



**ΑΝΩΤΑΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ  
ΙΔΡΥΜΑ ΚΡΗΤΗΣ**

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΠΟΛΥΜΕΣΩΝ**

**Πτυχιακή Εργασία**

*«Ανάπτυξη διαδικτυακής εφαρμογής για on-line παραγγελία  
φαγητού από κατάστημα delivery»*

**Σπουδαστής:**

- Ευναρόπουλος Χρήστος Α.Μ. 1305

**Καθηγητής:**

- Κλεισαρχάκης Μιχαήλ

# ΠΕΡΙΕΧΟΜΕΝΑ

1. Περίληψη	5
2. Διαδίκτυο και η αρχιτεκτονική 3 στρωμάτων	6
2.1 Το Διαδίκτυο (Internet)	6
2.1.1 Η ιστορία του διαδικτύου	6
2.1.2 Το διαδίκτυο σήμερα	7
2.1.3 Η κουλτούρα του διαδικτύου	7
2.1.4 Ο Παγκόσμιος Ιστός ( World Wide Web )	8
2.1.5 Διαδίκτυο 2.0 (Web 2.0)	8
2.2 Η αρχιτεκτονική πελάτη-εξυπηρετητή (client-server)	9
2.2.1 Περιγραφή των δύο στρωμάτων	9
2.2.2 Πλεονεκτήματα των τριών στρωμάτων	10
2.2.3 Αρχιτεκτονική Υλοποίησης	11
3.Τεχνολογία που χρησιμοποιήθηκε	12
3.1 Διακομιστές	12
3.1.1 Διακομιστές Web	12
3.1.2 Apache και το Κίνημα Ανοικτού Κώδικα (open source)	12
3.1.3 Διαχείριση Διακομιστή Web	14
3.1.4 Ρυθμίσεις εγκατάστασης Apache 2.2	14
3.2 HTML	15
3.2.1 Τι είναι HTML	15
3.2.2 Εκδόσεις HTML	15
3.2.3 Γλώσσες Προγραμματισμού στο Internet	16
3.3 Cascading Style Sheets – CSS	16
3.3.1 Περιγραφή	16
3.3.2. Πλεονεκτήματα - Μειονεκτήματα των CSS	17
3.3.3 Βασική δομή και κανόνες	17
3.4 PHP	20
3.4.1 Τι είναι η PHP;	20
3.4.2 Ιστορία της PHP	20
3.4.3 Τι μπορεί να κάνει η PHP;	21
3.4.4 Χαρακτηριστικά της php	22

3.4.5	Παράδειγμα χρήσης της PHP	23
3.4.6	Επικοινωνία της PHP με βάσεις δεδομένων	24
3.4.7	Ρυθμίσεις εγκαταστάσεις της PHP	26
3.5	MYSQL	27
3.5.1	Περιγραφή	27
3.5.2	Πλεονεκτήματα της MySQL	27
3.6	JAVASCRIPT	28
4.	Προγράμματα που χρησιμοποιήθηκαν	30
4.1	Dreamweaver 8	30
4.2	Photoshop CS2	32
4.3	Γραφικά εργαλεία διαχείρισης βάσης	33
4.3.1	MySQL Query Browser	33
4.3.2	Παράθυρο Διαλόγου Σύνδεσης	33
4.3.3	Περιήγηση στον MySQL Query Browser	34
4.3.4	Ρυθμίσεις	36
4.3.5	MySQL Administrator	36
4.3.6	Παράθυρο Διαλόγου Σύνδεσης	37
4.3.7	Περιήγηση στον MySQL Administrator	37
5.	Αρχιτεκτονική εφαρμογής	38
5.1	Επίπεδα χρηστών	38
5.2	Administrator	39
5.2.1	Λειτουργίες διαχειριστή	39
5.2.2	Προσθήκη κατηγορίας	40
5.2.3	Επεξεργασία κατηγορίας	42
5.2.4	Διαγραφή κατηγορίας	43
5.2.5	Προσθήκη προϊόντος	44
5.2.6	Επεξεργασία προϊόντος	46
5.2.7	Διαγραφή προϊόντων	48
5.2.8	Παραγγελίες	48
5.2.9	Ρυθμίσεις καταστήματος	51
5.2.10	Διαχείριση χρηστών	53
5.2.11	Προστασία κωδικών πρόσβασης	57
5.3	Περιγραφή της σελίδας του ηλεκτρονικού καταστήματος-Χρήστες	57
5.3.1	Δομή του ηλεκτρονικό κατάστημα	57

5.3.2	Επικοινωνία	60
5.3.3	Η εταιρία μας	62
5.3.4	Περιήγηση στα προϊόντα	62
5.3.5	Προβολή λεπτομερειών προϊόντος	65
5.3.6	Προβολή καλαθιού αγορών	67
5.3.7	Πληροφορίες παραγγελίας	69
6.	Βάση Δεδομένων	72
6.1	Αρχιτεκτονική της Βάσης	72
6.2	Μηχανές Αποθήκευσης	73
6.2.1	Γενικά	73
6.2.2	Μηχανή Αποθήκευσης MyISAM	74
6.3	Σύνδεση στη βάση	74
6.4	Ασφάλεια Βάσης Δεδομένων	74
6.5	Αντίγραφο ασφαλείας	75
7.	Βιβλιογραφία	79

# 1.Περίληψη

Η παρούσα πτυχιακή εργασία ασχολείται με την θεωρητική ανάλυση, μελέτη και κατασκευή ενός ηλεκτρονικού καταστήματος (e-shop) για παραγγελία φαγητού. Σκοπός της εργασίας ήταν ο σχεδιασμός και η κατασκευή μιας δυναμικής ιστοσελίδας, που θα ενσωμάτωνε όλες τις απαραίτητες λειτουργίες και ευκολίες τόσο για τον χρήστη της ιστοσελίδας όσο και για τον ιδιοκτήτη-διαχειριστή του ηλεκτρονικού καταστήματος. Το βασικό λογισμικό πάνω στο οποίο βασίστηκε η ιστοσελίδα είναι : η γλώσσα scripting “PHP” στην οποία γράφτηκε ο κώδικας, το σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων “MySQL” για την διαχείριση των δεδομένων και των πληροφοριών που απαιτούνται για την ομαλή λειτουργία της ιστοσελίδας και το λογισμικό “Apache HTTP server” για τη δημιουργία του διακομιστή (server) πάνω στον οποίο θα “τρέχει” η ιστοσελίδα.

## 2. Διαδίκτυο και η αρχιτεκτονική 3 στρωμάτων

### 2.1 Το Διαδίκτυο (Internet)

Διαδίκτυο είναι ένα δίκτυο ηλεκτρονικών υπολογιστών που (δια)συνδέει άλλα δίκτυα. Ο αντίστοιχος αγγλικός όρος *internet* προκύπτει από τη σύνθεση λέξεων *inter-network*.

Στην πιο εξειδικευμένη και περισσότερο χρησιμοποιούμενη μορφή, με τους όρους Διαδίκτυο, Ίντερνετ περιγράφεται το παγκόσμιο πλέγμα διασυνδεδεμένων υπολογιστών περιλαμβανομένων και των υπηρεσιών και πληροφοριών που παρέχει στους χρήστες του. Το Διαδίκτυο χρησιμοποιεί την μεταγωγική μετάδοση πακέτων (*packet switching*) και το πρωτόκολλο επικοινωνίας TCP/IP. Έτσι ο όρος *διαδίκτυο* κατέληξε να αναφέρεται στο παγκόσμιο αυτό δίκτυο. Η τεχνική της σύνδεσης δικτύων με αυτό τον τρόπο ονομάζεται *internetworking*.

#### 2.1.1 Η ιστορία του διαδικτύου

Ο πυρήνας του Διαδικτύου ξεκίνησε το 1969 με την ονομασία ARPANET στις Ηνωμένες Πολιτείες στην Υπηρεσία Προηγμένων Αμυντικών Ερευνών (Defense Advanced Research Projects Agency, *DARPA*) του υπουργείου Άμυνας των ΗΠΑ. Η αρχική έρευνα που συνέβαλε στο ARPANET περιλάμβανε εργασίες στα αποκεντρωμένα δίκτυα, *queueing theory* και ανταλλαγή πακέτων *packet switching*. Το 1983 το ARPANET άλλαξε το βασικό του δικτυακό πρωτόκολλο επικοινωνίας από το NCP στο TCP/IP ξεκινώντας έτσι το Διαδίκτυο όπως το γνωρίζουμε σήμερα.

Ένα σημαντικό βήμα στην ανάπτυξη έκανε το Εθνικό Ίδρυμα Επιστημών (National Science Foundation, *NSF*) των ΗΠΑ το οποίο έχτισε την πανεπιστημιακή ραχοκοκαλιά, το NSFNet, το 1986. Σημαντικά διαφορετικά δίκτυα που έχουν επιτυχώς ενσωματωθεί στο Διαδίκτυο είναι μεταξύ των άλλων το Usenet, το Fidonet και το Bitnet. Στη δεκαετία του 1990 το Διαδίκτυο προσάρμοσε επιτυχώς την πλειοψηφία των παλιότερων δικτύων υπολογιστών.

### **2.1.2 Το διαδίκτυο σήμερα**

Το Διαδίκτυο συγκροτείται από αμφίπλευρα ή πολύπλευρα εμπορικά συμβόλαια (π.χ. ομότιμες συμφωνίες) και από τεχνικές προδιαγραφές ή πρωτόκολλα που περιγράφουν την ανταλλαγή δεδομένων στο δίκτυο. Τα πρωτόκολλα αυτά μορφοποιούνται με συζητήσεις μέσα στο Internet Engineering Task Force (IETF) και τις ομάδες εργασίας του, οι οποίες είναι ανοιχτές για δημόσια συμμετοχή και κριτική. Αυτές οι επιτροπές παράγουν κείμενα που είναι γνωστά ως Αιτήματα για Σχολιασμό (ΑΓΣ). Ορισμένα ΑΓΣ εγείρονται από την κατάσταση του Προτύπου Διαδικτύου από το Συμβούλιο Αρχιτεκτονικής του Διαδικτύου (IAB).

Μερικά από τα πιο γνωστά πρωτόκολλα Internet protocol suite είναι το IP, TCP, το UDP, το DNS, το PPP, το SLIP, το ICMP, το POP3, IMAP, το SMTP, το HTTP, το HTTPS, το SSH, το Telnet, το FTP, το LDAP και το SSL. Μερικές από τις πιο γνωστές υπηρεσίες του internet που χρησιμοποιούν αυτά τα πρωτόκολλα είναι το ηλεκτρονικό ταχυδρομείο (e-mail), τα Usenet newsgroups, η διαμοίραση αρχείων (file sharing), ο Παγκόσμιος Ιστός (World Wide Web), το Gopher, το session access, το WAIS, το finger, το IRC, το MUD και το MUSH. Από αυτές, το ηλεκτρονικό ταχυδρομείο και ο Παγκόσμιος Ιστός είναι οι πιο ευρέως χρησιμοποιούμενες, ενώ πολλές άλλες υπηρεσίες έχουν βασιστεί πάνω σε αυτές, όπως ταχυδρομικές λίστες και αρχεία καταγραφής ιστού. Το Διαδίκτυο καθιστά δυνατή τη διάθεση υπηρεσιών σε πραγματικό χρόνο όπως ραδιόφωνο μέσω Ιστού και προβλέψεις μέσω Ιστού που μπορούν να προσπελαστούν από οπουδήποτε στον κόσμο.

### **2.1.3 Η κουλτούρα του διαδικτύου**

Το Ίντερνετ επίσης έχει μία μεγάλη επίδραση στην γνώση και τη διαμόρφωση απόψεων. Μέσα από την αναζήτηση λέξεων-κλειδιών (key words) μέσω της χρήσης μηχανών αναζήτησης, όπως το Google, εκατομμύρια άνθρωποι έχουν εύκολη και άμεση πρόσβαση σε ένα τεράστιο, παγκόσμιο και ποικίλο όγκο πληροφοριών. Συγκρινόμενο με τις έντυπες εγκυκλοπαίδειες και τις παραδοσιακές βιβλιοθήκες, το Ίντερνετ αντιπροσωπεύει μία ξαφνική και απότομη αποκέντρωση των πληροφοριών και των δεδομένων. Η γλώσσα που χρησιμοποιείται περισσότερο για την επικοινωνία στο Ίντερνετ είναι η Αγγλική κυρίως λόγω της καταγωγής του Ίντερνετ, της χρήσης της Αγγλικής στον προγραμματισμό λογισμικού και στην αδυναμία των πρώτων

γενιών υπολογιστών να χρησιμοποιήσουν άλλους χαρακτήρες πέραν του λατινικού αλφάβητου. Το δίκτυο μεγάλωσε αρκετά τα τελευταία χρόνια και επαρκές περιεχόμενο είναι πλέον διαθέσιμο στις γλώσσες των περισσότερο ανεπτυγμένων χωρών.

#### **2.1.4 Ο Παγκόσμιος Ιστός ( World Wide Web )**

Το 1993, το εργαστήριο CERN στην Ελβετία παρουσιάζει τον Παγκόσμιο Ιστό (World Wide Web - WWW) του Tim Berners-Lee.[Lee00]. Πρόκειται για ένα σύστημα διασύνδεσης πληροφοριών multimedia και παρουσίασής τους σε ηλεκτρονικές σελίδες. Το γραφικό αυτό περιβάλλον κάνει την εξερεύνηση του Internet πιο προσιτή στον απλό χρήστη. Παράλληλα, εμφανίζονται διάφορα εμπορικά δίκτυα που ανήκουν σε εταιρίες παροχής υπηρεσιών Διαδικτύου (Internet Service Providers - ISP) και προσφέρουν πρόσβαση σε όλους. Οποιοσδήποτε διαθέτει PC και modem μπορεί να συνδεθεί με το Internet. Το 1995, το NSFnet καταργείται πλέον επίσημα και το φορτίο του μεταφέρεται σε εμπορικά δίκτυα.

Το Διαδίκτυο, από το 1995 και εφεξής, άρχισε να λαμβάνει τη μορφή με την οποία μας είναι γνωστό σήμερα. Πλέον το μεγαλύτερο μέρος του πληθυσμού του πλανήτη ζει σε χώρες συνδεδεμένες στο Internet. Καθημερινά άνθρωποι από όλες τις γωνιές της γης συνδέονται στο Internet προκειμένου να εργαστούν, να ενημερωθούν, να επικοινωνήσουν μεταξύ τους στέλνοντας ηλεκτρονική αλληλογραφία, να ανταλλάξουν αρχεία, να κάνουν τις αγορές τους ή απλά να διασκεδάσουν.

#### **2.1.5 Διαδίκτυο 2.0 (Web 2.0)**

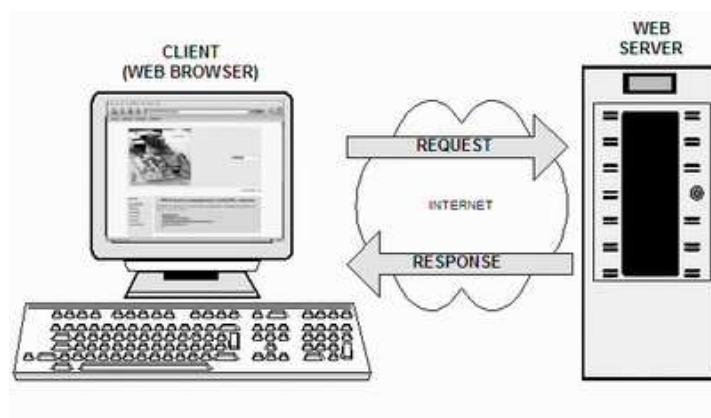
Το "Web 2.0" είναι μια φράση που έχει αποδοθεί στους O'Reilly Media το 2004 και αναφέρεται στο Διαδίκτυο της δεύτερης γενιάς, κυρίως για παροχές υπηρεσιών μεταξύ των χρηστών με συμμετοχή και συνεργασία, όπως κοινωνικές διασυνδέσεις, και τη χρήση εργαλείων όπως τα wikis, εργαλείων οργάνωσης και επικοινωνίας, και διαδικτυακές κοινότητες. Αυτό που δηλώνει κυρίως το Διαδίκτυο 2.0 είναι οι συνεργατικές διαδικασίες μέσω του Διαδικτύου και η συνεισφορά των χρηστών. Οι O'Reilly Media, σε συνεργασία με τους MediaLive International, χρησιμοποίησαν τη φράση ως τίτλο σε σειρά από συνέδρια, και από το 2004 πολλά



μέλη τεχνικών υπηρεσιών αλλά και μέλη διαδικτυακών κοινοτήτων υιοθέτησαν τον όρο.

## 2.2 Η αρχιτεκτονική πελάτη-εξυπηρετητή (client-server)

Το μοντέλο που αποτελεί την αρχιτεκτονική πελάτη-εξυπηρετητή (client-server) ή την αρχιτεκτονική 2 στρωμάτων (Two-Tier) και χρησιμοποιείται στο διαδίκτυο περιλαμβάνει μια σειρά διαδικασιών, την οποία εμείς ως χρήστες δεν αντιλαμβανόμαστε. Οι χρήστες υλοποιούν την πλοήγηση τους στον Παγκόσμιο Ιστό (Web) μέσω ενός προγράμματος περιήγησης (web browser). Το πρόγραμμα αυτό, δηλαδή ο browser αποστέλλει αιτήσεις σε έναν εξυπηρετητή δικτύου (web server) προκειμένου να ζητήσει κάποιες πληροφορίες. Ο server ανταποκρίνεται με το ζητούμενο περιεχόμενο, το οποίο προβάλλεται κατάλληλα διαμορφωμένο στον πελάτη (client), που στην περίπτωση αυτή είναι ο web browser. Στην εικόνα 2-2 που ακολουθεί αναπαρίσταται η αρχιτεκτονική αυτή κατά την αίτηση μιας HTML σελίδας από το διαδίκτυο. Στην παρακάτω εικόνα 2-2 αναπαρίσταται η αρχιτεκτονική πελάτη-εξυπηρετητή κατά την αίτηση μιας HTML σελίδας από το διαδίκτυο.



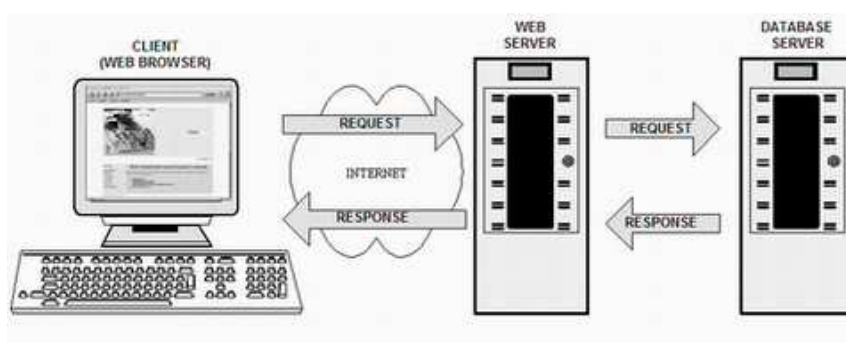
Εικόνα 2-2. Η αρχιτεκτονική πελάτη – εξυπηρετητή

### 2.2.1 Περιγραφή των δύο στρωμάτων

Το μοντέλο πελάτη-εξυπηρετητή ήταν το κυρίαρχο μοντέλο στις δικτυακές εφαρμογές, όσο το περιεχόμενο των ιστοσελίδων ήταν κυρίως στατικό και δεν υπήρχαν μεγάλες απαιτήσεις για την εξυπηρέτηση πολλών χρηστών ταυτόχρονα. Με την αύξηση της χρήσης του Internet, οι περισσότερες δικτυακές εφαρμογές

υλοποιούνται πλέον δυναμικά. Αυτό σημαίνει ότι οι σημερινές δικτυακές εφαρμογές απαιτούν πολύπλοκες λειτουργίες σε δεδομένα οι οποίες είναι καλύτερο να διενεργούνται από ένα ξεχωριστό σύστημα διαχείρισης βάσεων δεδομένων. Τέτοιες εφαρμογές είναι για παράδειγμα οι online αγορές, τα διαφόρων ειδών portals, web-mail, forums κ.α. Επίσης, όλες οι εταιρίες πλέον αναπτύσσουν site με δυναμικό περιεχόμενο που προϋποθέτουν χρήση βάσης δεδομένων.

Οι ανάγκες αυτές είχαν ως συνέπεια την ανάπτυξη μιας πιο πολύπλοκης αρχιτεκτονικής με τρία στρώματα. Ο web browser εξακολουθεί να είναι το στρώμα πελάτη (client-tier) το οποίο επικοινωνεί με τον εξυπηρετητή δικτύου. Ο εξυπηρετητής δικτύου γίνεται το μεσαίο στρώμα, το οποίο περιέχει τη δικτυακή εφαρμογή, και προστίθεται ένας εξυπηρετητής βάσης δεδομένων (Database server) στο τρίτο στρώμα, ο οποίος αναλαμβάνει την αποθήκευση των δεδομένων και την εξυπηρέτηση των ερωτήσεων της δικτυακής εφαρμογής προς τη βάση. Η παρακάτω εικόνα μας παρουσιάζει την αρχιτεκτονική τριών στρωμάτων.



*Εικόνα 2-3. Η αρχιτεκτονική τριών στρωμάτων*

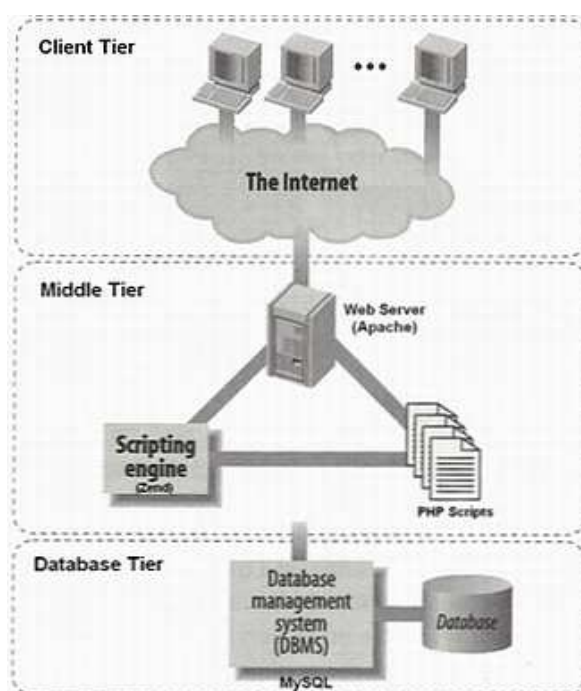
## **2.2.2 Πλεονεκτήματα των τριών στρωμάτων**

Ένα από τα πλεονεκτήματα της αρχιτεκτονικής τριών στρωμάτων έγκειται στο γεγονός ότι η εφαρμογή που βρίσκεται στον web server είναι ανεξάρτητη από τον τρόπο αποθήκευσης και προσπέλασης των δεδομένων γιατί πλέον ο database server είναι υπεύθυνος για την ασφαλή αποθήκευση και την γρήγορη προσπέλαση των δεδομένων. Αυτό έχει ως αποτέλεσμα, την ταυτόχρονη εξυπηρέτηση πολλών συνδεδεμένων χρηστών, αφού ο web server δεν επιβαρύνεται με την αποθήκευση των δεδομένων και ο database server είναι ικανός για την ασφαλή εξυπηρέτηση πολλών ταυτόχρονων αιτημάτων. Επίσης, είναι εφικτό να πραγματοποιηθούν αλλαγές σε

κάποιο στρώμα χωρίς να χρειάζεται απαραίτητα να αλλαχθούν και τα άλλα στρώματα. Τέλος, η αρχιτεκτονική τριών στρωμάτων παρέχει ευελιξία, προσφέρει καλύτερες επιδόσεις, ευκολότερη συντήρηση και δυνατότητα επέκτασης σε σύγκριση με την αρχιτεκτονική πελάτη-εξυπηρετητή.

### 2.2.3 Αρχιτεκτονική Υλοποίησης

Πρακτικά υπάρχουν πολλές διαφορετικές υλοποιήσεις δικτυακών εφαρμογών που ανήκουν στο μοντέλο της αρχιτεκτονικής τριών στρωμάτων. Η απλούστερη μορφή υλοποίησης είναι αυτή που περιλαμβάνει τον web server και το σύστημα διαχείρισης βάσεων δεδομένων εγκατεστημένα στο ίδιο φυσικό μηχάνημα. Αυτή η υλοποίηση είναι η πιο απλή, ασφαλέστερη και έχει την δυνατότητα να εξυπηρετήσει μεγάλο αριθμό αιτήσεων ανά ώρα. Γι' αυτούς ακριβώς τους λόγους εφαρμόσαμε και εμείς αυτή την υλοποίηση, η οποία παρουσιάζεται στο παρακάτω σχήμα.



Εικόνα 2-4. Η αρχιτεκτονική τριών στρωμάτων για τη δικτυακή εφαρμογή μας

Άλλος τρόπος υλοποίησης της αρχιτεκτονικής τριών στρωμάτων είναι με εγκατάσταση του web server και του database server σε διαφορετικά μηχανήματα, ώστε περισσότεροι πόροι να είναι διαθέσιμοι και να επιτυγχάνεται ταχύτερη απόκριση. Τέλος, παρέχεται η δυνατότητα για περαιτέρω διάσπαση του μεσαίου

στρώματος σε περισσότερα στρώματα, τα οποία αναλαμβάνουν διακριτούς ρόλους, όπως για παράδειγμα παρουσίαση των δεδομένων (data presentation), επεξεργασία δεδομένων (data process), οπότε μιλάμε πλέον για αρχιτεκτονική n-στρωμάτων (n-tier).

## 3. Τεχνολογία που χρησιμοποιήθηκε

### 3.1 Διακομιστές

#### 3.1.1 Διακομιστές Web

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

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

#### 3.1.2 Apache και το Κίνημα Ανοικτού Κώδικα (open source)

Ο Apache είναι ο πιο διάσημος διακομιστής που χρησιμοποιείται στο web. Είναι διαθέσιμος για μια ποικιλία διαφορετικών λειτουργικών συστημάτων, με το UNIX να είναι το πιο διάσημο. Ο Apache αποτελεί σημαντικό μέρος ενός κινήματος

γνωστού ως Κίνημα Ανοικτού Κώδικα. Η φιλοσοφία πίσω από αυτό είναι ότι το λογισμικό έχει γίνει τόσο "μπαρόκ" και περίπλοκο που καμία εταιρεία δεν μπορεί να έχει πλήρη γνώση όλων των προβλημάτων και των λαθών που μπορεί να εμφανιστούν. Ο μόνος τρόπος με τον οποίο μπορεί να γίνει αυτό είναι μέσω της διάθεσης σε μια μεγάλη κοινότητα προγραμματιστών του πηγαίου κώδικα και της άδειας σε αυτούς να αναζητήσουν και να διορθώσουν προβλήματα σ' αυτόν. Μ' αυτόν τον τρόπο θα υπάρχει μία γρήγορη διάδοση διορθώσεων και βελτιώσεων των προγραμμάτων.

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

Έχει την ικανότητα να εξυπηρετεί στατικό και δυναμικό περιεχόμενο στο περιβάλλον του παγκόσμιου ιστού. Η υποστήριξη που μπορεί να προσφέρει συνίσταται σε:

- Γλώσσες όπως η PHP, Python, Perl καθώς και άλλες γλώσσες προγραμματισμού
- Πρωτόκολλα SSL και TLS
- Σχήματα ταυτοποίησης
- Επεκτάσεις ταυτοποίησης
- Λειτουργίες επανεγγραφής URL
- Φιλτράρισμα μέσω `mod_include` και `mod_filtrer`

Ο Apache επιπλέον είναι παραμετροποιήσιμος σε μεγάλο βαθμό. Βέβαια η σωστή εγκατάσταση του και η παραμετροποίηση του δεν είναι εύκολη διαδικασία. Έτσι ένας αρχάριος χρήστης υπολογιστών πολύ δύσκολα θα καταφέρει να στήσει ένα server. Ακόμα και μετά την αρχική εγκατάσταση του, απαιτείται επέμβαση σε κάποια αρχεία ώστε να ρυθμιστούν κάποιες παράμετροι για την σωστή, σταθερή και αποδοτική λειτουργία του. Βεβαίως σε καμία περίπτωση δε σημαίνει ότι πρέπει κάποιος να αποθαρρύνεται στη χρήση του Apache λόγω απαιτήσεων στην εγκατάσταση.

Εξάλλου οποιοδήποτε software για στήσιμο server δεν είναι υπόθεση ενός νέου και αρχάριου χρήστη υπολογιστών.

### 3.1.3 Διαχείριση Διακομιστή Web

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

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

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

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

Να θέσει όρους ασφαλείας οι οποίοι επιτρέπουν ή αποτρέπουν την πρόσβαση στους χρήστες. Για παράδειγμα, κάποιες σελίδες ενός ιστοχώρου είναι προσβάσιμες μόνο από συνδρομητές. Στην περίπτωση αυτή ο webmaster ενεργοποιεί κάποια κατάλληλη ρύθμιση στον διακομιστή Web.

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

### 3.1.4 Ρυθμίσεις εγκατάστασης Apache 2.2

Ο παρακάτω κώδικας θα καταχωρηθεί στο httpd.conf αρχείο του apache μας, προκειμένου η εφαρμογή να δουλεύει ως εικονικό site τοπικά και να προσθέσουμε την php στην διαμόρφωση του apache. Έτσι, στην γραμμή διεύθυνσης του browser, θα γράψουμε localhost και θα εμφανιστεί ο δικτυακός τόπος του site μας.

```
LoadModule php5_module c:/php/php5apache.dll
```

```
AddModule mod_php5.c
```

```
AddType application/x-httpd-php .php
```

Action application/x-httpd-php “/php/php.exe”

## 3.2 HTML

### 3.2.1 Τι είναι HTML

Τα αρχικά **HTML** προέρχονται από τις λέξεις **HyperText Markup Language**. Η html δεν είναι μια γλώσσα προγραμματισμού. Είναι μια γλώσσα σήμανσης (*markup language*), δηλαδή ένας ειδικός τρόπος γραφής κειμένου. Ο καθένας μπορεί να δημιουργήσει ένα αρχείο HTML χρησιμοποιώντας απλώς έναν επεξεργαστή κειμένου. Αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα. Ο browser αναγνωρίζει αυτόν τον τρόπο γραφής και εκτελεί τις εντολές που περιέχονται σε αυτόν. Αξίζει να σημειωθεί ότι η html είναι η πρώτη και πιο διαδεδομένη γλώσσα περιγραφής της δομής μιας ιστοσελίδας. Η html χρησιμοποιεί τις ειδικές ετικέτες (τα tags) να δώσει τις απαραίτητες οδηγίες στον browser. Τα tags είναι εντολές που συνήθως ορίζουν την αρχή ή το τέλος μιας λειτουργίας. Τα tags βρίσκονται πάντα μεταξύ των συμβόλων < και >. Π.χ. <BODY> Οι οδηγίες είναι case insensitive, δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά (μικρά) ή κεφαλαία. Ένα αρχείο HTML πρέπει να έχει κατάληξη htm ή html.

### 3.2.2 Εκδόσεις HTML

Το 1990 ο Tim Berners-Lee από το Cern, το εργαστήριο φυσικής της Γενεύης, δημιούργησε ένα νέο πρωτόκολλο με το οποίο θα μπορούσαν να μεταφέρονται κάθε είδος αρχείων και αντικειμένων μέσα από το Internet. Το πρωτόκολλο αυτό ονομάστηκε HTTP (HyperText Transfer Protocol) και σηματοδότησε την αρχή του WWW όπως το ξέρουμε σήμερα. Οι σελίδες που ήταν η βάση του WWW ήταν γραμμένες στην πρώτη έκδοση της γλώσσα HTML. 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 την αντικατέστησε η έκδοση 3.2 (1996). Η τελευταία περιλάμβανε πολλές

από τις σημάνσεις (tags) που είχαν εισάγει οι δύο εταιρίες. Η έκδοση 4.0 παρουσιάστηκε τον Ιούνιο του 1997.

### **3.2.3 Γλώσσες Προγραμματισμού στο Internet**

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

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

## **3.3 Cascading Style Sheets - CSS**

### **3.3.1 Περιγραφή**

Τα Cascading Style Sheets (CSS) είναι οδηγίες που χρησιμοποιούνται από την HTML και την XHTML για να ορίσουν την εμφάνιση των ιστοσελίδων του Διαδικτύου. Εφαρμόζεται σε κάθε στοιχείο της σελίδας ξεχωριστά (χρώματα, γραμματοσειρές, τοποθέτηση εικόνων και άλλων στοιχείων κλπ).

Το CSS σχεδιάστηκε με σκοπό να προωθήσει το διαχωρισμό της πληροφορίας και της εμφάνισης για να μπορούν όλο και περισσότεροι χρήστες να έχουν πρόσβαση στη πληροφορία των ιστοσελίδων. Μειώνει κατά πολύ τον κώδικα των ιστοσελίδων,



αφού εξαλείφει τον επαναλαμβανόμενο κώδικα, με αποτέλεσμα να βοηθάει στη δημιουργία πιο γρήγορων ιστοσελίδων, οι οποίες μάλιστα είναι και προσβάσιμες σε μεγαλύτερο βαθμό από τις μηχανές αναζήτησης. Πρωτοεμφανίστηκε το 1996, αλλά η ολοκληρωμένη υποστήριξη από τους πιο δημοφιλείς browsers δεν έγινε μέχρι το 2000.

### 3.3.2. Πλεονεκτήματα - Μειονεκτήματα των CSS

Τα πλεονεκτήματα που προκύπτουν με την χρήση των CSS είναι:

- Οι σελίδες έχουν λιγότερο κώδικα, μικρότερο μέγεθος και έτσι εμφανίζονται ταχύτερα στον browser
- Καθιστά εύκολη την μορφοποίηση των HTML σελίδων
- Η γρήγορη και εύκολη αλλαγή της μορφοποίησης πολλών HTML σελίδων ταυτόχρονα, αλλάζοντας απλά μόνο το CSS αρχείο

Μειονεκτήματα:

- Οι παλιοί browsers δεν υποστηρίζουν CSS
- Ορισμένοι browsers δείχνουν με διαφορετικό τρόπο τις ίδιες σελίδες

### 3.3.3 Βασική δομή και κανόνες

**Για την εισαγωγή του CSS αρχείου στην HTML σελίδα υπάρχουν οι εξής τρόποι:**

1. Τοποθέτηση των CSS εντολών στην ετικέτα (tag) head
2. Τοποθέτηση μιας αναφοράς σε αρχείο που περιέχει τις εντολές
3. Τοποθέτηση του παρακάτω κώδικα μέσα στην ετικέτα head

Συνήθως χρησιμοποιούμε ξεχωριστό αρχείο για το styling με css και για την εισαγωγή του στην html σελίδα, χρησιμοποιούμε τον δεύτερο ή τον τρίτο τρόπο. Το παράδειγμα που ακολουθεί παρουσιάζει τον τρίτο τρόπο εισαγωγής που χρησιμοποιήθηκε στην εφαρμογή. Στο παράδειγμα μας το αρχείο ονομάζεται styles.css και βρίσκετε στον φάκελο ptikiaki.

```
<style type="text/css" media="screen"> <!-- @import url(ptixiaki/styles.css); --> </style>
```

## Σύνταξη

Επιλογέας { ιδιότητα: τιμή; } Ο επιλογέας καθορίζει πιο αντικείμενο της ιστοσελίδας θα επηρεάσει η μορφοποίηση, η ιδιότητα καθορίζει πιο χαρακτηριστικό του αντικειμένου θα επηρεαστεί και η τιμή είναι η τιμή που θα πάρει αυτό το χαρακτηριστικό. Οι χαρακτήρες '{' και '}' χρησιμοποιούνται για να περικλείσουμε τη λίστα των στυλ για κάθε ομάδα επιλογέων.

### Ο Επιλογέας (selector)

Ο επιλογέας καθορίζει ποιο αντικείμενο της ιστοσελίδας μας θα επηρεάσει η μορφοποίηση, οπότε μπορεί να είναι κάθε ετικέτα (tag) της html, όπως για παράδειγμα είναι οι ετικέτες body,, li, td κ.α. Μπορούμε να καθορίσουμε την ίδια μορφοποίηση (στυλ) για πολλούς επιλογείς .

**Όπως στο παρακάτω παράδειγμα:**

```
body, td, li{  
    background-color: #ffffff;  
}
```

### Η ιδιότητα (property)

Η ιδιότητα καθορίζει το χαρακτηριστικό του επιλογέα που θα επηρεαστεί.

**Παράδειγμα για τον επιλογέα *body* ακολουθούν:**

```
body{  
    background-color: red;  
    margin: 1px;  
    font-size: 11px;  
    color: green;  
}
```

### Η τιμή (value)

Η τιμή καθορίζει τη μέτρηση της ιδιότητας και μπορεί να είναι αριθμοί ή λέξεις.

Οι τιμές που χρησιμοποιούμε για μεγέθη μπορούν να είναι:

- pixels (px)
- ποσοστά (%)
- em, ένα em ...

- in, ίντσες
- cm, εκατοστά
- mm, χιλιοστά
- pt, στιγμές (points) - ένα pt είναι αντίστοιχο με το 1/72 της ίντσας.
- pc, pica - ένα pc είναι ανάλογο με 12 στιγμές (pt).

**Για χρώματα μπορείτε να χρησιμοποιείτε τις παρακάτω τιμές:**

- δεκαεξαδικές τιμές χρωμάτων σε τριάδες RGB π.χ. #FF0000.
- ονόματα χρωμάτων π.χ. red
- δεκαδικές τιμές χρωμάτων σε τριάδες RGB π.χ. rgb(255,0,0)
- ποσοστιαίες τιμές χρωμάτων σε τριάδες RGB π.χ. rgb(100%,0%,0%)

### **Σχόλια**

Τα σχόλια σε css αρχεία έχουν την παρακάτω μορφή

/\* ένα σχόλιο μέσα σε αρχείο CSS \*/

### **Κλάσεις (Classes)**

Οι κλάσεις ομαδοποιούν ετικέτες σε ξεχωριστές ομάδες, οπότε μπορούμε να τους δώσουμε ξεχωριστό στυλ. Ο επιλογέας μιας κλάσης ξεκινάει πάντα με τελεία (.) κατά τα άλλα ο καθορισμός στυλ σε κλάσεις είναι ίδιος όπως και γενικά σε ετικέτες.

### **Ταυτότητες (Ids)**

Τα IDs είναι σχεδόν ίδια με τις κλάσεις, μόνο που χρησιμοποιούνται μόνο μια φορά σε κάθε σελίδα. Συνήθως τα id χρησιμοποιούνται για να δημιουργούμε το στήσιμο μιας ιστοσελίδας (layout) με τη χρήση της ετικέτας *div*, όπου καθορίζουν το χρώμα και την εικόνα για το φόντο (background/background-image), το μέγεθος, τη θέση και τη στοίχιση αυτών των divs, ενώ το στυλ και το μέγεθος του κειμένου καθορίζεται με κλάσεις. Για να καθορίσουμε το στυλ ενός id κάνουμε το ίδιο, εκτός του ότι χρησιμοποιούμε # στη θέση της τελείας (.).

### **Κληρονομικότητα**

Ουσιαστικά όλοι οι επιλογείς που τοποθετούνται μέσα σε άλλους επιλογείς θα κληρονομήσουν τις τιμές για κάθε ιδιότητα που ορίζετε στον εξωτερικό επιλογέα

εκτός αν είναι διαφορετικά ορισμένο. π.χ. ένα χρώμα που καθορίζεται για το BODY θα εφαρμοστεί επίσης στο κείμενο μιας παραγράφου. Υπάρχουν μερικές περιπτώσεις όπου ο εσωτερικός επιλογέας δεν κληρονομεί τις τιμές του εξωτερικού επιλογέα, αλλά αυτές πρέπει να ξεχωρίζουν λογικά. π.χ. η ιδιότητα των περιθωρίων δεν κληρονομείται, μια παράγραφος δεν θα είχε το ίδιο περιθώριο με το σώμα εγγράφου.

## **3.4 PHP**

### **3.4.1 Τι είναι η PHP;**

Η PHP, της οποίας τα αρχικά αντιπροσωπεύουν το "PHP: Hypertext Preprocessor" είναι μια ευρέως χρησιμοποιούμενη, ανοιχτού κώδικα scripting γλώσσα προγραμματισμού, η οποία είναι ειδικά κατάλληλη για ανάπτυξη εφαρμογών για το Web και μπορεί να ενσωματωθεί στην HTML. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.

### **3.4.2 Ιστορία της PHP**

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, βασιζόμενη αυτή τη φορά στη γλώσσα C και αριθμώντας περισσότερους από 50.000 ιστότοποι που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία

συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Οι περισσότεροι ιστότοποι επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP.

### 3.4.3 Τι μπορεί να κάνει η PHP;

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

Υπάρχουν τρεις κύριοι τομείς που χρησιμοποιείται ένα PHP script.

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

platform εφαρμογών με αυτόν τον τρόπο. Το PHP-GTK είναι μια επέκταση της PHP και δεν συμπεριλαμβάνεται στην κύρια διανομή.

### 3.4.4 Χαρακτηριστικά της php

- Είναι γλώσσα ανοιχτού κώδικα (Open Source), το οποίο σημαίνει ότι διατίθεται δωρεάν, και ότι υποστηρίζεται από μια διεθνής κοινότητα προγραμματιστών.
- Είναι εύκολη στην κατανόηση (η σύνταξη μοιάζει πολύ με της C) και γρήγορη στη ανάπτυξη του κώδικα, με αποτέλεσμα να μειώνει κατά πολύ το χρόνο ανάπτυξης μιας εφαρμογής σε σχέση με πιο πολύπλοκες γλώσσες
- Έχει πολύ καλές επιδόσεις και δεν απαιτεί ακριβό εξοπλισμό.
- Μπορεί να χρησιμοποιηθεί σε όλα τα κύρια λειτουργικά συστήματα, συμπεριλαμβανομένου του Linux, πολλών εκδοχών του Unix (HP-UX, Solaris και OpenBSD), Microsoft Windows, Mac OS X.
- Η PHP υποστηρίζει τους Apache, Microsoft Internet Information Server, Personal Web Server, Netscape και iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, και πολλούς άλλους webserver. Για την πλειοψηφία των server η PHP έχει ένα module, για τους υπόλοιπους η PHP μπορεί να λειτουργήσει ως ένας CGI επεξεργαστής.
- Υποστηρίζει μεγάλο σύνολο βάσεων δεδομένων, μεταξύ των οποίων είναι η MySQL, PostgreSQL, Oracle, Informix, Sybase, κ.α. Επίσης, υπάρχει και μια αφαιρετική επέκταση DBX βάσεων δεδομένων (DBX database abstraction extension) που επιτρέπει διάφανα να χρησιμοποιηθεί οποιαδήποτε βάση δεδομένων υποστηρίζεται από αυτή την επέκταση. Επιπλέον η PHP υποστηρίζει το ODBC, το Open Database Connection standard (Ανοιχτό πρότυπο Σύνδεσης Βάσεων δεδομένων) έτσι μπορείτε να συνδεθείτε σε οποιαδήποτε βάση δεδομένων που υποστηρίζει αυτό το παγκόσμιο πρότυπο.
- Οι δυνατότητες της PHP συμπεριλαμβάνουν την εξαγωγή εικόνων, αρχείων PDF, ακόμη και ταινίες Flash (χρησιμοποιώντας τα libswf και Ming) παράγονται αμέσως. Μπορείτε επίσης να εξάγετε εύκολα οποιοδήποτε κείμενο όπως XHTML και οποιοδήποτε άλλο XML αρχείο. Η PHP μπορεί να δημιουργεί αυτόματα αυτά τα αρχεία και να τα αποθηκεύει στο σύστημα

αρχείων, αντί να τα εκτυπώνει, αποτελώντας έτσι μια server-side cache για το δυναμικό σας περιεχόμενο.

- Η PHP έχει εξαιρετικά χρήσιμα χαρακτηριστικά επεξεργασίας κειμένων, από την POSIX επέκταση ή τις Perl regular expressions μέχρι XML parsing αρχείων. Για τη μεταγλώττιση και την πρόσβαση αρχείων XML, υποστηρίζουμε τα πρότυπα SAX και DOM.

Έτσι με την PHP έχετε την ελευθερία επιλογής ενός λειτουργικού συστήματος και ενός web server. Επιπλέον, έχετε επίσης την ελευθερία να χρησιμοποιήσετε συναρτησιακό (procedural) ή αντικειμενοστραφή (object oriented) προγραμματισμό ή μια ανάμειξη τους. Αν και η παρούσα έκδοση δεν υποστηρίζει όλα τα πρότυπα χαρακτηριστικά, μεγάλες βιβλιοθήκες κώδικα και μεγάλες εφαρμογές (συμπεριλαμβανομένης και της βιβλιοθήκης PEAR) είναι γραμμένες μόνο με αντικειμενοστραφή κώδικα.

On-line tutorial : <http://de2.php.net/manual/en/tutorial.php>

### 3.4.5 Παράδειγμα χρήσης της PHP

**Εισαγωγικό παράδειγμα :**

```
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <?php
      echo "Hi, I'm a PHP script!";
    ?>
  </body>
</html>
```

Παρατηρούμε πως αυτό είναι διαφορετικό από ένα script γραμμένο σε άλλες γλώσσες προγραμματισμού όπως η Perl ή η C : Αντί να γράφετε ένα πρόγραμμα με πολλές εντολές για να εξάγετε HTML, γράφετε ένα HTML script με κάποιο ενσωματωμένο κώδικα για να κάνει κάτι (σε αυτή την περίπτωση, να εμφανίζει κάποιο κείμενο). Ο κώδικας PHP είναι εσώκλειστος σε ειδικά tags (ετικέτες) αρχής

και τέλους που σας επιτρέπουν να μεταφέρεστε μέσα και έξω από το "PHP mode" (PHP τρόπο λειτουργίας).

Αυτό που διαχωρίζει την PHP από κάτι σαν client-side JavaScript είναι ότι ο κώδικας εκτελείται στον server (εξυπηρετητή). Αν είχατε ένα script σαν το παραπάνω στον server σας, ο client θα έπαιρνε τα αποτελέσματα της εκτέλεσης αυτού του script, χωρίς να υπάρχει κανένας τρόπος να καταλάβει τι κώδικας υπάρχει από κάτω. Μπορείτε ακόμη να ρυθμίσετε τον web server σας να χειρίζεται όλα τα HTML αρχεία σας με την PHP, και τότε πραγματικά δεν υπάρχει τρόπος ο χρήστης να καταλάβει τι έχετε κάτω από το μανίκι σας.

Το καλύτερο πράγμα στην PHP είναι ότι είναι εξαιρετικά απλή για ένα νεοφερμένο αλλά προσφέρει πολλά προηγμένα χαρακτηριστικά για ένα επαγγελματία προγραμματιστή. Αν και η ανάπτυξη της PHP εστιάζεται σε server-side scripting, μπορούμε να κάνουμε πολύ περισσότερα με αυτή.

### 3.4.6 Επικοινωνία της PHP με βάσεις δεδομένων

Η PHP θεωρείται η καλύτερη επιλογή για την δημιουργία δυναμικών εφαρμογών που επικοινωνούν με βάσεις δεδομένων, ένας βασικός λόγος για να την επιλέξουμε είναι ότι διαθέτει ενσωματωμένες συναρτήσεις που επικοινωνούν με μεγάλο αριθμό εμπορικών συστημάτων βάσεων δεδομένων.

Το παρακάτω παράδειγμα παρουσιάζει τις συναρτήσεις της PHP, που χρησιμοποιούνται για τη σύνδεση με μια βάση δεδομένων στη MySQL, τη δημιουργία επερωτήσεων και την ανάκτηση των αποτελεσμάτων για περαιτέρω επεξεργασία τους από την εφαρμογή.

```
<?php
//Δημιουργία σύνδεσης προς τον MySQL Server ή εμφάνιση κωδικού σφάλματος σε
περίπτωση αποτυχημένης σύνδεσης
$dbConn = mysql_connect ($dbHost, $dbUser, $dbPass) or die ('MySQL connect
failed. ' . mysql_error());
//Επιλογή μιας βάσης δεδομένων ή εμφάνιση κωδικού σφάλματος σε περίπτωση
αποτυχημένης επιλογής
mysql_select_db($dbName) or die('Cannot select database. ' . mysql_error());
//επιλογή code της εμφάνισης των εγγραφών της βάσης
```



```

mysql_query("SET NAMES 'utf8'", $dbConn);
// Δημιουργία μιας επερώτησης (query) προς τη βάση και Επιστροφή των
αποτελεσμάτων του query
function dbQuery($sql)
{
    $result = mysql_query($sql) or die(mysql_error());
    return $result;
}
?>

```

### Επεξήγηση των συναρτήσεων:

- **mysql\_connect()** χρησιμοποιείται για τη σύνδεση με τον MySQL Server και δέχεται τις εξής παραμέτρους: το hostname ή την IP διεύθυνση του server, το όνομα του χρήστη που έχει πρόσβαση στον server και τον κωδικό ασφαλείας του. Η τιμή που επιστρέφει η συνάρτηση χρησιμοποιείται σαν παράμετρος στις μετέπειτα συναρτήσεις που καλούνται προς τον MySQL Server.
- **mysql\_error()** επιστρέφει το μήνυμα λάθους σε περίπτωση αποτυχίας της προηγούμενης ενέργειας που έγινε προς τον MySQL Server.
- **mysql\_select\_db()** διαλέγει μια συγκεκριμένη βάση δεδομένων η οποία έχει ήδη δημιουργηθεί στον MySQL Server. Οι παράμετροι που εμπεριέχονται είναι το όνομα της βάσης και η τιμή που επιστράφηκε από την mysql\_connect().
- **mysql\_query()** χρησιμοποιείται για την αποστολή μιας επερώτησης (query) στη βάση δεδομένων που ορίζει η δεύτερη παράμετρος (\$dbConn). Η πρώτη παράμετρος είναι το query που θέλουμε να εκτελεστεί. Σημαντικό να αναφέρουμε ότι και η php και mysql βάση πρέπει να χρησιμοποιούν το ίδιο codec για την αποθήκευση, ανάκτηση και εμφάνιση των εγγραφών της βάσης. Σε οποιαδήποτε άλλη περίπτωση τα δεδομένα δεν θα εμφανίζονται σωστά. Για αυτό και χρησιμοποιήσαμε το utf-8.

### 3.4.7 Ρυθμίσεις εγκαταστάσεις της PHP

Για να λειτουργήσει η php σωστά μετά την εγκατάσταση θα πρέπει να κάνουμε κάποιες ρυθμίσεις στο αρχείο php.ini έτσι ώστε να μπορεί να συνεργαστεί με τον apache 2.2 και την mysql 5. Αυτές οι ρυθμίσεις είναι:

Εντοπίστε την ενότητα που λέει:

#Example:

#Loadmodule ssl\_module modules/mod\_ssl.so

Στο τέλος αυτής της ενότητας προσθέστε:

**LoadModule php5\_module C:/php/php5apache2\_2.dll**

Επίσης προσθέστε την εντολή

**PHPIniDir "C:/php"**

Για να σιγουρευτούμε ότι ο Apache γνωρίζει που βρίσκεται το php.ini

Στο τέλος της ενότητας:

#

#AddType allows you to add to or override the MIME configuration

#file mime.types for specific file types.

#

προσθέστε της παρακάτω εντολή:

**AddType application/x-httpd-php .php .html**

//Ενεργοποίηση επεκτάσεων

**extension\_dir = c:/php/ext**

//ενεργοποίηση της βιβλιοθήκης για την mysql

**extension =phpmysql.dll**

//ενεργοποίηση της βιβλιοθήκης για αναγνώριση των εικόνων

**extension=php\_gd2.dll**

//Για να ξέρει η php τον αρχικό κατάλογο του apache

**doc\_root = "c:/program files/apache group/apache/htdocs"**

Για να κάνουμε upload τις εικόνες κάνουμε τις παρακάτω ρυθμίσεις:

**file\_uploads = On**

**upload\_tmp\_dir=**

**"c:/program files/**

**apache software foundation/apache2.2/htdocs/ptixiaki/images/temp"**

```
upload_max_filesize = 10M  
//Για την ανακατεύθυνση του header  
output_buffering = 4096  
//Για την αποστολή e-mail  
For Win32 only.  
SMTP = localhost  
SMTP = mailgate.forthnet.gr  
smtp_port = 25  
sendmail_from = chrisxina1@yahoo.gr
```

## 3.5 MYSQL

### 3.5.1 Περιγραφή

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

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

### 3.5.2 Πλεονεκτήματα της MySQL

Παρακάτω παραθέτονται μερικά από τα πλεονεκτήματα της MySQL που την καθιστούν ανταγωνιστική έναντι άλλων πακέτων διαχείρισης βάσεων δεδομένων. Τα βασικά πλεονεκτήματα είναι:

**Απόδοση.** Η MySQL είναι αρκετά γρήγορη. Πολλές δοκιμές που έχουν γίνει δείχνουν ότι υπερτερεί σε ταχύτητα έναντι των ανταγωνιστών της.

**Κόστος.** Η MySQL είναι προϊόν ανοικτού κώδικα και διατίθεται δωρεάν για προσωπική χρήση. Η εμπορική άδεια της διατίθεται σε χαμηλό κόστος. Αυτό σημαίνει ότι αν κάποιος θέλει να τη χρησιμοποιήσει για εφαρμογές προσωπικής χρήσης ή για εφαρμογές που θα είναι ανοικτού κώδικα δεν χρειάζεται να αγοράσει κάποια άδεια. Άδεια απαιτείται αν κάποιος τη χρειάζεται για εμπορικές εφαρμογές που δεν θα είναι ανοικτού κώδικα.

**Ευκολία Χρήσης.** Η MySQL είναι αρκετά εύκολη στην εκμάθηση της, ακόμα και όταν κάποιος που δεν έχει ξαναχρησιμοποιήσει παρόμοια προϊόντα κατασκευής βάσεων δεδομένων.

**Μεταφερσιμότητα.** Η MySQL μπορεί να χρησιμοποιηθεί σε πολλά σύγχρονα λειτουργικά συστήματα. Είναι συμβατή με πολλές εκδόσεις των Microsoft Windows και με λειτουργικά Unix, όπως οι διάφορες εκδόσεις του δημοφιλούς λειτουργικού ανοικτού κώδικα Linux.

**Πηγαίος κώδικας.** Η MySQL ανήκει στην οικογένεια του λογισμικού ανοικτού κώδικα όπως αναφέρθηκε προηγούμενα. Συνεπώς ο καθένας μπορεί να αποκτήσει και να τροποποιήσει τον πηγαίο κώδικα της, προσαρμόζοντας τον στις ανάγκες του ή διορθώνοντας τυχόν bugs. Επιπλέον το γεγονός ότι είναι διαθέσιμος ο πηγαίος κώδικας βοηθάει στη συνεχή ανανέωση και διόρθωση της MySQL αφού εκατομμύρια άνθρωποι σε όλο τον κόσμο εργάζονται πάνω σε αυτή. Έτσι δεν χρειάζεται κάποιος να περιμένει μια νέα επίσημη έκδοση κάποιας εταιρίας για τη διόρθωση ενός bug, αφού αυτό γίνεται πολύ γρήγορα από τους χρήστες της. Επιπλέον δε χρειάζεται να ανησυχεί κάποιος για τη μελλοντική υποστήριξη της και τη συνέχιση λειτουργίας της σε μελλοντικές συνθήκες.

**Υποστήριξη.** Στην σελίδα [www.mysql.com](http://www.mysql.com) υπάρχει μια τεράστια υποστήριξη πάνω στη MySQL με manual, tutorial, βοήθεια σε πιθανά προβλήματα.

### 3.6 JAVASCRIPT

Η JavaScript είναι μια γλώσσα σεναρίων βασισμένη στο αντικείμενο (object based scripting language) και η οποία έχει σαν σκοπό την παραγωγή δυναμικού περιεχομένου σε ιστοσελίδες. Ο κώδικας JavaScript ενσωματώνεται στον κώδικα HTML των ιστοσελίδων με την οδηγία `<script>`.

Όπως και η PHP, η Javascript έχει βασιστεί όσον αφορά τον τρόπο σύνταξης του κώδικά της στη γλώσσα προγραμματισμού C, με την οποία παρουσιάζει πολλές ομοιότητες. Όμως ενώ η PHP είναι μια server side γλώσσα προγραμματισμού, η Javascript είναι client side. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στον server, αλλά στο πρόγραμμα περιήγησης των επισκεπτών. Πιο συγκεκριμένα, η Javascript δεν έχει καμία απαίτηση από πλευράς δυνατοτήτων του server για να εκτελεστεί (επεξεργαστική ισχύ, συμβατό λογισμικό διακομιστή), αλλά βασίζεται στις δυνατότητες του browser των επισκεπτών. Επίσης μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Παρόλα αυτά, οι δυνατότητές της είναι σημαντικά μικρότερες από αυτές της PHP και δεν παρέχει συνδεσιμότητα με βάσεις δεδομένων. Ο κώδικας Javascript μιας σελίδας περικλείεται από τα HTML tags `<script language=javascript>` και `</script>`. Την απεικόνιση κειμένου αναλαμβάνει η εντολή `document.write`.

**Για παράδειγμα, ο ακόλουθος κώδικας Javascript:**

```
<script language=javascript>
document.write('Hello world!');
</script>
```

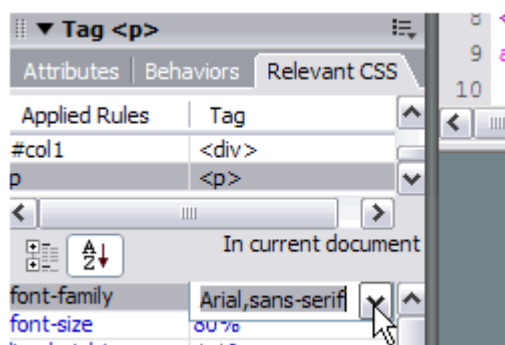
Θα απεικονίσει στο πρόγραμμα περιήγησης των επισκεπτών το κείμενο: Hello world! Αν ο κώδικας Javascript περιέχει περισσότερες από μία εντολές, αυτές θα πρέπει να διαχωριστούν μεταξύ τους με το χαρακτήρα του ελληνικού ερωτηματικού ';' (δηλαδή της λατινικής άνω τελείας). Η χρήση του χαρακτήρα αυτού για την τελευταία εντολή δεν είναι απαραίτητη. Η διαχώριση των εντολών στους νεότερους διακομιστές (browsers) δεν είναι απαραίτητη.

## 4. Προγράμματα που χρησιμοποιήθηκαν

### 4.1 Dreamweaver 8

Το Dreamweaver της Macromedia αποτελεί την επιλογή των επαγγελματιών για την κατασκευή ιστοσελίδων και εφαρμογών. Συνεχίζει την παράδοση του προϊόντος στην καινοτομία παρέχοντας ένα στιβαρό θεμέλιο για ευρεία υιοθέτηση του Cascading Style Sheet (CSS), ενός ευρέως αποδεκτού web standard, το οποίο προηγουμένως ήταν δύσκολο να εφαρμοστεί. Λόγω του ότι το CSS βοηθάει τους developers να διαχωρίζουν την παρουσίαση και το στυλ από το περιεχόμενο και οι ιστοσελίδες κατασκευασμένες με το CSS απαιτούν μικρότερο bandwidth, είναι ευκολότερο να διατηρηθούν και προσδίδουν σε developers και τελικούς χρήστες ακριβή σχεδιαστικό έλεγχο. Το Dreamweaver αποτελεί το μοναδικό εκτενές εργαλείο για να αποδώσει γερή υποστήριξη CSS σε ένα οπτικό περιβάλλον ανάπτυξης, μετακινώντας την καμπύλη μάθησης και καθιστώντας ικανούς τους developers να χρησιμοποιούν αυτό το standard, ώστε να παράγουν εκπληκτικές ιστοσελίδες και εφαρμογές για το web.

Το Dreamweaver 8 βασίζεται στην επιτυχημένη δομή του Dreamweaver MX και επεκτείνεται επιπλέον ενσωματώνοντας οπτικά εργαλεία περιγράμματος, χαρακτηριστικά ανάπτυξης εφαρμογών και υποστήριξη επεξεργασίας κώδικα μέσω ενός περιβάλλοντος τεχνολογίας ανεξαρτήτου πλατφόρμας. Υποστηρίζοντας τις σημαντικότερες τεχνολογίες HTML και server, το Dreamweaver 8 απελευθερώνει



τη δύναμη της προσβάσιμης ανάπτυξης βασισμένης σε πρότυπα για την κοινότητα web, ενώ βελτιώνει τα βασικά εργαλεία, τα οποία χρειάζονται οι επαγγελματίες. Το Dreamweaver περιλαμβάνει επίσης το SecureFTP, μια δυναμική λειτουργία επικύρωσης λειτουργικότητας με οποιονδήποτε browser, ενσωματωμένη

επεξεργασία γραφικών, ομαλή συνεργασία με τα Word και Excel της Microsoft και ενημερωμένη υποστήριξη για τεχνολογίες ASP.NET, PHP και ColdFusion server. Το Dreamweaver 8 περιλαμβάνει εξαιρετικές παραμέτρους σχεδίου βασισμένους σε

πρότυπα που εξασφαλίζουν υψηλής ποιότητας σχέδιο. Ολόκληρο το περιβάλλον σχεδίασης είναι φτιαγμένο με βάση το CSS, που επιτρέπει την ταχύτερη και αποτελεσματικότερη ανάπτυξη επαγγελματικών ιστοσελίδων καθαρού κώδικα με περίπλοκα σχέδια. Η δυναμική επικύρωση λειτουργικότητας με οποιονδήποτε browser ελέγχει αυτόματα ετικέτες και τους κανόνες του CSS για σχεδιαστική συμβατότητα μεταξύ των πρωτοπόρων browsers, πράγμα που αποτελεί και τον πιο συχνό "πονοκέφαλο" των developers. Ένας ενσωματωμένος επεξεργαστής γραφικών που χρησιμοποιεί την τεχνολογία του Fireworks της Macromedia δίνει τη δυνατότητα στους χρήστες την συγκέντρωση, την επαναταξινόμηση και επεξεργασία των γραφικών δίχως να χρειαστεί να φύγετε από το περιβάλλον του Dreamweaver.

Η ανοικτή τεχνολογία του Dreamweaver επιτρέπει στους developers να δουλεύουν με τις τεχνολογίες της επιλογής τους. Το περιβάλλον ανάπτυξης ανεξαρτήτου πλατφόρμας υποστηρίζει τις πιο σημαντικές τεχνολογίες server, συμπεριλαμβανομένων των ColdFusion, ASP.NET, JSP και PHP. Το SecureFTP κρυπτογραφεί όλες τις μεταφορές αρχείων και αποτρέπει κάθε αναρμόδια πρόσβαση σε στοιχεία, περιεχόμενα αρχείων, ονόματα χρηστών και κωδικούς πρόσβασης. Η απρόσκοπτη ενσωμάτωση αρχείων των Word και Excel της Microsoft επιτρέπει στους χρήστες να συντηρήσουν τη μορφοποίηση, όπως το CSS, όταν κάνετε αποκοπή και επικόλληση στο Dreamweaver.

Το Dreamweaver 8 επίσης ενσωματώνει ένα βελτιωμένο περιβάλλον σχεδίου και ανάπτυξης για να βοηθήσει τους επαγγελματίες του web να βρουν ό,τι χρειάζονται γρηγορότερα, αυξάνοντας την απόδοση, την παραγωγικότητα, και την ποιότητα της εργασίας τους. Βελτιώσεις στα κύρια χαρακτηριστικά, όπως η βελτιωμένη επεξεργασία πίνακα και μια προηγμένη διαδικασία επεξεργασίας αρχείων, όλα αυτά αποτελούν στοιχεία που βοηθούν τους χρήστες να διαχειρίζονται καλύτερα τα βασικά. Οι χρήστες μπορούν να γράψουν κώδικα αποτελεσματικά με τα εργαλεία κωδικοποίησης δεξιού κλικ, βελτιωμένη εύρεση και αντικατάσταση καθώς και με έναν βελτιωμένο επιθεωρητή tags, που επιτρέπει την γρήγορη και περιεκτική επεξεργασία του αντικειμένου. Το Dreamweaver 8 περιλαμβάνει ακόμη και προηγμένη συνεργασία με το Contribute 2 της Macromedia, ένα νέο εργαλείο που επιτρέπει στους χρήστες του Dreamweaver να αναπροσαρμόσουν την κάθε ενημέρωση του περιεχόμενου του web προς ικανοποίηση των ιδιοκτητών χωρίς να εγκαταλείπουν τον έλεγχο της ιστοσελίδας τους.

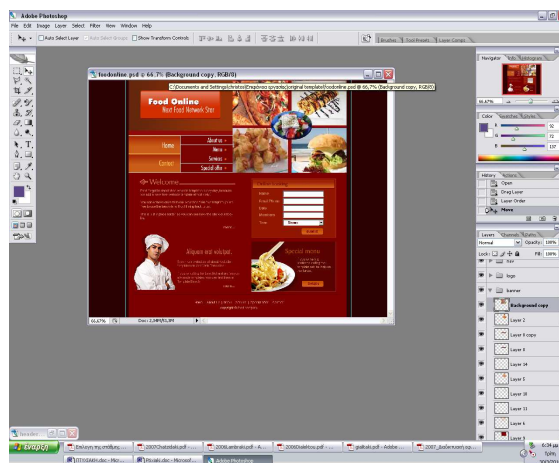
Το Dreamweaver 8 επίσης περιλαμβάνει τα Elements για HTML, για κατασκευή blocks εφαρμογών που επιτρέπουν στους developers να παράγουν γρήγορα εντυπωσιακά αποτελέσματα με συνεπές ύφος και αίσθηση. Τα Elements για HTML περιλαμβάνουν style sheets, περιεχόμενο δειγμάτων και templates. Τα style sheets αποτελούν εύκολα σημεία εκκίνησης για εξεζητημένη χρήση του CSS για προσδιορισμό θέσης με templates για κοινά περιγράμματα ιστοσελίδων.

## 4.2 Photoshop CS2

Το Photoshop υπήρξε ανέκαθεν το πρότυπο στην επεξεργασία ψηφιακής φωτογραφίας αλλά και στην δημιουργία layout για ιστοσελίδες (template). Μια πολύ χρήσιμη λειτουργία που παρέχει το Photoshop, είναι τα layers. Φανταστείτε τα layers σαν διαφορετικά επίπεδα το ένα πάνω από το άλλο τα οποία όλα μαζί δημιουργούν μια εικόνα. (γραφικά ή φωτογραφία). Το Photoshop πέρα από τις εκπληκτικές δυνατότητες που έχει στην επεξεργασία εικόνας, με τα χρόνια έχει εξελιχθεί αρκετά, προσφέροντας στους web designers ένα δυνατό εργαλείο για την σχεδίαση web sites.

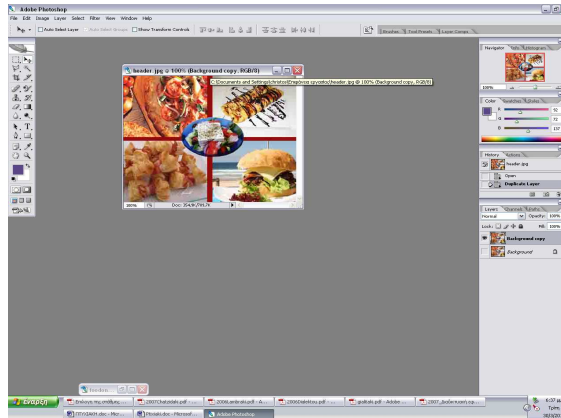
Αυτή τη στιγμή αποτελεί ηγέτη της αγοράς (*market leader*) των προγραμμάτων επεξεργασίας εικόνων, και είναι το προϊόν - σήμα κατατεθέν της Adobe Systems. Χαρακτηρίζεται ως "απαραίτητο" εργαλείο για τους επαγγελματίες γραφίστες και θεωρείται πως προώθησε τις αγορές των Macintosh, και στη συνέχεια των Windows.

Εμείς χρησιμοποιήσαμε το photoshop για να σχεδιάσουμε το logo του καταστήματος και για να κάνουμε resize τις εικόνες των προϊόντων.



Εικόνα 4-2 α. Η παραπάνω εικόνα μας δείχνει το template μας





Εικόνα 4-2 β. Το banner του καταστήματος

## 4.3 Γραφικά εργαλεία διαχείρισης βάσης

### 4.3.1 MySQL Query Browser

Ο MySQL Query Browser είναι ένα γραφικό εργαλείο, το οποίο παρέχεται από την MySQL AB, για την δημιουργία, εκτέλεση και βελτιστοποίηση αιτημάτων σε γραφικό περιβάλλον. Ο MySQL Query Browser έχει σχεδιαστεί για να σας βοηθήσει να θέτετε αιτήματα και να αναλύετε δεδομένα, τα οποία είναι αποθηκευμένα στην MySQL βάση δεδομένων σας.

Ενώ όλα τα αιτήματα, που μπορούν να εκτελεστούν με τον MySQL Query Browser, μπορούν επίσης να εκτελεστούν με την εφαρμογή γραμμής εντολών **mysql**, ο MySQL Query Browser επιτρέπει την υποβολή αιτημάτων και την επεξεργασία των δεδομένων με ένα πιο διαισθητικό, γραφικό τρόπο.

Ο MySQL Query Browser είναι σχεδιασμένος, ώστε να συνεργάζεται με τις εκδόσεις 4.0 και άνω της MySQL.

Ο MySQL Query Browser είναι σε μεγάλο βαθμό το αποτέλεσμα απόψεων και ανάδρασης, τα οποία έχει δεχθεί η MySQL AB, από πολλούς χρήστες σε μια περίοδο πολλών ετών.

### 4.3.2 Παράθυρο Διαλόγου Σύνδεσης

Μόλις ξεκινήσει ο MySQL Query Browser, εμφανίζεται ένα παράθυρο διαλόγου το οποίο αφορά την σύνδεση. Πρέπει να ορίσετε τον MySQL εξυπηρετητή,

με τον οποίο επιθυμείτε να συνδεθείτε, τα διαπιστευτήρια, τα οποία χρειάζονται για την πιστοποίησή σε αυτόν τον εξυπηρετητή, το μηχάνημα, στο οποίο τρέχει ο εξυπηρετητής (και σε ποια θύρα ακούει) και την προεπιλεγμένη (default) βάση δεδομένων (Σχήμα), στην οποία θα θέτετε αιτήματα.

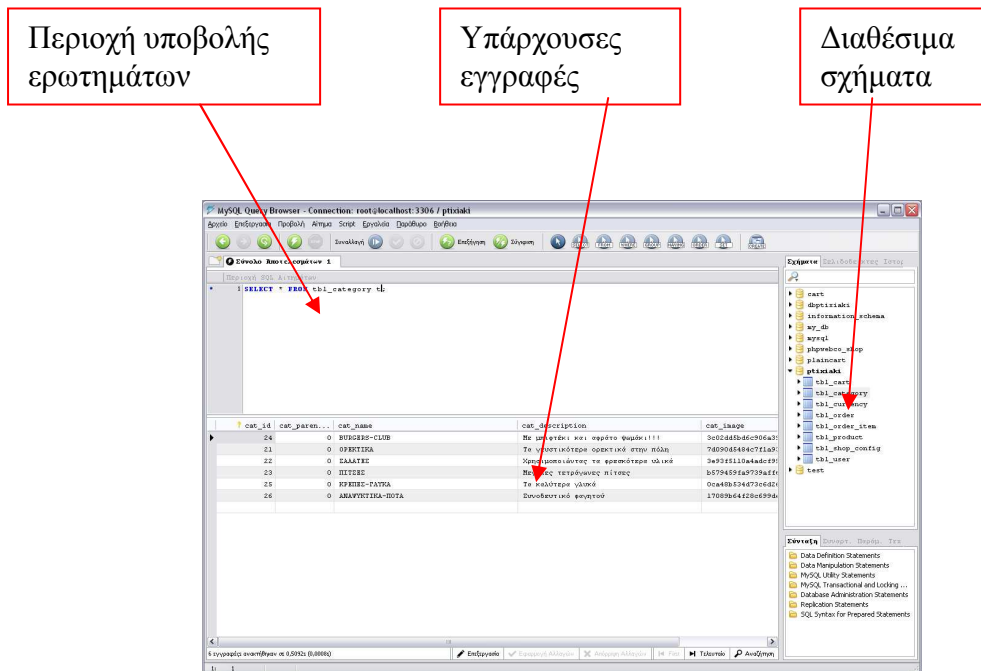


Εικόνα 4-3-2. Παράθυρο διαλόγου σύνδεσης

Αν η σύνδεση με τον εξυπηρετητή επιτευχθεί, όλες οι τιμές οι οποίες συμπληρώθηκαν στα πεδία του παραθύρου διαλόγου της σύνδεσης σώζονται για τις μελλοντικές συνδέσεις. Ωστόσο, το πεδίο Κωδικός είναι πάντα κενό, για λόγους ασφαλείας.

### 4.3.3 Περιήγηση στον MySQL Query Browser

Όταν συνδεθείτε επιτυχώς με έναν MySQL εξυπηρετητή, εμφανίζεται το κεντρικό παράθυρο αιτημάτων του MySQL Query Browser. Όλες οι λειτουργίες της εφαρμογής είναι προσβάσιμες από αυτό το παράθυρο.



Εικόνα 4-3-3. Το κεντρικό παράθυρο αιτημάτων του MySQL Query Browser

Το κεντρικό παράθυρο των αιτημάτων χωρίζεται σε διάφορους τομείς:

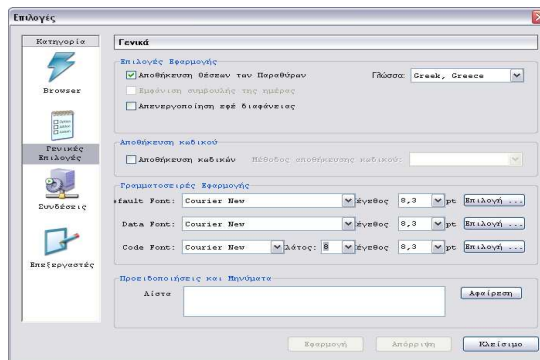
- **Η Γραμμή Εργαλείων των Αιτημάτων:** Η Γραμμή Εργαλείων των Αιτημάτων είναι το μέρος όπου δημιουργείτε και εκτελείτε τα αιτήματα.
- **Η Γραμμή Εργαλείων Ειδικών Επιλογών:** Η Γραμμή Εργαλείων Ειδικών Επιλογών περιέχει τρία σύνολα κουμπιών: τα κουμπιά Συναλλαγών (Start, Commit, Rollback), τα κουμπιά διαχείρισης των αιτημάτων (Επεξήγηση, Σύγκριση), και τα κουμπιά κατασκευής αιτημάτων (Select, From, Where, κτλ.)
- **Η Περιοχή των Αποτελεσμάτων:** Όλα τα αποτελέσματα των αιτημάτων εμφανίζονται στην Περιοχή των Αποτελεσμάτων. Πολλές καρτέλες μπορούν να είναι ενεργές ταυτόχρονα, δίνοντας σας έτσι την ευκαιρία να εργάζεστε σε πολλά αιτήματα
- **Το Ευρετήριο των Αντικειμένων:** Μπορείτε να επιλέξετε σε ποια βάση δεδομένων και πίνακα να θέσετε αιτήματα, να προσθέσετε τα αιτήματα που χρησιμοποιείτε συνήθως σε μια συλλογή από σελιδοδείκτες, να δείτε τα προηγούμενα αιτήματα, τα οποία είχατε θέσει, και να τα χρησιμοποιήσετε πάλι.

- **Το Ευρετήριο των Πληροφοριών:** χρησιμοποιείται για να βρίσκετε built-in functions και για την διαχείριση αιτημάτων, τα οποία περιέχουν παραμέτρους.

#### 4.3.4 Ρυθμίσεις

Χρησιμοποιήσαμε το MySQL Query Browser για την υποβολή ερωτημάτων διότι θα εισαγάγαμε ελληνικές εγγραφές στους πίνακες της βάσης. Αυτό δεν θα μπορούσαμε να το κάνουμε άμα χρησιμοποιούσαμε το MySQL Command Line Client, διότι δεν αναγνωρίζει τα ελληνικά.

Ακόμα και το MySQL Query Browser για να αναγνωρίσει τα ελληνικά πρέπει να εργαζόμαστε με γραμματοσειρά Courier New. Για να αλλάξουμε την γραμματοσειρά πατάμε το μενού Εργαλεία ->Επιλογές.



Εικόνα 4-3-4. Μενού αλλαγής γραμματοσειράς

#### 4.3.5 MySQL Administrator

Ο MySQL Administrator έχει σχεδιαστεί για την διαχείριση ενός MySQL εξυπηρετητή. MySQL Administrator είναι ένα ισχυρό visual administration κονσόλα για MySQL, η οποία επιτρέπει την εύκολη διαχείριση από τον χρήστη MySQL του περιβάλλον, και να αποκτήσει καλύτερη προβολή του τρόπου λειτουργίας της βάσης δεδομένων την οποία διαχειρίζεται. Είναι διαθέσιμο με αυτόχθονα GUI interfaces για τα Windows, Linux και Mac OS X.

### 4.3.6 Παράθυρο Διαλόγου Σύνδεσης

Μόλις ξεκινήσει ο MySQL Administrator, εμφανίζεται ένα παράθυρο διαλόγου το οποίο αφορά την σύνδεση. Πρέπει να ορίσετε τον MySQL εξυπηρετητή, με τον οποίο επιθυμείτε να συνδεθείτε, τα διαπιστευτήρια, τα οποία χρειάζονται για την πιστοποίησή σε αυτόν τον εξυπηρετητή, το μηχάνημα, στο οποίο τρέχει ο εξυπηρετητής (και σε ποια θύρα ακούει).

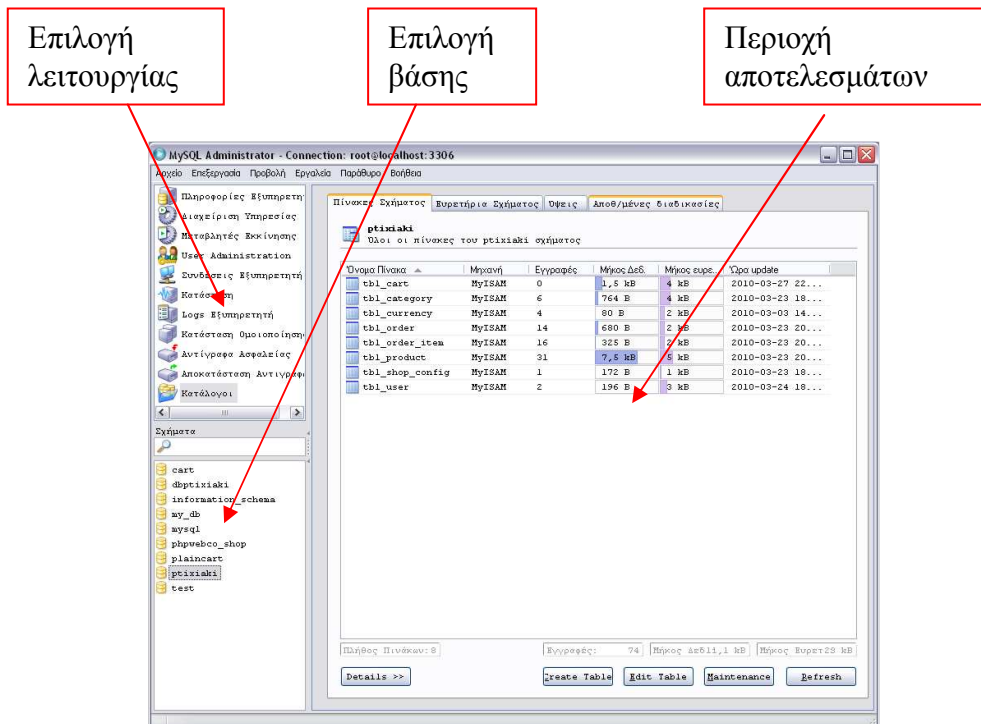


Εικόνα 4-3-6. Παράθυρο διαλόγου σύνδεσης

Αν η σύνδεση με τον εξυπηρετητή επιτευχθεί, όλες οι τιμές οι οποίες συμπληρώθηκαν στα πεδία του παραθύρου διαλόγου της σύνδεσης σώζονται για τις μελλοντικές συνδέσεις. Ωστόσο, το πεδίο Κωδικός είναι πάντα κενό, για λόγους ασφαλείας.

### 4.3.7 Περιήγηση στον MySQL Administrator

Όταν συνδεθείτε επιτυχώς με έναν MySQL εξυπηρετητή, εμφανίζεται το κεντρικό παράθυρο αιτημάτων του MySQL Administrator. Όλες οι λειτουργίες της εφαρμογής είναι προσβάσιμες από αυτό το παράθυρο.



Εικόνα 4-4-3. Το κεντρικό παράθυρο αιτημάτων του MySQL Administrator

Το κεντρικό παράθυρο των αιτημάτων χωρίζεται σε τρεις τομείς:

- **Η Γραμμή Λειτουργιών:** Σε αυτή υπάρχουν όλες οι λειτουργίες που μπορούν να εκτελεστούν από ένα MySQL εξυπηρετητή.
- **Το Ευρετήριο των Αντικειμένων:** Μπορείτε να επιλέξετε σε ποια βάση δεδομένων και πίνακα θα εργαστείτε.
- **Η Περιοχή των Αποτελεσμάτων:** Όλα τα αποτελέσματα των αιτημάτων εμφανίζονται στην Περιοχή των Αποτελεσμάτων. Είναι διαφορετική ανάλογα με την λειτουργία που εκτελούμε.

## 5. Αρχιτεκτονική εφαρμογής

### 5.1 Επίπεδα χρηστών

- Administrator

Οι χρήστες στην εφαρμογή μας χωρίζονται σε δύο επίπεδα. Στον administrator και στους απλούς χρήστες.

Ο administrator έχει όλα τα δικαιώματα. Μπορεί να προσθέτει, να επεξεργάζεται και να διαγράφει φαγητά. Επίσης μπορεί να προσθέτει, να

επεξεργάζεται και να διαγράφει κατηγορίες στις οποίες ανήκουν τα φαγητά. Επιπλέον έχει το δικαίωμα να προσθέτει χρήστες, να τους επεξεργάζεται, να τους διαγράφει , να βλέπει τις παραγγελίες και τέλος να επεξεργάζεται τις πληροφορίες καταστήματος (τηλέφωνο, e-mail). Τις παραγγελίες έχει την δυνατότητα είτε να τις δει και από το email του καταστήματος (gmail). Κάθε νέα παραγγελία που κάνει ένας χρήστης αυτομάτως στέλνεται και στο email. Είτε από το site του καταστήματος.

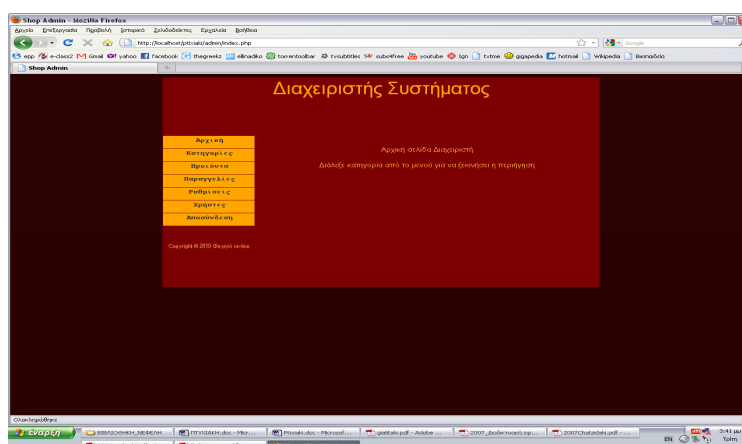
- Χρήστης

Ο χρήστης μπορεί μόνο να δει την περιγραφή των προϊόντων και τις τιμές χωρίς να μπορεί να αλλάξει κάτι. Μπορεί να κάνει παραγγελίες, να δει τις πληροφορίες καταστήματος και να στείλει μήνυμα στο κατάστημα μέσω της φόρμας επικοινωνίας.

## 5.2 Administrator

### 5.2.1 Λειτουργίες διαχειριστή

Η επιτυχής εισαγωγή στο σύστημα εμφανίζει την πρώτη σελίδα του περιβάλλοντος διαχείρισης του καταστήματος (εικόνα 5-2-1), όπου διακρίνονται δύο αλλαγές σε σχέση με τη σελίδα του πελάτη: Η πρώτη αλλαγή είναι η εμφάνιση στην επικεφαλίδα του καταστήματος της λέξης “ΔΙΑΧΕΙΡΙΣΤΗΣ ΣΥΣΤΗΜΑΤΟΣ”, που υπενθυμίζει ότι βρίσκεται στην περιοχή διαχείρισης και η δεύτερη αλλαγή είναι η εμφάνιση του μενού διαχείρισης στην αριστερή πλευρά.



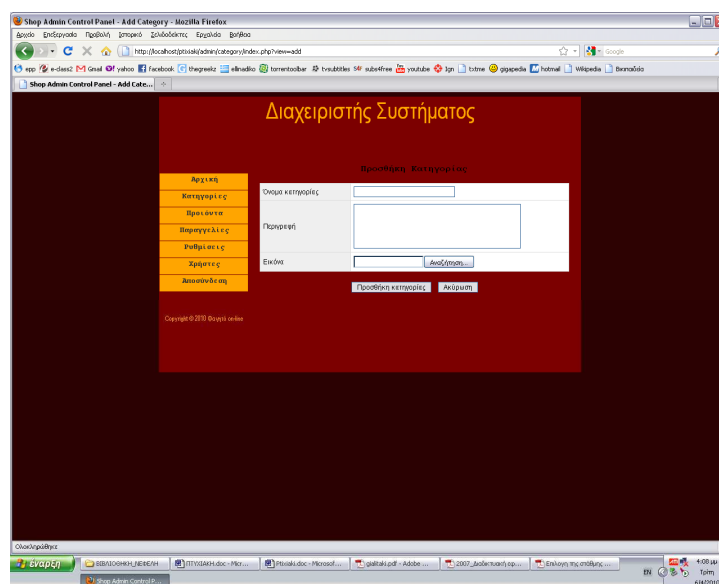
Εικόνα 5-2-1. Αρχικό μενού administrator

## 5.2.2 Προσθήκη κατηγορίας

Επιλέγοντας από το μενού «ΚΑΤΗΓΟΡΙΕΣ» εμφανίζονται όλες οι κατηγορίες του καταστήματος. Στη συνέχεια επιλέγοντας το κουμπί «Προσθήκη κατηγορίας» εμφανίζεται η παρακάτω φόρμα.



Εικόνα 5-2-2α Λίστα κατηγοριών



Εικόνα 5-2-2β. Προσθήκη κατηγορίας

Στη φόρμα αυτή ο διαχειριστής δίνει το όνομα της κατηγορίας, τη διαδρομή μιας εικόνας που αντιπροσωπεύει την συγκεκριμένη κατηγορία και μια περιγραφή της κατηγορίας που την βλέπει μόνο ο διαχειριστής.



Σημαντικό να πούμε ότι οι εικόνες που ανεβάζουμε είτε για προσθήκη κατηγορίας είτε για προσθήκη προϊόντος, πρέπει να έχουν πλάτος μέχρι 300 pixels αλλιώς δεν φορτώνονται καθόλου από το σύστημα. Αυτό το κάναμε έτσι ώστε να μην φορτωθεί πολύ μεγάλη εικόνα και χαλάσει το template μας.

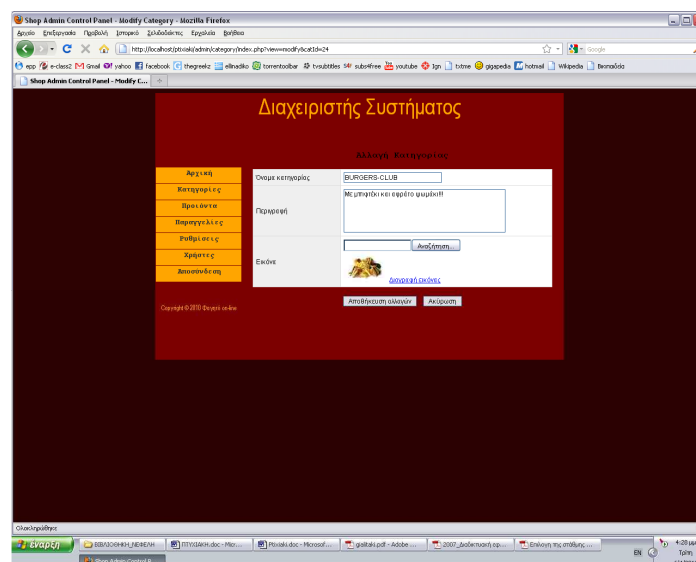
Παρακάτω παρατίθεται ο αντίστοιχος κώδικας php με σχολιασμό. Αυτός ο κώδικας υπάρχει στο αρχείο «admin/category/index».

```
<?php
//με την εντολή require_once καλούμε άλλα αρχεία
require_once '../library/config.php';
require_once '../library/functions.php';
//έλεγχος για τον σωστό χρήστη
$_SESSION['login_return_url'] = $_SERVER['REQUEST_URI'];
checkUser();
// συλλογή δεδομένων για τις κατηγορίες και τις λειτουργίες που έχει ο admin για τις
//κατηγορίες
$view = (isset($_GET['view']) && $_GET['view'] != "") ? $_GET['view'] : "";
switch ($view) {
    case 'list' :
        $content      = 'list.php';
        $pageTitle    = 'Shop Admin Control Panel - View Category';
        break;
    case 'add' :
        $content      = 'add.php';
        $pageTitle    = 'Shop Admin Control Panel - Add Category';
        break;
    case 'modify' :
        $content      = 'modify.php';
        $pageTitle    = 'Shop Admin Control Panel - Modify Category';
        break;
    default :
        $content      = 'list.php';
        $pageTitle    = 'Shop Admin Control Panel - View Category';
}
}
```

```
//εμφάνιση των προϊόντων σε ένα πίνακα από συνάρτηση που υπάρχει στο αρχείο
//category.js
$script = array('category.js');
require_once '../include/template.php';
?>
```

### 5.2.3 Επεξεργασία κατηγορίας

Επεξεργασία κατηγοριών απαιτείται στις περιπτώσεις που ο διαχειριστής επιθυμεί να αλλάξει το όνομα της κατηγορίας, την περιγραφή της, ή την εικόνα της. Παρακάτω φαίνεται η φόρμα επεξεργασίας κατηγοριών.



Εικόνα 5-2-3 Επεξεργασία κατηγορίας

Ο παρακάτω κώδικας βρίσκεται στο αρχείο «admin/category/modify». Βοηθάει στο να τροποποιήσουμε τα στοιχεία μιας κατηγορίας.

```
//έλεγχος για τον admin
if (!defined('WEB_ROOT')) {
    exit;
}
// Ελέγχουμε ότι η κατηγορία υπάρχει
if (isset($_GET['catId']) && (int)$_GET['catId'] > 0) {
    $catId = (int)$_GET['catId'];
} else {
    header('Location:index.php');
```

```

}
//πρωτού κάνουμε ένα ερώτημα στη βάση θα πρέπει να τις πούμε ότι τα δεδομένα τα
//θέλουμε με κωδικοποίηση utf8
mysql_query("SET NAMES 'utf8'", $dbConn);
//ερώτημα select προς την βάση
$sql = "SELECT cat_id, cat_name, cat_description, cat_image FROM tbl_category
WHERE cat_id = $catId";
//εισαγωγή των αποτελεσμάτων του ερωτήματος στη μεταβλητή result έτσι ώστε να
μπορούμε τα επεξεργαστούμε
$result = dbQuery($sql);
//εισαγωγή των αποτελεσμάτων σε γραμμές
$row = dbFetchAssoc($result);
//εμφάνιση των γραμμών
extract($row);
?>

```

#### 5.2.4 Διαγραφή κατηγορίας

Από το μενού «κατηγορίες» ο διαχειριστής μπορεί επίσης να διαγράψει μία κατηγορία. Διαγράφοντας μία κατηγορία, διαγράφονται και όλα τα προϊόντα που περιέχονται σε αυτή.

Η διαγραφή της κατηγορίας γίνεται μέσω μιας συνάρτησης javascript που καλείται από το αρχείο «admin/library/category.js». Παρακάτω είναι ο κώδικας

```

function deleteCategory(catId)
{
//επιβεβαίωση ότι θέλουμε να διαγράψουμε την κατηγορία
if (confirm('Διαγραφή της κατηγορίας θα διαγράψει και τα προϊόντα που
περιέχει.\nΣυνέχεια οπωσδήποτε?')) {
//διαγραφή της κατηγορίας
window.location.href = 'processCategory.php?action=delete&catId=' + catId;
}
}

```

## 5.2.5 Προσθήκη προϊόντος

Στο μενού επιλογών επιλέγοντας «προϊόντα» εμφανίζονται όλα τα προϊόντα. Επίσης δίνεται η δυνατότητα να δούμε τα προϊόντα ανά κατηγορίες κάνοντας τις κατάλληλες επιλογές από το drop down μενού που βρίσκεται στα δεξιά, όπως φαίνεται παρακάτω.(εικόνα 5-2-5α)

Επιπλέον πατώντας το κουμπί προσθήκη προϊόντος εμφανίζεται η φόρμα προσθήκης προϊόντος (εικόνα 5-2-5β).Στη φόρμα αυτή ο διαχειριστής προσθέτει το όνομα του προϊόντος, σε ποια κατηγορία ανήκει, περιγραφή, τιμή, εικόνα καθώς και σε τι ποσότητα είναι διαθέσιμο.



Εικόνα 5-2-5α Εμφάνιση προϊόντων ανά κατηγορία

Ο παρακάτω κώδικας εμφανίζει την λίστα των προϊόντων.

```
<?php
```

```
//έλεγχος για τον admin
```

```
if (!defined('WEB_ROOT')) {
```

```
    exit;
```

```
}
```

```
//ελέγχουμε αν υπάρχει η κατηγορία στην οποία ανήκουν τα προϊόντα
```

```
//και κάθε προϊόν να ανήκει σε μια συγκεκριμένη κατηγορία
```

```
if (isset($_GET['catId']) && (int)$_GET['catId'] > 0) {
```

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

```
    $sql2 = " AND p.cat_id = $catId";
```

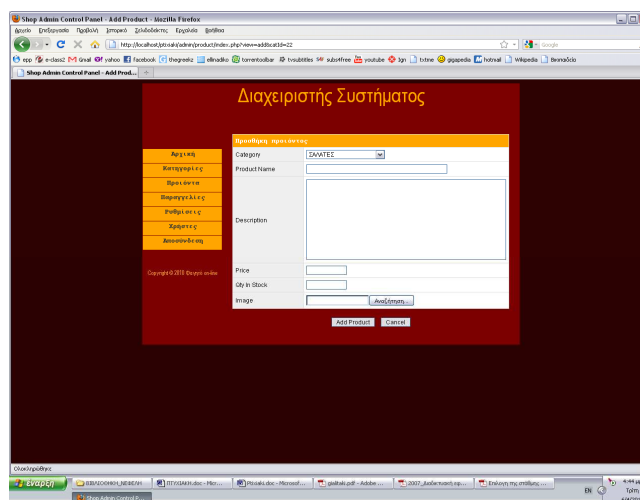
```
    $queryString = "catId=$catId";
```

```
} else {
```

```

$catId = 0;
$sql2 = "";
$queryString = "";
}
// για σελιδοποίηση. Πόσες γραμμές με προϊόντα θα υπάρχουν να σελίδα. Αν δεν
//χωράν σε μία τότε δημιουργούνται και άλλες. Αυτό γίνεται μέσω μιας συνάρτησης
στο αρχείο «admin/library/functions»
// πόσες γραμμές θα υπάρχουν ανα σελίδα
$rowsPerPage = 5;
//κωδικοποίηση που θέλουμε να έχουν οι εγγραφές της βάσης
mysql_query("SET NAMES 'utf8'", $dbConn);
$sql = "SELECT pd_id, c.cat_id, cat_name, pd_name, pd_thumbnail FROM
tbl_product p, tbl_category c WHERE p.cat_id = c.cat_id $sql2 ORDER BY
pd_name";
//εξαγωγή αποτελεσμάτων
$result = dbQuery(getPagingQuery($sql, $rowsPerPage));
//εισαγωγή των αποτελεσμάτων σε κάθε σελίδα
$pageLink = getPagingLink($sql, $rowsPerPage, $queryString);
//εμφάνιση των κατηγοριών
$categoryList = buildCategoryOptions($catId);
?>

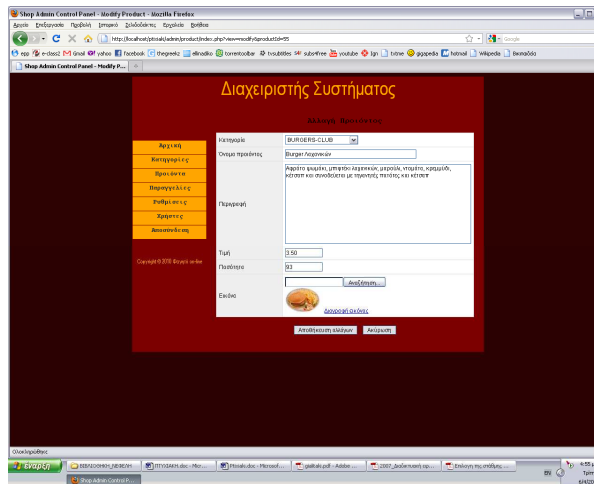
```



Εικόνα 5-2-5β Προσθήκη προϊόντος

## 5.2.6 Επεξεργασία προϊόντος

Επεξεργασία προϊόντος απαιτείται στις περιπτώσεις που ο διαχειριστής επιθυμεί να αλλάξει το όνομα ενός προϊόντος, την κατηγορία στην οποία ανήκει, την περιγραφή του, την τιμή του, το στοκ που είναι διαθέσιμο, ή την εικόνα του. Παρακάτω φαίνεται η φόρμα επεξεργασίας προϊόντων και ο κώδικας που χρειάστηκε.



Εικόνα 5-2-6 Επεξεργασία προϊόντος

```
<?php
```

```
//έλεγχος για τον admin
```

```
if (!defined('WEB_ROOT')) {
```

```
    exit;
```

```
}
```

```
// έλεγχος για το αν υπάρχει το προϊόν
```

```
if (isset($_GET['productId']) && $_GET['productId'] > 0) {
```

```
    $productId = $_GET['productId'];
```

```
} else {
```

```
    // redirect to index.php if product id is not present
```

```
    header('Location: index.php');
```

```
}
```

```
// ερώτημα προς την βάση για να πάρουμε τα στοιχεία του προϊόντος με την σωστή
```

```
//κωδικοποίηση
```

```
mysql_query("SET NAMES 'utf8'", $dbConn);
```

```

$sql = "SELECT pd.cat_id, pd_name, pd_description, pd_price, pd_qty, pd_image,
pd_thumbnail FROM tbl_product pd, tbl_category cat WHERE pd.pd_id =
$productId AND pd.cat_id = cat.cat_id";
$result = mysql_query($sql) or die('Cannot get product. ' . mysql_error());
$row = mysql_fetch_assoc($result);
extract($row);
// ερώτημα προς την βάση για να πάρουμε τα στοιχεία των κατηγοριών με την σωστή
//κωδικοποίηση
mysql_query("SET NAMES 'utf8'", $dbConn);
$sql = "SELECT cat_id, cat_parent_id, cat_name FROM tbl_category ORDER BY
cat_id";
$result = dbQuery($sql) or die('Cannot get Product. ' . mysql_error());
//δημιουργία λίστας μετα ονόματα των κατηγοριών
$categories = array();
while($row = dbFetchArray($result)) {
    list($id, $parentId, $name) = $row;

    if ($parentId == 0) {
        $categories[$id] = array('name' => $name, 'children' => array());
    } else {
        $categories[$parentId]['children'][] = array('id' => $id, 'name' =>
$name);
    }
}
// εισαγωγή αυτής της λίστας σε ένα combo box
$list = "";
foreach ($categories as $key => $value) {
    $name = $value['name'];
    $children = $value['children'];

    //$list .= "<optgroup label=\"\$name\">";
    $list .= "<option value=\"\$key\"";
        if ($key == $cat_id) {
            $list.= " selected";

```

```

    }
    $list .= ">$name</option>\r\n";

    foreach ($children as $child) {
        $list .= "<option value=\"{"$child['id']}\|\"";
        if ($child['id'] == $cat_id) {
            $list .= " selected";
        }
        $list .= ">{"$child['name']}</option>";
    }
    $list .= "</optgroup>";
}
?>

```

### 5.2.7 Διαγραφή προϊόντων

Από το μενού επιλογών «προϊόντα» μας δίνεται η δυνατότητα να διαγράψουμε ένα προϊόν. Πατώντας την επιλογή «διαγραφή» έχει σαν αποτέλεσμα να διαγράψουμε ένα προϊόν από το κατάστημά μας, καθώς και όλων των πληροφοριών που υπάρχουν για αυτό. Παρακάτω υπάρχει ο αντίστοιχος κώδικας

```

function deleteProduct(productId, catId)
{
    //επιβεβαίωση διαγραφής
    if (confirm('Διαγραφή προϊόντος?')) {
        //διαγραφή προϊόντος
        window.location.href = 'processProduct.php?action=deleteProduct&productId=' +
        productId + '&catId=' + catId;
    }
}

```

### 5.2.8 Παραγγελίες

Η επιλογή «Παραγγελίες» από το μενού εμφανίζει σε έναν πίνακα το σύνολο των παραγγελιών που έχουν γίνει από τους πελάτες, ταξινομημένες από την πιο πρόσφατη στην πιο παλιά.



Ο πίνακας με τις παραγγελίες έχει τέσσερις στήλες (εικόνα 5-2-8α). Η πρώτη αφορά τον κωδικό της παραγγελίας, η δεύτερη το ονοματεπώνυμο του πελάτη που την κατέθεσε, η τρίτη το συνολικό ποσό της παραγγελίας και η τέταρτη την ημερομηνία και ώρα καταχώρησης της παραγγελίας. Ο κωδικός της παραγγελίας είναι υπερσύνδεσμος που εμφανίζει όλες τις λεπτομέρειες της παραγγελίας (εικόνα 5-2-8β). Στη νέα σελίδα εμφανίζονται τα προϊόντα που αφορούν την παραγγελία, το πλήρες ιστορικό της και τα στοιχεία αποστολής της παραγγελίας.

Αριθμός Παραγγελίας	Όνομα πελάτη	Ποσό	Ημερομηνία και ώρα καταχώρησης
1020	Α.Α.	€10	2010-03-23 19:05:40
1029	Ανα Κια	€9	2010-03-23 17:53:47
1030	Σ.Κ.	€4	2010-03-23 17:41:28
1032	Χ.Κ.	€4	2010-03-23 17:38:33
1036	W.W.	€4	2010-03-23 17:12:56
1035	D.D.	€2	2010-03-23 17:10:43
1034	O.O.	€4	2010-03-23 17:09:38
1033	W.W.	€4	2010-03-23 17:07:44
1032	I.I.	€2	2010-03-23 16:45:28
1031	I.I.	€2	2010-03-23 15:34:22

Εικόνα 5-2-8α Λίστα παραγγελιών

Παραγγελία 1027		
Αριθμός παραγγελίας	1027	
Ημερομηνία παραγγελίας	2010-03-13 18:08:28	
Το συνολικό ποσό	2010-03-13 18:08:28	
Ποσών παραγγελίας		
Ποσό	Τιμή μονάδας	Σταθός
1 x 1000 Lene (σε σχήμα κροκέ)	€14	€14
	Σταθός	€14
Παραγγελία 1027 - Αναστασία		
Όνομα	Νίκος	
Επώνυμο	Παραδέντορας	
Διεύθυνση 1	Πικερδία Ιστορίας 5	
Τηλέφωνο	2330123456	
Παροχή	e-shop	

Εικόνα 5-2-8β Λεπτομέρειες παραγγελίας

Παρακάτω φαίνεται ο κώδικας που εμφανίζει την λίστα με τις παραγγελίες.

```
<?php
//αρχεία που χρειάζονται για τη εμφάνιση της λίστας
require_once '../library/config.php';
```

```

require_once '../library/database.php';
//έλεγχος για τον admin
if (!defined('WEB_ROOT')) {
    exit;
}
//έλεγχος για το αν υπάρχει η παραγγελία
if (isset($_GET['status']) && $_GET['status'] != "") {
    $status = $_GET['status'];
    $sql2 = " AND od_status = '$status'";
    $queryString = "&status=$status";
} else {
    $status = "";
    $sql2 = "";
    $queryString = "";
}
//σελιδοποίηση. Πόσες παραγγελίες θα εμφανίζονται ανά σελίδα
$rowsPerPage = 10;
mysql_query("SET NAMES 'utf8'", $dbConn);
//ερώτημα προς την βάση για να μας δώσει τα στοιχεία της παραγγελίας
$sql = "SELECT o.od_id, o.od_shipping_first_name, od_shipping_last_name,
od_date,SUM(pd_price * od_qty) AS od_amount FROM tbl_order o, tbl_order_item
oi, tbl_product p WHERE oi.pd_id = p.pd_id and o.od_id = oi.od_id $sql2 GROUP
BY od_id ORDER BY od_id DESC";
//εξαγωγή αποτελεσμάτων και σελιδοποίηση
$result = dbQuery(getPagingQuery($sql, $rowsPerPage));
$pagingLink = getPagingLink($sql, $rowsPerPage, $queryString);
?>

```

Ο παρακάτω κώδικας εμφανίζει τις λεπτομέρειες για κάθε παραγγελία.

```

<?php
//αρχεία που χρειάζονται για να εμφανιστούν οι λεπτομέρειες
require_once '../library/config.php';
require_once '../library/database.php';
if (!defined('WEB_ROOT')) {

```

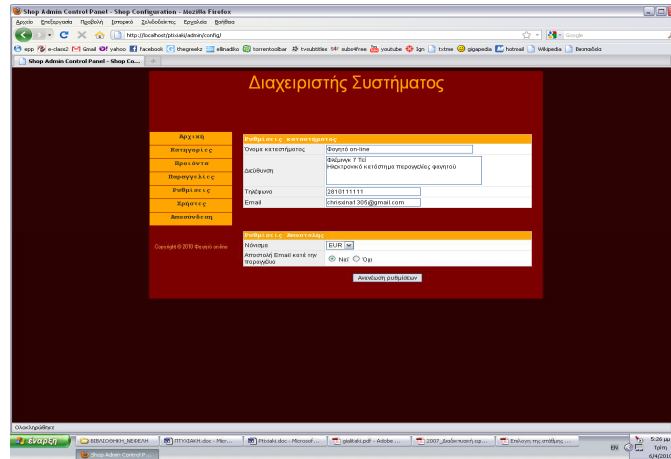
```

        exit;
    }
    //έλεγχος ότι υπάρχει η παραγγελία
    if (!isset($_GET['oid']) || (int)$_GET['oid'] <= 0) {
        header('Location: index.php');
    }
    $orderId = (int)$_GET['oid'];
    // ερώτημα για να πάρουμε τις πληροφορίες παραγγελίας
    mysql_query("SET NAMES 'utf8'", $dbConn);
    $sql = "SELECT pd_name, pd_price, od_qty FROM tbl_order_item oi, tbl_product p
    WHERE oi.pd_id = p.pd_id and oi.od_id = $orderId ORDER BY od_id ASC";
    //εξαγωγή αποτελεσμάτων παραγγελίας
    $result = dbQuery($sql);
    $orderedItem = array();
    //για την παραγγελία που εξάγαμε πριν κάνουμε ερώτημα για να πάρουμε το
    //αντίστοιχο προϊόν που ανήκει στην παραγγελία και τα στοιχεία του πελάτη
    while ($row = dbFetchAssoc($result)) {
        $orderedItem[] = $row;
    }
    // ερώτημα για τα στοιχεία του πελάτη
    mysql_query("SET NAMES 'utf8'", $dbConn);
    $sql = "SELECT od_date, od_last_update, od_status, od_shipping_first_name,
    od_shipping_last_name, od_shipping_address1, od_shipping_phone,
    od_shipping_state FROM tbl_order WHERE od_id = $orderId";
    ?>

```

### 5.2.9 Ρυθμίσεις καταστήματος

Σε αυτό το μενού ο διαχειριστής έχει την δυνατότητα να επεξεργαστεί πληροφορίες του καταστήματος, το νόμισμα που θα εμφανίζεται στην τιμή του προϊόντος καθώς και αν θα σταλεί email στο διαχειριστή κατά την ολοκλήρωση της παραγγελίας του πελάτη (εικόνα 5-2-9).



Εικόνα 5-2-9 Ρυθμίσεις καταστήματος

Παρακάτω φαίνεται ο κώδικας.

```
<?php
//έλεγχος για τον admin
if (!defined('WEB_ROOT')) {
    exit;
}
// ερώτημα για να πάρουμε τις υπάρχουσες εγγραφές
mysql_query("SET NAMES 'utf8'", $dbConn);
$sql = "SELECT sc_name, sc_address, sc_phone, sc_email, sc_currency,
sc_order_email FROM tbl_shop_config";
$result = dbQuery($sql);
//εξαγωγή αποτελεσμάτων. Έλεγχος για το αν επιστρέφει κενά
// make sure we query return a row
if (dbNumRows($result) > 0) {
    extract(dbFetchAssoc($result));
} else {
    // άμα δεν επιστρέφει σειρά τότε βάζουμε κενά
    // we just set blank values for all variables
    $sc_name = $sc_address = $sc_phone = $sc_email = $sc_currency = "";
    $sc_order_email = 'y';
}
// ερώτημα για να πάρουμε τα διαθέσιμα νομίσματα
mysql_query("SET NAMES 'utf8'", $dbConn);
```

```

$sql = "SELECT cy_id, cy_code FROM tbl_currency ORDER BY cy_code";
$result = dbQuery($sql);
//εισαγωγή των νομισμάτων σε drop down menu
$currency = "";
while ($row = dbFetchAssoc($result)) {
    extract($row);
    $currency .= "<option value=\"\$cy_id\"";
    if ($cy_id == $sc_currency) {
        $currency .= " selected";
    }
    $currency .= ">$cy_code</option>\r\n";
}
}??>

```

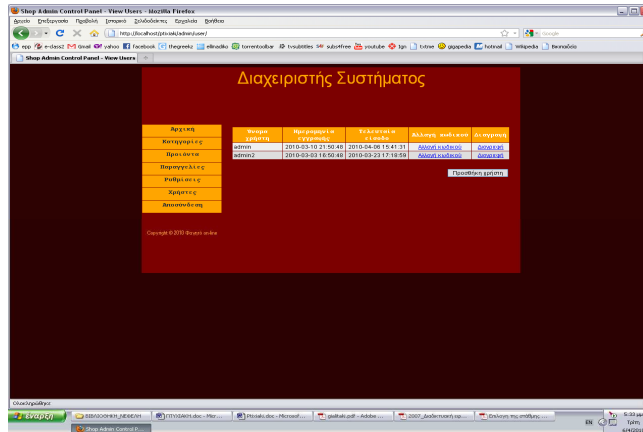
### 5.2.10 Διαχείριση χρηστών

Από το μενού επιλογών «χρήστες» ο διαχειριστής συστήματος μπορεί να εμφανίσει την λίστα με όλους τους διαχειριστές συστήματος (εικόνα 5-2-10α).

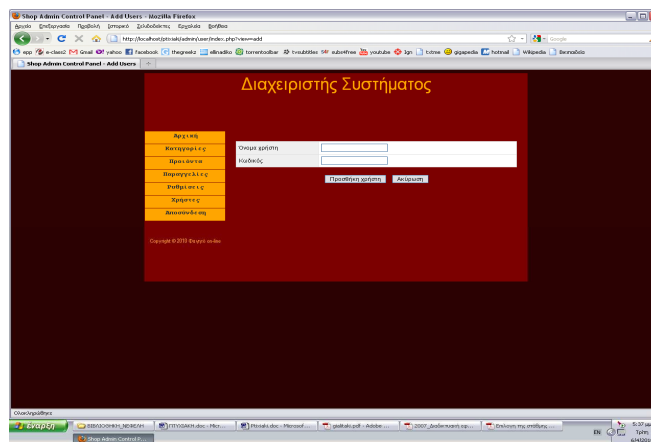
Από το κουμπί «προσθήκη χρήστη» ο διαχειριστής μπορεί να εισάγει ένα νέο διαχειριστή μέσω της φόρμας που εμφανίζεται (εικόνα 5-2-10β).

Επίσης έχει την δυνατότητα να αλλάζει τον κωδικό του άμα θέλει από την επιλογή «αλλαγή κωδικού». Σε αυτή την φόρμα που εμφανίζεται παρατηρούμε ότι δεν μπορούμε να αλλάξουμε το όνομα του χρήστη αλλά μόνο τον κωδικό. Για αυτό το λόγο είναι και συμπληρωμένο από πριν. Αυτό συμβαίνει γιατί ως εξουσιοδοτημένοι χρήστες υπάρχουν μόνο οι administrators (εικόνα 5-2-10γ).

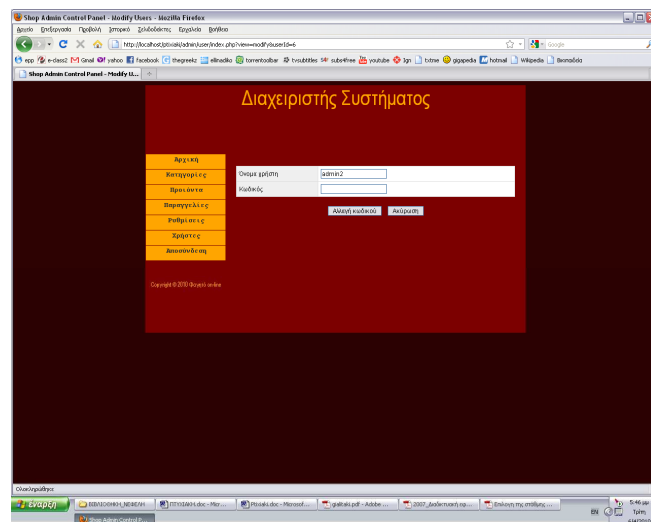
Τέλος, με την επιλογή διαγραφή μπορούμε να διαγράψουμε πλήρως τον χρήστη και δεν θα έχει πλέον δικαιώματα.



Εικόνα 5-2-10α Λίστα διαχειριστών συστήματος



Εικόνα 5-2-10β Προσθήκη νέου χρήστη



Εικόνα 5-2-10γ Αλλαγή κωδικού

Κώδικας που εμφανίζει όλους τους διαχειριστές

```
<?php
```

```
//ερώτημα για τσα στοιχεία του admin
```

```
$sql = "SELECT user_id, user_name, user_regdate, user_last_login FROM tbl_user
ORDER BY user_name";
```

```
//εξαγωγή στοιχείων
$result = dbQuery($sql);
?>
```

Κώδικας για την αλλαγή κωδικού.

```
<?php
//έλεγχος admin
if (!defined('WEB_ROOT')) {
    exit;
}
//έλεγχος για το αν υπάρχει όντως αυτός ο χρήστης
if (isset($_GET['userId']) && (int)$_GET['userId'] > 0) {
    $userId = (int)$_GET['userId'];
} else {
    header('Location: index.php');
}
//εμφάνιση μηνύματος λάθους
$errorMessage = (isset($_GET['error']) && $_GET['error'] != "") ? $_GET['error'] :
'&nbsp;';
//ερώτημα προς την βάση για τα στοιχεία του admin
$sql = "SELECT user_name FROM tbl_user WHERE user_id = $userId";
$result = dbQuery($sql);
extract(dbFetchAssoc($result));
function modifyUser()
{
    //αποθήκευση του user και του password σε μεταβλητές
    $userId = (int)$_POST['hidUserId'];
    $password = $_POST['txtPassword'];

    //αποθήκευση των νέων στοιχείων του admin στη βάση
    $sql = "UPDATE tbl_user SET user_password = sha1('$password') WHERE
user_id = $userId";
    dbQuery($sql);
    header('Location: index.php');
```

```
}  
?>
```

Κώδικας για προσθήκη Admin

```
function addUser()  
{  
    $userName = $_POST['txtUserName'];  
    $password = $_POST['txtPassword'];  
    // ο κωδικός πρέπει να έχει το λιγότερο 6 χαρακτήρες  
    // γράμματα και αριθμούς  
    if(strlen($password) < 6 || !preg_match('/[a-z]/i', $password) ||  
    !preg_match('/[0-9]/', $password)) {  
        //άκυρος κωδικός  
    }  
    // έλεγχος για το αν υπάρχει το username  
    $sql = "SELECT user_name FROM tbl_user WHERE user_name =  
    '$userName'";  
    $result = dbQuery($sql);  
    if (dbNumRows($result) == 1) {  
        header('Location: index.php?view=add&error=' . urlencode('Αυτό το  
        όνομα χρήστη χρησιμοποιείτε ήδη. Διάλεξε άλλο'));  
    } else {  
        //άμα είναι σωστό το όνομα και ο κωδικός τότε εισάγονται στη βάση  
        και επιστρέφουμε στη λίστα των διαχειριστών  
        $sql = "INSERT INTO tbl_user (user_name, user_password,  
        user_regdate) VALUES ('$userName', sha1('$password'), NOW())";  
        //$sql = "INSERT INTO tbl_user (user_name, user_password,  
        user_regdate) VALUES ('$userName', '$password', NOW())";  
        dbQuery($sql);  
        header('Location: index.php');  
    }  
}
```



### 5.2.11 Προστασία κωδικών πρόσβασης

Ένα σημαντικό σημείο που πρέπει να προσέξουμε είναι η αποθήκευση των κωδικών στη βάση. Όλα τα passwords αποθηκεύονται στη βάση με sha1 κωδικοποίηση. Με αυτή την κωδικοποίηση ο κωδικός πρόσβασης γίνεται μία συμβολοσειρά από δεκαεξαδικούς χαρακτήρες. Όταν ο administrator προσπαθεί να συνδεθεί στο site, ο κωδικός που γράφει κωδικοποιείται με την ίδια κωδικοποίηση και έτσι γίνεται η σύγκριση με τη βάση. Το ίδιο συμβαίνει για την προσθήκη νέου διαχειριστή και στην αλλαγή κωδικού. Όπως φαίνεται και παρακάτω:

```
$sql = "UPDATE tbl_user SET user_password = sha1('$password') WHERE user_id = $userId";
```

## 5.3 Περιγραφή της σελίδας του ηλεκτρονικού καταστήματος- Χρήστες

### 5.3.1 Δομή του ηλεκτρονικού κατάστημα

Η αρχική σελίδα του καταστήματος αποτελεί την σελίδα γνωριμίας με τους πελάτες και αποτελείται από τρεις περιοχές. Την επικεφαλίδα, η οποία περιέχει και τα links για περιήγηση στο υπόλοιπο κατάστημα, τον χώρο περιεχομένων, ο οποίος περιέχει και το login του διαχειριστή και το υποσέλιδο (εικόνα 5-3-1) και το υποσέλιδο.



Εικόνα 5-3-1. Αρχική σελίδα καταστήματος

Η επικεφαλίδα αποτελείται από τρία μέρη. Στο πάνω μέρος της υπάρχει το λογότυπο του καταστήματος και στα δεξιά ένα banner. Στο κάτω μέρος υπάρχουν links τα οποία βοηθούν για περιήγηση στο υπόλοιπο κατάστημα. Η επικεφαλίδα είναι πάντα ορατή καθ' όλη την περιήγηση.

Στο χώρο περιεχομένων υπάρχουν μηνύματα καλωσορίσματος και διαφημίσεις για το κατάστημα καθώς και την φόρμα login του διαχειριστή.

Στο υποσέλιδο υπάρχουν πληροφορίες επικοινωνίας με το κατάστημα, όπως τηλέφωνο, διεύθυνση και e-mail. Παρακάτω φαίνεται ο κώδικας «index» για την εμφάνιση της σελίδας index

```
<?php
//αρχείο που περιέχει τους κωδικούς τις βάσης και την διαδρομή του server
require_once 'library/config.php'
//αρχείο του χρειάζεται για το login του admin
require_once './admin/library/functions.php';
require_once 'logo.php';
$errorMessage = '&nbsp;';
//είσοδος του admin
if (isset($_POST['txtUserName'])) {
    $result = doLogin();
    if ($result != "") {
        $errorMessage = $result;
    }
}
?>
```

Κώδικας που υπάρχει στο αρχείο «admin/library/functions» για το login του admin.

```
function doLogin()
{
    // Άμα βρούμε λάθος το αποθηκεύουμε σε αυτή την μεταβλητή
    $errorMessage = "";
    //μεταβλητές με το όνομα και τον κωδικό του Admin
    $userName = $_POST['txtUserName'];
    $password = $_POST['txtPassword'];
    // έλεγχος ότι έχουν συμπληρωθεί και τα δύο πεδία. Username-password
```

```

if ($userName == "") {
    $errorMessage = 'Εισήγαγε όνομα χρήστη';
} else if ($password == "") {
    $errorMessage = 'Εισήγαγε κωδικό';
} else {
    // έλεγχος στη βάση για το αν υπάρχει ο συγκεκριμένος συνδυασμός
    //username-password
    $sql = "SELECT user_id FROM tbl_user WHERE user_name =
    '$userName' AND user_password = sha1('$password')";
    $result = dbQuery($sql);
    //Αφού επιβεβαιώθηκε ο ταυτότητα χρήστη ξεκινάμε νέο session
    if (dbNumRows($result) == 1) {
        $row = dbFetchAssoc($result);
        $_SESSION['plaincart_user_id'] = $row['user_id'];
        // αποθήκευση της ώρας που έκανε login ο admin
        $sql = "UPDATE tbl_user SET user_last_login = NOW()
        HERE user_id = '{$row['user_id']}'";
        dbQuery($sql);
        // Τώρα που ο χρήστης επιβεβαιώθηκε πάμε στη νέα σελίδα
        //Αν ο χρήστης είχε ξαναεπισκεφτεί τις admin σελίδες πάμε
        // στη τελευταία σελίδα που επισκέφτηκε
        if (isset($_SESSION['login_return_url'])) {
            header('Location: ' . $_SESSION['login_return_url']);
            exit;
        } else {
            //header('Location: index.php');
            header('Location: admin/index.php');
            exit;
        }
    } else {
        //μήνυμα λάθους άμα ο συνδυασμός username-password δεν
        //υπάρχει
        $errorMessage = 'Λάθος όνομα χρήστη ή κωδικός';
    }
}

```

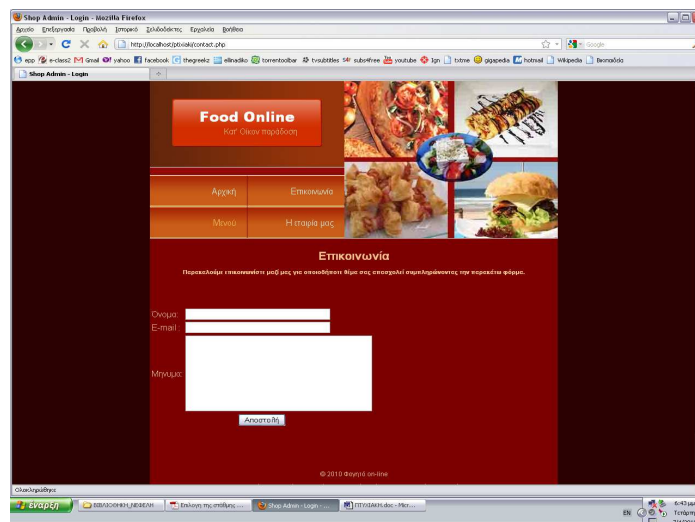
```

//εμφάνιση μηνύματος λάθους
return $errorMessage;
}

```

### 5.3.2 Επικοινωνία

Στο link «Επικοινωνία» ο χρήστης θα βρει μία φόρμα. Μέσω αυτής μπορεί να στείλει μήνυμα στο κατάστημα. Σημαντικό να πούμε ότι ο χρήστης θα πρέπει να συμπληρώσει σωστά τα στοιχεία του για να σταλεί το μήνυμα. Αυτό συμβαίνει γιατί γίνεται έλεγχος των περιεχομένων των πεδίων (εικόνα 5-3-2). Με το πάτημα του κουμπιού αποστολή το μήνυμα του χρήστη θα σταλεί στο email του καταστήματος (το gmail του υποσέλιδου). Παρακάτω φαίνεται ο κώδικας.



Εικόνα 5-3-2. Επικοινωνία

```
<?php
```

```
//αρχείο που ελέγχει την ορθότητα του email που εισάγαμε
```

```
require_once 'library/data_valid_fns.php';
```

```
//δημιουργία μεταβλητών
```

```
$name=$_POST['namee'];
```

```
$email=$_POST['email'];
```

```
$feedback=$_POST['feedback'];
```

```
//$eisaimelos=$_POST['selectmelos'];
```

```
//μήνυμα λάθους άμα δεν έχουν συμπληρωθεί όλα τα πεδία της φόρμας
```

```
try
```

```

{
    // check forms filled in
    if (!filled_out($_POST))
    {
        throw new Exception('Δεν έχετε συμπληρώσει σωστά την φόρμα - παρακαλώ
πηγαίνετε πίσω'.' και προσπαθήστε ξανά');
    }
    // μήνυμα λάθους άμα το email δεν είναι σωστό
    if (!valid_email($email))
    {
        throw new Exception('Το e-mail που συμπληρώσατε δεν είναι σωστό. Παρακαλώ
πηγαίνετε πίσω'.' και προσπαθήστε ξανά');
    }
    // μήνυμα λάθους άμα έχει λιγότερο από 6 χαρακτήρες
    if (!valid_email_2($email))
    {
        throw new Exception('Το e-mail που συμπληρώσατε δεν είναι σωστό. Παρακαλώ
πηγαίνετε πίσω'.' και προσπαθήστε ξανά');
    }
}
catch (Exception $e)
{
    echo $e->getMessage();
    echo '<p>&nbsp;</p>';
    echo '<p>&nbsp;</p>';
    require_once 'include/footer.php';
    exit;
}
//σε ποιο email θα σταλεί το μήνυμα
$toaddress = 'chrisxina1305@gmail.com';
//θέμα μηνύματος
$subject = 'Επιβεβαίωση αποστολής';
//μήνυμα
$mailcontent = "Όνομα πελάτη: ".$name."\n"

```

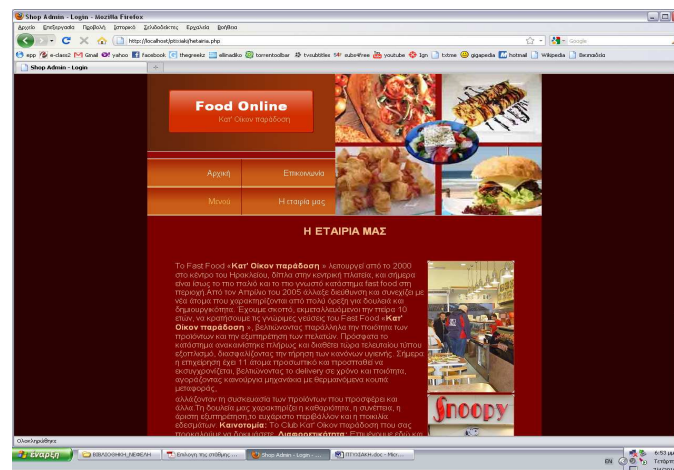
```

'E-mail: '$email.'"
//.Είσαι μέλος; '$eisaimelos.'"
."Μήνυμα πελάτη: \n".$feedback.'"";
//από ποιο email θα σταλεί το μήνυμα
$fromaddress = 'From: chrisxinal@yahoo.gr';
ini_set ('sendmail_from', 'chrisxinal@yahoo.gr');
//ρυθμίσεις στο αρχείο php.ini για να σταλεί σωστά το email
//ini_set ("SMTP", "mailgate.forthnet.gr");
//αποστολή μηνύματος
mail($toaddress, $subject, $mailcontent, $fromaddress);
?>

```

### 5.3.3 Η εταιρία μας

Αυτό το link περιέχει εν συντομία την ιστορία του καταστήματος από τότε που άνοιξε μέχρι σήμερα. Πόσα άτομα εργάζονται σε αυτό, με τι εξοπλισμό δουλεύει καθώς και μερικές φωτογραφίες του καταστήματος (εικόνα 5-3-3).

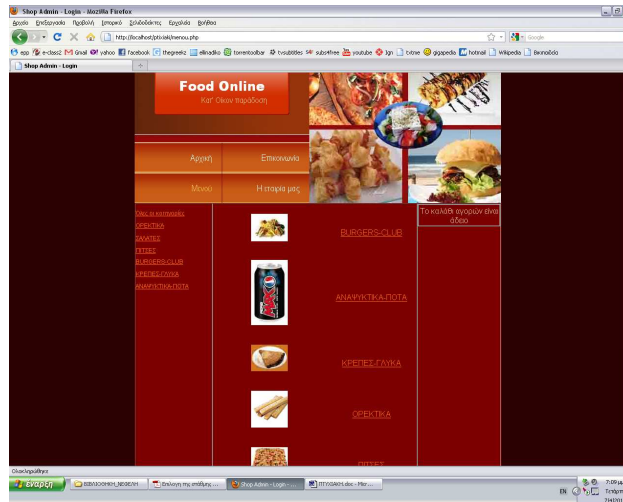


Εικόνα 5-3-3. Η εταιρία μας

### 5.3.4 Περιήγηση στα προϊόντα

Στο link μενού βρίσκουμε τον κατάλογο προϊόντων του καταστήματος. Τα προϊόντα είναι ομαδοποιημένα σε κατηγορίες. Ο κατάλογος χωρίζεται σε τρεις περιοχές. Η πρώτη μας δείχνει μια λίστα με όλες τις κατηγορίες. Η δεύτερη μας

δείχνει το εκάστοτε επιλεγμένο προϊόν και η τρίτη μας δείχνει το καλάθι αγορών. Οι κατηγορίες είναι ορεκτικά, σαλάτες, πίτσες, burgers-club, κρέπες-γλυκά, ποτά-αναψυκτικά. Ως αρχική σελίδα του μενού βλέπουμε όλες τις κατηγορίες (εικόνα 5-3-4). Παρακάτω φαίνεται και ο κώδικας



Εικόνα 5-3-4. Λίστα κατηγοριών προϊόντων

```
<?php
```

```
//πόσα προϊόντα θα φαίνονται ανά γραμμή
```

```
$productsPerRow = 1;
```

```
//πόσα προϊόντα θα φαίνονται ανά σελίδα
```

```
$productsPerPage = 4;
```

```
//δημιουργία πίνακα για τα προϊόντα
```

```
$children = array_merge(array($catId), getChildCategories(NULL, $catId));
```

```
$children = '(' . implode(', ', $children) . ')';
```

```
//εμφάνιση προϊόντων κατηγορίας
```

```
$sql = "SELECT pd_id, pd_name, pd_price, pd_thumbnail, pd_qty, c.cat_id
```

```
FROM tbl_product pd, tbl_category c WHERE pd.cat_id = c.cat_id AND pd.cat_id  
IN $children ORDER BY pd_name";
```

```
$result = dbQuery(getPagingQuery($sql, $productsPerPage));
```

```
$pagingLink = getPagingLink($sql, $productsPerPage, "c=$catId");
```

```
$numProduct = dbNumRows($result);
```

```
// Οι εικόνες των προϊόντων εμφανίζονται σε πίνακα.. Για να σιγουρευτούμε ότι
```

```
// έχει τον αντίστοιχο χώρο ορίζουμε το πλάτος κελιού του πίνακα
```

```

$columnWidth = (int)(100 / $productsPerRow);
//αν υπάρχουν προϊόντα σε μια κατηγορία τα εμφανίζουμε
if ($numProduct > 0 ) {
    $i = 0;
    while ($row = dbFetchAssoc($result)) {
        extract($row);
        if ($pd_thumbnail) {
            //άμα έχει εικόνα το προϊόν την εμφανίζουμε
            $pd_thumbnail = WEB_ROOT . 'images/product/' .
$pd_thumbnail;
        } else {
            //άμα δεν έχει εικόνα εμφανίζουμε μια εικόνα no-image
            $pd_thumbnail = WEB_ROOT . 'images/no-image-small.png';
        }
        //αλλαγή γραμμής
        if ($i % $productsPerRow == 0) {
            echo '<tr>';
        }
        // μορφή που θα εμφανίσουμε την τιμή
        $pd_price = displayAmount($pd_price);
        //εμφάνιση τιμής
        echo "<td width=\"\$columnWidth%\" align=\"center\"><img
src=\"\$pd_thumbnail\" border=\"0\"><br>$pd_name</a><br>Price : $pd_price";
        //μεγέθυνση εικόνας
        echo "<td width=\"\$columnWidth%\" align=\"center\"><a href=\"\" .
$_SERVER['PHP_SELF'] . "?c=$catId&p=$pd_id" . "\"><br>Μεγέθυνση εικόνας-
Λεπτομέρειες</a><br>";
        // Αν έχει τελειώσει το προϊόν εμφανίζεται τέλος αποθέματος
        if ($pd_qty <= 0) {
            echo "<br>Τέλος αποθέματος";
        }
        echo "</td>\r\n";
        if ($i % $productsPerRow == $productsPerRow - 1) {
            echo '</tr>';
        }
    }
}

```



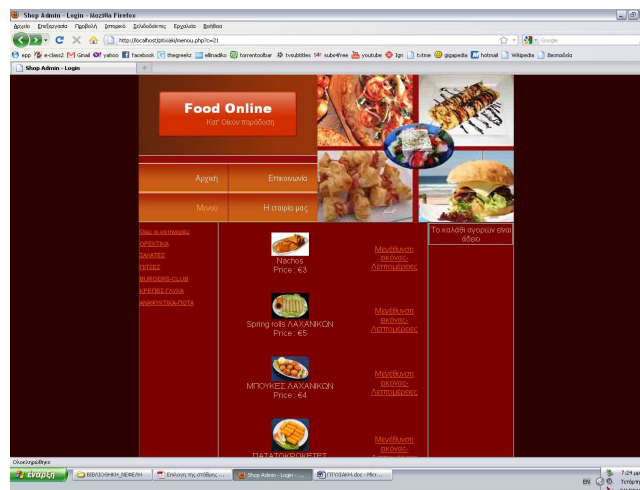
```

    }
    $i += 1;
}
?>

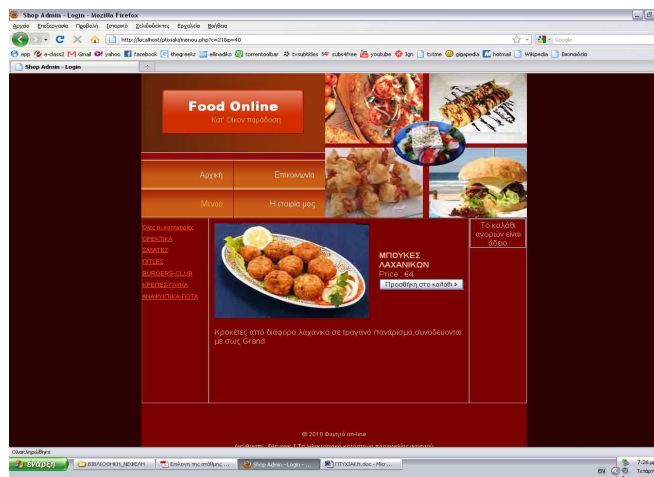
```

### 5.3.5 Προβολή λεπτομερειών προϊόντος

Επιλέγοντας μια συγκεκριμένη κατηγορία, εμφανίζεται μία λίστα των προϊόντων που εμπεριέχονται μέσα σε αυτή. Αυτή η λίστα μας δείχνει την εικόνα του προϊόντος, την τιμή του και ένα σύνδεσμο για εμφάνιση των λεπτομερειών του προϊόντος (εικόνα 5-3-5α). Επιλέγοντας το σύνδεσμο λεπτομέρειες βλέπουμε επιπλέον μια μεγέθυνση της εικόνας του προϊόντος, περιγραφή του προϊόντος και το κουμπί «προσθήκη στο καλάθι» (εικόνα 5-3-5β).



Εικόνα 5-3-5α Λίστα προϊόντων συγκεκριμένης κατηγορίας



Εικόνα 5-3-5β Προβολή λεπτομερειών προϊόντος

Παρακάτω υπάρχει ο κώδικας που εμφανίζει τις λεπτομέρειες των προϊόντων.

```
<?php
//συνάρτηση που βρίσκει τις λεπτομέρειες προϊόντος
function getProductDetail($pdId, $catId)
{
    //το session του server να είναι ίσο με το session του καταστήματος
    $_SESSION['shoppingReturnUrl'] = $_SERVER['REQUEST_URI'];
    // παίρνουμε τις πληροφορίες του προϊόντος από την βάση
    $sql = "SELECT pd_name, pd_description, pd_price, pd_image, pd_qty
FROM tbl_product WHERE pd_id = $pdId";
    //εξαγωγή αποτελεσμάτων
    $result = dbQuery($sql);
    $row = dbFetchAssoc($result);
    extract($row);
    //εξαγωγή της περιγραφής
    $row['pd_description'] = nl2br($row['pd_description']);
    //εξαγωγή της εικόνας
    if ($row['pd_image']) {
        $row['pd_image'] = WEB_ROOT . 'images/product/' .
$row['pd_image'];
    } else {
        $row['pd_image'] = WEB_ROOT . 'images/no-image-large.png';
    }
    $row['cart_url'] = "cart.php?action=add&p=$pdId";
    return $row;
}
//κλήση της συνάρτησης που βρίσκει τις λεπτομέρειες προϊόντος
$product = getProductDetail($pdId, $catId);
<?php
// άμα έχουμε απόθεμα εμφανίζεται κουμπί «προσθήκη στο καλάθι»
// show the 'Add to cart' button
if ($pd_qty > 0) {
?>
```

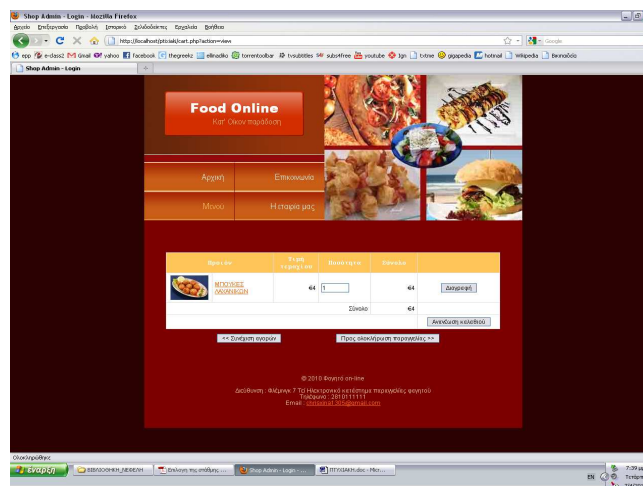
```

<input type="button" name="btnAddToCart" value="Προσθήκη στο καλάθι &gt;"
onClick="window.location.href='<?php echo $cart_url; ?>';"
class="addToCartButton">
<?php
} else {
    echo 'Τέλος Αποθεμάτων';
}
?>

```

### 5.3.6 Προβολή καλαθιού αγορών

Για την προβολή του καλαθιού αγορών αρκεί ο χρήστης να επιλέξει το link «Μετάβαση στο καλάθι» που υπάρχει στα δεξιά όταν είναι γεμάτο το καλάθι. Η εμφάνιση του καλαθιού είναι η παρακάτω εικόνα. Στο καλάθι μπορούμε να δούμε 6 στήλες. Το προϊόν που θα παραγγείλουμε και μια εικόνα του, την τιμή ανά προϊόν, την ποσότητα των προϊόντων, το συνολικό ποσό παραγγελίας και τα κουμπιά διαγραφή-ανανέωση καλαθιού. Επίσης μπορούμε να αλλάξουμε την ποσότητα του προϊόντος που θέλουμε να αγοράσουμε. Άμα αλλάξουμε την ποσότητα θα πρέπει να κάνουμε «ανανέωση καλαθιού» έτσι ώστε να ενημερωθούν οι τιμές. Επίσης μπορούμε να διαγράψουμε ένα προϊόν από το καλάθι. Τέλος μπορούμε μέσω του κουμπιού «συνέχιση αγορών» να ξαναπάμε στο κατάλογο προϊόντων και μέσω του κουμπιού «προς ολοκλήρωση παραγγελίας» να δούμε την φόρμα συμπλήρωσης στοιχείων του πελάτη. Για την προβολή του καλαθιού χρησιμοποιούμε διάφορες συναρτήσεις που βρίσκονται στο αρχείο «library/cart-functions»



Εικόνα 5-3-6 Προβολή καλαθιού

```

//Για το τρέχον session προσθέτει όλα τα προϊόντα στο καλάθι που έχουν επιλεγεί
function getCartContent()
{
    //δημιουργία πίνακα για το καλάθι
    $cartContent = array();
    //για το τρέχον session και τα προϊόντα υπάρχουν, βρίσκουμε τα στοιχεία
    //έτσι ώστε να προστεθούν στο καλάθι
    $sid = session_id();
    $sql = "SELECT ct_id, ct.pd_id, ct_qty, pd_name, pd_price, pd_thumbnail,
pd.cat_id FROM tbl_cart ct, tbl_product pd, tbl_category cat WHERE
ct_session_id = '$sid' AND ct.pd_id = pd.pd_id AND cat.cat_id = pd.cat_id";

    $result = dbQuery($sql);
    //εμφάνιση εικόνα προϊόντος
    while ($row = dbFetchAssoc($result)) {
        if ($row['pd_thumbnail']) {
            $row['pd_thumbnail'] = WEB_ROOT . 'images/product' .
            $row['pd_thumbnail'];
        } else {
            $row['pd_thumbnail'] = WEB_ROOT . 'images/no-image-
            small.png';
        }
        $cartContent[] = $row;
    }
    //εξαγωγή των περιεχομένων του καλαθιού
    return $cartContent;
}

```

```

//συνάρτηση διαγραφής από το καλάθι
function deleteFromCart($cartId = 0)
{
    //έλεγχος αν υπάρχει το προϊόν που έχει προστεθεί στο καλαθι
    if (!$cartId && isset($_GET['cid']) && (int)$_GET['cid'] > 0) {
        $cartId = (int)$_GET['cid'];
    }
}

```

```

}
if ($cartId) {
    //διαγραφή από το καλάθι
    $sql = "DELETE FROM tbl_cart WHERE ct_id = $cartId";
    $result = dbQuery($sql);
}
header('Location: cart.php');
}

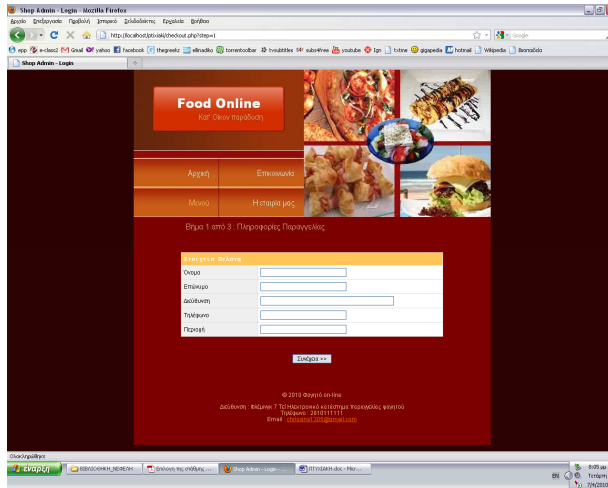
```

### 5.3.7 Πληροφορίες παραγγελίας

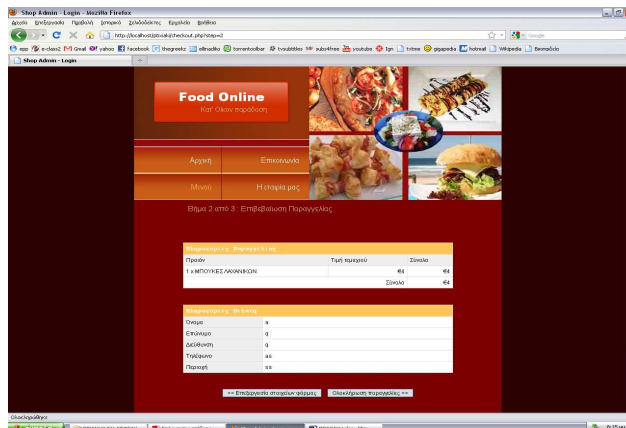
Όταν ο χρήστης βρίσκεται στο καλάθι και επιλέξει το κουμπί «προς ολοκλήρωση παραγγελίας», μεταβαίνουμε στη φόρμα «πληροφορίες παραγγελίας». Σε αυτή την φόρμα ο χρήστης πρέπει να δώσει τα στοιχεία του έτσι ώστε το κατάστημα να ξέρει πού να παραδώσει την παραγγελία. Όλα τα πεδία πρέπει να συμπληρωθούν υποχρεωτικά αλλιώς ο χρήστης δεν μπορεί να ολοκληρώσει την παραγγελία. Τα πεδία είναι το όνομα, το επώνυμο, η διεύθυνση, το τηλέφωνο και η περιοχή (εικόνα 5-3-7α).

Αφού συμπληρωθεί η φόρμα στοιχείων και πατήσουμε συνέχεια εμφανίζονται δύο πίνακες επιβεβαίωσης με όλες τις πληροφορίες τις παραγγελίας. Ο πρώτος μας δείχνει όλες τις πληροφορίες σχετικά με το προϊόν που παραγγείλαμε και ο δεύτερος μας δείχνει όλες τις πληροφορίες σχετικά με τον πελάτη. Η εμφάνιση των δύο πινάκων γίνεται για επαλήθευση. Άμα τα στοιχεία είναι λάθος τότε πατώντας το κουμπί «επεξεργασία στοιχείων φόρμας» επιστρέφουμε στην φόρμα στοιχείων του πελάτη και την ξανασυμπληρώνουμε (εικόνα 5-3-7β).

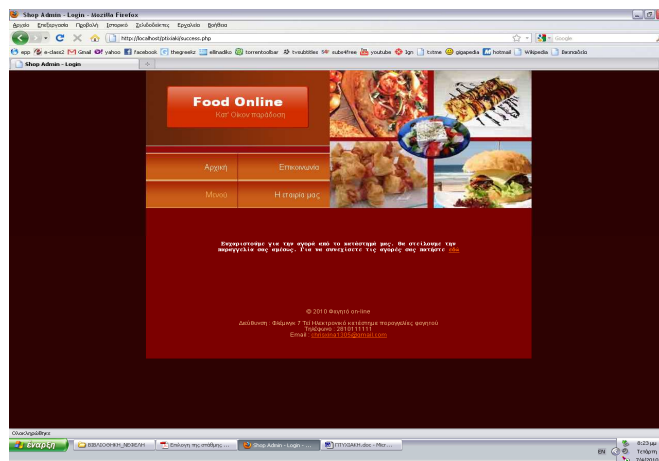
Τέλος άμα τα στοιχεία είναι σωστά, πατώντας «προς ολοκλήρωση παραγγελίας», μεταβαίνουμε σε μία σελίδα που μας επιβεβαιώνει ότι η παραγγελία έγινε με επιτυχία. Η παραγγελία στέλνεται στο email του καταστήματος (εικόνα 5-3-7γ).



Εικόνα 5-3-7α. Πληροφορίες παραγγελίας



Εικόνα 5-3-7β Πίνακες επιβεβαίωσης παραγγελίας



Εικόνα 5-3-7γ Επιβεβαίωση επιτυχούς παραγγελίας

Κώδικας προσθήκης στοιχείων του πελάτη

<?php

//πίνακας που μπαίνουν τα στοιχεία

```
$requiredField = array('txtShippingFirstName', 'txtShippingLastName',  
'txtShippingAddress1', 'txtShippingPhone', 'txtShippingState');
```

```
//αν δεν έχει συμπληρωθεί σωστά η φόρμα εμφανίζεται μήνυμα λάθους
```

```
if (!checkRequiredPost($requiredField)) {
```

```
    $errorMessage = 'Δεν έχετε συμπληρώσει όλα τα στοιχεία της φόρμας';
```

```
}
```

```
?>
```

```
//Ο έλεγχος για το αν συμπληρώθηκε η φόρμα σωστά γίνεται μέσω javascript
```

```
function checkShippingAndPaymentInfo()
```

```
{
```

```
    with (window.document.frmCheckout) {
```

```
        if (isEmpty(txtShippingFirstName, 'Εισήγαγε όνομα')) {
```

```
            return false;
```

```
        } else if (isEmpty(txtShippingLastName, 'Εισήγαγε επώνυμο')) {
```

```
            return false;
```

```
        } else if (isEmpty(txtShippingAddress1, 'Εισήγαγε διεύθυνση')) {
```

```
            return false;
```

```
        } else if (isEmpty(txtShippingPhone, 'Εισήγαγε τηλέφωνο')) {
```

```
            return false;
```

```
        } else if (isEmpty(txtShippingState, 'Εισήγαγε περιοχή')) {
```

```
            return false;
```

```
        }
```

```
    }
```

```
}
```

## 6.Βάση Δεδομένων

### 6.1 Αρχιτεκτονική της Βάσης

Όπως κάθε δυναμική ιστοσελίδα, η ιστοσελίδα του καταστήματος απαιτεί μια τουλάχιστον βάση δεδομένων. Ο σκοπός της βάσης είναι η φύλαξη δεδομένων και πληροφοριών που απαιτούνται για την λειτουργία της ιστοσελίδας. Στη συγκεκριμένη περίπτωση έχει κατασκευαστεί μία (1) βάση δεδομένων με το όνομα «rtixiaki» όπου αποθηκεύονται όλες οι πληροφορίες του καταστήματος, ομαδοποιημένες σε πίνακες (tables) ανάλογα με το είδος και τη χρήση τους. Στον επόμενο πίνακα (6-1) φαίνονται τα ονόματα των πινάκων της βάσης δεδομένων και η χρήση τους.

| Πίνακας(table)  | Περιγραφή   |
|-----------------|---|
| tbl_cart        | Πληροφορίες σχετικά με το καλάθι αγορών. Όπως τι προϊόν είναι μέσα και σε ποια κατηγορία ανήκει. Τι ποσότητα είναι η εκάστοτε παραγγελία και την ημερομηνία |
| tbl_category    | Περιέχει τις κατηγορίες προϊόντων, μια περιγραφή της, που την βλέπει μόνο ο admin για να ξέρει τι είναι.  |
| tbl_currency    | Περιέχει τον κωδικό και το σύμβολο για διαφορετικά νομίσματα. Σε αυτόν έχει πρόσβαση μόνο ο admin   |
| tbl_order       | Περιέχει τον κωδικό της παραγγελίας, την ημερομηνία καθώς και τα στοιχεία του πελάτη  |
| tbl_order_item  | Περιέχει το κωδικό της παραγγελίας, ποσότητα και προϊόν   |
| tbl_product     | Περιέχει όλες τις πληροφορίες για τα προϊόντα π.χ. κατηγορία που ανήκει, τιμή, περιγραφή, εικόνα  |
| tbl_shop_config | Πληροφορίες σχετικά με το κατάστημα, όπως τηλέφωνο,email, διεύθυνση   |
| tbl_user        | Πληροφορίες για τους εξουσιοδοτημένους χρήστες  |

Αναλυτικά κάθε πίνακας περιέχει τα εξής πεδία:

**tbl\_cart:** ct\_id, pd\_id, ct\_session\_id, ct\_date

**tbl\_category:** cat\_id, cat\_parent\_id, cat\_name, cat\_description, cat\_image

**tbl\_currency:** cy\_id, cy\_code, cy\_symbol

**tbl\_order:**

od\_id, od\_date, od\_last\_update, od\_status,

od\_shipping\_first\_name,

od\_shipping\_last\_name,

od\_shipping\_address1, od\_shipping\_phone, od\_shipping\_state

**tbl\_order\_item:** od\_id, pd\_id, od\_qty



**tbl\_product:** pd\_id, cat\_id, pd\_name, pd\_description, pd\_price, pd\_qty, pd\_image, pd\_thumbnail, pd\_date, pd\_last\_update

**tbl\_shop\_config:** sc\_name, sc\_address, sc\_phone, sc\_email, sc\_currency, sc\_order\_email

**tbl\_user:** user\_id, user\_name, user\_password, user\_regdate, user\_last\_login

Οι πινάκες ενώνονται μεταξύ τους μέσω των κλειδιών. Για να κάνουμε την αναζήτηση και αποθήκευση πληροφοριών ακόμα πιο γρήγορη χρησιμοποιήσαμε ευρετήρια.

Η βάση περιέχει εγγραφές στα ελληνικά. Για να δουλεύουν και να εμφανίζονται σωστά στο browser μέσω της php είναι σημαντικό η mysql και η php να χρησιμοποιούν την ίδια κωδικοποίηση. Έτσι έχουμε:

- Η php για να αποθηκεύει και να ανακτά δεδομένα από την βάση έχει κωδικοποίηση utf8.
- Ολόκληρη η βάση έχει κωδικοποίηση utf8. Επίσης κάθε πεδίο χαρακτήρων (char) έχει κωδικοποίηση utf8\_Unicode\_ci

## 6.2 Μηχανές Αποθήκευσης

### 6.2.1 Γενικά

Η MySQL υποστηρίζει διάφορες μηχανές αποθήκευσης, που μερικές φορές ονομάζονται επίσης τύποι πινάκων (table types). Αυτό σημαίνει ότι έχετε μία επιλογή για τον χειρισμό των υποκείμενων πινάκων της βάσης δεδομένων. Κάθε πίνακας της βάσης δεδομένων μας μπορεί να χρησιμοποιεί μία διαφορετική μηχανή αποθήκευσης και μπορείτε εύκολα να κάνετε μετατροπές μεταξύ τους.

Μπορείτε να ελέγξετε τον τύπο ενός πίνακα βάσης δεδομένων όταν δημιουργείτε ένα πίνακα, χρησιμοποιώντας την εντολή

```
CREATE TABLE table TYPE=type....
```

Οι πιθανοί τύποι είναι: MyISAM, ISAM, MERGE, BDB, InnoDB.

Εμείς χρησιμοποιήσαμε την MyISAM.

## 6.2.2 Μηχανή Αποθήκευσης MyISAM

Αυτός ο τύπος είναι η προεπιλογή. Είναι βασισμένος στο παραδοσιακό τύπο ISAM, ο οποίος είναι ακρωνύμιο των Indexed Sequential Access Method, και είναι μία τυπική μέθοδος αποθήκευσης εγγράφων και αρχείων. Ο MyISAM έχει διάφορα επιπλέον πλεονεκτήματα ως προς τον τύπο ISAM. Σε σύγκριση με τις άλλες μηχανές αποθήκευσης, ο MyISAM έχει τα περισσότερα εργαλεία για έλεγχο και επιδιόρθωση των πινάκων. Οι MyISAM πίνακες μπορούν να συμπιεστούν και μπορούν να υποστηρίξουν πλήρη αναζήτηση κειμένου. Δεν είναι ασφαλής ως προς τις συναλλαγές και δεν υποστηρίζει ξένα κλειδιά.

Στις περισσότερες web εφαρμογές, χρησιμοποιούνται είτε πίνακες MyISAM είτε InnoDB, είτε μία μίξη τους.

Πρέπει να χρησιμοποιείτε τον τύπο MyISAM όταν έχετε ένα μεγάλο αριθμό από SELECT ή INSERT σε ένα πίνακα (όχι και τα δύο μαζί) επειδή είναι ο γρηγορότερος τύπος για αυτές τις λειτουργίες. Για πολλές web εφαρμογές, όπως καταλόγους, ο MyISAM είναι η καλύτερη επιλογή. Θα πρέπει επίσης να χρησιμοποιείτε MyISAM εάν χρειάζεστε πλήρεις δυνατότητες αναζήτησης κειμένου.

## 6.3 Σύνδεση στη βάση

Η σύνδεση στη βάση επιτυγχάνετε μέσω της συνάρτησης `mysql_connect()` της PHP δίνοντας τα σχετικά ορίσματα

```
<?php
$dbHost= "localhost";
$dbUser= "webuser";
$dbPass= "webfood";
$dbConn = mysql_connect ($dbHost, $dbUser, $dbPass) or die ('MySQL connect
failed. ' . mysql_error());
?>
```

## 6.4 Ασφάλεια Βάσης Δεδομένων

Όπως θα παρατηρήσατε και παραπάνω, συνδεθήκαμε στη βάση χρησιμοποιώντας τον χρήστη `webuser` και όχι τον `root`. Αυτό το κάναμε για λόγους ασφαλείας. Ο χρήστης `root` είναι ο `administrator` της βάσης δεδομένων και άρα έχει

πλήρη δικαιώματα. Ενώ ο webuser έχει για δικαιώματα μόνο αυτά που του έχουν εκχωρηθεί από τον root.

Είναι πολύ κρίσιμο να μην χρησιμοποιούμε, για Web εφαρμογές, τον χρήστη root. Όπως επίσης είναι κρίσιμο να αποφεύγουμε να δίνουμε δικαιώματα GRANT και REVOKE ανεξέλεγκτα γιατί με αυτές τις παραμέτρους ο χρήστης μπορεί να ορίζει και να αφαιρεί δικαιώματα από χρήστες της βάσης. Με αυτό τον τρόπο μπορεί εύκολα να αποκτήσει πρόσβαση σε περιοχές που ίσως δεν πρέπει. Για την πρόσβαση στη βάση δεδομένων λοιπόν, δημιουργήσαμε ένα χρήστη (τον webuser) ο οποίος μπορεί να έχει πρόσβαση μόνο από το τοπικό μηχάνημα στο οποίο υπάρχει και ο MySQL Server (localhost). Ο χρήστης έχει συγκεκριμένα δικαιώματα, ορισμένα ξεχωριστά για κάθε Table της βάσης. Αυτό γίνεται γιατί δεν υπάρχει λόγος να παρέχουμε δικαιώματα χωρίς να υπάρχει πρακτικός λόγος. Τα βασικά δικαιώματα που παρέχονται είναι INSERT, SELECT, UPDATE, DELETE. Δεν υπάρχει κανένας λόγος ο webuser να κάνει κάποια επιπλέον ενέργεια στο σύστημα.

## 6.5 Αντίγραφο ασφαλείας

Στη MySQL υπάρχουν διάφοροι τρόποι να κάνετε αντίγραφο ασφαλείας. Αλλά εμείς χρησιμοποιήσαμε μόνο ένα. Αυτόν με την εντολή mysqldump.

Για να την χρησιμοποιήσουμε δουλέψαμε από την γραμμή εντολών του λειτουργικού συστήματος (command line). Αρχικά στη γραμμή εντολών με την βοήθεια των εντολών cd και cd.. μετακινούμαστε στο κατάλογο που είναι εγκατεστημένη η mysql. Μετά εκτελώντας την mysqldump κάνουμε τα αρχεία ασφαλείας. Η γενική σύνταξη της mysqldump είναι:

```
mysqldump -u [username] -p [password] [databasename] >  
[backupfile.sql]
```

Στην περίπτωσή μας έχουμε:

```
mysqldump -u root -p ptixiaki > ptixiaki.sql
```

Αυτή την εντολή βάζει ένα σύνολο από όλες τις SQL εντολές που απαιτούνται για να ξαναδημιουργηθεί η βάση δεδομένων στο αρχείο που ονομάζεται ptixiaki.sql .

Παρακάτω παρουσιάζεται ένα μέρος αυτού του αρχείου (ptixiaki.sql) για να καταλάβουν οι αναγνώστες περί τίνος πρόκειται :

```
-- MySQL dump 10.11
--
-- Host: localhost Database: ptixiaki
-----
-- Server version 5.0.51a-community-nt

/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0 */;
/*!40014 SET
@OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;
--
-- Table structure for table `tbl_cart`
--
DROP TABLE IF EXISTS `tbl_cart`;
SET @saved_cs_client = @@character_set_client;
SET character_set_client = utf8;
CREATE TABLE `tbl_cart` (
`ct_id` int(10) unsigned NOT NULL auto_increment,
```

```

`pd_id` int(10) unsigned NOT NULL default '0',
`ct_qty` mediumint(8) unsigned NOT NULL default '1',
`ct_session_id` char(32) collate utf8_unicode_ci NOT NULL default '',
`ct_date` datetime NOT NULL default '0000-00-00 00:00:00',
PRIMARY KEY (`ct_id`),
KEY `pd_id` (`pd_id`),
KEY `ct_session_id` (`ct_session_id`)
) ENGINE=MyISAM AUTO_INCREMENT=160 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci;
SET character_set_client = @saved_cs_client;
--
-- Dumping data for table `tbl_cart`
--
LOCK TABLES `tbl_cart` WRITE;
/*!40000 ALTER TABLE `tbl_cart` DISABLE KEYS */;
INSERT INTO `tbl_cart` VALUES
(143,55,1,'749781864cbdaf26aee83b28d4822a0','2010-03-23 15:32:51');
/*!40000 ALTER TABLE `tbl_cart` ENABLE KEYS */;
UNLOCK TABLES;
--
-- Table structure for table `tbl_category`
--
DROP TABLE IF EXISTS `tbl_category`;
SET @saved_cs_client = @@character_set_client;
SET character_set_client = utf8;
CREATE TABLE `tbl_category` (
  `cat_id` int(10) unsigned NOT NULL auto_increment,
  `cat_parent_id` int(11) NOT NULL default '0',
  `cat_name` varchar(50) collate utf8_unicode_ci NOT NULL default '',
  `cat_description` varchar(200) collate utf8_unicode_ci NOT NULL default '',
  `cat_image` varchar(255) collate utf8_unicode_ci NOT NULL default '',
  PRIMARY KEY (`cat_id`),
  KEY `cat_parent_id` (`cat_parent_id`),
  KEY `cat_name` (`cat_name`)

```

```

) ENGINE=MyISAM AUTO_INCREMENT=27 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci;
SET character_set_client = @saved_cs_client;
--
-- Dumping data for table `tbl_category`
--
LOCK TABLES `tbl_category` WRITE;
/*!40000 ALTER TABLE `tbl_category` DISABLE KEYS */;
INSERT INTO `tbl_category` VALUES (24,0,'BURGERS-CLUB','Με μπιφτέκι και
αφράτο
ψωμάκι!!!','3c02dd5bd6c906a392abffcc121d6a2a.jpg'),(21,0,'ΟΡΕΚΤΙΚΑ','Τα
γευστικότερα ορεκτικά στην
πόλη','7d090d5484c7f1a9373b58a6758ab856.jpg'),(22,0,'ΣΑΛΑΤΕΣ','Χρησιμοποιώντ
ας τα φρεσκότερα
υλικά','3e93f5110a4adcf998460b7ed5eed4ec.jpg'),(23,0,'ΠΙΤΣΕΣ','Μεγάλες
τετράγωνες πίτσες','b579459fa9739aff604e99620d7cb7c0.jpg'),(25,0,'ΚΡΕΠΕΣ-
ΓΛΥΚΑ','Τα καλύτερα
γλυκά','0ca48b534d73c6d266612389ae1c58cb.jpg'),(26,0,'ΑΝΑΨΥΚΤΙΚΑ-
ΠΟΤΑ','Συνοδευτικό φαγητού','17089b64f28c699def5599a1a86b19c7.jpg');
/*!40000 ALTER TABLE `tbl_category` ENABLE KEYS */;
UNLOCK TABLES;

```

## 7.Βιβλιογραφία

### Ελληνόγλωσση:

Meloni, J.C., (2009). Μάθετε PHP, MySQL και APACHE Όλα σε Ένα, Τέταρτη Έκδοση

Welling, L., Thomson, L., (2005). Ανάπτυξη Web Εφαρμογών με PHP και MySQL, Τρίτη Έκδοση

### Ξενόγλωσση:

Atkinson, L. (2000). Core PHP Programming, Second Edition August 3

Bauer, P. (2005). Photoshop CS4 for Dummies

Cabral, S.k. & Murphy, K. (2009). MySQL Administrator's

MySQL Labs AB, (2009). MySQL GUI Tools Manual

### Πηγές από το διαδίκτυο

#### Ελληνόγλωσσες:

Βικιπαιδεία: Η ελεύθερη εγκυκλοπαίδεια (2010). *HyperText Markup Language*

<http://el.wikipedia.org/wiki/HTML> (Τελευταία πρόσβαση στις 29/3/2010)

Βικιπαιδεία: Η ελεύθερη εγκυκλοπαίδεια (2010). *Apache server*

[http://el.wikipedia.org/wiki/Apache\\_HTTP\\_%CE%B5%CE%BE%CF%85%CF%80%CE%B7%CF%81%CE%B5%CF%84%CE%B7%CF%84%CE%AE%CF%82](http://el.wikipedia.org/wiki/Apache_HTTP_%CE%B5%CE%BE%CF%85%CF%80%CE%B7%CF%81%CE%B5%CF%84%CE%B7%CF%84%CE%AE%CF%82)

(Τελευταία πρόσβαση στις 29/3/2010)

Βικιπαιδεία: Η ελεύθερη εγκυκλοπαίδεια (2010). *Cascading Style Sheets*

<http://el.wikipedia.org/wiki/CSS> (Τελευταία πρόσβαση στις 29/3/2010)

Βικιπαιδεία: Η ελεύθερη εγκυκλοπαίδεια (2010). *Javascript*

<http://el.wikipedia.org/wiki/JavaScript> (Τελευταία πρόσβαση στις 29/3/2010)

#### Ξενόγλωσσες:

Antithesis Group (2006). *Πρακτικά θέματα για ανοικτές τεχνολογίες,*

<http://openspot.antithesis.gr/archives/8> (Τελευταία πρόσβαση στις 2/3/2010)

MySQL Labs AB (2010). *MySQL: The world's most popular open source database*

<http://dev.mysql.com/doc/refman/5.0/en/getting-information.html> (Τελευταία

πρόσβαση στις 5/1/2010)

PHP Support (2009). *Gallery: Your photos on your website*,  
<http://codex.gallery2.org/Gallery2:faq> (Τελευταία πρόσβαση στις 27/2/2010)

The PHP Group (2010). *What is PHP?* <http://gr.php.net/manual/en/function.mb-send-mail.php> (Τελευταία πρόσβαση στις 16/12/2009)

The PHP and MySQL Group (2010). *PHP MySQL Tutorial: Learn PHP and MySQL*,  
<http://www.php-mysql-tutorial.com/forums/p/18/284.aspx#284> (Τελευταία πρόσβαση στις 10/1/2010)

Wikipedia: The free Encyclopedia (2010) *HyperText Markup Language*  
[http://en.wikipedia.org/wiki/HyperText\\_Markup\\_Language](http://en.wikipedia.org/wiki/HyperText_Markup_Language) (Τελευταία πρόσβαση στις 29/3/2010)

Wikipedia: The free Encyclopedia (2010) *Apache server*  
[http://en.wikipedia.org/wiki/Apache\\_server](http://en.wikipedia.org/wiki/Apache_server) (Τελευταία πρόσβαση στις 29/3/2010)

Wikipedia: The free Encyclopedia (2010) *Cascading Style Sheets*  
[http://en.wikipedia.org/wiki/Cascading\\_Style\\_Sheets](http://en.wikipedia.org/wiki/Cascading_Style_Sheets) (Τελευταία πρόσβαση στις 29/3/2010)

Wikipedia: The free Encyclopedia (2010) *Javascript*  
<http://en.wikipedia.org/wiki/Javascript> (Τελευταία πρόσβαση στις 29/3/2010)