

**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΣΕΡΡΩΝ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΚΑΤΑΣΤΗΜΑΤΟΣ
ΦΩΤΟΓΡΑΦΙΚΩΝ ΕΙΔΩΝ ΜΕ ΔΥΝΑΤΟΤΗΤΑ ONLINE
ΕΚΤΥΠΩΣΗΣ**

**Πτυχιακή Εργασία της
Ζωγράφου Αγγελικής (1858)**

Επιβλέπων: Λαζαρίδου Μαρία, εργαστηριακός συνεργάτης



ΣΕΡΡΕΣ, ΙΟΥΝΙΟΣ 2011

➤ Περίληψη

Η πτυχιακή εργασία που παρουσιάζεται έχει ως θέμα την δημιουργία ενός ηλεκτρονικού καταστήματος φωτογραφικών ειδών, δηλαδή τη δημιουργία μιας ιστοσελίδας όπου θα παρουσιάζονται τα προϊόντα που διαθέτει το κατάστημα και θα δίνεται η δυνατότητα στο χρήστη να επιλέξει αυτά που τον ενδιαφέρουν και στη συνέχεια να τα αγοράσει. Επίσης το ηλεκτρονικό κατάστημα θα έχει τη δυνατότητα να δέχεται στο server του τις φωτογραφίες που επιθυμεί να εκτυπώσει ένας πελάτης. Έτσι οποιοσδήποτε άνθρωπος ενδιαφέρεται να εκτυπώσει τις φωτογραφίες του δεν χρειάζεται να πάει στο κατάστημα αλλά αρκεί να δει το ηλεκτρονικό κατάστημα όποια στιγμή θέλει στο χώρο που βρίσκεται.



➤ Περιεχόμενα

Περίληψη.....	Σελ.02
Εισαγωγή	Σελ.06
Βιβλιογραφική έρευνα.....	Σελ.13
1. Δημιουργία Βάσης δεδομένων.....	Σελ.14
2. Αρχική σελίδα.....	Σελ.19
2.1 Μορφοποίηση σελίδων με τη βοήθεια CSS.....	Σελ.21
3. Ταυτοποίηση χρήστη στο σύστημα.....	Σελ.23
3.1 Εγγραφή νέου χρήστη.....	Σελ.23
3.2 Σύνδεση χρήστη στο σύστημα.....	Σελ.32
3.3 Αποσύνδεση χρήστη από το σύστημα.....	Σελ.35
4. Περιήγηση στις κατηγορίες του eshop.....	Σελ.36
5. Προβολή προϊόντος.....	Σελ.39
6. Καλάθι αγορών.....	Σελ.49
6.1 Προσθήκη στο καλάθι.....	Σελ.49
6.2 Διαγραφή από το καλάθι.....	Σελ.55
6.3 Αλλαγή ποσότητας στο καλάθι.....	Σελ.57
7. Αγορά προϊόντων.....	Σελ.60
7.1 Από ταυτοποιημένο χρήστη.....	Σελ.60
7.1.1 Report αγοράς.....	Σελ.60
7.2 Από μη ταυτοποιημένο χρήστη.....	Σελ.65
8. Εκτύπωση φωτογραφιών.....	Σελ.66
8.1 Report εκτύπωσης.....	Σελ.67
9. Επικοινωνία με το κατάστημα.....	Σελ.77
10. Σελίδα διαχειριστή.....	Σελ.79
10.1 Πελατολόγιο.....	Σελ.79
10.1.1 Αλλαγή στοιχείων.....	Σελ.81
10.1.2 Διαγραφή πελάτη.....	Σελ.86
10.2 Προϊόντα.....	Σελ.86
10.2.1 Καταχώρηση νέου προϊόντος.....	Σελ.86
10.2.2 Αλλαγή στοιχείων προϊόντος.....	Σελ.89
10.2.3 Διαγραφή προϊόντος.....	Σελ.94
10.3 Κατηγορίες.....	Σελ.95
10.3.1 Καταχώρηση νέας κατηγορίας.....	Σελ.96
10.3.2 Αλλαγή ονόματος κατηγορίας.....	Σελ.97
10.3.3 Διαγραφή κατηγορίας.....	Σελ.99

11. Συμπεράσματα.....	Σελ.100
11.1 Αποτέλεσμα εργασίας.....	Σελ.100
11.2 Προβλήματα που αντιμετωπίστηκαν.....	Σελ.100
11.3 Γνώσεις που αποκτήθηκαν.....	Σελ.101
11.4 Μελλοντικές επεκτάσεις.....	Σελ.102

Βιβλιογραφία.....	Σελ.103
-------------------	---------

➤ **Ευρετήριο εικόνων**

Εικόνα.1 Πίνακες στη βάση δεδομένων.....	Σελ.16
Εικόνα 1.2. Σχεσιακό μοντέλο βάσης δεδομένων.....	Σελ.18
Εικόνα 2. Αρχική σελίδα.....	Σελ.19
Εικόνα 3. Φόρμα login.....	Σελ.23
Εικόνα 3.1.1 Φόρμα εγγραφής.....	Σελ.23
Εικόνα 3.1.2. Επιτυχής εγγραφή.....	Σελ.24
Εικόνα 3.1.3. Type='password'.....	Σελ.25
Εικόνα 3.2.1. Δυνατότητες χρήστη μετά την ταυτοποίηση του.....	Σελ.32
Εικόνα 4.1. Φωτογραφικές μηχανές.....	Σελ.36
Εικόνα 4.2. Ψηφιακές βιντεοκάμερες.....	Σελ.36
Εικόνα 4.3. Ηλεκτρονικές κορνίζες.....	Σελ.37
Εικόνα 4.4. Προστατευτικές θήκες.....	Σελ.37
Εικόνα 5. Προϊόν καταστήματος.....	Σελ.39
Εικόνα 6.1.1 Μενού ταυτοποιημένου χρήστη.....	Σελ.49
Εικόνα 6.1.2 Μενού μη ταυτοποιημένου χρήστη.....	Σελ.49
Εικόνα 6.3.1 Περιεχόμενα καλαθιού.....	Σελ.57
Εικόνα 6.3.2 Αλλαγή ποσότητας.....	Σελ.57
Εικόνα 6.3.3 Περιεχόμενα καλαθιού μετά την αλλαγή ποσότητας.....	Σελ.57
Εικόνα 7.1.1 Συμπλήρωση επιπλέον στοιχείων.....	Σελ.60
Εικόνα 7.1.2 Μήνυμα επιβεβαίωσης.....	Σελ.60

Εικόνα 7.1.1.1 Δομή φακέλων.....	Σελ.61
Εικόνα 7.1.1.2 Έγγραφο αγοράς.....	Σελ.61
Εικόνα 7.2.1 Μενού επιλογών μη ταυτοποιημένου χρήστη.....	Σελ.65
Εικόνα 8.1. Μήνυμα παρακίνησης για σύνδεση στο σύστημα.....	Σελ.66
Εικόνα 8.2. Επιλογή αριθμού φωτογραφιών.....	Σελ.66
Εικόνα 8.3. Συμπλήρωση στοιχείων φωτογραφίας.....	Σελ.66
Εικόνα 8.4. Επιβεβαίωση αγοράς.....	Σελ.67
Εικόνα 8.1.1 Δομή φακέλων.....	Σελ.67
Εικόνα 8.1.2. Δομή υποφακέλων.....	Σελ.67
Εικόνα 8.1.3. Περιεχόμενα υποφακέλων.....	Σελ.68
Εικόνα 8.1.4. Έγγραφο εκτύπωσης.....	Σελ.68
Εικόνα 9.1.1 Στοιχεία καταστήματος.....	Σελ.77
Εικόνα 10.1.1 Προβολή πελατολογίου.....	Σελ.79
Εικόνα 10.2.1.1. Φόρμα καταχώρησης προϊόντος.....	Σελ.86
Εικόνα 10.2.2.1 Ιδιότητες προϊόντων.....	Σελ.90
Εικόνα 10.2.2.2 Αλλαγή στοιχείων προϊόντων.....	Σελ.90
Εικόνα 10.3.1.1 Καταχώρηση νέας κατηγορίας.....	Σελ.96
Εικόνα 10.3.2.1 Αλλαγή κατηγορίας.....	Σελ.97

➤ Εισαγωγή

Ένα ολοκληρωμένο ηλεκτρονικό κατάστημα θα πρέπει να διαθέτει όλες εκείνες τις λειτουργίες που θα το κάνουν εύχρηστο τόσο στους χρήστες- πελάτες όσο και στον διαχειριστή του συστήματος. Έτσι έπειτα από μια έρευνα σε ήδη υπάρχοντα ηλεκτρονικά καταστήματα (παρουσιάζεται παρακάτω) αποφασίστηκε ότι η ιστοσελίδα θα διαθέτει τις εξής λειτουργίες:

- ❖ Ταυτοποίηση των χρηστών στο σύστημα
 - Εγγραφή νέου χρήστη
 - Σύνδεση χρήστη στο σύστημα
 - Αποσύνδεση χρήστη από το σύστημα
- ❖ Παρουσίαση φωτογραφικών ειδών
 - Προβολή κατηγοριών
 - Προβολή προϊόντων
- ❖ Καλάθι αγορών
 - Προσθήκη στο καλάθι
 - Αφαίρεση από το καλάθι
 - Αλλαγή ποσότητας
- ❖ Αγορά προϊόντος
 - Report αγοράς
- ❖ Εκτύπωση φωτογραφιών
 - Report εκτύπωσης
- ❖ Επικοινωνία με το κατάστημα
- ❖ Σελίδα διαχειριστή
 - Πελάτες
 - Προϊόντα
 - Κατηγορίες

Στη συνέχεια παρουσιάζονται οι περιπτώσεις χρήσης του project έτσι όπως διαμορφώθηκαν έπειτα από συζήτηση με τον ενδιαφερόμενο της ιστοσελίδας (στη προκειμένη περίπτωση από τον επιβλέπων καθηγητή).

Ταυτοποίηση των χρηστών στο σύστημα

Περίπτωση Χρήσης: Εγγραφή νέου χρήστη

Χειριστής:

Πελάτης ιστοσελίδας

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

--

Στόχος:

Δημιουργία φόρμας για τη συμπλήρωση των στοιχείων του χρήστη που ενδιαφέρεται να αποκτήσει πρόσβαση στο σύστημα.

Κυρίως Σενάριο:

1. Ο χρήστης συμπληρώνει τα πεδία της φόρμας.
2. Τα δεδομένα αποθηκεύονται στον πίνακα της βάσης δεδομένων.

Εξαιρέσεις:

- 2α. Αν κάποιο από τα πεδία δεν συμπληρωθεί σωστά τότε δεν πραγματοποιείται η εγγραφή στη βάση δεδομένων και επιστρέφεται μήνυμα σφάλματος.

Περίπτωση Χρήσης: Σύνδεση στο σύστημα

Χειριστής:

Πελάτης ιστοσελίδας, ιδιοκτήτης

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

Πρέπει να έχει γίνει ήδη εγγραφή στο σύστημα.

Στόχος:

Δημιουργία φόρμας για τη συμπλήρωση του login και password για τον έλεγχο ταυτότητας του χρήστη.

Κυρίως Σενάριο:

1. Ο χρήστης συμπληρώνει τα πεδία της φόρμας.
2. Γίνεται έλεγχος ταυτοποίησης.
3. Ο χρήστης εισέρχεται στο σύστημα.

Εξαιρέσεις:

- 2α. Αν κάποιο από τα πεδία δεν συμπληρωθεί σωστά τότε δεν πραγματοποιείται η είσοδος στο σύστημα.

Περίπτωση Χρήσης: Αποσύνδεση από το σύστημα

Χειριστής:

Πελάτης ιστοσελίδας, ιδιοκτήτης

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

Πρέπει να έχει γίνει ήδη εγγραφή στο σύστημα.

Στόχος:

Με την αποσύνδεση ο χρήστης παύει να έχει τα δικαιώματα που είχε όταν ήταν συνδεδεμένος. Με αυτό τον τρόπο παρέχεται ασφάλεια στις προσωπικές επιλογές και στα στοιχεία του χρήστη αφού όταν πραγματοποιηθεί χάνεται η ταυτοποίηση του με το σύστημα.

Κυρίως Σενάριο:

1. Ο χρήστης πατάει κάποιο κουμπί που εκτελεί τον κώδικα της αποσύνδεσης.
2. Πραγματοποιείται η έξοδος από το σύστημα

Εξαιρέσεις:

Παρουσίαση φωτογραφικών ειδών

Περίπτωση Χρήσης: Προβολή κατηγοριών

Χειριστής:

Πελάτης ιστοσελίδας

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

--

Στόχος:

Τα προϊόντα του καταστήματος θα χωριστούν σε κατηγορίες ώστε να γίνει ευκολότερη η αναζήτηση συγκεκριμένου προϊόντος από τον χρήστη.

Κυρίως Σενάριο:

1. Ο χρήστης βλέπει τις υπάρχουσες κατηγορίες.
2. Επιλέγει αυτή που τον ενδιαφέρει.
3. Βλέπει τα προϊόντα που υπάρχουν σε αυτή τη κατηγορία με τα βασικά χαρακτηριστικά τους τα οποία εμφανίζονται από τον πίνακα που έχει αποθηκεύσει τις κατηγορίες αλλά και από τον πίνακα των προϊόντων.

Εξαιρέσεις:

Περίπτωση Χρήσης: Προβολή προϊόντων

Χειριστής:

Πελάτης ιστοσελίδας

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

--

Στόχος:

Τα προϊόντα του καταστήματος εκτός από τα βασικά χαρακτηριστικά τους που θα φαίνονται στις κατηγορίες θα διαθέτουν και επιπλέον λεπτομέρειες που πιθανόν θα ήθελε ο χρήστης να γνωρίζει.

Κυρίως Σενάριο:

1. Ο χρήστης επιλέγει ένα προϊόν.
2. Εμφανίζονται οι λεπτομέρειες για το προϊόν αυτό από τον αντίστοιχο πίνακα της βάσης δεδομένων που έχουν αποθηκευθεί.

Εξαιρέσεις:

Καλάθι αγορών

Περίπτωση Χρήσης: Προσθήκη στο καλάθι

Χειριστής:

Πελάτης ιστοσελίδας

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

--

Στόχος:

Ο χρήστης επιλέγει τα προϊόντα που θέλει να αγοράσει και εκείνα εμφανίζονται στο καλάθι αγορών του.

Κυρίως Σενάριο:

1. Ο χρήστης επιλέγει ένα προϊόν.
2. Εμφανίζονται οι λεπτομέρειες για το προϊόν αυτό.
3. Επιλέγει να προστεθεί στο καλάθι του.
4. Επιλέγει την ποσότητα του προϊόντος.
5. Η επιλογή του εγγράφεται στον αντίστοιχο πίνακα της βάσης δεδομένων.

Εξαιρέσεις:

- 4α. Αν η ποσότητα είναι μεγαλύτερη από αυτή που διαθέτει το κατάστημα τότε δεν γίνεται η εγγραφή και επιστρέφεται μήνυμα λάθους.

Περίπτωση Χρήσης: Αφαίρεση από το καλάθι

Χειριστής:

Πελάτης ιστοσελίδας

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

--

Στόχος:

Ο χρήστης αφαιρεί τα προϊόντα που δε θέλει να αγοράσει και εκείνα δεν εμφανίζονται πλέον στο καλάθι αγορών του.

Κυρίως Σενάριο:

1. Ο χρήστης επιλέγει ένα προϊόν.
2. Το προϊόν διαγράφεται από τον πίνακα που αρχικά είχε εγγραφεί.

Εξαιρέσεις:

Περίπτωση Χρήσης: Αλλαγή ποσότητας

Χειριστής:

Πελάτης ιστοσελίδας

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

--

Στόχος:

Ο χρήστης επιλέγει το προϊόν που θέλει να αλλάξει την ποσότητα του και εισάγει τη νέα ποσότητα.

Κυρίως Σενάριο:

1. Ο χρήστης επιλέγει ένα προϊόν.
2. Εισάγει τη νέα ποσότητα.
3. Η ποσότητα εγγράφεται εκ νέου στο πίνακα με τις εγγραφές-επιλογές του χρήστη.

Εξαιρέσεις:

- 2α. Αν η ποσότητα είναι μεγαλύτερη από αυτή που διαθέτει το κατάστημα τότε δε γίνεται η αλλαγή και επιστρέφεται μήνυμα λάθους.

Περίπτωση Χρήσης: Αγορά προϊόντος

Χειριστής:

Πελάτης ιστοσελίδας, ιδιοκτήτης

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

Πρέπει να έχει γίνει ήδη εγγραφή στο σύστημα.

Στόχος:

Ο χρήστης συμπληρώνει τη φόρμα ολοκλήρωσης των αγορών ανάλογα με τον τρόπο πληρωμής που θα επιλέξει και ολοκληρώνει τις αγορές του.

Κυρίως Σενάριο:

1. Ο χρήστης επιλέγει τρόπο πληρωμής.
2. Εισάγει τα στοιχεία που του ζητούνται.
3. Τα δεδομένα εισάγονται στο πίνακα των αγορών αλλά και στο αρχείο των αγορών και εμφανίζεται μήνυμα επιτυχούς αγοράς.

Εξαιρέσεις:

2α. Αν τα στοιχεία που εισάγει είναι λανθασμένα τότε δεν γίνεται η αγορά και επιστρέφεται μήνυμα λάθους.

Περίπτωση Χρήσης: Εκτύπωση φωτογραφιών

Χειριστής:

Πελάτης ιστοσελίδας, ιδιοκτήτης

Ενδιαφερόμενοι:

Ιδιοκτήτης, πελάτης

Προϋποθέσεις:

Πρέπει να έχει γίνει ήδη εγγραφή στο σύστημα.

Στόχος:

Ο χρήστης συμπληρώνει τη φόρμα με τα πεδία που αφορούν τις λεπτομέρειες για την εκτύπωση και στη συνέχεια γίνεται το ανέβασμα των φωτογραφιών.

Κυρίως Σενάριο:

1. Ο χρήστης επιλέγει τις φωτογραφίες που επιθυμεί να ανεβάσει.
2. Εισάγει τα επιπλέον στοιχεία που του ζητούνται.
3. Οι φωτογραφίες μαζί με τα υπόλοιπα δεδομένα εμφανίζονται στο φάκελο του πελάτη.

Εξαιρέσεις:

1α. Αν τα αρχεία που εισάγει ο χρήστης δεν είναι φωτογραφίες τότε δε γίνεται το ανέβασμα και επιστρέφεται μήνυμα λάθους.

2α. Αν τα στοιχεία που εισάγει είναι λανθασμένα τότε δε γίνεται το ανέβασμα και επιστρέφεται μήνυμα λάθους.

Περίπτωση Χρήσης: Διαχείριση συστήματος

Χειριστής:

Ιδιοκτήτης

Ενδιαφερόμενοι:

Ιδιοκτήτης

Προϋποθέσεις:

Πρέπει να γνωρίζει τους κωδικούς πρόσβασης.

Στόχος:

Ο ιδιοκτήτης έχει πρόσβαση στη βάση δεδομένων και μέσω ενός γραφικού περιβάλλοντος μπορεί και κάνει τις αλλαγές που επιθυμεί.

Κυρίως Σενάριο:

1. Ο ιδιοκτήτης εισάγει τα στοιχεία πρόσβασης.
2. Γίνεται η ταυτοποίηση με το σύστημα.
3. Εμφανίζεται το μενού επιλογών του διαχειριστή.

Εξαιρέσεις:

2α. Αν τα στοιχεία που εισάγει είναι λανθασμένα τότε δε γίνεται η ταυτοποίηση και επιστρέφεται μήνυμα λάθους.

Για την υλοποίηση των παραπάνω περιπτώσεων χρήσης θα χρησιμοποιηθεί μια βάση δεδομένων όπου θα αποθηκεύονται τα στοιχεία των προϊόντων, των πελατών και των αγορών που πραγματοποιούνται. Επίσης θα δημιουργηθεί ένα γραφικό περιβάλλον το οποίο θα είναι εύχρηστο στους πελάτες του ηλεκτρονικού καταστήματος ώστε να πραγματοποιούνται εύκολα οι αγορές τους.

Τα εργαλεία που θα χρησιμοποιηθούν είναι το xampp το οποίο είναι μια πλατφόρμα που μας βοηθάει στη δημιουργία ιστοσελίδων και περιέχει την mysql, με την οποία θα δημιουργηθεί η βάση δεδομένων μας καθώς και τη γλώσσα php με την οποία θα γραφεί, κατά κύριο λόγο, ο κώδικας που θα τρέχει στον browser μας. Την πλατφόρμα αυτή μπορεί να την κατεβάσει οποιοσδήποτε από το <http://www.apachefriends.org/en/xampp.html> καθώς επίσης να βρει και οδηγίες εγκατάστασης στο <http://www.slideshare.net/rodotheos/xampp>.

Επιπλέον το notepad++ είναι ένα ακόμη εργαλείο, το οποίο αποτελεί βελτιωμένη έκδοση του κλασικού notepad και θα μας βοηθήσει στη καλύτερη γραφή του κώδικα μας αφού μπορεί και αναγνωρίζει τύπους αρχείων όπως php και css. Το πρόγραμμα αυτό μπορεί να το κατεβάσει κανείς από το <http://notepad-plus-plus.org/>. Αξίζει να αναφερθεί ότι και οι δύο εφαρμογές είναι προϊόντα ελεύθερου λογισμικού οπότε δεν επιβαρύνουν οικονομικά οποιοδήποτε θα ήθελε να ασχοληθεί με τη δημιουργία ιστοσελίδων.

Τα μαθήματα που πρέπει κανείς να έχει ιδιαίτερη εξοικείωση είναι οι βάσεις δεδομένων και η ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον τα οποία καλύπτουν μεγάλο φάσμα των γνώσεων που θα χρησιμοποιηθούν. Επίσης χρήσιμο, σε άποψη οργάνωσης των απαιτήσεων αλλά και μεθοδικότητας του τρόπου εργασίας, είναι το μάθημα Τεχνολογία Λογισμικού II.

➤ Βιβλιογραφική έρευνα

Η ανάλυση ενός ηλεκτρονικού καταστήματος αναφέρεται τόσο στη λειτουργικότητα του όσο και στην προδιάθεση που δίνει στο χρήστη ώστε να ανταποκριθεί θετικά στην αγορά προϊόντων. Μετά από μια μελέτη που έγινε στο ηλεκτρονικό εμπόριο προέκυψε ότι η χρήση των χρωμάτων κόκκινου και κίτρινου μπορεί να αυξήσει την παρακίνηση αγοράς από ότι κάποιο άλλο χρώμα. Επίσης η χρήση εικόνων και βίντεο αυξάνουν το χρόνο παραμονής στο ηλεκτρονικό κατάστημα. Επιπλέον με τη δημιουργία εύχρηστων μενού και οδηγιών ο χρήστης εξυπηρετείται καλύτερα, οπότε μένει σαφώς πιο ευχαριστημένος και μπορεί να επισκεφθεί ξανά το κατάστημα στο μέλλον[1].

Έτσι κάνοντας μια έρευνα σε παρόμοιες εφαρμογές διαπιστώθηκε ότι ο τρόπος παρουσίασης θα πρέπει να περιλαμβάνει τις κατηγορίες των προϊόντων που διαθέτει το κατάστημα, δυνατότητα προβολής λεπτομερειών των προϊόντων, χρήση κωδικών πρόσβασης, ευκολία πρόσβασης στο καλάθι αγορών καθώς και δυνατότητες αλλαγής- διαγραφής προϊόντων που επέλεξε ο χρήστης.

Παρακάτω παρουσιάζονται τα αποτελέσματα της έρευνας:

	Fotodesmos [2]	Print-photos- online [3]	Eshop [4]	Photo Print (εργασία)
Κατηγορίες προϊόντων	Ναι	Όχι	Ναι	Ναι
Πληροφορίες προϊόντων	Ναι	Όχι	Ναι	Ναι
Προσθήκη στο καλάθι	Ναι	Όχι	Ναι	Ναι
Αφαίρεση από το καλάθι	Ναι	Όχι	Ναι	Ναι
Αλλαγή ποσότητας προϊόντος	Ναι	Όχι	Ναι	Ναι
Εκτύπωση φωτογραφιών	Όχι	Ναι	Όχι	Ναι
Χρήση κωδικών πρόσβασης	Ναι	Ναι	Ναι	Ναι

Όπως μπορεί κανείς να διαπιστώσει η εκτύπωση φωτογραφιών και η αγορά προϊόντων δε συνδυάζεται σε κανένα από τα παραπάνω ηλεκτρονικά καταστήματα αλλά γίνεται πιο εξειδικευμένα σε κάθε τομέα. Δηλαδή το Print-photos-online δε διαθέτει ακόμα ηλεκτρονικές αγορές αλλά παρόλα αυτά δίνει πολλές δυνατότητες στο χρήστη όσον αφορά το θέμα της εκτύπωσης. Το Fotodesmos από την άλλη παρουσιάζει μεγάλη ποικιλία στα προϊόντα που διαθέτει ως κατάστημα αλλά δεν ασχολείται καθόλου με τις εκτυπώσεις φωτογραφιών. Το ίδιο ισχύει και για το Eshop το οποίο μάλιστα δεν αναφέρεται μόνο στις πωλήσεις φωτογραφικών ειδών αλλά και άλλων ηλεκτρονικών ειδών. Συμπερασματικά θα μπορούσαμε να αναφέρουμε ότι δεν βρέθηκε κάποιο ηλεκτρονικό κατάστημα που να πληρεί τις προϋποθέσεις και των δύο

βασικών λειτουργιών τις εργασίας, οι επιμέρους όμως πληροφορίες που διέθεταν ήταν αρκετά χρήσιμες.

Παρακάτω παρουσιάζεται η αναλυτική διαδικασία που ακολουθήθηκε για την υλοποίηση της πτυχιακής εργασίας:

Δομή εργασίας

Θεωρήθηκε αποτελεσματικότερο η εργασία να παρουσιαστεί ως θεματική ενότητα, γενικός σχολιασμός, εικόνα που βλέπει ο χρήστης, ο κώδικας που παράγει το αποτέλεσμα που πραγματεύεται η ενότητα και τέλος ο σχολιασμός των σημαντικών κομματιών κώδικα. Με τον τρόπο αυτό ο αναγνώστης μαθαίνει γενικές πληροφορίες για την ενότητα, έχει οπτική εικόνα για το περιβάλλον του χρήστη, μπορεί να διαβάσει αναλυτικά τον κώδικα που παράγει το αποτέλεσμα και τέλος να διαβάσει κάποιες διευκρινίσεις για τυχόν δυσνόητα κομμάτια.

1. Δημιουργία της Βάσης Δεδομένων

Ο παρακάτω κώδικας εκφράζει τη σύνδεση με τη βάση δεδομένων και ο λόγος που έγινε σε ξεχωριστό αρχείο είναι γιατί ο συγκεκριμένος κώδικας θα χρησιμοποιηθεί σε σχεδόν όλα τα αρχεία της εργασίας οπότε αφενός κερδίζουμε σε επαναχρησιμότητα και αφετέρου σε μία πιθανή αλλαγή του κωδικού πρόσβασης ή για τη μεταφορά της εργασίας σε άλλο υπολογιστή με διαφορετικό κωδικό πρόσβασης, θα χρειαστεί να γίνει μόνο μια αλλαγή στο συγκεκριμένο αρχείο και όχι σε κάθε αρχείο του project.

Αξίζει να σημειωθεί ότι σε όλο το project ο τρόπος ονοματολογίας των μεταβλητών έγινε με το στιλ γραφής τύπου java [5].

Κώδικας dbConnection:

```
<?php
$con=mysql_connect("localhost","root","");
if (!$con)
{
    die('Could not connect: ' . mysql_error());
}
?>
```

Κώδικας createDB:

Αν κάποιος «τρέξει» το συγκεκριμένο αρχείο έχει δημιουργήσει ουσιαστικά τη βάση δεδομένων eshop με τους πίνακες της και τα αντίστοιχα πεδία τους, όπως παρουσιάζονται παρακάτω:

```
<?php
include("dbConnection.php");
$query="CREATE DATABASE eshop";
```

```

mysql_query($query,$con);
mysql_select_db("eshop",$con);

$sqlC="CREATE TABLE category(categoryId int(10) AUTO_INCREMENT NOT
NULL,categoryName varchar(50) NOT NULL,PRIMARY KEY(categoryId))";
mysql_query($sqlC,$con);
echo"Category table created ";

$sqlP="CREATE TABLE product(productId int(10) AUTO_INCREMENT NOT
NULL,categoryId int(10) NOT NULL,type varchar(50) NOT NULL,info
varchar(100) NOT NULL, price float NOT NULL,image varchar(50) NOT NULL,
PRIMARY KEY(productId))";
mysql_query($sqlP,$con);
echo"Product table created ";

$sqlS="CREATE TABLE stock(productId int(10) AUTO_INCREMENT NOT
NULL,quantity int(10) NOT NULL,PRIMARY KEY(productId))";
mysql_query($sqlS,$con);
echo"stock table created ";

$sqlB="CREATE TABLE buy(buyId int(10) AUTO_INCREMENT NOT
NULL,customerId int(10) NOT NULL,dateOfBuy date NOT NULL,productId int(10)
NOT NULL,PRIMARY KEY(basketId))";
mysql_query($sqlB,$con);
echo"Basket table created ";

$sqlCu="CREATE TABLE customer(customerId int(10) AUTO_INCREMENT NOT
NULL,username varchar(50) NOT NULL,password varchar(50) NOT
NULL,firstName varchar(50) NOT NULL,lastName varchar(50) NOT
NULL,phoneNumber varchar(50) NOT NULL,address varchar(50),city
varchar(50),country varchar(50),email varchar(50),PRIMARY KEY(customerId))";
mysql_query($sqlCu,$con);
echo"Customer table created ";

$sqlBuy="CREATE TABLE visitorbasket(basketId int(10) AUTO_INCREMENT
NOT NULL,productId int(10) NOT NULL,dateOfBuy date NOT NULL,quantity
int(10) NOT NULL, PRIMARY KEY(basketId))";
mysql_query($sqlBuy,$con);
echo"Visitor Basket product table created ";

$sqlBuy="CREATE TABLE customerbasket(customerBasketId int(10)
AUTO_INCREMENT NOT NULL,customerId int(10) NOT NULL,productId int(10)
NOT NULL,dateOfBuy date NOT NULL,quantity int(10) NOT NULL, PRIMARY
KEY(basketId))";
mysql_query($sqlBuy,$con);
echo"Customer Basket product table created ";

mysql_close($con);
?>

```

Παρακάτω παρουσιάζονται οι πίνακες όπως τους βλέπουμε στο εργαλείο μας phpMyAdmin καθώς και η χρησιμότητα κάθε πίνακα:

Πίνακας	Ενέργεια	Εγγραφές ¹	Τύπος	Collation
buy		33	MyISAM	utf8_general_ci
category		4	MyISAM	utf8_general_ci
customer		5	MyISAM	utf8_general_ci
customerbasket		1	MyISAM	utf8_general_ci
product		20	MyISAM	utf8_general_ci
stock		20	MyISAM	utf8_general_ci
visitorbasket		0	MyISAM	utf8_general_ci
7 Πίνακας/Πίνακες	Σύνολο	83	MyISAM	utf8_general_ci

Εικόνα.1 Πίνακες στη βάση δεδομένων

Στη συνέχεια παρουσιάζεται αναλυτικά η χρησιμότητα κάθε πίνακα καθώς και ο λόγος που ο κάθε ένας διαθέτει αυτά τα πεδία.

- Ο πίνακας category δημιουργήθηκε για να αποθηκεύει τα ονόματα των κατηγοριών που αποτελούν το eshop. Τα πεδία του πίνακα θα εισάγονται αυτόματα στο κεντρικό μενού ώστε ο πελάτης να μπορεί να διαλέγει προϊόντα από όποια κατηγορία τον ενδιαφέρει. Το πεδίο categoryId είναι το κλειδί του πίνακα και προσδίδει μοναδικό γνώρισμα σε κάθε κατηγορία ώστε να είναι ένα μέσω για την αναγνώριση κάθε εγγραφής αλλά και συμβάλλει και στην αποφυγή διπλοεγγραφών.
- Τα πεδία του πίνακα product περιγράφουν όλα τα χαρακτηριστικά που πρέπει να διαθέτει ένα προϊόν για να είναι προς πώληση. Συγκεκριμένα:
 Το productId (κωδικό προϊόντος), που είναι το κλειδί του πίνακα, περιέχει ένα κωδικό αναγνώρισης για κάθε προϊόν (εγγραφή). Με τον τρόπο αυτό προσδίδουμε ένα μοναδικό γνώρισμα σε κάθε προϊόν και παράλληλα αποφεύγουμε τις διπλοεγγραφές. Το πεδίο categoryId περιέχει τον κωδικό της κατηγορίας που υπάγεται το συγκεκριμένο προϊόν. Το πεδίο type αφορά τον τύπο (μάρκα) του προϊόντος. Το πεδίο info περιέχει όλες τις πληροφορίες-χαρακτηριστικά κάθε μοντέλου για να γνωρίζει ο πελάτης διάφορες λεπτομέρειες για το προϊόν που θα αγοράσει. Το πεδίο price αναφέρεται στην αγοραστική αξία του συγκεκριμένου προϊόντος και τέλος το πεδίο image περιέχει το όνομα της φωτογραφίας που εμφανίζεται μαζί με τη περιγραφή του στην αντίστοιχη σελίδα.
- Ο πίνακας stock αναφέρεται στην ποσότητα που διαθέτει το κατάστημα για κάθε προϊόν, για αυτό και απαρτίζεται από τα πεδία productId που ουσιαστικά

αναφέρει σε ποιο προϊόν αντιστοιχεί η ανάλογη ποσότητα που περιέχει το πεδίο quantity.

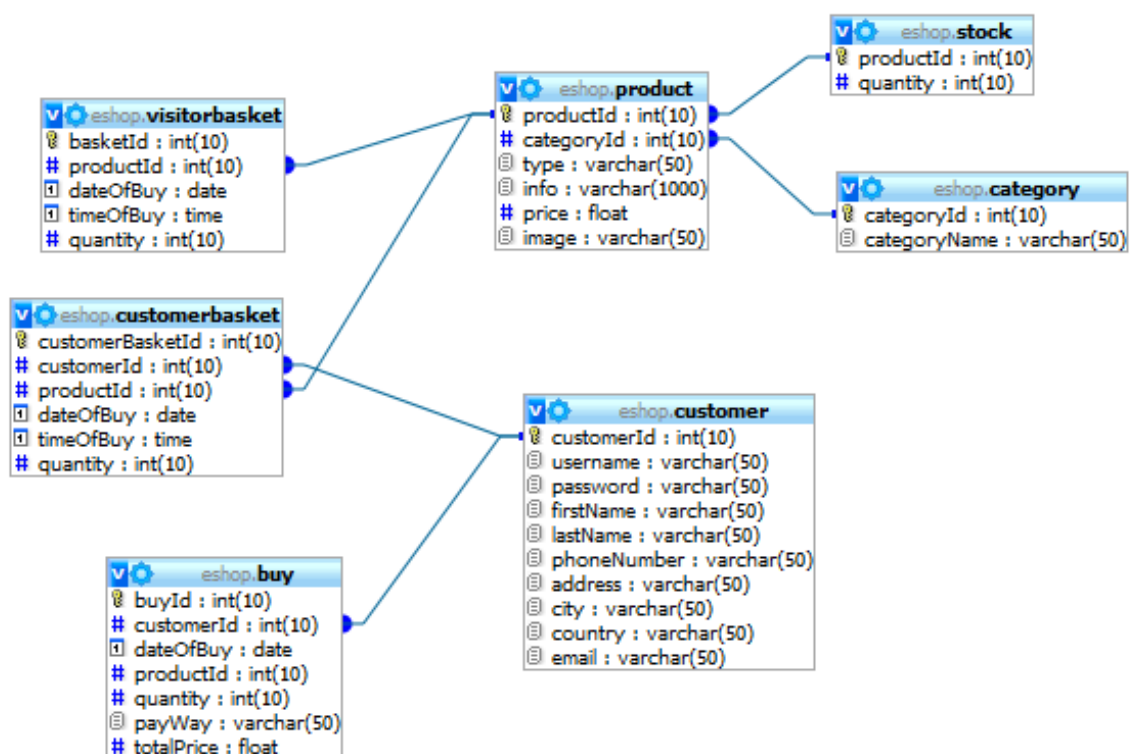
- Ο πίνακας customer περιέχει τα στοιχεία κάθε πελάτη τα οποία χρειάζονται τόσο για την ταυτοποίηση του στο σύστημα όσο και για τη διευκόλυνση των ηλεκτρονικών αγορών του. Ο πίνακας περιέχει τα πεδία: customerId που είναι το κλειδί του πίνακα και δίνει μοναδικό γνώρισμα σε κάθε πελάτη (εγγραφή), username που αναφέρεται στο όνομα πρόσβασης στο σύστημα, password που αφορά τον κωδικό πρόσβασης στο σύστημα, firstName που αναφέρεται στο όνομα του πελάτη, lastName που αναφέρεται στο επώνυμο του χρήστη, phoneNumber που περιέχει το τηλέφωνο του, address που περιέχει τη διεύθυνση του, city που περιέχει τη πόλη διαμονής του, country που περιέχει τη χώρα διαμονής του και τέλος το email που περιέχει την ηλεκτρονική του διεύθυνση email.

Οι επόμενοι τρεις πίνακες αφορούν τις ηλεκτρονικές αγορές και υλοποιούν την απλή επιλογή των προϊόντων, δηλαδή την εισαγωγή στο καλάθι αγορών, αλλά και τις ολοκληρωμένες αγορές.

- Ο πίνακας customerbasket χρησιμεύει στην αποθήκευση των προϊόντων που επιλέγει ένας ταυτοποιημένος χρήστης του συστήματος. Ο πίνακας αυτός ουσιαστικά αποθηκεύει τα προϊόντα που επιλέγει ο χρήστης αλλά δεν έχει ακόμα αγοράσει. Τα πεδία του πίνακα είναι: customerBasketId που είναι το κλειδί του πίνακα και αναφέρει τη μοναδική τιμή κάθε επιλογής- προσθήκης στο καλάθι, customerId που αναφέρει ποιος χρήστης κάνει τις συγκεκριμένες επιλογές, productId που αναφέρει ποιο προϊόν επιλέχθηκε, dateOfBuy που περιέχει την ημερομηνία επιλογής, timeOfBuy που περιέχει την ώρα επιλογής του προϊόντος και τέλος το πεδίο quantity που αναφέρεται στη ποσότητα κάθε προϊόντος που επιθυμεί να αγοράσει ο χρήστης.
- Ο πίνακας visitorbasket χρησιμεύει στην αποθήκευση των προϊόντων που επιλέγει ένας μη ταυτοποιημένος χρήστης του συστήματος. Ο πίνακας αυτός ουσιαστικά αποθηκεύει τα προϊόντα που επιλέγει ο χρήστης αλλά δεν έχει ακόμα αγοράσει. Τα πεδία του πίνακα είναι: basketId που είναι το κλειδί του πίνακα και αναφέρει τη μοναδική τιμή κάθε επιλογής- προσθήκης στο καλάθι, productId που αναφέρει ποιο προϊόν επιλέχθηκε, dateOfBuy που περιέχει την ημερομηνία επιλογής, timeOfBuy που περιέχει την ώρα επιλογής του προϊόντος και τέλος το πεδίο quantity που αναφέρει την ποσότητα κάθε προϊόντος που επιθυμεί να αγοράσει ο χρήστης.
- Ο τελευταίος πίνακας, ο πίνακας buy, αφορά τις αγορές που πραγματοποιήθηκαν. Δηλαδή χρησιμεύει ως αρχείο για τις αγορές που έχει κάνει κάθε πελάτης. Τα πεδία του πίνακα είναι: buyId που είναι το κλειδί του πίνακα και αναφέρετε στη μοναδική τιμή κάθε ολοκληρωμένης αγοράς, customerId που αναφέρει ποιος χρήστης κάνει τις συγκεκριμένες αγορές, dateOfBuy που περιέχει την ημερομηνία αγοράς, productId που αναφέρει ποιο προϊόν αγοράστηκε, quantity που αναφέρει την ποσότητα κάθε προϊόντος που αγόρασε ο χρήστης, payWay που περιέχει τον τρόπο πληρωμής των αγορών και τέλος το πεδίο totalPrice που περιέχει το συνολικό κόστος των αγορών που έκανε ο πελάτης.

Τέλος παρουσιάζεται το σχεσιακό μοντέλο της βάσης μας όπου βλέπουμε τις συνδέσεις μεταξύ των πινάκων. Οι πίνακες συνδέονται με τέτοιο τρόπο ώστε να υπάρχει σωστή συνοχή των δεδομένων ανάμεσα στους πίνακες. Δηλαδή:

- ο πίνακας category συνδέθηκε με τον πίνακα product γιατί κάθε προϊόν πρέπει υποχρεωτικά να ανήκει σε κάποια από τις κατηγορίες του καταστήματος.
- Ο πίνακας stock συνδέθηκε επίσης με τον πίνακα product γιατί κάθε προϊόν πρέπει να διαθέτει μια ποσότητα η οποία αναφέρεται στην ποσότητα στοκ του καταστήματος.
- Ο πίνακας visitorbasket συνδέθηκε με τον πίνακα product γιατί κάθε προϊόν που εισάγεται στο καλάθι αγορών πρέπει να αντιστοιχεί σε ένα καταγεγραμμένο προϊόν της βάσης και όχι σε μία τιμή που πιθανόν δεν θα είναι καταχωρημένη.
- Ο πίνακας customerbasket συνδέθηκε με τον πίνακα customer για να μπορεί κανείς να ξεχωρίσει ποιος πελάτης έκανε τις συγκεκριμένες επιλογές. Ουσιαστικά είναι ένας τρόπος ταυτοποίησης των επιλογών με το χρήστη. Ο πίνακας αυτός συνδέεται επίσης με τον πίνακα product για τον ίδιο λόγο που συνδέθηκε και ο πίνακας visitorbasket.
- Ο πίνακας buy συνδέθηκε με τον πίνακα customer για να μπορεί κανείς να ξεχωρίζει ποιος πελάτης έκανε τις συγκεκριμένες αγορές. Ουσιαστικά είναι ένας τρόπος ταυτοποίησης των αγορών με το χρήστη που τις πραγματοποίησε.



Εικόνα 1.2. Σχεσιακό μοντέλο βάσης δεδομένων

Τώρα πλέον η βάση δεδομένων μας είναι έτοιμη για την ορθή αποθήκευση των δεδομένων μας.

2. Αρχική σελίδα



Εικόνα 2. Αρχική σελίδα

Η αρχική σελίδα (index) δίνει στο χρήστη τη δυνατότητα να περιηγηθεί στα υπόλοιπα τμήματα (σελίδες) του eshop μέσω του κεντρικού μενού που διαθέτει καθώς και να συνδεθεί ή να εγγραφεί στο σύστημα. Τα χρώματα κόκκινο και κίτρινο επιλέχθηκαν για να τραβούν τη προσοχή του χρήστη και με τη βοήθεια της εικόνας και του βίντεο να αυξηθεί ο χρόνος παραμονής του επισκέπτη [1]. Παρακάτω παρουσιάζεται ο κώδικας που δίνει αυτό το αποτέλεσμα.

Κώδικας index.php :

```
<?php
session_start();
$user=$_SESSION['username'];
include("menu.php");
?>
<html>
<head>
<title>Αρχική σελίδα</title>
<meta content="text/html; charset=iso-8859-7">
<link rel="stylesheet" type="text/css" href="linkStyle.css">
<link rel="stylesheet" type="text/css" href="style.css">
<style type="text/css">
body
{
background-image:url('/ptuxiakh/photos/photoCamera.jpg');
background-position:55% 60%;
background-repeat: no-repeat;
background-color:#590000;
background-attachment:fixed;
```

```

}
</style>
</head>
<body>
<h1>Photo Print</h1>
<form action="login.php" method="post">
<table align="left">
<tr><td>
<p>Λένε πως μια εικόνα ισούται με χίλιες λέξεις...<br>
Εδώ λοιπόν λέμε τις ιστορίες μας δίνοντας <br>
μια νέα διάσταση στο κόσμο της φωτογραφίας.<br>
</p></td></tr>
<tr><td><p>
Στο κατάστημα μας θα βρείτε πολλές προσφορές <br>
σε φωτογραφικά είδη καθώς και τη δυνατότητα <br>
να εκτυπώσετε τις φωτογραφίες από το σπίτι σας!<br><br>
</p></td>
</tr>
<tr><td>
<embed src="/ptuxiakh/Moview.wmv" align="left" width="300" height="200"
autostart="false"></embed></td></tr>
</table>
<table align="right" cellpadding="5">

<?php
if(isset($_SESSION['validUser']))
{
    echo"<p align='right'><b>Έχετε συνδεθεί ως χρήστης:</b> ",$user,"<br> <a
href='logout.php'> Αποσύνδεση</a><br> <a href='showBasket.php'>Προβολή
Καλαθιού</a></p>";
}
else
{
    echo"<form action='login.php' method='post'>";
    echo"<table align='right' cellpadding='5'>";
    echo"<tr align='center'><td><a
href='userRegistration.php'><b>Εγγραφή</b></a></td>";
    echo"<tr align='right'><td><b>username </b><input type='text'
name='username'></td>";
    echo"<tr align='right'><td><b>password </b><input type='password'
name='password'></td></tr>";
    echo"<tr><td align='right'><input type='submit' value='Σύνδεση'></td></tr>";
}
?>

</table>
</form>
</body>
</html>

```

Όπως μπορεί κανείς να παρατηρήσει στο τμήμα head του κώδικα (με μπλε γραφή) υπάρχουν δύο αρχεία τύπου CSS [6], το style και το linkStyle. Το style.css αφορά το γενικό μοτίβο των χρωμάτων, της γραμματοσειράς και του στυλ που θα ακολουθεί συνολικά η ιστοσελίδα. Το linkStyle.css αφορά τα χρώματα και τις ιδιότητες των συνδέσμων (links) που υπάρχουν στην ιστοσελίδα. Επιπλέον ξεχωριστά από τις μορφοποιήσεις που υπάρχουν στα παραπάνω αρχεία, υπάρχει ακόμη ένα τμήμα CSS κώδικα που αφορά το τμήμα της ιστοσελίδας body και εσκεμμένα δεν βρίσκεται στο αρχείο style.css γιατί έχει ξεχωριστές τιμές για κάθε τμήμα της σελίδας. Παρακάτω παρουσιάζεται ο κώδικας των δύο αυτών αρχείων:

Κώδικας style.css:

```
h1
{
    color:#FFFF33;
    font-size:40px;
    font-weight:bold;
    font-style:italic;
    font-family:tahoma;
    position:relative;
    top: 0px;
    left: 500px;
}
p
{
    color:#FFFF33;
    font-style:italic;
    font-family:tahoma;
}
table
{
    color:#FFFF33;
    font-style:bold;
    font-family:tahoma;
}
```

Όπως μπορεί κανείς να διαπιστώσει για τη μορφοποίηση της σελίδας χρησιμοποιήθηκε το χρώμα FFFF33 για τους πίνακες, την επικεφαλίδα και τις παραγράφους. Επίσης η γραμματοσειρά που επιλέχθηκε είναι η tahoma και το στυλ γραφής είναι πλάγια γραφή (*italic*) στις παραγράφους και στην επικεφαλίδα, ενώ στους πίνακες επιλέχθηκε η έντονη (**bold**) γραφή.

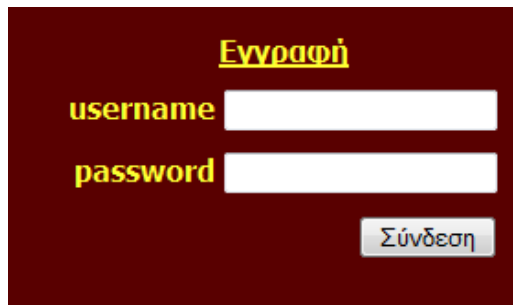
Κώδικας linkStyle.css:

```
a:link
{
    color:#FF9900;
    font-weight:bold;
    text-decoration:underline;
}
a:visited
{
    color:#FFFF33;
    font-weight:bold;
}
a:hover
{
    color:#FF9999;
    font-weight:bold;
    text-decoration:none;
}
a.noStyle
{
    color: white;
    text-decoration:none;
}
```

Για τη μορφοποίηση των συνδέσμων χρησιμοποιήθηκε το χρώμα FF9900 για τα link που θα υπάρχουν, το χρώμα FFFF33 για τα link που έχει ήδη επισκεφθεί ο χρήστης και το χρώμα FF9999 όταν ο κέρσορας βρίσκεται πάνω στο link. Το στυλ γραφής είναι έντονο (**bold**) και στο απλό link είναι και υπογραμμισμένο (underline).

3. Ταυτοποίηση χρήστη στο σύστημα

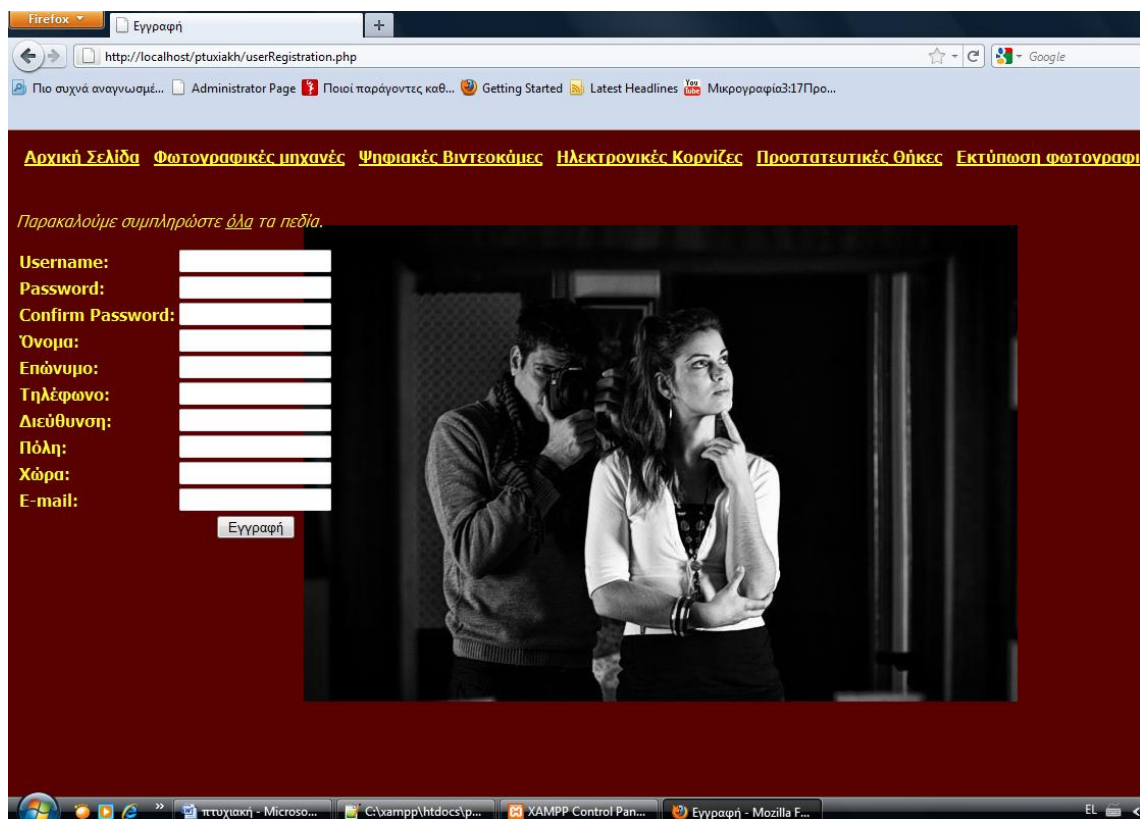
Αν ο χρήστης δεν είναι ήδη εγγεγραμμένος στο σύστημα τότε μπορεί πατώντας στο σύνδεσμο «εγγραφή», να συμπληρώσει τη φόρμα με τα στοιχεία του και να αποκτήσει κωδικούς πρόσβασης.



The image shows a registration form with a dark red background. At the top, the word "Εγγραφή" is written in yellow. Below it, there are two white input fields. The first is labeled "username" and the second is labeled "password". At the bottom right of the form, there is a button labeled "Σύνδεση" (Login).

Εικόνα 3. Φόρμα login

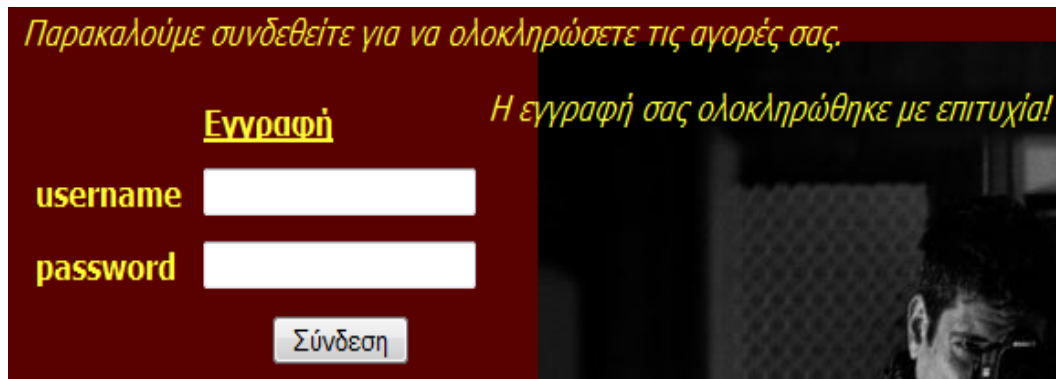
3.1 Εγγραφή νέου χρήστη



The image is a screenshot of a web browser displaying a registration page. The browser's address bar shows the URL "http://localhost/ptuxiakh/userRegistration.php". The page has a dark red header with navigation links in yellow: "Αρχική Σελίδα", "Φωτογραφικές μηχανές", "Ψηφιακές Βιντεοκάμερες", "Ηλεκτρονικές Κορνίζες", "Προστατευτικές Θήκες", and "Εκτύπωση Φωτογραφιών". Below the header, there is a message in Greek: "Παρακαλούμε συμπληρώστε όλα τα πεδία." (Please fill in all fields). The registration form consists of several input fields with labels in Greek: "Username:", "Password:", "Confirm Password:", "Όνομα:", "Επώνυμο:", "Τηλέφωνο:", "Διεύθυνση:", "Πόλη:", "Χώρα:", and "E-mail:". A button labeled "Εγγραφή" is located below the form. To the right of the form, there is a black and white photograph of a man and a woman. The browser's taskbar at the bottom shows several open applications, including "πτυχιακή - Micro...", "C:\xampp\htdocs\p...", "XAMPP Control Pan...", and "Εγγραφή - Mozilla F...".

Εικόνα 3.1.1 Φόρμα εγγραφής

Αν η φόρμα συμπληρωθεί σωστά ενημερώνεται ο χρήστης ότι έγινε επιτυχής εγγραφή και ζητείται εκ νέου η συμπλήρωση του username και του password ώστε να εισέλθει στο σύστημα.



Εικόνα 3.1.2. Επιτυχής εγγραφή

Τα αρχεία που χρησιμοποιήθηκαν είναι το userRegistration.php που περιέχει τη φόρμα συμπλήρωσης και το InsertIntoCustomer.php που κάνει τους ελέγχους για την ορθότητα των στοιχείων και την εγγραφή του πελάτη στη βάση. Τέλος, χρησιμοποιήθηκε και το αρχείο loginForm.php το οποίο περιέχει τη φόρμα συμπλήρωσης για την είσοδο στο σύστημα.

Κώδικας userRegistration.php:

```
<?php
include("menu.php");
?>
<html>
<head>
<title>Εγγραφή</title>
<meta content="text/html; charset=iso-8859-7">
<link rel="stylesheet" type="text/css" href="style.css">
<link rel="stylesheet" type="text/css" href="linkStyle.css">
<STYLE TYPE="text/css">
body
{
    background-image:url('/ptuxiakh/photos/image.jpg');
    background-position: 50% 50%;
    background-repeat: no-repeat;
    background-color:#590000;
    background-attachment:fixed;
}
td
{
    color:#FFFF33;
    font-weight: bold;
}
</STYLE>
</head>
<body>
<p>Παρακαλούμε συμπληρώστε <u>όλα</u> τα πεδία.</p>
<form action="insertIntoCustomer.php" method= "post">
<table border="0" align="left">
<tr>
<td> <b>Username:</b></td>
<td> <input type="text" name="username"></td>
```



```

</tr>
<tr><td> <b>Password:</b></td>
<td><input type="password" name="password"></td></tr>
<tr><td> <b>Confirm Password:</b></td>
<td><input type="password" name="cPassword"></td></tr>
<tr><td> <b>Όνομα:</b></td>
<td><input type="text" name="firstName"></td></tr>
<tr><td> <b>Επώνυμο:</b></td>
<td><input type="text" name="lastName"></td></tr>
<tr><td> <b>Τηλέφωνο:</b></td>
<td><input type="text" name="phoneNumber"></td></tr>
<tr><td> <b>Διεύθυνση:</b></td>
<td><input type="text" name="address"></td></tr>
<tr><td> <b>Πόλη:</b></td>
<td><input type="text" name="city"></td></tr>
<tr><td> <b>Χώρα:</b></td>
<td><input type="text" name="country"></td></tr>
<tr><td> <b>E-mail:</b></td>
<td><input type="text" name="email"></td></tr>
<tr><td></td>
<td align="center">
<input type="submit" value="Εγγραφή"></td></tr>
</table>
</form>
</body>
</html>

```

Όπως μπορεί κανείς να παρατηρήσει το αρχείο αυτό δημιουργεί τη φόρμα συμπλήρωσης μέσω της ετικέτας `<input>` με τύπο κείμενο (`type=text`) ώστε πατώντας ο χρήστης το submit button με ετικέτα «Εγγραφή» τα στοιχεία της φόρμας να αποσταλούν στη φόρμα `insertIntoCustomer` `<form action="insertIntoCustomer.php" method="post">`. Η μέθοδος που χρησιμοποιήθηκε για την αποστολή των δεδομένων είναι η post για να μη παρουσιάζονται οι πληροφορίες αυτές στη μπάρα διεύθυνσης (URL).

Τα πεδία password και confirm password είναι τύπου password και όχι τύπου text ώστε όταν πληκτρολογεί ο χρήστης τους κωδικούς του να φαίνονται κρυπτογραφημένοι με τις χαρακτηριστικές τελείες αντί για το κείμενο που πληκτρολογεί.



Εικόνα 3.1.3. Type='password'

Κώδικας InsertIntoCustomer.php:

```
<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query ('set character set utf8');
$user=$_POST['username'];

//check if this username is already used
function usedUsername($user)
{
    $search = mysql_query("SELECT username FROM customer where
    username='$user'");
    $row= mysql_fetch_array($search);
    if($row!="")
    {
        include("userRegistration.php");
        echo "<p>To username <b>",$row[0], "</b> υπάρχει ήδη.</p>";
        return true;
    }
    else
    {
        return false;
    }
}

function reservedName($user)
{
    if($user=="admin")
    {
        return true;
    }
    else
    {
        return false;
    }
}

//check if email have the right form
function validEmail($mailAddress)
{
    if(ereg('^([a-zA-Z0-9\.\_\-]+@([a-zA-Z0-9][a-zA-Z0-9\-\]*\.)+[a-zA-Z] + $',
    $mailAddress))
    {
        return true;
    }
    else
    {
        return false;
    }
}
```

```

}

//check if the data type is a string and the length of the string
function sanityCheck($string, $type, $length)
{
    $type = 'is_'. $type;
    if(!$type($string))                //check if the right data type
    {
        return FALSE;
    }
    elseif(empty($string))            // now we see if there is anything in the string
    {
        return FALSE;
    }
    elseif(strlen($string) > $length) // then we check how long the string is
    {
        return FALSE;
    }
    else
    {
        return TRUE; // if all is well, we return TRUE
    }
}

```

```

function checkSet() // Determine if a variable is set and is not NULL
{
    return isset($_POST["username"], $_POST["password"],
        $_POST["cPassword"],$_POST["firstName"],$_POST["lastName"],
        $_POST["phoneNumber"],$_POST["address"],$_POST["city"],
        $_POST["country"],$_POST["email"]);
}

```

```

//-----
-----//

```

```

if(checkSet()!==FALSE)
{
    if(sanityCheck($_POST["username"], 'string', 50) != true)
    {
        include("userRegistration.php");
        echo "<p>Παρακαλούμε εισάγετε username.</p>";
    }
    elseif (usedUsername($user)==true)
    {
        echo"<p>Παρακαλούμε διαλέξτε ένα άλλο.</p>";
    }
}

```

```

elseif(reservedName($user)==true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε διαλέξτε ένα άλλο username.</p>";
}
elseif(sanityCheck($_POST["password"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε password.</p>";
}
elseif(sanityCheck($_POST["cPassword"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Λανθασμένο password.Δοκιμάστε ξανά!</p>";
}

elseif(sanityCheck($_POST["firstName"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε το όνομα σας.</p>";
}
elseif(sanityCheck($_POST["lastName"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε το επιθετό σας.</p>";
}

elseif(sanityCheck($_POST["phoneNumber"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε το τηλέφωνό σας.</p>";
}

elseif(sanityCheck($_POST["address"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε τη διεύθυνση σας.</p>";
}

elseif(sanityCheck($_POST["city"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε τη πόλη διαμονής σας.</p>";
}

elseif(sanityCheck($_POST["country"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "<p>Παρακαλούμε εισάγετε τη χώρα διαμονής σας.</p>";
}

```

```

elseif(sanityCheck($_POST["email"], 'string', 50) != true)
{
    include("userRegistration.php");
    echo "Παρακαλούμε εισάγετε το email σας.";
}
elseif($_POST["cPassword"]!= $_POST["password"])
{
    include("userRegistration.php");
    echo"<p>Παρακαλούμε εισάγετε σωστό password</p>";
}
elseif(validEmail($_POST["email"])==false)
{
    include("userRegistration.php");
    echo"<p>Παρακαλούμε εισάγετε σωστό τύπο e-mail.</p>";
}

else
{
    $firstName=$_POST["firstName"];
    $lastName=$_POST["lastName"];
    $address=$_POST["address"];
    $city=$_POST["city"];
    $country=$_POST["country"];

    $password=sha1($_POST["password"]);

    $fName=iconv("ISO-8859-7","UTF-8",$firstName);
    $lName=iconv("ISO-8859-7","UTF-8",$lastName);
    $add=iconv("ISO-8859-7","UTF-8",$address);
    $ci=iconv("ISO-8859-7","UTF-8",$city);
    $scoun=iconv("ISO-8859-7","UTF-8",$country);

    mysql_query ('set character set utf8');
    $sqlCustomer="INSERT INTO customer (username, password,
    firstName, lastName,phoneNumber,address,city,country,email)
    VALUES ('".$_POST["username"]."', '$password', '$fName',
    '$lName','".$_POST["phoneNumber"]."', '$add', '$ci', '$scoun',
    '".$_POST["email"]."')";
    mysql_query($sqlCustomer,$con);
    include("loginForm.php");
    echo"<p>Η εγγραφή σας ολοκληρώθηκε με επιτυχία!</p>";
}
mysql_close($con);
}
?>

```

```

<html>
<head>
<title>Εγγραφή</title>
<meta content="text/html; charset=iso-8859-7">
<link rel="stylesheet" type="text/css" href="style.css">
<link rel="stylesheet" type="text/css" href="linkStyle.css">
<STYLE TYPE="text/css">
body
{
    background-image:url('/ptuxiakh/photos/image.jpg');
    background-position: 50% 50%;
    background-repeat: no-repeat;
    background-color:#590000;
    background-attachment:fixed;
}
td
{
    color:#FFFF33;
    font-weight: bold;
}
</STYLE>
</head>
<body>
</body>
</html>

```

Στο αρχείο αυτό αρχικά παρουσιάζονται οι συναρτήσεις οι οποίες χρησιμοποιούνται για τον έλεγχο των δεδομένων πριν τη καταχώρησή τους, συγκεκριμένα:

Η συνάρτηση **usedUsername** ελέγχει αν το username που επιθυμεί να έχει ο νέος χρήστης υπάρχει ήδη. Αν όντως υπάρχει επιστρέφει true και δεν γίνεται καταχώρηση στη βάση δεδομένων.

Η συνάρτηση **reservedName** αφορά τα ονόματα που έχουμε επιλέξει εμείς να μη καταχωρούνται στη βάση. Στη προκειμένη περίπτωση είναι το admin το οποίο θα μπορεί να το χρησιμοποιεί μόνο ο διαχειριστής της σελίδας για αυτό και δεν επιτρέπεται σε κανένα άλλο χρήστη να το κατοχυρώσει. Αν ο χρήστης προσπαθήσει να εισάγει ως username τη λέξη που δεσμεύσαμε, δηλαδή τη λέξη admin, η συνάρτηση θα επιστρέψει true και δε θα γίνει η καταχώρηση στη βάση δεδομένων.

Η συνάρτηση **validEmail** ελέγχει την ορθότητα καταχώρησης του email, δηλαδή αν διαθέτει τη κατάλληλη μορφή (περιλαμβάνει στο κείμενο @, τελεία κτλ) ώστε να είναι έγκυρο. Η συνάρτηση επιστρέφει true αν το κείμενο που δόθηκε αποτελεί email.

Η συνάρτηση **sanityCheck** δέχεται τρία ορίσματα: το \$string, που είναι η μεταβλητή που θα ελέγξει, ο τύπος \$type που αφορά το είδος της μεταβλητής (int, float, string κτλ) και το μήκος \$length της μεταβλητής. Έτσι αν ισχύουν τα κριτήρια που θα δώσουμε επιστρέφει true.

Τέλος η **checkSet** ελέγχει αν η μεταβλητή που δέχεται έχει οριστεί και δεν είναι μηδέν (NULL).

Μετά λοιπόν από τις συναρτήσεις γίνεται η κλήση τους και σύμφωνα με τις τιμές που δέχονται οι μεταβλητές μας και την εγκυρότητα τους γίνεται η εγγραφή του νέου πελάτη στη βάση μας ή εμφανίζεται ανάλογο μήνυμα σφάλματος.

Όπως μπορεί να παρατηρήσει κανείς το password πριν αποθηκευθεί στη βάση δεδομένων κωδικοποιείται μέσω της συνάρτησης **sha1 (Secure Hash Algorithm)** [7] η οποία παρέχει ένα επίπεδο ασφάλειας μέσω της κρυπτογράφησης ώστε οι κωδικοί πρόσβασης των πελατών να μην είναι εκτεθειμένοι στη βάση μας. Η sha1 έχει ως βασικό γνώρισμα την ιδιότητα να δίνει την ίδια κωδικοποίηση στην ίδια λέξη και όχι τη παραγωγή μιας ψευδοτυχαίας συμβολοσειράς κάθε φορά που καλείτε. Έτσι αν ο χρήστης πληκτρολογήσει απλά το password του μέσω της sha1 θα δοθεί το ίδιο αποτέλεσμα και θα πραγματοποιηθεί η ταυτοποίηση.

Αξίζει επίσης να σημειωθεί ότι η κωδικοποίηση των χαρακτήρων στη βάση έχει ρυθμιστεί να δέχεται χαρακτήρες utf8_general_ci, οπότε δεν μπορεί να αναγνωρίσει τους ελληνικούς χαρακτήρες που είναι τύπου ISO-8859-7. Έτσι με τη βοήθεια της **iconv** μετατρέπουμε τους χαρακτήρες που εισάγει ο χρήστης από το ISO-8859-7 σε utf8_general_ci ώστε να μπορέσουν να αποθηκευθούν σωστά στη βάση δεδομένων μας. Η σύνταξη της εντολής είναι η εξής:

iconv (string \$in_charset , string \$out_charset , string \$str) και δίνει αποτέλεσμα τύπου string [8].

Κώδικας loginForm.php

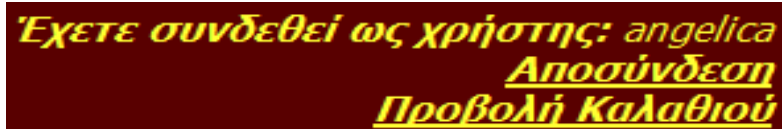
```
<?php
include("menu.php");
echo"<p>Παρακαλούμε συνδεθείτε για να ολοκληρώσετε τις αγορές σας.</p>";
echo"<form action=login.php' method= 'post'>";
echo"<table align='left' cellpadding='5'>";
echo"<tr><td></td>";
<td><a href='userRegistration.php'><b>Εγγραφή</b></a></td>";
echo"<tr><td><b>username</b></td>";
<td><input type='text' name='username'></td>";
echo"<tr><td> <b>password</b></td>";
<td><input type='password' name='password'></td>";
</tr>";
echo"<tr><td></td>";
<td align='center'><input type='submit' value='Σύνδεση'></td>";
</tr>";
echo"</table>";
echo"</form>";
?>
```

Το αρχείο αυτό περιέχει έναν πίνακα με τα πεδία συμπλήρωσης username και password για τη ταυτοποίηση του χρήστη στο σύστημα. Ουσιαστικά με το πάτημα του submit button αποστέλλονται τα στοιχεία στη φόρμα login.php όπου γίνεται και ο έλεγχος ορθής καταχώρης. Η φόρμα αυτή αναλύεται παρακάτω στη παράγραφο 3.2 «Σύνδεση χρήστη στο σύστημα».

3.2 Σύνδεση χρήστη στο σύστημα

Αν κάποιος χρήστης είναι εγγεγραμμένος στο σύστημα μπορεί να εισάγει το username του και το password του και να εισέλθει στο σύστημα ώστε να μπορεί να αγοράζει προϊόντα και να ανεβάζει φωτογραφίες (εικόνα 3).

Δηλαδή πατώντας το κουμπί «Σύνδεση», αν όντως έχει εισάγει σωστά τα στοιχεία του πραγματοποιείται η σύνδεση και του εμφανίζονται οι επιλογές «Αποσύνδεση» και «Προβολή Καλαθιού».



Εικόνα 3.2.1. Δυνατότητες χρήστη μετά την ταυτόποιηση του.

Ο κώδικας που δίνει αυτό το αποτέλεσμα βρίσκεται στο αρχείο **login.php** και παρουσιάζεται παρακάτω:

```
<?php
session_start();
?>
<html>
<head>
<title>Αρχική σελίδα</title>
<meta content="text/html; charset=iso-8859-7">
<link rel="stylesheet" type="text/css" href="linkStyle.css">
<link rel="stylesheet" type="text/css" href="style.css">
<style type="text/css">
body
{
    background-image:url('/ptuxiakh/photos/photoCamera.jpg');
    background-position: 53% -60%;
    background-repeat: no-repeat;
    background-color:#590000;
    background-attachment: fixed;
}
table
{
    color:#FFFF33;
    font-style:bold;
    font-family:tahoma;
}
</style>
</head>
<body>
</body>
</html>
```



```

<?php
if(isset($_POST['username']) && isset($_POST['password']))
{
    $user=$_POST['username'];
    $password=$_POST['password'];
}
else
{
    echo "Error.Try again.";
}

if(empty($_POST["username"])==true)
{
    include("loginForm.php");
    echo "<p>Παρακαλούμε εισάγετε το username.</p>";
    exit();
}
if(empty($_POST["password"])==true)
{
    include("loginForm.php");
    echo "<p>Παρακαλούμε εισάγετε το password.</p>";
    exit();
}
if($user=='admin' && $password=='admin')
{
    $_SESSION['username']=$user;
    include("adminPage.php");
}
else
{
    include("dbConnection.php");
    mysql_select_db("eshop",$con);
    $search = mysql_query("SELECT username,password,customerId FROM
customer where username='$user'");
    $row= mysql_fetch_array($search);

    if($row[0]==$user)
    {
        $pass=sha1($_POST['password']);
        if($row[1]==$pass)
        {
            $customerId=$row[2];
            $_SESSION['validUser']=$customerId;
            $_SESSION['flag']="customer";
            $_SESSION['username']=$user;

            if(isset($_SESSION['validUser']))
            {
                include("index.php");
            }
        }
    }
}

```

```

        else
        {
            echo "Error.Try again!";
        }
    }
    else
    {
        include("loginForm.php");
        echo "<p>Έχετε εισάγει λάθος password.Δοκιμάστε ξανά!
        </p>";
        exit();
    }
}
else
{
    include("loginForm.php");
    echo "<p>Δεν υπάρχει τέτοιος χρήστης.Δοκιμάστε ξανά!</p>";
}
}
?>

```

Με το αρχείο αυτό γίνεται ο έλεγχος ορθής καταχώρησης των στοιχείων του χρήστη, δηλαδή ελέγχεται αν το username υπάρχει στη βάση δεδομένων και αν ναι, τότε ελέγχεται αν το password που πληκτρολόγησε είναι ίδιο με αυτό που υπάρχει στη βάση για τον αντίστοιχο χρήστη. Μάλιστα για επιτευχθεί η αντιστοίχιση του password του χρήστη με το κωδικοποιημένο string που έχει αποθηκευθεί στη βάση δεδομένων χρησιμοποιείται πάλι η sha1 και έπειτα γίνεται ο έλεγχος. Αν τα δεδομένα είναι σωστά τότε δημιουργείται ένα session το οποίο θα μας βοηθήσει να μεταφέρουμε τα στοιχεία του χρήστη σε διάφορα άλλα τμήματα της σελίδας μας για να πραγματοποιηθούν οι ηλεκτρονικές αγορές. Αν δοθούν σαν είσοδο τα στοιχεία του administrator τότε ανοίγει η σελίδα διαχείρισης (adminPage) για να κάνει ο διαχειριστής τις όποιες ενέργειες επιθυμεί (η σελίδα του διαχειριστή θα αναλυθεί παρακάτω). Τέλος αν τα στοιχεία είναι λανθασμένα ο χρήστης επιστρέφει και πάλι στη φόρμα εισόδου.

3.3 Αποσύνδεση χρήστη από το σύστημα

Εάν ο χρήστης πατήσει πάνω στο σύνδεσμο «Αποσύνδεση» τότε παύει να είναι ταυτοποιημένος στο σύστημα και του εμφανίζεται ξανά η φόρμα συμπλήρωσης username και password.

Κώδικας logout.php

```
<?php
session_start();
unset($_SESSION['validUser']);
unset($_SESSION['flag']);
unset($_SESSION['username']);
session_destroy();
include("index.php");

?>
```

Με την «Αποσύνδεση» ουσιαστικά σβήνουμε τις τιμές των sessions που χρησιμοποιούμε και καταστρέφουμε το session που δημιουργήσαμε όταν έγινε η σύνδεση του χρήστη ώστε να χαθούν τα δεδομένα της ταυτοποίησής του. Με αυτό τον τρόπο διατηρείται το σύστημα ασφάλειας αφού μπορεί να εισέλθει ξανά ο ίδιος μόνο αν εισάγει τους κωδικούς του και όχι οποιοσδήποτε άλλος χρήστης βρεθεί στον ίδιο ηλεκτρονικό υπολογιστή.

4. Περιήγηση στις κατηγορίες του eshop

Ένας χρήστης είτε είναι ταυτοποιημένος είτε όχι μπορεί να δει τα προϊόντα που διαθέτει το ηλεκτρονικό κατάστημα, να διαβάσει επιπλέον πληροφορίες για αυτά και να τα επιλέξει πατώντας «Προσθήκη στο καλάθι».

Εδώ παρουσιάζονται τα προϊόντα που διαθέτει το κατάστημα στις κατηγορίες:

➤ Φωτογραφικές μηχανές

[Φωτογραφικές μηχανές](#) [Ψηφιακές Βιντεοκάμερες](#) [Ηλεκτρονικές Κορνίζες](#) [Προστατευτικές Θήκες](#) [Εκτύπωση φωτογραφιών](#)

» Προβολή ανά εταιρία παραγωγής« [» Προβολή ανά τιμή κόστους«](#) [» Προβολή ανά μοντέλο«](#)

 <p>» Canon- EOS 50D Τιμή Αγοράς: 1196.00€</p>	 <p>» Canon- A800 Τιμή Αγοράς: 123.00€</p>	 <p>» Nikon- COOLPIX L23 Τιμή Αγοράς: 69.00€</p>	 <p>» kodak- EASYSHARE M522 Τιμή Αγοράς: 99.00€</p>
 <p>» Nikon- COOLPIX S3100 Τιμή Αγοράς: 129.00€</p>	 <p>» Canon- POWERSHOT D10 Τιμή Αγοράς: 284.13€</p>		

Εικόνα 4.1. Φωτογραφικές μηχανές

➤ Ψηφιακές βιντεοκάμερες

[Φωτογραφικές μηχανές](#) [Ψηφιακές Βιντεοκάμερες](#) [Ηλεκτρονικές Κορνίζες](#) [Προστατευτικές Θήκες](#) [Εκτύπωση φωτογραφιών](#)

» Προβολή ανά εταιρία παραγωγής« [» Προβολή ανά τιμή κόστους«](#) [» Προβολή ανά μοντέλο«](#)

 <p>» Canon- LEGRIA HF R16 Τιμή Αγοράς: 460.00€</p>	 <p>» JVC- GZ-HD500 Τιμή Αγοράς: 398.00€</p>	 <p>» Aiptek- AHD H135 Τιμή Αγοράς: 89.00€</p>	 <p>» Canon- MD215 Τιμή Αγοράς: 175.03€</p>
 <p>» JVC- GZ-MS215B Τιμή Αγοράς: 199.00€</p>			

Εικόνα 4.2. Ψηφιακές βιντεοκάμερες

➤ Ηλεκτρονικές κορνίζες

φωτογραφικές μηχανές Ψηφιακές Βιντεοκάμες Ηλεκτρονικές Κορνίζες Προστατευτικές Θήκες Εκτύπωση φωτογραφιών

» Προβολή ανά εταιρία παραγωγής « » Προβολή ανά τιμή κόστους « » Προβολή ανά μοντέλο «

 <p>» Nilox- NX-160 Τιμή Αγοράς: 49.90€</p>	 <p>» Kodac- EASYSHARE P725 Τιμή Αγοράς: 57.81€</p>	 <p>» SIGMATEK- DN-1040 Τιμή Αγοράς: 89.90€</p>	 <p>» FujiFilm- V1 Τιμή Αγοράς: 299.00€</p>
 <p>» Telefunken- TFP75-200 Τιμή Αγοράς: 44.50€</p>			

Εικόνα 4.3. Ηλεκτρονικές κορνίζες

➤ Προστατευτικές θήκες

φωτογραφικές μηχανές Ψηφιακές Βιντεοκάμες Ηλεκτρονικές Κορνίζες Προστατευτικές Θήκες Εκτύπωση φωτογραφιών

» Προβολή ανά εταιρία παραγωγής « » Προβολή ανά τιμή κόστους « » Προβολή ανά μοντέλο «

 <p>» SONY- LCS-X20 Τιμή Αγοράς: 31.98€</p>	 <p>» Case Logic- TBC305K Τιμή Αγοράς: 20.91€</p>	 <p>» Hama- DV80 Τιμή Αγοράς: 24.60€</p>	 <p>» Olympus- 23453 Τιμή Αγοράς: 8.61€</p>
--	--	--	--

Εικόνα 4.4. Προστατευτικές θήκες

Ο κώδικας που χρησιμοποιήθηκε βρίσκεται στο αρχείο **showCategory.php**:

```
<?php
$fpa=0.23;
include("menu.php");
$categoryId=$_GET["categoryId"];
$value=$_GET["show"];
echo "<table cellpadding='5' align='center'>";
echo "<tr><td>
<a href='showCategory.php?categoryId=$categoryId&show=type'>
<b>&raquo;Προβολή ανά εταιρία παραγωγής&laquo;</b></a></td>";

echo "<td> <a href='showCategory.php?categoryId=$categoryId&show=price'>
<b>&raquo;Προβολή ανά τιμή κόστους&laquo;</b></a></td>";

echo "<td> <a href='showCategory.php?categoryId=$categoryId&show=image'>
<b>&raquo;Προβολή ανά μοντέλο&laquo;</b></a></td>";
echo "</table>";

include("dbConnection.php");
mysql_select_db("eshop",$con);
$search=mysql_query("SELECT image,type,productId,price FROM product WHERE
categoryId='$categoryId' ORDER BY $value");
$row=mysql_fetch_array($search);
$i=0;
while($row!="")
{
    $image[$i]=$row[0];
    $type[$i]=$row[1];
    $prId[$i]=$row[2];
    $number[$i]=$row[3]*$fpa+$row[3];
    $price[$i]=number_format($number[$i],2, '.', '');
    $i++;
    $row=mysql_fetch_array($search);
}
$end=$i;
$i=0;
echo "<table cellpadding='20' align='center'>";
while($i!=$end)
{
    echo "<td align='center'>
<a href='showProduct.php?productId=".$prId[$i]."' class='noStyle'> <img
src='/ptuxiakh/photos/".$image[$i].".jpg'/>
</a> <br><a href='showProduct.php?productId=".$prId[$i]."'
class='noStyle'><b>&raquo; ".$type[$i]."- ".$image[$i]."</b></a>
<br>Τιμή Αγοράς: <b>".$price[$i]."&euro;</b></td>";
    $i++;
    if($i==4)
    {
        echo "<tr></tr>";
    }
}
```

```

}
}
echo"</table>";
mysql_close($con);
?>

```

Στο αρχείο αυτό αξίζει να παρατηρήσουμε ότι οι τιμές των προϊόντων που έχουμε εισάγει στη βάση είναι υπολογισμένες χωρίς το Φ.Π.Α. ώστε σε μια περίπτωση αλλαγής της τιμής του να μη χρειάζεται να αλλαχθούν οι τιμές όλων των προϊόντων μία-μία αλλά να αλλάξουμε απλά τη τιμή της μεταβλητής \$fra (κόκκινα γράμματα). Επίσης, η προβολή των προϊόντων ανεξάρτητα της κατηγορίας τους αλλά και του κοινού χαρακτηριστικού τους (π.χ. Προβολή ανά εταιρία παραγωγής) εμφανίζεται πάντα από το συγκεκριμένο αρχείο απλά ξανακαλώντας το με τις τιμές που επιθυμούμε κάθε φορά. (μπλε γράμματα). Το ίδιο φυσικά συμβαίνει και για τη γενική προβολή των κατηγοριών όπου απλά είναι το ίδιο αρχείο, το showCaterogy δηλαδή, με ένα όρισμα στη μεταβλητή που αφορά τη κατηγορία ώστε να εμφανίζονται ξεχωριστά τα προϊόντα ανάλογα με την κατηγορία που έχουν εγγραφεί στη βάση δεδομένων μας.

Τέλος, η προβολή των προϊόντων γίνεται με ένα mysql_query χρησιμοποιώντας την εντολή SELECT από τον πίνακα product όπου βρίσκονται αποθηκευμένα όλα μας τα προϊόντα ανεξάρτητα από τη κατηγορία που ανήκουν.

5. Προβολή προϊόντος

[Αρχική Σελίδα](#)
[Φωτογραφικές μηχανές](#)
[Ψηφιακές βιντεοκάμερες](#)
[Ηλεκτρονικές Κορνίζες](#)
[Προστατευτικές θήκες](#)
[Εκτύπωση φωτογραφιών](#)
[Επικοινωνία](#)

Εικόνα



Πληροφορίες

Η νέα αδιάβροχη Canon D10 ενσωματώνει αισθητήρα των 12MPixels και 6 φακούς σε 5 ομάδες που συνεργάζονται για οξύτερα φωτογραφιών και οπτική μεγέθυνση 3x ώστε να έχετε το καλύτερο αποτέλεσμα. Η οθόνη προβολής και προεπισκόπησης 2,5 ιντσών προβάλλει κάθε μοναδική σας φωτογραφία, ενώ υποστηρίζεται και η εγγραφή βίντεο έως 640x480pixels με διάρκεια έως και 1 ώρα ανά κλιπ (με 4GB κάρτα). Η μοναδικότητα αυτής της ψηφιακής φωτογραφικής μηχανής εστιάζεται στην αντοχή στη σκόνη, στα χτυπήματα, τις ακραίες θερμοκρασίες (-10C έως 40C) και στη στεγανότητά της μέχρι βάθος 10 μέτρων. Όλες οι σκηνές, μέσα και έξω από το νερό, αποθηκεύονται με ασφάλεια σε κάρτες SD/SDHC, MMC, MMC Plus, HC MMC Plus, ενώ η επαναφορτιζόμενη μπαταρία εξασφαλίζει την απροβλημάτιστη χρήση για λήψη περίπου 220 φωτογραφιών.

Τύπος	Τιμή	Ποσότητα
Canon	284.13 €	<input style="width: 50px;" type="text" value="1"/>

Εικόνα 5. Προϊόν καταστήματος

Εάν ο χρήστης πατήσει πάνω σε κάποιο προϊόν οποιασδήποτε κατηγορίας θα του εμφανιστούν: η φωτογραφία του προϊόντος με τη δυνατότητα μεγέθυνσης (όπως φαίνεται παραπάνω), γενικές πληροφορίες για το συγκεκριμένο προϊόν, ο τύπος του

(μάρκα), η τιμή του (+ ΦΠΑ) καθώς και ένα text box με τη δυνατότητα αλλαγής της ποσότητας, ως προεπιλεγμένη τιμή είναι το 1 (ένα). Επίσης υπάρχει ένα κουμπί με ετικέτα «Προσθήκη στο καλάθι» το οποίο και επιβεβαιώνει ότι ο χρήστης επέλεξε το συγκεκριμένο προϊόν.

Κώδικας showProduct.php:

```
<html>
<head>
<title>e product</title>
<script type="text/javascript" src="mojoozoom.js"></script>
<link type="text/css" href="mojoozoom.css" rel="stylesheet"/>
<link rel="stylesheet" type="text/css" href="style.css">
<link rel="stylesheet" type="text/css" href="linkStyle.css">
<style TYPE="text/css">
body
{
    background-color:#590000;
}
img
{
    color: white;
    text-decoration:none;
}
</style>
</head>
<body>
<form action="InsertIntoBasket.php" method="get">
<?php
include("menu.php");
echo"<br>";
$fpa=0.23;
$productId=$_GET["productId"];
include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query ('set character set utf8');
$search=mysql_query("SELECT image,type,info,price,productId FROM product
WHERE productId=$productId");
$row=mysql_fetch_array($search);
echo"<table cellpadding='10' align='left' bgcolor='#993300'>";
echo"<tr>";
echo"<th>Εικόνα</th>";
echo"</tr>";
echo"<tr>";
echo"<td bgcolor='white'><img src='/ptuxiakh/photos/" . $row[0] . ".jpg' data-
zoomsrc='/ptuxiakh/photos/" . $row[0] . ".big.jpg' id='myimage'/></td>";
echo"</tr>";
echo"</table>";
echo"<table cellpadding='10' height='300' width='700' align='right'
bgcolor='#993300'>";
echo"<th>Πληροφορίες</th>";
```



```

$str=$row[2];
echo "<tr><td>".iconv("UTF-8","ISO-8859-7",$str)."</td></tr>";
echo"</table>";
echo"<table height='100' width='700' align='right' bgcolor='#993300'>";
echo"<tr align='center'>";
echo"<th>Τύπος</th>";
echo"<th>Τιμή</th>";
echo"<th>Ποσότητα</th>";
echo"</tr>";
echo"<tr align='center'>";
echo"<td>".$row[1]."</td>";
$number=$row[3]*$fpa+$row[3];
$price=number_format($number,2, '.', '');
echo"<td>$price &euro;</td>";
echo"<td><input type='text' name='quantity' value='1' size='7'></td>";
echo"<td><input type='hidden' name='productId' value='$productId'>";
echo"<td><input type='submit' value='Προσθήκη στο καλάθι'></td>";
echo"</tr>";
echo"</table>";
?>
</form>
</body>
</html>

```

Η προβολή της μεγεθυμένης εικόνας γίνεται με τη βοήθεια των αρχείων mojozoom.js και mojozoom.css. Για την ευκολότερη χρήση των φωτογραφιών οι δύο φωτογραφίες που χρησιμοποιούνται η μικρή και η μεγάλη έχουν το ίδιο όνομα αλλά η μεγάλη τελειώνει πάντα σε big.jpg όπως φαίνεται και παραπάνω.

Κώδικας mojozoom.js [9]

```

/*
 * MojoZoom 0.1.6 - JavaScript Image Zoom
 * Copyright (c) 2008 Jacob Seidelin, jseidelin@nihilogic.dk, http://blog.nihilogic.dk/
 * Licensed under the MPL License [http://www.nihilogic.dk/licenses/mpl-license.txt]
 */

var MojoZoom = (function() {
var $ = function(id) {return document.getElementById(id);};
var dc = function(tag) {return document.createElement(tag);};
var defaultWidth = 300;
var defaultHeight = 300;
function addEvent(element, ev, handler)
{
    var doHandler = function(e) {
        return handler(e||window.event);
    }
    if (element.addEventListener) {
        element.addEventListener(ev, doHandler, false);
    } else if (element.attachEvent) {

```

```

        element.attachEvent("on" + ev, doHandler);
    }
}
function getElementPos(element)
{
    var x = element.offsetLeft;
    var y = element.offsetTop;
    var parent = element.offsetParent;
    while (parent) {
        x += parent.offsetLeft - parent.scrollLeft;
        y += parent.offsetTop - parent.scrollTop;
        parent = parent.offsetParent;
    }
    return {
        x : x,
        y : y
    }
}

function getEventMousePos(element, e) {
    var scrollX = document.body.scrollLeft ||
document.documentElement.scrollLeft;
    var scrollY = document.body.scrollTop ||
document.documentElement.scrollTop;
    if (e.currentTarget) {
        var pos = getElementPos(element);
        return {
            x : e.clientX - pos.x + scrollX,
            y : e.clientY - pos.y + scrollY
        }
    }
    return {
        x : e.offsetX,
        y : e.offsetY
    }
}

function makeZoomable(img, zoomSrc, zoomImgCtr, zoomWidth, zoomHeight,
alwaysShow)
{
    // make sure the image is loaded, if not then add an onload event and return
    if (!img.complete && !img.__mojoZoomQueued) {
        addEvent(img, "load", function() {
            img.__mojoZoomQueued = true;
            setTimeout(function() {
                makeZoomable(img, zoomSrc, zoomImgCtr, zoomWidth,
                zoomHeight, alwaysShow);
            }, 1);
        });
    }
    return;
}

```

```

img.__mojoZoomQueued = false;

// Wrap everything in a timeout.
// this fixes a problem where, if makeZoomable is called a second time after changing
the src,
// FF would not have figured out the new offsetHeight of the image yet. A small
timeout helps though it's rather hackish.
setTimeout(function(){
// sorry
    var isIE = !!document.all && !!window.attachEvent && !window.opera;
    var w = img.offsetWidth;
    var h = img.offsetHeight;
    var oldParent = img.parentNode;
    if (oldParent.nodeName != "A") {
        var linkParent = dc("a");
        linkParent.setAttribute("href", zoomSrc);
        oldParent.replaceChild(linkParent, img);
        linkParent.appendChild(img);
    } else {
        var linkParent = oldParent;
    }

    linkParent.style.position = "relative";
    linkParent.style.display = "block";
    linkParent.style.width = w+"px";
    linkParent.style.height = h+"px";

    var imgLeft = img.offsetLeft;
    var imgTop = img.offsetTop;

    var zoom = dc("div");
    zoom.className = "mojozoom_marker";

    var zoomImg = dc("img");
    zoomImg.className = "mojozoom_img";

    zoomImg.style.position = "absolute";
    zoomImg.style.left = "-9999px";
    zoomImg.style.top = "-9999px";

    zoomImg.style.maxWidth = "none";
    zoomImg.style.maxHeight = "none";
    zoomImg.style.zIndex = 10000001;

    document.body.appendChild(zoomImg);
    var parent = img.parentNode;
    var ctr = dc("div");
    ctr.style.position = "absolute";
    ctr.style.left = imgLeft+"px";
    ctr.style.top = imgTop+"px";

```

```

ctr.style.width = w+"px";
ctr.style.height = h+"px";
ctr.style.overflow = "hidden";
ctr.style.display = "none";

ctr.appendChild(zoom);
parent.appendChild(ctr);

var zoomInput = parent;

// clear old overlay
if (img.__mojoZoomOverlay)
    parent.removeChild(img.__mojoZoomOverlay);
img.__mojoZoomOverlay = ctr;

// clear old high-res image
if (img.__mojoZoomImage && img.__mojoZoomImage.parentNode)

img.__mojoZoomImage.parentNode.removeChild(img.__mojoZoomImage);
img.__mojoZoomImage = zoomImg;

var useDefaultCtr = false;
if (!zoomImgCtr) {
    zoomImgCtr = dc("div");
    zoomImgCtr.className = "mojozoom_imgctr";

    var imgPos = getElementPos(img);
    zoomImgCtr.style.left = w + imgPos.x + "px";
    zoomImgCtr.style.top = imgPos.y + "px";

    zoomImgCtr.style.width = (zoomWidth ? zoomWidth :
    defaultWidth) + "px";
    zoomImgCtr.style.height = (zoomHeight ? zoomHeight :
    defaultHeight) + "px";

    document.body.appendChild(zoomImgCtr);
    useDefaultCtr = true;
}
zoomImgCtr.style.overflow = "hidden";

if (!alwaysShow) {
    zoomImgCtr.style.visibility = "hidden";
}

addEventListener(zoomImg, "load", function() {

// bail out if img has been removed from dom
if (!zoomImg.parentNode) return;

```

```

var zoomWidth = zoomImg.offsetWidth ? zoomImg.offsetWidth :
zoomImg.naturalWidth;
var zoomHeight = zoomImg.offsetHeight ? zoomImg.offsetHeight :
zoomImg.naturalHeight;
var ctrWidth = zoomImgCtr.offsetWidth;
var ctrHeight = zoomImgCtr.offsetHeight;

var ratioW = zoomWidth / w;
var ratioH = zoomHeight / h;

var markerWidth = Math.round(ctrWidth / ratioW);
var markerHeight = Math.round(ctrHeight / ratioH);
document.body.removeChild(zoomImg);
zoomImgCtr.appendChild(zoomImg);

var zoomFill = dc("div");
zoomFill.className = "mojozoom_fill";
zoom.appendChild(zoomFill);

var zoomBorder = dc("div");
zoomBorder.className = "mojozoom_border";
zoom.appendChild(zoomBorder);

zoom.style.width = markerWidth+"px";
zoom.style.height = markerHeight+"px";
if (alwaysShow) {
    zoom.style.left = "0px";
    zoom.style.top = "0px";

    zoomImg.style.left = "0px";
    zoomImg.style.top = "0px";
}

var isInImage = false;

if (!alwaysShow) {
    addEvent(zoomInput, "mouseout",
    function(e) {
        var target = e.target || e.srcElement;
        if (!target) return;
        if (target.nodeName != "DIV") return;
        var relTarget = e.relatedTarget || e.toElement;
        if (!relTarget) return;
        while (relTarget != target &&
        relTarget.nodeName != "BODY" &&
        relTarget.parentNode) {
            relTarget = relTarget.parentNode;
        }
        if (relTarget != target) {
            isInImage = false;
        }
    });
}

```

```

        ctr.style.display = "none";
        zoomImgCtr.style.visibility = "hidden";
    }
}
);
// a bit of a hack, mouseout is sometimes not caught if moving mouse really fast
addEvent(document.body, "mouseover",
function(e) {
    if (isInImage && !(e.toElement == zoomBorder || e.target == zoomBorder)) {
        ctr.style.display = "none";
        zoomImgCtr.style.visibility = "hidden";
        isInImage = false;
    }
}
);
}

addEvent(zoomInput, "mousemove",
function(e) {
    isInImage = true;
    var imgPos = getElementPos(img);

    if (useDefaultCtr) {
        zoomImgCtr.style.left = w + imgPos.x + "px";
        zoomImgCtr.style.top = imgPos.y + "px";
    }
    ctr.style.display = "block";
    zoomImgCtr.style.visibility = "visible";

    var pos = getEventMousePos(zoomInput, e);
    if (e.srcElement && isIE) {
        if (e.srcElement == zoom) return;
        if (e.srcElement != zoomInput) {
            var zoomImgPos = getElementPos(e.srcElement);
            pos.x -= (imgPos.x - zoomImgPos.x);
            pos.y -= (imgPos.y - zoomImgPos.y);
        }
    }
    var x = markerWidth/2;
    var y = markerHeight/2;
    if (!isIE) {
        pos.x -= imgLeft;
        pos.y -= imgTop;
    }

    if (pos.x < x) pos.x = x;
        if (pos.x > w-x) pos.x = w-x;
            if (pos.y < y) pos.y = y;
                if (pos.y > h-y) pos.y = h-y;

```

```

        var left = ((pos.x - x)|0);
        var top = ((pos.y - y)|0);

        zoom.style.left = left + "px";
        zoom.style.top = top + "px";

        zoomImg.style.left = -((pos.x*ratioW -
        ctrWidth/2)|0)+"px";
        zoomImg.style.top = -((pos.y*ratioH -
        ctrHeight/2)|0)+"px";
    }
    );
});

// I've no idea. Simply setting the src will make IE screw it self into a 100% CPU fest.
// In a timeout, it's ok.
setTimeout(function() {
    zoomImg.src = zoomSrc;
}, 1);

}, 1);
}
function init() {
    var images = document.getElementsByTagName("img");
    for (var i=0;i<images.length;i++) {
        var img = images[i];
        var zoomSrc = img.getAttribute("data-zoomsrc");
        if (zoomSrc) {
            makeZoomable(img, zoomSrc,
            document.getElementById(img.getAttribute("id") + "_zoom"),
            null, null, img.getAttribute("data-zoomalwaysshow")=="true");
        }
    }
}

return {
    addEvent : addEvent,
    init : init,
    makeZoomable : makeZoomable
};

})();

MojoZoom.addEvent(window, "load", MojoZoom.init);

```

Κώδικας mozoom.css

```
div.mozoom_marker
{
    position : absolute;
    cursor : crosshair;
}

div.mozoom_border
{
    position : absolute;
    left:0px;
    top:0px;

    border : 1px solid black;

    width:100%;
    height:100%;
}

div.mozoom_fill
{
    position : absolute;
    left:0px;
    top:0px;
    width:100%;
    height:100%;

    background-color : white;

    opacity : 0.2;
    -moz-opacity : 0.2;
    -khtml-opacity : 0.2;
    filter:alpha(opacity=20);
}

img.mozoom_img
{
    padding : 0px;
    border: 0px;
}

div.mozoom_imgctr
{
    position : absolute;
    margin-left : 20px;
    border : 1px solid black;
}
```


6. Καλάθι αγορών

Το καλάθι αγορών αποτελεί ουσιαστικά το ενδιάμεσο στάδιο μεταξύ της περιήγησης του χρήστη και της πραγματοποίησης των αγορών αφού στο καλάθι φυλάσσονται τα προϊόντα του κίνησαν το ενδιαφέρον και πιθανόν να τα αγοράσει.

6.1 Προσθήκη στο καλάθι

Αν ο χρήστης πατήσει το κουμπί «Προσθήκη στο καλάθι» όπως δείξαμε παραπάνω τότε τα δεδομένα θα αποσταλούν στη φόρμα InsertIntoBasket με τη μέθοδο get και ανάλογα με το αν ο χρήστης είναι ταυτοποιημένος στο σύστημα ή όχι θα εμφανιστεί και το ανάλογο παράθυρο. Συγκεκριμένα:

Αν ο χρήστης είναι ταυτοποιημένος στο σύστημα:

[Αρχική Σελίδα](#) [Φωτογραφικές μηχανές](#) [Ψηφιακές Βιν](#)

Περιεχόμενα του καλαθιού σας

Κατασκευαστής	Μοντέλο	Ποσότητα	Τιμή
Case Logic	TBC305K	1 Αλλαγή	20.91€ Διαγραφή
FujiFilm	V1	1 Αλλαγή	299.00€ Διαγραφή

Συνολικό κόστος αγορών: 319.91€
Άδειασμα καλαθιού

Τρόπος Πληρωμής:

Εικόνα 6.1.1 Μενού ταυτοποιημένου χρήστη

Αν ο χρήστης δεν είναι ταυτοποιημένος στο σύστημα

[Αρχική Σελίδα](#) [Φωτογραφικές μηχανές](#) [Ψηφιακές Βι](#)

Περιεχόμενα του καλαθιού σας

Κατασκευαστής	Μοντέλο	Ποσότητα	Τιμή
Telefunken	TFP75-200	1 Αλλαγή	44.50€ Διαγραφή
Canon	A800	1 Αλλαγή	123.00€ Διαγραφή

Συνολικό κόστος αγορών: 167.50€
Άδειασμα καλαθιού

Αγορά

Εικόνα 6.1.2 Μενού μη ταυτοποιημένου χρήστη

Κώδικας InsertIntoBasket.php:

```
<?php
session_start();

$fpa=0.23;
$quantity=$_GET["quantity"];
$prId=$_GET["productId"];
$scustId=$_SESSION['validUser'];

include("dbConnection.php");
mysql_select_db("eshop",$con);
$search=mysql_query("SELECT quantity FROM stock WHERE
productId=$prId");
$stockQuantity=mysql_fetch_array($search);

if($quantity<=0)
{
    include("showProduct.php");
    echo"<p>Έχετε εισάγει λανθασμένη ποσότητα! Δοκιμάστε ξανά.</p>";
    exit();
}
if($quantity>$stockQuantity[0])
{
    include("showProduct.php");
    echo"<p>Λυπούμαστε,δεν διαθέτουμε<b> ",$quantity," </b>τεμάχια.</p>";
    echo"<p>Παρακαλούμε εισαγάγετε ποσότητα μέχρι:<b> ",$stockQuantity[0],"
    </b>τεμάχια.</p><br><br>";
}
else
{
    if($_SESSION['flag']=="customer")
    {
        $now=date("H:i:s");
        mysql_query("DELETE FROM WHERE timeOfBuy<'$now'");
        $dateOfBuy=date("y-m-d");
        $time=date("H:i:s",mktime()+(15*60));
        $dateOfBuy=date("y-m-d");
        $sql="INSERT INTO customerbasket (productId, customerId,
        dateOfBuy, timeOfBuy,quantity) VALUES
        ('".$_GET["productId"]."',".$_SESSION["validUser"]."',
        '$dateOfBuy','$time','".$_GET["quantity"]."');";
        mysql_query($sql,$con);
        include("show.php");
    }
    else
    {
        $now=date("H:i:s");
```

```

mysql_query("DELETE FROM visitorbasket WHERE
timeOfBuy<'$now'");
$dateOfBuy=date("y-m-d");
$time = date("H:i:s",mktime()+(15*60));
$sql="INSERT INTO visitorbasket (productId, dateOfBuy,
timeOfBuy,quantity) VALUES ('".$_GET["productId"]."',
'$dateOfBuy','$time','".$_GET["quantity"]."');
mysql_query($sql,$con);
include("show.php");
}
}
mysql_close($con);

?>

```

Το αρχείο αυτό χρησιμεύει στην αποθήκευση των δεδομένων- επιλογών του χρήστη. Αρχικά γίνεται ένας έλεγχος εγκυρότητας όσον αφορά την ποσότητα που εισήγαγε ο χρήστης, δηλαδή τον έλεγχο της ως λογική τιμή (η ποσότητα είναι θετικό μέγεθος), αλλά και ως προς το απόθεμα του στοκ που υπάρχει στο κατάστημα για το συγκεκριμένο προϊόν που επέλεξε διότι μπορεί να ζητήσει περισσότερα τεμάχια από όσα διαθέτει το κατάστημα και να πραγματοποιηθεί μια λανθασμένη αγορά.

Επίσης το αρχείο αυτό έχει διπλή χρήση όσον αφορά την αποθήκευση των επιλογών του χρήστη αφού μπορεί και εξυπηρετεί ταυτοποιημένους και μη ταυτοποιημένους χρήστες. Έτσι, με τη βοήθεια της εντολής INSERT INTO τα στοιχεία εισάγονται είτε στο customerbasket αν ο χρήστης έχει κάνει login είτε στο visitorbasket αν ο χρήστης δεν έχει ταυτοποιηθεί. Για να μπορούν όμως να ξεχωρίζουν οι πρόσφατες από τις παλιές αγορές χρησιμοποιείται το πεδίο time όπου αποθηκεύει την ώρα που έγινε η επιλογή +15 λεπτά (κόκκινα γράμματα). Έτσι αν η τωρινή ώρα ξεπεράσει την στιγμή που έγινε η καταχώρηση (timeOfBuy<\$now), πρακτικά μετά από δεκαπέντε λεπτά, τότε οι εγγραφές αυτές διαγράφονται. Οπότε αν κάποιος χρήστης αποφασίσει να εγκαταλείψει τη σελίδα χωρίς να αδειάσει το καλάθι αγορών του ή χωρίς να ολοκληρώσει τις αγορές του, οι επιλογές του θα χαθούν μετά το πέρασμα δεκαπέντε λεπτών.

Όπως μπορεί κανείς να παρατηρήσει στο τέλος των if(\$_SESSION['flag']=="customer") και else υπάρχει μια εντολή include("show.php"). Το αρχείο αυτό είναι επίσης υπεύθυνο για αυτά που βλέπουμε αφού ουσιαστικά το InsertIntoBasket.php αποθηκεύει τα στοιχεία στη βάση και το show.php τα εμφανίζει στη σελίδα μας.

Κώδικας show.php:

```
<?php
session_start();
include("menu.php");
$custId=$_SESSION['validUser'];
$dateOfBuy=date("y-m-d");
$fpa=0.23;
include("dbConnection.php");
mysql_select_db("eshop",$con);
if($_SESSION['flag']=="customer")
{
    $visitor=mysql_query("SELECT productId,quantity,basketId FROM
    visitorbasket WHERE dateOfBuy='$dateOfBuy'");
    $visitorBasket= mysql_fetch_array($visitor);
    if($visitorBasket!="")
    {
        while($visitorBasket!="")
        {
            $sql="INSERT INTO customerbasket (customerId, productId,
            quantity,dateOfBuy) VALUES ('".$_SESSION['validUser']."',
            '$visitorBasket[0]','$visitorBasket[1]','$dateOfBuy)";
            mysql_query($sql,$con);
            $visitorBasket= mysql_fetch_array($visitor);
        }
    }
    mysql_query("DELETE LOW_PRIORITY FROM visitorbasket");
    $show=mysql_query("SELECT productId,quantity,customerbasketId FROM
    customerbasket WHERE dateOfBuy='$dateOfBuy' and
    customerId='$custId'");
    $row= mysql_fetch_array($show);

    if($row=="")
    {
        echo"<p>Το καλάθι αγορών σας είναι άδειο.</p>";
    }
    else
    {
        echo"<table border='1' cellpadding='5' bgcolor='#993300'>";
        echo"<caption><b>Περιεχόμενα του καλαθιού σας </b></caption>";
        echo"<tr>";
        echo"<th>Κατασκευαστής</th>";
        echo"<th>Μοντέλο</th>";
        echo"<th>Ποσότητα</th>";
        echo"<th>Τιμή</th>";
        echo"</tr>";
    }
}
```

```

while($row!="")
{
    $prdctId=$row[0];
    $quantity=$row[1];
    $customerbasketId=$row[2];
    $showProduct=mysql_query("SELECT type,image,price
    FROM product WHERE productId='$prdctId'");
    $productRow= mysql_fetch_array($showProduct);
    $price=$productRow[2]+$productRow[2]*$fpa;
    $finalPrice=number_format($price,2, '.', '');

    echo"<tr>";
    echo"<td>".$productRow[0]."</td>";
    echo"<td>".$productRow[1]."</td>";
    echo"<td align='center'>".$quantity."<br>
    <a class='noStyle' href='modifyQuantity.php?prId=$prdctId
    &custBasketId=$customerbasketId'> Αλλαγή</a></td>";
    echo"<td>".$finalPrice."&euro; <br>
    <a class='noStyle' href='deleteFromBasket.php?
    customerbasketId=$customerbasketId'>Διαγραφή</a> </td>";
    echo"</tr>";

    $sum=$sum+$quantity*$price;
    $productRow= mysql_fetch_array($showProduct);
    $row = mysql_fetch_array($show);
}
echo"</table>";
echo"<br><br>";
echo"</form>";

echo"<form action='buy.php' method='get'>";
echo"<table align='bottom' cellpadding='5' bgcolor='#993300'>";
$finalSum=number_format($sum,2, '.', '');
echo "<tr><td>Συνολικό κόστος αγορών:",$finalSum,"&euro; <br>
<a href='deleteFromBasket.php?customerbasketId=all'> <b>Αδειασμα
καλαθιού</b></a></td></tr>";
echo"<tr><td>Τρόπος Πληρωμής <select name='payWay'>
<option value='shop'>Παραλαβή από το κατάστημα</option>
<option value='delivery'>Με αντικαταβολή</option>
<option value='creditCard'>Πιστωτική κάρτα</option>
</select><b><br>";
echo"<input type='hidden' name='sum' value='".$finalSum.">";
echo"<input type='submit' value='Συνέχεια'></b></td></tr>";
echo"</form>";
}
}
else
{

```

```

$show=mysql_query("SELECT productId,quantity,basketId FROM
visitorbasket WHERE dateOfBuy='$dateOfBuy'");
$row= mysql_fetch_array($show);
if($row=="")
{
    echo"<p>Το καλάθι αγορών σας είναι άδειο.</p>";
}
else
{
    echo"<table border='1' cellpadding='5' bgcolor='#993300'>";
    echo"<caption><b>Περιεχόμενα του καλάθιού σας </b></caption>";
    echo"<tr>";
    echo"<th>Κατασκευαστής</th>";
    echo"<th>Μοντέλο</th>";
    echo"<th>Ποσότητα</th>";
    echo"<th>Τιμή</th>";
    echo"</tr>";
    while($row!="")
    {
        $prdctId=$row[0];
        $quantity=$row[1];
        $basketId=$row[2];
        $showProduct=mysql_query("SELECT type,image,price
FROM product WHERE productId='$prdctId'");
        $productRow= mysql_fetch_array($showProduct);

        $price=$productRow[2]+$productRow[2]*$fpa;
        $finalPrice=number_format($price,2, '.', '');

        echo"<tr>";
        echo"<td>".$productRow[0]."</td>";
        echo"<td>".$productRow[1]."</td>";
        echo"<td align='center'>".$quantity."<br> <a class='noStyle'
href='modifyQuantity.php?prId=$prdctId
&basketId=$basketId'>Αλλαγή</a></td>";
        echo"<td>".$finalPrice."&euro; <br> <a class='noStyle'
href='deleteFromBasket.php?basketId=$basketId'>
Διαγραφή</a></td>";
        echo"</tr>";

        $sum=$sum+$quantity*$price;
        $productRow= mysql_fetch_array($showProduct);
        $row = mysql_fetch_array($show);
    }
    echo"</table>";
    echo"<br><br>";

    echo"<table align='bottom' cellpadding='5' bgcolor='#993300'>";
    $finalSum=number_format($sum,2, '.', '');
    echo "<tr><td>Συνολικό κόστος αγορών:" , $finalSum, "&euro; <br>";

```

```

        <a href='deleteFromBasket.php?basketId=all'><b>'Αδειασμα
        καλαθιού</b></a></td></tr>";
        echo"<tr><td><a href='loginForm.php'><b>
        Αγορά</b></a></td></tr>";
    }
}

?>

```

Ουσιαστικά και εδώ με τη βοήθεια ενός session και μέσω μιας if συνθήκης ελέγχουμε αν ο χρήστης έχει κάνει login ή όχι και έπειτα με ένα SELECT από τον αντίστοιχο πίνακα που είναι καταχωρημένες οι εγγραφές μας γίνεται η εμφάνιση. Κάποια πεδία όπως το price (έντονα γράμματα), η τιμή δηλαδή, χρειάζονται επιπλέον υπολογισμό λόγω του κόστους ΦΠΑ για αυτό και πρώτα υπολογίζονται και έπειτα εμφανίζονται. Το ίδιο ισχύει και για το συνολικό κόστος αγορών (έντονα γράμματα) όπου και εκεί χρειάζεται να ληφθεί υπόψιν η ποσότητα του κάθε προϊόντος που έχει επιλέξει ο πελάτης. Επίσης εμφανίζονται οι υπόλοιπες δυνατότητες που έχει ο χρήστης. Δηλαδή: άδειασμα του καλαθιού του, αλλαγή της ποσότητας ή διαγραφή μεμονωμένου προϊόντος. Οι λειτουργίες αυτές εξηγούνται παρακάτω.

6.2 Διαγραφή από το καλάθι

Με τη δυνατότητα της διαγραφής από το καλάθι ένας χρήστης μπορεί να ακυρώσει είτε μια μεμονωμένη επιλογή του είτε όλες όσες έχει κάνει, ιδιότητα που του δίνει την ευελιξία αναίρεσης της επιλογής του και κάνει πιο εύκολη την αγορά των προϊόντων που τελικώς επιθυμεί να αγοράσει.

Κώδικας deleteFromBasket.php:

```

<?php
session_start();
include("dbConnection.php");
mysql_select_db("eshop",$con);
if($_SESSION['flag']=="customer")
{
    $custBasketId=$_GET["customerbasketId"];
    if($custBasketId=="all")
    {
        $dateOfBuy=date("y-m-d");
        $basket=mysql_query("DELETE FROM customerbasket WHERE
        dateOfBuy='$dateOfBuy'");
        include("menu.php");
        echo "<p>Επιτυχής διαγραφή προϊόντων.</p>";
    }
    else
    {
        $basket=mysql_query("DELETE FROM customerbasket WHERE
        customerbasketId=$custBasketId");
    }
}

```

```

        include("show.php");
    }
}
else
{
    $basketId=$_GET["basketId"];
    if($basketId=="all")
    {
        $dateOfBuy=date("y-m-d");
        $basket=mysql_query("DELETE FROM visitorbasket WHERE
        dateOfBuy='$dateOfBuy'");
        include("menu.php");
        echo "<p>Επιτυχής διαγραφή προϊόντων.</p>";
    }
    else
    {
        $basket=mysql_query("DELETE FROM visitorbasket WHERE
        basketId=$basketId");
        include("show.php");
    }
}
?>

```

Αν ανατρέξει κανείς στη δημιουργία της βάσης δεδομένων θα διαπιστώσει ότι οι πίνακες customerbasket και visitorbasket διαθέτουν ένα μοναδικό γνώρισμα, το κλειδί τους, το οποίο πρακτικά καθιστά ξεχωριστή κάθε εγγραφή, άρα αν ένας χρήστης θέλει να διαγράψει μια επιλογή του το μόνο που χρειάζεται να ξέρουμε είναι σε ποια εγγραφή-επιλογή αναφέρεται οπότε με τη βοήθεια μιας \$_GET[] μεταβλητής παίρνουμε το id της εγγραφής και με την εντολή DELETE FROM πίνακας WHERE ID=επιλεγμένη εγγραφή, πραγματοποιείται η διαγραφή μας.

Αν στη θέση του id δώσουμε την τιμή all (έντονα γράμματα), τότε διαγράφονται όλα τα προϊόντα του καλαθιού. Δηλαδή εκτελείται η λειτουργία «άδειασμα καλαθιού» (εικόνα 6.1.1 και 6.1.2) που αναφέραμε παραπάνω.

6.3 Αλλαγή ποσότητας στο καλάθι

Περιεχόμενα του καλαθιού σας			
Κατασκευαστής	Μοντέλο	Ποσότητα	Τιμή
Telefunken	TFP75-200	1 Αλλαγή	44.50€ Διαγραφή
Canon	A800	1 Αλλαγή	123.00€ Διαγραφή

Εικόνα 6.3.1 Περιεχόμενα καλαθιού

Αν ο χρήστης πατήσει πάνω στο link «αλλαγή» του αντίστοιχου προϊόντος που επιθυμεί να αλλάξει την ποσότητα θα του εμφανιστεί η φόρμα που βλέπουμε παρακάτω ώστε να εισάγει την ποσότητα που επιθυμεί να αγοράσει.

Παρακαλούμε εισάγετε την νέα ποσότητα.

Εικόνα 6.3.2 Αλλαγή ποσότητας

Έπειτα πατώντας το κουμπί «Αλλαγή» καταχωρείται η νέα ποσότητα στο σύστημα όπως φαίνεται παρακάτω.

Περιεχόμενα του καλαθιού σας			
Κατασκευαστής	Μοντέλο	Ποσότητα	Τιμή
Telefunken	TFP75-200	2 Αλλαγή	44.50€ Διαγραφή
Canon	A800	1 Αλλαγή	123.00€ Διαγραφή

Εικόνα 6.3.3 Περιεχόμενα καλαθιού μετά την αλλαγή ποσότητας

Για την υλοποίηση της αλλαγής ποσότητας χρησιμοποιήθηκαν τα αρχεία **modifyQuantity.php** και **updateQuantity.php** .

Κώδικας modifyQuantity.php:

```
<?php
session_start();
include("dbConnection.php");
mysql_select_db("eshop",$con);
include("menu.php");
$prdctId=$_GET["prId"];
```

```

echo"<form action='updateQuantity.php' method='get'>";
echo"<input type='hidden' name='prId' value='$prId'>";
if($_SESSION['flag']=="customer")
{
    $customerbasketId=$_GET["custBasketId"];
    echo"<input type='hidden' name='custBasketId' value='$customerbasketId'>";
}
else
{
    $basketId=$_GET["basketId"];
    echo"<input type='hidden' name='basketId' value='$basketId'>";
}
echo"<p>Παρακαλούμε εισάγετε την νέα ποσότητα.</p>";
echo"<input type='text' name='quantity' size='7'> <input type='submit'
value='Αλλαγή'>
?>

```

Κώδικας updateQuantity.php:

```

<?php
session_start();
include("dbConnection.php");
mysql_select_db("eshop",$con);

if (empty($_GET["quantity"])==true or ctype_digit($_GET["quantity"])==false)
{

    include("show.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης.</p>";
}
else
{

if($_SESSION['flag']=="customer")
{
    $search=mysql_query("SELECT quantity FROM stock WHERE
productId='".$_GET["prId"]."'");
$result=mysql_fetch_array($search);
if($result[0]>$_GET["quantity"])
    {
        mysql_query("UPDATE customerbasket SET
quantity='".$_GET["quantity"]."' WHERE
customerbasketId='".$_GET["custBasketId"]."'");
        include("show.php");
    }
    else
    {
        include("show.php");
    }
}
}

```

```

        echo"<p>Παρακαλούμε εισάγετε ποσότητα μέχρι ",
        $result[0]," τεμάχια.</p>";
    }
}
else
{
    $search=mysql_query("SELECT quantity FROM stock WHERE
productId='".$_GET["prId"]."");
$result=mysql_fetch_array($search);
if($result[0]>$_GET["quantity"])
    {
        mysql_query("UPDATE visitorbasket SET
quantity='".$_GET["quantity"]." WHERE
basketId='".$_GET["basketId"]."");
        include("show.php");
    }
    else
    {
        include("show.php");
        echo"<p>Παρακαλούμε εισάγετε ποσότητα μέχρι ",
        $result[0]," τεμάχια.</p>";
    }
}
}
?>

```

Ουσιαστικά μέσω της modifyQuantity ενεργοποιείται η updateQuantity όπου με τη βοήθεια μιας μεταβλητής τύπου GET παίρνουμε τη νέα ποσότητα. Όμως θα πρέπει να γνωρίζουμε σε ποιο προϊόν-εγγραφή του καλαθιού αναφέρεται η συγκεκριμένη αλλαγή και μάλιστα αν αφορά ταυτοποιημένο χρήστη ή όχι, οπότε με τη βοήθεια ενός input τύπου hidden για κάθε περίπτωση μεταφέρουμε την ταυτότητα id της εγγραφής που θα πρέπει να γίνει η αλλαγή. Στη συνέχεια ελέγχουμε αν η νέα ποσότητα δεν ξεπερνά την υπάρχουσα ποσότητα του στοκ (έντονα γράμματα) και έπειτα κάνουμε UPDATE στον αντίστοιχο πίνακα της βάσης δεδομένων μας ανάλογα με το αν το session με όνομα flag που ορίσαμε αρχικά μας δώσει τη τιμή customer ή όχι. Όταν λοιπόν επιλεχθεί και ο πίνακας, εγγράφεται η νέα ποσότητα.

7. Αγορά προϊόντων

7.1 Από ταυτοποιημένο χρήστη

Όπως είδαμε παραπάνω ένας χρήστης μπορεί να εισάγει προϊόντα στο καλάθι του πατώντας το κουμπί «Προσθήκη στο καλάθι» και να εκτελέσει διάφορες λειτουργίες όπως αναφέραμε παραπάνω. Τώρα, αν θέλει να ολοκληρώσει τις αγορές του πρέπει απλά να ολοκληρώσει τη διαδικασία μετά την προσθήκη στο καλάθι, δηλαδή να επιλέξει τρόπο πληρωμής και να συμπληρώσει κάποιες επιπλέον πληροφορίες (π.χ. αριθμό πιστωτικής ή διεύθυνση αποστολής) ώστε να γίνει η ολοκληρωμένη καταχώρηση στο σύστημα.



Παρακαλούμε συμπληρώστε τη διεύθυνση που επιθυμείτε να σταλούν τα προϊόντα σας.

Διεύθυνση:

Πόλη:

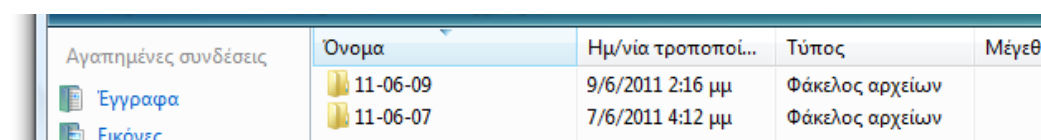
Εικόνα 7.1.1 Συμπλήρωση επιπλέον στοιχείων



Εικόνα 7.1.2 Μήνυμα επιβεβαίωσης

7.1.1 Report αγοράς

Όταν εμφανιστεί το μήνυμα «Η αγορά σας ολοκληρώθηκε με επιτυχία» ουσιαστικά έχει γίνει η αποθήκευση των δεδομένων-αγοράς στο πίνακα buy που χρησιμοποιείται για αυτό το σκοπό, και ταυτόχρονα εμφανίζεται ένα νέο report στο φάκελο report ώστε να γνωρίζει ο ιδιοκτήτης ότι πραγματοποιήθηκε μια αγορά. Το report αυτό περιέχει τα στοιχεία του χρήστη, τα προϊόντα, την ποσότητα και το συνολικό κόστος. Το report βρίσκεται σε ένα φάκελο με την ημερομηνία που πραγματοποιήθηκε η αγορά.



Αγαπημένες συνδέσεις	Όνομα	Ημ/νία τροποποι...	Τύπος	Μέγεθ
Εγγραφα	11-06-09	9/6/2011 2:16 μμ	Φάκελος αρχείων	
Εικόνες	11-06-07	7/6/2011 4:12 μμ	Φάκελος αρχείων	

Εικόνα 7.1.1.1 Δομή φακέλων

```
Πελάτης: Aggeliki Zografou 2310672345 agiwn pantwn 7 thes/niki ellada angele36@gmail.com
Αριθμός πιστωτικής: 16283940556
Προϊόν: FujiFilm V1
Ποσότητα: 1
Προϊόν: JVC GZ-HD500
Ποσότητα: 1
Συνολικό κόστος: 697.00
```

Εικόνα 7.1.1.2 Έγγραφο αγοράς

Κώδικας buy.php:

```
<?php
include("menu.php");
echo"<form action='shoppingReport.php' method='post'>";
if($_GET["payWay"]=="creditCard")
{
    echo"<p>Παρακαλούμε συμπληρώστε τον αριθμό της πιστωτικής σας κάρτας
για την ολοκλήρωση των αγορών σας.</p><br>";
    echo"<input type='hidden' name='payWay' value='creditCard'>";
    echo "<p>Αριθμός πιστωτικής:</p> <input type='text'
name='creditNumber'><br>";
    echo"<input type='submit' value='Ολοκλήρωση αγορών'>";
}
elseif($_GET["payWay"]=="delivery")
{
    echo"<p>Παρακαλούμε συμπληρώστε τη διεύθυνση που επιθυμείτε να
σταλούν τα προϊόντα σας.</p>";
    echo"<input type='hidden' name='payWay' value='delivery'>";
    echo"<p>Διεύθυνση: </p><input type='text' name='addressToSend'>";
    echo"<p>Πόλη: </p><input type='text' name='city'>";
    echo"<p><input type='submit' value='Ολοκλήρωση αγορών'></p>";
}
else
{
    echo"<input type='hidden' name='payWay' value='shop'>";
    echo"<p>Θα ενημερωθείτε τηλεφωνικά για την παραλαβή των προϊόντων
σας.<br>Ευχαριστούμε για την προτίμηση σας!</p>";
    echo"<p><input type='submit' value='Ολοκλήρωση αγορών'></p>";
}
$finalSum=$_GET["sum"];
echo"<input type='hidden' name='finalSum' value='$finalSum'>";
echo"</form>";
?>
```

Σε αυτή τη φόρμα προβάλλονται τα επιπλέον στοιχεία που πιθανόν χρειάζεται να συμπληρώσει ο χρήστης για να ολοκληρώσει τις αγορές του, ανάλογα πάντα με τον τρόπο πληρωμής που έχει επιλέξει για αυτό και βλέπουμε ότι τα πεδία εμφανίζονται ανάλογα με το ποια από τις if συνθήκες μας θα ισχύει (έντονα γράμματα). Η μεταβλητή που περιέχει τον τρόπο πληρωμής είναι η \$payWay η οποία στη συνέχεια

μεταφέρετε και στην επόμενη φόρμα για την ολοκλήρωση των αγορών με τη βοήθεια ενός input τύπου hidden ώστε να μην είναι ορατό στο χρήστη.

Κώδικας shoppingReport.php:

```
<?php
session_start();
include("menu.php");
include("dbConnection.php");
mysql_select_db("eshop",$con);
$custId=$_SESSION['validUser'];
$user=$_SESSION['username'];
$pay=$_POST["payWay"];
$totalPrice=$_POST["finalSum"];
if($_SESSION['flag']=="customer")
{
    $dateOfBuy=date("y-m-d");
    $reportDir ="c:/xampp/htdocs/ptuxiakh/report/$dateOfBuy";
    if (!file_exists($reportDir))
    {
        mkdir("c:/xampp/htdocs/ptuxiakh/report/$dateOfBuy", 0700);
    }

    $customerResult=mysql_query("SELECT firstName, lastName,
    phoneNumber,address,city,country,email FROM customer WHERE
    customerId='$custId'");
    $customer=mysql_fetch_array($customerResult);
    if($pay=="delivery")
    {
        $address=$_POST["addressToSend"];
        $city=$_POST["city"];
        if(empty($address))
        {
            echo"<p>Παρακαλούμε εισάγετε τη διεύθυνση
            αποστολής.</p>";
            exit();
        }
        if(empty($city) or ctype_alpha($city)==false)
        {
            echo"<p>Παρακαλούμε εισάγετε τη πόλη που θα σταλούν τα
            προϊόντα.</p>";
            exit();
        }
    }
    elseif($pay=="creditCard")
    {
        $number=$_POST["creditNumber"];
        if(empty($number) or is_numeric($number)==false)
        {
```

```

        echo"<p>Παρακαλούμε εισάγετε τον αριθμό πιστωτικής
        κάρτας.</p>";
        exit();
    }
}
else
{
    //the receiving from the store does not need control
}

$fileName="c:/xampp/htdocs/ptuxiakh/report/$dateOfBuy/" . $user . ".txt";
$fileHandle = fopen($fileName, 'a+') or die("can't open file");
fwrite($fileHandle, "\n Πελάτης: " . $customer[0]);
fwrite($fileHandle, "\t" . $customer[1]);
fwrite($fileHandle, "\t" . $customer[2]);
fwrite($fileHandle, "\t" . $customer[3]);
fwrite($fileHandle, "\t" . $customer[4]);
fwrite($fileHandle, "\t" . $customer[5]);
fwrite($fileHandle, "\t" . $customer[6]);
if($pay=="delivery")
{
    $address=$_POST["addressToSend"];
    $city=$_POST["city"];
    fwrite($fileHandle, "\n Διεύθυνση αποστολής: " . $address);
    fwrite($fileHandle, "\t" . $city);
}
elseif($pay=="creditCard")
{
    $number=$_POST["creditNumber"];
    fwrite($fileHandle, "\n Αριθμός πιστωτικής: " . $number);
}
else
{
    fwrite($fileHandle, "\n Παραλαβή από το κατάστημα.");
}

$basket=mysql_query("SELECT productId,quantity FROM customerbasket
WHERE dateOfBuy='$dateOfBuy' and customerId='$custId'");
$basketRow=mysql_fetch_array($basket);

while($basketRow!="")
{
    $buy="INSERT INTO buy (customerId, dateOfBuy, productId,
quantity, payWay, totalPrice) VALUES ('$custId', '$dateOfBuy',
'$basketRow[0]', '$basketRow[1]', '$pay', '$totalPrice')";
    mysql_query($buy,$con);
    $search = mysql_query("SELECT quantity FROM stock WHERE
productId='$basketRow[0]'");
    $stockQuantity = mysql_fetch_array($search);
}

```

```

$newStock=$stockQuantity[0]-$basketRow[1];
$stock=mysql_query("UPDATE stock SET quantity='$newStock'
WHERE productId='$basketRow[0]'");

$productResult=mysql_query("SELECT image,type FROM product
WHERE productId='$basketRow[0]'");
$product=mysql_fetch_array($productResult);

fwrite($fileHandle,"\n Προϊόν:". $product[1]);
fwrite($fileHandle,"\t" . $product[0]);
fwrite($fileHandle,"\n Ποσότητα: " . $basketRow[1]);

$basketRow = mysql_fetch_array($basket);
}
echo "<p>Η αγορά σας ολοκληρώθηκε με επιτυχία</p>";
$delete="DELETE FROM customerbasket WHERE
dateOfBuy='$dateOfBuy' and customerId='$custId'";
mysql_query($delete,$con);
echo"<b><p align='right'>Έχετε συνδεθεί ως χρήστης: <b>,$user,"</b> <a
href='logout.php'> Αποσύνδεση</a></p>";
fwrite($fileHandle,"\n Συνολικό κόστος: " . $totalPrice);
fwrite($fileHandle,"\n");
fclose($fileHandle);
}
else
{
    echo "error";
}
mysql_close($con);
?>

```

Το παραπάνω αρχείο είναι ουσιαστικά ο ελεγκτής για την ολοκλήρωση των αγορών του χρήστη. Αρχικά δημιουργεί το φάκελο με την ημερομηνία, με την εντολή mkdir (έντονα γράμματα). Η εντολή αυτή δέχεται ως παράμετρο έναν αριθμό που στη προκειμένη περίπτωση είναι το 0700. Ο αριθμός αυτός αφορά τα δικαιώματα που δίνονται σε αυτό το φάκελο και συγκεκριμένα ο αριθμός 0700 αναφέρεται στα default (προεπιλεγμένα) δικαιώματα. Σε περίπτωση που θα θέλαμε κάποια αλλαγή θα έπρεπε να χρησιμοποιήσουμε την εντολή chmod() [10].

Στη συνέχεια αφού έχει δημιουργηθεί ο φάκελος γίνεται έλεγχος στα στοιχεία που δόθηκαν. Αν δεν είναι σωστά εκτελείται η συνάρτηση exit() (έντονα γράμματα) ώστε να μη γίνει η εγγραφή λανθασμένων στοιχείων τόσο στη βάση όσο και στο report αγοράς. Αν όμως είναι σωστά τότε δημιουργείται το αρχείο στη μεταβλητή \$fileName (έντονα γράμματα) και έπειτα με την fwrite (έντονα γράμματα) εισάγουμε τα στοιχεία του report για κάθε προϊόν που περιέχει το καλάθι του χρήστη με τη βοήθεια ενός βρόχου while. Παράλληλα με ένα mysql_query εισάγουμε και τις εγγραφές- αγορές στο πίνακα buy της βάσης μας και επίσης ανανεώνουμε τα στοιχεία του πίνακα stock αφαιρώντας από την υπάρχουσα ποσότητα την ποσότητα που αγόρασε ο χρήστης. Όταν ολοκληρωθούν οι εγγραφές διαγράφονται τα περιεχόμενα του καλαθιού αγορών (DELETE FROM customerbasket WHERE –έντονα γράμματα)

και έπειτα εγγράφετε και το συνολικό ποσό αγορών στο αρχείο και με την fclose ολοκληρώνεται η εγγραφή του report.

7.2 Από μη ταυτοποιημένο χρήστη

Ο μη ταυτοποιημένος χρήστης έχει το δικαίωμα να εισάγει προϊόντα στο καλάθι αγορών του αλλά όχι να ολοκληρώσει τις αγορές του για αυτό και βλέπουμε ότι πατώντας το σύνδεσμο «Αγορά» του εμφανίζεται τη loginForm με το ανάλογο μήνυμα (εικόνα 3).



Κατασκευαστής	Μοντέλο	Ποσότητα	Τιμή
Canon	MD215	1 Αλλαγή	175.03€ Διαγραφή
Telefunken	TFP75-200	1 Αλλαγή	44.50€ Διαγραφή

Συνολικό κόστος αγορών: 219.53€
Άδειασμα καλαθιού
Αγορά

Εικόνα 7.2.1 Μενού επιλόγων μη ταυτοποιημένου χρήστη

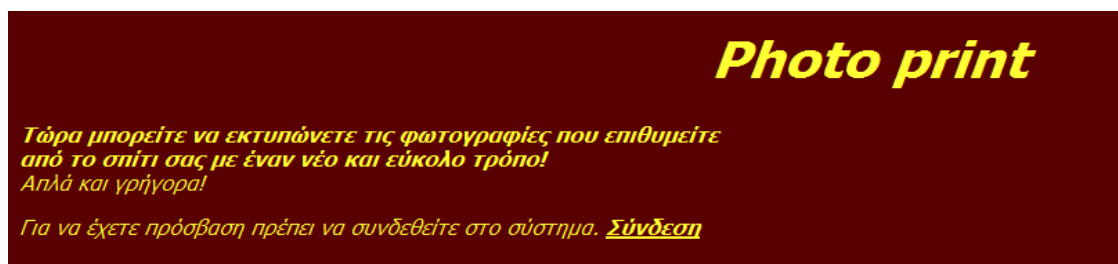
Ο χρήστης μπορεί είτε να εγγραφεί αν δεν είναι ήδη εγγεγραμμένος στο σύστημα είτε να εισάγει το login και το password του και να συνεχίσει τις αγορές του.

Μόλις ο χρήστης ταυτοποιηθεί στο σύστημα τότε μπορεί να δει τις προηγούμενες αγορές του πατώντας το σύνδεσμο «Προβολή καλαθιού» (εικόνα 3.2.1) και έπειτα να του εμφανιστεί το μενού επιλογών που έχει οριστεί για τους ταυτοποιημένους χρήστες ώστε να ολοκληρώσει τις αγορές του (εικόνα 6.1.1).

Τα αρχεία που χρησιμοποιήθηκαν είναι το loginForm.php, login.php, index.php και show.php τα οποία παρουσιάστηκαν και αναλύθηκαν παραπάνω.

8. Εκτύπωση φωτογραφιών

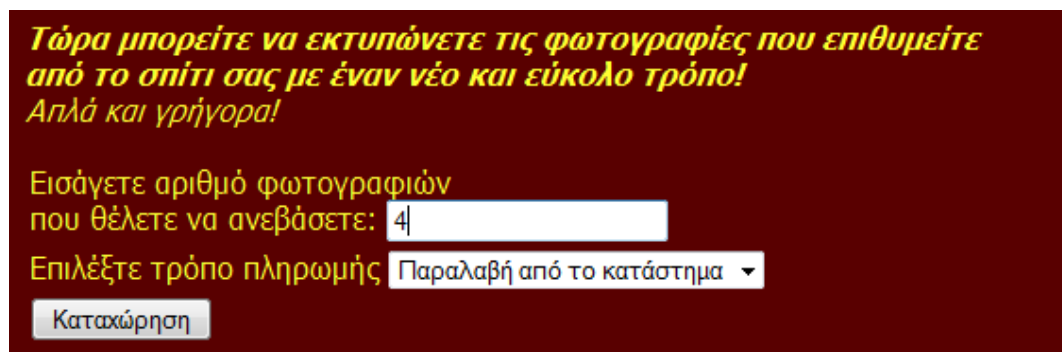
Για να εκτυπώσει κανείς τις φωτογραφίες που επιθυμεί θα πρέπει ως απαραίτητη προϋπόθεση να ταυτοποιηθεί στο σύστημα.



Εικόνα 8.1. Μήνυμα παρακίνησης για σύνδεση στο σύστημα

Αν λοιπόν εισέλθει στο σύστημα τότε:

- Επιλέγει τον αριθμό των φωτογραφιών που θέλει να ανεβάσει καθώς και τον τρόπο πληρωμής του. Έπειτα πατάει το κουμπί «καταχώρηση».



Εικόνα 8.2. Επιλογή αριθμού φωτογραφιών

- Με το κουμπί «αναζήτηση» επιλέγει ποια φωτογραφία επιθυμεί να ανεβάσει. Μπορεί επίσης για κάθε φωτογραφία ξεχωριστά να διαλέξει διαστάσεις, τύπο χαρτιού, και αριθμό αντιτύπων. Στη συνέχεια πατώντας το κουμπί «Upload» οι φωτογραφίες μαζί με το αντίστοιχο report εκτύπωσης ανεβαίνουν στο φάκελο του πελάτη.



Εικόνα 8.3. Συμπλήρωση στοιχείων φωτογραφίας

Upload:camera1.jpg
 Type:image/jpeg
 Size:61.9814453125 Kb
 Price:1.65€

Upload:fish.jpg
 Type:image/jpeg
 Size:16.3935546875 Kb
 Price:5.95€

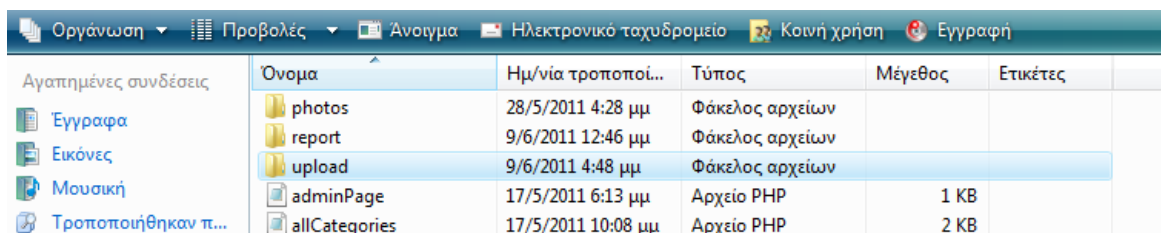
Upload:imageS.jpg
 Type:image/jpeg
 Size:8.724609375 Kb
 Price:1.8€

Συνολικό κόστος αγορών:10.1€

Εικόνα 8.4. Επιβεβαίωση αγοράς

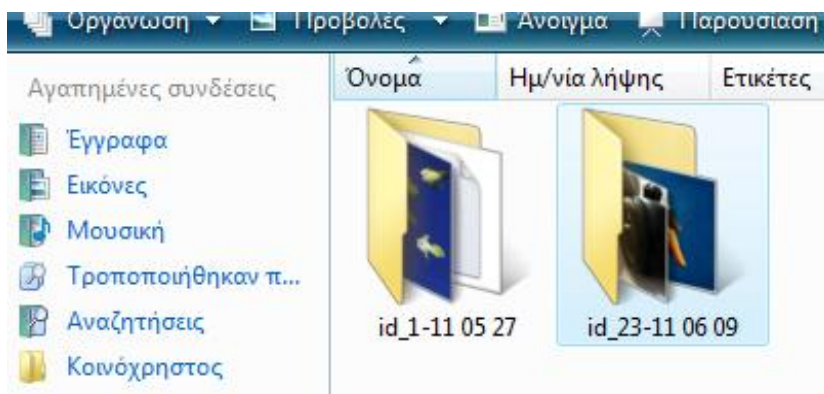
8.1 Report εκτύπωσης

- Ο φάκελος upload περιέχει υποφακέλους με τις φωτογραφίες και τα στοιχεία των πελατών που ανεβάζουν τις φωτογραφίες.



Εικόνα 8.1.1 Δομή φακέλων

- Οι υποφάκελοι δημιουργούνται δυναμικά και δέχονται ως όνομα το id του χρήστη και την ημερομηνία ώστε να ξεχωρίζουν και οι πελάτες αλλά και οι νέες από τις παλιές εκτυπώσεις.



Εικόνα 8.1.2. Δομή υποφακέλων

- Οι υποφάκελοι περιέχουν τις φωτογραφίες και το report εκτύπωσης.



Εικόνα 8.1.3. Περιεχόμενα υποφακέλων

- Το report εκτύπωσης περιέχει τα στοιχεία του πελάτη, αναλυτικά τις επιλογές του για κάθε φωτογραφία, το συνολικό ποσό που πρέπει να πληρώσει αλλά και τον τρόπο παραλαβής-πληρωμής των φωτογραφιών που ανέβασε.

```

2 Πελάτης: tzenh papoutsi 2310789087 filipoupolews 10 thes/niki Greece pap@gmail.com
3
4 Όνομα φωτογραφίας: art2.jpg
5 Διαστάσεις: 20x25
6 Τύπος χαρτιού: shine
7 Αντίτυπα: 1
8
9 Όνομα φωτογραφίας: camera1.jpg
10 Διαστάσεις: 15x21
11 Τύπος χαρτιού: mat
12 Αντίτυπα: 3
13
14 Όνομα φωτογραφίας: fish.jpg
15 Διαστάσεις: 13x18
16 Τύπος χαρτιού: mat
17 Αντίτυπα: 7
18
19 Όνομα φωτογραφίας: imageS.jpg
20 Διαστάσεις: 25x38
21 Τύπος χαρτιού: shine
22 Αντίτυπα: 2
23
24 Συνολικό ποσό:10.1 euro
25 Τρόπος πληρωμής:Παραλαβή από το κατάστημα
26 -----

```

Εικόνα 8.1.4. Έγγραφο εκτύπωσης

Τα αρχεία που χρησιμοποιήθηκαν για την υλοποίηση της on line εκτύπωσης είναι: onLinePrint.php,printMaker.php και uploader.php.

Κώδικας onLinePrint.php:

```
<?php
session_start();
include("menu.php");
echo"<html>";
echo"<head>";
    echo"<title>Εκτύπωση Φωτογραφιών</title>";
    echo"<meta content='text/html; charset=iso-8859-7'>";
    echo"<link rel='stylesheet' type='text/css' href='linkStyle.css'>";
    echo"<link rel='stylesheet' type='text/css' href='style.css'>";
    echo"<style type='text/css'>
    body
        {
            background-image:url('/ptuxiakh/photos/camera1.jpg');
            background-position:90% 80%;
            background-repeat:no-repeat;
            background-color:#590000;
            background-attachment: fixed;
        }";
    echo"</style>";
echo"</head>";
echo"<body>";
echo"<h1>Photo print</h1>";
echo"<p><b>Τώρα μπορείτε να εκτυπώνετε τις φωτογραφίες που επιθυμείτε
    <br>από το σπίτι σας με έναν νέο και εύκολο τρόπο!</b><br>
    <i>Απλά και γρήγορα!</i></p>";
if($_SESSION['flag']=="customer")
{
    echo"<form action='printMaker.php' method='get'>";
    echo"<table>";
    echo"<tr><td>Εισάγετε αριθμό φωτογραφιών <br>που θέλετε να ανεβάσετε:
    <input type='text' name='numberOfPhotos'></td></tr>";
    echo"<tr><td>Επιλέξτε τρόπο πληρωμής
    <select name='payWay'>
    <option value='shop'>Παραλαβή από το κατάστημα</option>
    <option value='delivery'>Με αντικαταβολή</option>
    <option value='creditCard'>Πιστωτική κάρτα</option>
    </select></td></tr>";
    echo"<tr><td><input type='submit' value='Καταχώρηση'></td></tr>";
    echo"</table>";
    echo"</form>";
}
else
{
    echo"<p>Για να έχετε πρόσβαση πρέπει να συνδεθείτε στο σύστημα.
    <a href='index.php'> Σύνδεση</a></p>";
```

```

}
echo"</body>";
echo"</html>";
?>

```

Αυτό το αρχείο υλοποιεί την αρχική μας φόρμα που ελέγχει αν ο χρήστης έχει εισέλθει στο σύστημα και αν ναι, τότε εμφανίζει το textbox για να εισάγει ο χρήστης τον αριθμό των φωτογραφιών που θέλει να ανεβάσει. Επίσης με ένα <select> ο χρήστης μπορεί να επιλέξει και τον τρόπο πληρωμής που επιθυμεί.

Κώδικας printMaker.php:

```

<?php
session_start();
$custId=$_SESSION['validUser'];

if (empty($_GET["numberOfPhotos"]))
{
    include("onLinePrint.php");
    echo"<p>Παρακαλούμε εισάγετε τον αριθμό των φωτογραφιών που
    επιθυμείτε να ανεβάσετε!</p>";

}
else
{

    $num=$_GET["numberOfPhotos"];
    $pay=$_GET["payWay"];

    if($num>50)
    {
        include("onLinePrint.php");
        echo"<p>Μπορείτε να ανεβάσετε μέχρι 50 φωτογραφίες τη
        φορά!</p>";
    }
    else
    {
        include("menu.php");
        echo"<form enctype='multipart/form-data' action='uploader.php'
        method='POST'>";
        echo"<p>Παρακαλούμε εισάγετε τις φωτογραφίες που θέλετε να
        ανεβάσετε.<br></p>";
        echo"<table>";
        for($i=0;$i<$num;$i++)
        {
            echo"<tr>";
            echo"<td><label for='file'>Αρχείο:</label>
            <input type='file' name='file$i'/><td>";
            echo"<td>Διάσταση φωτογραφίας:
            <select name='size$i'>

```

```

<option value='9x13'>9x13</option>
<option value='10x15'>10x15</option>
<option value='13x18'>10x18</option>
<option value='15x21'>15x21</option>
<option value='20x25'>20x25</option>
<option value='20x30'>20x30</option>
<option value='24x30'>24x30</option>
<option value='25x38'>25x38</option>
</select></td>";
echo"<td>Τύπος χαρτιού:
<select name='quality$i'>
<option value='shine'>Γυαλιστερό</option>
<option value='mat'>Ματ</option>
</select></td>";
echo"<td>Αριθμός αντιτύπων
<input type='text' name='copies$i' value='1'></td>";
echo"</tr>";

```

```

}

```

```

echo"<input type='hidden' name='number' value='$num' />";
echo"<input type='hidden' name='payway' value='$pay' />";
echo"</table>";
echo"<br>";
echo"<table>";

```

```

if($pay=="delivery")
{
    echo"<tr><td>Διεύθυνση αποστολής:<input type='text'
name='addressToSend'></td></tr>";
}
elseif($pay=="creditCard")
{
    echo"<tr><td>Παρακαλούμε εισάγετε αριθμό πιστωτικής:
<input type='text' name='creditNumber'></td></tr>";
}
else
{
    echo"<tr><td>Οι φωτογραφίες σας θα είναι έτοιμες σε 24 ώρες
μετά την αποστολή σας.</td></tr>";
}
echo"<tr><td><input type='submit' value='Upload' /></td></tr>";
echo"</table>";

```

```

}

```

```

}
?>

```

Αξίζει να παρατηρήσουμε ότι η δημιουργία των files μαζί με τα υπόλοιπα στοιχεία των επιλογών δημιουργούνται δυναμικά μέσα σε ένα βρόχο for (έντονα γράμματα) ο οποίος δέχεται ως τιμή τερματισμού την τιμή που έδωσε ο χρήστης όταν ρωτήθηκε πόσες φωτογραφίες θέλει να ανεβάσει. Έτσι δημιουργούνται τα αντίστοιχα <input type='file' > (έντονα γράμματα) για κάθε μια φωτογραφία. Μάλιστα επειδή τα επιπλέον στοιχεία για την εκτύπωση πρέπει να γίνονται με τέτοιο τρόπο ώστε να γνωρίζουμε σε ποια από τις φωτογραφίες αναφέρονται δίνεται και το αντίστοιχο όνομα σε κάθε <input> tag και <select>tag που χρησιμοποιούμε ώστε να ξέρουμε σε ποιο file αναφέρεται η αντίστοιχη μεταβλητή copies,size κτλ (έντονα γράμματα).

Κώδικας uploader.php:

```
<?php
session_start();
$custId=$_SESSION['validUser'];
$username=$_SESSION['username'];
include("menu.php");
$n=$_POST["number"];
$pay=$_POST["payway"];
$sum=0;

if($_SESSION['flag']=="customer")
{
    echo"<p align='right'><b>Έχετε συνδεθεί ως χρήστης:</b>",$username,"<br>
    <a href='logout.php'> Αποσύνδεση</a></p>";
    if($pay=="delivery")
    {
        $address=$_POST["addressToSend"];
        if(empty($address))
        {
            echo"<p>Παρακαλούμε εισάγετε τη διεύθυνση
            αποστολής.</p>";
            exit();
        }
    }
    elseif($pay=="creditCard")
    {
        $number=$_POST["creditNumber"];
        if(empty($number))
        {
            echo"<p>Παρακαλούμε εισάγετε τον αριθμό πιστωτικής
            κάρτας.</p>";
            exit();
        }
    }
    else
    {
        //no control
    }
    $date=date("y m d");
```



```

$uploadsDir = "c:/xampp/htdocs/ptuxiakh/upload/id_ $custId-$date";
if (!file_exists($uploadsDir))
{
    mkdir("c:/xampp/htdocs/ptuxiakh/upload/id_ $custId-$date", 0700);
}
include("dbConnection.php");
mysql_select_db("eshop", $con);
$customerResult=mysql_query("SELECT firstName, lastName,
phoneNumber, address, city,country,email FROM customer WHERE
customerId=' $custId'");
$customer=mysql_fetch_array($customerResult);

$fileName="c:/xampp/htdocs/ptuxiakh/upload/id_ $custId-
$date/". $username.".txt";

$fileHandle = fopen($fileName, 'a+') or die("can't open file");
fwrite($fileHandle, "\nΠελάτης: ". $customer[0]);
fwrite($fileHandle, "\t". $customer[1]);
fwrite($fileHandle, "\t". $customer[2]);
fwrite($fileHandle, "\t". $customer[3]);
fwrite($fileHandle, "\t". $customer[4]);
fwrite($fileHandle, "\t". $customer[5]);
fwrite($fileHandle, "\t". $customer[6]);
fwrite($fileHandle, "\n");

for($i=0;$i<$n;$i++)
{
    if ((($_FILES["file$i"]["type"] == "image/gif")||
($_FILES["file$i"]["type"] == "image/jpeg")||
($_FILES["file$i"]["type"] == "image/pjpeg"))
&& ($_FILES["file$i"]["size"]<90000) &&
isset($_FILES["file$i"]["type"])==true)
    {
        if ($_FILES["file$i"]["error"]>0)
        {
            echo "Return Code:". $_FILES["file$i"]["error"].
            "<br />";
        }
        else
        {

            $size=$_POST["size$i"];
            if($size=="9x13")
            {
                $sizePrice=0.20;
            }
            elseif($size=="10x15")
            {

```

```

        $sizePrice=0.30;
    }
elseif($size=="10x18")
{
    $sizePrice=0.40;
}
elseif($size=="15x21")
{
    $sizePrice=0.50;
}
elseif($size=="20x25")
{
    $sizePrice=0.60;
}
elseif($size=="20x30")
{
    $sizePrice=0.70;
}
else
{
    $sizePrice=0.80;
}

$quality=$_POST["quality$i"];
if($quality=="shine")
{
    $paperPrice=0.10;
}
else
{
    $paperPrice=0.05;
}
$copies=$_POST["copies$i"];
if($copies<=0)
{
    echo"<p>Παρακαλούμε εισάγετε σωστό αριθμό
    αντιτύπων.</p>";
}
else
{
    $photoPrice=($paperPrice+$sizePrice)*$copies;

    if(file_exists
    ("$_uploadsDir/" . $_FILES["file$i"]["name"]))
    {
        echo "<p><br>Το αρχείο
        " . $_FILES["file$i"]["name"] . " Υπάρχει
        ήδη.<br>";
    }
}

```

```

        Παρακαλούμε επιλέξτε άλλο αρχείο.</p>";
    }
    else
    {
        fwrite($fileHandle,"\nΌνομα φωτογραφίας:
        ".$_FILES["file$i"]["name"]);
        fwrite($fileHandle,"\nΔιαστάσεις: ".$size);
        fwrite($fileHandle,"\nΤύπος χαρτιού:
        ".$quality);
        fwrite($fileHandle,"\nΑντίτυπα: ".$copies);
        fwrite($fileHandle,"\n");
        echo"<table>";
        echo "<tr><td>
        Upload:".$_FILES["file$i"]["name"]."</td>
        </tr>";
        echo "<tr><td>
        Type:".$_FILES["file$i"]["type"]."</td> </tr>";
        echo "<tr><td>
        Size:".($_FILES["file$i"]["size"]/1024)." Kb
        </td></tr>";
        echo "<tr><td>
        Price:".$photoPrice."&euro;</td></tr>";
        echo"<tr></tr><tr></tr>";
        echo"</table>";
        $sum=$sum+$photoPrice;
        move_uploaded_file
        ($_FILES["file$i"]["tmp_name"],"$uploadsDir/"
        .$_FILES["file$i"]["name"]);
    }
}
}
}
else
{
    echo "<p><br>Το αρχείο ".$_FILES["file$i"]["name"]." δεν είναι
    κατάλληλης μορφής ή είναι πολύ μεγάλο.<br>
    Παρακαλούμε επιλέξτε άλλο αρχείο.</p>";
}
}

if(0!=filesize($fileName))
{
    echo"<p>Συνολικό κόστος αγορών:".$sum."&euro;</p>";
    fwrite($fileHandle,"\nΣυνολικό ποσό:".$sum." euro");
}

if(0!=filesize($fileName))
{

```

```

if($pay=="delivery")
{
    $address=$_POST["addressToSend"];
    fwrite($fileHandle,"\nΤρόπος πληρωμής:Με αντικαταβολή\n\n");
    fwrite($fileHandle,"\nΤόπος αποστολής: ".$address."\n\n");
}
elseif($pay=="creditCard")
{
    $number=$_POST["creditNumber"];
    fwrite($fileHandle,"\nΤρόπος πληρωμής:Με πιστωτική κάρτα\n\n");
    fwrite($fileHandle,"\nΑριθμός πιστωτικής: ".$number."\n\n");
}
else
{
    fwrite($fileHandle,"\nΤρόπος πληρωμής:Παραλαβή από το
    κατάστημα\n\n");
}
}
fwrite($fileHandle,"-----\n");
fclose($fileHandle);
}
else
{
    echo"Παρακαλούμε συνδεθείτε για να μπορέσετε να ανεβάσετε τις
    φωτογραφίες που επιθυμείτε.";
}
?>

```

Αρχικά όπως μπορεί κανείς να παρατηρήσει γίνονται οι έλεγχοι ορθής συμπλήρωσης των πεδίων πληρωμής. Αν δεν είναι σωστά συμπληρωμένα τότε η διαδικασία ανεβάσματος σταματάει με τη βοήθεια της συνάρτησης **exit()**. Αν είναι σωστά συμπληρωμένα τότε δημιουργείται ο φάκελος του πελάτη με τη βοήθεια της **mkdir**, έπειτα γίνεται έλεγχος για τον τύπο του αρχείου που ανέβασε ο χρήστης και το μέγεθος του και στη συνέχεια ελέγχεται αν υπάρχει ήδη το αρχείο που επιθυμεί να ανεβάσει με τη βοήθεια της συνάρτησης **file_exists**. Η συνάρτηση αυτή συντάσσεται ως εξής: **bool file_exists (string \$filename)** και το αποτέλεσμα που δίνει είναι λογικού τύπου δηλαδή true ή false [9].

Αν ισχύουν οι παραπάνω προϋποθέσεις τότε εγγράφονται οι επιλογές του χρήστη στο report εκτύπωσης και το αρχείο μετακινείται από τον temp φάκελο που βρίσκονταν στο φάκελο που αρχικά δημιουργήσαμε. Αυτό γίνεται με τη βοήθεια της **move_uploaded_file**. Τέλος εγγράφεται και το συνολικό ποσό πληρωμής καθώς και ο τρόπος πληρωμής με τη βοήθεια της εντολής **fwrite** και με την **fclose** ολοκληρώνεται το report και η διαδικασία εκτύπωσης φωτογραφιών.

9. Επικοινωνία με το κατάστημα

Η φόρμα αυτή δημιουργήθηκε για την προβολή των στοιχείων του καταστήματος ώστε αν θελήσει ένας πελάτης να βρει τη διεύθυνση του καταστήματος ή να τηλεφωνήσει να μπορεί να το κάνει απλά με ένα «κλικ» στην ιστοσελίδα και όχι να χρειαστεί να ψάξει σε τηλεφωνικούς καταλόγους.



Εικόνα 9.1.1 Στοιχεία καταστήματος

Το αρχείο που χρησιμοποιήθηκε είναι το contact.php.

Κώδικας contact.php:

```
<html>
<head>
<title>Επικοινωνία</title>
<meta content="text/html; charset=iso-8859-7">
<link rel="stylesheet" type="text/css" href="linkStyle.css">
<link rel="stylesheet" type="text/css" href="style.css">
<style type="text/css">
body
{
    background-image:url('/ptuxiakh/photos/photoCamera.jpg');
    background-position: 53% -140%;
    background-repeat: no-repeat;
    background-color:#590000;
}
h1
{
    color:#FFFF33;
    font-size:40px;
    font-weight:bold;
    font-style:italic;
    font-family:tahoma;
    text-decoration:blink;
    position: relative;
    top: 0px;
```

```

        left: 500px;
    }
    p
    {
        color:#FFFF33;
        font-style:italic;
        font-family:tahoma;
    }
</style>
</head>

<body>
<?php include("menu.php"); ?>
<h1>Photo print</h1>
<p>
Δίνει νέα διάσταση στο κόσμο της φωτογραφίας! <br>
Ανδριανού 33, Θεσ/νικη <br>
Τηλ: 2310-000000 <br><br>
Με συνέπεια και επαγγελματισμό δίπλα σας!
</p>
</body>
</html>

```

Στο παραπάνω αρχείο παρουσιάζει ενδιαφέρον το εφέ κίνησης που διαθέτει η επικεφαλίδα h1. Με την ιδιότητα του text-decoration, blink [6] το κείμενο φαίνεται σαν να αναβοσβήνει πράγμα που τραβάει αρκετά τη προσοχή του χρήστη και χρησιμοποιείται συχνά σε ιστοσελίδες παρόμοιου περιεχομένου. Συνήθως συναντάται σε προσφορές ή σε καινούργιες παραλαβές των ηλεκτρονικών καταστημάτων ώστε να κεντρίσει το ενδιαφέρον του πελάτη. Στη συγκεκριμένη περίπτωση χρησιμοποιήθηκε στην επωνυμία του καταστήματος για να τη θυμάται πιο εύκολα ο πελάτης αφού υπάρχουν περισσότερες πιθανότητες να διαβάσει ένα κείμενο που αναβοσβήνει.

10. Σελίδα διαχειριστή

Η χρησιμότητα της σελίδας του διαχειριστή αφορά αποκλειστικά το κατάστημα που προβάλλεται στην ιστοσελίδα και δεν δίνει καμία δυνατότητα στο χρήστη, για την ακρίβεια ο χρήστης δεν γνωρίζει καν την ύπαρξη του αφού οι λειτουργίες που εκτελούνται σχετίζονται αποκλειστικά και μόνο με την επιχείρηση. Έτσι, η σελίδα αυτή περιέχει τα στοιχεία που αφορούν το κατάστημα, δηλαδή τα προϊόντα, το στοκ, τους πελάτες κτλ και φυσικά λειτουργίες αλλαγής, προσθήκης και διαγραφής. Η είσοδος στην ιστοσελίδα αυτή μπορεί να γίνει μόνο την πληκτρολόγηση συγκεκριμένου username και password στη loginForm (εικόνα 3) που είναι το admin admin.

Μετά την επιτυχή είσοδο εμφανίζεται το παρακάτω μενού: πελατολόγιο, προϊόντα, κατηγορίες. Κάθε μια από τις τρεις κατηγορίες αναλύονται παρακάτω.

10.1 Πελατολόγιο



Όνομα	Επώνυμο	Τηλέφωνο	Διεύθυνση	Πόλη	Χώρα	e-mail	Αλλαγή	Διαγραφή
Aggeliki	Zografou	2310672345	agiwn pantwn 7	thes/niki	ellada	angele36@gmail.com	Αλλαγή	Διαγραφή
Αλέξανδρος	Παπαδιαμάντης	2310736451	Αγίας Τριάδος 12	Θεσ/νικη	Ελλάδα	blabla@hotmail.com	Αλλαγή	Διαγραφή
vaso	Zografou	2310789087	anakshmandrou 27	Athina	germania	vickyzograf@gmail.com	Αλλαγή	Διαγραφή
tzenh	papoutsi	2310789087	filiroupolews 10	thes/niki	Greece	pap@gmail.com	Αλλαγή	Διαγραφή
Αλέξανδρος	Papadopoulos	2310789087	Αγίας Τριάδος 12	Θεσ/νικη	Ελλάδα	pap@gmail.com	Αλλαγή	Διαγραφή
nino	ok	2109922136	filiroupolews 10	thes/niki	ellada	blabla@hotmail.com	Αλλαγή	Διαγραφή
Αντώνιος	Μπαντερίδης	2109922136	Συρρακουσών 23	Αθήνα	Ελλάδα	blabla@hotmail.com	Αλλαγή	Διαγραφή

Εικόνα 10.1.1 Προβολή πελατολογίου

Με την καρτέλα «πελατολόγιο» ο διαχειριστής μπορεί να βλέπει ποιοι είναι οι πελάτες του, δηλαδή ποιοι είναι καταχωρημένοι στο σύστημα. Τα δικαιώματα που έχει αφορούν την αλλαγή των στοιχείων του πελάτη, όλων εκτός του username και password που είναι προσωπικά και δεν χρειάζεται να έχει πρόσβαση σε αυτά. Επίσης έχει το δικαίωμα να διαγράψει κάποιον πελάτη.

Κώδικας allCustomers.php:

```
<?php
include("adminPage.php");
include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query('set character set utf8');
$showCustomer = mysql_query("SELECT firstName, lastName, phoneNumber,
address, city, country, email,customerId FROM customer ");
$scustRow= mysql_fetch_array($showCustomer);
echo"<table border='1'cellpadding='5' bgcolor='#993300'>";
echo"<tr>";
echo"<th>Όνομα</th>";
```

```

echo"<th>Επώνυμο</th>";
echo"<th>Τηλέφωνο</th>";
echo"<th>Διεύθυνση</th>";
echo"<th>Πόλη</th>";
echo"<th>Χώρα</th>";
echo"<th>e-mail</th>";
echo"</tr>";
while($custRow!="")
{
    echo"<tr>";
    $firstName=$custRow[0];
    $lastName=$custRow[1];
    $address=$custRow[3];
    $city=$custRow[4];
    $country=$custRow[5];
    $cId=$custRow[7];

    $fName=iconv("UTF-8","ISO-8859-7",$firstName);
    $lName=iconv("UTF-8","ISO-8859-7",$lastName);
    $add=iconv("UTF-8","ISO-8859-7",$address);
    $ci=iconv("UTF-8","ISO-8859-7",$city);
    $coun=iconv("UTF-8","ISO-8859-7",$country);

    echo"<td>".$fName."</td>";
    echo"<td>".$lName."</td>";
    echo"<td>".$custRow[2]."</td>";
    echo"<td>".$add."</td>";
    echo"<td>".$ci."</td>";
    echo"<td>".$coun."</td>";
    echo"<td>".$custRow[6]."</td>";
    echo"<td><a class='noStyle'
href='modifyCustomer.php?customerId=$cId'>Αλλαγή</a></td>";
    echo"<td><a class='noStyle'
href='deleteFromCustomer.php?customerId=$cId'>Διαγραφή</a> </td>";
    $custRow= mysql_fetch_array($showCustomer);
    echo"</tr>";
}
echo"</table>";
mysql_close($con);

?>

```

Η σελίδα αυτή με τη βοήθεια ενός SELECT παίρνει τα στοιχεία των πελατών από τον πίνακα customer της βάσης δεδομένων μας και τα εμφανίζει. Για να γίνει σωστή εμφάνιση των στοιχείων που αποθηκεύθηκαν στη βάση με κωδικοποίηση utf8 χρησιμοποιείται και πάλι η iconv [8]όμως αυτή τη φορά ζητούμε να γίνει από utf8 σε ISO-8859-7 και όχι το αντίθετο που κάναμε όταν ήταν να εισάγουμε τα στοιχεία μας στο πίνακα.

10.1.1 Αλλαγή στοιχείων πελάτη

Είναι πιθανό στη διάρκεια του χρόνου κάποια στοιχεία του πελάτη να μεταβληθούν (π.χ. διεύθυνση, τηλέφωνο κτλ) για αυτό και ο διαχειριστής έχει το δικαίωμα αλλαγής του. Τα αρχεία που χρησιμοποιήθηκαν για αυτή τη λειτουργία παρουσιάζονται παρακάτω.

Κώδικας modifyCustomer.php:

```
<?php
include("adminPage.php");
$customerId=$_GET["customerId"];
echo"<html>";
echo"<head>";
echo"<meta content='text/html; charset=iso-8859-7'>";
echo"</head>";
echo"<body>";
echo"<form action='updateCustomer.php' method='get'>";
echo"<table border = '0' align='left' WIDTH='450'>";
echo"<caption><b>&raquo;&raquo;&Aλλαγή στοιχείων πελάτη</b></caption>";
echo"<tr>";
echo"<input type='hidden' name='customerId' value=$customerId>";
echo"<td>Παρακαλούμε επιλέξτε το πεδίο που θέλετε να αλλάξετε</td>";
echo"<td><select name='field'>";
echo"<option value='firstName'>Όνομα</option>";
echo"<option value='lastName'>Επώνυμο</option>";
echo"<option value='phoneNumber'>Τηλέφωνο</option>";
echo"<option value='address'>Διεύθυνση</option>";
echo"<option value='city'>Πόλη</option>";
echo"<option value='country'>Χώρα</option>";
echo"<option value='email'>email</option>";
echo"</select></td>";
echo"</tr>";
echo"<tr>";
echo"<td>Νέα τιμή:</td>";
echo"<td><input type='text' name='value'></td>";
echo"</tr>";
echo"<tr>";
echo"<td></td>";
echo"<td align='center'><input type='submit' value='Καταχώρηση'></td>";
echo"</tr>";
echo"</table>";
echo"</form>";
?>
```

Με τη φόρμα αυτή ο διαχειριστής επιλέγει σε ποιο από τα πεδία θέλει να πραγματοποιήσει την αλλαγή με τη βοήθεια ενός <select>tag. Σε ποιο χρήστη θα γίνει αλλαγή το γνωρίζουμε ήδη με το customerId του οποίου πήραμε την τιμή με τη

βοήθεια της \$_GET (έντονα γράμματα). Στη συνέχεια με το πάτημα του submit button τα δεδομένα αποστέλλονται στην updateCustomer όπου ενημερώνεται η βάση δεδομένων για την αλλαγή.

Κώδικας updateCustomer.php:

```
<?php
function validEmail($mailAddress)
{
    if(ereg('^[a-zA-Z0-9\.\_\-]+@([a-zA-Z0-9][a-zA-Z0-9\-\]*\.)+[a-zA-Z]+$', $mailAddress))
    {
        return true;
    }
    else
    {
        return false;
    }
}
//-----
include("dbConnection.php");
mysql_select_db("eshop", $con);
mysql_query('set character set utf8');
$cust=mysql_query("SELECT customerId FROM customer WHERE customerId='".$_GET["customerId"]."");
$custId=mysql_fetch_array($cust);

if($custId[0]==$_GET["customerId"])
{
    if($_GET["field"]=="firstName")
    {
        if (empty($_GET["value"])==true or
ctype_alpha($_GET["value"])==false)
        {

            include("modifyCustomer.php");
            echo "<br><br>";
            echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
            στο κελί <b>Νέα τιμή</b><br></p>";
        }
        else
        {
            $str=$_GET["value"];
            $fName=iconv("ISO-8859-7", "UTF-8", $str);
            mysql_query("UPDATE customer SET firstName='".$fName.'"
            WHERE customerId='".$_GET["customerId"]."");
            include("allCustomers.php");
            echo "<p>Επιτυχής αλλαγή ονόματος!</p>";
        }
    }
}
```

```

}
elseif($_GET["field"]=="lastName")
{
    if(empty($_GET["value"])==true or
    ctype_alpha($_GET["value"])==false)
    {
        include("modifyCustomer.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        $str=$_GET["value"];
        $IName=iconv("ISO-8859-7","UTF-8",$str);
        mysql_query("UPDATE customer SET lastName='".$IName.'"
        WHERE customerId='".$_GET["customerId"].'");
        include("allCustomers.php");
        echo "<p>Επιτυχής αλλαγή επωνύμου!</p>";
    }
}
elseif($_GET["field"]=="phoneNumber")
{
    if (empty($_GET["value"])==true or
    is_numeric($_GET["value"])==false)
    {
        include("modifyCustomer.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        mysql_query("UPDATE customer SET
        phoneNumber='".$_GET["value"].'" WHERE
        customerId='".$_GET["customerId"].'");
        include("allCustomers.php");
        echo "<p>Επιτυχής αλλαγή τηλεφώνου!</p>";
    }
}
elseif($_GET["field"]=="address")
{
    if(empty($_GET["value"])==true)
    {
        include("modifyCustomer.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
}

```

```

    }
else
{
    $str=$_GET["value"];
    $new=iconv("ISO-8859-7","UTF-8",$str);
    mysql_query("UPDATE customer SET address='$new'
    WHERE customerId='".$_GET["customerId"]."'");
    include("allCustomers.php");
    echo "<p>Επιτυχής αλλαγή διεύθυνσης!</p>";
}
}
elseif($_GET["field"]=="city")
{
    if(empty($_GET["value"])==true or
    ctype_alpha($_GET["value"])==false)
    {
        include("modifyCustomer.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
else
{
    $str=$_GET["value"];
    $new=iconv("ISO-8859-7","UTF-8",$str);
    mysql_query("UPDATE customer SET city='".$_new.'"
    WHERE customerId='".$_GET["customerId"]."'");
    include("allCustomers.php");
    echo "<p>Επιτυχής αλλαγή πόλης!</p>";
}
}
elseif($_GET["field"]=="country")
{
    if(empty($_GET["value"])==true or
    ctype_alpha($_GET["value"])==false)
    {
        include("modifyCustomer.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
else
{
    $str=$_GET["value"];
    $new=iconv("ISO-8859-7","UTF-8",$str);
    mysql_query("UPDATE customer SET country='".$_new.'"
    WHERE customerId='".$_GET["customerId"]."'");
    include("allCustomers.php");
    echo "<p>Επιτυχής αλλαγή χώρας!</p>";
}
}

```

```

    }
elseif($_GET["field"]=="email")
{
    if (empty($_GET["value"])==true or
        validEmail($_GET["value"])==false)
    {
        include("modifyCustomer.php");
        echo "<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        mysql_query("UPDATE customer SET
        email='".$_GET["value"]." WHERE
        customerId='".$_GET["customerId"]."");
        include("allCustomers.php");
        echo "<p>Επιτυχής αλλαγή email!</p>";
    }
}
else
{
    echo"Error.";
}
}
else
{
    echo"<b>Δεν υπάρχει ο πελάτης. Δοκιμάστε ξανά.</b>";
}
?>

```

Όπως μπορεί κανείς να παρατηρήσει το αρχείο αυτό κάνει τους απαραίτητους ελέγχους για την ορθή καταχώρηση των δεδομένων που εισήγαγε ο διαχειριστής και αν είναι σωστά τότε ενημερώνεται ο πίνακας customer της βάσης μας για την αλλαγή. Για τον έλεγχο των τιμών χρησιμοποιήθηκε η συνάρτηση validUser που έχει ήδη αναλυθεί καθώς και οι συναρτήσεις ctype_alpha και is_numeric (έντονα γράμματα)[8]. Η συνάρτηση **ctype_alpha** (string \$text) ελέγχει αν η μεταβλητή που δόθηκε αποτελείται μόνο από γράμματα και δίνει ως έξοδο true ή false αφού είναι τύπου bool. Αντίθετα η συνάρτηση is_numeric επιστρέφει true μόνο όταν όλοι οι χαρακτήρες που περιέχει η μεταβλητή είναι αριθμοί. Η σύνταξη της εντολής είναι: bool **is_numeric** ([mixed](#) \$var). Επίσης αξίζει να σημειωθεί ότι αντίστοιχη λειτουργία έχει και η ctype_digit η οποία συντάσσεται: bool **ctype_digit** (string \$text) με τη διαφορά ότι αυτή δέχεται μόνο μεταβλητές τύπου string και όχι όλων των ειδών που διαθέτει η php (mixed) όπως η is_numeric. Στη συγκεκριμένη περίπτωση βέβαια θα μπορούσαν να χρησιμοποιηθούν και οι δύο συναρτήσεις.

10.1.2 Διαγραφή πελάτη

Με τη λειτουργία αυτή ο διαχειριστής μπορεί να διαγράψει κάποιο χρήστη-πελάτη που πλέον δεν πραγματοποιεί αγορές.

Κώδικας deleteFromCustomer.php:

```
<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);

$cust=mysql_query("SELECT customerId FROM customer WHERE
customerId='".$_GET["customerId"]."");
$custId=mysql_fetch_array($cust);

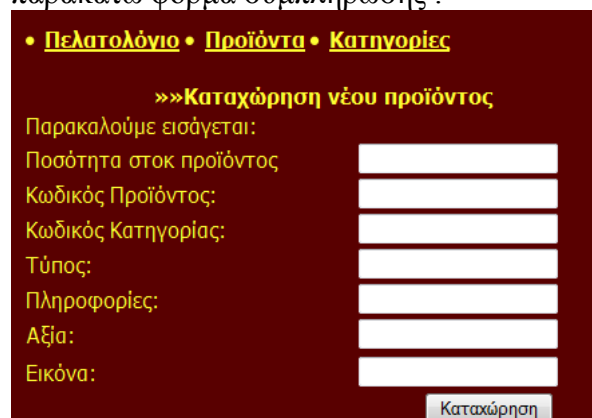
if($custId[0]==$_GET["customerId"])
{
    mysql_query("DELETE FROM customer WHERE
customerId='".$_GET["customerId"]."");
    include("allcustomers.php");
    echo "<p>Επιτυχής διαγραφή πελάτη.</p>";
}
?>
```

10.2 Προϊόντα

Τα προϊόντα αποτελούν το πιο βασικό κομμάτι της επιχείρησης για αυτό και θα πρέπει το μενού του διαχειριστή να είναι αρκετά εύχρηστο ώστε να μπορεί να ανανεώνει το ηλεκτρονικό του κατάστημα.

10.2.1 Καταχώρηση νέου προϊόντος

Όταν ο χρήστης πατήσει το κουμπί καταχώρηση νέου προϊόντος του εμφανίζεται η παρακάτω φόρμα συμπλήρωσης .



• Πελατολόγιο • Προϊόντα • Κατηγορίες

»»Καταχώρηση νέου προϊόντος

Παρακαλούμε εισάγεται:

Ποσότητα στοκ προϊόντος:

Κωδικός Προϊόντος:

Κωδικός Κατηγορίας:

Τύπος:

Πληροφορίες:

Αξία:

Εικόνα:

Καταχώρηση

Εικόνα 10.2.1.1. Φόρμα καταχώρησης προϊόντος

Κάθε προϊόν διαθέτει και την αντίστοιχη ποσότητα στοκ του καταστήματος για αυτό δεν αρκεί μόνο η συμπλήρωση των γενικών χαρακτηριστικών του αλλά χρειάζεται και η ποσότητα στοκ όπως φαίνεται στην εικόνα. Το αρχείο που χρησιμοποιήθηκε είναι το newProduct.php

Κώδικας newProduct.php:

```
<?php include("adminPage.php"); ?>
<html>
<head>
<meta content="text/html; charset=iso-8859-7">
</head>
<body>
<form action="insertIntoProduct.php" method="get">
<table border="0" align="left" WIDTH="450">
<caption><b>&raquo;&raquo;Καταχώρηση νέου προϊόντος</b></caption>
<tr><td>Παρακαλούμε εισάγεται:</td></tr>
<tr><td>Ποσότητα στοκ προϊόντος</td>
<td><input type="text" name="quantity"></td></tr>
<tr><td>Κωδικός Προϊόντος:</td>
<td><input type="text" name="productId"></td></tr>
<tr><td>Κωδικός Κατηγορίας:</td>
<td><input type="text" name="categoryId"></td></tr>
<tr><td>Τύπος:</td>
<td><input type="text" name="type"></td></tr>
<tr><td>Πληροφορίες:</td>
<td><input type="text" name="info"></td></tr>
<tr><td>Αξία:</td>
<td><input type="text" name="price"></td></tr>
<tr><td>Εικόνα:</td>
<td><input type="text" name="image"></td></tr>
<tr><td></td>
<td align="center"><input type="submit" value="Καταχώρηση"></td></tr>
</table>
</form>
</body>
</html>
```

Το αρχείο αυτό ενεργοποιεί τη φόρμα insertIntoProduct.php με την οποία γίνεται η καταγραφή του νέου προϊόντος στη βάση δεδομένων, αν τα δεδομένα που δώσει ο διαχειριστής είναι σωστά. Το αρχείο παρουσιάζεται παρακάτω.

Κώδικας insertIntoProduct.php:

```
<?php
function sanityCheck($string, $type, $length)
{
    $type = 'is_'.$type;
    if(!$type($string))                //check if the right data type
    {
        return FALSE;
    }
    elseif(strlen($string) > $length)  // then we check how long the string is
    {
        return FALSE;
    }
    else
    {
        return TRUE; // if all is well, we return TRUE
    }
}

include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query ('set character set utf8');
if (empty($_GET["quantity"])==true || ctype_digit($_GET["quantity"])==false)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
    <b>Ποσότητα στοκ προϊόντος</b></p>";
}
elseif (empty($_GET["productId"])==true ||
ctype_digit($_GET["productId"])==false)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
    <b>Κωδικός Προϊόντος</b></p>";
}
elseif (empty($_GET["categoryId"])==true ||
ctype_digit($_GET["categoryId"])==false)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
    <b>Κωδικός Κατηγορίας</b></p>";
}
elseif(empty($_GET["type"])==true && sanityCheck($_GET["type"], 'string', 50) !=
true)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
    <b>Τύπος</b></p>";
}
```



```

elseif(empty($_GET["info"])==true && sanityCheck($_GET["info"], 'string', 50) !=
true)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
<b>Πληροφορίες</b></p>";
}

elseif(empty($_GET["price"])==true || ctype_digit($_GET["price"])==false)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
<b>Αξία</b></p>";
}

elseif(empty($_GET["image"])==true && sanityCheck($_GET["image"], 'string', 50)
!= true)
{
    include("newProduct.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
<b>Εικόνα</b></p>";
}
else
{
    $stock="INSERT INTO stock(productId,quantity) VALUES
('".$_GET["productId"]."',".$_GET["quantity"].'");
    mysql_query($stock,$con);
    $info=$_GET["info"];
    $str=iconv("ISO-8859-7","UTF-8",$info);
    mysql_query ('set character set utf8');
    $product="INSERT INTO product
(productId,categoryId,type,info,price,image) VALUES
('".$_GET["productId"]."',".$_GET["categoryId"]."',".$_GET["type"]."',
'$str','".$_GET["price"]."',".$_GET["image"].'");
    mysql_query ('set character set utf8');
    mysql_query($product,$con);
    include("allproducts.php");
    echo"<p>Επιτυχής καταχώρηση προϊόντος.</p>";
    //mysql_close($con);
}
?>

```

10.2.2 Αλλαγή στοιχείων προϊόντος

Όταν ο διαχειριστής πατήσει στο μενού «Προϊόντα» του εμφανίζονται όλα τα προϊόντα που είναι καταχωρημένα στη βάση δεδομένων μας με τις δυνατότητες αλλαγής και διαγραφής όπως φαίνεται στην εικόνα 10.2.2.1.

• [Πελατολόγιο](#) • [Προϊόντα](#) • [Κατηγορίες](#) Έχετε συ

Καταχώρηση νέου προϊόντος

Κωδ. προϊόντος	Κωδ. κατηγορίας	Κατασκευαστής	Τιμή	Εικόνα	Ποσότητα		
1	1	Canon	972.36 €	EOS 50D	13	Αλλαγή	Διαγραφή
2	1	Canon	100 €	A800	8	Αλλαγή	Διαγραφή

Εικόνα 10.2.2.1 Ιδιότητες προϊόντων

Αν ο διαχειριστής πατήσει στο σύνδεσμο «Αλλαγή» του εμφανίζεται μια λίστα με τα πεδία που μπορεί να αλλάξει. Εκείνος μπορεί να διαλέξει το πεδίο που τον ενδιαφέρει και να κάνει την αλλαγή.

• [Πελατολόγιο](#) • [Προϊόντα](#) • [Κατηγορίες](#)

»»Αλλαγή στοιχείων προϊόντος

Παρακαλούμε επιλέξτε το πεδίο που θέλετε να αλλάξετε

Κωδικός Κατηγορίας

Νέα τιμή:

Καταχώρηση

Εικόνα 10.2.2.2 Αλλαγή στοιχείων προϊόντων

Τα αρχεία που χρησιμοποιήθηκαν είναι τα modifyProduct.php και updateProduct.php και παρουσιάζονται παρακάτω.

Κώδικας modifyProduct.php:

```
<?php
include("adminPage.php");
$productId=$_GET["productId"];
echo "<html>";
echo "<head>";
echo "<meta content='text/html; charset=iso-8859-7'>";
echo "</head>";
echo "<body>";
echo "<form action='updateProduct.php' method='get'>";
echo "<table border = '0' align='left' width='450'>";
echo "<caption><b>&raquo;&raquo;Αλλαγή στοιχείων προϊόντος</b></caption>";
echo "<tr>";
echo "<input type='hidden' name='productId' value=$productId>";
echo "<td>Παρακαλούμε επιλέξτε το πεδίο που θέλετε να αλλάξετε</td>";
echo "<td><select name='field'>";
echo "<option value='categoryId'>Κωδικός Κατηγορίας</option>";
echo "<option value='type'>Τύπος</option>";
echo "<option value='info'>Πληροφορίες</option>";
echo "<option value='price'>Αξία</option>";
echo "<option value='image'>Εικόνα</option>";
echo "<option value='stock'>Ποσότητα στοκ</option>";
```

```

echo"</select></td>";
echo"</tr><tr>";
echo"<td>Νέα τιμή:</td>";
echo"<td><input type='text' name='value'></td>";
echo"</tr><tr>";
echo"<td></td>";
echo"<td align='center'><input type='submit' value='Καταχώρηση'></td>";
echo"</tr>";
echo"</table>";
echo"</form>";
?>

```

Κώδικας updateProduct.php:

```

<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query ('set character set utf8');
$pr=mysql_query("SELECT productId FROM product WHERE
productId='".$_GET["productId"]."");
$prId=mysql_fetch_array($pr);
if($prId[0]==$_GET["productId"])
{
    if($_GET["field"]=="categoryId")
    {
        if (empty($_GET["value"])==true or
        ctype_digit($_GET["value"])==false)
        {
            include("modifyProduct.php");
            echo"<br><br>";
            echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
            στο κελί <b>Νέα τιμή</b><br></p>";
        }
        else
        {
            $cat=$_GET['value'];
            $showCategory = mysql_query("SELECT categoryId FROM
            category where categoryId=$cat");
            $categoryRow= mysql_fetch_array($showCategory);
            if($categoryRow[0]==$_GET['value'])
            {
                mysql_query("UPDATE product SET
                categoryId='".$_GET["value"]."
                WHERE productId='".$_GET["productId"]."");
                include("allProducts.php");
                echo "<p>Επιτυχής αλλαγή κωδικού κατηγορίας!</p>";
            }
            else
            {
                include("modifyProduct.php");
                echo"<br><br>";
            }
        }
    }
}

```

```

        echo"<p>Δεν υπάρχει τέτοια κατηγορία.Δοκιμάστε
        ξανά!</p>";
    }
}
elseif($_GET["field"]=="price")
{
    if (empty($_GET["value"])==true or
    is_numeric($_GET["value"])==false)
    {
        include("modifyProduct.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο
        καταχώρησης στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        mysql_query("UPDATE product SET
        price='".$_GET["value"]."
        WHERE productId='".$_GET["productId"]."");
        include("allProducts.php");
        echo "<p>Επιτυχής αλλαγή τιμής προϊόντος!</p>";
    }
}
elseif($_GET["field"]=="type")
{
    if(empty($_GET["value"])==true or
    ctype_alpha($_GET["value"])==false)
    {
        include("modifyProduct.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο
        καταχώρησης στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        mysql_query("UPDATE product SET
        type='".$_GET["value"]."
        WHERE productId='".$_GET["productId"]."");
        include("allProducts.php");
        echo "<p>Επιτυχής αλλαγή τύπου!</p>";
    }
}
elseif($_GET["field"]=="info")
{
    if(empty($_GET["value"])==true)
    {
        include("modifyProduct.php");
        echo"<br><br>";
    }
}

```

```

        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        $str=$_GET["value"];
        $new=iconv("UTF-8","ISO-8859-7",$str);
        mysql_query("UPDATE product SET info='$new' WHERE
        productId='".$_GET["productId"]."");
        include("allProducts.php");
        echo "<p>Επιτυχής αλλαγή πληροφοριών!</p>";
    }
}
elseif($_GET["field"]=="image")
{
    if(empty($_GET["value"])==true)
    {
        include("modifyProduct.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        mysql_query("UPDATE product SET
        image='".$_GET["value"]." WHERE
        productId='".$_GET["productId"]."");
        include("allProducts.php");
        echo "<p>Επιτυχής αλλαγή εικόνας!</p>";
    }
}
elseif($_GET["field"]=="stock")
{
    if (empty($_GET["value"])==true or
    ctype_digit($_GET["value"])==false)
    {
        include("modifyProduct.php");
        echo"<br><br>";
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης
        στο κελί <b>Νέα τιμή</b><br></p>";
    }
    else
    {
        mysql_query("UPDATE stock SET
        quantity='".$_GET["value"]." WHERE
        productId='".$_GET["productId"]."");
        include("allProducts.php");
        echo "<p>Επιτυχής αλλαγή ποσότητας στοκ προϊόντος!</p>";
    }
}
}

```

```

        else
        {
            echo"Error.";
        }
    }
else
{
    echo"<b>Δεν υπάρχει τέτοιο προϊόν. Δοκιμάστε ξανά.</b>";
}
?>

```

10.2.3 Διαγραφή προϊόντος

Όπως αναφέρθηκε ήδη όταν ο διαχειριστής πατήσει στο μενού «Προϊόντα» του εμφανίζονται όλα τα προϊόντα που είναι καταχωρημένα στη βάση δεδομένων μας με τις δυνατότητες αλλαγής και διαγραφής όπως φαίνεται στην εικόνα 10.2.2.1. Έτσι αν απλά πατήσει πάνω στο σύνδεσμο «Διαγραφή» του προϊόντος που θέλει να διαγράψει τότε εκείνο διαγράφεται από τη λίστα, δηλαδή σβήνεται από το πίνακα product (έντονα γράμματα) της βάσης μας και στη συνέχεια εμφανίζεται μήνυμα επιτυχούς διαγραφής.

Το αρχείο που χρησιμοποιήθηκε είναι το deleteFromProduct.php και παρουσιάζεται παρακάτω.

Κώδικας deleteFromProduct.php:

```

<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);

$pr=mysql_query("SELECT productId FROM product WHERE
productId='".$_GET["productId"]."");
$prId=mysql_fetch_array($pr);

if($prId[0]==$_GET["productId"])
{
    mysql_query("DELETE FROM product WHERE
productId='".$_GET["productId"]."");
    include("allProducts.php");
    echo "<p>Επιτυχής διαγραφή προϊόντος.</p>";
}
?>

```

10.3 Κατηγορίες

Οι κατηγορίες αποτελούν ουσιαστικά έναν εύκολο τρόπο ταξινόμησης των προϊόντων και επίσης αποτελούν το κορμό του μενού που παρουσιάζεται στην ιστοσελίδα αφού το μενού προσαρμόζεται δυναμικά με τις αλλαγές που γίνονται στις κατηγορίες. Αυτό είναι εύκολο να το διαπιστώσει κανείς αν δει τον κώδικα του αρχείου menu.php.

Κώδικας menu.php:

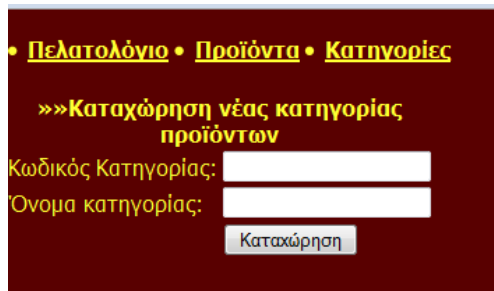
```
<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query('set character set utf8');
$category=mysql_query("SELECT categoryId,categoryName FROM category");
$catResult=mysql_fetch_array($category);
echo"<table cellpadding='6'>";
echo"<tr>";
echo"<td><a href='index.php'><b>Αρχική Σελίδα</b></a></td>";
while($catResult!="")
{
    $catId=$catResult[0];
    $str=$catResult[1];
    echo"<td> <a href='showCategory.php?categoryId=$catId&show=productId'>
    <b>".iconv("UTF-8","ISO-8859-7",$str)."</b></a></td>";
    $catResult=mysql_fetch_array($category);
}
echo"<td><a href='onLinePrint.php'><b>Εκτύπωση φωτογραφιών</b></a></td>";
echo"<td><a href='contact.php'><b>Επικοινωνία</b></a></td>";
echo"</tr>";
echo"</table>";
echo"</br>";
?>
```

Ουσιαστικά αυτό που γίνεται είναι με τη βοήθεια ενός select του πίνακα category να πάρουμε τα ονόματα των κατηγοριών και να τα εισάγουμε στο μενού με τη βοήθεια του αρχείου showCategory. Επειδή όμως δεν γνωρίζουμε πόσες κατηγορίες είναι αποθηκευμένες στη βάση μας χρησιμοποιούμε ένα βρόχο while για να είμαστε σίγουροι ότι θα προβάλλουμε όλα τα αποτελέσματα που βρέθηκαν στο πίνακα των κατηγοριών.

10.3.1 Καταχώρηση νέας κατηγορίας

Για να καταχωρηθεί μια νέα κατηγορία το μόνο που χρειάζεται να εισάγει ο διαχειριστής είναι το όνομα της και τον κωδικό της, οπότε μόλις πατήσει το submit button «Καταχώρηση» που βρίσκεται στο αρχείο newCaterogy.php ενεργοποιείται η φόρμα που εισάγει στη βάση τη νέα κατηγορία.

Τα αρχεία που χρησιμοποιήθηκαν είναι newCaterogy.php και insertIntoCategory.php



Εικόνα 10.3.1.1 Καταχώρηση νέας κατηγορίας

Κώδικας newCaterogy.php:

```
<?php include("adminPage.php"); ?>
<html>
<head>
<meta content="text/html; charset=iso-8859-7">
</head>
<body>
<form action="insertIntoCategory.php" method="get">
<table>
<caption><b>&raquo;&raquo;Καταχώρηση νέας κατηγορίας
προϊόντων</b></caption>
<tr><td>Κωδικός Κατηγορίας:</td>
<td><input type="text" name="categoryId"></td></tr>
<tr><td>Όνομα κατηγορίας:</td>
<td><input type="text" name="categoryName"></td></tr>
<tr><td></td>
<td><input type="submit" value="Καταχώρηση"></td></tr>
</table>
</form>
</body>
</html>
```


Κώδικας insertIntoCategory.php:

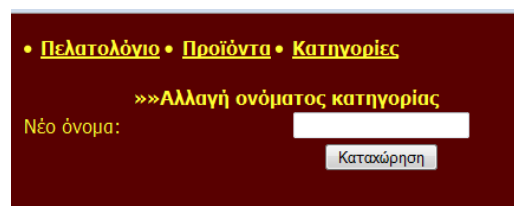
```
<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);
mysql_query ('set character set utf8');

if (ctype_digit($_GET["categoryId"])==false)
{
    include("newCategory.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
<b>Κωδικός Κατηγορίας</b></p>";
}

elseif(empty($_GET["categoryName"])==true or
ctype_alpha($_GET["categoryName"])==false)
{
    include("newCategory.php");
    echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
<b>Όνομα κατηγορίας</b></p>";
}
else
{
    $name=$_GET["categoryName"];
    $str=iconv("ISO-8859-7","UTF-8",$name);
    mysql_query ('set character set utf8');
    $cat="INSERT INTO category(categoryId,categoryName) VALUES
('".$_GET["categoryId"]."', '$str)";
    mysql_query($cat,$con);
    include("allCategories.php");
    echo"<p>Επιτυχής καταχώρηση κατηγορίας.</p>";
}
?>
```

10.3.2 Αλλαγή κατηγορίας

Το στοιχείο που μπορεί να αλλάξει σε μια κατηγορία είναι το όνομα της αφού είναι το μοναδικό πεδίο που επιδέχεται αλλαγή. Έτσι δημιουργήθηκε η αντίστοιχη φόρμα αλλαγής όπου δέχεται το νέο όνομα της κατηγορίας και στη συνέχεια ενημερώνεται ο πίνακας category της βάσης δεδομένων.



Εικόνα 10.3.2.1 Αλλαγή κατηγορίας

Τα αρχεία που χρησιμοποιήθηκαν είναι το modifyCategory.php και updateCategory.php, τα οποία παρουσιάζονται παρακάτω.

Κώδικας modifyCategory.php:

```
<?php include("adminPage.php");
$categoryId=$_GET["categoryId"];
echo "<html>";
echo "<head>";
echo "<meta content='text/html; charset=iso-8859-7'>";
echo "</head>";
echo "<body>";
echo "<form action='updateCategory.php' method='get'>";
echo "<table border = '0' align='left' WIDTH='450'>";
echo "<caption><b>&raquo;&raquo;Αλλαγή ονόματος κατηγορίας</b></caption>";
echo "<tr>";
echo "<input type='hidden' name='categoryId' value=$categoryId>";
echo "<td>Νέο όνομα:</td>";
echo "<td align='center'><input type='text' name='name'></td>";
echo "</tr>";
echo "<tr>";
echo "<td></td>";
echo "<td align='center'><input type='submit' value='Καταχώρηση'></td>";
echo "</tr>";
echo "</table>";
echo "</form>";
echo "</body>";
echo "</html>";
?>
```

Κώδικας updateCategory.php:

```
<?php
include("dbConnection.php");
mysql_select_db("eshop",$con);
$cat=mysql_query("SELECT categoryId FROM category WHERE
categoryId='".$_GET["categoryId"]."");
$catId=mysql_fetch_array($cat);
if($catId[0]==$_GET["categoryId"])
{
    if (ctype_alpha($_GET["name"])==false)
    {
        include("modifyCategory.php");
        echo "<p>Παρακαλούμε εισάγετε σωστό τύπο καταχώρησης στο κελί
        <b>Νέα τιμή</b></p>";
    }
}
```

```

else
{
    mysql_query("UPDATE category SET
categoryName='".$_GET["name"]." WHERE
categoryId='".$_GET["categoryId"]."");
    include("allCategories.php");
    echo "<p>Επιτυχής αλλαγή ονόματος κατηγορίας!</p>";
}
}
else
{
    echo"<b>Δεν υπάρχει τέτοια κατηγορία. Δοκιμάστε ξανά.</b>";
}
?>

```

10.3.3 Διαγραφή κατηγορίας

Η διαγραφή της κατηγορίας χρειάζεται γιατί μπορεί το κατάστημα να πάψει να έχει προϊόντα μιας κατηγορίας που δεν είναι πλέον δημοφιλής και είναι ασύμφορο να υπάρχουν ή απλά για να διορθώσει κάποιο λάθος ο διαχειριστής. Το αρχείο που χρησιμοποιήθηκε είναι το deleteFromCategory.php

Κώδικας deleteFromCategory.php:

```

<?php

$con = mysql_connect("localhost","root","");
mysql_select_db("eshop",$con);

$cat=mysql_query("SELECT categoryId FROM category WHERE
categoryId='".$_GET["categoryId"]."");
$catId=mysql_fetch_array($cat);
if($catId[0]==$_GET["categoryId"])
{
    mysql_query("DELETE FROM category WHERE
categoryId='".$_GET["categoryId"]."");
    include("allCategories.php");
    echo "<p>Επιτυχής διαγραφή κατηγορίας.</p>";
}
else
{
    echo"<p>Δεν υπάρχει τέτοια κατηγορία. Δοκιμάστε ξανά.</p>";
}
?>

```

11. Συμπεράσματα

11.1 Αποτέλεσμα εργασίας

Η ιστοσελίδα που δημιουργήθηκε θα μπορούσε να χρησιμοποιηθεί για οποιοδήποτε κατάσταση φωτογραφικών ειδών αφού παρουσιάζει όλα τα χαρακτηριστικά που έχει ένα ηλεκτρονικό κατάστημα. Επιπλέον η δυνατότητα όχι μόνο αγορών αλλά και εκτύπωσης των φωτογραφιών από το σπίτι είναι κάτι αρκετά χρήσιμο αφού ο πελάτης δεν χρειάζεται να μεταβεί στο κατάστημα για να εκτυπώσει τις φωτογραφίες του αλλά μπορεί πολύ απλά να τις στείλει ηλεκτρονικά και να τις παραλάβει στο χώρο που επιθυμεί. Μάλιστα μια τέτοια ευκολία κάνει το κατάστημα αυτό πιο ανταγωνιστικό στο χώρο της φωτογραφίας.

Επίσης με κάποιες αλλαγές στα δεδομένα της εργασίας δηλαδή στις κατηγορίες και στα προϊόντα θα μπορούσε να γίνει ένα ηλεκτρονικό κατάστημα οποιουδήποτε είδους αφού οι λειτουργίες που εκτελεί όπως είναι η προβολή των προϊόντων, η προθήκη στο καλάθι, η αγορά κτλ δε σχετίζονται με τα φωτογραφικά είδη αλλά με τα δεδομένα που είναι αποθηκευμένα στη βάση μας. Έτσι αν είχαμε ρούχα αντί για φωτογραφικά είδη στο πίνακα product θα μπορούσε με μερικές απλές μετατροπές να γίνει ένα ηλεκτρονικό κατάστημα για το ρουχισμό.

Με όλα αυτά καταλαβαίνουμε τη χρησιμότητα της βάσης δεδομένων μας γιατί πλέον τα στοιχεία-δεδομένα δεν παραμένουν στατικά στη σελίδα μας δυσχεραίνοντας αλλαγές του κώδικα αλλά αποτελούν ξεχωριστό κομμάτι της εργασίας το οποίο είτε μπορεί να χρησιμοποιηθεί είτε να διαγραφεί εντελώς και ένα νέο κομμάτι δεδομένων να αποτελέσει θέμα για την υλοποίηση- προβολή του ηλεκτρονικού καταστήματος.

11.2 Προβλήματα που αντιμετωπίστηκαν

Το μεγαλύτερο πρόβλημα αποτέλεσαν η δημιουργία των session και η εγγραφή ελληνικών στη βάση δεδομένων. Η δημιουργία των session έβγαζε μήνυμα σφάλματος ότι χρησιμοποιείται ήδη πράγμα που δεν ίσχυε αφού δημιουργούνταν εκείνη τη στιγμή. Τελικά η λύση του προβλήματος ήρθε όταν διαπιστώθηκε ότι η εντολή `session_start()`; πρέπει να βρίσκεται πρώτη στο αρχείο. Μάλιστα όχι απλώς πρώτη στο κώδικα `php` αλλά πρώτη γενικά αφού αν για παράδειγμα υπήρχε πάνω από την εντολή αυτή κώδικας σε `html` προέκυπτε και πάλι σφάλμα.

Η εγγραφή ελληνικών ήταν επίσης δύσκολη γιατί τα δεδομένα που εγγράφονταν ήταν από το χρήστη οπότε είχαν αναγκαστικά ελληνική κωδικοποίηση ISO-8859-7 ενώ η βάση μας δεχόταν UTF-8. Η λύση δόθηκε με την εντολή `iconv` η οποία μετέτρεπε τις κωδικοποιήσεις ανάλογα με το αν χρειαζόταν να γίνει εγγραφή στη βάση ή προβολή από τη βάση δεδομένων μας.

Ένα ακόμη θέμα που έπρεπε να αντιμετωπιστεί ήταν η κρυπτογράφηση των κωδικών πρόσβασης των χρηστών διότι δεν είναι σωστό να βρίσκονται εκτεθειμένοι στη βάση δεδομένων μας και οποιοσδήποτε έχει πρόσβαση σε αυτή να μπορεί να τους διαβάσει. Σε επαγγελματικό επίπεδο αυτό θα ήταν ανεπίτρεπτο γιατί οι κωδικοί πρόσβασης αποτελούν ευαίσθητα προσωπικά δεδομένα και σε περίπτωση που διαρρεύσουν μπορεί να δημιουργηθεί πρόβλημα τόσο στην εταιρία όσο και στο πελάτη που διέρρευσε τα στοιχεία. Για τη κρυπτογράφηση λοιπόν χρησιμοποιήθηκε η συνάρτηση `sha1`, η οποία έχει ως πλεονέκτημα, για την εφαρμογή μας, το γεγονός ότι δίνει την ίδια συμβολοσειρά για την ίδια λέξη. Οπότε η συνάρτηση κλήθηκε πριν από την εγγραφή του κωδικού πρόσβασης στο πίνακα μας και η συμβολοσειρά

αποθηκεύθηκε. Έπειτα για να μπορεί να έχει πρόσβαση ο χρήστης στο σύστημα πληκτρολογώντας το κωδικό που έχει επιλέξει και όχι την αποθηκευμένη συμβολοσειρά χρησιμοποιήθηκε και πάλι η sha1 ώστε να μετατρέψει τη λέξη που έγραψε ο χρήστης σε μια συμβολοσειρά. Αν η λέξη ήταν η ίδια με αυτή που έχει αποθηκευθεί κατά τη διάρκεια της εγγραφής τότε παράγονταν η ίδια συμβολοσειρά και γίνονταν η ταυτοποίηση, διαφορετικά εμφανιζόταν μήνυμα σφάλματος.

Τέλος ένα ακόμη πρόβλημα που αντιμετωπίστηκε αφορούσε την αποστολή των δεδομένων στην ίδια καρτέλα. Συγκεκριμένα στην αλλαγή της ποσότητας στο καλάθι αγορών του χρήστη αρχικά υπήρχε ένα input type=text για να εισάγει ο χρήστης τη νέα ποσότητα του προϊόντος. Όμως αν ο χρήστης έκανε μια αλλαγή σε ένα προϊόν του καλαθιού δεν μπορούσε να κάνει αλλαγή σε κανένα άλλο, έτσι ενώ τα δεδομένα αποστέλλονταν κανονικά δεν γίνονταν η εγγραφή στη βάση δεδομένων παρά μόνο στο πρώτο προϊόν και όχι στα υπόλοιπα. Το πρόβλημα αυτό αντιμετωπίστηκε μεταφέροντας το input tag για τη συμπλήρωση της ποσότητας σε άλλη φόρμα. Με τον τρόπο αυτό όσες αλλαγές και αν έκανε ο χρήστης σε όποιο από τα προϊόντα του καλαθιού του ήθελε γίνονταν όλες δεκτές.

11.3 Γνώσεις που αποκτήθηκαν

Η εργασία αυτή προσέφερε μεγάλη εξοικείωση στη δημιουργία php κώδικα αφού για την υλοποίηση της χρησιμοποιούνται πολλές διαφορετικές εντολές με αντίστοιχους τρόπους σύνταξης. Μέσα από την εργασία λοιπόν μελετήθηκε το σύστημα ταυτοποίησης των χρηστών με τη δυνατότητα σύνδεσης και αποσύνδεσης, μελετήθηκε το σύστημα ηλεκτρονικών αγορών καθώς και τα δικαιώματα που δίνονται στους χρήστες για τις αγορές τους. Επιπλέον μελετήθηκε η δυνατότητα ανεβάσματος αρχείων και ο τρόπος υλοποίησης της.

Επίσης επειδή η εργασία στηρίζεται στις εγγραφές που περιέχει η βάση δεδομένων, αποκτήθηκε μεγάλη εξοικείωση με τη δημιουργία πινάκων, με την εισαγωγή και την διαγραφή εγγραφών τόσο άμεσα, με τη βοήθεια του `xampp` όσο και έμμεσα, με τη χρήση εντολών `queries`. Επιπλέον χρειάστηκε να χρησιμοποιηθούν εντολές για τη προβολή, την ανανέωση και τη προσθήκη δεδομένων ώστε οι πληροφορίες της σελίδας να ανανεώνονται όταν αυτό χρειάζεται. Συνολικά θα μπορούσε να πει κανείς ότι μελετήθηκε ένα μεγάλο κομμάτι της διαχείρισης των βάσεων δεδομένων.

Τέλος με τη βοήθεια της εργασίας έγινε μια προσπάθεια υλοποίησης του μοντέλου UP για την ολοκλήρωση της όπου οι επαναλήψεις ορίστηκαν κάθε μήνα περίπου ώστε να συμπίπτουν και με την συνάντηση με τον επιβλέπων καθηγητή. Πριν τη συνάντηση υπήρχε πάντα ένα κομμάτι κώδικα εκτελέσιμο και τεσταρισμένο για παρουσίαση. Μετά από το τέλος κάθε επανάληψης, δηλαδή μετά από κάθε συνάντηση, ορίζονταν οι επιπλέον απαιτήσεις της εργασίας, αναλύονταν, δημιουργούνταν ο κώδικας, περνούσε το `testing` και ετοιμάζονταν η νέα `version` της εργασίας[11].

11.4 Μελλοντικές επεκτάσεις

Μια αρκετά χρήσιμη επέκταση για τις ηλεκτρονικές αγορές είναι η μελέτη του συστήματος πληρωμής μέσω τραπεζών, είτε με πιστωτικές κάρτες είτε με άλλων ειδών λογαριασμούς και η υλοποίηση των συστημάτων αυτών με κώδικα. Έτσι ο σπουδαστής θα μπορεί να εισάγει και ένα κομμάτι του τραπεζικού συστήματος σε ένα ηλεκτρονικό κατάστημα. Δηλαδή θα πρέπει να ενημερωθεί για τον τρόπο που θα μεταφέρονται τα χρήματα από το λογαριασμό που δίνει ο πελάτης στον λογαριασμό του ιδιοκτήτη του καταστήματος. Επιπλέον θα πρέπει να γνωρίζει τις δυνατότητες παρέχουν τα συστήματα αυτά από θέμα ασφάλειας αλλά και πως γίνεται η πρόσβαση στο τραπεζικό λογαριασμό και φυσικά τους όρους για την αφαίρεση του χρηματικού ποσού.

Μια ακόμη ενδιαφέρουσα επέκταση, όσον αφορά το θέμα των φωτογραφιών, θα ήταν να δίνεται η δυνατότητα στο χρήστη να επεξεργάζεται τις φωτογραφίες που ανεβάζει και πιθανόν να δημιουργεί κάποια κολάζ ή φωτογραφίες με διάφορα εφέ που θα υπάρχουν στην εργαλειοθήκη του ανεβάσματος ώστε να μη χρειάζεται να διαθέτει κάποιο προγραμματάκι για επεξεργασία εικόνων. Με τον τρόπο αυτό λοιπόν ο χρήστης θα μπορεί να βλέπει ποιες φωτογραφίες έχει ανεβάσει, πιθανόν και κάποιο άλμπουμ φωτογραφιών, θα μπορεί να εισάγει τα εφέ που δίνονται, είτε κάποια γενικευμένα είτε κάποια πιο ειδικά και έπειτα θα μπορεί να αποθηκεύει είτε στον online φάκελο του είτε στον ηλεκτρονικό υπολογιστή του τις αλλαγές που έχει κάνει. Μετά την αποθήκευση των αλλαγών πιθανόν να εμφανίζεται και πάλι το μενού εκτύπωσης ώστε σε περίπτωση που θέλει να εκτυπώσει και την επεξεργασμένη φωτογραφία να μπορεί να το κάνει χωρίς ιδιαίτερη δυσκολία.

Τέλος μια ακόμη επέκταση που θα μπόρεσε να γίνει είναι να ενσωματωθεί ένα σύστημα που θα φιλτράρει τις προτιμήσεις του χρήστη και θα εμφανίζει προσφορές ή προϊόντα που πιθανόν θα τον ενδιέφεραν να αγοράσει. Επίσης μαζί με αυτό θα μπορούσε να γίνει και μια «βαθμολόγηση» των χρηστών ανάλογα με τις αγορές τους και έτσι θα μπορούσε να δημιουργηθεί ένα σύστημα ανταμοιβής στους καλούς πελάτες.

Βιβλιογραφία

Πηγές:

- [1] Κριμπα Ειρήνη-Βασιλική, Το Πολυαισθητικό Μάρκετινγκ στο Ηλεκτρονικό Εμπόριο, Δεκέμβριος 2009, <http://www.msmfull.aueb.gr/programma/docs/msm6/25.pdf>
- [2] Ηλεκτρονικό κατάστημα φωτογραφικών ειδών. Διαθέσιμο στη διεύθυνση: <http://www.fotodesmos.gr>
- [3] Ηλεκτρονικό κατάστημα εκτύπωσης φωτογραφιών. Διαθέσιμο στη διεύθυνση: <http://www.print-photos-online.com>
- [4] Ηλεκτρονικό κατάστημα. Διαθέσιμο στη διεύθυνση: <http://www.e-shop.gr>
- [5] Γενική ονομασία μεταβλητών. Διαθέσιμο στη διεύθυνση: <http://saites.gr/programming/pos-na-onomazo-metavlites-kai-sinartiseis>
- [6] Σειρές μαθημάτων, Διαθέσιμο στη διεύθυνση: http://www.wlearn.gr/index.php?option=com_content&view=section&layout=blog&id=5&Itemid=118
- [7] Luke Welling, Laura Thomson, Ανάπτυξη web εφαρμογών με php και Mysql, εκδόσεις Μ.Γκιούρδας, Τρίτη έκδοση
- [8] PHP Hypertext Preprocessor. Διαθέσιμο στη διεύθυνση: <http://www.php.net/manual/en/>.
- [9] Ζoom εικόνας. Διαθέσιμο στη διεύθυνση: <http://www.nihilogic.dk/labs/mojozoom/>
- [10] Manual php. Διαθέσιμο στη διεύθυνση: <http://theserverpages.com/php/manual/en/function.mkdir.php>
- [11] Πεταλίδης Νικόλαος, Σημειώσεις θεωρίας Τεχνολογία Λογισμικού II, Software Metodologies