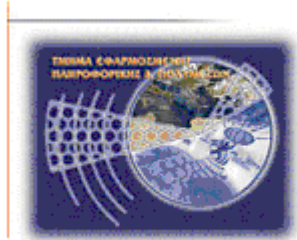




Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

**Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων**



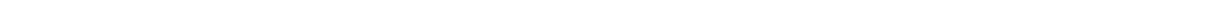
Πτυχιακή:

**‘Πληροφοριακό Σύστημα για εταιρία Κινητής
Τηλεφωνίας’**

Επιτηρητής: Κ. Παπαδάκης

ΑΡΓΥΡΩ ΓΙΑΝΝΑΡΑΚΗ (ΑΜ:2318)

Ηράκλειο – 01/02/12





Ευχαριστίες

Θα ήθελα να ευχαριστήσω τους ανθρώπους που με βοήθησαν αυτά τα χρόνια να αποδείξω ότι μπορώ και εγώ να είμαι ένα χρήσιμο μέλος σε αυτή την κοινωνία, δίνοντας μου τις σωστές αξίες στην ζωή μου, δηλαδή τους γονείς μου, που ήταν δίπλα μου και με στήριζαν όταν εγώ απογοητευόμουν.

Τους φίλους μου γιατί ήταν δίπλα μου σε κάθε προσπάθεια μου να με βοηθήσουν και να με στηρίξουν σε όλη την φοιτητική μου πορεία και τέλος στον καθηγητή που με βοήθησε να ολοκληρώσω την εκπαίδευση μου και τις γνώσεις μου με το να ολοκληρώσω το πτυχιακή μου και να πάρω το πτυχίο μου, τον κύριο Παπαδάκη.



Abstract:

At the following project i created a web- based page connects to a database and is used to manage a phone company. The main technologies used to achieve this goal are HTML, PHP and JavaScript for the development of my website and SQL for database management. My site connects directly the companies in Crete so they can learn and exchange information's about employees, clients, phone numbers, damages, accounts and the packets Association.



Σύνοψη:

Στο έργο που δημιούργησα μια web-based σελίδα συνδέεται με μια βάση δεδομένων και χρησιμοποιείται για να διαχειριστεί μια εταιρεία τηλεφωνίας. Οι κύριες τεχνολογίες που χρησιμοποιούνται για την επίτευξη αυτού του στόχου είναι HTML, PHP και τη Javascript για την ανάπτυξη της ιστοσελίδας και SQL μου για τη διαχείριση της βάσης δεδομένων. Το site μου συνδέει άμεσα τις εταιρείες της Κρήτης, ώστε να μπορούν να μάθουν και να ανταλλάσσουν πληροφορίες σχετικά με τους εργαζομένους, τους πελάτες, τους αριθμούς τηλεφώνου, ζημιές, τους λογαριασμούς και τα πακέτα σύνδεσης.



Πίνακας Περιεχομένων

Ευχαριστίες	i
Abstract	ii
Σύνοψη	iii
Πίνακας Περιεχομένων	iv
Πίνακας Εικόνων.....	ix
ΚΕΦΑΛΑΙΟ 1: Εισαγωγικά	1
1.1 Περίληψη.....	1
1.3 Κίνητρο για την Διεξαγωγή της Εργασίας	1
1.3 Σκοπός και Στόχοι Εργασίας.....	1
1.4 Δομή Εργασίας	2
ΚΕΦΑΛΑΙΟ 2: Μεθοδολογία Υλοποίησης	3
2.1 Μέθοδος Ανάλυσης & Ανάπτυξης Πτυχιακής.....	3
2.2 Θεωρίες	3
2.2.1. Διαδίκτυο(Internet).....	3
2.2.2 Παγκόσμιος Ιστός (World Wide Web -WWW).....	3
2.2.3 Κατηγορίες Ιστοσελίδων	4
ΚΕΦΑΛΑΙΟ 3 : Background	5
3.1 Τεχνολογίες διαδικτύου	5
3.1.1 HTML (Hypertext Markup Language).....	5
3.1.2 PHP (Hypertext Preprocessor).....	7
3.1.3 Perl.....	11
3.1.4 Python	13
3.1.5 JavaScript.....	14
3.1.6 JSP (Java Server Pages).....	17
3.1.7 Java Servlet.....	20
3.2 Τεχνολογίες Βάσεων Δεδομένων	24
3.2.1 MySQL	24
3.2.2 Oracle Database	25
3.2.3 PostgreSQL	26
3.3 Επιλογή Τεχνολογίας.....	29
ΚΕΦΑΛΑΙΟ 4 : Προγράμματα που θα χρειαστώ για την κατασκευή της ιστοσελίδα μου.....	31



4.1 Photoshop	31
4.2 Apache HTTP εξυπηρετητής	33
4.3 Dreamweaver	33
ΚΕΦΑΛΑΙΟ 5 : Σχεδιασμός Βάσεων Δεδομένων	35
5.1 Εισαγωγικά	35
5.1.1 Βασικές έννοιες	35
5.1.2 Ιστορική Εξέλιξη των Βάσεων Δεδομένων	35
5.1.3 Αρχιτεκτονική τριών επιπέδων	36
5.1.4 Μορφές Βάσεων Δεδομένων	38
5.1.5 Στοιχεία Βάσεων Δεδομένων	40
5.1.6 Συνηθισμένα προβλήματα που συναντώνται στις Βάσης Δεδομένων.....	41
5.2 Ανάλυση των Απαιτήσεων	42
5.2.1 Περιγραφή Συστήματος	42
5.2.2 Λειτουργικές απαιτήσεις (operational/functional specifications).....	42
5.2.3 Απαιτήσεις λογισμικού	44
5.3 Entity- Relationship Model.....	44
5.3.1 Ορισμός E-R model	44
5.3.2 Βασικά Δομικά Στοιχεία του E-R μοντέλου	44
5.3.3 Από E – R μοντέλο σε πίνακα (κανονικοποίηση)	46
ΚΕΦΑΛΑΙΟ 6 : SQL Κώδικες	51
6.1 Εξαγωγή SQL κώδικα	51
6.2 Εξαγωγή του κώδικα SQL κάθε υποκαταστήματος	51
6.2.1 Εξαγωγή του κώδικα SQL της βάσης των Χανίων	54
6.2.2 Εξαγωγή του κώδικα SQL της βάσης του Ηρακλείου.....	56
6.2.3 Εξαγωγή του κώδικα SQL της βάσης του Ρεθύμνου.....	57
6.2.4 Εξαγωγή του κώδικα SQL της βάσης του Λασιθίου	59
ΚΕΦΑΛΑΙΟ 7 : Λειτουργίες σε PHP.....	63
7.1 PHP Κώδικας του υποκαταστήματος των Χανίων	63
7.1.1 PHP Κώδικας για την σελίδα των Υπαλλήλων.....	63
7.1.2 PHP Κώδικας για την σελίδα των Πελατών.....	63
7.1.3 PHP Κώδικας για την σελίδα των Λογαριασμών.....	65
7.1.4 PHP Κώδικας για την σελίδα στις Βλάβες	66
7.1.5 PHP Κώδικας για την σελίδα στις Κεραιές	66
7.1.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων.....	66



7.2 PHP Κώδικας του κάθε υποκαταστήματος του Ηρακλείου	66
7.2.1 PHP Κώδικας για την σελίδα των Υπαλλήλων.....	66
7.2.2 PHP Κώδικας για την σελίδα των Πελατών.....	67
7.2.3 PHP Κώδικας για την σελίδα των Λογαριασμών.....	68
7.2.4 PHP Κώδικας για την σελίδα στις Βλάβες	69
7.2.5 PHP Κώδικας για την σελίδα στις Κεραίες	70
7.2.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων.....	70
7.3 PHP Κώδικας του κάθε υποκαταστήματος του Λασιθίου	70
7.3.1 PHP Κώδικας για την σελίδα των Υπαλλήλων.....	70
7.3.2 PHP Κώδικας για την σελίδα των Πελατών.....	71
7.3.3 PHP Κώδικας για την σελίδα των Λογαριασμών.....	72
7.3.4 PHP Κώδικας για την σελίδα στις Βλάβες	73
7.3.5 PHP Κώδικας για την σελίδα στις Κεραίες	73
7.3.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων.....	73
7.4 PHP Κώδικας του κάθε υποκαταστήματος του Ρεθύμνου	74
7.4.1 PHP Κώδικας για την σελίδα των Υπαλλήλων.....	74
7.4.2 PHP Κώδικας για την σελίδα των Πελατών.....	74
7.4.3 PHP Κώδικας για την σελίδα των Λογαριασμών.....	75
7.4.4 PHP Κώδικας για την σελίδα στις Βλάβες	76
7.4.5 PHP Κώδικας για την σελίδα στις Κεραίες	77
7.4.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων.....	77
ΚΕΦΑΛΑΙΟ 8 : HTML Σελίδες	79
8.1 HTML σελίδα της αρχικής σελίδας του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s.....	79
8.2 HTML αρχικής σελίδας για το κάθε υποκαταστήμα της κινητής τηλεφωνίας των Χανίων, του Ρεθύμνου, του Ηρακλείου, του Λασιθίου με την επωνυμία S@louf@s.....	79
8.2.1 HTML σελίδα για την σελίδα των Υπαλλήλων	83
8.2.2 HTML σελίδα για την σελίδα των Πελατών.....	84
8.2.3 HTML σελίδα για την σελίδα των Λογαριασμών	85
8.2.4 HTML σελίδα για την σελίδα στις Βλάβες	86
8.2.5 HTML σελίδα για την σελίδα στις Κεραίες	87
8.2.6 HTML σελίδα για την σελίδα Πακέτων σύνδεσης.....	88
8.2.7 HTML σελίδα για την σελίδα των Αριθμών Τηλεφώνων	89
8.3 HTML σελίδα της Επικοινωνίας του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s	90
ΚΕΦΑΛΑΙΟ 9 : Επίλογος.....	93



9.1 Αποτελέσματα.....	93
9.2 Συμπεράσματα.....	93
ΚΕΦΑΛΑΙΟ 10 : HTML Κώδικας	95
10.1 HTML Κώδικας της αρχικής σελίδας του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s.....	95
10.2 HTML Κώδικας της αρχικής του κάθε υποκαταστήματος.....	97
10.2.1 HTML Κώδικας καταστήματος των Χανίων.....	97
10.2.1.1 HTML Κώδικας για την σελίδα των Υπαλλήλων	97
10.2.1.2 HTML Κώδικας για την σελίδα των Πελατών	99
10.2.1.3 HTML Κώδικας για την σελίδα των Λογαριασμών	100
10.2.1.4 HTML Κώδικας για την σελίδα στις Βλάβες.....	100
10.2.1.5 HTML Κώδικας για την σελίδα στις Κεραίες.....	101
10.2.1.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης.....	101
10.2.1.7 HTML Κώδικας για την σελίδα των Αριθμών Τηλεφώνων	102
10.2.2 HTML Κώδικας του υποκαταστήματος του Ηρακλείου	102
10.2.2.1 HTML Κώδικας για την σελίδα των Υπαλλήλων	103
10.2.2.2 HTML Κώδικας για την σελίδα των Πελατών	104
10.2.2.3 HTML Κώδικας για την σελίδα των Λογαριασμών	105
10.2.2.4 HTML Κώδικας για την σελίδα στις Βλάβες.....	105
10.2.2.5 HTML Κώδικας για την σελίδα στις Κεραίες.....	106
10.2.2.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης.....	106
10.2.2.7 HTML Κώδικας για την σελίδα των Αριθμών Τηλεφώνων	107
10.2.3 HTML Κώδικας του υποκαταστήματος του Λασιθίου.....	107
10.2.3.1 HTML Κώδικας για την σελίδα των Υπαλλήλων	108
10.2.3.2 HTML Κώδικας για την σελίδα των Πελατών	109
10.2.3.3 HTML Κώδικας για την σελίδα των Λογαριασμών	110
10.2.3.4 HTML Κώδικας για την σελίδα στις Βλάβες.....	110
10.2.3.5 HTML Κώδικας για την σελίδα στις Κεραίες.....	111
10.2.3.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης.....	111
10.2.3.7 HTML Κώδικας για την σελίδα των Αριθμών Τηλεφώνων	112
10.2.4 HTML Κώδικας του υποκαταστήματος του Ρεθύμνου	112
10.2.4.1 HTML Κώδικας για την σελίδα των Υπαλλήλων	113
10.2.4.2 HTML Κώδικας για την σελίδα των Πελατών	115
10.2.4.3 HTML Κώδικας για την σελίδα των Λογαριασμών	115



10.2.4.4 HTML Κώδικας για την σελίδα στις Βλάβες.....	116
10.2.4.5 HTML Κώδικας για την σελίδα στις Κεραίες.....	116
10.2.4.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης.....	116
10.2.4.7 HTML Κώδικας για την σελίδα των Αριθμών Τηλεφώνων	117
10.3 HTML Κώδικας της Επικοινωνίας του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s.....	118
ΚΕΦΑΛΑΙΟ 11: Βιβλιογραφία	122
ΠΑΡΟΥΣΙΑΣΗ.....	123

Πίνακας Εικόνων

Εικόνα 1: Η επικοινωνία browsers και servers σύμφωνα με το μοντέλο του Client – Server.....	4
Εικόνα 2: Η λειτουργία των servlet	20
Εικόνα 3: Σχηματική απεικόνιση τις αρχιτεκτονικής των τριών επιπέδων.....	38
Εικόνα 4: Σχηματική απεικόνιση της ιεραρχικής Βάσης δεδομένων.....	39
Εικόνα 5: Σχηματική απεικόνιση των δικτυωτών βάσεων δεδομένων	39
Εικόνα 6: Σχηματική απεικόνιση της σχεσιακής βάσης δεδομένων.....	40
Εικόνα 7: Παράδειγμα σύνδεσης οντοτήτων.....	45
Εικόνα 8: Παράδειγμα σύνδεσης οντοτήτων.....	45
Εικόνα 9: Παράδειγμα σύνδεσης οντοτήτων.....	46
Εικόνα 10: Το μοντέλο οντοτήτων συσχετίσεων που αφορά την εργασία μου	47
Εικόνα 11: Το localhost τουxampp όπου δημιουργήσα και περιέχει τις βάσεις μου.	51
Εικόνα 12: Τα parent directory που αποθηκεύω τα αρχεία μου για να τρέξουν για το κάθε υποκατάστημα.....	52
Εικόνα 13: Η βάση των Χανίων με τους πίνακες που περιέχει στο εσωτερικό της.....	55
Εικόνα 14: Η βάση του Ηρακλείου με τους πίνακες που περιέχει στο εσωτερικό της	57
Εικόνα 15: Η βάση του Ρεθύμνου με τους πίνακες που περιέχει στο εσωτερικό της.	59
Εικόνα 16: Η βάση του Λασιθίου με τους πίνακες που περιέχει στο εσωτερικό της.....	61
Εικόνα 17: HTML σελίδα – η αρχική σελίδα του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s.....	79
Εικόνα 18: HTML σελίδα – η αρχική σελίδα του κάθε υποκαταστήματος της κινητής τηλεφωνίας.....	80
Εικόνα 19: HTML σελίδα για τους Υπαλλήλους	84
Εικόνα 20: HTML σελίδα για τους Πελάτες.....	85
Εικόνα 21: HTML σελίδα για τους Λογαριασμούς	86
Εικόνα 22: HTML σελίδα για τις Βλάβες	87
Εικόνα 23: HTML σελίδα για τις Κεραίες	88
Εικόνα 24: HTML σελίδα για τα Πακέτα σύνδεσης.....	89
Εικόνα 25: HTML σελίδα για των Αριθμό των Τηλεφώνων	90
Εικόνα 26: HTML σελίδα - η Επικοινωνία του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s.....	91



ΚΕΦΑΛΑΙΟ 1 : Εισαγωγικά

Εισαγωγή

Η πτυχιακή εργασία είναι ένα σημαντικό κομμάτι των σπουδών, διότι δίνει την δυνατότητα στον σπουδαστή να μελετήσει, να αναλύσει και να αναπτύξει διάφορα θέματα υπό την καθοδήγηση του εισηγητή της πτυχιακής. Από την όλη διαδικασία ο σπουδαστής γίνεται κάτοχος νέων γνώσεων και μέσα από την έρευνα μαθαίνει και γνωρίζει το αντικείμενο εις βάθος παίρνοντας εφόδια που θα του είναι χρήσιμα στην περαιτέρω επιστημονική και επαγγελματική του πορεία.

1.1 Περίληψη

Η παρούσα πτυχιακή εργασία ασχολείται με την ανάλυση, σχεδίαση και υλοποίηση μιας διαδικτυακής εφαρμογής μιας κινητής τηλεφωνίας. Η συγκεκριμένη εφαρμογή αντλεί και αποθηκεύει πληροφορίες σε μια βάση δεδομένων. Η εκτέλεση της εφαρμογής γίνεται σ' ένα web server για να είναι διαθέσιμη στο διαδίκτυο. Για να υλοποιηθούν όλα τα παραπάνω χρησιμοποιήθηκε το σύστημα διαχείρισης βάσεων δεδομένων Mysql και ο Apache web server. Για την ανάπτυξη ιστοσελίδων χρησιμοποιήθηκε η PHP και JavaScript. Όλα τα παραπάνω είναι λογισμικά ανοιχτού κώδικα και συνεργάζονται άριστα μεταξύ τους.

Η εφαρμογή θα δίνει την δυνατότητα σε υπάλληλους να βλέπουν σε κάθε υποκατάστημα της κινητής τηλεφωνίας πληροφορίες για υπαλλήλους, πελάτες, βλάβες, λογαριασμούς, πακέτα σύνδεσης, έχοντας σαν αποτέλεσμα την καλύτερη και αποδοτικότερη διαχείριση της επιχείρησης.

1.2 Κίνητρο για την Διεξαγωγή της Εργασίας

Η ραγδαία ανάπτυξη του διαδικτύου και των εφαρμογών που αναπτύσσονται σε αυτό, αποτέλεσε το βασικότερο λόγο που επέλεξα το συγκεκριμένο θέμα, όπως και η απόκτηση εμπειρίας στον τομέα αυτό. Επίσης, γνωρίζοντας ότι στην αγορά εργασίας υπάρχει ενδιαφέρον και ζήτηση για εξειδίκευση στην διαχείριση βάσεων δεδομένων, καθώς και στην ικανότητα ανάπτυξης ιστοσελίδων. Με αυτό τον τρόπο μπορεί η πτυχιακή μου να με βοηθήσει στην μετέπειτα επαγγελματική μου αποκατάσταση.

Η κινητή τηλεφωνία έχει μπει για τα καλά στην ζωή μας οπότε η ενασχόληση μου για μία ενδοεταιρική εφαρμογή η οποία θα βοηθάει στην καλύτερη διαχείριση μιας επιχείρησης είτε σε οικονομικό, είτε σε θέμα απόδοσης των εργαζομένων.

1.3 Σκοπός και Στόχοι Εργασίας

Σκοπός της πτυχιακής εργασίας μας είναι η εξοικείωση με τις βάσεις δεδομένων και ανάπτυξη ιστοσελίδων. Επίσης η επέκταση και εμβάθυνση προηγούμενων γνώσεων σε γλώσσες σχεδίασης προγραμματισμού (π.χ html, php, JavaScript, jsp κλπ), γλώσσες ανάπτυξης βάσεων δεδομένων (π.χ MySQL).

Σημαντικοί στόχοι για την εκπόνηση της εργασίας:

- α) Σχεδίαση του σχεσιακού μοντέλου οντοτήτων συσχετίσεων (E-R Model)
- β) Υλοποίηση μοντέλου στην MySQL
- γ) Σύνδεση βάσης με την ιστοσελίδα μας
- δ) Υλοποίηση του πρακτικού μέρους της πτυχιακής εργασίας
- ε) Έλεγχος λειτουργίας και διόρθωση σφαλμάτων
- ζ) Συγγραφή αναφοράς εργασίας
- η) Υποβολή αίτησης αξιολόγησης εργασίας
- θ) Προετοιμασία παρουσίασης αναφοράς
- ι) Παρουσίαση αναφοράς



1.4 Δομή Εργασίας

Η δομή της πτυχιακής εργασίας μου περιλαμβάνει κατά σειρά τα παρακάτω κεφάλαια:

- 1) Παρουσίαση των θεωρητικών θεμάτων και των όρων που σχετίζονται με την εργασία.
- 2) Περιγράφονται οι έννοιες του διαδικτύου, του παγκόσμιου ιστού και οι κατηγορίες των ιστοσελίδων που υπάρχουν.
- 3) Ανάλυση των τεχνολογιών διαδικτύου (π.χ html, php, jsp κτλ) και τεχνολογίες βάσεων δεδομένων (π.χ Mysql).
- 4) Παρουσίαση των λογισμικών που επιλέχτηκαν για την εκποίηση της εφαρμογής μου.
- 5) Ανάλυση απαιτήσεων, κατασκευή σεναρίων, σχεδιασμός της βάσης δεδομένων.
- 6) Κώδικες της δυναμικής μου σελίδας σε Mysql με τα ανάλογα print Screens.
- 7) Κώδικες σε PHP με τα ανάλογα print Screens.
- 8) Τα print Screens τις ιστοσελίδας μου.
- 9) Ο επίλογος με τα αποτελέσματα, συμπεράσματα και την βιβλιογραφία μου.
- 10) Οι κώδικες σε Javascript που χρειάστηκαν για την δημιουργία της ιστοσελίδας μου.
- 11) Η παρουσίαση της πτυχιακής μου στην επιτροπή αξιολόγησης.



ΚΕΦΑΛΑΙΟ 2 : Μεθοδολογία Υλοποίησης

2.1 Μέθοδος Ανάλυσης και Ανάπτυξης Πτυχιακής

Η ανάπτυξη ενός ενδοεταιρικού συστήματος για μια τηλεφωνική εταιρία είναι μια web εφαρμογή που αντλεί και αποθηκεύει πληροφορίες σε μια βάση δεδομένων και εκτελείται σε ένα web server για να είναι διαθέσιμη στο διαδίκτυο. Απαιτείται λοιπόν ανάπτυξη βάσης δεδομένων και παράλληλα ανάπτυξη λογισμικού για την άντληση πληροφοριών από την βάση καθώς και την αποθήκευση πληροφοριών στη βάση.

2.2 Θεωρίες

2.2.1 Διαδίκτυο (Internet).

Διαδικτύωση είναι η μέθοδος κατά την οποία παρέχεται οικουμενική εξυπηρέτηση μεταξύ ετερογενών δικτύων. Η Οικουμενική εξυπηρέτηση επιτρέπει σε κάθε υπολογιστή ενός οργανισμού να επικοινωνεί με οποιοδήποτε άλλο υπολογιστή σε οποιοδήποτε μέρος βρίσκεται. Είναι κάτι ανάλογο του τηλεφωνικού συστήματος. Ετερογενή δίκτυα είναι τα δίκτυα που χρησιμοποιούν διαφορετικές τεχνολογίες. Για την πραγματοποίηση της διαδικτύωσης απαιτείται πρόσθετο υλικό και λογισμικό. Το σύστημα των συνδεδεμένων φυσικών δικτύων που προκύπτει λέγεται **διαδίκτυο (internet)**.

Το βασικό στοιχείο που χρησιμοποιείται για τη σύνδεση 2 ή περισσότερων Ετερογενών δικτύων είναι ο **Δρομολογητής (router)**. Οι συμβατικοί υπολογιστές που είναι συνδεδεμένοι στο διαδίκτυο λέγονται **υπολογιστές υπηρεσίας (hosts)**.

Η υπηρεσία ενός δρομολογητή είναι να μετακινεί δεδομένα μεταξύ των δικτύων στα οποία είναι συνδεδεμένος και γενικότερα οι δρομολογητές μετακινούν δεδομένα από μια αφετηρία σε ένα προορισμό.

Το **TCP/ IP** είναι το σύνολο των πρωτοκόλλων που χρησιμοποιείται στο διαδίκτυο.

2.2.2 Παγκόσμιος Ιστός (World Wide Web - WWW).

Ο Παγκόσμιος ιστός είναι ένα Παγκόσμιο σύστημα υπερμέσων, με αλληλεπιδραστική πρόσβαση, δυναμικό, κατανεμημένο, ανεξάρτητο από υπολογιστή το οποίο λειτουργεί στο διαδίκτυο.

Ο όρος **υπερμέσα (hypermedia)** αναφέρεται στο περιεχόμενο του ιστού το οποίο εκτός από κείμενο μπορεί να περιέχει ψηφιοποιημένες φωτογραφίες και γραφικά. Ένα έγγραφο υπερμέσων που είναι διαθέσιμο στον ιστό λέγεται **ιστοσελίδα**.

Ο όρος κατανεμημένο υπονοεί ότι οι ιστοσελίδες δεν βρίσκονται σε ένα υπολογιστικό σύστημα αλλά σε πολλά. Οι υπολογιστές αυτοί οι οποίοι φιλοξενούν τις ιστοσελίδες και τις διαθέτουν σε ολόκληρο τον κόσμο λέγονται **διακομιστές ιστού (web servers)**.

Κάθε ιστοσελίδα ορίζεται μονοσήμαντα από την διεύθυνση της ή αλλιώς από το **URL (Uniform Resource Locator)**. Η προσπέλαση των ιστοσελίδων γίνεται με την βοήθεια αλληλεπιδραστικών προγραμμάτων τους **φυλομετρητές (browsers)**. Παραδείγματα τέτοιων προγραμμάτων είναι ο Internet Explorer, ο Mozilla Firefox κλπ.

Οι browsers και οι servers επικοινωνούν σύμφωνα με το μοντέλο του **Client – Server**. Ο browser παίζει το ρόλο του Client και υποβάλει αιτήσεις στον server ζητώντας το περιεχόμενο μιας ιστοσελίδας υποβάλλοντας το URL της. Ο server στέλνει την ιστοσελίδα η οποία μεταφράζεται από τον browser και είναι διαθέσιμη. Οι κανόνες επικοινωνίας μεταξύ browser – server γίνεται με το **Hypertext Transfer Protocol (HTTP)**. Σύμφωνα με αυτό το πρωτόκολλο ο Client (browser) στέλνει ένα αίτημα (request) και ο server αποκρίνεται (response).

Μια συναλλαγή βασισμένη στο πρωτόκολλο HTTP από τελείται από τα ακόλουθα στάδια :

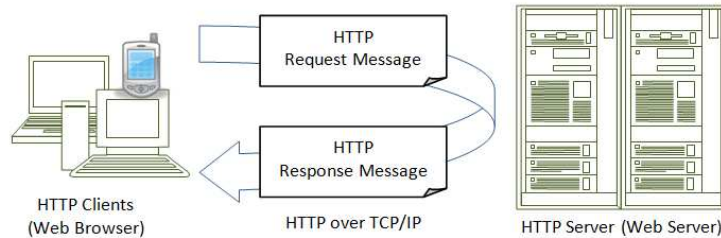
* **Σύνδεση** : Ο Client στέλνει **CONNECT** στον server

* **Αίτηση** : Ο Client στέλνει **HTTP Request** στον server



* **Απάντηση** : Ο server στέλνει **HTTP Response** στον Client

* **Κλείσιμο** : Μετά την απόκριση του server, τερματίζεται η συνδιαλλαγή τους και επομένως δεν είναι διαθέσιμες στη συνέχεια οι πληροφορίες για τη σύνδεση που πραγματοποιήθηκε. Είναι **Stateless protocol**.



Εικόνα 1: Η επικοινωνία browsers και servers σύμφωνα με το μοντέλο του Client – Server.

2.2.3 Κατηγορίες Ιστοσελίδων

Οι ιστοσελίδες ανάλογα με το πότε αλλάζει το περιεχόμενό τους κατατάσσονται σε 3 κατηγορίες :

A) **Στατικές ιστοσελίδες** των οποίων το περιεχόμενο παραμένει αμετάβλητο. Κάθε αίτηση έχει ακριβώς την ίδια απόκριση.

B) **Δυναμικές ιστοσελίδες** των οποίων το περιεχόμενο δημιουργείται από το web server σε κάθε αίτηση του browser. Δηλαδή τα περιεχόμενα της δυναμικής ιστοσελίδας διαφέρουν από αίτηση σε αίτηση. Ο web server εκτελεί ένα πρόγραμμα του οποίου η έξοδος επιστρέφεται στον browser.

Γ) **Ενεργές ιστοσελίδες** των οποίων το περιεχόμενο αλλάζει αφού φορτωθεί η ιστοσελίδα στον browser. Σε αυτή την περίπτωση η ιστοσελίδα περιλαμβάνει ένα πρόγραμμα το οποίο εκτελείται τοπικά στον browser και αλλάζει συνεχώς τα στοιχεία που εμφανίζονται στην οθόνη. Για παράδειγμα η εμφάνιση τιμών μετόχων απαιτεί ενεργή ιστοσελίδα διότι μπορεί να αλλάξει τις διαμορφούμενες τιμές χωρίς να απαιτείται οποιαδήποτε ενέργεια από τον χρήστη.



ΚΕΦΑΛΑΙΟ 3 : Background

3.1 Τεχνολογίες Διαδικτύου

3.1.1 HTML (Hypertext Markup Language)

Τα αρχικά HTML προέρχονται από τις λέξεις Hypertext Markup Language. Η html δεν είναι μια γλώσσα προγραμματισμού. Είναι μια γλώσσα σήμανσης (markup language), δηλαδή ένας ειδικός τρόπος γραφής κειμένου. Ο καθένας μπορεί να δημιουργήσει ένα αρχείο HTML χρησιμοποιώντας απλώς έναν επεξεργαστή κειμένου. Αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Ο browser αναγνωρίζει αυτόν τον τρόπο γραφής και εκτελεί τις εντολές που περιέχονται σε αυτόν.

Αξίζει να σημειωθεί ότι η html είναι η πρώτη και πιο διαδομένη γλώσσα περιγραφής της δομής μιας ιστοσελίδας. Η html χρησιμοποιεί τις ειδικές ετικέτες (τα tags) να δώσει τις απαραίτητες οδηγίες στον browser. Τα tags είναι εντολές που συνήθως ορίζουν την αρχή ή το τέλος μιας λειτουργίας. Τα tags βρίσκονται πάντα μεταξύ των συμβόλων <και>. Π.χ <BODY> Οι οδηγίες είναι case insensitive, δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά (μικρά) ή κεφαλαία. Ένα αρχείο HTML πρέπει να έχει κατάληξη html ή htm.

Κανόνες

Για να μπορούν οι browser να ερμηνεύσουν σχεδόν απόλυτα σωστά την html έχουν θεσπιστεί κάποιοι κανόνες. Αυτοί οι κανόνες είναι γνωστοί ως προδιαγραφές. Επομένως σχεδόν κάθε είδος υπολογιστή μπορεί να δείξει το ίδιο καλά μια ιστοσελίδα. Οι πρώτες προδιαγραφές ήταν η html 2.0. πρόβλημα προέκυψε όταν η Microsoft και η Netscape πρόσθεσαν στην html τέτοιες δυνατότητες που στην αρχή τουλάχιστον ήταν συμβατές μόνο με συγκεκριμένους browser. Ακόμη και σήμερα υπάρχουν διαφορές στην απεικόνιση κάποιας σελίδας από διαφορετικούς browsers. Ιδιαίτερο είναι το πρόβλημα όταν η ιστοσελίδα, εκτός από «καθαρή» HTML περιλαμβάνει και εφαρμογές JavaScript.

Η HTML σήμερα

Σήμερα πολλοί είναι εκείνοι που δημιουργούν μια ιστοσελίδα σε κάποιο πρόγραμμα που επιτρέπει την δημιουργία χωρίς την συγγραφή κώδικα. Η κοινή άποψη πάνω στο θέμα όμως είναι ότι κάτι τέτοιο είναι αρνητικό επειδή ο δημιουργός δεν έχει τον απόλυτο έλεγχο του κώδικα με αποτέλεσμα πολλές φορές να υπάρχει οπτικό χάος στην προσπάθεια των browser να εμφανίσουν την ιστοσελίδα. Για το σκοπό αυτό έχει δημιουργηθεί ειδικό λογισμικό, που επιτρέπει το «στήσιμο» της σελίδας οπτικά, χωρίς τη συγγραφή κώδικα, δίνει όμως τη δυνατότητα παρέμβασης ΚΑΙ στον κώδικα. Χαρακτηριστικό παράδειγμα το λογισμικό Dream weaver της Adobe και το FrontPage της Microsoft.

Εμφάνιση των αρχείων HTML

Οι σελίδες που γράφονται σε HTML είναι απλά αρχεία κειμένου και δε περιέχουν πληροφορίες συγκεκριμένες για ένα λειτουργικό σύστημα ή ένα πρόγραμμα, μπορούν να διαβάζονται από οποιοδήποτε συντάκτη υποστηρίζει απλό κείμενο. Τα tags της HTML είναι τα πράγματα που φαίνονται μέσα στα σύμβολα <>. Τα tags υποδεικνύουν χαρακτηριστικά ή στοιχεία μιας σελίδας. Τα <όνομα tag> επηρεαζόμενο κείμενο </όνομα tag >.

Το ίδιο όνομα του tag περικλείεται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (< >). Τα tags της HTML έχουν γενικά ένα tag αρχής και ένα tag τέλους. Τα δύο αυτά συντροφικά tags περικλείουν το κείμενο που επηρεάζουν. Το tag αρχής ενεργοποιεί μια λειτουργία ή ένα χαρακτηριστικό (όπως ο χαρακτηρισμός μιας επικεφαλίδας, η έντονη γραφή), ενώ το tag τέλους την απενεργοποιεί. Τα tags τέλους έχουν το ίδιο όνομα με τα tags αρχής, με το πρόθεμα τον χαρακτήρα '/'. Δεν έχουν μορφή ζεύγους όλα τα tags της HTML. Ορισμένα tags είναι "μονομελή", ενώ άλλα



περιέχουν επιπλέον πληροφορίες και κείμενο μέσα στα σύμβολα "<>". Όλα τα tags της HTML δεν κάνουν διάκριση μεταξύ κεφαλαίων και πεζών.

Σύνταξη και εντολές

Είναι μια περιγραφική γλώσσα μορφοποίησης δεδομένων και κλήσης άλλων αρχείων. Τα βασικά δομικά στοιχεία της html είναι οι ετικέτες (tags) που αναγνωρίζονται και μεταφράζονται σε εντολές από τους browsers και έτσι παρουσιάζεται η ιστοσελίδα.

Οι ετικέτες περικλείονται από τα σύμβολα < > και ομαδοποιούνται σε ζεύγη για άνοιγμα και κλείσιμο π.χ <h3> </h3>.

Κάθε σελίδα HTML περικλείεται από τα tags <html> </html> και χωρίζεται σε δύο τμήματα το <head> </head> και στο <body> </body> .

Στο τμήμα <head> περιέχονται : μεταπληροφορίες (metatags) για την ιστοσελίδα οι οποίες μπορούν να αξιοποιηθούν από τις μηχανές αναζήτησης βελτιώνοντας σημαντικά την κατάταξη της.

Ο τίτλος της ιστοσελίδας ανάμεσα στα tags <title> </title> ο οποίος εμφανίζεται στο πάνω μέρος του browser.

Ορίζεται η χρήση του CSS και JavaScript.

Στην περιοχή body ορίζεται το κύριο μέρος της ιστοσελίδας και μπορεί να περιέχει Ετικέτες μορφοποίησης

<p> Ορίζει έναρξη παραγράφου.

<h1><h2> <h6> Το κείμενο ανάμεσα σε αυτές τις ετικέτες έχει την μορφή επικεφαλίδας της οποίας το μέγεθος μειώνεται όσο μεταβαίνουμε από την <h1> στην <h6>.

<hr> Εμφανίζει μια οριζόντια γραμμή στην σελίδα μας.

<i> Εμφανίζει το κείμενο ανάμεσα στο εν λόγω tag με πλάγιους χαρακτήρες.

 Εμφανίζει το κείμενο με έντονη μορφή.

<u> Εμφανίζει το κείμενο υπογραμμισμένο.

 Αλλαγή γραμμής.

Λίστες Ταξινομημένες ή αταξινόμητες με τα tags και αντίστοιχα π.χ

 Red Green Blue

Πλαίσια (frames) ορίζονται με το tag <frame>

Εικόνες φορτώνονται με το tag img όπου ορίζουμε το όνομα και την διαδρομή του αρχείου εικόνας (scr), τις διαστάσεις της (with - height) και την επεξήγηση (title) π.χ

Υπερσυνδέσμους σε άλλες σελίδες με το tag <A> όπου ορίζουμε το όνομα και την διαδρομή της σελίδας (HREF), την επεξήγηση (title) π.χ

 HERE

Πίνακες δεδομένων ορίζονται με τα tags <table></table>. Το <table> έχει διάφορες ιδιότητες όπως width border, caption

π.χ <table width= "100%" border= "1"><caption> Πελάτες </caption> >

Οι γραμμές του πίνακα ορίζονται με το tag <tr></tr> ενώ οι στήλες με το tag <th>....</th> για επικεφαλίδες στηλών και <td>....</td> για δεδομένα στηλών. Π.χ <table> <tr> <th> Name </th> <th> address </th> <th> phone </th> </tr>

<tr><th rowspan= "2 J.papas </th><th> Herakleion</th> </tr>

<tr><th>123-444 </th> </tr></table>

Η rowspan ορίζει σε πόσες γραμμές απλώνεται ένα κελί παρόμοιο είναι και το colspan.

Φόρμες εισαγωγής δεδομένων από τον client και αποστολής τους στον server. Ορίζονται με τα tags <form></form>. Η <form> έχει 2 ιδιότητες την **Action** που ορίζει σε ποια σελίδα θα στείλουμε τα δεδομένα και την **Method** που ορίζει την μέθοδο του πρωτοκόλλου HTTP που θα χρησιμοποιήσουμε για την αποστολή δεδομένων **GET** ή **POST** <form action= "page1.php" method= "get">.

Με την μέθοδο POST τα δεδομένα στέλνονται μέσα στα HTTP μηνύματα που στέλνει ο browser και δεν τα βλέπει ο χρήστης, ενώ με την μέθοδο GET τα δεδομένα προστίθενται στο url π.χ

<http://www.teicrete.gr/page1.php?a1=12>

Μια φόρμα μπορεί να περιέχει τα παρακάτω στοιχεία:

- Στοιχείο **Text** για την εισαγωγή κειμένου π.χ <input type= "text" id= " c1" name= " code"



value= " 0" >. Πριν από το στοιχείο text μπορούμε να εμφανίσουμε μια ετικέτα για να γνωρίζει ο χρήστης τι είναι αυτό που εισάγει π.χ <label for="c1"> κωδικός: </label> .

- Στοιχείο **Textarea** για την εισαγωγή κειμένου περισσοτέρων από μια γραμμή π.χ <Textarea name="area1"> περιγραφή </Textarea> .

- Στοιχείο για **checkbox** για να τσεκάρει ο χρήστης μια η περισσότερες επιλογές π.χ <input type="checkbox" name="lang1" value="English"> English

<input type="checkbox" name="lang2" value="German"> German

<input type="checkbox" name="lang3" value="France"> France

- Στοιχείο **radio button** για να τσεκάρει ο χρήστης μια επιλογή π.χ

<input type="radio" name=" thesex" value=" male "> male

<input type="radio" name=" thesex" value=" female "> female

- Στοιχείο **select** ορίζει μια pop up list για να επιλέξει ο χρήστης μια εφαρμογή. Πχ

<select name =>"combo ")

<option > ag nikolaos </option >

<option > heraklion </option >

<option > rethimno </option >

<option > hania </option >

</select>

- Στοιχείο **button** ορίζουμε κουμπιά διαφόρων τύπων όπως submit για αποστολή της φόρμας στον server, reset για επαναφορά της φόρμας. Π.χ

<input type=" submit" name=" button1" value= "Αποστολή" >

<input type=" reset " name=" reset" value= " reset " >

Πλεονεκτήματα :

- Εύκολο στη χρήση.
- Χαλαρή σύνταξη (αν και επειδή είναι πάρα πολύ ευέλικτο δεν θα συμφωνεί με τα πρότυπα)
- Υποστηρίζεται σχεδόν σε κάθε πρόγραμμα περιήγησης, αν όχι όλους.
- Χρησιμοποιείται ευρέως. Ιδρύθηκε σχεδόν σε κάθε ιστοσελίδα, αν όχι όλες.
- Πολύ παρόμοια με την σύνταξη του XML, το οποίο χρησιμοποιείται όλο και περισσότερο για την αποθήκευση δεδομένων.
- Δωρεάν – δεν χρειάζεται να αγοράσετε κανένα λογισμικό.
- Εύκολο στη μάθηση και δημιουργία κώδικα ακόμα και για αρχάριους προγραμματιστές.

Μειονεκτήματα :

- Δεν μπορεί να παράγει δυναμική έξοδο από μόνη της, δεδομένου ότι είναι μια στατική γλώσσα.
- Μερικές φορές η δομή των HTML εγγράφων είναι δύσκολο να κατανοηθούν.
- Θα πρέπει να ενημερώνεται για ξεπερασμένες ετικέτες, και να φροντίζει να μην τις χρησιμοποιεί.
- Εμφανίζονται απαξιωμένες ετικέτες επειδή μια άλλη γλώσσα που λειτουργεί με την HTML έχει αντικαταστήσει το πρωτότυπο έργο της ετικέτας. Έτσι η άλλη γλώσσα πρέπει να μαθευτεί (τις περισσότερες φορές, είναι η CSS).
- Οι λειτουργίες ασφαλείας που προσφέρει η HTML είναι περιορισμένες.

3.1.2 PHP (Hypertext Preprocessor)

Είναι γλώσσα προγραμματισμού για την δημιουργία δυναμικών ιστοσελίδων. Είναι **Server side scripting** δηλαδή το πρόγραμμα εκτελείται στον web-server. Η εκτέλεση του προγράμματος στον web-server γίνεται με την βοήθεια του **μεταγλωττιστή PHP (parser)** που πρέπει να έχει εγκατασταθεί στον web-server. Η PHP είναι HTML ενσωματωμένη scripting γλώσσα. Δηλαδή κώδικας PHP συμπεριλαμβάνεται μέσα σε μια σελίδα HTML. Κατά την μεταγλώττιση η PHP αφήνει την σελίδα HTML όπως είναι μέχρι να συναντήσει ένα από τα ειδικά tags έναρξης php "<?" και τότε



αρχίζει να μεταγλωττίζει το κείμενο ως κώδικα PHP μέχρι να συναντήσει το επόμενο PHP tag κλεισίματος ">" δηλαδή οτιδήποτε βρίσκεται έξω από τα tags της PHP μένει όπως είναι. Το αποτέλεσμα της μεταγλώττισης μορφοποιείται σε HTML και όλη η ιστοσελίδα αποστέλλεται στον browser.

Από τα παραπάνω συμπεραίνεται ότι η PHP είναι **διερμηνέας (interpreter)** δηλαδή δεν παράγει εκτελέσιμο αρχείο. Είναι **command line scripting** δηλαδή ένα Php script μπορεί να τρέξει χωρίς browser και server. Χρειαζόμαστε μόνο τον μεταγλωττιστή. Υποστηρίζει πολλά πρωτόκολλα για επικοινωνία με άλλες υπηρεσίες όπως IMAP, SNMP, NNTP, HTTP, POP3 καθώς επίσης και τα socket. Υποστηρίζεται από τους περισσότερους web servers.

Επεκτάσεις αρχείων και διακομιστές

Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ *.php, *.php4, *.phphtml κ.α). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν το αρχείο έχει την επέκταση .php θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται κώδικα php. Ο διακομιστής Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με τα λειτουργικά συστήματα GNU/ Linux και Microsoft Windows, υποστηρίζει εξ ορισμού την εκτέλεση κώδικα PHP.

Ιστορία της PHP

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας την γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε άλλους φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφτασε στην έκδοση 2.0, βασιζόμενη αυτή τη φορά στην γλώσσα C και αριθμώντας περισσότερους από 50.000 ιστότοπους που την χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο την σημερινή μορφή της.

Στην συνέχεια, οι Andi Gutmans και Zeev Suraski δημιούργησαν την εταιρία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Οι περισσότεροι ιστότοποι επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP.

Σύνταξη και Εντολές της PHP

Ο κώδικας αρχίζει με <? και τελειώνει με ?>.

Κάθε εντολή τελειώνει με Ελληνικό ερωτηματικό (;)

Τα σχόλια αρχίζουν με //

Οι μεταβλητές ορίζονται με την απόδοση τιμής. Το όνομα τους Ξεκινά με \$ και ακολουθεί γράμμα ή κάτω παύλα και στην συνέχεια οτιδήποτε εκτός από χαρακτήρες +, -, *, &. Είναι case-sensitive π.χ \$total και \$Total είναι διαφορετικές μεταβλητές.

Υπάρχουν οι γνωστοί τύποι μεταβλητών συμβολοσειρές (strings), ακέραιοι (integers), αριθμοί κινητής υποδιαστολής (floats), λογικές τιμές (booleans), πίνακες (arrays) και αντικείμενα (objects).

Δομές ελέγχου

if (συνθήκη) {εντολές} else { εντολές }

Switch (μεταβλητή) case τιμή 1: εντολές break; case τιμή 2: εντολές break; Default: εντολές;



Δομές επανάληψης

```
while (συνθήκη) { εντολές }  
Do { εντολές } while (συνθήκη)  
For (Αρχική τιμή; Τελική τιμή; Βήμα)
```

Ορισμός πινάκων

```
$Pin1=new array(); //ορίζει ένα κενό πίνακα  
$colors= array('red', 'blue', 'green', 'yellow');  
$Pin2[]="TEI"; $Pin2[]="Ηράκλειο"; //Με αυτόν τον τρόπο ο τελεστής [] ορίζει τον πίνακα $Pin2 και  
του δίνει και περιεχόμενο στις θέσεις $Pin2[0] και $Pin2 [1] τις τιμές "TEI" και "Ηράκλειο"  
αντίστοιχα.
```

Η εμφάνιση των χρωμάτων που περιέχονται στον πίνακα \$colors που ορίσαμε παραπάνω γίνεται με τον βρόχο που ακολουθεί

```
foreach ($colors as $color) {echo "$color\n";}
```

Πρόσβαση σε MYSQL βάση

Για να συνδεθούμε με την βάση χρησιμοποιούμε την συνάρτηση:

Mysql_connect (hostname, username, password)

Hostname είναι η διεύθυνση του υπολογιστή που τρέχει ο MySQL server.

Η παραπάνω συνάρτηση σε περίπτωση επιτυχίας επιστρέφει έναν αναγνωριστή συνδέσμου τον οποίο πρέπει να αποθηκεύουμε σε μια μεταβλητή για να τον χρησιμοποιήσουμε σε επόμενες εντολές π.χ

```
$conn= mysql_connect (hostname, username, password)
```

Σε περίπτωση αποτυχίας η συνάρτηση mysql_connect επιστρέφει false.

mysql_select_db("όνομα βάσης", αναγνωριστής συνδέσμου"). Χρησιμοποιείται για την επιλογή βάσης δεδομένων π.χ mysql_select_db("test1", \$conn).

mysql_create_db("όνομα βάσης", αναγνωριστής συνδέσμου"). Χρησιμοποιείται για την δημιουργία βάσης δεδομένων π.χ mysql_create_db("dimos", \$conn).

mysql_drop_db("όνομα βάσης", αναγνωριστής συνδέσμου"). Χρησιμοποιείται για την διαγραφή βάσης δεδομένων π.χ mysql_drop_db("dimos", \$conn).

mysql_query ("ερώτημα", αναγνωριστής συνδέσμου"). Χρησιμοποιείται για την εκτέλεση ερωτημάτων.

Το αποτέλεσμα του ερωτήματος το αποθηκεύουμε σε μεταβλητή ειδικά αν έχουμε ερώτημα ανάκλησης δεδομένων από πίνακα

π.χ \$query1="select * from users";

```
$result=mysql_query($query1, $conn);
```

για να επεξεργαστούμε το περιεχόμενο του αποτελέσματος σε γραμμές χρησιμοποιούμε τη συνάρτηση **mysql_fetch_array()** η οποία κάθε φορά επιστρέφει μια γραμμή από το αποτέλεσμα για παράδειγμα έστω ότι θέλω να πάρω τα στοιχεία των χρηστών που είναι καταχωρημένη στον πίνακα users. Τα στοιχεία των χρηστών είναι ο κωδικός και το τηλέφωνο δηλαδή τα πεδία code, fullname και phone αντίστοιχα.

```
$sql="select code fullname, phone from users";
```

```
$result=mysql_query ($sql, $conn);
```

```
While ($row= mysql_fetch_array ($result))
```

```
{  
$code=$row ['code'];  
$fullname=$row ['fullname'];  
$phone=$row ['phone'];  
}
```

Αν κάτι δεν πάει καλά κατά την εκτέλεση των εντολών μπορούμε να χρησιμοποιήσουμε την συνάρτηση die() για να εμφανίσουμε κάποιο μήνυμα δικό μας ή το αντίστοιχο μήνυμα της mysql με την χρήση της συνάρτησης mysql_error() ή και τα δύο όπως στο παράδειγμα που ακολουθεί:



```
$conn=mysql_connect("127.0.0.1", "user1", "niki"); αλλιώς  
die ("μη εφικτή η σύνδεση" . mysql_error());
```

Πλεονεκτήματα:

- Είναι εύκολα προσβάσιμη.
- Είναι διαθέσιμη δωρεάν.
- Είναι διαθέσιμη με βοηθητικά έγγραφα σε πολλές γλώσσες.
- Υπάρχουν πολλά γκρουπ, forums, και ομάδες υποστήριξης.
- Υπάρχει πλούτος πληροφοριών στο διαδίκτυο.
- Είναι γρήγορη η ανάπτυξη κώδικα.
- Ένα βασικό script στην Php μπορεί να δημιουργηθεί χωρίς την απαραίτητη κατανόηση των εντολών του προγραμματισμού, την κατάρτιση, καθώς και άλλες σημαντικές έννοιες του προγραμματισμού.
- Η Php είναι χαλαρά δακτυλογραφημένη, γεγονός που καθιστά τα βασικά scripts πολύ πιο γρήγορα στην ανάπτυξη και με λιγότερη έμφαση στο σχεδιασμό.
- Προγραμματιστές της Java, Perl, BASIC, και άλλες δημοφιλείς γλώσσες μπορούν να βρουν πολλές ομοιότητες για την διευκόλυνση μετάβασης στην Php.
- Η Php είναι ευέλικτη. Χρησιμοποιεί OOP ή όχι. Χρησιμοποιεί την μετατροπή –ες ονομάτων ή όχι.
- Τρέχει σε πολλά διαφορετικά λειτουργικά συστήματα.
- Μπορεί να βελτιστοποιηθεί, ακόμη και αν είναι «καταρτισμένη» για τις επιδόσεις πιο κοντά σε εκείνη των περισσότερων γλωσσών.
- Είναι ανοιχτού κώδικα, άμεσα διαθέσιμο (μπορεί ήδη να το χρησιμοποιείτε σήμερα) και διπλής άδειας. Αν κάνετε μη κερδοσκοπική εργασία ή χωρίς τη χορήγηση αδειών, δεν υπάρχει κόστος.
- Πολύ εύκολο να καταλάβει κανείς την σύνταξη, μερικά πραγματικά εντυπωσιακά χαρακτηριστικά γνωρίσματα (δείκτες και άλλα...)
- Έχει πολύ εύκολα διεπαφή με Apache / MySQL.
- Λειτουργεί στον server (δεν χρειάζεται να μπλέκει με εγκατάσταση στον client – μόνο με θέματα rendering, εάν χρησιμοποιείτε CSS).
- Υπάρχουν άφθονες πηγές καλού πηγαίου κώδικα εκεί έξω για να χρησιμοποιηθεί και / ή να μάθουν από αυτόν, καθώς και όπως πολλές χρήσιμες βιβλιοθήκες για την εργασία με αρχεία PDF, γραφικά, κλπ.
- Πολλά καλά βιβλία και βοήθεια στο διαδίκτυο (php.net κ.α).
- Πλατφόρμα αγνωστικιστής, μπορεί να τρέξει σε Windows, Linux ή Mac servers. Επίσης εξαιρετικά επεκτάσιμη.
- Πολλοί περεχείς φιλοξενίας ιστοσελίδων το έχουν έτοιμο προς χρήση, καμία ειδική διαμόρφωση (εκτός και αν έχετε ειδικές ανάγκες ασφαλείας).
- Πολύ εύκολη η πρόσβαση σε άλλα web-based εργαλεία μέσω της Php (π.χ Google maps, κλπ).

Μειονεκτήματα :

- Η Php τείνει να εκτελεί πιο αργά από την assembly, την C, και άλλες γλώσσες.
- Είναι χαλαρά δακτυλογραφημένη. Για τους προγραμματιστές όλων των επιπέδων, αυτό αφήνει περιθώρια για την απροσδόκητη συμπεριφορά, λόγω λάθους του προγραμματιστή που άλλες γλώσσες δεν θα το επέτρεπαν (φυσικά είναι λίγες εάν όχι καμία γλώσσα που μπορούν να προστατέψουν από σφάλμα των προγραμματιστή!).
- Υπάρχουν πολλοί τρόποι για να κάνει ένα πράγμα, και πολλές περιπτώσεις όπου μια λειτουργία έχει διφορούμενο χειρισμό λόγω υποστήριξης της κληρονομιάς ή της ιστορίας ανάπτυξης.



- Αν θέλετε να κάνετε κάτι παραπάνω από απλές HTML / CSS σελίδες για τον πελάτη σας θα πρέπει να προσθέσετε επίσης JavaScript, Java ή άλλη client-side γλώσσα (ισχύει και για Perl και σε μερικές άλλες γλώσσες).
- Ο τρόπος με τον οποίο λειτουργούν τα προγράμματα περιήγησης κάνουν την διαχείριση των δεδομένων και την κωδικοποίηση των προγραμμάτων περισσότερο ενδιαφέροντα (τεχνικά κάθε σελίδα είναι μια νέα λειτουργία για το σύστημα, έτσι ώστε να πρέπει να διαχειριστείτε τις μεταβλητές σας που μπαίνουν και βγαίνουν έξω και ανάμεσα στις σελίδες. Δεν είναι δύσκολο, απλά διαφορετικό).
- Κάθε προγραμματισμός Web (ανεξάρτητα από την γλώσσα) είναι ανοιχτός σε κενά ασφαλείας λόγω των μη εφαρμοσμένων ή άγνωστων ευπαθειών (απαιτείτε λοιπόν λίγο περισσότερο προσοχή).

3.1.3 Perl

Η **Perl** είναι μία πολύ δημοφιλής αντικειμενοστραφής γλώσσα προγραμματισμού. Συνήθως ένα πρόγραμμα σε Perl εκτελείται χρησιμοποιώντας άμεσα ή έμμεσα το διερμηνέα της γλώσσας. Αυτό που διακρίνει την Perl από πολλές άλλες γλώσσες προγραμματισμού είναι το γεγονός ότι είναι διαθέσιμη για σχεδόν όλα τα λειτουργικά συστήματα. Η γλώσσα σχεδιάστηκε από τον Larry Wall και ο πηγαίος κώδικάς της διατίθεται βάση της αδειάς ανοικτού κώδικα GPL. Η πρώτη έκδοση της γλώσσας εμφανίστηκε το 1987 ενώ στις αρχές του 2010 η πιο πρόσφατη έκδοσή της είναι η 5.10.1.

Ιστορία

Η γλώσσα προγραμματισμού Perl σχεδιάστηκε επειδή ο Larry Wall ήθελε μία γλώσσα η οποία να συνδυάζει μόνο τα καλά στοιχεία του sed, της C, της awk και του κελύφους Bourne του λειτουργικού συστήματος Unix. Η πρώτη έκδοση της γλώσσας ανακοινώθηκε στο alt.comp.sources του Usenet στις 18 Δεκεμβρίου του 1987. Μετά την ταχύτερη διάδοση της γλώσσας ακολούθησαν οι εκδόσεις 2 (1988) και 3 (1989). Η τέταρτη έκδοση του 1991 δεν είχε σημαντικές διαφορές από την 3η αλλά δημιουργήθηκε ως έκδοση αναφοράς για το πρώτο βιβλίο με θέμα τη γλώσσα. Η πέμπτη έκδοση ξεκίνησε να γράφεται το 1993 με την σταθερή έκδοση της να εκδίδεται τον Οκτώβριο του 1994. Η έκδοση αυτή έφερε σημαντικές αλλαγές και προσέθεσε σημαντικές δυνατότητες ενώ ταυτόχρονα αύξησε τον αριθμό των υποστηριζόμενων λειτουργικών συστημάτων. Η έκδοση 5 συνεχίζει να αναπτύσσεται ακόμα και βρίσκεται στην έκδοση 5.14.1. Η *Perl 6* είναι η επόμενη έκδοση της γλώσσας η οποία θα επιφέρει αρκετές αλλαγές τόσο στο συντακτικό όσο και στις δυνατότητες της γλώσσας.

Χαρακτηριστικά

Ως γλώσσα προγραμματισμού η Perl έχει ένα δυναμικό σύστημα τύπων, δηλαδή μία μεταβλητή αποκτά τύπο μόνο μετά από την ανάθεση μιας τιμής σε αυτή. Η Perl μπορεί να διαχειριστεί αριθμούς (δεκαδικούς και ακεραίους) και συμβολοσειρές. Επιπλέον, μπορούμε να έχουμε πίνακες απλούς (arrays) και συσχετιστικούς (associative arrays ή απλά hash tables), οι οποίοι είναι μηχανισμοί οργάνωσης δεδομένων. Μάλιστα με τους συσχετιστικούς πίνακες μπορούμε να δημιουργήσουμε δυναμικές δομές όπως δένδρα, στοιβές, ουρές κ.λπ. Υπάρχει πρόνοια για τον διαχωρισμό μεταβλητών στις οποίες αναθέτουμε απλές τιμές ή σύνθετες τιμές.

Όταν ορίζουμε μία διαδικασία ή μία συνάρτηση στην Perl δεν σημειώνουμε τυπικές παραμέτρους αλλά απλά θεωρούμε πως η διαδικασία ή η συνάρτηση θα έχουν ως όρισμα έναν πίνακα. Άρα για μια συνάρτηση ή μία διαδικασία δεν υπάρχει προκαθορισμένος αριθμός ορισμάτων. Επίσης οι συναρτήσεις και οι διαδικασίες στην Perl μπορούν να οριστούν αναδρομικά.

Ένα άλλο βασικό χαρακτηριστικό της γλώσσας είναι η εξαιρετική υποστήριξη των κανονικών εκφράσεων, δηλαδή παραστάσεων που μπορούν να περιγράψουν την γενική δομή μιας συμβολοσειράς και μπορούν να χρησιμοποιηθούν στη διαχείριση συμβολοσειρών.

Η πιο χαρακτηριστική χρήση των κανονικών εκφράσεων είναι στην λεκτική ανάλυση γλωσσών



προγραμματισμού, δηλαδή στο κομμάτι εκείνο ενός μεταγλωττιστή που «σπάει» την είσοδο σε λεκτικές μονάδες (tokens).

Όπως αναφέρθηκε η Perl είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού. Στην Perl οι κλάσεις είναι πακέτα με ειδικές συναρτήσεις τα οποία επιστρέφουν την τιμή 1. Ο τρόπος δημιουργίας και χρήσης κλάσεων δεν είναι ιδιαίτερα κομψός ενώ παράλληλα δεν υποστηρίζει πολλές από τις δυνατότητες που υποστηρίζουν κατεξοχήν αντικειμενοστραφείς γλώσσες όπως η Java και η Scala.

Επίσης η Perl παρέχει εργαλεία για ταυτόχρονη επεξεργασία δεδομένων μέσω νημάτων ή κάνοντας χρήση της κλήσης συστήματος fork. Επιπλέον παρέχει υποδομές για δικτυακό προγραμματισμό. Επίσης παρέχει υποδομές για την εύκολη διασύνδεση με γνωστές βάσεις δεδομένων, ενώ τα τελευταία χρόνια έγινε ιδιαίτερη δημοφιλής λόγω της χρήσης της στη βιοπληροφορική. Τέλος, η Perl έγινε πολύ γνωστή λόγω της χρήσης της για τη δημιουργία των λεγομένων CGI scripts.

Χρήση

Η Perl χρησιμοποιείται από ένα ευρύ φάσμα χρηστών και σε πολλές εφαρμογές. Για παράδειγμα χρησιμοποιείται στη βιοπληροφορική, στη διαχείριση συστημάτων, στη διαχείριση ιστοτόπων κ.λπ.

Λειτουργικά συστήματα

Η Perl είναι προεγκατεστημένη στη συντριπτική πλειοψηφία των λειτουργικών συστημάτων τύπου Unix, ενώ υπάρχουν έτοιμες διανομές για τα περισσότερα λειτουργικά συστήματα. Πιο συγκεκριμένα η Perl μπορεί να χρησιμοποιηθεί στα παρακάτω λειτουργικά συστήματα:

- Linux
- OpenSolaris και Solaris
- Microsoft Windows (όλες οι εκδόσεις συμπεριλαμβανομένης της πλατφόρμας Pocket PC)
- Mac OS και Mac OS X
- FreeBSD, OpenBSD και NetBSD
- QNX

Πλεονεκτήματα :

- Παρέχει τα χαρακτηριστικά γνωρίσματα που απαιτούνται για τα μεγάλα projects, όπως Modularization, αντικειμενοστραφή τεχνικές και αυθαίρετα δεδομένα.
- Είναι ελεύθερο λογισμικό.
- Τρέχει σε όλες τις πλατφόρμες και μπορεί να μεταφερθεί μεταξύ UNIX και Windows χωρίς προβλήματα.

Μειονεκτήματα :

- Συμπεριλαμβάνει πάρα πολλές ειδικές κατασκευές.
- Είναι δύσκολο να δημιουργηθούν data structures.
- Έχει έλλειψη λειτουργικών υπογραφών το οποίο καθιστά δύσκολη τη λίστα των ονομάτων και τους τύπους των παραμέτρων.

Υποδείξεις – Παρατηρήσεις

Ένα από τα χαρακτηριστικά της Perl είναι ότι υπάρχουν περισσότεροι από ένας τρόπος για να κάνετε σχεδόν τα πάντα, γεγονός που καθιστά τη γλώσσα κάπως δύσκολο να την μάθουν αν κάποιος έχουν εκπαιδευτεί σε κάποιο συγκεκριμένο σύστημα. Μπορείτε να βρείτε βιβλία για τη Perl στο διαδίκτυο, καθώς και σε άλλες ηλεκτρονικές πηγές, αν έχετε οποιεσδήποτε ερωτήσεις. Η Perl πολλές φορές



μπορεί να υποφέρει από θέματα μειωμένης απόδοσης επειδή το πρόγραμμα πρέπει να συντάξει κάποιες πηγές για να μπορέσει να τρέξει μια εργασία.

3.1.4 Python

Η **Python** είναι μια γλώσσα προγραμματισμού η οποία δημιουργήθηκε από τον Ολλανδό Γκουίντο βαν Ρόσσουμ (Guido van Rossum) το 1990. Ο κύριος στόχος της είναι η αναγνωσιμότητα του κώδικά της και η ευκολία χρήσης της. Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησής της.

Η Python αναπτύσσεται ως ανοιχτό λογισμικό (open source) και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation. Ο κώδικας διανέμεται με την άδεια Python Software Foundation License η οποία είναι συμβατή με την GPL. Το όνομα της γλώσσας προέρχεται από την ομάδα Άγγλων κωμικών Μόντυ Πάιθον.

Ιστορικό

Αρχικά, η Python ήταν γλώσσα σεναρίων που χρησιμοποιούνταν στο λειτουργικό σύστημα Amoeba, ικανή και για κλήσεις συστήματος.

Η Python 2.0 κυκλοφόρησε στις 16 Οκτωβρίου του 2000. Στις 3 Δεκεμβρίου 2008 κυκλοφόρησε η έκδοση 3.0 (γνωστή και ως py3k ή python 3000). Πολλά από τα καινούργια χαρακτηριστικά αυτής της έκδοσης έχουν μεταφερθεί στις εκδόσεις 2.6 και 2.7 που είναι προς τα πίσω συμβατές.

Η python 3 είναι ιστορικά η πρώτη γλώσσα προγραμματισμού που σπάει την προς τα πίσω συμβατότητα με προηγούμενες εκδόσεις ώστε να διορθωθούν κάποια λάθη που υπήρχαν σε προγενέστερες εκδόσεις και να καταστεί ακόμα πιο σαφής ο απλός τρόπος με τον οποίο μπορούν να γίνουν κάποια πράγματα.

Δομή και σύνταξη

Η γλώσσα χρησιμοποιεί μεταγλωττιστή (compiler) για την δημιουργία του εκτελέσιμου κώδικα και σχετίζεται με τις γλώσσες προγραμματισμού Tcl, Perl, Scheme, Java και Ruby, καθώς και με την ABC η οποία υπήρξε η αρχική πηγή έμπνευσης για τη δημιουργία της.

Ένα από τα πιο απλά προγράμματα στην γλώσσα Python είναι η εμφάνιση ενός γραπτού αποτελέσματος (π.χ. Γεια σου, κόσμε!):

```
>>>print("Γεια σου, κόσμε!")  
Γεια σου, κόσμε!
```

Ένα ιδιαίτερο χαρακτηριστικό της γλώσσας είναι η χρήση κενών διαστημάτων (whitespace) για τον διαχωρισμό των συντακτικών δομών που προγράμματος, σε αντίθεση με την πρακτική σε άλλες γλώσσες όπου για τον ίδιο σκοπό χρησιμοποιούνται ειδικά σύμβολα (πχ αγκύλες). Αυτό, σε συνδυασμό με το ότι χρησιμοποιεί πλήρεις αγγλικές λέξεις στη θέση συμβόλων, καθιστούν τον κώδικα της Python ευανάγνωστο από όσους έχουν βασική γνώση των αγγλικών. Για παράδειγμα, ο παρακάτω γεννήτορας (generator) generate_primes παράγει πρώτους αριθμούς:

```
from itertools import count  
def generate_primes(stop_at=None):  
    primes = []  
    for n in count(2):  
        if stop_at is not None and n > stop_at:  
            return  
        composite = False  
        for p in primes:  
            if not n % p:  
                composite = True
```



```
break
elif p**2 > n:
    break
if not composite:
    primes.append(n)
yield n
```

Και για να καλέσουμε τη συνάρτηση, υπολογίζοντας τους πρώτους αριθμούς από το 1 στο 100:

```
for i in generate_primes():
    if i > 100: break
    print(i)
```

Μειονεκτήματα

- Γλώσσα Ερμηνευσης
- Δύσκολη μετάφραση σε άλλη γλώσσα προγραμματισμού
- Δεν είναι κατάλληλη για λειτουργίες που έχουν να κάνουν με μνήμη

Πλεονεκτήματα

- Καθαρός κώδικας
- Επεκτασιμότητα
- Portability
- Αντικειμενοστραφής προγραμματισμός
- Δωρεάν
- Scripting language for web applications
- Scientific computing (βιβλιοθήκες)

3.1.5 JavaScript

Η γλώσσα JavaScript είναι μια γλώσσα σεναρίων (scripting language) που βοηθά τον προγραμματιστή στην κατασκευή διαδραστικών και λειτουργικών εγγράφων στο Internet. Τα σεναρία γραμμένα σε JavaScript ενσωματώνονται στον κώδικα HTML και εκτελούνται από τον web browser να εκτελέσει ένα σεναριο γραμμένο σε JavaScript θα πρέπει να διαθέτει έναν διερμηνευτή JavaScript (interpreter). Οι γνωστότεροι browsers Firefox, Internet Explorer, Opera ενσωματώνουν διερμηνευτή της JavaScript.

Μεταξύ των δυνατοτήτων που δίνει η JavaScript είναι:

- Έλεγχος στο περιεχόμενο και στην εμφάνιση των ιστοσελίδων
- Έλεγχος της συμπεριφοράς και των λειτουργιών του browser
- Αλληλεπίδραση με φόρμες HTML
- Αλληλεπίδραση με το χρήστη με τη βοήθεια γεγονότων
- Ανάγνωση ή καταγραφή του H/Y του χρήστη μέσω cookies
- Δυνατότητα για εναλλαγή εικόνων

Από άποψη περιορισμών, η JavaScript δεν επιτρέπει προσπέλαση ή χειρισμό αρχείων στο H/Y του χρήστη, για λόγους ασφαλείας. Επίσης, τα σεναρία που εκτελούνται στο web browser του χρήστη δεν μπορούν να ενημερώσουν αρχεία που βρίσκονται στο web server. Πολλοί πιστεύουν ότι το JavaScript είναι Java λόγω της ομοιότητας του ονόματος. Αυτό όμως δεν είναι αλήθεια. Πιστεύουμε ότι θα έπαιρνε πολύ να καταλάβουμε όλες τις διαφορές - γι' αυτό απλώς απομνημονεύουμε ότι το JavaScript δεν είναι Java.



Η JavaScript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Η χρήση της λέξης «Java» στο όνομα της γλώσσας έχει περισσότερη σχέση με το προφίλ του προϊόντος ή άλλη στενή σχέση με την Java. Ρόλο σε αυτήν τη σύγχυση έπαιξε και ότι η Java και η JavaScript έχουν δεχτεί σημαντικές επιρροές από την γλώσσα C, ειδικά στο συντακτικό, ενώ είναι και οι δυο αντικειμενοστραφείς γλώσσες. Τονίζεται ότι ο σωστός τρόπος γραφής της είναι «JavaScript» και όχι «Java Script» σαν δύο λέξεις, όπως λανθασμένα γράφεται ορισμένες φορές.

Τι χρειάζεται ώστε να τρέξουν scripts γραμμένα σε JavaScript; Χρειαζόμαστε ένα browser που υποστηρίζει JavaScript – παραδείγματος χάριν το Netscape Navigator (από την έκδοση 2.0) ή το Microsoft Internet Explorer (MSIE εν συντομία – από την έκδοση 3.0). Εφόσον αυτοί οι δύο browsers είναι πολύ διαδεδομένοι, πολλοί χρήστες έχουν την δυνατότητα να τρέξουν scripts γραμμένα σε JavaScript.

Αυτό είναι βεβαίως ένας σημαντικός λόγος για να διαλέξουμε JavaScript για να εμπλουτίσουμε τις web σελίδες μας. Βεβαίως χρειάζεται πρώτα να κατανοήσουμε βασικές λειτουργίες της html.

Ένα παράδειγμα κώδικα JavaScript το βλέπουμε παρακάτω :

```
<html>
<body>
<h1> My first Web Page </h1>
<script type =”text/ JavaScript”>
Document.write(“<p>” + Date () + “</p> ”) ;
</script>
</body >
< /html>
```

Ο παραπάνω κώδικας θα μας εμφανίσει στην οθόνη το παρακάτω: **My first Web Page.**

Σε browsers που δεν υποστηρίζουν JavaScript, θα εμφανιστεί το JavaScript, όπως το περιεχόμενο της σελίδας. Για να τους εμποδίσουμε να κάνουν κάτι τέτοιο θα πρέπει να χρησιμοποιηθεί η ετικέτα σχολίου HTML για να «κρύψει» το JavaScript.

Απλά προσθέτουμε ένα σχόλιο HTML ετικέτα (στο τέλος του σχολίου) μετά την τελευταία δήλωση της JavaScript, όπως αυτός < - - πριν από την πρώτη δήλωση JavaScript, και ένα «!»), όπως το παρακάτω π.χ

```
<html>
<body>
<script type =”text/ JavaScript”>
<!--
document.getElementById(“demo”).innerHTML=Date();
// - >
</script>
</body >
< /html>
```

Μοντέλο εκτέλεσης

Η αρχική έκδοση της Javascript βασίστηκε στη σύνταξη στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες.

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο φυλλομετρητής (browser) του χρήστη, και χαρακτηρίστηκε σαν *client-side γλώσσα προγραμματισμού*. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (*server-side γλώσσες προγραμματισμού*).



Παρά την ευρεία χρήση της Javascript για συγγραφή προγραμμάτων σε περιβάλλον φυλλομετρητή, αξίζει να σημειωθεί ότι από την αρχή χρησιμοποιήθηκε και για τη συγγραφή κώδικα από την πλευρά του διακομιστή, από την ίδια τη Netscape στο προϊόν LiveWire, με μικρή επιτυχία. Η χρήση της Javascript στο διακομιστή εμφανίζεται πάλι σήμερα, με τη διάδοση του Node.js, ενός μοντέλου προγραμματισμού βασισμένο στα γεγονότα (events).

Δείγμα κώδικα Javascript

Ο κώδικας Javascript μιας σελίδας περικλείεται από τις ετικέτες της HTML `<script type="text/javascript">` και `</script>`. Για παράδειγμα, ο ακόλουθος κώδικας Javascript εμφανίζει ένα πλαίσιο διαλόγου με το κείμενο "Γεια σου, κόσμε!":

```
<script type="text/javascript">
alert("Γεια σου, κόσμε!");
</script>
```

Αν ο κώδικας Javascript περιέχει περισσότερες από μία εντολές, αυτές θα πρέπει να διαχωριστούν μεταξύ τους με το χαρακτήρα του ελληνικού ερωτηματικού ';' (δηλαδή της λατινικής άνω τελείας). Η χρήση του χαρακτήρα αυτού για την τελευταία εντολή δεν είναι απαραίτητη. Η διαχώριση των εντολών στους νεότερους φυλλομετρητές (browsers) δεν είναι απαραίτητη.

Μια άλλη βασική εντολή, η `window.prompt("μήνυμα προς το χρήστη")`, ζητάει από το χρήστη να συμπληρώσει ένα κομμάτι μιας αίτησης απευθείας ώστε τα δεδομένα να χρησιμοποιηθούν σαν κείμενο:

```
<script>
var FIRSTvariable = window.prompt("PLEASE FILL IN YOUR NAME")
alert("Your name is " + FIRSTvariable + ".")
</script>
```

Πλεονεκτήματα :

Η JavaScript είναι μια εξαιρετική λύση για τις εφαρμογές, όταν χρησιμοποιούνται φόρμες εισόδου από την πλευρά του client. Αυτό σημαίνει ότι εάν ένας χρήστης ξεχάσει να εισάγει το όνομα του σε μια φόρμα, μια λειτουργία JavaScript επικύρωσης μπορεί να εμφανίσει αναδυόμενο ένα μήνυμα για να τον ενημερώσει σχετικά με την παράλειψη. Αυτή είναι μια πολύ καλύτερη λύση από το να έχει η πλευρά του server μια ρουτίνα επικύρωσης για να χειριστεί το σφάλμα, επειδή ο server δεν έχει να κάνει οποιαδήποτε πρόσθετη επεξεργασία. Επίσης, μια ρουτίνα asp ή php θα μπορούσε να γραφτεί για να επιτευχθεί το ίδιο έργο, αλλά η JavaScript δεν θα επέτρεπε στη φόρμα να υποβληθεί εάν δεν είχε συμπληρωθεί σωστά από την αρχή, μια πολύ πιο ισχυρή λύση!

Ένας άλλος τομέας όπου η JavaScript υπερέχει είναι η δημιουργία δυναμικών εφέ, όπως rollover εικόνες και scripted slideshows, όπου η χρήση της έχει γίνει κοινή. Επειδή η JavaScript τρέχει μέσα στο πρόγραμμα περιήγησης του πελάτη μπορεί να χρησιμοποιηθεί για να αλλάξει την εμφάνιση της οθόνης των χρηστών αφού έχει ολοκληρωθεί η αποστολή της σελίδας από τον server. Αυτό του επιτρέπει να δημιουργήσει μερικά πολύ εντυπωσιακά δυναμικά αποτελέσματα εικόνας.

Μειονεκτήματα :

Ένα από τα σημαντικότερα μειονεκτήματα για την χρήση JavaScript είναι ότι τείνει να φορτώνει τις ιστοσελίδες. Ο κώδικας JavaScript μπορεί γρήγορα να προσθέσει εκατοντάδες γραμμές κώδικα, αν το χρησιμοποιούν για να κάνουν οτιδήποτε, ακόμη και εξ αποστάσεως, ενδιαφέρον. Το πρόβλημα του μεγάλου κώδικα JavaScript είναι εύκολο να λυθεί με την αποθήκευση του κώδικα σε ξεχωριστά αρχεία πηγαίου κώδικα JavaScript που έχουν μια επέκταση .Js . Αυτό καθαρίζει τον κώδικα της σελίδας, επειδή ο κώδικας JavaScript αποθηκεύεται χωριστά στην ίδια σελίδα HTML, αφήνοντας μια πολύ καθαρότερη και πιο εύχρηστη ιστοσελίδα.



Λόγω της τάσης της JavaScript να φορτώνει τις ιστοσελίδες μπορεί να είναι πολύ επιζήμια για την μηχανή αναζήτησης της ιστοσελίδας σας. Αυτό συμβαίνει γιατί όταν μια μηχανή αναζήτησης φθάνει στο site σας να ψάχνει για ποιοτικό περιεχόμενο και λέξεις – κλειδιά για να καθορίσει το περιεχόμενο της σελίδας σας. Το τελευταίο πράγμα που θέλει να δει είναι εκατοντάδες γραμμές κώδικα JavaScript. Και πάλι, το πρόβλημα αυτό λύνεται εύκολα με τακτοποιημένη αποθήκευση κώδικα JavaScript στα αρχεία με επέκταση .Js και σύνδεση με το αρχείο δέσμης ενεργειών σε έγγραφο HTML σας.

3.1.6 JSP(Java Server Pages)

JavaServer Pages (JSP) είναι μια Java τεχνολογία που βοηθά τους προγραμματιστές λογισμικού να εξυπηρετήσει δυναμικά παραγόμενες ιστοσελίδες που βασίζονται σε HTML , XML , ή άλλους τύπους εγγράφων. Κυκλοφόρησε το 1999, ως απάντηση της Sun στην ASP και PHP , η JSP σχεδιάστηκε για να αντιμετωπίσει την αντίληψη ότι το περιβάλλον προγραμματισμού Java δεν παρέχει στους προγραμματιστές αρκετή υποστήριξη για τον Ιστό.

Για να αναπτύξετε και να εκτελέσετε ένα συμβατό web server με Servlet απαιτείται container. Η Java Servlet και το JavaServer Pages (JSP) προδιαγραφές από την Sun Microsystems και την JCP πρέπει αφενός να καλυφθούν από το container.

Επισκόπηση

Αρχιτεκτονικά, JSP μπορεί να θεωρηθεί ως μια υψηλού επιπέδου αφαίρεσης της Java servlets . Οι σελίδες JSP φορτώνονται στον server και λειτουργούν από ένα δομημένο εγκατασταθεί ειδικό πακέτο Java server που λέγεται Java EE Web Application, με κατάληξη ως .war ή .ear αρχεία.

Η JSP επιτρέπει κώδικα Java και ορισμένες προκαθορισμένες δράσεις που θα περιλαμβάνονται με στατικό περιεχόμενο σήμανσης web, με τη τελική σελίδα να δημιουργείται και να εκτελείται στον server για να παραδώσει ένα HTML ή XML έγγραφο. Η δημιουργημένες σελίδες και οι εξαρτώμενες βιβλιοθήκες Java χρησιμοποιούν Java bytecode αντί για ένα κλασσικό λογισμικό, και συνεπώς πρέπει να εκτελεστεί μέσα σε εικονική μηχανή Java (JVM) που ενσωματώνει το λειτουργικό σύστημα υποδοχής να παρέχει μια αφηρημένη ανεξάρτητα από πλατφόρμα περιβάλλον.

Η JSP σύνταξη είναι ένα ρευστό μίγμα των δύο βασικές μορφές περιεχομένου: τα *στοιχεία scriptlet* και Markup. Το Markup είναι συνήθως πρότυπο HTML ή XML, ενώ τα *scriptlet* στοιχεία είναι οριοθετημένα blocks του κώδικα της Java που μπορεί να είναι αναμειγμένα με το Markup. Όταν ζητηθεί η σελίδα εκτελείται ο κώδικα Java και προστίθεται το αποτέλεσμα, σε συνδυασμό με τον περιβάλλοντα Markup για να δημιουργήσετε την τελευταία σελίδα. Οι σελίδες JSP πρέπει να καταρτίζονται σε κλάσεις Java bytecode πριν να μπορούν να εκτελούνται, αλλά η σύνταξη είναι απαραίτητη μόνο όταν έχει συμβεί μια αλλαγή στην πηγή JSP αρχείο.

Ο Java κώδικας δεν απαιτείται να είναι πλήρης (αυτόνομα) εντός ενός *scriptlet* μπλοκ, αλλά μπορεί να μην συμπίπτει με τη σήμανση του περιεχομένου που παρέχει τη σελίδα στο σύνολό της αν είναι συντακτικά σωστή (για παράδειγμα, οποιοδήποτε Java / if / for/while, οι οποία χρησιμοποιείται σε ένα στοιχείο *scriptlet* και πρέπει να κλείσει σωστά σε ένα μεταγενέστερο στοιχείο για την σελίδα για την επιτυχή μεταγλώττιση).

Αυτό το σύστημα του διασπασμένου εσωτερικού τομέα κώδικα ονομάζεται *step over scripting* επειδή μπορεί να τυλίξει γύρω από τη στατική σήμανσης με την ενίσχυση από πάνω του. Η σήμανση η οποία εμπίπτει μέσα σε μια διάσπαση μπλοκ του κώδικα υπόκειται στις εν λόγω κώδικα, έτσι η σήμανση μέσα σε ένα *if* μπλοκ, θα εμφανίζονται μόνο στην έξοδο, όταν η συνθήκη *if* αποτιμάται σε true. Επίσης σήμανση μέσα σε μια κατασκευή βρόχου μπορεί να εμφανιστεί πολλές φορές στην έξοδο, ανάλογα με πόσες φορές λειτουργεί το σώμα βρόχου.

Η σύνταξη JSP προσθέτει επιπλέον XML ετικέτες, που ονομάζεται JSP actions, να επικαλεσθεί ενσωματωμένη λειτουργικότητα. Επιπλέον, η τεχνολογία επιτρέπει τη δημιουργία των βιβλιοθηκών tag JSP που λειτουργούν ως επεκτάσεις στο πρότυπο ετικέτας HTML ή XML. Οι JVM βιβλιοθήκες tag παρέχουν μια ανεξάρτητη πλατφόρμα για την αύξηση των δυνατοτήτων ενός web server. Σημειώστε ότι δεν είναι όλοι οι εμπορικοί διακομιστές Java συμβατοί με προδιαγραφές Java EE.



Από την έκδοση 1.2 των προδιαγραφών JSP, οι JavaServer Pages έχουν αναπτυχθεί στο πλαίσιο της Διαδικασίας της Κοινοτικής Java . Το JSR 53 καθορίζει το JSP 1.2 και το JSP Servlet 2.3 και το JSR 152 ορίζει τις JSP 2.0 προδιαγραφές.

Από το Μάιο του 2006, η JSP 2.1 έχει κυκλοφορήσει κάτω από JSR 245 ως μέρος της Java EE 5. Από τις 10 Δεκέμβρη του 2009 η JSP 2.2 προδιαγραφές έχει κυκλοφορήσει ως ένα δελτίο συντήρησης JSR 245.

Παράδειγμα

Η JSPs καταρτίζονται σε servlets από ένα μεταγλωττιστή JSP . Ο compiler δημιουργεί ένα servlet σε κώδικα Java που στη συνέχεια καταρτίζεται από τον μεταγλωττιστή της Java, ή μπορεί να συγκεντρώνει τα servlet σε byte κώδικα η οποία είναι άμεσα εκτελέσιμη. Η JSPs μπορεί επίσης να ερμηνευθεί on-the-fly, στην μείωση του χρόνου που απαιτείται για να φορτώσετε τις αλλαγές.

Ανεξάρτητα από το αν ο compiler JSP παράγει πηγαίο κώδικα Java για ένα servlet ή εκπέμπει το byte κώδικα άμεσα, είναι χρήσιμο να κατανοήσουμε πώς ο JSP compiler μετατρέπει την σελίδα σε ένα servlet Java. Για π.χ:

Είσοδος JSP

```
<%@ page errorPage="myerror.jsp" %>
<%@ page import="com.foo.bar" %>
<html>
<head>
<%! int serverInstanceVariable = 1;%>
<% int localStackBasedVariable = 1; %>
<table>
<tr><td><%= toStringOrBlank( "expanded inline data " + 1 ) %></td></tr>
```

JSP 2.0

Η νέα μορφή των προδιαγραφών του JSP περιλαμβάνει νέα χαρακτηριστικά ως στόχο τη βελτίωση της παραγωγικότητας του προγραμματιστή. Δηλαδή:

- Μια γλώσσα έκφρασης (EL), η οποία επιτρέπει στους προγραμματιστές να δημιουργήσουν Velocity τύπου πρότυπα (μεταξύ άλλων).
- Ο ταχύτερος / ευκολότερος τρόπος για να εμφανίσετε τις τιμές παραμέτρου.
- Ένας σαφής τρόπος για να περιηγηθείτε στα nested beans.

Η JSP 2.0 εισήγαγε ένα πρόβλημα στο τμήμα της βιβλιοθήκης ετικέτας σχετικά με το πώς οι πληροφορίες έκδοσης JSP εκπροσωπήθηκαν. Η ίδια η προδιαγραφή έρχεται σε αντίθεση, μερικές φορές αναφέρεται σε μια JSP-έκδοση στοιχείου, και άλλες φορές ως ένα χαρακτηριστικό έκδοσης. Και η JSP 2.1 αξιοποιεί τις Servlet 2.5 προδιαγραφές για τη σημασιολογία του Ιστού.

Πλεονεκτήματα :

Η JSP έχει διάφορα πλεονεκτήματα πέρα από τις πολλές εναλλακτικές λύσεις της. Παρακάτω παρουσιάζονται μερικά από αυτά και τα οποία είναι:

- Είναι ευκολότερη και ταχύτερη η ανάπτυξη τους, κυρίως για μικρά Projects.
- Με τους servlets είναι δυσκολότερη η παραγωγή στατικού HTML κώδικα, χρειάζονται πολλές out.println εντολές, π.χ
out.println("<body>");
out.println("<h2> hello "+ user + "</ h2>");



- Δίνει έμφαση στην μορφή και παρουσίαση της σελίδας (HTML) και αφήνουν την προγραμματιστική λογική σε Java κώδικα.
- Αν έχουμε έτοιμο JSP κώδικα, μπορεί εύκολα να ενσωματωθεί σε HTML σελίδες από web authors που δε γνωρίζουν προγραμματισμό.
- Σε σύγκριση με τις Active Server Pages (ASP):

Η ASP είναι μια ανταγωνιστική τεχνολογία προερχόμενη από τη Microsoft. Τα πλεονεκτήματα της JSP είναι διπλά. Κατ' αρχάς, το δυναμικό μέρος γράφεται σε Java, κι όχι σε VBScript ή μια άλλη ASP-specific γλώσσα, κι έτσι αυτό την κάνει να είναι ισχυρότερη και καλύτερη που ταιριάζει στις σύνθετες εφαρμογές που απαιτούν τα επαναχρησιμοποιήσιμα συστατικά. Δεύτερον, η JSP είναι φορητή σε άλλα λειτουργικά συστήματα και Servers δικτύου στα οποία δεν είμαστε κλειδωμένοι στα Windows NT/2000 και IIS. Θα μπορούσαμε να προβάλλουμε το ίδιο επιχείρημα κατά τη σύγκριση της JSP με ColdFusion με JSP που μπορούμε να χρησιμοποιήσουμε την Java και δεν είμαστε δεμένοι σε ένα κατάλληλο προϊόν Server.

- Σε σύγκριση με την PHP:

Η PHP είναι ελεύθερη, ανοιχτού – κώδικα HTML – embedded scripting γλώσσα που είναι κάπως παρόμοια τόσο με την ASP, όσο επίσης, και με την JSP. Το πλεονέκτημα της JSP είναι ότι το δυναμικό μέρος γράφεται σε Java, για την οποία πιθανώς ήδη είμαστε ενήμεροι, το οποίο έχει ήδη ένα εκτενές API για τη δικτύωση, την πρόσβαση των βάσεων δεδομένων, τα διανεμημένα αντικείμενα, και τους ομοίους, ενώ η PHP απαιτεί μια εξ' ολοκλήρου νέα γλώσσα.

- Σε σύγκριση με τα καθαρά Servlets:

Η JSP δεν παρέχει οποιεσδήποτε ικανότητες που δεν θα μπορούσαν να ολοκληρωθούν σε γενικές γραμμές με ένα Servlet. Στην πραγματικότητα, τα έγγραφα JSP είναι αυτόματα μεταφρασμένα στα Servlets πίσω από τις σκηνές. Αλλά είναι καταλληλότερη να γράψει, αλλά και να τροποποιήσει κανονικό HTML από το να έχει τεράστιο πλήθος δηλώσεων `println` που παράγουν το HTML. Επιπλέον, με το χωρισμό της παρουσίας από το περιεχόμενο, μπορούμε να βάλουμε διαφορετικούς ανθρώπους σε διαφορετικούς στόχους: οι δικοί μας εμπειρογνώμονες σχεδίου ιστοσελίδας, μπορούν να χτίσουν το HTML χρησιμοποιώντας τα εξοικειωμένα εργαλεία και να αφήσουν τις θέσεις Για τους προγραμματιστές μας των Servlet για να παρεμβάλουν το δυναμικό Περιεχόμενο.

- Σε σύγκριση με τα Server – side Includes (SSI) :

Τα SSI είναι μια ευρέως υποστηριγμένη τεχνολογία για την είσοδο των εξωτερικά καθορισμένων κομματιών στατικής ιστοσελίδας. Η JSP είναι καλύτερη επειδή έχουμε ένα πλουσιότερο σύνολο εργαλείων για το χτίσιμο αυτού του εξωτερικού κομματιού κι έχουμε περισσότερες επιλογές σχετικά με το στάδιο της απάντησης HTML στην οποία το κομμάτι εισέρχεται πραγματικά. Εκτός αυτού, τα SSI προορίζονται πραγματικά μόνο για τους form data, κάνουν τις συνδέσεις των βάσεων δεδομένων, και τους ομοίους.

- Σε σύγκριση με την JavaScript :

Η JavaScript, που είναι απολύτως ευδιάκριτη από τη γλώσσα προγραμματισμού της Java, χρησιμοποιείται κανονικά για να παράγει το HTML δυναμικά στον client, που χτίζει τα μέρη ιστοσελίδας δεδομένου ότι ο browser «φορτώνει» το έγγραφο. Αυτή είναι μια χρήσιμη ικανότητα αλλά μόνο στην περίπτωση όταν χειρίζεται τις καταστάσεις όπου οι δυναμικές πληροφορίες είναι βασισμένες στο περιβάλλον του client. Με εξαίρεση τα cookies, το στοιχείο αιτήματος HTTP δεν είναι διαθέσιμο στις client-side ρουτίνες JavaScript. Κι από τις ρουτίνες ελλείψεων JavaScript για των προγραμματισμό δικτύων, ο κώδικας JavaScript στον client δεν μπορεί να προσπελάσει server-side πόρους όπως τις βάσεις δεδομένων, τους καταλόγους, τις πληροφορίες τιμολόγησης, και τους ομοίους. Η JavaScript μπορεί επίσης να χρησιμοποιηθεί στον server, κι ειδικότερα στους Server Netscape κι ως scripting γλώσσα για IIS. Η Java είναι με μεγάλη διαφορά πιο ισχυρή, εύκαμπτη, αξιόπιστη, και φορητή.

- Σε σύγκριση με την στατική HTML:

Το κανονικό HTML, φυσικά δεν μπορεί να περιέχει τις δυναμικές πληροφορίες, κι έτσι οι στατικές σελίδες HTML δεν μπορούν να βασιστούν στις πηγές στοιχείων εισόδου ή στα server-side δεδομένα πηγών. Η JSP είναι τόσο εύκολη και κατάλληλη που είναι αρκετά λογικό να αυξηθούν οι σελίδες HTML που ωφελούνται μόνο ελαφρώς από την εισαγωγή των δυναμικών στοιχείων. Προηγουμένως η δυσκολία της χρησιμοποίησης των δυναμικών στοιχείων απέκλεισε τη χρήση της σε όλες εκτός από



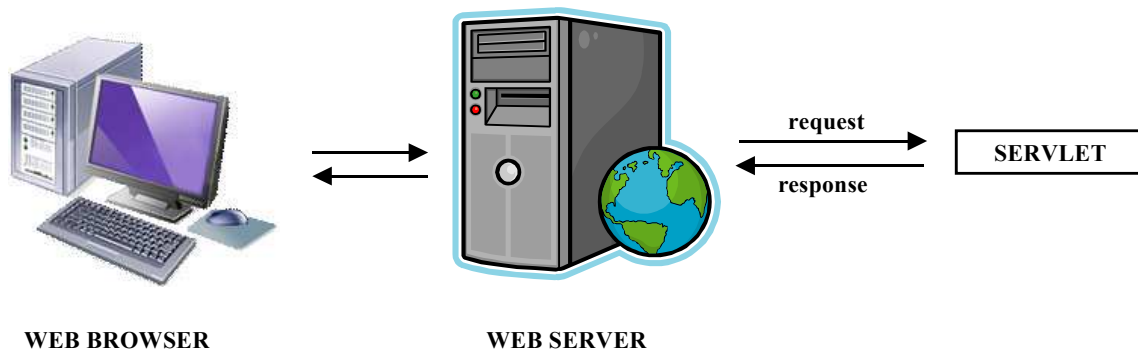
τις πολυτιμότερες περιπτώσεις.

Μειονεκτήματα :

- Εύρεση και διόρθωση λαθών (debugging) είναι πολύπλοκη.
- Μεγάλες ποσότητες ενσωματωμένου κώδικα σε scriplets μπορεί να οδηγήσουν σε σελίδες που είναι δύσκολο να συντηρηθούν.
- Η απόδοση JSPs είναι κατώτερη των Servlet (to compilation 200 JSPs θα δημιουργήσει 200 Servlets).
- Ας χρησιμοποιούσαμε τεχνολογία Servlets θα μπορούσαμε να ενσωματώσουμε περισσότερη λειτουργικότητα σε λιγότερους Servlets.

3.1.7 Java Servlet

Ένα servlet είναι μια κλάση προγραμματισμού Java γλώσσα που χρησιμοποιείται για να επεκτείνουν τις δυνατότητες των servers που φιλοξενούν τις εφαρμογές πρόσβασης μέσω ενός μοντέλου προγραμματισμού αίτημα-απάντηση. Αν και οι servlets μπορούν να ανταποκριθούν σε κάθε τύπο αιτήματος, τα μέρη που χρησιμοποιούνται συνήθως είναι για την επέκταση των εφαρμογών που φιλοξενείται από τους Web servers. Έτσι, μπορεί να θεωρηθεί ως μια βοηθητική εφαρμογή Java που τρέχει σε ένα διακομιστή αντί για ένα πρόγραμμα περιήγησης.



Εικόνα 2: Η λειτουργία των servlet.

Ένα **Servlet** είναι μια κλάση Java σε Java EE που συμμορφώνεται με το **Java Servlet API**, ένα πρωτόκολλο με το οποίο μια κλάση Java μπορεί να ανταποκριθεί στα αιτήματα. Δεν είναι συνδεδεμένη με ένα συγκεκριμένο πρωτόκολλο client-server, αλλά πιο συχνά χρησιμοποιείται με το HTTP πρωτόκολλο. Ως εκ τούτου, η λέξη "Servlet" χρησιμοποιείται συχνά κατά την έννοια του "Servlet HTTP". Έτσι, ένας προγραμματιστής μπορεί να χρησιμοποιήσει ένα servlet για να προσθέσει δυναμικό περιεχόμενο σε ένα διακομιστή Web χρησιμοποιώντας την πλατφόρμα Java. Το περιεχόμενο που δημιουργείται είναι συνήθως HTML, αλλά μπορεί να είναι και άλλα δεδομένα, όπως XML. Οι Servlets είναι το ομόλογό της Java για μη δυναμικές τεχνολογίες Java με περιεχόμενο στον Παγκόσμιο Ιστό, όπως CGI και ASP.NET. Οι Servlets μπορεί να διατηρήσουν σε μεταβλητές την κατάσταση περιόδου λειτουργίας σε πολλές συναλλαγές με τη χρήση διακομιστή HTTP cookies, ή URL επανεγγραφή.

Για να αναπτύξετε και να εκτελέσετε ένα Servlet, μπορεί να χρησιμοποιηθεί ο Apache Tomcat. Είναι ένα Servlet ανοιχτού κώδικα που έχει αναπτυχθεί από την Apache Software Foundation (ASF). Ο Tomcat εφαρμόζει τις Java Servlet και τις JavaServer Pages (JSP) προδιαγραφές από την Sun Microsystems, και παρέχει μια «καθαρή java» HTTP περιβάλλον του web server για κώδικα Java για να τρέξει.

Το Servlet API, που περιέχονται στο πακέτο Java ιεραρχία javax.servlet, καθορίζει τις αναμενόμενες αλληλεπιδράσεις των εμπορευματοκιβωτίων Web και ένα servlet. Ένα δοχείο Web



είναι ουσιαστικά το συστατικό ενός διακομιστή Web που αλληλεπιδρά με το servlets. Το δοχείο Web είναι υπεύθυνο για τη διαχείριση του κύκλου ζωής των servlets, χαρτογραφώντας ένα URL σε ένα συγκεκριμένο Servlet και την εξασφάλιση ότι ο αιτών URL έχει τα σωστά δικαιώματα πρόσβασης.

Ένα Servlet είναι ένα αντικείμενο το οποίο δέχεται αίτηση και παράγει μια απάντηση βασισμένο σε αυτό το αίτημα. Το βασικό πακέτο servlet καθορίζει Java αντικείμενα να αντιπροσωπεύουν αιτήματα Servlet και απαντήσεις, καθώς και αντικείμενα για να αντανακλά τις παραμέτρους διαμόρφωσης του servlet και το περιβάλλον εκτέλεσης. Το πακέτο javax.servlet.http καθορίζει HTTP υποκλάσεις των γενικών στοιχείων servlet, συμπεριλαμβανομένων των αντικειμένων της διαχείρισης συνεδρίας που παρακολουθούν πολλαπλά αιτήματα και οι απαντήσεις μεταξύ του Web server και ενός πελάτη. Τα Servlets μπορούν να συσκευαστούν σε ένα αρχείο WAR ως μια εφαρμογή Web .

Οι Servlets μπορούν να παραχθούν αυτόματα από JavaServer Pages (JSP) από τον compiler JavaServer Pages . Η διαφορά μεταξύ Servlets και JSP είναι ότι οι Servlets ενσωματώνονται συνήθως μέσα σε HTML κώδικα Java, ενώ η JSPs ενσωματώνεται Java κώδικα σε μορφή HTML. Ενώ η άμεση χρήση των Servlets να παράγουν HTML (όπως φαίνεται στο παρακάτω παράδειγμα) είναι σχετικά σπάνια στις μέρες μας, το υψηλότερο επίπεδο MVC web πλαισίου σε Java EE (JSF) εξακολουθεί να χρησιμοποιεί ρητά τη Servlet τεχνολογία για το χαμηλό επίπεδο χειρισμού αίτηση / απάντηση μέσω της FacesServlet. Μια κάπως μεγαλύτερης ηλικίας χρήση είναι η χρήση servlets, σε συνδυασμό με JSPs σε ένα σχέδιο που ονομάζεται " Μοντέλο 2 ", η οποία είναι μια γεύση από το μοντέλο view-controller.

Ιστορία

Η πλήρης προδιαγραφή servlet δημιουργήθηκε από την Sun Microsystems , με την έκδοση 1.0 που ολοκληρώθηκαν τον Ιούνιο του 1997. Από την έκδοση 2.3, οι προδιαγραφές servlet αναπτύχθηκε στο πλαίσιο της Διαδικασίας της Κοινότητας Java. Η JSR 53 ορίζει Servlet 2.3 και JavaServer Page 1.2 προδιαγραφές. Η JSR 154 διευκρινίζει τις Servlet 2.4 και 2.5 προδιαγραφές. Από τις 26 Μάρτιου 2010, η τρέχουσα έκδοση των προδιαγραφών servlet είναι 3.0.

Στο blog του στο java.net, η Sun βετεράνος και ο GlassFish οδηγούν τον Jim Driscoll με λεπτομέρειες από την ιστορία της servlet τεχνολογίας. Ο James Gosling σκέφτηκε πρώτος τους servlets κατά τις πρώτες ημέρες της Java, αλλά η ιδέα δεν είχε γίνει προϊόν μέχρι να αποσταλεί στο Web Server Java από την Sun . Αυτό ήταν πριν από αυτό που είναι τώρα η πλατφόρμα Java Platform, η έκδοση Enterprise έγινε σε μια προδιαγραφή.

Πλεονεκτήματα σε σχέση με CGI

Τα πλεονεκτήματα από τη χρήση servlets είναι γρήγορες επιδόσεις τους και την ευκολία χρήσης σε συνδυασμό με περισσότερη δύναμη πέρα από την παραδοσιακή CGI (Common Gateway Interface). Τα παραδοσιακά CGI scripts γραμμένα σε Java έχουν μια σειρά από μειονεκτήματα, όταν πρόκειται για την απόδοση:

- Όταν ένα αίτημα HTTP γίνεται, μια νέα διαδικασία δημιουργείται για κάθε κλήση του CGI script. Αυτή η επιβάρυνση της δημιουργίας διαδικασίας μπορεί να είναι ένα σύστημα υψηλής έντασης, ειδικά όταν το σενάριο έχει αρκετά γρήγορα εργασίες. Έτσι, η δημιουργία της διαδικασίας θα χρειαστεί περισσότερο χρόνο από ό, τι η CGI εκτέλεση του κώδικα. Τα Java servlets το έλυσαν αυτό, καθώς το servlet δεν είναι μια ξεχωριστή διαδικασία. Κάθε αίτηση για να ικανοποιούνται από ένα servlet γίνεται από ένα ξεχωριστό νήμα Java στο πλαίσιο της διαδικασίας διακομιστή Web, παραλείποντας ξεχωριστή διακλάδωση από το δαίμονα HTTP.
- Ταυτόχρονα CGI αίτηματα έχουν σαν αποτέλεσμα το CGI script να αντιγραφεί και να φορτωθεί στη μνήμη όσες φορές υπάρχουν αιτήματα. Ωστόσο, τα servlets, υπάρχουν στην ίδια ποσότητα θεμάτων και αιτημάτων, αλλά υπάρχει μόνο ένα αντίγραφο του servlet class που δημιουργείται στη μνήμη και μένει εκεί μεταξύ των αιτημάτων.
- Μόνο ένα παράδειγμα απαντά σε όλα τα αιτήματα ταυτόχρονα. Αυτό μειώνει τη χρήση της μνήμης και καθιστά τη διαχείριση των επίμονων δεδομένων εύκολη.



- Ένα servlet μπορεί να διευθύνεται από ένα servlet κινητήρα σε ένα περιοριστικό περιβάλλον, που ονομάζεται sandbox . Αυτό είναι παρόμοιο με ένα applet που τρέχει στο sandbox του προγράμματος περιήγησης στο Web. Αυτό κάνει μια περιοριστική χρήση των δυνητικά επιβλαβείς servlets δυνατό.

Ο κύκλου ζωής ενός servlet

1. Το δοχείο ζητά τον no-arg constructor .
2. Το δοχείο Web καλεί την init() μέθοδο. Η μέθοδος αυτή αρχικοποιεί το servlet και πρέπει να κληθεί πριν από τη ζωή του ενός servlet, η init() μέθοδος καλείται μόνο μία φορά.
3. Μετά την προετοιμασία, το servlet μπορεί να εξυπηρετήσει αιτήματα πελατών. Κάθε αίτηση εξυπηρετείται σε δικό του ξεχωριστό νήμα της. Το δοχείο Web κλήσεις της service() μέθοδο του servlet για κάθε αίτηση. Η service() μέθοδος αυτή προσδιορίζει το είδος του αιτήματος που γίνονται και το μετατρέπει σε μια κατάλληλη μέθοδο για να χειριστεί το αίτημα. Ο δημιουργός του servlet πρέπει να παρέχει μια υλοποίηση για τις μεθόδους αυτές. Αν μια αίτηση για μια μέθοδο που δεν τίθεται σε εφαρμογή από το servlet γίνεται, καλείται η μέθοδος της γονικής κλάσης, συνήθως με αποτέλεσμα ένα σφάλμα που επιστρέφεται στον αιτούντα.
4. Τέλος, το δοχείο Web ζητά την destroy() μέθοδος λαμβάνετε από το servlet της υπηρεσίας. Η destroy() μέθοδος, όπως και η init(), καλείται μόνο μία φορά στον κύκλο ζωής ενός servlet.

Τρεις μέθοδοι είναι κεντρικής σημασίας για τον κύκλο ζωής ενός servlet. Αυτά είναι init (), service(), και η destroy(). Υλοποιούνται από κάθε Servlet και γίνεται επίκληση σε συγκεκριμένες ώρες από τον διακομιστή. Ας εξετάσουμε ένα τυπικό σενάριο να καταλάβει ο χρήστης όταν καλούνται αυτές οι μέθοδοι.

1. Ας υποθέσουμε ότι ένας χρήστης μπαίνει σε μια Uniform Resource Locator (URL) σε ένα web browser.
 - Το πρόγραμμα περιήγησης, στη συνέχεια, δημιουργεί μια αίτηση HTTP για αυτό το URL.
 - Η αίτηση αυτή αποστέλλεται στη συνέχεια στον κατάλληλο διακομιστή.
2. Το αίτημα HTTP παραλαμβάνεται από τον web server.
 - Ο διακομιστής στέλνει αυτό το αίτημα σε ένα συγκεκριμένο servlet.
 - Τα servlet έχουν δυναμικά ανακτηθεί και έχουν τοποθετηθεί στον χώρο διευθύνσεων του διακομιστή.
3. Ο διακομιστής επικαλείται την init() μέθοδο του servlet.
 - Αυτή η μέθοδος καλείται μόνο όταν το servlet για πρώτη φορά φορτώνονται στη μνήμη.
 - Είναι δυνατόν να περάσετε ορίσματα αρχικοποίησης στο servlet έτσι ώστε να μπορεί να αυτορυθμιστεί.
4. Ο server ενεργοποιεί την service() μέθοδο του servlet.
 - Αυτή η μέθοδος καλείται να επεξεργαστεί την αίτηση HTTP.
 - Θα δείτε ότι είναι δυνατό για το servlet να διαβάσει τα δεδομένα που έχουν παρασχεθεί στην αίτηση HTTP.
 - Θα μπορεί επίσης να διατυπώνει μια απόκριση HTTP για τον πελάτη.
5. Το servlet παραμένει στο χώρο διευθύνσεων του server και είναι διαθέσιμο για την επεξεργασία κάθε άλλου αιτήματος HTTP που λαμβάνει από τους πελάτες.
 - Η service() μέθοδος καλείται για κάθε αίτηση HTTP.
6. Ο διακομιστής μπορεί, κάποια στιγμή, να αποφασίσει να ξεφορτώσουν το servlet από τη μνήμη του.
 - Οι αλγόριθμοι με την οποία ο προσδιορισμός γίνεται είναι ειδικές για κάθε server.
7. Ο διακομιστής καλεί την destroy() μέθοδος για να αποποιηθούν οποιουδήποτε πόρους, σημαντικά δεδομένα μπορούν να αποθηκευτούν σε ένα διαρκή κατάσταση.
8. Η μνήμη που διατίθεται για το servlet και τα αντικείμενά της μπορεί να συλλέξει και σκουπίδια.



Παράδειγμα

Το ακόλουθο παράδειγμα εκτυπώνει servlet ένα "Hello World" σε σελίδα HTML. Σημειώστε ότι HttpServlet είναι μια υποκατηγορία των GenericServlet, μια υλοποίηση του Servlet interface. Η service() αποστολές μεθόδων των αιτήσεων των μεθόδων doGet(), doPost(), doPut(), doDelete(), και ούτω καθεξής σύμφωνα με το αίτημα HTTP.

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class HelloWorld extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
    PrintWriter out = response.getWriter();
    out.println("<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 \" +
    \"Transitional//EN\">\n" +
    "<html>\n" +
    "<head><title>Hello World</title></head>\n" +
    "<body>\n" +
    "<h1>Hello, world!</h1>\n" +
    "</body></html>");
}
}
```

Χρήση

Τα Servlets χρησιμοποιούνται πιο συχνά για να :

- επεξεργασθούν και να αποθηκεύσουν τα δεδομένα που υποβλήθηκαν από μια φόρμα HTML
- παρέχει δυναμικό περιεχόμενο, όπως τα αποτελέσματα ενός ερωτήματος βάσης δεδομένων
- διαχειρίζεται πληροφορίες αναφέροντας ότι δεν υπάρχει στο ανιθαγενείς πρωτόκολλο HTTP, όπως η συμπλήρωση των άρθρων στο καλάθι αγορών του κατάλληλου πελάτη.

Πλεονεκτήματα

- Η αποδοτική χρήση (efficiency) - ένα servlet φορτώνεται δυναμικά στην μνήμη μία φορά, όταν η αντίστοιχη υπηρεσία (service) αρχικοποιηθεί και στην συνέχεια εξυπηρετεί τα requests με κλήση της μεθόδου service(), χωρίς να δημιουργεί μία καινούργια διεργασία (process) για κάθε request
- Η μονιμότητα (persistency) - μπορεί να διατηρεί την κατάστασή του ανάμεσα σε διαφορετικά requests που ανήκουν στο ίδιο session
- Η φορητότητα (portability) - λόγω του ότι είναι JAVA κλάση μπορεί να εκτελεστεί σε οποιαδήποτε JVM
- Η ευρωστία (robustness) - ως JAVA κλάση τα servlet κληρονομούν τις δυνατότητες της JAVA και επίσης έχουν πρόσβαση στην τεράστια βιβλιοθήκη της
- Η επεκτασιμότητα (extensibility) - μπορούν να χρησιμοποιήσουν τον μηχανισμό επέκτασης της JAVA σε αντικείμενα που εξυπηρετούν καλύτερα τις ανάγκες της υπηρεσίας (service)
- Η ασφάλεια (security) - εκτελούνται στον server και επομένως κληρονομούν την ασφάλεια που παρέχει αυτός
- Η ευρεία αποδοχή - ως JAVA κλάση δέχεται την υποστήριξη ενός μεγάλου αριθμού προγραμματιστών σε όλο τον κόσμο



Μειονεκτήματα

- Χρησιμοποιεί πολύτιμη επεξεργαστική ισχύ του server
- Η κλιμάκωση (scalability)

3.2 Τεχνολογίες Βάσεων Δεδομένων

3.2.1 MySQL

Η **MySQL** είναι μια σχεσιακή βάση δεδομένων που μετρά περισσότερες από 11 εκατομμύρια εγκαταστάσεις. Έλαβε το όνομά της από την κόρη του Μόντυ Βιντένιους, τη Μάι. Το πρόγραμμα τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων.

Ο κωδικός του εγχειρήματος είναι διαθέσιμος μέσω της GNU General Public License, καθώς και μέσω ορισμένων ιδιοκτητών συμφωνιών. Ανήκει και χρηματοδοτείται από μία και μοναδική κερδοσκοπική εταιρία, τη σουηδική MySQL AB, η οποία σήμερα ανήκει στην Oracle

Σύνταξη και εντολές

Η Mysql είναι ένα δυνατό σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων. Είναι ο περισσότερο διαδεδομένος database server και υποστηρίζεται από τα περισσότερα λειτουργικά συστήματα. Είναι λογισμικό ανοιχτού κώδικα (Open Source software) έτσι διατίθεται δωρεάν στην διεύθυνση <http://dev.mysql.com/downloads/mysql> με άδεια ανοιχτού κώδικα ή με χαμηλό κόστος για εμπορική άδεια. Ο database server είναι ένας πολυνηματικός πολλαπλών χρήσεων διακομιστής δηλαδή επιτρέπει να δουλεύουν πολλοί χρήστες ταυτόχρονα. Ελέγχει την πρόσβαση στα δεδομένα και επιτρέπει μόνο σε πιστοποιημένους χρήστες να έχουν πρόσβαση σε αυτά. Χρησιμοποιεί την SQL (Structured Query Language) γλώσσα ερωτημάτων για βάσεις δεδομένων.

Όπως όλα τα ΣΔΒΔ περιλαμβάνει εντολές για:

Διαχείριση χρηστών :

Δημιουργία : CREATE USER username@hostname IDENTIFIED BY password;

Διαγραφή χρήστη: DROP USER όνομα_χρήστη@hostname;

Διαχείριση βάσης δεδομένων :

Δημιουργία βάσης δεδομένων : CREATE DATABASE όνομα_βάσης;

Διαγραφή βάσης δεδομένων : DROP DATABASE όνομα_βάσης;

Εμφάνιση όλων των βάσεων που έχει ο χρήστης : SHOW DATABASES;

Σύνδεση με την βάση : USE όνομα_βάσης;

Διαχείριση Δικαιωμάτων σε χρήστη:

Εκχώρηση δικαιωμάτων σε χρήστη : GRANT [privilege] ON [db] [.table] TO user@host [IDENTIFIED BY password] [WITH GRANT OPTION];

Ανάκληση : REVOKE [privilege] ON [db] [.table] [.column] FROM user@host;

Διαχείριση πινάκων :

Δημιουργία πίνακα : CREATE TABLE ονομ_πίνακα (όνομ_πεδίου τύπος_πεδίου[NOT NULL] [DEFAULT value], πεδίο τύπος_πεδίου..., PRIMARY KEY (πεδία), [UNIQUE(πεδία)], [FOREIGN KEY (πεδίο σύνδεσης) REFERENCES ονομ_πίνακα(πεδίο σύνδεσης)] [ON DELETE RESTRICT| CASCADE| SET NULL| NO ACTION] [ON UPDATE RESTRICT| CASCADE| SET NULL| NO ACTION]);

Μεταβολή Δομής πίνακα: ALTER TABLE ονομ_πίνακα [ADD column πεδίο [first/alter πεδίο]] [ADD constraint περιορισμός] [ADD primary key/index/unique (πεδίο1, ...)] [MODIFY όνομ_πεδίου τύπος_πεδίου...] [DROP ,πεδίο, περιορισμός, κλειδί κλπ]

Διαγραφή πίνακα: DROP TABLE ονομ_πίνακα

Αλλαγή ονόματος πίνακα: rename table ονομ_πίνακα to νέο_όνομα

Εμφάνιση πινάκων : Show tables; ή Show tables from όνομα_βδ;

Εμφάνιση στηλών πίνακα : Describe ονομ_πίνακα; ή Show columns from ονομ_πίνακα;



Διαχείριση ευρετηρίων :

Δημιουργία ευρετηρίου : CREATE [UNIQUE] INDEX όνομα_ευρετηρίου ON όνομ_πίνακα(πεδίο [ASC/DESC][,πεδίο[ASC/DESC],]);

Διαγραφή ευρετηρίου : DROP INDEX όνομα_ευρετηρίου ON όνομ_πίνακα;

Εμφάνιση ευρετηρίου : Show index from όνομ_πίνακα;

Διαχείριση εγγραφών Πίνακα:

Εισαγωγή εγγραφών: INSERT INTO όνομ_πίνακα [(πεδίο1, πεδίο2....)] values (τιμή_πεδίου1, τιμή_πεδίου2....), (τιμή_πεδίου1, τιμή_πεδίου2....).....

Εισαγωγή εγγραφών από το text αρχείο: load data local infile 'διαδρομή/όνομα_αρχείου.txt' into table όνομ_πίνακα fields terminated by ',';

Εισαγωγή εγγραφών από πίνακα σε πίνακα: INSERT INTO όνομ_πίνακα SELECT πεδίο1 ή τιμή1, πεδίο2 ή τιμή2, FROM όνομ_πίνακα WHERE συνθήκη;

Ερώτημα διαγραφής εγγραφών: DELETE FROM όνομ_πίνακα WHERE συνθήκη;

Ερώτημα τροποποίησης εγγραφών: UPDATE όνομ_πίνακα SET πεδίο1 = τιμή1 [,πεδίο2 =τιμή2, ...] [WHERE συνθήκη]

Επιλογή εγγράφων από ένα πίνακα: SELECT [distinct] όνομ_πίνακα, FROM όνομ_πίνακα [WHERE συνθήκη] [order by όνομα_πεδίου [ASC/DESC],]

Όταν χρησιμοποιούμε παραπάνω από ένα πίνακες πρέπει να τους συνδέσουμε με INNER ή LEFT ή RIGHT JO

IN αλλιώς δημιουργείται καρτεσιανό γινόμενο π.χ

```
Select grades.am, lastname, firstname, code, semester, grade, from student left join grades on students.am=grades.am ;
```

Πλεονεκτήματα :

- Είναι ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων
- Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα
- Μπορούν να υπάρχουν ταυτόχρονα περισσότερες από μία συνδέσεις με τη βάση χωρίς να υπάρχουν πολλαπλά αντίγραφα της, όπως συμβαίνει με άλλα συστήματα βάσεων δεδομένων
- Η απόδοση της είναι καλύτερη σε μεγαλύτερο όγκο βάσεων δεδομένων
- Παρέχει ευκολίες στο backup
- Είναι ιδιαίτερα βελτιωμένη σε ταχύτητα για την ανάκτηση δεδομένων
- Είναι συμβατή και μεταφέρσιμη σε διάφορες πλατφόρμες και για διάφορα εργαλεία ανάπτυξης
- Είναι πιο κατάλληλη για χρήση στο Internet
- Είναι οικονομική
- Η MySQL είναι λογισμικό ανοιχτού κώδικα

Μειονεκτήματα :

- Η MySQL δεν υποστηρίζει ένα πολύ μεγάλο μέγεθος της βάσης δεδομένων πολύ αποτελεσματικά
- Η MySQL δεν υποστηρίζει ROLE, COMMIT και Stored διαδικασίες στις εκδόσεις της κάτω από 5
- Οι συναλλαγές δεν αντιμετωπίζονται πολύ αποτελεσματικά

3.2.2 Oracle Database

Το Oracle Database είναι λογισμικό διαχείρισης σχεσιακών βάσεων δεδομένων. Είναι δημιουργία και ιδιοκτησία της εταιρείας Oracle Corporation.

Το Oracle αποτελείται από ένα σύστημα κατανομημένων διεργασιών. Υποστηρίζεται από διάφορων τύπων servers, όπως π.χ. ο Apache. Το σύστημα διαχείρισης της βάσης μπορεί να φορτωθεί σε υπολογιστές που χρησιμοποιούν λειτουργικά συστήματα Linux, Windows, Solaris, κ.ά.



Η Oracle SQL (Structured Query Language) είναι μια γλώσσα υπολογιστή που έχει σχεδιαστεί για να διαχειριστεί τα δεδομένα που αποθηκεύονται σε βάσεις δεδομένων RDBMS (σχεσιακά συστήματα διαχείρισης βάσεων δεδομένων). Το RDBMS είναι ένα σύστημα πινάκων που αποθηκεύει δεδομένα και αντιπροσωπεύει τις σχέσεις μεταξύ τους. Η SQL έχει διάφορα δομικά στοιχεία, τα clauses, τα expressions, τα queries, τα statements και τα predicates. Η Oracle SQL χρησιμοποιείται συνήθως για την διαχείριση των δεδομένων που αποθηκεύονται σε βάσεις δεδομένων Oracle (ονομάζονται επίσης Oracle RDBMS ή απλά Oracle).

Πλεονεκτήματα :

- Μικρή εφαρμογή, ελαφριά και εύκολη στη διαχείριση.
- Οικεία σε πολλούς προγραμματιστές.
- Τώρα παρέχεται παγκόσμια κλάση υποστήριξη.
- Θα ενισχυθεί από την κοινότητα της Oracle που συνεργάζεται με την κοινότητα MySQL.
- Μπορεί να αναπτυχθεί μαζί με Oracle Database σε μια επιχείρηση εξυπηρετώντας διαφορετικές εφαρμογές και λειτουργίες των επιχειρήσεων.
- Συγκεντρωτικό σύστημα διαχείρισης και ελέγχου: Οι Oracle SQL δηλώσεις επιτρέπουν στα δεδομένα να ελέγχονται από ένα κεντρικό αποθετήριο πίνακα. Ένας διαχειριστής της βάσης δεδομένων (DBA) είναι υπεύθυνος για τη δημιουργία των χρηστών, να εκχωρεί δικαιώματα, να προσθέτει αρχεία, να διαγράφει περιττές πληροφορίες, να τροποποιεί τα υπάρχοντα δεδομένα και ερωτήματα. Αυτά τα κεντρικά αποθηκευμένα δεδομένα είναι κοινά και προσβάσιμα από διάφορες εφαρμογές. Αυτό εξαλείφει την εισαγωγή δεδομένων και πλεονασμό δεδομένων αποθήκευσης.
- Τυποποίηση: Ένα σημαντικό πλεονέκτημα της Oracle SQL είναι η τυποποίηση και η συνέπεια του στις διάφορες εφαρμογές. Η SQL για πρώτη φορά τυποποιήθηκε από το ANSI (American Standards Institute) το 1986, και περαιτέρω επικυρώθηκε το 1987 από τον Διεθνή Οργανισμό Τυποποίησης (ISO), ο οποίος παραμένει ο οργανισμός τυποποίησης της.

Μειονεκτήματα :

- Αδυναμία εφαρμογής αναδρομικής επεξεργασίας : σύμφωνα με το «SQL for Dummies» ένα από τα σημαντικότερα μειονεκτήματα της SQL είναι η αδυναμία της να εκτελέσει αναδρομική επεξεργασία. Η αναδρομική επεξεργασία είναι ένα είδος λειτουργίας υπολογιστή (ή προγράμματος), στην οποία ένα από τα βήματα του προγράμματος (ή διαδικασία). Η SQL στερείται looping δομές που είναι κοινές σε άλλες υψηλού επιπέδου γλώσσες προγραμματισμού, όπως τους «for» και «what» βρόχους. Δεν μπορεί να επαναλάβει τις δράσεις και δεν υπάρχει τρόπος να ορίσετε επαναλαμβανόμενες looping κατασκευές στην SQL.
- Ασυμβατότητα και Πολυπλοκότητα : Ένα από τα σημαντικότερα μειονεκτήματα της Oracle SQL είναι η ασυνέπεια και η ασυμβατότητα των δεδομένων στους τομείς του χρόνου και της γλώσσας είναι πολύπλοκη, με μια προσέγγιση του στυλ «λέξη - κλειδί» με δομή παρόμοια με της COBOL (common business – oriented language), με λιγότερους συντακτικούς και τους γραμματικούς κανόνες.
- Περιορισμένη λειτουργικότητα : Η SQL είναι τύπου domain ή γλώσσα ειδικού σκοπού, και η χρήση του περιορίζεται σε ένα συγκεκριμένο τομέα του προγράμματος. Οι δηλώσεις SQL λειτουργούν σε πίνακες και σύνολα δεδομένων, προσωπικές βάσεις δεδομένων και λογιστικά φύλλα. Η SQL είναι μια δηλωτική συγκεκριμένη γλώσσα τομέα που περιορίζεται στην πινακοποίηση δεδομένων.

3.2.3 PostgreSQL

Η PostgreSQL αποτελεί μια ανοιχτού κώδικα σχεσιακή βάση δεδομένων με πολλές δυνατότητες. Η ανάπτυξη της ήδη διαρκεί πάνω από 20 χρόνια και βασίζεται σε μια αποδεδειγμένα καλή



αρχιτεκτονική η οποία έχει δημιουργήσει μια ισχυρή αντίληψη των χρηστών της γύρω από την αξιοπιστία, την ακεραιότητα δεδομένων και την ορθή λειτουργία. Η PostgreSQL τρέχει σε όλα τα βασικά λειτουργικά συστήματα, περιλαμβάνοντας Linux, UNIX (AIX, BSD HP – UX, SGI IRIX, Mac OS X, Solaris, Tru64), και Windows. Είναι ACID συμβατή (ACID compliant), έχει ολοκληρωμένη υποστήριξη για foreign keys, joins, views, triggers, και stored procedures (σε διάφορες γλώσσες προγραμματισμού). Συμπεριλαμβάνει τα περισσότερα SQL92 και SQL99 data types, συμπεριλαμβανομένων INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, και TIMESTAMP επίσης υποστηρίζει αποθήκευση binary large objects, όπως εικόνες, ήχοι ή video. Διαθέτει native programming interfaces για C/C++, Java, Perl, Python, Ruby, Tel, ODBC, κ.α καθώς και τεκμηρίωση.

Η PostgreSQL υλοποιεί εξεζητημένα χαρακτηριστικά όπως Multi – Version Concurrency Control (MVCC), point in time recovery, tablespaces, asynchronous replication, nested transactions (savepoints), online/hot backups, a sophisticated query planner/optimizer, write ahead logging for fault tolerance. Υποστηρίζει διεθνή σετ χαρακτήρων, κωδικοποίηση χαρακτήρων σε πολλά byte, Unicode καθώς και δυνατότητα ταξινόμησης δεδομένων ανεξάρτητα από το locale. Η PostgreSQL μπορεί να διαχειριστεί εύκολα μεγάλους αριθμούς ταυτόχρονων χρηστών καθώς και μεγάλο όγκο δεδομένων. Υπάρχουν ενεργές εγκαταστάσεις σε περιβάλλοντα παραγωγής που διαχειρίζονται πάνω από 4 terabytes δεδομένων.

Η PostgreSQL απολαμβάνει αναγνώριση από τους χρήστες της και την βιομηχανία πληροφορικής, συμπεριλαμβανομένων των Linux New Media Award for Best Database System, και έχει υπάρξει 3 φορές νικήτρια στο Linux Journal Editors' Choice Award for best DBMS.

Γενικά χαρακτηριστικά και συμβατότητα με προδιαγραφές

Η PostgreSQL είναι συνεπής με τις προδιαγραφές. Η υλοποίηση της είναι απολύτως σύμφωνη με τις προδιαγραφές ANSI – SQL 92/99. Έχει ολοκληρωμένη υποστήριξη για subqueries (συμπεριλαμβανομένων subqueries μέσα από το FROM), read – committed και serializable transaction isolation level. Η PostgreSQL αποτελεί ένα πλήρες σχεσιακό σύστημα που υποστηρίζει πολλαπλά σχήματα ανά database, ο κατάλογος (πληροφορίες σχετικά με τους πίνακες, στήλες, views, ...) είναι διαθέσιμος διαμέσου του Information Schema όπως ορίζεται στο SQL standard.

Στα Data integrity χαρακτηριστικά συμπεριλαμβάνονται: primary keys, foreign keys με υποστήριξη restricting και cascading updates/deletes, check constraints, unique constraints, και not null constraints.

Η PostgreSQL έχει αρκετά προηγμένα χαρακτηριστικά όπως: auto – increment μέσω columns sequences, LIMIT/OFFSET που επιτρέπουν την επιστροφή partial result sets. Όσον αφορά τα indexes υποστηρίζει compound, unique, partial, και functional indexes τα οποία μπορούν να χρησιμοποιήσουν οποιονδήποτε από τους B-tree, R-tree, hash, ή GiST αλγόριθμους.

Το GiST (Generalized Search Tree) indexing αποτελεί ένα προηγμένο σύστημα το οποίο συνδυάζει ένα μεγάλο εύρος από διαφορετικούς αλγόριθμους ταξινόμησης και αναζήτησης όπως B-tree, R-tree, partial sum trees, ranked B+-trees και αρκετούς ακόμα. Επίσης διαθέτει interface το οποίο επιτρέπει τόσο την δημιουργία custom data types όσο και επεκτάσιμους τρόπους ερωτήσεων (query) για την αναζήτηση τους. Το GiST προσφέρει την ευελιξία στον προσδιορισμό του και με πιο τρόπο να αποθηκεύσεις, και την δυνατότητα να ορίσεις νέους τρόπους αναζήτησης.

Πάνω στο GiST και την PostgreSQL έχουν θεμελιωθεί πολλά projects όπως τα OpenFTS και PostGIS. Το OpenFTS (Open Source Full Text Search Engine) παρέχει online indexing και relevance ranking για αναζήτηση. Το PostGIS αποτελεί ένα project το οποίο προσθέτει υποστήριξη για geographic objects στην PostgreSQL, επιτρέποντας την χρήση της σαν spatial database για geographic information systems (GIS), περίπου όπως τα ESRI's SDE ή τα Oracle's Spatial extension.

Άλλα προηγμένα χαρακτηριστικά της PostgreSQL είναι: table inheritance, rules systems και database events. Το table inheritance (κληρονομικότητα πινάκων) πρόσθετη μια αντικειμενοστραφής διάσταση στην δημιουργία πινάκων, επιτρέποντας στους σχεδιαστές database να δημιουργούν νέους πίνακες από άλλους χρησιμοποιώντας τους σαν βάση. Ακόμα καλύτερα η PostgreSQL υποστηρίζει και μόνη και πολλαπλή κληρονομικότητα με το δικό της τρόπο.



Το rules systems, επίσης καλείται the query rewrite system, επιτρέπει στον σχεδιαστή βάσεων να δημιουργήσει κανόνες που ορίζουν συγκεκριμένες λειτουργίες για έναν πίνακα ή view, και να μετατρέπει δυναμικά την ώρα που εκτελούνται λειτουργίες σε άλλες εναλλακτικές.

Το events system αποτελεί ένα interprocess communication system στο οποίο μηνύματα και events μπορούν να μεταδοθούν μεταξύ πελατών (clients) χρησιμοποιώντας τις LISTEN και NOTIFY εντολές, επιτρέποντας από την απλή peer to peer επικοινωνία ως ένα εξελιγμένο συντονισμό βασισμένο σε database events. Εφόσον τα Notifications μπορεί να προέρχονται από triggers και stored procedures, PostgreSQL clients μπορούν να επιβλέπουν λειτουργίες όπως: updates, inserts ή deletes πινάκων όταν αυτά γίνονται.

Η PostgreSQL τρέχει stores procedures σε πολλές γλώσσες προγραμματισμού συμπεριλαμβανομένων Java, Perl, Python, Ruby, Tcl, C/C++, και της PL/pgSQL η οποία είναι παρόμοια με την PL/pgSQL της Oracle. Στην βασική βιβλιοθήκη συναρτήσεων της PostgreSQL συμπεριλαμβάνονται εκατοντάδες built-in συναρτήσεις οι οποίες καλύπτουν από βασικές μαθηματικές συναρτήσεις και διαχείριση συμβολοσειρών ως κρυπτογραφία και Oracle compatibility. Triggers και stores procedures μπορούν να γράφουν σε C και να φορτωθούν μέσα στην database ως βιβλιοθήκη, επιτρέποντας μεγάλη ευελιξία στην επέκταση των δυνατοτήτων της βάσης. Παρομοίως η PostgreSQL περιλαμβάνει framework που επιτρέπει τον ορισμό και την δημιουργία custom data types καθώς και βοηθητικές συναρτήσεις και τελεστές (operators) που θα περιγράφουν την λειτουργία τους. Σαν αποτέλεσμα ένα πλήθος από εξελιγμένα data types έχουν δημιουργηθεί από γεωμετρικά και spatial δεδομένα ως διεύθυνσης δικτύων και ISBN/ISSN (International Standard Book Number/ International Standard Serial Number) data types, τα οποία μπορούν κατ' επιλογή να προστεθούν στο σύστημα.

Η PostgreSQL όπως διαθέτει πολλές procedure languages έτσι διαθέτει και πολλά library interfaces, επιτρέποντας πολλές γλώσσες προγραμματισμού είτε compiled είτε interpreted να επικοινωνούν με την PostgreSQL. Υπάρχουν interfaces για Java (JDBC), ODBC, Perl, Python, Ruby, C, C++, PHP, Lisp, Scheme, Qt, κα.

Τέλος ο πηγαίος κώδικας της PostgreSQL είναι διαθέσιμος κάτω από την πιο ελεύθερη open source άδεια: το BSD license. Αυτή η άδεια δίνει την δυνατότητα χρήσης, μετατροπής και διανομής της PostgreSQL σε οποιαδήποτε μορφή, ανοιχτού ή κλειστού κώδικα. Η PostgreSQL δεν είναι μόνο μια δυνατή βάση δεδομένων ικανή να τρέχει μέσα σε επιχειρήσεις, είναι μια πλατφόρμα ανάπτυξης πάνω στην οποία δίνεται να γίνει ανάπτυξη in-house, web η εμπορικών εφαρμογών τα οποία χρειάζονται RDBMS.

Πλεονεκτήματα :

Η PostgreSQL προσφέρει πολλά πλεονεκτήματα για την εταιρεία ή την επιχείρησή σας σε σχέση με άλλα συστήματα βάσεων δεδομένων :

- **Ανοσία σε υπέρ-ανάπτυξη:** Η υπέρ-ανάπτυξη είναι αυτό που μερικοί προμηθευτές ιδιόκτητου λογισμικού βάσεων δεδομένων αναφέρουν ως το #1 πρόβλημα συμμόρφωσης της άδειας. Με την Η PostgreSQL, κανείς δεν μπορεί να σας μηνύσει για το σπάσιμο των συμφωνιών αδειών εκμετάλλευσης, καθώς δεν υπάρχει σχετικό κόστος αδειοδότησης για το λογισμικό. Αυτό έχει πολλά πρόσθετα πλεονεκτήματα:
 - 1) Περισσότερα κερδοφόρα επιχειρηματικά μοντέλα με ευρείας κλίμακας ανάπτυξη.
 - 2) Δεν υπάρχει δυνατότητα να ελεγχθούν ως προς τη συμμόρφωση άδειας σε οποιοδήποτε στάδιο.
 - 3) Η ευελιξία κάνει έρευνα και υλοποιήσεις χωρίς να χρειάζεται να συμπεριλάβει πρόσθετες δαπάνες αδειοδότησης
- **Καλύτερη υποστήριξη από τους ιδιόκτητους πωλητές:** Εκτός από τις ισχυρές προσφορές υποστήριξης μας, έχουμε μια ζωντανή κοινότητα από επαγγελματίες της PostgreSQL και οπαδούς που το προσωπικό μπορεί να βασιστεί και να συμβάλει.
- **Σημαντική εξοικονόμηση στο κόστος προσωπικού:** το λογισμικό έχει σχεδιαστεί και δημιουργηθεί για να έχει πολύ χαμηλότερες απαιτήσεις συντήρησης και ρύθμισης, αλλά εξακολουθεί να διατηρεί όλα τα χαρακτηριστικά, τη σταθερότητα και τις επιδόσεις. Εκτός από αυτό, τα προγράμματα κατάρτισης μας είναι γενικά γνωστό ως πολύ πιο αποδοτικό



οικονομικά, εύχρηστο, πρακτικό και στον πραγματικό κόσμο από τους κορυφαίους προμηθευτές ιδιόκτητου λογισμικού βάσεων δεδομένων.

- Αξιοπιστία και σταθερότητα : σε αντίθεση με πολλές αποκλειστικές βάσεις δεδομένων, είναι εξαιρετικά κοινό για τις εταιρίες να αναφέρουν ότι η PostgreSQL δεν έχει ποτέ δημιουργήσει προβλήματα, ούτε καν μια φορά. Απλά λειτουργεί.
- Επεκτάσιμο: Ο πηγαίος κώδικας είναι διαθέσιμος σε όλους χωρίς καμία επιβάρυνση. Εάν το προσωπικό σας έχει την ανάγκη να προσαρμόσει ή να επεκτείνει την PostgreSQL με οποιονδήποτε τρόπο, τότε είναι σε θέση να το πράξει με ελάχιστη προσπάθεια, και χωρίς να επισυνάπτεται το κόστος. Αυτό συμπληρώνεται από την κοινότητα των επαγγελματιών PostgreSQL και οπαδών σε όλο τον κόσμο που χρησιμοποιούν ενεργά την PostgreSQL σε καθημερινή βάση.
- Cross πλατφόρμα: Η PostgreSQL είναι διαθέσιμη για σχεδόν κάθε είδος Unix (34 πλατφόρμες με την τελευταία σταθερή έκδοση), και για τα windows είναι διαθέσιμη μέσω του πλαισίου Cygwin. Για τα windows είναι επίσης διαθέσιμη με την έκδοση 8.0 και άνω.
- Σχεδιασμένος για περιβάλλοντα υψηλού όγκου : Χρησιμοποιεί μια πολλαπλή στρατηγική για την αποθήκευση δεδομένων, η οποία ονομάζεται MVCC για να κάνει την PostgreSQL πολύ καλή σε απόκριση σε περιβάλλοντα υψηλού όγκου. Ο κορυφαίους προμηθευτής ιδιόκτητου λογισμικού βάσεων δεδομένων χρησιμοποιεί αυτή την τεχνολογία για τους ίδιους λόγους.
- Η GUI σχεδίαση βάσης δεδομένων και εργαλεία διαχείρισης: Υπάρχουν πολλά υψηλής ποιότητας εργαλεία GUI για την PostgreSQL από τους προγραμματιστές ανοιχτού κώδικα και τους φορείς του εμπορίου.

Μειονεκτήματα :

- Δεν υπάρχουν προεπιλεγμένες παράμετροι στο plpgsql
- Δεν υπάρχει καμία υποστήριξη ενσωματωμένη στην plpgsql για τη δημιουργία web-based διαδικασιών(όπως στην Oracle)
- Προβλήματα συμβατότητας με τα windows

3.3 Επιλογή τεχνολογίας

Με βάση τα παρακάτω καταλήξαμε στο συμπέρασμα ότι οι ιδανικότερες τεχνολογίες για την υλοποίηση της πτυχιακής είναι η HTML, η PHP, η JavaScript και η σχεσιακή βάση MySQL. Το σημαντικότερο πλεονέκτημα των παρακάτω είναι η ευκολότερη κατανόηση και ανάπτυξη κώδικα με την χρήση του καθώς και η μεγάλη κοινότητα υποστήριξης τους που υπάρχει στο διαδίκτυο με την οποία μπορούμε να αντλήσουμε πληροφορίες για την επίλυση των προβλημάτων.

Ειδικότερα η HTML μας δίνει την δυνατότητα καλύτερης δημιουργίας και προβολής των περιεχομένων της σελίδας μας με την βάση δεδομένων και την διαχείριση των περιεχομένων αυτής. Η JavaScript μας έδωσε την δυνατότητα να υλοποιήσουμε πρόσθετα χαρακτηριστικά όπως μηνύματα στον χρήστη μέσω Message Box και την ενσωμάτωση συμπληρωματικού περιεχομένου στην σελίδα μας.

Η MySQL επιλέχτηκε διότι είναι ιδιαίτερα γρήγορη στην ανάκτηση δεδομένων και είναι ανοιχτού κώδικα. Επίσης διαθέτει μια σειρά εύχρηστων εργαλείων για την ευκολότερη διαχείριση της.



ΚΕΦΑΛΑΙΟ 4 : Προγράμματα που θα χρειαστώ για την κατασκευή της ιστοσελίδα μου

4.1 Photoshop



Το **Adobe Photoshop**, ή απλά **Photoshop**, είναι ένα πρόγραμμα επεξεργασίας γραφικών που αναπτύχθηκε και κυκλοφόρησε από την Adobe Systems. Αυτή τη στιγμή αποτελεί ηγέτη της αγοράς (*market leader*) των προγραμμάτων επεξεργασίας εικόνων, και είναι το προϊόν - σήμα κατατεθέν της Adobe Systems. Χαρακτηρίζεται ως "απαραίτητο εργαλείο για τους επαγγελματίες γραφίστες" και θεωρείται πως προώθησε τις αγορές των Macintosh, και στη συνέχεια των Windows.

Η 12η, και πιο πρόσφατη έκδοση του Adobe Photoshop, είναι η Adobe Photoshop CS5, που κυκλοφόρησε τον Απρίλιο του 2010.

Το Photoshop είναι διαθέσιμο στις εξής γλώσσες: Πορτογαλικά Βραζιλίας, Απλοποιημένα Κινέζικα, Παραδοσιακά Κινέζικα, Τσέχικα, Δανέζικα, Ολλανδικά, Αγγλικά, Φινλανδικά, Γαλλικά, Γερμανικά, Ουγγρικά, Ιταλικά, Ιαπωνικά, Κορεατικά, Νορβηγικά, Πολωνικά, Ρουμάνικα, Ρωσικά, Ισπανικά, Σουηδικά, Τουρκικά και Ουκρανικά.

Τα αραβικά, ελληνικά και εβραϊκά εκδόσεις είναι διαθέσιμες από WinSoft .

Ιστορία

Το 1987 ο Τόμας Κνολ, ένας φοιτητής του Πανεπιστημίου του Μίσιγκαν, ανέπτυξε ένα πρόγραμμα που εμφάνιζε εικόνες σε αποχρώσεις του γκριζου (grayscale) σε μονοχρωματικό περιβάλλον. Αυτό το πρόγραμμα, το οποίο ονόμασε Display, τράβηξε την προσοχή του αδερφού του Τζον Κνολ, ο οποίος πρότεινε στον Τόμας να αναπτύξει ένα πλήρες πρόγραμμα επεξεργασίας εικόνας. Ο Τόμας έκανε διάλειμμα έξι μηνών από τις σπουδές του το 1988 και, σε συνεργασία με τον αδερφό του, ανέπτυξε το πρόγραμμα, το οποίο ονόμασαν ImagePro. Αργότερα το ίδιο έτος, ο Τόμας μετονόμασε το πρόγραμμα του σε Photoshop και έπειτα από συμφωνία με την κατασκευάστρια εταιρία σαρωτών Barneyscan, το πρόγραμμα διανεμήθηκε μαζί με μερικούς σαρωτές. Συνολικά διανεμήθηκαν 200 αντίγραφα του προγράμματος.

Εν τω μεταξύ, ο Τζον ταξίδεψε στο Σίλικον Βάλει και παρουσίασε το πρόγραμμα του στους μηχανικούς της Apple και στην Adobe. Και οι δύο παρουσιάσεις ήταν επιτυχείς, καθώς η Adobe αποφάσισε να αγοράσει την άδεια να διανείμει το πρόγραμμα τον Σεπτέμβριο του 1988. Η επόμενη έκδοση του προγράμματος, η Photoshop 1.0, κυκλοφόρησε το 1990 αποκλειστικά για συστήματα Macintosh και είχε μέγεθος 1.44 MB

Χαρακτηριστικά

Το Photoshop έχει δεσμούς με άλλο λογισμικό της Adobe για την επεξεργασία των μέσων ενημέρωσης, animation, και συγγραφής.

Το PSD (Photoshop Document), (η μητρική μορφή του Photoshop), αποθηκεύει μια εικόνα με υποστήριξη για τις περισσότερες επιλογές απεικόνισης που είναι διαθέσιμες στο Photoshop. Αυτές περιλαμβάνουν στρώματα με μάσκες, χώρους χρώμα, προφίλ ICC, τη διαφάνεια, το κείμενο, τα κανάλια άλφα και πλακάτων χρωμάτων, διαδρομές αποκοπής, και duotone ρυθμίσεις. Αυτό έρχεται σε αντίθεση με πολλές άλλες μορφές αρχείων (π.χ.. EPS ή GIF), που περιορίζουν το περιεχόμενο να παρέχει βελτιωμένη, προβλέψιμη λειτουργικότητα. Το PSD format περιορίζεται σε ένα μέγιστο ύψος και πλάτος των 30.000 pixels. Η PSB (Photoshop Big) μορφή, γνωστή και ως "μεγάλη μορφή εγγράφου" στο Photoshop, είναι η επέκταση της μορφή PSD για τις εικόνες έως 300.000 pixel σε πλάτος ή το ύψος. Το όριο επιλέχθηκε από την Adobe, και δεν βασίζεται σε περιορισμούς αριθμητικής υπολογιστών αλλά για ευκολία στη δοκιμή του λογισμικού. Η PSD και οι μορφές PSB είναι τεκμηριωμένες.



Η δημοτικότητα στο Photoshop σημαίνει ότι το PSD format χρησιμοποιείται ευρέως, και υποστηρίζεται σε κάποιο βαθμό από τα περισσότερα λογισμικά ανταγωνιστικά. Η PSD μορφή αρχείου μπορεί να εξαχθεί από και προς το Adobe Illustrator, Adobe Premiere Pro και After Effects, να κάνει το επαγγελματικό πρότυπο DVD και παρέχει μη γραμμικού μοντάζ και ειδικά εφέ υπηρεσίες, όπως υπόβαθρα, υφές, και ούτω καθεξής, για την τηλεόραση, τον κινηματογράφο, και το Web. Πρωτογενής δύναμη στο Photoshop είναι ένα pixel με βάση το πρόγραμμα επεξεργασίας εικόνας, σε αντίθεση με τα προγράμματα όπως το FreeHand Macromedia (δεν υπάρχει πλέον), Adobe Illustrator, Inkscape ή CorelDraw, το οποίο είναι διάνυσμα με βάση συντάκτες εικόνας. Ωστόσο, το Photoshop επιτρέπει επίσης τη δημιουργία, ενσωμάτωση, και τις πράξεις χειραγώγησης της, με διανυσματικά γραφικά μέσα από τα μονοπάτια του, με εργαλεία πένας, εργαλεία σχήματος, επίπεδα σχήματος, εργαλεία τύπου, εντολή Εισαγωγή και έξυπνες λειτουργίες αντικειμένου.

Το Photoshop χρησιμοποιεί χρώμα με μοντέλα RGB, Lab, CMYK, σε κλίμακα του γκρι, bitmap δυαδικό, και duotone. Το Photoshop έχει τη δυνατότητα να διαβάζει και να γράφει raster και vector αρχεία εικόνας όπως EPS, PNG, GIF, και JPEG.

CS3

Έξυπνα αντικείμενα οθόνης, φίλτρα που δεν μεταβάλουν την αρχική εικόνα (εδώ σε Mac OS X).

Τα νέα χαρακτηριστικά περιλαμβάνουν ένα βελτιωμένο περιβάλλον εργασίας, βελτιωμένη μηχανή υποστήριξη RAW, καλύτερο έλεγχο των επιλογών εκτύπωσης, ενισχυμένη υποστήριξη PDF, και καλύτερη διαχείριση με το Adobe Bridge. Στην επεξεργασία έχουμε νέα εργαλεία την παλέτα προέλευσης αντιγραφής και μη καταστροφικά έξυπνα φίλτρα, και άλλα χαρακτηριστικά, όπως η ρύθμιση της φωτεινότητας και της αντίθεσης και η Vanishing Point ενότητα ενισχύθηκαν. Η επιλογή Black and White βελτιώνει τον έλεγχο εγχειρίδιου σε μετατροπές αποχρώσεων του γκρι με ένα παράθυρο διαλόγου παρόμοιο με αυτό του Μείκτη καναλιών.

Το CS3 Extended περιλαμβάνει όλα τα χαρακτηριστικά του CS3 συν εργαλεία για την επεξεργασία και την εισαγωγή κάποιων 3D μορφών αρχείων γραφικών, την ενίσχυση της εικόνας, και ολοκληρωμένα εργαλεία ανάλυσης εικόνας, χρησιμοποιώντας MATLAB ολοκλήρωση και υποστήριξη αρχείων DICOM.

CS4

Το Photoshop CS4 διαθέτει νέα 3D μηχανή που επιτρέπει τη ζωγραφική απευθείας στα 3D μοντέλα, σε συσκευασία 2D εικόνων γύρω από 3D σχήματα, τη μετατροπή των χαρτών κλίσης σε 3D αντικείμενα, προσθέτοντας το βάθος στα στρώματα και το κείμενο. Στις 30 Απριλίου, κυκλοφόρησε το Adobe Photoshop CS4 Extended, το οποίο περιλαμβάνει όλα τα χαρακτηριστικά του Adobe Photoshop CS4 με την προσθήκη των δυνατοτήτων για την επιστημονική απεικόνιση, 3D, και high-end φιλμ και βίντεο χρηστών. Ο διάδοχος του Photoshop CS3, (το Photoshop CS4) είναι το πρώτο 64-bit Photoshop στους υπολογιστές των καταναλωτών (μόνο στα Windows. - η OS X έκδοση είναι 32-bit ακόμα).

CS5

Το Photoshop CS5 ξεκίνησε στις 12 Απριλίου 2010. Σε ένα βίντεο που αναρτήθηκε στην επίσημη σελίδα του Facebook, η ομάδα ανάπτυξης αποκάλυψε τις νέες τεχνολογίες υπό ανάπτυξη, συμπεριλαμβανομένων τριών διαστάσεων πινέλα και τα εργαλεία στρέβλωσης.

Τον Μάιο του 2011, στο Adobe Creative Suites 5.5 (CS5.5) αφέθηκαν ελεύθερες οι νέες εκδόσεις του σε κάποιες από τις εφαρμογές.



4.2 Apache HTTP εξυπηρετητής



Ο Apache HTTP γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού (web). Οποτε ένας χρήστης επισκέπτεται ένα ιστότοπο το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου HTTP, ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Ο Apache είναι ένας από τους δημοφιλέστερους, εν μέρει γιατί λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το Unix και το Mac OS X. Συντηρείται τώρα από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation). Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. Oracle, MySQL.

Το ASF κατέχει επίσης διάφορα συνέδρια ApacheCon κάθε χρόνο, αναδεικνύοντας Apache έργα, που αφορούν την τεχνολογία, και την ενθάρρυνση Apache προγραμματιστών να συγκεντρωθούν.

Ιστορία

Η πρώτη του έκδοση, γνωστή ως NCSA HTTPd, δημιουργήθηκε από τον Robert McCool και κυκλοφόρησε το 1993. Θεωρείται ότι έπαιξε σημαντικό ρόλο στην αρχική επέκταση του παγκόσμιου ιστού. Ήταν η πρώτη βιώσιμη εναλλακτική επιλογή που παρουσιάστηκε απέναντι στον εξυπηρετητή http της εταιρείας Netscape και από τότε έχει εξελιχθεί στο σημείο να ανταγωνίζεται άλλους εξυπηρετητές βασισμένους στο Unix σε λειτουργικότητα και απόδοση. Από το 1996 ήταν από τους πιο δημοφιλείς όμως από τον Μάρτιο του 2006 έχει μειωθεί το ποσοστό της εγκατάστασής του κυρίως από τον Microsoft Internet Information Services και την πλατφόρμα .NET . Τον Οκτώβριο του 2007 το μερίδιο του ήταν 47.73% από όλους τους ιστοτόπους.

Έργα

Η Apache διαιρεί το λογισμικό της ανάπτυξης των δραστηριοτήτων της σε ξεχωριστές ημιαυτόνομες περιοχές που ονομάζονται "top-level σχέδια» (επίσημως γνωστό ως "Επιτροπή Διαχείρισης του Προγράμματος" από τον Εσωτερικό Κανονισμό), μερικά από τα οποία έχουν μια σειρά επιμέρους έργων. Σε αντίθεση με ορισμένες άλλες οργανώσεις τα έργα ΕΛΛΑΚ υποδοχής, πριν από ένα έργο φιλοξενούνται σε Apache, για να εκχωρηθεί άδεια για το ASF με επιχορήγηση ή συνεισφέρουν στη συμφωνία. Με τον τρόπο αυτό, το ASF κερδίζει τα απαραίτητα δικαιώματα πνευματικής ιδιοκτησίας για την ανάπτυξη και διανομή όλων των έργων του .



4.3 Dreamweaver

Το Adobe Dreamweaver (πρώην Macromedia Dreamweaver) είναι μια ιδιόκτητη εφαρμογή στο web που αρχικά δημιουργήθηκε από την Macromedia, και τώρα αναπτύχθηκε από την Adobe Systems, η οποία απέκτησε το Macromedia το 2005.

Το Dreamweaver είναι διαθέσιμο τόσο για Mac και Windows λειτουργικά συστήματα. Οι πρόσφατες εκδόσεις έχουν ενσωματωμένη υποστήριξη για τεχνολογίες Web, όπως CSS, JavaScript, και διάφορες server-side scripting γλώσσες και πλαίσια, συμπεριλαμβανομένων των ASP, ColdFusion, και PHP.



Χαρακτηριστικά

Το Adobe Dreamweaver επιτρέπει στους χρήστες να δουν ιστοσελίδες σε ένα παράθυρο προεπισκόπησης ή σε τοπικά εγκατεστημένα προγράμματα περιήγησης στο Web.

Παρέχει σύγχρονα χαρακτηριστικά, και την δυνατότητα να βρείτε και να αντικαταστήσετε τις γραμμές του κειμένου ή του κώδικα με τους όρους αναζήτησης και κανονικές εκφράσεις σε ολόκληρο τον ιστότοπο. Ο πίνακας συμπεριφοράς επιτρέπει επίσης τη χρήση JavaScript χωρίς κωδικοποίηση γνώσης, και την ενσωμάτωση με Spry Ajax της Adobe πλαίσιο. Το Dreamweaver μπορεί να χρησιμοποιήσει "Επεκτάσεις" για την βασική επέκταση της λειτουργικότητας της εφαρμογής, το οποίο κάθε προγραμματιστής Ιστού μπορεί να γράψει (σε μεγάλο βαθμό σε HTML και JavaScript). Ο Dreamweaver υποστηρίζεται από μια μεγάλη κοινότητα προγραμματιστών που κάνουν εργασίες ανάπτυξης ιστοσελίδων.

Το Dreamweaver, όπως και οι άλλοι συντάκτες HTML, επεξεργάζονται τα αρχεία σε τοπικό επίπεδο, στη συνέχεια, τοποθετούν τις προσθήσεις τους στον απομακρυσμένο διακομιστή web χρησιμοποιώντας FTP, SFTP ή WebDAV. Το Dreamweaver CS4 υποστηρίζει τώρα το Subversion (SVN) σύστημα ελέγχου εκδόσεων.

Η έκδοση 5, του Dreamweaver υποστηρίζει τη χρωματική σήμανση για τις ακόλουθες γλώσσες από το κουτί:

- ActionScript
- Active Server Pages (ASP).
- ASP.NET (δεν υποστηρίζεται πλέον από την έκδοση CS4 - <http://kb2.adobe.com/cps/402/kb402489.html>)
- C #
- Cascading Style Sheets (CSS)
- ColdFusion
- EDML
- Extensible Hypertext Markup Language (XHTML)
- Extensible Markup Language (XML)
- Extensible Stylesheet Language Transformations (XSLT)
- HyperText Markup Language (HTML)
- Ιάβα
- JavaScript
- JavaServer Pages (JSP) (δεν υποστηρίζεται πλέον από την έκδοση CS4 - <http://kb2.adobe.com/cps/402/kb402489.html>)
- PHP: Hypertext Preprocessor (PHP)
- Visual Basic (VB)
- Visual Basic Script Edition (VBScript)
- Wireless Markup Language (WML)

Είναι επίσης δυνατό για τους χρήστες να προσθέσουν της δικές τους επισήμανσης στην σύνταξης της γλώσσας. Επιπλέον, η ολοκλήρωση του κώδικα είναι διαθέσιμη σε πολλές γλώσσες.

Διαθεσιμότητα σε Γλώσσες

Το Adobe Dreamweaver CS5 είναι διαθέσιμο στις ακόλουθες γλώσσες: Πορτογαλικά Βραζιλίας, Απλοποιημένα Κινέζικα, Παραδοσιακά Κινέζικα, Τσέχικα, Ολλανδικά, Αγγλικά, Γαλλικά, Γερμανικά, Ιταλικά, Ιαπωνικά, Κορεατικά, Πολωνικά, Ρωσικά, Ισπανικά, Σουηδικά και Τουρκικά.

Περιέχει και ειδικά χαρακτηριστικά για την αραβική και εβραϊκή γλώσσα. Η παλαιότερη Adobe Dreamweaver CS3 διαθέτει επίσης μια έκδοση της Μέσης Ανατολής που επιτρέπει την πληκτρολόγηση σε αραβικά, περσικά ή εβραϊκό κείμενο (γραμμένο από δεξιά προς αριστερά) κατά την προβολή κώδικα. Είτε το κείμενο είναι πλήρως Μέσης Ανατολής (γράφονται από τα δεξιά προς τα αριστερά) ή να περιλαμβάνει τόσο στα Αγγλικά όσο και της Μέσης Ανατολής κείμενο (γραμμένο αριστερά προς τα δεξιά και δεξιά προς τα αριστερά), το οποίο θα εμφανιστεί σωστά στο πρόγραμμα περιήγησης.



ΚΕΦΑΛΑΙΟ 5 : Σχεδιασμός Βάσεων Δεδομένων

5.1 Εισαγωγικά

5.1.1 Βασικές έννοιες

- **Τι είναι η βάση δεδομένων (ΒΔ) ή Database (DB);**

Με τον όρο **βάση δεδομένων** εννοείται μία συλλογή από *συστηματικά οργανωμένα* (formatted) σχετιζόμενα δεδομένα. Ένας τηλεφωνικός κατάλογος, για παράδειγμα, θεωρείται βάση δεδομένων, καθώς αποθηκεύει και οργανώνει σχετιζόμενα τμήματα πληροφορίας, όπως είναι το όνομα και ο αριθμός τηλεφώνου. Ωστόσο, στον κόσμο των υπολογιστών, με τον όρο βάση δεδομένων αναφερόμαστε σε μια συλλογή σχετιζόμενων δεδομένων τμημάτων πληροφορίας ηλεκτρονικά αποθηκευμένων.

- **Τι είναι το σύστημα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ) ή Database Management System (DBMS);**

Πέρα από την εγγενή της ικανότητα να αποθηκεύει δεδομένα, η βάση δεδομένων παρέχει βάσει του σχεδιασμού και του τρόπου ιεράρχησης των δεδομένων της σε προγράμματα ή συλλογές προγραμμάτων, τα αποκαλούμενα συστήματα διαχείρισης περιεχομένου, τη δυνατότητα γρήγορης άντλησης και ανανέωσης των δεδομένων. Η ηλεκτρονική βάση δεδομένων χρησιμοποιεί ιδιαίτερου τύπου λογισμικό προκειμένου να οργανώσει την αποθήκευση των δεδομένων της. Το διακριτό αυτό λογισμικό είναι γνωστό ως Σύστημα διαχείρισης βάσης δεδομένων συντομευμένα (DBMS).

- **Τι είναι το Σύστημα Βάσης δεδομένων (ΣΒΔ) ή Database System (DBS);**

ΣΒΔ καλείται ο συνδυασμός και η σύνδεση της βάσης με το λογισμικό.

- **Τι είναι ο Database Administrator (BDA) ή διαχειριστής μιας Βάσης δεδομένων (ΔΒΑ) και ποιος ο ρόλος του;**

Ο διαχειριστής μιας βάσης δεδομένων (Database Administrator) είναι αυτός που έχει την ευθύνη για τον σωστό, αποδοτικό και αξιόπιστο τρόπο δημιουργίας μιας βάσης δεδομένων. Οι αρμοδιότητες του είναι οι εξής:

- ❖ Η απόφαση για το είδος των πληροφοριών που πρέπει να αποθηκευτούν.
- ❖ Η απόφαση για τον τρόπο αποθήκευσης και πρόσβασης στις πληροφορίες αυτές.
- ❖ Η συνεργασία με τους τελικούς χρήστες.
- ❖ Η απόφαση για τον τρόπο εξασφάλισης των πληροφοριών
- ❖ Η απόφαση για το κάθε πότε θα γίνονται αντίγραφα ασφαλείας (backup) των αρχείων.
- ❖ Η παρακολούθηση της σωστής λειτουργίας της βάσης δεδομένων και η απαιτούμενη προσαρμογή της.

5.1.2 Ιστορική εξέλιξη των Βάσεων Δεδομένων:

- Δεκαετία του 1950

Κάρτες και ταινίες (σειριακή επεξεργασία): εξελίξεις σε συσκευές μαζικής αποθήκευσης τυχαίας πρόσβασης και αύξηση υπολογιστικής ισχύος θέτουν τις προϋποθέσεις για την ανάπτυξη συστημάτων διαχείρισης δεδομένων σε αντικατάσταση των συστημάτων διαχείρισης αρχείων.



- Αρχή του 1960

Πρώτες βάσεις δεδομένων, πρώτα συστήματα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ), στηριγμένα στο δικτυωτό μοντέλο – network data model – του Charles Bachman (1st Turing Award, 1973).

- Τέλη του 1960

Βάσεις δεδομένων και συστήματα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ), στηριγμένα στο ιεραρχικό μοντέλο δεδομένων – Information Management System (IMS της IBM) - 1970 Ορισμός του σχεσιακού μοντέλου από τον Codd της IBM. Ο Edgar Codd (IBM, San Jose) relational datamodel (Turing Award, 1981) Ερευνητικά προγράμματα: System R, INGRES-. Γλώσσες: SEQUEL, QBE, QUEL.

- Αρχές του 1980

SQL (μέρος του System R) αντικατέστησε την QUEL στο σύστημα INGRES. Η έννοια του Transaction management (Jim Gray, Turing Award, 1999). Τάσεις: αντικειμενοστραφή, αρχιτεκτονική πελάτη – εξυπηρετητή, καταναμημένες βάσεις δεδομένων.

- Δεκαετία του 1980

ΒΔ σε προσωπικούς υπολογιστές, σχεσιακά συστήματα σε pc: Dbase II (Ashton Tate) εξελίχθηκε μέχρι τις μέρες μας ως Paradox, Microsoft ACCESS. Σχεσιακά συστήματα με μεγάλη διάδοση: Oracle, SQL Server, Sybase, Informix, DB2.

- Δεκαετία του 1990 μέχρι σήμερα

Αντικειμενοστραφή συστήματα ΒΔ σύνδεση ΒΔ στο διαδίκτυο, ευρύτατη διάδοση τεχνολογίας που επιτρέπει την επικοινωνία των χρηστών με ΒΔ μέσω διαδικτύου (HTML, XML, ASP κλπ).

5.1.3 Αρχιτεκτονική τριών επιπέδων:

Ένα σύστημα βάσεων δεδομένων παρέχει στους χρήστες μια αφαιρετική άποψη. Το σύστημα κρύβει (από τους χρήστες) ορισμένες λεπτομέρειες για το πώς τα δεδομένα αποθηκεύονται και διατηρούνται. Γι' αυτό υπάρχουν τρία επίπεδα αφαίρεσης:

1. Φυσικό επίπεδο:

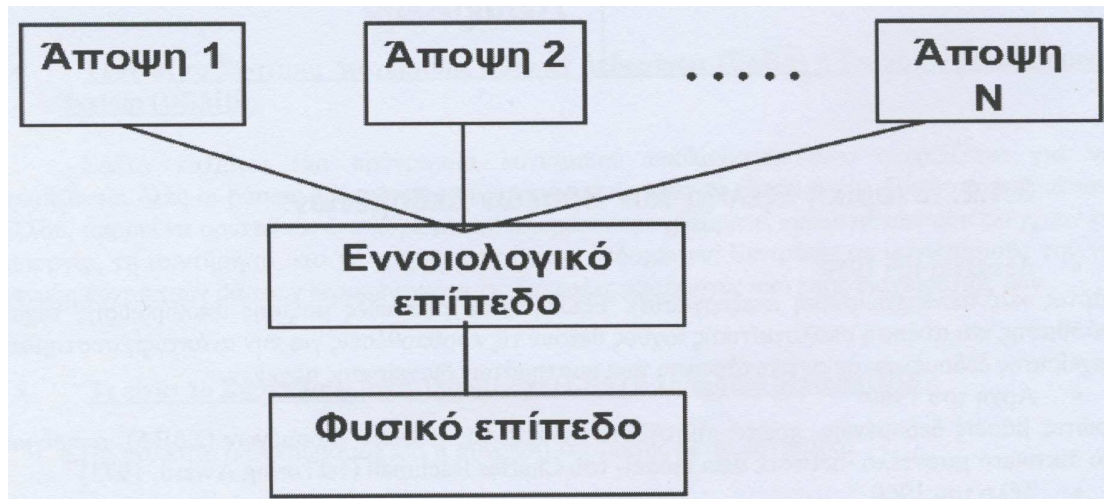
Είναι το χαμηλότερο επίπεδο αφαίρεσης και περιγράφει πως αποθηκεύονται τα δεδομένα.

2. Εννοιολογικό επίπεδο:

Το αμέσως υψηλότερο επίπεδο αφαίρεσης. Περιγράφει τι είδους δεδομένα αποθηκεύονται και τις συσχετίσεις μεταξύ αυτών των δεδομένων.

3. Επίπεδο άποψης :

Το πιο υψηλό επίπεδο, περιγράφει τμήματα της βάσης δεδομένων για κάθε συγκεκριμένη ομάδα χρηστών, μπορεί να υπάρχουν πολλές διαφορετικές απόψεις, π.χ Οι ταμίες σε μια τράπεζα παίρνουν μια άποψη των απολογισμών πελατών, αλλά όχι των δεδομένων μισθοδοτικών καταστάσεων.



Εικόνα 3: Σχηματική απεικόνιση τις αρχιτεκτονικής των τριών επιπέδων.

5.1.4 Μορφές Βάσεων Δεδομένων:

- **Οι Αντικειμενοστραφείς Βάσεις δεδομένων:**

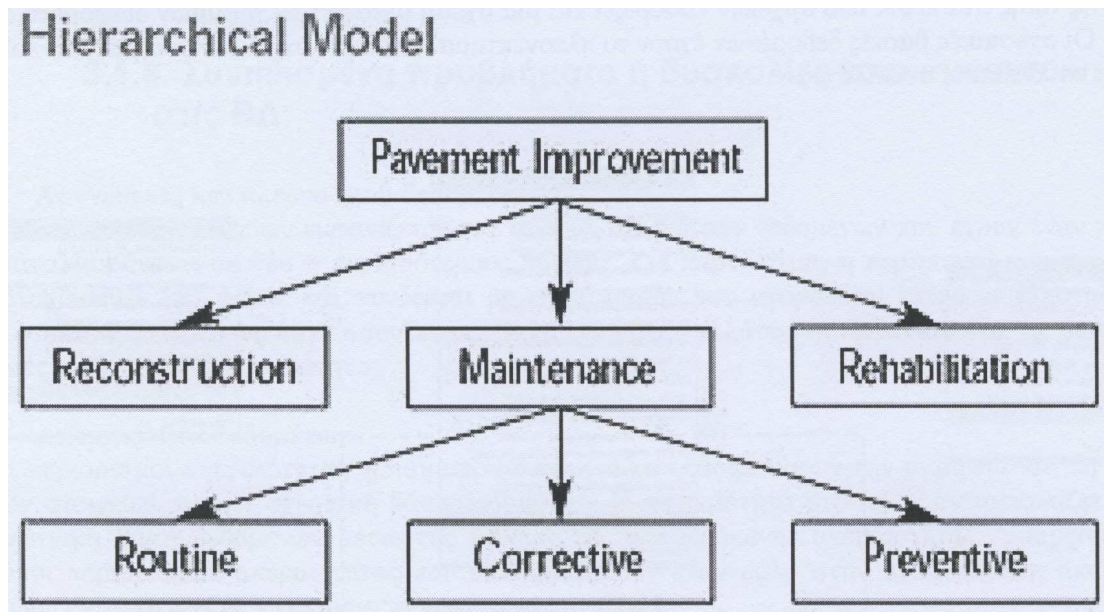
Μια βάση δεδομένων αντικειμένου (Object database) (επίσης αναφέρεται και ως αντικειμενοστραφές σύστημα διαχείρισης βάσεων δεδομένων) είναι ένα σύστημα διαχείρισης βάσεων δεδομένων στο οποίο οι πληροφορίες αναπαρίστανται με την μορφή αντικειμένων όπως χρησιμοποιούνται στον αντικειμενοστραφή προγραμματισμό. Οι βάσεις δεδομένων αντικειμένου είναι ένας τομέας μέσα στην ευρύτερη αγορά συστημάτων διαχείρισης βάσεων δεδομένων (DBMS) όπου κυριαρχούν τα μοντέλα σχεσιακής διαχείρισης βάσης δεδομένων.

- **Οι Ημι – δομημένες Βάσεις δεδομένων:**

Στο ημι – δομημένο μοντέλο (semi - structured) δεν υπάρχει κανένας διαχωρισμός μεταξύ των δεδομένων και του σχήματος, και το χρησιμοποιούμενο πόσο δομής εξαρτάται από το εκάστοτε σκοπό.

- **Οι Ιεραρχικές Βάσεις δεδομένων:**

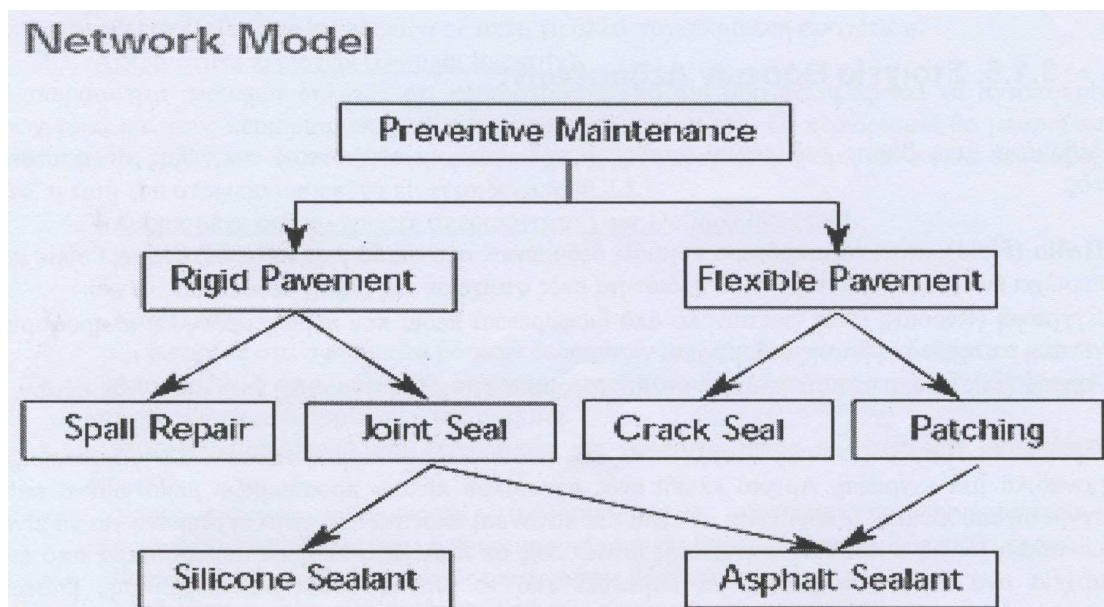
Στις ιεραρχικές (Hierarchical) βάσεις δεδομένων τα δεδομένα αναπαριστούνται με δενδρικής μορφής δομές δεδομένων και συνδέονται μεταξύ τους με συνδέσμους (links). Η κάθε εγγραφή μπορεί να συνδέεται προς τα πάνω μόνο με μια άλλη εγγραφή (γονέας), ενώ μπορεί να έχει έως δύο εγγραφές που να εξαρτώνται απ' αυτήν (παιδιά). Υπάρχει μια μόνο εγγραφή ρίζα (root), απ' την οποία εξαρτώνται στην επεξεργασία των εγγραφών τους. (προσθήκη, διαγραφή, τροποποίηση).



Εικόνα 4: Σχηματική απεικόνιση της ιεραρχικής Βάσης δεδομένων.

- **Οι Δικτυωτές Βάσεις δεδομένων:**

Στις δικτυωτές (Network) βάσεις δεδομένων τα δεδομένα αναπαριστούνται με δενδρικής μορφής δομές δεδομένων και συνδέονται μεταξύ τους με συνδέσμους (links), όπως ακριβώς και στις ιεραρχικές βάσεις δεδομένων, με την διαφορά ότι μια εγγραφή μπορεί να συνδέεται προς τα πάνω με περισσότερες από μια πατρικές εγγραφές (parent records). Είναι λογικά πιο δύσχρηστες αλλά και πιο γρήγορες από τις ιεραρχικές βάσεις δεδομένων.



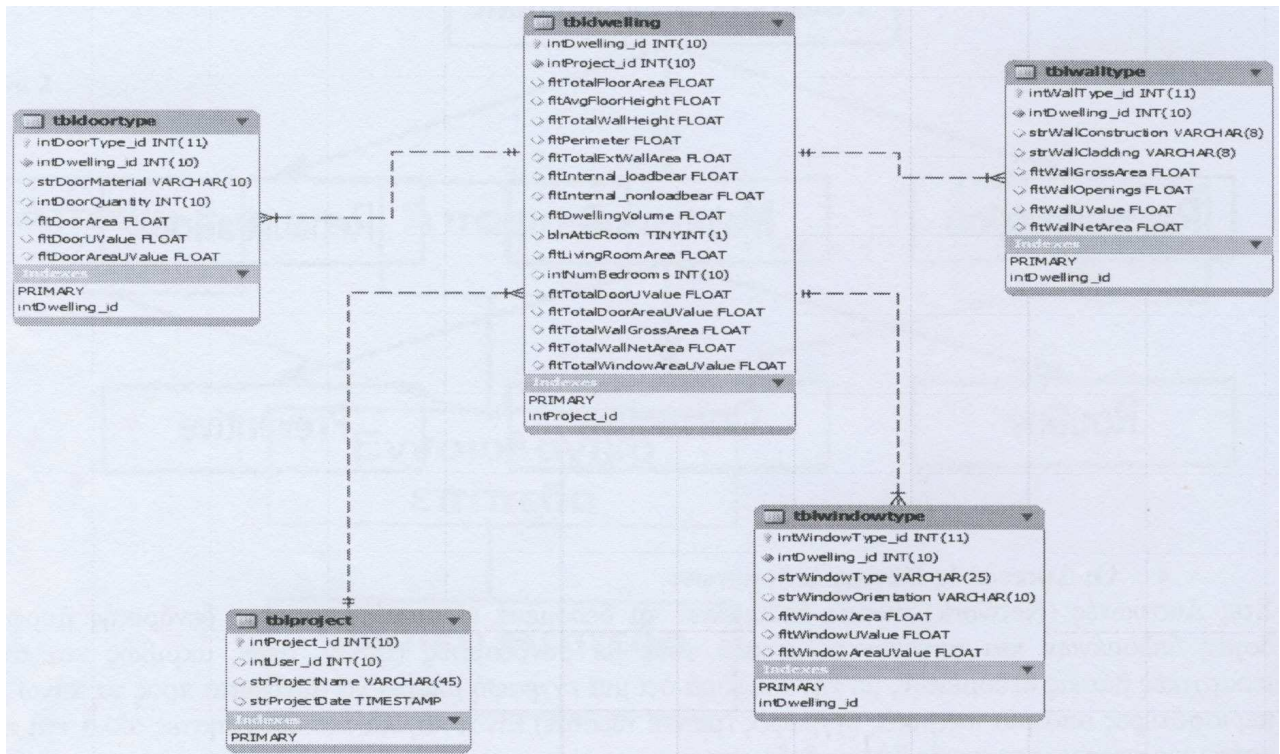
Εικόνα 5: Σχηματική απεικόνιση των δικτυωτών βάσεων δεδομένων.

- **Οι σχεσιακές βάσεις δεδομένων:**

Στις σχεσιακές (Relational) βάσεις δεδομένων, τα δεδομένα συνδέονται μεταξύ τους με σχέσεις (relations), οι οποίες προκύπτουν από τα κοινά πεδία που υπάρχουν σε διαφορετικά αρχεία. Τα αρχεία



αποκαλούνται πίνακες (tables), οι εγγραφές γραμμές (rows) και τα πεδία στήλες (columns). Η ύπαρξη μιας κοινής τιμής στα πεδία δύο αρχείων καθορίζει και μια σχέση μεταξύ των γραμμών διαφορετικών πινάκων. Οι σχεσιακές βάσεις δεδομένων έχουν το πλεονέκτημα ότι είναι λογικά κατανοητές και πολύ ευέλικτες και δεκτικές σε αλλαγές.



Εικόνα 6: Σχηματική απεικόνιση της σχεσιακής βάσης δεδομένων.

5.1.5 Στοιχεία Βάσεων Δεδομένων

Τα δεδομένα μιας βάσης δεδομένων αποθηκεύονται (και οργανώνονται) στις εξής στοιχειώδεις μορφές:

- **Πεδίο (Field)**, είναι το μικρότερο κομμάτι δεδομένων στο οποίο μπορούμε να αναφερθούμε και περιέχει ένα μόνο χαρακτηριστικό ή ιδιότητα ενός στοιχείου της βάσης δεδομένων.
- **Εγγραφή (Record)**, είναι ένα σύνολο από διαφορετικά πεδία που περιέχει όλες τις πληροφορίες για ένα στοιχείο της βάσης δεδομένων.
- **Αρχείο (File)**, είναι ένα σύνολο από πολλά παρόμοια στοιχεία (εγγραφές) της βάσης δεδομένων.
- **Πρωτεύον κλειδί (Primary Key)**, είναι ένα πεδίο ή συνδυασμός πεδίων που χαρακτηρίζει μοναδικά μια εγγραφή. Αρχικό κλειδί ενός σχεσιακού πίνακα προσδιορίζει μεμονωμένα κάθε εγγραφή του πίνακα. Μπορεί είτε να είναι μια κανονική ιδιότητα που είναι εγγυημένη για να είναι μοναδική (όπως ο αριθμός κοινωνικής ασφάλισης σε έναν πίνακα χωρίς περισσότερα από ένα αρχεία ανά άτομο) ή μπορεί να παραχθεί από το DBMS (σύστημα διαχείρισης βάσεων δεδομένων). Τα αρχικά κλειδιά μπορούν να αποτελούν τόσο απλές ιδιότητες όσο και πολλαπλές σε συνδυασμό.
- **Ξένο κλειδί (Foreign key)**, είναι ένα πεδίο που έχει το ίδιο σύνολο τιμών με το πρωτεύον κλειδί ενός άλλου αρχείου. Αναλυτικότερα ξένο κλειδί είναι μια ιδιότητα σε έναν σχεσιακό πίνακα που ταιριάζει με ένα υποψήφιο κλειδί ενός άλλου πίνακα που αναφέρεται σε μια στήλη ή ένα σύνολο στηλών σε έναν άλλο πίνακα.



5.1.6 Συνηθισμένα προβλήματα ή δυσκολίες που συναντώνται στις ΒΔ:

1) Ασυνέπειας και πλεονασμού δεδομένων

Ο πλεονασμός στοιχείων εμφανίζεται στα συστήματα βάσεων δεδομένων που έχουν ένα πεδίο που επαναλαμβάνεται σε δύο ή περισσότερους πίνακες. Για παράδειγμα, η περίπτωση που ένας πελάτης έχει καταχωρηθεί δύο φορές και συνδέεται με κάθε προϊόν που αγοράζεται, αυτό επιφέρει πλεονασμό στα δεδομένα, γνωστό ως πηγή ασυνέπειας δεδομένου ότι ο πελάτης θα εμφανιστεί με τις διαφορετικές τιμές για τις δεδομένες ιδιότητες.

2) Ακεραιότητα δεδομένων

Οι περιορισμοί ακεραιότητας χρησιμοποιούνται για να εξασφαλίσουν την ακρίβεια και την συνέπεια των στοιχείων σε μια σχεσιακή βάση δεδομένων. Η ακεραιότητα στοιχείων αντιμετωπίζεται σε μια σχεσιακή βάση δεδομένων μέσω της έννοιας της αναφερόμενης ακεραιότητας. Υπάρχουν πολλοί τύποι περιορισμών ακεραιότητας που διαδραματίζουν έναν ρόλο στην αναφερόμενη ακεραιότητα. Χωρίζονται στις εξής κατηγορίες σύμφωνα με τον Codd.

2.1. Ακεραιότητα οντοτήτων (Entity Integrity)

Ο περιορισμός ακεραιότητας οντοτήτων δηλώνει ότι καμία τιμή βασικού κλειδιού δεν μπορεί να είναι μηδενική. Αυτό είναι επειδή η τιμή βασικού κλειδιού χρησιμοποιείται για να προσδιορίσει μεμονωμένες πλειάδες σε μια σχέση. Η κατοχή της μηδενικής αξίας για το αρχικό κλειδί υπονοεί ότι δεν μπορούμε να προσδιορίσουμε σίγουρα κάποιες πλειάδες. Αυτό επίσης εξασφαλίζει ότι δεν μπορούν να υπάρξουν οποιεσδήποτε διπλές καταχωρίσεις στην αρχική βασική στήλη (column).

2.2. Αναφερόμενη ακεραιότητα (Referential Integrity)

Ο αναφερόμενος περιορισμός ακεραιότητας αφορά δύο συσχετίσεις και χρησιμοποιείται για να διατηρηθεί η συνέπεια μεταξύ πλειάδων στις δύο σχέσεις. Πιο απλά, ο αναφερόμενος περιορισμός ακεραιότητας δηλώνει ότι η πλειάδα σε μια συσχέτιση που αναφέρεται σε μια άλλη συσχέτιση πρέπει να αναφερθεί σε (reference to) πλειάδα σε αυτή την άλλη συγκεκριμένη συσχέτιση.

2.3. Ακεραιότητα περιοχών (Domain Integrity)

Η ακεραιότητα περιοχών δηλώνει ότι κάθε στοιχείο από μια συσχέτιση πρέπει να ανταποκρίνεται στον τύπο και τους περιορισμούς των αντίστοιχων ιδιοτήτων του. Οι περιορισμοί θα μπορούσαν να είναι το πεδίο τιμών που το στοιχείο μπορεί να έχει προκαθορισμένη τιμή εάν καμία δεν παρέχεται, τότε η τιμή του στοιχείου μπορεί να είναι μηδενική (NULL).

2.4. Καθορισμένη από το χρήστη ακεραιότητα (User Defined Integrity)

Π.χ.: $\text{Ηλικία} \geq 18 \ \&\& \ \text{Ηλικία} \leq 18$

3) Ασφάλειας

Οι διαρροές στα συστήματα βάσεων δεδομένων περιλαμβάνουν, παραδείγματος χάριν:

- Αναρμόδια ή απρομελέτητη δραστηριότητα ή κακή χρήση από τους εξουσιοδοτημένους χρήστες βάσεων δεδομένων ή και hacking.
- Μολύνσεις Malware που προκαλούν τα γεγονότα όπως η αρμόδια πρόσβαση και διαρροή.
- Υπερφορτώσεις, περιορισμοί απόδοσης με συνέπεια την ανικανότητα των εξουσιοδοτημένων χρηστών να χρησιμοποιήσουν τις βάσεις δεδομένων όπως θα έπρεπε.
- Φυσική ζημιά στους κεντρικούς υπολογιστές βάσεων δεδομένων.
- Σχεδιαστικά σφάλματα και bugs κατά των προγραμματισμό των ΒΔ.
- Απώλεια δεδομένων που προκαλούνται από την είσοδο των άκυρων καταχωρίσεων ή λανθασμένων εντολών, π.χ τα λάθη στις διαδικασίες διαχείρισης βάσεων δεδομένων κλπ.

4) Ανεξαρτησία δεδομένων (δηλαδή πολλαπλά αρχεία και formats)

Η ανεξαρτησία στοιχείων είναι μια μορφή διαχείρισης βάσεων δεδομένων που κρατά τα στοιχεία χωρισμένα από όλα τα προγράμματα που χρησιμοποιούν τα στοιχεία. Σαν ακρογωνιαίο λίθο για την ιδέα ενός ΣΔΒΔ (Σύστημα Διαχείρισης Βάσεων Δεδομένων) η ανεξαρτησία δεδομένων εξασφαλίζει ότι τα στοιχεία δεν μπορούν να επαναπροσδιοριστούν ή να αναδιοργανωθούν από οποιαδήποτε από τα προγράμματα τα οποία τα χρησιμοποιούν. Κατά αυτόν τον τρόπο, το στοιχείο παραμένει προσπελάσιμο, αλλά είναι επίσης σταθερό και δεν μπορεί να αλλοιωθεί από τις εφαρμογές χρησιμοποιώντας το.

5) Προσπέλαση δεδομένων



6) Ενημέρωση πολλών αρχείων

Συνεπώς στόχοι ώστε να επιτύχουμε μια ορθά σχεδιασμένη βάση δεδομένων πρέπει να είναι οι εξής:

- Ο περιορισμός της πολλαπλής αποθήκευσης των ίδιων στοιχείων (redundancy).
- Ο καταμερισμός (sharing) των ίδιων στοιχείων σ' όλους τους χρήστες.
- Η ομοιομορφία (uniformity) στον χειρισμό και την αναπαράσταση των δεδομένων.
- Η επιβολή κανόνων ασφαλείας (security).
- Η διατήρηση της ακεραιότητας (integrity) και της αξιοπιστίας (reliability) των δεδομένων.
- Η ανεξαρτησία των δεδομένων (data independence) και των προγραμμάτων από τον φυσικό τρόπο αποθήκευσης των δεδομένων.

Και τα κατά σειρά βήματα σχεδιασμού της:

Ο σχεδιασμός για την δημιουργία μιας βάσης δεδομένων περνάει από τρία βασικά στάδια.

❖ Ανάλυση των απαιτήσεων :

Δηλαδή τι τύπου δεδομένα θέλουμε να αποθηκευτούν, ποιες διεργασίες θα πρέπει να γίνονται πάνω σε αυτά ή συχνότητα λειτουργιών που χρησιμοποιούνται.

❖ Εννοιολογικός σχεδιασμός :

Ακριβής και υψηλού επιπέδου περιγραφή των δεδομένων που πρόκειται να αποθηκεύονται στη ΒΔ, ταυτόχρονη εμφάνιση των περιορισμών και σχέσεων μεταξύ δεδομένων (χρήση e-r μοντέλου).

❖ Λογικός σχεδιασμός :

Επιλογή ενός εργαλείου σχεδίασης βάσεων δεδομένων για την υλοποίηση του σχεδιασμού, μετατροπή του εννοιολογικού σχεδιασμού σε ένα σχήμα στο μοντέλο δεδομένων του επιλεγμένου ΣΔΒΔ (σ.σ ΣΔΒΔ: Σύστημα Διαχείρισης Βάσης Δεδομένων – Database Management System – είναι ένα πακέτο λογισμικού για τη συστηματική αποθήκευση - storage -, αναζήτηση - retrieval – και συντήρηση - maintenance- δεδομένων).

5.2 Ανάλυση των Απαιτήσεων

5.2.1 Περιγραφή συστήματος

Είναι ένα online πληροφοριακό σύστημα για μια εταιρία κινητής τηλεφωνίας που κατασκευάστηκε για να διαχειρίζεται όσο πιο αποδοτικά μπορεί τους υπαλλήλους του κάθε υποκαταστήματος της και να ενημερώνονται για την απόδοση τους και την ομαλή και καλή λειτουργία του αυτού καθ' αυτό υποκαταστήματος της.

Για να δημιουργηθεί η βάση δεδομένων που θα συνδέεται με την δυναμική ιστοσελίδα μου, θα χρειαστεί μια μεθοδολογία, την οποία θα δούμε στις επόμενες παραγράφους του κεφαλαίου αυτού, από την ανάλυση των απαιτήσεων της εφαρμογής μέχρι και των κώδικα μου.

5.2.2 Λειτουργικές απαιτήσεις (operational /functional specifications).

1. Αποθήκευση πληροφορίας:

1.1. Υποκατάστημα: το όνομα, της διεύθυνση

1.2. Υπάλληλο: όνομα, το επώνυμο, τον αριθμό ταυτότητας, τον μισθό, τα προστατευόμενα μέλη της οικογένειας τους (ονοματεπώνυμο, φύλο, και ηλικία).

Οι υπάλληλοι χωρίζονται σε δύο κατηγορίες

1.2.α) σε διοικητικούς για τους οποίους θέλουμε να αποθηκεύουμε πληροφορία για το πτυχίο τους (τίτλο και πανεπιστήμιο που το έδωσε και ημερομηνία κτήσης)

1.2.β) σε τεχνικούς για τους οποίους θέλουμε να αποθηκεύουμε πληροφορία για την ειδικότητα και τα έτη προϋπηρεσίας.



- 1.3. Κεραίες: όνομα, την διεύθυνση και τον τύπο της κεραίας.
- 1.4. Πελάτη: όνομα, την διεύθυνση, τον αριθμό ταυτότητας, το επάγγελμα.
- 1.5. Αριθμό τηλεφώνου: αριθμό κλήσης, τον κωδικό αριθμό (ruk1 και ruk2), αν είναι σύνδεση ή καρτοκινητό
- 1.6. Πακέτα χρέωσης: Το όνομα, τα λεπτά της δωρεάν ομιλίας και τα μηνύματα, το πάγιο, και την τιμή χρέωσης για τον παραπάνω χρόνο ομιλίας και το ίδιο για τα μηνύματα.
- 1.7. Βλάβη / επιδιόρθωση: Πότε έγινε, σε ποιά κεραία, τι είδους ήταν και ποίοι υπάλληλοι την επιδιορθώσανε.
- 1.8. Λογαριασμούς: ποσό χρέωσης, ημερομηνία έκδοση, ημερομηνία πληρωμής και σε ποίον αριθμό και πελάτη αντιστοιχεί.

2. Δυνατότητα διαγραφής και επεξεργασίας για όλες τις παραπάνω κατηγορίες.

3. Τρόπος σύνδεσης μιας κατηγορίας με μια άλλη.

3.1 Ένας πελάτης ανήκει σε ένα και μόνο ένα υποκατάστημα. Ένα υποκατάστημα έχει τουλάχιστον 2 υπαλλήλους.

3.2 Ένα υποκατάστημα έχει ένα και μόνο ένα υπάλληλο ως διευθυντή και ένας υπάλληλος είναι το πολύ σε ένα υποκατάστημα διευθυντής.

3.3 Ένας αριθμός συνδέεται με ένα το πολύ πελάτη αλλά ένας πελάτης μπορεί να έχει πολλούς αριθμούς.

3.4 Ένας υπάλληλος μπορεί να επιδιορθώνει πολλές κεραίες και μια κεραία μπορεί να επιδιορθώνεται από πολλούς υπαλλήλους.

3.5 Μια κεραία «ανήκει» σε ένα και μόνο ένα υποκατάστημα και ένα υποκατάστημα μπορεί να έχει πολλές κεραίες.

3.6 Ένας πελάτη που έχει σύνδεση έχει υπογράψει ένα και μόνο ένα συμβόλαιο με κάποιον υπάλληλο της εταιρίας και αυτό θα ανήκει σε ένα από τα πακέτα σύνδεσης.

4. Διαδικασίες που υποστηρίζονται.

4.1. Εισαγωγή και διαγραφή από τις πιο πάνω κατηγορίες.

4.2. Interface για την εισαγωγή του χρόνου ομιλίας και των μηνυμάτων που έγινε με την χρήση κάποιου αριθμού οπότε μετά θα υπολογίζεται το ποσόν χρέωσης ως

Ποσόν = πάγιο + [(λεπτά_ομιλίας - δωρεάν_λεπτά)/60]* extra + [αριθμός_μηνυμάτων - αριθμός_δωρεάν_μηνυμάτων]*χρέωση_μηνύματος

4.3. Αυτόματη μείωση του παγίου κατά 10% όταν συμπληρωθεί ένας χρόνος σύνδεσης.

4.4. Για κάθε ανανέωση συμβολαίου θα υπολογίζεται το ποσόν της επιδότησης και θα το εμφανίζει στην οθόνη. Το ποσό της επιδότησης προκύπτει ως εξής

Ποσόν_επιδότησης = άθροισμα_λογαριασμών_τελευταίου_χρόνου*0.12

4.5. Συγκεντρωτική αναφορά για τα νέα συμβόλαια και καρτοκινητά που κάνει τι κάθε υποκατάστημα σε ένα χρονικό διάστημα.

4.6. Συγκεντρωτική αναφορά για τις εισπράξεις κάθε καταστήματος σε ένα χρονικό διάστημα.

4.7. Συγκεντρωτική αναφορά για τις ανανεώσεις κάθε καταστήματος σε ένα χρονικό διάστημα.

4.8. Συγκεντρωτική αναφορά για κάθε υπάλληλο όσον αφορά α)τα νέα συμβόλαια που κάνει και β)τις ανανεώσεις

5.2.3 Απαιτήσεις λογισμικού



Το λογισμικό θα πρέπει να έχει την δυνατότητα διαχείρισης μέσω του διαδικτύου (over- the - web), και το συγκεκριμένο project πρέπει να είναι εύχρηστο, απλό και κατανοητό γιατί είναι ιδιαίτερα σημαντικό να μπορούν να το χρησιμοποιήσουν οι υπάλληλοι μιας εταιρίας ή και τα διοικητικά στελέχη της για την καλύτερη διαχείριση της εταιρίας τους.

Σαφώς σημαντικό λόγο παίζει και η ταχύτητα εκτέλεσης των εντολών του χρήστη και αυτό θα επιτευχθεί από την αποφυγή πολλών και περιττών Flash εφαρμογών και ανανεώσεων της σελίδας. Γι αυτό θα το επεξεργαστώ στο Dreamweaver όπου μου δίνεται η δυνατότητα αποφυγής πολλών επαναλήψεων και περιττών πληροφοριών στον κώδικα μου.

5.3 Entity- Relationship Model

5.3.1 Ορισμός E-R model

Στην τεχνολογία λογισμικού, το μοντέλο οντότητα συσχετίσεων (Relationship- Entity Model) είναι μια αφηρημένη και εννοιολογική αντιπροσώπευση των δεδομένων.

Η δημιουργία ενός μοντέλου οντοτήτων –συσχετίσεων είναι μια μέθοδος διαμόρφωσης βάσεων δεδομένων, που χρησιμοποιείται για να παραγάγει έναν τύπο εννοιολογικού σχήματος ή το σημασιολογικό πρότυπο στοιχείων ενός συστήματος, συχνά μια σχεσιακή βάση δεδομένων, και οι απαιτήσεις της σε μια τύπου από επάνω προς τα κάτω σχεδίαση (top to bottom design). Τα διαγράμματα που δημιουργούνται με αυτήν την διαδικασία καλούνται τα διαγράμματα οντότητα-σχέσης, διαγράμματα του ER ή συντομότερα ERDs.

Όταν μια σχεσιακή βάση δεδομένων πρόκειται να σχεδιαστεί, ένα διάγραμμα οντοτήτων – συσχετίσεων σχεδιάζεται σε ένα αρχικό στάδιο και αναπτύσσεται καθώς οι απαιτήσεις της βάσης δεδομένων και της επεξεργασίας της γίνονται καλύτερα κατανοητές. Οι λεπτομέρειες για το πώς να σχεδιαστούν τα διαγράμματα ποικίλλουν ελαφρώς από μια μέθοδο σε άλλη, αλλά όλες έχουν τα ίδια βασικά στοιχεία. Τους τύπους, τις ιδιότητες και τις σχέσεις οντοτήτων , αυτές οι τρεις κατηγορίες θεωρούνται επαρκείς για να διαμορφώσουν τα ουσιαστικά στατικά μέρη των αναγκών επεξεργασίας πληροφοριών οποιασδήποτε οργάνωσης σε μια οποιαδήποτε ΒΔ.

Στην πρώτη φάση της σχεδίασης πληροφοριακών συστημάτων χρησιμοποιούνται η σχεδίαση σε διαγράμματα, κατά την διάρκεια της οποίας, γίνεται ανάλυση των απαιτήσεων από τον σχεδιαστή προκειμένου να περιγραφούν οι απαιτήσεις πληροφοριών ή το είδος αυτών που θα αποθηκευτεί σε μια βάση δεδομένων.

5.3.2 Βασικά Δομικά Στοιχεία του E-R μοντέλου

Μια οντότητα μπορεί να οριστεί ως ένα πράγμα που αναγνωρίζεται ως ικανό μιας ανεξάρτητης ύπαρξης και που μπορεί να προσδιοριστεί μεμονωμένα. Οντότητα μπορεί να είναι φυσικό αντικείμενο όπως ένα σπίτι ή ένα αυτοκίνητο, ένα γεγονός όπως μια πώληση σπιτιών ή μια υπηρεσία αυτοκινήτων, ή μια έννοια όπως μια συναλλαγή ή μια διαταγή πελατών.

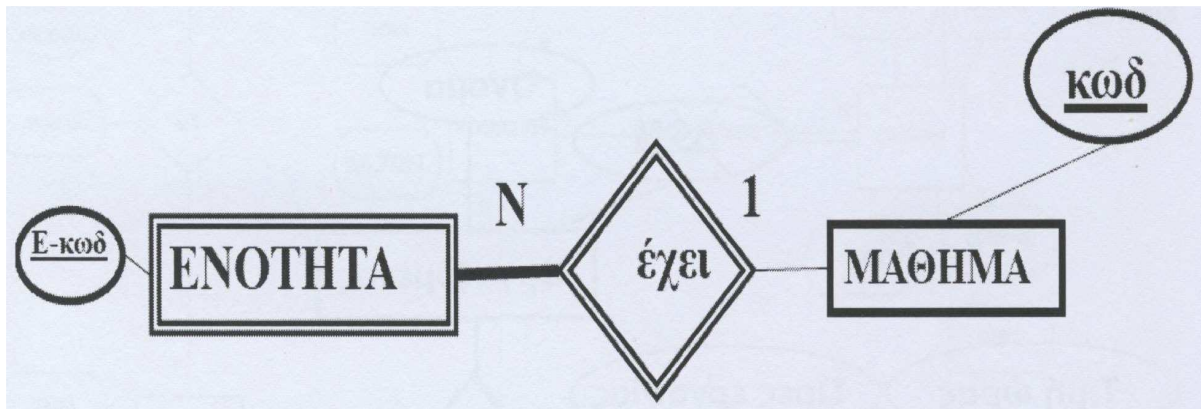
Μια σχέση συλλαμβάνει πώς δυο ή περισσότερες οντότητες συσχετίζονται με την μια την άλλη. Οι σχέσεις μπορεί να θεωρηθούν ως ρήματα, που συνδέουν δύο ή περισσότερα ουσιαστικά. Παράδειγμα: ένα υποκατάστημα προμηθεύεται από έναν προμηθευτή.

Ιδιότητες μπορούν και οι δύο να έχουν και οι οντότητες και οι σχέσεις. Παραδείγματα: μια οντότητα υπαλλήλων να έχει την δυνατότητα του μισθού τους, ενώ η σχέση μπορεί να έχει μια ιδιότητα ημερομηνίας πληρωμής. Κάθε οντότητα (εκτός αν είναι μια αδύνατη οντότητα) πρέπει να έχει κατά ελάχιστο μοναδική ιδιότητα με την οποία και αναγνωρίζεται κάθε της στιγμίου, το οποίο καλείται αρχικό κλειδί της οντότητας.

1. Γραμμές: για την σύνδεση των οντοτήτων με τα γνωρίσματα και με τις συσχετίσεις.
2. Ορθογώνια: που αναπαριστούν τα σύνολα των οντοτήτων.
3. Ρόμβοι: που αναπαριστούν τις συσχετίσεις (ή σχέσεις).
4. Ελλείψεις: που αναπαριστούν τα γνωρίσματα (ή ιδιότητες).
 - 4.1. Διπλές Ελλείψεις: Πλειότιμα γνωρίσματα.

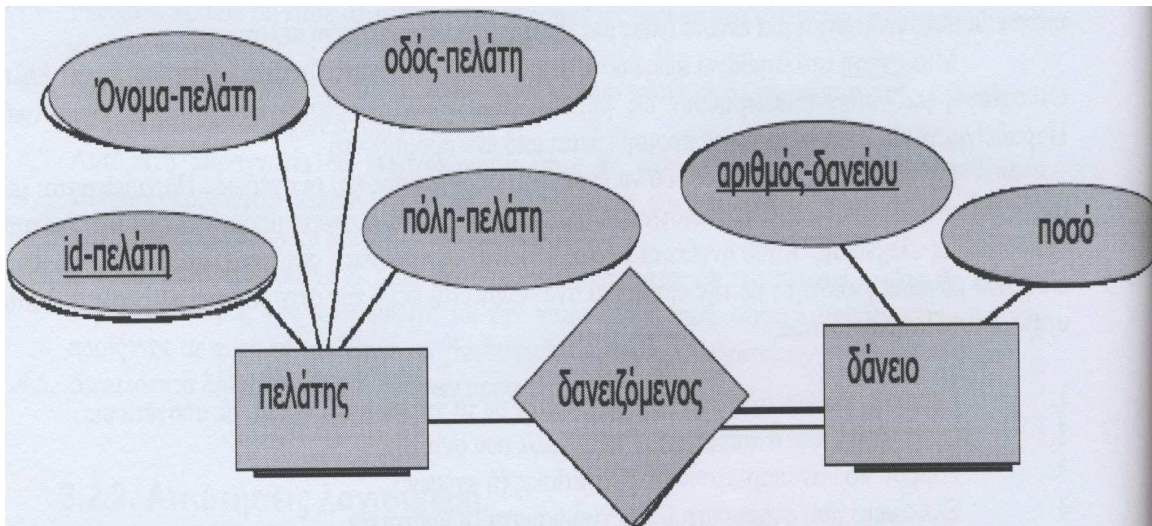


- 4.1. Διακεκομμένες Ελλείψεις: παραγόμενα γνωρίσματα.
- 5. Υπογράμμιση: γνωρίσματα που είναι πρωτεύοντα κλειδιά.
- 6. Ασθενείς οντότητες: (σχεδιάζεται με διπλό παραλληλόγραμμο) όταν μια οντότητα χρειάζεται να «δανειστεί» το κλειδί μιας άλλης.



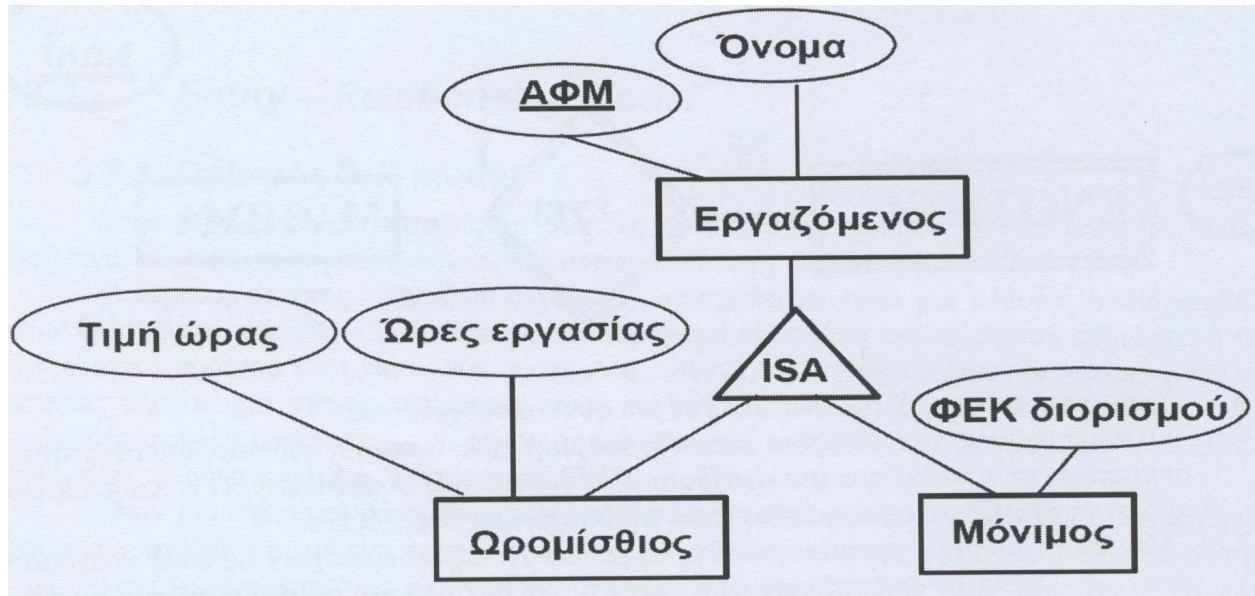
Εικόνα 7: Παράδειγμα σύνδεσης των οντοτήτων με τα γνωρίσματα τους και τις συσχετίσεις ανάμεσα τους.

- 7. Συμμετοχή ενός συνόλου οντοτήτων σε ένα σύνολο συσχετίσεων:
 - 7.1. Ολική συμμετοχή: (σχεδιάζεται με διπλή γραμμή) κάθε οντότητα του συνόλου οντοτήτων συμμετέχει σε τουλάχιστον μια συσχέτιση στο σύνολο των συσχετίσεων. Π.χ κάθε δάνειο πρέπει να σχετίζεται με έναν πελάτη μέσω της συσχέτισης δανειζόμενος.
 - 7.2. Μερική συμμετοχή: ορισμένες οντότητες μπορεί να μην συμμετέχουν σε κάθε συσχέτιση του συνόλου συσχετίσεων. Π.χ Μερική η συμμετοχή του πελάτη στο δανειζόμενος.
- Σχηματικά:



Εικόνα 8: Παράδειγμα σύνδεσης των οντοτήτων με τα γνωρίσματα τους και τις συσχετίσεις ανάμεσα τους.

- 8. Με την έννοια γενίκευση (generalization) εννοούμε τον εντοπισμό ενός συνόλου οντοτήτων (κλάση) που έχουν κοινά χαρακτηριστικά με πιο γενικευμένα αντικείμενα (υπερκλάση). Η εξειδίκευση (specialization) είναι το ακριβώς αντίθετο της γενίκευσης, δηλαδή ο εντοπισμός υποσυνόλων ενός τύπου οντοτήτων με κοινά χαρακτηριστικά, τα οποία τα διαφοροποιούν από τα υπόλοιπα μέλη του. Η συσχέτιση μεταξύ κάθε υποκλάσης και υπερκλάσης ονομάζεται ISA συσχέτιση.



Εικόνα 9: Παράδειγμα σύνδεσης των οντοτήτων με τα γνωρίσματα τους και τις συσχετίσεις ανάμεσα τους.

9. Η πληθικότητα (cardinality) – Ο λόγος πληθικότητας

Η πληθικότητα (cardinality), περιγράφει τον αριθμό στιγμιότυπων ενός τύπου οντοτήτων που μπορούν να αντιστοιχίζονται με μια οντότητα ενός άλλου τύπου σε μια συσχέτιση.

Ο λόγος πληθικότητας ή πληθικός λόγος (cardinality ratio), είναι ο λόγος των πληθικότητων μιας συσχέτισης.

Μπορούμε να έχουμε συσχετίσεις με λόγο πληθικότητας:

A) 1-1 (ένα – προς - ένα)

Αντιστοιχίζεται μια οντότητα ενός τύπου με το πολύ ή ακριβώς μια οντότητα ενός άλλου τύπου.

B) 1-N (ένα- προς- πολλά)

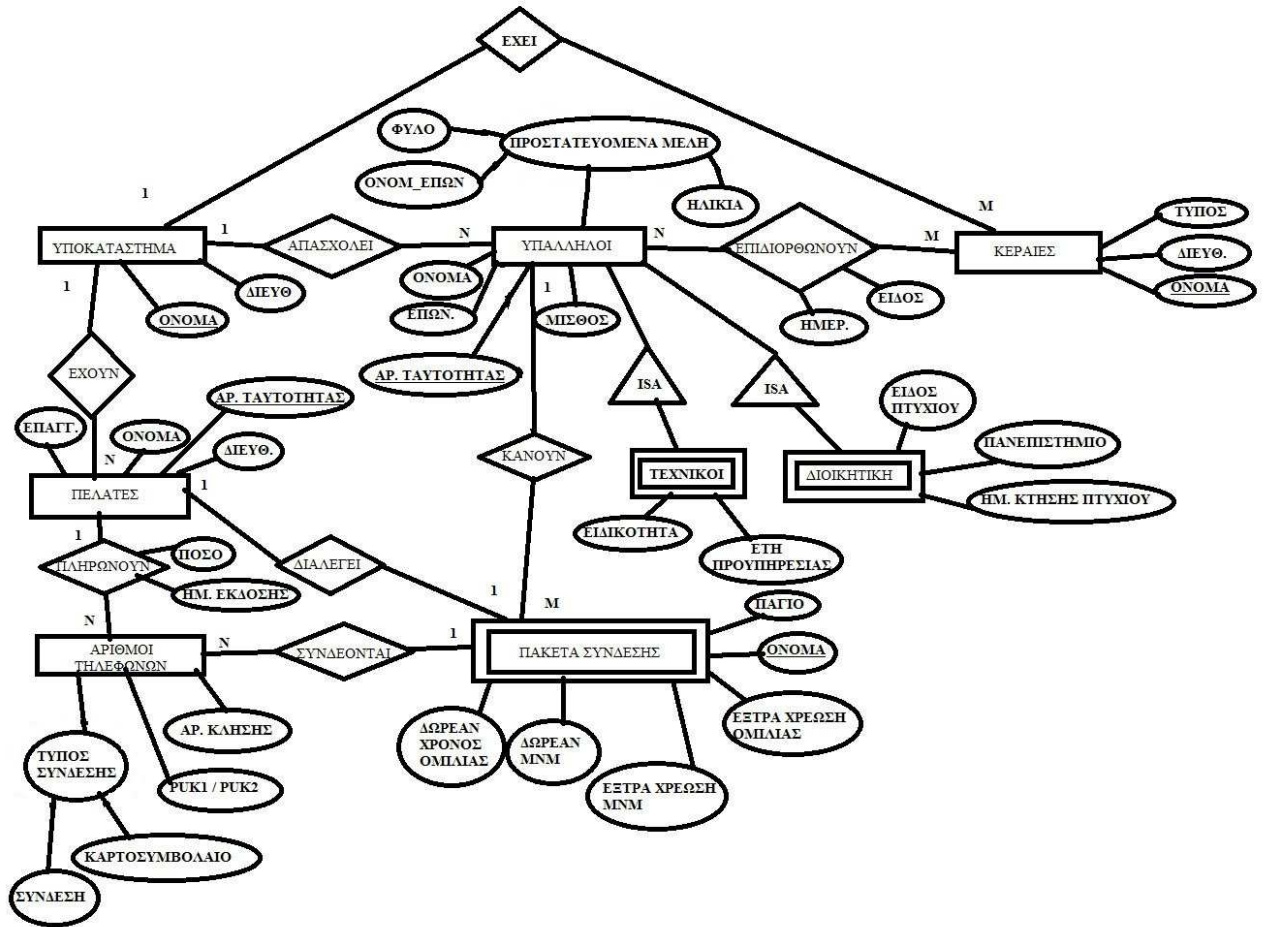
Αντιστοιχίζεται μια οντότητα ενός τύπου με κανένα, ένα ή πολλά στιγμιότυπα ενός άλλου τύπου.

Γ) M-N (πολλά – προς -πολλά)

Αντιστοιχίζεται κάθε στιγμιότυπο του ενός τύπου με ένα, κανένα ή πολλά στιγμιότυπα του άλλου τύπου.

5.3.3 Από E-R μοντέλο σε πίνακα (κανονικοποίηση)

Μέσω των Microsoft Paint των Windows 7 παρουσιάζεται το μοντέλο οντοτήτων συσχετίσεων που αφορά την εργασία το οποίο έγινε σε χαρτί κατά την αρχική μου σχεδίαση. Σε αυτό το σχήμα κατέληξα προκειμένου να είναι ορθές οι πληθικότητες όπως δόθηκαν από τις απαιτήσεις και να είναι εύκολη η πραγματοποίηση των απαραίτητων ειδικών διεργασιών.



Εικόνα 10: Το μοντέλο οντοτήτων συσχετίσεων που αφορά την εργασία μου.

Το σχεσιακό μοντέλο αναπαράστασης των δεδομένων μιας εφαρμογής (relational data model) καθιερώθηκε από τον Codd το 1970 και αποτέλεσε ένα από τα πιο απλά και ευέλικτα μοντέλα αυτού του είδους. Σε αυτό το μοντέλο, τα δεδομένα μιας εφαρμογής αναπαρίστανται ως ένα σύνολο από σχέσεις (relations) οι οποίες μπορεί να είναι πίνακες αρχεία. Στις πιο πολλές περιπτώσεις υιοθετείται η χρήση πινάκων (tablets) που περιέχουν ένα πλήθος γραμμών (rows) και στηλών (columns). Η κάθε μια από αυτές τις γραμμές – οι οποίες στην ορολογία του μοντέλου ονομάζονται και πλειάδες (tuples) – περιέχει ένα σύνολο απλών πεδίων (attributes) τα οποία συσχετίζονται μεταξύ τους. Επειδή όπως θα δούμε στις επόμενες παραγράφους οι πίνακες χρησιμοποιούνται για την αναπαράσταση των τύπων οντοτήτων καθώς και των τύπων συσχετίσεων που υφίστανται ανάμεσα τους, μπορούμε να θεωρήσουμε κάθε μια από τις γραμμές ενός πίνακα σαν ένα στιγμιότυπο οντότητας ή συσχέτισης ανάλογα με το αντικείμενο στο οποίο αναφέρεται.

Σχεδιασμός των τελικών σχέσεων (πινάκων) της βάσης δεδομένων με βάση το Διάγραμμα Οντοτήτων Συσχετίσεων. Στο παρακάτω κείμενο περιγράφουμε 7 απλά βήματα με τα οποία μπορεί κάποιος από ένα καλοσχεδιασμένο Διάγραμμα Οντοτήτων Συσχετίσεων (ΔΟΣ) να οδηγηθεί σε ένα ικανοποιητικό σχεδιασμό της βάσης δεδομένων, δηλ. στον προσδιορισμό των σχέσεων (πινάκων) με τα κατηγορήματα και τα κλειδιά τους. Η μέθοδος που δίνουμε στη συνέχεια εφαρμόζεται εύκολα και δεν απαιτεί ιδιαίτερη πείρα (είναι καθαρά αλγοριθμική), αρκεί να έχει σχεδιαστεί προηγουμένως ένα ορθό και πλήρες ΔΟΣ. Σημειώστε πως ο παρακάτω σχεδιασμός δεν είναι πάντα βέλτιστος, κάτι που επιτυγχάνεται με την εφαρμογή των κανόνων κανονικοποίησης που αποτελούν αντικείμενο πιο προχωρημένης μελέτης των βάσεων δεδομένων.



ΒΗΜΑ 1

Για κάθε κανονικό τύπο οντοτήτων E στο ΔΟΣ δημιουργήστε μια σχέση R που περιέχει όλα τα κατηγορήματα του E . Επιλέξτε ως πρωτεύον κλειδί της R ένα από τα κατηγορήματα κλειδιά του E . Αν το κλειδί του E είναι σύνθετο, το σύνολο των κατηγορημάτων του θα αποτελεί το πρωτεύον κλειδί της R .

ΒΗΜΑ 2

Για κάθε μη ισχυρό (ασθενή) τύπο οντοτήτων W στο ΔΟΣ με ιδιοκτήτη τον τύπο οντοτήτων E δημιουργήστε μια σχέση R στην οποία συμπεριλάβετε όλα τα κατηγορήματα του W ως κατηγορήματα της R . Επιπλέον συμπεριλάβετε ως κατηγορήματα της R τα κατηγορήματα των κλειδιών των σχέσεων που δημιουργήσατε για τους τύπους οντοτήτων – ιδιοκτήτες. Το κλειδί της R είναι ο συνδυασμός του κλειδιού (κλειδιών) του ιδιοκτήτη (των ιδιοκτητών) και αν υπάρχει, του μερικού κλειδιού του μη ισχυρού (ασθενούς) τύπου W .

ΒΗΜΑ 3

Για κάθε 1:1 δυαδική συσχέτιση R στο ΔΟΣ βρείτε τις σχέσεις S και T που αντιστοιχούν στους τύπους οντοτήτων που συμμετέχουν στην R . Επιλέξτε μια από τις σχέσεις (έστω την S) και συμπεριλάβετε την S , ως νέο γνώρισμα (ξένο κλειδί), το πρωτεύον κλειδί της T . Συμπεριλαμβάνοντας επίσης ως κατηγορήματα της S όλα τα κατηγορήματα της συσχέτισης R (αν υπάρχουν).

ΒΗΜΑ 4

Για κάθε 1:N δυαδική συσχέτιση R στο ΔΟΣ βρείτε τις σχέσεις S και T που αντιστοιχούν στους τύπους οντοτήτων που συμμετέχουν στην R . Εστω ότι ο T ο συμμετέχων τύπος οντοτήτων από την πλευρά N . Συμπεριλαμβάνοντας ως νέο γνώρισμα (ξένο κλειδί) στη σχέση T το πρωτεύον κλειδί της σχέσης S . Συμπεριλαμβάνονται επίσης ως κατηγορήματα της T όλα τα κατηγορήματα της συσχέτισης R (αν υπάρχουν).

ΒΗΜΑ 5

Για κάθε $M:N$ δυαδική συσχέτιση R στο ΔΟΣ δημιουργείται μια νέα σχέση S για να παρασταθεί η R . Συμπεριλαμβάνετε ως κατηγορήματα στην S τα πρωτεύοντα κλειδιά των σχέσεων που παριστάνουν τους συμμετέχοντες τύπους οντοτήτων. Ο συνδυασμός τους θα αποτελεί το πρωτεύον κλειδί της S . Συμπεριλαμβάνεται επίσης ως κατηγορήματα στην S όλα τα κατηγορήματα της συσχέτισης R (αν υπάρχουν).

Σημείωση: οι συσχετίσεις 1:1 και 1:N μπορούν να απεικονιστούν με τον ίδιο τρόπο με τις $M:N$ (δηλ. με την δημιουργία νέας σχέσης). Αυτή η προσέγγιση μπορεί να είναι καλύτερη όταν η συσχέτιση έχει λίγα στιγμιότυπα προκειμένου να αποφεύγονται τιμές NULL στα ξένα κλειδιά.

Το επόμενο βήμα αντιμετωπίζει τα πλειότιμα κατηγορήματα, δηλ. κατηγορήματα για τα οποία επιτρέπεται να δοθούν περισσότερες από μια τιμές (από ένα συγκεκριμένο πεδίο τιμών) για μια συγκεκριμένη εγγραφή. Υπενθυμίζουμε πως οντότητες που αντιστοιχίζονται σε σχέσεις με πλειότιμα κατηγορήματα (ΒΗΜΑ 1) δεν είναι νόμιμες.

ΒΗΜΑ 6

Για κάθε πλειότιμο γνώρισμα A κατασκευάστε μια νέα σχέση R η οποία θα περιλαμβάνει ένα γνώρισμα που θα αντιστοιχεί στο A καθώς και το γνώρισμα του πρωτεύοντος κλειδιού K της σχέσης που παριστάνει τον τύπο οντοτήτων ή τον τύπο συσχέτισης που έχει το R ως γνώρισμα. Πρωτεύον κλειδί της R είναι ο συνδυασμός του A και K .



ΒΗΜΑ 7

Για κάθε n -αδικό τύπο συσχέτισης R με $n > 2$, κατασκευάστε μια νέα σχέση S για την παράσταση του R . Συμπεριλαμβάνονται ως κατηγορήματα της S τα πρωτεύοντα κλειδιά των σχέσεων που παριστάνουν τους συμμετέχοντες τύπους οντοτήτων. Συμπεριλαμβάνετε επίσης ως κατηγορήματα της S όλα τα κατηγορήματα της συσχέτισης R (αν υπάρχουν). Το πρωτεύον κλειδί της S είναι συνήθως ένας συνδυασμός των (ξένων) κλειδιών που συμμετέχουν στην S .

Έτσι ακολουθώντας αυτά τα βήματα καταλήγουμε στους εξής πίνακες:

ΥΠΟΚΑΤΑΣΤΗΜΑ

ΟΝΟΜΑ	ΔΙΕΥΘ.
-------	--------

ΥΠΑΛΛΗΛΟΣ

ΟΝΟΜΑ	ΕΠΩΝ.	Α.Τ ΥΠΑΛΛ.	ΦΥΛΟ	ΟΝΟΜΑ_ ΠΡΟΣΤΑΤΕΥΜΕΝΟΥ ΜΕΛΟΥΣ	ΗΛΙΚΙΑ	ΜΙΣΘΟΣ	ΟΝΟΜΑ_ΥΠΟΚ.
-------	-------	---------------	------	------------------------------------	--------	--------	-------------

ΤΕΧΝΙΚΟΣ

Α.Τ_ΥΠΑΛΛ	ΕΙΔΙΚΟΤΗΤΑ	ΕΤΗ ΠΡΟΥΠΗΡΕΣΙΑΣ
-----------	------------	------------------

ΔΙΟΙΚΗΤΙΚΟΣ

Α.Τ_ΥΠΑΛΛ	ΕΙΔΟΣ_ΠΤΥΧΙΟΥ	ΠΑΝΕΠΙΣΤΗΜΙΟ	ΗΜ. ΚΤΗΣΗΣ ΠΤΥΧΙΟΥ
-----------	---------------	--------------	--------------------

ΕΠΙΔΙΟΡΘΩΝΟΥΝ

Α.Τ_ΥΠΑΛΛ	ΟΝ_ΚΕΡΑΙΑΣ	ΗΜΕΡ.	ΕΙΔΟΣ
-----------	------------	-------	-------

ΚΕΡΑΙΕΣ

ΟΝΟΜΑ	ΔΙΕΥΘ.	ΤΥΠΟΣ	ΟΝΟΜΑ_ΥΠΟΚ.
-------	--------	-------	-------------

ΠΕΛΑΤΕΣ

ΟΝΟΜΑ	ΔΙΕΥΘ.	Α.Τ. ΠΕΛΑΤΗ	ΕΠΑΓΓ.	ΟΝ. ΥΠΟΚΑΤΑΣΤΗΜΑΤΟΣ
-------	--------	-------------	--------	---------------------



ΠΛΗΡΩΝΟΥΝ

ΑΡ. ΚΛΗΣΗΣ	ΠΟΣΟ	ΗΜ. ΕΚΔΟΣΗΣ	Α.Τ. ΠΕΛΑΤΗ	ΗΜ. ΠΛΗΡΩΜΗΣ
-------------------	-------------	--------------------	--------------------	---------------------

ΑΡ. ΤΗΛΕΦΩΝΟΥ

ΑΡ. ΚΛΗΣΗΣ	PUK1 / PUK2	ΣΥΝΔΕΣΗ/ ΚΑΡΤΟΤΗΛΕΦΩΝΟ	Α.Τ.ΠΕΛΑΤΗ	ΟΝΟΜΑ ΠΑΚ. ΣΥΝΔ.
-------------------	--------------------	-------------------------------	-------------------	-------------------------

ΠΑΚΕΤΑ ΣΥΝΔΕΣΗΣ

Α.Τ_ΥΠΑΛΛ.	Α.Τ_ΠΕΛΑΤΩΝ	ΔΩΡΕΑΝ ΛΕΠΤΑ ΟΜΙΛΙΑΣ	ΔΩΡΕΑΝ ΜΝΜ	ΕΞΤΡΑ ΧΡΕΩΣΗ ΜΝΜ	ΕΞΤΡΑ ΧΡΕΩΣΗ ΟΜΙΛΙΑΣ	ΠΑΓΙΟ	ΟΝΟΜΑ
-------------------	--------------------	-------------------------------------	-----------------------	---------------------------------	-------------------------------------	--------------	--------------



ΚΕΦΑΛΑΙΟ 6 : SQL Κώδικας

6.1 Εξαγωγή SQL κώδικα

Σε αυτό το κομμάτι της εργασίας θα παρουσιαστεί και θα επεξηγηθεί ο κώδικας SQL που θα χρησιμοποιήσουμε για την δημιουργία της Β.Δ., δηλ. την μετατροπή των πινάκων της παραγράφου 3.2 σε βάση δεδομένων για υπολογιστή ή δίκτυο.

Πρέπει πρώτα όμως να καταλάβουμε τι κάνουν κάποιες λέξεις (ή φράσεις) κλειδιά που φαίνονται στον παρακάτω κώδικα MySQL. Την εντολή Drop Table χρησιμοποιείται για ασφάλεια και προστατεύει την βάση από επιπλοκές με άλλο πίνακα που πιθανόν να είχε το ίδιο όνομα. Στην ουσία διαγράφει όλα τα tablets με τέτοιο όνομα. Για την δημιουργία ενός πίνακα ακολουθείτε αυτό το πρότυπο

-> CREATE TABLE 'όνομα της βάσης'. 'όνομα του πίνακα'. Κατόπιν, δηλώνουμε της στήλες του πίνακα (γνωρίσματα) π.χ αν είναι αλφαριθμητική varchar, αν είναι ακέραιος int κ.α και το πρωτεύον κλειδί.

Σε περίπτωση, που υπάρχει ξένο κλειδί γίνονται τρία πράγματα. Αρχικά, η εντολή foreign key ('όνομα πεδίου') για την ονομασία του. Στη συνέχεια, δείχνουμε σε ποιο κλειδί άλλου πίνακα αναφέρεται αυτό -> REFERENCES 'όνομα της βάσης'. 'όνομα του πίνακα που αναφέρεται' ('όνομα του πρωτεύον κλειδιού'). Τελικά, επιλέγουμε ανάμεσα απ' το Cascade ή No Action για το τι θα συμβεί όταν σε ανανέωση ή διαγραφή στον πίνακα της αναφοράς (No Action: καμία αλλαγή δεν επιτρέπεται, Cascade: επιτρέπονται αλλαγές).

6.2 Εξαγωγή του κώδικα SQL κάθε υποκαταστήματος

Για την δημιουργία των πινάκων σε SQL άνοιξα το localhost του xampp (<https://localhost/phpmyadmin/>). Φτιάχνω εκεί την βάση μου (εννοείται πως για κάθε υποκατάστημα είναι διαφορετική). Φτιάχνω και εσωτερικούς πίνακες (πχ κεραία, υπάλληλος κτλ) που θα κρατούν τα στοιχεία των πεδίων σε κάθε υποκατάστημα. Τα στοιχεία είναι κανωνικοποιημένα με βάση τους όρους κανονικοποίησης των βάσεων.

The screenshot displays the phpMyAdmin web interface. On the left, a sidebar lists databases: information_schema (37), iraklio (6), lasithi (6), mysql (24), performance_schema (17), rethimno (6), and xania (6). The main area is titled 'Server: localhost' and contains several tabs: Databases, SQL, Status, Variables, Charsets, Engines, Privileges, Processes, Export, and Import. The 'Actions' section for 'MySQL localhost' features a 'Create new database' form with a 'Collation' dropdown set to 'utf8_general_ci' and a 'Create' button. Below this is the 'Interface' section with settings for Language (English), Theme / Style (Original), Custom color (Reset), and Font size (82%). The right sidebar provides system information: 'MySQL' section shows 'Server: localhost via TCP/IP', 'Server version: 5.5.16', 'Protocol version: 10', 'User: root@localhost', and 'MySQL charset: UTF-8 Unicode (utf8)'. The 'Web server' section lists 'Apache/2.2.12 (Win32) DAV/2 mod_ssl/2.2.12 OpenSSL/0.9.8k mod_autoindex_color PHP/5.3.0 mod_perl/2.0.4 Perl/v5.10.0', 'MySQL client version: 5.1.37', and 'PHP extension: mysqli'. At the bottom right, the 'phpMyAdmin' section shows 'Version information: 3.2.0.1' and links for 'Documentation', 'Wiki', 'Official Homepage', and '[ChangeLog] [Subversion] [Lists]'. The phpMyAdmin logo is visible in the bottom right corner.

Εικόνα 11: Το localhost του xampp όπου δημιούργησα και περιέχει τις βάσεις μου.



Για να τρέξω τα html αρχεία μου τα βάζω όλα μαζί με τις φωτογραφίες μου στο σκληρό δίσκο μου τον C μετά χαμpp-> htdocs (C:\χαμpp\htdocs). Εκεί αποθηκεύω σε κατηγορίες τα αρχεία μου (κάθε βάση ξεχωριστά ανά νομό Χανιά – Ηράκλειο – Ρέθυμνο – Λασιθί).

Index of /xania

Name	Last modified	Size	Description
Parent Directory			-
arxiki.html	29-□□□-2011 20:02	4.2K	
epikoinonies.html	04-□□□-2011 06:26	3.9K	
images/	29-□□□-2011 20:03	-	
ipalilos.php	17-□□□-2011 16:47	1.2K	
ipokastima_ipalilo.>	10-□□□-2011 11:48	8.7K	
ipokastima_kerea.>	03-□□□-2011 16:22	4.5K	
ipokastima_logaria.>	10-□□□-2011 11:51	4.7K	
ipokastima_paketa.>	29-□□□-2011 18:40	7.3K	
ipokastima_pelati.>	10-□□□-2011 11:51	5.4K	
ipokastima_tilefo.>	04-□□□-2011 19:18	4.7K	
ipokastima_vlaves.>	10-□□□-2011 11:49	4.6K	
kerea.php	07-□□□-2011 20:58	460	
logariasmoi.php	13-□□□-2011 17:24	1.7K	
pelatis.php	13-□□□-2011 17:11	3.2K	
tilefona.php	07-□□□-2011 20:59	455	
vlaves.php	07-□□□-2011 20:59	371	

Apache/2.2.12 (Win32) DAV/2 mod_ssl/2.2.12 OpenSSL/0.9.8k mod_autoindex_color PHP/5.3.0 mod_perl/2.0.4 Perl/v5.10.0 Server at localhost Port 80

Index of /rethimno

Name	Last modified	Size	Description
Parent Directory			-
arxiki.html	13-◆◆◆-2012 19:29	4.1K	
epikoinonies.html	14-◆◆◆-2012 09:27	3.8K	
images/	15-◆◆◆-2012 12:41	-	
ipalilos.php	30-◆◆◆-2011 15:47	1.0K	
ipokastima_arxiki.>	15-◆◆◆-2012 12:58	4.7K	
ipokastima_ipalilo.>	14-◆◆◆-2012 09:30	8.8K	
ipokastima_kerea.>	14-◆◆◆-2012 09:32	4.6K	
ipokastima_logaria.>	14-◆◆◆-2012 09:30	4.8K	
ipokastima_paketa.>	14-◆◆◆-2012 09:30	7.4K	
ipokastima_pelati.>	14-◆◆◆-2012 09:32	5.5K	
ipokastima_tilefo.>	14-◆◆◆-2012 09:32	4.8K	
ipokastima_vlaves.>	14-◆◆◆-2012 09:32	4.7K	
kerea2.php	09-◆◆◆-2011 15:56	464	
logariasmoi2.php	09-◆◆◆-2011 15:56	627	
pelatis.php	10-◆◆◆-2011 11:12	705	
tilefona2.php	09-◆◆◆-2011 15:56	459	
vlaves2.php	09-◆◆◆-2011 15:56	375	

Apache/2.2.12 (Win32) DAV/2 mod_ssl/2.2.12 OpenSSL/0.9.8k mod_autoindex_color PHP/5.3.0 mod_perl/2.0.4 Perl/v5.10.0 Server at localhost Port 80



Index of /iraklio

Name	Last modified	Size	Description
Parent Directory		-	
arxiki.html	13--2012 19:29	4.1K	
epikoinonies.html	14--2012 09:27	3.8K	
images/	15--2012 12:41	-	
ipalilos.php	30--2011 15:48	1.0K	
ipokatastima_arxiki.>	15--2012 12:59	4.7K	
ipokatastima_ipalilo.>	14--2012 09:33	8.8K	
ipokatastima_kerea.>	14--2012 09:37	4.6K	
ipokatastima_logaria.>	14--2012 09:33	4.8K	
ipokatastima_paketa.>	14--2012 09:33	7.4K	
ipokatastima_pelati.>	14--2012 09:36	5.5K	
ipokatastima_tilefo.>	14--2012 09:36	4.8K	
ipokatastima_vlaves.>	14--2012 09:35	4.7K	
kerea1.php	07--2011 20:55	463	
logariasmoi1.php	07--2011 20:55	626	
pelatis.php	10--2011 12:19	812	
tilefona1.php	07--2011 20:56	458	
vlaves1.php	07--2011 20:56	374	

Apache/2.2.12 (Win32) DAV/2 mod_ssl/2.2.12 OpenSSL/0.9.8k mod_autoindex_color PHP/5.3.0 mod_perl/2.0.4 Perl/v5.10.0 Server at localhost Port 80

Index of /lasithi

Name	Last modified	Size	Description
Parent Directory		-	
arxiki.html	13--2012 19:29	4.1K	
epikoinonies.html	14--2012 09:27	3.8K	
images/	15--2012 12:41	-	
ipalilos.php	30--2011 15:49	1.0K	
ipokatastima_arxiki.>	15--2012 12:59	4.7K	
ipokatastima_ipalilo.>	14--2012 09:44	8.8K	
ipokatastima_kerea.>	14--2012 09:45	4.6K	
ipokatastima_logaria.>	14--2012 09:43	4.8K	
ipokatastima_paketa.>	14--2012 09:42	7.4K	
ipokatastima_pelati.>	14--2012 09:45	5.5K	
ipokatastima_tilefo.>	14--2012 09:45	4.8K	
ipokatastima_vlaves.>	14--2012 09:44	4.7K	
kerea3.php	08--2011 18:13	465	
logariasmoi3.php	08--2011 18:13	628	
pelatis.php	10--2011 12:25	813	
tilefona3.php	08--2011 18:13	460	
vlaves3.php	08--2011 18:12	376	

Apache/2.2.12 (Win32) DAV/2 mod_ssl/2.2.12 OpenSSL/0.9.8k mod_autoindex_color PHP/5.3.0 mod_perl/2.0.4 Perl/v5.10.0 Server at localhost Port 80

Εικόνα 12: Τα parent directory που αποθηκεύω τα αρχεία μου για να τρέξουν για το κάθε υποκατάστημα.



6.2.1 Εξαγωγή του κώδικα SQL της βάσης των Χανίων

- ❖ Για βλάβες:

```
CREATE TABLE `xania`.`vlaves` (  
  `when` DATE NULL ,  
  `kerea_name1` VARCHAR( 20 ) NULL ,  
  `ergates` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `kerea_name1` )  
  ) ENGINE = InnoDB;
```

- ❖ Για λογαριασμούς:

```
CREATE TABLE `xania`.`logariasmoi` (  
  `poso_pliromis` VARCHAR( 7 ) NOT NULL ,  
  `date_ekdosis` DATE NOT NULL ,  
  `date_pliromis` DATE NOT NULL ,  
  `pelatis_name` VARCHAR( 60 ) NOT NULL ,  
  `phone_number` VARCHAR( 10 ) NOT NULL ,  
  `ADD PRIMARY KEY ( `phone_number` )  
  ) ENGINE = InnoDB;
```

- ❖ Για υπαλλήλους:

```
CREATE TABLE `xania`.`ipalilos` (  
  `ipalilos_name` VARCHAR( 60 ) NULL ,  
  `id_ipalilos` VARCHAR( 10 ) NULL ,  
  `misthos` VARCHAR( 7 ) NULL ,  
  `name_prostateuomeno_melos` VARCHAR( 60 ) NULL ,  
  `ilikia` VARCHAR( 2 ) NULL ,  
  `filo` VARCHAR( 8 ) NULL ,  
  `name_prostateuomeno_melos1` VARCHAR( 60 ) NULL ,  
  `ilikia1` VARCHAR( 2 ) NULL ,  
  `filo1` VARCHAR( 8 ) NULL ,  
  `eidikotita` VARCHAR( 60 ) NULL ,  
  `etoi_proetimasias` VARCHAR( 2 ) NULL ,  
  `titlos_spoudon` VARCHAR( 60 ) NULL ,  
  `imerominia_ptixiou` DATE NULL ,  
  `panepistimio` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `id_ipalilos` )  
  ) ENGINE = InnoDB;
```

- ❖ Για πελάτες:

```
CREATE TABLE `xania`.`pelatis` (  
  `pelatis_name` VARCHAR( 60 ) NULL ,  
  `pelatis_job` VARCHAR( 50 ) NULL ,  
  `pelatis_id` VARCHAR( 10 ) NULL ,  
  `pelatis_location` VARCHAR( 60 ) NULL ,  
  `xronos_omilias` VARCHAR( 6 ) NULL ,  
  `minimata` VARCHAR( 6 ) NULL ,
```



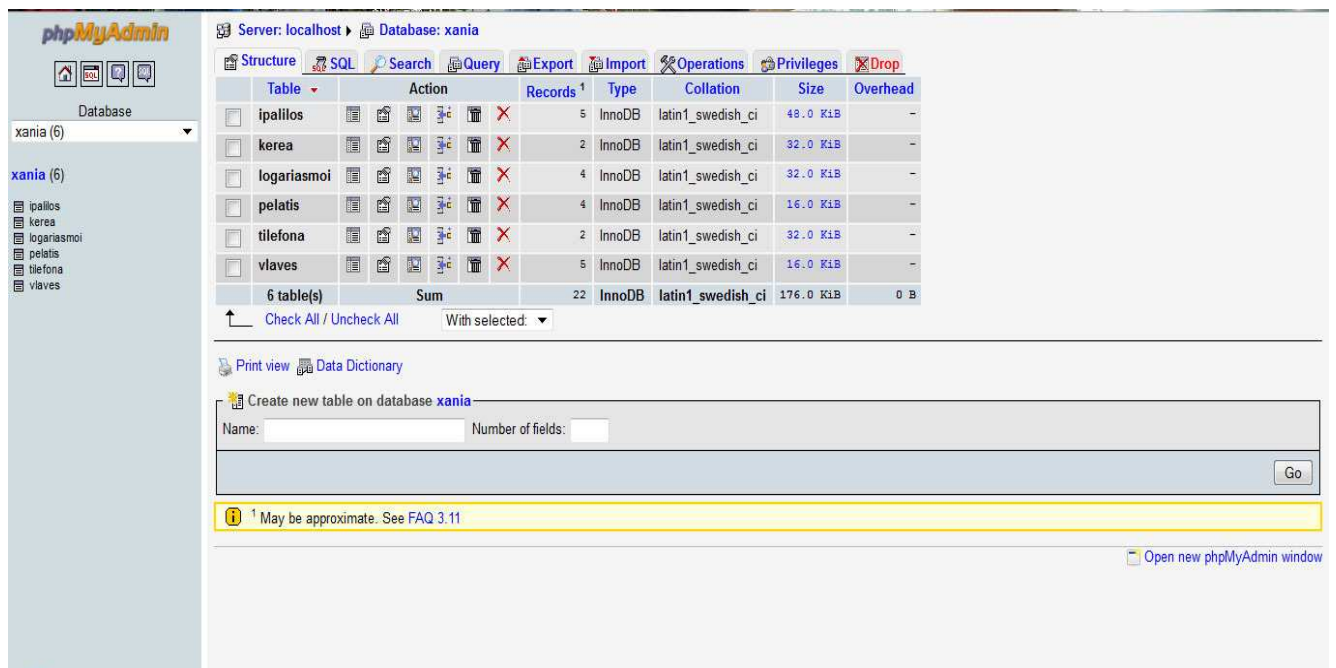
```
`paketo` VARCHAR( 15 ) NULL ,  
`date_connection` DATE NULL ,  
ADD PRIMARY KEY ( `pelatis_ld` )  
 ) ENGINE = InnoDB;
```

❖ Για τηλέφωνα:

```
CREATE TABLE `xania`.`telefona` (  
`phone_number` VARCHAR( 10 ) NULL ,  
`puk1` VARCHAR( 10 ) NULL ,  
`puk2` VARCHAR( 10 ) NULL ,  
`paketa` VARCHAR( 20 ) NULL ,  
PRIMARY KEY ( `phone_number` )  
 ) ENGINE = InnoDB;
```

❖ Για κεραία

```
CREATE TABLE `xania`.`kerea` (  
`kerea_name` VARCHAR( 20 ) NULL ,  
`kerea_type` VARCHAR( 20 ) NULL ,  
`kerea_location` VARCHAR( 60 ) NULL ,  
PRIMARY KEY ( `kerea_name` )  
 ) ENGINE = InnoDB;
```



Εικόνα 13: Η βάση των Χανίων με τους πίνακες που περιέχει στο εσωτερικό της.



6.2.2 Εξαγωγή του κώδικα SQL της βάσης του Ηρακλείου

❖ Για βλάβες:

```
CREATE TABLE `iraklio`.`vlaves1` (  
  `when` DATE NULL ,  
  `kerea_name1` VARCHAR( 20 ) NULL ,  
  `ergates` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `kerea_name1` )  
  ) ENGINE = InnoDB;
```

❖ Για λογαριασμούς:

```
CREATE TABLE `iraklio`.`logariasmoi1` (  
  `poso_pliromis` VARCHAR( 7 ) NOT NULL ,  
  `date_ekdosis` DATE NOT NULL ,  
  `date_pliromis` DATE NOT NULL ,  
  `pelatis_name` VARCHAR( 60 ) NOT NULL ,  
  `phone_number` VARCHAR( 10 ) NOT NULL ,  
  ADD PRIMARY KEY ( `phone_number` )  
  ) ENGINE = InnoDB;
```

❖ Για υπαλλήλους:

```
CREATE TABLE `iraklio`.`ipalilos1` (  
  `ipalilos_name` VARCHAR( 60 ) NULL ,  
  `id_ipalilos` VARCHAR( 10 ) NULL ,  
  `misthos` VARCHAR( 7 ) NULL ,  
  `name_prostateuomeno_melos` VARCHAR( 60 ) NULL ,  
  `ilikia` VARCHAR( 2 ) NULL ,  
  `filo` VARCHAR( 8 ) NULL ,  
  `name_prostateuomeno_melos1` VARCHAR( 60 ) NULL ,  
  `ilikia1` VARCHAR( 2 ) NULL ,  
  `filo1` VARCHAR( 8 ) NULL ,  
  `eidikotita` VARCHAR( 60 ) NULL ,  
  `etoi_proetimasias` VARCHAR( 2 ) NULL ,  
  `titlos_spoudon` VARCHAR( 60 ) NULL ,  
  `imerominia_ptixiou` DATE NULL ,  
  `panepistimio` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `id_ipalilos` )  
  ) ENGINE = InnoDB;
```

❖ Για πελάτες:

```
CREATE TABLE `iraklio`.`pelates1` (  
  `pelatis_name` VARCHAR( 60 ) NULL ,  
  `pelatis_job` VARCHAR( 50 ) NULL ,  
  `pelatis_id` VARCHAR( 10 ) NULL ,  
  `pelatis_location` VARCHAR( 60 ) NULL ,  
  `xronos_omilias` VARCHAR( 6 ) NULL ,  
  `minimata` VARCHAR( 6 ) NULL ,  
  `paketo` VARCHAR( 15 ) NULL ,  
  `date_connection` DATE NULL ,  
  ADD PRIMARY KEY ( `pelatis_id` )
```



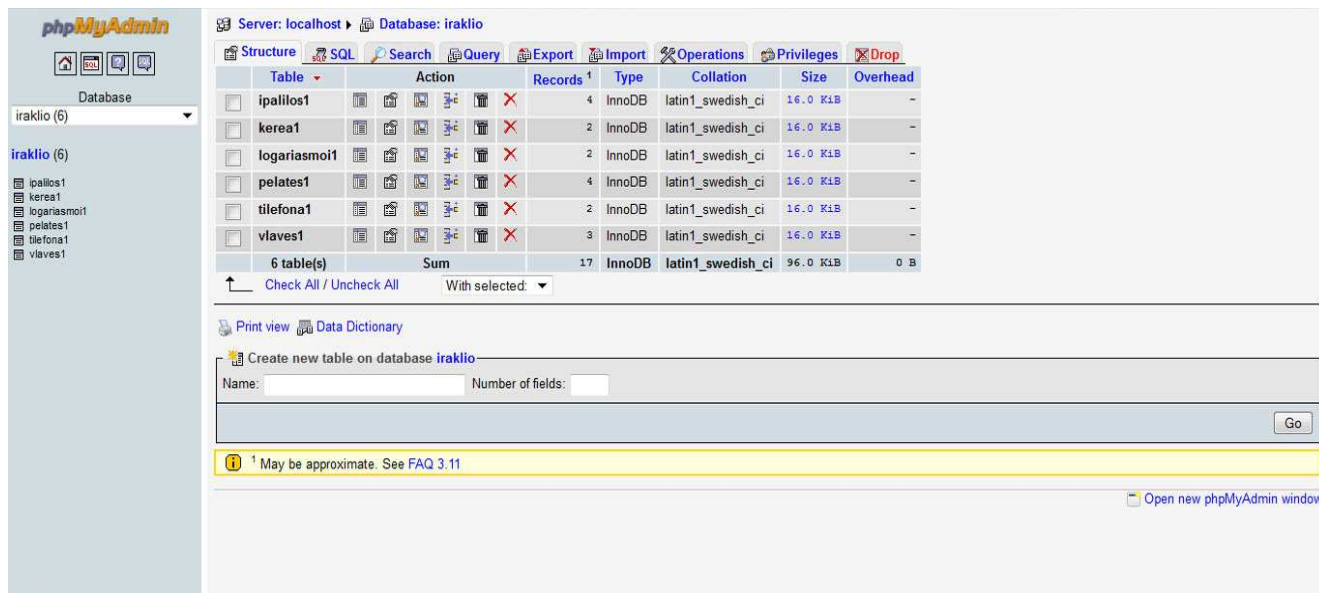
```
) ENGINE = InnoDB;
```

❖ Για τηλέφωνα:

```
CREATE TABLE `iraklio`.`tелефona1` (  
  `phone_number` VARCHAR( 10 ) NULL ,  
  `puk1` VARCHAR( 10 ) NULL ,  
  `puk2` VARCHAR( 10 ) NULL ,  
  `paketa` VARCHAR( 20 ) NULL ,  
  PRIMARY KEY ( `phone_number` )  
  ) ENGINE = InnoDB;
```

❖ Για κεραιά:

```
CREATE TABLE `iraklio`.`kerea1` (  
  `kerea_name` VARCHAR( 20 ) NULL ,  
  `kerea_type` VARCHAR( 20 ) NULL ,  
  `kerea_location` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `kerea_name` )  
  ) ENGINE = InnoDB;
```



Εικόνα 14: Η βάση του Ηρακλείου με τους πίνακες που περιέχει στο εσωτερικό της.

6.2.3 Εξαγωγή του κώδικα SQL της βάσης του Ρεθύμνου

❖ Για βλάβες:

```
CREATE TABLE `rethimno`.`vlaves2` (  
  `when` DATE NULL ,  
  `kerea_name1` VARCHAR( 20 ) NULL ,  
  `ergates` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `kerea_name1` )  
  ) ENGINE = InnoDB;
```



- ❖ Για λογαριασμούς:

```
CREATE TABLE `rethimno`.`logariasmoi2` (  
  `poso_pliromis` VARCHAR( 7 ) NOT NULL ,  
  `date_ekdosis` DATE NOT NULL ,  
  `date_pliromis` DATE NOT NULL ,  
  `pelatis_name` VARCHAR( 60 ) NOT NULL ,  
  `phone_number` VARCHAR( 10 ) NOT NULL ,  
  `ADD PRIMARY KEY ( `phone_number` )  
  ) ENGINE = InnoDB;
```

- ❖ Για υπαλλήλους:

```
CREATE TABLE `rethimno`.`ipalilos2` (  
  `ipalilos_name` VARCHAR( 60 ) NULL ,  
  `id_ipalilos` VARCHAR( 10 ) NULL ,  
  `misthos` VARCHAR( 7 ) NULL ,  
  `name_prostateuomeno_melos` VARCHAR( 60 ) NULL ,  
  `ilikia` VARCHAR( 2 ) NULL ,  
  `filo` VARCHAR( 8 ) NULL ,  
  `name_prostateuomeno_melos1` VARCHAR( 60 ) NULL ,  
  `ilikia1` VARCHAR( 2 ) NULL ,  
  `filo1` VARCHAR( 8 ) NULL ,  
  `eidikotita` VARCHAR( 60 ) NULL ,  
  `etoi_proetimasias` VARCHAR( 2 ) NULL ,  
  `titlos_spoudon` VARCHAR( 60 ) NULL ,  
  `imerominia_ptixiou` DATE NULL ,  
  `panepistimio` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `id_ipalilos` )  
  ) ENGINE = InnoDB;
```

- ❖ Για πελάτες:

```
CREATE TABLE `rethimno`.`pelates2` (  
  `pelatis_name` VARCHAR( 60 ) NULL ,  
  `pelatis_job` VARCHAR( 50 ) NULL ,  
  `pelatis_ld` VARCHAR( 10 ) NULL ,  
  `pelatis_location` VARCHAR( 60 ) NULL ,  
  `xronos_omilias` VARCHAR( 6 ) NULL ,  
  `minimata` VARCHAR( 6 ) NULL ,  
  `paketo` VARCHAR( 15 ) NULL ,  
  `date_connection` DATE NULL ,  
  ADD PRIMARY KEY ( `pelatis_ld` )  
  ) ENGINE = InnoDB;
```

- ❖ Για τηλέφωνα:

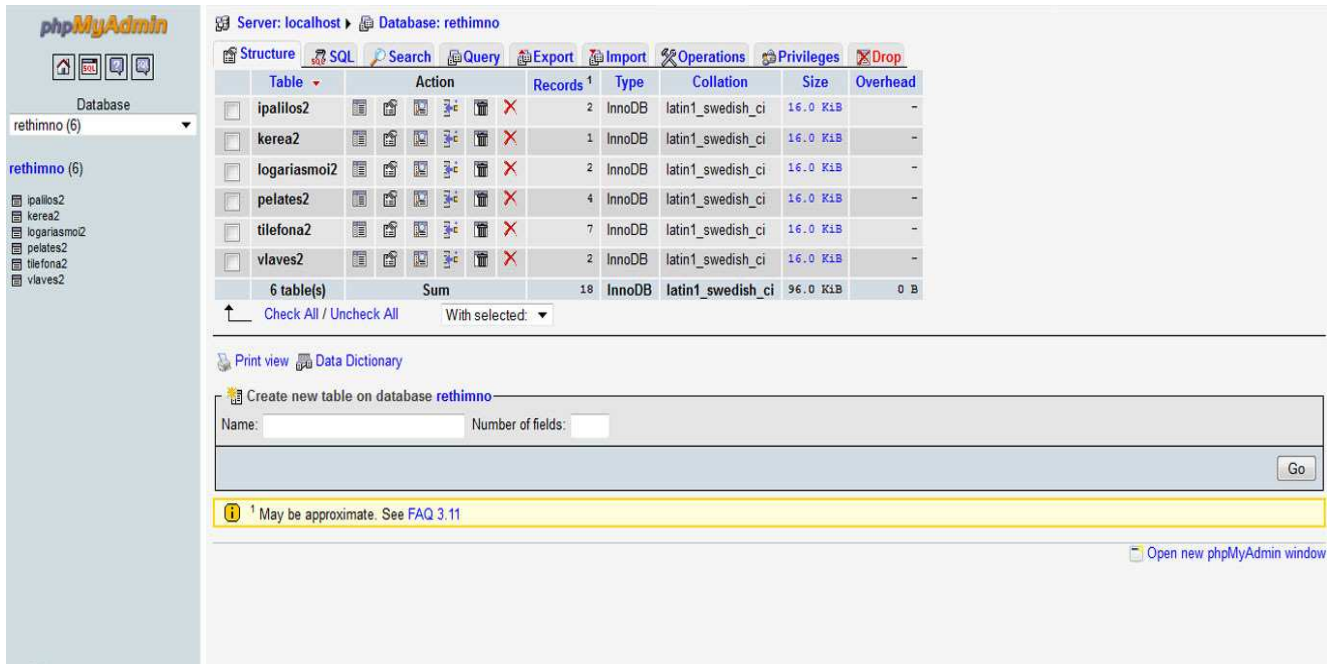
```
CREATE TABLE `rethimno`.`tilefona2` (  
  `phone_number` VARCHAR( 10 ) NULL ,  
  `puk1` VARCHAR( 10 ) NULL ,  
  `puk2` VARCHAR( 10 ) NULL ,  
  `paketa` VARCHAR( 20 ) NULL ,  
  PRIMARY KEY ( `phone_number` )
```




```
) ENGINE = InnoDB;
```

❖ Για κεραία:

```
CREATE TABLE `rethimno`.`kerea2` (  
  `kerea_name` VARCHAR( 20 ) NULL ,  
  `kerea_type` VARCHAR( 20 ) NULL ,  
  `kerea_location` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `kerea_name` )  
  ) ENGINE = InnoDB;
```



Εικόνα 15: Η βάση του Ρεθύμνου με τους πίνακες που περιέχει στο εσωτερικό της.

6.2.4 Εξαγωγή του κώδικα SQL της βάσης της Λασιθίου

❖ Για βλάβες:

```
CREATE TABLE `lasithi`.`vlaves3` (  
  `when` DATE NULL ,  
  `kerea_name1` VARCHAR( 20 ) NULL ,  
  `ergates` VARCHAR( 60 ) NULL ,  
  PRIMARY KEY ( `kerea_name1` )  
  ) ENGINE = InnoDB;
```

❖ Για λογαριασμούς:

```
CREATE TABLE `lasithi`.`logariasmoi3` (  
  `poso_pliromis` VARCHAR( 7 ) NOT NULL ,  
  `date_ekdosis` DATE NOT NULL ,  
  `date_pliromis` DATE NOT NULL ,
```



```
`pelatis_name` VARCHAR( 60 ) NOT NULL ,  
`phone_number` VARCHAR( 10 ) NOT NULL ,  
`ADD PRIMARY KEY ( `phone_number` )  
ENGINE = InnoDB;
```

❖ Για υπαλλήλους:

```
CREATE TABLE `lasithi`.`ipalilos3` (  
`ipalilos_name` VARCHAR( 60 ) NULL ,  
`id_ipalilos` VARCHAR( 10 ) NULL ,  
`misthos` VARCHAR( 7 ) NULL ,  
`name_prostateuomeno_melos` VARCHAR( 60 ) NULL ,  
`ilikia` VARCHAR( 2 ) NULL ,  
`filo` VARCHAR( 8 ) NULL ,  
`name_prostateuomeno_melos1` VARCHAR( 60 ) NULL ,  
`ilikia1` VARCHAR( 2 ) NULL ,  
`filo1` VARCHAR( 8 ) NULL ,  
`eidikotita` VARCHAR( 60 ) NULL ,  
`etoi_proetimasias` VARCHAR( 2 ) NULL ,  
`titlos_spoudon` VARCHAR( 60 ) NULL ,  
`imerominia_ptixiou` DATE NULL ,  
`panepistimio` VARCHAR( 60 ) NULL ,  
PRIMARY KEY ( `id_ipalilos` )  
ENGINE = InnoDB;
```

❖ Για πελάτες:

```
CREATE TABLE `lasithi`.`pelates3` (  
`pelatis_name` VARCHAR( 60 ) NULL ,  
`pelatis_job` VARCHAR( 50 ) NULL ,  
`pelatis_ld` VARCHAR( 10 ) NULL ,  
`pelatis_location` VARCHAR( 60 ) NULL ,  
`xronos_omilias` VARCHAR( 6 ) NULL ,  
`minimata` VARCHAR( 6 ) NULL ,  
`paketo` VARCHAR( 15 ) NULL ,  
`date_connection` DATE NULL ,  
ADD PRIMARY KEY ( `pelatis_ld` )  
ENGINE = InnoDB;
```

❖ Για τηλέφωνα:

```
CREATE TABLE `lasithi`.`telefona3` (  
`phone_number` VARCHAR( 10 ) NULL ,  
`puk1` VARCHAR( 10 ) NULL ,  
`puk2` VARCHAR( 10 ) NULL ,  
`paketa` VARCHAR( 20 ) NULL ,  
PRIMARY KEY ( `phone_number` )  
ENGINE = InnoDB;
```

❖ Για κεραία:

```
CREATE TABLE `lasithi`.`kerea3` (  

```



```
`kerea_name` VARCHAR( 20 ) NULL ,  
`kerea_type` VARCHAR( 20 ) NULL ,  
`kerea_location` VARCHAR( 60 ) NULL ,  
PRIMARY KEY ( `kerea_name` )  
) ENGINE = InnoDB;
```

Server: localhost Database: lasithi

Structure SQL Search Query Export Import Operations Privileges Drop

Table	Action	Records ¹	Type	Collation	Size	Overhead
<input type="checkbox"/> ipalilos3		3	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> kerea3		1	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> logariasmoi3		2	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> pelates3		4	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> tilefona3		3	InnoDB	latin1_swedish_ci	16.0 KiB	-
<input type="checkbox"/> vlaves3		2	InnoDB	latin1_swedish_ci	16.0 KiB	-
6 table(s)	Sum	15	InnoDB	latin1_swedish_ci	96.0 KiB	0 B

Check All / Uncheck All With selected: ▾

Print view Data Dictionary

Create new table on database lasithi

Name: Number of fields:

Go

¹ May be approximate. See FAQ 3.11

Open new phpMyAdmin window

Εικόνα 16: Η βάση του Λασιθίου με τους πίνακες που περιέχει στο εσωτερικό της.



ΚΕΦΑΛΑΙΟ 7 : Λειτουργίες σε PHP

Στα επόμενα υποκεφάλαια θα παρουσιαστούν οι βασικές λειτουργίες που εκτελώ στην δυναμική ιστοσελίδα μου. Από την δημιουργία της σελίδας μου, την επιλογή του υποκαταστήματος, την σύνδεση με την βάση δεδομένων του πληροφοριακού συστήματος. Θα δείξουμε τον κώδικα PHP αυτών των λειτουργιών και θα εξηγήσουμε τα βασικότερα του κομμάτια. Στην ανάλυση αυτή πρόσθεσα εικόνες, από την ιστοσελίδα που δημιούργησα, για κάθε εφαρμογή που θα δείχνω και θα επεξηγώ αντίστοιχα.

7.1. PHP Κώδικες του υποκαταστήματος των Χανίων

7.1.1 PHP Κώδικας για την σελίδα των Υπαλλήλων:

```
<?php
$ipalilos_name=$_POST['ipalilos_name'];
$id_ipalilos=$_POST['id_ipalilos'];
$misthos=$_POST['misthos'];
$name_prostateuomeno_melos=$_POST['name_prostateuomeno_melos'];
$ilikia=$_POST['ilikia'];
$filo=$_POST['filo'];
$name_prostateuomeno_melos1=$_POST['name_prostateuomeno_melos1'];
$ilikia1=$_POST['ilikia1'];
$filo1=$_POST['filo1'];
$eidikotita=$_POST['eidikotita'];
$eti_proipiresias=$_POST['eti_proipiresias'];
$titlos_spoudon=$_POST['titlos_spoudon'];
$imerominia_ptixiou=$_POST['imerominia_ptixiou'];
$panepistimio=$_POST['panepistimio'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("xania") or die(mysql_error());
if ($imerominia_ptixiou=="")
mysql_query("INSERT INTO `ipalilos` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', NULL , '$panepistimio')");
else
mysql_query("INSERT INTO `ipalilos` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', '$imerominia_ptixiou', '$panepistimio')");
Print "Your information has been successfully added to the database.";
?>
```

7.1.2 PHP Κώδικας για την σελίδα των Πελατών:

```
<?php
$pelatis_name=$_POST['pelatis_name'];
$pelatis_job=$_POST['pelatis_job'];
$pelatis_ld=$_POST['pelatis_ld'];
$pelatis_location=$_POST['pelatis_location'];
$xronos_omilias=$_POST['xronos_omilias'];
$minimata=$_POST['minimata'];
$paketo=$_POST['paketo'];
$date_connection=$_POST['date_connection'];
mysql_connect("localhost", "root", "") or die(mysql_error());
```



```
mysql_select_db("xania") or die(mysql_error());
mysql_query("INSERT INTO `pelatis` VALUES ('$pelatis_name', '$pelatis_job', '$pelatis_ld',
'$pelatis_location', '$xronos_omilias', '$minimata', '$paketo', '$date_connection')");
Print "Your information has been successfully added to the database. <br/>";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$date_connection1=strtotime($date_connection."+365 days");
$polipo= $date_connection1 - $date_simera ;
if ($polipo < 0){
    if ($paketo=="saloufas20")
    {
        $poso=20 *(20*0.1) + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);
    }
else if ($paketo=="saloufas45")
    {
        $poso=45 * (45 * 0.1) + (($xronos_omilias - 180) /60) * 0.0055 + (($minimata - 120) *0.1250);
    }
else if ($paketo=="saloufas70")
    {
        $poso=70 * (70 * 0.1)+ (($xronos_omilias - 360) /60) * 0.0055 + (($minimata - 200) *0.1250);
    }
else if($paketo=="saloufas100")
    {
        $poso=100 * (100 * 0.1)+ (($xronos_omilias - 580) /60) * 0.0055 + (($minimata - 360) *0.1250);
    }
else if ($paketo=="saloufas150")
    {
        $poso=150 * (150 * 0.1) + (($xronos_omilias - 1000) /60) * 0.0055 + (($minimata - 650)
*0.1250);
    }
echo "To poso me thn miosi tou pagiou einai: " . $poso;
    }else {
        if($paketo=="saloufas20")
        {
            $poso=20 + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);
        }
else if ($paketo=="saloufas45")
        {
            $poso=45 + (($xronos_omilias - 180) /60) * 0.0055 + (($minimata - 120) *0.1250);
        }
else if ($paketo=="saloufas70")
        {
            $poso=70 + (($xronos_omilias - 360) /60) * 0.0055 + (($minimata - 200) *0.1250);
        }
else if ($paketo=="saloufas100")
        {
            $poso=100 + (($xronos_omilias - 580) /60)* 0.0055 + (($minimata - 360) *0.1250);
        }
else if ($paketo=="saloufas150")
        {
            $poso=150 + (($xronos_omilias - 1000) /60) * 0.0055 + (($minimata - 650) *0.1250);
        }
echo "To poso pou ofilete einai:" . $poso;
    }
}
```




```
// an to $ipolipo>0 tote den exei perasei o xronos eno an to ipolipo<0 exei perasi o xronos//
?>
```

7.1.3 PHP Κώδικας για την σελίδα των Λογαριασμών:

```
<?php
$poso_pliromis=$_POST['poso_pliromis'];
$date_ekdosis=$_POST['date_ekdosis'];
$date_pliromis=$_POST['date_pliromis'];
$date_sindesis=$_POST['date_sindesis'];
$pelatis_name=$_POST['pelatis_name'];
$phone_number=$_POST['phone_number'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("xania") or die(mysql_error());
mysql_query("INSERT INTO `logariasmoi` VALUES ('$poso_pliromis', '$date_ekdosis',
'$date_pliromis', '$date_sindesis', '$pelatis_name', '$phone_number' )");
Print "Your information has been successfully added to the database.";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$sum=0;
$date_sindesis=strtotime($date_sindesis);
$ipolipo= $date_sindesis - $date_simera ;
if ($ipolipo < 0){
    print '<br />';
    echo "Exeis epidotisi";
    print '<br />';
    $query=mysql_query("SELECT poso_pliromis FROM logariasmoi WHERE
phone_number=6970123456");
    $query_num_rows=mysql_num_rows($query);
    $sum=0;
    $i=0;
    while ($i < $query_num_rows) {
        $row=mysql_fetch_array($query);
        $sum=$sum+$row['poso_pliromis'];
        $i++;
    }
    $poso_epidotisis= $sum *0.12;
    echo "To poso tis epidotisis tha einai:". $poso_epidotisis;
}else {
    echo "Den dikaiouste epidotisi giati den exete simplirosei 1 xrono apo thn imerominia sindesis sas
sto diktio mas";
    print '<br />';
}
$sum1=0;
$result=mysql_query("SELECT poso_pliromis FROM logariasmoi WHERE date_ekdosis
>='2011/01/01' and date_ekdosis <= '2011/12/31'");
$result_num_rows=mysql_num_rows($result);
$i=0;
while ($i < $result_num_rows) {
    $row=mysql_fetch_array($result);
    $sum1=$sum1+$row['poso_pliromis'];
    $i++;
}
print '<br />';
```



```
echo "Οι sinolikes apodoxes tou katastimatos xanion fetos einai:". $sum1;  
?>
```

7.1.4 PHP Κώδικας για την σελίδα στις Βλάβες:

```
<?php  
$when=$_POST['when'];  
$kerea_name1=$_POST['kerea_name1'];  
$ergates=$_POST['ergates'];  
mysql_connect("localhost", "root", "") or die(mysql_error());  
mysql_select_db("xania") or die(mysql_error());  
mysql_query("INSERT INTO `vlaves` VALUES('$when', '$kerea_name1', '$ergates')");  
Print "Your information has been successfully added to the database.";  
?>
```

7.1.5 PHP Κώδικας για την σελίδα στις Κεραίες:

```
<?php  
$kerea_name=$_POST['kerea_name'];  
$kerea_type=$_POST['kerea_type'];  
$kerea_location=$_POST['kerea_location'];  
mysql_connect("localhost", "root", "") or die(mysql_error());  
mysql_select_db("xania") or die(mysql_error());  
mysql_query("INSERT INTO `kerea` (kerea_name, kerea_type, kerea_location) VALUES  
('$kerea_name', '$kerea_type', '$kerea_location')");  
Print "Your information has been successfully added to the database.";  
?>
```

7.1.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων:

```
<?php  
$phone_number=$_POST['phone_number'];  
$puk1=$_POST['puk1'];  
$puk2=$_POST['puk2'];  
$paketa=$_POST['paketa'];  
mysql_connect("localhost", "root", "") or die(mysql_error());  
mysql_select_db("xania") or die(mysql_error());  
mysql_query("INSERT INTO `tilefona` (phone_number, puk1, puk2, paketa) VALUES  
('$phone_number', '$puk1', '$puk2', '$paketa')");  
Print "Your information has been successfully added to the database.";  
?>
```

7.2 PHP Κώδικες του υποκαταστήματος του Ηρακλείου

7.2.1 PHP Κώδικας για την σελίδα των Υπαλλήλων:

```
<?php  
$ipalilos_name=$_POST['ipalilos_name'];  
$id_ipalilos=$_POST['id_ipalilos'];  
$misthos=$_POST['misthos'];  
$name_prostateuomeno_melos=$_POST['name_prostateuomeno_melos'];  
$ilikia=$_POST['ilikia'];  
$filo=$_POST['filo'];  
$name_prostateuomeno_melos1=$_POST['name_prostateuomeno_melos1'];
```



```
$ilikia1=$_POST['ilikia1'];
$filo1=$_POST['filo1'];
$eidikotita=$_POST['eidikotita'];
$eti_proipiresias=$_POST['eti_proipiresias'];
$titlos_spoudon=$_POST['titlos_spoudon'];
$imerominia_ptixiou=$_POST['imerominia_ptixiou'];
$panepistimio=$_POST['panepistimio'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("iraklio") or die(mysql_error());
if ($imerominia_ptixiou=="")
mysql_query("INSERT INTO `ipalilos1` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', NULL , '$panepistimio')");
    else
mysql_query("INSERT INTO `ipalilos1` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', '$imerominia_ptixiou', '$panepistimio')");
Print "Your information has been successfully added to the database.";
?>
```

7.2.2 PHP Κώδικας για την σελίδα των Πελατών:

```
<?php
$pelatis_name=$_POST['pelatis_name'];
$pelatis_job=$_POST['pelatis_job'];
$pelatis_ld=$_POST['pelatis_ld'];
$pelatis_location=$_POST['pelatis_location'];
$xronos_omilias=$_POST['xronos_omilias'];
$minimata=$_POST['minimata'];
$paketo=$_POST['paketo'];
$date_connection=$_POST['date_connection'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("iraklio") or die(mysql_error());
mysql_query("INSERT INTO `pelatis1` VALUES ('$pelatis_name', '$pelatis_job', '$pelatis_ld',
'$pelatis_location', '$xronos_omilias', '$minimata', '$paketo', '$date_connection')");
Print "Your information has been successfully added to the database. <br/>";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$date_connection1=strtotime($date_connection."+365 days");
$ipolipo= $date_connection1 - $date_simera ;

if ($ipolipo < 0){
    if ($paketo=="saloufas20")
    {
        $poso=20 *(20*0.1) + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);
    }
else if ($paketo=="saloufas45")
    {
        $poso=45 * (45 * 0.1) + (($xronos_omilias - 180) /60) * 0.0055 + (($minimata - 120) *0.1250);
    }
else if ($paketo=="saloufas70")
    {
        $poso=70 * (70 * 0.1)+ (($xronos_omilias - 360) /60) * 0.0055 + (($minimata - 200) *0.1250);
    }
}
```



```
else if($paketo=="saloufas100")
{
    $poso=100 * (100 * 0.1)+ (($xronos_omilias - 580) /60) * 0.0055 + (($sminimata - 360) *0.1250);
}
else if ($paketo=="saloufas150")
{
    $poso=150 * (150 * 0.1) + (($xronos_omilias - 1000) /60) * 0.0055 + (($sminimata - 650)
*0.1250);
}
echo "To poso me thn miosi tou pagiou einai: " . $poso;
    }else {
        if($paketo=="saloufas20")
        {
            $poso=20 + (($xronos_omilias - 30) /60) * 0.0055 + (($sminimata - 100) *0.1250);
        }
else if ($paketo=="saloufas45")
{
    $poso=45 + (($xronos_omilias - 180) /60) * 0.0055 + (($sminimata - 120) *0.1250);
}
else if ($paketo=="saloufas70")
{
    $poso=70 + (($xronos_omilias - 360) /60) * 0.0055 + (($sminimata - 200) *0.1250);
}
else if ($paketo=="saloufas100")
{
    $poso=100 + (($xronos_omilias - 580) /60)* 0.0055 + (($sminimata - 360) *0.1250);
}
else if ($paketo=="saloufas150")
{
    $poso=150 + (($xronos_omilias - 1000) /60) * 0.0055 + (($sminimata - 650) *0.1250);
}
        echo "To poso pou ofilete einai:" . $poso;
    }
}
// an to $ipolipo>0 tote den exei perasei o xronos eno an to ipolipo<0 exei perasi o xronos
?>
```

7.2.3 PHP Κώδικας για την σελίδα των Λογαριασμών:

```
<?php
$poso_pliromis=$_POST['poso_pliromis'];
$date_ekdosis=$_POST['date_ekdosis'];
$date_pliromis=$_POST['date_pliromis'];
$date_sindesis=$_POST['date_sindesis'];
$pelatis_name=$_POST['pelatis_name'];
$phone_number=$_POST['phone_number'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("iraklio") or die(mysql_error());
mysql_query("INSERT INTO `logariasmoi1` VALUES ('$poso_pliromis', '$date_ekdosis',
'$date_pliromis', '$date_sindesis', '$pelatis_name', '$phone_number' )");
Print "Your information has been successfully added to the database.";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$sum=0;
```



```
$date_sindesis=strtotime($date_sindesis);
$ipolipo= $date_sindesis - $date_simera ;
if ($ipolipo < 0){
    print '<br />';
    echo "Exeis epidotisi";
    print '<br />';
    $query=mysql_query("SELECT poso_pliromis FROM logariasmoi1 WHERE
phone_number=6970123457");
    $query_num_rows=mysql_num_rows($query);
    $sum=0;
    $i=0;
    while ($i < $query_num_rows) {
        $row=mysql_fetch_array($query);
        $sum=$sum+$row['poso_pliromis'];
        $i++;
    }
    $poso_epidotisis= $sum *0.12;
    echo "To poso tis epidotisis tha einai:". $poso_epidotisis;
} else {
    echo "Den dikaiouste epidotisi giati den exete simplirosei 1 xrono apo thn imerominia
sindesis sas sto diktio mas";
    print '<br />';
}
$sum1=0;
$result=mysql_query("SELECT poso_pliromis FROM logariasmoi1 WHERE date_ekdosis
>='2011/01/01' and date_ekdosis <= '2011/12/31'");
$result_num_rows=mysql_num_rows($result);
$i=0;
while ($i < $result_num_rows) {
    $row=mysql_fetch_array($result);
    $sum1=$sum1+$row['poso_pliromis'];
    $i++;
}
    print '<br />';
echo "Oi sinolikes apodoxes tou katastimatos xanion fetos einai:". $sum1;
?>
```

7.2.4 PHP Κώδικας για την σελίδα στις Βλάβες:

```
<?php
$when=$_POST['when'];
$kerea_name1=$_POST['kerea_name1'];
$ergates=$_POST['ergates'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("iraklio") or die(mysql_error());
mysql_query("INSERT INTO `vlaves1` VALUES('$when', '$kerea_name1', '$ergates')");
Print "Your information has been successfully added to the database.";
?>
```



7.2.5 PHP Κώδικας για την σελίδα στις Κεραίες:

```
<?php
$kerea_name=$_POST['kerea_name'];
$kerea_type=$_POST['kerea_type'];
$kerea_location=$_POST['kerea_location'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("iraklio") or die(mysql_error());
mysql_query("INSERT INTO `kerea1` (kerea_name, kerea_type, kerea_location) VALUES
('$kerea_name', '$kerea_type', '$kerea_location')");
Print "Your information has been successfully added to the database.";
?>
```

7.2.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων:

```
<?php
$phone_number=$_POST['phone_number'];
$puk1=$_POST['puk1'];
$puk2=$_POST['puk2'];
$paketa=$_POST['paketa'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("iraklio") or die(mysql_error());
mysql_query("INSERT INTO `telefon1` (phone_number, puk1, puk2, paketa) VALUES
('$phone_number', '$puk1', '$puk2', '$paketa')");
Print "Your information has been successfully added to the database.";
?>
```

7.3 PHP Κώδικες του υποκαταστήματος του Λασιθίου

7.3.1 PHP Κώδικας για την σελίδα των Υπαλλήλων:

```
<?php
$ipalilos_name=$_POST['ipalilos_name'];
$id_ipalilos=$_POST['id_ipalilos'];
$misthos=$_POST['misthos'];
$name_prostateuomeno_melos=$_POST['name_prostateuomeno_melos'];
$ilikia=$_POST['ilikia'];
$filo=$_POST['filo'];
$name_prostateuomeno_melos1=$_POST['name_prostateuomeno_melos1'];
$ilikia1=$_POST['ilikia1'];
$filo1=$_POST['filo1'];
$eidikotita=$_POST['eidikotita'];
$eti_proipiresias=$_POST['eti_proipiresias'];
$titlos_spoudon=$_POST['titlos_spoudon'];
$imerominia_ptixiou=$_POST['imerominia_ptixiou'];
$panepistimio=$_POST['panepistimio'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("lasithi") or die(mysql_error());
if ($imerominia_ptixiou=="")
mysql_query("INSERT INTO `ipalilos3` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', NULL , '$panepistimio')");
else
```




```
mysql_query("INSERT INTO `ipalilos3` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos', '$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1', '$eidikotita', '$eti_proipiresias', '$titlos_spoudon', '$imerominia_ptixiou', '$panepistimio)");  
Print "Your information has been successfully added to the database."  
?>
```

7.3.2 PHP Κώδικας για την σελίδα των Πελατών:

```
<?php  
$pelatis_name=$_POST['pelatis_name'];  
$pelatis_job=$_POST['pelatis_job'];  
$pelatis_id=$_POST['pelatis_id'];  
$pelatis_location=$_POST['pelatis_location'];  
$xronos_omilias=$_POST['xronos_omilias'];  
$minimata=$_POST['minimata'];  
$paketo=$_POST['paketo'];  
$date_connection=$_POST['date_connection'];  
mysql_connect("localhost", "root", "") or die(mysql_error());  
mysql_select_db("lasithi") or die(mysql_error());  
mysql_query("INSERT INTO `pelatis3` VALUES ('$pelatis_name', '$pelatis_job', '$pelatis_id', '$pelatis_location', '$xronos_omilias', '$minimata', '$paketo', '$date_connection)");  
Print "Your information has been successfully added to the database. <br/>";  
$date_simera = date("Y-m-d");  
$date_simera=strtotime($date_simera);  
$date_connection1=strtotime($date_connection."+365 days");  
$ipolipo= $date_connection1 - $date_simera ;  
if ($ipolipo < 0){  
    if ($paketo=="saloufas20")  
    {  
        $poso=20 *(20*0.1) + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);  
    }  
else if ($paketo=="saloufas45")  
    {  
        $poso=45 * (45 * 0.1) + (($xronos_omilias - 180) /60) * 0.0055 + (($minimata - 120) *0.1250);  
    }  
else if ($paketo=="saloufas70")  
    {  
        $poso=70 * (70 * 0.1)+ (($xronos_omilias - 360) /60) * 0.0055 + (($minimata - 200) *0.1250);  
    }  
else if($paketo=="saloufas100")  
    {  
        $poso=100 * (100 * 0.1)+ (($xronos_omilias - 580) /60) * 0.0055 + (($minimata - 360) *0.1250);  
    }  
else if ($paketo=="saloufas150")  
    {  
        $poso=150 * (150 * 0.1) + (($xronos_omilias - 1000) /60) * 0.0055 + (($minimata - 650) *0.1250);  
    }  
echo "To poso me thn miosi tou pagiou einai: " . $poso;  
    }else {  
        if($paketo=="saloufas20")  
        {  
            $poso=20 + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);  
        }  
else if ($paketo=="saloufas45")
```



```
{
    $poso=45 + (($xronos_omilias - 180) /60) * 0.0055 + (($sminimata - 120) *0.1250);
}
else if ($paketo=="saloufas70")
{
    $poso=70 + (($xronos_omilias - 360) /60) * 0.0055 + (($sminimata - 200) *0.1250);
}
else if ($paketo=="saloufas100")
{
    $poso=100 + (($xronos_omilias - 580) /60)* 0.0055 + (($sminimata - 360) *0.1250);
}
else if ($paketo=="saloufas150")
{
    $poso=150 + (($xronos_omilias - 1000) /60) * 0.0055 + (($sminimata - 650) *0.1250);
}
echo "To poso pou ofilete einai:" . $poso;
}
// an to $ipolipo>0 tote den exei perasei o xronos eno an to ipolipo<0 exei perasi o xronos//

?>
```

7.3.3 PHP Κώδικας για την σελίδα των Λογαριασμών:

```
<?php
$poso_pliromis=$_POST['poso_pliromis'];
$date_ekdosis=$_POST['date_ekdosis'];
$date_pliromis=$_POST['date_pliromis'];
$date_sindesis=$_POST['date_sindesis'];
$pelatis_name=$_POST['pelatis_name'];
$phone_number=$_POST['phone_number'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("lasithi") or die(mysql_error());
mysql_query("INSERT INTO `logariasmoi3` VALUES ('$poso_pliromis', '$date_ekdosis',
'$date_pliromis', '$date_sindesis', '$pelatis_name', '$phone_number' )");
Print "Your information has been successfully added to the database.";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$sum=0;
$date_sindesis=strtotime($date_sindesis);
$ipolipo= $date_sindesis - $date_simera ;
if ($ipolipo < 0){
    print '<br />';
    echo "Exeis epidotisi";
    print '<br />';
    $query=mysql_query("SELECT poso_pliromis FROM logariasmoi3 WHERE
phone_number=6970123452");
    $query_num_rows=mysql_num_rows($query);
$sum=0;
    $i=0;
    while ($i < $query_num_rows) {
        $row=mysql_fetch_array($query);
        $sum=$sum+$row['poso_pliromis'];
        $i++;
    }
    $poso_epidotisis= $sum *0.12;
    echo "To poso tis epidotisis tha einai:". $poso_epidotisis;
```



```
}else {
    echo "Den dikaiouste epidotisi giati den exete simplirosei 1 xrono apo thn imerominia sindesis sas
sto diktio mas";
    print '<br />';
}
$sum1=0;
$result=mysql_query("SELECT poso_pliromis FROM logariasmoi3 WHERE date_ekdosis
>='2011/01/01' and date_ekdosis <= '2011/12/31'");
$result_num_rows=mysql_num_rows($result);
$i=0;
while ($i < $result_num_rows) {
    $row=mysql_fetch_array($result);
    $sum1=$sum1+$row['poso_pliromis'];
    $i++;
}
    print '<br />';
echo "Oi sinolikes apodoxes tou katastimatos xanion fetos einai:". $sum1;
?>
```

7.3.4 PHP Κώδικας για την σελίδα στις Βλάβες:

```
<?php
$when=$_POST['when'];
$kerea_name1=$_POST['kerea_name1'];
$ergates=$_POST['ergates'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("lasithi") or die(mysql_error());
mysql_query("INSERT INTO `vlaves3` VALUES('$when', '$kerea_name1', '$ergates')");
Print "Your information has been successfully added to the database.";
?>
```

7.3.5 PHP Κώδικας για την σελίδα στις Κεραίες:

```
<?php
$when=$_POST['when'];
$kerea_name1=$_POST['kerea_name1'];
$ergates=$_POST['ergates'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("lasithi") or die(mysql_error());
mysql_query("INSERT INTO `vlaves3` VALUES('$when', '$kerea_name1', '$ergates')");
Print "Your information has been successfully added to the database.";
?>
```

7.3.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων:

```
<?php
$phone_number=$_POST['phone_number'];
$puk1=$_POST['puk1'];
$puk2=$_POST['puk2'];
$paketa=$_POST['paketa'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("lasithi") or die(mysql_error());
mysql_query("INSERT INTO `tilefona3` (phone_number, puk1, puk2, paketa) VALUES
('$phone_number', '$puk1', '$puk2', '$paketa')");
```



```
Print "Your information has been successfully added to the database.";
?>
```

7.4 PHP Κώδικες του υποκαταστήματος του Ρεθύμνου

7.4.1 PHP Κώδικας για την σελίδα των Υπαλλήλων:

```
<?php
$ipalilos_name=$_POST['ipalilos_name'];
$id_ipalilos=$_POST['id_ipalilos'];
$misthos=$_POST['misthos'];
$name_prostateuomeno_melos=$_POST['name_prostateuomeno_melos'];
$ilikia=$_POST['ilikia'];
$filo=$_POST['filo'];
$name_prostateuomeno_melos1=$_POST['name_prostateuomeno_melos1'];
$ilikia1=$_POST['ilikia1'];
$filo1=$_POST['filo1'];
$eidikotita=$_POST['eidikotita'];
$eti_proipiresias=$_POST['eti_proipiresias'];
$titlos_spoudon=$_POST['titlos_spoudon'];
$imerominia_ptixiou=$_POST['imerominia_ptixiou'];
$panepistimio=$_POST['panepistimio'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("rethimno") or die(mysql_error());
if ($imerominia_ptixiou=="")
mysql_query("INSERT INTO `ipalilos2` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', NULL , '$panepistimio')");
else
mysql_query("INSERT INTO `ipalilos2` VALUES ('$ipalilos_name', '$id_ipalilos', '$misthos',
'$name_prostateuomeno_melos', '$ilikia', '$filo', '$name_prostateuomeno_melos1', '$ilikia1', '$filo1',
'$eidikotita', '$eti_proipiresias', '$titlos_spoudon', '$imerominia_ptixiou', '$panepistimio')");
Print "Your information has been successfully added to the database.";
?>
```

7.4.2 PHP Κώδικας για την σελίδα των Πελατών:

```
<?php
$pelatis_name=$_POST['pelatis_name'];
$pelatis_job=$_POST['pelatis_job'];
$pelatis_ld=$_POST['pelatis_ld'];
$pelatis_location=$_POST['pelatis_location'];
$xronos_omilias=$_POST['xronos_omilias'];
$minimata=$_POST['minimata'];
$paketo=$_POST['paketo'];
$date_connection=$_POST['date_connection'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("rethimno") or die(mysql_error());
mysql_query("INSERT INTO `pelatis2` VALUES ('$pelatis_name', '$pelatis_job', '$pelatis_ld',
'$pelatis_location', '$xronos_omilias', '$minimata', '$paketo', '$date_connection')");
Print "Your information has been successfully added to the database. <br/>";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$date_connection1=strtotime($date_connection."+365 days");
```



```
$ipolipo= $date_connection1 - $date_simera ;
if ($ipolipo < 0){
    if ($paketo=="saloufas20")
    {
        $poso=20 *(20*0.1) + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);
    }
else if ($paketo=="saloufas45")
    {
        $poso=45 * (45 * 0.1) + (($xronos_omilias - 180) /60) * 0.0055 + (($minimata - 120) *0.1250);
    }
else if ($paketo=="saloufas70")
    {
        $poso=70 * (70 * 0.1)+ (($xronos_omilias - 360) /60) * 0.0055 + (($minimata - 200) *0.1250);
    }
else if($paketo=="saloufas100")
    {
        $poso=100 * (100 * 0.1)+ (($xronos_omilias - 580) /60) * 0.0055 + (($minimata - 360) *0.1250);
    }
else if ($paketo=="saloufas150")
    {
        $poso=150 * (150 * 0.1) + (($xronos_omilias - 1000) /60) * 0.0055 + (($minimata - 650)
*0.1250);
    }
echo "To poso me thn miosi tou pagiou einai: " . $poso;
    }else {
        if($paketo=="saloufas20")
        {
            $poso=20 + (($xronos_omilias - 30) /60) * 0.0055 + (($minimata - 100) *0.1250);
        }
else if ($paketo=="saloufas45")
        {
            $poso=45 + (($xronos_omilias - 180) /60) * 0.0055 + (($minimata - 120) *0.1250);
        }
else if ($paketo=="saloufas70")
        {
            $poso=70 + (($xronos_omilias - 360) /60) * 0.0055 + (($minimata - 200) *0.1250);
        }
else if ($paketo=="saloufas100")
        {
            $poso=100 + (($xronos_omilias - 580) /60)* 0.0055 + (($minimata - 360) *0.1250);
        }
else if ($paketo=="saloufas150")
        {
            $poso=150 + (($xronos_omilias - 1000) /60) * 0.0055 + (($minimata - 650) *0.1250);
        }
echo "To poso pou ofilete einai:" . $poso;
    }
// an to $ipolipo>0 tote den exei perasei o xronos eno an to ipolipo<0 exei perasi o xronos//
?>
```

7.4.3 PHP Κώδικας για την σελίδα των Λογαριασμών:

```
<?php
$poso_pliromis=$_POST['poso_pliromis'];
$date_ekdosis=$_POST['date_ekdosis'];
```



```
$date_pliromis=$_POST['date_pliromis'];
$date_sindesis=$_POST['date_sindesis'];
$pelatis_name=$_POST['pelatis_name'];
$phone_number=$_POST['phone_number'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("rethimno") or die(mysql_error());
mysql_query("INSERT INTO `logariasmoi2` VALUES ('$poso_pliromis', '$date_ekdosis',
'$date_pliromis', '$date_sindesis', '$pelatis_name', '$phone_number' )");
Print "Your information has been successfully added to the database.";
$date_simera = date("Y-m-d");
$date_simera=strtotime($date_simera);
$sum=0;
$date_sindesis=strtotime($date_sindesis);
$ipolipo= $date_sindesis - $date_simera ;
if ($ipolipo < 0){
    print '<br />';
    echo "Exeis epidotisi";
    print '<br />';
    $query=mysql_query("SELECT poso_pliromis FROM logariasmoi2 WHERE
phone_number=6970123458");
    $query_num_rows=mysql_num_rows($query);
$sum=0;
    $i=0;
    while ($i < $query_num_rows) {
        $row=mysql_fetch_array($query);
        $sum=$sum+$row['poso_pliromis'];
        $i++;
    }
    $poso_epidotisis= $sum *0.12;
    echo "To poso tis epidotisis tha einai:". $poso_epidotisis;
} else {
    echo "Den dikaiouste epidotisi giati den exete simplirosei 1 xrono apo thn imerominia sindesis sas
sto diktio mas";
    print '<br />';
}
$sum1=0;
$result=mysql_query("SELECT poso_pliromis FROM logariasmoi2 WHERE date_ekdosis
>='2011/01/01' and date_ekdosis <= '2011/12/31'");
$result_num_rows=mysql_num_rows($result);
    $i=0;
    while ($i < $result_num_rows) {
        $row=mysql_fetch_array($result);
        $sum1=$sum1+$row['poso_pliromis'];
        $i++;
    }
    print '<br />';
echo "Oi sinolikes apodoxes tou katastimatos xanion fetos einai:". $sum1;
?>
```

7.4.4 PHP Κώδικας για την σελίδα στις Βλάβες:

```
<?php
$when=$_POST['when'];
$kerea_name1=$_POST['kerea_name1'];
$ergates=$_POST['ergates'];
```




```
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("rethimno") or die(mysql_error());
mysql_query("INSERT INTO `vlaves2` VALUES('$when', '$kerea_name1', '$ergates')");
Print "Your information has been successfully added to the database.";
?>
```

7.4.5 PHP Κώδικας για την σελίδα στις Κεραίες:

```
<?php
$kerea_name=$_POST['kerea_name'];
$kerea_type=$_POST['kerea_type'];
$kerea_location=$_POST['kerea_location'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("rethimno") or die(mysql_error());
mysql_query("INSERT INTO `kerea2` (kerea_name, kerea_type, kerea_location) VALUES
('$kerea_name', '$kerea_type', '$kerea_location' )");
Print "Your information has been successfully added to the database.";
?>
```

7.5.6 PHP Κώδικας για την σελίδα των Αριθμών Τηλεφώνων:

```
<?php
$phone_number=$_POST['phone_number'];
$puk1=$_POST['puk1'];
$puk2=$_POST['puk2'];
$paketa=$_POST['paketa'];
mysql_connect("localhost", "root", "") or die(mysql_error());
mysql_select_db("rethimno") or die(mysql_error());
mysql_query("INSERT INTO `tilefona2` (phone_number, puk1, puk2, paketa) VALUES
('$phone_number', '$puk1', '$puk2', '$paketa')");
Print "Your information has been successfully added to the database.";
?>
```

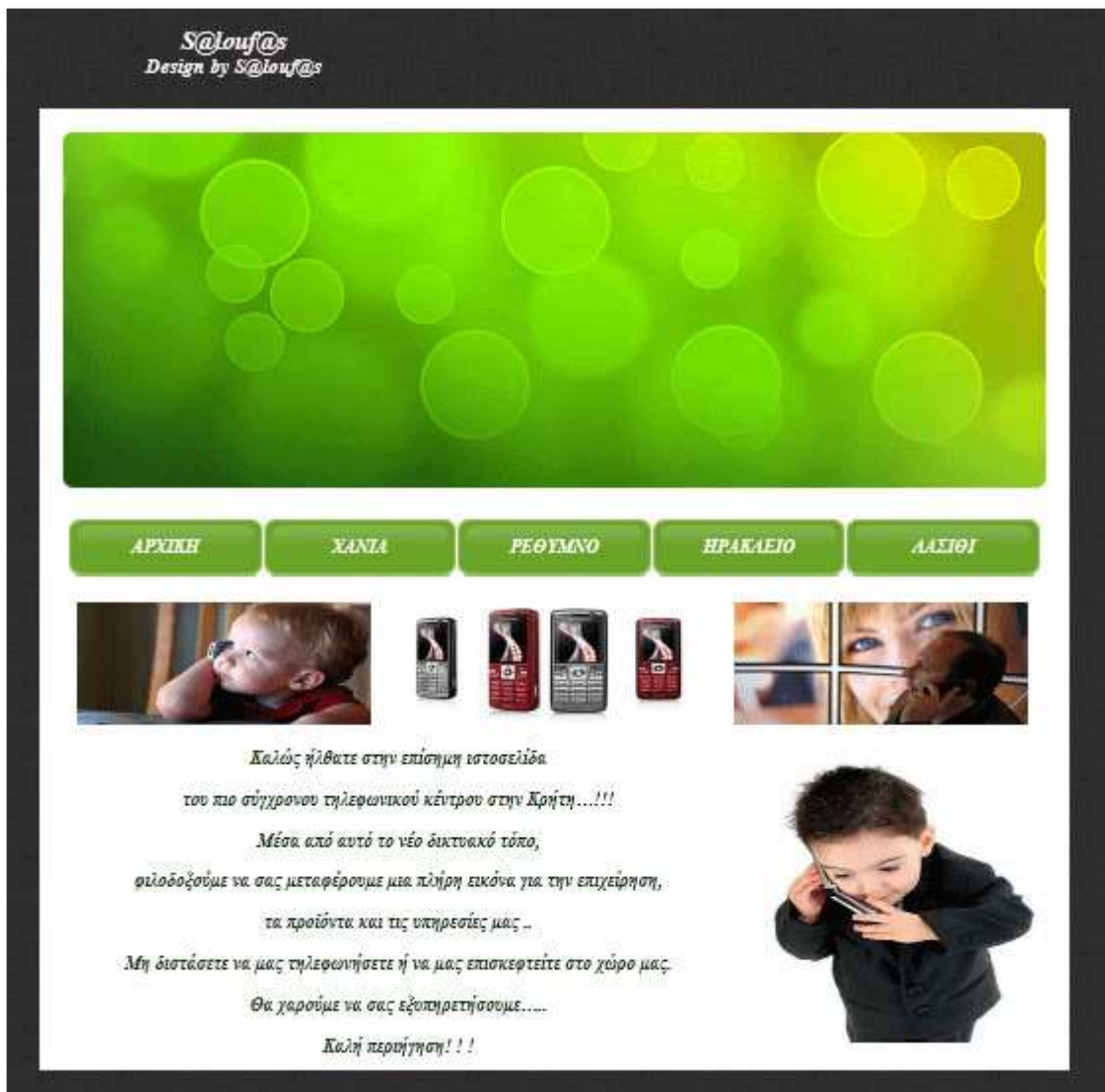



ΚΕΦΑΛΑΙΟ 8 : HTML Σελίδες

Για να δημιουργήσω την ιστοσελίδα μου έβαξα εικόνες και σχεδίασα τα κουμπιά, επίσης έφτιαξα animation την αρχική εικόνα για να δώσει ένα ιδιαίτερο στυλ στο site μου. Για να γίνει animation πας στο photoshop παίρνεις τις εικόνες που θες της βάζεις και τις 3 σε ένα νέο title ανάλογα με το μέγεθος των φωτογραφιών που θα αλλάζουν και πας windows->animation και επιλέγεις την σειρά των φωτογραφιών που θα τρέχουν και τον χρόνο που θα αλλάζει η κάθε φωτογραφία. Την αποθήκευσα με κατάληξη .gif για να κάνει το animation που θέλω. Έφτιαξα table και έβαλα τις εικόνες και τα κουμπιά μου. Η μετονομασία της επιχείρησης φτιάχτηκε σε banner και το background της σελίδας είναι μια εικόνα που επαναλαμβάνεται από πίσω από το site μου!

8.1 HTML σελίδα η αρχική του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s

Ο κώδικας της αρχικής σελίδας είναι στο παράρτημα Α.




Εικόνα 17






8.2 HTML σελίδα του κάθε υποκαταστήματος (Ηρακλείου, Χανίων, Ρεθύμνου και του Λασιθίου)

S@louf@S
Design by S@louf@S




ΑΡΧΙΚΗ ΧΑΝΙΑ ΡΕΘΥΜΝΟ ΗΡΑΚΛΕΙΟ ΛΑΣΙΘΙ



ΥΠΑΛΛΗΛΟΙ
ΠΕΛΑΤΕΣ
ΛΟΓΑΡΙΑΣΜΟΙ
ΒΛΑΒΕΣ
ΚΕΡΑΙΕΣ
ΠΑΚΕΤΑ ΣΥΝΔΕΣΗΣ
ΑΡΙΘΜΟΙ ΤΗΛΕΦΩΝΩΝ
ΕΠΙΚΟΙΝΩΝΙΑ

Καλώς ορίσατε στο υποκατάστημα της κινητής τηλεφωνίας του Ηρακλείου.
Εδώ μπορείτε να ενημερωθείτε μια πλήρη εικόνα για την διαχείριση του υποκαταστήματος, τα προϊόντα και τις υπηρεσίες μας...

Καλή περιήγηση...!!!





S@louf@S
Design by S@louf@S



[ΑΡΧΙΚΗ](#) [ΧΑΝΙΑ](#) [ΡΕΘΙΜΝΟ](#) [ΗΡΑΚΛΕΙΟ](#) [ΛΑΣΙΟΙ](#)



[ΥΠΑΛΛΗΛΟΙ](#)
[ΠΕΛΑΤΕΣ](#)
[ΛΟΓΑΡΙΑΣΜΟΙ](#)
[Β.14ΒΕΣ](#)
[ΚΕΡΑΙΕΣ](#)
[ΠΑΚΕΤΑ ΣΥΝΘΕΣΗΣ](#)
[ΑΡΙΘΜΟΙ ΤΗΛΕΦΩΝΩΝ](#)
[ΕΠΙΚΟΙΝΩΝΙΑ](#)


Καλώς ορίσατε στο υποκατάστημα της κινητής τηλεφωνίας των Χανίων. Εδώ μπορείτε να ενημερωθείτε μια πλήρη εικόνα για την διαχείριση του υποκαταστήματος, τα προϊόντα και τις υπηρεσίες μας...

Καλή περιήγηση...!!!








S@louf@S
Design by S@louf@S




[ΑΡΧΙΚΗ](#) [ΧΑΝΙΑ](#) [ΡΕΘΥΜΝΟ](#) [ΗΡΑΚΛΕΙΟ](#) [ΛΑΣΙΘΙ](#)



[ΥΠΕΛΛΗΛΟΙ](#)
[ΠΕΛΑΤΕΣ](#)
[ΛΟΓΑΡΙΑΣΜΟΙ](#)
[ΒΛΑΒΕΣ](#)
[ΚΕΡΑΙΕΣ](#)
[ΠΑΚΕΤΑ ΣΥΝΔΕΣΗΣ](#)
[ΑΡΙΘΜΟΙ ΤΗΛΕΦΩΝΩΝ](#)
[ΕΠΙΣΚΟΠΟΝΙΑ](#)

Καλώς ορίσατε στο υποκατάστημα της κινητής τηλεφωνίας του Ρεθύμνου. Εδώ μπορείτε να ενημερωθείτε μια πλήρη εικόνα για την διαχείριση του υποκαταστήματος, τα προϊόντα και τις υπηρεσίες μας...

Καλή περιήγηση...!!!





Εικόνα 18

8.2.1 HTML σελίδα για τους Υπαλλήλους

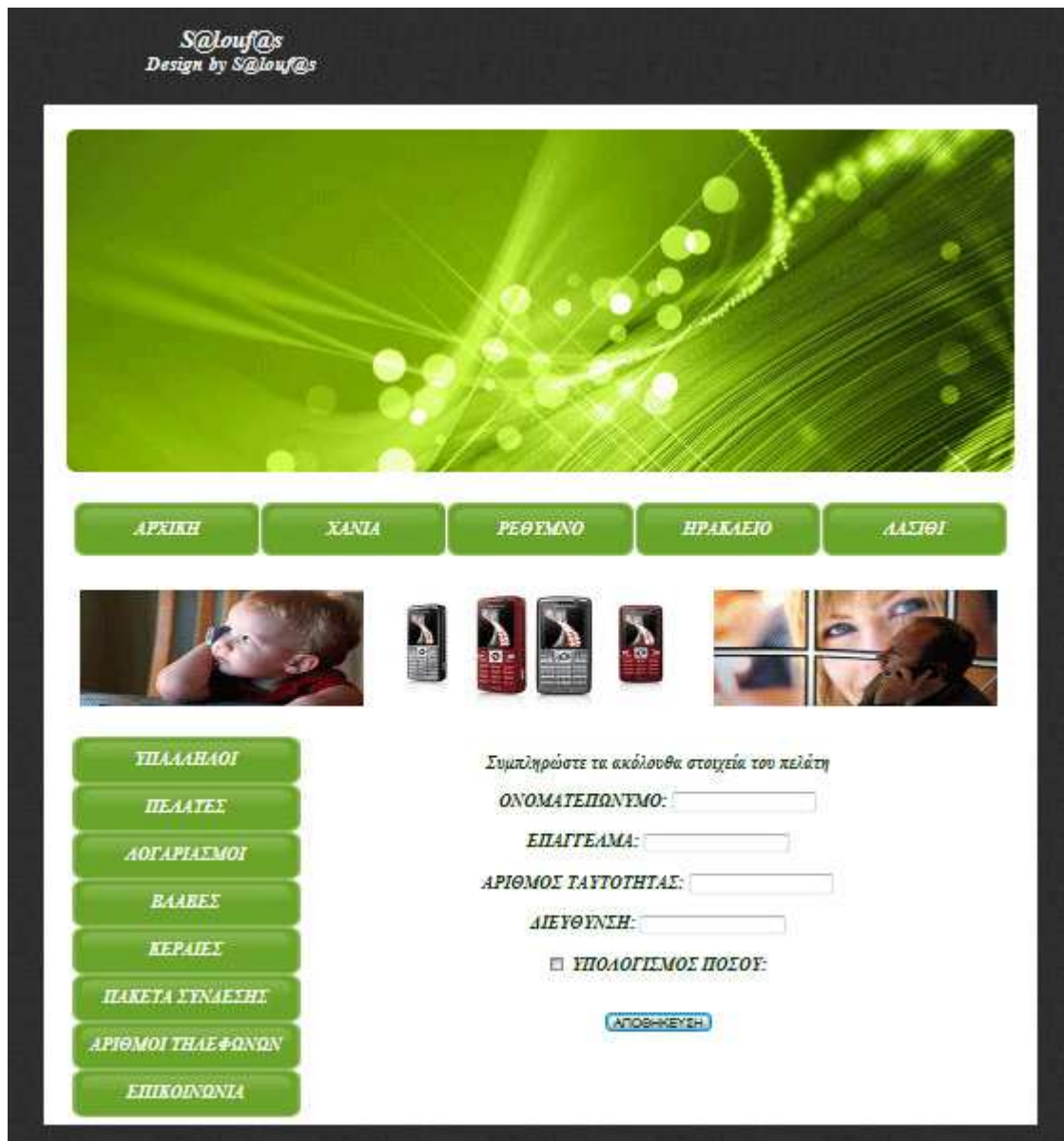
Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



Εικόνα 19

8.2.2 HTML σελίδα για τους Πελάτες

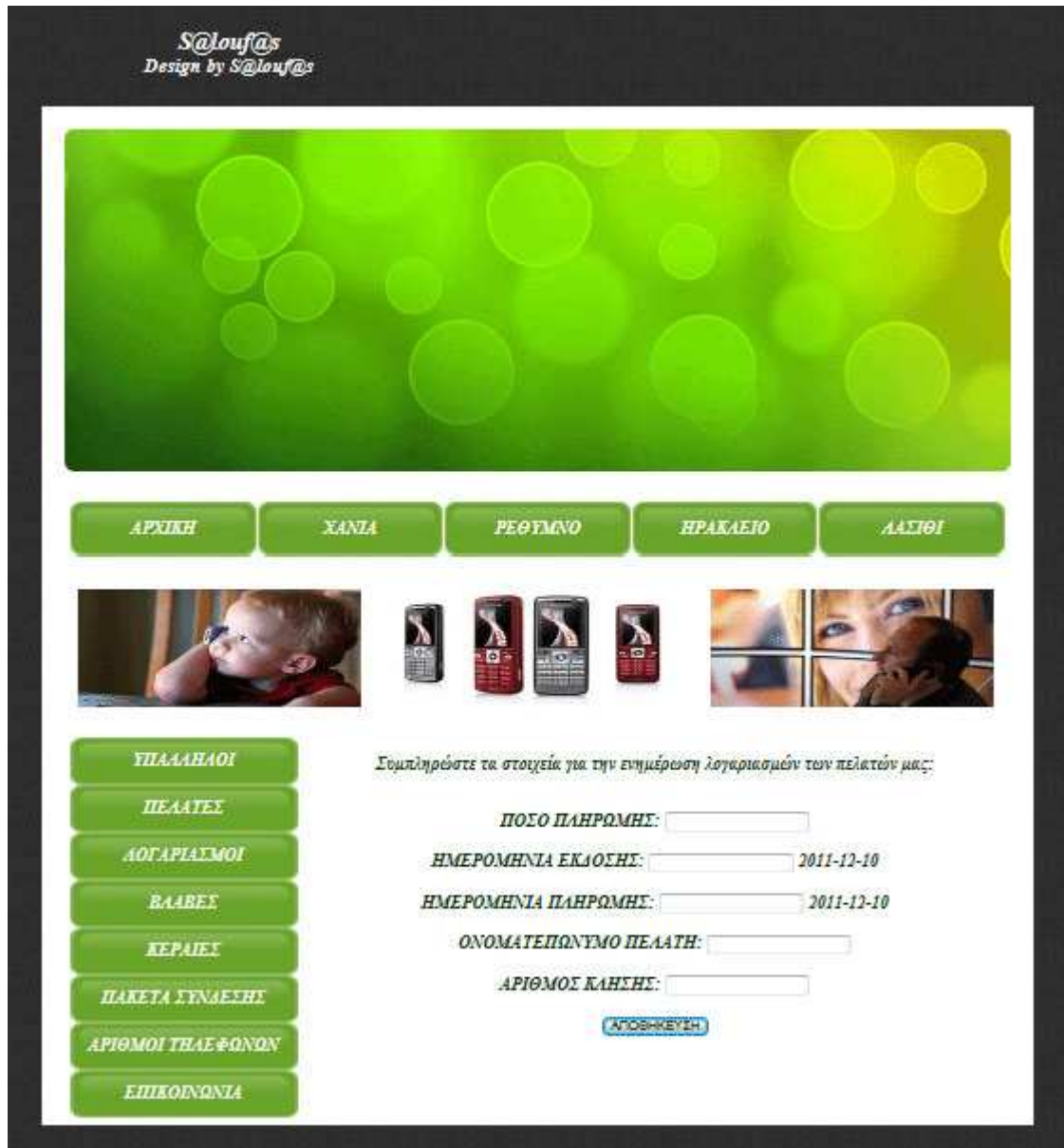
Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



Εικόνα 20

8.2.3 HTML σελίδα για τους Λογαριασμούς

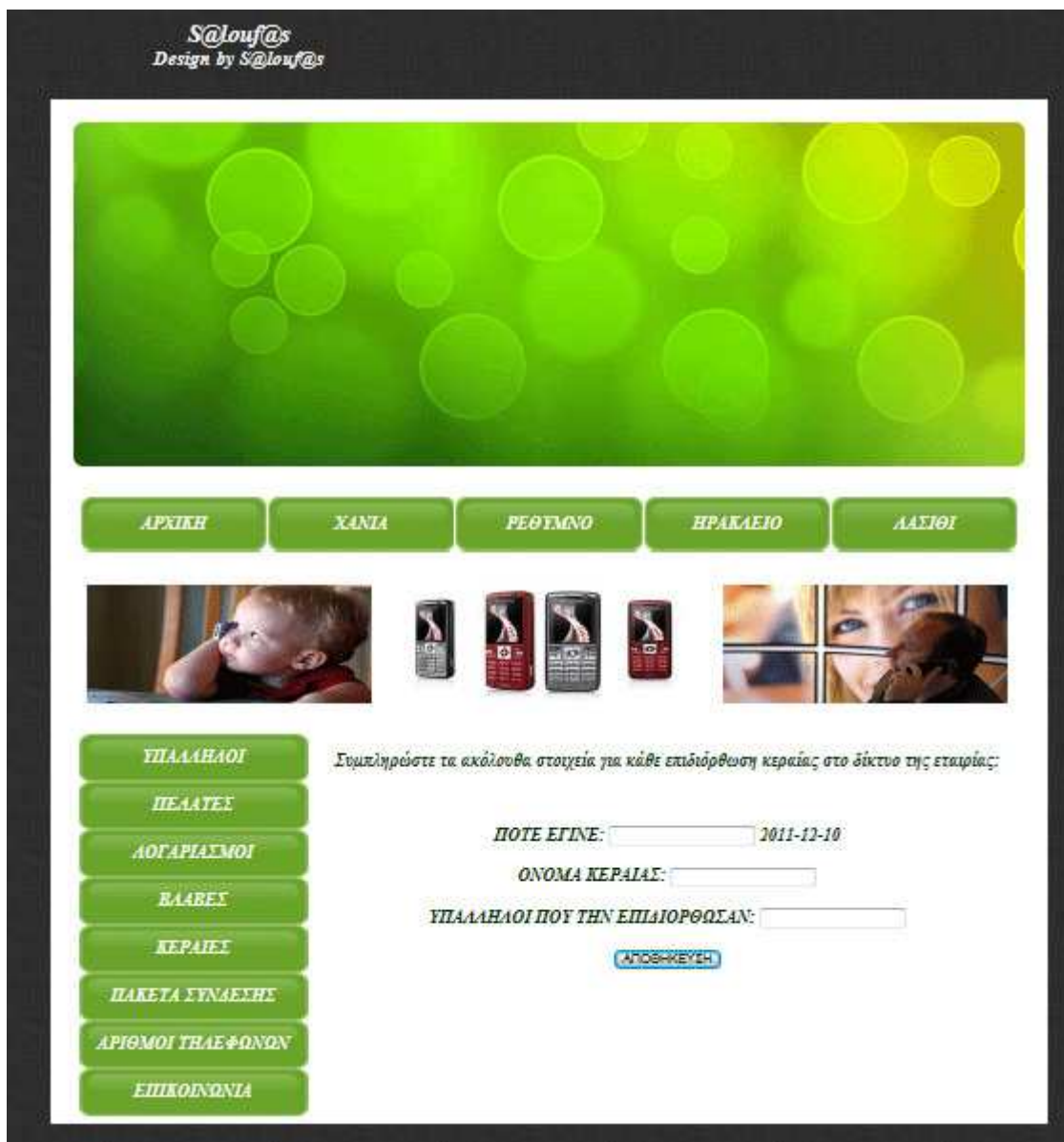
Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



Εικόνα 21

8.2.4 HTML σελίδα για τις Βλάβες

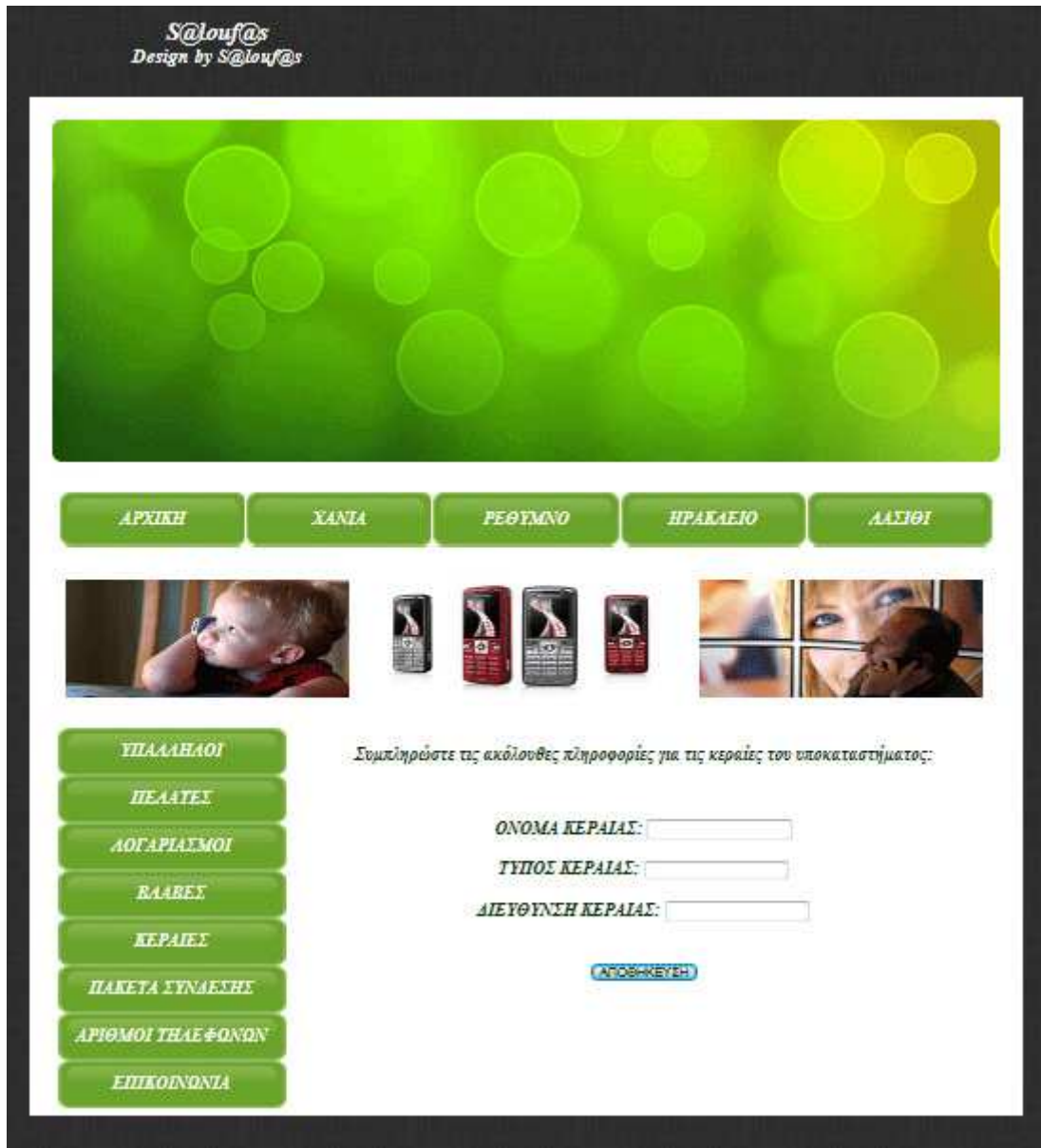
Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



Εικόνα 22

8.2.5 HTML σελίδα για τις Κεραίες

Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



Εικόνα 23

8.2.6 HTML σελίδα για τα Πακέτα σύνδεσης

Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



S@louf@S
Design by S@louf@S



ΑΡΧΙΚΗ ΣΑΝΙΑ ΡΕΘΥΜΝΟ ΗΡΑΚΛΕΙΟ ΛΑΣΙΘΙ



Ακολουθούν τα πακέτα σύνδεσης και καρτοσυμβολαίου της εταιρίας μας:

Πακέτα σύνδεσης	s@louf@s 20	s@louf@s 45	s@louf@s 70	s@louf@s 100	s@louf@s 150
Πάγιο	20 ευρώ	45 ευρώ	70 ευρώ	100 ευρώ	150 ευρώ
Δωρεάν λεπτά	30'	180'	360'	580'	1000'
Δωρεάν sms	100	120	200	360	650

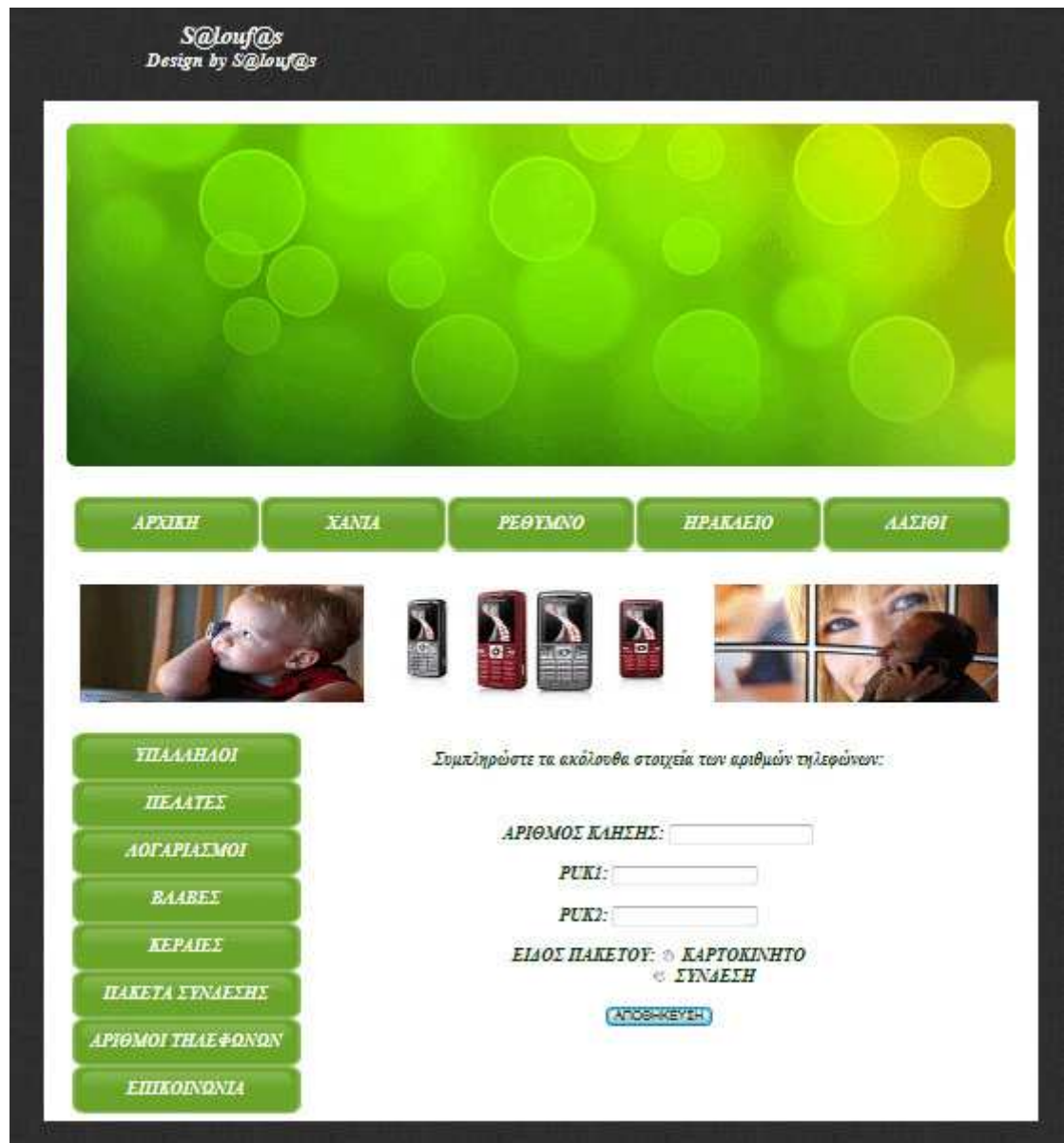
Εκτα χρέωση για εθνικές κλήσεις ομιλίας: 0,0055 ευρώ/δευτ.
Εκτα χρέωση για sms: 0,1250/sms

ΥΠΑΛΛΗΛΟΙ
ΠΕΛΑΤΕΣ
ΔΟΓΜΑΤΙΣΜΟΙ
ΒΛΑΒΕΣ
ΚΕΡΑΙΕΣ
ΠΑΚΕΤΑ ΣΥΝΔΕΣΗΣ
ΑΡΙΘΜΟΙ ΤΗΛΕΦΩΝΩΝ
ΕΠΙΚΟΙΝΩΝΙΑ

Εικόνα 24

8.2.7 HTML σελίδα για τον Αριθμό των Τηλεφώνων

Ο κώδικας της φόρμας της σελίδας μου βρίσκεται στο παράρτημα Α.



Εικόνα 25

8.3 HTML σελίδα της Επικοινωνίας του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@S

Ο κώδικας της σελίδας επικοινωνίας μου βρίσκεται στο παράρτημα Α.



Εικόνα 26



ΚΕΦΑΛΑΙΟ 9 : Επίλογος

9.1 Αποτελέσματα

Τα αποτελέσματα της πτυχιακής εργασίας μου ήταν η δημιουργία ενός ενδοεταιρικού ιστοχώρου μιας τηλεφωνικής εταιρίας. Η εν λόγω πτυχιακή, μου έδωσε την ευκαιρία να ασχοληθώ με όλα τα στάδια που απαιτούνται στην ανάπτυξη μιας διαδικαστικής εφαρμογής. Ασχολήθηκα με την ανάλυση λογισμικού, τον προγραμματισμό διαδικτύου και με τις βάσεις δεδομένων. Στο προγραμματισμό διαδικτύου μου δόθηκε η ευκαιρία να χρησιμοποιήσω την HTML καθώς και την PHP. Επίσης με την δημιουργία μιας βάσης δεδομένων ανέπτυξα όλα τα στάδια σχεδιασμού της σελίδας μου μέχρι και τη διαχείριση της με την υποβολή ερωτημάτων σε SQL.

9.2 Συμπεράσματα

Η PHP, MySql και ο Apache Server συνεργάζονται με επιτυχία και θεωρούνται τα πλέον κατάλληλα εργαλεία για την ανάπτυξη διαδικτυακών εφαρμογών με υψηλά κριτήρια αξιοπιστίας και απόδοσης. Η ανάπτυξη μιας πτυχιακής βοηθά τον σπουδαστή να εμπεδώσει θεωρητικά και τεχνικά θέματα τα οποία διδάχτηκε κατά την διάρκεια των σπουδών του, δημιουργώντας τις προϋποθέσεις να τα χρησιμοποιήσει στην μετέπειτα σταδιοδρομία του.

Επίσης αναπτύσσοντας μόνη μου την εφαρμογή αυτή βελτίωσα και ανέπτυξα τις γνώσεις μου στον διαχωρισμό διαφόρων εφαρμογών και την χρήση ξένου πηγαίου κώδικα. Συνοψίζοντας μπορώ να πω με βεβαιότητα ότι κάλυψα σε ένα καλό επίπεδο την λειτουργία της ιστοσελίδας, και την ευχρηστία της. Ακόμα η βάση δεδομένων που κατασκεύασα κανονικοποιήθηκε (έγινε σε μορφή πινάκων) με την χρήση των κανόνων κανονικοποίησης των βάσεων δεδομένων.

ΚΕΦΑΛΑΙΟ 10 : HTML Κώδικες

10.1 HTML Κώδικας της αρχικής σελίδας του καταστήματος της κινητής τηλεφωνίας με την επωνυμία S@louf@s:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Untitled Document</title>
<style type="text/css">
<!--
body {
    margin-left: 10px;
    margin-top: 10px;
    margin-right: 10px;
    margin-bottom: 10px;
    background-image: url(images/bg.jpg);
    background-repeat: repeat;
    background-color: #FFFDEA;
}
body,td,th {
    color: #FFF;
    font-size: large;
    font-style: italic;
    line-height: normal;
    font-weight: bold;
}
.logo1 {font-size: 25px}
.logo2 {font-size: 20px}
.text-green {color: #003300}
</style>
</head>
<body bgcolor="#FFFFFF" text="#006600">
<br/>
<table width="1050" border="0" align="center" cellpadding="0" cellspacing="0">
<tr>
<td><table width="468" border="0">
</table>
<table width="400" border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="center"><span class="logo1"> S@louf@s </span></td>
</tr>
<tr>
<td align="center" class="logo1"><small>Design by S@louf@s </small>
</td>
</tr>
</table></td>
</tr>
</table>
```



```
<table width="995" border="0" align="center" cellpadding="5" cellspacing="5">
  <tr>
    <th width="0" height="0" nowrap="nowrap" bgcolor="#FFFFFF" border="100">
      <table width="100%" border="0" align="center" cellpadding="10" cellspacing="10">
        <tr>
          <td bgcolor="#FFFFFF">
            </td>
        </tr>
      </table>
    </th>
  </tr>
  <tr>
    <td height="20">
      <table width="988" border="0" align="center">
        <tr>
          <th width="20%" height="55" background="images/buttons.png">ΑΡΧΙΚΗ</th>
          <th width="20%" height="55" background="images/buttons.png">ΧΑΝΙΑ</th>
          <th width="20%" height="55" background="images/buttons.png">ΠΕΘΥΜΝΟ</th>
          <th width="20%" height="55" background="images/buttons.png">ΗΠΑΚΛΕΙΟ</th>
          <th width="20%" height="55" background="images/buttons.png">ΛΑΣΙΘΙ</th>
        </tr>
      </table>
    </td>
    <td>
      <table width="1007" border="0" align="center" cellpadding="5" cellspacing="5">
        <tr>
          <td width="325"></td>
          <td width="330">
             </td>
          <td width="338">
            </td>
        </tr>
      </table>
    </td>
  </tr>
  <tr>
    <td>
      <table width="995" border="0" align="center" cellpadding="0" cellspacing="0">
        <tr>
          <td width="679" height="150" align="middle" class="text-green">
            <div align="center">Καλώς ήλθατε στην επίσημη ιστοσελίδα
              <br />
              <br />
              του πιο σύγχρονου τηλεφωνικού κέντρου στην Κρήτη...!!!
              <br />
              <br />
              Μέσα από αυτό το νέο δικτυακό τόπο,<br />
              <br />
              φιλοδοξούμε να σας μεταφέρουμε μια πλήρη εικόνα για την επιχείρηση,
              <br />
              <br />
              τα προϊόντα και τις υπηρεσίες μας ..
              <br />
              <br />
              Μη διστάσετε να μας τηλεφωνήσετε ή να μας επισκεφτείτε στο χώρο μας.
            </div>
          </td>
        </tr>
      </table>
    </td>
  </tr>
</table>
```



```
<br />
<br />
Θα χαρούμε να σας εξυπηρετήσουμε.....
<br />
<br />
Καλή περιήγηση!! </div></td>
<td width="316"></td>
</tr>
</table>
</td>
</tr>
</table>
</tr>
</table>
</body>
</html>
```

10.2 HTML Κώδικες του κάθε υποκαταστήματος

Η μονές αλλαγές που υπάρχουν στις σελίδες των υποκαταστημάτων είναι η φόρμες με τα στοιχεία που καταχωρούνται στην βάση μου. Για κάθε υποκατάστημα και ανάλογα την κατηγορία του θα διαμορφωθούν οι εξής κώδικες.

10.2.1 HTML Κώδικες του υποκαταστήματος των Χανίων

```
<td width="989" valign="top" class="text-green"><p><table width="758" border="0">
<tr>
<td width="420" align="center" valign="top"><p class="text-green">Καλώς ορίσατε στο
υποκατάστημα της κινητής <br/>τηλεφωνίας των Χανίων.<br/>
Εδώ μπορείτε να ενημερωθείτε μια πλήρη εικόνα <br/>για την διαχείριση του
υποκαταστήματος,<br/>
τα προϊόντα και τις υπηρεσίες μας...</p>
<p class="text-green">Καλή περιήγηση...!!!</p>
</p></td>
<td width="328"></td>
</tr>
</table>
</td>
```

10.2.1.1 HTML Κώδικας για την σελίδα των Υπαλλήλων:

```
<form name='frm' form action="ipalilos.php" method="post">
<p class="text-green">
ΟΝΟΜΑΤΕΠΩΝΥΜΟ:
<input type="text" name="ipalilos_name"><br> <br>
ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "id_ipalilos" > <br> <br>
ΜΙΣΘΟΣ: <input type="text" name = "misthos"><br> <br>
Συμπλήρωσε τα ακόλουθα στοιχεία αν υπάρχουν προστατευόμενα μέλη:<br />
<br />
<input name="chkbox" type="checkbox" id="name_prostateuomeno_melos"
onclick="showhidefield()" value="NAI" >
ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p>
```



```
</p>
<div id='hideablearea' style='display:none'>
  <p class="text-green">
    ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos" ><br>
    ΗΛΙΚΙΑ: <input type='text' name="ilikia"><br>
    ΦΥΛΟ:
    <select name="filo" class="styled">
      <option value="" selected="selected">DEFINE</option>
      <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
      <option value="MAN">ΑΝΤΡΑΣ</option>
    </select>
    <br/>
    <input name="chkbox3" type="checkbox" id="name_prostateuomeno_melos1"
onclick="showhidefield3()" value="NAIN" >
ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p>
</p>
<div id='hideablearea3' style='display:none'>
  <p class="text-green">
    ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos1" ><br>
    ΗΛΙΚΙΑ: <input type='text' name="ilikia1"><br>
    ΦΥΛΟ:
    <select name="filo1" class="styled">
      <option value="" selected="selected">DEFINE</option>
      <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
      <option value="MAN">ΑΝΤΡΑΣ</option>
    </select>
  </p>
</div>
</p></div>
<p>Συμπλήρωσε τα στοιχεία του υπάλληλου ανάλογα με πιο κλάδο της εταιρίας ανήκει:
<br />
<br />
  <input name="chkbox2" type="checkbox" id="texnikoi"
onclick="showhidefield2()" value="yes">
  ΤΕΧΝΙΚΟ
  <br>
</p>
</p>
<div id='hideablearea2' style='display:none'>
  <p class="text-green">
    ΕΙΔΙΚΟΤΗΤΑ:
    <input type='text' name="eidikotita"><br>
    ΕΤΗ ΠΡΟΫΠΗΡΕΣΙΑΣ: <input type='text' name="eti_proipiresias"><br />
  </p>
</div>
  <input name="chkbox1" type="checkbox" id="dioikitikoi" onclick="showhidefield1()"
value="yeap" >
  ΔΙΟΙΚΗΤΙΚΟ
  <br>
</p>
<div id='hideablearea1' style='display:none'>
  <p class="text-green">
    ΤΙΤΛΟΣ ΣΠΟΥΔΩΝ:
    <input type='text' name="titlos_spoudon"><br>
```



```
    ΗΜΕΡΟΜΗΝΙΑ ΚΤΗΣΗΣ ΠΤΥΧΙΟΥ: <input type='text2' name="imerominia_ptixiou">
2011-12-10<br>
    ΠΑΝΕΠΙΣΤΗΜΙΟ:<input type='text3' name="panepistimio"> <br>
</p>
</div>
<br /><input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">
</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα επί 4 φορές ξεχωριστά στο ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΤΕΧΝΙΚΟ και στο ΔΙΟΙΚΗΤΙΚΟ κλάδο της εταιρίας.

```
<script language="JavaScript">
function showhidefield()
{
if (document.frm.chkbox.checked)
{
document.getElementById("hideablearea").style.display = "block";
}
else
{
document.getElementById("hideablearea").style.display = "none";
}
}
</script>
```

10.2.1.2 HTML Κώδικας για την σελίδα των Πελατών:

```
<form name = 'frm' action="pelatis.php" method="post">
ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type="text" name="pelatis_name"><br> <br>
ΕΠΑΓΓΕΛΜΑ: <input type="text" name = "pelatis_job"><br> <br>
ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "pelatis_ld"><br> <br>
ΔΙΕΥΘΥΝΣΗ: <input type="text" name = "pelatis_location"><br> <br>
<input name="chkbox" type="checkbox" id="ipologismos_posou" onclick="showhidefield()"
value="tpt" >
ΥΠΟΛΟΓΙΣΜΟΣ ΠΟΣΟΥ:</p>
</p>
<div id='hideablearea' style='display:none'>
<p class="text-green">
ΣΥΝΟΛΙΚΟΣ ΧΡΟΝΟΣ ΟΜΙΛΙΑΣ: <input type='text' name="xronos_omilias" ><br><br>
ΜΗΝΥΜΑΤΑ: <input type='text' name="minimata"><br><br>
ΠΑΚΕΤΟ ΣΥΝΔΕΣΗΣ: <input type='text' name="paketo"><br><br>
ΗΜΕΡΟΜΗΝΙΑ ΣΥΝΔΕΣΗΣ: <input type='text' name="date_connection"> 2011-12-10 </p>
</div><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα για να υπολογίσω το ποσό.

```
<script language="JavaScript">
function showhidefield()
```



```
{
  if (document.frm.chkbox.checked)
  {
    document.getElementById("hideablearea").style.display = "block";
  }
  else
  {
    document.getElementById("hideablearea").style.display = "none";
  }
}
</script>
```

10.2.1.3 HTML Κώδικας για την σελίδα των Λογαριασμών:

```
<form action="logariasmoi.php" method="post">
ΠΟΣΟ ΠΛΗΡΩΜΗΣ: <input type="text" name="poso_pliromis"><br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΕΚΔΟΣΗΣ: <input type="text" name = "date_ekdosis"> 2011-12-10<br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΠΛΗΡΩΜΗΣ: <input type="text" name = "date_pliromis"> 2011-12-10<br><br>
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΠΕΛΑΤΗ: <input type="text" name = "pelatis_name"><br> <br>
ΑΡΙΘΜΟΣ ΚΛΗΣΗΣ: <input type="text" name="phone_number"><br> <br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

10.2.1.4 HTML Κώδικας για την σελίδα στις Βλάβες:

```
<form action="vlaves.php" method="post">
  <span class="text-green">ΠΟΤΕ ΕΓΙΝΕ:
<input type="text" name="when"> 2011-12-10<br> <br>
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_name1"><br> <br>
ΥΠΑΛΛΗΛΟΙ ΠΟΥ ΤΗΝ ΕΠΙΔΙΟΡΘΩΣΑΝ: <input type="text" name = "ergates">
<br><br>
  <input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
  </span>
</form>
```

10.2.1.5 HTML Κώδικας για την σελίδα στις Κεραίες:

```
<form action="kerea.php" method="post">
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name="kerea_name"><br> <br>
ΤΥΠΟΣ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_type"><br> <br>
ΔΙΕΥΘΥΝΣΗ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_location"><br> <br><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">
</form>
```

10.2.1.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης:

Έφτιαξα σε πίνακα τα πακέτα σύνδεσης αντι για φόρμα γιατί δεν χρειάζεται να προσθέσουμε στοιχεία στην βάση.

```
<table width="716" border="0">
  <tr>
    <td width="1076" align="center" valign="top"><p class="text-green">Ακολουθούν τα πακέτα
    σύνδεσης και καρτοσυμβολαίου της εταιρίας μας:</p></td>
```




```
</tr>
<tr>
  <td><table width="707" border="0">
    <tr>
      <td width="701"><table width="622" border="5" align="center" class="text-green">
        <tr>
          <td width="86" class="text-green"><p>Πακέτα σύνδεσης</p></td>
          <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
20</p></td>
          <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
45</p></td>
          <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
70</p></td>
          <td width="94" align="center" valign="middle"><p class="text-green">s@louf@s
100</p></td>
          <td width="99" align="center" valign="middle"><p class="text-green">s@louf@s
150</p></td>
        </tr>
        <tr>
          <td class="text-green"><p>Πάγιο</p></td>
          <td align="center" valign="middle" class="text-green"><p>20 euro</p></td>
          <td align="center" valign="middle" class="text-green"><p>45 euro</p></td>
          <td align="center" valign="middle" class="text-green"><p>70 euro</p></td>
          <td align="center" valign="middle" class="text-green"><p>100 euro</p></td>
          <td align="center" valign="middle" class="text-green"><p>150 euro</p></td>
        </tr>
        <tr>
          <td class="text-green"><p>Δωρεάν λεπτά</p></td>
          <td align="center" valign="middle" class="text-green"><p>30'</p></td>
          <td align="center" valign="middle" class="text-green"><p>180'</p></td>
          <td align="center" valign="middle" class="text-green"><p>360'</p></td>
          <td align="center" valign="middle" class="text-green"><p>580'</p></td>
          <td align="center" valign="middle" class="text-green"><p>1000'</p></td>
        </tr>
        <tr>
          <td class="text-green"><p>Δωρεάν sms</p></td>
          <td align="center" valign="middle" class="text-green"><p>100</p></td>
          <td align="center" valign="middle" class="text-green"><p>120</p></td>
          <td align="center" valign="middle" class="text-green"><p>200</p></td>
          <td align="center" valign="middle" class="text-green"><p>360</p></td>
          <td align="center" valign="middle" class="text-green"><p>650</p></td>
        </tr>
      </table></td>
    </tr>
    <tr>
      <td height="71" align="center" valign="middle"><p class="text-green">Extra χρέωση για
εθνικές κλήσεις ομιλίας: 0,0055 euro/δευτ.</p>
      <p class="text-green">Extra χρέωση για sms: 0,1250/sms</p></td>
    </tr>
  </table></td>
</tr>
</table></td>
```




```
<br/>
  <input name="chkbox3" type="checkbox" id="name_prostateuomeno_melos1"
onclick="showhidefield3()" value="NAIN" >
ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p>
</p>
<div id='hideablearea3' style='display:none'>
  <p class="text-green">
    ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos1" ><br>
    ΗΛΙΚΙΑ: <input type='text11' name="ilikia1"><br>
    ΦΥΛΟ:
    <select name="filo1" class="styled">
      <option value="" selected="selected">DEFINE</option>
      <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
      <option value="MAN">ΑΝΤΡΑΣ</option>
    </select>
  </p>
</div>
</p></div>
<p>Συμπλήρωσε τα στοιχεία του υπάλληλου ανάλογα με πιο κλάδο της εταιρίας ανήκει:
<br />
<br />
  <input name="chkbox2" type="checkbox" id="texnikoi"
onclick="showhidefield2()" value="yes">
  ΤΕΧΝΙΚΟ
  <br>
  </p>
</p>
<div id='hideablearea2' style='display:none'>
  <p class="text-green">
    ΕΙΔΙΚΟΤΗΤΑ:
    <input type='text123' name="eidikotita"><br>
    ΕΤΗ ΠΡΟΫΠΗΡΕΣΙΑΣ: <input type='text124' name="eti_proipiresias"><br />
  </p>
</div>
  <input name="chkbox1" type="checkbox" id="dioikitikoi" onclick="showhidefield1()"
value="yeap" >
  ΔΙΟΙΚΗΤΙΚΟ
  <br>
  </p>
<div id='hideablearea1' style='display:none'>
  <p class="text-green">
    ΤΙΤΛΟΣ ΣΠΟΥΔΩΝ:
    <input type='text1' name="titlos_spoudon"><br>
    ΗΜΕΡΟΜΗΝΙΑ ΚΤΗΣΗΣ ΠΤΥΧΙΟΥ: <input type='text2' name="imerominia_ptixiou">
    2011-12-10<br>
    ΠΑΝΕΠΙΣΤΗΜΙΟ: <input type='text3' name="panepistimio"> <br>
  </p>
</div>
  <br /><input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">
</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα επί 4 φορές ξεχωριστά στο ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο



ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΤΕΧΝΙΚΟ και στο ΔΙΟΙΚΗΤΙΚΟ κλάδο της εταιρίας.

```
<script language="JavaScript">
function showhidefield()
{
  if (document.frm.chkbox.checked)
  {
    document.getElementById("hideablearea").style.display = "block";
  }
  else
  {
    document.getElementById("hideablearea").style.display = "none";
  }
}
</script>
```

10.2.2.2 HTML Κώδικας για την σελίδα των Πελατών:

```
<form name = 'frm' action="pelatis.php" method="post">
ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type="text" name="pelatis_name"><br> <br>
ΕΠΑΓΓΕΛΜΑ: <input type="text" name = "pelatis_job"><br> <br>
ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "pelatis_ld"><br> <br>
ΔΙΕΥΘΥΝΣΗ: <input type="text" name = "pelatis_location"><br> <br>
<input name="chkbox" type="checkbox" id="ipologismos_posou" onclick="showhidefield()"
value="tpt" >
ΥΠΟΛΟΓΙΣΜΟΣ ΠΟΣΟΥ:</p>
</p>
  <div id='hideablearea' style='display:none'>
    <p class="text-green">
      ΣΥΝΟΛΙΚΟΣ ΧΡΟΝΟΣ ΟΜΙΛΙΑΣ: <input type='text' name="xronos_omilias" ><br><br>
      ΜΗΝΥΜΑΤΑ: <input type='text' name="minimata"><br><br>
      ΠΑΚΕΤΟ ΣΥΝΔΕΣΗΣ: <input type='text' name="paketo"><br><br>
      ΗΜΕΡΟΜΗΝΙΑ ΣΥΝΔΕΣΗΣ: <input type='text' name="date_connection"> 2011-12-10 </p>
    </div><br>
  <input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα για να υπολογίσω το ποσό.

```
<script language="JavaScript">
function showhidefield()
{
  if (document.frm.chkbox.checked)
  {
    document.getElementById("hideablearea").style.display = "block";
  }
  else
  {
    document.getElementById("hideablearea").style.display = "none";
  }
}
</script>
```



10.2.2.3 HTML Κώδικας για την σελίδα των Λογαριασμών:

```
<form action="logariasmoi1.php" method="post">
ΠΟΣΟ ΠΛΗΡΩΜΗΣ: <input type="text" name="poso_pliromis"><br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΕΚΔΟΣΗΣ: <input type="text" name = "date_ekdosis"> 2011-12-10<br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΠΛΗΡΩΜΗΣ: <input type="text" name = "date_pliromis"> 2011-12-10<br><br>
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΠΕΛΑΤΗ: <input type="text" name = "pelatis_name"><br> <br>
ΑΡΙΘΜΟΣ ΚΛΗΣΗΣ: <input type="text" name="phone_number"><br> <br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

10.2.2.4 HTML Κώδικας για την σελίδα στις Βλάβες:

```
<form action="vlaves1.php" method="post">
<span class="text-green">
ΠΟΤΕ ΕΓΙΝΕ: <input type="text" name="when"> 2011-12-10<br> <br>
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_name1"><br> <br>
ΥΠΑΛΛΗΛΟΙ ΠΟΥ ΤΗΝ ΕΠΙΔΙΟΡΘΩΣΑΝ: <input type="text" name = "ergates">
<br><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</span>
</form>
```

10.2.2.5 HTML Κώδικας για την σελίδα στις Κεραίες:

```
<form action="kerea1.php" method="post">
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name="kerea_name"><br> <br>
ΤΥΠΟΣ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_type"><br> <br>
ΔΙΕΥΘΥΝΣΗ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_location"><br> <br><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">
</form>
```

10.2.4.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης:

Έφτιαξα σε πίνακα τα πακέτα σύνδεσης αντι για φόρμα γιατί δεν χρειάζεται να προσθέσουμε στοιχεία στην βάση.

```
<table width="716" border="0">
<tr>
<td width="1076" align="center" valign="top"><p class="text-green">Ακολουθούν τα πακέτα
σύνδεσης και καρτοσυμβολαίου της εταιρίας μας:</p></td>
</tr>
<tr>
<td><table width="707" border="0">
<tr>
<td width="701"><table width="622" border="5" align="center" class="text-green">
<tr>
<td width="86" class="text-green"><p>Πακέτα σύνδεσης</p></td>
<td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
20</p></td>
<td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
45</p></td>
```




</form>

10.2.3 HTML Κώδικες του υποκαταστήματος του Λασιθίου

```
<td width="989" valign="top" class="text-green"><p><table width="758" border="0">
  <tr>
    <td width="420" align="center" valign="top"><p class="text-green">Καλώς ορίσατε στο
υποκατάστημα της κινητής <br/>τηλεφωνίας του Λασιθίου.<br/>
    Εδώ μπορείτε να ενημερωθείτε μια πλήρη εικόνα <br/>για την διαχείριση του
υποκαταστήματος.<br/>
    τα προϊόντα και τις υπηρεσίες μας...</p>
    <p class="text-green">Καλή περιήγηση...!!!</p>
  </p></td>
  <td width="328"></td>
</tr>
```

10.2.3.1 HTML Κώδικας για την σελίδα των Υπαλλήλων:

```
<form name='frm' form action="ipalilos3.php" method="post">
  <p class="text-green">
    ΟΝΟΜΑΤΕΠΩΝΥΜΟ:
    <input type="text" name="ipalilos_name"><br> <br>
    ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "id_ipalilos" > <br> <br>
    ΜΙΣΘΟΣ: <input type="text" name = "misthos"><br> <br>
    Συμπλήρωσε τα ακόλουθα στοιχεία αν υπάρχουν προστατευόμενα μέλη:<br />
    <br />
    <input name="chkbox" type="checkbox" id="name_prostateuomeno_melos"
    onclick="showhidefield()" value="NAI" >
    ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p> </p>
    <div id='hideablearea' style='display:none'>
    <p class="text-green">
      ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos" ><br>
      ΗΛΙΚΙΑ: <input type='text' name="ilikia"><br>
      ΦΥΛΟ:
      <select name="filo" class="styled">
        <option value="" selected="selected">DEFINE</option>
        <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
        <option value="MAN">ΑΝΤΡΑΣ</option>
      </select>
      <br/>
      <input name="chkbox3" type="checkbox" id="name_prostateuomeno_melos1"
      onclick="showhidefield3()" value="NAIN" >
      ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p> </p>
      <div id='hideablearea3' style='display:none'>
      <p class="text-green">
        ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos1" ><br>
        ΗΛΙΚΙΑ: <input type='text' name="ilikia1"><br>
        ΦΥΛΟ:
        <select name="filo1" class="styled">
          <option value="" selected="selected">DEFINE</option>
          <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
          <option value="MAN">ΑΝΤΡΑΣ</option>
        </select>
      </p>
```



```
</div>
</p></div>
<p>Συμπλήρωσε τα στοιχεία του υπάλληλου ανάλογα με πιο κλάδο της εταιρίας ανήκει:
<br />
<br />
<input name="chkbox2" type="checkbox" id="texnikoi"
onclick="showhidefield2()"value="yes">
  ΤΕΧΝΙΚΟ
<br>
</p>
</p>
<div id='hideablearea2' style='display:none'>
  <p class="text-green">
    ΕΙΔΙΚΟΤΗΤΑ:
    <input type='text123' name="eidikotita"><br>
    ΕΤΗ ΠΡΟΫΠΗΡΕΣΙΑΣ: <input type='text124' name="eti_proipiresias"><br />
  </p>
</div>
<input name="chkbox1" type="checkbox" id="dioikitikoi" onclick="showhidefield1() "
value="yeap" >
  ΔΙΟΙΚΗΤΙΚΟ
<br>
</p>
<div id='hideablearea1' style='display:none'>
  <p class="text-green">
    ΤΙΤΛΟΣ ΣΠΟΥΔΩΝ:
    <input type='text1' name="titlos_spoudon"><br>
    ΗΜΕΡΟΜΗΝΙΑ ΚΤΗΣΗΣ ΠΤΥΧΙΟΥ: <input type='text2' name="imerominia_ptixiou">
    2011-12-10<br>
    ΠΑΝΕΠΙΣΤΗΜΙΟ:<input type='text3' name="panepistimio"> <br>
  </p>
</div>
<br /><input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">

</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα επί 4 φορές ξεχωριστά στο ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΤΕΧΝΙΚΟ και στο ΔΙΟΙΚΗΤΙΚΟ κλάδο της εταιρίας.

```
<script language="JavaScript">
function showhidefield()
{
  if (document.frm.chkbox.checked)
  {
    document.getElementById("hideablearea").style.display = "block";
  }
  else
  {
    document.getElementById("hideablearea").style.display = "none";
  }
}
</script>
```



10.2.3.2 HTML Κώδικας για την σελίδα των Πελατών:

```
<form name = 'frm' action="pelatis.php" method="post">
ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type="text" name="pelatis_name"><br> <br>
ΕΠΑΓΓΕΛΜΑ: <input type="text" name = "pelatis_job"><br> <br>
ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "pelatis_id"><br> <br>
ΔΙΕΥΘΥΝΣΗ: <input type="text" name = "pelatis_location"><br> <br>
<input name="chkbox" type="checkbox" id="ipologismos_posou" onclick="showhidefield()"
value="tpt" >
ΥΠΟΛΟΓΙΣΜΟΣ ΠΟΣΟΥ:</p>
</p>
<div id='hideablearea' style='display:none'>
<p class="text-green">
ΣΥΝΟΛΙΚΟΣ ΧΡΟΝΟΣ ΟΜΙΛΙΑΣ: <input type='text' name="xronos_omilias" ><br><br>
ΜΗΝΥΜΑΤΑ: <input type='text' name="minimata"><br><br>
ΠΑΚΕΤΟ ΣΥΝΔΕΣΗΣ: <input type='text' name="paketo"><br><br>
ΗΜΕΡΟΜΗΝΙΑ ΣΥΝΔΕΣΗΣ: <input type='text' name="date_connection"> 2011-12-10 </p>
</div><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα για να υπολογίσω το ποσό.

```
<script language="JavaScript">
function showhidefield()
{
if (document.frm.chkbox.checked)
{
document.getElementById("hideablearea").style.display = "block";
}
else
{
document.getElementById("hideablearea").style.display = "none";
}
}
}
</script>
```

10.2.3.3 HTML Κώδικας για την σελίδα των Λογαριασμών:

```
<form action="logariasmoi3.php" method="post">
ΠΟΣΟ ΠΛΗΡΩΜΗΣ: <input type="text" name="poso_pliromis"><br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΕΚΔΟΣΗΣ: <input type="text" name = "date_ekdosis"> 2011-12-10<br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΠΛΗΡΩΜΗΣ: <input type="text" name = "date_pliromis"> 2011-12-10<br><br>
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΠΕΛΑΤΗ: <input type="text" name = "pelatis_name"><br> <br>
ΑΡΙΘΜΟΣ ΚΛΗΣΗΣ: <input type="text" name="phone_number"><br> <br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

10.2.3.4 HTML Κώδικας για την σελίδα στις Βλάβες:

```
<form action="vlaves3.php" method="post">
```



```
<span class="text-green">ΠΟΤΕ ΕΓΙΝΕ:  
<input type="text" name="when"> 2011-12-10<br> <br>  
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_name1"><br> <br>  
ΥΠΑΛΛΗΛΟΙ ΠΟΥ ΤΗΝ ΕΠΙΔΙΟΡΘΩΣΑΝ: <input type="text" name = "ergates">  
<br><br>  
  <input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />  
  </span>  
</form>
```

10.2.3.5 HTML Κώδικας για την σελίδα στις Κεραίες:

```
<form action="kerea3.php" method="post">  
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name="kerea_name"><br> <br>  
ΤΥΠΟΣ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_type"><br> <br>  
ΔΙΕΥΘΥΝΣΗ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_location"><br> <br><br>  
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">  
</form>
```

10.2.3.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης:

Έφτιαξα σε πίνακα τα πακέτα σύνδεσης αντι για φόρμα γιατί δεν χρειάζεται να προσθέσουμε στοιχεία στην βάση.

```
<table width="716" border="0">  
  <tr>  
    <td width="1076" align="center" valign="top"><p class="text-green">Ακολουθούν τα πακέτα  
    σύνδεσης και καρτοσυμβολαίου της εταιρίας μας:</p></td>  
  </tr>  
  <tr>  
    <td><table width="707" border="0">  
      <tr>  
        <td width="701"><table width="622" border="5" align="center" class="text-green">  
          <tr>  
            <td width="86" class="text-green"><p>Πακέτα σύνδεσης</p></td>  
            <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s  
20</p></td>  
            <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s  
45</p></td>  
            <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s  
70</p></td>  
            <td width="94" align="center" valign="middle"><p class="text-green">s@louf@s  
100</p></td>  
            <td width="99" align="center" valign="middle"><p class="text-green">s@louf@s  
150</p></td>  
          </tr>  
          <tr>  
            <td class="text-green"><p>Πάγιο</p></td>  
            <td align="center" valign="middle" class="text-green"><p>20 euro</p></td>  
            <td align="center" valign="middle" class="text-green"><p>45 euro</p></td>  
            <td align="center" valign="middle" class="text-green"><p>70 euro</p></td>  
            <td align="center" valign="middle" class="text-green"><p>100 euro</p></td>  
            <td align="center" valign="middle" class="text-green"><p>150 euro</p></td>  
          </tr>  
          <tr>  
            <td class="text-green"><p>Δωρέαν λεπτά</p></td>
```




10.2.4.1 HTML Κώδικας για την σελίδα των Υπαλλήλων:

```
<form name='frm' form action="ipalilos2.php" method="post">
  <p class="text-green">
    ΟΝΟΜΑΤΕΠΩΝΥΜΟ:
    <input type="text" name="ipalilos_name"><br> <br>
    ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "id_ipalilos" > <br> <br>
    ΜΙΣΘΟΣ: <input type="text" name = "misthos"><br> <br>
    Συμπλήρωσε τα ακόλουθα στοιχεία αν υπάρχουν προστατευόμενα μέλη:<br />
  <br />
  <input name="chkbox" type="checkbox" id="name_prostateuomeno_melos"
  onclick="showhidefield()" value="NAI" >
  ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p>
  </p>
  <div id='hideablearea' style='display:none'>
    <p class="text-green">
      ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos" ><br>
      ΗΛΙΚΙΑ: <input type='text' name="ilikia"><br>
      ΦΥΛΟ:
      <select name="filo" class="styled">
        <option value="" selected="selected">DEFINE</option>
        <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
        <option value="MAN">ΑΝΤΡΑΣ</option>
      </select>
      <br/>
      <input name="chkbox3" type="checkbox" id="name_prostateuomeno_melos1"
      onclick="showhidefield3()" value="NAIN" >
      ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ:</p>
    </p>
    <div id='hideablearea3' style='display:none'>
      <p class="text-green">
        ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type='text' name="name_prostateuomeno_melos1" ><br>
        ΗΛΙΚΙΑ: <input type='text' name="ilikia1"><br>
        ΦΥΛΟ:
        <select name="filo1" class="styled">
          <option value="" selected="selected">DEFINE</option>
          <option value="WOMAN">ΓΥΝΑΙΚΑ</option>
          <option value="MAN">ΑΝΤΡΑΣ</option>
        </select>
      </p>
    </div>
  </p></div>
  <p><br><br>
  <input name="chkbox2" type="checkbox" id="texnikoi"
  onclick="showhidefield2()" value="yes">
  ΤΕΧΝΙΚΟ
  <br>
  </p>
  </p>
  <div id='hideablearea2' style='display:none'>
    <p class="text-green">
      ΕΙΔΙΚΟΤΗΤΑ:
      <input type='text' name="eidikotita"><br>
```




```
    ΕΤΗ ΠΡΟΫΠΗΡΕΣΙΑΣ: <input type='text124' name='eti_proipiresias'><br />
  </p>
</div>
  <input name='chkbox1' type='checkbox' id='dioikitikoi' onclick='showhidefield1()'
value='yeap' >
  ΔΙΟΙΚΗΤΙΚΟ
  <br>
</p>
<div id='hideablearea1' style='display:none'>
  <p class='text-green'>
    ΤΙΤΛΟΣ ΣΠΟΥΔΩΝ:
    <input type='text1' name='titlos_spoudon'><br>
    ΗΜΕΡΟΜΗΝΙΑ ΚΤΗΣΗΣ ΠΤΥΧΙΟΥ: <input type='text2' name='imerominia_ptixiou'>
2011-12-10<br>
    ΠΑΝΕΠΙΣΤΗΜΙΟ:<input type='text3' name='panepistimio'> <br>
  </p>
</div>

  <br /><input type='submit' class='text-green' value='ΑΠΟΘΗΚΕΥΣΗ'>

</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα επί 4 φορές ξεχωριστά στο ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΕΠΙΠΛΕΟΝ ΠΡΟΣΤΑΤΕΥΟΜΕΝΟ ΜΕΛΟΣ, στο ΤΕΧΝΙΚΟ και στο ΔΙΟΙΚΗΤΙΚΟ κλάδο της εταιρίας.

```
<script language="JavaScript">
function showhidefield()
{
  if (document.frm.chkbox.checked)
  {
    document.getElementById("hideablearea").style.display = "block";
  }
  else
  {
    document.getElementById("hideablearea").style.display = "none";
  }
}
</script>
```

10.2.4.2 HTML Κώδικας για την σελίδα των Πελατών:

```
<form name = 'frm' action="pelatis.php" method="post">
ΟΝΟΜΑΤΕΠΩΝΥΜΟ: <input type="text" name="pelatis_name"><br> <br>
ΕΠΑΓΓΕΛΜΑ: <input type="text" name = "pelatis_job"><br> <br>
ΑΡΙΘΜΟΣ ΤΑΥΤΟΤΗΤΑΣ: <input type="text" name = "pelatis_ld"><br> <br> ΔΙΕΥΘΥΝΣΗ:
<input type="text" name = "pelatis_location"><br> <br>
<input name="chkbox" type="checkbox" id="ipologismos_posou" onclick="showhidefield()"
value="tpt" >
ΥΠΟΛΟΓΙΣΜΟΣ ΠΟΣΟΥ:</p>
</p>
```



```
<div id='hideablearea' style='display:none'>
  <p class="text-green">
    ΣΥΝΟΛΙΚΟΣ ΧΡΟΝΟΣ ΟΜΙΛΙΑΣ: <input type='text' name="xronos_omilias" ><br>
    ΜΗΝΥΜΑΤΑ: <input type='text' name="minimata"><br>
    ΠΑΚΕΤΟ ΣΥΝΔΕΣΗΣ: <input type='text' name="paketo"><br>
    ΗΜΕΡΟΜΗΝΙΑ ΣΥΝΔΕΣΗΣ: <input type='text' name="date_connection"> 2011-12-10 </p>
  </div><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

Για να κάνω τα πεδία να εμφανίζονται μόλις τσεκάρεις το checkbox βάζω πριν κλείσει το head των κάτω έτοιμο κώδικα τον οποίο βρήκα στο link: <http://www.tek-tips.com/faqs.cfm?fid=4585>. Τον κώδικα αυτό τον έβαλα για να υπολογίσω το ποσό.

```
<script language="JavaScript">
function showhidefield()
{
  if (document.frm.chkbox.checked)
  {
    document.getElementById("hideablearea").style.display = "block";
  }
  else
  {
    document.getElementById("hideablearea").style.display = "none";
  }
}
</script>
```

10.2.4.3 HTML Κώδικας για την σελίδα των Λογαριασμών:

```
<form action="logariasmoi2.php" method="post">
ΠΟΣΟ ΠΛΗΡΩΜΗΣ: <input type="text" name="poso_pliromis"><br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΕΚΔΟΣΗΣ: <input type="text" name = "date_ekdosis"> 2011-12-10<br> <br>
ΗΜΕΡΟΜΗΝΙΑ ΠΛΗΡΩΜΗΣ: <input type="text" name = "date_pliromis"> 2011-12-10<br><br>
ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΠΕΛΑΤΗ: <input type="text" name = "pelatis_name"><br> <br>
ΑΡΙΘΜΟΣ ΚΛΗΣΗΣ: <input type="text" name="phone_number"><br> <br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</form>
```

10.2.4.4 HTML Κώδικας για την σελίδα στις Βλάβες:

```
<form action="vlaves2.php" method="post">
<span class="text-green">ΠΟΤΕ ΕΓΙΝΕ:
<input type="text" name="when"> 2011-12-10<br> <br>
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_name1"><br> <br>
ΥΠΑΛΛΗΛΟΙ ΠΟΥ ΤΗΝ ΕΠΙΔΙΟΡΘΩΣΑΝ: <input type="text" name = "ergates">
<br><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ" />
</span>
</form>
```

10.2.4.5 HTML Κώδικας για την σελίδα στις Κεραίες:

```
<form action="kerea2.php" method="post">
```



```
ΟΝΟΜΑ ΚΕΡΑΙΑΣ: <input type="text" name="kerea_name"><br> <br>
ΤΥΠΟΣ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_type"><br> <br>
ΔΙΕΥΘΥΝΣΗ ΚΕΡΑΙΑΣ: <input type="text" name = "kerea_location"><br> <br><br>
<input type="submit" class="text-green" value="ΑΠΟΘΗΚΕΥΣΗ">
</form>
```

10.2.4.6 HTML Κώδικας για την σελίδα των Πακέτων σύνδεσης:

Έφτιαξα σε πίνακα τα πακέτα σύνδεσης αντι για φόρμα γιατί δεν χρειάζεται να προσθέσουμε στοιχεία στην βάση.

```
<table width="716" border="0">
  <tr>
    <td width="1076" align="center" valign="top"><p class="text-green">Ακολουθούν τα πακέτα
    σύνδεσης και καρτοσυμβολαίου της εταιρίας μας:</p></td>
  </tr>
  <tr>
    <td><table width="707" border="0">
      <tr>
        <td width="701"><table width="622" border="5" align="center" class="text-green">
          <tr>
            <td width="86" class="text-green"><p>Πακέτα σύνδεσης</p></td>
            <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
20</p></td>
            <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
45</p></td>
            <td width="94" align="center" valign="middle" class="text-green"><p>s@louf@s
70</p></td>
            <td width="94" align="center" valign="middle"><p class="text-green">s@louf@s
100</p></td>
            <td width="99" align="center" valign="middle"><p class="text-green">s@louf@s
150</p></td>
          </tr>
          <tr>
            <td class="text-green"><p>Πάγιο</p></td>
            <td align="center" valign="middle" class="text-green"><p>20 euro</p></td>
            <td align="center" valign="middle" class="text-green"><p>45 euro</p></td>
            <td align="center" valign="middle" class="text-green"><p>70 euro</p></td>
            <td align="center" valign="middle" class="text-green"><p>100 euro</p></td>
            <td align="center" valign="middle" class="text-green"><p>150 euro</p></td>
          </tr>
          <tr>
            <td class="text-green"><p>Δωρεάν λεπτά</p></td>
            <td align="center" valign="middle" class="text-green"><p>30'</p></td>
            <td align="center" valign="middle" class="text-green"><p>180'</p></td>
            <td align="center" valign="middle" class="text-green"><p>360'</p></td>
            <td align="center" valign="middle" class="text-green"><p>580'</p></td>
            <td align="center" valign="middle" class="text-green"><p>1000'</p></td>
          </tr>
          <tr>
            <td class="text-green"><p>Δωρεάν sms</p></td>
            <td align="center" valign="middle" class="text-green"><p>100</p></td>
            <td align="center" valign="middle" class="text-green"><p>120</p></td>
            <td align="center" valign="middle" class="text-green"><p>200</p></td>
            <td align="center" valign="middle" class="text-green"><p>360</p></td>
          </tr>
        </table>
      </tr>
    </table>
  </td>
</tr>
```




```
}
.logo1 {font-size: 25px}
.logo2 {font-size: 20px}
.text-green {color: #003300}
</style>
</head>
<body bgcolor="#FFFFFF" background="images/buttons_ipok.gif" text="#FFFFFF" class="text-
green">
<br/>
<table width="1050" border="0" align="center" cellpadding="0" cellspacing="0">
<tr>
<td><table width="468" border="0">
</table>
<table width="400" border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="center"><span class="logo1"> S@louf@s </span></td>
</tr>
<tr>
<td align="center" class="logo1"><small>Design by S@louf@s </small>
</p>
</td>
</tr>
</table></td>
</tr>
</table>
<table width="1050" border="0" align="center" cellpadding="5" cellspacing="5">
<tr>
<th width="1030" height="1115" nowrap="nowrap" bgcolor="#FFFFFF" border="100">
<table width="84%" border="0" align="center" cellpadding="10" cellspacing="10">
<tr>
<td bgcolor="#FFFFFF">
</td>
</tr>
</table>
<table width="995" border="0" align="center" cellpadding="5" cellspacing="5">
<tr>
<td height="20">
<table width="988" border="0" align="center">
<tr>
<th width="20%" height="55" background="images/buttons.png">ΑΡΧΙΚΗ</th>
<th width="20%" height="55" background="images/buttons.png">ΧΑΝΙΑ</th>
<th width="20%" height="55" background="images/buttons.png">ΠΕΘΥΜΝΟ</th>
<th width="20%" height="55" background="images/buttons.png">ΗΠΑΚΛΕΙΟ</th>
<th width="20%" height="55" background="images/buttons.png">ΛΑΣΙΘΙ</th>
</tr>
</table>
</tr>
<tr>
<td>
<table width="1007" border="0" align="center" cellpadding="5" cellspacing="5">
<tr>
<td width="325" height="140"></td>
<td width="330">
 </td>
<td width="338">
```



```
</td>
</tr>
</table>
</td>
</tr>
</table>
<table width="1002" height="478" border="0" align="center">
<tr>
<td width="742" height="474" valign="top" class="text-green"><table width="986" height="475"
border="0" align="center">
<tr>
<td width="736" height="471" valign="top" class="text-green"><table width="739" height="466"
border="0">
<tr>
<td height="462"><table width="1002" height="460" border="0" align="center">
<tr></tr>
<tr></tr>
<tr>
<td width="742" valign="top" class="text-green"><table width="986" height="420"
border="0" align="center">
<tr>
<td width="240" height="416" valign="top"><table width="240" border="0" align="right"
cellpadding="0" cellspacing="0">
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΥΠΑΛΛΗΛΟΙ </th>
</tr>
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΠΕΛΑΤΕΣ</th>
</tr>
<tr>
<th width="240" height="50" valign="middle"
background="images/buttons.jpg">ΛΟΓΑΡΙΑΣΜΟΙ</th>
</tr>
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΒΛΑΒΕΣ</th>
</tr>
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΚΕΡΑΙΕΣ</th>
</tr>
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΠΑΚΕΤΑ ΣΥΝΔΕΣΗΣ</th>
</tr>
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΑΡΙΘΜΟΙ ΤΗΛΕΦΩΝΩΝ </th>
</tr>
<tr>
<th width="240" height="50" valign="middle" background="images/buttons.jpg">
ΕΠΙΚΟΙΝΩΝΙΑ </th>
</tr>
```




Κεφάλαιο 11: Βιβλιογραφία

- [1]. E. F. Codd, *'The Relational Model for Database Management, Version 2'*, 2000.
- [2]. C. J. Date, *'An Introduction to Database Systems (8th Edition)'*, 2003.
- [3]. Tim Converse, Joyce Park, *'Php Bible', 2nd Edition*, 2002.
- [4]. R. Elmasri – S. B. Navathe, *'Θεμελιώδης Αρχές Συστημάτων Βάσεων Δεδομένων'*, 3^η Έκδοση Αναθεωρημένη, 2007.
- [5]. Marty Hall, Larry Brown, *'Servlets και σελίδες Διακομιστή Java'*, 2007.
- [6]. Andi Gutmans, Stig Bakken, Derick Rethan, *'Php 5 Power Programming'*, 2005.
- [7]. Luke Welling, Laura Thomson, *'Php and MySQL Web Development'*, 2008.
- [8]. Douglas E. Comer, *'Δίκτυα και διαδίκτυα υπολογιστών και εφαρμογές τους στο Ιντερνετ'*, 4^η Αμερικάνικη Έκδοση, Κλειδάριθμος 2007.
- [9]. Laura Lemay, *'Εγχειρίδιο της Html 3.2'*, 3^η Αμερικάνικη Έκδοση, Μ. Γκιούρδας, 1997.
- [10]. <http://www.w3schools.com/>
- [11]. <http://www.php.net/>.
- [12]. <http://www.mysql.com/>.
- [13]. <http://en.wikipedia.org>.
- [14]. <http://www.careerride.com/MySQL-disadvantages.aspx>.
- [15]. http://www.ehow.com/list_6309384_advantages-disadvantages-oracle-sql.html.
- [16]. <http://postgresql.gr/node/3>.
- [17]. <http://dide.flo.sch.gr/Plinet/Tutorials-PHP-Bible/Tutorials-PHP-Bible-1.html>.
- [18]. <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-MySQL.html>.
- [19]. <http://apache.org/>.
- [20]. <http://www.tek-tips.com/faqs.cfm?fid=4585>
- [21]. Παπαδάκης Νίκος, *'Σημειώσεις θεωρίας Θέματα προγραμματισμού Διαδικτύου'*, <https://eclass.teicrete.gr/modules/document/document.php?course=TP204&openDir=/4cbbda937046>, 2011.