

**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ
ΚΡΗΤΗΣ**
Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων

Πτυχιακή Εργασία



**“ Σχεδιασμός και ανάπτυξη δυναμικών διαδραστικών
εφαρμογών Διαδικτυακής Πύλης”**

“Λαμπράκη Χριστίνα”

ΕΙΣΗΓΗΤΗΣ: ΜΑΣΤΟΡΑΚΗΣ ΓΕΩΡΓΙΟΣ

Στους γονείς μου,
με ιδιαίτερη εκτίμηση και αγάπη ...

Ευχαριστίες

Με την ολοκλήρωση της πτυχιακής μου εργασίας, η οποία υλοποιήθηκε στο Εργαστήριο Τηλεπικοινωνιών και δικτύων (PASIPHAE) του Α.Τ.Ε.Ι ΚΡΗΤΗΣ, θα ήθελα να ευχαριστήσω τους ανθρώπους οι οποίοι βοήθησαν στην πραγματοποίησή της. Θα ήταν παράλειψη να μην αναφερθώ σε όλους εκείνους που συμπαραστάθηκαν σε αυτήν την προσπάθεια.

Κατά κύριο λόγο, οφείλω να ευχαριστήσω τον επιβλέποντα μου του Α.Τ.Ε.Ι ΚΡΗΤΗΣ κ. Μαστοράκη Γεώργιο ο οποίος με υποστήριξε καθ' όλη τη διάρκεια της εργασίας αυτής. Ευγνωμοσύνη οφείλω και στους Ερευνητές καθηγητές του Α.Τ.Ε.Ι ΚΡΗΤΗΣ Δρα Ζαχαρόπουλο Βασίλειο και Δρα Πάλλη Ευάγγελο, οι οποίοι με στήριξαν κατά τη διάρκεια υλοποίησης αυτής της μελέτης.

Θερμές ευχαριστίες οφείλω και στον κ. Δραγουμανάκη Γεώργιο για την πάντα πρόθυμη βοήθειά του στην υλοποίηση και δημιουργία του design της Διαδικτυακής Πύλης.

Τέλος, θέλω να ευχαριστήσω την οικογένεια μου και την παρέα μου που ήταν δίπλα μου σε όλη αυτή την προσπάθεια παρέχοντας απεριόριστη ψυχολογική υποστήριξη και κατανόηση.

Ηράκλειο, Σεπτέμβριος 2006
Χριστίνα Λαμπράκη

Περιεχόμενα

1. ΕΙΣΑΓΩΓΗ	8
1.1. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ.....	8
1.2. ΣΤΟΧΟΙ	8
1.3. ΔΟΜΗ.....	8
1.4. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	9
1.5. ΣΥΝΟΨΗ.....	9
2. ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΛΕΙΤΟΥΡΓΙΑΣ ΕΝΟΣ WEB SITE	10
2.2. ΑΡΑΧΗ ΚΑΙ ΤΟ ΚΙΝΗΜΑ ΑΝΟΙΚΤΟΥ ΚΩΔΙΚΑ (OPEN SOURCE).....	10
2.3. ΔΙΑΧΕΙΡΙΣΗ ΔΙΑΚΟΜΙΣΤΗ WEB.....	11
2.4. HTTP.....	13
2.5. HTML	17
2.5.2. Εκδόσεις HTML.....	18
2.5.3. Γλώσσες Προγραμματισμού στο Internet	18
2.5.4. Δημιουργία αρχείων HTML	18
2.6. JAVASCRIPT.....	19
2.6.1. Τι είναι JavaScript;.....	19
2.6.2. Τρόπος Λειτουργίας της JavaScript.....	19
2.7. CSS.....	20
2.8. PHP	20
2.8.1. Τι είναι η PHP;	20
2.8.2. Τι μπορεί να κάνει η PHP;.....	21
2.8.3. Πλεονεκτήματα της PHP.....	22
2.9. MYSQL	23
2.9.1. Τι είναι MySQL?.....	23
2.9.2. Πλεονεκτήματα της MySQL	24
3. ΤΡΟΠΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΗΣ PHP	25
3.1. ΧΡΗΣΗ ΤΗΣ PHP ΣΤΙΣ ΔΥΝΑΜΙΚΕΣ ΕΦΑΡΜΟΓΕΣ.....	25
3.2. ΠΡΟΚΑΘΟΡΙΣΜΕΝΕΣ ΜΕΤΑΒΛΗΤΕΣ.....	27
3.3. HTTP ΑΝΑΓΝΩΡΙΣΗ ΜΕ ΤΗΝ PHP	28
3.3.1. <i>header()</i>	28
3.3.2. <i>Χρησιμοποιώντας τις require() και include()</i>	29
3.4. ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΣΥΝΑΡΤΗΣΕΙΣ	30
3.4.1. <i>Mail συναρτήσεις</i>	30
3.4.2. <i>Upload με την POST μέθοδο</i>	31
3.4.3. <i>Επικύρωση upload αρχείων</i>	32
3.4.4. <i>is_uploaded_file()</i>	33
3.4.5. <i>move_uploaded_file()</i>	34
3.4.6. <i>basename()</i>	34
4. MYSQL ΜΕ PHP	36
4.1. ΕΙΣΑΓΩΓΗ.....	36
4.2. ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ ΣΤΗΝ PHP	36
4.3. ΣΥΝΔΕΣΗ ΤΗΣ MYSQL ΜΕ PHP	37
4.3.1. <i>mysql_connect()</i>	38
4.3.2. <i>mysql_close()</i> ;	39
4.3.3. <i>mysql_select_db()</i>	39
4.3.4. <i>mysql_query(\$query)</i> ;.....	40
4.4. ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ	40
4.5. ΕΜΦΑΝΙΣΗ ΔΕΔΟΜΕΝΩΝ	41
4.6. Ο ΤΟΜΕΑΣ ID.....	42
4.7. ΔΙΟΡΘΩΣΗ ΚΑΙ ΔΙΑΓΡΑΦΗ ΔΕΔΟΜΕΝΩΝ.....	43
4.8. ΔΙΑΓΡΑΦΗ ΔΕΔΟΜΕΝΩΝ	44
5. ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ.....	46
5.1. ΓΕΝΙΚΗ ΜΟΡΦΗ	46

5.2.	ΔΙΣΚΟΓΡΑΦΙΚΕΣ ΕΤΑΙΡΕΙΕΣ (ΕΛΛΑΔΑ-ΚΡΗΤΗΣ),.....	47
5.3.	ΠΡΟΓΡΑΜΜΑ ΕΒΔΟΜΑΔΑΣ.....	50
5.4.	ΤΟΠ 40 (ΕΛΛΗΝΙΚΟ, ΞΕΝΟ, ΚΡΗΤΙΚΟ).....	54
5.5.	ΜΑΝΤΙΝΑΔΕΣ.....	54
5.6.	ΠΑΡΑΓΩΓΟΙ.....	56
5.7.	ΜΟΥΣΙΚΑ ΝΕΑ.....	58
5.8.	ΨΗΦΟΦΟΡΙΑ.....	59
5.9.	ΑΣΦΑΛΕΙΑ ΔΙΑΧΕΙΡΙΣΤΗ.....	62
6.	ΠΕΡΙΛΗΨΗ.....	65
6.1.	ΣΥΜΠΕΡΑΣΜΑΤΑ.....	65
6.2.	ΜΕΛΛΟΝΤΙΚΗ ΔΟΥΛΕΙΑ.....	65
	ΑΚΡΩΝΥΜΙΑ.....	66
	ΠΑΡΑΡΤΗΜΑ.....	68
	Β. ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΚΩΔΙΚΑ ΡΗΡ ΜΕΣΑ ΣΤΙΣ ΕΦΑΡΜΟΓΕΣ.....	71
	ΒΙΒΛΙΟΓΡΑΦΙΑ.....	109

Πίνακες

Πίνακας 1 Top Developers	13
Πίνακας 2 Εντολές για διαχείριση φορμών	16
Πίνακας 3 Σύνολο Βάσεων Δεδομένων.....	22
Πίνακας 4 ιδιότητες τομέων στην MySQL.....	37
Πίνακας 5 Παράδειγμα δημιουργίας μιας βάσης Δεδομένων	37

Σχήματα

Σχ. 2 Διακομιστή βάση δεδομένων	11
Σχ. 3 Total Sites Across All Domains August 1995 - September 2005.....	12
Σχ. 4 Market Share for Top Servers Across All Domains August 1995	13
Σχ. 5 Απλό Παράδειγμα PHP	25
Σχ. 6 Παράδειγμα Header() με το string "HTTP/"	28
Σχ. 7 Παράδειγμα Header() με το string "HTTP/"	29
Σχ. 8 Ένα απλό παράδειγμα της mail().....	30
Σχ. 9 Παράδειγμα της mail() με επιπλέον παραμέτρους	30
Σχ. 10 Ένα μικρό παραδειγμα is_uploaded_file	34
Σχ. 11 Ένα παράδειγμα με basename().....	35
Σχ. 12 Design Διαδικτυακής Πύλης	46
Σχ. 13 δομή του template Διαχειριστή.....	47
Σχ. 14 Δομή του «Δισκογραφικές Εταιρείες Ελλάδας»	48
Σχ. 15 Φόρμα διόρθωσης «Δισκογραφικές Εταιρείες Ελλάδας»	49
Σχ. 16 Εισαγωγή νέων Δεδομένων «Δισκογραφικές Εταιρείες Ελλάδας».....	50
Σχ. 17 Δομή «Πρόγραμμα Εβδομάδας»	51
Σχ. 18 Δομή του template Διαχειριστή «Πρόγραμμα Εβδομάδας».....	52
Σχ. 19 Φόρμα εισαγωγής νέων δεδομένων «Πρόγραμμα Εβδομάδας».....	53
Σχ. 20 Δομή «Τοπ 40-ελληνικό»	54
Σχ. 21 Δομή «Μαντινάδες» με φόρμα αποστολής.....	55
Σχ. 22 Δομή Διαχείρισης εισαγωγή μαντινάδας	56
Σχ. 23 Μορφή Λειτουργίας του «Παραγωγοί».....	57
Σχ. 24 Δομή των «Μουσικών Νέων»	58
Σχ. 25 Δομή της σελίδας more_music_news.php.....	59
Σχ. 26 Μορφή Λειτουργίας της ψηφοφορίας (Poll Panel)	60
Σχ. 27. Επιλογές ερώτησης Add, Delete και Edit.....	60
Στο Σχ. 28 δίνει την δυνατότητα στον διαχειριστή να μπορεί να εισάγει νέες επιλογές ερωτήσεων, να διαγράψει και να διορθώσει κάποιες από αυτές (βλ. Παράρτημα Β,7). Ο πίνακας που χρησιμοποιείται είναι η p_choices (βλ. Παράρτημα Α, 7). Αυτός ο πίνακας περιλαμβάνει το όνομα της επιλογή ερώτησης (answer) και έναν τομέα (votes) που θα καταχωρούνται οι ψήφοι των χρηστών που θα δούμε παρακάτω.	61
Σχ. 29 Τρόπος Λειτουργίας Ερώτησης με Add, Delete και Edit.....	61

Σχ. 30 εισαγωγή δεύτερης νέας ερώτησης.....	61
Σχ. 31 Τρόπος χρήσης ψηφοφορίας από χρήστη.....	62
Σχ. 32 Κωδικός Admin για την εισαγωγή του Διαχειριστή.....	63
Σχ. 33 Φορμα εισαγωγή email για υπευθύμιση κωδικού διαχειριστή.....	63
Εάν ο διαχειριστής προσπαθήσει να μπει σε κάποια σελίδα του template χωρίς να κάνει Login με password, τότε θα του χτυπήσει ERROR(Σχ. 34) .	
Χρησιμοποιούμε την εντολή \$HTTP_COOKIE_VARS['userid'];, που υπάρχει σε όλες τις σελίδες του template ώστε να είναι γνωστό σε κάθε σελίδα οτι είναι μέσα ο διαχειριστής.	63
Σχ. 35 Σελίδα diskografies_el.php που δεν έχει κάνει login ο Διαχειριστής.....	64
Σχ. 36 Εισαγωγή Νέου Διαχειριστή.....	64

1. Εισαγωγή

1.1. Γενική Περιγραφή

Ο βασικός στόχος της πτυχιακής εργασίας είναι να σχεδιαστούν και να υλοποιηθούν όλες οι κατάλληλες διεπαφές (interfaces), τα εργαλεία υλικού και λογισμικού που θα χρησιμοποιηθούν καθώς και τα δυναμικά στοιχεία για τη δημιουργία μίας Διαδικτυακής Πύλης. Θα καθοριστούν και θα αναπτυχθούν δυναμικές δικτυακές εφαρμογές για το ραδιοφωνικό σταθμό Melody ενώ θα υπάρχει η δυνατότητα πρόσβασης σε χρήσιμες πληροφορίες σχετικά με το συγκεκριμένο αντικείμενο της ιστοσελίδας. Σε αυτό το πλαίσιο θα μελετηθεί η συνολική και εσωτερική διαμόρφωση των υποσέλιδων και ο τρόπος διασύνδεσης μεταξύ τους, ενώ παράλληλα θα σχεδιαστεί η κατάλληλη διεπαφή για φιλική πρόσβαση στον χρήστη και εύκολη πλοήγηση. Επιπλέον σε αυτήν την πτυχιακή εργασία θα αναπτυχθεί όλο το υλικό και το λογισμικό που απαιτείται για την εγκατάσταση του δικτυακού εξυπηρετητή (Web-server) που θα φιλοξενεί τη συγκεκριμένη Διαδικτυακή Πύλη και θα παρέχει πρόσβαση των χρηστών στις δυναμικές υπηρεσίες και στο διαθέσιμο περιεχόμενο.

1.2. Στόχοι

Στόχος της πτυχιακής εργασίας είναι η εύκολη πρόσβαση σε χρήσιμες πληροφορίες σχετικά με το συγκεκριμένο αντικείμενο της ιστοσελίδας. Τα στατικά περιεχόμενα μιας HTML Web τοποθεσίας δεν μας είναι αρκετά για την δημιουργία αυτών των δυναμικών εφαρμογών και οι χρήστες δεν θα μπορούν να συνδιαλέγονται με την τοποθεσία αυτή. Σε αυτή τη περίπτωση θα χρησιμοποιήσουμε μια άλλη γλώσσα, την PHP για την πραγματοποίηση δυναμικών εφαρμογών. Με αυτόν τον τρόπο επιτυγχάνεται η εύκολη πρόσβαση των χρηστών και η αποτελεσματική προώθηση του ραδιοφωνικού σταθμού MelodyFM.

1.3. Δομή

Ως συνέχεια αυτής της εισαγωγικής ενότητας, ακολουθεί το δεύτερο κεφάλαιο το οποίο εισάγει τον αναγνώστη στη λειτουργία ενός web site σε ένα Apache Web Server, πώς λειτουργεί ένας διακομιστής Web καθώς και στη σημασία του πρωτοκόλλου HTTP. Επίσης θα μελετηθούν γενικές έννοιες του HTML, Javascript, CSS, PHP και MySQL που είναι η βάση για την δημιουργία μιας Διαδικτυακής Πύλης.

Αντικείμενο του τρίτου κεφαλαίου, αποτελεί η θεωρία και γενικά θέματα που αφορούν την ανάπτυξη της PHP. Αυτά είναι απαραίτητα για τη δημιουργία των δυναμικών εφαρμογών που παρουσιάζονται σε άλλα κεφάλαια.

Στο τέταρτο κεφάλαιο, παρουσιάζεται η θεωρία και η σχεδίαση των σχεσιακών συστημάτων βάσεων δεδομένων, όπως είναι η MySQL, η σύνδεση της με την PHP και κάποια θέματα πιο προχωρημένα στην MySQL.

Τέλος, στο πέμπτο κεφάλαιο, αναλύονται ειδικότερα ο τρόπος χρήσης της PHP μέσα στις εφαρμογές καθώς και ο τρόπος λειτουργίας της μέσα στα υποσέλιδα της Διαδικτυακής Πύλης.

1.4. Θεωρητικό Υπόβαθρο

Ειδικότερα θα αναλυθεί ο τρόπος χρήσης της PHP μέσα στις εφαρμογές καθώς και ο τρόπος λειτουργίας της μέσα στα υποσέλιδα της Διαδικτυακής Πύλης. Επίσης θα παρουσιασθεί ο τρόπος που συνδέεται η βάση δεδομένων με τις εφαρμογές αυτές μέσα στον κώδικα της PHP.

1.5. Σύνοψη

Η PHP είναι μία γλώσσα ανοιχτού κώδικα για συγγραφή script, η οποία χρησιμοποιείται για τη δημιουργία δυναμικά παραγόμενων ιστοσελίδων. Με μία εύκολη στην χρήση σύνταξη και μία μεγάλη, επεκτάσιμη βιβλιοθήκη λειτουργικών μονάδων, η PHP διευκολύνει και κάνει ταχύτερη τη δημιουργία δυνατών δικτυακών τόπων, που καθοδηγούνται από τα δεδομένα, δικτυακών πυλών και άλλων εφαρμογών που βασίζονται στο Web. Αυτά τα στοιχεία θα εφαρμοσθούν στις εφαρμογές που παρουσιάζονται στα παρακάτω κεφάλαια.

2. Γενική Περιγραφή Λειτουργίας ενός Web site

2.1. Διακομιστές Web

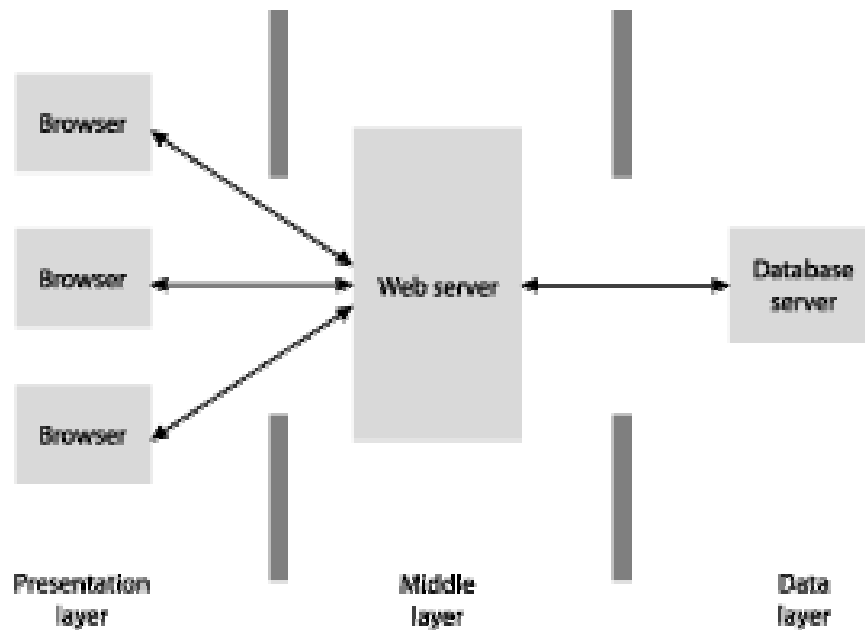
Ο διακομιστής Web είναι ένας εξειδικευμένος διακομιστής αρχείων που διανέμει αρχεία τα οποία περιέχουν ιστοσελίδες, εικόνες γραφικών, sound clips, video clips και άλλο πολυμεσικό περιεχόμενο. Όταν ένας χρήστης εκτελεί μια ενέργεια, όπως η χρήση ενός συνδέσμου ή να κατάθεση στοιχείων σε μία φόρμα, αποστέλλεται στον διακομιστή ένα μήνυμα, το οποίο μεταφέρει την ενέργεια που έλαβε χώρα, μαζί με τα σχετικά δεδομένα, για παράδειγμα το όνομα μέσα σ' ένα πεδίο κειμένου το οποίο πληκτρολογήθηκε από τον χρήστη. [5]

Το μήνυμα που στάλθηκε από τον browser λαμβάνεται από τον διακομιστή Web, ο οποίος στην συνέχεια αποφασίζει τι πρέπει να γίνει: εάν, για παράδειγμα, μία ιστοσελίδα πρέπει να σταλεί στον χρήστη ή εάν πρέπει να εκτελεστεί κάποιο πρόγραμμα. Στην τελευταία περίπτωση, αυτό ίσως συνεπάγεται την επεξεργασία μιας αποθηκευμένης ιστοσελίδας πριν την αποστολή της προς τον browser. Αυτή η επεξεργασία αναλύεται στο επόμενο κεφάλαιο. Το αίτημα εκτελείται και ο διακομιστής ετοιμάζεται για το επόμενο αίτημα το οποίο μπορεί να προέρχεται από τον ίδιο υπολογιστή / browser που έθεσε το προηγούμενο αίτημα αλλά μπορεί και όχι. Τα αιτήματα και οι απαντήσεις που αναφέρθηκαν, εκφράζονται με ένα πρωτόκολλο γνωστό ως **Hypertext Transfer Protocol (HTTP)**. [5]

2.2. Apache και το Κίνημα Ανοικτού Κώδικα (open source)

Ο Apache είναι ο πιο διάσημος διακομιστής που χρησιμοποιείται στο web. Είναι διαθέσιμος για μια ποικιλία διαφορετικών λειτουργικών συστημάτων, με το UNIX να είναι το πιο διάσημο. Ο Apache αποτελεί σημαντικό μέρος ενός κινήματος γνωστού ως Κίνημα Ανοικτού Κώδικα. Η φιλοσοφία πίσω από αυτό είναι ότι το λογισμικό έχει γίνει τόσο "μπαρόκ" και περίπλοκο που καμία εταιρεία δεν μπορεί να έχει πλήρη γνώση όλων των προβλημάτων και των λαθών που μπορεί να εμφανιστούν. Ο μόνος τρόπος με τον οποίο μπορεί να γίνει αυτό είναι μέσω της διάθεσης σε μια μεγάλη κοινότητα προγραμματιστών του πηγαίου κώδικα και της άδειας σε αυτούς να αναζητήσουν και να διορθώσουν προβλήματα σ' αυτόν. Μ' αυτόν τον τρόπο θα υπάρχει μία γρήγορη διάδοση διορθώσεων και βελτιώσεων των προγραμμάτων. [5]

Ο διακομιστής Web, μαζί με τα επιχειρηματικά αντικείμενα μιας εφαρμογής, λειτουργεί ως το μεσαίο επίπεδο ενός συστήματος τριών επιπέδων, με τον browser να έχει την θέση του επιπέδου παρουσίασης και τον διακομιστή βάσης δεδομένων να λειτουργεί ως το επίπεδο της βάσης δεδομένων. [5]



Σχ. 1 Διακομιστή βάσης δεδομένων

Το παραπάνω σχήμα είναι πολύ απλό μια και παρουσιάζει μόνο έναν διακομιστή βάσης δεδομένων, μπορεί να υπάρχουν πολλοί περισσότεροι μέσα σ' ένα δίκτυο. Αυτό μπορεί να περιπλέξει σημαντικά τον προγραμματισμό ενός τέτοιου συστήματος. [5]

2.3. Διαχείριση Διακομιστή Web

Ο διαχειριστής ενός δικτυακού τόπου (webmaster) πρέπει να φέρει εις πέρας ορισμένα ζητήματα:

Να βελτιστοποιήσει την απόδοση του διακομιστή. Για παράδειγμα, μία σημαντική απόφαση που πρέπει να πάρει ο διαχειριστής θέσης ιστού είναι ποιες σελίδες ν' αποθηκευθούν στην γρήγορη cache μνήμη. Για να το επιτύχει αυτό ο webmaster πρέπει να εξετάσει τα log αρχεία που τηρεί ο διακομιστής με σκοπό να δει ποιες είναι οι πιο δημοφιλείς σελίδες.

Να ορίσει τις ρυθμίσεις λειτουργίας του διακομιστή. Αυτό περιλαμβάνει πολλά, από μικρά ζητήματα όπως η δημιουργία της ηλεκτρονικής διεύθυνσης του webmaster έως την επιλογή των modules λογισμικού που θα υπάρχουν στον διακομιστή.

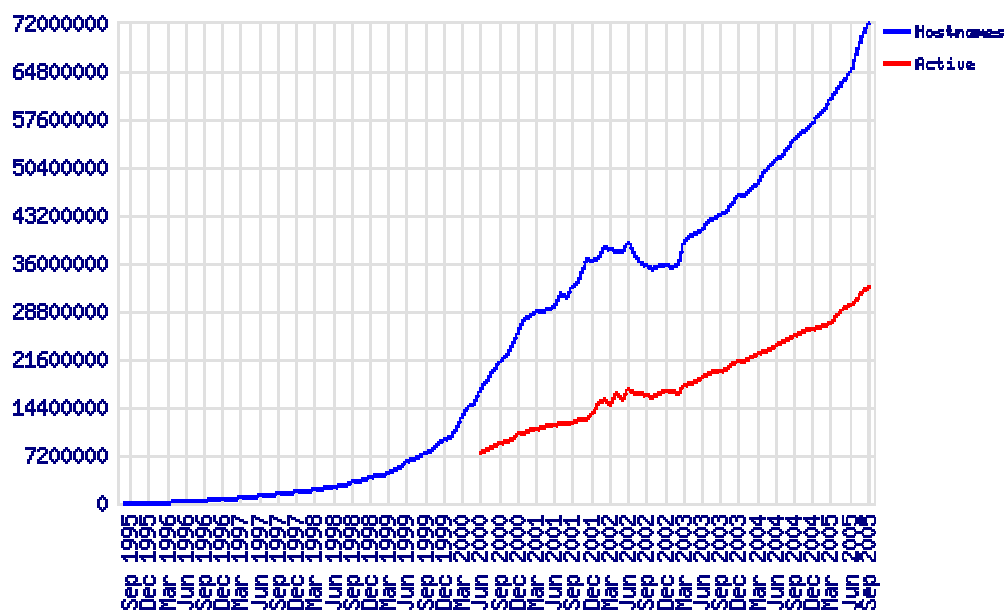
Οργάνωση των αρχικών καταλόγων και αρχείων, τα οποία θα αποτελέσουν μέρος του ιστοχώρου που θα διαχειρίζεται ο διακομιστής. Αυτά τα αρχεία φτιάχνονται και συντηρούνται συνήθως από τους σχεδιαστές, αν και σε μικρούς οργανισμούς ο webmaster έχει και θέση προγραμματιστή ιστοχώρου. [5]

Να θέσει όρους ασφαλείας οι οποίοι επιτρέπουν ή αποτρέπουν την πρόσβαση στους χρήστες. Για παράδειγμα, κάποιες σελίδες ενός ιστοχώρου είναι

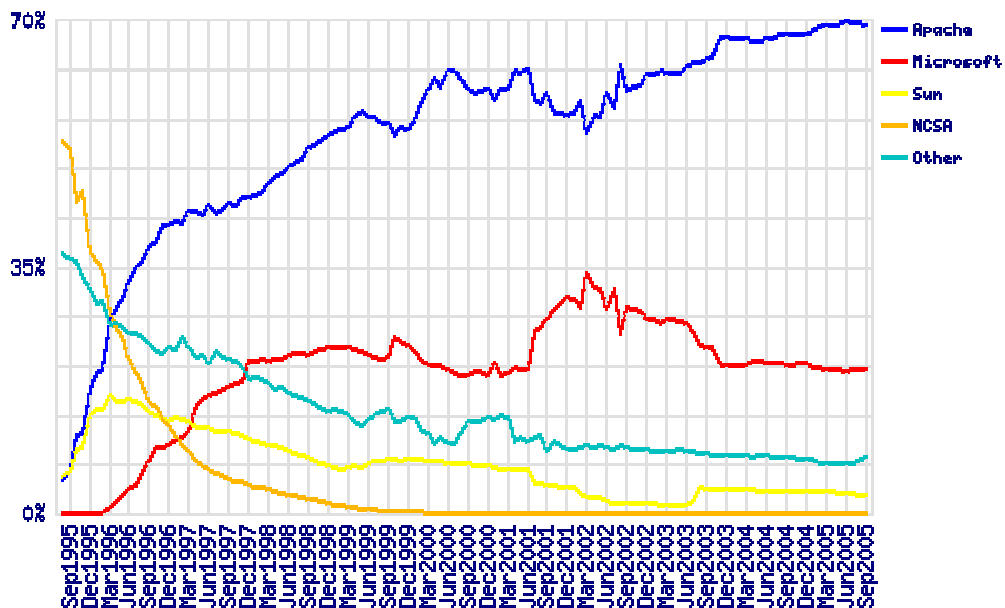
προσβάσιμες μόνο από συνδρομητές. Στην περίπτωση αυτή ο webmaster ενεργοποιεί κάποια κατάλληλη ρύθμιση στον διακομιστή Web. [6]

Να παρακολουθεί τα log αρχεία για πιθανές εισβολές οι οποίες μπορεί να φανούν επιβλαβείς για την ασφάλεια του διακομιστή. [6]

Ο **Apache Web Server** είναι ο πιο διαδεδομένος εξυπηρετητής στο Internet (σύμφωνα με τη σχετική έρευνα της Netscraft, καθώς αποτελεί ίσως τον περισσότερο προηγμένο τεχνολογικά, σταθερό και αποδοτικό web server. [6]



Σχ. 2 Total Sites Across All Domains August 1995 - September 2005[6]



Σχ. 3 Market Share for Top Servers Across All Domains August 1995 - September 2005[6]

Developer	August 2005	Percent	September 2005	Percent	Change
Apache	48895205	69.46	49598424	69.15	-0.31
Microsoft	14384006	20.43	14601553	20.36	-0.07
Sun	1864788	2.65	1868891	2.61	-0.04
Zeus	580675	0.82	584598	0.82	0.00

Πίνακας 1 Top Developers[6]

2.4. HTTP

Υπάρχουν δύο πλευρές στο πρωτόκολλο HTTP. Η πρώτη είναι το σύνολο των μηνυμάτων που αποστέλλονται από τον χρήστη που τρέχει τον browser και η δεύτερη είναι το σύνολο των απαντήσεων του διακομιστή. Θα εξετάσω πρώτα το πρώτο σύνολο μηνυμάτων. [4]

Το πιο κοινό παράδειγμα μηνύματος από την πλευρά του χρήστη είναι το μήνυμα / αίτημα GET. Αυτό που κάνει είναι απλά να ζητήσει από τον διακομιστή κάποιο αρχείο ή άλλο πόρο. Παρακάτω δίνεται ένα συγκεκριμένο παράδειγμα: [4]

```
GET /index.html HTTP/1.1
User-Agent: Lynx/2.4
```

Connection: Keep-Alive
Host: www.openaccess.com
Accept: text/html

Η πρώτη γραμμή διευκρινίζει ότι ο browser θέλει να ανακτήσει το αρχείο *index.html* και ότι αυτήν την στιγμή χρησιμοποιεί την έκδοση 1.1 του πρωτοκόλλου HTTP. [4]

Η επόμενη γραμμή δηλώνει ότι ο χρήστης τρέχει την έκδοση 2.4 του Lynx browser. Η επόμενη σειρά δηλώνει ότι ο browser επιθυμεί την διατήρηση της σύνδεσης με τον Web browser, για παράδειγμα για να κατεβάσει εικόνες που περιέχονται σε μία σελίδα που ζητείται. Η επόμενη σειρά προσδιορίζει τον υπολογιστή στον οποίο βρίσκεται αποθηκευμένο το αρχείο . Τέλος, η τελευταία γραμμή ενημερώνει τον διακομιστή Web ότι για την αίτηση αυτή αποδέχεται μόνο αρχεία κειμένου που περιέχουν κώδικα HTML. Όταν το αίτημα περιλαμβάνει την επεξεργασία μίας φόρμας τα δεδομένα που μεταφέρονται από την φόρμα θ' ακολουθήσουν την τελευταία από τις σειρές της κεφαλίδας. [4]

Η πρώτη σειρά είναι γνωστή ως **σειρά αιτήματος** και οι ακόλουθες ως **σειρές κεφαλίδας**. Κάθε σειρά κεφαλίδας αποτελείται από μια λέξη κλειδί ακολουθούμενη από άνω και κάτω τελεία και ένα όρισμα, που αντιστοιχεί στην τιμή της λέξης κλειδί. [4]

Όταν ο διακομιστής παραλαμβάνει το μήνυμα, ανταποκρίνεται σ' αυτό που ζήτησε ο browser. Έτσι, για παράδειγμα, αν το αίτημα αφορούσε ένα αρχείο HTML το οποίο υπάρχει στα αρχεία ενός διακομιστή Web τότε η απάντηση θα ήταν η ακόλουθη: [4]

```
HTTP/1.1 200 OK
Date: Thu, 22 July 1998 18:40:55 GMT
Server: Apache 1.3.5 (Unix) PHP/3.0.6
Last-Modified: Mon, 19 July 1997 16:03:22 GMT
Content-Type: text/html
Content-Length: 12987
```

...

Η απάντηση αποτελείται από τρία μέρη. Το πρώτο μέρος είναι η πρώτη σειρά που δηλώνει ότι ο διακομιστής χρησιμοποιεί την έκδοση 1.1 του πρωτοκόλλου HTTP και ότι το αίτημα του browser δεν προκάλεσε κάποιο λάθος (το 200 αντιπροσωπεύει έναν κώδικα κατάστασης που δηλώνει ακριβώς αυτό). Αυτό ονομάζεται **γραμμή κατάστασης**. [4]

Η επόμενη ομάδα σειρών είναι γνωστή ως πληροφορίες κεφαλίδας. Έχει την ίδια μορφή με τις πληροφορίες κεφαλίδας που εντοπίζονται στο αίτημα ενός πελάτη. Η πρώτη γραμμή προσδιορίζει τον χρόνο επεξεργασίας του αιτήματος, η επόμενη σειρά τον διακομιστή (Apache) που ανταποκρίνεται στο αίτημα, η αμέσως επόμενη διευκρινίζει το πότε ενημερώθηκε για τελευταία φορά ο πόρος, η προτελευταία διευκρινίζει ότι το αρχείο είναι ένα αρχείο κειμένου που περιέχει κώδικα HTML (αυτό είναι γνωστό ως τύπος MIME) και η τελευταία σειρά αποδίδει το μέγεθος του αρχείου σε χαρακτήρες. [4]

Τύποι MIME

Όταν ένας διακομιστής στέλνει ένα αρχείο πίσω στον χρήστη πρέπει να τον ενημερώσει για το περιεχόμενο του αρχείου, μιας και αυτό μπορεί να περιέχει τα πάντα, από απλή HTML μέχρι ένα εκτελέσιμο πρόγραμμα. Το κείμενο που περιγράφει το περιεχόμενο είναι γνωστό ως τύπος MIME (συντομογραφία του Multipurpose Internet Mail Extensions). Στο παραπάνω παράδειγμα το text/html είναι ο τύπος MIME που σχετίζεται με HTML εκφραζόμενη ως απλό κείμενο. Άλλοι τύποι MIME περιλαμβάνουν τον image/gif που δηλώνει μία εικόνα με την μορφή γραφικού gif, τον video/quicktime που δηλώνει ένα animation QuickTime και το text/plain που δηλώνει απλό κείμενο. Η γενική μορφή του τύπου MIME είναι ο τύπος Κατηγορία περιεχομένου/Τύπος Περιεχομένου, όπου η κατηγορία περιεχομένου προσδιορίζει την γενική κατηγορία του περιεχομένου του αρχείου, για παράδειγμα κείμενο, και ο τύπος περιεχομένου προσδιορίζει ποιο ακριβώς είδος περιεχομένου υπάρχει στο αρχείο, για παράδειγμα κείμενο HTML ή κείμενο ASCII. Οι τύποι MIME αρχικά χρησιμοποιούνταν για την επισύναψη σε μηνύματα ηλεκτρονικού ταχυδρομείου περιεχομένου που δεν ήταν απλό κείμενο. [4]

Οι πληροφορίες κεφαλίδας ακολουθούνται από κώδικα HTML, το περιεχόμενο του αρχείου που έχει ζητηθεί. Αυτού προηγείται μία κενή γραμμή. Αν ο τύπος του περιεχομένου ήταν διαφορετικός, θα ακολουθούσαν τα περιεχόμενα ενός άλλου τύπου αρχείου, π.χ. μιας εικόνας. [4]

Αφού ο browser παραλάβει την γραμμή κατάστασης, τις γραμμές κεφαλίδας και τον κώδικα HTML θα εμφανίσει στην οθόνη του χρήστη το περιεχόμενο μορφοποιημένο, αφού πρώτα ερμηνεύσει τα tags της HTML που υπάρχουν στο κείμενο. [4]

Οι τύποι αιτημάτων που μπορεί να θέσει ένας browser είναι πολλοί. Ο Πίνακας 6.2 τους παραθέτει περιληπτικά. Υπάρχουν ορισμένα πράγματα που αξίζει ν' αναφέρουμε για τον πίνακα αυτό και το περιεχόμενό του. Πρώτον, δεν φαίνονται όλες οι εντολές, μόνο κάποιες επιλεγμένες, οι πιο συνηθισμένες. Δεύτερον, η εντολή HEAD είναι παρόμοια με την εντολή GET. Ωστόσο, δεν επιστρέφει κάποιο περιεχόμενο, μόνο την κεφαλίδα HTTP. Χρησιμοποιείται όταν ο browser θέλει να αποκτήσει κάποια πληροφορία σχετικά μ' έναν συγκεκριμένο πόρο / αρχείο όπως την ημερομηνία τελευταίας τροποποίησης. Η εντολή PUT λειτουργεί μόνο όταν ο χρήστης έχει την άδεια να τοποθετήσει αρχεία στον διακομιστή Web. Τέλος, η εντολή OPTIONS έχει ως αποτέλεσμα την επιστροφή ενός μηνύματος που περιέχει πληροφορίες σχετικά με τον διακομιστή ή ένα συγκεκριμένο αρχείο / πόρο που υπάρχει στον διακομιστή, για παράδειγμα θα μπορούσε να έχει ως αποτέλεσμα την δημιουργία μιας λίστας διαθέσιμων αιτημάτων HTTP και την αποστολή της στον αιτούντα. [4]

Εντολή	Σημασία
GET	Πάρε ένα αρχείο από τον διακομιστή
HEAD	Το ίδιο με το GET αλλά χωρίς την επιστροφή περιεχομένου
POST	Κλήση προγράμματος για επεξεργασία μιας φόρμας
PUT	Αποθήκευση ενός αρχείου
OPTIONS	Επέστρεψε τις ρυθμίσεις του διακομιστή

Πίνακας 2 Εντολές για διαχείριση φορμών

Πριν προχωρήσουμε αξίζει να δούμε πως γίνεται η διαχείριση των φορμών δεδομένων. Όταν πατηθεί ένα κουμπί υποβολής (submit) σε μια φόρμα, αυτό θα έχει ως αποτέλεσμα την παραγωγή μιας γραμμής POST ή GET μαζί με κάποιες ακόμα γραμμές που περιέχουν σχετικές παραμέτρους. Για παράδειγμα, οι γραμμές

POST /cgi-bin/searcher/ HTTP/1.1

Content-Length: 46

userName=Darrel+Ince&email=d.r.ince@fizzer.com

Η πρώτη γραμμή ενημερώνει τον διακομιστή Web ότι μία φόρμα έχει κατατεθεί και το πρόγραμμα που πρέπει να εκτελεστεί για την επεξεργασία των φορμών βρίσκεται στον κατάλογο cgi-bin με το όνομα searcher.

Η επόμενη γραμμή δηλώνει ότι το περιεχόμενο που σχετίζεται με την εντολή (τα περιεχόμενα της φόρμας δηλαδή) έχει μέγεθος 46 χαρακτήρων. Η τρίτη γραμμή είναι το περιεχόμενο της εντολής. Περιγράφει τα περιεχόμενα που είχαν τα στοιχεία της φόρμας όταν πατήθηκε το κουμπί υποβολής. Στο παράδειγμα αυτό, το στοιχείο της φόρμας με το όνομα userName περιείχε το string 'Darrel Ince' (το κενό αντιπροσωπεύεται με το +) και το στοιχείο με το όνομα email περιείχε το string 'd.r.ince@fizzer.com'. Αυτό είναι γνωστό ως **string ερωτήματος**. Τα δεδομένα αυτά στην συνέχεια υφίστανται επεξεργασία από το πρόγραμμα. [4]

Κωδικοί Καταστάσεως

Όταν ένας διακομιστής Web ανταποκρίνεται στο αίτημα ενός browser απαντά με έναν κωδικό κατάστασης τριών ψηφίων και μ' ένα κείμενο που υποδεικνύει τι συνέβη με το αίτημα. Έχετε ήδη δει ένα παράδειγμα χρήσης του κωδικού κατάστασης όπου ο κώδικας 200 σημαίνει την επιτυχή εφαρμογή του GET. Αυτοί οι κωδικές κατάστασης διαιρούνται σε επιμέρους κατηγορίες. [4]

Οι κωδικοί κατάστασης που αρχίζουν με το 1 παρέχουν πληροφορίες διαφόρων ειδών στον πελάτη. Για παράδειγμα, μπορεί να παρέχει την πληροφορία ότι μια αλλαγή στο πρωτόκολλο που ζητήθηκε από τον πελάτη έγινε αποδεκτή. [4]

Οι κωδικοί κατάστασης που αρχίζουν με το 2 υποδεικνύουν ότι δεν υπάρχει πρόβλημα στην εκτέλεση του αιτήματος του πελάτη. Για παράδειγμα, ο κωδικός 202 υποδεικνύει ότι δεν υπάρχει πρόβλημα αλλά το αίτημα δεν έχει εκτελεστεί προς το παρόν. [4]

Οι κωδικοί κατάστασης που αρχίζουν με το 3 υποδεικνύουν ότι ο browser πρέπει να εκτελέσει κάποια άλλη ενέργεια ώστε το αίτημα να έχει επιτυχή έκβαση. Για παράδειγμα, ο κωδικός 301 υποδεικνύει ότι το αρχείο / πόρος που ζητήθηκε έχει μεταφερθεί μόνιμα σε κάποια άλλη τοποθεσία. [4]

Οι κωδικοί κατάστασης που αρχίζουν με το 4 υποδεικνύουν ότι κάτι πήγε στραβά. για παράδειγμα, ο πιο συχνός κωδικός αυτής της κατηγορίας είναι ο 404 που υποδεικνύει ότι ο αιτούμενος πόρος δεν μπορεί να βρεθεί. [4]

Οι κωδικοί κατάστασης που αρχίζουν με το 5 υποδεικνύουν ότι ο διακομιστής αντιμετώπισε κάποιο πρόβλημα. Για παράδειγμα, ο κωδικός κατάστασης 503 υποδεικνύει ότι η αιτούμενη υπηρεσία δεν ήταν δυνατόν να εκτελεστεί. Αυτό συμβαίνει συνήθως λόγω συντήρησης ρουτίνας του διακομιστή, ή λόγω υπερφόρτωσης του διακομιστή, με αποτέλεσμα την άρνηση οποιασδήποτε σύνδεσης. [4]

2.5. HTML

2.5.1. Τι είναι html?

Η **HTML** είναι το ακρωνύμιο των λέξεων **HyperText Markup Language** (γλώσσα μορφοποίηση υπερκειμένου) και είναι η βασική γλώσσα δόμηση σελίδων του *World Wide Web* (ή απλά ιστού: Web). Είναι μία γλώσσα προγραμματισμού. Χρησιμοποιείται για να σημαίνει ένα τμήμα κειμένου και να το κάνει να εμφανίζεται καλύτερα. Επιτρέπει την ενσωμάτωση ήχου και εικόνων στις web σελίδες. Αρχικά είχε κατασκευασθεί με σκοπό μόνο την μορφοποίηση κειμένου, αλλά μεγάλωσε και ενσωμάτωσε σχεδιαστικές τεχνικές κ.α.

Η γλώσσα χρησιμοποιεί ένα αριθμό από **tags** για την μορφοποίηση κειμένου, για την δημιουργία συνδέσμων (*links*) μετάβασης ανάμεσα των σελίδα, για την εισαγωγή εικόνων, ήχου κ.α. Όταν ένας Web Browser ανοίγει ένα αρχείο HTML τα στοιχεία (*tags*) μεταφράζονται σε κατάλληλα χαρακτηριστικά με αποτελέσματα στην εμφάνιση και στην λειτουργικότητα της συγκεκριμένης σελίδας.[8]

2.5.2. Εκδόσεις HTML

Το 1990 ο Tim Berners-Lee από το Cern, το εργαστήριο φυσικής της Γενεύης, δημιούργησε ένα νέο πρωτόκολλο με το οποίο θα μπορούσαν να μεταφέρονται κάθε είδος αρχείων και αντικειμένων μέσα από το Internet. Το πρωτόκολλο αυτό ονομάστηκε HTTP (HyperText Transfer Protocol) και σηματοδότησε την αρχή του WWW όπως το ξέρουμε σήμερα. Οι σελίδες που ήταν η βάση του WWW ήταν γραμμένες στην πρώτη έκδοση της γλώσσα HTML. [8]

Το 1994 αναπτύσσεται το πρότυπο HTML 2.0 από ένα διεθνή οργανισμό (Internet Engineering Task Force). Η επόμενη έκδοση η 3.0 δεν έγινε αποδεκτή από τις Microsoft και Netscape οπότε γρήγορα αντικαταστάθηκε από την έκδοση 3.2 (1996). Η τελευταία περιελάμβανε πολλές από τις σημάνσεις (tags) που είχαν εισάγει οι δύο εταιρίες. Η έκδοση 4.0 παρουσιάστηκε τον Ιούνιο του 1997. [8]

2.5.3. Γλώσσες Προγραμματισμού στο Internet

Οι ενδογενείς αδυναμίες της HTML, μιας κατεξοχήν γλώσσας μορφοποίησης υπερκειμένου, σύντομα οδήγησαν στην ανάπτυξη γλωσσών προγραμματισμού στο Internet. Η χρήση των γλωσσών προγραμματισμού κρίθηκε απαραίτητη αφού η εξάπλωση του Internet και η χρήση των σελίδων HTML για αλληλεπίδραση με τους χρήστες (δυναμική αποστολή και λήψη δεδομένων) δεν μπορούσε να πραγματοποιηθεί μέσω της HTML. Αρχικά, αναπτύχθηκε από την εταιρεία Netscape η γλώσσα **JavaScript** η οποία, όπως και η HTML μεταφράζεται από τον Web Browser κατά την εμφάνιση της σελίδας. Η Microsoft ανέπτυξε απ' την πλευρά της μια δική της έκδοση της γλώσσας JavaScript την οποία ονόμασε **JScript** και μια έκδοση της γλώσσας Basic ειδικά για το Internet που ονόμασε **VBScript**. [8]

Έτσι πλέον με την χρήση των παραπάνω γλωσσών είναι δυνατό να ελεγχθούν και να προγραμματιστούν όλα σχεδόν τα αντικείμενα που μπορεί να περιέχει μία ιστοσελίδα και με τη χρήση είτε εντολών διαδικαστικού χαρακτήρα, είτε εντολών διακλάδωσης, όπως για παράδειγμα η δομή [if ... then... else...]. Η χρήση τέτοιου είδους "προγραμμάτων" ή όπως ονομάζονται "scripts" είναι διαδικασία που έχει μεγάλες απαιτήσεις σε προγραμματισμό και απευθύνεται σε προγραμματιστές και μόνο. [8]

2.5.4. Δημιουργία αρχείων HTML

Η δημιουργία αρχείων HTML είναι πολύ απλή. Αρκεί να "τρέξουμε" έναν οποιοδήποτε διορθωτή κειμένου *text*, όπως το *Notepad* των Windows. Να γράψουμε τον κώδικα HTML που επιθυμούμε και να το αποθηκεύσουμε (σώσουμε) σε ένα αρχείο με κατάληξη **.htm** ή **.html**. Συνήθως αποθηκεύουμε με επέκταση **.html** όταν θα δημοσιεύσουμε τα αρχεία μας σε διακομιστή Unix. Αντίθετα ο εξυπηρετητής Windows NT καταλαβαίνει τα αρχεία και σαν **.htm** και σαν **.html**. [8]

2.6. JavaScript.

2.6.1. Τι είναι JavaScript;

Είναι μια γλώσσα συγγραφής σεναρίων (scripting language) για την προσθήκη διαδραστικότητας (interactivity) σε ιστοσελίδες. Δημιουργήθηκε από την εταιρεία Netscape με αρχικό όνομα LiveScript και είναι πολύ δημοφιλής στους δημιουργούς ιστοσελίδων καθώς είναι απλή στη σύνταξή της και υποστηρίζεται απ' όλους τους δημοφιλείς φυλλομετρητές. Ανταγωνιστής της είναι η γλώσσα συγγραφής σεναρίων VBScript της εταιρείας Microsoft.

Μια άλλη προσέγγιση στον προγραμματισμό από την πλευρά του πελάτη είναι η γλώσσα προγραμματισμού JavaScript. Για τον προγραμματισμό σε JavaScript πρέπει να εισάγετε δηλώσεις προγραμματισμού *απευθείας* στον κώδικα HTML. Αυτός ο κώδικας αλληλεπιδρά με τον browser, για παράδειγμα παράγει οπτικά εφέ, όπως rollover. Αυτό βρίσκεται σε αντίθεση με τα applet όπου ο κώδικας βρίσκεται στον διακομιστή, υπάρχει μόνο αναφορά για αυτόν στον κώδικα HTML, φορτώνεται στον πελάτη χρησιμοποιώντας την αναφορά και εκτελείται. [8]

2.6.2. Τρόπος Λειτουργίας της JavaScript

Η JavaScript είναι ενσωματωμένη σ' ένα αρχείο HTML ανάμεσα σε tags SCRIPT. Παρακάτω δίνεται ένα παράδειγμα ενός πολύ απλού προγράμματος:

```
<SCRIPT LANGUAGE = "JavaScript">
document.writeln("<H3> Hello there");
</SCRIPT>
```

Ο κώδικας βρίσκεται μέσα στα <SCRIPT> </SCRIPT> και αποτελείται από μία απλή δήλωση που εμφανίζει σε μια ιστοσελίδα το μήνυμα 'Hello there' στο σημείο όπου υπάρχει δήλωση μέσα στο HTML αρχείο. Το μήνυμα εμφανίζεται σε επικεφαλίδα μεγέθους 3, χρησιμοποιώντας το κατάλληλο tag. Το αντικείμενο document αντιπροσωπεύει το αρχείο HTML στο οποίο περικλείεται ο κώδικας. Η μέθοδος writeln του αντικειμένου document γράφει ένα string στον HTML κώδικα που θα εμφανιστεί. [8]

Η JavaScript μοιάζει πολύ με μια κανονική γλώσσα προγραμματισμού καθώς περιλαμβάνει:

- Αριθμητικούς τύπους δεδομένων, όπως ακέραιους, καθώς και strings.
- Ένα μεγάλο αριθμό τελεστών που μπορούν να εφαρμοστούν στους διάφορους τύπους δεδομένων.
- Δομές ελέγχου, όπως οι if και while.

- Συναρτήσεις: κομμάτια κώδικα που μπορούν να κληθούν επανειλημμένα και ανεξάρτητα όποτε χρειάζεται να εκτελεστεί η συγκεκριμένη λειτουργία που μέθοδος υλοποιεί.
- Πίνακες που περιέχουν συλλογές δεδομένων.

Αυτό που την ξεχωρίζει πάντως είναι η ικανότητα, χρησιμοποιώντας την τεχνολογία της δυναμικής HTML (DHTML), να αλληλεπιδρά με τα στοιχεία μίας ιστοσελίδας, για παράδειγμα μπορεί να έχει πρόσβαση στα στοιχεία μιας φόρμας HTML ή ν' αλλάξει την εμφάνιση ενός στοιχείου κειμένου. [8]

2.7. CSS

Η CSS αρχικά ξεκίνησε από την HTML. Τα τελευταία χρόνια, η πολυπλοκότητα του HTML αυξήθηκε δραματικά. Ειδικότερα, πολλά στοιχεία προστέθηκαν στην HTML για την υποστήριξη του στυλ, όπως τα <CENTER> και τα .

Επίσης, φάνηκε ότι η πρόσθεση όλο και περισσότερων στοιχείων στην HTML δεν θα κατέληγε σε μία μακροπρόθεσμη και βιώσιμη λύση επειδή οδηγεί σε πολύπλοκες ιστοσελίδες, που είναι δύσκολο να διαβαστούν και να συντηρηθούν, όπως και το ότι είναι αδικαιολόγητα μεγάλες. Το W3C ανταποκρίθηκε παρουσιάζοντας μία γλώσσα φύλλου στυλ, το CSS. Το CSS διαχωρίζει το στυλ από το περιεχόμενο της σελίδας. [9]

Μέχρι σήμερα υπάρχουν δύο εκδόσεις CSS: CSS1 και CSS2.

2.8. PHP

Σκοπός αυτής της πτυχιακής είναι να επιτρέψει να δημιουργήσουμε πραγματικές δυναμικές Web τοποθεσίες. Αν έχουμε δημιουργήσει Web τοποθεσίες χρησιμοποιώντας απλή HTML, θα καταλάβουμε τα όρια αυτής της προσέγγισης. Τα στατικά περιεχόμενα μίας Web τοποθεσίας, είναι απλώς στατικά και παραμένουν ίδια, εκτός και αν ενημερωθούν άμεσα. Οι χρήστες δεν μπορούν να συνδιαλέγονται με την τοποθεσία με κάποιο τρόπο που να είναι ουσιαστικός.

Χρησιμοποιώντας μια γλώσσα, όπως την PHP και μια βάση δεδομένων, όπως τη MySQL, υπάρχει η δυνατότητα να γίνουν οι τοποθεσίες δυναμικές, δηλαδή να είναι προσαρμόσιμες και να περιέχουν ενημερωμένες πληροφορίες.[1]

2.8.1. Τι είναι η PHP;

Γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Όσον αφορά τον κώδικά της, έχει τον ίδιο τρόπο σύνταξης με τη γλώσσα C++, περιλαμβάνοντας όμως ειδικές πρόσθετες λειτουργίες (functions) για τις ανάγκες του web. Μια σελίδα PHP περνά από επεξεργασία από συμβατό web server (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό

περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.

Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. *.php, *.php4 κ.ά.). 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 ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται κώδικα PHP. Ο server Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με λειτουργικό σύστημα Linux και Windows, υποστηρίζει εξ ορισμού επεξεργασία κώδικα PHP.[2]

2.8.2. Τι μπορεί να κάνει η PHP;

Η PHP επικεντρώνεται κυρίως στο server-side scripting, έτσι μπορεί να γίνει οτιδήποτε που ένα άλλο CGI πρόγραμμα μπορεί να κάνει, όπως να μαζέψει δεδομένα, να παράγει δυναμικό περιεχόμενο σελίδων, ή να στείλει και να πάρει cookies. Αλλά η PHP μπορεί να κάνει πολύ περισσότερα. [7]

Υπάρχουν τρεις κύριοι τομείς που χρησιμοποιείται ένα PHP script.

- **Server-side scripting.** Αυτό είναι το πιο παραδοσιακό και το κύριο πεδίο για την PHP. Χρειάζονται τρία πράγματα για να δουλέψει αυτό. Τον PHP μεταγλωττιστή (parser) (CGI ή server module), ένα webserver (εξυπηρετητή σελίδων) και ένα web browser ("φυλλομετρητή"). Πρέπει να τρέξει ο webserver, με μια συνδεδεμένη εγκατάσταση της PHP. Μπορούν να προσπελαστούν τα αποτελέσματα του PHP προγράμματος με ένα web browser, βλέποντας την σελίδα PHP μέσα από τον server. [7]
- **Command line scripting.** Μπορείτε να δημιουργηθεί ένα PHP script για να τρέχει χωρίς server ή browser. Χρειάζεται μόνο τον PHP μεταγλωττιστή για να χρησιμοποιηθεί με αυτόν τον τρόπο. Αυτός ο τύπος είναι ιδανικός για script που εκτελούνται συχνά με τη χρήση της cron (σε *nix ή Linux) ή με τον Task Scheduler (στα Windows). Αυτά τα script μπορούν επίσης να χρησιμοποιηθούν για απλές εργασίες επεξεργασίας κειμένου. [7]
- **Εγγραφή client-side GUI εφαρμογών** (Γραφικά περιβάλλοντα χρηστών). Η PHP ίσως να μην είναι η πιο καλή γλώσσα για να γράψει κανείς παραθυρικές εφαρμογές, αλλά αν γνωρίζει κάποιος PHP πολύ καλά και θέλει να χρησιμοποιήσει κάποια προχωρημένα χαρακτηριστικά της PHP στις client-side εφαρμογές, μπορεί επίσης να χρησιμοποιήσει το PHP-GTK για αυτού του είδους τα προγράμματα. Υπάρχει επίσης η δυνατότητα εγγραφής cross-platform εφαρμογών με αυτόν τον τρόπο. Το PHP-GTK είναι μια επέκταση της PHP και δεν συμπεριλαμβάνεται στην κύρια διανομή. [7]

2.8.3. Πλεονεκτήματα της PHP

Η PHP μπορεί να χρησιμοποιηθεί σε όλα τα κύρια λειτουργικά συστήματα, συμπεριλαμβανομένου του Linux, πολλών εκδοχών του SCO-Uinx (HP-UX, Solaris και OpenBSD), Microsoft Windows, Mac OS X, RISC OS και πιθανώς σε άλλα. Η PHP υποστηρίζει επίσης τους Apache, Microsoft Internet Information Server, Personal Web Server, Netscape και iPlanet servers, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd, και πολλούς άλλους webserver. Για την πλειοψηφία των server η PHP έχει ένα module, για τους υπόλοιπους η PHP μπορεί να λειτουργήσει ως ένας CGI επεξεργαστής. [1]

Έτσι με την PHP υπάρχει η ελευθερία επιλογής ενός λειτουργικού συστήματος και ενός web server. Επιπλέον, υπάρχει επίσης η δυνατότητα χρήσης συναρτησιακού (procedural) ή αντικειμενοστραφή (object oriented) προγραμματισμού ή μια ανάμειξη τους. Αν και η παρούσα έκδοση δεν υποστηρίζει όλα τα πρότυπα χαρακτηριστικά, μεγάλες βιβλιοθήκες κώδικα και μεγάλες εφαρμογές (συμπεριλαμβανομένης και της βιβλιοθήκης PEAR) είναι γραμμένες μόνο με αντικειμενοστραφή κώδικα. [1]

Με την PHP δεν είναι κάποιος περιορισμένος να εξάγει HTML. Οι δυνατότητες της PHP συμπεριλαμβάνουν την εξαγωγή εικόνων, αρχείων PDF, ακόμη και ταινίες Flash (χρησιμοποιώντας τα libswf και Ming) παράγονται αμέσως. Μπορεί επίσης να εξαχθεί εύκολα οποιοδήποτε κείμενο όπως XHTML και οποιοδήποτε άλλο XML αρχείο. Η PHP μπορεί να δημιουργεί αυτόματα αυτά τα αρχεία και να τα αποθηκεύει στο σύστημα αρχείων, αντί να τα εκτυπώνει, αποτελώντας έτσι μια server-side cache για το δυναμικό σας περιεχόμενο. [1],[2]

Ένα από τα πιο δυνατά και σημαντικά χαρακτηριστικά της PHP είναι η υποστήριξη που έχει για ένα μεγάλο σύνολο βάσεων δεδομένων. Η συγγραφή μιας σελίδας που υποστηρίζει βάσεις δεδομένων είναι εξαιρετικά απλή. Οι εξής βάσεις δεδομένων υποστηρίζονται μέχρι στιγμής:

Adabas D	Ingres	Oracle (OCI7 and OCI8)
dBase	InterBase	Ovrimos
Empress	FrontBase	PostgreSQL
FilePro (read-only)	mSQL	Solid
Hyperwave	Direct MS-SQL	Sybase
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

Πίνακας 3 Σύνολο Βάσεων Δεδομένων [7]

Έχουμε επίσης μια αφαιρετική επέκταση DBX βάσεων δεδομένων (DBX database abstraction extension) που επιτρέπει διάφανα να χρησιμοποιηθεί οποιαδήποτε βάση δεδομένων υποστηρίζεται από αυτή την επέκταση. Επιπλέον η PHP υποστηρίζει το ODBC, το Open Database Connection standard (Ανοιχτό

πρότυπο Σύνδεσης Βάσεων δεδομένων) έτσι μπορεί να συνδεθεί σε οποιαδήποτε βάση δεδομένων που υποστηρίζει αυτό το παγκόσμιο πρότυπο. [7]

Η PHP έχει επίσης υποστήριξη για επικοινωνία με άλλες υπηρεσίες χρησιμοποιώντας πρωτόκολλα όπως LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (στα Windows) και αμέτρητα άλλα. Υπάρχει επίσης η δυνατότητα να ανοιχθούν raw network sockets και να αλληλεπιδράσουν με οποιοδήποτε άλλο πρωτόκολλο. Η PHP έχει ακόμη υποστήριξη για την περίπλοκη ανταλλαγή δεδομένων WDDX μεταξύ σχεδόν όλων των Web programming γλωσσών. Μιλώντας για δια-επικοινωνία, η PHP υποστηρίζει instantiation αντικειμένων Java και τα χρησιμοποιεί διάφανα σαν αντικείμενα PHP. Μπορεί επίσης να χρησιμοποιηθεί η CORBA επέκταση για να προσπελαστούν remote (απομακρυσμένα) αντικείμενα.

Η PHP έχει εξαιρετικά χρήσιμα χαρακτηριστικά επεξεργασίας κειμένων, από την POSIX επέκταση ή τις Perl regular expressions μέχρι XML parsing αρχείων. Για τη μεταγλώττιση και την πρόσβαση αρχείων XML, υποστηρίζουμε τα πρότυπα SAX και DOM. Μπορεί να χρησιμοποιηθεί η XSLT επέκταση για να μετατρέπονται τα XML αρχεία σε άλλες μορφές. [1]

Καθώς χρησιμοποιείται η PHP στον τομέα του ecommerce, βρίσκονται οι Cybercash payment, CyberMUT, VeriSign Payflow Pro και CCVS συναρτήσεις χρήσιμες για τα online προγράμματα πληρωμής. [1]

Τελευταίο αλλά σημαντικό, έχουμε πολλές άλλες ενδιαφέρουσες επεκτάσεις, τις mmoGoSearch search engine συναρτήσεις, πολλά εργαλεία συμπίεσης (gzip, bz2), μετατροπές ημερολογίου, μεταφράσεις... [1]

2.9. MySQL

2.9.1. Τι είναι MySQL?

Η MySQL είναι ένας SQL Database Server. Ένας server είναι ένα κομμάτι λογισμικού (software) που τρέχει (εκτελείται) στο παρασκήνιο σ' έναν υπολογιστή. Θα ασχοληθούμε με δύο διαφορετικούς servers : τους HTTP servers και τους MySQL servers. Μια βάση δεδομένων (database) είναι μια δομημένη συλλογή από δεδομένα. Μια βάση δεδομένων είναι έτσι δομημένη ώστε να μπορούμε να βρίσκουμε πληροφορίες εύκολα και γρήγορα ανάλογα με το είδος των δεδομένων που περιέχει. [1]

Η MySQL είναι ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων (RDBMS, Relational Database Management System). Ένα RDBMS ορίζει σχέσεις (relations) ανάμεσα σε δομές δεδομένων (data structures). Τα αρχικά *SQL* σημαίνουν *Structured Query Language*, δηλ. *Δομημένη Γλώσσα Ερωτημάτων ή Ερωταποκρίσεων*. Η γλώσσα SQL μάς δίνει τη δυνατότητα να μιλάμε φιλικά με τον server ώστε να πάρουμε πληροφορίες από τη βάση δεδομένων μας και να μπορέσουμε να τις διαχειριστούμε. [1]

2.9.2. Πλεονεκτήματα της MySQL

Η MySQL είναι γρήγορη, πολύ αξιόπιστη και εύκολη στη χρήση. Επίσης, η MySQL αποτελεί την επιλογή της γλώσσας PHP και πολλές εταιρείες web hosting παρέχουν την MySQL και την PHP ως ένα στάνταρτ πακέτο. Το μεγαλύτερο πλεονέκτημα της MySQL είναι ότι είναι δωρεάν (free) και ανοικτού κώδικα (open source). Επίσης, μπορεί να εκτελεσθεί σε πολλές πλατφόρμες, όπως Windows, Linux, FreeBSD, Solaris κ.ά. [1]

3. Τρόπος Λειτουργίας της PHP

3.1. Χρήση της PHP στις δυναμικές εφαρμογές

Η PHP, της οποίας τα αρχικά αντιπροσωπεύουν το "PHP: Hypertext Preprocessor" είναι μια ευρέως χρησιμοποιούμενη, ανοιχτού κώδικα, γενικού σκοπού scripting γλώσσα προγραμματισμού, η οποία είναι ειδικά κατάλληλη για ανάπτυξη εφαρμογών για το Web και μπορεί να ενσωματωθεί στην HTML. [7]

Ένα εισαγωγικό παράδειγμα

```
<html>
  <head>
    <title>Example</title>
  </head>
  <body>

    <?php
      echo "Hi, I'm a PHP script!";
    ?>

  </body>
</html>
```

Σχ. 4 Απλό Παράδειγμα PHP

Παρατηρούμε πως αυτό είναι διαφορετικό από ένα script γραμμένο σε άλλες γλώσσες προγραμματισμού όπως η Perl ή η C : Αντί να γράφουμε ένα πρόγραμμα με πολλές εντολές για να εξάγουμε HTML, γράφουμε ένα HTML script με κάποιο ενσωματωμένο κώδικα για να κάνει κάτι (σε αυτή την περίπτωση, να εμφανίζει κάποιο κείμενο). Ο κώδικας PHP είναι εσώκλειστος σε ειδικά tags (ετικέτες) αρχής και τέλους που σας επιτρέπουν να μεταφέρεστε μέσα και έξω από το "PHP mode" (PHP τρόπο λειτουργίας). Αυτό που διαχωρίζει την PHP από κάτι σαν client-side Javascript είναι ότι ο κώδικας εκτελείται στον server (εξυπηρετητή). Αν έχουμε ένα script σαν το παραπάνω στον server σας, ο client θα έπαιρνε τα αποτελέσματα της εκτέλεσης αυτού του script, χωρίς να υπάρχει κανένας τρόπος να καταλάβει τι κώδικας υπάρχει από κάτω.

Όταν η PHP μεταγλωττίζει (parses) ένα αρχείο, απλά κάνει ένα πέρασμα στο κείμενο του αρχείου μέχρι να συναντήσει ένα από τα ειδικά tags που της λένε να αρχίσει να μεταφράζει το κείμενο ως κώδικα PHP. Ο parser (μεταγλωττιστής) τότε εκτελεί ολόκληρο τον κώδικα που βρίσκει, μέχρι να συναντήσει το επόμενο PHP tag κλεισίματος, το οποίο λέει στον parser να αρχίσει να κάνει ξανά, απλά ένα πέρασμα στο κείμενο. Αυτός είναι ο μηχανισμός που μας επιτρέπει να προσθέσουμε PHP κώδικα μέσα σε HTML: οτιδήποτε βρίσκεται έξω από τα tags της PHP μένει τελείως μόνο, ενώ οτιδήποτε μέσα μεταγλωττίζεται ως κώδικας.

Υπάρχουν τέσσερα σύνολα από tags που μπορούν να χρησιμοποιηθούν για να δηλώσουμε τα κομμάτια που έχουν κώδικα σε PHP. Από αυτά, μόνο δύο (<?php. . .?> and <script language="php">. . .</script>) είναι πάντα διαθέσιμα. Τα άλλα μπορούν να ενεργοποιηθούν και να απενεργοποιηθούν από το php.ini αρχείο ρυθμίσεων. Ενώ τα short-form tags και τα tags που μοιάζουν με αυτά της ASP

μπορεί να είναι βολικά, δεν είναι τόσο portable όσο οι μακρύτερες εκδόσεις. Επίσης, αν σκοπεύετε να προσθέσετε PHP κώδικα σε XML ή XHTML, θα χρειαστεί να χρησιμοποιήσετε την `<?php. . .?>` φόρμα για να προσαρμοστεί στην XML. [7]

1.

```
<?php echo("if you want to serve XHTML or XML documents, do like this\n");?>
```

`<?php. . .?>`, είναι και ο προτιμότερος, καθώς επιτρέπει τη χρήση της PHP σε κώδικα συμβατό με την XML όπως η XHTML.

2.

```
<? echo ("this is the simplest, an SGML processing instruction\n"); ?>
<?= expression ?> This is a shortcut for "<? echo expression ?>"
```

Δεν είναι πάντα διαθέσιμος. Τα σύντομα tags είναι διαθέσιμα μόνο όταν έχουν ενεργοποιηθεί. Αυτό μπορεί να γίνει μέσω της συνάρτησης **short_tags()** (μόνο στην PHP 3), ενεργοποιώντας την επιλογή ρύθμισης `short_open_tag` στο αρχείο ρυθμίσεων της PHP, ή κάνοντας `compile` την PHP με την επιλογή `--enable-short-tags` στο **configure**. Ακόμη και αν είναι ενεργοποιημένο ως προεπιλογή στο `php.ini-dist`, η χρήση των short tags δεν προτιμάται. [7]

3.

```
<script language="php">
    echo ("some editors (like FrontPage) don't like processing instructions");
</script>
```
4.

```
<% echo ("You may optionally use ASP-style tags"); %>
<%= $variable; # This is a shortcut for "<% echo . . ." %>
```

Είναι διαθέσιμος μόνο αν τα ASP-style tags έχουν ενεργοποιηθεί χρησιμοποιώντας την `asp_tags` επιλογή ρυθμίσεων. [7]

Με την ετικέτα `echo` ” ”, έχουμε ένα απλό αποτέλεσμα, τυπώνει την συμβολοσειρά που έχει περάσει σε αυτήν, στον browser. Το ερωτηματικό χρησιμοποιείται για να ξεχωρίζει τις εντολές της PHP, όπως μια τελεία χρησιμοποιείται για να ξεχωρίζει τις προτάσεις της γλώσσας. [7]

Τα ονόματα των μεταβλητών στην PHP ξεκινούν με \$, όπως \$name και αντιπροσωπεύει τα δεδομένα που πληκτρολογεί ο χρήστης.

3.2. Προκαθορισμένες μεταβλητές

Η PHP παρέχει έναν μεγάλο αριθμό από προκαθορισμένες μεταβλητές σε οποιοδήποτε script τρέχει. Αρκετές από αυτές τις μεταβλητές πάντως, δεν μπορούν να τεκμηριωθούν εντελώς αφού εξαρτώνται από τον server στον οποίο τρέχουν, την έκδοση και το setup του server, καθώς και από άλλους παράγοντες. Μερικές από αυτές τις μεταβλητές δε θα είναι διαθέσιμες όταν η PHP τρέχει σε [command line](#) (μια μέθοδος αλληλεπίδρασης με έναν υπολογιστή μέσω ενός τερματικού κειμένου, όπου οι εντολές εισάγονται ως γραμμές κειμένου από ένα πληκτρολόγιο, και η παραγωγή παραλαμβάνεται επίσης ως κείμενο.).

Από την έκδοση 4.1.0 και μετά, η PHP παρέχει ένα επιπρόσθετο σύνολο από προκαθορισμένους arrays που περιέχουν μεταβλητές από τον web server (αν είναι δυνατό), το environment (περιβάλλον), και αυτά που εισάγει ο χρήστης. Αυτοί οι νέοι arrays είναι μάλλον ιδιαίτεροι από την άποψη ότι είναι αυτόματα global--π.χ., αυτόματα διαθέσιμοι για κάθε εμβέλεια. Για αυτό το σκοπό, είναι συχνά γνωστοί και ως 'autoglobals' ή 'superglobals'. (Δεν υπάρχει μηχανισμός στην PHP για superglobals που μπορεί να ορίσει ο χρήστης.) Οι superglobals παρατίθενται παρακάτω. Επίσης, θα παρατηρήσουμε πώς οι παλιότερες προκαθορισμένες μεταβλητές (*\$HTTP_*_VARS*) υπάρχουν ακόμη. Από την PHP 5.0.0, τα μεγάλα προκαθορισμένα σταθερά array μπορούν να απενεργοποιηθούν με το register_long_arrays directive (**boolean** εκφράζει μια αληθινή τιμή που μπορεί να είναι είτε **TRUE** είτε **FALSE**). [7]

Οι PHP Superglobals

- *\$GLOBALS*: Περιέχουν μια αναφορά σε κάθε μεταβλητή που είναι διαθέσιμη μέσα στην global εμβέλεια του script. Τα κλειδιά αυτού του array είναι τα ονόματα των global μεταβλητών. Η *\$GLOBALS* υπάρχει από την PHP 3.
- *\$_SERVER*: Είναι οι μεταβλητές που ορίζονται από τον web server ή διαφορετικά είναι άμεσα συνδεδεμένες με το περιβάλλον εκτέλεσης του τρέχοντος script. Είναι ανάλογες με τον παλιό *\$HTTP_SERVER_VARS* array.
- *\$_GET*: Είναι οι μεταβλητές που παρέχονται στο script μέσω του HTTP GET. Είναι ανάλογες με τον παλιό *\$HTTP_GET_VARS* array .
- *\$_POST*: Είναι οι μεταβλητές που παρέχονται στο script μέσω του HTTP POST. Είναι ανάλογες με τον παλιό *\$HTTP_POST_VARS* array . Γενικότερα περιέχει της form τις μεταβλητές που στέλνονται μέσω της POST.
- *\$_COOKIE*: Είναι οι μεταβλητές που παρέχονται στο script μέσω της HTTP cookies. Είναι ανάλογες με τον παλιό *\$HTTP_COOKIE_VARS* array. Δεν μπορούμε να κάνουμε μια global *\$_COOKIE* για να έχει πρόσβαση μέσα στις λειτουργίες ή τις μεθόδους, όπως κάνουμε με *\$HTTP_COOKIE_VARS*.
- *\$_FILES*: Είναι οι μεταβλητές που παρέχονται στο script μέσω του HTTP post file uploads. Είναι ανάλογες με τον *\$HTTP_POST_FILES* array.
- *\$_ENV*: Είναι οι μεταβλητές που παρέχονται στο script μέσω του environment. Είναι ανάλογες με τον παλιό *\$HTTP_ENV_VARS* array .

- `$_SESSION` :Είναι οι μεταβλητές που είναι προς το παρόν εγγεγραμμένες σε ένα session ενός script. Είναι ανάλογες με τον παλιό `$HTTP_SESSION_VARS` array[7]

3.3. HTTP αναγνώριση με την PHP

3.3.1. header()

Η HTTP αναγνώριση με την PHP είναι διαθέσιμη μόνο όταν εκτελείται σαν Apache module και έτσι δεν είναι διαθέσιμη στην CGI έκδοση. Σε ένα Apache module PHP script, είναι δυνατόν να χρησιμοποιηθεί η **header()** συνάρτηση. Η συνάρτηση header στέλνει έναν ακατέργαστο HTTP header[7]. Η γενική της περιγραφή είναι:

```
int header ( string string [, bool replace [, int http_response_code]] )
```

Η **header()** χρησιμοποιείται για να σταλθούν ακατέργαστοι HTTP headers. Η προαιρετική παράμετρος *replace* δείχνει αν ο header πρέπει να αντικαταστήσει έναν προηγούμενο παρόμοιο header, ή να προσθέσει έναν δεύτερο header του ίδιου τύπου. Η προκαθορισμένη ενέργεια είναι η αντικατάσταση του, αλλά αν περάσετε ως δεύτερη παράμετρο το FALSE μπορούμε να στέλνουμε πολλαπλούς headers του ίδιου τύπου.

Υπάρχουν δύο ιδιαίτερες κλήσεις για headers. Η πρώτη είναι ένας header που αρχίζει με το string "HTTP/" το οποίο θα χρησιμοποιηθεί για να βρεθεί το HTTP status code που πρέπει να σταλεί. Για παράδειγμα, αν έχετε παραμετροποιήσει τον Apache να χρησιμοποιεί ένα PHP script για να χειρίζεται requests για χαμένα αρχεία (χρησιμοποιώντας την *ErrorDocument* ντιρεκτίβα), ίσως θέλετε να βεβαιωθείτε ότι το script σας παράγει τον κατάλληλο status code. [7]

```
<?php
header("HTTP/1.0 404 Not Found");
?>
```

Σχ. 5 Παράδειγμα Header() με το string "HTTP/"

Η δεύτερη ειδική περίπτωση είναι η "Location:" header. Δεν στέλνει μόνο τον header πίσω στον browser, αλλά επίσης επιστρέφει ένα *REDIRECT* (302) status code στον browser εκτός και αν κάποιος 3xx status code έχει ήδη τεθεί. [7]

```

<?php
header("Location: http://www.example.com/"); /* Redirect browser */

/* Make sure that code below does not get executed when we redirect. */
exit;
?>

```

Σχ. 6 Παράδειγμα Header() με το string "HTTP/"

3.3.2. Χρησιμοποιώντας τις require() και include()

Η PHP παρέχει δύο πολύ απλές, αλλά χρήσιμες εντολές, για να μπορούμε να ξαναχρησιμοποιήσουμε οποιοδήποτε είδος κώδικα. Χρησιμοποιώντας την εντολή require() ή include(), μπορούμε να φορτώσουμε ένα αρχείο στο PHP script. Το αρχείο μπορεί να παρέχει οτιδήποτε θα πληκτρολογήσαμε κανονικά σε ένα script, συμπεριλαμβανόμενων PHP εντολών, κειμένου, HTML ετικετών, PHP συναρτήσεων ή κλάσεων. Αυτές οι εντολές δουλεύουν παρόμοια με τα include από την πλευρά του διακομιστή, που προσφέρονται από πολλούς Web Διακομιστές και από τις εντολές #Include της C ή της C ++.[7]

Παράδειγμα με την εντολή require():

Ο παρακάτω κώδικας αποθηκεύεται σε ένα αρχείο που ονομάζεται test.php:

```

<?
Echo "This is a test!!"
?>

```

θα είναι στην μορφή `require(test.php);`

Η PHP δεν κοιτάζει την επέκταση του αρχείου στο απαιτούμενο αρχείο. Αυτό σημαίνει ότι μπορούμε να ονομάσουμε το αρχείο μας οτιδήποτε επιλέξουμε, εφόσον δεν πρόκειται να το καλέσουμε κατευθείαν. Όταν χρησιμοποιούμε την require() για να φορτώσουμε το αρχείο, τελικά θα γίνει μέρος ενός PHP αρχείου και θα εκτελεστεί έτσι. Κανονικά, δεν θα γινόταν επεξεργασία των PHP εντολών εάν βρισκόταν σε ένα αρχείο που ονομαζόταν, για παράδειγμα, page.html. Η PHP συνήθως καλείται μόνο για να αναλύσει αρχεία με ορισμένες επεκτάσεις, όπως .php. Αν φορτώσουμε την page.html μέσω μιας εντολής require(), θα γίνει επεξεργασία του κώδικα PHP που βρίσκεται μέσα σε αυτό το αρχείο. Συνεπώς μπορούμε να χρησιμοποιούμε οποιαδήποτε επέκταση προτιμάμε για τα συμπεριλάβουμε αρχεία, αλλά θα ήταν ιδέα να δοκιμάσουμε να παραμείνουμε σε μία λογική σύμβαση, όπως την .inc.

Αν τα αρχεία τελειώνουν σε .inc ή σε κάποια άλλη μη τυπική επέκταση, αποθηκευθούν στο Web δέντρο και οι χρήστες τα φορτώνουν κατευθείαν στο browser, θα μπορούν να δουν τον κώδικα σε απλό κείμενο, συμπεριλαμβανομένων και κωδικών πρόσβασης. συνεπώς είναι σημαντικό να αποθηκεύουμε συμπεριλαμβανόμενα αρχεία έξω από το δέντρο των εγγράφων ή να χρησιμοποιούμε τις τυπικές επεκτάσεις. [7]

3.4. Χρησιμοποιώντας συναρτήσεις

Στις περισσότερες γλώσσες προγραμματισμού υπάρχουν συναρτήσεις. Χρησιμοποιούνται για να ξεχωρίζουν κώδικα που εκτελεί μια καλά ορισμένη διαδικασία. Αυτό κάνει τον κώδικα ευκολότερο να διαβαστεί και να μας επιτρέπει να ξαναχρησιμοποιούμε τον κώδικα κάθε φορά που θέλουμε να κάνουμε την ίδια διαδικασία. Μια συνάρτηση είναι μια ανεξάρτητη λειτουργική μονάδα κώδικα που περιγράφει το περιβάλλον που κάνει την κλήση, εκτελεί τη διαδικασία και προαιρετικά επιστρέφει ένα αποτέλεσμα. Έχουμε ήδη διάφορες συναρτήσεις που θα χρησιμοποιήσουμε στις εφαρμογές μας και θα αναφέρουμε παρακάτω. [7]

3.4.1. Mail συναρτήσεις

Η συνάρτηση **mail()** επιτρέπει την αποστολή e-mail. Η γενική του περιγραφή είναι: `bool mail (string to, string subject, string message [, string additional headers [, string additional_parameters]])` [7]

Η `mail()` αυτόματα στέλνει με `mail` το μήνυμα όπως ορίζεται στο `message` στον παραλήπτη όπως ορίζεται στο `to`. Πολλαπλοί παραλήπτες μπορούν να οριστούν με την προσθήκη ενός κόμματος ανάμεσα σε κάθε διεύθυνση στο `to`. Email με συνημμένα αρχεία και ειδικούς τύπους περιεχομένου μπορούν να αποσταλούν με την χρήση αυτής της συνάρτησης. Η **mail()** επιστρέφει **TRUE** εάν το `mail` παραδόθηκε επιτυχώς, **FALSE** σε αντίθετη περίπτωση. [7]

```
<?php
mail("joecool@example.com", "My Subject", "Line 1\nLine 2\nLine 3");
?>
```

Σχ. 7 Ένα απλό παράδειγμα της `mail()`.

Εάν περαστεί ένα `fourth string argument`, αυτό το `string` εισάγεται στο τέλος του `header`. Αυτό χρησιμοποιείται κυρίως για να προστεθούν επιπλέον `headers`. Πολλαπλοί επιπλέον `headers` χωρίζονται με ένα `return (enter)` και μια νέα γραμμή.

Πρέπει να χρησιμοποιείτε το `\r\n` για να χωρίζετε τα `headers`, αν και κάποιοι Unix mail transfer agents ενδέχεται να δουλεύουν μόνο με μία νέα γραμμή (`\n`). [7]

```
<?php
mail("nobody@example.com", "the subject", $message,
    "From: webmaster@{$_SERVER['SERVER_NAME']}\r\n"
    ."Reply-To: webmaster@{$_SERVER['SERVER_NAME']}\r\n"
    ."X-Mailer: PHP/" . phpversion());
?>
```

Σχ. 8 Παράδειγμα της `mail()` με επιπλέον παραμέτρους

Η παράμετρος *additional_parameters* μπορεί να χρησιμοποιηθεί για να περαστεί μια επιπρόσθετη παράμετρος στο πρόγραμμα, το οποίο έχει ρυθμιστεί να χρησιμοποιεί όταν στέλνεται mail με χρήση της *sendmail_path* ρύθμισης. Για παράδειγμα, αυτό μπορεί να χρησιμοποιηθεί για να ορίσει την διεύθυνση του αποστολέα όταν γίνεται χρήση του sendmail με την επιλογή *-f*. Ίσως χρειαστεί να προσθέσετε τον χρήστη, που ο web server σας χρησιμοποιεί σύμφωνα με τις ρυθμίσεις του sendmail, για να αποτρέψετε να προστεθεί στο μήνυμα ένα 'X-Warning' header όταν ορίζετε τον αποστολέα με αυτή τη μέθοδο. [7]

3.4.2. Upload με την POST μέθοδο

Η PHP είναι ικανή για να λαμβάνει upload αρχείων από οποιοδήποτε RFC-1867 συμβατό browser (αυτό συμπεριλαμβάνει τους Netscape Navigator 3 ή μεγαλύτερο, Microsoft Internet Explorer 3 με ένα patch από τη Microsoft, ή μεγαλύτερο χωρίς κάποιο patch). Αυτό το χαρακτηριστικό επιτρέπει στους χρήστες να κάνουν upload τόσο κειμένου, όσο και binary αρχεία. Με τις συναρτήσεις αναγνώρισης και χειρισμού αρχείων της PHP, έχετε πλήρη έλεγχο στο ποιός επιτρέπεται να κάνει upload και τι θα γίνει με το αρχείο από τη στιγμή που έχει γίνει upload και μετά. [7]

Μια οθόνη για upload αρχείου μπορεί να φτιαχτεί δημιουργώντας μια ειδική φόρμα η οποία μοιάζει κάπως έτσι:

```
<form enctype="multipart/form-data" action="_URL_" method="post">
  <input type="hidden" name="MAX_FILE_SIZE" value="30000">
  Send this file: <input name="userfile" type="file">
  <input type="submit" value="Send File">
</form>
```

Η διαμόρφωση μιας μορφής για να φορτώσει το αρχείο απαιτεί δύο ιδιότητες μέσα στο <form.> την `method=post` και το `and "multipart/form-data"`. Όταν το στοιχείο στέλνεται, το αρχικό αρχείο (συμπεριλαμβανομένης της πλήρους πορείας) του αρχείου δεδομένου ότι ήταν στον υπολογιστή σας στέλνεται στον web server. [7]

Το `_URL_` πρέπει να δείχνει σε κάποιο PHP αρχείο. Το `MAX_FILE_SIZE` κρυφό πεδίο (`hidden`) πρέπει να προηγείται του `input` πεδίου για το αρχείο και η τιμή του είναι το μέγιστο μέγεθος αρχείου που επιτρέπεται. Η τιμή είναι σε `byte`. Το `MAX_FILE_SIZE` είναι ενημερωτικό για τον browser. Είναι εύκολο να παρακαμφθεί αυτή η μέγιστη τιμή. Έτσι μην στηρίζουμε ότι ο browser υπακούει την επιθυμία σας! Οι PHP-ρυθμίσεις όμως, για το μέγιστο μέγεθος (`maximum-size`), δεν μπορούν να ξεγελαστούν. Καλύτερα να προσθέτουμε το `MAX_FILE_SIZE` ούτως ή άλλως γιατί προστατεύει τους χρήστες από τον κόπο να περιμένουν για ένα μεγάλο αρχείο να μεταφερθεί μόνο και μόνο για να μάθουν πως ήταν πολύ μεγάλο μετά. [7]

Οι μεταβλητές που ορίζονται για αρχεία που έχουν γίνει upload αλλάζουν ανάλογα με την έκδοση της PHP και τις ρυθμίσεις. Η `autoglobal` μεταβλητή `$_FILES` υπάρχει από την PHP 4.1.0. Το `$HTTP_POST_FILES` array υπάρχει από την PHP 4.0.0. Αυτοί οι πίνακες θα περιέχουν όλες τις πληροφορίες των αρχείων που έχουν γίνει upload. Η χρήση της `$_FILES` προτιμάται. Αν το PHP

directive `register_globals` είναι ενεργοποιημένο, τα σχετικά ονόματα των μεταβλητών θα υπάρχουν επίσης. Το `register_globals` έχει ως προεπιλογή να είναι *απενεργοποιημένο* από την PHP 4.2.0. Τα περιεχόμενα της `$_FILES` από το παράδειγμα μας είναι ως ακολούθως. Σημειώστε πως αυτό υποθέτει τη χρήση του ονόματος αρχείου να είναι *userfile*, όπως χρησιμοποιείται στο παράδειγμα παραπάνω. [7]

`$_FILES['userfile']['name']` : Το αρχικό όνομα του αρχείου στο μηχανήμα του client.

`$_FILES['userfile']['type']` : Το mime type του αρχείου, αν ο browser έχει δώσει αυτή τη πληροφορία. Ένα παράδειγμα θα ήταν *"image/gif"*.

`$_FILES['userfile']['size']`: Το μέγεθος, σε byte, του αρχείου που έχει γίνει upload.

`$_FILES['userfile']['tmp_name']` : Το προσωρινό όνομα του αρχείου στο οποίο έχει αποθηκευτεί το αρχείο που έχει γίνει upload στον server.

`$_FILES['userfile']['error']` : Ο κωδικός σφάλματος που σχετίζεται με αυτό το upload αρχείου. Το *[error]* έχει προστεθεί στην PHP 4.2.0

`$HTTP_POST_FILES` και δεν είναι μια `autoglobal` μεταβλητή όπως είναι η `$_FILES`. Η PHP 3 δεν έχει υποστήριξη για την `$HTTP_POST_FILES`. [7]

Τα αρχεία προεπιλεγμένα θα αποθηκεύονται στον προσωρινό κατάλογο του server, εκτός και αν κάποια άλλη τοποθεσία δοθεί με το `upload_tmp_dir` directive στο `php.ini`. Ο προεπιλεγμένος κατάλογος μπορεί να αλλαχθεί ορίζοντας την μεταβλητή περιβάλλοντος `TMPDIR` στο περιβάλλον στο οποίο εκτελείται η PHP. Ορίζοντας το με την `putenv()` μέσα από ένα PHP script δεν θα δουλέψει. Αυτή η μεταβλητή περιβάλλοντος μπορεί επιπλέον να χρησιμοποιηθεί για να σιγουρευτείτε πως και άλλες ενέργειες επιτελούνται στα αρχεία που έχουν γίνει upload, επιπλέον. [7]

3.4.3. Επικύρωση upload αρχείων

Για να κατανοήσουν την επικύρωση upload αρχείων θα το μελετήσουμε πάνω σε ένα παράδειγμα. [7]

```
<?php
// In PHP earlier than 4.1.0, $HTTP_POST_FILES should be used
instead of $_FILES.
// In PHP earlier than 4.0.3, use copy() and is_uploaded_file()
instead of move_uploaded_file

$uploaddir = '/var/www/uploads/';

print "<pre>";
if (move_uploaded_file($_FILES['userfile']['tmp_name'], $uploaddir
. $_FILES['userfile']['name'])) {
    print "File is valid, and was successfully uploaded. Here's
```



```

some more debugging info:\n";
    print_r($_FILES);
} else {
    print "Possible file upload attack! Here's some debugging
info:\n";
    print_r($_FILES);
}

?>

```

Το PHP script το οποίο λαμβάνει το αρχείο πρέπει να εκτελέσει οποιαδήποτε λογική είναι απαραίτητη για να αποφασίσει τι θα γίνει με τα αρχεία. Μπορείτε για παράδειγμα να χρησιμοποιήσετε την `$_FILES['userfile']['size']` μεταβλητή για να πετάξετε οποιαδήποτε αρχεία είναι είτε πολύ μικρά είτε πολύ μεγάλα. Μπορείτε να χρησιμοποιήσετε την `$_FILES['userfile']['type']` μεταβλητή για να πετάξετε οποιαδήποτε αρχεία δεν ταιριάζουν σε ορισμένα κριτήρια του τύπου του αρχείου. Από την PHP 4.2.0, μπορείτε να χρησιμοποιείτε την `$_FILES['userfile']['error']` και να προγραμματίζετε τη λογική σας σύμφωνα με τους [κωδικούς σφαλμάτων](#). Οποιαδήποτε η λογική, θα πρέπει είτε να διαγράψετε το αρχείο από τον προσωρινό κατάλογο ή να το μετακινήσετε κάπου αλλού. [7]

Το αρχείο θα διαγραφεί από τον προσωρινό κατάλογο στο τέλος του request αν δεν έχει μετακινηθεί ή διαγραφεί. [7]

3.4.4. `is_uploaded_file()`

Η συνάρτηση `is_uploaded_file` λειτουργεί ως εξής: εάν το αρχείο φορτώθηκε μέσω post HTTP. Η κανονική του μορφή είναι `bool is_uploaded_file (string filename)`. Επιστρέφει TRUE εάν το αρχείο που ονομάστηκε από το όνομα αρχείου φορτώθηκε μέσω ταχυδρομείου HTTP. Αυτό είναι χρήσιμο να βοηθήσει να εξασφαλίσει ότι ένας κακόβουλος χρήστης δεν έχει προσπαθήσει να εξαπατήσει το χειρόγραφο στην εργασία στα αρχεία επάνω στα οποία δεν πρέπει-για παράδειγμα, `/etc/passwd`. Αυτό το είδος του ελέγχου είναι ιδιαίτερα σημαντικό εάν υπάρχει οποιαδήποτε πιθανότητα οτιδήποτε έγινε με τα upload αρχεία θα μπορούσε να αποκαλύψει περιεχόμενό τους στο χρήστη, ή ακόμα και σε άλλους χρήστες στο ίδιο σύστημα. Για την κατάλληλη εργασία, οι `is_uploaded_file ()` ανάγκες λειτουργίας ένα επιχείρημα όπως `$_FILES [«userfile»] [«tmp_name»]`, - το όνομα του upload αρχείου στη μηχανή `$_FILES [username] [name]` δεν λειτουργεί. [7]

```

<?php
if (is_uploaded_file($_FILES['userfile']['tmp_name'])) {
    echo "File ". $_FILES['userfile']['name'] ." uploaded successfully.\n";
    echo "Displaying contents\n";
    readfile($_FILES['userfile']['tmp_name']);
} else {
    echo "Possible file upload attack: ";
    echo "filename '". $_FILES['userfile']['tmp_name'] . "'.";
}
?>

```

Σχ. 9 Ένα μικρό παράδειγμα **is_uploaded_file**

3.4.5. move_uploaded_file()

Η συνάρτηση **move_uploaded_file()** μετακινεί ένα uploaded file σε μία νέα περιοχή. Η γενική περιγραφή της είναι: [7]

bool **move_uploaded_file** (string filename, string destination)

Η συνάρτηση αυτή ελέγχει για να εξασφαλίσει ότι το αρχείο που υποδεικνύεται από το filename είναι ένα upload το αρχείο (που σημαίνει ότι φορτώθηκε μέσω post HTTP των PHP's HTTP POST που φόρτωσε το μηχανισμό). Εάν το αρχείο ισχύει, θα κινηθεί προς το filename που δίνεται από τον προορισμό(destination). Εάν filename δεν ισχύει ότι είναι ένα upload file, τότε καμία ενέργεια δεν θα πραγματοποιηθεί, και η συνάρτηση **move_uploaded_file()** θα επιστρέψει **FALSE**. Εάν filename ισχύει ότι είναι ένα upload file, αλλά δεν μπορεί να μετακινηθεί για κάποιο λόγο, δεν θα πραγματοποιηθεί καμία ενέργεια, και η συνάρτηση **move_uploaded_file()** θα επιστρέψει **FALSE**. Επιπλέον, μια προειδοποίηση θα εκδοθεί. [7]

3.4.6. basename()

Άλλη μια συνάρτηση που θα χρησιμοποιήσουμε στις εφαρμογές μας στην επικύρωση upload αρχείων είναι η **basename()**. Η γενική της μορφή:

string **basename** (string path [, string suffix])

Λαμβάνοντας υπόψη ένα string που περιέχει ένα μονοπάτι σε ένα αρχείο, αυτή η συνάρτηση θα επιστρέψει το όνομα βάσης του αρχείου. Εάν το filename ολοκληρωθεί μέσα suffix, αυτό θα σταματήσει εκεί. Στα Windows, και η κάθετος (/) και η αντίστροφη κάθετος (\) χρησιμοποιούνται ως χαρακτήρας διαχωριστών directory. Σε άλλο περιβάλλον, είναι η μπροστινή κάθετος (/).[7]

```
<?php
$path = "/home/httpd/html/index.php";
$file = basename($path); // $file is set to "index.php"
$file = basename($path, ".php"); // $file is set to "index"
?>
```

Σχ. 10 Ένα παράδειγμα με basename()

4. MySQL με PHP

4.1. Εισαγωγή

Για πολλούς ανθρώπους, ο κύριος λόγος εκμάθησης μιας γλώσσας όπως η PHP είναι λόγω της αλληλεπίδρασης της με τις βάσεις δεδομένων. Σε αυτή την πτυχιακή θα παρουσιαστεί πως μπορεί να χρησιμοποιηθεί η PHP με μία MySQL βάση δεδομένων για να αποθηκεύονται δεδομένα στο WEB και να τα περιλαμβάνει στο website μας. Είναι πραγματικά εκπληκτικό πόσο χρήσιμη μπορεί να είναι μια βάση δεδομένων όταν χρησιμοποιηθεί στο website. Υπάρχει μια τεράστια ποικιλία από πράγματα που μπορούμε να κάνουμε, από την επίδειξη απλών καταλόγων μέχρι την λειτουργία ενός πλήρους website από την βάση δεδομένων. [7]

Χρειαζόμαστε τρία πράγματα για να τρέξουμε τα scripts της PHP που να έχουν πρόσβαση στις βάσεις δεδομένων MySQL. Αρχικά, θα χρειαστούμε ένα webserver. Αυτό μπορεί είτε να είναι σε έναν δικό μας υπολογιστή είτε σε έναν web host. Σε οποιοδήποτε λογισμικό web server μπορεί να λειτουργήσει η PHP και MySQL αλλά το καλύτερο που μπορεί να χρησιμοποιηθεί είναι ο Apache.

Οι βάσεις δεδομένων MySQL έχουν μια τυποποιημένη οργάνωση. Αποτελούνται από μια βάση δεδομένων, στην οποία είναι περιλαμβανόμενοι πίνακες. Κάθε ένας από αυτούς τους πίνακες είναι χωριστός και μπορεί να έχει τους διαφορετικούς τομείς κ.λ.π. ακόμα κι αν είναι μέρος μιας βάσης δεδομένων. Κάθε πίνακας περιέχει τα αρχεία που αποτελούνται από τους τομείς. Προτού κάνουμε κάτι με τη βάση δεδομένων μας, πρέπει να δημιουργήσουμε έναν πίνακα. Ένας πίνακας είναι ένα τμήμα της βάσης δεδομένων για την αποθήκευση των σχετικών πληροφοριών. Σε αυτόν πίνακα θα οργανώσουμε τους διαφορετικούς τομείς που θα χρησιμοποιηθούν σε εκείνο τον πίνακα. [7]

4.2. Δημιουργία πίνακα στην PHP

Για να δημιουργήσουμε ένα πίνακα στην PHP έχει αυτήν την μορφή:

```
CREATE TABLE tablename {
```

```
Τομείς
```

```
}
```

Οι Τομείς έχουν την εξής μορφή:

```
fieldname type(length) extra info,
```

Υπάρχει μια ευρεία ποικιλία των τομέων και των ιδιοτήτων διαθέσιμες σε MySQL και θα αναφέρουμε μερικές από αυτές:

Field Type	Description
TINYINT	Small Integer Number
SMALLINT	Small Integer Number
MEDIUMINT	Integer Number
INT	Integer Number

VARCHAR Text (maximum 256 characters)

TEXT Text

Πίνακας 4 ιδιότητες τομέων στην MySQL

Για παράδειγμα θέλουμε να δημιουργήσουμε μια Βάση Δεδομένων με επαφές όπου θέλουμε εισάγουμε πληροφορίες και να παρουσιαστούν στο Διαδίκτυο(βλ. Πίνακας 4.1.). [7]

Name	Type	Length	Description
id	INT	6	A unique identifier for each record
first	VARCHAR	15	The person's first name
last	VARCHAR	15	The person's last name
phone	VARCHAR	20	The person's phone number
mobile	VARCHAR	20	The person's mobile number
fax	VARCHAR	20	The person's fax number
email	VARCHAR	30	The person's e-mail address
web	VARCHAR	30	The person's web address

Πίνακας 5 Παράδειγμα δημιουργίας μιας βάσης Δεδομένων

Υπάρχει ένα άλλο πράγμα που πρέπει να γνωρίζουμε σε αυτήν την βάση δεδομένων. Ο τομέας id θα τεθεί επίσης όπως PRIMARY, INDEX, UNIQUE και θα τεθεί το auto_increment. Ο λόγος για αυτό είναι ότι αυτό θα είναι το αρχικό και δείκτης, και πρέπει έτσι να είναι μοναδικό. Η αυτόματη ρύθμιση αύξησης σημαίνει ότι όποτε προσθέτετε ένα αρχείο, εφ' όσον δεν διευκρινίζετε μια ταυτότητα (id), αυτό θα δοθεί τον επόμενο αριθμό. [7]

4.3. Σύνδεση της MySQL με PHP

Ο ακόλουθος κώδικας πρέπει να χρησιμοποιηθεί για να δημιουργήσει αυτόν τον πίνακα σε PHP γλώσσα. [7]

```
<?
$user="username";
$password="password";
$database="database";
mysql_connect(localhost, $user, $password);
@mysql_select_db($database) or die( "Unable to select database");

$query="CREATE TABLE contacts (
id int(6) NOT NULL auto_increment,
```

```

first varchar(15) NOT NULL,

last varchar(15) NOT NULL,

phone varchar(20) NOT NULL,

mobile varchar(20) NOT NULL,

fax varchar(20) NOT NULL,

email varchar(30) NOT NULL,web varchar(30) NOT NULL,

PRIMARY KEY (id),UNIQUE id (id),KEY id_2 (id));

mysql_query($query);
mysql_close();
?>

```

Το πρώτο πράγμα πρέπει να κάνουμε προτού γίνει οποιαδήποτε εργασία είναι να συνδέουμε τη βάση δεδομένων MySQL. Αυτό είναι ένα εξαιρετικά σημαντικό βήμα ως προς αν δεν συνδεθούμε, οι εντολές μας θα αποτύχουν στη βάση δεδομένων. Για τη σωστή χρησιμοποίηση των βάσεων δεδομένων είναι να διευκρινιστεί το όνομα χρήστη, ο κωδικός πρόσβασης και το όνομα βάσεων δεδομένων:

```

$username="username";
$password="password";
$dbase="your_database";

```

Σε αυτό το σημείο μπορεί να αναρωτηθούμε εάν υπάρχει κίνδυνος ασφάλειας, που κρατά τον κωδικό πρόσβασής μας στο αρχείο. Δεν πρέπει να ανησυχούμε, εν τούτοις, επειδή ο κώδικας της PHP είναι επεξεργασμένος από τον server πριν σταλεί στον browser (μηχανή αναζήτησης) έτσι είναι αδύνατο για το χρήστη να δει την πηγή του script.

4.3.1. mysql_connect()

Έπειτα, θα πρέπει να εκδώσουμε την εντολή για να αρχίσουμε μια σύνδεση βάσεων δεδομένων:

```
mysql_connect(localhost,$username,$password);
```

Η συνάρτηση **mysql_connect()** ανοίγει μία σύνδεση σε ένα MySQL Server. Μορφή αυτής της συνάρτησης για να γίνει η σύνδεση αυτή είναι

```
mysql_connect ( [string server [, string username [, string password [, bool new_link [, int client_flags]]]] )
```

Η παράμετρος *string server* είναι ο MySQL server. Περιλαμβάνει το μονοπάτι στο οποίο βρίσκεται η port ή σε μια τοπική υποδοχή π.χ.»: /path/to/socket " για το localhost. Η παράμετρος *string username* είναι το όνομα το χρήστη που είναι το αντίστοιχο όνομα του χρήστη στον MySQL server. Η παράμετρος *string password* είναι ο κωδικός πρόσβαση. Εάν μια δεύτερη κλήση γίνεται στο `mysql_connect ()` με τα ίδια επιχειρήματα, καμία νέα σύνδεση δεν θα καθιερωθεί, αλλά αντ' αυτού, το προσδιοριστικό συνδέσεων της ήδη ανοιγμένης σύνδεσης θα επιστραφεί. Η παράμετρος *new_link* τροποποιεί αυτήν την συμπεριφορά και κάνει `mysql_connect ()` πάντα να ανοίξει μια νέα σύνδεση, ακόμα κι αν `mysql_connect ()` κλήθηκε πριν με τις ίδιες παραμέτρους. Η παράμετρος *client_flags* μπορεί να είναι ένας συνδυασμός των σταθερών `MYSQL_CLIENT_COMPRESS`, `MYSQL_CLIENT_IGNORE_SPACE` ή `MYSQL_CLIENT_INTERACTIVE`.

4.3.2. `mysql_close()`;

Προτού παρουσιάσουμε πώς θα εργαστούμε με τη βάση δεδομένων, θα δείξουμε μια εντολή:

`mysql_close();` ή `mysql_disconnect()`

Είναι μια πολύ σημαντική εντολή δεδομένου ότι κλείνει τη σύνδεση με τον server της βάσης δεδομένων. Η σελίδα θα τρέξει ακόμα εάν δεν περιλαμβάνετε αυτήν την εντολή αλλά πάρα πολλές ανοικτές συνδέσεις MySQL μπορούν να προκαλέσουν τα προβλήματα για έναν web host. Είναι μία καλή επιλογή να υπάρχει πάντα αυτή η γραμμή μόλις έχουμε ολοκληρώσει όλες τις εντολές μας στη βάση δεδομένων, ώστε ο server να τρέχει σωστά.

4.3.3. `mysql_select_db()`

Αφότου έχουμε συνδεθεί με τον server της βάσης δεδομένων πρέπει έπειτα να επιλέξετε τη βάση δεδομένων που επιθυμούμε να χρησιμοποιήσουμε. Αυτό πρέπει να είναι μια βάση δεδομένων στην οποία το όνομα χρήστη μας έχει πρόσβαση. Η ακόλουθη εντολή:

`@mysql_select_db($database) or die("Unable to select database");`

χρησιμοποιείται για να επιλέξει η PHP τη βάση δεδομένων που αποθηκεύεται στο μεταβλητό `$database` (που θέτουμε νωρίτερα). Θέτει την τρέχουσα βάση δεδομένων ενεργή. Εάν δεν μπορέσει να συνδέσει θα σταματήσει η σελίδα και θα εμφανιστεί το κείμενο:

`Unable to select database`

Έχουμε συνδεθεί με τον server και έχουμε επιλέξει τη βάση δεδομένων που θέλουμε να δουλέψουμε και μπορούμε να ξεκινήσουμε τις εντολές στον server. Υπάρχουν δύο τρόποι να επεξεργαστούμε μια εντολή. Ο ένας τρόπος είναι να εισαχθεί ακριβώς η εντολή στην PHP. Αυτός ο τρόπος χρησιμοποιείται εάν δεν θα υπάρξει κανένα αποτέλεσμα ή πρόβλημα από τη λειτουργία. Ο άλλος τρόπος είναι να καθοριστεί η εντολή ως μεταβλητή. Αυτό θα θέσει τη μεταβλητή με τα αποτελέσματα της λειτουργίας.

4.3.4. `mysql_query($query);`

Η λειτουργία αυτής της εντολής είναι να στέλνει μία ερώτηση στην ενεργό Βάση δεδομένων. Η γενική της μορφή :

`mysql_query (string query [, resource link_identifier])`

Η παράμετρος query πραγματοποιεί την συγκεκριμένη ενέργεια που θέλουμε από την βάση Δεδομένων. Οι δηλώσεις που μπορεί να πάρει για να πραγματοποιήσει την ενέργεια είναι SELECT, SHOW, DESCRIBE or EXPLAIN. Άλλοι τύποι δηλώσεων είναι UPDATE, DELETE, DROP. Αν η `mysql_query()` είναι πραγματοποιήσιμη τότε στην εντολή `mysql_fetch_array()`, και άλλες λειτουργίες για την εξέταση των πινάκων ώστε να γίνουν τα οι αντίστοιχες λειτουργίες που θέλουμε.

4.4. Εισαγωγή Δεδομένων

Θα επιστρέψουμε στη βάση δεδομένων επαφών που δημιουργήσαμε προηγουμένως. Θα προσθέσουμε τώρα τις πρώτες πληροφορίες μας στη βάση δεδομένων:

First: John
Last: Smith
Phone: 01234 567890
Mobile: 00112 334455
Fax: 01234 567891
E-mail: johnsmith@gowansnet.com
Web: <http://www.gowansnet.com>

Όλα θα αυτά θα εισαχθούν με μία εντολή:

```
$query = "INSERT INTO contacts VALUES ('John','Smith','01234 567890','00112 334455','01234 567891','johnsmith@gowansnet.com','http://www.gowansnet.com');"
```

```
INSERT INTO contacts VALUES
```

Η παραπάνω εντολή ορίζει την εισαγωγή, στον πίνακα που λέγεται contacts, τα δεδομένα που ακολουθούν, δηλαδή το δεδομένο “John” θα καταχωρηθεί

αντίστοιχα στο δεύτερο τομέα του πίνακα της βάσης δεδομένων , στο 'first' . Παρατηρούμε στο παράδειγμα ότι στην εισαγωγή των δεδομένων δεν υπάρχει ο πρώτος τομέας που βρίσκεται στην βάση δεδομένων, δηλαδή το id. Αυτό συμβαίνει επειδή το id πρόκειται να ενεργήσει ως τομέας δεικτών. Στο id έχουμε ορίσει στην βάση δεδομένων αυτόματη ρύθμιση (Auto Increment) ώστε να μην υπάρχει πρόβλημα στην εισαγωγή δεδομένων με ίδια στοιχεία.

4.5. Εμφάνιση Δεδομένων

Εάν στην βάση δεδομένων μας έχουμε πολλά αρχεία, θα ήταν καλύτερο να γνωρίζαμε πως θα εμφανιζόταν χρησιμοποιώντας την PHP. Αυτό για να πραγματοποιηθεί θα χρησιμοποιήσουμε μια ερώτηση MySQL, που θα είναι η παρακάτω εντολή:

```
SELECT * FROM contacts
```

Είναι μια βασική εντολή MySQL που θα πει στο script να επιλέξει όλα τα αρχεία στον πίνακα contacts. Επειδή θα υπάρξει εκτέλεση από αυτήν την εντολή πρέπει να εκτελεσθεί με τα αποτελέσματα που ορίζονται σε μια μεταβλητή:

```
$query="SELECT * FROM contacts";  
$result=mysql_query($query);
```

Σε αυτήν την περίπτωση ολόκληρο το περιεχόμενο της βάσης δεδομένων περιλαμβάνεται τώρα σε μια πρόσθετη σειρά με το όνομα \$result. Προτού να μπορέσουμε να εκτελέσουμε αυτό το στοιχείο πρέπει να αλλάξουμε κάθε κομμάτι σε μια χωριστή μεταβλητή. Θα ακολουθήσουμε το εξής στάδιο, δηλαδή την χρήση ενός βρόγχου. Πρέπει να οργανώσουμε ένα βρόγχο για να πάρει σειρά κάθε αποτέλεσμα και να εκτυπώσει τα στοιχεία κράτησε στην οθόνη.

```
$i=0;  
while ($i < $num) {
```

Κώδικας

```
$i++;  
}
```

Αυτό είναι ένας βασικός βρόχος της PHP και θα εκτελέσει τον κώδικα ο σωστός αριθμός χρόνων. Κάθε φορά που \$i θα είναι ένα μεγαλύτερο από το χρόνο πριν. Αυτό είναι χρήσιμο, όπως \$i μπορεί να χρησιμοποιηθεί για να πει το χειρόγραφο που η γραμμή των αποτελεσμάτων πρέπει να διαβαστεί. Δεδομένου ότι η πρώτη γραμμή στην παραγωγή MySQL είναι 0, αυτό θα λειτουργήσει σωστά. Στις εφαρμογές μας θα χρησιμοποιήσουμε την παράμετρο

```
mysql_fetch_array ( resource result [, int result_type] )
```

Είναι σαν εσωτερικός δείκτης που δείχνει την πρώτη γραμμή του πίνακα που κινείται ανά γραμμή, μέχρι να φτάσει στο τέλος της γραμμής του πίνακα και να

εκπληρωθούν οι ενέργειες και λειτουργίες που απαιτούνται . Η παράμετρος result είναι ο πόρος αποτελέσματος που αξιολογείται Αυτό το αποτέλεσμα προέρχεται από μια κλήση στο mysql_query ().Αυτό θα το δούμε και πρακτικά και παρακάτω.

Είδαμε πως μπορούν να εκτελεστούν όλα τα δεδομένα του πίνακα contact. Η PHP έχει την δυνατότητα να επιλέγει συγκεκριμένα αρχεία ή αρχεία που ταιριάζουν με ορισμένα κριτήρια.. Για να το πετύχουμε αυτό πρέπει να χρησιμοποιήσουμε μια παραλλαγή της SELECT ερώτησης, δηλαδή:

```
SELECT * FROM contacts WHERE first='john'
```

Με αυτόν τον τρόπο θα εμφανίσει στην οθόνη όλα τα δεδομένα του πίνακα contact που θα έχουν στον τομέα first την μεταβλητή john'. Επίσης μπορούμε να εκτελέσουμε τα δεδομένα βασισμένα σε ένα τομέα (παραδείγματος χάριν που τοποθετεί τους χρήστες κατά αλφαβητική σειρά). Εξ ορισμού, η εκτέλεση από τα queries μας θα είναι με βάση σειράς τον τομέα id, ξεκινώντας από 1 προς τα πάνω. Μπορούμε να το ταξινομήσουμε σε οποιοδήποτε τομέα, εν τούτοις. Παραδείγματος χάριν, θα ήταν χρήσιμο να τοποθετηθούν όλοι οι χρήστες κατά αλφαβητική σειρά βασισμένη στο τελευταίο όνομά τους, δηλαδή από το Α στο Ω και αντίστροφα Ω στο Α . Για να κάνει αυτό θα χρησιμοποιούσατε το ακόλουθη query:

```
SELECT * FROM contacts ORDER BY last ASC
```

Θα μπορούσατε επίσης να χρησιμοποιήσετε DESC για να διατάξετε τα στοιχεία με την αντίστροφη σειρά.

4.6. Ο τομέας ID

Ας θυμηθούμε το παράδειγμα μας, στη δημιουργία της βάσης δεδομένων του πίνακα contact, περιλάβαμε έναν αριθμητικό τομέα αποκαλούμενο id. Αυτός ο τομέας τέθηκε ως auto_increment καθώς επίσης και ως αρχικός τομέας. Έχουμε εξηγήσει ήδη πώς αυτός ο τομέας είναι μοναδικός για κάθε ενιαίο αρχείο στη βάση δεδομένων, αλλά θα εξηγήσουμε τώρα ένα στάδιο περαιτέρω πώς αυτό μπορεί να χρησιμοποιηθεί για να επιλέξει ένα μεμονωμένο αρχείο από μια βάση δεδομένων. [3]

Πιο πριν είχαμε δει πώς η PHP να επιλέξει τα αρχεία από τη βάση δεδομένων βασισμένη στο περιεχόμενο των τομέων χρησιμοποιώντας `SELECT * FROM contacts WHERE field='value'`.

Τώρα, χρησιμοποιώντας το μοναδικό id τομέα, μπορούμε να επιλέξουμε ότι δεδομένα θέλουμε από την Βάση δεδομένων: [3]

```
SELECT * FROM contacts WHERE id='$id'
```

Όπου \$id είναι μια μεταβλητή διάφορη ένα αρχείο και μπορούμε να χρησιμοποιήσουμε αυτό πολύ αποτελεσματικά με διάφορους τρόπους. Παραδείγματος χάριν, εάν θελήσουμε να οργανωθούμε σε μια δυναμική περιοχή

μέσω μιας βάσης δεδομένων και ενός ενιαίου PHP κώδικα (script), θα μπορούσαμε να γράψουμε το κώδικα για να περιλάβουμε τα στοιχεία βάσεων δεδομένων στο design. Κατόπιν, χρησιμοποιώντας τον τομέα id, θα μπορούσαμε να επιλέξουμε κάθε μεμονωμένη σελίδα και να την βάλουμε να εκτελεστεί. Μπορούμε ακόμη και να χρησιμοποιήσουμε την URL της σελίδας για να διευκρινίσετε το αρχείο που θέλουμε, π.χ. <http://www.yoursite.com/news/items.php?id=7393> δηλαδή κάνει URL στην διεύθυνση αυτή και τα δεδομένα που θα περιέχει θα έχουν, θα αντιστοιχούν στην ταυτότητα id με τον αριθμό 7393. [3]

Χρησιμοποιώντας αυτής την μέθοδο να διαλέξει ένα αρχείο που χρησιμοποιεί το URL για να επιλέξει το αρχείο, μπορεί να επεκταθεί περαιτέρω με να παραγάγει το URL δυναμικά. Ακούγεται λίγο περίπλοκο έτσι που θα διαμορφώσουμε. Για να κάνει αυτό, μια άλλη στήλη θα περιληφθεί στη στήλη εκτέλεσης, με μια σύνδεση (link) UPDATE σε αυτό. Αυτή η σύνδεση UPDATE θα δείξει μια σελίδα που επιτρέπει στο χρήστη για να ενημερώσει το αρχείο. Για να επιλέξει το αρχείο σε αυτήν την σελίδα, θα βάλουμε: [3]

?id=\$id

Με να πάρει το id του αρχείου μαζί με τις άλλες πληροφορίες όταν παράγουμε τις πληροφορίες από τη βάση δεδομένων, αυτός ο κώδικας θα δημιουργήσει μια σύνδεση(link) που έχει τον αριθμό ID κάθε αρχείου σε αυτόν. Κατόπιν, στη σελίδα UPDATE, μπορεί να υπάρξει κώδικας για να επιλέξει ακριβώς αυτό το στοιχείο. Παρακάτω θα δούμε πως λειτουργούν τα queries UPDATE ΚΑΙ DELETE με βάση το id που είναι χρήσιμα για την διόρθωση και διαγραφή δεδομένων της βάσης δεδομένων. [3]

4.7. Διόρθωση και Διαγραφή δεδομένων

Μέχρι τώρα είχαμε μάθει πώς να βάζουμε τις πληροφορίες στη βάση δεδομένων MySQL μας, να δούμε τις πληροφορίες σε αυτήν και να επιλέγαμε ποιες πληροφορίες θα επιθυμούσαμε να δούμε. Σε αυτό το μέρος θα παρουσιάσουμε πώς να κάνουμε τις δύο τελικές ενέργειες, που ενημερώνουν τη βάση δεδομένων σας και που διαγράφουν τα αρχεία από αυτήν. [3]

Ας δούμε πως μπορεί να γίνει η διόρθωση πληροφοριών της βάσης Δεδομένων MySQL. Το πρώτο πράγμα που θα φτιάξουμε είναι ένα form σε HTML που μέσα σε αυτό θα φαίνονται τα δεδομένα της βάσης στο συγκεκριμένο id τομέα που αναφερόμαστε. Με βάση το προηγούμενο παράδειγμα: [3]

```
$query = "SELECT * FROM contact WHERE id=$id";  
$res = mysql_query( $query );  
$row = mysql_fetch_array( $res );  
$ first = $row[' first '];  
$ last = $row[' last '];  
$ phone = $row[phone '];  
$ mobile = $row[' mobile '];
```

```
mysql_disconnect;
```

```

<form method="post" action=" updated.php " enctype="multipart/form-data">
first:<input type="text" name=" ud_first " value="<?=$ first?>">
last: <textarea name="ud_last" rows=4 cols=40><?=$ last?></textarea>
phone :<textarea name=" ud_phone " rows=5 cols=40><?=$ phone?></textarea>
mobile :<input name=" ud_mobile " size=48 value="<?=$ mobile?>">
<input type="hidden" name=" ud_id" value="<?=$id?>">
<input type="submit" value="Καταχώρηση >>">
</form>

```

Όπως μπορούμε να δούμε, αυτός ο κώδικας παράγει μια τυποποιημένη φόρμα, αλλά αντί της κατοχής των κενών κουτιών στην φόρμα, αυτό έχει ήδη τις τρέχουσες πληροφορίες από τη βάση δεδομένων που παρεμβάλλεται σε το. Αυτό το καθιστά αποτελεσματικότερο για ένα UPDATE εντολών. Στο επόμενο στάδιο, αφού συμπληρωθούν οι αλλαγές που θέλουμε για κάθε πεδίο, θα ενημερωθεί η βάση αυτή με ένα query: `UPDATE database SET ;` Αυτό το query καλεί τη βάση δεδομένων να ενημερώσει τον πίνακα που έχουμε όπου το ID είναι η ίδια με την αξία που αποθηκεύεται σε `$ud_id` . Αυτό το query θα μπορούσε έπειτα να ενσωματωθεί σε ένα απλόscript:

```

$ud_id=$HTTP_POST_VARS ['ud_id'];
$ud_first=$HTTP_POST_VARS ['ud_first'];
$ud_last=$HTTP_POST_VARS ['ud_last'];
$ud_phone=$HTTP_POST_VARS ['ud_phone'];
$ud_mobile=$HTTP_POST_VARS ['ud_mobile'];
$username="username";
$password="password";
$databse="your_database";
mysql_connect(localhost,$username,$password);

```

```

$query="UPDATE contacts SET first='$ud_first', last='$ud_last', phone='$ud_phone',
mobile='$ud_mobile', fax='$ud_fax', email='$ud_email', web='$ud_web' WHERE
id='$ud_id'";
mysql_query($query);
echo "Record Updated";
mysql_close();

```

Ο κώδικας θα ενημέρωνε τη βάση δεδομένων και θα έδινε στο χρήστη μια επιβεβαίωση “ Record Updated”.[3]

4.8. Διαγραφή Δεδομένων

Το τελικό μέρος της βάσης δεδομένων του πίνακα contact, αυτό που πρέπει να δημιουργηθεί είναι μια σελίδα για να διαγράψει τα αρχεία. Όπως με τη σελίδα UPDATE, αυτό πρέπει να στείλει μια ταυτότητα αρχείων σε το στο URL π.χ.: `delete.php?id=9`

Ο κώδικας για να κάνει αυτό είναι ο ίδιος ώστε να ενημερωθεί η βάση δεδομένων, εκτός από με ένα ελαφρώς διαφορετικό query MySQL. Αντί του query UPDATE πρέπει να χρησιμοποιήσετε:
`DELETE FROM contacts WHERE id='$id'`

Αυτό θα χρησιμοποιούταν έπειτα με την σύνδεσης και επιβεβαίωσης όπως ανωτέρω. [3]

5. Ανάπτυξη Εφαρμογών

5.1. Γενική Μορφή

Από αυτήν την ξενάγηση τώρα, γνωρίσαμε τα βασικά για να χρησιμοποιήσουμε την PHP και MySQL, ώστε να δημιουργήσουμε μαζί τους βάση δεδομένων-Web και τα προγράμματα. Χρησιμοποιώντας τις βάσεις δεδομένων με τον Ιστό ανοίγουμε μια τεράστια νέα επιλογή των πραγμάτων που μπορούμε να κάνουμε. Με αυτόν τον τρόπο θα αναπτύξουμε δυναμικές μας εφαρμογές ώστε να έχουμε μια σωστή εσωτερική διαμόρφωση των υποσέλιδων τους, καθώς και την εύκολη πρόσβαση του χρήστη σε αυτές τις εφαρμογές. Εκτός όμως από την δημιουργία αυτών των διαδραστικών εφαρμογών, είναι και η δημιουργία του design του ιστοχώρου για την πιο εύκολη πλοήγηση των χρηστών και την σωστή ενημέρωση στα διάφορα θέματα που τους ενδιαφέρουν. Το περιεχόμενο αυτού του ιστοχώρου περιλαμβάνει εικόνες σε μορφή *.jpg ή *.gif, καθώς και κάποια flash movies για την πιο ευχάριστη και εντυπωσιακή πλοήγηση σε αυτό. Η μορφή αυτής της σελίδας είναι εξής:



Σχ. 11 Design Διαδικτυακής Πύλης

Ο κώδικας αυτής της σελίδας είναι σε μορφή HTML αλλά την αποθηκεύουμε σε μορφή *.php (συγκεκριμένα index.php) ώστε να δουλέψουν οι εφαρμογές μας σε γλώσσα PHP και να μην υπάρχει πρόβλημα σε αυτό. Επίσης η γλώσσα της HTML δεν επηρεάζεται καθόλου με αυτόν τον τρόπο αποθήκευσης, που είναι από τα θετικά της. Παράλληλα, πίσω από την σελίδα τρέχει μια βάση δεδομένων MySQL με το όνομα melodyfm_gr, μέσα σε αυτήν υπάρχει ένα πλήθος από πίνακες (που θα αναφέρουμε παρακάτω) και είναι πολύ σημαντικές για την δημιουργία των εφαρμογών μας.

Όλα τα παραπάνω είναι απαραίτητα για το ξεκίνημα των δυναμικών εφαρμογών μας, αλλά θα υπάρχει κι άλλο ένα template, που πάνω σε αυτό θα εργάζεται ο διαχειριστής του Melodyfm, όπου θα ενημερώνει τα θέματα που αφορούν τον χρήστη, διόρθωση αυτών, καθώς και στην περίπτωση που θέλει να διαγράψει κάποια από τα δεδομένα που δεν είναι αναγκαία πλέον. Όλα αυτά θα ενημερώνονται ταυτόχρονα από την βάση δεδομένων MySQL καθώς θα ενημερώνεται και η κεντρική σελίδα που βλέπει ο χρήστης. Η δομή του template αυτού είναι:



Σχ. 12 δομή του template Διαχειριστή

Κάποιες εφαρμογές που θα αναλύσουμε και είναι σημαντικές για την Διαδικτυακή Πύλη, κάποιες από αυτές είναι Παραγωγή, Πρόγραμμα Εβδομάδας, Τοπ 40 (Ελληνικό, Ξένο, Κρητικό), Δισκογραφικές Εταιρείες (Ελλάδα-Κρήτης), Μαντινάδες, Guest Book, Διάφορες ψηφοφορίες, "Γίνε Μέλος", Μουσικά Νέα κ.α. Αυτές είναι κάποιες από τις βασικές δυναμικές εφαρμογές που θα ενημερώνουν τους χρήστες σε κάποια θέματα αλλά θα συμμετέχουν σε αυτά.

5.2. Δισκογραφικές Εταιρείες (Ελλάδα-Κρήτης),

Σε αυτήν την εφαρμογή, έχουμε να ενημερώσουμε τους χρήστες για ποιες Δισκογραφικές Εταιρείες υπάρχουν στην χώρα μας, δίνοντας κάποια στοιχεία για

την κάθε μία, όπως όνομα , Logo Εταιρείας, Διεύθυνση, τηλέφωνο. Για αυτήν τη εφαρμογή θα φτιαχτεί ένας πίνακας με το όνομα diskografies_el(Παράρτημα Α,1).

Επίσης έχουμε και έναν τομέα id που είπαμε και πιο πάνω είναι το πρωτεύον κλειδί και είναι με λίγα λόγια “η ταυτότητα” του κάθε δεδομένου. Το id θα το χρησιμοποιήσουμε σε όλες τις εφαρμογές μας.

Η μορφή αυτής εφαρμογής θα είναι:



Σχ. 13 Δομή του «Δισκογραφικές Εταιρείες Ελλάδας»

Η παραπάνω σελίδα είναι αποθηκευμένη σε μορφή diskografies_el.php (βλ. Παράρτημα Β, 1).

Γίνεται κανονικά η σύνδεση της βάσης Δεδομένων MySQL με την σελίδα και όλες τις σχετικές λειτουργίες που χρειάζονται για να διαβαστούν τα δεδομένα από τον πίνακα diskografies_el στην σελίδα του Διαχειριστή αλλά και στους χρήστες του MelodyFm. Μετά την σύνδεση αυτή, έχουμε ένα διαφορετικό query, `mysql_query("SELECT count(*) c FROM diskografies_el");`

Κάνει έλεγχο σε όλο τον πίνακα diskografies_el, αν δει ότι υπάρχουν δεδομένα καταχωρημένα σε μορφή bit, με τον βρόγχο if κάνει τον έλεγχο αυτόν. Αν δεν υπάρχει περιεχόμενο, τότε εμφανίζει στο χρήστη το μήνυμα «Δεν υπάρχουν Δισκογραφίες Ελλάδας», αλλιώς προχωράει. Ένα σημαντικό κομμάτι που πρέπει να παρατηρήσουμε είναι ότι μέσα στην γλώσσα PHP, τα περιεχόμενα της HTML δεν αναγνωρίζονται, επομένως πρέπει να εκτελεστούν μέσα στην εντολή `echo ' ';`

Ένα άλλο απαραίτητο «καθήκον» του διαχειριστή είναι να μπορεί να διαγράφει και να διορθώνει κάποια από τα δεδομένα που θέλει. Στο template του διαχειριστή υπάρχουν δυο links για την διαγραφή και διόρθωση κάθε δεδομένου.

Για την διαγραφή, ``, το id δείχνει για πιο δεδομένο αναφερόμαστε (όπως είχαμε πει και στο προηγούμενο κεφάλαιο) (βλ. Παράρτημα Β, 1).

Με το query Delete διαγράφεται το δεδομένο που θέλουμε με βάση τον τομέα id.

Για την διόρθωση των δεδομένων της βάσης Δεδομένων είναι λίγο πιο περίπλοκα τα πράγματα (όπως είχαμε μελετήσει και στο προηγούμενο κεφάλαιο). Έχουμε την εξής φόρμα (βλ. Παράρτημα Β, 1):

The screenshot shows the 'Administrator' interface for Melody FM. The main navigation bar includes links for Melody, Παραγωγοί, Πρόγραμμα, email, top40, New Events, Fun, e-shop, Links, and Έξοδος. On the left, there are menu items for Δισκογραφίες, Κρητική Παράδοση, and Κρήτη. The main content area is titled 'Επεξεργασία Δισκογραφίας Ελλάδας' and shows a form for editing a record. The record's name is 'ΑΠΟΨΗ'. The address is 'Μεσογείων 315, & Λοκρίδος 1, Χαλάνδρι, 152 31'. The telephone numbers are '210 6756006', '210 6748409', and 'fax 210 6774220'. There is a field for 'Logo Εταιρείας' and a 'Καταχώρηση >>' button.

Σχ. 14 Φόρμα διόρθωσης «Δισκογραφικές Εταιρείες Ελλάδας»

Το βασικό είναι να διαβάσει τα δεδομένα από τον πίνακα `diskografies_el` με βάση το id, ώστε να γνωρίζει ο διαχειριστής τα προηγούμενα δεδομένα που υπάρχουν στην βάση. Αφού γίνουν οι αλλαγές που θέλει ο διαχειριστής, τρέχει την `update_diskografies_el.php` (βλ. Παράρτημα Β, 1).

Στο συγκεκριμένη σελίδα της PHP γίνεται μεταφορά των αλλαγών που έγινε στην φόρμα. Δημιουργούμε έναν έλεγχο μεταξύ των στοιχείων που υπάρχουν στον πίνακα με τα νέα δεδομένα. Αν είναι όμοια τότε δεν γίνεται καμία καταχώρηση στο βάση δεδομένων. Στην αντίθετη περίπτωση με το query κάνει UPDATE τα νέα δεδομένα.

Εκτός από την ενημέρωση, διαγραφή και διόρθωση των δεδομένων από την βάση Δεδομένων, ο διαχειριστής έχει την δυνατότητα να καταχωρεί και νέα δεδομένα στην βάση.

Σχ. 15 Εισαγωγή νέων Δεδομένων «Δισκογραφικές Εταιρείες Ελλάδας»

Το σημαντικό σε αυτή την εφαρμογή είναι πως μπορεί να γίνει εισαγωγή upload του logo της εταιρείας στο server. Αφού κάνουμε αναζήτηση το image που θέλουμε για το logo της εταιρείας, Για να γίνει upload στο server (βλ. Παράρτημα Β, 1).

Όλα αυτά μας είναι γνωστά, όπως τα αναλύσαμε στα αρχικά κεφαλαία. Το βασικό είναι ότι πριν γίνει από το query η εισαγωγή των δεδομένων, δημιουργείται ένας έλεγχος με το βρόγχο if, αν υπάρχει στο logo της εταιρείας (picture) περιεχόμενο σε bit τότε θα γίνει η καταχώρηση των νέων δεδομένων με το image αλλιώς ο των νέων δεδομένων.

Με ακριβώς τον ίδιο τρόπο θα φτιαχτεί και η εφαρμογή 'Δισκογραφικές Εταιρείες Κρήτης' αλλά με άλλο πίνακα στην βάση Δεδομένων MySQL ([diskografies_krait](#)).

5.3. Πρόγραμμα Εβδομάδας

Είναι άλλη μια εφαρμογή που θα ενημερώνει τους χρήστες για το μουσικό πρόγραμμα του ραδιοφωνικού σταθμού, το οποίο χωρίζεται σε 'Πρόγραμμα Δευτέρα – Παρασκευή', 'Πρόγραμμα Σαββάτου' και 'Πρόγραμμα Κυριακής'. Αυτά τα τρία είδη λειτουργούν με τον ίδιο ακριβώς τρόπο, με την διάφορα την αλλαγή του πίνακα της βάσης δεδομένων του.

Πρόγραμμα Εβδομάδας

Πρόγραμμα Δευτέρα έως Παρασκευή - Πρόγραμμα Σαββάτου

Πρόγραμμα Κυριακής

Πρόγραμμα Δευτέρα έως Παρασκευή

6:00 -8:00	Καλημέρα Κρήτη Χαρούμενο πρωινό ξύπνημα.
8:00 -10:00	ΣΥΝΔΕΣΗ ΜΕ ΑΘΗΝΑ ΜΕ ΤΟ ΡΑΔΙΟ SFERA Με τη γνωστή εκπομπή χιουμοριστικού-σατυρικού περιεχομένου του Βασίλη Δρυμούση (Γνωστός από την συμμετοχή του στην τηλεοπτική εκπομπή Καφές με την Ελένη στον ALPHA)
10:00 -11:30	Καλημέρα όλη μέρα Σχόλια επικαιρότητας ,πρωτοσέλιδα εφημερίδων.. και όλα τα τελευταία Νέα και εξελίξεις στην Ελλάδα και τον Κόσμο με τον Μανόλη Λεκάκη.

Σχ. 16 Δομή «Πρόγραμμα Εβδομάδας»

Φυσικά αυτή η εφαρμογή θα διαχειρίζεται κανονικά από το template του διαχειριστή για να κάνει ότι αλλαγές θέλει, διαγραφές , και νέες εισαγωγές αν χρειαστεί. Ο πίνακας της βάσης δεδομένων που θα χρειαστεί είναι ο `programma_m_to_fr` (βλ. Παράρτημα Α, 2).

Η δομή του σελίδας θα περιλαμβάνει ώρα του προγράμματος, τίτλος και σχόλια για το κάθε ένα. Η σελίδα θα είναι αποθηκευμένη στην μορφή `programma_m_to_fr.php` (βλ. Παράρτημα Β, 2).

Time Slot	Title	Description	Action
6:00 - 8:00	Καλημέρα Κρήτη	Χαρούμενο πρωινό ξύπνημα.	Delete
8:00 - 10:00	ΣΥΝΔΕΣΗ ΜΕ ΑΘΗΝΑ ΜΕ ΤΟ ΡΑΔΙΟ SFERA	Με τη γνωστή εκπομπή χιουμοριστικού-σατυρικού περιεχομένου του Βασίλη Δρυμούση (Γνωστός από την συμμετοχή του στην τηλεοπτική εκπομπή Καφές με την Ελένη στον ALPHA)	Delete
10:00 - 11:30	Καλημέρα όλη μέρα	Σχόλια επικαιρότητας , πρωτοσέλιδα εφημερίδων... και όλα τα τελευταία Νέα και εξελίξεις στην Ελλάδα και τον Κόσμο με τον Μανώλη Λεκάκη.	Delete
11:30 - 13:00	Η Κρήτη ξέρει να γλεντά	με τον Γιώργο Λεκάκη	Delete
13:00 - 15:00	Μόνο επιτυχίες	Νεανική εκπομπή με νέες κυκλοφορίες Ελληνικών τραγουδιών, ΤΟΠ , πρώτες μεταδόσεις, αφιερώσεις, διαγωνισμοί .Με πολλά δώρα με την συμμετοχή των ακροατών με τον Κιτσίκη Παπαγιαννίδη.	Delete

Σχ. 17 Δομή του template Διαχειριστή «Πρόγραμμα Εβδομάδας»

Για την Διαγραφή κάποιου δεδομένου του πίνακα, επιτυγχάνεται με το id του κάθε δεδομένου, που έχει οριστεί από την αρχή από τον πίνακα (όπως είχαμε πει και στο προηγούμενο κεφάλαιο) (βλ. Παράρτημα Β, 2). Με το query Delete διαγράφεται το δεδομένο που θέλουμε με βάση τον τομέα id.

Για την διόρθωση κάποιων δεδομένων, χρησιμοποιείται μια φόρμα, όπου θα περιέχονται και τα παλιά στοιχεία του δεδομένου. Πάνω σε αυτήν την φόρμα θα κάνουμε τις αλλαγές που θέλουμε να πραγματοποιήσουμε. (βλ. Παράρτημα Β, 2).

Και για να εισαχθούν μέσα στην Βάση δεδομένων οι νέες αλλαγές, θα γίνει ο ίδιος έλεγχος όπως στην εφαρμογή της «Δισκογραφικές εταιρείες Ελλάδας », μέσω της σελίδας update_programma_m_to_fr.php (βλ. Παράρτημα Β, 2).

The screenshot shows the 'Administrator' interface for Melody FM. At the top, there is a navigation menu with links: Melody, Παραγωγός, Πρόγραμμα, email, top40, New Events, Fun, e-shop, Links, Έξοδος. On the left, there are menu items: Δισκογραφίες, Κρητική Παράδοση, Κρήτη, and Διαχειριστές. The main content area displays the user 'Διαχειριστής kristinaki' and the title 'Εισαγωγή Προγράμματος Δευτέρα-Παρασκευή'. Below this, there is a form with three input fields: 'Ώρα', 'Τίτλος', and 'Εισαγωγή κειμένου'. A 'Καταχώρηση >>' button is located below the form. At the bottom, there is a link: [Επιστροφή στο Πρόγραμμα Δευτέρα-Παρασκευή](#).

Σχ. 18 Φόρμα εισαγωγής νέων δεδομένων «Πρόγραμμα Εβδομάδας»

Η εισαγωγή των νέων δεδομένων θα είναι πραγματοποιηθεί πάνω σε αυτήν την φόρμα (Σχ.5.5.). Στην συνέχεια τα νέα δεδομένα θα αποθηκευτούν μέσα στο πίνακα `programma_m_to_fr`. Αυτό εκτελεστεί βάση της σελίδας `insert_new_programma_m_to_fr.php` (βλ. Παράρτημα Β, 2).

5.4. Τοπ 40 (Ελληνικό, Ξένο, Κρητικό)

Αυτή η εφαρμογή θα ενημερώνει τους χρήστες για το μουσικό Τοπ 40 του ραδιοφωνικού σταθμού, το οποίο χωρίζεται σε Ελληνικό, Ξένο, Κρητικό. Αυτά τα τρία είδη λειτουργούν με τον ίδιο ακριβώς τρόπο, με την διάφορα την αλλαγή του πίνακα της βάσης δεδομένων του.

Θέση	Όνομα Τραγουδιού	Τραγουδιστής	Προηγ. Εβδομ.	Εταιρία
1	Ela mou	Xristou	2	Minoan Air Lines
3	new_titlos_tragoudiou	xristina	2	melody
4	a,dbadkasb`	.s.ds sdfn .sd,fj	2	etairia
26	CHRISTINA	SLDFJSD	00000	etairia

Σχ. 19 Δομή «Τοπ 40-ελληνικό»

Φυσικά αυτή η εφαρμογή θα διαχειρίζεται κανονικά από το template του διαχειριστή για να κάνει ότι αλλαγές θέλει, διαγραφές, καθώς και νέες εισαγωγές αν χρειαστεί. Ο πίνακας της βάσης δεδομένων που θα χρειαστεί είναι ο top40_greek ((βλ. Παράρτημα Α, 3).

Η δομή αυτής της εφαρμογής θα περιέχει την θέση που θα βρίσκεται το τραγούδι, το όνομα του τραγουδιού, ο τραγουδιστής, η θέση του τραγουδιού την προηγούμενη εβδομάδα και η εταιρεία έκδοσης του τραγουδιού. Η σελίδα θα είναι αποθηκευμένη στην μορφή top_greek.php (βλ. Παράρτημα Β, 3).

Επίσης για την διαγραφή, διόρθωση μεσω φόρμας και εισαγωγή νέων δεδομένων γίνεται με το ίδιο ακριβώς τρόπο όπως και στην προηγούμενες εφαρμογές (βλ. Παράρτημα Β, 3).

5.5. Μαντινάδες

Είναι άλλη μια εφαρμογή που θα ενημερώνει τους χρήστες αλλά θα μπορούν να συμμετέχουν σε αυτήν γράφοντας την αγαπημένη τους μαντινάδα. Οι μαντινάδες χωρίζονται σε τέσσερις κατηγορίες: Διάφορες, Στρατιωτικές, Ερωτικές και

Αστείες . Αυτά τα τέσσερις είδη λειτουργούν με τον ίδιο ακριβώς τρόπο, με την διάφορα την αλλαγή του πίνακα της βάσης δεδομένων του. Στην κεντρική σελίδα του χρήστη θα υπάρχει μια φόρμα στην οποία θα μπορεί να γράψει την μαντινάδα του και ταυτόχρονα θα στέλνεται με email στον διαχειριστή του Melodyfm ώστε να την αξιολογεί. Στην συνέχεια ο διαχειριστής μπορεί να ενημερώσει την νέα μαντινάδα μέσω του template καθώς και ότι άλλες ενέργειες χρειάζονται (βλ. Παράρτημα Β, 4).

Αστείες Μαντινάδες

Δημιούργησε την δικιά σου μαντινάδα

Όνομα:

Email:

Κατηγορία Μαντινάδας:
Αστείες

Mail Message:

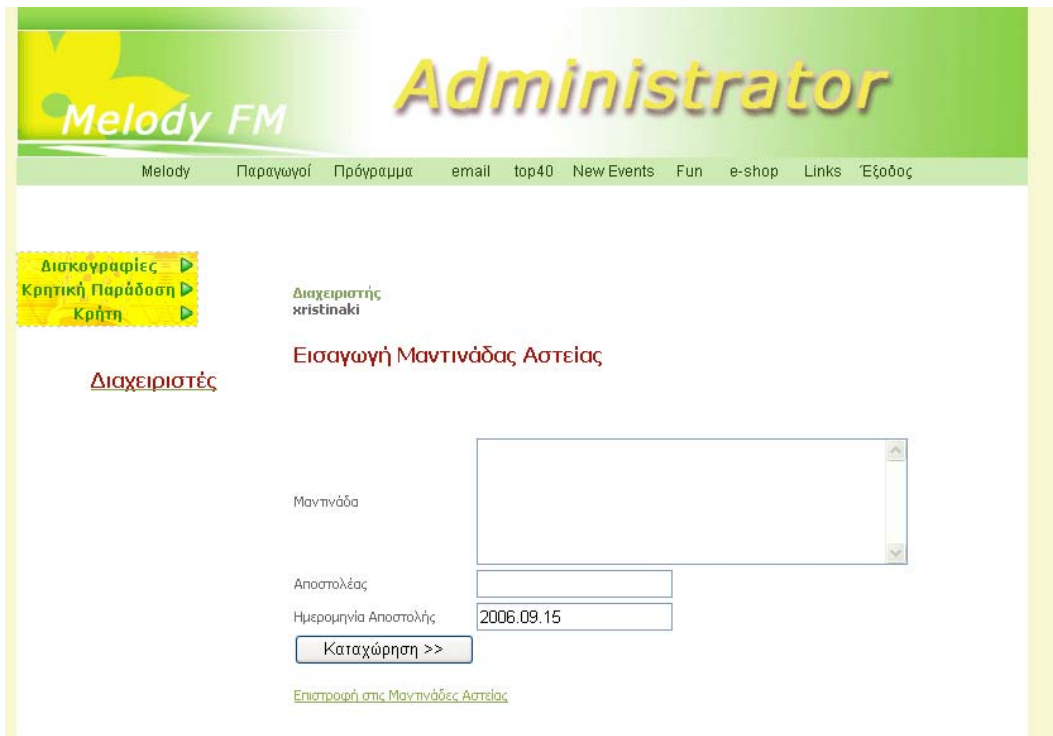
Send Mail

μπλα μπλα μπλα μπλα μπλα
μπλα μπλα μπλα μπλα μπλα
μπλα μπλα μπλα μπλα
xristina 2006-08-04

Σχ. 20 Δομή «Μαντινάδες» με φόρμα αποστολής

Για την αποστολή των περιεχομένων της φόρμας θα γίνει με την εντολή email(), όπως είχα αναλύσει και στο προηγούμενο κεφάλαιο(βλ. Παράρτημα Β, 4).

Αφού την λάβει ο Διαχειριστής και την αξιολογήσει, μπορεί να την εισάγει στο στην βάση Δεδομένων με τον πίνακα mantinades_asteies (βλ. Παράρτημα Α, 4).Η δομή εισαγωγής της μαντινάδας μέσα στην φόρμα όπου θα δουλεύει ο διαχειριστής είναι:



Σχ. 21 Δομή Διαχείρισης εισαγωγή μαντινάδας

Η εισαγωγή των νέων δεδομένων θα περιέχει την εισαγωγή μαντινάδας, το όνομα του αποστολέα και την ημερομηνία αποστολής (βλ. Παράρτημα Β, 4).

Για την διαγραφή, διόρθωση κάποιας μαντινάδας γίνεται με το ίδιο ακριβώς τρόπο όπως στις προηγούμενες εφαρμογές. Η διόρθωση της μαντινάδας γίνεται με μορφή της φόρμας με τα παλιά δεδομένα της μαντινάδας και πάνω σε αυτό θα γίνουν αλλαγές. Στην συνέχεια για να εισαχθούν μέσα στην Βάση δεδομένων οι νέες αλλαγές, θα γίνει ο ίδιος έλεγχος όπως έγινε και στις άλλες εφαρμογές που αναλύσαμε πριν (βλ. Παράρτημα Β, 4).

Με τον ίδιο τρόπο θα δημιουργηθούν και οι άλλες κατηγορίες με την αλλαγή μόνο του πίνακα της Βάση Δεδομένου MySQL.

5.6. Παραγωγοί

Αυτή η εφαρμογή έχει σκοπό να δώσει μια ενημέρωση για το ποιοι είναι οι συντελεστές και οι οργανωτές αυτού του ραδιοφωνικού σταθμού. Θα περιέχει μια photo του παραγωγού, και το όνομα του. Κατόπιν αυτό, θα είναι link σε μία άλλη σελίδα που θα παρουσιάζει μια μεγαλύτερη photo και περισσότερα χαρακτηριστικά του παραγωγού καθώς και μια φόρμα για να στείλουν οι χρήστες τα μηνύματα τους σε αυτούς.

Προσωπικό

The image shows a web application interface. On the left, under the heading "Προσωπικό", there is a list of three producers, each with a small photo and a name: Μανώλης Λεκάκης, Γιώργος Λεκάκης, and Κυριάκος Παπαγεωργίου. An orange arrow points from the first producer's name to a larger, detailed view of the same producer on the right. This detailed view includes a larger photo, the name "Μανώλης Λεκάκης", and a block of text in Greek. To the right of the text is a form titled "Στείλτε το μήνυμά σας" (Send your message) with fields for "Το όνομά σας", "Το email σας", and "Το μήνυμά σας", and a "Αποστολή" (Send) button.

Σχ. 22 Μορφή Λειτουργίας του «Παραγωγού»

Στην ουσία χρησιμοποιούμε ένα πίνακα. Prosopiko (βλ. Παράρτημα Α, 5).

Η δομή που θα τρέχει η εφαρμογή μας θα είναι αποθηκευμένη ως `pedagogue.php` (βλ. Παράρτημα Β, 5). Η διαδικασία είναι γνωστή: γίνεται κανονικά η σύνδεση της βάσης Δεδομένων MySQL με την σελίδα και όλες τις σχετικές λειτουργίες που χρειάζονται για να διαβαστούν τα δεδομένα από τον πίνακα `paragogoι`, δηλαδή εμφάνιση της photo του παραγωγού που θέλουμε σε `width=150` και το όνομα του παραγωγού. Στην συνέχεια κάνουμε link το όνομα του στην μορφή ``. Θα δημιουργηθεί μια URL με βάση την ταυτότητα `id`, παίρνοντας με αυτόν τον τρόπο αλλά στοιχεία του ίδιου πίνακα `paragogoι`. Στην σελίδα `show_paragogoι.php` (βλ. Παράρτημα Β, 5), γίνεται κανονικά πάλι η σύνδεση της βάσης Δεδομένων MySQL με την σελίδα και το διάβασμα του πίνακα `paragogoι` με βάση την ταυτότητα `id`. Δηλαδή, εμφάνιση της photo σε `width=200`, περισσότερα σχόλια για τον παραγωγό, καθώς και την `forma` που θα μπορούν οι χρήστες να στείλουν email στον παραγωγό. Αυτό θα πραγματοποιηθεί πάλι με την ταυτότητα `id`, που μέσα στην συνάρτηση `mail()` θα οριστεί το email του παραγωγού, που είναι καταχωρημένη μέσα στον πίνακα `paragogoι` (`contact_paragogoι.php`) (βλ. Παράρτημα Β, 5).

Για την διαγραφή και διόρθωση κάποιων δεδομένων των παραγωγών θα γίνει ακριβώς η ίδια διαδικασία που δείξαμε και στις άλλες εφαρμογές.

5.7. Μουσικά Νέα

Αυτή η εφαρμογή έχει ακριβώς την ίδια λειτουργία με τους παραγωγούς αλλά λιγότερη περιπλοκότητα. Η δομή της είναι :

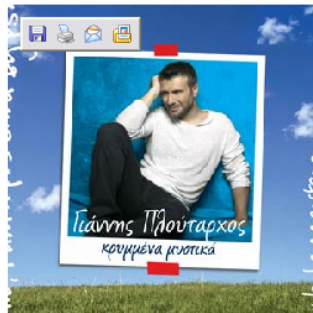


Σχ. 23 Δομή των «Μουσικών Νέων»

Βλέπουμε τον τίτλο του μουσικού νέου, μια photo, κάποια σχόλια και ένα link “Μάθετε περισσότερα” , που αντιστοιχεί σε μια άλλη URL με περισσότερα σχόλια και μια μεγάλη photo (βλ. Παράρτημα Β , 6). Ο πίνακα που θα χρησιμοποιήσουμε είναι ο music_news (βλ. Παράρτημα Α, 6).

Η διαδικασία είναι γνωστή: γίνεται κανονικά η σύνδεση της βάσης Δεδομένων MySQL με την σελίδα και όλες τις σχετικές λειτουργίες που χρειάζονται για να διαβαστούν τα δεδομένα από τον αντίστοιχο πίνακα , δηλαδή εμφάνιση της photo που θέλουμε σε width=200 και ένα μικρό κείμενο για το μουσικό νέο. Στην συνέχεια κάνουμε link το “Μάθετε περισσότερα”, ``. Θα δημιουργηθεί μια URL με βάση την ταυτότητα id, παίρνοντας με αυτόν τον τρόπο αλλά στοιχεία του ίδιου πίνακα. Η σελίδα more_music_news.php (βλ. Παράρτημα Β, 6):

"Κρυμμένα Μυστικά"...live!



Όπως έχουμε ξαναγράψει, το νέο album του Γιάννη Πλούταρχου, "Κρυμμένα Μυστικά", θα βρίσκεται στα δισκοπωλεία από τις 18 Σεπτεμβρίου. Πρώτο single είναι το κομμάτι "Δεν Είμαι Θεός", που υπογράφει ο Νίκος Αντύπας (μουσική) και η Ελεάνα Βραχάλη (στίχοι). Δύο μέρες μετά την κυκλοφορία του "Κρυμμένα Μυστικά", ο Γιάννης Πλούταρχος θα παρουσιάσει τα νέα του τραγούδια με μια μεγάλη συναυλία-party, που θα πραγμα τοποιηθεί στις 20 Σεπτεμβρίου στο Κέντρο Αθηνών, όπου θα εμφανίζεται φρέσος. Όπως γράψαμε και σε σχετική αποκλειστική είδηση την προηγούμενη εβδομάδα, στο album συμμετέχουν με τις δημιουργίες του και οι Αλέξης Σέρκος, Θανάσης Παλικανδριώτης, Δημήτρης Αυλωνίτης, Βασίλης Γαβριηλίδης, Θάνας Παπανικαλάου, Ρεβέκκα Ρούσση, Νίκος Τερζής, Στέλιος Μαρής, Ιωσήφ Σαρασιδής, Ναταλί, Σπύρος Γεωργίου αλλά και ο ίδιος ο Γιάννης Πλούταρχος, που υπογράφει τη μουσική σε δύο από τα 16 κομμάτια του δίσκου.

Επιστροφή στην αρχική

Σχ. 24 Δομή της σελίδας more_music_news.php

Αυτή η σελίδα περιέχει μια μεγάλη εικόνα με ένα μεγάλο κείμενο. Όλα αυτά περιέχονται τον ίδιο πίνακα βάσης δεδομένου `music_news`. Παράλληλα, υπάρχει στο τέλος ένα URL "Επιστροφή στην αρχική", όπου λειτουργεί με βάση την ταυτότητα `id`, και μας επιστρέφει στα Μουσικά νέα.

Όλα αυτά ενημερώνονται αυτόματα από τον διαχειριστή καθώς να έχει την δυνατότητα να εισάγει νέα μουσικά νέα, να γίνουν κάποια διαγραφή σε αυτά καθώς και διόρθωση αυτών.

5.8. Ψηφοφορία

Η εφαρμογή αυτή δίνει την δυνατότητα στον χρήστη να συμμετέχει σε κάποιες ερωτήσεις του ραδιοφωνικού σταθμού. Με την ψηφοφορία θα βγαίνουν κάποια στατιστικά αποτελέσματα ανάλογα τι ψήφους δίνουν οι χρήστες. Η διαδικασία αυτή δεν είναι τόσο απλή όσο ακούγεται. Η λειτουργία του μέσα στο `template` του διαχειριστή είναι:

The screenshot shows the 'Administrator' interface for Melody FM. At the top, there is a navigation bar with links: Melody, Παραγωγοί, Πρόγραμμα, email, top40, New Events, Fun, e-shop, Links, Έξοδος. Below this, there are menu items for 'Δισκογραφίες', 'Κρητική Παράδοση', 'Κρήτη', and 'Ψηφοφορία'. The main content area is titled 'Poll Admin Panel' and shows the administrator's name 'kristinaki'. It includes a list of actions: Δημιουργία Ερώτησης, Δημιουργία Επιλογών Ερώτησης, Διόρθωση Επιλογών Ερώτησης, Διόρθωση Ερώτησης, Διαγραφή Επιλογών Ερώτησης, and Διαγραφή Ερώτησης. A text box explains that only one poll can be active at a time. Below, a poll question is shown: 'Σας αρέσει το νέο πρόγραμμα του MelodyFM?' with options 'ναι', 'θα δούμε', 'όχι', and 'δεν απαντώ Dance'.

Σχ. 25 Μορφή Λειτουργίας της ψηφοφορίας (Poll Panel)

Στο Poll Panel του διαχειριστή (Σχ. 5.13.), δίνεται η δυνατότητα δημιουργία της ερώτησης, δημιουργία επιλογών ερώτησης, διαγραφή τους καθώς και την διόρθωση τους.

Για τις επιλογές ερώτησης, θα δημιουργηθούν αυτές οι εφαρμογές:

The image shows three forms for managing poll options. The first form, 'Add a choice option', has a text input field for 'Type in poll answer:' and an 'Add Poll Option' button. The second form, 'Delete Poll Choice', has a dropdown menu 'Select option to delete' and a 'submit' button. The third form, 'Edit Poll Choice', shows a table of existing options with 'Edit' links for each.

Answer	Edit
ναι	Edit
θα δούμε	Edit
όχι	Edit
δεν απαντώ	Edit
Dance	Edit

Σχ. 26. Επιλογές ερώτησης Add, Delete και Edit

Στο Σχ. 27 δίνει την δυνατότητα στον διαχειριστή να μπορεί να εισάγει νέες επιλογές ερωτήσεων, να διαγράψει και να διορθώσει κάποιες από αυτές (βλ. Παράρτημα Β,7). Ο πίνακας που χρησιμοποιείται είναι η p_choices (βλ. Παράρτημα Α, 7). Αυτός ο πίνακας περιλαμβάνει το όνομα της επιλογή ερώτησης (answer) και έναν τομέα (votes) που θα καταχωρούνται οι ψήφοι των χρηστών που θα δούμε παρακάτω.

The image shows two parts of the 'Poll Admin Panel'. The top part has two columns. The left column is titled 'Create Poll' and contains a text input field labeled 'Poll Question:' and a 'Create Poll' button. The right column is titled 'Delete Poll' and contains a warning message: 'Warning! Pressing the button below will empty all data to the current poll' and a 'Delete Poll' button. The bottom part of the image shows a single section titled 'Poll Admin Panel' with a text input field containing the question 'Σου αρέσει το νέο πρόγραμμα του MelodyFM?' and an 'edit poll' button.

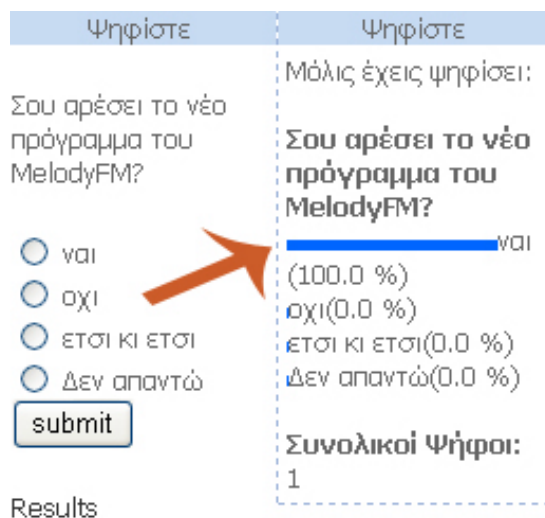
Σχ. 28 Τρόπος Λειτουργίας Ερώτησης με Add, Delete και Edit

Στο Σχ. 5.14. ο διαχειριστής με αυτόν τον τρόπο μπορεί να εισάγει, να διαγράψει και να διορθώσει τη ερώτηση που θέλει να παρουσιάσει στους χρήστες (βλ. Παράρτημα Β,7). Ο πίνακας που χρησιμοποιείται είναι p_question (βλ. Παράρτημα Α,7), που περιέχει τον τομέα ερώτηση(question). Στην εισαγωγή μιας νέας ερώτησης, ο διαχειριστής έχει το δικαίωμα μόνο μίας ερώτησης. Αν προσπαθήσει να εισάγει μια νέα ερώτηση, τότε πετάει το μήνυμα «There is already an existing poll, you cannot create another poll»(βλ. Σχ. 5.15.).

The image shows a list of links on the left side of the 'Poll Admin Panel': 'Δημιουργία Ερώτησης', 'Δημιουργία Επιλογων Ερώτησης', 'Διόρθωση Επιλογων Ερώτησης', 'Διόρθωση Ερώτησης', 'Διαγραφή Επιλογων Ερώτησης', and 'Διαγραφή Ερώτησης'. To the right, under the heading 'Create Poll', there is a warning message: 'There is already an existing poll, you cannot create another poll'. Below the warning, the text 'Question is: Σου αρέσει το νέο πρόγραμμα του MelodyFM?' is displayed.

Σχ. 29 εισαγωγή δεύτερης νέας ερώτησης

Όταν διαγραφεί η ερώτηση τότε αυτόματα διαγράφονται και επιλογές ερωτήσεων. Πλέον δεν υπάρχει ερώτηση και οι επιλογές ερωτήσεων δεν έχουν σημασία. Όλα αυτά επιτυγχάνονται με την εντολή query DELETE, διαγράφοντας όλα τα δεδομένα από τον πίνακα p_question και p_choice (βλ. Παράρτημα Β,7)



Σχ. 30 Τρόπος χρήσης ψηφοφορίας από χρήστη

Στο Σχ.5.16. παρουσιάζεται ο τρόπος χρήσης ψηφοφορίας από τους χρήστες. Επιλέγοντας την απάντηση που θέλουν αποθηκεύονται στον πίνακα p_choice και μέσω αυτού βγαίνει το αποτέλεσμα επί τις εκατό καθώς και τους συνολικούς ψήφους που έχουν γίνει. (βλ. Παράρτημα Β,7). Παράλληλα, υπάρχει και ένας άλλος πίνακας της Βάσης δεδομένου MySQL, p_ip (βλ. Παράρτημα Α,7), που κρατάει την ip του χρήστη. Με αυτό επιτυγχάνεται να γίνει σωστότερη η ψηφοφορία και να μην γίνεται η συνεχή επανάληψης ψήφου. Αυτό ελεγχεται με την εντολή \$_SERVER["REMOTE_ADDR"];, στην ουσία βλέπει την ip του χρήστη και την αποθηκεύει στον πίνακα p_ip. Όταν διαγραφεί η ερώτηση από τον διαχειριστή τότε αυτόματα διαγράφεται και ο πίνακας της p_ip. Με αυτόν τον τρόπο στη νέα ερώτηση που θα εισάγει ο διαχειριστής, ο χρήστης που είχε συμμετάσχει σε κάποια παλιά ερώτηση, να μπορεί να απαντήσει στη νέα.

5.9. Ασφάλεια Διαχειριστή

Το template στο οποίο ελέγχονται και ενημερώνονται όλες οι εφαρμογές γίνονται από τον διαχειριστή. Για περισσότερη ασφάλεια είναι καλό να υπάρχει ένας έλεγχος admin για το ποιος ενημερώνει το template ώστε να μην μπορεί οποιοσδήποτε να έχει το δικαίωμα αυτό.



Σχ. 31 Κωδικός Admin για την εισαγωγή του Διαχειριστή

Στο Σχ. 31 βλέπουμε πως μπορεί να εισαχθεί ο διαχειριστής μέσα στο template του διαχειριστή. Ο διαχειριστής θα πρέπει να έχει ένα userid και ένα κωδικό (password) για να μπορεί να μπει μέσα στην διαχείριση του MelodyFM. Σε περίπτωση που δεν θυμάται το password του μπορεί να εισάγει το email του μέσα στο στην φόρμα (Σχ. 32) ώστε να ενημερωθεί για το κωδικό του μέσω του email του.

Σχ. 32 Φόρμα εισαγωγή email για υπενθύμιση κωδικού διαχειριστή

Τα στοιχεία του διαχειριστή είναι καταχωρημένα μέσα σε ένα πίνακα της βάση δεδομένων της MySQL (admin_personel) (βλ. Παράρτημα Α,8). Σε αυτόν το πίνακα περιέχονται όλα τα στοιχεία του διαχειριστή (όνομα, επίθετο, email, τηλέφωνο, password). Βάζοντας το userid και το password, γίνεται έλεγχος και εάν υπάρχουν τα στοιχεία του διαχειριστή τότε μπαίνει μέσα στην διαχείριση δεδομένων του template (βλ. Παράρτημα Β,8).

Εάν ο διαχειριστής προσπαθήσει να μπει σε κάποια σελίδα του template χωρίς να κάνει Login με password, τότε θα του χτυπήσει ERROR(Σχ. 33) . Χρησιμοποιούμε την εντολή \$HTTP_COOKIE_VARS['userid'];, που υπάρχει σε όλες τις σελίδες του template ώστε να είναι γνωστό σε κάθε σελίδα ότι είναι μέσα ο διαχειριστής.

Στην περίπτωση που ο διαχειριστής κάνει logout (επιλογή στο template) τότε διαγράφονται τα COOKIES(βλ. Παράρτημα Β,8)..



Σχ. 34 Σελίδα diskografies_el.php που δεν έχει κάνει login ο Διαχειριστής

Παράλληλα ο Διαχειριστής έχει το δικαίωμα να εισάγει κι άλλα άτομα που να είναι διαχειριστές στο δικαίωμα αυτό. Μέσα στο template του Διαχειριστή υπάρχει μια φόρμα που μπορεί να εισάγει κάποιον νέο διαχειριστή έχοντας το δικαίωμα να διαγράψει κάποιον από αυτούς.

Διαχειριστής
kristinaki

Εισαγωγή Νέου Διαχειριστή

Όνομα

Επίθετο

Userid

Password

Email

Σχολία

[Επιστροφή στους Διαχειριστές](#)

Σχ. 35 Εισαγωγή Νέου Διαχειριστή

6. Περίληψη

6.1. Συμπεράσματα

Η διπλωματική εστίασε στο σχεδιασμό και την ανάπτυξη δυναμικών διαδραστικών εφαρμογών μιας Διαδικτυακής Πύλης με την χρήση της PHP μέσα σε αυτές. Εξετάστηκε ο τρόπος χρήσης της PHP με τη χρησιμοποίηση της Βάσης Δεδομένων MySQL που ήταν απαραίτητες για βάση δημιουργία αρχιτεκτονικής της Διαδικτυακής Πύλης.

Η αναλυτική αναφορά και μελέτη το πώς λειτουργεί η PHP και η MySQL, μας βοήθησαν στην πιο γρήγορη ανάπτυξη των εφαρμογών. Ως αποτέλεσμα ήταν η θετική και εύκολη πρόσβαση των χρηστών και η άμεση ικανοποίηση τους για την πλοήγηση τους μέσα σε αυτό.

6.2. Μελλοντική Δουλειά

Το θέμα που διερευνήθηκε στην παρούσα διπλωματική μπορεί να επεκταθεί ώστε να αποτελέσει αντικείμενο χρήσης του ραδιοφωνικού σταθμού MelodyFM. Πάνω σε αυτό θα εξασφαλίσουν την ενημέρωση των χρηστών στα θέματα που αναφέραμε παραπάνω και τους αφορούν αλλά και την διαφήμιση τους στο ηλεκτρονικό εμπόριο του internet.

AKΡΩΝΥΜΙΑ

C

CGI Common Gate Interface

CORBA Common Object Request Broker Architecture

CSS Cascading Style Sheets

F

FTP File Transfer Protocol

H

HTML HyperText Markup Language

HTTP HyperText Transfer Protocol

I

ICANN Internet Corporation for Assigned Names and Numbers

G

GUI Graphical User Interface

M

MySQL SQL Database Server

P

PHP Hypertext Preprocessor

R

RDMS Relational Database Management System

S

SRAM Static RAM (Random-Access Memory)

SQL Structured Query Language

U

URL Uniform Resource Locator

V

VBScript Microsoft Visual Basic Scripting Edition

W

WWW World Wide Web

X

XML Extensible Markup Language

XSLT eXtensible Stylesheet Language Transformations

ΠΑΡΑΡΤΗΜΑ

Α. Αναλυτική Περιγραφή της Βάσης Δεδομένου MySQL μέσα στις Εφαρμογές

1) Δισκογραφικές Εταιρείες (Ελλάδα-Κρήτης)

Ο πίνακας που χρησιμοποιείται για αυτήν την εφαρμογή είναι:

```
CREATE TABLE `diskografies_el` (  
  `id` tinyint(4) NOT NULL auto_increment,  
  `name` varchar(200) NOT NULL default '',  
  `address` varchar(150) NOT NULL default '',  
  `telephone` varchar(200) NOT NULL default '',  
  `picture` varchar(100) NOT NULL default '',  
  PRIMARY KEY (`id`)  
);
```

2) Πρόγραμμα Εβδομάδας

```
CREATE TABLE `programma_m_to_fr` (  
  `id` tinyint(4) NOT NULL auto_increment,  
  `date` varchar(20) NOT NULL default "",  
  `title` varchar(100) NOT NULL default "",  
  `text` varchar(255) NOT NULL default "",  
  PRIMARY KEY (`id`)  
);
```

3) Τοπ 40 (Ελληνικό, Ξένο, Κρητικό)

```
CREATE TABLE `top40_greek` (  
  `id` tinyint(4) NOT NULL auto_increment,  
  `titlos_tragoudiou` varchar(200) NOT NULL default "",  
  `tragoudistis` varchar(200) NOT NULL default "",  
  `thesi` int(10) NOT NULL default '0',  
  `image_thesis` varchar(10) NOT NULL default "",  
  `etairia` varchar(200) NOT NULL default "",  
  `thesi_proigoumeni_ebd` varchar(200) NOT NULL default "",  
  PRIMARY KEY (`id`)  
);
```

4) Μαντινάδες

```
CREATE TABLE `mantinades_asteies` (  

```

```

`id` tinyint(4) NOT NULL auto_increment,
`keimeno` varchar(250) NOT NULL default "",
`onoma` varchar(50) NOT NULL default "",
`date` date NOT NULL default '0000-00-00',
PRIMARY KEY (`id`)
);

```

5) Παραγωγοί

```

CREATE TABLE `prosopiko` (
`id` tinyint(4) NOT NULL auto_increment,
`name` varchar(150) NOT NULL default "",
`surname` varchar(150) NOT NULL default "",
`email` varchar(150) NOT NULL default "",
`telephone` varchar(100) NOT NULL default "",
`specialize` varchar(200) NOT NULL default "",
`commnts` varchar(255) NOT NULL default "",
`picture_sm` varchar(255) NOT NULL default "",
PRIMARY KEY (`id`)
);

```

6) Μουσικά Νέα

```

CREATE TABLE `music_news` (
`id` tinyint(4) NOT NULL auto_increment,
`foto` varchar(200) NOT NULL,
`text_mikro` varchar(200) NOT NULL,
`text_megalo` varchar(1200) NOT NULL,
`title` varchar(100) NOT NULL,
PRIMARY KEY (`id`)
);

```

7) Ψηφοφορία

Πίνακας p_choices

```

CREATE TABLE `p_choices` (
`ID` int(10) NOT NULL auto_increment,
`answer` varchar(100) NOT NULL default "",
`votes` bigint(21) NOT NULL default '0',
PRIMARY KEY (`ID`)
);

```

Πίνακας p_question

```

CREATE TABLE `p_question` (
`ID` int(21) NOT NULL auto_increment,
`question` varchar(100) NOT NULL default "",

```

```
PRIMARY KEY (`ID`)  
);
```

Πίνακας p_ip

```
CREATE TABLE `p_ip` (  
  `ID` bigint(21) NOT NULL auto_increment,  
  `IP` varchar(40) NOT NULL default "",  
  PRIMARY KEY (`ID`)  
);
```

8) Πίνακας στοιχείων Διαχειριστή

```
CREATE TABLE `admin_personel` (  
  `id` tinyint(4) NOT NULL auto_increment,  
  `userid` varchar(100) NOT NULL default "",  
  `password` varchar(100) NOT NULL default "",  
  `firstname` varchar(100) NOT NULL default "",  
  `email` varchar(100) NOT NULL default "",  
  `lastname` varchar(100) NOT NULL default "",  
  `comments` varchar(200) NOT NULL default "",  
  PRIMARY KEY (`id`)  
);
```

B. ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΚΩΔΙΚΑ PHP ΜΕΣΑ ΣΤΙΣ ΕΦΑΡΜΟΓΕΣ

1) Δισκογραφικές Εταιρείες (Ελλάδα-Κρήτης)

Ο κώδικας για την δημιουργία της εφαρμογής `diskografies_el.php`:

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
//echo "OK, successfully connected";

mysql_select_db( ' melodyfm_gr ' );

$res = mysql_query( "SELECT count(*) c FROM diskografies_el" );
$row = mysql_fetch_array( $res );
if( $row['c'] == 0 ) {
    echo "Δεν υπάρχουν Δισκογραφίες Ελλάδας<br>";
} else {
    echo "<table border=0 width=500>";

    echo "<tr><th colspan=5><h1>Δισκογραφίες Ελλάδας</h1></th></tr>";
    echo"<tr><td height=30></td></tr>";
    echo"<tr>";
    echo"<td align=center><H3>Όνομα</H3></td>";
    echo"<td align=center><H3>Logo Εταιρειας</H3></td>";
    echo"<td align=center><H3>Διεύθυνση</H3></td>";
    echo"<td align=center><H3>Τηλέφωνο</H3></td>";
    echo"</tr>";
    echo"<tr><td height=10></td></tr>";
    $res = mysql_query( "SELECT * FROM diskografies_el" );

    while( $row = mysql_fetch_array( $res ) ) {
        echo "<tr>";
        echo "<td align=center>";
        echo "<b>";
        echo $row['name'];
        echo "</b >";
        echo "<td align=center>";
            if( $row['picture'] ) {
                echo "<img src=\"uploads/diskografies_el/" .
$row['picture'] . "\" width=40 >";
            } else {
                echo "&nbsp;";
            }
        echo "</td>";
        echo "<td align=center>";
    }
}
```

```

        echo $row['address'];
        echo "</td>";
        echo "<td align=center width=70>";
        echo $row['telephone'];
        echo "</td>";
        echo "<td><a href=\"edit_diskografies_el.php?id=" . $row['id'] . "\"><img
src=\"images/pen.gif\" border=0 ></a>&nbsp;<a
href=\"delete_diskografies_el.php?id=" . $row['id'] . "\"><img
src=\"images/del.gif\" border=0 ></a></td>";
        echo "</tr>";

    }
    echo "</table>";
}
mysql_disconnect;
?>

```

Διαγραφή δεδομένων από τον πίνακα της `diskografies_el`:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id_to_delete = $_HTTP_GET_VARS['id'];
//echo "id_to_delete='$id_to_delete'<br>";

$query = "DELETE FROM diskografies_el WHERE id=$id_to_delete";
$res = mysql_query( $query );

mysql_disconnect;

header( "location: diskografies_el.php" );
?>

```

Φορμα διόρθωσης δεδομένων από Δισκογραφικές Εταιρείες:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id = $_HTTP_GET_VARS['id'];

```



```

$upload_dir = 'uploads/diskografies_el';
$upload_name = basename($_FILES['picture']['name']);
if( move_uploaded_file( $_FILES['picture']['tmp_name'],
$upload_dir.'/'.$upload_name ) ) {
    //echo "INFO: file successfully uploaded<br>";
}

//echo "DEBUG: name='$name', address='$address', telephone='$telephone'";

$query = "SELECT * FROM diskografies_el WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$name = $row['name'];
$address = $row['address'];
$telephone = $row['telephone'];
$picture = $row['picture'];

mysql_disconnect;
?>
<form method="post" name="edit_anakoinwsh"
action="update_diskografies_el.php" enctype="multipart/form-data">
<table border=0 cellspacing="0">
<tr><th colspan=2><h1>Επεξεργασία Δισκογραφίας Ελλάδας</h1></th></tr>
<tr><td>Όνομα</td><td><input type="text" name="name" length=32
value="<?=$name?>"></td></tr>
<tr><td>Διεύθυνση</td><td><textarea name="address" rows=4
cols=40><?=$address?></textarea></td></tr>
<tr><td>Τηλέφωνο</td><td><textarea name="telephone" rows=5
cols=40><?=$telephone?></textarea></td></tr>
<tr><td>Logo Εταιρείας</td><td><input type="file" name="picture" size=48
value="<?=$picture?>"></td></tr>
<input type="hidden" name="id" value="<?=$id?>">
<tr><td><input type="submit" value="Καταχώρηση >>"></td></tr>
</table>
</form>

```

Εισαγωγή των διορθωμένων δεδομένων από την φόρμα μέσω της update_diskografies_el.php:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id = $_HTTP_POST_VARS['id'];

```

```

$new_name = $HTTP_POST_VARS['name'];
$new_address = $HTTP_POST_VARS['address'];
$new_telephone = $HTTP_POST_VARS['telephone'];

$query = "SELECT * FROM diskografies_el WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$sold_name = $row['name'];
$sold_address = $row['address'];
$sold_telephone = $row['telephone'];

//echo "DEBUG: information for uploaded file:<br>";
//print_r( $_FILES );
//echo "name=" . $_FILES['picture']['name'] . "<br>";
//echo "type=" . $_FILES['picture']['type'] . "<br>";
//echo "size=" . $_FILES['picture']['size'] . "<br>";
//echo "tmp_name=" . $_FILES['picture']['tmp_name'] . "<br>";
//echo "error=" . $_FILES['picture']['error'] . "<br>";

$upload_dir = 'uploads/diskografies_el';
$upload_name = basename($_FILES['picture']['name']);
if( move_uploaded_file( $_FILES['picture']['tmp_name'],
$upload_dir.'/'.$upload_name ) ) {
    //echo "INFO: file successfully uploaded<br>";
}

$set_clause = "";
if( $new_name != $sold_name ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "name='$new_name'";
}
if( $new_address != $sold_address ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "address='$new_address'";
}
if( $new_telephone != $sold_telephone ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "telephone='$new_telephone'";
}
$set_clause .= ( $set_clause ? ", " : "" ) . "picture='$upload_name'";

$query = "UPDATE diskografies_el SET " . $set_clause . " WHERE id=$id";

//echo "DEBUG: executing query $query<br>";

$res = mysql_query( $query );

mysql_disconnect;

header( "location: diskografies_el.php" );
?>

```

Τρόπος Εισαγωγής Νέων δεδομένων

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );

$name = $_HTTP_POST_VARS['name'];
$address = $_HTTP_POST_VARS['address'];
$telephone = $_HTTP_POST_VARS['telephone'];

$upload_dir = 'uploads/diskografies_el';
$upload_name = basename($_FILES['picture']['name']);
if( move_uploaded_file( $_FILES['picture']['tmp_name'],
$upload_dir.'/'.$upload_name ) ) {
    //echo "INFO: file successfully uploaded<br>";
}

//echo "DEBUG: name='$name', address='$address', telephone='$telephone'";

mysql_select_db( 'melodyfm_gr' );

if( $_FILES['picture']['size'] > 0 ) {
    $query = "insert into diskografies_el (name,address,telephone,picture)
values ('$name','$address','$telephone','$upload_name')";
} else {
    $query = "insert into diskografies_el (name,address,telephone) values
('$name','$address','$telephone')";
}
//echo "DEBUG: executing query $query";
$res = mysql_query( $query );

mysql_disconnect;

header( "location: diskografies_el.php" );

?>
```

2) Πρόγραμμα Εβδομάδας

Ο κώδικας για την δημιουργία της εφαρμογής programma_m_to_fr.php

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
//echo "OK, successfully connected";

mysql_select_db( 'melodyfm_gr' );
```

```

$res = mysql_query( "SELECT count(*) c FROM programma_m_to_fr " );
$row = mysql_fetch_array( $res );
if( $row['c'] == 0 ) {
    echo "Δεν υπάρχει πρόγραμμα Δευτέρα έως Παρασκευή<br>";
} else {
    echo "<table border=0 width=500>";

    echo "<tr><th colspan=5><h1>Πρόγραμμα Δευτέρα έως
Παρασκευή</h1><br><br><br></th></tr>";

```

```

$res = mysql_query( "SELECT * FROM programma_m_to_fr " );

```

```

while( $row = mysql_fetch_array( $res ) ) {
    echo "<tr>";
    echo "<td valign=top width=100><b>";
    echo $row['date'];
    echo "</b></td>";
    echo "<td width=20 >";
    echo "</td>";
    echo "<td width=300>&nbsp;&nbsp;&nbsp;<b>";
    echo $row['title'];
    echo "</b>&nbsp;&nbsp;&nbsp;";
    echo $row['text'];
    echo "</td>";
    echo "<td><a href=\"edit__programma_m_fr.php?id=" . $row['id'] .
\"\"><img src=\"images/pen.gif\" border=0 ></a>&nbsp;&nbsp;&nbsp;<a
href=\"delete_programma_m_fr.php?id=" . $row['id'] . "\"><img
src=\"images/del.gif\" border=0 ></a></td>";
    echo "</tr>";
    echo "<tr><td height=30><BR><BR></td></tr>";
}
echo "</table></body></html>";
}
mysql_disconnect;
?>

```

Διαγραφή δεδομένων από τον πίνακα της programma_m_to_fr:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id_to_delete = $HTTP_GET_VARS['id'];
//echo "id_to_delete='$id_to_delete'<br>";

```

```

$query = "DELETE FROM programma_m_to_fr WHERE id=$id_to_delete";
$res = mysql_query( $query );

mysql_disconnect;

header( "location: programma_m_to_fr l.php" );
?>

```

Φορμα διόρθωσης δεδομένων από Πρόγραμμα Δευτέρα-Παρασκευή:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melody' );
//echo "OK, successfully connected";

$id = $_HTTP_GET_VARS['id'];

$query = "SELECT * FROM programma_m_to_fr WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$date = $row['date'];
$text = $row['text'];
$title = $row['title'];

mysql_disconnect;
?>
<form method="post" name="edit_anakoinwsh"
action="update_programma_m_to_fr.php" enctype="multipart/form-data">
<table border=0 cellspacing="0">
<tr><th colspan=2><h1>Επεξεργασία Προγράμματος Δευτέρας-
Παρασκευή</h1></th></tr>
<tr><td>Ωρα</td><td><input type="text" name="date" length=32 rows=5
value="<?=$date?>"></td></tr>
<tr><td>Τίτλος</td><td><textarea name="title" rows=2
cols=20><?=$title?></textarea></td></tr>
<tr><td>Σχόλια</td><td><textarea name="text" rows=2
cols=20><?=$text?></textarea></td></tr>
<input type="hidden" name="id" value="<?=$id?>">
<tr><td><input type="submit" value="Καταχώρηση >>"></td></tr>
</table>
</form>

```

**Εισαγωγή των διορθωμένων δεδομένων από την φόρμα μέσω της
update_programma_m_to_fr.php:**

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not connect!"
);
mysql_select_db( 'melody' );
//echo "OK, successfully connected";

$id = $HTTP_POST_VARS['id'];
$new_date = $HTTP_POST_VARS['date'];
$new_title = $HTTP_POST_VARS['title'];
$new_text = $HTTP_POST_VARS['text'];

$query = "SELECT * FROM programma_m_to_fr WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$sold_date = $row['date'];
$sold_title = $row['title'];
$sold_text = $row['text'];

$set_clause = "";
if( $new_date != $sold_date ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "date      ='$new_date'";
}
if( $new_title != $sold_title ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "title='$new_title'";
}
if( $new_text != $sold_text ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "text='$new_text'";
}

$query = "UPDATE programma_m_to_fr SET " . $set_clause . " WHERE
id=$id";

//echo "DEBUG: executing query $query<br>";

$res = mysql_query( $query );

mysql_disconnect;

header( "location: programma_m_fr.php" );
?>
```

Τρόπος Εισαγωγής Νέων δεδομένων

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not connect!"
);

$date = $_HTTP_POST_VARS['date'];
$title = $_HTTP_POST_VARS['title'];
$text = $_HTTP_POST_VARS['text'];
mysql_select_db( 'melody' );

$query = "insert into programma_m_to_fr (date,text,title) values
('$date','$text','$title)";

$res = mysql_query( $query );

mysql_disconnect;

header( "location: programma_m_fr.php" );

?>
```

3) Τοπ 40 (Ελληνικό, Ξένο, Κρητικό)

Ο κώδικας για την δημιουργία της εφαρμογής programma_m_to_fr.php

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
//echo "OK, successfully connected";

mysql_select_db( 'melodygm_gr' );

$res = mysql_query( "SELECT count(*) c FROM top40_greek ORDER BY
thesi desc" );
$row = mysql_fetch_array( $res );
if( $row['c'] == 0 ) {
echo "Δεν υπάρχει Top 40 Ελληνικό<br>";
} else {

echo "<h1>Top 40 Ελληνικό</h1>";
echo "<table border=0 width=560 style=\"border-left:1px dashed #9db3da;
border-top:1px dashed #9db3da; border-bottom:1px dashed #9db3da; border-
right:1px dashed #9db3da; \">";
echo"<tr>";
echo"<td align=center><H3>Θέση</H3></td>";
echo"<td align=center><H3>Όνομα Τραγουδιού </H3></td>";
```

```

echo"<td align=center><H3>Τραγουδιστής</H3></td>";
echo"<td align=center><H3>Προηγ. Εβδομ</H3></td>";
echo"<td align=center><H3>Εταιρία</H3></td>";
echo"</tr>";
echo"<tr valign=top><td colspan=6 ><hr></td></tr>";
$res = mysql_query( "SELECT * FROM top40_greek ORDER BY thesi asc"
);

while( $row = mysql_fetch_array( $res ) ) {
echo "<tr >";
echo "<td align=center style=\" border-bottom:1px dashed #9db3da; border-
right:1px dashed #9db3da; \">";
echo "<b>";
echo $row['thesi'];
echo "</b></td>";
echo "<td align=center style=\" border-bottom:1px dashed #9db3da; border-
right:1px dashed #9db3da; \">";
echo $row['titlos_tragoudiou'];
echo "</td>";
echo "<td align=center style=\" border-bottom:1px dashed #9db3da; border-
right:1px dashed #9db3da; \">";
echo $row['tragoudistis'];
echo "</td>";
echo "<td align=center style=\" border-bottom:1px dashed #9db3da; border-
right:1px dashed #9db3da; \">";
echo $row['thesi_proigoumeni_ebd'];
echo "</td>";

echo "<td align=center style=\" border-bottom:1px dashed #9db3da; \">";
echo $row['etairia'];
echo "</td>";
echo "<td><a href=\"edit_top_greek.php?id=" . $row['id'] . "\"" ><img
src=\"images/pen.gif\" border=0 ></a>&nbsp;<a
href=\"delete_top_greek.php?id=" . $row['id'] . "\"" ><img src=\"images/del.gif\"
border=0 ></a></td>";

echo "</tr>";

}
echo "</table>";

}
mysql_disconnect;
?>

```


Διαγραφή δεδομένων από τον πίνακα της top40_greek:

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id_to_delete = $_HTTP_GET_VARS['id'];
//echo "id_to_delete='$id_to_delete'<br>";

$query = "DELETE FROM top40_greek WHERE id=$id_to_delete";
$res = mysql_query( $query );

mysql_disconnect;

header( "location: top_greek.php" );
?>
```

Μορφή φόρμας διόρθωση πληροφοριών του top40_greek

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not connect!"
);
mysql_select_db( 'melody' );
//echo "OK, successfully connected";

$id = $_HTTP_GET_VARS['id'];

$query = "SELECT * FROM top40_greek WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$title_tragoudiou = $row['title_tragoudiou'];
$tragoudistis = $row['tragoudistis'];
$thesi = $row['thesi'];
$thesi_proigoumeni_ebd = $row['thesi_proigoumeni_ebd'];
$etairia = $row['etairia'];
$image_thesis = $row['image_thesis'];

mysql_disconnect;
?>
<form method="post" name="edit_anakoinwsh" action="update_top_greek.php"
enctype="multipart/form-data">
<table border=0 cellspacing="0">
<tr><td>Όνομα Τραγουδιού</td>
```

```

<td ><textarea name="titlos_tragoudiou"
><?=$titlos_tragoudiou?></textarea></td>
</tr>
<tr><td>Τραγουδιστής</td>
<td ><textarea name="tragoudistis" > <?=$tragoudistis?></textarea></td>
</tr>
<tr><td >Θέση</td><td colspan="3"><select name="thesi" size="1">
<option value=" 1 ">1 </option>
<option value=" 2 ">2 </option>
.....
.....
.....
<option value="40 ">40 </option>

</select></td></tr>

<tr><td>Προηγ. Εβδομ</td><td colspan="3"><textarea
name="thesi_proigoumeni_ebd" >
<?=$thesi_proigoumeni_ebd?></textarea></td></tr>

<tr><td>Εταιρία</td><td colspan="3"><textarea name="etairia" >
<?=$etairia?></textarea></td></tr>
<tr><td align="right"><br><input type="submit" value="Καταχώρηση >>"
></td></tr>
<tr><td height="15" colspan="3"><input type="hidden" name="id"
value="<?=$id?>"></td></tr>
</table>

</form>

```

Εισαγωγή των διορθωμένων δεδομένων από την φόρμα μέσω της update_programma_m_to_fr.php:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not connect!"
);
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id = $HTTP_POST_VARS['id'];
$new_titlos_tragoudiou = $HTTP_POST_VARS['titlos_tragoudiou'];
$new_tragoudistis = $HTTP_POST_VARS['tragoudistis'];
$new_thesi = $HTTP_POST_VARS['thesi'];
$new_thesi_proigoumeni_ebd = $HTTP_POST_VARS['thesi_proigoumeni_ebd'];
$new_etairia = $HTTP_POST_VARS['etairia'];

$query = "SELECT * FROM top40_greek WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );

```

```

Sold_titlos_tragoudiou = $row['titlos_tragoudiou'];
Sold_tragoudistis = $row['tragoudistis'];
Sold_thesi = $row['thesi'];
Sold_thesi_proigoumeni_ebd = $row['thesi_proigoumeni_ebd'];
Sold_etairia = $row['etairia'];

$set_clause = "";
if( $new_titlos_tragoudiou != $sold_titlos_tragoudiou ) {
$set_clause .= ( $set_clause ? ", " : "" ) .
"titlos_tragoudiou='new_titlos_tragoudiou'";
}
if( $new_tragoudistis != $sold_tragoudistis ) {
$set_clause .= ( $set_clause ? ", " : "" ) . "tragoudistis='$new_tragoudistis'";
}
if( $new_thesi != $sold_thesi ) {
$set_clause .= ( $set_clause ? ", " : "" ) . "thesi='$new_thesi'";
}
if( $new_thesi_proigoumeni_ebd != $sold_thesi_proigoumeni_ebd ) {
$set_clause .= ( $set_clause ? ", " : "" ) .
"thesi_proigoumeni_ebd='$new_thesi_proigoumeni_ebd'";
}
if( $new_etairia != $sold_etairia ) {
$set_clause .= ( $set_clause ? ", " : "" ) . "etairia='$new_etairia'";
}

$query = "UPDATE top40_greek SET " . $set_clause . " WHERE id=$id";

$res = mysql_query( $query );

mysql_disconnect;

header( "location: top_greek.php" );
?>

```

Τρόπος Εισαγωγής Νέων δεδομένων

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not connect!"
);

$titlos_tragoudiou = $HTTP_POST_VARS['titlos_tragoudiou'];
$tragoudistis = $HTTP_POST_VARS['tragoudistis'];
$thesi = $HTTP_POST_VARS['thesi'];
$thesi_proigoumeni_ebd = $HTTP_POST_VARS['thesi_proigoumeni_ebd'];
$etairia = $HTTP_POST_VARS['etairia'];

```

```

$image_thesis = $_HTTP_POST_VARS['image_thesis'];

mysql_select_db( 'melodyfm_gr' );

$query = "insert into top40_greek
(titlos_tragoudiou,tragoudistis,thesi,image_thesis,thesi_proigoumeni_ebd, etairia)
values
('$titlos_tragoudiou','$stragoudistis','$thesi','$image_thesis','$thesi_proigoumeni_eb
d','etairia)";

$res = mysql_query( $query );

mysql_disconnect;

header( "location: top_greek.php" );

?>

```

4) Μαντινάδες

Φόρμα αποστολής μαντινάδας

```

<form method="post" action="sendemail.php">

Όνομα: <br />
<input type="text" name="visitor" size="35" />
<br />
Email:<br />
<input type="text" name="visitormail" size="35" />
<br /> <br />
<br />
Κατηγορία Μαντινάδας:<br />
<select name="attn" size="1">
<option value=" Αστείες ">Αστείες </option>
<option value=" Ερωτικές ">Ερωτικές </option>
<option value=" Χωρισμού ">Χωρισμού </option>
<option value=" Στρατιωτικές ">Στρατιωτικές </option>
<option value=" Διάφορες ">Διάφορες </option>
</select>
<br /><br />
Mail Message:
<br />
<textarea name="notes" rows="4" cols="40"></textarea>
<br />
<input type="submit" value="Send Mail" />

```

```
<br />
</form>
</td></tr>
</table>
```

Τρόπος αποστολής της φόρμας μέσω email() (sendemail.php)

```
<?
$id = $_HTTP_POST_VARS['id'];

$visitor = $_POST['visitor'];
$visitormail = $_POST['visitormail'];
$notes = $_POST['notes'];
$attn = $_POST['attn'];
$email_headers = "";
$email_subject = "Mantinades";
$email="radio@melodyfm.gr"

$message = "
Name: $visitor \n
Email: $visitormail \n
Theme: $attn \n
Comments: $notes \n
";
    mail( $email, $email_subject, $message );
echo"Thank you";
?>
```

Εισαγωγής Νέας μαντινάδας στο template του Διαχειριστή

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
```

```
$name = $_HTTP_POST_VARS['onoma'];
$keimeno = $_HTTP_POST_VARS['keimeno'];
$date = $_HTTP_POST_VARS['date'];
//echo "DEBUG: name='$name', keimeno='$keimeno', date='$date'";
mysql_select_db( 'melodyfm_gr' );
```

```
$query = "insert into mantinades_asteies (onoma,keimeno,date) values
('$name','$keimeno','$date)";
```

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

```
mysql_disconnect;
```

```
header( "location: mantinades_asteies.php" );
```

```
?>
```

Διαγραφή δεδομένων από τον πίνακα της mantinades_asteies:

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id_to_delete = $_HTTP_GET_VARS['id'];
//echo "id_to_delete='$id_to_delete'<br>";

$query = "DELETE FROM mantinades_asteies WHERE id=$id_to_delete";
$res = mysql_query( $query );

mysql_disconnect;

header( "location: mantinades_asteies.php" );
?>
```

Μορφή φόρμας διόρθωση πληροφοριών του top40_greek

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id = $_HTTP_GET_VARS['id'];

$query = "SELECT * FROM mantinades_asteies WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$name = $row['onoma'];
$keimeno = $row['keimeno'];
$date = $row['date'];

mysql_disconnect;
?>
<form method="post" name="edit_anakoinwsh"
action="update_mantinades_asteies.php" enctype="multipart/form-data">
<table border=0 cellpadding="0">
<tr><th colspan=2><h1>Επεξεργασία Μαντινάδας Αστείας</h1></th></tr>
<tr><td>Μαντινάδα</td><td><textarea name="keimeno" rows=6
cols=40><?=$keimeno?></textarea></td></tr>
```

```

<tr><td>Αποστολέας</td><td><textarea name="onoma" rows=2
cols=20><?=$name?></textarea></td></tr>
<tr><td>Ημερομηνία</td><td><textarea name="date" rows=2
cols=20><?=$date?></textarea></td></tr>
<input type="hidden" name="id" value="<?=$id?>">
<tr><td><input type="submit" value="Καταχώρηση >>"></td></tr>
</table>
</form>

```

Εισαγωγή των διορθωμένων δεδομένων από την φόρμα μέσω της update_mantinades_asteies.php:

```

<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id = $_HTTP_POST_VARS['id'];
$new_onoma = $_HTTP_POST_VARS['onoma'];
$new_keimeno = $_HTTP_POST_VARS['keimeno'];
$new_date = $_HTTP_POST_VARS['date'];

$query = "SELECT * FROM mantinades_asteies WHERE id=$id";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$sold_onoma = $row['onoma'];
$sold_keimeno = $row['keimeno'];
$sold_date = $row['date'];

$set_clause = "";
if( $new_onoma != $sold_onoma ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "onoma      ='$new_onoma'";
}
if( $new_keimeno != $sold_keimeno ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "keimeno='$new_keimeno'";
}
if( $new_date != $sold_date ) {
    $set_clause .= ( $set_clause ? ", " : "" ) . "date='$new_date'";
}

$query = "UPDATE mantinades_asteies SET " . $set_clause . " WHERE
id=$id";

//echo "DEBUG: executing query $query<br>";

$res = mysql_query( $query );

```

```
mysql_disconnect;
```

```
header( "location: mantinades_asteies.php" );
```

```
?>
```

5) Παραγωγοί

Δομή της σελίδας paragogoi.php

```
<?php
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
//echo "OK, successfully connected";

mysql_select_db( 'melodyfm_gr' );

$res = mysql_query( "SELECT count(*) c FROM prosopiko" );
$row = mysql_fetch_array( $res );
if( $row['c'] == 0 ) {
    echo "There are no Προσωπικό<br>";
} else {
    echo "<table border=0 >";

    echo "<tr><th colspan=5><h1>Προσωπικό</h1></th></tr>";
    echo "<tr><td height=30></td></tr>";

    $res = mysql_query( "SELECT * FROM prosopiko" );

    while( $row = mysql_fetch_array( $res ) ) {
        echo "<tr>";
        echo "<td>";
            if( $row['picture_sm'] ) {
                echo "<img src=\"uploads/prosopiko/" . $row['picture_sm']
                    . "\" width=100 >";
            } else {
                echo "&nbsp;";
            }
            echo "</td>";
            echo "<td align=center width=150>
                <a href=\"show_paragogoi.php?id=" . $row['id'] . "\">
                .$row['specialize'] . "</a></td>";

            echo "</tr>";
        echo "<tr>";
        echo "<td colspan=2 width=150 align=left>
```



```

    <a href="show_paragogoi.php?id=" . $row['id'] . "\"> . $row['name'] . ' ' .
    $row['surname'] . "</a></td>";
    echo "<td rowspan=2>
    <a href="edit_paragogoi.php?id=" . $row['id'] . "\"></a>&nbsp;";
    <a href="delete_paragogoi.php?id=" . $row['id'] . "\"></a></td>";
    echo "</tr>";
    echo "<tr><td height=30></td></tr>";
}
echo "</table>";
}
mysql_disconnect;
?>

```

Δομή της σελίδας show_paragogoi.php μέσω του id.

```

<?
$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );
//echo "OK, successfully connected";

$id = $_HTTP_GET_VARS['id'];
echo "<table border=0 width=500>";
echo "<tr><th colspan=3></th></tr>";

$res = mysql_query( "SELECT * FROM prosopiko WHERE id=$id" );

$row = mysql_fetch_array( $res );
echo "<tr>";
    echo "<td valign=top>";
echo "<table border=0 width=200>";

echo "<tr valign=top>";
echo "<td align=left>";
    if( $row['picture_sm'] ) {
        echo "";
    } else {
        echo "&nbsp;";
    }
    echo "</td>";
echo "</td>";
echo "</tr>";
echo "<tr>";

```

```

echo "<td align=left><b>";
echo $row['name'];
    echo"&nbsp;";
echo $row['surname'];
echo"</b><br><br>";
echo"</td>";
echo "</tr>";

echo "<tr>";
echo "<td width=200 align=center>";
echo $row['commnets'];

echo"</td>";
echo"</tr>";
echo"</table>";
echo"</td>";
echo "<td>";
echo "<table border=0 width=200>";
echo"<tr>";
echo "<td>";
echo "<b>Στείλτε το μήνυμα σας</b><br>" ;
echo"</b><br><br>";
    echo"O &nbsp;";
    echo $row['name'];
    echo"&nbsp;";
echo $row['surname'];
echo"&nbsp; θα λάβει το μήνυμα σας και θα σας απαντήσει σύντομα";

echo"</td>";
echo "</tr>";

mysql_disconnect;
?>

<form method="post" action="contact_paragogoi.php"
enctype="multipart/form-data" >
<tr>
<td><br><br>
To όνομα σας<br>
<input type="text" name="name" length=100><br>
</td>
</tr>
<tr><td>To email σας
<br><input type="text" name="email" length=100><br>
<br></td></tr>
<tr><td>To μήνυμα σας<br><textarea name="comment" rows=3
cols=20></textarea><br></td></tr>
<tr><td><input type="submit" value="Αποστολή"></td></tr>
<input type="hidden" name="id" value="<?=$id?>">
</form>

```

```

<?php echo"</table>";
echo"</td>";
echo"</tr>";
echo"</table>";
?>

```

Δομή της σελίδας show_paragogoi.php μέσω του id.

```

<?
$id = $_HTTP_POST_VARS['id'];

$name = $_HTTP_POST_VARS['name'];
$email = $_HTTP_POST_VARS['email'];
$comments = $_HTTP_POST_VARS['comments'];
$email_subject = "S????a-????ματα";
$email_headers = "";

$conn = mysql_connect('localhost','xristina','xristina') || die( "could not
connect!" );
mysql_select_db( 'melodyfm_gr' );

$query = "SELECT * FROM prosopiko WHERE id='$id'";
$res = mysql_query( $query );
$row = mysql_fetch_array( $res );
$mail=$row['email'];
$message = "
Name: $name \n
Email: $email \n
Comments: $comments\n
";
    mail( $mail, $email_subject, $message );
    //echo "DEBUG: petixe email=$mail<br>";
mysql_disconnect;

header( "location: paragogoi.php" );
?>

```

6) Μουσικά Νέα

Ο κώδικας για την δημιουργία της εφαρμογής music_news.php

```

<?php
$res = mysql_query( "SELECT count(*) c FROM music_news" );
$row = mysql_fetch_array( $res );
if( $row['c'] == 0 ) {
    echo "There are no Μουσικά Νέα<br>";
} else {

```

```

echo "<table border=0 align='center' cellpadding='5' cellspacing='5' >";

echo "<tr><th colspan=5><h1>Μουσικά Νέα</h1></th></tr>";
echo"<tr><td height=30></td></tr>";

$res = mysql_query( "SELECT * FROM music_news " );

while( $row = mysql_fetch_array( $res ) ) {
    echo "<tr valign='top'>";
    echo "<td colspan='2'><b>";
    echo $row['title'] ;
    echo "</b></td>";
    echo "</tr>";

    echo "<tr valign='top'>";
    echo "<td>";
        if( $row['foto'] ) {
            echo "<img src=\"uploads/music/" . $row['foto'] . "\"
width=200 >";
        } else {
            echo "&nbsp;";
        }
    echo "</td>";
    echo "<td align=left width=150>";
    echo $row['text_mikro'];
    echo "<br>";
    echo "<a href=\"more_music_news.php?id=" . $row['id'] .
\">Μάθετε περισσότερα</a>";
    echo "</td>";

    echo "</tr>";
    echo "<tr >";
    echo "<td align='right' colspan='2'><a href=\"edit_music_news.php?id=" .
$row['id'] . "\"><img src=\"images/pen.gif\" border=0 ></a>&nbsp;<a
href=\"delete_music_news.php?id=" . $row['id'] . "\"><img src=\"images/del.gif\"
border=0 ></a></td>";
    echo "</tr>";
    echo "<tr valign='top'>";
    echo "<td colspan='2'><hr></td>";
    echo "</tr>";

    echo"<tr><td height=30></td></tr>";
}
echo "</table>";
}
mysql_disconnect;

```

?>

Δομή της σελίδας `more_music_news.php` μέσω του `id`.

```
<?php

$id = $_HTTP_GET_VARS['id'];
$res = mysql_query( "SELECT count(*) c FROM music_news" );
$row = mysql_fetch_array( $res );
if( $row['c'] == 0 ) {
    echo "There are no Μουσικά Νέα<br>";
} else {
    echo "<table border=0 align='center' cellpadding='5' cellspacing='5' >";

    echo "<tr><th colspan=5><h1>Μουσικά Νέα</h1></th></tr>";
    echo "<tr><td height=30></td></tr>";

    $res = mysql_query( "SELECT * FROM music_news WHERE id=$id" );

    while( $row = mysql_fetch_array( $res ) ) {

        echo "<tr valign='top'>";
        echo "<td>";
            if( $row['foto'] ) {
                echo "<img src='\"uploads/music/\" . $row['foto'] . \"\"
width=250 >";
            } else {
                echo "&nbsp;";
            }
        echo "</td>";
        echo "</tr>";
        echo "<tr valign='top'>";
        echo "<td height='10'></td>";
        echo "</tr>";
        echo "<tr>";
        echo "<td align=left width=250>";
        echo $row['text_megalo'];
        echo "<br>";
        echo "<a href='music_news.php'>Επιστροφή στα
προηγούμενα</a>";
        echo "</td>";

        echo "</tr>";
    }
}
```

```

        echo "<tr valign='top'>";
        echo "<td colspan='2'><hr></td>";
        echo "</tr>";

    }
    echo "</table>";
    }
    mysql_disconnect;
?>

```

7) Ψηφοφορία

Εισαγωγή Νεων επιλογών ερώτησης

```

include "connect.php";

print "<center><h3>Poll Admin Panel</h3></center><br>";
print "<center>";
print "<table border='0' cellspacing='20'>";
print "<tr><td width='25%' valign='top' rowspan='2'>";
    include 'left_voting.php';
print "</td>";
print "<td valign='top' width='75%'>";
    $checkpoll="SELECT * from P_question";

if(!get_magic_quotes_gpc())
{
    $_GET = array_map('mysql_real_escape_string', $_GET);
    $_POST = array_map('mysql_real_escape_string', $_POST);
    $_COOKIE = array_map('mysql_real_escape_string', $_COOKIE);
}
else
{
    $_GET = array_map('stripslashes', $_GET);
    $_POST = array_map('stripslashes', $_POST);
    $_COOKIE = array_map('stripslashes', $_COOKIE);
    $_GET = array_map('mysql_real_escape_string', $_GET);
    $_POST = array_map('mysql_real_escape_string', $_POST);
    $_COOKIE = array_map('mysql_real_escape_string', $_COOKIE);
}
$checkpoll2=mysql_query($checkpoll);
    while($checkpoll3=mysql_fetch_array($checkpoll2))
    {
        $sisthere=$checkpoll3[question];
    }
if($sisthere)

```

```

{
print "<center><h3>Create Poll</h3></center><br>";
print "<center>";
print "<table border='0' width='70%' cellpadding='20'>";
print "<tr>";
print "<td valign='top' width='75%'>";
print "There is already an existing poll, you cannot create another poll" ;
    print "<br><br>Question is:&nbsp;";
    echo "<b>$isthere</b>";
    print "</td></tr></table>";
}
else
{

if (isset($_POST['submit']))
{
    $pollname=$_POST['pollname'];
    $createpoll="INSERT into P_question(question) values('$pollname')";
    mysql_query($createpoll);
    print "<center><h3>Create Poll</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellpadding='20'>";
    print "<tr>";
    print "<td valign='top' width='75%'>";
    print "<b>Poll Created</b><br><br><br>";
        echo "<b>Ερώτηση της
Ψηφοφορίας:</b><br><br>";
        echo "$isthere";
    print "</td></tr></table>";

}
else
{
    print "<center><h3>Create Poll</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellpadding='20'>";
    print "<tr>";
    print "<td valign='top' width='75%'>";
    print "<form action='addchoice.php' method='post'>";
    print "Poll Question: <input type='text' name='pollname' size='20'>";
    print "<input type='submit' name='submit' value='Create Poll'>";
    print "</td></tr></table>";

}
}

    print "</td></tr> </table>";
print "</center>";

```

```
mysql_disconnect;  
?>
```

Διόρθωση επιλογών ερώτησης

```
<?php  
include "connect.php";  
  
print "<center><h3>Poll Admin Panel</h3></center><br>";  
print "<center>";  
print "<table border='0' cellspacing='20'>";  
print "<tr><td width='25%' valign='top' rowspan='2'>";  
include 'left_voting.php';  
print "</td>";  
print "<td valign='top' width='75%'>";  
if(isset($_POST['submit']))  
{  
$ID=$_POST['ID'];  
$choice=$_POST['choice'];  
$upans="UPDATE P_choices set answer='$choice' where ID='$ID';"  
mysql_query($upans) or die("Could not update answer");  
print "Answer Updated<br><br>";  
print "<table  
border='0'><tr><td><b>Answer</b></td><td><b>Edit</b></td></tr>";  
$getans="SELECT * from P_choices";  
$getans2=mysql_query($getans) or die("Could not get answer");  
while($getans3=mysql_fetch_array($getans2))  
{  
print "<tr><td>$getans3[answer]</td><td><a  
href='editchoice.php?ID=$getans3[ID]'>Edit</a></td></tr>";  
}  
print "</table>";  
}  
else if(isset($_GET['ID']))  
{  
print "Edit choice:<br>";  
$ID=$_GET['ID'];  
$getchoice="SELECT * from P_choices where ID='$ID';"  
$getchoice2=mysql_query($getchoice) or die("Could not get choice");  
$getchoice3=mysql_fetch_array($getchoice2);  
print "<form action='editchoice.php' method='post'>";  
print "<input type='hidden' name='ID' value='$ID'>";  
print "<input type='text' name='choice' value='$getchoice3[answer]'><br>";  
print "<input type='submit' name='submit' value='Edit Poll  
Choice'></form>";  
  
}
```



```

else
{
    print "<table
border='0'><tr><td><b>Answer</b></td><td><b>Edit</b></td></tr>";
    $getans="SELECT * from P_choices";
    $getans2=mysql_query($getans) or die("Could not get answer");
    while($getans3=mysql_fetch_array($getans2))
    {
        print "<tr><td>$getans3[answer]</td><td><a
href='editchoice.php?ID=$getans3[ID]'>Edit</a></td></tr>";
    }
    print "</table>";

}

        print "</td></tr> </table>";
        print "</center>";

mysql_disconnect;

?>

```

Διαγραφή επιλογών ερώτησης

```

<?php
include "connect.php";

print "<center><h3>Poll Admin Panel</h3></center><br>";
print "<center>";
print "<table border='0' cellspacing='20'>";
print "<tr><td width='25%' valign='top' rowspan='2'>";
    include 'left_voting.php';
print "</td>";
print "<td valign='top' width='75%'>";
    if(isset($_POST['submit']))
    {
        $choice=$_POST['choice'];
        $delchoice="DELETE from P_choices where answer='$choice'";
        mysql_query($delchoice);
        print "<center><h3>Delete Poll Choice</h3></center><br>";
        print "<center>";
        print "<table border='0' width='70%' cellspacing='20'>";
        print "<tr>";
        print "<td valign='top' width='75%'>";
        print "Poll Choice Deleted<br>";
        print "<form action='deletechoice.php' method='post'>";
    }

```

```

        $result = mysql_query("SELECT answer FROM P_choices");
        $row_array=mysql_fetch_row($result);

        $string='<select    name=choice><option    value="">Select    option    to
delete</option>';

        for ($i=0; $i < mysql_num_rows($result); $i++) {

            if ($row_array[0] == "") {
                $row_array=mysql_fetch_row($result);
            } else {
                $string .='<option value="'. $row_array[0]."'>'. $row_array[0];
                $row_array=mysql_fetch_row($result);
            }

        }

        $string .='</SELECT>';

        echo $string;
        print "<br><input type='submit' name='submit' value='submit'>";
        print "</form></td></tr></table>";
    }

    else
    {
        print "<center><h3>Delete Poll Choice</h3></center><br>";
        print "<center>";
        print "<table border='0' width='70%' cellspacing='20'>";
        print "<tr>";
        print "<td valign='top' width='75%'>";
        print "<form action='deletechoice.php' method='post'>";
        $result = mysql_query("SELECT answer FROM P_choices");
        $row_array=mysql_fetch_row($result);

        $string='<select    name=choice><option    value="">Select    option    to
delete</option>';

        for ($i=0; $i < mysql_num_rows($result); $i++) {

            if ($row_array[0] == "") {
                $row_array=mysql_fetch_row($result);
            } else {
                $string .='<option value="'. $row_array[0]."'>'. $row_array[0];
                $row_array=mysql_fetch_row($result);
            }

        }

    }

```

```

$string .= '</SELECT>';

echo $string;
print "<br><input type='submit' name='submit' value='submit'>";
print "</form></td></tr></table>";
}

```

```

        print "</td></tr> </table>";
    print "</center>";

```

```

mysql_disconnect;
?>

```

Δημιουργία ερώτησης

```

<?php
include "connect.php";

print "<center><h3>Poll Admin Panel</h3></center><br>";
print "<center>";
print "<table border='0' cellspacing='20'>";
print "<tr><td width='25%' valign='top' rowspan='2'>";
    include 'left_voting.php';
print "</td>";
print "<td valign='top' width='75%'>";
$doesexist="SELECT* from P_question";

if(!get_magic_quotes_gpc())
{
    $_GET = array_map('mysql_real_escape_string', $_GET);
    $_POST = array_map('mysql_real_escape_string', $_POST);
    $_COOKIE = array_map('mysql_real_escape_string', $_COOKIE);
}
else
{
    $_GET = array_map('stripslashes', $_GET);
    $_POST = array_map('stripslashes', $_POST);
    $_COOKIE = array_map('stripslashes', $_COOKIE);
    $_GET = array_map('mysql_real_escape_string', $_GET);
    $_POST = array_map('mysql_real_escape_string', $_POST);
    $_COOKIE = array_map('mysql_real_escape_string', $_COOKIE);
}

$doesexist2=mysql_query($doesexist);
while($doesexist3=mysql_fetch_array($doesexist2))
{
    $quest=$doesexist3[question];
}

```

```

if($quest)
{
    if(isset($_POST['submit']))
    {
        $pollchoice=$_POST['pollchoice'];
        $addchoice="INSERT into P_choices(answer)
VALUES('$pollchoice')";
        mysql_query($addchoice);
        print "<center><h3>Add a choice option</h3></center><br>";
        print "<center>";
        print "<table border='0' width='70%' cellpadding='20'>";
        print "<tr>";
            print "<td valign='top' width='75%'>";
            print "Choice added to poll<br><br>";
            print "<form action='createchoices.php' method='post'>";
            print "Type in poll answer: <input type='text' name='pollchoice'
size='15'>";
            print "<input type='submit' name='submit' value='Add Poll
Option'></form>";
            print "</td></tr>";
            print "<tr valign=top align=center><td>";
            print "<table border='0'><tr><td><b>Poll Answer</b></td></tr>";
            $getans="SELECT * from P_choices";
            $getans2=mysql_query($getans) or die("Could not get answer");
            while($getans3=mysql_fetch_array($getans2))
            {
                print "<tr><td align=center>$getans3[answer]</td> </tr>";
            }
            print "</table>";
            print "</td></tr></table>";
        }
    }

else
{
    print "<center><h3>Add a choice option</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellpadding='20'>";
    print "<tr>";
        print "<td valign='top' width='75%'>";
        print "<form action='createchoices.php' method='post'>";
        print "Type in poll answer: <input type='text' name='pollchoice'
size='15'>";
        print "<input type='submit' name='submit' value='Add Poll
Option'></form>";
        print "</td></tr>";
            print "<tr valign=top align=center><td>";
            print "<table border='0'><tr><td><b>Poll
Answer</b></td></tr>";

```

```

        $getans="SELECT * from P_choices";
        $getans2=mysql_query($getans) or die("Could not
get answer");
        while($getans3=mysql_fetch_array($getans2))
        {
            print "<tr><td
align=center>$getans3[answer]</td></tr>";
        }
        print "</table>";
        print "</td></tr></table>";
    } }

else
{
    print "<center><h3>Add a choice option</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellspacing='20'>";
    print "<tr>";
    print "<td valign='top' width='75%'>";
    print "There is no poll to add choices to, please create a poll first";
    print "</td></tr></table>";
}

    print "</td></tr> </table>";
    print "</center>";

mysql_disconnect;
?>

```

Διόρθωση ερώτησης

```

<?php
include "connect.php";

print "<center><h3>Poll Admin Panel</h3></center><br>";
print "<center>";
print "<table border='0' cellspacing='20'>";
print "<tr><td width='25%' valign='top' rowspan='2'>";
    include 'left_voting.php';
print "</td>";
print "<td valign='top' width='75%'>";
if(isset($_POST['submit']))
{
    $question=$_POST['question'];

```

```

$changequestion="UPDATE P_question set question='$question'";
mysql_query($changequestion) or die("Could not edit question");
print "Question Edited.";
    print "<br><br>Question is:&nbsp;";
        echo "<b>$question</b>";

    }
else
{
    $getpollquest="SELECT * from P_question";
    $getpollquest2=mysql_query($getpollquest) or die("Could not get poll
question");
    $getpollquest3=mysql_fetch_array($getpollquest2);
    print "<form action='editpoll.php' method='post'>";
    print "Poll Question:<br>";
    print "<input type='text' name='question' size='40'
value='$getpollquest3[question]'><br>";
    print "<input type='submit' name='submit' value='edit poll'></form>";

}

```

```

        print "</td></tr> </table>";
    print "</center>";

```

```

mysql_disconnect;
?>

```

Διαγραφή ερώτησης

```

<?php
include "connect.php";

print "<center><h3>Poll Admin Panel</h3></center><br>";
print "<center>";
print "<table border='0' cellspacing='20'>";
print "<tr><td width='25%' valign='top' rowspan='2'>";
    include 'left_voting.php';
print "</td>";
print "<td valign='top' width='75%'>";
    if (isset($_POST['submit']))
    {
        $check="Select * from P_question";
        $check2=mysql_query($check);
        while($check3=mysql_fetch_array($check2))
        {
            $check4=$check3[question];
        }
        if($check4)
        {

```

```

    $deletequestion="DELETE from P_question";
    mysql_query($deletequestion);
    $deleteip="DELETE from P_ip";
    mysql_query($deleteip);
    $deletechoices="DELETE from P_choices";
    mysql_query($deletechoices);
    print "<center><h3>Delete Poll</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellspacing='20'>";
    print "<tr>";
    print "<td valign='top' width='75%'>";
    print "The poll has been deleted, you can now create another poll";
}

else
{
    print "<center><h3>Delete Poll</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellspacing='20'>";
    print "<tr>";
    print "<td valign='top' width='75%'>";
    print "There is no poll to delete";
}

}

else
{
    print "<center><h3>Delete Poll</h3></center><br>";
    print "<center>";
    print "<table border='0' width='70%' cellspacing='20'>";
    print "<tr>";
    print "<td valign='top' width='75%'>";
    print "Warning! Pressing the button below will empty all data to the current
poll<br>";
    print "<form action='deletepoll.php' method='post'>";
    print "<input type='submit' name='submit' value='Delete Poll'>";
    print "</form>";
    print "</td></tr></table>";
    print "</center>";
}

    print "</td></tr> </table>";
    print "</center>";

mysql_disconnect;
?>

```

Δομή παρουσίασης της σελίδας στους Χρήστες

```
<?php
include "connect_poll.php";

$s=$_SERVER["REMOTE_ADDR"];
$ipcheck="SELECT * FROM P_ip where IP='$s'";
$ipcheck2=mysql_query($ipcheck);
while($ipcheck3=mysql_fetch_array($ipcheck2))
{
$ip=$ipcheck3[IP];
}
if($ip)
{

print "Ïüëéð Ý÷âéð øçöβóâé: <br><br>";
$pollquest="SELECT*From P_question";
$sp2=mysql_query($pollquest);
while($sp3 = mysql_fetch_array($sp2))
{
print "<b>$p3[question]</b><br>";
}
//Calculate total number of votes
$totalvotes=0;
$sans="SELECT * from P_choices";
$sans2=mysql_query($sans);
while($sans3=mysql_fetch_array($sans2))
{
$total=$total+$sans3[votes];
}
//now display results
$sans4="SELECT * from P_choices";
$sans5=mysql_query($sans4);
while($sans6=mysql_fetch_array($sans5))
{
if($total>0)
{

$imagewidth=(100*$sans6[votes])/$total;
$imagewidth = number_format($imagewidth, 1, '.', ');
```



```

    print "<img src='pollpic.gif height='5' width='$imagewidth'
border='0'>$ans6[answer]($imagewidth %)<br>";
    }
    else
    {
        print "Äáí ððáñ÷ïðí øçöïïðñßàð";
    }
}
print "<br><b>Óóířéééîß ØÐöié:</b> $total";
}

```

```

else
{
    $pollquest="SELECT*From P_question";
    $p2=mysql_query($pollquest);
    while($p3 = mysql_fetch_array($p2))
    {
        print "<br> $p3[question] <br>";
    }
    $pollans="Select * from P_choices";
    $pollans2=mysql_query($pollans);
    print "<form action='pollit.php' method='post'>";
    while($pollans3=mysql_fetch_array($pollans2))
    {
        print "<input type='radio' name='answer' value='$pollans3[ID]'>
        $pollans3[answer]<br>";
    }
    print "<input type='submit' value='submit' value='vote'>";
    print "</form>";
    print "<A href='results.php' height='300' width='300'
scrollbars='0'>Results</a><br><br>";

}
?>

```

Αποστολή της ip του χρήστη στον πίνακα P_ip

```

<?php
include "connect_poll.php";
$s=$_SERVER["REMOTE_ADDR"];
$ipchecks="SELECT*from P_ip where IP='$s'";
$ipchecks2=mysql_query($ipchecks);
while($ipchecks3=mysql_fetch_array($ipchecks2))
{
    $sip=$ipchecks3[IP];
}

if($sip)

```

```

    {
    print "You have already voted in this poll";
    }

else
    {
    $ID=$_POST['answer'];
    $ipinsert="Insert into P_ip(IP) VALUES('$s')";
    mysql_query($ipinsert);
    $vote = "UPDATE P_choices SET votes=votes+1
            WHERE ID = '$ID'";
    mysql_query($vote);
    }

header( "location: index.php" );

?>

```

Αποτελέσματα με μετρητή των ψήφων και πλήθος των συνολικών ψήφων

```

<?php
include "connect_poll.php";

$totalvotes=0;
$sans="SELECT * from P_choices";
$sans2=mysql_query($sans);
while($sans3=mysql_fetch_array($sans2))
    {
    $total=$total+$sans3[votes];
    }
//now display results
$sans4="SELECT * from P_choices";
$sans5=mysql_query($sans4);
while($sans6=mysql_fetch_array($sans5))
    {
    if($total>0)
        {
        $imagewidth=(100*$sans6[votes])/$total;

        print "<img src='pollpic.gif' height='5' width='$imagewidth'
border='0'>$sans6[answer]($imagewidth %)<br>";
        }
    else
        {
        print "No votes have been cast yet";
        }
    }

```

```
}  
}  
?>
```

8) Ασφάλεια Διαχειριστή

Έλεγχος στοιχείων του διαχειριστή

```
<?  
$userid = $_HTTP_POST_VARS['userid'];  
$password = $_HTTP_POST_VARS['password'];  
  
include "connect.php";  
  
$query = "SELECT * FROM admin_personel WHERE userid='$userid'";  
$res = mysql_query( $query );  
$row = mysql_fetch_array( $res );  
  
if( $password == $row['password'] ) {  
    //print "login correct<br>";  
    setcookie( 'userid', $userid );  
    setcookie( 'password', $password );  
    header( "location: main.php" );  
}  
?>
```

Logout.php

```
<?  
    setcookie( 'userid', " );  
    setcookie( 'password', " );  
    header( 'location: login.php' );  
?>
```

Έλεγχος οποιαδήποτε σελίδας εαν έχει κάνει login ο διαχειριστής

```
<?  
$userid = $_HTTP_COOKIE_VARS['userid'];  
$password = $_HTTP_COOKIE_VARS['password'];  
//print "userd='$userid', password='$password'";  
  
include "connect.php";  
$query = "SELECT * FROM admin_personel WHERE userid='$userid'";  
$res = mysql_query( $query );  
$row = mysql_fetch_array( $res );  
$firstname = $row['firstname'];
```

```
#  
# make sure that this user has logged in:  
#  
$userid != " || die( 'ERROR: no userid was specified' );  
$password == $row['password'] || die( 'ERROR: you are not logged in!' );  
>
```

Βιβλιογραφία

- [1] Ανάπτυξη Web εφαρμογών με PHP & MySQL
Luke Welling και Laura Thomson. Εκδότης :Μ. Γκιούρδας

- [2] PHP, Οδηγός Προγραμματισμού, Sterling Hughes

- [3] <http://www.freewebmasterhelp.com/tutorials/phpmysql/1> 1/09/2006

- [4] <http://eos.uom.gr/~kaklaman/book/Chapters/C6/How%20Web%20servers%20work%203.htm> 25/07/2006

- [5] <http://eos.uom.gr/~kaklaman/book/Chapters/C7/C7Header.htm>
25/07/2006

- [6] <http://news.netcraft.com/archives/2006/09/index.html> 25/07/2006

- [7] <http://www.php.net/> 20/09/2006

- [8] Εγχειρίδιο της HTML, Laura Lemay, Εκδότης Μ. Γκιούρδας

- [9] Οδηγός της XML με παραδείγματα , Β. Γκιούρδας Εκδοτική