



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών

Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων



Πτυχιακή Εργασία

Τίτλος: Ψηφιακό Βιβλιοπωλείο

Κουλουμπριτσίδου Σοφία (ΑΜ: 1545)

Επιβλέπων Καθηγητής:

Μαλάμος Αθανάσιος

Κλεισαρχάκης Μιχ.

Επιτροπή Αξιολόγησης: Ιωάννης Παχουλάκης

Κώστας Βασιλάκης

ABSTRACT

"Digital Books" is a website where users can choose their favorite books, and then purchase them into the internet.

This application is a 3-level users website (administrator, registered user, user as a guest), where each of them can perform the functions that correspond to him, according to his rights, as they will be described below.

The Web application was developed with the programming languages PHP, MYSQL and Html, and enables access and navigation to users through the Internet. The administrator has unlimited rights, as he can change, modify and correct details of the application. He can also supervise the actions of the members and their profile information.

According to the above, a user friendly website was developed which meets all the requirements of usability and security of personal data. At the same time it ensures the security of information which every registered member enters or modifies.

Various programs were used. More specifically, the program Dreamweaver CS3 for the website design, Flash CS3 for the creation of graphics and a combination of PHP - MySQL – Apache tools for the creation of the database, which are free distributed on the internet and they are analyzed in detail in forthcoming chapters. Moreover, the freeware tool phpMyAdmin was used which provides a graphical environment of the database system management. The HTML language in conjunction with the PHP language were used for the website programming, and SQL language for the database programming.

For any further questions that users may have, there has been developed a communication form with the administrator, who could offer users further details via e-mail.

ΣΥΝΟΨΗ

Η ιστοσελίδα “ΨΗΦΙΑΚΟ ΒΙΒΛΙΟΠΩΛΕΙΟ” είναι ένας ιστοχώρος στον οποίο οι χρήστες μπορούν να διαλέγουν τα αγαπημένα τους βιβλία, και έπειτα να μεταβαίνουν στην αγορά τους μέσω του Παγκόσμιου Ιστού.

Η συγκεκριμένη εφαρμογή αποτελεί μια ιστοσελίδα 3 επιπέδων χρηστών, (διαχειριστής, χρήστης με προνόμια, χρήστης επισκέπτης), όπου ο καθένας από αυτούς μπορεί να εκτελέσει τις λειτουργίες που του αντιστοιχούν, σύμφωνα με τα δικαιώματα του, όπως θα περιγραφούν παρακάτω.

Η Ιστοσελίδα αναπτύχθηκε με τις γλώσσες προγραμματισμού PHP, MYSQL και Html, και δίνει τη δυνατότητα πρόσβασης και πλοήγησης στους χρήστες μέσω του Παγκόσμιου Ιστού. Ο διαχειριστής έχει απεριόριστα δικαιώματα, δηλαδή να αλλάζει, να τροποποιεί και να διορθώνει στοιχεία της εφαρμογής και να επιβλέπει τις ενέργειες των μελών και τα στοιχεία αυτών.

Με βάση τα παραπάνω αναπτύχθηκε μια φιλική προς το χρήστη ιστοσελίδα, που πληροί όλες τις προϋποθέσεις ευχρηστίας και ασφάλειας προσωπικών δεδομένων. Ταυτόχρονα διασφαλίζει τον έλεγχο των στοιχείων που εισάγει ή τροποποιεί κάθε εγγεγραμμένο μέλος.

Πιο συγκεκριμένα, χρησιμοποιήθηκε το πρόγραμμα Dreamweaver CS3 για τον σχεδιασμό της ιστοσελίδας, το Flash CS3 για την δημιουργία γραφικών και για την δημιουργία της βάσης δεδομένων χρησιμοποιήθηκε ο συνδυασμός των εργαλείων PHP – MySQL – Apache που διανέμονται δωρεάν στο διαδίκτυο και αναλύονται διεξοδικά στα επόμενα κεφάλαια. Ακόμη χρησιμοποιήθηκε και το εργαλείο phpMyAdmin το οποίο επίσης διανέμεται δωρεάν και παρέχει ένα γραφικό περιβάλλον διαχείρισης του σχεσιακού συστήματος βάσης δεδομένων. Χρησιμοποιήθηκε η γλώσσα προγραμματισμού HTML σε συνδυασμό με την γλώσσα PHP για τον προγραμματισμό της ιστοσελίδας, και η γλώσσα προγραμματισμού SQL για τον προγραμματισμό της βάσης δεδομένων.

Για οποιαδήποτε απορία του χρήστη έχει σχεδιαστεί φόρμα επικοινωνίας με τον διαχειριστή της ιστοσελίδας, ο οποίος στέλνει τις ανάλογες διευκρινήσεις στο email του χρήστη.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ABSTRACT	2
ΣΥΝΟΨΗ	3
ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ	7
1.1 Πρόλογος	7
1.2 Κίνητρο για την διεξαγωγή της εργασίας.....	7
1.3 Σκοπός και Στόχοι της εργασίας.....	7
1.4 Δομή της εργασίας.....	8
ΚΕΦΑΛΑΙΟ 2: ΜΕΘΟΔΟΛΟΓΙΑ ΥΛΟΠΟΙΗΣΗΣ	9
2.1 Μέθοδος Ανάλυσης και Ανάπτυξης Πτυχιακής	9
2.1.1 Γνωστά ηλεκτρονικά βιβλιοπωλεία.....	9
2.1.2 Διαμόρφωση	10
2.2 Τεχνολογία.....	10
2.2.1 Δυναμικές Τοποθεσίες.....	11
2.2.2 HTML.....	11
2.2.3 PHP.....	12
2.2.4 Σχεδίαση Βάσεων δεδομένων	19
2.2.5 Κανονικές Παραστάσεις συμβατές με Perl	21
2.2.6 Δομή Τοποθεσίας	22
ΚΕΦΑΛΑΙΟ 3: ΣΧΕΔΙΟ ΔΡΑΣΗΣ ΓΙΑ ΤΗΝ ΕΚΠΙΟΝΗΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ	24
3.1 State of Art στο χώρο των ηλεκτρονικών βιβλιοπωλείων.....	24
3.2 State Of Art στην τεχνολογία	30
3.2.1 Διαδίκτυο.....	30
3.2.2 Παγκόσμιος Ιστός.....	30
3.2.3 Τοποθεσίες Ιστού και Ιστοσελίδες	31
3.2.4 Σχεδιασμός Σελίδων Ιστού	31
3.2.5 Φόρμες Ιστού.....	32
3.2.6 Διεύθυνση IP	32
3.2.7 Domain Name System.....	32
3.2.8 Φυλλομετρητής Ιστού	33
3.2.9 URL	33
3.2.10 Επικοινωνία μέσω Internet.....	34
3.2.11 Μηχανή αναζήτησης	34
3.2.12 Ηλεκτρονικό Εμπόριο	35

3.2.13	Ασφάλεια στο Internet.....	35
3.3	Χρονοδιάγραμμα Εργασίας	37
ΚΕΦΑΛΑΙΟ 4: ΚΥΡΙΟ ΜΕΡΟΣ ΠΤΥΧΙΑΚΗΣ.....		38
4.1	Ρόλοι και δικαιώματα πρόσβασης.....	38
4.1.1	Λειτουργίες Διαχειριστή.....	38
4.1.2	Λειτουργίες Χρήστη-Μέλους.....	38
4.1.3	Λειτουργίες Χρήστη-Επισκέπτη	39
4.1.4	Δημιουργία Χρηστών και Προνομιών.....	39
4.2	Σχεδιασμός Ιστοχώρου	41
4.2.1	Μορφοποίηση Σελίδων με Φύλλα CSS.....	41
4.2.2	Δημιουργία Διατάξεων με Πίνακες.....	46
4.2.3	Σύνδεσμοι.....	47
4.2.4	Διαδικασία κατασκευής του ιστοχώρου.....	47
4.3	Δημιουργία Flash Animation.....	62
4.4	Δημιουργία Βάσης Δεδομένων – Bookstore	66
4.4.1	Ονομασία Στοιχείων.....	66
4.4.2	Επιλογή τύπου στηλών.....	66
4.4.3	Περιγραφή Πινάκων.....	68
4.5	Προγραμματισμός με την PHP.....	77
4.5.1	Εγγραφή Χρήστη.....	77
4.5.2	Είσοδο – Έξοδο από τον Λογαριασμό του Χρήστη	85
4.5.3	Επικοινωνία με τον διαχειριστή	90
4.5.4	Δημιουργία Forum.....	92
4.5.5	Εμφάνιση Χρηστών.....	95
4.5.6	Διαγραφή Χρήστη	98
4.5.7	Αλλαγή Στοιχείων Λογαριασμού	101
4.4.8	Εμφάνιση Βιβλίων.....	106
4.5.9	Αναζήτηση Βιβλίων	113
4.5.10	Λίστα αγαπημένων βιβλίων	116
4.5.11	Καλάθι Αγορών.....	122
4.5.12	Ολοκλήρωση Αγοράς.....	130
4.5.13	Παραγγελία του χρήστη	132
4.5.14	Πίνακας αγορών	134
4.5.15	Αγορές Χρηστών.....	135

4.5.16	Διαγραφή Παραγγελίας	136
4.5.17	Προσθήκη Βιβλίου	138
4.6	Ασφάλεια Ιστοχώρου	151
4.7	Διαδικασία Πληρωμής.....	152
ΚΕΦΑΛΑΙΟ 5: ΑΠΟΤΕΛΕΣΜΑΤΑ		153
5.1	Συμπεράσματα.....	153
5.2	Μελλοντική Εργασία και Επεκτάσεις	153
ΒΙΒΛΙΟΓΡΑΦΙΑ.....		155
ΠΑΡΑΡΤΗΜΑΤΑ		156
ΠΑΡΑΡΤΗΜΑ Α		156
	Οδηγίες Εγκατάστασης	156
ΠΑΡΑΡΤΗΜΑ Β.....		156
	DreamWeaver CS3.....	156
	MYSQL.....	156
	Προγράμματα σχεδίασης βάσης δεδομένων	156
	Σύνδεση στη MySQL	156
	Τοπικός Server Apache	156
	Photoshop CS3	156
	Flash CS3	156
	phpMyAdmin	156

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ

1.1 Πρόλογος

Στο πλαίσιο των σπουδών στο τμήμα Πληροφορικής του Τ.Ε.Ι. Ηρακλείου εκπονήθηκε πτυχιακή εργασία με θέμα «Ψηφιακό Βιβλιοπωλείο» όπου γίνεται μια προσπάθεια να δημιουργηθεί ένας τρόπος επικοινωνίας της επιχείρησης με τους πελάτες της. Θα είναι μια εφαρμογή παραγγελίας βιβλίων. Θα παρουσιαστεί μια ιστοσελίδα όπου θα μπορεί κάποιος πελάτης να έχει τη δυνατότητα να παραγγέλνει προϊόντα διαμέσου του Διαδικτύου από το σπίτι του. Ο πελάτης θα έχει τη δυνατότητα να επιλέξει όποιο βιβλίο επιθυμεί και να δώσει την παραγγελία του. Ο κάθε πελάτης απαιτείται να γίνει μέλος έτσι ώστε να πραγματοποιήσει κάποια αγορά.

Επίσης, υπάρχει δυνατότητα επικοινωνίας των χρηστών σε μια σελίδα η οποία εμπεριέχει σχολιασμούς των βιβλίων. Ο κάθε χρήστης συμπληρώνοντας την φόρμα της σελίδας μπορεί να κάνει την δική του κριτική. Η επικοινωνία με τον διαχειριστή πραγματοποιείται σε διαφορετική σελίδα στην οποία ο χρήστης στέλνει ένα μήνυμα απευθείας στο email του διαχειριστή.

Τέλος για να γίνει η ιστοσελίδα πιο ελκυστική προς τον πελάτη είναι απαραίτητη η χρήση γραφικών.

1.2 Κίνητρο για την διεξαγωγή της εργασίας

Η απόφαση για την επιλογή της πτυχιακής πάρθηκε όταν μου ζητήθηκε η δημιουργία μιας ιστοσελίδας ενός βιβλιοπωλείου, μέσω της οποίας οι πελάτες θα προέβαιναν στην αγορά των βιβλίων. Οι χρήστες του Διαδικτύου μπορούν εύκολα και γρήγορα να περιηγηθούν σε μια βάση πληροφοριών, να κάνουν χρήση της ηλεκτρονικής αλληλογραφίας, και γενικά να χρησιμοποιήσουν διάφορες υπηρεσιών που έχουν στη διάθεσή τους.

1.3 Σκοπός και Στόχοι της εργασίας

Σκοπός της πτυχιακής αυτής ήταν η μελέτη, σχεδίαση και ανάπτυξη μιας ιστοσελίδας όπου οι χρήστες θα μπορούν να κάνουν online αγορές. Επίσης, μελετήθηκαν τρόποι δημιουργίας μιας βάσης δεδομένων όπου θα αποθηκεύονται όλα τα δεδομένα που θα εμφανίζονται στην ιστοσελίδα, καθώς και στοιχεία των πελατών όταν αυτοί δημιουργήσουν έναν λογαριασμό.

Η ιστοσελίδα θα δίνει τη δυνατότητα σε ένα χρήστη να ενημερωθεί για τα βιβλία των διαφόρων κατηγοριών, για εκείνα που μόλις κυκλοφόρησαν, για εκείνα που έγιναν best seller, αλλά και να παραθέσει την άποψη του στο forum της ιστοσελίδας. Επίσης μπορεί να αναζητήσει το βιβλίο που επιθυμεί και να δημιουργήσει μια λίστα με τα αγαπημένα του βιβλία και έπειτα να μεταβεί στην αγορά τους.

1.4 Δομή της εργασίας

Στο 1ο κεφάλαιο, παρουσιάζεται η εισαγωγή, στην οποία περιγράφονται ο σκοπός δημιουργίας της ιστοσελίδας, το κίνητρο για την υλοποίησή της, μια μικρή περίληψη και η δομή της εργασίας.

Στο 2ο κεφάλαιο, παρουσιάζεται η μεθοδολογία υλοποίησης της εργασίας.

Στο 3ο κεφάλαιο, παρουσιάζεται το σχέδιο με το οποίο αναπτύχθηκε η ιστοσελίδα.

Στη συνέχεια, παρουσιάζεται το Κύριο Μέρος της πτυχιακής. Συγκεκριμένα, στο 4ο κεφάλαιο, παρουσιάζεται αναλύονται οι ειδικές απαιτήσεις του λογισμικού και συγκεκριμένα οι λειτουργικές απαιτήσεις για κάθε επίπεδο χρηστών. Ο τρόπος δημιουργίας της Βάσης Δεδομένων. Η διαδικασία με την οποία σχεδιάστηκε ο ιστοχώρος. Καθώς και οι μέθοδος δημιουργίας λογαριασμού και αγοράς των βιβλίων που ο χρήστης επιθυμεί.

Έπειτα παρατίθεται η Βιβλιογραφία πάνω στην οποία βασίστηκε η πραγμάτωση της ιστοσελίδας.

Και τέλος, παρουσιάζονται τα παραρτήματα. Αρχικά παρατίθενται ο τρόπος εγκατάστασης του προγράμματος XAMPP το οποίο εμπεριέχει την PHP, την MySQL, τον Apache server και το πρόγραμμα phpMyAdmin.

ΚΕΦΑΛΑΙΟ 2: ΜΕΘΟΔΟΛΟΓΙΑ ΥΛΟΠΟΙΗΣΗΣ

2.1 Μέθοδος Ανάλυσης και Ανάπτυξης Πτυχιακής

Με την είσοδο της τεχνολογίας η έννοια του βιβλιοπωλείου μετεξελίχθηκε. Η διακίνηση του βιβλίου μέσω του Παγκόσμιου Ιστού είναι πλέον μια πραγματικότητα. Με την εμφάνιση του Διαδικτύου, λοιπόν, το βιβλιοπωλείο αποκτά νέες διαστάσεις και γίνεται προσβάσιμο από όλο τον κόσμο για να προσφέρει μεγαλύτερη ευκολία στις συναλλαγές των πελατών-χρηστών. Είναι αυτονόητο ότι το ηλεκτρονικό βιβλιοπωλείο διαφοροποιείται από το συμβατό όσον αφορά τη εμφάνιση, την εξυπηρέτηση των πελατών-χρηστών καθώς και στον τρόπο συναλλαγής μαζί τους. Με την εμφάνιση του ηλεκτρονικού εμπορίου οι χρήστες του Διαδικτύου, αφότου επιλέξουν το βιβλίο που επιθυμούν από το site του εκάστοτε βιβλιοπωλείου, έχουν την δυνατότητα να προβούν στην online αγορά του.

Το ηλεκτρονικό βιβλιοπωλείο, λοιπόν, δεν είναι τίποτα άλλο από μια διεύθυνση και μια σελίδα στο Διαδίκτυο. Είναι ένα κατάστημα χωρίς τοπικούς και χρονικούς περιορισμούς (χωρίς ουρές και καθυστερήσεις) που διαθέτει απεριόριστες δυνατότητες και πελάτες από όλο τον κόσμο. Ο καθένας μπορεί να αγοράσει το βιβλίο οποιουδήποτε συγγραφέα και εκδότη, ανεξαρτήτως γλώσσας, αλλά μόνο ηλεκτρονικά. Η επικοινωνία γίνεται μέσω ηλεκτρονικού ταχυδρομείου (email) και οι συναλλαγές πραγματοποιούνται μέσω πιστωτικών καρτών.

2.1.1 Γνωστά ηλεκτρονικά βιβλιοπωλεία

Amazon

Είναι ένα από τα μεγαλύτερα ηλεκτρονικά βιβλιοπωλεία. Εμφανίζεται σε διάφορες γλώσσες και προσφέρει πολύ μεγάλη ποικιλία θεματικών κατηγοριών για την αναζήτηση βιβλίων. Το Amazon μπορεί να ξεκίνησε ως ένα online βιβλιοπωλείο, αλλά σύντομα διαφοροποιήθηκε παρέχοντας και άλλα προϊόντα όπως DVD, CD, MP3 downloads, λογισμικό ηλεκτρονικών υπολογιστών, ηλεκτρονικά παιχνίδια, ηλεκτρονικά είδη, είδη ένδυσης, έπιπλα και τρόφιμα. Επιπλέον, παρέχεται η δυνατότητα συμμετοχής σε δημοπρασίες και η αγορά μεταχειρισμένων βιβλίων και προϊόντων. Έχει δημιουργήσει ξεχωριστές ιστοσελίδες στον Καναδά, το Ηνωμένο Βασίλειο, τη Γερμανία, τη Γαλλία, την Ιταλία, την Αυστρία, την Ιαπωνία και την Κίνα παρέχοντας επίσης τη διεθνή ναυτιλία σε ορισμένες χώρες για ορισμένα από τα προϊόντα.

Greek books

Πρόκειται για τον ιστότοπο ενός από τα μεγαλύτερα ελληνικά ηλεκτρονικά βιβλιοπωλεία. Φροντίζουν να κάνουν τις αγορές ελκυστικότερες με προσφορές, νέες κυκλοφορίες, προτεινόμενους τίτλους, δωρεάν αποστολές από κάποιο ύψος αγορών. Επίσης, προσφέρεται μία εκτενής συλλογή από τίτλους που κυκλοφορούν αυτή τη στιγμή στα βιβλιοπωλεία της χώρας. Εκτός από βιβλία όμως παρέχει παιδικά είδη, ταινίες, ηλεκτρονικά παιχνίδια κ.α. Η

εμφάνιση του καλαθιού αγοράς και της Wishlist της ιστοσελίδας μας βασίστηκε σε αυτήν του συγκεκριμένου ιστότοπου.

Bibliothora

Η Bibliothora είναι ένας ιστότοπος καθαρά αφιερωμένος στο βιβλίο. Πάνω στην διαμόρφωση αυτού του ιστοχώρου βασίστηκε η δημιουργία της ιστοσελίδας. Στην αρχή βρίσκεται το μενού και ακριβώς από κάτω ένα γραφικό μέσω του οποίου εμφανίζονται τα προτεινόμενα βιβλία. Αριστερά, όπως και στα περισσότερα ηλεκτρονικά βιβλιοπωλεία, βρίσκονται οι κατηγορίες των βιβλίων. Η εμφάνιση της μηχανής αναζήτησης και ο χώρος που καταλαμβάνει στην ιστοσελίδα βασίστηκε στην μηχανή αναζήτησης της Bibliothoras. Επίσης, στην αρχική σελίδα γίνεται η εμφάνιση των νέων κατηγοριών. Η διαφορά βρίσκεται στην φόρμα εισόδου που στην περίπτωση του δικού μας ιστοχώρου τοποθετήθηκε στην αρχική σελίδα.

2.1.2 Διαμόρφωση

Φυσικά, κάθε βιβλιοπωλείο ακολουθεί την δική του πολιτική και τακτική στην οργάνωση του καταστήματος. Τις περισσότερες φορές τα γραφικά είναι λίγα, ενώ μεγαλύτερη σημασία δίνεται στην περιγραφή των βιβλίων. Τα στοιχεία περιγραφής, εκτός από τον τίτλο, το συγγραφέα και τον εκδότη περιλαμβάνουν και μια φωτογραφία του εξώφυλλου, το ISBN, τον αριθμό των σελίδων και εννοείται την τιμή. Επίσης, υπάρχει μια λίστα με τα Best Sellers, δηλαδή τα καλύτερα βιβλία σύμφωνα με τις πωλήσεις, μια άλλη με τις πιο πρόσφατες κυκλοφορίες, και μια λίστα με τα βιβλία που προτείνει το εκάστοτε βιβλιοπωλείο. Εκτός από λίστες βιβλίων είναι πιθανό η ύπαρξη μιας ενότητας σχολιασμού των βιβλίων, όπου ο χρήστης μπορεί να διαβάσει τα σχόλια και την κριτική των αναγνωστών πριν αποφασίσει για κάποια αγορά του.

Η διαμόρφωση της κεντρικής σελίδα διαφέρει σε κάθε ηλεκτρονικό βιβλιοπωλείο. Στις περισσότερες περιπτώσεις υπάρχει το βασικό μενού με τις κύριες ενότητες του ιστότοπου, καθώς και μια μηχανή αναζήτησης ώστε ο χρήστης να εντοπίσει πιο εύκολα το βιβλίο που επιθυμεί. Επίσης, πολύ σημαντική είναι η κατηγοριοποίηση των βιβλίων για την καλύτερη πλοήγηση του χρήστη. Συχνά υπάρχει σελίδα με πληροφορίες για το ιστορικό, τους στόχους και την αιτία δημιουργίας του ηλεκτρονικού βιβλιοπωλείου.

Για την διεκπεραίωση της συναλλαγής ο χρήστης πρέπει να συμπληρώσει τα στοιχεία του σε μια ηλεκτρονική φόρμα που είναι απαραίτητα για να πραγματοποιηθεί η παραγγελία. Σε όλα τα ηλεκτρονικά βιβλιοπωλεία υπάρχει το καλάθι αγορών στο οποίο ο χρήστης τοποθετεί βιβλία που των ενδιαφέρουν, επιλέγει την ποσότητα των βιβλίων που θέλει να αγοράσει και προχωρεί στην πληρωμή ή στην ακύρωση της παραγγελίας.

2.2 Τεχνολογία

Η ιστοσελίδα θα κατασκευαστεί με το πρόγραμμα σχεδίασης ιστοσελίδων DreamWeaver. Το DreamWeaver είναι το κορυφαίο πρόγραμμα δημιουργίας και επεξεργασίας ιστοσελίδων που επιτρέπει στο χειριστή συστήματος να οργανώνει το περιεχόμενο και να προσαρμόζει την παρουσίαση του ιστοχώρου. Με το DreamWeaver μπορούμε να δημιουργήσουμε φόρμες (forms),

πλαίσια (frames), πίνακες (tables) και άλλα αντικείμενα της HTML. Επίσης, μπορεί να χρησιμοποιηθεί και για τη δημιουργία εφαρμογών πολυμέσων. Έχει δυνατότητες για δημιουργία δυναμικής HTML (DHMTL), δημιουργία επιπέδων (layers) και συγγραφή σεναρίων (scripts). Τέλος, περιέχει δικές του συμπεριφορές (behaviors), που είναι έτοιμα scripts τα οποία μπορούμε να προσθέσουμε πολύ εύκολα σ' ένα αντικείμενο.

2.2.1 Δυναμικές Τοποθεσίες

Οι δυναμικές ιστοσελίδες, δημιουργούνται με μία γλώσσα προγραμματισμού κατάλληλη για το διαδίκτυο, όπως είναι η PHP, η οποία ενσωματώνεται σε ετικέτες HTML και έτσι ουσιαστικά αποτελούν πλήρες εφαρμογές. Η διαφορά ανάμεσα σε δυναμικές και στατικές ιστοσελίδες αφορά την λειτουργία και όχι την εμφάνιση. Όταν λέμε πως μια ιστοσελίδα είναι στατική δεν εννοούμε πως τίποτα δεν κινείται, αντιθέτως μπορεί να περιλαμβάνει πολλά γραφικά, αλλά ότι το περιεχόμενο της είναι συγκεκριμένο. Από την άλλη οι δυναμικές ιστοσελίδες χρησιμοποιούν βάσεις δεδομένων απ' όπου αντλούν τα στοιχεία που θέλουν να εμφανίσουν. Έτσι η προσθαφαίρεση περιεχομένου είναι μια εύκολη υπόθεση.

Η κατασκευή των δυναμικών ιστοσελίδων μπορεί να είναι πιο πολύπλοκη σε σχέση με τις στατικές ιστοσελίδες, αλλά για την δημιουργία τους υπάρχουν διαθέσιμες πολλές τεχνολογίες. Οι πιο συνηθισμένες είναι οι ASP.NET, JSP, ColdFusion, Rudy on Rails, και PHP. Το περιεχόμενο της ιστοσελίδας, αποθηκεύεται και αντλείται δυναμικά από μία ή περισσότερες **βάσεις δεδομένων**, επιτρέποντας έτσι την εγγραφή και σύνδεση των χρηστών, την επίτευξη ηλεκτρονικού εμπορίου και παρόμοιες διαδικασίες. Αυτό είναι και το μεγάλο πλεονέκτημα που προσφέρουν έναντι των στατικών, δηλαδή την αλληλεπίδραση του χρήστη με τη σελίδα.

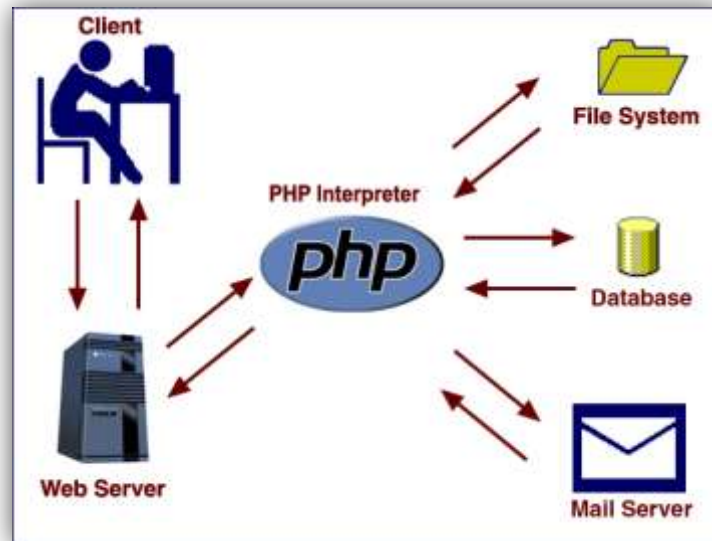
2.2.2 HTML

Τα αρχικά **HTML** (HyperText Markup Language) είναι μια γλώσσα σήμανσης δηλαδή αποτελούνται από λέξεις-κλειδιά, μέσα σε γωνιακές αγκύλες (όπως <html>). Οι ετικέτες HTML κανονικά έρχονται σε ζεύγη, όπως <h1> και </ h1>. Τα HTML στοιχεία αποτελούν τα δομικά στοιχεία του συνόλου των ιστοχώρων. Ο σκοπός ενός web browser είναι να διαβάσετε έγγραφα HTML και να τα συνθέσει σε οπτικό ή ακουστικό ιστοσελίδες. Το πρόγραμμα περιήγησης δεν εμφανίζει τις ετικέτες HTML, αλλά χρησιμοποιεί τις ετικέτες για να ερμηνεύσει το περιεχόμενο της σελίδας. Επίσης, παρέχει ένα μέσο για τη δημιουργία δομημένων εγγράφων υποδηλώνοντας δομική σημασιολογία για το κείμενο, όπως είναι οι κεφαλίδες, οι παράγραφοι, κατάλογοι, συνδέσεις, και άλλα στοιχεία. Είναι δυνατή η ενσωμάτωση scripts σε γλώσσες όπως JavaScript που επηρεάζουν τη συμπεριφορά των HTML ιστοσελίδων. Ακόμη όμως υπάρχουν διαφορές στην απεικόνιση κάποιας σελίδας από διαφορετικούς browsers ειδικά όταν περιλαμβάνονται και εφαρμογές Javascript.

Τα προεπιλεγμένες χαρακτηριστικά για κάθε στοιχείο της HTML ορίζονται στο πρόγραμμα περιήγησης, και τα χαρακτηριστικά αυτά μπορούν να τροποποιηθούν με πρόσθετη χρήση του σχεδιαστή ιστοσελίδα CSS. Τα φύλλα CSS παρέχουν τη δυνατότητα ορισμού κοινών κανόνων μορφοποίησης για ένα σύνολο ιστοσελίδων σε εξωτερικά αρχεία.

Τέλος, αξίζει να σημειωθεί ότι η html είναι η πρώτη και πιο διαδεδομένη γλώσσα περιγραφής της δομής μιας ιστοσελίδας. Παρ' όλα αυτά πολλοί είναι εκείνοι που δημιουργούν μια ιστοσελίδα χωρίς την συγγραφή κώδικα γι' αυτό και έχει δημιουργηθεί ειδικό λογισμικό, που επιτρέπει το δημιουργία μιας σελίδας μόνο οπτικά, χωρίς τη συγγραφή κώδικα, δίνοντας όμως τη δυνατότητα παρέμβασης και στον κώδικα. Ένα τέτοιο πρόγραμμα είναι το Dreamweaver της Adobe και το FrontPage της Microsoft.

2.2.3 PHP



Εικόνα 2.1: PHP

Η PHP είναι μια scripting γλώσσα προγραμματισμού σχεδιασμένη ειδικά για την παραγωγή δυναμικών ιστοσελίδων στο Διαδίκτυο. Η PHP ενσωματώνεται μέσα σε μια HTML και εκτελείται κάθε φορά που θα επισκεπτόμαστε μια σελίδα. Τα αρχεία που συμπεριλαμβάνουν κώδικα PHP θα πρέπει να έχουν την κατάλληλη επέκταση, όπως *.php, *.php4, *.phtml κ.ά. Όμως ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι κατάλληλα ρυθμισμένος έτσι ώστε να επεξεργαστεί τον κώδικα. Ο διακομιστής Apache, που χρησιμοποιείται στα λειτουργικά συστήματα GNU/Linux και Microsoft Windows, υποστηρίζει εξ ορισμού την εκτέλεση του κώδικα PHP. Επίσης, η PHP βρίσκεται στην πλευρά του διακομιστή. Αυτό σημαίνει πως οτιδήποτε κάνει ο κώδικας της PHP συμβαίνει στο διακομιστή.

Η PHP δημιουργήθηκε από ένα φοιτητή, τον Rasmus Lerdorf, το 1994, ο οποίος χρησιμοποίησε την γλώσσα προγραμματισμού Perl για να φτιάξει ένα απλό script. Υιοθετήθηκε και από άλλα άτομα και έχει περάσει από τρεις βασικές εκδόσεις. Αυτή τη στιγμή οι περισσότεροι ιστότοποι χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP. Σήμαινε *Personal Home Page* (προσωπική αρχική σελίδα), αλλά άλλαξε σύμφωνα με την σύμβαση GNU και τώρα σημαίνει *PHP Hypertext Preprocessor* (προεπεξεργαστής κειμένου PHP).

Η PHP διατίθενται σε όλους τους χρήστες χωρίς χρέωση από την σελίδα www.php.net.

Πλεονεκτήματα της PHP

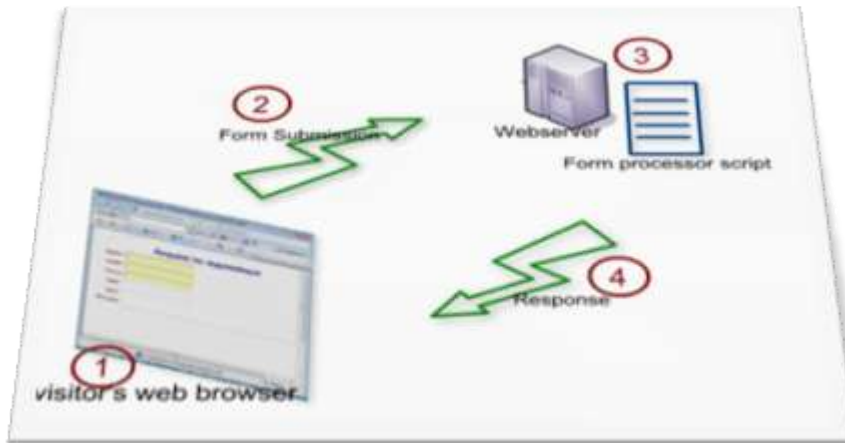
Πολλές γλώσσες υπάρχουν για την δημιουργία δυναμικών ιστοσελίδων: **Perl**, **Microsoft Active Server Pages (ASP)**, **Java Server Pages (JSP)** και **Allaire Cold Fusion**.

Η **PHP** έχει πολλά πλεονεκτήματα έναντι των παραπάνω γλωσσών προγραμματισμού, όπως :

- Υψηλή απόδοση (με έναν φτηνό server μπορούν να εξυπηρετηθούν εκατομμύρια χρήστες καθημερινά).
- Επιτυγχάνεται εύκολη συνεργασία με τους περισσότερους database servers.
- Ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες Web διαδικασίες (δυναμική δημιουργία εικόνων, αποστολή email, χειρισμός cookies).
- Χαμηλό κόστος (διατίθενται δωρεάν).

- Ευκολία μάθησης και χρήσης (η σύνταξη της μοιάζει με τις γλώσσες Java και C).
- Μεταφερσιμότητα (ο ίδιος κώδικας δουλεύει χωρίς αλλαγές και σε άλλο λειτουργικό σύστημα).
- Διαθεσιμότητα του κώδικα προέλευσης (μπορούν να πραγματοποιηθούν αλλαγές στη γλώσσα).

1. Δημιουργία Φορμών



Εικόνα 2.2: Επικοινωνία φυλλομετρητή με webserver.

Ο χειρισμός των φορμών με την γλώσσα PHP είναι πολύ σημαντική διεργασία σε μια δυναμική τοποθεσία. Πρώτα δημιουργούμε την φόρμα HTML, και έπειτα με ένα PHP σενάριο επεξεργαζόμαστε τα δεδομένα της φόρμας.

Μια φόρμα HTML δημιουργείται με την χρήση ετικετών form οι οποίες έχουν την εξής σύνταξη:

```
<form action="script.php" method="post">
</form>
```

Το πιο σημαντικό χαρακτηριστικό της ετικέτας είναι το action, που υπαγορεύει σε ποια σελίδα θα σταλούν τα δεδομένα. Το method χαρακτηριστικό καθορίζει τον τρόπο αποστολής των δεδομένων στη σελίδα που θα χειριστεί τα αποτελέσματα. Οι δύο επιλογές – get και post – αναφέρονται στη μέθοδο HTTP (Hypertext Transfer Protocol) που θα χρησιμοποιηθεί.

Η μέθοδος get στέλνει τα δεδομένα που έχει υποβάλλει ο χρήστης στη σελίδα-αποδέκτη ως μια σειρά ζευγών όνομα-τιμή προσαρμοσμένα στη διεύθυνση URL.

πχ. <http://www.example.com/script.php?name=Sofia&age=25>

Έτσι, η σελίδα script.php δέχεται τις τιμές των μεταβλητών mane και age. Γενικά η μέθοδος get χρησιμοποιείται για τη ζήτηση πληροφοριών, όπως είναι τα αποτελέσματα μιας αναζήτησης. Η μέθοδος post χρησιμοποιείται όταν απαιτείται από το χρήστη η εκχώρηση κάποια τιμής στα διάφορα πεδία της φόρμας.

Επίσης, υπάρχουν τα πλαίσια κειμένου (text fields) που επιτρέπουν στο χρήστη να καταχωρεί δεδομένα.

```
<form>
First name: <input type="text" name="firstname" /><br />
Last name: <input type="text" name="lastname" />
</form>
```

First name:
Last name:

Εικόνα 2.3: Text field

Καθώς και τα πλαίσια κειμένου password μέσω των οποίων εκχωρείται ο κωδικός του χρήστη στη βάση δεδομένων, αφού πρώτα κωδικοποιηθεί κατάλληλα από την συνάρτηση SH1() της PHP. Η συνάρτηση αυτή δημιουργεί μια κρυπτογραφημένη συμβολοσειρά μήκους ακριβώς 40 χαρακτήρων.

```
<form>
Password: <input type="password" name="pwd" size="40" maxlength="40" />
</form>
```

Password:

Εικόνα 2.4 : Δημιουργία πεδίου Password.

Τα ραδιοπλήκτρα (radio buttons) επιτρέπουν μια επιλογή, η οποία αποκλείει τις άλλες, καθώς η επιλογή ενός κουμπιού ακυρώνει οποιαδήποτε άλλη επιλογή. Έτσι, έχουν το ίδιο όνομα, αλλά διαφορετικές τιμές.

```
<form>
<input type="radio" name="sex" value="male" /> Male<br />
<input type="radio" name="sex" value="female" /> Female
</form>
```

Male
 Female

Εικόνα 2.5: radio buttons

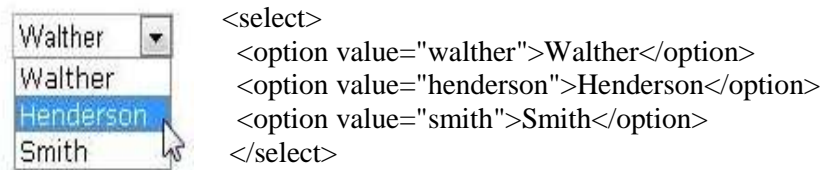
Τα πλαίσια ελέγχου (checkboxes) επιτρέπουν στους χρήστες να ενεργοποιήσουν όσες επιλογές θέλουν από μια λίστα επιλογών.

```
<form>
<input type="checkbox" name="vehicle" value="Bike" /> I have a bike<br />
<input type="checkbox" name="vehicle" value="Car" /> I have a car
</form>
```

I have a bike
 I have a car

Εικόνα 2.6: checkboxes

Το πτυσσόμενο μενού ξεκινάει με την ετικέτα `select`, και μετά κάθε ετικέτα `option` δημιουργεί μια ακόμη γραμμή στη λίστα των επιλογών.



Εικόνα 2.7: Πτυσσόμενο μενού

Μια περιοχή κειμένου (`textarea`) διαφέρει από το πλαίσιο κειμένου (`text`), γιατί παρουσιάζεται ως πλαίσιο και όχι ως μια γραμμή. Επιτρέπουν την πληκτρολόγηση πολλών πληροφοριών και είναι χρήσιμες για σχόλια του χρήστη.

```
<p><b>Comments:</b>
<textarea name="comments"
  rows="10" cols="50">
</textarea>
</p>
```



Εικόνα 2.8: Textarea

Τα κουμπιά υποβολής εκτελούν ενέργειες όταν πατιούνται. Οι προκαθορισμένες τιμές τους είναι “Submit” ή “Reset”. Μπορούμε όμως να τους ορίσουμε και άλλες εργασίες επεξεργασίας σε κώδικα.

```
<form name="input" action="html_form_action.asp" method="get">
  Username: <input type="text" name="user" />
  <input type="submit" value="Submit" />
</form>
```

Username:

Εικόνα 2.9: Δημιουργία buttons σε φόρμες

2. Επικύρωση δεδομένων φόρμας

Είναι πολύ σημαντικό να ελέγχουμε τα δεδομένα που καταχωρίζονται σε μια φόρμα για λόγους ασφάλειας και διαχείρισης σφαλμάτων. Κάθε σχεδιαστής μιας ιστοσελίδας είναι υπεύθυνος για την σωστή λειτουργία κάθε φόρμας που εμπεριέχεται σε ένα ιστοχώρο. Για την επικύρωση των δεδομένων μιας φόρμας απαιτείται η χρήση συνθηκών, συναρτήσεων, τελεστών, και παραστάσεων. Μία συνάρτηση που χρησιμοποιείται συνήθως είναι η `isset()`, που ελέγχει αν έχει δοθεί τιμή σε μία μεταβλητή. Ένα πρόβλημα της συνάρτησης `isset()` είναι ότι δεν μπορεί να διακρίνει αν μια συμβολοσειρά είναι κενή ή όχι. Για να ελέγξουμε, λοιπόν, αν ο χρήστης έχει δώσει τιμή σε ένα πεδίο κειμένου, χρησιμοποιούμε τη συνάρτηση `empty()`, η οποία ελέγχει αν μια μεταβλητή έχει κενή (`empty`) τιμή.

Κατά την επικύρωση μιας φόρμας πρώτα ελέγχουμε αν ο χρήστης καταχωρήσει ή επιλέξει κάποιο στοιχείο, αν υπάρχει κάποια λίστα τιμών. Έπειτα πρέπει να εξασφαλίσουμε ότι τα υποβαλλόμενα δεδομένα έχουν τον κατάλληλο τύπο, την σωστή μορφή, ή είναι κάποια συγκεκριμένη αποδεκτή τιμή.

3. Ενσωμάτωση PHP στην HTML



Εικόνα 2.10: Ενσωμάτωση PHP στην HTML

Το “σώμα” της HTML καθορίζεται από την ετικέτα body. Η PHP, όπως είπαμε, ενσωματώνεται στην HTML, δηλαδή αναμιγνύουμε κώδικα PHP και HTML μέσα στο ίδιο αρχείο. Για να προσθέσουμε τον κώδικα PHP πρέπει να τον γράψουμε ανάμεσα στις ετικέτες PHP:

```
<?php
```

```
?>
```

Ό,τι βρίσκεται μέσα στις ετικέτες αυτές ο διακομιστής το θεωρεί κώδικας PHP. Αλλιώς, αν βρίσκεται δηλαδή έξω από τις ετικέτες θεωρείται ως κώδικας HTML. Για να στείλει η PHP τα δεδομένα στο φυλλομετρητή χρησιμοποιεί τις συναρτήσεις echo() και print().

4. Πίνακες

Μέσα σε ένα πίνακα έχουμε τη δυνατότητα να αποθηκεύσουμε πολλές ξεχωριστές πληροφορίες. Σαν αποτέλεσμα προκύπτει μια λίστα τιμών, κάθε μια από τις οποίες μπορεί να είναι συμβολοσειρά, αριθμός, ή ένας άλλος πίνακας. Η PHP υποστηρίζει δύο είδη πινάκων: *αριθμοδεικτών*, που χρησιμοποιούν αριθμούς ως κλειδιά και *συνειρμικούς*, που χρησιμοποιούν ως κλειδιά συμβολοσειρές.

Για να αναφερθούμε σε μια συγκεκριμένη τιμή ενός πίνακα γράφουμε πρώτα το όνομα του πίνακα και έπειτα το κλειδί μέσα σε αγκύλες:

```
echo $authors[0];
echo $books['BK'];
```

Η εκτύπωση ενός πίνακα είναι πιο δύσκολη σε σχέση με εκείνη των συμβολοσειρών λόγω της διαφορετικής σύνταξης. Αν θέλουμε να εκτυπώσουμε ένα μεμονωμένο στοιχείο γράφουμε:

```
echo "Ο συγγραφέας αυτού του βιβλίου είναι ο $author[0]";
```

Αν όμως θέλουμε να εκτυπώσουμε ένα πίνακα που χρησιμοποιεί συμβολοσειρές ως κλειδιά, πρέπει να κλείσουμε σε άγκιστρα το όνομα του πίνακα.


```
echo "Ο συγγραφέας αυτού του βιβλίου είναι ο {$author['AT']}";
```

Υπερκαθολικοί Πίνακες

Υπερκαθολικές μεταβλητές ονομάζονται οι έτυμοι πίνακες που διαθέτει η PHP: `$_GET`, `$_POST`, `$_REQUEST`, `$_SERVER`, `$_ENV`, `$_SESSION` και `$_COOKIE`.

Στη μεταβλητή `$_GET`, αποθηκεύονται οι τιμές που μέσω της μεθόδου `get` μεταβιβάζονται σε ένα άλλο σενάριο της PHP. Στη μεταβλητή `$_POST` αποθηκεύονται τα δεδομένα που η μέθοδος `post` μεταβιβάζει σε ένα σενάριο PHP από μια φόρμα HTML. Οι δύο αυτές μεταβλητές, μαζί με την `$_COOKIE` αποτελούν υποσύνολα της μεταβλητής `$_REQUEST`. Στην μεταβλητή `$_SESSION` αποθηκεύονται τα δεδομένα μιας περιόδου εργασίας. Με την μεταβλητή `$_COOKIE` μπορούμε να ανακτήσουμε την τιμή ενός μπισκότου. Τέλος, στις μεταβλητές `$_SERVER` και `$_ENV` αποθηκεύονται πληροφορίες σχετικά με τον διακομιστή.

5. Χρήση cookies

Μέσω των μπισκότων δίνεται η δυνατότητα στον διακομιστή της αποθήκευσης πληροφοριών στον υπολογιστή του χρήστη. Έτσι μια τοποθεσία μπορεί να θυμάται τον χρήστη που την επισκέπτεται. Πολλοί χρήστες νομίζουν πως μέσω την χρήση μπισκότων ο διακομιστή θα μπορεί να έχει πρόσβαση σε προσωπικά τους στοιχεία. Αλλά, μέσα σε ένα μπισκότο αποθηκεύονται μόνον οι πληροφορίες τι οποίες δίνει ο χρήστης.

Είναι πολύ σημαντικό τα μπισκότα να στέλνονται από το διακομιστή στον πελάτη πριν από οποιαδήποτε άλλη πληροφορία. Γι' αυτό το λόγο πρέπει να ρυθμίζεται ο φυλλομετρητής για την εργασία που πρέπει να εκτελέσει όταν δέχεται ένα μπισκότο. Έτσι, ο φυλλομετρητής αφότου λάβει ένα μπισκότο από την PHP θα ενημερώνει πρώτα για τις πληροφορίες που περιέχει αυτό.

Η συνάρτηση μέσω της οποίας στέλνονται τα μπισκότα ονομάζεται `setcookie()`. Η συνάρτηση αυτή είναι μια από τις λίγες συναρτήσεις της PHP που μπορούν να έχουν διαφορετικά αποτελέσματα σε διαφορετικούς φυλλομετρητές.

```
πχ. setcookie(όνομα, τιμή);  
setcookie('user_name', 'sof');
```

Μέσω της υπερκαθολικής μεταβλητής `$_COOKIE` γίνεται η ανάκτηση του μπισκότου:
`$_COOKIE['user_name']`

Επίσης, η συνάρτηση `setcookie()` δέχεται μέχρι πέντε επιπλέον παραμέτρους:

```
setcookie(όνομα, τιμή, λήξη, διαδρομή, υπολογιστής_υπηρεσίας, ασφάλεια, αποκλειστικά_http);
```

Το όρισμα λήξης καθορίζει τη διάρκεια ζωής του μπισκότου σε δευτερόλεπτα από την εποχή του Unix.

```
πχ. setcookie(όνομα, τιμή, time()+1800);
```

Η παραπάνω γραμμή κώδικα καθορίζει το χρόνο λήξης του μπισκότου σε 30 λεπτά.

6. Χρήση περιόδων εργασίας

Μια άλλη μέθοδος για την διάθεση των δεδομένων σε πολλές σελίδες είναι οι περίοδοι εργασίας. Με την χρήση περιόδων εργασίας τα δεδομένα δεν αποθηκεύονται στο φυλλομετρητή, αλλά στο διακομιστή. Μέσω του χαρακτηριστικού περιόδου εργασίας μπορούμε να εντοπίσουμε τα

δεδομένα εγγραφής ενός χρήστη. Το αναγνωριστικό αυτό αποθηκεύεται στο φυλλομετρητή του χρήστη μέσω ενός μπισκότου, αλλά τα ευαίσθητα δεδομένα παραμένουν πάντα στο διακομιστή.

Για να χρησιμοποιήσει μια σελίδα τις περιόδους εργασίας πρέπει να καλέσει την συνάρτηση `session_start()` η οποία πληροφορεί την PHP ώστε να ξεκινήσει μια νέα περίοδο εργασίας ή να προσπελάσει μια υπάρχουσα. Όταν η συνάρτηση αυτή καλείται για πρώτη φορά προσπαθεί να στείλει ένα μπισκότο στο φυλλομετρητή με όνομα PHPSESSID (το όνομα της περιόδου εργασίας) γι' αυτό και πρέπει να καλείται πριν από την αποστολή οποιονδήποτε δεδομένων.

Σύνταξη μητρώου:

```
$_SESSION['κλειδί'] = τιμή;  
$_SESSION['name'] = 'Sofia';  
$_SESSION['id'] = 10;
```

Διαγραφή μιας περιόδου εργασίας:
`unset($_SESSION['μεταβλητή']);`

Διαγραφή όλων μεταβλητών των περιόδων εργασίας:
`$_SESSION = array();`

Αφαίρεση όλων των δεδομένων της περιόδου εργασίας από το διακομιστή:
`session_destroy();`

Για να χρησιμοποιηθεί κάποια από αυτές τις μεθόδους η σελίδα πρέπει να ξεκινάει με την κλήση της `session_start()`.

Περίοδοι εργασίας έναντι μπισκότων

- Οι περίοδοι εργασίας είναι πιο ασφαλείς, γιατί οι πληροφορίες παραμένουν αποθηκευμένες στο διακομιστή.
- Με τη χρήση περιόδων εργασίας μπορούμε να αποθηκεύσουμε περισσότερα δεδομένα.
- Οι χρήστες συνηθίζουν να απενεργοποιούν τα cookies. Οι περίοδοι εργασίας όμως μπορούν να λειτουργήσουν και χωρίς τα cookies.

7. Συμπερίληψη πολλών αρχείων

Η PHP χρησιμοποιώντας εξωτερικά αρχεία, διαιρεί τα σενάρια της τοποθεσίας σε διακριτά τμήματα. Έτσι ο κώδικας HTML αποσπάται από τον κώδικα PHP, ή ακόμη και κομμάτια του κώδικα PHP αποσπώνται ένα σενάριο. Η συμπερίληψη των αρχείων γίνεται με την χρήση των παρακάτω συναρτήσεων: `include()`, `include_once()`, `require()`, `require_once()`.

Πχ. `include_once('filename.php');`
`require('path/to/filename.html');`

Για να αναφερθούμε στο αρχείο συμπερίληψης μπορούμε να χρησιμοποιήσουμε είτε απόλυτη, είτε σχετική διαδρομή. Η απόλυτη διαδρομή καθορίζει τη θέση του αρχείου ξεκινώντας από το βασικό κατάλογο του υπολογιστή.

πχ. `include('C:/php/includes/file.php');`

Η σχετική διαδρομή ξεκινάει από τη θέση του γενικού αρχείου, δηλαδή αυτό που καλεί το εξωτερικό. Για να ανεβούμε κατά ένα φάκελο στη ιεραρχία, χρησιμοποιούμε δύο τελείες (`..`). Για να μεταφερθούμε μέσα σε ένα φάκελο, χρησιμοποιούμε το όνομα του φακέλου που ακολουθεί από μια κάθετο. Για παράδειγμα, αν το σενάριο βρίσκεται στο φάκελο `www/file1` και θέλουμε να συμπεριλάβουμε ένα αρχείο που βρίσκεται στο φάκελο `www/file2`, γράφουμε:

```
include('../file2/example.php');
```

Οι συναρτήσεις include() και require() διαφέρουν στη λειτουργία τους μόνο όταν αποτυγχάνουν. Αν μια συνάρτηση include() δεν επιτύχει, το σενάριο θα εξακολουθήσει να εκτελείτε κανονικά και απλά στο φυλλομετρητή θα εμφανιστεί μια προειδοποίηση. Αν αποτύχει η require(), το σενάριο θα διακοπεί εμφανίζοντας ένα μήνυμα σφάλματος.

Επίσης, όταν γράφουμε και το χαρακτηριστικό *_once(), τότε το εξωτερικό αρχείο συμπεριλαμβάνεται στον κώδικα μόνο μία φορά.

```
πχ. require_once('filename.php');  
include_once('filename.php');
```

8. Προσωρινή αποθήκευση εξόδου

Με την χρήση της προσωρινής αποθήκευσης εξόδου ότι τυπώνεται σε ένα σενάριο της PHP ή στον κώδικα HTML δεν στέλνεται απευθείας στον φυλλομετρητή, αλλά αποθηκεύεται σε μία προσωρινή μνήμη. Έτσι οι συναρτήσεις echo() και print() στέλνουν τα δεδομένα στην προσωρινή μνήμη και όχι στο φυλλομετρητή.

Για να ξεκινήσουμε την προσωρινή αποθήκευση, χρησιμοποιούμε τη συνάρτηση ob_start(). Στο τέλος του σεναρίου, καλούμε την συνάρτηση ob_end_flush() για να στείλουμε στο φυλλομετρητή τα δεδομένα. Για να διαγράψουμε τα δεδομένα αυτά καλούμε την συνάρτηση ob_end_clean() χωρίς να τα στείλουμε στο φυλλομετρητή.

2.2.4 Σχεδίαση Βάσεων δεδομένων

Πριν ξεκινήσουμε την σχεδίαση μιας βάσης δεδομένων πρέπει πρώτα να καθορίσουμε τη δομή της. Η σχεδίαση μιας βάσης δεδομένων είναι αυτή που καθορίζει την επιτυχημένη μακροπρόθεσμη διαχείριση των πληροφοριών. Χρησιμοποιώντας την *κανονικοποίηση των δεδομένων* εξαλείφονται οι πλεονασμοί και άλλα προβλήματα που θα μπορούσαν να βλάψουν την ακεραιότητα της βάσης δεδομένων.

1. Κανονικοποίηση

Κανονικοποίηση είναι η διαδικασία οργάνωσης μιας βάση δεδομένων. Ο ερευνητής E. F. Codd ανακάλυψε την σχεσιακή βάση δεδομένων (μια συλλογή δεδομένων οργανωμένων με συγκεκριμένο τρόπο). Επίσης, δημιούργησε μια σειρά από κανόνες, τις *κανονικές μορφές* (normal forms), που συντελούν στην οργάνωση των δεδομένων.

Πριν αρχίσουμε την κανονικοποίηση πρέπει πρώτα να ορίσουμε τον σκοπό της εφαρμογής. Η κατανόηση του τρόπου προσπέλασης των δεδομένων καθορίζει τη μοντελοποίηση. Στη συγκεκριμένη ιστοσελίδα θα πρέπει οι χρήστες να εγγράφονται και έπειτα να συνδέονται με το email και το κωδικό πρόσβασής τους. Έπειτα θα προσθέτουν όποιο βιβλίο επιθυμούν στο καλάθι αγορών τους, ή στη λίστα με τα αγαπημένα τους βιβλία. Επίσης, ο διαχειριστής θα προσθέτει τα βιβλία σε ένα πίνακα με τα ανάλογα χαρακτηριστικά από όπου θα εμφανίζονται στην ιστοσελίδα. Για την λειτουργία του forum είναι αναγκαία η ύπαρξη ενός πίνακα ο οποίος θα αποθηκεύει τα μηνύματα των χρηστών.

Μια καλή μέθοδος για τον προσδιορισμό των πληροφοριών που θα αποθηκεύονται στη βάση δεδομένων είναι ο προσδιορισμός των ερωτημάτων που θα υποβάλλονται και των δεδομένα που θα περιλαμβάνονται στις απαντήσεις των ερωτημάτων αυτών.

2. Κλειδιά

Τα κλειδιά παίζουν σημαντικό ρόλο σε μία κανονικοποιημένη βάση δεδομένων, γιατί μέσω αυτών γίνεται η συσχέτιση των πινάκων. Υπάρχουν δύο είδη κλειδιών: το *πρωτεύων* κλειδί (primary key) και το *ξένο* κλειδί (foreign key).

Πρωτεύων κλειδί είναι ένα πεδίο ή συνδυασμός πεδίων που χαρακτηρίζει μοναδικά μια εγγραφή. Πρωτεύων κλειδί πρέπει να έχει πάντα τιμή (να μην είναι δηλαδή NULL), η τιμή του να μην αλλάζει ποτέ και να είναι μοναδική για κάθε εγγραφή του πίνακα. Το καλύτερο παράδειγμα πρωτεύοντος κλειδιού είναι ο αριθμός φορολογικού μητρώου (ΑΦΜ). Συχνά πρέπει να δημιουργούμε ένα πρωτεύων κλειδί, επειδή κανένα από τα υπάρχοντα πεδία δεν είναι κατάλληλο. Η MySQL επιτρέπει μόνο ένα πρωτεύων κλειδί ανά πίνακα. Στην ιδανική περίπτωση, πρέπει να είναι ακέραιος αριθμός ώστε να υπάρχει καλύτερη απόδοση.

Το ξένο κλειδί είναι ένα πεδίο που έχει το ίδιο σύνολο τιμών με το πρωτεύων κλειδί ενός άλλου πίνακα. Για να μπορούν δύο πίνακες να συνδέονται, πρέπει το πρωτεύων κλειδί του ενός να συνδέεται με το ξένο κλειδί του άλλου.

3. Σχέσεις

Στις *Σχεσιακές* βάσεις δεδομένων τα δύο πίνακες συνδέονται μεταξύ τους μέσω των κοινών πεδίων που διαθέτουν. Οι σχέσεις των πινάκων χωρίζονται σε τρεις κατηγορίες:

- Αμφιμονοσήμαντη ή ένα προς ένα.
- Μονοσήμαντη ή ένα προς πολλά.
- Πολυσήμαντη ή πολλά προς πολλά.

Σε μία αμφιμονοσήμαντη σχέση, ένα μοναδικό στοιχείο του Πίνακα Α αντιστοιχεί σε ένα μοναδικό στοιχείο του πίνακα Β. Για παράδειγμα, κάθε χρήστης έχει ένα μοναδικό κωδικό και κάθε κωδικός αντιστοιχεί σε έναν μόνο χρήστη.

Σε μια μονοσήμαντη σχέση, ένα στοιχείο του Πίνακα Α μπορεί να αντιστοιχεί σε πολλά στοιχεία του πίνακα Β. Για παράδειγμα, ένας βιβλίο μπορεί να έχει έναν εκδότη και ένας εκδότης μπορεί να έχει πολλά βιβλία.

Σε μια πολυσήμαντη σχέση, πολλά στοιχεία του πίνακα Α αντιστοιχούν σε πολλά στοιχεία του πίνακα Β. Για παράδειγμα, ένα βιβλίο μπορεί να περιέχεται σε διάφορα καλάθια αγορών και κάθε καλάθι αγοράς μπορεί να έχει πολλά βιβλία. Για λόγους πλεονασμού των δεδομένων και είναι προτιμότερο να αποφεύγουμε τις πολυσήμαντες σχέσεις και να τις διαιρούμε σε δύο μονοσήμαντες χρησιμοποιώντας *ενδιάμεσους πίνακες*.

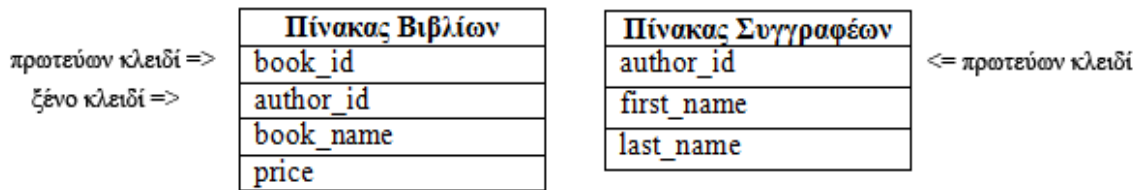
4. Πρώτη Κανονική Μορφή (1FN)

Για να συμφωνεί η βάση δεδομένων με την Πρώτη Κανονική Μορφή κάθε στήλη ενός πίνακα θα πρέπει να περιέχει μόνο μία τιμή. Επίσης, κανένας πίνακας δεν μπορεί να περιέχει επαναλαμβανόμενες ομάδες σχετικών μεταξύ τους δεδομένων. Ένας πίνακας με ένα πεδίο για ολόκληρη την ταχυδρομική διεύθυνση ενός ατόμου δεν συμφωνεί με την πρώτη κανονική μορφή, επειδή μια στήλη του περιέχει πολλές τιμές. Επίσης, ένας πίνακας βιβλίων με πολλές στήλες για τους συγγραφείς δεν είναι σύμφωνος με την πρώτη κανονική μορφή, επειδή περιέχει επαναλαμβανόμενες στήλες με ίδιες πληροφορίες. Έτσι αυτές θα πρέπει να μετατραπούν σε ξεχωριστό πίνακα.

5. Δεύτερη Κανονική Μορφή (2NF)

Για να είναι μια βάση δεδομένων στη Δεύτερη Κανονική Μορφή, πρέπει να είναι ήδη στην πρώτη. Έπειτα θα πρέπει κάθε στήλη του πίνακα να εξαρτάται από το πρωτεύων κλειδί. Για παράδειγμα, στον πίνακα των βιβλίων ο ένας συγγραφέας μπορεί να εμφανίζεται πολλές φορές. Για να ισχύει ο κανόνας 2NF η λύση είναι η δημιουργία ενός ξεχωριστού πίνακα συγγραφέων που θα

αναθέτει ένα πρωτεύων κλειδί σε κάθε συγγραφέα. Για την συσχέτιση των δύο πινάκων το πρωτεύων κλειδί του κάθε συγγραφέα θα γίνεται ξένο κλειδί στον πίνακα των βιβλίων.



Εικόνα 2.11: Διαμόρφωση Πινάκων με βάση τον κανόνα 2NF

6. Τρίτη Κανονική Μορφή (3NF)

Αυτό το υψηλότερο επίπεδο κανονικοποίησης πολλές φορές δεν είναι απαραίτητο, γιατί η βάση δεδομένων χάνει την απλότητά της. Για να είναι μια βάση δεδομένων στην Τρίτη Κανονική Μορφή πρέπει να είναι ήδη στη Δεύτερη Κανονική Μορφή και όλες οι στήλες που δεν είναι κλειδιά να είναι ανεξάρτητες η μια από την άλλη. Δηλαδή, όταν αλλάζουμε την τιμή μιας στήλης να μη χρειάζεται να αλλάζουμε και την τιμή μιας άλλης. Ένα παράδειγμα είναι ο πίνακας των χρηστών, στον οποίο έχουμε πολλές πληροφορίες που εξαρτώνται μεταξύ τους, όπως την Πόλη, την Διεύθυνση, τον Τ.Κ. Αυτές οι τιμές είναι απαραίτητες η μια για την άλλη, αλλά όχι για τον χρήστη που αντιστοιχεί στην εγγραφή. Για να ισχύσει ο κανόνας 3NF πρέπει αυτές οι πληροφορίες να εκχωρηθούν σε διαφορετικό πίνακα.

2.2.5 Κανονικές Παραστάσεις συμβατές με Perl

Οι κανονικές παραστάσεις θεωρείται ως ένα εξελιγμένο σύστημα ταυτοποίησης μοτίβων το οποίο είναι διαθέσιμο στις περισσότερες σύγχρονες γλώσσες προγραμματισμού. Χρησιμοποιώντας ενσωματωμένες συναρτήσεις της PHP γράφουμε πρώτα το μοτίβο και έπειτα το εφαρμόζουμε σε μια συμβολοσειρά. Η πιο σημαντική από τις συναρτήσεις αυτές είναι η `preg_match()`. Η συγκεκριμένη συνάρτηση επιστρέφει τιμή 0 ή 1, δηλώνοντας έτσι αν το μοτίβο ταιριάζει με τη συμβολοσειρά ή όχι.

`preg_match` (μοτίβο, υποκείμενο);

Η συνάρτηση `preg_match()` θα σταματάει όταν εντοπίζει την πρώτη συμβολοσειρά που ταιριάζει με το συγκεκριμένο μοτίβο που δηλώνουμε. Για να βρούμε όλες τις συμβολοσειρές που ταιριάζουν, χρησιμοποιούμε τη συνάρτηση `preg_match_all()`.

Επειδή το μοτίβο πρόκειται για συμβολοσειρά όταν το μεταβιβάζουμε στη συνάρτηση `preg_match()`, πρέπει να το κλίνουμε σε εισαγωγικά. Έπειτα το μοτίβο πρέπει να περικλείεται μέσα σε οριοθέτες. Οριοθέτης μπορεί να είναι οποιοσδήποτε χαρακτήρας εκτός από συμβολοσειρά και ανάποδη κάθετο. Ο ίδιος οριοθέτης πρέπει να χρησιμοποιείται στην αρχή και στο τέλος του μοτίβου.

πχ. `if (preg_match ('/a', 'cat')) {`

Αν το μοτίβο περιέχει κάθετο τότε σαν οριοθέτη χρησιμοποιούμε το χαρακτήρα διαφυγής (`\`).

Μεταχαρακτήρες	
ΧΑΡΑΚΤΗΡΑΣ	ΣΗΜΑΣΙΑ
<code>\</code>	Χαρακτήρας Διαφυγής
<code>^</code>	Αρχή Συμβολοσειράς
<code>\$</code>	Τέλος Συμβολοσειράς
<code>.</code>	Οποιοσδήποτε χαρακτήρας εκτός από την αλλαγή γραμμής

	Εναλλακτικό (ή)
[Αρχή κλάσης
]	Τέλος κλάσης
(Αρχή υπομοτίβου
)	Τέλος υπομοτίβου
{	Αρχή ποσοδείκτη
}	Τέλος ποσοδείκτη

Πίνακας 2.1: Μεταχαρακτήρες

Δύο μεταχαρακτήρες, η αιχμή(^) και το δολάριο(\$), καθορίζουν τη θέση της συμβολοσειράς έτσι ώστε να υπάρχει ταύτιση νε το μοτίβο. Για παράδειγμα, το μοτίβο ^a ταιριάζει με οποιαδήποτε συμβολοσειρά αρχίζει από a, ενώ το μοτίβο a\$ ταιριάζει με οποιαδήποτε συμβολοσειρά τελειώνει σε a. Οι ποσοδείκτες δηλώνουμε τον αριθμό των χαρακτήρων που μπορεί να έχει μια συμβολοσειρά.

Ποσοδείκτες	
ΧΑΡΑΚΤΗΡΕΣ	ΕΜΦΑΝΙΣΕΙΣ
?	0 ή 1
*	0 ή περισσότερες
+	1 ή περισσότερες
{x}	Ακριβώς x εμφανίσεις
{x, y}	Μεταξύ x και y
{x,}	Τουλάχιστον x εμφανίσεις

Πίνακας 2.2: Ποσοδείκτες

Για την απλοποίηση των αναφορών, χρησιμοποιούμε κλάσεις χαρακτήρων. Οι κλάσεις δημιουργούνται όταν τοποθετώντας χαρακτήρες μέσα σε αγκύλες. Για παράδειγμα, η κλάση [a-z] σημαίνει όλα τα πεζά γράμματα και η κλάση [A-Z] όλα τα κεφαλαία. Η κλάση [A-Za-z] σημαίνει οποιοδήποτε γράμμα, και η κλάση [0-9] οποιοδήποτε ψηφίο. Επίσης, η αιχμή στην αρχή της κλάσης λειτουργεί ως τελεστής άρνησης.

Μεταχαρακτήρες		
ΚΛΑΣΗ	ΣΥΝΤΟΜΕΥΣΗ	ΣΗΜΑΣΙΑ
[0-9]	\d	Οποιοδήποτε ψηφίο
[\fr t n v]	\s	Οποιοδήποτε χαρακτήρας
[A-Za-z0-9_]	\w	Οποιοδήποτε χαρακτήρας λέξης
[^0-9]	\D	Όχι ψηφίο
[^\fr t n v]	\S	Μη κενός χώρος
[^A-Za-z0-9_]	\W	Όχι χαρακτήρας λέξης

Πίνακας 2.3: Μεταχαρακτήρες

2.2.6 Δομή Τοποθεσίας

Σε μια τοποθεσία Ιστού, όταν χρησιμοποιούμε πολλά αρχεία, η συνολική της δομή αποκτά μεγάλη σημασία. Για το λόγο αυτό όταν διαμορφώνουμε μια ιστοσελίδα πρέπει να τις παρέχουμε ευκολία στην πλοήγηση και στην συντήρησή της, αλλά πάνω απ' όλα ασφάλειά της. Η χρήση εξωτερικών αρχείων στους κώδικες τις PHP, CSS, JavaScript, και HTML διευκολύνουν τη συντήρηση της τοποθεσίας, αφού ο κώδικας προς επεξεργασία βρίσκεται πάντα σε μια κεντρική θέση.

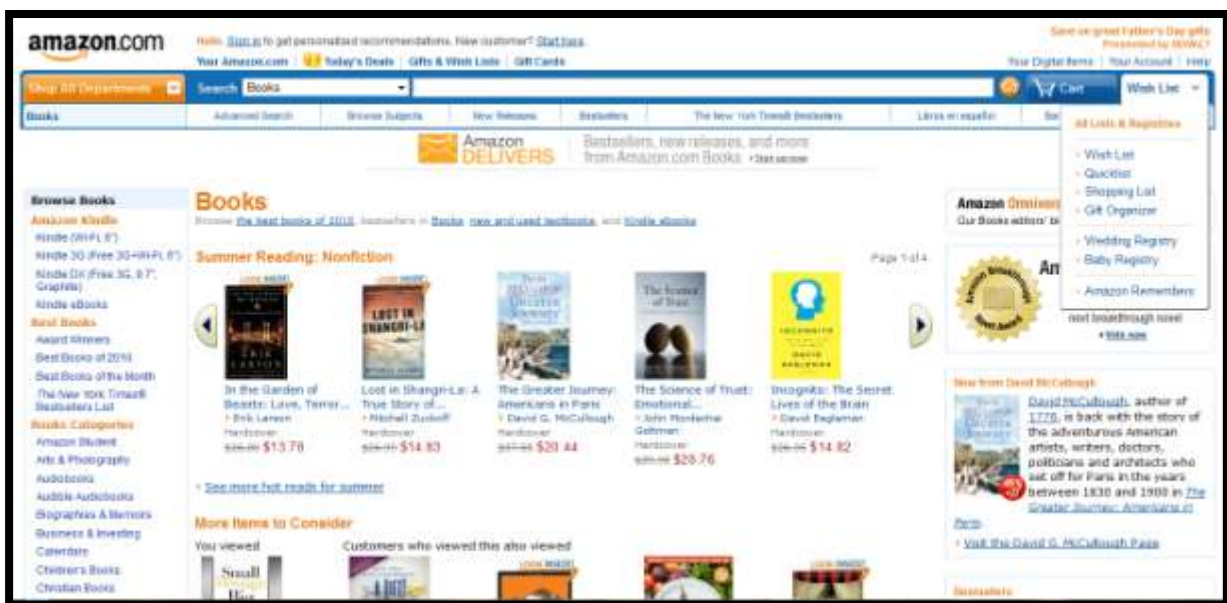
Συνήθως, δημιουργούμε έναν κατάλογο `includes` στον οποίο αποθηκεύονται τα αρχεία, ξεχωριστά από τα βασικά σενάρια.

Τέλος, μπορούμε να χρησιμοποιούμε την προέκταση `.inc` ή `.html` για έγγραφα που δεν μας απασχολεί τόσο πολύ η ασφάλειά τους, και την προέκταση `.php` για αρχεία που περιέχουν πιο ευαίσθητα δεδομένα.

ΚΕΦΑΛΑΙΟ 3: ΣΧΕΔΙΟ ΔΡΑΣΗΣ ΓΙΑ ΤΗΝ ΕΚΠΟΝΗΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ

3.1 State of Art στο χώρο των ηλεκτρονικών βιβλιοπωλείων.

Ένα από τα καλύτερα και πιο διαδεδομένα ηλεκτρονικό βιβλιοπωλείο στον κόσμο είναι το Amazon.com λόγω του πλήθους των δεδομένων και υπηρεσιών που προσφέρει στους χρήστες. Τον Ιούλιο του 2002 χρησιμοποιώντας το Διαδίκτυο μετέτρεψε την αγορά του βιβλίου σε μια γρήγορη και ευχάριστη εμπειρία. Είναι η πιο δημοφιλής εταιρία στην οποία εφαρμόστηκε το ηλεκτρονικό εμπόριο, γι' αυτό και σήμερα αποτελεί τον πιο κορυφαίο εμπορικό κόμβο.



Εικόνα 3.1: Αρχική σελίδα εμφάνισης βιβλίων

Όπως φαίνεται και στην **εικόνα** η εταιρεία Amazon δίνει την δυνατότητα πρόσβασης στο χρήστη σε μία τεράστια συλλογή βιβλίων, διαφόρων κατηγοριών, οι οποίες αναρτώνται στην αριστερή μεριά της ιστοσελίδας. Επίσης, έχουν δημιουργηθεί διάφορες λίστες (**Εικόνα**) στις οποίες ο χρήστης μπορεί να προσθέσει τα βιβλία που επιθυμεί. Καθώς επίσης και ένα καλάθι αγορών για την επίτευξη κάποιας παραγγελίας.



Bestsellers Bestsellers Archive Hot New Releases Movers & Shakers Most Gifted Most Wanted For

Any Category

Books

- 4-for-3 Books
- Arts & Photography
- Audible Audiobooks
- Bargain Books
- Biographies & Memoirs
- Books on Cassette
- Books on CD
- Business & Investing
- Calendars
- Children's Books
- Comics & Graphic Novels
- Computers & Internet
- Cooking, Food & Wine
- e-Docs
- Entertainment
- Gay & Lesbian
- Health, Mind & Body
- History
- Home & Garden
- Large Print
- Libros en español
- Literature & Fiction
- Mystery & Thrillers
- Nonfiction
- Oprah®
- Outdoors & Nature
- Parenting & Families
- Professional & Technical
- Reference
- Religion & Spirituality
- Romance
- Science
- Science Fiction & Fantasy
- Sports
- Teens
- Travel

Hot New Releases in Books

The bestselling new & future releases in Books. [Learn more](#)

- 


Put Your Mindset to Work: The One Asset You Really Need to Win and Keep the Job You Love
by James Reed, Paul G. Stoltz
★★★★☆ (6)
Release Date: May 31, 2011
List Price: \$17.00
Price: \$9.35
You Save: \$7.65 (45%)
35 used & new from \$7.90
- 

Those Guys Have All the Fun: Inside the World of ESPN
by James Andrew Miller, Tom Shales
★★★★☆ (9) | 3 customer discussions
Release Date: May 24, 2011
List Price: \$27.99
Price: \$14.91
You Save: \$13.08 (47%)
25 used & new from \$14.28
- 

In the Garden of Beasts: Love, Terror, and an American Family in Hitler's Berlin
by Erik Larson
★★★★☆ (112)
Release Date: May 10, 2011
List Price: \$26.00
Price: \$13.78
You Save: \$12.22 (47%)
85 used & new from \$12.97
- 

Incognito: The Secret Lives of the Brain
by David Eagleman
★★★★☆ (28)
Release Date: May 31, 2011
List Price: \$26.95
Price: \$14.82
You Save: \$12.13 (45%)
27 used & new from \$14.82

Εικόνα 3.2: Λίστες Βιβλίων του Amazon.com



Wish List Add-on for Firefox
Keep track of what you want. Add anything from any website to your Amazon Wish List.

[Install](#) [Learn more](#)

epp1545 [Show list profile](#)



[Upload](#)

Your Public Wish Lists
You don't have any public lists.
[Create a list your friends can see](#)

Your Private Wish Lists

Sofia's Wishlist
2 items (default)

[Create another Wish List](#)

Wish List Tips

- Wish from any website
- Set up Universal Wish List
- Make your list easy to find
- Add your photo
- Describe yourself

Sofia's Wishlist Saved

Only you can see this list. ([Change settings](#))

[Manage This List](#) [Print This List](#)

Page 1 of 1 (2 items)



How to Get Your Point Across in 30 Seconds or Less
by Milo O. Frank (Paperback)

~~\$12.00~~ **\$8.97**
In Stock. Offered by Amazon.com

139 Used & New from **\$0.01**

★★★★☆ (38 customer reviews)

[Add to Cart](#) [Move to a new list](#) | [Delete item](#)



Give Your Elevator Speech a Lift!
by Lorraine Howell (Paperback)

~~\$12.95~~ **\$10.15**
In Stock. Offered by Amazon.com

28 Used & New from **\$5.49**

★★★★☆ (10 customer reviews)

[Add to Cart](#) [Move to a new list](#) | [Delete item](#)

Εικόνα 3.3: Εμφάνιση βιβλίων στη Wishlist

Εκτός από την προσθήκη των βιβλίων στη λίστα με τα αγαπημένα βιβλία ο χρήστης μπορεί να μετονομάσει την λίστα του δίνοντάς της ένα πιο προσωπικό όνομα (Sofia's Wishlist), να προσθέσει τα βιβλία στο καλάθι του, αλλά και να δημιουργήσει μια καινούρια λίστα από το link (Create another Wish list). Τέλος, έχει δημιουργηθεί εφαρμογή αποκλειστικά για το Firefox έτσι ώστε να έχει ο κάθε χρήστης πρόσβαση στη Wishlist όποτε εκείνος επιθυμεί.

 **Shopping Cart** for epp1545 ([if you're not epp1545, click here](#))

[See more items like those in your Cart](#)

subtotal = \$25.42
Make any changes below? [Update](#)

Shopping Cart Items -- To Buy Now		Price:	Qty:
<small>Item added on June 1, 2011</small>	Before I Go to Sleep: A Novel - S. J. Watson; Hardcover <small>Condition: New</small> Available for Pre-order Save for later	\$15.27 You Save: \$10.72 (41%)	<input type="text" value="1"/>
Delete <small>Eligible for FREE Super Saver Shipping</small> <input type="checkbox"/> This will be a Gift Learn more			
<small>Item added on June 1, 2011</small>	Give Your Elevator Speech a Lift! - Lorraine Howell; Paperback <small>Condition: New</small> In Stock Save for later	\$10.15 You Save: \$2.80 (22%)	<input type="text" value="1"/>
Delete <small>Eligible for FREE Super Saver Shipping</small> * For epp1545's Wish List <input type="checkbox"/> This will be a Gift Learn more			

Εικόνα 3.4: Καλάθι αγορών του Amazon.com

Αυτό που κάνει την διαφορά στο καλάθι αυτό είναι ότι ο κάθε χρήστης μπορεί να στείλει κάποιον βιβλίο σαν δώρο γράφοντας την διεύθυνση στην οποία θα σταλεί όταν συμπληρώσει την φόρμα με τα προσωπικά του στοιχεία.

Εικόνα 3.5: Σύνθετη αναζήτηση του Amazon.com

Formats	Amazon Price	New from	Used from
Kindle Edition	\$9.95	-	-
Paperback, Large Print	\$11.85	\$11.11	\$11.11
Mass Market Paperback	\$7.99	\$4.45	\$4.40
Audible Audio Edition, Unabridged	\$23.76 or Free with Audible 30-day free trial		

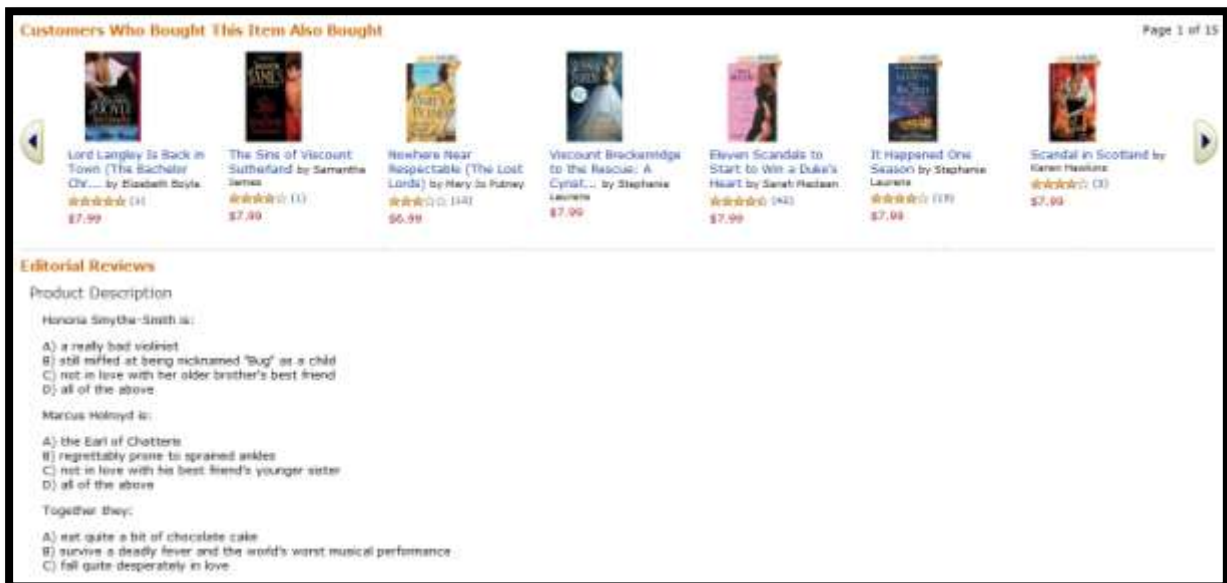
Frequently Bought Together

Price For All Three: \$23.97

This item: Just Like Heaven (The Smythe-Smiths) by Julia Quinn Mass Market Paperback \$7.99
 Lord Langley Is Back in Town (The Bachelor Chronicles: the Widows of Standon) by Elizabeth Boyle Mass Market Paperback \$7.99
 The Sins of Viscount Sutherland by Samantha James Mass Market Paperback \$7.99

Εικόνα 3.6: Εμφάνιση χαρακτηριστικών ενός βιβλίου

Στην αρχή της σελίδας εμφανίζονται διάφορες προσφορές που παρέχει το Amazon στους χρήστες. Επίσης, η σελίδα προτείνει στους χρήστες ποια άλλα βιβλία θα μπορούσε να αγοράσει μαζί με το αρχικό βιβλίο που επέλεξε.

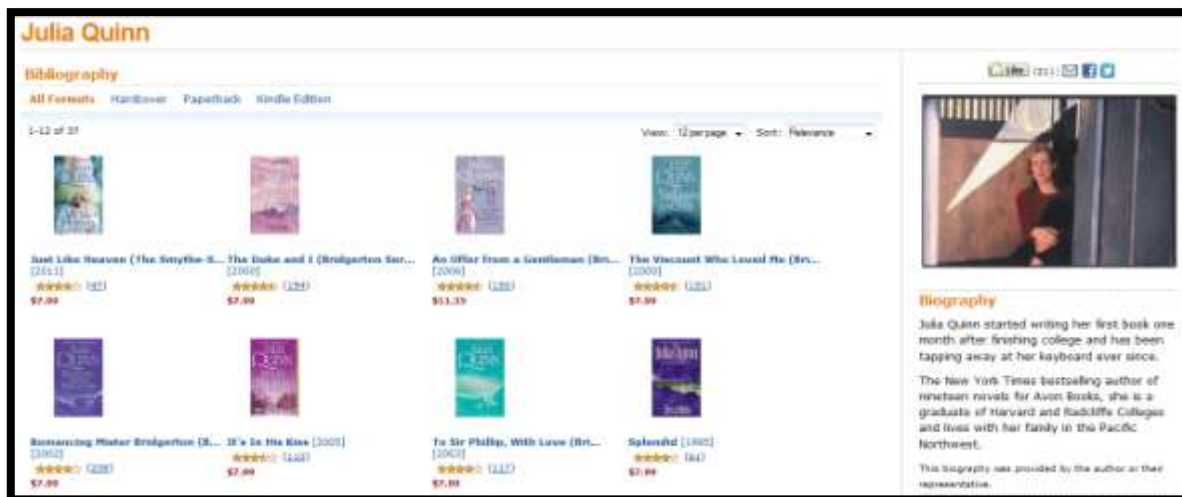


Εικόνα 3.7: Συνέχεια της σελίδας εμφάνισης ενός βιβλίου

Στη συνέχεια ένα script εμφανίζει τα βιβλία που αγόρασαν άλλοι χρήστες μαζί με το συγκεκριμένο βιβλίο ακολουθούμενο από την περιγραφή του βιβλίου.

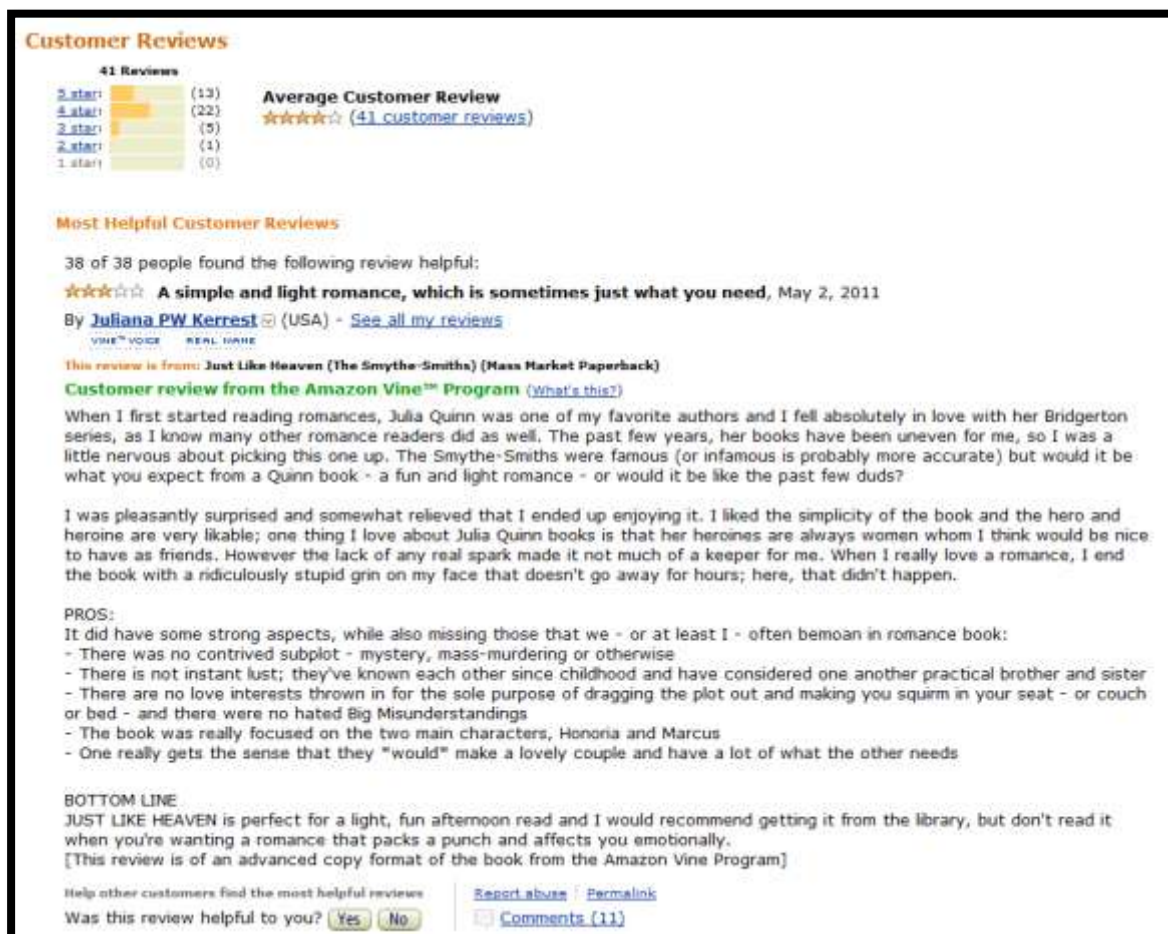


Εικόνα 3.8: Εμφάνιση στοιχείων βιβλίου και συγγραφέα



Εικόνα 3.9: Βιογραφία του Συγγραφέα

Το Amazon.com παρέχει πρόσβαση στους χρήστες στο βιογραφικό σημείωμα του κάθε συγγραφέα, καθώς και σε βιβλία που ο ίδιος έχει γράψει. Επίσης υπάρχει ενημέρωση για τα event που ο συγγραφέας έχει καθορίσει να διεξάγει.



Εικόνα 3.10: Κριτικές πελατών & σχόλια

3.2 State Of Art στην τεχνολογία

3.2.1 Διαδίκτυο

Το **Διαδίκτυο**, είναι το μεγαλύτερο δίκτυο υπολογιστών στο κόσμο το οποίο δημιουργήθηκε το 1969 από τέσσερα πανεπιστήμια των ΗΠΑ. Στην ουσία απαρτίζεται από πολλά δίκτυα και υπολογιστές που είναι συνδεδεμένοι μεταξύ τους. Τα δίκτυα αυτά μπορεί να είναι τοπικά (LAN), ή ευρείας περιοχής (WAN).

Οι υπολογιστές του διαδικτύου διαχωρίζονται σε εκείνους των Παροχέων Internet οποίοι είναι μόνιμα συνδεδεμένοι στο υπερδίκτυο, επειδή παρέχουν συνεχώς δεδομένα και πληροφορίες. Επίσης, υπάρχουν και αυτοί των χρηστών, συνδρομητών του Internet, οι οποίοι συνδέονται και διακόπτουν την επικοινωνία τους με αυτό ανάλογα με τις ανάγκες τους.

Οι κυριότερες εφαρμογές και υπηρεσίες που μας παρέχει το Διαδίκτυο είναι οι εξής:

- Το ηλεκτρονικό ταχυδρομείο (e-mail)
- Οι ομάδες συζητήσεων (newsgroups)
- Η διαμοίραση αρχείων (file sharing)
- Η επιφόρτωση αρχείων (file transfer)
- Ο Παγκόσμιος Ιστός (World Wide Web).

Πολλές υπηρεσίες έχουν βασιστεί πάνω στο ηλεκτρονικό ταχυδρομείο και τον Παγκόσμιο Ιστός όπως είναι οι ταχυδρομικές λίστες και τα αρχεία καταγραφής ιστού (blogs). Το Διαδίκτυο καθιστά δυνατή τη διάθεση υπηρεσιών σε πραγματικό χρόνο, υπηρεσίες όπως το ραδιόφωνο που είναι προσπελάσιμες από οπουδήποτε στον κόσμο.

Τα τελευταία χρόνια στην Ελλάδα παρατηρείται σημαντική αύξηση του αριθμού των χρηστών (από 13% το 2001 σε 31% το 2007) ηλικίας 15 έως 65 ετών που κατέχουν προσωπικό Η/Υ. Αντίστοιχα, παρατηρείται αύξηση των ωρών χρήσης του Διαδικτύου που φτάνουν κατά μέσο όρο τις 8,6 ανά εβδομάδα. Οι πιο κύριοι λόγοι χρήσης του Διαδικτύου είναι η ενημέρωση και η αποστολή μηνυμάτων μέσω του ηλεκτρονικού ταχυδρομείου. Αντίθετα, δεν είναι τόσο εξεζητημένη η αναζήτηση προϊόντων και υπηρεσιών. Μόνο το 18% των χρηστών προχώρησε σε μια online αγορά. Παρόλα αυτά, οι αγορές πραγματοποιήθηκαν κυρίως από ελληνικούς ιστοχώρους (41%) παρά από ξένους (35%). Οι χρήστες προβαίνουν σε μια αγορά μέσω του Διαδικτύου λόγω της προσιτής τιμής και της καλής εξυπηρέτησης. Τέλος, πολλοί χρήστες πιστεύουν ότι ο κίνδυνος διαρροής προσωπικών δεδομένων κατά τις ηλεκτρονικές αγορές είναι πολύ μεγάλος.

3.2.2 Παγκόσμιος Ιστός

Συχνά θεωρούμε ότι Παγκόσμιος ιστός είναι το Internet, αλλά η αντίληψη αυτή είναι λανθασμένη καθώς ο ιστός αποτελεί μία μόνο εφαρμογή του Internet. Ο Παγκόσμιος Ιστός αποτελείται από πακέτα δεδομένων τα οποία κυκλοφορούν μέσα στο Διαδίκτυο. Είναι, δηλαδή, ένα πλαίσιο εφαρμογών και υπηρεσιών προσφερόμενο προς τον χρήστη του Διαδικτύου για την διανομή εγγράφων, γραφικών, εικόνων και ήχου. Τα έγγραφα αυτά, που είναι γνωστά ως ιστοσελίδες, μπορούν να παρέχουν κείμενο, εικόνες, ήχους, συνδέσμους (links), και πολλά άλλα.

Όταν ο χρήστης τρέχει τον φυλλομετρητή του, βλέπει μια προκαθορισμένη αρχική σελίδα και, στη συνέχεια, χρησιμοποιώντας υπερσυνδέσμους (hyperlinks) μεταφέρεται σε άλλες σελίδες, οι οποίες είναι αποθηκευμένες σε άλλους διακομιστές που μπορεί να βρίσκονται σε οποιοδήποτε σημείο του κόσμου.

Ο Παγκόσμιος Ιστός αναπτύχθηκε στο CERN (Ευρωπαϊκό Κέντρο για την Πυρηνική Έρευνα) έτσι ώστε οι επιστήμονες ενώ βρίσκονται σε απομακρυσμένα σημεία να μπορούν να ανταλλάξουν τα αποτελέσματα των ερευνών τους.

3.2.3 Τοποθεσίες Ιστού και Ιστοσελίδες

Ένας ιστότοπος, ιστοχώρος ή δικτυακός τόπος (web site) είναι μία συλλογή από ιστοσελίδες, εικόνες, βίντεο και άλλα ψηφιακά στοιχεία, τα οποία φιλοξενούνται στο ίδιο domain (περιοχή) του Παγκόσμιου Ιστού. Βασίζεται στην υπηρεσία www (world wide web), με τη χρησιμοποίηση του πρωτοκόλλου HTTP. Η υπηρεσία αυτή δίνει τη δυνατότητα στους χρήστες του Internet να δημιουργήσουν οποιοδήποτε είδους περιεχόμενο στις ιστοσελίδες τους. Το σύνολο των ιστότοπων αποτελεί το world wide web (www).

Μία Ιστοσελίδα (web page) είναι ένα είδος εγγράφου του Παγκόσμιου Ιστού (WWW) που περιλαμβάνει πληροφορίες με την μορφή κειμένου, υπερκειμένου, εικόνας, βίντεο και ήχου. Οι ιστοσελίδες αλληλοσυνδέονται και μπορεί ο χρήστης να μεταβεί από τη μία στην άλλη κάνοντας «κλικ», επιλέγοντας δηλαδή συνδέσμους που υπάρχουν στο κείμενο ή στις φωτογραφίες της ιστοσελίδας. Οι σύνδεσμοι προς άλλες σελίδες εμφανίζονται συνήθως υπογραμμισμένοι και με μπλε χρώμα για να είναι γρήγορα ξεκάθαρο στον επισκέπτη ότι πρόκειται για σύνδεσμο προς άλλη ιστοσελίδα, χωρίς όμως πάντα να είναι αυτό απαραίτητο.

Η κατασκευή ιστοσελίδων γίνεται μέσω προγραμμάτων που κυκλοφορούν ελεύθερα, αλλά υπάρχουν και αυτοματοποιημένοι μηχανισμοί κατασκευής ιστοσελίδων που επιτρέπουν σε απλούς χρήστες να δημιουργήσουν εύκολα και γρήγορα προσωπικές ή και εμπορικές ιστοσελίδες. Επίσης, είναι πολλές οι εταιρίες, που εξειδικεύονται στη δημιουργία ιστοσελίδων που στοχεύουν στην αγορά κάποιου προϊόντος από τους χρήστες, στην επικοινωνία με τον ιδιοκτήτη του ιστότοπου ή απλά στο ανέβασμα του εταιρικού προφίλ μιας επιχείρησης.

Οι ιστοσελίδες μπορούν να είναι στατικές ή δυναμικές, που αναπροσαρμόζουν αυτόματα το περιεχόμενό τους ή την εμφάνισή τους βασιζόμενες σε μια ποικιλία παραγόντων, όπως μια καταχώρηση από τον τελικό-χρήστη, μια καταχώρηση ή αλλαγή στο περιβάλλον προγραμματισμού από τον χειριστή της ιστοσελίδας (όπως εν παραδείγματι μια μορφοποίηση της βάσης δεδομένων). Εξαιτίας της συνεχούς εξειδίκευσης στους τομείς της επικοινωνιακής Πληροφορικής, δημιουργείται μια σαφής τάση διαχωρισμού του σχεδιασμού στο Διαδίκτυο από την ανάπτυξη του διαδικτύου ως μέσου ροής της πληροφορίας και των αγαθών προς όλες τις διαδικτυακές υπηρεσίες.

Γενικότερα οι ιστοσελίδες διαχωρίζονται σε στατικές και δυναμικές:

- Οι στατικές δεν αλλάζουν περιεχόμενο και διάταξη με οποιοδήποτε αίτημα εκτός και αν ο προγραμματιστής αναβαθμίσει τη σελίδα. Μια απλή HTML σελίδα είναι παράδειγμα στατικού περιεχομένου.
- Οι δυναμικές προσαρμόζουν το περιεχόμενο και/ή την εμφάνισή τους σύμφωνα με την καταχώρηση/αλληλεπίδραση ή τις αλλαγές του τελικού χρήστη στο περιβάλλον προγραμματισμού (χρήστης, ώρα, τροποποιήσεις στη βάση δεδομένων, κτλ). Το περιεχόμενο μπορεί να αλλάξει στον υπολογιστή του τελικού-χρήστη με τη χρήση των γλωσσών προγραμματισμού που εκτελούνται στον υπολογιστή του χρήστη (JavaScript, VBScript, Actionscript κ.α.). Το περιεχόμενο στις δυναμικές σελίδες συχνά μεταφράζεται στον εξυπηρετητή (server) μέσω γλωσσών προγραμματισμού που εκτελούνται στον εξυπηρετητή (Perl, PHP, ASP, JSP, ColdFusion, .NET κτλ).

3.2.4 Σχεδιασμός Σελίδων Ιστού

Ο σχεδιασμός στο Διαδίκτυο αφορά την ικανότητα δημιουργίας παρουσιάσεων περιεχομένου (συνήθως κειμένου ή πολυμέσων) οι οποίες φτάνουν στον τελικό-χρήστη μέσω του Παγκόσμιου Ιστού, με τη χρήση λογισμικού όπως ένας φυλλομετρητής (web browser) ή άλλου λογισμικού σχεδιασμένου για το διαδίκτυο.

Μια ιστοσελίδα είναι αυτή που παρουσιάζει κείμενα και εφαρμογές πολυμέσων στον τελικό-χρήστη. Τέτοια στοιχεία όπως κείμενο, εικόνες (σύμφωνα με τα πρότυπα SVG, BMP, GIF, JPEG ή

PNG) και φόρμες μπορούν να τοποθετηθούν στη σελίδα με τη χρήση γλωσσών σήμανσης υπερκειμένου όπως HTML/XHTML/XML. Η αναπαραγωγή πιο σύνθετων πολυμέσων (ανυσματικών γραφικών, βίντεο, ήχων, γραφικών με ενσωματωμένο ήχο και εικόνα) μπορεί να γίνει με πρόσθετα (plug-ins) όπως το Flash, το QuickTime, το περιβάλλον χρόνου εκτέλεσης Java, κ.α ή με τεχνολογίες όπως οι γλώσσες σήμανσης HTML5, XML και MathML, και το πρότυπο φύλλων στυλ CSS 3.

Οι καινούριες εκδόσεις των προγραμμάτων περιήγησης (Internet Explorer 7, 8, Firefox 3.6, safari, chromium κλπ) που ακολουθούν τα W3C πρότυπα οδήγησαν σε μια ευρεία αποδοχή και χρήση των XHTML/XML σε συνδυασμό με τα CSS (Cascading Style Sheets) για την τοποθέτηση και διαχείριση των στοιχείων και αντικειμένων της ιστοσελίδας. Τα τελευταία πρότυπα στοχεύουν στο να αποκτήσουν τα προγράμματα περιήγησης την δυνατότητα να προσφέρουν μια ευρεία γκάμα επιλογής πολυμέσων και πρόσβασης στους πελάτες χωρίς τη χρήση των προσθέτων (plug-ins).

Με την συνεχόμενη ειδίκευση στην τεχνολογία της Πληροφορικής δημιουργείται η ανάγκη διαχωρισμού της Σχεδιασμού Ιστοσελίδων από την Προγραμματιστή Ιστοσελίδων.

Για τη διαδικασία σχεδιασμού μιας ιστοσελίδας, μιας εφαρμογής ή ενός πολυμέσου για το διαδίκτυο μπορεί να συνδυάζονται πολλοί κλάδοι όπως animation, συγγραφή, επικοινωνιακός σχεδιασμός, εταιρική ταυτότητα, σχεδιασμός γραφικών, αλληλεπίδραση ανθρώπου-υπολογιστή, αρχιτεκτονική υπολογιστών, σχεδιασμός αλληλεπίδρασης, marketing, φωτογραφία, βελτιστοποίηση μηχανών αναζήτησης και τυπογραφία.

3.2.5 Φόρμες Ιστού

Μια φόρμα Ιστού εμπεριέχεται μέσα σε μια ιστοσελίδα και περιέχει διάφορα πλαίσια κειμένου, κουμπιά επιλογών, πλαίσια ελέγχου κ.α. στα οποία οι χρήστες μπορούν να καταχωρίσουν ή να επιλέξουν διάφορα δεδομένα και να τα στείλουν στην τοποθεσία που η φόρμα καθορίζει..

Τα δεδομένα αυτά τα παραλαμβάνει στη συνέχεια ο διακομιστής της φόρμας. Για παράδειγμα, με τις φόρμες οι χρήστες μπορούν να στέλνουν σχόλια, ερωτήσεις, παραγγελίες και άλλα στοιχεία στο διαχειριστή μιας τοποθεσίας.

3.2.6 Διεύθυνση IP

Μία διεύθυνση IP (*Internet Protocol address*), είναι ένας μοναδικός αριθμός που χρησιμοποιείται για την αναγνώριση και συνεννόηση των συσκευών που βρίσκονται μέσα στο Διαδίκτυο. Ο συσκευές αυτές μπορεί να είναι δρομολογητές (routers), υπολογιστές, time-servers, εκτυπωτές, μηχανές για fax μέσω Internet, και ορισμένα τηλέφωνα. Κάθε μια από τις συσκευές αυτές πρέπει να έχουν την δική της μοναδική διεύθυνση. Όπως κάθε διεύθυνση κατοικίας και αριθμός τηλεφώνου αντιστοιχούν σε ένα και μοναδικό έτσι και μια IP address χρησιμοποιείται για τη μοναδική αναγνώριση ενός υπολογιστή ή άλλης συσκευής που συνδέεται στο Διαδίκτυο.

Μια διεύθυνση IP μπορεί να "μοιράζεται" σε πολλές συσκευές είτε επειδή αυτές είναι μέρος του ίδιου web server environment, είτε λόγω ενός proxy server (π.χ. ενός Παροχέα Υπηρεσιών Διαδικτύου (ISP) ή μιας υπηρεσίας για εξασφάλιση ανωνυμίας - anonymizer service) που λειτουργούν ως μεσολαβητές. Στην τελευταία περίπτωση (χρήση διακομιστή μεσολάβησης) η πραγματική διεύθυνση IP μπορεί να αποκρύπτεται από το διακομιστή που δέχεται αίτηση.

3.2.7 Domain Name System

Το Domain Name System ή DNS (Σύστημα Ονομάτων Τομέα) είναι ένα σύστημα με το οποίο αντιστοιχίζονται οι διευθύνσεις IP σε ονόματα τομέων (Domain Names). Είναι, δηλαδή, ο μηχανισμός διάθεσης ονομάτων και διευθύνσεων που χρησιμοποιούνται στο Διαδίκτυο. Τα ονόματα τομέων όπως και οι διευθύνσεις IP που αναπαριστούν είναι μοναδικά, έχουν μια ιεραρχία και διαβάζονται από αριστερά προς τα δεξιά.

Ένας κόμβος του Internet διαθέτει ένα συγκεκριμένο και μοναδικό όνομα που αποτελεί και την ταυτότητα του κόμβου. Για παράδειγμα, η CyberNet έχει συμβολικό όνομα “serres.hol.gr”, όπου “serres” είναι το όνομα του εξυπηρετή (server), το “hol” η επικράτεια που περιλαμβάνει τον εν λόγω εξυπηρετή, και η επικράτεια “gr” περιλαμβάνει όλους τους εξυπηρετές που βρίσκονται στην Ελλάδα.

Σε ένα όνομα μπορεί να αντιστοιχούν πολλές IP διευθύνσεις. Για παράδειγμα, η διεύθυνση www.google.gr αντιστοιχεί σε τρεις IP διευθύνσεις, την 66.102.9.99, την 66.102.9.104 και την 66.102.9.147. Σε αυτήν την περίπτωση έχουμε τρεις εξυπηρετητές που λειτουργούν ταυτόχρονα εκτελώντας την ίδια εργασία, αλλά μοιράζονται τον φόρτο εργασίας διά τρία. Σε αυτήν την περίπτωση ο διακομιστής DNS εκτελεί εξισορρόπηση φορτίου μεταξύ των τριών άλλων διακομιστών.

Domain names

Μια υπηρεσία εύρεσης δικτύου, το Domain Name System (DNS), δίνει τη δυνατότητα να αντιστοιχηθούν ονόματα υπολογιστών (hostnames) σε μια διεύθυνση IP. Με αυτό τον τρόπο οι άνθρωποι μπορούν εύκολα να θυμούνται ένα όνομα και όχι μια σειρά αριθμών. Το DNS επιτρέπει σε πολλαπλές διευθύνσεις και ονόματα να δείχνουν σε ένα πόρο του Διαδικτύου.

Ένας ακόμη λόγος ύπαρξης του DNS είναι να επιτρέπει, για παράδειγμα, σε έναν ιστότοπο που φιλοξενείται σε πολλούς διακομιστές (που ο καθένας από τους οποίους έχει τη δική του διεύθυνση IP) να παρέχει στοιχειώδη εξισορρόπηση φόρτου (load balancing), δηλαδή μία ορθή διαμοίραση της κίνησης που δέχεται ο ιστότοπος ανάμεσα στους υπολογιστές που τον φιλοξενούν.

Για παράδειγμα, το www.wikipedia.org δείχνει στο 207.142.131.248.

3.2.8 Φυλλομετρητής Ιστού

Ο φυλλομετρητή Ιστού (Web browser) είναι ένα πρόγραμμα που αποκωδικοποιεί τα περιεχόμενα των ιστοσελίδων και να εμφανίζει στην οθόνη στην κατάλληλη μορφή. Το κείμενο και οι εικόνες σε μια ιστοσελίδα μπορεί να περιέχουν υπερσυνδέσμους προς άλλες ιστοσελίδες του ίδιου ή διαφορετικού ιστότοπου. Ο Web browser επιτρέπει στον χρήστη την γρήγορη και εύκολη πρόσβαση σε πληροφορίες που βρίσκονται σε διάφορες ιστοσελίδες και ιστότοπους εναλλάσσοντας τις ιστοσελίδες μέσω των υπερσυνδέσμων. Οι φυλλομετρητές χρησιμοποιούν τη γλώσσα μορφοποίησης HTML για την προβολή των ιστοσελίδων, για αυτό η εμφάνιση μιας ιστοσελίδας μπορεί να διαφέρει ανάλογα με τον browser.

Για κάθε browser διατίθενται, επίσης, και αρκετά πρόσθετα στοιχεία (add-on ή plug-ins), με στόχο την επαύξηση των δυνατοτήτων τους, τη βελτίωση της χρηστικότητάς τους και την προστασία του χρήστη σε θέματα ασφάλειας.

Οι περισσότεροι χρησιμοποιούμενοι browsers είναι οι:

- Windows Internet Explorer
- Mozilla Firefox
- Apple Safari
- Netscape Navigator
- Opera
- Google Chrome

3.2.9 URL

Ως URL (Uniform Resource Location) ονομάζεται η διεύθυνση μιας ιστοσελίδας. Η διεύθυνση αυτή λειτουργεί ως ταυτότητα για τις σελίδες του Παγκόσμιου Ιστού - κάθε ιστοσελίδα έχει τη δική της μοναδική διεύθυνση URL.

Μία διεύθυνση URL έχει την εξής μορφή:

<http://www.example.com/sofia/photos.php>

Το πρώτο μέρος (`http://`) μας ενημερώνει για το πρωτόκολλο που θα πρέπει να χρησιμοποιήσει ο φυλλομετρητής για να συνδεθεί με τον υπολογιστή στον οποίο είναι αποθηκευμένη η ιστοσελίδα και να φορτώσει τα περιεχόμενά της. Οι χαρακτήρες `://` σε συνδυασμό μας ενημερώνουν ότι θα ακολουθήσει το όνομα του διακομιστή.

Το δεύτερο μέρος (`www.example.com`) ξεκινάει με το είδος του διακομιστή και το πλήρες όνομα της περιοχής (domain name) στην οποία βρίσκεται η ιστοσελίδα.

Το τρίτο μέρος μας ενημερώνει για τον υποφάκελο και το όνομα του αρχείου της ιστοσελίδας. Αποτελείται, δηλαδή, από το όνομα του καταλόγου που περιέχει την ιστοσελίδα και το όνομα του εγγράφου.

3.2.10 Επικοινωνία μέσω Internet

Οι **διαδικτυακές κοινότητες** αποτελούν μια αρκετά διαδεδομένη μορφή διαδικτυακής επικοινωνίας, την οποία δεν άργησαν να υιοθετήσουν οι επιχειρήσεις ως εργαλείο προβολής, προώθησης και πώλησης των προϊόντων τους. Οι ηλεκτρονικές κοινότητες δίνουν τη δυνατότητα σε διάφορες ομάδες ανθρώπων που ενδιαφέρονται για κάποιο εξειδικευμένο θέμα να ανταλλάσσουν απόψεις πάνω στο θέμα αυτό.

Η επιχείρηση μπορεί να παρέχει τα μέσα σε όλους αυτούς τους πελάτες (ή υποψήφιους πελάτες) να ενημερωθούν για εξειδικευμένα θέματα που τους ενδιαφέρουν, και αφορούν κυρίως τα προϊόντα ή τις υπηρεσίες της επιχείρησης, και να ανταλλάξουν απόψεις και πληροφορίες.

Η υιοθέτηση των διαδικτυακών κοινοτήτων παρέχει τη δυνατότητα για ενημέρωση και άμεση επίλυση προβλημάτων ή αποριών σε σχέση με κάποιο προϊόν, επικοινωνία με τους πελάτες, καλύτερη εξυπηρέτηση, αλλά και ανάπτυξη σχέσεων εμπιστοσύνης ανάμεσα στην επιχείρηση και τον πελάτη, στοιχείο απαραίτητο για την αντιμετώπιση του ανταγωνισμού.

Τα περισσότερα ηλεκτρονικά βιβλιοπωλεία προσφέρουν τη δυνατότητα σχολιασμού των βιβλίων τους από αναγνώστες. Έτσι, πολλοί υποψήφιοι αγοραστές μπορούν να πάρουν και μια άλλη γνώμη πριν αγοράσουν ένα βιβλίο. Ακόμη και στην περίπτωση που κάποιοι πελάτες εκφράσουν αρνητική γνώμη, θα έχει τη δυνατότητα να κάνετε διορθώσεις ή βελτιώσεις, δείχνοντας τον επαγγελματισμό της επιχείρησής και την καλή επικοινωνία με τους πελάτες.

Στην δική μας ιστοσελίδα ο χρήστης μπορεί να επικοινωνήσει με τον διαχειριστή μέσω μιας φόρμας επικοινωνίας, όπου συμπληρώνοντας τα στοιχεία του μπορεί να στείλει το μήνυμα που επιθυμεί. Επίσης δύναται η δυνατότητα στους χρήστες να ανταλλάξουν τις απόψεις τους για διάφορα βιβλία, συμπληρώνοντας την ανάλογη φόρμα. Είναι μια μορφή forum, του οποίου τα μηνύματα επιβλέπει ο διαχειριστής, ο οποίος αποφασίζει αν θα εμφανιστούνε στην ιστοσελίδα ή όχι.

3.2.11 Μηχανή αναζήτησης

Η μηχανή αναζήτησης είναι μια εφαρμογή που επιτρέπει την αναζήτηση κειμένων και αρχείων στο Διαδίκτυο. Τα αποτελέσματα παρουσιάζονται σε μια λίστα και μέσω υπερσυνδέσμων μεταβαίνει ο χρήστης στις πληροφορίες που επιθυμεί. Ένα πρόγραμμα υπολογιστή δημιουργεί μια βάση δεδομένων με τις πληροφορίες, και το κατάλληλο περιβάλλον που εμφανίζεται στον τελικό χρήστη ο οποίος χρησιμοποιεί την εφαρμογή από άλλον υπολογιστή συνδεδεμένο στο διαδίκτυο.

Εμπορική Εφαρμογή

Οι μηχανές αναζήτησης χρησιμοποιούνται πολύ από το αγοραστικό κοινό παγκοσμίως γι' αυτό και πολλές εταιρίες επιθυμούν να διαφημίζονται μέσω αυτών. Επειδή οι εταιρίες αυτές όλο και πληθαίνουν υπάρχει μεγάλος ανταγωνισμός για την κατάκτηση των πρώτων θέσεων στις ιστοσελίδες

αποτελεσμάτων αναζήτησης. Το φαινόμενο αυτό οδήγησε στην ανάπτυξη ενός άλλου κλάδου, του search engine optimization (Βελτιστοποίηση για τις μηχανές αναζήτησης) που περιλαμβάνει τεχνικές κατασκευής και δικτύωσης ιστοσελίδων που να προσφέρουν καλύτερες θέσεις στα αποτελέσματα.

Γνωστές Μηχανές Αναζήτησης

Υπάρχουν πολλές μηχανές αναζήτησης με τις μηχανές αναζήτησης της εταιρίας Google (Google) και της Yahoo (Yahoo! Search) πρώτες σε χρήση. Άλλες γνωστές μηχανές αναζήτησης από τη Microsoft η MSN Search και η Microsoft Live Search.

Ελληνικές μηχανές αναζήτησης μεταξύ άλλων είναι αυτές της Forthnet, της Otenet, του Cleverlook και του Pathfinder.

Δημιουργία της δικής μας Φόρμας Αναζήτησης.

Για να δώσουμε στο χρήστη την δυνατότητα αναζήτησης του βιβλίου που επιθυμεί δημιουργήσαμε μια φόρμα αναζήτησης όπου συμπληρώνοντας τα κατάλληλα πεδία της φόρμας εκείνη ερευνά την βάση **δεδομένων** για να εντοπίσει και να μας εμφανίσει τα στοιχεία που περιλαμβάνουν τις συγκεκριμένες λέξεις κλειδιά.

3.2.12 Ηλεκτρονικό Εμπόριο

Οι αγορές και οι πωλήσεις αγαθών και υπηρεσιών στο Internet, ονομάζεται ηλεκτρονικό εμπόριο. Το Internet δεν χρησιμοποιείται πλέον μόνο για αναζήτηση πληροφοριών και δεδομένων ή για να στείλουμε και να λαμβάνουμε ηλεκτρονικά μηνύματα. Έχουν εμφανιστεί πολλές νέες υπηρεσίες που μας επιτρέπουν να κάνουμε αγορές και πωλήσεις αγαθών και διαφόρων υπηρεσιών, το λεγόμενο ηλεκτρονικό εμπόριο. Να κάνουμε, δηλαδή, τα ψώνια μας με εύκολο τρόπο, χωρίς ταλαιπωρία, απλώς χρησιμοποιώντας έναν υπολογιστή και μια σύνδεση στο Internet.

Η διαδικασία αγοράς προϊόντων μέσω του Διαδικτύου είναι σχετικά τυποποιημένη.

- Επισκεπτόμαστε την ιστοσελίδα του προμηθευτή.
- Επιλέγουμε το προϊόν που θέλουμε να αγοράσουμε.
- Αν υπάρχει υπερσύνδεσμος προς το “καλάθι” μεταβαίνουμε σε αυτόν.
- Ο αγοραστής θα χρειαστεί να συμπληρώσει μια φόρμα με τα στοιχεία του, ώστε να παραγγείλει το βιβλίο που επιθυμεί.

Διαδικασία αγοράς βιβλίων

Στην ιστοσελίδα που δημιουργήσαμε ο χρήστης πρέπει πρώτα να γίνει μένος της ιστοσελίδας μας καταχωρώντας τα στοιχεία του στη φόρμα εγγραφής και έπειτα να συνδεθεί σε αυτή. Στη συνέχεια του δίνεται η δυνατότητα προσθήκης των βιβλίων που επιθυμεί να αγοράσει στο καλάθι του. Όταν έχει αποφασίσει ποια βιβλία θέλει να αγοράσει καθώς και την ποσότητα αυτών μεταβαίνει στην Ολοκλήρωση της Αγοράς.

3.2.13 Ασφάλεια στο Internet

Ένας χρήστης θα πρέπει να προστατευτεί:

- Από εξωτερικούς εισβολείς: Οι Hackers ή οι Crackers μπορούν να εισχωρήσουν στον υπολογιστή μας και να αντιγράψουν ή να διαγράψουν ή να αλλάξουν την διαμόρφωση των αρχεία από το σκληρό δίσκο. Επίσης μπορούν να εγκαταστήσουν ανεπιθύμητα προγράμματα, καθώς και να παρακολουθήσουν διάφορες δραστηριότητες που λαμβάνουν χώρο τόσο στο

δικό μας υπολογιστή όσο και στο Διαδίκτυο. Για το λόγο αυτό υπάρχει το τείχος προστασίας (firewall) των Windows το οποίο ελέγχει τα δεδομένα που αποστέλλονται και λαμβάνονται από τον υπολογιστή ή το δίκτυο και έχει τη δυνατότητα να μπλοκάρει την είσοδο και έξοδο τους.

- Από ιούς υπολογιστών: Υπάρχουν πολλά προγράμματα που εκτελούν ανεπιθύμητες ενέργειες στο σύστημά μας. Γι' αυτό και θα πρέπει να έχουμε εγκατεστημένο στον υπολογιστή μας ένα πρόγραμμα αντίιγus που θα προστατεύει τον υπολογιστή μας σκανάροντας όλα τα αρχεία μας και διαγράφοντας εκείνα που έχουν “μολυνθεί” από κάποιο ιό.
- Από την εμφάνιση ιστοσελίδων με ανεπιθύμητο περιεχόμενο, την εμφάνιση ενοχλητικών παραθύρων κατά την περιήγηση στον Ιστό, και από την λήψη ενοχλητικών μηνυμάτων.

Ασφαλείς Συναλλαγές

Οι περισσότερες τοποθεσίες για να ολοκληρώσουν μια ηλεκτρονική συναλλαγή απαιτούνε από τους χρήστες την χρήση πιστωτικής κάρτας. Γι' αυτό και είναι σχεδιασμένες με τέτοιο τρόπο ώστε να προστατεύουν τους χρήστες από πιθανή υποκλοπή των στοιχείων τους απαγορεύοντας σε μη εξουσιοδοτημένα άτομα να έχουν πρόσβαση στις πληροφορίες των χρηστών.

Ένας φυλλομετρητής Ιστού υποστηρίζει πρωτόκολλα ασφαλείας που χρησιμοποιούνται από ασφαλείς τοποθεσίες, έτσι ώστε όταν αποστέλλει τα δεδομένα του να μην υπάρχει κίνδυνος να προσπελαστούν από μη εξουσιοδοτημένα άτομα.

Κάθε φορά που ένας χρήστης επισκέπτεται μια ασφαλή τοποθεσία, η τοποθεσία του αποστέλλει ένα πιστοποιητικό, το οποίο εμφανίζεται στην γραμμή κατάστασης στο φυλλομετρητή Ιστού με τη μορφή λουκέτου.

Ψηφιακά Πιστοποιητικά

Το ψηφιακό πιστοποιητικό είναι ένα ηλεκτρονικό έγγραφο που χρησιμεύει στην αναγνώριση μιας οντότητας. Μέσω των ψηφιακό πιστοποιητικό γίνεται η επιβεβαίωση της ταυτότητας των χρηστών προστατεύοντάς τους από υποκλοπές ή αλλοιώσεις των δεδομένων που ανταλλάσσουν μέσω Internet. Επίσης, τα ψηφιακά πιστοποιητικά προφυλάγουν τον υπολογιστή μας από μη ασφαλές λογισμικό που διατίθεται στο Διαδίκτυο.

Υπάρχουν δύο κύριοι τύποι πιστοποιητικών:

- Τα προσωπικά πιστοποιητικά, τα οποία επιβεβαιώνουν την ταυτότητα ενός χρήστη και χρησιμοποιούνται για την προσπέλαση ιστοσελίδων που απαιτούν επιβεβαίωση των χρηστών.
- Τα πιστοποιητικά τοποθεσιών Ιστού, τα οποία επιβεβαιώνουν την γνησιότητα και την ασφάλεια μιας τοποθεσίας Ιστού και χρησιμεύουν στο να είμαστε σίγουροι ότι αποστέλλουμε τα προσωπικά μας στοιχεία ή ότι κατεβάζουμε αρχεία από ασφαλείς τοποθεσίες.

Κρυπτογράφηση

Με την κρυπτογράφηση τροποποιούμε τα δεδομένα ώστε να μην είναι δυνατή η ανάγνωση και η κατανόησή τους από μη εξουσιοδοτημένους χρήστες. Έτσι είμαστε σίγουροι ότι τα δεδομένα θα διαβαστούν μόνο από τον παραλήπτη που έχουμε καθορίσει. Η κρυπτογράφηση μπορεί να πραγματοποιηθεί από διάφορες εφαρμογές και λειτουργικά συστήματα.

Η διαδικασία μετατροπής ενός κρυπτογραφημένου αρχείου, ώστε να είναι δυνατή η ανάγνωσή του από εξουσιοδοτημένους χρήστες, λέγεται **αποκρυπτογράφηση** και πραγματοποιείται με ανάλογο λογισμικό.

3.3 Χρονοδιάγραμμα Εργασίας

	2011									
	Ιανουάριος		Φεβρουάριος		Μάρτιος		Απρίλιος		Μάιος	
	15	30	15	30	15	30	15	30	15	30
Σχεδιασμός Ιστοχώρου										
Σχεδιασμός Animation										
Δημιουργία Βάσης Δεδομένων										
Δημιουργία Φορμών με την PHP										
Εμφάνιση δεδομένων με την PHP										
Αναφορά Πτυχιακής										

Εικόνα 3.1: Χρονοδιάγραμμα Πτυχιακής

ΚΕΦΑΛΑΙΟ 4: ΚΥΡΙΟ ΜΕΡΟΣ ΠΤΥΧΙΑΚΗΣ

4.1 Ρόλοι και δικαιώματα πρόσβασης

4.1.1 Λειτουργίες Διαχειριστή

Ο Διαχειριστής βρίσκεται στην κορυφή της πυραμίδας της εφαρμογής διότι είναι υπεύθυνος για την σωστή λειτουργία της σελίδας. Έχει πρόσβαση σχεδόν σε ότι γίνεται στην εφαρμογή και του αναλογούν όλα τα δικαιώματα που έχει ο χρήστης-μέλος και ο απλός χρήστης-επισκέπτης της σελίδας, πέρα από αυτά που έχει μόνο αυτός. Επομένως του έχουν αποδοθεί οι σημαντικότερες λειτουργίες, οι οποίες περιγράφονται παρακάτω:

- Εμφάνιση όλων των εγγεγραμμένων χρηστών.

Με την λειτουργία αυτή ο διαχειριστής μπορεί να δει όλες τις εγγραφές που έχουν καταχωρηθεί στη βάση δεδομένων. Δίνετε ταυτόχρονα η δυνατότητα και το δικαίωμα να διαγράψει οποιαδήποτε εγγραφή.

- Πρόσθεση και Επεξεργασία των βιβλίων.

Με αυτήν την λειτουργία δίνεται το δικαίωμα στον διαχειριστή να επεξεργαστεί τα στοιχεία και τις πληροφορίες όλων των βιβλίων της ιστοσελίδας και να αποθηκεύσει τις αλλαγές στη βάση δεδομένων.

- Εμφάνιση Όλων των Παραγγελιών.

Με την λειτουργία αυτή εμφανίζονται στον διαχειριστή όλα οι παραγγελίες που υπάρχουν στην Β/Δ του συστήματος, τα οποία έχουν σταλεί από τους χρήστες της ιστοσελίδας για την αγορά των βιβλίων που επιθυμούν.

- Διαχείριση Μηνυμάτων του Forum.

Με την λειτουργία ο διαχειριστής είναι αυτός που αποφασίζει ποια από τα μηνύματα των χρηστών ή των απλών επισκεπτών της ιστοσελίδας θα εμφανίζονται στην σελίδα του forum.

4.1.2 Λειτουργίες Χρήστη-Μέλους

- Εγγραφή στο Σύστημα.

Με την λειτουργία αυτή ο κάθε επισκέπτης μπορεί να γραφτεί στην σελίδα και γίνει μέλος της, αποκτώντας πρόσβαση στα δικαιώματα του χρήστη-μέλους.

- Εισαγωγή στο σύστημα.

Με την λειτουργία αυτή μπορεί ο κάθε χρήστης να πληκτρολογήσει τα στοιχεία πρόσβασης του και να εισαχθεί στην ιστοσελίδα ως μέλος με τα αντίστοιχα δικαιώματα, εφόσον έχει εγγραφεί κανονικά.

- Επεξεργασία Στοιχείων.

Δίνεται η δυνατότητα στο χρήστη-μέλος να επεξεργαστεί τα προσωπικά στοιχεία του, καθώς και τα στοιχεία πρόσβασης.

- Αναζήτηση (Απλή-Σύνθετη).

Η λειτουργία αυτή είναι διαθέσιμη και στα τρία επίπεδα χρηστών, και είναι μια από τις σημαντικότερες λειτουργίες. Υπάρχουν δύο τρόποι αναζήτησης, αυτός της απλής, και αυτός της σύνθετης αναζήτησης.

Στην απλή αναζήτηση απαιτείται η συμπλήρωση ενός μόνο κριτηρίων αναζήτησης, ενώ στην σύνθετη αναζήτηση δίνεται στον χρήστη η δυνατότητα να εισάγει περισσότερα κριτήρια αναζήτησης, ώστε να επιτευχθούν συγκεκριμένα αποτελέσματα.

- Αποστολή Μηνύματος.

Με την λειτουργία αυτή μπορεί ο κάθε χρήστης να συμπληρώσει μια φόρμα αποστολής στοιχείων μαζί με κάποιο μήνυμα και να την στείλει στον διαχειριστή ώστε να μπορέσει να πληροφορηθεί για κάποιο βιβλίο.

- Αποσύνδεση από το σύστημα.

Με την λειτουργία αυτή αποσυνδέεται ο χρήστης- μέλος από την ιστοσελίδα, και επανέρχεται ως απλός επισκέπτης στην αρχική σελίδα.

- Προσθήκη Βιβλίων στο Καλάθι.

Δίνεται η δυνατότητα στο χρήστη-μέλος να προσθέτει όσα βιβλία επιθυμεί στο καλάθι του, έτσι ώστε να προβεί στην on-line αγορά τους.

- Προσθήκη Βιβλίων στη Wishlist.

Η Wishlist είναι μία λίστα στην οποία ο χρήστης-μέλος θα μπορεί να προσθέτει τα αγαπημένα του βιβλία.

4.1.3 Λειτουργίες Χρήστη-Επισκέπτη

Ο απλός επισκέπτης ενδιαφερόμενος της ιστοσελίδας, είναι ο στόχος της συγκεκριμένης εφαρμογής. Ο κάθε χρήστης θα πρέπει να περιηγείται με άνεση στις σελίδες της εφαρμογής, χωρίς να συναντά δυσκολίες στην εκτέλεση των λειτουργιών που δημιουργήθηκαν για αυτόν.

Δεν απαιτούνται από μέρος του ιδιαίτερες γνώσεις πληροφορικής. Απαιτούνται όμως οι στοιχειώδεις γνώσεις σχετικά με το διαδίκτυο και την πλοήγηση σε αυτό, και φυσικά η χρήση ηλεκτρονικού υπολογιστή.

4.1.4 Δημιουργία Χρηστών και Προνομίων

Το σύστημα προνομίων της MySQL σχεδιάστηκε με στόχο να περιορίσει την πρόσβαση συγκεκριμένων χρηστών σε συγκεκριμένες διαταγές για συγκεκριμένες βάσεις δεδομένων. Μ αυτό τον τρόπο ένας διακομιστής Ιστού εξασφαλίζει την ασφαλή προσπέλαση διαφόρων βάσεων δεδομένων από πολλούς χρήστες. Κάθε χρήστης στο σύστημα της MySQL μπορεί να έχει συγκεκριμένες δυνατότητες σε συγκεκριμένες βάσεις δεδομένων, οι οποίες βρίσκονται σε διάφορους υπολογιστές υπηρεσίας.

Ο βασικός χρήστης - της MySQL, όχι του συστήματος – έχει τις περισσότερες δυνατότητες και μπορεί να δημιουργεί άλλους δευτερεύοντες χρήστες, στους οποίους μπορεί να εκχωρεί δυνατότητες παρόμοιες ακόμη και με του βασικού χρήστη.

Η MySQL ελέγχει πρώτα αν ο χρήστης έχει την άδεια να συνδεθεί με το διακομιστή και στη συνέχεια ελέγχει αν ο χρήστης έχει το δικαίωμα να εκτελέσει τη συγκεκριμένη εντολή SQL στις συγκεκριμένες βάσεις δεδομένων.

Προνόμια της MySQL	
ΔΙΚΑΙΩΜΑΤΑ	ΕΠΙΤΡΕΠΕΙ
SELECT	Ανάγνωση εγγραφών από πίνακες.
INSERT	Προσθήκη νέων γραμμών δεδομένων σε πίνακες.
UPDATE	Τροποποίηση δεδομένων που ήδη υπάρχουν σε πίνακες.
DELETE	Διαγραφή δεδομένων από πίνακες.
INDEX	Δημιουργία και κατάργηση ευρετηρίων πινάκων.
ALTER	Τροποποίηση της δομής ενός πίνακα.
CREATE	Δημιουργία νέων πινάκων ή βάσεων δεδομένων.
DROP	Διαγραφή πινάκων ή βάσεων δεδομένων.
RELOAD	Επαναφόρτωση των δικαιωμάτων σε πίνακες.
SHUTDOWN	Τερματισμός του διακομιστή της MySQL.
PROCESS	Εξέταση και τερματισμός των διεργασιών της MySQL.
FILE	Εισαγωγή δεδομένων σε πίνακες από αρχεία κειμένου.
GRANT	Δημιουργία νέων χρηστών.
REVOKE	Αφαίρεση αδειών από χρήστες.

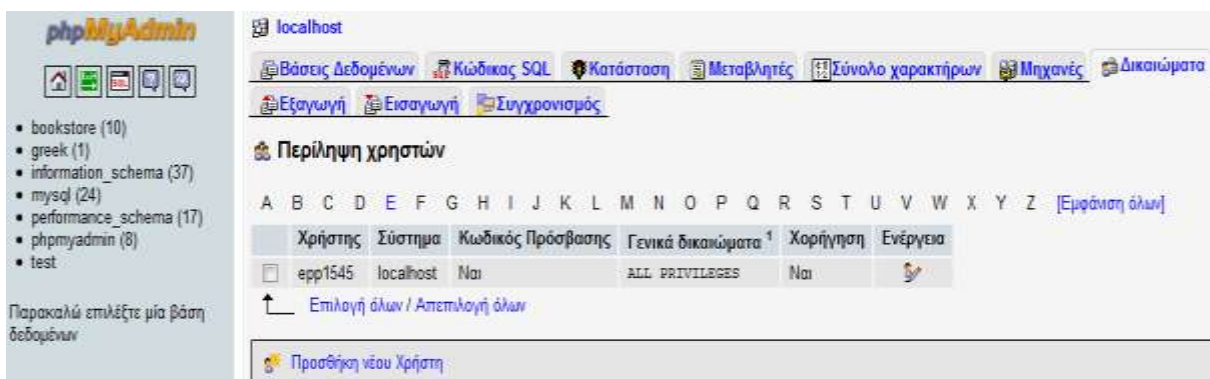
Πίνακας 4.1: Προνόμια της MySQL

Υπάρχουν πολλοί τρόποι καθορισμού χρηστών στην MySQL. Ένας τρόπος είναι μέσω της διαταγής GRANT στον πελάτη mysql.

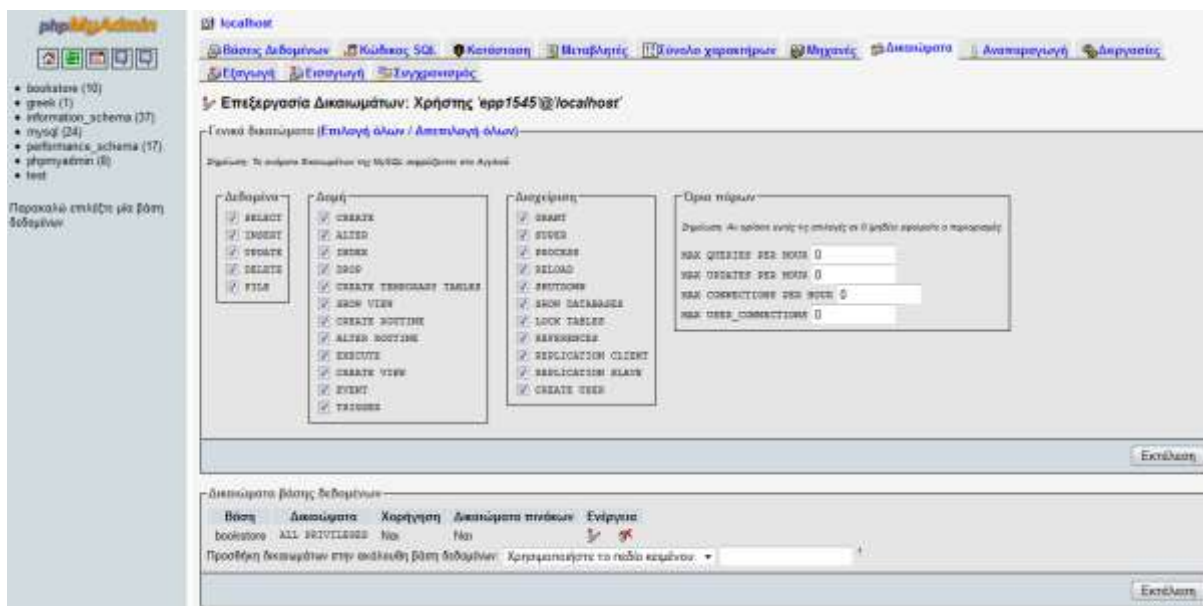
GRANT προνόμια ON βάση_δεδομένων.* TO όνομα_χρήστη@υπολογιστής_υπηρεσίας IDENTIFIED BY ‘κωδικός’

Η επεξεργασία όλων των βάσεων δεδομένων καθορίζετε με τη σύνταξη *.*
 Αν ο όρος IDENTIFIED BY ‘κωδικός’ παραλειφθεί ο χρήστης δε θα χρειάζεται να καταχωρεί κωδικό πρόσβασης. Επίσης, για να επιτραπεί οποιοσδήποτε υπολογιστής υπηρεσίας γράφουμε: όνομα_χρήστη@’%’

πχ. **CREATE DATABASE example;**
GRANT SELECT, INSERT, UPDATE, DELETE ON example.*
TO ‘someuser’@’localhost’
IDENTIFIED BY ‘somepass’;



Εικόνα 4.1: Χρήστες με δικαιώματα



Εικόνα 4.2: Εμφάνιση δικαιωμάτων χρηστών

4.2 Σχεδιασμός Ιστοχώρου

4.2.1 Μορφοποίηση Σελίδων με Φύλλα CSS

Τα επικαλυπτόμενα φύλλα στυλ (CSS) είναι μια συλλογή κανόνων μορφοποίησης που ελέγχει την εμφάνιση του περιεχομένου σε μια ιστοσελίδα. Με τη χρήση στυλ CSS για τη μορφοποίηση μιας σελίδας, το περιεχόμενο διαχωρίζεται από τον τρόπο παρουσίασης. Το περιεχόμενο της σελίδας μας - ο κώδικας HTML- βρίσκεται μέσα στο αρχείο HTML, ενώ οι κανόνες CSS που ορίζουν τον τρόπο παρουσίασης του κώδικα βρίσκονται μέσα σε ένα εξωτερικό φύλλο στυλ. Ο διαχωρισμός του περιεχομένου από τον τρόπο παρουσίασης διευκολύνει σημαντικά τη συντήρηση της εμφάνισης της τοποθεσίας μας από μια κεντρική θέση, επειδή δεν χρειάζεται να ενημερώνουμε κάθε ιδιότητα σε κάθε σελίδα όποτε θέλουμε να κάνετε μια αλλαγή. Επίσης, με τον διαχωρισμό του περιεχομένου από τον τρόπο παρουσίασης ο κώδικα HTML γίνεται πιο απλός, πράγμα που σημαίνει ταχύτερη φόρτωση στα προγράμματα περιήγησης.

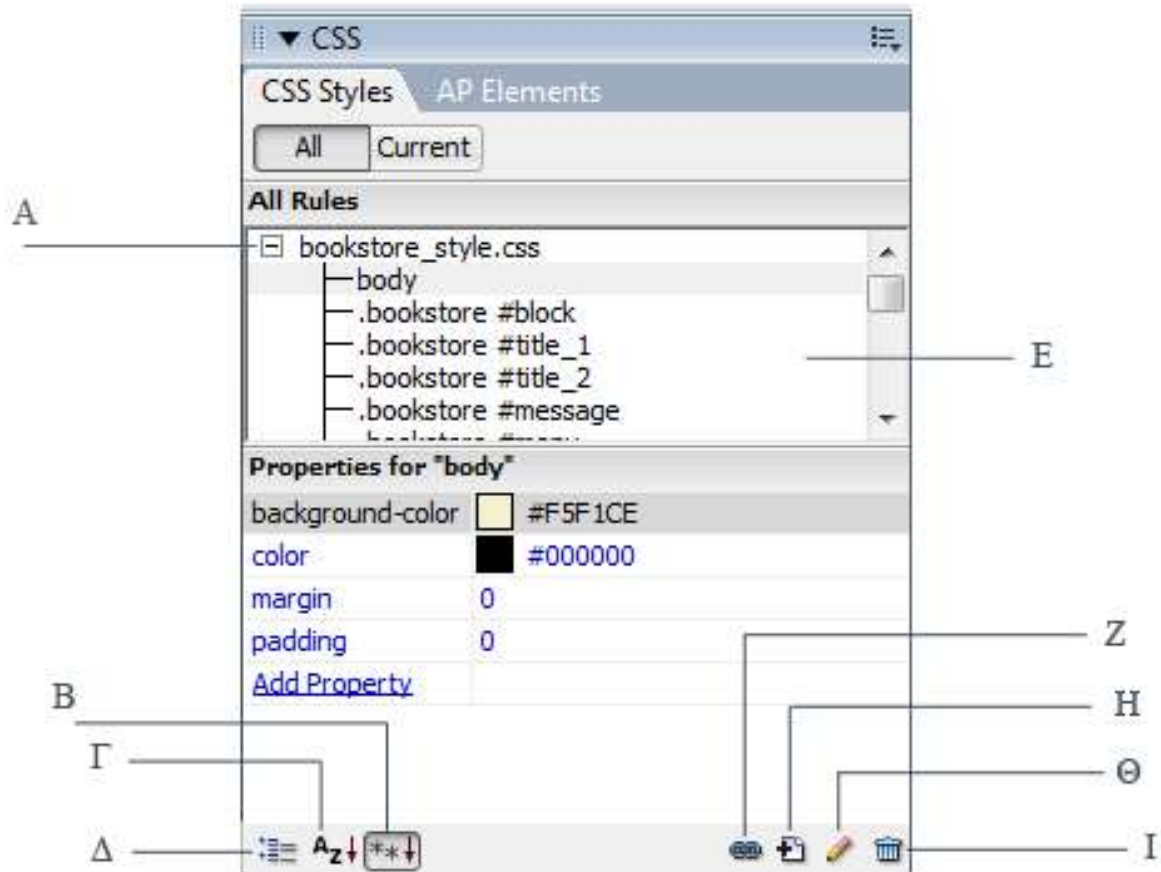
Τα φύλλα CSS σάς δίνουν μεγάλη ευελιξία και δυνατότητα ελέγχου με ακρίβεια της εμφάνισης της σελίδας σας. Με τα φύλλα CSS, μπορείτε να ελέγχετε πολλές ιδιότητες του κειμένου, όπως συγκεκριμένες γραμματοσειρές και μεγέθη χαρακτήρων, έντονους, πλάγιους ή υπογραμμισμένους χαρακτήρες, σκιές κειμένου, χρώμα κειμένου και φόντου, χρώμα και υπογράμμιση των συνδέσεων, και πολλές άλλες. Χρησιμοποιώντας CSS για τον έλεγχο των γραμματοσειρών σας, μπορείτε επίσης να διασφαλίσετε μεγαλύτερη συνέπεια στον χειρισμό της διαμόρφωσης και εμφάνισης της σελίδας σας σε διαφορετικά προγράμματα περιήγησης.

Εκτός από τη μορφοποίηση του κειμένου, μπορούμε να χρησιμοποιήσετε φύλλα CSS για να ελέγξουμε τη μορφοποίηση και την τοποθέτηση στοιχείων μιας ιστοσελίδας σε επίπεδο τμήματος. Στοιχείο επιπέδου τμήματος είναι ένα αυτόνομο κομμάτι περιεχομένου, που συνήθως διαχωρίζεται με μια νέα γραμμή στην HTML και μορφοποιείται οπτικά ως ένα τμήμα. Για παράδειγμα, τα tag h1, p και div δημιουργούν στοιχεία επιπέδου τμήματος σε μια σελίδα. Μπορούμε να ορίσουμε περιθώρια και

περιγράμματα για τα στοιχεία επιπέδου τμήματος, να τα τοποθετήσουμε σε μια συγκεκριμένη θέση, να τους προσθέσουμε χρώμα φόντου, να ρυθμίσετε το κείμενο ως κινούμενο γύρω από αυτά, κ.ο.κ. Ο χειρισμός των στοιχείων επιπέδου τμήματος είναι, ουσιαστικά, ο τρόπος με τον οποίο διαμορφώνουμε σελίδες με τη χρήση CSS.

Τα στυλ που έχουν δημιουργηθεί με CSS μπορούν να αποθηκευτούν απευθείας μέσα στο έγγραφο ή, για περισσότερη ευελιξία, σε ένα εξωτερικό φύλλο στυλ. Αν επισυνάψουμε ένα εξωτερικό φύλλο στυλ σε μερικές σελίδες, τότε οι αλλαγές που κάνετε στο φύλλο στυλ θα απεικονίζονται αυτόματα σε όλες αυτές τις σελίδες.

Πλαίσιο CSS



Εικόνα 4.3: Πλαίσιο CSS

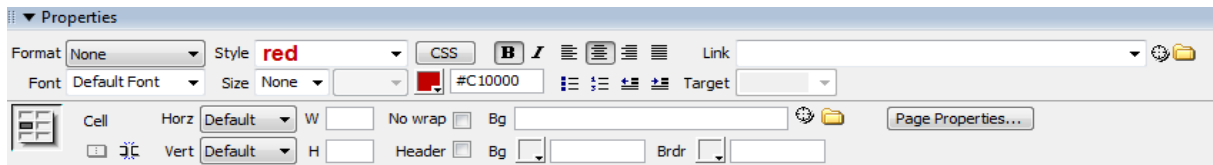
- A. Εξωτερικό Φύλλο CSS.
- B. Εμφάνιση μόνο ρυθμίσεων ιδιοτήτων.
- Γ. Προβολή Λίστας.
- Δ. Προβολή κατηγορίας
- E. Παράθυρο Κανόνων.
- Z. Επισύναψη φύλλου στυλ.
- H. Νέος κανόνας CSS.
- Θ. Επεξεργασία φύλλου στυλ.
- I. Διαγραφή Ενσωματωμένου φύλλου στυλ.

Τύποι κανόνων CSS

- Ετικέτα (Tag) : Ο σκοπός μιας ετικέτας είναι η αλλαγή της εμφάνισης μιας συγκεκριμένης ετικέτας HTML. Τα στυλ που βασίζονται σε ετικέτες θα μεταβάλουν την εμφάνιση της ετικέτας και των περιεχομένων της.
- Κλάση (Class): Μια προσαρμοσμένη κλάση μπορεί να αποκτήσει ένα μοναδικό όνομα και να μη σχετίζεται με μια συγκεκριμένη ετικέτα HTML. Μπορούμε να χρησιμοποιήσουμε την ίδια κλάση για έναν πίνακα, μια παράγραφο, ή πεδίο φόρμας όσες φορές επιθυμούμε. Η εφαρμογή μιας κλάσης σε μια ετικέτα γίνεται με τον εξής τρόπο :

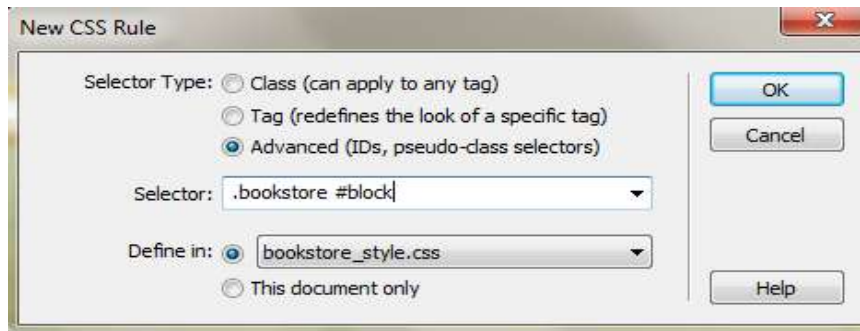
<body class="bookstore">....</body>

Όταν μορφοποιούμε κείμενο από το παράθυρο Properties, το Dreamweaver ονομάζει και δημιουργεί αυτόματα στυλ κλάσης, τα οποία μπορούμε να ξαναχρησιμοποιήσουμε σε άλλα στοιχεία. Τα στυλ κλάσεις μπορούμε εύκολα να τα αναγνωρίσουμε, γιατί το όνομα τους ακολουθεί πάντα μια τελεία. πχ. **.bookstore**



Εικόνα 4.4: Παράθυρο Properties

- Προηγμένη (Advanced) – Ταυτότητες & Ψευδοκλάσεις: Η ταυτότητα (ID) είναι ένας ειδικός τύπος κανόνα που είναι μοναδικός για ένα συγκεκριμένο στοιχείο της σελίδας. Οι ταυτότητες περιέχουν πληροφορίες οι οποίες είναι μοναδικές για ένα πλαίσιο, και έτσι δημιουργείται μια μοναδική ταυτότητα αποκλειστικά για το συγκεκριμένο στοιχείο. Τέλος, το όνομα της ταυτότητας ακολουθεί πάντα το σύμβολο της δέσμης (#). Σε κάθε κανόνα ταυτότητας αντιστοιχεί μια ετικέτα <div> μέσω του χαρακτηριστικού ταυτότητας. πχ. **<div id="block">.....</div>**



Εικόνα 4.5: Δημιουργία Κανόνα CSS

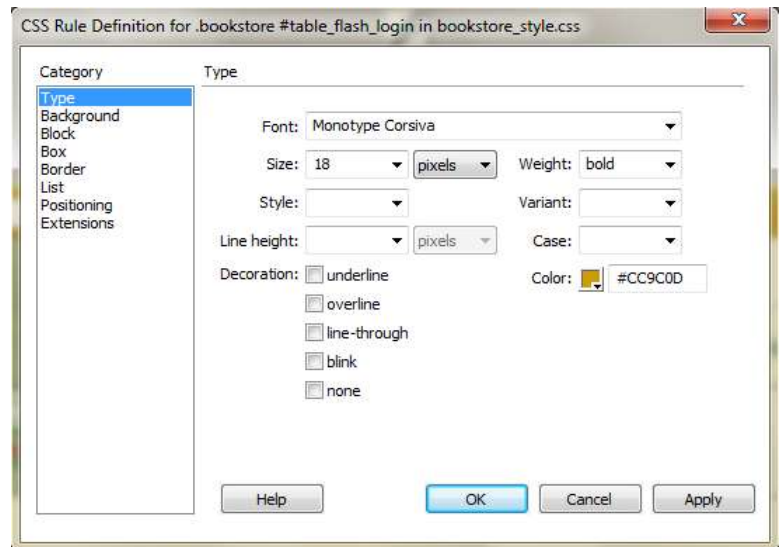
CSS Rule definition

Όταν προσθέτουμε στυλ από το πλαίσιο Styles εμφανίζεται το πλαίσιο διαλόγου CSS Rule definition, το οποίο περιέχει μία σειρά από κατηγοριοποιημένα πεδία κειμένου και μενού που μπορούμε να χρησιμοποιήσουμε για να επιλέξουμε ιδιότητες για το νέο μας κανόνα.

Στην καρτέλα Type προσθέτουμε ιδιότητες για την μορφοποίηση του κειμένου.

Στην καρτέλα Background επιλέγουμε το φόντο που θα έχει το πλαίσιο μας καθορίζοντας επίσης, αν θα επαναλαμβάνεται ή όχι.

Στην καρτέλα Block καθορίζουμε την εμφάνιση των παραγράφων, συμπεριλαμβανομένης και της ευθυγράμμισης και στοίχισης του κειμένου.



Εικόνα 4.6: Καρτέλα Type

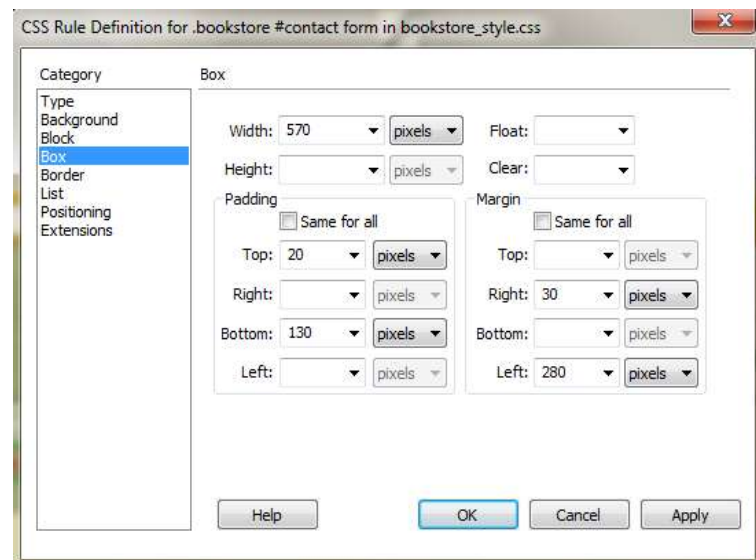
Η καρτέλα Box είναι πολύ σημαντική, γιατί καθορίζει τον χώρο που θα καταλαμβάνει το πλαίσιο μέσα στην σελίδα.

Width – Καθορίζει το πλάτος του πλαισίου.

Height - Καθορίζει το ύψος του πλαισίου.

Padding – Εσωτερικό κενό, δηλαδή η απόσταση των περιεχομένων από την πλευρά του πλαισίου.

Margin – Περιθώριο, δηλαδή η διαφανής περιοχή που περιβάλλει το πλαίσιο.



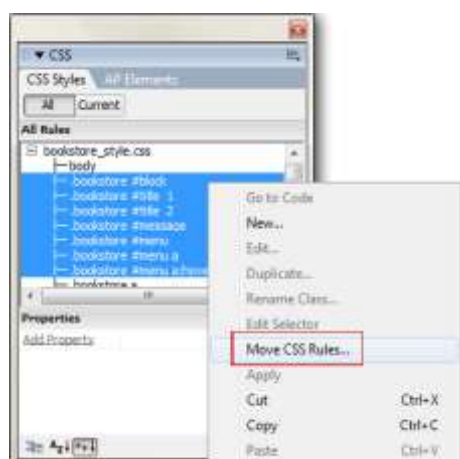
Εικόνα 4.7: Καρτέλα Box

Εξωτερικό φύλλο στυλ

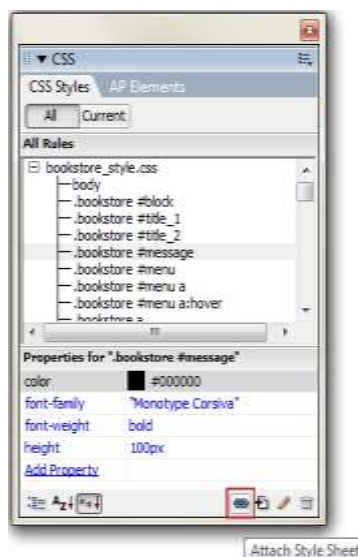
Μια συλλογή κανόνων που αποθηκεύονται σε ένα ξεχωριστό αρχείο .css ονομάζεται εξωτερικό φύλλο στυλ. Με τα εξωτερικά φύλλα στυλ όλες οι σελίδες του ιστοχώρου μοιράζονται τους ίδιους κανόνες στυλ. Όταν αλλάζουμε ένα εξωτερικό στυλ, οι σελίδες που είναι συνδεδεμένες με αυτό ενημερώνονται.

Μέσω της εντολής Move CSS Rules μπορούμε να εξάγουμε φύλλα CSS που έχουμε δημιουργήσει ήδη μέσα σε μια σελίδα και να τα αποθηκεύσουμε ως ένα εξωτερικό αρχείο με επέκταση .css .

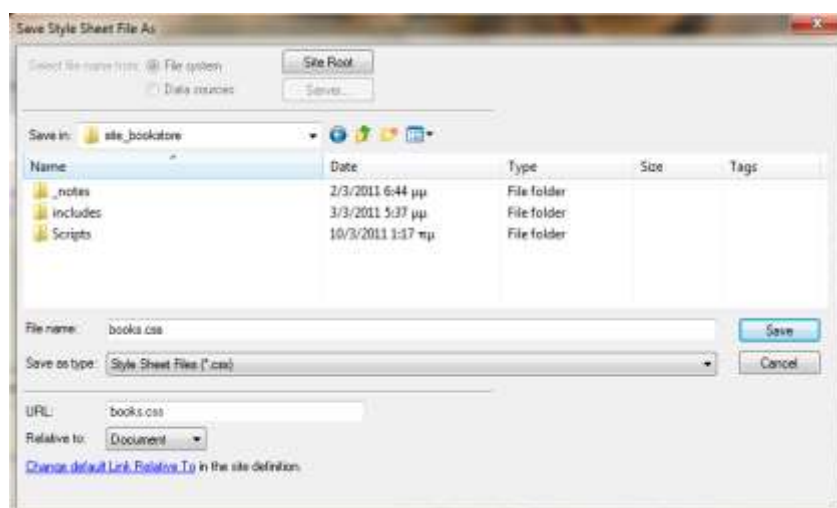
Η επισύναψη ενός εξωτερικού φύλλου στυλ γίνεται μέσω του Πλαισίου CSS επιλέγοντας Attach Style Sheet (Επισύναψη φύλλου στυλ).



Εικόνα 4.8: Move CSS Rules



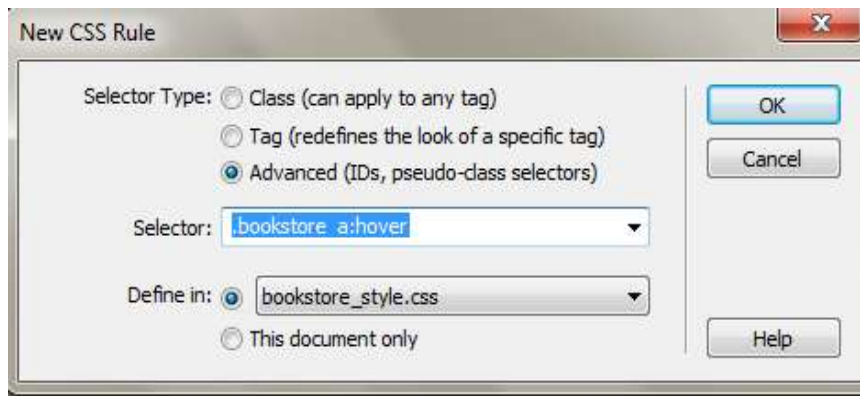
Εικόνα 4.9: Attach Style Sheet



Εικόνα 4.10: Αποθήκευση εξωτερικού αρχείου CSS

Επιλογείς ψευδοκλάσεων

Ένας επιλογέας ψευδοκλάσεων επηρεάζει ένα μέρος ή κατάσταση μιας επιλεγμένης ετικέτας ή κλάσης. Μια κατάσταση αναφέρεται συχνά στην εμφάνιση ενός στοιχείου καθώς αυτό αντιδρά σε ένα συμβάν, όπως είναι το πέρασμα του κέρσορα πάνω από αυτό. Ένας επιλογέας ψευδοκλάσης είναι ο `a:hover`, ο οποίος καθορίζει την εμφάνιση των συνδέσμων όταν περνά πάνω από αυτούς ο δείκτης του ποντικιού.



Εικόνα 4.11: Μορφοποίηση Ψευδοκλάσεων

4.2.2 Δημιουργία Διατάξεων με Πίνακες

Κατά τη μορφοποίηση πινάκων στην προβολή Design, μπορείτε να ορίσετε ιδιότητες για ολόκληρο τον πίνακα ή για επιλεγμένες γραμμές, στήλες ή κελιά του πίνακα. Όταν μια ιδιότητα, π.χ. χρώμα φόντου ή στοίχιση, έχει οριστεί σε μία τιμή για ολόκληρο τον πίνακα και σε μία άλλη για κάποια συγκεκριμένα κελιά, η μορφοποίηση των κελιών έχει προτεραιότητα έναντι της μορφοποίησης των γραμμών, η οποία με τη σειρά της έχει προτεραιότητα έναντι της μορφοποίησης του πίνακα.

Η σειρά προτεραιοτήτων για τη μορφοποίηση των πινάκων είναι η εξής:

- Κελί
- Γραμμές
- Πίνακας

Αν, για παράδειγμα, ορίσετε το χρώμα φόντου ενός συγκεκριμένου κελιού σε μπλε και, στη συνέχεια, ορίσετε το χρώμα φόντου ολόκληρου του πίνακα σε κίτρινο, το μπλε κελί δεν θα αλλάξει σε κίτρινο, αφού η μορφοποίηση του κελιού υπερισχύει της μορφοποίησης του πίνακα.

Όταν ορίζετε ιδιότητες σε μια στήλη, το Dreamweaver αλλάζει τα χαρακτηριστικά του tag *td* που αντιστοιχεί σε κάθε κελί της στήλης.

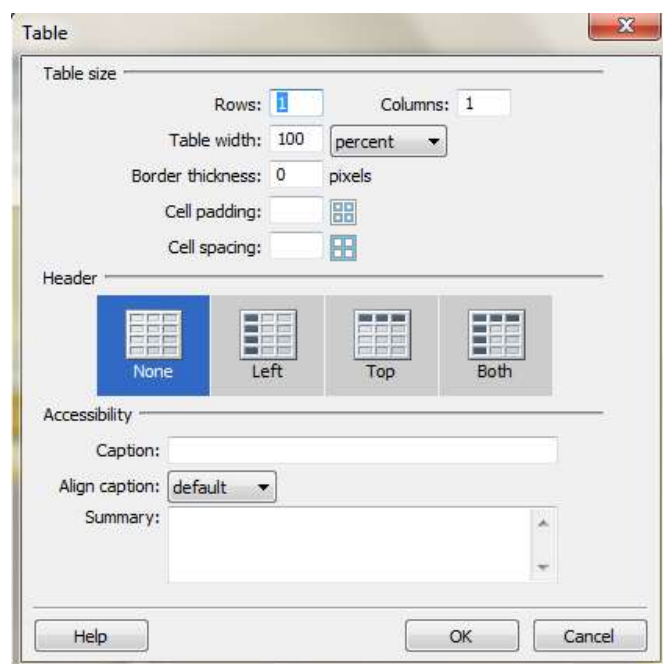
Rows – Γραμμές του πίνακα.

Columns – Στήλες του πίνακα.

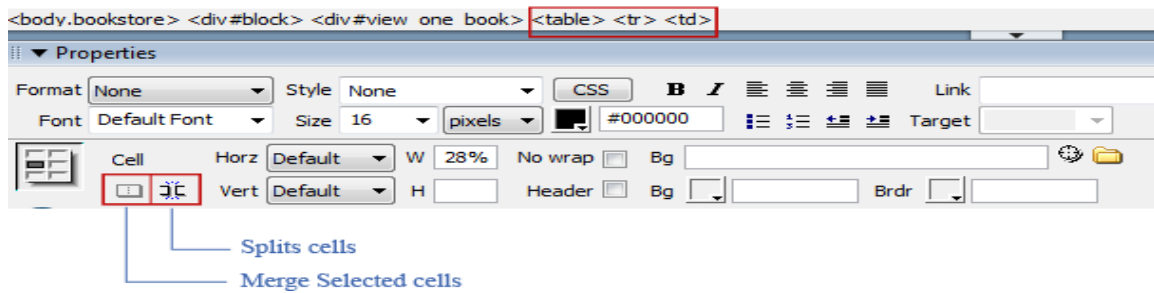
Table width – Πλάτος

Border thickness – Πάχος των κελιών.

Αν το Border thickness = 0 τότε τα κελιά του πίνακα δεν εμφανίζονται στη σελίδα όταν την τρέχουμε στον browser.



Εικόνα 4.12: Δημιουργία Πίνακα στο Dreamweaver



Εικόνα 4.13: Επεξεργασία Πινάκων από το Παράθυρο Properties

Η ετικέτα `<table>` αναφέρεται στο πίνακα.

Η ετικέτα `<tr>` αναφέρεται σε μια γραμμή του πίνακα.

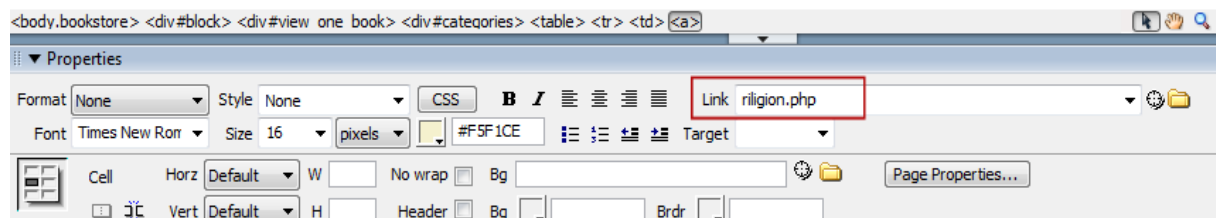
Η ετικέτα `<td>` αναφέρεται σε μια στήλη του πίνακα.

Μπορούμε να χωρίσουμε μια γραμμή του πίνακα χρησιμοποιώντας το κουμπί “Split cells”. Καθώς και να ενώσουμε δύο πεδία χρησιμοποιώντας το κουμπί “Merge selected cells”.

4.2.3 Σύνδεσμοι

Οι ιστοσελίδες αλληλοσυνδέονται και μπορεί ο χρήστης να μεταβεί από τη μία στην άλλη κάνοντας «κλικ», επιλέγοντας δηλαδή συνδέσμους που υπάρχουν στο κείμενο ή στις φωτογραφίες της ιστοσελίδας.

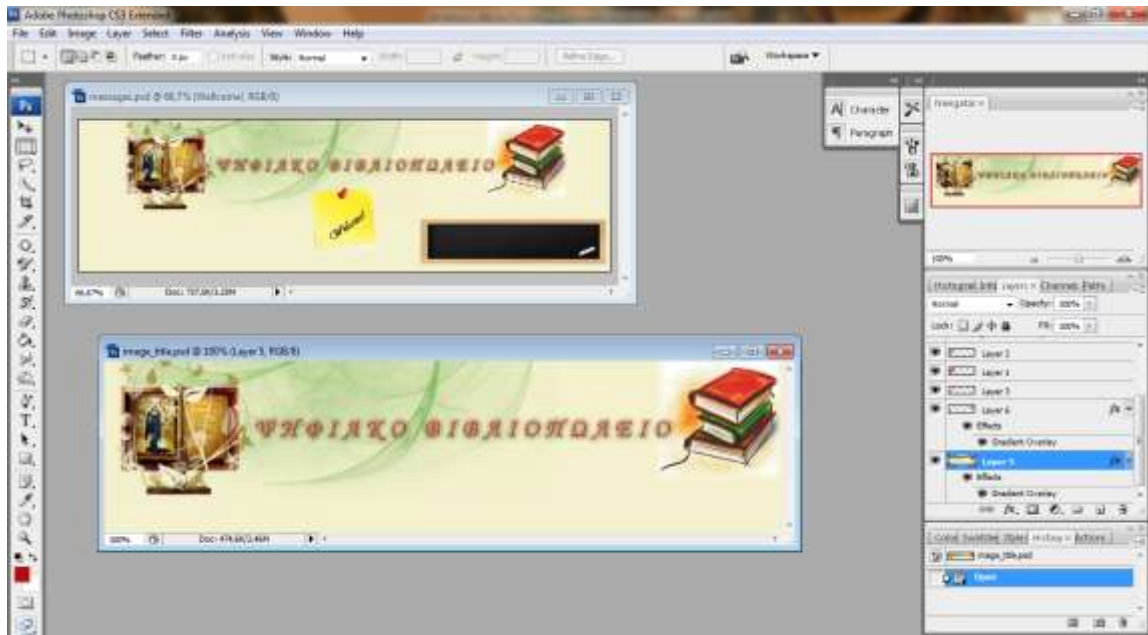
Η διαδικασία δημιουργίας συνδέσμων για να μεταβούμε από την μία σελίδα στην άλλη είναι πολύ απλή. Το μόνο που χρειάζεται να κάνουμε είναι να επιλέξουμε το «αντικείμενο» που θέλουμε να μετατρέψουμε σε σύνδεσμο και έπειτα να πληκτρολογήσουμε το όνομα του αρχείου που θέλουμε να μεταβούμε στο πλαίσιο Link.



Εικόνα 4.14: Σύνδεσμοι

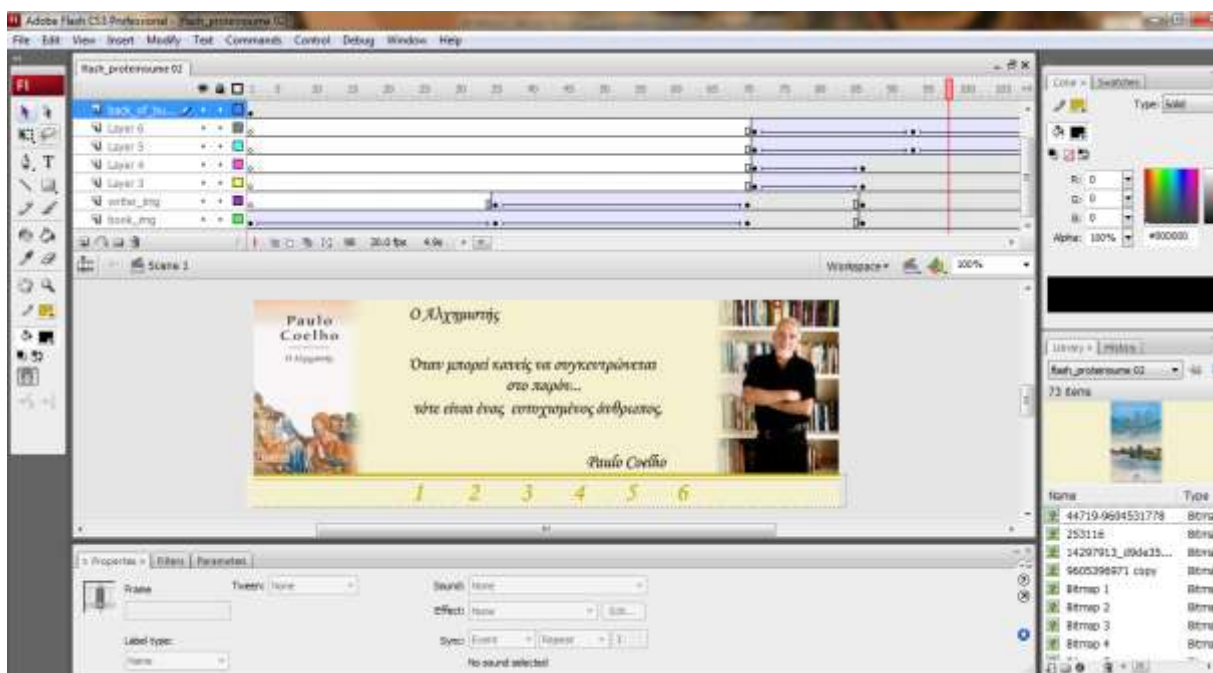
4.2.4 Διαδικασία κατασκευής του ιστοχώρου

- Δημιουργία αρχικής εικόνας στο Photoshop CS3.



Εικόνα 4.15: Δημιουργία αρχικής εικόνας στο Photoshop CS3.

- Δημιουργία Animation στο Flash CS3.



Εικόνα 4.16: Δημιουργία Animation στο Flash CS3.

- Δημιουργία μενού



```

<div id="menu">
  <table width="100%" border="0">
  <tr>
    <td><table width="842" height="27" border=0 cellpadding=0 cellspacing=0>
      <tbody>
        <tr>
          <td width="90" style="text-align: center;background-color: #CC9C00;"><a href="index.php">Αρχική</a></td>
          <td width="20">&nbsp;</td>
          <td width="120" style="text-align: center;background-color: #CC9C00;"><a href="suggestions.php">Προτεινόμενα</a></td>
          <td width="20">&nbsp;</td>
          <td width="100" style="text-align: center;background-color: #CC9C00;"><a href="best_sellers.php">Best Sellers</a></td>
          <td width="20">&nbsp;</td>
          <td width="120" style="text-align: center;background-color: #CC9C00;"><a href="contact.php">Επικοινωνία</a></td>
          <td width="20">&nbsp;</td>
          <td width="80" style="text-align: center;background-color: #CC9C00;"><a href="forum.php">Forum</a></td>
          <td width="20">&nbsp;</td>
          <td width="100" style="text-align: center;background-color: #CC9C00;"><a href="register.php">Εγγραφή</a></td>
        </tr>
      </tbody>
    </table>
    <tr>
      <td width="100%" height="1" colspan="12" style="background-color: #917826;">
      <table width="100%" border="0" cellspacing="0" cellpadding="0">
        <tr>
          <td height="1"></td>
        </tr>
      </table>
    </td>
  </tr>
</table>
</div><!-- end menu -->

```

Εικόνα 4.18: Δημιουργία του μενού των χρηστών.

Όπως βλέπουμε και από τον κώδικα το menu που δημιουργήθηκε παραπέμπει σε έξι σελίδες.

- Την Αρχική στην οποία εμφανίζονται οι Νέες Κυκλοφορίες (index.php)
- Στα προτεινόμενα βιβλία (suggestions.php)
- Στα Best Sellers (best_sellers.php)
- Στην φόρμα επικοινωνίας (contact.php)
- Στην περιοχή του forum (forum.php)
- Στην φόρμα εγγραφής (register.php)

Στα αρχεία αυτά έχει πρόσβαση ο κάθε χρήστης που επισκέπτεται την σελίδα. Αν όμως είναι εγγεγραμμένος, μετά από την είσοδό του τα αρχεία αλλάζουν προσφέροντάς του περισσότερες δυνατότητες.

Για τον χρήστη-μέλος:

- Η Αρχική σελίδα είναι η loggedin.php
- Τα Προτεινόμενα βιβλία εμφανίζονται στην σελίδα suggestions_after.php
- Τα Best Sellers στην best_sellers_after.php
- Η φόρμα επικοινωνίας στην contact_after.php
- Η περιοχή του forum στην forum_after.php
- Η φόρμα της εγγραφής δεν υπάρχει.

Οι σελίδες αυτές προσφέρουν πρόσβαση στο καλάθι αγορών, και στη Wishlist που έχουν δημιουργηθεί για τους εγγεγραμμένους χρήστες.

Αρχική	Προτεινόμε	Best Sellers	Επικοινωνία	Forum
--------	------------	--------------	-------------	-------

```

<div id="menu">
  <table width="100%" border="0">
    <tr>
      <td><table width="942" height="27" border="0" cellpadding="0" cellspacing="0">
        <tbody>
          <tr>
            <td width="80" style="text-align: center;background-color: #CC9C0D;"><a href="loggedin.php">Αρχική</a></td>
            <td width="20">&nbsp;</td>
            <td width="120" style="text-align: center;background-color: #CC9C0D;"><a href="suggestions_after.php">Προτεινόμε</a></td>
            <td width="20">&nbsp;</td>
            <td width="100" style="text-align: center;background-color: #CC9C0D;"><a href="best_sellers_after.php">Best Sellers</a></td>
            <td width="20">&nbsp;</td>
            <td width="120" style="text-align: center;background-color: #CC9C0D;"><a href="contact_after.php">Επικοινωνία</a></td>
            <td width="20">&nbsp;</td>
            <td width="80" style="text-align: center;background-color: #CC9C0D;"><a href="forum_after.php">Forum</a></td>
          </tr>
        </tbody>
      </table>
      <tr>
        <td width="100%" height="1" colspan="12" style="background-color: #917826;">
          <table width="100%" border="0" cellspacing="0" cellpadding="0">
            <tr>
              <td height="1"></td>
            </tr>
          </table>
        </td>
      </tr>
    </tbody>
  </table>
</div> <!-- end menu -->

```

Εικόνα 4.19: Δημιουργία του μενού των χρηστών-μελών.

Για τον administrator:

- Η Αρχική σελίδα είναι η admin.php
- Η σελίδα για τα προτεινόμενα η suggestions_admin.php
- Τα Best Sellers στην best_sellers_admin.php
- Η περιοχή του forum στην forum_admin.php
- Και τέλος μια σελίδα που εμφανίζει τον Πίνακα Αγορών με όνομα pinakas_agorwn.php

Αρχική	Προτεινόμε	Best Sellers	Forum	Πίνακας Αγορών
--------	------------	--------------	-------	----------------

```

<div id="menu">
  <table width="100%" height="27" border="0" cellpadding="0" cellspacing="0">
    <tbody>
      <tr>
        <td width="80" style="text-align: center;background-color: #CC9C0D;"><a href="admin.php">Αρχική</a></td>
        <td width="20">&nbsp;</td>
        <td width="120" style="text-align: center;background-color: #CC9C0D;"><a href="suggestions_admin.php">Προτεινόμε</a></td>
        <td width="20">&nbsp;</td>
        <td width="100" style="text-align: center;background-color: #CC9C0D;"><a href="best_sellers_admin.php">Best Sellers</a></td>
        <td width="20">&nbsp;</td>
        <td width="80" style="text-align: center;background-color: #CC9C0D;"><a href="forum_admin.php">Forum</a></td>
        <td width="20">&nbsp;</td>
        <td width="120" style="text-align: center;background-color: #CC9C0D;"><a href="pinakas_agorwn.php">Πίνακας Αγορών</a></td>
      </tr>
    </tbody>
  </table>
  <tr>
    <td width="100%" height="1" colspan="12" style="background-color: #917826;">
      <table width="100%" border="0" cellspacing="0" cellpadding="0">
        <tr>
          <td height="1"></td>
        </tr>
      </table>
    </td>
  </tr>
</div> <!-- end menu -->

```

Εικόνα 4.20: Δημιουργία του μενού για τον administrator.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #menu { margin-right: 30px; margin-left: 30px; font-family: "Monotype Corsiva"; font-size: 20px; color: #FFFFFF; }</pre>	<pre>.bookstore #menu a { color: #FFFFFF; font-family: "Monotype Corsiva"; font-size: 20px; text-decoration: none; } .bookstore #menu a:hover { color: #FFFFFF; font-family: "Monotype Corsiva"; font-size: 20px; text-decoration: underline; }</pre>

Πίνακας 4.1: Φύλλα CSS για την μορφοποίηση του μενού

- **Δημιουργία φόρμας login.**

Στην φόρμα εισόδου δημιουργούμε δύο πεδία, ένα για το email του χρήστη και ένα για τον κωδικό του. Τα δύο αυτά πεδία πρέπει να συμπληρώσει για να συνδεθεί στην ιστοσελίδα. Το χαρακτηριστικό action μας δείχνει ότι η φόρμα είναι παραμένουσα αφού τα στοιχεία στέλνονται στον εαυτό της. Μέσα από τον κώδικα PHP όμως που εκτελείται όταν ο χρήστης πατήσει το κουμπί “Είσοδος μέλους” ανάλογα με τα στοιχεία που δόθηκαν ο χρήστης αναδρομολογείται στην αντίστοιχη σελίδα.

```
<div id="form_login">
<form name="login" method="post" action="index.php">
<p>E-mail διεύθυνση : <input type="text" name="e_mail" size="40" maxlength="30" /></p>
<p>Κωδικός πρόσβασης : <input type="password" name="pass" size="40" maxlength="40" /></p>
<p><input type="submit" name="enter_login" value="Είσοδος μέλους" /></p>
<div align="center">
<p>
<input type="hidden" name="entered_login" value="TRUE" />
</p>
<p>&nbsp;</p>
</div>
</form>
</div>
```

The image shows a login form with a yellow background and a decorative border. It contains two input fields: the first is labeled "E-mail διεύθυνση:" and the second is labeled "Κωδικός πρόσβασης:". Below the second field is a button labeled "Είσοδος μέλους". At the bottom of the form, there is a small graphic of an open book with a light effect above it.

Εικόνα 4.21: Φόρμα Εισόδου χρήστη στην τοποθεσία.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #table_flash_login { margin-right: 30px; margin-left: 30px; padding-top: 20px; color: #CC9C0D; font-size: 18px; font-family: "Monotype Corsiva"; font-weight: bold; background-image: url(image_login.jpg); background-repeat: no-repeat; }</pre>	<pre>.bookstore #form_login { margin-right: 5px; margin-left: 5px; color: #CC9C0D; font-size: 16px; text-align: center; }</pre>

Πίνακας 4.2: Φύλλα CSS για την μορφοποίηση της φόρμας εισόδου.

- Μορφοποίηση μηνυμάτων αρχικής εικόνας

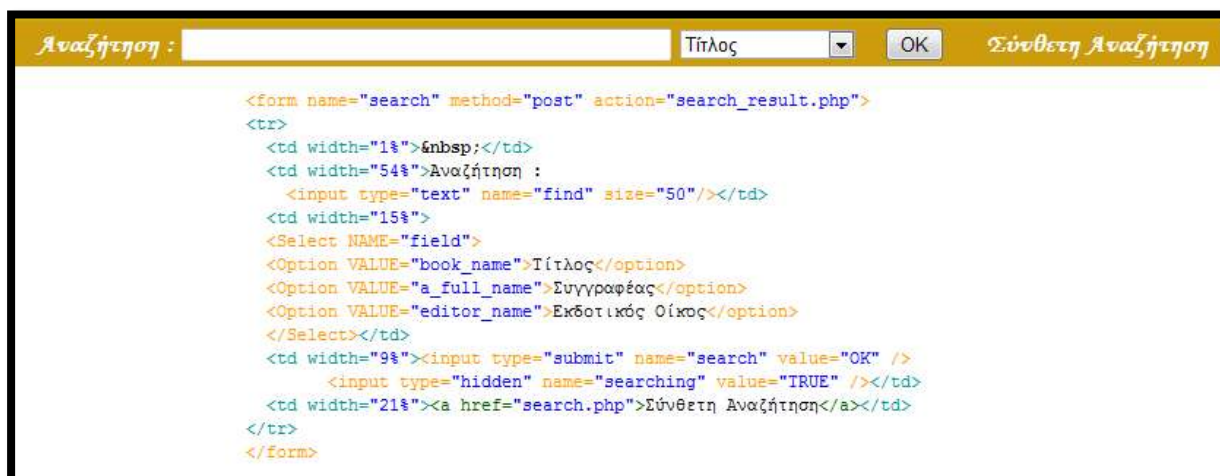


Εικόνα 4.22: Εμφάνιση ιστοσελίδας

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #title_1 { margin-right: 30px; margin-left: 30px; margin-top: 0px; padding-top: 155px; background-image: url(messages_1.jpg); background-repeat: no-repeat; padding-bottom: 20px; font-size: 18px; font-weight: bold; }</pre>	<pre>.bookstore #message { font-family: "Monotype Corsiva"; color: #000000; font-weight: bold; height: 100px; }</pre>

Πίνακας 4.3: Φύλλα CSS για την μορφοποίηση των μηνυμάτων.

- Δημιουργία Φόρμας Αναζήτησης



Εικόνα 4.23: Φόρμα απλής αναζήτησης

Στην απλή φόρμα αναζήτησης ο χρήστης επιλέγει ανάμεσα στον τίτλο, τον συγγραφέα και τον εκδότη για την εύρεση του βιβλίου που επιθυμεί. Όταν ο χρήστης πατήσει το κουμπί “OK” κατευθύνεται στην σελίδα search_result.php (όπως φαίνεται από την τιμή της action) στην οποία εμφανίζονται τα αποτελέσματα της αναζήτησης. Επίσης πατώντας στο link “Σύνθετη Αναζήτηση” μεταφερόμαστε στην σελίδα search.php στην οποία εμφανίζεται η φόρμα την σύνθετης αναζήτησης.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #search { margin-right: 90px; margin-left: 90px; padding-top: 1px; background-color: #CC9C0D; text-align: center; font-family: "Monotype Corsiva"; font-size: 18px; font-weight: bold; }</pre>	<pre>.bookstore #search a { font-family: "Monotype Corsiva"; color: #FFFFFF; font-size: 18px; }</pre>

Πίνακας 4.4: Φύλλα CSS για την μορφοποίηση της φόρμας απλής αναζήτησης

- Δημιουργία Φόρμας Σύνθετης Αναζήτησης

Στην σύνθετη αναζήτηση ο χρήστης μπορεί να συμπληρώσει περισσότερα από ένα στοιχεία για εύρεση του βιβλίου που επιθυμεί. Για τον απλό χρήστη τα αποτελέσματα εμφανίζονται στην σελίδα search_result2.php, για τον χρήστη-μέλος στην σελίδα search_result2_after.php και για τον administrator στην σελίδα search_result2_admin.php

Τίτλος :	<input type="text"/>
Συγγραφέας :	<input type="text"/>
Εκδόσεις :	<input type="text"/>
Κατηγορία :	<input type="text"/>

```

<form name="search_erea" method="post" action="search_result2_after.php">
<table width="100%" border="0">
  <tr>
    <td width="23%" height="73">&nbsp;</td>
    <td width="77%">&nbsp;</td>
  </tr>
  <tr>
    <td><p>Τίτλος :</p></td>
    <td align="right"><p><input type="text" name="book_name" size="80" maxlength="80" /></p></div></td>
  </tr>
</table>

<table width="100%" border="0">
  <tr>
    <td width="19%" height="34"><p>Συγγραφέας :</p></td>
    <td width="81%"><input type="text" name="author_name" size="72" maxlength="60" /></td>
  </tr>
  <tr>
    <td height="44"><p>Εκδόσεις :</p></td>
    <td><input type="text" name="editor_name" size="72" maxlength="60" /></td>
  </tr>
  <tr>
    <td><p>Κατηγορία :</p></td>
    <td><input type="text" name="category_name" size="72" maxlength="40" /></td>
  </tr>
</table>

<p>&nbsp;</p>
<p align="right">
  <input type="submit" name="search2" value="Αναζήτηση" />
</p>
<input type="hidden" name="searching2" value="TRUE" />
</form>

```

Εικόνα 4.24: Φόρμα σύνθετης αναζήτησης

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #search_erea { margin-right: 30px; margin-left: 30px; color: #000000; font-family: "Times New Roman", Times, serif; background-color: #000000; background-repeat: repeat-y; background-image: url(background%20img.jpg); font-size: 16px; padding-top: 50px; }</pre>	<pre>.bookstore #search_erea form{ text-align: left; width: 570px; font-weight: bold; margin-right: 30px; margin-left: 315px; }</pre>

- Πίνακας 4.5: Φύλλα CSS για την μορφοποίηση την φόρμα σύνθετης αναζήτησης

- Δημιουργία πίνακα logout για τον χρήστη-μέλος

Κατά την είσοδο του χρήστη-μέλους στη ιστοσελίδα (εμφάνιση σελίδας loggedin.php) εμφανίζεται ο πίνακας που περιέχει την φόρμα εξόδου. Η φόρμα αυτή αποτελείται από ένα κουμπί “Εξοδος μέλους” που όταν ο χρήστης το πατήσει μέσω του κώδικα PHP κατευθύνεται στην σελίδα logout.php. Η σελίδα logout.php είναι ίδια με την index.php μόνο που περιέχει και ένα μήνυμα αποχαιρετισμού.

Επίσης, ο πίνακας που φαίνεται και στην **Εικόνα 4.25** εκτός από την φόρμα εξόδου περιέχει και τρία link. Ένα προς την σελίδα αλλαγής στοιχείων του χρήστη (allagh_stoixeiwn.php), ένα προς το καλάθι αγορών του χρήστη (kalathi_agorwn.php) και ένα προς την λίστα των αγαπημένων βιβλίων του χρήστη (wishlist.php).



Εικόνα 4.25: Σύνδεσμοι χρήστη μέλους

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #table_flash_logout { margin-right: 30px; margin-left: 30px; padding-top: 20px; font-weight: bold; background-image: url(image_logout.jpg); background-repeat: no-repeat; }</pre>	<pre>.bookstore #table_flash_logout a { color: #c05711; font-size: 19px; } .bookstore #table_flash_logout a:hover { color: #c05711; font-size: 19px; text-decoration: underline; }</pre>

Πίνακας 4.6: Φύλλα CSS για την μορφοποίηση του πίνακα εξόδου

```

<table width="100%" border="0">
  <tr>
    <td colspan="3"><div align="center">
      <a href="allagh_stoixeiwn.php">Αλλαγή στοιχείων λογαριασμού</a>
    </div></td>
  </tr>
  <tr>
    <td width="18%"><div align="center"></div></td>
    <td width="18%"></td>
    <td width="64%"><a href="kalathi_agorwn.php">Καλάθι Αγορών</a></td>
  </tr>
  <tr>
    <td><div align="center"></div></td>
    <td></td>
    <td><a href="wishlist.php">Wishlist</a></td>
  </tr>
  <tr>
    <td colspan="3">
      <div id="form_logout">
        <form name="logout" method="post" action="logout.php">
          <p><input type="submit" name="enter_logout" value="Έξοδος μέλους" /></p>
          <div align="center">
            <p><input type="hidden" name="entered_logout" value="TRUE" /></p>
          </div>
        </form>
      </div>
    </td>
  </tr>
</table>

```

Εικόνα 4.26: Κώδικας HTML του πίνακα εξόδου.

- Δημιουργία πίνακα logout για τον administrator

Κατά την είσοδο του administrator εμφανίζεται ο πίνακας της **Εικόνας 4.27** που περιέχει την ίδια φόρμα εξόδου με αυτή του χρήστη μέλους. Επίσης, ο πίνακας αυτός περιέχει τρία link. Ένα προς την αλλαγή των στοιχείων του διαχειριστή (allagh_stoixeiwn_admin.php), ένα προς την φόρμα προσθήκης βιβλίων (add_book.php) και ένα προς την σελίδα όπου εμφανίζονται οι εγγεγραμμένοι χρήστες της σελίδας (view_users.php). Τα φύλλα CSS είναι ίδια με εκείνα που χρησιμοποιήθηκαν για την εμφάνιση του πίνακα logout του χρήστη-μέλους.



Εικόνα 4.27: Σύνδεσμοι του administrator


```

<table width="100%" border="0">

<tr>
<td colspan="3"><div align="center">
<a href="allagh_stoixeewn_admin.php">Αλλαγή στοιχείων διαχειριστή</a>
</div></td>
</tr>

<tr>
<td width="4%"><div align="center"></div>
<div align="center"></div></td>
<td width="26%"></td>
<td width="70%"><a href="add_book.php">Προσθήκη Βιβλίου</a><a href="add_book.php"></a></td>
</tr>

<tr>
<td><div align="center"></div></td>
<td></td>
<td><a href="view_users.php">Εμφάνιση χρηστών</a><a href="view_users.php"></a></td>
</tr>

<tr>
<td colspan="3">
<div id="form_logout">
<form name="logout" method="post" action="logout.php">
<p><input type="submit" name="enter_logout" value="Εξοδος μέλους" /></p>
<div align="center">
<p><input type="hidden" name="entered_logout" value="TRUE" /></p>
</div>
</form>
</div>
</td>
</tr>
</table>

```

Εικόνα 4.28: Κώδικας HTML του πίνακα εξόδου

- **Δημιουργία Πίνακα Κατηγοριών**

Τα link που αντιστοιχούν στην κάθε κατηγορία αλλάζουν ανάλογα με την ιδιότητα του χρήστη. Αν είναι απλός χρήστης τα αρχεία που αντιπροσωπεύουν την κάθε κατηγορία είναι:

- Εκπαίδευση – education.php
- Λογοτεχνία – logotexnia.php
- Ψυχολογία – psychology.php
- Θρησκεία – riligion.php
- Φιλοσοφία – philosophy.php
- Ιστορία, Γεωγραφία – history.php
- Μαγειρική – cooking.php

Αν έχει συνδεθεί ως administrator τα αρχεία που αντιπροσωπεύουν την κάθε κατηγορία είναι:

- Εκπαίδευση – education_admin.php
- Λογοτεχνία – logotexnia_admin.php
- Ψυχολογία – psychology_admin.php
- Θρησκεία – religion_admin.php
- Φιλοσοφία – philosophy_admin.php
- Ιστορία, Γεωγραφία – history_admin.php
- Μαγειρική – cooking_admin.php

Αν έχει συνδεθεί ως χρήστης-μέλος εμφανίζονται τα ακόλουθα αρχεία:

- Εκπαίδευση – education_after.php
- Λογοτεχνία – logotexnia_after.php
- Ψυχολογία – psychology_after.php
- Θρησκεία – religion_after.php
- Φιλοσοφία – philosophy_after.php
- Ιστορία, Γεωγραφία – history_after.php
- Μαγειρική – cooking_after.php

```
<div id="categories">
  <table width="250" height="281" border="0">
  <tr>
  <td height="55"><span class="style3">Κατηγορίες Βιβλίων</span></td>
  </tr>
  <tr>
  <td><a href="education.php">Εκπαίδευση</a></td>
  </tr>
  <tr>
  <td><a href="logotexnia.php">Λογοτεχνία</a></td>
  </tr>
  <tr>
  <td><a href="psychology.php">Ψυχολογία</a></td>
  </tr>
  <tr>
  <td><a href="religion.php">Θρησκεία</a></td>
  </tr>
  <tr>
  <td><a href="philosophy.php">Φιλοσοφία</a></td>
  </tr>
  <tr>
  <td><a href="history.php">Ιστορία, Γεωγραφία</a></td>
  </tr>
  <tr>
  <td><a href="cooking.php">Μαγειρική</a></td>
  </tr>
  <tr>
  </tr>
  </table>
</div> <!-- end categories -->
```

Κατηγορίες Βιβλίων
Εκπαίδευση
Λογοτεχνία
Ψυχολογία
Θρησκεία
Φιλοσοφία
Ιστορία, Γεωγραφία
Μαγειρική

Εικόνα 4.29: Πίνακας Κατηγοριών

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre> .bookstore #categories { float: left; width: 270px; margin-left: 10px; color: #FFFFFF; font-family: "Times New Roman", Times, serif; font-size: 16px; font-weight: normal; } </pre>	<pre> .bookstore #categories a { font-family: "Times New Roman", Times, serif; font-size: 16px; color: #F5F1CE; } .bookstore #categories a:hover { font-family: "Times New Roman", Times, serif; font-size: 16px; text-decoration: none; color: #C10000; } </pre>

Πίνακας 4.7: Φύλλα CSS για την μορφοποίηση του πίνακα κατηγοριών

- Δημιουργία Φόρμας Επικοινωνίας



```

<form name="contact" method="post" action="contact.php">
<p><span class="red">*</span> Όνομα : <input type="text" name="first_name" size="53" maxlength="20"
value="<?php if (isset($_POST['first_name'])) echo $_POST['first_name']; ?>" /></p>

<p><span class="red">*</span> Επώνυμο : <input type="text" name="last_name" size="53" maxlength="40"
value="<?php if (isset($_POST['last_name'])) echo $_POST['last_name']; ?>" /></p>

<p><span class="red">*</span> Email : <input type="text" name="e-mail" size="53" maxlength="30"
value="<?php if (isset($_POST['e-mail'])) echo $_POST['e-mail']; ?>" /></p>

<p><span class="red">*</span> Σχόλια : <textarea name="comments" rows="10" cols="50">
<?php if (isset($_POST['comments'])) echo $_POST['comments']; ?></textarea></p>

<p>&nbsp;</p>
<p align="right">
    <input type="submit" name="contact" value="Αποστολή" />
</p>
<p align="right">&nbsp;</p>
<input type="hidden" name="contacted" value="TRUE" />
</form>

```

Εικόνα 4.30: Φόρμα Επικοινωνίας

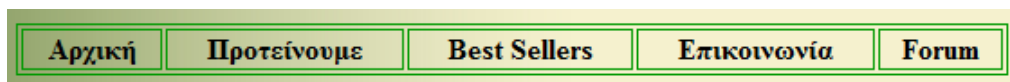
Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #contact { margin-right: 30px; margin-left: 30px; background-color: #000000; background-repeat: repeat-y; background-image: url(background%20img.jpg); padding-top: 50px; color: #000000; }</pre>	<pre>.bookstore #contact form{ text-align: right; width: 570px; font-weight: bold; margin-right: 30px; margin-left: 280px; font-size: 16px; font-family: "Times New Roman", Times, serif; color: #000000; padding-top: 20px; padding-bottom: 130px; }</pre>

Πίνακας 4.8: Φύλλα CSS για την μορφοποίηση της φόρμας επικοινωνίας

- **Πίνακα Τέλους**

Η κάθε ιστοσελίδα πρέπει να έχει ένα τέλος. Οι περισσότερες ιστοσελίδες του Διαδικτύου στο τέλος εμφανίζουν διάφορους χορηγούς. Στο δικό μας ιστοχώρο στο τέλος κάθε σελίδας υπάρχει ένας πίνακας που έχει ίδιες στήλες με αυτές του μενού.

Εμφάνιση πίνακα απλού χρήστη:



```
<div id="end">
<table width="550" border="1" bordercolor="#009900">
  <tr>
    <td><div align="center"><a href="index.php">Αρχική</a></div></td>
    <td><div align="center"><a href="suggestions.php">Προτείνουμε</a></div></td>
    <td><div align="center"><a href="best_sellers.php">Best Sellers</a></div></td>
    <td><div align="center"><a href="contact.php">Επικοινωνία</a></div></td>
    <td><div align="center"><a href="forum.php">Forum</a></div></td>
    <td><div align="center"><a href="register.php">Εγγραφή</a></div></td>
  </tr>
</table>
</div>
```

Εικόνα 4.31: Εμφάνιση πίνακα απλού χρήστη

Εμφάνιση πίνακα χρήστη-μέλους:



```

<div id="end">
  <table width="550" border="1" bordercolor="#009900">
    <tr>
      <td><div align="center"><a href="loggedin.php">Αρχική</a></div></td>
      <td><div align="center"><a href="suggestions_after.php">Προτείνουμε</a></div></td>
      <td><div align="center"><a href="best_sellers_after.php">Best Sellers</a></div></td>
      <td><div align="center"><a href="contact_after.php">Επικοινωνία</a></div></td>
      <td><div align="center"><a href="forum_after.php">Forum</a></div></td>
    </tr>
  </table>
</div>

```

Εικόνα 4.32: Εμφάνιση πίνακα χρήστη-μέλους

Εμφάνιση πίνακα administrator:



```

<div id="end">
  <table width="550" border="1" bordercolor="#009900">
    <tr>
      <td><div align="center"><a href="admin.php">Αρχική</a></div></td>
      <td><div align="center"><a href="suggestions_admin.php">Προτείνουμε</a></div></td>
      <td><div align="center"><a href="best_sellers_admin.php">Best Sellers</a></div></td>
      <td><div align="center"><a href="forum_admin.php">Forum</a></div></td>
      <td><div align="center"><a href="pinakas_agorwn.php">Πίνακας Αγορών</a></div></td>
    </tr>
  </table>
</div>

```

Εικόνα 4.33: Εμφάνιση πίνακα administrator

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre> .bookstore #end { padding-left: 275px; color: #000000; background-image: url(background%20img.jpg); padding-bottom: 20px; font-weight: bold; margin-right: 30px; margin-left: 30px; padding-top: 30px; background-repeat: repeat-y; } </pre>	<pre> .bookstore #end a { font-family: "Times New Roman", Times, serif; color: #000000; font-size: 16px; } .bookstore #end a:hover { font-family: "Times New Roman", Times, serif; color: #C10000; font-size: 16px; text-decoration: none; } </pre>

Πίνακας 4.9: Φύλλα CSS για την μορφοποίηση του πίνακα

4.3 Δημιουργία Flash Animation

Το Flash μπορεί να υπολογίζει τη διαφορά ανάμεσα σε δύο διαφορετικές καταστάσεις ενός αντικείμενου και να δημιουργεί αυτόματα τα ενδιάμεσα στάδια. Αν έχουμε, δηλαδή, ένα αντικείμενο στο αριστερό άκρο ενός καμβά και το μετακινήσουμε στο δεξί άκρο του, μπορούμε να πούμε στο Flash “βρες τι άλλαξε και φτιάξε μου μια κίνηση”. Το ίδιο μπορεί να γίνει με το μέγεθος ενός αντικείμενου, το χρώμα του κ.λπ. Έτσι, η δημιουργία κίνησης μέσα από το Flash είναι, τελικά, αρκετά εύκολη.

Βήματα Υλοποίησης Flash Animation Script

- Εισαγωγή εικόνων στη Βιβλιοθήκη (Library)

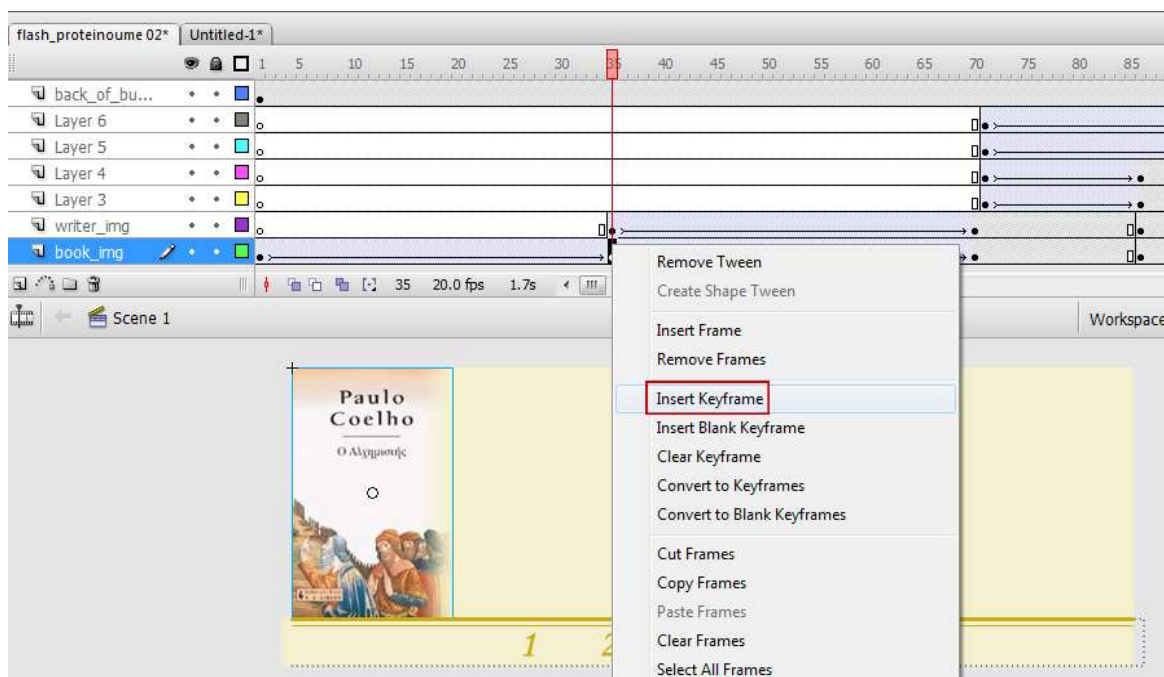


Εικόνα 4.34: Εισαγωγή εικόνων στο Flash CS3

- Αφού εισάγουμε την εικόνα που επιθυμούμε στη Βιβλιοθήκη έπειτα την τοποθετούμε στη Σκηνή μας. Αυτόματως καταλαμβάνει το πρώτο Layer του Timeline μεταβάλλοντας το πρώτο frame σε keyframe.

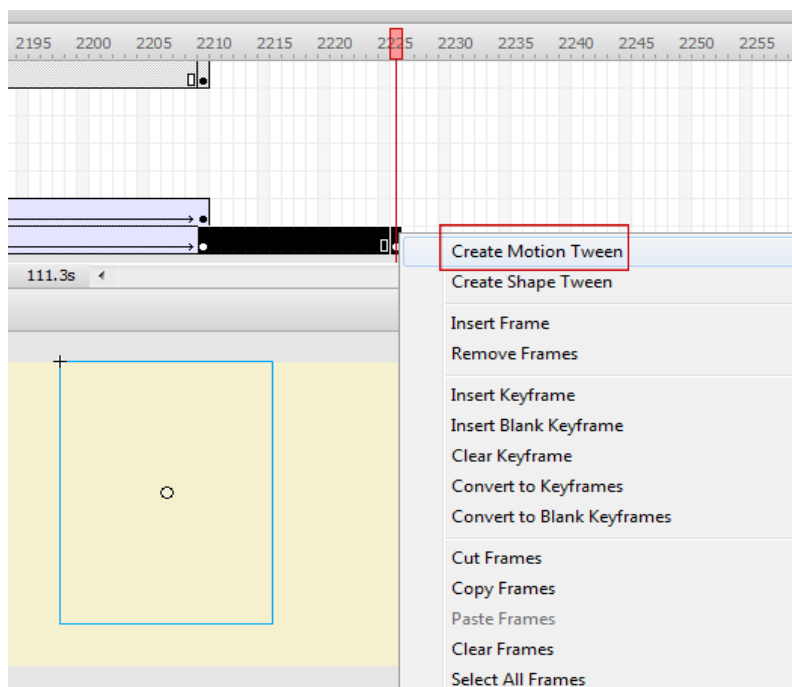
Για να κατανοήσουμε τον τρόπο λειτουργίας του Flash, θα πρέπει να αντιληφθούμε ότι δημιουργούμε πάνω στο χρόνο. Ο χρόνος κυλάει πάνω στο Timeline και μετριέται σε frames. Το κάθε Layer περιέχει το αντικείμενο που θέλουμε να εμφανιστεί. Για να μπορεί να είναι ορατό ένα αντικείμενο θα πρέπει να βρίσκεται μέσα στα πλαίσια της σκηνής.

Αν το αντικείμενό μας είναι μια εικόνα για να της δώσουμε κίνηση θα πρέπει να καθορίσουμε στο χρόνο το σημείο εκκίνησης και το σημείο ακινητοποίησης. Όπως προαναφέραμε ο χρόνος μετριέται σε frames. Έτσι το frame από το οποίο θα ξεκινάει η κίνηση το μεταβάλω σε keyframe, καθώς και το frame στο οποίο θα σταματάει.



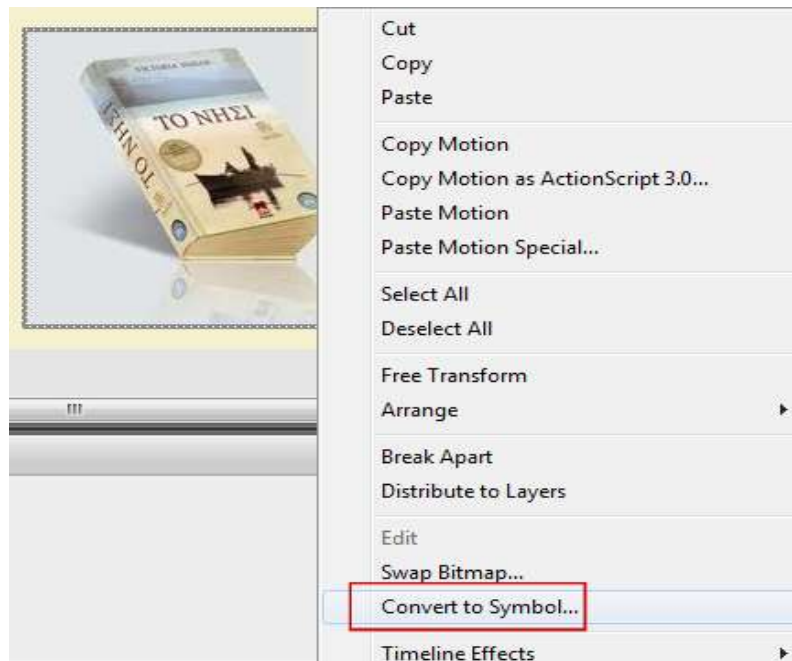
Εικόνα 4.35: Εισαγωγή keyframe

- Αφού εισάγαμε τα δύο keyframes, τσεκάρουμε την απόσταση από το ένα στο άλλο και επιλέγουμε “Create Motion Tween”. Με αυτό τον τρόπο προσθέτουμε κίνηση σε ένα αντικείμενο της σκηνής μας. Αν εκτελέσαμε σωστά την διεργασία αυτή τότε ένα βέλος μας δείχνει την αρχή και το τέλος της μεταφοράς.

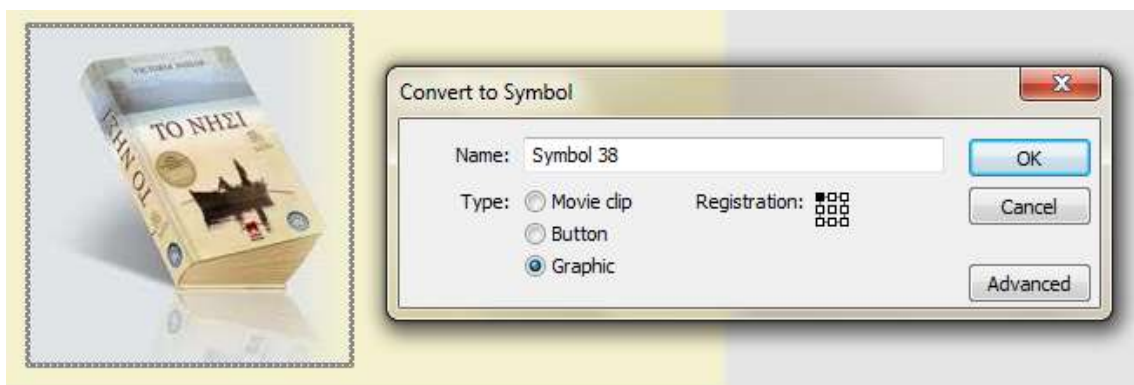


Εικόνα 4.36: Δημιουργία κίνησης

- Για να εμφανίζουμε και να εξαφανίζουμε σταδιακά μία εικόνα πρέπει πρώτα να την μετατρέψουμε σε σύμβολο. Αυτό γίνεται με την επιλογή “Convert to Symbol”.



Εικόνα 4.37: Μετατροπή εικόνας σε σύμβολο



Εικόνα 4.38: Επιλογές συμβόλων

Όπως φαίνεται και από την Εικόνα, το παράθυρο “ Convert to Symbol” μας δίνει τρεις επιλογές για την μετατροπή ενός αντικειμένου:

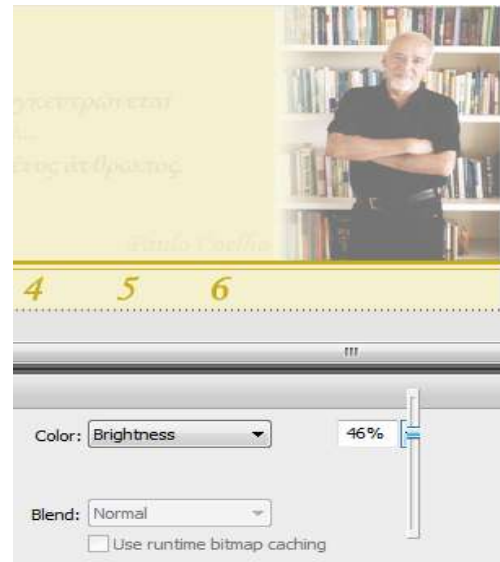
- Movie clip
- Button
- Graphic

Αυτός είναι και ο τρόπος που δημιουργούμε ένα κουμπί. Στην περίπτωση της εικόνας επιλέγουμε Graphic.

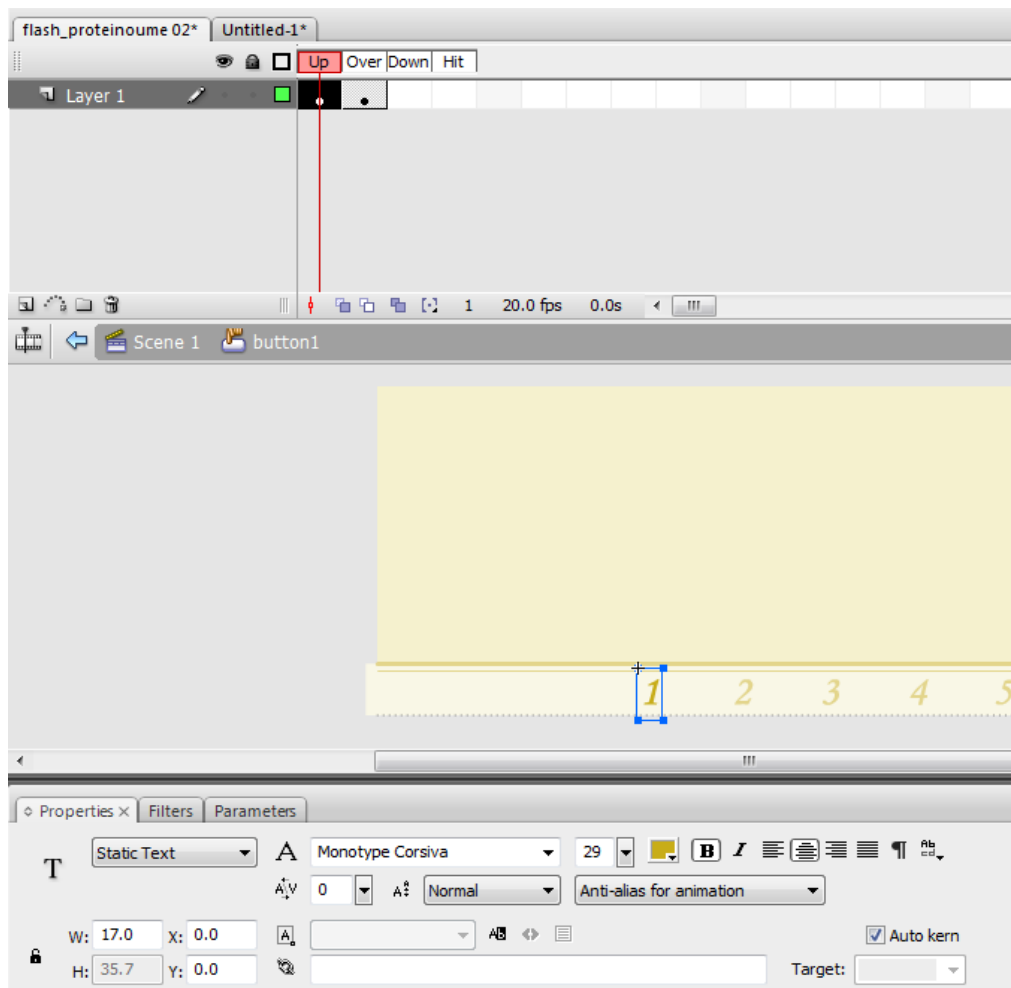
Μετατρέποντας μια εικόνα σε σύμβολο, στην καρτέλα των Properties εμφανίζεται η επιλογή Brightness. Αλλάζοντας την τιμή του Brightness η εικόνα χάνεται.

Η δημιουργία ενός button γίνεται με την επιλογή “Convert to Symbol”, όπως είδαμε πριν. Κάθε κουμπί που δημιουργείται στο Flash έχει 4 frame:

- Up – Αρχικά χαρακτηριστικά
- Over – Καθορισμός της εμφάνισης ενός κουμπιού όταν περάσει από πάνω του ο κέρσορας.
- Down – Καθορισμός της εμφάνισης όταν θα φύγει ο κέρσορας πάνω από ένα κουμπί.
- Hit – Τα χαρακτηριστικά που θα πάρει ένα κουμπί όταν πατηθεί.



Εικόνα 4.39: Brightness













Εικόνα 4.40: buttons in Flash

4.4 Δημιουργία Βάσης Δεδομένων – Bookstore

4.4.1 Ονομασία Στοιχείων

Κατά την δημιουργία μιας βάσης δεδομένων, θα πρέπει να προσέξουμε οι πίνακές μας να μην έχουν το ίδιο όνομα, όπως επίσης και οι στήλες ενός πίνακα. Μπορούμε όμως να δώσουμε το ίδιο όνομα σε στήλες διαφορετικών πινάκων.

Η βάση δεδομένων που δημιουργήθηκε για τις ανάγκες της ιστοσελίδας, με όνομα “Bookstore”, περιέχει δέκα πίνακες.

-  authors => Συγγραφείς
-  books => Βιβλία
-  categories => Κατηγορίες βιβλίων
-  editors => Εκδότες
-  kalathi => Καλάθι αγορών
-  messages => Μηνύματα Forum
-  orders => Χρήστες που μετέβηκαν σε αγορά βιβλίων.
-  order_contents => Παραγγελίες χρηστών
-  users => Στοιχεία χρηστών
-  wishlist => Λίστα Αγαπημένων βιβλίων

4.4.2 Επιλογή τύπου στηλών

Κατά την δημιουργία ενός πίνακα η MySQL απαιτεί την δήλωση των πληροφοριών που θα περιέχει κάθε στήλη. Υπάρχουν τρεις κύριοι τύποι δεδομένων:

- Κείμενο (δηλαδή, συμβολοσειρά)
- Αριθμοί
- Ημερομηνίες και ώρες

Η σωστή επιλογή για τις στήλες όχι μόνο καθορίζει τι πληροφορίες θα αποθηκεύονται σε αυτές και πως, αλλά και επηρεάζει τη συνολική απόδοση της βάσης δεδομένων. Πολλοί από τους τύπους δέχονται μία προαιρετική παράμετρο (μήκος) που περιορίζει το μέγεθός τους. Για την καλύτερη απόδοση πρέπει να βάζουμε περιορισμούς στο πλήθος των δεδομένων που μπορούν να αποθηκεύονται σε κάθε στήλη. Το μήκος που καθορίζεται πρέπει να αντιστοιχεί στη μέγιστη πιθανή αριθμητική τιμή ή στη μεγαλύτερη πιθανή συμβολοσειρά που πρόκειται να αποθηκευτεί στη στήλη.

Τύποι δεδομένων της MySQL	
ΤΥΠΟΣ	ΠΕΡΙΓΡΑΦΗ
CHAR[μήκος]	Πεδίο σταθερού μήκους
VARCHAR[μήκος]	Πεδίο μεταβλητού μήκους
INT[μήκος]	Ακέραιος
MEDIUMINT [μήκος]	Ακέραιος μεσαίου μεγέθους
TINYINT[μήκος]	Ακέραιος μικρού μεγέθους
DECIMAL[μήκος, δεκαδικά ψηφία]	Αριθμός DOUBLE αποθηκευμένος ως συμβολοσειρά, σταθερής υποδιαστολής
DATE	Ημερομηνία στη μορφή YYYY-MM-DD
DATETIME	Ημερομηνία στη μορφή YYYY-MM-DD HH:MM:SS
TIMESTAMP	Ημερομηνία στη μορφή YYYYMMDDHHMMSS

Πίνακας 4.10: Τύποι δεδομένων της MySQL

Τα δεδομένα τύπου CHAR αποθηκεύονται ως συμβολοσειρές με το πλήρες μήκος τους (αν υπάρχουν άδειες θέσεις, αυτές συμπληρώνονται με κενά διαστήματα και κατά την ανάκτηση της τιμής τους από τη βάση δεδομένων, τα τυχόν κενά αφαιρούνται). Αντίθετα τα δεδομένα τύπου VARCHAR καταλαμβάνουν χώρο ίσο με το μήκος της ίδιας της συμβολοσειράς. Κατά αυτόν τον τρόπο αν ένα πεδίο συμβολοσειράς θα έχει σταθερό μήκος χρησιμοποιούμε τον τύπο CHAR, αλλιώς χρησιμοποιούμε τον τύπο VARCHAR.

Επίσης, ο τύπος πεδίου TIMESTAMP ενημερώνεται αυτόματα με την τρέχουσα ημερομηνία και ώρα όταν εκτελείται η εντολή INSERT ή UPDATE.

Άλλες ιδιότητες στηλών

Κάθε στήλη ανεξάρτητα από τον τύπο της, μπορεί να οριστεί ως NOT NULL. Η τιμή NULL στις βάσεις δεδομένων και στον προγραμματισμό σημαίνει ότι το πεδίο δεν έχει τιμή. Σε μια σωστά σχεδιασμένη βάση δεδομένων, κάθε στήλη κάθε γραμμής σε κάθε πίνακα πρέπει έχει τιμή. Για να υποχρεώσουμε ένα πεδίο να έχει τιμή, προσθέτουμε την περιγραφή NOT NULL στον τύπο της στήλης του.

Όταν δημιουργούμε έναν πίνακα μπορούμε να ορίσουμε μια προεπιλεγμένη τιμή για οποιαδήποτε στήλη, ανεξάρτητα από τον τύπο της. Προκαθορίζοντας μια προεπιλεγμένη τιμή δεν θα χρειάζεται να την πληκτρολογούμε κάθε φορά που θα προσθέτουμε μια καινούρια εγγραφή.

Οι αριθμητικοί τύποι μπορούν να δηλωθούν ως UNSIGNED , και έτσι περιορίζονται τα δεδομένα μόνο σε θετικούς αριθμούς και 0. Αυτό διπλασιάζει στην ουσία το εύρος των θετικών αριθμών που μπορούν να αποθηκευθούν.

Προσθέτοντας την περιγραφή AUTO_INCREMENT σε μια στήλη η MySQL θα χρησιμοποιεί αυτόματα στο πεδίο τον επόμενο μεγαλύτερο αριθμό για κάθε επόμενη εγγραφή που θα προστίθεται.

Ευρετήρια

Τα ευρετήρια είναι ένα ειδικό σύστημα που χρησιμοποιούν οι βάσεις δεδομένων για τη βελτίωση της απόδοσης των ερωτημάτων επιλογής (SELECT). Τα ευρετήρια τοποθετούνται σε μία ή περισσότερες στήλες οποιαδήποτε τύπου, λέγοντας στη MySQL να προσέξει ιδιαίτερα αυτές τις στήλες. Επίσης, εξυπηρετούν στις αναζητήσεις που εκτελούνται συχνά στο ίδιο σύνολο τιμών (πχ. όνομα και επώνυμο).

Η MySQL έχει τέσσερα είδη ευρετηρίων:

- INDEX (τυπικό)
- UNIQUE (μοναδικό – κάθε γραμμή πρέπει να έχει διαφορετική τιμή στήλης)
- FULLTEXT (για αναζήτηση πλήρους κειμένου)
- PRIMARY KEY

Μια στήλη θα πρέπει να έχει πάντα το ίδιο μοναδικό ευρετήριο και όχι περισσότερα.

Χρήση διαφορετικών τύπων πινάκων

Η MySQL υποστηρίζει πολλούς διαφορετικούς τύπους πινάκων. Κάθε τύπος πίνακα υποστηρίζει διαφορετικές δυνατότητες, έχει τους δικούς του περιορισμούς και διαφέρει στην απόδοση ανάλογα με τις περιστάσεις.

Ο πιο σημαντικός τύπος πίνακα είναι ο MyISAM, ο προεπιλεγμένος τύπος πίνακα για όλα τα λειτουργικά συστήματα εκτός από τα Windows. Οι πίνακες MyISAM είναι ιδανικοί για τις περισσότερες εφαρμογές και χειρίζεται τα ερωτήματα INSERT και SELECT πολύ γρήγορα.

Η επόμενη πιο συνηθισμένη μηχανή αποθήκευσης είναι η InnoDB, ο προεπιλεγμένος τύπος πίνακα για τα Windows. Οι πίνακες InnoDB είναι κατάλληλοι για συναλλαγές και εκτελούν ερωτήματα ενημέρωσης χωρίς πρόβλημα. Το μειονέκτημα της απέναντι στη MyISAM είναι ότι χρειάζεται περισσότερο χώρο στο δίσκο του διακομιστή, λόγω του ότι είναι πιο αργή. Επίσης οι πίνακες InnoDB δεν υποστηρίζουν FULLTEXT.

Στη δημιουργία των πινάκων της βάσης δεδομένων “Bookstore” οι πίνακες είναι τύπου InnoDB εκτός από τον πίνακα των συγγραφέων ο οποίος περιέχει ευρετήριο FULLTEXT.

4.4.3 Περιγραφή Πινάκων

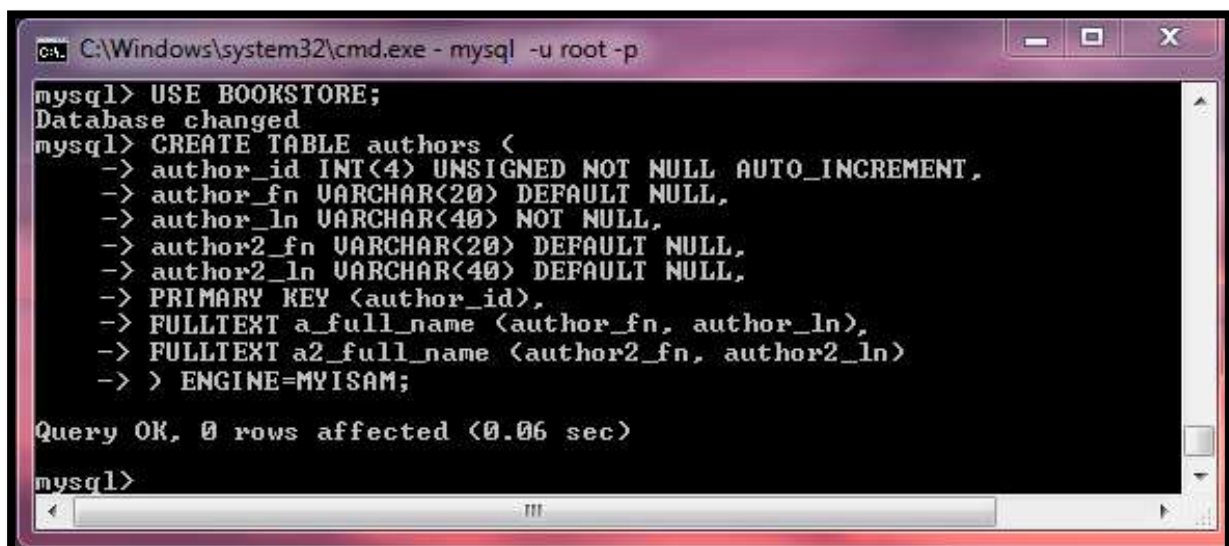
Πίνακας authors

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
author_id	Πρωτεύων κλειδί του πίνακα
author_fn	Μικρό όνομα πρώτου συγγραφέα
author_ln	Επώνυμο πρώτου συγγραφέα
author2_fn	Μικρό όνομα δεύτερου συγγραφέα
author2_ln	Επώνυμο δεύτερου συγγραφέα

Πίνακας 4.11: Στήλες του πίνακα authors

Όπως βλέπουμε και από την παρακάτω εικόνα, το πρωτεύων κλειδί (author_id) είναι ένας θετικός ακέραιος, οποίος αυξάνεται με κάθε εγγραφή που προστίθεται στον πίνακα authors (λόγω της περιγραφής AUTO_INCREMENT).

Το μόνο πεδίο που πρέπει να έχει τιμή, εκτός από το πρωτεύων κλειδί που δεν μπορεί να είναι NULL, είναι το Επώνυμο του πρώτου συγγραφέα. Όλα τα άλλα πεδία κατά τη δημιουργία του πίνακα τους δόθηκε η τιμή DEFAULT NULL, η οποία αλλάζει μόνο όταν εισάγουμε τα ονόματα των συγγραφέων από την φόρμα προσθήκης βιβλίων.

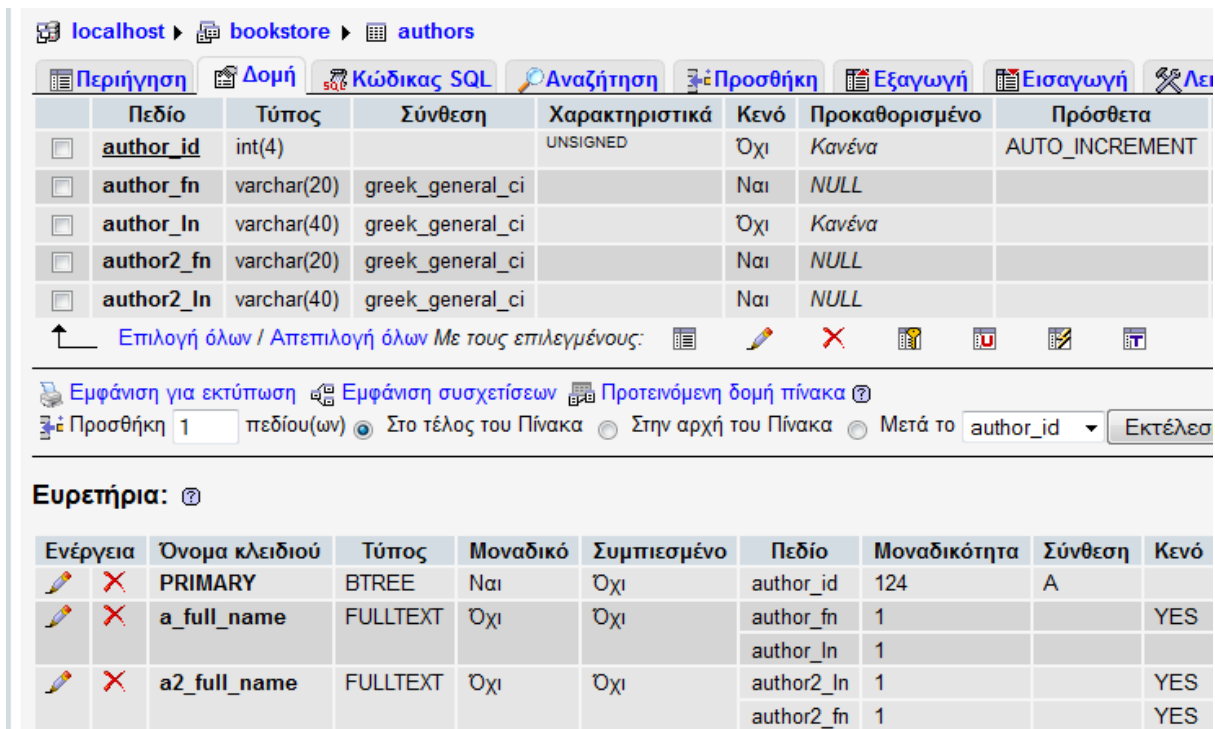


```
C:\Windows\system32\cmd.exe - mysql -u root -p
mysql> USE BOOKSTORE;
Database changed
mysql> CREATE TABLE authors (
  -> author_id INT(4) UNSIGNED NOT NULL AUTO_INCREMENT,
  -> author_fn VARCHAR(20) DEFAULT NULL,
  -> author_ln VARCHAR(40) NOT NULL,
  -> author2_fn VARCHAR(20) DEFAULT NULL,
  -> author2_ln VARCHAR(40) DEFAULT NULL,
  -> PRIMARY KEY (author_id),
  -> FULLTEXT a_full_name (author_fn, author_ln),
  -> FULLTEXT a2_full_name (author2_fn, author2_ln)
  -> ) ENGINE=MYISAM;

Query OK, 0 rows affected (0.06 sec)

mysql>
```

Εικόνα 4.41: Δημιουργία του πίνακα authors



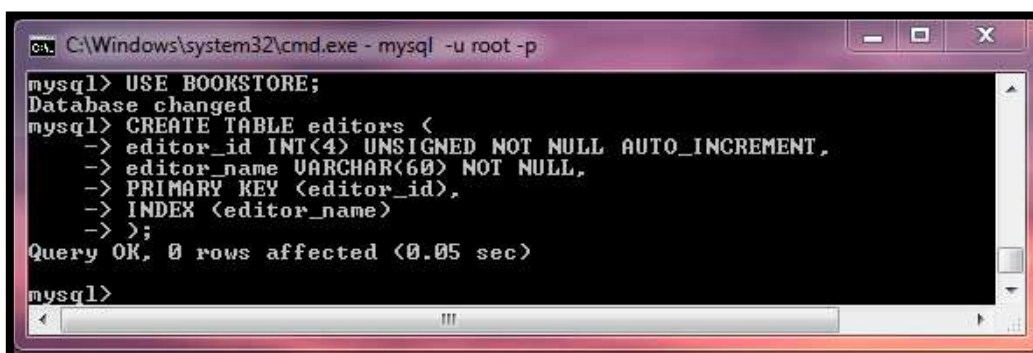
Εικόνα 4.42: Εμφάνιση στηλών στην phpMyAdmin

Στην σύνθεση των ονομάτων επιλέγουμε την περιγραφή `greek_general_ci` έτσι ώστε να ενημερώνεται η MySQL για την γλώσσα που θα έχουν οι εγγραφές. Στη συγκεκριμένη εντολή επιτρέπονται μόνο τα Ελληνικά και Αγγλικά. Αν θέλουμε να συμπεριλάβουμε περισσότερες γλώσσες επιλέγουμε την σύνθεση `utf8_general_ci`.

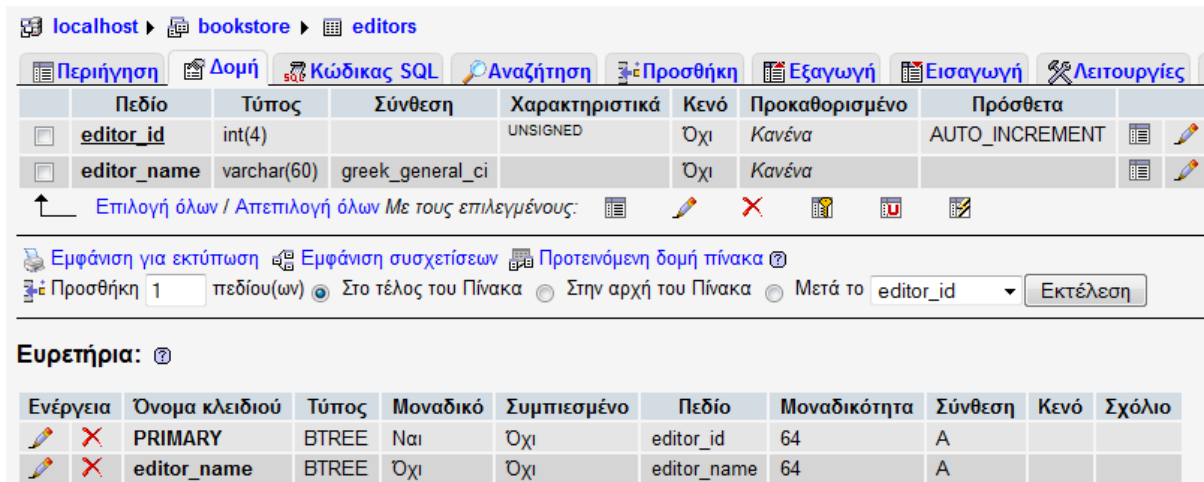
Πίνακας editors

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
editor_id	Πρωτεύων κλειδί του πίνακα
editor_name	Όνομα του εκδότη

Πίνακας 4.12: Στήλες του πίνακα editors



Εικόνα 4.43: Δημιουργία του πίνακα editors



Εικόνα 4.44: Εμφάνιση στηλών στην phpMyAdmin

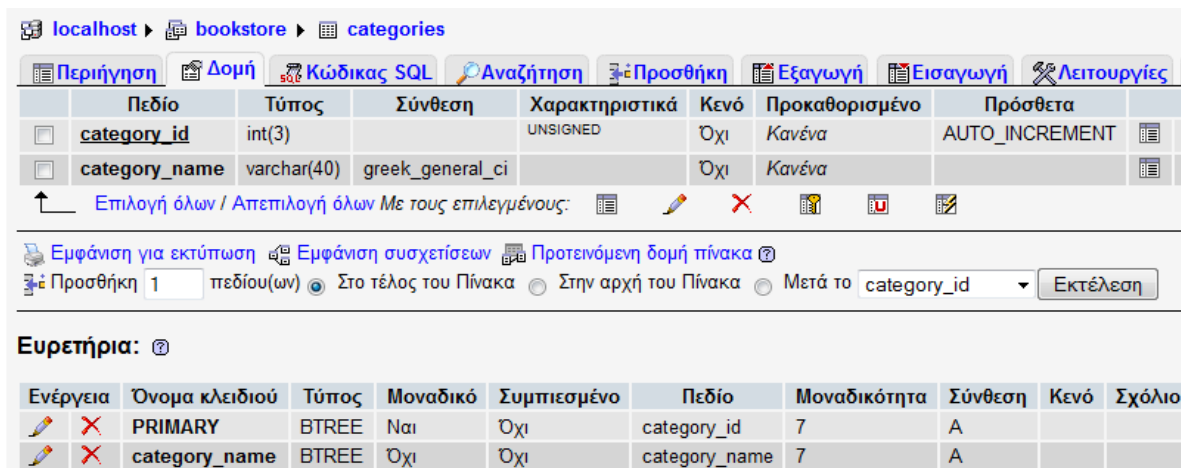
Πίνακας categories

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
category_id	Πρωτεύων κλειδί του πίνακα
category_name	Όνομα Κατηγορίας

Πίνακας 4.13: Στήλες του πίνακα categories



Εικόνα 4.45: Δημιουργία του πίνακα categories



Εικόνα 4.46: Εμφάνιση στηλών στην phpMyAdmin

Πίνακας books

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
book_id	Πρωτεύων κλειδί του πίνακα
author_id	Ξένο κλειδί του πίνακα authors
editor_id	Ξένο κλειδί του πίνακα editors
category_id	Ξένο κλειδί του πίνακα categories
book_name	Τίτλος βιβλίου
price	Τιμή βιβλίου
ISBN	Κωδικός βιβλίου
description	Περιγραφή
image_name_big	Όνομα μεγάλης εικόνας
image_name_small	Όνομα μικρής εικόνας
url	URL κατηγορίας βιβλίου – Για τον απλό χρήστη
url_after	URL κατηγορίας βιβλίου – Για τον χρήστη-μέλος
url_admin	URL κατηγορίας βιβλίου – Για τον administrator
new_book	Η στήλη αυτή έχει default τιμή 0. Αν της δώσουμε την τιμή 1, το βιβλίο συμπεριλαμβάνεται στις Νέες Κυκλοφορίες.
new_url	URL βιβλίου νέας κυκλοφορίας – Για τον απλό χρήστη
new_url_after	URL βιβλίου νέας κυκλοφορίας – Για τον χρήστη-μέλος
new_url_admin	URL βιβλίου νέας κυκλοφορίας – Για τον administrator
best_book	Η στήλη αυτή έχει default τιμή 0. Αν της δώσουμε την τιμή 1, το βιβλίο συμπεριλαμβάνεται στα Best Sellers.
best_url	URL βιβλίου-best seller – Για τον απλό χρήστη
best_url_after	URL βιβλίου-best seller – Για τον χρήστη-μέλος
best_url_admin	URL βιβλίου-best seller – Για τον administrator
sug_book	Η στήλη αυτή έχει default τιμή 0. Αν της δώσουμε την τιμή 1, το βιβλίο συμπεριλαμβάνεται στα Προτεινόμενα βιβλία.
sug_url	URL προτεινόμενου βιβλίου – Για τον απλό χρήστη

sug_url_after	URL προτεινόμενου βιβλίου – Για τον χρήστη-μέλος
sug_url_admin	URL προτεινόμενου βιβλίου – Για τον administrator

Πίνακας 4.14: Στήλες του πίνακα books

localhost ▶ bookstore ▶ books

	Πεδίο	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα
<input type="checkbox"/>	book_id	int(4)		UNSIGNED	Όχι	Κανένα	AUTO_INCREMENT
<input type="checkbox"/>	author_id	int(4)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/>	editor_id	int(4)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/>	category_id	int(3)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/>	book_name	varchar(80)	greek_general_ci		Όχι	Κανένα	
<input type="checkbox"/>	price	decimal(5,2)			Όχι	Κανένα	
<input type="checkbox"/>	ISBN	varchar(15)	greek_general_ci		Όχι	Κανένα	
<input type="checkbox"/>	description	varchar(3200)	greek_general_ci		Ναι	NULL	
<input type="checkbox"/>	image_name_big	varchar(70)	greek_general_ci		Όχι	Κανένα	
<input type="checkbox"/>	image_name_small	varchar(70)	greek_general_ci		Όχι	Κανένα	
<input type="checkbox"/>	url	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	url_after	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	url_admin	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	new_book	tinyint(1)			Ναι	NULL	
<input type="checkbox"/>	new_url	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	new_url_after	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	new_url_admin	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	best_book	tinyint(1)			Ναι	NULL	
<input type="checkbox"/>	best_url	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	best_url_after	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	best_url_admin	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	sug_book	tinyint(1)			Ναι	NULL	
<input type="checkbox"/>	sug_url	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	sug_url_after	varchar(100)	utf8_general_ci		Ναι	NULL	
<input type="checkbox"/>	sug_url_admin	varchar(100)	utf8_general_ci		Ναι	NULL	

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εμφάνιση για εκτύπωση Εμφάνιση συσχέτισεων Προτεινόμενη δομή πίνακα

Προσθήκη 1 πεδίου(ων) Στο τέλος του Πίνακα Στην αρχή του Πίνακα Μετά το book_id Εκτέλεση

Περιήγηση Δομή Κώδικας SQL Αναζήτηση Προσθήκη Εξαγωγή Εισαγωγή Λειτουργίες

Ευρετήρια:

Ενέργεια	Όνομα κλειδιού	Τύπος	Μοναδικό	Συμπίεσμένο	Πεδίο	Μοναδικότητα	Σύνθεση	Κενό	Σχόλιο
	PRIMARY	BTREE	Ναι	Όχι	book_id	126	A		
	author_id	BTREE	Όχι	Όχι	author_id	126	A		
	book_name	BTREE	Όχι	Όχι	book_name	126	A		
	category_id	BTREE	Όχι	Όχι	category_id	14	A		
	editor_id	BTREE	Όχι	Όχι	editor_id	126	A		

Εικόνα 4.47: Εμφάνιση στηλών στην phpMyAdmin

Πίνακας users

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
user_id	Πρωτεύων κλειδί του πίνακα
user_name	Εμφανιζόμενο όνομα χρήστη
first_name	Όνομα
last_name	Επώνυμο
e_mail	Διεύθυνση Ηλεκτρονικού Ταχυδρομείου
pass	Κωδικός Χρήστη
telephone	Τηλέφωνο
fax	Νούμερο - fax
city	Πόλη
zip_code	Ταχυδρομικός Κώδικας
address	Διεύθυνση
registration_date	Ημέρα εγγραφής

Πίνακας 4.15: Στήλες του πίνακα users



```

C:\Windows\system32\cmd.exe - mysql -u root -p
mysql> USE bookstore;
Database changed
mysql> CREATE TABLE users (
-> user_id INT UNSIGNED NOT NULL AUTO_INCREMENT,
-> user_name VARCHAR(20) NOT NULL,
-> first_name VARCHAR(20) NOT NULL,
-> last_name VARCHAR(40) NOT NULL,
-> email VARCHAR(40) NOT NULL,
-> pass CHAR(40) NOT NULL,
-> telephone VARCHAR(30) NOT NULL,
-> fax VARCHAR(30) NOT NULL,
-> city VARCHAR(30) NOT NULL,
-> zip_code VARCHAR(15) NOT NULL,
-> address VARCHAR(40) NOT NULL,
-> registration_date DATETIME NOT NULL,
-> PRIMARY KEY (user_id),
-> UNIQUE KEY (email),
-> INDEX login (email, pass)
-> );
Query OK, 0 rows affected (0.01 sec)
mysql> _

```

Εικόνα 4.48: Δημιουργία του πίνακα books

Πίνακας Καλαθιού

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
kal_id	Πρωτεύων κλειδί του πίνακα
book_id	Ξένο κλειδί του πίνακα books
user_id	Ξένο κλειδί του πίνακα users
kal_book	Τίτλος βιβλίου που προστέθηκε στο καλάθι
pr	Τιμή βιβλίου

quantity	Ποσότητα
k_url	URL κατηγορίας βιβλίου
date	Ημερομηνία προσθήκης του βιβλίου στο καλάθι

Πίνακας 4.16: Στήλες του πίνακα kalathi

The screenshot shows the phpMyAdmin interface for the 'kalathi' table. The table structure is as follows:

Πεδίο	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα
kal_id	int(3)		UNSIGNED	Όχι	Κανένα	AUTO_INCREMENT
book_id	int(4)		UNSIGNED	Όχι	Κανένα	
user_id	int(10)		UNSIGNED	Όχι	Κανένα	
kal_book	varchar(80)	greek_general_ci		Όχι	Κανένα	
pr	decimal(5,2)			Όχι	Κανένα	
quantity	int(2)			Όχι	Κανένα	
k_url	varchar(70)	utf8_general_ci		Όχι	Κανένα	
date	datetime			Όχι	Κανένα	

Below the table structure, the primary key is defined as:

Ενέργεια	Όνομα κλειδιού	Τύπος	Μοναδικό	Συμπίεσμένο	Πεδίο	Μοναδικότητα	Σύνθεση	Κενό	Σχόλιο
	PRIMARY	BTREE	Ναι	Όχι	kal_id	2	A		
	date	BTREE	Όχι	Όχι	date	2	A		

Εικόνα 4.49: Εμφάνιση στηλών στην phpMyAdmin

Πίνακας Wishlist

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
w_id	Πρωτεύων κλειδί του πίνακα
book_id	Ξένο κλειδί του πίνακα books
user_id	Ξένο κλειδί του πίνακα users
b_name	Τίτλος βιβλίου που προστέθηκε στη wishlist
img	Εικόνα βιβλίου
w_url	URL κατηγορίας βιβλίου
date	Ημερομηνία προσθήκης του βιβλίου στη wishlist

Πίνακας 4.17: Στήλες του πίνακα Wishlist

localhost ▶ bookstore ▶ wishlist

Περιήγηση Δομή Κώδικας SQL Αναζήτηση Προσθήκη Εξαγωγή Εισαγωγή Λειτουργίες

Πεδίο	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα
<input type="checkbox"/> w_id	int(10)		UNSIGNED	Όχι	Κανένα	AUTO_INCREMENT
<input type="checkbox"/> book_id	int(3)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/> user_id	int(10)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/> b_name	varchar(80)	greek_general_ci		Όχι	Κανένα	
<input type="checkbox"/> img	varchar(200)	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> w_url	varchar(70)	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> date	datetime			Όχι	Κανένα	

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εμφάνιση για εκτύπωση Εμφάνιση συσχετίσεων Προτεινόμενη δομή πίνακα

Προσθήκη 1 πεδίου(ων) Στο τέλος του Πίνακα Στην αρχή του Πίνακα Μετά το w_id Εκτέλεση

Ευρετήρια:

Ενέργεια	Όνομα κλειδιού	Τύπος	Μοναδικό	Συμπίεσμένο	Πεδίο	Μοναδικότητα	Σύνθεση	Κενό	Σχόλιο
	PRIMARY	BTREE	Ναι	Όχι	w_id	0	A		
	date	BTREE	Όχι	Όχι	date	0	A		

Εικόνα 4.50: Εμφάνιση στηλών στην phpMyAdmin

Πίνακας messages

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
msg_id	Πρωτεύων κλειδί του πίνακα
body_msg	Μήνυμα Χρήστη
date_entered	Ημερομηνία αποστολής μηνύματος
user_name	Όνομα Χρήστη
user_email	email Χρήστη

Πίνακας 4.18: Στήλες του πίνακα messages

localhost ▶ bookstore ▶ messages

Περιήγηση Δομή Κώδικας SQL Αναζήτηση Προσθήκη Εξαγωγή Εισαγωγή Λειτουργίες Αδειασμα

Πεδίο	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα
<input type="checkbox"/> msg_id	int(3)		UNSIGNED	Όχι	Κανένα	AUTO_INCREMENT
<input type="checkbox"/> body_msg	longtext	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> date_entered	timestamp		on update CURRENT_TIMESTAMP	Όχι	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP
<input type="checkbox"/> user_name	varchar(20)	utf8_general_ci		Όχι	Κανένα	
<input type="checkbox"/> user_email	varchar(40)	utf8_general_ci		Ναι	NULL	

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εμφάνιση για εκτύπωση Εμφάνιση συσχετίσεων Προτεινόμενη δομή πίνακα

Προσθήκη 1 πεδίου(ων) Στο τέλος του Πίνακα Στην αρχή του Πίνακα Μετά το msg_id Εκτέλεση

Ευρετήρια:

Ενέργεια	Όνομα κλειδιού	Τύπος	Μοναδικό	Συμπίεσμένο	Πεδίο	Μοναδικότητα	Σύνθεση	Κενό	Σχόλιο
	PRIMARY	BTREE	Ναι	Όχι	msg_id	9	A		

Εικόνα 4.51: Εμφάνιση στηλών στην phpMyAdmin

Πίνακας orders

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
order_id	Πρωτεύων κλειδί του πίνακα
user_id	Ξένο κλειδί του πίνακα users
total	Συνολικό ποσό παραγγελίας
order_date	Ημερομηνία Παραγγελίας

Πίνακας 4.19: Στήλες του πίνακα orders

localhost > bookstore > orders

Περιήγηση Δομή Κώδικας SQL Αναζήτηση Προσθήκη Εξαγωγή Εισαγωγή Λειτουργίες Αδειασμα

Πεδίο	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προκαθορισμένο	Πρόσθετα
<input type="checkbox"/> order_id	int(10)		UNSIGNED	Όχι	Κανένα	AUTO_INCREMENT
<input type="checkbox"/> user_id	int(10)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/> total	decimal(10,2)		UNSIGNED	Όχι	Κανένα	
<input type="checkbox"/> order_date	timestamp		on update CURRENT_TIMESTAMP	Όχι	CURRENT_TIMESTAMP	ON UPDATE CURRENT_TIMESTAMP

Επιλογή όλων / Απεπιλογή όλων Με τους επιλεγμένους:

Εμφάνιση για εκτύπωση Εμφάνιση συσχετίσεων Προτεινόμενη δομή πίνακα

Προσθήκη 1 πεδίου(ων) Στο τέλος του Πίνακα Στην αρχή του Πίνακα Μετά το order_id Εκτέλεση

Ευρετήρια:

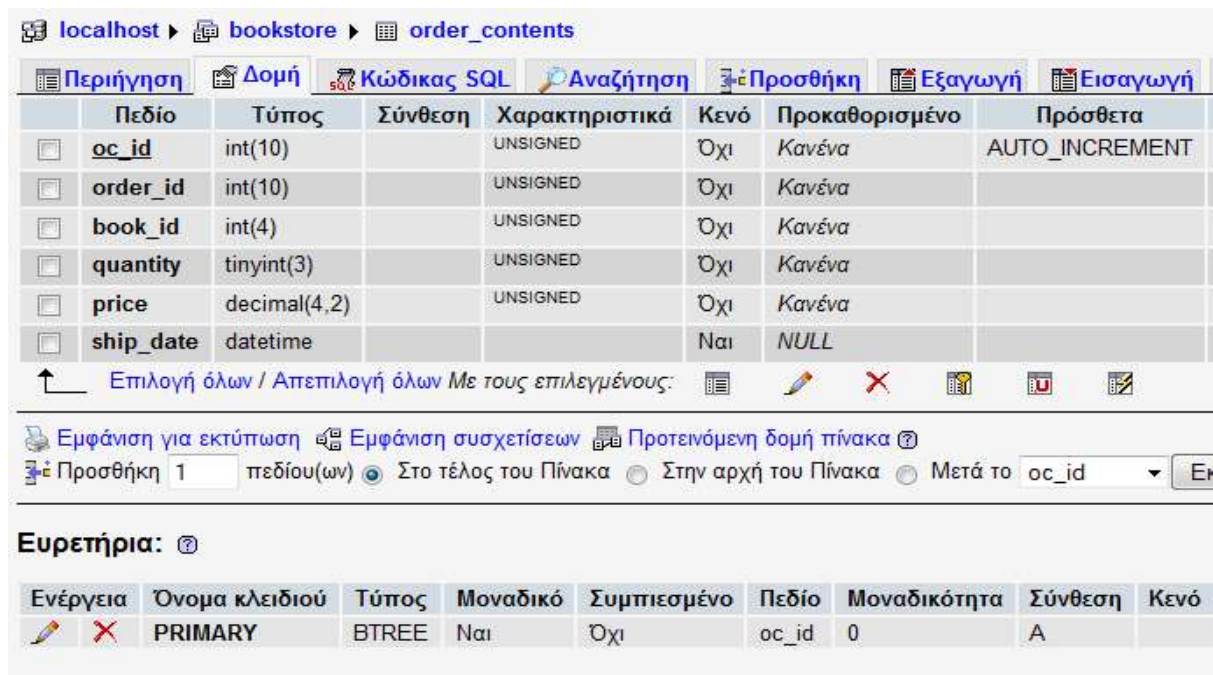
Ενέργεια	Όνομα κλειδιού	Τύπος	Μοναδικό	Συμπίεσμένο	Πεδίο	Μοναδικότητα	Σύνθεση	Κενό	Σχόλιο
<input type="checkbox"/>	PRIMARY	BTREE	Ναι	Όχι	order_id	0	A		

Εικόνα 4.52: Εμφάνιση στηλών στην phpMyAdmin

Πίνακας order_contents

ΣΤΗΛΕΣ	ΠΕΡΙΓΡΑΦΗ
oc_id	Πρωτεύων κλειδί του πίνακα
order_id	Ξένο κλειδί του πίνακα orders
book_id	Ξένο κλειδί του πίνακα books
quantity	Ποσότητα
pr	Τιμή βιβλίου
ship_date	Ημερομηνία παραγγελίας βιβλίου

Πίνακας 4.20: Στήλες του πίνακα order_contents



Εικόνα 4.53: Εμφάνιση στηλών στην phpMyAdmin

4.5 Προγραμματισμός με την PHP

4.5.1 Εγγραφή Χρήστη

Η εγγραφή του χρήστη πραγματοποιείται μέσω δύο αρχείων. Του register.php και του register2.php. Αυτό συμβαίνει για λόγους εμφάνισης. Στο πρώτο αρχείο εμφανίζεται το κατάλληλο μήνυμα που ενημερώνει τον χρήστη για την συμπλήρωση των πεδίων με αστεράκι. Στο δεύτερο αρχείο το αρχικό μήνυμα αντικαθίσταται από τα μηνύματα λαθών που μπορεί να εμφανιστούν, ή από την επιτυχή ενημέρωση της εγγραφής.

Επίσης. Ο δεύτερος λόγος χωρισμού του κώδικα σε δύο αρχεία, είναι η εμφάνιση των στοιχείων της επιτυχημένης εγγραφής στα ανάλογα πεδία της φόρμας.

Κώδικας Φόρμας (αρχείου register.php)

```
<form name="register" method="post" action="register_2.php">
<table width="100%" border="0">
<tr>
<td width="96%"><p align="justify">Προκειμένου να προχωρήσετε στην αγορά οποιουδήποτε
προϊόντος, είναι απαραίτητο να συμπληρώσετε τα στοιχεία σας στην ακόλουθη φόρμα. </p>
<p>* Η συμπλήρωση των πεδίων με αστερίσκο είναι υποχρεωτική.</p></td>
</tr>
</table>
<p><span class="red">*</span> Όνομα Χρήστη:
<input type="text" name="user_name" size="40" maxlength="20" /></p>
<p><span class="red">*</span> Όνομα :
```

```

<input type="text" name="first_name" size="40" maxlength="20" /></p>
<p><span class="red">*</span> Επώνυμο :
<input type="text" name="last_name" size="40" maxlength="40" /></p>
<p><span class="red">*</span> Διεύθυνση :
<input type="text" name="address" size="40" maxlength="40" /></p>
<p><span class="red">*</span> Πόλη :
<input type="text" name="city" size="40" maxlength="30" /></p>
<p><span class="red">*</span> T.K. :
<input type="text" name="zip_code" size="40" maxlength="10" /></p>
<p><span class="red">*</span> Τηλέφωνο :
<input type="text" name="telephone" size="40" maxlength="30" /></p>
<p><span class="red">*</span> Fax :
<input type="text" name="fax" size="40" maxlength="30" /></p>
<p><span class="red">*</span> Email :
<input type="text" name="e_mail" size="40" maxlength="30" /></p>
<p><span class="red">*</span> Κωδικός πρόσβασης :
<input type="password" name="pass1" size="40" maxlength="40" /></p>
<p><span class="red">*</span> Επιβεβαίωση κωδικού πρόσβασης :
<input type="password" name="pass2" size="40" maxlength="40" /></p>

<p align="right">&nbsp;</p>
<p align="right">
<input type="submit" name="register" value="Καταχώρηση Στοιχείων" />
</p>
<input type="hidden" name="registered" value="TRUE" />
</form>

```

Περιγραφή Φόρμας

Αυτό που πρέπει να προσέξουμε στον κώδικα της φόρμας αυτής είναι τα χαρακτηριστικά method και action. Με την τιμή post που δίνουμε η φόρμα καταχωρεί τα δεδομένα των πεδίων στο μητρώο \$_POST. Μέσω της action καθορίζουμε το αρχείο το οποίο θα παραλάβει τα δεδομένα αυτά.

Επίσης, η τιμή του χαρακτηριστικού size καθορίζει το μέγεθος του πεδίου όταν εμφανίζεται η φόρμα στο φυλλομετρητή, ενώ η τιμή του maxlength πρέπει να συμφωνεί με το μέγεθος της στήλης του πίνακα (της βάσης δεδομένων Bookstore) στον οποίο θα καταχωρηθούν τα δεδομένα που ο χρήστης θα δώσει.

Κώδικας PHP αρχείου register2.php

```
<?php
```

```

if (isset($_POST['registered']))
{
    $errors = array();
    $trimmed = array_map('trim', $_POST);
    $un = $fn = $ln = $ad = $c = $zip = $tel = $fax = $e = $p = FALSE;
    if (empty($trimmed['user_name']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Όνομά σας.';
    }else{
        if (preg_match('|^\w.{2,20}$|', $trimmed['user_name']))

```

```

        {
            $fn = mysqli_real_escape_string($dbc, $trimmed['user_name']);
        }else{
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το όνομα χρήστη.';
        }
    }
    if (empty($trimmed['first_name']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Όνομά σας.';
    }else{
        if (preg_match('/^\w.{2,20}$i', $trimmed['first_name']))
        {
            $fn = mysqli_real_escape_string($dbc, $trimmed['first_name']);
        }else{
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το μικρό σας όνομα.';
        }
    }
    if (empty($trimmed['last_name']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Επώνυμό σας.';
    }else{
        if (preg_match('/^\w.{2,40}$i', $trimmed['last_name']))
        {
            $ln = mysqli_real_escape_string($dbc, $trimmed['last_name']);
        }else{
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το επώνυμο σας.';
        }
    }
    if (empty($trimmed['address']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε τη Διεύθυνσή σας.';
    }else{
        if (preg_match('/^\w.+[ ][0-9]{2,40}$i', $trimmed['address']))
        {
            $ad = mysqli_real_escape_string($dbc, $trimmed['address']);
        }else{
            $errors[] = 'Μόνο ψηφία και χαρακτήρες επιτρέπονται για το όνομα
διεύθυνσης.';
        }
    }
    if (empty($trimmed['city']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε την Πόλη σας.';
    }else{
        if (preg_match('/^\w.{2,30}$i', $trimmed['city']))
        {
            $c = mysqli_real_escape_string($dbc, $trimmed['city']);
        }else{
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το όνομα της πόλης σας.';
        }
    }
}

```



```

if (empty($trimmed['zip_code']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Ταχυδρομικό σας Κώδικα.';
}
else{
    if (preg_match('/^[d-]{2,10}$/i', $trimmed['zip_code']))
    {
        $zip = mysqli_real_escape_string($dbc, $trimmed['zip_code']);
    }
    else{
        $errors[] = 'Μόνο ψηφία και η "-" επιτρέπονται για τον ταχυδρομικό κώδικα.';
    }
}
if (empty($trimmed['telephone']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Τηλέφωνό σας.';
}
else{
    if (preg_match('/^[d-]{2,30}$/i', $trimmed['telephone']))
    {
        $tel = mysqli_real_escape_string($dbc, $trimmed['telephone']);
    }
    else{
        $errors[] = 'Μόνο ψηφία και η "-" επιτρέπονται για τον αριθμό τηλεφώνου.';
    }
}
if (empty($trimmed['fax']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε τον αριθμό FAX.';
}
else{
    if (preg_match('/^[d-]{2,30}$/i', $trimmed['fax']))
    {
        $fax = mysqli_real_escape_string($dbc, $trimmed['fax']);
    }
    else{
        $errors[] = 'Μόνο ψηφία και η "-" επιτρέπονται για τον αριθμό Fax.';
    }
}
if (empty($trimmed['e_mail']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Email σας.';
}
else{
    if (preg_match('/^[w.-]+@[w.-]+\.[A-Za-z]{2,40}$/i', $trimmed['e_mail']))
    {
        $e = mysqli_real_escape_string($dbc, $trimmed['e_mail']);
    }
    else{
        $errors[] = 'Το email δεν είναι έγκυρο.';
    }
}
if (empty($trimmed['pass1']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Κωδικό σας.';
}
else{
    if (preg_match('/^\w{2,40}$/i', $trimmed['pass1']))
    {
        if ($trimmed['pass1'] == $trimmed['pass2'])

```

```

        {
            $p = mysqli_real_escape_string($dbc, $trimmed['pass1']);
        }else{
            $errors[] = 'Ο Κωδικός δεν επιβεβαιώθηκε σωστά.';
        }
    }else{
        $errors[] = 'Μόνο ψηφία και χαρακτήρες επιτρέπονται για το κωδικό
        πρόσβασης.';
    }
}
if (empty($errors))
{
    $q = "SELECT user_id FROM users WHERE e_mail='$e'";
    $r = mysqli_query($dbc, $q);

    if (mysqli_num_rows($r) == 0)
    {
        $q2 = "INSERT INTO users (user_name, first_name, last_name, e_mail, address, city,
        zip_code, telephone, fax, pass, registration_date) VALUES ('$un', '$fn', '$ln', '$e', '$ad', '$c', '$zip',
        '$tel', '$fax', SHA1('$p'), NOW() )";
        $r2 = mysqli_query($dbc, $q2);

        if (mysqli_affected_rows($dbc) == 1)
        {
            echo '<p>Ευχαριστούμε για την εγγραφή σας. Μπορείτε να συνδεθείτε στη
            σελίδα μας</p>';
        }else{
            echo '<p>Η σύνδεση με την Βάση Δεδομένων δεν ήταν εφικτή. Παρακαλούμε
            προσπαθήστε ξανά.</p>';
        }
        mysqli_close($dbc);
    }else{
        echo '<p>Το email που δώσατε υπάρχει ήδη.</p>';
    }
}else{
    echo '<p>Βρέθηκαν λάθη</p>';
    foreach ($errors as $msg)
    {
        echo " $msg\n<br />";
    }
}
?>

```

Περιγραφή κώδικα

Ο παραπάνω κώδικας θέλουμε να εκτελείται όταν ο χρήστης πατήσει “Καταχώρηση Στοιχείων”. Αυτό το επιτυγχάνουμε με τη συνθήκη:

```
if (isset($_POST['registered']))
```

Στην δημιουργία της φόρμας δώσαμε στο κουμπί με όνομα “Καταχώρηση Στοιχείων” την τιμή registered.

```
<input type="submit" name="register" value="Καταχώρηση Στοιχείων" />
<input type="hidden" name="registered" value="TRUE" />
```

Έτσι η PHP καταλαβαίνει ποιο κουμπί έχει πατηθεί από τον χρήστη και τρέχει τον ανάλογο κώδικα.

Στη συνέχεια δηλώνουμε τον πίνακα που καταγράφονται τα σφάλματα:

```
$errors = array();
```

Μέσα στο μητρώο \$_POST καταχωρούνται τα δεδομένα που πληκτρολογεί ο χρήστης στα πεδία της φόρμας. Από τα δεδομένα αυτά πρέπει να περικοπούν όλα τα κενά διαστήματα. Αυτό γίνεται με την συνάρτηση trim().

```
$trimmed = array_map('trim',$_POST);
```

Η array_map δέχεται σαν 1ο όρισμα μια συνάρτηση και σαν 2ο ένα μητρώο. Σε κάθε μητρώο \$_POST εφαρμόζουμε την συνάρτηση trim() και αναθέτουμε την τιμή σε ένα καινούργιο μητρώο \$trimmed.

Φυσικά τίποτα δεν είναι σίγουρο, γι’ αυτό και πρέπει να δώσουμε αρχικές τιμές στις μεταβλητές που θα χρησιμοποιήσουμε.

```
$un = $fn = $ln = $ad = $c = $zip = $tel = $fax = $e = $p =FALSE;
```

Για να ελέγξουμε αν ο χρήστης όντως συμπλήρωσε το πεδίο user_name – Όνομα Χρήστη εκτελούμε την συνθήκη:

```
if (empty($trimmed['user_name']))
```

Με αυτό τον τρόπο ελέγχουμε αν είναι άδειο το μητρώο \$trimmed['user_name']. Αν ναι τότε στον πίνακα errors[] προσθέτουμε το ανάλογο σφάλμα. Αν όχι, δηλαδή αν ο χρήστης έχει δώσει τιμή, ελέγχουμε την τιμή αυτή μέσω των παραστάσεων Perl:

```
if (preg_match('|^\w{2,20}$|', $trimmed['user_name']))
```

Τα δεδομένα που υποβάλουν οι χρήστες δεν είναι πάντα αξιόπιστα. Ένας σκοπός των επιθέσεων με “ενέσεις” SQL είναι η δημιουργία εσφαλμένων συντακτικά ερωτημάτων, που θα αποκαλύψουν πληροφορίες για το σενάριο ή τη ΒΔ στο μήνυμα σφάλματος που θα εμφανιστεί. Η mysql_real_escape_string() φιλτράρει(“καθαρίζει”) τα δεδομένα για λόγους ασφαλείας τοποθετώντας το χαρακτήρα διαφυγής πριν από τους πιθανόν προβληματικούς χαρακτήρες:

```
$fn = mysql_real_escape_string($dbc, $trimmed['user_name']);
```

Οι συγκεκριμένοι έλεγχοι πραγματοποιούνται για όλα τα πεδία της φόρμας εγγραφής.

Τέλος, για να σιγουρευτούμε ότι δεν υπήρξε κανένα σφάλμα κατά την καταχώρηση των στοιχείων από το χρήστη εκτελούμε την συνθήκη:

```
if (empty($errors))
```

Αν δεν υπάρχουν λάθη είναι πολύ βασικό να ελέγξουμε αν το email που έδωσε ο χρήστης υπάρχει ήδη, γιατί πρέπει να είναι μοναδικό για κάθε χρήστη. Γι’ αυτό και εκτελούμε το ερώτημα της επιλογής (SELECT) με το οποίο ανακτάμε από την βάση δεδομένων όποια εγγραφή έχει διεύθυνση ηλεκτρονικού ταχυδρομείου ίδια με αυτή που πληκτρολόγησε ο χρήστης.

```
$q = "SELECT user_id FROM users WHERE e_mail='$e'";
```

Η mysql_query() είναι η συνάρτηση που εκτελεί τα ερωτήματα. Δέχεται ως 1ο όρισμα τη σύνδεση με τη βάση δεδομένων και ως 2ο όρισμα το ερώτημα. Η μεταβλητή \$r (=result) θα έχει τιμή TRUE ή FALSE ανάλογα αν το ερώτημα εκτελέστηκε σωστά ή όχι:

```
$r = mysql_query($dbc, $q);
```

Η συνάρτηση mysql_num_rows(\$r) επιστρέφει το πλήθος των γραμμών που έχει επιστρέψει το ερώτημα επιλογής SELECT. Αν η συνάρτηση mysql_num_rows(\$r) = 0 γνωρίζουμε ότι το email

δεν έχει ήδη καταχωρηθεί στη βάση δεδομένων και επομένως μπορούμε να εκτελέσουμε με ασφάλεια το ερώτημα της εισαγωγής (INSERT).

```
if (mysqli_num_rows($r) == 0)
{
    $q2 = "INSERT INTO users (user_name, first_name, last_name, e_mail,
address, city, zip_code, telephone, fax, pass, registration_date) VALUES ('$un', '$fn', '$ln',
'$e', '$ad', '$c', '$zip', '$tel', '$fax', SHA1('$p'), NOW() )";
    $r2 = mysqli_query($dbc, $q2);
```

Με την εντολή INSERT INTO μπορούμε να εισάγουμε δεδομένα σε όποιον πίνακα επιθυμούμε. Με όποια σειρά γράψουμε τις στήλες του πίνακα με την ίδια σειρά πρέπει να γράψουμε και τις μεταβλητές από τις οποίες θα πάρουμε τις τιμές και θα τις εκχωρήσουμε στις ανάλογες στήλες.

Αν το ερώτημα λειτούργησε σωστά η συνάρτηση mysqli_affected_rows() επιστρέφει το πλήθος των γραμμών που έχουν επηρεαστεί από ένα ερώτημα εισαγωγής.

```
if (mysqli_affected_rows($dbc) == 1)
    Έπειτα διακόπτουμε την σύνδεση με την βάση δεδομένων:
    mysqli_close($dbc);
    Όλα τα σφάλματα εμφανίζονται με τη βοήθεια ενός βρόχου foreach:
    foreach ($errors as $msg) { echo " $msg\n<br />"; }
```

Κώδικας Φόρμας (αρχείου register2.php)

```
<form name="register" method="post" action="register_2.php">
<p><span class="red">*</span> Όνομα Χρήστη:
    <input type="text" name="user_name" size="40" maxlength="20" value="<?php echo $un; ?>" /></p>
<p><span class="red">*</span> Όνομα :
    <input type="text" name="first_name" size="40" maxlength="20" value="<?php echo $fn; ?>" /></p>
<p><span class="red">*</span> Επώνυμο :
    <input type="text" name="last_name" size="40" maxlength="40" value="<?php echo $ln; ?>" /></p>
<p><span class="red">*</span> Διεύθυνση :
    <input type="text" name="address" size="40" maxlength="40" value="<?php echo $ad; ?>" /></p>
<p><span class="red">*</span> Πόλη :
    <input type="text" name="city" size="40" maxlength="30" value="<?php echo $c; ?>" /></p>
<p><span class="red">*</span> T.K. :
    <input type="text" name="zip_code" size="40" maxlength="10" value="<?php echo $zip ?>" /></p>
<p><span class="red">*</span> Τηλέφωνο :
    <input type="text" name="telephone" size="40" maxlength="30" value="<?php echo $tel; ?>" /></p>
<p><span class="red">*</span> Fax :
    <input type="text" name="fax" size="40" maxlength="30" value="<?php echo $fax; ?>" /></p>
<p><span class="red">*</span> Email :
    <input type="text" name="e_mail" size="40" maxlength="30" value="<?php echo $e; ?>" /></p>
<p><span class="red">*</span> Κωδικός πρόσβασης :
    <input type="password" name="pass1" size="40" maxlength="40" /></p>
<p><span class="red">*</span> Επιβεβαίωση κωδικού πρόσβασης :
    <input type="password" name="pass2" size="40" maxlength="40" /></p>
<p>&nbsp;</p>
<p align="right"><input type="submit" name="register" value="Καταχώρηση Στοιχείων" /></p>
<input type="hidden" name="registered" value="TRUE" />
</form>
```

Εικόνα 4.54: Κώδικας HTML για την Φόρμα Εγγραφής

Περιγραφή Φόρμας

Οι δύο αυτές φόρμες είναι σχεδόν ίδιες, αλλά έχουν δύο βασικές διαφορές. Η δεύτερη φόρμα είναι μια παραμένουσα φόρμα, δηλαδή εκτελείται πάντα στο ίδιο αρχείο (action = register2.php), και τα στοιχεία που καταχωρεί ο χρήστης εμφανίζονται στα ανάλογα πεδία της.



* Όνομα Χρήστη:

* Όνομα :

* Επώνυμο :

* Διεύθυνση :

* Πόλη :

* Τ.Κ. :

* Τηλέφωνο :

* Fax :

* Email :

* Κωδικός πρόσβασης :

* Επιβεβαίωση κωδικού πρόσβασης :

Εικόνα 4.55: Φόρμα Εγγραφής

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #text_register2 { margin-right: 30px; margin-left: 30px; color: #000000; font-family: "Times New Roman", Times, serif; background-repeat: repeat; background-image: url(background%20img.jpg); font-size: 16px; padding-top: 80px; width: 946px; }</pre>	<pre>.bookstore #text_register2 form{ text-align: right; width: 560px; font-weight: bold; padding-bottom: 40px; margin-left: 290px; margin-right: 30px; padding-top: 20px; }</pre>

Πίνακας 4.21: Φύλλα CSS για την μορφοποίηση της φόρμας εγγραφής

4.5.2 Είσοδο – Έξοδος από τον Λογαριασμό του Χρήστη

Κωδικός PHP (Εισόδου)

```
<?php
```

```
require_once ('C:/xampp/config.inc.php');
require_once ('C:/xampp/connect_to_mysql.php');
$row = FALSE ;
$row2 = FALSE;
if (isset($_POST['entered_login']))
{
    $trimmed = array_map('trim',$_POST);
    $e = $p = FALSE;
    if (empty($trimmed['e_mail']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Email σας.';
    }else{
        if (preg_match('/^[w.-]+@[w.-]+\.[A-Za-z]{2,40}$/i', $trimmed['e_mail']))
        {
            $e = mysqli_real_escape_string($dbc, $trimmed['e_mail']);
        }
    }
    if (empty($trimmed['pass']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Κωδικό σας.';
    }else{
        if (preg_match('/^\w{2,30}$/i', $trimmed['pass']))
        {
            $p = mysqli_real_escape_string($dbc, $trimmed['pass']);
        }
    }
    if (empty($errors))
    {
        $q = "SELECT user_id, user_name, first_name, last_name, e_mail, pass, address, city,
telephone, fax, zip_code FROM users WHERE (e_mail='$e' AND pass=SHA1('$p'))";
        $r = mysqli_query($dbc, $q);

        if (mysqli_num_rows($r) == 1)
        {
            $_SESSION = mysqli_fetch_array ($r, MYSQLI_ASSOC);
            $_SESSION['agent'] = md5($_SERVER['HTTP_USER_AGENT']);
            if ($_SESSION['e_mail'] == EMAIL)
            {
                mysqli_free_result($r);
                mysqli_close($dbc);
                $url = BASE_URL . 'admin.php' ;
                ob_end_clean();
                header("Location: $url");
                exit();
            }
        }
    }
}
```


Στο αρχείο config.inc.php ορίζουμε τέσσερις σταθερές, τις LIVE, EMAIL, BASE_URL και MYSQL. Επίσης, δημιουργούμε μια δική μας συνάρτηση χειρισμών σφαλμάτων την my_error_handler η οποία μας εμφανίζει το κατάλληλο μήνυμα σφάλματος.

Η σταθερά LIVE είναι μια σημαία που μας δείχνει αν η τοποθεσία είναι σε λειτουργία ή όχι. Η διαφορά είναι σημαντική, επειδή αλλιώς χειριζόμαστε τα σφάλματα όταν η τοποθεσία είναι σε λειτουργία και αλλιώς όταν βρίσκεται στην φάση της ανάπτυξης. Όταν η τοποθεσία δεν είναι σε λειτουργία εμφανίζεται ένα λεπτομερές μήνυμα σφάλματος, ενώ αν είναι σε λειτουργία θα τυπωθεί ένα απλό απολογητικό μήνυμα που θα ενημερώνει το χρήστη ότι προέκυψε κάποιο σφάλμα.

Τέλος, με την εντολή set_error_handler('my_error_handler') λέμε στην PHP να χρησιμοποιήσει την δική μας συνάρτηση χειρισμού σφαλμάτων αντί για την προεπιλεγμένη.

```
<?php
ξdbc = @mysqli_connect ('localhost', 'epp1545', 'mybookstore', 'bookstore')
OR die ('Could not connect to MYSQL : ' . mysqli_connect_error() );
?>
```

Εικόνα 4.57: Σενάριο PHP για την σύνδεση με τη MySQL

Το αρχείο connect_to_mysql.php εγκαθιστά μια σύνδεση με την MySQL και επιλέγει τη βάση δεδομένων. Αν η συνάρτηση mysqli_connect() δεν μπορεί να επιστρέψει έναν έγκυρο σύνδεσμο πόρου εκτελείται το τμήμα OR die() της εντολής. Η συνάρτηση die() δεχόμενη μια συμβολοσειρά που θα εμφανιστεί στον φυλλομετρητή τερματίζει την εκτέλεση του σεναρίου. Στο συγκεκριμένο αρχείο η συμβολοσειρά αποτελείται από το κείμενο “Could not connect to MySQL:” και το σχετικό μήνυμα σφάλματος της MySQL.

- localhost = υπολογιστής υπηρεσίας
- epp1545 = όνομα χρήστη
- mybookstore = κωδικός πρόσβασης
- bookstore = βάση δεδομένων

Αν ο χρήστης πατήσει το κουμπί “Είσοδος μέλους” θα εκτελεστεί το σενάριο της PHP:

```
if (isset($_POST['entered_login']))
```

Τα κενά διαστήματα από τα δεδομένα που εκχωρήθηκαν θα περικοπούν:

```
$trimmed = array_map('trim',$_POST);
```

Αν ο χρήστης δεν έδωσε τιμή για το email του: **if (empty(\$trimmed['e_mail']))**

Προστίθεται το κατάλληλο μήνυμα σφάλματος στον πίνακα errors[]:

```
$errors[] = 'Παρακαλούμε συμπληρώστε το Email σας.');
```

Αν ο χρήστης έχει δώσει τιμή, ελέγχουμε αν η τιμή αυτή ταιριάζει με το κατάλληλο μοτίβο των παρατάσεων Perl:

```
if (preg_match('/^[\\w.-]+@[\\w.]+.[A-Za-z]{2,6}$/i', $trimmed['e_mail']))
```

Το μοτίβο για το email του χρήστη είναι της μορφής: '/^[\\w.-]+@[\\w.]+.[A-Za-z]{2,40}\$/i'
Οι διευθύνσεις ηλεκτρονικού ταχυδρομείου αρχίζουν από γράμματα, αριθμούς, και το χαρακτήρα υπογράμμισης (που αντιπροσωπεύονται από την κλάση \\w), συν μία τελεία και μια παύλα. Στη

συνέχεια ακολουθεί ένας μόνο χαρακτήρας @. Μετά από αυτό μπορεί να ακολουθεί οποιοδήποτε πλήθος από γράμματα, αριθμούς, τελείες και παύλες. Τέλος, όλες οι διευθύνσεις ολοκληρώνονται με μια τελεία και από δύο έως έξι γράμματα.

Αν υπάρχει ταύτιση του μοτίβου και τις τιμές καλούμε την συνάρτηση `mysqli_real_escape_string` η οποία φιλτράρει τα δεδομένα. Το αποτέλεσμα εκχωρείται στη μεταβλητή `$e`:

```
$e = mysqli_real_escape_string($dbc, $trimmed['e_mail']);
```

Την ίδια διαδικασία εκτελούμε και για τον κωδικό του χρήστη. Έπειτα αν ο πίνακας των σφαλμάτων είναι άδειος - `if (empty($errors))` - εκτελούμε το ερώτημα SQL:

```
$q = "SELECT user_id, user_name, first_name, last_name, e_mail, pass, address, city, telephone, fax, zip_code FROM users WHERE (e_mail='$e' AND pass=SHA1('$p'))";
```

```
$r = mysqli_query($dbc, $q);
```

Με το ερώτημα αυτό της SQL επιλέγουμε τα στοιχεία του χρήστη-μέλους που αντιστοιχούν στο email και τον κωδικό πρόσβασης που καταχωρήθηκαν στα πεδία της φόρμας.

Αν τελικά το ερώτημα έχει επιστρέψει μια τιμή - `if (mysqli_num_rows($r) == 1)` - τότε στο μητρώο `$_SESSION` καταχωρούνται τα στοιχεία που το ερώτημα SELECT έχει επιστρέψει, δηλαδή τα στοιχεία του χρήστη:

```
$_SESSION = mysqli_fetch_array($r, MYSQLI_ASSOC);
```

Η `mysqli_fetch_array()` δέχεται ως όρισμα τα αποτελέσματα του ερωτήματος και επιστρέφει μια γραμμή δεδομένων τη φορά σε μορφή πίνακα που η μορφή του καθορίζεται από την `MYSQLI_ASSOC`.

ΣΤΑΘΕΡΑ	ΠΑΡΑΔΕΙΓΜΑ
<code>MYSQLI_ASSOC</code>	<code>\$row['στήλη'];</code>
<code>MYSQLI_NUM</code>	<code>\$row['0'];</code>
<code>MYSQLI_BOTH</code>	<code>\$row['0'];</code> ή <code>\$row['στήλη'];</code>

Πίνακας 4.22: Σταθερές της `mysqli_fetch_array`

Επειδή μέσα στις περιόδους εργασίας αποθηκεύονται σημαντικά δεδομένα, είναι πολύ πιθανό ένας κακόβουλος χρήστης να επιχειρήσει να υποκλέψει τα δεδομένα μέσω του αναγνωριστικού της περιόδου εργασίας. Αν μάθει το αναγνωριστικό μπορεί εύκολα να ξεγελάσει τον διακομιστή και να το θεωρεί ως δικό του. Με την τιμή `HTTP_USER_AGENT` για να υποκλέψει κάποιος την περίοδο εργασίας πρέπει να χρησιμοποιεί ακριβώς τον ίδιο φυλλομετρητή και λειτουργικό σύστημα. Έτσι αποθηκεύουμε μια κρυπτογραφημένη μορφή της τιμής `HTTP_USER_AGENT` σε μία περίοδο εργασίας:

```
$_SESSION['agent'] = md5($_SERVER['HTTP_USER_AGENT']);
```

Απελευθέρωση πόρων τους οποίους δεσμεύουν τα αποτελέσματα και αποσύνδεση από την βάση δεδομένων:

```
mysqli_free_result($r);
```

```
mysqli_close($dbc);
```

Στη συνέχεια ελέγχουμε αν το email του χρήστη είναι ίδιο με εκείνο του administrator:

```
if ($_SESSION['e_mail'] == EMAIL)
```

Αν τελικά είναι ίδιο γίνεται η ανακατεύθυνση του χρήστη στη σελίδα `admin.php` και διαγραφή των δεδομένων της προσωρινής μνήμης:

```
$url = BASE_URL . 'admin.php' ;  
ob_end_clean();  
header("Location: $url");
```

Στο τέλος καλούμε την συνάρτηση exit() για να τερματίσουμε την εκτέλεση του σεναρίου.

```
exit();
```

Αν όμως ο χρήστης είναι ένα απλό μέλος της ιστοσελίδας και όχι ο διαχειριστής εκτελούμε την ίδια διαδικασία με την διαφορά ότι ανακατευθύνουμε τον χρήστη στη σελίδα loggedin.php. Φυσικά αν το σενάριο δεν εκτελέστηκε σωστά εκτυπώνονται τα ανάλογα σφάλματα.

Κωδικός PHP (Εξόδου)

```
<?php
```

```
require_once ('c:/xampp/config.inc.php');  
require_once ('C:/xampp/connect_to_mysql.php');
```

```
if (!isset($_SESSION['agent']) OR ($_SESSION['agent'] != md5($_SERVER['HTTP_USER_AGENT'])))  
{
```

```
    $url = BASE_URL . 'false_login.php';  
    ob_end_clean();  
    header("Location : $url");  
    exit();
```

```
}else{
```

```
    echo "<p>{$_SESSION['user_name']}, έχεις συνδεθεί στη σελίδα μας.</p>";
```

```
}
```

```
if (isset($_POST['entered_logout']))
```

```
{
```

```
    $_SESSION = array();  
    session_destroy();  
    setcookie (session_name(), ' ', time()-3600, '/', ' ', 0, 0);  
    $url = BASE_URL . 'logout.php';  
    ob_end_clean();  
    header("Location : $url");  
    exit();
```

```
}
```

```
?>
```

Περιγραφή Κώδικα

Πρώτα από όλα πρέπει να ελέγξουμε αν έχει οριστεί η μεταβλητή \$_SESSION['agent'] και αν η κρυπτογραφημένη τιμή της μεταβλητής \$_SERVER['HTTP_USER_AGENT'] διαφέρει από την τιμή της αποθηκευμένης μεταβλητής \$_SESSION['agent']. Αν κάποια από αυτές τις συνθήκες αληθεύει, ο χρήστης θα ανακατευθυνθεί στην σελίδα false_login.php. Αλλιώς θα εμφανιστεί ένα μήνυμα ενημέρωσης:

```
echo "<p>{$_SESSION['user_name']}, έχεις συνδεθεί στη σελίδα μας.</p>";
```

\$_SESSION['user_name'] είναι το user_name του χρήστη που είναι αποθηκευμένο μέσα στο μητρώο \$_SESSION.

Αν ο χρήστης πατήσει το κουμπί “Εξοδος μέλους” θα εκτελεστεί το σενάριο της PHP:

```
if (isset($_POST['entered_logout']))
```

Γίνεται η διαγραφή όλων των μεταβλητών της περιόδου εργασίας: `$_SESSION = array();`

Έπειτα διαγράφονται όλα τα δεδομένα της περιόδου εργασίας από τον διακομιστή: `session_destroy();`

Στέλνει ένα μπισκότο που έχει ήδη λήξει για να αντικαταστήσει το υπάρχων μπισκότο στο φυλλομετρητή. `setcookie (session_name(), ' ', time()-3600, '/', ' ', 0, 0);`

Τέλος γίνεται η ανακατεύθυνση του χρήστη στη σελίδα `logout.php`, η διαγραφή των δεδομένων από την προσωρινή μνήμη και ο τερματισμός εκτέλεσης του σεναρίου:

```
$url = BASE_URL . 'logout.php';
```

```
ob_end_clean();
```

```
header("Location : $url");
```

4.5.3 Επικοινωνία με τον διαχειριστή

Κώδικας PHP

```
<?php
```

```
if (isset($_POST['contacted']))
```

```
{
```

```
    function spam_scrubber ($value)
```

```
    {
```

```
        $bad = array('to:', 'cc:', 'bcc:', 'contact-type:', 'mine-version:', 'multipart-mixed:',  
                    'content-transfer-encoding:');
```

```
        foreach ($bad as $v)
```

```
        {
```

```
            if (stripos($value, $v) !== FALSE) return ' ';
```

```
        }
```

```
        $value = str_replace (array("\r", "\n", "%a", "%0d"), ' ', $value);
```

```
        trim($value);
```

```
    }
```

```
    $scrubbed = array_map('spam_scrubber', $_POST);
```

```
    if (!empty($scrubbed['first_name']) && !empty($scrubbed['last_name']) &&
```

```
        !empty($scrubbed['e-mail']) && !empty($scrubbed['comments']))
```

```
    {
```

```
        $body = "Όνομα : {$scrubbed['first_name']}\nΕπώνυμο :
```

```
        {$scrubbed['last_name']}\n\nΣχόλια : {$scrubbed['comments']}";
```

```
        $body = wordwrap($body, 60);
```

```
        mail('admin@localhost.com', 'Μήνυμα Χρήστη', $body, "From: {$scrubbed['e-mail']}");
```

```
        echo '<p>Ευχαριστούμε που επικοινωνήσατε μαζί μας.</p>';
```

```
        $_POST = array();
```

```
    }else{
```

```
        echo '<p>Παρακαλούμε συμπληρώστε σωστά τα στοιχεία σας.</p>';
```

```
    }
```

```
}
```

```
?>
```

Περιγραφή κώδικα

Αν ο χρήστης πατήσει το κουμπί “Αποστολή” θα εκτελεστεί το σενάριο της PHP:

```
if (isset($_POST['contacted']))
```

Οι προβληματικοί χαρακτήρες του μηνύματος θα “συλληφθούν” από τη συνάρτηση `spam_scrubber()` η οποία δέχεται ένα όρισμα, την μεταβλητή `$value`: **function spam_scrubber (\$value)**

Έπειτα δημιουργείται ένας κατάλογος με ανεπιθύμητες συμβολοσειρές που αν υπάρχουν στα δεδομένα που υποβάλει μια φόρμα είναι ένδειξη προσπάθειας αποστολής ανεπιθύμητης αλληλογραφίας:

```
$bad = array('to:', 'cc:', 'bcc:', 'contact-type:', 'mine-version:', 'multipart-mixed:',  
'content-transfer-encoding:');
```

Για να διευκολυνθεί ο έλεγχος, οι συμβολοσειρές αποθηκεύονται σε ένα μητρώο που διατρέχει ένας βρόχος: **foreach (\$bad as \$v)**

Διατρέχοντας τον πίνακα με έναν βρόχο η συνάρτηση `stripos()` ελέγχει αν τα ανεπιθύμητα στοιχεία περιέχονται στη συμβολοσειρά `$value`. Αν ναι επιστρέφει μια κενή συμβολοσειρά και τερματίζει: **if (stripos(\$value, \$v) !== FALSE) return ' '**;

Στη συνέχεια γίνεται αντικατάσταση των χαρακτήρων αλλαγής γραμμής με κενά διαστήματα. Η συνάρτηση `str_replace()` ελέγχει την τιμή του 3ου ορίσματος (`$value`) και αντικαθιστά όλες τις παρουσίες των χαρακτήρων του 1ου ορίσματος με το χαρακτήρα ή τους χαρακτήρες του 2ου ορίσματος: **\$value = str_replace(array("\r", "\n", "%a", "%0d"), ' ', \$value);**

Με την συνάρτηση `trim()` περικόπτουμε τα κενά διαστήματα: **trim(\$value);**

Καλούμε την συνάρτηση `spam_scrubber()` μέσω της `array_map()` η οποία δέχεται σαν 1ο όρισμα την συνάρτηση και σαν 2ο ένα μητρώο. Έτσι η `spam_scrubber` καλείται μια φορά για κάθε στοιχείο του μητρώου και το αποτέλεσμα αναθέτεται σε μια νέα μεταβλητή. Στο σενάριο αυτό το μητρώο `$_POST` περιέχει έξι στοιχεία: `first_name`, `last_name`, `e-mail`, `comments`, `contact` και `contacted`. Έτσι και το μητρώο `$scrubbed` θα περιέχει έξι στοιχεία: `$scrubbed['first_name']`, `$scrubbed['last_name']`, `$scrubbed['e-mail']`, `$scrubbed['comments']`, `$scrubbed['contact']`, `$scrubbed['contacted']`.

```
$scrubbed = array_map('spam_scrubber', $_POST);
```

Ελέγχουμε αν τα τέσσερα αυτά μητρώα έχουν τιμή: **if (!empty(\$scrubbed['first_name']) && !empty(\$scrubbed['last_name']) && !empty(\$scrubbed['e-mail']) && !empty(\$scrubbed['comments']))**

Δημιουργούμε το σώμα του μηνύματος: **\$body = "Όνομα : {\$scrubbed['first_name']}\nΕπώνυμο : {\$scrubbed['last_name']}\n\nΣχόλια : {\$scrubbed['comments']}";**

Με την συνάρτηση `wordwrap` προσθέτουμε έναν χαρακτήρα αλλαγής γραμμής κάθε 60 χαρακτήρες:

```
$body = wordwrap($body, 60);
```

Με τη συνάρτηση mail() γίνεται η αποστολή του μηνύματος - mail(προς, θέμα, σώμα, [κεφαλίδες]).
mail('admin@localhost.com', 'Μήνυμα Χρήστη', \$body, "From: {\$scrubbed['e-mail']}");

Επειδή η φόρμα είναι παραμένουσα οι τιμές της φόρμας αν η αποστολή του μηνύματος είναι επιτυχής δεν χρειάζεται να εμφανίζονται ξανά τα δεδομένα. Για να το αποφύγουμε καθαρίζουμε το μητρώο \$_POST:
\$_POST = array();

4.5.4 Δημιουργία Forum

Κώδικας PHP

```
<?php
```

```
$q2 = "SELECT user_name, user_email, body_msg, date_entered FROM messages LIMIT 9";  
$r2 = mysqli_query($dbc, $q2);
```

```
if (mysqli_num_rows($r2) >= 1)
```

```
{  
    while ($row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC))  
    {  
        echo "<p>{$row2['body_msg']}</p>";  
        echo "\n";  
        echo "<p>{$row2['date_entered']} - {$row2['user_name']}</p>";  
        echo "-----";  
        echo "\n";  
    }  
}
```

```
}else{  
    echo "<p>Δεν υπάρχουν μηνύματα.</p>";  
}
```

```
if (isset($_POST['sent']))
```

```
{  
    $errors = array();  
    $trimmed = array_map('trim', $_POST);  
    $un = $ue = $bmsg = FALSE;  
    if (empty($trimmed['user_name']))  
    {  
        $errors[] = 'Δεν συμπληρώσατε τον όνομά σας.';  
    }else{  
        if (preg_match("/^\w.{2,20}$i", $trimmed['user_name']))  
        {  
            $un = mysqli_real_escape_string($dbc, $trimmed['user_name']);  
        }else{  
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το όνομά σας.';  
        }  
    }  
}
```

```

}

if (empty($trimmed['body_msg']))
{
    $errors[] = 'Δεν δώσατε τα σχόλιά σας.';
}
else{
    if (preg_match('/^\w.{2,40}$i', $trimmed['body_msg']))
    {
        $bmsg = mysqli_real_escape_string($dbc, $trimmed['body_msg']);
    }
    else{
        $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για τα σχόλιά σας.';
    }
}
}
if (!empty($trimmed['user_email']))
{
    if (preg_match('/^[w.-]+@[w.]+\.[A-Za-z]{2,40}$i', $trimmed['user_email']))
    {
        $sue = mysqli_real_escape_string($dbc, $trimmed['user_email']);
    }
    else{
        $errors[] = 'Το email δεν είναι έγκυρο.';
    }
}
}
if (empty($errors))
{
    $q = "INSERT INTO messages (user_name, user_email, body_msg, date_entered)
VALUES ('$un', '$sue', '$bmsg', NOW() )";
    $r = mysqli_query($dbc, $q);
    if ($r)
    {
        echo '<p>Η αποστολή ολοκληρώθηκε με επιτυχία. Το σχόλιό σας δεν θα
εμφανιστεί μέχρι να εγκριθεί από τους διαχειριστές.</p>';
    }
    else{
        $errors[] = 'Η σύνδεση με την Βάση Δεδομένων δεν ήταν εφικτή. Παρακαλούμε
προσπαθήστε ξανά.';
    }
    mysqli_close($dbc);
}
else{
    echo '<p>Βρέθηκαν λάθη</p>';
    foreach ($errors as $msg)
    {
        echo " $msg\n<br />";
    }
}
}
?>

```


Περιγραφή Κώδικα

Στη αρχή εκτελείται ένα ερώτημα SELECT με το οποίο επιλέγουμε τις καταχωρήσεις που υπάρχουν στον πίνακα messages. Το χαρακτηριστικό LIMIT δείχνει πόσες εγγραφές θα εμφανιστούν. Αυτό γίνεται για να μην εμφανίζεται αμέσως το μήνυμα που ο κάθε χρήστης θα στέλνει, αλλά όταν και αν το αποφασίσει ο διαχειριστής.

Έπειτα μέσω ενός βρόχου (while) εμφανίζουμε τα μηνύματα που ανακτήσαμε από τον πίνακα μέσω του ερωτήματος.

Αν ο χρήστης πατήσει το κουμπί “Αποστολή” θα εκτελεστεί το σενάριο της PHP:

```
if (isset($_POST['sent']))
```

Όπως σε άλλα σενάρια έτσι και εδώ ελέγχουμε αν ο χρήστης έχει συμπληρώσει όλα τα πεδία και αν οι τιμές συμφωνούν με τα ανάλογα μοτίβα.

Μόνο αν δεν υπάρχει κανένα λάθος εκτελείται το ερώτημα εισαγωγής που προσθέτει τα καινούρια δεδομένα στον πίνακα messages.

Στο τέλος αν η καταχώρηση των στοιχείων είναι επιτυχής εμφανίζεται ένα μήνυμα ενημέρωσης, αλλιώς τυπώνονται τα σφάλματα που προέκυψαν.

Ο κώδικας για την περίπτωση του administrator είναι διαφορετικός, γιατί δεν υπάρχει φόρμα αποστολής μηνύματος. Εμφανίζονται μόνο τα δεδομένα που οι χρήστες έχουν στείλει, αλλά χωρίς περιορισμό.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #forum { color: #000000; font-size: 16px; font-weight: bolder; margin-right: 30px; margin-left: 30px; padding-top: 50px; background-image: url(background%20img.jpg); background-repeat: repeat-y; }</pre>	<pre>.bookstore #forum form { text-align: right; }</pre>

Πίνακας 4.23: Φύλλα CSS για την μορφοποίηση του Forum

Κώδικας PHP (forum_admin.php)

```
<?php
```

```
$q2 = "SELECT user_name, user_email, body_msg, date_entered FROM messages";
```

```
$r2 = mysqli_query($dbc, $q2);
```

```
if (mysqli_num_rows($r2) >= 1)
```

```
{
```

```
    while ($row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
```

```
    {
```

```
        echo "<p>{$row2['body_msg']}</p>";
```

```

        echo "\n";
        echo "<p>{$row2['date_entered']} - {$row2['user_name']}</p>";
        echo "-----";
        echo "\n";
    }
}else{
    echo "<p>Δεν υπάρχουν μηνύματα.</p>";
}
?>

```

4.5.5 Εμφάνιση Χρηστών

Είναι πολύ σημαντικό για τον διαχειριστή μιας ιστοσελίδας να γνωρίζει ποιοι χρήστες είναι εγγεγραμμένοι σε αυτή. Για το σκοπό αυτό δημιουργήθηκε μία σελίδα (view_users.php) όπου εκτυπώνονται τα στοιχεία των χρηστών και στη οποία έχει πρόσβαση μόνο ο διαχειριστής

Κώδικας PHP

```

<?php

echo '
<tr>
<td height="55" class="style3"><div align="right" class="style6 style11">Πίνακας Χρηστών</div></td>
</tr>
<tr>
<td><table width="100%" border="0">
<tr>
<td width="21%"><div align="left">Όνομα Χρήστη</div></td>
<td width="17%"><div align="center">Email</div></td>
<td width="15%"><div align="center">Πόλη</div></td>
<td width="14%"><div align="center">Διεύθυνση</div></td>
<td width="12%"><div align="center">Τ.Κ.</div></td>
<td width="13%"><div align="center">Ημερομηνία Εγγραφής</div></td>
<td width="8%"><div align="center">Αφαίρεση</div></td>
</tr>
</table></td>
</tr>
<tr>
<td height="21">-----
-----</td>
</tr>';

$uid = $un = $un2 = $e = $c = $ad = $tel = $f = $zip = $date = FALSE;

$q = "SELECT user_id, CONCAT(first_name, ' ', last_name) AS user_name, e_mail, city, address,
zip_code, registration_date FROM users ORDER BY registration_date";
$r = mysqli_query($dbc, $q);

```

```

if (mysqli_num_rows($r) < 1)
{
    echo '<p> Δεν υπάρχουν εγγεγραμμένοι χρήστες.</p>';
}
else{

while ($row = mysqli_fetch_array ($r, MYSQLI_ASSOC))
{
    $un = $row['user_name'];
    $e = $row['e_mail'];
    $c = $row['city'];
    $ad = $row['address'];
    $zip = $row['zip_code'];
    $date = $row['registration_date'];

    echo '<tr>
        <td>
            <table width="100%" border="0">
                <tr>
                    <td width="21%"><div align="left">'. $un. '</div></td>
                        <td width="17%"><div align="center">'. $e. '</div></td>
                        <td width="15%"><div align="center">'. $c. '</div></td>
                        <td width="14%"><div align="center">'. $ad. '</div></td>
                        <td width="12%"><div align="center">'. $zip. '</div></td>
                        <td width="13%"><div align="center">'. $date. '</div></td>
                        <td width="8%"><div align="center"><div id="but_delete">'. '<a
href="delete_user.php?uid={$row['user_id']}">'. ''. '</a>'. '</div></div></td>
                    </tr>
                </table>
            </td>
        </tr>
        ';
}
echo '
</tr>
<td height="30"></td>
<tr>
<td height="55" class="style3"><div align="right" class="style6 style11">
    <div align="left" class="style7">Επικοινωνία</div></div>
</td>
</tr>

<tr>
<td><table width="100%" border="0">
<tr>
    <td width="25%"><div align="left">Όνομα Χρήστη</div></td>
    <td width="17%"><div align="center">Τηλέφωνο</div></td>
    <td width="15%"><div align="center">Fax</div></td>
    <td width="43%">&nbsp;</td>
</tr>
</table></td>

```

```

</tr>
<td height="21">-----
-----</td>;

    $q2 = "SELECT CONCAT(first_name, ' ', last_name) AS user_name, telephone, fax
FROM users ORDER BY registration_date";
    $r2 = mysqli_query($dbc, $q2);
    if ($r2)
    {
        while ($row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
        {
            $un2 = $row2['user_name'];
            $tel = $row2['telephone'];
            $f = $row2['fax'];
            echo '<tr>
                <td>
                <table width="100%" border="0">
                <tr>
                <td width="25%"><div align="left">'. $un2. '</div></td>
                <td width="17%"><div align="center">'. $tel. '</div></td>
                <td width="15%"><div align="center">'. $f. '</div></td>
                <td width="43%">&nbsp;</td>
                </tr>
                </table>
                </td>
                </tr>
                ';
            }
        }
    }
?>

```

Περιγραφή Κώδικα

Το σενάριο αυτό δε διαφέρει πολύ από τα προηγούμενα παρά μόνο στην εκτέλεση της εντολής echo(). Για να εμφανίζεται ο κώδικας της HTML μέσω της echo πρέπει να περικλείεται σε μονά εισαγωγικά. Όταν όμως επιθυμούμε να εκτυπώσουμε μια μεταβλητή της PHP πρέπει να αλλάζουμε τα μονά εισαγωγικά σε διπλά.

πχ. echo 'κώδικας html'. 'μεταβλητή'. 'κώδικας html';

Στην αρχή εκτυπώνουμε μέσω της echo() τα ονόματα των στηλών (Όνομα Χρήστη, Email, Πόλη...) και έπειτα μέσω ενός ερωτήματος επιλογής ανακτάμε τα στοιχεία του πίνακα users από τη βάση δεδομένων. Με την συνάρτηση CONCAT() πραγματοποιείται η συνένωση δύο στηλών του πίνακα users. Της στήλης first_name και της στήλης last_name. Έπειτα στη συνένωση αυτή δίνουμε ένα όνομα: user_name.

Αν το ερώτημα δεν επιστρέφει καμία τιμή τότε δεν υπάρχουν εγγεγραμμένοι χρήστες στη βάση δεδομένων. Αλλιώς μέσω ενός βρόχου while εμφανίζουμε τα δεδομένα των χρηστών.

Μια δυνατότητα που έχει ο διαχειριστής είναι να διαγράψει όποιον χρήστη επιθυμεί. Αυτό επιτυγχάνεται μέσω ενός κουμπιού διαγραφής που μας μεταβιβάζει στη σελίδα delete_user.php. Για να γνωρίζουμε όμως ποιος είναι ο χρήστης που θα διαγραφεί πρέπει να μεταβιβάσουμε στη σελίδα

delete_user.php το κωδικό του χρήστη (user_id). Αυτό γίνεται με την προσάρτηση της μεταβλητής \$uid στην URL διεύθυνση χρησιμοποιώντας την σύνταξη:

page.php?name1=value1&name2=value2... =>

page.php = delete_user.php, name1 = uid, value1 = {\$row['user_id']}

Οι δύο ξεχωριστές SELECT έγιναν για λόγους μορφοποίησης. Στην δεύτερη επιλέγουμε τα στοιχεία επικοινωνίας.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #view_users { background-color: #000000; color: #000000; font-size: 14px; font-weight: bolder; background-repeat: repeat-y; margin-right: 30px; margin-left: 30px; padding-top: 30px; background-image: url(background%20img.jpg); padding-bottom: 70px; padding-left: 10px; }</pre>	<pre>.bookstore #but_delete a{ color: #F5F1CE; }</pre>

Πίνακας 4.24: Φύλλα CSS για την μορφοποίηση του πίνακα χρηστών

4.5.6 Διαγραφή Χρήστη

Κώδικας PHP

```
<?php
```

```
echo '
```

```
<tr>
```

```
<td height="55" class="style3"><div align="right" class="style6 style11">Πίνακας Χρηστών</div></td>
```

```
</tr>
```

```
<tr>
```

```
<td><table width="100%" border="0">
```

```
<tr>
```

```
<td width="21%"><div align="left">Όνομα Χρήστη</div></td>
```

```
<td width="17%"><div align="center">Email</div></td>
```

```
<td width="15%"><div align="center">Πόλη</div></td>
```

```
<td width="14%"><div align="center">Διεύθυνση</div></td>
```

```
<td width="12%"><div align="center">T.K.</div></td>
```

```
<td width="13%"><div align="center">Ημερομηνία Εγγραφής</div></td>
```

```
<td width="8%"><div align="center">Αφαίρεση</div></td>
```

```
</tr>
```

```
</table></td>
```

```
</tr>
```

```
<tr>
```

```

        <td height="21">-----</td>
    -----</td>
</tr>;

$un = $un2 = $e = $c = $ad = $tel = $f = $zip = $date = FALSE;
if (isset($_GET['uid']) && is_numeric($_GET['uid']))
{
    $uid = (int) $_GET['uid'];
    $q = "DELETE FROM users WHERE user_id = $uid ";
    $r = mysqli_query($dbc, $q);
    if ($r)
    {
        $q2 = "SELECT user_id, CONCAT(first_name, ' ', last_name) AS user_name, e_mail,
city, address, zip_code, registration_date FROM users ORDER BY registration_date";
        $r2 = mysqli_query($dbc, $q2);
        if (mysqli_num_rows($r2) < 1)
        {
            echo '<p> Δεν υπάρχουν εγγεγραμμένοι χρήστες.</p>';
        }else{
            while ($row = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
            {
                $un = $row['user_name'];
                $e = $row['e_mail'];
                $c = $row['city'];
                $ad = $row['address'];
                $zip = $row['zip_code'];
                $date = $row['registration_date'];

                echo '<tr>
                    <td>
                        <table width="100%" border="0">
                            <tr>
                                <td width="21%"><div align="left">'. $un .'</div></td>
                                <td width="17%"><div align="center">'. $e .'</div></td>
                                <td width="15%"><div align="center">'. $c .'</div></td>
                                <td width="14%"><div align="center">'. $ad .'</div></td>
                                <td width="12%"><div align="center">'. $zip .'</div></td>
                                <td width="13%"><div align="center">'. $date .'</div></td>
                                <td width="8%"><div align="center"><div id="but_delete">'. "<a
href='\"delete_user.php?uid = {$row['user_id']}\">\".' \".' </a>\".' </div></div></td>
                            </tr>
                        </table>
                    </td>
                </tr>
                ;
            }
        }
    }
    echo '
    </tr>
    <td height="30"></td>
    <tr>

```

```

<td height="55" class="style3"><div align="right" class="style6 style11">
<div align="left" class="style7">Επικοινωνία</div>
</div></td>
</tr>

<tr>
<td><table width="100%" border="0">
<tr>
<td width="25%"><div align="left">Όνομα Χρήστη</div></td>
<td width="17%"><div align="center">Τηλέφωνο</div></td>
<td width="15%"><div align="center">Fax</div></td>
<td width="43%">&nbsp;</td>
</tr>
</table></td>
<tr>
<td height="21">-----
-----</td>';

```

```

$q3 = "SELECT CONCAT(first_name, ' ', last_name) AS user_name, telephone,
fax FROM users ORDER BY registration_date";

```

```

$r3 = mysqli_query($dbc, $q3);
if ($r3)
{
    while ($row2 = mysqli_fetch_array ($r3, MYSQLI_ASSOC))
    {
        $un2 = $row2['user_name'];
        $tel = $row2['telephone'];
        $f = $row2['fax'];
        echo '<tr>
        <td>
        <table width="100%" border="0">
        <tr>
        <td width="25%"><div align="left">'. $un2. '</div></td>
        <td width="17%"><div align="center">'. $tel. '</div></td>
        <td width="15%"><div align="center">'. $f. '</div></td>
        <td width="43%">&nbsp;</td>
        </tr>
        </table>
        </td>
        </tr>
        ';
    }
}
}
?>

```


Περιγραφή Κώδικα

Η εμφάνιση των στοιχείων γίνεται με τον ίδιο τρόπο όπως και στο σενάριο της σελίδας view_users.php. Η διαφορά βρίσκεται στην πρόσθεση του ερωτήματος DELETE με το οποίο γίνεται η διαγραφή του χρήστη.

Μέσω του μητρώου \$_GET παίρνουμε τον κωδικό του χρήστη (\$uid) και ελέγχουμε αν είναι αριθμός και μάλιστα ακέραιος. Αυτός ο έλεγχος γίνεται για να μην προκύψει σφάλμα κατά την εκτέλεση του ερωτήματος.

4.5.7 Αλλαγή Στοιχείων Λογαριασμού

Κώδικας PHP

```
<?php
```

```
$un = $_SESSION['user_name'];  
$fn = $_SESSION['first_name'];  
$ln = $_SESSION['last_name'];  
$ad = $_SESSION['address'];  
$c = $_SESSION['city'];  
$zip = $_SESSION['zip_code'];  
$tel = $_SESSION['telephone'];  
$fax = $_SESSION['fax'];  
$e = $_SESSION['e_mail'];  
$p = $_SESSION['pass'];
```

```
if (isset($_POST['changed']))
```

```
{
```

```
    $errors = array();
```

```
    $trimmed = array_map('trim', $_POST);
```

```
    if (empty($trimmed['user_name']))
```

```
    {
```

```
        $errors[] = 'Παρακαλούμε συμπληρώστε το Όνομά σας.';
```

```
    }else{
```

```
        if (preg_match("/^\w.{2,20}$i", $trimmed['user_name']))
```

```
        {
```

```
            $un = mysqli_real_escape_string($dbc, $trimmed['user_name']);
```

```
        }else{
```

```
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το όνομα χρήστη.';
```

```
        }
```

```
    }
```

```
    if (empty($trimmed['first_name']))
```

```
    {
```

```
        $errors[] = 'Παρακαλούμε συμπληρώστε το Όνομά σας.';
```

```
    }else{
```

```
        if (preg_match("/^\w.{2,20}$i", $trimmed['first_name']))
```

```
        {
```

```
            $fn = mysqli_real_escape_string($dbc, $trimmed['first_name']);
```

```
        }else{
```

```
            $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το μικρό σας όνομα.';
```

```
        }
```

```

}
if (empty($trimmed['last_name']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Επώνυμό σας.';
}
else{
    if (preg_match('/^\w.{2,40}$/i', $trimmed['last_name']))
    {
        $ln = mysqli_real_escape_string($dbc, $trimmed['last_name']);
    }
    else{
        $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το επώνυμό σας.';
    }
}
if (empty($trimmed['address']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε τη Διεύθυνσή σας.';
}
else{
    if (preg_match('/^\w.+[ ][0-9]{2,40}$/i', $trimmed['address']))
    {
        $sad = mysqli_real_escape_string($dbc, $trimmed['address']);
    }
    else{
        $errors[] = 'Μόνο ψηφία και χαρακτήρες επιτρέπονται για το όνομα
διεύθυνσης.';
    }
}
if (empty($trimmed['city']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε την Πόλη σας.';
}
else{
    if (preg_match('/^\w.{2,30}$/i', $trimmed['city']))
    {
        $c = mysqli_real_escape_string($dbc, $trimmed['city']);
    }
    else{
        $errors[] = 'Μόνο χαρακτήρες επιτρέπονται για το όνομα της πόλης σας.';
    }
}
if (empty($trimmed['zip_code']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Ταχυδρομικό σας Κώδικα.';
}
else{
    if (preg_match('/^[d-]{2,10}$/i', $trimmed['zip_code']))
    {
        $zip = mysqli_real_escape_string($dbc, $trimmed['zip_code']);
    }
    else{
        $errors[] = 'Μόνο ψηφία και η "-" επιτρέπονται για τον ταχυδρομικό κώδικα.';
    }
}
if (empty($trimmed['telephone']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε το Τηλέφωνό σας.';
}
else{
    if (preg_match('/^[d-]{2,30}$/i', $trimmed['telephone']))

```

```

        {
            $tel = mysqli_real_escape_string($dbc, $trimmed['telephone']);
        }else{
            $errors[] = 'Μόνο ψηφία και η "-" επιτρέπονται για τον αριθμό τηλεφώνου.';
        }
    }
    if (empty($trimmed['fax']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε τον αριθμό FAX.';
    }else{
        if (preg_match('/^[d-]{2,30}$/i', $trimmed['fax']))
        {
            $fax = mysqli_real_escape_string($dbc, $trimmed['fax']);
        }else{
            $errors[] = 'Μόνο ψηφία και η "-" επιτρέπονται για τον αριθμό Fax.';
        }
    }
    if (empty($trimmed['e_mail']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Email σας.';
    }else{
        if (preg_match('/^[w.-]+@[w.]+\.[A-Za-z]{2,40}$/i', $trimmed['e_mail']))
        {
            $e = mysqli_real_escape_string($dbc, $trimmed['e_mail']);
        }else{
            $errors[] = 'Το email δεν είναι έγκυρο.';
        }
    }
    if (empty($trimmed['pass1']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το Κωδικό σας.';
    }else{
        if (preg_match('/^\w{2,40}$/i', $trimmed['pass1']))
        {
            if ($trimmed['pass1'] == $trimmed['pass2'])
            {
                $p = mysqli_real_escape_string($dbc, $trimmed['pass1']);
            }else{
                $errors[] = 'Ο Κωδικός δεν επιβεβαιώθηκε σωστά.';
            }
        }else{
            $errors[] = 'Μόνο ψηφία και χαρακτήρες επιτρέπονται για το κωδικό
πρόσβασης.';
        }
    }
}

if (empty($errors))
{
    $q = "UPDATE users SET user_name = '$un', first_name = '$fn', last_name = '$ln',
e_mail = '$e', address = '$ad', city = '$c', zip_code = '$zip', telephone = '$tel', fax = '$fax', pass =
SHA1('$p'), registration_date = NOW() WHERE user_id={$_SESSION['user_id']}";

```

```

$r = mysqli_query($dbc, $q);
if (mysqli_affected_rows($dbc) == 1)
{
    echo '<p>Η αλλαγή των στοιχείων ήταν επιτυχής.</p>';
    $q2 = "SELECT user_id, user_name, first_name, last_name, e_mail, address,
city, telephone, fax, zip_code FROM users WHERE user_id={$_SESSION['user_id']}";
    $r2 = mysqli_query($dbc, $q2);
    if ($r2)
    {
        $_SESSION = array();
        $_SESSION = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
    }else{
        echo '<p>Η σύνδεση με την Βάση Δεδομένων δεν ήταν εφικτή.
Παρακαλούμε προσπαθήστε ξανά.</p>';
    }
}else{
    echo '<p>Δεν ήταν δυνατή η ανανέωση των στοιχείων σας.</p>';
}
mysqli_close($dbc);
}else{
    echo '<p>Βρέθηκαν λάθη</p>';
    foreach ($errors as $msg)
    {
        echo " $msg\n<br />";
    }
}
}
?>

```

Περιγραφή Κώδικα

Στην αρχή δίνουμε αρχικές τιμές στις μεταβλητές που θα χρησιμοποιήσουμε για την εκτέλεση του σεναρίου. Οι μεταβλητές αυτές παίρνουν τα δεδομένα που είναι αποθηκευμένα στο μητρώο `$_SESSION` και αντιστοιχούν στα στοιχεία του χρήστη.

Όταν λοιπόν ο χρήστης καταχωρίσει τις καινούργιες τιμές στα πεδία τις φόρμας και πατήσει το κουμπί “Καταχώρηση Στοιχείων” - `if (isset($_POST['changed']))` – πρώτα θα γίνει έλεγχος των καινούριων δεδομένων όπως στην φόρμα εγγραφής χρηστών και αν δεν υπάρξουν σφάλματα θα εκτελεστεί το ερώτημα της SQL για την ανανέωση των στοιχείων του χρήστη.

```

$q = "UPDATE users SET user_name = '$un', first_name = '$fn', last_name = '$ln', e_mail = '$e', address = '$ad', city = '$c', zip_code = '$zip', telephone = '$tel', fax = '$fax', pass = SHA1('$p'), registration_date = NOW() WHERE user_id={$_SESSION['user_id']}";

```

Με το ερώτημα UPDATE αλλάζουμε τις υπάρχουσες καταχωρίσεις του πίνακα users με τα νέα δεδομένα. Η αλλαγή όμως δεν γίνεται σε όλες τις εγγραφές παρά μόνο σε εκείνη που ο κωδικός του εγγεγραμμένου χρήστη (user_id) ισούται με τον κωδικό του χρήστη που εκτελεί την φόρμα `{$_SESSION['user_id']}`.

Αν η εκτέλεση του ερωτήματος είναι επιτυχής επιλέγουμε τα καινούρια δεδομένα του χρήστη από τον πίνακα users και καταχωρούμε τις νέες τιμές στο μητρώο `$_SESSION` αφού πρώτα το αδειάσουμε.

```
$_SESSION = array();
$_SESSION = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
```

Φόρμα Αλλαγής στοιχείων

```
<form name="register" method="post" action="allagh_stoixeiwn.php">
  <table width="100%" border="0"><tr>
    <td width="96%" height="55" class="style3"><div align="right" class="style8
style11">Αλλαγή Στοιχείων</div></td></tr>
  </table>

  <p><span class="red">*</span> Όνομα Χρήστη:
<input type="text" name="user_name" size="40" maxlength="20" value="<?php echo $un; ?>" /></p>
  <p><span class="red">*</span> Όνομα :
<input type="text" name="first_name" size="40" maxlength="20" value="<?php echo $fn; ?>" /></p>
  <p><span class="red">*</span> Επώνυμο :
<input type="text" name="last_name" size="40" maxlength="40" value="<?php echo $ln; ?>" /></p>
  <p><span class="red">*</span> Διεύθυνση :
<input type="text" name="address" size="40" maxlength="40" value="<?php echo $ad; ?>" /></p>
  <p><span class="red">*</span> Πόλη :
<input type="text" name="city" size="40" maxlength="30" value="<?php echo $c; ?>" /></p>
  <p><span class="red">*</span> T.K. :
<input type="text" name="zip_code" size="40" maxlength="10" value="<?php echo $zip; ?>" /></p>
  <p><span class="red">*</span> Τηλέφωνο :
<input type="text" name="telephone" size="40" maxlength="30" value="<?php echo $tel; ?>" /></p>
  <p><span class="red">*</span> Fax :
<input type="text" name="fax" size="40" maxlength="30" value="<?php echo $fax; ?>" /></p>
  <p><span class="red">*</span> Email :
<input type="text" name="e_mail" size="40" maxlength="30" value="<?php echo $e; ?>" /></p>
  <p><span class="red">*</span> Κωδικός πρόσβασης :
<input type="password" name="pass1" size="40" maxlength="40" /></p>
  <p><span class="red">*</span> Επιβεβαίωση κωδικού πρόσβασης :
<input type="password" name="pass2" size="40" maxlength="40" /></p>

<p>&nbsp;</p>
<p align="right"><input type="submit" name="change" value="Καταχώρηση Στοιχείων" /></p>
<input type="hidden" name="changed" value="TRUE" />

</form>
```

Το ίδιο σενάριο και η ίδια φόρμα εκτελείται και για την αλλαγή των στοιχείων του διαχειριστή.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre> .bookstore #text_erea { margin-right: 30px; margin-left: 30px; color: #000000; font-family: "Times New Roman", Times, serif; background-color: #000000; background-repeat: repeat-y; background-image: url(background%20img.jpg); font-size: 16px; padding-top: 50px; } </pre>	<pre> .bookstore #text_erea form{ text-align: right; width: 570px; font-weight: bold; margin-right: 30px; margin-left: 280px; padding-bottom: 20px; } </pre>

Πίνακας 4.25: Φύλλα CSS για την μορφοποίηση της φόρμας αλλαγής στοιχείων

4.4.8 Εμφάνιση Βιβλίων

Τα βιβλία της ιστοσελίδας είναι χωρισμένα σε διάφορες κατηγορίες (Best Sellers, Μαγειρική, Εκπαίδευση, Προτεινόμενα κ.α.) όπου εμφανίζονται μόνο τρία από τα στοιχεία τους: ο τίτλος, ο συγγραφέας και ο εκδότης. Ο χρήστης μπορεί να έχει πρόσβαση σε όλα τα χαρακτηριστικά του βιβλίου που επιθυμεί πατώντας πάνω στον τίτλο του κάθε βιβλίου ο οποίος αποτελεί σύνδεσμο προς την αντίστοιχη σελίδα εμφάνισης όλων των χαρακτηριστικών.

Σενάριο Εμφάνιση Βιβλίων Best Seller

```
<?php
```

```

$sath_fn = $sath_ln = $sath_fn2 = $img = FALSE;
$q2 = "SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name,
image_name_small, best_url FROM books, authors, editors WHERE best_book =1 AND
books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id
DESC";
$r2 = mysqli_query($dbc, $q2);
if ($r2)
{
    while ($row = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
    {
        $sath_fn = $row['author_fn'];
        $sath_fn2 = $row['author2_fn'];
        $sath_ln = $row['author2_ln'];
        $img = "includes/books images/{$row['image_name_small']}";
echo '
<tr>
<td>
<table width="100%" border="0">
<tr>
<td width="27%">'.<img src=\'\".$img.\"\' />'.</td>

```

```

        <td width="73%">
        <div id="details">
        <table width="100%" border="0">
        <tr>
        <td width="24%" height="21">Τίτλος: </td>
<td width="76%">'<a href="\\". $row['best_url']\\"> <p>{$row['book_name']}</p> </a>'</td>
        </tr>
        <tr>
        <td width="24%" height="21">Συγγραφέας:</td>
<td width="76%">';
        if ($ath_fn != 'NULL')
        {
            echo "$ath_fn, ";
        }

        echo "{$row['author_in']}";

        if ($ath_fn2 != 'NULL')
        {
            echo " / $ath_fn2";
        }
        if ($ath_in != 'NULL')
        {
            echo ", $ath_in";
        }
        echo '</td>
        </tr>
        <tr>
        <td width="24%" height="21">Εκδότης: </td>
        <td width="76%">'<{$row['editor_name']}</td>
        </tr>
        <tr>
        </table>
        </div>
        </td>
        </tr>
        </table> </td>
        </tr>
        <td height="23">-----
        -----</td>
        </tr>
        ';
}}?>

```


Περιγραφή Σεναρίου

Στην αρχή επιλέγουμε τα δεδομένα των πινάκων books, authors και editors από την βάση δεδομένων. Στον πίνακα books υπάρχουν τρεις στήλες που καθορίζουν αν το βιβλίο ανήκει στα Best sellers, στις Νέες Κυκλοφορίες ή στα Προτεινόμενα. Αυτές είναι οι new_book, best_book και sug_book. Αν κάποια από αυτές έχει τιμή 1 τότε το βιβλίο ανήκει σε μια από αυτές τις κατηγορίες. Για να επιλέξουμε τα βιβλία που ανήκουν στα Best Sellers γράφουμε:

WHERE best_book =1

Η συνένωση των πινάκων γίνεται μέσω των ξένων κλειδιών:

books.author_id = authors.author_id AND books.editor_id = editors.editor_id

Στο τέλος ταξινομούμε τα δεδομένα με βάση τον κωδικό (book_id) και με φθίνουσα σειρά (DESC) έτσι ώστε όταν προσθέτουμε ένα βιβλίο στην λίστα να εμφανίζεται πρώτο.

Αν δεν προκύψει κάποιο σφάλμα με την εκτέλεση του ερωτήματος εμφανίζουμε τα δεδομένα μέσω ενός βρόχου. Τα δεδομένα αυτά αφορούν τον τίτλο, τους συγγραφείς, τον εκδότη, μια εικόνα και το url που αντιστοιχεί στο κάθε βιβλίο (το url αλλάζει ανάλογα με την ιδιότητα του χρήστη).

Επίσης είναι πολύ πιθανό το όνομα κάποιου συγγραφέα να έχει NULL τιμή, αφού το μόνο υποχρεωτικό πεδίο είναι το επώνυμο του πρώτου συγγραφέα. Γι' αυτό το λόγο πρέπει πρώτα να γίνεται έλεγχος αυτών των τιμών και μετά η εκτύπωσή τους.

Το σενάριο αυτό είναι ίδιο για όλες τις κατηγορίες της ιστοσελίδας. Η μόνο που διαφέρει είναι το ερώτημα της SQL, αφού τα δεδομένα αλλάζουν για κάθε κατηγορία.

Νέες Κυκλοφορίες	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, new_url FROM books, authors, editors WHERE new_book =1 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Προτεινόμενα βιβλία	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, sug_url FROM books, authors, editors WHERE sug_book =1 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Εκπαίδευση	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors WHERE category_id =1 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Λογοτεχνία	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors WHERE category_id =2 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Ψυχολογία	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors WHERE category_id =3 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Θρησκεία	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors

	WHERE category_id =5 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Φιλοσοφία	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors WHERE category_id =6 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Ιστορία, Γεωγραφία	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors WHERE category_id =4 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC
Μαγειρική	SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_small, url FROM books, authors, editors WHERE category_id =7 AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id ORDER BY book_id DESC

Πίνακας 4.26: Ερωτήματα SQL για την επιλογή των δεδομένων από κάθε κατηγορία

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #details { font-size: 16px; font-family: "Times New Roman", Times, serif; } .bookstore #details a { color: #000000; font-family: "Times New Roman", Times, serif; font-size: 18px; font-style: italic; } .bookstore #details a:hover { color: #C10000; font-family: "Times New Roman", Times, serif; font-size: 18px; font-style: italic; }</pre>	<pre>.bookstore #view_books { background-color: #000000; color: #000000; font-size: 16px; font-weight: bolder; background-repeat: repeat-y; margin-right: 30px; margin-left: 30px; padding-top: 50px; background-image: url(background%20img.jpg); }</pre>

Πίνακας 4.27: Φύλλα CSS για την μορφοποίηση της εμφάνισης των κατηγοριών

Κατηγορίες Βιβλίων
Λογοτεχνία

Εκπαίδευση

Λογοτεχνία

Ψυχολογία

Θρησκεία

Φιλοσοφία

Ιστορία, Γεωγραφία

Μαγειρική



Τίτλος: *Το φθινόπωρο της μάγισσας*

Συγγραφέας: Καίτη, Οικονόμου

Εκδότης: Ψυχογιός



Τίτλος: *Το κοιμητήριο της Πράγας*

Συγγραφέας: Umberto, Eco

Εκδότης: Ψυχογιός



Τίτλος: *Κομμένα λουλούδια*

Εικόνα 4.58: Εμφάνιση βιβλίων κατηγοριών

Σενάριο Εμφάνιση ενός βιβλίου

```

<table width="620" border="0">
  <tr>
    <td height="24" colspan="2"><div align="right" class="style8 style11">Best Sellers</div></td>
    <td>
      <?php
        $q2 = "SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name,
image_name_big, category_name, price, ISBN, description FROM books, authors, editors, categories
WHERE best_book =1 AND books.author_id = authors.author_id AND books.editor_id =
editors.editor_id AND books.category_id = categories.category_id ORDER BY book_id LIMIT 3, 1";
        $r2 = mysqli_query($dbc, $q2);
        $row = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
        $img = "includes/books images/{$row['image_name_big']}";
      ?>
    </td>
  </tr>
  <tr>
    <td width="28%"><?php echo "<img src='\"'\".\"$img.\"'\" />"; ?></td>
    <td width="72%" rowspan="2" bordercolor="#000000">
      <table width="101%" border="0">
        <tr>
          <td colspan="3"><div align="center" class="style9"><strong><?php echo
"<p>{$row['book_name']}</p>"; ?></strong></div></td>

```

```

        </tr>
    <tr>
        <td width="45%" height="24"><div align="right"><span
class="stele4">Συγγραφέας:</span></div></td>
        <td class="stele4">&nbsp;</td>
        <td class="stele4"><?php echo "<p>{$row['author_fn']}, {$row['author_ln']} / {$row['author2_fn']},
{$row['author2_ln']}</p>"; ?></td>
    </tr>
    <tr>
        <td width="45%"><div align="right"><span class="stele4">Εκδόσεις:</span></div></td>
        <td class="stele4">&nbsp;</td>
        <td class="stele4"><?php echo "<p>{$row['editor_name']}</p>"; ?></td>
    </tr>
    <tr>
        <td width="45%"><div align="right"><span class="stele4">ISBN: </span></div></td>
        <td class="stele4">&nbsp;</td>
        <td class="stele4"><?php echo "<p>{$row['ISBN']}</p>"; ?></td>
    </tr>
    <tr>
        <td width="45%"><div align="right"><span class="stele4">Κατηγορία: </span></div></td>
        <td width="2%" class="stele4">&nbsp;</td>
        <td width="53%" class="stele4"><?php echo "<p>{$row['category_name']}</p>"; ?></td>
    </tr>
    <tr>
        <td width="45%"><div align="right"><span class="stele4">Τιμή:</span></div></td>
        <td class="stele4">&nbsp;</td>
        <td colspan="2" class="stele4"><?php echo "<p>{$row['price']} €</p>"; ?></td>
    <td height="211" colspan="2"><p class="style10"><strong>Περιγραφή</strong></p>
    <p align="justify" class="style5"><?php echo "{$row['description']}"; ?></p>
    </td>
    </tr>
</table>

```

Περιγραφή Σεναρίου

Στο σενάριο αυτό δε χρειάζεται να χρησιμοποιήσουμε κάποιο βρόχο για την εμφάνιση των δεδομένων, γιατί επιλέγουμε μόνο ένα βιβλίο από την βάση δεδομένων. Σαν αποτέλεσμα έχουμε την διάσπαση του κώδικα PHP μέσα στα πεδία ενός πίνακα.

Στο συγκεκριμένο σενάριο επιλέγουμε το τέταρτο βιβλίο από την λίστα που δημιουργείται καθώς εκτελούμε το ερώτημα SELECT.

```

SELECT book_name, author_fn, author_ln, author2_fn, author2_ln, editor_name, image_name_big,
category_name, price, ISBN, description FROM books, authors, editors, categories WHERE best_book =1
AND books.author_id = authors.author_id AND books.editor_id = editors.editor_id AND
books.category_id = categories.category_id ORDER BY book_id

```

book_name	author_fm	author_ln	author2_fm	author2_ln	editor_name	image_name_big	category_name	price	ISBN	description
Όταν αγαπάς είναι για πάντα	Μαρία	Τζέριτα	NULL	NULL	Ψυχολόγος	Όταν αγαπάς είναι για πάντα - big.jpg	Λογοτεχνία	17.70	960-496-000-8	Αυτά τα δυο τόσο διαφορετικά παιδιά, η Αφροδίτη κα...
Ο θάλας	Stephen	King	NULL	NULL	Βελ / Χαρλένκ Ελλάς	Ο θάλας - big.jpg	Λογοτεχνία	14.00	960-450-000-7	Είναι ένα ησυχαιο φθισομαρμένο πρινοά στη μικρή πόλη...
Λευκή αρχίδα	Καίτη	Δικονόμου	NULL	NULL	Ψυχολόγος	Λευκή αρχίδα -- big.jpg	Λογοτεχνία	20.18	960-453-775-3	Ήθελα να σου πάρω ένα ρολόι για να θυμάσαι κάθε ώρ...
Δέκα ετών, διαζευγμένη	Delphine	Μπουι	Νογούδ	Αλι	Modern Times	Δέκα ετών, διαζευγμένη - big.jpg	Ιστορία, Γεωγραφία	13.08	960-691-745-2	Το όνομά μου είναι Νοζόν και μεγάλωσα σ' ένα χωρ...
Τα σακά	Ιωάννα	Καρυστιάνη	NULL	NULL	Καστανιώτη	Τα σακά - big.jpg	Λογοτεχνία	17.04	960-03-5129-5	Τι έχουν, μαρέ, τα δικά σας σακά. Μια χρωμακασία...
Αργυρά. Κάθε μέρα ελληνικά	Αργυρά	Μπαρμπάριγου	NULL	NULL	Λυμπερή	Αργυρά Κάθε μέρα ελληνικά - big.jpg	Μαγειρική	30.43	960-99261-3-4	Η φιλοσοφία των συταγιών της Αργυρούς είναι η αναζή...
Η εβδομη αίσθηση	Μίλης	Ανδρουλάκης	NULL	NULL	Καστανιώτη	Η εβδομη αίσθηση - big.jpg	Λογοτεχνία	16.15	960-03-5175-9	Ο λαθρεμπόστης με το κόκορα και το μούρο Έβδομη...
Ψυρί κι αλάτι	Γιάννης	Λουκάκος	NULL	NULL	Ψυχολόγος	Ψυρί κι αλάτι - big.jpg	Μαγειρική	25.24	960-496-101-2	Στο εξαιρετικό ασθητικό από βιβλίο, ο Γιάννης...
Αγαστή συνεργασία	Αναστασία	Καλλινοτή	NULL	NULL	A. A. Λυβάνη	Αγαστή συνεργασία - big.jpg	Λογοτεχνία	15.21	960-14-2206-4	Ο Κωνσταντίνος είχε όλες τις προδιαγραφές να ζήσει...
Πάρτο Λέονε	Λευτέρης	Κατσώνης	NULL	NULL	A. A. Λυβάνη	Πάρτο Λέονε - big.jpg	Λογοτεχνία	16.66	960-14-2166-1	Ο Πάρτο Λέονε είναι ένα μυθιστόρημα για τον Πάρτο...
Το κύμα του έρωτα	Γιάννης	Αλεξάνδρου	Μαρία	Αλεξάνδρου	A. A. Λυβάνη	Το κύμα του έρωτα - big.jpg	Λογοτεχνία	16.23	960-14-2213-7	>Η Άρτεμη και ο Δανιήσης σήκουν σε κόρφοις απ'θ...
Ένα πικασμένο στόμα	Λένα	Διβάνη	NULL	NULL	Καστανιώτη	Ένα πικασμένο στόμα - big.jpg	Λογοτεχνία	17.04	960-03-5087-6	Ο Γιάννης Γεωργιάδης είναι ένας φιλόδοξος κομμωτής...

Εικόνα 4.59: Περιεχόμενα πίνακα books

Με την πρόσθεση του χαρακτηριστικού **LIMIT 3, 1** ζητάμε να επιλεγούν **1** εγγραφές ξεκινώντας από την εγγραφή **3**. Δηλαδή επιλέγουμε το βιβλίο “Δέκα ετών, διαζευγμένη”. Αυτή τη φορά διαλέγουμε όλα τα στοιχεία του βιβλίου μαζί με την μεγάλη εικόνα που θέλουμε να εμφανίζεται.

Με τον ίδιο τρόπο επιλέγουμε από την βάση δεδομένων το βιβλίο που επιθυμούμε ανάλογα την κατηγορία στην οποία ανήκει και την σειρά που έχει κατά την εκτέλεση του ερωτήματος.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση

```
.bookstore #view_one_book {
    background-image: url(background%20img.jpg);
    color: #000000;
    font-size: 16px;
    font-style: normal;
    font-weight: normal;
    background-repeat: repeat;
    margin-right: 30px;
    margin-left: 30px;
    padding-top: 80px;
    padding-bottom: 40px;
}
```

Πίνακας 4.28: Φύλλα CSS για την μορφοποίηση της εμφάνιση των στοιχείων ενός βιβλίου

Εκπαίδευση
Λογοτεχνία
Ψυχολογία
Θρησκεία
Φιλοσοφία
Ιστορία, Γεωγραφία
Μαγειρική

Δέκα ετών, διαζευγμένηΣυγγραφέας: *Delphine, Minoui / Nojoud, Ali*Εκδόσεις: *Modern Times*ISBN: *960-691-745-2*Κατηγορία: *Ιστορία, Γεωγραφία*Τιμή: *13.08 €***Περιγραφή**

Το όνομά μου είναι Νοζούντ και μεγάλωσα σ' ένα χωριό της Υεμένης. Είμαι δέκα ετών, τουλάχιστον έτσι νομίζω. Στη χώρα μου, τα παιδιά της επαρχίας δεν έχουν επίσημα έγγραφα ούτε πιστοποιητικά γέννησης. Οι γονείς μου με ανάγκασαν να παντρευτώ έναν άντρα που είχε την τριπλάσια ηλικία από εμένα. Με κακοποιούσε σεξουαλικά και όταν αντιστεκόμουν με χτυπούσε. Ένα πρωί που έφυγα από το σπίτι για να αγοράσω ψωμί, ανέβηκα τρέμοντας στο λεωφορείο και πήγα στο δικαστήριο. Θέλω να μιλήσω στο δικαστή! είπα. Από τότε που θυμάμαι τον εαυτό μου, έλεγα ναί σε όλα. Εκείνη τη μέρα όμως, είχα αποφασίσει να πω όχι. Η αληθινή ιστορία μιας μικρής κοπέλας από την Υεμένη που τόλμησε να αγηφήσει τις αυστηρές

Εικόνα 4.59: Εμφάνιση στοιχείων ενός βιβλίου

4.5.9 Αναζήτηση Βιβλίων

Σενάριο Απλής Αναζήτησης

```
<?php
```

```
if (isset($_POST['searching']))
{
    $errors = array();
    $trimmed = array_map('trim',$_POST);
    $ath_fn = $ath_ln = $ath_fn2 = $f = FALSE;
    if (empty($trimmed['find']))
    {
        $table[] = 'Δεν συμπληρώσατε τον όρο αναζήτησής σας.';
    }else{
        $f = mysqli_real_escape_string($dbc, $trimmed['find']);
    }
    if (empty($errors))
    {
        $q2 = "SELECT book_name, author_fn, author_ln, author2_fn, author2_ln,
editor_name, category_name, url
FROM books, authors, editors, categories
```

```

WHERE ( MATCH (author_fn, author_ln) AGAINST ('$f') OR MATCH (author2_fn, author2_ln)
AGAINST ('$f') OR {$trimmed['field']} LIKE '%$f%')
AND books.author_id = authors.author_id
AND books.editor_id = editors.editor_id
AND books.category_id = categories.category_id";

```

```

    $r2 = mysqli_query($dbc, $q2);
    if ($r2)
    {
        while ($row = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
        {
            $ath_fn = $row['author_fn'];
            $ath_fn2 = $row['author2_fn'];
            $ath_ln = $row['author2_ln'];
            echo "<a href='\"'\".\"$row['url'].\"'\"'\"><p>{$row['book_name']}</p></a>";
            echo "\n";
            if ($ath_fn != 'NULL')
            {
                echo "$ath_fn, ";
            }
            echo "{$row['author_ln']}";
            if ($ath_fn2 != 'NULL')
            {
                echo " / $ath_fn2";
            }
            if ($ath_ln != 'NULL')
            {
                echo ", $ath_ln";
            }
            echo "\n";
            echo "<p>{$row['editor_name']}</p>";
            echo "\n";
            echo "<p>{$row['category_name']}</p>";
            echo "\n";
            echo "-----";
            echo "\n";
        }
    }else
        $table[] = 'Δεν βρέθηκαν αποτελέσματα.';
}
else{
    echo '<p>Βρέθηκαν λάθη</p>';
    foreach ($table as $t)
    {
        echo "$t\n<br />";
    }
}
echo "<p>Αναζήτηση για: $f</p>";
}
?>

```


Περιγραφή Σεναρίου

Στην αρχή ελέγχουμε αν ο χρήστης έχει συμπληρώσει τον όρο αναζήτησης, δηλαδή αν έχει δώσει τιμή στο πεδίο find της φόρμας. Αν ναι τότε εκτελείται το ερώτημα SELECT το οποίο επιστρέφει την καταχώριση της βάσης δεδομένων που περιέχει τον όρο αναζήτησης.

Αφού έχουμε ορίσει ευρετήριο τύπου FULLTEXT στον πίνακα authors μπορούμε με τον όρο MATCH...AGAINST να εμφανίσουμε τις εγγραφές που ταιριάζουν με τον όρο αναζήτησης.

SELECT * FROM όνομα_πίνακα WHERE MATCH (στήλες) AGAINST (λέξεις-κλειδιά)

Επίσης με τον όρο LIKE ελέγχουμε αν δύο συμβολοσειρές είναι ίσες. Το σύμβολο του ποσοστού (%) αντιπροσωπεύει κανένα, έναν, ή περισσότερους χαρακτήρες.

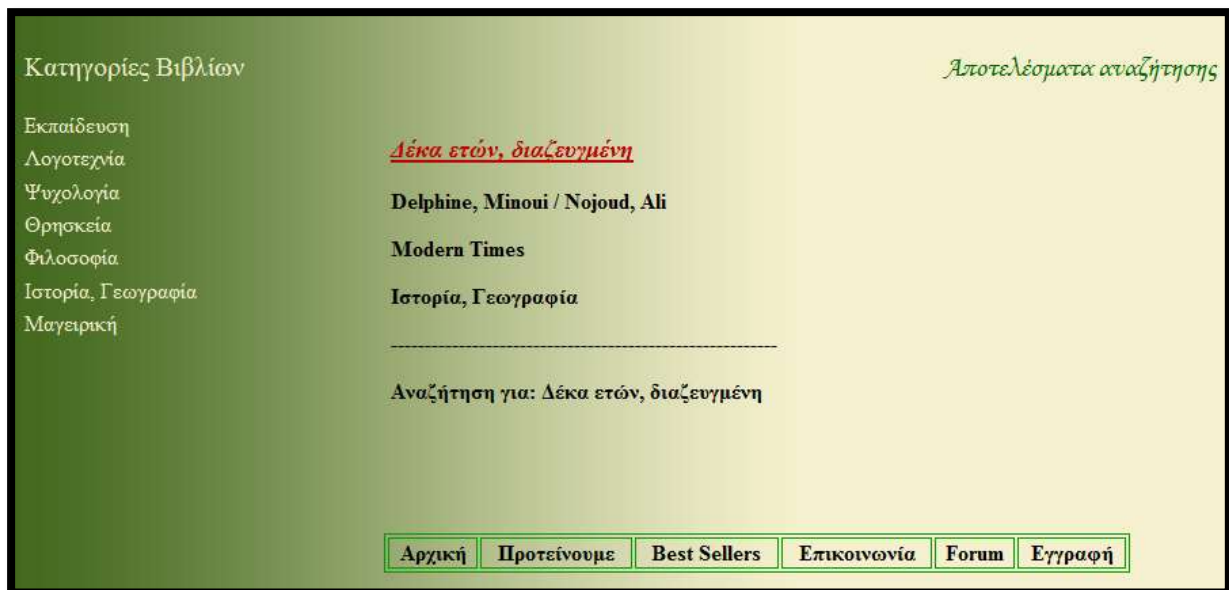
πχ. **SELECT * FROM users WHERE first_name LIKE '%Sofia%'**

Αυτό το ερώτημα θα επιστρέψει όλες τις γραμμές που στο πεδίο first_name έχουν τιμή που ξεκινάει και τελειώνει με τη συμβολοσειρά Sofia.

Στο συγκεκριμένο σενάριο ο όρος `{$trimmed['field']}` αντιστοιχεί στην στήλη που θα επιλέξει ο χρήστης από την φόρμα (τίτλος, συγγραφέας, εκδότης) και η μεταβλητή `$f` αποτελεί τον όρο αναζήτησης.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre>.bookstore #search_result { color: #000000; font-size: 16px; font-weight: bolder; margin-right: 30px; margin-left: 30px; padding-top: 50px; background-image: url(background%20img.jpg); background-repeat: repeat-y; font-family: "Times New Roman", Times, serif; }</pre>	<pre>.bookstore #search_result a { color: #F5F1CE; font-family: "Times New Roman", Times, serif; font-size: 16px; } .bookstore #search_result a:hover { color: #C10000; font-family: "Times New Roman", Times, serif; font-size: 16px; }</pre>

Πίνακας 4.29: Φύλλα CSS για την μορφοποίηση των αποτελεσμάτων αναζήτησης



Εικόνα 4.60: Εμφάνιση αποτελεσμάτων αναζήτησης

4.5.10 Λίστα αγαπημένων βιβλίων

Υπάρχουν τρεις σελίδες που απευθύνονται στην λίστα με τα αγαπημένα. Η πρώτη σελίδα αφορά την προσθήκη ενός βιβλίου στην λίστα (wishlist_add.php) στην οποία μεταφερόμαστε από τον σύνδεσμο “προσθήκη στη wishlist” που εμπεριέχεται στις σελίδες των βιβλίων. Στη δεύτερη (wishlist.php) μεταφερόμαστε από τον σύνδεσμο “Wishlist” που βρίσκεται στον πίνακα που περιέχει την φόρμα logout. Η τρίτη σελίδα (wishlist_delete.php) διαγράφει το βιβλίο από την λίστα όταν πατάμε στον σύνδεσμο “Διαγραφή από την Wishlist”.

Σενάριο PHP (wishlist_add.php)

<?php

```

if (isset($_GET['bid']) && is_numeric($_GET['bid']))
{
    $bid = (int) $_GET['bid'];
    $q = "SELECT book_id, book_name, image_name_small, url_after, FROM books WHERE
book_id = $bid";
    $r = mysqli_query($dbc, $q);
    if ($r)
    {
        $row = mysqli_fetch_array ($r, MYSQLI_ASSOC);
        $imgs = "includes/books images/{$row['image_name_small']}";
        $q2 = "SELECT * FROM wishlist WHERE book_id = $bid";
        $r2 = mysqli_query($dbc, $q2);
        if (mysqli_num_rows($r2) == 1)
        {
            echo '<p>Το βιβλίο υπάρχει ήδη στα αγαπημένα σας.</p>';
        }
    }

```

```

        exit();
    }else{

        $q3 = "INSERT INTO wishlist (book_id, user_id, b_name, img, w_url, date)
VALUES ('{$row['book_id']}', '$_SESSION[user_id]', '{$row['book_name']}', '$imgs',
'{$row['url_after']}', NOW() )";
        $r3 = mysqli_query($dbc, $q3);
        if ($r3)
        {
            $wurl = $wb = $wi = $imgs = FALSE;
            $q4 = "SELECT * FROM wishlist WHERE user_id = {$_SESSION[user_id]}
ORDER BY date";
            $r4 = mysqli_query($dbc, $q4);
            if ($r4)
            {
                while ($row = mysqli_fetch_array ($r4, MYSQLI_ASSOC))
                {
                    $wurl = $row['w_url'];
                    $wb = $row['b_name'];
                    $wi = $row['img'];

                    echo '
                        <td>
                        <table width="100%" border="0">
                        <tr>
                        <td width="24%" height="27" rowspan="3">'.<img src=\''.$wi.'" />'.<div align="left">
                        </div></td>
                        <td height="60" colspan="2"><span class="style10">'.<a href=\''.$wurl.'"><p>$wb</p>
                        </a>'.</span></td>
                        </tr>
                        <tr>
                        <td height="45"></td>
                        <td height="45">'.<a href="kalathi_add.php?bid={$row['book_id']}">'.<b>Προσθήκη στο
                        <b>Καλάθι</b>'.</a>'.</td>
                        </tr>
                        <tr>
                        <td width="8%"></td>
                        <td width="68%">'.<a href="wishlist_delete.php?bid={$row['book_id']}">'.<b>Διαγραφή από τη
                        <b>Wishlist</b>'.</a>'.</td>
                        </tr>
                        </table></td></tr>
                        <tr>
                        <td height="21">-----
                        -----</td>
                        </tr>
                    ';
                }
            }else{
                echo '<p>Προσπαθήστε ξανά.</p>';
            }
        }
    }
}
?>

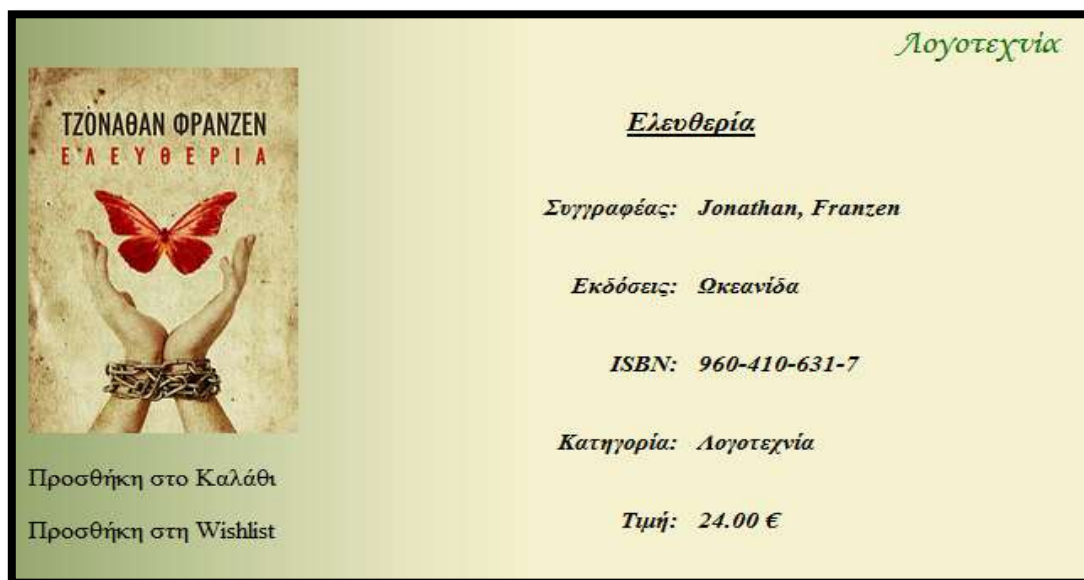
```

Περιγραφή Σεναρίου

Όταν ένας χρήστης εισέρχεται στον λογαριασμό στις σελίδες που εμφανίζονται τα στοιχεία των βιβλίων προσθέτονται δύο σύνδεσμοι. Ο ένας είναι για την προσθήκη του βιβλίου στο καλάθι και ο άλλος για την προσθήκη του βιβλίου στη Wishlist.

```
<div id="buttons">
  <?php
  echo "<p><a href='\"kalathi_add.php?bid={\$row['book_id']}\">Προσθήκη στο
  Καλάθι</a></p>";
  echo "<p><a href='\"wishlist_add.php?bid={\$row['book_id']}\">Προσθήκη στη
  Wishlist</a></p>"; ?>
</div>
```

Στο κομμάτι αυτό του κώδικα έχουμε δύο URL. Το ένα μας μεταφέρει στην σελίδα kalathi_add.php και το άλλο στην σελίδα wishlist_add.php. Και στις δύο διευθύνσεις προσθέτουμε τον κωδικό του βιβλίου για να γνωρίζει η σελίδα που θα μεταβούμε ποιο βιβλίο θα προστεθεί.



Εικόνα 4.61: Σύνδεσμοι προς το καλάθι και την wishlist

Το σενάριο ξεκινάει με τον έλεγχο του book_id (κωδικός βιβλίου) και συνεχίζει με την επιλογή των χαρακτηριστικών του από την βάση δεδομένων. Αν το ερώτημα εκτελέστηκε σωστά εκχωρούμε στη μεταβλητή \$imgs την διαδρομή προς την εικόνα του βιβλίου.

Έπειτα ελέγχουμε αν το βιβλίο υπάρχει ήδη στη Wishlist. Αν όχι τότε εκτελείτε το ερώτημα εισαγωγής INSERT που προσθέτει στον πίνακα wishlist της βάσης δεδομένων τον κωδικό του βιβλίου, τον κωδικό του χρήστη, τον τίτλο του βιβλίου, την εικόνα του και το όνομα της σελίδας που εμφανίζει τα χαρακτηριστικά του.

Στη συνέχεια επιλέγουμε τα βιβλία που έχει καταχωρήσει ο συγκεκριμένος χρήστης στον πίνακα wishlist και εμφανίζουμε τον τίτλο και την εικόνα του κάθε βιβλίου μαζί με δύο συνδέσμους. Ο ένας σύνδεσμος επιτρέπει την προσθήκη του βιβλίου στο καλάθι και ο δεύτερος την διαγραφή του βιβλίου από την Wishlist.


```

        <tr>
<td width="8%"></td>
<td width="68%">'.<a href=\'wishlist_delete.php?bid={ $row['book_id']}\\'>'. Διαγραφή από τη
Wishlist'.</a>'.</td>
        </tr>
</table>
</td>
</tr>
<tr>
<td height="21">-----
-----</td>
</tr>
';
    }
}
?>

```

Το σενάριο αυτό είναι ίδιο με εκείνο της σελίδας wishlist_add.php για την εμφάνιση των βιβλίων της wishlist.

Σενάριο PHP (wishlist_delete.php)

```
<?php
```

```

$wurl = $wb = $wi = $imgs = FALSE;
if (isset($_GET['bid']) && is_numeric($_GET['bid']))
{
    $bid = (int) $_GET['bid'];
    $q = "DELETE FROM wishlist WHERE book_id = '$bid' ";
    $r = mysqli_query($dbc, $q);

    $q2 = "SELECT * FROM wishlist WHERE user_id = {$_SESSION['user_id']}";
    $r2 = mysqli_query($dbc, $q2);
    if (mysqli_num_rows($r2) < 1)
    {
        echo '<p>Η λίστα με τα αγαπημένα σας είναι κενή.</p>';
    }else{
        while ($row = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
        {
            $wurl = $row['w_url'];
            $wb = $row['b_name'];
            $wi = $row['img'];
            echo '
<td>
<table width="100%" border="0">
<tr>
<td width="24%" height="27" rowspan="3">'.<img src=\'\'.$wi.\' />'.<div align="left">
</div></td>
<td height="60" colspan="2"><span class="style10">'.<a href=\'\'.$wurl.\'><p>$wb</p>
</a>'.</span></td></tr>

```

```

<tr>
<td height="45"></td>
<td height="45">'.<a href="kalathi_add.php?bid={$row['book_id']}">'.Προσθήκη στο
Καλάθι.'</a>'</td></tr>
<tr>
<td width="8%"></td>
<td width="68%">'.<a href="wishlist_delete.php?bid={$row['book_id']}">'.Διαγραφή από τη
Wishlist.'</a>'</td>
</tr></table>
</td></tr>
<tr>
<td height="21">-----
-----</td>
</tr>';
}} ?>

```

Περιγραφή Σεναρίου

Ο κώδικας που προστίθεται είναι η διαγραφή του βιβλίου με ένα ερώτημα DELETE. Η συνέχεια του σεναρίου είναι ίδια με τα δύο προηγούμενα.

DELETE FROM wishlist WHERE book_id = '\$bid'

Το κώδικα του βιβλίου τον παίρνουμε μέσω του μητρώου \$_GET() αφού πρώτα τον έχουμε προσαρτήσει στην διεύθυνση URL.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre> .bookstore #wishlist { background-color: #000000; color: #c05711; font-size: 20px; font-weight: normal; background-repeat: repeat-y; margin-right: 30px; margin-left: 30px; padding-top: 50px; background-image: url(background%20img.jpg); padding-bottom: 70px; font-family: "Monotype Corsiva"; } .bookstore #wishlist a { font-family: "Monotype Corsiva"; font-size: 20px; color: #c05711; } </pre>	<pre> .bookstore #wishlist a: hover { font-family: "Monotype Corsiva"; font-size: 20px; color: #c05711; text-decoration: underline; } .bookstore #buttons a { font-family: "Times New Roman", Times, serif; font-size: 16px; color: #000000; } .bookstore #buttons a: hover { font-family: "Times New Roman", Times, serif; font-size: 16px; color: #C10000; text-decoration: none; } </pre>

Πίνακας 4.30: Φύλλα CSS για την μορφοποίηση της Wishlist

4.5.11 Καλάθι Αγορών

Στο καλάθι αγορών ο χρήστης μπορεί να καταχωρήσει τα βιβλία που θέλει να αγοράσει. Η προσθήκη ενός βιβλίου γίνεται μέσω του συνδέσμου “Προσθήκη στο καλάθι” που βρίσκεται στην σελίδα εμφάνισης των χαρακτηριστικών του κάθε βιβλίου. Επίσης μπορεί να έχει πρόσβαση στο καλάθι του για να επεξεργαστεί τις καταχωρίσεις του από τον σύνδεσμο “Καλάθι αγορών” (στον πίνακα με την φόρμα logout) και να μεταβεί στην αγορά τους – Ολοκλήρωση Αγοράς .

Για σκοπό αυτό δημιουργήθηκαν τέσσερις σελίδες:

- kalathi_add.php – Προσθήκη βιβλίου στο καλάθι
- kalathi_agorwn.php – Εμφάνιση περιεχομένων καλαθιού
- kalathi_delete.php – Διαγραφή ενός βιβλίου από το καλάθι
- kalathi_en.php – Ενημέρωση καλαθιού

Σενάριο PHP (kalathi_add.php)

<?php

```
if (isset($_GET['bid']) && is_numeric($_GET['bid']))
{
    $bid = (int) $_GET['bid'];
    $qty = FALSE;
    $q = "SELECT book_id, book_name, price, url_after, category_id, new_book, best_book,
sug_book FROM books WHERE book_id = $bid";
    $r = mysqli_query($dbc, $q);
    $row = mysqli_fetch_array ($r, MYSQLI_ASSOC);
    if ($r)
    {
        $q2 = "SELECT * FROM kalathi WHERE book_id = $bid";
        $r2 = mysqli_query($dbc, $q2);
        $row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
        $qty = $row2['quantity'];
        if (mysqli_num_rows($r2) == 1)
        {
            $qty = $qty + 1;
            $q3 = "UPDATE kalathi SET quantity = $qty WHERE book_id = $bid";
            $r3 = mysqli_query($dbc, $q3);
        }else{
            $qty = 1;
            $q3 = "INSERT INTO kalathi (book_id, user_id, kal_book, pr, quantity, k_url,
date) VALUES ('{$row['book_id']}', '$_SESSION[user_id]', '{$row['book_name']}', '{$row['price']}',
'$qty', '{$row['url_after']}', NOW() )";
            $r3 = mysqli_query($dbc, $q3);
        }
        if ($r3)
        {
            $q4 = "SELECT * FROM kalathi WHERE user_id = {$_SESSION[user_id]}
ORDER BY date";
            $r4 = mysqli_query($dbc, $q4);
            if ($r4)
```

```

    {
        $t = $kurl = $kb = $kp = $subtotal = FALSE;
        $qu = array();
        $total = 0;
        while ($row3 = mysqli_fetch_array ($r4, MYSQLI_ASSOC))
        {
            $kurl = $row3['k_url'];
            $kb = $row3['kal_book'];
            $kp = $row3['pr'];
            $qty = $row3['quantity'];
            $subtotal = $qty * $kp;
            $total = $total + $subtotal;

            echo '
<form name="update_kal" action="kalathi_en.php" method="post">
<tr>
<td>
        <table width="100%" border="0">
        <tr>
        <td width="52%"><div align="left">'.<a href="\''.$kurl.'"><p>$kb</p></a>'.</div></td>
        <td width="12%"><div align="left">'.<b>$kp.</b> €</div></td>
        <td width="19%"><div align="center"><input type="text" size="2"
name="qu['.$row3['kal_id'].']" value=".$qty.'" /></div></td>
        <td width="17%"><div align="center"><div id="but_delete">'.<a
href="\''kalathi_delete.php?bid={$row3['book_id']}\''>'.'.</a>'.</div></div></td>
        </tr>
        </table></td></tr>
        ';
    }
    echo '
<tr>
<td height="23">-----
-----</td></tr>
<tr>
<td>
<table width="100%" border="0">
<tr>
<td><p align="left"><input type="submit" name="update" value="Ενημέρωση Καλαθιού" /></p>
<input type="hidden" name="updated" value="TRUE" /></td>
<td align="right">
<table width="135" border="0">
<tr>
<td width="47%"><b>Σύνολο:</b></td>
<td width="53%"><p>'.<b>$total.</b> €</p></td>
</tr>
</table></div></td></tr>
<tr>
<td><div align="left"><div id="but_agora">'.<p><a href="\''order.php?t=$total\''>'.'.</a></p>'.</div></td>

```

```

<td><div align="right"><div id="but_paraggelia">'.<p><a href="paraggelies.php?t=$total\ ">'.'.</a></p>'.</div></td>
</tr>
</table></td></tr>
</form>';

}}
?>

```

Περιγραφή Σεναρίου

Ξεκινώντας το σενάριο επιλέγουμε τα στοιχεία του βιβλίου από τον πίνακα books. Έπειτα ελέγχουμε αν το βιβλίο υπάρχει ήδη στον καλάθι. Αν ναι αυξάνουμε την ποσότητα κατά ένα, αλλιώς την κάνουμε ίση με τη μονάδα και εισάγουμε τα στοιχεία του βιβλίου στον πίνακα kalathi της βάσης δεδομένων. Στη συνέχεια επιλέγουμε τα βιβλία που ο συγκεκριμένος χρήστης έχει καταχωρίσει στον πίνακα kalathi και τα εμφανίζουμε μέσω ενός βρόχου while.

Επίσης δημιουργήθηκε μια φόρμα η οποία έχει ένα πεδίο κειμένου. Εκεί ο χρήστης καταχωρεί την ποσότητα των βιβλίων που επιθυμεί να παραγγείλει.

```
<input type="text" size="2" name="qu['.$row3['kal_id'].']" value=".$qty." />
```

Η τιμή της ποσότητας του κάθε βιβλίου αποθηκεύεται στον πίνακα qu[]. Όταν ο χρήστης αλλάζει την ποσότητα κάποιου βιβλίου πρέπει να πατήσει το κουμπί της φόρμας “Ενημέρωση καλάθιού” έτσι ώστε να ενημερωθεί η βάση δεδομένων με τις νέες τιμές.

Για την διαγραφή ενός βιβλίου δημιουργήθηκε μια εικόνα που είναι σύνδεσμος προς την σελίδα kalathi_delete.php. Με τον ίδιο τρόπο ο χρήστης μεταβαίνει στην αγορά των βιβλίων (order.php) και στην εμφάνιση των παραγγελιών του (paraggelies.php).

Κατηγορίες Βιβλίων		Περιεχόμενα καλάθιού		
Εκπαίδευση Λογοτεχνία Ψυχολογία Θρησκεία Φιλοσοφία Ιστορία, Γεωγραφία Μαγειρική	Τίτλος	Τιμή	Ποσότητα	Αφαίρεση
	Αισθήσεις και ψευδαισθήσεις της προόδου	20.19 €	3	
	Casta Diva	17.70 €	1	
	Της ζωής και της αγάπης	20.28 €	3	
	Το τελευταίο τσιγάρο	20.18 €	1	
Ενημέρωση Καλάθιού			Σύνολο: 159.29 €	
Ολοκλήρωση Αγοράς			Οι παραγγελίες σας	

Εικόνα 4.63: Εμφάνιση βιβλίων στο καλάθι αγορών

Σενάριο PHP (kalathi_agorwn.php)

```
<?php
```

```
$q = "SELECT * FROM kalathi WHERE user_id = {$_SESSION['user_id']} ORDER BY date";
$r = mysqli_query($dbc, $q);
if (mysqli_num_rows($r) < 1)
{
    echo '<tr>
    <td>Το καλάθι σας είναι άδειο.</td>
    </tr>
    <tr>
    <td><div align="right"><div id="but_paraggelia">' . '<p><a href="' . "paraggelies.php" . '>' . '' . '</a></p>' . '</div></td>
    </tr>';
}
else{
    $t = $kurl = $kid = $kb = $kp = $qty = $subtotal = FALSE;
    $qu = array();
    $total = 0;
    while ($row = mysqli_fetch_array ($r, MYSQLI_ASSOC))
    {
        $kurl = $row['k_url'];
        $kb = $row['kal_book'];
        $kp = $row['pr'];
        $qty = $row['quantity'];
        $subtotal = $qty * $kp;
        $total = $total + $subtotal;
        echo '
<form name="update_kal" action="kalathi_en.php" method="post">
<tr>
<td>
    <table width="100%" border="0">
    <tr>
    <td width="52%"><div align="left">' . '<a href="' . $kurl . '"><p>$kb</p></a>' . '</div></td>
    <td width="12%"><div align="left">' . '$kp.' €</div></td>
    <td width="19%"><div align="center"><input type="text" size="2"
name="qu['.$row3['kal_id'].']" value="" . $qty.' /></div></td>
    <td width="17%"><div align="center"><div id="but_delete">' . '<a
href="kalathi_delete.php?bid={ $row3[\'book_id\']}">' . '' . '</a>' . '</div></div></td>
    </tr>
    </table></td></tr>
    ';
}
echo '
<tr>
<td height="23">-----
-----</td></tr>
<tr>
<td>
<table width="100%" border="0">
```

```

<tr>
<td><p align="left"><input type="submit" name="update" value="Ενημέρωση Καλαθιού" /></p>
<input type="hidden" name="updated" value="TRUE" /></td>
<td align="right">
<table width="135" border="0">
<tr>
<td width="47%">Σύνολο:</td>
<td width="53%"><p>'.$total. €</p></td>
</tr>
</table></div></td></tr>
<tr>
<td align="left"><div id="but_agora">'.<p><a href="order.php?t=$total">'.'.</a></p>'.</div></td>
<td align="right"><div id="but_paraggelia">'.<p><a href="paraggelies.php?t=$total">'.'.</a></p>'.</div></td>
</tr>
</table></td></tr>
</form>;
}
?>

```

Περιγραφή Σεναρίου

Η εμφάνιση των αποτελεσμάτων ακολουθεί την ίδια διαδικασία με το σενάριο της σελίδας kalathi_add.php. Αυτό που προστίθεται είναι ο έλεγχος για την ύπαρξη καταχωρίσεων στο καλάθι. Αν όχι τότε εμφανίζεται ένα κατάλληλο μήνυμα μαζί με τον σύνδεσμο των παραγγελιών, αλλιώς το σενάριο συνεχίζει με την εκτύπωση των περιεχομένων του πίνακα kalathi.

Σενάριο PHP (kalathi_en.php)

```

<?php

if (isset($_POST['updated']))
{
    foreach ($_POST['qu'] as $k => $value)
    {
        $skid = (int) $k;
        $qty = (int) $value;

        if ($qty == 0){
            $q2 = "DELETE FROM kalathi WHERE kal_id = '$skid'";
            $r2 = mysqli_query($dbc, $q2);
        }elseif ($qty >= 1){
            $q2 = "UPDATE kalathi SET quantity = $qty WHERE kal_id = $skid";
            $r2 = mysqli_query($dbc, $q2);
        }
    }
}

$q = "SELECT * FROM kalathi WHERE user_id = {$_SESSION['user_id']} ORDER BY date";
$r = mysqli_query($dbc, $q);

```

```

if (mysqli_num_rows($r) < 1)
{
    echo '<p>Το καλάθι σας είναι άδειο.</p>';
}
else{
    $t = $kurl = $kb = $kp = $qty = $subtotal = FALSE;
    $qu = array();
    $total = 0;
    while ($row = mysqli_fetch_array ($r, MYSQLI_ASSOC))
    {
        $kurl = $row['k_url'];
        $kb = $row['kal_book'];
        $kp = $row['pr'];
        $qty = $row['quantity'];
        $subtotal = $qty * $kp;
        $total = $total + $subtotal;
    }
    echo '
<form name="update_kal" action="kalathi_en.php" method="post">
<tr>
<td>
        <table width="100%" border="0">
        <tr>
        <td width="52%"><div align="left">'.<a href="\''.$kurl.'"><p>$kb</p></a>'.</div></td>
        <td width="12%"><div align="left">'. $kp.' €</div></td>
        <td width="19%"><div align="center"><input type="text" size="2"
name="qu['.$row3['kal_id'].']" value="'. $qty.'" /></div></td>
        <td width="17%"><div align="center"><div id="but_delete">'.<a
href="kalathi_delete.php?bid={$row3['book_id']}">'.'.</a>'.</div></div></td>
        </tr>
        </table></td></tr>
        ';
    }
    echo '
<tr>
<td height="23">-----
-----</td></tr>
<tr>
<td>
<table width="100%" border="0">
<tr>
<td><p align="left"><input type="submit" name="update" value="Ενημέρωση Καλαθιού" /></p>
<input type="hidden" name="updated" value="TRUE" /></td>
<td><div align="right">
<table width="135" border="0">
<tr>
<td width="47%">Σύνολο:</td>
<td width="53%"><p>'. $total.' €</p></td>
</tr>
</table></div></td></tr>
<tr>

```

```

<td><div align="left"><div id="but_agora">'.<p><a href=\|"order.php?t=$total\|">'.'.</a></p>".</div></td>
<td><div align="right"><div id="but_paraggelia">'.<p><a href=\|"paraggelies.php?t=$total\|">'.'.</a></p>".</div></td>
</tr>
</table></td></tr>
</form>;
}
?>

```

Περιγραφή Σεναρίου

Η σελίδα αυτή δημιουργήθηκε για την ανανέωση των ποσοτήτων των βιβλίων που ο χρήστης επιθυμεί να αγοράσει. Αυτό επιτυγχάνεται με έναν βρόχο foreach. Ο βρόχος αυτός παίρνει τις τιμές που πίνακα qu[] (`qu['.$row3['kal_id'].'] = $qty`). Οι κάθε στήλη του πίνακα αντιπροσωπεύει τον κωδικό του βιβλίου και η κάθε αποθηκευμένη τιμή την ποσότητα αυτού.

Κάθε φορά που τρέχει ο βρόχος foreach δίνουμε στην μεταβλητή \$k τον κωδικό του κάθε βιβλίου και στην μεταβλητή \$value την ποσότητα αυτού. Έπειτα μεταβάλλουμε τις τιμές τους σε ακέραιες και τις καταχωρούμε στις μεταβλητές \$kid και \$qty.

```
foreach ($_POST['qu'] as $k => $value)
```

```
$kid = (int) $k;
```

```
$qty = (int) $value;
```

Αν η ποσότητα έχει τιμή μηδέν διαγράφουμε το βιβλίο από τον πίνακα kalathi της βάσης δεδομένων, αλλιώς εκτελούμε το ερώτημα UPDATE το οποίο καταχωρεί τις καινούργια τιμή της ποσότητας στο πίνακα.

Στη συνέχεια ο κώδικας εκτελείται όπως και στα προηγούμενα σενάρια.

Σενάριο PHP (kalathi_delete.php)

```
<?php
```

```
if (isset($_GET['bid']) && is_numeric($_GET['bid']))
```

```
{
```

```
    $bid = (int) $_GET['bid'];
```

```
    $q = "DELETE FROM kalathi WHERE book_id = '$bid' ";
```

```
    $r = mysqli_query($dbc, $q);
```

```
    if ($r)
```

```
    {
```

```
        $t = $kurl = $kb = $kp = $qty = $subtotal = FALSE;
```

```
        $total = 0;
```

```
        $qu = array();
```

```
        while ($row = mysqli_fetch_array ($r, MYSQLI_ASSOC))
```

```
        {
```

```
            $kurl = $row['k_url'];
```

```
            $kb = $row['kal_book'];
```

```
            $kp = $row['pr'];
```

```
            $qty = $row['quantity'];
```

```
            $subtotal = $qty * $kp;
```



```

        $total = $total + $subtotal;
        echo '
<form name="update_kal" action="kalathi_en.php" method="post">
<tr>
<td>
        <table width="100%" border="0">
        <tr>
        <td width="52%"><div align="left">'<a href=\''.$kurl.'"><p>$kb</p></a>'</div></td>
        <td width="12%"><div align="left">'<math>\text{€}</math>'</div></td>
        <td width="19%"><div align="center"><input type="text" size="2"
name="qu['.$row3['kal_id'].']" value=".$qty." /></div></td>
        <td width="17%"><div align="center"><div id="but_delete">'<a
href=\''kalathi_delete.php?bid={$row3['book_id']}\''>''</a>'</div></div></td>
        </tr>
        </table></td></tr>
        ';
    }
    echo '
<tr>
<td height="23">-----
-----</td></tr>
<tr>
<td>
<table width="100%" border="0">
<tr>
<td><p align="left"><input type="submit" name="update" value="Ενημέρωση Καλαθιού" /></p>
<input type="hidden" name="updated" value="TRUE" /></td>
<td align="right">
<table width="135" border="0">
<tr>
<td width="47%"><math>\Sigma</math><math>\nu</math><math>\nu</math><math>\lambda</math><math>:\lt;/math></td>
<td width="53%"><p>'<math>\text{€}</math>'</p></td>
</tr>
</table></div></td></tr>
<tr>
<td align="left"><div id="but_agora">'<p><a href=\''order.php?t=$total\''>''</a></p>'</div></td>
<td align="right"><div id="but_paraggelia">'<p><a href=\''paraggelies.php?t=$total\''>''</a></p>'</div></td>
</tr>
</table></td></tr>
</form>;
}}
?>

```

Περιγραφή Σεναρίου

Μέσω του URL μεταβιβάζουμε τον κωδικό του βιβλίου (όπως και στα προηγούμενα σενάρια) γι' αυτό έχουμε την δυνατότητα να διαγράψουμε το συγκεκριμένο βιβλίο από την βάση δεδομένων και έπειτα να εμφανίσουμε τις απομείναντες καταχωρίσεις του πίνακα.

Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση	
<pre> .bookstore #kalathi { background-color: #000000; color: #000000; font-size: 16px; font-weight: bolder; background-repeat: repeat-y; margin-right: 30px; margin-left: 30px; padding-top: 50px; background-image: url(background%20img.jpg); padding-bottom: 70px; } </pre>	<pre> .bookstore #kalathi a{ color: #000000; font-family: "Times New Roman", Times, serif; font-size: 16px; } </pre>

Πίνακας 4.31: Φύλλα CSS για την μορφοποίηση του καλαθιού αγορών

4.5.12 Ολοκλήρωση Αγοράς

Στο καλάθι του χρήστη υπάρχει ένας σύνδεσμος “Ολοκλήρωση Αγοράς”. Ο σύνδεσμος αυτός μας μεταφέρει στη σελίδα order.php. Επίσης μέσω του URL μεταβιβάζουμε το συνολικό ποσό της παραγγελίας.

Σενάριο PHP (order.php)

```
<?php
```

```

if (isset($_GET['t']))
{
    $total = $_GET['t'];
    $oid = $ord_bid = $ord_book = $ord_pr = $ord_qty = FALSE;
    $q = "INSERT INTO orders (user_id, total) VALUES ('".$_SESSION['user_id'].', '$total)";
    $r = mysqli_query($dbc, $q);
    if (mysqli_affected_rows($dbc) == 1)
    {
        $q2 = "SELECT order_id FROM orders WHERE user_id = {'".$_SESSION['user_id']}";
        $r2 = mysqli_query($dbc, $q2);
        if ($r2)
        {
            $row = mysqli_fetch_array ($r3, MYSQLI_ASSOC);
            $oid = $row['order_id'];

            $q3 = "SELECT * FROM kalathi WHERE user_id = {'".$_SESSION['user_id']}";
            $r3 = mysqli_query($dbc, $q3);
            if ($r3)
            {
                while ($row2 = mysqli_fetch_array ($r3, MYSQLI_ASSOC))
                {
                    $ord_bid = $row2['book_id'];

```

```

        $ord_pr = $row2['pr'];
        $ord_qty = $row2['quantity'];
        $q4 = "INSERT INTO order_contents (order_id, book_id, quantity, price, ship_date)
VALUES ('$oid', '$ord_bid', '$ord_qty', '$ord_pr', NOW() )";
        $r4 = mysqli_query($dbc, $q4);
    }
    if ($r4)
    {
        $q5 = "DELETE FROM kalathi WHERE user_id = {$_SESSION['user_id']} ";
        $r5 = mysqli_query($dbc, $q5);
        if ($r5)
        {
            $q6 = "SELECT * FROM order_contents WHERE order_id = $oid ORDER BY
ship_date";
            $r6 = mysqli_query($dbc, $q6);
            if ($r6)
            {
                while ($row3 = mysqli_fetch_array ($r6, MYSQLI_ASSOC))
                {
                    $ord_bid = $row3['book_id'];
                    $ord_prise = $row3['price'];
                    $ord_qty = $row3['quantity'];
                    $ord_date = $row3['ship_date'];
                    $q7 = "SELECT book_name, url_after FROM books WHERE
book_id = $ord_bid";
                    $r7 = mysqli_query($dbc, $q7);
                    $row4 = mysqli_fetch_array ($r7, MYSQLI_ASSOC);
                    if ($r7)
                    {
                        $book_url = $row4['url_after'];
                        $ord_book = $row4['book_name'];
                    }
                }
            }
        }
    }

    echo '
<tr>
<td>
<table width="100%" border="0">
<tr>
<td width="45%" height="27">
<div align="left">'.<a href="\". $book_url. \"\"><p>$ord_book</p> </a>\".'</div></td>
<td width="13%"><div align="left">'. $ord_prise. '</div></td>
<td width="14%"><div align="center">'. $ord_qty. '</div></td>
<td width="28%"><div align="right">'. $ord_date. '</div></td>
</tr>
</table>
</td>
</tr>
';
}}}}
}}
}
?>

```

Περιγραφή Σεναρίου

Στον πίνακα orders της βάσης δεδομένων αποθηκεύονται ο κωδικός του χρήστη, το συνολικό ποσό και η ημέρα που έγινε η παραγγελία. Στο πίνακα order_contents τα βιβλία που θέλει να αγοράσει ο χρήστης.

Το σενάριο αρχίζει με την προσθήκη των δεδομένων στον πίνακα orders. Αν το ερώτημα εκτελεσθή με επιτυχία επιλέγουμε το πρωτεύων κλειδί του πίνακα orders για να γίνει ξένο κλειδί του πίνακα order_contents. Στη συνέχεια επιλέγουμε τα βιβλία που ο χρήστης είχε καταχωρίσει στο καλάθι του και τα προσθέτουμε στον πίνακα order_contents. Αφού τα δεδομένα μεταφερθούν με επιτυχία προβαίνουμε στην διαγραφή τους από τον καλάθι.

Επιλέγοντας τα δεδομένα του πίνακα order_contents που αντιστοιχούν στον συγκεκριμένο χρήστη τα εμφανίζουμε μέσω ενός βρόχου while και την συνάρτησης echo (όπως και στα προηγούμενα σενάρια).



Κατηγορίες Βιβλίων	Τίτλος	Τιμή	Ποσότητα	Ημερομηνία Παραγγελίας
Εκπαίδευση	Αγαστή συνεργασία	15.21	1	2011-05-25 19:55:46
Λογοτεχνία	Λουλούδι της ερήμου	14.20	2	2011-05-25 19:55:46
Ψυχολογία	Αισθήσεις και ψευδαισθήσεις της προόδου	20.19	3	2011-05-26 02:56:00
Θρησκεία	Casta Diva	17.70	1	2011-05-26 02:56:00
Φιλοσοφία	Της ζωής και της αγάπης	20.28	3	2011-05-26 02:56:00
Ιστορία, Γεωγραφία	Το τελευταίο τσιγάρο	20.18	1	2011-05-26 02:56:00
Μυθολογία	Κλάψε Παναγιά μου	10.90	1	2011-05-26 03:06:04

Εικόνα 4.64: Εμφάνιση παραγγελιών

4.5.13 Παραγγελία του χρήστη

Στη σελίδα αυτή μπορεί ο χρήστης να βλέπει τις παραγγελίες που έχει κάνει έως τώρα (paraggelies.php). Ο κώδικας αυτός είναι ένα κομμάτι του προηγούμενου σεναρίου που αφορούσε την αγορά των βιβλίων, αφού ο τρόπος εμφάνισης των αποτελεσμάτων είναι ο ίδιος.

Σενάριο PHP

```
<?php
```

```
$oid = $ord_bid = $ord_book = $ord_pr = $ord_qty = FALSE;
$q = "SELECT order_id FROM orders WHERE user_id = {$_SESSION['user_id']}";
$r = mysqli_query($dbc, $q);
if ($r)
{
    $row = mysqli_fetch_array ($r, MYSQLI_ASSOC);
    $oid = $row['order_id'];
    $q2 = "SELECT * FROM order_contents WHERE order_id = $oid ORDER BY ship_date";
    $r2 = mysqli_query($dbc, $q2);
    if (mysqli_num_rows($r) < 1)
    {
        echo '<tr>
            <td>Δεν έχετε παραγγελίες.</td>
        </tr>';
    }else{

        while ($row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC))
        {
            $ord_bid = $row2['book_id'];
            $ord_prise = $row2['price'];
            $ord_qty = $row2['quantity'];
            $ord_date = $row2['ship_date'];
            $q3 = "SELECT book_name, url_admin FROM books WHERE book_id =
$ord_bid";

            $r3 = mysqli_query($dbc, $q3);
            $row3 = mysqli_fetch_array ($r3, MYSQLI_ASSOC);
            if ($r3)
            {
                $book_url = $row3['url_admin'];
                $ord_book = $row3['book_name'];

                echo '
<tr>
<td>
<table width="100%" border="0">
tr>
<td width="45%" height="27"><div align="left">'.<a href="\".$book_url.\"\"><p>$ord_book</p>
</a>'.</div></td>
<td width="13%"><div align="left">'. $ord_prise.'</div></td>
<td width="14%"><div align="center">'. $ord_qty.'</div></td>
<td width="28%"><div align="right">'. $ord_date.'</div></td>
</tr>
</table>
</td>
</tr>
';
}}}?>
```

4.5.14 Πίνακας αγορών

Ο πίνακας αγορών αφορά μόνο το διαχειριστή της ιστοσελίδας ο οποίος θα μπορεί να ενημερώνεται για τις παραγγελίες όλων των χρηστών. Επίσης, μπορεί να διαγράψει όποια παραγγελία επιθυμεί (delete_order.php).

Σενάριο PHP (pinakas_agorwn.php)

```
<?php
```

```
$oid = $ord_uid = $ord_total = $ord_date = FALSE;
$qu = array();
$q = "SELECT * FROM orders ORDER BY order_date";
$r = mysqli_query($dbc, $q);
if (mysqli_num_rows($r) < 1)
{
    echo '<p>Δεν υπάρχουν καταχωρημένες αγορές.</p>';
}
else{
    while ($row = mysqli_fetch_array ($r, MYSQLI_ASSOC))
    {
        $oid = $row['order_id'];
        $ord_uid = $row['user_id'];
        $ord_total = $row['total'];
        $ord_date = $row['order_date'];
        $q2 = "SELECT CONCAT(first_name, ' ', last_name) AS un FROM users WHERE user_id =
        $ord_uid";
        $r2 = mysqli_query($dbc, $q2);
        $row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
        $ord_un = $row2['un'];
        if ($r2)
        {
            echo '<tr>
            <td>
            <table width="100%" border="0">
            <tr>
            <td width="40%" height="27"><div align="left">'.<a href="order_contents.php?
            oid={$row['order_id']}\>$ord_un'.</div></td>
            <td width="17%"><div align="center">'. $ord_total. '€</div></td>
            <td width="31%"><div align="center">'. $ord_date. '</div></td>
            <td width="12%"><div align="center"><div id="but_delete">'.<a
            href="delete_order.php?oid={$row['order_id']}\>'.'.</a>'.</div></div></td>
            </tr>
            </table>
            </td>
            </tr>
            ';
        }
    }
?>
```

Περιγραφή Κώδικα

Πρώτα από όλα επιλέγουμε τους καταχωρημένους χρηστών από τον πίνακα orders, δηλαδή αυτούς που έχουν κάνει κάποια παραγγελία. Αν δεν υπάρχουν καταχωρήσεις στον πίνακα εκτυπώνουμε ένα κατάλληλο μήνυμα, αλλιώς εκτελούμε έναν βρόχο while μέσα στον οποίο επιλέγουμε τα ονόματα των χρηστών από τον πίνακα users έτσι ώστε να εκτυπωθούν στην σελίδα μέσω της συνάρτησης echo().

Επίσης, το όνομα του κάθε χρήστη γίνεται σύνδεσμος προς την σελίδα order_contents έτσι ώστε να έχει πρόσβαση ο διαχειριστής στα βιβλία που έχει παραγγείλει ο κάθε χρήστης.

Η συνάρτηση echo() εκτυπώνει τα ονόματα των χρηστών, το συνολικό ποσό και την ημερομηνία της παραγγελίας.



Κατηγορίες Βιβλίων	Χρήστης	Συνολικό Ποσό	Ημερομηνία Παραγγελίας	Αφαίρεση
Εκπαίδευση Λογοτεχνία Ψυχολογία Θρησκεία Φιλοσοφία Ιστορία, Γεωγραφία Μαγειρική	ΠΟΠΗ ΧΩΛΙΔΟΥ	43.61€	2011-05-25 19:55:46	X
	Μάκης Δελιόπουλος	36.52€	2011-05-26 00:49:25	X
	ΠΟΠΗ ΧΩΛΙΔΟΥ	159.29€	2011-05-26 02:56:00	X
	ΠΟΠΗ ΧΩΛΙΔΟΥ	10.90€	2011-05-26 03:06:04	X

Αρχική Προτείνουμε Best Sellers Forum Πίνακας Αγορών

Εικόνα 4.65: Πίνακας αγορών

4.5.15 Αγορές Χρηστών

Όπως είδαμε και στο προηγούμενο σενάριο ο διαχειριστής πατώντας πάνω στο όνομα του χρήστη μπορεί να ενημερώνεται για τα βιβλία που έχει παραγγείλει και μάλιστα ανάλογα με την ημέρα παραγγελίας. Αυτό γίνεται μέσω της σελίδας order_contents.php στην οποία μεταβιβάζουμε τον κωδικό παραγγελίας του κάθε χρήστη (\$oid). Με τον τρόπο αυτό μπορούμε μέσω ενός ερωτήματος SELECT να επιλέξουμε τα βιβλία από τον πίνακα order_contents της βάσης δεδομένων με βάση τον κωδικό παραγγελίας του κάθε χρήστη.

Σενάριο PHP (order_contents.php)

<?php

```
$ord_bid = $ord_book = $ord_prise = $ord_qty = $ord_date = $book_url = FALSE;  
if (isset($_GET['oid']) && is_numeric($_GET['oid']))
```

```

{
    $oid = (int) $_GET['oid'];
    $q = "SELECT * FROM order_contents WHERE order_id = '$oid' ORDER BY ship_date";
    $r = mysqli_query($dbc, $q);
    if ($r)
    {
        while ($row = mysqli_fetch_array ($r, MYSQLI_ASSOC))
        {
            $ord_bid = $row['book_id'];
            $ord_prise = $row['price'];
            $ord_qty = $row['quantity'];
            $ord_date = $row['ship_date'];

            $q2 = "SELECT book_name, url_admin FROM books WHERE book_id = $ord_bid";
            $r2 = mysqli_query($dbc, $q2);
            $row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
            if ($r2)
            {
                $book_url = $row2['url_admin'];
                $ord_book = $row2['book_name'];
                echo '<tr>
                    <td>
<table width="100%" border="0">
<tr>
<td width="38%" height="27"><div align="left">'<a href="">'. $book_url. "'>
<p>$ord_book</p></a>'.</div></td>
<td width="9%"><div align="left">'<b>$.ord_prise.</div></td>
<td width="13%"><div align="center">'<b>$.ord_qty.</div></td>
td width="28%"><div align="center">'<b>$.ord_date.</div></td>
</tr>
</table>
</td>
</tr>
';
}}}}
?>

```

4.5.16 Διαγραφή Παραγγελίας

Το μόνο που χρειάζεται είναι να διαγράψουμε από τους πίνακες orders και order_contents, της βάσης δεδομένων, τις παραγγελίες του χρήστη με βάση τον κωδικό παραγγελίας του (\$oid).

```

$q = "DELETE FROM orders WHERE order_id = '$oid' ";
$q2 = "DELETE FROM order_contents WHERE order_id = '$oid' ";

```

Σενάριο PHP (delete_order.php)


```
<?php
```

```
if (isset($_GET['oid']) && is_numeric($_GET['oid']))
{
    $oid = (int) $_GET['oid'];
    $q = "DELETE FROM orders WHERE order_id = '$oid' ";
    $r = mysqli_query($dbc, $q);
    if ($r)
    {
        $q2 = "DELETE FROM order_contents WHERE order_id = '$oid' ";
        $r2 = mysqli_query($dbc, $q2);
        if ($r2)
        {
            $soid = $sord_uid = $sord_total = $sord_date = FALSE;
            $q3 = "SELECT * FROM orders ORDER BY order_date";
            $r3 = mysqli_query($dbc, $q3);
            if (mysqli_num_rows($r3) < 1)
            {
                echo '<p>Δεν υπάρχουν καταχωρημένες αγορές.</p>';
            }else{
                while ($row = mysqli_fetch_array ($r3, MYSQLI_ASSOC))
                {
                    $sord_uid = $row['user_id'];
                    $sord_total = $row['total'];
                    $sord_date = $row['order_date'];

                    $q4 = "SELECT CONCAT(first_name, ' ', last_name) AS un FROM
users WHERE user_id = $sord_uid";
                    $r4 = mysqli_query($dbc, $q4);
                    $row2 = mysqli_fetch_array ($r4, MYSQLI_ASSOC);
                    $sord_un = $row2['un'];
                    if ($r4)
                    {
                        echo '<tr>
<td>
<table width="100%" border="0">
<tr>
<td width="40%" height="27"><div align="left">'.<a
href="order_contents.php?oid={$row['order_id']}">$sord_un'.</div></td>
<td width="17%"><div align="center">'. $sord_total. '€</div></td>
<td width="31%"><div align="center">'. $sord_date. '</div></td>
<td width="12%"><div align="center"><div id="but_delete">'.<a
href="delete_order.php?oid={$row['order_id']}">'>'.'>'.</a>'.</div></div></td>
</tr>
</table>
</td>
</tr>
';
}}}}]?>
```

4.5.17 Προσθήκη Βιβλίου

Κώδικας Φόρμας

```
<form enctype="multipart/form-data" action="add_book.php" method="post">

<input type="hidden" name="MAX_FILE_SIZE" value="524288" />

<p><span class="style12">Τίτλος Βιβλίου:</span>
<input type="text" name="book_name" size="80" maxlength="80"
value="<?php if (isset($_POST['book_name'])) echo htmlspecialchars($_POST['book_name']); ?>"
/></p>

<p><b class="style12">Μικρή Εικόνα:</b> <input type="file" name="image_small" />
<p><b class="style12">Μεγάλη Εικόνα:</b> <input type="file" name="image_big" />
<p>
```

Επειδή αυτή η φόρμα θα επιτρέπει στους χρήστες να ανεβάζουν αρχεία, στην ετικέτα <form> πρέπει να περιλαμβάνει το χαρακτηριστικό enctype και την κρυφή είσοδο MAX_FILE_SIZE. Το τμήμα enctype δείχνει ότι η φόρμα πρέπει να χειρίζεται διάφορους τύπου δεδομένων. Επίσης, η κρυφή είσοδος MAX_FILE_SIZE είναι ένας περιορισμός της φόρμας για το μέγιστο μέγεθος του επιλεγμένου αρχείου. Η παραμένουσα φόρμα χρησιμοποιεί τη συνάρτηση htmlspecialchars() για να μετατρέψει τους ειδικούς χαρακτήρες σε οντότητες HTML, επειδή δημιουργούν πρόβλημα στο χαρακτηριστικό value των εισόδων. Τέλος, η είσοδος τύπου file δημιουργεί στη φόρμα το κατάλληλο κουμπί. Όταν υποβληθεί η φόρμα το αρχείο που ανέβηκε θα αποθηκευθεί στην υπερκαθολική μεταβλητή \$_FILES. \$_FILES['name'] => Το αρχικό όνομα του αρχείου
\$_FILES['tmp_name'] => Το προσωρινό όνομα του αρχείου στο διακομιστή

Πρώτος Συγγραφέας

Υπάρχων Συγγραφέας

```
<b class="style12">1ος Συγγραφέας:</b>
<p><input type="radio" name="author1" value="existing_author1" <?php if (isset($_POST['author1'])
&& ($_POST['author1'] == 'existing_author1')) echo ' checked="checked"' ; ?> />
```

Υπάρχων Συγγραφέας => <select name = "existing_author1"> <option>Επιλογή Συγγραφέα</option>
<?php

```
$q7= "SELECT author_id, author_fn, author_ln FROM authors ORDER BY author_ln, author_fn";
$r7 = mysqli_query ($dbc, $q7);
```

```
if ($r7)
{
    while ($row5 = mysqli_fetch_array ($r7, MYSQLI_ASSOC))
    {
        $afn = $row5['author_fn'];
        $aln = $row5['author_ln'];
        if ($afn == 'NULL')
        {
            $afn = NULL;
        }
    }
}
```

```

        if (isset($_POST['existing_author1']) && ($_POST['existing_author1'] ==
$row5['author_id'] ) echo ' selected="selected"';
        echo "<option value='{ $row5['author_id'] }'>$aln $afn<option>";
    }
} else {
    echo '<option>Εισαγωγή νέου συγγραφέα.</option>';
} ?>
</select>
</p>]

```

Δημιουργείται ένα ραδιοπλήκτρο για την επιλογή του συγγραφέα από ένα πτυσσόμενο μενού. Με το ερώτημα SELECT επιλέγουμε τα ονόματα των συγγραφέων τα οποία μεταβιβάζουμε στο πτυσσόμενο μενού. Επίσης ελέγχουμε το μικρό όνομα του συγγραφέα ώστε αν έχει τιμή 'NULL' να του δίνουμε «κενή» τιμή. Τα ραδιοπλήκτρα της ίδιας κατηγορίας μπορούν να έχουν το ίδιο όνομα, αφού κάθε φορά επιλέγεται ένα από αυτά. Στη συγκεκριμένη φόρμα έχουν διαφορετικά ονόματα για λόγους ευκολίας εκτέλεσης του σεναρίου.

Εισαγωγή Νέου Συγγραφέα

```

<p><input type="radio" name="author1" value="new_author1" <?php if (isset($_POST['author1']) &&
($_POST['author1'] == 'new_author1') ) echo ' checked="checked"' ; ?> />

```

Νέος Συγγραφέας => </p>

```

<p><b>Όνομα:</b> <input type="text" name="author_fn" size="20" maxlength="20"
value=" <?php if (isset($_POST['author_fn'])) echo $_POST['author_fn']; ?>" />

```

```

<?php echo '    ' ;?>

```

```

<b>Επώνυμο:</b> <input type="text" name="author_ln" size="40" maxlength="40"
value=" <?php if (isset($_POST['author_ln'])) echo $_POST['author_ln']; ?>" />
</p>

```

Εκτός από ένα ραδιοπλήκτρο δημιουργούνται δύο πεδία κειμένου για την εισαγωγή των στοιχείων του νέου συγγραφέα.

Δεύτερος Συγγραφέας

Υπάρχων Συγγραφέας

```

<b class="style12">2ος Συγγραφέας:</b>

```

```

<p><input type="radio" name="author2" value="existing_author2" <?php if (isset($_POST['author2'])
&& ($_POST['author2'] == 'existing_author2') ) echo ' checked="checked"' ; ?> />

```

Υπάρχων Συγγραφέας => <select name="existing_author2"><option>Επιλογή Συγγραφέα</option>

```

<?php

```

```

$q8= "SELECT author_id, author2_fn, author2_ln FROM authors WHERE author2_ln != 'NULL'
ORDER BY author2_ln, author2_fn";

```

```

$r8 = mysqli_query ($dbc, $q8);

```

```

if ($r8)
{
    while ($row6 = mysqli_fetch_array($r8, MYSQLI_ASSOC))
    {
        $afn2 = $row6['author2_fn'];
        $aln2 = $row6['author2_ln'];
        if ($afn2 == 'NULL')
        {
            $afn2 = NULL;
        }
        if (isset($_POST['existing_author2']) && ($_POST['existing_author2'] ==
$row6['author_id'])) echo ' selected="selected"';
        echo "<option value=\"{"$row6['author_id']}\">$aln2 $afn2<option>";
    }
}
else{
    echo '<option>Εισαγωγή νέου συγγραφέα.</option>';
} ?>
</select>
</p>

```

Εισαγωγή Νέου Συγγραφέα

<p><input type="radio" name="author2" value="new_author2" <?php if (isset(\$_POST['author2']) && (\$_POST['author2'] == 'new_author2')) echo ' checked="checked"' ; ?> />

Νέος Συγγραφέας => </p>

<p>**Όνομα:** <input type="text" name="author2_fn" size="20" maxlength="20" value="" <?php if (isset(\$_POST['author2_fn'])) echo \$_POST['author2_fn']; ?> /> <?php echo ' ' ; ?>

Επώνυμο: <input type="text" name="author2_ln" size="40" maxlength="40" value="" <?php if (isset(\$_POST['author2_ln'])) echo \$_POST['author2_ln']; ?> /> </p>

Ανύπαρκτος Συγγραφέας

<p><input type="radio" name="author2" value="none" <?php if (isset(\$_POST['author2']) && (\$_POST['author2'] == 'none')) echo ' checked="checked"' ; ?> />

Κενό

Ο δεύτερος συγγραφέας μπορεί να μην υπάρχει γι' αυτό υπάρχει μια τρίτη επιλογή που δίνει τιμή NULL στο μικτό όνομα και στο επώνυμο του δεύτερου συγγραφέα.

Εκδότης

Υπάρχων Εκδότης

<b class="style12">Εκδότης:

<p><input type="radio" name="editor" value="existing_editor" <?php if (isset(\$_POST['editor']) && (\$_POST['editor'] == 'existing_editor')) echo ' checked="checked"' ; ?> />

Υπάρχων Εκδότης => <select name="existing_editor"><option>Επιλογή Εκδότη</option>


```
?></select></p>
```

Εισαγωγή Νέας Κατηγορίας

```
<p><input type="radio" name="category" value="new_category" <?php if (isset($_POST['category'])  
&& ($_POST['category'] == 'new_category')) echo ' checked="checked"'; ?> />
```

```
Νέα Κατηγορία => Όνομα Κατηγορίας: <input type="text" name="category_name" size="40"  
maxlength="40" value="<?php if (isset($_POST['category_name'])) echo $_POST['category_name']; ?>"  
></p>
```

Τιμή Βιβλίου

```
<p><b class="style12">Τιμή:</b>  
<input type="text" name="price" size="10" maxlength="10" value="<?php if (isset($_POST['price']))  
echo $_POST['price']; ?>" />  
<?php echo ' ' ;?>  
<span class="stele4"><small>(Μην αναγράφετε το σήμα του νομίσματος.)</small></span></p>
```

ISBN

```
<p><b class="style12">ISBN:</b>  
<input type="text" name="ISBN" size="15" maxlength="15" value="<?php if (isset($_POST['ISBN']))  
echo $_POST['ISBN']; ?>" />  
</p>
```

Checkboxes

```
<p><input type = 'checkbox' Name = 'ch1' value = "new" <?php print $ch1; ?> />  
<b>Νέες Κυκλοφορίες</b>  
</p>  
<p><input type = 'checkbox' Name = 'ch2' value = "best" <?php print $ch2; ?> />  
<b>Best Sellers</b>  
</p>  
<p><input type = 'checkbox' Name = 'ch3' value = "sug" <?php print $ch3; ?> />  
<b>Προτεινόμενα Βιβλία</b></p>
```

Περιγραφή

```
<p><b class="style12">Περιγραφή:</b><textarea name="description" cols="80" rows="40">  
<?php if (isset($_POST['description'])) echo $_POST['description']; ?>  
</textarea>  
</p>
```

Button

```
<div align="center"><input type="submit" name="add" value="Προσθήκη" /></div>  
<input type="hidden" name="added" value="TRUE" />
```

```
</form>
```

Σενάριο PHP (add_book.php)

<?php

```
$ch1 = $ch2 = $ch3 = FALSE;
$bn = $p = $isbn = $d = FALSE;
$afn = $aln = $afn2 = $aln2 = $a = $a_id = $a_id2 = FALSE;
$edn = $ed_id = FALSE;
$catn = $cat_id = FALSE;
$temp_bn = $temp_sn = $img_sn = $img_bn = FALSE;

if(isset($_POST['added']))
{
    $errors = array();
    if (empty($_POST['book_name']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε τον τίτλο του βιβλίου';
    }else{
        $bn = trim($_POST['book_name']);

    }
}

/*-----*/
if (is_uploaded_file($_FILES['image_small']['tmp_name']))
{
    $temp_sn = 'C:/xampp/htdocs/site_bookstore/includes/books
images'.md5($_FILES['image_small']['name']);
    if (move_uploaded_file($_FILES['image_small']['tmp_name'], $temp_sn))
    {
        $img_sn = $_FILES['image_small']['name'];
    }else{
        $errors[] = 'Η εικόνα δεν μπόρεσε να μεταφερθεί';
        $temp_sn = $_FILES['image_small']['tmp_name'];
    }
}else{
    $errors[] = 'Ανεπιτυχής ανέβασμα αρχείου';
    $temp_sn = NULL;
}
}
```

Η συνάρτηση `is_uploaded_file()` επιστρέφει τιμή TRUE αν ανέβηκε κάποιο αρχείο και τιμή FALSE αν όχι. Αν ανέβηκε κάποιο αρχείο το σενάριο θα επιχειρήσει να το μεταφέρει στον κατάλογο `books images`. Σε περίπτωση επιτυχίας θα ανατεθεί στην μεταβλητή `$img_sn` το όνομα του αρχείου. Για λόγους ασφαλείας δημιουργείται ένα προσωρινό όνομα για το αρχείο που κρυπτογραφείται κατάλληλα με την συνάρτηση `md5()`.

```
/*-----*/
if (is_uploaded_file($_FILES['image_big']['tmp_name']))
{
    $temp_bn = 'C:/xampp/htdocs/site_bookstore/includes/books
images'.md5($_FILES['image_big']['name']);
    if (move_uploaded_file($_FILES['image_big']['tmp_name'], $temp_bn))
    {

```

```

        $img_bn = $_FILES['image_big']['name'];
    }else{
        $errors[] = 'Η εικόνα δεν μπόρεσε να μεταφερθεί';
        $temp_bn = $_FILES['image_big']['tmp_name'];
    }
}else{
    $errors[] = 'Ανεπιτυχής ανέβασμα αρχείου';
    $temp_bn = NULL;
}

/*-----*/
if (empty($_POST['price']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε την τιμή του βιβλίου';
}else{
    $p = trim($_POST['price']);
}

/*-----*/
if (empty($_POST['ISBN']))
{
    $errors[] = 'Παρακαλούμε συμπληρώστε τον ISBN κωδικό.';
}else{
    $isbn = trim($_POST['ISBN']);
}

/*-----*/

if (empty($_POST['description']))
{
    $d = NULL;
}else{
    $d = trim($_POST['description']);
}

/*-----*/

$ch1 = 'unchecked';
if (isset($_POST['ch1']) && ($_POST['ch1'] == 'new'))
{
    $ch1 = 'checked';
    $new_b = 1;
}else{
    $new_b = 0;
}

$ch2 = 'unchecked';
if (isset($_POST['ch2']) && ($_POST['ch2'] == 'best'))
{
    $ch2 = 'checked';
    $best_b = 1;
}else{
    $best_b = 0;
}

```



```

}

$ch3 = 'unchecked';
if (isset($_POST['ch3']) && ($_POST['ch3'] == 'sug'))
{
    $ch3 = 'checked';
    $sug_b = 1;
}else{
    $sug_b = 0;
}

```

Αν ο διαχειριστής επιλέξει κάποιο από τα checkboxes οι μεταβλητές θα πάρουν την τιμή 1, αλλιώς θα τους εκχωρηθεί μηδενική τιμή. Αναθέτουμε στις μεταβλητές τις συγκεκριμένες τιμές λόγω των τιμών που δέχονται οι τρεις στήλες new_book, best_book και sug_book του πίνακα books, της βάσης δεδομένων

```

/*-----*/
if (isset($_POST['author1']) && ($_POST['author1'] == 'new_author1'))
{

```

Αν ο διαχειριστής επιλέξει να εισάγει το όνομα του πρώτου συγγραφέα στη βάση δεδομένων.

```

if (isset($_POST['author2']) && ($_POST['author2'] == 'new'))
{
    if (empty($_POST['author2_fn']))
    {
        $afn2 = 'NULL';
    }else{
        $afn2 = trim($_POST['author2_fn']);
    }

    if (empty($_POST['author2_ln']))
    {
        $aln2 = 'NULL';
    }else{

        $aln2 = trim($_POST['author2_ln']);
    }
}

```

Αν ο διαχειριστής επιλέξει να εισάγει το όνομα του δεύτερου συγγραφέα στη βάση δεδομένων ελέγχουμε αν έχει δώσει τιμές στα πεδία της φόρμας και έπειτα εκχωρούμε την τιμές στις αντίστοιχες μεταβλητές

```

}elseif (isset($_POST['author2']) && ($_POST['author2'] == 'existing_author2') &&
($_POST['existing_author2'] > 0)){
    $a_id = (int) $_POST['existing_author2'];
    $q = "SELECT author2_fn, author2_ln FROM authors WHERE author_id = $a_id";
    $r = mysqli_query ($dbc, $q);
    $row = mysqli_fetch_array ($r, MYSQLI_ASSOC);
    $afn2 = $row['author2_fn'];

```

```
$aln2 = $row['author2_ln'];
```

Αν ο διαχειριστής επιλέξει τον δεύτερο συγγραφέα από το πτυσσόμενο μενού εκχωρούμε τον κωδικό του συγγραφέα στην μεταβλητή \$a_id. Έπειτα μέσω του κωδικού επιλέγουμε το μικρό όνομα και το επώνυμο του συγκεκριμένου συγγραφέα από την βάση δεδομένων τα οποία αποθηκεύονται στις μεταβλητές \$afn2 και \$aln2.

```
}elseif (isset($_POST['author2']) && ($_POST['author2'] == 'none')){  
    $afn2 = 'NULL';  
    $aln2 = 'NULL'
```

Αν ο διαχειριστής επιλέξει να μην εισάγει δεύτερο συγγραφέα στη βάση δεδομένων δίνουμε στις μεταβλητές \$afn2 και \$aln2 τιμή “NULL”.

```
    }elseif (isset($_POST['author2']) && ($_POST['author2'] == 'none')){  
        $afn2 = 'NULL';  
        $aln2 = 'NULL'  
    }  
  
    if (empty($_POST['author_fn']))  
    {  
        $afn = 'NULL';  
    }elseif (isset($_POST['author_fn']))  
    {  
        $afn = trim($_POST['author_fn']);  
    }  
}
```

Ελέγχουμε αν ο διαχειριστής έχει εισάγει το μικρό όνομα του πρώτου συγγραφέα στο πεδίο της φόρμας. Αν ναι αποθηκεύεται στη μεταβλητή \$afn , αλλιώς η μεταβλητή παίρνει την τιμή “NULL”.

```
if (empty($_POST['author_ln']))  
{  
    $errors[] = 'Παρακαλούμε συμπληρώστε το επώνυμο του συγγραφέα';  
}elseif (isset($_POST['author_ln']))  
{  
    $aln = trim($_POST['author_ln']);  
    $q2 = "INSERT INTO authors (author_fn, author_ln, author2_fn, author2_ln)  
VALUES (?, ?, ?, ?)";  
    $stmt = mysqli_prepare($dbc, $q2);  
    mysqli_stmt_bind_param($stmt, 'ssss', $afn, $aln, $afn2, $aln2);  
    mysqli_stmt_execute($stmt);  
    if ($stmt)  
    {  
        $a = mysqli_insert_id($dbc);  
    }elseif (isset($_POST['author2']) && ($_POST['author2'] == 'none'))  
    {  
        $errors[] = 'Ο νέος συγγραφέας δεν έχει προστεθεί στη βάση δεδομένων.';  
    }  
    mysqli_stmt_close($stmt);  
}
```

Το επώνυμο του πρώτου συγγραφέα δεν μπορεί να έχει τιμή “NULL” γι’ αυτό αν ο διαχειριστής δεν δώσει κάποια τιμή καταχωρείται ένα σχετικό μήνυμα στον πίνακα σφαλμάτων. Σε άλλη περίπτωση εκτελούμε ένα ερώτημα εισαγωγής στοιχείων στη βάση δεδομένων. Η εισαγωγή γίνεται με την χρήση έτοιμων εντολών για λόγους ασφαλείας και καλύτερης απόδοσης. Όταν δεν χρησιμοποιούνται έτοιμες εντολές, ολόκληρο το ερώτημα με, μαζί με την σύνταξη της SQL, και τις συγκεκριμένες τιμές στέλνεται στην MySQL η οποία αναλύει συντακτικά το ερώτημα και το εκτελεί. Με ένα έτοιμο ερώτημα πρώτα στέλνεται η σύνταξη της SQL στη MySQL για να αναλυθεί η ορθότητά του και έπειτα στέλνονται ξεχωριστά οι τιμές. Η MySQL συναρμολογεί το ερώτημα με αυτές τις τιμές και το εκτελεί. Αν το ερώτημα εκτελεστεί σωστά στην μεταβλητή \$a αποθηκεύεται ο κωδικός του συγγραφέα για να προστεθεί στην συνέχεια στον πίνακα books.

```
}elseif (isset($_POST['author1']) && ($_POST['author1'] == 'existing_author1') &&
($_POST['existing_author1'] > 0)){
    $a_id = (int) $_POST['existing_author1'];
    $q ="SELECT author_fn, author_ln FROM authors WHERE author_id = $a_id";
    $r = mysqli_query ($dbc, $q);
    $row = mysqli_fetch_array ($r, MYSQLI_ASSOC);
    $afn = $row['author_fn'];
    $aln = $row['author_ln'];
```

Αν ο διαχειριστής επιλέξει τον πρώτο συγγραφέα από το πτυσσόμενο μενού εκχωρούμε τον κωδικό του συγγραφέα στην μεταβλητή \$a_id. Έπειτα μέσω του κωδικού επιλέγουμε το μικρό όνομα και το επώνυμο του συγκεκριμένου συγγραφέα από την βάση δεδομένων τα οποία αποθηκεύονται στις μεταβλητές \$afn και \$aln.

```
/*-----*/
```

```
if (isset($_POST['author2']) && ($_POST['author2'] == 'new'))
{
    echo 'Εισαγωγή νέου συγγραφέα';
    if (empty($_POST['author2_fn']))
    {
        $afn2 = 'NULL';
    }else{
        $afn2 = trim($_POST['author2_fn']);
    }
    if (empty($_POST['author2_ln']))
    {
        $aln2 = 'NULL';
    }else{
        $aln2 = trim($_POST['author2_ln']);
    }

    $q3 = "INSERT INTO authors (author_fn, author_ln, author2_fn, author2_ln) VALUES
(?, ?, ?, ?)";
    $stmt = mysqli_prepare($dbc, $q3);
    mysqli_stmt_bind_param($stmt, 'ssss', $afn, $aln, $afn2, $aln2);
    mysqli_stmt_execute($stmt);
```

```

if ($stmt)
{
    $a = mysqli_insert_id($dbc);
}else{
    $errors[] = 'Ο νέος συγγραφέας δεν έχει προστεθεί στη βάση δεδομένων.';
}
mysqli_stmt_close($stmt);

```

Αν ο διαχειριστής επιλέξει να εισάγει το όνομα του δεύτερου συγγραφέα στη βάση δεδομένων ελέγχουμε αν έχει δώσει τιμές στα πεδία της φόρμας. Αν ναι οι τιμές αποθηκεύονται στις μεταβλητές \$afn2 και \$aln, αλλιώς οι μεταβλητές παίρνουν την τιμή “NULL”. Έπειτα γίνεται η εισαγωγή των ονομάτων στη βάση δεδομένων μέσω έτοιμων εντολών.

```

}elseif (isset($_POST['author2']) && ($_POST['author2'] == 'existing_author2') &&
($_POST['existing_author2'] > 0)){
    $a_id2 = (int) $_POST['existing_author2'];
    $q2 = "SELECT author2_fn, author2_ln FROM authors WHERE author_id = $a_id2";
    $r2 = mysqli_query ($dbc, $q2);
    $row2 = mysqli_fetch_array ($r2, MYSQLI_ASSOC);
    $afn2 = $row2['author2_fn'];
    $aln2 = $row2['author2_ln'];

    $q3 = "INSERT INTO authors (author_fn, author_ln, author2_fn, author2_ln) VALUES
(?, ?, ?, ?)";
    $stmt = mysqli_prepare($dbc, $q3);
    mysqli_stmt_bind_param($stmt, 'ssss', $afn, $aln, $afn2, $aln2);
    mysqli_stmt_execute($stmt);

    if ($stmt)
    {
        $a = mysqli_insert_id($dbc);
    }else{
        $errors[] = 'Ο νέος συγγραφέας δεν έχει προστεθεί στη βάση δεδομένων.';
    }
    mysqli_stmt_close($stmt);

```

Αν ο διαχειριστής επιλέξει τον δεύτερο συγγραφέα από το πτυσσόμενο μενού εκχωρούμε τον κωδικό του συγγραφέα στην μεταβλητή \$a_id2. Έπειτα μέσω του κωδικού επιλέγουμε το μικρό όνομα και το επώνυμο του συγκεκριμένου συγγραφέα από την βάση δεδομένων τα οποία αποθηκεύονται στις μεταβλητές \$afn2 και \$aln2. Το ερώτημα εισαγωγής εκτελείται με τον ίδιο τρόπο αποθηκεύοντας τα στοιχεία του συγγραφέα στη βάση δεδομένων.

```

}elseif (isset($_POST['author2']) && ($_POST['author2'] == 'none')){
    $afn2 = 'NULL';
    $aln2 = 'NULL';

    $q = "SELECT author_id FROM authors WHERE author_ln = '$aln' AND author_fn =
'$afn'";

```

```

$r = mysqli_query ($dbc, $q);
$row2 = mysqli_fetch_array($r, MYSQLI_ASSOC);
$a = $row2['author_id'];

```

Αν ο διαχειριστής επιλέξει να μην εισάγει δεύτερο συγγραφέα στη βάση δεδομένων δίνουμε στις μεταβλητές \$afn2 και \$aln2 τιμή “NULL”. Έπειτα ανακτάμε από την βάση δεδομένων τον κωδικό του πρώτου συγγραφέα που επέλεξε ο διαχειριστής από το πτυσσόμενο μενού.

```

}else{
    $errors[] = 'Παρακαλούμε εισάγετε ή επιλέξτε τον 2ο συγγραφέα.';
}

}else{
    $errors[] = 'Παρακαλούμε εισάγετε ή επιλέξτε τον 1ο συγγραφέα.';
}

/*-----*/
if (isset($_POST['editor']) && ($_POST['editor'] == 'new_editor'))
{
    if (empty($_POST['editor_name']))
    {
        $errors[] = 'Παρακαλούμε συμπληρώστε το όνομα του εκδότη';
    }else{
        $edn = trim($_POST['editor_name']);

        $q4 = "INSERT INTO editors (editor_name) VALUES (?)";
        $stmt2 = mysqli_prepare($dbc, $q4);
        mysqli_stmt_bind_param($stmt2, 's', $edn);
        mysqli_stmt_execute($stmt2);

        if ($stmt2)
        {
            $ed_id = mysqli_insert_id($dbc);
        }else{
            $errors[] = 'Ο νέος εκδότης δεν έχει προστεθεί στη βάση δεδομένων.';
        }
        mysqli_stmt_close($stmt2);
    }
}
}elseif (isset($_POST['editor']) && ($_POST['editor'] == 'existing_editor') &&
($_POST['existing_editor'] > 0)){
    $ed_id = (int) $_POST['existing_editor'];
}
}

```

Αν ο διαχειριστής επιλέξει να εισάγει το όνομα του εκδότη στη βάση δεδομένων ελέγχουμε αν έχει εκχωρήσει τιμή στο πεδίο της φόρμας . Αν ναι γίνεται η είσοδος των δεδομένων στη βάση μέσω του

ερωτήματος INSERT. Αν ο διαχειριστής επιλέξει τον εκδότη από το πτυσσόμενο μενού εκχωρούμε τον κωδικό του εκδότη στην μεταβλητή \$ed_id.

```
/*-----*/
    if (isset($_POST['category']) && ($_POST['category'] == 'new_category'))
    {
        if (empty($_POST['category_name']))
        {
            $errors[] = 'Παρακαλούμε συμπληρώστε το όνομα της κατηγορίας.';
        }else{
            $catn = trim($_POST['category_name']);

            $q5 = "INSERT INTO categories (category_name) VALUES (?)";
            $stmt3 = mysqli_prepare($dbc, $q5);
            mysqli_stmt_bind_param($stmt3, 's', $catn);
            mysqli_stmt_execute($stmt3);

            if ($stmt3)
            {
                $cat_id = mysqli_insert_id($dbc);
            }else{
                $errors[] = 'Ο νέα κατηγορία δεν έχει προστεθεί στη βάση δεδομένων.';
            }
            mysqli_stmt_close($stmt3);
        }
    }elseif (isset($_POST['category']) && ($_POST['category'] == 'existing_category') &&
($_POST['existing_category'] > 0)){
        $cat_id = (int) $_POST['existing_category'];
    }else{
        $errors[] = 'Παρακαλούμε εισάγετε ή επιλέξτε την κατηγορία.';
    }
}

/*-----*/

if (empty($errors))
{
    $q6 = "INSERT INTO books (author_id, editor_id, category_id, book_name, price, ISBN,
description, image_name_small, image_name_big, new_book, best_book, sug_book) VALUES
('$a', '$ed_id', '$cat_id', '$bn', '$p', '$isbn', '$d', '$img_sn', '$img_bn', '$new_b', '$best_b',
'$sug_b')";
    $r6 = mysqli_query($dbc, $q6);
    if (mysqli_affected_rows($dbc) == 1)
    {
        echo '<p>Το βιβλίο έχει καταχωρηθεί.</p>';

        rename ($temp_sn, "C:/xampp/htdocs/site_bookstore/includes/books images/$img_sn");
        rename ($temp_bn, "C:/xampp/htdocs/site_bookstore/includes/books images/$img_bn");
        $_POST = array();
    }else{
        echo '<p>Σφάλμα!</p>';
    }
}
```

```
}
```

Αν το μητρώο \$errors είναι κενό, τότε όλοι οι έλεγχοι επικύρωσης ήταν επιτυχείς και το βιβλίο μπορεί να προστεθεί στην βάση δεδομένων. Επίσης, πρέπει να μετονομαστούν οι εικόνες που βρίσκονται στον φάκελο books images, αλλά με προσωρινό όνομα. Τέλος καθαρίζεται το μητρώο \$_POST ώστε οι τιμές του να εμφανιστούν στην παραμένουσα φόρμα.

```
/*-----*/
    if (isset($temp_sn) && file_exists($temp_sn) && is_file($temp_sn))
    {
        if (isset($temp_bn) && file_exists($temp_sn) && is_file($temp_bn))
        {
            unlink ($temp_sn);
            unlink ($temp_bn);
        }
    }
}
} //end added
```

Διαγράφουμε τα αρχεία που βρίσκονται στο διακομιστή, επειδή δεν έχουν αποκτήσει μόνιμη όνομα και θέση.

```
if (!empty($errors))
{
    foreach ($errors as $msg)
    {
        echo "$msg<br />\n";
    }
}
?>
```

| Φύλλα CSS που δημιουργήθηκαν για την μορφοποίηση | |
|---|---|
| <pre>.bookstore #add_book { background-color: #F5F1CE; color: #000000; font-size: 16px; background-repeat: repeat-y; margin-left: 30px; padding-top: 50px; background-image: url(background%20img.jpg); padding-bottom: 70px; }</pre> | <pre>.bookstore #add_book a{ color: #F5F1CE; font-family: "Times New Roman", Times, serif; font-size: 16px; }</pre> |

Πίνακας 4.32: Φύλλα CSS για την μορφοποίηση της φόρμας προσθήκης βιβλίου

4.6 Ασφάλεια Ιστοχώρου

Στις τοποθεσίες ηλεκτρονικού εμπορίου υπάρχουν τέσσερα ζητήματα ασφαλείας. Το πρώτο αφορά στον τρόπο αποθήκευσης των δεδομένων στο διακομιστή. Πρέπει να προστατεύεται η ίδια η βάση δεδομένων καθορίζοντας τα ανάλογα δικαιώματα πρόσβασης και τον κατάλογο όπου αποθηκεύονται οι πληροφορίες της περιόδου εργασίας. Η χρήση του μη κοινόχρηστου υπολογιστή υπηρεσίας βελτιώνει την ασφάλεια της τοποθεσίας.

Το δεύτερο ζήτημα ασφαλείας αφορά την προστασία της πρόσβασης σε ευαίσθητες πληροφορίες. Ο διαχειριστής της τοποθεσίας, που έχει την δυνατότητα να βλέπει τις εγγραφές πελατών, πρέπει να προστατεύεται στο υψηλότερο επίπεδο. Αυτό σημαίνει την απαίτηση πιστοποίησης ταυτότητας για την πρόσβαση, τον περιορισμό αυτών που θα γνωρίζουν τις πληροφορίες πρόσβασης, την χρήση ασφαλών συνδέσεων κ.α.

Το τρίτο ζήτημα αφορά την προστασία των δεδομένων κατά την μετάδοση. Πρέπει να χρησιμοποιούνται ασφαλείς συναλλαγές όπου χρειάζονται αριθμοί πιστωτικών καρτών. Αυτό συνεπάγεται την εγκατάσταση Επιπέδου ασφαλών υποδοχών (Security Sockets Layer – SSL) στο διακομιστή με ένα έγκυρο πιστοποιητικό και, στη συνέχεια, τη χρήση του προτύπου <https://> στις διευθύνσεις URL. Επίσης, χρειάζεται ο έλεγχος των πληροφοριών που διακινούνται μέσω ηλεκτρονικού ταχυδρομείου, επειδή τα μηνύματα αυτά δεν διακινούνται έσω ασφαλών διαδρομών.

Το τέταρτο ζήτημα αφορά τον χειρισμό των στοιχείων πληρωμής. Σε καμία περίπτωση δεν πρέπει να διατηρούνται τα στοιχεία αυτά στην τοποθεσία. Η ιδανική λύση είναι η ανάθεση των πληρωμών σε τρίτο πρόσωπο.

4.7 Διαδικασία Πληρωμής

Η διαδικασία ολοκλήρωσης της παραγγελίας αποτελείται από τρία βήματα:

- Επιβεβαίωση της παραγγελίας
- Επιβεβαίωση/υποβολή των στοιχείων χρέωσης και αποστολής
- Επεξεργασία των στοιχείων χρέωσης.

Τα δύο πρώτα βήματα είναι εύκολα στην υλοποίησή τους. Τα περισσότερα δεδομένα που χρειάζονται για το δεύτερο βήμα είναι αποθηκευμένα στο πίνακα users μετά από την εγγραφή του χρήστη. Το τρίτο βήμα είναι πιο δύσκολο. Οι λεπτομέρειες ποικίλουν, ανάλογα με τον τρόπο χειρισμού της χρέωσης και ποιος την κάνει. Ακόμη πιο πολύπλοκα γίνονται τα πράγματα λόγω των διαφορών στη νομοθεσία αν το προς πώληση προϊόν πρόκειται να σταλεί αργότερα ή να παραδοθεί αμέσως

(όπως ένα αρχείο που ο χρήστης μπορεί να κατεβάσει).

Οι περισσότερες τοποθεσίες ηλεκτρονικού εμπορίου αναθέτουν τις χρηματικές συναλλαγές σε κάποιο τρίτο μέρος. Αυτό περιλαμβάνει την αποστολή των στοιχείων χρέωσης, την συνολική αξία της παραγγελίας και ενός αριθμού καταστήματος (που δείχνει την ίδια την τοποθεσία ηλεκτρονικού εμπορίου) σε κάποια άλλη τοποθεσία. Αυτή η τοποθεσία θα χειρίζεται τη διαδικασία χρέωσης του πελάτη και πίστωσης του καταστήματος. Το μέρος που αναλαμβάνει τον χειρισμό της παραγγελίας παρέχει στον κατασκευαστή της τοποθεσίας ηλεκτρονικού εμπορίου τον απαραίτητο κωδικό και οδηγίες για την αλληλεπίδραση με το δικό του σύστημα.

ΚΕΦΑΛΑΙΟ 5: ΑΠΟΤΕΛΕΣΜΑΤΑ

5.1 Συμπεράσματα

Για να είναι οι ιστοσελίδες του Παγκόσμιου Ιστού πιο ελκυστικές πρέπει η διαμόρφωσή τους να περιέχει διάφορα γραφικά, να ενημερώνονται συχνά και να προσαρμόζονται στις ανάγκες των χρηστών. Για τους χρήστες οι ιστοσελίδες μοιάζουν σαν κοινότητες που επισκέπτονται συχνά. Όσο για τους διαχειριστές πρέπει να ενημερώνουν και να συντηρούν συχνά τις τοποθεσίες για να διατηρήσουν το ενδιαφέρον των χρηστών. Για τους λόγους αυτούς, αλλά και άλλους, η PHP και MySQL έχουν γίνει πρότυπα για την δημιουργία δυναμικών τοποθεσιών. Ο βαθμός ενοποίησής τους είναι ένας ακόμα λόγος που οι σχεδιαστές ιστοσελίδων τις προτιμούν.

Με την χρήση της PHP ο διακομιστής μπορεί να παρακολουθεί τις διεργασίες που εκτελεί ο χρήστης μέσω δύο εργαλείων που διαθέτει η γλώσσα αυτή: τα μπισκότα (cookies) και τις περιόδους εργασίας (sessions). Μια επιπλέον δυνατότητα που μας προσφέρει η PHP είναι ο χειρισμός των σφαλμάτων. Ο χειρισμός των σφαλμάτων πρέπει να υπάρχει σε μια ιστοσελίδα, αλλά είναι ουτοπικό να νομίζουμε ότι δεν θα γίνουν λάθη. Υπάρχουν όμως τεχνικές εντοπισμού, χειρισμού και εξάλειψης των σφαλμάτων οι οποίες σίγουρα τα ελαχιστοποιούν. Μια από αυτές τις τεχνική χειρισμού των σφαλμάτων είναι οι κανονικές παραστάσεις Perl με τις οποίες εισάγουμε την ταυτοποίηση των μοτίβων στοκ κώδικά μας. Πρώτα γράφουμε το μοτίβο και έπειτα το εφαρμόζουμε σε μια τιμή (συμβολοσειρά).

Όσον αφορά τις δυναμικές τοποθεσίες προτιμούνται από τις στατικές για το λόγο ότι συντηρούνται πιο εύκολα, μπορούν να αλληλεπιδρούν με το χρήστη, και το περιεχόμενό τους τροποποιείται ανάλογα τις συνθήκες.

Επίσης, η ασφάλεια των εφαρμογών είναι ένα πολύ μεγάλο κεφάλαιο, αλλά αυτό που πρέπει πρώτα να κατανοήσουμε είναι ότι δεν υπάρχουν μόνο δύο καταστάσεις – ασφαλής και μη ασφαλής. Όταν προγραμματίζουμε πρέπει να σκεφτόμαστε τι μεθόδους θα χρησιμοποιήσουμε για να γίνει η τοποθεσία *περισσότερο* ασφαλής. Για να αυξήσει ο διακομιστής την ασφάλεια της ιστοσελίδας χρειάζεται περισσότερος κώδικας, μεγαλύτερος έλεγχος και περισσότερες απαιτήσεις από τους χρήστες.

Τέλος, για το ηλεκτρονικό εμπόριο ο τρόπος διαχείρισης των χρημάτων εξαρτάται πάντα από την συγκεκριμένη τοποθεσία. Γι' αυτό και η ιστοσελίδα που δημιουργήθηκε επικεντρώνεται στις βασικές λειτουργίες μιας τοποθεσίας ηλεκτρονικού εμπορίου (σχεδίαση βάσης δεδομένων, συμπλήρωση του καταλόγου προϊόντων από τον διαχειριστή, παρουσίαση των προϊόντων στους χρήστες, δημιουργία καλαθιού αγοράς και αποθήκευση των παραγγελιών στη βάση δεδομένων).

5.2 Μελλοντική Εργασία και Επεκτάσεις

Στην συγκεκριμένη ιστοσελίδα υλοποιήθηκαν κάποιες βασικές λειτουργίες που προσφέρονται σε μια τοποθεσία ηλεκτρονικού εμπορίου (όσο αφορά το βιβλίο). Ωστόσο μπορούν να γίνουν πολλές επεκτάσεις πάνω σε αυτή την έτσι ώστε να αντιπροσωπεύει μια ολοκληρωμένη διαδικτυακή ιστοσελίδα. Οι υπηρεσίες που θα μπορούσε να προσφέρει επιπλέον είναι:

- Πολυγλωσσική Εμφάνιση της ιστοσελίδας: Με την διεθνοποίηση του ιστοχώρου μπορεί να γίνει προσβάσιμο και από χρήστες που δεν μιλάνε την ίδια γλώσσα.

- Προτάσεις για παρόμοια βιβλία: Όταν ο χρήστης επισκέπτεται την σελίδα ενός βιβλίου θα μπορούσε ο διαχειριστής να του προτείνει παρόμοια βιβλία που ίσως να τον ενδιέφερε να διαβάσει.
- Δημιουργία ιστορικού του συγγραφέα: Ο χρήστης θα μπορούσε να έχει πρόσβαση στην βιογραφία του εκάστοτε συγγραφέα έτσι ώστε να ενημερωνόταν και για άλλα βιβλία του έχει γράψει.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Χρήστος Γιουρτίδης, “Οδηγός Επιτυχίας για το δίπλωμα ECDL 4”. Κλειδάριθμος, 2^η Έκδοση, 2005.
2. Γκιμπερίτης Βαγγέλης, “INTERNET ΟΔΗΓΟΣ ΓΙΑ ΟΛΟΥΣ”. Α. ΤΖΙΟΛΑ Ε., 3^η Έκδοση, 1999.
3. Noel Estabrook. “Μάθετε το Internet σε 24 ώρες”. Μ. Γκιούρδας, 1^η Αμερικανική Έκδοση, 1997.
4. Wikipedia, the free encyclopedia (en.wikipedia.org και el.wikipedia.org).
5. MySQL, the world’s most popular open source database (www.mysql.com).
6. PHP, Hypertext Preprocessor (www.php.net).
7. LARRY ULLMAN, “ΕΙΣΑΓΩΓΗ ΣΤΙΣ PHP 6 & MYSQL 5 ΜΕ ΕΙΚΟΝΕΣ”. Κλειδάριθμος, 3^η Έκδοση, 2009.
8. ΚΟΥΙΜΤΖΟΓΛΟΥ ΛΑΜΠΡΟΣ, “ΔΥΝΑΜΙΚΗ ΕΚΜΑΘΗΣΗ DREAMWEAVER CS3”. Γκιούρδας Β., Έκδοση με DVD, 2008.
9. Meloni Julie, “Μάθετε PHP, MYSQL και Apache”. Μ. Γκιούρδας, 4^η Έκδοση, 2008.
10. Ned Snell, Bob Temple, T. Michael Clark, “Μάθετε το Internet και Web”. Μ. Γκιούρδας, Σειρά: Όλα σε Ένα, 2004.

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ Α

Οδηγίες Εγκατάστασης

ΠΑΡΑΡΤΗΜΑ Β

DreamWeaver CS3

MYSQL

Προγράμματα σχεδίασης βάσης δεδομένων

Σύνδεση στη MySQL

Τοπικός Server Apache

Photoshop CS3

Flash CS3

phpMyAdmin