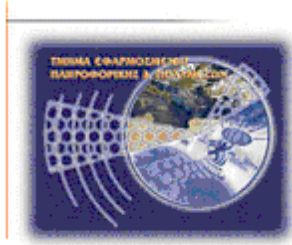




**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης**

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



**Πτυχιακή εργασία**

**Τίτλος: Διαδικτυοκεντρικό Σύστημα για on-line  
διαχείριση των πτυχιακών με XML**

**Κρικώνης Νικόλαος (ΑΜ:1282)**

**Επιβλέπων καθηγητής :Παπαδάκης Νικόλαος**

**Επιτροπή Αξιολόγησης :**

**Ημερομηνία παρουσίασης:**

## Ευχαριστίες

Με την ολοκλήρωση της πτυχιακής εργασίας θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κύριο Νικόλαο Παπαδάκη για την βοήθεια, την στήριξη και την αμέριστη συμπαράσταση. Θα ήθελα να ευχαριστήσω την οικογένεια μου για την συμπαράσταση στην ολοκλήρωση της πτυχιακής εργασίας αλλά και των σπουδών όλα αυτά τα χρόνια.

## **Abstract**

The purpose of this project is to create a reliable application that effectively, easily and quickly attaches to the important processes for the Manage Theses for Education Foundation.

The main technologies used are HTML and PHP to develop dynamic websites. Was studying and using XML for storage, processing and data management. For the implementation of the application used the free software XAMPP.

## Σύνοψη

Σκοπός της παρούσας πτυχιακής είναι η δημιουργία μιας αξιόπιστης εφαρμογής η οποία με αποτελεσματικότητα, ευκολία και ταχύτητα να αποδίδει τις σημαντικές διεργασίες για την Διαχείριση των Πτυχιακών Εργασιών ενός Εκπαιδευτικού Ιδρύματος.

Οι βασικές τεχνολογίες που χρησιμοποιήσα είναι η HTML και η PHP για την ανάπτυξη της δυναμικής ιστοσελίδας. Έγινε μελέτη και χρήση της XML για την αποθήκευση, επεξεργασία και διαχείριση των δεδομένων. Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε και το ελεύθερο λογισμικό XAMPP.

## Περιεχόμενα

Ευχαριστίες.....	2
Abstract.....	3
Σύνοψη.....	4
Περιεχόμενα.....	5
Πίνακας εικόνων.....	7
Κεφάλαιο 1.....	8
Εισαγωγή.....	8
1.1 Περίληψη.....	8
1.2 Κίνητρο για την Διεξαγωγή της Εργασίας.....	8
1.3 Σκοπός της Εργασίας.....	8
1.4 Δομή Εργασίας.....	9
Κεφάλαιο 2.....	10
2.1 Μέθοδος Ανάλυσης και Ανάπτυξης Πτυχιακής.....	10
2.2 Θεωρίες.....	10
2.2.1 Παγκόσμιος Ιστός.....	10
2.2.2 Διαδίκτυο.....	10
2.2.3 Web 2.0.....	12
2.2.4 Web 3.0 Σημασιολογικός Ιστός.....	13
2.2.5 Τι είναι μια Ιστοσελίδα.....	17
2.2.6 Φυλλομετρητής ιστοσελίδων.....	17
2.2.7 Κατασκευή ιστοσελίδων.....	18
2.2.8 Στατική ιστοσελίδα.....	21
2.2.9 Δυναμική ιστοσελίδα.....	22
Κεφάλαιο 3.....	24
Τεχνολογίες και Εργαλεία Ανάπτυξης.....	24
3.1 Τι είναι HTML.....	24
3.2 Η Γλώσσα PHP.....	28
3.3 Εισαγωγή στις Βάσεις Δεδομένων.....	33
3.4 XML.....	28
3.4.1 Εισαγωγή στην XML.....	38
3.4.2 Web-formed και XML DOM.....	46
3.4.3 Προσπέλαση αρχείων XML.....	47
3.4.3.1 XML Parser.....	47
3.4.3.2 Τι είναι SimpleXML.....	48
3.5 Τι είναι ο Apache Web Server.....	49
3.6 Τι είναι XAMPP.....	50
Κεφάλαιο 4.....	52
Ανάλυση Προβλήματος.....	52
4.1 Γενική περιγραφή.....	52
4.2 Διαδικασία ανάπτυξης.....	53
4.3 Εγκατάσταση Λογισμικού XAMPP.....	54
4.4 Σχεδίαση Βάσης δεδομένων.....	62
4.5 Υλοποίηση της Εφαρμογής.....	64
4.5.1 Αρχική σελίδα.....	64
4.5.2 Καταχώρηση Στοιχείων Φοιτητή.....	65
4.5.3 Καταχώρηση Στοιχείων Καθηγητών.....	69
4.5.4 Καταχώρηση Στοιχείων Πτυχιακών Εργασιών.....	72
4.5.5 Εμφάνιση στοιχείων Φοιτητών.....	75
4.5.6 Εμφάνιση στοιχείων Καθηγητών.....	77
4.5.7 Εμφάνιση όλων των Πτυχιακών.....	78
4.5.8 Εμφάνιση Ελεύθερων Πτυχιακών.....	80

4.5.9 Εμφάνιση Φοιτητών που έχουν κατοχυρώσει πτυχιακή.....	81
4.5.10 Ανάθεση Πτυχιακής σε Φοιτητή.....	83
4.5.11 Ενημέρωση Βαθμού Πτυχιακής.....	86
4.5.12 Αποδέσμευση Πτυχιακής από Φοιτητή.....	89
4.5.13 Ενημέρωση μέσου όρου Φοιτητών.....	92
Κεφάλαιο 5.....	96
5.1 Αποτελέσματα – Συμπεράσματα.....	96
5.2 Επεκτάσεις.....	96
Βιβλιογραφία .....	97
Παράρτημα.....	98
Διαφάνειες Παρουσίασης.....	98

## Πίνακας εικόνων

Εικόνα 1. Ιστορική αναδρομή web resources.....	14
Εικόνα 2. Στοιβα Σημασιολογικού Ιστού .....	15
Εικόνα 3. Στατική ιστοσελίδα.....	21
Εικόνα 4. Δυναμική ιστοσελίδα.....	22
Εικόνα 5. Στάδια επεξεργασίας μιας PHP σελίδας.....	30
Εικόνα 6. Σύστημα DBMS.....	34
Εικόνα 7. Βάση δεδομένων και Connectors.....	36
Εικόνα 8. Ιεραρχική δόμηση εγγράφου XML.....	40
Εικόνα 9. Αρχική σελίδα www.apachefriends.org.....	54
Εικόνα 10. Σελίδα επιλογής λειτουργικών συστημάτων για XAMPP.....	54
Εικόνα 11. Εκδόσεις προγραμμάτων XAMPP.....	55
Εικόνα 12. Επιλογή Installer δηλαδή το αρχείο .exe.....	56
Εικόνα 13. Παράθυρο λήψης αρχείου εγκατάστασης XAMPP.....	56
Εικόνα 14. Λήψη αρχείου εγκατάστασης XAMPP.....	57
Εικόνα 15. Εκκίνηση εγκατάστασης XAMPP.....	57
Εικόνα 16. Επιλογή φακέλου εγκατάστασης στο δίσκο.....	58
Εικόνα 17. Επιλογές XAMPP.....	58
Εικόνα 18. Εκκίνηση εγκατάστασης.....	59
Εικόνα 19. Ολοκλήρωση εγκατάστασης.....	59
Εικόνα 20. XAMPP Control Panel Application.....	60
Εικόνα 21. Αρχεία και φάκελοι XAMPP.....	60
Εικόνα 22. Αρχεία και φάκελοι στο htdocs και δημιουργία νέου φακέλου.....	61
Εικόνα 23. Δημιουργία και ονομασία νέου φακέλου στο htdocs.....	61
Εικόνα 24. Αρχική Σελίδα.....	64
Εικόνα 25. Καταχώρηση Στοιχείων Φοιτητή.....	65
Εικόνα 26. Καταχώρηση Στοιχείων Καθηγητών.....	69
Εικόνα 27. Καταχώρηση Στοιχείων Πτυχιακών Εργασιών.....	72
Εικόνα 28. Εμφάνιση στοιχείων Φοιτητών.....	75
Εικόνα 29. Εμφάνιση στοιχείων Καθηγητών.....	77
Εικόνα 30. Εμφάνιση όλων των Πτυχιακών.....	78
Εικόνα 31. Εμφάνιση Ελεύθερων Πτυχιακών.....	80
Εικόνα 32. Εμφάνιση Φοιτητών που έχουν κατοχυρώσει πτυχιακή .....	81
Εικόνα 33. Ανάθεση Πτυχιακής σε Φοιτητή.....	83
Εικόνα 34. Ενημέρωση Βαθμού Πτυχιακής.....	86
Εικόνα 35. Αποδέσμευση Πτυχιακής από Φοιτητή.....	89
Εικόνα 36. Ενημέρωση μέσου όρου Φοιτητών.....	92

## Κεφάλαιο 1

### Εισαγωγή

Η πτυχιακή εργασία, όσον αφορά τον εκπαιδευτικό της χαρακτήρα, είναι ένα σημαντικό κομμάτι των σπουδών. Προσομοιάζει το πραγματικό εργασιακό περιβάλλον στο οποίο θα δραστηριοποιηθούν οι απόφοιτοι και περιλαμβάνει την μεγαλύτερη συμμετοχή του καθηγητή από οποιοδήποτε άλλο μάθημα της σχολής. Ο σπουδαστής από την διαδικασία αυτήν γίνεται γνώστης διαφόρων αντικειμένων τα όποια θα είναι χρήσιμα για την περαιτέρω επαγγελματική του πορεία.

### 1.1 Περίληψη

Σκοπός της πτυχιακής αυτής ήταν η μελέτη, σχεδίαση και ανάπτυξη ενός συστήματος για τις ανάγκες της διαχείρισης των πτυχιακών εργασιών ενός ιδρύματος . Η εφαρμογή αυτή είναι μια δυναμική ιστοσελίδα η όποια συνδέεται με μια βάση δεδομένων στην όποια αποθηκεύονται δεδομένα και εξάγονται πληροφορίες με εύκολο και κατανοητό τρόπο ως προς τις επιλογές του χρηστή.

Οι διαδικασίες και η εκτέλεση της εφαρμογής γίνονται σε έναν web server για τον όποιο χρησιμοποιήθηκε το λογισμικό ανοικτού κώδικα Apache web server .Για την ανάπτυξη της εφαρμογής χρησιμοποιήθηκε η HTML αλλά κυρίως η PHP και XML και με την συνεργασία τους έφερε το αποτέλεσμα.

### 1.2 Κίνητρο για την Διεξαγωγή της Εργασίας

Η παρούσα πτυχιακή εργασία με την κατασκευή του συστήματος της Διαχείρισης των πτυχιακών εργασιών στοχεύει να κάνει κάποιες βασικές διαδικασίες αποτελεσματικότερα και πιο γρήγορα ως προς την καταχώρηση ,διαχείριση και οργάνωση των πτυχιακών εργασιών.

### 1.3 Σκοπός της Εργασίας

Σκοπός της παρούσας πτυχιακής είναι η δημιουργία μιας αξιόπιστης εφαρμογής η όποια με αποτελεσματικότητα , ευκολία και ταχύτητα να αποδίδει τις σημαντικές διεργασίες, που είναι οι εξής :

- Εμφάνιση όλων των Πτυχιακών
- Καταχώρηση Πτυχιακών
- Εμφάνιση Ελεύθερων Πτυχιακών
- Εμφάνιση Φοιτητών που έχουν κατοχυρώσει Θέμα Πτυχιακής
- Ανάθεση Πτυχιακής σε Φοιτητή
- Ενημέρωση Βαθμού Πτυχιακής
- Αποδέσμευση Πτυχιακής από Φοιτητή
- Εμφάνιση Στοιχείων Φοιτητών
- Καταχώρηση Στοιχείων Φοιτητών
- Ενημέρωση Μέσου Όρου Φοιτητών
- Εμφάνιση Στοιχείων Καθηγητών
- Καταχώρηση Στοιχείων Καθηγητών



## 1.4 Δομή Εργασίας

Η δομή της εργασίας περιλαμβάνει τα παρακάτω κεφάλαια:

Στο πρώτο Κεφάλαιο , γίνεται μια εισαγωγική περιγραφή για το σκοπό και το κίνητρο της πτυχιακής εργασίας.

Στο δεύτερο Κεφάλαιο , περιγράφονται η μέθοδος ανάλυσης και ανάπτυξης πτυχιακής καθώς και οι θεωρίες που χρειάζονται.

Στο τρίτο Κεφάλαιο , γίνεται αναφορά στις τεχνολογίες και εργαλεία ανάπτυξης που χρησιμοποιήθηκαν στην εργασία και πιο αναλυτική περιγραφή στην γλώσσα XML το οποίο είναι το σημαντικότερο κομμάτι

Στο τέταρτο Κεφάλαιο , διαδραματίζεται το κύριο μέρος της εργασίας μας όπου αναλύουμε τις κινήσεις που έγιναν για να δημιουργηθεί το σύστημα διεξοδικά μαζί με φωτογραφίες από το interface του συστήματος και επεξήγηση κώδικα όπου χρειάζεται.

Στο πέμπτο Κεφάλαιο , εξάγονται τα αποτελέσματα και τα συμπεράσματα από την υλοποίηση της πτυχιακής αυτής καθώς και οι επεκτάσεις της.

## Κεφάλαιο 2

### 2.1 Μέθοδος Ανάλυσης και Ανάπτυξης Πτυχιακής

Η ανάπτυξη Πληροφοριακού Συστήματος για γραμματεία ενός εκπαιδευτικού ιδρύματος είναι μια web εφαρμογή η οποία αντλεί, αποθηκεύει και επεξεργάζεται πληροφορίες σε μία βάση δεδομένων. Η εκτέλεση της εφαρμογής γίνεται σε ένα web server. Έτσι χρειαζόμαστε ανάπτυξη βάσης δεδομένων και παράλληλα ανάπτυξη λογισμικού για την άντληση πληροφοριών από τη βάση μας καθώς και την αποθήκευση τους σε αυτή.

### 2.2 Θεωρίες

#### 2.2.1 Παγκόσμιος Ιστός

Παγκόσμιος ιστός και Internet συχνά θεωρούνται το ίδιο πράγμα. Η αντίληψη αυτή είναι λανθασμένη καθώς ο ιστός αποτελεί μία μόνο εφαρμογή του Internet. Για την ακρίβεια, την δημοφιλέστερη. Σε αντίθεση με το Internet, που έχει και υλική υπόσταση, ο ιστός δεν έχει, μιας και αποτελείται από πακέτα πληροφορίας. Η τεχνολογία του ιστού καθιστά δυνατή την δημιουργία "υπερκειμένων", μία διασύνδεση δηλαδή πάρα πολλών μη ιεραρχημένων στοιχείων που παλαιότερα ήταν απομονωμένα. Τα στοιχεία αυτά μπορούν να πάρουν και άλλες μορφές πέραν της μορφής του γραπτού κειμένου, όπως εικόνας και ήχου.

Η τεχνολογία του ιστού δημιουργήθηκε το 1989 από τον Βρετανό Τιμ Μπέρνερς Λι, που εκείνη την εποχή εργαζόταν στον Ευρωπαϊκό Οργανισμό Πυρηνικών Ερευνών (CERN) στην Γενεύη της Ελβετίας. Το όνομα που έδωσε στην εφεύρεσή του ο ίδιος ο Lee είναι World Wide Web, όρος γνωστός στους περισσότερους από το "www". Αυτό που οδήγησε τον Lee στην εφεύρεση του Παγκόσμιου ιστού ήταν το όραμά του για ένα κόσμο όπου ο καθένας θα μπορούσε να ανταλλάσσει πληροφορίες και ιδέες άμεσα προσβάσιμες από τους υπολοίπους. Το σημείο στο οποίο έδωσε ιδιαίτερο βάρος ήταν η μη ιεράρχηση των διασυνδεδεμένων στοιχείων. Οραματίστηκε κάθε στοιχείο, κάθε κόμβο του ιστού ίσο ως προς την προσβασιμότητα με τα υπόλοιπα. Αν σκεφτεί, όμως, κανείς τον βαθμό ιεράρχησης με τον οποίο λειτουργούν οι μηχανές αναζήτησης του ιστού, όπως για παράδειγμα το google, γίνεται εύκολα κατανοητό ότι στην πράξη κάτι τέτοιο δεν συμβαίνει, τουλάχιστον στον βαθμό που το είχε οραματιστεί ο Lee.

#### Μοντέλο Παγκοσμίου Ιστού

Ως πληροφοριακό σύστημα παρέχει συγκεκριμένο μοντέλο δεδομένων το οποίο βασίζεται σε κόμβους και υπερσυνδέσμους. Το μοντέλο δεδομένων του παγκόσμιου ιστού παρέχει κόμβους (αγγλ. nodes), άγκυρες (αγγλ. anchors) και συνδέσμους (ή υπερσυνδέσμους) (αγγλ. links ή hyperlinks). Οι κόμβοι είναι φορείς του περιεχομένου ενώ οι άγκυρες είναι περιοχές του περιεχομένου των κόμβων από όπου ξεκινούν ή καταλήγουν σύνδεσμοι. Οι σύνδεσμοι συνδέουν δύο κόμβους.

#### 2.2.2 Διαδίκτυο

Το Διαδίκτυο (*Internet*), είναι το μέρος του παγκόσμιου ιστού στο οποίο εφαρμόζονται τα πρωτόκολλα TCP/IP. Επιπλέον, είναι ένα μέσο μαζικής επικοινωνίας. Ως μέσο έχει διπλή υπόσταση: η υλική (που αποτελείται από τον συνδυασμό δικτύων βασισμένων σε λογισμικό και υλικό), και η άυλη (αυτό, δηλαδή, που προσφέρει στην κοινωνία το Διαδίκτυο ως μέσο).

#### Οι πληροφορίες στο Διαδίκτυο

Το Ίντερνετ, σε συνδυασμό με την ολοένα αναπτυσσόμενη ψηφιακή τεχνολογία, έχει δημιουργήσει μία τεράστια αγορά γνώσεων/πληροφοριών. Παραδοσιακές μορφές τέχνης (όπως για

παράδειγμα ο κινηματογράφος και η μουσική) μέσω της ψηφιακής τεχνολογίας παίρνουν την ίδια μορφή (αρχείων δεδομένων) με αντικείμενα που εκ πρώτης όψεως είναι εντελώς διαφορετικά (όπως για παράδειγμα η ιατρική επιστήμη ή κάποιο πρόγραμμα λογισμικού). Παρατηρείται λοιπόν μία συγκέντρωση γνώσης ή, αν είναι δυνατό να λεχτεί, πολιτιστικής κληρονομιάς, που σχετίζεται άμεσα με το Ίντερνετ. Το μεγάλο ερώτημα που προκύπτει πλέον είναι το "ποιος θα διοικήσει, ποιος θα ελέγξει την γνώση αυτή".

Από τη στιγμή που το Διαδίκτυο είναι ένα δίκτυο συνδεδεμένων υπολογιστών, κάθε χρήστης έχει την δυνατότητα να μοιραστεί πληροφορίες με άλλους χρήστες γενόμενος, πολλές φορές, ο ίδιος δημιουργός και πάροχος των πληροφοριών αυτών. Δεν υπάρχει άμεσος έλεγχος των πληροφοριών που "ανεβαίνουν" στο Διαδίκτυο από κάποιον ιεραρχικά ανώτερο χρήστη ή οργανισμό. Το θέμα της μη ιεραρχημένης πληροφορίας, όμως, τίθεται υπό αμφισβήτηση. Ο όγκος της πληροφορίας στο Διαδίκτυο είναι πράγματι μεγάλος. Παρ' όλα αυτά, υπάρχουν πληροφορίες ευκολότερα και δυσκολότερα προσβάσιμες από τον χρήστη.

Το Ίντερνετ έκανε δυνατή την συγκέντρωση μεγάλου όγκου πληροφοριών και επηρέασε σημαντικά τον τρόπο διάθεσής τους. Δεν συμβαίνει, όμως, στον ίδιο βαθμό το ίδιο και στον τρόπο παραγωγής αυτών. Για παράδειγμα, ο τρόπος παραγωγής μιας κινηματογραφικής ταινίας δεν έχει επηρεαστεί σημαντικά από την ύπαρξη του Ίντερνετ, ανεξάρτητα από το αν έχει επηρεαστεί ή όχι από την ψηφιακή τεχνολογία. Παρ' όλα αυτά, και σύμφωνα με την ιντερνετοφιλική προσέγγιση, το Διαδίκτυο ασκεί μεγάλη επίδραση στην διαδικασία παραγωγής δημοσιογραφικών προϊόντων. Η δημιουργία της είδησης παύει να είναι πλέον μονοπώλιο λίγων, αφού ο κάθε χρήστης μπορεί εάν το επιθυμεί να δημιουργήσει πληροφορία ανά πάσα στιγμή. Το πιο τρανταχτό παράδειγμα της επίδρασης αυτής είναι τα ιστολόγια (blogs), όπου μπορεί κανείς να εκφέρει απόψεις και να σχολιάσει γεγονότα πάσης φύσεως (βλ. δημοσιογραφία στον ιστό και δημοσιογραφία των πολιτών). Ως αποτέλεσμα της επιρροής αυτής του Ίντερνετ στη παραγωγή ειδήσεων τα όρια μεταξύ ενός απλού χρήστη του διαδικτύου και ενός επαγγελματία δημοσιογράφου γίνονται περισσότερο δυσδιάκριτα. Αυτό με τη σειρά του οδηγεί στην ανάγκη για επαναπροσδιορισμό της έννοιας της δημοσιογραφίας καθώς και της απαραίτητης εκπαίδευσης των δημοσιογράφων. Η ανάγκη για τον επαναπροσδιορισμό της δημοσιογραφίας, όμως, δεν είναι τόσο μεγάλη σύμφωνα με τους υποστηρικτές της "αντιπλουραλιστικής" προσέγγισης, καθώς θεωρούν πως το Ίντερνετ δεν μπορεί να ασκήσει ουσιαστική επίδραση στην επικοινωνία γενικότερα και στην δημοσιογραφία ειδικότερα.

Επίσης, λόγω της μεγάλης συγκέντρωσης γνώσης στο Διαδίκτυο, η έννοια της κοινωνικής ισότητας παίρνει και πάλι μεγάλη σημασία. Το χάσμα ανάμεσα σε πληροφοριακά πλούσιους και πληροφοριακά φτωχούς θα διευρύνεται όσο αυξάνεται η συγκέντρωση της γνώσης αυτής. Το παραπάνω αποτελεί ακόμα έναν λόγο που κάνει πιο επιτακτική την ανάγκη για διερεύνηση του αρχικού ερωτήματος "ποιος θα ελέγξει τη γνώση αυτή".

Η γλώσσα που χρησιμοποιείται περισσότερο στη διακίνηση της πληροφορίας στο Διαδίκτυο είναι η Αγγλική. Έχοντας αναπτυχθεί τα τελευταία χρόνια, το Διαδίκτυο περιλαμβάνει πλέον ποιοτικά και ποσοτικά ευρύ περιεχόμενο και στις υπόλοιπες γλώσσες των περισσότερο αναπτυγμένων χωρών.

### Πρόσβαση στο Διαδίκτυο

Κοινές μέθοδοι πρόσβασης στο Διαδίκτυο είναι η επιλογική και η ευρυζωνική. Δημόσιοι χώροι για χρήση του Διαδικτύου περιλαμβάνουν τις βιβλιοθήκες και τα Internet cafes, όπου υπάρχουν διαθέσιμοι Η/Υ με σύνδεση στο Διαδίκτυο. Υπάρχουν, επίσης, σημεία πρόσβασης στο Διαδίκτυο σε δημόσιους χώρους όπως αίθουσες αναμονής αεροδρομίων, μερικές φορές μόνο για σύντομη χρήση ενόσω βρισκόμαστε σε αναμονή. Τέτοια σημεία είναι γνωστά και με διάφορους άλλους όρους, όπως «δημόσια περίπτερα Διαδικτύου», «δημόσια τερματικά Διαδικτύου» και «ιστο - τηλέφωνα».

Η δικτύωση μέσω Wi-Fi παρέχει ασύρματη πρόσβαση στο Διαδίκτυο. Ασύρματα σημεία πρόσβασης (hotspot) που παρέχουν τέτοια πρόσβαση περιλαμβάνουν τα Wifi-cafes, όπου κάποιος αρκεί να φέρει τις δικές του/της ασύρματες συσκευές όπως φορητό Η/Υ ή PDA. Οι υπηρεσίες αυτές μπορεί να είναι δωρεάν σε όλους, είτε δωρεάν μόνο σε πελάτες, είτε επί πληρωμή. Ένα hotspot δεν χρειάζεται να περιορίζεται σε ένα συγκεκριμένο περιβάλλον. Ολόκληρες πανεπιστημιούπολεις και πάρκα έχουν αυτή τη δυνατότητα, ακόμα και ολόκληρες περιοχές. Προσπάθειες να συνδεθεί και ο αγροτικός πληθυσμός έχουν οδηγήσει στα ασύρματα κοινοτικά δίκτυα.

Τα πλεονεκτήματα της πρόσβασης ενός χρήστη μέσω του δικού του υπολογιστή (αντί μέσω δημόσιου τερματικού) περιλαμβάνουν τη δυνατότητα για κατέβασμα και ανέβασμα αρχείων χωρίς περιορισμούς, τη χρήση του αγαπημένου του φυλλομετρητή (web browser) και των ρυθμίσεων αυτού (το μενού των ρυθμίσεων μπορεί να απενεργοποιηθεί σε έναν δημόσιο υπολογιστή) και την εκτέλεση δραστηριοτήτων στο Ίντερνετ με τη χρήση δικών του προγραμμάτων και δεδομένων.

### 2.2.3 Web 2.0

Ο όρος Web 2.0 (Ιστός 2.0), χρησιμοποιείται για να περιγράψει τη νέα γενιά του Παγκόσμιου Ιστού η οποία βασίζεται στην όλο και μεγαλύτερη δυνατότητα των χρηστών του Διαδικτύου να μοιράζονται πληροφορίες και να συνεργάζονται online. Αυτή η νέα γενιά είναι μια δυναμική διαδικτυακή πλατφόρμα στην οποία μπορούν να αλληλεπιδρούν χρήστες χωρίς εξειδικευμένες γνώσεις σε θέματα υπολογιστών και δικτύων.

#### Ιστορία

Η φράση Web 2.0 ειπώθηκε για πρώτη φορά το 2004 κατά τη διάρκεια ενός συνεδρίου μεταξύ της O'Reilly Media και της MediaLive International όπου προτείνονταν ιδέες για την αναβάθμιση του παγκόσμιου ιστού. Ο Dale Dougherty και O'Reilly VP, παρατήρησε ότι το διαδίκτυο είχε αρχίσει να γίνεται πολύ δημοφιλές και σημαντικό μέρος της καθημερινότητας όλο και μεγαλύτερου ποσοστού ανθρώπων. Συνεχώς έβγαιναν νέες εφαρμογές και ιστοσελίδες οι οποίες αναγνωρίζονταν από το ευρύ κοινό σε σύντομο χρονικό διάστημα. Επιπλέον οι περισσότερες εταιρείες άρχισαν να στρέφουν την επιχειρηματική τους δράση στο διαδίκτυο και να προσπαθούν να φέρουν τους καταναλωτές τους προς αυτό το κανάλι.

Έχοντας όλα αυτά ως δεδομένα εξέτασαν το Web 2.0 ως μια δεύτερη γενιά υπηρεσιών βασισμένων στο Διαδίκτυο. Χρησιμοποίησαν αυτή την φράση σαν τίτλο για μια σειρά από συνέδρια με τον τίτλο *Web 2.0*. Τα συνέδρια αυτά συνεχίζονται και μέχρι σήμερα για να αποσαφηνιστεί ο όρος και να προωθηθεί η ιδέα σε όλα τα ενδιαφερόμενα μέρη.

#### Χαρακτηριστικά

Όπως έχει αναφερθεί το Web 2.0 ξεπερνά τα όρια της περιορισμένης πλατφόρμας ενός υπολογιστή. Ο χρήστης θα μπορεί να δρα στον Παγκόσμιο Ιστό όπως δρούσε μέχρι τώρα στον υπολογιστή του. Οι πιο ειδικοί μιλούν για έναν νέο τρόπο σχεδίασης των ιστοσελίδων ο οποίος θα βασίζεται στην διάδραση του χρήστη. Θα επιτρέπει στον χρήστη να αλλάξει τόσο το περιβάλλον της σελίδας όσο και να παρέμβει στο περιεχόμενό της. Χαρακτηριστικές εφαρμογές του Web 2.0 είναι τα κοινωνικά μέσα (social media), τα wiki και τα blog. Πολλές από τις εντολές διάδρασης που χαρακτηρίζουν την λειτουργία του Web 2.0 μας είναι ήδη γνωστές από διάφορες ιστοσελίδες social media όπως το facebook ή το youtube για παράδειγμα. Τέτοιες εκφράσεις είναι η **αναζήτηση** (search), το **tag**, η παράθεση **links** ή το **authoring** όπως λειτουργεί σε πολλά wiki όπου οι χρήστες μπορούν να δημιουργήσουν άρθρα αλλά και να ανανεώσουν ή να διαγράψουν ήδη υπάρχοντα.

#### Κριτική

Το ακριβές νόημά του όρου παραμένει ανοιχτό προς αντιπαράθεση, και μερικοί ειδικοί, συμπεριλαμβανομένου και του Τιμ Μπέρνερς Λι, έχουν αμφισβητήσει εάν ο όρος έχει κάποιο πραγματικό νόημα.

Μεταξύ άλλων έχει κατηγορηθεί ότι αποτελεί εφεύρεση του μάρκετινγκ. Αναφέρεται σαν επιχείρημα πως οι τεχνολογίες που είναι σε χρήση στο Ίντερνετ αναβαθμίζονται συνεχώς και η προηγούμενη μέρα πριν το Web 2.0 δεν απείχε πολύ. Επίσης με την ίδια λογική, μερικούς μήνες μετά την χρήση ενός υποτιθέμενου Web 2.0, το Web 2.1 θα εμφανιζόταν. Είναι όμως αδύνατο να εκφραστεί η τεχνολογική ιδιότητα του ίντερνετ ακριβώς με έναν αριθμό μιας και το αμάλγαμα τεχνολογιών που είναι αυτή τη στιγμή σε χρήση στο ίντερνετ είναι αχανώς πολυσύνθετο. Επιπροσθέτως, λειτουργίες

όπως επικοινωνία με τον χρήστη υπήρχαν πριν πολλά χρόνια, όπως η χρήση σελίδων χρήστη (home pages), τα chat (IRC) και πολλά άλλα. Αυτό που μπορεί να ειπωθεί ότι άλλαξε είναι η ευκολία χρήσης.

#### 2.2.4 Web 3.0 Σημασιολογικός Ιστός

Με το Web 2.0 το οποίο μπαίνουμε στη φάση διάβασμα-γράψιμο, σύμφωνα πάλι με τον ορισμό που δίνει ο Tim Berners-Lee (read-write web). Ο όρος Web 2.0, χρησιμοποιήθηκε το 2004 για επαγγελματικούς λόγους αλλά στη συνέχεια επικράτησε σαν αναφορά στη γενιά 2 του Web. Η επικοινωνία είναι πλέον αμφίδρομη, αφού υπάρχει η δυνατότητα ανάρτησης σχολίων και συνδιαλλαγής δημόσια. Το περιεχόμενο της πληροφορίας ανεβαίνει στα κοινωνικά μέσα (social media) και στη συνέχεια γίνεται κτήμα όλων και προωθείται μέσα από φίλους, γνωστούς, συναδέλφους ή παντελώς άγνωστους ανθρώπους με κοινά ενδιαφέροντα.

Τα blogs, τα wikis, η κοινωνική σελιδοσήμανση (social bookmarking), ο διαμοιρασμός φωτογραφιών (flickr.com) και βίντεο (youtube.com) και τα περιβάλλοντα εικονικής πραγματικότητας όπως το Second Life (SL), έχουν διασυνδέσει τεράστιο αριθμό ανθρώπων σε δίκτυα.

Ενώ το Web 2.0 δεν έχει ακόμη ωριμάσει και γίνει κτήμα πολλών ανθρώπων, κάποιος –ο Tim Berners Lee και οι συνεργάτες του συγκεκριμένα– οραματίζονται ήδη το Web 3.0. Με το Web 3.0, που βρίσκεται ακόμη σε πειραματικό στάδιο παρόλο που υπάρχουν μεμονωμένες εφαρμογές, οι μηχανές θα γίνουν εξυπνότερες. Η εξειδικευμένη αναζήτηση θα μπορεί να φέρει αποτελέσματα στην ερώτηση ‘φέρε μου οτιδήποτε πάνω στο X θέμα, λαμβάνοντας υπόψη το Ψ στοιχείο’. Η μηχανή θα μπορεί να καταλάβει ποια είμαι, τι κάνω και ποιο θέλω να είναι το επόμενο βήμα.

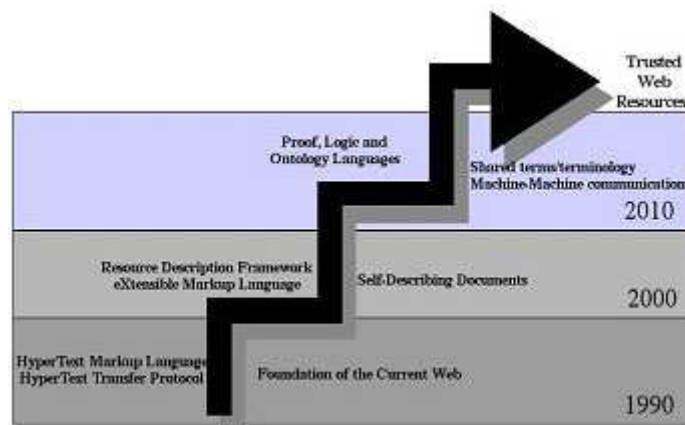
Η κοινωνική αναζήτηση (social search) θα φέρνει αποτελέσματα διαφορετικά από τις σημερινές σελίδες με χαρακτηριστικό στοιχείο την εξατομίκευση της πληροφορίας. Θα μπορούμε να έχουμε μια ψηφιακή προσωπική συλλογή με ότι έχουμε συγκεντρώσει (φωτογραφίες, βίντεο, ιστοσελίδες, blogs κλπ.) και κάθε συναλλαγή μας με το σύστημα θα έχει τη θέση της στον δικό μας ψηφιακό χώρο. Οι φίλοι μας από το Facebook θα μπορούν να συναντήσουν τους άλλους μας φίλους στο Twitter, αφού η διαλειτουργικότητα θα κάνει τα συστήματα και τις εφαρμογές να επικοινωνούν.

Η δεύτερη γενιά του Web δεν διαφοροποιείται ουσιαστικά από την πρώτη. Το νέο στοιχείο είναι η αλλαγή στον τρόπο με τον οποίο οι άνθρωποι αντιμετωπίζουν το μέσο, αφού σιγά-σιγά μεγαλώνει και γενιά που είναι εξοικειωμένη τεχνολογικά με το διαδίκτυο. Το συμμετοχικό Web είναι πραγματικότητα και έχει φέρει μεγάλες αλλαγές στη ζωή μας σε πολλούς τομείς. Το Web 3.0, όταν μπορέσει να εφαρμοστεί με τη μορφή που οραματίζονται οι εμπνευστές του θα είναι άλλη μια τεχνολογική επανάσταση του ανθρώπου γιατί θα αλλάξει ριζικά τον τρόπο που θα επικοινωνούμε, θα εργαζόμαστε και θα αλληλεπιδρούμε γενικότερα. Μέχρι τότε ας μοιραστούμε αρχεία με φίλους, συνεργάτες και άλλους ανθρώπους και ας απολαύσουμε αυτόν τον νέο τρόπο επικοινωνίας.

#### Σημασιολογικός Ιστός (Semantic Web)

Τα τελευταία χρόνια γίνεται πολύς λόγος και για το Semantic Web, ενός διαδικτύου δηλαδή στο οποίο η πληροφορία είναι καλύτερα ορισμένη επιτρέποντας με αυτόν τον τρόπο την ολοκλήρωση, αυτοματοποίηση και επαναχρησιμοποίηση των δεδομένων.





**Εικόνα 1. Ιστορική αναδρομή web resources**

Ο Ιστός (Web) του μέλλοντος προβλέπεται να αποτελεί μια παγκόσμια βάση δεδομένων και γνώσης με πληροφορίες οι οποίες θα είναι "κατανοητές" από μηχανές (machine-understandable information). Οι κύριες τεχνολογίες για την υλοποίηση του Σημασιολογικού Ιστού είναι η σημασιολογικό εμπλουτισμός και η χρήση των οντολογιών.

Η λέξη "Σημασιολογία" έχει ρίζα τις Ελληνικές λέξεις "σημάδι", "σημαίνω" και "σημαντικός" και σήμερα αναφέρεται στο νόημα συχνά σε επίπεδο γλώσσας. Μπορούμε να πούμε ότι ο Σημασιολογικός Ιστός αποτελεί το μεγαλύτερο σε παγκόσμιο επίπεδο έργο ευφυής ενσωμάτωσης συστημάτων ώστε να συνεργάζονται δια-λειτουργικά.

Ο Tim Berners-Lee, που επινόησε τον Παγκόσμιο Ιστό το 1989, είχε το όραμα, που τώρα συμερίζονται πολλοί - ενός ιστού δεδομένων που μπορούν να επεξεργαστούν από μηχανές. "Ο Σημασιολογικός Ιστός είναι μια επέκταση του σημερινού ιστού όπου η πληροφορία έχει καλά καθορισμένο νόημα, καθιστώντας τη συνεργασία μεταξύ ανθρώπων και υπολογιστών πιο αποτελεσματική", Tim Berners-Lee, James Hendler, Ora Lassila, The Semantic Web, Scientific American, Μάιος 2001.

Το κέντρο βάρους του περιεχομένου του Ιστού μετατοπίζεται συνεχώς από τον άνθρωπο προς τα δεδομένα. Για να φτάσει ο Ιστός το μέγιστο των δυνατοτήτων του, πρέπει να εξελιχθεί σε ένα Σημασιολογικό Ιστό, ο οποίος παρέχει μια διεθνώς προσβάσιμη πλατφόρμα που επιτρέπει σε αυτοματοποιημένα εργαλεία αλλά και σε ανθρώπους να μοιράζονται και να επεξεργάζονται δεδομένα.

Ο Σημασιολογικός Ιστός αποτελεί πρωτοβουλία της Κοινοπραξίας του Παγκοσμίου Ιστού (W3C) και η σχετική Δραστηριότητα (W3C Semantic Web Activity) έχει δημιουργηθεί για να εξυπηρετήσει έναν ηγετικό ρόλο, τόσο στο σχεδιασμό προδιαγραφών, όσο και στην ανοικτή ανάπτυξη της τεχνολογίας μέσω της συνεργασίας.

### **Μεταδεδομένα και Σημασιολογικός Χαρακτηρισμός**

Εκτός από τα μετα- δεδομένα τα οποία προκύπτουν από τις προτιμήσεις των χρηστών ή την δομή ενός περιβάλλοντος hypertext, υπάρχουν και άλλα τα οποία σχετίζονται με αντικείμενα, ιδέες, διαδικασίες καθώς και των σχέσεων που τις διέπουν. Αυτά τα δεδομένα έχουν σημασιολογικό νόημα το οποίο με την σειρά του είναι δυνατόν να περιγραφεί από φορμαλισμούς ή λεξιλόγια που καλούνται οντολογίες. Ο σημασιολογικός χαρακτηρισμός της πληροφορίας και η χρήση οντολογιών για την αναπαράσταση της είναι σχετικά ένας καινούργιος τομέας στην περιοχή της διαχείρισης γνώσης.

Η διαδικασία της μετατροπής της πληροφορίας σε σημασιολογική είναι δυνατή μέσω ενός πλαισίου περιγραφής, του RDF (Resource Description Framework) και των RDF schemas. το World Wide Web Consortium (W3C) έχει δημιουργήσει μία ομάδα για την δημιουργία της γλώσσας OWL η οποία θα περιγράφει τις οντολογίες που υπάρχουν στο διαδύκτιο, μία φιλόδοξη και πολύ δύσκολη προσπάθεια.

Για την περιγραφή, διαχείριση και εισαγωγή μεταδεδομένων σε κοινά έγγραφα του διαδικτύου έχουν κάνει την εμφάνισή τους τα τελευταία χρόνια αρκετά πρότυπα όπως περιγραφές τύπων κειμένου (Document Type Definitions, XML-DTD), το νέο πρότυπο περιγραφής πολυμέσων

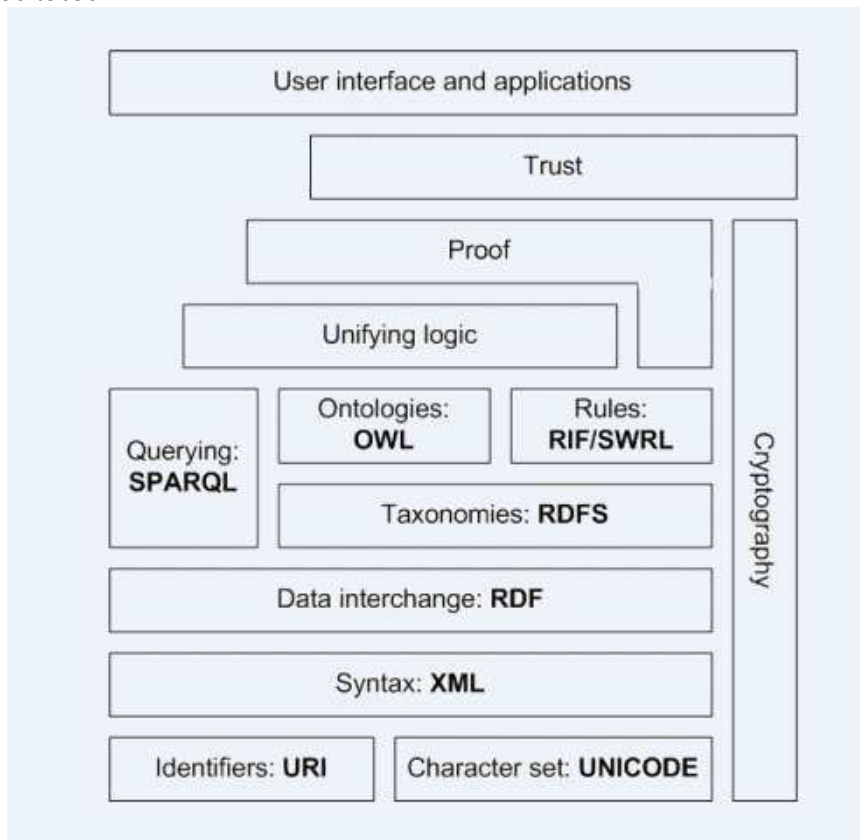
MPEG-7 και γενικά η χρήση της γλώσσας περιγραφής XML (Extensible Markup Language), Topic Maps (σε μορφή XTM) και Semantic nets.

Αρκετά δημοφιλής είναι η τεχνολογία των γλωσσών XML/XSL που είναι κατάλληλες για δομημένες περιγραφές. Με βάση την τεχνολογία αυτή είναι δυνατόν να δημιουργηθούν μεταδεδομένα που περιγραφεί ικανοποιητικά ψηφιακό υλικό (εικόνες, videos, 3D γραφικά κτλ). Από την άλλη πλευρά οι πληροφορίες που αφορούν εμπορικές συναλλαγές μπορούν να περιγραφούν μέσω του προτύπου XML/EDI (ebXML).

### Τα συστατικά του Σημασιολογικού Ιστού

Ο Σημασιολογικός Ιστός στηρίζεται από τις ακόλουθες γλώσσες και πρότυπα:

- **XML (Extensible Markup Language):** Είναι μια γλώσσα περιγραφής δεδομένων τα οποία είναι εύκολο να διαβαστούν και να επεξεργαστούν από ανθρώπους και προγράμματα. Δεν επιβάλλει κανέναν σημασιολογικό περιορισμό στα δεδομένα που περιγράφει.
- **XML Schema:** Είναι μια γλώσσα η οποία περιορίζει τη δομή των XML εγγράφων.
- **RDF:** Είναι ένα μοντέλο περιγραφής και επεξεργασίας μεταδεδομένων.
- **RDF Schema:** Είναι ένας μηχανισμός περιγραφής πόρων και των σχέσεων ανάμεσα τους και αποτελεί σημασιολογική επέκταση του RDF.
- **OWL:** Παρέχει έναν τρόπο περιγραφής όρων και σχέσεων γύρω από ένα πεδίο ενδιαφέροντος, προσφέροντας πιο ισχυρό συνακτικό από τις RDF και RDF Schema καθώς και πιο ισχυρή σημασιολογία που βασίζεται στη λογική (logic-based semantics).
- **SPARQL:** Είναι ένα πρωτόκολλο και γλώσσα ερωτημάτων για τις πηγές δεδομένων του σημασιολογικού ιστού



Εικόνα 2 . Στοιβά Σημασιολογικού Ιστού

## Οντολογίες

Από την άλλη πλευρά, η έρευνα στην περιοχή των οντολογιών βρίσκεται στα πρώτα στάδια της, να και έχουν ήδη εμφανιστεί αρκετές γλώσσες που επιτρέπουν την αναπαράστασή του. Γλώσσες όπως η SHOE, η DAML, η OIL και η υβριδική DAML+OIL[32] αποτελούν επεκτάσεις της RDF δανειζόμενες χαρακτηριστικά από αντίστοιχες γλώσσες αναπαράστασης του πεδίου της τεχνητής νοημοσύνης. Όλες αυτές οι γλώσσες βρίσκονται σε προκαταρκτικό στάδιο.

Από την άλλη πλευρά, η έρευνα στην περιοχή των οντολογιών βρίσκεται στα πρώτα στάδια της, να και έχουν ήδη εμφανιστεί αρκετές γλώσσες που επιτρέπουν την αναπαράστασή του. Γλώσσες όπως η SHOE, η DAML, η OIL και η υβριδική DAML+OIL[32] αποτελούν επεκτάσεις της RDF δανειζόμενες χαρακτηριστικά από αντίστοιχες γλώσσες αναπαράστασης του πεδίου της τεχνητής νοημοσύνης. Όλες αυτές οι γλώσσες βρίσκονται σε προκαταρκτικό στάδιο.

Ένας αριθμός οντολογιών έχει επίσης εμφανιστεί για την διευκόλυνση της ανταλλαγής δεδομένων μεταξύ εμπορικών εφαρμογών τύπου e-Commerce μέσω της αποδοχής κοινής σημασιολογίας για την περιγραφή της πληροφορίας. Αυτές περιλαμβάνουν την CBL (Common Business Library), την cXML (commerce XML), την OCF (Open Catalog Format), την OFX (Open Financial Exchange), το UN/SPSC, το RosetaNet και άλλες. Σημαντικές πρωτοβουλίες έχουν παρθεί για την δημιουργία οντολογιών για επιστημονικά πεδία όπως η Γενετική καθώς και της εφαρμογής τους σε δίκτυα GRID.

Όσον αφορά την αυτόματη παραγωγή μετα-δεδομένων και του αυτόματου σημασιολογικού χαρακτηρισμού της πληροφορίας, έχουν παρουσιαστεί κάποιες πρώτες προσπάθειες όπως τα εργαλεία MnM και S-Cream τα οποία όμως είναι σε πρώιμο στάδιο.

Ο σημασιολογικός χαρακτηρισμός της πολυμεσικής πληροφορίας καθώς και η διαχείριση των μετα-δεδομένων που προκύπτουν από αυτή αποτελεί ένα ανοικτό ερευνητικό πεδίο καθώς υπάρχουν πολλά ανοικτά θέματα τα οποία πρέπει να αντιμετωπιστούν. Μεταξύ αυτών η έλλειψη των κατάλληλων εργαλείων για την εξαγωγή των μεταδεδομένων και η έλλειψη των κατάλληλων οντολογιών για την αναπαράστασή τους.

## Συμπεράσματα

Ήδη κυκλοφορούν κάποια συστήματα υπό τη μορφή πρωτοτύπων, που λειτουργούν καταναεμημένοι χώροι αποθήκευσης (repository) για τον Παγκόσμιο Ιστό (WWW). Τα προϊόντα αυτά, υποστηρίζουν την ομοσπονδιακή (federated) ανάπτυξη και αποθήκευση των μοντέλων αναφοράς και των οντολογιών (ontologies), καθώς και την αντιστοίχιση και προβολή ανάμεσα σε διαφορετικά συστήματα μοντέλα/σχήματα, ενώ υποστηρίζουν τα πρότυπα του W3C (όπως XML και RDF).

Ο σημασιολογικός Ιστός (Semantic Web) αποτελεί μια καινοτομία "εν τη γενέσει" της, η οποία υπόσχεται την ενσωμάτωση όψεων (views) βάσεων δεδομένων, "συναλλαγών" μεταξύ βάσεων δεδομένων (database transactions), λογικών αναπαραστάσεων, συνδέσμων Ιστού (Web links) και αντικειμενοστρεφών αναπαραστάσεων, σε μια σημασιολογική βάση.

Τα παραπάνω χαρακτηριστικά οδηγούν σε ένα νέο τρόπο στην επεξεργασία ερωτήσεων (query processing). Η τεχνολογία που παρουσιάζεται θα είναι υποχρεωτική, κατά τους δημιουργούς της, στις περισσότερες καταναεμημένες εφαρμογές-πελάτη, ενώ δεν υπάρχει περιορισμός ως προς τα πεδία (αγορές) εφαρμογής της.

Το όραμα της δημιουργίας του Semantic Web στηρίζεται στην επέκταση των υπάρχοντων πλαισίων περιγραφής μετα-δεδομένων και ειδικότερα στην ύπαρξη σημασιολογικού περιεχομένου που είναι δυνατόν να υπόκειται σε αυτόματη επεξεργασία από τον υπολογιστή χωρίς την επέμβαση του ανθρώπινου παράγοντα. Η ερευνητική προσπάθεια έγκειται στην δημιουργία γενικών πλαισίων όπως το UN/SPSC και γλωσσών όπως η OWL τα οποία θα υποστηρίζουν οντολογικά όσο το δυνατόν περισσότερα πεδία εφαρμογών γίνεται. Πρόσφατα, η ομάδα εργασίας οντολογίας του Παγκοσμίου Ιστού εξέδωσε ένα υπό διαμόρφωση προσχέδιο τελικού σχολιασμού των παραδειγμάτων της OWL το οποίο συνοδεύει τον ορισμό της γλώσσας.



### 2.2.5 Τι είναι μια Ιστοσελίδα

Ιστοσελίδα (αγγλικά: *web page*) είναι ένα είδος εγγράφου του παγκόσμιου ιστού (WWW) που περιλαμβάνει πληροφορίες με την μορφή κειμένου, υπερκειμένου, εικόνας, βίντεο και ήχου.

Πολλές ιστοσελίδες μαζί συνθέτουν έναν ιστότοπο (εναλλακτικές ονομασίες: *ιστοχώρος* ή *δικτυακός τόπος*, αγγλ. *web site* ή *Internet site*). Οι σελίδες ενός ιστοτόπου εμφανίζονται κάτω από το ίδιο όνομα χώρου (domain) π.χ. microsoft.com. Οι ιστοσελίδες αλληλοσυνδέονται και μπορεί ο χρήστης να μεταβεί από τη μία στην άλλη κάνοντας «κλικ», επιλέγοντας δηλαδή συνδέσμους που υπάρχουν στο κείμενο ή στις φωτογραφίες της ιστοσελίδας. Οι σύνδεσμοι προς άλλες σελίδες εμφανίζονται συνήθως υπογραμμισμένοι και με μπλε χρώμα για να είναι γρήγορα ξεκάθαρο στον επισκέπτη ότι πρόκειται για σύνδεσμο προς άλλη ιστοσελίδα, χωρίς όμως πάντα να είναι αυτό απαραίτητο.

Η κατασκευή ιστοσελίδων είναι κάτι που μπορεί να γίνει πολύ εύκολα με προγράμματα που κυκλοφορούν ελεύθερα, αλλά υπάρχουν και αυτοματοποιημένοι μηχανισμοί κατασκευής ιστοσελίδων που επιτρέπουν σε απλούς χρήστες να δημιουργήσουν εύκολα και γρήγορα προσωπικές ή και εμπορικές ιστοσελίδες. Από την άλλη μεριά υπάρχουν και πολλές εταιρίες, που εξειδικεύονται στη δημιουργία ελκυστικών και λειτουργικών ιστοσελίδων που έχουν σαν στόχο να οδηγήσουν τους επισκέπτες στην αγορά κάποιου προϊόντος, στην επικοινωνία με τον ιδιοκτήτη του ιστοτόπου ή απλά στο ανέβασμα του εταιρικού προφίλ μιας επιχείρησης.

### 2.2.6 Φυλλομετρητής ιστοσελίδων

Ένας **Web browser** (*φυλλομετρητής ιστοσελίδων*, *πλοηγός Web*, *πρόγραμμα περιήγησης Web* ή *περιηγητής Ιστού*) είναι ένα λογισμικό που επιτρέπει στον χρήστη του να προβάλλει, και να αλληλεπιδρά με, κείμενα, εικόνες, βίντεο, μουσική, παιχνίδια και άλλες πληροφορίες συνήθως αναρτημένες σε μια ιστοσελίδα ενός ιστότοπου στον Παγκόσμιο Ιστό ή σε ένα τοπικό δίκτυο. Το κείμενο και οι εικόνες σε μια ιστοσελίδα μπορεί να περιέχουν υπερσυνδέσμους προς άλλες ιστοσελίδες του ίδιου ή διαφορετικού ιστότοπου. Ο Web browser επιτρέπει στον χρήστη την γρήγορη και εύκολη πρόσβαση σε πληροφορίες που βρίσκονται σε διάφορες ιστοσελίδες και ιστότοπους εναλλάσσοντας τις ιστοσελίδες μέσω των υπερσυνδέσμων. Οι φυλλομετρητές χρησιμοποιούν τη γλώσσα μορφοποίησης HTML για την προβολή των ιστοσελίδων, για αυτό η εμφάνιση μιας ιστοσελίδας μπορεί να διαφέρει ανάλογα με τον browser.

Οι πλοηγοί Web ουσιαστικά αποτελούν λογισμικό πελάτη του δικτυακού πρωτοκόλλου επιπέδου εφαρμογών HTTP. Για κάθε browser διατίθενται, επίσης, και αρκετά πρόσθετα στοιχεία («add-ons» ή «plug-in»), με στόχο την επαύξηση των δυνατοτήτων τους, τη βελτίωση της χρηστικότητάς τους και την προστασία του χρήστη σε θέματα ασφάλειας.

Οι περισσότερο χρησιμοποιούμενοι browsers είναι οι:

- Windows Internet Explorer
- Mozilla Firefox
- Apple Safari
- Netscape Navigator
- Opera
- Pandora
- Google Chrome

### 2.2.7 Κατασκευή ιστοσελίδων

Κατασκευή ιστοσελίδων (αγγλικά: *Web development*) είναι η διαδικασία δημιουργίας παρουσιάσεων περιεχομένου (συνήθως υπερκειμένου, ή πολυμέσων), οι οποίες προβάλλονται στον τελικό χρήστη του Διαδικτύου, μέσω ενός προγράμματος περιήγησης (browser) ή άλλων υπηρεσιών όπως διαδικτυακή τηλεόραση, ιστολόγια (blogs) και RSS Feeds.

#### Περιεχόμενο

Μπορούν να χρησιμοποιηθούν διάφορες τεχνικές προβολής του περιεχομένου μιας σελίδας, μεταξύ αυτών: κινούμενα σχέδια, γραφιστική, αλληλεπίδραση ανθρώπου-υπολογιστή, μάρκετινγκ, φωτογραφία, βελτιστοποίηση μηχανών αναζήτησης και τυπογραφία.

#### Τεχνολογίες

Τεχνολογίες που χρησιμοποιούνται στην ανάπτυξη ιστοσελίδων:

- γλώσσες σελίδας: HTML και XHTML
- φύλλα στυλ CSS
- γλώσσες περιγραφής δεδομένων: XML, JSON
- γλώσσες προγραμματισμού σεναρίων: Perl, PHP, Python, Ruby
- πλατφόρμες: ASP και ASP.NET της Microsoft, Java Enterprise της Sun, agile frameworks όπως το Django και το Ruby on Rails
- συγγραφή κώδικα στην πλευρά του πελάτη: Javascript

#### Σχεδιασμός Σελίδων Ιστού (Web Design)

Ο σχεδιασμός στο Διαδίκτυο αφορά την ικανότητα δημιουργίας παρουσιάσεων περιεχομένου (συνήθως κειμένου ή πολυμέσων) οι οποίες φτάνουν στον τελικό-χρήστη μέσω του Παγκόσμιου Ιστού, με τη χρήση λογισμικού όπως ένας φυλλομετρητής (web browser) ή άλλου λογισμικού σχεδιασμένου για το διαδίκτυο όπως η τηλεόραση μέσω διαδικτύου, κινητών τηλεφώνων κλπ.

Η ιστοσελίδα είναι ένα ηλεκτρονικό αρχείο ή ένα σύνολο από ηλεκτρονικά αρχεία που υπάρχουν σε έναν ή και περισσότερους εξυπηρετητές (server/servers) και παρουσιάζει κείμενα και εφαρμογές πολυμέσων στον τελικό-χρήστη. Τέτοια στοιχεία όπως κείμενο, εικόνες (σύμφωνα με τα πρότυπα SVG, BMP, GIF, JPEG ή PNG) και φόρμες μπορούν να τοποθετηθούν στη σελίδα με τη χρήση γλωσσών σήμανσης υπερκειμένου όπως HTML/XHTML/XML. Η αναπαραγωγή πιο σύνθετων πολυμέσων (ανυσματικών γραφικών, βίντεο, ήχων, γραφικών με ενσωματωμένο ήχο και εικόνα) μπορεί να γίνει με πρόσθετα (plug-in) όπως το Flash, το QuickTime, το περιβάλλον χρόνου εκτέλεσης Java, κ.α ή με τεχνολογίες όπως οι γλώσσες σήμανσης (X) HTML5, XML και MathML, και το πρότυπο φύλλων στυλ CSS 3.

Οι καινούριες εκδόσεις των προγραμμάτων περιήγησης (Internet Explorer 7, 8, Firefox 3.6, safari, chromium κλπ) που ακολουθούν τα W3C πρότυπα οδήγησαν σε μια ευρεία αποδοχή και χρήση των XHTML/XML σε συνδυασμό με τα CSS (Cascading Style Sheets) για την τοποθέτηση και διαχείριση των στοιχείων και αντικειμένων της ιστοσελίδας. Τα τελευταία πρότυπα στοχεύουν στο να αποκτήσουν τα προγράμματα περιήγησης την δυνατότητα να προσφέρουν μια ευρεία γκάμα επιλογής πολυμέσων και πρόσβασης στους πελάτες χωρίς τη χρήση των προσθέτων (plug-in).

Γενικότερα οι ιστοσελίδες διαχωρίζονται σε στατικές και δυναμικές:

- Οι στατικές δεν αλλάζουν περιεχόμενο και διάταξη (layout) με οποιοδήποτε αίτημα εκτός και αν ο προγραμματιστής αναβαθμίσει (update) τη σελίδα. Μια απλή HTML σελίδα είναι παράδειγμα στατικού περιεχομένου.
- Οι δυναμικές προσαρμόζουν το περιεχόμενο και/ή την εμφάνισή τους σύμφωνα με την καταχώρηση/αλληλεπίδραση ή τις αλλαγές του τελικού χρήστη στο περιβάλλον προγραμματισμού (χρήστης, ώρα, τροποποιήσεις στη βάση δεδομένων, κτλ). Το περιεχόμενο μπορεί να αλλάζει στον υπολογιστή του τελικού-χρήστη με τη χρήση των γλωσσών προγραμματισμού που εκτελούνται στον υπολογιστή του χρήστη (JavaScript, VBScript, Actionscript, etc.). Το περιεχόμενο στις δυναμικές σελίδες συχνά μεταφράζεται στον εξυπηρετητή (server) μέσω γλωσσών προγραμματισμού που εκτελούνται στον εξυπηρετητή (Perl, PHP, ASP, JSP, ColdFusion, .NET κτλ).

Με την συνεχόμενη ειδίκευση στην τεχνολογίας της Πληροφορικής δημιουργείται η ανάγκη διαχωρισμού του Σχεδιασμού Ιστοσελίδων από τον Προγραμματιστή Ιστοσελίδων.

Για τη διαδικασία σχεδιασμού μιας ιστοσελίδας, μιας εφαρμογής ή ενός πολυμέσου για το διαδίκτυο μπορεί να συνδυάζονται πολλοί κλάδοι όπως animation, συγγραφή, επικοινωνιακός σχεδιασμός, εταιρική ταυτότητα, σχεδιασμός γραφικών, αλληλεπίδραση ανθρώπου-υπολογιστή, αρχιτεκτονική υπολογιστών, σχεδιασμός αλληλεπίδρασης, marketing, φωτογραφία, βελτιστοποίηση μηχανών αναζήτησης και τυπογραφία.

Τα εργαλεία και οι τεχνολογίες που χρησιμοποιούνται περιλαμβάνουν μεταξύ άλλων:

- Γλώσσες σήμανσης: HTML, XHTML, XML
- Γλώσσες φύλλων στυλ: CSS, XML
- Σενάρια στον πελάτη (Client-side scripting): JavaScript
- Σενάρια στον εξυπηρετητή (Server-side scripting): PHP, ASP, Perl, Ruby/Ruby on Rails
- Τεχνολογίες βάσεων δεδομένων: MySQL, PostgreSQL, Microsoft SQL Server
- Τεχνολογίες πολυμέσων: Flash, Silverlight

Οι ιστοσελίδες μπορούν να είναι στατικές ή δυναμικές, που αναπροσαρμόζουν αυτόματα το περιεχόμενό τους ή την εμφάνισή τους βασιζόμενες σε μια ποικιλία παραγόντων, όπως μια καταχώρηση από τον τελικό-χρήστη, μια καταχώρηση ή αλλαγή στο περιβάλλον προγραμματισμού από τον χειριστή της ιστοσελίδας ( όπως εν παραδείγματι μια μορφοποίηση της βάσης δεδομένων).

Εξαιτίας της συνεχούς εξειδίκευσης στους τομείς της επικοινωνιακής Πληροφορικής, δημιουργείται μια σαφής τάση διαχωρισμού του σχεδιασμού στο Διαδίκτυο από την ανάπτυξη του διαδικτύου ως μέσου ροής της πληροφορίας και των αγαθών προς όλες τις διαδικτυακές υπηρεσίες.

### **Προσβασιμότητα Ιστοσελίδων**

Για να είναι προσβάσιμη μια ιστοσελίδα θα πρέπει να ακολουθεί κάποιες αρχές προσβασιμότητας. Αυτές οι αρχές είναι γνωστές ως WCAG όταν μιλάμε για το περιεχόμενο και μπορούν να ομαδοποιηθούν στις ακόλουθες κατηγορίες.

- Χρήση σημασιολογικής σήμανσης (semantic markup), η οποία προσφέρει μια ενιαία δομή στο αρχείο.
- Το semantic markup επίσης αναφέρεται στην οργάνωση της δομής της ιστοσελίδας αλλά και στην ανακοίνωση/έκδοση/δημιουργία ορισμών των διαδικτυακών υπηρεσιών με τρόπο ώστε να είναι αναγνωρίσιμες από άλλες διαδικτυακές υπηρεσίες σε διαφορετικές ιστοσελίδες. Τα πρότυπα για το semantic web θέτονται βάσει IEEE.
- Χρήση έγκυρων γλωσσών σήμανσης που να ανταποκρίνονται σε ένα δημοσιευμένο DTD ή Schema.
- Να παρέχει το ισοδύναμο της πληροφορίας σε κείμενο για κάθε πληροφορία που παρέχεται με γραφικά ή πολυμέσα.

- Να χρησιμοποιεί συνδέσμους ενσωματωμένους στο κείμενο.
- Να μην χρησιμοποιεί πλαίσια.
- Χρήση CSS αντί για HTML για την κατασκευή πινάκων.
- Συγγραφή της σελίδας με τρόπο τέτοιο ώστε όταν ο κώδικας διαβαστεί γραμμή προς γραμμή από user agents (όπως είναι τα screen readers) να παραμένει ευανάγνωστος.

Παρόλα αυτά η W3C επιτρέπει ως εξαίρεση στους πίνακες που προορίζονται για εμφάνιση στην ιστοσελίδα να παραμένουν κατανοητοί όταν γραμμικοποιούνται ή σε περίπτωση που κάτι αντίστοιχο μπορεί να εφαρμοστεί.

Η προσβασιμότητα των ιστοσελίδων αλλάζει επίσης εξαιτίας των Συστημάτων Διαχείρισης Περιεχομένου (Content Management Systems), τα οποία επιτρέπουν να γίνονται αλλαγές στις ιστοσελίδες χωρίς να απαιτείται ιδιαίτερη γνώση από προγραμματισμό.

Η Ανάπτυξη ιστοσελίδων απαιτεί τη συνεργασία πολλών διαφορετικών στοιχείων ώστε το διαδίκτυο να είναι προσβάσιμο σε ανθρώπους με αναπηρίες.

Αυτά τα στοιχεία περιλαμβάνουν:

- Περιεχόμενο – η πληροφορία που υπάρχει σε μια ιστοσελίδα ή σε κάποια εφαρμογή του διαδικτύου περιλαμβάνει:
  - Τη φυσική πληροφορία όπως κείμενο, εικόνες και ήχους
  - Κώδικα ή επισήμανση που διευκρινίζει τη δομή, παρουσίαση κλπ.
- Προγράμματα πλοήγησης, προγράμματα αναπαραγωγής πολυμέσων κλπ.
- Υποβοηθητική τεχνολογία, για ορισμένες περιπτώσεις – screen readers, εναλλακτικό πληκτρολόγιο, διακόπτες, προγράμματα σάρωσης κλπ.
- Γνώση του επιπέδου των χρηστών, των εμπειριών τους και σε ορισμένες περιπτώσεις προσαρμοσμένες στρατηγικές με χρήση του διαδικτύου
- Προγραμματιστές-Σχεδιαστές, συγγραφείς κτλ, συμπεριλαμβανομένου Προγραμματιστών με αναπηρίες και χρηστών που συνεισφέρουν στη διαμόρφωση του περιεχομένου
- Συντακτικών εργαλείων – προγραμμάτων που κατασκευάζουν ιστοσελίδες
- Εργαλεία Αξιολόγησης – Εργαλείων αξιολόγησης της Διαδικτυακή Προσβασιμότητας, HTML εγγυητών (HTML validators), CSS εγγυητών (CSS validators), κλπ.

### Ιστορικά Στοιχεία

Ο Τιμ Μπέρνερς-Λι δημοσίευσε αυτό που σήμερα θεωρούμε ότι ήταν η πρώτη ιστοσελίδα των Αύγουστο του 1991. Ο Τιμ Μπέρνερς-Λι ήταν ο πρώτος που συνδύασε τη Διαδικτυακή επικοινωνία (μέσω της οποίας μετέφερε την ηλεκτρονική αλληλογραφία και το Usenet επί δεκαετίες) με υπερκείμενο (Hypertext) (το οποίο επίσης υπήρχε για δεκαετίες με περιορισμένες δυνατότητες στην αναζήτηση πληροφοριών οι οποίες ήταν αποθηκευμένες σε έναν μόνο υπολογιστή). Οι ιστοσελίδες γράφονται σε γλώσσες σήμανσης κειμένου (markup languages) που ονομάζονται HTML. Οι προγενέστερες εκδόσεις της HTML ήταν περιορισμένων δυνατοτήτων, μόνο για να δίνουν στην ιστοσελίδα τη βασική της δομή (Επικεφαλίδες και παραγράφους), και τη δυνατότητα να χρησιμοποιούν υπερκείμενο. Αυτή ήταν μια νέα μορφή επικοινωνίας διαφορετική από τις ήδη υπάρχουσες-οι χρήστες μπορούν να οδηγηθούν σε άλλες σελίδες με συνεχόμενα συνδέσμων (hyperlinks).

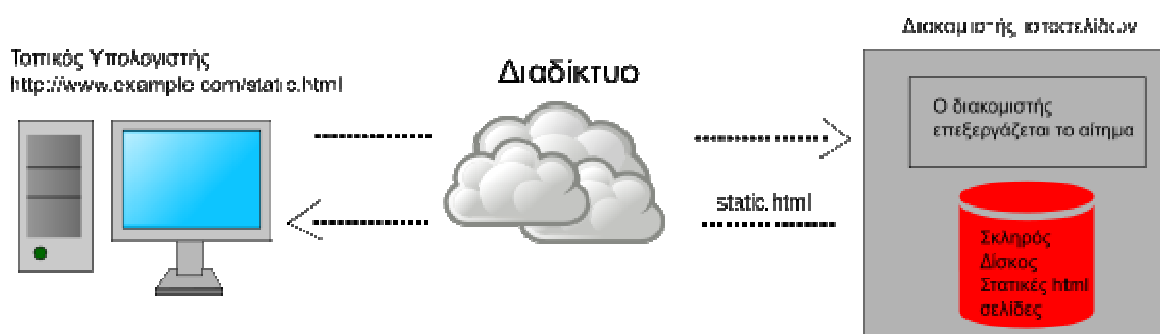
### Σχεδιασμός Ιστοσελίδων

Ο σκοπός του σχεδιασμού στο Διαδίκτυο είναι πολυσύνθετος αλλά βασικός κατά την εφαρμογή του. Πριν τη δημιουργία και το «ανέβασμα» μιας ιστοσελίδας είναι σημαντικό να σχεδιαστεί ακριβώς ότι είναι αναγκαίο για την ιστοσελίδα. Είναι πολύ σημαντικό να αποφασιστούν, το κοινό στο οποίο θα απευθύνεται, ο σκοπός της δημιουργίας της και το περιεχόμενο της.

## Περιεχόμενο

Ο σχεδιασμός που αφορά το διαδίκτυο είναι κατά μία έννοια παρόμοιος με την κλασσική τυπογραφία. Κάθε ιστοσελίδα είναι ένα σύνολο αναπαραγόμενων πληροφοριών, όπως ένα βιβλίο και κάθε σελίδα της ιστοσελίδας είναι το αντίστοιχο της σελίδας ενός βιβλίου. Ωστόσο στο web design χρησιμοποιείται ένα πλαίσιο εργασίας (framework) βασισμένο σε ένα ψηφιακό κώδικα και σε μια τεχνολογία απεικονίσεων για να δημιουργήσει και να διατηρήσει ένα περιβάλλον που θα διανέμει την πληροφορία σε πολλαπλές μορφές αρχείων (formats) όπως pdf, html, doc, κλπ. Οι δυνατότητες του web design το καθιστούν αδιαμφισβήτητα ως τον πλέον κομψό και σύνθετα αναπτυσσόμενο τρόπο επικοινωνίας στο σημερινό κόσμο.

### 2.2.8 Στατική ιστοσελίδα



**Εικόνα 3. Στατική ιστοσελίδα: Μεταφέρεται από τον διακομιστή στην μορφή που είναι αποθηκευμένη.**

**Στατική ιστοσελίδα** (Αγγλικά: static web-page ή flat web-page) ονομάζεται μια ιστοσελίδα της οποίας το περιεχόμενο μεταφέρεται στον χρήστη ακριβώς στην μορφή που είναι αποθηκευμένο στον εξυπηρετητή ιστοσελίδων (web server), σε αντίθεση με τις δυναμική ιστοσελίδα όπου το περιεχόμενο δημιουργείται από μια εφαρμογή η οποία εκτελείται στον εξυπηρετητή ιστοσελίδων.<sup>[1]</sup>

Τα περιεχόμενα μιας στατικής ιστοσελίδας εμφανίζονται με την ίδια μορφή σε όλους του χρήστες με την μορφή που είναι αποθηκευμένα στο σύστημα αρχείων του εξυπηρετητή ιστοσελίδων. Οι στατικές ιστοσελίδες είναι αποθηκευμένες συνήθως σε μορφή HTML και μεταφέρονται χρησιμοποιώντας το πρωτόκολλο HTTP.

## Πλεονεκτήματα και Μειονεκτήματα

### Πλεονεκτήματα

- Δεν χρειάζονται προγραμματιστικές δεξιότητες για να δημιουργήσει κάποιος μια στατική σελίδα.
- Η σελίδα μπορεί να βρίσκεται με εύκολο τρόπο κλωνοποιημένη σε περισσότερους από έναν εξυπηρετητές.
- Δεν χρειάζεται ειδικό λογισμικό στον εξυπηρετητή ιστοσελίδων για την δημοσίευση στατικών σελίδων.
- Η σελίδα μπορεί να είναι διαθέσιμη στον φυλλομετρητή κατευθείαν από ένα αποθηκευτικό μέσο όπως ένα CD-ROM ή USB μνήμης χωρίς να χρειάζεται να διαμεσολαβήσει ένα εξυπηρετητής ιστοσελίδων με κατάλληλο λογισμικό χειρισμό ιστοσελίδων (για παράδειγμα οι δυναμικές σελίδες σε γλώσσα PHP θέλουν έναν διερμηνέα PHP ο οποίος τρέχει στον

εξυπηρετητή ιστοσελίδων και δημιουργεί δυναμικά την σελίδα πριν αυτή αποσταλεί στον φυλλομετρητή του χρήστη).

### Μειονεκτήματα

- Δεν είναι εύκολη η διαδραστικότητα με τον χρήστη.
- Η διαχείριση μεγάλου αριθμού στατικών ιστοσελίδων δεν είναι εύκολη χωρίς αυτόματα εργαλεία. Θα πρέπει να αλλάχουν όλες οι στατικά αποθηκευμένες ιστοσελίδες.

### 2.2.9 Δυναμική ιστοσελίδα



**Εικόνα 4 . Δυναμική ιστοσελίδα: παράδειγμα δημιουργίας δυναμικής ιστοσελίδας στον εξυπηρετητή. Η σελίδα login.php περιέχει σενάριο εντολών PHP το οποίο εκτελείται από τον διερμηνέα PHP στον εξυπηρετητή και χρησιμοποιώντας στοιχεία από την βάση MySQL αποστέλνεται πίσω η δυναμική σελίδα HTML.**

**Δυναμική ιστοσελίδα** (Αγγλικά: dynamic web page) είναι μια ιστοσελίδα η οποία δημιουργείται δυναμικά την στιγμή της πρόσβασης σε αυτή ή την στιγμή που ο χρήστης αλληλεπιδρά με τον εξυπηρετητή ιστοσελίδων. Οι δυναμικές ιστοσελίδες θεωρούνται δομικό στοιχείο της νέας γενιάς του παγκόσμιου ιστού (Web 2.0) όπου η πληροφορία διαμοιράζεται σε πολλαπλές ιστοσελίδες.

Η δυναμική ιστοσελίδα μπορεί να δημιουργείται δυναμικά από ένα σενάριο εντολών, το οποίο εκτελείται τοπικά στο πελάτη ή στον εξυπηρετητή ή και στον πελάτη και στον εξυπηρετητή.

### Ιστορία

Είναι δύσκολο να πει κάποιος με ακρίβεια πότε οι "δυναμικές ιστοσελίδες" ξεκίνησαν αλλά η γενικότερη ιδέα ξεκίνησε μετά την εξάπλωση και ανάπτυξη των ιστοσελίδων. Το πρωτόκολλο HTTP άρχισε να χρησιμοποιείται από το 1990 και η γλώσσα HTML ξεκίνησε να χρησιμοποιείται για τον σχεδιασμό ιστοσελίδων από το 1996. Η εξάπλωση χρήσης των φυλλομετρητών ξεκίνησε το 1993 με τον φυλλομετρητή Mosaic. Η ιδέα της δημιουργίας δυναμικών ιστοσελίδων υπήρχε και πριν την ανάπτυξη των ιστοσελίδων. Για παράδειγμα πριν το 1990 υπήρχαν "Βάσεις Δεδομένων" (Bulletin Board Systems) στα οποία οι χρήστες συνδέονταν μέσω μόντεμ και το περιεχόμενο που έβλεπαν δημιουργούνταν δυναμικά.

### Δυναμική ιστοσελίδα τοπικά στο πελάτη

Η σελίδα συνήθως περιέχει ένα σενάριο εντολών το οποίο ενσωματώνει μια διαδραστικότητα με τον χρήστη (η διαδραστικότητα γίνεται με χρήση πληκτρολογίου - ποντικιού). Το περιεχόμενο της ιστοσελίδας δημιουργείται δυναμικά στον τοπικό υπολογιστή μετά την εκτέλεση του σεναρίου εντολών το οποίο λαμβάνεται από τον απομακρυσμένο εξυπηρετητή.



Για παράδειγμα η γλώσσα σεναρίου εντολών JavaScript ή ActionScript (γλώσσα σεναρίων της Macromedia - σήμερα είναι η εταιρία Adobe Systems) χρησιμοποιείται για την τοπική δημιουργία δυναμικών ιστοσελίδων τεχνολογίας DHTML ή Flash οι οποίες συχνά χρησιμοποιούνται για την ενσωμάτωση ήχων, κινούμενων εικόνων ή αλλαγές και διαμόρφωση του κείμενου της σελίδας. Η γλώσσα σεναρίου η οποία εκτελείται τοπικά μπορεί να καλεί απομακρυσμένα σεσάρια εντολών στον εξυπηρετητή ιστοσελίδων χρησιμοποιώντας τεχνολογίες όπως hidden Frame, XMLHttpRequest ή άλλες υπηρεσίες ιστού (web services).

Η πρώτη "ευρέως διαδεδομένη" έκδοση της γλώσσας σεναρίου εντολών JavaScript ενσωματώθηκε το 1996 στο φυλλομετρητή Netscape 3.

### **Δυναμική ιστοσελίδα στον εξυπηρετητή**

Μια δυναμική ιστοσελίδα συνήθως δημιουργείται από ένα λογισμικό το οποίο τρέχει στον απομακρυσμένο εξυπηρετητή ιστοσελίδων. Ο χρήστης-πελάτης επισκέπτεται τον εξυπηρετητή ιστοσελίδων ο οποίος δημιουργεί δυναμικά την ιστοσελίδα την οποία επιστρέφει σε μορφή HTML στον πελάτη. Τέτοιες τεχνολογίες για την δημιουργία δυναμικών ιστοσελίδων στον εξυπηρετητή είναι η τεχνολογίες ASP, ColdFusion, Perl, PHP, WebDNA και άλλες. Οι γλώσσες αυτές συνήθως (με εξαίρεση την ASP.NET και την JSP) χρησιμοποιούν ένα σεσάριο εντολών CGI κατά την δυναμική δημιουργία των σελίδων.

Πολλές φορές οι σελίδες είναι αποθηκευμένες στατικά σε κρυφή-προσωρινή μνήμη και αποστέλλονται κατευθείαν στο πελάτη. Με αυτό το τρόπο αποφεύγεται η υπερφόρτωση του εξυπηρετητή ιστοσελίδων και καλείται η μηχανή δημιουργία δυναμικών ιστοσελίδων μόνο όταν το αποθηκευμένο περιεχόμενο έχει αλλάξει.

### **Συνδυασμός δημιουργίας δυναμικής σελίδας στον πελάτη και στον εξυπηρετητή**

Η τεχνολογία AJAX χρησιμοποιεί σεσάρια εντολών και στον πελάτη και στον απομακρυσμένο εξυπηρετητή για την δημιουργία της δυναμικής ιστοσελίδας. Στην τεχνολογία AJAX γίνεται ανταλλαγή περιεχομένου και δεδομένων από τον υπολογιστή του πελάτη και του απομακρυσμένου εξυπηρετητή και η σελίδα διαμορφώνεται από το σεσάριο εντολών το οποίο τρέχει τοπικά στο πελάτη. Το πλεονέκτημα της τεχνολογίας αυτής είναι ότι περιορίζεται η χρήση του εξυπηρετητή ιστοσελίδων και αποστέλλεται μόνο το περιεχόμενο. Για παράδειγμα οι χάρτες της Google είναι ένα παράδειγμα δικτυακής εφαρμογής η οποία χρησιμοποιεί την τεχνολογία AJAX.

## Κεφάλαιο 3

### Τεχνολογίες και Εργαλεία Ανάπτυξης

#### 3.1 Τι είναι HTML

Η **HTML** (ακρωνύμιο του αγγλικού **HyperText Markup Language**, ελλ. Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων.

Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από *ετικέτες*, οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα `<html>`), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα `<h1>` και `</h1>`), με την πρώτη να ονομάζεται *ετικέτα έναρξης* και τη δεύτερη *ετικέτα λήξης* (ή σε άλλες περιπτώσεις *ετικέτα ανοίγματος* και *ετικέτα κλεισίματος* αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.

Ο σκοπός ενός web browser είναι να διαβάσει τα έγγραφα HTML και τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας.

Τα στοιχεία της HTML χρησιμοποιούνται για να κτίσουν όλους του ιστότοπους. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου) καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, παραθέσεις και άλλα. Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

Οι Web browsers μπορούν επίσης να αναφέρονται σε στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου.

#### Προέλευση

Το 1980, ο φυσικός Τιμ Μπέρνερς Λι, ο οποίος εργαζόταν στο CERN, επινόησε το ENQUIRE, ένα σύστημα χρήσης και διαμοιρασμού εγγράφων για τους ερευνητές του CERN, και κατασκεύασε ένα πρωτότυπό του. Αργότερα, το 1989, πρότεινε ένα σύστημα βασισμένο στο διαδίκτυο, το οποίο θα χρησιμοποιούσε υπερκείμενο. Έτσι, έφτιαξε την προδιαγραφή της HTML και έγραψε τον browser και το λογισμικό εξυπηρετητή στα τέλη του 1990. Τον ίδιο χρόνο, ο Μπέρνερς Λι και ο μηχανικός συστημάτων πληροφορικής του CERN Robert Cailliau συνεργάστηκαν σε μια κοινή προσπάθεια εύρεσης χρηματοδότησης, αλλά το έργο δεν υιοθετήθηκε ποτέ επίσημα από το CERN. Στις προσωπικές του σημειώσεις από το 1990<sup>[3]</sup>, ο Μπέρνερς Λι αριθμεί «μερικές από τις πολλές χρήσεις του υπερκειμένου», και αναφέρει πρώτα από όλες μια εγκυκλοπαίδεια.



## Πρώτες προδιαγραφές

Η πρώτη δημόσια διαθέσιμη περιγραφή της HTML ήταν ένα έγγραφο με το όνομα *Ετικέτες HTML*, το οποίο πρωτοαναφέρθηκε στο Διαδίκτυο από τον Μπέρνερς Λι στα τέλη του 1991. Περιέγραφε τα 20 στοιχεία τα οποία αποτελούσαν τον αρχικό και σχετικά απλό σχεδιασμό της HTML. Εκτός από την ετικέτα υπερσυνδέσμου, οι υπόλοιπες ήταν έντονα επηρεασμένες από την SGMLguid, μια μορφή δημιουργίας τεκμηρίωσης, φτιαγμένη στο CERN και βασισμένη στην SGML. Δεκατρία από εκείνα τα αρχικά στοιχεία υπάρχουν ακόμα σήμερα στην HTML 4.

Το ίδιο το πρότυπο SGML αναπαράγει μερικές από τις τεχνικές των τυπογράφων, αλλά εκτός από απλή μίμηση της τυπογραφίας προσθέτει γενικευμένη σήμανση βασισμένη σε στοιχεία, τα οποία μπορούν να εμφωλεύονται το ένα μέσα στο άλλο και να φέρουν ιδιότητες. Ακόμα, το SGML διαχωρίζει τη δομή από το περιεχόμενο, κατεύθυνση προς την οποία αργότερα κινήθηκε και η HTML, με τα CSS. Πολλά από τα στοιχεία κειμένου προέρχονται από την τεχνική αναφορά ISO TR 9537, *Techniques for using SGML* (τεχνικές χρήσης της SGML), η οποία με τη σειρά της καλύπτει τα χαρακτηριστικά των πρώιμων γλωσσών μορφοποίησης κειμένου που χρησιμοποιούνταν από τα TYPSET και RUNOFF, και είχαν αναπτυχθεί στις αρχές της δεκαετίας του 1960 για το λειτουργικό σύστημα CTSS.

Ο Μπέρνερς Λι θεώρησε την HTML ως μια υλοποίηση του SGML. Αυτό ορίστηκε και επίσημα από το Internet Engineering Task Force (IETF) με τη δημοσίευση της πρώτης πρότασης για μια προδιαγραφή της HTML, στα μέσα του 1993, η οποία περιλάμβανε και έναν Ορισμό τύπου εγγράφου (DTD, Document Type Definition) της SGML, ο οποίος όριζε την γραμματική. Αυτό το πρόχειρο έληξε μετά την πάροδο έξι μηνών, αλλά περιέχει κάτι αξιοσημείωτο: την αναγνώριση της ετικέτας του NCSA Mosaic για την ενσωμάτωση εικόνων μέσα στο κείμενο, η οποία αντικατοπτρίζει την φιλοσοφία του IETF για ενσωμάτωση επιτυχημένων πρωτότυπων μέσα στα πρότυπα. Κάτι παρόμοιο περιείχε και το ανταγωνιστικό πρόχειρο του Dave Raggett, «HTML+ (Hypertext Markup Format)», από τα τέλη του, το οποίο πρότεινε την προτυποποίηση μερικών ήδη υλοποιημένων δυνατοτήτων, όπως οι πίνακες και οι φόρμες.

Μετά που τα πρόχειρα HTML και HTML+ έληξαν, στις αρχές του 1994, το IETF δημιούργησε την Ομάδα Εργασίας για την HTML, η οποία το 1995 ολοκλήρωσε την «HTML 2.0», με την πρόθεση να αποτελέσει την πρώτη προδιαγραφή πάνω στην οποία θα βασίζονταν οι μελλοντικές υλοποιήσεις. Η HTML 2.0 δημοσιεύτηκε ως RFC 1866, και περιείχε ιδέες από τα πρόχειρα HTML και HTML+. Η αριθμηση 2.0 σκόπευε απλά να ξεχωρίσει την νέα έκδοση από τα πρόχειρα που προηγήθηκαν. designation was intended to distinguish the new edition from previous drafts.

Η περαιτέρω ανάπτυξη κάτω από την επίβλεψη του IETF καθυστέρησε λόγω σύγκρουσης ενδιαφερόντων. Από το 1996 και μετά, οι προδιαγραφές της HTML τηρούνται, μαζί με ανάδραση από τους δημιουργούς λογισμικού, από το World Wide Web Consortium (W3C). Ωστόσο, το 2000 η HTML έγινε επίσης παγκόσμιο πρότυπο (ISO/IEC 15445:2000). Η τελευταία προδιαγραφή της HTML, η HTML 4.01 δημοσιεύτηκε από το W3C το 1999, και το 2001 δημοσιεύτηκαν επίσης και τα λάθη και οι παραλείψεις της (errata).

## Δημιουργία με προγράμματα WYSIWYG

Ένα θέμα αντιπαράθεσης στις κοινότητες των δημιουργών περιεχομένου για τον Ιστό είναι η χρήση προγραμμάτων WYSIWYG (What You See Is What You Get), δηλαδή προγραμμάτων στα οποία ο χρήστης σχεδιάζει οπτικά τη σελίδα όπως θα φαίνεται μετά στον browser του επισκέπτη. Αυτό κάνει τη γνώση της HTML προαιρετική για την κατασκευή μιας ιστοσελίδας. Αυτό το μοντέλο δημιουργίας με προγράμματα WYSIWYG έχει γνωρίσει έντονες κριτικές, κυρίως λόγω της κακής ποιότητας του κώδικα που δημιουργείται αυτόματα. Παρόλα αυτά, οι επεξεργαστές ιστοσελίδων WYSIWYG είναι δημοφιλείς λόγω της ευκολίας που παρέχουν.

## HTML 5 Η νεότερη έκδοση

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

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

Οι ιδέες πίσω από την HTML5 εμφανίστηκαν αρχικά το 2004 από την ομάδα WHATWG. Η HTML5 εμπεριέχει το πρότυπο *Web Forms 2.0* που είναι επίσης της WHATWG.

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

## Σήμανση

Η σήμανση HTML αποτελείται από μερικά βασικά συστατικά, συμπεριλαμβανομένων των *στοιχείων* (και των *ιδιοτήτων* τους), τους βασισμένους σε χαρακτήρες *τύπους δεδομένων*, τις *αναφορές χαρακτήρων* και τις *αναφορές οντοτήτων*. Ένα ξεχωριστό σημαντικό συστατικό είναι η *δήλωση τύπου εγγράφου* (document type declaration), η οποία ορίζει στον browser τον τρόπο εμφάνισης της σελίδας.

Στην HTML, το πρόγραμμα Hello world, ένα συνηθισμένο πρόγραμμα υπολογιστή που χρησιμεύει για τη σύγκριση γλωσσών προγραμματισμού, γλωσσών σεναρίων και γλωσσών σήμανσης, φτιάχνεται με 9 γραμμές κώδικα, παρότι οι νέες γραμμές είναι προαιρετικές στην HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Hello HTML</title>
  </head>
  <body>
    <p>Hello world</p>
  </body>
</html>
```

(Το κείμενο ανάμεσα στο `<html>` και το `</html>` περιγράφει την ιστοσελίδα, και το κείμενο μεταξύ του `<body>` και του `</body>` είναι το ορατό μέρος της. Το σημασμένο κείμενο `<title>Hello HTML</title>` καθορίζει τον τίτλο που θα εμφανίζεται στην μπάρα τίτλου του browser.)

Το Document Type Declaration στον πιο πάνω κώδικα είναι για την HTML5. Αν δεν συμπεριλαμβάνεται η δήλωση `<!doctype html>`, τότε μερικοί browser θα καταφύγουν στην λειτουργία quirks για την εμφάνιση της σελίδας.

## Στοιχεία

Τα έγγραφα HTML αποτελούνται από στοιχεία HTML τα οποία στην πιο γενική μορφή τους έχουν τρία συστατικά: ένα ζεύγος από *ετικέτες*, την «ετικέτα εκκίνησης» και την «ετικέτα τερματισμού», μερικές *ιδιότητες* μέσα στην ετικέτα εκκίνησης, και τέλος το κείμενο ή το γραφικό *περιεχόμενο* μεταξύ των ετικετών, το οποίο μπορεί να περιλαμβάνει και άλλα στοιχεία εμφωλευμένα μέσα του. Το στοιχείο HTML μπορεί να είναι οτιδήποτε ανάμεσα στις ετικέτες εκκίνησης και τερματισμού. Τέλος, κάθε ετικέτα περικλείεται σε σύμβολα «μεγαλύτερο από» και «μικρότερο από», δηλαδή < και >.

Επομένως, η γενική μορφή ενός στοιχείου HTML είναι: <tag attribute1="value1" attribute2="value2">content</tag>. Μερικά στοιχεία HTML περιγράφονται ως *άδεια στοιχεία*, έχουν τη μορφή <tag attribute1="value1" attribute2="value2" >, και δεν έχουν καθόλου περιεχόμενο. Το όνομα κάθε στοιχείου HTML είναι το ίδιο όνομα που χρησιμοποιείται στις αντίστοιχες ετικέτες. Το όνομα της ετικέτας τερματισμού ξεκινά με μια κάθετο </>, η οποία παραλείπεται στα άδεια στοιχεία. Τέλος, αν δεν αναφέρονται ρητά οι ιδιότητες ενός στοιχείου, τότε χρησιμοποιούνται οι προεπιλογές σε κάθε περίπτωση.

## Παραδείγματα στοιχείων

Κεφαλίδα του εγγράφου HTML:<head>...</head>. Συνήθως περιέχει τον τίτλο, π.χ:

```
<head>
  <title>The title</title>
</head>
```

Επικεφαλίδες: οι επικεφαλίδες στην HTML ορίζονται με τις ετικέτες <h1> έως <h6>:

```
<h1>Επικεφαλίδα1</h1>
<h2>Επικεφαλίδα2</h2>
<h3>Επικεφαλίδα3</h3>
<h4>Επικεφαλίδα4</h4>
<h5>Επικεφαλίδα5</h5>
<h6>Επικεφαλίδα6</h6>
```

Παράγραφοι:

```
<p>Paragraph 1</p> <p>Paragraph 2</p>
```

Αλλαγή γραμμής: <br>. Η διαφορά ανάμεσα στο <br> και το <p> είναι ότι το <br> αλλάζει γραμμή χωρίς να αλλάζει την σημαντική δομή της σελίδας, ενώ το <p> τεμαχίζει τη σελίδα σε παραγράφους. Το <br> είναι ένα *άδειο στοιχείο*, δηλαδή δεν έχει περιεχόμενο, ούτε και χρειάζεται ετικέτα τερματισμού.

```
<p>This <br> is a paragraph <br> with <br> line breaks</p>
```

Σχόλια:

```
<!-- This is a comment -->
```

Τα σχόλια μπορούν να βοηθήσουν στην κατανόηση της σήμανσης. Δεν εμφανίζονται στην ιστοσελίδα.

### Υπάρχουν διάφοροι τύπου στοιχείων στην HTML.

- Η δομική σήμανση περιγράφει τον σκοπό του κειμένου. Για παράδειγμα, το `<h2>Golf</h2>` σημαίνει ότι η λέξη «Golf» θα είναι μια επικεφαλίδα δεύτερου επιπέδου. Η δομική σήμανση δεν εμπεριέχει συγκεκριμένο τρόπο εμφάνισης, αλλά οι περισσότεροι browser έχουν δικό τους προκαθορισμένο τρόπο για τη μορφοποίηση όλων των στοιχείων. Το περιεχόμενο μπορεί να μορφοποιηθεί οπτικά με τη χρήση των CSS.
- Η σήμανση οπτικής μορφοποίησης περιγράφει την εμφάνιση του κειμένου, άσχετα από τον σκοπό του. Για παράδειγμα το `<b>έντονο κείμενο</b>` υποδεικνύει ότι οι συσκευές που θα εμφανίσουν το κείμενο θα το κάνουν έντονο, αλλά δεν λέει τι θα κάνουν με το κείμενο οι συσκευές που δεν μπορούν να το εμφανίσουν, όπως για παράδειγμα οι συσκευές φωνητικής ανάγνωσης σελίδων. Τόσο στην περίπτωση του `<b>έντονο</b>` όσο και του `<i>πλάγιο</i>`, υπάρχουν άλλα στοιχεία τα οποία μπορεί να οπτικά να εμφανίζονται ίδια, αλλά είναι πιο σημαντικά, όπως τα `<strong>σημαντικό κείμενο</strong>` και `<em>κείμενο με έμφαση</em>` αντίστοιχα. Είναι ευκολότερα να ερμηνεύσει ένας φωνητικός browser τα τελευταία δύο στοιχεία. Ωστόσο, δεν είναι ισότιμα με τα αντίστοιχα στοιχεία οπτικής μορφοποίησης. Για παράδειγμα ένα πρόγραμμα φωνητικής ανάγνωσης της σελίδας δεν θα πρέπει να προφέρει με έμφαση τον τίτλο ενός βιβλίου, αλλά όταν το ίδιο περιεχόμενο εμφανίζεται σε μια οθόνη τότε ο τίτλος θα είναι με πλάγια γράμματα. Τα περισσότερα στοιχεία οπτικής μορφοποίησης έχουν θεωρούνται ξεπερασμένα μετά την προδιαγραφή HTML 4.0, και έχουν αντικατασταθεί από τα CSS.
- Η σήμανση υπερκειμένου κάνει μερικά τμήματα ενός εγγράφου να συνδέουν με άλλα έγγραφα. Τα στοιχεία anchor δημιουργεί έναν υπερσύνδεσμο στο έγγραφο, και η ιδιότητα href ορίζει τον στόχο του συνδέσμου. Για παράδειγμα, η σήμανση HTML `<a href="http://el.wikipedia.org/">Βικιπαίδεια</a>` εμφανίζει το κείμενο «Wikipedia» ως υπερσύνδεσμο. Για την εμφάνιση μιας εικόνας ως συνδέσμου, μπορεί να εισαχθεί ένα στοιχείο `<img>` ως περιεχόμενο του στοιχείου «a». Όπως και το «br», το `<img>` είναι άδειο στοιχείο. Έχει ιδιότητες αλλά δεν έχει περιεχόμενο, ούτε ετικέτα τερματισμού: `<a href="http://example.org/"></a>`.

## 3.2 Η Γλώσσα PHP

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.

### Επεκτάσεις αρχείων και διακομιστές

Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. \*.php, \*.php4, \*.phtml κ.ά.). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται και να μεταγλωττίζει τον κώδικα PHP σε HTML που καταλαβαίνει το πρόγραμμα πελάτη. Ο διακομιστής Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με τα λειτουργικά συστήματα GNU/Linux, Microsoft Windows, Mac OS X υποστηρίζει εξ ορισμού την εκτέλεση κώδικα PHP, είτε με την χρήση ενός πρόσθετου (mod\_php) ή με την αποστολή του κώδικα προς εκτέλεση σε εξωτερική διεργασία CGI ή FCGI ή με την έλευση της php5.4 υποστηρίζονται η εκτέλεση σε πολυάσχολους ιστοχώρους, FastCGI Process Manager (FPM).

## Εναλλακτικός τρόπος εκτέλεσης ιστοσελίδων χωρίς χρονοβόρες διαδικασίες

Ο συνδυασμός Linux/Apache/PHP/MySQL, που είναι η πιο δημοφιλής πλατφόρμα εκτέλεσης ιστοσελίδων είναι γνωστός και με το ακρωνύμιο LAMP. Παρόμοια, ο συνδυασμός \*/Apache/PHP/MySQL ονομάζεται \*AMP, όπου το πρώτο αρχικό αντιστοιχεί στην πλατφόρμα, στην οποία εγκαθίστανται ο Apache, η PHP και η MySQL (π.χ. Windows, Mac OS X).

Ο LAMP συνήθως εγκαθίσταται και ρυθμίζεται στο Linux με τη βοήθεια του διαχειριστή πακέτων της εκάστοτε διανομής. Στην περίπτωση άλλων λειτουργικών συστημάτων, επειδή το κατέβασμα και η ρύθμιση των ξεχωριστών προγραμμάτων μπορεί να είναι πολύπλοκη, υπάρχουν έτοιμα πακέτα προς εγκατάσταση, όπως το XAMPP και το WAMP για τα Windows και το MAMP για το Mac OS X.

## Ιστορία της PHP

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, βασιζόμενη αυτή τη φορά στη γλώσσα C και αριθμώντας περισσότερους από 50.000 ιστότοπους που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Οι περισσότεροι ιστότοποι επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP.

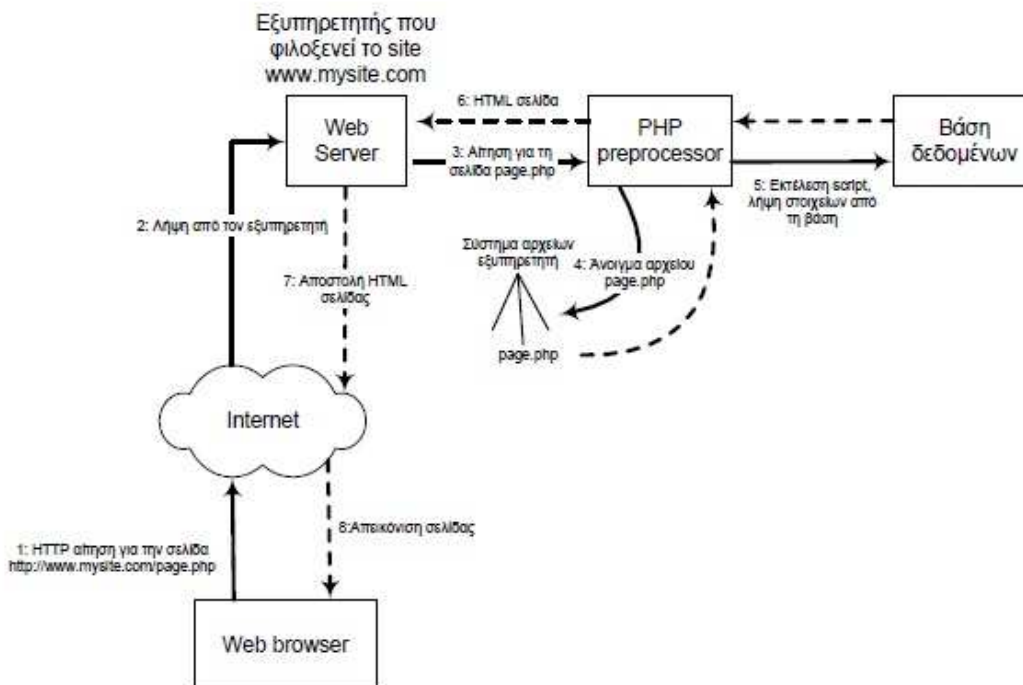
## Πλεονεκτήματα χρήσης

Η PHP είναι ιδιαίτερα δημοφιλής για την ανάπτυξη web εφαρμογών με κυριότερα πλεονεκτήματα τα εξής:

- είναι γρήγορη στην εκτέλεση,
- έχει δυνατότητα επικοινωνίας με τις περισσότερες εμπορικές βάσεις δεδομένων,
- διαθέτει πολλές βιβλιοθήκες,
- διανέμεται και χρησιμοποιείται δωρεάν,
- είναι εύκολη στην εκμάθησή της καθώς μοιάζει με τις C/C++, Java και perl,
- υποστηρίζει τα τυπικά γνωρίσματα μιας αντικειμενοστρεφούς γλώσσας, (κλάσεις, κληρονομικότητα, πολυμορφισμός),
- είναι μεταφέρσιμη,
- η υλοποίησή της μηχανής εκτέλεσής της είναι ανοικτού κώδικα,
- υποστηρίζεται αποτελεσματικά από την εταιρεία Zend Technologies



### Στάδια επεξεργασίας μιας PHP σελίδας



Εικόνα 5. Στάδια επεξεργασίας μιας PHP σελίδας

Θεωρούμε ότι ο web browser εκτελείται σε ένα υπολογιστή ο οποίος είναι συνδεδεμένος στο διαδίκτυο. Ο χρήστης ζητάει τη σελίδα `page.php` που βρίσκεται στο δικτυακό τόπο `www.mysite.com`. Ο παραπάνω δικτυακός τόπος φιλοξενείται σε εξυπηρετητή ο οποίος είναι συνδεδεμένος στο διαδίκτυο και εκτελεί ένα web server π.χ. τον Apache web server (<http://www.apache.org>). Η μηχανή εκτέλεσης της PHP (PHP Preprocessor) εκτελείται α) ως υποσύστημα του web server (το οποίο φορτώνεται κατά την έναρξη της εκτέλεσής του) ή β) ως πρόγραμμα το οποίο καλείται από τον web server όταν του ζητηθεί η εξυπηρέτηση μιας σελίδας `*.php`. Στην πρώτη περίπτωση ο εκτελέσιμος κώδικας του PHP Preprocessor αποτελεί μια βιβλιοθήκη η οποία φορτώνεται δυναμικά από τον web server, ενώ στη δεύτερη έχει τη μορφή προγράμματος που βρίσκεται στον κατάλογο `/cgi-bin` της εγκατάστασης του web server. Η «Βάση Δεδομένων» μπορεί να βρίσκεται στο ίδιο μηχάνημα με τον web server ή σε διαφορετικό (το οποίο είναι προσβάσιμο μέσω δικτύου).

Ας δούμε πιο αναλυτικά τα στάδια επεξεργασίας μιας PHP σελίδας:

1. Ο χρήστης ζητάει την εμφάνιση της σελίδας `page.php` βάζοντας τη διεύθυνση `www.mysite.com/page.php`.
2. Η αίτηση του χρήστη στέλνεται μέσω του web browser στην διεύθυνση `www.mysite.com` όπου παραλαμβάνεται από τον web server.
3. Ο web server εξετάζει την κατάληξη (`*.php`) της ζητούμενης σελίδας και διαπιστώνει ότι είναι PHP σελίδα. Έτσι αναθέτει την επεξεργασία της σε πρώτο στάδιο στον PHP Preprocessor. Η ανάθεση γίνεται γιατί ο web server δεν μπορεί να εκτελέσει τα PHP scripts που ενσωματώνει η σελίδα. Οι καταλήξεις των σελίδων που προωθούνται στον PHP Preprocessor για την επεξεργασία τους ορίζονται στο αρχείο παραμέτρων του web server. Στην περίπτωση του Apache το αρχείο αυτό είναι το `httpd.conf` το οποίο βρίσκεται στο φάκελο `conf` της εγκατάστασης της Windows εγκατάστασης του Apache.

4. Ο PHP Preprocessor παραλαμβάνει την σελίδα page.php από το τοπικό σύστημα αρχείων κάνει συντακτικό έλεγχο και προχωρά στην εκτέλεση των scripts που ενσωματώνει. Η εκτέλεση ενός PHP script μπορεί να παράγει HTML κώδικα. Ο κώδικας αυτός παρεμβάλλεται στη θέση του script, το οποίο αφαιρείται από την HTML σελίδα μετά την εκτέλεσή του. Έτσι το αποτέλεσμα της εκτέλεσης μιας PHP σελίδας (η οποία αποτελείται από HTML και PHP κώδικα) είναι μια καθαρά HTML σελίδα.
5. Κατά την εκτέλεσή του, ένα PHP script μπορεί να ανοίγει σύνδεση με κάποια βάση δεδομένων και να λαμβάνει από αυτή δεδομένα, πιθανώς για την ενσωμάτωσή τους στην HTML σελίδα.
6. Όταν ολοκληρωθεί η εκτέλεση των PHP scripts μιας PHP σελίδας, η παραγόμενη HTML σελίδα επιστρέφεται στον web server.
7. Ο web server αναλαμβάνει την αποστολή της σελίδας στο μηχάνημα από το οποίο παρέλαβε την αντίστοιχη αίτηση.
8. Η σελίδα παραλαμβάνεται και απεικονίζεται από το browser του χρήστη.

### Παράδειγμα κώδικα PHP

Ο κώδικας PHP μιας ιστοσελίδας περικλείεται από τα tags `<?php` και `?>`

Την απεικόνιση κειμένου αναλαμβάνουν οι εντολές `print` ή `echo`.

Για παράδειγμα, ο ακόλουθος κώδικας PHP:

```
<?php
echo "Hello world!";
?>
```

Θα εμφανίσει στο πρόγραμμα περιήγησης των επισκεπτών τη φράση:

Hello world!

### Μεταβλητές

Όπως οι περισσότερες γλώσσες προγραμματισμού, έτσι και η PHP υποστηρίζει μεταβλητές. Στην PHP οι μεταβλητές φέρουν ως πρόθεμα τον χαρακτήρα του δολαρίου (\$).

Ένα πολύ απλό παράδειγμα όπου ορίζεται η τιμή μιας μεταβλητής και στη συνέχεια απεικονίζεται στον browser των επισκεπτών, είναι το εξής:

```
<?php
$text = "Καλημέρα!";
print $text;

?>
```

Η εκτέλεση του παραπάνω κώδικα θα εμφανίσει το περιεχόμενο της μεταβλητής \$text, δηλαδή τη φράση:

Καλημέρα!

## Operators

Οι operators θέτουν προϋποθέσεις για την εκτέλεση ενός τμήματος κώδικα. Οι operators διατυπώνονται με τις εντολές **if**, **elseif**, **else** ή **switch/case** σε συνδυασμό με τα σύμβολα **!=**, **==**, **>**, **<**, **<=**, **>=**.

Ολοκληρωμένο παράδειγμα με τα if, elseif και else:

```
<?php
// Αν το $mynumber είναι ίσο με 6
if ($mynumber == 6) {
print "text 6";
print "hello";
}

// ή αν το $mynumber είναι ίσο με 8
elseif ($mynumber == 8) {
print "text 8";
}

// αλλιώς, για οποιαδήποτε άλλη τιμή του $mynumber...
else {
print "text other";
}
?>
```

## GET και POST data

Αν μέχρι τώρα έχετε επισκεφθεί πολλά sites με σελίδες php, σίγουρα θα έχετε παρατηρήσει ότι συχνά στο URL τους υπάρχει ο χαρακτήρας του αγγλικού ερωτηματικού ? ακολουθούμενο από άλλες παραμέτρους. Π.χ. κάτι σαν `article.php?id=256&page=news&t=25432&something=other`

Οι παράμετροι αυτές λαμβάνονται από τον web server και αφού ενσωματωθούν ως μεταβλητές στο αρχείο PHP, υφίστανται επεξεργασία ώστε η σελίδα να εμφανίσει συγκεκριμένες πληροφορίες. Ακολουθεί ένα παράδειγμα κώδικα όπου τα GET data χρησιμοποιούνται για να εμφανίσουν διαφορετικό κείμενο:

```
<?php
$id = $_GET['id'];

if ($id == 1) {
print "Κείμενο 1";
}
elseif ($id == 2) {
print "Δεύτερο κείμενο";
}
else {
print "Άλλο κείμενο";
}
?>
```

Υποθέτουμε ότι το αρχείο με τον παραπάνω κώδικα ονομάζεται `page.php`

Ο επισκέπτης που θα ανοίξει τη σελίδα με τη μορφή `page.php?id=1` θα δει το κείμενο "Κείμενο 1". Αν



ανοίξει τη σελίδα σαν `page.php?id=2` θα δει στην οθόνη του τη φράση "Δεύτερο κείμενο", ενώ για οποιοδήποτε άλλο `id`, θα δει τη φράση "Άλλο κείμενο".

Αν ο αριθμός του `id` πρόκειται να σταλεί μέσω μια φόρμας που χρησιμοποιεί τη μέθοδο `POST`, αρκεί να αντικατασταθεί η γραμμή:

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

με την ακόλουθη:

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

Προειδοποίηση ασφαλείας

Σε περίπτωση που το `$id` πρόκειται να χρησιμοποιηθεί σε επερωτήματα βάσεων δεδομένων, έχοντας πάντα αριθμητική τιμή, προτείνεται η χρήση του `intval()` για την αποφυγή απώλειας δεδομένων από τον `server` μέσω `SQL injections`. Με τη χρήση του `intval`, αποκόπτονται οι μη αριθμητικοί χαρακτήρες από μια μεταβλητή.

Η λειτουργία του εξηγείται στο ακόλουθο παράδειγμα:

```
<?php
$text = "something223344";
$text = intval($text);
print "$text";
?>
```

που θα έχει σαν αποτέλεσμα την εμφάνιση του αριθμού 223344

### 3.3 Εισαγωγή στις Βάσεις Δεδομένων

#### Τι Είναι οι Βάσεις Δεδομένων (DataBases)

Μια **Βάση Δεδομένων (DataBase)** είναι ένας οργανωμένος τρόπος αποθήκευσης πληροφοριών και πρόσβασής τους με πολλούς τρόπους με διάφορα προγράμματα. Μια βάση δεδομένων είναι κάτι παραπάνω από μια απλή συλλογή αποθηκευμένων στοιχείων.

Ένας άλλος ορισμός είναι ότι μια βάση δεδομένων είναι ένα ολοκληρωμένο σύστημα που αποτελείται από δεδομένα (`data`) και από το κατάλληλο λογισμικό (`software`), τα οποία χρησιμοποιώντας το υλικό (`hardware`) βοηθούν στην ενημέρωση και πληροφόρηση των χρηστών (`users`).

Ένα πρόγραμμα που διαχειρίζεται βάσεις δεδομένων αποκαλείται *Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS, DataBase Management System)* και με την βοήθειά του μπορούμε να αποθηκεύσουμε, προσθέσουμε, τροποποιήσουμε, εμφανίσουμε ή και διαγράψουμε τα αποθηκευμένα δεδομένα.

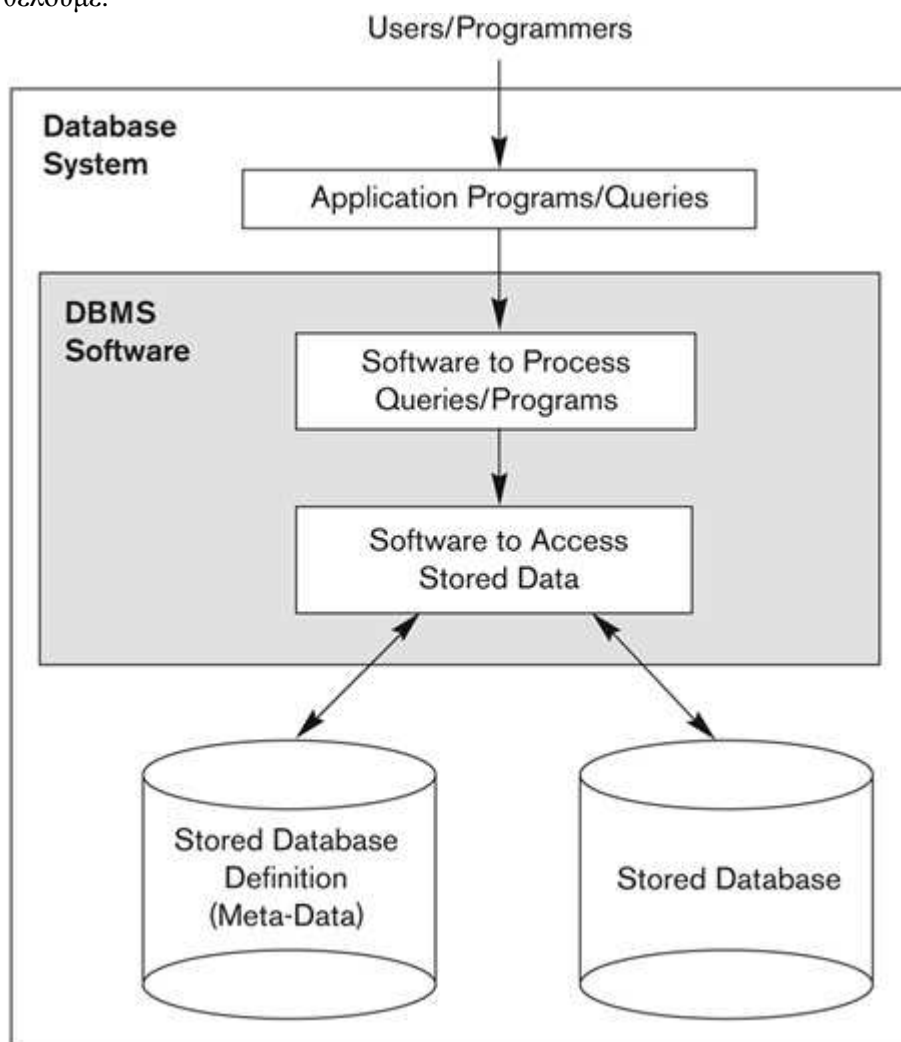
Τα δεδομένα που υπάρχουν στις βάσεις δεδομένων πρέπει να είναι :

- **Ολοκληρωμένα (Integrated)**, δηλ. τα δεδομένα πρέπει να είναι αποθηκευμένα σε ομοιόμορφα οργανωμένα σύνολα αρχείων όπου δεν πρέπει να υπάρχει επανάληψη ή πλεονασμός (redundancy) των ίδιων στοιχείων.
- **Καταμεριζόμενα (Shared)**, δηλ. να μπορούν περισσότεροι του ενός χρήστες να βλέπουν και να μοιράζονται τα ίδια δεδομένα την ίδια χρονική στιγμή.

### Τι Είναι το DBMS

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων (**DBMS, DataBase Management System**) είναι ένα σύνολο από προγράμματα που επιτρέπουν τον χειρισμό των δεδομένων μιας ή περισσοτέρων βάσεων δεδομένων που ανήκουν στο ίδιο σύστημα. Το DBMS περιέχει κάποια εργαλεία γενικής χρήσης για να μπορούμε να δημιουργούμε και να χειριζόμαστε τα δεδομένα.

Στα νεώτερα DBMS, όπως είναι η Oracle και η Informix, μπορούμε να έχουμε άμεση πληροφόρηση χωρίς να απαιτείται η παρουσία ενός προγραμματιστή. Τα δεδομένα ενός DBMS μπορούν να χρησιμοποιηθούν σε κάθε μορφής ερώτημα (query) για να πάρουμε ότι πληροφορίες θέλουμε.



Εικόνα 6. Σύστημα DBMS

### Τυπικές Λειτουργίες σε μια DBMS

- **A) Ορισμός βάσης:** τύποι δεδομένων (data types), δομές (structures), και περιορισμοί (constraints)
- **B) Δημιουργία ή Φόρτωση του αρχικού περιεχομένου της βάσης δεδομένων στη δευτερεύουσα μνήμη** (secondary storage medium)
- **Γ) Επεξεργασία της Βάσης:**
  - **Ανάκτηση:** Επερωτήσεις (Querying), Δημιουργία Αναφορών (Generating reports),...
  - **Τροποποίηση:** Εισαγωγές, Διαγραφές και Ενημερώσεις
  - **Ταυτόχρονη πρόσβαση** της βάσης από εφαρμογές, web, κτλ από πολλούς χρήστες
    - ... διασφαλίζοντας ότι τα δεδομένα είναι **ορθά (valid)** και **συνεπή (consistent)**
      - π.χ., δυο διαφορετικοί φοιτητές δεν μπορούν να έχουν τον ίδιο αριθμό ταυτότητας. Αυτός ο κανόνας πρέπει να ισχύει συνέχεια.
  - Μέτρα **Προστασίας** ή **Ασφάλειας** για περιορισμό μη-εξουσιοδοτημένων (unauthorized) προσβάσεων στα δεδομένα.
  - **Αναπαράσταση** (Representation) και **Γραφική Απεικόνιση** (Visualization) των δεδομένων
  - Βελτιστοποίηση Εκτέλεσης Ερωτημάτων, Χώρου Αποθήκευσης

### Βασικά Πλεονεκτήματα της Προσέγγισης των Βάσεων

- Αυτό-Περιγραφική Φύση μιας Βάσης:
  - Μια DBMS περιέχει ένα Κατάλογο(Catalog) ο οποίος αποθηκεύει την περιγραφή μιας βάσης (π.χ. , δομή πινάκων, τύποι δεδομένων, περιορισμοί, κτλ )
  - Η πληροφορία ονομάζεται meta-data.
  - Αυτό επιτρέπει την πρόσβαση στις πληροφορίες χωρίς να γνωρίζουμε τη δομή της αποθηκευμένης πληροφορίας.
- Ανεξαρτησία Προγράμματος – Δεδομένων (Program-data independence)
  - Επιτρέπει να αλλάξουμε την δομή και φυσική οργάνωση των δεδομένων χωρίς να χρειαστεί να αλλάξουμε τα Προγράμματα DBMS Application Programs / Queries)
- Παροχή Πολλαπλών Όψεων (Views) της ίδιας πληροφορίας.
  - Π.χ. η γραμματεία της πληροφορικής βλέπει μόνο όσες εγγραφές φοιτητών συσχετίζονται με τις εγγραφές των φοιτητών του Τμήματος Πληροφορικής.
- Διαμοιρασμός Δεδομένων (Μεταξύ Χρηστών) και Επεξεργασία Δοσοληπιών Πολλαπλών Χρηστών :
  - Επιτρέπει σε ταυτόχρονους χρήστες να ανακτούν και να ενημερώνουν την βάση.
  - Το υποσύστημα Ταυτοχρονίας της DBMS διασφαλίζει ότι τα δεδομένα θα παραμείνουν σε ορθή και συνεπή κατάσταση.
  - Το Υποσύστημα Ανάκαμψης διασφαλίζει ότι κάθε ολοκληρωμένη δοσοληπία (πρόγραμμα βάσης) θα καταγράψει μόνιμα τα αποτελέσματα στη βάση δεδομένων.

### Οι Στόχοι μιας Βάσης Δεδομένων

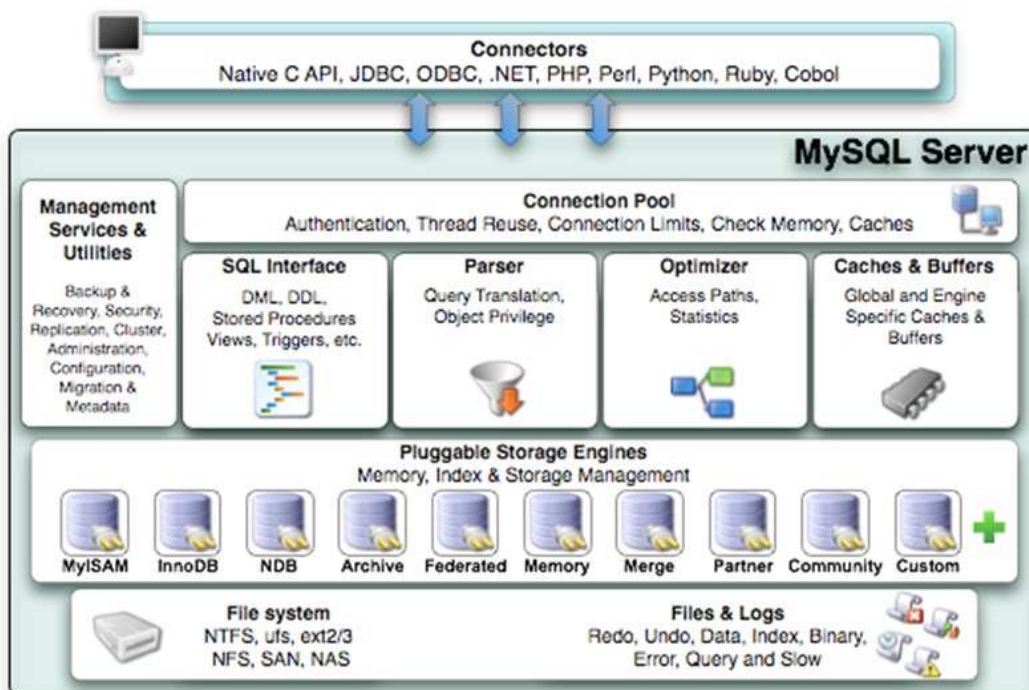
Οι στόχοι μιας βάσης δεδομένων είναι οι εξής :

- Ο περιορισμός της πολλαπλής αποθήκευσης των ίδιων στοιχείων (redundancy).
- Ο καταμερισμός (sharing) των ίδιων στοιχείων σ' όλους τους χρήστες.
- Η ομοιομορφία (uniformity) στον χειρισμό και την αναπαράσταση των δεδομένων.
- Η επιβολή κανόνων ασφαλείας (security).
- Η διατήρηση της ακεραιότητας (integrity) και της αξιοπιστίας (reliability) των δεδομένων.
- Η ανεξαρτησία των δεδομένων (data independence) και των προγραμμάτων από τον φυσικό τρόπο αποθήκευσης των δεδομένων.

## Τα Στοιχεία μιας Βάσης Δεδομένων

Τα δεδομένα μιας βάσης δεδομένων αποθηκεύονται (οργανώνονται) στις εξής στοιχειώδεις μορφές :

- **Πεδίο (Field)**, είναι το μικρότερο κομμάτι δεδομένων στο οποίο μπορούμε να αναφερθούμε και περιέχει ένα μόνο χαρακτηριστικό ή ιδιότητα ενός στοιχείου της βάσης δεδομένων.
- **Εγγραφή (Record)**, είναι ένα σύνολο από διαφορετικά πεδία που περιέχει όλες τις πληροφορίες για ένα στοιχείο της βάσης δεδομένων.
- **Αρχείο (File)**, είναι ένα σύνολο από πολλά παρόμοια στοιχεία (εγγραφές) της βάσης δεδομένων.
- **Πρωτεύον Κλειδί (Primary Key)**, είναι ένα πεδίο ή συνδυασμός πεδίων που χαρακτηρίζει μοναδικά μια εγγραφή.
- **Κλειδί (Key)**, είναι ένα πεδίο που δεν έχει κατ' ανάγκη μοναδική τιμή και που μπορούμε να το χρησιμοποιήσουμε για να κάνουμε αναζήτηση σ' ένα αρχείο.
- **Ξένο Κλειδί (Foreign Key)**, είναι ένα πεδίο που έχει το ίδιο σύνολο τιμών με το πρωτεύον κλειδί ενός άλλου αρχείου.



Εικόνα 7. Βάση δεδομένων και Connectors

## Τα Εργαλεία Χειρισμού Πληροφοριών

Τα εργαλεία χειρισμού πληροφοριών μιας βάσης δεδομένων είναι γνωστά και σαν "Γλώσσες Εντολών" και με τη βοήθειά τους μπορούμε να δώσουμε εντολές χειρισμού των δεδομένων. Η πιο γνωστή και ευρέως διαδεδομένη γλώσσα εντολών για τις σύγχρονες βάσεις δεδομένων είναι η *Δομημένη Γλώσσα Ερωτήσεων SQL (Structured Query Language)*, η οποία αποτελείται από τα εξής μέρη :

- **DDL (Data Definition Language, Γλώσσα Ορισμού Δεδομένων)**, με την οποία καθορίζουμε τις δομές και τα τμήματα μιας βάσης δεδομένων.
- **DML (Data Manipulation Language, Γλώσσα Χειρισμού Δεδομένων)**, με την οποία επεξεργαζόμαστε τα δεδομένα μιας βάσης δεδομένων.
- **DCL (Data Control Language, Γλώσσα Ελέγχου Δεδομένων)**, με την οποία εξασφαλίζουμε την ασφάλεια και την ακεραιότητα των δεδομένων μιας βάσης δεδομένων.

## Οι Γλώσσες 4ης Γενιάς (4GL)

Οι *Γλώσσες 4ης Γενιάς 4GL (4th Generation Languages)* είναι γλώσσες προγραμματισμού υψηλού επιπέδου, δηλ. έχουμε την δυνατότητα να κάνουμε πολλές και σύνθετες λειτουργίες με την βοήθειά τους και με λίγες μόνο εντολές.

Οι γλώσσες αυτές στηρίζουν την αποθήκευση των δεδομένων στην θεωρία των βάσεων δεδομένων και έτσι εξασφαλίζεται η ακεραιότητα των πληροφοριών ενώ παρέχουν τα βασικά εργαλεία αναζήτησης και χειρισμού των δεδομένων. Για την Oracle υπάρχει το *Accell* και για την Informix το *Informix-4GL*.

## Οι Ιεραρχικές Βάσεις Δεδομένων

Στις **Ιεραρχικές (Hierarchical)** βάσεις δεδομένων τα δεδομένα αναπαρίστανται με δενδρικής μορφής δομές δεδομένων και συνδέονται μεταξύ τους με συνδέσμους (links). Η κάθε εγγραφή μπορεί να συνδέεται προς τα πάνω μόνο με μία άλλη εγγραφή (γονέας), ενώ μπορεί να έχει έως δύο εγγραφές που να εξαρτώνται απ' αυτήν (παιδιά). Υπάρχει μία μόνο εγγραφή ρίζα (root), απ' την οποία εξαρτώνται όλες οι άλλες εγγραφές της βάσης δεδομένων. Έχουν το μειονέκτημα ότι είναι πολύπλοκες στην επεξεργασία των εγγραφών τους (προσθήκη, διαγραφή, τροποποίηση).

## Οι Δικτυωτές Βάσεις Δεδομένων

Στις **Δικτυωτές (Network)** βάσεις δεδομένων τα δεδομένα αναπαρίστανται με δενδρικής μορφής δομές δεδομένων και συνδέονται μεταξύ τους με συνδέσμους (links), όπως ακριβώς και στις ιεραρχικές βάσεις δεδομένων, με την διαφορά ότι μια εγγραφή μπορεί να συνδέεται προς τα πάνω με περισσότερες από μία πατρικές εγγραφές (parent records). Είναι λογικά πιο δύσχρηστες αλλά και πιο γρήγορες από τις ιεραρχικές βάσεις δεδομένων.

## Οι Σχεσιακές Βάσεις Δεδομένων

Στις **Σχεσιακές (Relational)** βάσεις δεδομένων, τα δεδομένα συνδέονται μεταξύ τους με **σχέσεις (relations)**, οι οποίες προκύπτουν από τα κοινά πεδία που υπάρχουν σε διαφορετικά αρχεία. Τα αρχεία αποκαλούνται **πίνακες (tables)**, οι εγγραφές **γραμμές (rows)** και τα πεδία **στήλες (columns)**. Η ύπαρξη μιας κοινής τιμής στα πεδία δύο αρχείων καθορίζει και μια σχέση μεταξύ των γραμμών διαφορετικών πινάκων.

Οι σχεσιακές βάσεις δεδομένων έχουν το πλεονέκτημα ότι είναι λογικά κατανοητές και πολύ ευέλικτες και δεκτικές σε αλλαγές.

### Τι Είναι ο DBA

Ο Διαχειριστής μιας Βάσης Δεδομένων (**DBA**, *DataBase Administrator*) είναι αυτός που έχει την ευθύνη για τον σωστό, αποδοτικό και αξιόπιστο τρόπο δημιουργίας και λειτουργίας μια βάσης δεδομένων. Οι αρμοδιότητές του είναι οι εξής :

- Η απόφαση για το είδος των πληροφοριών που πρέπει να αποθηκευθούν.
- Η απόφαση για τον τρόπο αποθήκευσης και πρόσβασης στις πληροφορίες αυτές.
- Η συνεργασία με τους τελικούς χρήστες.
- Η απόφαση για τον τρόπο εξασφάλισης των πληροφοριών.
- Η απόφαση για το κάθε πότε θα γίνονται αντίγραφα ασφαλείας (backup) των αρχείων.
- Η παρακολούθηση της σωστής λειτουργίας της βάσης δεδομένων και η απαιτούμενη προσαρμογή της.

### Εφαρμογές Βάσεων Δεδομένων

- Αεροπορικές κρατήσεις (expedia.com)
- Νοσοκομεία, Χημεία
- Τραπεζικές Εφαρμογές (Bank of Cyprus, Laiki, κτλ.)
- Λογιστήρια
- Σύστημα εγγραφής φοιτητών (banner)
- Κοινωνικές Ασφαλίσεις, Φόροι (taxisnet)
- Πωλήσεις – Αποθήκες (Amazon.com, eBay)
- Πιστωτικές κάρτες (JCC)
- Χρηματιστήριο
- Τηλεπικοινωνίες

### Τύποι Βάσεων Δεδομένων

- **Παραδοσιακές Εφαρμογές:**
  - **Αριθμητικές Βάσεις (Numerical) and Βάσεις Κειμένου (Text)**
  - Παράδειγμα **Sprint Telecom** (Αμερική): TOP-5 Database.
    - Καταγραφή στοιχείων συνδιάλεξης (π.χ., ο X τηλεφώνησε του Y για 10 λεπτά και 20 δευτερόλεπτα)
    - 2,85 τρισεκατομμύρια ( $\times 10^{12}$ ) εγγραφές!
    - 385 εκατομμύρια ( $\times 10^6$ ) νέες εγγραφές **την ημέρα!**
    - Σε εξαιρετικές περιπτώσεις μέχρι και **70,000** νέες εγγραφές το Δευτερόλεπτο.
- **Πρόσφατες Εφαρμογές:**
  - Βάσεις Πολυμέσων και Γεωγραφικών Πληροφ. Συστημάτων (GIS)
  - Αποθήκες Δεδομένων (Data Warehouses)
  - Βάσεις Ροών (Stream Databases), Βάσεις Νέφους (Cloud Databases), Βάσεις XML, κτλ

## 3.4 XML

### 3.4.1 Εισαγωγή στην XML

Η Επεκτάσιμη Γλώσσα Σήμανσης (Extensible Markup Language , XML) αποτελεί σήμερα την πιο πολλά υποσχόμενη γλώσσα αποθήκευσης και μετάδοσης πληροφοριών στον Παγκόσμιο Ιστό.

Αν και η Γλώσσα Σήμανσης Υπέρ-κειμένου (Hypertext Markup Language, HTML) είναι προς το παρόν η πιο διαδεδομένη γλώσσα στο χώρο της δημιουργίας ιστοσελίδων διαθέτει περιορισμένες δυνατότητες αποθήκευσης πληροφοριών. Αντίθετα, η XML διαθέτει εξαιρετικά ευέλικτο συντακτικό



με το οποίο μπορείτε ουσιαστικά να περιγράψετε οποιαδήποτε πληροφορία, από απλά δεδομένα μέχρι μια σύνθετη βάση δεδομένων (απ' όπου και ο όρος επεκτάσιμη). Εκτός αυτού ένα έγγραφο XML σε συνδυασμό με ένα φύλλο στυλ ή μια συμβατική σελίδα HTML μπορεί να απεικονιστεί εύκολα σε ένα φυλλομετρητή ιστού (Web Browser). Χάρη στην ικανότητα που έχει ένα έγγραφο XML να δομεί και να περιγράφει τόσο αποτελεσματικά τις πληροφορίες που περιέχει (μέσω ετικετών), ο φυλλομετρητής μπορεί να εντοπίζει, να εξάγει, να ταξινομεί, να φιλτράρει, να ταχτοποιεί, και να χειρίζεται με εξαιρετικά ευέλικτους τρόπους τις πληροφορίες αυτές.

Η XML προσφέρει την ιδανική λύση στο πρόβλημα χειρισμού της ταχύτητα αυξανόμενης ποσότητας και πολυπλοκότητας των πληροφοριών που χρειάζεται να δημοσιεύθουν στον Ιστό.

### Η ανάγκη για την XML

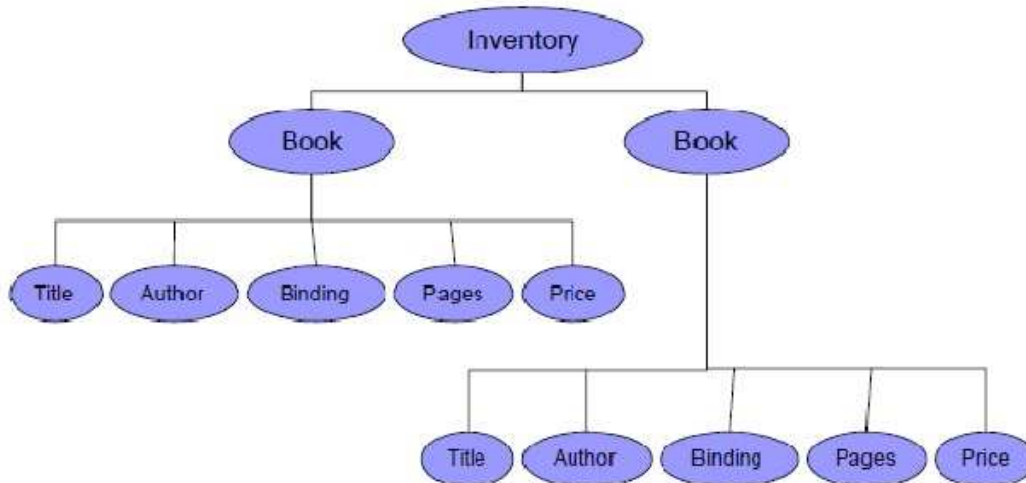
Η HTML παρέχει ένα σταθερό σύνολο προκαθορισμένων στοιχείων με τα οποία μπορείτε να συμβολίζετε τα περιεχόμενα μιας τυπικής ιστοσελίδας γενικού περιεχομένου. Παραδείγματα τέτοιων στοιχείων είναι οι επικεφαλίδες, οι ενότητες, οι λίστες, οι πινάκες, οι εικόνες και οι σύνδεσμοι. Αν και το σύνολο των προκαθορισμένων στοιχείων HTML έχει εμπλουτιστεί σημαντικά από την πρώτη έκδοση της HTML, η γλωσσά αυτή εξακολουθεί να είναι ακατάλληλη για τον προσδιορισμό πολλών ειδών εγγραφών. Ακολουθούν ορισμένα παραδείγματα εγγραφών που δεν μπορούν να προσδιοριστούν επαρκώς με την HTML:

- Έγγραφα που δεν αποτελούνται από τυπικά στοιχεία (επικεφαλίδες, ενότητες, λίστες, πινάκες, κτλ.). Για παράδειγμα, η HTML δεν έχει τα απαραίτητα στοιχεία που χρειάζονται για τη σήμανση μιας μουσικής παρτιτούρας ή ενός συνόλου μαθηματικών εξισώσεων.
- Βάσεις δεδομένων, όπως για παράδειγμα ένας κατάλογος βιβλίων. Μπορείτε να χρησιμοποιήσετε μια σελίδα HTML για να αποθηκεύσετε και να εμφανίσετε στατικές πληροφορίες (όπως μια λίστα με περιγραφές βιβλίων). Όμως, αν θέλετε να ταξινομήσετε, να επιλέξετε, να εντοπίσετε, και να δουλέψετε με τις πληροφορίες σας με κάποιο διαφορετικό τρόπο, για να κάθε πληροφορία (πεδίο) θα πρέπει να οριστεί και από μια ετικέτα (αυτό κάνουν και τα προγράμματα βάσεων δεδομένων όπως η Microsoft Access). Η HTML δε διαθέτει τα απαραίτητα στοιχεία για κάτι τέτοιο.
- Έγγραφα που θέλετε να οργανώσετε ιεραρχικά σε μορφή δένδρου. Ας υποθέσουμε, για παράδειγμα ότι γράφετε ένα βιβλίο και θέλετε να το χωρίσετε σε μέρη, κεφάλαια, ενότητες Α, ενότητες Β, ενότητες Γ, και ούτω καθεξής. Σε αυτή την περίπτωση, ένα πρόγραμμα θα μπορούσε να χρησιμοποιήσει αυτό το δομημένο έγγραφο για να δημιουργήσει έναν πίνακα περιεχομένων, να εμφανίσει τη διάθρωση του με διάφορα επίπεδα λεπτομερειών, να εξαγάγει κάποιες συγκεκριμένες ενότητες, και να διαχειριστεί με άλλους τρόπους τις πληροφορίες. Το στοιχείο επικεφαλίδας της HTML όμως επισημαίνει μόνο το κείμενο της επικεφαλίδας. Για παράδειγμα:

```
<H2>Web Site Contents</H2>
```

Το πραγματικό κείμενο και τα στοιχεία που ανήκουν σε μια ενότητα του εγγράφου δεν περιέχονται μέσα στο στοιχείο της αντίστοιχης επικεφαλίδας, γι' αυτό και τα στοιχεία αυτά δεν μπορούν να χρησιμοποιηθούν για να αποδώσουν με σαφήνεια την ιεραρχική δομή του εγγράφου.

# Ιεραρχική δόμηση εγγράφου XML



Εικόνα 8. Ιεραρχική δόμηση εγγράφου XML

## XML & HTML

Η XML δεν έχει σκοπό να αντικαταστήσει την HTML αλλά να την συμπληρώσει. Η HTML σχεδιάστηκε για να παρουσιάζει δεδομένα δίνοντας έμφαση στο πως αυτά φαίνονται. Η XML σχεδιάστηκε για να περιγράφει δεδομένα δίνοντας έμφαση στο τι είδος δεδομένα είναι.

### Ένα απλό XML αρχείο `example.xml`

```

<?xml version="1.0">

<messages>
  <message>Hello World! </message>
</messages>
    
```

### Περιγραφή λίστας βιβλίων

```

<INVENTORY>
  <BOOK>
    <TITLE>The Adventures of Huckleberry Finn</TITLE>
    <AUTHOR>Mark Twain</AUTHOR>
    <BINDING>mass market paperback</BINDING>
    <PAGES>298</PAGES>
    <PRICE>$5.49</PRICE>
  </BOOK>
</INVENTORY>
    
```

### Η λύση της XML

Ο ορισμός της XML αποτελείται από ένα στοιχειώδες μόνο συντακτικό. Όταν δημιουργείτε ένα έγγραφο XML, αντί να χρησιμοποιήσετε ένα περιορισμένο σύνολο προκαθορισμένων στοιχείων,



δημιουργείτε τα δικά σας στοιχεία και τους δίνετε όποιες ονομασίες θέλετε αυτή την έννοια έχει και ο όρος επεκτάσιμη (extensible) στην ονομασία της γλώσσας. Επομένως, μπορείτε να χρησιμοποιήσετε την XML για να περιγράψετε οποιαδήποτε είδος έγγραφου, από μουσικές παρτιτούρες μέχρι βάσεις δεδομένων .

### Δημιουργία εγγράφων XML

Επειδή η XML δεν περιέχει προκαθορισμένα στοιχεία, ίσως να δίνει την εντύπωση ενός σχετικά ανεπίσημου πρότυπου. Διαθέτει, όμως αυστηρά καθορισμένο συντακτικό. Για παράδειγμα σε αντίθεση με την HTML κάθε στοιχείο της XML πρέπει να έχει μια ετικέτα αρχής και μια ετικέτα τέλους. Επίσης, οποιαδήποτε ένθετο στοιχείο πρέπει να περιέχεται ολόκληρο στο στοιχείο που το περικλείει.

Για την ακρίβεια η ίδια η ευκολία να δημιουργείς τα δικά σου στοιχεία προϋποθέτει την τήρηση ενός αυστηρού συντακτικού. Γι αυτό και η προσαρμόσιμη φύση των εγγράφων XML απαιτεί προσαρμόσιμα προγράμματα για να χειρίζονται και να εμφανίζουν τις πληροφορίες που περιέχουν αυτά τα έγγραφα. Το αυστηρό συντακτικό της XML δίνει στα έγγραφα μια προβλέψιμη μορφή, κάνοντας έτσι πιο εύκολη τη γραφή αυτών των προγραμμάτων. Η εύκολη εφαρμογή είναι ένας από τους πρωταρχικούς στόχους αυτής της γλώσσας.

### Εμφάνιση εγγράφων XML

**Σύνδεση φύλλων στυλ.** Σύνδεση ενός φύλλου στυλ με το έγγραφο XML. Το φύλλο στυλ είναι ξεχωριστό αρχείο που περιέχει οδηγίες για τη μορφοποίηση στοιχείων της XML:

- Φύλλο επαλλήλων στυλ (cascading style sheet - CSS) που χρησιμοποιείται και για σελίδες HTML
- Φύλλο στυλ επεκτάσιμης γλώσσας (Extensible Style sheet Language, XSL) πιο ισχυρό από CSS, έχει σχεδιαστεί ειδικά για έγγραφα XML

**Δέσμευση δεδομένων** (Data binding) – Δημιουργία σελίδας HTML, σύνδεση με αυτή του εγγράφου XML και δέσμευση στη σελίδα προτύπων στοιχείων HTML, όπως το TABLE, με στοιχεία XML. Έτσι τα στοιχεία HTML θα εμφανίσουν αυτόματα τις πληροφορίες των στοιχείων XML με τα οποία είναι συνδεδεμένα.

**Γραφή σεναρίων.** (Scripting) – Δημιουργία σελίδας HTML, σύνδεση με αυτή του εγγράφου XML, προσπέλαση και εμφάνιση κάθε στοιχείου XML γράφοντας κώδικα σεναρίων (είτε σε JavaScript είτε σε Microsoft Visual Basic Scripting Edition Edition — VBScript). Ο φυλλομετρητής παρουσιάζει το έγγραφο XML ως μοντέλο αντικειμένου εγγράφου (Document Object Model, DOM), το οποίο παρέχει μεγάλο αριθμό αντικειμένων, ιδιοτήτων, και μεθόδων που ο κώδικας σεναρίων μπορεί να χρησιμοποιήσει για να προσπελάσει, να χειριστεί, και να εμφανίσει τα στοιχεία XML.

### SGML, HTML και XML

Η γλώσσα SGML, της οποίας τα αρχικά σημαίνουν Structured Generalized Markup Language, είναι η μητέρα όλων των γλωσσών σήμανσης. Η HTML και η XML έχουν προέλθει και οι 2 από την SGML. Η SGML ορίζει ένα βασικό συντακτικό, σας επιτρέπει όμως να δημιουργήσετε τα δικά σας στοιχεία (γι αυτό και ο όρος Generalized - γενικευμένη). Αν θέλετε να χρησιμοποιήσετε την SGML για να περιγράψετε κάποιο συγκεκριμένο έγγραφο, πρέπει να επινοήσετε ένα κατάλληλο σύνολο στοιχείων και μια δομή εγγράφου. Η HTML είναι μια εφαρμογή της SGML που δημιουργήθηκε το 1991 με σκοπό την περιγραφή ιστοσελίδων.

Η SGML μπορεί να φαίνεται η ιδανική επεκτάσιμη γλώσσα για την περιγραφή εγγράφων στον Ιστό. Ωστόσο τα μέλη του W3C που ασχολούνται με αυτά τα θέματα θεώρησαν την SGML πολύ σύνθετη και δύσχρηστη για την αποδοτική μεταφορά πληροφοριών στον Ιστό. Η ευελιξία και η πληθώρα των χαρακτηριστικών της SGML θα έκαναν δύσκολη τη συγγραφή του λογισμικού που χρειάζεται για την επεξεργασία και την εμφάνιση των πληροφοριών SGML στους φυλλομετρητές Ιστού. Χρειάζοταν ένα πιο αποδοτικό υποσύνολο της SGML σχεδιασμένο ειδικά για την μεταφορά πληροφοριών στον Ιστό. Το 1996 η αντίστοιχη ερευνητική ομάδα του W3C ανέπτυξε αυτό το υποσύνολο και το ονόμασε Extensible Markup Language (Επεκτάσιμη Γλώσσα Σήμανσης). Η XML σχεδιάστηκε με στόχο την ευκολία της υλοποίησης.

Η XML είναι μια απλοποιημένη έκδοση της SGML βελτιστοποιημένη για τον Ιστό. Όπως συμβαίνει και με την SGML, η XML σας επιτρέπει να ορίζετε τα δικά σας σύνολα στοιχείων όταν περιγράφετε κάποιο συγκεκριμένο έγγραφο. Επίσης όπως και με την SGML, μια εφαρμογή XML μπορεί να οριστεί από έναν ιδιώτη ή από έναν οργανισμό προτυποποίησης. Εφαρμογή XML είναι ένα σύνολο στοιχείων γενικής χρήσης και μια δομή εγγράφου που να μπορεί να χρησιμοποιηθεί για την περιγραφή εγγράφων συγκεκριμένου τύπου.

Το συντακτικό της XML προσφέρει λιγότερες επιλογές από εκείνο της SGML και έτσι είναι πιο εύκολο για τους χρήστες να διαβάζουν τα έγγραφα XML και για τους προγραμματιστές να γράφουν φυλλομετρητές, σενάρια, και ιστοσελίδες που να μπορούν να προσπελάζουν και να εμφανίζουν τις πληροφορίες του εγγράφου.

### Οι επίσημοι στόχοι της XML

1. Η XML θα είναι χρησιμοποιήσιμη στο Internet.

Η XML σχεδιάστηκε αρχικά για να αποθηκεύει και να μεταφέρει πληροφορίες στον Ιστό.

2. Η XML θα υποστηρίζει μεγάλη ποικιλία εφαρμογών.

Αν και ο αρχικός της σκοπός είναι η μεταφορά πληροφοριών στον Ιστό μέσω προγραμμάτων διακομιστών και φυλλομετρητών, η XML σχεδιάστηκε για να χρησιμοποιηθεί και από άλλα είδη προγραμμάτων. Για παράδειγμα χρησιμοποιείται ήδη για ανταλλαγή πληροφοριών ανάμεσα σε προγράμματα χρηματοπιστωτικού χαρακτήρα, για διανομή και αναβάθμιση λογισμικού και για συγγραφή φωνητικών σεναρίων (voice scripts) που μεταφέρονται μέσω τηλεφώνου.

3. Η XML θα είναι συμβατή με την SGML

Η XML είναι ένα υποσύνολο της SGML ειδικής χρήσης. Το θετικό που προκύπτει από αυτό το γεγονός είναι ότι τα εργαλεία λογιστικού της SGML μπορούν να προσαρμοστούν εύκολα για την XML.

4. Θα είναι εύκολη η συγγραφή προγραμμάτων που επεξεργάζονται έγγραφα XML.

Αν θέλουμε η XML να είναι μια πρακτική γλώσσα, τότε θα πρέπει να είναι εύκολη η δημιουργία φυλλομετρητών και άλλων προγραμμάτων που θα επεξεργάζονται έγγραφα XML. Στην πραγματικότητα, ο βασικός λόγος που δημιουργήθηκε το υποσύνολο της XML από την SGML ήταν η δυσκολία συγγραφής προγραμμάτων για την επεξεργασία εγγράφων της SGML.

5. Ο αριθμός των προαιρετικών χαρακτηριστικών στην XML θα διατηρηθεί στο κατώτατο (ιδανικά μηδενικό) επίπεδο.

Λόγω του ελάχιστου αριθμού προαιρετικών δυνατοτήτων της XML, γίνεται πιο εύκολη η συγγραφή προγραμμάτων που επεξεργάζονται έγγραφα XML. Η πληθώρα προαιρετικών δυνατοτήτων της SGML ήταν ο βασικός στόχος που θεωρήθηκε μη πρακτική για τον ορισμό εγγράφων του Ιστού. Τα προαιρετικά χαρακτηριστικά της περιλαμβάνουν επαναπροσδιορισμό των χαρακτήρων οριοθέτησης στις ετικέτες (κανονικά αυτοί είναι οι

χαρακτήρες <και>) και απαλοιφή της ετικέτας τέλους στην περίπτωση που ο επεξεργαστής μπορεί να καταλάβει σε ποιο σημείο τελειώνει ένα στοιχείο. Ένα στιβαρό πρόγραμμα επεξεργασίας εγγράφων SGML θα έπρεπε να μπορεί να χειριστεί όλα τα προαιρετικά χαρακτηριστικά, ακόμα και εκείνα που χρησιμοποιούνται σπάνια.

6. Τα έγγραφα της XML πρέπει να είναι αναγνώσιμα από τον άνθρωπο και σαφή.

Η XML σχεδιάστηκε για να αποτελεί μια κοινή γλώσσα με σκοπό την ανταλλαγή πληροφοριών ανάμεσα σε χρήστες και προγράμματα σε όλο τον κόσμο. Η δυνατότητα ανάγνωσης της XML διευκολύνει αυτόν τον στόχο δίνοντας την δυνατότητα στους ανθρώπους αλλά και σε εξειδικευμένα προγράμματα λογισμικού να γράφουν και να διαβάζουν έγγραφα XML. Αυτό είναι το χαρακτηριστικό που διαχωρίζει την XML από τις περισσότερες αποκλειστικές μορφές που χρησιμοποιούνται για τις βάσεις δεδομένων και τα έγγραφα επεξεργασίας κειμένου.

Τα έγγραφα XML μπορούν να διαβαστούν εύκολα από άνθρωπο επειδή είναι γραμμένα σε απλή μορφή και έχουν λογική δομή δένδρου. Μπορείτε να βελτιώσετε την αναγνωσιμότητα της XML επιλέγοντας λογικά ονόματα για τα στοιχεία, τις ιδιότητες και τις ενότητες των εγγράφων σας και προσθέτοντας χρήσιμα σχόλια.

7. Ο σχεδιασμός της XML θα πρέπει να ολοκληρωθεί σύντομα.

Η XML θα αποτελεί βιώσιμο πρότυπο μόνο αν η κοινότητα των προγραμματιστών και των χρηστών το υιοθετήσει. Για το λόγο αυτό έπρεπε να ολοκληρωθεί πριν η κοινότητα άρχιζε να υιοθετεί άλλα εναλλακτικά πρότυπα, τα οποία οι εταιρείες παραγωγής λογισμικού τείνουν να παράγουν με πολύ γρήγορους ρυθμούς.

8. Ο σχεδιασμός της XML θα πρέπει να είναι τυπικός και περιεκτικός.

Η προδιαγραφή της XML είναι γραμμένη σε μια τυπική γλώσσα που χρησιμοποιείται για τον ορισμό των γλωσσών προγραμματισμού και είναι γνωστή ως επεκτεταμένη σημειογραφία Backus-Naur (Extended Backus-Naur Form, EBNF). Η τυπική αυτή γλώσσα, αν και είναι δύσκολο να διαβαστεί από οποιονδήποτε αποκαθιστά της ασάφειες και τελικά διευκολύνει τη συγγραφή εγγράφων XML και ειδικά τη δημιουργία λογισμικού για την επεξεργασία εγγράφων XML, ενθαρρύνοντας έτσι τη χρήση της XML.

9. Τα έγγραφα XML θα δημιουργούνται εύκολα.

Αν θέλουμε η XML να είναι μια πρακτική γλώσσα σήμανσης για τα έγγραφα του Ιστού, τότε πρέπει όχι μόνο να είναι εύκολη η γραφή προγραμμάτων επεξεργασίας XML, αλλά πρέπει επίσης να είναι εύκολη και η δημιουργία των ίδιων των εγγράφων.

10. Στη γλώσσα XML έχει ελάχιστη σημασία η λακωνικότητα.

Όπως αναφέρεται και στο στόχο 6 (Τα έγγραφα της XML πρέπει να είναι αναγνώσιμα από τον άνθρωπο και σαφή), η XML δεν πρέπει να είναι υπερβολικά λακωνική ώστε να γίνεται δυσνόητη.

## Τυπικές εφαρμογές XML

Μια εφαρμογή XML κανονικά ορίζεται με τη δημιουργία ενός ορισμού τύπου εγγράφου (document type definition ,DTD), το οποίο αποτελεί προαιρετικό στοιχείο ενός εγγράφου XML. Το DTD μοιάζει με σχήμα βάσης δεδομένων (database schema): ορίζει και δίνει ονόματα στα στοιχεία που μπορούν να χρησιμοποιηθούν στο έγγραφο, τη σειρά εμφάνισής τους, τις ιδιότητές τους που μπορούν να χρησιμοποιηθούν, καθώς και άλλα χαρακτηριστικά του εγγράφου. Τα πλεονεκτήματα που προκύπτουν από τη χρήση μιας τυπικής εφαρμογής XML για την ανάπτυξη των εγγράφων είναι το γεγονός ότι μπορούμε να μοιράζεστε τα έγγραφα σας με όλους τους υπόλοιπους χρήστες της ίδιας

εφαρμογής, καθώς και ότι μπορείτε να επεξεργάζεστε και να εμφανίζετε τα έγγραφα σας χρησιμοποιώντας λογισμικό που έχει ήδη δημιουργηθεί για αυτήν την συγκεκριμένη εφαρμογή.

### **Εφαρμογές XML για τη βελτίωση εγγράφων XML**

Εκτός από τις εφαρμογές XML που χρησιμοποιούνται για την περιγραφή ειδικών τάξεων εγγράφων, έχουν οριστεί και αρκετές άλλες εφαρμογές τις οποίες μπορείτε να χρησιμοποιήσετε σε οποιαδήποτε είδος εγγράφου. Οι εφαρμογές αυτές διευκολύνουν τη δημιουργία των εγγράφων και σας επιτρέπουν να επιφέρετε βελτιώσεις σε αυτά. Ακολουθούν ορισμένα παραδείγματα :

- **Επεκτάσιμη Γλώσσα Φύλλων Στυλ (Extensible Stylesheet Language — XSL).** Δημιουργία ισχυρών φύλλων στυλ χρησιμοποιώντας το συντακτικό της XML
- **Μέθοδος XML (XML Schema) .** Λεπτομερή σχήματα (schemas) για τα έγγραφα χρησιμοποιώντας το καθιερωμένο συντακτικό της XML.
- **Γλώσσα Σύνδεσης XML (XML Linking Language - Xlink).** Σύνδεση εγγράφων και χρήση πολλαπλών προορισμών συνδέσμων
- **Γλώσσα Δεικτών XML (XML Pointer Language - Xpointer).** Ορισμός ευέλικτων προορισμών σύνδεσης.

Η XML δεν αποτελεί απλώς ένα εξαιρετικό χρήσιμο εργαλείο για τον ορισμό εγγράφων, αλλά λειτουργεί και ως πλαίσιο για την δημιουργία των εφαρμογών και των βελτιώσεων της XML που γίνονται απαραίτητες καθώς το Internet εξελίσσεται συνεχώς.

### **Πρακτικές χρήσεις XML**

- **Αποθήκευση βάσεων δεδομένων.** Όπως συμβαίνει και με τις αποκλειστικές μορφές βάσεων δεδομένων, η XML μπορεί να χρησιμοποιηθεί για να τοποθετήσει ετικέτες σήμανσης σε οποιαδήποτε πεδίο πληροφοριών μέσα σε οποιαδήποτε βάση δεδομένων. (Για παράδειγμα θα μπορούσε να ορίσει ετικέτες για κάθε όνομα, διεύθυνση, και αριθμό τηλεφώνου στις εγγραφές μιας βάσης δεδομένων με διευθύνσεις.) Με τον τρόπο αυτόν σας δίνει την ευκαιρία να εμφανίζετε τα δεδομένα καθώς επίσης και να αναζητάτε συγκεκριμένα δεδομένα, να ταξινομείτε, να τα φιλτράρετε και να τα επεξεργάζεστε με πολλούς και ποικίλους τρόπους.
- **Δόμηση εγγράφων.** Η δενδροειδής δομή των εγγράφων XML κάνει τη γλώσσα αυτή ιδανική για την περιγραφή της δομής εγγράφων όπως μυθιστορήματα, βιβλία γενικού ενδιαφέροντος και θεατρικά έργα. Κάτι τέτοιο επιτρέπει στο λογισμικό να εμφανίζει ή να τυπώνει το έγγραφο με την κατάλληλη μορφοποίηση, να εντοπίζει, να εξάγει, ή να χειρίζεται τις πληροφορίες του εγγράφου, να δημιουργεί αυτόματα πίνακες περιεχομένων, διαρθρώσεις, και περιλήψεις και γενικά να χειρίζεται τις πληροφορίες σας με διάφορους τρόπους.
- **Αποθήκευση διανυσματικών γραφικών.** (VML ή Vector Markup Language — Γλώσσα Σήμανσης Διανυσμάτων.)
- **Περιγραφή παρουσιάσεων με πολυμέσα.** (SMIL ή Synchronized Multimedia Integration Language και HTML+TIME, ή HTML Timed Interactive Multimedia Extensions).
- **Ορισμός καναλιών.** Τα κανάλια είναι ιστοσελίδες που προωθούνται (στέλνονται αυτόματα) σε συνδρομητές. (CDF ή Channel Definition Format)
- **Περιγραφή πακέτων λογισμικού και των αλληλεξαρτήσεών τους.** Αυτές οι περιγραφές επιτρέπουν τη διανομή και την αναβάθμιση λογισμικού μέσω δικτύων. (OSD ή Open Software Description)

- **Επικοινωνία μεταξύ εφαρμογών στον Ιστό με ανοιχτό και επεκτάσιμο τρόπο, χρησιμοποιώντας τρόπο μηνύματα που βασίζονται στην XML.** Τα μηνύματα είναι ανεξάρτητα από τα λειτουργικά συστήματα, τα μοντέλα αντικειμένων, και τις γλώσσες προγραμματισμού που χρησιμοποιούνται. (SOAP ή Simple Object Access Protocol)
- **Αποστολή ηλεκτρονικών επαγγελματικών καρτών μέσω ηλεκτρονικού ταχυδρομείου.**
- **Ανταλλαγή χρηματοπιστωτικών πληροφοριών .** Οι πληροφορίες αυτές ανταλλάσσονται με έναν ανοιχτό αναγνωρίσιμο τρόπο ανάμεσα σε χρηματοπιστωτικά προγράμματα (όπως το Quicken και το Microsoft Money ) και οργανισμούς (όπως τράπεζες και αμοιβαία κεφάλαια). (OFX ή Open Financial Exchange)
- **Δημιουργία, διαχείριση και χρήση πολύπλοκων ψηφιακών φορμών για τις εμπορικές συναλλαγές μέσω Internet.** Οι φόρμες μπορεί να περιλαμβάνουν ψηφιακές υπογραφές που τις κάνουν νομικά δεσμευτικές. (XFDL ή Extensible Forms Description Language)
- **Ανταλλαγή περιγραφών θέσεων εργασίας και βιογραφικών σημειωμάτων.** (HRMML ή Human Resource Management Markup Language)
- **Μορφοποίηση μαθηματικών τύπων και επιστημονικών πληροφοριών στον Ιστό.** (MathML ή Mathematical Markup Language)
- **Περιγραφή μοριακών δομών.** (CML ή Chemical Markup Language)
- **Κωδικοποίηση και εμφάνιση πληροφοριών σχετικά με ακολουθίες DNA, RNA, και πρωτεϊνών.** (BSML ή Bioinformatic Sequence Markup Language)
- **Κωδικοποίηση γενεαλογικών στοιχείων.** (GedML ή Genealogical Data Mark-up Language)
- **Ανταλλαγή πληροφοριών σχετικά με την αστρονομία.** (AML ή Astronomical Markup Language)
- **Συγγραφή μουσικής παρτιτούρας .** (MusicML ή Music Markup Language)
- **Αποθήκευση φωνητικών σεναρίων για μεταφορά τους μέσω τηλεφώνου.** Φωνητικά σενάρια μπορούν για παράδειγμα να χρησιμοποιηθούν για την εκκώνηση οδηγιών φωνητικού ταχυδρομείου, αξιών μετοχών, και δελτίων καιρού.
- **Αποθήκευση πληροφοριών εντοπισμού για υπηρεσίες courier.** Η Federal Express για παράδειγμα χρησιμοποιεί την XML για αυτόν τον σκοπό.
- **Υποβολή αγγελιών σε εφημερίδες σε ψηφιακή μορφή.** (Ad Markup)
- **Αρχειοθέτηση νομικών εγγράφων και ανταλλαγή νομικών πληροφοριών ηλεκτρονικά.** (XCL ή XML Court Interface)
- **Κωδικοποίηση δελτίων καιρού.** (OMF ή Weather Observation Markup Format)
- **Ανταλλαγή πληροφοριών σχετικά με συναλλαγές ακίνητης περιουσίας.** (RETS ή Real Estate Transaction Standard)
- **Ανταλλαγή πληροφοριών σχετικά με ασφάλειες.**

- **Ανταλλαγή ειδήσεων και πληροφοριών με τη χρήση ανοικτών προτύπων Ιστού.** (XMLNews)
- **Αναπαράσταση πληροφοριών θεολογικού περιεχομένου και σήμανση εκκλησιαστικών κειμένων.** (ThML Theological Markup Language LitML Liturgical Markup Language)

### 3.4.2 Well-formed και XML DOM

#### Σωστά διατυπωμένο έγγραφο Well-formed xml

Ένα σωστά διατυπωμένο έγγραφο Well-formed XML πρέπει να πληρεί τις απαραίτητες προϋποθέσεις. Πρέπει:

- Όλα τα μη κενά elements πρέπει να περιλαμβάνουν ένα tag ανοίγματος (opening tag) και ένα tag κλεισίματος (closing tag)

`<surname>Webber</surname>`

- Τα elements που δεν έχουν τιμή (empty elements ή self-closing elements) έχουν δύο εναλλακτικές γραφές

π.χ. `<surname/>` ή `<surname></surname>`

- Οι τιμές των attributes μπαίνουν σε μονά (') ή συνηθέστερα διπλά εισαγωγικά (")

π.χ. `<surname order="first">`

- Η XML πρέπει να περιλαμβάνει τουλάχιστον ένα element που ονομάζεται root element ή document element . Όλα τα elements είναι φωλιασμένα (nested) εκτός από το root element. Απαγορεύεται η επικάλυψη των elements

- Γίνεται διάκριση ανάμεσα στα κεφαλαία και τα μικρά των ονομάτων, π.χ

`<surname>`

είναι διαφορετικό tag από το

`<Surname>`

- Το xml declaration είναι προαιρετικό, όταν όμως υπάρχει θα πρέπει να τηρείται σε όλη την xml η κωδικοποίηση χαρακτήρων που δηλώνεται σε αυτό.

#### XML DOM (Document Object Model)

Η XML ως δενδρική δομή

- Κάθε xml έγγραφο (document) ενέχει την ιεραρχική σχέση πατέρα (parent node) παιδιών (child nodes), ξεκινώντας από το root element που είναι ο πατέρας όλων.
- Κάθε xml έγγραφο μπορεί να παρουσιαστεί ως μία δενδρική δομή

XML DOM (Document Object Model)

- Το DOM μοντέλο ορίζει έναν W3C (World Wide Web Consortium) στάνταρντ τρόπο προσπέλασης και δυναμικής διαχείρισης ενός xml εγγράφου (document).



- Το DOM μοντέλο παρουσιάζει ένα έγγραφο ως δενδρική δομή αποτελούμενη από κόμβους (nodes) όπου κάθε κόμβος μπορεί να είναι ένα element, ένα attribute, ένα σχόλιο ή απλό κείμενο.
- Ολόκληρο το xml έγγραφο αποτελεί έναν document κόμβο (node).
- Κάθε document έχει ένα root element από το οποίο αρχίζει και το οποίο είναι ουσιαστικά η κορυφή της δενδρικής (ιεραρχικής) δομής.
- Όλα τα nodes, εκτός από το root element, έχουν έναν πατέρα (parent node).
- Ένα node, εκτός από τα φύλλα του δένδρου, μπορεί να έχει έναν οποιονδήποτε αριθμό παιδιών (child nodes)
- Τα text nodes δεν έχουν παιδιά.
- Τα nodes που βρίσκονται στο ίδιο επίπεδο ονομάζονται siblings.

### 3.4.3 Προσπέλαση αρχείων XML

#### 3.4.3.1 XML Parser

Για να μπορέσουμε να προσπελάσουμε και να διαχειριστούμε κόμβους (nodes) ενός xml document θα πρέπει να χρησιμοποιήσουμε έναν parser. Οι οποίοι διαβάζουν ένα XML αρχείο, ψάχνουν για συντακτικούς περιορισμούς, και κάνουν τα δεδομένα διαθέσιμα σε μια εφαρμογή.

Υπάρχουν δύο γενικές κατηγορίες xml parser:

- **SAX** (Simple API for XML parser). Είναι ένας μονής κατεύθυνσης (one pass) σειριακός parser ο οποίος εκτελεί ενέργειες ανάλογα με τα γεγονότα (events) που συναντάει. Τέτοια events μπορεί να είναι ένα element node ή ένα text node. Έχει σειριακή πρόσβαση στο αρχείο και είναι ιδανικό για λίστες, πίνακες στοιχείων και απλά μοντέλα δεδομένων.
- **DOM parser**. Είναι ένα προγραμματιστικό εργαλείο για την επεξεργασία XML εγγράφων. Καθορίζει τους τρόπους πρόσβασης και επεξεργασίας ενός εγγράφου. Χτίζει στην μνήμη την αντίστοιχη δενδρική δομή όλου του xml document τους κόμβους του οποίου μπορεί να προσπελάσει προς κάθε δυνατή κατεύθυνση.

Διαφορές :

- Οι **SAX parsers** είναι γρήγοροι και με πολύ λιγότερες απαιτήσεις σε μνήμη επιτρέποντας την επεξεργασία πολύ μεγάλων xml documents. Αλλά όμως πρέπει να κάνεις πολλά μόνος σου και δεν είναι πολύ χρήσιμο αν χρειάζεται να μεταβάλεις δυναμικά το κείμενο οπότε χρειάζεται να είναι φορτωμένο όλο στην μνήμη.
- Οι **DOM parsers** είναι απλούστεροι απαιτώντας λιγότερες κλήσεις μεθόδων για την προσπέλαση και την διαχείριση των κόμβων του DOM tree και ιδανικοί όταν η δενδρική δομή πρέπει να διαπεραστεί πολλές φορές και σε διαφορετικές κατευθύνσεις. Όμως μπορεί να γίνει αργό (χρειάζεται να παράγει το δέντρο), και μπορεί να χρειάζεται πολύ μνήμη και το DOM interface στο προγραμματισμό είναι δύσχρηστο.

### 3.4.3.2 Τι είναι SimpleXML

Με την έκδοση PHP4 ήταν πιο δύσκολο και χρειαζόταν εξωτερικές βιβλιοθήκες για να μπορέσουμε να προσπελάσουμε, να αναλύσουμε, και να κάνουμε αλλαγές σε αρχεία XML. Όμως με την PHP5 ήρθε μια σειρά από βιβλιοθήκες που βρίσκονται στον πυρήνα και μια είναι η SimpleXML η οποία μας βοηθάει με εύκολο τρόπο και με λίγες γραμμές κώδικα να διαβάζει δεδομένα από το κείμενο ενός στοιχείου. Μπορεί να πάρει τα χαρακτηριστικά και το κείμενο ενός στοιχείου εάν γνωρίζουμε την διάταξη του εγγράφου XML.

Η SimpleXML μετατρέπει το έγγραφο XML σε ένα αντικείμενο, με τον τρόπο αυτό:

- Στοιχεία(Elements) - μετατρέπονται σε απλές ιδιότητες του αντικειμένου SimpleXMLElement. Όταν υπάρχουν περισσότερα από ένα στοιχείο σε ένα επίπεδο, όπου τοποθετούνται μέσα σε ένα πίνακα(array).
- Χαρακτηριστικά(Attributes) - έχετε πρόσβαση χρησιμοποιώντας συνειρμικούς πίνακες(associative arrays), όπου ο δείκτης αντιστοιχεί στο όνομα του χαρακτηριστικού.
- Δεδομένα (Element Data)- τα δεδομένα στοιχεία από το κείμενο μετατρέπονται σε strings. Αν ένα στοιχείο έχει περισσότερους από έναν κόμβο κειμένου, θα πρέπει να τοποθετούνται με τη σειρά που βρίσκονται

Η SimpleXML είναι γρήγορη και εύκολη στη χρήση κατά την εκτέλεση βασικών εργασιών, όπως:

- Ανάγνωση αρχείων XML
- Εξαγωγή δεδομένων από XML strings
- Επεξεργασία κειμένου κόμβους ή ιδιότητες

#### Παράδειγμα χρήσης SimpleXML

Παρακάτω είναι ένα αρχείο XML:

test.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

Εδώ πρέπει να κάνουμε τα εξής:

- Να φορτώσουμε το αρχείο XML.
- Να πάρουμε το όνομα του πρώτου στοιχείου.
- Να δημιουργήσουμε ένα βρόγχο(loop) ο οποίος θα περνάει από κάθε κόμβο παιδιού, χρησιμοποιώντας την συνάρτηση children().
- Στην έξοδο θα έχουμε το όνομα του στοιχείου και τα δεδομένα για κάθε κόμβο παιδί.

#### Κώδικας PHP

```
<?php
$xml = simplexml_load_file("test.xml");

echo $xml->getName() . "<br />";

foreach($xml->children() as $child)
```

```
{  
echo $schild->getName() . " : " . $schild . "<br />";  
}  
?>
```

Η έξοδος του κώδικα θα είναι:

```
note  
to: Tove  
from: Jani  
heading: Reminder  
body: Don't forget me this weekend!
```

### 3.5 Τι είναι ο Apache Web Server

Ο Apache Web Server είναι αυτό ακριβώς που δηλώνει το όνομά του. Πρόκειται δηλαδή για έναν εξυπηρετητή (server) του παγκόσμιου Ιστού (Web). Με τον όρο server το μυαλό μας πηγαίνει ίσως σε ηλεκτρονικούς υπολογιστές που φιλοξενούν ιστοσελίδες και όχι άδικα. Ο όρος αυτός χρησιμοποιείται και για το μηχανήμα εξυπηρετητή (hardware) αλλά και για το πρόγραμμα (software). Στο άρθρο αυτό θα ασχοληθούμε μόνο με το software και συγκεκριμένα με τον Apache.

Ο Apache εγκαθίσταται σε έναν υπολογιστή ο οποίος μπορεί να χρησιμοποιεί διάφορα λειτουργικά συστήματα όπως Linux, Unix, Microsoft Windows, GNU, FreeBSD, Solaris, Novell NetWare, Mac OS X, OS/2, TPF. Ο ρόλος του Apache είναι να αναμένει αιτήσεις από διάφορα προγράμματα – χρήστες (clients) όπως είναι ένας ο φυλλομετρητής (browser) ενός χρήστη και στη συνέχεια να εξυπηρετεί αυτές τις αιτήσεις “σερβίροντας” τις σελίδες που ζητούν είτε απευθείας μέσω μιας ηλεκτρονικής διεύθυνσης (URL), είτε μέσω ενός συνδέσμου (link). Ο τρόπος με τον οποίο ο Apache εξυπηρετεί αυτές τις αιτήσεις, είναι σύμφωνος με τα πρότυπα που ορίζει το πρωτόκολλο HTTP (Hypertext Transfer Protocol).

#### Το ξεκίνημα και η εξέλιξη του Apache

Η περίοδος έναρξης της δημιουργίας του προγράμματος χρονολογείται στις αρχές του 1990, όταν άρχισε να αναπτύσσεται από τον Robert McCool, ως ένα project του National Center for Supercomputing Applications (NCSA) με το όνομα HTTPd (HTTP daemon). Το 1994 ο Robert McCool αποχώρησε από το NCSA με αποτέλεσμα το NCSA HTTPd να μείνει σχεδόν εγκαταλειμμένο, πέρα από κάποιες διορθώσεις (patches) που ανέπτυσαν και διένειμαν εκτός από τον McCool και άλλοι προγραμματιστές. Το 1995 ανέλαβε το πρόγραμμα το Ίδρυμα Λογισμικού Apache (Apache Software Foundation), το οποίο διατηρεί την εποπτεία του έως και σήμερα.

#### Ο Apache σήμερα

Ο Apache HTTP αναπτύσσεται από την “Κοινότητα Ανοιχτού Λογισμικού” και η εποπτεία, υποστήριξη, και διάθεση του προγράμματος γίνεται από το Apache Software Foundation. Το πρόγραμμα είναι ανοιχτού κώδικα (open source), κάτι που σημαίνει ότι σύμφωνα με την άδεια χρήσης του (license), διατίθεται δωρεάν και μπορούν να γίνουν ελεύθερα από το χρήστη προσθήκες και τροποποιήσεις στον κώδικα του.

## Η προέλευση του ονόματος του

Υπάρχουν δύο εκδοχές σχετικά με την προέλευση του ονόματος του. Η πρώτη εντοπίζεται στα πρώτα χρόνια της δημιουργίας του, τότε που ως NCSA HTTPd έπρεπε να αναπτύσσονται συνεχώς διορθώσεις (patches) για να ενσωματωθούν στον αρχικό του κώδικα με αποτέλεσμα να του δοθεί το όνομα a patchy server. Η δεύτερη εκδοχή σύμφωνα με το ίδρυμα Apache, αναφέρει ότι το όνομα αυτό δόθηκε προς τιμήν των ιθαγενών Ινδιάνων της Αμερικής και συμβολίζει το μαχητικό πνεύμα και την αντοχή.

## Χαρακτηριστικά και λειτουργίες του Apache HTTP

Ο Apache διαθέτει ποικιλία χαρακτηριστικών και μπορεί να υποστηρίξει μια μεγάλη γκάμα εφαρμογών με τις οποίες και συνεργάζεται. Οι δυνατότητες του προγράμματος αυτού καθαυτού και τα χαρακτηριστικά του δεν είναι και τόσο πολλά. Ένα από τα βασικότερα χαρακτηριστικά του όμως, το οποίο και του δίνει μεγάλες δυνατότητες, είναι ότι μπορεί να προσαρμόσει επάνω του πολλές προσθήκες προγραμμάτων (modules), τα οποία με τη σειρά τους παρέχουν διαφορετικές λειτουργίες. Μερικά από τα πιο γνωστά modules του Apache HTTP είναι τα modules πιστοποίησης, όπως για παράδειγμα τα mod\_access, mod\_auth, mod\_digest κ.λπ. Παρέχει επίσης SSL σε TLS μέσω των (mod\_ssl), και proxy module (mod\_proxy), πραγματοποιεί ανακατευθύνσεις διευθύνσεων (URL rewrites) μέσω του mod\_rewrite, καταγραφές συνδέσεων μέσω του mod\_log\_config, συμπίεση αρχείων μέσω του mod\_gzip και πολλά άλλα modules τα οποία διατίθενται είτε απ'ο το Apache Software Foundation, είτε από τρίτες εταιρίες λογισμικού.

Ένα άλλο χαρακτηριστικό – δυνατότητα του Apache HTTP, όπως έχω αναφέρω πιο πάνω, είναι ότι μπορεί να εγκατασταθεί σε διάφορα λειτουργικά συστήματα. Ο Apache HTTP υποστηρίζει επίσης αρκετές διάσημες εφαρμογές και γλώσσες προγραμματισμού όπως MySQL, PHP, Perl, Python κ.λπ.

Αυτά είναι μερικά από τα χαρακτηριστικά και τις λειτουργίες του που κάνουν τον Apache τον πιο δημοφιλή Web Server από το 1996 έως τις μέρες μας. Περισσότερο από το 50% των ιστοχώρων του παγκόσμιου ιστού, χρησιμοποιεί τον Apache ως εξυπηρετητή. Το υπόλοιπο ποσοστό καλύπτουν αντίστοιχα προγράμματα, όπως το Microsoft Internet Information Services (IIS), ο Sun Java System Web Server, ο Zeus Web Server κ.α.

## 3.6 Τι είναι XAMPP

Το XAMPP είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα και ανεξάρτητου πλατφόρμας το οποίο περιέχει το εξυπηρετητή ιστοσελίδων http Apache, την βάση ιστοσελίδων MySQL και ένα διεργασμένο για σενάρια γραμμένα σε γλώσσες προγραμματισμού PHP και Perl.

Το XAMPP είναι ακρωνύμιο και αναφέρεται στα παρακάτω αρχικά:

- X (αναφέρεται στο "cross-platform" που σημαίνει λογισμικό ανεξάρτητο πλατφόρμας)
- Apache HTTP εξυπηρετητής
- MySQL
- PHP
- Perl

Το XAMPP είναι ένα ελεύθερο λογισμικό το οποίο περιέχει ένα εξυπηρετητή ιστοσελίδων το οποίο μπορεί να εξυπηρετεί και δυναμικές ιστοσελίδες τεχνολογίας PHP/MySQL. Είναι ανεξάρτητο πλατφόρμας και τρέχει σε Microsoft Windows, Linux, Solaris, and Mac OS X και χρησιμοποιείται ως

πλατφόρμα για την σχεδίαση και ανάπτυξη ιστοσελίδων με την τεχνολογίες όπως PHP, JSP και Servlets.

### **Δυνατότητες και απαιτήσεις**

Το XAMPP προϋποθέτει μόνο τα λογισμικά συμπίεσης αρχείων zip, tar, 7z ή exe κατά την διάρκεια της εγκατάστασης. Το XAMPP έχει δυνατότητα αναβάθμισης σε νέες εκδόσεις του εξυπηρετητή ιστοσελίδων http Apache, της βάσης δεδομένων MySQL, της γλώσσας PHP και Perl. Το XAMPP συμπεριλαμβάνει επίσης τα πακέτα OpenSSL και το phpMyAdmin.

### **Χρήση**

Επίσημα οι σχεδιαστές του XAMPP προόριζαν το λογισμικό ως εργαλείο ανάπτυξης και δοκιμής ιστοσελίδων τοπικά στον υπολογιστή χωρίς να είναι απαραίτητη η σύνδεση στο διαδίκτυο. Για να είναι δυνατή η χρήση του, πολλές σημαντικές λειτουργίες ασφάλειας έχουν απενεργοποιηθεί. Στην πράξη το XAMPP ορισμένες φορές χρησιμοποιείται και για την φιλοξενία ιστοσελίδων. Υπάρχει ειδικό εργαλείο το οποίο περιέχεται στο XAMPP για την προστασία με κωδικό των σημαντικών μερών. Το XAMPP υποστηρίζει την δημιουργία και διαχείριση βάσεων δεδομένων τύπου MySQL και SQLite.

Όταν το XAMPP εγκατασταθεί στον τοπικό υπολογιστή διαχειρίζεται τον localhost ως ένα απομακρυσμένο κόμβο, ο οποίος συνδέεται με το πρωτόκολλο μεταφοράς αρχείων FTP. Η σύνδεση στον localhost μέσω του FTP μπορεί να γίνει με το όνομα χρήστη "newuser" και το κωδικό "wampp". Για την βάση δεδομένων MySQL ο υπάρχει ο χρήστης "root" χωρίς κωδικό πρόσβασης.

## Κεφάλαιο 4

### Ανάλυση Προβλήματος

#### 4.1 Γενική περιγραφή

Για την διαχείριση των πτυχιικών εργασιών ενός Εκπαιδευτικού Ιδρύματος αναπτύχθηκε μια δυναμική ιστοσελίδα η οποία αποθηκεύει και επεξεργάζεται δεδομένα μέσω μιας βάσης δεδομένων, η οποία θα υλοποιήθηκε με XML .Η διασύνδεση της βάσης δεδομένων με την εφαρμογή και οι λειτουργίες οι οποίες υποστηρίζονται υλοποιήθηκαν με την γλώσσα PHP . Την εφαρμογή θα την χειρίζεται ένας χρήστης-υπάλληλος μιας γραμματείας ενός Εκπαιδευτικού Ιδρύματος ο οποίος θα είναι υπεύθυνος για την διαχείριση των πτυχιικών εργασιών. Βασικό κριτήριο είναι η εφαρμογή μας να είναι εύκολη και κατανοητή προς τον χρήστη.

Μέσω της εφαρμογής θα πρέπει να υποστηρίζονται οι λειτουργίες:

- **Καταχώριση Στοιχείων Φοιτητών**

Για κάθε φοιτητή θα αποθηκεύουμε:

1. Όνοματεπώνυμο
2. Όνομα Πατρός
3. Όνομα Μητρός
4. Αριθμός Ταυτότητας
5. Τοπική Διεύθυνση
6. Μόνιμη Διεύθυνση
7. Τηλέφωνο
8. Σειρά Επιτυχίας
9. Email
10. Μέσος Όρος ο οποίος θα ενημερώνεται κάθε φορά που θα περνάει ή αναβαθμολογεί κάποιο μάθημα
11. Αριθμός Μητρώου ο οποίος είναι μοναδικός και γίνεται αυτόματη ανάθεση

- **Καταχώριση Στοιχείων Καθηγητών**

Για κάθε καθηγητή θα αποθηκεύουμε:

1. Όνοματεπώνυμο
2. Αριθμός Ταυτότητας
3. Διεύθυνση Γραφείου
4. Τηλέφωνο
5. Τηλέφωνο Πανεπιστημίου
6. Βαθμίδα
7. Τμήμα στο οποίο ανήκει
8. Κωδικός Καθηγητή ο οποίος είναι μοναδικός και γίνεται αυτόματη ανάθεση

- **Καταχώριση Στοιχείων Πτυχιικών**

Για κάθε πτυχιική θα αποθηκεύουμε:

1. Όνομασία πτυχιικής
2. Τμήμα στο οποίο δίνεται
3. Επιβλέπων Καθηγητής
4. Κωδικός Πτυχιικής ο οποίος είναι μοναδικός και γίνεται αυτόματη ανάθεση

- **Εμφάνιση Στοιχείων Καθηγητών**



- Εμφάνιση Στοιχείων Φοιτητών
- Εμφάνιση Στοιχείων όλων των Πτυχιακών
- Εμφάνιση Ελεύθερων Πτυχιακών
- Εμφάνιση Φοιτητών που έχουν κατοχυρώσει θέμα Πτυχιακής και Βαθμό αν την έχουν ολοκληρώσει
- Ανάθεση Πτυχιακής σε Φοιτητή
- Ενημέρωση Βαθμού Πτυχιακής
- Αποδέσμευση Πτυχιακής από Φοιτητή
- Ενημέρωση μέσου όρου Φοιτητών

## 4.2 Διαδικασία ανάπτυξης

Για την διαδικασία υλοποίησης της εφαρμογής πρέπει να εγκαταστήσουμε έναν web server ο οποίος θα φιλοξενεί τις ιστοσελίδες, με προτιμότερη λύση τον Apache ο οποίος διατίθεται δωρεάν. Στην προκειμένη περίπτωση εγκαταστήσαμε το πακέτο προγραμμάτων ελεύθερου λογισμικού XAMPP. Έπειτα γράψαμε τον κώδικα της εφαρμογής με την βοήθεια ενός απλού text editor ,εδώ για διευκόλυνση χρησιμοποιήσαμε το ελεύθερο λογισμικό notepad++ . Τα αρχεία που δημιουργήσαμε τα αποθηκεύσαμε στο site root του Apache στο htdocs και στη συνέχεια με τη χρήση του φυλλομετρητή θα τρέχει η εφαρμογή . Η εφαρμογή αποτελείται από αρχεία HTML, PHP, και XML.

Η PHP είναι η γλώσσα προγραμματισμού για τις δυναμικές ιστοσελίδες και τις φόρμες που αναπτύχθηκαν. Η βάση δεδομένων για την αποθήκευση των δεδομένων και την επεξεργασία υλοποιήθηκε με XML. Οι στατικές ιστοσελίδες αναπτύχθηκαν με την χρήση της HTML.

### Βασικές υποδείξεις στον κώδικα της γλώσσας PHP για την υλοποίηση της εφαρμογής

Για να μπορέσουμε να προσπελάσουμε, να καταχωρίσουμε και να κάνουμε αλλαγές στα αρχεία XML που αποτελούν στην βάση δεδομένων , χρησιμοποιήσαμε την μέθοδο SimpleXML την οποία αναλύσαμε στο Κεφάλαιο 3.

Στον κώδικα των σελίδων που έχουν γραφτεί με PHP και γίνεται καταχώριση στοιχείων, με την SimpleXML φορτώνουμε το αρχείο XML, παράδειγμα: `$xml=simplexml_load_file($xml_file)`. Έπειτα τα στοιχεία από τις φόρμες τα παίρνουμε με την μέθοδο POST και αποθηκεύουμε τα δεδομένα για κάθε ετικέτα στο αρχείο XML καλώντας την συνάρτηση `asXML()` , παράδειγμα: `$xml->asXML($xmlfile)`

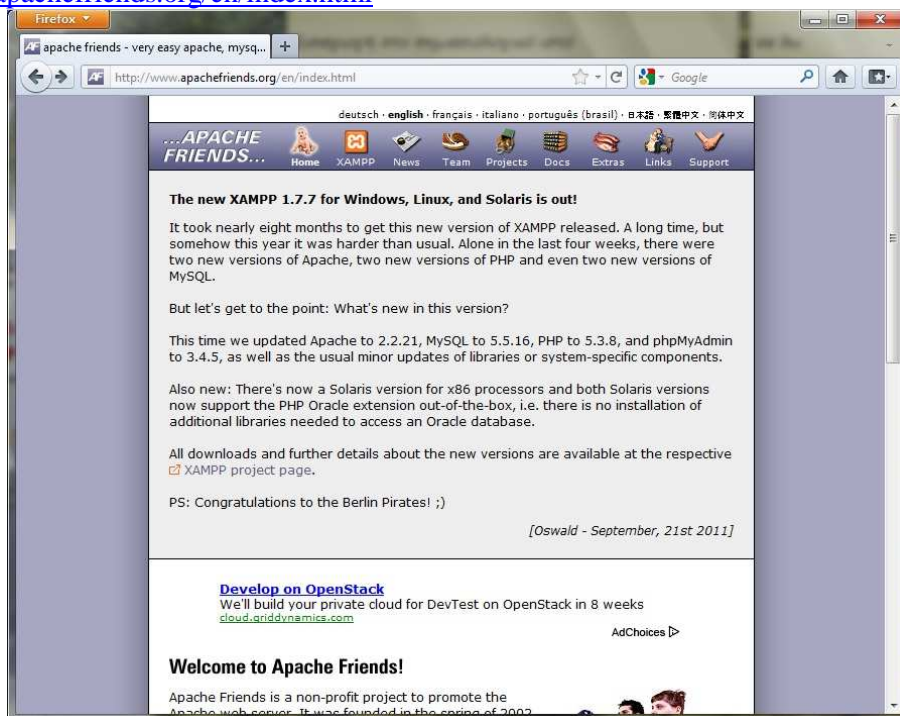
Στον κώδικα των σελίδων που έχουν γραφτεί με PHP και γίνεται εμφάνιση στοιχείων, με την SimpleXML φορτώνουμε το αρχείο XML, παράδειγμα: `$xml=simplexml_load_file($xml_file)`. Έπειτα με την συνάρτηση `foreach` δημιουργήσαμε ένα βρόγχο ο οποίος περνάει από κάθε ετικέτα του αρχείου και στην έξοδο έχουμε όνομα του στοιχείου και τα δεδομένα για κάθε ετικέτα.

Στον κώδικα των σελίδων που έχουν γραφτεί με PHP και γίνονται λειτουργίες όπως ανάθεση πτυχιακής σε φοιτητή και άλλες, δηλαδή καταχώριση και αλλαγές στα στοιχεία, με την SimpleXML φορτώνουμε το αρχείο XML ή και παραπάνω αρχεία XML, παράδειγμα: `$xml=simplexml_load_file($xml_file)`. Έπειτα με τις αλλαγές στις εγγραφές των ετικετών που γίνονται για να αποθηκεύσουμε το νέο Document σε αρχείο ως XML μετατρέπουμε το Document σε XMLDocument και καλούμε την μέθοδο `fwrite`.

### 4.3 Εγκατάσταση Λογισμικού XAMPP

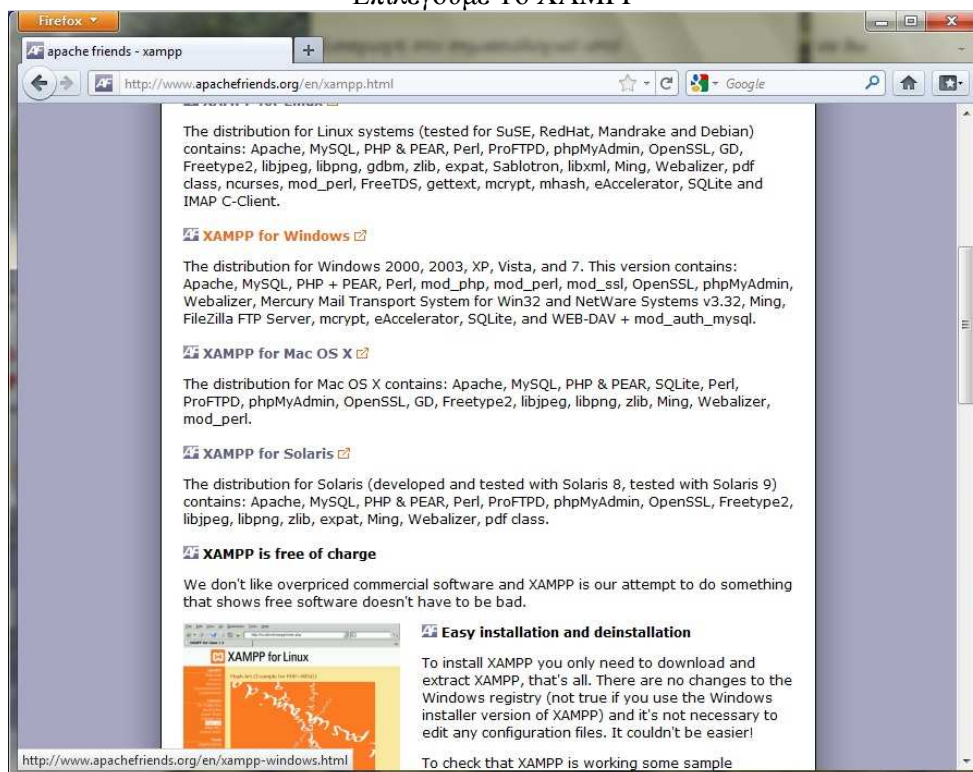
Για να κατεβάσουμε και να εγκαταστήσουμε το XAMPP στον υπολογιστή μας κατευθυνόμαστε στην επίσημη ιστοσελίδα

<http://www.apachefriends.org/en/index.html>

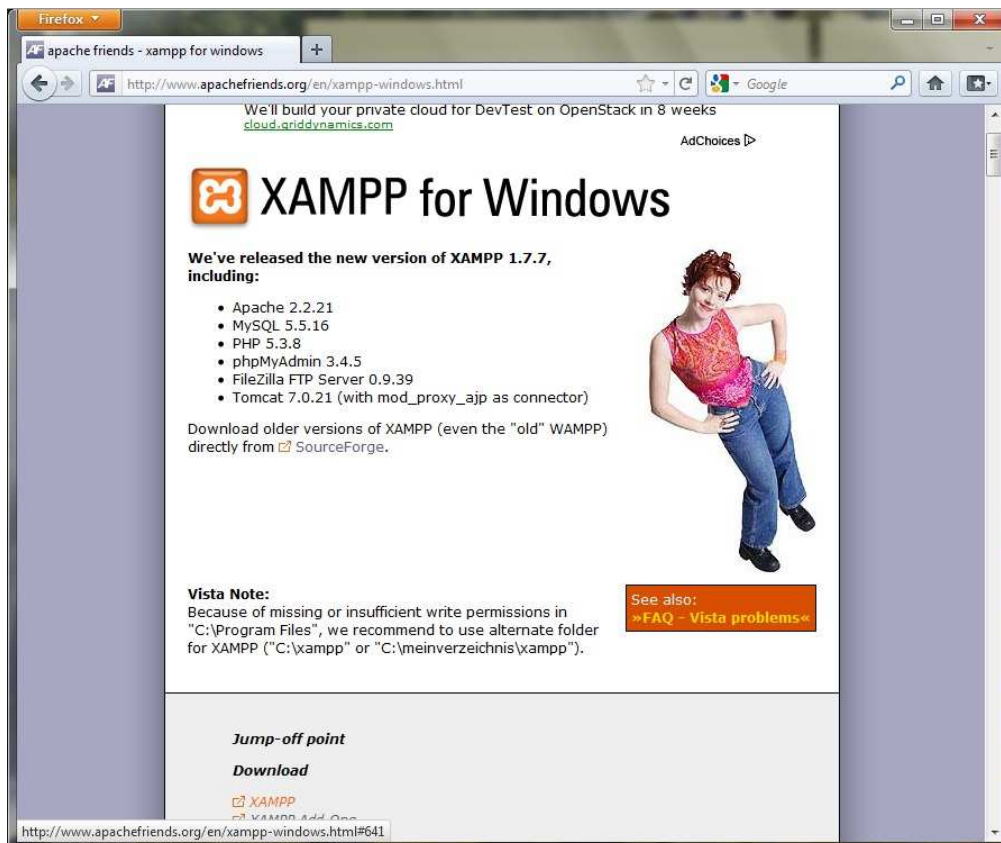


Εικόνα 9 .Αρχική σελίδα [www.apachefriends.org](http://www.apachefriends.org)

#### Επιλέγουμε Το XAMPP



Εικόνα 10 .Σελίδα επιλογής λειτουργικών συστημάτων για XAMPP  
Στην συνέχεια επιλέγουμε την έκδοση «XAMPP for Windows»

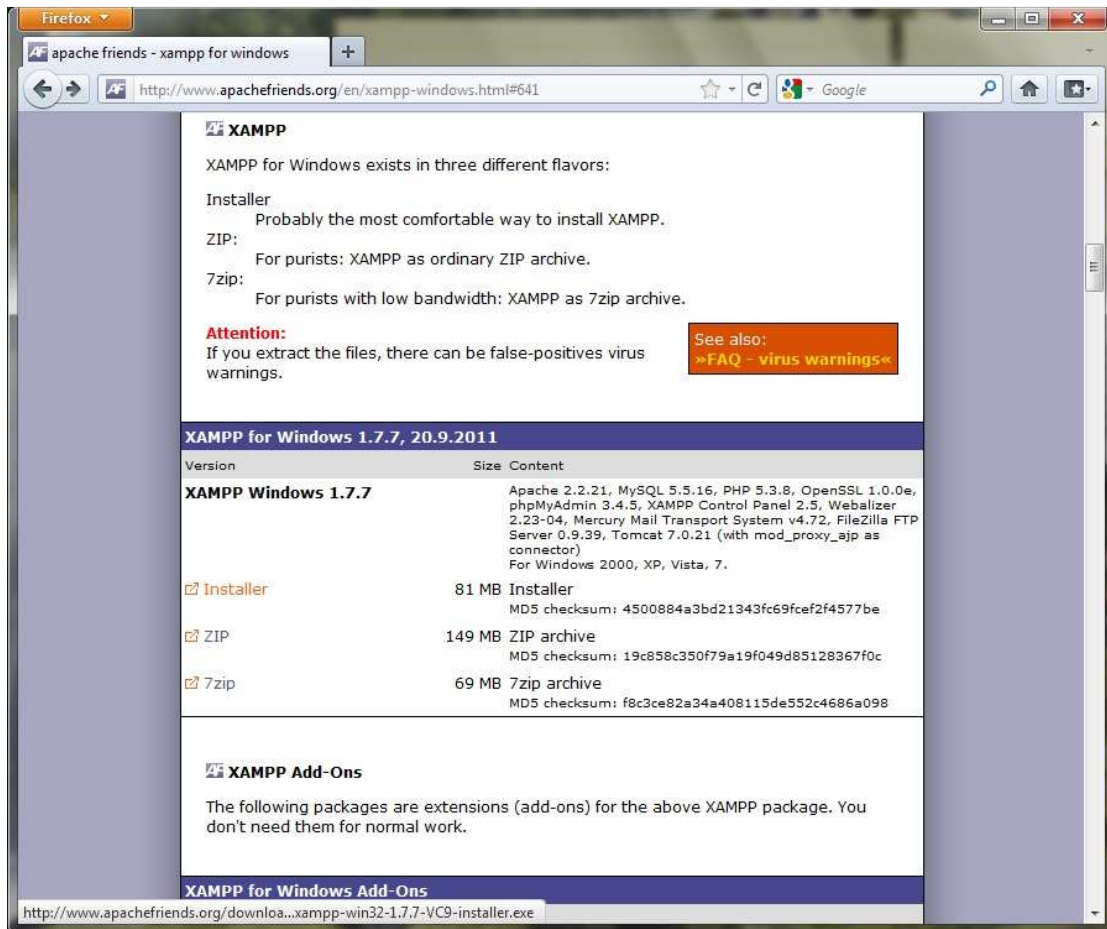


Εικόνα 11 .Εκδόσεις προγραμμάτων XAMPP

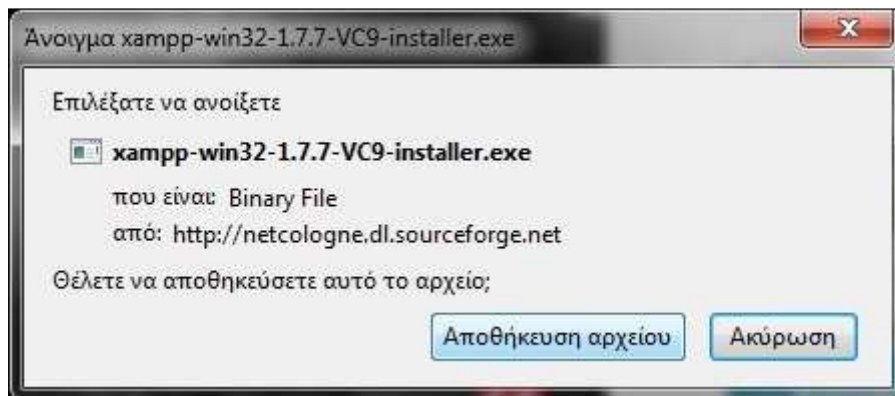
Σε αυτήν τη σελίδα βλέπουμε τις εκδόσεις των προγραμμάτων που περιέχονται στο λογισμικό XAMPP for Windows και επιλέγουμε από κάτω το XAMPP για να δούμε τις επιλογές στο αρχείο που θα κατεβάσουμε.

Οι νεότερες εκδόσεις που περιέχονται στο XAMPP είναι:

- Apache 2.2.21
- MySql 5.5.16
- PHP 5.3.8
- phpMyAdmin 3.4.5
- FileZilla FTP Server 0.9.39
- Tomcat 7.0.21

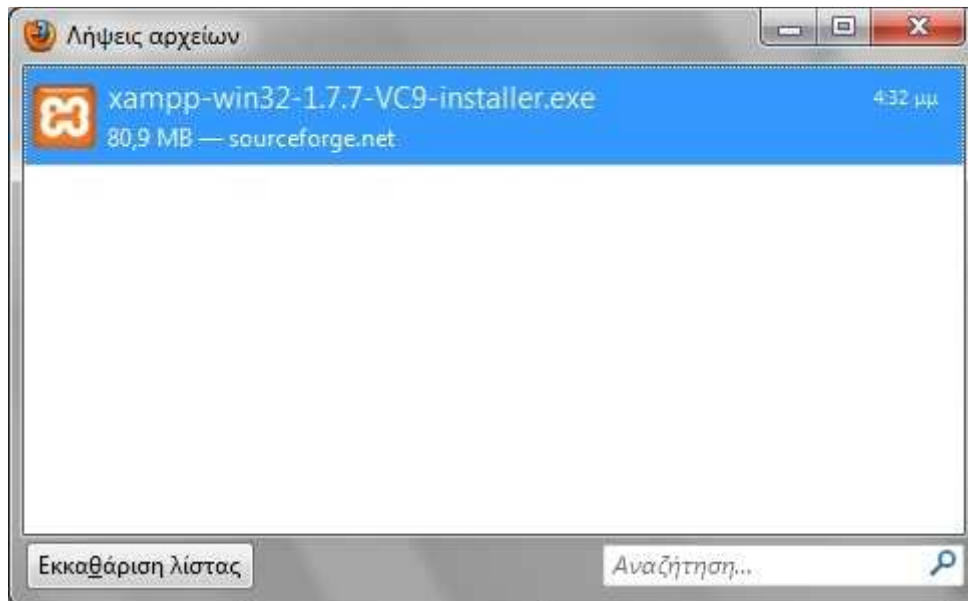


Εικόνα 12 .Επιλογή Installer δηλαδή το αρχείο .exe



Εικόνα 13 .Παράθυρο λήψης αρχείου εγκατάστασης XAMPP

Επιλέγουμε «Αποθήκευση αρχείου»



Εικόνα 14 .Λήψη αρχείου εγκατάστασης XAMPP

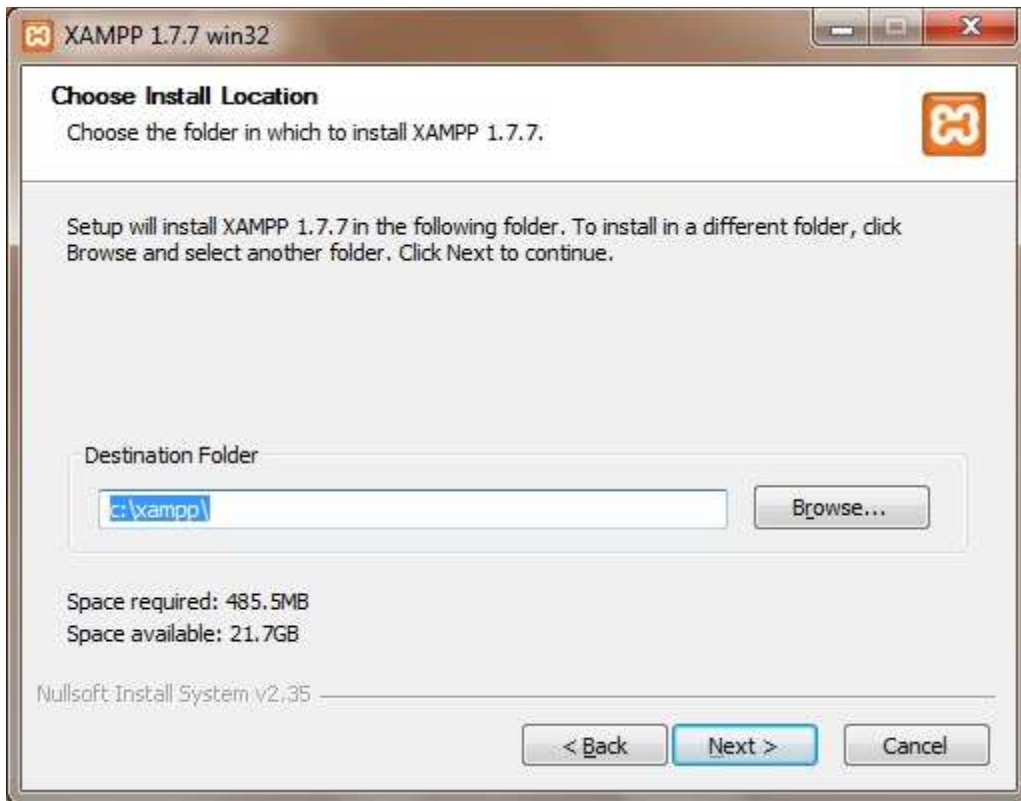
Ολοκληρώθηκε το κατέβασμα του αρχείου στον υπολογιστή μας και ξεκινάμε την εγκατάσταση



Εικόνα 15 .Εκκίνηση εγκατάστασης XAMPP

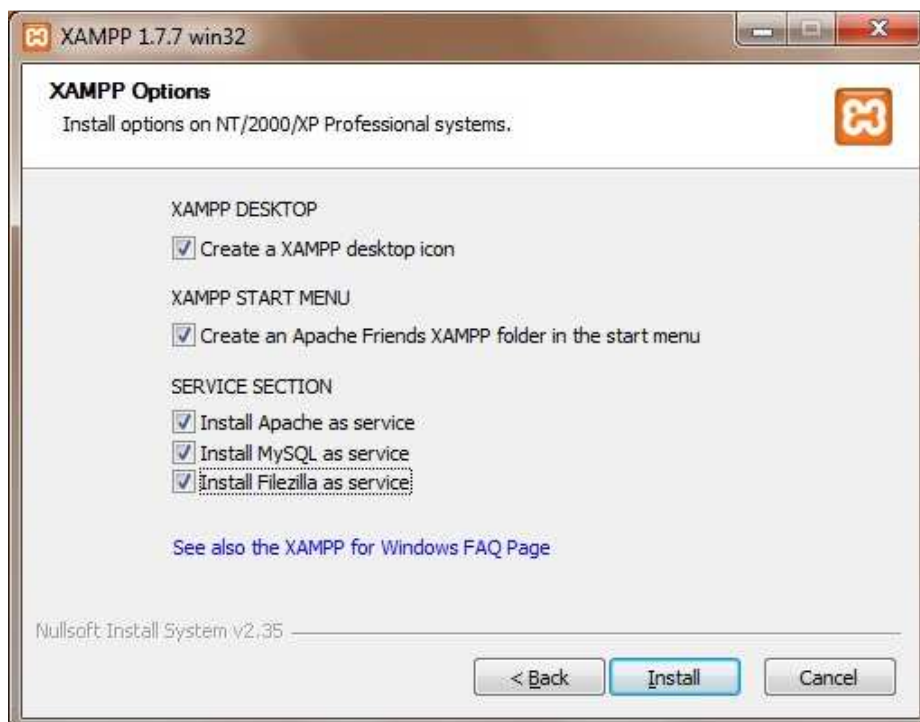
Μόλις ανοίξει το παράθυρο της εγκατάστασης επιλέγουμε «Next» για συνέχεια





Εικόνα 16 . Επιλογή φακέλου εγκατάστασης στο δίσκο

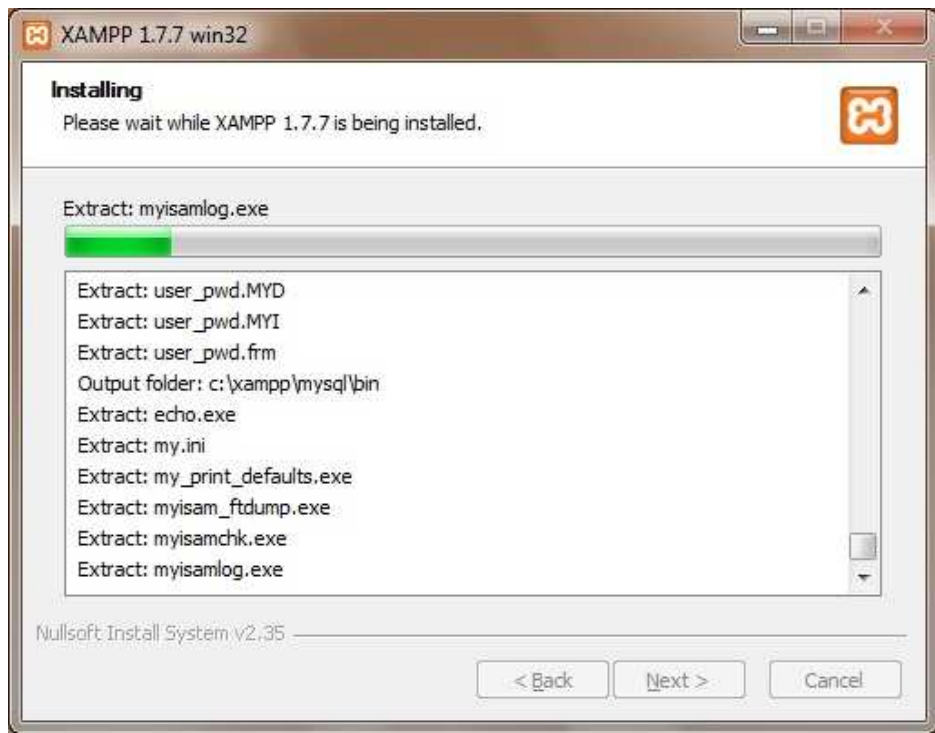
Επιλέγουμε σε πιο μέρος του δίσκου θα το εγκαταστήσουμε και συνεχίζουμε την εγκατάσταση επιλέγοντας «Next»



Εικόνα 17 .Επιλογές XAMPP

Τσεκάρουμε τις επιλογές και επιλέγουμε «Install»





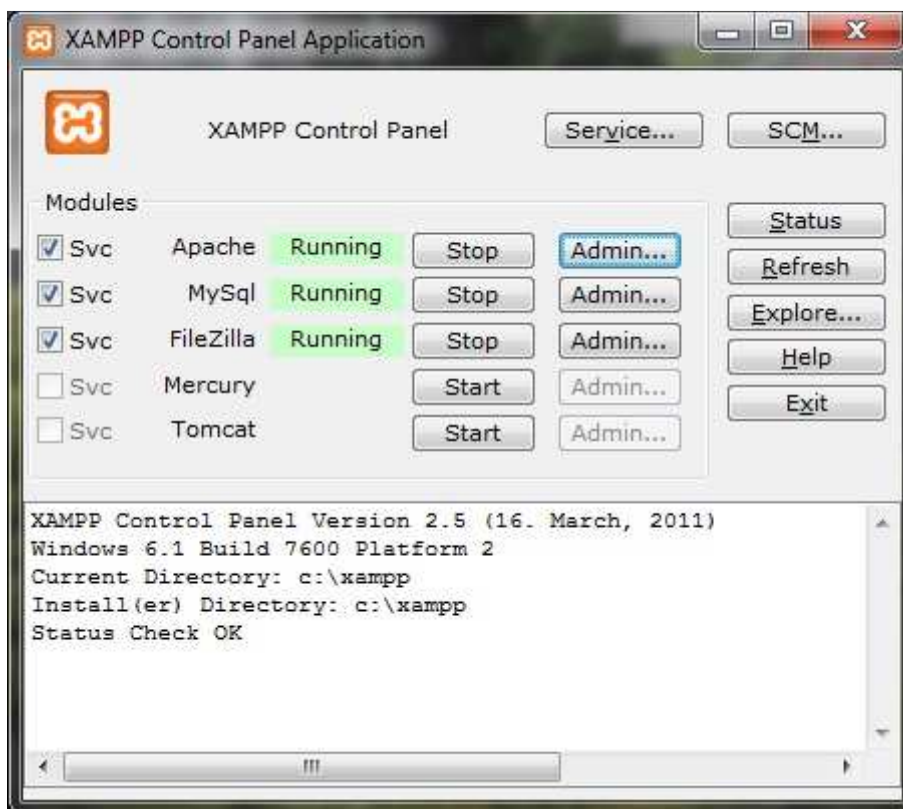
Εικόνα 18 .Εκκίνηση εγκατάστασης

Η εγκατάσταση ξεκίνησε και παρατηρούμε την μπάρα προόδου



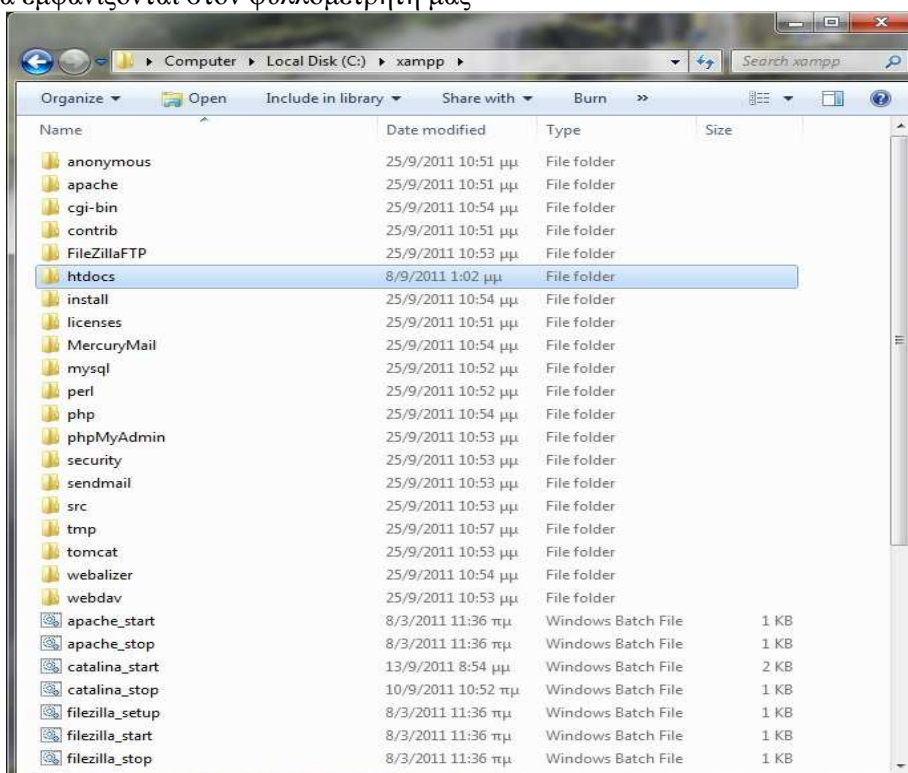
Εικόνα 19.Ολοκλήρωση εγκατάστασης

Η εγκατάσταση έχει ολοκληρωθεί επιλέγουμε το «**Finish**» και ξεκινάμε το λογισμικό



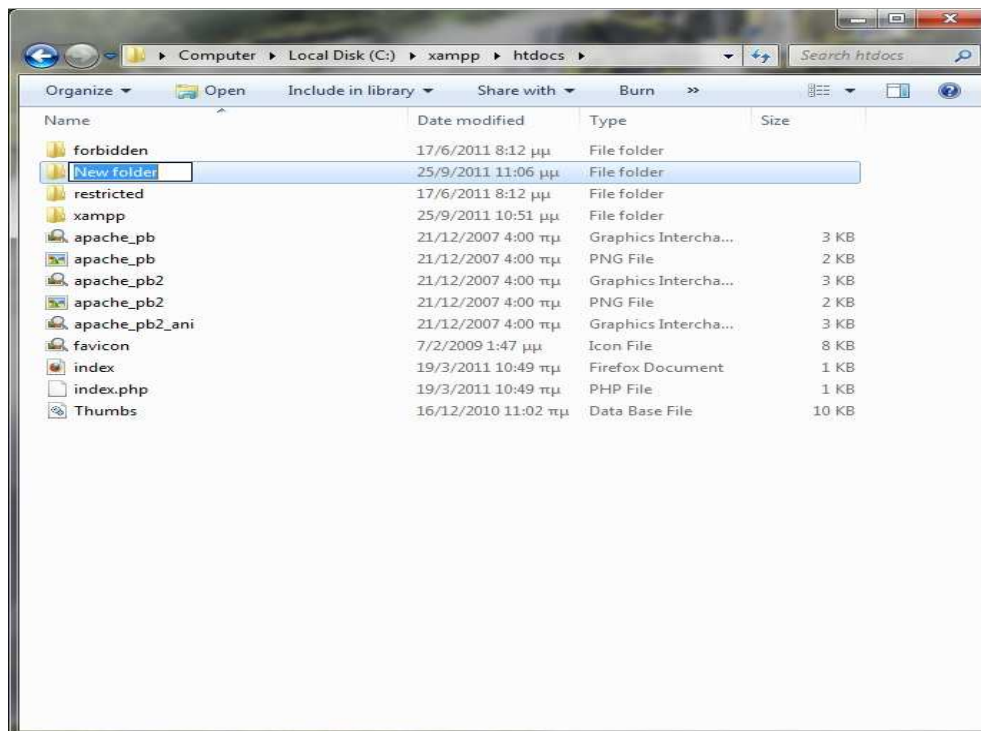
**Εικόνα 20 . XAMPP Control Panel Application**

Από το «XAMPP Control Panel Application» επιλέγουμε ποιες διεργασίες θα ξεκινήσουν. Στην συνέχεια πηγαίνουμε στα αρχεία του XAMPP και ανοίγουμε τον φάκελο htdocs όπου θα αποθηκεύουμε τα αρχεία της εφαρμογής που θα δημιουργήσουμε και από αυτόν τον φάκελο του server μας θα εμφανίζονται στον φυλλομετρητή μας



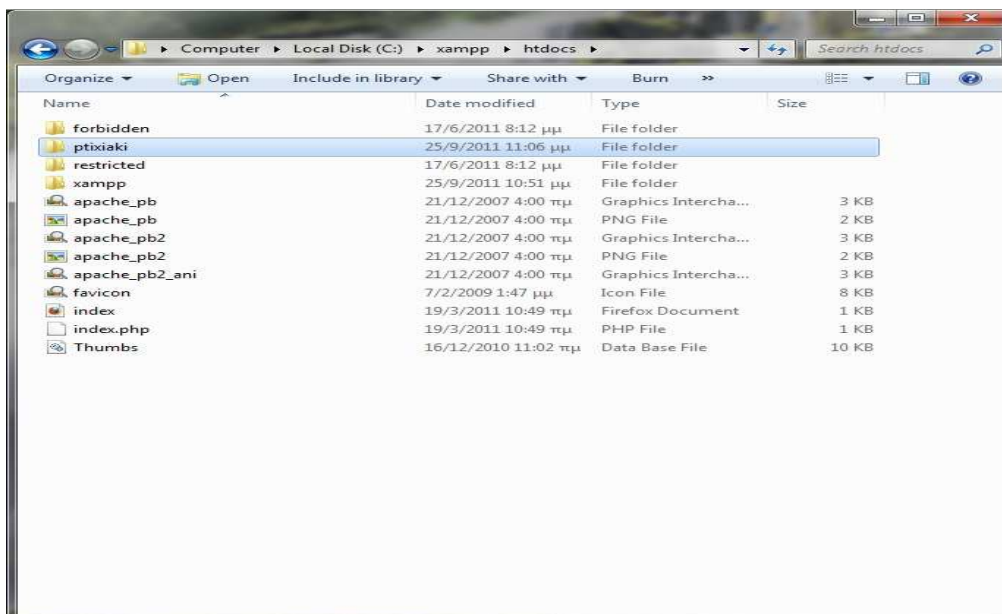
**Εικόνα 21 .Αρχεία και φάκελοι XAMPP**

Βλέπουμε τα αρχεία και τους φακέλους του XAMPP επιλέγουμε και ανοίγουμε τον φάκελο htdocs



Εικόνα 22. Αρχεία και φακέλοι στο htdocs και δημιουργία νέου φακέλου

Βλέπουμε τα αρχεία και τους φακέλους του htdocs και δημιουργούμε ένα νέο φάκελο



Εικόνα 23. Δημιουργία και ονομασία νέου φακέλου στο htdocs

Τον νέο φάκελο τον ονομάζουμε ptixiaki στην προκειμένη περίπτωση και σε αυτόν αποθηκεύουμε τα αρχεία της εφαρμογής μας, τα οποία θα εμφανίζονται στο φυλλομετρητή με τη διεύθυνση : <http://localhost/ptixiaki>

#### 4.4 Σχεδίαση Βάσης δεδομένων

Η βάση δεδομένων υλοποιήθηκε σε μορφή XML η οποία διευκολύνει τον υπολογιστή να παράγει δεδομένα, να διαβάζει δεδομένα και να εξασφαλίζει τη σαφήνεια της δομής των δεδομένων. Η βάση δεδομένων αποτελείται από τρία αρχεία XML όπου αποτελούν τους πίνακες της βάσης και συνδέονται με την εφαρμογή μέσω της PHP ώστε να γίνεται η αποθήκευση και επεξεργασία των δεδομένων.

##### Περιγραφή πινάκων της βάσης δεδομένων

1)Ο πίνακας foitites χρησιμοποιείται για την καταγραφή των στοιχείων των φοιτητών. Τα στοιχεία των φοιτητών που αποθηκεύουμε σε αντιστοιχία με τις ετικέτες του αρχείου XML είναι :

Αριθμός Μητρώου→am

Όνοματεπώνυμο→ onomateponymof

Όνομα Πατρός→ onomapatros

Όνομα Μητρός→ taytotitaf

Αριθμός Ταυτότητας→ taytotitaf

Τοπική Διεύθυνση→ tdiefthinsi

Μόνιμη Διεύθυνση→ mdiefthinsi

Τηλέφωνο→ tilefwno

Σειρά Επιτυχίας→ seiraepityxias

Email→ email

Μέσος Όρος→ mesosoros

Το αρχείο foitites.xml με τις ετικέτες είναι:

```
<foitites>
  <foititis>
    <am></am>
    <onomateponymof></onomateponymof>
    <onomapatros></onomapatros>
    <onomamhtros></onomamhtros>
    <taytotitaf></taytotitaf>
    <tdiefthinsi></tdiefthinsi>
    <mdiefthinsi></mdiefthinsi>
    <tilefwno></tilefwno>
    <seiraepityxias></seiraepityxias>
    <email></email>
    <mesosoros></mesosoros>
  </foititis>
</foitites>
```

2) Ο πίνακας kathigites χρησιμοποιείται για την καταγραφή των στοιχείων των καθηγητών. Τα στοιχεία των καθηγητών που αποθηκεύουμε σε αντιστοιχία με τις ετικέτες του αρχείου XML είναι :

Κωδικός Καθηγητή → kodikosk  
Όνοματεπώνυμο → onomateponymok  
Αριθμός Ταυτότητας → taytotitak  
Διεύθυνση Γραφείου → diefthinsigrafeio  
Τηλέφωνο → tilefonok  
Τηλέφωνο Πανεπιστημίου → tilefonopane  
Βαθμίδα → vathmida  
Τμήμα στο οποίο ανήκει → tmimaanikei

Το αρχείο kathigites.xml με τις ετικέτες είναι:

```
<kathigites>
  <kathigitis>
    <kodikosk></kodikosk>
    <onomateponymok></onomateponymok>
    <taytotitak></taytotitak>
    <diefthinsigrafeio></diefthinsigrafeio>
    <tilefonok></tilefonok>
    <tilefonopane></tilefonopane>
    <vathmida></vathmida>
    <tmimaanikei></tmimaanikei>
  </kathigitis>
</kathigites>
```

3) Ο πίνακας ptixiakes χρησιμοποιείται για την καταγραφή των στοιχείων των πτυχιακών. Τα στοιχεία των πτυχιακών που αποθηκεύουμε σε αντιστοιχία με τις ετικέτες του αρχείου XML είναι :

Κωδικός Πτυχιακής → kodikospt  
Όνομασία πτυχιακής → onomasia  
Τμήμα στο οποίο δίνεται → tmima  
Επιβλέπων Καθηγητής → onomateponymoptk  
Όνοματεπώνυμο φοιτητή που έχει κατοχυρώσει πτυχιακή → onomateponymoptf  
Βαθμός πτυχιακής → vathmospt

Το αρχείο ptixiakes.xml με τις ετικέτες είναι:

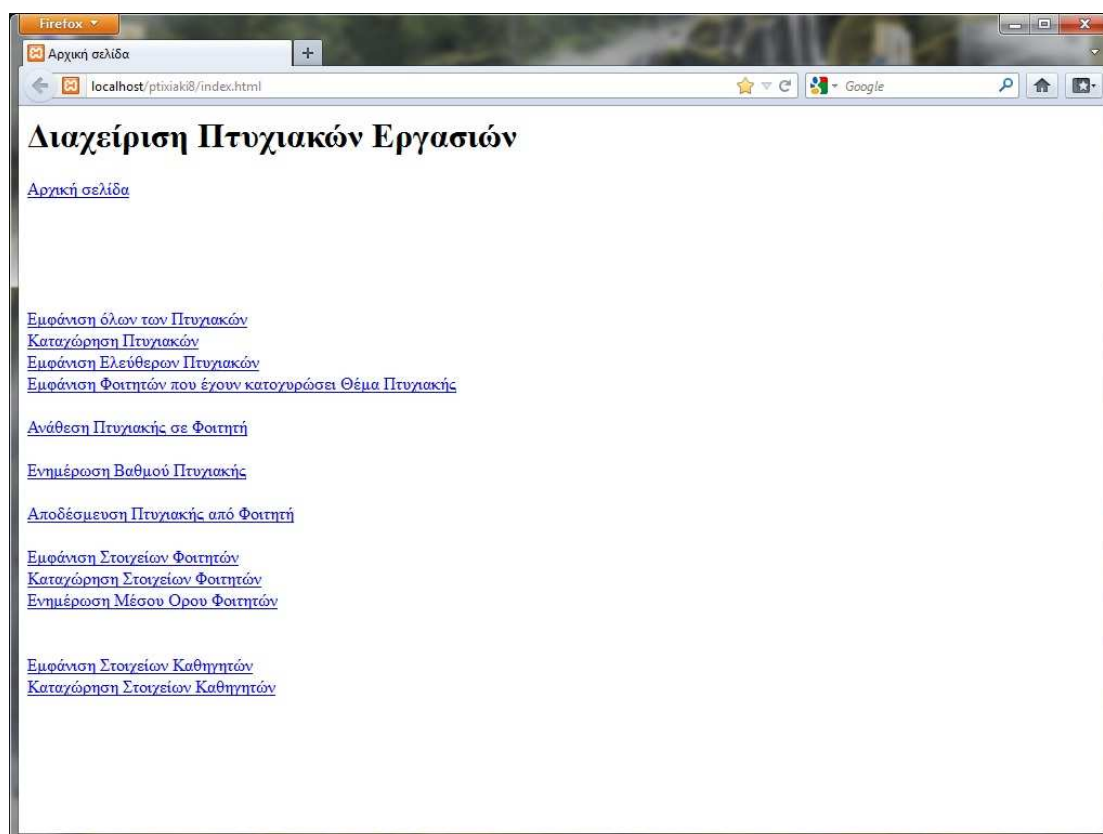
```
<ptixiakes>
  <ptixiaki>
    <kodikospt></kodikospt>
    <onomasia></onomasia>
    <tmima></tmima>
    <onomateponymoptk></onomateponymoptk>
    <onomateponymoptf></onomateponymoptf>
    <vathmospt></vathmospt>
  </ptixiaki>
</ptixiakes>
```

## 4.5 Υλοποίηση της Εφαρμογής

Στην ενότητα αυτήν παραθέτουμε στιγμιότυπα από την εφαρμογή με σενάρια χρήσης και επεξήγηση κώδικα.

Τρέχουμε το λογισμικό XAMPP για να ξεκινήσει ο web server Apache και έπειτα γράφουμε στον φυλλομετρητή στην διεύθυνση <http://localhost/> και δεξιά από το / το όνομα του αρχείου που θέλουμε να τρέξουμε ,στην προκειμένη περίπτωση βάζουμε το όνομα του αρχείου της Αρχικής Σελίδας [index.html](#) .Τα αρχεία html και php τα όποια τρέχουμε στην συγκεκριμένη εφαρμογή βρίσκονται στον φάκελο htdocs που βρίσκεται στον φάκελο που εγκαταστάθηκε το XAMPP.

### 4.5.1 Αρχική σελίδα



Εικόνα 24. Αρχική Σελίδα

Εδώ βλέπουμε την «[Αρχική σελίδα](#)» όπου ο χρήστης βρίσκεται στο κεντρικό μενού της εφαρμογής απ' όπου μπορεί να επιλέξει μια λειτουργία.

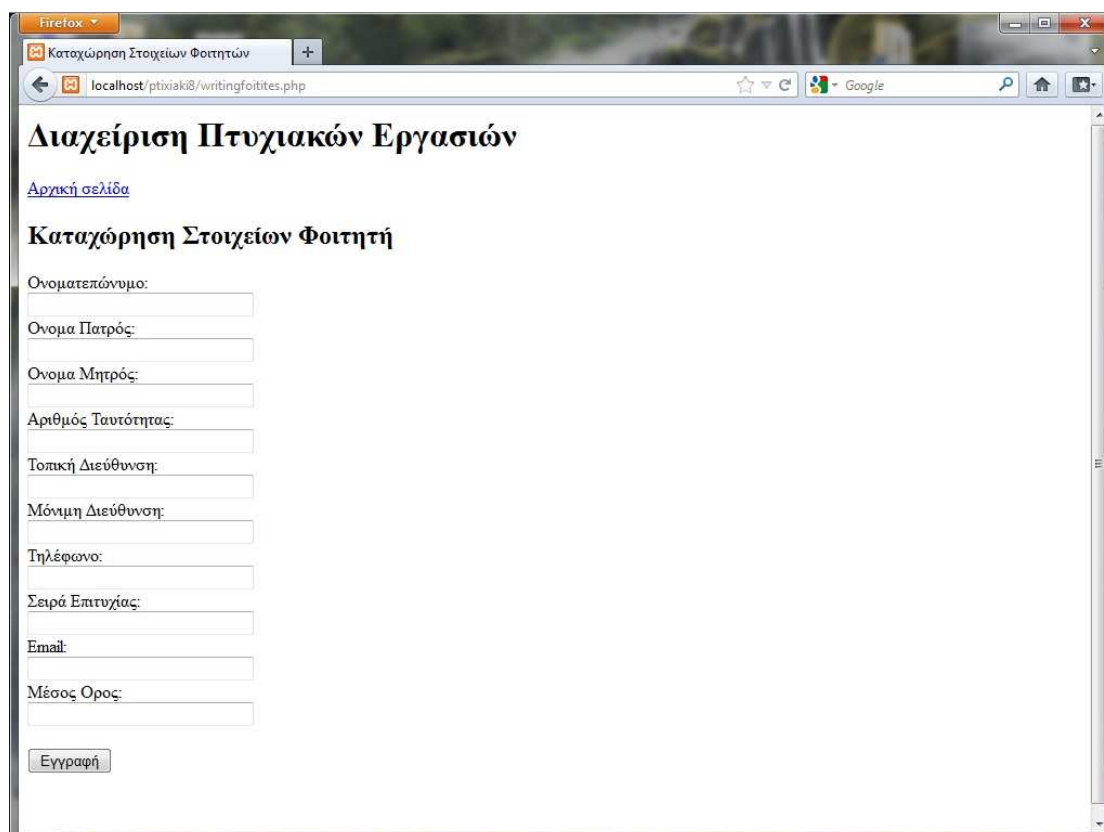
#### Ο κώδικας της σελίδας [Αρχική σελίδα](#) [index.html](#)

```
<html>
<head>
<title>Αρχική σελίδα</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<br>
```



```
<a href="readingptixiakes.php">Εμφάνιση όλων των Πτυχιακών</a><br />
<a href="writingptixiakes.php">Καταχώρηση Πτυχιακών</a><br />
<a href="freertixiakes.php">Εμφάνιση Ελεύθερων Πτυχιακών</a><br />
<a href="readregptixiakes.php">Εμφάνιση Φοιτητών που έχουν κατοχυρώσει Θέμα
Πτυχιακής</a><br />
<br>
<a href="anatesi.php">Ανάθεση Πτυχιακής σε Φοιτητή</a><br />
<br>
<a href="vathmosptixiakis.php">Ενημέρωση Βαθμού Πτυχιακής</a><br />
<br>
<a href="apodesmeusi.php">Αποδέσμευση Πτυχιακής από Φοιτητή</a><br />
<br>
<a href="readingfoitites.php">Εμφάνιση Στοιχείων Φοιτητών</a><br />
<a href="writingfoitites.php">Καταχώρηση Στοιχείων Φοιτητών</a><br />
<a href="mesosorosfoitites.php">Ενημέρωση Μέσου Ορου Φοιτητών</a>
<br>
<a href="readingkathigites.php">Εμφάνιση Στοιχείων Καθηγητών</a><br />
<a href="writingkathigites.php">Καταχώρηση Στοιχείων Καθηγητών</a>
<br>
</body>
</html>
```

#### 4.5.2 Καταχώρηση Στοιχείων Φοιτητή



The screenshot shows a Firefox browser window with the address bar displaying 'localhost/ptixiakib/writingfoitites.php'. The page title is 'Καταχώρηση Στοιχείων Φοιτητών'. Below the title, there is a link for 'Αρχική σελίδα'. The main heading is 'Καταχώρηση Στοιχείων Φοιτητή'. The form contains the following fields: Ονοματεπώνυμο, Όνομα Πατρός, Όνομα Μητρός, Αριθμός Ταυτότητας, Τοπική Διεύθυνση, Μόνιμη Διεύθυνση, Τηλέφωνο, Σειρά Επιτυχίας, Email, and Μέσος Ορος. At the bottom of the form is a button labeled 'Εγγραφή'.

Εικόνα 25. Καταχώρηση Στοιχείων Φοιτητή

Στην παραπάνω σελίδα ο χρήστης βλέπει μια φόρμα ώστε να μπορεί να εισάγει τα απαραίτητα στοιχεία για κάθε φοιτητή .Μόλις ολοκληρώσει την καταγραφή των στοιχείων στην φόρμα πατάει το

κουμπί «**Εγγραφή**» και αφού γίνει ένας έλεγχος ότι συμπληρωθήκαν όλα τα πεδία τότε τα στοιχεία αποθηκεύονται στην βάση δεδομένων και βγάζει το μήνυμα ελέγχου «**Έγινε η καταχώρηση**». Αν δεν συμπληρώσει ο χρήστης όλα τα πεδία και πατήσει το κουμπί «**Εγγραφή**» τότε θα εμφανίσει το μήνυμα ελέγχου «**Όλα να πεδία πρέπει να συμπληρωθούν**». Στο πάνω μέρος της σελίδας υπάρχει ο σύνδεσμος «**Αρχική Σελίδα**» όπου ο χρήστης μπορεί να επιστρέψει στην Αρχική Σελίδα

## Ο κώδικας της σελίδας Καταχώρηση Στοιχείων Φοιτητή

### writingfoitites.php

```
<html>
<head>
<title>Καταχώρηση Στοιχείων Φοιτητών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
//Σύνδεση με την βάση δεδομένων foitites.xml
$xml_file = "foitites.xml";

if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}

if(isset($_POST["u_onomateponymof"]) &&isset($_POST["u_onomapatros"])
&&isset($_POST["u_onomamhtros"])&&isset($_POST["u_taytotitaf"])&&isset($_POST["u_
tdiefthinsi"])
&&isset($_POST["u_mdiefthinsi"])&&isset($_POST["u_tilefwno"])&&isset($_POST["u_sei
raepityxias"])
&&isset($_POST["u_email"])&&isset($_POST["u_mesosoros"])){
//Αποθήκευση δεδομένων τα όποια πήραμε με την μέθοδο POST από την φόρμα σε τοπικές
μεταβλητές

    $u_onomateponymof = stripslashes($_POST['u_onomateponymof']);
    $u_onomapatros = stripslashes($_POST['u_onomapatros']);
    $u_onomamhtros = stripslashes($_POST['u_onomamhtros']);
    $u_taytotitaf = stripslashes($_POST['u_taytotitaf']);
    $u_tdiefthinsi = stripslashes($_POST['u_tdiefthinsi']);
    $u_mdiefthinsi = stripslashes($_POST['u_mdiefthinsi']);
    $u_tilefwno = stripslashes($_POST['u_tilefwno']);
    $u_seiraepityxias = stripslashes($_POST['u_seiraepityxias']);
    $u_email = stripslashes($_POST['u_email']);
    $u_mesosoros = stripslashes($_POST['u_mesosoros']);

//Έλεγχος αν συμπληρώθηκαν όλα τα πεδία στην φόρμα

    if(!empty($u_onomateponymof)&& !empty($u_onomapatros)
&& !empty($u_onomamhtros)&& !empty($u_taytotitaf)&& !empty($u_tdiefthinsi)
&& !empty($u_mdiefthinsi)&& !empty($u_tilefwno)&& !empty($u_seiraepityxias)
&& !empty($u_email)&& !empty($u_mesosoros)){
//Έλεγχος του προηγούμενου αριθμού μητρώου και αυτόματη ανάθεση στον αμέσως επόμενο αριθμό
μητρώου με την νέα καταχώρηση στοιχείων φοιτητή
```

```

        foreach($xml as $foititis){
            $last_id = $foititis->am;
            }

            $id = $last_id+1;
//Αποθήκευση των στοιχείων των τοπικών μεταβλητών στις ετικέτες της βάσης δεδομένων

            $x = $xml->addChild("foititis");
            $x->addChild("am",$id);
            $x->addChild("onomateponymof",$u_onomateponymof);
            $x->addChild("onomapatros",$u_onomapatros);
            $x->addChild("onomamhtros",$u_onomamhtros);
            $x->addChild("taytotitaf",$u_taytotitaf);
            $x->addChild("tdiefthinsi",$u_tdiefthinsi);
            $x->addChild("mdiefthinsi",$u_mdiefthinsi);
            $x->addChild("tilefwno",$u_tilefwno);
            $x->addChild("seiraepityxias",$u_seiraepityxias);
            $x->addChild("email",$u_email);
            $x->addChild("mesosoros",$u_mesosoros);

            $xml->asXML($xml_file);

            echo ("<font color='blue'>Έγινε η Καταχώρηση</font>");
            }else {
                echo ("<font color='red'>Όλα τα πεδία πρέπει να
συμπληρωθούν.</font>");
            }

        }

?>

```

**//Δημιουργία της φόρμας της σελίδας**

<h2>Καταχώρηση Στοιχείων Φοιτητή</h2>

<form action="writingfoitites.php" method="post" name="xml\_writing" >

Όνοματεπώνυμο:<br />

<input type="text" name="u\_onomateponymof" size="30" />

<br />

Όνομα Πατρός:<br />

<input type="text" name="u\_onomapatros" size="30" />

<br />

Όνομα Μητρός:<br />

<input type="text" name="u\_onomamhtros" size="30" />

<br />

Αριθμός Ταυτότητας:<br />

<input type="text" name="u\_taytotitaf" size="30" />

<br />

Τοπική Διεύθυνση:<br />

<input type="text" name="u\_tdiefthinsi" size="30" />

<br />

Μόνιμη Διεύθυνση:<br />

```
<input type="text" name="u_mdiefthinsi" size="30" />
<br />
Τηλέφωνο:<br />
<input type="text" name="u_tilefwno" size="30" />
<br />
Σειρά Επιτυχίας:<br />
<input type="text" name="u_seiraepityxias" size="30" />
<br />
Email:<br />
<input type="text" name="u_email" size="30" />
<br />
Μέσος Όρος:<br />
<input type="text" name="u_mesosoros" size="30"/>
<br />
<br />

<input type="submit" value="Εγγραφή" />

</form>

<br />
<br />
```

```
</body>
</html>
```

### 4.5.3 Καταχώρηση Στοιχείων Καθηγητών

Εικόνα 26. Καταχώρηση Στοιχείων Καθηγητών

Στην παραπάνω σελίδα ο χρήστης βλέπει μια φόρμα ώστε να μπορεί να εισάγει τα απαραίτητα στοιχεία για κάθε καθηγητή. Μόλις ολοκληρώσει την καταγραφή των στοιχείων στην φόρμα πατάει το κουμπί «**Εγγραφή**» και αφού γίνει ένας έλεγχος ότι συμπληρωθήκαν όλα τα πεδία τότε τα στοιχεία αποθηκεύονται στην βάση δεδομένων και βγάζει το μήνυμα ελέγχου «**Έγινε η καταχώρηση**». Αν δεν συμπληρώσει ο χρήστης όλα τα πεδία και πατήσει το κουμπί «**Εγγραφή**» τότε θα εμφανίσει το μήνυμα ελέγχου «**Όλα να πεδία πρέπει να συμπληρωθούν**».

**Ο κώδικας της σελίδας Καταχώρηση Στοιχείων Καθηγητών**

**writingkathigites.php**

```
<html>
<head>
<title>Καταχώρηση Στοιχείων Καθηγητών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
```

*//Σύνδεση με την βάση δεδομένων kathigites.xml*

```
$xml_file = "kathigites.xml";
```

```
if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
```

```
if(isset($_POST["u_onomateponymok"]) && isset($_POST["u_taytotitak"])
&& isset($_POST["u_diefthinsigrafeio"])&& isset($_POST["u_tilefonok"])&&
isset($_POST["u_tilefonopane"])
&& isset($_POST["u_vathmida"])&& isset($_POST["u_tmimaanikei"])){
```

*//Αποθήκευση δεδομένων τα όποια πήραμε με την μέθοδο POST από την φόρμα σε τοπικές μεταβλητές*

```
$u_onomateponymok = stripslashes($_POST['u_onomateponymok']);
$u_taytotitak = stripslashes($_POST['u_taytotitak']);
$u_diefthinsigrafeio = stripslashes($_POST['u_diefthinsigrafeio']);
$u_tilefonok = stripslashes($_POST['u_tilefonok']);
$u_tilefonopane = stripslashes($_POST['u_tilefonopane']);
$u_vathmida = stripslashes($_POST['u_vathmida']);
$u_tmimaanikei = stripslashes($_POST['u_tmimaanikei']);
```

*//Ελεγχος αν συμπληρώθηκαν όλα τα πεδία στην φόρμα*

```
if(!empty($u_onomateponymok)&& !empty($u_taytotitak)&& !empty($u_diefthinsigrafeio)
&& !empty($u_tilefonok)&& !empty($u_tilefonopane)&& !empty($u_vathmida)&&
!empty($u_tmimaanikei)){
```

*//Ελεγχος του προηγούμενου κωδικού καθηγητή και αυτόματη ανάθεση στον αμέσως επόμενο κωδικό καθηγητή με την νέα καταχώρηση στοιχείων καθηγητή*

```
foreach($xml as $kathigitis){
    $last_id = $kathigitis->kodikosk;
}
```

```
$sid = $last_id+1;
```

*//Αποθήκευση των στοιχείων των τοπικών μεταβλητών στις ετικέτες της βάσης δεδομένων*

```
$x = $xml->addChild("kathigitis");
$x->addChild("kodikosk",$sid);
$x->addChild("onomateponymok",$u_onomateponymok);
```



```

    $x->addChild("taytotitak",$u_taytotitak);
    $x->addChild("diefthinsigrafeio",$u_diefthinsigrafeio);
    $x->addChild("tilefonok",$u_tilefonok);
    $x->addChild("tilefonopane",$u_tilefonopane);
    $x->addChild("vathmida",$u_vathmida);
    $x->addChild("tmimaanikei",$u_tmimaanikei);

    $xml->asXML($xml_file);

    echo("<font color='blue'>Έγινε η Καταχώρηση</font>");
    }else {
        echo("<font color='red'>Όλα τα πεδία πρέπει να
συμπληρωθούν.</font>");
    }

}

?>

<br />

```

*//Δημιουργία της φόρμας της σελίδας*

```

<h2>Καταχώρηση Στοιχείων Καθηγητών</h2>

<form action="writingkathigites.php" method="post" name="xml_writing" >

Ονοματεπώνυμο:<br />
<input type="text" name="u_onomateponymok" size="40"/>
<br />
Αριθμός Ταυτότητας:<br />
<input type="text" name="u_taytotitak" size="40"/>
<br />
Διεύθυνση γραφείου:<br />
<input type="text" name="u_diefthinsigrafeio" size="40"/>
<br />
Τηλέφωνο:<br />
<input type="text" name="u_tilefonok" size="40"/>
<br />
Τηλέφωνο Πανεπιστημίου:<br />
<input type="text" name="u_tilefonopane" size="40"/>
<br />
Βαθμίδα:<br />
<input type="text" name="u_vathmida" size="40"/>
<br />
Τμήμα στο οποίο ανήκει:<br />
<input type="text" name="u_tmimaanikei" size="40"/>
<br />
<br >

<input type="submit" value="Εγγραφή" />

</form>

<br />

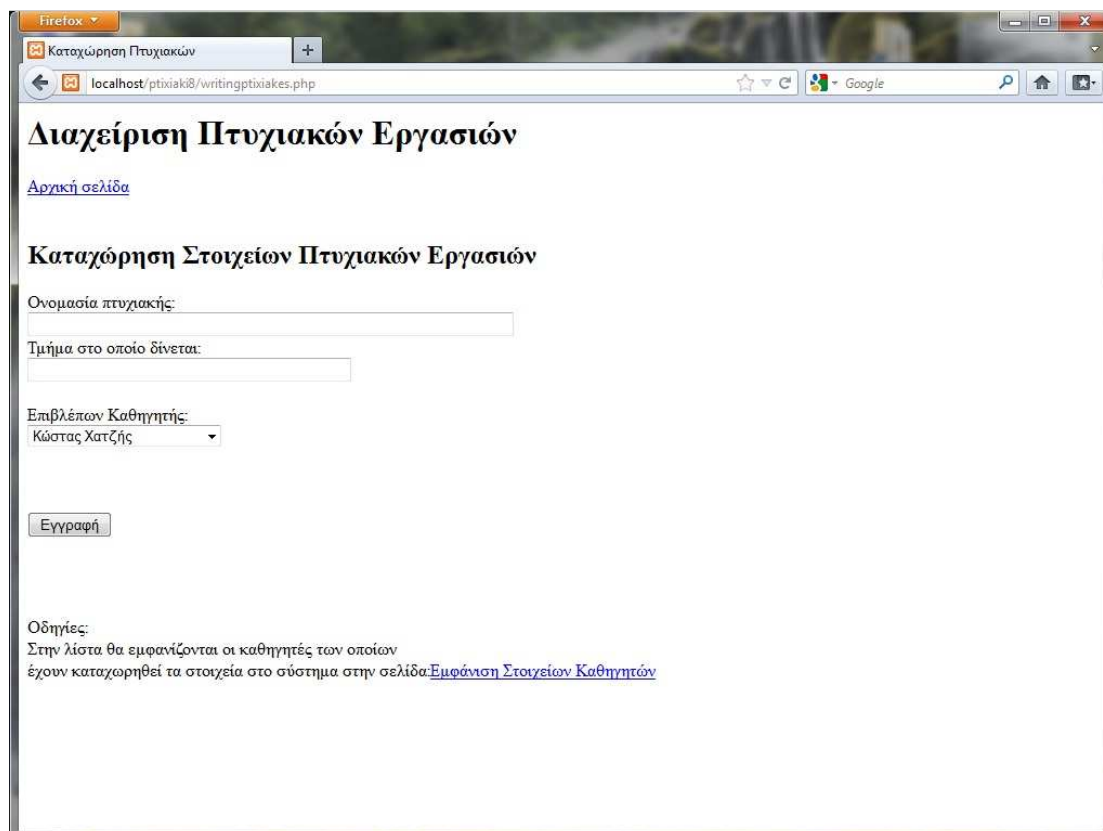
```

<br />

</body>

</html>

#### 4.5.4 Καταχώρηση Στοιχείων Πτυχιακών Εργασιών



Εικόνα 27. Καταχώρηση Στοιχείων Πτυχιακών Εργασιών

Στην παραπάνω σελίδα ο χρήστης βλέπει μια φόρμα ώστε να μπορεί να εισάγει τα απαραίτητα στοιχεία για κάθε πτυχιακή. Στο πεδίο της φόρμας «**Επιβλέπων Καθηγητής**» υπάρχει μια λίστα από την οποία μπορεί να επιλέξει ο χρήστης κάποιον καθηγητή που υπάρχει την βάση δεδομένων για να επιβλέπει την συγκεκριμένη πτυχιακή. Μόλις ολοκληρώσει την καταγραφή των στοιχείων στην φόρμα και επιλέξει καθηγητή από την λίστα πατάει το κουμπί «**Εγγραφή**» και αφού γίνει ένας έλεγχος ότι συμπληρωθήκαν όλα τα πεδία τότε τα στοιχεία αποθηκεύονται στην βάση δεδομένων και βγάζει το μήνυμα ελέγχου «**Έγινε η καταχώρηση**». Αν δεν συμπληρώσει ο χρήστης όλα τα πεδία και πατήσει το κουμπί «**Εγγραφή**» τότε θα εμφανίσει το μήνυμα ελέγχου «**Όλα να πεδία πρέπει να συμπληρωθούν**».

**Ο κώδικας της σελίδας Καταχώρηση Στοιχείων Πτυχιακών Εργασιών**

**writingptixiak.es.php**

```

<html>
<head>
<title>Καταχώρηση Πτυχιακών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
//Σύνδεση με την βάση δεδομένων ptixiak.es.xml
$xml_file = "ptixiak.es.xml";

        if(!$xml=simplexml_load_file($xml_file)){
            trigger_error('Error reading XML file',E_USER_ERROR);
        }
//Σύνδεση με την βάση δεδομένων kathigites.xml , γίνεται επειδή θέλουμε να έχουμε πρόσβαση στα
στοιχεία των καθηγητών και να πάρουμε το ονοματεπώνυμο του καθηγητή που θα επιλέξει ο χρήστης
για να επιβλέπει την συγκεκριμένη πτυχιακή που καταχωρεί και να το αποθηκεύσει στην βάση
ptixiak.es.xml
$xml_file2 = "kathigites.xml";
        if(!$xml2=simplexml_load_file($xml_file2)){
            trigger_error('Error reading XML file',E_USER_ERROR);
        }

        if(isset($_POST["u_onomasia"]) &&
isset($_POST["u_tmima"])&&isset($_POST["u_record2"])){

//Αποθήκευση δεδομένων τα όποια πήραμε με την μέθοδο POST από την φόρμα σε τοπικές
μεταβλητές
        $u_onomasia = stripslashes($_POST['u_onomasia']);
        $u_tmima = stripslashes($_POST['u_tmima']);
//Η μεταβλητή $u_record2 θα παίρνει το ονοματεπώνυμο του καθηγητή που έχει επιλέξει ο χρήστης
από την λίστα για να το αποθηκεύσει στην βάση για τις πτυχιακές στα στοιχεία της συγκεκριμένης
πτυχιακής που έχει καταχωρήσει
        $u_record2 = stripslashes($_POST['u_record2']);

//Εδώ η μεταβλητή παίρνει την τιμή 0 ώστε όταν γίνει ανάθεση Πτυχιακής σε Φοιτητή να ενημερωθεί
με το όνομα του φοιτητή
        $u_onomateponymoptf = 0;

//Εδώ η μεταβλητή παίρνει την τιμή "Δεν έχει ολοκληρωθεί" ώστε όταν ολοκληρωθεί να μπει ο
βαθμός της πτυχιακής στην θέση αυτήν
        $u_vathmospt = "Δεν έχει ολοκληρωθεί";

        if(!empty($u_onomasia)&& !empty($u_tmima)){

//Έλεγχος του προηγούμενου κωδικού πτυχιακής και αυτόματη ανάθεση στον αμέσως επόμενο
κωδικό πτυχιακής με την νέα καταχώρηση στοιχείων πτυχιακής
                foreach($xml as $ptixiaki){
                    $last_id = $ptixiaki->kodikospt;

```

```

    }

    $id = $last_id+1;

```

*//Αποθήκευση των στοιχείων των τοπικών μεταβλητών στις ετικέτες της βάσης δεδομένων*

```

    $x = $xml->addChild("ptixiaki");
    $x->addChild("kodikospt",$id);
    $x->addChild("onomasia",$u_onomasia);
    $x->addChild("tmima",$u_tmima);
    $x->addChild("onomateponymoptk",$u_record2);
    $x->addChild("onomateponymoptf",$u_onomateponymoptf);
    $x->addChild("vathmospt",$u_vathmospt);

    $xml->asXML($xml_file);

    echo ("<font color='blue'>Έγινε η Καταχώρηση</font>");
    }else {
        echo ("<font color='red'>Όλα τα πεδία πρέπει να
συμπληρωθούν.</font>");
    }

}

?>

<br />

```

*//Δημιουργία της φόρμας της σελίδας*

```

<h2>Καταχώρηση Στοιχείων Πτυχιακών Εργασιών</h2>

<form action="writingptixiakes.php" method="post" name="xml_writing" >

Ονομασία πτυχιακής:<br />
<input type="text" name="u_onomasia" size="70"/>
<br />
Τμήμα στο οποίο δίνεται:<br />
<input type="text" name="u_tmima" size="45" />
<br />
<br>
Επιβλέπων Καθηγητής:<br />
<select name="u_record2" >
    <?php

        foreach($xml2 as $kathigitis){
            echo('<option value="" . $kathigitis->onomateponymok . "'>' .
$kathigitis->onomateponymok . '</option>');
        }

    ?>

</select>
<br />
<br>
<br>

```

```
<br>
<input type="submit" value="Εγγραφή" />

</form>
<br />

<p>Οδηγίες:<br />
Στην λίστα θα εμφανίζονται οι καθηγητές των οποίων<br />
έχουν καταχωρηθεί τα στοιχεία στο σύστημα στην σελίδα:<a
href="readingkathigites.php">Εμφάνιση Στοιχείων Καθηγητών</a><br />
<p>

</body>
</html>
```

#### 4.5.5 Εμφάνιση στοιχείων Φοιτητών



Εικόνα 28. Εμφάνιση στοιχείων Φοιτητών

Στην παραπάνω σελίδα ο χρήστης βλέπει όλα τα στοιχεία των φοιτητών που έχουν καταχωρηθεί.

Ο κώδικας της σελίδας Εμφάνιση στοιχείων Φοιτητών

readingfoitites.php

```
<html>
<head>
<title>Εμφάνιση Στοιχείων Φοιτητών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
```

//Σύνδεση με την βάση δεδομένων foitites.xml

```
$xml_file = "foitites.xml";

if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
echo("<h2>Εμφάνιση Στοιχείων Φοιτητών</h2>");
```

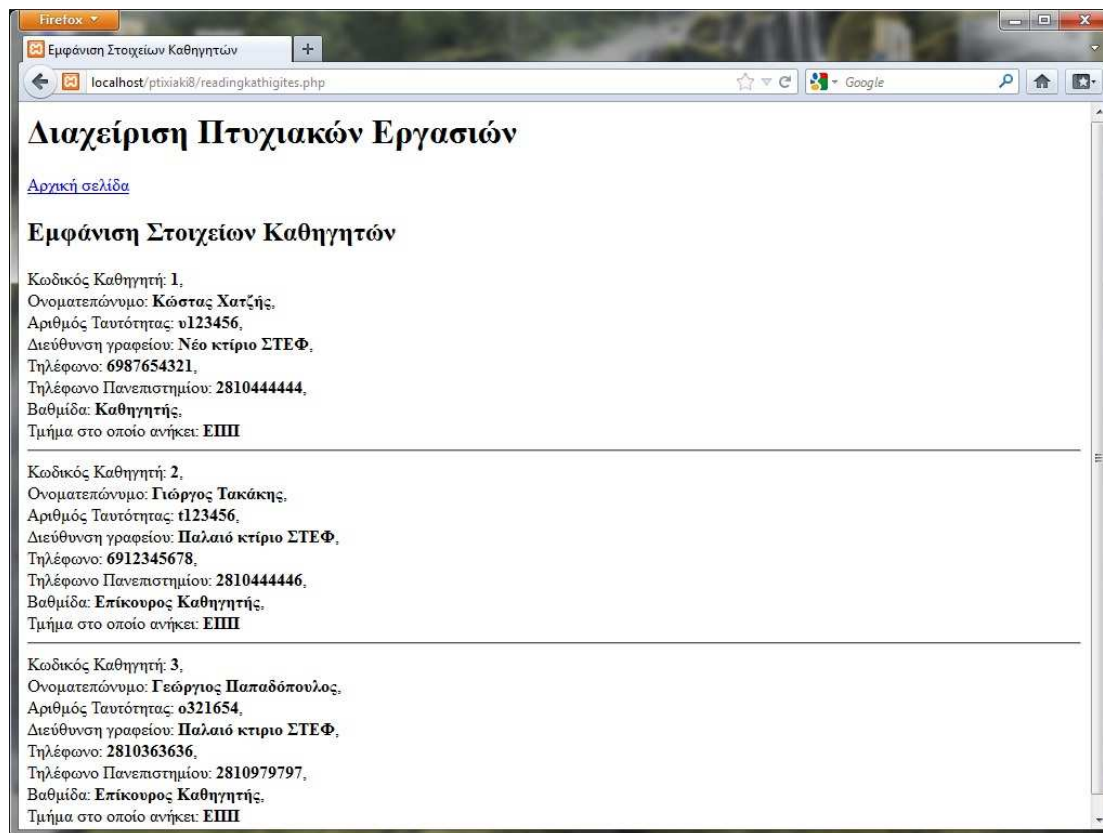
//Με την βοήθεια της συνάρτησης `foreach` εμφανίζονται οι εγγραφές των στοιχείων των φοιτητών στην οθόνη μας

```
foreach($xml as $foititis){
echo 'Αριθμός Μητρώου: <strong>'. $foititis->am.'</strong>,<br>
    Ονοματεπώνυμο: <strong>'. $foititis->onomateponymof.'</strong>,<br>
    Ονομα Πατρός: <strong>'. $foititis->onomapatros.'</strong>,<br>
    Ονομα Μητρός: <strong>'. $foititis->onomamhtros.'</strong>,<br>
    Αριθμός Ταυτότητας: <strong>'. $foititis->taytotitaf.'</strong>,<br>
    Τοπική Διεύθυνση: <strong>'. $foititis->tdiefthinsi.'</strong>,<br>
    Μόνιμη Διεύθυνση: <strong>'. $foititis->mdiefthinsi.'</strong>,<br>
    Τηλέφωνο: <strong>'. $foititis->tilefwno.'</strong>,<br>
    Σειρά Επιτυχίας: <strong>'. $foititis->seiraepityxias.'</strong>,<br>
    Email: <strong>'. $foititis->email.'</strong>,<br>
    Μέσος Ορος: <strong>'. $foititis->mesosoros.'</strong> <br /><hr />;
}

?>
</body>
</html>
```



#### 4.5.6 Εμφάνιση στοιχείων Καθηγητών



Εικόνα 29. Εμφάνιση στοιχείων Καθηγητών

Στην παραπάνω σελίδα ο χρήστης βλέπει όλα τα στοιχεία των καθηγητών που έχουν καταχωρηθεί.

#### Ο κώδικας της σελίδας Εμφάνιση στοιχείων Καθηγητών

[readingkathigites.php](http://localhost/ptixiakid/readingkathigites.php)

```
<html>
<head>
<title>Εμφάνιση Στοιχείων Καθηγητών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
```

*//Σύνδεση με την βάση δεδομένων kathigites.xml*

```
$xml_file = "kathigites.xml";
```

```
    if(!$xml=simplexml_load_file($xml_file)){
        trigger_error('Error reading XML file',E_USER_ERROR);
    }
```

```
echo("<h2>Εμφάνιση Στοιχείων Καθηγητών</h2>");
```

//Με την βοήθεια της συνάρτησης foreach εμφανίζονται οι εγγραφές των στοιχείων των καθηγητών στην οθόνη μας

```
foreach($xml as $kathigitis){
    echo 'Κωδικός Καθηγητή: <strong>'. $kathigitis->kodikosk.'</strong>,<br>
        Ονοματεπώνυμο: <strong>'. $kathigitis->onomateponymok.'</strong>,<br>
        Αριθμός Ταυτότητας: <strong>'. $kathigitis->taytotitak.'</strong>,<br>
        Διεύθυνση γραφείου: <strong>'. $kathigitis->diefthinsigrafeio.'</strong>,<br>
        Τηλέφωνο: <strong>'. $kathigitis->tilefonok.'</strong>,<br>
        Τηλέφωνο Πανεπιστημίου: <strong>'. $kathigitis->tilefonopane.'</strong>,<br>
        Βαθμίδα: <strong>'. $kathigitis->vathmida.'</strong>,<br>
        Τμήμα στο οποίο ανήκει: <strong>'. $kathigitis->tmimaanikei.'</strong> <br /><hr />';
}
?>
</body>
</html>
```

#### 4.5.7 Εμφάνιση όλων των Πτυχιακών



Εικόνα 30.Εμφάνιση όλων των Πτυχιακών

Στην παραπάνω σελίδα ο χρήστης βλέπει όλα τα στοιχεία όλων των πτυχιακών που έχουν καταχωρηθεί.

Ο κώδικας της σελίδας Εμφάνιση όλων των Πτυχιακών

readingptixiakes.php

```
<html>
<head>
<title>Εμφάνιση όλων των Πτυχιακών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
```

//Σύνδεση με την βάση δεδομένων ptixiakes.xml

```
$xml_file = "ptixiakes.xml";
```

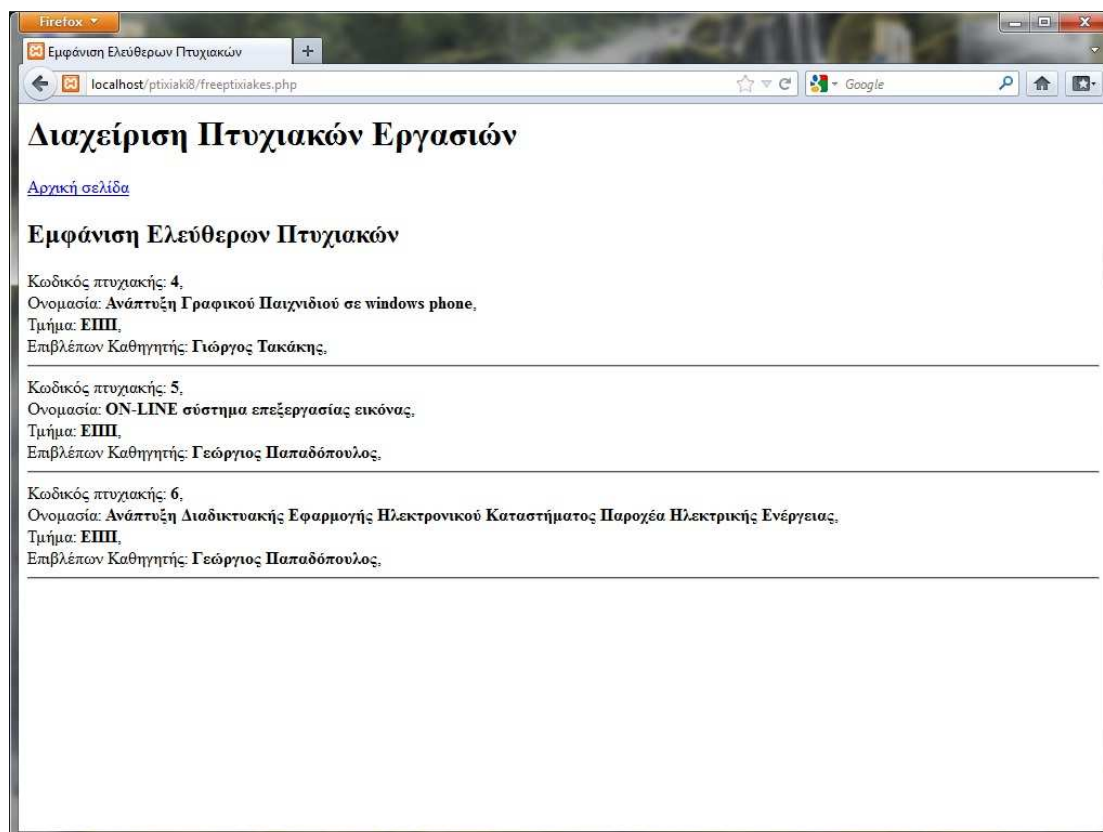
```
if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
echo("<h2>Εμφάνιση όλων των Πτυχιακών</h2>");
```

//Με την βοήθεια της συνάρτησης foreach εμφανίζονται οι εγγραφές των στοιχείων όλων των πτυχιακών στην οθόνη μας

```
foreach($xml as $ptixiaki){
echo 'Κωδικός πτυχιακής: <strong>'.$ptixiaki->kodikospt.'</strong>,<br>
    Ονομασία: <strong>'.$ptixiaki->onomasia.'</strong>, <br>
    Τμήμα: <strong>'.$ptixiaki->tmima.'</strong>,<br>
    Επιβλέπων Καθηγητής: <strong>'.$ptixiaki->onomateponymoptk.'</strong>,<br>
/><hr />';
}

?>
</body>
</html>
```

#### 4.5.8 Εμφάνιση Ελεύθερων Πτυχιακών



Εικόνα 31. Εμφάνιση Ελεύθερων Πτυχιακών

Στην παραπάνω σελίδα ο χρήστης βλέπει όλα τα στοιχεία των πτυχιακών που δεν έχουν ανατεθεί σε φοιτητές.

#### Ο κώδικας της σελίδας Εμφάνιση Ελεύθερων Πτυχιακών

##### freptixiakes.php

```
<html>
<head>
<title>Εμφάνιση Ελεύθερων Πτυχιακών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
```

```
//Σύνδεση με την βάση δεδομένων ptixiakes.xml
```

```
$xml_file = "ptixiakes.xml";
```

```
if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
echo("<h2>Εμφάνιση Ελεύθερων Πτυχιακών</h2>");
```

//Με την βοήθεια της συνάρτησης foreach εμφανίζονται οι εγγραφές των στοιχείων των πτυχιακών στην οθόνη μας

```

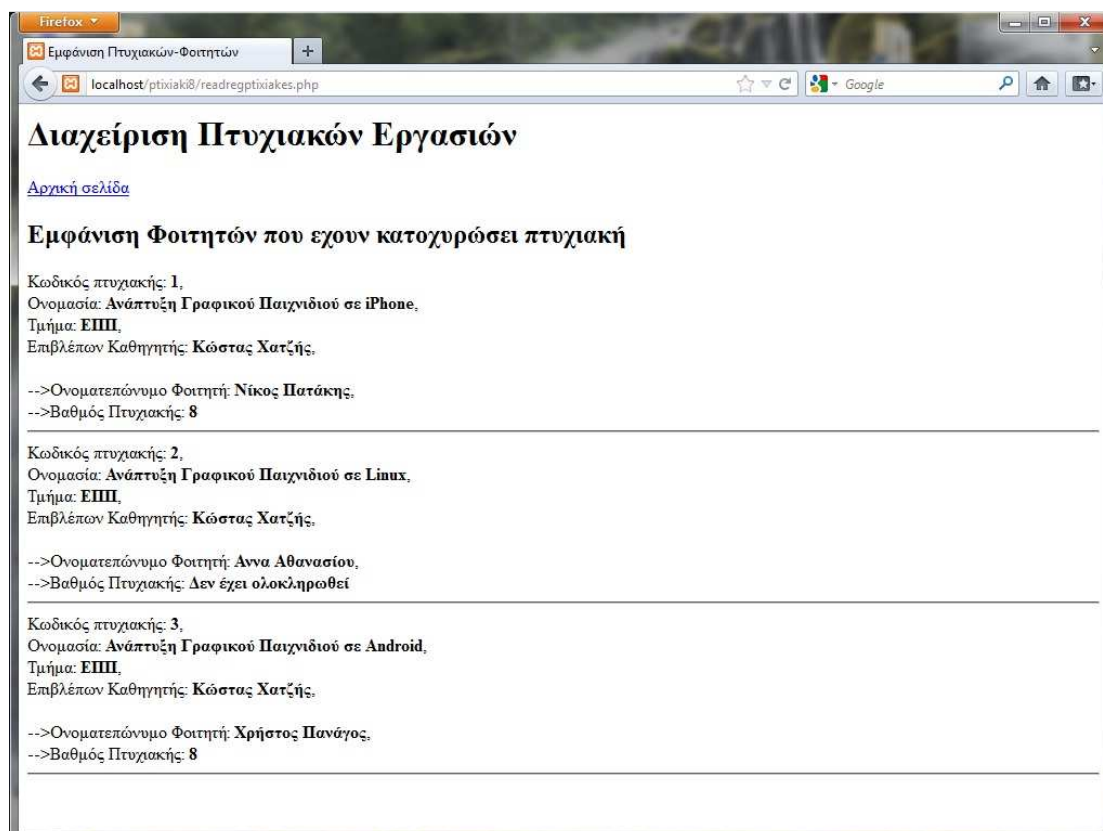
foreach($xml->ptixiaki as $ptixiaki){

// Στο πεδίο onomateponymoptf καταχωρείται το όνομα φοιτητή εφόσον έχει ανατεθεί σε αυτόν το
αντίστοιχο θέμα της πτυχιακής εργασίας . Έτσι με αυτόν τον έλεγχο εμφανίζονται τα στοιχεία μόνο
των πτυχιακών που έχουν την τιμή 0 δηλαδή αυτές που δεν έχουν ανατεθεί σε φοιτητές
    if ($ptixiaki->onomateponymoptf=='0')
        echo 'Κωδικός πτυχιακής: <strong>'. $ptixiaki->kodikospt.'</strong>,<br>
            Ονομασία: <strong>'. $ptixiaki->onomasia.'</strong>, <br>

            Τμήμα: <strong>'. $ptixiaki->tmima.'</strong>,<br>
            Επιβλέπων Καθηγητής: <strong>'. $ptixiaki->
>onomateponymoptk.'</strong>,<br /><hr />';
}
?>
</body>
</html>

```

#### 4.5.9 Εμφάνιση Φοιτητών που έχουν κατοχυρώσει πτυχιακή



Εικόνα 32. Εμφάνιση Φοιτητών που έχουν κατοχυρώσει πτυχιακή

Στην παραπάνω σελίδα ο χρήστης βλέπει όλα τα στοιχεία των πτυχιακών που έχουν ανατεθεί σε φοιτητές και αν την έχουν ολοκληρώσει και τι βαθμό έχουν πάρει.

**Ο κώδικας της σελίδας Εμφάνιση Φοιτητών που έχουν κατοχυρώσει πτυχιακή**

**readregptixiak.es.php**

```

<html>
<head>
<title>Εμφάνιση Πτυχιακών-Φοιτητών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
//Σύνδεση με την βάση δεδομένων ptixiak.es.xml

$xml_file = "ptixiak.es.xml";

    if(!$xml=simplexml_load_file($xml_file)){
        trigger_error('Error reading XML file',E_USER_ERROR);
    }
    echo("<h2>Εμφάνιση Φοιτητών που έχουν κατοχυρώσει πτυχιακή</h2>");
//Με την βοήθεια της συνάρτησης foreach εμφανίζονται οι εγγραφές των στοιχείων των πτυχιακών
στην οθόνη μας

    foreach($xml as $ptixiak){
// Στο πεδίο onomateponymoptf καταχωρείται το όνομα φοιτητή εφόσον έχει ανατεθεί σε αυτόν το
αντίστοιχο θέμα της πτυχιακής εργασίας . Έτσι με αυτόν τον έλεγχο εμφανίζονται τα στοιχεία μόνο
των πτυχιακών που δεν έχουν την τιμή 0 δηλαδή αυτές που έχουν ανατεθεί σε φοιτητές

        if ($ptixiak->onomateponymoptf!='0')
            echo 'Κωδικός πτυχιακής: <strong>'. $ptixiak->kodikospt.'</strong>,<br>

            Ονομασία: <strong>'. $ptixiak->onomasia.'</strong>, <br>
            Τμήμα: <strong>'. $ptixiak->tmima.'</strong>, <br>
            Επιβλέπων Καθηγητής: <strong>'. $ptixiak->onomateponymoptk.'</strong>,<br>
                <br>

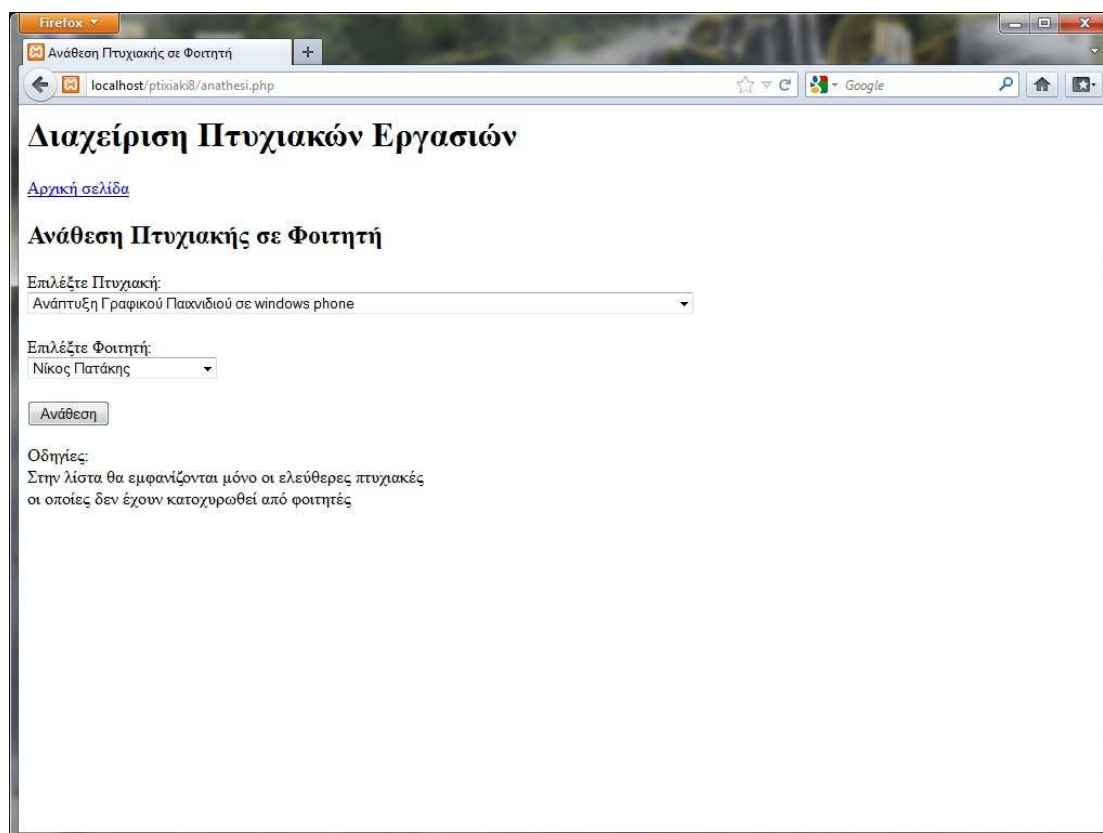
            -->Ονοματεπώνυμο Φοιτητή: <strong>'. $ptixiak->onomateponymoptf.'</strong>,<br>
            -->Βαθμός Πτυχιακής: <strong>'. $ptixiak->vathmospt.'</strong><br /><hr />;
        }

?>
</body>
</html>

```



#### 4.5.10 Ανάθεση Πτυχιακής σε Φοιτητή



Εικόνα 33. Ανάθεση Πτυχιακής σε Φοιτητή

Στην παραπάνω σελίδα ο χρήστης πρέπει να αναθέσει ένα θέμα πτυχιακής σε έναν φοιτητή. Επιλέγει από την λίστα «**Επιλέξτε Πτυχιακή**» θέμα πτυχιακής, η λίστα αυτή εμφανίζει μόνο τις πτυχιακές οι οποίες δεν έχουν κατοχυρωθεί από φοιτητή. Από την επομένη λίστα «**Επιλέξτε Φοιτητή**» ο χρήστης επιλέγει τον φοιτητή που θα ανατεθεί το θέμα της πτυχιακής που έχει επιλέξει από την προηγούμενη λίστα. Έπειτα πατάει το κουμπί «**Ανάθεση**» για να ενημερωθεί η βάση δεδομένων και θα δείξει ο μήνυμα «**Έγινε η Ανάθεση**»..

#### Ο κώδικας της σελίδας Ανάθεση Πτυχιακής σε Φοιτητή

##### [anathesi.php](#)

```
<html>
<head>
<title>Ανάθεση Πτυχιακής σε Φοιτητή</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
//Σύνδεση με την βάση δεδομένων ptixiakes.xml
```

```
$xml_file = "ptixiakes.xml";
if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
```

//Σύνδεση με την βάση δεδομένων foitites.xml , γίνεται επειδή θέλουμε να έχουμε πρόσβαση στα στοιχεία των φοιτητών και να πάρουμε το ονοματεπώνυμο του φοιτητή που θα επιλέξει ο χρήστης για να του ανατεθεί η συγκεκριμένη πτυχιακή και τα στοιχεία του αποθηκεύονται στην βάση ptixiakes.xml

```
$xml_file1 = "foitites.xml";
if(!$xml1=simplexml_load_file($xml_file1)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
if(isset($_POST["u_record"]) &&isset($_POST["u_record1"])){
```

//Αποθήκευση δεδομένων τα όποια πήραμε με την μέθοδο POST από την φόρμα σε τοπικές μεταβλητές

//Η μεταβλητή \$u\_record θα παίρνει την ονομασία της πτυχιακής που έχει επιλέξει ο χρήστης από την λίστα

```
$u_record = stripslashes($_POST['u_record']);
```

//Η μεταβλητή \$u\_record1 θα παίρνει το ονοματεπώνυμο του φοιτητή που έχει επιλέξει ο χρήστης από την λίστα για να το αποθηκεύσει στην βάση για τις πτυχιακές στα στοιχεία της συγκεκριμένης πτυχιακής που θέλει να αναθέσει

```
$u_record1 = stripslashes($_POST['u_record1']);
```

```
$xml_rebuild = "<?xml version=\"1.0\" encoding=\"utf-8\"?>\n<ptixiakes>\n";
```

//Με την συνάρτηση foreach και με την μεταβλητή \$u\_record αφού έχει επιλεγεί η πτυχιακή και πατώντας το κουμπί «**Ανάθεση**» κάνει τις αλλαγές στα στοιχεία της πτυχιακής δηλαδή αλλάζει την τιμή στην ετικέτα onomateponymoptf από 0 που ήταν στην αρχική κατάσταση με το ονοματεπώνυμο του φοιτητή που έχει επιλεγεί

```
foreach($xml as $ptixiaki){
    if($ptixiaki->kodikospt!=$u_record){
        $xml_rebuild .=
            "<ptixiaki>\n
<kodikospt>" . $ptixiaki->kodikospt . "</kodikospt>
<onomasia>" . $ptixiaki->onomasia . "</onomasia>
<tmima>" . $ptixiaki->tmima . "</tmima>
<onomateponymoptk>" . $ptixiaki->onomateponymoptk . "</onomateponymoptk>
<onomateponymoptf>" . $ptixiaki->onomateponymoptf . "</onomateponymoptf>
<vathmospt>" . $ptixiaki->vathmospt . "</vathmospt>
</ptixiaki>\n";
    }
    else{
        $xml_rebuild .=
            "<ptixiaki>\n
<kodikospt>" . $ptixiaki->kodikospt . "</kodikospt>
<onomasia>" . $ptixiaki->onomasia . "</onomasia>
<tmima>" . $ptixiaki->tmima . "</tmima>
```

```
<onmateponymoptk>" . $ptixiaki->onmateponymoptk . "</onmateponymoptk>
```

//Σε αυτό το σημείο γίνεται η αποθήκευση του ονοματεπώνυμου του φοιτητή που έχει επιλεγεί από την λίστα των φοιτητών για να γίνει η ανάθεση με την μεταβλητή \$u\_record1

```
<onmateponymoptf>" . $u_record1 . "</onmateponymoptf>
<vathmospt>" . $ptixiaki->vathmospt . "</vathmospt>
    </ptixiaki>\n";
    }
}

$xml_rebuild .= "</ptixiakes>";

$fh = fopen($xml_file, 'w') or die("can't open file");
fwrite($fh, $xml_rebuild);
fclose($fh);
echo ("<font color='blue'>Έγινε η Ανάθεση</font>");
}
```

?>

```
<h2>Ανάθεση Πτυχιακής σε Φοιτητή</h2>
```

```
<form action="anthesi.php" method="post" name="xml_modifying">
```

```
Επιλέξτε Πτυχιακή:<br />
```

```
<select name="u_record" >
```

```
<?php
```

```
foreach($xml as $ptixiaki){
```

```
if ($ptixiaki->onmateponymoptf=='0')
```

```
echo('<option value="" . $ptixiaki->kodikospt . "">' . $ptixiaki->onomasia . '</option>');
```

```
}
```

```
?>
```

```
</select>
```

```
<br />
```

```
<br>
```

```
Επιλέξτε Φοιτητή:<br />
```

```
<select name="u_record1" >
```

```
<?php
```

```
foreach($xml1 as $foititis){
```

```
echo('<option value="" . $foititis->onmateponymof . "">' . $foititis->onmateponymof . '</option>');
```

```
}
```

```
?>
```

```
</select>
```

```
<br />
```

```
<br>
```

```
<input type="submit" value="Ανάθεση" />
```

```
</form>
```

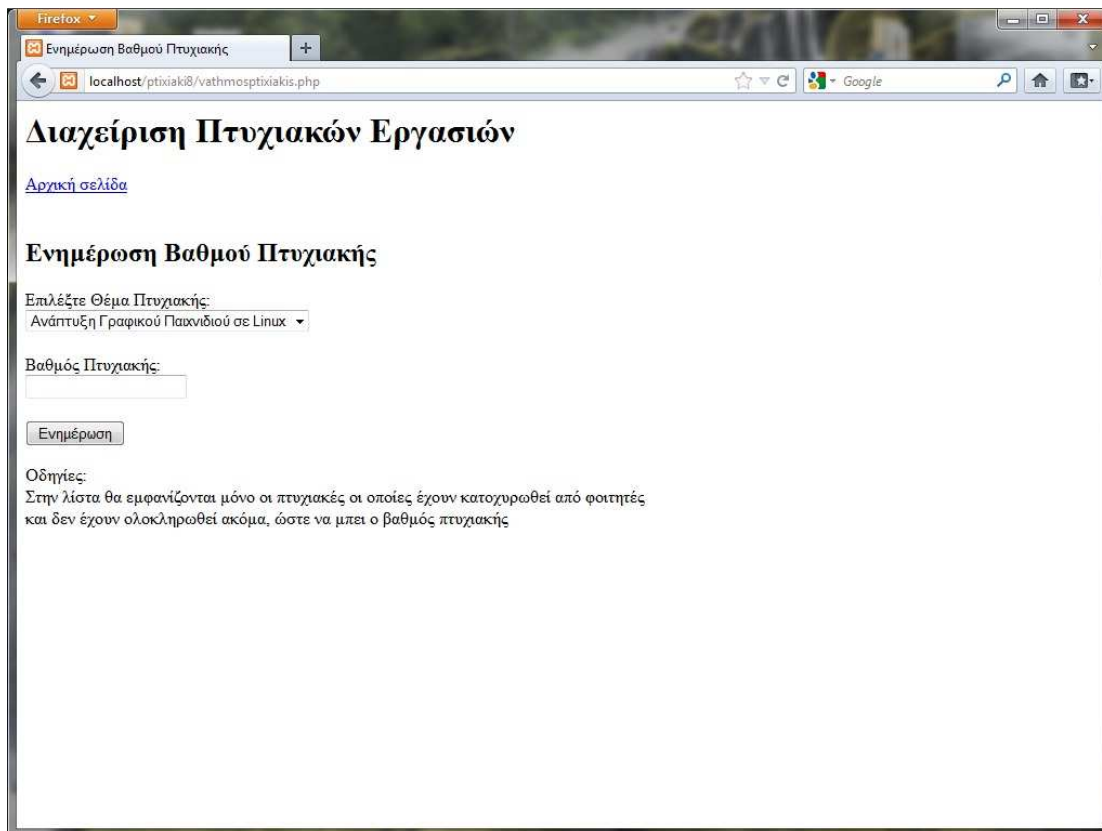
```
<p>Οδηγίες:<br />
```

Στην λίστα θα εμφανίζονται μόνο οι ελεύθερες πτυχιακές  
οι οποίες δεν έχουν κατοχυρωθεί από φοιτητές

</body>

</html>

#### 4.5.11 Ενημέρωση Βαθμού Πτυχιακής



Εικόνα 34. Ενημέρωση Βαθμού Πτυχιακής

Στην παραπάνω σελίδα ο χρήστης θέλει κάνει ενημέρωση βαθμού σε μια πτυχιακή εργασία που έχει ολοκληρωθεί από έναν φοιτητή . Επιλέγει από την λίστα «**Επιλέξτε Θέμα Πτυχιακής**» θέμα πτυχιακής, η λίστα αυτή εμφανίζει μόνο τις πτυχιακές οι οποίες έχουν κατοχυρωθεί σε κάποιον φοιτητή αλλά δεν έχουν ολοκληρωθεί. Στο πεδίο «**Βαθμός Πτυχιακής**» ο χρήστης καταχωρεί τον βαθμό για την πτυχιακή που έχει επιλέξει από την λίστα , έπειτα πατάει το κουμπί «**Ενημέρωση**» και αποθηκεύονται οι αλλαγές βγάζοντας το μήνυμα ελέγχου «**Ενημερώθηκε ο βαθμός πτυχιακής**» . Αν ο χρήστης κάνει λάθος και δεν συμπληρώσει το πεδίο «**Βαθμός Πτυχιακής**» τότε θα βγει το μήνυμα ελέγχου «**Όλα τα πεδία πρέπει να συμπληρωθούν**».

## Ο κώδικας της σελίδας Ενημέρωση Βαθμού Πτυχιακής

### vathmosptixiakis.php

```
<html>
<head>
<title>Ενημέρωση Βαθμού Πτυχιακής</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
```

*//Σύνδεση με την βάση δεδομένων ptixiakis.xml*

```
$xml_file = "ptixiakis.xml";
if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
if(isset($_POST["u_record"]) &&isset($_POST["u_vathmos"])){
```

*//Αποθήκευση δεδομένων τα οποία πήραμε με την μέθοδο POST από την φόρμα σε τοπικές μεταβλητές*

*//Η μεταβλητή \$u\_record θα παίρνει την ονομασία της πτυχιακής που έχει επιλέξει ο χρήστης από την λίστα*

```
$u_record = stripslashes($_POST['u_record']);
$u_vathmos = stripslashes($_POST['u_vathmos']);
```

```
if(!empty($u_vathmos)){
```

```
$xml_rebuild = "<?xml version='1.0' encoding='utf-8'?'>\n<ptixiakis>\n";
```

*//Με την συνάρτηση foreach και με την μεταβλητή \$u\_record αφού έχει επιλεγεί η πτυχιακή και καταχωρηθεί η τιμή στο πεδίο «Βαθμός Πτυχιακής» πατώντας το κουμπί «Ενημέρωση» κάνει τις αλλαγές στα στοιχεία της πτυχιακής, δηλαδή αλλάζει την τιμή στην ετικέτα vathmospt από το «Δεν έχει ολοκληρωθεί» που ήταν στην αρχική κατάσταση με την τιμή του βαθμού που έχει καταχωρηθεί στην φόρμα*

```
foreach($xml as $ptixiaki){
```

```
    if($ptixiaki->kodikospt!=$u_record){
```

```
        $xml_rebuild .=
```

```
        "<ptixiaki>\n
```

```
        <kodikospt>" . $ptixiaki->kodikospt . "</kodikospt>
```

```
        <onomasia>" . $ptixiaki->onomasia . "</onomasia>
```

```
        <tmima>" . $ptixiaki->tmima . "</tmima>
```

```
        <onomateponymoptk>" . $ptixiaki->onomateponymoptk . "</onomateponymoptk>
```

```
        <onomateponymoptf>" . $ptixiaki->onomateponymoptf . "</onomateponymoptf>
```

```
        <vathmospt>" . $ptixiaki->vathmospt . "</vathmospt>
```

```
        </ptixiaki>\n";
```

```
    }
```

```
    else{
```

```
        $xml_rebuild .=
```

```

        "<ptixiaki>\n
<kodikospt>" . $ptixiaki->kodikospt . "</kodikospt>
<onomasia>" . $ptixiaki->onomasia . "</onomasia>
<tmima>" . $ptixiaki->tmima . "</tmima>
<onomateponymoptk>" . $ptixiaki->onomateponymoptk . "</onomateponymoptk>
<onomateponymoptf>" . $ptixiaki->onomateponymoptf . "</onomateponymoptf>
<vathmospt>" . $u_vathmos . "</vathmospt>
        </ptixiaki>\n";
    }
}

$xml_rebuild .= "</ptixiakes>";

$fh = fopen($xml_file, 'w') or die("can't open file");
fwrite($fh, $xml_rebuild);
fclose($fh);

echo ("<font color='blue'>Ενημερώθηκε ο Βαθμός Πτυχιακής</font>");
} else {
        echo ("<font color='red'>Όλα τα πεδία πρέπει να
συμπληρωθούν.</font>");
}
}
?>

<br />
<h2>Ενημέρωση Βαθμού Πτυχιακής</h2>

<form action="vathmosptixiakis.php" method="post" name="xml_modifying">

Επιλέξτε Θέμα Πτυχιακής:<br />
<select name="u_record" >
    <?php

        foreach($xml as $ptixiaki){
if($ptixiaki->onomateponymoptf!='0' && $ptixiaki->vathmospt=="Δεν έχει ολοκληρωθεί")

echo('<option value="" . $ptixiaki->kodikospt . "">' . $ptixiaki->onomasia . '</option>');
        }

    ?>
</select>
<br />
<br>
Βαθμός Πτυχιακής:<br />
<input type="text" name="u_vathmos" />
<br />
<br>
<input type="submit" value="Ενημέρωση" />
</form>

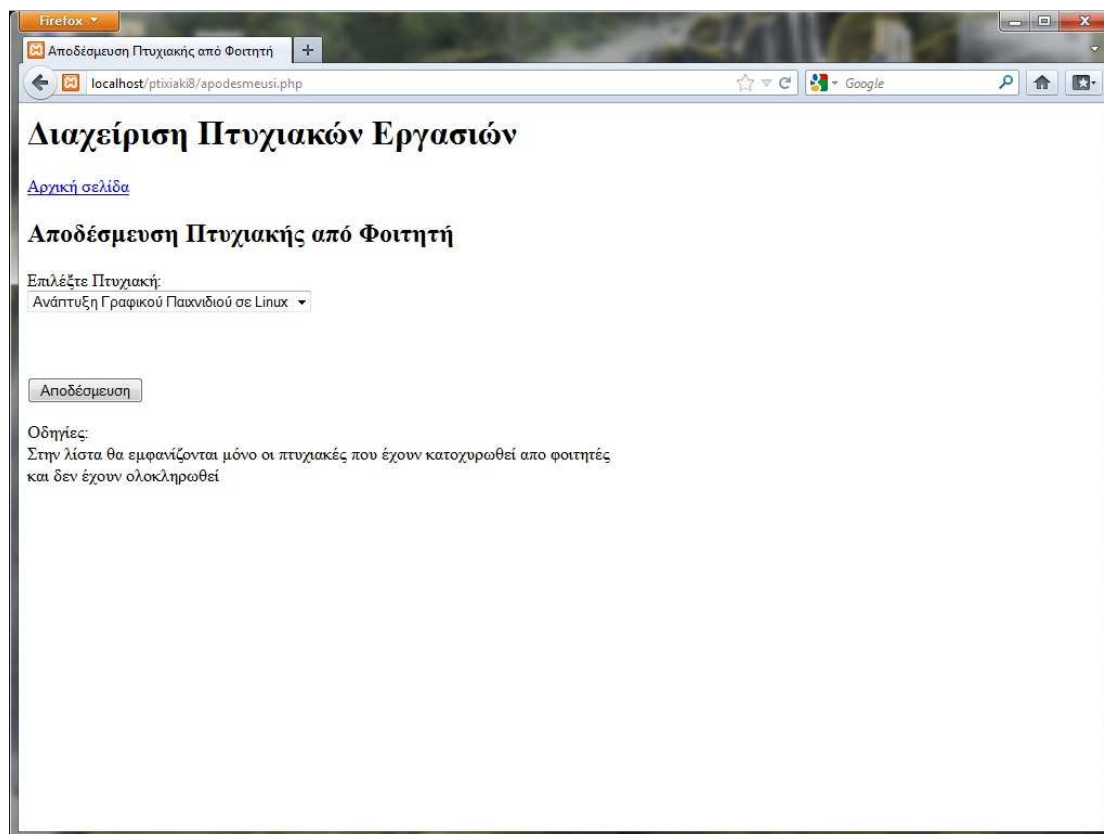
<p>Οδηγίες:<br />
Στην λίστα θα εμφανίζονται μόνο οι πτυχιακές οι οποίες έχουν κατοχυρωθεί
από φοιτητές<br />
και δεν έχουν ολοκληρωθεί ακόμα, ώστε να μπει ο βαθμός πτυχιακής<p>

```



```
</body>  
</html>
```

#### 4.5.12 Αποδέσμευση Πτυχιακής από Φοιτητή



Εικόνα 35 . Αποδέσμευση Πτυχιακής από Φοιτητή

Στην παραπάνω σελίδα ο χρήστης θέλει κάνει αποδέσμευση σε μια πτυχιακή εργασία που δεν έχει ολοκληρωθεί από έναν φοιτητή ώστε έπειτα να μπορεί να ανατεθεί σε άλλον φοιτητή. Επιλέγει από την λίστα «**Επιλέξτε Πτυχιακή**» θέμα πτυχιακής, η λίστα αυτή εμφανίζει μόνο τις πτυχιακές οι οποίες έχουν κατοχυρωθεί σε κάποιον φοιτητή αλλά δεν έχουν ολοκληρωθεί. Έπειτα πατάει το κουμπί «**Αποδέσμευση**» και αφού ολοκληρωθεί η ενέργεια με επιτυχία τότε εμφανίζεται το μήνυμα «**Έγινε η Αποδέσμευση**».

## Ο κώδικας της σελίδας Αποδέσμευση Πτυχιακής από Φοιτητή

### apodesmeusi.php

```

<html>
<head>
<title>Αποδέσμευση Πτυχιακής από Φοιτητή</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php

    $k = 0;
    //Σύνδεση με την βάση δεδομένων ptixiakes.xml

    $xml_file = "ptixiakes.xml";
    if(!$xml=simplexml_load_file($xml_file)){
        trigger_error('Error reading XML file',E_USER_ERROR);
    }
    if(isset($_POST["u_record"] )){
    //Αποθήκευση δεδομένων τα όποια πήραμε με την μέθοδο POST από την φόρμα σε τοπικές
    μεταβλητές

    //Η μεταβλητή $u_record θα παίρνει την ονομασία της πτυχιακής που έχει επιλέξει ο χρήστης από την
    λίστα

        $u_record = stripslashes($_POST['u_record']);

        $xml_rebuild = "<?xml version=\\"1.0\\" encoding=\\"utf-8\\"?>\n<ptixiakes>\n";
        //Με την συνάρτηση foreach και με την μεταβλητή $u_record αφού έχει επιλεγεί η πτυχιακή,
        πατώντας το κουμπί «Αποδέσμευση» κάνει τις αλλαγές στα στοιχεία της πτυχιακής, δηλαδή αλλάζει
        την τιμή στην ετικέτα onomateponymoptf από το ονοματεπώνυμο του φοιτητή που περιείχε πριν ,με
        την τιμή 0 η οποία βρίσκεται στην μεταβλητή $k

        foreach($xml as $ptixiaki){
            if($ptixiaki->kodikospt!=$u_record){
                $xml_rebuild .=
                "<ptixiaki>\n
                <kodikospt>" . $ptixiaki->kodikospt . "</kodikospt>
                <onomasia>" . $ptixiaki->onomasia . "</onomasia>
                <tmima>" . $ptixiaki->tmima . "</tmima>
                <onomateponymoptk>" . $ptixiaki->onomateponymoptk . "</onomateponymoptk>
                <onomateponymoptf>" . $ptixiaki->onomateponymoptf . "</onomateponymoptf>
                <vathmospt>" . $ptixiaki->vathmospt . "</vathmospt>
                </ptixiaki>\n";
            }
            else{
                $xml_rebuild .=

```

```

        "<ptixiaki>\n
<kodikospt>" . $ptixiaki->kodikospt . "</kodikospt>
<onomasia>" . $ptixiaki->onomasia . "</onomasia>
<tmima>" . $ptixiaki->tmima . "</tmima>
<onomateponymoptk>" . $ptixiaki->onomateponymoptk . "</onomateponymoptk>
<onomateponymoptf>" . $k . "</onomateponymoptf>
<vathmospt>" . $ptixiaki->vathmospt . "</vathmospt>
        </ptixiaki>\n";
    }
}

$xml_rebuild .= "</ptixiakes>";

$fh = fopen($xml_file, 'w') or die("can't open file");
fwrite($fh, $xml_rebuild);
fclose($fh);
echo ("<font color='blue'>Έγινε η Αποδέσμευση</font>");
}
?>

<h2>Αποδέσμευση Πτυχιακής από Φοιτητή</h2>

<form action="apodesmeusi.php" method="post" name="xml_modifying">

Επιλέξτε Πτυχιακή:<br />
<select name="u_record" >
    <?php

        foreach($xml as $ptixiaki){
            if ($ptixiaki->onomateponymoptf!='0' && $ptixiaki->vathmospt=="Δεν έχει
ολοκληρωθεί")
                echo('<option value="" . $ptixiaki->kodikospt . "'>' . $ptixiaki-
>onomasia . '</option>');
        }

    ?>
</select>
<br />
<br>

<br />
<br>
<input type="submit" value="Αποδέσμευση" />

</form>

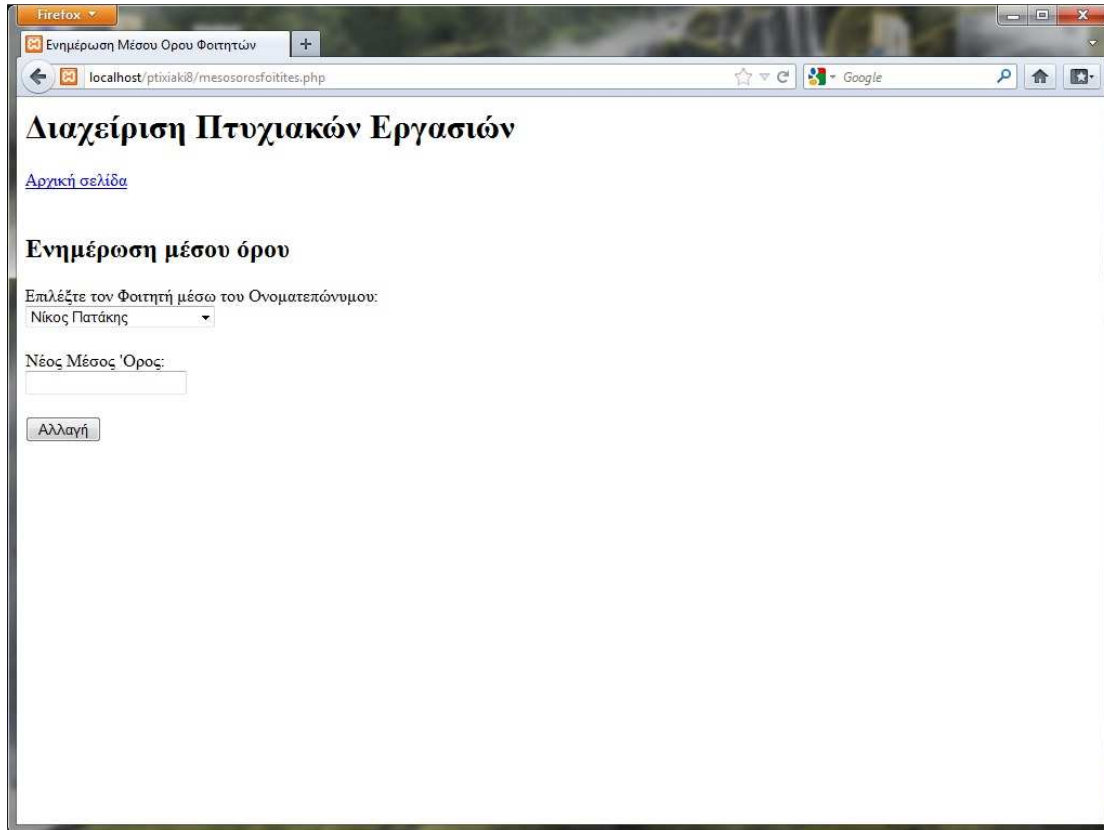
<p>Οδηγίες:<br />
Στην λίστα θα εμφανίζονται μόνο οι πτυχιακές που έχουν κατοχυρωθεί απο
φοιτητές <br />
και δεν έχουν ολοκληρωθεί<p>

</body>

```

</html>

#### 4.5.13 Ενημέρωση μέσου όρου Φοιτητών



Εικόνα 36 . Ενημέρωση μέσου όρου Φοιτητών

Στην παραπάνω σελίδα ο χρήστης θέλει κάνει ενημέρωση μέσου όρου σε έναν φοιτητή . Επιλέγει από την λίστα «**Επιλέξτε τον Φοιτητή μέσω του Ονοματεπώνυμου**» τον φοιτητή που θέλει και στο πεδίο «**Νέος Μέσος Όρος**» καταχωρεί τον νέο βαθμό έπειτα πατάει το κουμπί «**Αλλαγή**» και αποθηκεύονται οι αλλαγές βγάζοντας το μήνυμα ελέγχου «**Ενημερώθηκε ο νέος Μέσος Όρος**». Αν ο χρήστης κάνει λάθος και δεν συμπληρώσει το πεδίο «**Νέος Μέσος Όρος**» τότε θα βγει το μήνυμα ελέγχου «**Όλα τα πεδία πρέπει να συμπληρωθούν**» .

**Ο κώδικας της σελίδας Ενημέρωση μέσου όρου Φοιτητών**

**mesosorosfoitites.php**

```

<html>
<head>
<title>Ενημέρωση Μέσου Όρου Φοιτητών</title>
<h1>Διαχείριση Πτυχιακών Εργασιών</h1>
</head>
<body>
<a href="index.html">Αρχική σελίδα</a><br />
<?php
//Σύνδεση με την βάση δεδομένων foitites.xml

$xml_file = "foitites.xml";
if(!$xml=simplexml_load_file($xml_file)){
    trigger_error('Error reading XML file',E_USER_ERROR);
}
if(isset($_POST["u_record"]) &&isset($_POST["u_mesosoros"])){
//Αποθήκευση δεδομένων τα οποία πήραμε με την μέθοδο POST από την φόρμα σε τοπικές
μεταβλητές

//Η μεταβλητή $u_record θα παίρνει το ονοματεπώνυμο του φοιτητή που έχει επιλέξει ο χρήστης από
την λίστα

    $u_record = stripslashes($_POST['u_record']);
    $u_mesosoros = stripslashes($_POST['u_mesosoros']);

    if(!empty($u_mesosoros)){

        $xml_rebuild = "<?xml version=\"1.0\" encoding=\"utf-8\"?>\n<foitites>\n";

//Με την συνάρτηση foreach και με την μεταβλητή $u_record αφού έχει επιλεγεί το ονοματεπώνυμο
του φοιτητή και καταχωρηθεί η τιμή στο πεδίο «Νέος Μέσος Όρος» πατώντας το κουμπί «Αλλαγή»
κάνει τις αλλαγές στα στοιχεία του φοιτητή, δηλαδή καταχωρεί την τιμή του νέου βαθμού στην
ετικέτα mesosoros

        foreach($xml as $foititis){
            if($foititis->am!=$u_record){
                $xml_rebuild .=
                    "<foititis>\n
                    <am>" . $foititis->am . "</am>
                    <onomateponymof>" . $foititis->onomateponymof . "</onomateponymof>
                    <onomapatros>" . $foititis->onomapatros . "</onomapatros>
                    <onomamhtros>" . $foititis->onomamhtros . "</onomamhtros>
                    <taytotitaf>" . $foititis->taytotitaf . "</taytotitaf>
                    <tdiefthinsi>" . $foititis->tdiefthinsi . "</tdiefthinsi>

                    <mdiefthinsi>" . $foititis->mdiefthinsi . "</mdiefthinsi>

                    <tilefwno>" . $foititis->tilefwno . "</tilefwno>
                    <seiraepityxias>" . $foititis->seiraepityxias . "</seiraepityxias>

                    <email>" . $foititis->email . "</email>

```

```

<mesosoros>" . $foititis->mesosoros . "</mesosoros>

        </foititis>\n";
    }
    else{
        $xml_rebuild .=
            "<foititis>\n
<am>" . $foititis->am . "</am>
<onmateponymof>" . $foititis->onmateponymof . "</onmateponymof>
<onomapatros>" . $foititis->onomapatros . "</onomapatros>
<onomamhtros>" . $foititis->onomamhtros . "</onomamhtros>
<taytotitaf>" . $foititis->taytotitaf . "</taytotitaf>
<tdiefthinsi>" . $foititis->tdiefthinsi . "</tdiefthinsi>
<mdiefthinsi>" . $foititis->mdiefthinsi . "</mdiefthinsi>
<tilefwno>" . $foititis->tilefwno . "</tilefwno>
<seiraepityxias>" . $foititis->seiraepityxias . "</seiraepityxias>

<email>" . $foititis->email . "</email>
<mesosoros>" . $_mesosoros . "</mesosoros>
        </foititis>\n";
    }
}

$xml_rebuild .= "</foitites>";

$fh = fopen($xml_file, 'w') or die("can't open file");
fwrite($fh, $xml_rebuild);
fclose($fh);
echo("<font color='blue'>Ενημερώθηκε ο νέος Μέσος Όρος</font>");
}else {
        echo("<font color='red'>Όλα τα πεδία πρέπει να
συμπληρωθούν.</font>");
    }
}

?>

<br />
<h2>Ενημέρωση μέσου όρου</h2>

<form action="mesosorosfoitites.php" method="post" name="xml_modifying">

Επιλέξτε τον Φοιτητή μέσω του Ονοματεπώνυμου:<br />
<select name="u_record" >
    <?php
        foreach($xml as $foititis){
            echo('<option value="" . $foititis->am . "">' . $foititis-
>onmateponymof . '</option>');
        }
    }

?>

</select>
<br />
<br>

```

```
    Νέος Μέσος Όρος:<br />
    <input type="text" name="u_mesosoros" />
    <br />
    <br>
    <input type="submit" value="Αλλαγή" />
  </form>
</body>
</html>
```



## Κεφάλαιο 5

### 5.1 Αποτελέσματα - Συμπεράσματα

Η πτυχιακή εργασία με θέμα «Διαδικτυοκεντρικό Σύστημα για on-line διαχείριση των πτυχιακών με XML» είναι μια εφαρμογή για την διαχείριση των πτυχιακών εργασιών ενός Εκπαιδευτικού Ιδρύματος. Είναι μια δυναμική ιστοσελίδα η οποία αποθηκεύει και επεξεργάζεται δεδομένα μέσω μιας βάσης δεδομένων, η οποία υλοποιήθηκε με XML .Η διασύνδεση της βάσης δεδομένων με την εφαρμογή και οι λειτουργίες οι οποίες υποστηρίζονται, υλοποιήθηκαν με την γλώσσα PHP. Βασικό κριτήριο είναι η εφαρμογή μας να είναι εύκολη και κατανοητή προς τον χρήστη. Η εφαρμογή θα βοηθήσει να ελέγχουμε καλύτερα τα δεδομένα τα οποία αποθηκεύουμε και να τα διαχειριζόμαστε ευκολότερα.

Το βασικό κομμάτι της εργασίας ήταν η μελέτη και η χρήση της γλώσσας XML. Η XML είναι μια σχετικά νέα και ταχύτατα αναπτυσσόμενη τεχνολογία που έχει καθιερωθεί σαν πρότυπο για τη μοντελοποίηση και επεξεργασία των δεδομένων που διακινούνται στο Internet. Περιλαμβάνει αρκετά πρότυπα και λογισμικό που επιτρέπουν τη χρήση της σε ευρεία κλίμακα. Παίζει πρωταγωνιστικό ρόλο στη σύγχρονη μορφή του διαδικτύου ενοποιώντας τις web-based εφαρμογές και συμβάλλοντας στη δημιουργία ενός επιπέδου μετά-γνώσης (semantic web). Τα επιτεύγματα της XML είναι ότι καθιέρωσε ένα κοινό format στη μοντελοποίηση των web δεδομένων, εισήγαγε στη πράξη την έννοια της αυτοπεριγραφής σε ένα block δεδομένων (meta-data), τα δεδομένα πλέον δεν είναι κατανοητά μόνο από υπολογιστές, αλλά προορίζονται και για χρήστες, καθιέρωσε ένα κοινό πρότυπο στην επεξεργασία και ερμηνεία των δεδομένων από τις εφαρμογές του Internet.

Η XML είναι μια γλώσσα περιγραφής δεδομένων τα οποία είναι εύκολο να διαβαστούν και να επεξεργαστούν από ανθρώπους και προγράμματα. Δεν επιβάλλει κανέναν σημασιολογικό περιορισμό στα δεδομένα που περιγράφει. Επίσης το μέλλον του διαδικτύου είναι ο σημασιολογικός ιστός, ενός διαδικτύου δηλαδή στο οποίο η πληροφορία είναι καλύτερα ορισμένη επιτρέποντας με αυτόν τον τρόπο την ολοκλήρωση, αυτοματοποίηση και επαναχρησιμοποίηση των δεδομένων. Η μια από τις γλώσσες που στηρίζεται ο σημασιολογικός ιστός είναι η XML, άρα η XML θα έχει αρκετό μέλλον.

Αποκόμισα γενικές γνώσεις πάνω στην κατασκευή μιας δυναμικής ιστοσελίδας με προοπτική για περαιτέρω ανάπτυξη και εκπαίδευση πάνω στην κατασκευή ιστοσελίδων. Αυτή η πτυχιακή εργασία σίγουρα ωφέλησε εμένα και θα μπορέσει να βοηθήσει νέους web developers που αναπτύσσουν εφαρμογές PHP από το μηδέν. Η συνολική εργασία είναι ένα αποτέλεσμα δημιουργημένο εξολοκλήρου από κώδικα. Με την βοήθεια του ελεύθερου πακέτου λογισμικού XAMPP εγκατέστησα την PHP και δούλεψα τον web server Apache ο οποίος είναι πολύτιμο εργαλείο για φιλοξενία ιστοσελίδων.

### 5.2 Επεκτάσεις

Μελλοντικά με την δομή, με βάση την κεντρική ιδέα που έχει η εφαρμογή και σύμφωνα με τις μεγάλες δυνατότητες της PHP και XML θα μπορούσε να επεκταθεί και να τροποποιηθεί αρκετά όπως θα μας ζητηθεί. Συγκεκριμένα θα μπορούσε να αποτελεί ένα τμήμα του συστήματος όπου κάνουν ανανέωση και δήλωση μαθημάτων οι φοιτητές και μπορούν να βλέπουν την αναλυτική βαθμολογία τους. Οι φοιτητές θα μπορούσαν να έχουν πρόσβαση στο κομμάτι της εφαρμογής της Διαχείρισης Πτυχιακών Εργασιών όπου εμφανίζονται οι ελεύθερες πτυχιακές προς κατοχύρωση και να έκαναν αίτηση μέσα από την εφαρμογή στον αντίστοιχο επιβλέποντα καθηγητή για κατοχύρωση του αντίστοιχου θέματος πτυχιακής εργασίας.

## Βιβλιογραφία


- [1] XML ΒΗΜΑ ΒΗΜΑ [Author] Michael J. Young 1<sup>st</sup> Edition 2000 by Microsoft Corporation Εκδόσεις Κλειδάριθμος
- [2] <http://en.wikipedia.org/>
- [3] <http://el.wikipedia.org/wiki>
- [4] <http://www.apache.org>
- [5] <http://www.apachefriends.org/en/index.html>
- [6] <http://notepad-plus-plus.org/>
- [7] <http://www.w3schools.com/>
- [8] <http://www.php.net/>
- [9] <http://pc-news.gr/tag/web-3.0.html>
- [10] <http://dsmc.eap.gr/semweb.php>
- [11] <http://el.wikipedia.org/wiki/HTML>
- [12] <http://el.wikipedia.org/wiki/PHP>
- [13] <http://www.techteam.gr/wiki/PHP>
- [14] <http://mytwocents.gr/apache-web-server/>
- [15] <http://debuggable.com/posts/parsing-xml-using-simplexml>
- [16] [http://www.w3schools.com/php/php\\_xml\\_simplexml.asp](http://www.w3schools.com/php/php_xml_simplexml.asp)

Παράρτημα

Διαφάνειες Παρουσίασης

 <p>Τμήμα Εφαρμοσμένης Πληροφορικής &amp; Πολυμέσων</p> <h2>Πτυχιακή Εργασία</h2> <p>Διαδίκτυοκεντρικό Σύστημα για on-line διαχείριση των πτυχιακών με XML</p> <p>Σπουδαστής: Κρικιώνης Νικόλαος (AM1282) Εισηγητής: Παπαδάκης Νικόλαος</p>	<h2>Εισαγωγή</h2> <p><b>Σκοπός</b></p> <p>Σκοπός της παρούσας πτυχιακής είναι η δημιουργία μιας αξιόπιστης εφαρμογής η οποία να είναι εύκολη και κατανοητή προς τον χρήστη και αποδίδει τις σημαντικές διεργασίες για την Διαχείριση των Πτυχιακών Εργασιών ενός Εκπαιδευτικού Ιδρύματος.</p> <p><b>Στόχος</b></p> <p>Στόχος είναι να κάνει κάποιες βασικές λειτουργίες αποτελεσματικότερα και πιο γρήγορα ως προς την καταχώρηση, διαχείριση και οργάνωση των πτυχιακών εργασιών.</p>
<h2>Βασικές λειτουργίες εφαρμογής</h2> <ul style="list-style-type: none"> <li>• Καταχώριση Στοιχείων Φοιτητών</li> <li>• Καταχώριση Στοιχείων Καθηγητών</li> <li>• Καταχώριση Στοιχείων Πτυχιακών</li> <li>• Εμφάνιση Στοιχείων Καθηγητών</li> <li>• Εμφάνιση Στοιχείων Φοιτητών</li> <li>• Εμφάνιση Στοιχείων όλων των Πτυχιακών</li> <li>• Εμφάνιση Ελεύθερων Πτυχιακών</li> <li>• Εμφάνιση Φοιτητών που έχουν κατοχυρώσει θέμα Πτυχιακής και Βαθμό αν την έχουν ολοκληρώσει</li> <li>• Ανάθεση Πτυχιακής σε Φοιτητή</li> <li>• Ενημέρωση Βαθμού Πτυχιακής</li> <li>• Αποδέσμευση Πτυχιακής από Φοιτητή</li> <li>• Ενημέρωση μέσου όρου Φοιτητών</li> </ul>	<h2>Μέθοδος Ανάλυσης και Ανάπτυξης</h2> <ul style="list-style-type: none"> <li>• Η ανάπτυξη Πληροφοριακού Συστήματος για διαχείριση των πτυχιακών είναι μια δυναμική ιστοσελίδα η οποία αντλεί, αποθηκεύει και επεξεργάζεται πληροφορίες σε μία βάση δεδομένων.</li> <li>• Η εκτέλεση της εφαρμογής γίνεται σε ένα web server. Έτσι χρειαζόμαστε ανάπτυξη βάσης δεδομένων και παράλληλα ανάπτυξη λογισμικού για την άνεληση πληροφοριών από τη βάση μεθώς και την αποθήκευσή τους σε αυτή.</li> </ul>
<h2>Δυναμική ιστοσελίδα</h2> <p><b>Ορισμός</b></p> <p>Σε αντίθεση με μια στατική ιστοσελίδα, δεν είναι απλό HTML έγγραφο, αλλά συμπεριλαμβάνει προγραμματισμό και σύνδεση με βάση δεδομένων</p> <p><b>Πλεονεκτήματα</b></p> <ul style="list-style-type: none"> <li>• Απλότητα</li> <li>• Ταχεία εφαρμογή</li> <li>• Ευκολία στη διαχείριση</li> <li>• Συνεχής ενημέρωση περιεχομένου</li> <li>• Μειωμένο κόστος συντήρησης</li> </ul>	<h2>Τεχνολογίες και εργαλεία ανάπτυξης</h2> <ul style="list-style-type: none"> <li>• HTML</li> <li>• PHP</li> <li>• XML</li> <li>• Apache Web Server</li> <li>• XAMPP</li> </ul>


### Τεχνολογίες και εργαλεία ανάπτυξης



#### HTML

- Είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες
- Τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων
- Τα στοιχεία της HTML αποτελούνται από ετικέτες π.χ. <html>
- Τα στοιχεία της HTML χρησιμοποιούνται για να κτίσουν όλους τους ιστότοπους
- Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στην σελίδα και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες

### Τεχνολογίες και εργαλεία ανάπτυξης



#### PHP

- Η PHP είναι μια γλώσσα προγραμματισμού και ιδιαίτερα δημοφιλής για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο.
- Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.

### Τεχνολογίες και εργαλεία ανάπτυξης

#### PHP

#### Πλεονεκτήματα χρήσης

- Είναι γρήγορη στην εκτέλεση.
- Έχει δυνατότητα επικοινωνίας με τις περισσότερες εμπορικές βάσεις δεδομένων.
- Διαθέτει πολλές βιβλιοθήκες.
- Διανέμεται και χρησιμοποιείται δωρεάν.
- Είναι εύκολη στην εκμάθησή της καθώς μοιάζει με τις C/C++, Java και perl.
- Υποστηρίζει τα τυπικά γνωρίσματα μιας αντικειμενοστραφούς γλώσσας (κλάσεις, κληρονομικότητα, πολυμορφισμός).
- Είναι μεταφράσιμη.
- Η υλοποίησή της μηχανής εκτέλεσής της είναι ανοικτού κώδικα

### Τεχνολογίες και εργαλεία ανάπτυξης


#### PHP

#### Στάδια επεξεργασίας μιας PHP σελίδας



The diagram illustrates the process: 1. A web browser requests a page (page.php) from a web server. 2. The web server sends the request to a PHP preprocessor. 3. The preprocessor checks for database connections and processes the PHP code. 4. The preprocessor sends the resulting HTML to the web server. 5. The web server then sends the HTML to the browser. A database is also shown connected to the preprocessor.

### Τεχνολογίες και εργαλεία ανάπτυξης



#### Apache Web Server

- Είναι εξυπηρετητής (Server)
- Ο Apache εγκαθίσταται σε έναν υπολογιστή ο οποίος μπορεί να χρησιμοποιεί διάφορα λειτουργικά συστήματα όπως Linux, Unix, Microsoft Windows, Solaris, Mac OS X
- Ο ρόλος του Apache είναι να αναμένει αιτήσεις από διάφορα προγράμματα – χρήστες και στη συνέχεια να εξυπηρετεί αυτές τις αιτήσεις “σερβιρόντας” τις σελίδες που ζητούν είτε απευθείας μέσω μιας ηλεκτρονικής διεύθυνσης (URL), είτε μέσω ενός συνδέσμου (link).
- Ο τρόπος με τον οποίο ο Apache εξυπηρετεί αυτές τις αιτήσεις, είναι σύμφωνα με τα πρότυπα που ορίζει το πρωτόκολλο HTTP

### Τεχνολογίες και εργαλεία ανάπτυξης



#### XAMPP

- Το XAMPP είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα και ανεξάρτητου πλατφόρμας.
- Περιέχει τα προγράμματα Apache, MySQL, PHP, phpMyAdmin, FileZilla FTP Server Tomcat
- Το XAMPP είναι ακρωνύμιο και αναφέρεται στα παρακάτω αρχικά:
  1. X (αναφέρεται στο "cross-platform" που σημαίνει λογισμικό ανεξάρτητο πλατφόρμας)
  2. Apache HTTP εξυπηρετητής
  3. MySQL
  4. PHP
  5. Perl

### Τεχνολογίες και εργαλεία ανάπτυξης



#### XML

- Η Επεκτάσιμη Γλώσσα Σήμανσης (Extensible Markup Language, XML) αποτελεί την ιδανική λύση στο πρόβλημα χειρισμού της ταχύτητα αυξανόμενης ποσότητας και πολυπλοκότητας των πληροφοριών που χρειάζεται να δημοσιεύθουν στον Ιστό.
- Ένα απλό XML αρχείο
 

```
<?xml version="1.0">
<messages>
  <message>Hello World! </message>
</messages>
```

### Τεχνολογίες και εργαλεία ανάπτυξης

#### XML

#### Η ανάγκη για την XML

- Έγγραφα που δεν αποτελούνται από τυπικά στοιχεία (επικεφαλίδες, ενότητες, λίστες, πίνακες, κτλ.).
- Βάσεις δεδομένων (Αν θέλουμε να ταξινομήσουμε, να επιλέξουμε, να εντοπίσουμε, και να δουλέψουμε με τις πληροφορίες μας με κάποιο διαφορετικό τρόπο, για να κάθε πληροφορία (πεδίο) θα πρέπει να οριστεί και από μια ετικέτα)
- Έγγραφα που θέλουμε να οργανώσουμε ιεραρχικά σε μορφή δένδρου.



## Τεχνολογίες και εργαλεία ανάπτυξης

### XML Οι επίσημοι στόχοι της XML

- Η XML θα είναι χρησιμοποιήσιμη στο Internet.
- Η XML θα υποστηρίζει μεγάλη ποικιλία εφαρμογών.
- Θα είναι εύκολη η συγγραφή προγραμμάτων που επεξεργάζονται έγγραφα XML.
- Τα έγγραφα της XML πρέπει να είναι αναγνώσιμα από τον άνθρωπο και σαφή.
- Ο σχεδιασμός της XML θα πρέπει να ολοκληρωθεί σύντομα. Τα έγγραφα XML θα δημιουργούνται εύκολα.
- Ο σχεδιασμός της XML θα πρέπει να είναι τυπικός και περιεκτικός.

## Τεχνολογίες και εργαλεία ανάπτυξης

### XML XML Parser

- Για να μπορέσουμε να προσπελάσουμε και να διαχειριστούμε κόμβους (nodes) ενός xml document θα πρέπει να χρησιμοποιήσουμε έναν parser.
- Υπάρχουν δύο γενικές κατηγορίες xml parser:
  - 1) **SAX** (Simple API for XML parser)
    - Σειριακή πρόσβαση στο αρχείο
    - Ιδανικό για λίστες, πίνακες στοιχείων και απλά μοντέλα δεδομένων.
  - 2) **DOM** parser
    - Χτίζει στην μνήμη την αντίστοιχη δενδρική δομή όλου του xml document τους κόμβους του οποίου μπορεί να προσπελάσει προς κάθε δυνατή κατεύθυνση.

## Τεχνολογίες και εργαλεία ανάπτυξης

### XML XML Parser Διαφορές

#### SAX

- Γρήγορος
- Λιγότερες απαιτήσεις σε μνήμη
- Περισσότερες κλήσεις μεθόδων για προσπέλαση των κόμβων

#### DOM

- Αργός
- Περισσότερες απαιτήσεις σε μνήμη
- Το DOM interface στο προγραμματισμό είναι δύσκολο.
- Λιγότερες κλήσεις μεθόδων για προσπέλαση των κόμβων

## Ανάπτυξη Εφαρμογής

### «Διαχείριση Πτυχιακών Εργασιών»

- Η συνολική εργασία είναι ένα αποτέλεσμα δημιουργημένο εξολοκλήρου από κώδικα
- Με την XML δημιουργήθηκε η βάση δεδομένων όπου αποθηκεύονται και επεξεργάζονται τα δεδομένα
- Με την PHP αναπτύχθηκαν όλες οι βασικές λειτουργίες της εφαρμογής και η διασύνδεση με την XML
- Με την HTML όλα τα στατικά περιεχόμενα των ιστοσελίδων

## Ανάπτυξη Εφαρμογής «Διαχείριση Πτυχιακών Εργασιών»

Το βασικό κομμάτι στην κατασκευή της εφαρμογής ήταν ο τρόπος διασύνδεσης της PHP με την XML και αυτό έγινε με την SimpleXML.

Η SimpleXML είναι γρήγορη και εύκολη στη χρήση

Με λίγες γραμμές κώδικα εκτελεί βασικές εργασίες, όπως:

- Ανάγνωση αρχείων XML
- Εξαγωγή δεδομένων από XML strings
- Επεξεργασία κειμένου κόμβου ή ιδιότητες

Πρέπει όμως να γνωρίζουμε την διάταξη του εγγράφου XML.

## Ανάπτυξη Εφαρμογής

### «Διαχείριση Πτυχιακών Εργασιών»

- Η βιβλιοθήκη SimpleXML βρίσκεται στον πυρήνα της έκδοσης PHP 5
- Γι αυτό και στον υπολογιστή στον οποίο θα τρέχει η εφαρμογή πρέπει να είναι εγκατεστημένη η έκδοση της PHP από 5 και πάνω

## Τέλος Παρουσίασης

Σας ευχαριστώ