσο καιρό έχουμε στην διάθεσή μας κάποιο που η διάρκεια ζωής του να μην έχει περάσει, μπορούμε με ανάλογες εντολές να χειριστούμε πτυχές του profile μας. Στην περίπτωση της εφαρμογής μας, όπως αναφέρθηκε παραπάνω, χρησιμοποιήθηκε το Facebook SDK για την PHP (έκδοση 3.2.2). Με τον όρο SDK (Software Development Kit) χαρακτηρίζουμε ένα σετ προγραμματιστικών εργαλείων που βοηθούν ή επιτρέπουν την δημιουργία εφαρμογών για ένα συγκεκριμένο σύστημα, στην δικιά μας περίπτωση το Facebook. Αντίστοιχα, SDK κυκλοφορούν για την δημιουργία λογισμικού που μπορεί να προορίζεται για κονσόλες βιντεοπαιχνιδιών ή για συγκεκριμένα λειτουργικά συστήματα ή για οποιοδήποτε συγκεκριμένο σύστημα. Συνήθως τα SKD δημιουργούνται από τους ίδιους τους προγραμματιστές της πλατφόρμας ή του συστήματος του οποίου τον προγραμματισμό θα βοηθάει το SDK, όπως άλλωστε και στην περίπτωση του Facebook.

Σύμφωνα με το Facebook SDK για την PHP, το υπεύθυνο για την δημοσίευση αρχείο πρέπει να ελέγξει στην βάση δεδομένων για το ποιο πρόσφατο Access Token και με αυτό να δημιουργήσει ένα αντικείμενο τύπου Facebook, το οποίο αποτελείται από στοιχεία όπως το Application ID και το Application Secret της εφαρμογής μας, μαζί με κάποιες έξτρα επιλογές.

Ύστερα, δημιουργούμε το πακέτο που θα στείλουμε στο Facebook, το οποίο αποτελείται από το Access Token και ότι άλλο στοιχείο χρειαζόμαστε, όπως για παράδειγμα, το κείμενο που θέλουμε να δημοσιευτεί. Τελευταίο βήμα είναι η ίδια η αποστολή του πακέτου με τις πληροφορίες, στην οποία σημειώνουμε και την λειτουργία που θέλουμε να επιτελεστεί, στην δικιά μας περίπτωση φυσικά είναι η δημοσίευση.

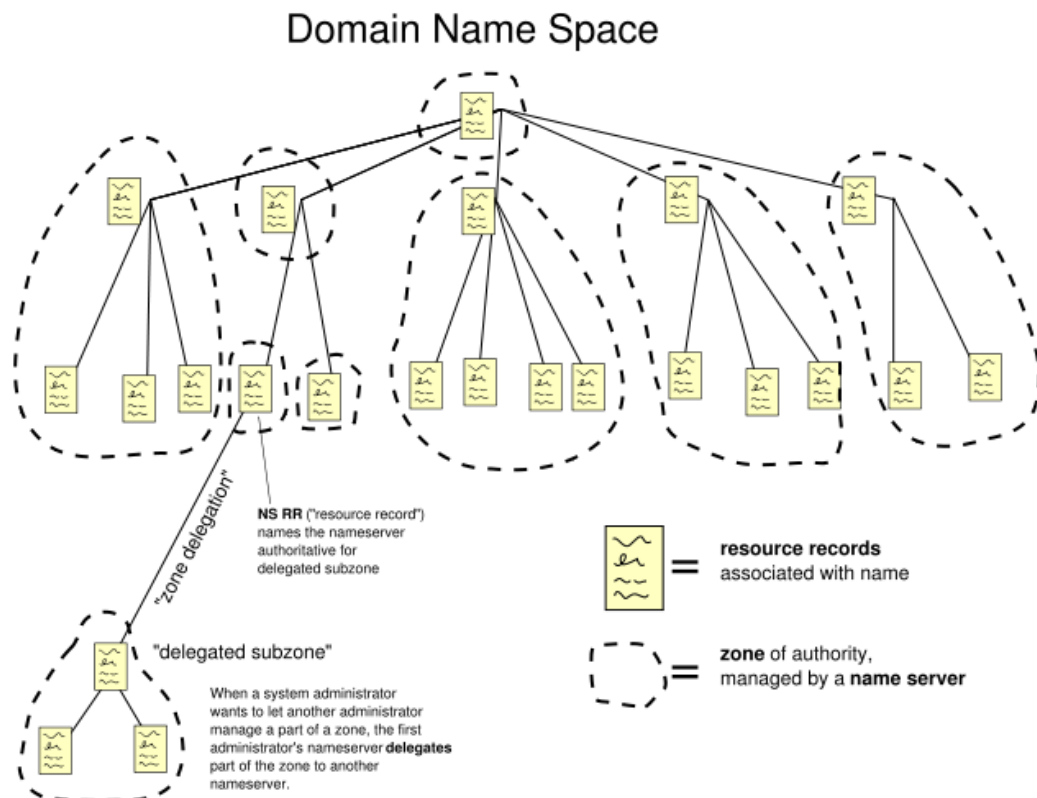
Μετά και από αυτό το βήμα, έχουμε ολοκληρώσει την δημιουργία της ιστοσελίδας μας. Το τελικό αποτέλεσμα είναι πλήρως λειτουργικό και χάρη στο Twitter Bootstrap είναι ευπρόσιτο τόσο από σταθερούς υπολογιστές όσο και από κινητές συσκευές. Ουσιαστικά είναι το πρόσωπο της εφαρμογής, καθώς ο τελικός χρήστης δεν μπορεί να ξέρει την δουλειά που εκτελείται στο μηχανικό κομμάτι της εφαρμογής, δηλαδή στον Arduino, αλλά βλέπει κατευθείαν το αποτέλεσμα αυτής μέσα από αυτήν την σελίδα.

Τελευταίο βήμα φυσικά είναι η μεταφορά των τοπικών αρχείων στον απομακρυσμένο υπολογιστή με την χρήση του WinSCP που είδαμε σε προηγούμενο κεφάλαιο. Αφού ολοκληρωθεί η μεταφορά τους στον σωστό φάκελο, μπορούμε πια να προσπελάσουμε το site μας πληκτρολογώντας την IP διεύθυνση του απομακρυσμένου υπολογιστή στην μπάρα διευθύνσεων ενός browser.

2.3.5 Διαχείριση Domain Name

Αν και από άποψη λειτουργικότητας , η ιστοσελίδα μας είναι έτοιμη, προφανώς η χρήση της διεύθυνσης IP για την είσοδο σε αυτήν είναι πολύ δύσχρηστη μέθοδος.

Για αυτό, θα χρειαστεί η αγορά και η χρήση ενός Domain Name ή όπως λέγεται στα ελληνικά, ενός Ονόματος Τομέα. Τα Domain Names είναι μία σειρά από χαρακτήρες, στους οποίους περιλαμβάνονται μόνο τα αλφαριθμητικά στοιχεία και οι παύλες, τα οποία δημιουργούνται και απαραίτητα ακολουθούν τους κανόνες του Domain Name System ή αλλιώς DNS και ουσιαστικά μπορούν να χρησιμοποιηθούν αντί μιας διεύθυνσης του πρωτοκόλλου IP. Το DNS είναι ένα ιεραρχικό σύστημα ονοματοδοσίας για δίκτυα υπολογιστών που χρησιμοποιούν το πρωτόκολλο IP, με δενδρική δομή.



Εικόνα 17 Ιεραρχία και δομή DNS

Χάρη σε αυτό γίνεται η αντιστοίχιση των αριθμητικών διευθύνσεων με κανονικά ονόματα, μια ανάγκη που δημιουργήθηκε από την αδυναμία του ανθρώπου να θυμάται κάτι το οποίο δεν είναι φανερό το νόημά του. Τα Domain Names που αντιστοιχούν στον παγκόσμιο ιστό ξεκινάνε αναγκαστικά με το γνωστό www. (World Wide Web) ακολουθούμενο από την σειρά χαρακτήρων που επιλέγει ο χρήστης και τέλος την κατάληξη που επιθυμεί, όπως για παράδειγμα το γνωστό .com .

Παλιότερα αυτές οι καταλήξεις ήταν λίγες και συγκεκριμένες, τώρα όμως έχουν αυξηθεί πάρα πολύ και χωρίζονται κυρίως σε δύο μεγάλες κατηγορίες, την gTLD (generic top-level domain) η οποία περιλαμβάνει καταλήξεις όπως .net (προορίζεται για χρήση από υποδομές δικτύων), .edu (προορίζεται για

χρήση από ινστιτούτα ανώτερης εκπαίδευσης με έδρα την Αμερική) και .org (προορίζεται για χρήση από οργανισμούς που δεν περιγράφονται από πιο συγκεκριμένες κατηγορίες), και την ccTLD (country code top-level domain) η οποία περιλαμβάνει τις κατάληξεις υποδηλώνουν την προέλευση από συγκεκριμένη χώρα, όπως χαρακτηριστικά το .gr .

Η χρήση που προορίζεται για την κάθε κατάληξη domain στις περισσότερες των περιπτώσεων είναι απλά προαιρετική, αλλά υπάρχουν και μερικές που θέτουν προαπαιτούμενα για να τις αποκτήσεις. Ένα Domain Name δεν μπορεί να αγοραστεί από κανέναν, παρά μόνο να παραχωρηθεί έναντι χρηματικού ποσού για συγκεκριμένο χρονικό διάστημα κατά το οποίο μπορεί ο χρήστης να το διαχειριστεί όπως θέλει. Η διαδικασία αυτή γίνεται μέσω των Domain name registrar, δηλαδή οργανισμών που μπορούν να δεσμεύσουν ένα Domain Name έναντι αμοιβής.



Εικόνα 18 Godaddy : ο μεγαλύτερος Domain name registrar

Για την εφαρμογή μας, μέσω ενός τέτοιου οργανισμού, επιλέχθηκε, αφού ελέγχθηκε πως δεν έχει ήδη καταχωρηθεί σε άλλον χρήστη, το Domain Name smstoauto.info . Η μίσθωση του Domain Name μας δίνει πρόσβαση σε έναν πίνακα ελέγχου από τον οποίο μπορούμε να αλλάξουμε διάφορες ρυθμίσεις για το νεοαποκτηθέν Domain μας.

Εδώ θα χρειαστεί να περάσουμε τις κατάλληλες ρυθμίσεις ώστε να αντιστοιχήσουμε το domain μας στην γνωστή μας IP του απομακρυσμένου υπολογιστή. Αυτό θα γίνει μέσω του DSN Manager (σύμφωνα με τον Domain name registrar που επιλέγει ο καθένας, τα ονόματα των εργαλείων αυτών μπορεί να διαφέρουν αλλά η τελική λειτουργία τους παραμένει η ίδια) .

Μέσα στον Domain Manager μας δίνεται μια λίστα από επιλογές όπως οι παρακάτω.

- A Record

Αλλιώς και Address Record, είναι υπεύθυνο για την σύνδεση του domain name με μία διεύθυνση IP.

- AAAA Record

Αντίστοιχο με το A Record, μόνο που χρησιμοποιείται για την σύνδεση με διεύθυνσης του πρωτοκόλλου IP Version 6.

- CNAME Record

Διαφορετικά και Canonical Name Record, δημιουργεί ένα ψευδώνυμο το οποίο μπορεί να συνδεθεί και να αντιστοιχηθεί με ένα κανονικό domain name.

- MX Record

Αλλιώς και Mail eXchanger Record, χρησιμοποιείται για να καθορίσει τον server στο Internet που θα διαχειρίζεται τα e-mails για το συγκεκριμένο domain name.

Στην δική μας περίπτωση φτάνει η ρύθμιση μονάχα του A Record, στο οποίο πρέπει να τοποθετήσουμε την διεύθυνση του απομακρυσμένου υπολογιστή. Μετά από αυτή την ρύθμιση, θα πρέπει να περιμένουμε μέχρι να ενημερωθούν οι Name Servers του διαδικτύου, δηλαδή οι servers που αποθηκεύουν τις διάφορες αντιστοιχίες Domain με IP. Αυτή η διαδικασία μπορεί να κρατήσει μερικά λεπτά έως και δύο ημέρες, στο διάστημα το οποίο ο χρήστης για να ελέγχει την εξέλιξη της διαδικασίας μπορεί είτε να δοκιμάζει να προσπελάσει την ιστοσελίδα του με το domain είτε να χρησιμοποιήσει κάποιο διαδικτυακό εργαλείο που να υποστηρίζει το DNS Lookup, μια υπηρεσία που μπορεί να εμφανίσει πάρα πολλές πληροφορίες για ένα domain name, όπως τα Records που έχουν ρυθμιστεί για αυτό.



Εικόνα 19 What's My DNS?, χρήσιμο εργαλείο για αλλαγές DNS

Μετά και από την ενημέρωση των Name Servers, η συνολική εγκατάσταση της ιστοσελίδας μας έχει φτάσει στο τέλος της. Τώρα πια μπορούμε να προσπελάσουμε την σελίδα χρησιμοποιώντας το καινούργιο μας Domain Name και να χρησιμοποιήσουμε όλες τις λειτουργίες της εφαρμογής.

2.4 Εγκατάσταση και ολοκλήρωση κατασκευής

Η εφαρμογή μας αυτή τη στιγμή μπορεί και εκπληρώνει όλους τους στόχους που είχαμε δώσει πριν την αρχή της υλοποίησής της. Όμως, επειδή εκτός του software, η εφαρμογή μας περιλαμβάνει και hardware κομμάτι, κρίνεται αναγκαίο η σωστή εγκατάστασή του σε ένα κουτί κατασκευής που να επιτρέπει αφενός την εύκολη μεταφορά της αφετέρου την ασφαλή και αδιάκοπτη χρήση της, καθώς η λειτουργία απροστάτευτων ηλεκτρονικών

εξαρτημάτων δεν είναι ενδεδειγμένη. Παρακάτω, θα περιγράψουμε εν συντομία την διαδικασία που ακολουθήσαμε για την σωστή εγκατάσταση των εξαρτημάτων μας όπως και το τελικό προϊόν.

Αρχικά, για να γίνει η σωστή επιλογή κατάλληλου κουτιού, πρέπει να υπολογιστούν οι επιθυμητές διαστάσεις του. Για αυτό, πρέπει να ελέγξουμε τις διαστάσεις των επιμέρους εξαρτημάτων και να συνυπολογίσουμε παράγοντες όπως την αναμενόμενη θερμοκρασία, την σωστή ροή του αέρα και την κατάλληλη τοποθέτησή τους. Στον παρακάτω πίνακα μπορούμε να δούμε τις διαστάσεις του κάθε εξαρτήματος χωριστά.

Εξάρτημα	Μήκος (mm)	Πλάτος (mm)	Ύψος (mm)
Arduino MEGA2560	108	53	15
Ethernet Shield	73	54	17
GSM Module	100	58	16
LCD Module	80	36	12

Πίνακας 2 Διάγραμμα Συνδέσεων

Σύμφωνα με τον παραπάνω πίνακα και την λογική διαρρύθμιση που θα τηρηθεί μέσα στο κουτί, καθώς προφανώς οι πλευρές των εξωτερικών συνδέσεων των εξαρτημάτων θα πρέπει να τοποθετηθούν στην πίσω πλευρά ενώ η οθόνη θα πρέπει να τοποθετηθεί στην μπροστινή πλευρά του κουτιού, μπορούμε να έρθουμε στο συμπέρασμα πως η διαστάσεις του κουτιού πρέπει να είναι τουλάχιστον 150 x 130 x 40 (μήκος , πλάτος , ύψος αντίστοιχα). Για την κατασκευή μας επιλέχθηκε ένα ελαφρώς μεγαλύτερο κουτί, φτιαγμένο από

αλουμίνιο, με διαστάσεις 198 x 160 x 50 και ανοιχτούς αεραγωγούς στις δύο πλευρές του.



Εικόνα 20 Κουτί κατασκευών

Για να εγκατασταθούν σωστά και σταθερά στο εσωτερικό του τα εξαρτήματά μας θα πρέπει να τα βιδώσουμε στις θέσεις τους. Ευτυχώς, τόσο ο Arduino, όσο και τα GSM Module και LCD Module έχουν κατάλληλες τρύπες στις πλακέτες τους για αυτόν ακριβώς τον λόγο. Στον Arduino αρκούν δύο βίδες για να σταθεροποιηθεί στο πάτωμα του κουτιού, στο οποίο προστίθενται άλλες τέσσερις για το GSM Module, ενώ η οθόνη χρειάζεται άλλες τέσσερις βίδες, αυτή την φορά όπως είναι λογικό στην πρόσοψη του κουτιού.

Τελευταίο βήμα για την εγκατάσταση της εφαρμογής μας είναι το κατάλληλο άνοιγμα τρυπών στην μπροστινή και στην οπίσθια πλευρά του κουτιού, ώστε από την μία να είναι εμφανής η οθόνη TFT και από την άλλη να είναι προσβάσιμες οι εξωτερικές θύρες σύνδεσης των εξαρτημάτων αλλά και η

κεραία του GSM. Μερικά από τα εργαλεία που χρησιμοποιήθηκαν για τις παραπάνω εργασίες είναι τα παρακάτω.

- Δράπανο χειρός
- Μέτρο
- Τροχός
- Ηλεκτρονικό Παχύμετρο
- Τρυπάνι



Εικόνα 21 Εργαλεία παραμετροποίησης κουτιού

Μετά και από αυτό το βήμα, η παραμετροποίηση του κουτιού έχει τελειώσει και μαζί του και η σωστή εγκατάσταση του hardware της εφαρμογής μας. Η παρακάτω δύο φωτογραφίες απεικονίζουν την τελική εικόνα του κουτιού με εγκατεστημένο το hardware μας.



Εικόνα 22 Εμπρόσθιο μέρος της κατασκευής μας

Αξίζει να αναφέρουμε και τις συνδέσεις τις οποίες παρέχει η εφαρμογή μας. Από τα αριστερά όπως φαίνεται και στην εικόνα 21, υπάρχει η σύνδεση για τροφοδοτικό που προορίζετε για το GSM Module, όπως και η θύρα για σειριακή επικοινωνία σε περίπτωση που θέλουμε να το συνδέσουμε με κάποιον υπολογιστή. Από τα δεξιά βλέπουμε την θύρα Ethernet στην οποία πρέπει να συνδέσουμε αντίστοιχο καλώδιο για την παροχή σύνδεσης στο διαδίκτυο ενώ από κάτω της βλέπουμε την θύρα τροφοδοσίας του Arduino και την θύρα USB τύπου B η οποία μπορεί να χρησιμοποιηθεί είτε για τον προγραμματισμό του Arduino είτε για την τροφοδοσία του, καθώς η λειτουργία της LCD οθόνης δεν χρειάζεται πολύ ρεύμα, οπότε η παροχή της USB αρκεί.



Εικόνα 23 Οπίσθιο μέρος της κατασκευής μας

Κεφάλαιο 3

Εφαρμογές , Προβλήματα και Βελτιώσεις

3.1 Πηγή αναφοράς και πιθανή εφαρμογή της εργασίας

Η αρχική ιδέα πίσω από την εργασία μας προήλθε από ένα άρθρο του Erico Guizzo στον ιστότοπο του περιοδικού IEEE Spectrum. Το περιοδικό αυτό κυκλοφορεί υπό την αιγίδα του IEEE (Institute of Electrical and Electronics Engineers ή αλλιώς Ινστιτούτο Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών) και μπορεί ο καθένας, επαγγελματίας ή μη, να στείλει το δικό του άρθρο πάνω σε ένα θέμα του χώρου και αυτό μετά από μία διαδικασία αξιολόγησης μπορεί τελικά να δημοσιευθεί.

Ο Erico Guizzo είχε γράψει ένα άρθρο για την κατασκευή μιας παρόμοιας εφαρμογής με την δικιά μας, μόνο που εκεί ο αποστολές χρησιμοποιούσε το δημοφιλές δίκτυο Twitter ενώ τα μηνύματά του εμφανίζονταν έξω από το γραφείο του σε μια TFT οθόνη.

Έχοντας ως αρχική ιδέα το συγκεκριμένο άρθρο θελήσαμε να το διευρύνουμε και να το μεταλλάξουμε έτσι ώστε να καλύπτει ένα πρόβλημα της καθημερινότητάς μας. Αυτό είναι η επικοινωνία των καθηγητών μιας σχολής

με τους φοιτητές τους για πράγματα απλά, που όμως αν μεταδοθούν έγκαιρα μπορούν να τους γλιτώσουν από αχρείαστο κόπο. Κάποια από τα βασικά παραδείγματα άλλωστε που στοχοθέτησαν την δημιουργία της παρούσας εφαρμογής ήταν η πιθανή καθυστέρηση ενός καθηγητή ή μία ξαφνική αναβολή του μαθήματός του.

Παραδοσιακά, η ενημέρωση των φοιτητών γίνεται συνήθως μέσα από το site της σχολής ή του μαθήματος, μόνο που σε πολλές περιπτώσεις, αναγκαστικά είτε η πρόσβαση του καθηγητή στο Internet σε δεδομένη στιγμή δεν είναι εύκολή είτε η τελική ενημέρωση του φοιτητή είναι εκτός χρόνου, γιατί δυστυχώς δεν υπάρχει κάποιο σύστημα αυτόματης ενημέρωσης για νέες δημοσιεύσεις, οπότε μπορεί ο φοιτητής να καθυστερήσει να ελέγξει το κατάλληλο site ή ακόμα και να μην το κάνει ποτέ.

Για την λύση του πρώτου μισού του παραπάνω προβλήματος, στην δημιουργία της εργασίας αυτής επιλέχθηκε η αποστολή μηνύματος SMS για τον αποστολέα, στην οποία υπηρεσία ο καθένας έχει πρόσβαση από παντού αρκεί να είναι χρήστης κινητής τηλεφωνίας. Για την επίλυση του δεύτερου μισού, επιλέχθηκε η χρήση της δημοφιλέστερης ιστοσελίδας social media, Facebook, στην οποία σχεδόν κατά κανόνα όλοι οι φοιτητές, και όχι μόνο, διατηρούν λογαριασμό και προσπαθούν να ενημερώνονται συνεχώς από αυτόν. Προφανώς, η χρήση της εφαρμογής δεν ενδείκνυται για μεγάλες ενημερώσεις όπως για παράδειγμα η ανακοίνωση μιας νέας άσκησης, αφού και το κείμενο είναι περιορισμένο (ένα SMS δεν μπορεί να ξεπεράσει τους 120 χαρακτήρες) και δεν υπάρχει η δυνατότητα για αποστολή αρχείων. Ενώ, σε προβλήματα περιπτώσεων όπως των παραδειγμάτων της προηγούμενης παραγράφου, η εφαρμογή μας θα μπορούσε να αποδειχθεί μία άκρως ικανοποιητική λύση.

3.2 Προβλήματα που παρουσιάστηκαν

Κατά την διάρκεια της δημιουργίας της πτυχιακής αυτής εμφανίστηκαν και χρειάστηκαν να αντιμετωπιστούν αρκετά προβλήματα, τόσο στο κομμάτι του υλικού όσο και στο κομμάτι του λογισμικού.

- Αρχικό πρόβλημα ήταν η εύρεση και η αγορά των κατάλληλων εξαρτημάτων, καθώς στην ελληνική αγορά είτε η ποικιλία είναι μικρή είτε υπερτιμημένη, με μόνη αναγκαστική διέξοδο την επιλογή της παραγγελίας από το εξωτερικό. Αν και αυτή η κίνηση δίνει πρόσβαση σε άπειρα εργαλεία, εντούτοις δημιουργεί το πρόβλημα της καθυστέρησης, καθώς η αποστολή των εξαρτημάτων από την μία χώρα στην άλλη αναγκαστικά είναι μια χρονοβόρα διαδικασία.
- Ένα περίεργο πρόβλημα που παρουσιάστηκε στην συνέχεια αλλά δεν βρέθηκε ποτέ η αιτία του, είναι η σχετική αδυναμία του GSM Module να δουλέψει με τις κάρτες SIM όλων των εταιρειών. Συγκεκριμένα, με κάρτα SIM της εταιρείας Vodafone λειτουργεί αψεγάδιαστα, την στιγμή που με κάρτα SIM της εταιρείας Cosmote χάνει συχνά το σήμα σε περιοχή όμως που δεν υπάρχει τέτοιο πρόβλημα. Η περιέργεια του προβλήματος απορρέει από το γεγονός πως και οι δύο εταιρείες χρησιμοποιούν το ίδιο εύρος συχνοτήτων του GSM πρωτοκόλλου. Προφανώς αυτό επιλύθηκε με την επιλογή της βέλτιστης κάρτας SIM.

- Γενικότερο πρόβλημα ήταν η ανεύρεση επαρκούς βοήθειας για την κατασκευή της πτυχιακής, ειδικά στο κομμάτι του υλικού. Φυσικά, χρησιμοποιώντας ως πηγή πληροφοριών το Internet μπορέσαμε να βρούμε όσα πράγματα χρειαζόμασταν.
- Σημαντικό πρόβλημα που παρουσιάστηκε ήταν η λάθος επιλογής μοντέλου Arduino, καθώς στην αρχή ο σχεδιασμός έγινε γύρω από τον Arduino Duemilanove, ένα παλιότερο μοντέλο Arduino το οποίο επιτρέπει κώδικα που να καταλαμβάνει λιγότερο από 30Kbytes, μέγεθος που το δικό μας πρόγραμμα το ξεπέρασε. Λύση σε αυτό έδωσε η αγορά του Arduino MEGA2560, ο οποίος επιτρέπει πολλαπλάσιο μέγεθος κώδικα.
- Άλλο ένα πρόβλημα είναι η ενσωμάτωση του OAuth2.0 στην πλατφόρμα του Facebook, καθώς η διαδικασία που χρειάστηκε τελικά με αυτήν που χρειαζόταν χωρίς το OAuth είναι πάρα πολύ διαφορετική ενώ τώρα είναι απαραίτητη η επαναλαμβανόμενη, έστω και αραιά, επίσκεψη του διαχειριστή της εφαρμογής, στην κατάλληλη σελίδα ώστε να συνεχίσει να λειτουργεί.
- Τελευταίο πρόβλημα, αν μπορεί να χαρακτηριστεί έτσι, είναι η έλλειψη ασφάλειας στην χρήση της εφαρμογής. Αυτό δεν είναι ακριβώς πρόβλημα καθώς εξ αρχής ο σκοπός ήταν η λειτουργία της και όχι η διαφύλαξη της από την κακόβουλη χρήση, για την οποία θα πρέπει

να παρθούν διάφορα μέτρα όπως για παράδειγμα η υλοποίηση συστήματος αναγνώρισης του χρήστη (Arduino) κατά την διάρκεια της εισαγωγής δεδομένων στην βάση MySQL, καθώς τώρα ο οποιοσδήποτε που γνωρίζει ποιο αρχείο πρέπει να προσπελάσει μπορεί να ανεβάσει το δικό του μήνυμα.

- Πέραν των παραπάνω, υπάρχει και ένα πρόβλημα για το οποίο δεν βρέθηκε κάποια λύση πέρα από την προφανή. Συγκεκριμένα, μία κάρτα SIM μπορεί να λειτουργήσει χωρίς να της κάνουν ανανέωση μονάδων, όπως δηλαδή στην δική μας περίπτωση, για έναν χρόνο. Μετά από αυτόν τον χρόνο το νούμερο σταματάει αυτόματα να λειτουργεί. Έτσι, είναι αναγκαίο μια φορά τον χρόνο ο διαχειριστής της εφαρμογής να βγάζει την κάρτα SIM από το GSM Module και μέσω ενός κινητού τηλεφώνου να κάνει μια ανανέωση μονάδων ανεξαρτήτου ποσού, ώστε να μην σταματήσει να λειτουργεί το νούμερο.

3.3 Πιθανές βελτιώσεις και παραμετροποιήσεις

Η εφαρμογή που κατασκευάσαμε έχει πολλές δυνατότητες επέκτασης και βελτίωσης, είτε με την προσθήκη έξτρα hardware είτε με την αλλαγή του software μέρους της. Παρακάτω ακολουθούν μερικές προτάσεις που θα μπορούσαν εύκολα να υλοποιηθούν αν χρειαστεί.

- Μία καλή μετατροπή που όμως εύκολα θα μπορούσε να μετατραπεί σε ένα καινούργιο πρόβλημα, είναι η αμφίδρομη επικοινωνία. Η εφαρμογή μας έχει σχεδιαστεί έτσι ώστε να λαμβάνει τα SMS μόνο συγκεκριμένων ανθρώπων. Αντί αυτού, θα μπορούσε να λαμβάνει και SMS φοιτητών, τα οποία να κατευθύνονται σε αντίστοιχα μέσα στα οποία θα μπορεί ο καθηγητής εύκολα να αποκτήσει πρόσβαση. Αυτό όπως γίνεται εύκολα αντιληπτό, μπορεί να καταλήξει σε πρόβλημα, αν η χρήση του από τους φοιτητές δεν είναι η κατάλληλη.
- Η χρήση της κάρτας microSD μετατρέπει την κατασκευή σε ένα τύπο Data Logger, καθώς μπορεί και αποθηκεύει εκεί μέσα δεδομένα που έρχονται σε αυτό, μέσω του GSM. Η ανάγνωση αυτών των δεδομένων μπορεί να γίνει αυτή τη στιγμή μόνο με έναν τρόπο, βγάζοντας την κάρτα από την κατασκευή και συνδέοντάς την σε έναν ηλεκτρονικό υπολογιστή, ώστε να μπορεί ο χρήστης να δει τα περιεχόμενά της. Μία εναλλακτική λύση θα ήταν η δημιουργία μέσω

του Ethernet Module ενός Server ο οποίος να επιτρέπει στον χρήστη που τον επισκέπτεται να προσπελάσει τα αρχεία αυτά.

- Τώρα το web κομμάτι της εφαρμογής περιορίζεται στην ιστοσελίδα www.smstoauto.info και στο αντίστοιχο προφίλ στην ιστοσελίδα Facebook. Με αντίστοιχο τρόπο με αυτόν που αναλύσαμε σε προηγούμενο κεφάλαιο, μπορούμε να επιτρέψουμε την δημοσίευση των μηνυμάτων και σε άλλα διαδικτυακά media, όπως το Twitter και το LinkedIn, για να διευρύνουμε ακόμα περισσότερο το δυνητικό κοινό μας.
- Μία γρήγορη μετατροπή που θα βελτίωνε πολύ την ευχέρεια της ενημέρωσης θα ήταν η υποστήριξη ενός email server, ο οποίος θα δεχόταν μια λίστα από emails στα οποία η εφαρμογή μας θα έστελνε απευθείας είτε την ενημέρωση ότι έχει έρθει κάποιο νέο SMS, ώστε να επισκεφτούν την ιστοσελίδα και να ενημερωθούν, είτε το ίδιο το SMS.
- Τέλος, αν και η χρήση της ιστοσελίδας από mobile συσκευές είναι ήδη αρκετά εύχρηστη, λόγω του Bootstrap, ενδιαφέρον θα ήταν η δημιουργία εφαρμογών για Android και iPhone που να ενημερώνουν, ας πούμε με ηχητικό τόνο, όταν εμφανιστεί νέο SMS.

ΒΙΒΛΙΟΓΡΑΦΙΑ – ΠΗΓΕΣ ΠΛΗΡΟΦΟΡΙΩΝ

Arduino

- Arduino , *Learning + Reference* , <http://www.arduino.cc>
- Maik Schmidt , *Arduino: A Quick Start Guide (Pragmatic Programmers)* , ISBN : 978-1-934-35666-1
- Massimo Banzi , *Getting Started with Arduino* , ISBN : 978-0-596-15551-3
- Michael Margolis , *Arduino Cookbook* , ISBN : 978-1-449-31387-6
- Harold Timmis , *Practical Arduino Engineering* , ISBN : 978-1-430-23885-0
- Freeduino , *Index of Arduino & Freeduino Knowledge* , <http://www.freeduino.org>
- Siemens , *AT Command Set* , TC35 Manual

Cloud

- Kris A. Jamsa , *Cloud Computing* , ISBN : 978-1-449-64739-1
- Christopher M. Moyer , *Building Applications in the Cloud* , ISBN : 978-0-132-10794-5
- George Reese , *Cloud Application Architectures* , ISBN : 978-0-596-15636-7

LAMP

- Wale Soyinka , *Linux Administration A Beginners Guide* , ISBN : 978-0-071-76758-3
- Lynn Beighley and Michael Morrison , *Head First PHP & MySQL* , ISBN : 978-0-596-00630-3
- Hugh E. Williams and David Lane , *Web Database Applications with PHP and MySQL* , ISBN : 978-0-596-00041-7
- David Cochran , *Twitter Bootstrap Web Development How-To* , ISBN : 978-1-849-51882-6

- Jake Spurlock , *Bootstrap* , ASIN : B00CPBCHE4
- Twitter , Twitter Bootstrap , <http://twitter.github.io/bootstrap/>

Facebook App

- OAuth, *OAuth 2.0* , <http://oauth.net/2/>
- Ryan Boyd , *Getting Started with OAuth 2.0* , ISBN : 978-1-449-31160-5
- Facebook , *Facebook SDK for PHP* ,
<https://developers.facebook.com/docs/reference/php/>
- Facebook , Developer Documentation ,
<https://www.developers.facebook.com/docs>

Παράρτημα Α' – Κώδικας Arduino

```
#include <SD.h> // προσθήκη των
#include <SPI.h> // αναγκαίων βιβλιοθηκών
#include <Ethernet.h>
#include <HTTPClient.h>
#include <LiquidCrystal.h>

LiquidCrystal lcd(31, 30, 29, 28, 27, 26); // δήλωση των pin της LCD
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; // δήλωση mac του Ethernet
IPAddress ip(192,168,178, 100); // Module όπως και την IP του
EthernetServer server(80); //
EthernetClient client;
bool connected = false; // Δήλωση βοηθητικών
File myFile; // μεταβλητών
int arx=0;
int tel=0;
char inputline;
String minima="";
String SMSstruct[4];
String noumero="";
String Onomata[20];
String Noumera[20];
int plithosnoumeron=0;
int thesi=1;
int arxi=1;
long xronosa;
```

```

void setup(){ // Αρχή της ρουτίνας setup

digitalWrite(36, HIGH); // Αναγκαίο βήμα για το
pinMode(36, OUTPUT); // αυτόματο reset

lcd.begin(16, 2); // ανάβουμε την οθόνη

Serial.begin(9600); // ξεκινάμε την σειριακή
Serial2.begin(9600); // επικοινωνία με PC και GSM

pinMode(53, OUTPUT); // Ξεκινάμε σωστά το Ethernet
pinMode(10,OUTPUT); // Module ώστε να λειτουργεί
digitalWrite(10,HIGH); // και το Ethernet και η SD

lcd.setCursor(0,0); // Πάμε στον πρώτο χαρακτήρα
lcd.print("SD"); // της οθόνης και γράφουμε

if (!SD.begin(4)) { // Ελέγχουμε και ενημερώνουμε
    lcd.setCursor(8,0); // αν ξεκίνησε σωστά η SD
    lcd.print("FAIL!");
}
else {
    lcd.setCursor(8,0);
    lcd.print("OK!");
}

lcd.setCursor(0,1); // Και προχωράμε στο Ethernet
lcd.print("ETH");

if(!Ethernet.begin(mac)){ // Για το οποίο κάνουμε τα

```

```

        lcd.setCursor(8,1);                // αντίστοιχα βήματα
        lcd.print("FAIL!");
    }
    else {
        lcd.setCursor(8,1);
        lcd.print("OK!");
    }

    digitalWrite(10,HIGH);

    minima="";
    delay(200);
    myFile=SD.open("noumera.txt");        // Ανοίγουμε το αρχείο
    if(myFile){                            // με τα αποθηκευμένα
        int i=0;                            // τηλέφωνα και τα
        while (myFile.available()) {        // κρατάμε σε πίνακα
            inputline=myFile.read();
            if (inputline!='\r'){
                if (inputline!='\n'){
                    noumero+=inputline;
                }
            }
            else{
                int voithitiki=noumero.indexOf(',');
                Noumera[i]=noumero.substring(voithitiki+1);
                Onomata[i]=noumero.substring(0,voithitiki);
                plithosnoumeron=plithosnoumeron+1;
                noumero="";
                i=i+1;
            }
        }
    }
}

```

```

        myFile.close();
    }
    else {
        Serial.println("error opening test.txt");
    }

    delay(500);
    minima="";

    Serial.println("\r\nCommunication started\r\n");           // Ενημερώνουμε
    lcd.clear();                                               // η επικοινωνία
    lcd.print("Communication");                                 // έχει ξεκινήσει
    lcd.setCursor(0,1);
    lcd.print("started");

    }

    void loop(){                                               // Αρχή της ρουτίνας loop

    if (millis())>604800000){                                   // Έλεγχος συνεχόμενης
        reset();                                               // λειτουργίας
    }

    if (arxi==1){                                              // Κατάλληλη εκκίνηση
        Serial2.print("AT+CMGF=1\r");                          // του GSM Modem
        delay(400);
        arxi=2;
    }

    if(Serial2.available())>0){
        while (Serial2.available())>0){

```

```

        inputline=Serial2.read();
        minima += inputline;
    }
}

if (arxi==2){
    minima="";
    arxi=3;
}

while(thesi<=10){
    recievesms(thesi);
    xronosa=millis();
    while (xronosa+1000>millis()){
        diavasma();
    }
    ektiposi();
    thesi=thesi+1;
    delay(1000);
    delay(100);
}

if (thesi==11){
    thesi=1;
}

}

void recievesms(int k){
    Serial2.print("AT+CMGR=");
    Serial2.print(k);
    Serial2.print("\r");
}

```

```

}

void deleteSMS(int k){                                     // ρουτίνα για την διαγραφή SMS
    Serial2.print("AT+CMGD=");
    Serial2.print(k);
    Serial2.print("\r");
}

void diavasma(){                                         // ρουτίνα για το διάβασμα από GSM
    if(Serial2.available(>0){
        while (Serial2.available(>0){                 // Διάβασμα ροής από GSM
            inputline=Serial2.read();
            minima += inputline;
        }
    }
}

void ektiposi(){                                         // ρουτίνα για την αποκωδικοποίηση
if (minima!=""){                                         // και τον έλεγχο του SMS
    if (minima.length(>60){
        arx=minima.indexOf("")+1;                     // διαδοχικός διαμερισμός του
        tel=minima.indexOf("",arx+1);                 // μηνύματος στα κατάλληλα
        Serial.print ("to minima einai diavasmeno : "); // κομμάτια
        SMSstruct[0]= (minima.substring(arx,tel));    // και αποθήκευση σε
        Serial.println (SMSstruct[0]);                // μεταβλητή
        arx=minima.indexOf("",tel+1)+1;
        tel=minima.indexOf("",arx);
        Serial.print ("to minima to estile to noumero : ");
        SMSstruct[1]= minima.substring(arx,tel);
        Serial.println (SMSstruct[1]);
        arx=minima.indexOf("",tel+1)+1;
    }
}
}

```

```

tel=minima.indexOf("",arx);
Serial.print ("to minima irthe stis : ");
SMSstruct[2]= minima.substring(arx,tel);
Serial.println (SMSstruct[2]);
Serial.print ("to minima akolouthi: ");
arx=tel+3;
tel=minima.indexOf('\r',arx);
SMSstruct[3]= minima.substring(arx,tel);
Serial.println (SMSstruct[3]);

for (int i=0; i<plithosnoumeron;i++){           // έλεγχος αν έχει
    if (SMSstruct[1].substring(3).equals(Noumera[i])){ // σταλεί
        SMSstruct[1]=Onomata[i];           // από
                                                // αποδεκτό νούμερο
        rhp();                             // αποστολή στο Internet
        delay(100);
        karta();                           // αποστολή στην SD
        delay(100);
        deleteSMS(thesi);                 // διαγραφή από SIM
        diavasma();
        delay(100);
        break;
    }

}

deleteSMS(thesi);
diavasma();

}
else{
    Serial.println ("adio");
}

```

```

    }
}
else{
    Serial.println ("adio");
}
minima="";
}

void karta(){ // ρουτίνα για εγγραφή στην SD

int voithitiki=SMSstruct[2].indexOf(','); // δημιουργία κατάλληλου
String filename=SMSstruct[2].substring(0,voithitiki);
filename.replace("/", "-"); // ονόματος για το αρχείο
filename=filename+".txt";
char filename2[filename.length()+1];
filename.toCharArray(filename2,filename.length()+1);
myFile = SD.open(filename2, FILE_WRITE); // άνοιγμα του αρχείου για
// εγγραφή

if (myFile) { // Εγγραφή των χρήσιμων
    Serial.print("Writing to test.txt..."); // πληροφοριών στο αρχείο
    for (int i=0;i<4;i++){ // .txt
        myFile.print("stoixeio ");
        myFile.println(i+1);
        myFile.println(SMSstruct[i]);
        SMSstruct[i]="";
    }
    myFile.close();
    Serial.println("done.");
} else {

```



```

Serial.print("error opening ");
Serial.println(filename2);
}
}

void php(){ // ρουτίνα για επικοινωνία με PHP

if(!connected) {
    if (client.connect("smstoauto.info",80)){
        connected = true;

        client.print("GET /add.php?numero="); // αποστολή στον
        client.print(SMSstruct[1]); // server των
        client.print("&&ora="); // χρήσιμων
        client.print(SMSstruct[2]); // πληροφοριών
        client.print("&&kimeno=");
        String kimenourl=SMSstruct[3];

        kimenourl.replace(" ", "%20"); // αποφυγή επικίνδυνων
        kimenourl.replace("$", "%24"); // χαρακτήρων για το URL
        kimenourl.replace("&", "%26"); // από το κείμενο του
        kimenourl.replace("+", "%2B"); // εισερχόμενου SMS
        kimenourl.replace(",", "%2C");
        kimenourl.replace("/", "%2F");
        kimenourl.replace(":", "%3A");
        kimenourl.replace(";", "%3B");
        kimenourl.replace("=", "%3D");
        kimenourl.replace("?", "%3F");
        kimenourl.replace("@", "%40");

```

```

    kimenourl.replace("<","%3C");
    kimenourl.replace(">","%3E");
    kimenourl.replace("#","%23");
    kimenourl.replace("%","%25");
    kimenourl.replace("{","%7B");
    kimenourl.replace("}","%7D");
    kimenourl.replace("|","%7C");
    kimenourl.replace("\\","%5C");
    kimenourl.replace("^","%5E");
    kimenourl.replace("~","%7E");
    kimenourl.replace("[","%5B");
    kimenourl.replace("]","%5D");
    kimenourl.replace("\"","%60");
    kimenourl.replace("\'", "%22");

    client.print(kimenourl);
    client.println(" HTTP/1.1");
    client.println("Host: smstoauto.info");
    client.println("User-Agent: arduino-ethernet");
    client.println("Accept: text/html");
    client.println("Connection: close");
    client.println();
    delay(20000);

    client.stop();
    connected = false;
}
else{
    Serial.println("Cannot connect to Server");
}
}

```

```
}
```

```
void reset() { // ρουτίνα για το reset της εφαρμογής
```

```
digitalWrite(36, LOW);
```

```
}
```

Παράρτημα Β' – Κώδικας PHP για σύνδεση με το Facebook

```
<?php

//-- Στοιχεία Εφαρμογής --//

$app_id = "1111111111111111";

$app_secret = "11111111111111111111111111111111";

$my_url = "http://smstoauto.info/tokens.php";

//-- Αρχή ενός session --//

session_start();

//-- Ελέγχουμε αν έχουμε ήδη πάρει το στοιχείο code --//

$code = $_REQUEST["code"];

//-- Αν όχι, στέλνουμε τον χρήστη στην εφαρμογή για επιβεβαίωση --//

if(empty($code))

{

    $_SESSION['state'] = md5(uniqid(rand(), TRUE));

    //-- Χτίσιμο του URL που δίνει την άδεια --//

    $dialog_url = "https://www.facebook.com/dialog/oauth?client_id="

        . $app_id . "&redirect_uri=" . urlencode($my_url) . "&state="

        . $_SESSION['state'] . "&scope=publish_actions";

    echo("<script> top.location.href='" . $dialog_url . "'</script>");

}
```

```

if($_SESSION['state'] && ($_SESSION['state'] === $_REQUEST['state']))
{
    //-- Χτίσιμο URL για επιμήκυνση του Token μας σε 60 ημέρες --//
    $token_url = "https://graph.facebook.com/oauth/access_token?"
    . "client_id=" . $app_id . "&redirect_uri=" . urlencode($my_url)
    . "&client_secret=" . $app_secret . "&code=" . $code;

    //-- Απόκτηση του Token και του χρόνου ζωής του --//
    $response = file_get_contents($token_url);

    $params = null;

    parse_str($response, $params);

    $_SESSION['access_token'] = $params['access_token'];

    $_SESSION['expires'] = $params['expires'];

    //-- Εκτύπωση του Token και της διάρκειας ζωής του --//
    echo 'Token: ' . $_SESSION['access_token'] . '<br>';

    $expireDate = time() + $_SESSION['expires'];

    echo 'Expiration: ' . date('Y-m-d', $expireDate) . '<br>';

    //-- Αποθήκευση των στοιχείων στην βάση δεδομένων μας --//

    $mysql_host = "";

    $mysql_user = "";

    $mysql_password = "";

    $mysql_db = "";

    $mysql_table = "";

```

```

if (!$mysql_link = mysql_connect($mysql_host, $mysql_user, $mysql_password))
{
    echo 'Could not connect to mysql';
    exit;
}

if (!$mysql_select_db($mysql_db, $mysql_link))
{
    echo 'Could not select database';

    exit;
}

$query = "INSERT INTO $mysql_table (token, expire) VALUES
(\"\".$_SESSION['access_token'].\"\", \"\"$expireDate\"\")";

if (!$mysql_query($query, $mysql_link))
{
    echo 'Error: ' . mysql_error();

    exit;
}

echo "Stored token and expiration date in database";
}

else
{
    echo("The state does not match. You may be a victim of CSRF.");
}

?>

```