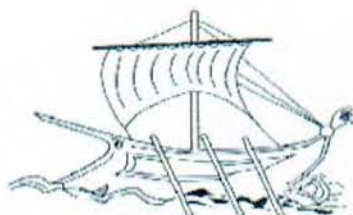


#1/y

SOL  
AUT.



Τεχνολογικό Εκπαιδευτικό Ίδρυμα  
**Τ.Ε.Ι. ΠΕΙΡΑΙΑ**

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΑΥΤΟΜΑΤΙΣΜΟΥ

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

«ΔΙΑΔΙΚΤΥΑΚΗ ΕΦΑΡΜΟΓΗ ΠΡΟΚΡΑΤΗΣΗΣ ΕΙΣΗΤΗΡΙΩΝ ΑΓΩΝΑ ΚΑΛΑΘΟΣΦΑΙΡΙΣΗΣ»

ΠΑΠΑΔΟΠΟΥΛΟΣ ΙΑΣΟΝΑΣ

Α.Μ: 33994

ΜΠΑΛΑΣΗΣ ΣΤΑΥΡΟΣ

Α.Μ: 32541



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ

ΕΠΟΠΤΗΣ: ΑΓΓΕΛΟΠΟΥΛΟΣ ΙΩΑΝΝΗΣ

ΒΙΒΛΙΟΘΗΚΗ  
ΤΕΙ ΠΕΙΡΑΙΑ

ΑΘΗΝΑ, ΜΑΙΟΣ 2012

**Α.Τ.Ε.Ι. ΠΕΙΡΑΙΑ**  
**ΤΜΗΜΑ ΑΥΤΟΜΑΤΙΣΜΟΥ.**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ: ΔΙΑΔΙΚΤΥΑΚΗ ΕΦΑΡΜΟΓΗ  
ΠΡΟΚΡΑΤΗΣΗΣ ΕΙΣΗΓΗΤΗΡΙΩΝ ΑΓΩΝΑ ΚΑΛΑΘΟΣΦΑΙΡΙΣΗΣ**

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΣΠΟΥΔΑΣΤΩΝ: ΜΠΑΛΑΣΗΣ ΣΤΑΥΡΟΣ  
ΠΑΠΑΔΟΠΟΥΛΟΣ ΙΑΣΟΝΑΣ**

**ΕΠΟΠΤΕΥΩΝ: ΑΓΓΕΛΟΠΟΥΛΟΣ ΙΩΑΝΝΗΣ**

**ΑΘΗΝΑ 2012**

# ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΕΧΟΜΕΝΑ .....	1
ΑΝΤΙ - ΠΡΟΛΟΓΟΥ .....	3
ΠΕΡΙΛΗΨΗ.....	4
ΕΙΣΑΓΩΓΗ.....	5
ΚΕΦΑΛΑΙΟ 1 .....	6
1.1 INTERNET, WORLD WIDE WEB ΚΑΙ ΣΗΜΕΡΙΝΗ ΕΠΟΧΗ .....	6
1.1.1 Πρωτόκολλα και υπηρεσίες.....	8
1.1.2 Ιστορία.....	10
1.1.3 World wide web: Η κύρια δικτυακή υπηρεσία σήμερα .....	11
1.2 DATABASES:.....	12
1.2.1 Εισαγωγή – Ορισμοί.....	12
1.2.2 Ιστορική αναδρομή.....	15
1.2.3 DBMS - Γενικά.....	16
1.2.4 Διαδικασία Ανάπτυξης Βάσεων Δεδομένων.....	18
1.2.5 Σχεσιακό μοντέλο.....	20
1.2.6 Σύνοψη.....	22
1.3 PHP.....	22
1.3.1 Γενικά .....	22
1.3.2 Επεκτάσεις αρχείων και διακομιστές .....	24
1.3.3 Ιστορία της rhp .....	24
1.3.4 Rhp και βάσεις .....	25
1.4 STRUCTURED QUERY LANGUAGE (SQL):.....	26
1.4.1 Ορισμός:.....	26
1.4.2 Γενικά:.....	26
1.5 MYSQL.....	28
1.5.1 Τι είναι MySQL;.....	28
1.5.2 Ιστορία MySQL.....	31
1.5.3 Τα κύρια χαρακτηριστικά γνωρίσματα MySQL.....	31
1.5.4 Επισκόπηση MySQLAB.....	38
1.5.5 Σύνοψη.....	39
1.6 ΚΕΝΤΡΙΚΟΣ ΥΠΟΛΟΓΙΣΤΗΣ HTTP APACHE.....	40
1.6.1 Γενικά .....	41
1.6.2 Τι είναι;.....	42
1.6.3 Ιστορία.....	42
1.6.4 Ομάδα Ανάπτυξης .....	43
1.6.5 Χαρακτηριστικά γνωρίσματα.....	44
1.6.6 Χρήση.....	45
1.6.7 Άδεια.....	46
ΚΕΦΑΛΑΙΟ 2 .....	46
2.1 ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ ΚΑΙ ΚΥΚΛΟΣ ΖΩΗΣ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ .....	46
2.1.1 Γενικά .....	46
2.1.2 Βασικές έννοιες.....	47
2.1.3 Βήματα ανάπτυξης λογισμικού .....	49
2.1.4 Κύκλος ζωής .....	50
2.1.5 Μοντέλο του καταρράκτη .....	50
2.2 ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ .....	53
2.2.1 Είδη απαιτήσεων .....	54
2.2.2 Κατηγορίες απαιτήσεων .....	56
2.3 ΠΕΡΙΓΡΑΦΗ ΕΦΑΡΜΟΓΗΣ .....	56

ΚΕΦΑΛΑΙΟ 3 .....	72
3.1 ΣΥΜΠΕΡΑΣΜΑΤΑ .....	72
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	72

## ΑΝΤΙ - ΠΡΟΛΟΓΟΥ

Σκοπός του παρόντος κειμένου είναι η πλήρης περιγραφή και παρουσίαση της πτυχιακής μας εργασίας. Αντικείμενό της αποτέλεσε η εκ του μηδενός σχεδίαση, υλοποίηση και κατασκευή ενός υπολογιστικού συστήματος , στην περίπτωση μας μιας ηλεκτρονικής ιστοσελιδάς για την κράτηση εισιτηρίων, μέσω της οποίας οι χρήστες του διαδικτύου θα μπορούν να κάνουν μια ηλεκτρονική καταχώρηση.Από το μέρος του διαχειριστή θα υπάρχει η δυνατότητα προσθήκης νέων αγώνων, η δημιουργία αντιγράφων του σταδίου για επιλογή θέσεων απο τους πελάτες καθώς επίσης και προβολή των εσόδων και των θέσεων που έχουν κατηληφθεί.

Η διπλωματική αυτή εργασία εκπονήθηκε με την επίβλεψη του καθηγητή κ. Αγγελόπουλου Ιωάννη όπου κατά την ανάπτυξη του συστήματος παρείχε πολύτιμες συμβουλές.

## ΠΕΡΙΛΗΨΗ

Για την περιγραφή του αντικειμένου αυτής της πτυχιακής εργασίας με λίγα λόγια , θα λέγαμε ότι βασικός στόχος είναι η δημιουργία ενός συστήματος ηλεκτρονικής αγοράς (e-market) . Το αποτέλεσμα που επιτυγχάνεται , με την ανάπτυξη της εφαρμογής για την κράτηση θέσης σε γήπεδο, είναι η απλοποίηση της διαδικασίας και η πιο γρήγορη συνδιαλλαγή γιατί τα άτομα στα οποία απευθύνεται η εφαρμογή είναι εκείνα που δεν έχουν τον χρόνο να περιμένουν στην <<ουρά>> των εκδοτηρίων .

Τα βασικά στοιχεία αυτής της πτυχιακής εργασίας είναι δύο:

- Το πρώτο έχει να κάνει με τη χρήση βάσης δεδομένων για την καταχώριση των στοιχείων του χρηστή , την οργάνωση, την αξιοπιστία και την ασφάλεια του.
- Και το δεύτερο πραγματεύεται τις παραμέτρους για την σωστή λειτουργία και την αισθητική παρουσίαση της εφαρμογής καθώς επίσης την ευχρηστία , την ευκολία πρόσβασης και την<<ελαφρότητα>> της (το μικρό σε όγκο κομμάτι του κώδικα που χρησιμοποιήθηκε για την ανάπτυξή της).

Η εφαρμογή υλοποιήθηκε με την γλώσσα προγραμματισμού PHP σε συνεργασία με το σύστημα διαχείρισης βάσεων δεδομένων MySQL και τον διακομιστή Apache, μέσα από ένα πακέτο που τα παρείχε όλα το οποίο ονομάζεται Wamp (Windows , Apache , MySQL , PHP).

## **ΕΙΣΑΓΩΓΗ**

Το θέμα της εργασίας μας είναι η σχεδίαση και η ανάπτυξη συστήματος ηλεκτρονικής αγοράς εισητηριών μέσω της οποίας οι χρήστες του διαδικτύου θα μπορούν να κάνουν μια ηλεκτρονική καταχώρηση. Βασικός μας στόχος είναι η εύκολη και γρήγορη εξυπηρέτηση των πελατών που δεν έχουν χρόνο για αγορά εισητηριων απο τα εκδοτήρια . Προκειμένου να υλοποιήσουμε την παρούσα εφαρμογή χρησιμοποιήσαμε μια σειρά από εργαλεία τα οποία συνέβαλαν στην αποτελεσματική διαχείριση της εφαρμογής. Αναπτύξαμε ένα πρόγραμμα: μία διαδικτυακή (Php και Html) με την δική της βάση δεδομένων. Η βάση δεδομένων περιέχει στοιχεία για πελάτες, στοιχεία πιστωτικών καρτών και καταστάσεις για όλα τα events που θα φιλοξενηθούν στο γήπεδο. Για την είσοδο στην εφαρμογή χρησιμοποιήσαμε κωδικό πρόσβασης. Η εφαρμογή αυτή είναι μια on-line εφαρμογή και ο χρήστης μπορεί να συλλέξει τις πληροφορίες που επιθυμεί μέσω του Ίντερνετ και να κάνει μια κράτηση από τον προσωπικό του υπολογιστή. Το παρόν εγχειρίδιο περιέχει, επιπλέον, αναλυτική περιγραφή της εφαρμογής και των ενεργειών που μπορεί να εκτελέσει ο χρήστης και ο διαχειριστής.

# ΚΕΦΑΛΑΙΟ 1

## 1.1 Internet, World Wide Web και σημερινή εποχή

Το διαδίκτυο (Internet) έχει μπει πλέον για τα καλά στην καθημερινή ζωή μιας μεγάλης πλειοψηφίας ανθρώπων της σημερινής κοινωνίας, και ιδιαίτερα εκείνων που σχετίζονται άμεσα με την ακαδημαϊκή κοινότητα και έχει γίνει αναπόσπαστο κομμάτι της. Πρόκειται για μια νέα πραγματικότητα, για έναν νέο τρόπο εργασίας, εκπαίδευσης και άντλησης πληροφοριών, πέρα βέβαια από τις όποιες άλλες δυνατότητες μπορεί αυτό να παρέχει στον άνθρωπο, όσον αφορά τη διασκέδαση και την ψυχαγωγία του.

Η περιήγηση στο Internet ανοίγει τους κρουνοί της πληροφορίας και της πληροφόρησης, με αποτέλεσμα τα πάντα να βρίσκονται σε απόσταση ενός click, αλλά και καταργεί τις γεωγραφικές αποστάσεις μεταξύ των ανθρώπων κάνοντας την επικοινωνία πιο εύκολη από ποτέ: αναφορές σε σημαντικά γεγονότα, τα τελευταία νέα, τα καιρικά φαινόμενα, οι εξελίξεις στην τεχνολογία ακόμα και τα αποτελέσματα αθλητικών αγώνων, αλλά και οι φίλοι και συγγενείς σε μια άλλη ήπειρο είναι πλέον στο δωμάτιό μας. Επιτυγχάνεται με άλλα λόγια γρήγορη και άμεση επικοινωνία ανθρώπων παγκοσμίως και σε όλα τα επίπεδα (ψυχαγωγικό, ενημερωτικό, επαγγελματικό, εμπορικό, κ.τ.λ.)...

Πώς όμως τα καταφέρνει όλα αυτά το Internet σήμερα;

Το Internet είναι το μεγαλύτερο δίκτυο υπολογιστών. Αριθμεί εκατοντάδες χιλιάδες υπολογιστές συνδεδεμένους μεταξύ τους και καινούριοι να συνδέονται σε αυτό με ολοένα αυξανόμενο ρυθμό. Ξεκίνησε κυρίως για εκπαιδευτικούς σκοπούς, αλλά μετεξελίχθηκε σε ένα παγκόσμιο δίκτυο ποικίλων δραστηριοτήτων. Αποτελείται στην ουσία από πολλούς υπολογιστές συνδεδεμένους μεταξύ τους σε υποδίκτυα τοπικού χαρακτήρα, τα λεγόμενα Local Area Networks

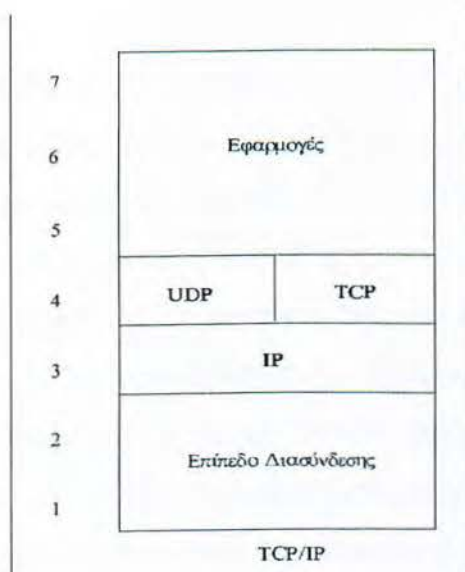
(LAN), τα οποία συνδέονται περαιτέρω υλοποιώντας μεγαλύτερα Wide Area Networks (WAN), τα οποία και συνεχίζουν την αλυσίδα κατά κάποιο τρόπο δημιουργώντας τελικά έναν πλήρως αποκεντρωμένο δίκτυο, με εκατομμύρια κόμβους-υπολογιστές. Χαρακτηριστικό του αποτελεί η εκ φύσεως παντελής έλλειψη κεντρικής διοίκησης, γεγονός που του προσδίδει τεράστια δύναμη ως μέσο. Οι προοπτικές ανάπτυξης του είναι τεράστιες. Ήδη νέες μορφές των βασικών του πρωτοκόλλων κάνουν την εμφάνισή τους, με ριζοσπαστικές αλλαγές στην σχεδίαση και την υλοποίησή τους, γεγονός που του δίνει νέες διαστάσεις. Το Internet βρίσκεται ήδη προ των πυλών με ταχύτητες πολλαπλάσιες του υπάρχοντος δικτύου, ενώ μια πληθώρα τεχνολογιών και εφαρμογών στηρίζεται, χρησιμοποιεί και αναπτύσσει το ίδιο το Internet.

### **1.1.1 Πρωτόκολλα και υπηρεσίες**

Προγραμματιστικά κοιτώντας, το Internet έχει χτιστεί πάνω σε μια σουίτα πρωτοκόλλων που είναι γνωστά με την ονομασία Transmission Control Protocol/Internet Protocol - TCP/IP (ή User Datagram Protocol/Internet Protocol - UDP/IP, ανάλογα το είδος της μεταφοράς των πακέτων) και αποτελούν de facto standards για τη μετάδοση δεδομένων σε αυτό. Η ανάπτυξή τους έγινε αρχικά από το Υπουργείο Εθνικής Άμυνας των Ηνωμένων Πολιτειών της Αμερικής (Η.Π.Α.) στη δεκαετία του 1970 και η πρώτη τους ευρεία χρησιμοποίηση πραγματοποιήθηκε στο δίκτυο υπολογιστών ARPANET, το πρώτο πειραματικό WAN με μεταγωγή πακέτου. Το NSFNET αποτέλεσε τη διάδοχη κατάσταση στις Η.Π.Α., για να φτάσουμε τελικά στη σημερινή μορφή του Internet γύρω στο 1993. Η ανάπτυξη του τελευταίου ήταν ταχύτερη και μάλλον άναρχη, με μία κεντρική αρχή διαχείρισης, τη λίγο έως πολύ γνωστή Internet Network Information Center – InterNIC. Η διασύνδεση των τοπικών LANs γίνεται με μεταγωγείς πακέτου



(ηλεκτρονικούς ή πλέον και οπτικούς), routers και gateways, που πραγματοποιούν δρομολόγηση βασισμένη στα πρωτόκολλα IP, η δε αξιοπιστία της μετάδοσης των πακέτων εξασφαλίζεται από τους ακραίους χρήστες μέσω των πρωτοκόλλων TCP ή UDP. Η στοίβα των επιπέδων στην οικογένεια των TCP/IP προτύπων έχει ως εξής και βασίζεται στην αρχιτεκτονική των στρωμάτων, μεθοδολογία απαραίτητη στην υλοποίηση πρωτοκόλλων επικοινωνίας:



Εικόνα 1.1.1 Α

Οι εφαρμογές που προσφέρονται στους τελικούς χρήστες είναι πολλές και ποικίλες. Ενδεικτικά αναφέρουμε:

- **www:** Ίσως η πιο διαδεδομένη από τις εφαρμογές και σίγουρα η υπεύθυνη για την μαζική απήχηση του Internet στους ανθρώπους. Πολύ απλή στη χρήση ακόμα και για τον αρχάριο χρήστη, εύκολα προσβάσιμη και με τεράστιες δυνατότητες επικοινωνίας, αποτελεί και τη βασική εφαρμογή που θα χρησιμοποιήσει το σύστημα εγγραφών.

- e-mail: Πολύ διαδεδομένη και αυτή η εφαρμογή, καθώς δίνει τη δυνατότητα εύκολης ανταλλαγής μηνυμάτων σε παγκόσμιο επίπεδο μέσα σε λίγα δευτερόλεπτα.
- ftp: Εφαρμογή μεταφοράς αρχείων υπολογιστών.
- snmp: Εφαρμογή διαχείρισης κόμβων ενός δικτύου
- telnet: Εφαρμογή πρόσβασης σε απομακρυσμένους υπολογιστές, κ.ο.κ.

### 1.1.2 Ιστορία

Μέχρι και τις αρχές της δεκαετίας του 1990, τον πληθυσμό του διαδικτύου αποτελούσαν κυρίως ακαδημαϊκοί, κυβερνητικοί και βιομηχανικοί ερευνητές. Αυτό άλλαξε όταν το 1989 στο CERN, το ευρωπαϊκό κέντρο πυρηνικών ερευνών (European laboratory for particle physics ή conseil europeene pour la recherche nucleaire) επινοήθηκε μια καινούργια εφαρμογή από τον φυσικό Tim Berners-Lee. Η εφαρμογή αυτή ονομάστηκε παγκόσμιος ιστός (world wide web) και έφερε εκατομμύρια νέους, μη ακαδημαϊκούς χρήστες στο διαδίκτυο. Ο Tim Bernerd- Lee δεν άλλαξε τίποτα από τις υποκείμενες υπηρεσίες αλλά τις κατέστησε περισσότερο εύχρηστες. Το www μαζί με την πρώτη διεπαφή που δημιουργήθηκε για τον χρήστη, το mosaic (Φεβρουάριος 1993), κατέστησε δυνατή τη δημιουργία, σε κάποιο ποσοστό σελίδων πληροφορίας που περιέχουν κείμενο, εικόνες, ήχο ακόμα και βίντεο, με παραπομπές σε άλλες σελίδες. Με ένα «κλικ» του ποντικιού πάνω στην παραπομπή, ο χρήστης μεταφέρεται ξαφνικά στη σελίδα που δείχνει η παραπομπή. Η ομάδα, λοιπόν, του CERN δημιούργησε ένα πρωτόκολλο βασισμένη σε υπέρ-κείμενο (hypertext) που καθιστά δυνατή τη σύνδεση περιεχομένων στον ιστό χρησιμοποιώντας υπέρ-ζεύξεις (hyperlinks). Η τεράστια δημοτικότητα του παγκοσμίου ιστού έγκειται στο γεγονός ότι έχει μια έγχρωμη γραφική διεπαφή, εύχρηστη για αρχάριους χρήστες και

παρέχει τεράστιο πλούτο πληροφοριών για κάθε πιθανό θέμα. Σύμφωνα με τον ιδρυτή του ιστού , «Το όνειρο που κρύβεται πίσω από τον παγκόσμιο ιστό είναι ένας κοινός χώρος / ένα κοινό διάστημα πληροφορίας, μες στον οποίο επικοινωνούμε μοιράζοντας πληροφορίες. Η παγκοσμιότητα του είναι ουσιώδης: μια παραπομπή ή ένας σύνδεσμος υπερκειμένου μπορεί να οδηγήσει, να δείξει σε οτιδήποτε, προσωπικό, τοπικό, παγκόσμιο, πρόχειρο ή πολύ τελειοποιημένο». Η θεαματική ανάπτυξη του διαδικτύου συνεχίζεται όλο και περισσότερο, το διαδίκτυο καλωσορίζει κάθε μήνα εκατομμύρια καινούργιους χρήστες και οι μεγαλύτερες εταιρείες στον κόσμο επενδύουν εκατομμύρια. Με βάση αυτά τα στοιχεία οι παρακάτω προβλέψεις σχετικά με το μέλλον του διαδικτύου είναι μάλλον βέβαιες:

- Τα εργαλεία ανάπτυξης εξυπηρετητών παγκόσμιου ιστού θα τείνουν όλο και περισσότερο στην παροχή δυνατοτήτων που θα ελκύουν τους χρήστες.
- Η διανομή του λογισμικού θα γίνεται όλο και περισσότερο μέσω του διαδικτύου με αποτέλεσμα να μειωθεί η χρησιμοποίηση cd-rom και δισκετών.
- Το video και audio streaming θα διαδοθεί ακόμα περισσότερο και η σύγκλιση της τηλεόρασης και υπολογιστή είναι πολύ κοντά.
- Θα αναπτυχθεί ιδιαίτερα το ηλεκτρονικό εμπόριο (electronic commerce).
- Η ανωνυμία ένα από τα ελκυστικότερα χαρακτηριστικά του για τους περισσότερους σιγά σιγά θα μειωθεί καθώς αναμένεται να μπου κανονισμοί και αρχές στη λειτουργία του.

- Η στρατηγική χρέωσης των υπηρεσιών του διαδικτύου αναμένεται να αλλάξει και να προσαρμοστεί στο είδος δεδομένων που κάθε χρήστης διακινεί.

### **1.1.3 World wide web: Η κύρια δικτυακή υπηρεσία σήμερα**

Ας δούμε, όμως, λίγο αναλυτικότερα το world wide web (www), μιας και η χρήση του αποτελεί σημαντικότατο κομμάτι του συστήματος μας, και μοναδικό τρόπο πρόσβασης σε αυτό για τους χρήστες του. Δίνει την δυνατότητα εύκολης και προσιτής αναπαράστασης της πληροφορίας. Την δυνατότητα αυτή την οφείλει κυρίως στο γραφικό περιβάλλον που χρησιμοποιεί, στην δυνατότητα χρήσης υπηρεσιών πολυμέσων (Εικόνα – γραφικά – ήχος – video) και στην ανοιχτή σε νέους τύπους δεδομένων αρχιτεκτονική του. Είναι δομημένο σε σελίδες (web pages) και ανοικτό ουσιαστικά σε όλους, όσους με στοιχειώδεις γνώσεις της γλώσσας συγγραφής web pages Hypertext Markup Language (html) μπορούν να φτιάξουν μία σελίδα και να την κρεμάσουν σε κάποιον κόμβο-server του δικτύου. Η ουσία της ζωής του www είναι η δυνατότητα αλληλοσύνδεσης των σελίδων μέσω hyperlinks (ή απλά: links), δημιουργώντας τελικά έναν ιστό (web) από αυτές. Το www δηλαδή χρησιμοποιεί ένα σύστημα οργάνωσης και κωδικοποίησης πληροφοριών βασισμένο σε hypertexts, διευκολύνοντας έτσι την πλοήγηση των χρηστών στις διάφορες περιοχές δεδομένων. Το κυριότερο χαρακτηριστικό και πλεονέκτημα των hypertexts είναι η δυνατότητα δημιουργίας hyperlinks μεταξύ κειμένων που βρίσκονται ακόμα και σε διαφορετικούς υπολογιστές συνδεδεμένους σε κάποιο δίκτυο (εν προκειμένω στο Internet). Τα αρχεία αυτά των hypertexts είναι γνωστά σαν web pages. Τέλος, το πρωτόκολλο επικοινωνίας των κόμβων-υπολογιστών που έχουν αναρτημένες τις σελίδες και λειτουργούν 24ώρες/24ώρο ως servers, με οποιονδήποτε υπολογιστή-client στο δίκτυο

θελήσει να δει μία σελίδα, είναι συνήθως το Hypertext Transfer Protocol (http), ενώ σε περιπτώσεις που υπάρχει ανάγκη για extra ασφάλεια στη συναλλαγή χρησιμοποιείται το Secure – Hypertext Transfer Protocol – (https). Οι clients χρησιμοποιούν web-browsers για την απεικόνιση των σελίδων και γενικά την πλοήγηση στο www, ενώ οι servers τρέχουν ειδικούς δαίμονες, τους web servers.

## **1.2 Databases:**

### **1.2.1 Εισαγωγή – Ορισμοί**

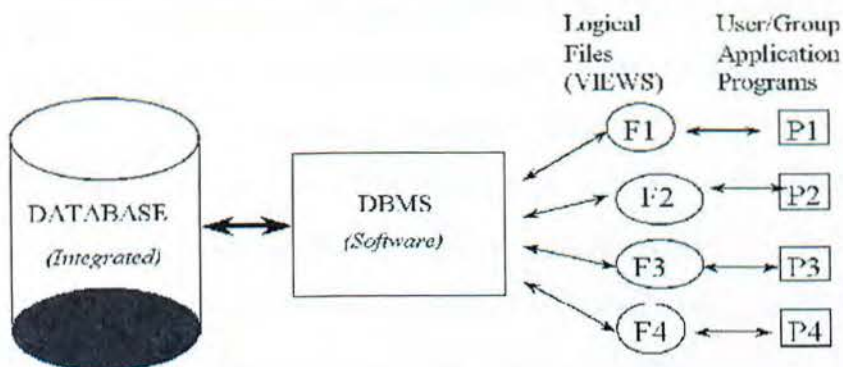
Με τον όρο «*database - βάση δεδομένων*» εννοούμε γενικά μια συλλογή αλληλοσχετιζόμενων δεδομένων (data) που αποθηκεύονται μαζί χωρίς άχρηστους πλεονασμούς (redundancies) για την εξυπηρέτηση πολλών εφαρμογών. Η αποθήκευση των δεδομένων είναι τέτοια ώστε τα δεδομένα είναι ανεξάρτητα των προγραμμάτων που διαχειρίζονται. Η εισαγωγή νέων δεδομένων, η τροποποίηση και η ανάκτηση δεδομένων από τη βάση δεδομένων ακολουθεί κοινή και ελεγχόμενη προσέγγιση για όλες τις εφαρμογές. Η δομή των δεδομένων παρέχει το θεμέλιο για ανάπτυξη μελλοντικής εφαρμογής. Δηλαδή, η οργάνωση των δεδομένων είναι τέτοια ώστε μπορούμε να υλοποιήσουμε νέες εφαρμογές χωρίς ιδιαίτερο κόπο και χωρίς να αλλάξει οτιδήποτε στις παλιές. Τέλος λέμε ότι ένα σύστημα περιέχει μια συλλογή από βάσεις δεδομένων αν οι βάσεις αυτές είναι τελείως ξεχωριστές σε δομή.

Ακολουθεί μια λίστα από κρίσιμα χαρακτηριστικά μιας βάσης δεδομένων:

- Αναπαριστά κάποια όψη του πραγματικού κόσμου και τα δεδομένα της είναι γνωστά, «αναγνωρίσιμα» (events, facts) που μπορούν να καταγραφούν και έχουν κάποια σημασία για τους χρήστες.

- Η αποθήκευση των δεδομένων είναι τέτοια ώστε τα δεδομένα να είναι ανεξάρτητα των προγραμμάτων που διαχειρίζονται.
- Η εισαγωγή νέων δεδομένων, η τροποποίηση και η ανάκτηση δεδομένων από την βάση δεδομένων ακολουθεί κοινή και ελεγχόμενη προσέγγιση για όλες τις εφαρμογές.
- Η οργάνωση των δεδομένων είναι τέτοια ώστε μπορούμε να υλοποιήσουμε νέες εφαρμογές χωρίς ιδιαίτερο κόπο και χωρίς να αλλάξει οτιδήποτε στις παλιές.
- Η οργάνωση των δεδομένων διευκολύνει, επίσης, την υλοποίηση νέων εφαρμογών στήριξης αποφάσεων
- Μια βάση δεδομένων μπορεί να έχει οποιοδήποτε μέγεθος και πολυπλοκότητα και συντηρείται κατά κανόνα από κάποιο Database Management System (DBMS). Ένα σύστημα διαχείρισης βάσεων δεδομένων είναι ένα λογισμικό που επιτρέπει στο χρήστη την υλοποίηση, την επεξεργασία και την συντήρηση βάσεων δεδομένων. Με άλλα λόγια, επιτρέπει τον ορισμό δεδομένων και τη διαχείριση δεδομένων. Επίσης το σύστημα αυτό αναλαμβάνει τη διαχείριση δεδομένων όλων των εφαρμογών και συνεργαζόμενο με το λειτουργικό σύστημα τα αποθηκεύει. Το σενάριο επεξεργασίας με ένα DBMS είναι σε γενικές γραμμές ως εξής:

## Το Σενάριο Επεξεργασίας με DBMS



Εικόνα 1.2.1 Α

Ο ορισμός μιας βάσης δεδομένων περιλαμβάνει την προδιαγραφή των τύπων, των δομών και των περιορισμών των δεδομένων που θα αποθηκευτούν σε αυτή. Η κατασκευή της είναι η διαδικασία αποθήκευσης των δεδομένων σε ένα μέσο αποθήκευσης όπως π.χ. ο σκληρός δίσκος ενός υπολογιστή. Τέλος ο χειρισμός μιας βάσης δεδομένων περιλαμβάνει λειτουργίες όπως υποβολή queries για ανάκτηση συγκεκριμένων δεδομένων, ενημέρωση της βάσης έτσι ώστε να βρίσκεται σε συμφωνία με τον μικρόκοσμο που αναπαριστά, καθώς και παραγωγή αναφορών από τα δεδομένα.

### 1.2.2 Ιστορική αναδρομή

Ήταν το 1970 όταν ο Ted Codd ανέπτυξε το θεωρητικό ορισμό του σχεσιακού μοντέλου δεδομένων, το οποίο αποτέλεσε το θεμέλιο λίθο της τεχνολογίας βάσεων δεδομένων όπως τη γνωρίζουμε σήμερα. Το μοντέλο αυτό χαρακτηρίζεται από τη δυνατότητα μαζικής επεξεργασίας των δεδομένων και από την ανεξαρτησία των εφαρμογών από τη φυσική υλοποίηση. Όμως οι βάσεις δεδομένων δεν ήταν ιδιαίτερα διαδεδομένες

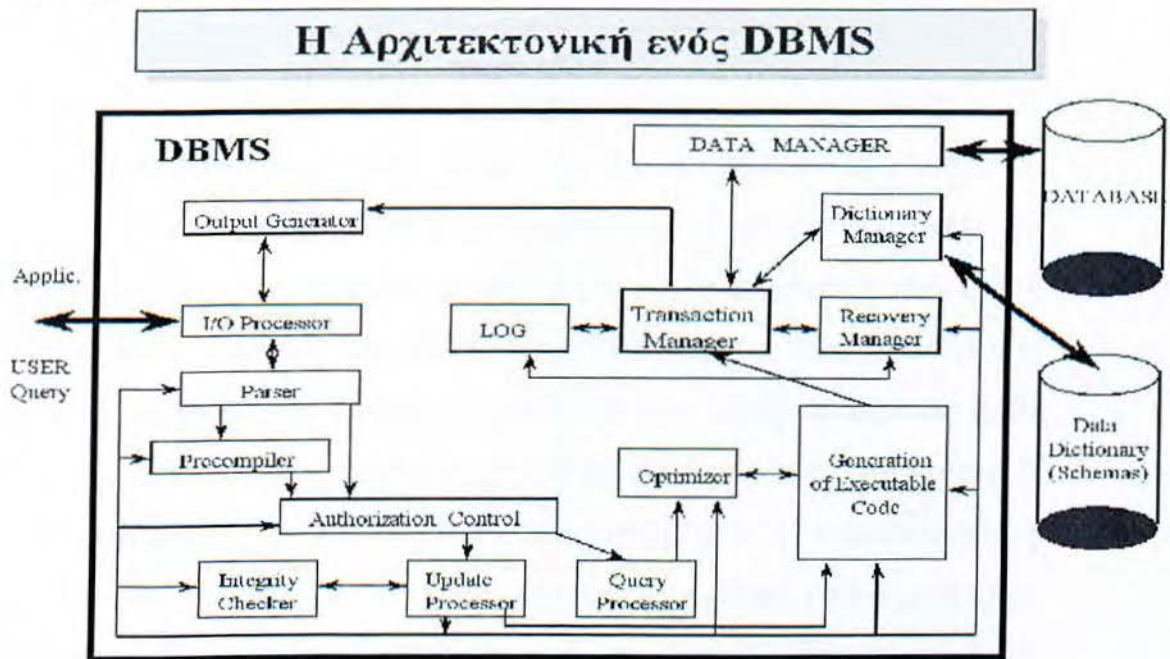
στη δεκαετία του 1970 και ιδιαίτερα στους τελικούς χρήστες. Στα μέσα της δεκαετίας κυριαρχούσαν, αν μπορεί να ειπωθεί κάτι τέτοιο, δύο σχεσιακά συστήματα, η Ingres και το System R. Σήμερα απόγονοι των παραπάνω συστημάτων θεωρούνται η Ingres, η Sybase και ο Microsoft SQL Server για την Ingres, και η Oracle και IBM DB2 για το System R. Στις αρχές της δεκαετίας του 1980 ξεκίνησε η ουσιαστική ανάπτυξη των σχεσιακών συστημάτων, με την Oracle να κυριαρχεί σταδιακά στο χώρο. Την αλματώδη ανάπτυξη τη δεκαετία του 1980 βοήθησαν σημαντικά και άλλοι παράγοντες, όπως η ανάπτυξη υψηλού επιπέδου ισχυρότατων γλωσσών προγραμματισμού και επεξεργασίας, η πρόοδος στη θεωρία των βάσεων δεδομένων και η ανάπτυξη εν τέλει της ίδιας της Structured Query Language (SQL). Ήταν αναμφισβήτητα η εποχή των Relational Database Systems. Τέλος, προς το τέλος της δεκαετίας έκαναν την εμφάνισή τους και τα πρώτα αντικειμενοστραφή και ευφυή συστήματα. Η προηγούμενη δεκαετία του 1990 ήταν η εποχή των σύνθετων οντοτήτων στις βάσεις δεδομένων, όπως τα πολυμέσα, software components και engineering αντικείμενα τα οποία διαμόρφωσαν μια νέα κατηγορία συστημάτων βάσεων δεδομένων, τα Object-Relational Database Systems. Περαιτέρω, γινόταν όλο και περισσότερο λόγος για έννοιες όπως κατανεμημένες βάσεις δεδομένων, παράλληλοποίηση, συστήματα βάσεων δεδομένων σε προσωπικούς υπολογιστές, συστήματα βάσεων δεδομένων στο web. Τελικά σήμερα η βάση δεδομένων αποτελεί αναπόσπαστο κομμάτι οποιουδήποτε medium ή large scale συστήματος, τόσο σε επίπεδο προσωπικών υπολογιστών, όσο και στο Internet. Με χρήση ενός προγράμματος βάσης δεδομένων απλοποιείται και διευκολύνεται πληθώρα εργασιών, από την πιο απλή, όπως η οργάνωση μιας αποθήκης, έως την πιο σύνθετη, όπως η συλλογή δεδομένων από ένα διαστημόπλοιο. Στο Internet δε, με τον όγκο της πληροφορίας να



αυξάνεται συνεχώς, η χρήση database καθίσταται επιτακτική για οποιοδήποτε αρκετά μεγάλο web site.

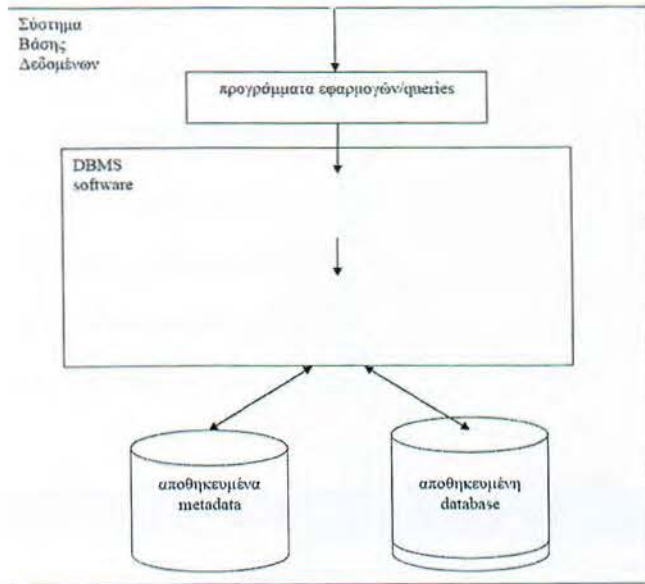
### 1.2.3 DBMS - Γενικά

Η αρχιτεκτονική ενός DBMS παρουσιάζεται αναλυτικά στο παρακάτω σχήμα και σε πρώτη όψη τρομάζει τον ανίδεο αναγνώστη. Με μια πιο ψύχραιμη ματιά παρατηρεί κανείς την ύπαρξη διακριτών οντοτήτων, κάθε μια από τις οποίες έχει συγκεκριμένη αποστολή και στόχο στο όλο σύστημα και η οποία αποτελεί αναπόσπαστο τμήμα του.



Εικόνα 1.2.3 Α

Αν απλουστεύσουμε λίγο τα παραπάνω, θα καταλήγαμε ίσως στο εξής σχήμα:



**Εικόνα 1.2.3 Β**

Και λέμε ίσως, γιατί είναι πιθανές διάφορες αρχιτεκτονικές DBMS χωρίς κάποια να είναι δεσμευτική. Ο βασικός λόγος της ευρείας χρήσης και διάδοσης των DBMS είναι η οργάνωση των πληροφοριών μέσα από εξειδικευμένους μηχανισμούς αποθήκευσης των δεδομένων, καθώς και οι user-friendly γλώσσες και περιβάλλοντα αναζήτησής τους. Μεταξύ των πλεονεκτημάτων, που παρέχει η διαχείριση μιας βάσης δεδομένων μέσα από ένα DBMS, είναι η ανάκαμψη από καταστροφές του συστήματος, η συνδρομικότητα, η γρήγορη ανάπτυξη εφαρμογών, η ακεραιότητα, η ασφάλεια, κ.α. Στην ανεξαρτησία των προγραμμάτων από τις φυσικές δομές βοηθούν πεδία αφαίρεσης, ενώ η επιτυχής διαχείριση αυτή καθαυτή ενός τέτοιου συστήματος αποτελεί ιδιαίτερα δύσκολη εργασία που πραγματοποιούν υπεύθυνοι διαχειριστές (Database Administrators - DBAs).

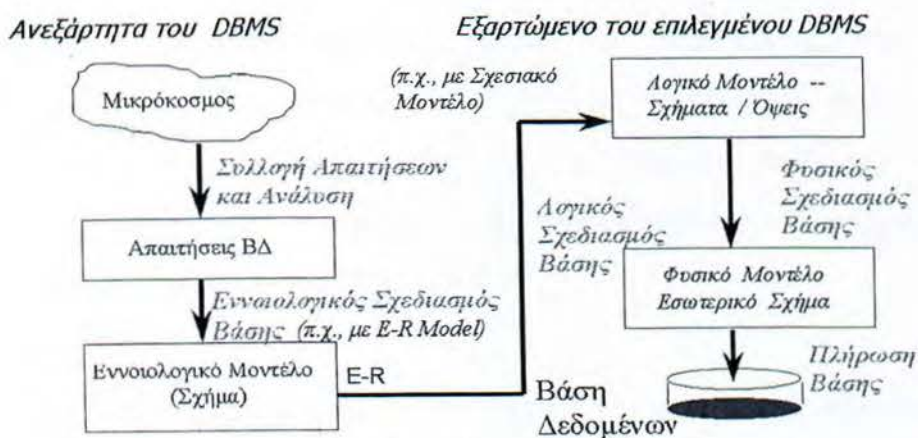
#### **1.2.4 Διαδικασία Ανάπτυξης Βάσεων Δεδομένων**

Η διαδικασία της ανάπτυξης μιας βάσης δεδομένων είναι ουσιαστικά η διαδικασία μετατροπής της γνώσης του πραγματικού μικρόκοσμου σε

για μια βάση δεδομένων και για καλύτερη κατανόηση της διακρίνεται σε συνεχείς αυτοτελείς διαδικασίες:

- Συλλογή απαιτήσεων και ανάλυση
- Εννοιολογικός σχεδιασμός της βάσης
- Λογικός σχεδιασμός της βάσης
- Φυσικός σχεδιασμός της βάσης
- Φόρτωση της βάσης με δεδομένα

### Πλήρης Διαδικασία Ανάπτυξης ΒΔ



Εικόνα 1.2.4 Α

1. Η πρώτη φάση ανάπτυξης της βάσης δεδομένων είναι απαραίτητη για την «εξόρυξη» των απαιτήσεων από περιγραφές του πραγματικού μικρόκοσμου. Πρόκειται για μία δύσκολη εννοιολογικά διαδικασία η οποία απαιτεί εμπειρία και γνώση, καθώς συχνά στηρίζεται σε ασαφείς και ημιτελείς περιγραφές.
2. Στη συνέχεια ακολουθεί ο σχεδιασμός της βάσης, το αποτέλεσμα του οποίου θα είναι και το σχήμα της. Ο σχεδιασμός πραγματοποιείται ακολουθώντας ένα συγκεκριμένο μοντέλο, ενώ υπάρχουν τρεις διαφορετικοί αυτοτελείς σχεδιασμοί που δημιουργούν τελικά τρία διαφορετικά εννοιολογικά σχήματα της

βάσης: Ο εννοιολογικός σχεδιασμός, ο λογικός σχεδιασμός και ο φυσικός σχεδιασμός. Αναλυτικότερα:

- Ο εννοιολογικός σχεδιασμός αποτελεί ουσιαστικά μία προσπάθεια για ξεκαθάρισμα των εννοιών και ως κύριο στόχο έχει την παραγωγή μίας αφαιρετικής, αλλά και ταυτόχρονα πλήρους περιγραφής του τμήματος του μικρόκοσμου που θα αναπαρασταθεί στη βάση. Στη φάση αυτή πραγματοποιείται ο προσδιορισμός των σημασιολογικών αντικειμένων, των σχέσεων μεταξύ τους καθώς και των κανόνων που ισχύουν εν γένει. Το εννοιολογικό μοντέλο που κυριάρχησε, από τη δεκαετία του 1980 κιόλας, είναι το μοντέλο οντοτήτων-συσχετίσεων (Entity-Relationship model, E-R). Στους βασικούς λόγους της ευρείας διάδοσης του συγκαταλέγονται η απλότητά του, η σαφήνεια του και ο γραφικός συμβολισμός που ακολουθεί.
- Ο λογικός σχεδιασμός ασχολείται με την διαδικασία μετατροπής ενός εννοιολογικού μοντέλου σε τυπικά σχήματα εκφρασμένα στο υποστηριζόμενο από το DBMS μοντέλο δεδομένων (π.χ., Σχεσιακό Μοντέλο).
- Τέλος, ο φυσικός σχεδιασμός έχει ως κύριο στόχο την **απόδοση** του όλου συστήματος και αποτελεί τη διαδικασία καθορισμού των αρχείων που υλοποιούν τον λογικό σχεδιασμό και των οργανώσεων αυτών, καθώς και των δομών ευρετηρίων που αποτελούν το εσωτερικό σχήμα.

Η τελική φάση είναι η διαδικασία φόρτωσης της βάσης με τα δεδομένα, συνήθως μέσω βοηθητικών προγραμμάτων του DBMS για μαζική

επεξεργασία. Για τις παραπάνω φάσεις υπάρχουν διάφορα προγραμματιστικά εργαλεία που αυτοματοποιούν τις διαδικασίες σε μεγάλο βαθμό.

### 1.2.5 Σχεσιακό μοντέλο

Στη σημερινή εποχή το πλέον δημοφιλές μοντέλο είναι το σχεσιακό που προτάθηκε όπως αναφέρθηκε από τον Codd το 1970. Σχεδόν όλα τα μεγάλα DBMS είναι καθαρά σχεσιακά. Η βασική αρχή μιας σχεσιακής βάσης δεδομένων είναι ότι έχει ένα σύνολο από σχέσεις (relations-tables), που είναι και το μοναδικό είδος δομών. Κάθε στήλη στον πίνακα έχει επικεφαλίδα που ονομάζεται γνώρισμα (attribute-field) και πεδία τιμών, ενώ κάθε γραμμή ονομάζεται πλειάδα (tuple) και παριστάνει τα χαρακτηριστικά μιας οντότητας στο μοντέλο. Ένα στιγμιότυπο σχέσης (relation instance) είναι προφανώς ένα σύνολο πλειάδων. Τα βασικά χαρακτηριστικά των σχέσεων συνοψίζονται στα εξής:

- Η διάταξη των γνωρισμάτων σε μία σχέση έχει σημασία, σε αντίθεση με την διάταξη των πλειάδων.
- Κάθε πλειάδα αποθηκεύεται **μία** φορά σε μία σχέση.
- Μια τιμή μπορεί να παρουσιάζεται πολλές φορές σε μία στήλη και είναι ατομική.
- Υπάρχει μια ειδική τιμή το κενό (null) και χρησιμοποιείται για να παραστήσει στη βάση μιας τιμή που είναι μη-εφαρμόσιμη (non-applicable) ή άγνωστη (unknown).

Υπάρχουν διάφορα είδη περιορισμών, όπως:

- Δομικοί περιορισμοί: κλειδιού, ακεραιότητας οντότητας και αναφορικής ακεραιότητας. Ανάλογα το εμπορικό πακέτο που χρησιμοποιείται υπάρχει και αντίστοιχη υλοποίησή τους στη βάση.
- Περιορισμοί πεδίου τιμών.

- Περιορισμοί στηλών, οι οποίοι αναφέρονται στις τιμές των γνωρισμάτων.
- User-defined περιορισμοί, όπου μια σειρά μηχανισμών χρησιμοποιούνται για την υποστήριξη τέτοιων κανόνων σε ένα σχεσιακό σύστημα, όπως stored Procedures, triggers, methods (για object-oriented συστήματα).

### 1.2.6 Σύνοψη

Συμπερασματικά, αν θελήσουμε να αναφέρουμε τα χαρακτηριστικά εκείνα που έχουν μετατρέψει πλέον τις βάσεις δεδομένων και τα DBMS σε αναπόσπαστο τμήμα των υπολογιστικών εφαρμογών και όχι μόνο, θα καταλήγαμε στα εξής σημεία:

- Σε σχέση με τις παραδοσιακές εφαρμογές επεξεργασίας αρχείων οι βάσεις δεδομένων έχουν να αντιπαρατάξουν την αφαίρεση δεδομένων, την ιδιαίτερα σημαντική ανεξαρτησία προγραμμάτων-δεδομένων και προγραμμάτων-πράξεων, την υποστήριξη όψεων πολλαπλών χρηστών και το διαμοιρασμό των δεδομένων μεταξύ πολλαπλών δοσοληψιών.
- Στον τομέα της διαχείρισης, του σχεδιασμού και της χρήσης μιας βάσης δεδομένων, τα DBMS παρέχουν πολύπλοκους μηχανισμούς ελέγχου των δεδομένων, περιορισμούς μη εξουσιοδοτημένης προσπέλασης, δυνατότητες για εξαγωγή συμπερασμάτων μέσω επαγωγικών κανόνων, παροχή πολλαπλών interfaces, αναπαράσταση πολύπλοκων συσχετίσεων μεταξύ των δεδομένων, επιβολή περιορισμών ορθότητας και τέλος την πολύ σημαντική παροχή αντιγράφων και δυνατότητας recovery.
- Περαιτέρω πλεονεκτήματα θεωρούνται οι δυνατότητες για την επιβολή προτύπων, η ελάττωση του χρόνου ανάπτυξης των

εφαρμογών και η διαθεσιμότητα ενημερωμένων πληροφοριών σε όλους τους χρήστες.

## 1.3 PHP

### 1.3.1 Γενικά

Η PHP είναι μια ισχυρή γλώσσα προγραμματισμού γενικού σκοπού scripting για την συγγραφή σεναρίων, η οποία συνεργάζεται πολύ καλά με την html. Αποτελεί δε ένα πολύ καλό εργαλείο για τον σχεδιασμό δυναμικών ιστοσελίδων και είναι επισήμως γνωστή ως: HyperText preprocessor. Ένα από τα σημαντικότερα χαρακτηριστικά της php είναι ότι είναι μια γλώσσα ανοιχτού κώδικα και διανέμεται ελεύθερα, το οποίο σημαίνει ότι κάθε υλικό αναφοράς της μπορεί να εντοπιστεί εύκολα και να χρησιμοποιηθεί με ευκολία σε διάφορα λειτουργικά συστήματα. Για τον έλεγχο των προγραμμάτων php απαιτείται ένας εξυπηρετητής που υποστηρίζει τη γλώσσα. Τα στοιχεία HTML στον πηγαίο κώδικα μένουν ως έχουν, αλλά ο PHP κώδικας ερμηνεύεται και εκτελείται. Ο κώδικας PHP μπορεί να θέσει ερωτήματα σε βάσεις δεδομένων, να δημιουργήσει εικόνες, να διαβάσει και να γράψει αρχεία, να συνδεθεί με απομακρυσμένους υπολογιστές, κ.ο.κ. Σε γενικές γραμμές οι δυνατότητες που μας δίνει είναι απεριόριστες.

Η php είναι ένα πρόγραμμα που μπορεί να εγκατασταθεί σε ένα εξυπηρετητή ιστού σαν ενσωματωμένο επιπλέον λογισμικό (module), είτε σαν αυτόνομος εκτελέσιμος κώδικας. Προτιμάται η πρώτη περίπτωση, αφού έτσι είναι γρηγορότερη η εκτέλεση και δεν επιβαρύνεται ο εξυπηρετητής με επιπλέον απαιτήσεις σε μνήμη και χώρο στο δίσκο. Χρησιμοποιείται με εκδόσεις των apache, Microsoft iis, Netscape enterprise server και άλλων πακέτων. Η σύνταξη της γλώσσας php είναι παρόμοια με αυτήν της Perl και της C. Όταν κάποιος πελάτης

(χρήστης) επισκεφτεί την ιστοσελίδα που περιέχει κώδικα php ο εξυπηρετητής εκτελεί τον κώδικα και αυτό που επιστρέφεται στο χρήστη είναι μια ιστοσελίδα html. Όλη τη δουλειά την κάνει ο εξυπηρετητής και όχι κάποιος φυλλομετρητής. Ο πελάτης (χρήστης) από την πλευρά του δεν χρειάζεται κάποιο πρόσθετο εργαλείο ή πρόγραμμα για να δει το αποτέλεσμα της php. Η php είναι μια γλώσσα για την συγγραφή σεναρίων. Αυτό σημαίνει πως ο κώδικας, όπως και της html, δεν χρειάζεται να μεταγλωττιστεί πριν χρησιμοποιηθεί. Η php μπορεί να :

- Πάρει πληροφορίες από φόρμες και να τις χρησιμοποιήσει με διάφορους τρόπους : να την υποθηκεύσει σε μια βάση δεδομένων να δημιουργήσει σελίδες που εξαρτώνται από τα στοιχεία της φόρμας, να τοποθετήσει cookies στο φυλλομετρητή του χρήστη και να στείλει e-mail.
- Πιστοποιήσει την αυθεντικότητα και να εντοπίσει χρήστες.
- Φιλοξενήσει συζητήσεις στην ιστοσελίδα που κατασκευάζουμε.
- Προσαρμόσει την εμφάνιση μιας ιστοσελίδας στις απαιτήσεις που έχουν οι φυλλομετρητές ή οι συσκευές που χρησιμοποιούν οι χρήστες.
- Δημοσιεύσει μια ολόκληρη ιστοσελίδα χρησιμοποιώντας μόνο μια φόρμα σχεδίου (server-side includes-style)
- Εξυπηρετήσει σελίδες xml.

### **1.3.2 Επεκτάσεις αρχείων και διακομιστές**

Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. \*.php, \*.php4, \*.phtml κ.ά.). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να



επεξεργάζεται κώδικα PHP. Ο διακομιστής Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με τα λειτουργικά συστήματα Linux και Microsoft Windows , υποστηρίζει εξ ορισμού επεξεργασία κώδικα PHP.

### **1.3.3 Ιστορία της php**

Η ιστορία της PHP ξεκινά από το 1995, όταν ένας φοιτητής, ο Rasmus Lerdof 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 ιστότοπους 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.

### 1.3.4 Php και βάσεις

Η PHP "συνεργάζεται" τέλεια με βάσεις δεδομένων που υποστηρίζουν επερωτήματα SQL, όπως είναι οι MySQL, Microsoft SQL server, PostgreSQL, Oracle κ.ά. Αυτό επιτρέπει τη δημιουργία αξιόλογων εφαρμογών που επιτρέπουν την αποθήκευση δεδομένων και ανταλλαγή τους μεταξύ online χρηστών. Η σύνδεση σε έναν MySQL server στον ίδιο ηλεκτρονικό υπολογιστή όπου "τρέχει" και ο web server, γίνεται με την εντολή `mysql_connect()` ως εξής:

```
<?php
mysql_connect("localhost", "username", "password");
?>
```

και στη συνέχεια, η κατάλληλη βάση δεδομένων επιλέγεται με την εντολή `mysql_select_db()`:

```
<?php
mysql_select_db("database_name");
?>
```

## 1.4 Structured Query Language (SQL):

### 1.4.1 Ορισμός:

Η Structured Query Language (SQL) αποτελεί τη standard διαδραστική γλώσσα στις σχεσιακές βάσεις δεδομένων. Είναι επίσημη σύσταση τόσο του American National Standards Institute (ANSI), και του International Organization for Standardization (ISO) και πρωτοέκανε την εμφάνιση της

ως Sequel 75 στο System-R της IBM τη δεκαετία του 1970. Τελευταίο κοινά αποδεκτό και επίσημο standard της γλώσσας είναι η SQL/92, ενώ και η SQL-3 με τις object-oriented προεκτάσεις της βρίσκεται πολύ κοντά σε τελική αναγνώριση ως πρότυπο. Παρόλα αυτά, πολλά εμπορικά DBMS υποστηρίζουν και κάποιες επιπλέον προεκτάσεις της SQL.

#### 1.4.2 Γενικά:

Η SQL είναι ειδική στον ορισμό και τη διαχείριση δομημένων δεδομένων μέσω ενός υψηλού επιπέδου interface, που επιτρέπει την επερώτηση και την ανανέωσή τους. Επίσης παρέχει τη δυνατότητα διαχείρισης του σχήματος της βάσης και των τυχόν διαφορετικών όψεων για την υποστήριξη πολλαπλών χρηστών, μέσα από ένα πλούσιο μοντέλο μετα-πληροφορίας. Διαχειρίζεται πολύ καλά την ταυτόχρονη πρόσβαση χρηστών με διαφορετικά δικαιώματα προσπέλασης και διαχείρισης των δεδομένων,

ενώ παρέχει και περιορισμούς ακεραιότητας. Στηρίζεται σε ένα καλά θεμελιωμένο μαθηματικό υπόβαθρο, το οποίο βασίζεται στον κατηγορικό λογισμό πρώτης τάξης (Relational Calculus). Ουσιαστικά η SQL χωρίζεται σε δύο βασικά μέρη, το DDL και το DML:

1. Data Definition Language (DDL): Αποτελεί τη γλώσσα ορισμού της SQL και χρησιμοποιείται για create, drop και alter τις περιγραφές των σχέσεων στη βάση δεδομένων.
2. Data Manipulation Language (DML): Παρέχονται δυνατότητες βασισμένες στη σχεσιακή άλγεβρα, καθώς και δυνατότητες εισαγωγής και διαγραφής στη βάση. Υπάρχουν τέσσερις βασικές εντολές:
  - *select*
  - *insert*
  - *update*
  - *delete*,

Το δε αποτέλεσμα μιας εντολής ή πράξης σε σχέσεις είναι πάντοτε μία καινούρια σχέση. Χρήσιμες λειτουργίες της SQL είναι τα nested queries, η διάταξη των αποτελεσμάτων με βάση την επιθυμία του χρήστη, οι λειτουργίες συνόλων (set Operations), τα aggregate functions (min, max, avg, count, sum) και τα joined relations. Σημαντική διαφορά από την παραπάνω παρουσίαση περί βάσεων αποτελεί το γεγονός ότι στην SQL οι πίνακες δεν είναι σχέσεις, με άλλα λόγια επιτρέπουν την ύπαρξη διπλές πλειάδες.

## **1.5 MySQL**

### **1.5.1 Τι είναι MySQL;**

MySQL, το δημοφιλέστερο σύστημα ανοιχτού κώδικα διαχείρισης βάσεων δεδομένων SQL, αναπτύσσεται, διανέμεται, και υποστηρίζεται από την MySQL AB. Η MySQL AB είναι μια εμπορική επιχείρηση, που ιδρύθηκε από τους υπεύθυνους για την ανάπτυξη της MySQL. Είναι επιχείρηση ανοικτής πηγής δεύτερης γενεάς που ενώνει τις αξίες και τη μεθοδολογία του ανοιχτού κώδικα με ένα επιτυχές επιχειρησιακό πρότυπο. Η MySQL είναι ένα πολύ γρήγορο και δυνατό , σύστημα διαχείρισης βάσεων δεδομένων. Μια βάση δεδομένων σας επιτρέπει να αποθηκεύετε , να αναζητάτε , να ταξινομείτε και να ανακαλείτε τα δεδομένα αποτελεσματικά . Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα σας , για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα , για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση. Συνεπώς η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Χρησιμοποιεί την SQL ( Structured Query Language ) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια . Ο server βάσεων δεδομένων MySQL είναι ο παγκόσμια

δημοφιλέστερος server ανοιχτού -κώδικα. Πάνω από έξι εκατομμύρια εγκαταστάσεις χρησιμοποιούν τον server βάσεων δεδομένων MySQL στους υψηλούς δυναμικούς ιστοχώρους αλλά και σε άλλα κρίσιμα επιχειρησιακά συστήματα συμπεριλαμβανομένων των ηγετών της βιομηχανίας όπως τη NASA, Yahoo, The Associated Press (AP), Suzuki Η MySQL είναι μια ελκυστική εναλλακτική λύση στην υψηλού κόστους, και σύνθετη τεχνολογία βάσεων δεδομένων. Η βραβεύσεις τις στην αξιοπιστία, την εκλεξιμότητα και την ταχύτητά της την κάνουν τη σωστή επιλογή για ένα ευρύ φάσμα εταιρικών τμημάτων , υπεύθυνων για την ανάπτυξη Ιστοχώρων και προμηθευτών λογισμικού. Ο ιστοχώρος MySQL (<http://www.mysql.com/>) παρέχει τις πιο πρόσφατες πληροφορίες για λογισμικό MySQL και πληροφορίες για την MySQL AB.

- Η MySQL είναι ένα σύστημα διαχείρισης βάσεων δεδομένων.

Μια βάση δεδομένων είναι μια δομημένη συλλογή στοιχείων. Μπορεί να είναι οτιδήποτε από έναν απλό κατάλογο αγορών σε μια συλλογή εικόνων ή τα απέραντα ποσά πληροφοριών σε ένα εταιρικό δίκτυο. Για να προσθέσουμε, να έχουμε πρόσβαση, και να υποθηκεύσουμε στοιχεία σε μια βάση δεδομένων υπολογιστών, χρειάζεται ένα σύστημα διαχείρισης βάσεων δεδομένων όπως ο διαμεσολαβητής MySQL. Δεδομένου ότι οι υπολογιστές είναι πολύ καλοί στο χειρισμό των μεγάλων ποσών δεδομένων, τα συστήματα διαχείρισης βάσεων δεδομένων διαδραματίζουν έναν κεντρικό ρόλο στον υπολογισμό, ως αυτόνομες εφαρμογές, ή ως μέρη άλλων εφαρμογών.

- Η MySQL είναι σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων. Μια σχεσιακή βάση δεδομένων αποθηκεύει τα

στοιχεία σε χωριστούς πίνακες σε αντίθεση με άλλα συστήματα που τα τοποθετούν όλα σε μια μεγάλη αποθήκη. Αυτό προσδίδει ταχύτητα και ευελιξία. Το SQL μέρος της "MySQL" αντιπροσωπεύει τη "δομημένη γλώσσα διατύπωσης ερωτήσεων." Η SQL είναι η πιο κοινή τυποποιημένη γλώσσα που χρησιμοποιείται για να υπάρχει πρόσβαση στις βάσεις δεδομένων και καθορίζεται από τα πρότυπα Ansi/Toy ISO SQL. Τα πρότυπα SQL έχουν εξελιχθεί από το 1986 και διάφορες εκδόσεις υπάρχουν..

- MySQL το λογισμικό ανοιχτού κώδικα. Ο ανοιχτός κώδικας σημαίνει ότι είναι δυνατό για καθέναν να χρησιμοποιήσει και να τροποποιήσει το λογισμικό. Οποιοσδήποτε μπορεί να κατεβάσει το λογισμικό MySQL από το Διαδίκτυο και να το χρησιμοποιήσει χωρίς να πληρώσει τίποτα. Εάν επιθυμείτε, μπορείτε να μελετήσετε τον πηγαίο κώδικα και να τον αλλάξετε για να ανταποκριθεί καλύτερα στις ανάγκες σας. Το λογισμικό MySQL χρησιμοποιεί το GPL (άδεια ευρέος κοινού GNU), για να καθορίσει τι μπορείτε και τι δεν μπορείτε να κάνετε με το λογισμικό σε διάφορες περιπτώσεις.
- Ο MySQL server είναι πολύ γρήγορος, αξιόπιστος, και εύχρηστος. Ο MySQL server έχει ένα πρακτικό σύνολο χαρακτηριστικών γνωρισμάτων που αναπτύσσεται σε στενή συνεργασία με τους χρήστες. Ο MySQL sever αναπτύχθηκε αρχικά για να χειριστεί τις μεγάλες βάσεις δεδομένων πολύ γρηγορότερα από τις υπάρχουσες λύσεις και έχει χρησιμοποιηθεί ιδιαίτερα επιτυχώς σε απαιτητικά περιβάλλοντα παραγωγής για αρκετά έτη. Αν και υπό σταθερή ανάπτυξη, ο MySQL server προσφέρει σήμερα ένα πλούσιο και χρήσιμο σύνολο λειτουργιών. Η συνδεσιμότητα, η ταχύτητα, και η

ασφάλειά του, τον καθιστούν ιδιαίτερα αποδοτικό στη πρόσβαση των βάσεων δεδομένων για το διαδίκτυο.

- Ο MySQL server λειτουργεί στον πελάτη/τον εξυπηρετητή υπολογιστή ή σε ενσωματωμένα συστήματα. Το λογισμικό βάσεων δεδομένων MySQL είναι ένα σύστημα πελατών/εξυπηρετητών που αποτελείται από έναν πολύπλοκο εξυπηρετητή SQL που υποστηρίζει διαφορετικά πρότυπα δεδομένων, διαφορετικά προγράμματα και βιβλιοθήκες πελατών, διοικητικά εργαλεία, και ένα ευρύ φάσμα εφαρμογών προγραμματισμού για τον χρήστη (APIs). Ο MySQL server παρέχεται επίσης και σαν ενσωματωμένη πολύπλοκη βιβλιοθήκη που μπορεί να συνδεθεί σε εφαρμογή και να έχουμε έτσι έναν μικρότερο, γρηγορότερο και εύκολο στη διαχείριση αυτόνομο προϊόν.

### 1.5.2 Ιστορία MySQL

Άρχισε με την πρόθεση να συνδεθεί το mSQL σύστημα βάσεων δεδομένων με πίνακές χρησιμοποιώντας τις χαμηλού επιπέδου ρουτίνες (ISAM). Εντούτοις, μετά από κάποιες δοκιμές, βγήκε το συμπέρασμα ότι το mSQL δεν ήταν αρκετά γρήγορο ή αρκετά εύκαμπτο για διάφορες ανάγκες. Αυτό οδήγησε σε νέο χρηστικό περιβάλλον SQL στη βάση δεδομένων αλλά με σχεδόν την ίδια διεπαφή API όπως του mSQL. Αυτό το API είχε ως σκοπό να επιτρέψει τον thirty-party κώδικα που γράφτηκε για χρήση με mSQL για να είναι εύκολα για τη χρήση με MySQL.

Το όνομα MySQL το δώσε ο κατασκευαστής Monty Widenius ,από το όνομα της κόρης του My. Το όνομα του δελφινιού MySQL (το λογότυπό) είναι "Sakila," που επιλέχτηκε από τους ιδρυτές MySQL AB με διαγωνισμό από έναν τεράστιο κατάλογο ονομάτων που προτάθηκαν

από τους χρήστες. Το όνομα νίκης υποβλήθηκε από τον Ambrose Twebaze, υπεύθυνος για την ανάπτυξη λογισμικού ανοιχτού κώδικα από τη Σουαζιλάνδη, Αφρική. Σύμφωνα με Ambrose, το θηλυκό όνομα Sakila έχει τις ρίζες του στη SiSwati, τοπική γλώσσα της Σουαζιλάνδης. Το Sakila είναι επίσης το όνομα μιας πόλης στην Agusha, Τανζανία, κοντά στη χώρα προέλευσης του Ambrose, στην Ουγκάντα.

### 1.5.3 Τα κύρια χαρακτηριστικά γνωρίσματα MySQL

Αυτό το τμήμα περιγράφει μερικών από τα σημαντικά χαρακτηριστικά του λογισμικού βάσεων δεδομένων MySQL.

Εσωτερικά στοιχεία και φορητότητα:

- Γραμμένος σε C και το C ++,
- Δοκιμασμένος με μια ευρεία σειρά διαφορετικών μεταγλωττιστών.
- Λειτουργεί για πολλές διαφορετικές πλατφόρμες.
- Χρησιμοποιεί GNU, Automake, Autoconf, και Libtool για φορητότητα.
- Το σχέδιο του MySQL server είναι πολυστρωματικό με ανεξάρτητες ενότητες.
- Μπορεί να χρησιμοποιήσει πολλαπλές CPU εάν αυτές είναι διαθέσιμες.
- Παρέχει τις συναλλακτικές και μη-συναλλακτικές μηχανές αποθήκευσης.
- Χρησιμοποιεί πολύ γρήγορους πίνακες b-tree (MyISAM) με συμπίεση δεικτών.
- Σχετική ευκολία στο να προσθέσει κανείς άλλες μηχανές αποθήκευσης. Αυτό είναι χρήσιμο εάν θέλετε να παρέχετε μια διεπαφή SQL για μια εσωτερική βάση δεδομένων.



- Ένα πολύ γρήγορο νηματικό σύστημα κατανομής μνήμης.
- Μνημονικοί πίνακες, οι οποίοι χρησιμοποιούνται ως προσωρινοί πίνακες.
- Οι λειτουργίες SQL εφαρμόζονται χρησιμοποιώντας μια ιδιαίτερα βελτιστοποιημένη βιβλιοθήκη και είναι όσο το δυνατόν γρηγορότερες. Συνήθως δεν υπάρχει καθόλου κατανάλωση μνήμης μετά από την έναρξη ερώτησης.
- Ο κώδικας MySQL εξετάζεται με Purify (ένας εμπορικός ανιχνευτής διαρροής μνήμης) καθώς επίσης και με Valgrind, ένα εργαλείο GPL
- Ο server είναι διαθέσιμος ως χωριστό πρόγραμμα για τη χρήση πελάτη/εξυπηρετητή δικτυωμένων υπολογιστών . Είναι επίσης διαθέσιμο ως βιβλιοθήκη που μπορεί να ενσωματωθεί (συνδεδεμένος) στις αυτόνομες εφαρμογές. Τέτοιες εφαρμογές μπορούν να χρησιμοποιηθούν μεμονωμένα ή στα περιβάλλοντα όπου κανένα δίκτυο δεν είναι διαθέσιμο.

#### Τύποι δεδομένων:

- Πολλοί τύποι στοιχείων: υπογεγραμμένοι/ανυπόγραφοι ακέραιοι INTEGERS αριθμοί 1 ..2 ..3 ..4, με 8 bytes μέγεθος FLOAT, DOUBLE CHAR, VARCHAR TEXT, BLOB DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM, και χωρικοί τύποι OpenGIS.
- Καθορισμένου και μεταβλητός-μήκους εγγραφές.

#### Λειτουργίες:

- Πλήρης υποστήριξη χειριστών και λειτουργίας στην εντολή SELECT και WHERE κατά την πρόταση των ερωτήσεων. Παραδείγματος χάριν:

```
mysql > SELECT CONCAT(first name, ' ', last name)
```

```
- > FROM citizen
```

```
- > WHERE income/dependents > 10000 AND age > 30;
```

- Πλήρης υποστήριξη για τις SQL εντολές GROUP BY και ORDER BY. Υποστήριξη λειτουργιών ομαδοποίησης ( COUNT(), COUNT(DISTINCT), AVG(),STD(), SUM() MAX(), MIN(), και GROUP\_CONCAT()).
- Υποστήριξη για LEFT OUTER JOIN και RIGHT OUTER JOIN και με την τυποποιημένη σύνταξη SQL και ODBC.
- Υποστήριξη για τα ψευδώνυμα στους πίνακες και τις στήλες όπως απαιτείται από το τυποποιημένο SQL.
- DELETE INSERT, REPLACE, και UPDATE επιστρέφει τον αριθμό σειρών που άλλαξαν (επηρεασμένες). Είναι δυνατό να επιστραφεί ο αριθμός σειρών που αντιστοιχούνται αντ' αυτού με τη ρύθμιση μιας σημαίας κατά σύνδεση με τον κεντρικό υπολογιστή.
- Η συγκεκριμένη εντολή SHOW μπορεί να χρησιμοποιηθεί για να ανακτήσει τις πληροφορίες για τις βάσεις δεδομένων, τις μηχανές αποθήκευσης, τους πίνακες, και τους δείκτες. Η MySQL 5,0 προσθέτει την υποστήριξη INFORMATION\_SCHEMA βάση δεδομένων, που εφαρμόζεται σύμφωνα με το τυποποιημένο SQL.
- Η EXPLAIN δήλωση μπορεί να χρησιμοποιηθεί για να καθορίσει πώς ο optimizer επιλύει μια ερώτηση.
- Τα ονόματα λειτουργίας δεν διαφωνούν με τα ονόματα πινάκων ή στηλών. Παραδείγματος χάριν, ABS είναι ένα έγκυρο όνομα

στηλών. Ο μόνος περιορισμός είναι ότι για μια κλήση λειτουργίας, κανένα διάστημα δεν επιτρέπεται μεταξύ του ονόματος λειτουργίας και “(” που το ακολουθεί

- Μπορείτε να αναφερθείτε στους πίνακες από τις διαφορετικές βάσεις δεδομένων στην ίδια δήλωση.

#### Ασφάλεια:

- Ένα προνόμιο και σύστημα κωδικού πρόσβασης που είναι πολύ εύκαμπτο και ασφαλές, που επιτρέπει την βασισμένη στον host επαλήθευση.
- Οι κωδικοί πρόσβασης είναι ασφαλείς επειδή όλη η κυκλοφορία κωδικού πρόσβασης κρυπτογραφείται όταν συνδέετε με έναν κεντρικό υπολογιστή.

#### Εξελιξιμότητα και όρια:

- Μεγάλες βάσεις δεδομένων χρησιμοποιούν τον εξυπηρετητή MySQL με βάσεις δεδομένων που περιέχουν 50 εκατομμύριο αρχεία. Υπάρχουν επίσης χρήστες που χρησιμοποιούν τον server MySQL με 60.000 πίνακες και περίπου 5.000.000.000 σειρές.
- Μέχρι 64 δείκτες ανά πίνακα επιτρέπονται (32 πριν από MySQL 4.1.2). Κάθε δείκτης μπορεί να αποτελείται από 1 έως 16 στήλες ή μέρη των στηλών. Το μέγιστο πλάτος δεικτών είναι 1000 bytes (767 για InnoDB) πριν από MySQL 4.1.2, το όριο ήταν 500bytes. Ένας δείκτης μπορεί να χρησιμοποιήσει ένα πρόθεμα μιας στήλης για CHAR VARCHAR, BLOB, ή TEXT τύπους στηλών.

#### Συνδεσιμότητα

- Οι πελάτες μπορούν να συνδεθούν με τον εξυπηρετητή MySQL χρησιμοποιώντας διάφορα πρωτόκολλα:

Οι πελάτες μπορούν να συνδεθούν χρησιμοποιώντας το TCP/IP σε οποιαδήποτε πλατφόρμα. Στα Windows (NT, 2000, XP, 2003, ή vista), οι πελάτες μπορούν να συνδέσουν τη χρησιμοποίηση των named pipes εάν ο εξυπηρετητής αρχίζει με την επιλογή `--enable-named-pipe`. Σε MySQL 4,1 και υψηλότερες εκδόσεις, οι windows servers υποστηρίζουν επίσης τις συνδέσεις shared-memory εάν αρχίζουν με την `--shared-memory` επιλογή. Οι πελάτες μπορούν να συνδεθούν μέσω της shared memory με τη χρησιμοποιώντας την επιλογή `--protocol=memory`. Στα συστήματα Unix, οι πελάτες μπορούν να συνδεθούν τη χρησιμοποιώντας τα αρχεία υποδοχής περιοχών Unix.

- MySQL προγράμματα πελατών μπορούν να γραφτούν σε πολλές γλώσσες. Μια βιβλιοθήκη πελατών που γράφεται σε C είναι διαθέσιμη για τους πελάτες με C ή με C ++, ή για οποιαδήποτε γλώσσα που παρέχει τις συνδέσεις C.
- APIs για την C, τη C ++, Eiffel, Java, Perl, Php, Python, Ruby, και Tcl είναι διαθέσιμα, επιτρέποντας στους πελάτες MySQL να γραφούν σε πολλές γλώσσες.
- Connector/(MyODBC) η διεπαφή ODBC παρέχει ενίσχυση MySQL για τα προγράμματα πελατών που χρησιμοποιούν (ανοικτή συνδεσιμότητα βάσεων δεδομένων) τις συνδέσεις ODBC. Παραδείγματος χάριν, μπορείτε να χρησιμοποιήσετε το MSAccess για να συνδεθείτε με τον κεντρικό υπολογιστή MySQL. Οι πελάτες μπορούν να λειτουργούν στα Windows ή σε συστήματα Unix. Όλες οι λειτουργίες ODBC 2,5 υποστηρίζονται, όπως και πολλές άλλες.

- Η διεπαφή Connector/J παρέχει ενίσχυση MySQL για τα προγράμματα πελατών Java που χρησιμοποιούν τις συνδέσεις JDBC. Οι πελάτες μπορούν να λειτουργήσουν και στα Windows ή σε συστήματα Unix.
- MySQL Connector/net επιτρέπει στους υπεύθυνους για την ανάπτυξη να δημιουργήσουν εύκολα δικτυακές εφαρμογές που απαιτούν ασφάλεια και υψηλής απόδοσης συνδεσιμότητα δεδομένων με MySQL. Εφαρμόζει τις απαραίτητες διεπαφές δικτύου ADO.NET και ενσωματώνει στο δίκτυο ADO.NET τα ενήμερα εργαλεία. Οι υπεύθυνοι για την ανάπτυξη μπορούν να χτίσουν τις εφαρμογές χρησιμοποιώντας γλώσσα δικτύου της επιλογής τους.

#### Προσαρμοστικότητα:

- Ο MySQL server μπορεί να παρέχει τα μηνύματα λάθους στους πελάτες σε πολλές γλώσσες..
- Πλήρης υποστήριξη για διαφορετικά σύνολα χαρακτήρων, συμπεριλαμβανομένου latin1 (cp1252) german, big5 ujis, και πολλά ακόμη. Η υποστήριξη Unicode είναι διαθέσιμη από την MySQL 4,1 και μετά.
- Ολόκληρη η βάση δεδομένων σώζεται σε επιλεγμένο σύνολο χαρακτήρων.
- Η ταξινόμηση και οι συγκρίσεις γίνονται σύμφωνα με το επιλεγμένο σύνολο χαρακτήρων και την παραβολή (να χρησιμοποιήσει latin1 και σουηδική παραβολή εξ ορισμού). Είναι δυνατό να αλλαχτεί αυτό όταν αρχίζει ο MySQL server. Ο MySQL server υποστηρίζει πολλά διαφορετικά σύνολα

χαρακτήρων που μπορούν να διευκρινιστούν συνταθούν στον χρόνο και στον χρόνο εκτέλεσης.

- Από την MySQL 4,1 και ύστερα η χρονική ζώνη των εξυπηρετητών μπορεί να αλλάξει δυναμικά, και οι μεμονωμένοι πελάτες μπορούν να καθορίσουν τη χρονική ζώνη τους.

#### Πελάτες και εργαλεία:

- Η MySQL AB παρέχει διάφορα προγράμματα πελατών. Αυτά περιλαμβάνουν προγράμματα γραμμών-εντολών όπως το **mysqldump** και το **mysqladmin**, και τα γραφικά προγράμματα όπως ο MySQL administrator και MySQL Query Browser.
- Ο εξυπηρετητής MySQL έχει την ενσωματωμένη υποστήριξη για τις δηλώσεις SQL που ελέγχουν, βελτιστοποιούν, και επισκευάζουν τους πίνακες. Αυτές οι δηλώσεις είναι διαθέσιμες από τη γραμμή εντολών μέσω του πελάτη **mysqlcheck**. Ο server MySQL περιλαμβάνει επίσης **myisamchk**, μια πολύ γρήγορη και χρήσιμη εντολή για την εκτέλεση των διαδικασιών MyISAM στους πίνακες.
- Τα MySQL προγράμματα έχουν την επιλογή να ζητηθεί σε απευθείας σύνδεση βοήθεια με τις εντολές `--help` ή `-?`.

#### 1.5.4 Επισκόπηση MySQLAB

Η MySQL AB είναι η επιχείρηση των ιδρυτών της MySQL και των κύριων υπεύθυνων για την ανάπτυξη. Η MySQL AB καθιερώθηκε αρχικά στη Σουηδία από τους David Axmark, Allan Larsson, and Michael "Monty" Widenius. Αφιερώνεται στην ανάπτυξη του λογισμικού βάσεων δεδομένων MySQL και την προώθηση του στους νέους χρήστες.

Η MySQL AB είναι κύρια των πνευματικών δικαιωμάτων στον πηγαίο κώδικα της MySQL, του λογότυπου MySQL και του εμπορικό σήματος.

Οι αξίες της MySQL AB παρουσιάζουν αφιέρωσή στη MySQL και στον ανοικτό κώδικα.

Οι παρακάτω αξίες δείχνουν πως η MySQL AB λειτουργεί με το λογισμικό των MySQL server:

- Να είναι ο καλύτερος και ο ευρύτατα διαδεδομένος server βάσεων δεδομένων στον κόσμο
- Να είναι διαθέσιμος και προσιτός για όλους
- Να είναι εύχρηστος
- Να βελτιώνεται συνεχώς
- Να είναι γρήγορος και ασφαλής
- Να είναι διασκεδαστικός στη χρήση
- Να είναι απαλλαγμένος από λάθη

Ο ιστοχώρος MySQL (<http://www.mysql.com/>) παρέχει τις πιο πρόσφατες πληροφορίες για MySQL και MySQL AB.

Επ' ευκαιρία το "AB" μέρος του ονόματος της επιχείρησης είναι το αρκτικόλεξο για το σουηδική "aktiebolag," ή την "επιχείρηση αποθεμάτων". Μεταφράζετε σε MySQL A.E." Στην πραγματικότητα, MySQL INC και MySQL GmbH είναι παραδείγματα των υποκαταστημάτων MySQL AB, στις Ηνωμένες Πολιτείες και τη Γερμανία, αντίστοιχα.

### 1.5.5 Σύνοψη

Συνοψίζοντας η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Χρησιμοποιεί την SQL ( Structured Query Language ) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια . Ο server βάσεων δεδομένων MySQL είναι ο παγκόσμια δημοφιλέστερος server ανοιχτού -κώδικα. Πάνω από έξι εκατομμύρια εγκαταστάσεις χρησιμοποιούν τον server βάσεων δεδομένων MySQL στους υψηλούς δυναμικούς ιστοχώρους αλλά και σε άλλα κρίσιμα επιχειρησιακά συστήματα. Ο MySQL server προσφέρει διάφορα βασικά πλεονεκτήματα:

Αξιοπιστία και απόδοση. Η MySQL AB καθιστά τις πρώιμες εκδόσεις όλου του λογισμικού βάσεων δεδομένων, διαθέσιμες στην κοινότητα ανοικτή-πηγής για να επιτρέψει αρκετούς μήνες "εξέτασης" τους από διάφορους χρήστες προτού να τις κρίνει έτοιμες για παραγωγή και ανάπτυξη και επέκταση.

Ευκολία στη χρήση και την ανάπτυξη. Η αρχιτεκτονική της MySQL την καθιστά ένα εργαλείο εξαιρετικά γρήγορο και απλό να προσαρμοστεί. Η μοναδική αρχιτεκτονική μηχανών πολύ-αποθήκευσης MySQL δίνει στους χρήστες την ευελιξία που χρειάζονται με ένα πρόγραμμα διαχείρισης βάσεων δεδομένων χωρίς ταίρι στη σταθερότητα, την ταχύτητα, την πυκνότητα, και την ευκολία της επέκτασης.

Ελευθερία από Platform Lock-in. Με την παροχή πρόσβασης στον πηγαίο κώδικα , η MySQL AB με αυτόν τον τρόπο εξασφαλίζει ελευθερία, αποτρέποντας το κλείδωμα της γλώσσας από έναν ενιαίο προμηθευτή ή μια πλατφόρμα.



Πολύ πλατφορμική υποστήριξη. Η MySQL είναι διαθέσιμη σε περισσότερες από είκοσι διαφορετικές πλατφόρμες συμπεριλαμβανομένων όλων των σημαντικών διανομών Linux, του Unix, των Windows της Microsoft, και της MAC OS X.

Εκατομμύρια εκπαιδευμένοι μηχανικοί λογισμικού ασχολούνται για την ανάπτυξη της MySQL. Η MySQL είναι η δημοφιλέστερη παγκοσμίως βάση δεδομένων ανοιχτού κώδικα. Αυτό καθιστά εύκολο να βρεις γνώστες και πεπειραμένους επαγγελματίες για την ανάπτυξη βάσεων δεδομένων.

## 1.6 Κεντρικός υπολογιστής HTTP Apache

### 1.6.1 Γενικά

Ο κεντρικός υπολογιστής HTTP Apache, συνήθως καλούμενος απλά Apache, είναι ένας κεντρικός υπολογιστής δικτύου ξεχωριστός για τη διαδραμάτιση ενός

#### Κεντρικός υπολογιστής HTTP Apache



<b>Υπεύθυνος για την ανάπτυξη:</b>	Ίδρυμα λογισμικού Apache
<b>Πιο πρόσφατη έκδοση:</b>	2.2.6/ 6 Σεπτεμβρίου, 2007
<b>OS:</b>	Cross-platform
<b>Ύφος:</b>	Κεντρικός υπολογιστής δικτύου (server)
<b>Άδεια:</b>	Άδεια Apache
<b>Ιστοχώρος:</b>	<a href="http://httpd.apache.org/">http://httpd.apache.org/</a>

βασικού ρόλου στην αρχική αύξηση του World Wide Web. Το Apache ήταν η πρώτη βιώσιμη εναλλακτική λύση στον κεντρικό υπολογιστή δικτύου της εταιρίας Netscape Communications (αυτήν την περίοδο γνωστό ως Sun Java System Web Server), και έχει εξελιχθεί από τότε για να συναγωνιστεί άλλους βασισμένους σε Unix κεντρικούς υπολογιστές δικτύου από την άποψη της λειτουργίας και της απόδοσης. Από τον Απρίλιο του 1996 ο Apache είναι ο δημοφιλέστερος κεντρικός

υπολογιστής HTTP στο World Wide Web από τον Μάρτιο του 2006 εντούτοις έχει μια σταθερή πτώση του μεριδίου αγοράς του, χάνοντας κυρίως ενάντια στις υπηρεσίες πληροφοριών της Microsoft (Internet Information Services) και της πλατφόρμας .net . Από τον Οκτώβριο του 2007 ο Apache εξυπηρετεί το 47,73% όλων των ιστοχώρων

Το όνομα του προγράμματος επιλέχτηκε για δύο λόγους:

1. από το σεβασμό της εγγενούς αμερικανικής ινδικής φυλής Apache (Indi), γνωστής για την αντοχή τους και τις δεξιότητές τους στην εχθροπραξία,
2. λόγω στις ρίζες του προγράμματος ως σύνολο μπαλωμάτων (patches) στη βάση του κώδικα του NCSA HTTPd 1,3 – δημιουργώντας έναν “patchy” server.

Το Apache αναπτύσσεται και διατηρείται από μια ανοικτή κοινότητα των υπεύθυνων για την ανάπτυξη υπό την αιγίδα του ιδρύματος λογισμικού Apache. Η εφαρμογή είναι διαθέσιμη για μια ευρεία ποικιλία των λειτουργικών συστημάτων συμπεριλαμβανομένων των Windows της Microsoft, Novell NetWare και του Unix καθώς Unix-τύπου λειτουργικά συστήματα όπως FreeBSD, Linux, Solaris και η MAC OS X. Απελευθερωμένο με την άδεια από το ίδρυμα Apache, ότι το Apache είναι ελεύθερο λογισμικό / ανοικτό λογισμικό πηγής.

### **1.6.2 Τι είναι;**

Το πρόγραμμα κεντρικών υπολογιστών HTTP Apache server είναι μια προσπάθεια ανάπτυξης λογισμικού που στοχεύει στη δημιουργία μιας γερής, τρέχουσας, με πολλά χαρακτηριστικά, και ελεύθερα-διαθέσιμης πρόσβασης στον πηγαίο κώδικα εφαρμογής ενός server HTTP (Ιστός). Το πρόγραμμα αυτό ρυθμίζεται από κοινού από μια ομάδα εθελοντών που βρίσκονται σε όλο τον κόσμο, χρησιμοποιώντας το Διαδίκτυο και τον Ιστό για να επικοινωνούν, να προγραμματίζουν, και να αναπτύσσουν

τον server και τα σχετικά έγγραφα του. Αυτό το πρόγραμμα είναι μέρος του ιδρύματος λογισμικού Apache. Επιπλέον, εκατοντάδες χρήστες έχουν συμβάλει ιδέες, κώδικα, και εγχειρίδια.

### 1.6.3 Ιστορία

Η πρώτη έκδοση του κεντρικού υπολογιστή δικτύου Apache δημιουργήθηκε από Robert McCool, ο οποίος αναμίχθηκε σε μεγάλο ποσοστό με το εθνικό κέντρο για supercomputing κεντρικών υπολογιστών δικτύου εφαρμογών (National Center for Supercomputing Applications), γνωστό απλά ως NCSA HTTPd. Όταν ο Mc Cool άφησε το NCSA στα μέσα του 1994, η ανάπτυξη του httpd χρονοτριβεί, αφήνοντας ποικίλα μπαλώματα (patches) για τις βελτιώσεις του να κυκλοφορούν μέσω των ηλεκτρονικών ταχυδρομείων. Ο Rob McCool δεν ήταν μόνος στις προσπάθειές του. Διάφοροι άλλοι υπεύθυνοι για την ανάπτυξη τον βοήθησαν να διαμορφώσουν την αρχική "ομάδα Apache": Brian Behlendorf, Roy T. Fielding, Rob Hartill, David Robinson, Cliff Skolnick, Randy Terbush, Robert S. Thau, Andrew Wilson, Eric Hagberg, Frank Peters, and Nicolas Pioch.

Η έκδοση 2 του server Apache ήταν μια ουσιαστική αντιγραφή ενός μεγάλου μέρους της έκδοσης 1.x Apache, με μια ισχυρή εστίαση στην περαιτέρω ρύθμιση και την ανάπτυξη ενός επιπέδου φορητότητας, Apache Portable Runtime. Ο πυρήνας του Apache 2.x έχει διάφορες σημαντικές αυξήσεις πέρα από Apache 1.x. Αυτές περιλαμβάνουν το UNIX threading, την καλύτερη υποστήριξη για τις πλατφόρμες μη-Unix (Microsoft Windows), ένα νέο Apache API, και την υποστήριξη IPv6. Η πρώτη ολοκληρωμένη έκδοση Apache 2 ήταν τον Μάρτιο του 2000. Η έκδοση 2.2 εισήγαγε μια νέα API που επιτρέπει περισσότερη ευελιξία. Υποστηρίζει επίσης τις βελτιωμένες ενότητες κρύπτης (cache modules) και τις ενότητες πληρεξούσιου (proxy modules).

### 1.6.4 Ομάδα Ανάπτυξης

Υπάρχει μια βασική ομάδα συνεισφερόντων, που απαρτίζεται αρχικά από τους ιδρυτές του προγράμματος, και που αυξάνεται κατά διαστήματα από άλλους σημαντικούς συνεισφέροντες. Υπάρχουν χρήστες οι "committers"(αφοσιωμένοι), που έχουν πρόσβαση στα repositories ελέγχου κώδικα πηγής για να διατηρήσουν το πρόγραμμα ή τα έγγραφα, και η βασική ομάδα που διαχειρίζεται τώρα το πρόγραμμα, το οποίο καλείται Επιτροπή διαχείρισης του προγράμματος HTTP Apache (PMC). Στην πραγματικότητα, κάθε εργασία του ιδρύματος λογισμικού Apache έχει το δικό του PMC , για να καθορίσει τους committers, την κατεύθυνση του προγράμματος και τη γενική διαχείριση.

### 1.6.5 Χαρακτηριστικά γνωρίσματα

Ο Apache υποστηρίζει ποικίλα χαρακτηριστικά γνωρίσματα, πολλά που εφαρμόζονται ως συνταγμένες ενότητες που επεκτείνουν τη λειτουργία του πυρήνα του.

Αυτά μπορούν να κυμανθούν από την υποστήριξη server-side προγραμματισμού έως επικύρωση σχημάτων (schemes). Υποστηρίζει αρκετές διεπαφές κοινής γλώσσας προγραμματισμού όπως mod\_perl, mod\_python, Tcl, και php. Περιλαμβάνονται και οι δημοφιλείς ενότητες επικύρωσης mod\_access, mod\_auth, και mod\_digest.

Ένα δείγμα άλλων χαρακτηριστικών γνωρισμάτων που περιλαμβάνει είναι υποστήριξη SSL και TLS (mod\_ssl), μια ενότητα proxy, ένα χρήσιμο URL rewriter (επίσης γνωστός ως rewrite engine, που εφαρμόζεται κάτω από το mod\_rewrite), αρχεία καταγραφής (mod\_log\_config), και υποστήριξη φιλτραρίσματος (mod\_include και mod\_ext\_filter). Τα αρχεία καταγραφής του Apache μπορούν να αναλυθούν μέσω ενός φυλλομετρητή Ιστού χρησιμοποιώντας τα ελεύθερα scripts όπως AWStats/W3Perl ή Visitors

Η εικονική φιλοξενία (virtual hosting) επιτρέπει με έναν Apache server να εξυπηρετήσει πολλούς διαφορετικούς πραγματικούς ιστοχώρους. Παραδείγματος χάριν, ένας υπολογιστής, με μια εγκατάσταση Apache θα μπορούσε ταυτόχρονα να εξυπηρετήσει το `www.example.com`, το `www.test.com`, `test47.test-server.test.com`, κλπ...

Το Apache υποστηρίζει επίσης τη διαμόρφωση στα μηνύματα λάθους, DBMS (πρόγραμμα διαχείρισης βάσεων δεδομένων)- βάσεις δεδομένων επικύρωσης, και διαπραγμάτευση περιεχομένου (content negotiation). Ακόμα συνεργάζεται με διάφορες γραφικές διεπαφές με τον χρήστη (GUIs) που επιτρέπουν την ευκολότερη, και πιο διαισθητική διαμόρφωση του server.

### 1.6.6 Χρήση

Ο Apache χρησιμοποιείται πρώτιστα για να εξυπηρετήσει και στατικές και δυναμικές ιστοσελίδες στο World Wide Web. Πολλές εφαρμογές Ιστού (web applications) σχεδιάζονται αναμένοντας το περιβάλλον και τα χαρακτηριστικά γνωρίσματα που ο Apache παρέχει. Ο Apache server είναι τμήμα του δημοφιλούς πακέτου εφαρμογών κεντρικών υπολογιστών XAMPP, παράλληλα με MySQL, και τις γλώσσες προγραμματισμού Php/Perl /Python. Ο Apache ανακατανέμεται ως τμήμα των διάφορων ιδιόκτητων πακέτων λογισμικού συμπεριλαμβανομένης της βάσης δεδομένων Oracle ή του IBM WebSphere server. Η MAC OS X ενσωματώνει τον Apache ως κεντρικό server της και ως υποστήριξη για τον server εφαρμογών WebObjects . Υποστηρίζεται επίσης με κάποιο τρόπο από την Borland στα εργαλεία ανάπτυξης Kylix και Delphi. Ο Apache συμπεριλαμβάνεται και με το Novell NetWare 6.5, όπου είναι ο προεπιλεγμένος κεντρικός υπολογιστής δικτύου. Ο Apache χρησιμοποιείται και για πολλούς άλλους σκοπούς όταν χρειάζεται περιεχόμενο να παρασχεθεί με έναν ασφαλή και

αξιόπιστο τρόπο. Ένα παράδειγμα είναι όταν τα αρχεία από έναν προσωπικό υπολογιστή μοιράζονται μέσω του Διαδικτύου. Ένας χρήστης που έχει εγκατεστημένο τον Apache στον υπολογιστή του μπορεί να βάλει τα αυθαίρετα αρχεία στο φάκελο αρχείων του Apache και έπειτα μπορεί να τα μοιραστεί μέσω του διαδικτύου. Οι προγραμματιστές που αναπτύσσουν εφαρμογές Ιστού χρησιμοποιούν συχνά μια τοπικά εγκατεστημένη έκδοση Apache προκειμένου να δοκιμάσουν και να προεπισκοπήσουν τον κώδικα που δημιουργούν κατά την φάση της ανάπτυξης του. Οι υπηρεσίες πληροφοριών Διαδίκτυο της Microsoft (IIS) είναι ο κύριος ανταγωνιστής του Apache, που συνεργάζεται με τη Sun Microsystems' την Sun Java System Web Server και ένα πλήθος άλλων εφαρμογών όπως Zeus Web Server

### 1.6.7 Άδεια

Η άδεια λογισμικού κάτω από την οποία το λογισμικό διανέμεται από το ίδρυμα Apache είναι ένα διακριτικό μέρος της ιστορίας και της παρουσίας του κεντρικού υπολογιστή HTTP Apache στην κοινότητα λογισμικού ανοιχτού κώδικα . Η άδεια του Apache επιτρέπει τη διανομή και των ανοικτών και κλειστών παράγωγων του πηγαίου κώδικα. Το ίδρυμα ελεύθερου λογισμικού δεν θεωρεί την άδεια του Apache συμβατή με την δεύτερη έκδοση του GNU Γενική Άδεια Δημόσιας Χρήσης (GPL). Σε εκείνα τα λογισμικά που χορηγούνται με άδεια Apache δεν μπορεί να ενσωματωθεί λογισμικό που διανέμεται κάτω από το GPL: Αυτή είναι μια ελεύθερη άδεια λογισμικού αλλά είναι ασυμβίβαστη με το GPL. Η άδεια λογισμικού Apache είναι ασυμβίβαστη με το GPL επειδή έχει μια συγκεκριμένη απαίτηση που δεν είναι στο GPL: έχει ορισμένες περιπτώσεις κατοχυρωμένης ευρεσιτεχνίας που το GPL δεν απαιτεί. Δεν σκεφτόμαστε ότι οι κατοχυρωμένες πατέντες είναι εγγενώς μια κακή ιδέα, αλλά εν τούτοις είναι ασυμβίβαστες με το GNU

GPL.(<http://www.gnu.org/philosophy/license-list.html>)Εντούτοις, η έκδοση 3 του GPL περιλαμβάνει μια διάταξη (τμήμα 7e) που επιτρέπει στο apache να είναι συμβατό με τις άδειες που έχουν τις προτάσεις κατοχύρωσης ευρεσιτεχνίας, συμπεριλαμβανομένης και της άδειας του Apache. Το όνομα Apache είναι ένα καταχωρημένο εμπορικό σήμα και μπορεί μόνο να χρησιμοποιηθεί με τη σαφή άδεια του κατόχου εμπορικών σημάτων.

## **ΚΕΦΑΛΑΙΟ 2**

### **2.1 Τεχνολογία λογισμικού και κύκλος ζωής ανάπτυξης λογισμικού**

#### **2.1.1 Γενικά**

Είναι γεγονός ότι όλες οι αναπτυγμένες χώρες σήμερα βασίζονται σε πολύπλοκα υπολογιστικά συστήματα για την ανάπτυξη των προϊόντων τους. Το λογισμικό σε αυτά τα συστήματα αντιπροσωπεύει ένα ολοένα και αυξανόμενο ποσοστό του συνολικού κόστους του συστήματος. Ως εκ τούτου, η κατασκευή λογισμικού με ένα οικονομικά αποδοτικό τρόπο είναι απαραίτητη για τη λειτουργία εθνικών και διεθνών οικονομιών. Η Τεχνολογία Λογισμικού λοιπόν έχει ως στόχο την οικονομικά αποδοτική ανάπτυξη συστημάτων λογισμικού. Το λογισμικό δεν περιορίζεται από υλικά, δεν κυβερνάται από φυσικούς νόμους ή από κατασκευαστικές διαδικασίες. Κατά κάποιο τρόπο, αυτό απλοποιεί την τεχνολογία λογισμικού, αφού δεν υπάρχουν φυσικά όρια για την ανάπτυξη λογισμικού. Αυτή όμως η έλλειψη φυσικών ορίων σημαίνει ότι το λογισμικό μπορεί εύκολα να γίνει εξαιρετικά πολύπλοκο και συνεπώς δύσκολο στην κατανόηση. Η αρχική εμπειρία στην κατασκευή συστημάτων λογισμικού έδειξε ότι μία ανεπίσημη προσέγγιση για την ανάπτυξη λογισμικού δεν ήταν αρκετά καλή. Μεγάλες εργασίες

χρειάζονταν χρόνια για να διεκπεραιωθούν, ενώ τα κόστη υπερέβαιναν τα αρχικά προβλεπόμενα. Η κρίση αυτή στην ανάπτυξη λογισμικού οδήγησε στην ανάπτυξη νέων τεχνικών, έτσι ώστε να ελεγχθεί η έμφυτη πολυπλοκότητα στα μεγάλα συστήματα λογισμικού. Με τον τρόπο αυτό, η εξέλιξη της τεχνολογίας λογισμικού έχει βελτιώσει κατά πολύ το ίδιο το λογισμικό. Τώρα πια έχουμε πολύ καλύτερη κατανόηση των δραστηριοτήτων που εμπλέκονται στην ανάπτυξη λογισμικού και έχουμε αναπτύξει αποτελεσματικές μεθόδους για τον προσδιορισμό, το σχεδιασμό και την εφαρμογή του λογισμικού.

### 2.1.2 Βασικές έννοιες

Ας δούμε όμως στο σημείο αυτό κάποιες βασικές έννοιες που θα μας εισάγουν γενικά στο θέμα της Τεχνολογίας Λογισμικού, αλλά και ειδικότερα στην εφαρμογή που θα περιγράψουμε στη συνέχεια της εργασίας αυτής.

**Λογισμικό:** Ενώ πολλοί άνθρωποι ταυτίζουν την έννοια του λογισμικού με τα προγράμματα υπολογιστών, το λογισμικό δεν είναι μόνο τα προγράμματα αλλά και η σχετιζόμενη τεκμηρίωση και τα διαμορφωμένα δεδομένα που είναι απαραίτητα για να λειτουργούν τα προγράμματα σωστά. Ένα σύστημα λογισμικού συνήθως αποτελείται από έναν αριθμό διαφορετικών προγραμμάτων, αρχεία για την ενοποίηση των προγραμμάτων αυτών, τεκμηρίωση του συστήματος που περιγράφει τη δομή του συστήματος και τεκμηρίωση του χρήστη που εξηγεί πως χρησιμοποιείται το σύστημα και, για προϊόντα λογισμικού, ιστοσελίδες για τους χρήστες που επιθυμούν να λαμβάνουν νέες πληροφορίες για το προϊόν. Το λογισμικό χωρίζεται σε δύο επιμέρους κατηγορίες. Στην πρώτη ανήκουν τα αυτόνομα συστήματα που κατασκευάζονται από έναν οργανισμό ανάπτυξης λογισμικού και είναι διαθέσιμα στην αγορά για όποιον πελάτη έχει την επιθυμία και την ικανότητα να τα αγοράσει. Στην



δεύτερη ανήκουν τα συστήματα, η κατασκευή των οποίων ανατίθεται σε ένα εργολήπτη από ένα συγκεκριμένο πελάτη. Το λογισμικό διαμορφώνεται ακριβώς με βάση τις απαιτήσεις του πελάτη. Η κυριότερη διαφορά μεταξύ των δύο τύπων λογισμικού είναι το γεγονός ότι στα εμπορικά προϊόντα ο οργανισμός που κατασκευάζει το λογισμικό ελέγχει τις προδιαγραφές του, ενώ στη δεύτερη κατηγορία λογισμικού οι προδιαγραφές του καθορίζονται και ελέγχονται από τον ίδιο τον οργανισμό που αγοράζει το λογισμικό. Για την ανάπτυξη του λογισμικού υπάρχουν οι προγραμματιστές, οι οποίοι ασχολούνται με τις λεπτομέρειες της υλοποίησης, της πακετοποίησης και της αλλαγής αλγορίθμων και δομών δεδομένων που έχουν γραφτεί σε συγκεκριμένη γλώσσα προγραμματισμού.

**Τεχνολογία Λογισμικού:** Η Τεχνολογία Λογισμικού είναι μία διαδικασία που ασχολείται με όλους τους τομείς της κατασκευής λογισμικού από τα πρώτα στάδια του προσδιορισμού του συστήματος μέχρι τη συντήρηση του συστήματος αφού αυτό έχει μπει σε λειτουργία. Αυτός που εμπλέκεται στην διαδικασία της Τεχνολογίας Λογισμικού είναι ο μηχανικός λογισμικού, ο οποίος, πέρα της δουλειάς του προγραμματιστή, ασχολείται με θέματα ανάλυσης συστημάτων, σχεδιασμού, επαλήθευσης, τεκμηρίωσης, συντήρησης λογισμικού και διαχείρισης έργου. Οι μηχανικοί λογισμικού είναι αυτοί που, με λίγα λόγια, κάνουν τα πράγματα να δουλεύουν. Εφαρμόζουν θεωρίες, μεθόδους και εργαλεία, όπου αυτά είναι απαραίτητα, αλλά τα χρησιμοποιούν επιλεκτικά και πάντα προσπαθούν να βρίσκουν λύσεις σε προβλήματα, ακόμα και όταν δεν υπάρχουν εφαρμόσιμες θεωρίες και μέθοδοι για να υποστηρίξουν την προσπάθειά τους. Οι μηχανικοί, επίσης, αναγνωρίζουν το γεγονός ότι πρέπει να δουλεύουν μέσα σε οργανωτικά και οικονομικά όρια, κι έτσι αναζητούν λύσεις μέσα στα όρια αυτά.

### 2.1.3 Βήματα ανάπτυξης λογισμικού

Η Τεχνολογία Λογισμικού δεν ασχολείται μόνο με τις τεχνικές διαδικασίες της ανάπτυξης του λογισμικού, αλλά και με δραστηριότητες όπως η διαχείριση έργου και με την ανάπτυξη εργαλείων, μεθόδων και θεωριών για την υποστήριξη της κατασκευής του λογισμικού. Γενικά, οι μηχανικοί λογισμικού υιοθετούν μία συστηματική και οργανωμένη προσέγγιση για τη δουλειά τους, αφού αυτός είναι συνήθως ο πιο αποτελεσματικός τρόπος να παράγουν υψηλής ποιότητας λογισμικό. Για την ανάπτυξη του απαιτούμενου λογισμικού πρέπει να ακολουθήσει ο κάθε μηχανικός λογισμικού κάποια βασικά στάδια για να φτάσει στο τελικό επιθυμητό προϊόν. Τα γενικά αυτά στάδια φαίνονται παρακάτω:

- Καθορισμός της λειτουργικότητας του λογισμικού και των προδιαγραφών για τη λειτουργία του.
- Κατασκευή του λογισμικού που ανταποκρίνεται στις δοθείσες προδιαγραφές. Εδώ περιλαμβάνεται η ανάπτυξη της λογικής που πρέπει να χρησιμοποιηθεί και η υλοποίηση των προγραμμάτων.
- Έλεγχος του λογισμικού. Στον έλεγχο εντάσσεται η επαλήθευση και η επικύρωση, το αν δηλαδή το προϊόν έχει αναπτυχθεί σωστά και το αν έχει αναπτυχθεί το σωστό προϊόν αντίστοιχα.
- Τεκμηρίωση του λογισμικού που έχει αναπτυχθεί. Όταν μιλάμε για τεκμηρίωση εννοούμε την εξήγηση των χαρακτηριστικών ενός εγγράφου. Υπάρχουν και δύο άλλοι τύποι τεκμηρίωσης, η εσωτερική και η εξωτερική τεκμηρίωση. Η εσωτερική τεκμηρίωση του πηγαίου κώδικα περιγράφει τα χαρακτηριστικά του κώδικα, ενώ η εξωτερική τεκμηρίωση εξηγεί τα χαρακτηριστικά των εγγράφων που συνοδεύουν τον κώδικα.
- Συντήρηση των προγραμμάτων, έτσι ώστε να ανταποκρίνονται στις μεταβαλλόμενες ανάγκες των πελατών.

#### **2.1.4 Κύκλος ζωής**

Θα πρέπει να σημειωθεί ότι, αντίθετα από ότι θα περίμενε κανείς, το μεγαλύτερο χρονικό και οικονομικό κόστος κατά τον όλο κύκλο ζωής της ανάπτυξης του λογισμικού το απαιτεί ο έλεγχος και η συντήρηση του λογισμικού, ενώ πολύ μικρότερο ποσοστό του κόστους καταλαμβάνει η εν γένει ανάπτυξη του συστήματος. Υπάρχουν διάφορα μοντέλα για την περιγραφή του κύκλου ζωής της ανάπτυξης του λογισμικού, όπως ο Ανιχνευτικός Προγραμματισμός, η Ανάπτυξη Πρωτοτύπου και η Συναρμολόγηση Συστήματος από Επαναχρησιμοποιούμενα Κομμάτια.

#### **2.1.5 Μοντέλο του καταρράκτη**

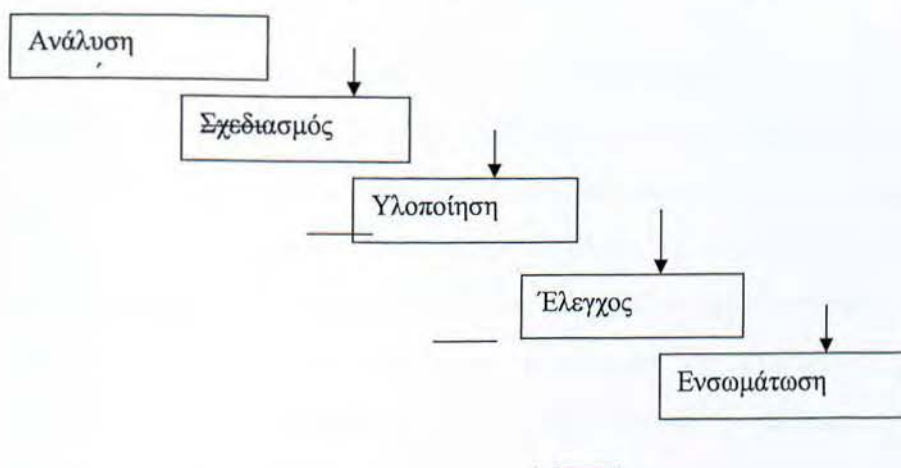
Το μοντέλο για τον κύκλο ζωής της ανάπτυξης του λογισμικού που θα χρησιμοποιήσουμε στην εφαρμογή που θα αναπτύξουμε είναι το μοντέλο του Καταρράκτη. Η διαγραμματική απεικόνιση του μοντέλου φαίνεται στην επόμενη σελίδα.

Ας δούμε όμως αναλυτικά τις εργασίες που γίνονται σε κάθε στάδιο του κύκλου ζωής:

1. Ανάλυση απαιτήσεων: Μέσω συμβουλευτικής διαδικασίας με τους χρήστες του συστήματος προσδιορίζονται οι υπηρεσίες που θα παρέχει το σύστημα, οι περιορισμοί και οι στόχοι του.
2. Σχεδιασμός συστήματος και λογισμικού: Η διαδικασία σχεδιασμού του συστήματος χωρίζει τις απαιτήσεις σε απαιτήσεις υλικού και απαιτήσεις λογισμικού. Εγκαθιδρύει μία γενική αρχιτεκτονική συστήματος. Ο σχεδιασμός του λογισμικού περιλαμβάνει την αναγνώριση και την περιγραφή των σημαντικών μερών του συστήματος λογισμικού και των σχέσεων μεταξύ τους.
3. Υλοποίηση και έλεγχος ενοτήτων: Κατά τη διάρκεια αυτού του σταδίου, ο σχεδιασμός του λογισμικού γίνεται αντιληπτός ως ένα σύνολο προγραμμάτων ή ενοτήτων προγραμμάτων. Ο έλεγχος των

ενοτήτων περιλαμβάνει την επαλήθευση του γεγονότος ότι κάθε ενότητα εκπληρώνει τις προδιαγραφές της.

4. Ενσωμάτωση και έλεγχος συστήματος: Οι ανεξάρτητες ενότητες των προγραμμάτων ή τα ίδια τα διαφορετικά προγράμματα συγχωνεύονται και ελέγχονται σαν ένα ολοκληρωμένο σύστημα για να διασφαλιστεί ότι έχουν εκπληρωθεί οι απαιτήσεις του λογισμικού. Μετά τον έλεγχο, το σύστημα λογισμικού παραδίδεται στον πελάτη.
5. Λειτουργία και συντήρηση: Κανονικά, αλλά όχι απαραίτητα, αυτή είναι η πιο χρονοβόρα φάση του κύκλου ζωής. Το σύστημα εγκαθίσταται και μπαίνει σε χρήση στην πράξη. Η συντήρηση περιλαμβάνει τη διόρθωση λαθών που δεν έγιναν αντιληπτά σε νωρίτερα στάδια του κύκλου ζωής, βελτιώνοντας την εφαρμογή των διαφόρων ενοτήτων του συστήματος και τις υπηρεσίες που προσφέρει το σύστημα, αφού ανακαλύπτονται νέες απαιτήσεις.



Εικόνα 2.1.5 Α : Κύκλος ζωής λογισμικού – Μοντέλο καταρράκτη

Σύμφωνα με τις βασικές αρχές, το αποτέλεσμα κάθε φάσης είναι ένα ή περισσότερα έγγραφα που πρέπει να περάσουν τη διαδικασία της έγκρισης. Η επόμενη φάση δε θα μπορούσε να ξεκινήσει στην περίπτωση αυτή αν η προηγούμενη δεν είχε ολοκληρωθεί. Στην πράξη αυτά τα

στάδια υπερκαλύπτονται και ανταλλάσσουν πληροφορίες. Κατά τη διάρκεια του σχεδιασμού ανακύπτουν προβλήματα σε σχέση με τις απαιτήσεις, κατά τη διάρκεια της κατασκευής του πηγαιού κώδικα βρίσκονται προβλήματα, κτλ. Η ανάπτυξη του λογισμικού δεν είναι ένα απλό γραμμικό μοντέλο αλλά περιλαμβάνει μία ακολουθία επαναλήψεων των δραστηριοτήτων εξέλιξης. Λόγω του κόστους της κατασκευής και της έγκρισης εγγράφων, οι επαναλήψεις έχουν μεγάλο κόστος και απαιτούν σημαντική επαναληπτική εργασία. Ως εκ τούτου, μετά από ένα μικρό αριθμό επαναλήψεων, είναι φυσιολογικό να «παγώνουν» τμήματα της διαδικασίας, όπως ο καθορισμός προδιαγραφών, και να συνεχίζεται η διαδικασία στα επόμενα στάδια. Πιθανά προβλήματα αφήνονται για μεταγενέστερη επίλυση ή αγνοούνται. Το πρόωρο πάγωμα των απαιτήσεων μπορεί να σημαίνει ότι το σύστημα δεν θα κάνει αυτό που θέλει ο χρήστης. Μπορεί επίσης να οδηγήσει σε άσχημα δομημένα συστήματα, μιας και τα σχεδιαστικά προβλήματα παρακάμπτονται με κόλπα στην εφαρμογή.

Κατά τη διάρκεια της τελευταίας φάσης του κύκλου ζωής το λογισμικό μπαίνει σε λειτουργία. Λάθη και παραλείψεις στις αυθεντικές απαιτήσεις του λογισμικού ανακαλύπτονται. Προγραμματιστικά και σχεδιαστικά λάθη ενοποιούνται και αναγνωρίζεται η ανάγκη για νέα λειτουργικότητα. Το σύστημα λοιπόν πρέπει να εξελιχθεί για να παραμείνει χρήσιμο. Το να γίνουν όμως οι απαραίτητες αλλαγές (συντήρηση του λογισμικού) μπορεί να σημαίνει την επανάληψη κάποιων ή όλων των προγενέστερων φάσεων της διαδικασίας. Επειδή το Μοντέλο του Καταρράκτη ανταποκρίνεται δύσκολα στις αλλαγές των απαιτήσεων του πελάτη (αφού χωρίζει το έργο σε σχετικά δύσκαμπτα στάδια) θα πρέπει στην αρχή της διαδικασίας να καθοριστούν σαφώς και να κατανοηθούν πλήρως οι απαιτήσεις. Έτσι τα λάθη και οι παραλείψεις θα τείνουν να ελαχιστοποιηθούν. Επίσης θα παρουσιάζονται αρχικά οι σχετικές με το αντίστοιχο θέμα μεθοδολογίες

και στη συνέχεια θα γίνεται πλήρης ανάλυση της εφαρμογής των μεθοδολογιών αυτών στην πράξη κατά τη διαδικασία της κατασκευής του συστήματος διαχείρισης των εργασιών της εφαρμογής.

## **2.2 Ανάλυση απαιτήσεων**

Σε αρκετές περιπτώσεις, τα προβλήματα που πρέπει να επιλύσουν οι μηχανικοί λογισμικού είναι πολύ πολύπλοκα. Η κατανόηση της φύσης του προβλήματος μπορεί να είναι πολύ δύσκολη, ειδικά αν το σύστημα είναι καινούριο, όπως το σύστημα που θα κατασκευάσουμε στην παρούσα εφαρμογή για την ηλεκτρονική αγορά εισητηριών. Συνεπώς είναι δύσκολο να καθορίσουμε ακριβώς τι θα πρέπει να κάνει το σύστημα. Οι περιγραφές των υπηρεσιών και των περιορισμών είναι οι απαιτήσεις για το σύστημα και η διαδικασία της εύρεσης, της ανάλυσης, της τεκμηρίωσης και του ελέγχου αυτών των υπηρεσιών και των περιορισμών ονομάζεται τεχνολογία απαιτήσεων.

Ο όρος *απαίτηση* δεν χρησιμοποιείται με τον ίδιο τρόπο σε όλη τη βιομηχανία λογισμικού. Σε μερικές περιπτώσεις, μία απαίτηση αντιμετωπίζεται σαν μία υψηλού επιπέδου εισαγωγική δήλωση της υπηρεσίας που ένα να παρέχει ένα σύστημα ή σαν ένας περιορισμός του συστήματος. Από την άλλη πλευρά, είναι μία μαθηματικά επίσημη εξήγηση της λειτουργίας του συστήματος. Στην εφαρμογή που θα περιγραφεί στη συνέχεια θα δούμε ότι και οι δύο αντιλήψεις είναι αποδεκτές και εφαρμόζονται από κοινού για την επαρκή κατανόηση των στόχων του συστήματος και την επιτυχή υλοποίησή του στη συνέχεια.

### **2.2.1 Είδη απαιτήσεων**

Κάποια προβλήματα που προκύπτουν κατά τη διαδικασία προσδιορισμού των απαιτήσεων του συστήματος είναι αποτέλεσμα της αποτυχίας στο να

διαχειριστές έργου. Και πάλι θα χρησιμοποιείται από προσωπικό και του πελάτη και του εργολήπτη. Οι τελικοί χρήστες του συστήματος μπορούν να διαβάσουν και τα δύο αυτά έγγραφα. Τέλος, οι προδιαγραφές του σχεδιασμού λογισμικού είναι ένα έγγραφο που κατευθύνεται προς την υλοποίηση του συστήματος και γράφεται για του μηχανικούς λογισμικού που θα κατασκευάσουν το σύστημα.

### **2.2.2 Κατηγορίες απαιτήσεων**

Πριν προχωρήσουμε στην ανάλυση του τι πρέπει να περιέχει ένα έγγραφο προσδιορισμού απαιτήσεων θα κάνουμε μία ακόμα κατηγοριοποίηση των απαιτήσεων για ένα σύστημα η οποία, αν και καθαρά υποθετική, μπορεί να μας βοηθήσει στην κατανόηση των εκάστοτε απαιτήσεων:

- **Λειτουργικές απαιτήσεις:** Είναι οι δηλώσεις των υπηρεσιών που πρέπει να παρέχει το σύστημα, πως το σύστημα θα πρέπει να αντιδράσει με την εισαγωγή συγκεκριμένων δεδομένων και πως το σύστημα θα πρέπει να συμπεριφέρεται σε συγκεκριμένες καταστάσεις. Σε μερικές περιπτώσεις οι λειτουργικές απαιτήσεις μπορεί επίσης να εκφράζουν σαφώς τι δεν θα πρέπει να κάνει το σύστημα.
- **Μη-λειτουργικές απαιτήσεις:** Είναι οι περιορισμοί πάνω στις υπηρεσίες ή στις λειτουργίες που θα παρέχει το σύστημα. Συμπεριλαμβάνουν χρονικούς περιορισμούς, περιορισμούς πάνω στη διαδικασία της ανάπτυξης, διάφορα στάνταρ, κλπ.
- **Απαιτήσεις τομέα:** Είναι απαιτήσεις που προέρχονται από τον τομέα της εφαρμογής του συστήματος και που αντικατοπτρίζουν

χαρακτηριστικά αυτού του τομέα. Μπορεί να είναι είτε λειτουργικές είτε μη-λειτουργικές απαιτήσεις.

### **2.3 Περιγραφή του εφαρμογής**

Σκοπός αυτής της ενότητας δεν είναι να μάθει στον χρήστη να <<γράφει κώδικα>> αλλά να κάνει μια εισαγωγή στην φιλοσοφία της εφαρμογής και στην λογική του πλάνου.

Για την υλοποίηση του project η μόνη παραδοχή που έγινε είναι η δημιουργία του κεντρικού διαχειριστή. Είναι η μόνη εμπλοκή μας στην βάση δεδομένων και χρειάζεται για την περαιτέρω λειτουργία της εφαρμογής.



Μέρος Α : Η διαδικασία για την εγγραφή νέου χρήστη.

The image shows a registration form titled "St\_Proj" with a decorative header image of raspberries. The form contains the following fields and values:

- FirstName : Γεώργιος
- SecondName : Ιωάννου
- Username : giiwa
- Password : .....
- Confirm Password : .....
- Email address : g.iwa@gmail.com

Below the fields is a "register" button. A white arrow points from the bottom right towards the "register" button.

Εικ 1 : Αρχικά, ο χρήστης καλείται να συμπληρώσει όλα τα στοιχεία που του ζητούνται στα κενά και ύστερα να προχωρήσει.

```
<?php
$a=$_POST['username'];
$b=( $_POST['password'] );
$c=( $_POST['password2'] );
$d=$_POST['FirstName'];
$e=$_POST['SecondName'];
$f=$_POST['email'];

if ( !$a==" " && !$b==" " && !$c==" " && !$d==" " && !$e==" " && !$f==" " )
{

if ($b==$c)
```

Εικ 2: Τα στοιχεία που έχει τοποθετήσει ο χρήστης περνάνε στην επόμενη σελίδα και αφού οι προϋποθέσεις ξεκινά η εγγραφή στην βάση δεδομένων.

```
$host="localhost";  
$username="root";  
$password="";  
$schema="πτυχιaki";  
$table="login";  
  
mysql_connect("$host", "$username", "$password") or die(mysql_error());  
mysql_select_db("$schema") or die(mysql_error());  
$query="SELECT * FROM {$table} WHERE username='{$a}'";  
$results = mysql_query($query) or die(mysql_error());  
//Γίνεται έλεγχος για το εάν το όνομα χρήστη είναι μοναδικό.  
$row=mysql_num_rows($results);
```

Εικ 3 : Είσοδος στην MySQL βάση δεδομένων και έλεγχος για το εάν το όνομα του χρήστη δεν είναι δεσμευμένο.

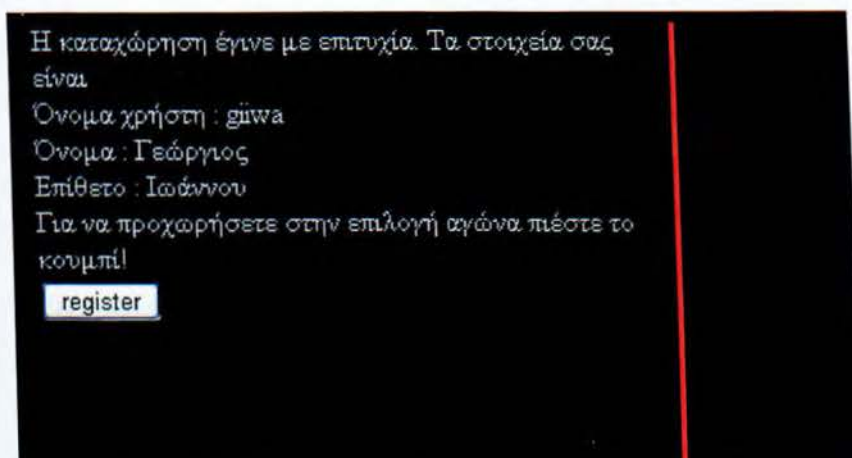
```

if($row==) //Εάν η τιμή είναι 0 τότε δεν υπάρχει //
//ίδιο όνομα χρήστη και άρα μπορεί να γίνει η εγγραφή//

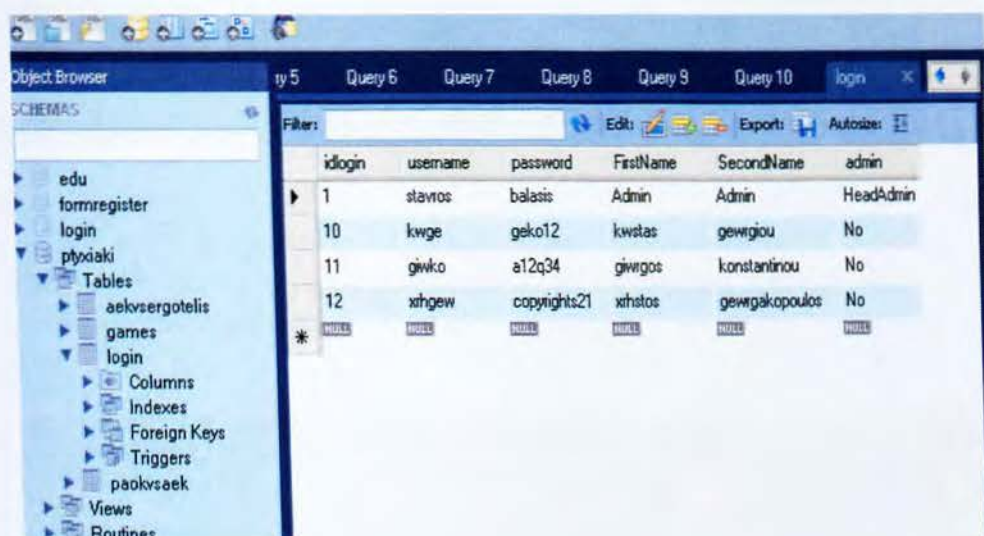
    $mysql="INSERT INTO ($table)
    (username,password,FirstName,SecondName,admin)
    VALUES '$a','$b','$d','$e','No')";
mysql_query($mysql) or die(mysql_error());
echo "<div id='leftnav'>Η καταχώρηση έγινε με επιτυχία.
    Τα στοιχεία σας είναι:<br />
    Όνομα χρήστη : '$a.'<br />Όνομα : '$d.'
    '<br />Επίθετο : '$e.'<br />
    Για να προχωρήσετε στην επιλογή αγώνα
    πιάστε το κουμπι:
    <form action = "commonuserinterface.php"method="post">
    <input type="hidden" name="FirstName" value="'$d.'">
    <input type="hidden" name="SecondName" value="'$e.'">
    <input type="submit" name="submit" value="register">
    </form></div>
    <div id="div_right"></div>";

```

Εικ 4 : Εγγραφή του νέου χρήστη στον επιλεγμένο πίνακα της βάσης δεδομένων.

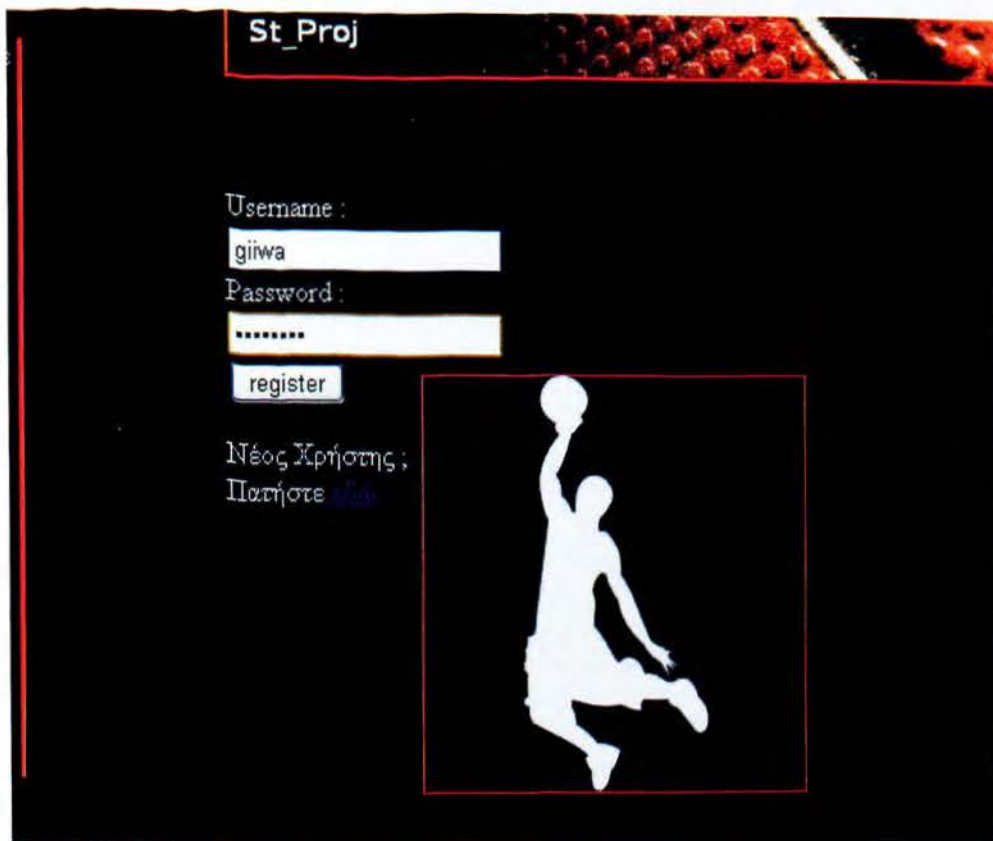


Εικ 5 : Ενημέρωση του χρήστη για τα αποθηκευμένα πλέον στοιχεία.



Εικ 6 : Στιγμιότυπο απο την βάση δεδομένων.

Μέρος Β : Το κομμάτι της εφαρμογής που έχουμε μέχρι τώρα περιγράψει αφορά την εγγραφή ενός χρήστη , δηλαδή την αυτοματοποιημένη μέθοδο εισαγωγής δεδομένων στην βάση δεδομένων. Στο παρακάτω κομμάτι θα δοθεί μια αμφίδρομη εισαγωγή-εξαγωγή δεδομένων που είναι απαραίτητη για την κράτηση μιας θέσης.



Εικ 7 : Αρχικά ο χρήστης κάνει εισαγωγή στην φόρμα της εφαρμογής για να προχωρήσει περαιτέρω. Η εγγραφή έχει ήδη γίνει αντικείμενο συζήτησης.



Εικ 8 : Η εισαγωγή ήταν επιτυχής και τα στοιχεία προωθούνται στην επόμενη σελίδα . Το κουμπι proceed θα παραπέμψει τον χρήστη

ανάλογα με τα δικαιώματα που του έχουν δοθεί (admin , headadmin , common user) .

```
<?php
$username = $_POST['username'];
$password = $_POST['password'];
$schema = "ptyxiaki";
$table = "login";
$db_host = "localhost";
$db_usrnm = "root";
$db_pswrd = "";
if (!$username==" " && !$password==" ")
//Έλεγχος για το εάν είναι
//κενά τα πεδία usrn&pswrd.
{
//Εάν όχι τότε σύνδεση με MySQL.
$a=mysql_connect ("$db_host", "$db_usrnm", "$db_pswrd")
or die (mysql_error ());
```

Εικ 9 : Στην εικόνα φαίνεται η μεταφορά των στοιχείων και η έναρξη διαδικασίας για την μεταφορά του χρήστη ανάλογα με τα δικαιώματα του.

```

    (//Έλεγχος στο πεδίο admin για το τι είδους//
    //χρήστη πρόκειται//
    echo '<div id="leftnav">Admin</div>';
    $interface="admininterface.php";
    $FirstName=$row['FirstName'];
    $SecondName=$row['SecondName'];
    }
elseif ($row['admin']=="HeadAdmin")
{
    $interface="headadmininterface.php";
    $FirstName=$row['FirstName'];
    $SecondName=$row['SecondName'];
}
elseif ($row['admin']=="No")
{
    $interface="commonuserinterface.php";
    $FirstName=$row['FirstName'];
    $SecondName=$row['SecondName'];
}

```

Εικ 10 : Στο πεδίο admin του πίνακα της βάσης δεδομένων γίνεται έλεγχος για την τιμή του. Ανάλογα με το περιεχόμενο του πεδίου γίνεται και η τοποθέτηση σε μια μεταβλητή (που χρησιμοποιείται αργότερα) της διεύθυνσης που θα ακολουθήσει ο χρήστης με το πάτημα του κουμπιού.

```

echo '<div id="leftnav">Πιέστε το πλήκτρο για να συνεχίσετε.
<form action="'. $interface. '"method="post">
<input type="hidden" name="FirstName" value="'. $FirstName. '">
<input type="hidden" name="SecondName" value="'. $SecondName. '">
<input type="submit" name="proceed" value="proceed">
</form></div>
<div id="div_success"></div>';

```

Εικ 11 : Η διεύθυνση της επόμενης σελίδας τοποθετείται στην μεταβλητή και καλείται μέσα απο την φόρμα . Ανάλογα με τα δικαιώματα του χρήστη γίνεται και η κατάλληλη προώθηση.



Εικ 12 : Ο χρήστης που επιθυμεί να κλείσει θέση καλείται να επιλέξει το παιχνίδι που τον ενδιαφέρει.



Εικ 13 : Στο στιγμιότυπο της εικόνας βλέπουμε την τοποθέτηση των θέσεων. Στο πλάι της εικόνας φαίνεται το τιμολόγιο και ο τύπος της κάθε θέσης, ενώ πάνω πάνω εμφανίζεται το παιχνίδι που έχει επιλεγεί.



```

for ($i=0 ; $i<=99 ; $i +=1 )
{
$seattaken_query =
"SELECT seattaken
FROM ptyxiaki.{ $game_selection} ";
$query = mysql_query($seattaken_query)
or die(mysql_error());
$result = mysql_result($query, $b)
or die(mysql_error());
$arrayfirst[$b]=$result;
$ViP_query = "SELECT ViP
FROM ptyxiaki.{ $game_selection} ";
$query2 = mysql_query($ViP_query)
or die(mysql_error());
$result2 = mysql_result($query2, $b)
or die(mysql_error());
$arraythird[$b]=$result2;
}

```

Εικ 14 : Εδώ φαίνεται η επαναληπτική διαδικασία για τον έλεγχο πληρότητας του γηπέδου. Στην ουσία αυτό που μας ενδιαφέρει είναι τα πεδία ViP και seattaken του πίνακα (παιχνιδιού) της MySQL που ο χρήστης έχει επιλέξει.

```

for ($j=0 ; $j<=99 ; $j +=1 )
{
if (" $arrayfirst[$j]" == "NO") //θέση που
//δεν είναι κρατημένη.
{
if (" $arraythird[$j]"=="Yes") //ViP θέση.
{
$arraysecond[$j] = "a1.gif";
}
else //θέση απλή.
{
$arraysecond[$j] = "a2.gif";
}
}
else //θέση κρατημένη.
$arraysecond[$j] = "a3.gif";
}

```

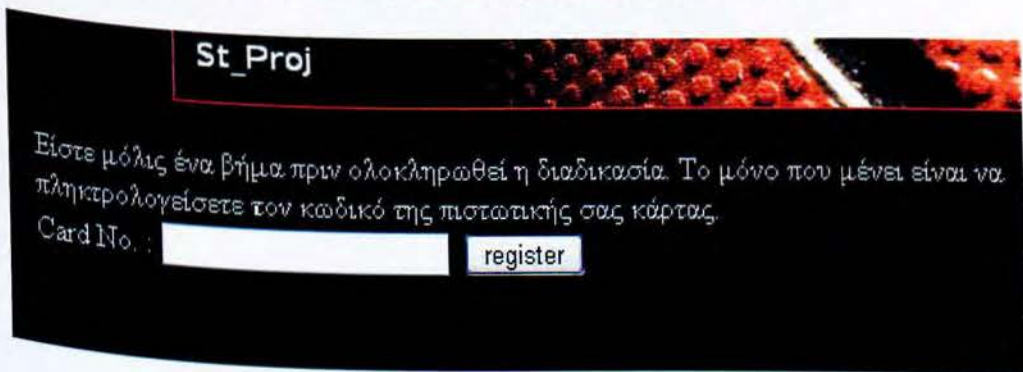
Εικ 15 : Ανάλογα με την τιμή που έχουν τα 2 πεδία seattaken και ViP παίρνουν και την κατάλληλη εικόνα.

```

<form id="three" action = "OFFICIAL2.php"
method="post">
<input type="hidden"
name="idseats" value="3">
<input type="hidden"
name="vip" value="NO">
<input type="hidden" name="game_selection"
value="<?php echo $game_selection ?>">
<input type="hidden" name="FirstName"
value="<?php echo $FirstName ?>">
<input type="hidden" name="SecondName"
value="<?php echo $SecondName ?>">
<input type="image" src="<?php echo $arraysecond[5] ?>"
name="submit" value="register">
</form>

```

Εικ 16 : Η φόρμα που χρησιμοποιείται για την υλοποίηση μίας θέσης του γηπέδου. Η τοποθέτηση στο ακριβές σημείο της σελίδας γίνεται από το κατάλληλο αρχείο css (cascade style sheet).



Εικ 17 : Στο σημείο αυτό ο χρήστης δίνει τον αριθμό της κάρτας του για να πραγματοποιηθεί η κράτηση της θέσης.

```

$f="UPDATE $db_table SET firstname='$FirstName'
WHERE idseat='$b'";
mysql_query($f) or die(mysql_error());
$g="UPDATE $db_table SET secondname='$SecondName'
WHERE idseat='$b'";
mysql_query($g) or die(mysql_error());
$h="UPDATE $db_table SET cardno='$cardno'
WHERE idseat='$b'";
mysql_query($h) or die(mysql_error());
$i="UPDATE $db_table SET seattaken='YES'
WHERE idseat='$b'";
mysql_query($i) or die(mysql_error());

```

Εικ. 18 : Η εισαγωγή νέων δεδομένων στα πεδία του πίνακα που αφορούν τη συγκεκριμένη θέση.

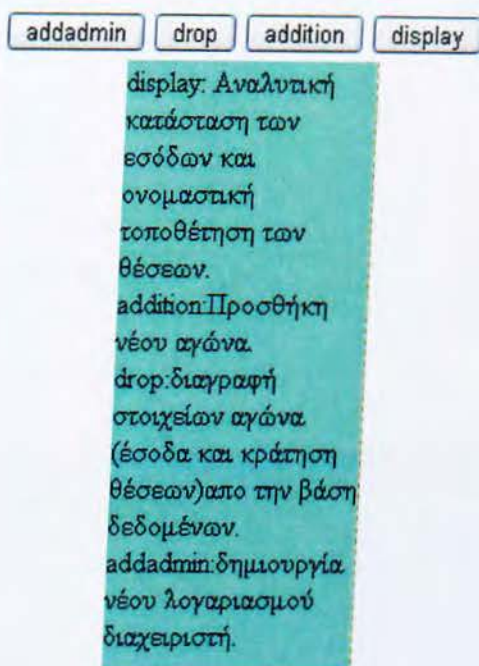
idseat	firstname	secondname	cardno	seattaken	VIP	Price
61	NULL	NULL	NULL	NO	No	40
62	kwstas	gewrgiou		YES	No	40

Εικ 19 : Η εγγραφή στην βάση δεδομένων MySQL των στοιχείων του χρήστη.



Εικ 20 : Ο χρήστης , αφού έχει τελειώσει η διαδικασία , μπορεί να επιστρέψει για να κάνει και άλλες κρατήσεις.

Μέρος Γ : Οι ενέργειες που μπορεί να κάνει ο διαχειριστής περιγράφονται σε αυτό το σημείο.Επειδή ενδέχεται όμως ο διαχειριστής να μην έχει την τεχνογνωσία να κάνει αλλαγές στον κώδικα , έχουμε ετοιμάσει μια διεπαφή (interface) με αυτοματοποιημένες μεθόδους για όλες τις λειτουργίες που τον ενδιαφέρουν.



Εικ. 21: ο διαχειριστής έχει επιλογές να προσθέσει νέους αγώνες, να διαγράψει τους παλιους, να δημιουργήσει νέους λογαριασμούς διαχειριστών καθώς και να δει την αναλυτική κατάσταση των συνολικών εσόδων απο τους αγώνες.



Εικ. 22: Ο διαχειριστής επιλέγει ποιούς αγώνες επιθυμεί να διαγράψει.

```

$entry = $_POST['mydropdown'];
$schema = "ptyxiaki";
$db_host = "localhost";
$db_usrnrm = "root";
$db_pswrd = "";
$a=mysql_connect ("$db_host",
|$db_usrnrm", "$db_pswrd")
or die(mysql_error());
$b=mysql_select_db ($schema, $a)
or die(mysql_error());
$c="DELETE FROM {$schema} . games
WHERE name = '{ $entry}'";
$results = mysql_query($c);
$d="drop table ptyxiaki . { $entry} ";
$results = mysql_query($d);

```

Εικ. 23: Το κομμάτι του κώδικα που εκτελείται απο την MySQL για την διαγραφή του επιλεγμένου αγώνα.

Ομάδα 1

Ομάδα 2

Για να γίνει εισαγωγή νέου αγώνα θα πρέπει να συμπληρώσετε τα 2 κουτά με τις αντίπαλες ομάδες, ύστερα να πατήσετε το κουμπί combine και τέλος Register.

Εικ. 24 : Εδώ γίνεται η εισαγωγή ενός νέου αγώνα.Ζητείται απο τον διαχειριστή να δηλώσει τις δύο αντίπαλες ομάδες και στην συνέχεια να γίνει η αποστολή του αιτήματος στην βάση δεδομένων.

```

$query = "CREATE TABLE ptyxiaki .{ $table}

    idseat INT NOT NULL AUTO_INCREMENT ,

    firstname VARCHAR(45) NULL ,

    secondname VARCHAR(45) NULL ,

    cardno VARCHAR(45) NULL ,

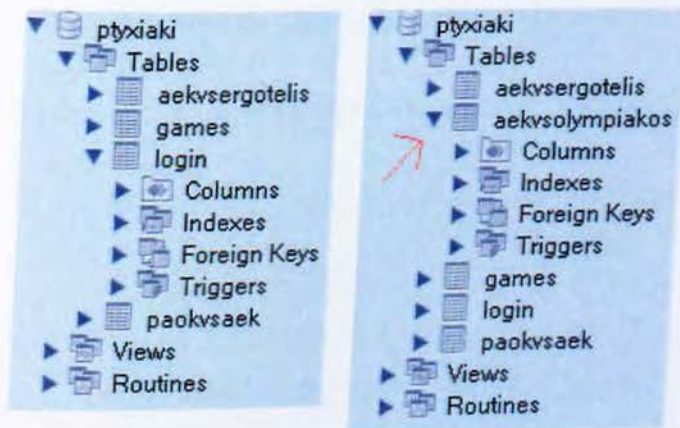
    seattaken VARCHAR(45) NULL ,

    ViP VARCHAR(45) NULL ,

    Price INT NULL ,

```

Εικ. 25 : Το κομμάτι του κώδικα που δημιουργεί και αρχικοποιεί τον πίνακα και όλες τις παραμέτρους του. (idseat , firstname , secondname , cardno , seattaken , ViP , Price).



Εικ.26 : Πριν και μετά οι δύο καταστάσεις της βάσης δεδομένων. Το κόκκινο βέλος στην δεύτερη εικόνα δείχνει τον νέο αγώνα που δημιούργησε ο διαχειριστής.

ΠΑΟΚVsAEK

1 - giorgws - gewrgiou  
31 - iwannis - iwannou  
32 - iwannis - iwannou  
40 - giorgws - gewrgiou  
42 - giorgws - gewrgiou  
47 - iasonas - papadopoulos  
49 - iasonas - papadopoulos  
52 - giorgws - gewrgiou  
53 - giorgws - gewrgiou  
54 - jas - on  
55 - giorgws - gewrgiou  
62 - kwstas - gewrgiou  
96 - giorgws - gewrgiou

Το σύνολο των εσόδων από τα εισητήρια είναι 600 Ευρώ.

Εικ. 27 : Το κουμπί display που εμφανίζει μια συνολική εικόνα των εσόδων και των κρατήσεων αυτή την στιγμή.

```
echo $arrayfirst[$i];  
echo "</br>";  
$query2 = "SELECT * FROM {$arrayfirst[$i]}";  
$results2 = mysql_query($query2) ;  
$seatprices=0;  
while($row2 = mysql_fetch_array($results2))  
{  
    if ($row2['seattaken']=="YES")  
    {  
        echo $row2['idseat'] . " - " .  
        $row2['firstname'] . " - " .  
        $row2['secondname'] . "<br />";  
    }  
}
```

Εικ.28 : Οι γραμμές του κώδικα που εκτυπώνουν στην οθόνη του διαχειριστή τις πληροφορίες των αγώνων.

Τέλος για την δημιουργία νέου διαχειριστή η μεθοδολογία είναι ακριβώς ίδια με αυτή για την εγγραφή νέου χρήστη , με την διαφορά ότι

στο πεδίο admin εισάγεται η λέξη YES αντί για NO όπως γίνεται με τον πελάτη.

## **ΚΕΦΑΛΑΙΟ 5**

### **5.1 Συμπεράσματα**

Όσον αφορά τα συμπεράσματα τα οποία εξήχθησαν από τη διπλωματική αυτή εργασία, μπορούμε να τα συνοψίσουμε στα ακόλουθα:

- Η πρόσβαση στην πληροφορία μιας Βάσης Δεδομένων απλοποιείται με τη χρήση του Internet.
- Ο συνδυασμός των τριών τεχνολογιών MySQL, PHP και Apache Server αποτελεί μια ιδιαίτερα σταθερή, αξιόπιστη και ανεξάρτητη πλατφόρμας μέθοδο, για την δημιουργία δυναμικών διαδικτυακών εφαρμογών με υψηλά κριτήρια αξιοπιστίας, απόδοσης, ταχύτητας και ασφάλειας και παρέχουν εκτός των άλλων τη μεταφερσιμότητα της εφαρμογής.
- Το www αποτελεί μια κατανεμημένη πηγή πληροφόρησης, με διαρκώς αυξανόμενη χρήση, γεγονός που δικαιολογεί την επιλογή του ως περιβάλλον ανάπτυξης και παρουσίασης οποιουδήποτε συστήματος ανάλογου περιεχομένου.

## **ΒΙΒΛΙΟΓΡΑΦΙΑ**

1. «PHP APACHE & MYSQL Όλα Σε Ένα», (Melonie Julie)  
ΓΚΙΟΥΡΔΑΣ Μ. 2004



2. «Beginning PHP and MySQL 5: From Novice to Professional»,  
(W. Jason Gilmore) APRESS 2006
3. «The Definitive Guide to MySQL 5», (Michael Kofler) APRESS  
2005
4. «Σχεσιακές Βάσεις Δεδομένων» (Χρ. Σκουρλάς) ΝΕΕΣ  
ΤΕΧΝΟΛΟΓΙΕΣ
5. «Τεχνολογίες Διαδικτύου», ( Ι. Στ. Βενιέρης, Ε. Νικολούζου)  
ΤΖΙΟΛΑΣ
6. «Θεμελιώδεις Αρχές Συστημάτων Βάσεων Δεδομένων» (R.  
Elmasri, S.B. Navathe). ΔΙΑΥΛΟΣ 1996
7. «PRO CSS Techniques» ,(Jeff Croft Ian Lloyd and Dan Rubin)  
APRESS 2006

### ***Πηγές Διαδικτύου***

1. <http://el.wikipedia.org/wiki/> (WIKIPEDIA)
2. <http://public.web.cern.ch/public/fr/About/WebStory-fr.html>  
(CERN WEBSITE)
3. <http://www.php.net/> (PHP WEBSITE)
4. <http://www.mysql.com/> (MySQL WEBSITE)
5. <http://www.apache.org/> (APACHE WEBSITE)
6. <http://www.scripts.com/php-scripts/> (SITE με PHP SCRIPTS)
7. <http://www.xoopsgreece.gr/modules/smartsection/item.php?itemid=1>  
(ΟΔΗΓΟΣ ΕΓΚΑΤΑΣΤΑΣΗΣ )
8. <http://www.elated.com/articles/form-validation-with-javascript/>  
(VALIDATION FORM με JAVASCRIPT)

### ***PDF***

1. ADOBE Dreamweaver CS3 (MANUAL)
2. PREMIUM SOFT Navicat 8.0 for MySQL (MANUAL)
3. Build Your Own Databases Driven Website Using Php &  
MySQL