

**ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΠΑΡΑΡΤΗΜΑ ΠΥΡΓΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΜΜΕ**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Δημιουργία ενός Συστήματος Διαχείρισης Περιεχομένου (CMS)
με εκπαιδευτικό σκοπό σχετικά με τεχνολογίες Διαδικτύου**

Θανάση Μοκλοβάν

Βελμάχος Θεόδωρος



Επιβλέπων: Διονύσιος Σ. Καρούσος

Πυργο, Σεπτέμβριος 2013

Υπεύθυνη Δήλωση

Εμείς οι Θανάσης Μικλοβάν και Βέλμαχος Θεόδωρος σπουδαστές του τμήματος Πληροφορικής και Μέσων Μαζικής Ενημέρωσης του ΤΕΙ Δυτικής Ελλάδας (Παράρτημα Πύργου) δηλώνουμε υπεύθυνα ότι είμαστε συγγραφείς αυτής της πτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχαμε για τη προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στη πτυχιακή εργασία. Επίσης, έχουμε αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης, βεβαιώνουμε ότι τα προαναφερθέντα κεφάλαια της πτυχιακής εργασίας προετοιμάστηκαν από εμάς προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Πληροφορικής και Μέσων Μαζικής Ενημέρωσης του ΤΕΙ Δυτικής Ελλάδας (Παράρτημα Πύργου)

Οι δηλούντες,
Θανάσης Μικλοβάν
Βέλμαχος Θεόδωρος

ΠΕΡΙΛΗΨΗ

Σε πρώτη φάση μελετήθηκε η εκτέλεση των ιστοσελίδων από στατιστικές σε δυναμικές και τα πλεονεκτήματα και μειονεκτήματα που υπάρχουν σε κάθε περίπτωση. Στη συνέχεια ,μελετήθηκαν τα συστήματα διαχείρισης περιεχομένου αναλύοντας τα πλεονεκτήματα τους και παρουσιάζοντας κάποια από τα δημοφιλέστερα CMS. Εδώ βλέπουμε επίσης τα πλεονεκτήματα δημιουργίας ενός δικού μας CMS με βάση τις αρχές σχεδίασης μιας ιστοσελίδας.

Τέλος συνεχίζουμε στην υλοποίηση της εργασίας δημιουργώντας στην αρχή την κύρια σελίδα μας και όλα τα στοιχεία αυτά που είναι πλέον απαραίτητα για έναν ιστότοπο και μετά δημιουργώντας ένα σύστημα διαχείρισης περιεχομένου από το οποίο θα πρέπει να ορίσουμε το περιεχόμενο της σελίδας μας.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή	5
1.1 Αντικείμενο της Πτυχιακής Εργασίας	6
1.2 Διάρθρωση της Πτυχιακής Εργασίας	6
ΚΕΦΑΛΑΙΟ 2: Στατικές και Δυναμικές Ιστοσελίδες	8
2.1 Στατικές ιστοσελίδες.....	9
2.2 Δυναμικές Ιστοσελίδες	10
2.3 Πλεονεκτήματα και Μειονεκτήματα Στατικής και Δυναμικής Ιστοσελίδας.....	11
2.4 Πλεονεκτήματα δυναμικής Ιστοσελίδας	12
2.5 Χαρακτηριστικά δυναμικών ιστοσελίδων	12
ΚΕΦΑΛΑΙΟ 3: Συστήματα διαχείρισης περιεχομένου (CMS).....	13
3.1 Τι προσφέρει το σύστημα διαχείρισης περιεχομένου	14
3.2 Ποια είναι τα πλεονεκτήματα των CMS.....	15
3.3 Διαθέσιμα Συστήματα Διαχείρισης Περιεχομένου	16
3.4 Πλεονεκτήματα δημιουργίας ενός δικού μας CMS.....	19
3.5 Τι είναι το Wamp Server;	20
3.5.1 Apache Server	21
3.5.2 PHP	22
3.5.3 MySQL	22
3.5.4 PHPMyAdmin	22
ΚΕΦΑΛΑΙΟ 4: Σχεδιασμός και Ανάπτυξη Ιστοχώρου.....	23
4.1 Header.....	24
4.1.1 Λογότυπο	24
4.1.2 Κυρίως Μενού	25

4.1.3 Social Media	26
4.1.4 RSS Feed	26
4.1.5 Metatags	29
4.2 Search Box	29
4.3 Περιεχόμενο	31
4.4 Link Labels	32
4.5 Sitemap	33
ΚΕΦΑΛΑΙΟ 5: Σχεδιασμός του CMS για τον ιστότοπό μας	34
5.1 Δημιουργία μιας φόρμας ελέγχου (Login Form)	35
5.2 Σελιδοποίηση παρουσίασης αποτελεσμάτων (Pagination)	35
5.3 TinyMCE Editor	37
Εγκατάσταση του TinyMCE editor	37
5.4 Δημιουργία καινούργιας σελίδας	38
5.5 Επεξεργασία σελίδας	40
5.6 Δημιουργία Ασκήσεων με την βοήθεια του Try it	43
5.6.1 Δημιουργία ασκήσεων – Administrator Panel	45
ΚΕΦΑΛΑΙΟ 6: Σύνοψη	48
6.1 Συμπεράσματα	48
6.2 Μελλοντική Εργασία	48

ΚΕΦΑΛΑΙΟ 1: Εισαγωγή

Η παρούσα Πτυχιακή Εργασία πραγματεύεται την δημιουργία ενός Συστήματος Διαχείρισης Περιεχομένου (CMS) με εκπαιδευτικό σκοπό σχετικά με τεχνολογίες Διαδικτύου. Το κεφάλαιο αυτό περιγράφει συνοπτικά το πλαίσιο δράσης της εργασίας αλλά και το αντικείμενό της.

The screenshot shows the 'infoMM' website with a navigation menu and a main content area. The main content area is titled 'Σειρές Μαθημάτων' (Course Series) and lists several courses:

- Μαθήματα HTML**: Η HTML είναι η γλώσσα με την οποία κατασκευάζουμε ιστοσελίδες. Οι ιστοσελίδες που επισκεπτόμαστε στα Internet δεν είναι τίποτα άλλο παρά αρχεία τα οποία περιέχουν κώδικα γραμμένο στην γλώσσα HTML. Οι browsers όπως οι Internet Explorer, Mozilla Firefox, Opera και αρκετά άλλα Li διαβάζουν τα αρχεία αυτά και εμφανίζουν το αποτέλεσμα του κώδικα HTML στην οθόνη μας. [Μαθήματα >>>](#)
- Μαθήματα XHTML**: Η XHTML είναι συνδυασμός της HTML και της XML. Για συγκεκριμένα η XHTML είναι μια παραλλαγή της HTML γραμμένη με κανόνες σύνταξης της XML. Δηλαδή ο κώδικας ενός XHTML αρχείου μπορεί να περιέχει σχεδόν όλες τις ετικέτες που περιέχει και η HTML, με διαφορά την αυστηρότερη σύνταξη του κώδικα. [Μαθήματα >>>](#)
- Μαθήματα CSS**: CSS σημαίνει Cascading Style Sheets και είναι σελίδ που μπορούμε να ορίσουμε για τις HTML σελίδες. Γράφοντας τις σελίδες μας μόνο με HTML κώδικα, μπορούμε να ελέγξουμε το χρώμα και το μέγεθος του κειμένου αλλά και άλλων στοιχείων της σελίδας όπως πλαίσια, links, λίστες κτλ. [Μαθήματα >>>](#)
- Μαθήματα AJAX**: Η χρήση της τεχνικής AJAX απελευθερεί σε μια ιστοσελίδα να συνδυάζει ασύγχρονα (αυτοχρηματούμενη) αυτοελέγοντας επί παραδείγματι μακρούς όγκους δεδομένων με τον κεντρικό παράγοντα ως αυτοελεγχόμενα μέρη της ιστοσελίδας. Είναι δηλ για παράδειγμα χωρίς να ανανεώνεται ολόκληρη η σελίδα! [Μαθήματα >>>](#)
- Μαθήματα Photoshop**: Το πρόγραμμα Photoshop της εταιρείας Adobe είναι ένα πολύ δυνατό και δημοφιλές πρόγραμμα επεξεργασίας εικόνας, με το οποίο μπορούμε να επεξεργαστούμε φωτογραφικές εικόνες που έχουμε στον σκληρό δίσκο του υπολογιστή μας και περιέχει βασικά δύο ομάδες εργαλείων, μια για συγγραφή και μια για επεξεργασία εικόνας. [Μαθήματα >>>](#)

On the right side, there are sections for 'Τελευταία Μαθήματα' (Latest Courses) and 'Τελευταία Νέα' (Latest News), each with a list of recent updates and links.

1.1 Αντικείμενο της Πτυχιακής Εργασίας

Η εργασία αυτή ασχολείται με την δημιουργία ενός εκπαιδευτικού site με την χρήση CMS. Βασικό στοιχείο του όλου επιχειρήματος είναι αφενός η σύντομη επισκόπηση των υπαρχόντων τεχνολογιών στα συστήματα Διαχείρισης Περιεχομένου και αφετέρου η υλοποίηση ενός νέου Συστήματος Διαχείρισης Περιεχομένου χρησιμοποιώντας τις τεχνολογίες PHP, MySQL, Javascript, RSS και XML. Το βάρος του σχεδιασμού του συστήματος έχει πέσει στην κάλυψη των αναγκών μιας εκπαιδευτικής ιστοσελίδας που στόχο έχει να αναδείξει τα βασικά στάδια των τεχνολογιών ανάπτυξης Ιστοσελίδων.

Σκοπός και στόχοι της εργασίας

- Σύντομη μελέτη των CMS
- Εγκατάσταση κατάλληλου περιβάλλοντος για ανάπτυξη Ιστοσελίδων
- Σχεδιασμός και υλοποίηση Νέου CMS με βάση τα κριτήρια που τίθενται από την εκφώνηση της εργασίας
- Σχεδιασμός και ανάπτυξη ιστοσελίδας εκπαιδευτικού περιεχομένου στο νέο CMS.

1.2 Διάρθρωση της Πτυχιακής Εργασίας

Στο επόμενο κεφάλαιο παρουσιάζετε μια σύγκριση ανάμεσα στις στατικές και δυναμικές ιστοσελίδες. Εδώ προβάλλονται τα πλεονεκτήματα και τα μειονεκτήματα της στατικής και δυναμικής σελίδας και ειδικότερα τα πλεονεκτήματα και τα χαρακτηριστικά των δυναμικών ιστοσελίδων.

Στο τρίτο κεφάλαιο γίνεται μια γενική ανάλυση στα συστήματα διαχείρισης περιεχομένου όπως: Τι προσφέρει το σύστημα διαχείρισης περιεχομένου. Ποια τα πλεονεκτήματα των CMS. Χρήση ενός συστήματος διαχείρισης περιεχομένου και κλείνει με τα πλεονεκτήματα δημιουργίας ενός δικού μας CMS το οποίο αποτελεί και σκοπό αυτής της εργασίας.

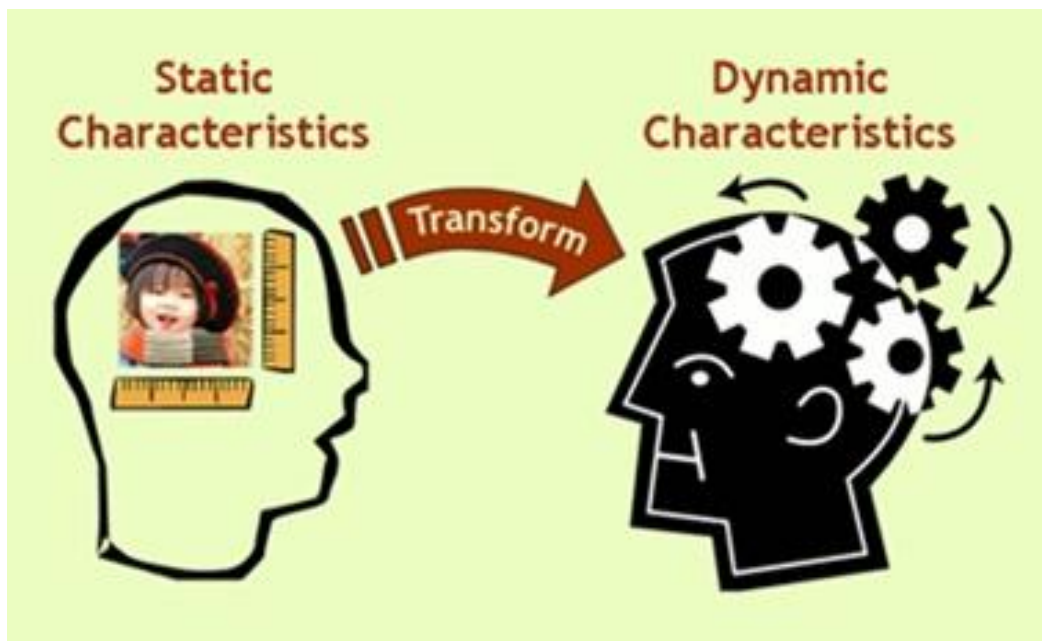
Στο τέταρτο κεφάλαιο αναπτύσσετε ο σχεδιασμός και η ανάπτυξη της κύριας σελίδας μας, η οποία αποτελείτε από διάφορα στοιχεία όπως header, RSS Feed και Search Box και Sitemap.

Στο κεφάλαιο αυτό περιγράφετε αναλυτικά η δημιουργία του κάθε στοιχείου για την ανάπτυξη ενός συστήματος διαχείρισης περιεχομένου για τον ιστότοπο μας, δίνοντας έμφαση στα τέσσερα κυριότερα σημεία: Δημιουργία φόρμας ελέγχου, Δημιουργία καινούργιας σελίδας, Επεξεργασία μιας ήδη υπάρχουσας σελίδας, Δημιουργία Ασκήσεων με την βοήθεια του Try it.

Τέλος στο έκτο και τελευταίο κεφάλαιο παρουσιάζετε η σύνοψη της εργασίας και τα συμπεράσματα που εξάγονται από όλη την εργασία για την δημιουργία ενός συστήματος διαχείρισης περιεχομένου με εκπαιδευτικό σκοπό σχετικά με τις νέες τεχνολογίες διαδικτύου.

ΚΕΦΑΛΑΙΟ 2: Στατικές και Δυναμικές Ιστοσελίδες

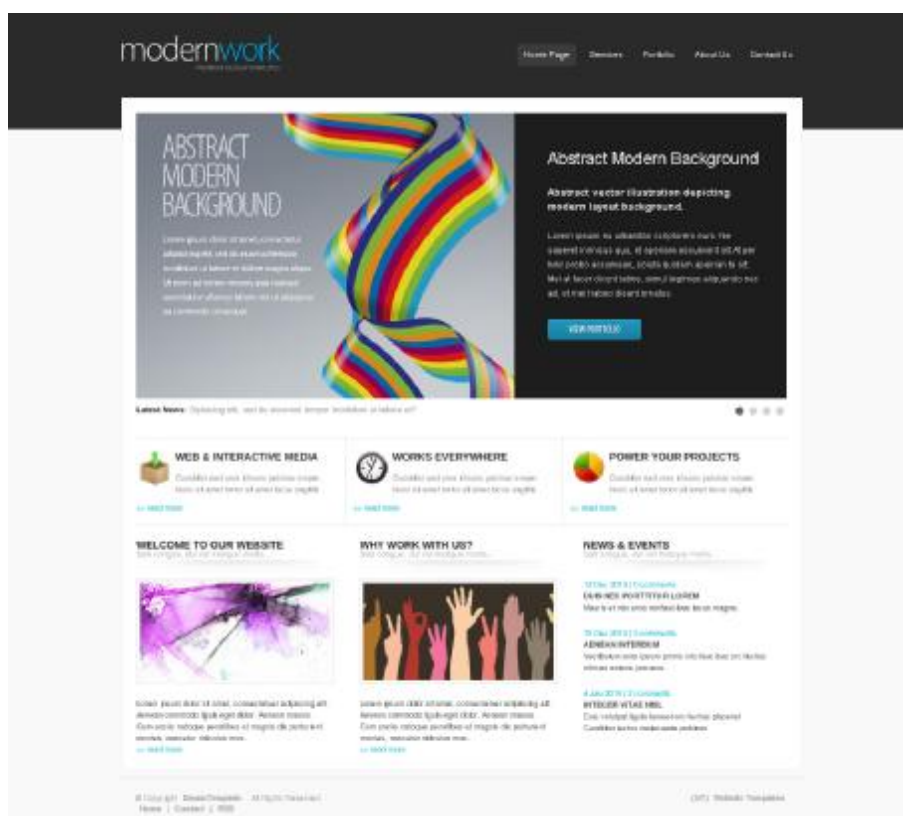
Υπάρχουν εκατομμύρια ιστοσελίδες στο διαδίκτυο οι οποίες είναι σε διαφορετικά σχήματα και μεγέθη. Όλες αυτές μπορούν να χωριστούν σε δυο κατηγορίες οι οποίες είναι οι στατικές και δυναμικές σελίδες. Οι διαφορές ανάμεσα στις δυο κατηγορίες δεν γίνονται πάντα αντιληπτές όταν παρατηρούμε έναν ιστότοπο αλλά υπάρχει μια τεραστία διαφορά στην λειτουργία της ιστοσελίδας.



Εικόνα 2.1: Στατικές και Δυναμικές Ιστοσελίδες

2.1 Στατικές ιστοσελίδες

Στατική ιστοσελίδα καλείται μια ιστοσελίδα που παρουσιάζεται στο χρήστη ακριβώς όπως έχει αποθηκευτεί στο server. Οι στατικές ιστοσελίδες περιέχουν HTML κώδικα ο οποίος καθορίζει τη δομή και το περιεχόμενο της ιστοσελίδας. Κάθε φορά που «φορτώνει» μια στατική ιστοσελίδα έχει πάντα την ίδια όψη (ίδιο περιεχόμενο). Ο μόνος τρόπος να αλλάξει το περιεχόμενο μιας στατικής ιστοσελίδας είναι όταν ο προγραμματιστής ενημερώσει τον κώδικα και δημοσιεύσει το αρχείο στον server.



Εικόνα 2.1: Στατική Ιστοσελίδα. [Πηγή: <http://templateaccess.com>]

Υπάρχουν πολλές στατικές ιστοσελίδες στο Διαδίκτυο, είναι δύσκολο να πούμε αμέσως εάν είναι στατική, αλλά είναι πιο πιθανό, εάν η σελίδα μοιάζει απλή και είναι για μια μικρότερη εταιρεία, και απλά να παρέχει πληροφορίες χωρίς πολλά πρόσθετα πράγματα, να είναι μια στατική ιστοσελίδα.. Οι στατικές ιστοσελίδες είναι φθηνότερες στην σχεδίαση, ανάπτυξη και υλοποίηση, και πολλές μικρές εταιρίες και ιδιώτες εξακολουθούν να τις χρησιμοποιούν για την παρουσίαση της δικτυακής τους ταυτότητας.

2.2 Δυναμικές Ιστοσελίδες

Δυναμικές ιστοσελίδες μπορούν να χαρακτηριστούν οι σελίδες οι οποίες χρησιμοποιούν php, asp, jsp, sql με την απαραίτητη παρουσία μιας βάσης δεδομένων. Αυτές οι ιστοσελίδες περιέχουν ειδικού τύπου κώδικα στο server ο οποίος επιτρέπει στον server να δημιουργεί δυναμικά το περιεχόμενο της σελίδας κάθε φορά που καλούμε την συγκεκριμένη διεύθυνση url [Robin Nixon , 2009].

Για παράδειγμα ένας server μπορεί να αναπαριστά πληροφορίες που δεν είναι στάσιμες όπως είναι η τρέχουσα ημερομηνία και ώρα στην σελίδα. Μπορεί επίσης να δίνει διαφορετική απάντηση κάθε φορά που κάποιος χρήστης συμπληρώνει μία φόρμα σε μια ιστοσελίδα με διαφορετικά δεδομένα.



Εικόνα 2.2: Δυναμική Ιστοσελίδα. [Πηγή: <http://www.gavick.com/>]

Πολλές δυναμικές ιστοσελίδες χρησιμοποιούν κώδικα για να προσπελάσουν μια βάση δεδομένων στο server η οποία δίνει τη δυνατότητα στην σελίδα να ενημερώσει το περιεχόμενό της ανάλογα με τα δεδομένα που έχουν αποθηκευτεί στη βάση.

2.3 Πλεονεκτήματα και Μειονεκτήματα Στατικής και Δυναμικής Ιστοσελίδας

∅ Πλεονεκτήματα των στατικών ιστοσελίδων

- Είναι πολύ γρήγορα στην σχεδίαση και την ανάπτυξη
- Είναι αρκετά φτηνά στην υλοποίηση
- Είναι αρκετά μικρά τα κόστη φιλοξενίας τους σε server

∅ Μειονεκτήματα των στατικών ιστοσελίδων

- Χρειάζονται ιδιαίτερες γνώσεις προγραμματισμού ιστοσελίδων για την ενημέρωσή τους.
- Το site δεν είναι και πολύ χρήσιμο για το χρήστη (επειδή το περιεχόμενο παραμένει στάσιμο)

∅ Πλεονεκτήματα των δυναμικών ιστοσελίδων

- Δημιουργούν ένα πολύ πιο λειτουργικό site
- Είναι πιο εύκολο να ανανεωθούν
- Το συνεχώς αναβαθμιζόμενο περιεχόμενο παροτρύνει τους επισκέπτες της σελίδας να την ξαναεπισκευτούν και βοηθά τον χρήστη να αποκτήσει πρόσβαση σε αυτήν μέσω των μηχανών αναζήτησης.
- Μπορεί να λειτουργήσει ως ένα σύστημα που επιτρέπει την συνεργασία ανάμεσα στο προσωπικό της σελίδας και τους χρήστες ή ανάμεσα στους χρήστες μεταξύ τους.

∅ Μειονεκτήματα των δυναμικών ιστοσελίδων

- Είναι πιο αργές στην προσπέλαση
- Η φιλοξενία του site (hosting) κοστίζει λίγο παραπάνω.

2.4 Πλεονεκτήματα δυναμικής Ιστοσελίδας

Ως πλεονεκτήματα μιας δυναμικής ιστοσελίδας μπορούν να θεωρηθούν τα ακόλουθα [Dr. Andreas Mauthe; Dr. Peter Thomas, 2004]:

- Δημιουργία παρουσίας επιχείρησης στο διαδίκτυο
- Αύξηση αναγνωσιμότητας επιχείρησης ή/και προϊόντων
- Προώθηση επιχείρησης ή/και προϊόντων μέσω διαφημίσεων και διαδικτυακών εκστρατειών
- Αύξηση πωλήσεων
- Αύξηση του δικτύου πωλήσεων, πελατών και συνεργατών
- Υποστήριξη πελατών
- Εύκολη επικοινωνία με πελάτες
- Παροχή πληροφοριών σχετικά με νέα και ανακοινώσεις για την επιχείρηση ή/και τα προϊόντα

2.5 Χαρακτηριστικά δυναμικών ιστοσελίδων

Τα χαρακτηριστικά των δυναμικών ιστοσελίδων είναι [Dr. Andreas Mauthe; Dr. Peter Thomas, 2004]:

- Ειδικής κατασκευής σχεδιασμός ιστοσελίδας (custom made web design)
- Σύστημα διαχείρισης Περιεχομένου (Content Management System – CMS)
- Απεριόριστος αριθμός σελίδων
- Απεριόριστος αριθμός καταχωρήσεων
- Συμβατότητα με πολλαπλούς browsers
- Συμβατότητα με πρότυπα διαδικτύου
- Βελτιστοποίηση χρόνου απόκρισης (response time optimized)
- Βασικό Search Engine Optimization (SEO)

ΚΕΦΑΛΑΙΟ 3: Συστήματα διαχείρισης περιεχομένου (CMS)

Το CMS είναι μια διαδικτυακή εφαρμογή που παρέχει δυνατότητες για πολλούς χρήστες με διαφορετικά επίπεδα δικαιωμάτων για τη διαχείριση περιεχομένου, δεδομένων, ή πληροφοριών σε μια ιστοσελίδα. Η διαχείριση περιεχομένου αναφέρεται ουσιαστικά στην δημιουργία, επεξεργασία, αρχειοθέτηση, δημοσίευση, τη διανομή περιεχομένου ,δεδομένων και πληροφοριών σε μια ιστοσελίδα.

Πολλά συστήματα διαχείρισης περιεχομένου προσφέρουν μια Web-based διεπαφή χρήστη, που επιτρέπει στους εκδότες να έχουν πρόσβαση στο site χρησιμοποιώντας μόνο ένα Web browser από οποιονδήποτε υπολογιστή. Επίσης, ένα CMS σχεδιασμένο για την δημοσίευση υλικού στο internet , προσφέρει την δυνατότητα να μπορεί οποιοσδήποτε χρήστης να αποκτήσει πρόσβαση σε αυτό το υλικό μέσα από την μηχανή αναζήτησης περιεχομένου του site χρησιμοποιώντας συγκεκριμένες λέξεις-κλειδιά.

Σε περιπτώσεις που η σωστή αρχειοθέτηση αποτελούσε απαραίτητο χαρακτηριστικό για την ορθή λειτουργία μιας εταιρίας, πριν τη χρήση του CMS τα δεδομένα ήταν διάσπαρτα μέσα στην εταιρία σε ξεχωριστά αρχεία. Κάθε αρχείο έπρεπε να ενημερώνεται ατομικά μέσα από μια μη αυτόματη διαδικασία, η οποία συχνά οδηγούσε σε λάθη και ανακρίβειες. Το πλεονέκτημα της χρήσης των CMS προσέφερε τη λύση δίνοντας ένα κεντρικό σύστημα διαχείρισης και ανανέωσης δεδομένων. Επειδή κάθε κομμάτι του περιεχομένου αποθηκεύεται μόνο μία φορά σε ένα CMS, μπορεί να επαναχρησιμοποιηθεί σε ένα ή πολλαπλά έγγραφα.

Κάθε εγγραφή στο CMS ανανεώνεται αυτόματα μόλις πραγματοποιηθεί αλλαγή των δεδομένων σε έναν σταθμό εργασίας. Αυτό σημαίνει ότι τα αρχεία είναι διαρκώς ενημερωμένα και βρίσκονται σε ένα συγκεκριμένο σημείο στο server ορατά από όλους του χρήστες που έχουν πρόσβαση σε αυτά

3.1 Τι προσφέρει το σύστημα διαχείρισης περιεχομένου



Εικόνα 3.1.1: CMS. [Πηγή: Google Images]

Χρησιμοποιώντας ένα σύστημα διαχείρισης περιεχομένου μπορούμε να καταφέρουμε την αλλαγή του περιεχομένου μιας σελίδας χωρίς ιδιαίτερες γνώσεις που έχουν σχέση με την δημιουργία ιστοσελίδας και γραφικών στοιχείων. Τα CMS χρησιμοποιούν διάφορα html editors που επιτρέπουν την μορφοποίηση ενός κειμένου όποτε αυτό χρειαστεί. Κάθε αλλαγή στην ιστοσελίδα μπορεί να γίνει από οποιονδήποτε υπολογιστή χωρίς να χρειαστούν διάφορα ειδικά προγράμματα για την επεξεργασία μας. Μέσα από το ίντερνετ ο διαχειριστής μπορεί να συντάξει εύκολα μια ανάρτηση και να ενημερώσει άμεσα την ιστοσελίδα του.

Για να κατανοήσουμε καλύτερα την αξία ενός CMS θα πρέπει να σκεφτούμε ότι για ένα δυαδικτυακό τόπο που χρειάζεται καθημερινή ενημέρωση και δεν χρησιμοποιεί κάποιο σύστημα διαχείρισης περιεχομένου θα πρέπει ο διαχειριστής να τοποθετήσει το περιεχόμενο στα σημεία της ιστοσελίδας που αυτός θέλει χρησιμοποιώντας HTML. Μετά θα πρέπει να ενημερωθούν οι σύνδεσμοι όλων των προηγούμενων σελίδων για το καινούργιο περιεχόμενο, έτσι ώστε να συνδέονται με αυτήν. Αφού γίνουν όλες αυτές οι αλλαγές θα πρέπει να ανεβάσουμε τα καινούργια αρχεία στα οποία έγιναν αλλαγές ξανά στον σέρβερ για να ενημερώσει τη σελίδα μας σχετικά με όλες τις καινούργιες αλλαγές και αναρτήσεις.

Αντιθέτως χρησιμοποιώντας ένα CMS ο διαχειριστής απλά θα χρειαστεί να γράψει την ανάρτηση και να πατήσει δημοσίευση. Όλες οι άλλες διεργασίες όπως ενημέρωση ιστοσελίδας και συνδέσμων γίνονται αυτόματα από το σύστημα προσφέροντας έτσι πολλά πλεονεκτήματα, ταχύτητα και ευκολίες.

3.2 Ποια είναι τα πλεονεκτήματα των CMS

Για τον χρήστη

- Δυνατότητα να προσθέσει ή να επεξεργαστεί το περιεχόμενο της σελίδας ο ίδιος διαδραστικά χωρίς την χρήση κάποιου εξειδικευμένου προγράμματος.
- Η αίσθηση του άμεσου ελέγχου της ιστοσελίδας ως δικτυακής ταυτότητας της εταιρίας αλλά και ως επένδυσης, δημιουργεί ένα συναίσθημα ασφάλειας. Ειδικότερα για μια οργάνωση ή επιχείρηση που είναι δυναμική και πρέπει οι ενημερώσεις στην ιστοσελίδα να υλοποιούνται άμεσα.
- Δεν χρειάζεται να πληρώσει προγραμματιστή σε μηνιαία βάση (ή σε ωριαία χρέωση) για την συντήρηση και για τυχόν αλλαγές στο περιεχόμενο της ιστοσελίδας
- Είναι πολύ χρήσιμο σε οργανισμούς, που έχουν ιστοσελίδες με ευαίσθητα προσωπικά δεδομένα συνεργατών τα οποία απαιτούν απόλυτο έλεγχο στον διαμοιρασμό της πληροφορίας.
- Πολλοί CMS προσφέρουν τη δυνατότητα να ανατεθούν ρόλοι αρθρογράφων, συντακτών και αρχισυντακτών όπου καθένας από αυτούς έχει συγκεκριμένα δικαιώματα τα οποία ορίζονται από τον διαχειριστή του ιστότοπου. Ένα CMS μπορεί να είναι ιδανικό για αυτόν τον τύπο της ροής εργασίας.

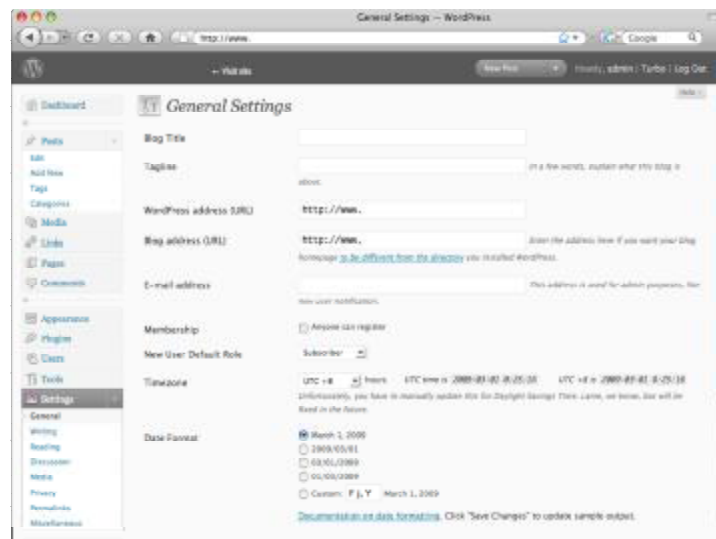
Για τον σχεδιαστή ιστοσελίδων (Web Designer)

- Μπορούμε να ενσωματώσουμε διάφορα modules – plugins τα οποία έχουν κατασκευαστεί από τρίτες εταιρίες και παρέχουν την δυνατότητα να χρήσης τεχνολογιών που εξυπηρετούν συγκεκριμένες ανάγκες των χρηστών.
- Έχει πολύ γρήγορη υλοποίηση. Ένα καλό CMS μπορεί να φέρει εις πέρας διάφορες λειτουργίες που σε διαφορετική περίπτωση θα ήταν πολύ επίπονο να υλοποιήσουμε γράφοντας κώδικα από το μηδέν.
- Ο διαχειριστής μπορεί χρησιμοποιώντας διάφορα templates να αλλάξει εύκολα οποιαδήποτε στιγμή την εμφάνιση της ιστοσελίδας του.

3.3 Διαθέσιμα Συστήματα Διαχείρισης Περιεχομένου

Κάποια από τα δημοφιλέστερα συστήματα διαχείρισης περιεχομένου που μελετήσαμε, καθώς και μια σύντομη περιγραφή τους παρατίθεται ακολούθως [URL: Makeuseof]:

Wordpress

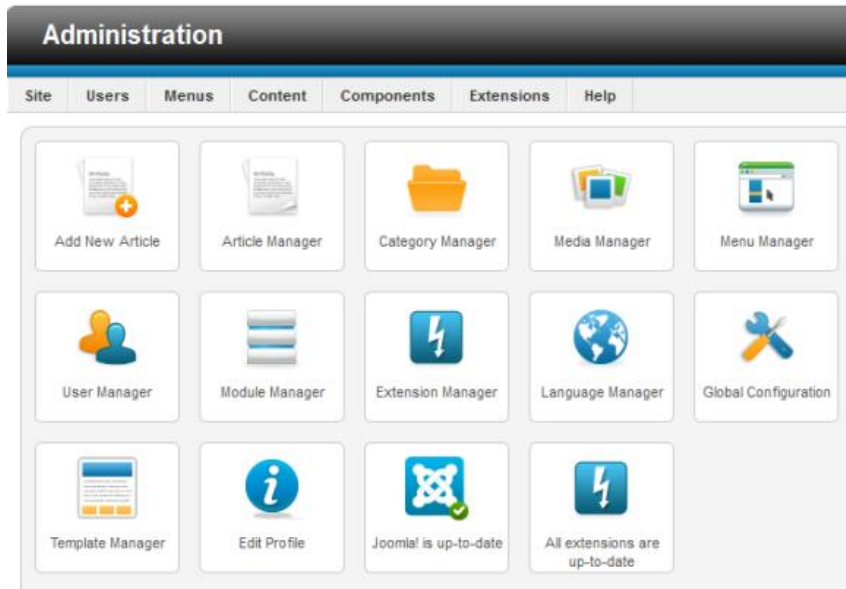


Εικόνα 3.3.1: Wordpress CMS. [Πηγή:Google Images]

Το πιο προσιτό και ενδεχομένως πιο δημοφιλές CMS είναι το WordPress. Η δυναμικότητα και η ευχρηστία του οφείλεται κυρίως στην γρήγορη εγκατάστασή του και στην τεράστια υποστήριξη από διάφορες κοινότητες χρηστών, η οποία οδηγεί σε μια μεγάλη σειρά από plugins και βελτιώσεις για την πλατφόρμα [URL: Wordpress].

Joomla

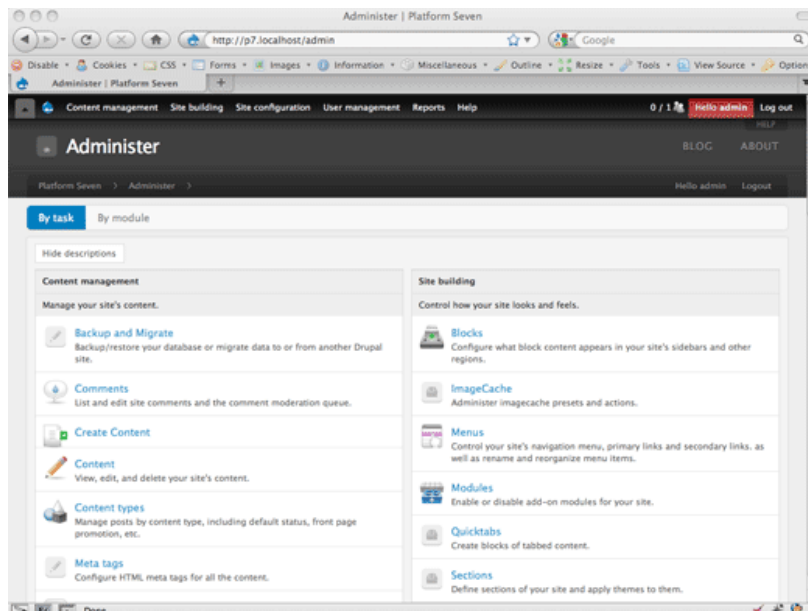
Μια γρήγορη επίσκεψη στην ιστοσελίδα του Joomla θα αποκαλύψει την αλήθεια του ισχυρισμού ότι εκατομμύρια ιστοσελίδες χρησιμοποιούν το συγκεκριμένο CMS [URL: Joomla]. Ο λόγος για αυτό είναι απλός. Η Joomla είναι εξαιρετικά προσαρμόσιμη, κατάλληλη για σχεδόν οποιοδήποτε σκοπό.



Εικόνα 3.3.2: Joomla CMS. [Πηγή:Google Images]

Γι' αυτό και η Joomla χρησιμοποιείται συχνά από μικρές και μεσαίες επιχειρήσεις, μεγάλους οργανισμούς, μη-κερδοσκοπικούς οργανισμούς καθώς και από ιδιώτες.

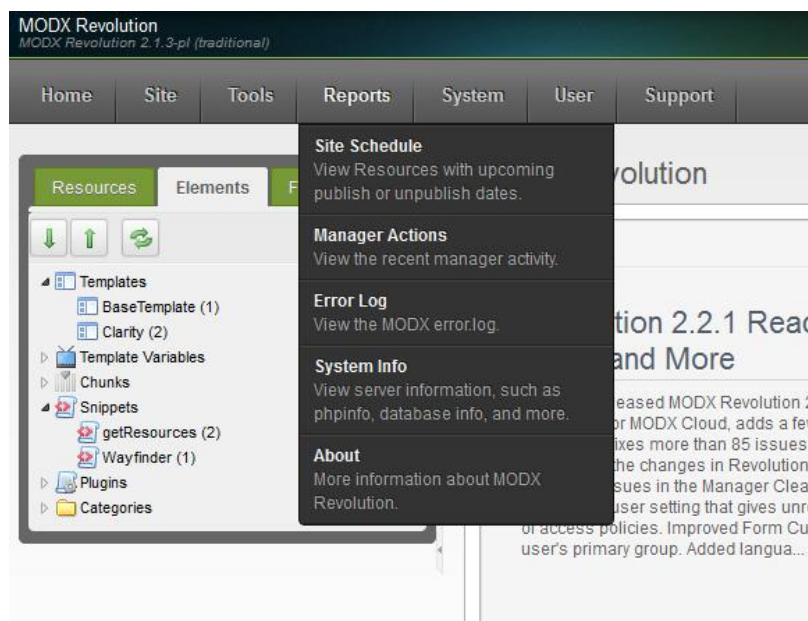
Drupal



Εικόνα 3.3.3: Drupal CMS. [Πηγή:Google Images]

Ένα δημοφιλές δωρεάν και ανοικτού κώδικα CMS, το Drupal είναι συχνά μια από τις πρώτες επιλογές όταν δημιουργείται μια νέα ιστοσελίδα. Όπως και πολλά από τα υπόλοιπα εργαλεία που μελετήσαμε, το Drupal μπορεί να κλιμακωθεί για προσωπικά blogs ή για τεράστια site επιχειρήσεων, και όπως στο WordPress, υπάρχουν χιλιάδες plugins που μπορούν να προστεθούν για να αυξήσουμε τη λειτουργικότητα.

Modx



Εικόνα 3.3.4: Modx CMS. [Πηγή: Google Images]

Το Modx είναι ένα από πιο τρανταχτά παραδείγματα του επιχειρήματος ότι δεν απαιτούνται ιδιαίτερες γνώσεις για την δημιουργία μιας ιστοσελίδας. Με πάνω από 100.000 σελίδες να έχουν υλοποιηθεί τόσο από ιδιώτες όσο και από επιχειρήσεις. Το ModX είναι εύκολο στη χρήση, επιτρέπει σε μη τεχνικό προσωπικό την επεξεργασία περιεχομένου και προσφέρει διάφορα πλεονεκτήματα όπως το να χρησιμοποιηθούν πολλαπλά styles (CSS), στην ίδια σελίδα.

DotNetNuke



Εικόνα 3.3.5: DotNetNuke CMS. [Πηγή: Google Images]

Παρόλο που σχεδόν όλα τα CMS είναι γραμμένα σε PHP κώδικα, το DotNetNuke, αποτελεί μια εξαίρεση. Το Λογισμικό είναι γραμμένο σε Microsoft ASP .net και είναι ειδικά σχεδιασμένο για Windows Servers. Έχοντας ως δεδομένο ότι πολλές εταιρίες έχουν τα εσωτερικά τους δίκτυα σε windows Servers, Δημιουργείται ένας επιπρόσθετος λόγος να επιλέξουν DotNetNuke για την υλοποίηση της ιστοσελίδας τους.

3.4 Πλεονεκτήματα δημιουργίας ενός δικού μας CMS

Ο κυριότερος λόγος δημιουργίας ενός νέου δικού μας CMS ήταν γιατί είχαμε μια πολύ συγκεκριμένη, ξεκάθαρη ιδέα του τι θέλαμε να σχεδιάσουμε και με ποιες θα ήταν οι ιδανικότερες συνθήκες υπό τις οποίες θα έπρεπε να λειτουργεί.

Ο επικρατέστερος υποψήφιος ανάμεσα στα πιθανά υπάρχοντα CMS ήταν το Wordpress για λόγους περισσότερο υποκειμενικούς, καθώς έρχονταν αρκετά κοντά στο τι θα θέλαμε να σχεδιάσουμε. Ωστόσο όπως και τα υπόλοιπα CMS, το Wordpress απαρτίζεται από ένα πλήθος plugins και τμημάτων προεπεξεργασμένου κώδικα για να μπορέσει να λειτουργήσει σωστά.

Το κυριότερο επιχείρημα που τελικά αποτέλεσε και τον καθοριστικό παράγοντα είναι ότι η παρουσία ενός ξεκάθαρα, ισχυρού, και προσβάσιμου από την πλευρά του πελάτη κώδικα είναι σημαντικότερη από την δυνατότητα του server να δημιουργήσει τις σελίδες ευκολότερα.

Όλοι οι λόγοι για τους οποίους δημιουργήσαμε ένα ξεχωριστό δικό μας CMS, μπορούν να συνοψιστούν σε έναν. Μπορούμε με αυτόν τον τρόπο να έχουμε το σύστημα που θέλουμε το οποίο να κάνει όλα όσα θέλουμε και τίποτα περισσότερο. Επίσης όταν πρόκειται για την πραγματική χρήση του συστήματος σε καθημερινή βάση, το να υλοποιήσουμε το δικό μας σύστημα σημαίνει ότι ήδη ξέρουμε ακριβώς πως λειτουργεί το admin interface γιατί εμείς έχουμε γράψει τον κώδικα πίσω από αυτό.

Το ίδιο ισχύει και για τις αναβαθμίσεις και βελτιώσεις του συστήματος. Αν για παράδειγμα χρειαστεί να προσθέσουμε νέες δυνατότητες αργότερα ή να αλλάξουμε τον τρόπο που λειτουργεί κάτι, ξέρουμε ακριβώς που να παρέμβουμε στον κώδικα για να το πραγματοποιήσουμε.

Εντέλει δεν χρειάζεται να σκεφτούμε ποτέ ότι έχει να κάνει με περιορισμό της χρήσης κάποιων χαρακτηριστικών λόγω ελλιπούς άδειας χρήσης ή αντιγραφής του προγράμματος καθώς είναι δικό μας και μπορούμε να κάνουμε με αυτό ότι επίπεδο παραμετροποίησης επιθυμούμε.

3.5 Τι είναι το Wamp Server;

Το Wamp Server είναι ένα περιβάλλον για ανάπτυξη ιστοσελίδων σε Windows που μας επιτρέπει να δημιουργούμε WEB εφαρμογές χρησιμοποιώντας apache, php και βάσεις δεδομένων SQL. Παράλληλα το phpmyadmin μας επιτρέπει να χειριστούμε εύκολα τη βάση δεδομένων [URL: Wamp Server].

Το Wamp Server εγκαθιστά αυτόματα όλα όσα χρειάζεται ο χρήστης για να ξεκινήσει την ανάπτυξη των εφαρμογών WEB και είναι σχεδιασμένος με τέτοιο τρόπο ώστε να μπορεί να σηκωθεί ένα τοπικός server χωρίς να χρειαστεί να ρυθμιστεί από τον χρήστη ούτε ένα αρχείο καθώς όλα γίνονται αυτόματα από την διεπαφή.

Τα πλεονεκτήματα που συγκεντρώνει η αρχιτεκτονική του Wamp server συνοψίζονται στα παρακάτω

- § Μπορεί να χρησιμοποιηθεί από όλους τους φυλλομετρητές
- § Μπορεί να εγκατασταθεί σε οποιοδήποτε λειτουργικό σύστημα
- § Είναι portable (εκτελείται σε αφαιρούμενο μέσο αποθήκευσης χωρίς να χρειάζεται εγκατάσταση)
- § Έχει σπονδυλωτή δομή (αποτελείται από επιμέρους τμήματα που είναι ανεξάρτητα ως προς τη λειτουργία τους)
- § Δεν απαιτείται σύνδεση στο internet για την χρήση του



Εικόνα 3.3.5: WampServer Panel. [Πηγή:Google Images]

3.5.1 Apache Server

Αρχικά σχεδιασμένος για περιβάλλον Unix, ο Apache Web Server έχει κωδικοποιηθεί για τα Windows και άλλα λειτουργικά συστήματα δικτύου. Παρέχει ένα πλήρες φάσμα χαρακτηριστικών των Web server , συμπεριλαμβανομένων των CGI, SSL και εικονικών domains. Ο Apache υποστηρίζει επίσης την προσθήκη plugin modules για επεκτασιμότητα. Ο Apache είναι δωρεάν λογισμικό, που διανέμεται από το Apache Software Foundation που προωθεί διάφορες και προηγμένες Web τεχνολογίες ελεύθερου και ανοικτού κώδικα.

3.5.2 PHP

Αυτό που διαχωρίζει την PHP από την client-side Javascript είναι ότι ο κώδικας εκτελείται στον server, δημιουργώντας HTML κώδικα ο οποίος στη συνέχεια αποστέλλεται στον χρήστη. Ο χρήστης θα έπαιρνε τα αποτελέσματα της εκτέλεσης αυτού του script, αλλά δεν θα ήξερε ποιος είναι ο βασικός κώδικας της σελίδας. Μπορεί ακόμη να ρυθμιστεί ο web server να χειρίζεται όλα τα HTML αρχεία με την PHP, και τότε πραγματικά δεν υπάρχει τρόπος ο χρήστης να καταλάβει σε τι κώδικα είναι γραμμένη μια σελίδα [URL: PHP].

Τα καλύτερο πράγμα στην PHP είναι ότι είναι εξαιρετικά απλή για έναν νέο στον προγραμματισμό αλλά προσφέρει πολλά προηγμένα χαρακτηριστικά για ένα επαγγελματία προγραμματιστή.

3.5.3 MySQL

Το MySQL είναι ένα σύστημα διαχείρισης βάσεων δεδομένων. Μια βάση δεδομένων είναι μια δομημένη συλλογή δεδομένων. Μπορεί να είναι οτιδήποτε, από μια απλή λίστα για ψώνια σε μια πινακοθήκη ή τεράστιες ποσότητες πληροφοριών σε ένα εταιρικό δίκτυο [URL: MySQL].

Για να γίνει προσθήκη, προσπέλαση και επεξεργασία των δεδομένων που αποθηκεύονται σε μια βάση, θα πρέπει να υπάρχει ένα σύστημα διαχείρισης βάσεων δεδομένων όπως η MySQL Server.

Δεδομένου ότι οι υπολογιστές είναι πολύ καλοί στο χειρισμό μεγάλου όγκου δεδομένων, τα συστήματα διαχείρισης βάσεων δεδομένων διαδραματίζουν κεντρικό ρόλο στην πληροφορική, ως αυτόνομες επιχειρήσεις κοινής ωφέλειας, ή ως τμήματα άλλων εφαρμογών.

3.5.4 PHPMyAdmin

Το PhpMyAdmin είναι μία από τις πιο δημοφιλείς εφαρμογές για τη διαχείριση βάσεων δεδομένων MySQL. Είναι ένα δωρεάν εργαλείο γραμμένο σε PHP. Μέσα από αυτό το λογισμικό υπάρχει η δυνατότητα για δημιουργία, επεξεργασία, διαγραφή, εισαγωγή και εξαγωγή πινάκων της βάσης δεδομένων.

ΚΕΦΑΛΑΙΟ 4: Σχεδιασμός και Ανάπτυξη Ιστοχώρου

Για να φτιάξουμε την κύρια σελίδα η οποία θα είναι η βάση μας για όλες τις άλλες σελίδες θα πρέπει να βασιστούμε σε κάποιες κανόνες δημιουργίας ιστοσελίδων έτσι ώστε το site να έχει μια επαγγελματική εμφάνιση και σωστή λειτουργία [URL: TechRepublic].

Προσπαθούμε να αποφεύγουμε ενοχλητικές λειτουργίες

Ο ευκολότερος τρόπος να διώξεις τους επισκέπτες από ένα site είναι να έχεις πολλά “Pop Ups” λειτουργίες τα οποία κάνουν πιο δύσκολη και ενοχλητική την περιήγηση στο site.

Ευχρηστία στην πλοήγηση

Πρέπει να είναι εύκολο για κάθε χρήστη να βρίσκει αυτό που αναζητά με όσο το δυνατόν λιγότερα κλικ. Για ένα site όπου το περιεχόμενο αυξάνει συνεχώς είναι πλέον απαραίτητη η ύπαρξη Sitemap το οποίο μπορεί να διευκολύνει ακόμα περισσότερο την πλοήγηση.

Διαλέγουμε τα σωστά χρώματα

Διαλέγοντας φωτεινά χρώματα δεν θα είναι πολύ λειτουργικό για την σελίδα μας, υπολογίζοντας ότι ένα αρκετό μεγάλο ποσοστό του πληθυσμού υποφέρουν από διάφορα προβλήματα όρασης. Τα χρώματα πρέπει να είναι απαλά και χωρίς μεγάλη αντίθεση μεταξύ τους για να μην είναι κουραστικά στο χρήστη.

Δίνουμε στον χρήστη έναν λόγο να επιστρέψει

Στην δημιουργία μιας ιστοσελίδας πέρα από το design, ένα σημαντικό στοιχείο είναι το περιεχόμενό της. Το περιεχόμενο είναι ο λόγος που ο χρήστης επισκέπτεται ουσιαστικά μια σελίδα και πρέπει να δώσουμε σε αυτό το μεγαλύτερο βάρος.

Γρήγορη φόρτωση σελίδας

Είναι αλήθεια ότι «μια εικόνα ίσον χίλιες λέξεις» όμως χρησιμοποιώντας πολλές εικόνες και άλλα γραφικά στοιχεία, μπορεί να έχουν σαν αποτέλεσμα μια αργή φόρτωση τις σελίδας. Είναι πολύ σημαντικό για τα δεδομένα μιας ιστοσελίδας να κρατάμε το χρόνο φόρτισης κάτω από 4 δευτερόλεπτα οπότε να μην επιβαρύνουμε πολύ τις σελίδες με οπτικοακουστικό υλικό.

4.1 Header



Εικόνα 4.1.1: Header

Το header είναι το σημαντικότερο πράγμα που πρέπει να έχουμε υπόψη κατά την δημιουργία μιας ιστοσελίδας. Αυτό γιατί το Header είναι το πρώτο σημείο που θα δει ο χρήστης κατά την επίσκεψη του στην ιστοσελίδα μας. Ουσιαστικά αυτό αποτελεί και την ταυτότητα του ιστότοπου. Κάποια από τα βασικά στοιχεία που θα χρειαστούμε για την υλοποίηση του header είναι : Λογότυπο, κυρίως μενού, Social media, Rss Feed.

4.1.1 Λογότυπο



Εικόνα 4.1.2: Λογότυπο

Το λογότυπο είναι η εικόνα που αντιπροσωπεύει συμβολικά την ιστοσελίδα μας. Κάποιοι από τους βασικούς κανόνες που πρέπει να ακολουθήσουμε για την δημιουργία ενός σωστού λογότυπου είναι [URL: WebDesignerdepot]:

Απλότητα

Είναι ευκολότερο να απομνημονεύσουμε μια απλή εικόνα από μια περίπλοκη ή από κάποια που έχει πολλά γραφικά στοιχεία.

Παρέχουμε ένα σαφές μήνυμα

Με το λογότυπο πρέπει να καταφέρουμε να περάσουμε ένα σαφές μήνυμα στον χρήστη το οποίο να παραπέμπει στην επιχείρησή μας ή στο προϊόν που επιθυμούμε να προωθήσουμε.

Αξιοπιστία

Στις μέρες μας υπάρχουν πολλές επιχειρήσεις που λειτουργούν στον ίδιο τομέα και έτσι προσπαθώντας να έχουμε ένα μοναδικό και λειτουργικό λογότυπο θα μπορέσουμε να αποκτήσουμε μια αξιοπιστία στο χώρο μας.

4.1.2 Κυρίως Μενού



Εικόνα 4.1.3: Κυρίως Μενού

Η εργασία μας έχει ως βασικό σκοπό την δημιουργία έννους εκπαιδευτικού site πάνω σε διάφορες τεχνολογίες όπως HTML, CSS, Photoshop και για αυτό το λόγο το menu και ειδικά όλο το header πρέπει να είναι κάτι όμορφο και ταυτόχρονα λειτουργικό.

Για την δημιουργία του μενού πλοήγησης έχουμε χρησιμοποιήσει μια εικόνα η οποία προσδίδει ένα φωτεινό εφέ στο menu μας και με την βοήθεια του CSS την καλούμε στην ιδιότητα hover. Στην εμφάνιση του sub menu μας, έχουμε προσθέσει ένα εφέ μετάβασης (transition effect) με την βοήθεια του CSS3, το οποίο έχει ως αποτέλεσμα το submenu να εμφανίζεται σταδιακά. Σκεφτήκαμε ότι θα ήταν ωραίο το background του μενού να αντικατοπτρίζεται πιο κάτω και έτσι να λειτουργήσει σαν μια γέφυρα σύνδεσης ανάμεσα στο header και στην υπόλοιπη σελίδα.

4.1.3 Social Media



Εικόνα 4.1.4: Social Media

Τα Social Media αντιπροσωπεύουν ένα χαμηλού κόστους εργαλείο τα οποία χρησιμοποιούνται από κάθε ιστοσελίδα για την κοινωνική αλληλεπίδραση των χρηστών και πλέον αποτελούν ένα πολύ σημαντικό κομμάτι μιας ιστοσελίδας. Δίνουν στους χρήστες την δυνατότητα να μοιράζονται κείμενα, multimedia και δεδομένα και το όλο εγχείρημα έχει ως αποτέλεσμα την αύξηση της επισκεψιμότητας στην σελίδα και καλύτερη κατάταξη στις μηχανές αναζήτησης.

4.1.4 RSS Feed



Εικόνα 4.1.3: RSS Feed

RSS σημαίνει “Real Simple Syndication” και ακριβώς όπως οι εφημερίδες “εξυπηρετούν” σε καθημερινή βάση τους πολίτες με δομημένο περιεχόμενο έτσι και τα RSS Feed [URL: WikiPedia] δομούν το περιεχόμενο της ιστοσελίδας σας και το προσφέρουν αυτό σε κάθε χρήστη.

Η τεχνολογία των RSS δίνει το πλεονέκτημα στον χρήστη να αποκτήσει πρόσβαση στις πληροφορίες που του παρέχει χωρίς να επισκεφτεί τις σελίδες που αναφέρει το συγκεκριμένο Feed. Η διαδικασία πρόσβασης περιλαμβάνει την χρήση ενός “News Agregator” η “Feed Reader” τα οποία μπορεί είτε να τα χρησιμοποιήσει online σε κάποια από τις σελίδες που είναι διαθέσιμα ή να τα εγκαταστήσει ως μορφή desktop εφαρμογής στον υπολογιστή του.

Με τη διαδικασία αυτή, ο χρήστης μπορεί να δημιουργήσει μια λίστα από RSS πηγές που τον ενδιαφέρουν και να αποκτήσει έτσι μια καθημερινή μόνιμη ενημέρωση μόνο στα θέματα που τον ενδιαφέρουν. Αξιοσημείωτο είναι δε ότι πολλές εταιρίες προσπαθούν να ενσωματώσουν αυτήν την μορφή ενημέρωσης (customized) στις νέες τηλεοράσεις και home cinema.

```

1 <?php
2 header('Content-Type: text/xml');
3
4 DEFINE ('DB_USER', 'womeninh_pti');
5 DEFINE ('DB_PASSWORD', 'ptixiaki2013');
6 DEFINE ('DB_HOST', 'localhost');
7 DEFINE ('DB_NAME', 'womeninh_ptixiaki');
8
9 echo '<?xml version="1.0" encoding="UTF-8"?>
10 <rss version="2.0">
11 <channel>
12 <title>infomm.gr</title>
13 <description>Στην σελίδα μας θα βρείτε δωρεάν μαθήματα για διαφορές
14 web τεχνολογίες όπως Photoshop, HTML, CSS, Ajax και επίσης τεχνολογικά νέα.</description>
15 <link>http://infomm.gr/</link>';
16
17 $connection = @mysql_connect(DB_HOST, DB_USER, DB_PASSWORD)
18 or die('Could not connect to database');
19 mysql_select_db(DB_NAME)
20 or die ('Could not select database');
21 mysql_query("SET NAMES utf8", $connection);
22
23 $get_articles = "SELECT id, pagetitle, description,
24 lastmodified FROM pages ORDER BY id DESC LIMIT 15";
25
26 $articles = mysql_query($get_articles) or die(mysql_error());
27
28 while ($article = mysql_fetch_array($articles)){
29
30     echo '
31         <item>
32             <title>'.$article[pagetitle]. '</title>
33             <description> '.$article[description]. '</description>
34             <link>http://ntokimanter.gr/ptixiaki/index.php?pid='.$article[id]. '</link>
35             <pubDate>'.$article[lastmodified]. '</pubDate>
36         </item>';
37     }
38 echo '</channel>
39 </rss>';
40 ?>

```

Λόγω του ότι χρησιμοποιούμε μια σελίδα η οποία παίρνει το περιεχόμενο δυναμικά από τη βάση της, τότε και το RSS μας θα πρέπει να δημιουργείται με δυναμικό τρόπο έτσι ώστε να είναι πάντα ενημερωμένο και να λειτουργεί αυτόματα.

```

1 <?php
2 header('Content-Type: text/xml');
3
4 DEFINE ('DB_USER', 'womeninh_pti');
5 DEFINE ('DB_PASSWORD', 'ptixiaki2013');
6 DEFINE ('DB_HOST', 'localhost');
7 DEFINE ('DB_NAME', 'womeninh_ptixiaki');
8

```

Στην αρχή πρέπει να δηλώσουμε τα στοιχεία της βάσης μας έτσι ώστε να μπορούμε να συνδεθούμε για να πάρουμε τις πληροφορίες που χρειαζόμαστε.

```

9   echo '<?xml version="1.0" encoding="UTF-8"?>
10  <rss version="2.0">
11  <channel>
12  <title>infomm.gr</title>
13  <description>Στην σελίδα μας θα βρείτε δωρεάν μαθήματα για διαφορές
14  web τεχνολογίες όπως Photoshop, HTML, CSS, Ajax και επίσης τεχνολογικά νέα.</description>
15  <link>http://infomm.gr/</link>';

```

Παρατηρούμε ότι η βασική δομή ενός XML αρχείου δεν διαφέρει πολύ από την δομή ενός HTML αρχείου. Πρέπει να κατανοούμε και να ξεχωρίζουμε τις PHP εντολές από το xml αρχείο μας. Η PHP μαζί με κάποιες βασικές "MYSQL" εντολές μας βοηθάνε για να δημιουργήσουμε ένα δυναμικό RSS Feed. Όπως και η Html, έτσι η xml αποτελείται από ετικέτες (tags) ανοίγματος και κλεισίματος οι οποίες περιέχουν εμφωλευμένες δομές ή πληροφορίες. Η βασική διαφορά τους είναι ότι σε ένα xml αρχείο τα ονόματα των tags μπορεί να τα ορίσουμε εμείς ανάλογα με το αντικείμενο μας.

```

17  $connection = @mysql_connect(DB_HOST, DB_USER, DB_PASSWORD)
18      or die('Could not connect to database');
19  mysql_select_db(DB_NAME)
20      or die ('Could not select database');
21  mysql_query("SET NAMES utf8", $connection);
22
23  $get_articles = "SELECT id, pagetitle, description,
24  lastmodified FROM pages ORDER BY id DESC LIMIT 15";
25
26  $articles = mysql_query($get_articles) or die(mysql_error());
27
28  while ($article = mysql_fetch_array($articles)){
29
30      echo '

```

Στον παραπάνω κώδικα χρησιμοποιούμε "MYSQL" εντολές για να συνδεθούμε με την βάση μας και να πάρουμε από αυτήν τις πληροφορίες που χρειαζόμαστε και που στη δική μας περίπτωση θα είναι: Id, pagetitle, Description και Lastmodified. Προσέξτε από τον παραπάνω κώδικα ότι έχουμε ορίσει το Limit 15 και Order Desc, αυτό ουσιαστικά σημαίνει ότι θέλουμε να εμφανίσουμε μόνο τα 15 τελευταία στοιχεία από τη βάση μας.

```

30      echo '
31          <item>
32              <title>'. $article[pagetitle]. '</title>
33              <description> '. $article[description]. '</description>
34              <link>http://ntokimanter.gr/ptixiaki/index.php?pid=' . $article[id]. '</link>
35              <pubDate>'. $article[lastmodified]. '</pubDate>
36          </item>';
37  }

```

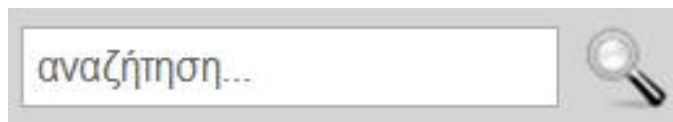
Τέλος χρησιμοποιώντας PHP εντολές δηλώνουμε τις μεταβλητές μας στις xml ετικέτες με τη σειρά που θέλουμε να εμφανιστούν στο RSS Feed.

4.1.5 Metatags

Τα metatags είναι πληροφορίες που εισάγονται ανάμεσα στις ετικέτες του Header και χρησιμοποιούνται συνήθως για να βοηθήσουν τις μηχανές αναζήτησης να κατανοήσουν καλύτερα το περιεχόμενο μιας ιστοσελίδας. Υπάρχουν διάφορα metatags αλλά τα πιο σημαντικά από αυτά είναι τα Μετά Description και Μετά Keyword.

Η google έχει πάψει να υπολογίζει πλέον τα Μετά Keyword και αυτό γιατί πολλές ιστοσελίδες έκαναν κατάχρηση της συγκεκριμένης λειτουργίας για να έχουν καλύτερα αποτελέσματα στις μηχανές αναζήτησης. Υπάρχουν ωστόσο ακόμα κάποιες μηχανές αναζήτησης που τα χρησιμοποιούν όπως Yahoo και Bing και για αυτό το λόγο αποφασίσαμε να τα εντάξουμε στην ιστοσελίδα μας.

4.2 Search Box



Εικόνα 4.2.1: Search Box

Το Search box καθίσταται απαραίτητο εργαλείο σε μια ιστοσελίδα η οποία περιέχει ένα σημαντικό εύρος περιεχομένου το οποίο καθιστά δύσκολη την άμεση ανεύρεση της πληροφορίας από τον χρήστη.

Το κυρίως μενού αντιμετωπίζει αυτό το πρόβλημα αρχειοθετώντας την πληροφορία σε κατηγορίες και υποκατηγορίες ωστόσο στις περισσότερες περιπτώσεις αυτό δημιουργεί προβλήματα αντί να τα λύνει. Εκεί όπου έχουμε μεγάλο όγκο πληροφορίας και χρειαζόμαστε μεγάλο βάθος πλοήγησης για την σωστή ταξινόμησή του, ο χρήστης βρίσκεται σε μειονεκτική θέση καθώς χρειάζεται περισσότερα από 3 κλικ για να αποκτήσει πρόσβαση σε εκείνο για οποίο επισκέφτηκε την ιστοσελίδα.

Το search box μας παρέχει τη δυνατότητα να πραγματοποιήσουμε αναζήτηση (πολλές φορές στο χάος των δεδομένων μιας σελίδας) με βάση συγκεκριμένες λέξεις κλειδιά σχετικά με το περιεχόμενο που μας ενδιαφέρει .

```

144 <?php
145     $squery = $_GET['query'];
146
147     $min_length = 3;
148
149
150     if(strlen($squery) >= $min_length){
151
152         $squery = htmlspecialchars($squery);
153
154
155         $squery = mysql_real_escape_string($squery);
156
157
158         $raw_results = mysql_query("SELECT * FROM pages
159             WHERE ('pagetitle' LIKE '%$squery.%') OR ('linklabel' LIKE '%$squery.%')") or die(mysql_error());
160
161
162         if(mysql_num_rows($raw_results) > 0){
163
164             while($results = mysql_fetch_array($raw_results)){
165                 $pid = $results["id"];
166
167                 echo "<h4><a href='index.php?pid=" . $pid . "'>".$results['linklabel']."</a></h4>";
168                     // posts results gotten from database(title and text) you can also show id ($results['id'])
169                 }
170
171             }
172             else{ // if there is no matching rows do following
173                 echo "<h4>Δεν υπάρχουν αποτελέσματα...</h4>";
174             }
175
176         }
177         else{ // if query length is less than minimum
178             echo "<h4>Χρειάζονται 3 ή περισσότερα γράμματα για να πραγματοποιηθεί μια αναζήτηση... </h4>" ;
179         }
180     ?>

```

Οι βασικές λειτουργίες του παραπάνω κώδικα είναι:

§ Παίρνει τη λέξη κλειδί από το πεδίο αναζήτησης του Search box και την δηλώνει σε μια μεταβλητή.

§ Ελέγχει τον αριθμό των γραμμάτων που ο χρήστης έχει πληκτρολογήσει και εάν αυτό είναι μικρότερο των 3 τότε εμφανίζει ένα σχετικό μήνυμα.

§ Γίνεται αναζήτηση των λέξεων κλειδιών στις εγγραφές της βάσης μας και συγκεκριμένα στα πεδία “title” και “Linklabel”. Είναι προτιμότερο να χρησιμοποιηθούν αυτά τα δύο πεδία από το

πεδίο “content” όπου η λέξη κλειδί μπορεί απλά να αναφέρεται και να μην υπάρχει κάποια ουσιαστική πληροφορία σχετική με αυτή.

§ Εμφανίζει όλους τους τίτλους των αναρτήσεων τα οποία περιλαμβάνουν τη λέξη κλειδί σε μορφή συνδέσμου, Χρησιμοποιώντας το id τις κάθε ανάρτησης. Εάν η αναζήτηση ήταν ανεπιτυχής, εμφανίζει σχετικό μήνυμα.

4.3 Περιεχόμενο

Αν αναλογιστούμε ότι το περιεχόμενο θα πρέπει να εμφανίζεται δυναμικά ανάλογα με την κατηγορία που επιλέγουμε, τότε κατανοούμε ότι έχει μεγάλη σημασία η δομή και η κατηγοριοποίησή του στην βάση δεδομένων [Ann Rockley , 2002]. Αυτό σημαίνει ότι οφείλουμε να προσεγγίσουμε όσο το δυνατόν περισσότερο τα ιδιαίτερα στοιχεία που απαρτίζουν την διαθέσιμη πληροφορία έτσι ώστε να είμαστε σε θέση να έχουμε αποτελεσματικότερες αναζητήσεις στην πορεία.

Η κεντρική ιδέα που χρησιμοποιούμε εδώ ακολουθεί την ίδια δομή με τα queries του search box, χρησιμοποιούμε MySQL εντολές για να πάρουμε τα δεδομένα που θέλουμε από τη βάση μας.

```
20 $sqlCommand = "SELECT pagebody FROM pages WHERE id='$pageid' LIMIT 1";
21 $query = mysql_query($sqlCommand);
22 while ($row = mysql_fetch_array($query)) {
23     $body = $row["pagebody"];
24 }
25 mysql_free_result($query);
```

```
27 $sqlCommand = "SELECT pagetitle, description, keyword FROM pages WHERE id='$pageid' LIMIT 1";
28 $query = mysql_query($sqlCommand) or die (mysqli_error());
29 while ($row = mysql_fetch_array($query)) {
30
31     $pagetitle = $row["pagetitle"];
32     $description = $row["description"];
33     $keyword = $row["keyword"];
34 }
35 mysql_free_result($query);
```

Χρησιμοποιώντας PHP εντολές εμφανίζουμε τα αποτελέσματα της αναζήτησης στο σημείο που επιθυμούμε να εμφανιστούν στην ιστοσελίδα μας.

```
160 <div class="content_main"> <h1><span id="titlos_xroma"><?php echo $pagetitle; ?></span></h1>
161 <div id="pagebody"><?php echo $body; ?></div></div>
```

Παρατηρούμε στην `mysql` ότι η κύρια συνθήκη που βασιζόμαστε για να πάρουμε το περιεχόμενο που θέλουμε είναι η `"id='$pageid'"` και αυτό γιατί για να λειτουργήσει το site πρέπει να υπάρχει μια σύνδεση ανάμεσα στα `pagetitle`, `linklabel` και το `pagebody`. Για να κατανοήσουμε καλύτερα αυτό το ζήτημα θα πρέπει να περιγράψουμε και την λειτουργία των `Linklabels` τα οποία έχουν άμεση σχέση με το περιεχόμενο μας.

4.4 Link Labels

Το `Link labels` ουσιαστικά είναι οι σύνδεσμοι που χρησιμοποιούμε για την εμφάνιση μιας ανάρτησης. Θα μπορούσαμε να χρησιμοποιήσουμε και τον τίτλο τις κάθε ανάρτησης για να κάνουμε την σύνδεση αλλά επιλέξαμε τα `Link Labels` έτσι ώστε να έχουμε μεγαλύτερη ελευθερία. Αυτό συμβαίνει γιατί ο σύνδεσμος μπορεί να είναι διαφορετικός από τον τίτλο της κάθε ανάρτησης.

```

38 $sqlCommand = "SELECT id, linklabel FROM pages WHERE class != 'News' ORDER BY id DESC LIMIT 10 ";
39 $query = mysql_query($sqlCommand);
40
41 $menuDisplay = '';
42 while ($row = mysql_fetch_array($query)) {
43     $pid = $row["id"];
44     $linklabel = $row["linklabel"];
45
46     $menuDisplay .= '<a href="index.php?pid=' . $pid . '">' . $linklabel . '</a><br />';
47
48 }
49 mysql_free_result($query);

```

Παρατηρούμε ότι στην δημιουργία του κάθε συνδέσμου χρησιμοποιούμε μια μεταβλητή `$pid`, η οποία δηλώνει στο τέλος του κάθε url το `id` τις κάθε ανάρτησης. Έτσι για να συνδεθούν όλα μεταξύ τους μας λείπει ένα μικρό κομμάτι κώδικα το οποίο θα πρέπει να πάρει το `id` από κάθε url και να το δηλώσει σε μια μεταβλητή που θα χρησιμοποιήσουμε για την `mysql` συνθήκη μας.

```

14 if (!$_GET['pid']) {
15     $pageid = '180';
16 } else {
17     $pageid = ereg_replace("[^0-9]", "", $_GET['pid']);
18 }

```

Ο κώδικας αυτός παίρνει αυτόματα το `id` από ένα url εκτός εάν εμείς δηλώσουμε το `id` του post που θέλουμε να εμφανίζεται. Αυτό το χρειαζόμαστε επειδή το κυρίως μενού είναι στατικό και πρέπει με κάποιο τρόπο να δηλώσουμε τι θα περιλαμβάνει η πρώτη σελίδα από κάθε κατηγορία.

Έτσι έχουμε καταφέρει να έχουμε μια πλήρη λειτουργία των βασικών μας συνδέσμων.

4.5 Sitemap



Sitemap

Εικόνα 4.5.1: Sitemap

Το sitemap μπορεί να χαρακτηριστεί και ως χάρτης τις ιστοσελίδας. Περιλαμβάνει κάθε περιεχόμενο και αυτό βοηθάει τις μηχανές αναζήτησης για να ερευνήσουν καλύτερα την ιστοσελίδα και ταυτόχρονα μπορεί να τις ενημερώνει για οποιαδήποτε αλλαγή. Η Πλοήγηση στην ιστοσελίδα γίνεται ευκολότερη χρησιμοποιώντας ένα sitemap.

Η δημιουργία του sitemap δεν διαφέρει πολύ από τα παραπάνω παραδείγματα. Πρώτα πρέπει να αποθηκεύσουμε τα δεδομένα τις κάθε κατηγορίας σε μεταβλητές χρησιμοποιώντας MySQL.

```

20 $sqlCommand = "SELECT id, linklabel FROM pages WHERE class = 'HTML' ORDER BY id ASC ";
21 $query = mysql_query($sqlCommand);
22
23 $menuDisplay = '';
24 while ($row = mysql_fetch_array($query)) {
25     $pid = $row["id"];
26     $linklabel = $row["linklabel"];
27
28     $html .= '
29     <a href="index.php?pid=' . $pid . '"> . $linklabel . '</a><br />';
30
31 }
32 mysql_free_result($query);

```

Στο δεύτερο κομμάτι για την υλοποίηση του χάρτη μας θα πρέπει να φτιάξουμε μια λίστα στην οποία στο δυναμικό κομμάτι της θα πρέπει να βάλουμε τους μεταβλητές που φτιάξαμε παραπάνω.

```

227 <div id="sitemap">
228
229 <dl>
230 <dt class="s_section">Σειρές Μαθημάτων </dt><br />
231 <dd class="s_katigories">Μαθήματα HTML</dd><br />
232 <dd><?php echo $html; ?></dd><br />
233 <dd class="s_katigories">Μαθήματα XHTML</dd><br />
234 <dd><?php echo $xhtml; ?></dd><br />
235 <dd class="s_katigories">Μαθήματα CSS</dd><br />
236 <dd><?php echo $css; ?></dd><br />
237 <dd class="s_katigories">Μαθήματα Ajax</dd><br />
238 <dd><?php echo $ajax; ?></dd><br />
239 <dd class="s_katigories">Μαθήματα Photoshop</dd><br />
240 <dd><?php echo $photoshop; ?></dd><br />
241 <dt class="s_section">Νέα στο Διαδίκτυο</dt><br />
242 <dd><?php echo $news; ?></dd>
243 </dl>
244
245 </div>

```

ΚΕΦΑΛΑΙΟ 5: Σχεδιασμός του CMS για τον ιστότοπό μας

Η δημιουργία του administrator panel είναι εκείνη που προσδίδει τον χαρακτηρισμό CMS στη σελίδα μας. Από το administrator panel μπορούμε:

- § Να δημιουργήσουμε μια καινούργια σελίδα
- § Να επεξεργαστούμε τις ήδη υπάρχουσες σελίδες
- § Να δημιουργήσουμε ασκήσεις για την λειτουργία try it

Το site μας έχει ως σκοπό να συμπεριλάβει εκπαιδευτικό υλικό πάνω σε διάφορες τεχνολογίες του internet οπότε από το administrator panel πρέπει με κάποιο τρόπο να γίνεται υλοποίηση της δημιουργίας και επεξεργασίας των try it παραδειγμάτων. Η λειτουργία που επιτελούν τα try it παραδείγματα είναι να βοηθούν τον χρήστη να κατανοήσει καλύτερα ένα παράδειγμα. Επίσης πρέπει με κάποιο τρόπο όλες οι αναρτήσεις του site μας να εμφανίζονται στο administrator panel χωρισμένες σε σελίδες έτσι ώστε να έχουμε το απόλυτο έλεγχο της κάθε ανάρτησης. Τέλος, για λόγους ασφαλείας πρέπει να υπάρχει ένα Login form έτσι ώστε μόνο ο διαχειριστής να μπορεί να χρησιμοποιήσει το administrator panel.

Το Administrator panel ουσιαστικά δίνει στον διαχειριστή την δυνατότητα να μπορεί να δημιουργήσει αναρτήσεις, να επεξεργάζεται τις ήδη υπάρχουσες και να μπορεί να διαγράψει κάποιες από αυτές. Με την εντολή Try it Admin μπορεί να δημιουργήσει παραδείγματα τα οποία θα βοηθούν στον να γίνει ακόμα πιο χρήσιμη μια ανάρτηση – όπως πχ. στις σελίδες εκμάθησης HTML η CSS. Βασικό στοιχείο στην αρχική σελίδα του administrator panel είναι η χρήση ενός Page Navigation κώδικα ο οποίος χρησιμεύει στο να μοιράζουμε το περιεχόμενο της βάσης σε σελίδες όπου σε κάθε σελίδα ορίζουμε τον αριθμό των αναρτήσεων που εμφανίζονται. Οι λειτουργίες «Δημιουργία», «επεξεργασία» και «try it» χρίζουν ιδιαίτερου ενδιαφέροντος και θα αναλυθούν εκτενέστερα σε επόμενες παραγράφους.

5.1 Δημιουργία μιας φόρμας ελέγχου (Login Form)

Ουσιαστικά η λειτουργία τις φόρμας είναι πολύ απλή. Ελέγχει εάν το username και το password ταιριάζουν με αυτά που έχουμε δηλώσει εμείς. Η φόρμα μπορεί να δημιουργηθεί πολύ εύκολα χρησιμοποιώντας HTML κώδικα. Αυτό στο οποίο αξίζει να δώσουμε ιδιαίτερο βάρος είναι η δημιουργία ενός PHP κώδικα ελέγχου ασφάλειας των στοιχείων που εισάγονται για το λογαριασμό χρήστη. Ένας απλός και λειτουργικός τρόπος να γίνει αυτό είναι:

```

1  <?php
2
3  $error_msg = "";
4  if ($_POST['username']) {
5
6      $username = $_POST['username'];
7      $password = $_POST['password'];
8
9      $admin = "ptixiaki";
10     $adminpass = "ptixiaki";
11
12     if (($username != $admin) || ($password != $adminpass)) {
13         $error_msg = '<font color="#FF0000">Λάθος Username ή Password</font>';
14     } else {
15         session_register('admin');
16         $_SESSION['admin'] = $username;
17         require_once "admin.php";
18         exit();
19     }
20 }
21 }
22 ?>

```

Ο κώδικας αυτός παίρνει τα δεδομένα από την φόρμα και τα δηλώνει σε δυο μεταβλητές (\$username , \$password), και μετά ελέγχει εάν αυτές οι μεταβλητές έχουν ίδια τιμή με αυτούς που έχουμε αντιστοιχίσει στον κώδικα (\$admin , \$adminpass). Σε περίπτωση που οι μεταβλητές δεν ταιριάζουν σημαίνει ότι ο χρήστης έχει πατήσει λάθος κωδικό ή password και έτσι η σελίδα θα βγάλει και το κατάλληλο μήνυμα. Διαφορετικά ο χρήστης θα οδηγηθεί στο administrator panel μέσω της εντολής action στο Login form.

5.2 Σελιδοποίηση παρουσίασης αποτελεσμάτων (Pagination)

Η σελιδοποίηση των αποτελεσμάτων που μας επιστρέφει δεδομένα από την βάση μας είναι απαραίτητη όταν έχουμε μεγάλο αριθμό εγγράφων. Για αυτόν το λόγο θα πρέπει να έχουμε μια php function, με την οποία ρυθμίζοντας κάποιους παραμέτρους μπορούμε να δημιουργήσουμε την κατάλληλη σελιδοποίηση.

Για τις ανάγκες τις εργασίας μας χρησιμοποιήσαμε μια έτοιμη php function στην οποία καταλήξαμε μετά από μια μικρή έρευνα σε διάφορα έτοιμα script.

Με την εύρεση του Pagination, έγιναν οι κατάλληλες παραμετροποιήσεις για να προσαρμόσουμε το script αυτό στο δικό μας CMS.

```

1 <?php
2
3     include('../scripts/ps_pagination.php');
4
5
6     $conn = mysql_connect('localhost','root','');
7     mysql_query("SET NAMES utf8", $conn );
8     if(!$conn) die("Failed to connect to database!");
9     $status = mysql_select_db('pt1', $conn);
10    if(!$status) die("Failed to select database!");
11    $sql = 'SELECT * FROM pages ORDER BY id DESC';
12
13    $pager = new PS_Pagination($conn, $sql, 20, 5);
14
15    $pager->setDebug(true);
16
17    ?>

```

Στο παραπάνω κώδικα ουσιαστικά καλούμε το php αρχείο ps_pagination.php από το administrator panel, και εν συνεχεία γίνεται η σύνδεση με την βάση δεδομένων έτσι ώστε να μπορούμε να πάρουμε τα κατάλληλα δεδομένα και τέλος με την εντολή:

```

13     $pager = new PS_Pagination($conn, $sql, 20, 5);

```

μπορούμε να ρυθμίσουμε τον αριθμό των καταχωρίσεων που θα έχει η κάθε σελίδα (20 καταχωρίσεις) και τον αριθμό των λίνκς που θα εμφανίζονται κάτω από κάθε σελίδα χωρίς να παραμετροποιήσουμε το ps_pagination.php αρχείο μας.

Για την εμφάνιση των δεδομένων σε μορφή σελιδοποίησης χρησιμοποιούμε τον παρακάτω κώδικα.

```

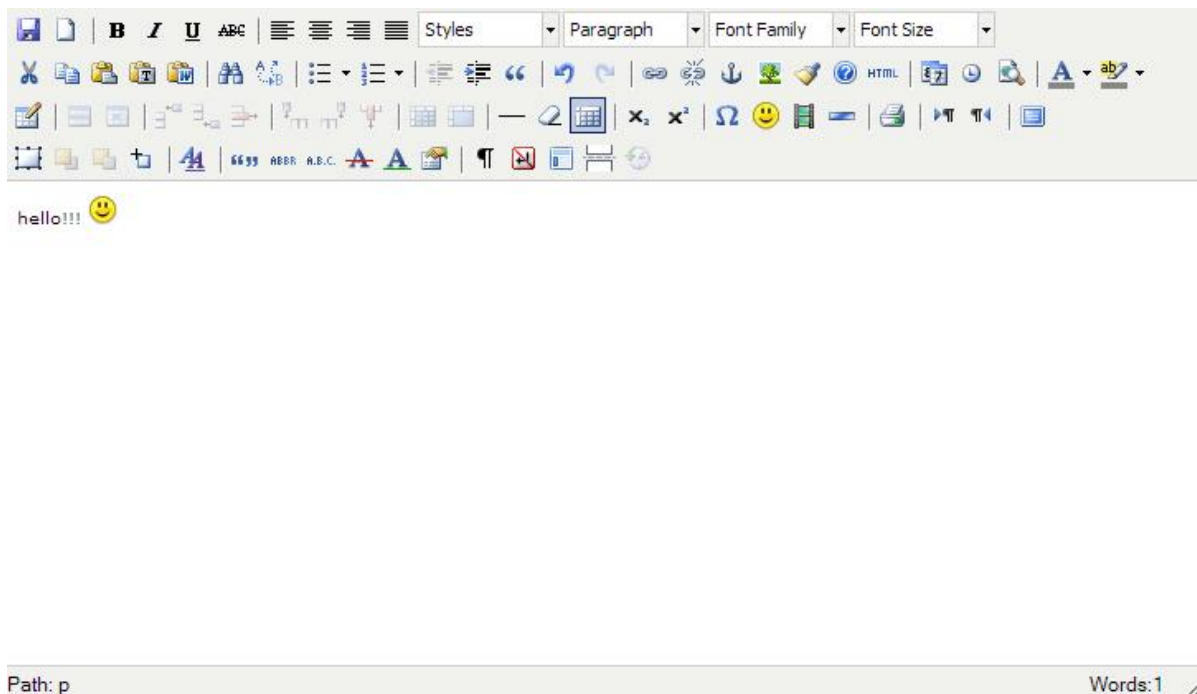
170 <?php $rs = $pager->paginate();
171 if(!$rs) die(mysql_error());
172 while($row = mysql_fetch_assoc($rs)) {
173     $pid = $row["id"];
174     echo "<table cellpadding='0'><tr><td width='40'>";
175         echo $row['id'], "</td><td width='310'>";
176         echo $row['pagetitle'], "</td><td width='90'>";
177         echo $row['class'], "</td><td width='120'>";
178         echo $row['lastmodified'], "</td><td width='70' align='center'>";
179         echo ("<a href='\"edit_page.php?pid=$pid\"'>Edit</a></td><td width='90' align='center'>");
180         echo ("<a href='\"page_delete_page.php?pid=$pid\"'");
181         onclick='\"return confirm(\"Είστε σίγουρος ότι θέλετε να διαγράψετε την σελίδα αυτή:\")\">Delete</a> </td>");
182     echo "</tr></table>";
183 } ?>

```

5.3 TinyMCE Editor

Το TinyMCE, είναι μια αυτόνομη πλατφόρμα βασισμένη σε javascript και HTML, η οποία μπορεί να μας βοηθήσει μετατρέποντας το περιεχόμενο μας σε HTML κώδικα. Το editor αυτό, υποστηρίζει πολλά browser όπως: Mozilla, Internet Explorer, Firefox, Opera, Safari και Chrome [URL: TinyMCE Editor].

Όπως αναφέραμε στο κεφάλαιο 3 ένα από τα πλεονεκτήματα των CMS είναι η ευκαιρία δημιουργίας αναρτήσεων και για αυτό τον λόγο η χρησιμοποίηση ενός editor θεωρείται απαραίτητη για την υλοποίηση της εργασίας μας.



Εικόνα 5.3.1: TinyMCE Editor

Εγκατάσταση του TinyMCE editor

Η εγκατάσταση του TinyMCE editor στην σελίδα μας είναι μια πολύ εύκολη διαδικασία η οποία περιλαμβάνει τα εξής βήματα:

1- Τοποθετούμε ανάμεσα στις ετικέτες <head> τον παρακάτω κώδικα:


```

48 <script type="text/javascript" src="../../js/tiny_mce/tiny_mce.js"></script>
49 <script type="text/javascript">
50   tinyMCE.init({
51     // General options
52     entity_encoding: 'raw',
53     mode: 'textareas',
54     theme: 'advanced',
55     verify_html: false,
56     plugins: 'pagebreak,style,layer,table,advhr,advimage,advlink,emotions,ispell,inlinepopups,
57     insertdatetime,preview,media,searchreplace,print,contextmenu,paste,directionality,fullscreen,noneditable,
58     visualchars,nonbreaking,html4xtras,template,wordcount,advlist,autosave',
59     // Theme options
60     theme_advanced_buttons1: 'save,newdocument,|,bold,italic,underline,strikethrough,|,justifyleft,justifyleft,justifyleft,
61     justifyright,justifyfull,styleselect,formatselect,fontselect,fontsizeselect',
62     theme_advanced_buttons2: 'cut,copy,paste,pastetext,pasteword,|,search,replace,|,bullet,numlist,|,outdent,indent,blockquote,|,undo,redo,|,
63     link,unlink,anchor,image,cleanup,help,code,|,insertdate,inserttime,preview,|,forecolor,backcolor',
64     theme_advanced_buttons3: 'tablecontrols,|,hr,removeformat,visualaid,|,sub,sup,|,charmap,emotions,ispell,
65     media,advhr,|,print,|,ltr,rtl,|,fullscreen',
66     theme_advanced_buttons4: 'insertlayer,moveforward,movebackward,absolute,|,styleprops,|,cite,abbr,acronym,
67     del,ins,attribs,|,visualchars,nonbreaking,template,pagesbreak,restoredraft',
68     theme_advanced_toolbar_location: 'top',
69     theme_advanced_toolbar_align: 'left',
70     theme_advanced_statusbar_location: 'bottom',
71     theme_advanced_resizing: true,
72     // Example content CSS (should be your site CSS)
73     content_css: 'css/content.css',
74     // Drop lists for link/image/media/template dialogs
75     template_external_list_url: 'lists/template_list.js',
76     external_link_list_url: 'lists/link_list.js',
77     external_image_list_url: 'lists/image_list.js',
78     media_external_list_url: 'lists/media_list.js',
79     // Style formats
80     style_formats: [
81       {title: 'Bold text', inline: 'b'},
82       {title: 'Red text', inline: 'span', styles: {color: '#ff0000'}},
83       {title: 'Red header', block: 'h1', styles: {color: '#ff0000'}},
84       {title: 'Example 1', inline: 'span', classes: 'example1'},
85       {title: 'Example 2', inline: 'span', classes: 'example2'},
86       {title: 'Table styles',},
87       {title: 'Table row 1', selector: 'tr', classes: 'tablerow1'}
88     ],
89     // Replace values for the template plugin
90     template_replace_values: {
91       username: 'Some User',
92       staffid: '001234'
93     }
94   });
95 </script>

```

Σε αυτό το σημείο θα πρέπει να προσέξουμε το path που οδηγεί στον φάκελο που βρίσκεται το javascript αρχείο του editor (κώδικας στην γραμμή 48).

2 – Τοποθετούμε σε ένα σημείο της φόρμας μια textarea ετικέτα η οποία αυτόματα με βάση τον παραπάνω κώδικα μετατρέπεται σε ενεργή περιοχή του editor.

```

231 <textarea name="pagebody" cols="88" rows="25" id="pagebody"></textarea>

```

5.4 Δημιουργία καινούργιας σελίδας

Για την δημιουργία καινούργιας σελίδας χρειαζόμαστε δυο βασικές λειτουργίες οι οποίες είναι:

1) Φόρμα καταχώρισης δεδομένων, 2) καταχώριση δεδομένων στην βάση μας.

Φόρμα καταχώρισης δεδομένων

Η φόρμα καταχώρισης δεδομένων μπορεί να διαμορφωθεί χρησιμοποιώντας απλά HTML κώδικα. Βασικά στοιχεία της συγκεκριμένης φόρμας θα είναι:

§ Page Title – Ο τίτλος μιας σελίδας

§ Link Label – Είναι το όνομα του συνδέσμου που μπορεί να έχει η κάθε ανάρτηση. Ο σύνδεσμος πολλές φορές θέλουμε να είναι διαφορετικός από τον τίτλο μιας ανάρτησης και για αυτό τον λόγο χρησιμοποιούμε το Link Label.

§ Page Class - Η κατηγορία στην οποία θα εμφανίζετε η σελίδα

§ Page Body – Είναι το περιεχόμενο μιας ανάρτησης

§ Description (Meta Description) - Είναι μια μικρή περιγραφή κάθε σελίδας η οποία δεν εμφανίζεται κάπου στην σελίδα αλλά χρησιμοποιείτε μόνο από τις μηχανές αναζήτησης.

§ Keywords (Meta Keyword) - Λέξεις κλειδιά τα οποία μπορούν να περιγράψουν καλύτερα το περιεχόμενο της κάθε σελίδας. Χρησιμοποιούνται πλέον ελάχιστα από τις μηχανές αναζήτησης για την κατάταξη μιας σελίδας.

Καταχώριση δεδομένων στην βάση μας

Για την καταχώριση δεδομένων στην βάση μας μπορεί να δημιουργήσουμε μια php function που μπορεί να συλλέξει τα δεδομένα της φόρμας μετά το Submit Button και να τα καταχωρίσει στη βάση δεδομένων χρησιμοποιώντας MySQL εντολές.

```

12 <?php
13 if (isset($_REQUEST['Submit'])) {
14     mysql_query("SET NAMES utf8");
15     $sql = "INSERT INTO `db_table` (pagetitle, class, Categories, linklabel, pagebody, description, keyword, lastmodified) VALUES
16     ('" . mysql_real_escape_string(stripslashes($_REQUEST['pagetitle'])) . "', '" . mysql_real_escape_string(stripslashes($_REQUEST['class'])) . "',
17     '" . mysql_real_escape_string(stripslashes($_REQUEST['Categories'])) . "', '" . mysql_real_escape_string(stripslashes($_REQUEST['linklabel'])) . "',
18     '" . mysql_real_escape_string(stripslashes($_REQUEST['pagebody'])) . "', '" . mysql_real_escape_string(stripslashes($_REQUEST['description'])) . "',
19     '" . mysql_real_escape_string(stripslashes($_REQUEST['keyword'])) . "', NOW()) ";
20     if($result = mysql_query($sql, $db)) {
21         echo "<h1>Thank you</h1>Your information has been entered into our database<br><br>Back to <a href='\"admin.php\">Admin Page</a>";
22     } else {
23         echo "ERROR: " . mysql_error();
24     }
25 } else {
26     ?>

```

Μπορεί ο κώδικας αυτός να σας μπερδεύει και να σας φαίνεται περίεργο αλλά μετά από δόκιμες καταλήξαμε στο συμπέρασμα ότι η συνάρτηση αυτή λειτουργεί καλύτερα όταν έχουμε τεράστιο όγκο δεδομένων σε σχέση με την πιο απλή μορφή της την οποία χρησιμοποιούμε για την επεξεργασία μιας σελίδας.

```

21 mysql_query("SET NAMES utf8", $db);
22 $query = mysql_query("UPDATE pages SET pagetitle='".$pagetitle', class='".$class', linklabel='".$linklabel',
23 Categories='".$Categories', pagebody='".$pagebody', description='".$description', keyword='".$keyword',
24 lastmodified= NOW() WHERE id='".$pid'" ) or die (mysql_error($db));

```

5.5 Επεξεργασία σελίδας

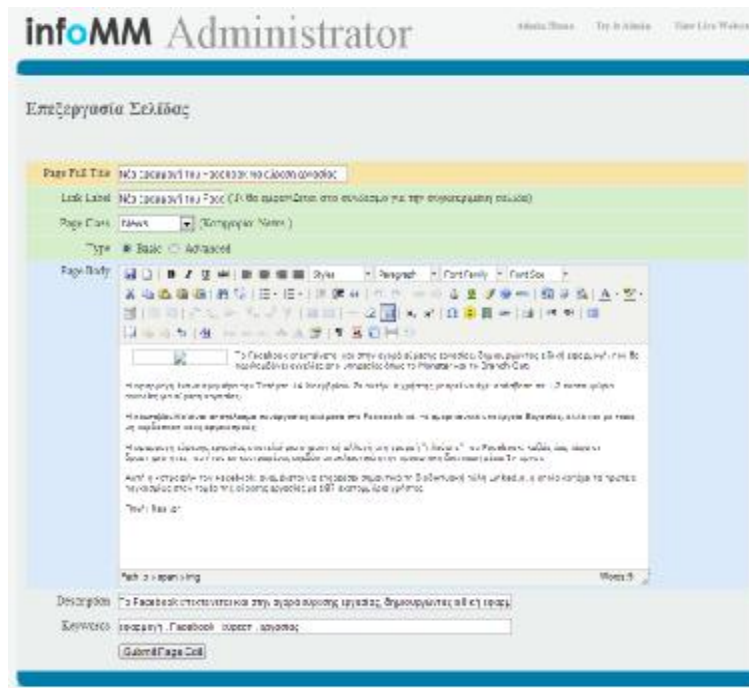
Στην επεξεργασία σελίδας ο χρήστης όπως και σε άλλα CMS πρέπει να μπορεί εύκολα να επεξεργαστεί κάθε ανάρτηση όπως το Administrator Panel. Ουσιαστικά θα πρέπει να υπάρχει ένας σύνδεσμος (Edit) που να μπορεί να μας μεταφέρει στην ανάρτηση αυτή για να μπορούμε να επεξεργαστούμε όλα τα στοιχεία της.

Περιεχόμενο Ιστοσελίδας

ID	Title	Class	Date	Edit Page	Delete Page
201	Οι σύγχρονοι σκλάβοι της Amazon	News	2013-02-26	Edit	Delete
200	Με αργία ταχύτερη η σύνδεση των Ελλήνων στο διαδίκτυο	News	2013-02-26	Edit	Delete
199	Η Microsoft επισύρει τον Messenger	News	2013-02-26	Edit	Delete
198	Καρέκλα φορτίζει συσκευές της Apple ενώ κινείται	News	2013-02-26	Edit	Delete

Εικόνα 5.5.1: Επεξεργασία σελίδας

Εμφανισιακά δεν υπάρχει καμία ουσιαστική διαφορά ανάμεσα στην δημιουργία σελίδας και επεξεργασία σελίδας και κάτι τέτοιο είναι λογικό δεδομένου ότι πρόκειται να επεξεργαστούμε ένα ήδη υπάρχων post το οποίο έχει όλα αυτά τα πεδία που εμείς δηλώσαμε στην φόρμα δημιουργίας ανάρτησης.



Εικόνα 5.5.2: Επεξεργασία σελίδας

Οι μόνες ουσιαστικές διαφορές βρίσκονται πίσω από τη σελίδα στον PHP και MySQL κώδικα.

Εάν στην πρώτη περίπτωση έπρεπε απλά να δηλώσουμε τα δεδομένα μας στη βάση, τώρα προτού κάνουμε κάποια αλλαγή πρέπει πρώτα να καλέσουμε την συγκεκριμένη ανάρτηση από τη βάση για να υποστεί επεξεργασία κάποια από τις τιμές των πεδίων.

```

10 $db = mysql_connect($hostname, $db_user, $db_password);
11 mysql_select_db($database,$db);
12 mysql_query("SET NAMES utf8", $db);
13 $sqlCommand = "SELECT pagetitle, linklabel, Categories, class, description, keyword, pagebody FROM pages WHERE id='\$pid' LIMIT 1";
14 $query = mysql_query($sqlCommand);
15 while($row = mysql_fetch_array($query)){
16     $pagetitle = $row["pagetitle"];
17     $linklabel = $row["linklabel"];
18     $Categories = $row["Categories"];
19     $class = $row["class"];
20     $description = $row["description"];
21     $keyword = $row["keyword"];
22     $pagebody = $row["pagebody"];
23 }
24 }
25 mysql_free_result($query);

```

Οι τιμές αυτές μπορούν να εμφανιστούν σε κάθε πεδίο της φόρμας χρησιμοποιώντας την εντολή “echo”.

```

227 <input name="pagetitle" type="text" id="pagetitle" size="50" maxlength="64" value="<?php echo $pagetitle; ?>" />
228 <input name="linklabel" type="text" id="linklabel" maxlength="24" value="<?php echo $linklabel; ?>" />

```

Μετά το Submit button τα καινούργια δεδομένα περνάνε από ένα php αρχείο το οποίο χρησιμοποιώντας το id της συγκεκριμένης ανάρτησης κάνει update τα καινούργια στοιχεία στη βάση μας.

```

1 <?php
2 $hostname = "localhost";
3 $db_user = "root";
4 $db_password = "";
5 $database = "pt1";
6 $db_table = "pages";
7
8 $db = mysql_connect($hostname, $db_user, $db_password);
9 mysql_select_db($database,$db);
10
11 $pid = $_POST['pid'];
12 $pagetitle = $_POST['pagetitle'];
13 $class = $_POST['class'];
14 $Categories = $_POST['Categories'];
15 $linklabel = $_POST['linklabel'];
16 $pagebody = $_POST['pagebody'];
17 $description = $_POST['description'];
18 $keyword = $_POST['keyword'];
19
20 mysql_query("SET NAMES utf8", $db);
21 $query = mysql_query("UPDATE pages SET pagetitle='\$pagetitle', class='\$class',linklabel='\$linklabel',
22 Categories='\$Categories', pagebody='\$pagebody',description='\$description',keyword='\$keyword',
23 lastmodified= NOW() WHERE id='\$pid'") or die (mysql_error($db));
24
25 echo 'Operation Completed Successfully! <br /><br /><a href="admin.php">Click Here</a>';
26 exit();
27 ?>

```

Εμφάνιση Κατηγορίας – Λίστα Επιλογών

Σε αυτό το κομμάτι υπάρχει ένα μειονέκτημα για την λίστα επιλογών η οποία μας δείχνει την κατηγορία στην οποία βρίσκεται κάθε ανάρτηση. Εδώ για την εμφάνιση της κατηγορίας δεν μπορούμε να χρησιμοποιούμε την `php` εντολή `echo`, διότι η ετικέτα “`select`” δεν μπορεί να πάρει ιδιότητα `value` σε αντίθεση με τις άλλες ετικέτες μιας φόρμας. Για αυτόν τον λόγο για να μπορούμε να εμφανίσουμε την συγκεκριμένη κατηγορία θα πρέπει ανάλογα με την κατηγορία της κάθε ανάρτησης να δηλώσουμε μέσα στην ετικέτα “`option`” την ιδιότητα `selected=` “`selected`”.

Παρακάτω παρουσιάζουμε τον κώδικα `php` ο οποίος ανάλογα με την τιμή της μεταβλητής `class` προσθέτει την ιδιότητα `selected=` “`selected`”, στην αντίστοιχη κατηγορία.

```

34 $a1 = $a2 = $a3 = $a4 = $a5 = $a6 = $a7 = '';
35 if ( $class == 'Home Page' ) {
36     $a1 = "selected='selected'";
37 } elseif ( $class == 'HTML' ) {
38     $a2 = "selected='selected'";
39 } elseif ( $class == 'CSS' ) {
40     $a3 = "selected='selected'";
41 } elseif ( $class == 'XHTML' ) {
42     $a4 = "selected='selected'";
43 } elseif ( $class == 'AJAX' ) {
44     $a5 = "selected='selected'";
45 } elseif ( $class == 'Photoshop' ) {
46     $a6 = "selected='selected'";
47 } elseif ( $class == 'News' ) {
48     $a7 = "selected='selected'";
49 }

```

Τα αποτελέσματα του παραπάνω κώδικα θα πρέπει να εμφανίζονται στην λίστα επιλογών της φόρμας.

```

236 <select name="class" >
237     <option <?php echo $a1; ?> >Home Page</option>
238     <option <?php echo $a2; ?> >HTML</option>
239     <option <?php echo $a3; ?> >CSS</option>
240     <option <?php echo $a4; ?> >XHTML</option>
241     <option <?php echo $a5; ?> >AJAX</option>
242     <option <?php echo $a6; ?> >Photoshop</option>
243     <option <?php echo $a7; ?> >News</option>
244 </select>

```

5.6 Δημιουργία Ασκήσεων με την βοήθεια του Try it

Ένα “Try it” Editor παίζει σημαντικό ρόλο στην κατανόηση παραδειγμάτων πάνω στις διάφορες τεχνολογίες και για αυτό το λόγο πρέπει να καταφέρουμε να ενσωματώσουμε την λειτουργία αυτή στην ιστοσελίδα μας.

Μια απλή κατασκευή αυτού του Editor μπορεί να γίνει χρησιμοποιώντας html/css και Javascript τεχνολογίες. Το Editor αυτό ουσιαστικά θα πρέπει να έχει μια textarea περιοχή στο ένα κομμάτι του, στην οποία ο χρήστης θα πρέπει να επεξεργαστεί τον κώδικα της κάθε άσκησης και ένα iframe στα δεξιά το οποίο θα μας δείχνει κάθε φορά τα αποτελέσματα.

Υπάρχουν διάφορα έτυμα Editor τα οποία μπορούμε να χρησιμοποιήσουμε και να προσαρμόσουμε στις ανάγκες της ιστοσελίδας μας, παρόλα αυτά έχοντας υπόψη ότι ένα σύστημα διαχείρισης περιεχομένου θα πρέπει να δίνει την δυνατότητα δημιουργίας αναρτήσεων εύκολα και γρήγορα το εξελίσαμε ακόμα περισσότερο το Editor αυτό δίνοντας έτσι την δυνατότητα δημιουργίας παραδειγμάτων από το administrator panel.

Η δομή του Try it Editor

Η δομή του Try it Editor είναι πολύ απλή και αυτό μπορούμε να το κατανοήσουμε εάν υπολογίσουμε ότι το Editor αυτό θα πρέπει να περιλαμβάνει τον κώδικα μιας άσκησης αριστερά και να μας δείχνει κάθε φορά το αποτέλεσμα η καλύτερα τι θα εμφανίσει ένας browser διαβάζοντας τον κώδικα αυτόν [URL: Try it Editor].



Εικόνα 5.4.1: Η δομή του Try it Editor

Στην περίπτωση μας το Editor αυτό θα πρέπει να είναι ένα php αρχείο που να μπορεί να καλεί δυναμικά κάθε φορά το παράδειγμα που εμείς θέλουμε χρησιμοποιώντας το id του. Έτσι η περιοχή textarea στην φόρμα μας θα έχει τη παρακάτω μορφή.

```

56 <form style="margin:0px" action="tryit_view.php" method="post" target="view">
57 <input style="margin-bottom:5px;font-family:verdana;" name="submit" type="submit" value="Κάντε κλικ εδώ >>" onclick="displayed()" />
58 <textarea class="code_input" width="100%" height="400px" name="code" wrap="logical" rows="21" cols="42">
59
60 <?php echo $tryit ?>
61
62 </textarea>
63 </form>

```

Χρησιμοποιώντας php μπορούμε κάθε φορά να καλούμε ένα συγκεκριμένο παράδειγμα από την βάση μας.

```

1 <?php
2
3 $hostname = "localhost";
4 $db_user = "root";
5 $db_password = "";
6 $database = "tryit_db";
7 $db_table = "paradeigmata";
8
9 $db = mysql_connect($hostname, $db_user, $db_password);
10 mysql_select_db($database,$db);
11 mysql_query( "SET NAMES utf8", $db );
12
13
14 if (!$_GET['example_id']) {
15     $paradeigma = '1';
16 } else {
17     $paradeigma = ereg_replace("[^0-9]", "", $_GET['example_id']);
18 }
19
20 $sqlCommand = "SELECT pagebody FROM paradeigmata WHERE id='$paradeigma' LIMIT 1";
21 $query = mysql_query($sqlCommand);
22 while ($row = mysql_fetch_array($query)) {
23     $tryit = $row["pagebody"];
24
25 }
26 mysql_free_result($query);
27
28 $sqlCommand = "SELECT pagetitle FROM paradeigmata WHERE id='$paradeigma' LIMIT 1";
29 $query = mysql_query($sqlCommand);
30 while ($row = mysql_fetch_array($query)) {
31
32     $ptitlos = $row["pagetitle"];
33 }
34 mysql_free_result($query);
35
36 ?>

```

Το δεξί μέρος του Editor αποτελείται από μια περιοχή iframe η οποία εμφανίζει στην ίδια σελίδα ένα δεύτερο php αρχείο, που χρειαζόμαστε για την εμφάνιση του αποτελέσματος.

```

67
68 <iframe class="result_output" width="100%" height="400px" frameborder="0" name="view" src="tryit_view.php"></iframe>
69

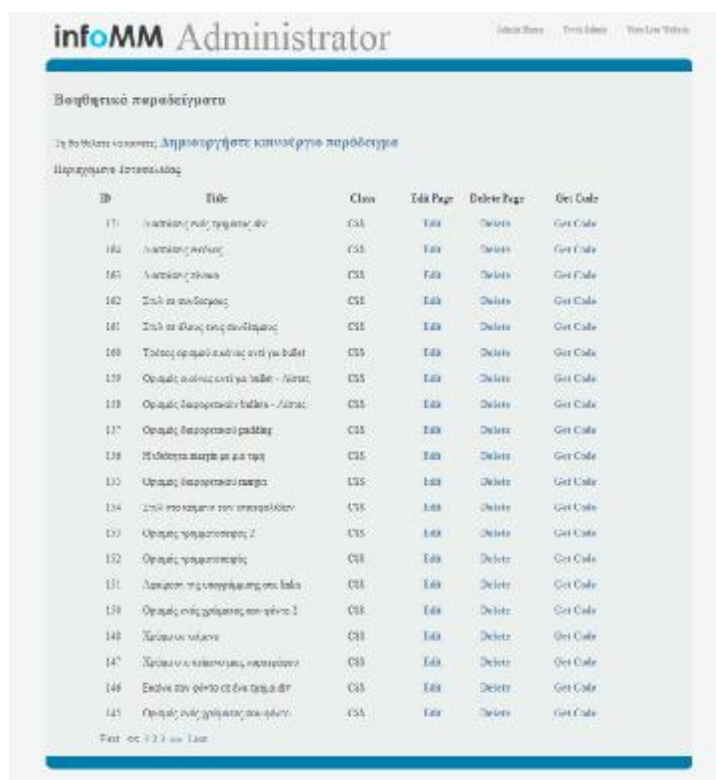
```

Το δεύτερο php αρχείο αποτελείται ουσιαστικά μόνο από μια εντολή echo της μεταβλητής code η οποία περνάει τα δεδομένα μέσω της φόρμας μας.

```
1 <?php
2
3 echo $_POST['code'];
4
5 ?>
```

5.6.1 Δημιουργία ασκήσεων – Administrator Panel

Για την δημιουργία ασκήσεων μέσα από το Administrator Panel θα πρέπει να δημιουργήσουμε ένα καινούργιο php αρχείο το οποίο θα μπορεί να αποθηκεύει τις ασκήσεις σε μια βάση δεδομένων για να μπορούμε να τις καλούμε μέσα από την σελίδα μας.



The screenshot shows the 'infoMM Administrator' interface. At the top, there are navigation links for 'Admin Home', 'Test Home', and 'View User Details'. Below the header, the page title is 'Βοηθητικό παραδείγματα' (Helpful examples). A sub-header reads 'Τη βοήθεια να κάνετε: Δημιουργήστε καινούργια παραδείγματα' (Use the help to: Create new examples). Underneath, it says 'Παράδειγμα: Διατάξεις' (Example: Arrangements). The main content is a table listing various exercises with columns for ID, Title, Class, Edit Page, Delete Page, and Get Code.

ID	Τίτλος	Class	Edit Page	Delete Page	Get Code
171	Λειτουργία ενός τραπέζιου div	CS5	Edit	Delete	Get Code
182	Λειτουργία πινάκων	CS5	Edit	Delete	Get Code
187	Λειτουργία πίνακα	CS5	Edit	Delete	Get Code
182	Σελίδα εισαγωγής	CS5	Edit	Delete	Get Code
181	Σελίδα εισαγωγής συνδέσμου	CS5	Edit	Delete	Get Code
188	Τοποθεσία ομαδών εντός για bulletin	CS5	Edit	Delete	Get Code
119	Ομάδες ονόματι επί για bulletin - Αίτιας	CS5	Edit	Delete	Get Code
118	Ομάδες διαφορετικών bulletin - Αίτιας	CS5	Edit	Delete	Get Code
117	Ομάδες διαφορετικού rank/level	CS5	Edit	Delete	Get Code
118	Η ιδιότητα πατήρι σε μια τάξη	CS5	Edit	Delete	Get Code
115	Ομάδες διαφορετικού πατήρι	CS5	Edit	Delete	Get Code
114	Σελίδα στο κείμενο των υποπεριόδων	CS5	Edit	Delete	Get Code
113	Ομάδες προγραμματισμού 2	CS5	Edit	Delete	Get Code
112	Ομάδες προγραμματισμού	CS5	Edit	Delete	Get Code
111	Αίτιας της υποστήριξης στη βάση	CS5	Edit	Delete	Get Code
110	Ομάδες ενός χρήστη στο φόντο 2	CS5	Edit	Delete	Get Code
140	Χρήστες ονόματι	CS5	Edit	Delete	Get Code
147	Χρήστες ονόματι κείμενο, κείμενο	CS5	Edit	Delete	Get Code
146	Εκπαιδευτικό φόντο σε ένα τραπέζι div	CS5	Edit	Delete	Get Code
145	Ομάδες ενός χρήστη στο φόντο	CS5	Edit	Delete	Get Code

Page 10 of 11

Εικόνα 5.4.2: Δημιουργία ασκήσεων

Έτσι όπως στην επεξεργασία σελίδας και εδώ ο διαχειριστής θα μπορεί να δημιουργεί, επεξεργάζεται ή να διαγράφει κάποιο παράδειγμα. Όλα τα στοιχεία αυτά βασίζονται στην ίδια λογική με την διαχείριση αναρτήσεων την οποία αναλύσαμε πιο πάνω και για αυτόν τον λόγο θεωρούμε πως δεν είναι απαραίτητο να επεκταθούμε περισσότερο στην επεξήγησή τους.

Get Code - Try it Panel

Για την μορφοποίηση και την εμφάνιση του κάθε παραδείγματος στην σελίδα, πέρα από την html πρέπει να χρησιμοποιήσουμε και CSS κώδικα.



Εικόνα 5.4.3: Get Code - Try it Panel

Έτσι σκεφτήκαμε ότι το Tryit panel θα πρέπει να έχει ακόμη μια επιλογή που να μπορεί να δώσει όλο τον html κώδικα μαζί με τα divs στα οποία έχουμε ορίσει μια κλάση και μέσω της css να παίρνει την τελική του εμφάνιση. (Δείτε την εικόνα 5.1). Η επιλογή αυτή μας βοηθά να ενσωματώσουμε ακόμα πιο γρήγορα ένα παράδειγμα στην ανάρτηση μας, έχοντας έτσι ένα πλήρες λειτουργικό Σύστημα Διαχείρισης Περιεχομένου.

Για την υλοποίηση της επιλογής αυτής θα πρέπει πρώτα να δημιουργήσουμε ένα php αρχείο που να μπορεί να καλεί τα δεδομένα του αντίστοιχου παραδείγματος από την βάση μας.

```

1  <?php
2  $pid = ereg_replace("[^0-9]", "", $_GET['pid']);
3
4  $hostname = "localhost";
5  $db_user = "root";
6  $db_password = "";
7  $database = "tryit_db";
8  $db_table = "paradeigmata";
9
10 $db = mysql_connect($hostname, $db_user, $db_password);
11 mysql_select_db($database,$db);
12 mysql_query( "SET NAMES utf8", $db );
13 $sqlCommand = "SELECT pagetitle, pagebody, class FROM paradeigmata WHERE id='$pid' LIMIT 1";
14 $qry = mysql_query($sqlCommand);
15 while ($row = mysql_fetch_array($qry)){
16     $pagetitle = $row["pagetitle"];
17
18     $class = $row["class"];
19
20     $pagebody = $row["pagebody"];
21     $utf8_encode = utf8_encode(htmlentities($pagebody));
22 }
23 mysql_free_result($qry);
24
25 ?>

```

Να εμφανίσει όλο τον κώδικα που θα χρειαστούμε μαζί με τις περιοχές div σε μια textarea περιοχή.

```

50 <form style="margin:0px" action="tryit_view.php" method="post" target="view">
51 <textarea class="code_input" name="code" wrap="logical" rows="21" cols="42">
52
53 <div class="kod_frame">
54     <div class="kod_titlos"> <?php echo $pagetitle; ?> </div>
55     <div class="kodikias">
56 <?php echo utf8_encode(htmlentities($utf8_encode)); ?>
57
58 </div>
59     <div class="kod_link"><a href="tryit/tryit.php?example_id=<?php echo $pid; ?>" target="_blank">
60 </a></div>
61 </div>
62 </textarea>
63 </form>
64

```

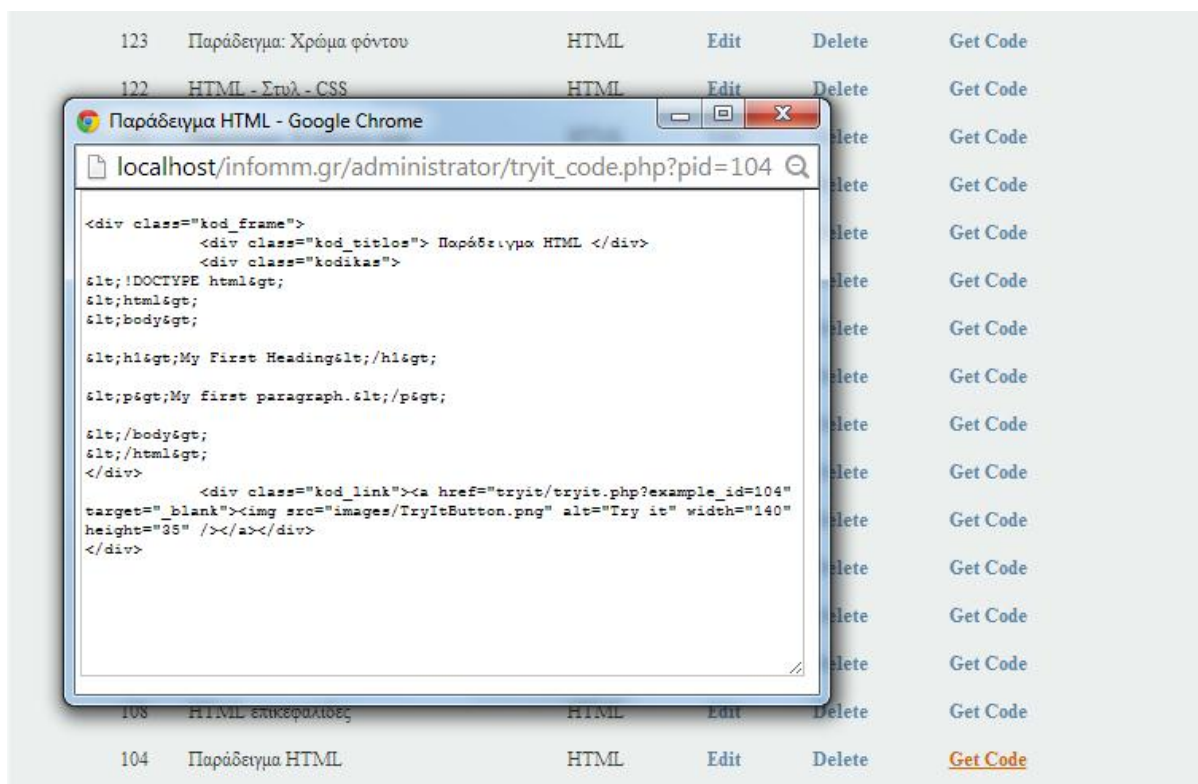
Και τέλος μπορούμε να καλέσουμε τον κώδικα μέσα από την Tryit Panel, ανοίγοντας έτσι ένα καινούργιο παράθυρο χρησιμοποιώντας javascript.

```

176
177 echo ("<a href=\"javascript:void(0)\" onclick=\"window.open('tryit_code.php?pid=$pid',
178 'linkname', 'height=420, width=630,scrollbars=no')\">Get Code</a></td>");
179

```

Τελικό αποτέλεσμα:



Εικόνα 5.4.4: Get Code

ΚΕΦΑΛΑΙΟ 6: Σύνοψη

6.1 Συμπεράσματα

Κατά την εκπόνηση της εργασίας μελετήσαμε τα CMS συστήματα που υπάρχουν στην αγορά των πληροφοριακών συστημάτων και αξιολογήσαμε τα θετικά και τα αρνητικά στοιχεία του καθενός. Βάσει των συμπερασμάτων της αρχικής μας έρευνας, σχεδιάσαμε και υλοποιήσαμε σε PHP ένα νέο δικό μας CMS σύμφωνα με τις προδιαγραφές ενός εκπαιδευτικού αλλά παράλληλα ενημερωτικού site σχετικά με την δημιουργία ιστοσελίδων. Το εκπαιδευτικό site που δημιουργήσαμε αξιολογήθηκε βάσει:

- § Της ευκολίας στην κατανόηση πλοήγησης
- § Της ευκολίας στην πρόσβαση
- § Την ταχύτητα πλοήγησης
- § Του χρόνου που απαιτείται για την πρόσβαση στην πληροφορία από μη γνωστικό χρήστη
- § Την επίτευξη του στόχου εκπαίδευσης του χρήστη
- § Η απόδοση του site σε κάθε ένα από τα παραπάνω test ήταν περισσότερο από ικανοποιητική.

6.2 Μελλοντική Εργασία

Κάποιες από τις ιδέες για την περαιτέρω ανάπτυξη ενός εκπαιδευτικού portal θα ήταν:

- § Η ενσωμάτωση μιας πλατφόρμας ασύγχρονης τηλεεκπαίδευσης (όπως το moodle) με υλικό εγκεκριμένο από ανώτερα και ανώτατα εκπαιδευτικά ιδρύματα.
- § Η προσθήκη ενός συστήματος που θα μπορεί να δέχεται σχόλια χρηστών (blogs) όπου κάθε μαθητευόμενος θα μπορούσε να ανταλλάσσει απόψεις και θέσεις με κάποιον άλλο ή και με προσωπικό υποστήριξης του site.
- § Η ανάθεση προσωπικού με αρμοδιότητες καθημερινής ενημέρωσης δεδομένων και εξυπηρέτησης χρηστών στην χρήση του site και στην εκμάθηση των αντικειμένων που προβάλλονται.

Βιβλιογραφία

Έντυπη

Dr Andreas Mauthe & Dr Peter Thomas (2004). Professional Content Management Systems: Handling Digital Media Assets

Bob Boiko (2001). Content Management Bible. ISBN-10: 076454862X

Ann Rockley (2002). Managing Enterprise Content: A Unified Content Strategy

Robin Nixon (2009). Learning PHP, MySQL, and JavaScript: A Step-By-Step Guide to Creating Dynamic Websites

Ηλεκτρονική

[W3 Schools] at <http://www.w3schools.com/>

[Wlearn] at <http://www.wlearn.gr/>

[Ελληνικά Βοηθήματα] at <http://www.greektuts.net/>

[PHP.net] at <http://php.net/>

[MySQL] at <http://www.mysql.com/>

[Wamp Server] at <http://www.wampserver.com/en/>

[Joomla] at <http://www.joomla.org/>

[Wordpress] at <http://wordpress.org/>

[Drupal] at <https://drupal.org/>

[TinyMCE Editor] at <http://www.tinymce.com/>

[WikiPedia] at <http://www.wikipedia.org/>

[Try it Editor] at <http://tryiteditor.webs.com/>

[Makeuseof] at <http://www.makeuseof.com/tag/10-popular-content-management-systems-online/>

[TechRepublic] at <http://www.techrepublic.com/article/13-design-rules-that-every-web-site-designer-should-know-about/>

[WebDesignerdepot] at <http://www.webdesignerdepot.com/2009/06/12-essential-rules-to-follow-when-designing-a-logo/>

ΠΑΡΑΡΤΗΜΑ

1. Σχεδιασμός και Ανάπτυξη Ιστοχώρου

1.1 Home Page (index.php)

```
<?php

$hostname = "localhost";

$db_user = "root";

$db_password = "";

$databse = "ptixiaki";

$db_table = "pages";

$db = mysql_connect($hostname, $db_user, $db_password);

mysql_select_db($databse,$db);

mysql_query( "SET NAMES utf8", $db );

if (!$_GET['pid']) {

    $pageid = '180';

} else {

    $pageid = ereg_replace("[^0-9]", "", $_GET['pid']);

}

$sqlCommand = "SELECT pagebody FROM pages WHERE id='$pageid' LIMIT 1";

$query = mysql_query($sqlCommand);

while ($row = mysql_fetch_array($query)) {

    $body = $row["pagebody"];

}

mysql_free_result($query);

$sqlCommand = "SELECT pagetitle, description, keyword FROM pages WHERE id='$pageid' LIMIT 1";

$query = mysql_query($sqlCommand) or die (mysql_error());

while ($row = mysql_fetch_array($query)) {

    $pagetitle = $row["pagetitle"];

    $description = $row["description"];
```

```

        $keyword = $row["keyword"];
    }

    mysql_free_result($query);

    $sqlCommand = "SELECT id, linklabel FROM pages WHERE class != 'News' ORDER BY id DESC LIMIT 10 ";
    $query = mysql_query($sqlCommand);
    $menuDisplay = "";
    while ($row = mysql_fetch_array($query)) {
        $pid = $row["id"];
        $linklabel = $row["linklabel"];

        $menuDisplay .= '<a href="index.php?pid=' . $pid .
        "'>'. $linklabel . '</a><br />';
    }

    mysql_free_result($query);

    $sqlCommand = "SELECT id, linklabel FROM pages WHERE class='News' ORDER BY id DESC LIMIT 10 ";
    $query = mysql_query($sqlCommand) or die (mysql_error());
    $advanced = "";
    while ($row = mysql_fetch_array($query)) {
        $pid = $row["id"];
        $linklabel = $row["linklabel"];

        $advanced .= '<a href="index.php?pid=' . $pid . "'>'.
        $linklabel . '</a><br />';
    }

    mysql_free_result($query);
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

```

```

<title><?php echo $pagetitle; ?></title>

<meta content="<?php echo $description; ?> " name="description"/>

<meta content=" <?php echo $keyword; ?> " name="keywords"/>

<link rel="shortcut icon" href="favicon.ico" />

<link href="css/style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<div id="main_wrapper">

<div id="wrapper">

    <div id="top_space"></div>

        <div id="header">

            <div id="site_title">

                <a href="index.php" target="_parent"><span>Informatics and Mass
Media</span></a></div>

                <!-- end of site_title -->

            <div id="header_right">

                <div id="socialmedia">

                    <table width="255" align="right">

<tr>

    <td><a href="#"></a></td>

    <td><a href="#"></a></td>

    <td><a href="feed"></a></td>

    <td> <div class="socialtext">Follow Me :)</div></td>

</tr>

</table>

        </div>

```

```

    <div class="cleaner"></div>

</div>

<div class="cleaner"></div>

</div>

<div id="main_menu">

<div class="dropMenu">

    <ul>

<li class="current"><a href="index.php"><b>Home</b><!--[if lte IE
6]><table><tr><td></td></tr></table><![endif]--></a></li>

<li><a href="#"><b>Σειρές Μαθημάτων</b><!--[if gte IE 7]><!--></a><!--<![endif]-->
<!--[if lte IE 6]><table><tr><td><![endif]-->

<div>

<ul>

<li class="dropMenu_b"><a href="html.php">HTML</a></li><li><a href="xhtml.php">XHTML</a></li><li><a
href="css.php">CSS</a></li><li><a href="ajax.php">AJAX</a></li><li><a
href="photoshop.php">Photoshop</a></li>

</ul>

</div>

<!--[if lte IE 6]></td></tr></table></a><![endif]-->

</li>

<li><a href="news.php">Νέα στο Διαδίκτυο<!--[if lte IE 6]><table><tr><td></td></tr></table><![endif]--
></a></li>

<li><a href="forum/index.php" target="_blank">Forum<!--[if lte
IE6]><table><tr><td></td></tr></table><![endif]--></a></li>

<li><a href="contact.php"><b>Επικοινωνία</b><!--[if gte IE 7]><!--></a><!--<![endif]-->
<!--[if lte IE 6]><table><tr><td><![endif]-->

</li><li class="last"><a href="#url"><!--[if lte IE 6]><table><tr><td></td></tr></table><![endif]--></a></li>

</ul>

</div></div>

<div id="content_wrapper"></div>

```



```

<div id="c_main">
  <div id="content_left">
<div class="content_main"> <h1><span id="titlos_xroma"><?php echo $pagetitle; ?></span></h1>
  <div id="pagebody"><?php echo $body; ?></div></div>
</div>
  <div id="content_right">
  <div id="sidebar">
<div id="searchbox"><form action="search.php" method="GET">
  <input id="searchtext" type="text" name="query" value="αναζήτηση..." onclick="this.value='';" />
  <input type="image" id="gosearch" src="images/search-button.png" height="35" width="35"
value="Search"/>
</form></div>
  <div class="sidebar_title">Τελευταία Μαθήματα</div>
  <div class="sidebar_box">
  <?php echo $menuDisplay; ?>
</div>
  <div class="cleaner_h20"></div>
  <div class="sidebar_title">Τελευταία Νέα</div>
  <div class="sidebar_box">
  <?php echo $advanced; ?>
</div>
  <div class="cleaner_h20"></div>
  <div class="sidebar_title">Οι χορηγοί μας</div>
  <div class="sidebar_box">
  <table width="210" cellpadding="5"><tr>
  <td><a href="#"></a></td>
  <td><a href="#"></a></td> </tr><tr>

```

```

<td><a href="#"></a></td>

<td><a href="#"></a></td></tr>

<tr><td><a href="#"></a></td>

    <td><a href="#"></a></td> </tr><tr>

<td><a href="#"></a></td>

<td><a href="#"></a></td></tr>

</table>

</div></div></div>

    <div class="cleaner"></div>

</div>

<div id="bottom_round"></div>

<div id="footer">

    <a href="sitemap.php">Sitemap</a> - <a href="about.php">Σχετικά με εμάς</a> - <a
href="termsfuse.php">Όροι Χρήσης</a> - <a href="news.php">Νέα στο Διαδίκτυο</a> - <a href="forum"
target="_blank">Forum</a> - <a href="advertise.php">Διαφημιστείτε μαζί μας</a> - <a
href="contact.php">Επικοινωνία </a></div>

</div></div>

</body>

</html>

```

1.2 RSS Feed

```

<?php
header('Content-Type: text/xml');
DEFINE ('DB_USER', 'root');
    DEFINE ('DB_PASSWORD', '');
    DEFINE ('DB_HOST', 'localhost');

```

```

    DEFINE ('DB_NAME', 'ptixiaki');
echo '<?xml version="1.0" encoding="UTF-8"?>
<rss version="2.0">
<channel>
<title>infomm.gr</title>
<description>Στην σελίδα μας θα βρείτε δωρεάν μαθήματα για διαφορές web τεχνολογίες όπως Photoshop, HTML,
CSS, Ajax και επίσης τεχνολογικά νέα.</description>
<link>http://infomm.gr</link>;

$connection = @mysql_connect(DB_HOST, DB_USER, DB_PASSWORD)
    or die('Could not connect to database');
mysql_select_db(DB_NAME)
    or die ('Could not select database');
mysql_query("SET NAMES utf8", $connection);
$get_articles = "SELECT id, pagetitle, description,
lastmodified FROM pages ORDER BY id DESC LIMIT 15";
$articles = mysql_query($get_articles) or die(mysql_error());
while ($article = mysql_fetch_array($articles)){

    echo '
        <item>
            <title>'. $article[pagetitle]. '</title>
            <description> '. $article[description]. '</description>
            <link>http://ntokimanter.gr/ptixiaki/index.php?pid='.\$article\[id\]. '</link>
            <pubDate>'. \$article\[lastmodified\]. '</pubDate>
        </item>;
    }
echo '</channel>
</rss>;
?>

```

1.3 Search Box - Αποτελέσματα αναζήτησης

```

<?php
$hostname = "localhost";
$db_user = "root";
$db_password = "";
$database = "ptixiaki";

```

```

$db_table = "pages";
if (!$_GET['pid']) {
    $pageid = '180';
} else {
    $pageid = ereg_replace("[^0-9]", "", $_GET['pid']);
}
$db = mysql_connect($hostname, $db_user, $db_password);
mysql_select_db($database,$db);
mysql_query( "SET NAMES utf8", $db );
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class != 'News' ORDER BY id DESC LIMIT 10 ";
$query = mysql_query($sqlCommand);
$menuDisplay = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $menuDisplay .= '<a href="index.php?pid=' . $pid .
    "'> . $linklabel . </a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class='News' ORDER BY id DESC LIMIT 10 ";
$query = mysql_query($sqlCommand) or die (mysql_error());
$advanced = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $advanced .= '<a href="index.php?pid=' . $pid . "'> .
    $linklabel . </a><br />';
}
mysql_free_result($query);

?>

```

Εμφάνιση αποτελεσμάτων αναζήτησης

```

<?php
                $squery = $_GET['query'];
// gets value sent over search form
    $min_length = 3;
// you can set minimum length of the query if you want

```

```

if(strlen($squery) >= $min_length){ // if query length is more or equal minimum length then
    $squery = htmlspecialchars($squery);
    // changes characters used in html to their equivalents, for example: < to &lt;
    $squery = mysql_real_escape_string($squery);
    // makes sure nobody uses SQL injection
    $raw_results = mysql_query("SELECT * FROM pages
        WHERE (`pagetitle` LIKE '%" . $squery . "%') OR (`linklabel` LIKE '%" . $squery . "%')") or die(mysql_error());
    if(mysql_num_rows($raw_results) > 0){ // if one or more rows are returned do following
        while($results = mysql_fetch_array($raw_results)){
            $pid = $results['id'];
echo "<h4><a href='index.php?pid=" . $pid . "'>". $results['linklabel']. "</a></h4>";
            // posts results gotten from database(title and text) you can also show id ($results['id'])
        }
    }
    else{ // if there is no matching rows do following
        echo "<h4>Δεν υπάρχουν αποτελέσματα...</h4>";
    }
}
else{ // if query length is less than minimum
    echo "<h4> Χρειάζονται 3 η περισσότερα γράμματα για να πραγματοποιηθεί μια αναζήτηση... </h4>";
}
?>

```

1.4 Sitemap

```

<div class="content_main"> <h1>Sitemap</h1>
    <div id="sitemap">
        <dl>
            <dt class="s_section">Σειρές Μαθημάτων </dt><br />
            <dd class="s_katigories">Μαθήματα HTML</dd><br />
            <dd><?php echo $html; ?></dd><br />
            <dd class="s_katigories">Μαθήματα XHTML</dd><br />
            <dd><?php echo $xhtml; ?></dd><br />
            <dd class="s_katigories">Μαθήματα CSS</dd><br />
            <dd><?php echo $css; ?></dd><br />
            <dd class="s_katigories">Μαθήματα Ajax</dd><br />
            <dd><?php echo $ajax; ?></dd><br />
            <dd class="s_katigories">Μαθήματα Photoshop</dd><br />

```

```

<dd><?php echo $photoshop ; ?></dd><br />
<dt class="s_section">Νέα στο Διαδίκτυο</dt><br />
<dd><?php echo $news; ?></dd>
</dl>
</div></div>

```

PHP Κώδικας

```

<?php
$hostname = "localhost";
$db_user = "root";
$db_password = "";
$database = "ptixiaki";
$db_table = "pages";
$db = mysql_connect($hostname, $db_user, $db_password);
mysql_select_db($database,$db);
mysql_query("SET NAMES utf8", $db );

if (!$_GET['pid']) {
    $pageid = '180';
} else {
    $pageid = ereg_replace("[^0-9]", "", $_GET['pid']);
}

$sqlCommand = "SELECT id, linklabel FROM pages WHERE class = 'HTML' ORDER BY id ASC ";
$query = mysql_query($sqlCommand);
$menuDisplay = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $html .= '<a href="index.php?pid=' . $pid . "'> . $linklabel .
</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class = 'XHTML' ORDER BY id ASC ";
$query = mysql_query($sqlCommand);
$menuDisplay = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];

```

```

    $xhtml .= '<a href="index.php?pid=' . $pid . "'>'.
$linklabel . '</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class = 'CSS' ORDER BY id ASC ";
$query = mysql_query($sqlCommand);
$menuDisplay = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $css .= '<a href="index.php?pid=' . $pid . "'>'.
$linklabel . '</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class = 'AJAX' ORDER BY id ASC ";
$query = mysql_query($sqlCommand);
$menuDisplay = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $ajax .= '<a href="index.php?pid=' . $pid . "'>'. $linklabel .
'</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class='Photoshop' ORDER BY id DESC ";
$query = mysql_query($sqlCommand) or die (mysql_error());
$advanced = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $photoshop .= '<a href="index.php?pid=' . $pid . "'>'.
$linklabel . '</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class='News' ORDER BY id DESC ";
$query = mysql_query($sqlCommand) or die (mysql_error());
$advanced = "";
while ($row = mysql_fetch_array($query)) {

```

```

    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $news .= '<a href="index.php?pid=' . $pid . "'>' . $linklabel .
'</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class != 'News' ORDER BY id DESC LIMIT 10 ";
$query = mysql_query($sqlCommand);
$menuDisplay = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $menuDisplay .= '<a href="index.php?pid=' . $pid . "'>' .
$linklabel . '</a><br />';
}
mysql_free_result($query);
$sqlCommand = "SELECT id, linklabel FROM pages WHERE class='News' ORDER BY id DESC LIMIT 10 ";
$query = mysql_query($sqlCommand) or die (mysql_error());
$advanced = "";
while ($row = mysql_fetch_array($query)) {
    $pid = $row["id"];
    $linklabel = $row["linklabel"];
    $advanced .= '<a href="index.php?pid=' . $pid . "'>' .
$linklabel . '</a><br />';
}
mysql_free_result($query);
?>

```

2. Σύστημα Διαχείρισης Περιεχομένου (CMS)

2.1 Login Form

```

<?php
$error_msg = "";
if ($_POST['username']) {
    $username = $_POST['username'];
    $password = $_POST['password'];
    $admin = "ptixiaki";
    $adminpass = "ptixiaki";

```



```

if (($username != $admin) || ($password != $adminpass)) {
    $error_msg = '<font color="#FF0000">Λάθος Username ή Password</font>';
} else {
    session_register('admin');
    $_SESSION['admin'] = $username;
    require_once "admin.php";
    exit();
}
}
?>
<div id="admin_login">
<?php
if ($_SESSION['admin'] != "gianni84") {
    echo '
<table width="250" border="0" align="center">
<form action="admin.php" method="post" target="_self">
<tr><td align="center"></td></tr>
<tr><td> . $error_msg . '</td></tr>
<tr><td>
<table id="login_forma" width="250" border="0" cellpadding="4">
<tr><td>Username:<br /><input type="text" name="username" id="username" size="18" /></td> </tr>
<tr><td>Password:<br /> <input type="password" name="password" id="password" size="18" /></td></tr>
<tr><td ><INPUT TYPE="image" SRC=" ../images/login-button.jpg" HEIGHT="30" WIDTH="85" BORDER="0"
ALT="Submit Form"></td>
</tr></table> </td></tr>
<td align="right" ><br/><a href=".."> ← Back to the homepage</a></td></tr>
</form>
</table><br /><br /><br />';
exit();
}
?>
</div>

```

2.2 Διαχείριση Περιεχομένου (admin.php)

```

<?php
    include('../scripts/ps_pagination.php');
    $conn = mysql_connect('localhost','root','');
    mysql_query( "SET NAMES utf8", $conn );
    if(!$conn) die("Failed to connect to database!");
    $status = mysql_select_db('ptixiaki', $conn);
    if(!$status) die("Failed to select database!");
    $sql = 'SELECT * FROM pages ORDER BY id DESC';
    $pager = new PS_Pagination($conn, $sql, 20, 5);
    $pager->setDebug(true);
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Διαχείριση περιεχομένων</title>
<script type="text/javascript">
function validate_form1 ( ) {
    valid = true;
    if ( document.form1.pid.value == "" ) {
        alert ( "Please enter the ID number for the page to be edited." );
        valid = false;
    }
    return valid;
}
function validate_form2 ( ) {
    valid = true;
    if ( document.form2.pid.value == "" ) {
        alert ( "Please enter the ID number for the page to be deleted." );
        valid = false;
    }
    return valid;
}
</script>
</head>

```



```

        <td class="datetext">&nbsp;&nbsp;&nbsp;<span class="style4">Class</span></td>
        <td class="classtext">&nbsp;&nbsp;&nbsp;<span class="style4">Date</span></td>
        <td class="edittext"><strong>Edit Page </strong></td>
        <td class="style5">Delete Page </td>
    </tr>
    <tr>
        <td colspan="6"><?php $rs = $pager->paginate();
            if(!$rs) die(mysql_error());
            while($row = mysql_fetch_assoc($rs)) {
                $pid = $row["id"];
                echo "<table cellpadding='8'><tr><td width='40'>";
                    echo $row['id'], "</td><td width='310'>";
                    echo $row['pagetitle'], "</td><td width='90'>";
                    echo $row['class'], "</td><td width='120'>";
                    echo $row['lastmodified'], "</td><td width='70' align='center'>";
                    echo ("<a href='edit_page.php?pid=$pid'>Edit</a></td><td width='90' align='center'>");
                    echo ("<a href='page_delete_parse.php?pid=$pid' onclick='return confirm('Είστε σίγουρος ότι
θέλετε να διαγράψετε την σελίδα αυτή;')'>Delete</a> </td>");
                echo "</tr></table>";
            } ?></td></tr>
    <tr>
        <td colspan="6"><?php echo $pager->renderFullNav();?></td></tr>
    </table></td></tr></table>
</td></tr></table>
</div>
<div id="endline"></div>
</td></tr>
</table>
<br/><br/>
</body>
</html>

```