

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Υλοποίηση Εφαρμογής για Εποπτεία και
Έλεγχο Συστημάτων Ενεργειακού Ελέγχου
Κτιρίων (BEMS)**

Κωνσταντίνος Τσίχλης

Εξεταστική Επιτροπή:

Κωνσταντίνος Καλαϊτζάκης, Καθηγητής (Επιβλέπων)

Αντώνιος Δεληγιαννάκης, Αναπληρωτής Καθηγητής

Ευύχιος Κουτρούλης, Επίκουρος Καθηγητής

Χανιά, Ιούνιος 2014

*Τσίχλης Κωνσταντίνος
Πολυτεχνείο Κρήτης, ΗΜΜΥ
Χανιά, Ιούνιος 2014*

ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε στο τμήμα Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών του Πολυτεχνείου Κρήτης υπό την επίβλεψη του καθηγητή κυρίου Κωνσταντίνου Καλαϊτζάκη, κατά το ακαδημαϊκό έτος 2013-2014.

Αντικείμενο της παρούσας διπλωματικής εργασίας είναι η υλοποίηση μίας εφαρμογής για πλατφόρμα κινητού τηλεφώνου, μέσω της οποίας πραγματοποιείται η εποπτεία και ο έλεγχος Συστημάτων Ενεργειακού Ελέγχου Κτιρίων (BEMS) και συγκεκριμένα για το κτίριο Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης.

Με την ολοκλήρωση αυτής της εργασίας, θα ήθελα να εκφράσω τις ειλικρινείς μου ευχαριστίες προς τον επιβλέποντα καθηγητή κύριο Κωνσταντίνο Καλαϊτζάκη για την καθοριστική συμβολή, την ουσιαστική υποστήριξη, την καθοδήγηση και την εμπιστοσύνη που μου έδειξε, αλλά και τον πολύτιμο χρόνο που αφιέρωσε καθ' όλη τη διάρκεια εκπόνησης της εργασίας.

Θα ήθελα επίσης να εκφράσω τις θερμές μου ευχαριστίες στον κύριο Αντώνιο Δελιγιαννάκη και στον κύριο Ευτύχιο Κουτρούλη, για την αξιολόγηση της διπλωματικής μου εργασίας ως μέλη της εξεταστικής επιτροπής.

Ιδιαίτερες ευχαριστίες θα ήθελα να εκφράσω προς τον κύριο Κυριάκο Κατσιγαράκη για την αμέριστη βοήθεια και προσωπικό χρόνο που αφιέρωσε, παραχωρώντας απαραίτητα στοιχεία και δεδομένα ώστε να υλοποιηθεί επιτυχώς η εργασία.

Επιπλέον, χρωστάω ένα μεγάλο ευχαριστώ στους συμφοιτητές και φίλους μου, για τα όμορφα, δημιουργικά, ξέγνοιαστα και παραγωγικά χρόνια που περάσαμε μαζί.

Τέλος, δε θα μπορούσα να παραλείψω να πω ένα μεγάλο ευχαριστώ προς την οικογένεια μου. Η οικονομική και ψυχολογική υποστήριξη που έδειξαν καθ' όλη τη διάρκεια των τελευταίων ετών ήταν τεράστια και χωρίς αυτήν δεν θα ήταν εφικτό να ολοκληρώσω τις σπουδές μου. Για τον λόγο αυτό, τους αφιερώνω εξ ολοκλήρου την εργασία αυτή.

Τσίχλης Κωνσταντίνος,

Χανιά 2014

ΠΕΡΙΛΗΨΗ

Τα τελευταία χρόνια ο τομέας της κινητής τηλεφωνίας έχει παρουσιάσει τρομερή εξέλιξη και αποτελεί πλέον σημαντικό κομμάτι της καθημερινής ζωής του ανθρώπου. Τα κινητά τηλέφωνα έχουν αλλάξει σημαντικά τον τρόπο ζωής και εργασίας του. Οι σημαντικές τεχνολογικές εξελίξεις οδήγησαν στην υιοθέτηση νέων χαρακτηριστικών στα κινητά τηλέφωνα, μετατρέποντάς τα σε «έξυπνα» κινητά, γνωστά με τον όρο «smartphones». Τα smartphones αποτελούν μικρούς υπολογιστές, και έχουν υποκαταστήσει τις ψηφιακές μηχανές, τα συστήματα GPS, τα συστήματα επικοινωνίας και κοινωνικής δικτύωσης ακόμα και τις παιχνιδομηχανές. Έτσι τα τελευταία τέσσερα χρόνια έχει γίνει μία μεγάλη επανάσταση στην ανάπτυξη εφαρμογών για κινητά τηλέφωνα. Παλιότερα ήταν προνόμιο μόνο των κατασκευαστών κινητών τηλεφώνων, αλλά πλέον είναι στην διάθεση κάθε προγραμματιστή για περαιτέρω βελτίωση.

Παράλληλα το μεγάλο πρόβλημα της ενέργειας και το ξέσπασμα της οικονομικής κρίσης τα τελευταία χρόνια σε παγκόσμια κλίμακα, οδήγησαν την Ευρώπη να πάρει σημαντικές αποφάσεις. Θέσπισε ερευνητικά προγράμματα όπως το PEBBLE, στην προσπάθειά της να μετατρέψει τα κτίρια, τα οποία είναι υπεύθυνα για το 40% της καταναλισκόμενης ενέργειας, σε αυτόνομες μονάδες εξοικονόμησης ενέργειας.

Επίσης τα τελευταία δύο χρόνια γίνεται συνεχής προσπάθεια για εξέλιξη της τεχνολογίας στον τομέα αυτοματοποίησης των σπιτιών από τις μεγαλύτερες εταιρείες κινητής τηλεφωνίας. Η διαχείριση της κατοικίας μέσω του διαδικτύου αποτελεί μέσο εξοικονόμησης ενέργειας αλλά σίγουρα και εξοικονόμησης χρημάτων για τον καταναλωτή.

Επομένως, η παρούσα εργασία πραγματεύεται την δημιουργία μίας εφαρμογής σε πλατφόρμα κινητού τηλεφώνου iPhone (iOS), με το όνομα **S.M.C.**, μέσω της οποίας πραγματοποιείται η εποπτεία και ο έλεγχος ενός Συστήματος Ενεργειακού Ελέγχου Κτιρίων, Building Energy Management System (BEMS), με στόχο την εξοικονόμηση ενέργειας. Συγκεκριμένα, αναπτύχθηκε για το κτίριο Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης, στα πλαίσια του ευρωπαϊκού προγράμματος PEBBLE. Η ανάπτυξη της εφαρμογής περιλαμβάνει και την υλοποίηση ενός server, ο οποίος επιτρέπει την επικοινωνία της εφαρμογής με τις βάσεις δεδομένων. Σε αυτές διατηρούνται όλα τα δεδομένα σχετικά με τους αισθητήρες και τους ενεργοποιητές που έχουν εγκατασταθεί στο συγκεκριμένο κτίριο. Πρόκειται για έναν web server που ακολουθεί το πρωτόκολλο REST και είναι υλοποιημένος σε γλώσσα προγραμματισμού PHP. Αναλύονται οι τεχνολογίες που χρησιμοποιήθηκαν καθώς και η αρχιτεκτονική της εφαρμογής και του server. Επίσης, περιγράφονται οι λειτουργίες και οι δυνατότητες που παρέχει σε κάθε χρήστη. Τέλος, ο σχεδιασμός της εφαρμογής έγινε με κριτήριο την ευελιξία, ώστε να έχει την δυνατότητα προσαρμογής και σε άλλα συστήματα ή κτίρια μελλοντικά, στοχεύοντας ουσιαστικά στην αξιολόγηση και κατόπιν βελτιστοποίηση των συστημάτων διαχείρισης ενέργειας κτιρίων (BEMS).

ABSTRACT

The last few years, cell phones have shown tremendous progress and have become an important part of everyday human life. Cell phones have coherently changed the way of human life and work. The significant technological advances led to the adoption of new features to cell phones, converting them into clever phones, known with the term “smartphones”. Smartphones are mini computers, which have substituted cameras, GPS systems, communication systems, social network systems, even game consoles. So the last four years a major breakthrough happened, the development of mobile applications which was once a prerogative of mobile phone manufacturers has been made available to the world of developers for further improvements.

Alongside the big energy problem and the outbreak of the financial crisis worldwide in recent years, led Europe to take important decisions. Europe established many programs, such as PEBBLE, in trying to convert the buildings, which are responsible for 40% of the energy consumed, into autonomous units of energy saving.

In addition, the last two years has been made a constant effort, by major mobile companies, to evolve the technology in the field of automation of the houses. The management of home via Internet is a mean of saving energy but certainly saving money for the consumer too.

Thus, this thesis develops a mobile application in iPhone (iOS), named **S.M.C.**, through which the monitoring and control of a Building Energy Management System (BEMS) is performed, aiming to save energy. Specifically **S.M.C.** was developed for the Technical Services (maintenance support) building of TUC, as part of the European energy program PEBBLE. The development of the application includes the implementation of one server, which allows communication between the application and the databases. These databases maintain all the data of sensors and actuators that are established in the building. This server follows the REST protocol and is written in PHP. It's about a REST PHP web server. The architecture and technologies of the application and server are being analyzed. In addition, the functionalities and capabilities that the application provides, are described in detail. Finally, the application's design was made in the light of flexibility, in order to be adapted to other systems or buildings in the future, aiming to the essential evaluation and then optimization of the Building Energy Management Systems, BEMS.

Πίνακας περιεχομένων

1. ΕΙΣΑΓΩΓΗ	9
1.1 Γενικά	9
1.2 Στόχος Διπλωματικής Εργασίας	11
1.3 Συνεισφορά Διπλωματικής Εργασίας	12
1.4 Σύντομη περιγραφή Λειτουργιών και δυνατοτήτων	13
1.5 Διάρθρωση επόμενων κεφαλαίων	13
2. ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΕΡΕΥΝΗΤΙΚΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ PEBBLE	15
2.1 Γενικά για το P.E.B.B.L.E.	15
2.1.1 Δομή του προγράμματος P.E.B.B.L.E.	16
2.1.2 Το P.E.B.B.L.E. στο Πολυτεχνείο Κρήτης	19
2.2 Διαδικτυακή (Web) Εφαρμογή P.E.B.B.L.E.	23
2.3 Εφαρμογή σε πλατφόρμα κινητού τηλεφώνου στα πλαίσια του P.E.B.B.L.E.	24
2.3.1 Πλεονεκτήματα – Μειονεκτήματα native mobile application	24
2.3.2 Εφαρμογή Sensor Monitor Controller (S.M.C.)	26
2.3.3 Γιατί προτιμήθηκε το iPhone – iOS	26
3. CLIENT SIDE - Αρχιτεκτονική Συστήματος και Τεχνολογίες που χρησιμοποιήθηκαν στην εφαρμογή S.M.C	28
3.1 Λειτουργικό Σύστημα iOS	28
3.1.1 Γενικά	28
3.1.2 Αρχιτεκτονική και Πακέτο Ανάπτυξης Λογισμικού	29
3.1.3 Περιορισμοί ανάπτυξης λογισμικού iOS	31
3.2 Objective-C	32
3.2.1 Εισαγωγή στην Objective-C	32
3.2.2 Χαρακτηριστικά Objective-C	33
3.3 Μοντέλο σχεδιασμού MVC (Model View Controller)	34
3.4 Xcode	36
3.5 Εφαρμογή Sensor Monitor Controller	38
3.5.1 Προϋποθέσεις και Υπόβαθρο	38
3.5.2 Εγκατάσταση εφαρμογής σε συσκευή iPhone	38
3.5.3 Βιβλιοθήκες που χρησιμοποιήθηκαν	40
4. SERVER SIDE - Αρχιτεκτονική Συστήματος και Τεχνολογίες που χρησιμοποιήθηκαν στον εξυπηρετητή	42

4.1	Υπηρεσίες Δικτύου (Web Services).....	42
4.1.1	Τι είναι οι Υπηρεσίες Δικτύου (Web Services).....	42
4.1.2	Σχεδιαστικές αρχές υπηρεσιών δικτύου	43
4.1.3	Πλεονεκτήματα	44
4.2	Επεκτάσιμη Γλώσσα Σήμανσης (eXtensible Markup Language, XML)	45
4.3	JavaScript Object Notation (JSON).....	46
4.4	Σύγκριση μεταξύ XML και JSON.....	47
4.5	Representational State Transfer (REST)	48
4.5.1	Γενικά	48
4.5.2	Περιορισμοί.....	48
4.6	Τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του Restful Web Service	50
4.6.1	Προεπεξεργαστής Υπερκειμένου PHP (PHP Hypertext Preprocessor)	50
4.6.1.1	Γενικά – Ιστορική Αναδρομή.....	50
4.6.1.2	Πλεονεκτήματα	51
4.6.2	MySQL.....	53
4.6.2.1	Γενικά.....	53
4.6.2.2	Πλεονεκτήματα	54
4.7	Υλοποίηση του εξυπηρετητή στα πλαίσια της παρούσας εφαρμογής.....	55
4.7.1	Εισαγωγή.....	55
4.7.2	RESTful Web Service – REST PHP Server.....	56
4.7.3	Βάση Δεδομένων.....	58
4.7.4	Επικοινωνία iPhone - Web Server σε επίπεδο προγραμματισμού.....	59
4.7.5	Χρήση Ιδιωτικών Βιβλιοθηκών.....	59
4.8	Προγραμματιστικά Εργαλεία για την υλοποίηση του εξυπηρετητή της εφαρμογής	60
4.8.1	Apache HTTP Server	60
4.8.2	PHP.....	61
4.8.3	MySQL Community Server	61
4.8.4	Navicat Premium	62
5.	ΑΡΧΕΣ ΣΧΕΔΙΑΣΗΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	63
5.1	Βασική Δομή των Οθονών του User Interface.....	63
5.2	Ενημέρωση Χρήστη	68
5.3	Καθοδήγηση Χρήστη και Ευκολία Πλοήγησης.....	69
5.4	Ομαδοποίηση Λειτουργιών	70
5.5	Επιλογή Χρωμάτων και Συνέπεια	70

5.6	Χρήση Ενεργού Περιοχής και Χειρισμός Λαθών	73
5.7	Χρήση και διάθεση εφαρμογής σε δύο γλώσσες.....	73
5.8	Συμβατότητα με άλλες Συσκευές της Apple	74
6.	ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	75
6.1	Εισαγωγή - Λειτουργικότητα Εφαρμογής.....	75
6.2	Λειτουργία Εφαρμογής σε Πραγματικό Χρόνο	75
6.3	Ανάλυση Απαιτήσεων Λειτουργιών Εφαρμογής	76
6.3.1	Εγγραφή στο Σύστημα	76
6.3.2	Είσοδος – Έξοδος στο Σύστημα.....	77
6.3.3	Ανάκτηση Κωδικού	78
6.3.4	Διαχείριση Προσωπικού Λογαριασμού.....	78
6.3.5	Εποπτεία – Έλεγχος δωματίων-μονάδων κλιματιστικού.....	78
6.3.6	Προβολή Διαγράμματος.....	79
6.3.7	Περισσότερες Πληροφορίες - Πλοήγηση.....	79
6.4	Γραφική Απεικόνιση Λειτουργιών Εφαρμογής.....	80
6.4.1	Οθόνη Καλωσορίσματος.....	80
6.4.2	Οθόνη Εισόδου Χρήστη στην Εφαρμογή.....	81
6.4.3	Οθόνη Ανάκτησης Κωδικού.....	83
6.4.4	Οθόνη Εποπτείας ζωνών, δωματίων και μονάδων A/C	86
6.4.5	Οθόνη Σημάτων Ελέγχου	89
6.4.6	Οθόνη Προβολής Διαγράμματος.....	91
6.4.7	Οθόνη Διαχείρισης Προσωπικού Λογαριασμού	98
6.4.8	Οθόνη Πληροφοριών και Πλοήγησης.....	103
6.4.9	Περιπτώσεις Προβλήματος Δικτύου ή Server.....	107
6.4.10	Λειτουργία σύνδεσης με Wi-Fi και 3G	108
6.4.11	Λειτουργία στην Ελληνική Γλώσσα.....	109
6.4.12	Λειτουργία Αδράνειας Χρήστη	110
7.	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	111
8.	ΠΡΟΤΑΣΕΙΣ ΒΕΛΤΙΩΣΗΣ	113
9.	ΒΙΒΛΙΟΓΡΑΦΙΑ	116

1. ΕΙΣΑΓΩΓΗ

1.1 Γενικά

Η ενέργεια αποτέλεσε από την πρώτη στιγμή το κύριο μέσο στις δραστηριότητες του ανθρώπου. Είναι η πηγή όχι μόνο για την διευκόλυνση της ζωής του ανθρώπου, αλλά και για την ικανοποίηση των βασικών αναγκών του. Τα τελευταία χρόνια ιδιαίτερο ενδιαφέρον έχει παρουσιαστεί στον τομέα της ενεργειακής απόδοσης και εξοικονόμησης. Η ταχεία αύξηση της ενεργειακής κατανάλωσης ανά τον κόσμο, έχει ήδη εγείρει ανησυχίες για την εξάντληση των συμβατικών ενεργειακών πόρων και για τις περιβαλλοντικές επιπτώσεις μεγάλης κλίμακας. Ενδεικτικά, η υπερθέρμανση του πλανήτη οδηγεί σε αύξηση της θερμοκρασίας του περιβάλλοντος, που στα μεγάλα αστικά κέντρα μεταφράζεται σε δραματική αύξηση της απαιτούμενης ενέργειας για το δροσίσιμο των κτιρίων κατά την καλοκαιρινή περίοδο. Ο κτιριακός τομέας συνεισφέρει με ποσοστό 40% στη συνολική καταναλισκόμενη ενέργεια παγκοσμίως. Συγκεκριμένα, στις χώρες της Ευρωπαϊκής Ένωσης εκτιμάται ότι το ποσοστό αυτό κυμαίνεται από 20% έως 45% της συνολικής ενέργειας κατανάλωσης¹.

Στην Ελλάδα, η κατανάλωση ενέργειας στα κτίρια αγγίζει πολύ υψηλά επίπεδα, γεγονός που οφείλεται στην έλλειψη επαρκούς μόνωσης και ιδιαίτερα σε όσα κατασκευάστηκαν πριν το 1980². Τα ελληνικά κτίρια χαρακτηρίζονται ως ενεργοβόρα, καθώς παρουσιάζουν θερμικές απώλειες από τις πόρτες και τα παράθυρα. Αυτό έχει ως αποτέλεσμα, εκτός της σημαντικής οικονομικής επιβάρυνσης λόγω του υψηλού κόστους της ενέργειας, τη μεγάλη επιβάρυνση της ατμόσφαιρας με ρύπους, κυρίως διοξείδιο του άνθρακα (CO₂), που ευθύνεται για το φαινόμενο του θερμοκηπίου. Ο τομέας των κτιρίων, στην Ελλάδα, καταναλώνει περίπου 35% ενέργεια και συμβάλλει στην παραγωγή του 43% του διοξειδίου του άνθρακα που εκλύεται στην ατμόσφαιρα.

Σύμφωνα με τα τις παραπάνω συνθήκες γίνεται κατανοητό ότι είναι αναγκαία η ύπαρξη αλλαγών και ανάπτυξη πρωτοβουλιών. Πρόσφατα, η Ευρώπη πλήγηκε από την παγκόσμια οικονομική κρίση, της οποίας οι συνέπειες είναι πολυάριθμες και απρόβλεπτες. Στην προσπάθεια της Ευρώπης να ανταπεξέλθει σε αυτή την δύσκολη κατάσταση και να ενισχύσει τον ρόλο της ως πρωτοπόρο των διεθνών θεμάτων, απαιτούνται δυναμικές αλλά και λογικές αποφάσεις. Στα πλαίσια αυτής της

¹ Η Πορτογαλία έχει το χαμηλότερο ποσοστό με 20% και η Ιρλανδία το υψηλότερο, με 45%. [1]

² Το 1980 αποτελεί την ημερομηνία εφαρμογής κανονισμού θερμομόνωσης.

προσπάθειας η Ευρωπαϊκή Ένωση έχει αποφασίσει για μείωση κατά 20% στη χρήση πρωτογενούς ενέργειας σε σύγκριση με τα προβλεπόμενα επίπεδα του 1990, κάλυψη ποσοστού 20% των ενεργειακών αναγκών από ανανεώσιμες πηγές ενέργειας, και μείωση κατά τουλάχιστον 20% των εκπομπών αερίων του θερμοκηπίου μέχρι το 2020³. Η επίτευξη αυτών των στόχων, όπου αναφέρονται ως στόχοι 20-20-20, απαιτεί την ανάπτυξη νέων τεχνολογιών εξοικονόμησης και σωστής διαχείρισης ενέργειας σε όλους τους παραγωγικούς τομείς και βελτίωση της ενεργειακής απόδοσης.

Η πρόοδος στους τομείς του βιοκλιματικού σχεδιασμού, η βελτίωση της λειτουργίας των συστημάτων ελέγχου ενεργειακών στοιχείων του κτιρίου (όπως για παράδειγμα τα συστήματα εξαερισμού, κλιματισμού και θέρμανσης, τα ηλιακά συστήματα, συστήματα σκιάστρον και φυσικού εξαερισμού), αποτελούν νέες προοπτικές στην υλοποίηση βιώσιμων συστημάτων εξοικονόμησης ενέργειας και ενεργειακής απόδοσης, τόσο στην λειτουργία νέων όσο και υφιστάμενων κτιρίων. Η ανακαίνιση και η μόνωση κτιρίων αποτελούν πολύ δαπανηρές και χρονοβόρες λύσεις. Αυτό έχει αποτέλεσμα την ανάπτυξη δράσεων, οι οποίες αποσκοπούν αφενός στην ανάπτυξη νέων αποδοτικότερων τεχνολογιών, όπως η διάδοση της χρήσης Ανανεώσιμων Πηγών Ενέργειας (Α.Π.Ε.) για κάλυψη μέρους των ενεργειακών αναγκών, και αφετέρου στην αποτελεσματικότερη χρήση της διαθέσιμης ενέργειας. Τέτοιου είδους τεχνολογίες, που σχετίζονται με την ορθότερη χρήση των συστημάτων HVAC (Heating, Ventilation & Air-conditioning), ονομάζονται Συστήματα Διαχείρισης Ενέργειας Κτιρίων (Building Energy Management Systems, BEMS). Η παραπάνω απαίτηση είναι δυνατό να επιτευχθεί, με χρήση πολύ καλού παθητικού ενεργειακού σχεδιασμού, όπως χρήση μόνωσης συνδυασμένη με ΑΠΕ, ώστε να καλυφθούν οι υπόλοιπες ενεργειακές ανάγκες για θέρμανση, ψύξη και ζεστά νερά χρήσης. Συνεπώς τα Κτίρια με Μηδενικό Ενεργειακό Ισοζύγιο (ΚΜΕΙ) ή και με Θετικό Ενεργειακό Ισοζύγιο (ΚΘΕΙ) αποτελούν πλέον βασική προτεραιότητα. Μακροπρόθεσμος στόχος είναι η ανάπτυξη τεχνογνωσίας, προϊόντων και εργαλείων για την κατασκευή των Κτιρίων Θετικού Ενεργειακού Ισοζυγίου (ΚΘΕΙ), τα οποία, σε ετήσια βάση, παράγουν περισσότερη ενέργεια από όση απαιτείται για τη λειτουργία τους.

³ Η Ε.Ε. πήρε την απόφαση τον Απρίλιο του 2009

1.2 Στόχος Διπλωματικής Εργασίας

Στα πλαίσια της προσπάθειας υλοποίησης των στόχων που αναφέρονται παραπάνω εντάσσεται το ευρωπαϊκό πρόγραμμα P.E.B.B.L.E.⁴, που αποτελεί μελέτη εφαρμογής (case study) για την παρούσα εργασία [2]. Ουσιαστικά η κατασκευή έξυπνων και ενεργειακά αποδοτικών κτιρίων εξαρτάται από την επιτυχή υλοποίηση και των δύο ξεχωριστών φάσεων του κύκλου ζωής των κτιρίων. Πρώτον, κατά την διάρκεια του σχεδιασμού και κατασκευής των κτιρίων, επιλέγοντας ένα αποδοτικό ενεργειακό σχέδιο και δεύτερον, κατά την διάρκεια λειτουργίας τους, επιλέγοντας συστήματα και συσκευές που καταναλώνουν όσο δυνατόν λιγότερη ενέργεια, εξασφαλίζοντας παράλληλα θερμική πληρότητα-άνεση. Αυτό κατά ένα μέρος επιτυγχάνεται με την εφαρμογή αισθητήρων σε πολλούς χώρους και σημεία των κτιρίων, αλλά και με την τοποθέτηση ενεργοποιητών (actuators) σε σημεία όπως παράθυρα, πόρτες, air-condition. Έτσι είναι εφικτή η συνεχή παρακολούθηση, συγκέντρωση και ανάλυση των δεδομένων, όπως θερμοκρασία, κατανάλωση ενέργειας, υγρασία και παράλληλα δίνεται η δυνατότητα ελέγχου, βάση της ανάλυσης των δεδομένων που έχει προκύψει. Επομένως, γίνεται εύκολα αντιληπτό ότι η εποπτεία των κτιρίων αλλά κυρίως ο έλεγχος αυτών θα ήταν εξαιρετικά χρήσιμο να γίνονται μέσω μίας πλατφόρμας που λειτουργεί σε πραγματικό χρόνο με τη βοήθεια των «ενοίκων». Το συγκεκριμένο πρόγραμμα προβλέπει τη δημιουργία μίας εφαρμογής, η οποία με την βοήθεια του διαδικτύου θα παρέχει την δυνατότητα σε κάθε χρήστη-μέλος του προγράμματος να συνδεθεί στο σύστημα ανά πάσα στιγμή όπου κι αν βρίσκεται, να παρακολουθήσει και να επεξεργαστεί κάποια δεδομένα.

Στόχος της παρούσας διπλωματικής εργασίας είναι η υλοποίηση μίας πρότυπης (native mobile) εφαρμογής σε πλατφόρμα κινητού τηλεφώνου και λειτουργικό σύστημα iOS, μέσω της οποίας γίνεται η εποπτεία και ταυτόχρονα ο έλεγχος ενός Συστήματος Ενεργειακής Διαχείρισης Κτιρίων (BEMS). Συγκεκριμένα, η εφαρμογή αυτή συντελεί στην εποπτεία όλων των αισθητήρων (sensors) και τον έλεγχο όλων των ενεργοποιητών (actuators) που έχουν τοποθετηθεί σε χώρους και αντικείμενα ενός κτιρίου, όπως αναφέρθηκε παραπάνω. Έτσι ο χρήστης έχει την δυνατότητα να συμμετέχει ενεργά και να επηρεάζει δυναμικά και ουσιαστικά την διαδικασία βελτιστοποίησης των μοντέλων που εφαρμόζονται για την εξοικονόμηση ενέργειας του κτιρίου. Αντικείμενο της εφαρμογής ουσιαστικά αποτελεί η εξασφάλιση άμεσης και γρήγορης επικοινωνίας, αλλά και να προσφέρει μεγαλύτερη σταθερότητα και ασφάλεια από μία εφαρμογή διαδικτύου (web Application), που έχει υλοποιηθεί ήδη στα πλαίσια του προγράμματος.

Επίσης αξίζει αναφοράς ότι επιπλέον κίνητρο για την υλοποίηση της συγκεκριμένης συσκευής, αποτέλεσε η αυξανόμενη χρήση των «έξυπνων» κινητών τηλεφώνων (smartphones) [5] στην παγκόσμια αγορά, που δημιουργεί και την ανάγκη για

⁴ Positive Energy Buildings thru Better control decisions

ανάπτυξη εφαρμογών που απευθύνονται αποκλειστικά στις συσκευές αυτές. Μάλιστα το 2014 ο αριθμός των χρηστών των έξυπνων κινητών, αναμένεται να φτάσει τα 1,75 δισεκατομμύρια [6]. Τα smartphones αποτελούν κινητά με έξυπνο και εύχρηστο λογισμικό σχεδόν όμοιο με το αντίστοιχο λογισμικό που χρησιμοποιείται στους ηλεκτρονικούς υπολογιστές. Τα τελευταία χρόνια έχουν εισέλθει δυναμικά στην ελληνική αγορά παρουσιάζοντας μία σταθερά ανοδική πορεία πωλήσεων με εξαιρετικές προοπτικές περαιτέρω ανάπτυξης [7]. Ειδικότερα, τα smartphones ανήκουν στα ελάχιστα προϊόντα τεχνολογίας που το 2012 παρουσίασαν ανοδική πορεία πωλήσεων και μάλιστα της τάξης του 20%, ενώ από το 2014 υπερτερούν των υπολογιστών σε χρήση για πρόσβαση στο διαδίκτυο σε παγκόσμια κλίμακα [8], άρα η υλοποίηση μίας native mobile εφαρμογής είναι ιδανική για αυτήν την περίπτωση.

Παράλληλα, τα τελευταία χρόνια γίνεται συνεχόμενη προσπάθεια να δημιουργηθούν τα λεγόμενα «έξυπνα» σπίτια. Όλες οι μεγάλες εταιρείες⁵ αναπτύσσουν λογισμικό (software) και ηλεκτρονικό εξοπλισμό (hardware), με αποτέλεσμα αυτή η ιδέα να γίνει περισσότερο προσιτή και υλοποιήσιμη στο άμεσο μέλλον και σε όσο το δυνατόν περισσότερο ποσοστό καταναλωτών [3]. Αυτή η τεχνολογία αφορά την διαχείριση των ηλεκτρονικών συσκευών, όπως τηλεόραση, air-condition, ραδιόφωνο, την διαχείριση του φωτισμού, της θέρμανσης ή ακόμα και του συστήματος συναγερμού.

1.3 Συνεισφορά Διπλωματικής Εργασίας

Η εφαρμογή **S.M.C.** που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας, συντελεί ουσιαστικά στη δημιουργία κτιρίων με χαμηλή κατανάλωση ενέργειας, παρέχοντας παράλληλα θερμική πληρότητα και άνεση στους «ενοίκους». Επομένως, αποτελεί ένα επιστημονικό και ερευνητικό εργαλείο, καθώς συμβάλλει στην έρευνα, μελέτη και κατόπιν βελτίωση της ενεργειακής και θερμικής συμπεριφοράς ενός κτιρίου. Ουσιαστικά επιτρέπει την δυναμική και άμεση συμβολή του ανθρώπινου παράγοντα, στην εφαρμογή κατάλληλων μοντέλων που στοχεύουν στην εξοικονόμηση ενέργειας του κτιρίου. Επίσης αξίζει αναφοράς το γεγονός ότι η εξοικονόμηση ενέργειας έχει αποτέλεσμα την προστασία του περιβάλλοντος, καθώς μειώνεται η εκπομπή ρύπων που παράγονται από την υπερβολική κατανάλωση ενέργειας.

Παράλληλα, η εφαρμογή διατηρεί έναν εκπαιδευτικό ρόλο, καθώς οι χρήστες εφαρμόζουν αρχικά οποιεσδήποτε αλλαγές επιθυμούν στην θερμοκρασία του κτιρίου και έπειτα δέχονται από το σύστημα τις συνέπειες των αλλαγών που πραγματοποίησαν. Αυτό έχει αποτέλεσμα την συνειδητοποίησή τους σε ενεργειακά θέματα, αλλά και την εκπαίδευσή τους στον τομέα της λήψης σωστών αποφάσεων.

⁵ Η Apple ανακοίνωσε ότι το νέο λογισμικό iOS 8 θα υποστηρίζει εφαρμογές για έλεγχο και διαχείριση έξυπνων σπιτιών. Το ίδιο ισχύει και με την εταιρεία Samsung. [4]

1.4 Σύντομη περιγραφή Λειτουργιών και δυνατοτήτων

Η εφαρμογή ξεκινάει με την σύνδεση του χρήστη στη βάση δεδομένων. Ο χρήστης δεν έχει την δυνατότητα δημιουργίας λογαριασμού και προφίλ καθώς η εφαρμογή δεν έχει χαρακτήρα κοινωνικής δικτύωσης, ενώ σε πρωταρχικό στάδιο υλοποιήθηκε για εκπαιδευτικούς και ερευνητικούς λόγους. Το προφίλ κάθε χρήστη και γενικότερα η πρόσβαση σε κάθε χρήστη δίνεται από τον διαχειριστή της βάσης δεδομένων (administrator).

Έπειτα ο χρήστης αφού έχει συνδεθεί επιτυχώς, έχει την δυνατότητα να περιηγηθεί σε όσα δωμάτια – χώρους τού έχει δοθεί δυνατότητα πρόσβασης από τον διαχειριστή και να δει όλα τα σήματα - αισθητήρες που βρίσκονται σε κάθε ένα από αυτά. Επίσης υπάρχει η δυνατότητα σε κάθε δωμάτιο – χώρο που διαθέτει μία μονάδα κλιματιστικού και υπάρχουν σήματα ελέγχου (actuators), να δει την κατάστασή τους αλλά και να επέμβει σε αυτά αλλάζοντας σε όποια τιμή επιθυμεί. Για παράδειγμα ο χρήστης έχει την δυνατότητα να ενεργοποιήσει το air-condition στο γραφείο του σε συγκεκριμένους βαθμούς κελσίου. Τέλος, κάθε χρήστης έχει την δυνατότητα να δει την πορεία των τιμών για κάθε αισθητήρα αλλά και την ακριβή τιμή του σε δεδομένη χρονική στιγμή, μέσα από ένα διάγραμμα επιλέγοντας να εμφανίζονται τα δεδομένα ανά ώρα, ημέρα και μήνα.

1.5 Διάρθρωση επόμενων κεφαλαίων

Στις προηγούμενες παραγράφους περιγράφηκε σε γενικές γραμμές, ο στόχος της παρούσας διπλωματικής εργασίας, η σχέση με το πρόγραμμα PEBBLE, αποφασηνίστηκε τι θα υλοποιηθεί μέσω της συγκεκριμένης εφαρμογής και ποια είναι η συνεισφορά της στο πρόγραμμα.

Στο **2ο Κεφάλαιο** που ακολουθεί, θα γίνει εκτενής αναφορά και ανάλυση του προγράμματος PEBBLE. Θα αναφερθούν οι συνθήκες κάτω από τις οποίες δημιουργήθηκε, θα περιγραφεί η δομή και η εφαρμογή του στο Πολυτεχνείο Κρήτης. Επίσης, γίνει αναφορά στα αίτια τα οποία οδήγησαν στην υλοποίηση μίας εφαρμογής για πλατφόρμα κινητού τηλεφώνου και ειδικά στο λογισμικό (software) και υλικό (hardware) της εταιρείας Apple.

Ακολουθώντας, στο **3ο Κεφάλαιο** θα γίνει αναφορά για την υλοποίηση της εφαρμογής S.M.C.. Αναλύονται εκτενώς, η αρχιτεκτονική του MVC μοντέλου και η γλώσσα προγραμματισμού Objective C που αποτελούν την βάση για να δομηθεί η εφαρμογή.

Επίσης θα εξεταστεί η τεχνολογία του λειτουργικού iOS, ενώ γίνεται αναφορά στα προγραμματιστικά εργαλεία που συνετέλεσαν στην υλοποίηση της εφαρμογής. Τέλος περιγράφεται ο τρόπος εγκατάστασης της εφαρμογής σε συσκευή iPhone.

Το **4ο Κεφάλαιο** έχει ίδια δομή με το τρίτο κεφάλαιο, αλλά με την διαφορά ότι αναφέρεται στην υλοποίηση του εξυπηρετητή (server). Συγκεκριμένα, αναλύεται η αρχιτεκτονική των υπηρεσιών δικτύου, του πρωτοκόλλου REST, ενώ γίνεται αναφορά στις τεχνολογίες της XML, του JSON, της PHP και της MySQL. Τέλος, περιγράφονται τα εργαλεία που βοήθησαν στην υλοποίηση ενός REST webserver.

Στο **5ο Κεφάλαιο** που ακολουθεί αναπτύσσεται το ζήτημα των αρχών και των κανόνων που τηρήθηκαν για την ορθή σχεδίαση και υλοποίηση του γραφικού περιβάλλοντος της εφαρμογής (User Interface), ενώ αναφέρονται και άλλες σχεδιαστικές λεπτομέρειες που πληροί η εφαρμογή S.M.C..

Στο **6ο Κεφάλαιο** γίνεται εκτενής ανάλυση των αποτελεσμάτων χρήσης της εφαρμογής. Συγκεκριμένα, αναλύεται λεπτομερώς η λειτουργικότητα της εφαρμογής, με χρήση αντίστοιχων screenshots που περιγράφουν τις δυνατότητες αλλά και όλες τις ενέργειες και λειτουργίες που μπορεί να πραγματοποιήσει κάθε χρήστης.

Στην συνέχεια, στο **7ο Κεφάλαιο**, γίνεται ανασκόπηση της παρούσας εργασίας και αναφέρονται τα συμπεράσματα που διεξήχθησαν από την υλοποίηση της εφαρμογής S.M.C..

Τέλος, στο **8ο Κεφάλαιο** γίνεται σύντομη αναφορά σε μελλοντικές βελτιώσεις που επιδέχεται η εφαρμογή, ώστε να γίνει περισσότερο λειτουργική, αποτελεσματική και να παρέχει μεγαλύτερη ασφάλεια.

Στο **9ο Κεφάλαιο** γίνεται απλή αναφορά στη βιβλιογραφία, τις πηγές και τους συνδέσμους που χρησιμοποιήθηκαν και ενέπνευσαν την συγγραφή της παρούσας εργασίας.

2. ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΕΡΕΥΝΗΤΙΚΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ PEBBLE

2.1 Γενικά για το P.E.B.B.L.E.

Το P.E.B.B.L.E. (**P**ositive **E**nergy **B**uildings thru **B**etter contro**L** d**E**cisions) [2] αποτελεί ένα ευρωπαϊκό πρόγραμμα που εκπονείται στα πλαίσια επίτευξης των στόχων 20-20-20 που έχει θέσει η Ευρωπαϊκή Ένωση και αναφέρθηκαν στο προηγούμενο κεφάλαιο. Το πρόγραμμα χρηματοδοτείται από την Ευρωπαϊκή Ένωση, στα πλαίσια του 7^{ου} Προγράμματος Πλαισίου στην περιοχή των Τεχνολογιών Πληροφορικής και Επικοινωνιών για εξοικονόμηση ενέργειας και συμμετέχουν επτά ερευνητικοί φορείς (πανεπιστήμια, ερευνητικά κέντρα και εταιρείες) από 5 ευρωπαϊκές χώρες (Αυστρία, Γαλλία, Γερμανία, Ελβετία, Ελλάδα). Το πρόγραμμα συντονίζεται από το Πολυτεχνείο Κρήτης και το συνολικό κόστος του έργου ανέρχεται στα 3,3Μ€.

Στόχος της προτεινόμενης ερευνητικής δραστηριότητας όπως αποτυπώνεται και αναφέρεται στο πλήρες όνομα του προγράμματος “**P**ositive **E**nergy **B**uildings thru **B**etter contro**L** d**E**cisions”, είναι η ανάπτυξη μιας ολοκληρωμένης μεθοδολογίας βασιζόμενη σε μια δυναμική θεώρηση του κτιρίου, η οποία αποσκοπεί μέσα από τον περιορισμό των ενεργειακών απαιτήσεων κατά τη λειτουργία των κτιρίων και τη βέλτιστη χρήση ανανεώσιμων πηγών ενέργειας, στη δημιουργία κτιρίων Μηδενικής ή ακόμα και Θετικής Ενεργειακής κατανάλωσης (ΚΜΕΙ ή ΚΘΕΙ) με χρήση Συστημάτων Βελτιστοποίησης και Ελέγχου των Κτιρίων (ΣΒΕΚ). Ο απώτερος σκοπός είναι η μετατροπή των υφιστάμενων κτιρίων στην Ευρώπη σε κτίρια μηδενικής ή και θετικής ενεργειακής αποδοτικότητας σύμφωνα με την απόφαση της Ευρωπαϊκής Ένωσης.

Το ευρωπαϊκό ερευνητικό πρόγραμμα P.E.B.B.L.E. έχει αντικείμενο την ανάπτυξη τεχνογνωσίας και τεχνολογιών, οι οποίες θα υποστηρίζουν την κατασκευή ΚΘΕΙ. Οι τεχνολογίες αυτές αφορούν την δεύτερη φάση της ζωής ενός κτιρίου, αυτήν κατά τη διάρκεια λειτουργίας του. Στα σύγχρονα κτίρια γραφείων αυτό επιτυγχάνεται με την εγκατάσταση συστημάτων διαχείρισης ενέργειας, BEMS, τα οποία διέπονται από ένα στατικό σύνολο κανόνων και έτσι η αποδοτικότητα είναι χαμηλή, ενώ οι απώλειες ενέργειας πολλές φορές ανέρχονται στο 20%. Αυτό οφείλεται συνήθως είτε σε λάθος εγκατάσταση, είτε σε λάθος μελέτη κατά την φάση της σχεδίασης του συστήματος. Ακόμα και σε περίπτωση σωστής εγκατάστασης και μελέτης του συστήματος, είναι πολύ δύσκολο να ενσωματωθούν στρατηγικές τόσο απλοϊκές όσο αυτές που χρησιμοποιούνται σε αυτά τα συστήματα, ιδιαίτερα σε περιπτώσεις που τα κτίρια έχουν υψηλές ενεργειακές απαιτήσεις ή υπάρχει μεγάλη πολυπλοκότητα στο σύστημα και πρέπει να τηρείται αρμονία υπολογίζοντας πάντα και την επιρροή των συνθηκών του καιρού που επικρατούν.

Επιπλέον, στόχος είναι η κατασκευή νέων συστημάτων διαχείρισης ενέργειας κτιρίων, τα οποία θα λαμβάνουν αποφάσεις σε πραγματικό χρόνο για τη λειτουργία των υποσυστημάτων του κτιρίου (κλιματιστικά, φωτισμός, εξαερισμός), λαμβάνοντας υπόψη και τη διαθεσιμότητα ενέργειας από εγκατεστημένες ΑΠΕ. Αποτέλεσμα αυτού θα είναι η βέλτιστη λειτουργία του κτιρίου από πλευράς ενεργειακών απαιτήσεων καθώς και η επίτευξη συνθηκών θερμικής άνεσης για τους χρήστες. Τα βασικά συστατικά της προτεινόμενης μεθοδολογίας είναι: πρώτον, η ύπαρξη υπολογιστικών μοντέλων υψηλής ακρίβειας, τα οποία επιτρέπουν την ακριβή πρόβλεψη των ενεργειακών απαιτήσεων του κτιρίου, καθώς και εκτίμηση της διαθέσιμης ενέργειας (ΑΠΕ)· δεύτερον, η ανάπτυξη μεθοδολογίας ελέγχου, η οποία με τη βοήθεια των υπολογιστικών μοντέλων, θα μπορεί να λαμβάνει βέλτιστες αποφάσεις για τη λειτουργία των συστημάτων του κτιρίου· και τρίτον, η ανάπτυξη τεχνολογιών για ασύρματους αισθητήρες και ενεργοποιητές οι οποίοι θα επιτρέπουν την καταγραφή της κατάστασης του κτιρίου και την εκτέλεση των εντολών ελέγχου.

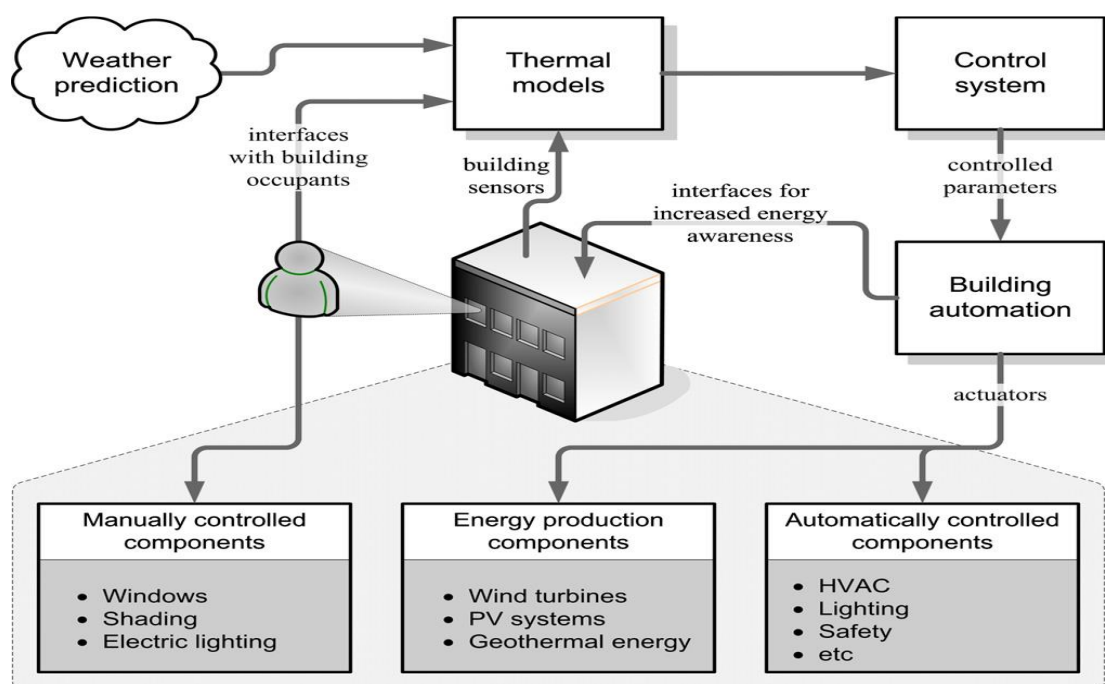
Στα πλαίσια του προγράμματος έχει προβλεφθεί η εγκατάσταση του συστήματος και η αξιολόγησή της απόδοσής του σε τρία κτίρια γραφείων στην Ευρώπη: το πρώτο βρίσκεται στο Κάσελ, Γερμανίας (Fraunhofer Institute for Building Physics), το δεύτερο στο Άαχεν, Γερμανίας (E.ON Energy Building, RWTH Aachen) και το τρίτο στα Χανιά, Ελλάδα (κτίριο Τεχνικών Υπηρεσιών, Πολυτεχνείο Κρήτης) [9].

2.1.1 Δομή του προγράμματος P.E.B.B.L.E.

Στις παρακάτω εικόνες (Εικόνα 2.1, Εικόνα 2.2) παρουσιάζονται τα τρία απαραίτητα στοιχεία του προτεινόμενου συστήματος που αναπτύσσεται στο πρόγραμμα P.E.B.B.L.E. [10]. Το πρώτο στοιχείο αφορά τα ακριβή και αποδοτικά μοντέλα προσομοίωσης για ΚΘΕΙ, τα οποία ενσωματώνουν όλα τα παθητικά και ενεργητικά συστήματα παραγωγής ενέργειας, και τα οποία προβλέπουν τη θερμική απόκριση του κτιρίου, τις ενεργειακές απαιτήσεις του και τη θερμική άνεση των ενοίκων. Αυτά βασίζονται είτε σε μοντέλα πρόβλεψης των καιρικών συνθηκών, είτε σε χρονοσειρές δεδομένων καιρικών συνθηκών. Τα συγκεκριμένα προγνωστικά μοντέλα κτιρίων βασίζονται σε φυσική μοντελοποίηση, αποφεύγοντας έτσι τις κλασικές και λιγότερο αξιόπιστες προσεγγίσεις που βασίζονται σε χρονοσειρές δεδομένων. Ένα επίσης ιδιαίτερα σημαντικό στοιχείο είναι οι αισθητήρες, οι ενεργοποιητές και οι διεπαφές χρηστών, που θα χρησιμοποιούνται με σκοπό την διευκόλυνση της επικοινωνίας και της ανταλλαγής πληροφορίας από το φυσικό επίπεδο, στο επίπεδο προσομοίωσης. Τέλος, το τρίτο σημαντικό στοιχείο είναι τα αποδοτικά και αξιόπιστα Συστήματα Βελτιστοποίησης και Ελέγχου του Κτιρίου (ΣΒΕΚ), τα οποία συνδυάζουν την διαθέσιμη πληροφορία από τους αισθητήρες με τα θερμικά μοντέλα προσομοίωσης, ώστε να αξιολογήσουν τα πιθανά σενάρια, και τελικά να λάβουν την βέλτιστη απόφαση για την λειτουργία των υποσυστημάτων, καθώς και την επίτευξη της αποδεκτής κατάστασης κλιματισμού από τους χρήστες του κτιρίου, σε (σχεδόν) πραγματικό

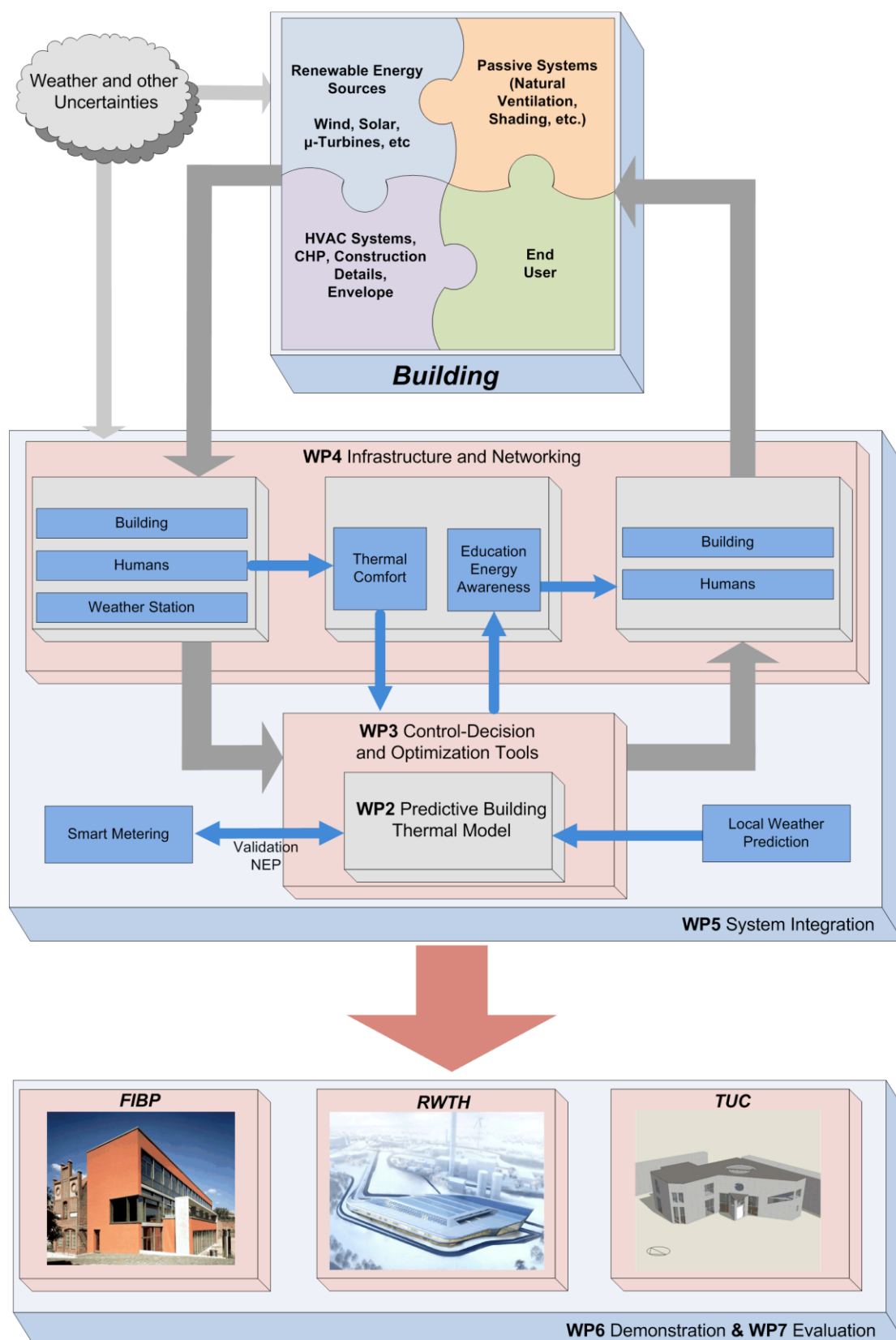
χρόνο. Επιπροσθέτως, μία από τις πρωτοτυπίες του προτεινόμενου έργου, αποτελεί ο διπλός ρόλος των ενοίκων, πότε ως αισθητήρες και πότε ως ενεργοποιητές. Συγκεκριμένα, οι χρήστες δρουν ως αισθητήρες μέσω των διεπαφών χρηστών, ειδοποιώντας το σύστημα για τις προτιμήσεις τους ως προς την θερμική άνεση, ενώ δέχονται από το σύστημα πληροφορία σχετική με τις συνέπειες των εκάστοτε ενεργειών τους, με απώτερο στόχο την συνειδητοποίησή τους σε ενεργειακά θέματα, αλλά και την εκπαίδευσή τους στον τομέα της λήψης σωστών ενεργειών.

Ουσιαστικά το πρόγραμμα PEBBLE πραγματεύεται την ανάπτυξη μίας τεχνολογίας πληροφοριών και επικοινωνίας⁶, ή διαφορετικά ενός υπολογιστικού πληροφοριακού συστήματος που θα λύνει τα προβλήματα που αναφέρθηκαν παραπάνω. Αυτό έχει ως αποτέλεσμα την υλοποίηση μίας διαδικασίας τριών επιπέδων **Sense, Think, Act** (STA) [9]. Μέσω της συγκεκριμένης διαδικασίας τα συστήματα BEMS δεν διέπονται από ένα στατικό σύνολο κανόνων, αλλά υπηρετούν στρατηγικές που αλλάζουν αυτόματα συνεχώς, με δυναμικό τρόπο και πάντα βελτιστοποιώντας την υπάρχουσα κατάσταση. Το πρώτο επίπεδο, **Sense**, αφορά την συλλογή και συνάθροιση όλων των πληροφοριών από έναν αριθμό ανομοιογενών πηγών, όπως αισθητήρες και μετεωρολογικό σταθμό. Το δεύτερο επίπεδο, **Think**, περιλαμβάνει μεθοδολογίες διαφόρων μοντέλων με στόχο την βελτιστοποίηση και την αυτόματη παραγωγή ενεργειακά έξυπνων αποφάσεων. Τέλος, το τρίτο επίπεδο, **Act**, αναλαμβάνει την σύνδεση μεταξύ των βέλτιστων αποφάσεων με το κτίριο.



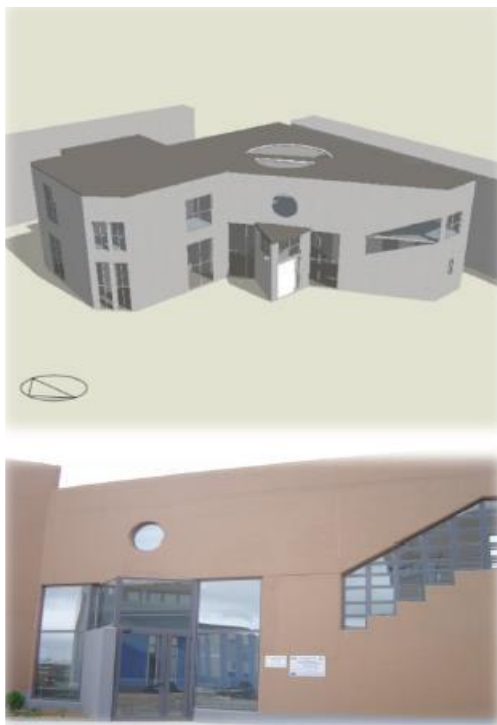
Εικόνα 2.1 – Η προσέγγιση του PEBBLE ως προς τα ΣΒΕΚ

⁶ Τεχνολογία πληροφορίας και επικοινωνίας – Information and Communication Technology, ICT



Εικόνα 2.2 – Γενική γραφική απεικόνιση του συστήματος PEBBLE

2.1.2 Το P.E.B.B.L.E. στο Πολυτεχνείο Κρήτης



Εικόνα 2.3 – Απεικόνιση κτιρίου Τεχνικών Υπηρεσιών, Πολυτεχνείο Κρήτης

Στα πλαίσια του προγράμματος PEBBLE εντάσσεται και η παρούσα εργασία. Συγκεκριμένα, αφορά την εφαρμογή του προγράμματος στο κτίριο Τεχνικών Υπηρεσιών που βρίσκεται στον χώρο του Πολυτεχνείου Κρήτης, στην περιοχή Κουνουπιδιανά Χανίων. Το κτίριο αποτελείται από δύο ορόφους, περιέχει 19 θερμικές ζώνες, έχει υψηλές ενεργειακές απαιτήσεις (περίπου 130 KWh/m^2 το χρόνο), σημαντικά προβλήματα θερμικής άνεσης και φυσικό αερισμό [11]. Επίσης αξίζει να σημειωθεί ότι το κτίριο δεν διέθετε κανένα σύστημα διαχείρισης ενέργειας BEMS πριν το πρόγραμμα. Τα ενεργειακά συστήματα που περιλαμβάνει είναι φωτοβολταϊκές συστοιχίες, ανεμογεννήτριες, παράθυρα οροφής αυτόματης ρύθμισης καθώς και σκίαστρα αυτόματης ρύθμισης. Ως προς το σύστημα HVAC του κτιρίου, υπάρχει σύστημα κεντρικής θέρμανσης, με χρήση λέβητα πετρελαίου, και για την ψύξη υπάρχουν air-conditions [12].

Το συνολικό πρόγραμμα για την εποπτεία των αισθητήρων αλλά και τον έλεγχο των αντίστοιχων σημάτων που έχουν εγκατασταθεί στο κτίριο Τεχνικών Υπηρεσιών ως προς την υλοποίησή του, απαιτεί αρκετή δουλειά σε επίπεδο προγραμματισμού (software) αλλά και υλικού (hardware). Απαιτήθηκε να γίνει καινούρια εγκατάσταση αφού δεν υπήρχε εγκατεστημένο κανένα σύστημα διαχείρισης ενέργειας. Έτσι εγκαταστάθηκαν 200 αισθητήρες (sensors) και ενεργοποιητές (actuators) που

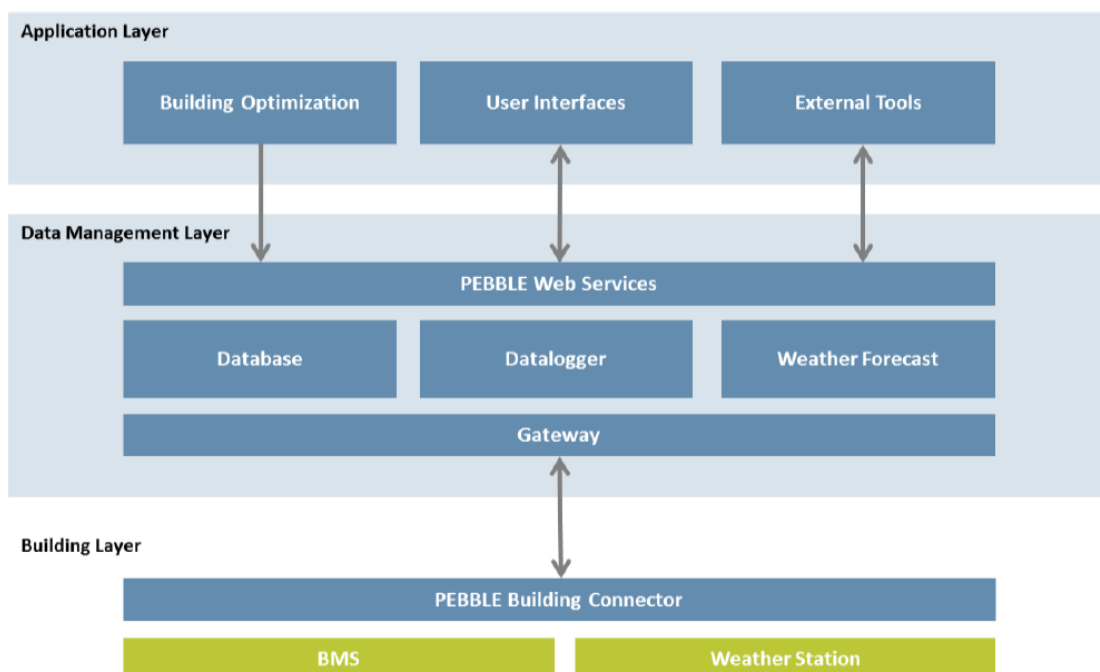
λειτουργούν ασύρματα χρησιμοποιώντας διαφορετικά πρωτόκολλα. Η πρόσβαση στα δεδομένα που παρέχουν τα σήματα αυτά, υλοποιήθηκε με την βοήθεια ενός εξυπηρετητή (OPC server) που παρέχει ασφάλεια και πιο άμεση επικοινωνία με τα PLC's μέσω ενός γραφικού περιβάλλοντος που επίσης υλοποιήθηκε. Επιπλέον, υλοποιήθηκε συνδεσιμότητα με εξωτερικές πηγές για την λήψη και αποστολή δεδομένων για τις συνθήκες του καιρού από και προς τον μετεωρολογικό σταθμό. Γενικότερα αναπτύχθηκαν πληθώρα interfaces για την εύκολη, γρήγορη, άμεση διαχείριση και αλληλεπίδραση του χρήστη με τα μέρη του υλικού (hardware) του συστήματος.

Το κομμάτι της υλοποίησης και αρχιτεκτονικής του PEBBLE διακρίνεται σε τρία βασικά επίπεδα όπως διακρίνεται στην *Εικόνα 2.4*. Σε γενικές γραμμές αναλύονται ως εξής:

- Το πρώτο επίπεδο (κατώτερο), **Building Layer**, αποτελεί την εγκατάσταση των αισθητήρων καταγραφής δεδομένων και αισθητήρων ελέγχου στους χώρους – δωμάτια καθώς και την εγκατάσταση των προγραμματιζόμενων λογικών ελεγκτών (P.L.C.) που είναι υπεύθυνα για την συλλογή, συγκέντρωση των δεδομένων αλλά και την αποστολή σημάτων στους αισθητήρες ελέγχου. Ουσιαστικά αποτελεί την επικοινωνία μεταξύ των PLC's και του μετεωρολογικού σταθμού.
- Το δεύτερο επίπεδο, **Data Management Layer**, αφορά τον προγραμματισμό των P.L.C.'s και κυρίως την συλλογή και καταγραφή (Data logger) όλων των δεδομένων σε κατάλληλες βάσεις δεδομένων (Databases) σε συγκεκριμένους servers. Επίσης υποστηρίζει συνδεσιμότητα με εξωτερικές υπηρεσίες, όπως αυτή της πρόγνωσης καιρικών συνθηκών (Weather Forecast). Ο καταγραφέας δεδομένων (Data logger) έχει την δυνατότητα να καταγράφει δεδομένα ταυτόχρονα σε περισσότερες από μία βάσεις δεδομένων, γεγονός που τον καθιστά γρήγορο, καθώς έχει την δυνατότητα να καταγράφει σε μία μικρότερη τοπική βάση δεδομένων και έπειτα σε μία μεγαλύτερη που μπορεί βρίσκεται οπουδήποτε. Επίσης σημαντικό κομμάτι του επιπέδου αυτού αποτελούν οι υπηρεσίες δικτύου (Web-Services) που έχουν υλοποιηθεί για την επικοινωνία του επιπέδου αυτού με το τρίτο επίπεδο, Application layer.
- Το τρίτο (ανώτερο) και τελευταίο επίπεδο, **Application Layer**, είναι ουσιαστικά το interface των δύο προηγούμενων επιπέδων, δηλαδή αποτελεί το γραφικό περιβάλλον χρήστη (User Interface) για την διαχείρισή τους. Αφορά την εφαρμογή, είτε αυτή είναι μία web εφαρμογή (web application), είτε είναι μια εφαρμογή σε πλατφόρμα κινητού τηλεφώνου (mobile native application). Αυτή είναι υπεύθυνη ώστε να παρέχει ένα εύχρηστο και φιλικό προς τον χρήστη περιβάλλον λογισμικού, για να διαχειρίζεται τα σήματα (διακόπτες) ελέγχου όπως επίσης και να εποπτεύει τις τιμές των σημάτων, όχι μόνον σε πραγματικό χρόνο αλλά και σε συνολικό διάστημα που ορίζει ο ίδιος. Επίσης περιλαμβάνει και την υπηρεσία βελτιστοποίησης κτιρίου (Building Optimization Service). Στα πλαίσια της παρούσας εργασίας σχεδιάστηκε, οργανώθηκε και υλοποιήθηκε μία τέτοια

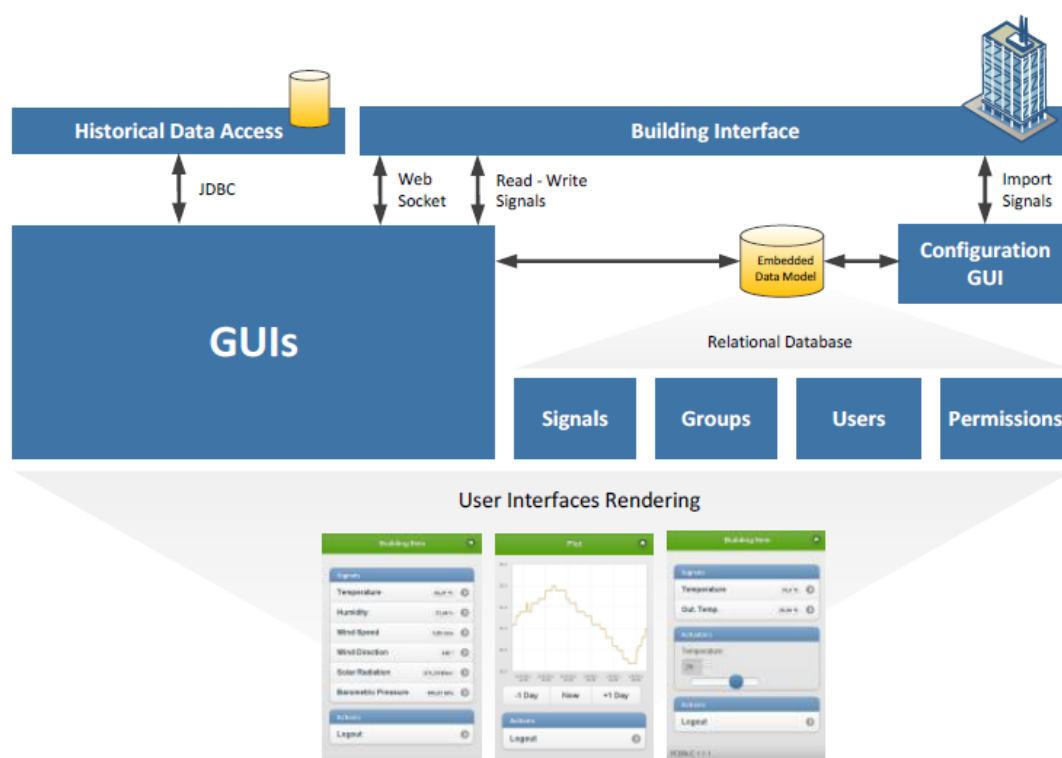
εφαρμογή (native mobile application) σε λειτουργικό σύστημα κινητού τηλεφώνου iOS, με την ονομασία Sensor Monitor Controller (**S.M.C.**) Application.

Επίσης αξίζει αναφοράς ότι παράλληλα με τα διάφορα GUI's⁷ έχουν δημιουργηθεί ομάδες χρηστών, καθώς δεν γίνεται όλοι οι χρήστες να έχουν πρόσβαση σε όλα τα δεδομένα - αισθητήρες. Έτσι, μόνο οι χρήστες - διαχειριστές έχουν πρόσβαση σε όλα τα γραφεία και όλους τους χώρους του κτιρίου, ενώ άλλου είδους χρήστες έχουν πρόσβαση μόνο στα γραφεία τους. Μέσα στο PEBBLE οι χρήστες έχουν την δυνατότητα να λειτουργούν είτε ως αισθητήρες είτε ως ενεργοποιητές δίνοντας σημαντική βοήθεια στις αποφάσεις ελέγχου του συστήματος. Οι χρήστες έχουν περιορισμένο έλεγχο στο σύστημα ψύξης, ενώ για να έχουν την δυνατότητα να πάρουν ενεργειακά αποδοτικές αποφάσεις πρέπει να παρέχονται πληροφορίες για τις τιμές των αισθητήρων ενός χώρου-δωματίου καθώς και γραφικές παραστάσεις για την κατανάλωση ενέργειας.

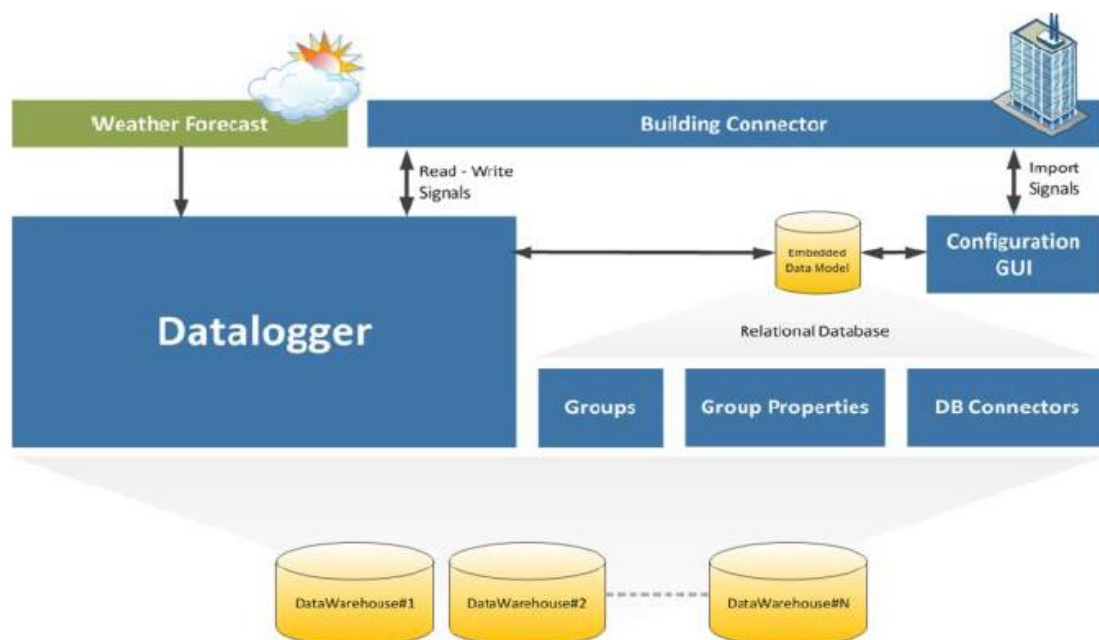


Εικόνα 2.4 – Απεικόνιση αρχιτεκτονικής PEBBLE

⁷ GUI means Graphic User Interface:Γραφικό περιβάλλον χρήστη



Εικόνα 2.5 – Απεικόνιση Υπηρεσίας Γραφικού Περιβάλλοντος (GUI)



Εικόνα 2.6 – Απεικόνιση Υπηρεσίας Καταγραφέα Δεδομένων (Data Logger)

2.2 Διαδικτυακή (Web) Εφαρμογή P.E.B.B.L.E.

Στην παράγραφο [2.1.2] έγινε αναφορά σχετικά με τα τρία επίπεδα που απαιτούνται για το κομμάτι της υλοποίησης του προγράμματος και συγκεκριμένα της εφαρμογής **S.M.C.**. Καθίσταται σαφές ότι για την υλοποίηση της εφαρμογής που βρίσκεται στο τρίτο και ανώτερο επίπεδο, αποτελεί απαραίτητη προϋπόθεση η ύπαρξη και υλοποίηση των δύο πρώτων επιπέδων. Όταν ξεκίνησε η εκπόνηση της παρούσας εργασίας, τα επίπεδα αυτά ήταν ήδη υλοποιημένα και κυρίως το τρίτο επίπεδο, δηλαδή μία web εφαρμογή, η οποία έχει την δυνατότητα να λειτουργεί σε όλες τις πλατφόρμες των λειτουργικών συστημάτων μέσω χρήσης του ιντερνέτ και οποιουδήποτε προγράμματος πλοήγησης (browser). Ο κύριος Δημήτριος Ρόβας σε συνεργασία με τον κύριο Κυριάκο Κατσιγαράκη και την υπόλοιπη ομάδα του ερευνητικού προγράμματος PEBBLE, έκαναν την υλοποίηση και των τριών επιπέδων. Αξίζει αναφοράς το γεγονός ότι η συγκεκριμένη web εφαρμογή αποτέλεσε την βάση για την εφαρμογή **S.M.C.** που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας και περιγράφεται λεπτομερώς στην συνέχεια. Παράδειγμα του interface της web εφαρμογής φαίνεται στην *Εικόνα 2.7*.



*Εικόνα 2.7 – Απεικόνιση Υπηρεσίας Καταγραφέα Δεδομένων (Data Logger)
(πηγή: <http://host251.dpem.tuc.gr>)*

2.3 Εφαρμογή σε πλατφόρμα κινητού τηλεφώνου στα πλαίσια του P.E.B.B.L.E.

2.3.1 Πλεονεκτήματα – Μειονεκτήματα native mobile application

Η υλοποίηση μίας εφαρμογής σε πλατφόρμα λογισμικού για κινητά τηλέφωνα (native mobile application) αποτελεί μεγάλο και ενδιαφέρον κίνητρο για τους προγραμματιστές. Η ανάπτυξη, λοιπόν, μίας τέτοιας εφαρμογής έναντι των εφαρμογών διαδικτύου (web), εμπεριέχει κάποια μειονεκτήματα αλλά τα πλεονεκτήματα είναι περισσότερα. Τα κυριότερα σημειώνονται ως εξής :

- Αρχικά αξίζει αναφοράς το γεγονός ότι αυτού του τύπου οι εφαρμογές εγκαθίστανται απευθείας στο λογισμικό του κινητού τηλεφώνου, το οποίο σημαίνει ότι λειτουργούν μόνο στις συγκεκριμένες πλατφόρμες λογισμικού, σε αντίθεση με τις web εφαρμογές που δεν απαιτούν εγκατάσταση αλλά λειτουργούν μέσω οποιουδήποτε περιηγητή (browser).
- Επιπροσθέτως, η κατασκευή τους απαιτεί πλήρη γνώση μίας συγκεκριμένης γλώσσας προγραμματισμού, ανάλογα με την πλατφόρμα λογισμικού που θα λειτουργήσει η συγκεκριμένη εφαρμογή, ενώ για την κατασκευή web εφαρμογής απαιτείται γνώση πολλών διαφορετικών τεχνολογιών και γλωσσών προγραμματισμού. Συγκεκριμένα, οι εφαρμογές για κινητά τηλέφωνα που λειτουργούν με πλατφόρμα λογισμικού Android, απαιτείται καλή γνώση της γλώσσας προγραμματισμού Java. Αντίστοιχα σε πλατφόρμα Windows mobile χρειάζεται πλήρης γνώση της C++, ενώ για την ανάπτυξη κάποιας εφαρμογής σε λειτουργικό iOS της Apple, απαιτείται καλή γνώση της γλώσσας προγραμματισμού Objective-C.
- Τα κυριότερα πλεονεκτήματα των native εφαρμογών έναντι των web εφαρμογών είναι ότι προσφέρουν :
 1. *Ταχύτητα*

Οι native εφαρμογές όπως αναφέρθηκε παραπάνω, είναι εγκατεστημένες στην συσκευή του κινητού τηλεφώνου και αξιοποιούν περισσότερους πόρους. Συγκεκριμένα, εκμεταλλεύονται περισσότερη μνήμη και ισχύ από τον επεξεργαστή με αποτέλεσμα να εκτελούνται με μεγαλύτερη ταχύτητα.
 2. *Αποτελεσματικότητα*

Για τους ίδιους λόγους η native εφαρμογή αξιοποιώντας περισσότερους πόρους δίνει πολύ καλύτερα αποτελέσματα.

3. Ασφάλεια

Πολλές φορές παρατηρείται το γεγονός ότι υπάρχει κακόβουλο λογισμικό σε ιστοσελίδες και προσπάθεια παραβίασης προσωπικών δεδομένων με χρήση ιών (malware), όπου τις περισσότερες φορές ο υπεύθυνος δεν είναι εφικτό να εντοπιστεί. Επομένως, είναι προφανές ότι παρέχεται μεγαλύτερη ασφάλεια μέσω των native εφαρμογών, καθώς παρέχουν ασφάλεια των προσωπικών δεδομένων και της «υγείας» των συσκευών, αποτρέποντας οποιονδήποτε μέσω ιντερνέτ να τις παραβιάσει. Επίσης, σε περίπτωση τέτοιου είδους προβλήματος, υπεύθυνος είναι είτε ο χρήστης είτε ο προγραμματιστής που παρέχει την εφαρμογή, γεγονός που βοηθάει στην επαναφορά της.

4. Ποιότητα

Ανέκαθεν η ποιότητα αποτελεί βασικό κριτήριο επιλογής. Συμπέρασμα των προηγούμενων χαρακτηριστικών που προσφέρουν οι native εφαρμογές είναι ότι παρέχουν ανώτερη ποιότητα και επομένως υπερτερούν των web εφαρμογών.

- Σημαντικό πλεονέκτημα πρακτικής σημασίας, αποτελεί η εκμετάλλευση όλων των πόρων του υλικού (hardware) του κινητού τηλεφώνου που προσφέρει μία native εφαρμογή, σε αντίθεση με αυτήν μίας web εφαρμογής. Πλέον οι συσκευές των κινητών τηλεφώνων της νέας τεχνολογίας, γνωστά και ως «έξυπνα τηλέφωνα» (smartphones), παρέχουν πολλά χαρακτηριστικά τα οποία μία web εφαρμογή δεν έχει την δυνατότητα να τα αξιοποιήσει. Συγκεκριμένα, τα smartphones περιλαμβάνουν διάφορους αισθητήρες όπως κάμερα, σύστημα εντοπισμού γεωγραφικής θέσης (GPS), γυροσκόπιο, επιταχυνσιόμετρο (accelerometer) το οποίο επιτρέπει στη συσκευή να αναγνωρίζει τον προσανατολισμό της και συνεπώς να αλλάζει την απεικόνιση της οθόνης ανάλογα με τον τρόπο που το κρατά ο χρήστης. Επομένως καθίσταται σαφές το γεγονός, ότι η εκμετάλλευση – αξιοποίηση των επιπλέον αισθητήρων (hardware) που προσφέρουν σήμερα οι συσκευές των κινητών τηλεφώνων, είναι απαραίτητη και δίνει την δυνατότητα μεγαλύτερης ευελιξίας και περισσότερων λειτουργιών από τις εφαρμογές (software).
- Τέλος, σημαντικό ρόλο επιτελεί το οικονομικό όφελος. Το οικονομικό σκέλος αποτελεί μειονέκτημα αλλά ταυτόχρονα και πλεονέκτημα. Συγκεκριμένα, η υλοποίηση μίας native εφαρμογής αρκετές φορές, αν όχι τις περισσότερες, έχει μεγαλύτερο κόστος αλλά ταυτόχρονα δίνει την δυνατότητα για οικονομικά οφέλη. Αυτό συμβαίνει διότι η διάθεση των εφαρμογών και η εγκατάστασή τους στην πλατφόρμα των κινητών τηλεφώνων γίνεται μέσω των ηλεκτρονικών καταστημάτων. Κάθε εταιρεία έχει το αντίστοιχο ηλεκτρονικό κατάστημα εφαρμογών και παρέχει τις αντίστοιχες εφαρμογές. Οι εταιρείες λοιπόν, δίνουν δυνατότητα στους προγραμματιστές να παρέχουν τις εφαρμογές που κατασκευάζουν στο καταναλωτικό κοινό έναντι ενός αντίτιμου, πληρώνοντας

πάντα κάποιους βασικούς οικονομικούς όρους. Κατ' αυτόν τον τρόπο ο προγραμματιστής έχει την δυνατότητα να πουλήσει την εφαρμογή του και να βγάλει πολλά χρήματα, σε αντίθεση με τις web εφαρμογές που αυτό επιτυγχάνεται μόνο μέσω των διαφημίσεων, το οποίο είναι πολύ χρονοβόρο και δύσκολο.

2.3.2 Εφαρμογή Sensor Monitor Controller (S.M.C.)

Η παρούσα εργασία πραγματεύεται την υλοποίηση μίας native mobile εφαρμογής. Η συγκεκριμένη εφαρμογή σχεδιάστηκε και υλοποιήθηκε έχοντας ως βάση την ήδη υπάρχουσα web εφαρμογή, παρέχοντας όμως όσο δυνατόν περισσότερες βελτιώσεις και διορθώσεις. Ο λόγος που προτιμήθηκε να σχεδιαστεί εκ νέου μία εφαρμογή σε περιβάλλον λογισμικού για κινητά τηλέφωνα, όπως αναφέρθηκε στο πρώτο κεφάλαιο, είναι ότι η χρήση των κινητών τηλεφώνων αποτελεί πλέον σημαντικό και απαραίτητο εργαλείο όχι μόνο σε προσωπικό αλλά κυρίως σε επαγγελματικό επίπεδο. Πλέον οι συσκευές κινητών τηλεφώνων είναι «έξυπνες», γνωστές με την ονομασία smartphones, καθώς περιλαμβάνουν πολύπλοκο λογισμικό, περίπου όμοιων δυνατοτήτων με αυτό που χρησιμοποιείται στους υπολογιστές. Η χρήση τους αυξάνεται ραγδαία με την πάροδο των χρόνων, γεγονός που ήταν αδύνατο να παραληφθεί στην παρούσα εργασία. Επιπλέον, αποτελεί κύριο στόχο για το πρόγραμμα να δημιουργηθεί αυτή η εφαρμογή ώστε να καλυφθούν πλήρως οι απαιτήσεις των χρηστών, να παρέχονται περισσότερες δυνατότητες, μεγαλύτερη ασφάλεια, μεγαλύτερη ταχύτητα και σταθερότητα, και γενικότερα να υπάρξει μια πολύπλευρη προσέγγιση του προβλήματος ως προς το κομμάτι της υλοποίησης.

2.3.3 Γιατί προτιμήθηκε το iPhone – iOS

Η εφαρμογή, στη παρούσα εργασία, υλοποιήθηκε σε περιβάλλον λογισμικού iOS και σε πλατφόρμα κινητού τηλεφώνου iPhone. Αρκετοί λόγοι συνετέλεσαν στο γεγονός να προτιμηθεί το υλικό και το λογισμικό που έχει δημιουργήσει η Apple, σε αντίθεση με άλλα δημοφιλέστατα λειτουργικά συστήματα όπως τα Android και Windows Phone.

Αρχικά, κύριος λόγος είναι ότι το iPhone αποτελεί το πιο δημοφιλές κινητό τηλέφωνο στην παγκόσμια αγορά⁸, καθώς από το 2007 που έκανε την εμφάνισή του μέχρι σήμερα, μετράει τις περισσότερες πωλήσεις από κάθε άλλο κινητό τηλέφωνο. Σύμφωνα με έρευνες που γίνονται από αξιόπιστα πανεπιστήμια και εταιρείες δείχνουν

⁸ Περισσότερες από 516 εκατομμύρια συσκευές iPhone έχουν πωληθεί από το 2007 μέχρι τον Ιούνιο του 2014, σύμφωνα με τα επίσημα οικονομικά στοιχεία της Apple. [13]

ότι οι χρήστες συσκευών iPhone είναι πολύ ενθουσιασμένοι⁹ σε τόσο μεγάλο βαθμό που φτάνουν τα όρια του εθισμού [14]. Επίσης η ικανοποίησή τους από την χρήση της συσκευής είναι πολύ μεγάλη και αγγίζει το ποσοστό του 96%¹⁰ [16]. Παράλληλα αποδεικνύεται ότι οι χρήστες iPhone, κατ' επέκταση του λειτουργικού iOS, είναι περισσότερο διατεθειμένοι να πληρώσουν ώστε να εγκαταστήσουν εφαρμογές στη συσκευή τους, σε αντίθεση με τους χρήστες άλλων λειτουργικών όπως Android ή Windows. Το Apple App Store έχει το μεγαλύτερο μερίδιο σε πωλήσεις, φτάνοντας μέχρι και το 70% στην παγκόσμια αγορά. Αυτός ο παράγοντας λοιπόν, από μόνος του συνιστά σημαντικό γεγονός και μεγάλο κίνητρο για κάθε προγραμματιστή, διότι έχοντας τις γνώσεις και την δυνατότητα να δημιουργεί εφαρμογές σε αυτήν την πλατφόρμα λογισμικού, η αναγνώριση της δουλειάς του αλλά και τα οικονομικά οφέλη είναι μεγάλα.

Επιπροσθέτως, σημαντικό παράγοντα αποτελεί το γεγονός ότι για την υλοποίηση μιας εφαρμογής σε λειτουργικό iOS, απαιτείται καλό υπόβαθρο γνώσης της γλώσσας προγραμματισμού Objective-C. Η Objective-C, είναι μία από τις πιο διαδεδομένες γλώσσες προγραμματισμού παγκοσμίως, είναι πολύ απλή και εύχρηστη, εύκολη στην εκμάθηση, ενώ υπάρχει τεράστια ποικιλία οδηγιών εκμάθησής της στο διαδίκτυο, τους πληρέστερους και περισσότερους από τους οποίους προσφέρει η ίδια η Apple.

Επιπλέον, η πολιτική της Apple είναι πολύ αυστηρή. Συγκεκριμένα, μόνο εφαρμογές που προσφέρονται από το επίσημο ηλεκτρονικό κατάστημα, Apple App Store, επιτρέπεται να εγκατασταθούν στην συσκευή iPhone. Επίσης εφαρμογές που χρησιμοποιούν ιδιωτικά API's (Application Programming Interfaces), δηλαδή API's που έχουν δημιουργηθεί από μη εγγεγραμμένους προγραμματιστές ή από την ίδια την Apple, απορρίπτονται. Αυτό όμως σημαίνει ότι προσφέρει ασφάλεια στους χρήστες, καθώς το λειτουργικό και η συσκευή έχουν ελάχιστες πιθανότητες να προσβληθούν από διάφορα κακόβουλα λογισμικά. Παράλληλα, από την πλευρά του προγραμματιστή, το γεγονός ότι η εταιρεία Apple παρέχει ένα κλειστό λογισμικό-λειτουργικό, αποτελεί μεγάλη πρόκληση.

⁹ Πάνω από 2 εκατομμύρια προ-παραγγελίες μέσα σε 24 ώρες όταν ανακοινώθηκε το iPhone 5 [17]

¹⁰ Σύμφωνα με την Apple, οι χρήστες του iOS 7 είναι ευχαριστημένοι σε ποσοστό 97%. [15]

3. CLIENT SIDE -

Αρχιτεκτονική Συστήματος και Τεχνολογίες που χρησιμοποιήθηκαν στην εφαρμογή S.M.C

Σε αυτό το κεφάλαιο περιγράφονται όλες οι σημαντικές τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής **Sensor Monitor Controller (S.M.C.)**. Πρόκειται, λοιπόν για τεχνολογίες από την πλευρά του πελάτη (client), που στη συγκεκριμένη περίπτωση, πελάτη αποτελεί η εφαρμογή που υλοποιήθηκε σε πλατφόρμα λογισμικού **iOS**. Επομένως, αναφέρονται τα χαρακτηριστικά του λειτουργικού **iOS**, αλλά επίσης γίνεται εκτενής παρουσίαση της γλώσσας προγραμματισμού **Objective-C**, που βασίζεται το συγκεκριμένο λειτουργικό σύστημα και χρησιμοποιήθηκε κατά αποκλειστικότητα στην παρούσα εργασία. Επίσης γίνεται αναφορά στο μοντέλο **MVC** που στηρίζεται η αρχιτεκτονική της εφαρμογής αλλά και για το προγραμματιστικό εργαλείο **Xcode** που χρησιμοποιήθηκε και εγκαταστάθηκε ώστε να γραφεί ο κώδικας.

3.1 Λειτουργικό Σύστημα iOS

3.1.1 Γενικά

Δεν υπάρχει αμφιβολία ότι η διάθεση του iPhone στην αγορά το 2007, άλλαξε ολόκληρη την βιομηχανία της κινητής τηλεφωνίας. Περισσότερες από 500 εκατομμύρια συσκευές iPhone έχουν πωληθεί από εκείνη την ημερομηνία¹¹. Οποιοσδήποτε μπορεί εύκολα να παρατηρήσει ότι όλες οι εταιρείες κινητής τηλεφωνίας έχουν υιοθετήσει τον σχεδιασμό και την φιλοσοφία του iPhone. Όλα τα σύγχρονα smartphones διαθέτουν πλέον οθόνη αφής τύπου capacitive¹² και τεχνολογία πολλαπλής αφής (multitouch) που πρώτη η Apple παρουσίασε. Παράλληλα, όλα τα λειτουργικά συστήματα όπως, το Android, το Windows Phone OS, Symbian και άλλα, διαθέτουν πλέον τα δικά τους «App Stores», που πάλι η Apple παρουσίασε πρώτη. Το λειτουργικό σύστημα που χρησιμοποιείται από την Apple στην συσκευή του iPhone, iOS, θεωρείται από την συντριπτική πλειοψηφία ως ένα από τα καλύτερα λογισμικά κινητού τηλεφώνου, ή ακόμα και το πιο προηγμένο λειτουργικό σύστημα για φορητές

¹¹ Επίσης έχουν πωληθεί πάνω από 800 εκατ. συσκευές που χρησιμοποιούν το λειτουργικό iOS. [18]

¹² Η Capacitive οθόνης αφής λειτουργεί αντιλαμβανόμενη την ηλεκτρική αγωγιμότητα του δέρματος των δαχτύλων του χεριού, χωρίς να απαιτείται πίεση της οθόνης. [19]

συσκευές στον κόσμο. Επιπλέον, διαθέτει ένα εκπληκτικά κομψό και εύχρηστο περιβάλλον εργασίας, προσφέροντας πολλές δυνατότητες και σταθερότητα. Ο σχεδιασμός του είναι τέτοιος που το καθιστά ως το πιο εύχρηστο και φιλικό προς τον απλό χρήστη λειτουργικό στον κόσμο [20]. Τέλος αξίζει να αναφερθεί ότι το συγκεκριμένο λογισμικό χρησιμοποιείται από την Apple σε πληθώρα άλλων συσκευών, όπως iPad, iPod Touch που φαίνονται στην *Εικόνα 3.1*.



Εικόνα 3.1 –Οι τελευταίες εκδόσεις των συσκευών που χρησιμοποιούν λειτουργικό iOS.

3.1.2 Αρχιτεκτονική και Πακέτο Ανάπτυξης Λογισμικού

Η εταιρεία της Apple παρέχει στους προγραμματιστές ένα πακέτο ανάπτυξης λογισμικού (Software Development Kit, SDK) [21], ώστε να μπορούν να υλοποιήσουν εφαρμογές για το iPhone και όλες τις συσκευές που έχουν iOS και να τις διαθέσουν στο App Store [22]. Κάθε χρήστης κινητού τηλεφώνου iPhone έχει πρόσβαση σε αυτό το online κατάστημα εφαρμογών, δίνοντας έτσι ισχυρό κίνητρο σε κάθε προγραμματιστή να σκεφτεί σοβαρά την διαδικασία υλοποίησης μίας εφαρμογής. Το πακέτο αυτό προσφέρει στον προγραμματιστή όλες τις δυνατότητες και τα χαρακτηριστικά της συσκευής iPhone, ώστε να τα προγραμματίσει και να τα εκμεταλλευτεί στο μέγιστο, ενώ η γλώσσα προγραμματισμού που χρησιμοποιεί το λειτουργικό iOS είναι η Objective-C, που αναλύεται παρακάτω.

Στην *Εικόνα 3.2* φαίνονται τα βασικά επίπεδα που αποτελούν την αρχιτεκτονική του λειτουργικού iOS.

- Το πρώτο επίπεδο είναι το **Core OS**. Αποτελεί το κατώτερο ιεραρχικά επίπεδο καθώς περιλαμβάνει το υλικό και τις τεχνολογίες που είναι πιο κοντά στην φιλοσοφία του υλικού (hardware) της συσκευής. Ουσιαστικά αποτελεί ένα Unix (kernel) λειτουργικό σύστημα, το οποίο περιλαμβάνει API's γραμμένα σε γλώσσα C, δηλαδή μη αντικειμενοστραφή γλώσσα προγραμματισμού.
- Το δεύτερο επίπεδο είναι το **Core Services**. Αποτελεί το ανώτατο επίπεδο του Core OS και περιγράφει λειτουργίες αλλά σε αντικειμενοστραφή λογική.
- Το τρίτο επίπεδο είναι τα **Media**. Περιλαμβάνει λειτουργίες που αφορούν τα παιχνίδια, βίντεο, φωτογραφίες και μουσική, καθώς το iPhone αποτελεί μία κατά βάση συσκευή πολυμέσων.
- Τέλος, το ανώτερο ιεραρχικά επίπεδο είναι το **Cocoa Touch**. Αποτελεί τη σημαντικότερη κατηγορία, κυρίως για την εφαρμογή που υλοποιήθηκε στην παρούσα εργασία. Βασίζεται στη τεχνολογία Cocoa, αλλά προφανώς είναι προσαρμοσμένο και ενισχυμένο, όπως αναφέρεται και στην ονομασία Touch (Αφή), με λειτουργίες που αφορούν την συσκευή του iPhone. Το Cocoa αποτελεί ουσιαστικά ένα Application Programming Interface (API) για το λειτουργικό σύστημα Mac OS X με αντικειμενοστραφή λογική και είναι κατά βάση γραμμένο σε γλώσσα Objective-C. Πρόκειται για μία τεχνολογία που υπάρχει 30 χρόνια, από τις αρχές της δεκαετίας του 1980 όπου έκανε την εμφάνισή του το λειτουργικό Mac OS. Αξίζει να σημειωθεί ότι για την διαχείριση της μνήμης χρησιμοποιείται η τεχνική του automatic referencing counting (ARC) [24]. Επίσης, το Cocoa Touch υποστηρίζει και προσφέρει πληθώρα frameworks για τον έλεγχο και προγραμματισμό λειτουργιών της συσκευής, όπως για παράδειγμα το multitasking (ταυτόχρονη εκτέλεση διεργασιών) ή το gesture recognizer (αναγνώριση κινήσεων δαχτύλων στην οθόνη αφής). Περιλαμβάνει όλα τα γραφικά που εμφανίζονται στην οθόνη της συσκευής, δηλαδή ουσιαστικά αποτελεί ένα User Interface (UI) framework για το λειτουργικό του iOS. Τέλος, το Cocoa Touch ακολουθεί το Model-View-Controller (MVC) πρότυπο αρχιτεκτονικής λογισμικού που περιγράφεται παρακάτω.



Εικόνα 3.2 –Απεικόνιση επιπέδων της αρχιτεκτονικής του λειτουργικού iOS.

3.1.3 Περιορισμοί ανάπτυξης λογισμικού iOS

Η Apple στην προσπάθεια να αποθαρρύνει την πειρατεία και την παρανομία στην τεχνολογία που παράγει, δυστυχώς έχει εφαρμόσει πολλούς περιορισμούς ως προς το λογισμικό, iOS, αλλά κυρίως ως προς την συσκευή iPhone. Συγκεκριμένα, το λειτουργικό σύστημα iOS δεν επιτρέπει την χρήση ιδιωτικών βιβλιοθηκών και προγραμμάτων που προσπαθούν να αποκτήσουν πρόσβαση στο υλικό (hardware) και γενικότερα στα χαρακτηριστικά της συσκευής του iPhone. Για παράδειγμα ότι αφορά την παροχή πληροφοριών σχετικά με το σήμα και το δίκτυο που συνδέεται κάθε συσκευή, το iOS τα απορρίπτει και τα διατηρεί κλειδωμένα. Επίσης η πρόσβαση στο σύστημα διαχείρισης αρχείων του iOS δεν επιτρέπεται.

Επιπλέον δεν μπορεί ο καθένας να εγκαταστήσει εφαρμογές στο iPhone, παρά μόνο εάν συμμετέχει στο πρόγραμμα προγραμματιστών πληρώνοντας το ποσό των 99 δολαρίων. Η Apple δεν θέλει ο οποιοσδήποτε να έχει πλήρη πρόσβαση στο λογισμικό και στην συσκευή του iPhone¹³, θεωρώντας επίσης ότι παραβιάζεται και η «εμπειρία του χρήστη» που έχει δημιουργήσει.

¹³ Η Apple λαμβάνει το 30% του κέρδους από κάθε εφαρμογή που πωλείται στο App Store.

Η ανάπτυξη και υλοποίηση εφαρμογών στο iPhone είναι διαθέσιμη μέσω της πλατφόρμας ανάπτυξης λογισμικού Xcode. Το συγκεκριμένο εργαλείο λειτουργεί μόνο σε λειτουργικό σύστημα Mac OS X για υπολογιστές Macintosh. Οι υπολογιστές αυτοί έχουν περίπου ίδια αρχιτεκτονική με αυτή των υπολοίπων PC's, αλλά δυστυχώς η εγκατάσταση των Mac OS X σε οποιοδήποτε υπολογιστή είναι πολύ δύσκολη. Αυτό οφείλεται στο γεγονός ότι το λειτουργικό σύστημα ελέγχει πολλές παραμέτρους του υλικού (hardware) ώστε να επιβεβαιώσει ότι ο παρών υπολογιστής είναι αυθεντικός υπολογιστής Macintosh. Επομένως για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε υπολογιστής iMac 24-inch (Early 2009).

3.2 Objective-C

3.2.1 Εισαγωγή στην Objective-C

Το λειτουργικό σύστημα iOS χρησιμοποιεί ως βασική γλώσσα προγραμματισμού την **Objective-C** [23]. Όπως δηλώνει και το όνομά της, πρόκειται για μια αντικειμενοστραφή (object-oriented) γλώσσα προγραμματισμού βασισμένη στην C, μία από τις πιο διαδεδομένες και παλαιότερες υψηλού επιπέδου γλώσσες προγραμματισμού [25].

Η Objective-C οφείλει την παρουσία της στους δύο ιδιοκτήτες της εταιρείας Stepstone, Brad Cox και Tom Love. Οι βασικοί δημιουργοί της ήταν αυτοί οι δύο προγραμματιστές και την πρώτη της εμφάνιση έκανε στις αρχές της δεκαετίας του 1980. Η γλώσσα έκτοτε δεν ήταν πολύ δημοφιλής έναντι των άλλων δημοφιλέστερων αντικειμενοστραφών γλωσσών όπως Java και C++. Η επιλογή της Objective-C από την Apple ώστε να αποτελέσει τη βάση για το Mac OS X και το iOS αργότερα δεν ήταν τυχαία, καθώς είχε επιλεγεί το 1988 για το λειτουργικό σύστημα NeXTSTEP OS που δημιούργησε ο Steve Jobs μέσω της NeXT Inc. Μέχρι το 2008, όπου παρουσιάστηκε το επίσημο ηλεκτρονικό κατάστημα εφαρμογών, App store της Apple από τον Steve Jobs, η γλώσσα ήταν πολύ χαμηλά στην λίστα της παγκόσμιας κοινότητας προγραμματισμού TIOBE. Συγκεκριμένα, παρατηρείται ότι μέχρι και την στιγμή που γράφεται η παρούσα εργασία, παραμένει στην τρίτη θέση της σχετικής λίστας, κάτω από τις Java και C. Η άνοδος της δημοτικότητάς της, εκτός των άλλων, οφείλεται και στην ευκολία εκμάθησής της μέσα από τα πολύ οργανωμένα σεμινάρια που προσφέρονται από διάφορα γνωσμένα πανεπιστήμια δωρεάν, όπως αυτό του Στάνφορντ. Επιπλέον, το γεγονός της δωρεάν διάθεσης του προγραμματιστικού εργαλείου Xcode όπου είναι απαραίτητο για την υλοποίηση εφαρμογών σε λειτουργικό iOS, βοήθησε αρκετά στην διάδοση της γλώσσας.

3.2.2 Χαρακτηριστικά Objective-C

Η Objective-C ουσιαστικά είναι μία αντικειμενοστραφής C, δηλαδή βασίζεται σε έννοιες όπως αυτές της κλάσης (class), του στιγμιότυπου (instance) και προφανώς του αντικειμένου (object). Επιπλέον, εξαιτίας του γεγονότος ότι έχει ως βάση της την C, σε επίπεδο συντακτικού έχει πολλές ομοιότητες αλλά πολλές περισσότερες δυνατότητες δυναμικού χαρακτήρα. Ένα απλό παράδειγμα που επιβεβαιώνει την εξάρτηση από την C, είναι ότι η δέσμευση-απελευθέρωση μνήμης γίνεται, τουλάχιστον στις πρώτες εκδόσεις των iOS SDK's (Software Development Kit), χειροκίνητα από τον προγραμματιστή.

Η Objective-C, λοιπόν, έχει δώσει λύσεις στον προγραμματισμό εφαρμογών για κινητές συσκευές όπως τα κινητά τηλέφωνα και τα tablets. Η πληθώρα των API's και των βιβλιοθηκών που διαθέτει, επιτρέπουν την πλήρη αξιοποίηση του υλικού των συσκευών και συντελούν στην αρμονική συνύπαρξη λογισμικού (software) και υλικού (hardware). Η εφαρμογή που υλοποιήθηκε στην παρούσα εργασία έχει στηριχθεί πάνω στην Objective-C. Συγκεκριμένα, χρησιμοποιείται εξολοκλήρου για την σχεδίαση, ανάπτυξη της εφαρμογής **S.M.C.**, την σύνδεση με την βάση δεδομένων (web service) και επικοινωνία με το υλικό του iPhone.

Κύρια χαρακτηριστικά της Objective-C [26] επιγραμματικά είναι:

- Η Objective-C προσφέρει βιβλιοθήκες για μεγάλο πλήθος αναγκών, ως μέρος του Objective-C platform.
- Η γλώσσα έχει σχεδιαστεί ώστε να προσφέρει ευκολία και μεγαλύτερη παραγωγικότητα στον προγραμματιστή
- Η Objective-C είναι γενικά εύκολη στην χρήση και την εκμάθηση, έχοντας διατηρήσει τα καλά στοιχεία άλλων αντικειμενοστραφών γλωσσών.
- Η γλώσσα είναι πολύ απλή και παράλληλα πολύ ισχυρή. Αυτό οφείλεται στο γεγονός ότι κάθε κλάση αποτελεί ένα αντικείμενο. Έτσι, η δημιουργία ενός αντικειμένου γίνεται με απλή αποστολή ενός μηνύματος στην αντίστοιχη κλάση, χωρίς χρήση κάποιου ιδιαίτερου μηχανισμού κονστράκτορα (constructor).
- Η γλώσσα έχει την δυνατότητα να υποστηρίζει τον δυναμικό αλλά και τον στατικό ορισμό ενός αντικειμένου. Επίσης εισάγει την έννοια «κατηγορίες» (categories), όπου δίνει την δυνατότητα να προστεθούν νέες μέθοδοι σε κλάσεις χωρίς να υπάρχει γνώση του πηγαίου κώδικά τους.
- Εισάγει την έννοια της ενδοσκόπησης (introspection). Η ενδοσκόπηση είναι πάρα πολύ χρήσιμη βοηθώντας τον προγραμματιστή να γνωρίζει πολύ εύκολα κάθε αντικείμενο σε ποια κλάση ανήκει αλλά και αν διαθέτει κάποιες μεθόδους.

- Διαθέτει ένα συντακτικό πολύ εύκολο, απλό και εύχρηστο. Υπάρχει αλληλεπίδραση με τα αντικείμενα μέσω αποστολής απλών μηνυμάτων. Τα μηνύματα αυτά κάθε αντικείμενο τα αντιλαμβάνεται άμεσα και είτε τα εκτελεί το ίδιο, είτε τα προωθεί σε άλλα αντικείμενα είτε τα διαδίδει σε πολλά άλλα αντικείμενα. Η μορφή των μηνυμάτων είναι όπως μια απλή πρόταση στη φυσική γλώσσα. Ένα απλό παράδειγμα όπου θέλουμε να εισάγουμε ένα αντικείμενο στην θέση 10 του πίνακα γράφεται εξής: `[myArray insert:myObject atIndex:10];`
- Η Objective-C αποτελεί ένα υπερσύνολο της C, με αποτέλεσμα να μπορεί να εκμεταλλεύεται όλες τις δυνατότητές της και επιπλέον να παρέχει πρόσβαση σε βιβλιοθήκες που χρησιμοποιεί η C.

3.3 Μοντέλο σχεδιασμού MVC (Model View Controller)

Η υλοποίηση της εφαρμογής της παρούσας εργασίας βασίστηκε στο μοντέλο ανάπτυξης - σχεδιασμού **MVC (Model-View-Controller)** [27]. Το συγκεκριμένο μοντέλο αποτελεί ένα δημοφιλές και παγκοσμίως διαδεδομένο μοτίβο σχεδιασμού λογισμικού, καθώς διαθέτει τρομερά πλεονεκτήματα και διασφαλίζει την δυνατότητα επαναχρησιμοποίησης και αναβάθμισης κάποιων μερών ή και ολόκληρου του κώδικα.

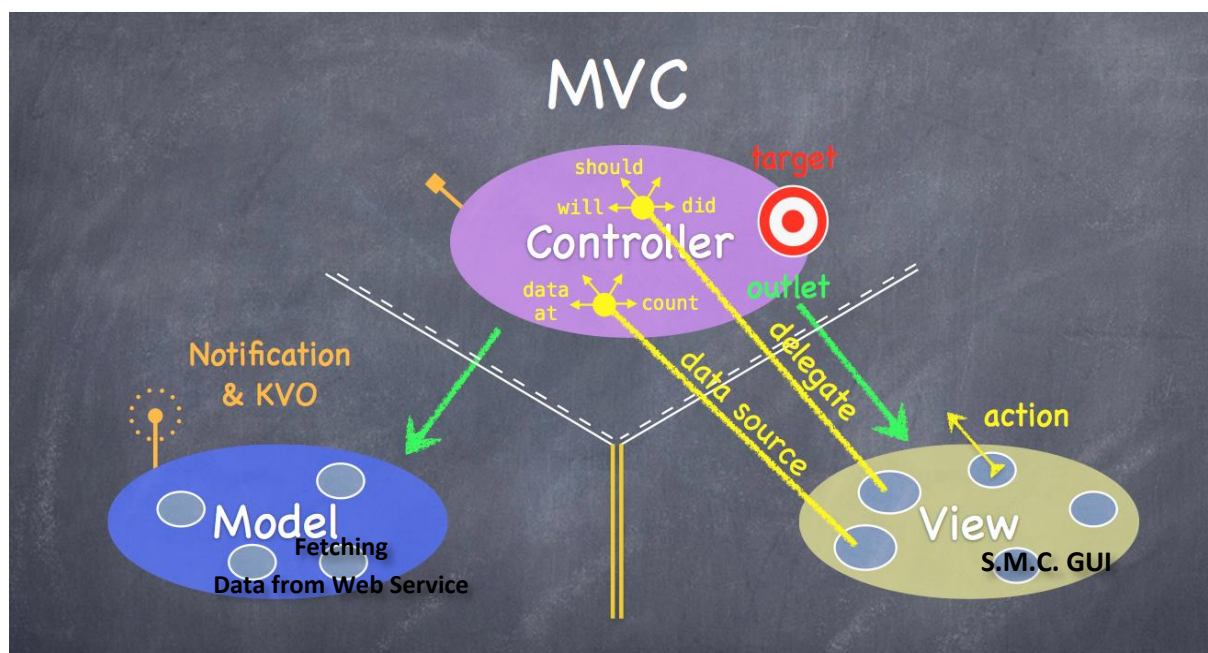
Το μοντέλο διαχωρίζει την εφαρμογή (γενικά τα αντικείμενα) σε τρία επίπεδα. Ουσιαστικά επιτρέπει τον διαχωρισμό της λειτουργικότητας από το περιβάλλον εργασίας χρήστη (user interface) της εφαρμογής αλλά και την χρήση πολλαπλών interfaces για το ίδιο μοντέλο. Τα τρία αυτά επίπεδα είναι ανεξάρτητα και αυτόνομα με αποτέλεσμα να δίνεται η δυνατότητα αλλαγής μελλοντικά κάποιου από τα τρία επίπεδα, για παράδειγμα αλλαγή του interface, χωρίς όμως να επηρεάζονται τα υπόλοιπα. Αναλυτικότερα τα τρία επίπεδα είναι τα εξής:

- **Model (Μοντέλο):** Περιγράφει την εφαρμογή αυτή καθαυτή αλλά χωρίς την περιγραφή του περιβάλλοντος εργασίας (interface), των γραφικών της. Συγκεκριμένα, αποτελεί τα δεδομένα (data) της εφαρμογής αλλά και τους κανόνες που διέπουν την πρόσβαση σε αυτά. Τα δεδομένα αυτά συνήθως βρίσκονται αποθηκευμένα σε κάποια βάση δεδομένων (στο iPhone CoreData, SQLite). Ένα καλό μοντέλο (Model) μπορεί να απεικονιστεί με διαφορετικούς τρόπους.
- **View (Απεικόνιση):** Αποτελεί τα γραφικά, το περιβάλλον εργασίας χρήστη (interface - GUI). Περιγράφει αυτό ακριβώς που βλέπει ο χρήστης. Ουσιαστικά αποτελεί τη γραφική αναπαράσταση των στοιχείων του Model (δεδομένων).

Μία καλή Απεικόνιση (View) μπορεί να προσαρμόζεται εύκολα, γρήγορα και κατάλληλα σε οποιοδήποτε μοντέλο αλλά και ενδεχόμενη αλλαγή του.

- Controller (Ελεγκτής):** Περιγράφει τον τρόπο με τον οποίο αναπαριστώνται στον χρήστη τα δεδομένα, δηλαδή το μοντέλο (Model). Ουσιαστικά αποτελεί την διασύνδεση μεταξύ των δύο προηγούμενων επιπέδων, Model και View. Ο ελεγκτής διαχειρίζεται οποιαδήποτε αλλαγή υπάρχει στα άλλα δύο επίπεδα και ευθύνεται για την ενημέρωσή τους, όπως επίσης διαχειρίζεται οποιαδήποτε λάθη προκύπτουν. Επομένως, αν υπάρξει αλλαγή στο Model τότε ενημερώνει το View και αντίστροφα. Αποτελεί το βασικότερο κομμάτι της εφαρμογής καθώς αποτελεί το μεγαλύτερο μέρος προγραμματισμού σε κώδικα.

Επιπλέον, όπως παρουσιάζεται και στην *Εικόνα 3.3*, αξίζει να σημειωθεί ότι ο ελεγκτής έχει την δυνατότητα να «μιλάει» άμεσα με τα άλλα δύο επίπεδα χρησιμοποιώντας KVO (Key Value Observing) μεταβλητές, αντιθέτως τα επίπεδα Model και View, δεν έχουν την δυνατότητα να επικοινωνούν μεταξύ τους άμεσα παρά μόνο μέσω του Controller. Επιτυγχάνεται με αυτόν τον τρόπο, λοιπόν η πλήρης επαναχρησιμοποίηση του κώδικα μελλοντικά, ώστε πιθανή αλλαγή για παράδειγμα, ολόκληρου του μοντέλου (Model) να μην επηρεάζει καθόλου την απεικόνιση (View).



Εικόνα 3.3 – Διαγραμματική απεικόνιση του μοντέλου MVC.

3.4 Xcode



Εικόνα 3.4 – Λογότυπο Xcode

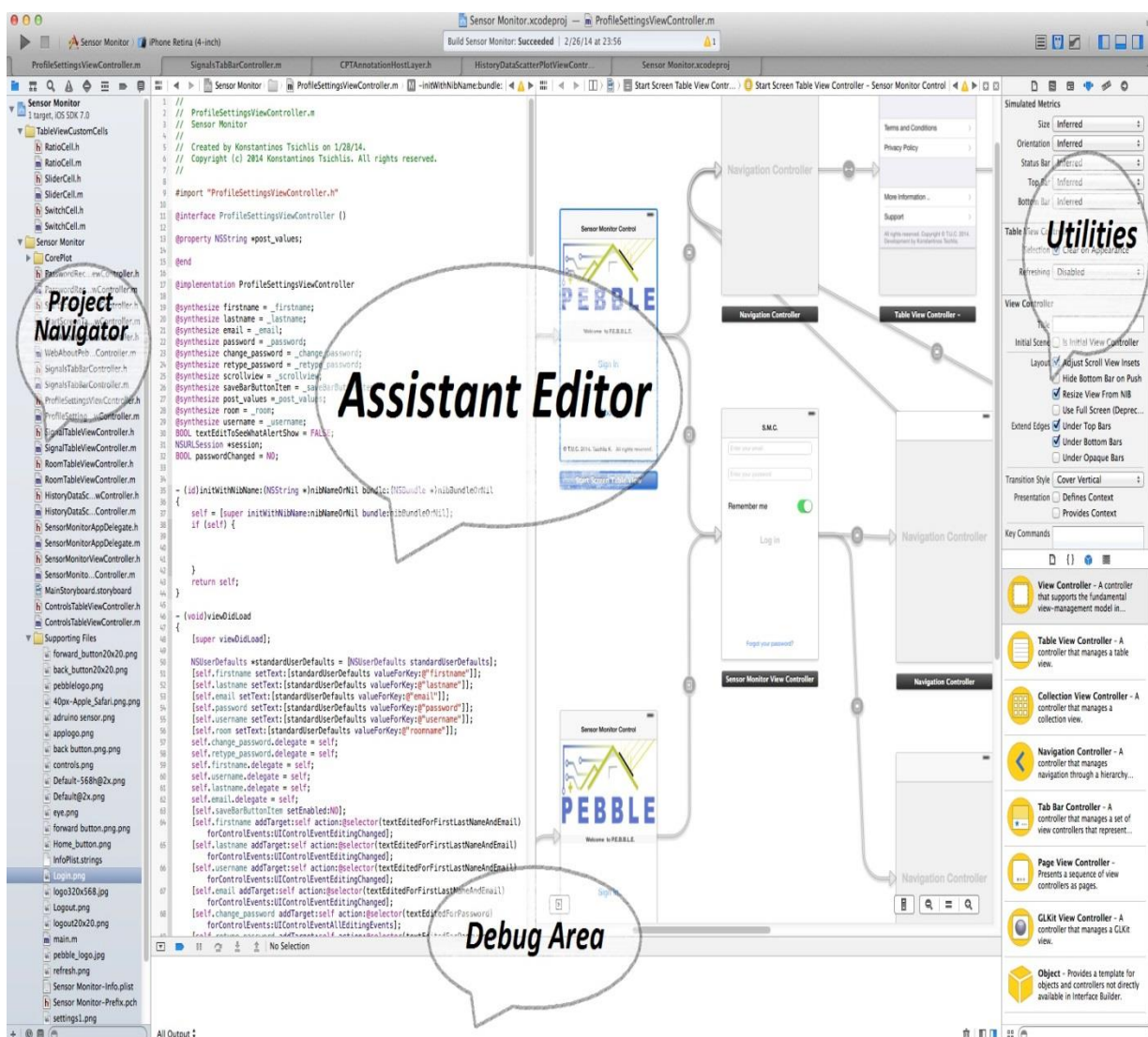
Η εφαρμογή υλοποιήθηκε, όπως αναφέρθηκε νωρίτερα, στο περιβάλλον **Xcode** [28]. Το **Xcode** αποτελεί μια ανοιχτή πλατφόρμα ανάπτυξης λογισμικού σε γλώσσες Objective-C, C, C++, Java, AppleScript, Python, Ruby, Objective-C++ και άλλες. Ουσιαστικά αποτελεί ένα ισχυρό και ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment – IDE) της Apple, για τη δημιουργία εφαρμογών για υπολογιστές Macintosh, κινητά iPhone και ταμπλέτες iPad. Το interface του Xcode είναι άρτια σχεδιασμένο προσφέροντας τον Assistant Editor με την διπλή οθόνη (για προβολή παραπάνω του ενός αρχείων) για την επεξεργασία του κώδικα, τον Interface Builder για τον σχεδιασμό των γραφικών της εφαρμογής (User Interface) χωρίς χρήση κώδικα, τον Debugger για τον έλεγχο και τον εντοπισμό σφαλμάτων, παραπομπή στις βιβλιοθήκες με ένα απλό πάτημα και όλα αυτά μέσα σε ένα παράθυρο. Διαθέτει επίσης τον LLVM μεταγλωττιστή (compiler) της Apple, ο οποίος είναι πάρα πολύ έξυπνος και διαδραστικός καθώς υπογραμμίζει τα λάθη του κώδικα αυτόματα κατά την διάρκεια της πληκτρολόγησης.

Προσφέρει επίσης πολλά εργαλεία που είναι πολύ εύκολα στη χρήση και καλύπτουν όλα τα στάδια υλοποίησης μιας εφαρμογής, δηλαδή από τον προγραμματισμό, την εκσφαλμάτωση και τη μεταγλώττιση μέχρι την προσομοίωση αλλά και εφαρμογή του λογισμικού που παράχθηκε σε πραγματικές συνθήκες. Τέτοια εργαλεία είναι ο προσομοιωτής (iOS simulator), που προσομοιώνει ολόκληρο τον κώδικα σε πραγματικές συνθήκες και το εργαλείο instruments, το οποίο βοηθάει στην μελέτη, ανάλυση και παρακολούθηση της συμπεριφοράς του κώδικα της εφαρμογής και κυρίως την απόδοσή της. Συγκεκριμένα, έχει την δυνατότητα να εντοπίσει και να εξετάσει την συμπεριφορά μίας ή περισσότερων διεργασιών, να αναλύσει την συμπεριφορά του επεξεργαστή και της μνήμης της συσκευής που «τρέχει» η εφαρμογή, δηλαδή τους πόρους που καταναλώνει κατά τη διάρκεια που βρίσκεται στο προσκήνιο/παρασκήνιο (background), όπως για παράδειγμα να εντοπίσει πιθανές απώλειες μνήμης. Άλλα εργαλεία είναι ο προσομοιωτής εκτύπωσης (printer simulator), ο φορτωτής εφαρμογών (application loader) που συμβάλλει στην προσθήκη της εφαρμογής στο επίσημο

ηλεκτρονικό κατάστημα εφαρμογών (app store) της Apple και ο επιθεωρητής προσβασιμότητας (accessibility inspector).

Στη παρούσα εργασία έγινε χρήση της έκδοσης Xcode 5.0.1 που περιλαμβάνει και τις τελευταίες εκδόσεις (SDK's) λογισμικού iOS 7 αλλά και τον μεταγλωττιστή LLVM 5.0. Το συγκεκριμένο εργαλείο διατίθεται εντελώς δωρεάν από την εταιρεία της Apple, η οποία παρέχει κάθε είδους υποστήριξη, διαθέτοντας επιπλέον εργαλεία, οδηγούς χρήσης (tutorials) και βίντεο, δωρεάν αναβαθμίσεις-διορθώσεις κάθε έκδοσης αλλά και επιπλέον βιβλιοθήκες (API's). Στην παρακάτω εικόνα (3.5) φαίνεται μία απεικόνιση του περιβάλλοντος του Xcode.

Τέλος να αξίζει να αναφερθεί ότι το εργαλείο Xcode εγκαταστάθηκε και «τρέχει» σε πλατφόρμα λογισμικού Mac OS X 10.9.2.



Εικόνα 3.5 – Γραφική απεικόνιση του περιβάλλοντος (interface) του εργαλείου Xcode

3.5 Εφαρμογή Sensor Monitor Controller

3.5.1 Προϋποθέσεις και Υπόβαθρο

Στα πλαίσια του προγράμματος σπουδών του τμήματος Ηλεκτρονικών Μηχανικών και Μηχανικών Υπολογιστών, ΗΜΜΥ, στο Πολυτεχνείο Κρήτης, δυστυχώς δεν περιλαμβάνεται κάποιο μάθημα σχετικό με την ανάπτυξη εφαρμογών σε iOS, αλλά ούτε σε κάποιο άλλο πανεπιστήμιο στην Ελλάδα. Όμως, στο πανεπιστήμιο του Stanford διδάσκεται ένα μάθημα και διατίθεται δωρεάν σε όλους στο διαδίκτυο, μέσω του προγράμματος iTunes της Apple και συγκεκριμένα μέσω της ενότητας iTunes U¹⁴ [29]. Το συγκεκριμένο μάθημα προσφέρει μια καλή και λεπτομερή εισαγωγή στον κόσμο του προγραμματισμού σε iPhone. Επίσης αναλύει όλες τις βασικές αρχές και τα προγραμματιστικά μοτίβα του λειτουργικού iOS, όπως το MVC που περιγράφηκε σε προηγούμενη παράγραφο [3.3], την γλώσσα προγραμματισμού Objective-C [3.2]. Επιπλέον οι διαλέξεις των μαθημάτων διατίθενται σε μορφή αρχείων pdf αλλά και βίντεο, ενώ παρουσιάζονται πληθώρα παραδειγμάτων που περιγράφονται λεπτομερώς κατά την διάρκεια αυτών. Προφανώς, η παρακολούθηση του συγκεκριμένου μαθήματος ήταν επιβεβλημένη και αποτέλεσε την βάση για την υλοποίηση της εφαρμογής **S.M.C.**, καθώς βοήθησε στην εκμάθηση της γλώσσας προγραμματισμού Objective-C αλλά και στην εξοικείωση με τον προγραμματισμό σε iPhone. Τέλος, αξίζει αναφοράς ότι σημαντική βοήθεια προσέφεραν πληθώρα οδηγιών και παραδειγμάτων υλοποίησης εφαρμογών σε iOS, που βρίσκονται σε όλο το διαδίκτυο [30].

3.5.2 Εγκατάσταση εφαρμογής σε συσκευή iPhone

Από την στιγμή που η συγκεκριμένη εφαρμογή υλοποιήθηκε σε λειτουργικό iOS είναι προφανές ότι μπορεί να εγκατασταθεί μόνο σε κινητό iPhone. Όπως έχει ήδη αναφερθεί σε προηγούμενη παράγραφο [3.1.3], η εγκατάσταση μίας εφαρμογής σε οποιαδήποτε συσκευή iPhone που δεν έχει υλοποιηθεί από ένα επίσημο μέλος – προγραμματιστή της Apple δεν επιτρέπεται. Απαιτείται λοιπόν ένα επίσημο πιστοποιητικό, υπογεγραμμένο από την Apple, ώστε να επιτραπεί η εγκατάσταση και ο έλεγχος οποιασδήποτε εφαρμογής σε οποιαδήποτε συσκευή iPhone. Σε διαφορετική περίπτωση ο μόνος τρόπος να ελεγχτεί η λειτουργία μίας εφαρμογής είναι ο προσομοιωτής, iPhone-iPad Simulator, που προσφέρει το Xcode. Όμως ο προσομοιωτής iOS έχει το

¹⁴ Πρόκειται για το μάθημα CS 193P – iPhone Application Development.

σημαντικό μειονέκτημα ότι δεν υποστηρίζει κάποια χαρακτηριστικά που διαθέτει μία πραγματική συσκευή iPhone, όπως το GPS ή την σύνδεση σε δίκτυο κινητής τηλεφωνίας 3G.

Η λύση στο συγκεκριμένο πρόβλημα είναι η διαδικασία του Jailbreak. Αν λοιπόν δεν μπορεί κάποιος να γίνει συνδρομητής-προγραμματιστής της Apple, το πρώτο βήμα που πρέπει να κάνει, είναι jailbreak. Πρόκειται για μία νόμιμη διαδικασία που όμως η Apple δεν αποδέχεται, αφαιρώντας έτσι την εγγύηση αν εντοπιστεί κάποιο ίχνος jailbreak [31]. Συγκεκριμένα, ο όρος αυτός αναφέρεται στην εγκατάσταση ενός προγράμματος που αποκλείει-αναιρεί τους περιορισμούς της Apple και παρέχει πλήρη πρόσβαση στο σύστημα αρχείων της συσκευής, χωρίς όμως να επηρεάζει ή να προκαλεί κάποια ζημιά στο λογισμικό ή την συσκευή. Έπειτα απαιτείται μια διαδικασία προσθήκης και τροποποίησης συγκεκριμένων ρυθμίσεων της εφαρμογής στο προγραμματιστικό εργαλείο Xcode. Παράλληλα, χρειάζεται η δημιουργία ενός ψεύτικου πιστοποιητικού και ενός ψεύτικου προφίλ προγραμματιστή (99\$/χρόνο), ώστε να αναγνωριστεί από το περιβάλλον εργασίας του iOS ως αυθεντικό και να επιτραπεί η εγκατάσταση της εφαρμογής σε πραγματική συσκευή iPhone [32]. Έτσι είναι εφικτό να εκτελεστεί η εφαρμογή στην συσκευή και να γίνει αποσφαλμάτωση (debugging) στον κώδικά της.

Ακολουθώντας αυτά τα βήματα επιτεύχθηκε η εγκατάσταση της εφαρμογής σε συσκευή iPhone 4 με έκδοση λογισμικού iOS 7.0.4, γεγονός που βοήθησε πάρα πολύ στον εντοπισμό οποιονδήποτε σφαλμάτων (debugging) αλλά κυρίως ήταν ένας δημιουργικός τρόπος εύρεσης και προσθήκης των περισσότερων λειτουργιών που υποστηρίζει η εφαρμογή. Η εφαρμογή έχει υλοποιηθεί για τις εκδόσεις λογισμικού iOS 6 και 7 και για όλες τις συσκευές της Apple, iPhone 3GS,4,4s,5c,5, iPod touch και iPad.



Εικόνα 3.6 - Λογότυπο Εφαρμογής S.M.C.

3.5.3 Βιβλιοθήκες που χρησιμοποιήθηκαν

Αξίζει σε αυτό το σημείο να γίνει σύντομη αναφορά στις βιβλιοθήκες και τα πακέτα προγραμματιστικού κώδικα (frameworks) που απαιτήθηκαν για την υλοποίηση κάποιων λειτουργιών που προσφέρει η εφαρμογή **S.M.C.** Συγκεκριμένα, η εφαρμογή **S.M.C.** προσφέρει στον χρήστη την δυνατότητα να παρακολουθήσει την διακύμανση των τιμών κάθε αισθητήρα μέσω απεικόνισης διαγραμμάτων συναρτήσει του χρόνου [6.4.6]. Το πακέτο ανάπτυξης λογισμικού, iOS SDK, προσφέρει αυτήν την δυνατότητα μέσω του Quartz Core framework [33]. Η υλοποίηση όμως του διαγράμματος μέσω του συγκεκριμένου framework είναι πολύ πολύπλοκη με αποτέλεσμα να αναζητηθούν άλλα frameworks που διατίθενται από προγραμματιστές.

Υπάρχουν αρκετές επιλογές εκ των οποίων κάποιες διατίθενται δωρεάν και κάποιες προσφέρονται έναντι κάποιου ποσού. Στην προσπάθεια επιλογής της καλύτερης δυνατής λύσης, δοκιμάστηκαν αρκετές βιβλιοθήκες [34]. Συγκεκριμένα πρόκειται για τις:

- “BEMSimpleLineGraph”
- “iOSPlot”
- “JBChartView”
- “Core Plot”

που διατίθενται δωρεάν και τις

- “ShinobiCharts”
- “NChart3D”

που είναι διαθέσιμες μέσω δοκιμαστικών εκδόσεων [35].

Μετά την δοκιμή τους αποδείχτηκε πληρέστερη και καλύτερη λύση, αυτή της “ShinobiCharts” [36]. Η συγκεκριμένη βιβλιοθήκη προσφέρει πληθώρα επιλογών, πολύ όμορφα γραφικά, πλήρες και απλό εγχειρίδιο χρήσης. Μοναδικό αλλά ίσως σημαντικό μειονέκτημα αποτελεί το γεγονός ότι η διάθεσή της γίνεται μόνο μέσω πληρωμής ετήσιας συνδρομής. Στην παρούσα εφαρμογή όμως χρησιμοποιήθηκε η δοκιμαστική έκδοση που έχει μηνιαία ισχύ με δυνατότητα ανανέωσης μετά την λήξη. Αναφέρεται ως «σημαντικό μειονέκτημα», καθώς η δοκιμαστική έκδοση λειτουργεί άρτια στον προσομοιωτή iOS του Xcode, αλλά δεν έχει την δυνατότητα λειτουργίας σε πραγματική συσκευή iPhone. Η δυνατότητα πλήρους λειτουργίας προσφέρεται μέσω της πληρωμής ετήσιας συνδρομής αξίας 400 δολαρίων, κάτι που είναι εφικτό οποιαδήποτε στιγμή μελλοντικά. Τέλος, αξίζει να αναφερθεί ότι η επιλογή της βιβλιοθήκης “Core Plot” [38] επικράτησε αρχικά, αλλά το γεγονός ότι δεν προσέφερε τις ίδιες δυνατότητες, είχε αποτέλεσμα την απόρριψή της. Για παράδειγμα, η

υλοποίηση άξονα χρόνου δεν είναι εφικτή, ενώ η ποιότητα των γραφικών είναι πολύ κατώτερη από αυτή που προσφέρει η βιβλιοθήκη της **ShinobiCharts**.

Επιπλέον, δοκιμάστηκε μία βιβλιοθήκη σχεδιασμού και ανάπτυξης γραφημάτων που ονομάζεται “**JsGraph**” [37]. Η συγκεκριμένη βιβλιοθήκη έχει αντικειμενοστραφή λογική και χρησιμοποιεί την γλώσσα προγραμματισμού PHP. Επομένως, η υλοποίησή της γίνεται πάντα στο επίπεδο του εξυπηρετητή (server). Τα κυριότερα χαρακτηριστικά και οι βασικές δυνατότητες που παρέχει και απαιτούνται για την υλοποίηση της παρούσας εφαρμογής, διατίθενται δωρεάν. Όμως, υπάρχει και η πλήρης έκδοση που απαιτεί πληρωμή για την απόκτηση άδειας χρήσης. Η συγκεκριμένη βιβλιοθήκη, εξάγει διαγράμματα με μεγαλύτερη ταχύτητα, ωραία γραφικά αλλά ούτε αυτή προσφέρει κάποιες σημαντικές δυνατότητες, όπως για παράδειγμα αυτήν της μεγέθυνσης ή σμίκρυνσης (ζουμ) της γραφικής παράστασης. Όμως, χρησιμοποιήθηκε και έγινε μία επιπλέον υλοποίηση εξαγωγής διαγραμμάτων για κάθε σήμα, ώστε να υπάρχει η δυνατότητα προβολής τους σε πραγματική συσκευή iPhone. Η “**JsGraph**”, λοιπόν, αποτελεί μία εναλλακτική λύση ώστε να υπάρχει αυτή η δυνατότητα, καθώς ενδέχεται να μην είναι εφικτή η επιλογή της βιβλιοθήκης “**ShinobiCharts**” μελλοντικά, εξαιτίας του κόστους πληρωμής για απόκτηση άδειας χρήσης. Η υλοποίηση έγινε στην μεριά του εξυπηρετητή και έπειτα ενσωματώθηκε στην υλοποίηση της εφαρμογής ως διαφορετική έκδοση.

Επίσης έγινε χρήση κάποιων βιβλιοθηκών που αφορούν το user interface της εφαρμογής **S.M.C.** και ιδιαίτερα την προσπάθεια καλύτερης σχεδίασης του γραφικού περιβάλλοντος. Συγκεκριμένα, πρόκειται για τις βιβλιοθήκες

- “**MBProgressHUD**” [39]
- “**AHKActionSheet**” [40]

Η μεν πρώτη αφορά την εμφάνιση κατάλληλων μηνυμάτων κατά την διάρκεια εκτέλεσης διαφόρων διεργασιών από την εφαρμογή, όπως για παράδειγμα κατά την διάρκεια σύνδεσης του χρήστη στο σύστημα. Ως προς την δεύτερη, εφαρμόστηκε για την αρτιότερη εμφάνιση των μενού που προσφέρουν διάφορες ενέργειες και λειτουργίες στον χρήστη. Παραδείγματα χρήσης τους φαίνονται στο 6^ο κεφάλαιο [6.4].

4. SERVER SIDE -

Αρχιτεκτονική Συστήματος και Τεχνολογίες που χρησιμοποιήθηκαν στον εξυπηρετητή

Στο προηγούμενο κεφάλαιο έγινε εκτενής αναφορά στις τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν στην υλοποίηση της ίδιας της εφαρμογής (client-side). Απαιτείται όμως η υλοποίηση ενός εξυπηρετητή που αντλεί όλα τα δεδομένα των αισθητήρων, των ενεργοποιητών (actuators), των τιμών τους, καθώς και της γενικότερης δομής του κτιρίου Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης. Αυτή η ενότητα, λοιπόν, πραγματεύεται την ανάλυση της δομής, των τεχνολογιών και των αρχών λειτουργίας του συστήματος που υλοποιήθηκε στην μεριά του εξυπηρετητή (server side) και το οποίο αναλαμβάνει ουσιαστικά την επικοινωνία μεταξύ πελάτη (client) και βάσης δεδομένων. Ουσιαστικά συμβάλει στην πραγματοποίηση της επικοινωνίας του χρήστη με το κτίριο Τεχνικών Υπηρεσιών. Για πρώτη φορά γίνεται αναφορά στην χρήση των web services, της **MySQL** για την διαχείριση της βάσης δεδομένων, της **PHP**, της **XML**, του **JSON** και της αρχιτεκτονικής **REST**. Επιπλέον γίνεται εκτενής αναφορά για τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος.

4.1 Υπηρεσίες Δικτύου (Web Services)

4.1.1 Τι είναι οι Υπηρεσίες Δικτύου (Web Services)

Υπάρχουν πολλοί ορισμοί για το τι είναι web services, περίπου όσες και οι εταιρείες πληροφορικής που αναπτύσσουν εργαλεία για web services. Ένας πολύ καλός ορισμός δίνεται από την IBM. [41]

*“Τα **Web Services** είναι μία τεχνολογία που επιτρέπει στις εφαρμογές να επικοινωνούν μεταξύ τους ανεξαρτήτως πλατφόρμας και γλώσσας προγραμματισμού. Ένα web service είναι μία διεπαφή λογισμικού (software interface) που περιγράφει μία συλλογή από λειτουργίες, οι οποίες μπορούν να προσεγγιστούν από το δίκτυο μέσω πρότυπων μηνυμάτων σε γλώσσα XML (eXtensible Markup Language) ή JSON (JavaScript Object Notation). Χρησιμοποιεί πρότυπα βασισμένα στη γλώσσα XML για να περιγράψει μία λειτουργία (operation) προς εκτέλεση και τα δεδομένα προς ανταλλαγή με κάποια άλλη εφαρμογή. Μία ομάδα από web services οι οποίες αλληλεπιδρούν μεταξύ τους καθορίζει μίας εφαρμογή web services”.*

Επίσης ένας περιεκτικός ορισμός περιγράφεται από το W3C¹⁵: “*Το W3C προσδιορίζει μια υπηρεσία διαδικτύου (**Web Service**) σαν ένα σύστημα λογισμικού σχεδιασμένο να υποστηρίζει διαλειτουργική αλληλεπίδραση υπολογιστή με υπολογιστή πάνω από το Διαδίκτυο (Internet)*” [42]. Σύμφωνα με το W3C, μια υπηρεσία έχει συνήθως μια διεπαφή (σημείο στο οποίο είναι ορατή από το διαδίκτυο), προσφέρει πρόσβαση σε λειτουργίες (operations) μιας εφαρμογής λογισμικού (στη βιβλιογραφία αναφέρεται συχνά και ως application server) και η πρόσβαση σε αυτή επιτυγχάνεται μέσω ενός συγκεκριμένου μηχανισμού πρόσβασης. Το W3C επίσης αναφέρει ότι μπορούν να εντοπιστούν δύο είδη υπηρεσιών διαδικτύου, αυτές που ακολουθούν το πρότυπο REST (Representational State Transfer) χρησιμοποιώντας το πρωτόκολλο HTTP και χωρίς να αποθηκεύουν την κατάσταση, δηλαδή δεν διαθέτουν μνήμη (stateless) και αυτές που χρησιμοποιούν το πρωτόκολλο SOAP και έχουν διάφορες λειτουργίες (operations).

4.1.2 Σχεδιαστικές αρχές υπηρεσιών δικτύου

Μέσα στους στόχους των δικτυακών υπηρεσιών είναι να συνδυάσουν αρμονικά τεχνολογίες από διαφορετικούς κατασκευαστές, πάνω από το διαδίκτυο, με μια κοινή γλώσσα ανταλλαγής μηνυμάτων, αλλά κυρίως και η εύκολη επαναχρησιμοποίηση. Έτσι, οι βασικές σχεδιαστές αρχές των δικτυακών υπηρεσιών είναι:

- *Χρήση πρωτοκόλλων που βασίζονται στο δίκτυο (web)*
Τα πρωτόκολλα έχουν την δυνατότητα να διαβούν τα τείχη προστασίας (firewalls) και μπορούν να λειτουργούν σε ετερογενές περιβάλλον, όπως χαρακτηριστικά κάνει το HTTP πρωτόκολλο.
- *Διαλειτουργικότητα*
Το JSON καθορίζει ένα κοινό πρότυπο που επιτρέπει σε διαφορετικά συστήματα να επικοινωνήσουν.
- *Σχήμα XML (XML Schema) ή JSON*
Έγγραφα αναγνωρίσιμα από άνθρωπο και από μηχανή.
- *Συναρμολογησιμότητα*
Τα συστατικά υπηρεσιών (βιβλιοθήκες με τις οποίες δημιουργήθηκαν) είναι επαναχρησιμοποιήσιμα
- *Διαθεσιμότητα*
Οι υπηρεσίες είναι διαθέσιμες στα συστήματα που επιθυμούν να τις χρησιμοποιήσουν και οφείλουν να εκτεθούν έξω από αυτά.
- *Εφαρμογή-ανεξαρτησία*
Διαθεσιμότητα ανεξάρτητη των διεπαφών των διαφόρων υπηρεσιών.
- *Περιγραφή αναγνώσιμη από μηχανή και άνθρωπο*
Χρησιμοποιείται ώστε να προσδιορίσει τη διεπαφή, τη θέση και τις πληροφορίες πρόσβασης.

¹⁵ World Wide Web Consortium

4.1.3 Πλεονεκτήματα

Η τεχνολογία των υπηρεσιών δικτύου βάση των σχεδιαστικών αρχών που αυτές δημιουργήθηκαν, είναι προφανές ότι υπερτερεί παλαιότερων κατανεμημένων τεχνολογιών. Συγκεκριμένα, τα κυριότερα πλεονεκτήματά είναι η διαλειτουργικότητα, η εύκολη δημοσίευση και η επεκτασιμότητα. Αναλυτικά:

- *Διαλειτουργικότητα*: Μια υπηρεσία διαδικτύου παρέχει ανεξαρτησία τόσο από λειτουργικό σύστημα όσο και από το υλικό (hardware). Οποιοδήποτε πρόγραμμα που συμβαδίζει με αυτή την τεχνολογία μπορεί πολύ εύκολα να προσπελάσει μια τέτοια υπηρεσία.
- *Ενσωμάτωση*: Σε ένα υπάρχον πληροφοριακό σύστημα που λειτουργεί μέσα στο Διαδίκτυο, η δημιουργία μιας υπηρεσίας ιστού δεν απαιτεί μεγάλες αλλαγές στον μηχανισμό του συστήματος.
- *Διαθεσιμότητα και δημοσίευση*: Οι πληροφορίες για τις υπηρεσίες ιστού δημοσιεύονται, με αποτέλεσμα η εύρεση και η χρήση τους να είναι ταχύτατες. Επιπλέον βοήθεια για την χρήση μιας υπηρεσίας ιστού παρέχεται και από τα εργαλεία ανάπτυξης λογισμικού (Integrated Development Environments – IDE).
- *Επεκτασιμότητα*: Μια έτοιμη υπηρεσία ιστού είναι δυνατό να ανανεωθεί με εύκολο τρόπο παρέχοντας έτσι επιπρόσθετες υπηρεσίες στους χρήστες του. Παράλληλα, μπορούν να δημοσιεύονται καινούριες εκδόσεις μιας υπηρεσίας χωρίς «σπάνε» λογισμικά που έχουν πρόσβαση στις προηγούμενες εκδόσεις.
- *Μικρό κόστος δημιουργίας και χρήσης*: Εφόσον σε ένα λογισμικό υπάρχει ήδη κάποια διαδικασία που χρειάζεται να επεκταθεί σε online υπηρεσία, η δημιουργία μιας υπηρεσίας ιστού γίνεται πολύ γρήγορα και εύκολα στις περισσότερες των περιπτώσεων. Επίσης, το κόστος ενσωμάτωσης μιας υπηρεσίας ιστού σε κάποια ιστοσελίδα ή σε δικτυακή εφαρμογή είναι πάρα πολύ μικρό και γίνεται πολύ εύκολα, με τη χρήση κατάλληλων προγραμματιστικών εργαλείων. Ακόμα και στις περιπτώσεις που η χρήση κάποιας υπηρεσίας ιστού γίνεται με ενοικίαση, το συνολικό κόστος της χρήσης είναι αρκετά πιο μικρό από το κόστος δημιουργίας της υπηρεσίας αυτής.
- *Χρήση λογισμικού*: Όλο το λογισμικό και ειδικότερα οι ιστοσελίδες που χρησιμοποιούν υπηρεσίες γίνονται πιο λειτουργικές και πιο φιλικές αφού παρέχουν περισσότερες υπηρεσίες στους χρήστες.
- *XML/JSON*: Χρησιμοποιείται XML/JSON στο επίπεδο αναπαράστασης δεδομένων για όλα τα πρωτόκολλα υπηρεσιών ιστού και των τεχνολογιών που δημιουργούνται, έτσι οι τεχνολογίες μπορούν να διαλειτουργούν στο βασικό τους επίπεδο. Σαν επίπεδο μεταφοράς η XML ή η JSON αποβάλλει κάθε δικτύωση, λειτουργικό σύστημα ή δέσμευση πλατφόρμας που έχει πρωτόκολλο.
- *Δυνατότητα να είναι σύγχρονες και ασύγχρονες*: Ο συγχρονισμός αναφέρεται στη σύνδεση του πελάτη με την εκτέλεση της υπηρεσίας. Στις σύγχρονες επικλήσεις, ο πελάτης εμποδίζει και περιμένει την υπηρεσία να ολοκληρώσει την λειτουργία της πριν συνεχίσει. Οι ασύγχρονες λειτουργίες επιτρέπει στον πελάτη να καλέσει μια υπηρεσία και έπειτα να εκτελέσει άλλες λειτουργίες (για παράδειγμα, μια κλήση υπηρεσίας διαδικτύου από κινητό πρέπει να είναι ασύγχρονη, καθώς όσο

το κινητό περιμένει για δεδομένα πρέπει να μπορεί να ανταποκρίνεται στις εντολές του χρήστη). Οι ασύγχρονοι πελάτες ανακτούν το αποτέλεσμα τους αργότερα, ενώ οι σύγχρονοι πελάτες λαμβάνουν το αποτέλεσμα τους όταν η υπηρεσία έχει ολοκληρωθεί.

4.2 Επεκτάσιμη Γλώσσα Σήμανσης (eXtensible Markup Language, XML)

Η XML είναι μία γλώσσα σήμανσης, που περιέχει ένα σύνολο κανόνων για την ηλεκτρονική κωδικοποίηση κειμένων [43]. Όλα τα έγγραφα υπηρεσιών διαδικτύου είναι γραμμένα σε γλώσσα XML και το XML σχήμα χρησιμοποιείται για να καθορίσει τα στοιχεία που χρησιμοποιούνται στην επικοινωνία υπηρεσιών ιστού. Η ονομασία XML προκύπτει από τα αρχικά των λέξεων eXtensible Markup Language και ξεκίνησε να αναπτύσσεται από το W3C (World Wide Web Consortium) κυρίως για να ξεπεράσει τους περιορισμούς της HTML¹⁶. Η HTML είναι μια πάρα πολύ δημοφιλής γλώσσα σήμανσης. Σύμφωνα με κάποιες μελέτες υπάρχουν πάνω από ένα δισεκατομμύριο σελίδες στο δίκτυο και όλες βασίζονται στην HTML. Υποστηρίζεται από χιλιάδες εφαρμογές συμπεριλαμβανομένων των πλοηγών διαδικτύου, των επεξεργαστών κειμένου, του λογισμικού για ηλεκτρονικό ταχυδρομείο, των βάσεων δεδομένων και άλλων. Αν και η HTML είναι μια δημοφιλής και επιτυχημένη γλώσσα σήμανσης, έχει ορισμένα βασικά μειονεκτήματα. Η XML αναπτύχθηκε για να αντιμετωπίσει αυτά τα μειονεκτήματα και όχι για νεωτερισμό. Η XML υπάρχει λόγω της επιτυχίας της HTML, με αποτέλεσμα να έχει υιοθετήσει ορισμένα επιτυχημένα χαρακτηριστικά της.

Η XML σχεδιάστηκε δίνοντας έμφαση στην απλότητα, τη γενικότητα και τη χρησιμότητα στο Διαδίκτυο. Αν και η σχεδίαση της XML εστιάζει στα κείμενα, χρησιμοποιείται ευρέως για την αναπαράσταση αυθαίρετων δομών δεδομένων, που προκύπτουν για παράδειγμα στις υπηρεσίες ιστού. Αν συνοψίζαμε την XML σε μια πρόταση, θα λέγαμε ότι είναι «μία ομάδα προτύπων» για την ανταλλαγή και δημοσίευση πληροφοριών με ένα δομημένο τρόπο». Η έμφαση στη δομή δεν μπορεί να υποτιμηθεί.

Η XML είναι μια γλώσσα που χρησιμοποιείται στην περιγραφή και τον χειρισμό δομημένων εγγράφων. Εκατοντάδες γλώσσες βασίζονται στην XML συμπεριλαμβανομένων του RSS, του SOAP καθώς επίσης και προεπιλεγμένες κωδικοποιήσεις βασισμένες στην XML υπάρχουν για τις περισσότερες σουίτες εφαρμογών γραφείου, συμπεριλαμβανομένων του Microsoft Office, του OpenOffice.org και του iWork της εταιρίας Apple.

¹⁶ Hypertext Markup Language – Γλώσσα σήμανσης υπερκειμένου

4.3 JavaScript Object Notation (JSON)

Το **JSON (JavaScript Object Notation)** είναι ένα ελαφρύ πρότυπο ανταλλαγής δεδομένων [44]. Ο χρήστης μπορεί εύκολα, γρήγορα και απλά να το κατανοήσει και να γράψει σε αυτή την μορφή. Είναι εύκολο για τις μηχανές να το αναλύσουν (parse) και να το παράγουν (generate). Είναι βασισμένο πάνω σε ένα υποσύνολο της γλώσσας προγραμματισμού JavaScript. Το JSON είναι ένα πρότυπο κειμένου το οποίο είναι τελείως ανεξάρτητο από γλώσσες προγραμματισμού αλλά χρησιμοποιεί πρακτικές (conventions) οι οποίες είναι γνωστές στους προγραμματιστές της οικογένειας προγραμματισμού C, συμπεριλαμβανομένων των C, C++, C#, Java, JavaScript, Perl, Python, και πολλών άλλων. Αυτές οι ιδιότητες κάνουν το JSON μια ιδανική γλώσσα προγραμματισμού ανταλλαγής δεδομένων.

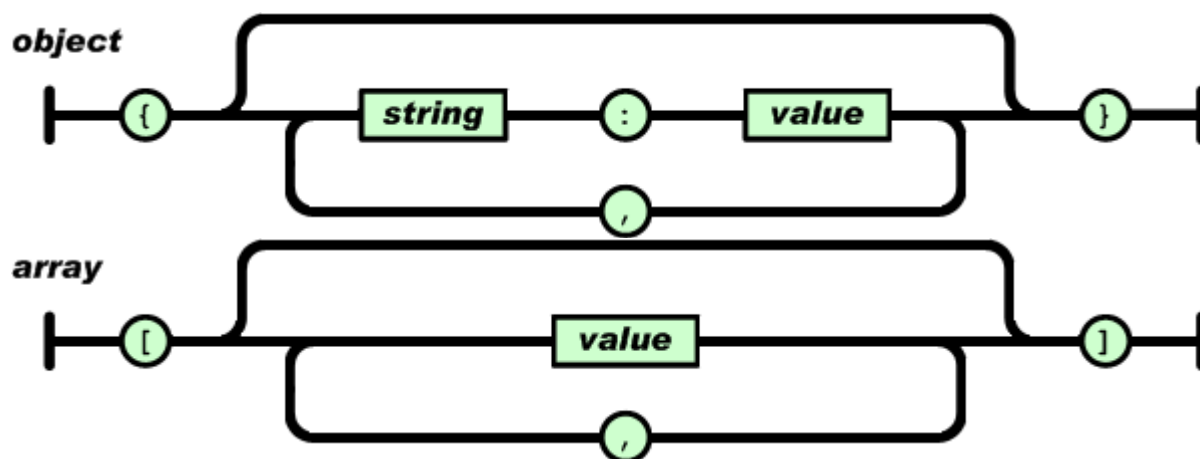
Το JSON βασίζεται σε δύο δομές:

- Μια συλλογή από ζευγάρια ονομάτων-τιμών. Σε διάφορες γλώσσες προγραμματισμού, αυτό θεωρείται ως ένα αντικείμενο (object), μία καταχώριση, μία δομή, ένα λεξικό, ένας πίνακας hash (hash table), μία λίστα κλειδιών, ή ένας associative πίνακας.
- Μία ταξινομημένη λίστα τιμών. Στις περισσότερες γλώσσες προγραμματισμού, αυτό θεωρείται ως ένας πίνακας (array), ένα διάνυσμα, μία λίστα, ή μία ακολουθία.

Αυτές οι δομές δεδομένων (data structures) είναι καθολικά αναγνωρισμένες και υποστηρίζονται από όλες τις σύγχρονες γλώσσες προγραμματισμού γεγονός που είναι λογικό, καθώς αφορά ένα πρότυπο που έχει την ικανότητα να μεταβάλλεται και να προσαρμόζεται πολύ γρήγορα και εύκολα, ενώ παράλληλα αφορά γλώσσες προγραμματισμού που υποστηρίζουν όλες τις μορφές που μπορεί να αποκτήσει.

Ένα αντικείμενο (object) είναι ένα άτακτο σύνολο από ζευγάρια ονομάτων-τιμών, ενώ ένας πίνακας (array) είναι μια συλλογή από τιμές σε σειρά και έχουν την μορφή που φαίνεται στην *Εικόνα 4.1*. Τέλος, η τιμή (value) μπορεί να αποτελεί είτε ένα string¹⁷, είτε έναν αριθμό, είτε μία δυαδική τιμή (Boolean), ή ένα αντικείμενο ή ακόμα και έναν πίνακα. Αυτές οι τιμές μπορεί να είναι και ανακατεμένες. Ένα απλό και χαρακτηριστικό παράδειγμα της μορφής που έχει ένα σύνολο αντικειμένων όταν πακετάρεται σε μορφή JSON φαίνεται στην *Εικόνα 4.2*.

¹⁷ Ένα string είναι μια συλλογή από μηδέν ή περισσότερους Unicode χαρακτήρες, μέσα σε εισαγωγικά (quotes).



Εικόνα 4.1- Τρόπος αναπαράστασης ενός αντικειμένου (πάνω) ή ενός πίνακα (κάτω) σε μορφή JSON

```
{
  "widget": {
    "debug": "on",
    "window": {
      "title": "Sample Konfabulator Widget",
      "name": "main_window",
      "width": 500,
      "height": 500
    },
    "image": {
      "src": "Images/Sun.png",
      "name": "sun1",
      "hOffset": 250,
      "vOffset": 250,
      "alignment": "center"
    },
    "text": {
      "data": "Click Here",
      "size": 36,
      "style": "bold",
      "name": "text1",
      "hOffset": 250,
      "vOffset": 100,
      "alignment": "center",
      "onMouseUp": "sun1.opacity = (sun1.opacity / 100) * 90;"
    }
  }
}
```

Εικόνα 4.2 – Απλό παράδειγμα JSON από πακετάρισμα αντικειμένων σε απλό κείμενο

4.4 Σύγκριση μεταξύ XML και JSON

Τα τελευταία χρόνια έχει αρχίσει και χρησιμοποιείται πολύ περισσότερο το JSON. Αυτό γιατί, με την άνηση της χρήσης των κινητών συσκευών, οι προγραμματιστές άρχισαν να το χρησιμοποιούν περισσότερο, καθώς είναι πιο ελαφρύ οπότε και μεταφέρει λιγότερα δεδομένα μέσω των δικτύων κινητής τηλεφωνίας, όπως αναφέρθηκε παραπάνω. Παρόλα αυτά, η XML είναι πιο εύκολα αναγνώσιμη από άνθρωπο, ενώ, από πολλούς θεωρείται ότι λόγω της παλαιότητάς της, υπάρχουν πολλές γλώσσες προγραμματισμού που έχουν ενσωματωμένες βιβλιοθήκες για την ανάκτηση και δημιουργία μηνυμάτων XML. Αυτό δεν ισχύει πλέον, καθώς το JSON είναι πάρα πολύ διαδεδομένο με αποτέλεσμα η χρήση του να είναι καθολική. Γενικότερα, η χρήση της μίας ή της άλλης γλώσσας εξαρτάται καταρχάς από την αρχιτεκτονική/πρωτόκολλο

(SOAP ή REST), αλλά και από τις γνώσεις/εμπειρία των προγραμματιστών. Στην παρούσα εργασία προτιμήθηκε το JSON, καθώς υλοποιήθηκε το πρωτόκολλο REST, που θα περιγραφεί αμέσως μετά, αλλά και για τον λόγο ότι είναι απλούστερο, πιο γρήγορο και πιο ελαφρύ.

4.5 Representational State Transfer (REST)

4.5.1 Γενικά

Αποτελεί ένα web service που υπακούει σε συγκεκριμένους κανόνες. Ακολουθεί το πρωτόκολλο HTTP, χρησιμοποιώντας συγκεκριμένες μεθόδους για την επικοινωνία μεταξύ του εξυπηρετητή (server) και του πελάτη (client). Οι κυριότερες μέθοδοι είναι οι GET, POST, PUT και DELETE.

Το **Representational State Transfer (REST)** [45] είναι ένα στυλ αρχιτεκτονικής για κατανεμημένα συστήματα το οποίο χρησιμοποιεί ίδιες τεχνικές με το πρωτόκολλο του διαδικτύου (World Wide Web). Ο όρος εισήχθη από τον Roy Fielding στην διδακτορική του διατριβή το 2000. Ο Fielding είναι ένας από τους συγγραφείς του HTTP πρωτοκόλλου. Το REST αναπτύχθηκε παράλληλα με το HTTP, βασισμένο στο σχεδιασμό του HTTP. Η μεγαλύτερη υλοποίηση του REST είναι το δίκτυο παγκοσμίου ιστού (world wide web), όπου δείχνει πως η αρχιτεκτονική του Internet αναπτύχθηκε με τον χαρακτηρισμό και την κατηγοριοποίηση των τεσσάρων τμημάτων του, origin servers, gateways, proxies και clients. Η αρχιτεκτονική του REST αποτελείται από πελάτες και εξυπηρετητές (clients and servers). Οι πελάτες δημιουργούν αιτήματα προς τους εξυπηρετητές και, εκείνοι τα επεξεργάζονται και επιστρέφουν κατάλληλες απαντήσεις. Τα αιτήματα και οι απαντήσεις είναι κατασκευασμένα με βάση τη μεταφορά πόρων. Ένας πόρος μπορεί να είναι κάτι στο δίκτυο στο οποίο μπορεί να του απευθυνθεί αίτημα. Ο πελάτης μπορεί να αρχίσει να στέλνει αιτήματα, και να περιμένει για απάντηση. Τα μηνύματα που μεταφέρονται μέσω REST μπορεί να είναι οποιασδήποτε μορφής, όπως για παράδειγμα δυαδικά (binary), POX (Plain Old XML – κλασσικά XML έγγραφα) ή JSON (JavaScript Object Notation – μια ειδική γλώσσα σήμανσης [4.3]).

4.5.2 Περιορισμοί

Η αρχιτεκτονική του REST περιγράφει τους έξι παρακάτω περιορισμούς, ενώ αφήνει τις λεπτομέρειες της υλοποίησης ανοιχτές στον προγραμματιστή [46].

- **Διαχωρισμός πελάτη (client) και εξυπηρετητή (server).** Οι πελάτες είναι διαχωρισμένοι από τους εξυπηρετητές με μια συγκεκριμένη διεπαφή. Για παράδειγμα, οι πελάτες δεν ανησυχούν για την αποθήκευση των δεδομένων,

αλλά αφήνεται στον εξυπηρετητή. Έτσι, υπάρχει φορητότητα στον κώδικα του πελάτη, ενώ οι εξυπηρετητές δεν ασχολούνται με την διεπαφή του χρήστη, άρα είναι απλούστεροι και επεκτάσιμοι.

- **Stateless.** Η επικοινωνία πελάτη και εξυπηρετητή επιβάλλει να μην αποθηκεύεται πληροφορία (client context) στον εξυπηρετητή κατά τη διάρκεια αιτήματος/απάντησης. Κάθε αίτημα από τον πελάτη πρέπει να ενσωματώνει χαρακτηριστικά για την αναγνώριση του αιτήματος (όπως για παράδειγμα πληροφορίες εισόδου – login information).
- **Cacheable.** Το HTTP πρωτόκολλο, από τη φύση του, επιτρέπει στους πελάτες να αποθηκεύουν (cache) πληροφορίες τις οποίες κατεβάζουν από κάποιον εξυπηρετητή. Έτσι, οι REST απαντήσεις προσδιορίζουν αν είναι υπάρχει δυνατότητα αποθήκευσης στη μεριά του πελάτη, με αποτέλεσμα να ελαττώνεται ο φόρτος του εξυπηρετητή.
- **Layered System.** Ένας πελάτης δεν μπορεί να προσδιορίσει αν είναι κατευθείαν συνδεδεμένος στον εξυπηρετητή ή σε κάποιον ενδιάμεσο. Οι ενδιάμεσοι εξυπηρετητές μπορούν να βελτιώσουν την απόδοση και την ασφάλεια του συστήματος.
- **Code on demand.** Οι εξυπηρετητές μπορούν να μεταφέρουν/επεκτείνουν τμήμα της λογικής στον πελάτη (π.χ. με χρήση JavaScript, Silverlight, Flash).
- **Uniform Interface.** Οι REST υπηρεσίες θα πρέπει να είναι loosely coupled, ώστε να μπορούν να αναπτυχθούν παράλληλα σε πελάτη και εξυπηρετητή.

4.6 Τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του Restful Web Service

4.6.1 Προεπεξεργαστής Υπερκειμένου PHP (PHP Hypertext Preprocessor)



Εικόνα 4.3 – Λογότυπο PHP

4.6.1.1 Γενικά – Ιστορική Αναδρομή

Η **PHP** είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων διαδικτύου (web) με δυναμικό περιεχόμενο [47]. Μια ιστοσελίδα PHP επεξεργάζεται από έναν συμβατό εξυπηρετητή (server) του παγκόσμιου ιστού (για παράδειγμα Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα πλοήγησης (browser) των επισκεπτών σε μορφή κώδικα HTML. Επομένως, σε μια HTML σελίδα είναι εύκολο να ενσωματωθεί PHP κώδικας, που εκτελείται κάθε φορά που γίνεται επίσκεψη στη συγκεκριμένη σελίδα.

Πριν ακόμα την εμφάνιση της PHP, υπήρχε μεγάλη δραστηριότητα στο δικτυακό προγραμματισμό (Web programming). Παλαιότερα, ο κώδικας για να γίνει επεξεργασία των δεδομένων (καταχωρήσεις) μιας φόρμας, γραφόταν σε γλώσσα προγραμματισμού C. Ο κώδικας αυτός ήταν πολύ γρήγορος στην εκτέλεσή του, όμως ήταν πολύ δύσκολος και πολύπλοκος στο γράψιμό του. Ο κύριος λόγος ήταν ότι η C δεν είχε σχεδιασθεί ειδικά για το δίκτυο και έτσι δεν υπήρχε έτοιμος κώδικας για την εκτέλεση κάποιων κοινών εργασιών, με αποτέλεσμα ο προγραμματιστής (Web developer) να τα κάνει όλα μόνος του. Αυτό το πρόβλημα αντιμετωπίστηκε μερικώς με μια ευκολότερη γλώσσα υψηλού επιπέδου, την Perl, όπου τα αρχικά προέρχονται από τις λέξεις "Practical Extraction and Report Language". Η Perl, αν και αρχικά δημιουργήθηκε ως μια γλώσσα επεξεργασίας κειμένου, είχε δυνατότητες για επεξεργασία των καταχωρήσεων μιας HTML φόρμας και όχι μόνο. Η σχεδίαση της Perl ήταν απλή, καθώς ένα script της Perl μπορούσε να κάνει όλες τις απαραίτητες λειτουργίες μιας ιστοσελίδας και μπορούσε να ενσωματωθεί σε αυτό εύκολα κώδικας της HTML. Η Perl διέθετε ακόμη πάρα πολλές συναρτήσεις (functions) για να γίνονται

εύκολα πολλές εργασίες και έτσι έγινε σύντομα πολύ δημοφιλής ανάμεσα στους Web developers. Αν και αποτέλεσε ένα σημαντικό βήμα για το Web development, η Perl απέιχε ακόμα πολύ από το ιδανικό, με το μεγαλύτερο ελάττωμά της να αποτελεί το γεγονός ότι ήταν Perl-Centric. Αυτό σημαίνει ότι για να δημιουργηθεί HTML κώδικας, έπρεπε να ενσωματώσουμε τον HTML κώδικα μέσα στην Perl.

Η PHP δημιουργήθηκε το 1994 και ήταν αρχικά η δουλειά ενός ατόμου, του Rasmus Lerdorf. Η PHP αρχικά σήμαινε Personal Home Page (προσωπική αρχική σελίδα), αλλά άλλαξε σύμφωνα με την σύμβαση GNU και τώρα σημαίνει PHP Hypertext Preprocessor (προεπεξεργαστής κειμένου PHP). Τότε, ο κύριος σκοπός ήταν να υπάρχει η λιγότερη δυνατή ποσότητα λογικής στην επίτευξη του αποτελέσματος και αυτό οδήγησε την PHP στο να γίνει HTML-centric, δηλαδή ο κώδικας της PHP ήταν ενσωματωμένος μέσα στον κώδικα της HTML. Η πρώτη έκδοση είχε πολλά προβλήματα, κυρίως στο κομμάτι του αναλυτή (parser). Σε επόμενες εκδόσεις της PHP υπήρχαν τεράστιες και καινοτόμες αλλαγές - βελτιώσεις, με αποτέλεσμα να καθιερωθεί παγκόσμια. Υιοθετήθηκε και από άλλα ταλαντούχα άτομα και έχει περάσει από πολλές βασικές εκδόσεις. Τον Ιανουάριο του 2001 χρησιμοποιούνταν σχεδόν σε πέντε εκατομμύρια τομείς παγκόσμια και αυτός ο αριθμός μεγαλώνει γρήγορα. Η PHP είναι ένα προϊόν ανοιχτού κώδικα. Έτσι είναι δυνατή οποιαδήποτε χρήση της χωρίς χρέωση, αλλά κυρίως οποιαδήποτε αλλαγή και αναδιανομή της. Απέκτησε πλήρη συμβατότητα με πολλούς εξυπηρετητές δικτύου (web servers), έγινε πολύ ταχύτερη εφαρμόζοντας τη λογική «μεταγλώττιση πρώτα - εκτέλεση αργότερα», ενώ εισήγαγε την πολυεπεξεργασία (multi-threading), όπου μπορούν κάποιες συναρτήσεις να εκτελούνται ανεξάρτητα από το κυρίως script. Η πέμπτη έκδοση της PHP αποτελεί μεγάλη καινοτομία αλλά και μεγάλη πρόοδο για την ίδια την γλώσσα, καθώς υποστηρίζει τον αντικειμενοστραφή προγραμματισμό (object-oriented). Επίσης, υπάρχει μια μεγάλη ποικιλία από συναρτήσεις για αντικείμενα (objects) που τα κάνει πολύ πιο ευέλικτα και εύκολα στη χρήση τους.

4.6.1.2 Πλεονεκτήματα

Η γλώσσα προγραμματισμού PHP όπως αναφέρθηκε παραπάνω, δεν αποτέλεσε την απαρχή των γλωσσών προγραμματισμού για δίκτυα, αλλά αντιθέτως βασίστηκε σε άλλες γλώσσες προγραμματισμού. Παράλληλα, με την πάροδο τον χρόνων, όπως είναι λογικό, απέκτησε πολλές γλώσσες προγραμματισμού ανταγωνιστές. Όμως, η PHP υπερτερεί των περισσότερων ανταγωνιστών της και τα κυριότερα πλεονεκτήματα που διαθέτει είναι τα εξής:

- *Υψηλή Απόδοση:* Η PHP είναι πολύ αποτελεσματική. Μετά από πολυάριθμες δοκιμές σε φθηνούς εξυπηρετητές (servers) από μεγάλες εταιρείες, ανταποκρίθηκε πλήρως και αποδείχθηκε ότι είναι καλύτερη των ανταγωνιστών της.
- *Συμβατότητα με συστήματα Βάσεων Δεδομένων:* Η PHP έχει εγγενείς συνδέσεις για πολλά συστήματα βάσεων δεδομένων. Εκτός από την πιο διαδεδομένη και

γνωστή σε παγκόσμια κλίμακα, MySQL, έχει την δυνατότητα να συνδεθεί άμεσα με όλες σχεδόν τις γνωστές βάσεις δεδομένων, εκ των οποίων οι περισσότεροι γνωστές είναι οι PostgreSQL, Oracle, Informix, InterBase, Sybase, filePro.

- *Ενσωμάτωση πληθώρας βιβλιοθηκών:* Ο κύριος λόγος που σχεδιάστηκε η PHP ήταν η χρήση της στο διαδίκτυο. Αυτό έχει σαν αποτέλεσμα να ενσωματώνει πάρα πολλές βιβλιοθήκες και να γίνεται εύκολη η χρήση λειτουργιών σχετικών με το διαδίκτυο. Μερικές από τις πιο σημαντικές είναι η σύνδεση με υπηρεσίες άλλων δικτύων, η αποστολή μηνυμάτων μέσω του ηλεκτρονικού ταχυδρομείου, η χρήση και εύκολη διαχείριση των cookies και η δημιουργία εγγράφων pdf.
- *Χαμηλό Κόστος:* Συγκεκριμένα, η χρήση της έχει μηδενικό κόστος καθώς οποιαδήποτε έκδοση διατίθεται μέσω της επίσημης ιστοσελίδας¹⁸ εντελώς δωρεάν.
- *Ευκολία στη χρήση και στην εκμάθηση:* Το συντακτικό της βασίζεται σε άλλες γλώσσες προγραμματισμού όπως η Perl και η C, που είναι υψηλού επιπέδου γλώσσες και διαδεδομένες σε παγκόσμια κλίμακα.
- *Συμβατότητα με πληθώρα λειτουργικών συστημάτων:* Ο κώδικας σε PHP είναι πλήρως συμβατός με οποιοδήποτε ελεύθερο λειτουργικό σύστημα, όπως Linux, Android (κινητά τηλέφωνα) αλλά και πιο εμπορικές εκδόσεις όπως Windows και Macintosh, iOS (κινητά τηλέφωνα). Ο κώδικας σχεδόν πάντα ανταποκρίνεται χωρίς καμία αλλαγή σε όλα αυτά τα συστήματα.
- *Ανοιχτού κώδικα Λογισμικό:* Αυτό σημαίνει ότι υπάρχει πρόσβαση στον πηγαίο κώδικα της PHP από κάθε προγραμματιστή. Αυτό πάντα αποτελεί πλεονέκτημα κυρίως για τους προγραμματιστές, οι οποίοι πάντα θέλουν πειραματίζονται, είτε να αλλάζουν είτε να προσθέτουν πράγματα στον κώδικα. Όπως σε κάθε ανοιχτό λογισμικό αυτό είναι εφικτό χωρίς να απαιτείται αναμονή ώστε να υπάρξουν βελτιώσεις και νεότερες εκδόσεις από τον ίδιο τον κατασκευαστή.

¹⁸ PHP official site : www.php.net

4.6.2 MySQL



Εικόνα 4.4 – Λογότυπο MySQL

4.6.2.1 Γενικά

Η **MySQL** είναι διαθέσιμη από το 1996 αλλά η ιστορία της ξεκινάει από το 1979. Πλέον αποτελεί ιδιοκτησία της εταιρείας Oracle [48]. Η MySQL είναι ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων. Μία βάση δεδομένων παρέχει την δυνατότητα αποθήκευσης, αναζήτησης, ταξινόμησης και ανάκλησης δεδομένων με αποτελεσματικό τρόπο. Ένας MySQL εξυπηρετητής ελέγχει την πρόσβαση στα δεδομένα που βρίσκονται στην βάση και δίνει τη δυνατότητα σε πολλούς χρήστες να την χρησιμοποιούν ταυτόχρονα. Παρέχει, λοιπόν, ταχύτητα αλλά και ασφάλεια, καθώς μόνο πιστοποιημένοι χρήστες έχουν δικαίωμα προσπέλασης στην βάση δεδομένων, με αποτέλεσμα η MySQL να χαρακτηρίζεται ως ένας πολυνηματικός εξυπηρετητής (multithread server) πολλαπλών χρηστών. Η ονομασία MySQL αποτελείται από δύο συνθετικά. Το My είναι το όνομα της κόρης του συνιδρυτή του συστήματος Monty Widenius και το SQL αναφέρεται στη γλώσσα SQL (Structured Query Language). Η SQL είναι η τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων παγκοσμίως. Συγκεκριμένα, αποτελεί μια γλώσσα υπολογιστών που σχεδιάστηκε ανεξάρτητα από την MySQL, με στόχο τη διαχείριση δεδομένων σε ένα σύστημα διαχείρισης βάσεων δεδομένων. Το πρότυπο της γλώσσας SQL αναπτύχθηκε ξεχωριστά από τις υλοποιήσεις των συστημάτων διαχείρισης βάσεων δεδομένων (όπως της MySQL, της PostgreSQL κλπ). Κάθε τέτοια υλοποίηση μπορεί να διαφέρει από άλλη παρέχοντας επιπλέον δυνατότητες και βελτιωμένα χαρακτηριστικά. Οι εντολές της SQL μπορούν να κληθούν μέσα από το περιβάλλον της γλώσσας προγραμματισμού PHP. Οι δυναμικές ιστοσελίδες χρησιμοποιούν βάσεις δεδομένων στις οποίες αρχικά αποθηκεύουν και έπειτα σε κάθε κλήση ανακτούν το περιεχόμενό τους και εμφανίζονται με δυναμικό τρόπο σε κάθε χρήστη – πελάτη (client).

4.6.2.2 Πλεονεκτήματα

Η MySQL έχει διαδοθεί και χρησιμοποιείται σε παγκόσμια κλίμακα. Φυσιολογικό είναι όμως το γεγονός ότι έχει αρκετούς, ισάξιους κάποιες φορές, ανταγωνιστές, με τους κυριότερους από αυτούς να είναι οι PostgreSQL, Microsoft SQL και Oracle. Όμως συνεχίζει να παραμένει η καλύτερη επιλογή στις περισσότερες περιπτώσεις, καθώς διαθέτει πολλά πλεονεκτήματα έναντι των αντιπάλων της. Τα κυριότερα είναι τα εξής:

- *Υψηλή Απόδοση:* Η MySQL είναι δίχως καμία αμφιβολία αρκετά γρήγορη. Έπειτα από πολλές δοκιμές που έχουν γίνει, φαίνεται ότι είναι στις περισσότερες περιπτώσεις ταχύτερη από τους ανταγωνιστές της.
- *Χαμηλό Κόστος:* Η MySQL λειτουργεί είτε με άδεια ανοιχτού κώδικα (open source), όπου η αγορά είναι δωρεάν, είτε με εμπορική άδεια, παρέχοντας περισσότερες δυνατότητες έναντι, όμως, ενός χαμηλού αντίτιμου.
- *Ανοιχτού κώδικα λογισμικό:* Από την στιγμή που παρέχεται άδεια ανοιχτού κώδικα, αυτόματα σημαίνει ότι παρέχεται ο πηγαίος κώδικας (source code) της MySQL προς τροποποίηση και βελτιώσεις από οποιονδήποτε προγραμματιστή.
- *Συμβατότητα:* Η MySQL έχει την δυνατότητα να χρησιμοποιηθεί και να εγκατασταθεί σε πολλά διαφορετικά συστήματα (λειτουργικά) και πλατφόρμες όπως είναι τα Unix, Microsoft Windows, Apple Macintosh, Android, Apple iOS (κινητά τηλέφωνα).
- *Ευκολία στην χρήση - εκμάθηση:* Οι περισσότερες μοντέρνες βάσεις δεδομένων, αν όχι όλες, χρησιμοποιούν την πρότυπη γλώσσα SQL που αποτελεί μία απλή, εύχρηστη και παγκόσμια αναγνωρισμένη γλώσσα. Η MySQL την υποστηρίζει, γεγονός που την καθιστά εύκολη στη χρήση και την εκμάθηση.

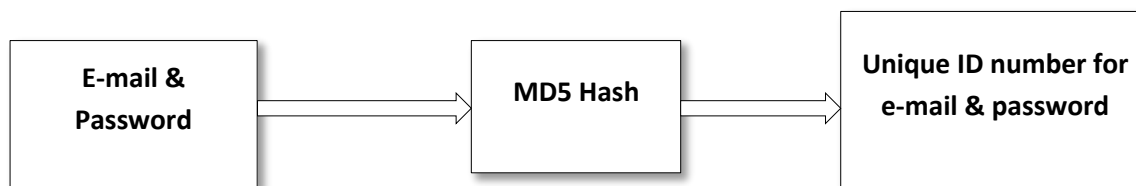
4.7 Υλοποίηση του εξυπηρετητή στα πλαίσια της παρούσας εφαρμογής

4.7.1 Εισαγωγή

Η επιτυχής λειτουργία της εφαρμογής **S.M.C.**, όπως αναφέρθηκε νωρίτερα, απαιτεί την ύπαρξη ενός εξυπηρετητή που θα επιτρέπει την επικοινωνία της με το κτίριο Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης, όπου βρίσκονται εγκατεστημένοι οι αισθητήρες και οι ενεργοποιητές. Ουσιαστικά απαιτείται ένας τρόπος επικοινωνίας μεταξύ της εφαρμογής και της βάσης δεδομένων που περιέχει όλα αυτά τα δεδομένα, δηλαδή χρειάζεται μία ζεύξη μεταξύ client και server. Υπεύθυνος για αυτή την επικοινωνία λοιπόν είναι ο εξυπηρετητής, κοινώς server. Επομένως, εύκολα αντιλαμβάνεται κανείς πόσο σημαντικό κομμάτι αποτελεί η άρτια, μεθοδική και αποτελεσματική υλοποίηση ενός κατάλληλου server, που στην «γλώσσα προγραμματισμού» εφαρμογών αναφέρεται και ως backend, για το σύνολο της εφαρμογής της που πραγματεύεται η παρούσα εργασία.

Για την μεταφορά των δεδομένων μεταξύ client και server απαιτείται να πακεταριστούν με κάποιο τρόπο. Ο τρόπος που πακετάρονται και η μορφή που παίρνουν είναι αυτή που ορίζει η τεχνολογία του JSON. Μετατρέπονται, λοιπόν, σε JSON μορφή από τον server στο iPhone και αντίστροφα, μέσω της μεθόδου POST του πρωτοκόλλου HTTP. Η χρήση του JSON, όπως περιγράφηκε νωρίτερα [4.4] προτιμάται στην περίπτωση αυτή, καθώς αποτελεί μία πιο «ελαφριά» μορφή ανταλλαγής δεδομένων που επιτρέπει στα δεδομένα να πακετάρονται εύκολα και γρήγορα σε αντικείμενα, με χρήση μορφής απλού κειμένου. Θεωρείται πολύ απλούστερο από την εναλλακτική επιλογή χρήσης της τεχνολογίας XML. Επιπροσθέτως, το πακετάρισμα αντικειμένων σε μορφή JSON και η αντίστροφη διαδικασία υποστηρίζονται από τις περισσότερες γλώσσες προγραμματισμού και πλατφόρμες.

Αξίζει αναφοράς το γεγονός ότι η ασφάλεια των προσωπικών στοιχείων κάθε χρήστη αποτέλεσε βασική αρχή και προϋπόθεση για την υλοποίηση της παρούσας εφαρμογής. Συγκεκριμένα, η μεταφορά δεδομένων και η αποθήκευσή τους στις βάσεις δεδομένων περιέχει κίνδυνο παραβίασης και υποκλοπής. Επομένως, ένας τρόπος ώστε τα προσωπικά στοιχεία του χρήστη να είναι περισσότερο ασφαλή, αποτελεί η κρυπτογράφηση του προσωπικού κωδικού αλλά και του προσωπικού e-mail που χρησιμοποιούνται για την πιστοποίηση και είσοδό του στο σύστημα. Η κρυπτογράφηση γίνεται πριν από την αποθήκευση στην βάση δεδομένων, μέσω της συνάρτησης MD5-hash. Η συγκεκριμένη συνάρτηση δημιουργεί μοναδικούς αριθμούς για κάθε ένα από αυτά, ενώ για την αποκωδικοποίησή τους εφαρμόζεται πάλι η ίδια συνάρτηση. Απλή αναπαράσταση της λειτουργίας της φαίνεται στην *Εικόνα 4.5*.



Εικόνα 4.5 – Απεικόνιση διαδικασίας κρυπτογράφησης για το προσωπικό e-mail και κωδικό του χρήστη

4.7.2 RESTful Web Service – REST PHP Server

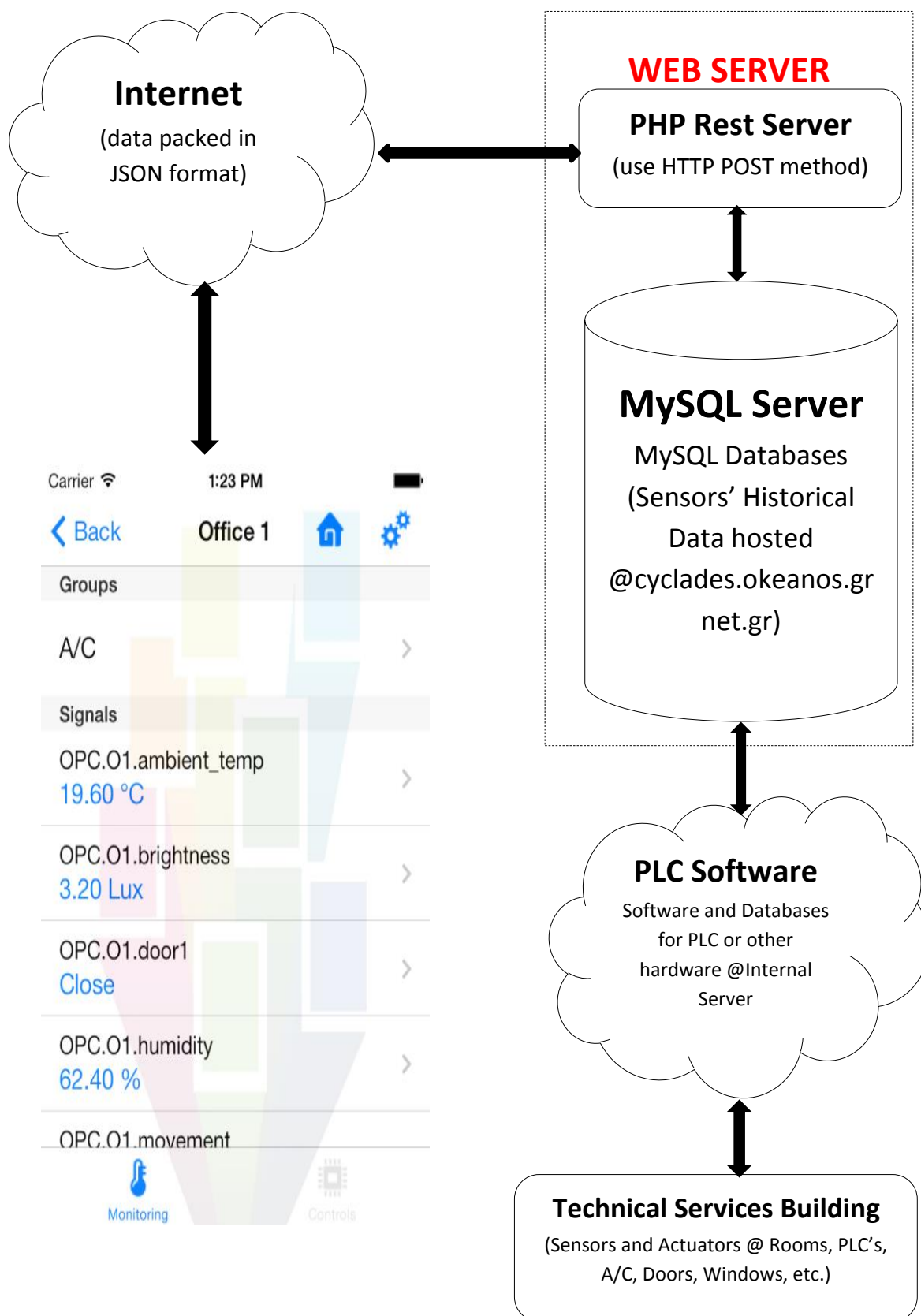
Το backend της παρούσας εφαρμογής επιλέχθηκε να είναι ένα RESTful web service. Πρόκειται λοιπόν, για έναν server που υπακούει στο πρωτόκολλο REST και φιλοξενεί τις βάσεις δεδομένων που διαθέτουν όλες τις πληροφορίες για τα δωμάτια, τις μονάδες κλιματιστικών και γενικότερα τις εγκαταστάσεις που έχουν τοποθετηθεί οι αισθητήρες και οι ενεργοποιητές, καθώς επίσης και τις τιμές τους ανά ένα λεπτό της ώρας.

Ο εξυπηρετητής (web server) δημιουργήθηκε και φιλοξενείται από την υπηρεσία εικονικών δικτύων Cyclades, στα πλαίσια του προγράμματος Okeanos¹⁹ της ΕΔΕΤ (GRNET)²⁰. Ο εξυπηρετητής είναι σε περιβάλλον Ubuntu Server 12.04 LTS [50], βρίσκεται σε σύνδεση 24 ώρες, όλες τις ημέρες της εβδομάδος (online 24h/7d) και έχει υλοποιηθεί σε γλώσσα προγραμματισμού PHP. Ουσιαστικά ο web server αποτελείται από ένα PHP REST server, ένα MySQL server και φυσικά τον Apache HTTP server, όπως περιγράφεται παρακάτω.

Στην *Εικόνα 4.6* παρουσιάζεται ο τρόπος με τον οποίο πραγματοποιείται η επικοινωνία μεταξύ του κινητού τηλεφώνου και της βάσης δεδομένων μέσω του web server. Συγκεκριμένα, το κινητό τηλέφωνο επικοινωνεί με τον PHP Server με χρήση της μεθόδου POST του HTTP πρωτοκόλλου. Ο PHP Server με την σειρά του, πραγματοποιεί τα κατάλληλα ερωτήματα (queries) στον MySQL Server και ανακτά τα ζητούμενα δεδομένα. Τα δεδομένα για να «ταξιδέψουν» μετατρέπονται πάντα σε μορφή JSON, που αποτελεί έναν γρήγορο, εύελικτο και εύκολο τρόπο πακεταρίσματος των δεδομένων. Σε κάθε νέα εισερχόμενη σύνδεση, τα δεδομένα ανακτώνται από τις μεταβλητές POST του HTTP πρωτοκόλλου, αναλύονται και γίνονται οι κατάλληλες ερωτήσεις στην βάση δεδομένων MySQL. Η ανάλυση των δεδομένων γίνεται από τον PHP server, καθώς η επίσημη βιβλιοθήκη της γλώσσας PHP διαθέτει συναρτήσεις κατάλληλες για το «(ξε-)πακετάρισμα» των δεδομένων σε μορφή JSON.

¹⁹ Υπηρεσία Okeanos προσφέρει δυνατότητα δημιουργίας εικονικών μηχανημάτων στα πλαίσια του προγράμματος Synnefo [49]

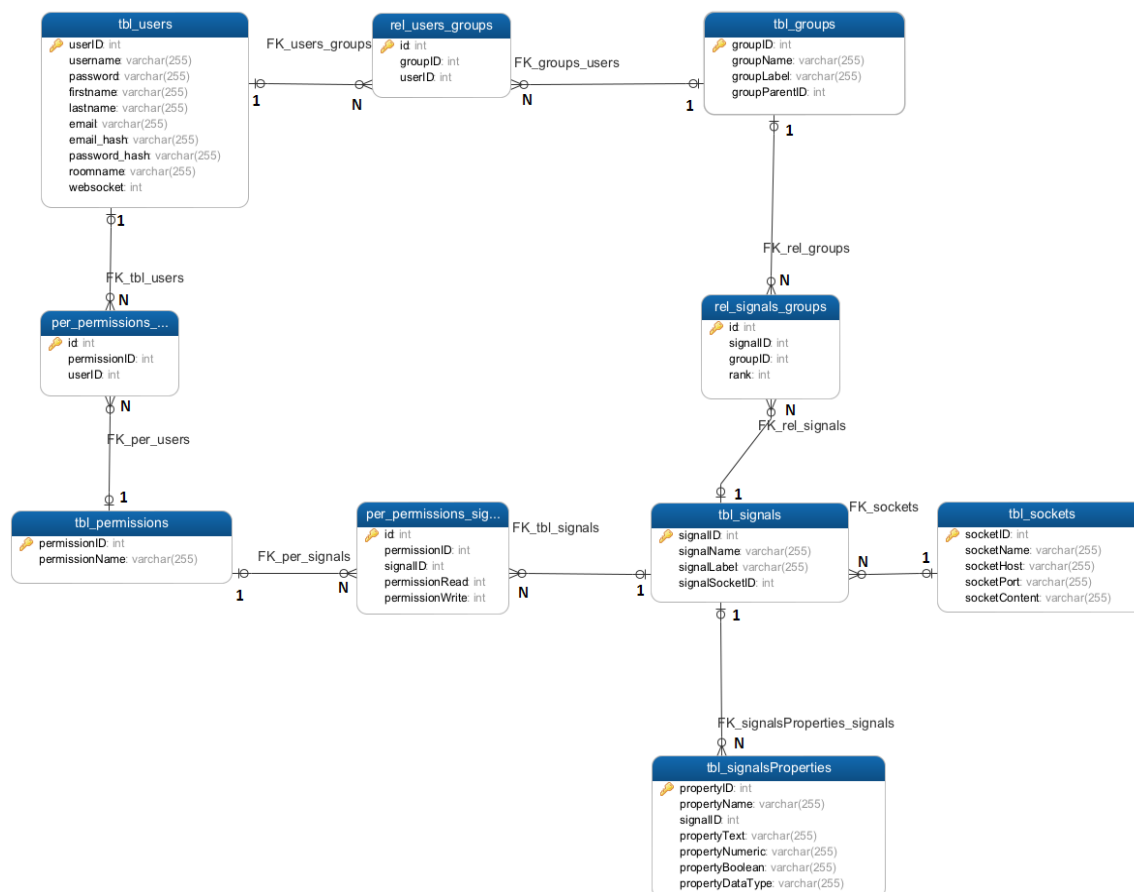
²⁰ Εθνικό Δίκτυο Έρευνας και Τεχνολογίας (Greek Research & Technology Network). Περισσότερα στο www.grnet.gr



Εικόνα 4.6 – Γραφική απεικόνιση επικοινωνίας/μεταφοράς δεδομένων μεταξύ πλατφόρμας κινητού τηλεφώνου και εξυπηρετητή (frontend – backend components)

4.7.3 Βάση Δεδομένων

Όπως αναφέρθηκε προηγουμένως, απαιτείται η υλοποίηση μίας βάσης δεδομένων για την αποθήκευση όλων των σημάτων και των τιμών τους. Στην παρούσα εργασία υλοποιήθηκαν δύο βάσεις δεδομένων, τηρώντας πιστά την υλοποίηση που υπήρχε ήδη. Ουσιαστικά, δόθηκε αυτούσιο το μεγαλύτερο μέρος των βάσεων δεδομένων που χρησιμοποιούνται στο κτίριο Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης και έχουν υλοποιηθεί από την ομάδα του PEBBLE. Πρόκειται για δύο ξεχωριστές MySQL βάσεις δεδομένων. Η πρώτη χρησιμοποιείται για την αποθήκευση των σημάτων, αισθητήρων και ενεργοποιητών, των χαρακτηριστικών τους, των δωματίων-χώρων, των στοιχείων του λογαριασμού κάθε χρήστη, των κατηγοριών των χρηστών αλλά και των περιορισμών κάθε κατηγορίας. Η δεύτερη βάση υλοποιήθηκε ώστε να διατηρεί τις τιμές όλων των σημάτων, δηλαδή τα ιστορικά δεδομένα, όπου διατηρεί τις τιμές κάθε σήματος ανά ένα λεπτό, από την αρχή του προγράμματος. Στην *Εικόνα 4.7* φαίνεται το διάγραμμα οντοτήτων-σχέσεων, ER, της πρώτης και βασικής βάσης δεδομένων, όπως εξήχθη από το πρόγραμμα Navicat [4.8.4].



Εικόνα 4.7 – Διάγραμμα οντοτήτων-σχέσεων βάσης δεδομένων, ER.

4.7.4 Επικοινωνία iPhone - Web Server σε επίπεδο προγραμματισμού

Για την μεταφορά των δεδομένων από τον web server στο iPhone και αντίστροφα, απαιτούνται αρκετά βήματα σε επίπεδο προγραμματισμού, δηλαδή σε επίπεδο Objective-C. Αξίζει μερίδιο αναφοράς στο τρόπο επικοινωνίας προγραμματιστικά λοιπόν. Αρχικά για την εγκαθίδρυση της επικοινωνίας του iPhone με τον server χρησιμοποιείται η μέθοδος **NSURLSession**. Η συγκεκριμένη μέθοδος προσφέρει ασύγχρονη σύνδεση, δηλαδή η ανταλλαγή δεδομένων γίνεται χωρίς να μπλοκάρεται η λειτουργία της εφαρμογής, όση ώρα διαρκεί αυτή. Επίσης μέσω της μεθόδου **NSJsonSerialization** γίνεται κάθε φορά το «πακετάρισμα» και «ξεπακετάρισμα» των δεδομένων, από μορφή **NSData**, σε μορφή **NSDictionary**, δηλαδή ένα είδος πίνακα με μεγαλύτερη ευελιξία.

Για την προβολή λοιπόν των αισθητήρων σε μία ζώνη, γίνεται ένα αίτημα στον server, με χρήση της μεθόδου POST του πρωτοκόλλου HTTP, για την συγκεκριμένη ζώνη. Ο server στέλνει μία απάντηση, που περιλαμβάνει σε μορφή JSON τους αισθητήρες και την τιμή τους τη δεδομένη χρονική στιγμή. Τότε τα δεδομένα ξεπακετάρονται σε μορφή **NSDictionary**, και έπειτα παρουσιάζονται στην οθόνη του iPhone.

4.7.5 Χρήση Ιδιωτικών Βιβλιοθηκών

Παρακάτω θα παρουσιαστεί μία λειτουργία που υποστηρίζει η εφαρμογή **S.M.C.**, αλλά έγινε η υλοποίησή της στην μεριά του εξυπηρετητή (server). Συγκεκριμένα πρόκειται για την λειτουργία ανάκτησης του κωδικού πρόσβασης για κάθε χρήστη. Αυτή η λειτουργία απαιτεί όπως είναι προφανές, την αποστολή ενός e-mail στον χρήστη που θα περιέχει τον κωδικό πρόσβασης. Η αποστολή ενός e-mail μέσω της PHP γίνεται με χρήση της συνάρτησης **mail()**, αλλά αυτή αφενός δεν προσφέρει πολλές δυνατότητες, αφετέρου δοκιμάστηκε και δεν είχε επιτυχία στην αποστολή, δηλαδή τα μηνύματα δεν εστάλιναν ποτέ. Επομένως, χρησιμοποιήθηκε η βιβλιοθήκη **“PHPMailer”** [51]. Είναι πολύ απλή και εύκολη στην χρήση, προφανώς γραμμένη σε PHP και έχει την δυνατότητα να χρησιμοποιεί το SMTP πρωτόκολλο ώστε να πραγματοποιεί αποστολή ενός πλήρους e-mail μέσα σε λίγες γραμμές κώδικα.

4.8 Προγραμματιστικά Εργαλεία για την υλοποίηση του εξυπηρετητή της εφαρμογής

Αξίζει να αναφερθεί ότι όλα τα εργαλεία που αναφέρονται παρακάτω, εγκαταστάθηκαν αρχικά στην φάση υλοποίησης στον τοπικό server του οικιακού δικτύου, σε πλατφόρμα λογισμικού Mac OS X 10.9.2. Σε δεύτερη φάση, αυτήν της τελικής υλοποίησης του REST Webserver, τα εργαλεία έγιναν εγκατάσταση σε Ubuntu Server 12.04 LTS, όπως αναφέρθηκε νωρίτερα [4.7.2], ακολουθώντας μία διαδικασία με συγκεκριμένα βήματα που περιγράφεται σε πολλές ιστοσελίδες στο διαδίκτυο [50]. Τα συγκεκριμένα εργαλεία αναφέρονται συνήθως με τον όρο (L/M)AMP, που προέρχεται από τα αρχικά τους, Apache, MySQL, PHP και Linux/Macintosh ως προς το λειτουργικό σύστημα.

4.8.1 Apache HTTP Server



Εικόνα 4.8 – Λογότυπο Apache HTTP Server

Για την υλοποίηση του back-end μέρους της εφαρμογής, της παρούσας εργασίας, χρησιμοποιήθηκε το εργαλείο **Apache HTTP Server** [52]. Πρόκειται για έναν εξυπηρετητή διαδικτύου (web server), που εφαρμόζει το HTTP πρωτόκολλο. Για παράδειγμα, κάθε φορά που ένας χρήστης επισκέπτεται μία σελίδα, το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου HTTP, ο οποίος με την σειρά του παράγει τις σελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Πρόκειται για έναν από τους πιο δημοφιλείς και πιο διαδεδομένους εξυπηρετητές ιστού στον κόσμο, γεγονός που οφείλεται κατά έναν τρόπο στο ότι λειτουργεί σε διάφορες πλατφόρμες όπως Windows, Linux, Unix και Mac OS X. Επίσης αποτελεί λογισμικό ανοιχτού κώδικα και αναπτύχθηκε από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation, ASF). Ο Apache όμως, χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων. Με αυτόν τον τρόπο ακριβώς χρησιμοποιείται και στην παρούσα εργασία. Η έκδοση που εγκαταστάθηκε ήταν η 2.2.26 του Apache HTTP Server.

4.8.2 PHP



Εικόνα 4.9 – Λογότυπο PHP

Επίσης αξίζει αναφοράς το γεγονός ότι χρησιμοποιήθηκε η έκδοση 5.4.24 για την γλώσσα **PHP** [47]. Στη παρούσα εργασία αποτέλεσε την κύρια γλώσσα προγραμματισμού για την δημιουργία της σύνδεσης μεταξύ εξυπηρετητή (server) και πελάτη (client). Συγκεκριμένα μέσω της PHP πραγματοποιούνται οι HTTP αιτήσεις (requests) από τον πελάτη (iPhone app, S.M.C.), επεξεργάζονται και στη συνέχεια γίνονται οι αντίστοιχες ερωτήσεις (queries) στην βάση δεδομένων MySQL. Τελικώς, ο εξυπηρετητής, ανταποκρίνεται στον πελάτη επιστρέφοντας απαντήσεις (responses) σε μορφή JSON, είτε τα δεδομένα που έχουν ζητηθεί, είτε την ανάλογη απάντηση αν έχει προκύψει κάποιο σφάλμα.

4.8.3 MySQL Community Server



Εικόνα 4.10 – Λογότυπο MySQL Server

Για την αποθήκευση των δεδομένων χρησιμοποιήθηκε βάση δεδομένων τύπου **MySQL** [48]. Όπως περιγράφηκε νωρίτερα η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Το εργαλείο το οποίο χρησιμοποιήθηκε για το χειρισμό της βάσης ήταν ο MySQL Community Server. Η έκδοση που εγκαταστάθηκε ήταν η 5.6.14.

4.8.4 Navicat Premium



Εικόνα 4.11 – Λογότυπο Navicat Premium

Για την διαχείριση της βάσης δεδομένων χρησιμοποιήθηκε το εργαλείο **Navicat Premium** [53]. Το εργαλείο αυτό αποτελεί ένα αυτόνομο πρόγραμμα client-based, δηλαδή εκτελείται και αφορά το περιβάλλον του πελάτη (περιηγητή), το οποίο πραγματοποιεί την σύνδεση με τοπικό ή απομακρυσμένο εξυπηρετητή (server) MySQL. Ουσιαστικά αποτελεί ένα γραφικό περιβάλλον για τη διαχείριση της βάσης δεδομένων. Στην παρούσα εργασία χρησιμοποιείται για τη δημιουργία και κυρίως για την διαχείριση (προβολή - τροποποίηση) των δύο βάσεων δεδομένων με τους χώρους-δωμάτια, τους αισθητήρες και τα δεδομένα-μετρήσεις τους, που βρίσκονται στο κτίριο Τεχνικών Υπηρεσιών στο Πολυτεχνείο Κρήτης (Χανιά). Η έκδοση που εγκαταστάθηκε ήταν η Premium 11.0.10.

5. ΑΡΧΕΣ ΣΧΕΔΙΑΣΗΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Η σωστή σχεδίαση του interface μίας εφαρμογής αποτελεί βασική προϋπόθεση ώστε αυτή να θεωρείται επιτυχημένη. Σημαντικό είναι να μπορεί να συνδυάζει ελκυστικότητα και ευχρηστία. Να παροτρύνει και όχι να αποθαρρύνει τον χρήστη να την χρησιμοποιήσει. Για να επιτευχθεί αυτό κρίνεται αναγκαίο να ακολουθηθούν κάποιες βασικές αρχές σχεδίασης από την επικοινωνία Ανθρώπου-Υπολογιστή (Human-Computer Interaction, HCI). Σημαντική είναι εξίσου και η δομή του γραφικού περιβάλλοντος χρήσης, καθώς πρέπει να δίνεται η αίσθηση της αρτιότητας και να διευκολύνεται ο χρήστης κατά τον χειρισμό.

Η συγκεκριμένη εφαρμογή που δημιουργήθηκε στα πλαίσια της παρούσας εργασίας για πλατφόρμα κινητού τηλεφώνου, βασίστηκε κατά κύριο λόγο στις σχεδιαστικές αρχές της ήδη υπάρχουσας web εφαρμογής [2.2]. Κατά την υλοποίησή της, είναι προφανές ότι έγιναν οι απαραίτητες προσθήκες και βελτιώσεις ώστε να ανταποκρίνεται πλήρως στις δυνατότητες και τις προδιαγραφές ενός κινητού τηλεφώνου²¹. Παράλληλα, αξίζει αναφοράς ότι η εφαρμογή, σε πρώτη φάση τουλάχιστον, δεν απευθύνεται σε απλούς χρήστες, αλλά χρήστες που έχουν γνώση του αντικειμένου και κυρίως χρήστες που είναι εξοικειωμένοι με την τεχνολογία των smartphones.

5.1 Βασική Δομή των Οθονών του User Interface

Σε κάθε native mobile εφαρμογή η σχεδίαση αποτελείται από «οθόνες²²», δηλαδή γραφικές απεικονίσεις των λειτουργιών της (GUI), και όχι από σελίδες όπως αναφέρονται οι web εφαρμογές. Η συγκεκριμένη εφαρμογή, λοιπόν, αποτελείται από πληθώρα οθονών όπου κάθε μία είναι υπεύθυνη για την προβολή όλων των διαθέσιμων πληροφοριών και δυνατοτήτων της στον χρήστη. Επομένως, υπάρχουν κάποια βασικά templates, αλλά η οθόνη που έχει το μεγαλύτερο ενδιαφέρον είναι αυτή στην οποία προβάλλονται κάθε φορά τα δωμάτια-χώροι (ζώνες), οι αισθητήρες που υπάρχουν σε αυτά αλλά και τα σήματα ελέγχου (actuators) του air-condition κάθε χώρου. Το βασικό

²¹ Ένα παράδειγμα είναι ότι το μικρότερο μέγεθος οθόνης επέβαλε αλλαγές στην δομή της σελίδας σύνδεσης του χρήστη.

²² Ουσιαστικά είναι τα Views όπως αυτά αναφέρονται και στο μοντέλο MVC.

template της οθόνης αυτής παρουσιάζεται στην *Εικόνα 5.1*. Όπως φαίνεται, λοιπόν, αποτελείται από πέντε βασικά επιμέρους τμήματα.

Στην κορυφή υπάρχει η μπάρα πλοήγησης (Navigation Bar) όπου στο κέντρο τοποθετείται κάθε φορά ο αντίστοιχος τίτλος, όπως για παράδειγμα το όνομα του δωματίου (ζώνης) ή του σήματος που προβάλλεται. Στα αριστερά υπάρχει ένα κουμπί που βοηθάει τον χρήστη να γυρίσει ένα επίπεδο πίσω, δηλαδή να γυρίσει στην οθόνη που βρέθηκε την τελευταία φορά. Αυτό το κουμπί είναι προφανές ότι είναι διαθέσιμο εάν και εφόσον έχει υπάρξει πλοήγηση από κάποια άλλη οθόνη. Επίσης υπάρχει ένα κουμπί στα δεξιά όπου αποτελεί το «**Home Button**», οδηγεί στην αρχική οθόνη, αλλά και το κουμπί των ρυθμίσεων, **Settings**, που προσφέρει ένα μενού με τρεις επιλογές. Αυτές οι επιλογές είναι:

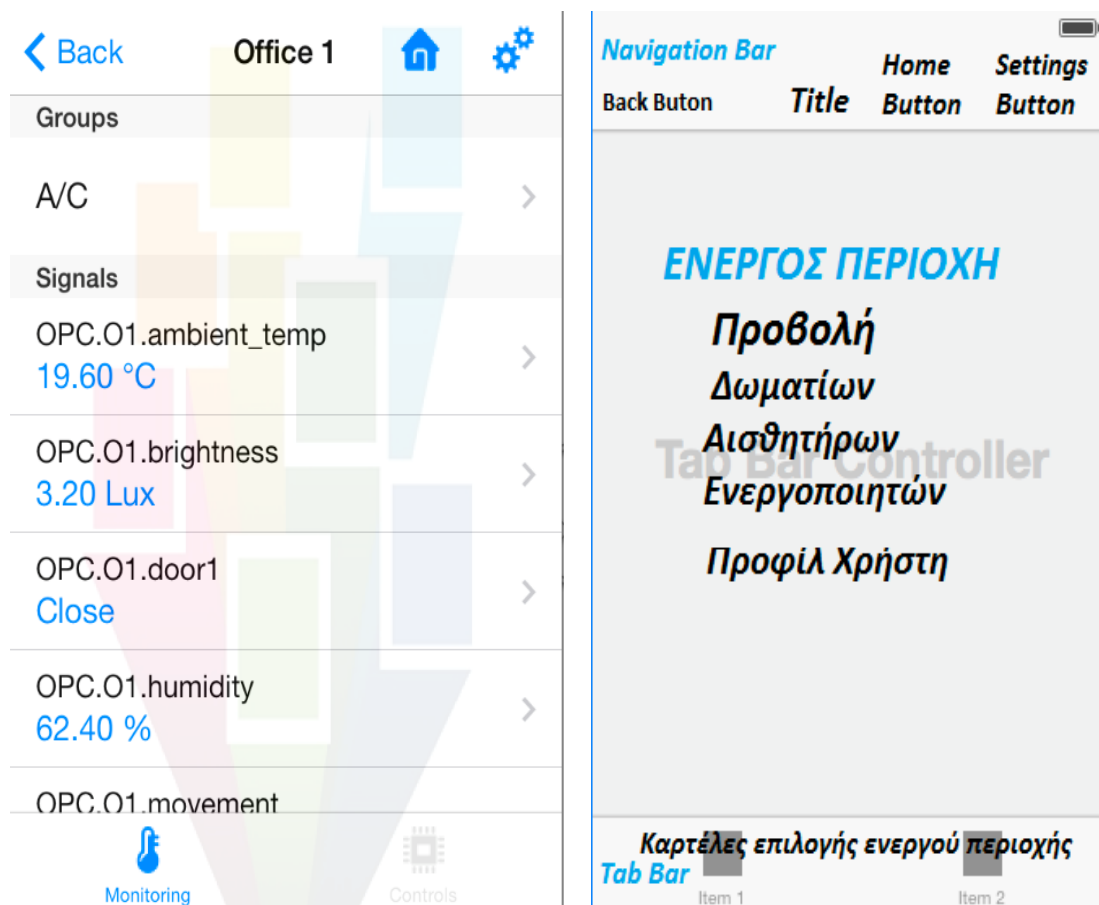
Το **Edit Profile**, που οδηγεί στην οθόνη του προφίλ (λογαριασμού) του χρήστη.

Το **About**, που οδηγεί στην οθόνη των πληροφοριών της εφαρμογής. Συγκεκριμένα ο χρήστης ενημερώνεται για του όρους χρήσης, την πολιτική απορρήτου, την έκδοση της εφαρμογής, τον δημιουργό της εφαρμογής αλλά και πιθανές ερωτήσεις σχετικά με άγνωστους όρους της εφαρμογής, ενώ δίνεται η δυνατότητα περιήγησης στην επίσημη ιστοσελίδα του προγράμματος P.E.B.B.L.E. και άλλων ιστοσελίδων που επιθυμεί.

Και τέλος το **Logout**, όπου ο χρήστης έχει την δυνατότητα να αποσυνδεθεί από τον λογαριασμό του και παραπέμπεται στην αρχική οθόνη.

Η ενεργός περιοχή, δηλαδή το κεντρικό και μεγαλύτερο μέρος της οθόνης περιλαμβάνει τα ονόματα των δωματίων (ζωνών), των αισθητήρων και των σημάτων ελέγχου. Τέλος, η περιοχή στο κάτω μέρος της οθόνης, η μπάρα καρτελών (Tab Bar), περιλαμβάνει δύο καρτέλες που δίνουν στον χρήστη την δυνατότητα να επιλέξει είτε να προβληθούν τα σήματα και οι τιμές τους είτε να προβληθούν οι διακόπτες ελέγχου που διαθέτει το συγκεκριμένο δωμάτιο-χώρος, στην ενεργό περιοχή.

Οι υπόλοιπες οθόνες διαθέτουν το ίδιο template με κάποιες προσθήκες ή αφαιρέσεις ως προς τις μπάρες πλοήγησης και καρτελών, αλλά υπάρχουν και οθόνες που έχουν διαφορετικό template στατικής δομής, όπως αυτό της αρχική οθόνης της εφαρμογής. Παραδείγματα φαίνονται στην *Εικόνα 5.2*.

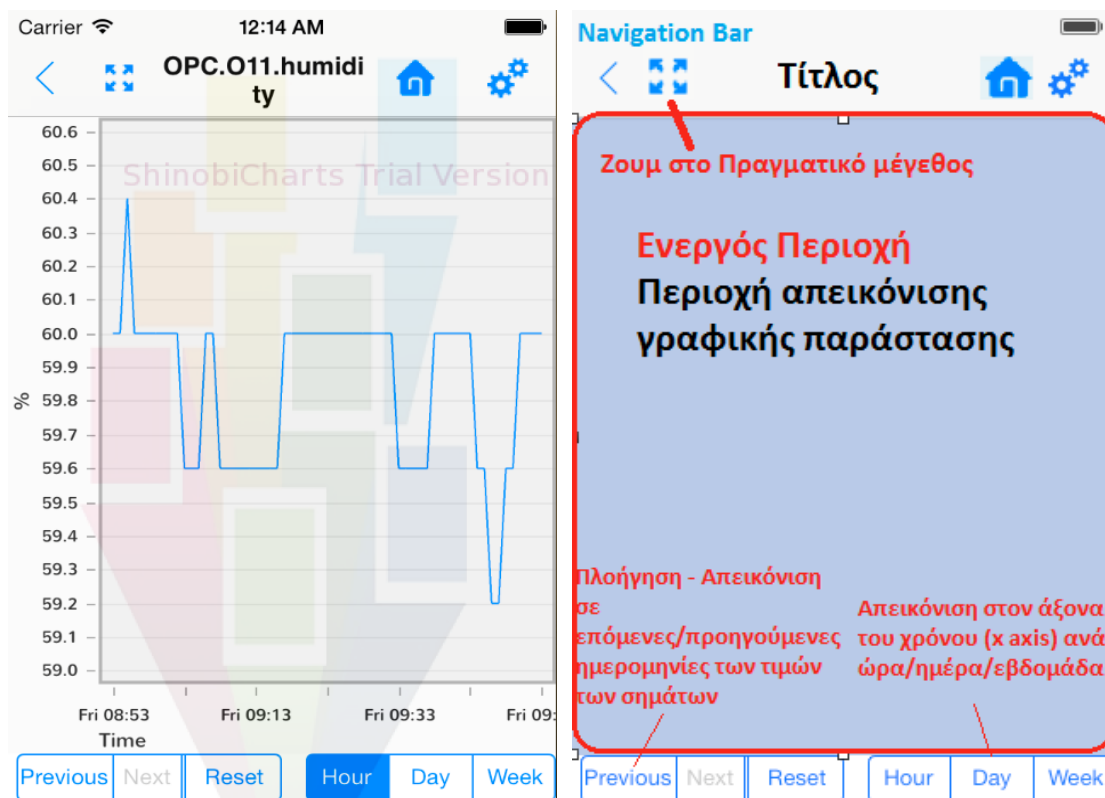


Εικόνα 5.1 – *Αριστερά*. Παράδειγμα εμφάνισης Οθόνης για τον χώρο Office 1 – *Δεξιά*. Template Οθόνης περιήγησης στα δωμάτια-ζώνες, τους αισθητήρες, τα σήματα ελέγχου, το διάγραμμα κάθε σήματος



Εικόνα 5.2 – Πάνω Αριστερά. Αρχική Οθόνη που εμφανίζεται μόλις ξεκινήσει η εφαρμογή – Πάνω Δεξιά. Οθόνη σύνδεσης του χρήστη στο σύστημα. – Κάτω Αριστερά. Οθόνη απεικόνισης λειτουργιών για περισσότερες πληροφορίες που αφορούν την σχεδίαση και υλοποίηση της εφαρμογής. – Κάτω Δεξιά. Οθόνη απεικόνισης πληροφοριών σχετικών με την υλοποίηση της εφαρμογής.

Άξιο αναφοράς είναι το template της οθόνης απεικόνισης των γραφικών παραστάσεων σημάτων – αισθητήρων. Συγκεκριμένα, όπως παρουσιάζεται στην *Εικόνα 5.3* (δεξιά), αποτελείται από την μπάρα πλοήγησης που είναι ακριβώς η ίδια με αυτή που περιγράφηκε νωρίτερα, έχοντας επιπλέον ένα κουμπί στα αριστερά. Το συγκεκριμένο κουμπί αφορά το διάγραμμα και παρέχει την δυνατότητα επαναφοράς του ποσοστού ζουμ που έχει γίνει στο διάγραμμα, σε μηδενικά επίπεδα και ουσιαστικά στο αρχικό-πραγματικό μέγεθος. Επίσης υπάρχουν στο κάτω μέρος της οθόνης δύο ομάδες κουμπιών επιλογής τρόπου απεικόνισης του διαγράμματος. Στην αριστερή μεριά, οι διακόπτες αφορούν την εμφάνιση του διαγράμματος κάθε σήματος σε προηγούμενα ή επόμενα χρονικά διαστήματα, τέτοια που ορίζονται από τους διακόπτες στην δεξιά μεριά. Ουσιαστικά στην δεξιά μεριά είναι οι διακόπτες που δίνουν την δυνατότητα εμφάνισης του διαγράμματος κάθε σήματος ως προς τον άξονα του χρόνου (x axis) για τις τιμές του σε διάρκεια μίας ώρας ή μίας ημέρας ή μίας εβδομάδας. Σημειώνεται ότι η τελευταία επιλογή, της εβδομάδας, υλοποιήθηκε στην παρούσα εφαρμογή για τον λόγο ύπαρξής της και στην ήδη υπάρχουσα web εφαρμογή. Όμως, λόγω της μικρής οθόνης του κινητού τηλεφώνου, σε κάποιες περιπτώσεις δεν είναι τόσο λειτουργική²³.



Εικόνα 5.3 – Αριστερά. Παράδειγμα από την πραγματική απεικόνιση διαγράμματος ενός αισθητήρα υγρασίας. – Δεξιά. Το Template Οθόνης απεικόνισης γραφικών παραστάσεων σημάτων.

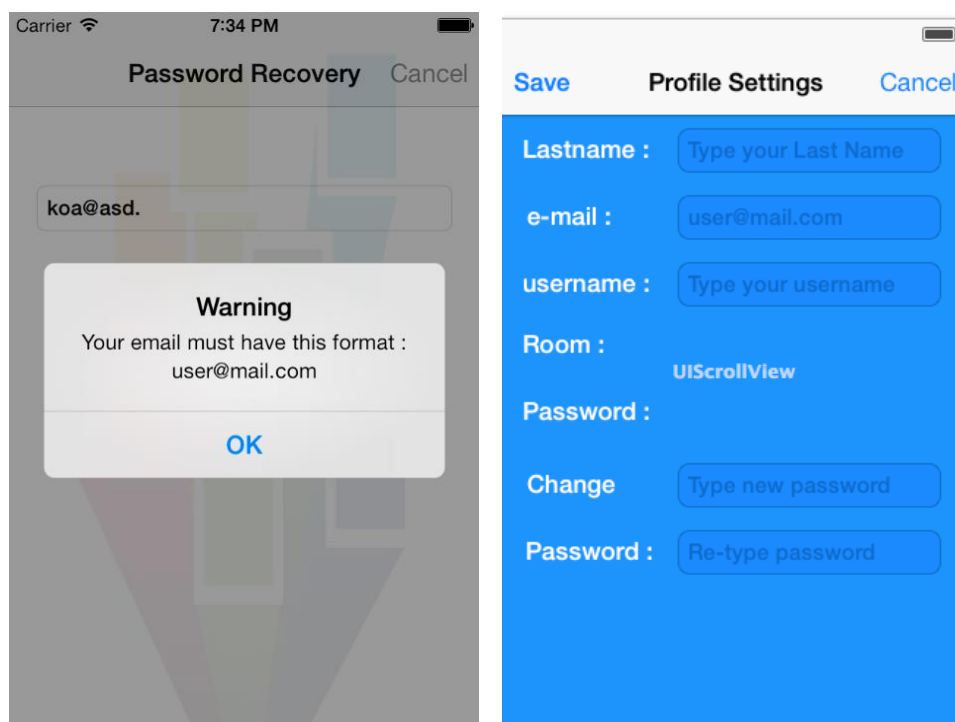
²³ Απαιτεί μεγάλο ποσοστό ζουμ για να απεικονιστεί καλύτερα το διάγραμμα.

5.2 Ενημέρωση Χρήστη

Στην σχεδίαση της εφαρμογής δόθηκε μεγάλη έμφαση στην ανάγκη που έχει κάθε χρήστης, ώστε να γνωρίζει κάθε στιγμή που ακριβώς βρίσκεται μέσα στο σύστημα. Συγκεκριμένα, όπως αναφέρθηκε στην προηγούμενη παράγραφο, το πάνω μέρος της εφαρμογής, η μπάρα πλοήγησης, διαθέτει το μέρος του τίτλου όπου φαίνεται ακριβώς σε ποια θέση βρίσκεται ο χρήστης. Επίσης, στην περίπτωση προβολής δωματίων-χώρων ή μονάδων (κλιματιστικών), υπάρχει στο κάτω μέρος της οθόνης η μπάρα καρτελών όπου τηρείται η τακτική να φωτίζεται η επιλεγμένη καρτέλα. Ένα απλό παράδειγμα φαίνεται στην *Εικόνα 5.1 (Αριστερά)* όπου ο χρήστης έχει επιλέξει να προβάλλει το δωμάτιο (ζώνη) με την ονομασία «Office 1» και συγκεκριμένα να δει τα σήματα που περιέχονται στο συγκεκριμένο δωμάτιο (ζώνη), με αποτέλεσμα να χρωματίζεται με μπλε χρώμα η καρτέλα «monitoring». Έτσι ο κανόνας²⁴ που προβλέπει ότι ο χρήστης πρέπει να μπορεί να δει την θέση του στην εφαρμογή για να μην χάνει την αυτοπεποίθηση του ακολουθείται πιστά.

Ο χρήστης επίσης έχει την δυνατότητα κατά την συμπλήρωση διαφόρων πεδίων να ενημερώνεται. Πάντα υπάρχει μέσα σε όλα τα πεδία συμπλήρωσης μία «placeholder» φράση που ενημερώνει για την μορφή του κειμένου που απαιτείται. Σε κάποιες περιπτώσεις, όπως αυτή της συμπλήρωσης κάποιου email, ο χρήστης ενημερώνεται και με ένα μήνυμα σε περίπτωση λανθασμένης εισαγωγής. Τέλος, ο χρήστης ενημερώνεται πάντα για επιτυχή ή ανεπιτυχή έκβαση των ενεργειών του με χρήση κατάλληλων (popup) μηνυμάτων. Παραδείγματα φαίνονται παρακάτω στην *Εικόνα 5.4*.

²⁴ Κανόνας No. 1 του Jacob Nielsen. Περισσότερα στο www.nngroup.com/articles/ten-usability-heuristics



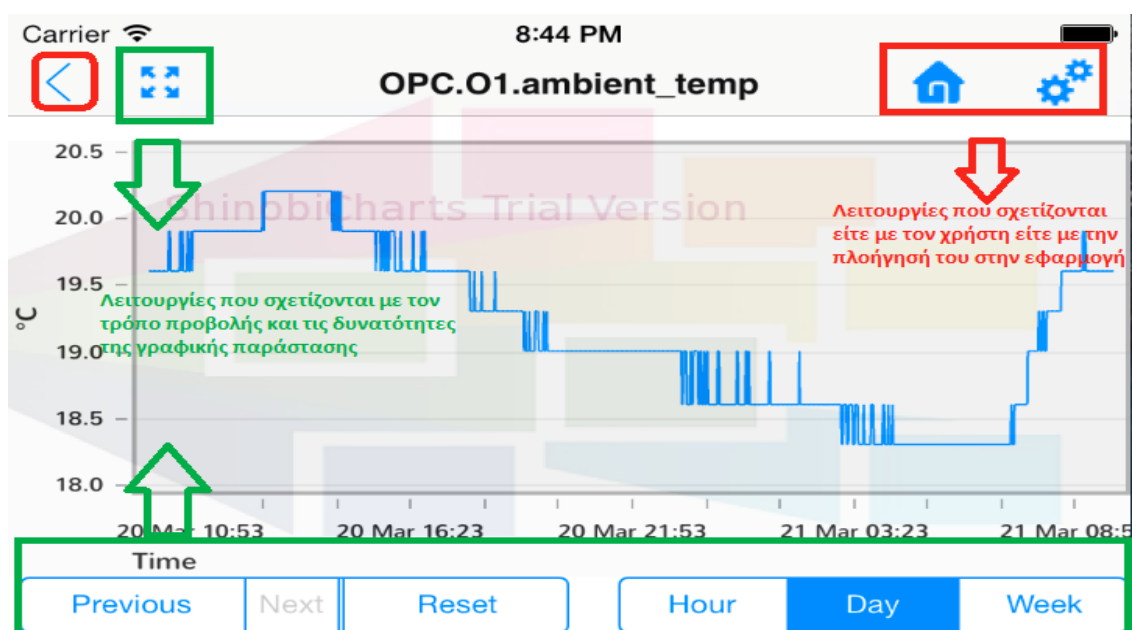
Εικόνα 5.4 – Αριστερά. Παράδειγμα ενημέρωσης χρήστη με εμφάνιση popup μηνύματος λανθασμένης συμπλήρωσης email. – Δεξιά. Παράδειγμα ενημέρωσης χρήστη με «placeholder» φράσεις.

5.3 Καθοδήγηση Χρήστη και Ευκολία Πλοήγησης

Το γραφικό περιβάλλον χρήστη (GUI) που σχεδιάστηκε στην παρούσα εφαρμογή όχι μόνο βοηθάει, αλλά καθοδηγεί πλήρως τον χρήστη στην πραγματοποίηση διαφόρων ενεργειών. Προσφέρεται, λοιπόν, στον χρήστη η δυνατότητα γρήγορης και εύκολης πλοήγησης στις οθόνες, μέσω διαφόρων κουμπιών και σημάτων που αποτελούν συνδέσμους προς τις επόμενες οθόνες. Συγκεκριμένα, ένας χρήστης που θέλει αλλάξει τον κωδικό πρόσβασης, επιλέγει το κουμπί των ρυθμίσεων και οδηγείται στην οθόνη των στοιχείων του λογαριασμού του. Εφόσον αλλάξει τον κωδικό με επιτυχή τρόπο, τότε ο νέος κωδικός αποθηκεύεται και οδηγείται ο χρήστης στην οθόνη από την οποία βρέθηκε στον λογαριασμό του. Επίσης ο σχεδιασμός του User Interface έγινε λαμβάνοντας υπόψη ότι κάποιοι χρήστες ενδέχεται να μην είναι εξοικειωμένοι με το κινητό τηλέφωνο και το διαδίκτυο γενικότερα. Έτσι έχει γίνει χρήση εικονιδίων στα αντίστοιχα κουμπιά που είναι πλήρως αντιπροσωπευτικά των λειτουργιών που προσφέρουν.

5.4 Ομαδοποίηση Λειτουργιών

Η ομαδοποιημένη λειτουργικότητα αποτελεί μία βασική απαίτηση για web εφαρμογές, αλλά και εφαρμογές που αναφέρονται σε πλατφόρμες κινητών τηλεφώνων. Έτσι η παρούσα εφαρμογή στην προσπάθεια διευκόλυνσης του χρήστη, εφαρμόζει την ομαδοποίηση των λειτουργιών όπου αυτό είναι εφικτό. Συγκεκριμένα, όπως φαίνεται παρακάτω στην *Εικόνα 5.5*, στη απεικόνιση του διαγράμματος κάθε σήματος, υπάρχουν ομαδοποιημένες ενέργειες που αφορούν τις λειτουργίες και τις δυνατότητες που παρέχει η εφαρμογή σε πράσινο πλαίσιο. Παράλληλα, σε κόκκινο πλαίσιο παρουσιάζονται οι λειτουργίες που αφορούν την γενικότερη πλοήγηση της εφαρμογής και των ενεργειών που αφορούν τον ίδιο τον χρήστη. Αυτό έχει αποτέλεσμα ο χρήστης να μην βρίσκεται σε σύγχυση, αλλά να πλοηγείται γρήγορα κι άμεσα και να διατηρείται γενικότερα μία τάξη στην οθόνη της εφαρμογής.



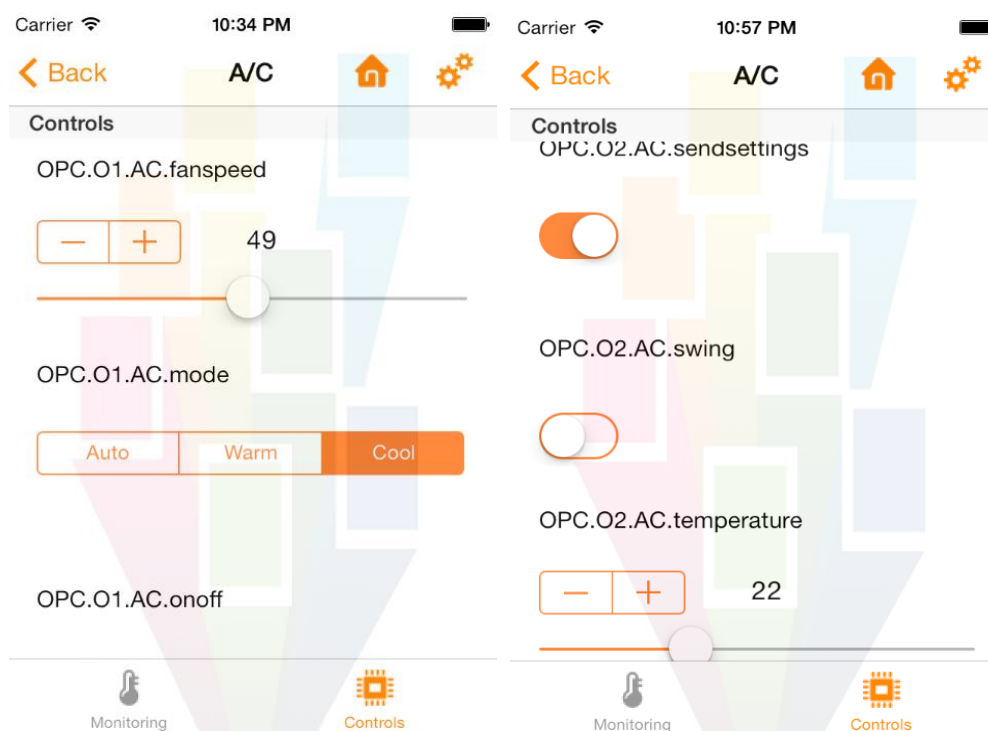
Εικόνα 5.5 – Ομαδοποίηση λειτουργιών.

5.5 Επιλογή Χρωμάτων και Συνέπεια

Ένας ακόμη σημαντικός παράγοντας για την ορθή υλοποίηση του User Interface είναι αποτελεί η κατάλληλη χρήση χρωμάτων. Σκοπός της σωστής επιλογής των χρωμάτων είναι να γίνει το interface περισσότερο φιλικό, ευχάριστο και αποδοτικό προς τον χρήστη, αλλά και να καθιστά την πλοήγηση στην εφαρμογή εύκολη και γρήγορη. Έτσι προτιμήθηκε η επιλογή των βασικών χρωμάτων, μαύρο, άσπρο και μπλε για το φόντο,

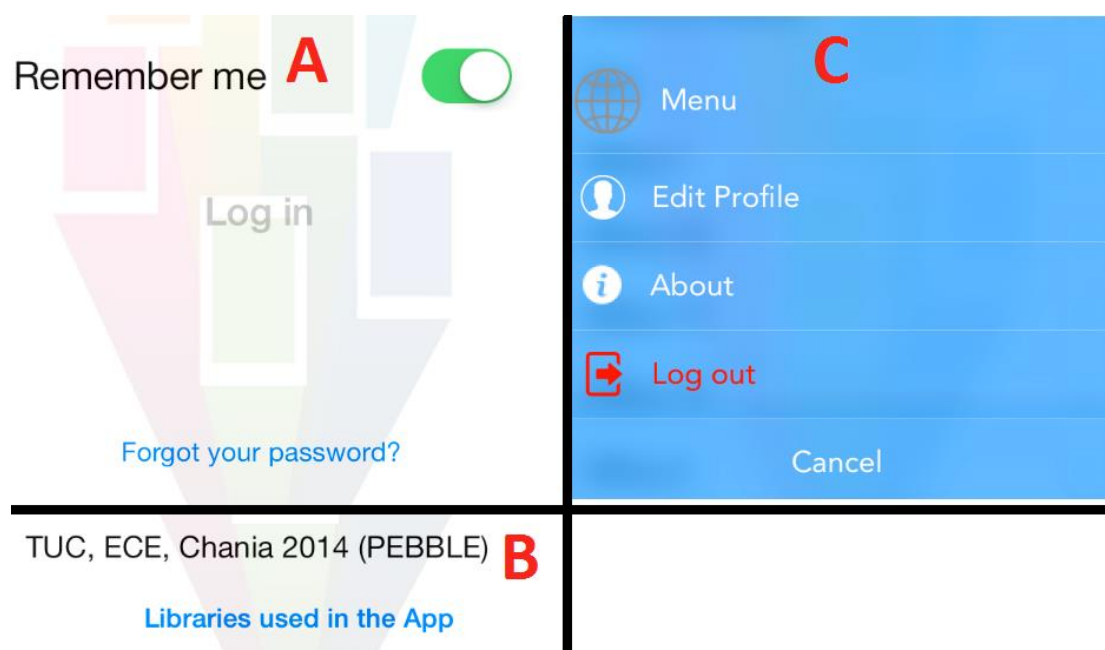
τους διακόπτες, τα ονόματα των σημάτων και των ζωνών. Επίσης, διαφορετικές αποχρώσεις του μπλε, όπως το γαλάζιο χρώμα, επιλέχθηκαν στην περίπτωση της τιμής και μονάδας μέτρησης των σημάτων. Στην οθόνη όπου εμφανίζονται οι πληροφορίες για τον λογαριασμό του χρήστη προτιμήθηκε στο φόντο το μπλε χρώμα, σε αντίθεση με τις υπόλοιπες που κυριαρχεί το άσπρο, ώστε να δοθεί έμφαση στο γεγονός ότι η συγκεκριμένη οθόνη έχει να κάνει αποκλειστικά και μόνο με τον χρήστη και τον λογαριασμό του (Εικόνα 5.4). Αυτά τα χρώματα είναι ευχάριστα και ξεκουράζουν το ανθρώπινο μάτι, ενώ ο συνδυασμός τους παρουσιάζει αντίθεση, με αποτέλεσμα να διακρίνονται τα διαφορετικά τμήματα μεταξύ τους. Χαρακτηριστικό παράδειγμα φαίνεται στην Εικόνα 5.1 (Αριστερά).

Επίσης στην περίπτωση της οθόνης που περιέχει τα σήματα ελέγχου των αισθητήρων (air-condition) προτιμήθηκε να γίνει επιλογή ενός διαφορετικού χρώματος και συγκεκριμένα του πορτοκαλί. Προτιμήθηκε διαφορετικό χρώμα ώστε να ξεχωρίζει ουσιαστικά το είδος των ενεργειών που μπορεί να πραγματοποιήσει ο χρήστης στην συγκεκριμένη οθόνη. Επίσης το συγκεκριμένο χρώμα είναι ευχάριστο και ευδιάκριτο σε συνδυασμό με το άσπρο χρώμα του φόντου και το μαύρο χρώμα των ονομάτων των ενεργοποιητών (actuators) όπως φαίνεται στην Εικόνα 5.6.



Εικόνα 5.6 – Παράδειγμα χρήσης πορτοκαλί χρώματος στην οθόνη με σήματα (διακόπτες) ελέγχου.

Τέλος σε περιπτώσεις όπου υπάρχουν σύνδεσμοι (links) προτιμήθηκε το γαλάζιο χρώμα ώστε να αναγνωρίζονται εύκολα και γρήγορα από τον χρήστη, ενώ για την επισήμανση της ενέργειας αποσύνδεσης του χρήστη προτιμήθηκε το κόκκινο χρώμα. Παραδείγματα φαίνονται στην *Εικόνα 5.7*.



Εικόνα 5.7 – (A,B) Παράδειγμα συνδέσμων. Χρήση γαλάζιου χρώματος. – (C) Παράδειγμα ενέργειας-κουμπιού για αποσύνδεση χρήστη. Χρήση κόκκινου χρώματος.

Παράλληλα το User Interface της εφαρμογής σχεδιάστηκε έτσι ώστε να τηρεί τις αρχές που αφορούν την συνέπεια. Το γεγονός επιλογής του μπλε χρώματος στις περιπτώσεις των συνδέσμων (links) που αναφέρθηκε προηγουμένως, συμβάλει στην συνέπεια. Ένα επιπλέον χαρακτηριστικό παράδειγμα είναι ότι οι ενέργειες που μπορεί να πραγματοποιήσει ο χρήστης, όπως αυτή της μετάβασης στην αρχική σελίδα ή αυτή της αποσύνδεσής του από το σύστημα ή της πλοήγησης στα στοιχεία του λογαριασμού του, είναι πάντοτε διαθέσιμα στην μπάρα πλοήγησης ανεξάρτητα από την οθόνη στην οποία βρίσκεται ο χρήστης.

5.6 Χρήση Ενεργού Περιοχής και Χειρισμός Λαθών

Η αποδοτική και σωστή χρήση της ενεργού περιοχής της οθόνης είναι απαραίτητη. Η συγκεκριμένη περιοχή αποτελεί έναν χώρο που προβάλλει δεδομένα με δυναμικό τρόπο. Αυτό σημαίνει ότι τα δεδομένα που απεικονίζονται είναι διαφορετικά και σχετίζονται με τις επιλογές του χρήστη κάθε φορά. Η μπάρα πλοήγησης και η μπάρα καρτελών πάντα παραμένουν ίδιες και αμετάβλητες. Αυτό εξασφαλίζει ότι τα δεδομένα που προβάλλονται στην ενεργό περιοχή της οθόνης εμφανίζονται σε αυτήν και μόνο την περιοχή. Δεν είναι εφικτό, δηλαδή, να παρουσιαστούν αυτά τα δεδομένα στο κάτω ή στο πάνω μέρος της οθόνης.

Επίσης ο χειρισμός των λαθών αποτελεί ένα απαραίτητο στοιχείο που πρέπει να διαθέτει ένα καλά σχεδιασμένο user interface. Συγκεκριμένα, στην παρούσα εφαρμογή γίνονται οι απαραίτητες ενέργειες όπου αυτό απαιτείται, ώστε να αντιμετωπίζονται τα λάθη που κάνει ο χρήστης ή ακόμα και να γίνεται πρόβλεψη των λαθών που ενδέχεται να κάνει ο χρήστης. Ως προς την πρόβλεψη, ένα χαρακτηριστικό παράδειγμα είναι ότι στην περίπτωση της απεικόνισης του διαγράμματος κάποιου σήματος, η επιλογή συγκεκριμένης ημερομηνίας δεν διατίθεται παρά μόνο από τα κουμπιά previous/next και hour/day/week. Επίσης όπως αναφέρθηκε και παραπάνω, η χρήση κατάλληλων μηνυμάτων σε περίπτωση λάθους από τον χρήστη συντελεί στην αντιμετώπισή τους.

5.7 Χρήση και διάθεση εφαρμογής σε δύο γλώσσες

Η παρούσα εφαρμογή αρχικά σχεδιάστηκε και υλοποιήθηκε, όπως φαίνεται και σε όλες τις εικόνες, στα Αγγλικά που αποτελούν την επίσημη διεθνή γλώσσα. Στην συνέχεια έγινε μετάφραση στην Ελληνική γλώσσα, κυρίως για λόγους προσωπικούς του συγγραφέα της παρούσας εργασίας αλλά και λόγους που αφορούν το γεγονός ότι η παρούσα εργασία εκπονήθηκε, σχεδιάστηκε, υλοποιήθηκε και παρουσιάστηκε στην Ελλάδα. Επομένως, ανάλογα με την γλώσσα που επιλέγει ο χρήστης στο κινητό τηλέφωνο θα έχει και την ανάλογη έκδοση, ενώ είναι προφανές ότι ρύθμιση του κινητού τηλεφώνου σε οποιαδήποτε άλλη γλώσσα εκτός των δύο προαναφερθέντων, ενεργοποιεί την έκδοση με την γλώσσα στα Αγγλικά.

5.8 Συμβατότητα με άλλες Συσκευές της Apple

Η εφαρμογή αναπτύχθηκε, όπως έχει ήδη αναφερθεί, σε λειτουργικό σύστημα iOS 7.0.4 και στην συσκευή iPhone 4, με αποτέλεσμα να έχει την δυνατότητα να «τρέχει» σε οποιαδήποτε άλλη συσκευή που διαθέτει το συγκεκριμένο λογισμικό. Όμως, έγινε κατάλληλη σχεδίαση και προσαρμογή στις απαιτήσεις της συσκευής iPad. Συγκεκριμένα, προσαρμόστηκε κατάλληλα ολόκληρο το user interface ώστε να ανταποκρίνεται στην μεγαλύτερη οθόνη και ανάλυση της συσκευής αυτής και εν συνεχεία ελέγχθηκε στον προσομοιωτή του εργαλείου Xcode, για επαλήθευση της σωστής σχεδίασης και λειτουργίας. Έτσι είναι διαθέσιμη για εγκατάσταση σε όλες της συσκευές της Apple, όπως iPod touch και iPad αλλά και τις άλλες εκδόσεις των συσκευών iPhone, 3GS, 4s, 5, 5c και 5s. Μια μικρή επισήμανση αξίζει να γίνει στο παρόν σημείο ως προς τη συσκευή του iPad που διαθέτει οθόνη Retina και των iPhones 5, 5c, 5s, καθώς για την άρτια λειτουργία της εφαρμογής χρειάζονται ελάχιστες τροποποιήσεις λόγω της υψηλότερης ανάλυσης και του μεγαλύτερου μεγέθους²⁵ αντίστοιχα, της οθόνης. Συγκεκριμένα, απαιτούνται μετατροπές μόνο σε κάποιες εικόνες και κάποια εικονίδια σε μεγαλύτερο μέγεθος.

Γενικότερα, για άλλη μια φορά τονίζεται ότι οι αρχές σχεδίασης, ανάπτυξης και υλοποίησης της εφαρμογής **S.M.C.**, έχουν γίνει με κατάλληλο τρόπο, ώστε να γίνεται εύκολα η προσαρμογή της σε οποιεσδήποτε άλλες «συνθήκες». Τέτοιες συνθήκες αφορούν είτε άλλες πλατφόρμες, είτε την προσαρμογή της στα πλαίσια άλλων προγραμμάτων και συστημάτων διαχείρισης και εξοικονόμησης ενέργειας σε κτίρια, BEMS.

²⁵ iPhone 3GS, 4, 4s : 3,5 inches – iPhone 5, 5s, 5c : 4 inches

6. ΑΠΟΤΕΛΕΣΜΑΤΑ ΧΡΗΣΗΣ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

6.1 Εισαγωγή - Λειτουργικότητα Εφαρμογής

Στο συγκεκριμένο κεφάλαιο παρουσιάζονται τα αποτελέσματα από την χρήση της εφαρμογής, **S.M.C**, που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας. Ουσιαστικά τα αποτελέσματα φαίνονται και περιγράφονται μέσα από την παρουσίαση και απεικόνιση των λειτουργιών της εφαρμογής. Επομένως, στην συνέχεια ακολουθεί αναλυτική περιγραφή και απεικόνιση όλων των λειτουργιών της και η υλοποίησή της στην πράξη. Ένα απλό παράδειγμα χρήσης της εφαρμογής είναι ότι κάποιος χρήστης που διαθέτει ένα γραφείο στο κτίριο Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης και δεν βρίσκεται σε αυτό, έχει την δυνατότητα να συνδεθεί, να παρακολουθήσει την θερμοκρασία κι αν αυτή είναι υψηλή, να ενεργοποιήσει το air-condition σε συγκεκριμένους βαθμούς, ώστε να δημιουργήσει καλύτερες συνθήκες. Βέβαια, η συγκεκριμένη εφαρμογή υλοποιήθηκε στα πλαίσια του προγράμματος PEBBLE, γεγονός που την καθιστά χρήσιμο εργαλείο για την γενικότερη προσαρμογή των συνθηκών ενός κτιρίου και την επίτευξη όσο δυνατόν λιγότερης κατανάλωσης ενέργειας, μέσω πλήρους αυτοματοποίησης, μελλοντικά, όλων των ηλεκτρικών και ηλεκτρονικών μηχανισμών που διαθέτουν.

6.2 Λειτουργία Εφαρμογής σε Πραγματικό Χρόνο

Σε αυτό το σημείο είναι σημαντικό να επισημανθεί ότι η εφαρμογή **S.M.C.**, όπως έχει αναφερθεί στα προηγούμενα κεφάλαια, σχεδιάστηκε και αναπτύχθηκε με την προϋπόθεση να υπάρχει πρόσβαση στον πραγματικό server του Πολυτεχνείου Κρήτης, δηλαδή στις βάσεις δεδομένων και γενικότερα στο σύστημα που έχει εγκατασταθεί στο κτίριο Τεχνικών Υπηρεσιών. Αρχικά, λοιπόν, σχεδιάστηκε και υλοποιήθηκε ο εξυπηρετητής, REST Webserver, που περιγράφηκε και αναλύθηκε λεπτομερώς στο 4^ο Κεφάλαιο [4.7], ώστε να δοκιμαστεί η λειτουργία της εφαρμογής **S.M.C.**, και έπειτα να γίνει η κατάλληλη τροποποίηση στις σχεδιαστικές προδιαγραφές που πληροί ο server του Πολυτεχνείου. Δυστυχώς, για διάφορους λόγους το σύστημα και ο server του Πολυτεχνείου έκλεισαν, επομένως η παρούσα εργασία και εφαρμογή συνέχισαν με

την χρήση του REST Webserver. Ο συγκεκριμένος εξυπηρετητής (server) που υλοποιήθηκε, επισημαίνεται ότι είναι συνδεδεμένος με τις βάσεις δεδομένων που περιέχουν όλες τις πληροφορίες για τους αισθητήρες και τους ενεργοποιητές του Κτιρίου Τεχνικών Υπηρεσιών του Πολυτεχνείου, αλλά και τις τιμές τους για ένα ολόκληρο έτος και συγκεκριμένα από τις 21 Μαρτίου 2013 έως και τις 21 Μαρτίου 2014. Επομένως, εύκολα καταλαβαίνει κανείς ότι δεν έχει δυνατότητα για εποπτεία και έλεγχο σε πραγματικό χρόνο²⁶. Οι τιμές των σημάτων δεν είναι σε πραγματικό χρόνο αλλά και οποιεσδήποτε αλλαγές στους διακόπτες ελέγχου δεν επηρεάζουν στην πραγματικότητα τις μονάδες κλιματισμού.

Όμως αυτό είναι εφικτό με απλές τροποποιήσεις. Συγκεκριμένα, ως προς το κομμάτι του εξυπηρετητή (server) απαιτείται περισσότερη δουλειά, καθώς πρέπει να υλοποιηθεί ένα σημαντικό μέρος του σε γλώσσα προγραμματισμού JavaScript. Αυτό οφείλεται στο γεγονός ότι χρησιμοποιείται η τεχνολογία των “web sockets” για την ανάκτηση των τιμών των σημάτων σε πραγματικό χρόνο και εν συνεχεία για την ενημέρωση των οθονών που τις εμφανίζουν²⁷. Παράλληλα, ως προς το κομμάτι της εφαρμογής **S.M.C.**, χρειάζεται μόνο μία μικρή αλλαγή της μεθόδου, η οποία είναι υπεύθυνη για την πραγματοποίηση της σύνδεσης με τον server και της ανάκτησης των δεδομένων. Η βιβλιοθήκη που διαθέτει τέτοιου είδους μέθοδο, είναι η “Socket Rocket” [54]. Αυτή προτείνεται ως καταλληλότερη καθώς χρησιμοποιήθηκε και δοκιμάστηκε επιτυχώς.

6.3 Ανάλυση Απαιτήσεων Λειτουργιών Εφαρμογής

Μια εφαρμογή τέτοια που περιγράφηκε στην παράγραφο [6.1] περιληπτικά, πρέπει να πληροί κάποιες βασικές προϋποθέσεις ως προς τις λειτουργίες που διαθέτει. Οι βασικές λειτουργίες αναλύονται στις επόμενες παραγράφους.

6.3.1 Εγγραφή στο Σύστημα

Η παρούσα εφαρμογή υλοποιήθηκε και είναι διαθέσιμη μόνο σε χρήστες που έχουν λογαριασμό στο σύστημα, δηλαδή διαθέτουν έγκυρο όνομα χρήστη και κωδικό πρόσβασης. Η ιδιαιτερότητα με την συγκεκριμένη εφαρμογή έγκειται στο γεγονός ότι η λειτουργία της εγγραφής δεν είναι διαθέσιμη. Ο λόγος είναι ότι, όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, η παρούσα εφαρμογή υλοποιήθηκε στα πλαίσια του

²⁶ Δεν είναι δυνατή η πρόσβαση σε real-time δεδομένα.

²⁷ Πρόκειται για τις οθόνες που προβάλλουν τα σήματα κάθε ζώνης αλλά και το διάγραμμα τιμών κάθε σήματος

προγράμματος PEBBLE και με βάση τις σχεδιαστικές αρχές της προϋπάρχουσας web εφαρμογής. Εξυπηρετεί και βοηθάει στην πραγματοποίηση δηλαδή, της ενεργειακής διαχείρισης κάποιων κτιρίων, άρα είναι προφανές ότι δεν υπάρχει λόγος εγγραφής οποιουδήποτε χρήστη στο σύστημα. Επομένως, ο αριθμός των χρηστών είναι περιορισμένος και απαιτείται έγκριση από τους διαχειριστές της βάσης δεδομένων, ώστε να δημιουργηθεί ένας λογαριασμός. Η λειτουργία της εγγραφής όμως παραμένει μία σημαντική λειτουργία, που εφαρμόζεται κυρίως σε κοινωνικά δίκτυα ή φόρουμ, κατηγορίες όπου αυτή η εφαρμογή δεν ανήκει, αλλά είναι πάρα πολύ εύκολο να προστεθεί μελλοντικά.

6.3.2 Είσοδος – Έξοδος στο Σύστημα

Κάθε χρήστης που διαθέτει έναν λογαριασμό στο σύστημα, έχει την δυνατότητα να συνδεθεί σε αυτόν, χρησιμοποιώντας το προσωπικό του όνομα χρήστη (username) και τον προσωπικό κωδικό (password) του. Στην προκειμένη περίπτωση το username είναι πάντα το προσωπικό email του χρήστη. Για την επιτυχή εισαγωγή του χρήστη στο σύστημα απαιτείται να είναι σωστά και τα δύο αυτά στοιχεία, καθώς σε διαφορετική περίπτωση δεν μπορεί να εισέλθει σε αυτό. Προφανώς, οποιαδήποτε στιγμή ο χρήστης είναι συνδεδεμένος στο σύστημα, έχει την δυνατότητα να αποσυνδεθεί από αυτό πατώντας απλά το κουμπί αποσύνδεσης, Logout.

Παράλληλα, έχει υλοποιηθεί και μία εναλλακτική λειτουργία εξόδου από το σύστημα. Η λειτουργία αυτή λαμβάνει υπόψη την αδράνεια του χρήστη. Συγκεκριμένα, εφόσον ο χρήστης είναι συνδεδεμένος αλλά για οποιοδήποτε λόγο δεν ασχολείται με την εφαρμογή για συγκεκριμένο χρονικό διάστημα, είτε έχει κλειδώσει το κινητό είτε ασχολείται με κάποια άλλη εφαρμογή, τότε η ίδια η εφαρμογή τον αποσυνδέει και επανέρχεται στην αρχική οθόνη. Το χρονικό διάστημα αυτό έχει επιλεγεί να είναι διάρκειας τριάντα λεπτών και υλοποιείται με χρήση της τεχνολογίας των cookies, στην μεριά του εξυπηρετητή (server). Σκοπός αυτής της λειτουργίας είναι η ασφάλεια των προσωπικών στοιχείων του χρήστη, καθώς αποφεύγεται το γεγονός, σε περίπτωση κλοπής του κινητού τηλεφώνου, να παραμένει συνδεδεμένος και να παραβιαστούν τα προσωπικά στοιχεία πρόσβασης.

6.3.3 Ανάκτηση Κωδικού

Μία πάρα πολύ σημαντική λειτουργία που παρέχει η εφαρμογή σε κάθε χρήστη που διαθέτει έναν λογαριασμό στο σύστημα, είναι αυτή της ανάκτησης του προσωπικού του κωδικού. Σε περίπτωση που ο χρήστης για κάποιο λόγο ξεχάσει τον προσωπικό κωδικό του, τότε έχει την δυνατότητα ανάκτησής του, μέσω της λειτουργίας αυτής. Το μόνο που έχει να κάνει ο χρήστης είναι, απλά να εισάγει το email του και αποστέλλεται ο κωδικός του σε αυτό άμεσα. Επομένως, είναι προφανές πόσο σημαντική είναι αυτή η λειτουργία καθώς αρκετοί χρήστες ξεχνούν τους κωδικούς τους αλλά μπορούν άμεσα και γρήγορα να τον «θυμηθούν».

6.3.4 Διαχείριση Προσωπικού Λογαριασμού

Σημαντικό είναι ο χρήστης να μπορεί να δει ολοκληρωμένο τον προσωπικό λογαριασμό που διατηρεί, αλλά επίσης να έχει την δυνατότητα να αλλάξει ανά πάσα στιγμή, οποιοδήποτε στοιχείο επιθυμεί. Έτσι έχει την δυνατότητα να αλλάξει το όνομα χρήστη ή τον κωδικό πρόσβασης ή το προσωπικό email, όποτε αυτός θελήσει. Τα νέα στοιχεία που εισάγει πρέπει πάντα να πληρούν τις προδιαγραφές εγκυρότητας, όπως για παράδειγμα ότι ο κωδικός πρόσβασης πρέπει να είναι τουλάχιστον τεσσάρων χαρακτήρων. Επίσης το email πρέπει να έχει μία συγκεκριμένη μορφή (user@mail.com), αλλά και να είναι μοναδικό, δηλαδή να μην υπάρχει στην βάση δεδομένων καταχωρημένο για άλλον χρήστη. Το όνομα χρήστη, username, επίσης πληροί τις συνθήκες μοναδικότητας, ενώ για όλα τα στοιχεία, ακόμα και για αυτά του ονόματος ή του επιθέτου, ο χρήστης οφείλει να τα έχει συμπληρωμένα με τουλάχιστον ένα χαρακτήρα, δηλαδή δεν έχει την δυνατότητα αλλαγής τους με κενές λέξεις.

6.3.5 Εποπτεία – Έλεγχος δωματίων-μονάδων κλιματιστικού

Σε κάθε χρήστη δίνεται η δυνατότητα να περιηγηθεί στα δωμάτια-χώρους του κτιρίου Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης, που έχουν τοποθετηθεί οι αισθητήρες και οι ενεργοποιητές. Κάθε χρήστης ανήκει σε μία κατηγορία η οποία επιτρέπει να έχει όλα ή κάποια από τα σήματα διαθέσιμα προς εποπτεία. Για παράδειγμα, υπάρχει η κατηγορία του διαχειριστή (administrator) όπου έχει διαθέσιμα όλα τα σήματα, υπάρχει η κατηγορία του απλού χρήστη (simple user) όπου ελάχιστα σήματα διατίθενται και

άλλες κατηγορίες που είναι λιγότερο σημαντικές για την παρούσα εργασία. Σε κάθε περίπτωση λοιπόν, ο χρήστης μπορεί να δει ποια σήματα ακριβώς περιέχει κάθε δωμάτιο ή μονάδα κλιματιστικού, αλλά και την τιμή τους τη δεδομένη²⁸ χρονική στιγμή. Παράλληλα, οι μονάδες κλιματιστικών στα γραφεία²⁹ διαθέτουν σήματα ελέγχου (actuators). Έτσι δίνεται η δυνατότητα σε κάθε χρήστη για παράδειγμα να ενεργοποιήσει ή να απενεργοποιήσει το air-condition και να ρυθμίσει την θερμοκρασία ή την ταχύτητα του ανεμιστήρα που διαθέτει.

6.3.6 Προβολή Διαγράμματος

Μία πάρα πολύ χρήσιμη λειτουργία που παρέχεται στην παρούσα εφαρμογή είναι αυτή της γραφικής απεικόνισης διαγραμμάτων για κάθε σήμα. Συγκεκριμένα, το διάγραμμα απεικονίζει τις τιμές του σήματος σε συνάρτηση με τον χρόνο, δηλαδή τις τιμές που είχε την τελευταία ώρα από την δεδομένη³⁰ χρονική στιγμή ή την τελευταία μέρα (24 ώρες), ή ακόμα και για χρονικό διάστημα μίας εβδομάδας. Επίσης ο χρήστης έχει την δυνατότητα να παρακολουθήσει την διακύμανση των τιμών κάθε σήματος σε οποιαδήποτε χρονική στιγμή για το τελευταίο έτος, χρησιμοποιώντας συγκεκριμένα κουμπιά. Επίσης υπάρχει η δυνατότητα να πραγματοποιήσει ζουμ στην γραφική παράσταση και να δει περισσότερες λεπτομέρειες και κυρίως τις τιμές του σήματος ανά λεπτό. Επίσης ότι αφορά την άλλη έκδοση που δημιουργήθηκε, ως προς την προβολή των διαγραμμάτων και περιγράφηκε στην παράγραφο [3.5.3], αυτή προσφέρει όλες τις λειτουργίες που αναφέρθηκαν εκτός από αυτήν της μεγέθυνσης-σμίκρυνσης (ζουμ).

6.3.7 Περισσότερες Πληροφορίες - Πλοήγηση

Η συγκεκριμένη εφαρμογή παρέχει την δυνατότητα περαιτέρω ενημέρωσης χωρίς να απαιτείται η σύνδεση του χρήστη στο σύστημα. Συγκεκριμένα, κάθε χρήστης μπορεί να πληροφορηθεί για όλα όσα αφορούν την ανάπτυξη και υλοποίηση της εφαρμογής. Για παράδειγμα μπορεί να δει πληροφορίες σχετικά με τον προγραμματιστή της εφαρμογής, να ενημερωθεί για την ημερομηνία δημιουργίας της, τους συντελεστές της, για τις βιβλιοθήκες που χρησιμοποιήθηκαν σε επίπεδο λογισμικού, την ανάλυση

²⁸ Ουσιαστικά πρόκειται για την τελευταία χρονική στιγμή που υπάρχει στην βάση δεδομένων, 21-3-2013 09:53:00

²⁹ Γραφεία με ονομασίες Office 1, Office 2, Office 3, Office 13 και άλλα.

³⁰ Ουσιαστικά πρόκειται για την τελευταία χρονική στιγμή που υπάρχει στην βάση δεδομένων, 21-3-2013 09:53:00

κάποιων όρων που χρησιμοποιούνται συχνά στην εφαρμογή, ή ακόμα και την πολιτική απορρήτου ή τους όρους χρήσης.

Παράλληλα έχει υλοποιηθεί και ενσωματωθεί σε αυτήν ένα μίνι πρόγραμμα πλοήγησης (browser). Ο συγκεκριμένος browser έχει ως αρχική σελίδα την επίσημη ιστοσελίδα του προγράμματος PEBBLE, για λόγους που αφορούν την περαιτέρω ενημέρωση του χρήστη για αυτό. Όμως, δίνεται και η δυνατότητα κανονικής πλοήγησης του χρήστη σε οποιαδήποτε άλλη ιστοσελίδα καθώς ακόμα και της αναζήτησης στην μεγαλύτερη μηχανή αναζήτησης της Google. Τέλος, σε περίπτωση που ο χρήστης είναι περισσότερο απαιτητικός και ουσιαστικά επιθυμεί την χρήση ενός κανονικού browser, υπάρχει η επιλογή μεταφοράς στον επίσημο browser της Apple, Safari, ή αντιγραφής του συνδέσμου της διεύθυνσης που βρίσκεται τη δεδομένη στιγμή.

6.4 Γραφική Απεικόνιση Λειτουργιών Εφαρμογής

6.4.1 Οθόνη Καλωσορίσματος

Η οθόνη που φαίνεται στην *Εικόνα 6.1* αποτελεί την αρχική οθόνη που συναντά ο χρήστης, δηλαδή την εικόνα που βλέπει μόλις ανοίγει την εφαρμογή. Στόχος είναι να «καλωσορίσει» τον χρήστη και κυρίως να του κινήσει το ενδιαφέρον. Αρχικά παρατηρείται ότι στο ανώτερο σημείο υπάρχει η ονομασία της εφαρμογής, ενώ μεγάλο μέρος της οθόνης καταλαμβάνει το λογότυπο του προγράμματος PEBBLE. Στο μεσαίο και κάτω τμήμα υπάρχουν δύο κουμπιά, καθένα από τα οποία παραπέμπει σε διαφορετικές οθόνες. Όπως είναι προφανές το πρώτο, **Sign In**, μεταφέρει τον χρήστη στην οθόνη σύνδεσης στο σύστημα, ενώ το δεύτερο, **About**, μεταφέρει τον χρήστη στην οθόνη πληροφόρησης. Έτσι η εφαρμογή δίνει την δυνατότητα επιλογής στον χρήστη από την πρώτη στιγμή, προτού καν συνδεθεί, να συγκεντρώσει όσες πληροφορίες αφορούν την εφαρμογή ώστε να είναι ενημερωμένος για το τι θα συναντήσει σε ενδεχόμενη σύνδεσή του.

Επιπλέον παρατηρείται ότι στο κατώτερο σημείο της οθόνης αναγράφονται τα πνευματικά δικαιώματα, ενώ υπάρχει και ένα τυπικό καλωσόρισμα ανάμεσα στην εικόνα-λογότυπο και το κουμπί σύνδεσης. Τέλος όπως φαίνεται, για λόγους ομορφιάς και κίνησης του ενδιαφέροντος του χρήστη, σε όλες τις οθόνες, των οποίων θα ακολουθήσει περιγραφή και ανάλυση, το φόντο επιλέχθηκε να έχει μία εικόνα με μεγάλο συντελεστή διαφάνειας.

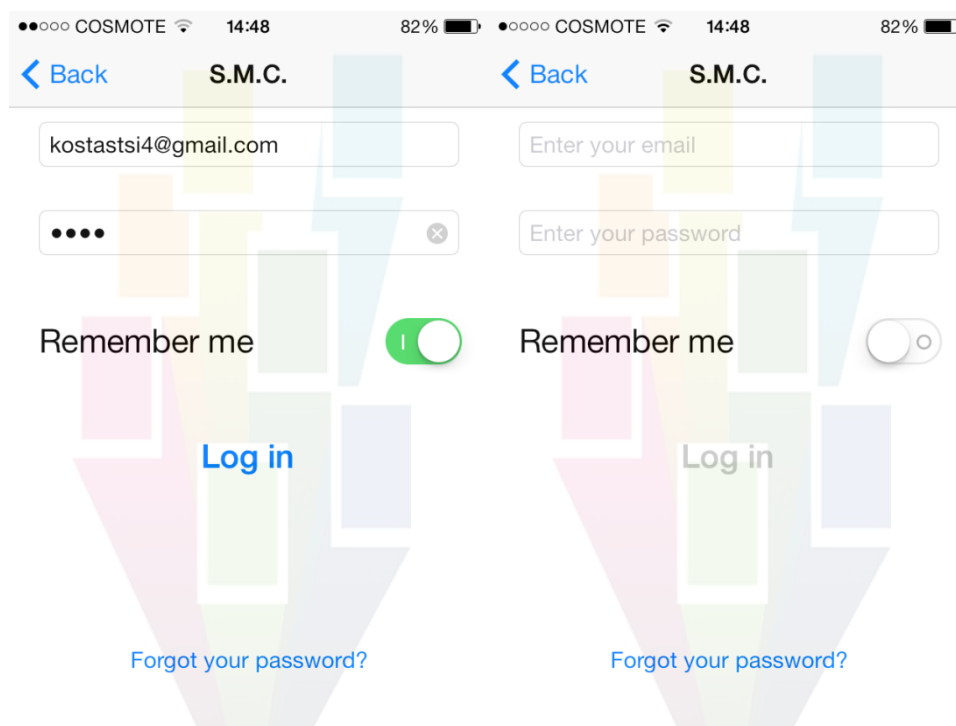


Εικόνα 6.1 – *Αριστερά.* Αρχική Οθόνη Καλωσορίσματος. – *Δεξιά.* Εικόνα καλωσορίσματος εφαρμογής – λογότυπο Πολυτεχνείου Κρήτης. Προηγείται της οθόνης καλωσορίσματος.

6.4.2 Οθόνη Εισόδου Χρήστη στην Εφαρμογή

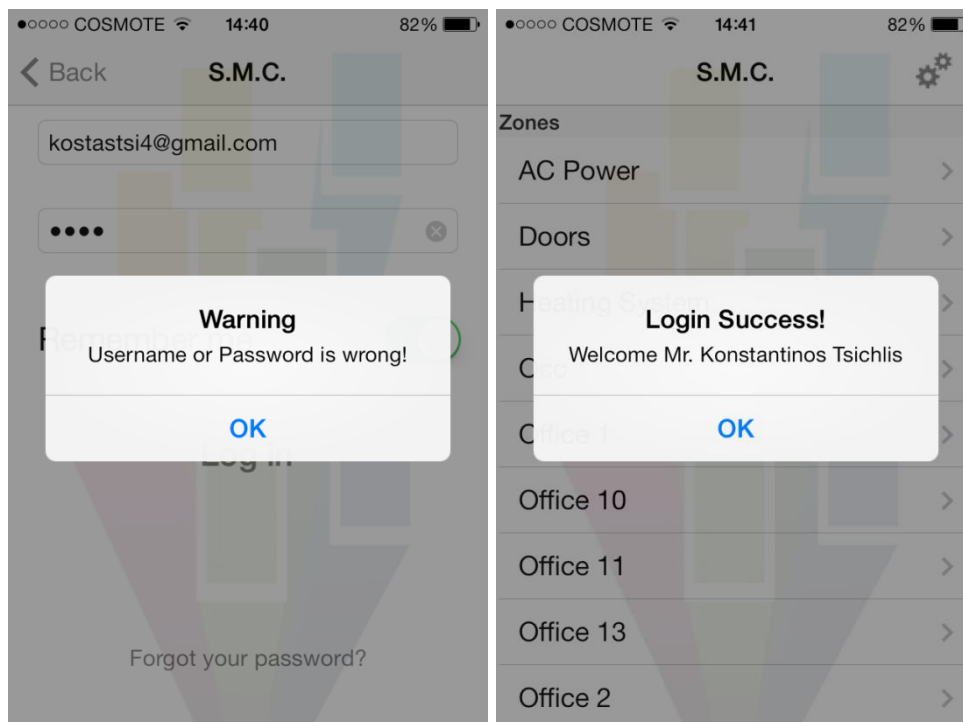
Η οθόνη που φαίνεται στην Εικόνα 6.2 αποτελεί οθόνη που συναντά ο χρήστης, ώστε να συνδεθεί στο σύστημα. Υπάρχουν λοιπόν, δύο πεδία όπου το πρώτο είναι για την εισαγωγή του ονόματος χρήστη και το δεύτερο για τον κωδικό πρόσβασης, κάτι που διευκρινίζεται με την χρήση φράσεων προτροπής, γνώστες και ως «placeholder text». Επισημαίνεται ότι ο κωδικός πρόσβασης εμφανίζεται με κουκίδες για λόγους ασφαλείας. Στην μέση περίπου της οθόνης υπάρχει το κουμπί για την σύνδεση, **Log in**, το οποίο παραμένει απενεργοποιημένο και ενεργοποιείται μόνο εφόσον έχουν εισαχθεί στοιχεία και στα δύο πεδία. Επίσης υπάρχει και η επιλογή, **Remember me**, που αν είναι ενεργοποιημένη τότε η εφαρμογή αποθηκεύει, για λόγους ασφάλειας όπως είναι προφανές, μόνο το όνομα χρήστη και όχι τον κωδικό πρόσβασης. Έτσι κάθε φορά που ανοίγει την εφαρμογή ο χρήστης, το πεδίο του ονόματος χρήστη θα είναι πάντα συμπληρωμένο εξοικονομώντας αρκετό χρόνο. Τέλος στο κατώτατο τμήμα της οθόνης

υπάρχει η επιλογή ανάκτησης κωδικού πρόσβασης. Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, σε περίπτωση που ο χρήστης έχει ξεχάσει τον κωδικό του, τότε έχει την δυνατότητα να τον ανακτήσει.



Εικόνα 6.2 – Οθόνη σύνδεσης χρήστη στο σύστημα.

Σε περίπτωση που ο χρήστης εισάγει λαθεμένα προσωπικά στοιχεία τότε εμφανίζεται ένα μήνυμα (popup) που τον ειδοποιεί ότι έκανε λάθος. Σε διαφορετική περίπτωση, ο χρήστης εισάγεται στο σύστημα και εμφανίζεται ένα μήνυμα (popup) καλωσορίσματος. Παραδείγματα φαίνονται στην *Εικόνα 6.3*.



Εικόνα 6.3 – Αριστερά. Παράδειγμα λανθασμένης εισαγωγής στοιχείων. – Δεξιά. Παράδειγμα επιτυχούς σύνδεσης, μήνυμα καλωσορίσματος.

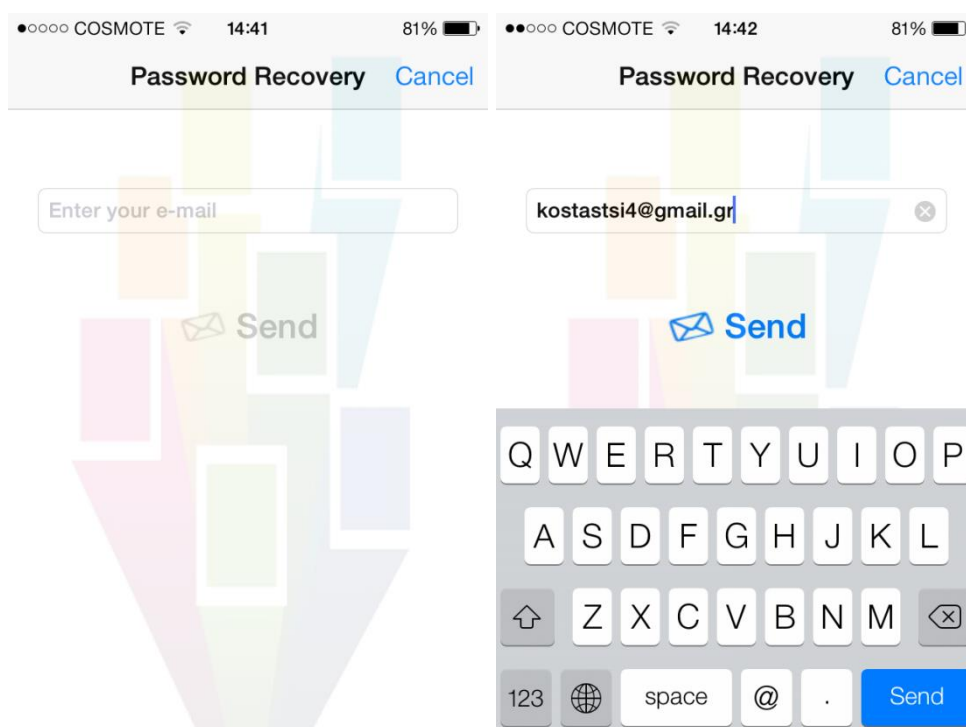
6.4.3 Οθόνη Ανάκτησης Κωδικού

Η συγκεκριμένη οθόνη, όπως φαίνεται στην *Εικόνα 6.4*, διαθέτει ένα πεδίο στο οποίο παροτρύνεται ο χρήστης να εισάγει το προσωπικό του email και έπειτα να πατήσει το κουμπί αποστολής, **Send**, ώστε να αποσταλεί στο συγκεκριμένο email ο προσωπικός του κωδικός. Υπάρχει και εδώ ένα μήνυμα προτροπής (placeholder text), ενώ το κουμπί είναι απενεργοποιημένο όσο το πεδίο είναι κενό και ενεργοποιείται μόλις εισαχτεί έστω και ένας χαρακτήρας. Παραδείγματα φαίνονται στην *Εικόνα 6.4*.

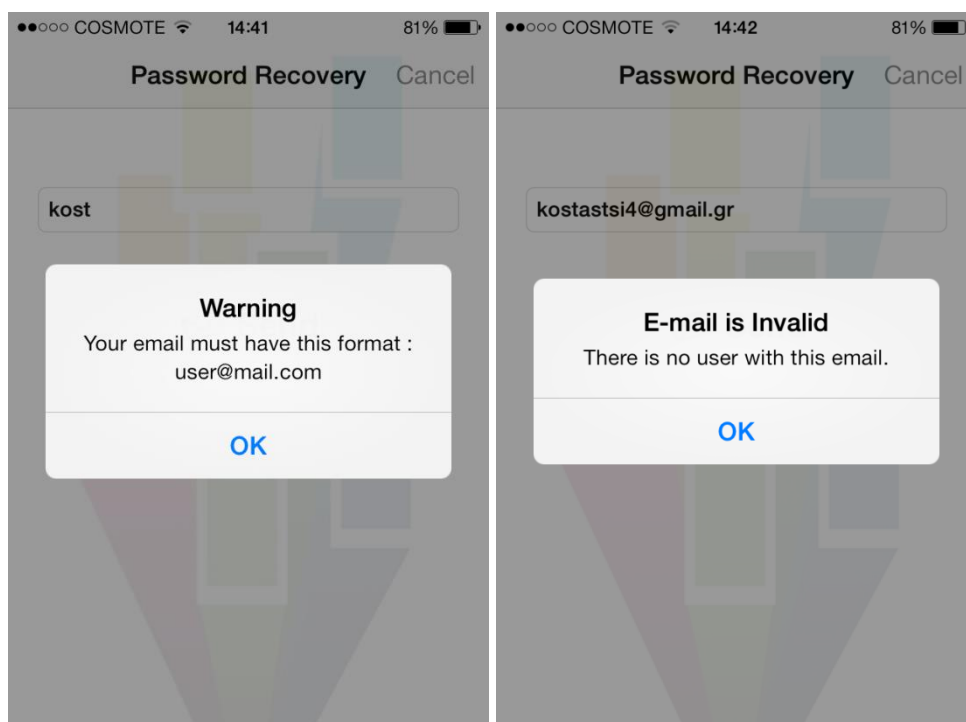
Το email που εισάγει ο χρήστης πρέπει να έχει συγκεκριμένη μορφή, [user@mail.com](#). Επομένως αν δεν είναι στη σωστή μορφή, τότε εμφανίζεται ένα μήνυμα λάθους. Επίσης σε περίπτωση που εισαχθεί ένα email που δεν υπάρχει στην βάση δεδομένων του συστήματος εμφανίζεται αντίστοιχο μήνυμα ειδοποίησης. Παραδείγματα εμφανίζονται στην *Εικόνα 6.5*.

Αν όλα είναι σωστά τότε αποστέλλεται ο κωδικός πρόσβασης στο email που έχει δοθεί από τον χρήστη, το οποίο τονίζεται ακόμα μια φορά ότι πρέπει να είναι έγκυρο και υπαρκτό. Παράλληλα, διευκρινίζεται ότι πάντα κατά την διάρκεια ελέγχου ορθότητας του email και αποστολής, εμφανίζεται ένα ενημερωτικό μήνυμα (*Εικόνα 6.6*). Στην

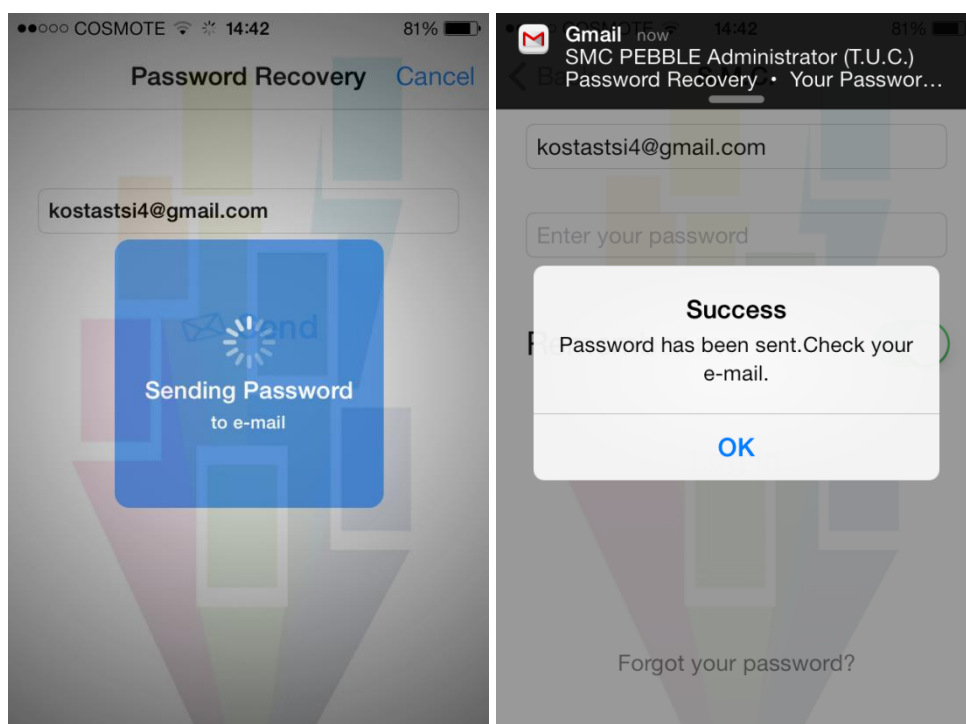
περίπτωση επιτυχής αποστολής λοιπόν, εμφανίζεται μήνυμα που ενημερώνει τον χρήστη να ανατρέξει στο προσωπικό του email, ώστε να δει τον κωδικό πρόσβασης που έχει αποσταλεί. Στην *Εικόνα 6.6* φαίνεται το μήνυμα αυτό αλλά και την ίδια στιγμή το notification του Gmail ότι λήφθηκε email με αποστολέα το SMC Pebble Administrator. Τέλος στην *Εικόνα 6.7* παρουσιάζεται ένα παράδειγμα της μορφής του email που αποστέλλεται κάθε φορά στον χρήστη.



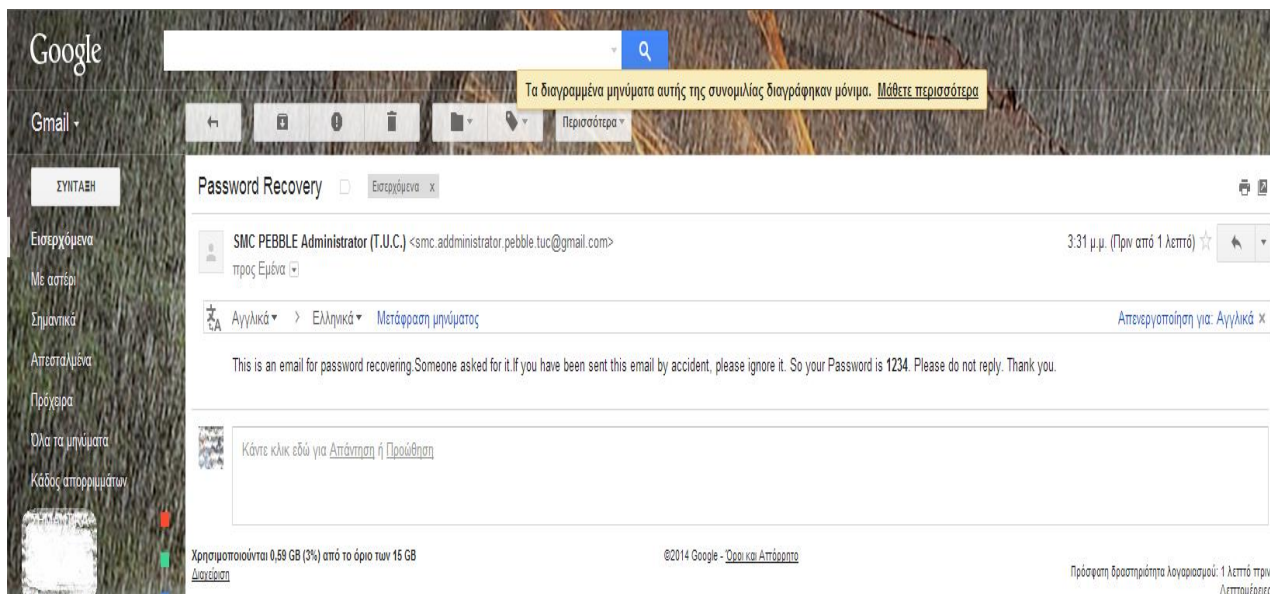
Εικόνα 6.4 – Οθόνη ανάκτησης κωδικού πρόσβασης. Απενεργοποιημένο (αριστερά) και ενεργοποιημένο (δεξιά) το κουμπί Send.



Εικόνα 6.5 – **Αριστερά.** Παράδειγμα λανθασμένης εισαγωγής email. – **Δεξιά.** Παράδειγμα email που δεν υπάρχει στην βάση δεδομένων του συστήματος.



Εικόνα 6.6 – **Αριστερά.** Μήνυμα κατά τη διάρκεια ελέγχου ορθότητας email και αποστολής κωδικού. – **Δεξιά.** Παράδειγμα εμφάνισης μηνύματος επιτυχής αποστολής κωδικού πρόσβασης.



Εικόνα 6.7 – Παράδειγμα email από αίτηση ανάκτησης κωδικού.

6.4.4 Οθόνη Εποπτείας ζωνών, δωματίων και μονάδων A/C

Στην Εικόνα 6.8 παρουσιάζεται η όψη των οθονών που απεικονίζουν τις ζώνες, τα σήματα κάθε ζώνης αλλά και τις μονάδες (κλιματιστικών) που ίσως διαθέτει κάθε ζώνη. Οι ζώνες αποτελούν:

- Ομαδοποιημένες κατηγορίες αισθητήρων, όπως είναι η κατανάλωση ενέργειας των κλιματιστικών (AC Power), κατανάλωση ενέργειας στο κτίριο (Heat Meter), όλοι οι αισθητήρες θερμοκρασίας (Temperature), ή οι αισθητήρες που έχουν εγκατασταθεί πάνω στις πόρτες
- Ομάδες αισθητήρων που βρίσκονται στον ίδιο χώρο, όπως τα όλα τα γραφεία (Office 1-13).

Κάτω ακριβώς από την μπάρα πλοήγησης παρατηρείται ότι υπάρχει το όνομα, **Zones**, σαν τίτλος που προσδιορίζει ότι τα ονόματα που ακολουθούν αναφέρονται σε ζώνες, ενώ δεξιά από κάθε όνομα υπάρχει ένας δείκτης που προσδιορίζει στον χρήστη ότι οδηγεί σε άλλη οθόνη αν πατηθεί. Συγκεκριμένα οδηγεί στην οθόνη εμφάνισης των

σημάτων που περιέχει η συγκεκριμένη ζώνη. Σημειώνεται ότι οι ζώνες και τα σήματα που εμφανίζονται στον χρήστη είναι άμεσα εξαρτώμενα από τι είδος χρήστη είναι³¹.

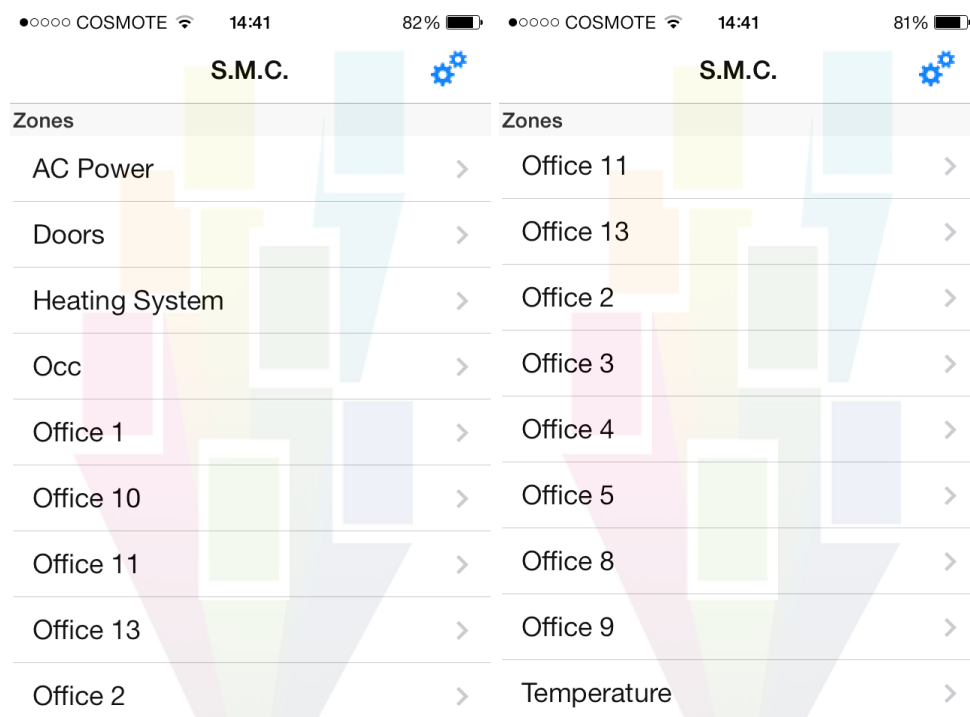
Η οθόνη κάθε ζώνης περιέχει τα σήματα, **Signals**, που είναι εγκατεστημένα σε αυτήν αλλά και οποιαδήποτε άλλη ομάδα σημάτων, **Groups**, όπως είναι τα κλιματιστικά. Τα **Groups** έχουν ακριβώς την ίδια συμπεριφορά με αυτήν των ζωνών που περιγράφηκε προηγουμένως, ενώ τα **Signals** αποτελούνται από το όνομα κάθε σήματος σε μαύρο χρώμα, την τιμή του σήματος μαζί με την μονάδα μέτρησης (αν υπάρχει) σε γαλάζιο χρώμα και στα δεξιά έχουν ένα δείκτη. Αυτός ο δείκτης παρακινεί την χρήστη να πατήσει και να μεταφερθεί στην οθόνη όπου απεικονίζεται το διάγραμμα των τιμών του.

Επίσης στο τέλος της οθόνης απεικονίζονται οι καρτέλες **Monitoring** και **Controls** που υποδεικνύουν αν ο χρήστης βρίσκεται στην εποπτεία των σημάτων των αισθητήρων ή των σημάτων ελέγχου. Σημειώνεται ότι πάντα η καρτέλα που είναι ενεργή φωτίζεται, ενώ σχετικά με την καρτέλα των **Controls**, αυτή δεν είναι πάντα ενεργοποιημένη³², καθώς δεν διαθέτουν όλες οι ζώνες σήματα ελέγχου, όπως για παράδειγμα το γραφείο **Office 1**. Έτσι ο χρήστης ακόμα και να προσπαθήσει να αλλάξει καρτέλα, η εφαρμογή δεν του το επιτρέπει (*Εικόνα 6.9 – Αριστερά*). Αντίθετα, στην περίπτωση που παρακολουθεί ο χρήστης την μονάδα του κλιματιστικού στο γραφείο αυτό, τότε η καρτέλα είναι ενεργοποιημένη διαθέτοντας ένα πιο σκούρο γκρι χρώμα (*Εικόνα 6.9 - Δεξιά*).

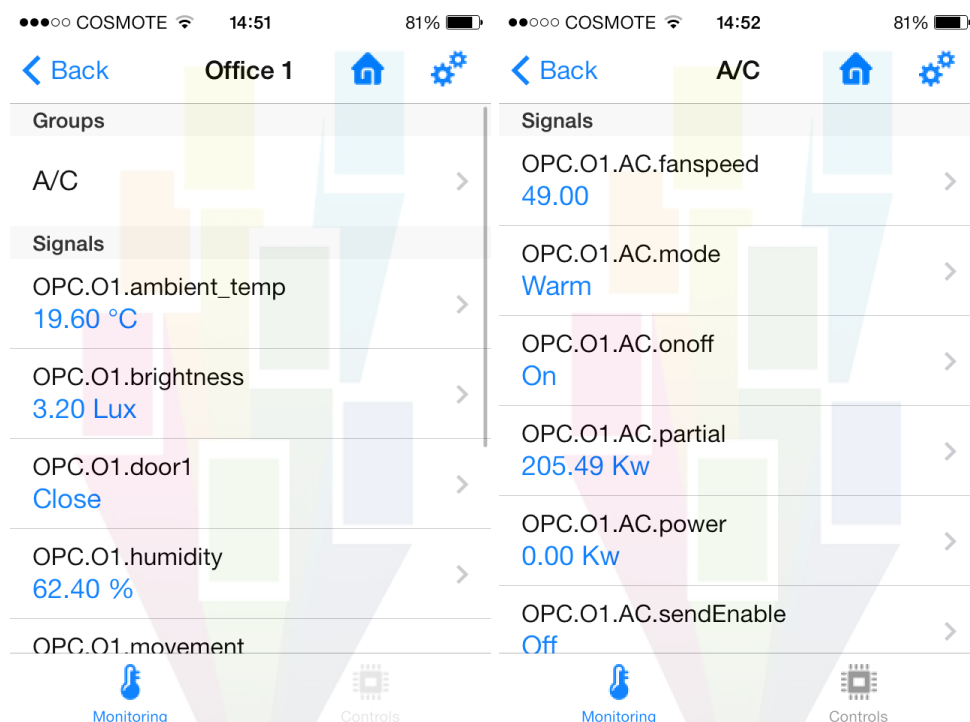
Η μπάρα πλοήγησης, όπως περιγράφηκε και στο πέμπτο κεφάλαιο διαθέτει εικονίδια που αντιπροσωπεύουν τις λειτουργίες τους. Η ενέργεια του **Home Button** οδηγεί πάντα στην οθόνη των ζωνών, καθώς αποτελεί την πρώτη οθόνη που συναντά ο χρήστης αμέσως μετά την είσοδό του στο σύστημα.. Αυτό γίνεται για λόγους ευκολίας του χρήστη, ώστε αν μπορεί γρήγορα και εύκολα να μεταβεί σε αυτήν από όπου κι αν βρίσκεται. Συνεπώς δεν υπάρχει το κουμπί, **Home Button**, στην οθόνη των ζωνών (*Εικόνα 6.8*).

³¹ Είδη χρηστών έχουν αναφερθεί σε προηγούμενη ενότητα. Απλή υπενθύμιση :Administrator, Simple User.

³²Ο όρος Ενεργοποιημένη είναι διαφορετικός του όρου Ενεργή. Σημαίνει ότι δεν μπορεί να πατηθεί από τον χρήστη και επομένως ότι δεν υπάρχουν διακόπτες ελέγχου.

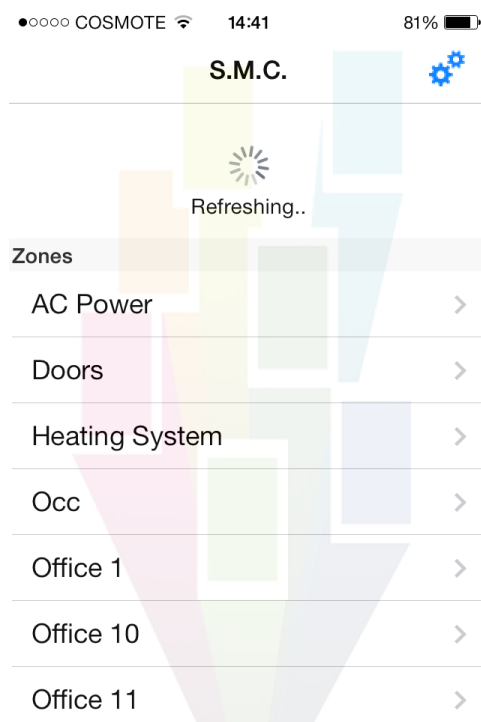


Εικόνα 6.8 – Παράδειγμα αρχικής οθόνης που απεικονίζει τις ζώνες.



Εικόνα 6.9 – **Αριστερά.** Παράδειγμα οθόνης που απεικονίζει την ζώνη Office 1 (Απενεργοποιημένο το Controls) – **Δεξιά.** Παράδειγμα οθόνης που απεικονίζει την μονάδα κλιματιστικού του Office 1 (Ενεργοποιημένο το Controls).

Επίσης δίνεται η δυνατότητα στον χρήστη να ανανεώσει ανά πάσα στιγμή όλα τα σήματα και τις ζώνες τραβώντας με το δάχτυλό του από την κορυφή της οθόνης προς τα κάτω, όπως φαίνεται στην *Εικόνα 6.10*.



Εικόνα 6.10 – Παράδειγμα ανανέωσης των σημάτων-ζωνών.

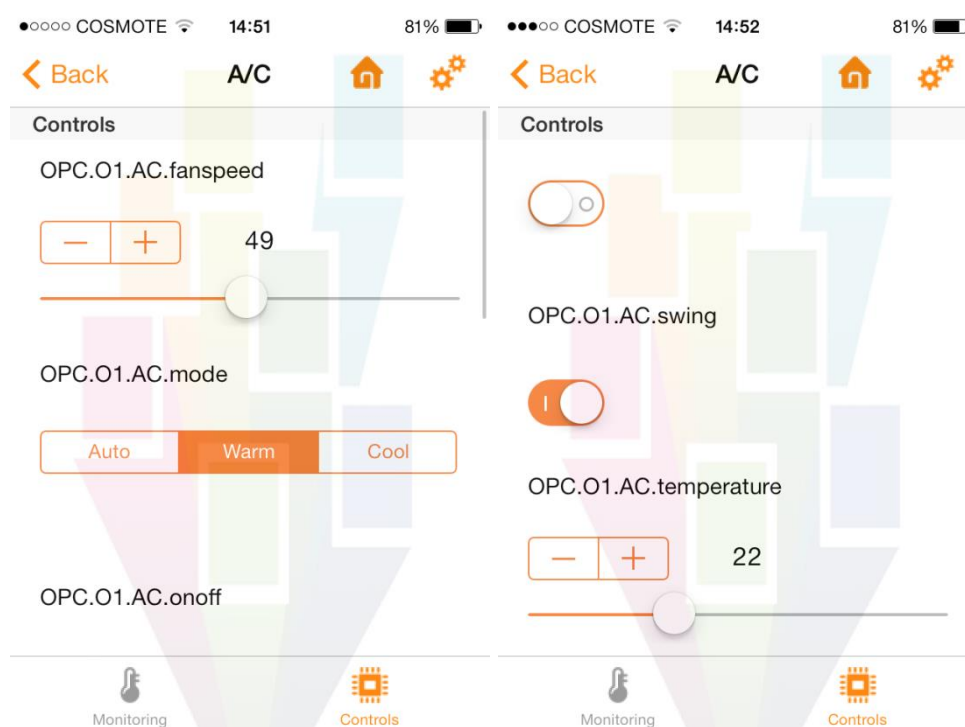
6.4.5 Οθόνη Σημάτων Ελέγχου

Αν ο χρήστης επιλέξει να δει κάποια μονάδα κλιματιστικού ή γενικά μία ζώνη ή ομάδα σημάτων που διαθέτουν σήματα ελέγχου, τότε η καρτέλα **Controls**, όπως αναφέρθηκε προηγουμένως, είναι ενεργοποιημένη. Έτσι αν ο χρήστης την επιλέξει μεταφέρεται στην οθόνη με τα σήματα ελέγχου. Τα σήματα ελέγχου είναι τριών ειδών:

- **Διακόπτης διπλής επιλογής:** Ο συγκεκριμένος διακόπτης περιγράφει σήματα τύπου On/Off, όπως ο διακόπτης ενεργοποίησης/απενεργοποίησης του κλιματιστικού.
- **Διακόπτης αύξησης/μείωσης:** περιγράφει σήματα που οι τιμές τους κυμαίνονται από ένα ελάχιστο έως ένα μέγιστο με συγκεκριμένο βήμα, όπως ο διακόπτης ρύθμισης θερμοκρασίας.

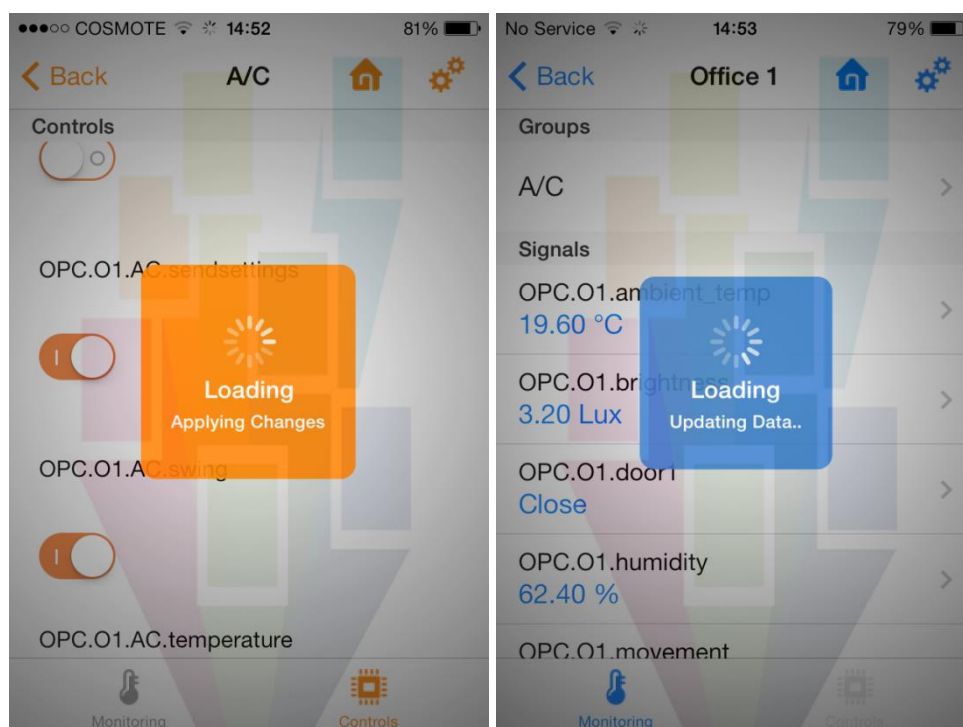
- **Διακόπτης τριπλής επιλογής:** περιγράφει σήματα που οι τιμές τους κυμαίνονται ανάμεσα σε τρεις καταστάσεις, όπως είναι ο διακόπτης ρύθμισης του τύπου λειτουργίας του air-condition σε θέρμανση, ψύξη ή αυτόματο.

Έπειτα ο χρήστης έχει την δυνατότητα να αλλάξει όποιον διακόπτη επιθυμεί. Για παράδειγμα στην *Εικόνα 6.11* που εμφανίζεται το κλιματιστικό του Office 1, ο χρήστης μπορεί να το ενεργοποιήσει, να το θέσει σε κατάσταση ψύξης και να ρυθμίσει συγκεκριμένη θερμοκρασία και συγκεκριμένες στροφές του ανεμιστήρα.



Εικόνα 6.11 – Παράδειγμα οθόνης διακοπών ελέγχου air-condition στο Office 1.

Αξίζει να σημειωθεί ότι σε οποιαδήποτε αλλαγή προβεί ο χρήστης, τότε αυτόματα ενημερώνεται το σύστημα και εμφανίζεται ένα μήνυμα για όση ώρα διαρκεί η ενημέρωση. Γενικότερα, έπειτα από οποιαδήποτε αλλαγή, αν ο χρήστης μεταβεί σε προηγούμενες οθόνες που αφορούν τα σήματα που άλλαξαν οι τιμές τους, τότε εμφανίζεται κατάλληλο μήνυμα, για όση ώρα διαρκεί η ανανέωση των τιμών τους, όπως φαίνεται στην *Εικόνα 6.12*.

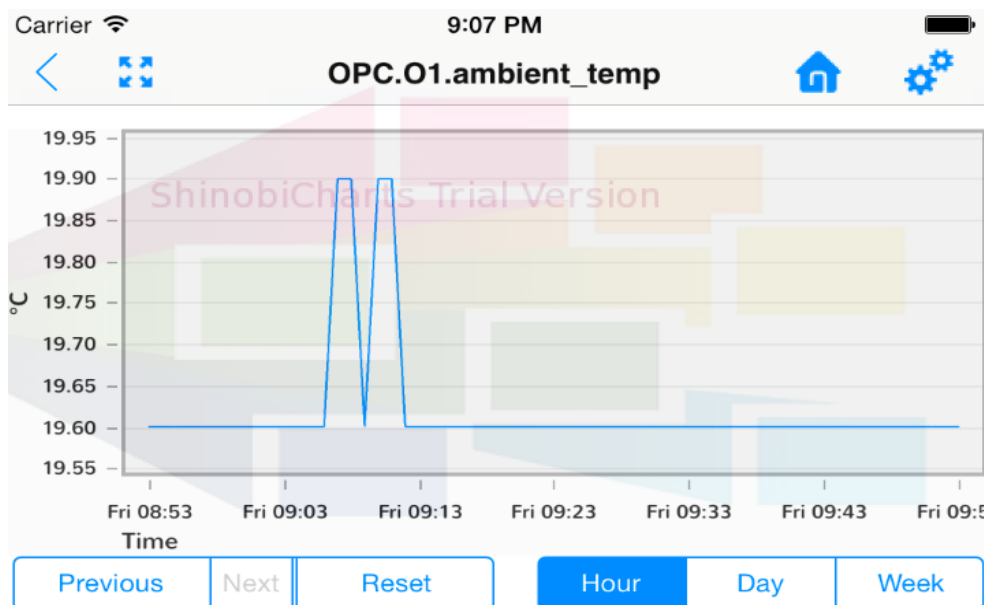


Εικόνα 6.12 – *Αριστερά.* Παράδειγμα ανανέωσης οθόνης σημάτων ελέγχου. – *Δεξιά.* Παράδειγμα οθόνης ζώνης Office 1 έπειτα από αλλαγή σημάτων ελέγχου.

6.4.6 Οθόνη Προβολής Διαγράμματος

Όπως αναφέρθηκε σε προηγούμενη παράγραφο, ο χρήστης έχει την δυνατότητα πατώντας πάνω σε κάθε σήμα να παρακολουθήσει ένα διάγραμμα των τιμών του συναρτήσει του χρόνου. Αρχικά το διάγραμμα, μόλις φορτώσει για πρώτη φορά, απεικονίζει σε χρονικό διάστημα μίας ώρας, και συγκεκριμένα της τελευταίας ώρας από την δεδομένη³³ χρονική στιγμή, τις τιμές τους σήματος. Όπως φαίνεται στην Εικόνα 6.13 υπάρχουν πολλές επιλογές σχετικά με το διάγραμμα.

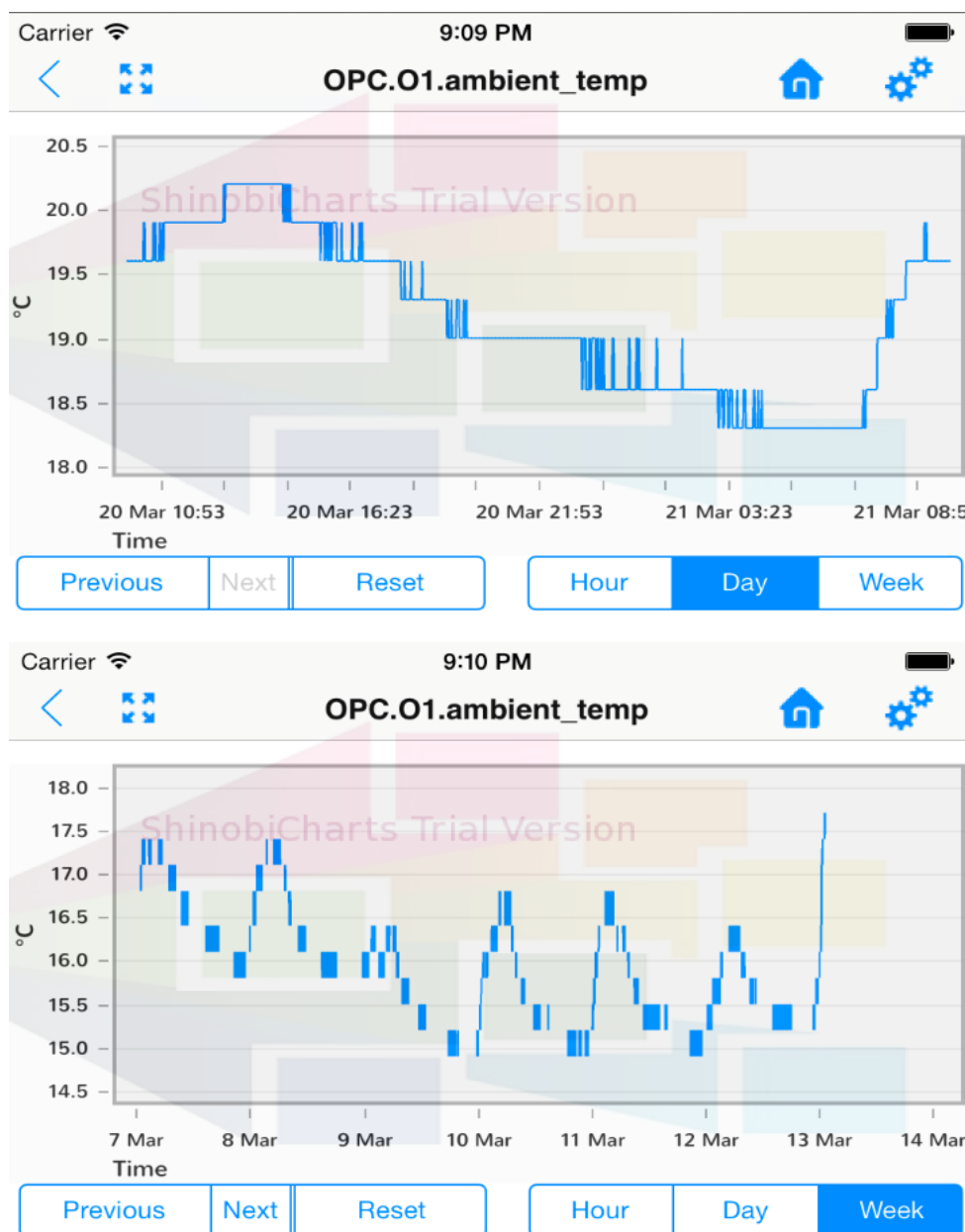
³³ Ουσιαστικά πρόκειται για την τελευταία χρονική στιγμή που υπάρχει στην βάση δεδομένων, 21-3-2013 09:53:00



Εικόνα 6.13 – Οθόνη διαγράμματος του σήματος **OPC.O1.ambient_temp** σε διάστημα μίας ώρας. Πρόκειται για αισθητήρα θερμοκρασίας καθώς έχει μονάδα μέτρησης βαθμούς Κελσίου.

Αρχικά παρατηρείται ότι ο τίτλος στο κέντρο της μπάρας πλοήγησης αφορά το όνομα του σήματος που απεικονίζεται στο διάγραμμα. Αριστερά του ονόματος βρίσκεται το εικονίδιο επαναφοράς του διαγράμματος στις πραγματικές διαστάσεις και λειτουργεί μόνο όταν έχει γίνει ζουμ από τον χρήστη. Στο μέρος της ενεργού περιοχής προβάλλεται το διάγραμμα όπου στα αριστερά, στον κάθετο άξονα, είναι οι τιμές του σήματος μαζί με την μονάδα μέτρησής του, ενώ στον οριζόντιο άξονα, είναι τιμές του χρόνου. Στο κάτω μέρος της οθόνης υπάρχουν δύο ομάδες πλήκτρων. Στα δεξιά παρέχεται η δυνατότητα στον χρήστη να επιλέξει το χρονικό διάστημα που θέλει να προβάλλονται οι τιμές στο διάγραμμα, συναρτήσει του χρόνου. Έχει δηλαδή να επιλέξει ανάμεσα σε διάστημα μίας ώρας, μίας ημέρας ή μίας εβδομάδας. Στα αριστερά παρέχεται η δυνατότητα στον χρήστη να επιλέξει πόσες ώρες/ημέρες/ εβδομάδες νωρίτερα ή αργότερα από την δεδομένη χρονική στιγμή θέλει να δει το διάγραμμα τιμών του σήματος, ενώ ανά πάσα στιγμή μπορεί να αναιρέσει την επιλογή και να βρεθεί στην ώρα/ημέρα/εβδομάδα της δεδομένης στιγμής. Χαρακτηριστικά παραδείγματα φαίνονται στην Εικόνα 6.14.

Σε αυτό το σημείο αξίζει να σημειωθεί ότι η συγκεκριμένη λειτουργία υλοποιήθηκε με τέτοιο τρόπο, ώστε σε περίπτωση που ο χρήστης περιηγείται σε συγκεκριμένη ημερομηνία και αλλάζει το χρονικό διάστημα προβολής του διαγράμματος, τότε γίνεται προσέγγιση στην πιο παλαιά ημερομηνία και ώρα, δηλαδή στην ημερομηνία και ώρα που βρίσκεται στα αριστερά του άξονα χρόνου. Στην υπάρχουσα web εφαρμογή το διάγραμμα αρχικοποιείται στην πιο πρόσφατη ημερομηνία, οπότε θεωρήθηκε αναγκαίο να αλλάξει αυτό, ώστε ο χρήστης να συνεχίζει από το ίδιο χρονικό σημείο που βρισκόταν.



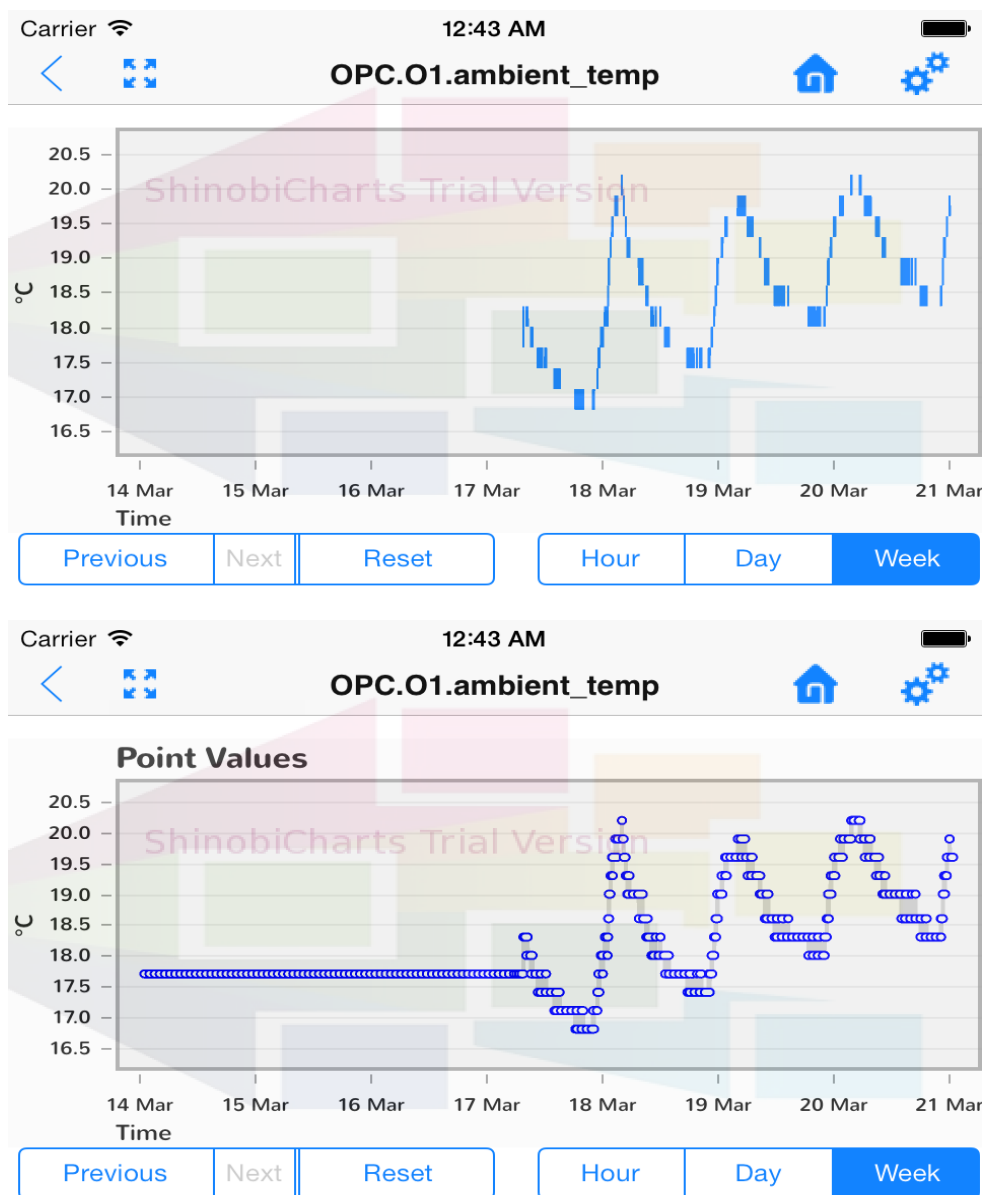
Εικόνα 6.14 – Πάνω. Οθόνη διαγράμματος του σήματος **OPC.O1.ambient_temp** σε χρονικό διάστημα μίας ημέρας. – Κάτω. Οθόνη διαγράμματος του σήματος **OPC.O1.ambient_temp** σε χρονικό διάστημα μίας εβδομάδας.

Προφανώς υπάρχει η δυνατότητα πραγματοποίησης ζουμ³⁴, όπως αναφέρθηκε και προηγουμένως, αλλά μία αρκετά σημαντική λειτουργία είναι αυτή κατά την οποία ο χρήστης χτυπώντας το δάχτυλο μόνο μία φορά στην οθόνη του διαγράμματος, τότε εμφανίζονται τα σημεία των τιμών του σήματος³⁵ ως κουκίδες-σημεία (Εικόνα 6.15). Αυτή η λειτουργία προστέθηκε ώστε ο χρήστης να διευκολυνθεί, καθώς στην

³⁴ Η λειτουργία του ζουμ λειτουργεί με τον κλασικό τρόπο που εφαρμόζεται σε κάθε smartphone.

³⁵ Ουσιαστικά πρόκειται για αλλαγή της γραφικής παράστασης του σήματος σε ένα είδος «scatterplot».

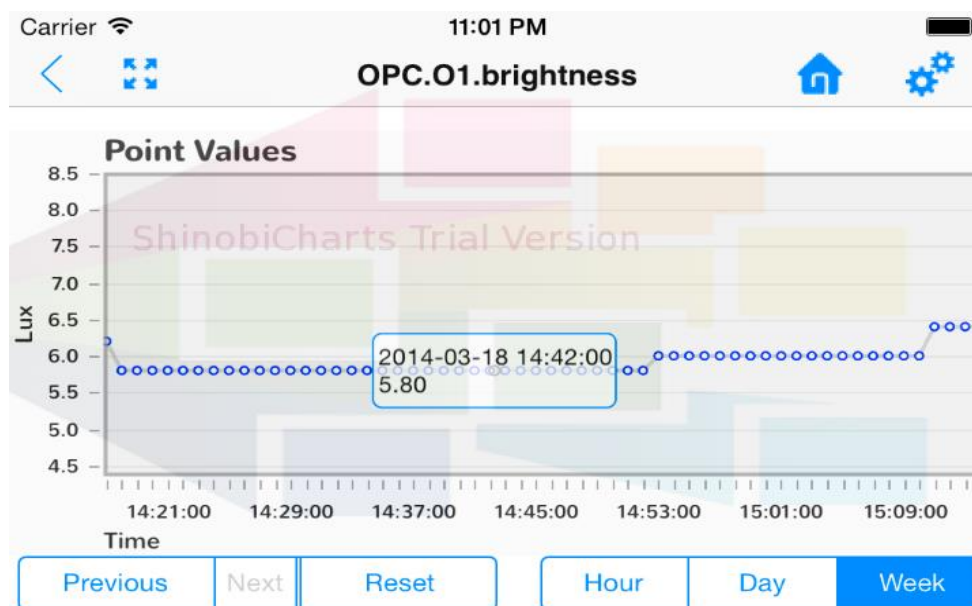
περίπτωση επιλογής εμφάνισης του διαγράμματος ανά εβδομάδα, είναι πολύ πυκνή λόγω μικρού μεγέθους οθόνης του κινητού τηλεφώνου. Έτσι με αυτή την λειτουργία γίνονται πιο ξεκάθαρες οι τιμές του σήματος και σε συνδυασμό με τη λειτουργία πραγματοποίησης ζουμ, ο χρήστης έχει την δυνατότητα να δει ξεκάθαρα όποια τιμή και σε οποιαδήποτε χρονική στιγμή θέλει.



Εικόνα 6.15 – Πάνω. Παράδειγμα εμφάνισης σήματος σε χρονικό διάστημα μίας εβδομάδας. – Κάτω. Παράδειγμα εμφάνισης λειτουργίας Point Values. Εμφάνιση των ακριβή σημείων-τιμών του σήματος. Εμφανής η διαφορά.

Αξίζει να σημειωθεί σε αυτό το σημείο, ότι υπάρχουν περιπτώσεις που κάποια σήματα δεν διαθέτουν τιμές για συγκεκριμένες χρονικές στιγμές³⁶. Επομένως, έγινε μία παραδοχή ώστε το διάγραμμα να μπορεί να σχεδιαστεί χωρίς σφάλματα. Συγκεκριμένα, αποφασίστηκε για κάθε σημείο που συμβαίνει κάτι τέτοιο να λαμβάνει την τιμή του σημείου που βρίσκεται ακριβώς πριν από αυτό. Αυτό αφορά το σχεδιαστικό κομμάτι του λογισμικού της εφαρμογής και το γεγονός των προβλημάτων που παρατηρούνται στις τιμές των σημάτων, είναι εκτός αντικειμένου της παρούσας εργασίας.

Επίσης, όπως φαίνεται στην *Εικόνα 6.16*, για να διευκολυνθεί ο χρήστης ακόμα περισσότερο, υπάρχει η δυνατότητα πατώντας με το δάχτυλο σε οποιοδήποτε σημείο του διαγράμματος και κρατώντας το πατημένο, να εμφανιστεί ένα ορθογώνιο πλαίσιο που απεικονίζει τις τιμές στους δύο άξονες του σήματος στο συγκεκριμένο σημείο. Αν ο χρήστης συνεχίσει να έχει πατημένο το σημείο, έπειτα μπορεί να σύρει το δάχτυλό του και να δει τις τιμές σε όλα τα σημεία, καθώς το πλαίσιο μετακινείται ανάλογα με τα σημεία των τιμών του σήματος (point values).



Εικόνα 6.16 – Παράδειγμα εμφάνισης ορθογώνιου πλαισίου που απεικονίζει τις τιμές του σήματος στο συγκεκριμένο σημείο. Πρόκειται για αισθητήρα φωτεινότητας, καθώς έχει μονάδα μέτρησης Lux, και απεικονίζεται σε χρονικό διάστημα μίας εβδομάδας με μεγάλο ποσοστό zoom.

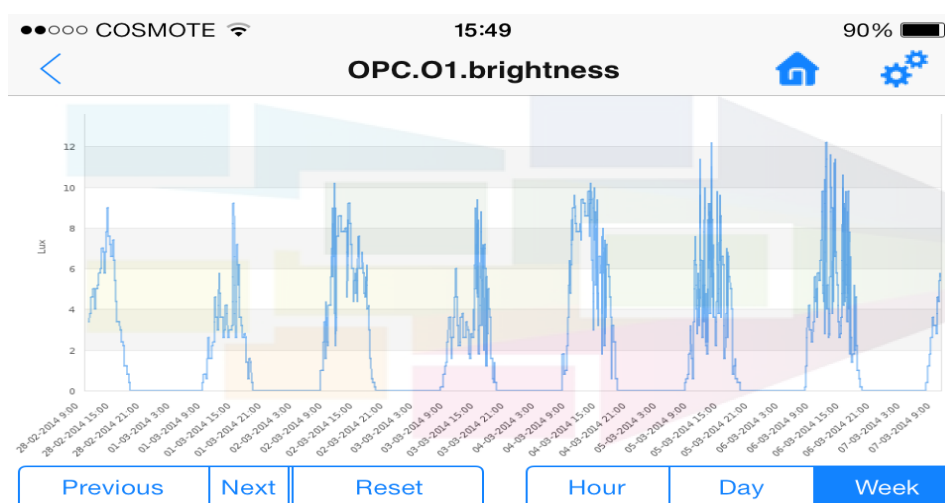
Τέλος κατά την διάρκεια φόρτωσης των τιμών του σήματος και σχεδιασμού του διαγράμματος εμφανίζεται ένα μήνυμα ενημέρωσης του χρήστη όπως φαίνεται παρακάτω στην *Εικόνα 6.17*.

³⁶ Υπάρχουν πολλά «κενά» στις τιμές των σημάτων, δηλαδή οι τιμές τους είναι NULL, με αποτέλεσμα να υπάρχει πρόβλημα στην δημιουργία του διαγράμματος.



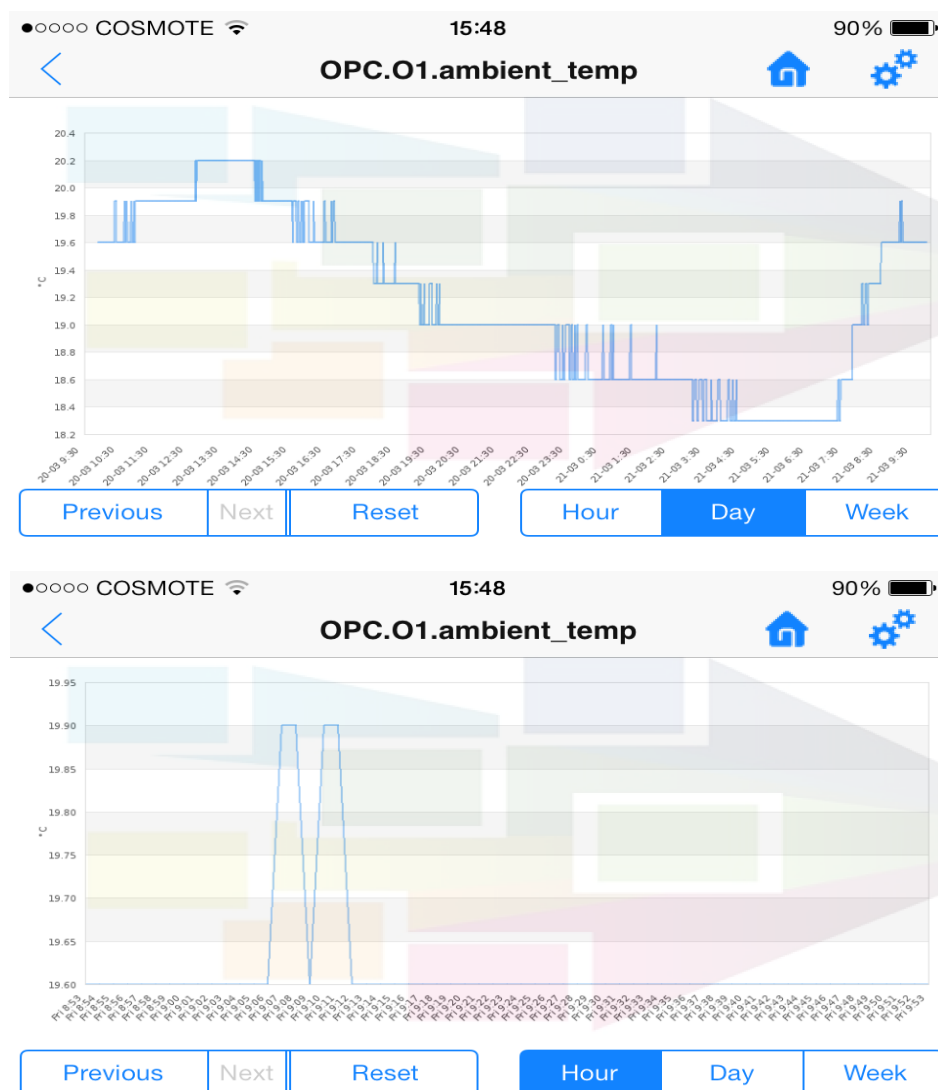
Εικόνα 6.17 – Παράδειγμα εμφάνισης λειτουργίας *Point Values*. Εμφάνιση των ακριβή σημείων-τιμών του σήματος.

Σε αυτό το σημείο αξίζει αναφοράς η περίπτωση χρήσης της βιβλιοθήκης **JpGraph** [3.5.3], της οποίας η οθόνη φαίνεται στην Εικόνα 6.18. Συγκεκριμένα, φαίνεται να υπάρχουν τα ίδια κουμπιά λειτουργιών που περιγράφηκαν έως τώρα, αλλά λείπει το κουμπί επαναφοράς του διαγράμματος στις πραγματικές του διαστάσεις, πάνω αριστερά στην μπάρα πλοήγησης. Αυτό αποδίδεται στο γεγονός ότι η συγκεκριμένη βιβλιοθήκη δεν παρέχει την δυνατότητα πραγματοποίησης ζουμ, άρα δεν υπάρχει λόγος ύπαρξής του. Μία επιπλέον παράλειψη έναντι της προηγούμενης περίπτωσης, είναι ότι ο χρήστης δεν έχει την δυνατότητα να πατήσει σε οποιοδήποτε σημείο του γραφήματος και να εμφανιστεί κάποιο επιπλέον μήνυμα, το οποίο εμπεριέχει αναλυτικά τις τιμές σε αυτό το σημείο.



Εικόνα 6.18 – Παράδειγμα προβολής διαγράμματος ενός σήματος σε διάρκεια μίας εβδομάδας με χρήση της βιβλιοθήκης *JpGraph*.

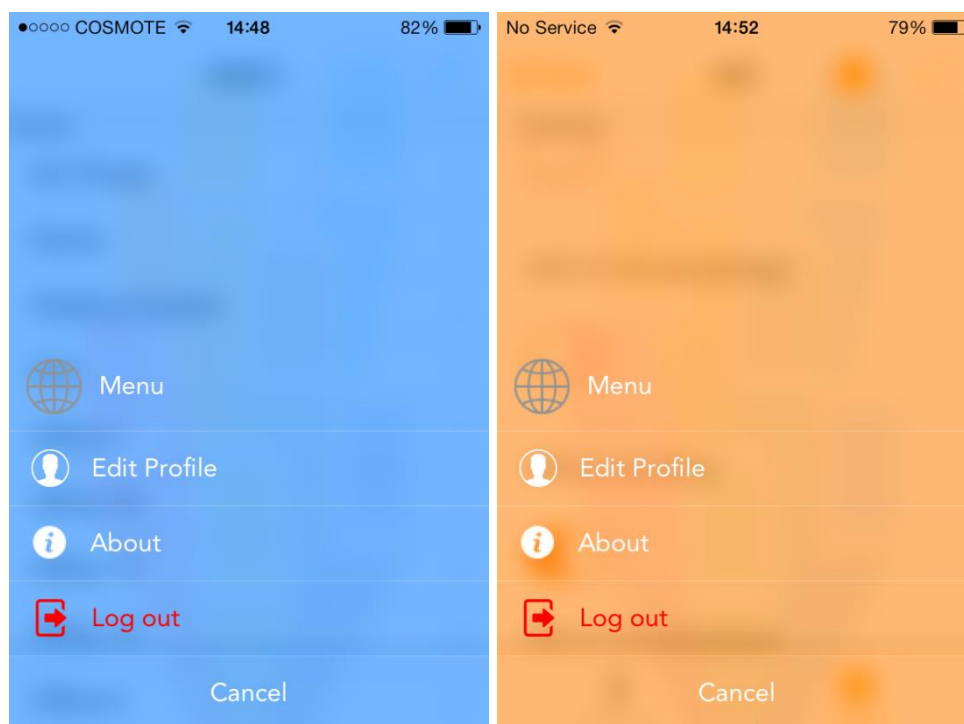
Τέλος, στην *Εικόνα 6.19* απεικονίζονται οι περιπτώσεις προβολής ενός σήματος σε διάρκεια μίας ημέρας αλλά και μίας ώρας, προσαρμόζοντας ανάλογα με την περίπτωση διαφορετική μορφή της ημερομηνίας και ώρας στον άξονα του χρόνου.



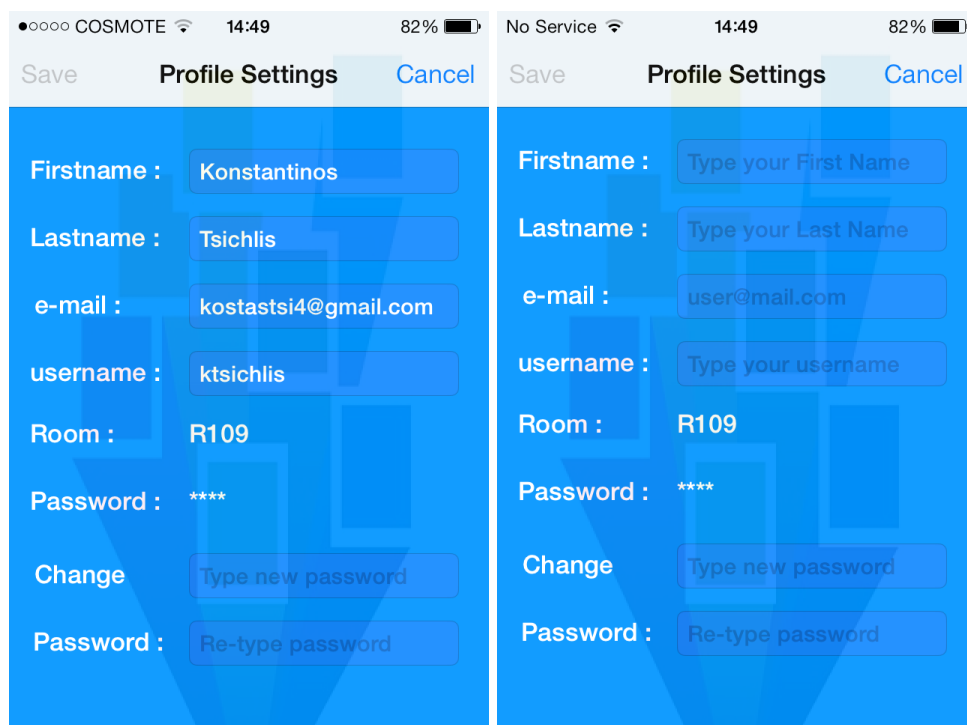
Εικόνα 6.19 – Πάνω. Παράδειγμα προβολής διαγράμματος ενός σήματος σε διάρκεια μίας ημέρας με χρήση της βιβλιοθήκης JpGraph. – Κάτω. Παράδειγμα προβολής διαγράμματος ενός σήματος σε διάρκεια μίας ώρας με χρήση της βιβλιοθήκης JpGraph.

6.4.7 Οθόνη Διαχείρισης Προσωπικού Λογαριασμού

Ο χρήστης οποιαδήποτε στιγμή θέλει να διαχειριστεί τον λογαριασμό του, μπορεί να το κάνει πατώντας το εικονίδιο που βρίσκεται δεξιά στην μπάρα πλοήγησης, και επιλέγοντας από το μενού που φαίνεται στην *Εικόνα 6.20*, την επιλογή **Edit Profile**. Στην οθόνη διαχείρισης του λογαριασμού όλα τα πεδία είναι συμπληρωμένα με τα προσωπικά στοιχεία του χρήστη εκτός από τα πεδία που αφορούν την αλλαγή του κωδικού πρόσβασης. Επίσης όπως φαίνεται στην *Εικόνα 6.21* στο πεδίο του κωδικού για λόγους ασφαλείας, ο κωδικός παρουσιάζεται με αστερίσκους. Επίσης αν ο χρήστης θέλει να αλλάξει τον κωδικό πρόσβασης πρέπει να εισάγει τον νέο κωδικό δύο φορές, ενώ και σε αυτά τα πεδία ο κωδικός πρόσβασης αποκρύπτεται για τον ίδιο λόγο.

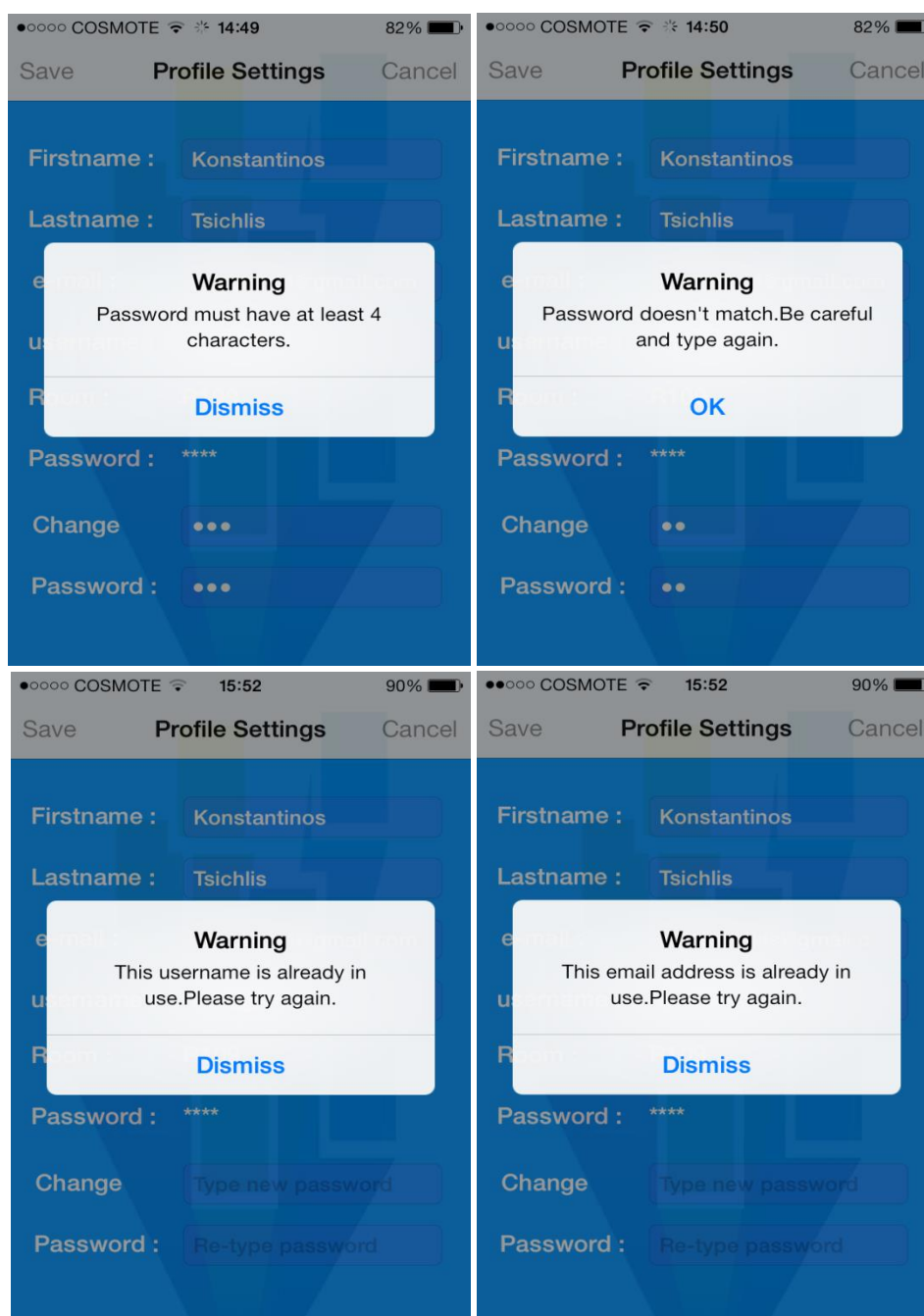


*Εικόνα 6.20 – Οθόνη του μενού που εμφανίζεται όταν επιλέγεται το εικονίδιο **Settings**, δεξιά στην μπάρα πλοήγησης.*



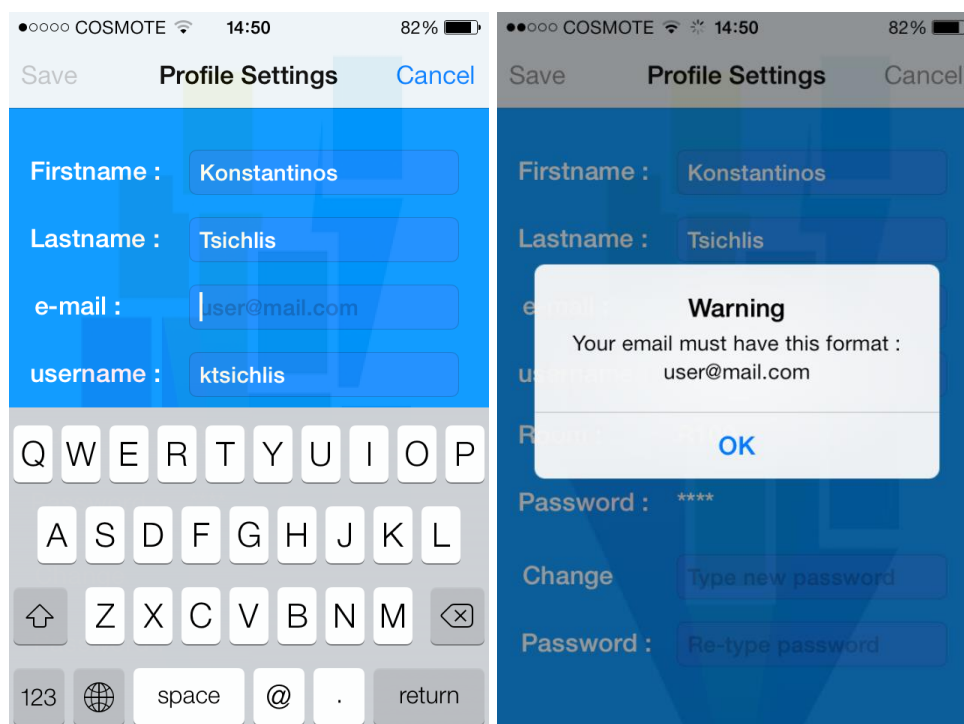
Εικόνα 6.21 – Οθόνη διαχείρισης λογαριασμού χρήστη. Στα δεξιά φαίνονται οι φράσεις προτροπής (placeholder text) που χρησιμοποιούνται σε όλα τα πεδία.

Ο κωδικός πρόσβασης πρέπει να πληροί τον κανόνα εγκυρότητας, δηλαδή να είναι μήκους τουλάχιστον τεσσάρων χαρακτήρων/αριθμών, όπως επίσης πρέπει οι κωδικές λέξεις και των δύο πεδίων να συμφωνούν. Σε διαφορετικές περιπτώσεις εμφανίζονται μηνύματα λάθους όπως φαίνεται στην Εικόνα 6.22. Στην συγκεκριμένη εικόνα φαίνονται και οι περιπτώσεις μοναδικότητας του email και του username, καθώς ο χρήστης προσπάθησε να τα αλλάξει σε κάποια που χρησιμοποιούνται από άλλους.



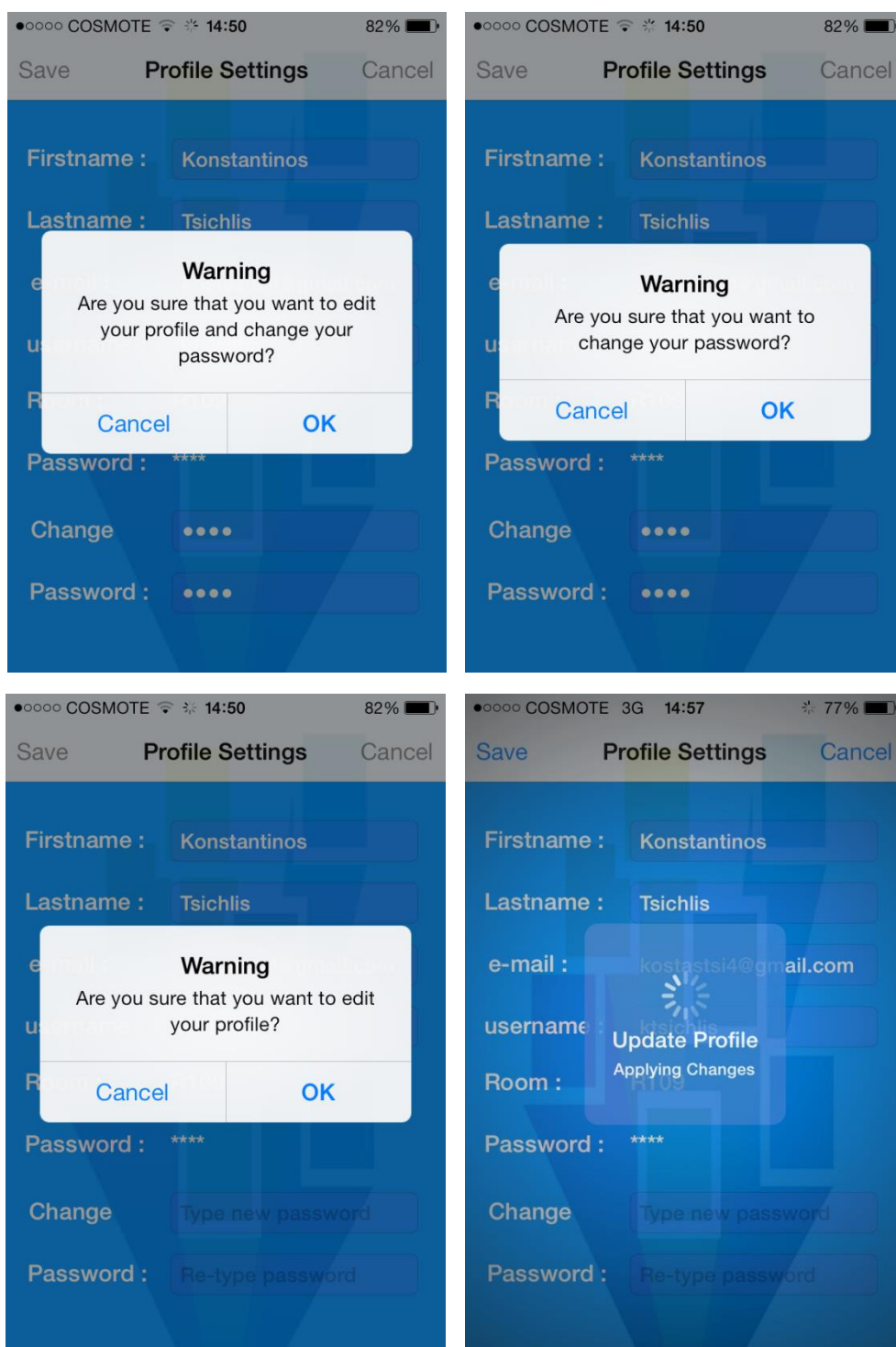
Εικόνα 6.22 – Πάνω Αριστερά. Μήνυμα λάθους σε περίπτωση που ο νέος κωδικός έχει μήκος μικρότερο από 4 χαρακτήρες. – Πάνω Δεξιά. Μήνυμα λάθους σε περίπτωση που στα πεδία εισαγωγής νέου κωδικού δεν έχει εισαχθεί ο ίδιος κωδικός. – Κάτω Αριστερά & Δεξιά. Μηνύματα προειδοποίησης για ήδη χρησιμοποιούμενα username και email αντίστοιχα.

Μήνυμα λάθους όμως εμφανίζεται και σε περίπτωση που ο χρήστης προσπαθήσει να αλλάξει το e-mail του, και αυτό δεν είναι έγκυρο, δηλαδή δεν έχει την απαραίτητη μορφή που πρέπει όπως φαίνεται στην *Εικόνα 6.23*. Παράλληλα τονίζεται ότι οποιοδήποτε από τα πεδία, **First Name**, **Last Name**, **e-mail** και **username** είναι κενά τότε το κουμπί **Save** παραμένει ανενεργό, καθώς δεν επιτρέπεται να είναι κενά αυτά τα πεδία. Επίσης παρατηρεί εύκολα κανείς ότι στο πεδίο **Room** δεν υπάρχει δυνατότητα αλλαγής, καθώς θεωρείται κάτι που αλλάζει και ρυθμίζει ο διαχειριστής.



*Εικόνα 6.23 – Αριστερά. Παράδειγμα όπου δεν είναι ενεργοποιημένο το κουμπί **Save** καθώς το πεδίο **e-mail** δεν είναι συμπληρωμένο. – Δεξιά. Μήνυμα λάθους σε περίπτωση που το e-mail εισήχθη λανθασμένα.*

Συνεπώς όταν ο χρήστης έχει κάνει αλλαγές στον λογαριασμό του, τότε έχει την δυνατότητα να αποθηκεύσει τις αλλαγές πατώντας το πλήκτρο **Save**. Κάθε φορά εμφανίζεται ένα μήνυμα επιβεβαίωσης αποθήκευσης των αλλαγών, ενώ κατά την διάρκεια αποθήκευσης των αλλαγών ένα άλλο μήνυμα κάνει την εμφάνισή του ενημερώνοντας τον χρήστη όπως φαίνεται στην *Εικόνα 6.24*.

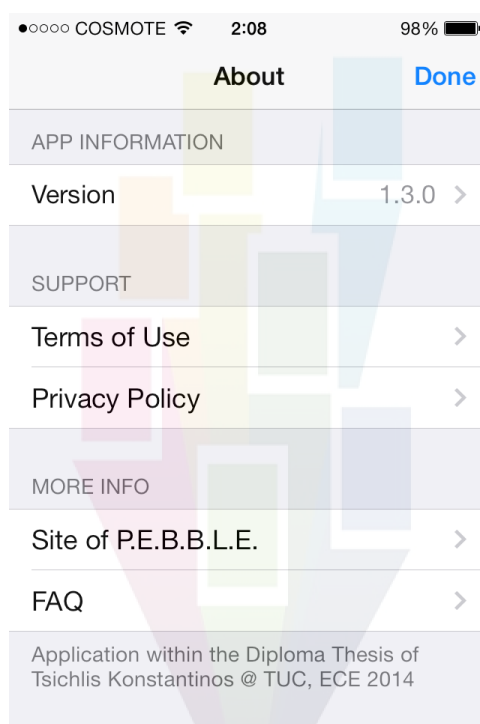


Εικόνα 6.24 – **Αριστερά Πάνω.** Μήνυμα επιβεβαίωσης αλλαγών στον λογαριασμό και αλλαγή κωδικού. – **Δεξιά Πάνω.** Μήνυμα επιβεβαίωσης μόνο αλλαγής κωδικού. – **Αριστερά Κάτω.** Μήνυμα επιβεβαίωσης αλλαγών μόνο στον λογαριασμό. – **Δεξιά Κάτω.** Μήνυμα αποθήκευσης αλλαγών.

6.4.8 Οθόνη Πληροφοριών και Πλοήγησης

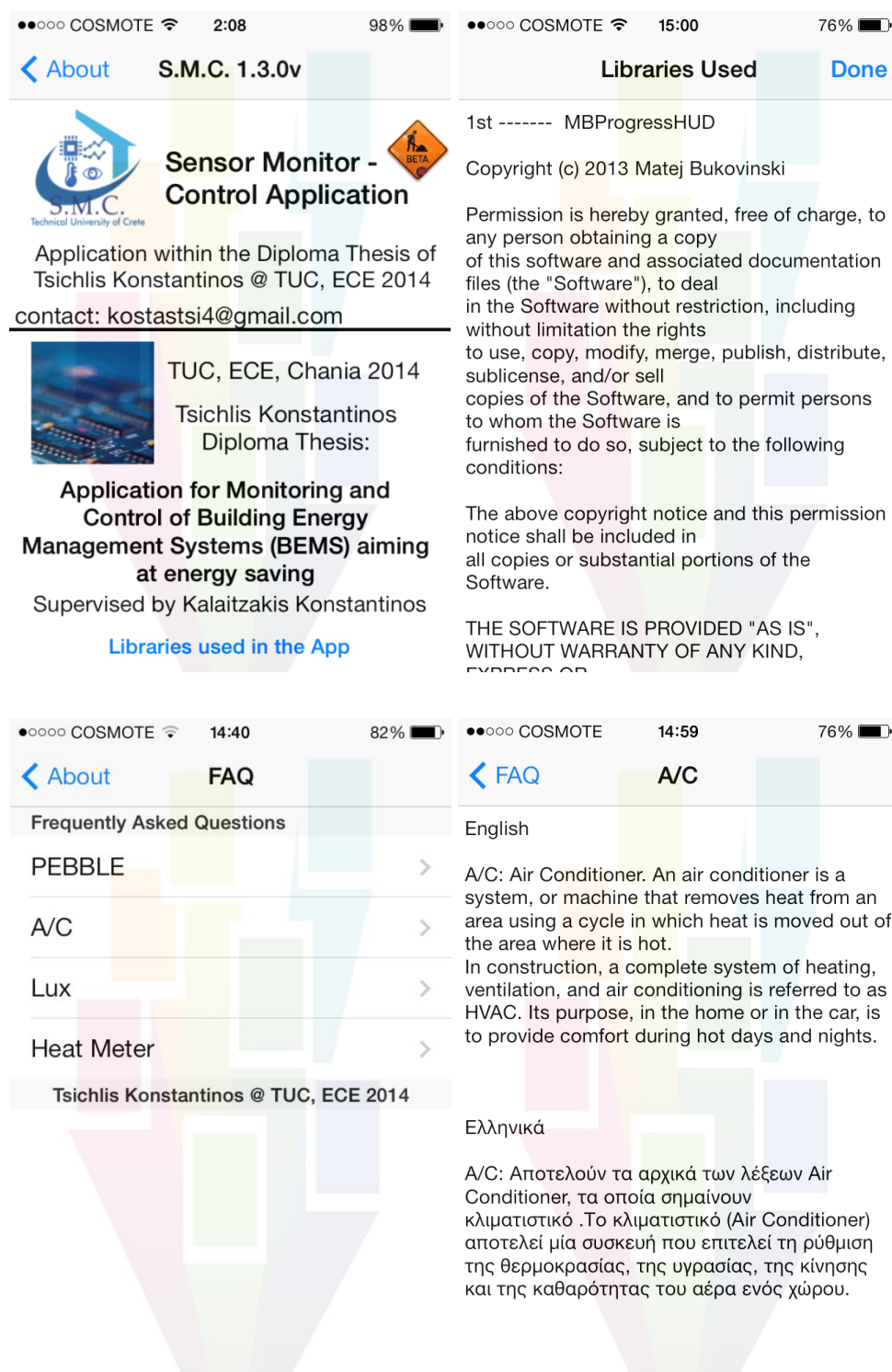
Ο χρήστης έχει την επιλογή είτε είναι συνδεδεμένος, μέσω του εικονιδίου των **Settings** που βρίσκεται δεξιά στην μπάρα πλοήγησης, είτε πριν συνδεθεί, μέσω της αρχικής οθόνης καλωσορίσματος, να ενημερωθεί για την ανάπτυξη, σχεδίαση και υλοποίηση της παρούσας εφαρμογής. Συγκεκριμένα, στην ενότητα **About** συναντά μία νέα οθόνη με τις πληροφορίες που φαίνονται στην *Εικόνα 6.25*.

- Πληροφορίες για την έκδοση της εφαρμογής
- Πληροφορίες για τους όρους χρήσης και την πολιτική απορρήτου
- Πληροφορίες για όρους που χρησιμοποιούνται στην εφαρμογή
- Πλοήγηση στον ισότοπο του προγράμματος PEBBLE αλλά και οπουδήποτε αλλού



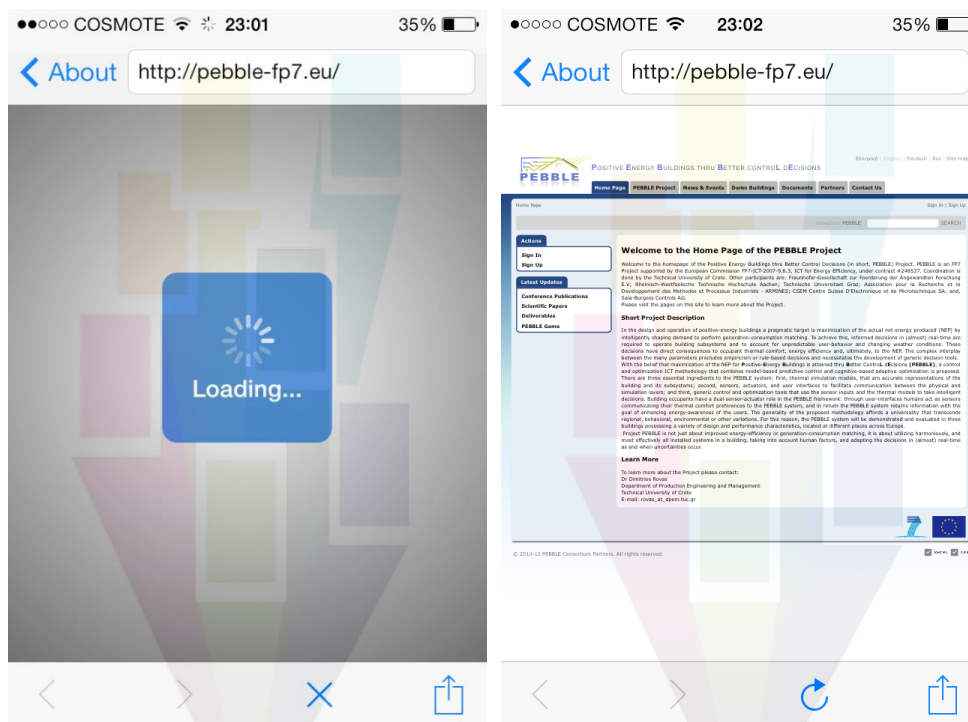
Εικόνα 6.25 – Οθόνη παροχής πληροφοριών στον χρήστη για την εφαρμογή.

Ο χρήστης πατώντας την καρτέλα **Version**, μεταβαίνει στην οθόνη που φαίνεται στην *Εικόνα 6.26*. Είναι μια επισκόπηση για τα πνευματικά δικαιώματα της εφαρμογής, αναφορά για την παρούσα διπλωματική εργασία, η εικόνα λογότυπο της εφαρμογής, ενώ στο τέλος υπάρχει ένας σύνδεσμος που οδηγεί στην οθόνη που παρουσιάζεται στην *Εικόνα 6.26 (Πάνω Δεξιά)*, όπου αναφέρονται οι βιβλιοθήκες που χρησιμοποιήθηκαν κατά τον προγραμματισμό της εφαρμογής. Στην οθόνη **FAQ** επεξηγούνται κυρίως όροι που ενδέχεται ο χρήστης να μην γνωρίζει και χρησιμοποιούνται στην εφαρμογή.

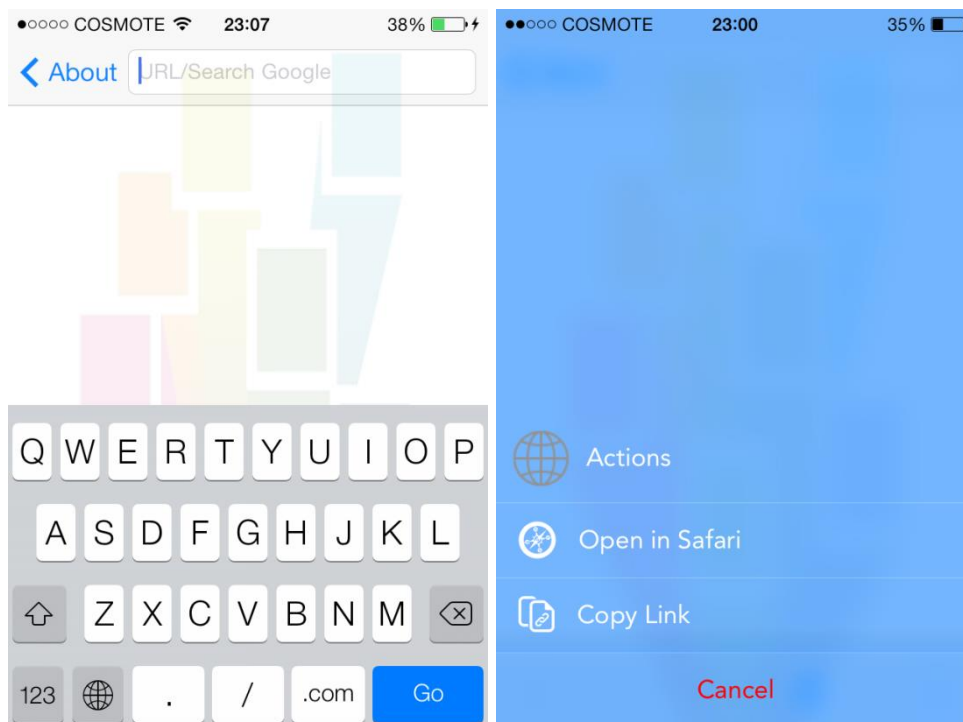


Εικόνα 6.26 – **Αριστερά Πάνω.** Οθόνη πληροφοριών έκδοσης, **Version.** – **Δεξιά Πάνω.** Οθόνη αναφοράς βιβλιοθηκών ως προς τον προγραμματισμό της εφαρμογής. – **Αριστερά Κάτω.** Οθόνη με όρους και συχνές απορίες ενός χρήστη. – **Δεξιά Κάτω.** Οθόνη ανάλυσης όρου A/C.

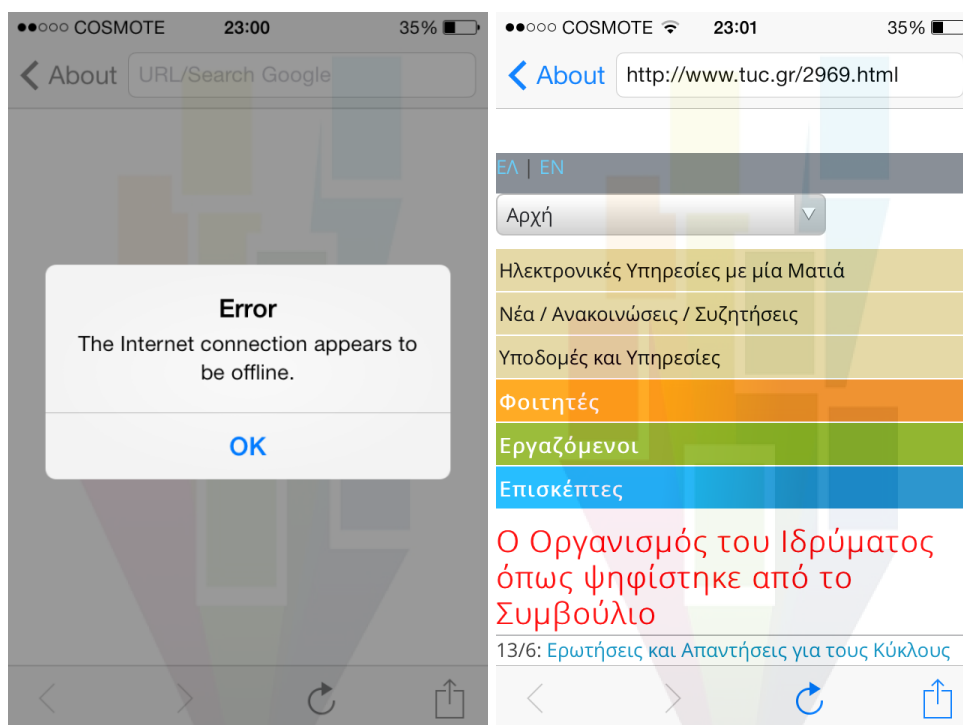
Τέλος, όπως αναφέρθηκε στην παράγραφο [6.3.7], η εφαρμογή έχει σχεδιαστεί ώστε να παρέχει στον χρήστη έναν μίνι browser. Όταν ο χρήστης επιλέξει την καρτέλα **Site of PEBBLE**, μεταφέρεται στο πρόγραμμα πλοήγησης στην ιστοσελίδα του προγράμματος PEBBLE. Στόχος είναι η περαιτέρω ενημέρωση του χρήστη για το πρόγραμμα αυτό. Στην *Εικόνα 6.27* φαίνεται η δομή του. Συγκεκριμένα περιλαμβάνει ένα πεδίο που μπορεί να δεχτεί είτε μια διεύθυνση είτε μια φράση πραγματοποιώντας αναζήτηση στην μηχανή Google. Στο κάτω μέρος διαθέτει κουμπιά με χαρακτηριστικά εικονίδια, παρέχοντας στον χρήστη τις δυνατότητες να γυρίσει σε προηγούμενη ιστοσελίδα ή να μεταβεί σε επόμενη, να ανανεώσει ή να διακόψει την φόρτωση κάποιας ιστοσελίδας, να ανοίξει τον επίσημο browser της Apple, Safari, ή να αντιγράψει τον σύνδεσμο της διεύθυνσης για μετέπειτα χρήση του. Παράλληλα κατά την φόρτωση μίας ιστοσελίδας ή αν προκύψει καποιο σφάλμα, εμφανίζεται ανάλογο μήνυμα όπως παρουσιάζεται στις εικόνες (6.28, 6.29) που ακολουθούν.



Εικόνα 6.27 – Αριστερά. Μήνυμα κατά την φόρτωση της ιστοσελίδας. – Δεξιά. Οθόνη του browser έχοντας φορτώσει την ιστοσελίδα του PEBBLE.



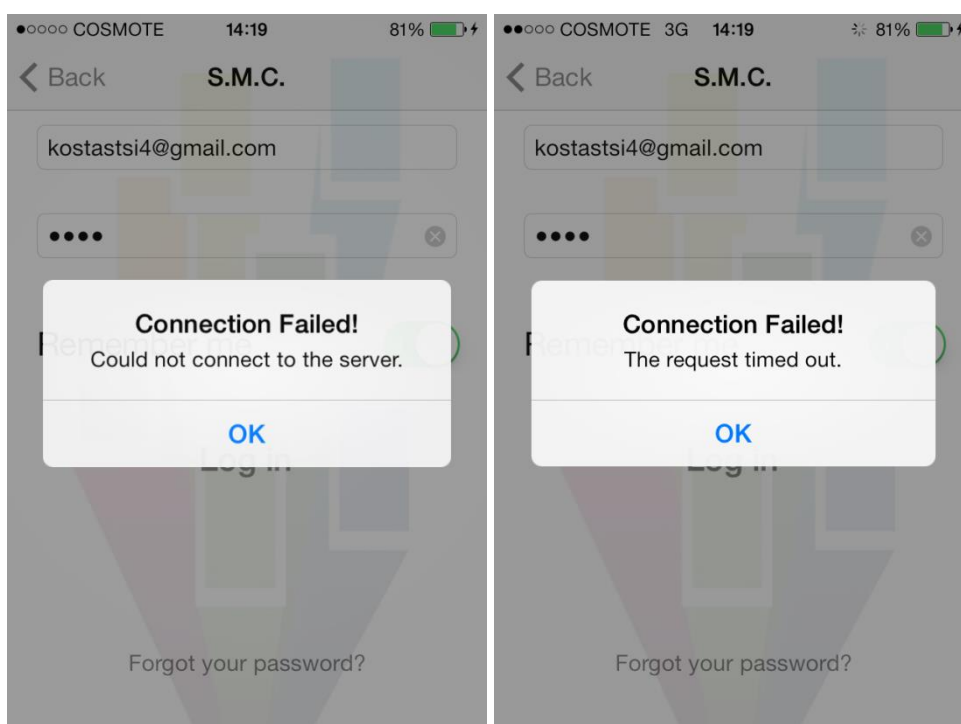
Εικόνα 6.28 – **Αριστερά.** Παράδειγμα μηνύματος προτροπής (placeholder text) στο πεδίο εισαγωγής διεύθυνσης. – **Δεξιά.** Οθόνη ενεργειών για την μετάβαση στην εφαρμογή του Safari ή την αντιγραφή του συνδέσμου διεύθυνσης που βρίσκεται ο browser για μετέπειτα χρήση.



Εικόνα 6.29 – **Αριστερά.** Μήνυμα σφάλματος που προέκυψε κατά την φόρτωση της ιστοσελίδας. – **Δεξιά.** Οθόνη του browser έχοντας φορτώσει την ιστοσελίδα του Πολυτεχνείου Κρήτης.

6.4.9 Περιπτώσεις Προβλήματος Δικτύου ή Server

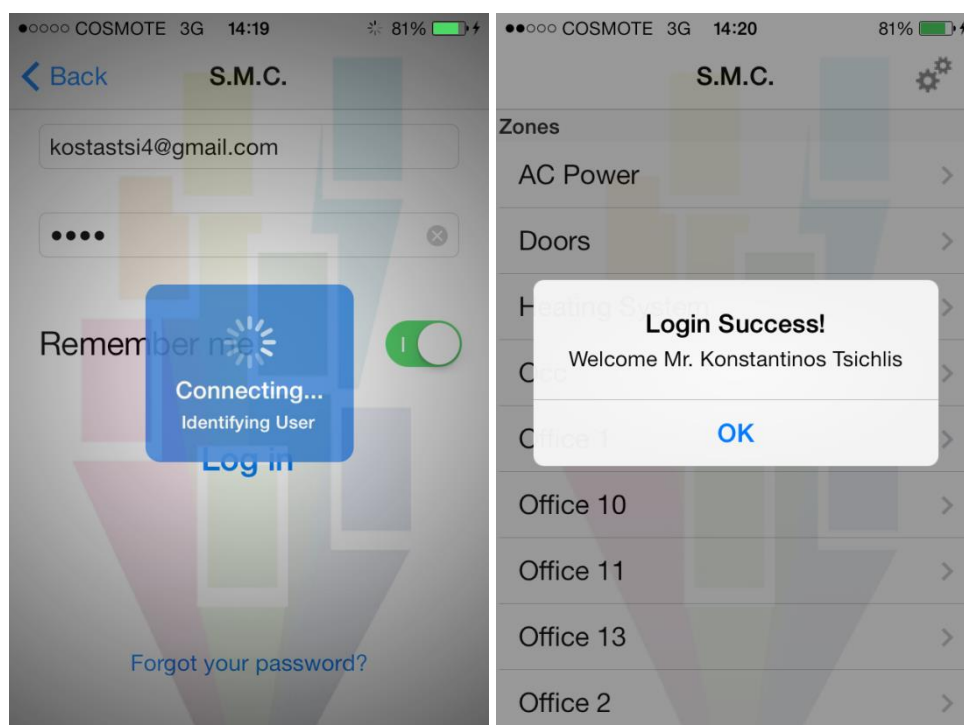
Σε οποιαδήποτε στιγμή συμβεί ένα σφάλμα η εφαρμογή εμφανίζει ανάλογα popup μηνύματα για να ενημερώνει τον χρήστη. Τέτοια προβλήματα μπορεί να είναι είτε προβλήματα σύνδεσης στον server, είτε προβλήματα σύνδεσης στην βάση δεδομένων.



Εικόνα 6.30 – Μηνύματα σφάλματος που προέκυψαν κατά την σύνδεση στον server.

6.4.10 Λειτουργία σύνδεσης με Wi-Fi και 3G

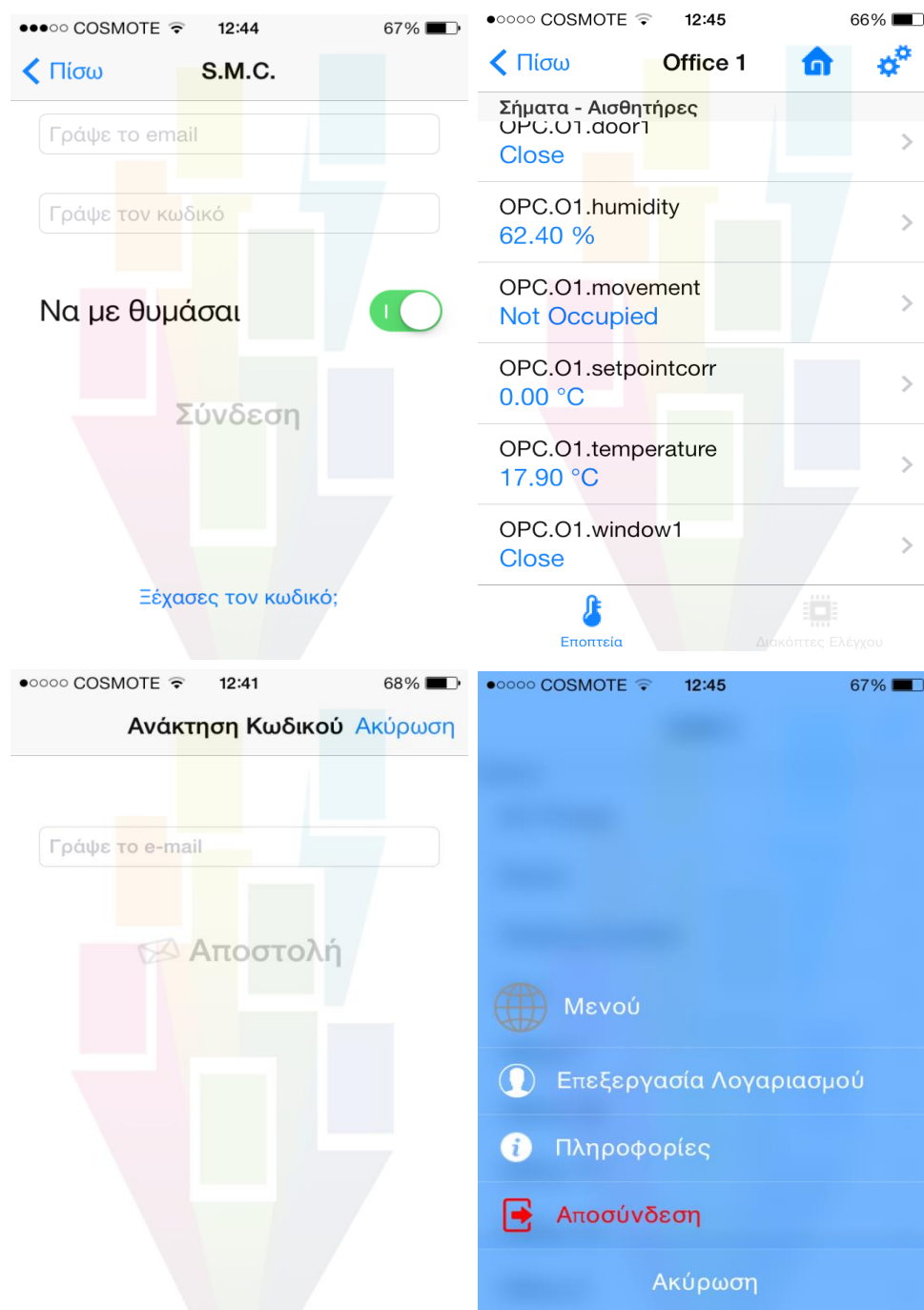
Η εφαρμογή σχεδιάστηκε και υλοποιήθηκε ώστε να έχει τις προδιαγραφές να υποστηρίζει την σύνδεση του κινητού τηλεφώνου με όλες τις πιθανές τεχνολογίες. Υποστηρίζει λοιπόν, σύνδεση μέσω Wi-Fi αλλά και μέσω 3G δικτύου του παρόχου της κινητής τηλεφωνίας. Κάποια χαρακτηριστικά παραδείγματα παρουσιάζονται στην *Εικόνα 6.31*.



Εικόνα 6.31 – Παράδειγμα σύνδεσης του χρήστη στο σύστημα, με χρήση του δικτύου 3G του παρόχου τηλεφωνίας. Εικονίδιο 3G δίπλα από τον πάροχο πάνω αριστερά.

6.4.11 Λειτουργία στην Ελληνική Γλώσσα

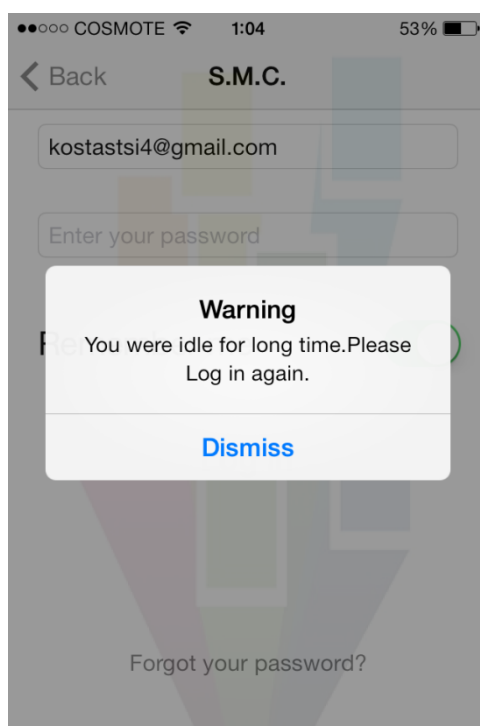
Η εφαρμογή όπως αναφέρθηκε στην παράγραφο [5.7] έχει τις προδιαγραφές να στηρίζει και την γλώσσα των Ελλήνων. Έγινε πλήρης μετάφραση, όλων των απαραίτητων όρων, από την Αγγλική γλώσσα στα Ελληνικά. Κάποια χαρακτηριστικά παραδείγματα παρουσιάζονται στην *Εικόνα 6.32*.



Εικόνα 6.32 – Παραδείγματα οθονών μεταφρασμένες στην Ελληνική γλώσσα.

6.4.12 Λειτουργία Αδράνειας Χρήστη

Άλλη μία σημαντική λειτουργία που υποστηρίζεται από την εφαρμογή είναι αυτή της «αναγκαστικής» αποσύνδεσης του χρήστη που πραγματοποιεί το ίδιο το σύστημα. Όπως αναφέρθηκε και στην παράγραφο [6.3.2], σε περίπτωση που ο χρήστης παραμείνει για πολύ ώρα αδρανής, δηλαδή είτε κλειδώσει το κινητό του τηλέφωνο, είτε βάλει την εφαρμογή στο παρασκήνιο και ασχολείται με άλλη, τότε η εφαρμογή τον αποσυνδέει και τον μεταφέρει στην οθόνη σύνδεσης, εμφανίζοντας παράλληλα ένα ενημερωτικό-προειδοποιητικό μήνυμα. Αυτό το γεγονός οφείλεται στην ασφάλεια που πρέπει να υφίσταται και στηρίζεται στην τεχνολογία των cookies στην μεριά του server. Χαρακτηριστικό παράδειγμα φαίνεται στην *Εικόνα 6.33*. Ο χρόνος αδράνειας του χρήστη έχει τεθεί στα τριάντα πρώτα λεπτά.



Εικόνα 6.33 – Παράδειγμα μηνύματος για βεβαιωμένη αποσύνδεση του χρήστη από το σύστημα λόγω αδράνειας.

7. ΣΥΜΠΕΡΑΣΜΑΤΑ

Με την παρούσα διπλωματική εργασία αναπτύχθηκε μία εφαρμογή σε πλατφόρμα κινητού τηλεφώνου iPhone, που προσφέρει την δυνατότητα παρακολούθησης και ελέγχου οποιουδήποτε συστήματος διαχείρισης ενέργειας κτιρίων (BEMS). Αξίζει αναφοράς ότι αποτελεί την πρώτη εφαρμογή σε λειτουργικό σύστημα iOS, που ασχολείται με το συγκεκριμένο αντικείμενο. Η παρούσα εφαρμογή υλοποιήθηκε επιτυχώς και πλέον μπορεί να βοηθήσει κάθε χρήστη – μέλος ενός συστήματος ενεργειακής διαχείρισης κτιρίων (BEMS), ώστε να εισέλθει σε αυτό, να παρακολουθήσει τις τιμές των αισθητήρων καθ' όλη την διάρκεια του τελευταίου έτους και να εφαρμόσει οποιεσδήποτε αλλαγές επιθυμεί μέσω των διακοπών ελέγχου, δηλαδή των ενεργοποιητών (actuators). Η λειτουργικότητα της εφαρμογής πιστοποιήθηκε με την δοκιμή της στο κτίριο Τεχνικών Υπηρεσιών του Πολυτεχνείου Κρήτης. Η επαναλειτουργία του server που βρίσκεται στο Πολυτεχνείο Κρήτης και παρέχει πρόσβαση στο σύστημα των αισθητήρων και των ενεργοποιητών του κτιρίου Τεχνικών Υπηρεσιών, θα αναδείξει τις πραγματικές δυνατότητες της εφαρμογής. Πρωτίστως όμως, απαιτείται η προσαρμογή του στα δεδομένα της εφαρμογής, όπως περιγράφηκε στην παράγραφο [6.2], ώστε να έχει μεγαλύτερη και ουσιαστικότερη συνεισφορά στην ενεργειακή διαχείριση του κτιρίου.

Οι σχεδιαστικές αρχές και η λειτουργικότητα της εφαρμογής υλοποιήθηκαν με κατάλληλο τρόπο, ώστε να μπορούν εύκολα και γρήγορα να εφαρμοστούν σε οποιαδήποτε άλλη πλατφόρμα κινητού τηλεφώνου (smartphone). Όμως, το πιο σημαντικό, που αναφέρθηκε και προηγουμένως, είναι ότι η συγκεκριμένη εφαρμογή έχει την δυνατότητα να αποτελέσει έναν μηχανισμό για οποιοδήποτε άλλο σύστημα διαχείρισης ενέργειας κτιρίων, BEMS. Ουσιαστικά η παρούσα εργασία κατάφερε να υλοποιήσει τον στόχο της, σύμφωνα με τον οποίο έπρεπε να υλοποιηθεί μία εφαρμογή, που να αποτελεί βασικό και αξιόπιστο μέσο αξιολόγησης τέτοιου είδους συστημάτων, συμβάλλοντας παράλληλα στην βελτιστοποίησή τους.

Επιπροσθέτως, ο εξυπηρετητής «REST Webserver» υλοποιήθηκε με σχεδιαστικά πρότυπα, τα οποία υποστηρίζονται από την πλειοψηφία των συστημάτων επικοινωνίας που ενδέχεται να λειτουργήσουν ως πελάτες (clients). Συγκεκριμένα ο σχεδιασμός της

βάσης δεδομένων³⁷, ER schema, που έγινε από την ομάδα του PEBBLE και δόθηκε αυτούσια για την υλοποίηση της παρούσας εργασίας, αλλά και η υλοποίηση του εξυπηρετητή (server) με χρήση της τεχνολογίας JSON για την μεταφορά των δεδομένων, μπορούν να χρησιμοποιηθούν ανεξάρτητα από την εφαρμογή **S.M.C.** σε πολλές άλλες πλατφόρμες. Η υλοποίησή του δηλαδή, έγινε σύμφωνα με ένα γενικό μοντέλο που εφαρμόζεται στα δίκτυα κινητής τηλεφωνίας.

Η προσπάθεια εξοικονόμησης ενέργειας των κτιρίων διαθέτει ένα επιπλέον ισχυρό όπλο. Η ύπαρξη μίας εφαρμογής που βοηθάει τον ανθρώπινο παράγοντα να επεμβαίνει γρήγορα, εύκολα και δυναμικά στο σύστημα, δίνει δυνατότητα για αποτελέσματα που διαθέτουν μεγαλύτερη ακρίβεια, αποτελεσματικότητα και αποδοτικότητα. Ο ανθρώπινος παράγοντας παρέχει σημαντική βοήθεια στα μοντέλα που εφαρμόζονται για την βελτίωση της θερμικής άνεσης των κτιρίων, και κατ' επέκταση για την εξοικονόμηση ενέργειας σε αυτά. Επομένως, η παρούσα εφαρμογή αποτελεί σημαντικό εργαλείο για την μετατροπή τους σε κτίρια μηδενικού ή ακόμα και θετικού ισοζυγίου ενέργειας και κυρίως για την προστασία του περιβάλλοντος από τους ρύπους της υπερβολικής παραγωγής (ηλεκτρικής) ενέργειας.

³⁷ Η αναφορά συμπεριλαμβάνει δύο βάσεις δεδομένων. Την βάση που διατηρεί τα δεδομένα των χώρων του κτιρίου Τεχνικών Υπηρεσιών, τους αισθητήρες, τα σήματα ελέγχου (actuators) αλλά και την βάση δεδομένων των τιμών όλων αυτών των σημάτων (historical_data).

8. ΠΡΟΤΑΣΕΙΣ ΒΕΛΤΙΩΣΗΣ

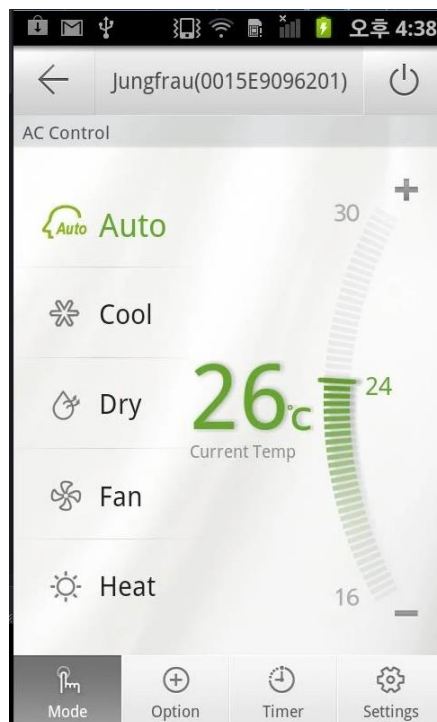
Μετά από την υλοποίηση της εφαρμογής και την δοκιμή της λειτουργίας της σε πραγματική συσκευή κινητού τηλεφώνου iPhone, διαπιστώθηκαν κάποια σημεία που ίσως επιδέχονται βελτίωσης, αλλά και κάποιες προσθήκες που ενδεχομένως φανούν χρήσιμες μελλοντικά. Στο σημείο αυτό αξίζει να γίνει μια μικρή επισκόπηση, λοιπόν, των προτάσεων για βελτίωση των λειτουργιών αλλά και του γραφικού περιβάλλοντος χρήσης (user interface) της εφαρμογής μελλοντικά.

1. Λειτουργία εγγραφής

Όπως αναφέρθηκε στο 6^ο Κεφάλαιο, η λειτουργία της εγγραφής του χρήστη στο σύστημα δεν παρέχεται από την εφαρμογή. Ο βασικός λόγος είναι ότι η υλοποίησή της έγινε στα πλαίσια του προγράμματος PEBBLE, το οποίο σε πρώτο βαθμό τουλάχιστον δεν προβλέπει κάτι τέτοιο. Όμως, είναι μια λειτουργία που υπάρχει πάντα σε δίκτυα κοινωνικής δικτύωσης. Επομένως αν μελλοντικά θεωρηθεί ότι η παρούσα εφαρμογή μπορεί να αποκτήσει χαρακτήρα κοινωνικής δικτύωσης, τότε υπάρχει όλη η βασική υποδομή, σε επίπεδο λογισμικού (κώδικα), ώστε εύκολα να προστεθεί.

2. Βελτίωση User Interface οθόνης σημάτων ελέγχου

Είναι προφανές ότι το γραφικό περιβάλλον της οθόνης των σημάτων ελέγχου, το οποίο παρέχει ουσιαστικά την δυνατότητα ελέγχου των air-conditions στα γραφεία, επιδέχεται βελτίωσης. Για παράδειγμα μπορεί να υλοποιηθεί έχοντας την μορφή ενός τηλεκοντρόλ για air-condition ή κάτι παρόμοιο με αυτό που φαίνεται στην εικόνα παρακάτω. Τους τελευταίους μήνες έχουν κάνει την εμφάνισή τους στην αγορά, εφαρμογές για συσκευές κινητών τηλεφώνων που παρέχουν έλεγχο ηλεκτρονικών συσκευών, όπως το air-condition, μέσω του διαδικτύου.



Εικόνα 7.1 – Παράδειγμα οθόνης από εφαρμογή που παρέχει απομακρυσμένο έλεγχο σε air-condition (Smart Air Conditioner App by Samsung Electronics CO.LTD.)

3. Λειτουργία περιορισμένης πρόσβασης χωρίς σύνδεση

Μία προσθήκη που θα μπορούσε να γίνει, είναι αυτή της λειτουργίας χωρίς σύνδεση. Συγκεκριμένα, η εφαρμογή θα μπορούσε να παρέχει στον χρήστη την δυνατότητα να μπορεί να παρακολουθήσει κάποιον αισθητήρα χωρίς να υπάρχει πρόσβαση στο διαδίκτυο, δηλαδή μέσω σύνδεσης Wi-Fi ή 3G. Πιο αναλυτικά, οι δυνατότητες που θα έχει ο χρήστης θα είναι απλά να παρακολουθήσει κάποιον αισθητήρα αλλά και το διάγραμμα των τιμών του για χρονικό διάστημα παλιότερο από την δεδομένη χρονική στιγμή. Δεν θα έχει την δυνατότητα είτε να αλλάξει το προφίλ του, είτε να πειράξει κάποιον διακόπτη ελέγχου αφού δεν θα υπάρχει σύνδεση. Ο σκοπός αυτής της λειτουργίας θα είναι απλά να δίνει στον χρήστη την δυνατότητα να ανατρέχει στα διαγράμματα των αισθητήρων, ώστε να δει τιμές τους σε δεδομένη παλαιότερη χρονική στιγμή. Η συγκεκριμένη λειτουργία μπορεί να υλοποιηθεί με χρήση της βιβλιοθήκης Core Data και της SQLite βάσης δεδομένων που διαθέτει κάθε συσκευή iPhone και λογισμικό iOS.

4. Εμπλουτισμός δυνατοτήτων διαγραμμάτων

Γενικότερα ο τρόπος λειτουργίας των διαγραμμάτων επιδέχεται κάποιες βελτιώσεις. Προτείνεται λοιπόν, μελλοντικά να αναβαθμιστούν οι επιλογές που παρέχονται στον χρήστη. Για παράδειγμα θα μπορούσε ο χρήστης να έχει την δυνατότητα να επιλέξει δύο ή περισσότερους αισθητήρες και να προβάλλει στο ίδιο διάγραμμα τις τιμές τους, ώστε να τους συγκρίνει. Επίσης θα μπορούσε να υπάρχει η επιλογή «μεταπήδηση σε συγκεκριμένη ημερομηνία», δηλαδή ο χρήστης να επιλέγει για κάποιο αισθητήρα – σήμα μία συγκεκριμένη ημερομηνία που τον ενδιαφέρει η τιμή του. Αυτή την στιγμή υπενθυμίζεται ότι ο χρήστης μεταπηδάει σε ημερομηνίες μέσω των πλήκτρων **Previous** και **Next** στο κάτω μέρος της οθόνης απεικόνισης διαγραμμάτων [6.4.6].

5. Μεγαλύτερη ασφάλεια-προστασία στον server

Είναι πολύ σημαντικό να προστεθούν περισσότερα επίπεδα ασφάλειας στον server. Η προστασία των προσωπικών δεδομένων και γενικότερα του συνολικού συστήματος από επιθέσεις κακόβουλων λογισμικών ή hackers, είναι βασική προτεραιότητα κάθε διαδικτυακού συστήματος. Επομένως η υλοποίηση ενός **HTTPS** πρωτοκόλλου [55] θεωρείται επιβεβλημένη μελλοντικά, όπως επίσης και χρήση κάποιου είδους πρωτοκόλλου πιστοποίησης, όπως το **OAuth** [56].

Γενικότερα, εύκολα κανείς αντιλαμβάνεται ότι η εφαρμογή **S.M.C.** που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας, βρίσκεται στα πρώτα της βήματα και προσφέρει πολλές ευκαιρίες για μελλοντική βελτίωση και εξέλιξη.

9. ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] “ΕΝΕΡΓΕΙΑ ΚΑΙ ΚΤΙΡΙΟ – ΕΝΕΡΓΕΙΑΚΗ ΚΑΤΑΝΑΛΩΣΗ ΚΤΙΡΙΩΝ”, Μάνθος Σανταμούρης, available online: http://www.buildings.gr/greek/aiforos/exikonomisi/m_santamouris.htm
- [2] “Positive Energy Buildings thru Better Control Decisions, P.E.B.B.L.E.”, available online: <http://pebble-fp7.eu/>
- [3] “Apple introduces HomeKit for iOS 8”, available online: <http://www.cnet.com/news/apple-introduces-homekit-for-ios-8/>
- [4] Samsung designs smartphone app to turn off lights and home appliances”, <http://www.dailymail.co.uk/sciencetech/article-2534751/No-flicking-switches-Samsung-designs-smartphone-app-turn-lights-home-appliances.html>
- [5] “The rise and rise of smartphones”, 4 February, 2014, *Economist*, available online: <http://www.economist.com/blogs/graphicdetail/2014/02/daily-chart>
- [6] “Smartphone Users Worldwide Will Total 1.75 Billion in 2014”, *survey by eMarketer*, 16 January, 2014, available online: <http://www.emarketer.com/Article/Smartphone-Users-Worldwide-Will-Total-175-Billion-2014/1010536>
- [7] “Εξυπνο το ένα στα δύο κινητά που πωλούνται στην ελληνική αγορά”, 21 May 2014, available online: <http://www.tovima.gr/society/article/?aid=598430>
- [8] “More time on internet through mobile phones than PCs”, *survey by Nielsen*, Feb 2014, available online: <http://marketingland.com/nielsen-time-accessing-internet-smartphones-pcs-73683>
- [9] Dimitrios V. Rovas, K. Katsigarakis, Giorgos D. Kontes, Georgios I. Giannakis, Georgios N. Lilis, “A Sense-Think-Act methodology for intelligent building energy management”, *Proceedings of the 30th CIB W78 International Conference – October 2013, Beijing, China*

- [10] D. Kolokotsa, D. Rovas, E. Kosmatopoulos, K. Kalaitzakis, “A roadmap towards intelligent net zero- and positive-energy buildings”, *Solar Energy Volume85, Issue12, December 2011, Pages 3067-3084*
- [11] D. Rovas, K. Katsigarakis, E. Kosmatopoulos, “Model-assisted control design to improve building energy performance”, *ICT for Sustainable Homes, Nice, France 24th October, 2011*
- [12] M. F. Pichler, A. Droscher, H. Schranzhofer, Ana Contantin, Max Huber, G. Giannakis, N. Exizidou and D.V. Rovas, “Simulation Model Improvements”, *Positive-Energy Buildings through Better control decisions, November 7, 2011*
- [13] “Global Apple iPhone sales from 3rd quarter 2007 to 2nd quarter 2014 (in million units)”, available online: <http://www.statista.com/statistics/263401/global-apple-iphone-sales-since-3rd-quarter-2007/>
- [14] Professor T. Luhrmann, “iPhone Addiction Survey” by *Stanford University*, available online: <http://blog.appleworldhellas.com/τα-στατιστικά-της-apple-που-έδωσε-ο-tim-cook-στην-ε/>
- [15] “Τα στατιστικά της Apple που έδωσε ο Tim Cook στην εισαγωγική του ομιλία στο WWDC14”, available online: <http://www.enet.gr/?i=news.el.episthmh-technologia&id=139631>
- [16] J.D. Power and Associates, “Η Apple καταλαμβάνει την πρώτη θέση στην ικανοποίηση των καταναλωτών”, available online: <http://www.iphonehellas.gr/38808/apple-satisfaction-top/>,
“96% των χρηστών είναι ικανοποιημένοι με το iPhone 4S” by *Research firm Change Wave*, available online: <http://www.iphonehellas.gr/32034/iphone-4s-satisfaction-rate/>
- [17] Apple, “iPhone 5 Pre-Orders Top Two Million in First 24 Hours”, available online: <https://www.apple.com/pr/library/2012/09/17iPhone-5-Pre-Orders-Top-Two-Million-in-First-24-Hours.html>,
Apple, “iPhone 5 First Weekend Sales Top Five Million”, available online: <http://www.apple.com/pr/library/2012/09/24iPhone-5-First-Weekend-Sales-Top-Five-Million.html>
- [18] “Sales of Apple iOS devices exceed 800 million units”, available online: <http://www.kitguru.net/apple/anton-shilov/sales-of-apple-ios-devices-exceed-800-million-units/>
- [19] “Τα μυστικά των οθονών αφής”, 24 *Ιανουαρίου 2011*, available online: <http://www.onlinemagazine.gr/?p=216>
- [20] Apple, “Apple - iOS 7 - Design”, available online: <http://www.apple.com/ios/design/>

- [21] Apple, “iOS Dev Center – Apple Developer”, available online: <https://developer.apple.com/devcenter/ios/index.action>
- [22] Apple, “iOS Developer Library, Documentation and References”, available online: <https://developer.apple.com/library/ios/navigation/>
- [23] Apple, “The Objective-C Programming Language, Reference Manual”, available online: <https://developer.apple.com/library/mac/documentation/cocoa/conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>
- [24] Apple, “Advanced Memory Management Programming Guide”, available online: <https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual/MemoryMgmt/Articles/MemoryMgmt.html>
- [25] “Objective-C: Η Τρίτη πιο δημοφιλής γλώσσα προγραμματισμού”, available online: <http://iappshellas.gr/?p=14690>, “Objective-C: Η γλώσσα προγραμματισμού της Apple 3^η στην λίστα της TIOBE”, *Ιούλιος 2012*, available online: <http://www.myphone.gr/forum/showthread.php?t=338994>
- [26] “Some Nice Features of the Objective-C Language”, *March 13, 2008 by Philippe Mougin*, available online: <http://pmougin.wordpress.com/2008/03/13/some-nice-features-of-the-objective-c-language/>
- [27] Apple, «Objective-C Programming: Model-View-Controller”, available online: <https://developer.apple.com/library/ios/documentation/general/conceptual/CocoaEncyclopedia/Model-View-Controller/Model-View-Controller.html>, Wikipedia, ”Model View Controller, MVC model”, available online: <http://en.wikipedia.org/wiki/Model-view-controller>
- [28] Apple, ”Xcode”, available online: <https://developer.apple.com/xcode/>
- [29] Stanford University, Computer Science Department, “CS 193P iPhone Application Development Course”, *included in syllabus at 2009*, available on iTunes U and online: <http://www.stanford.edu/class/cs193p/cgi-bin/drupal/>
- [30] Ray Wenderlich, “iPhone tutorials”, available online: <http://www.raywenderlich.com/tutorials>
- [31] U.S. Copyright Office decision, “Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies”, <http://www.cnet.com/news/on-call-go-ahead-and-jailbreak-its-legal-now/>

- [32] YouTube βίντεο, “How to fake code-sign Xcode 5 for iOS 7.0.4 on Mackintosh”, <https://www.youtube.com/watch?v=AqDVXh2S1P0>
- [33] Apple, “Quartz Core Framework Reference”, available online: <https://developer.apple.com/library/mac/documentation/graphicsimaging/reference/QuartzCoreRefCollection/index.html>
- [34] “iOS Frameworks”, available online: <http://iosframeworks.com/frameworks?tagged=chart>
- [35] Github, other free iOS Chart Libraries, available online: <https://github.com/Jawbone/JBChartView>, <https://github.com/Boris-Em/BEMSimpleLineGraph>
- [36] “Interactive and customizable iOS Charts: ShinobiControls”, available online: <http://www.shinobicontrols.com/ios/shinobicharts>
- [37] “JPGRAPH: Most Powerful PHP-driven charts”, available online: <http://jpgraph.net/>
- [38] “Core Plot : Cocoa plotting framework for OS X and iOS”, available online: <https://code.google.com/p/core-plot/>
- [39] Github, “MBProgressHUD, an iOS activity indicator view” *by matej*, available online: <https://github.com/matej/MBProgressHUD>
- [40] Github, “An alternative to the UIAlertController inspired by the Spotify app” *by fastred*, available online: <https://github.com/fastred/AHKActionSheet>
- [41] “Εισαγωγή στα Web Services”, available online: <http://www.it.uom.gr/project/soap/Theory/introduction.html>
- [42] W3Schools, “Web Services Introduction”, available online: http://www.w3schools.com/webservices/ws_intro.asp
- [43] W3Schools, “XML tutorial”, available online: <http://www.w3schools.com/xml/default.ASP>
- [44] “JSON”, available online: <http://www.json.org/>
- [45] Wikipedia, “Representational state transfer”, available online: http://en.wikipedia.org/wiki/Representational_state_transfer,
- [46] “RESTful Web services: The basics”, *by developer Works*, available online: <http://www.ibm.com/developerworks/webservices/library/ws-restful/>

- [47] “PHP : Hypertext Preprocessor”, available online: <http://www.php.net/>, “Η βίβλος της PHP – Μέρος 1^ο - Εισαγωγή”, available online: <http://dide.flo.sch.gr/Plinet/Tutorials-PHP-Bible/Tutorials-PHP-Bible-1.html>
- [48] “MySQL: The world’s most popular open source database”, available online : <http://www.mysql.com/>
- [49] “Okeanos | Home ”, available online: <https://okeanos.grnet.gr/home/>, “Synnefo | Open Source Cloud Software”, available online: <https://www.synnefo.org/>
- [50] Ubuntu Forum,”Στήσιμο LAMP Stack και Vhosts σε Ubuntu.”, available online: <https://forum.ubuntu-gr.org/viewtopic.php?f=9&t=28526>
- [51] Github, “PHPMailer”, *The classic email sending library for PHP by Synchro*, available online: <https://github.com/PHPMailer/PHPMailer>
- [52] “Apache Software Foundation”, available online: <http://www.apache.org/>
- [53] Navicat, “Navicat Premium | Connects to multiple databases on a single GUI”, available online: <http://www.navicat.com/products/navicat-premium>
- [54] Github, “SocketRocket”, *A conforming Objective-C WebSocket client library by square*, available online: <https://github.com/square/SocketRocket>
- [55] “HTTPS – Τι είναι και γιατί πρέπει να μας νοιάζει”, available online: <http://www.netcomm.gr/https-ti-einai-kai-giati-prepei-na-mas-noiazai>
- [56] “Εισαγωγή στο OAuth”, available online: <http://www.webz.gr/isagogi-sto-oauth.html>
- [57] Emmanouil Alibertis, “Community RF Sensing”, *A Thesis Submitted in partial fulfillment of the requirements for the diploma degree of Electronic and Computer Engineering, October 2012, TUC, ECE Department*
- [58] Σκλάβος Σμάλης, “Κοινωνικό Δίκτυο Ανασκόπησης Ταξιδιών βασισμένο στο Google earth”, *Διπλωματική Εργασία, 2012, Πολυτεχνείο Κρήτης, HMMY*