

Πτυχιακή Εργασία

ΠΛΗΡΟΦΟΡΙΑΚΟ ΣΥΣΤΗΜΑ ΓΙΑ ΕΤΑΙΡΙΑ ΚΙΝΗΤΗΣ ΤΗΛΕΦΩΝΙΑΣ
ΖΟΥΡΑΡΑΚΗΣ ΕΜΜΑΝΟΥΗΛ

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ	4
ΠΙΝΑΚΑΣ ΠΙΝΑΚΩΝ.....	6
ΠΕΡΙΛΗΨΗ	6
ABSTRACT	6
1 ΣΚΟΠΟΣ ΤΗΣ ΕΡΓΑΣΙΑΣ	7
2 ΕΙΣΑΓΩΓΗ	7
2.1 Η αλληλεπίδραση του χρήστη με το Διαδίκτυο.....	7
2.2 Οι γλώσσες προγραμματισμού σήμερα.....	8
2.3 Οι κύριες τεχνολογίες ανάπτυξης ιστοσελίδων	9
2.4 Το υπάρχον μοντέλο ανάπτυξης.....	10
3 ΥΠΑΡΧΟΥΣΕΣ ΤΕΧΝΟΛΟΓΙΕΣ	11
3.1 Server side interpreters: Php	11
3.1.1 Σχεδιαστική Φιλοσοφία	11
3.2 Java & Java Server Pages.....	12
3.2.1 Σχεδιαστική Φιλοσοφία	12
3.2.2 Java Server Pages	13
3.3 Asp & Asp.net.....	14
3.3.1 Σχεδιαστική Φιλοσοφία και Συντακτικό	15
3.3.2 Το .Net framework και η καινοτομική αρχιτεκτονική της asp.net.....	15
3.4 Javascript.....	16
3.4.1 Σχεδιαστική φιλοσοφία.....	16
3.4.2 Σύνταξη και χαρακτηριστικά	17
3.5 Flash & Flex.....	18
3.5.1 Γραφικές και πολυμεσικές δυνατότητες.....	19
3.5.2 Ο προγραμματισμός στο flash	20
4 ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΒΑΣΩΝ ΔΕΔΟΜΕΝΩΝ.....	21
4.1 ΧΑΜΡΡ.....	21
4.1.1 Εγκατάσταση ΧΑΜΡΡ	21
5 ΔΙΑΓΡΑΜΜΑ ΟΝΤΟΤΗΤΩΝ ΣΧΕΣΕΩΝ	25
6 ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ.....	25
7 ΔΗΜΙΟΥΡΓΙΑ ΒΑΣΗΣ.....	27

8	ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ ΣΥΣΤΗΜΑΤΟΣ	35
8.1	Βασικά Χαρακτηριστικά Σελίδων Συστήματος.....	35
8.2	Login - Logout.....	37
8.3	Χρήστες.....	38
8.3.1	Αρχική Οθόνη Χρηστών.....	38
8.3.2	Δημιουργία Χρήστη.....	39
8.3.3	Διαγραφή Χρήστη.....	40
8.4	Τμήματα	41
8.4.1	Αρχική Οθόνη Τμημάτων	41
8.4.2	Δημιουργία Τμήματος.....	41
8.4.3	Διαγραφή Τμήματος.....	43
8.5	Υπάλληλοι.....	44
8.5.1	Αρχική Οθόνη Υπαλλήλων.....	44
8.5.2	Δημιουργία Υπαλλήλου.....	44
8.5.3	Διαγραφή Υπαλλήλου	46
8.6	Τύποι Υπαλλήλων (Διοικ./Τεχν.)	47
8.6.1	Αρχική Οθόνη Τύπων υπαλλήλων.....	47
8.6.2	Δημιουργία Τύπου Υπαλλήλων	48
8.6.3	Διαγραφή Τύπου Υπαλλήλου.....	49
8.7	Διευθυντές	51
8.7.1	Αρχική Οθόνη Διευθυντών.....	51
8.7.2	Δημιουργία Διευθυντή.....	51
8.7.3	Διαγραφή Διευθυντή.....	53
8.8	Τηλεφωνικά Προγράμματα.....	54
8.8.1	Αρχική Οθόνη Τηλεφωνικών Προγραμμάτων.....	54
8.8.2	Δημιουργία Τηλεφωνικού Προγράμματος	55
8.8.3	Διαγραφή Τηλ.Προγράμματος	56
8.9	Οικογενειακά Μέλη	58
8.9.1	Αρχική Οθόνη Οικογενειακών Μελών	58
8.9.2	Δημιουργία Οικογενειακού Μέλους.....	58
8.9.3	Διαγραφή Οικογενειακού Μέλους	59
8.10	Πελάτες	61
8.10.1	Αρχική Οθόνη Πελατών.....	61
8.10.2	Δημιουργία Πελάτη.....	61

8.10.3	Διαγραφή Πελάτη.....	63
8.11	Τηλεφωνικοί Αριθμοί.....	64
8.11.1	Αρχική Οθόνη Τηλεφωνικών Αριθμών.....	64
8.11.2	Δημιουργία Τηλεφωνικού Αριθμού.....	64
8.11.3	Διαγραφή Τηλ. Αριθμού.....	66
8.12	Λογαριασμοί.....	67
8.12.1	Αρχική Οθόνη Λογαριασμών.....	67
8.12.2	Δημιουργία Λογαριασμού.....	67
8.12.3	Διαγραφή Λογαριασμού.....	69
8.13	Κεραίες.....	70
8.13.1	Αρχική Οθόνη Κεραιών.....	70
8.13.2	Δημιουργία Κεραίας.....	70
8.13.3	Διαγραφή Κεραίας.....	72
8.14	Βλάβες.....	73
8.14.1	Αρχική Οθόνη Βλαβών.....	73
8.14.2	Δημιουργία Βλάβης.....	73
8.14.3	Διαγραφή Βλάβης.....	75
8.15	Επιδιορθώσεις.....	76
8.15.1	Αρχική Οθόνη Επιδιορθώσεων.....	76
8.15.2	Δημιουργία Επιδιόρθωσης.....	76
8.15.3	Διαγραφή Επιδιόρθωσης.....	78
8.16	Ετήσια Ανανέωση Παγίων - Επιδόσεις.....	79
8.17	Στατιστικά.....	82
8.17.1	Ετήσια Συγκεντρωτική Εσόδων ανά Τμήμα ανά μήνα.....	82
8.17.2	Ετήσια Συγκεντρωτική Συμβολαίων ανά Υπάλληλο ανά μήνα.....	83
8.17.3	Ετήσια Συγκεντρωτική Συμβολαίων ανά Τμήμα ανά μήνα.....	84
8.17.4	Ετήσια Συγκεντρωτική Καρτοτηλεφώνων ανά Τμήμα ανά μήνα.....	85
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		87

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1:Εγκατάσταση xampp	21
Εικόνα 2: Ρυθμίσεις xampp.....	22
Εικόνα 3:Ολοκλήρωση Ρυθμίσεων xampp	22
Εικόνα 4:Εκκίνηση xampp Control Panel	23
Εικόνα 5:Εκκίνηση Apache και Mysql	23
Εικόνα 6:Επιλογή admin λειτουργίας για τον Apache	24
Εικόνα 7: Ολοκλήρωση εγκατάστασης xampp	24
Εικόνα 8: Διάγραμμα Οντοτήτων Σχέσεων	25
Εικόνα 9:Έλεγχος κωδικού-ονόματος χρήστη.....	37
Εικόνα 10: Εισαγωγή στο σύστημα - Μήνυμα καλωσορίσματος	37
Εικόνα 11: Αρχική Οθόνη Χρηστών.....	38
Εικόνα 12: Φόρμα Προσθήκης Νέου Χρήστη	39
Εικόνα 13: Ολοκλήρωση προσθήκης Νέου Χρήστη.....	39
Εικόνα 14: Διαγραφή Χρήστη 1.....	40
Εικόνα 15: Διαγραφή Χρήστη 2.....	40
Εικόνα 16: Αρχική Οθόνη Τμημάτων	41
Εικόνα 17: Φόρμα Προσθήκης Τμήματος.....	42
Εικόνα 18:Ολοκλήρωση προσθήκης Νέου Τμήματος.....	42
Εικόνα 19: Διαγραφή Τμήματος 1.....	43
Εικόνα 20:Διαγραφή Τμήματος 2.....	43
Εικόνα 21: Αρχική Οθόνη Υπαλλήλων.....	44
Εικόνα 22: Φόρμα Προσθήκης Νέου Υπαλλήλου	45
Εικόνα 23: Ολοκλήρωση προσθήκης Νέου Υπαλλήλου	45
Εικόνα 24: Διαγραφή Υπαλλήλου 1	46
Εικόνα 25: Διαγραφή Υπαλλήλου 2	47
Εικόνα 26: Αρχική Οθόνη Τύπων Υπαλλήλων.....	48
Εικόνα 27: Φόρμα Προσθήκης Τύπου Υπαλλήλου	48
Εικόνα 28:Ολοκλήρωση προσθήκης Νέου Τύπου Υπαλλήλου	49
Εικόνα 29: Διαγραφή Τύπου Υπαλλήλου 1	50
Εικόνα 30:Διαγραφή Τύπου Υπαλλήλου 2.....	50
Εικόνα 31: Αρχική Οθόνη Διευθυντών.....	51
Εικόνα 32: Φόρμα Προσθήκης Νέου Διευθυντή	52
Εικόνα 33: Ολοκλήρωση προσθήκης Νέου Διευθυντή.....	52
Εικόνα 34: Διαγραφή Διευθυντή 1.....	53
Εικόνα 35: Διαγραφή Διευθυντή 2.....	54
Εικόνα 36: Αρχική Οθόνη Τηλ. Προγραμμάτων	55
Εικόνα 37: Φόρμα Προσθήκης Νέου Τηλ.Προγράμματος.....	55
Εικόνα 38: Ολοκλήρωση προσθήκης Νέου Τηλ.Προγράμματος	56
Εικόνα 39: Διαγραφή Τηλ. Προγράμματος 1	57
Εικόνα 40: Διαγραφή Τηλ. Προγράμματος 2	57
Εικόνα 41: Αρχική Οθόνη Οικογενειακών Μελών	58
Εικόνα 42: Φόρμα Προσθήκης Νέου Οικογενειακού Μέλους	59
Εικόνα 43: Ολοκλήρωση προσθήκης Νέου Οικογενειακού Μέλους.....	59

Εικόνα 44: Διαγραφή Οικογενειακού Μέλους 1.....	60
Εικόνα 45: Διαγραφή Οικογενειακού Μέλους 2.....	60
Εικόνα 46: Αρχική Οθόνη Πελατών.....	61
Εικόνα 47: Φόρμα Προσθήκης Νέου Πελάτη	62
Εικόνα 48: Ολοκλήρωση προσθήκης Νέου Πελάτη.....	62
Εικόνα 49: Διαγραφή Πελάτη 1.....	63
Εικόνα 50: Διαγραφή Πελάτη 2.....	63
Εικόνα 51: Αρχική Οθόνη Τηλεφωνικών Αριθμών.....	64
Εικόνα 52: Φόρμα Προσθήκης Νέου Τηλ. Αριθμού.....	65
Εικόνα 53: Ολοκλήρωση προσθήκης Νέου Τηλ.Αριθμού	65
Εικόνα 54: Διαγραφή Τηλ. Αριθμού1.....	66
Εικόνα 55: Διαγραφή Τηλ. Αριθμού1.....	66
Εικόνα 56: Αρχική Οθόνη Λογαριασμών.....	67
Εικόνα 57: Φόρμα Προσθήκης Νέου Λογαριασμού	68
Εικόνα 58: Ολοκλήρωση προσθήκης Νέου Λογαριασμού	68
Εικόνα 59: Διαγραφή Λογαριασμού 1	69
Εικόνα 60: Διαγραφή Λογαριασμού 2	69
Εικόνα 61: Αρχική Οθόνη Κεραιών	70
Εικόνα 62: Φόρμα Προσθήκης Νέας Κεραίας.....	71
Εικόνα 63: Ολοκλήρωση προσθήκης Νέας Κεραίας.....	71
Εικόνα 64: Διαγραφή Κεραίας 1.....	72
Εικόνα 65: Διαγραφή Κεραίας 1.....	72
Εικόνα 66: Αρχική οθόνη Βλαβών.....	73
Εικόνα 67: Φόρμα Προσθήκης Νέας Βλάβης.....	74
Εικόνα 68: Ολοκλήρωση προσθήκης Νέας Βλάβης	74
Εικόνα 69: Διαγραφή Βλάβης 1.....	75
Εικόνα 70: Διαγραφή Βλάβης 2.....	75
Εικόνα 71: Αρχική οθόνη Επιδιορθώσεων.....	76
Εικόνα 72: Φόρμα Προσθήκης Νέας Επιδιόρθωσης	77
Εικόνα 73: Ολοκλήρωση προσθήκης Νέας Επιδιόρθωσης.....	77
Εικόνα 74: Διαγραφή Επιδιόρθωσης 1	78
Εικόνα 75: Διαγραφή Επιδιόρθωσης 2	78
Εικόνα 76: Πίνακας Τηλ.Αριθμών ΠΡΙΝ την ετήσια ανανέωση	79
Εικόνα 77: Εισαγωγή μελλοντικής ημερομηνίας για ετήσια ανανέωση Παγίων	80
Εικόνα 78: Πίνακας Τηλ.Αριθμών ΜΕΤΑ την ετήσια ανανέωση	80
Εικόνα 79: Ετήσια Συγκεντρωτική Εσόδων ανά μήνα.....	82
Εικόνα 80: Ετήσια Συγκεντρωτική Συμβολαίων ανά Υπάλληλο ανά μήνα.....	83
Εικόνα 81: Ετήσια Συγκεντρωτική Συμβολαίων ανά Τμήμα ανά μήνα	84
Εικόνα 82: Ετήσια Συγκεντρωτική Καρτοτηλεφώνων ανά Τμήμα ανά μήνα	86

ΠΙΝΑΚΑΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Διαθέσιμες λειτουργίες συστήματος ανά ρόλο	34
Πίνακας 2: Περιεχόμενα φακέλων-αρχείων	36
Πίνακας 3: Ενέργειες κώδικα κάθε αρχείου	36

ΠΕΡΙΛΗΨΗ

Στην παρούσα εργασία πραγματοποιήσαμε την ανάπτυξη ενός συστήματος διαχείρισης για μια εταιρεία διαχείρισης κινητών τηλεφώνων. Αρχικώς, κάναμε την εγκατάσταση των αναγκαίων προγραμμάτων που μας βοηθάνε να υποστηρίξουμε ένα σύστημα διαχείρισης βάσεων δεδομένων βασισμένο στο διαδίκτυο. Ακολούθως, η καταγραφή, η μοντελοποίηση και η αποτύπωση των αναγκαίων χαρακτηριστικών του συστήματος στις διάφορες φόρμες λειτουργίας ήταν ένα από τα μέρη με το οποίο ασχολείται η παρούσα εργασία αρκετά. Επίσης, προχωρήσαμε στην υλοποίηση διαβαθμισμένης χρήσης (διαχειριστή-χρηστών) στο σύστημα όπου ο κάθε χρήστης ανάλογα το ρόλο του έχει συγκεκριμένες λειτουργίες στη διάθεση του. Τέλος, ακολουθήσαμε διαφορετικά σενάρια χρήσης και λειτουργίας με σκοπό να διασφαλίσουμε τη εύρυθμη λειτουργία του συστήματος μας.

ABSTRACT

In this thesis, we implemented a database management system for a mobile company. Firstly, we proceed to the installation of the programs which assisted us in creating an online database system. Furthermore, we made the requirement analysis of our system. Moreover, we enabled the role-based use of administrators and simple users of the system, giving to each role different functionalities to handle. Finally, we run several live case studies to ensure the full functionality of the system.

1 ΣΚΟΠΟΣ ΤΗΣ ΕΡΓΑΣΙΑΣ

Ο σκοπός της παρούσας εργασίας είναι ανάπτυξη ενός συστήματος διαχείρισης για μια εταιρεία διαχείρισης κινητών τηλεφώνων. Αρχικώς, κάναμε την εγκατάσταση των αναγκαίων προγραμμάτων που μας βοηθάνε να υποστηρίξουμε ένα σύστημα διαχείρισης βάσεων δεδομένων βασισμένο στο διαδίκτυο. Ακολούθως, η καταγραφή, η μοντελοποίηση και η αποτύπωση των αναγκαίων χαρακτηριστικών του συστήματος στις διάφορες φόρμες λειτουργίας ήταν ένα από τα μέρη με το οποίο ασχολείται η παρούσα εργασία αρκετά. Επίσης, προχωρήσαμε στην υλοποίηση διαβαθμισμένης χρήσης (διαχειριστή-χρηστών) στο σύστημα όπου ο κάθε χρήστης ανάλογα το ρόλο του έχει συγκεκριμένες λειτουργίες στη διάθεση του. Τέλος, ακολουθήσαμε διαφορετικά σενάρια χρήσης και λειτουργίας με σκοπό να διασφαλίσουμε τη εύρυθμη λειτουργία του συστήματος μας.

2 ΕΙΣΑΓΩΓΗ

2.1 Η αλληλεπίδραση του χρήστη με το Διαδίκτυο

Ο παγκόσμιος ιστός (world wide web) είναι ίσως ο τομέας του διαδικτύου που παρουσιάζει τις εντυπωσιακότερες αλλαγές. Πράγματι, ξεκινώντας από απλές σελίδες υπερ-κειμένου, είναι πλέον σε θέση να υποστηρίζει πολύπλοκες πολυμεσικές, πολυχρηστικές εφαρμογές όπως το facebook και το youtube. Συγκρίνοντας παρόν με παρελθόν, παρατηρούμε ότι οι υπολογιστικές αρχιτεκτονικές έχουν αλλάξει εντυπωσιακά, τα μοντέλα αλληλεπίδρασης με το χρήστη έχουν επαναπροσδιοριστεί, εντούτοις η βάση του παγκόσμιου ιστού παραμένει σταθερή: Για να υποστηρίξει την εκρηκτική εξάπλωση της τεχνολογίας, η βάση αυτή επεκτάθηκε προς πάσα κατεύθυνση με ετερόκλητα, άναρχα στοιχεία και ανταγωνιστικές τεχνολογίες.

Με κριτήριο το μοντέλο αλληλεπίδρασης με το χρήστη και τις υποστηριζόμενες τεχνολογίες, μπορούμε να εντοπίσουμε τρία διακριτά στάδια στην ανάπτυξη του Παγκόσμιου Ιστού (1):

1. Αρχικά ο παγκόσμιος ιστός αποτελείται από στατικές σελίδες, οι οποίες κατά κύριο λόγο περιέχουν υπερκείμενο και διανθίζονται σταδιακά από απλά γραφικά, κινούμενες εικόνες και ήχους. Με την έλευση του Macromedia flash, οι σελίδες αποκτούν και κινούμενα γραφικά.

2. Ακολούθως ένα μεγάλο μέρος του παγκόσμιου ιστού γίνεται δυναμικό, δηλαδή το περιεχόμενο που προβάλλεται στο χρήστη παράγεται από κάποιο πρόγραμμα στον εξυπηρετητή ανάλογα με τις δράσεις του χρήστη. Γλώσσες όπως η php επιτρέπουν στους προγραμματιστές να αναπτύσσουν ολόκληρα προγράμματα που παράγουν δυναμικό περιεχόμενο, χωρίς να γνωρίζουν την υπολογιστική αρχιτεκτονική του εξυπηρετητή. Νέες εφαρμογές όπως τα blogs και τα forums κάνουν την εμφάνισή τους.

3. Η εξέλιξη αυτή φέρνει τα web applications και web services. Η αλληλεπίδραση της ιστοσελίδας με το χρήστη αλλάζει σημαντικά ώστε να προσομοιώνει παραθυρική εφαρμογή. Αυτό γίνεται εφικτό με τη χρήση κώδικα που τρέχει στον υπολογιστή του χρήστη, που επιτρέπει τη δυναμική ανταλλαγή πληροφοριών από τον εξυπηρετητή. Ιστορικά λοιπόν, ο Παγκόσμιος Ιστός ξεκίνησε ως σύνολο στατικών σελίδων υπερκειμένου. Επεκτάθηκε με την εισαγωγή ενός επιπέδου

προγραμματισμού στον εξυπηρετητή ώστε οι σελίδες να παράγονται δυναμικά, ενώ πρόσφατα εισήχθη (ή ακριβέστερα, συστηματοποιήθηκε) ένα ακόμα επίπεδο προγραμματισμού στον υπολογιστή-πελάτη, ώστε πλέον να μιλάμε για διαδικτυακές εφαρμογές αντί για ιστοσελίδες. Κάθε μια από τις παραπάνω φάσεις συνιστά και μια επανάσταση στο τρόπο που οι ιστοσελίδες αλληλεπιδρούν με τους χρήστες τους. Επίσης,, κάθε μια από τις παραπάνω φάσεις ανάπτυξης πραγματώθηκε με την ωρίμανση και συστηματοποίηση ενός συνόλου τεχνολογιών και γλωσσών προγραμματισμού. Έτσι από την HTML περάσαμε στις server-side γλώσσες και την XHTML ενώ πρόσφατα με το AJAX συστηματοποιείται και η χρήση javascript(2)(3) ενώ προτείνονται ειδικές γλώσσες (πχ OWL) για την ανταλλαγή δεδομένων μεταξύ ιστοτόπων. Αφού η εξέλιξη του διαδικτύου είναι συνδεδεμένη με τις γλώσσες προγραμματισμού, σκόπιμο κρίνεται να εξετάσουμε τις σχετικές τεχνολογίες.

2.2 Οι γλώσσες προγραμματισμού σήμερα

Οι γλώσσες προγραμματισμού θα μπορούσαν να χωριστούν σε τρεις κύριες κατηγορίες:

1. Γλώσσες με μεταφραστή (compilers). Κύριο χαρακτηριστικό τους είναι η off-line μετάφραση του πηγαίου κώδικα σε κώδικα άμεσα εκτελέσιμο από μια υπολογιστική μηχανή. Παράγεται δηλαδή κατά τη μετάφραση ένα εκτελέσιμο αρχείο από τη μηχανή, που αποτελεί και το πρόγραμμα. Αντιπρόσωποι αυτών των γλωσσών είναι η C, η Pascal αλλά και η Java (4).

2. Γλώσσες σήμανσης (markup languages), όπως η XML και η HTML, οι οποίες περιγράφουν σε μια καλώς ορισμένη δομή χαρακτηριστικά ή ιδιότητες ενός αντικειμένου.

3. Γλώσσες με διερμηνευτή (interpreter) που έχουν επικρατήσει να καλούνται και scripting-γλώσσες. Σε αυτές τις γλώσσες ο πηγαίος κώδικας διερμηνεύεται σε πραγματικό χρόνο από τον διερμηνευτή, ή μεταφράζεται σε αντικειμενικό κώδικα διαφανώς προς το χρήστη, δηλαδή δεν παράγεται κάποιο εκτελέσιμο αρχείο από τη διαδικασία. Ας σημειωθεί εδώ ότι σε πολλές περιπτώσεις παρέχεται η δυνατότητα αποθήκευσης του αντικειμενικού κώδικα σε ένα εκτελέσιμο αρχείο (για παράδειγμα στην Lightwave Scripting Language) (5), γεγονός που κάνει δυσδιάκριτα τα όρια μεταφραστή και διερμηνευτή. Στο εξής θα αναφερόμαστε στις γλώσσες με διερμηνευτή ως scripting languages και στα προγράμματα μιας τέτοιας γλώσσας ως scripts.

Ο κόσμος των διερμηνευτών θα μπορούσε να χαρακτηριστεί στάσιμος μέχρι την έλευση του Διαδικτύου και των σχετικών εφαρμογών(1). Πράγματι, παραλλαγές γλωσσών με μεταγλωττιστή όπως οι C και Pascal χρησιμοποιούνταν κατά κόρον για την ανάπτυξη εφαρμογών σε εταιρικά και παραθυρικά περιβάλλοντα. Διερμηνευτές (interpreters) χρησιμοποιούνταν κυρίως για βοηθητικές εργασίες (shell scripts, automation scripts). Η βελτιωμένη απόδοση του μεταγλωττισμένου κώδικα αλλά και η σχετική αντοχή του σε τεχνικές reverse engineering είναι πιθανότατα οι κύριοι λόγοι για τη κυριαρχία των μεταγλωττιστών.

Η επέκταση του διαδικτύου ωστόσο έφερε αλλαγές στα μοντέλα ανάπτυξης. Ένα σημαντικό μερίδιο εφαρμογών πλέον εκτελείται σε ετερογενή περιβάλλοντα αγνώστου αρχιτεκτονικής, ενώ η πλειοψηφία των (δυναμικών) δικτυακών ιστοσελίδων φιλοξενούνται σε κάποιον εξυπηρετητή μιας εταιρίας παροχής υπηρεσιών διαδικτύου. Συνεπώς τόσο οι δυναμικές ιστοσελίδες όσο και οι διαδικτυακές εφαρμογές, είναι απαραίτητο να είναι ανεξάρτητες αρχιτεκτονικής και ευέλικτες στην

ανάπτυξη. Νέες γλώσσες προγραμματισμού, βασισμένες στο μοντέλο του διερμηνευτή σχεδιάζονται με στόχο να υποστηρίξουν την ανάπτυξη τέτοιων εφαρμογών. Δεδομένου ότι η υπολογιστική ισχύς επιτρέπει πλέον τη δημιουργία διερμηνευτών υψηλής πολυπλοκότητας, γλώσσες όπως η Php και η Asp γίνονται δημοφιλείς.

Στους περισσότερους διερμηνευτές σήμερα, ο πηγαίος κώδικας μετατρέπεται (μεταγλωττίζεται) αυτόματα σε ενδιάμεσο κώδικα (byte-code), ο οποίος εκτελείται από μια ιδεατή μηχανή. Η ιδεατή μηχανή μεταφέρεται σε κάθε υποστηριζόμενη αρχιτεκτονική, επιτρέποντας στον ενδιάμεσο κώδικα να είναι ανεξάρτητος αρχιτεκτονικής. Γλώσσες όπως η Php, η Java αλλά και η Visual Basic παλαιότερα, ακολουθούν αυτό το παράδειγμα(4). Λόγω της ανάγκης για ανεξαρτησία από την υπολογιστική αρχιτεκτονική αλλά και την αυξημένη απόδοση, προτιμάται η χρήση byte-code από τη δημιουργία εγγενούς κώδικα.

Ας σημειωθεί εδώ ότι εκτός από το διαδίκτυο οι διερμηνευτές βρίσκουν εφαρμογή και σε άλλες περιπτώσεις: Οι περισσότερες σύγχρονες εφαρμογές διαθέτουν ένα σύστημα αυτοματοποίησης (scripting system) το οποίο επιτρέπει σε προχωρημένους χρήστες να επεκτείνουν τις λειτουργίες τους γράφοντας μικρά προγράμματα (scripts). Η φιλοσοφία αυτή βρίσκει εφαρμογή και στη ψηφιακή διασκέδαση, αφού τα περισσότερα βιντεοπαιχνίδια χρησιμοποιούν διερμηνευτές (πχ unreal script και Lua) για να περιγράψουν δυναμικά τόσο τα σενάρια παιχνιδιού όσο και στοιχεία λογικής (6)(7).

2.3 Οι κύριες τεχνολογίες ανάπτυξης ιστοσελίδων

Η ανάπτυξη του παγκόσμιου ιστού έφερε και ένα σύνολο νέων τεχνολογιών ανάπτυξης εφαρμογών, το οποίο χαρακτηρίζεται από έντονη ετερογένεια. Για παράδειγμα, μια μέση διαδικτυακή εφαρμογή απαιτεί τη γνώση τουλάχιστον τριών γλωσσών για τη συγγραφή της (8)(9)(10):

- Php, Asp, Jsp ή παρόμοια server-side γλώσσα για τη δημιουργία δυναμικού περιεχομένου από τη μεριά του εξυπηρετητή,
- HTML (XML, CSS) για τη περιγραφή του περιεχομένου που θα δείξει ο φυλλομετρητής (web browser)
- Javascript για τη δυναμική παρουσίαση και διαχείριση του περιεχομένου από τη μεριά του πελάτη. Ας σημειωθεί εδώ ότι κάθε browser φέρει τη δική του ιδιαίτερη υλοποίηση της γλώσσας, με αποτέλεσμα να απαιτείται σημαντική αφαίρεση (abstraction) κατά τον προγραμματισμό, ώστε να υποστηριχθούν όλοι οι γνωστοί web browsers.

Σε πραγματικές διαδικτυακές εφαρμογές χρησιμοποιούνται περισσότερες τεχνολογίες και γλώσσες, όπως για παράδειγμα οι Flash/actionscript για δυναμικά γραφικά και video και η SQL για διασύνδεση με βάσεις δεδομένων. Εκτενής αναφορά στις ιδιαιτερότητες κάθε γλώσσας γίνεται στα επόμενα κεφάλαια.

2.4 Το υπάρχον μοντέλο ανάπτυξης

Παρατηρούμε ότι η συντριπτική πλειοψηφία των παραπάνω τεχνολογιών χρησιμοποιείται για την υλοποίηση client-server εφαρμογών και τη μεταφορά δεδομένων μέσω δικτύου. Ειδικά στη περίπτωση των διαδικτυακών εφαρμογών, παρατηρείται μια πολύπλοκη και ετερογενής συμμετοχή διαφορετικών τεχνολογιών κατά τη συγγραφή της εφαρμογής (8)(1).

Πράγματι, στη συνήθη περίπτωση, ο προγραμματιστής καλείται να χρησιμοποιήσει μια server-side scripting language (php, jsr, asp), η οποία με χρήση αλφαριθμητικών θα παράγει σε μορφή κειμένου τον πηγαίο κώδικα μιας άλλης γλώσσας (Html, Javascript). Ουσιαστικά, ο πηγαίος κώδικας της εφαρμογής στον υπολογιστή-πελάτη βρίσκεται εμφωλευμένος μέσα στις διάφορες ρουτίνες του κώδικα της εφαρμογής στον server. Το μοντέλο αυτό, παρόλο που χαρακτηρίζεται από ευελιξία, όπως θα δειχθεί παρακάτω, πάσχει από σοβαρά προβλήματα αναγνωσιμότητας, απόδοσης, συντήρησης του κώδικα αλλά και επεκτασιμότητας.

Ας σημειωθεί εδώ ότι οι περισσότερες server-side γλώσσες δίνουν την δυνατότητα να ενσωματωθούν μέσα σε αρχεία .html ή xml, τα οποία «φιλτράρονται» αυτόματα από το διερμηνευτή πριν προβληθούν στο χρήστη (8). Παρ' ότι βολική, αυτή η δυνατότητα επιτρέπει στους προγραμματιστές να "κρύβουν" κομμάτια κώδικα σε περισσότερα αρχεία, δυσχεραίνοντας περισσότερο τη διαχείρισή του.

Το κύριο ζήτημα που ανακύπτει στο υπάρχον μοντέλο ανάπτυξης διαδικτυακών εφαρμογών είναι ότι χρησιμοποιείται ένα ετερογενές σύνολο από γενικές, δίκτυο-αγνωστικές γλώσσες για να υλοποιηθούν καθαρά δικτυακές εφαρμογές (client-server). Μάλιστα, για να διατηρηθεί η προς τα πίσω συμβατότητα με το πρωτόκολλο http, οι περισσότερες server-side γλώσσες επικεντρώνονται στη διαχείριση αλφαριθμητικών ώστε να παράγουν τον κώδικα (html, javascript κλπ) που θα παρουσιαστεί στο πελάτη.

Ένα παράδειγμα των προβλημάτων που αυτή η αντιμετώπιση φέρει, είναι η υλοποίηση των συνόδων (sessions) στις server-side γλώσσες (11)(8). Ως γνωστόν, το πρωτόκολλο http είναι stateless, δηλαδή δεν ενδιαφέρεται για το ιστορικό των συνδέσεων ενός πελάτη με τον εξυπηρετητή. Εντούτοις οι περισσότερες εφαρμογές απαιτούν ο χρήστης να περιηγηθεί από πολλές html σελίδες και ο εξυπηρετητής να διατηρεί στοιχεία τόσο για το χρήστη όσο και για τις σελίδες που εκείνος επισκέφτηκε. Για να αντιμετωπιστεί το πρόβλημα και να διατηρηθεί η προς τα πίσω συμβατότητα, αναπτύχθηκε η έννοια της «συνόδου» (browsing session) ως το σύνολο των σελίδων που διαβάσει ένας χρήστης κατά την επίσκεψή του στον ιστοτόπο. Οι browsers εισήγαγαν τα cookies ως μονάδες τοπικής αποθήκευσης δεδομένων που αντιστοιχούν σε μια σύνοδο.

Στην έκδοση 2, η γλώσσα PHP δεν διέθετε κάποιο μηχανισμό για υποστήριξη συνόδου πέρα από κάποιες ρουτίνες για διαχείριση cookies (11),(1). Αυτό έκανε δυσχερή και άκομψη τη συγγραφή διαδικτυακών εφαρμογών αφού επαφιόταν στο προγραμματιστή να διαχειριστεί τα cookies μιας συνόδου. Με την υλοποίηση ρουτινών για την υποστήριξη συνόδων, η ανάπτυξη τέτοιων εφαρμογών έγινε ευκολότερη, καθώς ένα επίπεδο αφαίρεσης αναλάμβανε την αυτόματη διαχείριση (θέση και ανάγνωση) όλων των σχετικών cookies ώστε να υπάρχει συνοχή από σελίδα σε σελίδα σε μια σύνοδο. Παρ' όλα αυτά, το γεγονός ότι κάθε σελίδα θεωρείται αυτόνομη και όχι μέρος μιας συνόδου, τόσο από το http όσο και από τη php, αναγκάζει το προγραμματιστή να συμπεριλάβει ειδικό κώδικα στις σελίδες που αναπτύσσει. Άλλες γλώσσες επιτρέπουν γρηγορότερη

ανάπτυξη διαδικτυακών εφαρμογών ακριβώς λόγω του καλύτερου συστήματος αφαίρεσης που διαθέτουν.

Αν λάβουμε υπ' όψη την εκτενή χρήση javascript και τα ασύγχρονα μοντέλα που χρησιμοποιούν οι τρέχουσες διαδικτυακές εφαρμογές γίνεται προφανής η ανάγκη για ένα απλούστερο, περισσότερο φιλικό στο προγραμματιστή περιβάλλον ανάπτυξης εφαρμογών.

Συνοψίζοντας, τα κύρια μειονεκτήματα που εντοπίζονται στο υπάρχον μοντέλο ανάπτυξης είναι τα ακόλουθα

- Έντονα ετερογενές περιβάλλον: Εμφώλευση κώδικα μιας γλώσσας στο πηγαίο κώδικα μιας άλλης. Αυτή η τεχνική δυσχεραίνει την αναγνωσιμότητα και διαχείριση του κώδικα αλλά και τον εντοπισμό λαθών.

- Δυσκολία προτυποποίησης και συστηματοποίησης της συγγραφής κώδικα: Χάριν της προς τα πίσω συμβατότητας και λόγω ασυμβατοτήτων μεταξύ τεχνολογιών, ο προγραμματιστής αφιερώνει το σημαντικότερο ποσοστό του χρόνου του για να διαχειριστεί ασυμβατότητες μεταξύ λογισμικού ή να υποστηρίξει παλαιές τεχνολογίες (πχ ανάγκη χρήσης διαφορετικού javascript κώδικα για κάθε τύπο browser)

- Μη-ειδικευμένα εργαλεία: Η χρήση γλωσσών γενικής χρήσεως που δεν έχουν σχεδιαστεί για το client-server μοντέλο, δυσχεραίνει την ανάπτυξη σχετικών εφαρμογών.

- Κατακερματισμός τεχνολογιών: Το μεγαλύτερο μέρος της προσπάθειας του προγραμματιστή δεν είναι η ανάπτυξη αυτής καθαυτής της εφαρμογής, αλλά η υποστήριξη όλων των ετερογενών προτύπων και τεχνολογιών στο κώδικά του. Αναγκάζεται να εντοπίζει και να επιλύει προβλήματα που δημιουργεί όχι ο κώδικας του, αλλά οι ασυμβατότητες των ετερογενών τεχνολογιών και η ελλιπής υποστήριξη τους από τους φυλλομετρητές (πχ διαφορετική σημειογραφία μεταξύ εκδόσεων της javascript ή διαφορετική συμπεριφορά συγκεκριμένων stylesheet tags).

3 ΥΠΑΡΧΟΥΣΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

Σε αυτή την ενότητα παρουσιάζονται οι κυριότερες τεχνολογίες και μέθοδοι για την ανάπτυξη διαδικτυακών εφαρμογών. Αναλύεται η φιλοσοφία κάθε

3.1 Server side interpreters: Php

Η Php είναι μια από τις πιο διαδεδομένες γλώσσες προγραμματισμού δυναμικών ιστοσελίδων. Πρόκειται για διερμηνευτές υψηλής απόδοσης που συνήθως διασυνδέονται με το λογισμικό web server ως αρθρώματα, αλλά μπορούν να εκτελεστούν και ανεξάρτητα.

3.1.1 Σχεδιαστική Φιλοσοφία

Η Php ξεκίνησε ως ένα σετ εργαλείων CGI (common gateway interface) με στόχο τη συγγραφή προσωπικών ιστοσελίδων (αρχικά το όνομα php σήμαινε Personal Home Page Tool). (1)(11)Την πρώτη της έκδοση ανέπτυξε ως βοηθητικό εργαλείο ο Rasmus Ledorf. Αργότερα την διένειμε ως ανοιχτό κώδικα, ενώ παράλληλα άλλαξε το όνομα σε “Php Hypertext Preprocessor”.

Ο γλωσσικός αναλυτής (parser) υπέστη σημαντικές αλλαγές από τους Zeev Suraski και Andi Gutmans, οι οποίοι, στην έκδοση 3, άλλαξαν την αρχιτεκτονική ώστε ο κώδικας να μετατρέπεται on-the-fly σε byte-code και να εκτελείται από μια ιδεατή μηχανή (virtual machine), τη Zend Engine. Αργότερα οι Suraski και Gutmans ίδρυσαν την Zend Technologies, η οποία εκμεταλλεύεται και εμπορικά τη Zend engine (www.zend.com). Παράλληλα, η Php αναπτύσσεται στα πλαίσια λογισμικού ανοιχτού κώδικα από το Php group.

Η Php μπορεί να συνδεθεί ως άρθρωμα με τους περισσότερους web server, να διασυνδεθεί μέσω Cgi (Common Gateway Interface) ή και να κληθεί από τη γραμμή εντολών για να εκτελέσει αυτόνομα και τοπικά ένα Php πρόγραμμα. Για τη χρήση σε περιβάλλον web server, η διασύνδεση μέσω cgi θεωρείται ασφαλέστερη.

Όπως στις περισσότερες server side γλώσσες, η php μπορεί να εντεθεί σε κώδικα html. Ο λεκτικός αναλυτής αναγνωρίζει ως πηγαίο κώδικα μόνο το κείμενο που βρίσκεται εντός των αναγνωριστικών `<?php` και `?>` (ή συντομογραφικά `<?` και `?>`). Εναλλακτικά μπορούν να χρησιμοποιηθούν και τα html tags `<script language="php">` και `</script>`. Το κείμενο εκτός των php tags δεν αναγνωρίζεται από τον αναλυτή και αντιγράφεται αυτούσιο στην έξοδο (standard output). Αυτό επιτρέπει την εμφώλευση κώδικα php σε αρχεία html ή οποιαδήποτε αρχείο κειμένου.

3.2 Java & Java Server Pages

Η Java είναι μια πλήρως αντικειμενοστραφής γλώσσα προγραμματισμού με μεταφραστή, που αναπτύχθηκε από τη Sun Microsystems Inc και χρησιμοποιείται ευρέως χάρις στα cross-platform χαρακτηριστικά που διαθέτει. Η τεχνολογία Java Server Pages επιτρέπει τη χρήση της Java για τη συγγραφή δυναμικών σελίδων αλλά και ολοκληρωμένων εφαρμογών εξυπηρετητή, ομογενοποιημένων με τον παγκόσμιο ιστό (4)(12).

3.2.1 Σχεδιαστική Φιλοσοφία

Ο κώδικας java μεταφράζεται όχι για κάποια συγκεκριμένη αρχιτεκτονική υλικού, αλλά για την αρχιτεκτονική μιας ιδεατής μηχανής (Java Virtual Machine), η οποία έχει ένα σαφώς ορισμένο σύνολο χαρακτηριστικών(4). Η ιδεατή αυτή μηχανή υλοποιείται σε κάθε υποστηριζόμενη πλατφόρμα υλικού, ακολουθώντας αυστηρές συμβάσεις ως προς τους τύπους δεδομένων και τις δυνατότητες που υποστηρίζονται. Αυτό επιτρέπει στα προγράμματα που γράφονται σε java, να εκτελούνται όμοια και να βγάζουν ίδια αποτελέσματα ανεξάρτητα από την υφιστάμενη αρχιτεκτονική υλικού.

Όπως αναφέρθηκε παραπάνω, υπάρχουν και άλλες γλώσσες που αξιοποιούν τη φιλοσοφία της ιδεατής μηχανής. Η java είναι η πρώτη γλώσσα ισχυρών τύπων με καθαρά αντικειμενοστραφή

φιλοσοφία, στην οποία οι βασικοί τύποι (ακέραιοι, δεκαδικοί, πίνακες κλπ) αποθηκεύονται με σαφώς ορισμένο τρόπο ανεξαρτήτως της αρχιτεκτονικής υλικού. Για παράδειγμα, ενώ κάποιες αρχιτεκτονικές υλικού θεωρούν τον τύπο char ως 8-bit signed integer, κάποιες άλλες υλοποιούν τον ίδιο τύπο ως 8-bit unsigned integer. Η java Virtual Machine υλοποιεί τον τύπο char με ενιαίο και ομογενή τρόπο σε όλες τις πλατφόρμες. Έτσι, ο μεταγλωττισμένος κώδικας java μπορεί να εκτελεστεί σε οποιαδήποτε πλατφόρμα υποστηρίζεται από τη java virtual machine.

Δυστυχώς η ενιαία αυτή αντιμετώπιση φέρει προβλήματα αποδοτικότητας, καθώς σε περίπτωση που κάποια αρχιτεκτονική δεν υποστηρίζει εγγενώς ένα τύπο δεδομένων, όλες οι προσβάσεις στον τύπο αυτό γίνονται μέσω λογισμικού. Η σημαντικά μειωμένη ταχύτητα του κώδικα java αποτέλεσε και τη κυριότερη τροχοπέδη στη διάδοσή της. Παρ' όλα αυτά, ένα σετ τεχνολογιών όπως οι Just In Time Compilers επιτρέπουν βελτιστοποίηση της απόδοσης ενός προγράμματος java.

Σε αντίθεση με τη φιλοσοφία της Java, άλλες γλώσσες υλοποιούν Virtual Machines που χρησιμοποιούν τους εκάστοτε εγγενείς τύπους της αρχιτεκτονικής. Δηλαδή ένας τύπος char μπορεί να σημαίνει είτε ένα 8 bit unsigned integer, είτε ένα 8 bit signed integer στο μεταγλωττισμένο κώδικα, ανάλογα με την υφιστάμενη αρχιτεκτονική. Η φιλοσοφία αυτή

συνεπάγεται ότι ο ίδιος πηγαίος κώδικας μπορεί να μεταφραστεί σε διαφορετικές πλατφόρμες, όμως δεν υπάρχουν εγγυήσεις ότι θα εκτελεστεί με τον ίδιο τρόπο.(8) Πάντως ένα μεγάλο μέρος του κώδικα θα έχει κοινή συμπεριφορά σε ανεξάρτητες πλατφόρμες. Προφανώς, επειδή γίνεται χρήση εγγενών τύπων, η απόδοση της ιδεατής μηχανής είναι βελτιωμένη.

3.2.2 Java Server Pages

Λόγω της ανεξαρτησίας της από το υλικό, η Java αποτελεί ιδανική επιλογή για ανάπτυξη επεκτάσιμων εφαρμογών. Εντούτοις σε περιβάλλον παγκόσμιου ιστού η Java αυτή καθ' αυτή πάσχει από δύο προβλήματα:

- Η σύνταξή της δεν είναι ευέλικτη, συνεπώς δε συνίσταται για απλές εργασίες. Προφανώς δεν είναι εφικτή η εμφώλευση κώδικα HTML σε καθαρό κώδικα java στα πρότυπα της rhp
- Για τη εκτέλεση του κώδικα απαιτείται πρώτα η μετάφραση του και αργότερα η εκτέλεση του μέσω της ιδεατής μηχανής. Δεδομένου ότι η ανάπτυξη κώδικα για τον Παγκόσμιο ιστό γίνεται συνήθως σε κάποιο απομακρυσμένο σταθμό εργασίας και ο κώδικας πρέπει να μεταφερθεί και εκτελεστεί στον εξυπηρετητή, μια κομψότερη λύση όπως αυτή της rhp είναι απαραίτητη.

Για την επίλυση των παραπάνω προβλημάτων, η Sun επέκτεινε το σύνολο δυνατοτήτων της Java και ανέπτυξε μια βοηθητική γλώσσα, την JSP (13)(1).

Η JSP επιτρέπει στο προγραμματιστή να εμφωλεύσει κώδικα java σε κώδικα HTML με τη χρήση μια σύνταξης ανάλογης με αυτών της rhp και asp. Επίσης χρησιμοποιεί ένα σύνολο XML tags για να ορίσει συγκεκριμένες λειτουργίες. Τα tags αυτά μπορούν να διασυνδεθούν με τάξεις java οι οποίες εκτελούνται σε χαμηλότερα στρώματα του εξυπηρετητή (backend process, δηλαδή πίσω από τον web server). Παράλληλα, η JSP δίνει ένα σύνολο μηχανισμών για τη διασύνδεση του java τμήματος ενός server με το web τμήμα του, κάποιοι από τους οποίους αναλύονται παρακάτω.

Σε αντίθεση με τη php, η jsp χρησιμοποιεί μια πληθώρα από διαφορετικά jsp tags, ενώ επιτρέπει στο χρήστη να δηλώσει και νέες τάξεις tags αν αυτό είναι αναγκαίο. Πρόσθετα tags (<jsp: tags) υλοποιούν άλλες λειτουργίες web server, όπως η ενσωμάτωση αντικειμένων και η εκτέλεση ειδικού κώδικα. Τέλος, ενδιαφέρον είναι το plugin tag, το οποίο χρησιμοποιείται για να ενσωματώσει αντικείμενα (applets) σε μια σελίδα, υλοποιώντας κατ' ουσία τη λειτουργικότητα του <object> tag της html, αλλά με σύνταξη κατάλληλη για το φυλλομετρητή του τρέχοντος χρήστη (τον οποίο γνωρίζει ο εξυπηρετητής):

Τέλος, μέσω των JSP tag libraries, οι προγραμματιστές μπορούν να ορίσουν εξειδικευμένα tags και να τα ενθέσουν σε HTML κώδικα. Με τον τρόπο αυτό μπορούν εύκολα να διασυνδέσουν το τμήμα αλληλεπίδρασης με το χρήστη της εφαρμογής με το εσωτερικό σύστημα που παρέχει πρόσβαση και επεξεργασία στα εσωτερικά δεδομένα του εξυπηρετητή. Τα tags αυτά μπορούν να υλοποιηθούν σε καθαρή java με κώδικα.

Ο κώδικας java που περιέχεται σε μια jsp σελίδα μεταγλωττίζεται αυτόματα από ένα ειδικό άρθρωμα του εξυπηρετητή (το οποίο στην ουσία επεκτείνει τις δυνατότητες του java compiler) και αυτόματα αποθηκεύεται προς άμεση εκτέλεση. Σε περίπτωση που ο πηγαίος jsp κώδικας υποστεί αλλαγές, θα μεταγλωττιστεί αυτόματα από τον εξυπηρετητή. Με αυτό το τρόπο επιτρέπεται η εύκολη συγγραφή δυναμικών ιστοσελίδων, οι οποίες έχουν τη δυνατότητα να συνεργαστούν με το υπόλοιπο σύνολο λειτουργικότητας που παρέχει ένας εξυπηρετητής.

Η προαναφερθείσα δυνατότητα δίνει ένα σημαντικό πλεονέκτημα στο πρότυπο ανάπτυξης της JSP, καθώς κάνει εφικτό το διαχωρισμό της εφαρμογής σε δύο μεγάλα τμήματα: αυτό της παρουσίασης στο χρήστη (web – front end) και σε αυτό της λογικής επεξεργασίας των δεδομένων (back end). Το backend μπορεί να διαχωριστεί περαιτέρω σε μοντέλο δεδομένων (model) και λογικής (control er). Έτσι οδηγούμαστε στο μοντέλο ανάπτυξης model-view-controler, το οποίο διαχωρίζει την εφαρμογή σε τρία τμήματα:

1. Model: Το μοντέλο δεδομένων της εφαρμογής. Τα μη-εφήμερα δεδομένα αποθηκεύονται σύμφωνα με τις απαιτήσεις και τις δομές που ορίζει το μοντέλο δεδομένων
2. Controler: Το τμήμα διαχείρισης και επεξεργασίας δεδομένων της εφαρμογής. Η επεξεργασία γίνεται με java servlets ή με ένα σύνολο από JSP σελίδες.
3. View: Το τμήμα αλληλεπίδρασης με το χρήστη. Συγκεκριμένες σελίδες (JSP) χρησιμοποιούν τα JSP tags είτε για να τυπώσουν δεδομένα στην οθόνη είτε για να καλέσουν μεθόδους που θα αλλάξουν ή επεξεργαστούν τα δεδομένα αυτά. Με το διαχωρισμό αυτό ο κώδικας γίνεται ευκολότερα διαχειρίσιμος και επεκτάσιμος. Επίσης, ισχυροποιείται η ασφάλεια του κώδικα αφού είναι δυνατόν να ελεγχθεί η πρόσβαση στα εσωτερικά δεδομένα.

3.3 Asp & Asp.net

Η γλώσσα Asp είναι μια μηχανή συγγραφής δυναμικών ιστοσελίδων και εφαρμογών ιστού. ASP σημαίνει Active Server Pages (8). Αναπτύσσεται από τη Microsoft, ενώ η πρώτη έκδοση της ονομαζόταν dbWeb και είχε αναπτυχθεί από την Aspect Software Engineering(1). Το κύριο χαρακτηριστικό της Asp είναι η στενή του συνεργασία με το λογισμικό web server της Microsoft, IIS (Internet Information Services). Η Asp μπορεί να αναγνωρίσει κώδικα σε οποιαδήποτε γλώσσα είναι

συμβατή με το «Microsoft Active Scripting» στάνταρ. Συνηθέστερα χρησιμοποιούμενη είναι η VBScript. Η επιλογή της γλώσσας που χρησιμοποιείται για την ερμηνεία του κώδικα γίνεται μέσω της εντολής @language ή μέσω του tag <script language="language" runat="server">.

Άλλες συμβατές γλώσσες περιλαμβάνουν τη Jscript (μια υλοποίηση ECMAScript από τη Microsoft) και την PerlScript. Ας τονιστεί εδώ ότι εκτός της Asp, η Microsoft αναπτύσσει και τη γλώσσα asp.net, η οποία έχει ουσιαστικές διαφορές με την asp: βασίζεται στο .Net framework της εταιρίας και χρησιμοποιεί just-in-time compilation τεχνικές. Αρχεία κώδικα της απλής Asp έχουν κατάληξη “.asp”, ενώ αρχεία asp.net έχουν συνήθως κατάληξη “aspx” .

3.3.1 Σχεδιαστική Φιλοσοφία και Συντακτικό

Η δυναμική δημιουργία περιεχομένου στη ASP γίνεται με τη χρήση αντικειμένων τα οποία υλοποιούν συγκεκριμένες λειτουργίες. Στην έκδοση 2, η ASP διαθέτει 6 τέτοια αντικείμενα: Application, ASPError, Request, Response, Server και Session. Κάθε αντικείμενο υλοποιεί κάποια βασική web-server λειτουργικότητα. Για παράδειγμα, το αντικείμενο Session αποθηκεύει δεδομένα από σελίδα σε σελίδα με τη χρήση των Browser Cookies.

Όμοια με τη php, στη κλασική asp επιτρέπεται η εμφώλευση html κώδικα με τη χρήση των αναγνωριστικών <% και %>. Το αναγνωριστικό <%= χρησιμοποιείται αντί του <% για να καταδείξει μια «έκφραση» (expression). Επίσης, όμοια με την php, μέθοδοι μπορούν να κληθούν και χωρίς τη χρήση παρενθέσεων.

Συναρτήσεις ορίζονται στα πρότυπα της εκάστοτε ενεργής γλώσσας, στη περίπτωση της VBScript με τα αναγνωριστικά function και end function, ενώ για σχολιασμό χρησιμοποιείται ο χαρακτήρας “'”:

Ένα σημαντικό μέρος της Asp λειτουργικότητας περιλαμβάνεται σε βιβλιοθήκες αντικειμένων, οι οποίες απλοποιούν επαναληπτικές διαδικασίες, όπως για παράδειγμα η παρουσίαση λίστας στοιχείων από μια βάση δεδομένων.

3.3.2 Το .Net framework και η καινοτομική αρχιτεκτονική της asp.net

Με την έλευση του .Net framework, η Microsoft προσπάθησε να υποστηρίξει την ανάπτυξη πολύπλοκων web-εφαρμογών μέσα από μια ολοκληρωμένη πλατφόρμα. Έτσι ξανασχεδίασε τις τεχνολογίες τις για τη δημιουργία δυναμικών ιστοσελίδων, με στόχο τη καλύτερη διασύνδεση με το .Net framework, τη καλύτερη κλιμάκωση και την ευκολότερη συντήρηση. Έτσι παρουσιάστηκε η asp.net (14). Όπως και με τη απλή asp, η asp.net μπορεί να διασυνδέσει κώδικα οποιασδήποτε γλώσσας με HTML στοιχεία, όμως συνίσταται η χρήση της γλώσσας C# για ανάπτυξη web-εφαρμογών.

Το πέρασμα στη νέα πλατφόρμα .net, σηματοδοτεί και μια καινοτομική φιλοσοφία στην ανάπτυξη σελίδων. Σε αντίθεση με παλαιότερες πρακτικές, η asp.net προσπαθεί να εξομοιώσει τον τρόπο λειτουργίας και προγραμματισμού του γραφικού περιβάλλοντος των windows στη συγγραφή σελίδων. Παρέχει μια πλήρη σειρά από γραφικά αντικείμενα (κουμπιά, λίστες, πίνακες, data views κλπ) με τα οποία ο προγραμματιστής μπορεί να αναπτύξει σελίδες, με τρόπο που ομοιάζει περισσότερο με ανάπτυξη εφαρμογής παρά σελίδας. Ενώ όμως σε μια παραθυρική εφαρμογή τα

διάφορα γραφικά στοιχεία σχεδιάζονται χρησιμοποιώντας κλήσεις γραφικών, στην asp.net τα εκάστοτε στοιχεία σχεδιάζονται παράγοντας HTML και javascript κώδικα. Ένα σύστημα Μηνυμάτων και Γεγονότων (messages & events) χρησιμοποιείται για τη διασύνδεση των στοιχείων αυτών με κώδικα asp.net. Ουσιαστικά γίνεται προσπάθεια να δημιουργηθεί ένα περιβάλλον ανταλλαγής μηνυμάτων μέσα από το (stateless) html πρωτόκολλο. Με τον τρόπο αυτό απλοποιείται σημαντικά η δημιουργία πολύπλοκων web-εφαρμογών (1).

Μια σημαντική διαφορά της Asp.net σε σχέση με τη απλή asp είναι η χρήση μεταγλωττισμένου κώδικα. Για να διατηρηθεί η ευχρηστία ανάπτυξης κώδικα, ο εξυπηρετητής (Microsoft IIS) μεταφράζει αυτόματα (just in time) τον κώδικα ενός συνόλου σελίδων asp.net (οι οποίες έχουν συνήθως κατάληξη .aspx) και παράγει ένα .Net module το οποίο **τρέχει εγγενώς** στην αρχιτεκτονική (windows server) (8)(14). Αυτό συνεπάγεται μια σημαντική καθυστέρηση κατά τη μεταγλώττιση της σελίδας, όταν υπάρχουν αλλαγές. Καθώς όμως ο κώδικας που παράγεται είναι εγγενής για την αρχιτεκτονική, οι μετέπειτα επισκέψεις στην ίδια σελίδα επιταχύνονται εντυπωσιακά. Ο εξυπηρετητής γνωρίζει αν ο χρήστης έκανε αλλαγές σε κάποιο αρχείο κώδικα και το μεταγλωττίζει αυτόματα. Έτσι η επεξεργασία κώδικα πάνω στο server απλοποιείται σημαντικά. Ένα πρόβλημα που δημιουργείται λόγω της δημιουργίας εγγενούς κώδικα είναι η ασυμβατότητα με non-microsoft αρχιτεκτονικές εξυπηρετητή.

3.4 Javascript

Η javascript είναι η βασική γλώσσα προγραμματισμού της ιστοσελίδας από τη άκρη του χρήστη. Χρησιμοποιείται για να περιγράψει διαδικασίες που πρέπει να γίνουν στο φυλλομετρητή (browser) (8).

3.4.1 Σχεδιαστική φιλοσοφία

Στοιχεία javascript εμφωλεύονται σε html στοιχεία μέσω του tag <script>. Τα στοιχεία αυτά, τα οποία μπορούν να περιλαμβάνουν κώδικα in-line ή να αναφέρονται σε αρχεία κώδικα, ανιχνεύονται από τον φυλλομετρητή, ο οποίος και διερμηνεύει τον κώδικα σε πραγματικό χρόνο.

Η javascript συνήθως χρησιμοποιείται για τη δυναμική διαχείριση των διαφόρων στοιχείων μιας ιστοσελίδας σε πραγματικό χρόνο και χωρίς να απαιτείται αλληλεπίδραση πελάτη-εξυπηρετητή. Για παράδειγμα, μέσω javascript ο προγραμματιστής μπορεί να μετακινήσει ομάδες στοιχείων, να κρύψει στοιχεία ή να αλλάξει τα περιεχόμενα τους δυναμικά. Για να γίνει αυτό, πρέπει πρώτα να αποκτήσει πρόσβαση στα στοιχεία αυτά (δηλαδή στο Document Object Model), χρησιμοποιώντας τις μεθόδους που παρέχει ο φυλλομετρητής.

Δυστυχώς το πρότυπο της Javascript δεν επιβάλλει κάποια κοινή ονοματολογία για τη πρόσβαση στα στοιχεία μιας ιστοσελίδας. Έτσι κάθε φυλλομετρητής παρέχει τις δικές του μεθόδους για τη πρόσβαση στα στοιχεία αυτά. Αυτός είναι ο κύριος λόγος που η ανάπτυξη κώδικα javascript είναι δύσκολη και επίπονη διαδικασία: απαιτείται να αναπτυχθεί ειδικός κώδικας για την υποστήριξη των διαφορετικών φυλλομετρητών. Αυτός εξ άλλου είναι και ο λόγος που η χρήση της

javascript ήταν περιορισμένη μέχρι την έλευση κάποιων wrapper-βιβλιοθηκών και τη βελτίωση της υποστήριξης της από τους φυλλομετρητές.

Για την επίλυση του παραπάνω προβλήματος, η επιτροπή προτύπων του παγκόσμιου ιστού έχει προτείνει ένα κοινό πρότυπο, το HTML DOM (html Document Object Model), το οποίο φαίνεται να κερδίζει έδαφος και να υποστηρίζεται από όλο και περισσότερους φυλλομετρητές.

3.4.2 Σύνταξη και χαρακτηριστικά

Κάθε φυλλομετρητής υλοποιεί τον δικό του διερμηνευτή για javascript (συνήθως καλείται javascript engine), ο οποίος μάλιστα είναι και καθοριστικής σημασίας για την απόδοση του φυλλομετρητή όσον αφορά τη συμβατότητα, σταθερότητα, ταχύτητα και ασφάλεια.

Συνήθως ο κώδικας javascript διερμηνεύεται, δε μεταγλωττίζεται. Εντούτοις για να βελτιωθεί η απόδοση, υπάρχουν φυλλομετρητές που ακολουθούν μεθόδους just-in-time μεταγλώττισης, όπως ο Google Chrome με μια καινοτομική javascript engine (οι οποία ονομάζεται V8 και είναι ανοιχτού κώδικα)

Η σύνταξη javascript είναι κοινή μεταξύ φυλλομετρητών, παρόλο που υπάρχουν διαφορές στην ονοματολογία των βασικών μεθόδων. Η σύνταξη θυμίζει php, με τη διαφορά ότι δεν χρησιμοποιείται το χαρακτηριστικό '\$' στα ονόματα των μεταβλητών. Πρόκειται για μια γλώσσα χαλαρών τύπων όπου οι μεταβλητές μπορούν να αλλάξουν τύπο δυναμικά. Αλφαριθμητικά μεταβλητού μεγέθους υποστηρίζονται εγγενώς. Για τη δήλωση μιας μεταβλητής χρησιμοποιείται η λέξη-κλειδί "var" ενώ για τον ορισμό συνάρτησης χρησιμοποιείται η λέξη "function", με ένα συντακτικό όμοιο της php (δεν απαιτείται να δηλωθεί ο τύπος του αποτελέσματος). Δεν υπάρχουν δείκτες, ενώ υποστηρίζονται τάξεις.

Τις περισσότερες φορές ζητείται η χρήση της javascript όταν ο χρήστης πράξει μια συγκεκριμένη ενέργεια (πχ πατήσει κάποιο πλήκτρο ή μετακινήσει κάποιο στοιχείο της σελίδας). Για το λόγο αυτό τα διάφορα HTML στοιχεία ορίζουν συγκεκριμένα συμβάντα, και δέχονται ως παράμετρο μια σειρά από javascript εντολές. Στη περίπτωση αυτή, ο κώδικας javascript παρέχεται in-line στον ορισμό του html-στοιχείου, και συγκεκριμένα ως παράμετρος κειμένου μέσα στο HTML tag του στοιχείου. Για παράδειγμα, ένα html button element ορίζει ένα "onClick" event, το οποίο δέχεται javascript κώδικα κατά τη σύνταξη:

```
<button onclick="javascriptfunction();" > button text </button>
```

Στο πεδίο onclick μπορεί να εντεθεί πιο πολύπλοκος κώδικας, όπως στο παρακάτω παράδειγμα, όπου τα στοιχεία

ενός πεδίου αντιγράφονται σε ένα άλλο:

```
<html><body>
```

```
A: <input type="text" id="field1" value="Hello World!"><br />
```

```
B: <input type="text" id="field2"><br /><br />
```

Click the button below to copy the content of field A to field B.

```
<br />
```

```
<button onclick="document.getElementById('field2').value=
document.getElementById('field1').value">Copy Text</button>
```

```
</body></html>
```

Η δυνατότητα αυτή απλοποιεί σημαντικά τον προγραμματισμό σε javascript, αλλά καθώς ενοποιεί τον html με τον javascript κώδικα, μπορεί να δημιουργήσει προβλήματα: αφενός είναι δυσκολότερο για το προγραμματιστή να διακρίνει τα διάφορα στοιχεία javascript κώδικα όταν κάνει αποσφαλμάτωση, αφετέρου εισάγει σοβαρά ζητήματα ασφαλείας: Για παράδειγμα, έστω ένα διαδικτυακό forum, στο οποίο οι χρήστες μπορούν να εισάγουν κείμενο σε μορφή html. Έστω λοιπόν ότι ο χρήστης ενός forum εισάγει το παρακάτω html-formatted μήνυμα:

Check this picture:

```
<a href="" onclick="alert('attack')"> picture </a>
```

Το μήνυμα του θα αποθηκευτεί σε μια βάση δεδομένων ως απλό κείμενο. Όταν όμως κάποιος χρήστης προσπαθήσει να το διαβάσει, αυτό το κείμενο **θα εντεθεί σε html κώδικα** και ο φυλλομετρητής του θα θεωρήσει το <a> πεδίο ως μέρος του κώδικα αυτού. Όταν ανυποψίαστοι χρήστες ακολουθήσουν το σύνδεσμο στην υποτιθέμενη εικόνα, θα εκτελεστεί στους υπολογιστές τους ο javascript κώδικας του κακόβουλου χρήστη! Μάλιστα, δεν υπάρχει τρόπος, εφόσον πρόκειται για html στοιχείο, ο φυλλομετρητής να γνωρίζει ότι το onclick event προέρχεται από κάποιον κακόβουλο χρήστη και όχι από το (υποτιθέμενα) ασφαλές forum. Για το λόγο αυτό πρέπει ο προγραμματιστής του forum να φιλτράρει κατάλληλα κάθε μήνυμα για να διαγράψει κακόβουλα javascript στοιχεία. Εντούτοις, η διαδικασία αυτή είναι δύσκολη, καθώς δεν υπάρχει κάποια ιδιαίτερη σημειογραφία για τον javascript κώδικα (δηλαδή δεν υπάρχει κάποιο αναγνωριστικό του τύπου <?rhp ή <%= που να σηματοδοτεί τον javascript κώδικα)

3.5 Flash & Flex

Η τεχνολογία Flash αναπτύχθηκε από τη Macromedia, η οποία αργότερα εξαγοράστηκε από την Adobe. Πρόκειται για μια τεχνολογία ανάπτυξης δυναμικών ιστοσελίδων που χρησιμοποιείται ευρύτατα στο διαδίκτυο, χάρις στην υποστήριξη πολυμεσικών χαρακτηριστικών (15).

Το Flash αποτελείται από δύο βασικά στοιχεία, τα οποία αλληλεπιδρούν έντονα μεταξύ τους.

1. Μια μηχανή ορισμού, προβολής και κίνησης διανυσματικών γραφικών με προχωρημένες πολυμεσικές δυνατότητες (rendering engine).

2. Μια γλώσσα προγραμματισμού, την actionscript, η οποία μπορεί να διαχειριστεί πολυμεσικά αντικείμενα και να υποστηρίξει την ανάπτυξη ολοκληρωμένων εφαρμογών. Αρχικά το

flash είχε σχεδιαστεί ως περιβάλλον ανάπτυξης, σχεδιασμού και παρουσίασης διαστάσεων γραφικών με κίνηση και ήχο. Αργότερα επεκτάθηκε με την εισαγωγή ολοκληρωμένων στοιχείων GUI (κουμπιά, λίστες κλπ) αλλά και με τη δυνατότητα αναπαραγωγής video. Επίσης, βελτιώθηκε σημαντικά η δυνατότητα προγραμματισμού της γενικής λογικής αλλά και των γραφικών στοιχείων, ενώ η ενσωματωμένη γλώσσα προγραμματισμού βελτιστοποιήθηκε και έγινε πιο αντικειμενοστραφής. Αυτή τη στιγμή το flash video είναι ο πιο ευρέως χρησιμοποιούμενος τρόπος μετάδοσης video, σε πραγματικό χρόνο, στο διαδίκτυο, ενώ το flash χρησιμοποιείται σε ένα μεγάλο αριθμό από ιστοσελίδες ως βασικό στοιχείο που βελτιώνει τη διάδραση με το χρήστη.

Η τεχνολογία αποτελείται από μια μικρή εφαρμογή (player) που αναλαμβάνει να προβάλλει κατάλληλα επεξεργασμένο, μεταγλωττισμένο περιεχόμενο (το οποίο μπορεί να περιέχει πολυμεσικό υλικό ή και κώδικα), και από μια εφαρμογή ανάπτυξης περιεχομένου, η οποία ενοποιεί με σχετική κομψότητα τις διαδικασίες επεξεργασίας κινούμενων γραφικών και ανάπτυξης προγραμμάτων.

Τα αρχεία flash που προορίζονται για προβολή σε ιστοσελίδα, καλούνται compiled (μεταγλωττισμένα) αρχεία και συνήθως έχουν κατάληξη “.swf”. Τα αρχεία που χρησιμοποιούνται από το σχεδιαστή και προγραμματιστή για τη δημιουργία περιεχομένου συνήθως έχουν κατάληξη “.fla”. Στο εξής θα καλούμε τα αρχεία αυτά «πηγαία αρχεία»

Ένα .fla αρχείο περιλαμβάνει διανυσματικά δεδομένα γραφικών, εικόνες και ήχους, τα οποία και επεξεργάζονται ώστε να καθοριστούν οι οπτικές και κινητικές τους ιδιότητες από την εφαρμογή ανάπτυξης περιεχομένου. Η εφαρμογή προβολής περιεχομένου (player) δεν μπορεί να αναγνωρίσει μη-μεταγλωττισμένα αρχεία.

Τα μεταγλωττισμένα flash αρχεία μπορούν να εισαχθούν σε μια ιστοσελίδα και να παρουσιασθούν στο χρήστη, εφόσον αυτός έχει εγκαταστήσει στο σύστημά του την εφαρμογή προβολής (player). Δηλαδή το περιεχόμενο flash **δεν υποστηρίζεται εγγενώς** από το φυλλομετρητή και δεν παρεμβάλει html στοιχεία στο κώδικα. Αντίθετα, εισάγεται στην ιστοσελίδα ως **ειδικό αντικείμενο**, με τρόπο παρόμοιο με εκείνον που εισάγονται εικόνες ή ήχοι. Το ειδικό αυτό αντικείμενο ανιχνεύεται από τον υπολογιστή του χρήστη, ο οποίος καλείται να εγκαταστήσει το λογισμικό διαχείρισης τέτοιων αντικειμένων.

3.5.1 Γραφικές και πολυμεσικές δυνατότητες

Η τεχνολογία επιτρέπει στο χρήστη να σχεδιάσει εκ του μηδενός πολύπλοκες πολυμεσικές παρουσιάσεις με γραφικά που μεταβάλλονται στο χρόνο. Όλα τα πολυμεσικά δεδομένα που προβάλλονται στο χρήστη τοποθετούνται και διαχειρίζονται σε μία ειδική αφαιρετική δομή η οποία καλείται «σκηνή». Η σκηνή αντιστοιχεί με **έναν καμβά** στον οποίο ένας ζωγράφος θα δημιουργήσει το έργο του. Ο σχεδιαστής μπορεί να κατασκευάσει μια αλληλουχία από σκηνές, η κάθε μία από τις οποίες περιέχουν ένα σύνολο μεταβλητών στο χρόνο, διαδραστικών και προγραμματιζόμενων στοιχείων (μπορούν να περιέχουν και άλλες σκηνές) και να τις προβάλλει στο χρήστη κατά βούληση, μεταγλωττίζοντάς τες.

Το περιβάλλον επεξεργασίας, το οποίο είναι αντίστοιχο με αυτό ενός programming IDE (integrated development environment) επιτρέπει στους χρήστες να επεξεργαστούν όλα τα δεδομένα και ακολούθως να παράγουν τα compiled flash αρχεία που θα συμπεριληφθούν στις ιστοσελίδες τους. Οι χρήστες του περιβάλλοντος επεξεργασίας έχουν μεταξύ άλλων τις παρακάτω δυνατότητες:

- Να σχεδιάσουν, χρησιμοποιώντας τα ενσωματωμένα εργαλεία του συστήματος, οποιοδήποτε διανυσματικό σχέδιο (γραφικό)
- Να συμπεριλάβουν στη σκηνή εικόνες, τις οποίες μπορούν ακολούθως να επεξεργαστούν
- Να δημιουργήσουν συλλογές αντικειμένων
- Να ομαδοποιήσουν αντικείμενα και να δημιουργήσουν μικρότερες υπο-σκηνές, οι οποίες λειτουργούν ως «ηθοποιοί» στην ευρύτερη σκηνή
- Να κινήσουν αντικείμενα στο χώρο και να αλλάξουν δυναμικά τις ιδιότητες τους στο χρόνο.
- Να ορίσουν εξωτερικές πηγές δεδομένων στο διαδίκτυο και να τις συμπεριλάβουν δυναμικά στη σκηνή τους
- Να ορίσουν βασικές ιδιότητες των διαδραστικών στοιχείων και να ορίσουν πλήρως τις οπτικές τους ιδιότητες

Οι παραπάνω ιδιότητες γίνονται με εντελώς γραφικά μέσα, χωρίς να απαιτείται προγραμματισμός. Εντούτοις ένα σύνολο από δυνατότητες που παρέχει το flash απαιτούν την εισαγωγή και χρήση κώδικα ώστε να αξιοποιηθούν στο έπακρο.

3.5.2 Ο προγραμματισμός στο flash

Παράλληλα με τα γραφικά, το flash διαθέτει και μια πολύ ανεπτυγμένη γλώσσα προγραμματισμού, την actionscript, η οποία εκτός των άλλων μπορεί να αλληλεπιδράσει και επεξεργαστεί όλα τα γραφικά αντικείμενα της σκηνής (αντίστοιχα με τη javascript σε μια ιστοσελίδα) αλλά και να δημιουργήσει δυναμικά νέα αντικείμενα. Το περιβάλλον ανάπτυξης ενσωματώνει δυνατότητες ανάπτυξης κώδικα σε όλα τα εργαλεία του. Με τον τρόπο αυτό η συγγραφή actionscript κώδικα απλοποιείται: Σχεδιαστές μπορούν να ενσωματώσουν κώδικα actionscript τόσο σε πολυμεσικά στοιχεία (ορίζοντας δυναμικά τη συμπεριφορά τους) όσο και σε συγκεκριμένα γεγονότα ή σημεία στο χώρο και στο χρόνο.

Όμοια με την Html και τη javascript, η ένθεση κώδικα στο περιεχόμενο αφενός απλοποιεί την ανάπτυξη μικρών εφαρμογών, αφετέρου κάνει τη διαχείριση του κώδικα πιο σύνθετη. Για το λόγο αυτό επιτρέπονται τεχνικές “code-behind” δηλαδή η σύζευξη στοιχείων με κώδικα actionscript ο οποίος υπάρχει σε εξωτερικά αρχεία κειμένου. Ο εξωτερικός κώδικας διαβάζεται όταν παράγεται το μεταγλωττισμένο flash αρχείο και δε μπορεί να μεταγλωττιστεί σε

χρόνο εκτέλεσης (run time) αλλά μόνο σε χρόνο μεταγλώττισης. Συνεπώς, όταν ο προγραμματιστής αλλάξει οποιοδήποτε στοιχείο του κώδικα, είτε εσωτερικά στο πηγαίο flash αρχείο είτε σε κάποιο εξωτερικό αρχείο κώδικα, απαιτείται να μεταγλωττίσει ξανά την εφαρμογή flash και να παράγει ένα μεταγλωττισμένο flash αρχείο(16)(15).

4 ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΒΑΣΩΝ ΔΕΔΟΜΕΝΩΝ

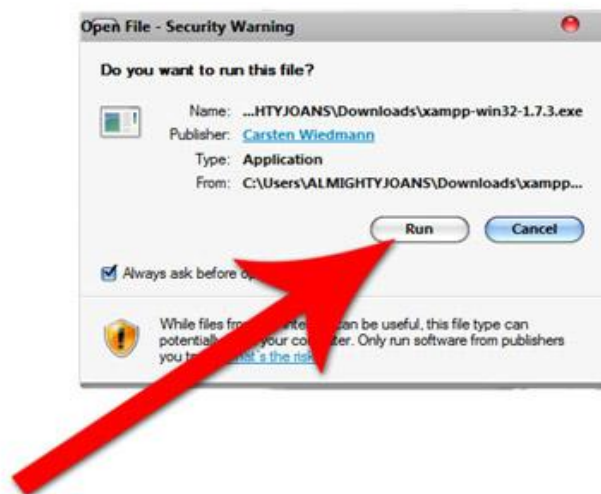
4.1 ΧΑΜΡΡ

Το ΧΑΜΡΡ είναι ένα πακέτο προγραμμάτων λογισμικού ανοιχτού κώδικα το οποίο περιέχει τον εξυπηρετητή σελίδων Apache, τη βάση δεδομένων MySQL και ένα διερμηνέα για τις γλώσσες προγραμματισμού PHP και Perl.

4.1.1 Εγκατάσταση ΧΑΜΡΡ

Για την εγκατάσταση του πακέτου προγραμμάτων λογισμικού ανοιχτού κώδικα ΧΑΜΡΡ ακολουθήσαμε τα έξης βήματα:

1. Εισαγωγή στο πρόγραμμα περιήγησης τον σύνδεσμο <http://www.apachefriends.org/en/xampp-windows.html>
2. Λήψη του εργαλείου
3. Εγκατάσταση του xampp



Εικόνα 1:Εγκατάσταση xampp

4. Αποδεχόμαστε τις προεπιλεγμένες ρυθμίσεις. Άπλα πατάμε το κουμπί enter.

```
C:\Windows\system32\cmd.exe

#####
# XAMPP 1.7.3 - Setup
# Copyright 2009 Carsten Wiedmann <FreeBSD License>
# Authors: Carsten Wiedmann <carsten_stt@t@gnx.de>
#           Kay Vogelsang <kvo@apachefriends.org>
#####
Should I add shortcuts to the startmenu/desktop? <y/n>: y
```

Εικόνα 2: Ρυθμίσεις xampp

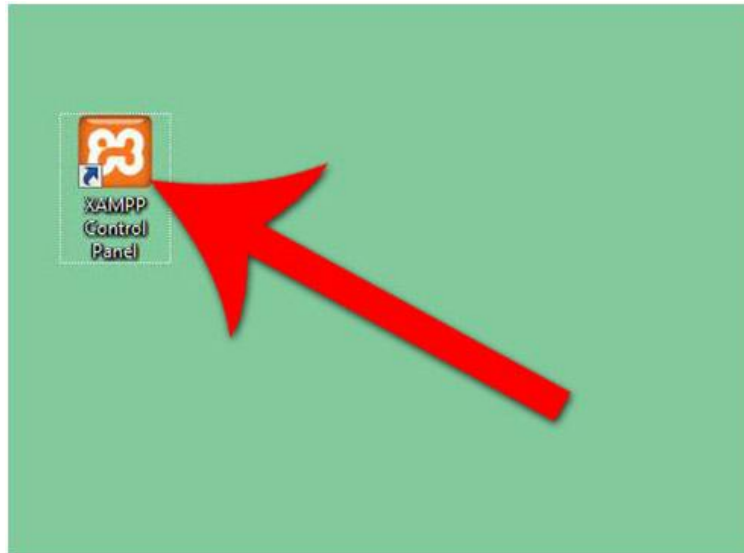
5. Όταν θα έχει ολοκληρωθεί η εγκατάσταση άπλα κλείνουμε το παράθυρο.

```
C:\Windows\system32\cmd.exe

#####
# XAMPP 1.7.3 - Setup
# Copyright 2009 Carsten Wiedmann <FreeBSD License>
# Authors: Carsten Wiedmann <carsten_stt@t@gnx.de>
#           Kay Vogelsang <kvo@apachefriends.org>
#####
1. start XAMPP Control Panel
2. relocate XAMPP
   <current path: C:\xampp>
3. disable HTTPS <SSL>
4. disable Server Side Includes <SSI>
5. enable IPv4 only <current: IPv4/6 <auto>>
6. disable mod_perl
7. disable Apache::ASP
x Exit
Please choose <1-7/x>: x
```

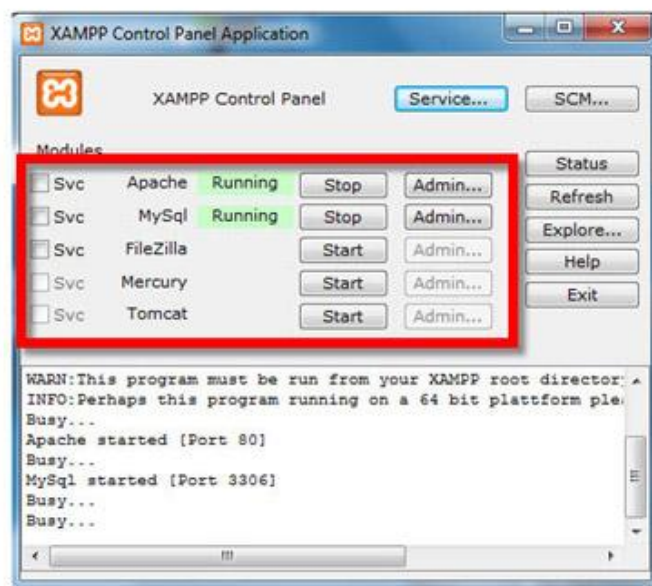
Εικόνα 3: Ολοκλήρωση Ρυθμίσεων xampp

6. Εκκίνηση του XAMPP control panel



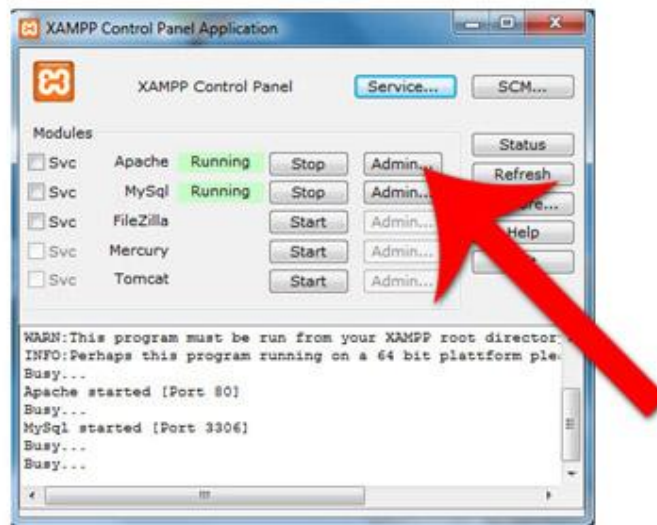
Εικόνα 4:Εκκίνηση xampp Control Panel

7. Εκκίνηση του Apache και MySQL.



Εικόνα 5:Εκκίνηση Apache και Mysql

8. Επαληθεύουμε την εγκατάσταση του Apache με το να επιλέξουμε τον σύνδεσμο του διαχειριστή του Apache από τον πίνακα ελέγχου.



Εικόνα 6: Επιλογή admin λειτουργίας για τον Apache

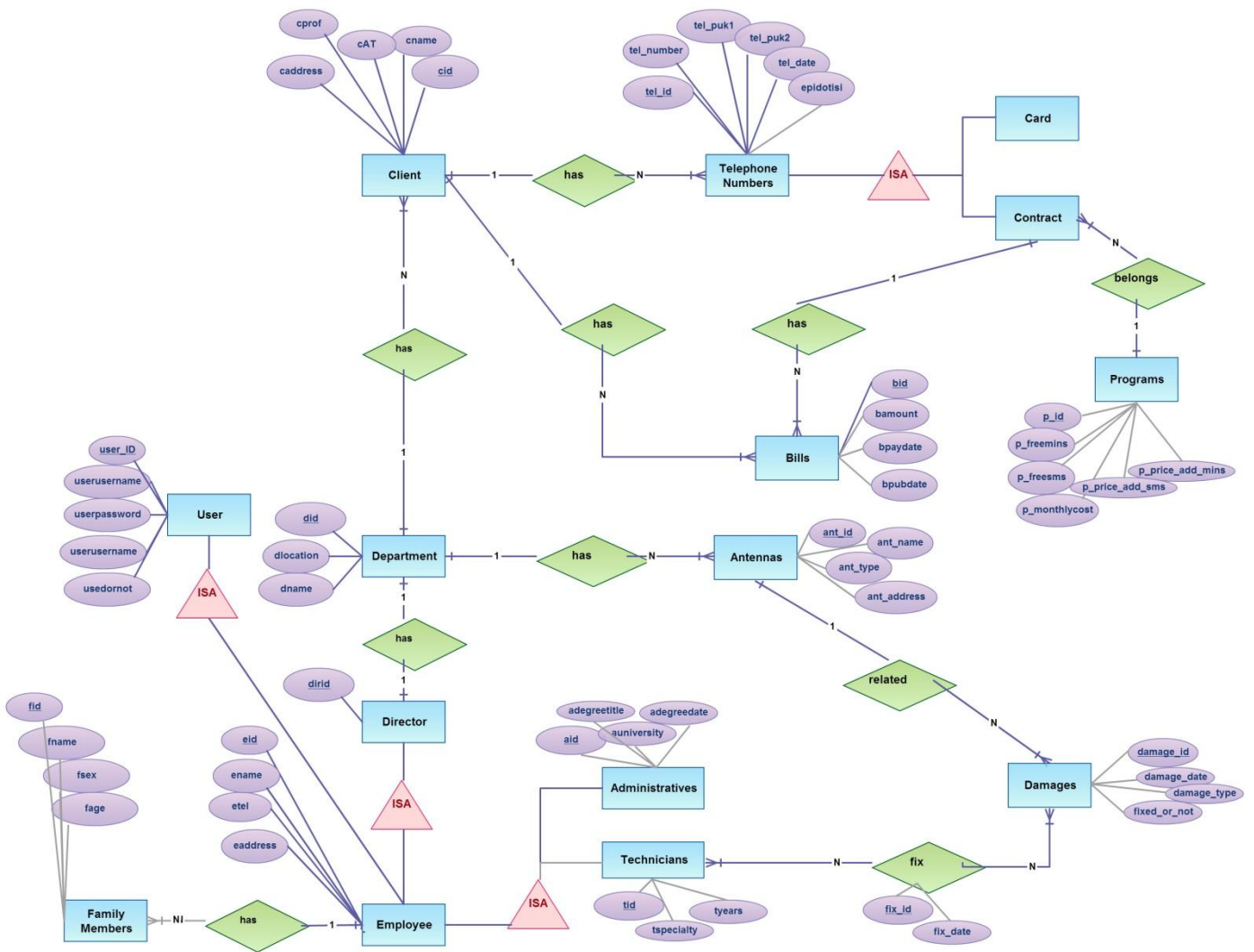
9. Ακολουθούμε την αντίστοιχη διαδικασία με τον Apache για την MySQL. Αν η εγκατάσταση του XAMPP έχει πραγματοποιηθεί με επιτυχία όταν θα ανοίξουμε ένα πρόγραμμα περιήγησης (web browser) θα εμφανιστεί ένα μήνυμα που θα λέει ότι έχουμε εγκαταστήσει με επιτυχία το XAMPP στο σύστημα μας.



Εικόνα 7: Ολοκλήρωση εγκατάστασης xampp

5 ΔΙΑΓΡΑΜΜΑ ΟΝΤΟΤΗΤΩΝ ΣΧΕΣΕΩΝ

E-R Diagram for Mobile Company Management System



Εικόνα 8: Διάγραμμα Οντοτήτων Σχέσεων

6 ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ

Η μετάφραση του διαγράμματος οντοτήτων-σχέσεων στο σχεσιακό μοντέλο σύμφωνα με τις πληθικότητες που προκύπτουν ανάμεσα στις οντότητες. Στους παρακάτω πίνακες του σχεσιακού μοντέλου φαίνονται με υπογράμμιση τα πρωτεύοντα κλειδιά όπως επίσης (italic – όχι bold, στο τέλος κάθε πίνακα) τα κλειδιά τα οποία κληρονομούνται από την μία οντότητα στην άλλη βάσει των πληθικότητων.

USERS

<u>userID</u>	userRole	userusername	userPassword	usedornot
---------------	----------	--------------	--------------	-----------

DEPARTMENTS

<u>did</u>	dname	dlocation	<i>dirid</i>
------------	-------	-----------	--------------

EMPLOYEES

<u>eid</u>	efirstname	elastname	eaddress	etel	eAT	esalary	<i>did</i>	<i>userID</i>
------------	------------	-----------	----------	------	-----	---------	------------	---------------

ADMINISTRATIVES

<u>aid</u>	adegree_title	auniversity	adegree_date	<i>eid</i>
------------	---------------	-------------	--------------	------------

TECHNICIANS

<u>tid</u>	tspecialty	tyears	<i>eid</i>
------------	------------	--------	------------

DIRECTORS

<u>dirid</u>	<i>eid</i>	<i>did</i>
--------------	------------	------------

FAMILY_MEMBERS

<u>fid</u>	fname	fsex	fage	<i>eid</i>
------------	-------	------	------	------------

CLIENTS

<u>cid</u>	cname	caddress	cAT	cprof	<i>did</i>
------------	-------	----------	-----	-------	------------

ANTENNAS

<u>ant_id</u>	ant_name	ant_address	ant_type	<i>did</i>
---------------	----------	-------------	----------	------------

TELEPHONES

<u>tel_id</u>	tel_number	tel_puk1	tel_puk1	tel_type	tel_date	epidotisi	<i>cid</i>	<i>p_id</i>	<i>eid</i>	<i>p_monthlycost</i>
---------------	------------	----------	----------	----------	----------	-----------	------------	-------------	------------	----------------------

PROGRAMS

<u>p_id</u>	p_name	p_freemins	p_freesms	p_monthlycost	p_price_add_mins	p_price_add_sms
-------------	--------	------------	-----------	---------------	------------------	-----------------

BILLS

<u>bid</u>	bamount	bpubdate	bpaydate	<i>tel_id</i>	<i>cid</i>
------------	---------	----------	----------	---------------	------------

DAMAGES

<u>damage_id</u>	damage_date	damage_type	fixed_or_not	<i>ant_id</i>
------------------	-------------	-------------	--------------	---------------

FIXANTENNASDAMAGES

<u>fix_id</u>	fix_date	t_id	damage_id
---------------	----------	------	-----------

Ο πίνακας FixAntennasDamages δημιουργήθηκε από τη N-N σχέση involves1 μεταξύ των οντοτήτων Antennas και Damages.

7 ΔΗΜΙΟΥΡΓΙΑ ΒΑΣΗΣ

Για να δημιουργήσουμε τη βάση μας θα ακολουθήσουμε τα παρακάτω βήματα.

Ανοίγουμε τη **Γραμμή Εντολών (Command Prompt)** στον υπολογιστή μας.

Start -> πληκτρολόγηση cmd ->Enter

Πηγαίνουμε στο σωστό φάκελο όπου βρίσκεται εγκατεστημένη η mysql, γράφοντας την εντολή:

```
cd C:\xampp\mysql\bin
```

Μπαίνουμε μέσα στη mysql, γράφοντας την εντολή:

```
mysql -u root -p
```

και όταν μας ζητάει password, πατάμε enter κατευθείαν.

Η πρώτη εντολή

```
create database mobiledb;
```

δημιουργεί τη βάση για το σύστημά μας ενώ με τη δεύτερη εντολή, ξεκινάμε να τη χρησιμοποιούμε

```
use mobiledb;
```

Με τις επόμενες εντολές δημιουργούμε τους πίνακες που προέκυψαν από τη μετάφραση του σχεσιακού μοντέλου, δηλώνοντας τα πρωτεύοντα και τα ξένα κλειδιά βάση των κανόνων κληρονόμησης.

```
create table users(  
    userID          integer not null auto_increment,  
    userRole        varchar(10) not null,  
    username        varchar(20) not null,  
    userPassword    varchar(20) not null,  
    usedornot       integer,  
    primary key (userID)  
);
```

CREATE TABLE departments

```
( did            integer not null auto_increment,  
  dname          varchar(32) not null,  
  dlocation     varchar(32) not null,  
  
  dirid         integer,  
  
  primary key(did),  
  foreign key(dirid) references directors );
```

CREATE TABLE employees

```
( eid            integer not null auto_increment,  
  efirstname    varchar(32) not null,  
  elastname     varchar(32) not null,  
  eaddress      varchar(32) not null,  
  etel         varchar(32) not null,  
  eAT          varchar(32) not null,  
  esalary      float(32) not null,  
  
  did           integer not null,  
  userID       integer not null,  
  
  primary key(eid),  
  foreign key(did) references departments,  
  foreign key(userID) references users  
);
```

CREATE TABLE family_members

```
( fid            integer not null auto_increment,  
  fname         varchar(32) not null,  
  fsex         integer not null,  
  fage        varchar(32) not null,  
  
  eid          integer not null,
```

```
    primary key(fid),
    foreign key(eid) references employes
);
```

CREATE TABLE directors

```
( dirid          integer not null auto_increment,

    eid           integer not null,
    did           integer not null,

    primary key(dirid),
    foreign key(did) references departments ,
    foreign key(eid) references employees
);
```

CREATE TABLE administratives

```
( aid           integer not null auto_increment,

    adegree_title varchar(32) not null,
    auniversity varchar(32) not null,
    adegree_date date,

    eid           integer not null,

    primary key(aid),
    foreign key(eid) references employees
);
```

CREATE TABLE technicians

```
( tid           integer not null auto_increment,

    tspecialty varchar(32) not null,
    tyears integer not null,
```

```
    eid                integer not null,  
  
    primary key(tid),  
    foreign key(eid) references employees  
);
```

CREATE TABLE antennas

```
( ant_id              integer not null auto_increment,  
  ant_name            varchar(32) not null,  
  ant_address         varchar(32) not null,  
  ant_type            varchar(32) not null,  
  
  did                 integer not null,  
  
  primary key(ant_id),  
  foreign key(did) references departments  
);
```

CREATE TABLE clients

```
( cid                 integer not null auto_increment,  
  cname               varchar(32) not null,  
  caddress            varchar(32) not null,  
  cAT                 varchar(32) not null,  
  cprof               varchar(32) not null,  
  
  did                 integer not null,  
  
  primary key(cid),  
  foreign key(did) references departments  
);
```

CREATE TABLE telephones

```
( tel_id          integer not null auto_increment,
  tel_number      varchar(32) not null,
  tel_puk1       varchar(32) not null,
  tel_puk2       varchar(32) not null,
  tel_type       varchar(32) not null,

  cid            integer not null,
  p_id          integer,
  eid           integer not null,

  tel_date      date,

  p_monthlycost float,
  epidotisi    float,

  primary key(tel_id),
  foreign key(cid) references clients ,
  foreign key(p_id) references programs,
  foreign key(eid) references employees ,
  foreign key(p_monthlycost) references programs
);
```

CREATE TABLE programs

```
( p_id          integer not null auto_increment,
  p_name       varchar(32) not null,
  p_freemins   integer not null,
  p_freesms    integer not null,
  p_monthlycost float not null,
  p_price_add_mins float not null,
  p_price_add_sms float not null,

  primary key(p_id)
);
```


CREATE TABLE bills

```
( bid                integer not null auto_increment,
  bamount            float not null,
  bpubdate           date,
  bpaydate           date,

  tel_id             integer not null,
  cid                integer not null,

  primary key(bid),
  foreign key(tel_id) references telephones,
  foreign key(cid) references clients
);
```

CREATE TABLE damages

```
( damage_id          integer not null auto_increment,
  damage_date         date,
  damage_type         varchar(32) not null,
  fixed_or_not        integer,

  ant_id              integer,

  primary key(damage_id),
  foreign key(ant_id) references antennas
);
```

CREATE TABLE FixAntennasDamages

```
( fix_id             integer not null auto_increment,

  fix_date           date,

  tid                integer not null,
  damage_id          integer not null,
```

```
primary key(fix_id),
foreign key(tid) references technicians ,
foreign key(damage_id) references damages
);
```

Με την ακριβώς επόμενη εντολή δημιουργούμε τον πρώτο χρήστη του συστήματος μας, στον οποίο δίνουμε το ρόλο του administrator έτσι ώστε να έχει τον απόλυτο έλεγχο του όλου συστήματος.

```
insert into users (userRole, userusername, userPassword, usedornot) values ('admin', 'admin', 'admin', 0);
```

Οι ρόλοι του συστήματος θα είναι 2, ο admin και ο απλός user.

Ο διαχωρισμός έγινε για να μπορεί να υπάρχει γενικός έλεγχος στο σύστημα από την πλευρά του admin και γιατί ο απλός χρήστης θα έχει τη δυνατότητα χρήσης μόνο όσων λειτουργιών χρειάζεται.

Οι λειτουργίες που υποστηρίζει το σύστημα είναι οι εξής:

1. Login - Logout
 2. Προσθήκη – Διαγραφή Χρήστη
 3. Προσθήκη – Διαγραφή Τμήματος
 4. Προσθήκη – Διαγραφή Υπαλλήλου
 5. Προσθήκη – Διαγραφή Τύπου Υπαλλήλου Τεχνικό/Διοικητικό
 6. Προσθήκη – Διαγραφή Διευθυντή
 7. Προσθήκη – Διαγραφή Τηλ.Προγραμμάτων
 8. Προσθήκη – Διαγραφή Οικογενειακών Μελών
 9. Προσθήκη – Διαγραφή Πελάτη
 10. Προσθήκη – Διαγραφή Τηλεφωνικού αριθμού
 11. Προσθήκη – Διαγραφή Λογαριασμού
 12. Προσθήκη – Διαγραφή Κεραίας
 13. Προσθήκη – Διαγραφή Βλάβης
 14. Προσθήκη – Διαγραφή Επιδιόρθωσης
 15. Ετήσια Ανανέωση Παγίων - Επιδότηση
 16. Στατιστικά
- ✓ Ετήσια Συγκεντρωτική **Εσόδων** ανά **Τμήμα** ανά μήνα
 - ✓ Ετήσια Συγκεντρωτική **Συμβολαίων** ανά **Υπάλληλο** ανά μήνα
 - ✓ Ετήσια Συγκεντρωτική **Συμβολαίων** ανά **Τμήμα** ανά μήνα
 - ✓ Ετήσια Συγκεντρωτική **Καρτοτηλεφώνων** ανά **Τμήμα** ανά μήνα

Ο παρακάτω πίνακας δείχνει ποιες λειτουργίες έχει διαθέσιμες ο διαχειριστής (admin) και ποιες ένας απλός χρήστης του συστήματος (user).

ΛΕΙΤΟΥΡΓΙΕΣ	Admin	User
Login - Logout	Ναι	
Προσθήκη – Διαγραφή Χρήστη	Ναι	
Προσθήκη – Διαγραφή Τμήματος	Ναι	
Προσθήκη – Διαγραφή Υπαλλήλου	Ναι	
Προσθήκη – Διαγραφή Τύπου Υπαλλήλου Τεχνικό/Διοικητικό	Ναι	
Προσθήκη – Διαγραφή Διευθυντή	Ναι	
Προσθήκη – Διαγραφή Τηλ.Προγραμμάτων	Ναι	
Προσθήκη – Διαγραφή Οικογενειακών Μελών	Ναι	
Προσθήκη – Διαγραφή Πελάτη	Ναι	Ναι
Προσθήκη – Διαγραφή Τηλεφωνικού αριθμου	Ναι	Ναι
Προσθήκη – Διαγραφή Λογαριασμού	Ναι	Ναι
Προσθήκη – Διαγραφή Κεραίας	Ναι	Ναι
Προσθήκη – Διαγραφή Βλάβης	Ναι	Ναι
Προσθήκη – Διαγραφή Επιδιόρθωσης	Ναι	Ναι
Ετήσια Ανανέωση Παγιών - Επιδότηση	Ναι	
Στατιστικά	Ναι	

Πίνακας 1: Διαθέσιμες λειτουργίες συστήματος ανά ρόλο

8 ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ ΣΥΣΤΗΜΑΤΟΣ

8.1 Βασικά Χαρακτηριστικά Σελίδων Συστήματος

Ο παρακάτω πίνακας συγκεντρώνει όλα τα αρχεία του συστήματος μας κατηγοριοποιημένα στους αντίστοιχους φακέλους, περιγράφοντας τη λειτουργία του καθενός.

Φάκελοι αρχείων	Αρχεία	Περιγραφή
Mobiledb (κεντρικός φάκελος)	Index.php	Εδώ καθορίζεται ανάλογα α) με το ρόλο β) με το αν εισαι logged in ή όχι και γ) με την παράμετρο page στο url ποια σελίδα θα «φορτώσει».
admin	Users.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Χρηστών
	Departments.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Τμημάτων.
	Directors.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Διευθυντών.
	Employees.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Υπαλλήλων.
	EmployeesInfo.inc	Αντιστοιχεί στη διαχειριστική σελίδα των τύπων Υπαλλήλων (Διοικ./Τεχν.)
	FamilyMembers.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Οικογενειακών Μελών των υπαλλήλων
	Programs.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Προγραμμάτων
	MonthlyCostsupdate.inc	Αντιστοιχεί στη διαχειριστική σελίδα της ανανέωσης Παγίων.
	Statistics.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Στατιστικών.
whole	Antennas.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Κεραιών.
	Bills.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Λογαριασμών.
	Clients.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Πελατών.
	Index.inc	Αντιστοιχεί στο intro page του συστήματος.
	Damages.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Βλαβών.
	Fixes.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Επιδιορθώσεων.
	Telephonesx.inc	Αντιστοιχεί στη διαχειριστική σελίδα των Τηλεφωνικών Αριθμών.
Template	menu_admin.inc	Αντιστοιχεί στο τμήμα του Template που περιέχει τις διαχειριστικές επιλογές που έχει διαθέσιμες ο admin
	login.inc	Αντιστοιχεί στο τμήμα του Template που περιέχει τη φόρμα του Login-Logout.
	menu_employee.inc	Αντιστοιχεί στο τμήμα του Template που περιέχει τις διαχειριστικές επιλογές που έχει διαθέσιμες ο κάθε χρήστης.
	meta.inc	Αντιστοιχεί στο τμήμα του Template που περιέχει τις metadata πληροφορίες.
	menu_stats.inc	Αντιστοιχεί στο τμήμα του Template που περιέχει τις επιλογές Στατιστικών που έχει διαθέσιμες ο admin.

	Template.inc	Αντιστοιχεί στο σύνθεση-τρόπο εμφάνισης των επιμέρους τμημάτων του Template.
	Validate.inc	Αντιστοιχεί στο τμήμα του Template που κάνει την πιστοποίηση εισαγωγής σωστών username-password στη φόρμα του Login.

Πίνακας 2: Περιεχόμενα φακέλων-αρχείων

Χρησιμοποιήσαμε .inc αρχεία ώστε να είναι πιο εύκολη

A) η σύνθεση-εμφάνιση των επιμέρους μενού ανάλογα με το ρόλο (admin ή απλός χρήστης)

B) η διαχείριση των κάθε λειτουργικών σελίδων ανάλογα με το ρόλο.

Η βασική κατηγοριοποίηση κάθε αρχείου που αναφέρεται σε διαχειριστική λειτουργία είναι οι παρακάτω:

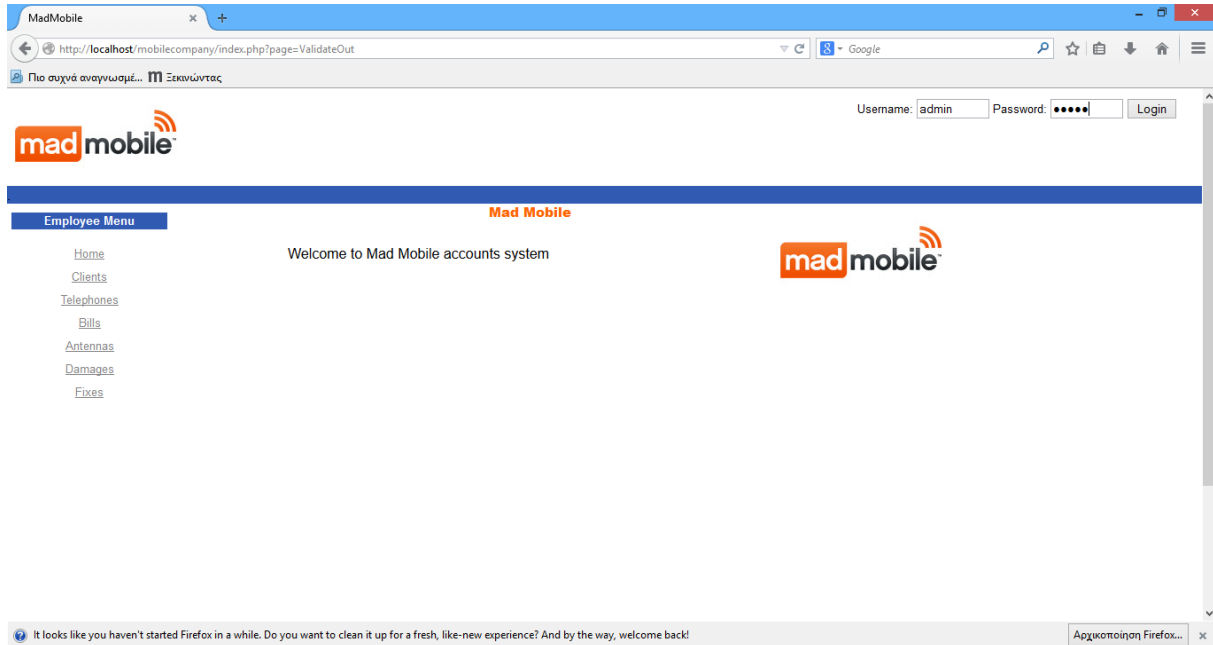
Actions	Flag	Fields check	Id	Result
add	true	All with data	-	Ολοκλήρωση Καταχώρησης
		Missing data	-	Εμφάνιση φόρμας με μήνυμα λάθους "ότι δεν έχουν συμπληρωθεί όλα τα πεδία"
	(false)	-	-	Εμφάνιση φόρμας καταχώρησης
delete	true	-	yes	Διαγραφή καταχώρησης με το συγκεκριμένο id
No action (else)	-	-	-	A) Εμφάνιση αρχικής οθόνης με σύνδεσμο για προσθήκη και B) λίστα όλων των ήδη καταχωρημένων εγγραφών με σύνδεσμο επεξεργασίας/διαγραφής για καθεμία

Πίνακας 3: Ενέργειες κώδικα κάθε αρχείου

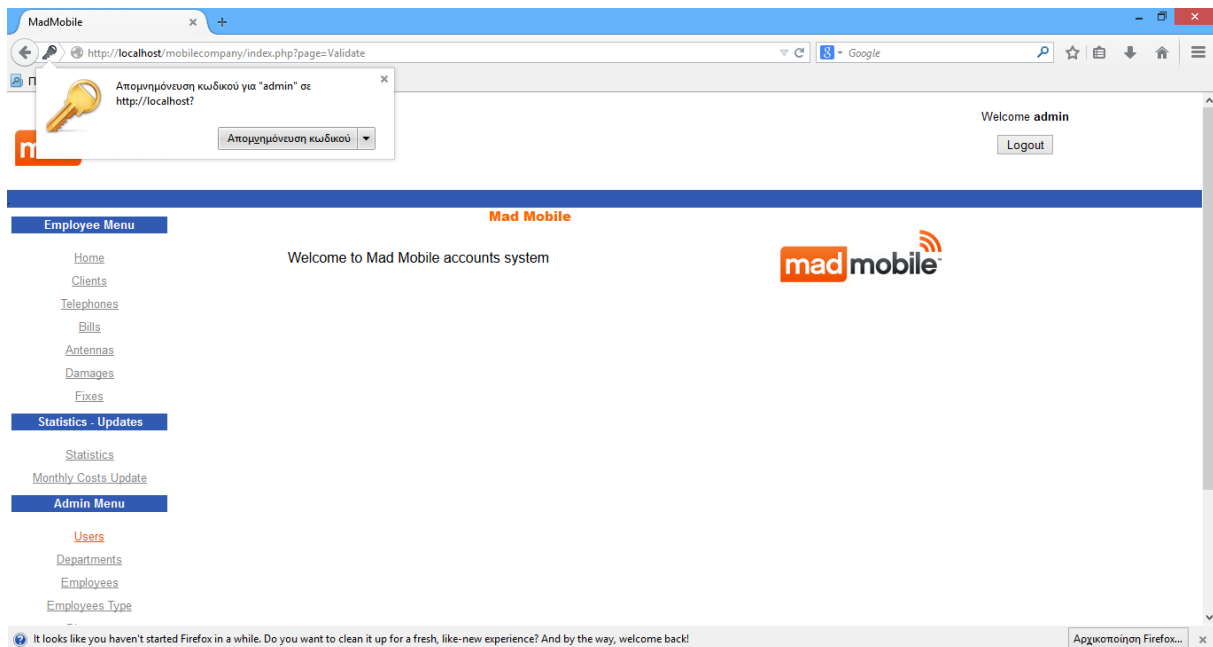
8.2 Login - Logout

Πραγματοποιείται έλεγχος σωστής εισαγωγής Ονόματος χρήστη – Κωδικού

Screenshots:



Εικόνα 9: Έλεγχος κωδικού-ονόματος χρήστη



Εικόνα 10: Εισαγωγή στο σύστημα - Μήνυμα καλωσορίσματος

SQL Query:

```
select * from users where UserUsername='\$user' and UserPassword='\$pass'
```

Εάν υπάρχει τέτοιος χρήστης τότε ενεργοποιείται η είσοδος στο σύστημα με καλωσόρισμα του χρήστη.

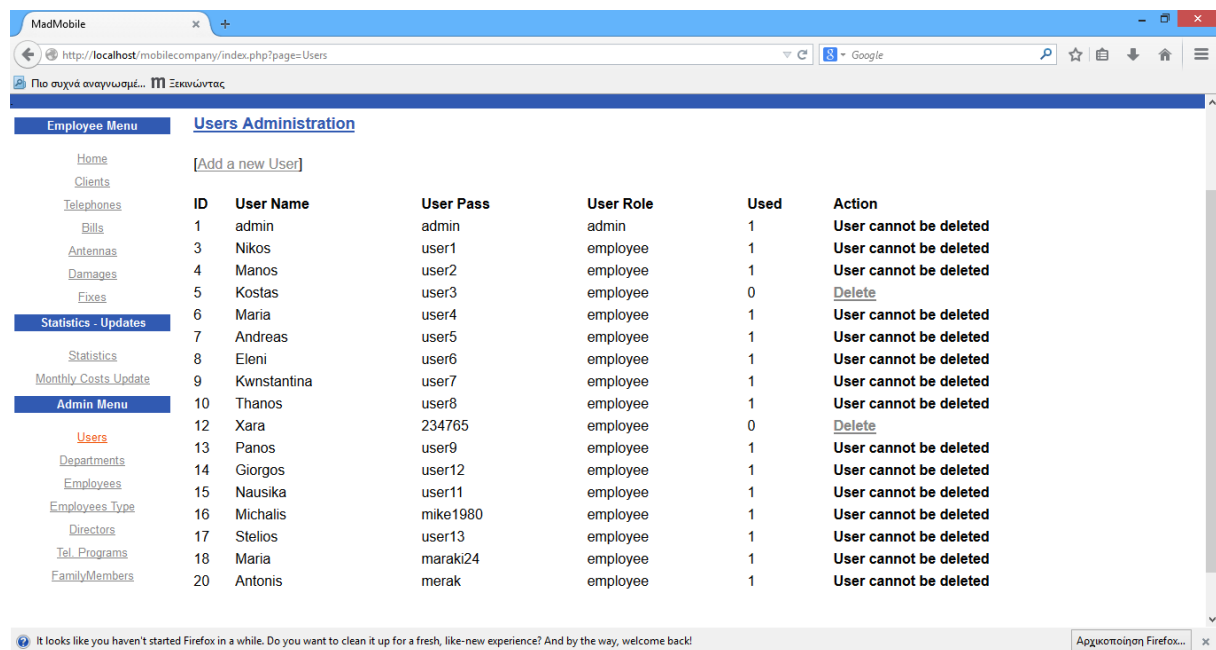
Επίσης ενεργοποιούνται οι php SESSION μεταβλητές τις οποίες θα χρειαστούμε κατά κύριο λόγο στο διαχωρισμό των λειτουργιών που πρέπει να έχει στη διάθεση του ο admin και ένας απλός user

```
$_SESSION['Logged']=1;  
$_SESSION['Name']=$_temp[2];  
$_SESSION['Role']=$_temp[1];  
$_SESSION['userID']=$_temp[0];
```

8.3 Χρήστες

8.3.1 Αρχική Οθόνη Χρηστών

Screenshots:



The screenshot shows a web browser window displaying the 'Users Administration' page. The page has a navigation menu on the left and a main content area with a table of users. The table has the following columns: ID, User Name, User Pass, User Role, Used, and Action. The data in the table is as follows:

ID	User Name	User Pass	User Role	Used	Action
1	admin	admin	admin	1	User cannot be deleted
3	Nikos	user1	employee	1	User cannot be deleted
4	Manos	user2	employee	1	User cannot be deleted
5	Kostas	user3	employee	0	Delete
6	Maria	user4	employee	1	User cannot be deleted
7	Andreas	user5	employee	1	User cannot be deleted
8	Eleni	user6	employee	1	User cannot be deleted
9	Kwnstantina	user7	employee	1	User cannot be deleted
10	Thanos	user8	employee	1	User cannot be deleted
12	Xara	234765	employee	0	Delete
13	Panos	user9	employee	1	User cannot be deleted
14	Giorgos	user12	employee	1	User cannot be deleted
15	Nausika	user11	employee	1	User cannot be deleted
16	Michalis	mike1980	employee	1	User cannot be deleted
17	Stelios	user13	employee	1	User cannot be deleted
18	Maria	maraki24	employee	1	User cannot be deleted
20	Antonis	merak	employee	1	User cannot be deleted

Εικόνα 11: Αρχική Οθόνη Χρηστών

Στην αρχική οθόνη των χρηστών εμφανίζεται η λίστα με τους υπάρχοντες χρήστες και δίπλα σε κάθε έναν, οι σύνδεσμοι Διαγραφής (Delete) για κάθε έναν από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος χρήστης, τότε εμφανίζεται μήνυμα μη ύπαρξης χρηστών.

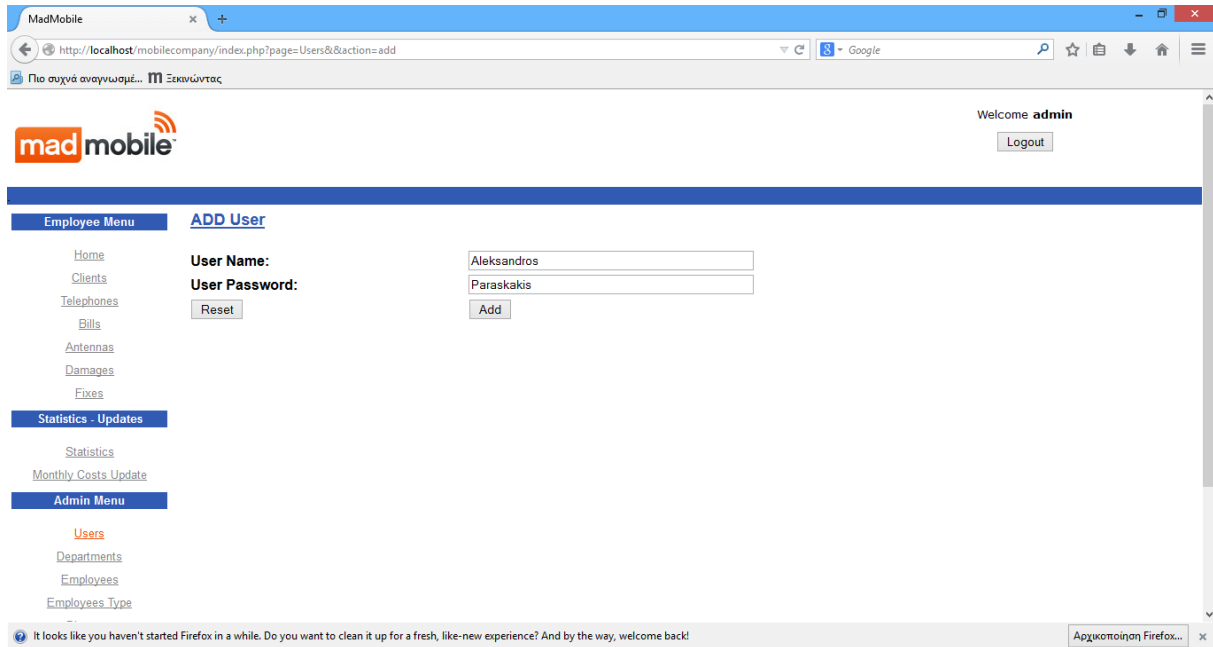
Επίσης, στο πάνω μέρος της αρχικής οθόνης χρηστών, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας χρήστη.

Ένας λογαριασμός χρήστη που έχει γίνει ανάθεση σε έναν υπάλληλο, δεν μπορεί να διαγραφεί.

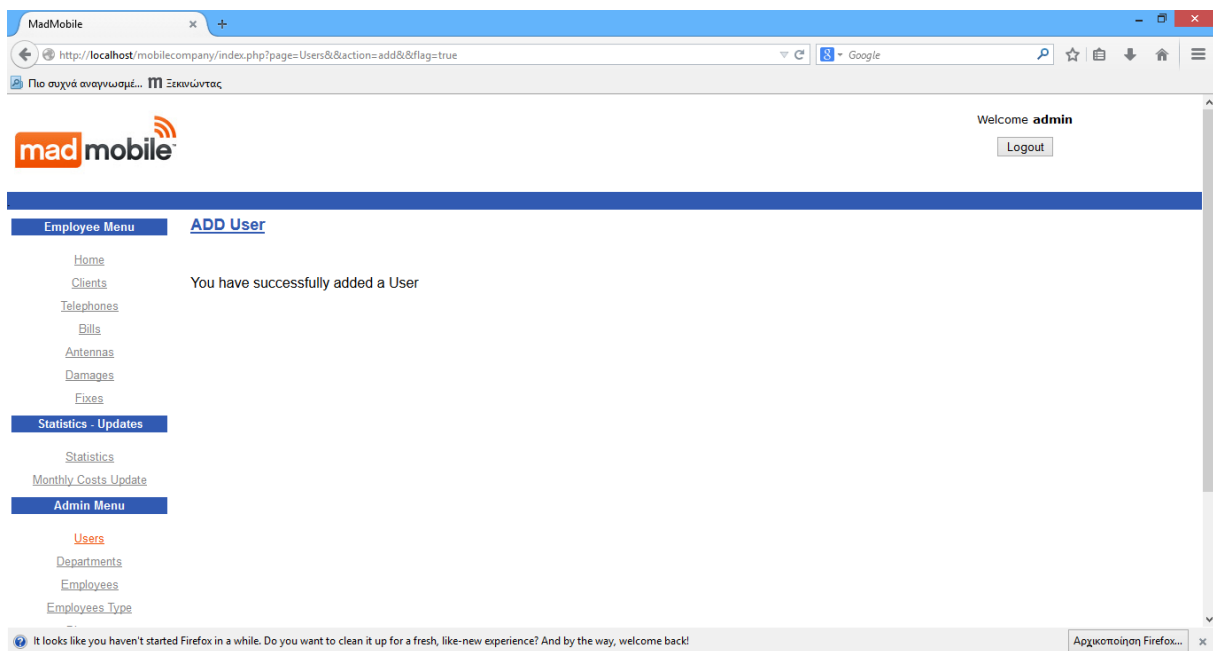
8.3.2 Δημιουργία Χρήστη

Screenshots:

Στην οθόνη Δημιουργίας Χρήστη, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας χρήστης.



Εικόνα 12: Φόρμα Προσθήκης Νέου Χρήστη



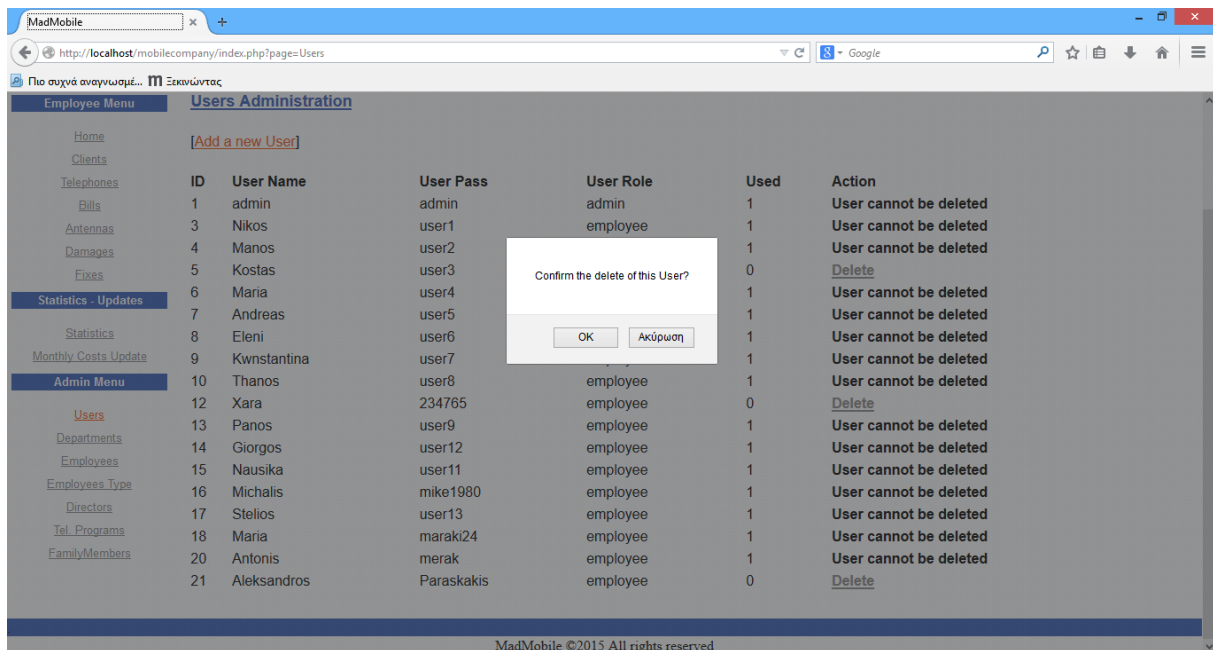
Εικόνα 13: Ολοκλήρωση προσθήκης Νέου Χρήστη

SQL query:

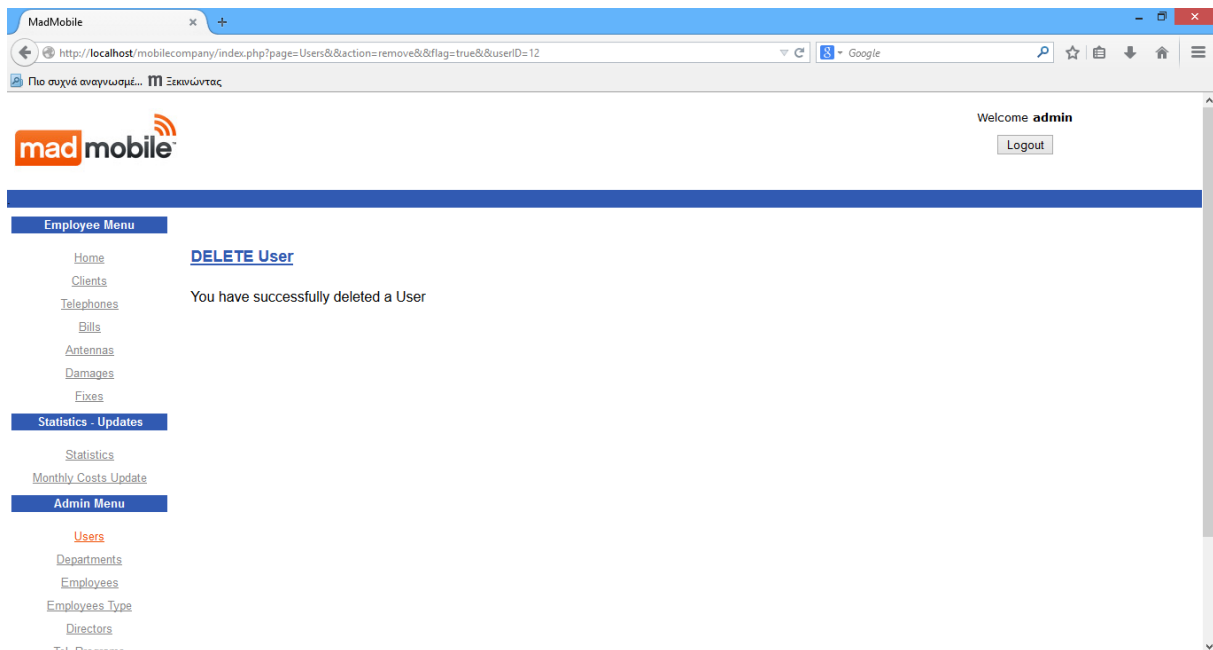
```
insert into users (userRole, username, password, is_active)
values ('employee', '$username', '$password', 0)
```


8.3.3 Διαγραφή Χρήστη

Screenshots:



Εικόνα 14: Διαγραφή Χρήστη 1



Εικόνα 15: Διαγραφή Χρήστη 2

SQL query:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου χρήστη, περνώντας το id του ως παράμετρο.

```
delete from users where userID=$${strtsID}
```

8.4 Τμήματα

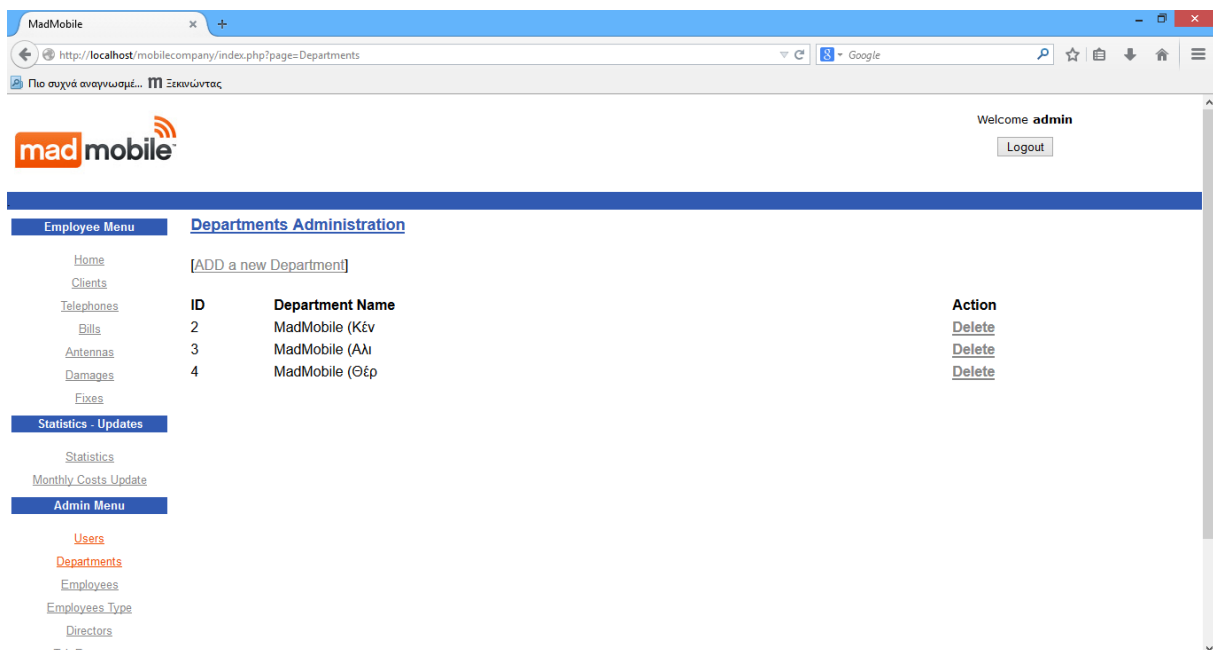
8.4.1 Αρχική Οθόνη Τμημάτων

Screenshots:

Στην αρχική οθόνη των τμημάτων εμφανίζεται η λίστα με τα υπάρχοντα τμήματα και δίπλα σε κάθε ένα, οι σύνδεσμοι και Διαγραφής (Delete) για κάθε ένα από αυτά.

Εάν δεν έχει δημιουργηθεί κάποιο τμήμα, τότε εμφανίζεται μήνυμα μη ύπαρξης τμημάτων.

Επίσης, στο πάνω μέρος της αρχικής οθόνης τμημάτων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας τμήματος.

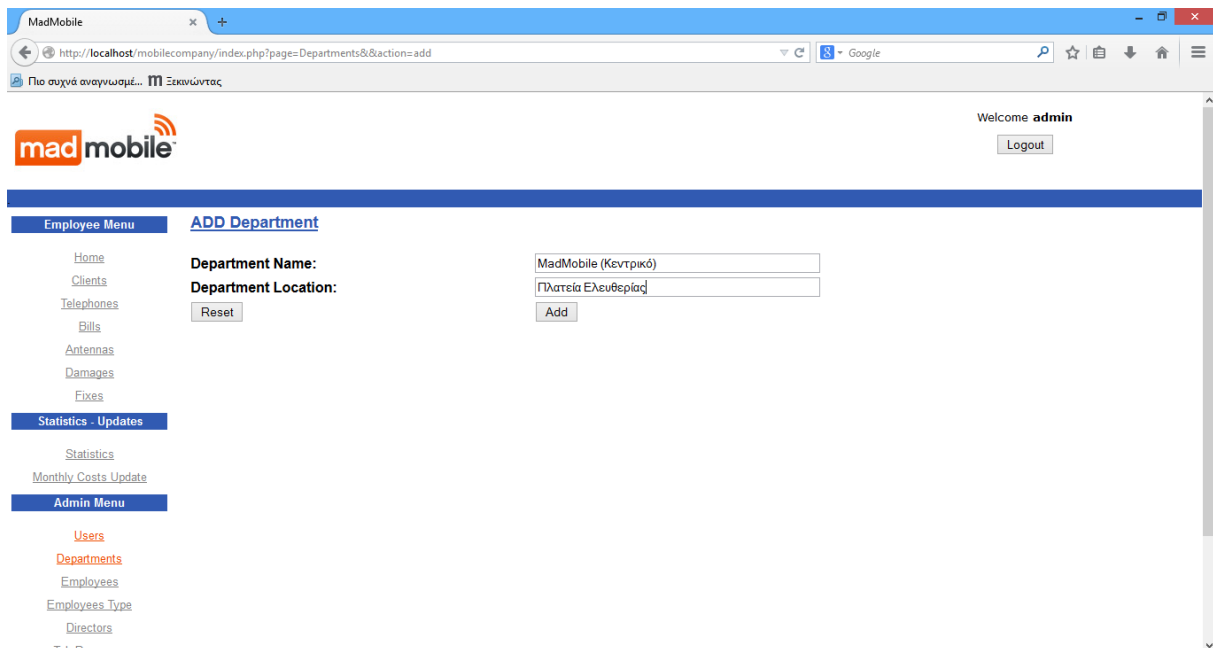


Εικόνα 16: Αρχική Οθόνη Τμημάτων

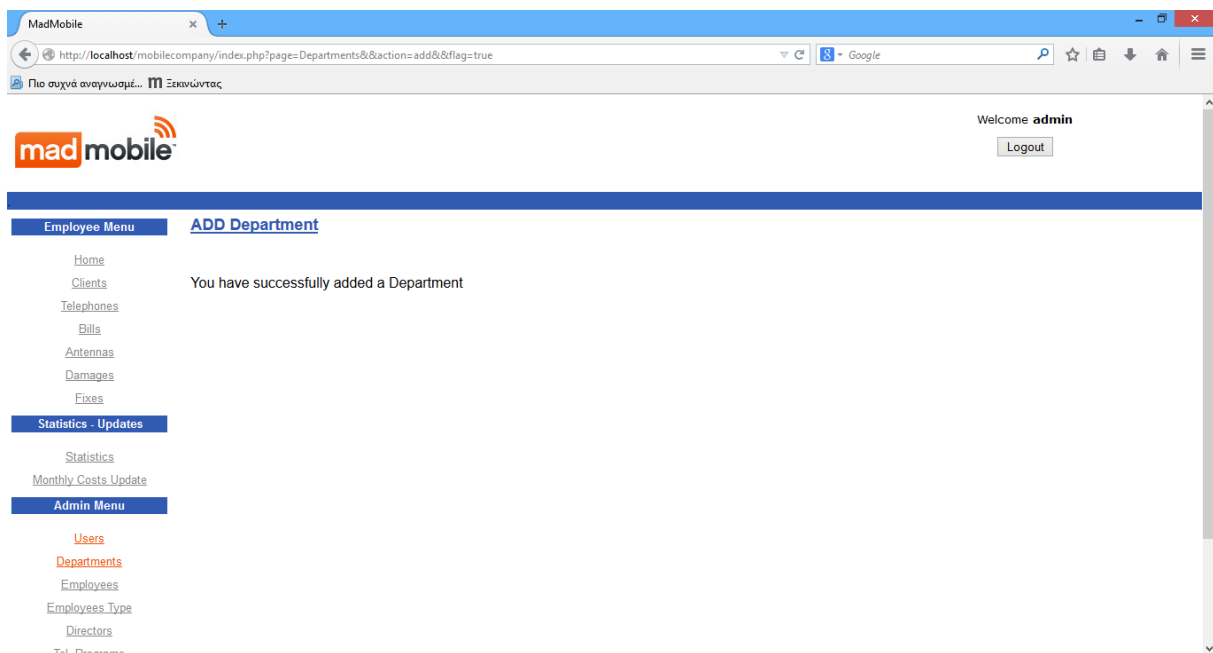
8.4.2 Δημιουργία Τμήματος

Screenshots:

Στην οθόνη Δημιουργίας Τμήματος, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένα τμήμα.



Εικόνα 17: Φόρμα Προσθήκης Τμήματος



Εικόνα 18: Ολοκλήρωση προσθήκης Νέου Τμήματος

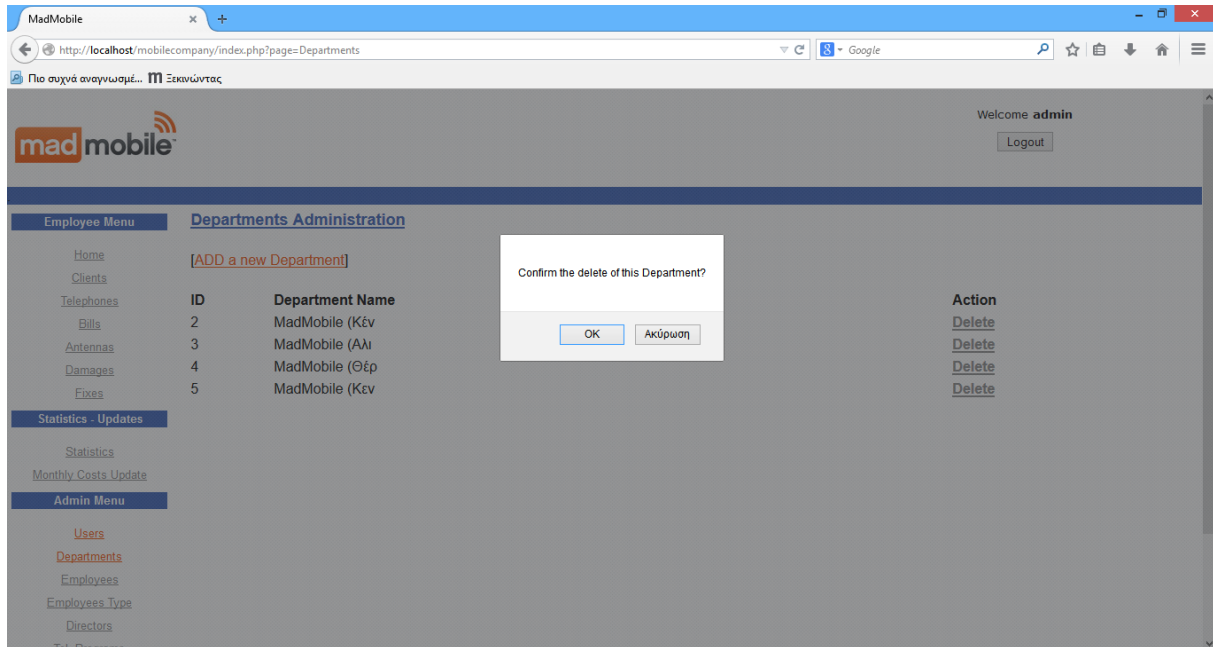
SQL query:

insert into departments (dname, dlocation, dirid) values ('\$dname', '\$dlocation', '')

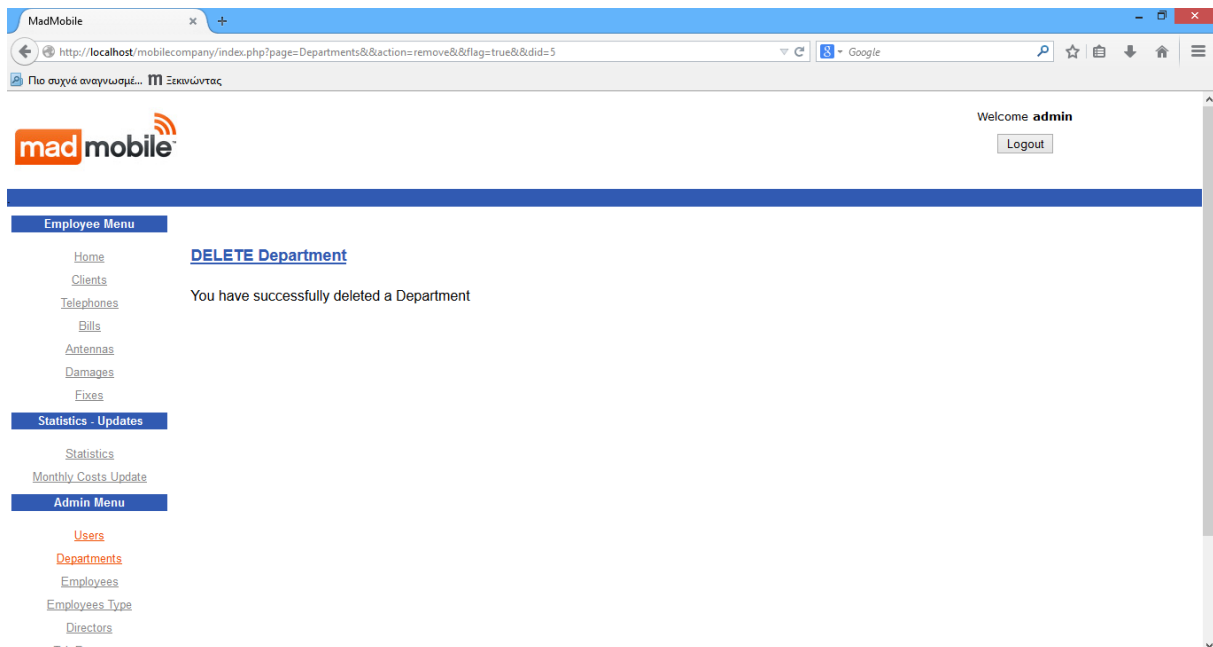
8.4.3 Διαγραφή Τμήματος

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου τμήματος, περνώντας το id του ως παράμετρο.



Εικόνα 19: Διαγραφή Τμήματος 1



Εικόνα 20: Διαγραφή Τμήματος 2

SQL query:

delete from departments where did=\$StrtsID

8.5 Υπάλληλοι

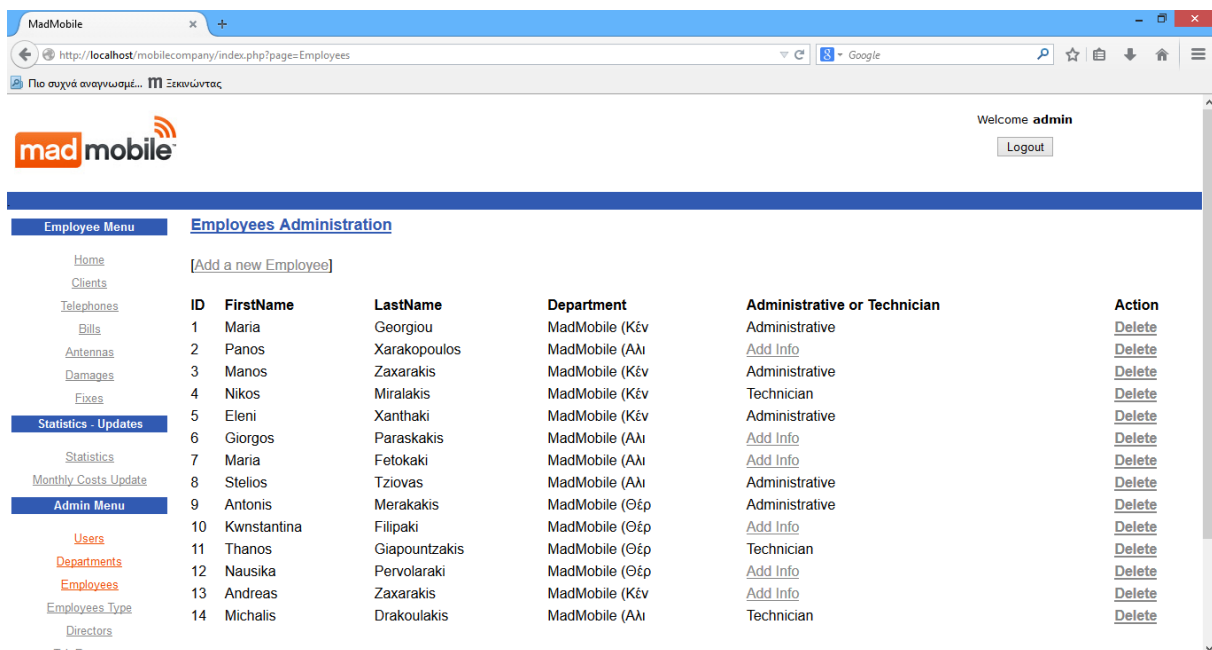
8.5.1 Αρχική Οθόνη Υπαλλήλων

Screenshots:

Στην αρχική οθόνη των υπαλλήλων εμφανίζεται η λίστα με τους υπάρχοντες υπάλληλους και δίπλα σε κάθε ένα, οι σύνδεσμοι Επεξεργασίας (Edit) και Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος υπάλληλος, τότε εμφανίζεται μήνυμα μη ύπαρξης υπαλλήλων.

Επίσης, στο πάνω μέρος της αρχικής οθόνης υπαλλήλων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας υπαλλήλου.



ID	FirstName	LastName	Department	Administrative or Technician	Action
1	Maria	Georgiou	MadMobile (Κέν	Administrative	Delete
2	Panos	Xarakopoulos	MadMobile (Αλι	Add Info	Delete
3	Manos	Zaxarakis	MadMobile (Κέν	Administrative	Delete
4	Nikos	Miralakis	MadMobile (Κέν	Technician	Delete
5	Eleni	Xanthaki	MadMobile (Κέν	Administrative	Delete
6	Giorgos	Paraskakis	MadMobile (Αλι	Add Info	Delete
7	Maria	Fetokaki	MadMobile (Αλι	Add Info	Delete
8	Stelios	Tziovas	MadMobile (Αλι	Administrative	Delete
9	Antonis	Merakakis	MadMobile (Θέρ	Administrative	Delete
10	Kwnstantina	Filipaki	MadMobile (Θέρ	Add Info	Delete
11	Thanos	Giapountzakis	MadMobile (Θέρ	Technician	Delete
12	Nausika	Pervolaraki	MadMobile (Θέρ	Add Info	Delete
13	Andreas	Zaxarakis	MadMobile (Κέν	Add Info	Delete
14	Michalis	Drakoulakis	MadMobile (Αλι	Technician	Delete

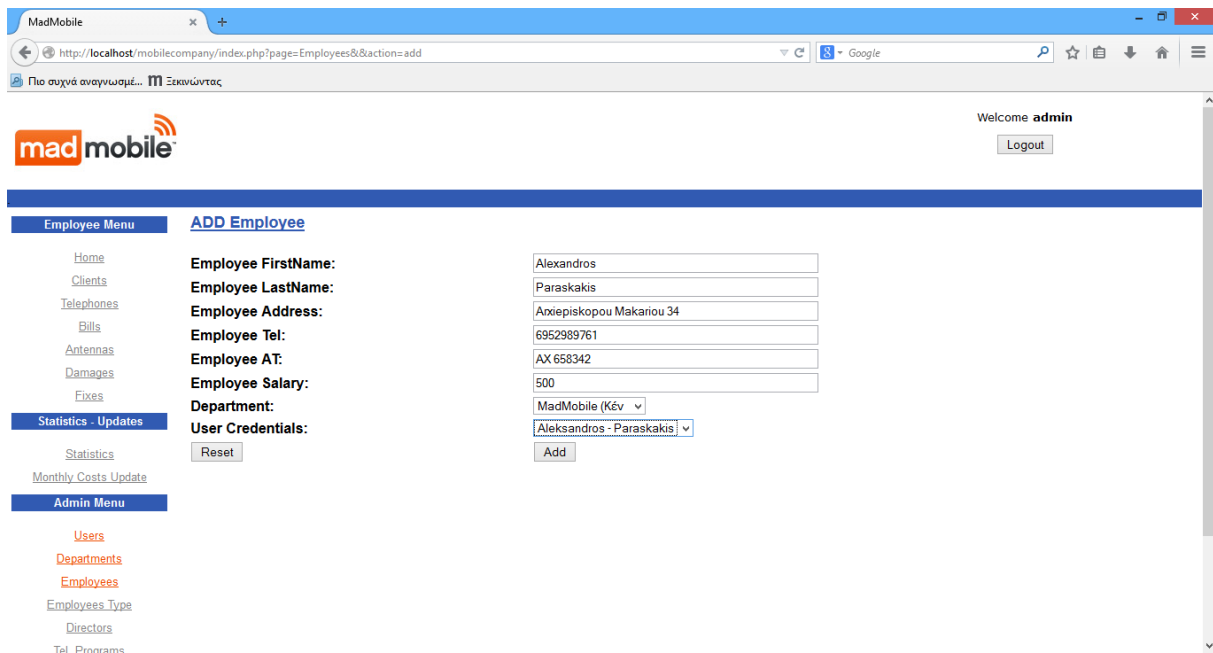
Εικόνα 21: Αρχική Οθόνη Υπαλλήλων

8.5.2 Δημιουργία Υπαλλήλου

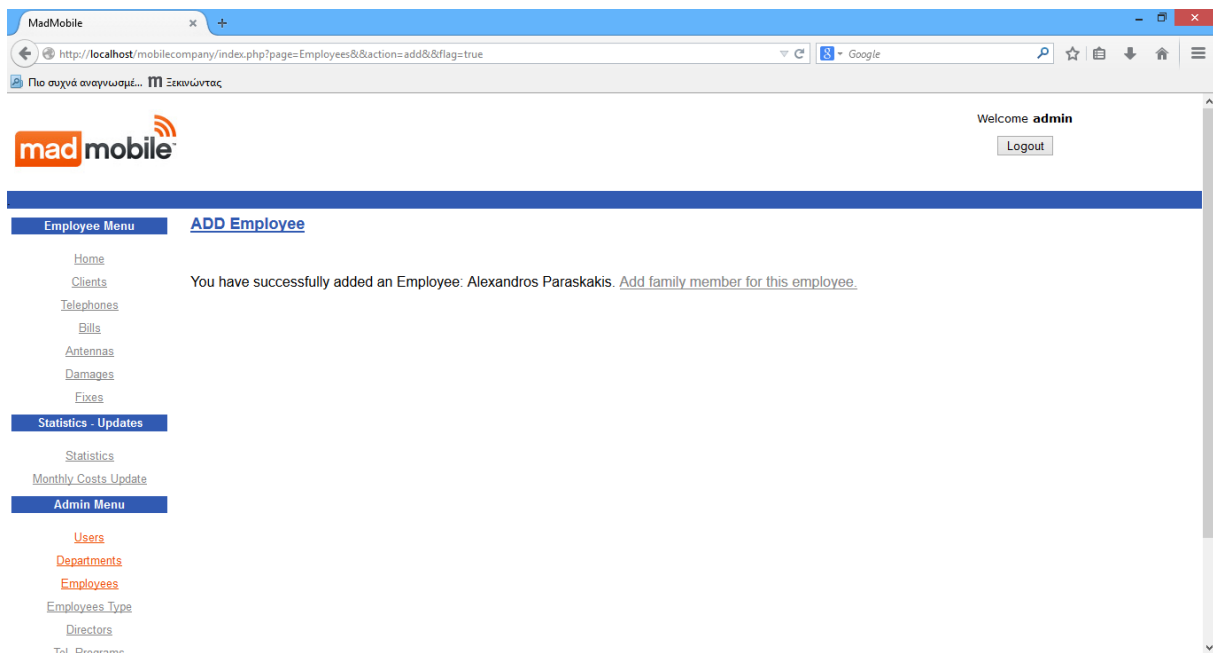
Screenshots:

Στην οθόνη Δημιουργίας Υπαλλήλου, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας χρήστης.

- Η φόρμα δημιουργίας υπαλλήλου έχει λίστα επιλογής με **ΟΛΑ** τα διαθέσιμα τμήματα.
- Η φόρμα δημιουργίας υπαλλήλου έχει λίστα επιλογής με τους διαθέσιμους κωδικούς που **ΔΕΝ** έχουν δοθεί σε άλλο υπάλληλο.



Εικόνα 22: Φόρμα Προσθήκης Νέου Υπαλλήλου



Εικόνα 23: Ολοκλήρωση προσθήκης Νέου Υπαλλήλου

SQL query:

Δύο queries εκτελούνται κατά τη δημιουργία ενός υπαλλήλου:

A) Η νέα εγγραφή στον πίνακα employees

insert into employees (firstname, elastname, eaddress, etel, eAT, esalary, did, userID)

values ('\$firstname', '\$elastname', '\$eaddress', '\$etel', '\$eAT', \$salary, \$did, \$userID)

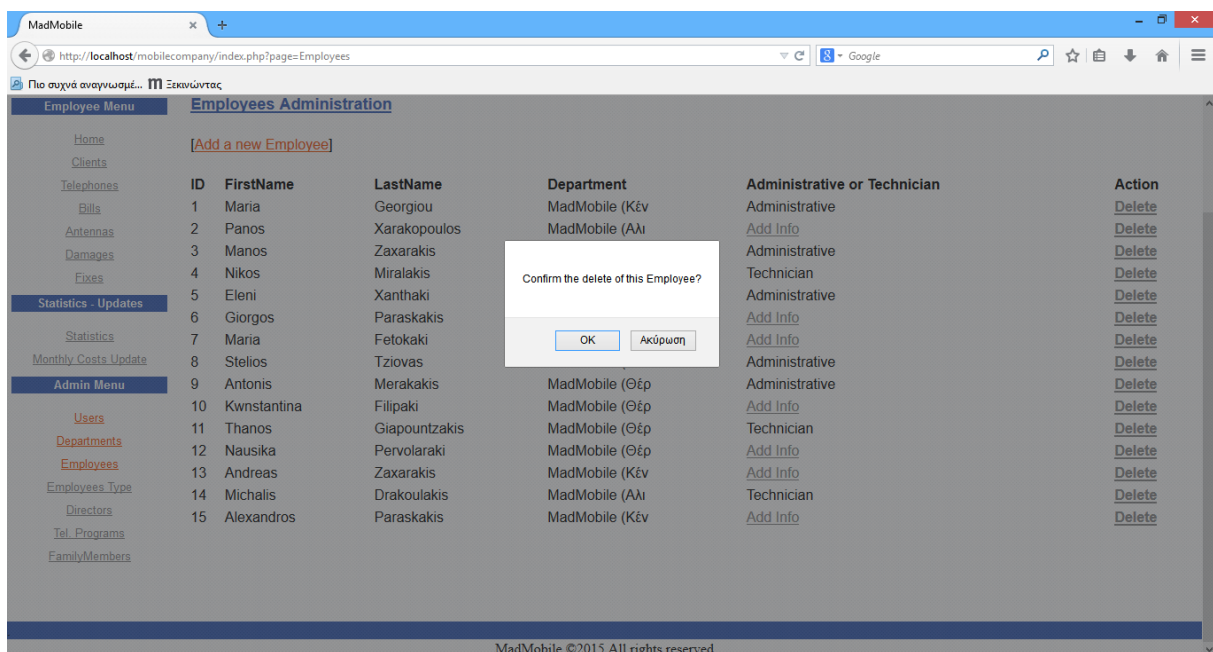
B) Η ανανέωση στον πίνακα users, ώστε ο συγκεκριμένος λογαριασμός χρήστη που αποδόθηκε στον υπάλληλο που προσθέσαμε μόλις, να μην είναι διαθέσιμος από εδώ και στο εξής.

update users set usedornot=1 where userID=\$userID

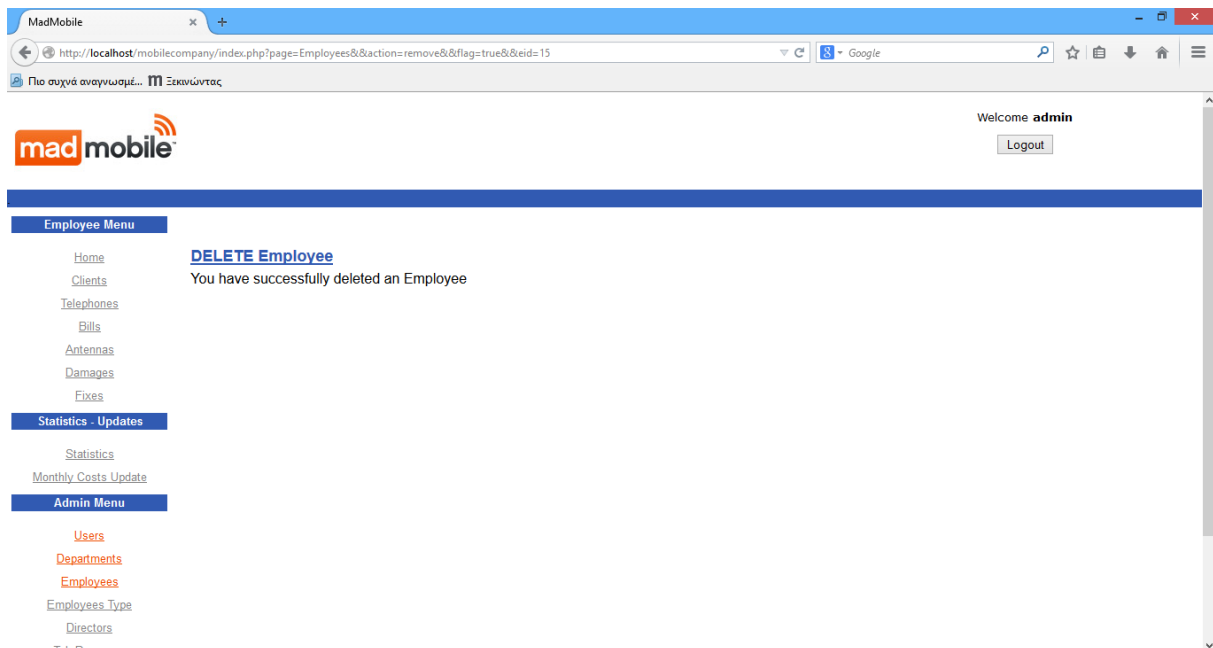
8.5.3 Διαγραφή Υπαλλήλου

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου υπαλλήλου, περνώντας το id του ως παράμετρο.



Εικόνα 24: Διαγραφή Υπαλλήλου 1



Εικόνα 25: Διαγραφή Υπαλλήλου 2

SQL query:

delete from employees where eid=5StrtsID

8.6 Τύποι Υπαλλήλων (Διοικ./Τεχν.)

8.6.1 Αρχική Οθόνη Τύπων υπαλλήλων

Screenshots:

Στην αρχική οθόνη των Τύπων Υπαλλήλων εμφανίζεται η λίστα με τα υπάρχοντες υπαλλήλους και δίπλα σε κάθε ένα, οι σύνδεσμοι και Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος υπάλληλος, τότε εμφανίζεται μήνυμα μη ύπαρξης υπαλλήλων.

Επίσης, στο πάνω μέρος της αρχικής οθόνης Τύπων Υπαλλήλων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας τμήματος.

The screenshot shows the 'Employees Info Administration' page. On the left is a navigation menu with categories like 'Employee Menu', 'Statistics - Updates', and 'Admin Menu'. The main content area has a sub-menu 'Employees Info Administration' and a link to 'Add a new Employee Type'. Below this is a table of employees:

Employee ID	Name	Administrative ID	Degree Title	Degree Univ.	Action
1	Maria Georgiou	5	Business Administration	TEI Kritis	Delete
3	Manos Zaxarakis	1	Computer Technician	TEI Kritis	Delete
5	Eleni Xanthaki	4	Business Administration	TEI Kritis	Delete
8	Stelios Tziovas	2	Business Administration	University of Atens	Delete
9	Antonis Merakakis	3	Business Administration	University of Atens	Delete

Below the employee table is another table for technicians:

Employee ID	Name	Technician ID	Specialty	Years	Action
4	Nikos Miralakis	1	Phantom Antenna	12	Delete
11	Thanos Giapountzakis	2	Technician Anntenas	8	Delete
14	Michalis Drakoulakis	3	Technician Anntenas	10	Delete

Εικόνα 26: Αρχική Οθόνη Τύπων Υπαλλήλων

8.6.2 Δημιουργία Τύπου Υπαλλήλων

Screenshots:

Στην οθόνη Δημιουργίας Τύπου Υπαλλήλων εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένα Τύπος Υπαλλήλου.

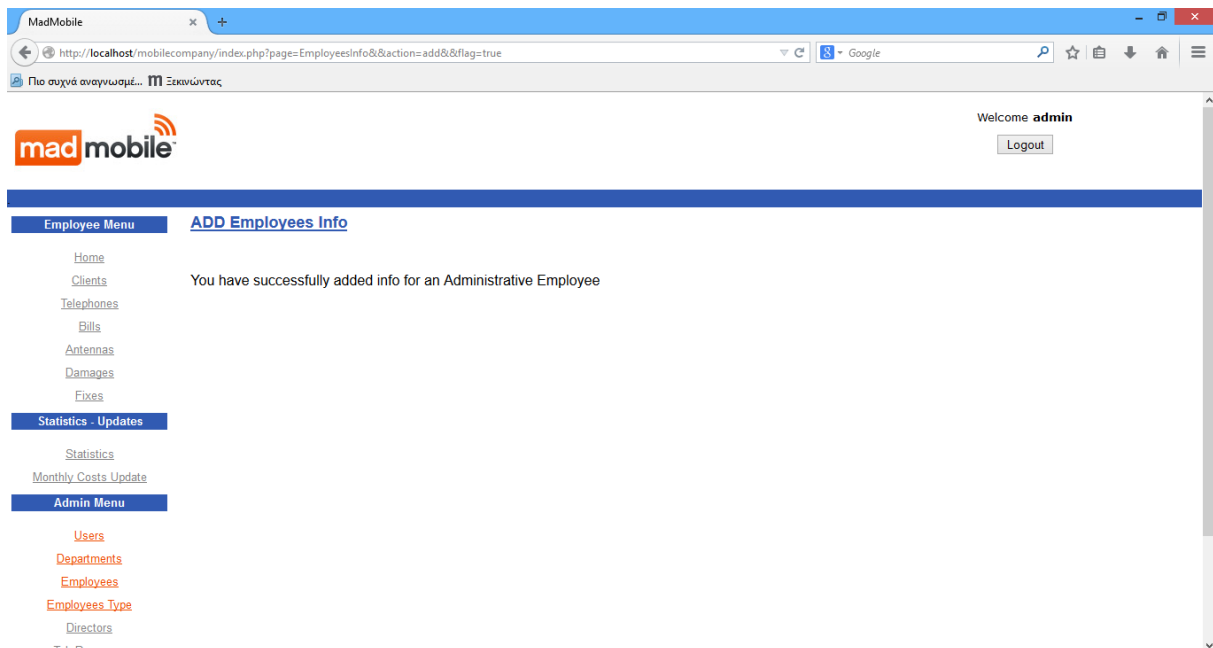
- Η φόρμα δημιουργίας τύπου υπαλλήλου έχει λίστα επιλογής με ΟΛΟΥΣ τους διαθέσιμους υπαλλήλους.

The screenshot shows the 'ADD Employees Info' form. The form is divided into two sections based on the 'Employee Type' selection. The 'Employee' field is a dropdown menu with 'Andreas Zaxarakis' selected. The 'Employee Type' is a dropdown menu with 'Administrative' selected. Below this, there are instructions: 'If you choose Administrative option, fill the following info.' and 'If you choose Technician option, fill the following info.' The form includes the following fields:

- Administrative Degree Title:** Business Administration
- Administrative Degree University:** Tei of Athens
- Administrative Degree Date:** 05/06/2000
- Technician Specialty:** (empty field)
- Technician Years of Profession:** (empty field)

At the bottom of the form are 'Reset' and 'Add' buttons.

Εικόνα 27: Φόρμα Προσθήκης Τύπου Υπαλλήλου



Εικόνα 28:Ολοκλήρωση προσθήκης Νέου Τύπου Υπαλλήλου

SQL query:

```
insert into administratives (adegree_title, auniversity, adegree_date, eid)
```

```
values ('$adegree_title', '$auniversity', '$adegree_date', $StrtsID)
```

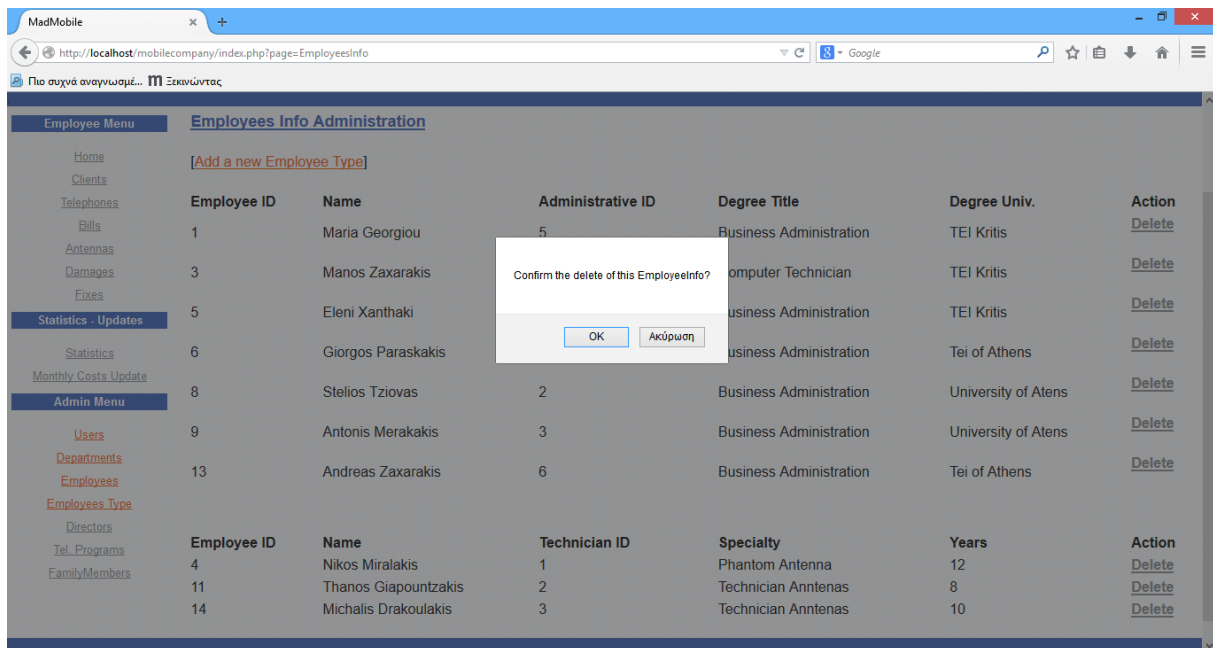
ή

```
insert into technicians (tspecialty, tyears, eid) values ('$tspecialty', $tyears, $StrtsID)
```

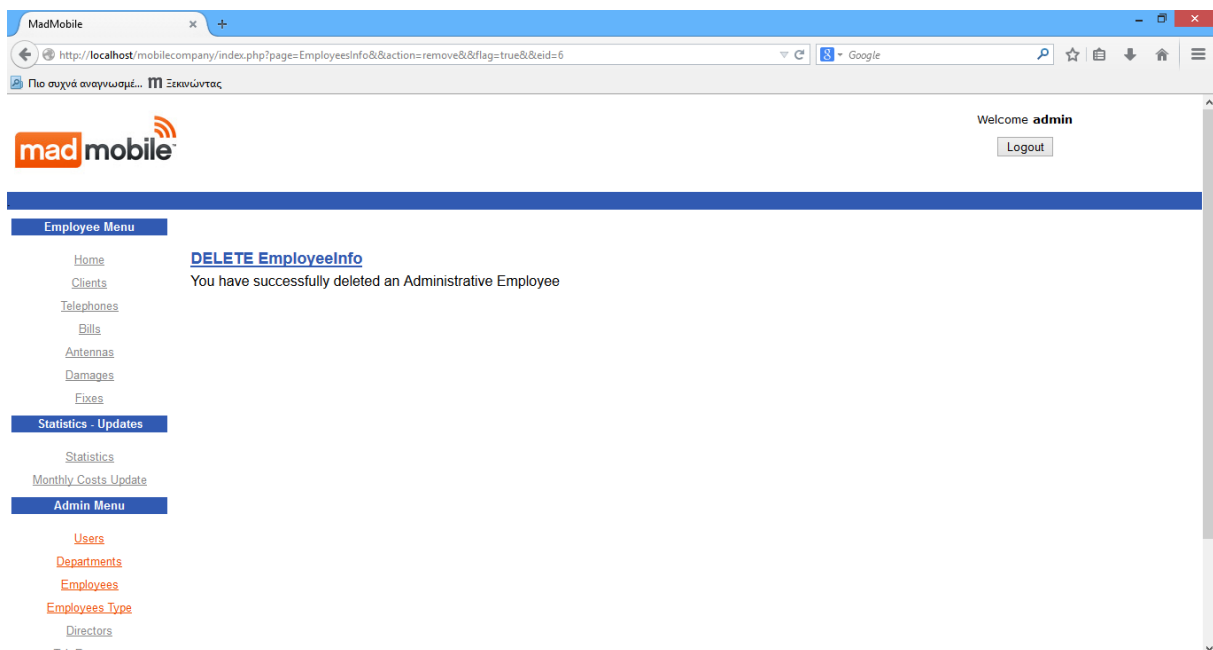
8.6.3 Διαγραφή Τύπου Υπαλλήλου

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου τύπου υπαλλήλου, περνώντας το id του ως παράμετρο.



Εικόνα 29: Διαγραφή Τύπου Υπαλλήλου 1



Εικόνα 30: Διαγραφή Τύπου Υπαλλήλου 2

SQL query:

delete from administratives where eid=\$\$StrtsID

ή

delete from technicians where eid=\$\$StrtsID

8.7 Διευθυντές

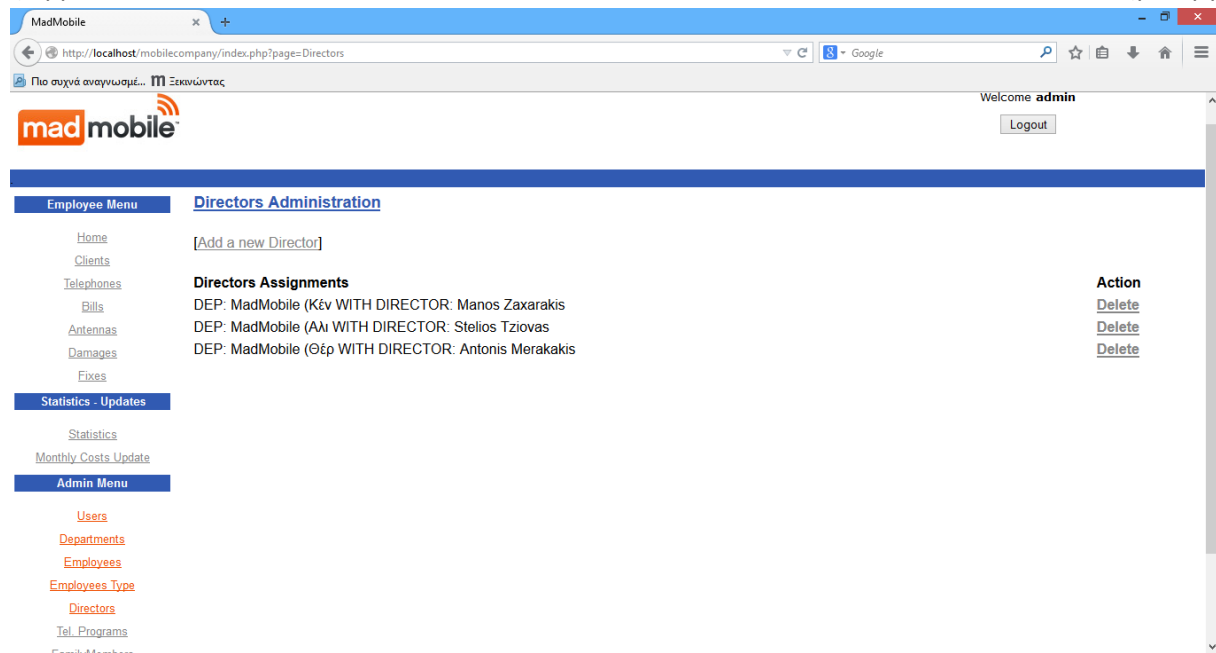
8.7.1 Αρχική Οθόνη Διευθυντών

Screenshots:

Στην αρχική οθόνη των διευθυντών εμφανίζεται η λίστα με τους υπάρχοντα διευθυντές και δίπλα σε κάθε ένα, οι σύνδεσμοι Επεξεργασίας (Edit) και Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος διευθυντής, τότε εμφανίζεται μήνυμα μη ύπαρξης διευθυντών.

Επίσης, στο πάνω μέρος της αρχικής οθόνης διευθυντών, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργ



ίας διευθυντή.

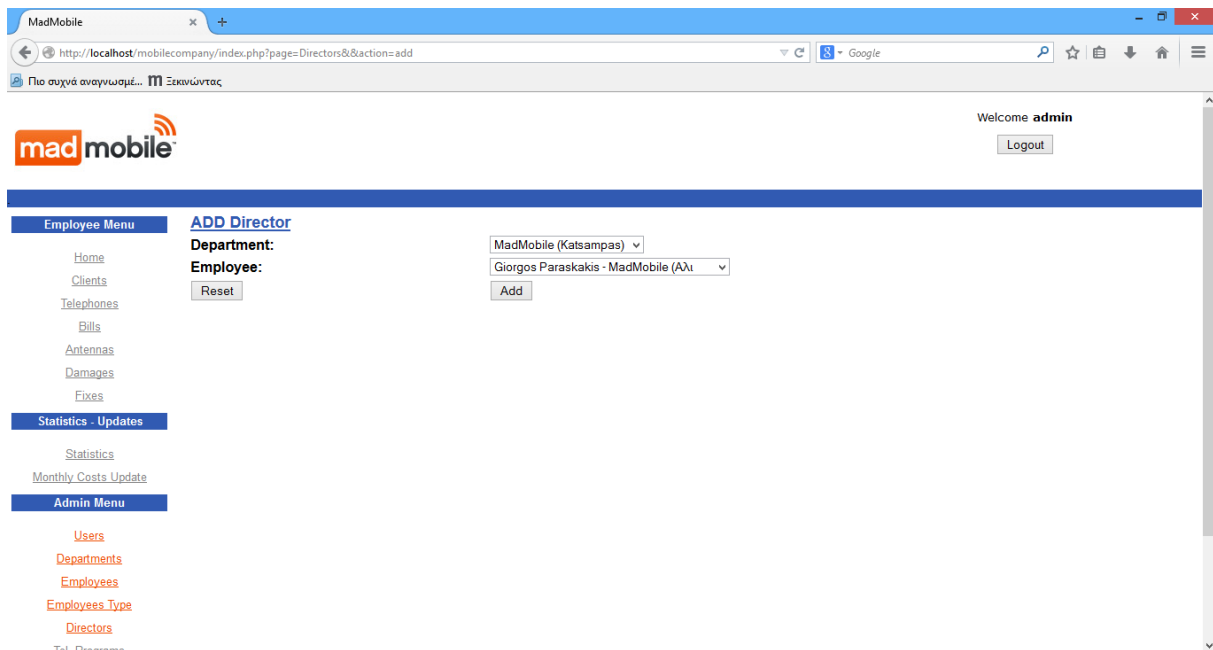
Εικόνα 31: Αρχική Οθόνη Διευθυντών

8.7.2 Δημιουργία Διευθυντή

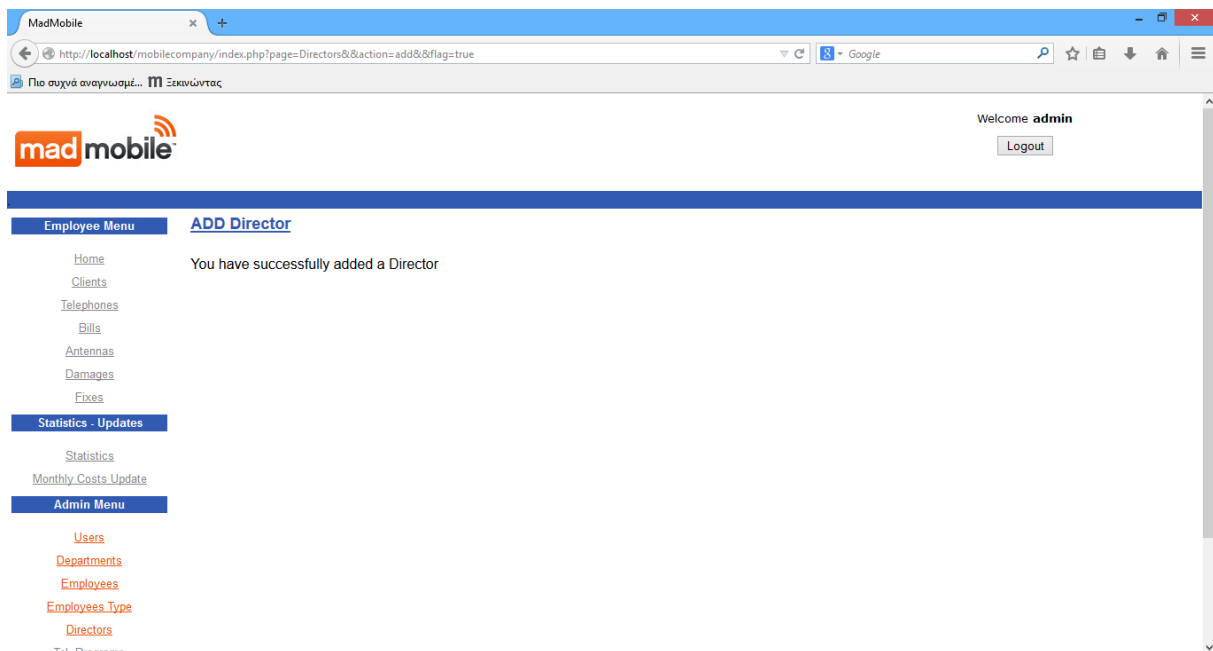
Screenshots:

Στην οθόνη Δημιουργίας Διευθυντή, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας διευθυντής.

- Η φόρμα δημιουργίας διευθυντή έχει λίστα επιλογής με ΟΛΑ τα διαθέσιμα τμήματα.
- Η φόρμα δημιουργίας διευθυντή έχει λίστα επιλογής με ΟΛΟΥΣ τους διαθέσιμους υπαλλήλους.



Εικόνα 32: Φόρμα Προσθήκης Νέου Διευθυντή



Εικόνα 33: Ολοκλήρωση προσθήκης Νέου Διευθυντή

SQL query:

//ΠΡΟΣΘΗΚΗ ΔΙΕΥΘΥΝΤΗ

```
insert into directors (did, eid) values ( $did, $eid )
```

//ΕΠΙΛΟΓΗ ΤΟΥ ΤΕΛΕΥΤΑΙΟΥ ID ΠΟΥ ΜΠΗΚΕ

```
SELECT LAST_INSERT_ID()
```

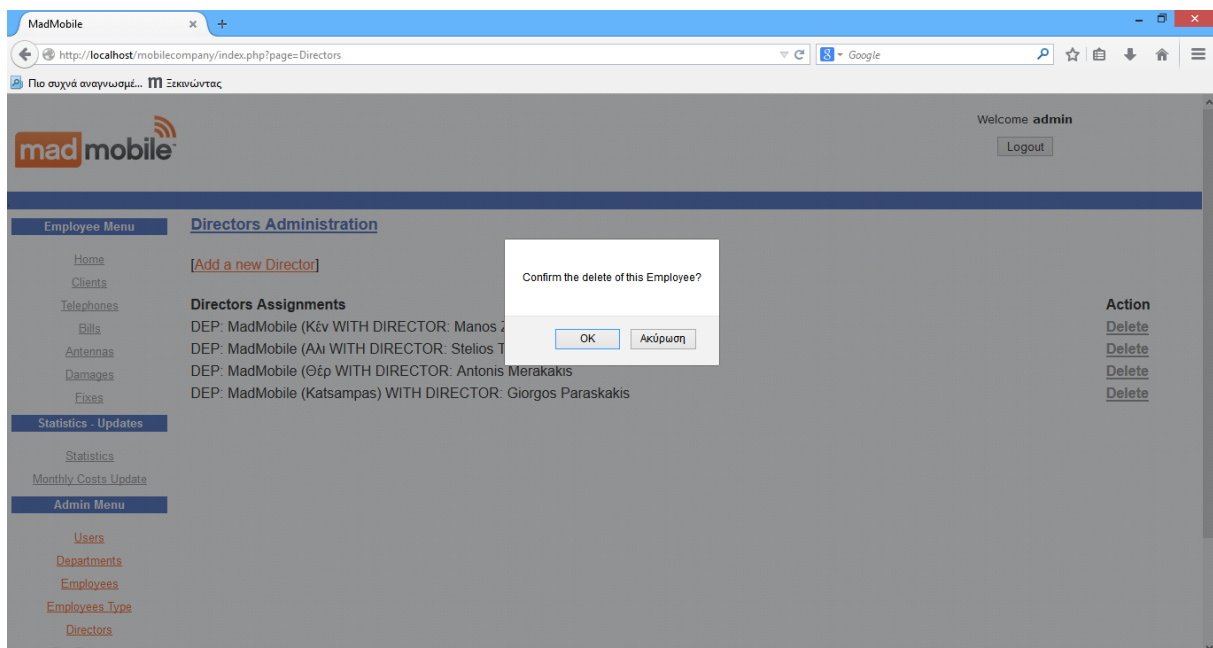
//ΑΝΑΝΕΩΣΗ ΣΥΓΚΕΚΡΙΜΕΝΟΥ ΤΜΗΜΑΤΟΣ ΟΤΙ ΕΧΕΙ ΔΙΕΥΘΥΝΤΗ

```
update departments set dirid=$last_id where did=$did
```

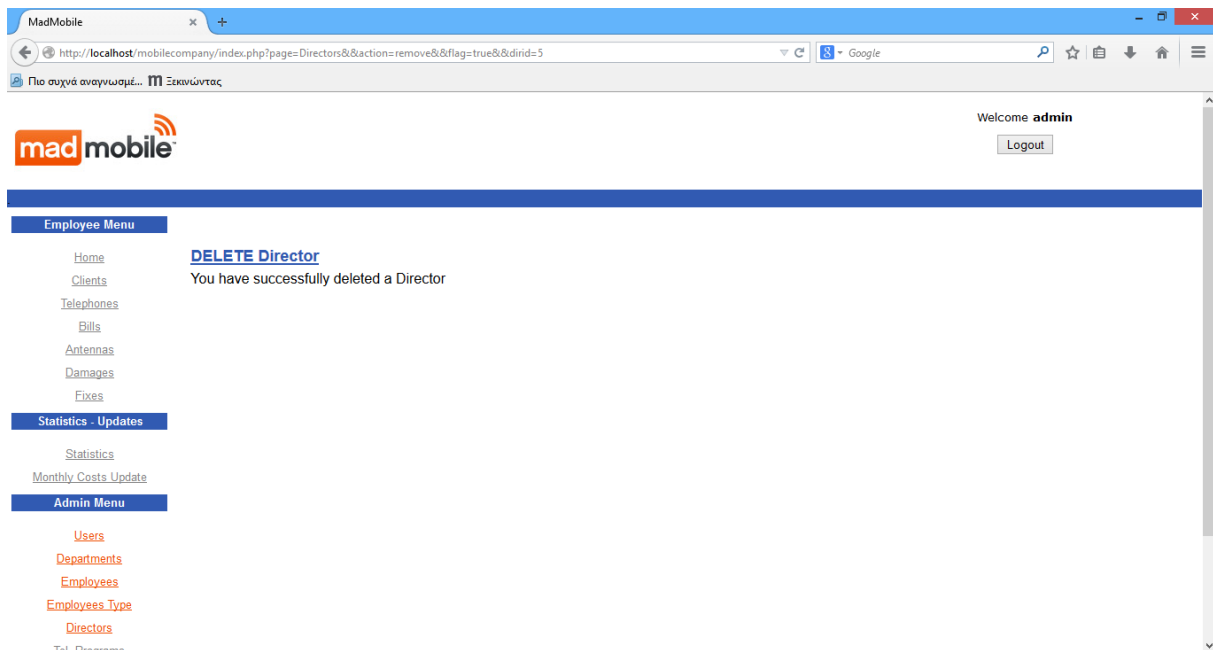
8.7.3 Διαγραφή Διευθυντή

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου διευθυντή, περνώντας το id του ως παράμετρο.



Εικόνα 34: Διαγραφή Διευθυντή 1



Εικόνα 35: Διαγραφή Διευθυντή 2

SQL query:

//ΔΙΑΓΡΑΦΗ ΔΙΕΥΘΥΝΤΗ

```
delete from directors where dirid=$StrtsID
```

//ΑΝΑΝΕΩΣΗ ΤΜΗΜΑΤΟΣ ΟΤΙ ΔΕΝ ΕΧΕΙ ΠΛΕΟΝ ΔΙΕΥΘΥΝΤΗ

```
update departments set dirid="" where dirid=$StrtsID
```

8.8 Τηλεφωνικά Προγράμματα

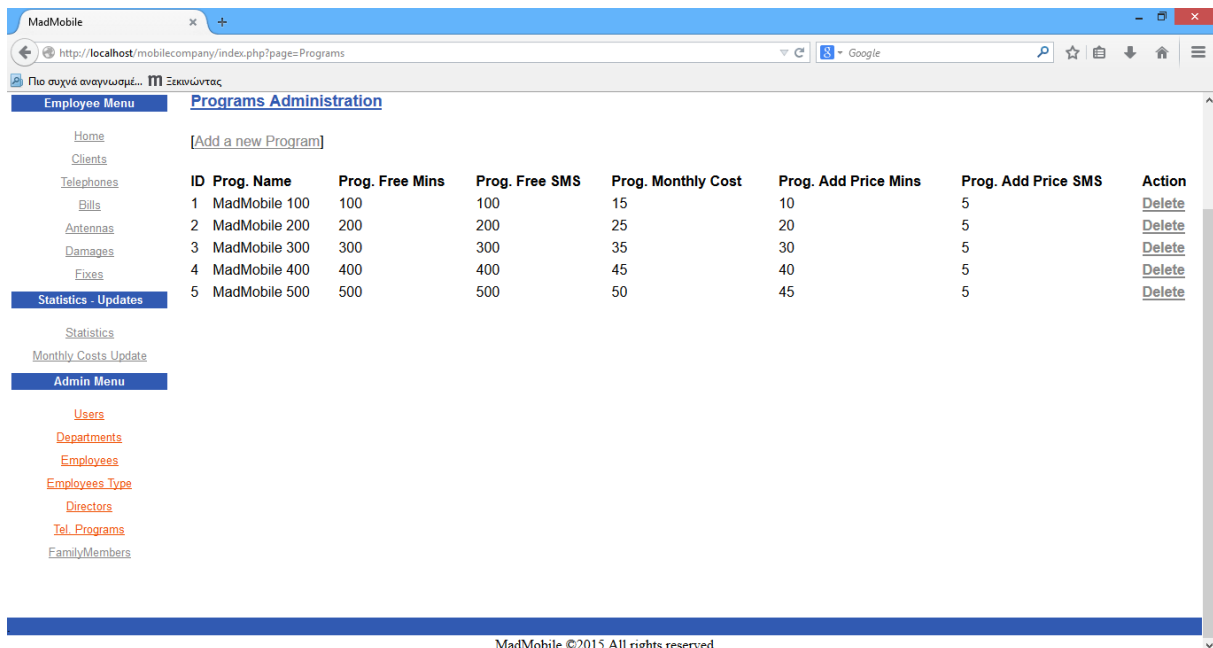
8.8.1 Αρχική Οθόνη Τηλεφωνικών Προγραμμάτων

Screenshots:

Στην αρχική οθόνη των τηλ.προγραμμάτων εμφανίζεται η λίστα με τα υπάρχοντα τηλ.πρόγραμματα και δίπλα σε κάθε ένα, οι σύνδεσμοι Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος τηλ.πρόγραμμα, τότε εμφανίζεται μήνυμα μη ύπαρξης τηλ.προγράμματος.

Επίσης, στο πάνω μέρος της αρχικής οθόνης τηλ.προγραμμάτων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας τηλ.προγράμματος.

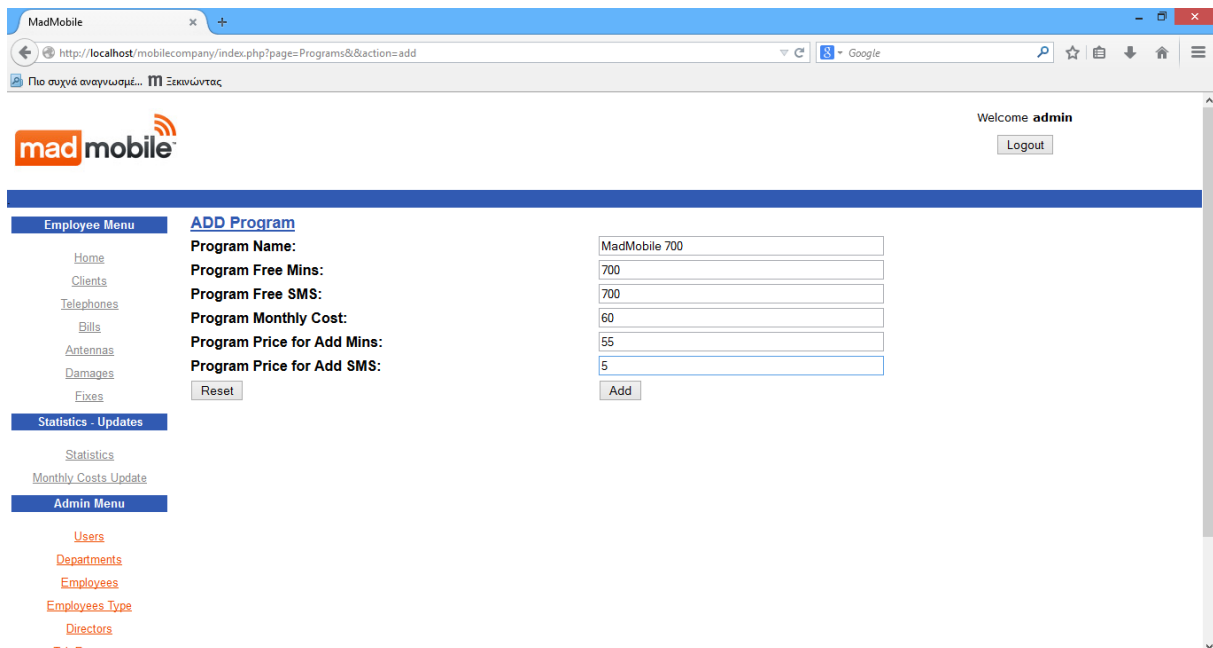


Εικόνα 36: Αρχική Οθόνη Τηλ. Προγραμμάτων

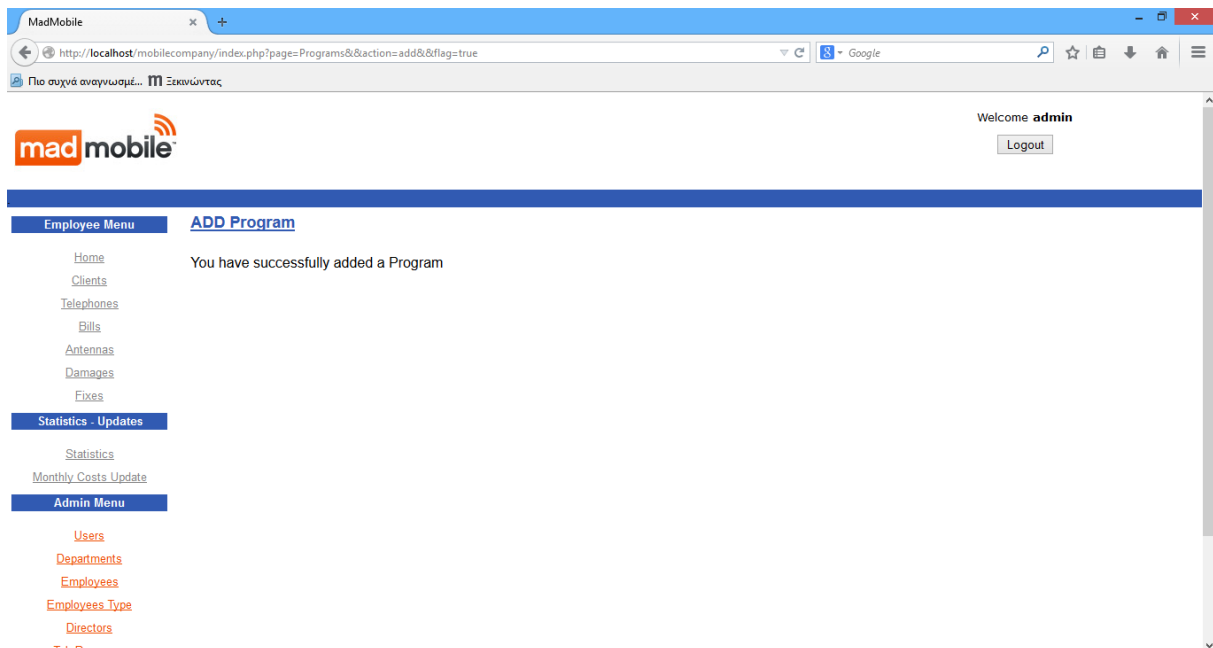
8.8.2 Δημιουργία Τηλεφωνικού Προγράμματος

Screenshots:

Στην οθόνη Δημιουργίας Τηλ. Προγράμματος, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας Τηλ. Πρόγραμμα.



Εικόνα 37: Φόρμα Προσθήκης Νέου Τηλ. Προγράμματος



Εικόνα 38: Ολοκλήρωση προσθήκης Νέου Τηλ.Προγράμματος

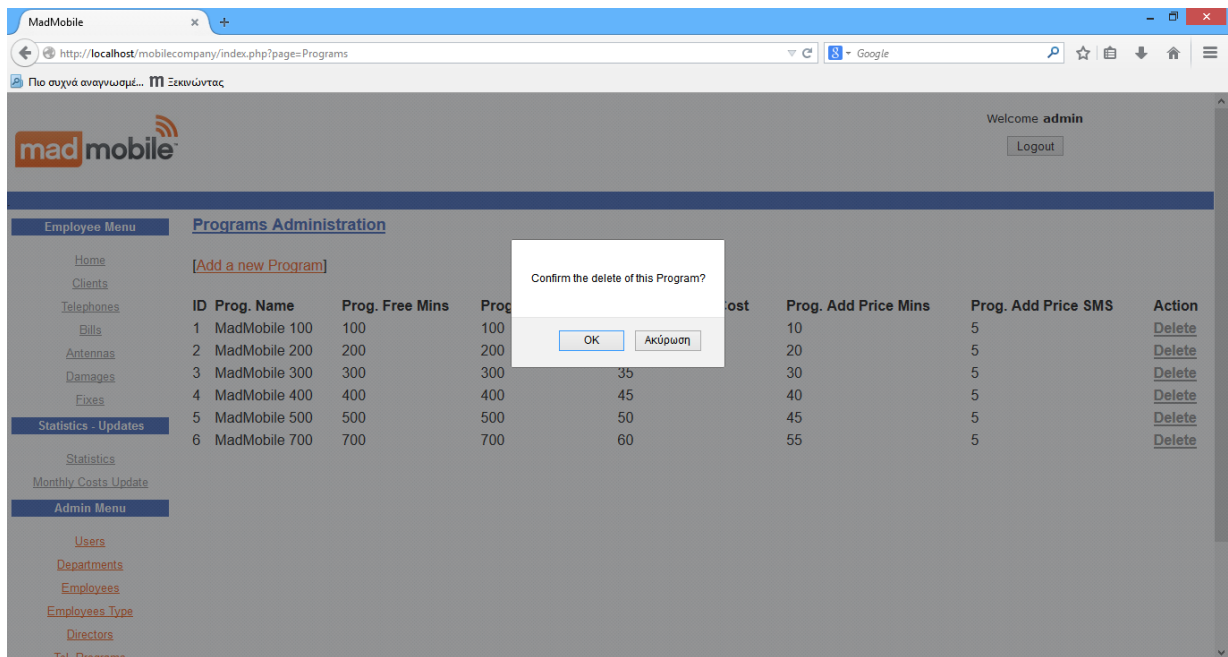
SQL query:

```
insert into programs (p_name, p_freesms, p_freemins, p_monthlycost, p_price_add_mins,  
p_price_add_sms) values ('$p_name', $p_freesms, $p_freemins, $p_monthlycost,  
$p_price_add_mins, $p_price_add_sms)
```

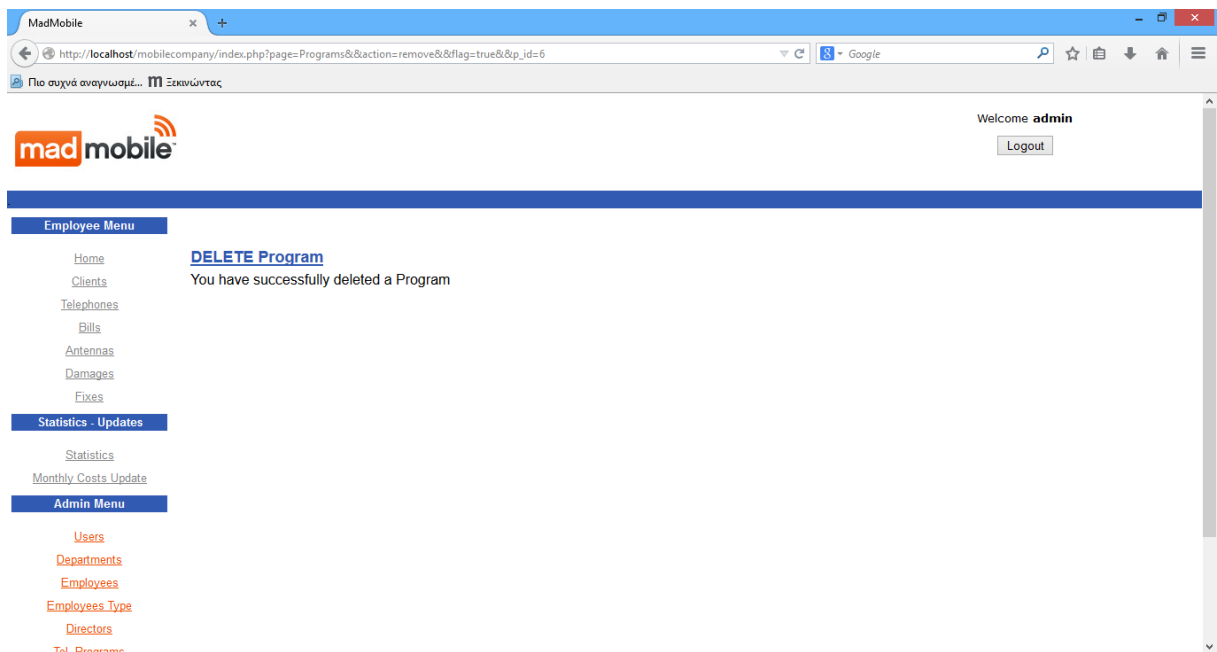
8.8.3 Διαγραφή Τηλ.Προγράμματος

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου τηλ.προγράμματος, περνώντας το id του ως παράμετρο.



Εικόνα 39: Διαγραφή Τηλ. Προγράμματος 1



Εικόνα 40: Διαγραφή Τηλ. Προγράμματος 2

SQL query:

delete from programs where p_id=\$StrtsID

8.9 Οικογενειακά Μέλη

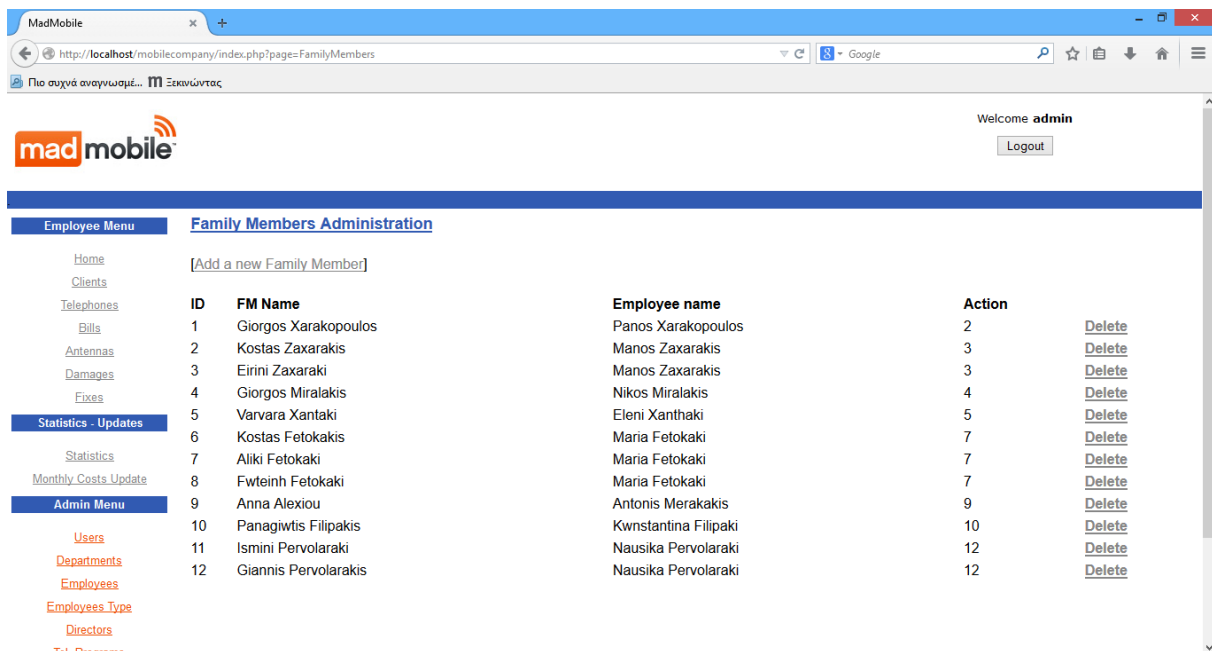
8.9.1 Αρχική Οθόνη Οικογενειακών Μελών

Screenshots:

Στην αρχική οθόνη των οικογενειακών μελών εμφανίζεται η λίστα με τα υπάρχοντα οικογενειακά μέλη και δίπλα σε κάθε ένα, οι σύνδεσμοι Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος οικογενειακό μέλος, τότε εμφανίζεται μήνυμα μη ύπαρξης οικογενειακού μέλους.

Επίσης, στο πάνω μέρος της αρχικής οθόνης οικογενειακών μελών, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας οικογενειακού μέλους.



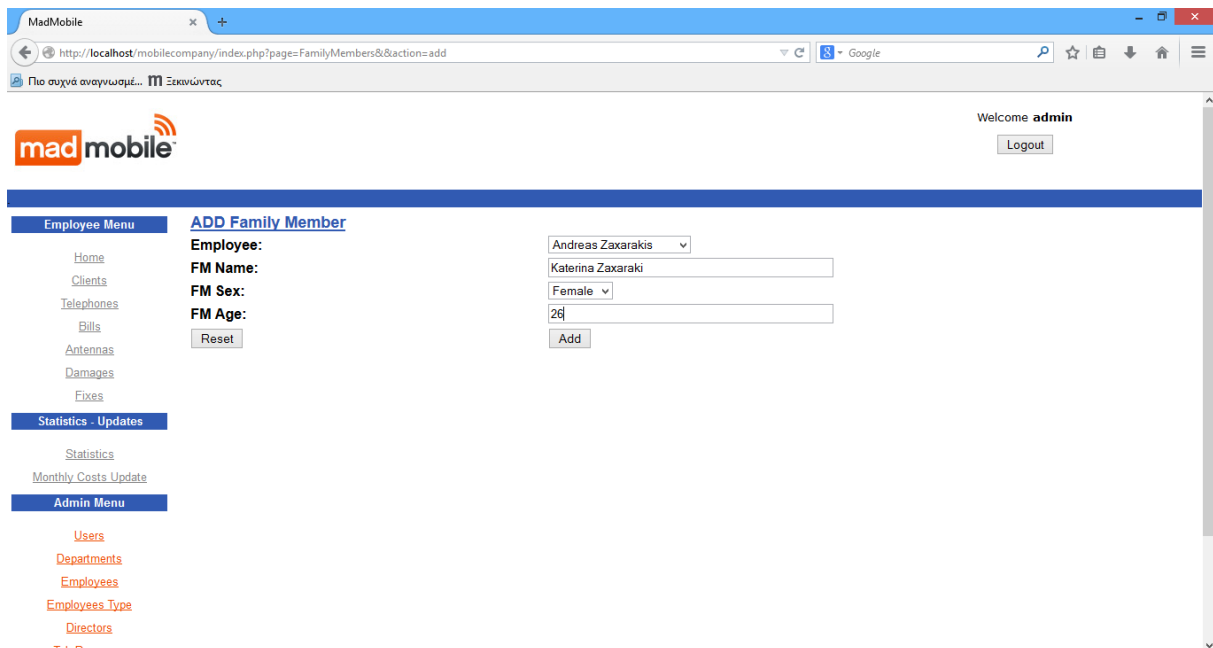
Εικόνα 41: Αρχική Οθόνη Οικογενειακών Μελών

8.9.2 Δημιουργία Οικογενειακού Μέλους

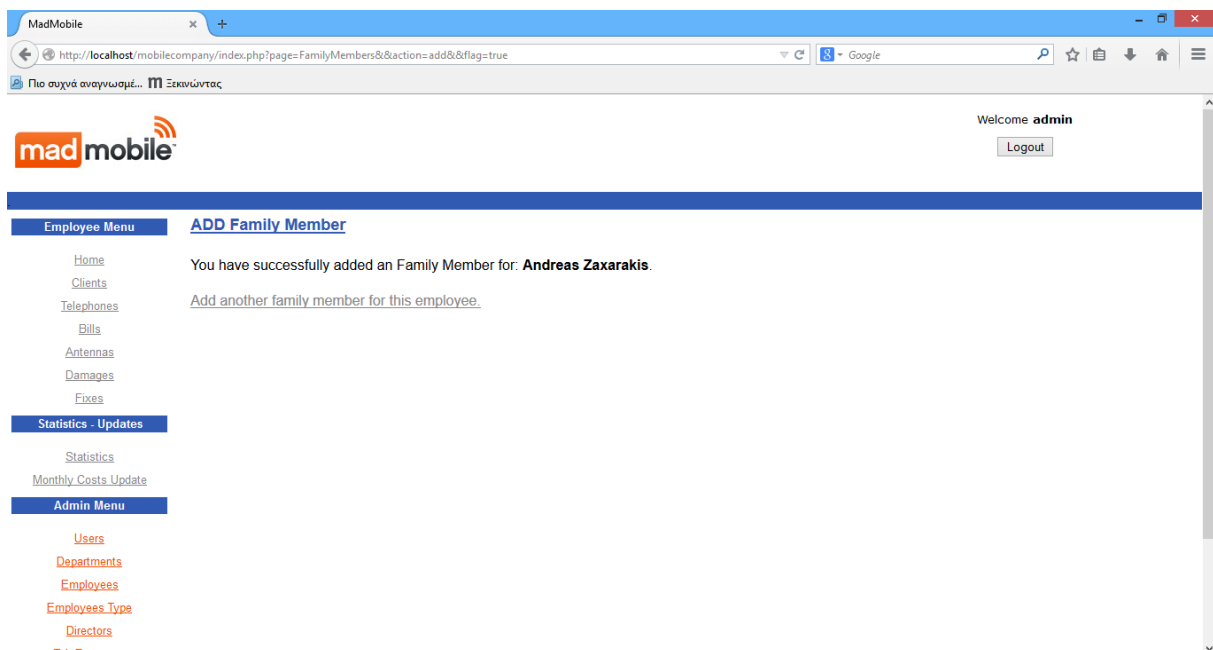
Screenshots:

Στην οθόνη Δημιουργίας οικογενειακού μέλους, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένα οικογενειακό μέλος.

- Η φόρμα δημιουργίας οικογενειακού μέλους έχει λίστα επιλογής με ΟΛΟΥΣ τους διαθέσιμους υπαλλήλους.



Εικόνα 42: Φόρμα Προσθήκης Νέου Οικογενειακού Μέλους



Εικόνα 43: Ολοκλήρωση προσθήκης Νέου Οικογενειακού Μέλους

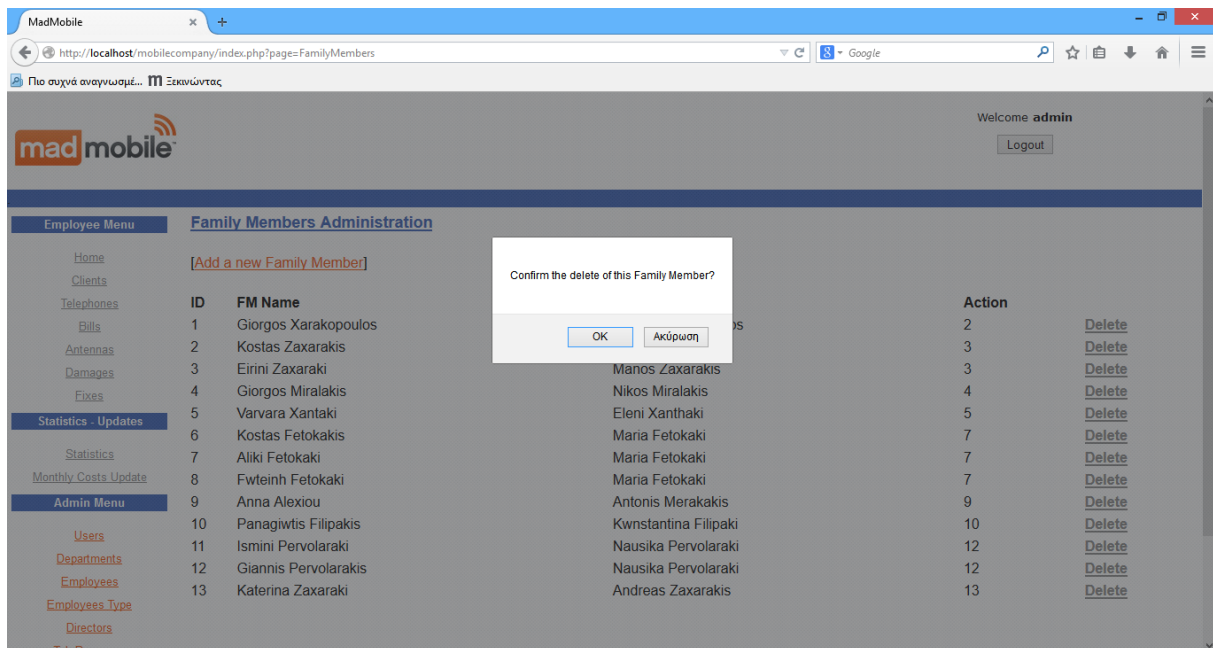
SQL query:

insert into family_members (fname, fsex, fage, eid) values ('\$fname', \$fsex, \$fage, \$eid)

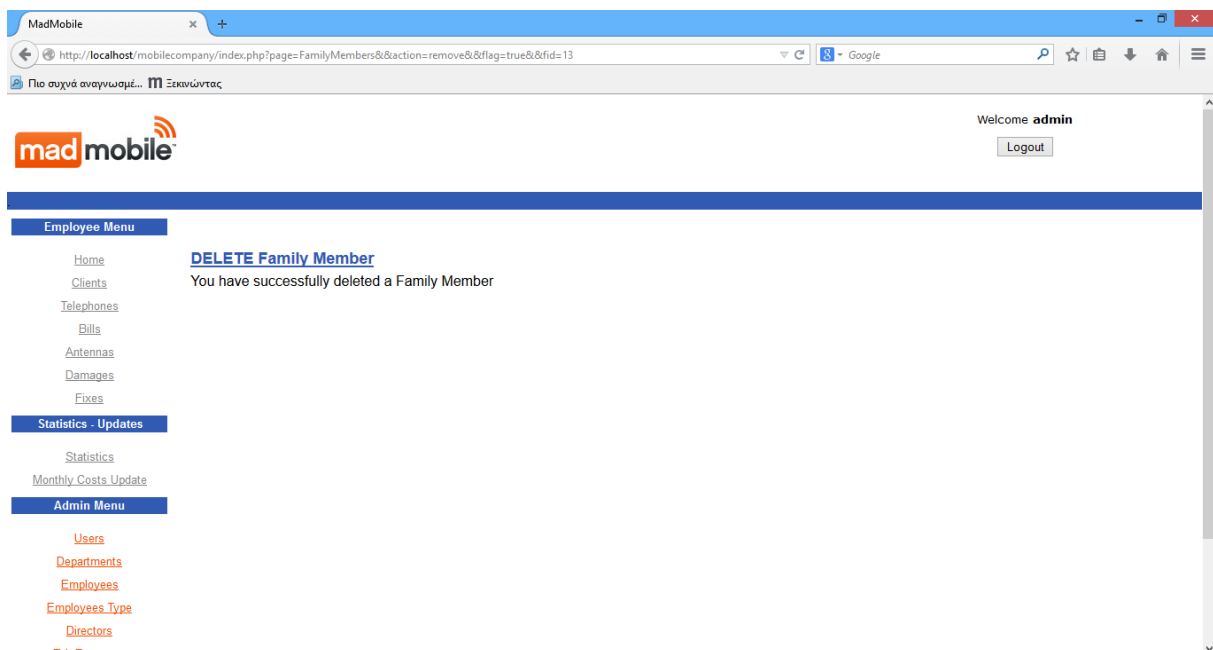
8.9.3 Διαγραφή Οικογενειακού Μέλους

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου οικογενειακού μέλους, περνώντας το id του ως παράμετρο.



Εικόνα 44: Διαγραφή Οικογενειακού Μέλους 1



Εικόνα 45: Διαγραφή Οικογενειακού Μέλους 2

SQL query:

delete from family_members where fid=\$SttrtsID

8.10 Πελάτες

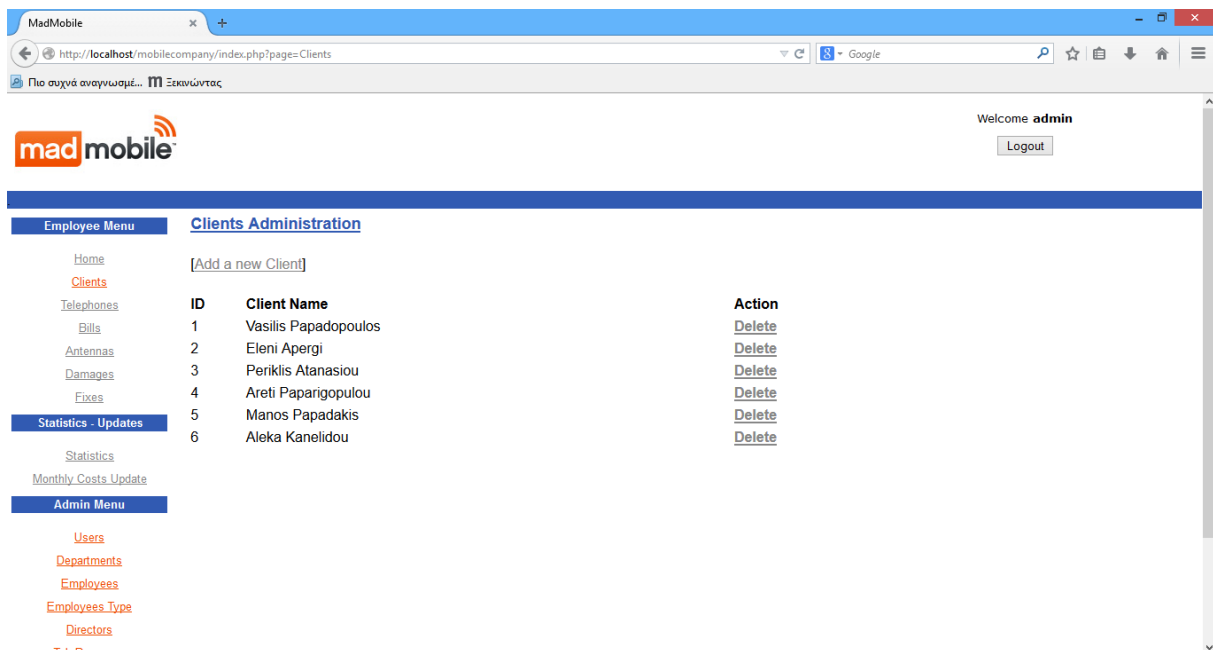
8.10.1 Αρχική Οθόνη Πελατών

Screenshots:

Στην αρχική οθόνη των πελατών εμφανίζεται η λίστα με τους υπάρχοντες πελάτες και δίπλα σε κάθε ένα, οι σύνδεσμοι Επεξεργασίας (Edit) και Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιο πελάτης, τότε εμφανίζεται μήνυμα μη ύπαρξης πελατών.

Επίσης, στο πάνω μέρος της αρχικής οθόνης πελατών, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας πελάτη.

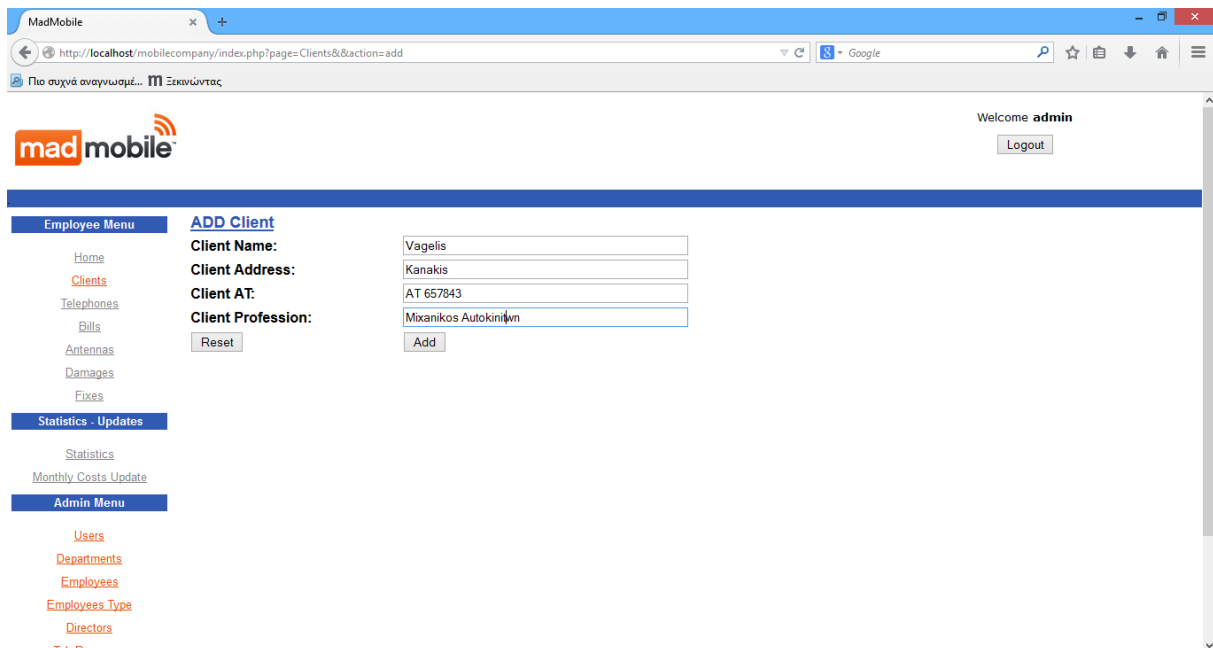


Εικόνα 46: Αρχική Οθόνη Πελατών

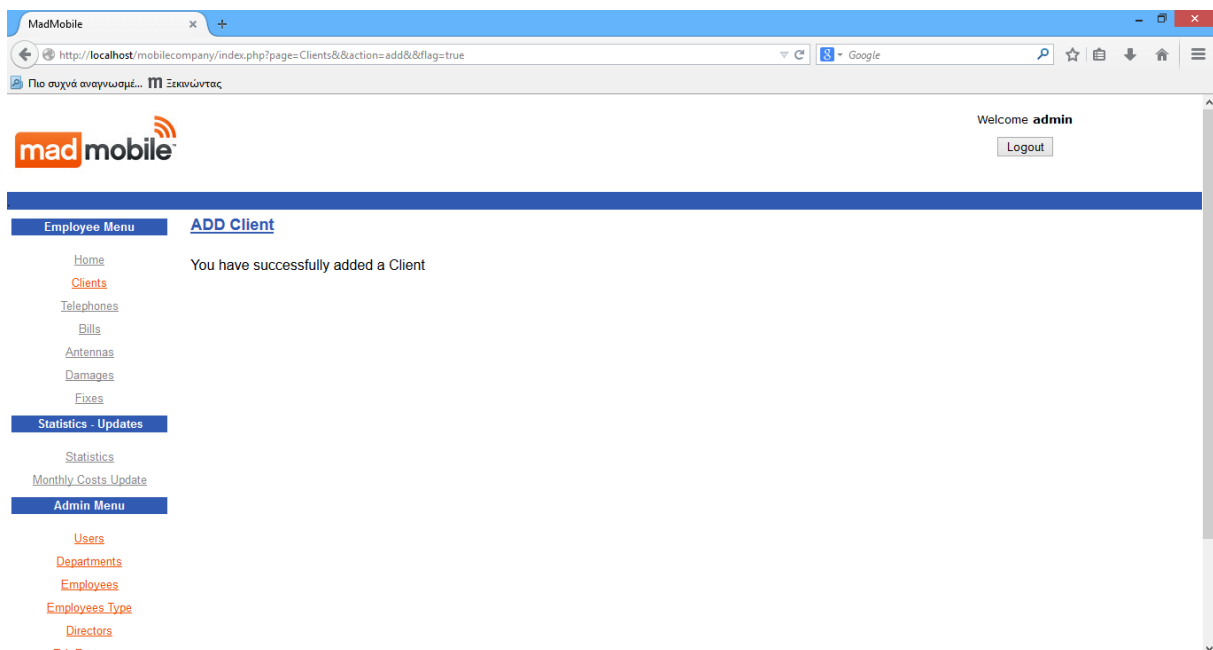
8.10.2 Δημιουργία Πελάτη

Screenshots:

Στην οθόνη Δημιουργίας Πελάτη, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας πελάτης.



Εικόνα 47: Φόρμα Προσθήκης Νέου Πελάτη



Εικόνα 48: Ολοκλήρωση προσθήκης Νέου Πελάτη

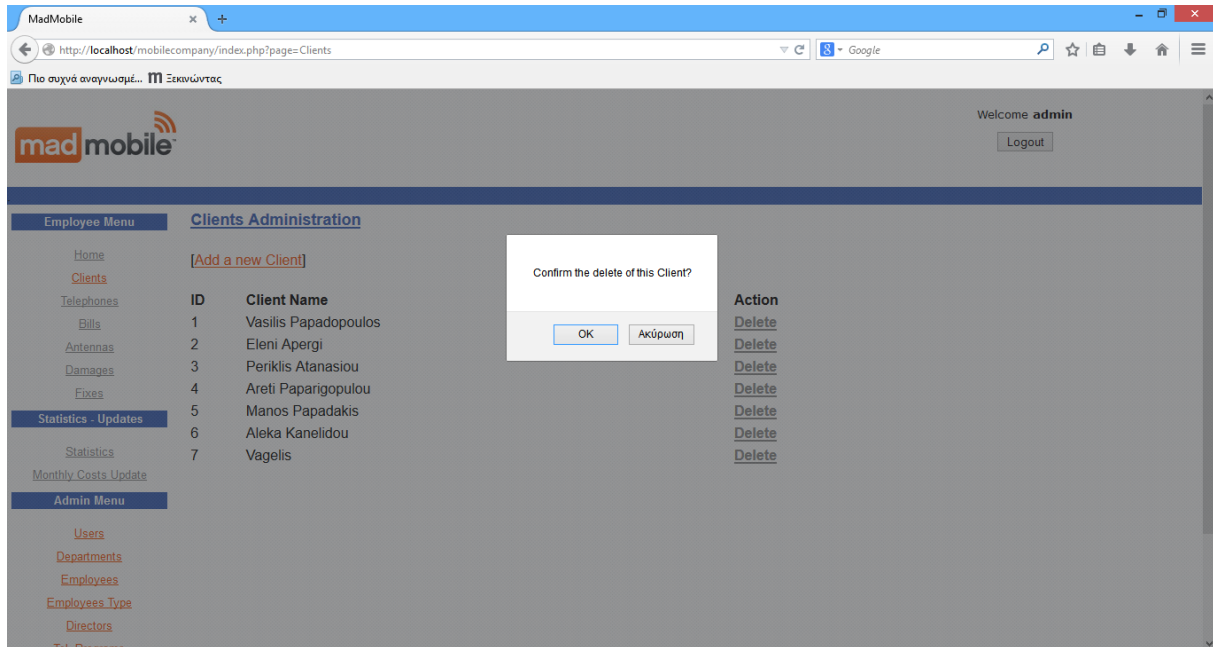
SQL query:

insert into clients (cname, address, cAT, cprof) values ('\$cname', '\$caddress', '\$cAT', '\$cprof')

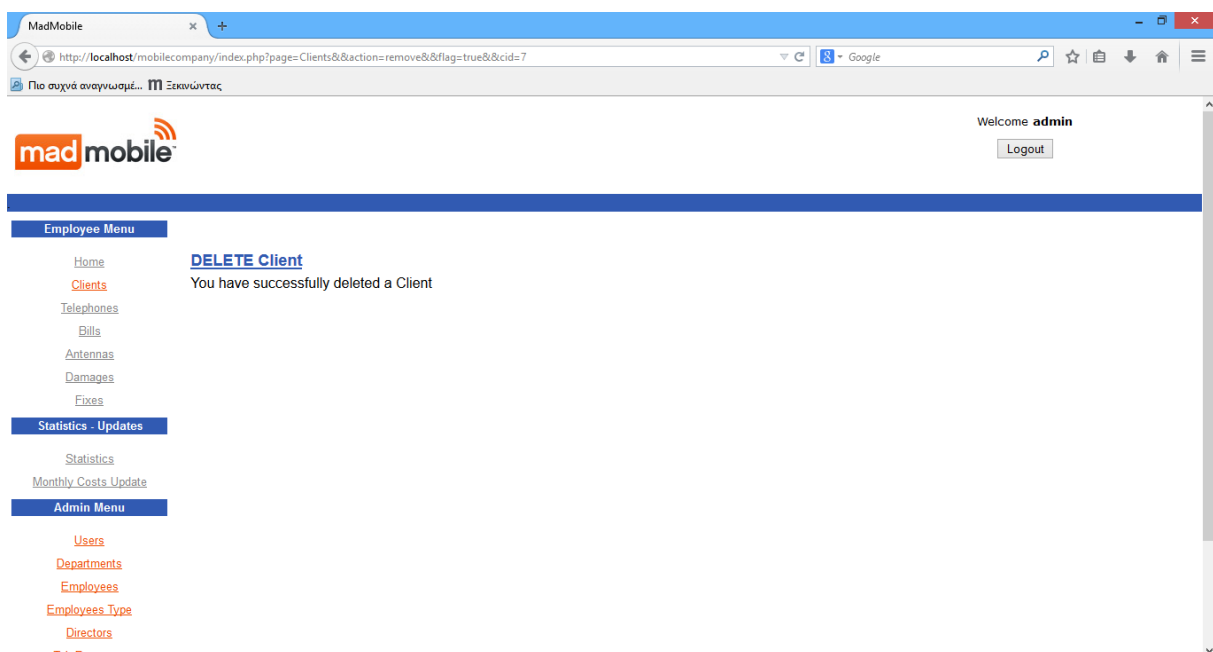
8.10.3 Διαγραφή Πελάτη

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου πελάτη, περνώντας το id του ως παράμετρο.



Εικόνα 49: Διαγραφή Πελάτη 1



Εικόνα 50: Διαγραφή Πελάτη 2

SQL query:

```
delete from clients where cid=$StrtsID
```


8.11 Τηλεφωνικοί Αριθμοί

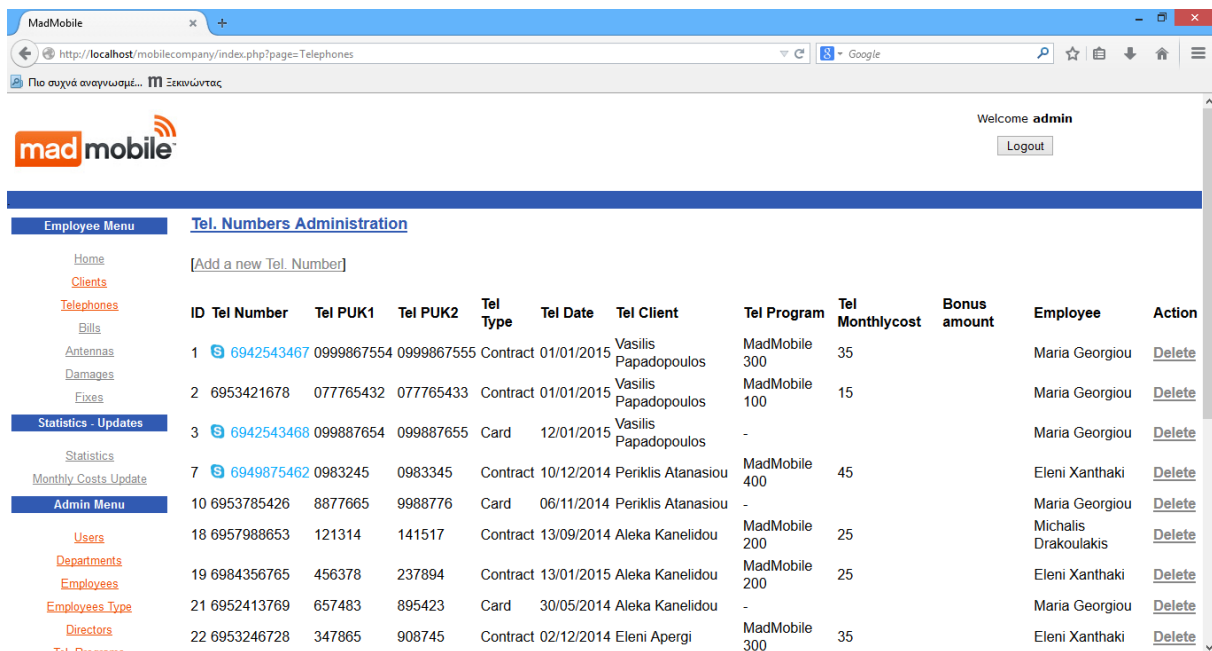
8.11.1 Αρχική Οθόνη Τηλεφωνικών Αριθμών

Screenshots:

Στην αρχική οθόνη των τηλ. αριθμών εμφανίζεται η λίστα με τους υπάρχοντες τηλ. αριθμούς και δίπλα σε κάθε ένα, οι σύνδεσμοι Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος τηλ. αριθμός, τότε εμφανίζεται μήνυμα μη ύπαρξης τηλ. αριθμού.

Επίσης, στο πάνω μέρος της αρχικής οθόνης τηλ. αριθμων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας τηλ. αριθμού.



ID	Tel Number	Tel PUK1	Tel PUK2	Tel Type	Tel Date	Tel Client	Tel Program	Tel Monthlycost	Bonus amount	Employee	Action
1	6942543467	0999867554	0999867555	Contract	01/01/2015	Vasilis Papadopoulos	MadMobile 300	35		Maria Georgiou	Delete
2	6953421678	077765432	077765433	Contract	01/01/2015	Vasilis Papadopoulos	MadMobile 100	15		Maria Georgiou	Delete
3	6942543468	099887654	099887655	Card	12/01/2015	Vasilis Papadopoulos	-			Maria Georgiou	Delete
7	6949875462	0983245	0983345	Contract	10/12/2014	Periklis Atanasiou	MadMobile 400	45		Eleni Xanthaki	Delete
10	6953785426	8877665	9988776	Card	06/11/2014	Periklis Atanasiou	-			Maria Georgiou	Delete
18	6957988653	121314	141517	Contract	13/09/2014	Aleka Kanelidou	MadMobile 200	25		Michalis Drakoulakis	Delete
19	6984356765	456378	237894	Contract	13/01/2015	Aleka Kanelidou	MadMobile 200	25		Eleni Xanthaki	Delete
21	6952413769	657483	895423	Card	30/05/2014	Aleka Kanelidou	-			Maria Georgiou	Delete
22	6953246728	347865	908745	Contract	02/12/2014	Eleni Apergi	MadMobile 300	35		Eleni Xanthaki	Delete

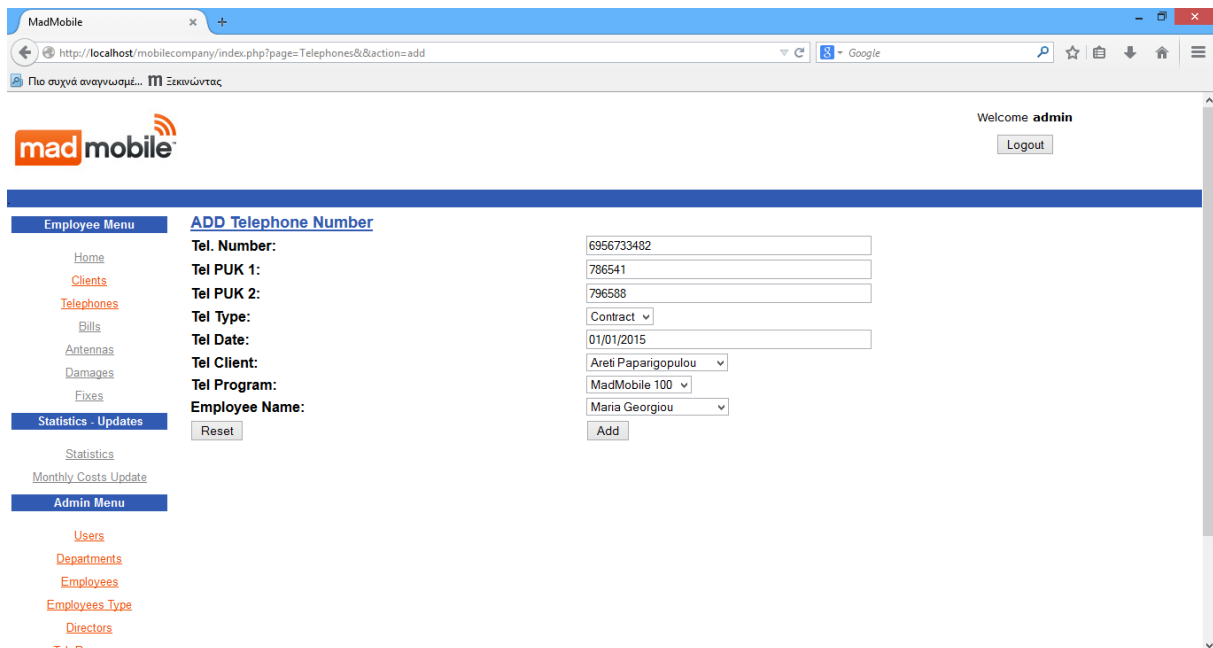
Εικόνα 51: Αρχική Οθόνη Τηλεφωνικών Αριθμών

8.11.2 Δημιουργία Τηλεφωνικού Αριθμού

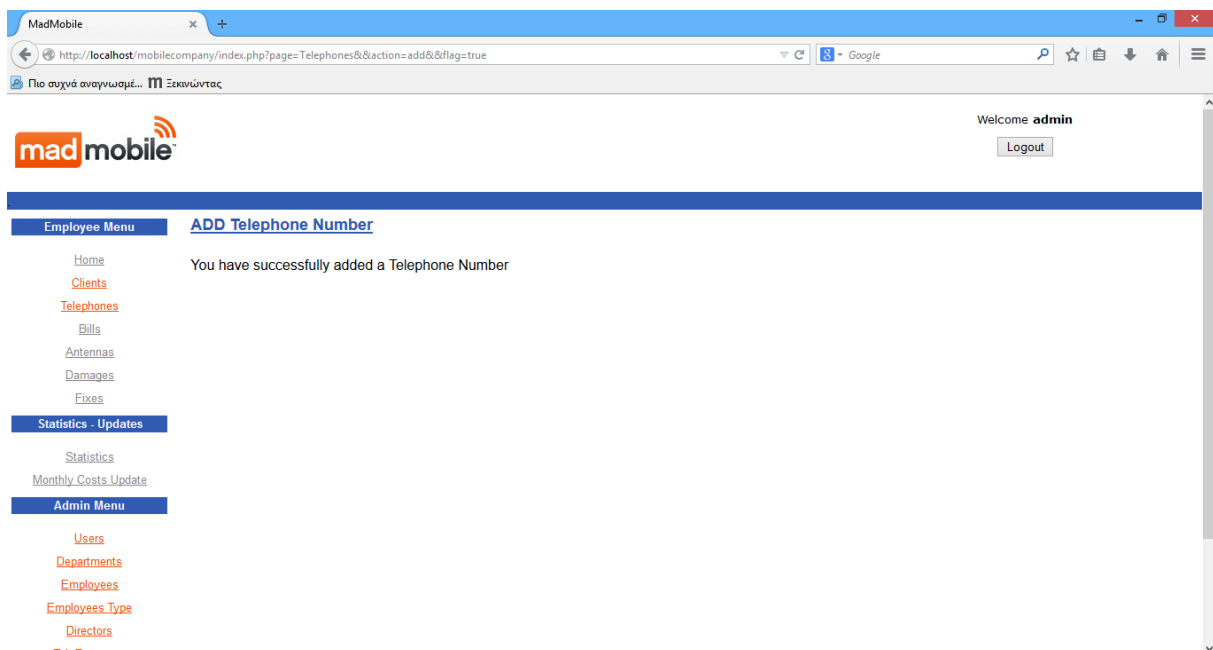
Screenshots:

Στην οθόνη Δημιουργίας Τηλ. Αριθμού εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας Τηλ. Αριθμός.

- Η φόρμα δημιουργίας τηλ.αριθμού έχει λίστα επιλογής με ΟΛΟΥΣ τους διαθέσιμους πελάτες.
- Η φόρμα δημιουργίας τηλ.αριθμού έχει λίστα επιλογής με ΟΛΑ τα διαθέσιμα τηλ. προγράμματα.



Εικόνα 52: Φόρμα Προσθήκης Νέου Τηλ. Αριθμού



Εικόνα 53: Ολοκλήρωση προσθήκης Νέου Τηλ.Αριθμού

SQL query:

Συμβόλαιο

insert into telephones (tel_number, tel_puk1, tel_puk2, tel_type, cid, p_id, eid, tel_date, p_monthlycost) values ('\$tel_number', '\$tel_puk1', '\$tel_puk2', '\$tel_type', \$cid, \$p_id, \$eid, '\$tel_date', \$p_monthlycost)

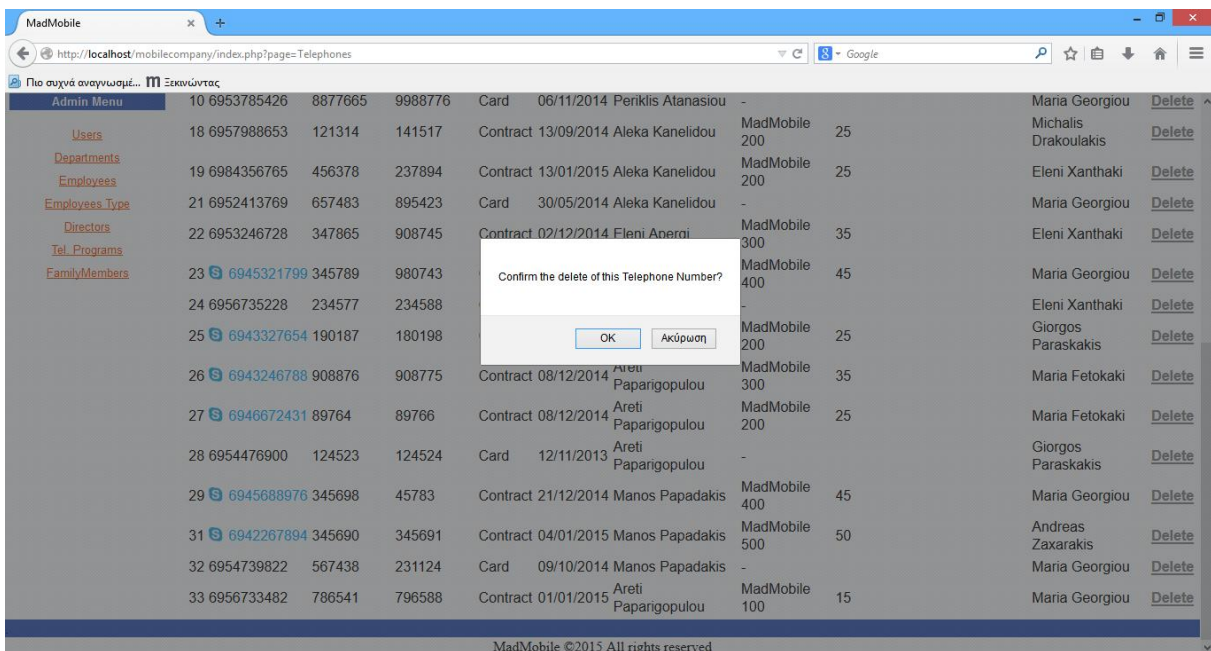
Κάρτα

insert into telephones (tel_number, tel_puk1, tel_puk2, tel_type, cid, eid, tel_date) values ('\$stel_number', '\$stel_puk1', '\$stel_puk2', \$stel_type, \$cid, \$eid, '\$stel_date')

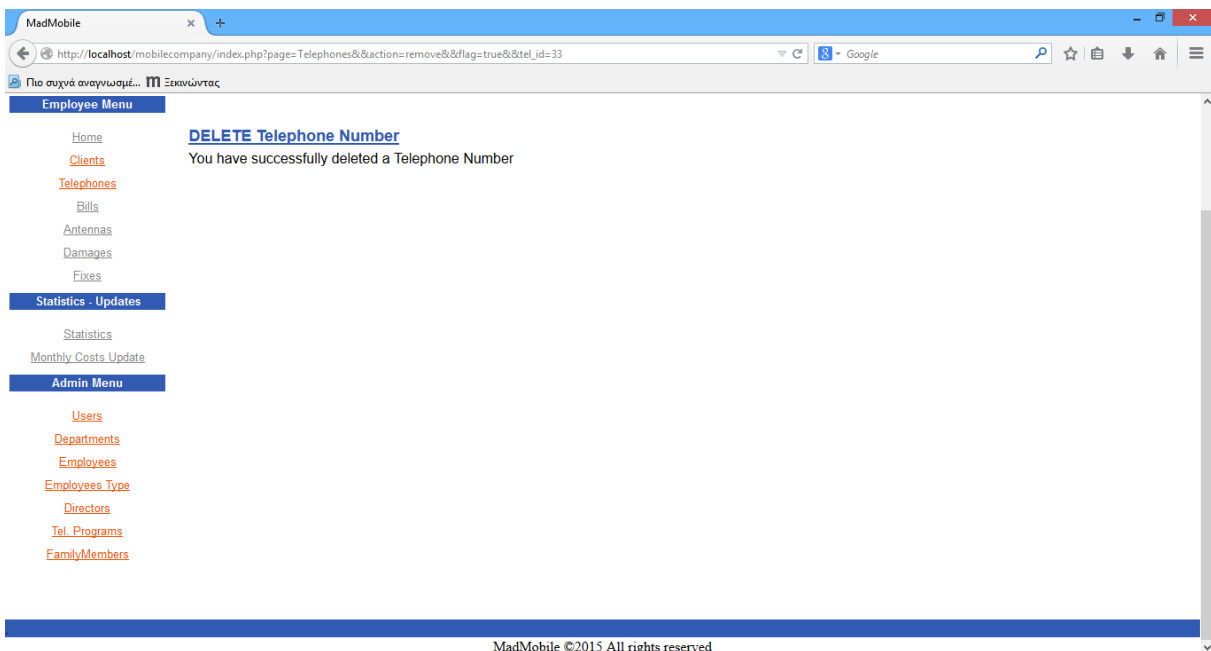
8.11.3 Διαγραφή Τηλ. Αριθμού

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου τηλ. αριθμού, περνώντας το id του ως παράμετρο.



Εικόνα 54: Διαγραφή Τηλ. Αριθμού1



Εικόνα 55: Διαγραφή Τηλ. Αριθμού1

SQL query:

```
delete from telephones where tel_id=$StrtsID
```

8.12 Λογαριασμοί

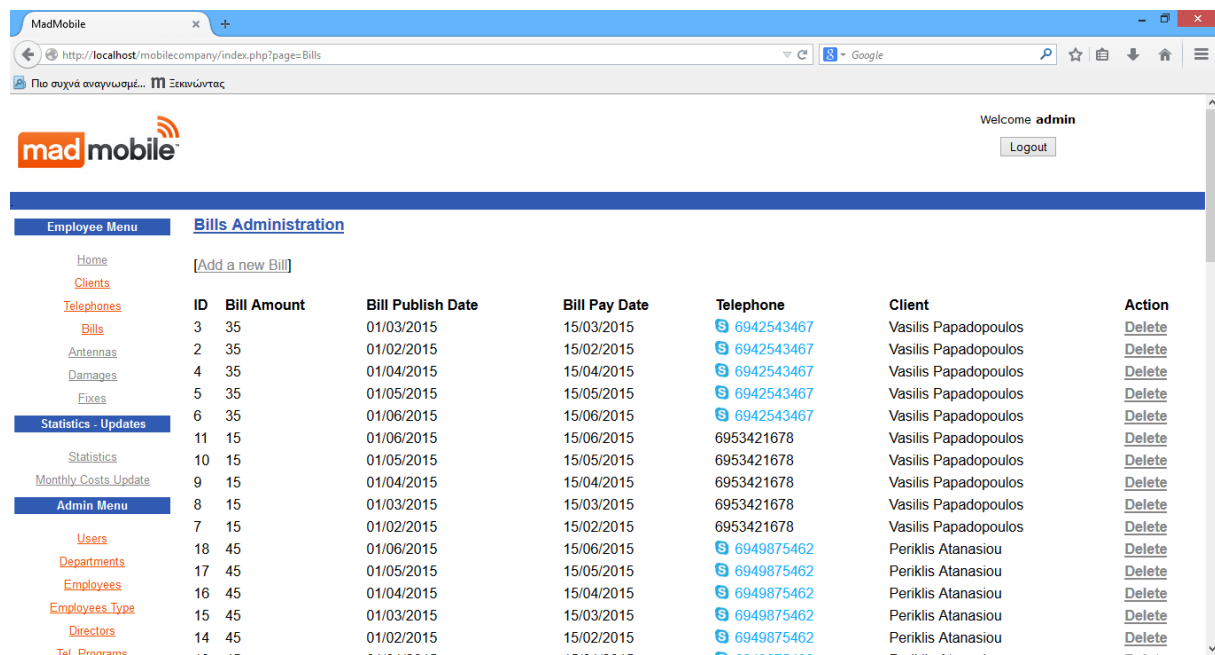
8.12.1 Αρχική Οθόνη Λογαριασμών

Screenshots:

Στην αρχική οθόνη των λογαριασμών εμφανίζεται η λίστα με τους υπάρχοντες λογαριασμούς και δίπλα σε κάθε ένα, οι σύνδεσμοι Διαγραφής (Delete) για κάθε ένα από αυτούς.

Εάν δεν έχει δημιουργηθεί κάποιος λογαριασμός, τότε εμφανίζεται μήνυμα μη ύπαρξης λογαριασμού.

Επίσης, στο πάνω μέρος της αρχικής οθόνης λογαριασμών, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας λογαριασμού.



ID	Bill Amount	Bill Publish Date	Bill Pay Date	Telephone	Client	Action
3	35	01/03/2015	15/03/2015	6942543467	Vasilis Papadopoulos	Delete
2	35	01/02/2015	15/02/2015	6942543467	Vasilis Papadopoulos	Delete
4	35	01/04/2015	15/04/2015	6942543467	Vasilis Papadopoulos	Delete
5	35	01/05/2015	15/05/2015	6942543467	Vasilis Papadopoulos	Delete
6	35	01/06/2015	15/06/2015	6942543467	Vasilis Papadopoulos	Delete
11	15	01/06/2015	15/06/2015	6953421678	Vasilis Papadopoulos	Delete
10	15	01/05/2015	15/05/2015	6953421678	Vasilis Papadopoulos	Delete
9	15	01/04/2015	15/04/2015	6953421678	Vasilis Papadopoulos	Delete
8	15	01/03/2015	15/03/2015	6953421678	Vasilis Papadopoulos	Delete
7	15	01/02/2015	15/02/2015	6953421678	Vasilis Papadopoulos	Delete
18	45	01/06/2015	15/06/2015	6949875462	Periklis Atanasiou	Delete
17	45	01/05/2015	15/05/2015	6949875462	Periklis Atanasiou	Delete
16	45	01/04/2015	15/04/2015	6949875462	Periklis Atanasiou	Delete
15	45	01/03/2015	15/03/2015	6949875462	Periklis Atanasiou	Delete
14	45	01/02/2015	15/02/2015	6949875462	Periklis Atanasiou	Delete

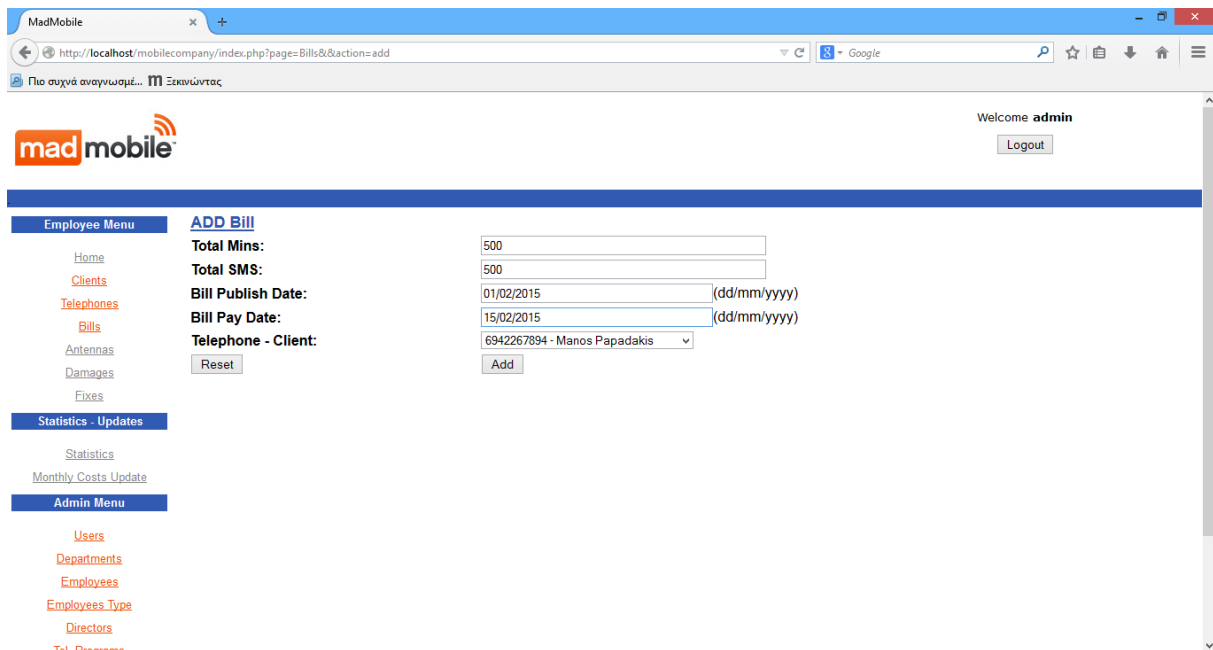
Εικόνα 56: Αρχική Οθόνη Λογαριασμών

8.12.2 Δημιουργία Λογαριασμού

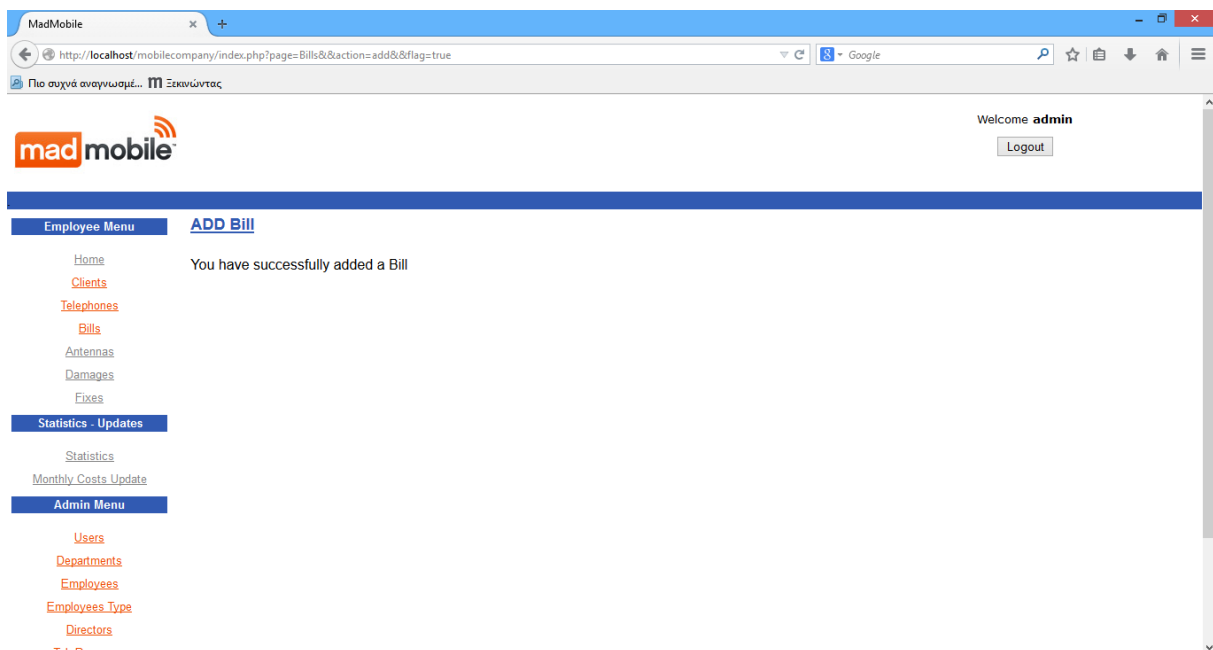
Screenshots:

Στην οθόνη Δημιουργίας Λογαριασμού, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί ένας Λογαριασμός.

- Η φόρμα δημιουργίας λογαριασμού έχει λίστα επιλογής με ΟΛΟΥΣ τους διαθέσιμους τηλ.αριθμούς και τον πελάτη που αντιστοιχούν.



Εικόνα 57: Φόρμα Προσθήκης Νέου Λογαριασμού



Εικόνα 58: Ολοκλήρωση προσθήκης Νέου Λογαριασμού

SQL query:

Υπολογισμός ποσού

$$\text{\$bamount} = \text{\$p_monthlycost} + (\text{\$total_mins} - \text{\$p_freemins}) * \text{\$p_price_add_mins} + (\text{\$total_sms} - \text{\$p_freesms}) * \text{\$p_price_add_sms};$$

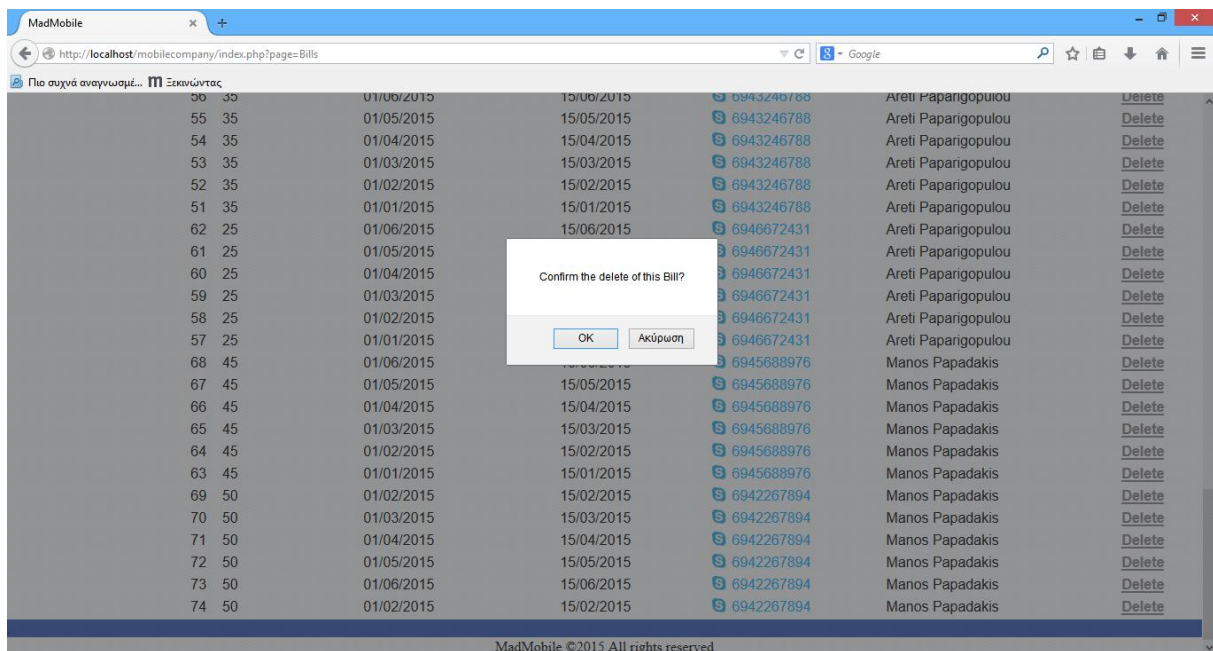
Δημιουργία λογαριασμού

insert into bills (bamount, bpubdate, bpaydate, tel_id, cid) values (\$bamount, '\$bpubdate', '\$bpaydate', \$tel_id, \$cid)

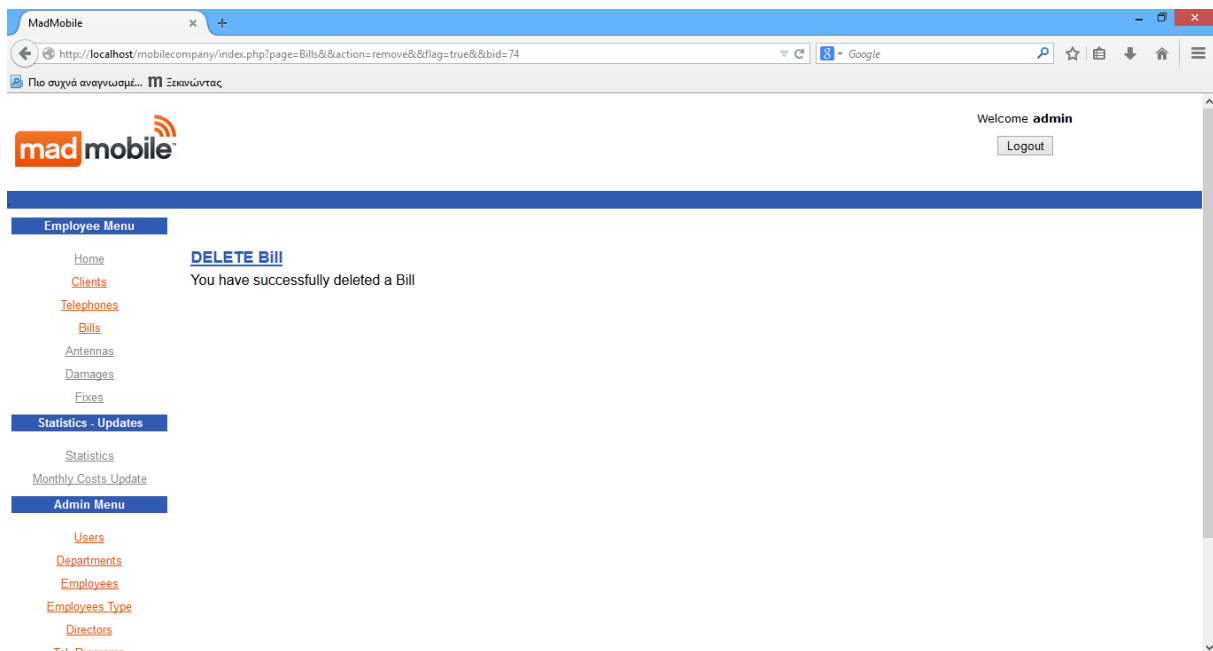
8.12.3 Διαγραφή Λογαριασμού

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής του συγκεκριμένου λογαριασμού, περνώντας το id του ως παράμετρο.



Εικόνα 59: Διαγραφή Λογαριασμού 1



Εικόνα 60: Διαγραφή Λογαριασμού 2

SQL query:

delete from bills where bid=\$StrtsID

8.13 Κεραίες

8.13.1 Αρχική Οθόνη Κεραίων

Screenshots:

Στην αρχική οθόνη των κεραίων εμφανίζεται η λίστα με τις υπάρχοντες κεραίες και δίπλα σε κάθε μία, οι σύνδεσμοι Διαγραφής (Delete) για κάθε μία από αυτές.

Εάν δεν έχει δημιουργηθεί κάποια κεραία, τότε εμφανίζεται μήνυμα μη ύπαρξης κεραίας.

Επίσης, στο πάνω μέρος της αρχικής οθόνης κεραίων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας κεραίας.

ID	Antenna Name	Antenna Address	Antenna Type	Department	Action
1	Anntena 1	Epimenidou 56	sector antennas	MadMobile (Κέν	Delete
2	Anntena 2	Plateia Eleutherias	sector antennas	MadMobile (Κέν	Delete
3	Anntena 3	Mauswlou 23	dish antenna	MadMobile (ΑΑ	Delete
4	Anntena 4	Mauswlou 23	sector antennas	MadMobile (ΑΑ	Delete
5	Anntena 5	Leoforos 62 Marturwn 45	sector antennas	MadMobile (Θέρ	Delete
6	Anntena 6	Leoforos 62 Marturwn 45	dish antenna	MadMobile (Θέρ	Delete

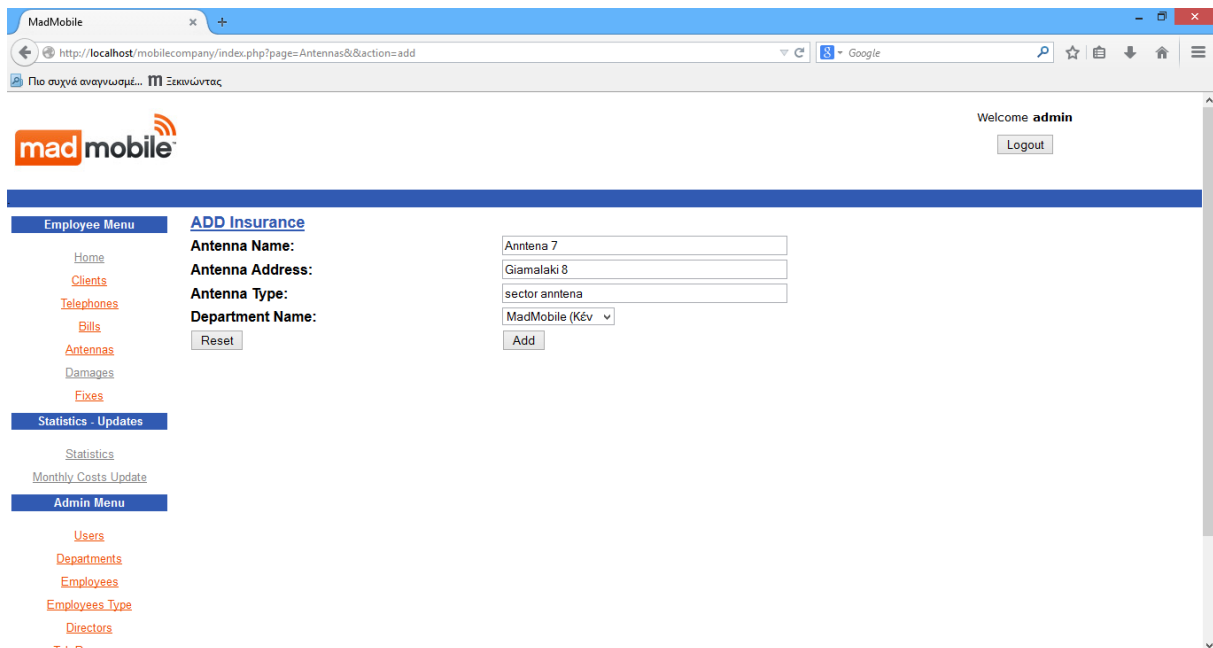
Εικόνα 61: Αρχική Οθόνη Κεραίων

8.13.2 Δημιουργία Κεραίας

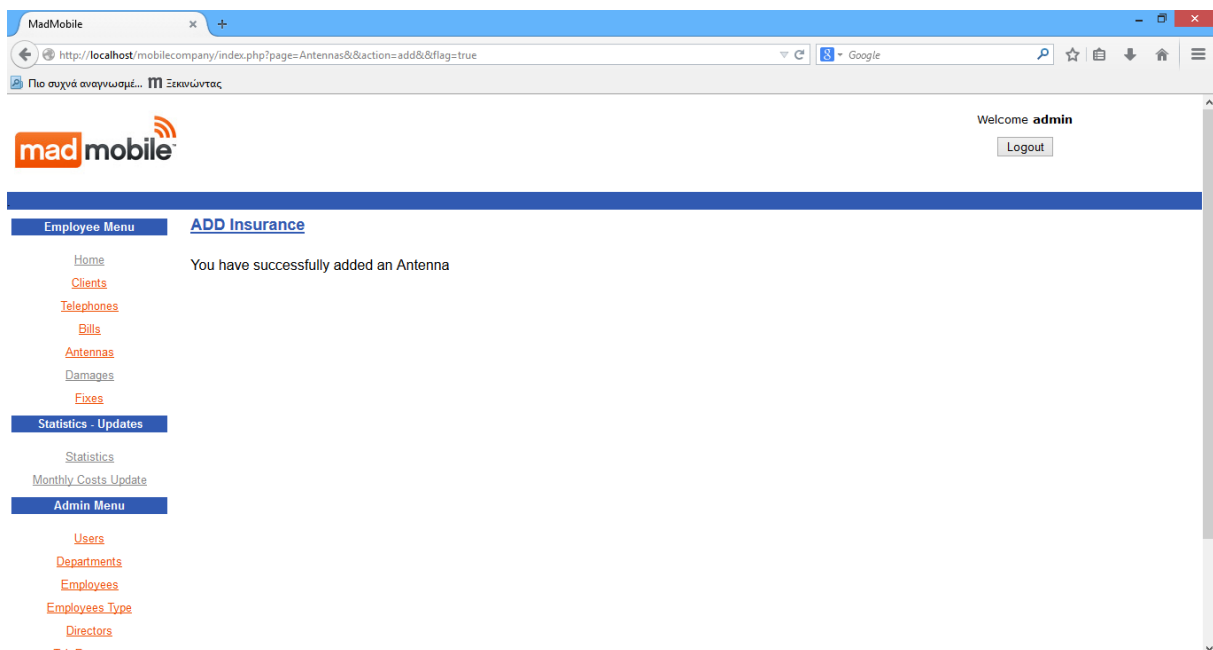
Screenshots:

Στην οθόνη Δημιουργίας Κεραίας, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί μία Κεραία.

- Η φόρμα δημιουργίας κεραίας έχει λίστα επιλογής με ΟΛΑ τα διαθέσιμα τμήματα.



Εικόνα 62: Φόρμα Προσθήκης Νέας Κεραίας



Εικόνα 63: Ολοκλήρωση προσθήκης Νέας Κεραίας

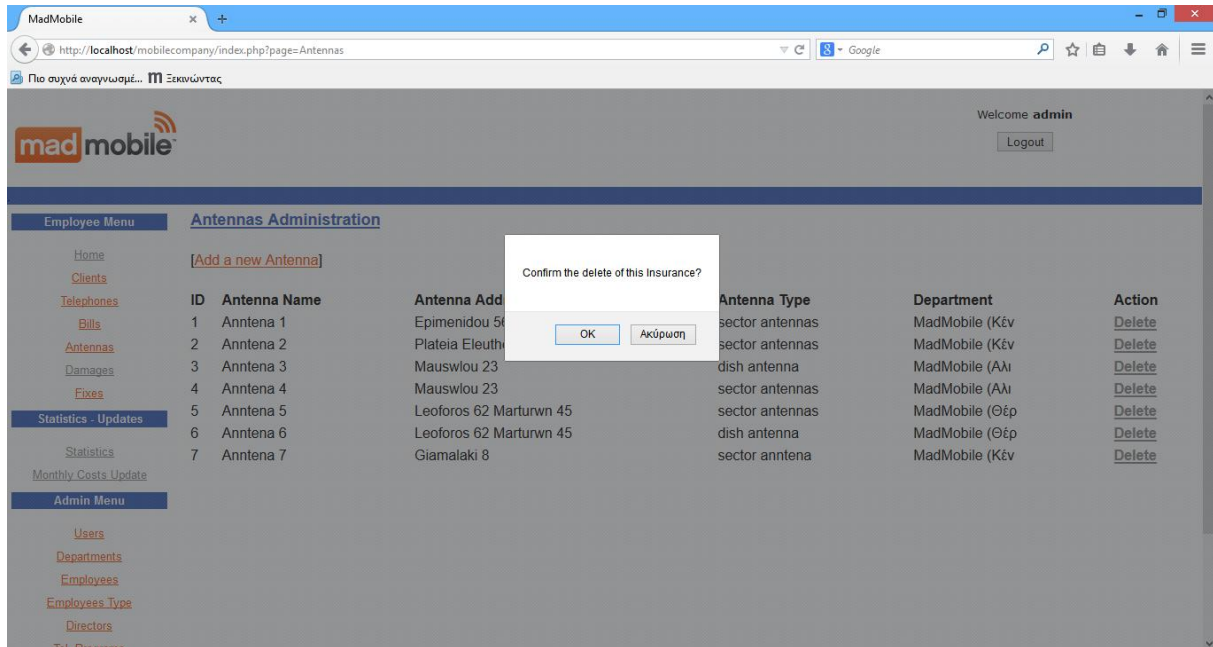
SQL query:

insert into antennas (ant_name, ant_address, ant_type, did) values ('\$ant_name', '\$ant_address', '\$ant_type', \$did)

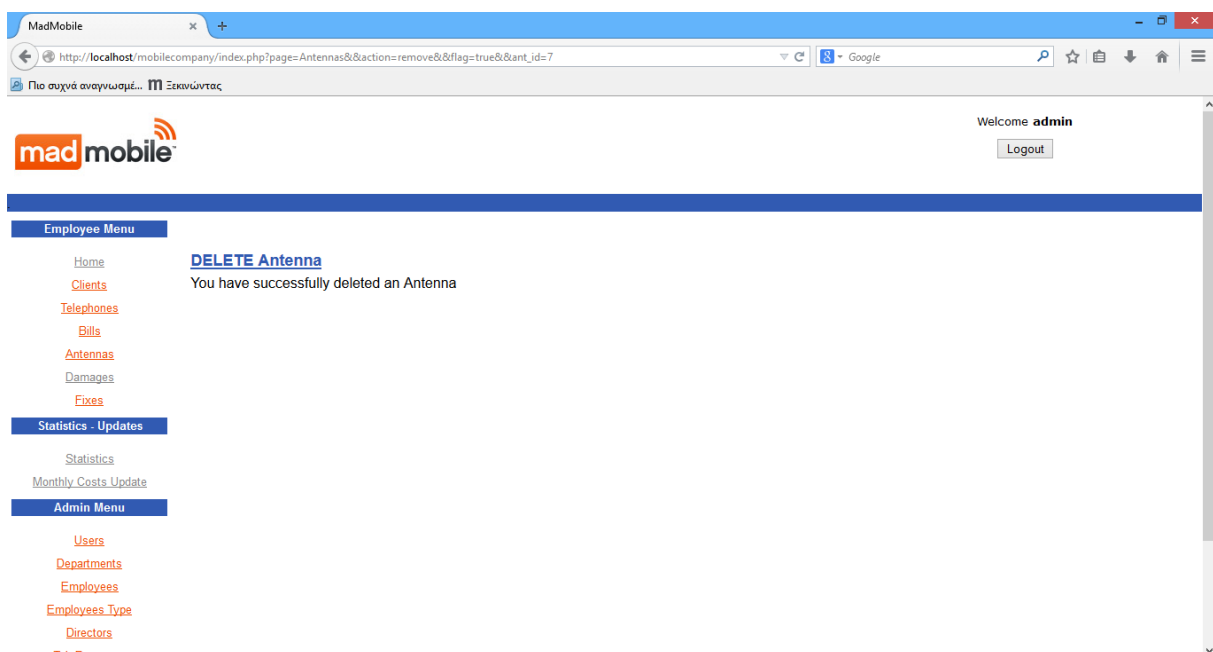
8.13.3 Διαγραφή Κεραίας

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής της συγκεκριμένης κεραίας, περνώντας το id της ως παράμετρο.



Εικόνα 64: Διαγραφή Κεραίας 1



Εικόνα 65: Διαγραφή Κεραίας 1

SQL query:

delete from antennas where ant_id=\$StirtsID

8.14 Βλάβες

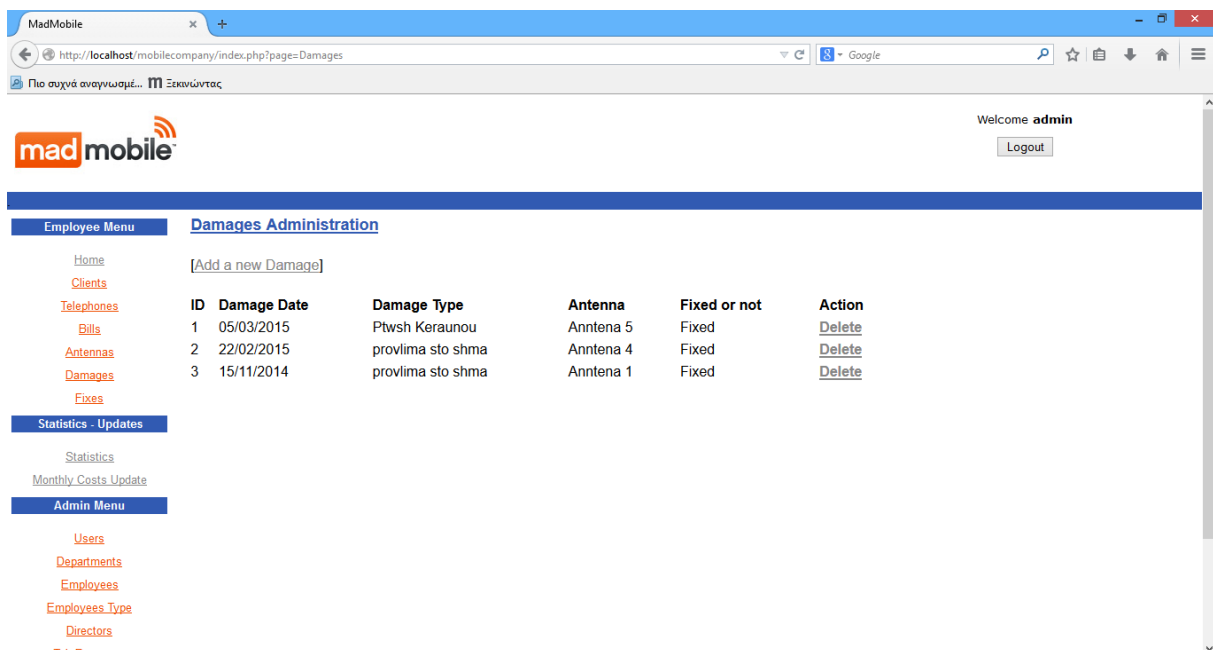
8.14.1 Αρχική Οθόνη Βλαβών

Screenshots:

Στην αρχική οθόνη των κεραιών εμφανίζεται η λίστα με τις υπάρχοντες βλάβες και δίπλα σε κάθε μία, οι σύνδεσμοι Διαγραφής (Delete) για κάθε μία από αυτές.

Εάν δεν έχει δημιουργηθεί κάποια βλάβη, τότε εμφανίζεται μήνυμα μη ύπαρξης βλάβης.

Επίσης, στο πάνω μέρος της αρχικής οθόνης βλαβών, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας βλάβης.



The screenshot shows the 'Damages Administration' page in the MadMobile application. The page features a table with the following data:

ID	Damage Date	Damage Type	Antenna	Fixed or not	Action
1	05/03/2015	Ptwsh Keraunou	Antenna 5	Fixed	Delete
2	22/02/2015	provlima sto shma	Antenna 4	Fixed	Delete
3	15/11/2014	provlima sto shma	Antenna 1	Fixed	Delete

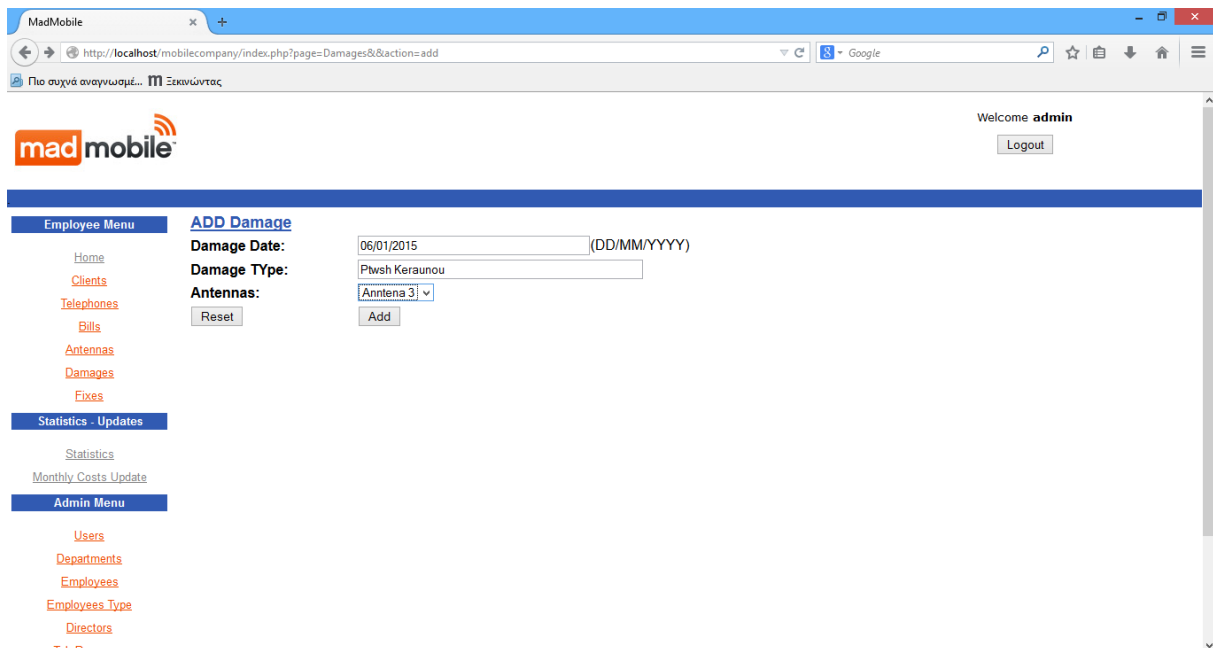
Εικόνα 66: Αρχική οθόνη Βλαβών

8.14.2 Δημιουργία Βλάβης

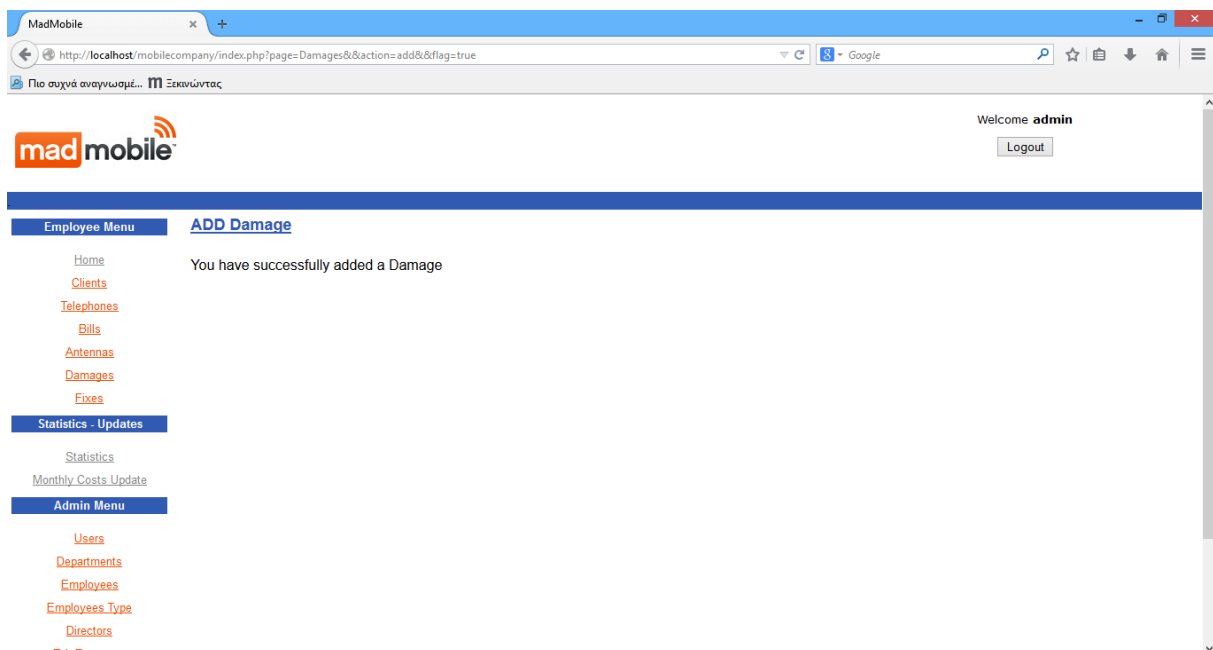
Screenshots:

Στην οθόνη Δημιουργίας Βλάβης, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί μία Βλάβη.

- Η φόρμα δημιουργίας βλάβης έχει λίστα επιλογής με ΟΛΕΣ τις διαθέσιμες κεραιές.



Εικόνα 67: Φόρμα Προσθήκης Νέας Βλάβης



Εικόνα 68: Ολοκλήρωση προσθήκης Νέας Βλάβης

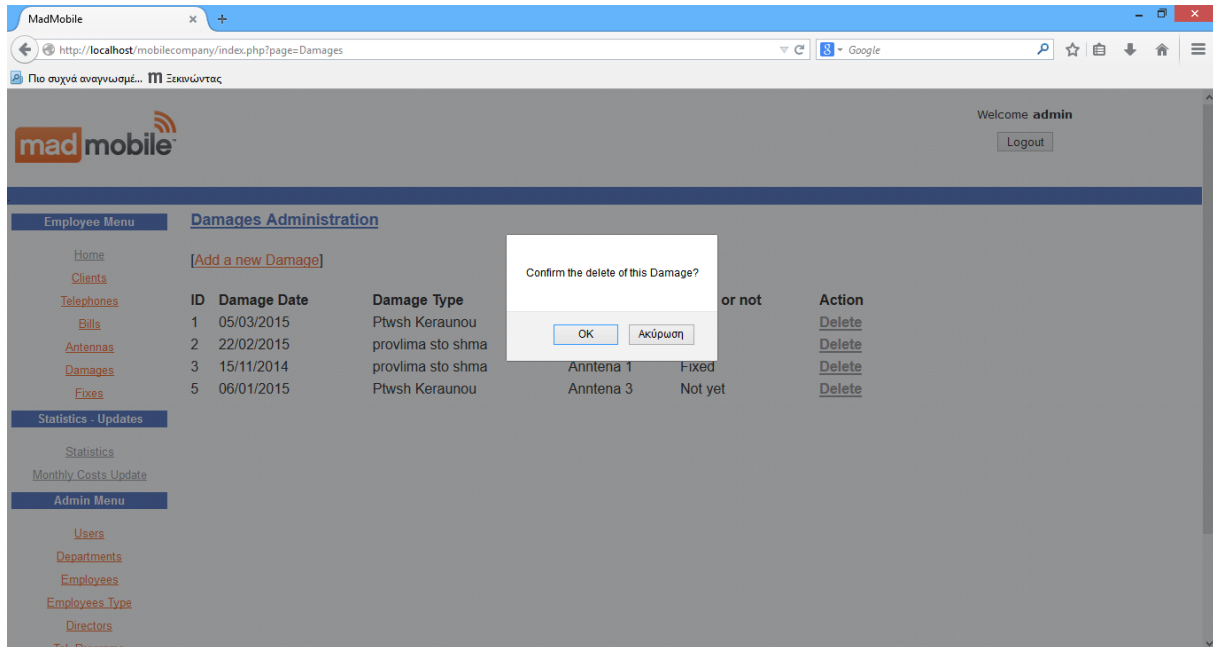
SQL query:

insert into damages (damage_date, damage_type, ant_id, fixed_or_not) values ('\$damage_date', '\$damage_type', \$ant_id, 0)

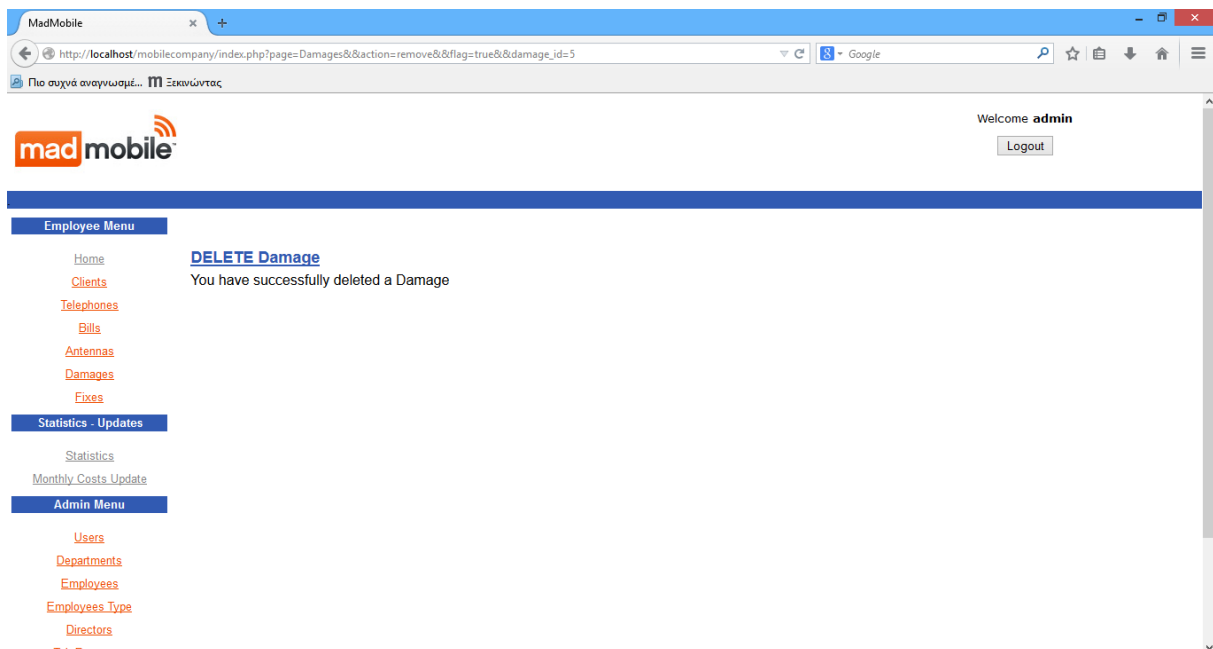
8.14.3 Διαγραφή Βλάβης

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής της συγκεκριμένης βλάβης, περνώντας το id της ως παράμετρο.



Εικόνα 69: Διαγραφή Βλάβης 1



Εικόνα 70: Διαγραφή Βλάβης 2

SQL query:

delete from damages where damage_id=\$StrtsID

8.15 Επιδιορθώσεις

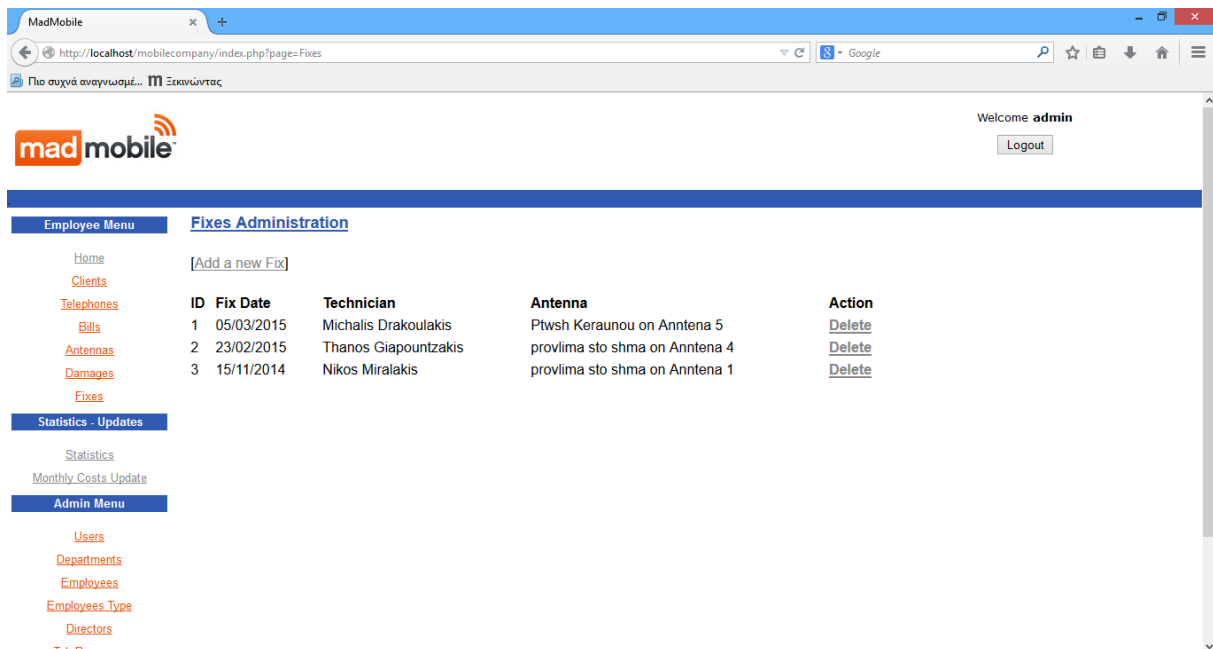
8.15.1 Αρχική Οθόνη Επιδιορθώσεων

Screenshots:

Στην αρχική οθόνη των κεραιών εμφανίζεται η λίστα με τις υπάρχοντες επιδιορθώσεις και δίπλα σε κάθε μία, οι σύνδεσμοι Διαγραφής (Delete) για κάθε μία από αυτές.

Εάν δεν έχει δημιουργηθεί κάποια επιδιόρθωση, τότε εμφανίζεται μήνυμα μη ύπαρξης επιδιόρθωσης.

Επίσης, στο πάνω μέρος της αρχικής οθόνης επιδιορθώσεων, υπάρχει σύνδεσμος που μας οδηγεί στη φόρμα δημιουργίας επιδιόρθωσης.



ID	Fix Date	Technician	Antenna	Action
1	05/03/2015	Michalis Drakoulakis	Ptwsh Keraunou on Anntena 5	Delete
2	23/02/2015	Thanos Giapountzakis	provlima sto shma on Anntena 4	Delete
3	15/11/2014	Nikos Miralakis	provlima sto shma on Anntena 1	Delete

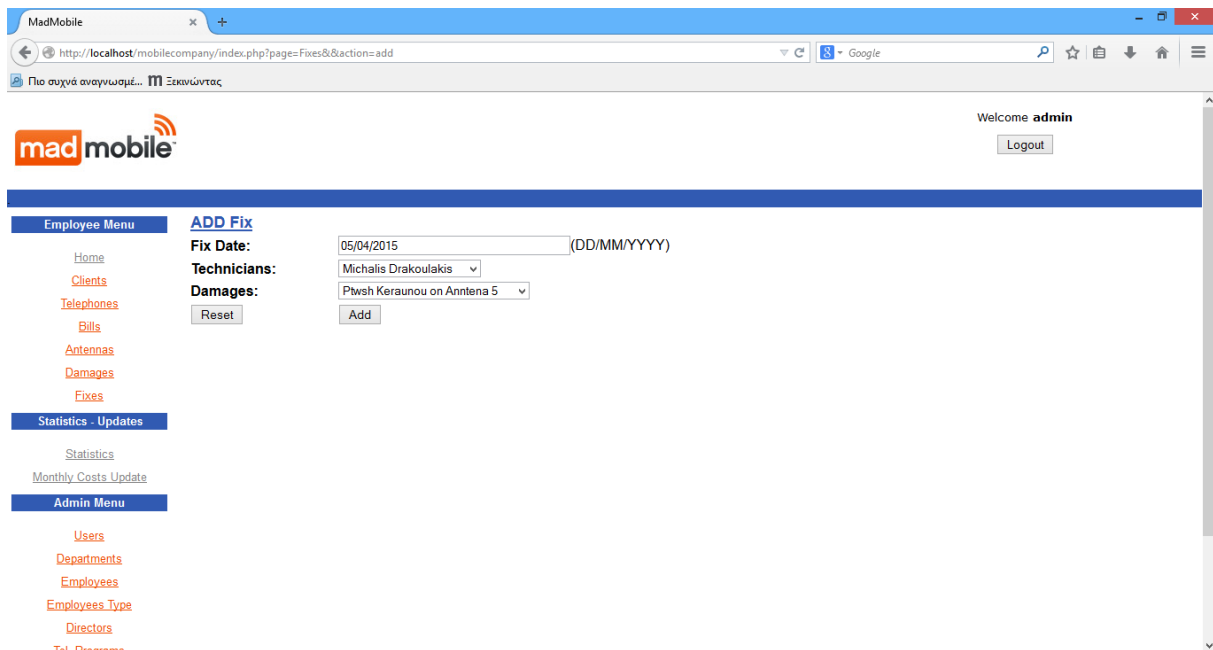
Εικόνα 71: Αρχική οθόνη Επιδιορθώσεων

8.15.2 Δημιουργία Επιδιόρθωσης

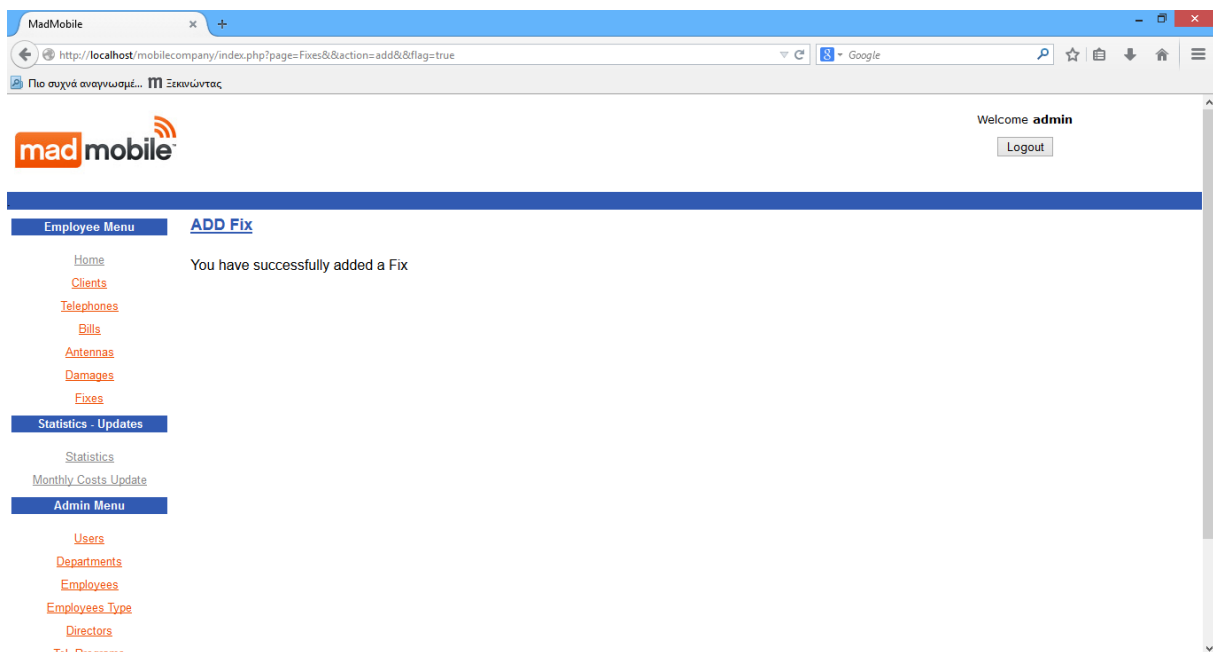
Screenshots:

Στην οθόνη Δημιουργίας Επιδιόρθωσης, εμφανίζεται η φόρμα με τα απαραίτητα πεδία που πρέπει να συμπληρωθούν ώστε να δημιουργηθεί μία Επιδιόρθωση.

- Η φόρμα δημιουργίας επιδιόρθωσης έχει λίστα επιλογής με ΟΛΕΣ τις διαθέσιμες βλάβες.



Εικόνα 72: Φόρμα Προσθήκης Νέας Επιδιόρθωσης



Εικόνα 73: Ολοκλήρωση προσθήκης Νέας Επιδιόρθωσης

SQL query:

Δημιουργία Επιδιόρθωσης

insert into FixAntennasDamages (fix_date, tid, damage_id) values ('\$fix_date', \$tid, \$damage_id)

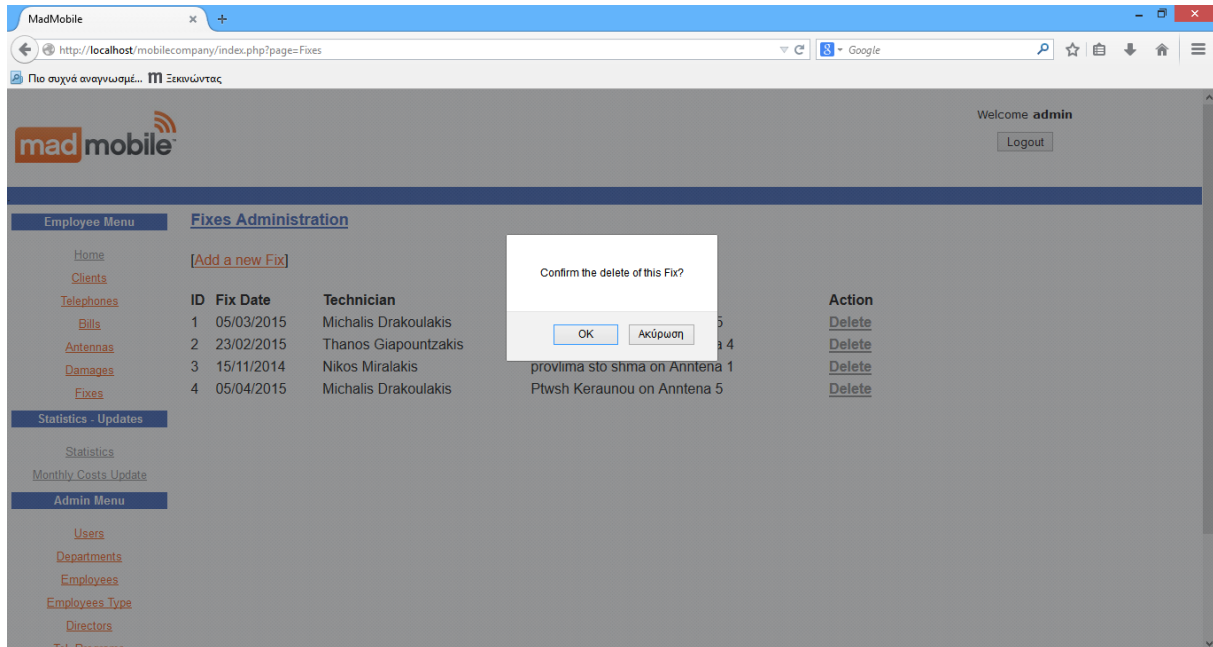
Ανανέωση Βλάβης ως επιδιορθωμένη

update damages set fixed_or_not=1 where damage_id=\$damage_id

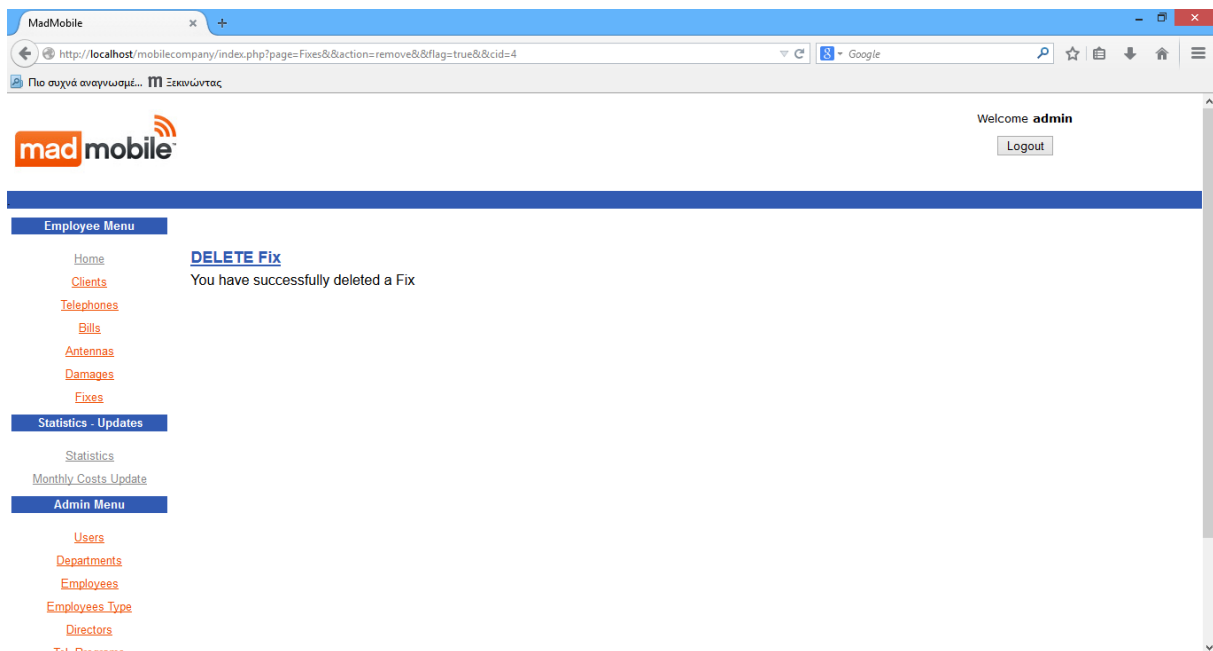
8.15.3 Διαγραφή Επιδιόρθωσης

Screenshots:

Κάθε σύνδεσμος Delete, αντιστοιχεί στην διαδικασία Διαγραφής της συγκεκριμένης επιδιόρθωσης, περνώντας το id της ως παράμετρο.



Εικόνα 74: Διαγραφή Επιδιόρθωσης 1



Εικόνα 75: Διαγραφή Επιδιόρθωσης 2

SQL query:

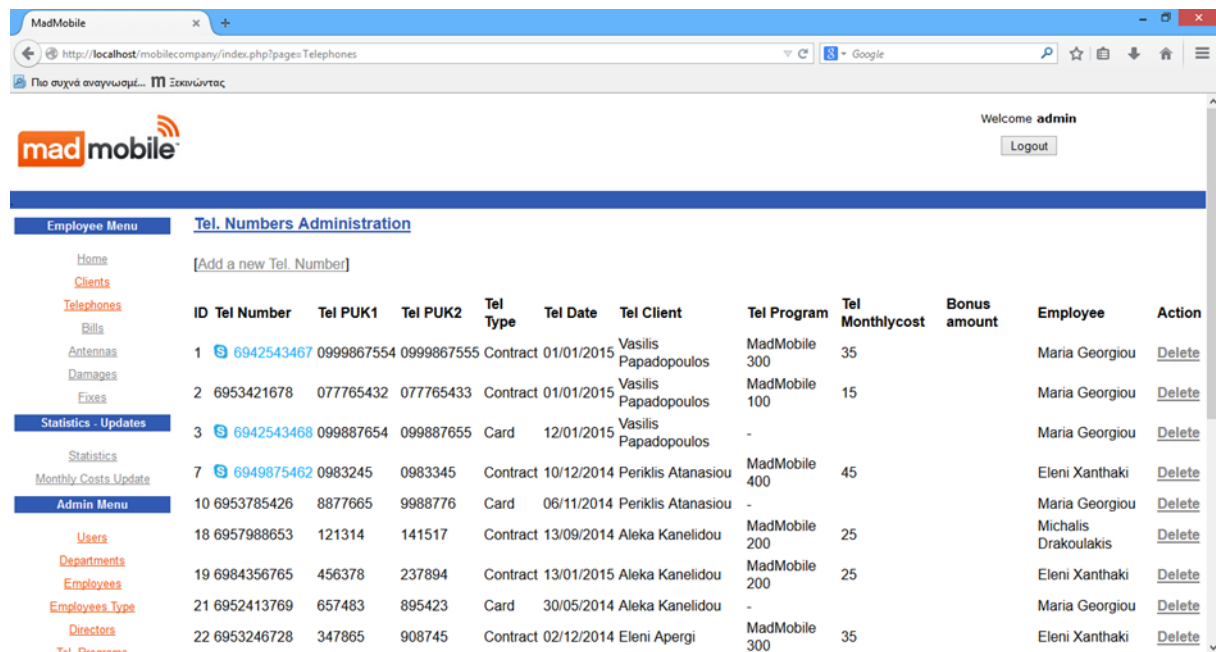
delete from FixAntennasDamages where fix_id=\$StrtsID

8.16 Ετήσια Ανανέωση Παγιών - Επιδόσεις

Screenshots:

Η επιλογή αυτή σε ένα σύστημα θα γίνεται μετά την ολοκλήρωση ενός συγκεκριμένου διαστήματος στο οποίο θα πρέπει να ανανεωθούν κάποια πάγια εφόσον συμπληρώσουν ένα χρόνο σε πρόγραμμα συμβολαίου. Στα πλαίσια της πτυχιακής, επειδή το διάστημα αυτό, το προσομοιώνουμε με την εισαγωγή μιας μελλοντικής ημερομηνίας (πάνω από ένα έτος) σε σύγκριση με την ημερομηνία δημιουργίας του συμβολαίου που έχουμε ήδη εισάγει στο σύστημα.

ΠΡΙΝ – ΠΙΝΑΚΑΣ ΤΗΛΕΦΩΝΙΚΩΝ ΑΡΙΘΜΩΝ



ID	Tel Number	Tel PUK1	Tel PUK2	Tel Type	Tel Date	Tel Client	Tel Program	Tel Monthlycost	Bonus amount	Employee	Action
1	6942543467	0999867554	0999867555	Contract	01/01/2015	Vasilis Papadopoulos	MadMobile 300	35		Maria Georgiou	Delete
2	6953421678	077765432	077765433	Contract	01/01/2015	Vasilis Papadopoulos	MadMobile 100	15		Maria Georgiou	Delete
3	6942543468	099887654	099887655	Card	12/01/2015	Vasilis Papadopoulos	-			Maria Georgiou	Delete
7	6949875462	0983245	0983345	Contract	10/12/2014	Periklis Atanasiou	MadMobile 400	45		Eleni Xanthaki	Delete
10	6953785426	8877665	9988776	Card	06/11/2014	Periklis Atanasiou	-			Maria Georgiou	Delete
18	6957988653	121314	141517	Contract	13/09/2014	Aleka Kanelidou	MadMobile 200	25		Michalis Drakoulakis	Delete
19	6984356765	456378	237894	Contract	13/01/2015	Aleka Kanelidou	MadMobile 200	25		Eleni Xanthaki	Delete
21	6952413769	657483	895423	Card	30/05/2014	Aleka Kanelidou	-			Maria Georgiou	Delete
22	6953246728	347865	908745	Contract	02/12/2014	Eleni Aperi	MadMobile 300	35		Eleni Xanthaki	Delete

Εικόνα 76: Πίνακας Τηλ.Αριθμών ΠΡΙΝ την ετήσια ανανέωση

ΕΙΣΑΓΩΓΗ ΜΕΛΛΟΝΤΙΚΗΣ ΗΜΕΡΟΜΗΝΙΑΣ

MadMobile

http://localhost/mobilecompany/index.php?page=MonthlyCostsupdate&&action=add&&flag=true

Welcome admin

Logout

Employee Menu [Set Current Date](#)

Home
Clients
Telephones
Bills
Antennas
Damages
Fixes

Statistics - Updates

Statistics
Monthly Costs Update

Admin Menu

Users
Departments
Employees
Employees_Type
Directors
Tel Programs

You have successfully updated Monthly Costs

Εικόνα 77: Εισαγωγή μελλοντικής ημερομηνίας για ετήσια ανανέωση Παγίων

ΜΕΤΑ – ΠΙΝΑΚΑΣ ΑΣΦΑΛΙΣΤΡΩΝ

MadMobile

http://localhost/mobilecompany/index.php?page=Telephones

Employee Menu [Tel. Numbers Administration](#)

Home
Clients
Telephones
Bills
Antennas
Damages
Fixes

Statistics - Updates

Statistics
Monthly Costs Update

Admin Menu

Users
Departments
Employees
Employees_Type
Directors
Tel Programs
FamilyMembers

[Add a new Tel. Number]

ID	Tel Number	Tel PUK1	Tel PUK2	Tel Type	Tel Date	Tel Client	Tel Program	Tel Monthlycost	Bonus amount	Employee	Action
1	6942543467	0999867554	0999867555	Contract	01/01/2015	Vasilis Papadopoulos	MadMobile 300	18.6004	353.4	Maria Georgiou	Delete
2	6953421678	077765432	077765433	Contract	01/01/2015	Vasilis Papadopoulos	MadMobile 100	7.97161	353.4	Maria Georgiou	Delete
3	6942543468	099887654	099887655	Card	12/01/2015	Vasilis Papadopoulos	-	-	-	Maria Georgiou	Delete
7	6949875462	0983245	0983345	Contract	10/12/2014	Periklis Atanasiou	MadMobile 400	23.9148	342	Eleni Xanthaki	Delete
10	6953785426	8877665	9988776	Card	06/11/2014	Periklis Atanasiou	-	-	-	Maria Georgiou	Delete
18	6957988653	121314	141517	Contract	13/09/2014	Aleka Kanelidou	MadMobile 200	11.9574	322.2	Michalis Drakoulakis	Delete
19	6984356765	456378	237894	Contract	13/01/2015	Aleka Kanelidou	MadMobile 200	14.7622	322.8	Eleni Xanthaki	Delete
21	6952413769	657483	895423	Card	30/05/2014	Aleka Kanelidou	-	-	-	Maria Georgiou	Delete
22	6953246728	347865	908745	Contract	02/12/2014	Eleni Apergi	MadMobile 300	18.6004	342	Eleni Xanthaki	Delete
23	6945321799	345789	980743	Contract	08/11/2014	Eleni Apergi	MadMobile 400	21.5234	339	Maria Georgiou	Delete
24	6956735228	234577	234588	Card	12/01/2015	Eleni Apergi	-	-	-	Eleni Xanthaki	Delete
25	6943327654	190187	180198	Contract	04/01/2015	Periklis Atanasiou	MadMobile 200	14.7622	322.8	Giorgos Paraskakis	Delete

Εικόνα 78: Πίνακας Τηλ.Αριθμών META την ετήσια ανανέωση

SQL query:

//ΕΠΙΛΕΓΩ ΤΗΝ ΗΜΕΡΟΜΗΝΙΑ ΚΑΙ ΤΑ ID ΤΩΝ ΤΗΛΕΦΩΝΙΚΩΝ ΑΡΙΘΜΩΝ ΜΕ ΣΥΜΒΟΛΑΙΟ

```
select tel_date, tel_id from telephones where tel_type=1
```

//ΓΙΑ ΚΑΘΕ ΑΡΙΘΜΟ

```
while( $temp3=mysql_fetch_row($db3)){
```

//ΒΡΙΣΚΩ ΤΗ ΔΙΑΦΟΡΑ ΤΗΣ ΜΕΛΛΟΝΤΙΚΗΣ ΗΜΕΡΟΜΗΝΙΑΣ ΜΕ ΤΗΝ ΗΜΕΡΟΜΗΝΙΑ ΠΟΥ

//ΔΗΜΙΟΥΡΓΗΘΗΚΕ ΤΟ ΑΣΦΑΛΙΣΤΡΟ

```
    $date1=strtotime($temp3[0]);
```

```
    $date2=strtotime($current_date);
```

```
    $diff=$date2-$date1;
```

```
    $x=floor($diff/3600/24);
```

//ΑΝ Η ΔΙΑΦΟΡΑ ΕΙΝΑΙ 1, 2 Ή 3 ΧΡΟΝΙΑ, ΚΑΝΩ ΤΗΝ ΑΝΑΛΟΓΗ ΜΕΙΩΣΗ ΣΤΟ ΠΑΓΙΟ

```
    if($x>=365 && $x<730){
```

```
        $query4=" update telephones set p_monthlycost=p_monthlycost-  
10*p_monthlycost/100 where tel_id=$temp3[1]";
```

```
        $db4=mysql_query($query4);
```

//ΚΑΙ ΥΠΟΛΟΓΙΖΩ ΤΗΝ ΕΠΙΔΟΤΗΣΗ ΠΟΥ ΔΙΚΑΙΟΥΤΑΙ

```
    $wanted_start=$temp3[0];
```

```
    $year_start=substr($temp3[0], 0, 4);
```

```
    $month_start=substr($temp3[0], 5, 2);
```

```
    $day_start=substr($temp3[0], 8, 2);
```

```
    $year_bonus=$year_start+1;
```

```
    $wanted_end="$year_bonus-$month_start-$day_start";
```

```
    $query5="select sum(a.bamount) from Bills a, Telephones b where  
a.tel_id=b.tel_id and a.bpubdate between '$wanted_start' and '$wanted_end'";
```

```
    $db5=mysql_query($query5);
```

```
    $temp5=mysql_fetch_row($db5);
```

```
    $bonus_amount = $temp5[0] * 0.12;
```

```

        $query6="update telephones set epidotisi=$bonus_amount where
tel_id=$temp3[1]";
        $db6=mysql_query($query6);
    }
    else if($x>=730 && $x<1095){
        $query4="update telephones set p_monthlycost=p_monthlycost-
20*p_monthlycost/100 where tel_id=$temp3[1]";
        $db4=mysql_query($query4);
    }
    else if($x>=1095){
        $query4="update telephones set p_monthlycost=p_monthlycost-
30*p_monthlycost/100 where tel_id=$temp3[1]";
        $db4=mysql_query($query4);
    }
}
}

```

8.17 Στατιστικά

8.17.1 Ετήσια Συγκεντρωτική Εσόδων ανα Τμήμα ανά μήνα

Screenshots:

TOTAL	MadMobile (Κέν)	MadMobile (Αλ)	MadMobile (Θέπ)
1/2015	170	85	0
2/2015	295	85	0
3/2015	295	85	0
4/2015	295	85	0
5/2015	295	85	0
6/2015	295	85	0
7/2015	95	0	0
8/2015	95	0	0
9/2015	95	0	0
10/2015	95	0	0
11/2015	95	0	0
12/2015	95	0	0
TOTAL	2215	510	0

Εικόνα 79: Ετήσια Συγκεντρωτική Εσόδων ανά μήνα

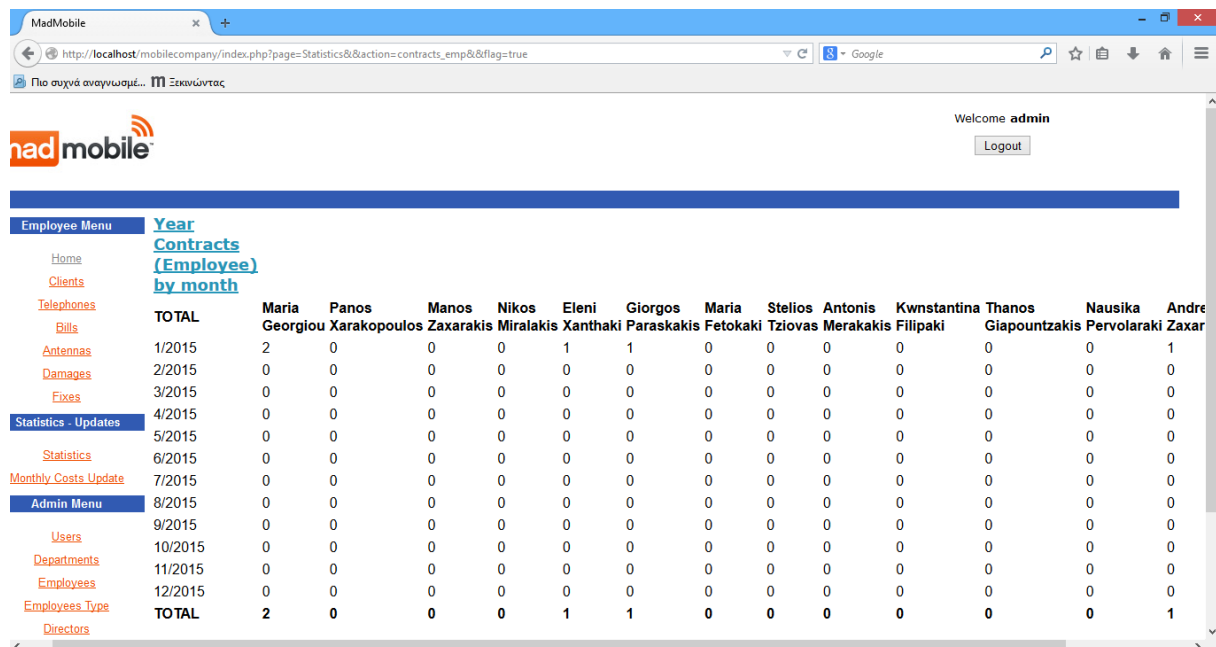
SQL query:

-Μέσα σε επανάληψη 12 φορές, μία για κάθε μήνα

```
select sum(bamount) from Bills where tel_id in (  
    select tel_id from Telephones where tel_type =1 and eid in (  
        select eid from Employees where did=$temp[0]))  
and brubdate between '$wanted_start' and '$wanted_end'
```

8.17.2 Ετήσια Συγκεντρωτική Συμβολαίων ανά Υπάλληλο ανά μήνα

Screenshots:



Year	Contracts (Employee) by month	Maria Georgiou	Panos Xarakopoulos	Manos Zaxarakis	Nikos Miralakis	Eleni Xanthaki	Giorgos Paraskakis	Maria Fetokaki	Stelios Tziovas	Antonis Merakakis	Kwnstantina Filipaki	Thanos Giapountzakis	Nausika Pervolaraki	Andre Zaxar
TOTAL		2	0	0	0	1	1	0	0	0	0	0	0	1
1/2015		2	0	0	0	1	1	0	0	0	0	0	0	1
2/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
3/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
4/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
5/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
6/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
7/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
8/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
9/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
10/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
11/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
12/2015		0	0	0	0	0	0	0	0	0	0	0	0	0
TOTAL		2	0	0	0	1	1	0	0	0	0	0	0	1

Εικόνα 80: Ετήσια Συγκεντρωτική Συμβολαίων ανά Υπάλληλο ανά μήνα

SQL query:

-Μέσα σε επανάληψη 12 φορές, μία για κάθε μήνα

```
$x=0;
```

```
//ΕΠΙΛΕΓΩ ΤΑ IDS ΟΛΩΝ ΤΩΝ ΥΠΑΛΛΗΛΩΝ
```

```
$queryx="select eid from Employees order by eid asc";
```

```
$dbx=mysql_query($queryx);
```

```
//ΓΙΑ ΚΑΘΕ ΕΝΑΝ ΑΠΟ ΑΥΤΟΥΣ
```

```
while($temp[0]=mysql_fetch_row($dbx)){
```

//ΜΕΤΡΩ ΤΑ ΣΥΜΒΟΛΑΙΑ ΤΟΥ ΚΑΘΕ ΥΠΑΛΛΗΛΟΥ ΤΟΝ ΚΑΘΕ ΜΗΝΑ

```
$query=" select count(a.tel_id) from Telephones a, Employees b
      where a.eid=b.eid
      and b.eid=$tempx[0]
      and a.tel_type=1
      and a.tel_date between '$wanted_start' and '$wanted_end';"
```

```
$db=mysql_query($query);
```

```
$temp=mysql_fetch_row($db);
```

```
echo("<td>$temp[0]</td>");
```

//ΔΗΜΙΟΥΡΓΩ ΤΟ ΠΙΝΑΚΑ ΑΘΡΟΙΣΜΑΤΩΝ ΓΙΑ ΤΑ ΣΥΜΒΟΛΑΙΑ ΤΟΥ ΚΑΘΕ ΥΠΑΛΛΗΛΟΥ

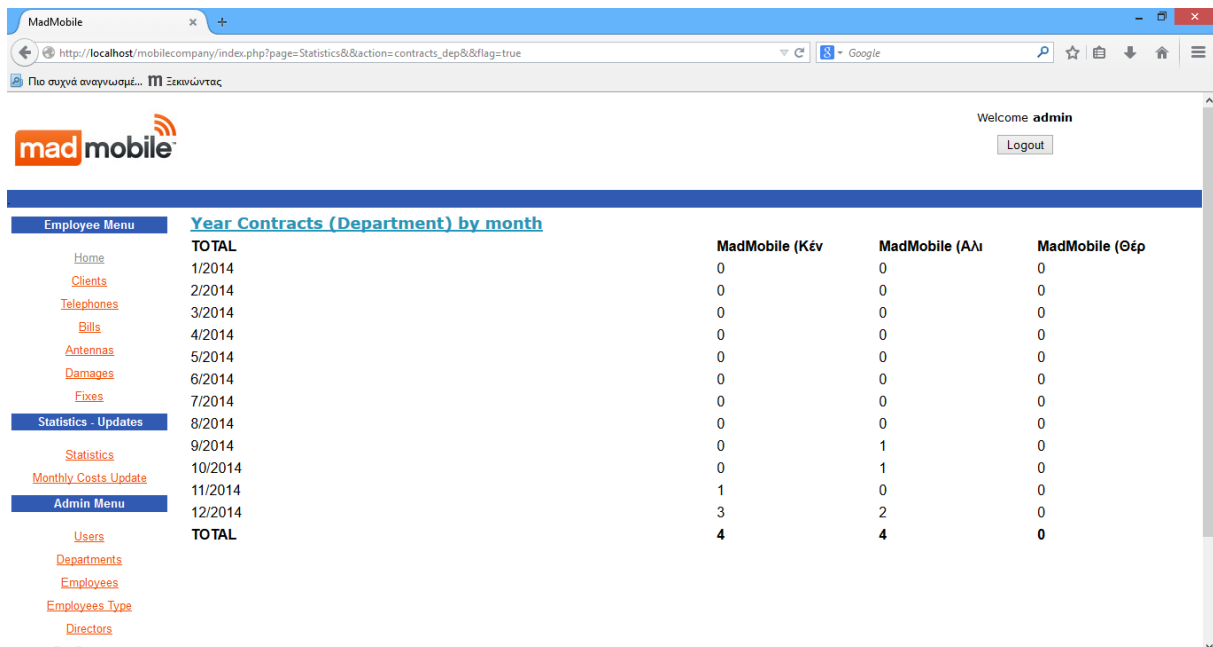
```
$sum[$x]=$sum[$x]+$temp[0];
```

```
$x++;
```

```
}
```

8.17.3 Ετήσια Συγκεντρωτική Συμβολαίων ανά Τμήμα ανά μήνα

Screenshots:



Employee Menu	Year Contracts (Department) by month	MadMobile (Κέν)	MadMobile (Αλι)	MadMobile (Θέρ)
Home	TOTAL	0	0	0
Clients	1/2014	0	0	0
Telephones	2/2014	0	0	0
Bills	3/2014	0	0	0
Antennas	4/2014	0	0	0
Damages	5/2014	0	0	0
Fixes	6/2014	0	0	0
Statistics - Updates	7/2014	0	0	0
Statistics	8/2014	0	0	0
Monthly Costs Update	9/2014	0	1	0
Admin Menu	10/2014	0	1	0
Users	11/2014	1	0	0
Departments	12/2014	3	2	0
Employees	TOTAL	4	4	0
Employees Type				
Directors				
Tel Parameters				

Εικόνα 81: Ετήσια Συγκεντρωτική Συμβολαίων ανά Τμήμα ανά μήνα

SQL query:

```
$x=0;
//ΕΠΙΛΕΓΩ ΤΑ IDS ΟΛΩΝ ΤΩΝ ΤΜΗΜΑΤΩΝ
$queryx="select did from Departments order by did asc";
$dbx=mysql_query($queryx);
//ΓΙΑ ΚΑΘΕ ΕΝΑΝ ΑΠΟ ΑΥΤΑ
while($tempx=mysql_fetch_row($dbx)){
    //ΜΕΤΡΑΩ ΤΑ ΣΥΜΒΟΛΑΙΑ ΤΟΥ ΚΑΘΕ ΤΜΗΜΑΤΟΣ ΤΟΝ ΚΑΘΕ ΜΗΝΑ
    $query=" select count(tel_id) from Telephones
            where eid in ( select eid from Employees where did=$tempx[0])
            and tel_type=1
            and tel_date between '$wanted_start' and '$wanted_end'";
    $db=mysql_query($query);
    $temp=mysql_fetch_row($db);
    //ΔΗΜΙΟΥΡΓΩ ΤΟ ΠΙΝΑΚΑ ΑΘΡΟΙΣΜΑΤΩΝ ΓΙΑ ΤΑ ΣΥΜΒΟΛΑΙΑ ΤΟΥ ΚΑΘΕ ΤΜΗΜΑΤΟΣ
    echo("<td>$temp[0]</td>");
    $sum[$x]=$sum[$x]+$temp[0];
    $x++;
}
```

8.17.4 Ετήσια Συγκεντρωτική Καρτοτηλεφώνων ανά Τμήμα ανά μήνα

Screenshots:

Employee Menu	Year Cards (Department) by month	MadMobile (Κέβ)	MadMobile (Αλι)	MadMobile (Θέπ)
Home	TOTAL	2	0	0
Clients	1/2014	0	0	0
Telephones	2/2014	0	0	0
Bills	3/2014	0	0	0
Antennas	4/2014	0	0	0
Damages	5/2014	1	0	0
Fixes	6/2014	0	0	0
Statistics - Updates	7/2014	0	0	0
Statistics	8/2014	0	0	0
Monthly Costs Update	9/2014	0	0	0
Admin Menu	10/2014	0	0	0
Users	11/2014	1	0	0
Departments	12/2014	0	0	0
Employees	TOTAL	2	0	0
Employees Type				
Directors				
Tel Programs				

Εικόνα 82: Ετήσια Συγκεντρωτική Καρτοτηλεφώνων ανά Τμήμα ανά μήνα

SQL query:

```
$x=0;
```

```
//ΕΠΙΛΕΓΩ ΤΑ IDS ΟΛΩΝ ΤΩΝ ΤΜΗΜΑΤΩΝ
```

```
$queryx="select did from Departments order by did asc";
```

```
$dbx=mysql_query($queryx);
```

```
//ΓΙΑ ΚΑΘΕ ΕΝΑΝ ΑΠΟ ΑΥΤΑ
```

```
while($tempx=mysql_fetch_row($dbx)){
```

```
    //ΜΕΤΡΩ ΤΑ ΚΑΡΤΟΤΗΛΕΦΩΝΑ ΚΑΘΕ ΤΜΗΜΑΤΟΣ ΑΝΑ ΜΗΝΑ
```

```
    $query=" select count(tel_id) from Telephones
```

```
        where eid in ( select eid from Employees where did=$tempx[0])
```

```
        and tel_type=2
```

```
        and tel_date between '$wanted_start' and '$wanted_end'";
```

```
    $db=mysql_query($query);
```

```
    $temp=mysql_fetch_row($db);
```

```
        echo("<td>$temp[0]</td>");
```

```
        $sum[$x]=$sum[$x]+$temp[0];
```

```
        $x++;
```

```
}
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. **Wikipedia.** Code Examples on Ajax, CGI, Java, Javascript, Actionscript, Asp, Php, C++ and .Net. www.wikipedia.com.
2. *Some Trends in Web Application Development.* **Jazayeri, Mehdi.** 2007. Future of Software Engineering. σ. International conference on Software Engineering.
3. **Google.** Google Chrome: Μια νέα προσέγγιση στο πρόγραμμα περιήγησης. <http://www.google.com/chrome/intl/el-GR/why.htm>.
4. **Baldwin, Richard.** Similarities and Differences between Java and C++. 1997. <http://www.dickbaldwin.com/java/Java008.htm>.
5. **Wheeler, Scott και Marshall, Brian.** *LScript Reference manual.* s.l. : Newtek, 2002.
6. **Dawson, Bruce.** Game Scripting in Python. (Games Developer Conference 2002). http://www.gamasutra.com/features/20020821/dawson_01.htm.
7. **Sweeny, Tim.** UnrealScript language Reference (Epic Megagames).
 - a. <http://unreal.epicgames.com/UnrealScript.htm>.
8. **W3 Schools.** On-line Tutorials and Examples on XML, Javascript, PHP, Asp. <http://www.w3schools.com/>.
9. *A comparison of VBScript, Javascript and Jscript.* **Kailash, Chandra, Suhani, Chandra Sapana και Suhana, Chandra Shyamal.** Journal of Computing Sciences in Colleges.
10. *Server-centric Web frameworks: An overview.* **Reahl, Iwan Vosloo και Kourie, Derrick G.** ACM Computing Surveys (CSUR).
11. **The PHP Group.** Php language Documentation. <http://www.php.net/docs.php>.
12. *An online bookstore using JSP technology.* **Hughes, Jose.** s.l. : ACM-SE 43: , Τόμ. Proceedings of the 43rd annual Southeast regional conference, Volume 2.
13. *Java servlets and Java server pages (JSP).* **Gordon, Aaron.** 2002, Journal of Computing Sciences in Colleges.
14. **Microsoft Inc.** ASP.NET reference. *Microsoft Developer Network.*
 - a. <http://msdn.microsoft.com/en-us/asp.net/default.aspx>.
15. **Adobe Inc.** Adobe Flash Resources and Reference.
 - a. <http://www.adobe.com/support/documentation/en/flash/>.
16. *ActionScript: a gentle introduction to programming.* **Crawford, Stewart και Boese, Elizabeth.**
 - a. 2006, Journal of Computing Sciences in Colleges.
17. **Ecma.** Ecma 2007. ECMAScript Edition 4 specification wiki. <http://wiki.ecmascript.org>. 2007. <http://www.ecma-international.org/publications/standards/Ecma-262.htm>.
18. **Adobe Inc.** Adobe Flex Resources & Reference.
 - a. <http://www.adobe.com/support/documentation/en/flex/>.