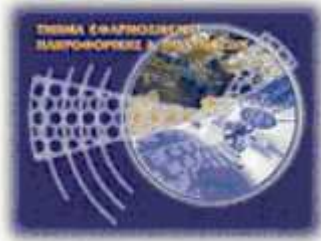




Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

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



Πτυχιακή εργασία

ΔΙΑΔΙΚΤΥΟΚΕΝΤΡΙΚΟ ΣΥΣΤΗΜΑ ΓΙΑ ONLINE ΔΗΜΟΠΡΑΣΙΕΣ ΜΕ ΧΡΗΣΗ XML

ΧΕΙΛΟΥΔΑΚΗ ΓΕΩΡΓΙΑ (ΑΜ: 2177)

ΒΙΔΑΛΑΚΗ ΑΡΤΕΜΙΣ (ΑΜ:2252)

Επιβλέπων Καθηγητής: Δρ. Παπαδάκης Νικόλαος

Επιτροπή αξιολόγησης:

Ημερομηνία Παρουσίασης:

Abstract

The rapid spread of the internet has led to a proportional increase of e-commerce applications, which, beyond the traditional store functions, provide a large number of original internet-only services. This thesis deals with the creation of an electronic auction web application, covering the selection of the appropriate technologies and the writing of code, including a detailed presentation of the development process, and breaking down the decisions that had to be taken at each step.

Σύνοψη

Η ραγδαία εξάπλωση του διαδικτύου έχει οδηγήσει σε μία ανάλογη αύξηση των υπηρεσιών ηλεκτρονικού εμπορίου, όπου πέρα από τα τυπικά ηλεκτρονικά καταστήματα παρέχεται και ένας μεγάλος αριθμός πρωτότυπων υπηρεσιών. Αυτή η εργασία ασχολείται με την δημιουργία μίας διαδικτυακής εφαρμογής ηλεκτρονικών δημοπρασιών, καλύπτοντας όλα τα βήματα από την επιλογή των κατάλληλων τεχνολογιών μέχρι και την συγγραφή του κώδικα, με αναλυτική παρουσίαση της διαδικασίας ανάπτυξης και των αποφάσεων που χρειάστηκε να ληφθούν σε κάθε βήμα.

Περιεχόμενα

Abstract	4
Σύνοψη	5
1 Εισαγωγή.....	9
1.1 Περίληψη.....	9
1.2 Κίνητρο για την διεξαγωγή της εργασίας.....	9
1.3 Σκοπός και στόχοι εργασίας.....	9
1.4 Δομή εργασίας.....	9
2 Μεθοδολογία υλοποίησης	10
2.1 Ηλεκτρονικό εμπόριο στο διαδίκτυο	10
2.2 Δυναμικές και στατικές ιστοσελίδες	10
2.2.1 Στατική ιστοσελίδα.....	10
2.2.2 Δυναμικές ιστοσελίδες	11
2.3 Τεχνολογίες που χρησιμοποιήθηκαν	12
2.3.1 HTML / CSS	12
2.3.2 PHP.....	13
2.3.3 SQL / MySQL	13
3 Ανάλυση απαιτήσεων	14
3.1 Εισαγωγή.....	14
3.2 Σενάρια χρήσης	14
3.3 Ανάλυση σεναρίων χρήσης	15
3.3.1 Εγγραφή χρήστη.....	15
3.3.2 Αλλαγή στοιχείων λογαριασμού	16
3.3.3 Αναζήτηση δημοπρασιών.....	18
3.3.4 Προσφορά σε δημοπρασία	19
3.3.5 Προσθήκη προϊόντος.....	20
3.3.6 Διαγραφή προϊόντος.....	22
3.3.7 Τροποποίηση προϊόντος	23
3.3.8 Προσθήκη δημοπρασίας.....	25
3.3.9 Τροποποίηση δημοπρασίας	26
3.3.10 Προβολή συγκεντρωτικών αναφορών.....	28
3.4 Σχεδιαγράμματα σελίδων	29
3.4.1 Κεντρική σελίδα.....	29
3.4.2 Αρχική σελίδα χρήστη.....	30
3.4.3 Προφίλ χρήστη	30
3.4.4 Τροποποίηση στοιχείων χρήστη.....	31
3.4.5 Αναζήτηση δημοπρασιών.....	31
3.4.6 Αρχική σελίδα διαχειριστή.....	32
3.4.7 Διαχείριση δημοπρασίας	32

3.4.8	Τροποποίηση δημοπρασίας.....	33
3.4.9	Προσθήκη δημοπρασίας.....	33
3.4.10	Διαχείριση προϊόντων.....	34
3.4.11	Τροποποίηση προϊόντων.....	34
3.4.12	Προσθήκη προϊόντων.....	35
3.4.13	Συγκεντρωτικές αναφορές.....	35
3.5	Μοντέλο οντοτήτων - σχέσεων.....	36
4	Ανάπτυξη εφαρμογής.....	38
4.1	Εισαγωγή.....	38
4.2	Περιβάλλον ανάπτυξης.....	38
4.3	Βάση δεδομένων.....	38
4.3.1	Πίνακας δημοπρασιών.....	38
4.3.2	Πίνακας κατηγοριών.....	40
4.3.3	Πίνακας πελατών.....	40
4.3.4	Πίνακας προϊόντων.....	41
4.3.5	Πίνακας προσφορών.....	42
4.4	Δομή εφαρμογής.....	43
4.4.1	Φάκελος js.....	44
4.4.2	Φάκελος css.....	44
4.4.3	Φάκελος images.....	44
4.4.4	Φάκελος mysql.....	44
4.4.5	Φάκελος includes.....	46
4.4.6	Φάκελος classes.....	50
4.4.7	Φάκελος scripts.....	70
5	Εγχειρίδιο χρήσης.....	72
5.1	Εισαγωγή.....	72
5.2	Εγγραφή χρήστη.....	72
5.3	Είσοδος χρήστη.....	72
5.4	Προφίλ χρήστη.....	74
5.5	Επεξεργασία του προφίλ.....	74
5.6	Υποβολή προσφοράς.....	75
5.7	Διαχείριση δημοπρασιών.....	76
5.8	Προσθήκη νέας δημοπρασίας.....	77
5.9	Διαχείριση προϊόντων.....	77
5.10	Τροποποίηση προϊόντος.....	78
5.11	Προσθήκη νέου προϊόντος.....	79
5.12	Συγκεντρωτικές αναφορές.....	80
6	Συμπεράσματα.....	81
7	Βιβλιογραφία.....	82

1 Εισαγωγή

1.1 Περίληψη

Σε αυτή την εργασία θα δείξουμε τα βήματα που ακολουθήσαμε για την διαδικασία ανάπτυξης μίας διαδικτυακής εφαρμογής διαδικτυακών δημοπρασιών. Θα αναλύσουμε τα στάδια της ανάλυσης απαιτήσεων, της δημιουργίας του σχήματος της βάσης και την συγγραφή του κώδικα, παρουσιάζοντας στο τέλος τα συμπεράσματά μας σχετικά με τις δημοφιλέστερες τεχνολογίες που χρησιμοποιούνται σε τέτοιου είδους εφαρμογές.

1.2 Κίνητρο για την διεξαγωγή της εργασίας

Η έκρηξη του ηλεκτρονικού εμπορίου έχει οδηγήσει στην ανάπτυξη ενός μεγάλου αριθμού ηλεκτρονικών καταστημάτων, τα οποία μεταφέρουν την λειτουργικότητα των παραδοσιακών καταστημάτων στο διαδίκτυο. Όμως οι υπάρχουσες τεχνολογίες δίνουν πολύ περισσότερες δυνατότητες σε σχέση με αυτό. Στόχος μας είναι να αξιολογήσουμε τις σύγχρονες τεχνολογίες ανάπτυξης διαδικτυακών εφαρμογών για την δημιουργία μίας πρωτότυπης εφαρμογής ηλεκτρονικού εμπορίου, που θα έχει την μορφή ενός καταστήματος δημοπρασιών.

1.3 Σκοπός και στόχοι εργασίας

Σκοπός της εργασίας είναι η ανάπτυξη μίας ιστοσελίδας ηλεκτρονικών δημοπρασιών με την χρήση σύγχρονων τεχνολογιών που χρησιμοποιούνται στο διαδίκτυο. Μέσω αυτής της διαδικασίας σκοπεύουμε να παρουσιάσουμε τα αντιπροσωπευτικά βήματα της διαδικασίας ανάπτυξης μιας τέτοιας εφαρμογής και να αξιολογήσουμε τις πιο αντιπροσωπευτικές τεχνολογίες και τον τρόπο χρήση τους.

1.4 Δομή εργασίας

Στο δεύτερο κεφάλαιο της εργασίας παρουσιάζουμε τις τεχνολογίες που χρησιμοποιήσαμε με κάποια σύντομα στοιχεία για την ιστορία τους και τις δυνατότητές τους.

Στο τρίτο κεφάλαιο δείχνουμε την διαδικασία ανάλυσης απαιτήσεων και τα σενάρια χρήσης.

Το τέταρτο κεφάλαιο περιγράφει την δομή του κώδικα καθώς και τα πιο ουσιαστικά σημεία του.

Τέλος το πέμπτο κεφάλαιο δίνει τις οδηγίες χρήσης της εφαρμογής και καταλήγουμε με τα συμπεράσματά μας στο έκτο κεφάλαιο.

2 Μεθοδολογία υλοποίησης

2.1 Ηλεκτρονικό εμπόριο στο διαδίκτυο

Η ραγδαία αύξηση του αριθμού των χρηστών του διαδικτύου που ξεκίνησε από το τέλος του προηγούμενου αιώνα έκανε πολύ γρήγορα το internet ένα από τα βασικά μέσα επικοινωνίας. Αυτό συνοδεύτηκε από μία ανάλογη έκρηξη στις εφαρμογές ηλεκτρονικού εμπορίου, οι οποίες προσπάθησαν να εκμεταλλευτούν το μεγάλο κοινό του νέου μέσου και την πρόσβαση που επέτρεπε σε έναν μεγάλο αριθμό πιθανών αγοραστών.

Αρχικά οι περιορισμοί που επιβάλλονταν από την πρόιμη κατάσταση των τεχνολογιών αποτελούσαν ένα εμπόδιο στην δημιουργία των αντίστοιχων εφαρμογών, με αποτέλεσμα η δημιουργία ενός ηλεκτρονικού καταστήματος να είναι μία διαδικασία που είχε μεγάλο κόστος, απαιτούσε ιδιαίτερα εξειδικευμένο προσωπικό και πολύ συχνά παρήγαγε αποτελέσματα χαμηλής ποιότητας. Όμως με το πέρασμα του χρόνου οι διαθέσιμες τεχνολογίες εξελίχθηκαν.

Οι σημερινές ιστοσελίδες δεν μοιάζουν σε τίποτα με τις σελίδες του παρελθόντος, με πολλές από αυτές να ανταγωνίζονται τόσο στην εμφάνιση όσο και στις δυνατότητες τις αντίστοιχες παραδοσιακές εφαρμογές. Στόχος μας είναι να μελετήσουμε πως οι τεχνολογίες που χρησιμοποιούνται σε τόσο μεγάλο βαθμό στο διαδίκτυο μπορούν να χρησιμοποιηθούν για την ανάπτυξη μίας εφαρμογής ηλεκτρονικών δημοπρασιών, δηλαδή έξω από τα πλαίσια του παραδοσιακού ηλεκτρονικού καταστήματος.

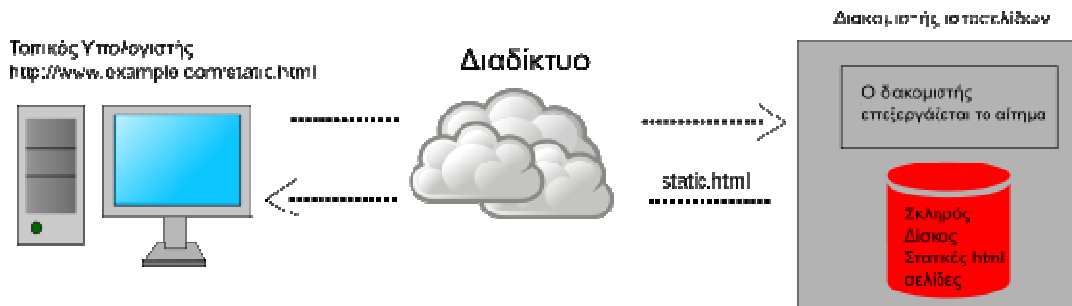
2.2 Δυναμικές και στατικές ιστοσελίδες

Οι δύο μεγάλες κατηγορίες ιστοσελίδων είναι οι λεγόμενες στατικές (static) και οι δυναμικές (dynamic). Όσον αφορά την όψη/εμφάνισή τους, δεν έχουν μεγάλες διαφορές μεταξύ τους, αλλά όσον αφορά την λειτουργία και τις δυνατότητές τους οι διαφορές είναι πολλές και ουσιαστικές.

Αν και η εφαρμογή μας είναι μία δυναμική ιστοσελίδα θα παρουσιάσουμε επιγραμματικά τα χαρακτηριστικά των δύο ειδών ώστε να γίνει ξεκάθαρος ο μεταξύ τους διαχωρισμός.

2.2.1 Στατική ιστοσελίδα

Στατική ιστοσελίδα (static web-page ή flat web-page) ονομάζεται μια ιστοσελίδα της οποίας το περιεχόμενο μεταφέρεται στον χρήστη ακριβώς στην μορφή που είναι αποθηκευμένο στον εξυπηρετητή ιστοσελίδων (web server), σε αντίθεση με τις δυναμική ιστοσελίδα όπου το περιεχόμενο δημιουργείται από μια εφαρμογή η οποία εκτελείται στον εξυπηρετητή ιστοσελίδων. Τα περιεχόμενα μιας στατικής ιστοσελίδας εμφανίζονται με την ίδια μορφή σε όλους του χρήστες με την μορφή που είναι αποθηκευμένα στο σύστημα αρχείων του εξυπηρετητή ιστοσελίδων. Είναι ουσιαστικά απλά ηλεκτρονικά "έγγραφα". Μπορούν να περιέχουν κείμενα, φωτογραφίες, συνδέσμους (links), αρχεία για download, κινούμενα σχέδια (animated graphics).



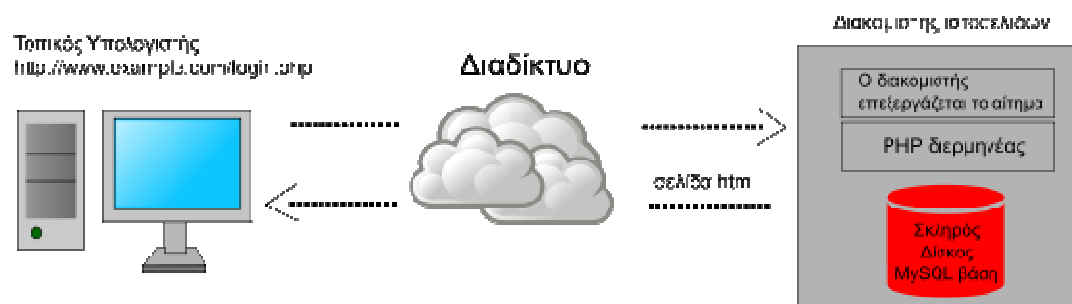
Εικόνα 2.1 Η ροή των πληροφοριών σε μία στατική ιστοσελίδα

Είναι κατάλληλες κυρίως για την δημιουργία "μόνιμων/στατικών παρουσιάσεων", όπου δεν υπάρχει συχνά η ανάγκη να τροποποιείται το περιεχόμενό τους. Δηλαδή, θα μπορούσε να χρησιμοποιηθεί μια στατική ιστοσελίδα για την γενική παρουσίαση μιας επιχείρησης και των προϊόντων ή υπηρεσιών της. Οι στατικές ιστοσελίδες είναι αποθηκευμένες συνήθως σε μορφή HTML και μεταφέρονται χρησιμοποιώντας το πρωτόκολλο HTTP.

2.2.2 Δυναμικές ιστοσελίδες

Δυναμική ιστοσελίδα (dynamic web page) είναι μια ιστοσελίδα η οποία δημιουργείται δυναμικά την στιγμή της πρόσβασης σε αυτή ή την στιγμή που ο χρήστης αλληλοεπιδρά με τον εξυπηρετητή ιστοσελίδων. Οι δυναμικές ιστοσελίδες θεωρούνται δομικό στοιχείο της νέας γενιάς του παγκόσμιου ιστού (Web 2.0) όπου η πληροφορία διαμοιράζεται σε πολλαπλές ιστοσελίδες. Η δυναμική ιστοσελίδα μπορεί να δημιουργείται δυναμικά από ένα σενάριο εντολών, το οποίο εκτελείται τοπικά στο πελάτη ή στον εξυπηρετητή ή και στον πελάτη και στον εξυπηρετητή. Οι δυναμικές ιστοσελίδες, μπορεί στην εμφάνιση, σε πολλές περιπτώσεις, να μην έχουν μεγάλη διαφορά με τις στατικές, όμως οι δυνατότητές είναι πολύ περισσότερες, από πολλές πλευρές, καθώς στην περίπτωση αυτή ουσιαστικά πρόκειται για μία εφαρμογή (πρόγραμμα), και όχι ένα απλό ηλεκτρονικό έγγραφο.

Συνήθως, οι δυναμικές ιστοσελίδες, χρησιμοποιούν κάποια βάση δεδομένων (database), όπου αποθηκεύουν πληροφορίες και απ' όπου αντλούν το περιεχόμενό τους, ανάλογα με το τι ζητάει ο χρήστης/επισκέπτης σε κάθε του "κλικ". Η χρήση των βάσεων δεδομένων, είναι αυτή που επιτρέπει την εύκολη προσθαφαίρεση περιεχομένου στις δυναμικές ιστοσελίδες, καθώς δεν απαιτείται να επεξεργάζεται κανείς κάθε φορά την ίδια την ιστοσελίδα, αλλά απλά να διαχειρίζεται έμμεσα το περιεχόμενο στην βάση δεδομένων και οι υπόλοιπες διαδικασίες γίνονται αυτοματοποιημένα από τον "μηχανισμό" της ιστοσελίδας.



Εικόνα 2.2 Η ροή εργασιών σε μία δυναμική ιστοσελίδα

Για την εύκολη **διαχείριση του περιεχομένου** σε μία δυναμική ιστοσελίδα, υπάρχει πρόσθετα στην ιστοσελίδα ένας εύχρηστος μηχανισμός "**CMS**" (Content Management System), μέσω του οποίου η προσθαφαίρεση περιεχομένου μπορεί να γίνει ακόμη και από κάποιον αρχάριο. Φυσικά, η "περιοχή διαχείρισης" της ιστοσελίδας, προστατεύεται με κωδικό πρόσβασης (password), και δεν μπορούν να εισέλθουν σε αυτή οι επισκέπτες της ιστοσελίδας.

2.3 Τεχνολογίες που χρησιμοποιήθηκαν

2.3.1 HTML / CSS

Η συντριπτική πλειοψηφία των σελίδων στο διαδίκτυο είναι γραμμένες στη γλώσσα HTML. Ο βασικότερος λόγος ήταν αρχικά η συμβατότητα με όλους τους browsers. Το γεγονός πως όλοι οι χρήστες μπορούν να δουν μία σελίδα HTML ήταν από την αρχή στοιχειώδες στην επιτυχία της, καθώς υποσκέλιζε τις αρχικά περιορισμένες δυνατότητες της γλώσσας.

Μέρος της HTML είναι και η γλώσσα CSS (Cascading Style Sheets). Οι δύο γλώσσες αλληλοσυμπληρώνονται, με την HTML να περιγράφει την δομή και το περιεχόμενο μίας σελίδας ενώ το CSS δίνει οδηγίες για την αισθητική της παρουσίαση.

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

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

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

Η HTML5 είναι μια υπό ανάπτυξη γλώσσα σήμανσης για τον Παγκόσμιο Ιστό που όταν ετοιμαστεί θα είναι η επόμενη μεγάλη έκδοση της HTML (Γλώσσα Υπερκειμένου, HyperText Markup Language). Η ομάδα Web Hypertext Application Technology Working Group (WHATWG) άρχισε δουλειά σε αυτή την έκδοση τον Ιούνιο του 2004 με το όνομα Web Applications 1.0. Το Φεβρουάριο του 2010 το πρότυπο ήταν ακόμη σε κατάσταση "Last Call" στο WHATWG.

Η HTML5 προορίζεται για αντικατάσταση της HTML 4.01, της XHTML 1.0, και της DOM Level 2 HTML. Ο σκοπός είναι η μείωση της ανάγκης για ιδιόκτητα plug-in και πλούσιες διαδικτυακές εφαρμογές (RIA) όπως το Adobe Flash, το Microsoft Silverlight, το Apache Pivot, και η Sun JavaFX. Οι ιδέες πίσω από την HTML5 εμφανίστηκαν αρχικά το

2004 από την ομάδα WHATWG. Η HTML5 εμπεριέχει το πρότυπο Web Forms 2.0 που είναι επίσης της WHATWG.

Το πρότυπο HTML5 υιοθετήθηκε ως αρχικό βήμα για τις εργασίες της νέας ομάδας εργασίας HTML του W3C το 2007. Αυτή η ομάδα εργασίας δημοσίευσε το Πρώτο Δημόσιο Working Draft του προτύπου στις 22 Ιανουαρίου 2008. Το πρότυπο είναι ακόμη υπό ανάπτυξη, και αναμένεται να παραμείνει έτσι για πολλά χρόνια, παρόλο που μέρη της HTML5 θα τελειώσουν και θα υποστηριχτούν από περιηγητές πριν το όλο πρότυπο φτάσει στη τελική κατάσταση Recommendation.

2.3.2 PHP

Η PHP είναι μια scripting γλώσσα υψηλού επιπέδου, που σχεδιάστηκε για τη δημιουργία δυναμικών ιστοσελίδων. Η php στηρίζεται στο μεγαλύτερο μέρος της σύνταξης στην C, στην Java και στην Perl, αλλά έχει και μερικά δικά της μοναδικά χαρακτηριστικά. Σκοπός της γλώσσας είναι να δώσει τη δυνατότητα στους web developers να δημιουργούν δυναμικά παραγόμενες ιστοσελίδες. Αν και έχει εξελιχθεί σε μια γλώσσα γενικής χρήσης, η κύρια δύναμη της παραμένει στο χώρο του web development.

Πολλά από τα γνωστότερα προγράμματα online forum ή wiki είναι γραμμένα σε PHP, και ο συνδυασμός της με τον Apache web server και το σύστημα διαχείρισης βάσης δεδομένων MySQL, είναι τόσο συχνός που αναφέρεται συχνά ως LAMP. Η σύνταξη της θυμίζει αρκετά αυτή της C, και είναι αρκετά απλή στην εκμάθηση, ακόμα και από αρχάριους προγραμματιστές. Αυτό, σε συνδυασμό με μερικές κακές επιλογές από τους σχεδιαστές της γλώσσας στο παρελθόν, έχει συντελέσει στο ομολογουμένως κακό ιστορικό ασφαλείας πολλών εφαρμογών γραμμένων σε PHP.

Παρόμοιες γλώσσες προγραμματισμού της τεχνολογίας php είναι η asp (Active Server Pages) από την Microsoft, η cfml (ColdFusion Markup Language) της εταιρείας Allaire και jsp (JavaServer Pages) της εταιρείας Sun. Παρόλα αυτά είναι μακράν η δημοφιλέστερη του είδους, σε βαθμό που πολλές φορές της αποδίδεται ένα μέρος από την ραγδαία ανάπτυξη του διαδικτύου.

2.3.3 SQL / MySQL

Η SQL (Structured Query Language) είναι μία γλώσσα υπολογιστών στις βάσεις δεδομένων, που σχεδιάστηκε για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Database Management System, RDBMS) και η οποία, αρχικά, βασίστηκε στη σχεσιακή άλγεβρα. Η γλώσσα περιλαμβάνει δυνατότητες ανάκτησης και ενημέρωσης δεδομένων, δημιουργίας και τροποποίησης σχημάτων και σχεσιακών πινάκων, αλλά και ελέγχου πρόσβασης στα δεδομένα. Η SQL ήταν μία από τις πρώτες γλώσσες για το σχεσιακό μοντέλο του Edgar F. Codd, στο σημαντικό άρθρο του το 1970, και έγινε η πιο ευρέως χρησιμοποιούμενη γλώσσα για τις σχεσιακές βάσεις δεδομένων.

Η βάση δεδομένων MySQL είναι η πιο διαδεδομένη υλοποίηση της γλώσσας SQL. Πρόκειται για ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων ανοικτού κώδικα. Το πρόγραμμα τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων.

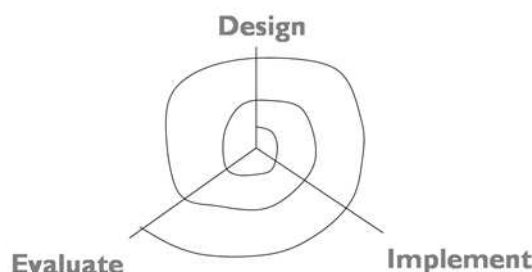
3 Ανάλυση απαιτήσεων

3.1 Εισαγωγή

Σε αυτή την ενότητα θα δούμε τα βήματα που ακολουθήσαμε για την ανάλυση των απαιτήσεων της εφαρμογής.

Για τη δημιουργία της εφαρμογής ακολουθήσαμε το σπειροειδές μοντέλο. Ξεκινώντας από έναν αρχικό σχεδιασμό της διεπαφής της εφαρμογής σε χαρτί προχωρήσαμε στην υλοποίησή του στον υπολογιστή παρέχοντας τη βασική λειτουργικότητα. Χρησιμοποιώντας τη διεπαφή στα αρχικά στάδια υλοποίησης μπορούσαμε να βρούμε τυχόν προβλήματα με τη διάταξη των μενού. Αυτό το στάδιο ονομάζεται στάδιο αξιολόγησης. Κάθε πρόβλημα που εμφανιζόταν στο στάδιο αξιολόγησης μας οδηγούσε πάλι στο στάδιο του σχεδιασμού, ώστε να διορθωθούν οι ελλείψεις και τα προβλήματα που παρατηρήθηκαν.

Spiral Model



Εικόνα 3.1 Το σπειροειδές μοντέλο ανάπτυξης.

Η παραπάνω διαδικασία μοιάζει με έναν κύκλο από το στάδιο της σχεδίασης στην υλοποίηση και την αξιολόγηση. Όμως καθώς σε κάθε επανάληψη τα αποτελέσματα είναι περισσότερο πλήρη, αφού περιέχουν τις διορθώσεις που έγιναν στα προηγούμενα βήματα, μπορεί να αναπαρασταθεί καλύτερα με μία σπείρα.

3.2 Σενάρια χρήσης

Κατά την διαδικασία σχεδίασης της εφαρμογής διακρίναμε κάποια βασικά σενάρια χρήσης. Αυτά είναι τα παρακάτω:

1. Εγγραφή χρήστη
2. Αλλαγή στοιχείων λογαριασμού
3. Αναζήτηση δημοπρασιών
4. Προσφορά σε δημοπρασία
5. Προσθήκη προϊόντος
6. Διαγραφή προϊόντος
7. Τροποποίηση προϊόντος
8. Προσθήκη δημοπρασίας
9. Τροποποίηση δημοπρασίας
10. Προβολή συγκεντρωτικών αναφορών

Βάση αυτών προχωρήσαμε στην ανάλυση του κάθε σεναρίου και στην δημιουργία των κατάλληλων διαγραμμάτων ροής.

3.3 Ανάλυση σεναρίων χρήσης

3.3.1 Εγγραφή χρήστη

Σκοπός

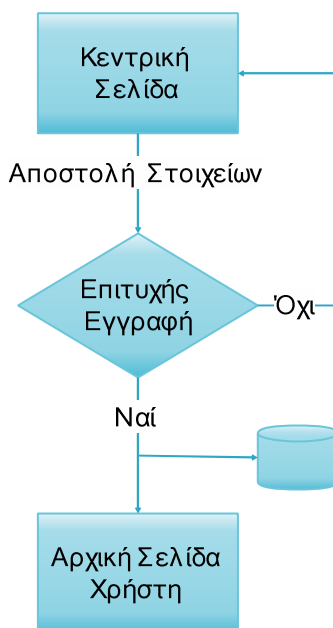
Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι η απόκτηση ενός λογαριασμού στο σύστημα, ώστε να έχει την δυνατότητα να δίνει προσφορές και να αγοράζει διάφορα προϊόντα που βρίσκονται σε δημοπρασία.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τη φόρμα «Εγγραφή χρήστη», εισάγει τα στοιχεία του και πατάει το κουμπί «Αποστολή Στοιχείων».
- 4) Το σύστημα αποθηκεύει τα στοιχεία του χρήστη στην βάση, εμφανίζει μήνυμα ότι η εγγραφή ολοκληρώθηκε επιτυχώς, συνδέει το χρήστη αυτόματα και εμφανίζει την «Αρχική Σελίδα Χρήστη».

Εναλλακτική Ροή

- 4.α.1. Ο χρήστης είτε δεν έχει συμπληρώσει όλα τα απαραίτητα στοιχεία που απαιτούνται για την εγγραφή είτε έχει δώσει κάποια μη έγκυρα στοιχεία.
- 4.α.2 Το σύστημα επιστρέφει το αντίστοιχο μήνυμα λάθους στην κεντρική σελίδα.
- 4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής.



3.3.2 Αλλαγή στοιχείων λογαριασμού

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι να κάνει διάφορες αλλαγές στα στοιχεία χρήσης του λογαριασμού του. (π.χ. κωδικού πρόσβασης, email, ονόματος κλπ.)

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία στη φόρμα «Login» και πατάει σύνδεση.
- 4) Το σύστημα συνδέει το χρήστη και τον μεταφέρει στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Προφίλ Χρήστη» από το μενού της σελίδας.
- 6) Το σύστημα εμφανίζει την σελίδα «Προφίλ Χρήστη».
- 7) Ο χρήστης επιλέγει το κουμπί «Αλλαγή Στοιχείων» από το μενού της σελίδας.
- 8) Το σύστημα εμφανίζει τη φόρμα «Αλλαγή στοιχείων».

- 9) Ο χρήστης συμπληρώνει στη φόρμα τις αλλαγές που επιθυμεί και πατάει το κουμπί «Υποβολή Στοιχείων».
- 10) Το σύστημα αποθηκεύει τις αλλαγές στη βάση και επιστρέφει το χρήστη στη σελίδα «Προφίλ Χρήστη» όπου εμφανίζονται τα ανανεωμένα στοιχεία του χρήστη.

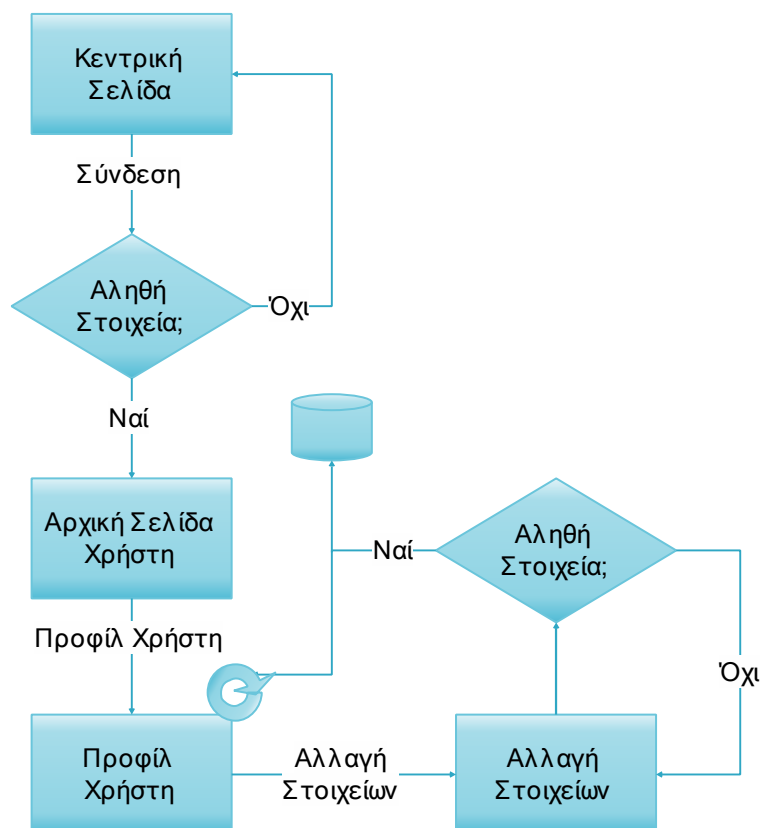
Εναλλακτική Ροή 1

- 4.α.1. Τα στοιχεία που έχει συμπληρώσει ο χρήστης δεν είναι έγκυρα.
- 4.α.2. Το σύστημα εμφανίζει μήνυμα λάθους ότι τα στοιχεία δεν είναι έγκυρα.
- 4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της εναλλακτικής ροής.

Εναλλακτική Ροή 2

10.α.1. Ο Χρήστης έχει δώσει κάποια μη έγκυρα στοιχεία (π.χ. ένα email που δεν περιέχει "@" , είτε έναν μη έγκυρο κωδικό πρόσβασης) 10.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους στη φόρμα, και ζητάει από το χρήστη να βάλει έγκυρα στοιχεία.

10.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 9 της βασικής ροής.



3.3.3 Αναζήτηση δημοπρασιών

Σκοπός

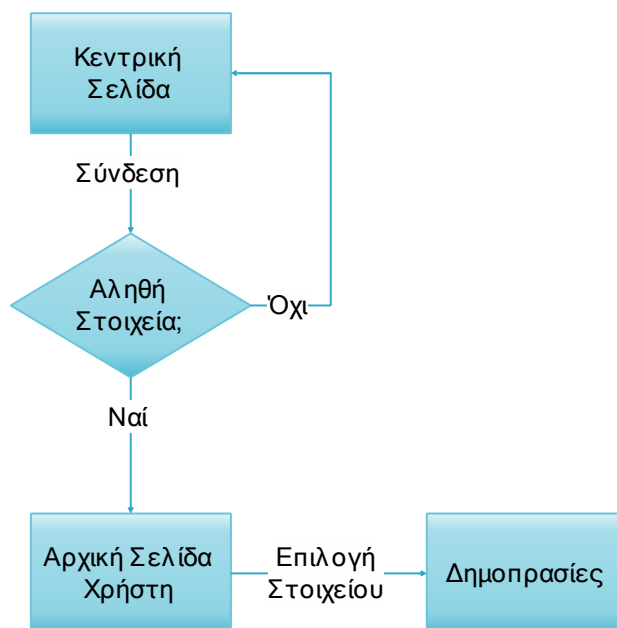
Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι η είσοδος στο σύστημα, ώστε να έχει την δυνατότητα να ενημερωθεί για τα προϊόντα που βρίσκονται σε δημοπρασία καθώς και να κάνει αναζήτηση από το μενού ανά κατηγορία εταιρία ή προϊόν.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης επιλέγει να συμπληρώσει τη φόρμα «Login» είσοδο στην σελίδα και πατάει σύνδεση.
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη» στην οποία ο χρήστης έχει τη δυνατότητα να δει την τρέχουσα και τις μελλοντικές δημοπρασίες και να επιλέξει τον από το μενού τον τρόπο αναζήτησης που επιθυμεί.
- 5) Ο χρήστης επιλέγει κάποια κατηγορία εταιρία ή προϊόν από το μενού.
- 6) Το σύστημα τον μεταφέρει στην σελίδα όπου εμφανίζονται οι δημοπρασίες που πληρούν τα κριτήρια αναζήτησης.

Εναλλακτική Ροή 1

- 4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.
- 4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.
- 4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής.



3.3.4 Προσφορά σε δημοπρασία

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι η είσοδος στο σύστημα, ώστε να κάνει προσφορά σε κάποιο προϊόν δημοπρασίας προκειμένου να το αποκτήσει.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία στη φόρμα «Login» και πατάει σύνδεση.
- 4) Το σύστημα εισάγει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης συμπληρώνει το επιθυμητό ποσό στο πεδίο προσφοράς της δημοπρασίας που τον ενδιαφέρει και πατάει το κουμπί προσφορά.
- 6) Το σύστημα εμφανίζει μήνυμα ότι η προσφορά στάλθηκε και αποθηκεύει την προσφορά στη βάση.

Εναλλακτική Ροή 1

3.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.

3.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.

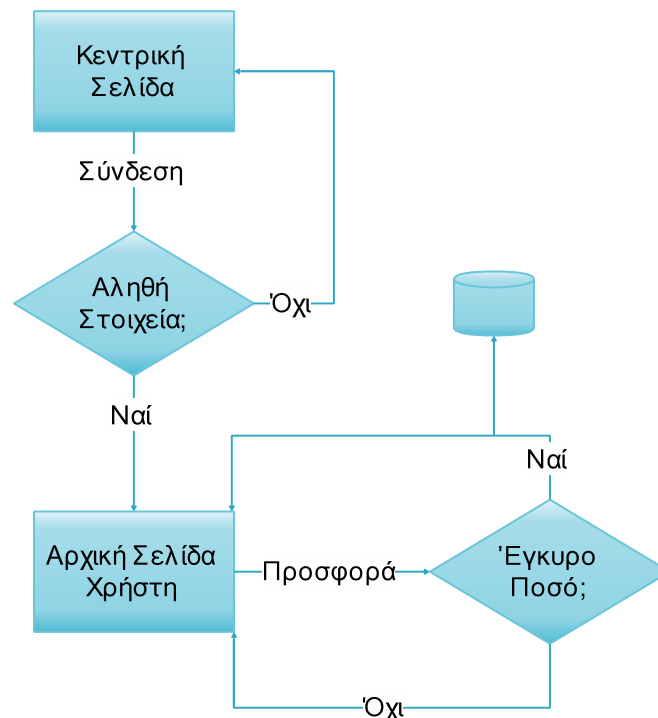
3.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής.

Εναλλακτική Ροή 2

5.α.1. Ο Χρήστης έχει δώσει μη έγκυρο ποσό (π.χ. αρνητικό αριθμό, αλφαριθμητικό κλπ.)

5.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους στη σελίδα, και ζητάει από το χρήστη να βάλει έγκυρο ποσό.

5.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 5 της βασικής ροής.



3.3.5 Προσθήκη προϊόντος

Σκοπός

Ο σκοπός του χρήστη (διαχειριστή) σε αυτή την περίπτωση είναι να προσθέσει ένα νέο προϊόν στην σελίδα με σκοπό τη μελλοντική δημοπράτηση του.

Βασική ροή

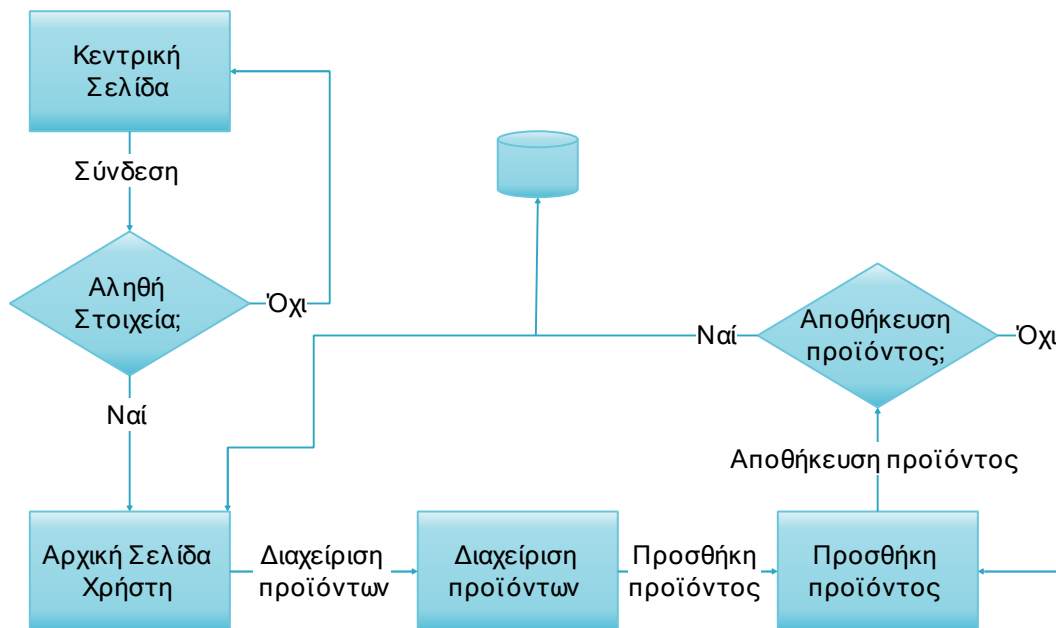
- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία στη φόρμα του διαχειριστή «Login» και πατάει σύνδεση.
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Διαχείριση προϊόντων» από το μενού της σελίδας.
- 6) Το σύστημα εμφανίζει τη σελίδα «Διαχείριση προϊόντων».
- 7) Ο χρήστης επιλέγει το σύνδεσμο «Προσθήκη προϊόντος» από το μενού της σελίδας.
- 8) Το σύστημα εμφανίζει την φόρμα «Προσθήκη Προϊόντος».
- 9) Ο χρήστης εισάγει τα στοιχεία του προϊόντος στη φόρμα και πατάει το κουμπί «Αποθήκευση Προϊόντος».
- 10) Το σύστημα αποθηκεύει το νέο προϊόν στη βάση, εμφανίζει μήνυμα επιτυχούς καταχώρησης προϊόντος και επιστρέφει το χρήστη στη σελίδα «Αρχική Σελίδα Χρήστη».

Εναλλακτική Ροή 1

- 4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.
- 4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.
- 4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής.

Εναλλακτική Ροή 2

- 8.α.1. Ο Χρήστης ή δεν έχει συμπληρώσει όλα τα απαραίτητα πεδία της φόρμας ή τα στοιχεία που έχει συμπληρώσει δεν είναι έγκυρα ή το προϊόν που εισήγαγε είναι ήδη καταχωρημένο στη βάση.
- 8.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους στη φόρμα, ότι τα στοιχεία δεν είναι αποδεκτά και ότι το προϊόν δεν μπορεί να αποθηκευτεί στη βάση και ζητάει από το χρήστη να βάλει έγκυρα στοιχεία.
- 8.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 9 της βασικής ροής.



3.3.6 Διαγραφή προϊόντος

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι να διαγράψει κάποιο προϊόν από την σελίδα μας.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία του διαχειριστή στη φόρμα «Login» και πατάει σύνδεση.
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Διαχείριση προϊόντων» από το μενού της σελίδας.
- 6) Το σύστημα εμφανίζει την σελίδα διαχείριση προϊόντων.
- 7) Ο χρήστης πατάει το κουμπί «Διαγραφή» δίπλα από το προϊόν που επιθυμεί να διαγράψει.

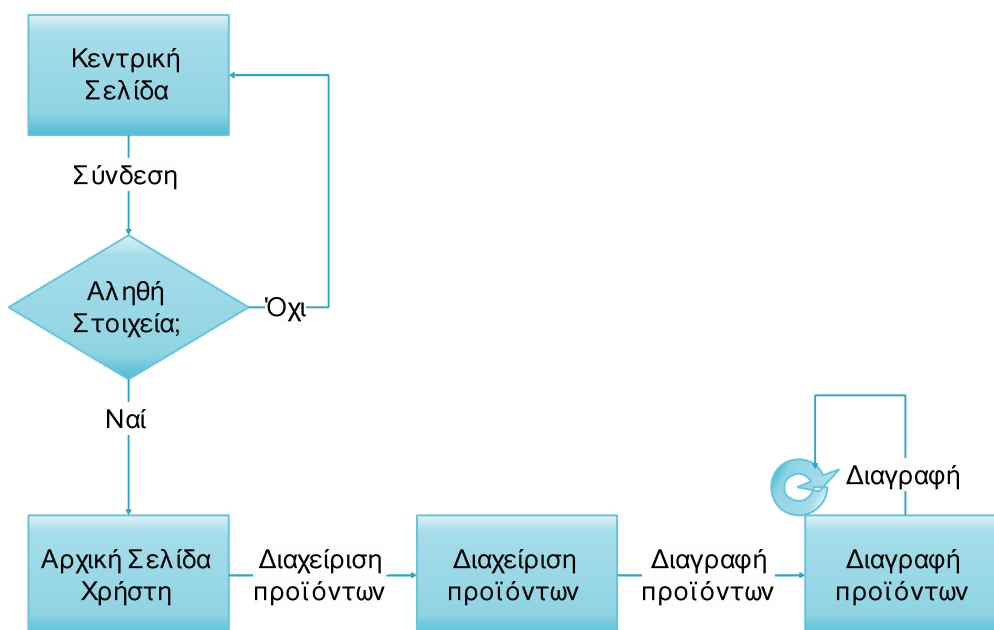
- 8) Το σύστημα διαγράφει το προϊόν, εμφανίζει μήνυμα επιτυχούς διαγραφής και επιστρέφει το χρήστη στη σελίδα «Αρχική Σελίδα Χρήστη».

Εναλλακτική Ροή 1

4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.

4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.

4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής



3.3.7 Τροποποίηση προϊόντος

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι να διορθώσει ή να συμπληρώσει στοιχεία σε κάποιο προϊόν.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.

- 3) Ο χρήστης συμπληρώνει τα στοιχεία στη φόρμα «Login» και πατάει σύνδεση.
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Διαχείριση προϊόντων» από το μενού της σελίδας.
- 6) Το σύστημα εμφανίζει την σελίδα διαχείρισης προϊόντων.
- 7) Ο χρήστης πατάει το κουμπί «Τροποποίηση» δίπλα από το προϊόν που επιθυμεί να τροποποιήσει.
- 8) Το σύστημα του ανοίγει την φόρμα του προϊόντος που θέλει να τροποποιήσει δίνοντας του την δυνατότητα να αλλάξει όποιο στοιχείο θέλει.
- 9) Ο χρήστης κάνει την τροποποίηση και πατάει αποθήκευση.
- 10) Το σύστημα αποθηκεύει τις αλλαγές στη βάση και επιστρέφει το χρήστη στη σελίδα «Αρχική Σελίδα Χρήστη».

Εναλλακτική Ροή 1

4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.

4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.

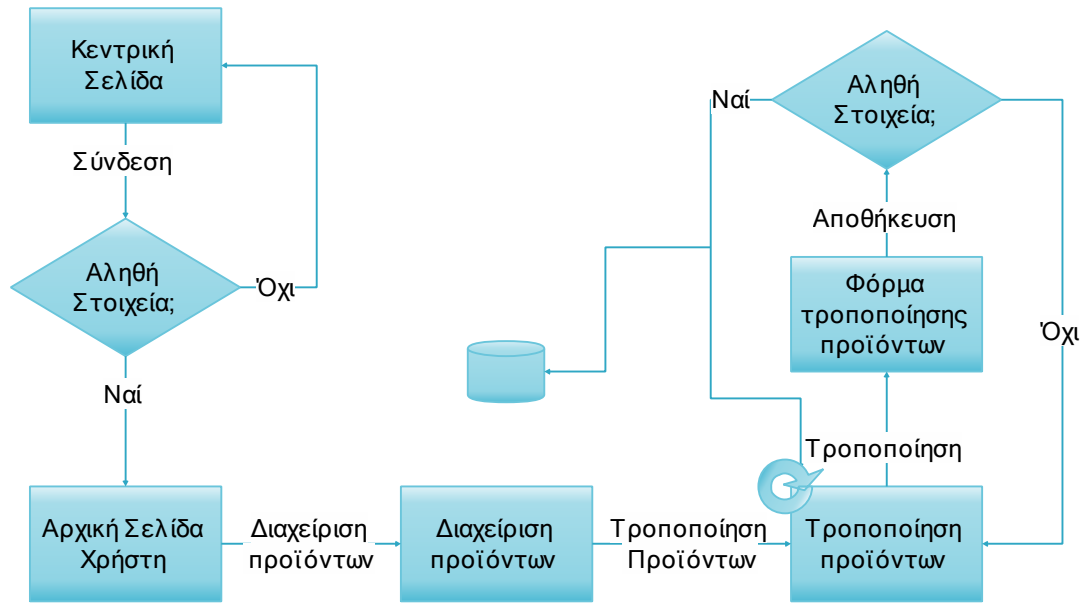
4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής.

Εναλλακτική Ροή 2

10.α.1. Τα στοιχεία που έχει συμπληρώσει δεν είναι έγκυρα.

10.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους στη φόρμα, ότι τα στοιχεία δεν είναι αποδεκτά και ότι το προϊόν δεν μπορεί να τροποποιηθεί και ζητάει από το χρήστη να βάλει έγκυρα στοιχεία.

10.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 9 της βασικής ροής.



3.3.8 Προσθήκη δημοπρασίας

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι να προσθέσει μια δημοπρασία ώστε να πουλήσει κάποιο προϊόν .

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία του διαχειριστή στη φόρμα «Login» και πατάει σύνδεση.
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Διαχείριση δημοπρασιών»
- 6) Το σύστημα εμφανίζει τη σελίδα διαχείριση δημοπρασιών.
- 7) Ο χρήστης επιλέγει το σύνδεσμο «Προσθήκη δημοπρασίας» από το μενού της σελίδας.
- 8) Το σύστημα του εμφανίζει την σελίδα με την φόρμα για την προσθήκη της δημοπρασίας .
- 9) Ο χρήστης συμπληρώνει την φόρμα με τα στοιχεία για την δημοπρασία και πατάει το κουμπί « Αποθήκευση ».

- 10) Το σύστημα αποθηκεύει τις αλλαγές στη βάση και επιστρέφει το χρήστη στη σελίδα «Αρχική Σελίδα Χρήστη» με τη νέα δημοπρασία να εμφανίζεται στην «Αρχική Σελίδα Χρήστη».

Εναλλακτική Ροή 1

4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.

4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.

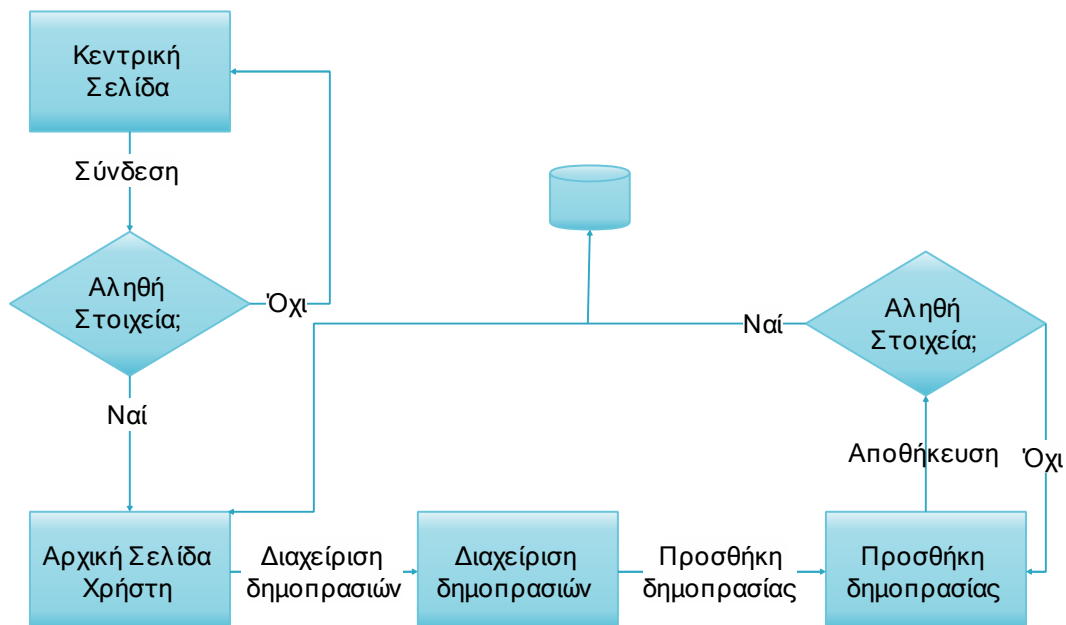
4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής

Εναλλακτική Ροή 2

8.α.1. Ο Χρήστης είτε δεν έχει συμπληρώσει όλα τα απαραίτητα πεδία της φόρμας είτε τα στοιχεία που έχει συμπληρώσει δεν είναι έγκυρα.

8.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους στη φόρμα, ότι τα στοιχεία δεν είναι αποδεκτά και ότι η δημοπρασία δεν μπορεί να αποθηκευτεί στη βάση και ζητάει από το χρήστη να βάλει έγκυρα στοιχεία.

8.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 7 της βασικής ροής.



3.3.9 Τροποποίηση δημοπρασίας

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι να τροποποιήσει μια δημοπρασία.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία του διαχειριστή στη φόρμα «Login» και πατάει σύνδεση
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Διαχείριση δημοπρασιών»
- 6) Το σύστημα εμφανίζει τη σελίδα διαχείριση δημοπρασιών.
- 7) Ο χρήστης πατάει το κουμπί «Τροποποίηση» δίπλα από τη δημοπρασία που επιθυμεί να τροποποιήσει.(τρέχουσα ή μελλοντική)
- 8) Το σύστημα του εμφανίζει την φόρμα της δημοπρασίας που επέλεξε.
- 9) Ο χρήστης τροποποιεί τα δεδομένα που θέλει και πατάει αποθήκευση.
- 10) Το σύστημα αποθηκεύει τις αλλαγές στη βάση και επιστρέφει το χρήστη στη σελίδα «Αρχική Σελίδα Χρήστη»

Εναλλακτική Ροή 1

4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.

4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.

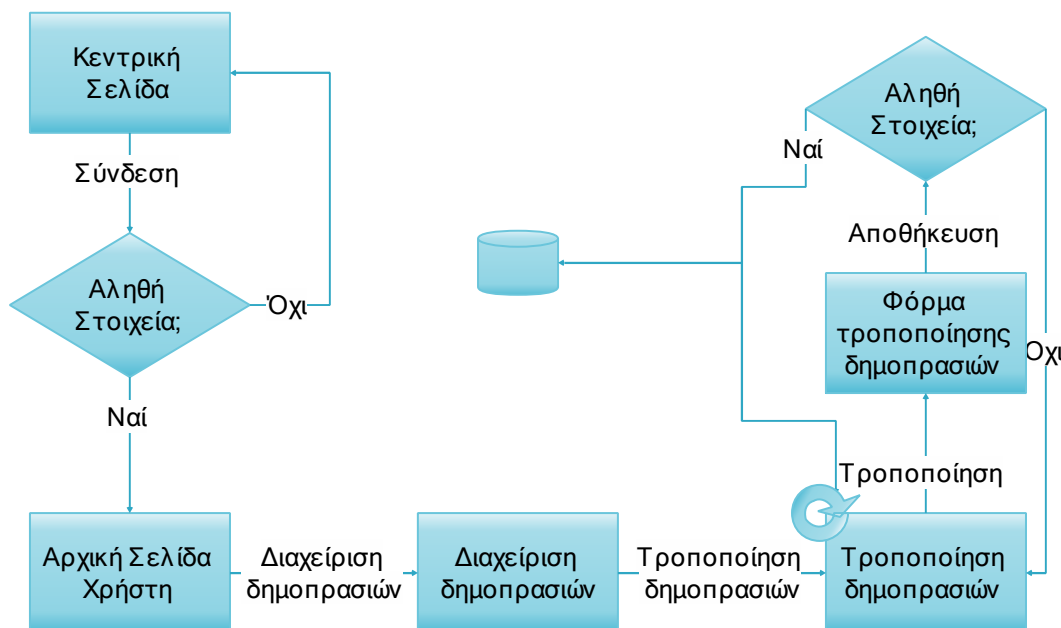
4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής

Εναλλακτική Ροή 2

10.α.1. Τα στοιχεία που έχει συμπληρώσει δεν είναι έγκυρα.

10.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους στη φόρμα, ότι τα στοιχεία δεν είναι αποδεκτά και ότι η δημοπρασία δεν μπορεί να τροποποιηθεί και ζητάει από το χρήστη να βάλει έγκυρα στοιχεία.

10.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 9 της βασικής ροής.



3.3.10 Προβολή συγκεντρωτικών αναφορών

Σκοπός

Ο σκοπός του χρήστη σε αυτή την περίπτωση είναι να αποκτήσει πρόσβαση στα συγκεντρωτικά στοιχεία που παρέχονται στην σελίδα μας. Δηλαδή, να δει τις συγκεντρωτικές αναφορές για:

- α) Πόσες δημοπρασίες ήταν άγονες
- β) Πόσο είναι ο μέσος χρόνος μια δημοπρασίας.
- γ) Ποια προϊόντα δεν πωλήθηκαν λόγω άγονων δημοπρασιών και ποια ήταν η μεγαλύτερη προσφορά για το καθένα και ποια η εναρκτήρια τιμή της αντίστοιχης δημοπρασίας.

Βασική ροή

- 1) Ο χρήστης ανοίγει έναν περιηγητή διαδικτύου και επισκέπτεται τη σελίδα μας.
- 2) Το σύστημα εμφανίζει την «Κεντρική Σελίδα» της ιστοσελίδας μας.
- 3) Ο χρήστης συμπληρώνει τα στοιχεία στη φόρμα «Login» και πατάει σύνδεση
- 4) Το σύστημα συνδέει το χρήστη στην σελίδα «Αρχική Σελίδα Χρήστη».

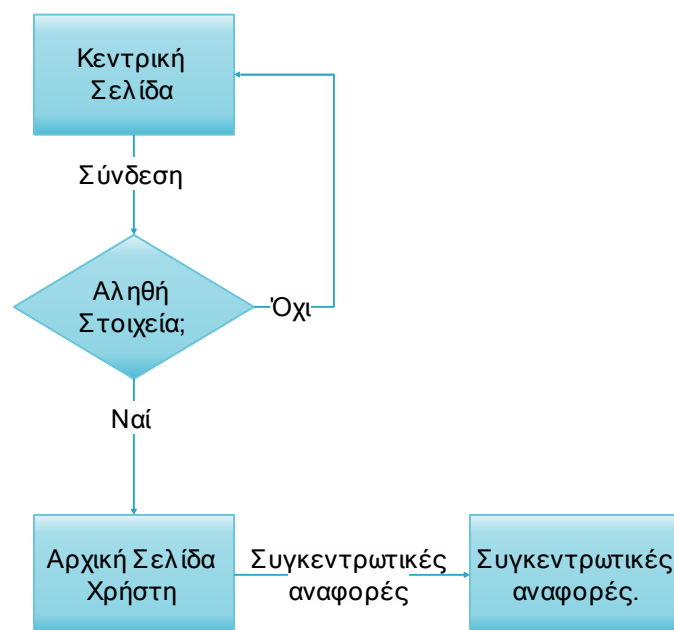
- 5) Ο χρήστης επιλέγει το σύνδεσμο «Συγκεντρωτικές αναφορές» από το μενού της σελίδας.
- 6) Το σύστημα εμφανίζει στην σελίδα τις διαθέσιμες συγκεντρωτικές αναφορές.

Εναλλακτική Ροή 1

4.α.1. Ο Χρήστης δεν έχει συμπληρώσει κάποιο ή και τα δυο από τα στοιχεία του είτε τα έχει συμπληρώσει λανθασμένα.

4.α.2. Το σύστημα εμφανίζει το αντίστοιχο μήνυμα λάθους ότι τα στοιχεία δεν είναι σωστά και να τα εισάγει ξανά.

4.α.3. Η περίπτωση χρήσης συνεχίζει από το βήμα 3 της βασικής ροής.

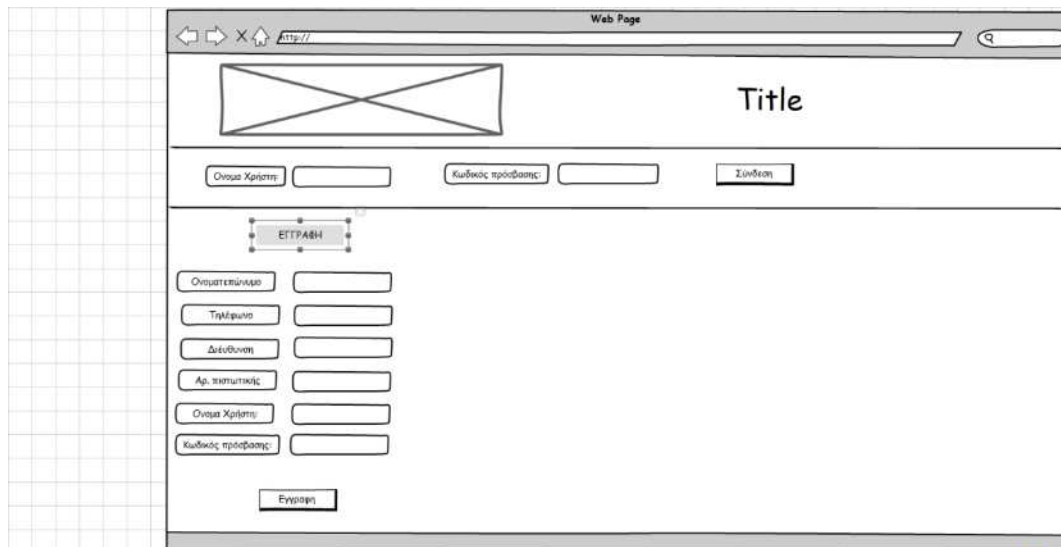


3.4 Σχεδιαγράμματα σελίδων

Εδώ θα παρουσιάσουμε τα σχεδιαγράμματα των σελίδων που σχεδιάστηκαν μετά την ανάλυση των απαιτήσεων.

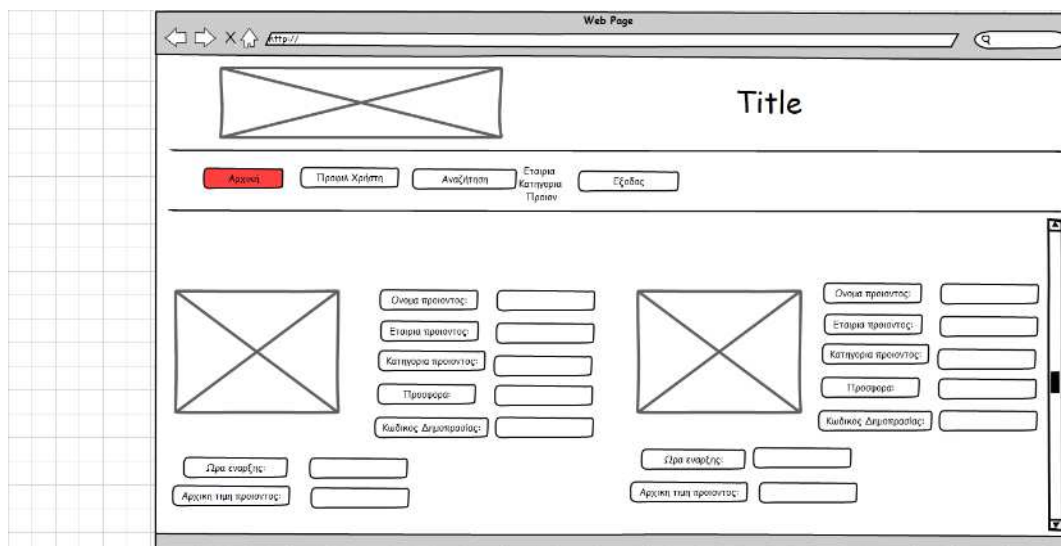
3.4.1 Κεντρική σελίδα

Η κεντρική σελίδα δίνει στον χρήστη τη δυνατότητα να κάνει login στο σύστημα συμπληρώνοντας τα στοιχεία του ή να εγγραφεί με τη συμπλήρωση της κατάλληλης φόρμας. Περιέχει την επικεφαλίδα στις σελίδας με το λογότυπο της εφαρμογής.



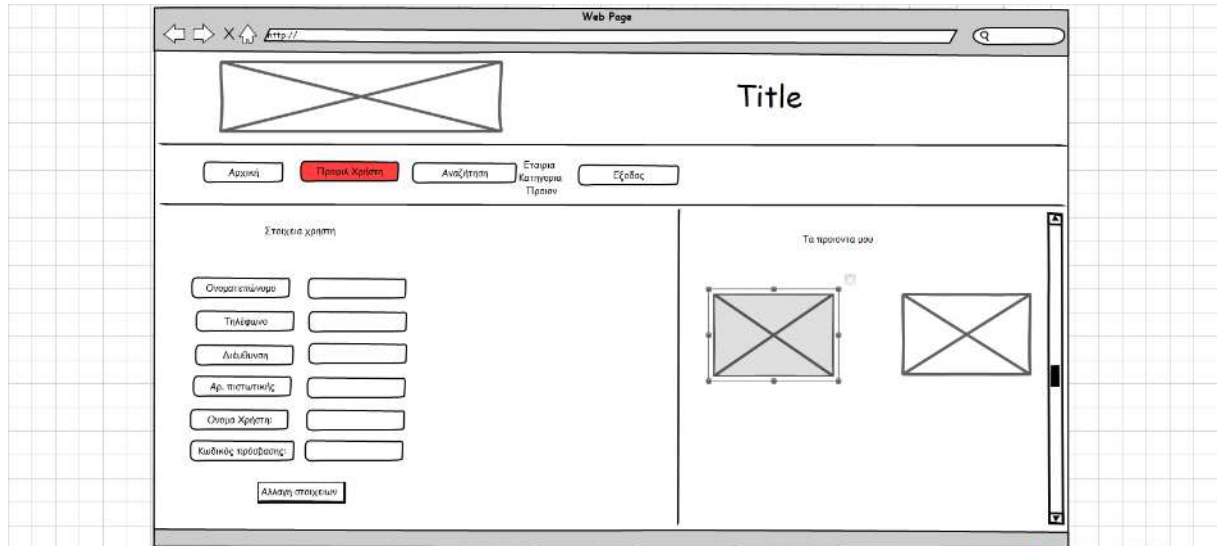
3.4.2 Αρχική σελίδα χρήστη

Η αρχική σελίδα χρήστη, που παρουσιάζεται μετά το login εμφανίζει στον χρήστη τα σχετικά προϊόντα. Επιπλέον δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα. Περιέχει την επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



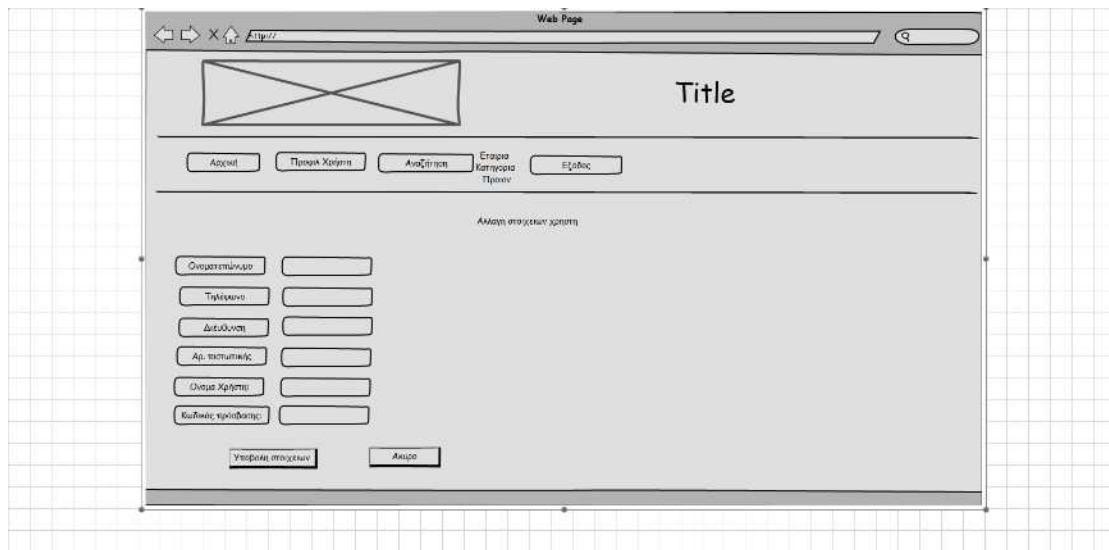
3.4.3 Προφίλ χρήστη

Η σελίδα προφίλ δίνει την δυνατότητα αλλαγής των στοιχείων του χρήστη και προβολής των προϊόντων του. Επιπλέον δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα. Περιέχει την επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



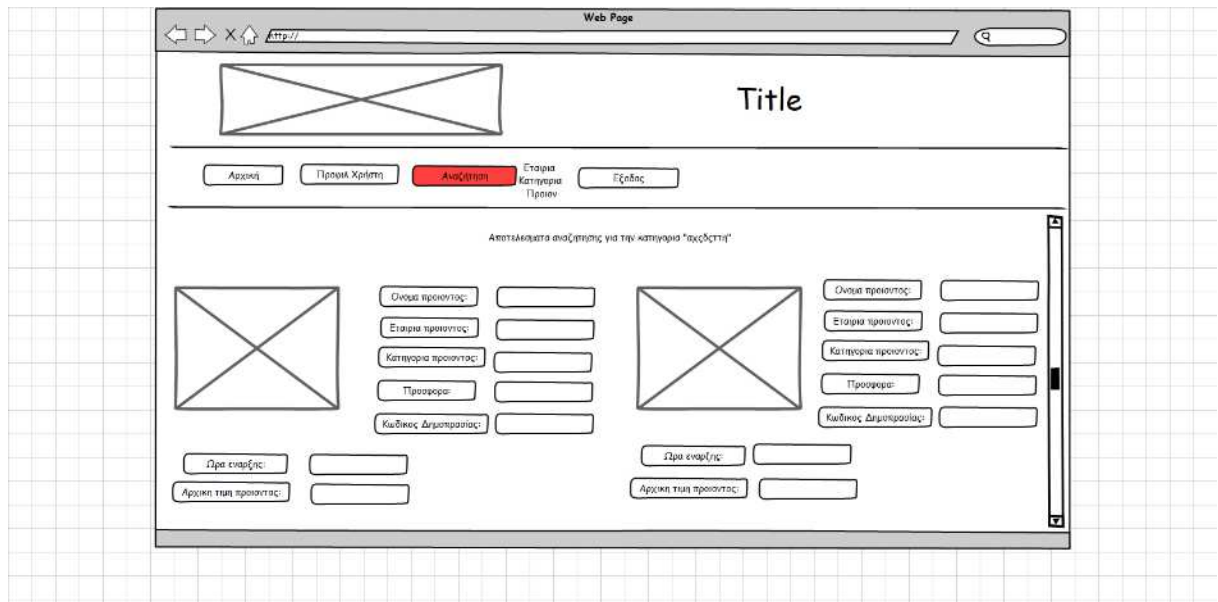
3.4.4 Τροποποίηση στοιχείων χρήστη

Αυτή η σελίδα περιέχει την φόρμα με τα στοιχεία του χρήστη, δίνοντας την δυνατότητα τροποποίησης τους και υποβολής των αλλαγών. Επιπλέον δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα. Περιέχει την επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



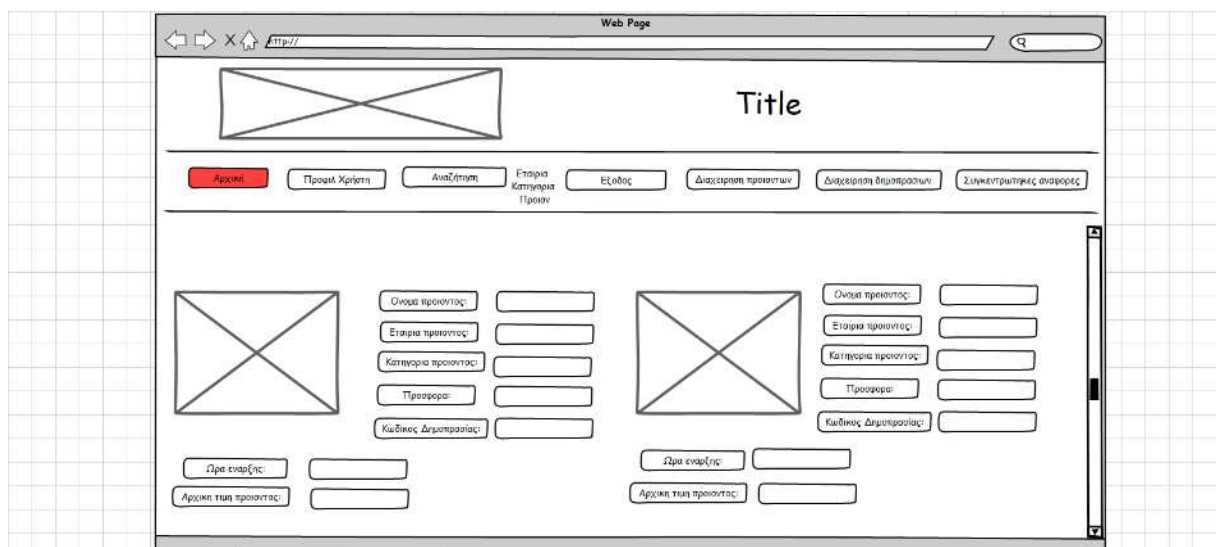
3.4.5 Αναζήτηση δημοπρασιών

Εδώ παρουσιάζονται οι δημοπρασίες που ανταποκρίνονται στα κριτήρια που έθεσε ο χρήστης. Όπως και πριν η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα. Περιέχει την επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



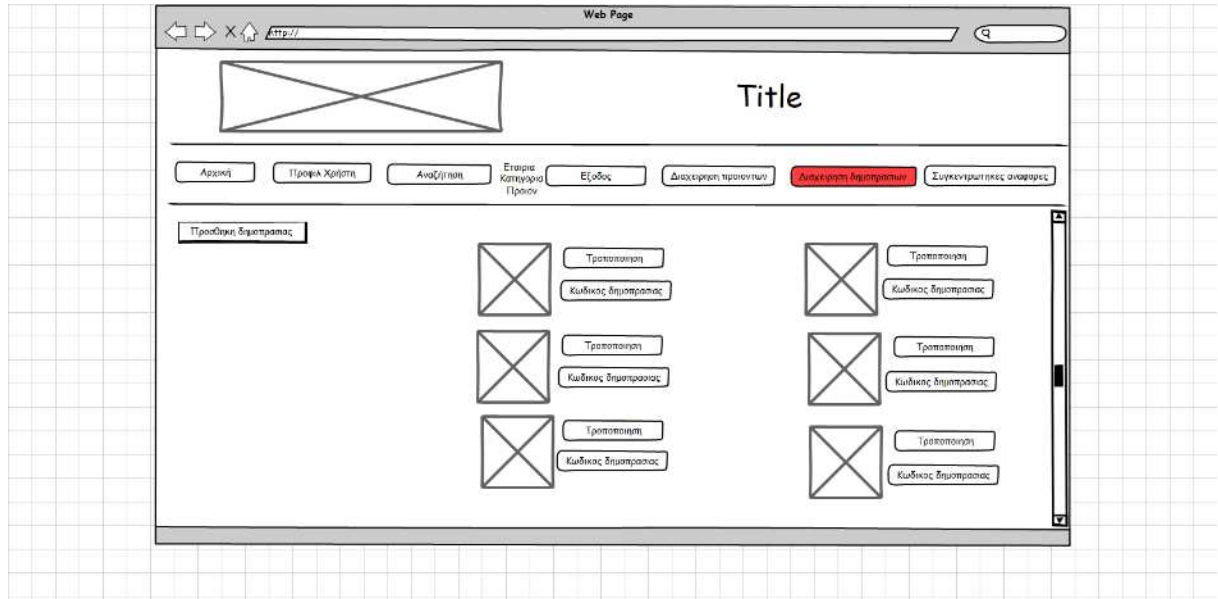
3.4.6 Αρχική σελίδα διαχειριστή

Ο διαχειριστής με την είσοδό του στο σύστημα βλέπει τις αρχικές δημοπρασίες. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



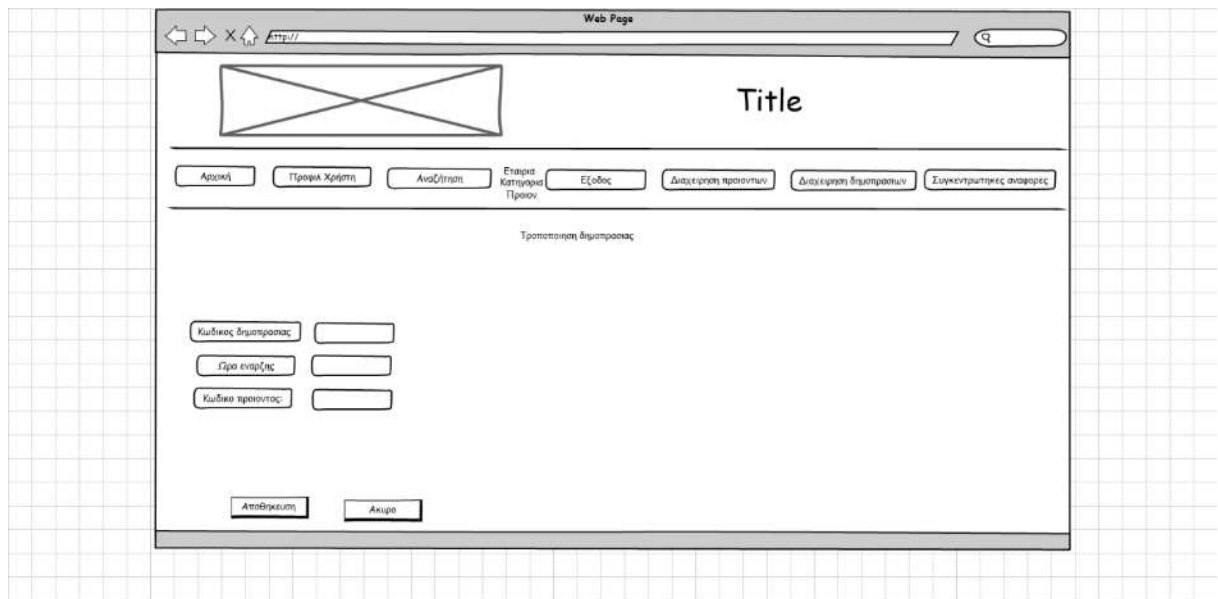
3.4.7 Διαχείριση δημοπρασίας

Ο διαχειριστής βλέπει μία λίστα με τις δημοπρασίες και τους κωδικούς τους, με δυνατότητα τροποποίησης της κάθε μίας. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



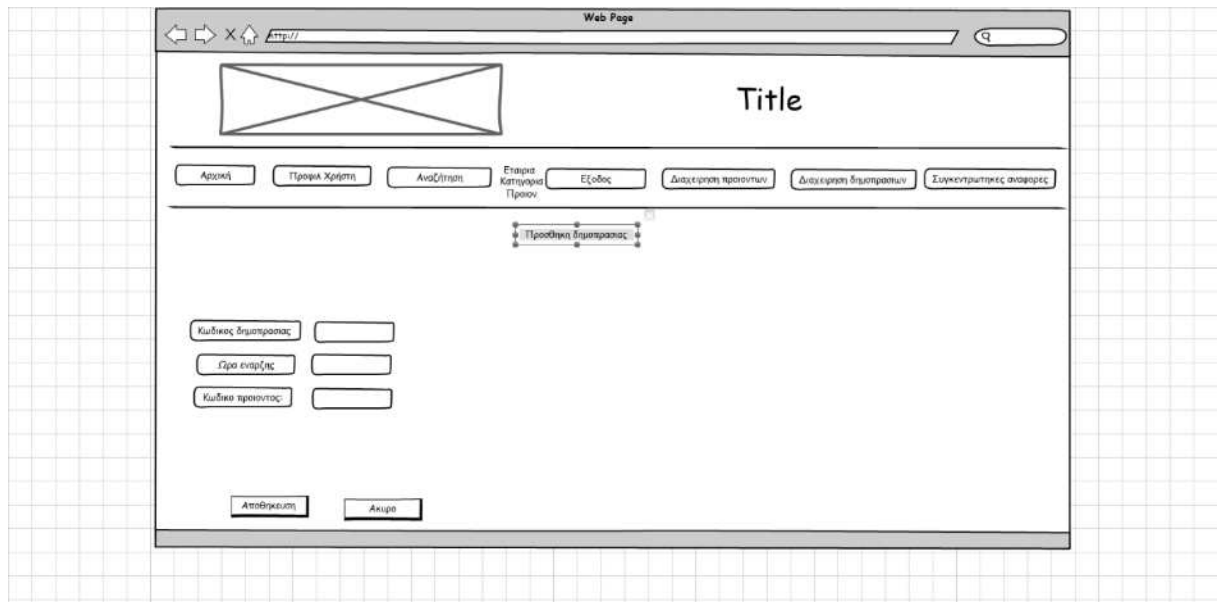
3.4.8 Τροποποίηση δημοπρασίας

Παρουσιάζεται μία φόρμα με τα στοιχεία της δημοπρασίας, τα στοιχεία της οποίας μπορούν να τροποποιηθούν και να αποθηκευτούν. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



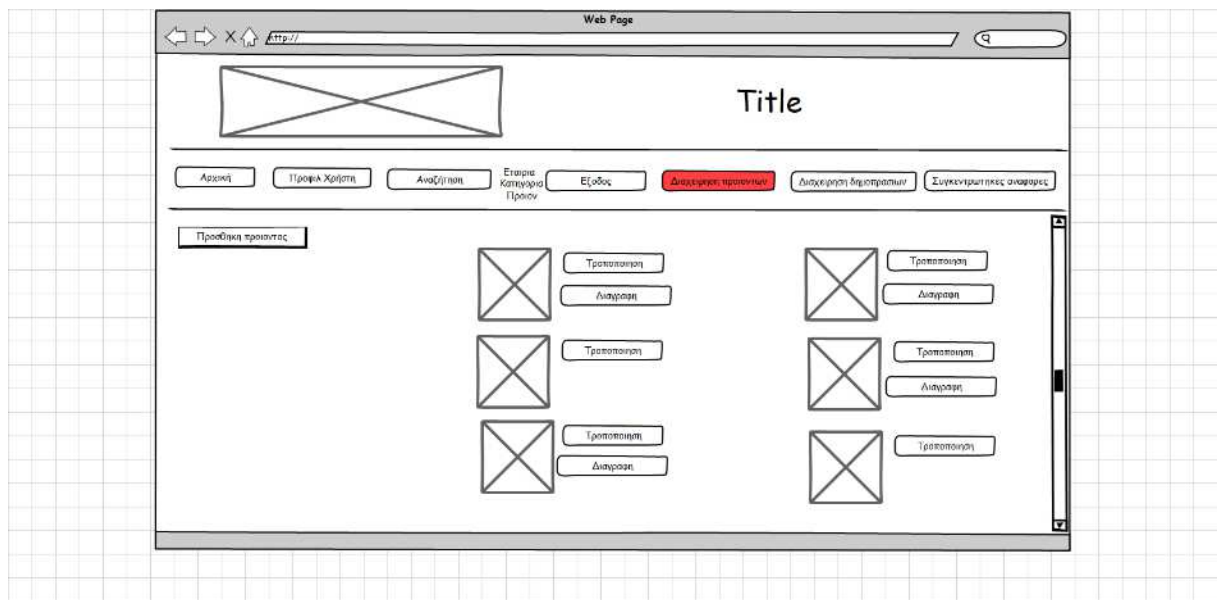
3.4.9 Προσθήκη δημοπρασίας

Παρουσιάζεται η φόρμα με τα στοιχεία της νέας δημοπρασίας, τα στοιχεία της οποίας μπορούν να τροποποιηθούν και να αποθηκευτούν. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



3.4.10 Διαχείριση προϊόντων

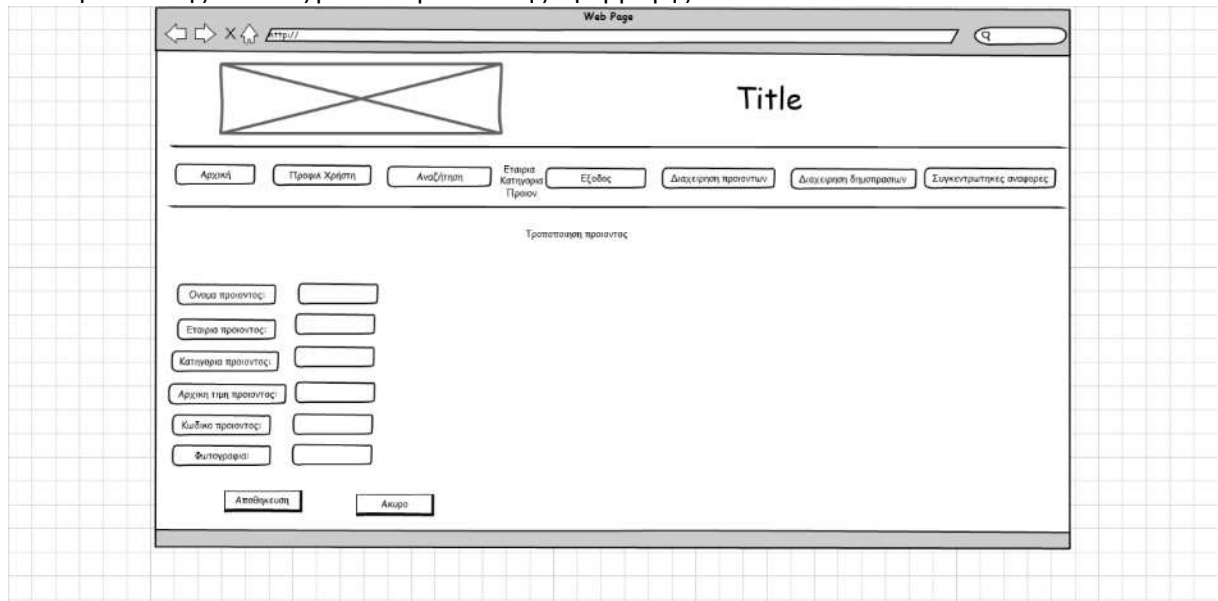
Παρουσιάζεται μία λίστα με τις υπάρχουσες δημοπρασίες, κάθε μία από τις οποίες έχει επιλογή για τροποποίηση των στοιχείων της. Επίσης δίνεται επιλογή για την προσθήκη μίας νέας δημοπρασίας. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



3.4.11 Τροποποίηση προϊόντων

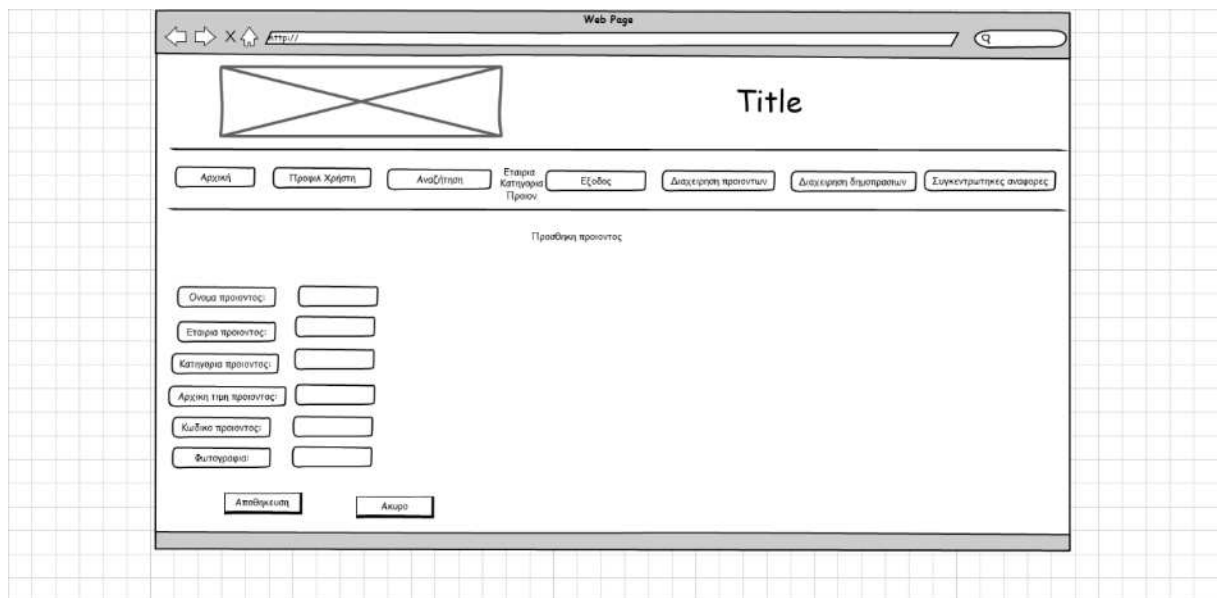
Παρουσιάζεται μία φόρμα με τα στοιχεία του προϊόντος, με δυνατότητα αποθήκευσης των αλλαγών. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα

διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



3.4.12 Προσθήκη προϊόντων

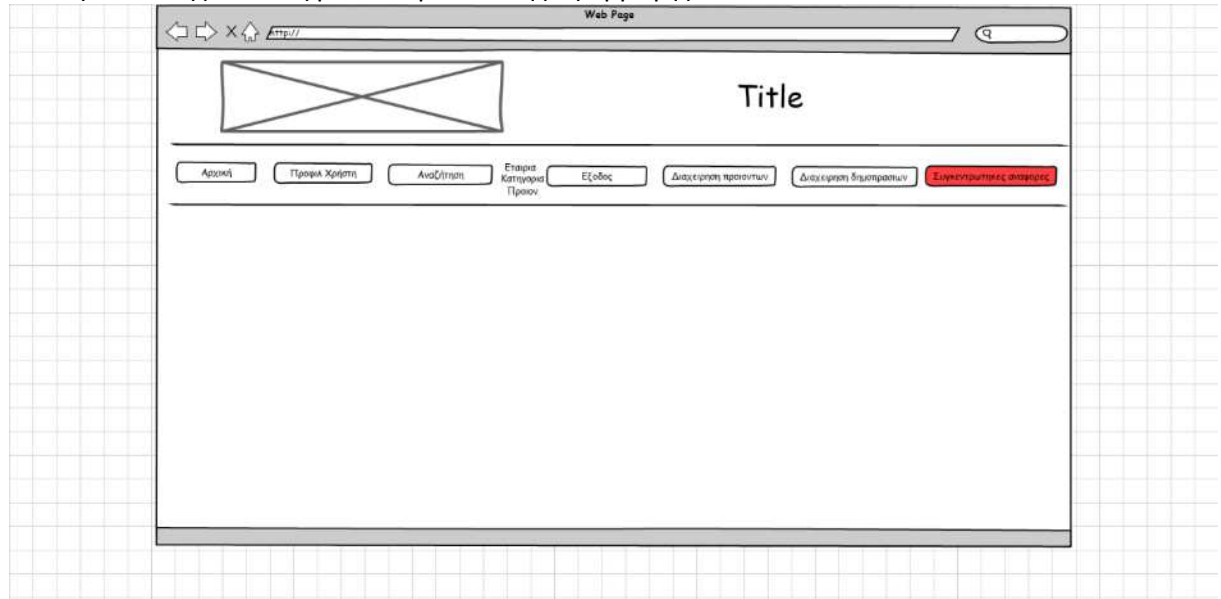
Παρουσιάζεται μία φόρμα με τα πεδία που χρειάζεται να αποθηκευτούν για την αποθήκευση ενός νέου προϊόντος. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης, προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



3.4.13 Συγκεντρωτικές αναφορές

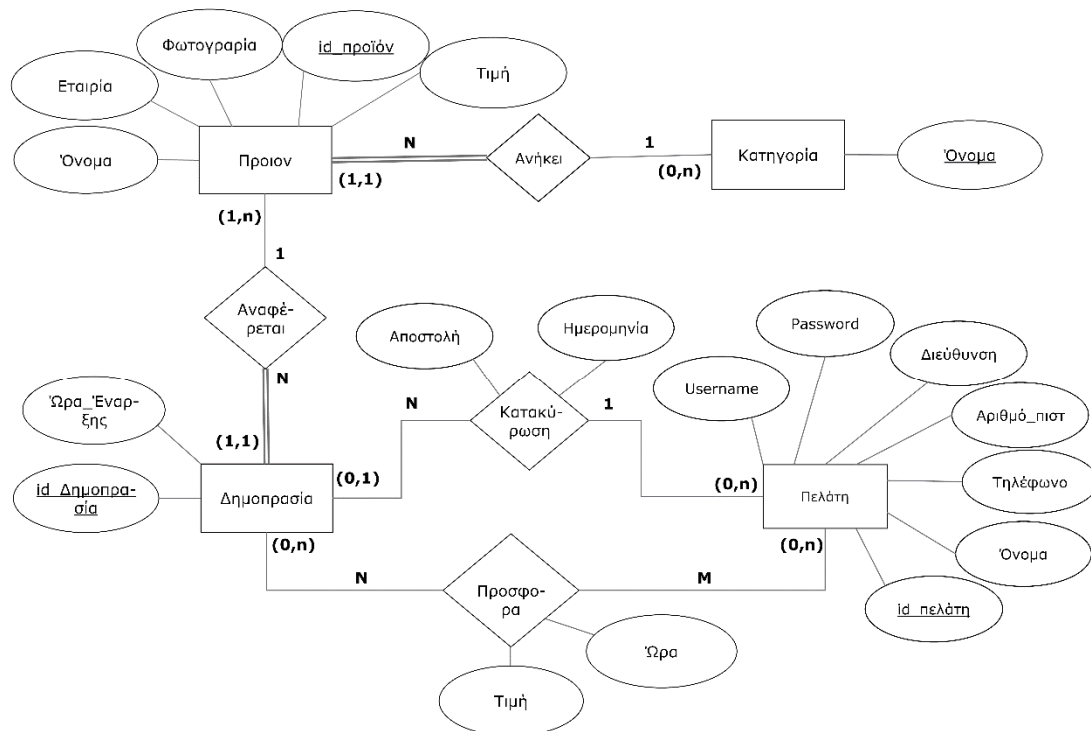
Παρουσιάζεται η σελίδα που περιέχει τα συγκεντρωτικά στοιχεία για όλες τις δημοπρασίες και τα προϊόντα. Η στάνταρ κεφαλίδα δίνει δυνατότητα αναζήτησης,

προβολής του προφίλ και εξόδου από το σύστημα, ενώ επιπλέον έχει πρόσβαση στη σελίδα διαχείρισης προϊόντων, δημοπρασιών και στις συγκεντρωτικές αναφορές. Περιέχεται η επικεφαλίδα της σελίδας με το λογότυπο της εφαρμογής.



3.5 Μοντέλο οντοτήτων - σχέσεων

Από την ανάλυση των απαιτήσεων της εφαρμογής προκύπτει το παρακάτω μοντέλο οντοτήτων - σχέσεων.



Συγκεκριμένα οι απαιτήσεις που ικανοποιούνται από αυτό είναι:

- Κάθε προϊόν ανήκει σε μία κατηγορία.

- Κάθε κατηγορία μπορεί να έχει πολλά προϊόντα.
- Κάθε προϊόν μπορεί να αναφέρεται σε πολλές δημοπρασίες.
- Κάθε δημοπρασία αναφέρεται σε ένα προϊόν.
- Ένας πελάτης μπορεί να κάνει μία προσφορά για πολλά προϊόντα.
- Ένα προϊόν μπορεί να έχει προσφορές από πολλούς πελάτες.
- Κάθε προσφορά έχει την τιμή που προσφέρθηκε και την ώρα που έγινε.
- Μία δημοπρασία κατοχυρώνεται το πολύ σε έναν πελάτη.
- Ένας πελάτης μπορεί να κατοχυρώσει καμία, μία ή περισσότερες δημοπρασίες.
- Κάθε κατοχύρωση χαρακτηρίζεται από την ημερομηνία και το αν έχει αποσταλεί.

4 Ανάπτυξη εφαρμογής

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δούμε τα βήματα που ακολουθήσαμε για την διαδικασία υλοποίησης της εφαρμογής, από την ρύθμιση του περιβάλλοντος και την δημιουργία της βάσης δεδομένων μέχρι και την δομή των αρχείων και τα σημαντικά σημεία του κώδικα.

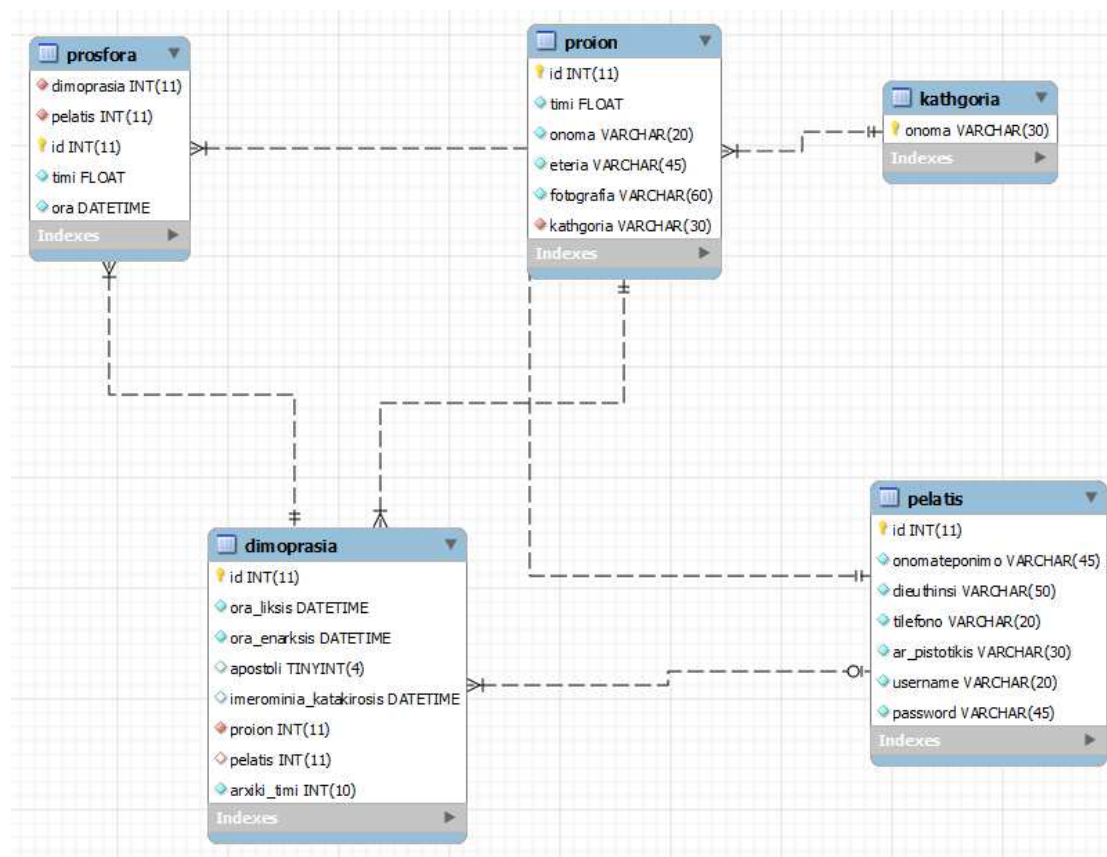
4.2 Περιβάλλον ανάπτυξης

Η εφαρμογή είναι γραμμένη στην γλώσσα PHP, ενώ αποθηκεύει τα δεδομένα σε μία βάση δεδομένων.

Χρησιμοποιήσαμε τον Apache web server σε συνδυασμό με την PHP ώστε να εκτελούνται τα script σε κάθε σελίδα. Η βάση δεδομένων που χρησιμοποιήσαμε είναι η MySQL.

4.3 Βάση δεδομένων

Το μοντέλο οντοτήτων – σχέσεων που προέκυψε από την ανάλυση των απαιτήσεων της εφαρμογής μετατράπηκε σε σχεσιακό μοντέλο και έπειτα σε πίνακες που εισήγαμε στη βάση. Προέκυψε το παρακάτω μοντέλο.



Τα περιεχόμενα κάθε πίνακα παρουσιάζονται αναλυτικά παρακάτω.

4.3.1 Πίνακας δημοπρασιών

Ο πίνακας dimoprasia είναι ο πίνακας στον οποίο αποθηκεύονται τα στοιχεία κάθε δημοπρασίας.

Name: dimoprasia
Comment:

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsig...	Allow ...	Zer...	Default	Comment	Collation
1	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREME...		
2	ora_liksis	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default		
3	ora_enarksis	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default		
4	apostoli	TINYINT	4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		
5	imerominia...	DATETIME		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		
6	proion	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default		
7	pelatis	INT	11	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		
8	arxiki_timi	INT	10	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default		

Τα πεδία του είναι:

- **Id:** ο κωδικός της δημοπρασίας και το πρωτεύον κλειδί.
- **Ora_liksis:** η ώρα λήξης της δημοπρασίας.
- **Ora_enarksis:** η ώρα έναρξης της δημοπρασίας.
- **Apostoli:** εάν έχει γίνει αποστολή της δημοπρασίας.
- **Imerominia_katakirosis:** δείχνει την ημερομηνία και την ώρα που έχει γίνει κατακύρωση στο προϊόν.
- **Proion:** είναι δευτερεύον κλειδί και δείχνει πιο προϊόν είναι κατακυρωμένο.
- **Pelatis:** είναι δευτερεύον κλειδί και δείχνει σε ποιον πελάτη είναι κατακυρωμένο.
- **Arxiki_timi:** είναι η αρχική τιμή του προϊόντος.

4.3.1.1 Ορισμός SQL

Ο κώδικας SQL που χρειάζεται για την δημιουργία του πίνακα είναι ο παρακάτω:

```
CREATE TABLE IF NOT EXISTS `dimoprasia` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `ora_liksis` datetime NOT NULL,
  `ora_enarksis` datetime NOT NULL,
  `apostoli` tinyint(4) DEFAULT NULL,
  `imerominia_katakirosis` datetime DEFAULT NULL,
  `proion` int(11) NOT NULL,
  `pelatis` int(11) DEFAULT NULL,
  `arxiki_timi` int(10) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_dimoprasia_proion1_idx` (`proion`),
  KEY `fk_dimoprasia_pelatis1_idx` (`pelatis`),
  CONSTRAINT `fk_dimoprasia_pelatis1` FOREIGN KEY (`pelatis`)
REFERENCES `pelatis` (`id`) ON UPDATE CASCADE,
  CONSTRAINT `fk_dimoprasia_proion1` FOREIGN KEY (`proion`)
REFERENCES `proion` (`id`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.3.2 Πίνακας κατηγοριών

Ο πίνακας *kathgoria* περιέχει τα στοιχεία κάθε κατηγορίας.

Name: kathgoria
Comment:

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsig...	Allow...	Zer...	Default	Comment	Collation	Expressic
1	onoma	VARCHAR	30		<input type="checkbox"/>		No default			

Τα πεδία του είναι:

- **Onoma:** είναι κλειδί του πίνακα και περιέχει τις κατηγορίες που ανήκουν τα προϊόντα δημοπρατούνται.

4.3.2.1 Ορισμός SQL

Ο κώδικας SQL που χρειάζεται για την δημιουργία του πίνακα είναι ο παρακάτω:

```
CREATE TABLE IF NOT EXISTS `kathgoria` (  
  `onoma` varchar(30) NOT NULL,  
  PRIMARY KEY (`onoma`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.3.3 Πίνακας πελατών

Ο πίνακας *relatis* περιέχει τα στοιχεία των πελατών. Έχει την παρακάτω μορφή:

Name: relatis
Comment:

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsig...	Allow...	Zer...	Default	Comment
1	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREME...	
2	onomatepo...	VARCHAR	45		<input type="checkbox"/>		No default	
3	dieuthinsi	VARCHAR	50		<input type="checkbox"/>		No default	
4	tilefono	VARCHAR	20		<input type="checkbox"/>		No default	
5	ar_pistotikis	VARCHAR	30		<input type="checkbox"/>		No default	
6	username	VARCHAR	20		<input type="checkbox"/>		No default	
7	password	VARCHAR	45		<input type="checkbox"/>		No default	

Τα πεδία του είναι:

- **Id:** είναι το κλειδί του πίνακα και είναι ο κωδικός του χρήστη.
- **Onomateponimo:** είναι το ονοματεπώνυμο του χρήστη.
- **Dieuthinsi:** είναι η διεύθυνση του χρήστη.
- **Tilefono:** το τηλέφωνο του χρήστη.
- **Ar_pistotikis:** ο αριθμός πιστωτικής του χρήστη.
- **Username:** είναι το όνομα που χρησιμοποιεί ο χρήστης για την εισαγωγή του στο σύστημα.
- **Password:** είναι το συνθηματικό που χρησιμοποιεί ο χρήστης για την εισαγωγή του στο σύστημα.

4.3.3.1 Ορισμός SQL

Ο κώδικας SQL που χρειάζεται για την δημιουργία του πίνακα είναι ο παρακάτω:

```
CREATE TABLE IF NOT EXISTS `pelatis` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `onomateponimo` varchar(45) NOT NULL,  
  `dieuthinsi` varchar(50) NOT NULL,  
  `tilefono` varchar(20) NOT NULL,  
  `ar_pistotikis` varchar(30) NOT NULL,  
  `username` varchar(20) NOT NULL,  
  `password` varchar(45) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `username_UNIQUE` (`username` )  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.3.4 Πίνακας προϊόντων

Ο πίνακας προϊόντων περιέχει εγγραφές με τα προϊόντα που δημοπρατούνται. Έχει την μορφή που φαίνεται στο σχήμα.

Name: proion
Comment:

Columns: Add Remove Up Down

#	Name	Datatype	Length/Set	Unsig...	Allow ...	Zer...	Default	Comment
1	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREME...	
2	timi	FLOAT		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	
3	onoma	VARCHAR	20		<input type="checkbox"/>		No default	
4	eteria	VARCHAR	45		<input type="checkbox"/>		No default	
5	fotografia	VARCHAR	60		<input type="checkbox"/>		No default	
6	kathgoria	VARCHAR	30		<input type="checkbox"/>		No default	

Τα πεδία του είναι:

- **Id:** είναι το κλειδί του πίνακα και ο κωδικός του προϊόντος
- **Timi:** η τιμή του προϊόντος
- **Onoma:** το όνομα του προϊόντος.
- **Eteria:** η εταιρία στην οποία ανήκει το προϊόν.
- **Fotografia:** φωτογραφία από το προϊόν.
- **Kathgoria:** η κατηγορία στην οποία ανήκει.

4.3.4.1 Ορισμός SQL

Ο κώδικας SQL που χρειάζεται για την δημιουργία του πίνακα είναι ο παρακάτω:

```
CREATE TABLE IF NOT EXISTS `proion` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `timi` float NOT NULL,  
  `onoma` varchar(20) NOT NULL,  
  `eteria` varchar(45) NOT NULL,  
  `fotografia` varchar(60) NOT NULL,  
  `kathgoria` varchar(30) NOT NULL,  
  PRIMARY KEY (`id`),  
  KEY `fk_proion_kathgoria_idx` (`kathgoria`),  
  CONSTRAINT `fk_proion_kathgoria` FOREIGN KEY (`kathgoria`)  
  REFERENCES `kathgoria` (`onoma`) ON UPDATE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.3.5 Πίνακας προσφορών

Ο πίνακας prosfora περιέχει τις προσφορές που κάνουν οι χρήστες για μία δημοπρασία. Έχει την παρακάτω μορφή.

Name: prosfora

Comment:

#	Name	Datatype	Length/Set	Unsig...	Allow ...	Zer...	Default	Comm
1	dimoprasia	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	
2	pelatis	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	
3	id	INT	11	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREME...	
4	timi	FLOAT		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	
5	ora	DATETIME		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	

Τα πεδία του είναι:

- **Dimoprasia:** ο κωδικός της δημοπρασίας είναι και δευτερεύον κλειδί.
- **Pelatis:** ο κωδικός της του πελάτη είναι και δευτερεύον κλειδί.
- **Id:** ο κωδικός του πίνακα προσφορών είναι και πρωτεύον κλειδί.
- **Timi:** είναι η τιμές που κάνουν προσφορά για την δημοπρασία ο κάθε χρήστης.
- **Ora:** είναι η ώρα που έκανε την προσφορά ο χρήστης.

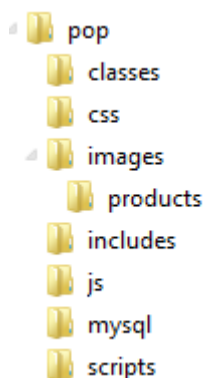
4.3.5.1 Ορισμός SQL

Ο κώδικας SQL που χρειάζεται για την δημιουργία του πίνακα είναι ο παρακάτω:

```
CREATE TABLE IF NOT EXISTS `prosfora` (
  `dimoprasia` int(11) NOT NULL,
  `pelatis` int(11) NOT NULL,
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `timi` float NOT NULL,
  `ora` datetime NOT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_dimoprasia_has_pelatis_pelatis1_idx` (`pelatis`),
  KEY `fk_dimoprasia_has_pelatis_dimoprasia1_idx`
(`dimoprasia`),
  CONSTRAINT `fk_dimoprasia_has_pelatis_dimoprasia1` FOREIGN
KEY (`dimoprasia`) REFERENCES `dimoprasia` (`id`) ON UPDATE
CASCADE,
  CONSTRAINT `fk_dimoprasia_has_pelatis_pelatis1` FOREIGN KEY
(`pelatis`) REFERENCES `pelatis` (`id`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

4.4 Δομή εφαρμογής

Καθώς η εφαρμογή αποτελείται από αρκετά αρχεία τα έχουμε χωρίσει σε διαφορετικούς φακέλους. Η δομή φαίνεται στο παρακάτω σχήμα.



Θα αναλύσουμε για ποιον λόγο φτιάχτηκε κάθε φάκελος καθώς και τον κώδικα των αρχείων που περιέχει. Η εξαντλητική παρουσίαση του κώδικα δεν έχει ιδιαίτερο νόημα, οπότε απλά θα παρουσιάσουμε τα κύρια σημεία των αλγορίθμων παραλείποντας τα τμήματα που έχουν αναλυθεί προηγουμένως και επαναλαμβάνονται.

4.4.1 Φάκελος js

Ο φάκελος js είναι ένας φάκελος στο οποίο αποθηκεύονται τα αρχεία JavaScript που χρησιμοποιούμε για την εφαρμογή.

4.4.2 Φάκελος css

Ο φάκελος CSS κρατάει τα αρχεία που χρησιμοποιούνται για την εφαρμογή. Συγκεκριμένα χρησιμοποιούμε το main.css που έχουμε ορίσει τους κανόνες μορφοποίησης της εφαρμογής.

4.4.3 Φάκελος images

Στον φάκελο images αποθηκεύουμε τις εικόνες της εφαρμογής. Συγκεκριμένα στον φάκελο images αποθηκεύονται οι εικόνες της εφαρμογής ενώ υπάρχει ένας υποφάκελος, ο products. Στον φάκελο products αποθηκεύονται οι εικόνες των προϊόντων που φαίνονται στις δημοπρασίες.

4.4.4 Φάκελος mysql

Ο φάκελος mysql περιέχει κάποια αρχεία php που ασχολούνται με τις χαμηλού επιπέδου λειτουργίες της βάσης δεδομένων.

4.4.4.1 Αρχείο configuration.php

Το αρχείο configuration.php περιέχει τις πληροφορίες για την βάση δεδομένων. Συγκεκριμένα έχει το όνομα της βάσης, το όνομα και τον κωδικό χρήστη και τη διεύθυνση της βάσης.

```
<?php
    $dbhost = "localhost:3306";
    $dbuser = "root";
    $dbpass = "mysqladmin";
    $database_name = 'mydb';
?>
```

Αυτό το αρχείο χρησιμοποιείται έπειτα από όλα τα αρχεία που χρειάζονται τη βάση. Με αυτόν τον τρόπο αν χρειαστεί να γίνει μία αλλαγή στα αρχεία της βάσης αρκεί να τροποποιήσουμε ένα αρχείο και όχι όλα τα αρχεία που συνδέονται στην βάση.

4.4.4.2 Αρχείο *mysql_lib.php*

Στο αρχείο *mysql_lib.php* έχουμε κάποια κομμάτια κώδικα που χρησιμοποιούνται σε όλες τις λειτουργίες της βάσης δεδομένων.

Αρχικά το αρχείο εισάγει το *configuration.php* για να πάρει τα στοιχεία της βάσης.

```
include('configuration.php');
```

Έπειτα υπάρχει η συνάρτηση *opendb* που ανοίγει τη σύνδεση με τη βάση.

```
function opendb(){
    $conn = mysql_connect($dbhost,$dbuser,$dbpass) or die
('Error connecting to mysql');
    if ($conn == false) die ('Error connecting to mysql');
    $result = mysql_select_db($database_name);
    if ($result == false) {
        echo ".mysql_error();
        mysql_close($conn);
        die();
    }
    return $conn;
}
```

Όπως φαίνεται στα στοιχεία της βάσης χρησιμοποιούνται οι μεταβλητές που έχουν δηλωθεί στο αρχείο ρυθμίσεων.

Έπειτα υπάρχει η συνάρτηση *closedb* που κλείνει την σύνδεση.

```
function closedb($conn){
    if ($conn != false)
        mysql_close($conn);
    else
        die('Connection is already closed!');
}
```

4.4.5 Φάκελος includes

Στον φάκελο includes υπάρχουν τα αρχεία που περιέχουν κομμάτια κώδικα που χρησιμοποιούνται σε πολλά άλλα σημεία της εφαρμογής. Σκοπός είναι ο περιορισμός των επαναλαμβανόμενων κομματιών κώδικα, με σκοπό τόσο τον περιορισμό των λαθών από την συνεχή αντιγραφή τους όσο και την εύκολη διόρθωση ή τροποποίησή τους.

4.4.5.1 Αρχείο header.inc

Κάθε σελίδα της εφαρμογής έχει στην αρχή της ένα στάνταρ κομμάτι κώδικα με την επικεφαλίδα της HTML και τα βασικά tags που δίνουν την μορφή της. Αυτό τον κώδικα τον περιλαμβάνουμε στο αρχείο header.inc, το οποίο έπειτα κάνουμε include στην αρχή κάθε σελίδας.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8" />
<title> <?php echo $pagetitle; ?> </title>
<link rel="stylesheet" type="text/css" href="./css/main.css"
/>
<script src="./js/helper.js"></script>
</head>
<body>
<div id="wrapper">
<div id="top">
    
    <div id="site-title">Σύστημα Ηλεκτρονικής Διαχείρισης
    Δημοπρασιών - ΤΕΙ Κρήτης</div>
    <div style="clear:both"></div>
</div>
<div id="pagetitle"><?php echo $pagetitle; ?></div>
<br/>
```

Παρόλο που πρόκειται για στατικές εντολές πρέπει να δοθεί βάρος στην γραμμή που βάζει την επικεφαλίδα.

```
<div id="pagetitle"><?php echo $pagetitle; ?></div>
```

Αυτή χρησιμοποιεί την μεταβλητή `pagetitle`, η οποία χρειάζεται να είναι ορισμένη πριν γίνει το `include` ώστε να εμφανιστεί ο κατάλληλος τίτλος στο πάνω μέρος της σελίδας.

4.4.5.2 Αρχείο `bottom.inc`

Αντίστοιχα με το `header.inc`, στο τέλος κάθε σελίδας υπάρχει κώδικας που κλείνει τα tags που έχουν ανοίξει προηγουμένως. Για να αποφύγουμε και πάλι την επανάληψη κώδικα και να διευκολύνουμε τυχόν αλλαγές έχουμε δημιουργήσει το `bottom.inc` με τις αντίστοιχες εντολές, το οποίο χρειάζεται να γίνεται `include` στο τέλος του κώδικα HTML που παράγουμε.

```
<div id="bottomdiv">
  <a href='http://localhost/pop/feed.xml'>
    <img src='http://localhost/pop/images/rss.png' />
  </a>
</div>
</div>
</body>
</html>
```

Όπως φαίνεται παραπάνω σε κάθε σελίδα φαίνεται και ο σύνδεσμος με το XML feed των δημοπρασιών που περιέχεται σε κάθε σελίδα.

4.4.5.3 Αρχείο `search_form.inc.php`

Το μενού αναζήτησης έχει μπει στο `search_form.inc.php`. Σε αντίθεση με τα αρχεία `header` και `bottom` αυτό περιέχει αρκετό κώδικα PHP και αντλεί δεδομένα από τη βάση.

Αρχικά συμπεριλαμβάνουμε το αρχείο `proionta.php` και κατασκευάζουμε ένα instance αυτού του αντικειμένου ώστε να συνδεθούμε.

```
include_once('./classes/proionta.php');
$proionta = new Proionta();
```

Έπειτα παίρνουμε τις εταιρείες με χρήση της αντίστοιχης συνάρτησης από το `Proionta`. Καθώς μπορεί αν γίνει κάποιο λάθος χρειάζεται να ελέγξουμε την ορθότητα του αποτελέσματος και αν έχει γίνει λάθος να σταματήσουμε την διαδικασία.

```
if ($proionta->getEtairies()) {  
    $lista_eterion = $proionta->getResults();  
} else {  
    $errors = $proionta->getErrors();  
    echo "<pre>";  
    var_dump($errors);  
    echo "</pre>";  
    die();  
}
```

Με την ίδια λογική παίρνουμε την λίστα με τα προϊόντα κάνοντας του ίδιους ελέγχους και αν όλα πάνε καλά τα αποθηκεύουμε σε μία μεταβλητή.

```
$lista_proionton = $proionta->getResults();
```

Επιπλέον το τρίτο μενού έχει τις κατηγορίες. Για αυτές χρειάζεται να κάνουμε include το αρχείο kathgoria και να δημιουργήσουμε μία τέτοια μεταβλητή.

```
include_once('./classes/kathgoria.php');  
$katigoriam = new Kathgoriam();
```

Έπειτα τις εισάγουμε σε μία μεταβλητή, κάνοντας τους ίδιους ελέγχους με πριν.

```
$lista_katigorion = $katigoriam->getResults();
```

Μετά μπορούμε να παράγουμε τον κατάλληλο κώδικα HTML. Για να βγάλουμε τα μενού με τα περιεχόμενα των παραπάνω μεταβλητών χρειάζεται να διατρέξουμε τα περιεχόμενά τους. Έτσι για τα προϊόντα έχουμε το εξής:

```
<select name="select_etairia">  
<?php  
    echo "<option value='-1'>Καμία επιλογή</option>";  
    for ($i=0;$i<count($lista_eterion);$i++){  
        echo "<option>".$lista_eterion[$i]."</option>";  
    }  
<?>  
</select>
```


Όπως φαίνεται η πρώτη επιλογή, όπου ο χρήστης δεν έχει επιλέξει τίποτα δημιουργείται στατικά με τιμή -1. Με αυτόν τον τρόπο το αντιστοιχίζουμε σε ένα id που δεν είναι έγκυρο ώστε μετέπειτα να το αγνοήσουμε.

Μετά φτιάχνουμε το select με την λίστα των προϊόντων, όπου κάθε option έχει σαν value το id του προϊόντος και σαν κείμενο το όνομά του.

```
<select name="select_proion">
<?php
    echo "<option value='-1'>Καμία επιλογή</option>";
    for ($i=0;$i<count($lista_proionton);$i++){
        echo "<option
value='". $lista_proionton[$i]['id' ]. "'>". $lista_proionton[$i][
'onoma' ]. "</option>";
    }
?>
</select>
```

Τέλος κατασκευάζεται το select με την λίστα των κατηγοριών, όπου κάθε πεδίο έχει για περιεχόμενο το αντίστοιχο όνομα.

```
<select name="select_kategoria">
<?php
    echo "<option value='-1'>Καμία επιλογή</option>";
    for ($i=0;$i<count($lista_kategorion);$i++){
        echo "<option>". $lista_kategorion[$i]. "</option>";
    }
?>
</select>
```

4.4.5.4 Αρχείο *locate_current_auction.inc.php*

Το αρχείο *locate_current_auction.inc.php* βρίσκει την δημοπρασία που είναι ενεργή αυτή τη στιγμή και φορτώνει τα στοιχεία της.

Αρχικά παίρνει την τρέχουσα ώρα από ένα αντικείμενο DateTime και δημιουργεί ένα string με μία συγκεκριμένη μορφή.

```
$now = new DateTime();
$time=$now->format('Y-n-j G:i:s');
```

Αυτή η μορφή είναι η μορφή με την οποία η συνάρτηση *strtotime* μπορεί να μετατρέψει ένα string σε ακέραιο.

```
$time = strtotime($time);
```

Έπειτα διατρέχουμε τη λίστα με τις δημοπρασίες ώστε να βρούμε αυτή που έχει περάσει η ώρα έναρξής της. Όταν βρεθεί κρατάμε τη θέση της στον πίνακα με τις δημοπρασίες και το id της.

```
for($j=0;$j<count($auction_list);$j++){  
    $ora_enarksis =  
    strtotime($auction_list[$j]['ora_enarksis']);  
    if($time>$ora_enarksis) {  
        $trexousa = $j;  
        $trexousa_id = $auction_list[$j]['auction_id'];  
    }  
}
```

Τέλος αν βρέθηκε μία δημοπρασία κρατάμε την μεγαλύτερη προσφορά για να την χρησιμοποιήσουμε παρακάτω.

```
if ($trexousa > -1){  
    $dimoprasia->getMaxPrice($trexousa_id);  
    $megisti_timi = $dimoprasia->getResults();  
}
```

4.4.6 Φάκελος classes

Στον φάκελο classes αποθηκεύονται οι κλάσεις που έχουμε δημιουργήσει για την εφαρμογή. Αυτά είναι τα παρακάτω:

4.4.6.1 Κλάση *pelates.php*

Η κλάση *Pelates* περιέχει τις λειτουργίες που χρειάζονται για τους πελάτες του συστήματος σε σχέση με τη βάση δεδομένων.

Αρχικά συμπεριλαμβάνει το αρχείο σύνδεσης με τη βάση.

```
include_once("./mysql/mysql_lib.php");
```

Η κλάση περιέχει δύο εσωτερικές μεταβλητές *results* και *errors* με τους αντίστοιχους getters ώστε να μπορούμε να πάρουμε τις τιμές τους.

```
class Pelates {
    private $results;
    private $errors;

    public function getResults(){
        return $this->results;
    }

    public function getErrors(){
        return $this->errors;
    }
}
```

Αυτές τροποποιούνται μέσα από τις υπόλοιπες συναρτήσεις.

Κάθε λειτουργία που μπορεί να γίνει σε έναν πελάτη αναπαρίσταται από μία συνάρτηση που παίρνει τα αντίστοιχα δεδομένα, εκτελεί τις κατάλληλες εντολές PHP και αποθηκεύει τα αποτελέσματα, αν υπάρχουν, στην μεταβλητή result. Τυχών λάθη αποθηκεύονται στην μεταβλητή errors.

Η πρώτη λειτουργία είναι ο έλεγχος της εγκυρότητας των στοιχείων που δίνει ο χρήστης κατά το login. Αυτό γίνεται με την παρακάτω συνάρτηση, που παίρνει για στοιχεία το όνομα και τον κωδικό που έδωσε.

```
public function authenticate($username,$password)
```

Η συνάρτηση συνδέεται στη βάση και εφόσον δεν υπάρχουν λάθη εκτελεί την παρακάτω εντολή SQL ώστε να βρει τον πελάτη με αυτά τα στοιχεία.

```
SELECT * FROM pelatis WHERE username = '$username' AND
password = '$password'
```

Έπειτα αν δε βρεθεί τέτοιο αποτέλεσμα μπαίνει το κατάλληλο μήνυμα λάθους στη μεταβλητή errors και επιστρέφει false.

```
$this->errors[] = "Το όνομα ή ο κωδικός χρήστης που δώσατε
είναι λάθος!";
```

Αν βρεθεί τέτοιος πελάτης τότε αποθηκεύεται το id του στη μεταβλητή results και επιστρέφει true.

```
if(mysql_num_rows($rs)>0){  
    $row = mysql_fetch_assoc($rs);  
    $this->results = $row['id'];  
    return true;  
} else {
```

Η συνάρτηση register παίρνει για παράμετρο έναν πίνακα με τα στοιχεία του πελάτη και αν είναι έγκυρα τα αποθηκεύει στη βάση.

```
public function register($info) {  
...  
}
```

Πριν την εισαγωγή ελέγχεται αν ο πελάτης έχει διαλέξει μοναδικό όνομα, ενώ σε αντίθετη περίπτωση ενημερώνει για το λάθος που έγινε.

```
$query = "SELECT * FROM pelatis WHERE username =  
'$info[username]' ";  
if (!$rs = mysql_query($query)) {  
    echo "";  
    die('Πρόβλημα σύνδεσης με την βάση');  
} else {  
    $valid = true;  
    if(mysql_num_rows($rs)>0){  
        $this->errors[] = "Το όνομα χρήστη που επιλέξατε  
χρησιμοποιείται ήδη από άλλον χρήστη!";  
        $valid = false;  
    }  
}
```

Έπειτα ελέγχεται αν το όνομα και ο κωδικός περιέχουν πάνω από τέσσερα γράμματα.

```
if (strlen($info['username']) < 4) {  
    $this->errors[] = "Παρακαλώ επιλέξτε ένα όνομα χρήστη με  
τουλάχιστον 4 χαρακτήρες!";  
    $valid = false;  
}  
if (strlen($info['password']) < 4) {  
    $this->errors[] = "Παρακαλώ επιλέξτε έναν κωδικό χρήστη  
με τουλάχιστον 4 χαρακτήρες!";  
    $valid = false;  
}
```

Τέλος αν όλα πάνε καλά τα νέα στοιχεία αποθηκεύονται στη βάση.

```
INSERT INTO pelatis (onomateponimo, dieuthinsi, tilefono,  
ar_pistotikis, username, password)  
VALUES  
('$info[fullname]', '$info[address]', '$info[telephone]',  
 '$info[cardnumber]', '$info[username]', '$info[password]')
```

Η συνάρτηση getPelatis παίρνει ένα username και επιστρέφει τα στοιχεία ενός πελάτη.

```
public function getPelatis($username){  
...  
}
```

Η εντολή με την οποία επιλέγεται είναι η παρακάτω:

```
SELECT * FROM pelatis WHERE username = '$username'
```

Έπειτα αν βρεθεί τα στοιχεία του αποθηκεύονται σαν πίνακας με τα αντίστοιχα στοιχεία.

```
$row = mysql_fetch_assoc($rs);  
$this->results['onomateponimo'] = $row['onomateponimo'];  
$this->results['dieuthinsi'] = $row['dieuthinsi'];  
$this->results['tilefono'] = $row['tilefono'];  
$this->results['id'] = $row['id'];  
$this->results['ar_pistotikis'] = $row['ar_pistotikis'];
```

Η συνάρτηση updatePelatis παίρνει για παράμετρο το όνομα ενός πελάτη και έναν πίνακα με τα στοιχεία του.

```
public function updatePelatis($data, $username){  
...  
}
```

Αυτά εισάγονται στη βάση με την παρακάτω εντολή.

```
UPDATE pelatis
SET onomateponimo = '$data[name]',
    dieuthinsi = '$data[address]',
    tilefono = '$data[tilefono]',
    ar_pistotikis = '$data[cardnumber]'
WHERE username = '$username'
```

4.4.6.2 Κλάση *kathgoria.php*

Το αρχείο *kathgoria.php* περιέχει τον ορισμό της κλάσης *Kathgoria*, η οποία περιέχει τις λειτουργίες για τις κατηγορίες.

Αρχικά περιλαμβάνεται το αρχείο της κλάσης και ορίζονται όπως και παραπάνω οι μεταβλητές *errors* και *results*.

```
include_once("./mysql/mysql_lib.php");

class Kathgoria
{
    private $results;
    private $errors;
```

Η συνάρτηση *getAll* επιστρέφει έναν πίνακα με όλες τις κατηγορίες.

```
public function getAll(){
...
}
```

Η επιλογή τους γίνεται με την παρακάτω εντολή.

```
SELECT * FROM kathgoria
```

Έπειτα εισάγονται σαν πίνακας στο *results*.

```
$i=0;
while ($row = mysql_fetch_assoc($rs)){
    $this->results[$i] = $row['onoma'];
    $i++;
}
```

4.4.6.3 Κλάση *proionta.php*

Το αρχείο *proionta.php* περιέχει την δήλωση της κλάσης *Proionta* που περιέχει τις λειτουργίες που σχετίζονται με την εισαγωγή και την διαχείριση των προϊόντων. Όπως και παραπάνω περιέχει τις μεταβλητές *results* και *errors* και φυσικά κάνει *include* το αρχείο *mysql_lib.php*.

Η μέθοδος *getFutureProducts* επιστρέφει τα στοιχεία των προϊόντων που βρίσκονται σε μελλοντικές δημοπρασίες.

```
public function getFutureProducts(){  
...  
}
```

Για την εύρεσή τους χρειάζεται να πάρουμε την τωρινή ώρα και να την μετατρέψουμε σε μία μορφή που είναι κατανοητή από την MySQL.

```
$now = new DateTime();  
$time=$now->format('Y-n-j G:i:s');
```

Μετά επιλέγουμε αυτά τα προϊόντα. Χρειαζόμαστε τα στοιχεία του προϊόντος σε συνδυασμό με την δημοπρασία στην οποία ανήκει, ώστε να πάρουμε και την ώρα έναρξης του.

```
SELECT p.id, p.onoma, p.fotografia  
FROM dimoprasia AS d INNER JOIN proion AS p ON d.proion =  
p.id  
WHERE ora_enarksis>'$time'
```

Μετά αποθηκεύονται σαν πίνακας στο *results*. Διατρέχουμε κάθε γραμμή του αποτελέσματος, παίρνουμε το κλειδί της και το χρησιμοποιούμε σαν κλειδί της αντίστοιχης θέσης στο *results*.

```
$i=0;  
while ($row = mysql_fetch_assoc($rs)){  
    $keys = array_keys($row);  
    for($x = 0; $x < count($keys); $x++)  
    {  
        $this->results[$i][$keys[$x]] = $row[$keys[$x]];  
    }  
    $i++;  
}
```

Τέλος, αν δεν έχει γίνει λάθος επιστρέφεται *true*.

Η μέθοδος `getUnused` επιστρέφει τα προϊόντα που δεν χρησιμοποιούνται σε δημοπρασίες.

```
public function getUnused(){  
...  
}
```

Αυτά τα βρίσκουμε επιλέγοντας τα προϊόντα που τα κλειδιά τους δεν υπάρχουν στον πίνακα με τις δημοπρασίες

```
SELECT prod.id  
FROM proion AS prod  
WHERE NOT EXISTS (  
    SELECT * FROM dimoprasia WHERE dimoprasia.proion =  
    prod.id  
)
```

Τα id αυτών των προϊόντων εισάγονται στον πίνακα `results`.

```
$i=0;  
while ($row = mysql_fetch_assoc($rs)){  
    $this->results[$i] = $row['id'];  
    $i++;  
}
```

Τα προϊόντα που δεν έχουν πουληθεί σε μία δημοπρασία επιστρέφονται από την μέθοδο `getUnsaled`.

```
public function getSaled (){  
...  
}
```

Για να τα βρούμε χρειάζεται να βρούμε την τωρινή ώρα σαν `string` που καταλαβαίνει η MySQL.

```
$now = new DateTime();  
$time=$now->format('Y-n-j G:i:s');
```

Αυτό είναι απαραίτητο για να βρούμε τα προϊόντα που οι δημοπρασίες τους έχουν λήξει και δεν έχουν κατοχυρωθεί σε κάποιον πελάτη.


```
SELECT *  
FROM proion AS prod INNER JOIN dimoprasia AS dimo ON prod.id  
= dimo.proion  
WHERE (dimo.pelatis IS NULL) AND (ora_liksis < '$time')
```

Έπειτα για κάθε ένα από αυτά αποθηκεύουμε τα στοιχεία του σε έναν πίνακα.

```
$keys = array_keys($row);  
for($x = 0; $x < count($keys); $x++)  
{  
    $this->results[$i][$keys[$x]] = $row[$keys[$x]];  
}
```

Επιλέγουμε για το τρέχον προϊόν την μέγιστη προσφορά που έχει γίνει.

```
SELECT MAX(timi) AS maxtimi  
FROM prosfora  
WHERE dimoprasia = ".$this->results[$i]['id']
```

Έπειτα αποθηκεύουμε και αυτή στη λίστα με τα αποτελέσματα.

```
SELECT MAX(timi) AS maxtimi  
FROM prosfora  
WHERE dimoprasia = ".$this->results[$i]['id']
```

Οι εταιρείες επιστρέφονται από τη μέθοδο getEtairies.

```
public function getEtairies ()  
...  
}
```

Αυτή επιλέγει όλες τις εταιρίες που περιέχονται στα προϊόντα, ζητώντας μόνο μία φορά το κάθε αποτέλεσμα με την λέξη DISTINCT.

```
SELECT DISTINCT eteria FROM proion
```

Για να πάρουμε ένα προϊόν με έναν συγκεκριμένο κωδικό χρησιμοποιούμε την μέθοδο `getProion`, που παίρνει για παράμετρο τον κωδικό του.

```
public function getProion (){  
...  
}
```

Αυτή διαλέγει το συγκεκριμένο προϊόν.

```
SELECT * FROM proion WHERE id='$kodiko'
```

Έπειτα τα στοιχεία του αποθηκεύονται στον πίνακα `results`.

Για να πάρουμε όλα τα προϊόντα ενός συγκεκριμένου πελάτη χρησιμοποιούμε την μέθοδο `getProionta` που παίρνει για παράμετρο τον κωδικό του πελάτη.

```
public function getProionta (){  
...  
}
```

Αυτή έχει διαλέγει τα προϊόντα που ανήκουν σε μία δημοπρασία στην οποία έχει κερδίσει ο πελάτης με το συγκεκριμένο `id`.

```
SELECT p.onoma, d.id, d.imerominia_katakiosis  
FROM dimoprasia AS d INNER JOIN proion AS p ON d.proion=p.id  
WHERE d.pelatis= '$kodiko'
```

Τα στοιχεία του κάθε προϊόντος αποθηκεύονται στον πίνακα με τα αποτελέσματα, και έπειτα επιλέγεται η μέγιστη προσφορά.

```
SELECT MAX(timi) AS poso FROM profora  
WHERE dimoprasia = $row[id]
```

Μετά και αυτή με τη σειρά της αποθηκεύεται στο αποτέλεσμα.

```
$rs1 = mysql_query($query);  
$row1 = mysql_fetch_assoc($rs1);  
$this->results[$i]['timi'] = $row1['poso'];
```

Για να πάρουμε όλα τα προϊόντα χρησιμοποιούμε την μέθοδο getAll.

```
public function getAll (){  
...  
}
```

Αυτή επιλέγει όλες τις εγγραφές του πίνακα και βάζει τα αποτελέσματα στον πίνακα results.

```
SELECT * FROM proion
```

Για την εισαγωγή ενός προϊόντος χρησιμοποιείται η μέθοδος insertProion, η οποία παίρνει για παράμετρο έναν πίνακα με τα στοιχεία του νέου προϊόντος. Πέρα από τα στοιχεία που αποθηκεύονται στη βάση παίρνει και ένα επιπλέον στοιχείο του πίνακα που είναι το όνομα του προσωρινού αρχείου που έχει ανεβάσει ο χρήστης.

Αρχικά μετακινούμε αυτό το αρχείο στον κατάλληλο φάκελο.

```
move_uploaded_file( $data['tempname'], $data['newpath']);
```

Έπειτα εισάγονται τα στοιχεία στη βάση με την κατάλληλη εντολή insert.

```
INSERT INTO proion (timi,onoma,eteria,fotografia,kathgoria)  
VALUES ('$data[timi]', '$data[onoma]', '$data[eteria]',  
 '$data[filename]', '$data[kathgoria]')
```

Για την ανανέωση των στοιχείων ενός προϊόντος καλούμε την μέθοδο updateProion, η οποία παίρνει για παράμετρο τον κωδικό του προϊόντος που θα τροποποιηθεί και τα νέα στοιχεία.

```
public function updateProion($kodiko,$data){  
...  
}
```

Εδώ η διαχείριση της εικόνας είναι πιο πολύπλοκη. Αρχικά πρέπει να ελέγξουμε αν έχει ανέβει μία νέα εικόνα. Αν ναι, τότε είναι απαραίτητο να την διαγράψουμε και να βάλουμε στη θέση της την νέα.

```
if (isset($data['filename'])) {  
    if (strlen($data['old_image_name']) > 0) {  
        $path = "./images/products/" . $data['old_image_name'];  
        unlink($path);  
    }  
    move_uploaded_file( $data['tempname'], $data['newpath']);  
}
```

Μετά ανανεώνουμε τα στοιχεία του προϊόντος.

```
UPDATE proion  
SET timi= '$data[timi]', onoma = '$data[onoma]', eteria =  
'$data[eteria]', fotografia = '$data[filename]',  
kathgoria='$data[kathgoria]'  
WHERE id= '$kodiko'
```

Η τελευταία συνάρτηση που σχετίζεται με τις λειτουργίες των προϊόντων είναι η deleteProion, η οποία παίρνει για παράμετρο τον κωδικό ενός προϊόντος και το διαγράφει.

```
public function deleteProion ($kodiko){  
...  
}
```

Η διαγραφή δεν γίνεται ανεξέλεγκτα. Αν το προϊόν χρησιμοποιείται ήδη σε μία δημοπρασία δεν μπορεί να διαγραφεί, οπότε πρώτα ελέγχεται αυτή η περίπτωση.

```
$query="SELECT * FROM dimoprasia WHERE proion=$kodiko";  
$rs1=mysql_query($query);  
if( mysql_num_rows($rs1)>0){  
    $this->errors[]= "Το προϊόν χρησιμοποιείται ήδη σε μία  
dimoprasia";  
    return false;  
}
```

Αν δεν ισχύει κάτι τέτοιο τότε διαγράφεται με την αντίστοιχη εντολή SQL.

```
DELETE FROM proion WHERE id=$kodiko
```

4.4.6.4 Κλάση *prosfora.php*

Το αρχείο *prosfora.php* περιέχει τις ενέργειες που σχετίζονται με την προσφορά που κάνει ένας χρήστης σε μία δημοπρασία. Όπως και οι προηγούμενες κλάσεις περιέχει τις μεταβλητές *errors* και *results*, και τους αντίστοιχους *getters*, ενώ κάνει *include* το αρχείο με τις λειτουργίες σύνδεσης στη βάση.

Η μέθοδος *recent_offers* παίρνει για παράμετρο το *id* ενός χρήστη και βρίσκει τις προσφορές που έχει κάνει.

```
public function recent_offers($uid){  
...  
}
```

Για να πάρουμε όλα τα απαραίτητα στοιχεία πρέπει να συνδυάσουμε τα στοιχεία της δημοπρασίας με την προσφορά.

```
SELECT p.ora, p.timi, d.id, d.arxiki_timi  
FROM dimoprasia AS d INNER JOIN prosfora AS p ON  
d.id=p.dimoprasia  
WHERE p.pelatis= '$uid'
```

Για την εισαγωγή μίας νέας προσφοράς χρησιμοποιούμε την μέθοδο *insertProsfora*, η οποία παίρνει για παραμέτρους το *idd* που είναι το *id* της δημοπρασίας, το *idp* που είναι το *id* της προσφοράς και το ποσό της προσφοράς.

```
public function insertProsfora($idd,$idp,$timi){  
...  
}
```

Πριν γίνει η καταχώρηση ελέγχουμε ότι η δημοπρασία δεν έληξε πριν ο χρήστης υποβάλλει την προσφορά του.

```
SELECT ora_liksis FROM dimoprasia WHERE id = $idd
```

Από αυτό παίρνουμε την ώρα λήξης σαν ημερομηνία και την μετατρέπουμε σε αριθμό. Έπειτα παίρνουμε την παρούσα ώρα και την μετατρέπουμε και αυτή σε αριθμό και κάνουμε αριθμητική σύγκριση των δύο τιμών.

```
$ora_liksis = strtotime($row['ora_liksis']);  
$now = new DateTime();  
$nowtime= strtotime($now->format('Y-n-j G:i:s'));  
if ($nowtime >= $ora_liksis) {  
    $this->errors[] = "Η δημοπρασία έχει λήξει! Η προσφορά  
σας δεν θα αποθηκευτεί!";  
    return false;  
}
```

Έπειτα ελέγχουμε ότι η προσφορά που υποβάλει ο χρήστης δεν είναι μικρότερη από προηγούμενη προσφορά που έχει κάνει. Γι' αυτό βρίσκουμε την μεγαλύτερη προσφορά που έχει ήδη κάνει και, αν υπάρχει τέτοια προσφορά, την συγκρίνουμε με την τωρινή.

```
$query = "SELECT MAX(timi) AS poso FROM prosfora WHERE  
pelatis = $idp AND dimoprasia = $idd";  
  
$rs = mysql_query($query);  
$row = mysql_fetch_assoc($rs);  
  
if ($row != null)  
{  
    $poso = floatval($row['poso']);  
    if ($timi<=$poso) {  
        $this->errors[] = "Η προσφορά αυτή είναι χαμηλότερη  
από προηγούμενες προσφορές που έχετε κάνει στην δημοπρασία  
αυτή και άρα δεν πρόκειται να αποθηκευτεί!!! ";  
        return false;  
    }  
}
```

Αν όλοι οι έλεγχοι ολοκληρωθούν χωρίς πρόβλημα τότε υπολογίζουμε την τρέχουσα ώρα στη μορφή που απαιτείται από την MySQL και την εισάγουμε.

```
$now = new DateTime();  
$time=$now->format('Y-n-j G:i:s');  
  
$query = "INSERT INTO prosfora (dimoprasia,pelatis,timi,ora)  
VALUES ($idd,$idp,$timi,'$time)";
```

4.4.6.5 Κλάση *dimoprasia.php*

Το αρχείο *dimoprasia.php* περιέχει τον ορισμό της κλάσης *Dimoprasia* η οποία χρησιμοποιεί το αρχείο *mysql_lib.php* για τη σύνδεση με τη βάση και επιπλέον χρειάζεται τις εσωτερικές μεταβλητές *results* και *errors*.

Η συνάρτηση *search* χρησιμοποιεί τις μεταβλητές που έχουν υποβληθεί από τη φόρμα αναζήτησης για να κατασκευάσει μία εντολή SQL με τα κατάλληλα κριτήρια.

```
public function search(){  
...  
}
```

Η βασική εντολή είναι η παρακάτω, που επιλέγει όλα τα κριτήρια για την επιλογή των κατάλληλων πεδίων και επιπλέον βρίσκει όλες τις δημοπρασίες που δεν έχουν λήξει.

```
$query = "SELECT d.id AS auction_id, p.id AS product_id,  
ora_liksis, ora_enarksis, proion, arxiki_timi, onoma, eteria,  
fotografia, kathgoria FROM dimoprasia AS d JOIN proion AS p  
ON d.proion = p.id WHERE ora_liksis > '$time';"
```

Έπειτα αν ο χρήστης έχει επιλέξει προϊόν τότε η μεταβλητή *select_proion* θα έχει τιμή που δεν θα είναι -1, οπότε επικολλάται το αντίστοιχο κριτήριο στην επερώτηση.

```
if($_POST['select_proion'] != -1) {  
    $query = $query." AND proion =  
    '" . $_POST['select_proion'] . "' ";  
}
```

Επιπλέον αν το *select_etairia* δεν έχει τιμή -1 σημαίνει πως έχει διαλέξει και εταιρεία, και ενώνεται και αυτό το κριτήριο στην επερώτηση.

```
if($_POST['select_etairia'] != -1) {  
    $query = $query." AND eteria =  
    '" . $_POST['select_etairia'] . "' ";  
}
```

Με την ίδια λογική ελέγχεται και η τιμή του *select_kathgoria* για το αν είναι -1 η κατηγορία που έχει επιλέξει ή όχι.

```
if($_POST['select_kategoria'] != -1) {  
    $query = $query." AND kathgoria =  
    '". $_POST['select_kategoria']."'";  
}
```

Τέλος ταξινομούμε τα αποτελέσματα ανάλογα με την ώρα έναρξης.

```
$query = $query." ORDER BY ora_enarksis ASC";
```

Αφού πάρουμε τα αποτελέσματα τα βάζουμε στον πίνακα result.

Για την εύρεση του πλήθους του αριθμού των δημοπρασιών που έχουν λήξει έχει δημιουργηθεί η μέθοδος countLigmenes.

```
public function countLigmenes(){  
...  
}
```

Αυτή βρίσκει την τωρινή ημερομηνία και ώρα στην κατάλληλη μορφή και έπειτα επιλέγει το πλήθος των δημοπρασιών που έχουν μικρότερη ώρα λήξης από αυτή που υπολογίστηκε.

```
SELECT COUNT(*) AS total  
FROM dimoprasia  
WHERE ora_liksis<'$time'
```

Για την εύρεση του πλήθους των δημοπρασιών που είναι άγονες, δηλαδή που έχουν λήξει χωρίς να κατατεθεί προσφορά υπάρχει η συνάρτηση countAgones.

```
public function countAgones (){  
...  
}
```

Για την εύρεσή τους επιλέγονται όλες αυτές που έχουν στην ιδιότητα αποστολή την τιμή -1.


```
SELECT * FROM dimoprasia WHERE apostoli = -1
```

Τα αποτελέσματα έπειτα καταμετρούνται και αποθηκεύεται στο result το πλήθος τους.

Για την εύρεση της μέγιστης προσφοράς που έχει γίνει σε μία δημοπρασία υπάρχει η μέθοδος getMaxPrice, η οποία παίρνει σαν παράμετρο το id της εν λόγω δημοπρασίας.

```
public function getMaxPrice($id){  
...  
}
```

Αυτή αρχικά βρίσκει την αρχική τιμή της δημοπρασίας.

```
$query = "SELECT * FROM dimoprasia WHERE id = $id";  
$row = mysql_fetch_assoc($rs);  
$arxiki_timi = $row['arxiki_timi'];
```

Μετά επιλέγει την μέγιστη τιμή από όλες τις προσφορές για αυτή τη δημοπρασία.

```
$query = "SELECT MAX(timi) AS maxtimi FROM prosfora WHERE  
dimoprasia = $id";  
$row = mysql_fetch_assoc($rs);  
$max_prosfora = $row['maxtimi'];
```

Αν αυτή είναι μεγαλύτερη από την αρχική τιμή τότε την βάζει στο result, αλλιώς χρησιμοποιείται η παραπάνω αρχική τιμή.

```
if ($max_prosfora > $arxiki_timi)  
    $this->results = $max_prosfora;  
else  
    $this->results = $arxiki_timi;
```

Για τον υπολογισμό του μέσου όρου διάρκειας μίας δημοπρασίας υπάρχει η συνάρτηση getDurations που επιστρέφει έναν πίνακα με τις διάρκειες (σε ώρες) όλων των ολοκληρωμένων δημοπρασιών.

```
public function getDurations(){  
...  
}
```

Αρχικά υπολογίζεται η τωρινή ημερομηνία στην κατάλληλη μορφή για χρήση με την βάση δεδομένων και επιλέγονται όλες οι δημοπρασίες που έχουν λήξει.

```
$now = new DateTime();  
$time=$now->format('Y-n-j G:i:s');  
  
$query = "SELECT ora_liksis, ora_enarksis FROM dimoprasia  
WHERE ora_liksis<'$time'";
```

Μετά διατρέχουμε τα αποτελέσματα. Για κάθε ένα από αυτά υπολογίζεται η ώρα έναρξης και η ώρα λήξης σε δευτερόλεπτα. Η διαφορά τους διαιρείται με το 3600 για να υπολογιστεί η διάρκεια σε ώρες, η οποία τοποθετείται στο αποτέλεσμα.

```
$t1 = strtotime($dates[$j]['ora_enarksis']);  
$t2 = strtotime($dates[$j]['ora_liksis']);  
$this->results[] = (($t2-$t1)/3600);
```

Οι δημοπρασίες που δεν έχουν ολοκληρωθεί επιστρέφονται από τη μέθοδο getFutureOnes.

```
public function getFutureOnes (){  
...  
}
```

Αυτή πάλι υπολογίζει την ώρα στην κατάλληλη μορφή αλλά ψάχνει να βρει τις δημοπρασίες που η ώρα λήξης τους είναι μεγαλύτερη από την τρέχουσα.

```
$now = new DateTime();  
$time=$now->format('Y-n-j G:i:s');  
  
$query = "SELECT d.id AS auction_id, p.id AS product_id,  
ora_liksis, ora_enarksis, proion, arxiki_timi, onoma, eteria,  
fotografia, kathgoria FROM dimoprasia AS d INNER JOIN proion  
AS p ON d.proion = p.id WHERE ora_liksis>'$time' ORDER BY  
ora_enarksis ASC";
```

Οι κατακυρωμένες δημοπρασίες επιστρέφονται από τη μέθοδο getKatakiromenes.

```
public function getKatakiromenes (){  
...  
}
```

Για να είναι μία δημοπρασία κατακυρωμένη πρέπει να έχει λήξει και να έχει αποσταλεί. Άρα βρίσκουμε την ώρα ώστε να ελέγξουμε πως η ώρα λήξης είναι μεγαλύτερη από την τωρινή και πως έχουν αποσταλεί.

```
$now = new DateTime();  
$time=$now->format('Y-n-j G:i:s');  
  
$query = "SELECT d.id AS auction_id, d.proion, p.id, p.onoma,  
d.pelatis, d.apostoli, d.imerominia_katakirosis FROM  
dimoprasia AS d INNER JOIN proion AS p ON d.proion = p.id  
WHERE apostoli >= 0 ORDER BY imerominia_katakirosis DESC";
```

Τα αποτελέσματα ταξινομούνται από το παλιότερο στο πιο πρόσφατο.

Για να πάρουμε τα στοιχεία μίας συγκεκριμένης δημοπρασίας έχουμε γράψει την μέθοδο getDimoprasia που παίρνει για παράμετρο τον κωδικό της ζητούμενης δημοπρασίας.

```
public function getDimoprasia($kodiko){  
...  
}
```

Επιλέγεται από τη βάση η ζητούμενη και τα στοιχεία της μπαίνουν στον πίνακα result.

Για τις περιπτώσεις που πέρα από τα στοιχεία της δημοπρασίας χρειάζονται και τα στοιχεία του προϊόντος της υπάρχει η getDimoprasiaWithProion, η οποία επίσης παίρνει τον κωδικό του προϊόντος.

```
public function getDimoprasiaWithProion ($kodiko){  
...  
}
```

Η διαφορά της σε σχέση με την προηγούμενη είναι πως διαλέγει όλα τα περιεχόμενα του συσχετισμένου προϊόντος.

```
SELECT *  
FROM dimoprasia JOIN proion ON dimoprasia.proion = proion.id  
WHERE dimoprasia.id = '$kodiko'
```

Για την εισαγωγή μίας δημοπρασίας υπάρχει η συνάρτηση insertDimoprasia που παίρνει σαν παράμετρο έναν πίνακα με τα στοιχεία της νέας δημοπρασίας.

```
public function insertDimoprasia($data){  
...  
}
```

Πριν την εισαγωγή πρέπει να ελέγξουμε αν υπάρχει επικάλυψη με μία από τις υπάρχουσες δημοπρασίες.

```
$query = "SELECT * FROM dimoprasia WHERE  
((ora_enarksis<'$data[ora_enarksis]') AND  
(ora_liksis>'$data[ora_enarksis]')) OR  
((ora_enarksis<'$data[ora_liksis]') AND  
(ora_liksis>'$data[ora_liksis]')) ";  
$rs1=mysql_query($query);  
if(mysql_num_rows($rs1)>0){  
    $this->errors[] = "Οι ημερομηνίες έναρξης και λήξης  
δημιουργούν αλληλοεπικάλυψη με άλλη δημοπρασία";  
    return false;  
}
```

Αν όχι τότε χρειάζεται να βρούμε το προϊόν για το οποίο γίνεται η δημοπρασία για να βρούμε την τιμή του.

```
$query = "SELECT timi FROM proion WHERE id = $data[proion]";  
  
$row = mysql_fetch_assoc($rs2);  
$arxiki_timi = $row['timi'];
```

Έπειτα εισάγουμε τη νέα δημοπρασία.

```
INSERT INTO dimoprasia  
(ora_enarksis,ora_liksis,proion,arxiki_timi) VALUES  
('$data[ora_enarksis]','$data[ora_liksis]','$data[proion]','$ar  
xiki_timi)
```

Για την ανανέωση των στοιχείων της δημοπρασίας υπάρχει η μέθοδος updateDimoprasia, που παίρνει για παράμετρο τα στοιχεία της δημοπρασίας και τον κωδικό αυτής που θα τροποποιηθεί.

```
public function updateDimoprasia($kodiko,$data){  
...  
}
```

Η ανανέωση γίνεται με την παρακάτω εντολή.

```
UPDATE dimoprasia SET ora_enarksis=  
'$data[ora_enarksis]',proion = '$data[proion]' WHERE id =  
'$kodiko'
```

Για την αλλαγή της κατάστασης αποστολής μίας δημοπρασίας υπάρχει η συνάρτηση sendProduct, που παίρνει για παράμετρο το id μίας δημοπρασίας. Αυτή αλλάζει την τιμή του πεδίου apostoli μιας κατακυρωμένης δημοπρασίας από 0 αν δεν έχει αποσταλεί σε 1, πως έχει αποσταλεί.

```
public function sendProduct($auction_id){  
...  
}
```

Η ανανέωση των στοιχείων γίνεται με την παρακάτω εντολή.

```
UPDATE dimoprasia SET apostoli = 1 WHERE id = $auction_id
```

Τέλος για την διαγραφή μίας δημοπρασίας υπάρχει η μέθοδος deleteDimoprasia που παίρνει για παράμετρο το id μίας δημοπρασίας και την διαγράφει.

```
public function deleteDimoprasia($id){  
...  
}
```

Η διαγραφή γίνεται με την εντολή:

```
DELETE FROM dimoprasia WHERE id=$id
```

4.4.7 Φάκελος scripts

4.4.7.1 Δημιουργία XML feed

Η δημιουργία του XML feed γίνεται από το αρχείο updateFeeds.php, που βρίσκεται στον φάκελο scripts.

Αυτό αρχικά κάνει include την κλάση Dimoprasia και δημιουργεί ένα τέτοιο αντικείμενο.

```
include_once('../classes/dimoprasia.php');  
$dimoprasia = new Dimoprasia();
```

Έπειτα παίρνει όλες τις μελλοντικές δημοπρασίες και τις αποθηκεύει σε έναν πίνακα.

```
if ($dimoprasia->getFutureOnes()) {  
    $mellontikes = $dimoprasia->getResults();  
}
```

Στην αρχή δημιουργείται η αρχή του feed βάζοντας τα κατάλληλα αποσπάσματα κώδικα XML μέσα σε ένα string.

```
$feed = "<?xml version=\"1.0\"?>";  
$feed = $feed."<rss version=\"2.0\"><channel>";  
$feed = $feed."<title>Κανάλι  
δημοπρασιών</title><link>http://localhost/pop/login.php</link>  
<description>Δημοπρασίες που θα εκτελεστούν στο προσεχές  
μέλλον!</description>";
```

Έπειτα διατρέχουμε τη λίστα με τις δημοπρασίες και για κάθε μία δημιουργούμε το tag item με όλα τα περιεχόμενά του.

```
foreach ($mellontikes as $auction) {
    $feed = $feed."<item><title>Δημοπρασία με κωδικό:
    ".$auction['auction_id']."</title><link>http://localhost/pop/1
    ogin.php</link><description><![CDATA[Δημοπρασία ενός
    <strong>".$auction['onoma']."</strong> με αρχική τιμή τα
    ".$auction['arxiki_timi']." ευρώ.<br/> Έναρξη δημοπρασίας στις
    <strong>".$auction['ora_enarksis']."</strong>.]></description
    >";
    if (strlen($auction['fotografia']) > 0 )
        $feed = $feed."<enclosure
    url=\"http://localhost/pop/images/products/\".$auction['fotogra
    fia'].\"\" type=\"image/png\" />";
    $feed = $feed."</item>";
}
```

Τέλος κλείνουμε τα tags της XML.

```
$feed = $feed."</channel></rss>";
```

Το αποτέλεσμα σώζεται στο αρχείο feed.xml.

```
$file=fopen('../feed.xml','w');
fwrite($file,$feed);
fclose($file);
```

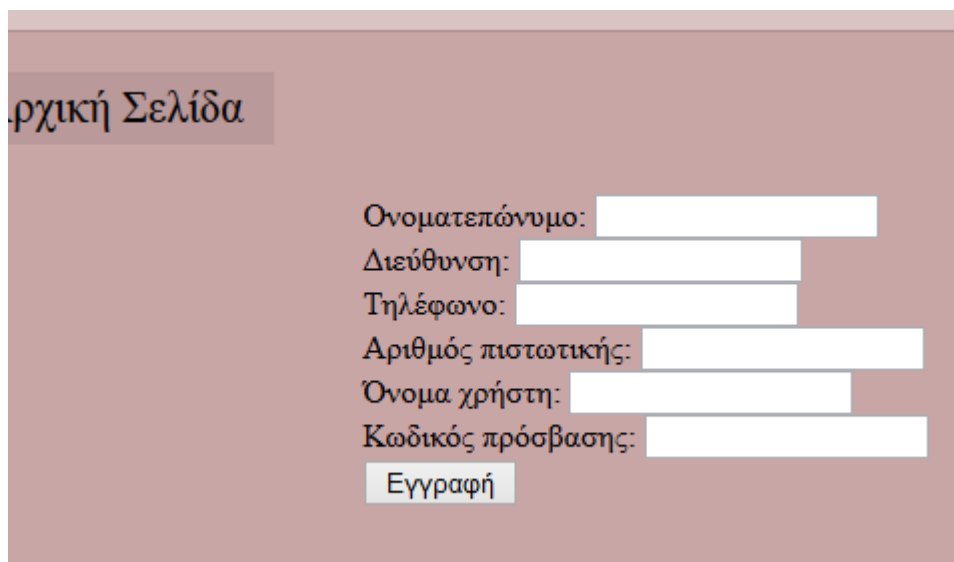
5 Εγχειρίδιο χρήσης

5.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα δώσουμε αναλυτικές οδηγίες για την λειτουργία του συστήματος.

5.2 Εγγραφή χρήστη

Για να εγγραφεί ένας νέος χρήστης πρέπει να συμπληρώσει τα στοιχεία του στην αρχική σελίδα.



Αρχική Σελίδα

Όνοματεπώνυμο:

Διεύθυνση:

Τηλέφωνο:

Αριθμός πιστωτικής:

Όνομα χρήστη:

Κωδικός πρόσβασης:

5.3 Είσοδος χρήστη

Για να μπει ένας χρήστης στο σύστημα χρειάζεται να βάλει το όνομα χρήστη και τον κωδικό που έχει επιλέξει κατά την εγγραφή του στο κατάλληλο πεδίο της αρχικής σελίδας.



 Σύστημα

Username:

Password:

Αν δώσει λάθος στοιχεία θα εμφανιστεί ένα μήνυμα λάθους.

Το όνομα ή ο κωδικός χρήστης που δώσατε είναι λάθος!



Σύστημα Η

Username:

Password:

Σε αντίθετη περίπτωση θα μπει στο σύστημα και παραπέμπεται στην κεντρική σελίδα.



Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών

[Το προφίλ μου](#) [Αποσύνδεση](#)

Εταιρία	Προϊόν	Κατηγορία	
Καμία επιλογή ▾	Καμία επιλογή ▾	Καμία επιλογή ▾	<input type="button" value="Αναζήτηση"/>

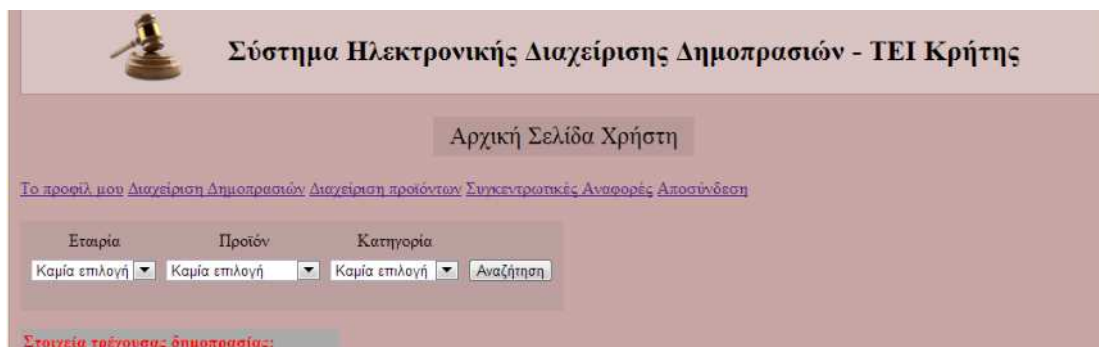
Στοιχεία τρέχουσας δημοπρασίας:

Κωδικός δημοπρασίας:	8
Όνομα προϊόντος:	Olympus VR-325
Εταιρία:	Olympus
Κατηγορία:	digital cameras
Ωρα έναρξης:	2013-05-07 16:43:28
Ωρα λήξης:	2013-07-07 16:43:23
Τρέχουσα τιμή:	215
Υποβολή προσφοράς	<input type="text"/> €

Στοιχεία μελλοντικής δημοπρασίας:

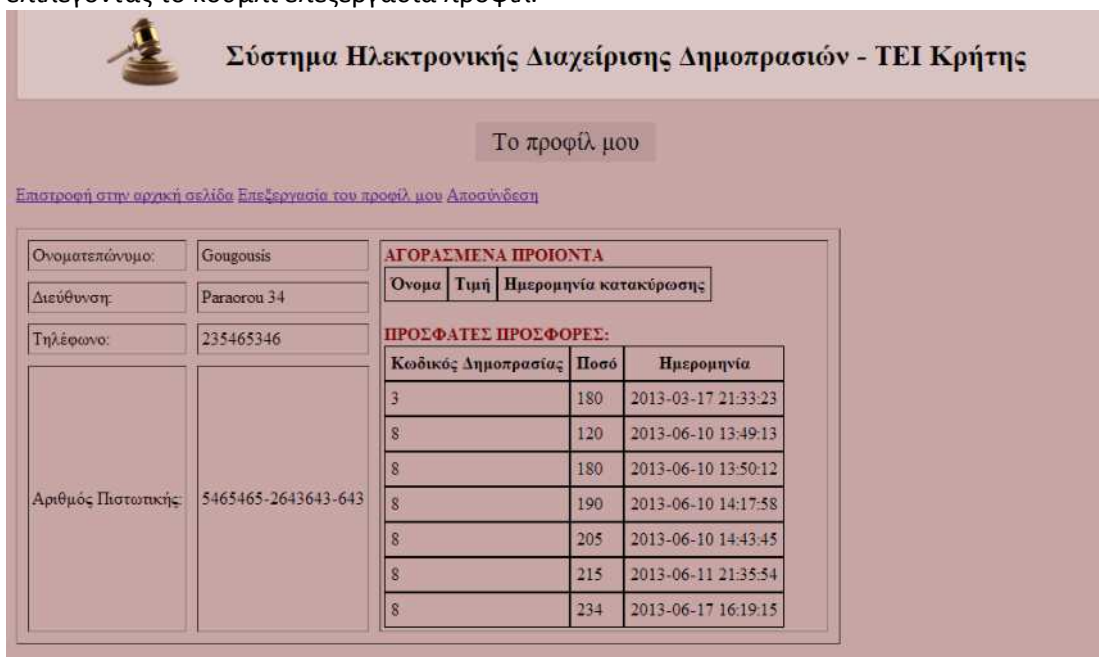
Κωδικός δημοπρασίας:	6
----------------------	---

Αν ο χρήστη είναι διαχειριστής θα έχει περισσότερες επιλογές, δηλαδή θα βλέπει διαφορετική μπάρα για αυτές.



5.4 Προφίλ χρήστη

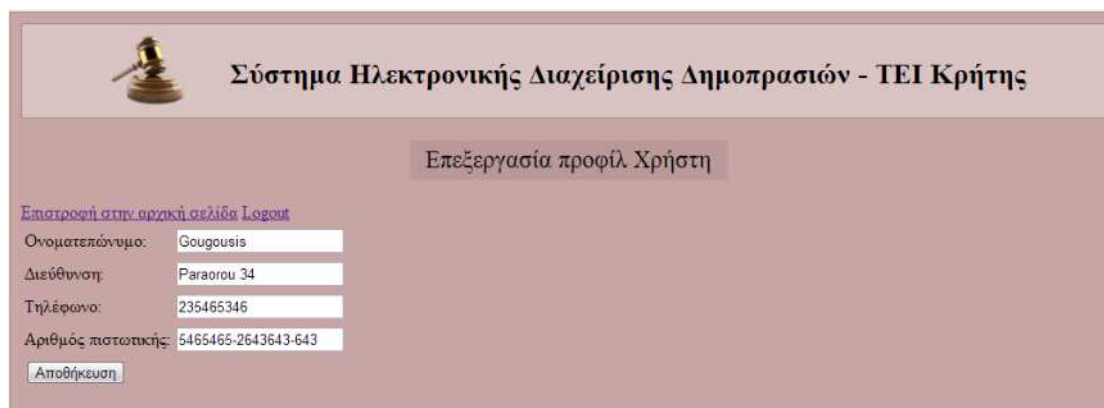
Στην σελίδα αυτή ο διαχειριστής μπορεί να δει τα στοιχεία τα αγορασμένα προϊόντα του και τις πρόσφατες προσφορές του ακόμη μπορεί να επεξεργαστεί το προφίλ το επιλέγοντας το κουμπί επεξεργασία προφίλ.




Όνομα	Τιμή	Ημερομηνία κατακύρωσης
3	180	2013-03-17 21:33:23
8	120	2013-06-10 13:49:13
8	180	2013-06-10 13:50:12
8	190	2013-06-10 14:17:58
8	205	2013-06-10 14:43:45
8	215	2013-06-11 21:35:54
8	234	2013-06-17 16:19:15

5.5 Επεξεργασία του προφίλ

Στην σελίδα αυτή ο διαχειριστής μπορεί να τροποποιήσει το προφίλ του. Πατώντας αποθήκευση οι αλλαγές αποθηκεύονται στην βάση και τον επιστρέφει στο προφίλ του με τα νέα στοιχεία.



 **Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών - ΤΕΙ Κρήτης**

Επεξεργασία προφίλ Χρήστη

[Επιστροφή στην αρχική σελίδα Logout](#)

Όνοματεπώνυμο:

Διεύθυνση:

Τηλέφωνο:

Αριθμός πιστωτικής:

5.6 Υποβολή προσφοράς

Όταν ένας χρήστης μπει στο σύστημα εμφανίζεται μία λίστα με τις τρέχουσες δημοπρασίες.

Στοιχεία τρέχουσας δημοπρασίας:	
Κωδικός δημοπρασίας:	8
Όνομα προϊόντος:	Olympus VR-325
Εταιρία:	Olympus
Κατηγορία:	digital cameras
Ωρα έναρξης:	2013-05-07 16:43:28
Ωρα λήξης:	2013-07-07 16:43:23
Τρέχουσα τιμή:	215
Υποβολή προσφοράς	<input type="text"/> €

Στοιχεία μελλοντικής δημοπρασίας:	
Κωδικός δημοπρασίας:	6
Όνομα προϊόντος:	HTC One V
Εταιρία:	HTC
Κατηγορία:	smartphones
Ωρα έναρξης:	2013-07-12 16:42:41
Ωρα λήξης:	2013-09-12 16:42:31
Αρχική τιμή:	201
Υποβολή προσφοράς	<input type="text"/> €

Στοιχεία μελλοντικής δημοπρασίας:	
Κωδικός δημοπρασίας:	9
Όνομα προϊόντος:	HTC One V
Εταιρία:	HTC
Κατηγορία:	smartphones
Ωρα έναρξης:	2013-07-12 16:42:41
Ωρα λήξης:	2013-09-12 16:42:31
Αρχική τιμή:	201
Υποβολή προσφοράς	<input type="text"/> €


Εκεί μπορεί να συμπληρώσει το ποσό που θέλει να υποβάλει σαν προσφορά και να πατήσει το κουμπί Υποβολή προσφοράς. Εάν ο χρήστης κάνει προσφορά με μικρότερη τιμή από αυτή που υπάρχει το σύστημα του εμφανίζει:



5.7 Διαχείριση δημοπρασιών

Στην σελίδα αυτή ο διαχειριστής έχει την δυνατότητα να επιλέξει εάν θέλει να κάνει προσθήκη νέας δημοπρασίας ή διαγραφή κάποιας άλλης .



Τρέχουσες και Μελλοντικές Δημοπρασίες	
Κωδικός Δημοπρασίας:8 Προϊόν:  Διαγραφή	Κωδικός Δημοπρασίας:6 Προϊόν:  Διαγραφή
Κωδικός Δημοπρασίας:9 Προϊόν:  Διαγραφή	Κωδικός Δημοπρασίας:7 Προϊόν:  Διαγραφή

Κατακυρωμένες Δημοπρασίες	
Κωδικός Δημοπρασίας:2 Προϊόν: Notice: Undefined index: fotografia in C:\xampp\htdocs\pop\adminDimoprasies.php on line 116  Ημερομηνία Κατακύρωσης: 2013-03-19 23:13:45 Έχει αποσταλεί: Ναι	Κωδικός Δημοπρασίας:1 Προϊόν: Notice: Undefined index: fotografia in C:\xampp\htdocs\pop\adminDimoprasies.php on line 116  Ημερομηνία Κατακύρωσης: 2013-03-17 20:09:24 Έχει αποσταλεί: Ναι

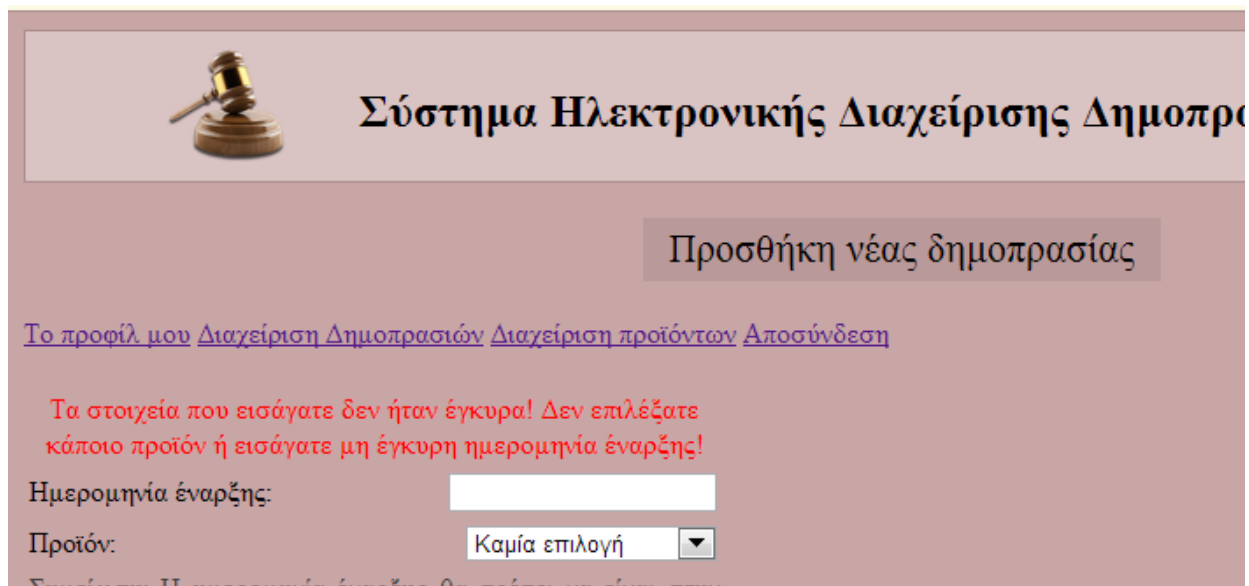
Προσθήκη νέας δημοπρασίας

5.8 Προσθήκη νέας δημοπρασίας

Στην σελίδα αυτή ο διαχειριστής έχει την δυνατότητα να κάνει προσθήκη μιας νέας δημοπρασίας επιλέγοντας ημερομηνία έναρξης και το προϊόν. Πατώντας αποθήκευση η νέα δημοπρασία αποθηκεύετε στην βάση και ξεκινάει να είναι ενεργή.



Στην σελίδα αυτή ο διαχειριστής έχει εισάγει κάποιο πεδίο της φόρμας λάθος και του εμφανίζει σχετικό μήνυμα.



5.9 Διαχείριση προϊόντων

Στην σελίδα αυτή ο διαχειριστής μπορεί να επιλέξει να διαγράψει κάποιο προϊόν το οποίο δεν χρησιμοποιείτε σε κάποια δημοπρασία ή να το τροποποιήσει.



Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών - ΤΕΙ Κρήτης


Διαχείριση Προϊόντων

[Αρχική Σελίδα Αποσύνδεση](#)

Εταιρία	Μοντέλο	Κατηγορία	Φωτογραφία		
Olympus VR-325	Olympus	digital cameras		Τροποποίηση	Διαγραφή
Canon Ixus 240 HS	Canon	digital cameras		Τροποποίηση	Διαγραφή
Samsung WB800	Samsung	digital cameras		Τροποποίηση	Διαγραφή
Sony Xperia J	Sony-Ericsson	smartphones		Τροποποίηση	Διαγραφή
HTC One V	HTC	smartphones		Τροποποίηση	Διαγραφή
HTC Desire X	HTC	smartphones		Τροποποίηση	Διαγραφή
Dell Inspiron 3521	Dell	laptops		Τροποποίηση	Διαγραφή

5.10 Τροποποίηση προϊόντος

Στην σελίδα αυτή ο διαχειριστής μπορεί να επεξεργαστεί ένα υπάρχων προϊόν τροποποιώντας του τα στοιχεία που επιθυμεί. Πατώντας αποθήκευση οι αλλαγές αποθηκεύονται στην βάση.



Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών - ΤΕΙ Κρήτης

Τροποποίηση προϊόντος


[Το προφίλ μου](#) [Διαχείριση Προϊόντων](#) [Αποσύνδεση](#)

Όνομα:

Εταιρία:

Κατηγορία:


Αρχική Τιμή:

Φωτογραφία: 

Νέα Φωτογραφία: Δεν έχει επιλεγεί κανένα αρχείο

5.11 Προσθήκη νέου προϊόντος

Στην σελίδα αυτή ο διαχειριστής έχει την δυνατότητα συμπληρώνοντας την φόρμα να εισάγει ένα νέο προϊόν και να φορτώσει και φωτογραφία. Πατώντας αποθήκευση το προϊόν αποθηκεύεται στην βάση.



Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών - ΤΕΙ Κρήτης

Προσθήκη νέου προϊόντος

[Το προφίλ μου](#) [Διαχείριση Προϊόντων](#) [Αποσύνδεση](#)

Όνομα:


Εταιρία:

Κατηγορία:

Αρχική Τιμή:

Φωτογραφία: Δεν έχει επιλεγεί κανένα αρχείο

Στην σελίδα αυτή ο διαχειριστής έχει εισάγει κάποιο στοιχείο που δεν ήταν έγκυρο ή παράλειψε κάποιο πεδίο με αποτέλεσμα να του εμφανίζει μήνυμα λάθους.



Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών - ΤΕΙ Κρήτης

Προσθήκη νέου προϊόντος

[Το προφίλ μου Διαχείριση Προϊόντων Αποσύνδεση](#)

Τα στοιχεία που εισάγατε δεν ήταν έγκυρα! Έχετε παραλείψει κάποιο υποχρεωτικό πεδίο ή έχετε εισάγει μη έγκυρες τιμές!

Όνομα:

Εταιρία:

Κατηγορία:

Αρχική Τιμή:

Φωτογραφία: Δεν έχει επιλεγεί κανένα αρχείο

5.12 Συγκεντρωτικές αναφορές

Στην σελίδα αυτή ο διαχειριστής μπορεί να δει ποιες δημοπρασίες ήταν άγονες, το σύνολο των δημοπρασιών, το ποσοστό των άγονων δημοπρασιών την μέση διάρκεια των δημοπρασιών και ποια προϊόντα έμειναν απούλητα.



Σύστημα Ηλεκτρονικής Διαχείρισης Δημοπρασιών - ΤΕΙ Κρήτης

Συγκεντρωτικές αναφορές

[Κεντρική Σελίδα Αποσύνδεση](#)

Άγονες δημοπρασίες: 7
Σύνολο δημοπρασιών: 9
Ποσοστό άγονων δημοπρασιών: 77.8%
Μέση διάρκεια δημοπρασιών: 269.17 ώρες

Απούλητα προϊόντα:

Όνομα	Φωτογραφία	Αρχική Τιμή	Μέγιστη προσφορά
HTC Desire X		200	500
Sony Xperia J		178	100
HTC Desire X		200	Χωρίς προσφορά!

6 Συμπεράσματα

Μετά την ενασχόλησή μας με την εργασία οι εντυπώσεις που μας άφησε ο συγκεκριμένος τομέας είναι ανάμεικτες. Η μεγαλύτερη απορία που μας δημιουργήθηκε δεν προήλθε από κάποιον περιορισμό της τεχνολογίας αλλά από την έλλειψη επιλογών.

Σε πρώτη φάση προσπαθήσαμε να ερευνήσουμε την προοπτική υλοποίησης της εφαρμογής με κάποιο έτοιμο σύστημα διαχείρισης περιεχομένου. Όμως, παρόλο που όλα τα γνωστά CMS έχουν τουλάχιστον από ένα πρόσθετο που επιτρέπει την δημιουργία ενός κλασικού ηλεκτρονικού καταστήματος, οι προσφερόμενες λύσεις κατά κανόνα σταματάνε εκεί.

Προφανώς ένα module για ένα κλασικό ηλεκτρονικό κατάστημα είναι εξαιρετικά πιο δημοφιλές από ένα module για ένα site δημοπρασιών, όμως αυτό δεν σημαίνει πως το δεύτερο είναι μία πρωτότυπη ή πρωτοποριακή ιδέα. Αντίθετα αυτή η λειτουργικότητα προσφέρεται από πάρα πολλές γνωστές σελίδες, οπότε η σχεδόν μηδενική προσφορά τέτοιων λύσεων δείχνει κατά τη γνώμη μας περισσότερο έλλειψη φαντασίας. Αυτό είναι ένα μεγάλο αρνητικό σημείο καθώς η πιο αποτελεσματική εκμετάλλευση του νέου τρόπου επικοινωνίας που προσφέρει το διαδίκτυο δεν μπορεί να εξαντληθεί με την μεταφορά των παραδοσιακών υπηρεσιών στο νέο μέσο αλλά απαιτεί νέες και πρωτότυπες προσεγγίσεων.

Πέρα από αυτό, οι τεχνολογίες που χρησιμοποιήσαμε στην διαδικασία ανάπτυξης μας προκάλεσαν τον θαυμασμό με το πλήθος των δυνατοτήτων που προσφέρουν. Αν και κάτι τέτοιο το περιμέναμε μέχρι ένα βαθμό λόγω της ωριμότητάς τους και της ευρείας χρήσης τους, το πλήθος των δυνατοτήτων σε συνδυασμό με την σχετική ευκολία με την οποία υλοποιούνται είναι εντυπωσιακά.

Όμως όλες αυτές οι δυνατότητες έχουν και ένα κόστος. Για την δημιουργία μίας ιστοσελίδας χρειάζεται καλή γνώση πέντε διαφορετικών γλωσσών με τελείως διαφορετική φιλοσοφία: HTML, CSS, JavaScript, PHP και SQL. Αν και κάθε μία από μόνη της είναι σχετικά απλή ο συνδυασμός τους δημιουργεί αρκετά προβλήματα σε κάποιον που δεν είναι εξοικειωμένος με αντίστοιχες εφαρμογές και τον κατάλληλο τρόπο προγραμματισμού. Αυτό συνεπάγεται πως για να μπορέσει κάποιος να δημιουργήσει μία ολοκληρωμένη εφαρμογή πρέπει να έχει αρκετά καλή θεωρητική κατάρτιση και πλήρη κατανόηση του σκοπού που εξυπηρετεί κάθε μία από αυτές.

7 Βιβλιογραφία

Database. *Wikipedia*. [Ηλεκτρονικό] <http://en.wikipedia.org/wiki/Database>.

Dynamic Website. *Marvel Infomatics*. [Ηλεκτρονικό]
http://www.marvelinfomatics.com/web-development/dynamic_website.html.

Dynamic Websites . *smooth-step.com*. [Ηλεκτρονικό] <http://www.smooth-step.com/web-design/dynamic-websites>.

HTML. *Βικιπαίδεια*. [Ηλεκτρονικό] <http://el.wikipedia.org/wiki/HTML>.

HTML5. *Βικιπαίδεια*. [Ηλεκτρονικό] <http://el.wikipedia.org/wiki/HTML5>.

MySQL. *Βικιπαίδεια*. [Ηλεκτρονικό] <http://el.wikipedia.org/wiki/MySQL>.

PHP: Hypertext Preprocessor. *PHP.net*. [Ηλεκτρονικό] <http://php.net/>.

Processing static web pages. *etutorials.org*. [Ηλεκτρονικό]
<http://etutorials.org/Macromedia/Dream+Weaver+Online+Help/Getting+Started+with+Dreamweaver/Understanding+Web+Applications/How+a+web+application+works/Processing+static+web+pages/>.

Γλώσσες Προγραμματισμού. [Ηλεκτρονικό]
https://foss.ntua.gr/wiki/index.php/%CE%93%CE%BB%CF%8E%CF%83%CF%83%CE%B5%CF%82_%CE%A0%CF%81%CE%BF%CE%B3%CF%81%CE%B1%CE%BC%CE%BC%CE%B1%CF%84%CE%B9%CF%83%CE%BC%CE%BF%CF%8D#PHP.

Εισαγωγή στις Βάσεις Δεδομένων (DataBases). *KENTPO ΠΑΗ.ΝΕ.Τ. Ν. ΦΛΩΡΙΝΑΣ*. [Ηλεκτρονικό] <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBases.html>.

Η Γλώσσα Προγραμματισμού PHP . *KENTPO ΠΑΗ.ΝΕ.Τ. Ν. ΦΛΩΡΙΝΑΣ*. [Ηλεκτρονικό] <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-Php-Analytical.html>.

Η γλώσσα προγραμματισμού php - Σύντομο ιστορικό. [Ηλεκτρονικό]
<http://grevena.pblogs.gr/2007/09/102382.html>.

Τι είναι html? [Ηλεκτρονικό] <http://pacific.jour.auth.gr/html/>.