

ΑΝΩΤΑΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής και Πολυμέσων



Πτυχιακή Εργασία

ΔΗΜΙΟΥΡΓΙΑ ΔΥΝΑΜΙΚΗΣ ΕΦΑΡΜΟΓΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΕΚΔΡΟΜΩΝ



Επιβλέπων Καθηγητής : Κλεισαρχάκης Μιχαήλ

Σπουδάστρια: Λαντζουράκη Χρυσούλα

Α.Μ:1957

Ηράκλειο Ιούνιος 2010

Περιεχόμενα

1	Ευχαριστίες	5
2	Web Εφαρμογή.....	6
2.1	Περιγραφή θέματος web εφαρμογής και συνοπτική περιγραφή των χρηστών	6
2.2	Ευελιξία που προσφέρει μια δυναμική web Εφαρμογή.....	6
2.3	Οφέλη Web Εφαρμογής προς τους χρήστες	7
2.4	Οφέλη web εφαρμογής προς τις εταιρίες που την παρέχουν	8
3	Τεχνικές Ανάπτυξης.....	9
3.1	Apache web server	9
3.2	PHP Script Language	11
3.3	MySQL Database	13
3.4	JavaScript	14
3.5	JQUERY.....	15
3.6	HTML	16
3.7	Cascading Style Sheets (CSS).....	17
4	Εργαλεία Ανάπτυξης	18
4.1	WAMP.....	18
4.2	Adobe Photoshop CS3	19
4.3	PhpMyAdmin.....	20
5	Κατασκευή Διαδικτυακών εφαρμογών: Η χρήση του PHP- Apache- MySQL	20
5.1	Εγκατάσταση APACHE - PHP – MySQL: Χρήση WAMP	20
6	Δημιουργία Δυναμικού Ιστοχώρου: Συστήματος αναζήτησης διακοπών.....	24
6.1	Εισαγωγή.....	24
6.2	Βάση δεδομένων	25
6.3	Συνοπτική περιγραφή εννοιών μίας βάσης δεδομένων	25
	Ορισμός βάσης δεδομένων	25
	Στόχοι μίας βάσης δεδομένων.....	26
6.4	Δημιουργία βάσης δεδομένων και χρήστη	27
6.5	Αρχιτεκτονική της βάσης δεδομένων	28
7	Δομή και Οργάνωση της Πληροφορίας.....	31
8	Δομή και Οργάνωση ιστοχώρου	33
8.1	Διαγράμματα περιήγησης χρηστών	38
9	Περιγραφή της λειτουργίας των χρηστών αναλυτικά.....	39
10	Περιγραφή και επεξήγηση αναλυτικών κομματιών κώδικα	46
10.1	Δημιουργία των πινάκων της βάσης	46
10.2	Σύνδεση στη βάση δεδομένων	50
10.3	Εισαγωγή ενδεικτικών εγγραφών σε πίνακα	53
10.4	Εμφάνιση περιεχομένων εγγραφών	53
10.5	Εισαγωγή εγγραφής	55
	Στο αρχείο actions.php γίνονται οι διεργασίες, όπως η προσθήκη κάποιας εγγραφής, η οποία πραγματοποιείται όταν δεν δίνεται κάποιο id	57
10.6	Διαγραφή κάποιας συγκεκριμένης εγγραφής.....	58
10.7	Περιγραφή της εφαρμογής για το διαχειριστή.....	61
10.8	Περιγραφή της εφαρμογής για τον χρήστη.....	64
11	Συμπεράσματα	74
12	Βιβλιογραφία- Πηγές	75

Εικόνες

Εικόνα 1 Συνδυασμός Τεχνολογιών.....	10
Εικόνα 2 Δέντρο Τεχνολογιών	11
Εικόνα 3 Server-side Processing	12
Εικόνα 4 Σύνδεση ODBC με MySql.....	14
Εικόνα 5 Javascript events	15
Εικόνα 6 Παράδειγμα κώδικα jquery.....	16
Εικόνα 7 HTML δέντρο	17
Εικόνα 8 Παράδειγμα σύνταξης css.....	18
Εικόνα 9 Αρχική οθόνη εγκατάστασης WAMPSEVER.....	21
Εικόνα 10 Καθορισμός υποκαταλόγου εγκατάστασης	21
Εικόνα 11 Αποδέσμευση λειτουργιών Apache Web Server.....	22
Εικόνα 12 Μήνυμα επιτυχούς εγκατάστασης	22
Εικόνα 13 Πρόσβαση χρήστη στα δεδομένα	24
Εικόνα 14 Προβολή χρήστη και προνομιών	28
Εικόνα 15 Πίνακες βάσης δεδομένων travel_agency	28
Εικόνα 16 Δομή του πίνακα users.....	29
Εικόνα 17 Δομή του πίνακα continent	29
Εικόνα 18 Δομή του πίνακα country	29
Εικόνα 19 Δομή του πίνακα category	30
Εικόνα 20 Δομή του πίνακα offers.....	30
Εικόνα 21 Συσχετίσεις μεταξύ των πινάκων της βάσης	31
Εικόνα 22 Διάταξη συνδέσμων πλοήγησης σε σχήμα "Γ"	32
Εικόνα 23 Δομή και οργάνωση αρχείου users	34
Εικόνα 24 Δομή και οργάνωση αρχείου country	35
Εικόνα 25 Δομή και οργάνωση αρχείου continent.....	35
Εικόνα 26 Δομή και οργάνωση αρχείου category.....	36
Εικόνα 27 Δομή και οργάνωση αρχείου offers	36
Εικόνα 28 Δομή και οργάνωση αρχείου log-in.....	37
Εικόνα 29 Δικαιώματα admin και τουριστικών γραφείων στην εφαρμογή.....	37
Εικόνα 30 Δυνατότητες τουριστικού γραφείου στην εφαρμογή.....	38
Εικόνα 31 Δυνατότητες διαχειριστή στην εφαρμογή.....	38
Εικόνα 32 Δυνατότητες απλού χρήστη στην εφαρμογή	39
Εικόνα 33 Είδη χρηστών.....	40
Εικόνα 34 Γρήγορη αναζήτηση χρήστη.....	40
Εικόνα 35 Αρχική σελίδα εφαρμογής	41
Εικόνα 36 Σελίδα εφαρμογής με λεπτομέρειες προσφοράς.....	42
Εικόνα 37 Φόρμα επικοινωνίας με διαχειριστή	42
Εικόνα 38 Φόρμα επικοινωνίας με τουριστό γραφείο	43
Εικόνα 39 Log_in διαχειριστή	43
Εικόνα 40 Διαχειριστικό περιβάλλον.....	44
Εικόνα 41 Log_in τουριστικού γραφείου	45
Εικόνα 42 Διαχειριστικό περιβάλλον εγγεγραμμένων τουριστικών γραφείων	45
Εικόνα 43 Εγγραφές του πίνακα users.....	53
Εικόνα 44 Εγγεγραμμένα τουριστικά γραφεία.....	54
Εικόνα 45 Δήλωση των υποχρεωτικών πεδίων της φόρμας στο αρχείο init.js.....	74

Πίνακες επιλεγμένου κώδικα

Κώδικας 1 Δομή του πίνακα category.....	46
Κώδικας 2 Δομή του πίνακα continent.....	47
Κώδικας 3 Δομή του πίνακα country	47
Κώδικας 4 Δομή του πίνακα offers	48
Κώδικας 5 Δομή του πίνακα users	49
Κώδικας 6 Σύνδεση στην βάση δεδομένων (php)	51
Κώδικας 7 Επιλογή εγγεγραμμένων από τη βάση	52
Κώδικας 8 Εισαγωγή εγγεγραμμένων στο πίνακα users	53
Κώδικας 9 Εμφάνιση εγγεγραμμένων τουριστικών γραφείων με τη χρήση της sql.....	54
Κώδικας 10 Φόρμα επεξεργασίας τουριστικού γραφείου στο αρχείο add_editUsers	56
Κώδικας 11 Φόρμα προσθήκης τουριστικού γραφείου στο αρχείο add_editUsers	57
Κώδικας 12 Προσθήκη τουριστικού γραφείου στο αρχείο actions	58
Κώδικας 13 Διαγραφή επιλεγμένου τουριστικού γραφείου στο αρχείο actions	59
Κώδικας 14 Επεξεργασία επιλεγμένου τουριστικού γραφείου στο αρχείο actions	60
Κώδικας 15 Έλεγχος ταυτότητας διαχειριστή με έλεγχο συνόδων λειτουργίας στο αρχείο admin ..	61
Κώδικας 16 Φόρμα εισαγωγής στοιχείων του διαχειριστή, για την εισοδο του στο διαχειριστικό περιβάλλον	62
Κώδικας 17 Έλεγχος ταυτότητας διαχειριστή στο αρχείο actions.....	63
Κώδικας 18 Κλήση της συνάρτησης content_site() στο index αρχείο	64
Κώδικας 19 δήλωση της συνάρτησης content_site() στο αρχείο functions.....	64
Κώδικας 20 Έλεγχο ταυτότητας χρήστη με έλεγχο συνόδων λειτουργίας στο αρχείο index	66
Κώδικας 21 Φόρμα εισαγωγής στοιχείων των τουριστικών γραφείων, για την εισοδο τους στο διαχειριστικό περιβάλλον.....	66
Κώδικας 22 Έλεγχος ταυτότητας τουριστικού γραφείου στο αρχείο actions.....	68
Κώδικας 23 Χρήση φίλτρων για γρήγορη αναζήτηση χρήστη στο αρχείο offers.....	71

1 Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή κ.Μιχαήλ Κλεισαρχάκη που μου ανέθεσε το συγκεκριμένο θέμα στο πλαίσιο της διπλωματικής μου εργασίας.

Επίσης ευχαριστώ την οικογένειά μου και τα άτομα του άμεσου περιβάλλοντός μου που με ενθάρρυναν και μου συμπαραστάθηκαν κατά την διάρκεια των σπουδών μου.

2 Web Εφαρμογή

2.1 Περιγραφή θέματος web εφαρμογής και συνοπτική περιγραφή των χρηστών

Η παρούσα πτυχιακή αποσκοπεί στην ανάλυση και υλοποίηση διαδικτυακής εφαρμογής ενός συστήματος αναζήτησης διακοπών με δυνατότητα προσθήκης προσφορών από Τουριστικά Γραφεία. Στην ιστοσελίδα γίνεται παρουσίαση προσφορών με απλή και σύνθετη αναζήτηση όπως και παρουσίαση των τουριστικών γραφείων. Η παρούσα εφαρμογή έχει τριών ειδών κατηγορίες χρηστών:

A. Τα Τουριστικά Γραφεία , τα οποία χρησιμοποιούν την εφαρμογή με σκοπό να παρέχουν στους χρήστες την ευελιξία των προσφορών τους. Τα Τουριστικά αυτά γραφεία έχουν την δυνατότητα να ανανεώνουν περιστασιακά με τις προσφορές τους την εφαρμογή αφήνοντας έτσι τους χρήστες που επισκέπτονται το site , αναζητώντας ένα ιδανικό προορισμό για διακοπές, ενήμερους ακόμα και στις πιο πρόσφατες προσφορές. Σε ένα διαχειριστικό περιβάλλον τα Τουριστικά γραφεία έχουν δικαίωμα στην προσθαφαίρεση και επεξεργασία μόνο των προσφορών τους, διασφαλίζοντας έτσι την ακεραιότητα των προσφορών του κάθε Τουριστικού γραφείου.

B. Το διαχειριστή , ο οποίος χρησιμοποιεί την εφαρμογή. Σε ένα διαχειριστικό περιβάλλον ο διαχειριστής έχει κάθε δικαίωμα επεξεργασίας του περιεχομένου.

Γ. Τους χρήστες που επισκέπτονται την εφαρμογή, αναζητώντας τον ιδανικό προορισμό για τις διακοπές τους. Η εφαρμογή παρέχει την δυνατότητα στους χρήστες να αναζητούν την διαθεσιμότητα των προσφορών με κριτήρια, όπως την χρονική περίοδο που επιθυμούν να πραγματοποιήσουν τις διακοπές τους, την περιοχή ενδιαφέροντος καθώς και την κατηγορία του ταξιδιού.

Οποιοσδήποτε χρήστης επισκεφτεί την ιστοσελίδα, αναζητώντας προσφορές για τις διακοπές του, του δίνεται η δυνατότητα να επιλέξει την προσφορά που θέλει και μπορεί να συμπληρώσει μία φόρμα με τα στοιχεία του όπου ενημερώνετε η βάση δεδομένων (για το ιστορικό) και το Τουριστικό Γραφείο της προσφοράς με ένα e-mail.

2.2 Ευελιξία που προσφέρει μια δυναμική web Εφαρμογή

Ο δυναμικός μηχανισμός δημιουργεί τις σελίδες διαβάζοντας πληροφορίες που προέρχονται από μια βάση δεδομένων, εισάγοντας τις πληροφορίες (data) σε ένα "template" και στέλνοντας τη σελίδα πίσω στον web browser.

Το template θα μπορούσαμε να το αποδώσουμε με τον όρο "πρότυπο εμφάνισης". Είναι ο τρόπος εμφάνισης μιας σελίδας, το "καλούπι" και το συγκεκριμένο της στήσιμο ανεξαρτήτως περιεχομένου. Για παράδειγμα, θα μπορούσαμε να ορίσουμε από την αρχή ότι μια σελίδα παρουσίασης κάποιας προσφοράς θα αποτελείται από τον τίτλο, μια σύντομη περιγραφή της προσφοράς, μία μικρή φωτογραφία της, την κατηγορία στην οποία ανήκει η προσφορά, τη χρονική περίοδο, την τιμή της, καθώς και το τουριστικό γραφείο μέσω του οποίου παρέχεται η προσφορά. Μπορούμε να ακολουθήσουμε το ίδιο στήσιμο σε όλες τις προσφορές .

Αντί να σχεδιάσουμε λοιπόν μια ξεχωριστή σελίδα για καθεμία από τις προσφορές που διαθέτει ένα τουριστικό γραφείο, φτιάχνουμε μία και μόνο, η οποία αποτελεί το βασικό σκελετό παρουσίασης των προσφορών. Ο μηχανισμός που λειτουργεί από πίσω (back end) φροντίζει μέσα στο σκελετό αυτό να εμφανίσει τη προσφορά που θα ζητήσει εκείνη τη στιγμή ο χρήστης (front end).

Ένας δυναμικός δικτυακός τόπος επιτρέπει τα ακόλουθα:

- Να αλλάζετε με ευκολία κάποια πληροφορία σε όλες τις σελίδες του site ταυτόχρονα, κάνοντας μόνο μία αλλαγή στη βάση δεδομένων ή στο template, η οποία θα ενημερώνει όλες τις σελίδες.
- Να αλλάζετε εύκολα τις πληροφορίες σχετικά με τις προσφορές. Για παράδειγμα, επιθυμείτε να αλλάξετε την τιμή κάποιας συγκεκριμένης προσφοράς
- Να επιλέγει ο επισκέπτης τον τρόπο με τον οποίο θα βλέπει τις προσφορές, καθώς έχει τη δυνατότητα να εξατομικεύει τις σελίδες χρησιμοποιώντας διάφορα κριτήρια δικής του επιλογής, που υπάρχουν ήδη στη βάση δεδομένων, π.χ. χώρα, κατηγορία, χρονική περίοδο.

2.3 Οφέλη Web Εφαρμογής προς τους χρήστες

Τα τελευταία χρόνια υπάρχει μία ραγδαία αύξηση του αριθμού των χρηστών στο διαδίκτυο την οποία βοήθησε η παροχή γρήγορων και οικονομικών υπηρεσιών πρόσβασης. Αυτό είχε σαν συνέπεια να αυξηθούν και οι εταιρίες παροχής υπηρεσιών και πώλησης προϊόντων μέσω διαδικτύου.

- Τα ηλεκτρονικά καταστήματα είναι ανοιχτά 24 ώρες το 24ωρο. Έτσι ο χρήστης μπορεί ανά πάσα στιγμή να αναζητήσει τα προϊόντα και τις υπηρεσίες που τον ενδιαφέρουν, ρίχνοντας

μια ματιά στο διαδίκτυο, χωρίς να χρειάζεται να επισκεφτεί το εκάστοτε κατάστημα που τον ενδιαφέρει.

- Το κόστος των προϊόντων που πωλούνται και των υπηρεσιών που παρέχονται μέσω Internet είναι κατά γενικό κανόνα πολύ χαμηλότερο από τις τιμές του εμπορίου, αφού ένα ηλεκτρονικό κατάστημα είναι απαλλαγμένο από μεγάλο μέρος του λειτουργικού κόστους ενός πραγματικού καταστήματος (ενοικίαση χώρου και «αέρα», ηλεκτρικό, νερό κλπ) και γενικά απαιτεί πολύ λιγότερο υπαλληλικό προσωπικό.
- Η αγορά είναι πραγματικά παγκόσμια. Με άλλα λόγια, μπορείτε μέσω του υπολογιστή σας να αγοράσετε ακόμα και κάτι το οποίο δεν κυκλοφορεί στην Ελλάδα, χωρίς να πρέπει πια να περιμένετε τότε κάποιος φίλος σας θα ταξιδέψει στο εξωτερικό για να σας το φέρει.
- Η συναλλαγή είναι γρήγορη και άμεση. Με άλλα λόγια, από τη στιγμή που ολοκληρώνετε την παραγγελία σας, το αργότερο σε 3-4 ημέρες την έχετε λάβει, ακόμα και αν εκείνη τη στιγμή το προϊόν βρισκόταν στην άλλη άκρη του πλανήτη.
- καθένας βρίσκει αυτό που θέλει, όποτε το θέλει, χωρίς να κάνει βήμα, χωρίς δηλαδή κόπο και χωρίς καμία σπατάλη χρόνου. Με άλλα λόγια απλά και εύκολα ψώνια από το σπίτι ή το γραφείο!

2.4 Οφέλη web εφαρμογής προς τις εταιρίες που την παρέχουν

- Όπως προαναφέραμε, κάθε εταιρία που έχει ηλεκτρονική παρουσία μπορεί να διευρύνει τον κύκλο εργασιών της επεκτείνοντας τα γεωγραφικά όρια των συναλλαγών της. Αυτό σημαίνει πως κάθε επιχείρηση που διαθέτει τα προϊόντα της online μπορεί και αποκτά πελάτες σε περιοχές που βρίσκονται μακριά από την έδρα της, ακόμα και στο εξωτερικό. Με άλλα λόγια, κάθε επιχείρηση που έχει ένα ηλεκτρονικό κατάστημα, είναι σαν να έχει υποκαταστήματα σε πολλές περιοχές και μάλιστα με ελάχιστο λειτουργικό κόστος.
- Κάθε εταιρία που χρησιμοποιεί τις νέες τεχνολογίες- όπως το Internet- γίνεται εξ ορισμού πιο ανταγωνιστική, αφού μπορεί να ενημερώνεται πιο εύκολα για τις τρέχουσες εξελίξεις στο χώρο της. Με άλλα λόγια και με δεδομένο το ότι σε λίγα χρόνια όλες οι εμπορικές δραστηριότητες θα γίνονται μέσω Internet, το ηλεκτρονικό εμπόριο είναι η νέα μεγάλη πρόκληση για κάθε εταιρία που θέλει να είναι ανταγωνιστική.
- Οι ηλεκτρονικές συναλλαγές επιτρέπουν την αμφίδρομη σχέση μεταξύ επιχείρησης και καταναλωτή (interaction). Αυτό σημαίνει πως κάθε εταιρία μέσω των ηλεκτρονικών

συναλλαγών μπορεί να συλλέξει πολλά στοιχεία για τις συνήθειες, τις ανάγκες και τα γούστα των καταναλωτών και σύμφωνα με αυτά να αναπροσαρμόσει την πολιτική της προς το θετικότερο.

- Τέλος, γνωρίζοντας τις συγκεκριμένες ανάγκες των πελατών τους, οι εταιρίες μπορούν να προχωρήσουν στη δημιουργία συγκεκριμένων προϊόντων είτε ανταποκρινόμενων σε έναν καταναλωτή, είτε σε μια ομάδα καταναλωτών που χρειάζονται ένα νέο προϊόν το οποίο δεν υπάρχει ακόμα στην αγορά.

3 Τεχνικές Ανάπτυξης

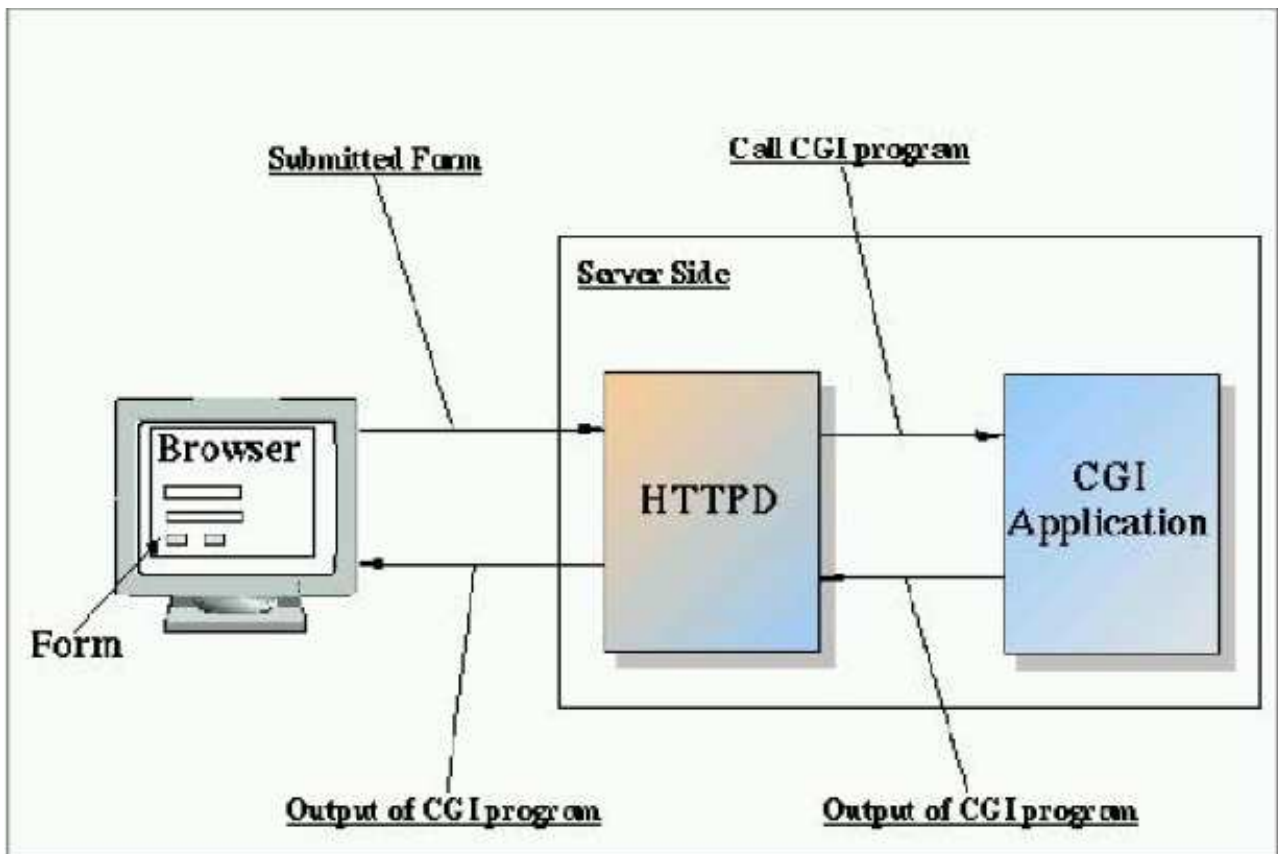
3.1 Apache web server

Apache: Είναι ο πιο διαδεδομένος διακομιστής που χρησιμοποιείται στο web και η λειτουργία του είναι να στέλνει html σελίδες (κυρίως) σε όποιον browser της ζητήσει.

Apache, αλλιώς γνωστό ως Apache HTTP Server, είναι ένα καθιερωμένο πρότυπο για την online διανομή της ιστοσελίδας του υπηρεσιών, ο οποίος έδωσε την αρχική ώθηση για την επέκταση του World Wide Web. Είναι ένα open-source πλατφόρμα web server, η οποία εγγυάται την ηλεκτρονική διάθεση της πλειοψηφίας των δικτυακών τόπων που δραστηριοποιούνται σήμερα. Το Web μπορεί να επεκταθεί και πέρα από τη στατική μεταφορά σελίδων, στην εκτέλεση προγραμμάτων είτε στη «μεριά» του πελάτη (client-side processing), ή στη «μεριά» του εξυπηρετητή(server-side processing). Ανάλογα με το που εκτελούνται τα προγράμματα χρησιμοποιούνται διάφορες τεχνολογίες (CGI, PHP, ASP, JSP, Servlets, JDBC στον εξυπηρετητή και scripting γλώσσες στον πελάτη).

Server-side Processing

Το Common Gateway Interface (CGI) είναι μια τυποποίηση για την αλληλεπίδραση εξωτερικών εφαρμογών με εξυπηρετητές πληροφοριών όπως web servers. Ένα πρόγραμμα CGI, είναι εκτελέσιμο κάθε φορά που αυτό ζητείται από κάποιον πελάτη, έτσι ώστε να μπορεί να δημιουργεί δυναμική πληροφορία. Το CGI είναι ένας τυποποιημένος τρόπος να περνάμε δεδομένα φορμών HTML από ένα browser σε ένα server. Όταν η πληροφορία συλλέγεται από τον browser, στέλνεται σ'ένα HTTP server (Hyper Text Transfer Protocol) ο οποίος καθορίζεται από την HTML φόρμα. Αυτός ο server ξεκινάει ένα πρόγραμμα, το οποίο περιγράφεται επίσης στην HTML φόρμα και επεξεργάζεται την πληροφορία. Τέτοια προγράμματα είναι γνωστά σαν Common Gateway Interface προγράμματα ή CGI scripts.



Εικόνα 1 Συνδυασμός Τεχνολογιών

Client-side Processing

Οι τεχνολογίες στη μεριά του πελάτη αξιοποιούν σε σημαντικό βαθμό τις επενδύσεις σε προσωπικούς υπολογιστές, μεταφέροντας επεξεργασία (και συνεπώς φόρτο) από τον εξυπηρετητή προς τον πελάτη. Οι τεχνολογίες που λειτουργούν στη μεριά του πελάτη μπορούν να βελτιώσουν το user interface με το να αποκρίνονται άμεσα στις ενέργειες των χρηστών. Στο σχεδιασμό εφαρμογών web, δε χρειάζεται να επιλεγεί είτε η μια τεχνολογία ή η άλλη για κάποια συγκεκριμένη εφαρμογή. Ένας τυπικός συνδυασμός από τεχνολογίες περιλαμβάνει τη χρήση HTML, Javascript, CGI με Perl και συνδεσιμότητα με βάσεις δεδομένων

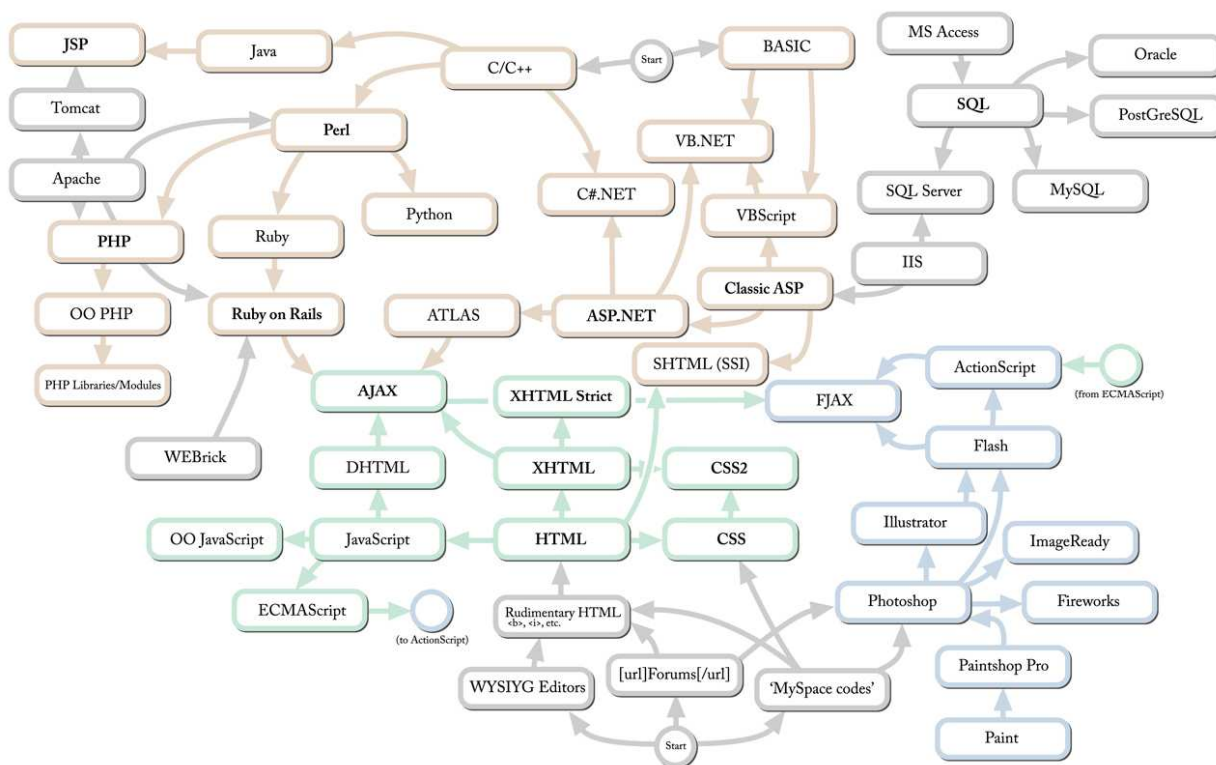
Για παράδειγμα:

- ✓ Φόρμες HTML χρησιμοποιούνται για τη συλλογή δεδομένων από τον χρήστη

Επίσης:

- ✓ Η JavaScript χρησιμοποιείται στη μεριά του πελάτη για να επικυρώσει τις εισόδους που δίνει ο χρήστης στα πεδία της φόρμας, πριν αυτά περαστούν στον εξυπηρετητή
- ✓ Ο προγραμματισμός CGI με χρήση Perl, χρησιμοποιείται στον εξυπηρετητή για την ανάλυση και επεξεργασία των δεδομένων

- ✓ Μια σύνδεση με βάση δεδομένων στον εξυπηρετητή χρησιμοποιείται για την προσπέλαση της ζητούμενης πληροφορίας ή την αποθήκευση επιπλέον πληροφορίας
- ✓ Επιπλέον προγράμματα CGI στον εξυπηρετητή χρησιμοποιούνται για την κατάλληλη μορφοποίηση της απάντησης στο χρήστη



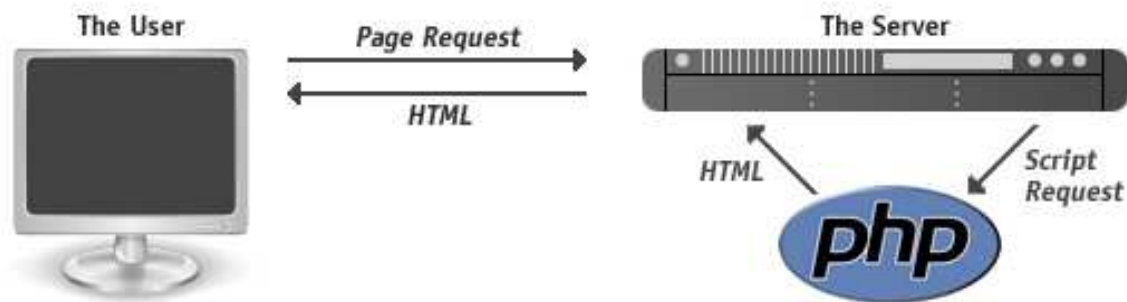
Εικόνα 2 Δέντρο Τεχνολογιών

3.2 PHP Script Language

Η PHP είναι μία γλώσσα που χρησιμοποιείται κατά κόρον για την κατασκευή «δυναμικών ιστοσελίδων». Είναι Scripting γλώσσα που σημαίνει ότι μπορεί να τη χρησιμοποιήσει κάποιος για να δημιουργήσει κάποια κομμάτια PHP κώδικα σε μία HTML σελίδα και θα εκτελούνται κάθε φορά που θα επισκέπτεστε τη σελίδα πραγματοποιώντας κάποιες συγκεκριμένες λειτουργίες. Ο PHP κώδικας μεταφράζεται στο Web διακομιστή και δημιουργεί HTML στην άλλη έξοδο που θα δει ο επισκέπτης. Η PHP λοιπόν δεν στέλνεται άμεσα σε έναν πελάτη (client), αντ' αυτού πρώτα αναλύεται και μετά αποστέλλεται το παραγόμενο αποτέλεσμα. Τα συνοδευτικά στοιχεία HTML στον πηγαίο κώδικα μένουν ως έχουν, όμως ο PHP κώδικας ερμηνεύεται και εκτελείται. Ο κώδικας PHP μπορεί να θέσει ερωτήματα σε βάσεις δεδομένων, να δημιουργήσει εικόνες, να διαβάσει και να γράψει αρχεία, να συνδεθεί με απομακρυσμένους υπολογιστές, και άλλες δυνατότητες. Η PHP είναι επίσης τεχνολογίας server side. Αυτό σημαίνει ότι ένα οποιοδήποτε php script που υπάρχει σε

μία php ιστοσελίδα εκτελείται στο server. Έστω ότι ένας χρήστης πληκτρολογεί `http://www...` και ζητάει κάποια ιστοσελίδα. Εάν αυτή η σελίδα περιέχει κώδικα σε php τότε, όταν βρεθεί στον server η αντίστοιχη ιστοσελίδα ΠΡΩΤΑ ΕΚΤΕΛΕΙΤΑΙ εκεί ο κώδικας και μετά στέλνεται στο browser του χρήστη.

Άλλα χαρακτηριστικά της php είναι ότι είναι ΔΩΡΕΑΝ, διότι είναι open-source ,είναι σχετικά απλή στη μάθηση και είναι προσφέρει σταθερότητα στις διάφορες εφαρμογές που δημιουργούμε με αυτή.



Εικόνα 3 Server-side Processing

Προκειμένου να επαληθεύσουμε την ορθότητα της λειτουργίας του Apache Web server και της PHP, δημιουργούμε το παρακάτω αρχείο `test.php`, το οποίο περιέχει την εσωτερική συνάρτηση `phpinfo()` της PHP:

test.php

```
<html>
<? php phpinfo()?>
</html>
```

Τα σύμβολα `<?` και `?>` δείχνουν την αρχή και το τέλος του τμήματος που αφορά την PHP. Μια σελίδα γραμμένη σε PHP για να εκτελεστεί από το διακομιστή θα πρέπει να έχει κατάληξη `.php` (αντί της συνηθισμένης `.html` ή `htm` που αφορά στατικές ιστοσελίδες) και μπορούμε να προσπελάσουμε σε αυτή αν δώσουμε στο φυλλομετρητή το

url: `http://localhost/[όνομα_αρχείου].php`

Κάποιοι από τους βασικούς ανταγωνιστές της PHP είναι ο Perl, Microsoft Active

Server Pages (ASP) , Java Server Pages (JSP) και Allaire Cold Fusion .

Σε σύγκριση με αυτά τα προϊόντα, η PHP έχει πολλά πλεονεκτήματα όπως :

- Υψηλή απόδοση
- Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων
- Ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες Web διαδικασίες
- Χαμηλό κόστος
- Ευκολία μάθησης και χρήσης
- Μεταφερσιμότητα
- Διαθεσιμότητα του κώδικα προέλευσης

3.3 MySQL Database

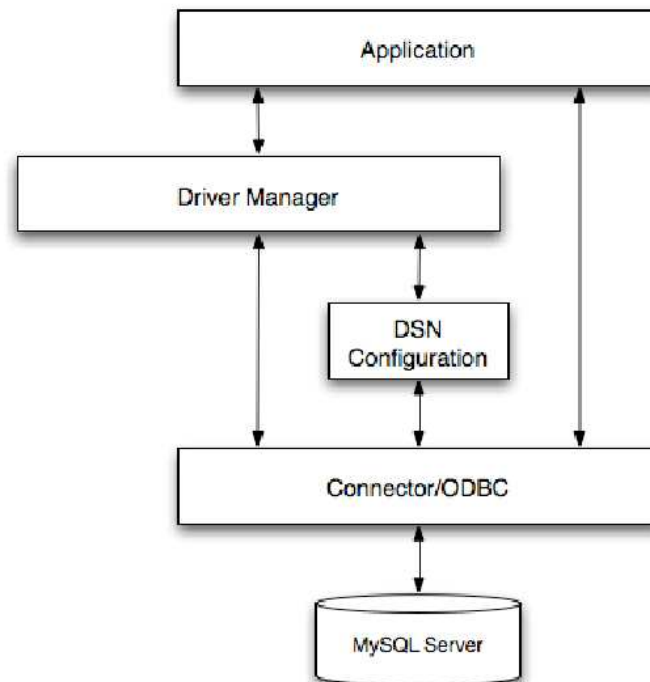
Η MySQL είναι ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων. Μια βάση δεδομένων επιτρέπει την αποθήκευση , την αναζήτηση , την ταξινόμηση και την ανάκληση των δεδομένων αποτελεσματικά . Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα , για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα , για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση. Συνεπώς η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών.

Έχοντας επαληθεύσει ότι όλα τα συστατικά της εγκατάστασης λειτουργούν ορθά, είμαστε σε θέση εκτός από PHP scripts και απλές ιστοσελίδες να δημιουργήσουμε και βάσεις δεδομένων με τη χρήση της MySQL. Για την επαλήθευση λειτουργίας της, ανοίγουμε ένα Dos command prompt και πληκτρολογώντας την αμέσως παρακάτω εντολή και δίνοντας ως συνθηματικό το Enter (δηλαδή χωρίς συνθηματικό), θα πρέπει να λάβουμε τα παρακάτω αποτελέσματα:

```
C:\>mysql -h localhost -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 5 to server version: 5.0.1-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

Μερικοί από τους κύριους ανταγωνιστές της MySQL είναι οι PostgreSQL , Microsoft SQL και Oracle.

Η MySQL έχει πολλά πλεονεκτήματα , όπως χαμηλό κόστος , εύκολη διαμόρφωση και μάθηση και ο κώδικας προέλευσης είναι διαθέσιμος.

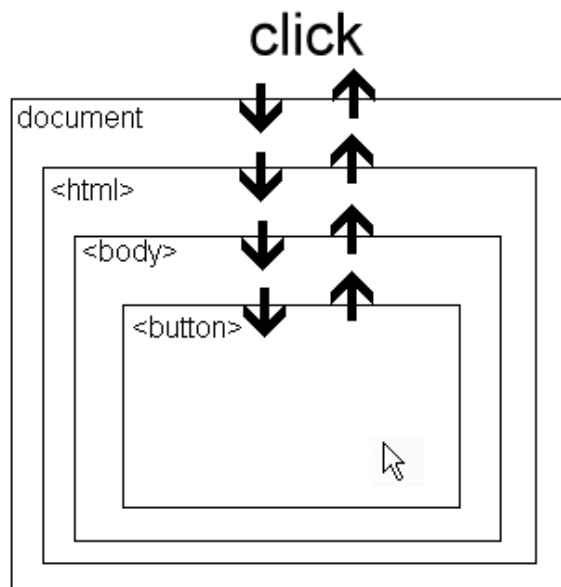


Εικόνα 4 Σύνδεση ODBC με MySql

3.4 JavaScript

Όπως και η PHP, η Javascript έχει βασιστεί όσον αφορά τον τρόπο σύνταξης του κώδικά της στη γλώσσα προγραμματισμού C, με την οποία παρουσιάζει πολλές ομοιότητες. Όμως ενώ η PHP είναι μια server side γλώσσα προγραμματισμού, η Javascript είναι client side.

Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στον server, αλλά στο πρόγραμμα περιήγησης των επισκεπτών. Αυτή η διαφορά έχει και πλεονεκτήματα και μειονεκτήματα για καθεμιά από τις δύο γλώσσες. Συγκεκριμένα, η Javascript δεν έχει καμία απαίτηση από πλευράς δυνατοτήτων του server για να εκτελεστεί (επεξεργαστική ισχύ, συμβατό λογισμικό διακομιστή), αλλά βασίζεται στις δυνατότητες του browser των επισκεπτών. Επίσης μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Παρόλα αυτά, οι δυνατότητές της είναι σημαντικά μικρότερες από αυτές της PHP και δεν παρέχει συνδεσιμότητα με βάσεις δεδομένων.



Εικόνα 5 Javascript events

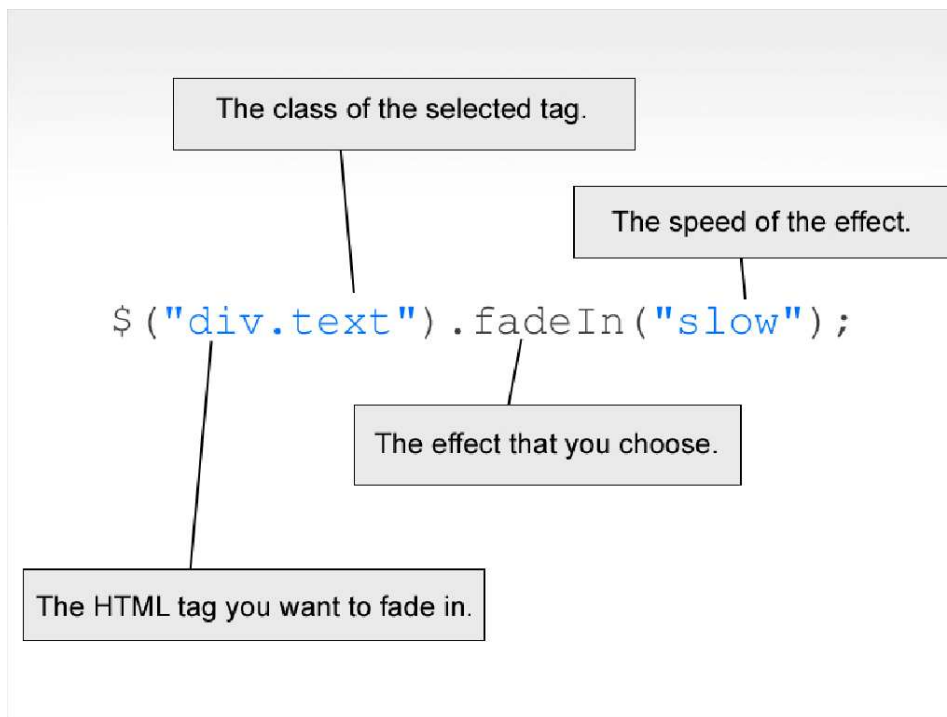
Πολύ σύντομα η Microsoft αναγνώρισε τις δυνατότητες που έδινε η JavaScript στον αντίπαλο browser και αποφάσισε να εφοδιάσει το δικό της με μία παρόμοια γλώσσα, τη JScript. Οι δύο γλώσσες πραγματικά έχουν ελάχιστες διαφορές, που στην πράξη μπορούν να αγνοηθούν.

3.5 JQUERY

Το jQuery είναι μια ελαφριά βιβλιοθήκη για την JavaScript, συμβατή με όλους τους φυλλομετρητές της αγοράς, που δίνει έμφαση στην αλληλεπίδραση της HTML με την JavaScript.

Μερικά από τα πλεονεκτήματα και χαρακτηριστικά της χρήσης της βιβλιοθήκης jQuery:

- Μειώνει δραματικά τον αριθμό των γραμμών κώδικα που πρέπει να συνταχθεί σε σύγκριση με την καθαρή Javascript.
- Είναι πολύ πιο εύκολο να την καταλάβουμε σε σχέση με τον καθαρό κώδικα Javascript.
- Το documentation είναι εξαιρετικά καλά οργανωμένο.
- Θα κάνει τη χρήση της τεχνολογίας Ajax εξαιρετικά εύκολη.
- Έχει αναπτυχθεί ένα ευρύ φάσμα από plugins και extensions για την jQuery με αποτέλεσμα να είναι εύκολο για σας να πάρετε την ακριβή λειτουργία που σας ενδιαφέρει.

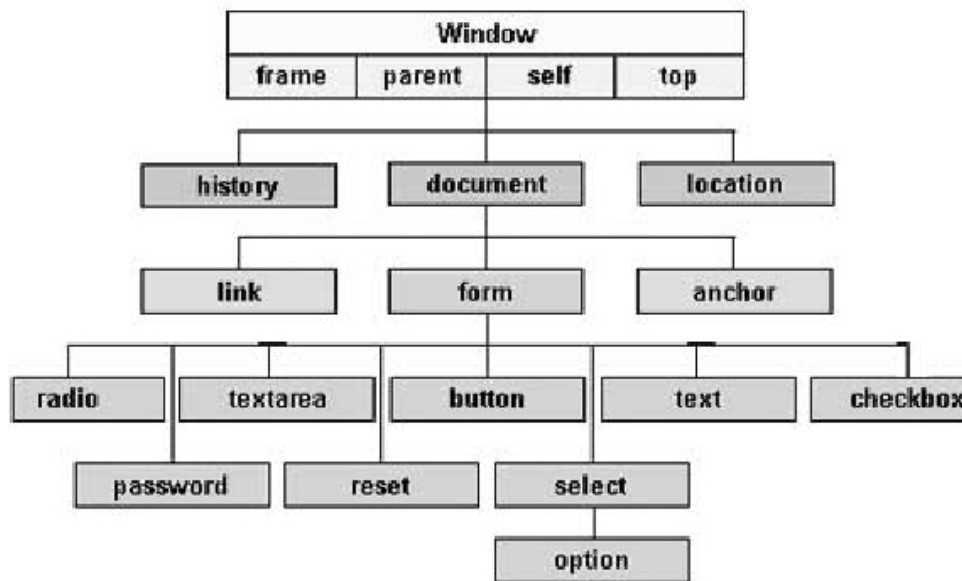


Εικόνα 6 Παράδειγμα κώδικα jquery

3.6 HTML

Η HTML είναι το ακρωνύμιο των λέξεων HyperText Markup Language(γλώσσα μορφοποίησης υπερκειμένου) , είναι μία γλώσσα σήμανσης(markup language), δηλαδή ένας τρόπος γραφής κειμένου και αποτελεί τη βασική γλώσσα δόμησης σελίδων του world wide web.

Ο καθένας μπορεί να δημιουργήσει ένα αρχείο HTML χρησιμοποιώντας έναν επεξεργαστή κειμένου. Χρησιμοποιείται για να σημάνει ένα τμήμα κειμένου και να το κάνει να εμφανίζεται καλύτερα. Επιτρέπει επίσης την ενσωμάτωση ήχου και εικόνας στις web σελίδες. Αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Ο browser αναγνωρίζει αυτόν τον τρόπο γραφής και εκτελεί τις εντολές που περιέχονται σε αυτόν. Αξίζει να σημειωθεί ότι η HTML είναι η πρώτη και πιο διαδεδομένη γλώσσα περιγραφής μίας html σελίδας. Η HTML χρησιμοποιεί τις ειδικές ετικέτες(τα tags) να δώσει τις απαραίτητες οδηγίες στο browser. Τα tags είναι εντολές που συνήθως ορίζουν την αρχή και το τέλος μίας λειτουργίας. Τα tags βρίσκονται πάντα μεταξύ των συμβόλων <και> π.χ. <body>. Οι οδηγίες είναι case insensitive, δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά (μικρά) ή κεφαλαία. Ένα αρχείο HTML πρέπει να έχει κατάληξη htm ή html.



Εικόνα 7 HTML δέντρο

3.7 Cascading Style Sheets (CSS)

Η HTML σχεδιάστηκε σαν γλώσσα δομής για την περιγραφή διαφορετικών σημείων ενός αρχείου (πχ. παράγραφος, λίστα κτλ) και δεν είχε ως σκοπό την οπτική απεικόνιση του αρχείου αυτού αφού επικεντρωνόταν στο «ρόλο» των στοιχείων και όχι στην παρουσίασή τους. Στην προσπάθεια απόδοσης κειμένων στο διαδίκτυο δημιουργήθηκε με την πάροδο του χρόνου η ανάγκη εύχρηστης παρουσίασης αυτών, χωρίς όμως οι εκδόσεις της μεταγλώσσας να επιτρέπουν την εύκολη μορφοποίηση μέσω ιδιαίτερων χαρακτηριστικών, οδηγώντας τους σχεδιαστές διαδικτυακών τόπων σε λύσεις «ανάγκης», που προκαλούσαν προβλήματα λόγω των ιδιοτήτων στην υλοποίησή τους. Ανταποκρινόμενη στις αυξημένες απαιτήσεις των χρηστών αλλά και των δημιουργών αναφορικά με την επίλυση των προβλημάτων αυτών, η HTML εμπλουτίστηκε στην πορεία με στοιχεία παρουσίασης (χρώμα υποστρώματος, χρώμα γραμμάτων, μέγεθος γραμμάτων κτλ). Αποτέλεσμα των επιδράσεων αυτών ήταν η μορφοποίηση του περιεχομένου να επιτυγχάνεται κατά κύριο λόγο μέσω πινάκων και χρήση ετικετών , με άμεση συνέπεια να χαθεί ο αρχικός σκοπός της χρήσης της HTML, ο οποίος ήταν η διαμόρφωση της δομής των εγγράφων, μετατρέποντάς την σε μία «προβληματική» μεταγλώσσα οπτικής παρουσίασης. Ακολουθώντας αυτή τη προσέγγιση, ένας σχεδιαστής χρησιμοποιούσε την ετικέτα για την παρουσίαση των τίτλων των σελίδων (page titles) αντί για τις μετέπειτα υποστηριζόμενες από νεότερες εκδόσεις ετικέτες κεφαλίδας (heading tags), όπως <h1>, <h2> κτλ.



Εικόνα 8 Παράδειγμα σύνταξης css

4 Εργαλεία Ανάπτυξης

4.1 WAMP

Ο WAMP είναι ένας μίνι server ο οποίος μπορεί να λειτουργήσει σε ένα οποιοδήποτε λειτουργικό σύστημα. Το WAMP περιλαμβάνει Apache 2, PHP 5., Mysql (PhpMyAdmin που έχει εγκατασταθεί για τη διαχείριση βάσεων δεδομένων).

Το wamp είναι ένα πρόγραμμα, το οποίο το κατεβάζουμε στον υπολογιστή μας και δημιουργεί ένα server περιβάλλον στο οποίο μπορούμε να τεστάρουμε ιστοσελίδες και scripts που δημιουργήσαμε στον υπολογιστή μας. Η εγκατάσταση χρήση του WAMP είναι πάρα πολύ απλή και εύκολη , όπως προαναφέρθηκε. Το Wamp παρέχει τα τέσσερα βασικά στοιχεία ενός wamp server: επικοινωνία με το λειτουργικό σύστημα (windows), web server (Apache) και λογισμικό δημιουργίας κώδικα web(Php). Η συνδυασμένη αυτή χρήση των προγραμμάτων καλείται στοίβα (server stack). Σε αυτή τη στοίβα , τα windows είναι το λειτουργικό σύστημα, ο Apache είναι ο Web Server , η Mysql διαχειρίζεται τα εξαρτήματα της βάσης δεδομένων και η Php είναι μία δυναμική γλώσσα προγραμματισμού.

Για να ζητήσουμε σελίδες από έναν web server αρκεί να πατήσουμε την διεύθυνσή του, `http://localhost` (εφόσον τρέχουμε τον web server στο pc μας) . Τα αρχεία που μπορεί να στείλει ο server (το wamp) είναι μέσα σε στο `www` directory που δημιουργήσαμε. Και έτσι εμφανίζεται το παρακάτω παράθυρο:



4.2 Adobe Photoshop CS3

Το Photoshop υπήρξε ανέκαθεν το πρότυπο στην επεξεργασία ψηφιακής φωτογραφίας αλλά και στην δημιουργία layout για ιστοσελίδες (template). Μια πολύ χρήσιμη λειτουργία που παρέχει το Photoshop, είναι τα layers.

Φανταστείτε τα layers σαν διαφορετικά επίπεδα το ένα πάνω από το άλλο τα οποία όλα μαζί δημιουργούν μια εικόνα. (γραφικά ή φωτογραφία). Το Photoshop πέρα από τις εκπληκτικές δυνατότητες που έχει στην επεξεργασία εικόνας, με τα χρόνια έχει εξελιχθεί αρκετά, προσφέροντας στους web designers ένα δυνατό εργαλείο για την σχεδίαση web sites. Σε αυτό βοηθάει το εργαλείο slice tool, το οποίο προστέθηκε στο Photoshop στην έκδοση 4 και η επιλογή save for web που προστέθηκε στην έκδοση 5.5. Το slice tool δίνει την δυνατότητα στον χρήστη να «κόβει» την εικόνα που έχει σχεδιάσει σε επιμέρους τμήματα και επιλέγοντας την εντολή save for web να αποθηκεύει το κάθε κομμάτι (slice) σαν ξεχωριστό αρχείο εικόνας. Επιπλέον η επιλογή save for web δίνει, εκτός από την δυνατότητα που έχει να «σπάει» την εικόνα σε επιμέρους εικόνες, την δυνατότητα να αποθηκεύουμε το κάθε κομμάτι (slice) με διαφορετικό είδος συμπίεσης (Gif, Jpeg, Png κτλ.), όπως επίσης και την δυνατότητα να δημιουργεί αρχεία html τα οποία είναι και το τελικό προϊόν (template).

4.3 *PhpMyAdmin*

Το phpMyAdmin είναι ένα εργαλείο γραμμένο σε php που μας επιτρέπει να διαχειριζόμαστε τις βάσεις δεδομένων MySQL. Μπορούμε να εκτελέσουμε ερωτήματα SQL, να εργαστούμε με fields (add, edit, delete), με tables (create, alter, drop), να δημιουργήσουμε πρόσθετες βάσεις δεδομένων και άλλα. Μπορούμε εύκολα να διαχειριστούμε τους πίνακες και τα πεδία που περιέχονται στη βάση δεδομένων μας, να αλλάξουμε τύπους πεδίων και να ενημερώσουμε τους πίνακες της βάσης μας. Ο Tobias Ratschiller , σύμβουλος της IT και αργότερα ιδρυτής της εταιρίας Maguma , ξεκίνησε να εργάζεται βασιζόμενος σε php περιβάλλον το 1998 και όταν το 2000 εγκατέλειψε το project εξαιτίας έλλειψης χρόνου , το phpMyAdmin είχε ήδη γίνει ένα από τις πιο δημοφιλείς php εφαρμογές.

5 Κατασκευή Διαδικτυακών εφαρμογών: Η χρήση του PHP-Apache- MySQL

5.1 Εγκατάσταση APACHE - PHP – MySQL: Χρήση WAMP

Δεδομένου ότι τα Windows PCs δεν τρέχουν Apache (αλλά IIS), είναι απαραίτητο να κατεβάσουμε και να εγκαταστήσουμε το WAMP 2,0 (WampServer 2,0) το οποίο παρέχει την πλατφόρμα για να μιμείται τον web server Apache τοπικά σε έναν υπολογιστή Windows. Το WAMP είναι μία μορφή application server ο οποίος λειτουργεί σε κάθε σχεδόν λειτουργικό σύστημα Windows.

Το WAMP, το οποίο προσφέρεται με GPL licence, περιλαμβάνει και εγκαθιστά στο προσωπικό υπολογιστή τα παρακάτω λογισμικά:

WampServer 2.0: <http://www.wampserver.com/en/download.php>

(March 7 2008)

1. Apache 2.2.8
2. PHP 5.2.5 + PECL
3. SQLitmanager (για τη διαχείριση της MySQL)
4. MySQL 5.0.51a
5. Phpmyadmin (για τη διαχείριση της PHP). Αφού έχουμε κατεβάσει την τελευταία έκδοση του WAMP 2,0, ακολουθούμε τα εξής βήματα για να το εγκαταστήσουμε στον τοπικό υπολογιστή μας:

1. Αφού κατεβεί η τρέχουσα έκδοση (WampServer 2.0), η εγκατάσταση ξεκινά με την επόμενη οθόνη



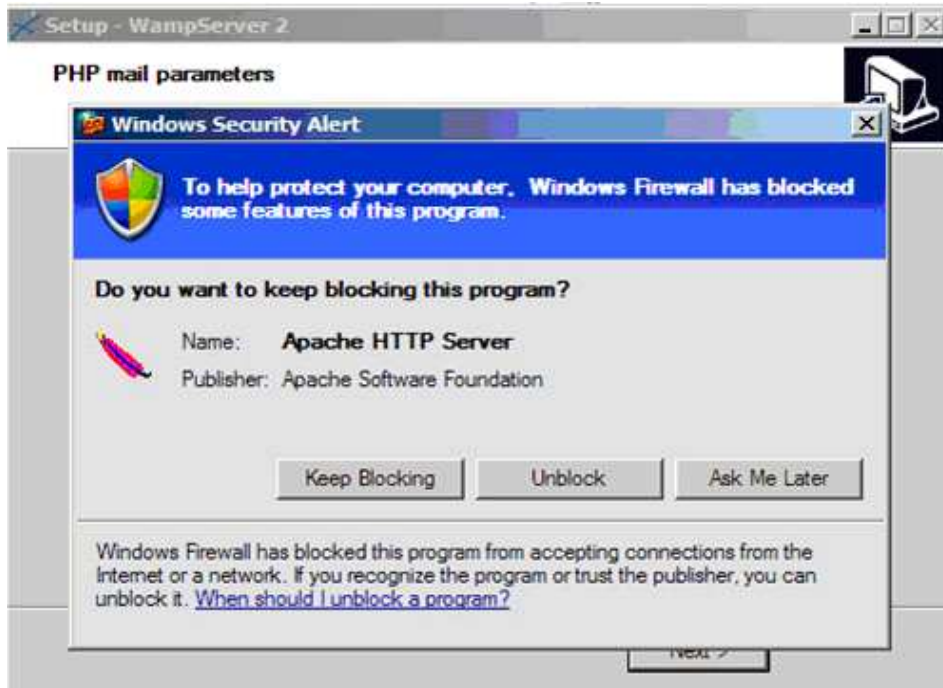
Εικόνα 9 Αρχική οθόνη εγκατάστασης WAMPSEVER

2. Κατόπιν εισάγεται ο υποκατάλογος εγκατάστασης όπου θα αποθηκευτούν τα σχετικά αρχεία εγκατάστασης, ενώ βάση του υποκαταλόγου αυτού καθορίζεται (αρχικά) ο χώρος στον οποίο θα πρέπει να τοποθετούνται οι ιστοσελίδες προκειμένου να γίνουν προσβάσιμες με τη χρήση του.



Εικόνα 10 Καθορισμός υποκαταλόγου εγκατάστασης

3. Στην συνηθισμένη περίπτωση όπου στον τοπικό υπολογιστή υπάρχει εγκατεστημένο και ενεργό το firewall των Windows, σε αυτό θα πρέπει να οριστεί ότι ο web server έχει δικαιώματα πρόσβασης



Εικόνα 11 Αποδέσμευση λειτουργιών Apache Web Server

4. Η επιτυχής εγκατάσταση όλων των συστατικών επαληθεύεται με το παρακάτω μήνυμα



Εικόνα 12 Μήνυμα επιτυχούς εγκατάστασης

Με την ενεργοποίηση του WAMP server, ένα εικονίδιο στη γραμμή εργασιών θα εμφανίζει την κατάστασή του, ενημερώνοντας για το αν:

1. το WAMP είναι σε θέση να εκτελεί υπηρεσίες, αλλά αυτές δεν έχουν ακόμα ενεργοποιηθεί(κόκκινο εικονίδιο),
2. το WAMP έχει ενεργοποιηθεί καθώς και μία υπηρεσία αυτού (κίτρινο εικονίδιο)
3. το WAMP λειτουργεί με όλες τις υπηρεσίες του (λευκό εικονίδιο).

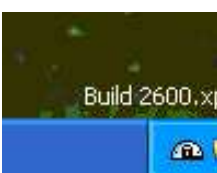
Ο Apache και η MySQL θεωρούνται υπηρεσίες και μπορούν οποτεδήποτε να ενεργοποιηθούν και να απενεργοποιηθούν από το αριστερό κλικ στο εικονίδιο του WAMP. Οι ιστοσελίδες που φιλοξενούνται στο WAMP server μπορούν να προσπελαστούν με την πληκτρολόγηση:

- <http://localhost/> ή
- <http://127.0.0.1/>

στη γραμμή διεύθυνσης του web browser

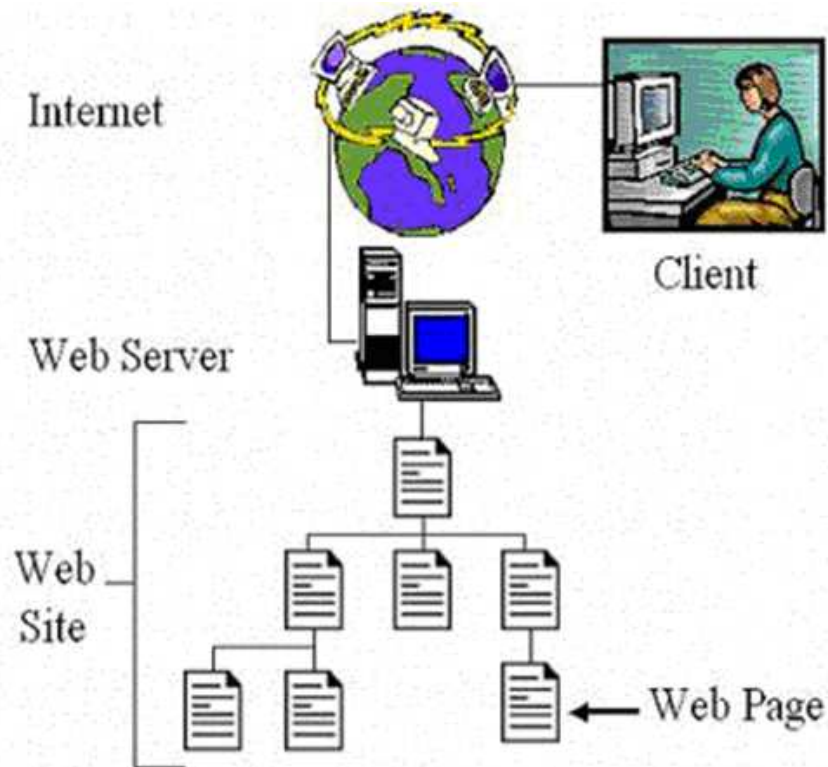


Το Wamp 2.0 ξεκινά. Το εικονίδιο του WampServer φορτώνεται στο δίσκο του συστήματος.



Αφού εγκαταστήσουμε το Wamp Server , δημιουργούμε ένα φάκελο www μέσα στο C:\wamp δηλαδή (C:\wamp\www) που εκεί μέσα βρίσκεται η εργασία και με τα ρηρ αρχεία που δημιουργήσαμε.

Ο χρήστης μέσω web browser και με τη χρήση μίας διεύθυνσης προσπελαύνει περιεχόμενο σε όλο τον κόσμο.



Εικόνα 13 Πρόσβαση χρήστη στα δεδομένα

6 Δημιουργία Δυναμικού Ιστοχώρου: Συστήματος αναζήτησης διακοπών

6.1 Εισαγωγή

Σε αυτή την ενότητα παρατίθενται τα βασικά βήματα για τη δημιουργία ενός δυναμικού ιστοχώρου, από τον οποίο ο τελικός χρήστης μπορεί να πληροφορηθεί σχετικά με τις διαθέσιμες εκδρομές. Επίσης γίνεται αναφορά σε κάποιες εισαγωγικές έννοιες για το τι είναι μία βάση δεδομένων, έτσι ώστε ο αναγνώστης να σχηματίσει μία σφαιρική άποψη επί του θέματος.

6.2 Βάση δεδομένων

Θα ξεκινήσουμε την υλοποίηση σχεδιάζοντας τη βάση δεδομένων βρίσκοντας τις οντότητες(πίνακες) που θα εμπλέκονται στις συσχετίσεις και τα γνωρίσματα της κάθε οντότητας. Ο ρόλος των συσχετίσεων (relationships) σε μια σχεσιακή βάση δεδομένων, είναι η διασύνδεση των πινάκων της βάσης, με τέτοιο τρόπο, ώστε να είναι δυνατή η ανάκτηση δεδομένων, που συσχετίζονται μεταξύ τους, αλλά βρίσκονται αποθηκευμένα σε διαφορετικούς πίνακες.

Ο τρόπος με τον οποίο λαμβάνουν χώρα αυτού του είδους οι συσχετίσεις, καθορίζεται κατά το στάδιο του λογικού σχεδιασμού της βάσης, που οδηγεί τελικά στη δημιουργία του μοντέλου οντοτήτων συσχετίσεων. Η βασική αρχή σχεδίασης αυτού του μοντέλου, είναι η ομαδοποίηση των πεδίων των τύπων οντότητας, και η δημιουργία πινάκων για τους τύπους οντότητας και συσχέτισης.

Δημιουργούμε δηλαδή τους πίνακες με τα γνωρίσματα τα οποία κρίνουμε ότι είναι απαραίτητα για την πραγματοποίηση της εφαρμογής μας. Επιδιώκουμε την σωστή σχεδίαση της δομής των πινάκων της βάσης, έτσι ώστε να ικανοποιεί τις προδιαγραφές που έχουν τεθεί και ταυτόχρονα να είναι ευέλικτη και αποδοτική στη χρήση της . Να είναι δηλαδή εφικτή στους χρήστες η προσπέλαση των δεδομένων της βάσης με απλή και σύνθετη αναζήτηση.

6.3 Συνοπτική περιγραφή εννοιών μίας βάσης δεδομένων

Ορισμός βάσης δεδομένων

Μια Βάση Δεδομένων (DataBase) είναι ένας οργανωμένος τρόπος αποθήκευσης πληροφοριών και πρόσβασής τους με πολλούς τρόπους με διάφορα προγράμματα. Μια βάση δεδομένων είναι κάτι παραπάνω από μια απλή συλλογή αποθηκευμένων στοιχείων.

Ένας άλλος ορισμός είναι ότι μια βάση δεδομένων είναι ένα ολοκληρωμένο σύστημα που αποτελείται από δεδομένα (data) και από το κατάλληλο λογισμικό (software), τα οποία χρησιμοποιώντας το υλικό (hardware) βοηθούν στην ενημέρωση και πληροφόρηση των χρηστών (users).

Ένα πρόγραμμα που διαχειρίζεται βάσεις δεδομένων αποκαλείται Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS, DataBase Management System) και με την βοήθειά του μπορούμε να αποθηκεύσουμε, προσθέσουμε, τροποποιήσουμε, εμφανίσουμε ή και διαγράψουμε τα αποθηκευμένα δεδομένα.

Τα δεδομένα που υπάρχουν στις βάσεις δεδομένων πρέπει να είναι :

- Ολοκληρωμένα (Integrated), δηλ. τα δεδομένα πρέπει να είναι αποθηκευμένα σε ομοιόμορφα οργανωμένα σύνολα αρχείων όπου δεν πρέπει να υπάρχει επανάληψη ή πλεονασμός (redundancy) των ίδιων στοιχείων.
- Καταμεριζόμενα (Shared), δηλ. να μπορούν περισσότεροι του ενός χρήστες να βλέπουν και να μοιράζονται τα ίδια δεδομένα την ίδια χρονική στιγμή.

Στόχοι μίας βάσης δεδομένων

Οι στόχοι μιας βάσης δεδομένων είναι οι εξής :

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

Τα στοιχεία μίας βάσης δεδομένων

Τα δεδομένα μιας βάσης δεδομένων αποθηκεύονται (οργανώνονται) στις εξής στοιχειώδεις μορφές
Πεδίο (Field), είναι το μικρότερο κομμάτι δεδομένων στο οποίο μπορούμε να αναφερθούμε και περιέχει ένα μόνο χαρακτηριστικό ή ιδιότητα ενός στοιχείου της βάσης δεδομένων.

Εγγραφή (Record), είναι ένα σύνολο από διαφορετικά πεδία που περιέχει όλες τις πληροφορίες για ένα στοιχείο της βάσης δεδομένων.

Αρχείο (File), είναι ένα σύνολο από πολλά παρόμοια στοιχεία (εγγραφές) της βάσης δεδομένων.

Πρωτεύον Κλειδί (Primary Key), είναι ένα πεδίο ή συνδυασμός πεδίων που χαρακτηρίζει μοναδικά μια εγγραφή.

Κλειδί (Key), είναι ένα πεδίο που δεν έχει κατ' ανάγκη μοναδική τιμή και που μπορούμε να το χρησιμοποιήσουμε για να κάνουμε αναζήτηση σ' ένα αρχείο.

Ξένο Κλειδί (Foreign Key), είναι ένα πεδίο που έχει το ίδιο σύνολο τιμών με το πρωτεύον κλειδί ενός άλλου αρχείου.

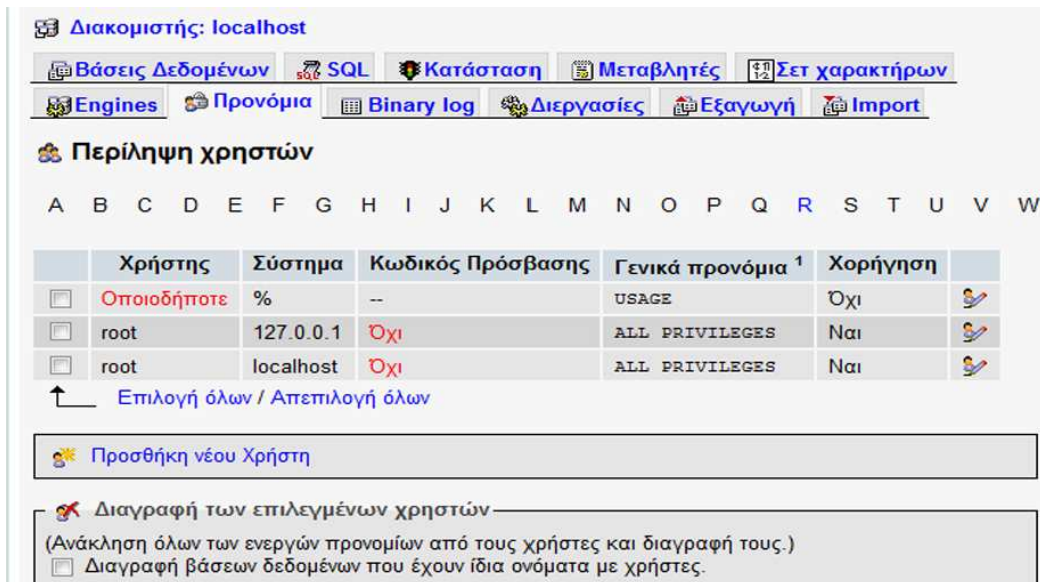
6.4 Δημιουργία βάσης δεδομένων και χρήση

Κάθε πρόγραμμα προσομοίωσης server για τοπική εργασία όπως το wamp διαθέτουν Mysql server και ένα πρόγραμμα διαχείρισης της Mysql που λέγεται Phpmyadmin. Επομένως η πρώτη μας κίνηση είναι να εντοπίσουμε και να ανοίξουμε το Phpmyadmin. Το πρόγραμμα αυτό επικοινωνεί με τον mysql server και αριστερά στην οθόνη μας δείχνει τις υπάρχουσες βάσεις δεδομένων. Σε περίπτωση τοπικού επιπέδου όλος ο server μας ανήκει και μπορούμε να δημιουργούμε απεριόριστες βάσεις συνήθως μια για κάθε script που έχουμε υπο έλεγχο ή κατασκευή. Για να ανοίξουμε το Phpmyadmin για πρώτη φορά ψάχνουμε στο control panel που μας διαθέτει ο provider. Για να δημιουργήσουμε μια βάση δεδομένων συμπληρώνουμε το όνομα της βάσης στο πεδίο "Δημιουργία νέας βάσης" στην αρχική σελίδα του. Δίπλα από το πεδίο με το όνομα της βάσης θα πρέπει να καθορίσουμε τα Collation της βάσης. Εδώ θέλει προσοχή εφόσον στην βάση αποθηκεύουμε και δεδομένα στα Ελληνικά. Για να έχουμε σωστά Ελληνικά και στην βάση και στις σελίδες του site μας πρέπει να δώσετε σωστό τύπο collation έτσι επιλέγουμε το utf8_general_ci

Πατάμε δημιουργία και η βάση θα δημιουργηθεί στο server. Το όνομα της βάσης θα εμφανιστεί αριστερά και δίπλα μέσα σε παρένθεση θα λέει 0. Αυτό δείχνει οτι ακόμη δεν υπάρχουν πίνακες μέσα στην βάση.

Επιλέγουμε τη βάση που δημιουργήσαμε και έπειτα μας δίνεται η δυνατότητα να δημιουργήσουμε κάποιο πίνακα στην υπάρχουσα βάση εφόσον συμπληρώσουμε το όνομα του πίνακα που θέλουμε να δημιουργήσουμε και τα πεδία από τα οποία θα αποτελείται, καθώς και κάποιους παραμέτρους που θα έχουν τα πεδία αυτά(όπως ο τύπος, π.χ. ακέραιος). Αν όλα πάνε καλά θα εισαχθούν όλα δομή και δεδομένα στην βάση που δημιουργήσαμε πριν λίγο.

Αν φτάσουμε έως εδώ μένει ένα ακόμη βήμα μέχρι την ολοκλήρωση. Το script για να επικοινωνήσει με την βάση χρειάζεται να γνωρίζει 4 στοιχεία. Αυτά είναι: το που είναι ο server (συνήθως αυτό είναι localhost), το όνομα της βάσης (αυτό που δώσατε προηγουμένως κατά την δημιουργία της βάσης) και username και password του χρήστη που έχει δικαίωμα πρόσβασης στη βάση (συνήθως για τοπική χρήση wamp είναι το username "root" και το password κενό ή ότι έχετε δώσει.) . Αυτά τα στοιχεία τα ρυθμίζουμε όταν πατήσουμε στο διακομιστή:localhost και επιλέγοντας την επιλογή προνόμια. Πατώντας σε αυτή την επιλογή μας δίνεται η δυνατότητα να ρυθμίσουμε ανάλογα με το ποιος θέλουμε να είναι ο διαχειριστής της βάσης, τα δικαιώματα που θα έχει στη βάση. Στη δική μας περίπτωση ο διαχειριστής έχει κάθε δικαίωμα επεξεργασίας του περιεχομένου της βάσης για αυτό και επιλέγουμε ALL PRIVILEGES.



Εικόνα 14 Προβολή χρήστη και προνομίων

6.5 Αρχιτεκτονική της βάσης δεδομένων

Η βάση δεδομένων travel_agency είναι η καρδιά του συστήματος. Αποτελείται από 5 πίνακες στους οποίους αποθηκεύονται όλα τα δεδομένα του συστήματος

Πίνακας	Ενέργεια	Εγγραφές 1	Τύπος	Collation	Μέγεθος	Επιβάρυνση
<input type="checkbox"/> category		11	MyISAM	utf8_general_ci	2,8 KB	80 Bytes
<input type="checkbox"/> continent		6	MyISAM	utf8_general_ci	2,3 KB	160 Bytes
<input type="checkbox"/> country		8	MyISAM	utf8_general_ci	6,9 KB	-
<input type="checkbox"/> offers		6	MyISAM	utf8_general_ci	8,9 KB	1,2 KB
<input type="checkbox"/> users		3	MyISAM	utf8_general_ci	2,4 KB	-
5 Πίνακες/Πίνακες	Σύνολο	34	MyISAM	utf8_general_ci	23,3 KB	1,4 KB

Εικόνα 15 Πίνακες βάσης δεδομένων travel_agency

Στη δημιουργία της εφαρμογής του συστήματος αναζήτησης διακοπών θα πρέπει να δημιουργήσουμε ένα πίνακα(οντότητα) για τους users, δηλαδή τα τουριστικά γραφεία. Η οντότητα αυτή αποτελείται από κάποια χαρακτηριστικά ή γνωρίσματα(όνομα, διεύθυνση, φάξ,site, κωδικός, email, εικόνα) τα οποία περιγράφουν την οντότητα και το πρωτεύον κλειδί της οντότητας το οποίο θα μας βοηθήσει στην συσχέτιση μεταξύ των οντοτήτων καθώς και στην ανάκτηση, διαγραφή και επεξεργασία κάποιων δεδομένων από τη βάση μου.

Διακομιστής: localhost ▶ Βάση: travel_agency ▶ Πίνακας : users

Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> id	int(11)			Όχι	None	auto_increment	
<input type="checkbox"/> name	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> address	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> fax	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> site	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> password	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> username	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> email	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> photo	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> image	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> date	int(11)			Όχι	None		

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εικόνα 16 Δομή του πίνακα users

Ένας άλλος πίνακας που θα πρέπει να δημιουργήσουμε είναι οι ήπειροι

Διακομιστής: localhost ▶ Βάση: travel_agency ▶ Πίνακας : continent

Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> id	int(11)			Όχι	None	auto_increment	
<input type="checkbox"/> name	text	utf8_general_ci		Όχι	None		

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εικόνα 17 Δομή του πίνακα continent

Ο πίνακας country, όπου καταχωρούνται όλοι οι διαθέσιμοι προορισμοί. Όπως βλέπουμε το ένα από τα γνωρίσματα αυτού του πίνακα είναι το id της ηπείρου(continent_id), δηλαδή πραγματοποιείται κάποια συσχέτιση μεταξύ των πινάκων country και continent και έτσι δίνεται η δυνατότητα επιλέγοντας τη χώρα να εμφανίζεται και σε ποια ήπειρο ανήκει αυτή η χώρα(συμπληρώνοντας το continent_id στο πίνακα country, το οποίο id αντιστοιχεί σε κάποια συγκεκριμένη καταχώρηση στο πίνακα continent)

Διακομιστής: localhost ▶ Βάση: travel_agency ▶ Πίνακας : country

Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> id	int(11)			Όχι	None	auto_increment	
<input type="checkbox"/> name	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> history	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> continent_id	int(11)			Όχι	None		
<input type="checkbox"/> image	text	utf8_general_ci		Όχι	None		
<input type="checkbox"/> history_img	text	utf8_general_ci		Όχι	None		

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εικόνα 18 Δομή του πίνακα country

Ο πίνακας category, όπου καταχωρούνται οι διαθέσιμες κατηγορίες

Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> id	int(11)			Όχι	None	auto_increment	[Icons]
<input type="checkbox"/> name	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> image	text	utf8_general_ci		Όχι	None		[Icons]

Εικόνα 19 Δομή του πίνακα category

Και τέλος ο πίνακας offers, ο οποίος περιέχει όλες τις διαθέσιμες προσφορές και στον οποίο ως γνωρίσματα εμπεριέχονται τα κλειδιά των άλλων πινάκων(δηλαδή το country_id, category_id, user_id), πραγματοποιώντας έτσι συσχετίσεις μεταξύ των πινάκων αυτών. Το όφελος των εν λόγω συσχετίσεων βοηθάει στο να γίνεται εφικτή στους χρήστες η διαθεσιμότητα των προσφορών τόσο με απλή όσο και σύνθετη αναζήτηση.

Πεδίο	Τύπος	Collation	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα	Ενέργεια
<input type="checkbox"/> id	int(11)			Όχι	None	auto_increment	[Icons]
<input type="checkbox"/> title	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> description	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> country_id	int(11)			Όχι	None		[Icons]
<input type="checkbox"/> category_id	int(11)			Όχι	None		[Icons]
<input type="checkbox"/> user_id	int(11)			Όχι	None		[Icons]
<input type="checkbox"/> date_from	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> date_to	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> price	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> image	text	utf8_general_ci		Όχι	None		[Icons]
<input type="checkbox"/> map	text	utf8_general_ci		Όχι	None		[Icons]

Εικόνα 20 Δομή του πίνακα offers

Συσχετίσεις μεταξύ των πινάκων της βάσης δεδομένων

Οι συσχετίσεις καθορίζουν με μονοσήμαντο τρόπο το είδος της αλληλεπίδρασης που υφίσταται ανάμεσα στους πινάκων της εφαρμογής. Δύο οντότητες (πίνακες) μπορεί να συνδέονται με μία ή περισσότερες συσχετίσεις των παρακάτω τύπων:

Ένα – προς - Ένα (1:1)

Ένα – προς – Πολλά (1:M)

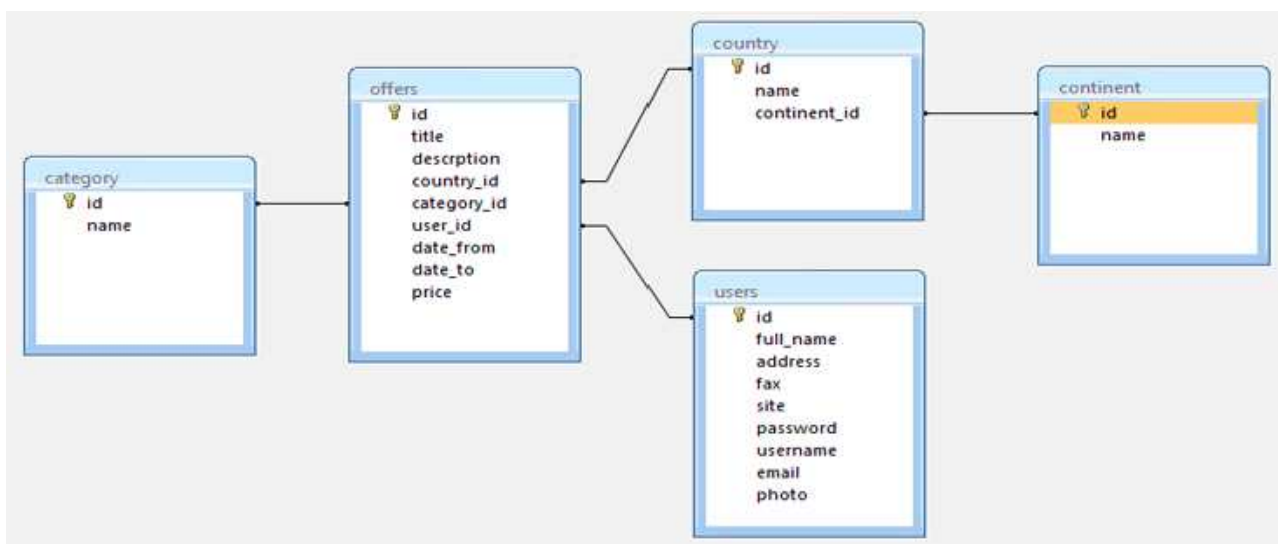
Πολλά – προς – Ένα (M:1)

Πολλά– προς – Πολλά (M:M)

Στη παρούσα εφαρμογή έχουμε τις εξής συσχετίσεις:

1. Οι οντότητες country και continent συνδέονται με μία συσχέτιση «ένα προς πολλά», γιατί σε κάθε ήπειρο μπορεί να αντιστοιχούν πολλές χώρες.
2. Οι οντότητες offers και category συνδέονται με μία συσχέτιση «πολλά προς ένα», γιατί μπορεί να έχουμε πολλές προσφορές που να αντιστοιχούν στην ίδια κατηγορία.
3. Όμοια οι οντότητες offers και country συνδέονται με μία συσχέτιση «πολλά προς ένα», γιατί μπορεί να έχουμε πολλές προσφορές που να αντιστοιχούν στην ίδια χώρα.
4. Και τέλος οι οντότητες offers και users συνδέονται με μία συσχέτιση «πολλά προς ένα», γιατί μπορεί να έχουμε πολλές προσφορές που να αντιστοιχούν στο ίδιο τουριστικό γραφείο.

Όλες αυτές οι συσχετίσεις μεταξύ των οντοτήτων(πινάκων) που προαναφέρθηκαν, φαίνονται στο παρακάτω σχήμα μέσα από το περιβάλλον της Microsoft Access.



Εικόνα 21 Συσχετίσεις μεταξύ των πινάκων της βάσης

7 Δομή και Οργάνωση της Πληροφορίας

Από τα βασικότερα θέματα που αφορά την αρχική σχεδίαση ενός ιστοχώρου, η οποία επηρεάζει καταλυτικά το βαθμό προσβασιμότητας και ευχρηστίας του, είναι η ορθή χωροθέτηση της πληροφορίας που περιέχει. Κάθε χρήστης εξερευνώντας τα διάφορα μέρη του διαδικτυακού τόπου, αναζητώντας την προσφερόμενη πληροφορία και τις υπηρεσίες του, θα πρέπει εύκολα να μπορεί να συνθέσει μια συγκεκριμένη και σταθερή «εικόνα» για τη δομή του.

Σημαντικό σημείο αναφοράς στην εικόνα είναι η τοποθέτηση των στοιχείων που αφορούν την πλοήγηση, τα οποία θα πρέπει να βρίσκονται σε σταθερά ευκρινή σημεία – ζώνες, ενώ η κατανομή της υπόλοιπης πληροφορίας είναι στη δημιουργική ευχέρεια του σχεδιαστή. Η έλλειψη δομής και αυτών των ευκρινών σημείων πλοήγησης δημιουργεί αρκετά προβλήματα στους χρήστες κατά τη διαδικασία ανάκτησης της πληροφορίας, καθώς και στην περαιτέρω ανάπτυξη του περιεχομένου του χώρου από τους διαχειριστές του. Για την ευκολότερη οριοθέτηση των περιοχών αυτών η κάθε σελίδα του διαδικτυακού τόπου θα μπορούσε να χωριστεί νοητά σε τέσσερις κύριες ζώνες:

- Κεφαλίδα (Header)
- Πλοήγηση (Navigation)
- Κύριο μέρος (Main Page)
- Υποσέλιδο (Footer)
-

Σκοπός μας λοιπόν είναι να δημιουργήσουμε όπως προαναφέρθηκε ένα template(καλούπι), του οποίου η δομή θα είναι η ίδια για όλες τις σελίδες του μενού, δηλαδή θα αποτελείται από την κεφαλίδα, την πλοήγηση, το κυρίως μενού και το υποσέλιδο. Στη συγκεκριμένη περίπτωση η δομή της εφαρμογής είναι η εξής:



Εικόνα 22 Διάταξη συνδέσμων πλοήγησης σε σχήμα "Γ"

Στην **κεφαλίδα** συμπεριλαμβάνονται:

- Οι σύνδεσμοι προσβασιμότητας (Accessibility links). Αυτοί οι σύνδεσμοι πρέπει να είναι το πρώτο "αντικείμενο" στη σελίδα, για λόγους καλύτερης προσβασιμότητας.
- Η "εργαλειοθήκη" δικτυακού τόπου (Site Toolbar), η οποία συνήθως βρίσκεται στοιχισμένη δεξιά και περιλαμβάνει : Λογότυπο

Πλοήγηση:

Οι σύνδεσμοι πλοήγησης, συνήθως, συμπεριλαμβάνονται σε οριζόντια ή σε κατακόρυφη μπάρα σε ένα σταθερό σημείο της ιστοσελίδας. Σε σύνθετες δομές, η πλοήγηση είναι πιο σύνθετη και για το λόγο αυτό οι σύνδεσμοι κατηγοριοποιούνται ανάλογα με τη λειτουργικότητα τους. Ένα παράδειγμα κατηγοριοποίησης θα μπορούσε να είναι το ακόλουθο:

- Κύριοι σύνδεσμοι πλοήγησης (Main navigation)
- Δευτερεύοντες σύνδεσμοι πλοήγησης (Secondary navigation)

Σε αυτή την περίπτωση δημιουργούνται δύο μπλοκ πλοήγησης τα οποία μπορούν να τοποθετηθούν(όπως στην περίπτωση της παρούσας εφαρμογής, όπου οι σύνδεσμοι του μενού για να κάνουν οι χρήστες απλή αναζήτηση είναι οι κύριοι σύνδεσμοι πλοήγησης ενώ οι δευτερεύοντες σύνδεσμοι πλοήγησης είναι κάποια κριτήρια που έχει ορίσει ο κατασκευαστής της βάσης προκειμένου να γίνεται μία γρήγορη αναζήτηση):Σε σχήμα "Γ" (Top-and-left navigation)

Κύριο μέρος σελίδας

Στο μέρος αυτό τοποθετείται το περιεχόμενο μίας ιστοσελίδας. Οι περισσότεροι χρήστες διαβάζουν συνήθως τις πρώτες παραγράφους και προτάσεις στην αρχή του περιεχομένου και με αυτόν τον τρόπο αποφασίζουν για το αν το περιεχόμενο εμπίπτει στα ενδιαφέροντα τους. Αυτό το κύριο μέρος της σελίδας θα είναι το μοναδικό κομμάτι το περιεχόμενο του οποίου θα μεταβάλλεται κάθε φορά που θα γίνει επιλογή σε κάποιο από τα link του μενού και θα αντλεί τα περιεχόμενα αυτά από τη βάση με τη χρήση sql ερωτημάτων, προκειμένου να γίνονται διαθέσιμα στο χρήστη της εφαρμογής.

Υποσέλιδο

Το υποσέλιδο συνήθως περιέχει συνδέσμους πλοήγησης, σχετικά με το περιεχόμενο της σελίδας όπως υπεύθυνο σελίδας

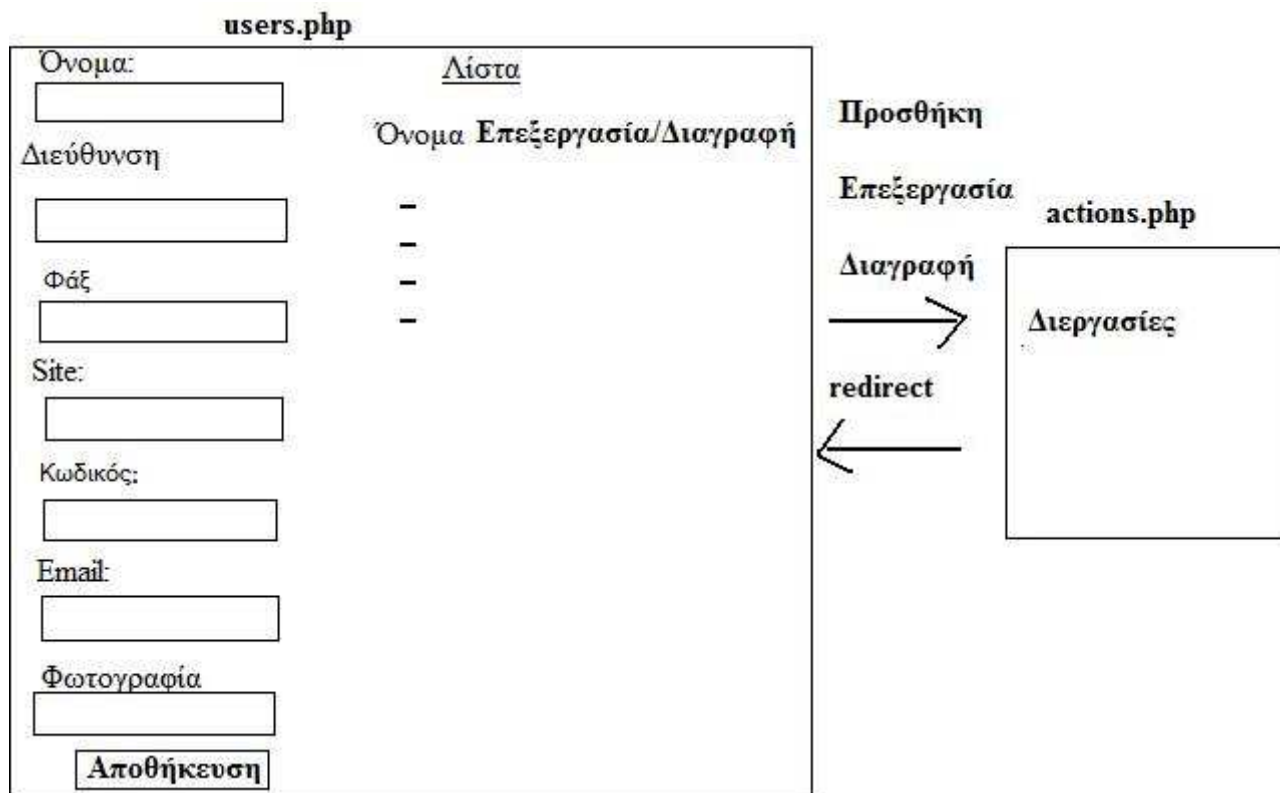
Η διαμόρφωση της δομής των επιμέρους σελίδων της εφαρμογής θα γίνει με τη χρήση της HTML και CSS , η διασύνδεση μεταξύ των σελίδων του μενού θα γίνει με τη χρήση της PHP, ενώ η άντληση της πληροφορίας από τη βάση δεδομένων θα γίνει με sql ερωτήματα με τη χρήση MySQL

8 Δομή και Οργάνωση ιστοχώρου

Το επόμενο βήμα είναι να δημιουργήσουμε κάποια html αρχεία που θα περιέχουν κομμάτια php κώδικα και θα καλούνται κάθε φορά που επισκέπτεται κάποιος χρήστης τη σελίδα έτσι ώστε να γίνονται διαθέσιμες οι πληροφορίες που έχουν καταχωρηθεί στη βάση μέσω του Apache.

Μέσω του Apache γίνεται μετάφραση του PHP κώδικα σε HTML ο οποίος αποστέλλεται στο browser και έτσι τα html αυτά αρχεία εμφανίζονται στην άλλη έξοδο που θα δει ο επισκέπτης.

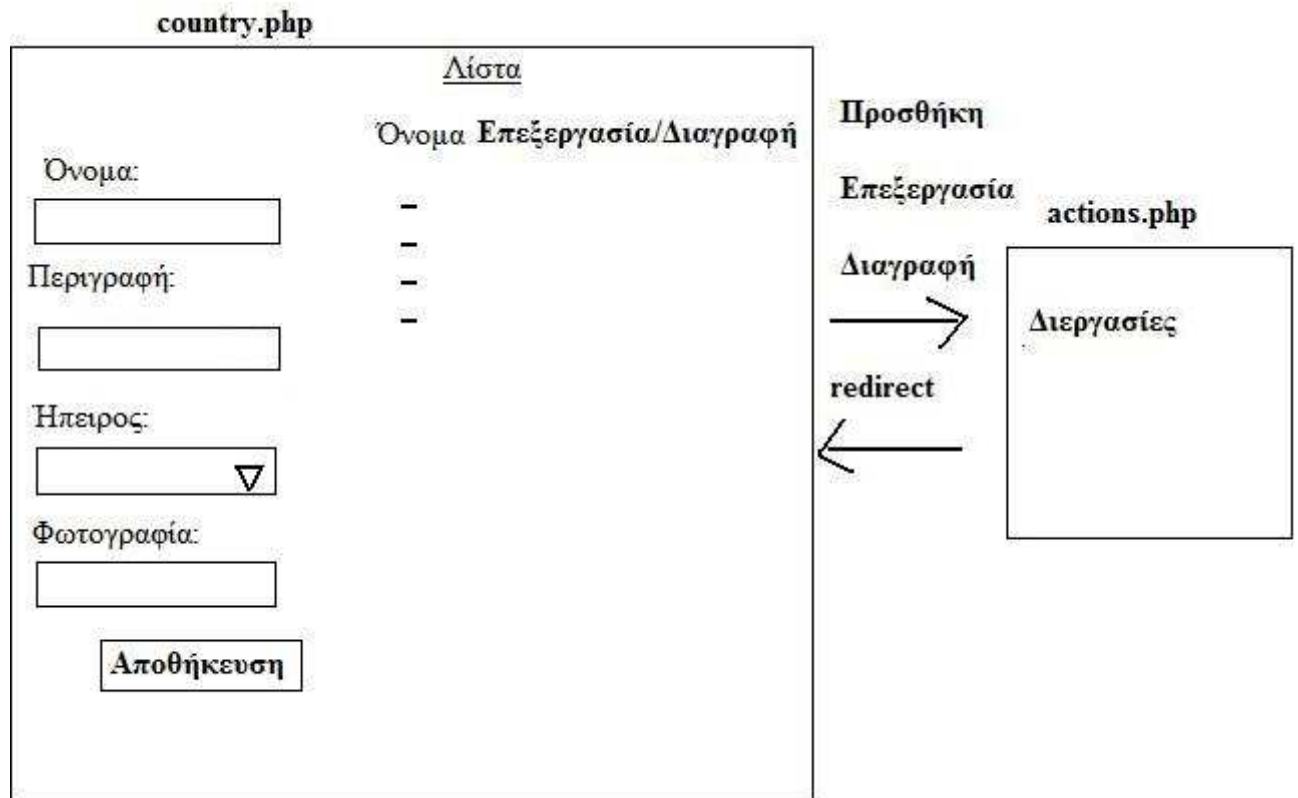
Στη συγκεκριμένη εφαρμογή τα αρχεία που δημιουργήθηκαν προκειμένου να ικανοποιούνται οι απαιτήσεις της εφαρμογής και να γίνονται διαθέσιμες οι πληροφορίες που έχουν καταχωρηθεί στη βάση είναι τα εξής:



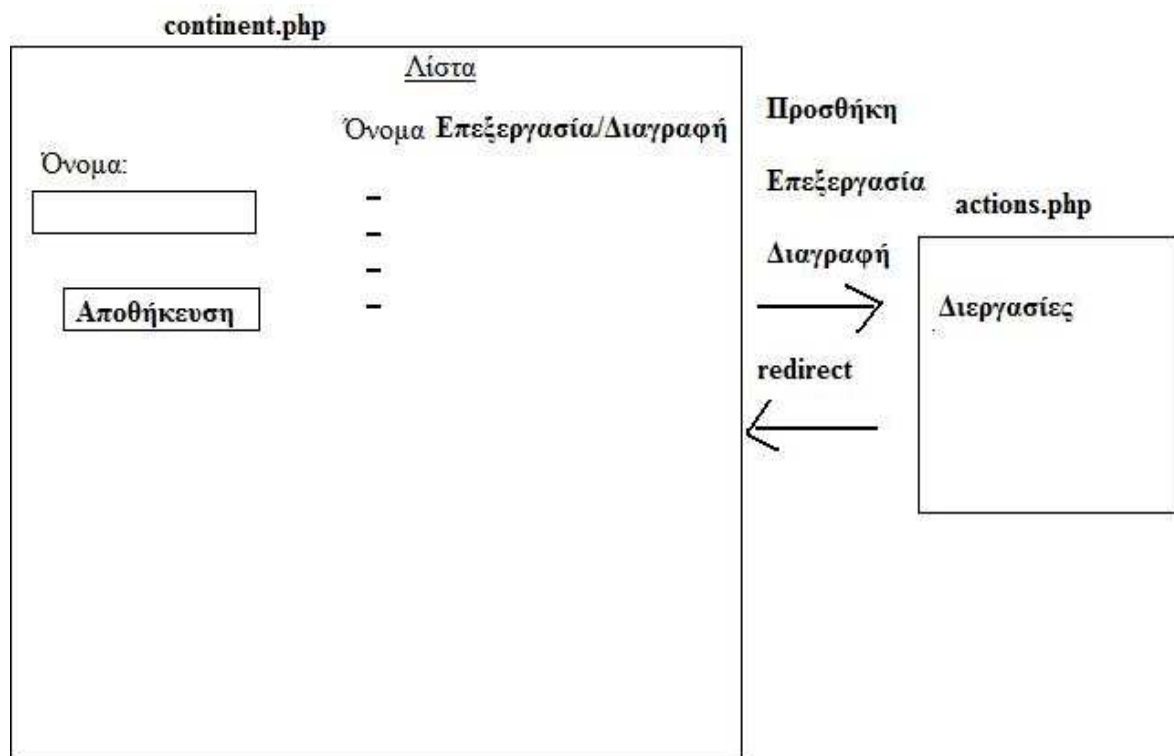
Εικόνα 23 Δομή και οργάνωση αρχείου users

Όπως βλέπουμε στο παραπάνω σχήμα έχουν δημιουργηθεί δύο αρχεία, ένα για τους χρήστες(τα τουριστικά γραφεία), το οποίο περιέχει μία φόρμα η οποία αποτελείται από τα πεδία της βάσης που δημιουργήσαμε, προκειμένου να γίνεται η προσθήκη στοιχείων από τη φόρμα στη βάση(και αντίστροφα), επίσης στο αρχείο αυτό υπάρχει μία λίστα όπου εμφανίζονται τα υπάρχοντα ονόματα των τουριστικών γραφείων και δίπλα υπάρχουν δύο επιλογές(επεξεργασίας και διαγραφής των υπάρχοντων τουριστικών γραφείων). Αυτές οι διεργασίες που προαναφέρθηκαν(προσθήκη, διαγραφή, επεξεργασία τουριστικού γραφείου) γίνονται σε ένα ξεχωριστό αρχείο, το `actions.php`.

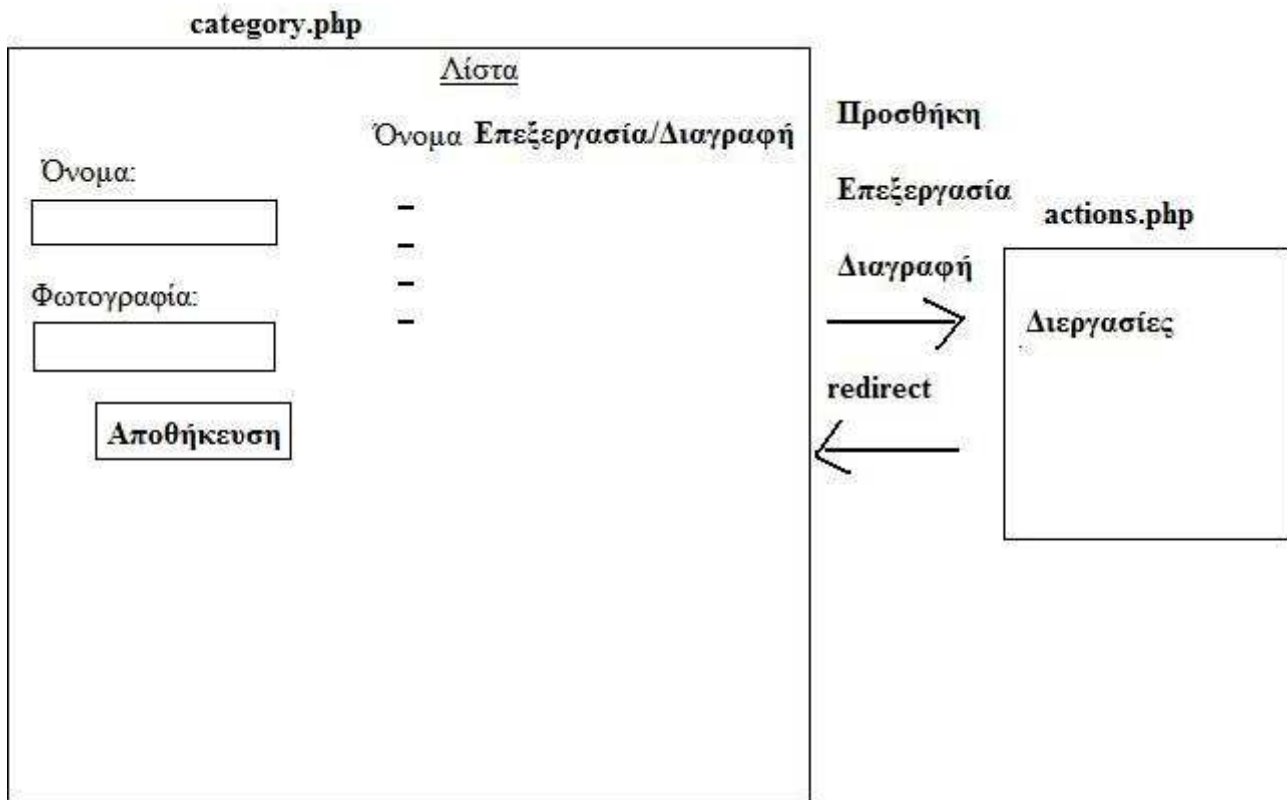
Ομοίως:



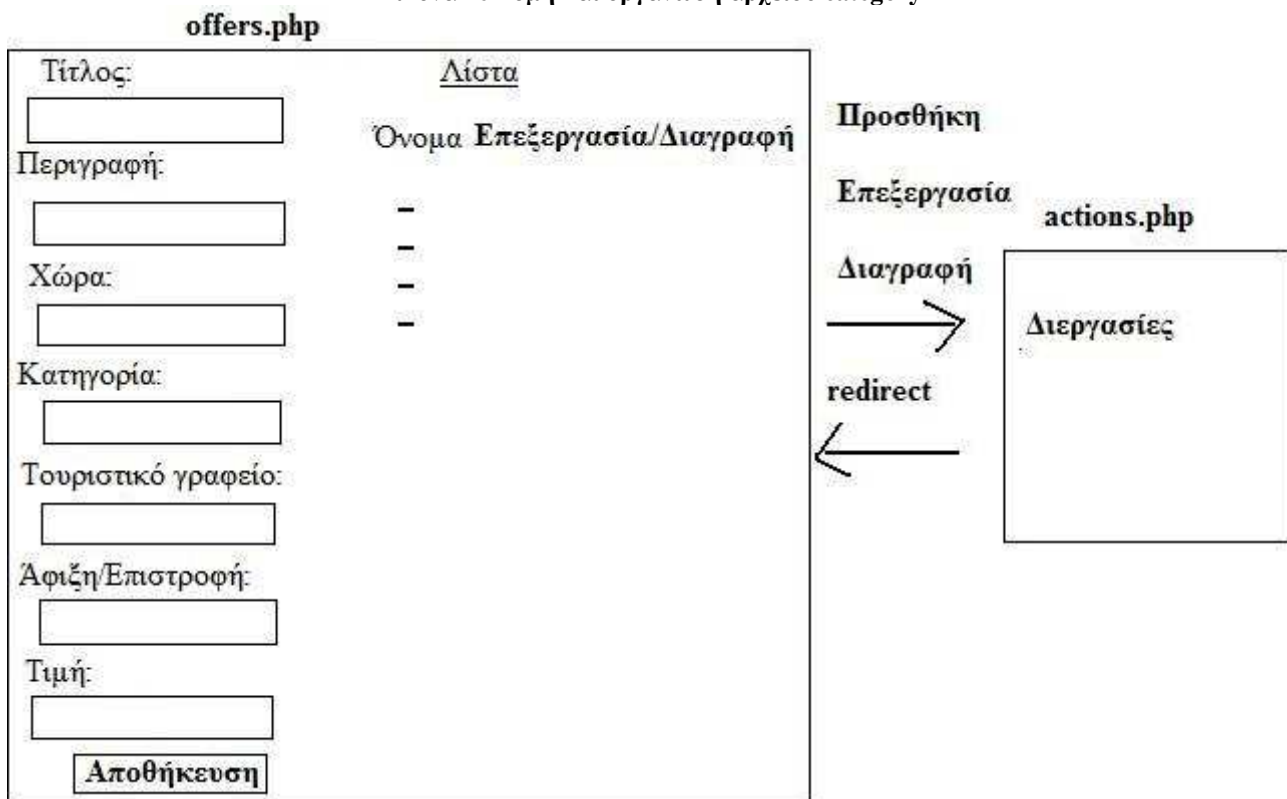
Εικόνα 24 Δομή και οργάνωση αρχείου country



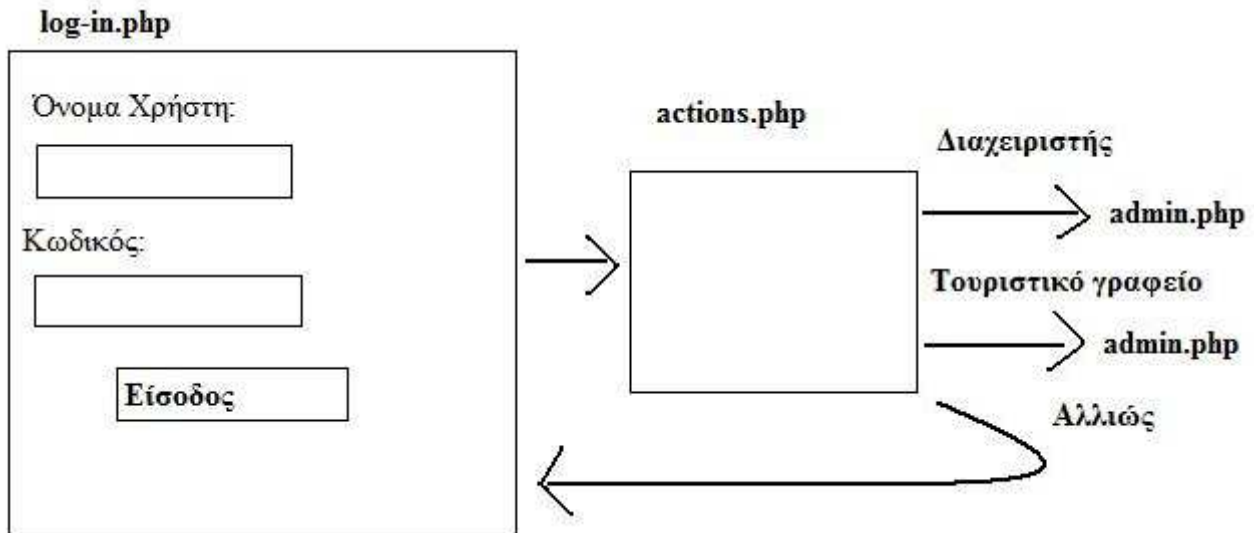
Εικόνα 25 Δομή και οργάνωση αρχείου continent



Εικόνα 26 Δομή και οργάνωση αρχείου category



Εικόνα 27 Δομή και οργάνωση αρχείου offers



Εικόνα 28 Δομή και οράνωση αρχείου log-in

Στο παραπάνω σχήμα βλέπουμε δύο αρχεία. Το log-in.php και το actions.php τα οποία είναι και τα βασικότερα αρχεία καθώς εξασφαλίζουν την ακεραιότητα της βάσης. Εφόσον κάποιος χρήστης(διαχειριστής ή τουριστικό γραφείο), συμπληρώσει το όνομα του(username) και τον κωδικό του(password) , γίνεται κάποιος έλεγχος στο actions.php με τα στοιχεία της βάσης. Αυτός ο έλεγχος είναι αν ο χρήστης με το όνομα και τον κωδικό που προσπαθεί να κάνει Log-in είναι ήδη εγγεγραμμένος στη βάση. Αν είναι εγγεγραμμένος τότε του επιτρέπεται η πρόσβαση, αλλιώς τον στέλνει πίσω στο αρχείο log-in και του εμφανίζει ξανά τη φόρμα για να ξανα πληκτρολογήσει τα στοιχεία του.

Στη περίπτωση που ο χρήστης που κάνει log_in είναι ο διαχειριστής τότε έχει όλα τα δικαιώματα επεξεργασίας του περιεχομένου, καθώς έχει πρόσβαση σε όλες τις προσφορές όλων των τουριστικών γραφείων. Αντίθετα όταν κάποιο τουριστικό γραφείο κάνει log-in έχει περιορισμένα δικαιώματα, καθώς το κάθε τουριστικό γραφείο έχει πρόσβαση μόνο στις δικές του προσφορές, τις οποίες έχει τη δυνατότητα να επεξεργαστεί και να προσθαφαιρέσει. Με τον τρόπο λοιπόν αυτό διασφαλίζεται η ακεραιότητα των προσφορών του κάθε τουριστικού γραφείου.

Δικαιώματα για admin και Τουριστικά γραφεία:

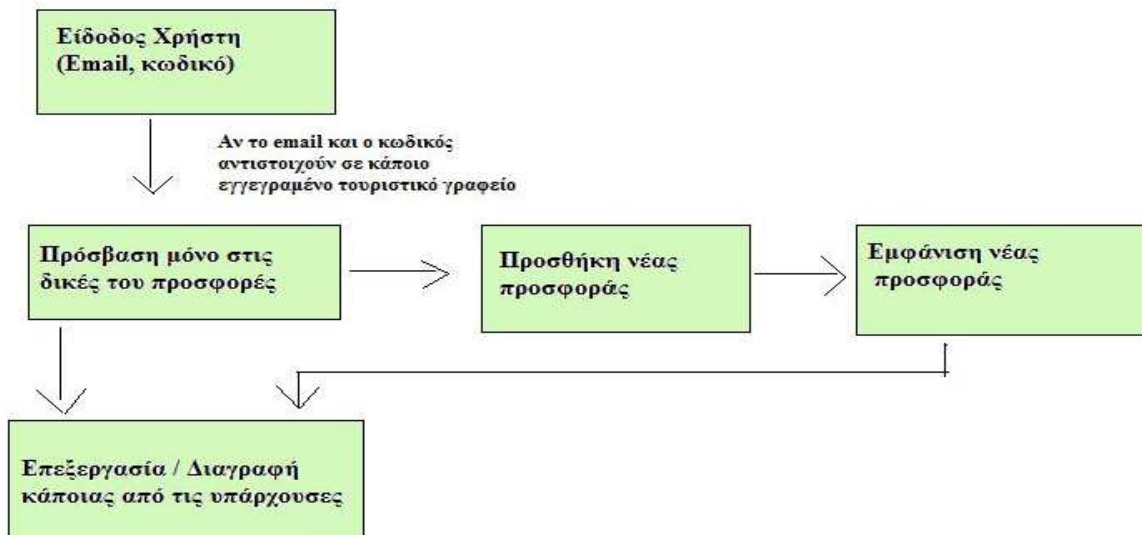
	admin	Τουριστικά γραφεία
user	✓	✗
category	✓	✗
continent	✓	✗
country	✓	✗
offer	✓	✓

Εικόνα 29 Δικαιώματα admin και τουριστικών γραφείων στην εφαρμογή

Παρακάτω απεικονίζονται τα διαγράμματα των λειτουργιών που μπορεί να κάνει ο διαχειριστής και τα τουριστικά γραφεία κάνοντας log_in στην εφαρμογή

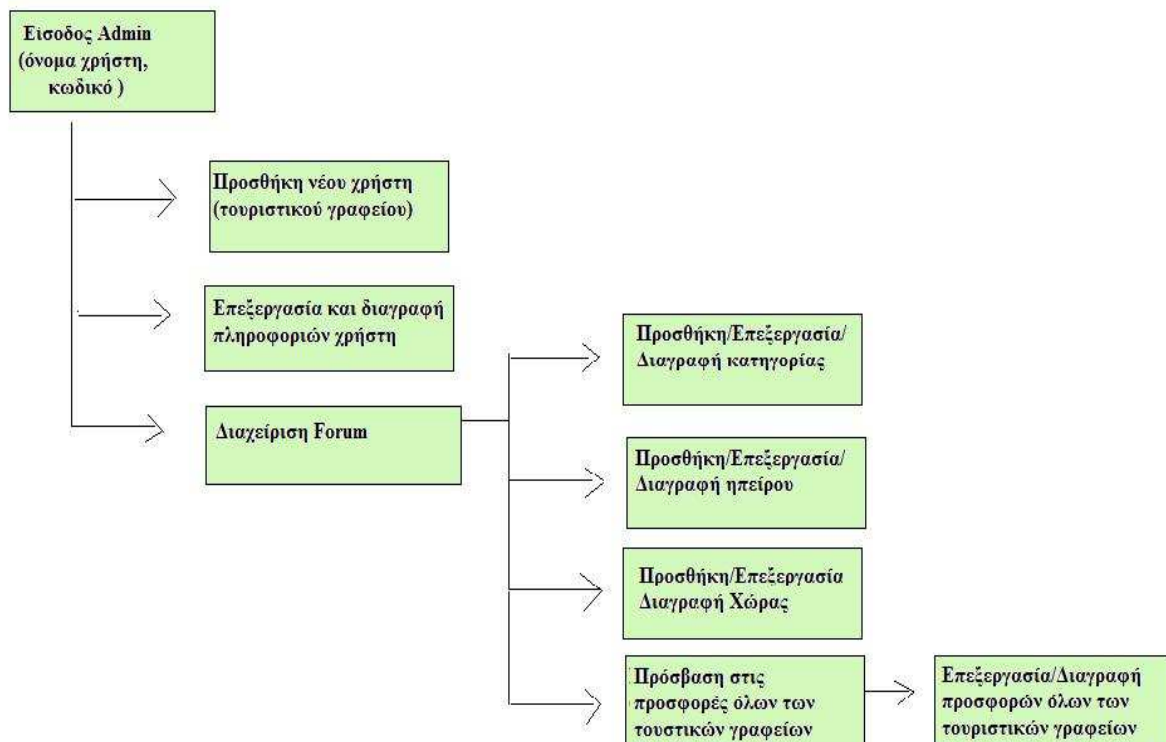
8.1 Διαγράμματα περιήγησης χρηστών

Είσοδος Τουριστικού γραφείου στην εφαρμογή



Εικόνα 30 Δυνατότητες τουριστικού γραφείου στην εφαρμογή

Είσοδος Διαχειριστή στην εφαρμογή:

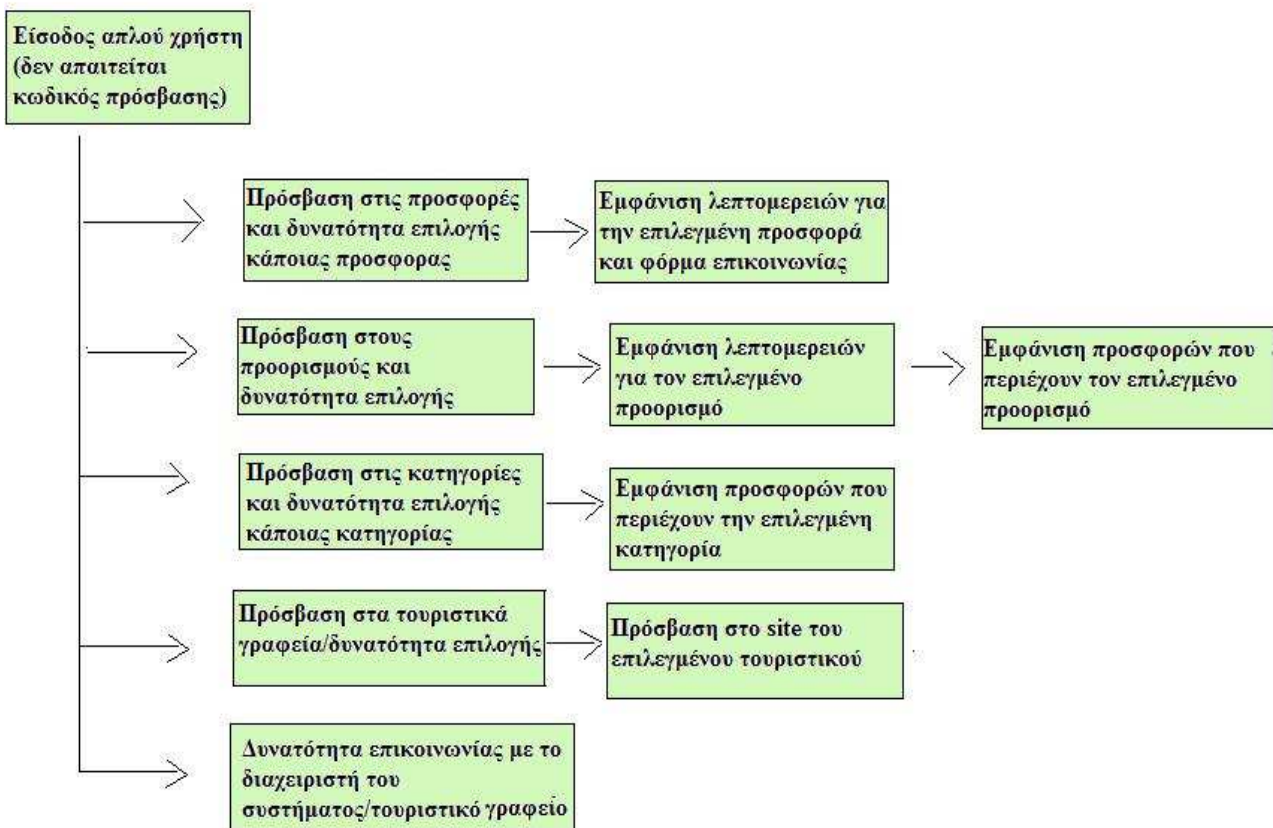


Εικόνα 31 Δυνατότητες διαχειριστή στην εφαρμογή

Στο διάγραμμα φαίνονται όλες οι λειτουργίες που μπορεί να κάνει ο διαχειριστής.

Αρχικά πρέπει να δώσει όνομα και κωδικό πρόσβασης για να μπορέσει να μπει στις λειτουργίες. Ο διαχειριστής μπορεί να εγγράψει έναν χρήστη (τουριστικό γραφείο)στο φόρουμ , να δει τις πληροφορίες των εγγεγραμμένων μελών , να τις αλλάξει ή ακόμη και να διαγράψει τον χρήστη. Επίσης έχει τη δυνατότητα διαχείρισης του Forum με την προσθήκη/διαγραφή/επεξεργασία κάποιας κατηγορίας, ηπείρου, χώρας καθώς επίσης και την επεξεργασία και διαγραφή των προσφορών όλων των τουριστικών γραφείων στις οποίες έχει πρόσβαση

Είσοδος απλού χρήστη στην εφαρμογή:



Εικόνα 32 Δυνατότητες απλού χρήστη στην εφαρμογή

Οποιοσδήποτε απλός χρήστης έχει πρόσβαση στην εφαρμογή, προκειμένου να αναζητήσει την διαθεσιμότητα εκδρομών.

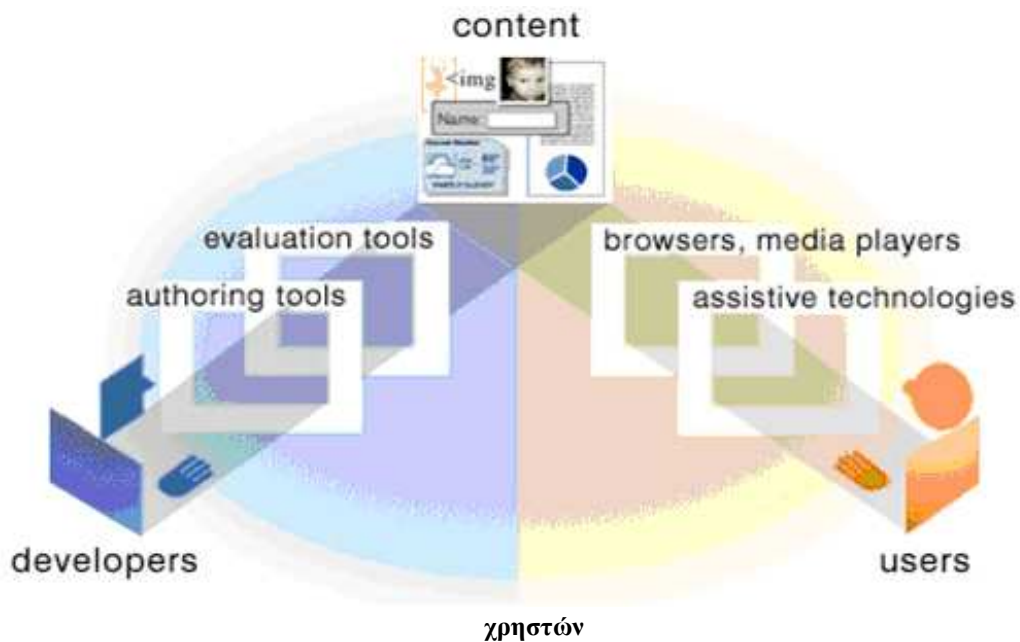
9 Περιγραφή της λειτουργίας των χρηστών αναλυτικά

Υπάρχουν τρία επίπεδα χρηστών που έχουν πρόσβαση στην εφαρμογή. Οι χρήστες κάθε επιπέδου έχουν πρόσβαση σε διαφορετικά τμήματα και λειτουργίες της εφαρμογής.

Τα τρία είδη χρηστών είναι τα εξής:

Εικόνα
Είδη

33



- **Απλός επισκέπτης** : Σε αυτό το επίπεδο πρόσβασης, επιτρέπεται στο χρήστη η πλοήγηση στην εφαρμογή αναζητώντας κάποιο προορισμό για διακοπές. Ένας απλός επισκέπτης της εφαρμογής, έχει πρόσβαση σε όλες τις λειτουργίες της εφαρμογής εκτός του περιβάλλοντος διαχείρισης. Για να είναι πιο ευέλικτη η εφαρμογή για τον χρήστη του δίνεται η δυνατότητα να κάνει μία γρήγορη αναζήτηση με κάποια συγκεκριμένα κριτήρια που έχει ορίσει ο σχεδιαστής της βάσης δεδομένων σημαντικά για την επιλογή της διαθεσιμότητας εκδρομών. Τα κριτήρια αυτά είναι η χώρα ενδιαφέροντος, η κατηγορία εκδρομής που τον ενδιαφέρει και η χρονική περίοδος κατά την οποία θέλει να πραγματοποιήσει την εκδρομή.

Γρήγορη αναζήτηση χρήστη:

The screenshot shows a search form with a green background. The title is 'Γρήγορη αναζήτηση'. There are four input fields: 'Χώρες:' with a dropdown menu showing 'Επιλέξτε χώρα', 'Κατηγορίες:' with a dropdown menu showing 'Επιλέξτε κατηγορία', 'Ημερομηνία Αναχώρησης:' with a text input field, and 'Ημερομηνία Επιστροφής:' with a text input field. At the bottom right, there is a button labeled 'Αναζήτηση'.

Εικόνα 34 Γρήγορη αναζήτηση χρήστη

Έπειτα υπάρχουν και κάποια άλλα κριτήρια τα οποία ενδεχομένως να ενδιαφέρουν το χρήστη όπως για παράδειγμα η τιμή κάποιας συγκεκριμένης εκδρομής, στα οποία μπορεί να έχει πρόσβαση μέσω απλής αναζήτησης, δηλαδή επιλέγοντας κάποια από τις επιλογές του μενού



Εικόνα 35 Αρχική σελίδα εφαρμογής

Στη παραπάνω εικόνα βλέπουμε την αρχική σελίδα της εφαρμογής, όπου εμφανίζονται οι τρεις πρώτες προσφορές. Αν επιλέξει κάποια από αυτές τις προσφορές, εμφανίζεται μία νέα σελίδα με λεπτομέρειες της προσφοράς αυτής. Στη σελίδα αυτή εκτός από τις λεπτομέρειες της προσφοράς, υπάρχει και κάποια φόρμα την οποία μπορεί να συμπληρώσει ο χρήστης και η οποία αποστέλλεται στο αντίστοιχο τουριστικό γραφείο στο οποίο ανήκει η συγκεκριμένη προσφορά.

Διακοπές στο Μαρόκο

Το Μαρόκο είναι ένας από τους πιο δημοφιλείς τουριστικούς προορισμούς της Αφρικής και βρίσκεται στην βορειοδυτική άκρη της ηπείρου. Δεν υπάρχει τίποτα που μπορεί να ζητήσει κανείς και να μην το βρει στο Μαρόκο: πανέμορφες αμμώδεις παραλίες, φανταστικές οροσειρές, ερήμους και οάσεις, παραδοσιακά χωριουδάκια, Ρωμαϊκά μνημεία, καταπράσινες κοιλάδες και ορεινές περιοχές, πανέμορφες αρχαίες και πολύχρωμες πόλεις και για τους λάτρεις του καλού φαγητού, υπέροχη Μαροκινή κουζίνα.

Λεπτομέρειες της προσφοράς

Κατηγορία: Ατομικά ταξίδια
Ημερομηνία Αναχώρησης: 01/01/1970
Ημερομηνία Επιστροφής: 01/01/1970
Τιμή: 800\$
Τουριστικό γραφείο: Travel agency

Φωτογραφία



Φόρμα ενδιαφέροντος για την συγκεκριμένη προσφορά

Αν ενδιαφέρεστε για την προσφορά Διακοπές στην Τυνησία συμπληρώστε την παρακάτω φόρμα ενδιαφέροντος και σύντομα ο οργανωτής της εκδρομής θα επικοινωνήσει μαζί σας

Όνομα (*)
Email (*)
Τηλέφωνο
Χώρα
Σχόλια

Χάρτης




Παρακαλώ συμπληρώστε τα υποχρεωτικά πεδία(*), προκειμένου να αποσταλεί η φόρμα σας

Εικόνα 36 Σελίδα εφαρμογής με λεπτομέρειες προσφοράς

Αν ο χρήστης επιλέξει από το μενού την επικοινωνία, του δίνονται δύο δυνατότητες, ή να επικοινωνήσει με το διαχειριστή του συστήματος ή να επικοινωνήσει με κάποιο τουριστικό γραφείο της επιλογής του.

Επικοινωνία με το διαχειριστή του συστήματος

Επικοινωνία με το τουριστικό γραφείο


 Για οποιαδήποτε πληροφορία σχετικά με τη χρήση της ιστοσελίδας είμαστε στη διάθεσή σας. Για να εκδηλώσετε το ενδιαφέρον σας για οποιαδήποτε από τις εκδρομές που προβάλλονται μεταβείτε στο παραπάνω link, για να επικοινωνήσετε με το επιθυμητό τουριστικό γραφείο.

Όνομα: (*)
Email: (*)
Τηλέφωνο:
Χώρα:
Θέμα μηνύματος:
Εισαγωγή μηνύματος:

Παρακαλώ κρατήστε ένα αντίγραφο αυτού του μηνύματος στη διεύθυνσή σας.

Εικόνα 37 Φόρμα επικοινωνίας με διαχειριστή

Επικοινωνία με το τουριστικό γραφείο

 Παρακαλούμε αναγράψτε στο θέμα του μηνύματος το τίτλο της εκδρομής που σας ενδιαφέρει, και σύντομα ο οργανωτής της εκδρομής θα επικοινωνήσει μαζί σας.

Επιλέξτε τουριστικό γραφείο:

Όνομα:

Email:

Θέμα μηνύματος:


Εισαγωγή μηνύματος:

Τίτλος Ενδιαφέρουσα προσφοράς:

Παρακαλώ κρατήστε ένα αντίγραφο αυτού του μηνύματος στη διεύθυνσή σας.

Εικόνα 38 Φόρμα επικοινωνίας με τουριστό γραφείο

- **Διαχειριστής:** Σε ένα διαχειριστικό περιβάλλον ο διαχειριστής έχει κάθε δικαίωμα επεξεργασίας του περιεχομένου. Αρχικά ο διαχειριστής εισάγει το όνομα και τον κωδικό του και αν τα στοιχεία αυτά είναι έγκυρα του επιτρέπεται η είσοδος στο διαχειριστικό περιβάλλον και η επεξεργασία όλων των δεδομένων της βάσης. Διαφορετικά του εμφανίζει το ίδιο παράθυρο για να ξανά πληκτρολογήσει τα σωστά στοιχεία έτσι ώστε να του επιτραπεί η πρόσβαση στο διαχειριστικό περιβάλλον.

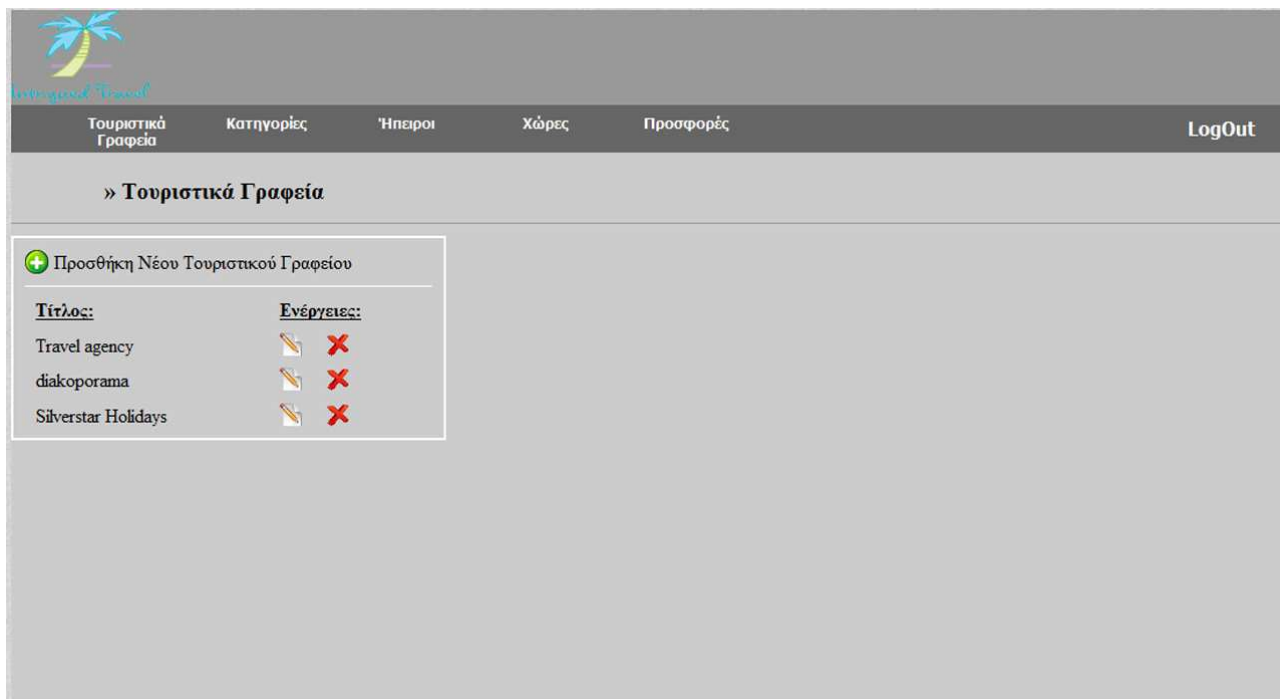

Island Travel
Log In

Χρήστης

Κωδικός

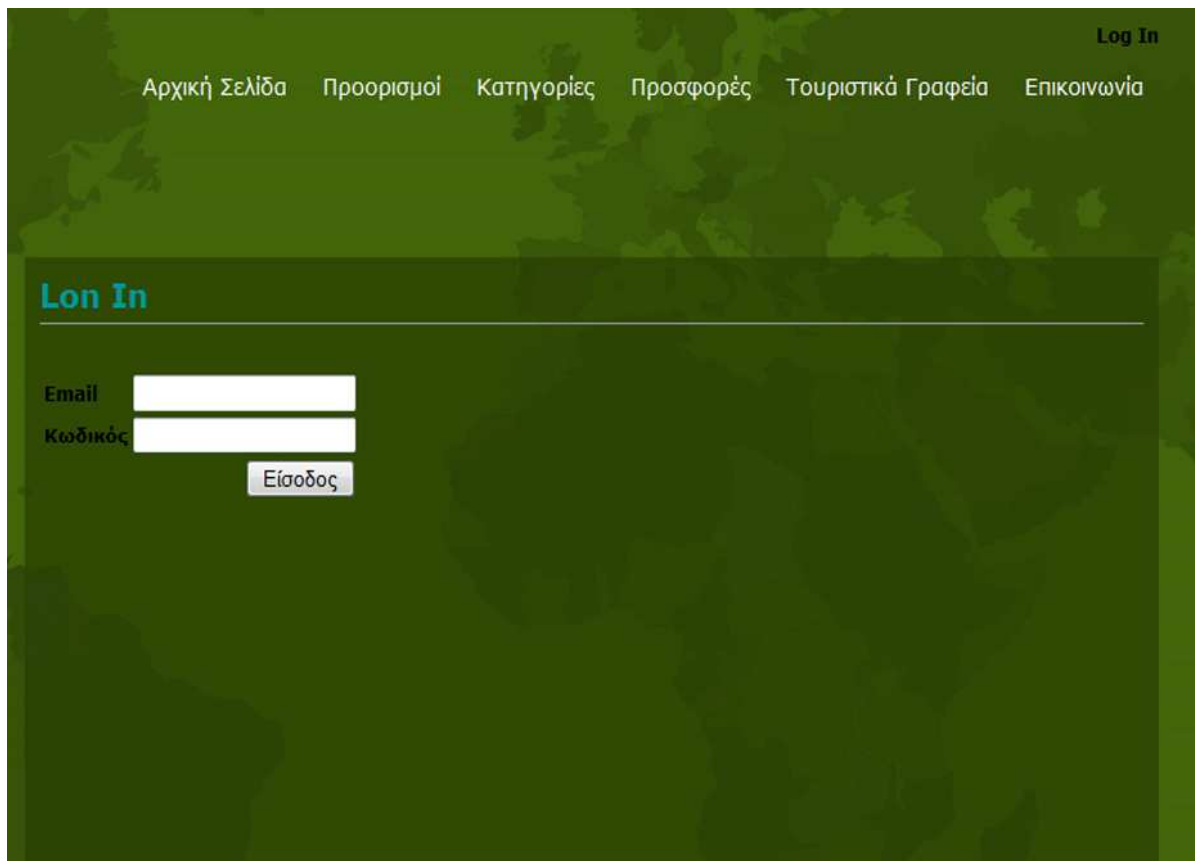
Εικόνα 39 Log_in διαχειριστή

Εφόσον επιτραπεί η πρόσβαση στο χρήστη εμφανίζεται το παρακάτω διαχειριστικό περιβάλλον στο οποίο έχει τη δυνατότητα να προσθαφαιρέσει χρήστες, καθώς επίσης να επεξεργαστεί τα στοιχεία των προσφορές όλων των τουριστικών γραφείων.



Εικόνα 40 Διαχειριστικό περιβάλλον

- **Τουριστικά γραφεία:** Σε ένα διαχειριστικό περιβάλλον τα τουριστικά γραφεία έχουν δικαίωμα στην προσθαφαίρεση και επεξεργασία μόνο των προσφορών τους, διασφαλίζοντας έτσι την ακεραιότητα των προσφορών του κάθε τουριστικού γραφείου. Αρχικά εμφανίζεται ένα παράθυρο για να εισάγει το τουριστικό γραφείο το όνομα και τον κωδικό του. Γίνεται έλεγχος με τα στοιχεία της βάσης αν το όνομα και ο κωδικός αυτός αντιστοιχούν σε κάποιο συγκεκριμένο τουριστικό γραφείο. Αν αντιστοιχούν τότε επιτρέπεται η είσοδος στο διαχειριστικό περιβάλλον του τουριστικού γραφείου, όπου και εμφανίζονται οι προσφορές μόνο του συγκεκριμένου τουριστικού γραφείου που έκανε login, τις οποίες έχει τη δυνατότητα να προσθαφαιρέσει ή να επεξεργαστεί.



Εικόνα 41 Log_in τουριστικού γραφείου

Αφού πληκτρολογήσει το email και τον κωδικό που αντιστοιχούν σε κάποιο από τα υπάρχοντα εγγεγραμμένα τουριστικά γραφεία της βάσης, εμφανίζεται το εξής διαχειριστικό περιβάλλον.



Εικόνα 42 Διαχειριστικό περιβάλλον εγγεγραμμένων τουριστικών γραφείων

10 Περιγραφή και επεξήγηση αναλυτικών κομματιών κώδικα

Εισαγωγή

Σε αυτή την ενότητα παρατίθενται τα βασικά βήματα για τη δημιουργία ενός δυναμικού ιστοχώρου, από τον οποίο ο τελικός χρήστης μπορεί να πληροφορηθεί σχετικά με το προσωπικό μίας εταιρείας.

10.1 Δημιουργία των πινάκων της βάσης

Δομή του πίνακα `category`:

```
CREATE TABLE `travel_agency`.`category` (  
  
/* ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΠΙΝΑΚΑ category της ΒΑΣΗΣ travel_agency ΜΕ ΤΑ ΕΞΗΣ ΠΕΔΙΑ*/  
  
`id` INT NOT NULL AUTO_INCREMENT ,  
  
//το πεδίο id είναι τύπου ακεραίου και είναι NOT NULL AUTO_INCREMENT(*)  
  
`name` TEXT CHARACTER NOT NULL `image` TEXT CHARACTER NOT NULL ,  
  
//το πεδίο name είναι τύπου κειμένου και είναι NOT NULL(*)  
  
PRIMARY KEY ( `id` ))  
  
//το id είναι το πρωτεύον κλειδί(*) του πίνακα  
  
ENGINE = MYISAM CHARACTER SET utf8 COLLATE utf8_general_ci ;  
  
// Επιλεγμένη κωδικοποίηση utf8 COLLATE utf8_general_ci,
```

Κώδικας 1 Δομή του πίνακα `category`

(*) Το `AUTO_INCREMENT` είναι μία ειδική λειτουργία της MySQL που μπορούμε να χρησιμοποιήσουμε σε ακέραια πεδία του πίνακα. Σημαίνει ότι αν αφήσουμε αυτό το πεδίο κενό όταν εισάγουμε εγγραφές στο πίνακα η MySQL θα δημιουργήσει αυτόματα ένα μοναδικό αναγνωριστικό.

(*) το `NOT NULL` σημαίνει ότι όλα τα πεδία του πίνακα πρέπει να έχουν μία τιμή.

(*) Το `PRIMARY KEY` μετά από ένα πεδίο καθορίζει ότι το πεδίο αυτό είναι πρωτεύον κλειδί του πίνακα. Τα στοιχεία αυτού του πεδίου θα πρέπει να είναι μοναδικά

Δομή του πίνακα continent:

```
CREATE TABLE `travel_agency`.`continent` (  
  
/* ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΠΙΝΑΚΑ continent της ΒΑΣΗΣ travel_agency ΜΕ ΤΑ ΕΞΗΣ ΠΕΔΙΑ */  
  
`id` INT NOT NULL AUTO_INCREMENT ,  
  
//το πεδίο id είναι τύπου ακεραίου και είναι NOT NULL AUTO_INCREMENT (*)  
  
`name` TEXT CHARACTER NOT NULL ,  
  
//το πεδίο name είναι τύπου κειμένου και είναι NOT NULL(*)  
  
PRIMARY KEY (`id`)) //το id είναι το πρωτεύον κλειδί(*) του πίνακα  
  
ENGINE = MYISAM CHARACTER SET utf8 COLLATE utf8_general_ci ;  
  
// Επιλεγμένη κωδικοποίηση utf8 COLLATE utf8_general_ci
```

Κώδικας 2 Δομή του πίνακα continent

Δομή του πίνακα country:

```
CREATE TABLE `travel_agency`.`country` (  
  
/* ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΠΙΝΑΚΑ country της ΒΑΣΗΣ travel_agency ΜΕ ΤΑ ΕΞΗΣ ΠΕΔΙΑ */  
  
`id` INT(11) NOT NULL AUTO_INCREMENT ,  
  
//το πεδίο id είναι τύπου ακεραίου και είναι NOT NULL AUTO_INCREMENT (*)  
  
`name` TEXT NOT NULL , `history` TEXT NOT NULL ,//το πεδίο name είναι τύπου κειμένου και  
είναι NOT NULL(*)  
  
`continent_id` INT NOT NULL , //το πεδίο continent_id είναι τύπου ακεραίου και είναι NOT  
NULL(*)  
  
`image` TEXT NOT NULL ,//το πεδίο image είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
`history_img` TEXT NOT NULL ,//το πεδίο history_img είναι τύπου ακεραίου και είναι NOT  
NULL(*)  
  
PRIMARY KEY (`id`)) //το id είναι το πρωτεύον κλειδί(*) του πίνακα  
  
ENGINE = MYISAM CHARACTER SET utf8 COLLATE utf8_general_ci;  
  
// Επιλεγμένη κωδικοποίηση utf8 COLLATE utf8_general_ci
```

Κώδικας 3 Δομή του πίνακα country

Δομή του πίνακα offers:

```
CREATE TABLE `travel_agency`.`offers` (  
  
  /* ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΠΙΝΑΚΑ offers της ΒΑΣΗΣ travel_agency ΜΕ ΤΑ ΕΞΗΣ ΠΕΔΙΑ */  
  
  `id` INT(11) NOT NULL AUTO_INCREMENT ,  
  
  //το πεδίο id είναι τύπου ακεραίου και είναι NOT NULL AUTO_INCREMENT (*)  
  
  `title` TEXT NOT NULL ,  
  
  //το πεδίο title είναι τύπου κειμένου και είναι NOT NULL(*)  
  
  `description` TEXT NOT NULL ,  
  
  //το πεδίο description είναι τύπου κειμένου και είναι NOT NULL(*)  
  
  `country_id` INT NOT NULL ,  
  
  //το πεδίο country_id είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
  `category_id` INT NOT NULL ,  
  
  //το πεδίο category_id είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
  `user_id` INT NOT NULL ,  
  
  //το πεδίο users_id είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
  `date_from` TEXT NOT NULL ,  
  
  //το πεδίο date_from είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
  `date_to` TEXT NOT NULL ,  
  
  //το πεδίο date_to είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
  `price` TEXT NOT NULL ,//το πεδίο price είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
  `image` TEXT NOT NULL ,//το πεδίο image είναι τύπου κειμένου και είναι NOT NULL(*)  
  
  `map` TEXT NOT NULL ,//το πεδίο map είναι τύπου κειμένου και είναι NOT NULL(*)  
  
  PRIMARY KEY ( `id` ) //το id είναι το πρωτεύον κλειδί(*) του πίνακα  
  
  ) ENGINE = MYISAM CHARACTER SET utf8 COLLATE utf8_general_ci;  
  
  // Επιλεγμένη κωδικοποίηση utf8 COLLATE utf8_general_ci
```

Κώδικας 4 Δομή του πίνακα offers

Δομή του πίνακα users:

```
CREATE TABLE `travel_agency`.`users` (  
  
/* ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ ΠΙΝΑΚΑ users της ΒΑΣΗΣ travel_agency ΜΕ ΤΑ ΕΞΗΣ ΠΕΔΙΑ */  
  
`id` INT(11) NOT NULL AUTO_INCREMENT ,  
  
//το πεδίο id είναι τύπου ακεραίου και είναι NOT NULL AUTO_INCREMENT (*)  
  
`name` TEXT NOT NULL , //το πεδίο name είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`address` TEXT NOT NULL , //το πεδίο address είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`fax` TEXT NOT NULL , //το πεδίο fax είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`site` TEXT NOT NULL , //το πεδίο site είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`password` TEXT NOT NULL , //το πεδίο password είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`username` TEXT NOT NULL , //το πεδίο username είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`email` TEXT NOT NULL , //το πεδίο email είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`photo` TEXT NOT NULL , //το πεδίο photo είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`image` TEXT NOT NULL , //το πεδίο image είναι τύπου κειμένου και είναι NOT NULL(*)  
  
`date` INT NOT NULL , //το πεδίο date είναι τύπου ακεραίου και είναι NOT NULL(*)  
  
PRIMARY KEY (`id`)) //το id είναι το πρωτεύον κλειδί(*) του πίνακα  
  
ENGINE = MYISAM CHARACTER SET utf8 COLLATE utf8_general_ci;  
  
// Επιλεγμένη κωδικοποίηση utf8 COLLATE utf8_general_ci
```

Κώδικας 5 Δομή του πίνακα users

Εισαγωγή

Με τη χρήση της PHP μπορούμε να συνδεθούμε σε έναν οποιοδήποτε MySQL Server στον οποίο έχουμε λογαριασμό, να πάρουμε δεδομένα από πίνακες που ανήκουν σε ήδη υπάρχουσες βάσεις δεδομένων, να εισάγουμε δεδομένα στους πίνακες, να ανανεώσουμε κάποια υπάρχοντα δεδομένα, να φτιάξουμε νέες βάσεις και νέους πίνακες και γενικά να κάνουμε οτιδήποτε γίνεται με μια MySQL βάση δεδομένων. Επομένως, έχοντας την κατάλληλη λειτουργικότητα σε μία ιστοσελίδα είναι δυνατή η πλήρης διαχείριση μιας MySQL

Βασικότερες Συναρτήσεις της PHP για τη MySQL

Στη παράγραφο αυτή θα δούμε τις βασικές συναρτήσεις της PHP με τις οποίες μπορούμε να συνδεθούμε με τη MySQL, να αποσυνδεθούμε από αυτήν, να επιλέξουμε τη βάση που θα ανοίξει για να αρχίσουμε την επεξεργασία της και τέλος να υποβάλουμε ερωτήσεις προς τη βάση

10.2 Σύνδεση στη βάση δεδομένων

Η σύνδεση γίνεται μέσω της συνάρτησης `mysql_connect()` της PHP δίνοντας τα σχετικά ορίσματα, στο αρχείο `config.php`. Με τη χρήση της συνάρτησης `mysql_connect` μπορούμε να συνδεθούμε σε μια βάση της MySQL. Η συνάρτηση αυτή δέχεται 3 ορίσματα:

- Το όνομα του διακομιστή (server) στον οποίο είναι εγκατεστημένη η MySQL ο οποίος είναι γνωστός με την έκφραση `hostname`,
- το `login` του χρήστη, που έχει δημιουργηθεί για την πρόσβασή του στη βάση,
- το αντίστοιχο `Password` του χρήστη.

Η συνάρτηση επιστρέφει έναν ακέραιο αριθμό (link identifier) ο οποίος αντιστοιχεί στην συγκεκριμένη σύνδεση, αν αυτή βέβαια είναι επιτυχής. Τον αριθμό αυτό (ο οποίος είναι μοναδικός και αποθηκευμένος σε μια μεταβλητή) τον χρησιμοποιούμε όταν θέλουμε να αναφερθούμε στη σύνδεση με τη MySQL. Για παράδειγμα:

```
$dbh = mysql_connect($sql_host,$sql_username,$sql_password);
```

Η επιλογή της βάσης γίνεται μέσω της συνάρτησης `mysql_select_dbh()`: Με αυτή τη συνάρτηση επιλέγουμε κάποια ήδη δημιουργηθείσα βάση με την οποία θέλουμε να δουλέψουμε. Δέχεται ως όρισμα το όνομα της βάσης και η συνάρτητη επιστρέφει μια λογική τιμή (true ή false) ανάλογα αν πέτυχε ή όχι η επιλογή της βάσης. Για παράδειγμα:

```
mysql_select_db($sql_database);
```

Με τη χρήση της συνάρτησης `mysql_query()` , μπορούν να εκτελεστούν ερωτήσεις προς τη MySQL. Δέχεται ως όρισμα:

- σε μορφή κειμένου (string) την ερώτηση και επιστρέφει μια μη μηδενική τιμή (επιτυχία), ή την τιμή `false` (αποτυχία)

```

<? php

/*      στοιχεία της βάσης για σύνδεση      */

$sql_host = 'localhost';
// Το όνομα του διακομιστή (server) στον οποίο είναι εγκατεστημένη η MySQL ο οποίος είναι
γνωστός με την έκφραση hostname

$sql_username = 'root';
//όνομα χρήστη, το οποίο έχει δημιουργηθεί για την πρόσβαση του στη βάση

$sql_password = "";
//κωδικός χρήστη, με τον οποίο ο χρήστης μπορεί να έχει πρόσβαση στη βάση

$sql_database = 'travel_agency';
//όνομα της βάσης δεδομένων

/*      Σύνδεση στην βάση      */

$dbh = mysql_connect($sql_host,$sql_username,$sql_password);
// σύνδεση στη mysql βάση δεδομένων στο localhost server, με τις παραπάνω μεταβλητές

mysql_select_db($sql_database);
//επιλογή της βάσης δεδομένων

mysql_query ("SET NAMES 'utf8'");
//κωδικοποίηση της βάσης δεδομένων

?>

```

Κώδικας 6 Σύνδεση στην βάση δεδομένων (php)

Με τον τρόπο αυτό δηλώνεται η μεταβλητή \$dbh και αποθηκεύεται σε αυτή η σύνδεση (pointer to database) με τη βάση δεδομένων MySQL, στην οποία περιέχονται το όνομα του server("localhost"), το username του χρήστη της βάσης δεδομένων με δικαιώματα τουλάχιστον ανάγνωσης πληροφορίας από αυτή (εδώ χρησιμοποιείται ο χρήστης "root", δηλαδή ο διαχειριστής της βάσης δεδομένων MySQL που κατασκευάζεται με την εγκατάσταση) και το αντίστοιχο συνθηματικό του ("", δηλαδή το κενό).

Εάν θα θέλαμε να δώσουμε περιφραστικά τη σημασία της εντολής αυτής στην αγγλική θα γράφαμε το ακόλουθο ορισμό: «*connect to MySQL database in localhost server with the username root and password ""*». Στην περίπτωση όπου το συνθηματικό του χρήστη root έχει αλλάξει ή στην ενδεχόμενη περίπτωση όπου χρησιμοποιείται χρήστης με λιγότερα δικαιώματα χρήσης, τότε τα δεδομένα αυτά αλλάζουν αναλόγως. Με αυτό τον τρόπο η μεταβλητή \$dbh αποτελεί ένα δείκτη (pointer) στη βάση δεδομένων του server localhost (τοπικό μηχάνημα εγκατάστασης).

Για την πρόσβαση στη βάση λοιπόν δημιουργήσαμε ένα χρήστη (τον root) ο οποίος μπορεί να έχει πρόσβαση μόνο από το τοπικό μηχάνημα στο οποίο υπάρχει και ο MySQL Server(localhost).Ο χρήστης αυτός έχει κάθε δικαίωμα επεξεργασίας του περιεχομένου της βάσης(ALL PRIVILEGES), καθώς ο ίδιος μπορεί ακόμα και να εγγράψει ή να διαγράψει κάποιο από τα τουριστικά γραφεία που διαχειρίζονται και ανανεώνουν περιστασιακά τη βάση με τις προσφορές τους, όπως και να επεξεργαστεί τα στοιχεία προσφορών όλων των τουριστικών γραφείων στα οποία έχει πρόσβαση. Τόσο η επεξεργασία όσο και η δημιουργία της βάσης που προαναφέρθηκε γίνεται μέσω του προγράμματος της MySQL, το οποίο είναι το phpMyAdmin.

Με την ακόλουθη αλληλουχία εντολών PHP, (σε περίπτωση επιτυχίας εκτέλεσης) θα συγκεντρωθεί στη μεταβλητή \$result το περιεχόμενων όλων των εγγραφών του πίνακα users της βάσης δεδομένων travel_agency:

```
$dbh = mysql_connect("localhost", "root", "");  
  
// σύνδεση στη mysql βάση δεδομένων στο localhost server του χρήστη, με όνομα root και κωδικό ''  
mysql_select_db("travel_agency",$dbh);  
  
//επιλογή δημιουργείσας βάσης travel_agency, με την οποία θέλουμε να δουλέψουμε  
$result = mysql_query("SELECT * FROM users",$dbh);  
  
//εκτελείται ένα sql ερώτημα προς την MySQL,μέσω του οποίου επιλέγει τις εγγραφές του πίνακα  
users της MySQL βάσης, με την οποία έχει συνδεθεί ο χρήστης και τις συγκεντρώνει στη μεταβλητή  
$result
```

Κώδικας 7 Επιλογή εγγραφών από τη βάση

10.3 Εισαγωγή ενδεικτικών εγγραφών σε πίνακα

Κατόπιν εισάγουμε εγγραφές στον πίνακα users, δίνοντας τιμές που συμμορφώνονται με τη δομή του:

```
INSERT INTO `travel_agency`.`users` (`id`, `name`, `address`, `fax`, `site`, `password`, `username`, `email`, `photo`, `image`, `date`) VALUES (NULL, 'Travel agency', kati 123', '3423432', 'travelagency.gr', '123', '', 'lala@gmail.com', 'fgdfgd', 'logo_travelAgency.jpg');
```

```
INSERT INTO `travel_agency`.`users` (`id`, `name`, `address`, `fax`, `site`, `password`, `username`, `email`, `photo`, `image`, `date`) VALUES (NULL, 'diakoporama', 'kati 12345', '3423432', 'diakoporama.gr', '654321', '', 'diakoporama16@gmail.com', 'fgdfgd', 'logo_diakoporama.jpg');
```


//Εισαγωγή εγγραφών σε καθένα από τα πεδία του πίνακα users της βάσης travel_agency

Κώδικας 8 Εισαγωγή εγγραφών στο πίνακα users

Για να επαληθευτεί η ορθότητα της εισαγωγής βάση των παραπάνω εντολών, μπορούμε να χρησιμοποιήσουμε ένα SQL query προς τη βάση, με βάση το οποίο θα λάβουμε ως αποτέλεσμα όλα τα στοιχεία όλων των εγγραφών που περιέχονται σε συγκεκριμένο πίνακα αυτής:

```
mysql>SELECT * FROM `users`;
```

// sql ερώτημα προς τη βάση δεδομενων travel_agency, βάση του οποίου επιλέγονται όλα τα στοιχεία όλων των εγγραφών από τον πίνακα users

	id	name	address	fax	site	password	username	email	photo	image
 	4	Travel agency	kati 123	3423432	travelagency.gr	123		lala@gmail.com	fgdfgd	logo_travelAgency.jpg
 	5	diakoporama	kati 12345	3423432	diakoporama.gr	654321		diakoporama16@gmail.com	fgdfgd	logo_diakoporama.jpg

Εικόνα 43 Εγγραφές του πίνακα users

10.4 Εμφάνιση περιεχομένων εγγραφών

Με δεδομένη την ύπαρξη ενός πίνακα ο οποίος περιέχει κάποια στοιχεία όπως στη συγκεκριμένη περίπτωση ο πίνακας users, με τη χρήση του παρακάτω κώδικα PHP στα πλαίσια του οποίου

αποστέλλεται προς εκτέλεση στη βάση δεδομένων ένα συγκεκριμένο SQL query, λαμβάνονται μέσω του browser τα αποτελέσματα του query αυτού :

Αρχείο users.php

```
<? $sql="SELECT *FROM `users`";  
  
// sql ερώτημα προς τη βάση δεδομενων travel_agency, βάση του οποίου επιλέγονται όλα τα στοιχεία  
όλων των εγγραφών από τον πίνακα users  
  
$result=mysql_query($sql);  
  
//εκτελείται ένα sql ερώτημα , μέσω του οποίου επιλέγονται οι εγγραφές του πίνακα users της MySQL  
βάσης, με την οποία έχει συνδεθεί ο χρήστης και συγκεντρώνονται στη μεταβλητή $result  
  
while($row=mysql_fetch_array($result)){  
  
//κάθε μία από τις εγγραφές του πίνακα users αποθηκεύονται σε ένα πίνακα  
  
?>  
  
<div class="users_item">  
  
// για κάθε μία από τις εγγραφές φορτώνεται και εμφανίζεται μία εικόνα (το λογότυπο του κάθε  
τουριστικού γραφείου)  
  
<a href="http://<? echo $row['site']; ?>" target="_blank"><? echo $row['name']; ?></a></div>  
  
<? } ?>  
  
// για κάθε μία από της εγγραφές εμφανίζεται το όνομα του τουριστικού γραφείου από τη βάση (με τη  
χρήση της php), όπου πατώντας το σε παραπέμπει στο site του αντίστοιχου τουριστικού
```

Κώδικας 9 Εμφάνιση εγγεγραμμένων τουριστικών γραφείων με τη χρήση της sql και php



Εικόνα 44 Εγγεγραμμένα τουριστικά γραφεία

Αφού έχει γίνει η σύνδεση του χρήστη που δημιουργήσαμε με τη βάση είναι δυνατή η ανάκτηση των δεδομένων από τη βάση με τη χρήση sql ερωτημάτων .

Εισάγεται λοιπόν το SQL query βάση του οποίου αναμένονται αποτελέσματα από τη συγκεκριμένη βάση και αποθηκεύονται στη μεταβλητή "\$result (" \$result = mysql_query(\$sql);"). Στο συγκεκριμένο παράδειγμα το SQL query "SELECT * FROM users" σημαίνει ότι αναζητούνται όλα τα στοιχεία όλων των εγγραφών του πίνακα users. Η εκτέλεση του συγκεκριμένου query προς τη βάση δεδομένων γίνεται με τη χρήση της συνάρτησης mysql_query() και τα αποτελέσματα αυτής της εκτέλεσης αποθηκεύονται στη μεταβλητή \$result. Σε αυτό το χρονικό σημείο πλέον είναι δυνατή η εμφάνιση των αποτελεσμάτων και για το λόγο αυτό δίνονται οι κατάλληλες εντολές παρουσίασης των αποτελεσμάτων αυτών με τη συνήθη μορφή, όπως στη συγκεκριμένη περίπτωση, πίνακα δεδομένων. Με τη χρήση της συνάρτησης mysql_fetch_array(), εξάγονται μία προς μία όλες οι εγγραφές, τα στοιχεία κάθε μίας των οποίων αποθηκεύονται στη μεταβλητή \$row. Με τη διαδικασία αυτή η μεταβλητή \$row περιέχει όλη τη διαθέσιμη πληροφορία ανά εγγραφή για όλες τις εγγραφές που έχουν εξαχθεί και στην μεταβλητή \$result. Με στόχο τη καλύτερη παρουσίαση των αποτελεσμάτων ενός query, τα στοιχεία-πεδία μίας εγγραφής παρουσιάζονται όπως προαναφέρθηκε σε μορφή πίνακα δεδομένων. Γνωρίζοντας ότι η μεταβλητή \$row λαμβάνει εξ ορισμού μορφή πίνακα με επιμέρους ονόματα πεδίων ίδια με εκείνα που αναμένονται από το SQL query από το οποίο προέρχονται τα αποτελέσματα, είναι δυνατή η παρουσίαση των αποτελεσμάτων με τη χρήση της συνάρτησης echo και του σχετικού πεδίου (όπως η χρήση του "echo \$row["name"];"). Η παρουσίαση των δεδομένων ανά εγγραφή βρίσκεται μέσα σε επαναληπτικό loop while() με σκοπό να συνεχίζεται η παρουσίαση των στοιχείων ανά εγγραφή έως ότου υπάρχουν σχετικές εγγραφές στην \$result.

10.5 Εισαγωγή εγγραφής

Σε περίπτωση που δίνεται κάποιο id, δηλαδή σε περίπτωση που υπάρχει έστω και μία καταχώρηση στο πίνακα users, δίνεται η δυνατότητα επεξεργασίας και διαγραφής κάποιας συγκεκριμένης καταχώρησης (με βάση το id της)

```

<? if($_GET['id']){
// έλεγχος αν υπάρχει το id , τότε τρέχει τον παρακάτω κώδικα
$id=$_GET['id'];
// γίνεται ανάκτηση του id
    $sql="SELECT * FROM `users` where id=$id";
// sql ερώτημα μέσω του οποίου επιλέγονται όλα τα στοιχεία της εγγραφής του πίνακα users, με το
επιλεγμένο id
    $result=mysql_query($sql);
// εκτέλεση του παραπάνω sql ερωτήματος , μέσω του οποίου επιλέγονται όλα τα στοιχεία της
εγγραφής του πίνακα users και συγκεντρώνονται στη μεταβλητή $result
    $row=mysql_fetch_array($result);
// όλα τα στοιχεία της εγγραφής του πίνακα users αποθηκεύονται σε ένα πίνακα
?>
<form method="post" action="actions.php?action=editUsers&id=<?=$row['id']?>"
enctype="multipart/form-data">
<!-- όλα τα στοιχεία της εγγραφής του πίνακα users αποστέλλονται με μέθοδο post, στο αρχείο
actions.php όπου εκεί γίνονται οι διεργασίες, όπως η επεξεργασία των στοιχείων της εγγραφής με
βάση το id της το οποίο και εμφανίζουμε, για να φαίνεται ποια εγγραφή επεξεργαζόμαστε. -->
<!--Εμφάνιση εγγραφών του πίνακα users-->
<table border="0" cellpadding="10" cellspacing="0" align="center"
    <tr>
        <td>Όνομα</td>
        <td><input type="text" name="name" value="<?=$row['name']?>" /></td>
    </tr>
<!--Ομοίως εμφανίζονται και υπόλοιπες εγγραφές του πίνακα users-->
</table>
</form>

```

Κώδικας 10 Φόρμα επεξεργασίας τουριστικού γραφείου στο αρχείο add_editUsers

Στο αρχείο actions.php γίνονται οι διεργασίες, όπως η προσθήκη κάποιας εγγραφής, η οποία πραγματοποιείται όταν δεν δίνεται κάποιο id

```
<? }else{ ?> //Αλλιώς(σε περίπτωση που δεν υπάρχει κάποιο id)

<form method="post" action="actions.php?action=addUsers">//γίνεται αποστολή με μέθοδο post
στο αρχείο actions.php(όπου γίνονται οι διεργασίες),εκεί που το actions ισούται με addUsers

/* Εμφάνιση της φόρμας όπου ο χρήστης θα πληκτρολογήσει τα στοιχεία του */

<table border="0" cellspacing="0" cellpadding="10" align="center"><p
align="center"><b>Προσθήκη Τουριστικού γραφείου</b></p>

<tr>

<td>Όνομα</td>

<td><input type="text" name="name" /></td>

</tr>

<!--όμοια σχηματίζονται και τα υπόλοιπα πεδία της φόρμας-->

<tr>

<td colspan="2"><input type="submit" name="submit" value="Αποθήκευση" /></td>

</tr></table>

</form>

<? } ?>
```

Κώδικας 11 Φόρμα προσθήκης τουριστικού γραφείου στο αρχείο add_editUsers

Αρχείο actions.php

```
<? if($_GET['action']=='addUsers'){  
//Έλεγχος αν το actions ισούται με το addUsers, τότε εκτελούνται τα παρακάτω  
/* γίνεται ανάκτηση των των στοιχείων αφού συμπληρωθούν από το χρήστη */  
$name=$_POST['name']; // γίνεται ανάκτηση του ονόματος του τουριστικού γραφείου  
$address=$_POST['address']; // γίνεται ανάκτηση της διεύθυνσης του τουριστικού γραφείου  
$fax=$_POST['fax']; // γίνεται ανάκτηση του fax του τουριστικού γραφείου  
$site=$_POST['site']; // γίνεται ανάκτηση του site του τουριστικού γραφείου  
$password=$_POST['password']; // γίνεται ανάκτηση του password του τουριστικού γραφείου  
$username=$_POST['username']; // γίνεται ανάκτηση του username του τουριστικού γραφείου  
$email=$_POST['email']; // γίνεται ανάκτηση του email του τουριστικού γραφείου  
$photo=$_POST['photo']; // γίνεται ανάκτηση της photo του τουριστικού γραφείου  
$sql= "INSERT INTO `users` (id , name , address , fax , site , password , username , email ,  
photo) VALUES (null, ".$name.", ". $address.", ".$fax.", ".$site.", ".$password.",  
".$username.", ".$email.", ".$photo.)";  
//sql ερώτημα το μέσω του οποίου εισάγονται στα πεδία του πίνακα τα στοιχεία του χρήστη  
$error=mysql_query($sql);  
//εκτέλεση του παραπάνω sql ερωτήματος  
redirect('admin.php?page=users');  
//γίνεται ανακατεύθυνση στο αρχείο admin.php στη σελίδα των χρηστών, όπου εμφανίζονται τα  
εγγεγραμμένα τουριστικά γραφεία  
} ?>
```

Κώδικας 12 Προσθήκη τουριστικού γραφείου στο αρχείο actions

10.6 Διαγραφή κάποιας συγκεκριμένης εγγραφής

Στο αρχείο actions.php όπου γίνονται όλες οι διεργασίες, γίνεται και η διαγραφή με βάση το id που αποστέλλεται από το αρχείο users.php, εφόσον δίνεται κάποιο id (δηλαδή εφόσον υπάρχει κάποια

εγγραφή στο πίνακα users

Στο αρχείο actions.php όπου γίνονται όλες οι διεργασίες, γίνεται και η διαγραφή με βάση το id που αποστέλλεται από το αρχείο users.php ,εφόσον δίνεται κάποιο id(δηλαδή εφόσον υπάρχει κάποια εγγραφή στο πίνακα users)

Αρχείο actions.php

```
<? if($_GET['action']=='deleteUsers'){  
  
    // έλεγχος αν το action ισούται με deleteUsers , τότε τρέχει τον παρακάτω κώδικα  
  
    $id=$_GET['id'];  
  
    // γίνεται ανάκτηση του id  
  
    $sql=" DELETE FROM `travel_agency`.`users` WHERE `users`.`id` = '$id' ";  
  
    // sql ερώτημα προς τη βάση, μέσω του οποίου διαγράφουμε μία συγκεκριμένη εγγραφή από τον  
    πίνακα users(της βάσης δεδομένων travel_agency) με βάση το id της εγγραφής αυτής  
  
    $error=mysql_query($sql);  
  
    // Για να εκτελέσουμε το παραπάνω sql ερώτημα χρησιμοποιούμε την εντολή mysql_query().  
  
    redirect('admin.php?page=users');  
  
    // γίνεται ανακατεύθυνση στο αρχείο admin.php, στη σελίδα users(όπου εμφανίζεται στο browser όλοι  
    οι υπάρχοντες χρήστες, μετά τη διαγραφή της εγγραφής)  
  
    }  
  
?>
```

Κώδικας 13 Διαγραφή επιλεγμένου τουριστικού γραφείου στο αρχείο actions

Με τον ίδιο τρόπο γίνεται και η ενημέρωση κάποιας συγκεκριμένης εγγραφής του πίνακα(δηλαδή με βάση το id της εγγραφής που θέλουμε να τροποποιήσουμε, το οποίο αποστέλλεται από το users.php)

Αρχείο actions.php

```
<? if($_GET['action']=='editUsers'){  
  
    // έλεγχος αν το action ισούται με editUsers , τότε τρέχει τον παρακάτω κώδικα  
  
    $id=$_GET['id']; // γίνεται ανάκτηση του id  
  
    /* γίνεται ανάκτηση των των στοιχείων αφού συμπληρωθούν από το χρήστη */  
  
    $name=$_POST['name']; // γίνεται ανάκτηση του ονόματος του τουριστικού γραφείου  
  
    $address=$_POST['address']; // γίνεται ανάκτηση του διεύθυνσης του τουριστικού γραφείου  
  
    $fax=$_POST['fax']; // γίνεται ανάκτηση του fax του τουριστικού γραφείου  
  
    $site=$_POST['site']; // γίνεται ανάκτηση του site του τουριστικού γραφείου  
  
    $password=$_POST['password']; // γίνεται ανάκτηση του password του τουριστικού γραφείου  
  
    $email=$_POST['email']; // γίνεται ανάκτηση του ονόματος του email τουριστικού γραφείου  
  
    $photo=$_POST['photo']; // γίνεται ανάκτηση της photo του τουριστικού γραφείου  
  
    $sql="UPDATE `travel_agency`.`users` SET `name` = '$name' , `address` = '$address' ,  
  
        `fax` = '$fax' ,  
  
        `site` = '$site' ,  
  
        `password` = '$password' ,  
  
        `email` = '$email' ,  
  
        `photo` = '$photo'  
  
        WHERE `users`.`id` = '$id' "; // γίνεται ενημέρωση των στοιχείων του πίνακα users για  
κάποια συγκεκριμένη εγγραφή, ανάλογα με το id που έχει δοθεί  
  
    $error=mysql_query($sql); // εκτέλεση του παραπάνω sql ερωτήματος  
  
    redirect('admin.php?page=users&id='.$id); // ανακατεύθυνση στο αρχείο admin.php, στη  
σελίδα των χρηστών( για να μπορέσει το τουριστικό να δει την αλλαγή που έκανε στη προσφορά του)  
}  
?>
```

Κώδικας 14 Επεξεργασία επιλεγμένου τουριστικού γραφείου στο αρχείο actions

10.7 Περιγραφή της εφαρμογής για το διαχειριστή

LOG_IN διαχειριστή στην εφαρμογή

Καλούμε τη session_start στην αρχή του script του admin , θα φορτώσει όλες τις εγγεγραμμένες μεταβλητές της συνόδου(το admin_user_id) για να μπορέσει το script να τις χειριστεί

admin.php

```
<?
session_start();
// Το ξεκίνημα ενός session γίνεται με μια κλήση στη συνάρτηση session_start
if(!$_SESSION['admin_user_id'] && $_REQUEST['page']!='log_in'){
//Αν δεν δίνεται admin_user_id
redirect('admin.php?page=log_in');
// ανακατεύθυνση στο αρχείο admin.php, στη σελίδα log_in(όπου βρίσκεται η φόρμα για να
ξαναπληκτρολογήσει ο χρήστης τα στοιχεία του)
}
if($_SESSION['admin_user_id']){
//Αν δίνεται το id του διαχειριστή στο πίνακα SESSION
?>
<a href="actions.php?action=log_out2">LogOut</a>
<!-- ανακατεύθυνση στο αρχείο actions.php στη διεργασία log_out2 (που δίνει την δυνατότητα στο
διαχειριστή να εξέρθει από την εφαρμογή)-->
<? } ?>
```

Κώδικας 15 Έλεγχος ταυτότητας διαχειριστή με έλεγχο συνόδων λειτουργίας στο αρχείο admin

log_in.php

```
<form method="post" action="actions.php?action=log_in2" >
//Αποστολή των στοιχείων της φόρμας με μέθοδο post στο αρχείο actions.php, στη διεργασία
log_in2(όπου εκεί θα γίνει έλεγχος αν τα στοιχεία αυτά είναι έγγυρα)

    <!--πεδία της φόρμας -->
<table>
    <tr>
        <td><b>Χρήστης</b></td>
        <td><input type="text" name="username"/></td>
    </tr>
    <tr>
        <td><b>Κωδικός</b></td>
        <td><input type="password" name="password"/></td>
    </tr>
    <tr>
        <td align="right"><input type="submit" name="submit" value="Είσοδος"/></td>
    </tr>
</table>
</form>
```

Κώδικας 16 Φόρμα εισαγωγής στοιχείων του διαχειριστή, για την εισοδο του στο διαχειριστικό περιβάλλον

«Παραδοσιακός» έλεγχος ταυτότητας χρηστών

Αποθηκεύουμε σε ένα πίνακα μιας ΒΔ τα επιτρεπτά ζεύγη username/passwords.

«Προστατεύουμε» την κεντρική σελίδα της τοποθεσίας (web site) με μια φόρμα για login

Σε περίπτωση που ο χρήστης δώσει σωστά username/password, του επιτρέπεται η πρόσβαση στην κεντρική σελίδα και από εκεί (μέσω links) στις υπόλοιπες σελίδες του site.

Αρχείο actions.php

```
<? if($_GET['action']=='log_in2'){ //έλεγχος αν το action ισούται με log_in2, τότε τρέχει τον
παρακάτω κώδικα

$username = $_REQUEST['username'];

// ανάκτηση του username από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το
input με όνομα username

$password = $_REQUEST['password'];

// ανάκτηση του 'password' από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το
input με όνομα 'password'

if($username=='administrator' && $password=='03081987'){

// Γίνεται έλεγχος αν το όνομα χρήστη και ο κωδικός είναι έγγυρα, δηλαδή είναι ίδια με αυτά που έχει
ορίσει ο διαχειριστής

    $_SESSION['admin_user_id'] = '1';// ενημερώνεται ο πίνακας SESSION με πεδίο
admin_user_id, το οποίο σημαίνει ότι επιτυγχάνεται η είσοδος του χρήστη στην εφαρμογή
}else{

    $_SESSION['admin_user_id'] = "";

}

// Αν δεν είναι έγγυρα τα στοιχεία του χρήστη τότε γίνεται καταχώρηση στο πεδίο admin_user_id του
πίνακα SESSION μηδενική τιμή, δηλαδή το λαμβάνει σαν να μην έχει δοθεί κάποιο admin_user_id

    redirect('admin.php'); // γίνεται ανακατεύθυνση στο αρχείο actions.php
}

if($_GET['action']=='log_out2'){ //Αν η επιλεγμένη διεργασία είναι η log_out2(δηλαδή η έξοδος του
διαχειριστή στην εφαρμογή)

    $_SESSION['admin_user_id'] = ""; //Γίνεται καταχώρηση στο πεδίο admin_user_id του
πίνακα SESSION μηδενική τιμή(για να μην επιτραπεί στο χρήστη η έξοδος από την εφαρμογή )

    redirect('admin.php'); // γίνεται ανακατεύθυνση στο αρχείο admin.php
}

?>
```

Κώδικας 17 Έλεγχος ταυτότητας διαχειριστή στο αρχείο actions

10.8 Περιγραφή της εφαρμογής για τον χρήστη

Διασύνδεση των σελίδων του μενού

Στο αρχείο index.php γίνεται η κλήση της συνάρτησης content_site(), στο σημείο εκείνο του template που θα έχει τα περιεχόμενα και θέλουμε να μεταβάλλεται καθώς ο χρήστης επιλέγει κάποια από τις επιλογές του μενού. Η συνάρτηση αυτή δηλώνεται στο αρχείο functions.php.

Index.php

```
<div id="content">
    <? content_site();?> <!--κλήση της συνάρτησης content_site() -->
</div>
```

Κώδικας 18 Κλήση της συνάρτησης content_site() στο index αρχείο

functions.php

```
function content_site(){
// δήλωση της συνάρτησης content_site(){
$page = $_GET['page'];
// γίνεται ανάκτηση του page
if($page==""){// Έλεγχος της μεταβλητή page αν είναι κενή
$page='home';
// γίνεται καταχώρηση στη μεταβλητή page η τιμή 'home' αν δεν υπάρχει τιμή στην μεταβλητή $page.
Η 'home' είναι ουσιαστικά το περιεχόμενο της αρχικής σελίδας
}
include('pages/'.$page.'.php');
// Ενσωματώνει στο index το αντίστοιχο php αρχείο, με όνομα το περιεχόμενο της μεταβλητής page
}
```

Κώδικας 19 δήλωση της συνάρτησης content_site() στο αρχείο functions

Ο τρόπος λειτουργίας αυτής της συνάρτησης είναι ο εξής:

Στο αρχείο index.php έχουμε το menu ,όπου για κάθε επιλογή το link είναι index.php?page=..... , όπου το page αλλάζει με βάση την κάθε επιλογή. Η συνάρτηση content_site() μέσω της GET τραβάει τη μεταβλητή page και ενσωματώνει στο index το αντίστοιχο php αρχείο με όνομα το περιεχόμενο της μεταβλητής page.

LOG_IN τουριστικού γραφείου στην εφαρμογή

index.php

```
<? session_start(); ?>

<!-- Το ξεκίνημα ενός session γίνεται με μια κλήση στη συνάρτηση session_start -->

<? if($_SESSION['user_id']){

// Έλεγχος αν δίνεται user_id, τότε τρέχει τον παρακάτω κώδικα

    $sql="SELECT * FROM `users` WHERE id='".$_SESSION['user_id']."' ";

// sql ερώτημα προς τη βάση, βάση του οποίου επιλέγονται από το πίνακα users όλα τα στοιχεία της
εγγραφής, με id της εγγραφής ίδιο με το user_id που έχει κρατήσει η συνάρτηση session, με το που
θα κάνει ο χρήστης login στην εφαρμογή (αν φυσικά υπάρχει)

$result=mysql_query($sql);

// Για να εκτελέσουμε το παραπάνω sql ερώτημα χρησιμοποιούμε την εντολή mysql_query();

    $row=mysql_fetch_array($result);

// όλα τα στοιχεία της εγγραφής του πίνακα users αποθηκεύονται σε ένα πίνακα

echo 'Καλώς ήρθες <b>'.$row['name'].'</b>';

// Εμφάνιση μηνύματος 'Καλώς ήρθες' και το όνομα του χρήστη του οποίου το id στη βάση είναι το
ίδιο με το user_id που έχει αποθηκευτεί στο πίνακα SESSION

?>

<?    $sql="SELECT * FROM `users` WHERE id='".$_SESSION['user_id']."' ";

//sql ερώτημα προς τη βάση, βάση του οποίου επιλέγονται από το πίνακα users τα όλα τα στοιχεία
της εγγραφής, με id της εγγραφής ίδιο με το user_id που έχει κρατήσει η συνάρτηση session(αν
φυσικά υπάρχει)
```

```

$result=mysql_query($sql);

// Εκτελείται το παραπάνω sql ερώτημα

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

// Για κάθε μία από τις εγγραφές για τις οποίες ισχύει το παραπάνω sql ερώτημα, αποθηκεύονται τα
στοιχεία της εγγραφής σε ένα πίνακα και εκτελούνται τα παρακάτω

?>

<a href="actions.php?action=log_out"><b>Log Out</b></a><? } ?>

<!-- γίνεται ανακατεύθυνση στο αρχείο actions.php στη διεργασία action=log_out, όπου δίνεται η
δυνατότητα στο χρήστη να εξέρθει από την εφαρμογή-->

<? }else{?>

<a href="index.php?page=log_in"><b>Log In</b></a><br />

<!-- Αν δεν δίνεται κάποιο user_id γίνεται ανακατεύθυνση στο αρχείο index.php, στη σελίδα που
βρίσκεται η φόρμα(γία να μπορέσει ο χρήστης να συμπληρώσει τα στοιχεία του και να εισέρθει στην
εφαρμογή)-->

<? }?>

```

Κώδικας 20 Έλεγχος ταυτότητας χρήστη με έλεγχο συνόδων λειτουργίας στο αρχείο index

```

<form method="post" action="actions.php?action=log_in" >

// Αποστολή των στοιχείων της φόρμας με μέθοδο post στο αρχείο actions.php, στη διεργασία
log_in2(όπου εκεί θα γίνει έλεγχος αν τα στοιχεία αυτά είναι έγγυρα)

    <!--πεδία της φόρμας -->

<table>

    <tr>

        <td><b>Email</b></td>

        <td><input type="text" name="email" /></td>

    </tr>

<!--Ομοίως σχηματίζονται και τα υπόλοιπα πεδία της φόρμας-->

</form>

```

Κώδικας 21 Φόρμα εισαγωγής στοιχείων των τουριστικών γραφείων, για την εισοδο τους στο διαχειριστικό περιβάλλον

```

if($_GET['action']=='log_in'){
// έλεγχος αν το action ισούται με log_in,, τότε τρέχει τον παρακάτω κώδικα

$email = $_REQUEST['email'] ;

// ανάκτηση του email 'από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το input με
όνομα 'email'

$password = $_REQUEST['password'] ;

// ανάκτηση του password 'από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το
input με όνομα 'password'

$sql="SELECT * FROM `users` WHERE email='$email' AND password='$password' ";

// sql ερώτημα προς τη βάση, μέσω του οποίου γίνεται επιλογή όλων των στοιχείων της εγγραφής του
πίνακα users, η οποία εγγραφή έχει email και κωδικό ίδια με αυτά που δόθηκαν από το χρήστη

$result=mysql_query($sql);

// εκτέλεση του παραπάνω sql ερωτήματος

$num_rows = mysql_num_rows($result);/

// Η συνάρτηση num_rows, μας δίνει τον αριθμό των γραμμών που επιστρέφουν από το παραπάνω sql
ερώτημα

if($num_rows>0 ){

// Γίνεται έλεγχος αν υπάρχει έστω και μία εγγραφή με email και κωδικό ίδια με αυτά που δόθηκαν
από το χρήστη

$row=mysql_fetch_array($result);

// Αποθηκεύονται τα στοιχεία της εγγραφής σε ένα πίνακα

$_SESSION['user_id'] = $row['id'];

// Γίνεται καταχώρηση του id της εγγραφής της βάσης , που είχε αποθηκευτεί στο πίνακα row, στο
πίνακα SESSION στο πεδίο user_id. Δηλαδή ο πίνακας session κρατάει το id του χρήστη που έχει
κάνει login (αφού έχει γίνει ο έλεγχος ότι το ζεύγος $email και $password είναι σωστό,
ερωτώντας τη ΒΔ)

}else{

$_SESSION['user_id'] = ";

```

```

// Διαφορετικά το πεδίο 'user_id' του πίνακα SESSION παίρνει μηδενική τιμή
}
redirect('index.php?page=log_in');
// Γίνεται ανακατεύθυνση στο αρχείο index.php, στη σελίδα log_in
}
if($_GET['action']=='log_out'){
// Αν η επιλεγμένη διεργασία είναι η έξοδος(log_out)
$_SESSION['user_id'] = "";
// Γίνεται καταχώρηση στο πεδίο user_id του πίνακα SESSION μηδενική τιμή
redirect('index.php');
// Γίνεται ανακατεύθυνση στο αρχείο 'index.php'
}

```

Κώδικας 22 Έλεγχος ταυτότητας τουριστικού γραφείου στο αρχείο actions

Γρήγορη αναζήτηση χρήστη

Όπως προαναφέρθηκε δίνεται η δυνατότητα στο χρήστη ο οποίος αναζητά κάποιο προορισμό για τις διακοπές του, να κάνει μία γρήγορη αναζήτηση με βάση κάποια στοιχεία τα οποία ο κατασκευαστής της εφαρμογής κρίνει ότι είναι σημαντικά, προκειμένου να διευκολύνει το χρήστη (όπως χώρα, κατηγορία, ημ/νια αναχώρησης και επιστροφής). Αφού γίνει η δημιουργία της φόρμας και επιλέξει ο χρήστης τα στοιχεία που επιθυμεί, αποστέλλονται με μέθοδο get στο αρχείο offers.php. Στο αρχείο offers.php γίνεται με τη χρήση φίλτρων έλεγχος αν έχει δοθεί category_id, country_id. Από το αρχείο category.php αποστέλλεται το category, με το που θα επιλεγεί το link κάποιας συγκεκριμένης κατηγορίας(και αντιστοιχεί στο id της κατηγορίας αυτής), όμοια από το country_details.php αποστέλλεται το country_id και αντιστοιχεί στο id της χώρας που έχει επιλεγεί. Με τη χρήση φίλτρων λοιπόν γίνεται καταχώρηση σε μία μεταβλητή where η οποία είχε αρχικά μηδενική τιμή κάποιου περιορισμού για το category_id ή το country_id, ανάλογα με το αν έχει επιλεγεί κάποιο option στη φόρμα που χρησιμοποιείται για την γρήγορη αναζήτηση. Καθώς επίσης γίνεται προσθήκη στην μεταβλητή \$where επιπλέον περιορισμό σε περίπτωση που το \$date_from και το \$date_to δεν είναι κενά

```

<?
$where = '';

/*    Filtro: category_id    */

$category_id = $_GET['category_id'];

if($category_id){

$where = $where." category_id = '$category_id' AND " ;

// προσθήκη στην μεταβλητή $where περιορισμό σε περίπτωση που το category_id δεν είναι κενό
}

/*    Filtro: country_id    */

$country_id = $_GET['country_id'];

if($country_id){

    $where = $where." country_id = '$country_id' AND " ;

// προσθήκη στην μεταβλητή $where επιπλέον περιορισμό σε περίπτωση που το country_id δεν είναι
κενό
}

/*    Filtro: date_from    */

$date_from = $_GET['date_from'];

$date_to = $_GET['date_to'];

if($date_from && $date_to){

$date_from_array = explode('/', $date_from);

$date_from = $date_from_array[2].'-'.$date_from_array[1].'-'.$date_from_array[0];

// Δημιουργία πίνακα από τη μεταβλητή date_from, η οποία χωρίζεται σε τρία κελιά από το
χαρακτήρα '/'. Μετατρέπουμε την ημερομηνία σε sql ημερομηνία π.χ 12/5/2010->2010/5/12

$date_to_array = explode('/', $date_to);

$date_to = $date_to_array[2].'-'.$date_to_array[1].'-'.$date_to_array[0];

// Δημιουργία πίνακα από τη μεταβλητή date_from, η οποία χωρίζεται σε τρία κελιά από το
χαρακτήρα '/'. Μετατρέπουμε την ημερομηνία σε sql ημερομηνία π.χ 12/5/2010->2010/5/12

```

```

$where = $where." ( date_from >= UNIX_TIMESTAMP('$date_from') AND date_from <=
UNIX_TIMESTAMP('$date_to') OR date_to >= UNIX_TIMESTAMP('$date_from') AND
date_to <= UNIX_TIMESTAMP('$date_to') ) AND ";

// προσθήκη στην μεταβλητή $where επιπλέον περιορισμό σε περίπτωση που το $date_from και το
$date_to δεν είναι κενά
}

<?

$sql="SELECT * FROM `offers` WHERE $where id<>0 ";

// προσθήκη της μεταβλητής $where όπου βρίσκονται όλοι οι περιορισμοί για να βρούμε τα αντίστοιχα
offers από την βάση δεδομένων με τη χρήση sql ερωτήματος προς τη βάση

$result=mysql_query($sql);

// εκτέλεση του παραπάνω sql ερωτήματος

$num_rows = mysql_num_rows($result);

// Η συνάρτηση num_rows, μας δίνει τον αριθμό των γραμμών που επιστρέφουν από το παραπάνω sql
ερώτημα

if($num_rows>0){

//Γίνεται έλεγχος αν υπάρχει έστω και μία εγγραφή με τα επιλεγμένα πεδία να είναι ίδια με τα
αντίστοιχα στη βάση

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

//Τότε για κάθε μία εγγραφή που ισχύει το παραπάνω

?>

        <div class="offers_item">
<a href="index.php?page=offer_details&offers_id=<?=$row['id']?>"><? echo $row['title'];
?></a></div>

// Φορτώνεται και εμφανίζεται μία εικόνα (που αντιστοιχεί στην επιλεγμένη προφορά) καθώς και το
όνομα της προσφοράς

        <? }

    }else{

        $message='Δεν υπάρχουν αποτελέσματα με αυτα τα στοιχεία';

```

```
// Διαφορετικά αν δεν υπάρχει καμία προσφορά με τα επιλεγμένα στοιχεία εμφανίζεται το παραπάνω
μήνυμα
?>
<div class="message"><? echo $message;?><? } ?></div>
</div>
```

Κώδικας 23 Χρήση φίλτρων για γρήγορη αναζήτηση χρήστη στο αρχείο `offers`

Φόρμα επικοινωνίας χρήστη με το διαχειριστή του συστήματος

```
<form method="post" action="actions.php?action=communication_form" class="forma">
// Αποστολή των στοιχείων της φόρμας με μέθοδο post στο αρχείο actions.php, στη διεργασία
communication_form (για να επιτευχθεί η επικοινωνία του χρήστη με το διαχειριστή του συστήματος)
<table align="center" cellpadding="10" cellspacing="0">
  <tr>
    <td><label for="name">Όνομα:</label></td>
    <td><input name="contact_name" type="text" id="name" size="25" maxlength="50"
class="check" />(*) </td>
  </tr>
<!-- Όμοια σχηματίζονται και τα υπόλοιπα πεδία της φόρμας-->
  <tr>
    <td align="right"><input type="submit" name="Submit" value="Αποστολή"
style="width:100px"></td>
  </tr>
</table>
</form>
```

Κώδικας 24 Φόρμα επικοινωνίας με το διαχειριστή στο αρχείο `communication_form`

```

if($_GET['action']=='communication_form'){
// Γίνεται έλεγχος αν το action ισούται με communication_form, τότε εκτελούνται τα παρακάτω
$name=$_REQUEST['name'];
// ανάκτηση του name από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το input με
όνομα 'name'
$email = $_REQUEST['email'] ;
// ανάκτηση του email από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το input με
όνομα 'email'
$phone = $_REQUEST['phone'] ;
// ανάκτηση του phone από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το input με
όνομα ' phone '
$country = $_REQUEST['country'] ;
// ανάκτηση του country από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το input
με όνομα ' country '
$msg_subject = $_REQUEST['msg_subject'] ;
// ανάκτηση του msg_subject από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το
input με όνομα ' msg_subject '
$msg = $_REQUEST['msg'] ;
// ανάκτηση του msg από τον πίνακα REQUEST, αφού συμπληρωθεί από το χρήστη από το input με
όνομα 'email'
$to = 'xrisalan@gmail.com'; // email διαχειριστή με τον οποίο θα επικοινωνήσει ο χρήστης
mail( $to, "Message Subject: $contact_subject", $contact_textmsg, "From: $contact_email" ,
"Name: $name", "Phone: $phone", "Country: $country");
// αποστολή mail στο διαχειριστή με τα στοιχεία που έχουν ανακτηθεί από τη φόρμα
redirect('index.php?page=contactThanks');
// Ανακατεύθυνση στο αρχείο contactThanks(όπου ο διαχειριστής ευχαριστεί το χρήστη για το mail
του)
}

```

Κώδικας 25 Αποστολή mail στο διαχειριστή στο αρχείο actions


```

$(document).ready(function(){

    $(".datepicker").datepicker({ date Format: 'dd/mm/yy' });

    $(".forma").submit(function() { // τρέχει όταν κάνουμε submit στη φόρμα

        var error = checkFillInputs($(this));

        if (error) {

            return true; // τρέχει η φόρμα

        }

        return false; // error στη φόρμα, δηλαδή δεν έχει συμπληρωθεί κάποιο πεδίο

    });

    $(".forma").find(".check").focus(function () { // όταν κάνω focus, δηλαδή όταν μπει ο
κέρσορας στο πεδίο

        $(this).removeClass("form_error"); // βγάζουμε την class "form_error" από το πεδίο

    });

});

function checkFillInputs(forma){

    var msg = 'Παρακαλώ';

    var inputs = forma.find(".check").get(); // πίνακας με όλα τα πεδία που έχουν class
"form_error"

    var i=0;

    var empty=true;

    for(i=0; i<inputs.length; i++){

        if(jQuery.trim($(inputs[i]).val())=="") { // έλεγχος στο πεδίο αν έχει συμπληρωθεί

            empty = false;

            $(inputs[i]).addClass("form_error"); // προσθέτω class με όνομα
"form_error" και είναι το ροζ χρώμα στο πεδίο(που εμφανίζεται σε περίπτωση που δεν έχουν
συμπληρωθεί τα υποχρεωτικά αυτά πεδία στη φόρμα))

        }

    }

}

```

```

    }

}

if(!empty){

    msg += ",\nσυμπληρώστε όλα τα υποχρεωτικά στοιχεία";

}

if(!empty){

    alert(msg);

}

return empty;

}

```

Εικόνα 45 Δήλωση των υποχρεωτικών πεδίων της φόρμας στο αρχείο `init.js`

11 Συμπεράσματα

- Κατά την διάρκεια κατασκευής της παρούσας εφαρμογής έγινε αντιληπτό πόσο χρήσιμος είναι ο δυναμικός μηχανισμός κατασκευής ιστοσελίδων, ο οποίος δημιουργεί σελίδες διαβάζοντας πληροφορίες που προέρχονται από μια βάση δεδομένων. Η μέθοδος αυτή είναι αρκετά ευέλικτη τόσο για τον χρήστη ο οποίος μπορεί να εξατομικεύει τις σελίδες χρησιμοποιώντας διάφορα κριτήρια δικής του επιλογής καθώς επίσης και για το χειριστή της εφαρμογής, ο οποίος μπορεί να αλλάζει με ευκολία κάποια πληροφορία σε όλες τις σελίδες του site ταυτόχρονα, κάνοντας μόνο μία αλλαγή στη βάση δεδομένων ή στο template, η οποία θα ενημερώνει όλες τις σελίδες.
- Η ακεραιότητα της βάσης δεδομένων είναι το σημαντικότερο σημείο κατά τη κατασκευή της εφαρμογής λόγω του ότι μόνο πιστοποιημένοι χρήστες πρέπει να έχουν πρόσβαση στο διαχειριστικό περιβάλλον της εφαρμογής, για να διασφαλίζεται έτσι η ακεραιότητα των δεδομένων. Επομένως στο στάδιο αυτό είναι η χρήση της συνάρτησης `session_start` για τον εξής λόγο οτι:όταν ένας χρήστης ζητά μια σελίδα, και στη συνέχεια μια άλλη, το HTTP δεν έχει τρόπο να καταλάβει ότι οι δυο αιτήσεις ήρθαν από τον ίδιο χρήστη.Η ιδέα των συνόδων λειτουργίας (sessions) είναι να μπορούμε να παρακολουθούμε ένα χρήστη κατά τη διάρκεια της επίσκεψής του σε ένα web site. Επομένως με τον τρόπο αυτό επιτυγχάνεται η εμφάνιση

περιεχομένων σύμφωνα με το επίπεδο πιστοποίησης ή προσωπικών προτιμήσεων και η παρακολούθηση της συμπεριφοράς του χρήστη.

- Επίσης είναι γεγονός πόσο σημαντικός είναι ο συνδιασμός των τεχνολογιών που λειτουργούν στη μεριά του πελάτη (όπως html,css κ.λ.π) που μπορούν να βελτιώσουν το user interface με το να αποκρίνονται άμεσα στις ενέργειες των χρηστών. Χρειάζεται προσαρμογή των τεχνολογιών αυτών για να εμφανίζονται σωστά τα περιεχόμενα της εφαρμογής σε οποιοδήποτε πρόγραμμα πλοήγησης επιλέξει ο χρήστης(Mozilla, internet Explorer)
- Σημαντικό επίσης είναι και ο σχεδιασμός της ιστοσελίδας και η διεπαφή της. Πολλά μέρη όπως το μενού, η αναζήτηση μένουν σταθερά και αλλάζει μόνο το περιεχόμενο της συγκεκριμένης σελίδας. Έτσι ο χρήστης ξέρει που βρίσκεται στην ιστοσελίδα και η πλοήγηση γίνεται πιο εύκολη.

12 Βιβλιογραφία- Πηγές

HTML

- ✚ <http://www.w3c.org/MarkUp>
- ✚ <http://www.w3.org/TR/html4>
- ✚ <http://www.htmlreference.com/complete.html>
- ✚ http://www.webdeveloper.com/html/html_metatags.html

CSS

- ✚ <http://www.w3schools.com/Css/default.asp>

PHP

- ✚ http://www.tizag.com/phpT/fileupload.php?MAX_FILE_SIZE=100000&uploadedfile=logo.jpg
- ✚ <http://akisplace.com/php-tutorials/php-intro-part-1/>

- <http://www.php.net/manual/en/book.session.php>
- <http://www.w3schools.com/php/default.asp>
- [PHPVideoTutorials.com_Database_Website_From_Scratch_\(2008\)](http://www.phpvideo.com/Database_Website_From_Scratch_(2008))
- <http://www.freestuff.gr/forums/viewtopic.php?t=18943>
- [http://www.tizag.com/phpT/fileupload.php\](http://www.tizag.com/phpT/fileupload.php)
- <http://en.wikipedia.org/wiki/Php>

SQL

- <http://en.wikipedia.org/wiki/Mysql>
- <http://www.w3schools.com/sql/default.asp><http://dev.mysql.com/doc/#manual>
- <http://www.clearlearning.com/v4/install/mysql.html>

APACHE

- <http://www.wampserver.com/en/download.php>
- http://en.wikipedia.org/wiki/Apache_HTTP_Server
- <http://httpd.apache.org/>
- <http://www.simplehelp.net/2008/08/25/how-to-install-and-setup-apache-mysql-and-php-in-windows/el/>
- <http://www.bulletprooftemplates.com/how-to-install-wamp-20>

JAVASCRIPT

- http://docs.jquery.com/Main_Page
- <http://el.wikipedia.org/wiki/JavaScript>
- <http://www.w3schools.com/js/default.asp>