

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ



Σπουδαστές

Παπαδομανωλάκης Σπυρίδων Α.Μ. 2896

Δαμιανάκης Μιχαήλ Α.Μ. 2135

Τσιμπουράκη Λαμπρινή Α.Μ. 3010

ΘΕΜΑ

«Τουριστικός Οδηγός του Μεσολογγίου σε xml»

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων καθηγητής – εισηγητής

κος Παπαδάκης Νικόλαος

Ηράκλειο

Μαίος, 2015

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ.....	4
ABSTRACT	7
ΣΥΝΟΨΗ	8
ΚΕΦΑΛΑΙΟ 1.....	9
ΕΙΣΑΓΩΓΗ.....	9
1.1 Περίληψη.....	9
1.2 Κίνητρο για τη διεξαγωγή της εργασίας.....	9
1.3 Σκοπός και στόχοι εργασίας.....	11
1.4 Δομή της εργασίας.....	11
ΚΕΦΑΛΑΙΟ 2.....	12
ΜΕΘΟΔΟΛΟΓΙΑ ΥΛΟΠΟΙΗΣΗΣ.....	12
2.1 Μέθοδος Ανάλυσης & Ανάπτυξης Πτυχιακής.....	12
Θεωρίες.....	12
2.2.1 Μοντέλο αρχιτεκτονικής λογισμικού πελάτη- διακομιστή (client-server).....	12
2.2.2 Internet (Διαδίκτυο).....	16
2.2.3 World Wide Web.....	19
2.2.4 Όνομα τομέα (Domain Name).....	20
2.2.5 Υπερκείμενο (Hypertext).....	22
2.2.6 Ιστοσελίδα (Web-page)	24
2.2.6.1 Στατική ιστοσελίδα.....	25
2.2.6.2 Δυναμική ιστοσελίδα.....	26
2.2.7 Γλώσσες σήμανσης.....	28
2.2.8 Standard Generalized Markup Language	28
2.2.9 HTML (HyperText Markup Language)	29
2.2.10 XHTML(eXtensible HyperText Markup Language).....	30
2.2.11 Web browser.....	34
2.2.12 CSS	34
2.2.13 PHP.....	36
2.2.14 Javascript	37
2.2.15 Ajax(Asynchronous JavaScript and XML)	39
2.2.16 Web Hosting – Φιλοξενία ιστοσελίδων.....	42
ΚΕΦΑΛΑΙΟ 3.....	43
ΣΧΕΔΙΟ ΔΡΑΣΗΣ ΓΙΑ ΤΗΝ ΕΚΠΟΝΗΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ.....	43
3.1 ΠρωτόκολλοXML (eXtensible Markup Language)	43
3.1.1 Γενική περιγραφή εγγράφου XML.....	45

3.1.2 Ορθότητα XML εγγράφου.....	50
3.1.2.1 DTD (Document Type Definition)	50
3.1.2.2 XSD (XML Schemas Definitions)	52
3.1.3 XML Parsers.....	55
3.1.4 Διαχείριση και μορφοποίηση κειμένων XML.....	56
3.1.4.1 XPath(XMLPathLanguage).....	57
3.1.4.2 XSL (XML Style Language)	59
3.1.4.3 XQUERY	64
3.1.5 Βάσεις Δεδομένων και XML.....	66
3.1.6 Πλεονεκτήματα και Μειονεκτήματα της XML.....	68
3.2 Ελεύθερο Λογισμικό	69
3.3 Συστήματα Διαχείρισης Περιεχομένου (Content Management System CMS)	70
3.4 CMS Ανοιχτού Κώδικα.....	72
ΚΕΦΑΛΑΙΟ 4.....	73
ΤΕΧΝΙΚΟ ΜΕΡΟΣ ΤΗΣ ΠΤΥΧΙΑΚΗΣ.....	73
4.1 Ανάλυση προβλήματος.....	73
4.2 Σχεδιασμός υλοποίησης.....	73
4.3 Υλοποίηση.....	75
4.3.1 XAMPP	75
4.3.2 GetSimple	79
ΚΕΦΑΛΑΙΟ 5.....	110
ΑΠΟΤΕΛΕΣΜΑΤΑ	110
5.1 Συμπεράσματα.....	110
5.2 Μελλοντική εργασία.....	111
ΒΙΒΛΙΟΓΡΑΦΙΑ	112

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1 : Σύστημα Client - Server.....	13
Εικόνα 2: Apache HTTP εξυπηρετητής.....	15
Εικόνα 3: Το διαδίκτυο	17
Εικόνα 4: Wi-Fi μέσω Access Points	19
Εικόνα 5 : DNS	21
Εικόνα 6: Ιεραρχική οργάνωση χώρου ονομάτων DNS.....	21
Εικόνα 7: Γραμμικό κείμενο (1) και υπερκείμενο (2).....	23
Εικόνα 8: Στατική Ιστοσελίδα.....	25
Εικόνα 9: Δυναμική ιστοσελίδα	27
Εικόνα 10: Δείγμα κώδικα HTML	30
Εικόνα 11: CSS	35
Εικόνα 12: Php	36
Εικόνα 13: Javascript	38
Εικόνα 14: Ajax.....	40
Εικόνα 15: Κλασική ιστοσελίδα σε σύγκρισή με ιστοσελίδα με Ajax	41
Εικόνα 16 : Διάγραμμα XML.....	47
Εικόνα 17: Διάγραμμα XML με αναφορές	48
Εικόνα 18: Παράδειγμα αιτήματος πελάτη σε εξυπηρετητή, που εμπλέκει τομετασχηματισμό κειμένου XML σε κείμενο HTML.	57
Εικόνα 19: Απεικόνιση μιας σχέσης (πίνακα) σε κείμενο XML: Η δομή του κειμένου XML (α), το αντίστοιχο διάγραμμα XML (β), και ένα παράδειγμα απεικόνισης (γ).	58
Εικόνα 20: Σύγκριση των XSLT και CSS.....	64
Εικόνα 21: XAMPP.....	73
Εικόνα 22: GetSimple CMS.....	74
Εικόνα 23: Εγκατάσταση Xampp Βήμα 1	76
Εικόνα 24: Εγκατάσταση Xampp Βήμα 2.....	76
Εικόνα 25: Εγκατάσταση Xampp Βήμα 3.....	77
Εικόνα 26: Εγκατάσταση Xampp Βήμα 4.....	77
Εικόνα 27: Εγκατάσταση Xampp Βήμα 5.....	78
Εικόνα 28: Control Panel του XAMPP	79
Εικόνα 29: Εγκατάσταση GetSimple	80
Εικόνα 30: Εγκατάσταση GetSimple	81
Εικόνα 31: Εγκατάσταση GetSimple	81
Εικόνα 32: Εγκατάσταση GetSimple	82
Εικόνα 33: Εγκατάσταση GetSimple	82

Εικόνα 34:Εγκατασταση GetSimple	83
Εικόνα 35: GetSimple Backend	83
Εικόνα 36:GetSimple Frontend	84
Εικόνα 37:C:\xampp\htdocs\mesologgi\data\users	84
Εικόνα 38:Administrator.xml	84
Εικόνα 39:GetSimple Backend Tabs.....	85
Εικόνα 40:GetSimple Backend Side Menu	86
Εικόνα 41:GetSimple Backend Pages Tab	86
Εικόνα 42:GetSimple Backend Files Tab	87
Εικόνα 43:GetSimple Backend Theme Tab	87
Εικόνα 44:GetSimple Backend Backups Tab.....	88
Εικόνα 45:GetSimple Backend Plugins Tab	88
Εικόνα 46:GetSimple Backend Logout & Settings	89
Εικόνα 47:GetSimple Backend Settings.....	89
Εικόνα 48:GetSimple Backend Logout.....	90
Εικόνα 49:GetSimple Theme Travels and Tours	90
Εικόνα 50:Το site όπως είναι στην τελική του μορφή	91
Εικόνα 51:CKeditor.....	91
Εικόνα 52:Βρισκουμε το gconfig.php	92
Εικόνα 53: Βρισκουμε την γραμμή που είναι ο WYSIWYG editor	92
Εικόνα 54:Και τον ενεργοποιούμε βγάζοντας τον από τα σχόλια	92
Εικόνα 55:WYSIWYG editor	93
Εικόνα 56:C:\xampp\htdocs\mesologgi\data\pages.....	93
Εικόνα 57:Backend Pages-Ιστορία.....	94
Εικόνα 58:History.xml	94
Εικόνα 59:Backend Pages-Ιστορία.....	95
Εικόνα 60:Frontend Pages-Ιστορία	95
Εικόνα 61:pages.xml	96
Εικόνα 62:Backend uploads	97
Εικόνα 63:Backend browse uploads.....	97
Εικόνα 64:Backend select browse uploads.....	97
Εικόνα 65:Backend Uploading images	98
Εικόνα 66:Backend Εισαγωγή εικόνας σε κείμενο	98
Εικόνα 67:Backend Εισαγωγή εικόνας σε κείμενο	99
Εικόνα 68:Backend Εισαγωγή εικόνας σε κείμενο	99
Εικόνα 69:Backend Εισαγωγή εικόνας σε κείμενο	100
Εικόνα 70:Backend Εισαγωγή εικόνας σε κείμενο	100

Εικόνα 71:Backend Εισαγωγή εικόνας σε κείμενο	101
Εικόνα 72:Τελική μορφή κειμένου στο frontend	101
Εικόνα 73:Edit Components στο tab Theme	102
Εικόνα 74:Συμπληρώνουμε τον κώδικα για το πλευρικό μενού	102
Εικόνα 75:Ενεργοποιούμε το component στον κώδικα της PHP	102
Εικόνα 76:Εμφανίζεται το μενού στην πλευρική στήλη	103
Εικόνα 77: Το πλευρικό μενού σε μορφή δέντρου μετα από εμπλουτισμό με CSS	103
Εικόνα 78:Ενεργοποίηση προσθέτων.....	104
Εικόνα 79:Χρήση προσθέτου Π18Ν στο pages	104
Εικόνα 80: Ορισμός google map στο backend.....	105
Εικόνα 81:Προσθηκη εντολής στη σελίδα για να εμφανιστεί ο χάρτης στο frontend	105
Εικόνα 82: Χάρτης στο frontend	105
Εικόνα 83:Plugins Activated	106
Εικόνα 84:Plugins.xml	106
Εικόνα 85:Plugin enabled:False	107
Εικόνα 86:Plugin Disabled.....	108
Εικόνα 87:Plugin enabled:True	108
Εικόνα 88:Plugin Enabled	109

ABSTRACT

Computer science in recent years developed a frantic pace in every part of it. Especially the Web has witnessed tremendous progress and now consists an integral part of our everyday life. Most people use the Internet and to update them through websites.

This thesis has a double objective. First goal was to create a complete and attractive website to inform the visitor of the beautiful region of Messologgi. Second was that the site is done by using XML technology.

The online travel guide with the name Visit Mesologgi is a dynamic website that provides information for Messologgi area offering all the necessary information about the history, attractions, activities, distances, useful telephone numbers, proposals for hotels and gallery of photographs in order to present the city's beauties.

The design of the site was made by GetSimple which is an open source web content management system based in XML markup language. This system is easy to use, flexible and offers many features with the appropriate customizing get a functional and good result.

The thesis consists of five chapters, which initially do a little analysis on the targets work and methodology followed for its preparation. Then we present in details the technologies commonly used for website development. After analyzing the XML technology which is the basic for this thesis and the content management systems. Next step is the analysis of the tools, implementation and operation of the application, listing snapshots and suitable examples. Finally, summarize the conclusions and recommendations made for some future additions and improvements.

ΣΥΝΟΨΗ

Η επιστήμη της πληροφορικής τα τελευταία χρόνια αναπτύσσεται με φρενήρεις ρυθμούς σε κάθε κομμάτι της. Ειδικά ο Παγκόσμιος Ιστός έχει γνωρίσει τρομερή εξέλιξη αφού πλέον αποτελεί αναπόσπαστο κομμάτι της καθημερινότητας μας. Οι περισσότεροι άνθρωποι χρησιμοποιούν το διαδίκτυο και για την ενημέρωση τους μέσα από ιστοσελίδες.

Η παρούσα πτυχιακή είχε διπλό στόχο. Πρώτος στόχος ήταν η δημιουργία ενός πλήρη και όμορφου ιστότοπου με σκοπό την ενημέρωση του επισκέπτη του για την όμορφη περιοχή του Μεσολογγίου. Δεύτερος στόχος ήταν ο ιστότοπος αυτός να γίνει με την χρήση της τεχνολογίας XML.

Ο διαδικτυακός τουριστικός οδηγός με το όνομα Visit Mesologgi είναι μια δυναμική ιστοσελίδα που παρέχει πληροφόρηση για την περιοχή του Μεσολογγίου προσφέροντας όλες τις απαραίτητες πληροφορίες για την ιστορία, τα αξιοθέατα, τις δραστηριότητες, τις χιλιομετρικές αποστάσεις, τα χρήσιμα τηλέφωνα, προτάσεις για ξενοδοχεία και φωτογραφικό υλικό από τις ομορφιές της πόλης.

Ο σχεδιασμός του ιστότοπου έγινε με ένα σύστημα διαχείρισης διαδικτυακού περιεχομένου ανοιχτού κώδικα που βασίζεται στην γλώσσα σήμανσης XML, το GetSimple. Το συγκεκριμένο σύστημα είναι εύκολο στη χρήση, ευέλικτο και προσφέρει πολλές λειτουργίες που με την κατάλληλη παραμετροποίηση παίρνουμε ένα λειτουργικό και καλό αποτέλεσμα.

Η πτυχιακή εργασία αποτελείται από πέντε κεφάλαια, στα οποία αρχικά κάνουμε μια μικρή ανάλυση για τους στόχους που έχει η εργασία και την μεθοδολογία που ακολουθήσαμε για την εκπόνηση της. Στην συνέχεια παρουσιάζουμε αναλυτικά τις τεχνολογίες που χρησιμοποιούνται ευρέως για την ανάπτυξη ιστοτόπων. Έπειτα αναλύουμε την βασική τεχνολογία XML που χρησιμοποιήθηκε για την εργασία όπως και τα συστήματα διαχείρισης περιεχομένου. Επόμενο βήμα είναι η ανάλυση των εργαλείων, της υλοποίησης και της λειτουργίας της εφαρμογής, παραθέτοντας στιγμιότυπα και κατάλληλα παραδείγματα. Τέλος, συνοψίζονται, τα συμπεράσματα και γίνονται προτάσεις για κάποιες μελλοντικές προσθήκες και βελτιώσεις.

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Περίληψη

Η παρούσα πτυχιακή εργασία αποτελεί έναν διαδικτυακό τόπο βασισμένο στην XML για την παροχή πληροφοριών και παράλληλα την καθοδήγηση των επισκεπτών για ψυχαγωγία και διακοπές στην περιοχή του Μεσολογγίου. Ο στόχος του ταξιδιωτικού οδηγού είναι η ενημέρωση και η εξυπηρέτηση του επισκέπτη της Ιεράς Πόλης του Μεσολογγίου κ γενικότερα την ευρύτερη περιοχή του δήμου Μεσολογγίου παρέχοντάς του με πολλούς και εύκολους τρόπους τις πληροφορίες που χρειάζεται για να κάνει τις διακοπές του όμορφες και να του χαρίσει μια ευχάριστη διαμονή. Μέσα από αυτό ο χρήστης έχει τη δυνατότητα να κάνει μια «ψηφιακή βόλτα» στη πόλη μαθαίνοντας για την ιστορία της, τα αξιοθέατά της, να αντλήσει πληροφορίες για το που να μείνει, τι μπορεί να κάνει για να περάσει τον χρόνο του ευχάριστα, τι να δοκιμάσει κ τέλος να βοηθηθεί από τον χάρτη που παρέχεται από την υπηρεσία Google maps.

Ο οδηγός μας έχει σκοπό να διευκολύνει τους χρήστες και τους υποψήφιους επισκέπτες ώστε να ενημερώνονται και να οργανώνουν εύκολα τις διακοπές και τις δραστηριότητες τους από το σπίτι τους.

Για να υλοποιηθεί το σύστημα έγινε χρήση του εργαλείου Xampp για την κατασκευή του localhost server, το GetSimple, ένα XML-Based CMS που χρησιμοποιήθηκε για τη δημιουργία του δικτυακού τόπου καθώς και οι τεχνολογίες XML,PHP, Javascript, HTML, CSS και Ajax.

1.2 Κίνητρο για τη διεξαγωγή της εργασίας

Το Μεσολόγγι, που αποκαλείται και η Ιερά πόλη, έχει ταυτιστεί περισσότερο από οποια άλλη πόλη με τον Αγώνα του 1821. Η πύλη της Εξόδου, ο κήπος των Ηρώων με τον τύμβο, ο χωματένιος φράχτης, ανδριάντες, αγαλμάτα και προτομές αγωνιστών και φιλελλήνων, όπως του Μάρκου Μπότσαρη, του λόρδου Βύρωνα, του Κυριάκου Μαυρομιχάλη, αποτελούν αποδείξεις του ρόλου που έπαιξε η πόλη στον αγώνα της ανεξαρτησίας. Η ευρύτερη περιοχή του Μεσολογγίου συμπεριλαμβάνει, μαζί με την Ιερά Πόλη του Μεσολογγίου, το Αιτωλικό, την αρχαία πόλη των

Οινιάδων, το Νεοχώρι και αποτελεί τουριστικό προορισμό πολλών Ελλήνων όσο και ξένων επισκεπτών.¹

Η ραγδαία αύξηση της ηλεκτρονικής ενημέρωσης είναι ένα χαρακτηριστικό της εποχής μας. Η ενημέρωση μέσω διαδικτύου είναι ένας άμεσος φθηνός και πολύ εύκολος τρόπος ενημέρωσης που είναι στη διάθεση μας συνέχεια. Αρα είναι λογικό να υπάρχει συνεχόμενα μεγαλύτερη αύξηση στη χρήση του διαδικτύου για ενημέρωση και υπάρχει μεγάλη ανάπτυξη αυτού του τομέα με αυξανόμενους δικτυακούς τόπους και εφαρμογές. Αρκετές εφημερίδες έχουν υιοθετήσει πέρα από την έντυπη ενημέρωση και την ηλεκτρονική μέσω διαδικτύου.

Σύμφωνα με τα ευρήματα της έρευνας που έγινε το 2012 το 65% των Ευρωπαίων χρησιμοποιούν το internet κατά μέσο όρο 14,8 ώρες την εβδομάδα. Πρώτος στόχος τους, είναι η ενημέρωση. Οι ιστοσελίδες ειδήσεων είναι πολύ δημοφιλείς, καθώς το 90% των χρηστών του διαδικτύου στην Ευρώπη καλύπτει τις ανάγκες της ενημέρωσης του μέσω internet.²

Ανάλογα συμπεράσματα προκύπτουν από έρευνα της VPRC που έγινε το 2013 για την εφημερίδα «Η Ελλάδα αύριο», σύμφωνα με την οποία το διαδίκτυο είναι το μέσο που χρησιμοποιούν περισσότερο οι Έλληνες για την ενημέρωσή τους. Συγκεκριμένα το 39% των Ελλήνων προτιμάει το ίντερνετ, αντι για το 32% που επιλέγει την τηλεόραση.³

Έχοντας υπόψη τα παραπάνω γεγονότα, είναι απαραίτητο να υπάρχει ένας διαδικτυακός τόπος που να λειτουργεί σαν τουριστικός οδηγός που σκοπό έχει την γενική πληροφόρηση ή και πιο συγκεκριμένη σχετικά με τα καταλύματα, τα αξιοθέατα, και τις δραστηριότητες που μπορεί να κάνει ένας επισκέπτης στο Μεσολόγγι. Ένας διαδικτυακός τόπος που όχι μόνο ενημερώνει για τις επιλογές που προσφέρονται αλλά μπορεί και να αποτελέσει διαφήμιση αφού προβάλλει και προωθεί τον τόπο σε μελλοντικούς επισκέπτες.

Όλο και περισσότεροι διαδικτυακοί τόποι υλοποιούνται τα τελευταία χρόνια, με βασικό σκοπό την προβολή κάποιου τουριστικού προορισμού. Για να καλυφθούν

¹<http://el.wikipedia.org/wiki/Μεσολόγγι>

²ΕΝΗΜΕΡΩΣΗ ΜΕΣΩ ΔΙΑΔΙΚΤΥΟΥ «ΨΗΦΙΖΟΥΝ» ΟΙ ΕΥΡΩΠΑΙΟΙ, 6 Ιουνίου 2012, <http://www.newsbomb.gr/media-agb/story/139241/enimerosi-meso-diadiktyou-psifizoun-oi-eyropaioi>

³ Ενημέρωση μέσω τηλεόρασης – ενημέρωση μέσω διαδικτύου σημειώσατε 1, 12 Ιανουαρίου 2012, <http://www.videotex.gr/%CE%B5%CE%BD%CE%B7%CE%BC%CE%AD%CF%81%CF%89%CF%83%CE%B7-%CE%BC%CE%AD%CF%83%CF%89-%CE%B4%CE%B9%CE%B1%CE%B4%CE%B9%CE%BA%CF%84%CF%8D%CE%BF%CF%85-%CE%B5%CE%BD%CE%B7%CE%BC%CE%AD%CF%81%CF%89%CF%81%CE%B7/>

οι παραπάνω ανάγκες, δημιουργήθηκε ο συγκεκριμένος τουριστικός οδηγός παρέχοντας τη δυνατότητα της ηλεκτρονικής ενημέρωσης προτείνοντας τα καλύτερα της περιοχής με φωτογραφικό υλικό για την πιο καλή πληροφόρηση του επισκέπτη.

1.3 Σκοπός και στόχοι εργασίας

Η παρούσα πτυχιακή εργασία αποσκοπεί στο να συνδυάσει την παροχή πληροφοριών και την καθοδήγηση των επισκεπτών για διακοπές στο Μεσολόγγι, με ένα ολοκληρωμένο δικτυακό τόπο χωρίς την βοήθεια κάποιας βάσης δεδομένων, αλλά εξ ολοκλήρου βασισμένο στην τεχνολογία XML.

Ο στόχος για της εργασίας αυτής ήταν να αποδείξουμε ότι με τις XML τεχνολογίες μπορούμε να πάρουμε ένα αποτέλεσμα εξίσου καλό με ένα δικτυακό τόπο που χρησιμοποιεί κάποια βάση δεδομένων.

1.4 Δομή της εργασίας

Παρακάτω θα γίνει περιγράψουμε αναλυτικά όλα τα βήματα που ακολουθήθηκαν για να υλοποιηθεί η παρούσα πτυχιακή εργασία.

- **Κεφάλαιο 2:** Παρουσιάζεται αναλυτικά η μεθοδολογία υλοποίησης, δηλαδή οι θεωρίες των τεχνολογιών για τον σχεδιασμό και την υλοποίηση του παρόντος συστήματος.

- **Κεφάλαιο 3:** Αναλύονται οι βασικές τεχνολογίες που καταλήξαμε να χρησιμοποιήσουμε σαν βάση για να εκπονηθεί η συγκεκριμένη πτυχιακή δηλαδή την γλώσσα XML και τα Συστήματα Διαχείρισης Περιεχομένου.

- **Κεφάλαιο 4:** Αναλύουμε το πρόβλημα και παρουσιάζονται τα εργαλεία που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας. Επιπλέον δείχνουμε και πως υλοποιήθηκε η εργασία όπως και κάποιες από τις βασικές λειτουργίες της.

- **Κεφάλαιο 5:** Τέλος βγάζουμε τα συμπεράσματα και προτείνουμε κάποια μελλοντικά σχέδια στα οποία μπορεί να χρησιμοποιηθεί σαν βάση η εργασία μας.

ΚΕΦΑΛΑΙΟ 2

ΜΕΘΟΔΟΛΟΓΙΑ ΥΛΟΠΟΙΗΣΗΣ

2.1 Μέθοδος Ανάλυσης & Ανάπτυξης Πτυχιακής

Για την πραγματοποίηση της παρούσας πτυχιακής χρειάστηκε να γίνουν μια σειρά από ενέργειες, όπως

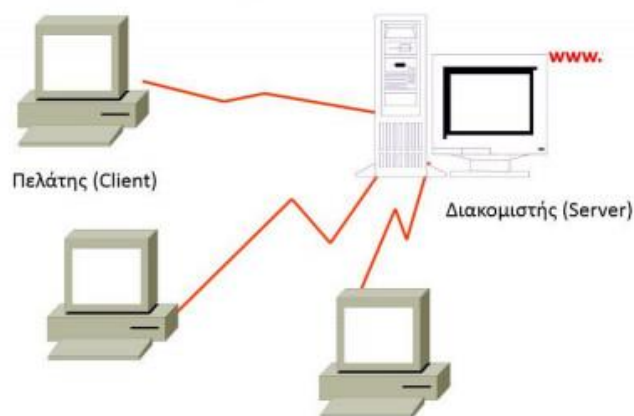
1. Να βρούμε πληροφορίες σχετικά με τις τεχνολογίες που θα χρησιμοποιήσουμε
2. Να στηθεί ο localhost με τη βοήθεια του εργαλείου xampp,
3. Να βρεθεί το κατάλληλο CMS,
4. Να βρούμε, να προσαρμόσουμε και να επεξεργαστούμε το theme,
5. Να προσθέσουμε κάποια στοιχεία προκειμένου να γίνει πιο όμορφος και φιλόξενος ο δικτυακός τόπος μας,
6. Να βρούμε πληροφορίες για την ιστορία, τα αξιοθέατα, τις δραστηριότητες που μπορεί να κάνει κάποιος στο Μεσολόγγι και τα τοπικά προϊόντα.
7. Να βρούμε και να επεξεργαστούμε φωτογραφίες του τόπου
8. Να βρεθούν τα κατάλληλα plugins για να μπορέσουμε να προσθέσουμε παραπάνω στοιχεία και να κάνουμε το site πιο εύχρηστο.

Θεωρίες

2.2.1 Μοντέλο αρχιτεκτονικής λογισμικού πελάτη- διακομιστή (client-server) Εξυπηρετητής ή διακομιστής (server) θεωρείται το υλικό ή και το λογισμικό που αναλαμβάνει την παροχή διάφορων υπηρεσιών, «εξυπηρετώντας» αιτήσεις άλλων υλικών ή/και λογισμικών, επονομάζομενους ως πελάτες (clients) που μπορούν να τρέχουν στον ίδιο υπολογιστή ή σε δίκτυο. Όταν ένας υπολογιστής τρέχει τέτοια προγράμματα συνεχόμενα, 24 ώρες την ημέρα, δηλαδή να είναι αυτή η κύρια λειτουργία του, τότε μπορούμε να αναφερθούμε σε όλον τον υπολογιστή ως server. Ομοίως, ως πελάτης μπορεί να θεωρηθεί είτε κάποιο λογισμικό που επικοινωνεί και υποβάλει αιτήματα στον εξυπηρετητή και οι 2 υπολογιστές είναι συνδεδεμένοι σε ένα δίκτυο.

Το μοντέλο αρχιτεκτονικής λογισμικού πελάτη-διακομιστή (client-server) αποτελεί μία συνήθη μέθοδο ανάπτυξης λογισμικού στην οποία ο πελάτης ζητά κάτι

π.χ. έναν πόρο ή τα αποτελέσματα κάποιου υπολογισμού κ.ο.κ. και ο διακομιστής το επιστρέφει. Κάθε διακομιστής έχει την δυνατότητα να εξυπηρετεί ταυτόχρονα παραπάνω από ένα πελάτες. Το μοντέλο πελάτη-διακομιστή συντελεί μία από τις μεθόδους ανάπτυξης και λειτουργίας κατανεμημένων συστημάτων, όπου τόσο ο πελάτης όσο και ο διακομιστής θεωρούνται διαφορετικά τμήματα της ίδιας κατανεμημένης εφαρμογής.



Εικόνα 1 : Σύστημα Client - Server

Ο διακομιστής και ο πελάτης εκτελούνται σε διαφορετικές διεργασίες, οι οποίες μπορούν να εκτελούνται σε διαφορετικούς υπολογιστές, οπότε απαιτείται ένα δίκτυο για να επικοινωνήσουν οι διεργασίες μεταξύ τους. Αυτή η επικοινωνία μεταξύ πελάτη και εξυπηρετητή γίνεται μέσω ενός τοπικού, ή μητροπολιτικού, δικτύου ή του Διαδικτύου. Στα μεγάλα δίκτυα που ο διακομιστής έχει να εξυπηρετήσει πολλές διεργασίες ή υπολογιστές είναι συνήθως υπολογιστής που είναι διαφορετικός από άλλους κοινούς υπολογιστές ως προς τη σύνθεσή του, μιας και οι υποχρεώσεις του είναι σαφώς αναβαθμισμένες. Τα κύρια χαρακτηριστικά ενός διακομιστή είναι οι επεξεργαστές που υποστηρίζει και χρησιμοποιεί για την επεξεργασία των δεδομένων, οι σκληροί δίσκοι μεγάλης ταχύτητας και χωρητικότητας ,αλλά και οι ταχύτερες μνήμες που υποστηρίζει. Το συνήθες είναι να συνεργάζεται με σύστημα διπλής τροφοδοσίας και από συσκευή αδιάλειπτης παροχής ενέργειας (UPS), για να είναι πιο αξιόπιστος και να έχει σιγουριά στις παρεχόμενες υπηρεσίες.

Οι υπολογιστές που χρησιμοποιούνται ως εξυπηρετητές δικτύου είναι υποχρεωμένοι να μπορούν να δουλεύουν όλη την ημέρα και συνεχώς χωρίς διακοπές. Γι αυτό οι εταιρείες που κατασκευάζουν υπολογιστές και τμημάτων υπολογιστών προσφέρουν και διαθέτουν στην ευρεία αγορά ξεχωριστές κατασκευές και εξαρτήματα για εξυπηρετητές που μπορούν να χρησιμοποιηθούν και για προσωπικό

υπολογιστή άλλα ανεβάζουν κατά πολύ το κόστος από ένα κοινό προσωπικό υπολογιστή.

Συνηθισμένα χαρακτηριστικά για το υλικό ενός διακομιστή είναι :

- Motherboards με μεγάλη αντοχή στις θερμοκρασίες, ενσωματωμένα συστήματα RAID, ενσωματωμένη κάρτα οθόνης μικρής μνήμης και ταχύτητας, πιο άνετα κατανεμημένα τα στοιχεία της μνήμης και του επεξεργαστή ώστε να είναι δυνατή η ψύξη και των δύο από ξεχωριστά δυνατά ανεμιστηράκια, λιγότερες θύρες επέκτασης, περισσότερα ενδεικτικά στοιχεία λειτουργίας και δυνατότητα χρήσης μνήμης ECC.

- Ειδικά κουτιά που έχουν την δυνατότητα ενσωμάτωσης σε rack, βαριά με ειδικό κλείδωμα, θέσεις για περισσότερα ανεμιστηράκια και ειδικά διαμορφωμένους εσωτερικούς χώρους ώστε να γίνεται πιο καλή κυκλοφορία του αέρα.

- Σκληροί δίσκοι που έχουν κατασκευαστεί για μεγαλύτερο συνεχόμενο χρόνο χρήσης που μπορούν να αποσυνδύονται την ώρα που είναι σε λειτουργία

Σχετικά με το λογισμικό και πιο ειδικά με τα λειτουργικά συστήματα, τα περισσότερα από αυτά έχουν ειδικές εκδόσεις για χρήση ως διακομιστή ή μπορούν να διαμορφωθούν έτσι εκ των υστέρων εκτελώντας λογισμικό που κάνει την εξυπηρέτηση.

Οι διακομιστές που χρησιμοποιούνται για το διαδίκτυο αναλαμβάνουν πολύ δουλειά και είναι κατάλληλα οργανωμένοι με χαρακτηριστικά που τους διαφοροποιούν από απλούς εξυπηρετητές από τα οποία είναι:

- Λειτουργούν χωρίς την ύπαρξη πληκτρολογίου, οθόνης, γραφικού περιβάλλοντος και κάρτας ήχου.

- Μπορείς να επιλέξεις τον χρόνο επεξεργασίας των διάφορων προγραμμάτων.

- Το πρόγραμμα επικοινωνίας με το UPS.

- Χρησιμοποιούν παραπάνω από ένα επεξεργαστές.

- Δύναται συνεπεξεργασία με άλλους υπολογιστές- εξυπηρετητές.

- Πρόσθετες δυνατότητες για αλλαγή του υλικού και αναβάθμισης του λειτουργικού χωρίς την ανάγκη επανεκκίνησης του συστήματος.

- Περαιτέρω συστήματα ασφαλείας.

Πολλά σύγχρονα προγράμματα δουλεύουν με τη λογική πελάτη – εξυπηρετητή ακόμα και το ίδιο το λειτουργικό σύστημα. Τα προγράμματα ζητάνε να γίνει κάποια ενέργεια και το λειτουργικό σύστημα αναλαμβάνει να τα εξυπηρετήσει εκτελώντας τις καταλλήλες λειτουργίες που του ζητήθηκαν. Τα πιο πολλά προγράμματα εξυπηρετητών απαιτούν και ξεχωριστό υπολογιστή-εξυπηρετητή χωρίς αυτό να είναι πάντα απαραίτητο.

Οι πιο συνηθισμένες μορφές εξυπηρετητών-προγράμματος στο Ίντερνετ είναι, παγκόσμιου ιστού με το πρωτόκολλο http (http server), Domain Name System (DNS server), ηλεκτρονικού ταχυδρομείου (mail server), μεταφοράς αρχείων με το πρωτόκολλο FTP (ftp server), Irc και instant messaging (irc server), επικοινωνίες φωνής, Streaming audio και video (streaming server), Online παιχνίδια.

Ο Apache HTTP γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού. Όταν κάποιος χρήστης επισκέπτεται ένα ιστότοπο ο browser επικοινωνεί με έναν server μέσω του πρωτοκόλλου HTTP, που παράγει τις ιστοσελίδες και μετά, αποστέλονται στον browser. Ο Apache είναι ένας από τους πιο δημοφιλείς εξυπηρετητές ιστού επειδή λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το Unix και το Mac OS X. Κυκλοφορεί με την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Η συντήρηση του γίνεται από μια κοινότητα ανοικτού κώδικα και επιτηρείται από το Software Foundation.



Εικόνα 2: Apache HTTP εξυπηρετητής

Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. MySQL.

Η πρώτη του έκδοση, γνωστή ως NCSA HTTPd, δημιουργήθηκε από τον Robert McCool και η κυκλοφορία του έγινε το 1993 και έπαιξε σημαντικό ρόλο στην επέκταση του παγκόσμιου ιστού. Ήταν η πρώτη εναλλακτική επιλογή που αξιοποιήθηκε και παρουσιάστηκε απέναντι στον εξυπηρετητή http της εταιρείας

Netscape και από τότε εξελίχθηκε, σε λειτουργικότητα και απόδοση, τόσο ώστε να ανταγωνίζεται άλλους εξυπηρετητές βασισμένους στο Unix⁴.

2.2.2 Internet (Διαδίκτυο)

Η λέξη Διαδίκτυο έχει την καταγωγή της από τις λέξεις Διασύνδεση Δικτύων και αναφέρεται σε ένα σύνολο υπολογιστών και δικτύων που συνδέονται μεταξύ τους σε ένα παγκόσμιο δίκτυο έτσι ώστε να μπορούν να επικοινωνούν και να μοιράζονται πληροφορίες. Στα Αγγλικά η λέξη Internet παράγεται από τις λέξεις International Network που σημαίνει Διεθνές Δίκτυο Υπολογιστών.

Με την εμφάνιση του, ο τομέας της επικοινωνίας αναμφισβήτητα επηρεάστηκε. Σχετικά με το επίπεδο που μπορεί η συγκεκριμένη τεχνολογία του διαδικτύου να αλλάξει τον τρόπο με τον οποίο επικοινωνούν και πληροφορούνται μαζικά οι άνθρωποι, ακόμα κ σήμερα αρκετά χρόνια μετά την δημιουργία του, υπάρχουν αρκετές διαφορετικές και αντικρουόμενες προσεγγίσεις και απόψεις πάνω στο ζήτημα αυτό. Αυτό οφείλεται κυρίως από την τεχνολογία του νέου μέσου.

Η τεχνολογία του είναι κυρίως βασισμένη στην διασύνδεση επιμέρους δικτύων ανά τον κόσμο και πολυάριθμα πρωτόκολλα επικοινωνίας. Στην πιο εξειδικευμένη και χρησιμοποιούμενη μορφή του, το Διαδίκτυο περιγράφεται σαν το παγκόσμιο πλέγμα διασυνδεδεμένων υπολογιστών και των υπηρεσιών και πληροφοριών που παρέχει στους χρήστες του, έτσι σήμερα, ο όρος διαδίκτυο αναφέρεται στο παγκόσμιο αυτό δίκτυο. Το Διαδίκτυο χρησιμοποιεί μεταφορά πακέτων και στοίβα πρωτοκόλλων και τεχνική της διασύνδεσης δικτύων μέσω αυτών ονομάζεται Διαδικτύωση.

⁴Apache HTTP Server, 2015, <http://httpd.apache.org/> & Benatallah, B.; Casati, F.; Toumani, F. (2004). "Web service conversation modeling: A cornerstone for e-business automation". *IEEE Internet Computing*



Εικόνα 3: Το διαδίκτυο

Οι πρώτες προσπάθειες για την δημιουργία ενός τέτοιου δικτύου ξεκίνησαν στις ΗΠΑ κατά την διάρκεια του ψυχρού πολέμου. Ο δορυφόρος Σπούτνικ 1 είχε σταλθεί από τη Ρωσία στο διάστημα και αυτό έκανε τους Αμερικανούς να φοβούνται για την ασφάλεια της χώρας τους. Για την προστασία από μια πιθανή επίθεση των Ρώσων δημιούργησαν την υπηρεσία προηγμένων αμυντικών ερευνών ARPA (Advanced Research Project Agency), σήμερα γνωστή ως DARPA (Defense Advanced Research Projects Agency) που σαν κύρια αποστολή της ήταν η τεχνολογική ανάπτυξη των στρατιωτικών δυνάμεων των ΗΠΑ και η δημιουργία ενός δικτύου επικοινωνίας το οποίο θα μπορούσε να επιβιώσει σε μια ενδεχόμενη επίθεση.

Το αρχικό θεωρητικό υπόβαθρο δόθηκε από τον J.C.R. Licklider που έγραψε για το "γαλαξιακό δίκτυο". Το σενάριο αυτό υποστήριζε την ύπαρξη ενός δικτύου υπολογιστών που θα ήταν συνδεδεμένοι μεταξύ τους και θα μπορούσαν να ανταλλάσσουν γρήγορα πληροφορίες και προγράμματα. Το επόμενο ζήτημα που προέκυψε ήταν ότι το δίκτυο αυτό θα έπρεπε να ήταν αποκεντρωμένο έτσι ώστε ακόμα κι αν κάποιος κόμβος του δεχόταν επίθεση να υπήρχε επικοινωνία για τους υπόλοιπους υπολογιστές. Η λύση δόθηκε από τον Paul Baran με τον σχεδιασμό ενός κατακεντρωμένου δικτύου επικοινωνίας που χρησιμοποιούσε ψηφιακή τεχνολογία. Σημαντική συμβολή είχε και η θεωρία ανταλλαγής πακέτων του Leonard Kleinrock,

υποστηρίζοντας ότι πακέτα πληροφοριών που θα περιείχαν την προέλευση και τον προορισμό τους μπορούσαν να σταλούν από έναν υπολογιστή σε έναν άλλο.

Έχοντας στηριχθεί σε αυτές τις τρεις θεωρίες δημιουργήθηκε το ARPANET που είναι ουσιαστικά το πρώτο είδος διαδικτύου. Η εγκατάσταση και η λειτουργία του έγιναν για πρώτη φορά το 1969 με 4 κόμβους μέσω των οποίων συνδέονται 4 mini computers 12k του πανεπιστημίου της Καλιφόρνια στην Σάντα Μπάρμπαρα, το τμήμα του στο Λος Άντζελες, το SRI που έδρευε στο Στάνφορντ και το πανεπιστήμιο της Γιούτα. Το δίκτυο είχε ταχύτητα 50 kbps και έτσι επιτεύχθηκε η πρώτη dial up σύνδεση μέσω τηλεφωνικών γραμμών. Μέχρι το 1972 οι συνδεδεμένοι στο ARPANET υπολογιστές είχαν ανέβει στους 23, και εφάρμοσαν πρώτη φορά το σύστημα διαχείρισης ηλεκτρονικού ταχυδρομείου (e-mail).

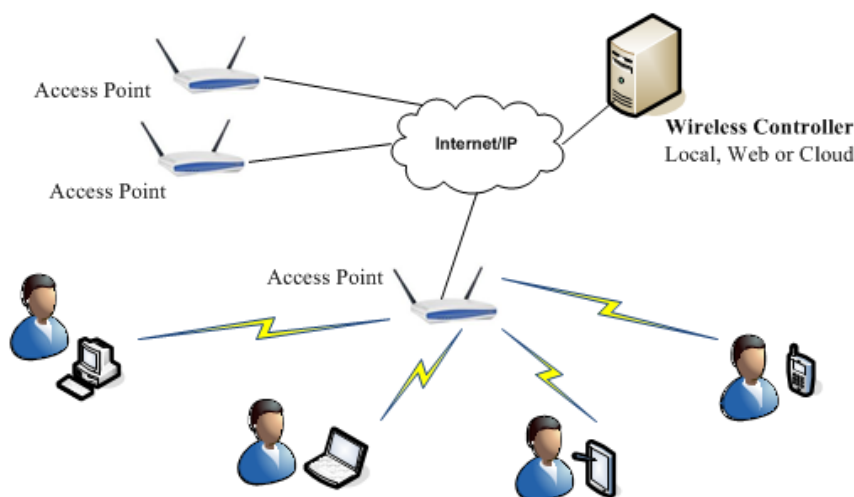
Παράλληλα δημιουργήθηκαν και άλλα δίκτυα, που χρησιμοποιούσαν άλλα πρωτόκολλα, όπως το x.25 και το UUCP, που γινόταν σύνδεση με το ARPANET. Το ARPANET χρησιμοποιούσε το πρωτόκολλο NCP (Network Control Protocol), του οποίου, όμως, το μειονέκτημα ήταν ότι λειτουργούσε μόνο με συγκεκριμένους τύπους υπολογιστών. Χρειάζονταν ένα πρωτόκολλο που θα ένωνε όλα τα δίκτυα που είχαν δημιουργηθεί μέχρι τότε. Η λύση δόθηκε με την μελέτη των Vint Cerf και Bob Kahn από την οποία προέκυψε το πρωτόκολλο TCP (Transmission Control Protocol) που αργότερα το 1978 έγινε TCP/IP, προστέθηκε δηλαδή το Internet Protocol (IP), ώσπου το 1983 έγινε το μοναδικό πρωτόκολλο που ακολουθούσε το ARPANET.

Το 1984 υλοποιείται το πρώτο DNS (Domain Name System) σύστημα στο οποίο καταγράφονται 1000 κεντρικοί κόμβοι και η αναγνώριση των υπολογιστών του διαδικτύου γίνεται πλέον από διευθύνσεις κωδικοποιημένων αριθμών. Ένα βήμα στην ανάπτυξη του Διαδικτύου έκανε το Εθνικό Ίδρυμα Επιστημών (National Science Foundation, NSF) των ΗΠΑ, το οποίο δημιούργησε το NSFNet, το 1986. Ακολούθησε η ενσωμάτωση και άλλων σημαντικών δικτύων, όπως το Usenet, το Fidonet και το Bitnet.

Ο όρος Διαδίκτυο/Ίντερνετ ξεκίνησε να χρησιμοποιείται την εποχή που συνδέθηκε το ARPANET με το NSFNet σήμαινε οποιοδήποτε δίκτυο χρησιμοποιούσε TCP/IP. Η μεγάλη άνθηση του Διαδικτύου όμως, έγινε όταν εφαρμόστηκε η υπηρεσία του Παγκόσμιου Ιστού από τον Τιμ Μπέρνερς-Λι στο ερευνητικό ίδρυμα CERN το 1989, ο οποίος ουσιαστικά είναι, η πλατφόρμα πάνω στην οποία κάνει εύκολη την πρόσβαση στο Ίντερνετ, ακόμα και στη σημερινή του μορφή.

Το διαδίκτυο παρομοιάζεται με «υπερλεωφόρο πληροφοριών». Κάθε μέρα διακινείται πλήθος δεδομένων με οποιαδήποτε μορφή όπως κείμενα , εικόνες , ήχοι, μουσική, βίντεο, φέρνοντας στην οθόνη του υπολογιστή μας ένα τεράστιο αριθμό ψηφιακών πηγών πληροφόρησης.

Κοινές μέθοδοι πρόσβασης στο Διαδίκτυο είναι η επιλογική και η ευρυζωνική.



Εικόνα 4: Wi-Fi μέσω Access Points

Το δικαίωμα των Ευρωπαίων πολιτών για ελεύθερη πρόσβαση στο Διαδίκτυο κατοχυρώνεται στο άρθρο 11 του Χάρτη των Θεμελιωδών Δικαιωμάτων της Ευρωπαϊκής Ένωσης περί ελευθερίας της έκφρασης και της ενημέρωσης⁵.

2.2.3 World Wide Web

Παγκόσμιος ιστός και Internet λανθασμένα θεωρούνται το ίδιο πράγμα, καθώς ο ιστός αποτελεί την δημοφιλέστερη εφαρμογή του Internet. Το Internet έχει και υλική υπόσταση, ενώ ο ιστός δεν έχει, αφού αποτελείται από πακέτα πληροφορίας.

Παγκόσμιος ιστός είναι το δίκτυο των συνδεδεμένων υπολογιστών και δικτύων παγκοσμίως, το οποίο χρησιμοποιεί το πρωτόκολλο επικοινωνίας HTTP(Hypertext Transfer Protocol). Κάθε μονάδα του διαδικτύου αποτελείται από

⁵Robins, Kevin & Frank Webster. (1999). Times of the Technoculture: From the Information Society to the Virtual Life, London: Routledge. & Hauben, M. (1994) 'Netizens an Anthology' & Εισαγωγή στην προσβασιμότητα στο διαδίκτυο (WebAIM), 2015, <http://microo.net/article.php?intro-to-web-accessibility> & A Privacy Manifesto for the Web 2.0 Era, 2015, <https://gigaom.com/2008/01/08/a-privacy-manifesto-for-the-web-20-era/>

συνδεδεμένους υπολογιστές σε τοπικό επίπεδο και αυτά με τη σειρά τους συνδέονται σε μεγαλύτερα δίκτυα. Παγκόσμιος ιστός είναι, το ευρύτερο δίκτυο το οποίο είναι μοναδικό και συμπεριλαμβάνονται τα δίκτυα που βρίσκονται στον πλανήτη γη και αυτά των δορυφόρων της και άλλων διαστημικών συσκευών.

Η τεχνολογία του ιστού δημιουργήθηκε το 1989 από τον Βρετανό Τιμ Μπέρνερς Λη και το όνομα που έδωσε στην εφεύρεσή του είναι World Wide Web, όρος γνωστός από το "www". Το σημείο στο οποίο έδωσε ιδιαίτερο βάρος ήταν η μη ιεράρχηση των διασυνδεδεμένων στοιχείων, δηλαδή κάθε κόμβος ή στοιχείο του ιστού, να είναι ίσο ως προς την προσβασιμότητα με τα υπόλοιπα.

Πλέον η τεχνολογία του ιστού μπορεί να δημιουργήσει "υπερκειμένα", δηλαδή την διασύνδεση πάρα πολλών μη ιεραρχημένων στοιχείων που παλαιότερα ήταν απομονωμένα τα οποία μπορούν να πάρουν και άλλες μορφές πέραν της μορφής του γραπτού κειμένου, όπως εικόνας και ήχου.

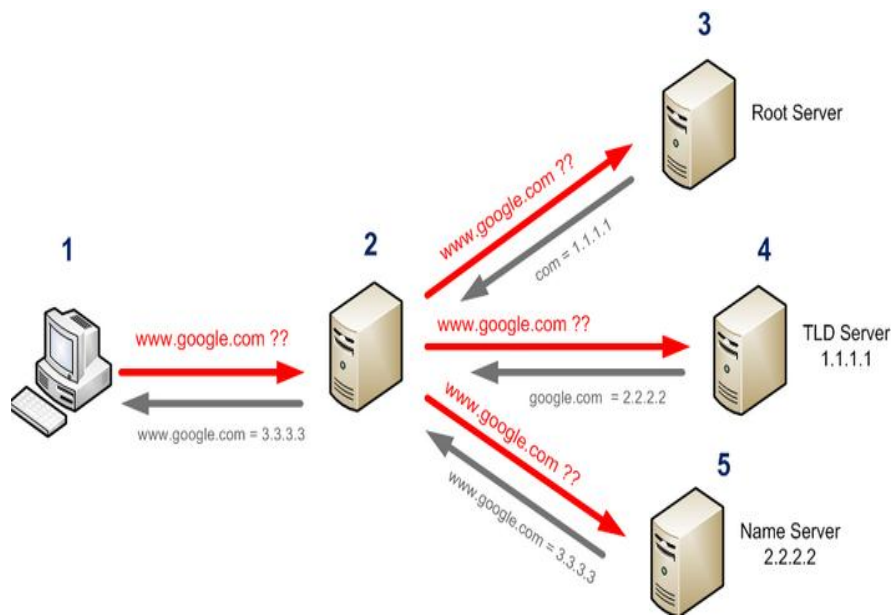
Το μοντέλο δεδομένων του παγκόσμιου ιστού βασίζεται σε κόμβους και υπερσυνδέσμους έτσι παρέχει κόμβους (nodes), άγκυρες (anchors) και συνδέσμους ή υπερσυνδέσμους (links ή hyperlinks). Οι κόμβοι είναι φέρουν περιεχόμενο ενώ οι άγκυρες είναι περιοχές του περιεχομένου των κόμβων από όπου ξεκινούν ή καταλήγουν σύνδεσμοι. Οι σύνδεσμοι συνδέουν δύο κόμβους⁶.

2.2.4 Όνομα τομέα (Domain Name)

Όνομα χώρου ή τομέα ή περιοχής (domain name) στο Διαδίκτυο είναι ένας περιορισμένος τομέας των διεθνών πόρων του Συστήματος Ονομάτων Χώρου (DNS) ο οποίος παρέχεται, για χρήση σε ένα πρόσωπο που έχει μόνο την δυνατότητα χρήσης του για κάποιο χρονικό διάστημα. Το όνομα χώρου συνηθίζεται να έχει ποικίλες καταλήξεις όπως .com, .gr, .net κ.α., ανάλογως πως χρησιμοποιείται και τη χώρα που προέρχεται.

Στα ονόματα χώρου επιτρέπεται μόνο η χρήση γραμμάτων, αριθμών και παυλών. Άλλος περιορισμός είναι ότι αν ένα όνομα χώρου θεωρείται κοινόχρηστο ή γεωγραφικός όρος μπορεί να εκχωρηθεί μόνο στους αντίστοιχους οργανισμούς τοπικής αυτοδιοίκησης.

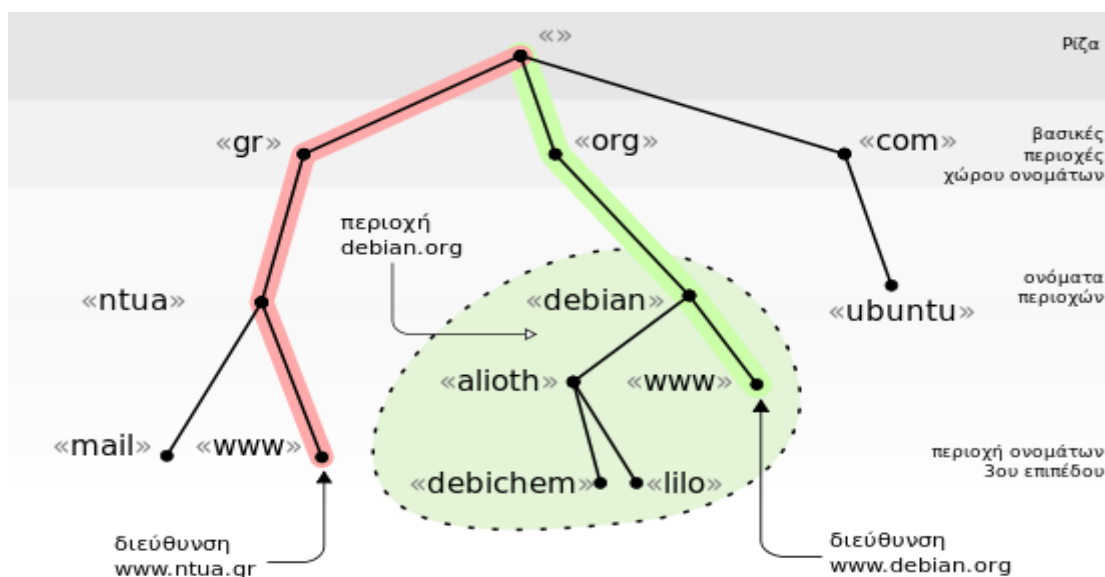
⁶Παγκόσμιος Ιστός, 2014, http://el.wikipedia.org/wiki/%CE%A0%CE%B1%CE%B3%CE%BA%CF%8C%CF%83%CE%BC%CE%B9%CE%BF%CF%82_%CE%99%CF%83%CF%84%CF%8C%CF%82



Εικόνα 5 : DNS

Το Domain Name System ή DNS (Σύστημα Ονομάτων Τομέων ή Χώρων ή Περιοχών) είναι ένα σύστημα ονοματοδοσίας για δίκτυα υπολογιστών, που χρησιμοποιούν το πρωτόκολλο IP τα οποία μπορεί και αντιστοιχίζει το σύστημα DNS.

Ο χώρος ονομάτων τομέων (Domain Name Space) του DNS είναι δομημένος ιεραρχικά σε δομή δέντρου. Η αντιστοίχιση λέξεων με αριθμητικές διευθύνσεις έγινε επειδή οι άνθρωποι θυμούνται ευκολότερα ονόματα και οι αριθμητικές διευθύνσεις IP δεν είναι εύχρηστες από αυτούς. Για ένα χρήστη είναι ευκολότερο να θυμάται το όνομα της `www.google.gr` παρά το `x.y.z.v:80` δηλαδή τον συνδυασμό διεύθυνσης IP και θύρας στην οποία βρίσκεται ο εξυπηρετητής του `www.google.gr`.



Εικόνα 6:Ιεραρχική οργάνωση χώρου ονομάτων DNS

Στην εικόνα 6 περιγράφεται η Ιεραρχική οργάνωση χώρου ονομάτων DNS. Η περιοχή ονομάτων της Ελλάδας είναι το gr. Για το Εθνικό Μετσόβιο Πολυτεχνείο έχει κατοχυρωθεί το όνομα ntua και ο διακομιστής σελίδων Παγκόσμιου Ιστού έχει το τρίτο επίπεδο www δημιουργώντας την διεύθυνση www.ntua.gr. Η σελίδα του email στο ΕΜΠ είναι η mail.ntua.gr. Με τον ίδιο τρόπο μπορούμε να δούμε στο σχεδιάγραμμα πως δημιουργούνται οι διευθύνσεις των ιστοσελίδων για τα www.debian.org και ubuntu.com.

Το Σύστημα DNS έχει την βάση του σε μια διανεμημένη βάση δεδομένων η οποία τρέχει στους servers του συστήματος και αποτελείται από ζώνες DNS οργανωμένες σε μια δομή δέντρου.

Οι χώροι ή περιοχές ή τομείς χωρίζονται σε επίπεδα, και κάθε επίπεδο συχνά περιέχει παρακάτω επίπεδα. Η αλλαγή επιπέδου των ονομάτων χώρου ισούται με αλλαγή ζώνης DNS (DNS zone).

Οι ζώνες DNS συνήθως σημαίνουν ότι αλλάζει η διαχείριση μιας περιοχής. Όταν κατοχυρώνεται ένα όνομα χώρου ουσιαστικά ελέγχεται η ζώνη DNS του ονόματος χώρου.

Η σχέση που συνδέει τα ονόματα και τις διευθύνσεις IP δεν είναι 1 προς 1 δηλαδή σε ένα όνομα μπορεί να αντιστοιχούν πολλές διευθύνσεις IP και σε μια διεύθυνση πολλά ονόματα. Αυτό κάνει ευκολότερη την διαχείριση των διαδικτυακών εξυπηρετητών και βοηθά στην οικονομία διευθύνσεων IP.⁷

2.2.5 Υπερκείμενο (Hypertext)

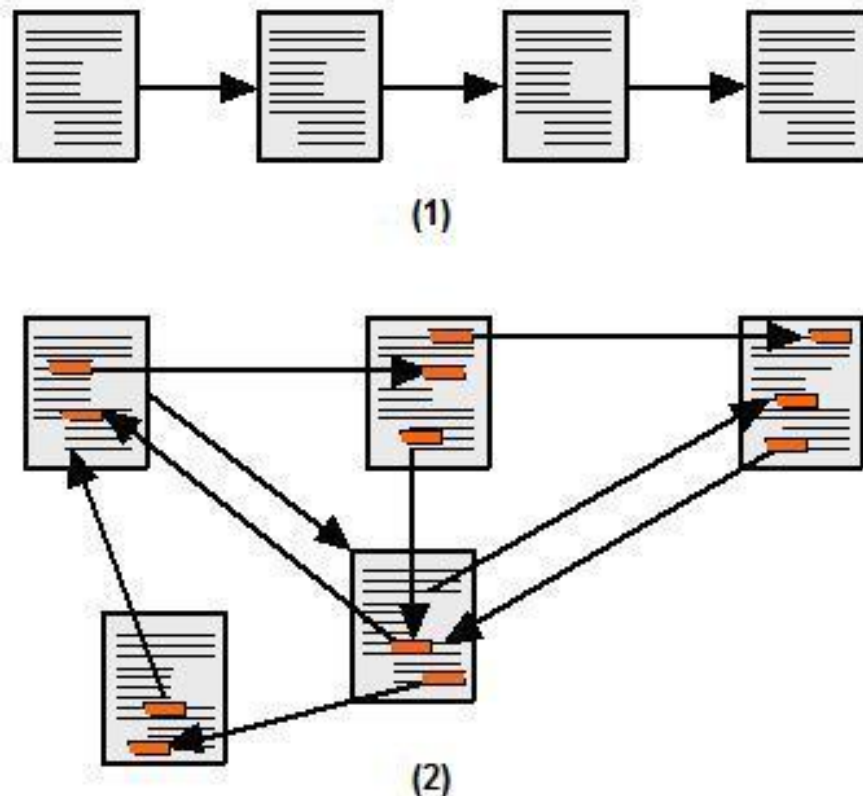
Το υπερκείμενο (hypertext) είναι τρόπος που οργανώνονται οι πληροφορίες, όταν βρίσκονται σε μορφή κειμένου, αλλά δεν έχουν ιεραρχημένη γραμμικότητά. Το υπερκείμενο χαρακτηρίζεται ως μη γραμμικό κείμενο σε αντίθεση με το βιβλίο που είναι γραμμικό. Το βιβλίο χαρακτηρίζεται ως γραμμικό επειδή ο αναγνώστης πρέπει να διαβάζει τη μία σελίδα ή παράγραφο μετά την άλλη για να κατανοήσει το περιεχόμενό του, χωρίς να του δίδεται η δυνατότητα να πλοηγείται ελεύθερα εντός αυτού. Ο αναγνώστης έτσι είναι αναγκασμένος να υπακούσει στους περιορισμούς που επιβάλλει ο συγγραφέας και κατ' επέκταση το βιβλίο ως μέσο.

⁷Domain Name System, 2014, http://el.wikipedia.org/wiki/Domain_Name_System

& Microsoft Technet, 2014, <https://technet.microsoft.com/en-us/library/cc958978.aspx>

Το υπερκείμενο ως μέσο ξεπερνάει τους περιορισμούς αυτούς επιτρέποντας την ελεύθερη πλοήγηση του αναγνώστη. Η πλοήγηση γίνεται με υπερσυνδέσμους. Λόγω του ότι οι υπερσυνδέσμοι είναι οι μηχανισμοί που αποκλείουν τη γραμμικότητα αποτελούν τη θεμελιώδη έννοια στο υπερκείμενο.

Ο Παγκόσμιος Ιστός στηρίζεται πάνω στις ιδέες του υπερκειμένου και υλοποιεί αυτόν τον τρόπο διασύνδεσης και οργάνωσης πληροφοριών .



Εικόνα 7:Γραμμικό κείμενο (1) και υπερκείμενο (2).

Τον όρο υπερκείμενο χρησιμοποίησε πρώτος ο Τεντ Νέλσον το 1965 στο βιβλίο του *Literary Machines*. Αν και ο όρος πρωτοεμφανίστηκε το 1965, οι ιδέες που αποτέλεσαν τις βάσεις του εμφανίζονται νωρίτερα. Το 1945 ο Βανέβαρ Βουζ περιγράφει την *Memex*, μια μηχανή που επέτρεπε τη διασύνδεση της ανθρώπινης και σήμερα, ο παγκόσμιος ιστός θεωρείται μία υλοποίηση της υποθετικής αυτής μηχανής⁸.

⁸ WorldWideWeb: Proposal for a HyperText Project, 1990, <http://www.w3.org/Proposal.HTML>

2.2.6 Ιστοσελίδα (Web-page)

Η Ιστοσελίδα (web page) είναι ένα ηλεκτρονικό αρχείο ή ένα σύνολο από ηλεκτρονικά αρχεία που υπάρχουν σε έναν ή και περισσότερους servers και παρουσιάζει κείμενα και εφαρμογές πολυμέσων. Τέτοια στοιχεία όπως κείμενο, εικόνες και φόρμες μπορούν να τοποθετηθούν στη σελίδα με τη χρήση γλωσσών σήμανσης υπερκειμένου. Οι ιστοσελίδες γράφονται σε γλώσσες σήμανσης κειμένου (markup languages). Η βασική από αυτές τις γλώσσες είναι η HTML (Hypertext Markup Language).

Πολλές ιστοσελίδες μαζί μας κάνουν έναν ιστότοπο (web site). Οι ιστοσελίδες συνδέονται η μία με την άλλη και μπορεί ο χρήστης να μεταβεί σε κάθεμία από αυτές, με συνδέσμους που εμφανίζονται διαφορετικά από το υπόλοιπο κείμενο ώστε να είναι γρήγορα αντιληπτό ότι πρόκειται για σύνδεσμο προς άλλη ιστοσελίδα.

Ο Τιμ Μπέρνερς-Λι συνδύασε τη διαδικτυακή επικοινωνία με υπερκείμενο δημοσιεύοντας την πρώτη ιστοσελίδα των Αύγουστο του 1991. Χρησιμοποιούσε τις πρώτες εκδόσεις της HTML που οι δυνατότητες της ήταν περιορισμένες δίνοντας στην ιστοσελίδα, μόνο τη βασική της δομή δηλαδή επικεφαλίδες και παραγράφους και την χρήση υπερκειμένου. Αυτή ήταν μια νέα μορφή επικοινωνίας διαφορετική από αυτές που υπήρχαν με την οποία οι χρήστες μπορούν να οδηγηθούν σε άλλες σελίδες.

Για να είναι προσβάσιμη μια ιστοσελίδα θα πρέπει να ακολουθεί τις αρχές προσβασιμότητας που είναι γνωστές ως WCAG (Web Content Accessibility Guidelines) 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 και δημιουργία ορισμών των διαδικτυακών υπηρεσιών με τρόπο ώστε να είναι αναγνωρίσιμες από άλλες διαδικτυακές υπηρεσίες σε διαφορετικές ιστοσελίδες. Τα πρότυπα για το semantic web τα θέτει το IEEE.
- Χρήση γλωσσών σήμανσης που ανταποκρίνονται σε ένα δημοσιευμένο DTD ή Schema.
- Να υπάρχει 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.
- Οι ιστοσελίδες διαχωρίζονται σε στατικές και δυναμικές⁹.

2.2.6.1 Στατική ιστοσελίδα

Στατική ιστοσελίδα (static web-page ή flat web-page) ονομάζεται μια ιστοσελίδα της οποίας το περιεχόμενο μεταφέρεται στον χρήστη ακριβώς στην μορφή που είναι αποθηκευμένο στον εξυπηρετητή ιστοσελίδων, σε αντίθεση με τις δυναμικές ιστοσελίδες όπου το περιεχόμενο δημιουργείται από μια εφαρμογή η οποία εκτελείται στον εξυπηρετητή ιστοσελίδων. Τα περιεχόμενα μιας στατικής ιστοσελίδας εμφανίζονται με την ίδια μορφή σε όλους τους χρήστες με την μορφή που είναι αποθηκευμένα στο σύστημα αρχείων του εξυπηρετητή ιστοσελίδων. Το περιεχόμενο και η διάταξη τους δεν αλλάζουν εκτός και αν ο προγραμματιστής κάνει αναβαθμίση στη σελίδα. Μια απλή HTML σελίδα θεωρείται στατικό περιεχόμενο. Οι στατικές ιστοσελίδες είναι αποθηκευμένες συνήθως σε μορφή HTML και για την μεταφορά τους χρησιμοποιούν το πρωτόκολλο HTTP.



Εικόνα 8:Στατική Ιστοσελίδα

⁹World Wide Web Consortium, 2015, <http://www.w3.org/standards/webdesign/>

Τα πλεονεκτήματα των στατικών ιστοσελίδων είναι ότι δεν χρειάζονται προγραμματιστικές δεξιότητες για να δημιουργήσει κάποιος μια τέτοια σελίδα, μπορεί να βρísκεται με εύκολο τρόπο κλωνοποιημένη σε παραπάνω από έναν εξυπηρετητές, δεν χρειάζεται ειδικό λογισμικό στον εξυπηρετητή ιστοσελίδων για την ανάρτηση ή δημοσίευση στατικών σελίδων και τέλος ότι η σελίδα μπορεί να είναι διαθέσιμη στον φυλλομετρητή κατευθείαν από ένα αποθηκευτικό χώρο, χωρίς να χρειάζεται να διαμεσολαβήσει ένα εξυπηρετητής ιστοσελίδων με κατάλληλο λογισμικό χειρισμό ιστοσελίδων, όπως για παράδειγμα οι δυναμικές σελίδες σε γλώσσα PHP, που χρειάζονται έναν διερμηνέα PHP, ο οποίος τρέχει στον εξυπηρετητή ιστοσελίδων και δημιουργεί δυναμικά την σελίδα, πριν αυτή αποσταλεί στον browser του χρήστη.

Αντίστοιχα κάποια από τα μειονεκτήματα τους είναι ότι δεν είναι διαδραστικές με τον χρήστη και ότι η διαχείριση πολλών στατικών ιστοσελίδων δεν είναι εύκολη χωρίς αυτόματα εργαλεία¹⁰.

2.2.6.2 Δυναμική ιστοσελίδα

Δυναμική ιστοσελίδα (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 είναι ένα από τα δομικά στοιχεία της νέας γενιάς του παγκόσμιου ιστού όπου η πληροφορία διαμοιράζεται σε πολλαπλές ιστοσελίδες.

Η γενικότερη ιδέα ξεκίνησε μετά την εξάπλωση και ανάπτυξη των ιστοσελίδων. Το HTTP ξεκίνησε να χρησιμοποιείται από το 1990 και η HTML άρχισε να χρησιμοποιείται για τον σχεδιασμό ιστοσελίδων από το 1996. Η ιδέα και μια μορφή δομής υπήρχε από πριν μίας και το 1990 υπήρχαν τα Bulletin Board Systems στα οποία οι χρήστες συνδέονταν μέσω μόντεμ και το περιεχόμενο που έβλεπαν δημιουργούνταν δυναμικά.

¹⁰"The Information Revolution", J. R. Okin. ISBN 0-9763857-4-0. Ed. Ironbound Press, 2005.



Εικόνα 9: Δυναμική ιστοσελίδα

Ο τρόπος λειτουργίας των δυναμικών ιστοσελίδων στον client γίνεται με τα ακόλουθα βήματα. Ένα λογισμικό τρέχει στον απομακρυσμένο εξυπηρετητή ιστοσελίδων. Ο χρήστης-πελάτης επισκέπτεται την σελίδα η οποία περιέχει ένα σενάριο εντολών το οποίο ενσωματώνει μια διαδραστικότητα με τον χρήστη, που γίνεται με χρήση πληκτρολογίου ή ποντικιού. Το περιεχόμενο της ιστοσελίδας δημιουργείται δυναμικά στον τοπικό υπολογιστή μετά την εκτέλεση του σεναρίου εντολών το οποίο λαμβάνεται από τον απομακρυσμένο εξυπηρετητή και η σελίδα επιστρέφεται σε μορφή HTML στον πελάτη. Υπάρχει περίπτωση το σενάριο να εκτελεστεί στο πελάτη ή στον εξυπηρετητή ή και στους δυο ταυτόχρονα.

Η σελίδα login.php, στο παράδειγμα της εικόνας 9, περιέχει σενάριο εντολών PHP το οποίο εκτελείται από τον διερμηνέα PHP στον εξυπηρετητή και χρησιμοποιώντας στοιχεία από την βάση MySQL αποστέλλεται πίσω η δυναμική σελίδα HTML.

Τέτοιες τεχνολογίες για την δημιουργία δυναμικών ιστοσελίδων στον εξυπηρετητή είναι η τεχνολογίες PHP, ASP, ColdFusion, Perl, WebDNA και άλλες όπως για παράδειγμα η γλώσσα σεναρίου εντολών JavaScript ή ActionScript χρησιμοποιείται για την τοπική δημιουργία δυναμικών ιστοσελίδων τεχνολογίας DHTML ή Flash οι οποίες συχνά χρησιμοποιούνται για να ενσωματωθούν ήχοι, κινούμενες εικόνες ή αλλαγές και διαμόρφωση του κείμενου της σελίδας. Η γλώσσα σεναρίου έχει την δυνατότητα να καλεί απομακρυσμένα σενάρια εντολών από και προς τον εξυπηρετητή ιστοσελίδων.

Πολλές φορές οι σελίδες είναι αποθηκευμένες στατικά σε κρυφή-προσωρινή μνήμη και αποστέλλονται κατευθείαν στον χρήστη. Με αυτό τον τρόπο αποφεύγεται

τη υπερφόρτωση του εξυπηρετητή ιστοσελίδων και καλείται η μηχανή που είναι υπεύθυνη για την δημιουργία δυναμικών ιστοσελίδων μόνο όταν το αποθηκευμένο περιεχόμενο έχει αλλάξει από τον προγραμματιστή.

Τα πλεονεκτήματα των δυναμικών σελίδων είναι ότι το περιεχόμενο μπορεί να ανανεώνεται από ανθρώπους που δεν γνωρίζουν προγραμματισμό και ετσι υπαρχει η δυνατοτητα πιο συχνης ανανewσης. Ο αριθμός των σελίδων που μπορεί να περιέχει είναι απεριόριστος άρα και πολύ περιεχόμενο από τις στατικές ιστοσελίδες και τέλος είναι εύκολη η διαχειριστείς τους χρήστες και τους διαχειριστές μιας τέτοιας ιστοσελίδας.

Αντίστοιχα τα μειονεκτήματα τους είναι ότι η κατασκευή μιας δυναμικής ιστοσελίδας χρειάζεται περισσότερο χρόνο για την ολοκλήρωση της. Για τη σχεδίαση της χρειάζεται παραπάνω κόστος από μια στατική ιστοσελίδα μιας και χρειάζονται εξειδικευμένες γνώσεις όπως επίσης και το κόστος φιλοξενίας είναι επίσης αυξημένο μιας και είναι ανάλογο των μεγαλύτερων απαιτήσεων. Τέλος λόγω των διαφορετικών σεναρίων που τρέχουν ίσως σε παραπανω από ένα εξυπηρετητές εκτελείται πιο αργά από μία στατική σελίδα¹¹.

2.2.7 Γλώσσες σήμανσης

Μία γλώσσα σήμανσης (markup language) είναι ένα σύνολο εντολών που επιτρέπουν την προσθήκη πληροφορίας στο περιεχόμενο μιας πηγής δεδομένων . Το περιεχόμενο μπορεί να είναι κείμενο, εικόνα ή οποιαδήποτε άλλη μορφή ηλεκτρονικής πληροφορίας. Οι εντολές των γλωσσών σήμανσης ονομάζονται ετικέτες (tags) και είναι απλές και κατανοητές. Η δομη τους δεν είναι τοσο αυστηρη και δεν μπορούν να εκτελεσουν σύνθετους υπολογισμους αντιθετα με τις κλασσικές γλώσσες προγραμματισμού όπως η C και η java. Για να εκτελεστον δεν χρειάζονται compilers, και γενικά αυτό που κάνουν είναι να περιγράφουν πώς θα γίνει μια εργασία, παρά να την εκτελούν¹².

2.2.8 Standard Generalized Markup Language

Η SGML (Standard Generalized Markup Language), αποτελεί το διεθνές πρότυπο γενικευμένης γλώσσας σήμανσης. Καθορίζει το πώς αναπαριστается η

¹¹ "The Information Revolution", J. R. Okin. ISBN 0-9763857-4-0. Ed. Ironbound Press, 2005.

¹² Σαμψών, Δ. (2003). Η γλώσσα σήμανσης XML. Πανεπιστήμιο Πειραιώς.

πληροφορία όπως και τις μεθόδους της, οι οποίες είναι ανεξάρτητες από οποιοδήποτε σύστημα ή μηχανήμα. Οι μέθοδοι αυτοί έχουν καθοριστεί ώστε να είναι κατανοητοί από ανθρώπους και μηχανές. Όλες οι γλώσσες σήμανσης οι οποίες συμμορφώνονται με τους κανόνες που καθορίζει η SGML είναι εφαρμογές της. Επιπλέον, η SGML είναι μία μετα-γλώσσα παρέχει τη δυνατότητα καθορισμού νέων γλωσσών σήμανσης που είναι υποσύνολά της. Αν και η SGML έχει ουσιαστικό ρόλο στον καθορισμό εγγράφων σύμφωνα με τις προδιαγραφές του προτύπου επικοινωνίας World Wide Web Consortium (W3C), είναι πολύπλοκη για τους σκοπούς του διαδικτύου¹³.

2.2.9 HTML (HyperText Markup Language)

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

Το 1980, ο φυσικός Τιμ Μπέρνερς Λι επινόησε το ENQUIRE που ήταν ένα σύστημα για την χρήση και διαμοιρασμού εγγράφων για τους εργαζομένους του CERN. Έπειτα πρότεινε ένα σύστημα εξ ολοκλήρου βασισμένο στο διαδίκτυο, το οποίο θα ενθαρρυνόταν η χρήση υπερκειμένου. Έτσι, έφτιαξε μια πρώτη μορφή της HTML που ήταν έντονα επηρεασμένη από την SGML, έγραψε ένα browser και λογισμικό εξυπηρετητή στα τέλη του 1990. Από το 1996 και μετά, οι προδιαγραφές της HTML τηρούνται, από το World Wide Web Consortium (W3C). Το 2000 η HTML έγινε επίσημα παγκόσμιο πρότυπο (ISO/IEC 15445:2000).

Η σήμανση HTML αποτελείται από μερικά βασικά συστατικά, μαζί με τα στοιχεία και τις ιδιότητές 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 δείχνει στον browser πως θα εμφανίζει την σελίδα.

Τα έγγραφα HTML αποτελούνται από στοιχεία HTML τα οποία στην γενικευμένη τους μορφή έχουν τρία συστατικά: το πρώτο, ένα ζεύγος από ετικέτες, την «ετικέτα εκκίνησης» και την «ετικέτα τερματισμού», το δεύτερο, ιδιότητες μέσα στην ετικέτα εκκίνησης, και τέλος το κείμενο και το περιεχόμενο μεταξύ των ετικετών, το οποίο μπορεί να περιλαμβάνει και άλλα στοιχεία φωλιασμένα μέσα του.

¹³ Σαμψών, Δ. (2003). Η γλώσσα σήμανσης XML. Πανεπιστήμιο Πειραιώς.

Το στοιχείο HTML μπορεί να έχει οποιαδήποτε μορφή και να υπάρχει ανάμεσα στις ετικέτες εκκίνησης και τερματισμού. Τέλος, κάθε ετικέτα περικλείεται σε σύμβολα «μεγαλύτερο από» και «μικρότερο από», δηλαδή < και >.

Τα στοιχεία της HTML χρησιμοποιούνται για να χτίσουν όλους του ιστότοπους. Η HTML επιτρέπει να ενσωματωθούν εικόνες και άλλα αντικείμενα μέσα στη σελίδα, οπωρ επίσης και για την εμφάνιση διαδραστικών φόρμων. Παρέχει τις μεθόδους δημιουργίας εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης αυτού, καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, συνδέσμους και άλλα. Επιτρέπει την ενσωμάτωση σεναρίων εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

```
25 </head>
26 <body text="#000000
    bgcolor="#FFFFFF">
27 <table width="1000"
28     <tr>
29         <td width="200"
30         </td>
31         <td valign="top"
32             <div align="c
33             </div>
34             <p class="Boc
35             <h1 class="He
36             <p class="Cap
Entertainment</a>
37         | <a href=
```

Εικόνα 10: Δείγμα κώδικα HTML

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 οι ιδιότητες ενός στοιχείου, τότε χρησιμοποιούνται οι προεπιλογές.

Στην HTML ορίζονται τύποι δεδομένων που αφορούν για το περιεχόμενο των στοιχείων, όπως σεναρία εντολών ή stylesheet, και μια πληθώρα τύπων για τις τιμές

των ιδιοτήτων, συμπεριλαμβανομένων των ID, των name, των URI, διαφόρων αριθμών και μονάδων μήκους, γλωσσών, τύπων αρχείων πολυμέσων, χρωμάτων, κωδικοποιήσεων χαρακτήρων, ημερομηνιών κλπ.

Τα έγγραφα HTML πρέπει να αρχίζουν με μια δήλωση τύπου εγγράφου η οποία ανεπίσημα λέγεται και «doctype». Η δήλωση αυτή βοηθά τους browser να κατανοήσουν πώς πρέπει να διαβάσουν το περιεχόμενο του εγγράφου και πώς να κάνουν την παρουσίαση του μετά. Ο σκοπός του doctype είναι να επιτραπεί η ανάλυση και η επιβεβαίωση των εγγράφων HTML από εργαλεία SGML τα οποία βασίζονταν στο Document Type Definition (DTD). Το DTD περιέχει μια γραμματική σε γλώσσα μηχανής, που καθορίζει τι επιτρέπεται και τι απαγορεύεται δηλαδή τους περιορισμούς που υπάρχουν μέσα στο έγγραφο.

Ο Παγκόσμιος Ιστός αποτελείται κυρίως από αρχεία HTML τα οποία μεταφέρονται από εξυπηρετητές προς browsers μέσω της χρήσης του πρωτοκόλλου HTTP. Ωστόσο, μια που το ίδιο πρωτόκολλο μπορεί επιπλέον να μεταφέρει εικόνες, ήχο και άλλο περιεχόμενο, μαζί με το ίδιο το αρχείο, μεταφέρονται και μερικές επιπλέον πληροφορίες, ή μεταδεδομένα, όπως και η κωδικοποίηση χαρακτήρων που χρησιμοποιείται.

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) και υιοθετήθηκε ως αρχικό βήμα για τις εργασίες της νέας ομάδας εργασίας HTML του W3C το 2007¹⁴.

2.2.10 XHTML(eXtensible HyperText Markup Language)

Τα αρχικά XHTML σημαίνουν eXtensible HyperText Markup Language, δηλ. εκτεταμένη γλώσσα σήμανσης υπερκειμένου και είναι σχεδόν ίδια με την HTML 4.01. Η XHTML αποτελεί μια αυστηρότερη και καθαρότερη έκδοση (version) της HTML. Ουσιαστικά η XHTML είναι η HTML ορισμένη σαν μια εφαρμογή της XML.

¹⁴Γενικά σχετικά με την HTML, 2015, <http://www.it.uom.gr/project/HTML2/tut0.HTML> & Μαθήματα Κατασκευής Web Σελίδων με τη Γλώσσα HTML, 2005, <http://www.eeei.gr/odhgos/HTMLfaq.htm> & HTML 5. A vocabulary and associated APIs for HTML and XHTML, 2014, <http://www.w3.org/TR/HTML5/>

Είναι στάνταρτ του Web και η XHTML 1.0 έγινε επίσημα παγκόσμιο πρότυπο του W3C το 2000.

Το βασικό χαρακτηριστικό είναι ότι η XHTML αποτελεί μια αναδιάταξη της HTML στην XML και μπορεί να χρησιμοποιηθεί με τους υπάρχοντες φυλλομετρητές ακολουθώντας μερικούς απλούς κανόνες.

Η XHTML αποτελεί τη νέα γενιά της HTML, και δεν είναι πολύ διαφορετική. Μια από τις σημαντικότερες διαφορές ανάμεσα στις δύο γλώσσες είναι ότι τα στοιχεία της XHTML πρέπει να είναι σωστά φωλιασμένα. Στην HTML μερικά στοιχεία μπορεί να μην είναι σωστά φωλιασμένα το ένα μέσα στο άλλο. Αντίθετα, στην XHTML όλα τα στοιχεία πρέπει να φωλιάζουν σωστά το ένα μέσα στο άλλο. Επιπλέον τα έγγραφα της XHTML πρέπει να είναι σωστά μορφοποιημένα. Όλα τα στοιχεία της XHTML πρέπει να είναι φωλιασμένα μέσα στο αρχικό (root) στοιχείο <HTML> και όλα τα άλλα στοιχεία μπορούν να έχουν στοιχεία παιδιά, τα οποία πρέπει να είναι σε ζευγάρια και σωστά φωλιασμένα μέσα στο πατρικό τους στοιχείο. Επίσης τα ονόματα των tags πρέπει να γράφονται με πεζά γράμματα. Αυτό γίνεται για το λόγω ότι τα έγγραφα της XHTML αποτελούν εφαρμογές της XML και η XML ξεχωρίζει τους πεζούς από τους κεφαλαίους χαρακτήρες. Αυτό σημαίνει ότι tags όπως τα
 και
 αντιμετωπίζονται σαν διαφορετικά tags. Τέλος όλα τα μη κενά στοιχεία της XHTML πρέπει να είναι κλειστά. Τα μη κενά στοιχεία πρέπει να έχουν ένα τελικό tag ή το αρχικό tag πρέπει να τελειώνει με τους χαρακτήρες />.

Για να κάνουμε την XHTML να είναι συμβατή με τους σημερινούς browsers, πρέπει να προσθέσουμε ένα επιπλέον κενό πριν από το σύμβολο /, ως εξής :
 και <hr />.

Η συγγραφή κώδικα XHTML χρειάζεται να έχει μια καθαρή σύνταξη HTML. Όλα τα έγγραφα της XHTML πρέπει να έχουν μια δήλωση DOCTYPE, τα στοιχεία HTML, head και body και το στοιχείο title που πρέπει να βρίσκεται μέσα στο στοιχείο head. Η δήλωση DOCTYPE δεν αποτελεί μέρος του εγγράφου της XHTML και έτσι δεν πρέπει να έχει tag κλεισίματος.

Η HTML ορίζει το χαρακτηριστικό name για τα στοιχεία a, applet, frame, iframe, img και map. Στην XHTML το χαρακτηριστικό name δεν χρησιμοποιείται και χρησιμοποιούμε αντι για αυτο το id. Για να υπάρχει συμβατότητα με τους πιο παλιούς browsers, πρέπει να χρησιμοποιούμε και το name και το id, με τις ίδιες τιμές χαρακτηριστικών Το χαρακτηριστικό lang εφαρμόζεται σχεδόν σε κάθε στοιχείο της XHTML και καθορίζει τη γλώσσα του περιεχομένου μέσα σ' ένα στοιχείο. Αν

χρησιμοποιούμε το χαρακτηριστικό lang σ' ένα στοιχείο, είναι αναγκαιο να προστεθεί το χαρακτηριστικό XML:lang

Βασικοί συντακτικοί κανόνες της XHTML είναι ότι τα ονόματα των χαρακτηριστικών πρέπει να γράφονται με πεζά γράμματα, οι τιμές των χαρακτηριστικών πρέπει να περικλείονται από εισαγωγικά , δεν επιτρέπεται η ελαχιστοποίηση των χαρακτηριστικών, το χαρακτηριστικό id αντικαθιστά το χαρακτηριστικό name, και τέλος ότι το XHTML DTD ορίζει τα υποχρεωτικά στοιχεία.

Το XHTML standard ορίζει τρεις ορισμούς τύπου εγγράφου (Document Type Definitions), όπου το πιο κοινό είναι το XHTML Transitional.

Το <!DOCTYPE> είναι υποχρεωτικό και ένα έγγραφο της XHTML αποτελείται από τα εξής τρία βασικά τμήματα :

- Το DOCTYPE
- Το Head
- Το Body

Η βασική δομή του εγγράφου είναι η εξής :

```
<!DOCTYPE ... >
<HTML>
    <head> ... </head>
    <body> ... </body>
</HTML>
```

Η δήλωση DOCTYPE απαιτείται να είναι πάντα η πρώτη γραμμή σ' ένα XHTML έγγραφο.

Το παρακάτω παράδειγμα αποτελεί ένα απλό έγγραφο της XHTML. Η δήλωση DOCTYPE ορίζει τον τύπο εγγράφου και το υπόλοιπο του εγγράφου μοιάζει με την HTML ¹⁵:

```
<!DOCTYPE HTML
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
```

¹⁵ ΚΕΝΤΡΟ ΠΛΗ.ΝΕ.Τ. Ν. ΦΛΩΡΙΝΑΣ Η Γλώσσα Προγραμματισμού XHTML, 2015, <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-XHTML.HTML>

```
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<HTML>
    <head>
        <title> απλό έγγραφο </title>
    </head>
    <body>
        <p> μια απλή παράγραφος </p>
    </body>
</HTML>
```

2.2.11 Web browser

Ο Web browser ή φυλλομετρητής ιστοσελίδων ή περιηγητής Ιστού, είναι ένα λογισμικό που επιτρέπει στον χρήστη να προβάλλει, και να αλληλεπιδρά με, κείμενα, εικόνες και άλλες πληροφορίες συνήθως αναρτημένες σε μια ιστοσελίδα ενός ιστότοπου σε επίπεδο Παγκόσμιου Ιστού ή σε επίπεδο τοπικού δικτύου. Στο κείμενο και στις εικόνες μιας ιστοσελίδας υπάρχει περίπτωση να εμπεριέχουν υπερσυνδέσμους προς άλλες ιστοσελίδες για τον ίδιο ή διαφορετικό ιστότοπο. Η δουλειά του browser είναι να δίνει στον χρήστη την γρήγορη και εύκολη πρόσβαση σε πληροφορίες που βρίσκονται σε διάφορες ιστοσελίδες και ιστότοπους εναλλάσσοντας τις ιστοσελίδες μέσω των υπερσυνδέσμων.

Ο σκοπός ενός web browser είναι να διαβάζει τα έγγραφα HTML να μην εμφανίζει τις ετικέτες της αλλά να τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας και τα συνθέτει σε περιεχόμενο που μπορεί κανείς να διαβάσει ή να ακούσει και αναφέρονται επίσης σε στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού¹⁶.

2.2.12 CSS

CSS σημαίνει Cascading Style Sheets και είναι μια γλώσσα που χρησιμοποιείται για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός

¹⁶ Jacobs, Ian; Walsh, Norman (15 December 2004). "URI/Resource Relationships". Architecture of the World Wide Web, Volume One.

ιστοτόπου. Η CSS αναπτύσσει αισθητικά μια ιστοσελίδα δηλαδή διαμορφώνει περισσότερα χαρακτηριστικά, να αλλάζει χρώματα, στοίχιση Με αυτή ορίζουμε το χρώμα των γραμμάτων, το μέγεθος της γραμματοσειράς, την γραφή (bold, underline, κτλ.), το χρώμα του φόντου και των αντικειμένων, τις διαστάσεις των αντικειμένων, τις αποστάσεις και άλλες ιδιότητες που μπορεί να έχουν τα στοιχεία μιας ιστοσελίδας και δίνει περισσότερες δυνατότητες όταν σχετίζεται με την HTML.



Εικόνα 11: CSS

Η χρήση CSS κάνει πολύ εύκολη την διαχείριση της εμφάνισης των σελίδων. Αν σε μια ιστοσελίδα, χωρίς χρήση CSS, πρέπει να αλλάξει το χρώμα του φόντου σε όλες τις επικεφαλίδες όλων των πινάκων θα πρέπει να αλλάξει σε κάθε μια επικεφαλίδα και να οριστεί το χρώμα. Ενώ σε μια σελίδα με χρήση CSS θα χρειαστεί να αλλάξει το χρώμα αυτό μια μόνο φορά και αυτό θα εφαρμοστεί για όλες τις επικεφαλίδες της σελίδας. Φυσικά υπάρχει η δυνατότητα αλλαγής στυλ, όχι μόνο για στοιχεία μιας σελίδας, αλλά για στοιχεία όλων των σελίδων κάποιου website.

Σήμερα σχεδόν όλες οι ιστοσελίδες χρησιμοποιούν CSS. Η χρήση των στυλ κάνει την σχεδίαση σελίδων πολύ πιο εύκολη δημιουργώντας έτσι websites των οποίων η διαχείριση είναι ευκολη και γρηγορη.

Τα πλεονεκτήματα των φύλλων στυλ CSS είναι ο διαχωρισμός του περιεχομένου από την σχεδίαση και η ελαχιστοποίηση του χρόνου για τις αλλαγές στην σχεδίαση αφού όλα τα στυλ των στοιχείων περιέχονται σε ένα αρχείο. Γι αυτό

δημιουργούνται μικρότερου μεγέθους αρχεία με αποτέλεσμα τον καθαρότερο και αυτούσιο κώδικα HTML. Επιπλέον υπάρχει ομοιόμορφη εμφάνιση των ιστοσελίδων που συνδέονται με το συγκεκριμένο CSS αρχείο, αφού τα στοιχεία δεν αλλάζουν και μεγαλύτερη ταχύτητα εμφάνισης της ιστοσελίδας σε όλους τους Web Browsers.

Υπάρχει πιστοποίηση από την W3C, το μεγαλύτερο οργανισμό Web Standards ο οποίος ενθαρρύνει τη χρήση των CSS για σκοπούς παρουσίασης του περιεχομένου¹⁷.

2.2.13 PHP

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία dynamic;vn σελίδων web. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού ,ώστε να παραχθεί σε πραγματικό χρόνο το ζητούμενο περιεχόμενο, το οποί και στέλνεται στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.



Εικόνα 12:Php

Η ιστορία της PHP ξεκινά το 1994, όταν ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi. Η γλώσσα τότε ονομάστηκε PHP από τα αρχικά Personal Home Page.

Ορισμένα από τα πλεονεκτήματα της php είναι η υψηλή απόδοση της, η ευκολία διασύνδεσης με πολλά διαφορετικά συστήματα βάσεων δεδομένων, υπάρχουν ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες διαδικασίες στον ιστο. Το κόστος της είναι μηδενικό, αφού διανέμεται δωρεάν, ευκολία μάθησης και χρήσης, τα προγράμματα που έχουν γραφεί σε php μεταφέρονται εύκολα, ο πηγαίος κωδικας είναι διαθέσιμος ανα πασα ωρα, τρέχει σε πολλά διαφορετικά λειτουργικά συστήματα, λειτουργεί στον Server. Δεν χρειάζεται ο client, εάν χρησιμοποιείται CSS, πολύ εύκολη η πρόσβαση σε άλλα web-based εργαλεία μέσω της PHP πχ Google maps.

¹⁷Wlearn, 2014, <http://www.wlearn.gr/index.php/CSS-84> & CSS, 2015, <http://el.wikipedia.org/wiki/CSS>

Αντίστοιχα τα μειονεκτήματα της είναι η μειωμένη ασφάλεια. Επειδή είναι ανοιχτή γλώσσα, και όλοι οι άνθρωποι είτε προγραμματιστές, είτε χρήστες μπορούν να δουν τον πηγαίο κώδικά της και εφόσον υπάρχουν σφάλματα στον πηγαίο κώδικα, μπορεί να χρησιμοποιηθεί από τους ανθρώπους για να εξερευνηθεί κάποιες αδυναμίες της. Επίσης δεν είναι κατάλληλη για μεγάλες εφαρμογές. Είναι αρκετά δύσκολο να διατηρηθεί, δεδομένου ότι δεν είναι σπονδυλωτή (modular). Για να δημιουργηθεί κάτι παραπάνω από μια απλή HTML / CSS σελίδα, θα πρέπει να προστεθεί κώδικας κάποιας άλλης client-side γλώσσας όπως JavaScript ή Java. Η PHP τείνει να εκτελεί πιο αργά άλλες γλώσσες, όπως από την C και την Assembly¹⁸.

2.2.14 Javascript

Η JavaScript (JS) αρχικά αποτέλεσε μέρος των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται.

Η JavaScript είναι μια prototype-based γλώσσα, δυναμική και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη και βασισμένη στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες. Οι βασικές αρχές σχεδιασμού της JavaScript έχουν τις ρίζες τους στις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστραφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού.

Η JavaScript εφαρμόζεται και σε πληθώρα εφαρμογών πέρα από τις ιστοσελίδες. Τέτοια παραδείγματα είναι τα έγγραφα PDF, οι site-specific browsers και οι μικροεφαρμογές της επιφάνειας εργασίας. Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side) και περιλήφθηκε σε διάφορα προγράμματα περιήγησης στο Web.

Η γλώσσα προγραμματισμού JavaScript δημιουργήθηκε από τον Brendan Eich και το πρωτο της όνομα ήταν Mocha. Αργότερα μετονομάστηκε σε LiveScript,

¹⁸PHP Manual, 2015, <http://php.net/manual/en/index.php> & The greek technology team, 2011, <http://www.techteam.gr/wiki/PHP>

και τέλος σε JavaScript, κυρίως επειδή η ανάπτυξή της επηρεάστηκε πιο πολύ από τη γλώσσα προγραμματισμού Java. LiveScript ήταν το επίσημο όνομα της γλώσσας όταν για πρώτη φορά κυκλοφόρησε στην αγορά σε beta εκδόσεις με το πρόγραμμα περιήγησης στο Web, Netscape Navigator εκδοχή 2.0 τον Σεπτέμβριο του 1995.

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του client, που ήταν ο browser του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του πηγαίου κώδικα Javascript και η αναπαραγωγή του τελικού περιεχομένου HTML δεν γίνεται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα από άλλες γλώσσες, όπως η PHP που εκτελούνται στο διακομιστή η Javascript εμφανίζεται με τη μορφή ενός μοντέλου προγραμματισμού βασισμένο στα γεγονότα (events).

Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά οι δύο αυτές γλώσσες δε έχουν σχέση μεταξύ τους και έχουν πολύ διαφορετική σημασιολογία. Η Java, είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Η χρήση της λέξης "Java" στο όνομα της γλώσσας έχει πιο πολλή σχέση με το προφίλ της JavaScript που έπρεπε να έχει και λιγότερο με κάποια πιθανή συμβατότητα ή κάποια άλλη στενή σχέση με τη Java. Μεγάλο ρόλο έπαιξε και ότι η Java και η Javascript έχουν επηρεαστεί σημαντικά από τη γλώσσα C, ειδικά στο συντακτικό, ενώ είναι και οι δύο αντικειμενοστραφείς γλώσσες.



Εικόνα 13: Javascript

Ο κώδικας Javascript μιας σελίδας περικλείεται από τις ετικέτες της HTML `<script type="text/javascript">` και `</script>`.

Επειδή ο κώδικας JavaScript μπορεί να εκτελεστεί τοπικά σε έναν browser και όχι σε έναν απομακρυσμένο διακομιστή, ανταποκρίνεται στις ενέργειες των χρηστών με μεγάλη ταχύτητα, κάνοντας την JavaScript πιο ευέλικτη. Επιπλέον, ο κώδικας

JavaScript μπορεί να ανιχνεύσει τις ενέργειες ενός χρήστη, κάτι που η HTML δεν μπορεί να κάνει μόνη της, όπως λόγω χάρη την ανίχνευση πατημένων πλήκτρων ή την θέση του ποντικιού στην ιστοσελίδα¹⁹.

2.2.15 Ajax(Asynchronous JavaScript and XML)

Τεχνικά, όλες οι υλοποιήσεις ιστοσελίδων στα πρότυπα του Web 2.0 πραγματοποιείται με την χρήση τεχνολογιών AJAX (Asynchronous JavaScript and XML) από την πλευρά του πελάτη, οι οποίες προσδίδουν στην εφαρμογή χαρακτήρα αλληλεπίδρασης.

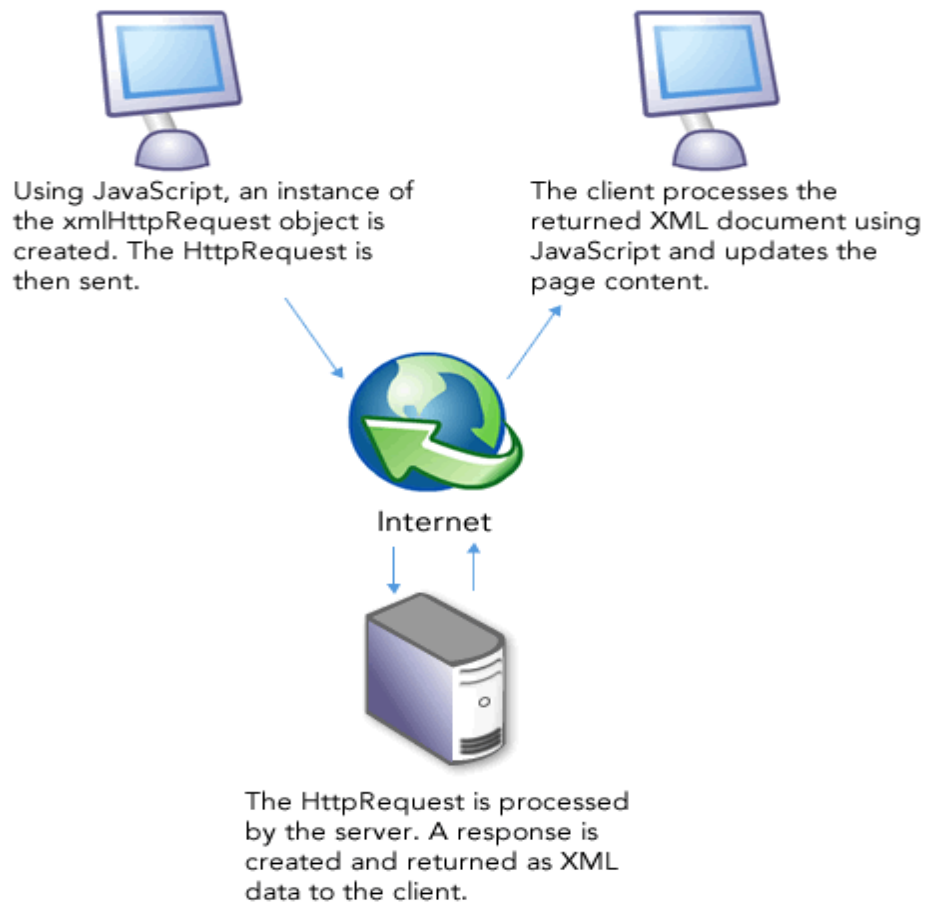
Το δυνατό χαρακτηριστικό της τεχνολογίας AJAX είναι το γεγονός ότι οι εφαρμογές μπορούν να λαμβάνουν δεδομένα από τον διακομιστή ασύγχρονα στο παρασκήνιο χωρίς να παρεμβαίνουν στην υπόλοιπη σελίδα, δίνοντας έτσι την εντύπωση μιας desktop εφαρμογής στον τελικό χρήστη. Τα δεδομένα συνήθως λαμβάνονται μέσω του αντικειμένου XMLHttpRequest, του οποίου παρά δε το όνομα, δεν καθίσταται απαραίτητη η χρήση της XML, ούτε οι αιτήσεις στον διακομιστή να είναι ασύγχρονες.

Το αντικείμενο XMLHttpRequest είναι ένα API(Application programming interface) που υποστηρίζει η javascript και το οποίο προσφέρει όλες τις συναρτήσεις (functions), χαρακτηριστικά (properties) και συμβάντα (events) για τη σωστή λειτουργία μιας εφαρμογής ajax. χρησιμοποιείται για την ασύγχρονη μεταφορά δεδομένων από τον server στον browser, μεταφορά που πραγματοποιείται στο παρασκήνιο χωρίς ο επισκέπτης της σελίδας να ανανεώσει το περιεχόμενο της.

Η χρήση της τεχνολογίας AJAX έχει συνεισφέρει στην ραγδαία εξέλιξη των διαδραστικών και δυναμικών εφαρμογών σε ιστοσελίδες. Αξίζει να σημειωθεί ότι η τεχνολογία AJAX δεν είναι μια τεχνολογία από μόνη της, αλλά ένας συνδυασμός τεχνολογιών. Η AJAX χρησιμοποιεί έναν HTML και CSS για την σήμανση της δομής και της εμφάνισης. Η χρήση της JavaScript σε συνδυασμό με το αντικείμενο XMLHttpRequest έρχεται να καλύψει τον χρόνο που κάνει μια σελίδα για να φορτώσει (page loading). Δηλαδή, με την χρήση της τεχνολογίας αυτής, δεν υφίσταται page loading, παρά μόνο φόρτωση συγκεκριμένης πληροφορίας (partial loading). Το παραπάνω, μας προσφέρει μεγαλύτερη ταχύτητα και λιγότερο bandwidth

¹⁹ Harris, Andy (2001). JavaScript Programming for the Absolute Beginner. Premier Press. ISBN 0-7615-3410-5.
Goodman, Danny; Eich, Brendan (2001). JavaScript Bible. John Wiley & Sons. ISBN 0-7645-3342-8.

– traffic, αφού πλέον δεν φορτώνεται ολόκληρη η σελίδα, αλλά μόνο το κομμάτι που θέλουμε να ανανεώσουμε. Το αντικείμενο XMLHttpRequest αρχικά δημιουργήθηκε και χρησιμοποιήθηκε από την Microsoft αργότερα όμως και μετά την ευρεία χρήση από την Google το αντικείμενο πέρασε στην υποστήριξη της W3C.

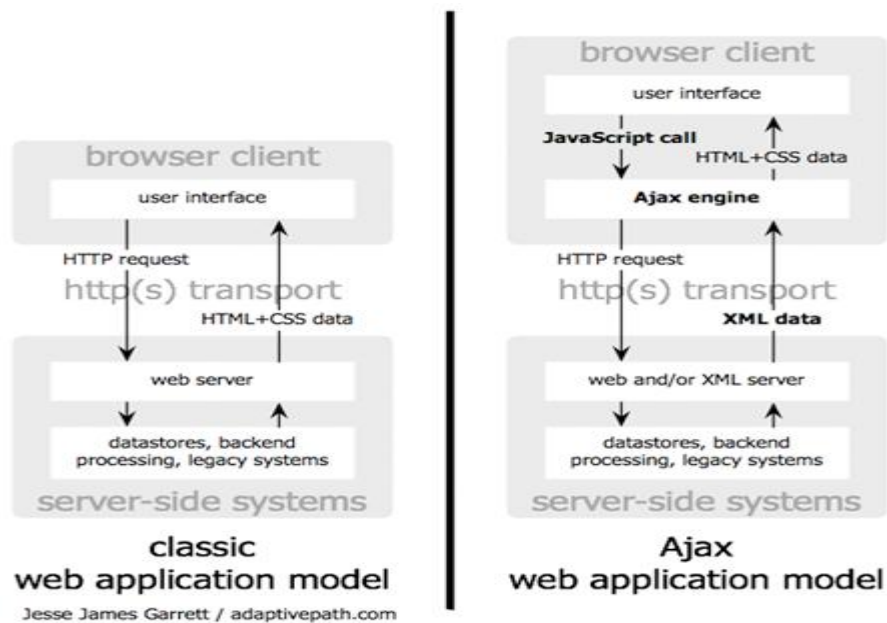


Εικόνα 14:Ajax

Οι Εφαρμογές AJAX, μπορούν να αυξήσουν σημαντικά τον αριθμό των αιτήσεων προς τον διακομιστή καθώς και την περαιτέρω επικοινωνία με βάσεις δεδομένων. Το παραπάνω μπορεί να οδηγήσει σε αύξηση του χρόνου απόκρισης της εφαρμογής (application response time) ανάλογα με τον αριθμό των αιτήσεων

Συγκεντρωτικά λοιπόν τα πλεονεκτήματα σε σχέση με άλλες web εφαρμογές ποικίλουν. Σε μια κλασσική web εφαρμογή, όταν ο web server στέλνει την σελίδα στον browser, μπορούν να χρησιμοποιηθούν πολλαπλά connection threads για να επιταχύνει η μεταφορά. Αυτό αξιοποιεί η καινούρια τεχνολογία. Οπότε σε μια σελίδα φορτώνουμε τα βασικά scripts και CSS αρχεία μία φορά και έπειτα αξιοποιούμε τις

πολλαπλές συνδέσεις για να μεταφέρουμε στη σελίδα μας το περιεχόμενο που επιθυμούμε. Επιπλέον το interface είναι πολύ πιο διαδραστικό, διότι δεν χρειάζεται να ξαναφορτίσει ολόκληρο η σελίδα. Απλά μεταφέρεται μικρότερη και η απαραίτητη πληροφορία που ζητήθηκε. Επίσης ο χριστής έχει την αίσθηση ότι οι ενάργειες του έχουν άμεσο αποτέλεσμα. Η κίνηση από και προς το server μειώνεται σημαντικά, οπότε μειώνεται και η χρήση του bandwidth. Σε περίπτωση που παρουσιαστεί σφάλμα τότε επηρεάζεται μόνο το συγκεκριμένο τμήμα χωρίς να επηρεάζει το υπόλοιπα και να χαθούν κρίσιμα ίσως δεδομένα. Στην περίπτωση που ο χρηστής υποβάλει μια φόρμα δεν χρειάζεται να περιμένει όλη την σελίδα να ξαναφορτίσει ώστε να αποσταλούν τα δεδομένα της φόρμας. Αντί αυτού μπορεί να συνεχίσει να δουλεύει στην σελίδα, ενώ τα δεδομένα αποστέλλονται.



Εικόνα 15:Κλασσική ιστοσελίδα σε σύγκρισή με ιστοσελίδα με Ajax

Παρά τα θετικά που έχει να προσφέρει η τεχνολογία AJAX, υπάρχουν πάντα και τα αρνητικά της χρήσης αυτής. Οι διεπαφές που χρησιμοποιούν AJAX είναι συχνά δυσκολότερες στην υλοποίηση συγκριτικά με τις στατικές ιστοσελίδες – καθαρά και μόνο από το γεγονός ότι είναι δυναμικές. Θεωρείται πιο δύσκολη η ανάπτυξη λόγω των πολλών εμπλεκόμενων τεχνολογιών. Οι σελίδες που δημιουργούνται δυναμικά με την χρήση AJAX δεν καταχωρούνται αυτόματα στο ιστορικό ενός browser, επομένως η δυνατότητα του browser να γυρίσει μια σελίδα πίσω δεν είναι εφικτή με τον τρόπο που γνωρίζουμε, μιας και θα ξεκινήσει την εφαρμογή από την αρχή. Για τον ίδιο λόγο με το παραπάνω, δεν μπορούμε να

τοποθετήσουμε σελιδοδείκτη σε μια δυναμική ιστοσελίδα, παρά μόνο εάν αυτή έχει υλοποιηθεί με κάποιον συγκεκριμένο τρόπο, αλλά και πάλι δεν έχουμε το ίδιο αποτέλεσμα με μια στατική ιστοσελίδα. Επειδή τα περισσότερα web crawlers που χρησιμοποιούνται από μηχανές αναζήτησης για την εύρεση ιστοσελίδων και πληροφοριών σχετικά με αυτές, δεν υποστηρίζουν JavaScript, η εύρεση δυναμικών ιστοσελίδων με AJAX καθίσταται αρκετά δύσκολη καθώς ο σχεδιαστής της εφαρμογής θα πρέπει να μεριμνήσει για να εμφανίζει τις κατάλληλες σελίδες και χωρίς την χρήση της τεχνολογίας AJAX. Όπως όλες οι τεχνολογίες διαδικτύου, έτσι και η AJAX έχει και αυτή τα δικά της τρωτά σημεία που οι προγραμματιστές πρέπει να αντιμετωπίσουν. Προγραμματιστές που είναι ήδη εξοικειωμένοι με άλλες τεχνολογίες διαδικτύου, ίσως χρειαστεί να μάθουν νέες μεθόδους για βρίσκονται σε θέση να γράψουν ασφαλείς εφαρμογές AJAX²⁰.

2.2.16 Web Hosting – Φιλοξενία ιστοσελίδων

Για να είναι μία ιστοσελίδα διαθέσιμη online, είναι απαραίτητο να φιλοξενηθεί σε έναν κατάλληλο υπολογιστή, δηλαδή έναν web server.

Ο web server, είναι υπεύθυνος για την υποδοχή και επεξεργασία των αιτήσεων των επισκεπτών και της παρουσίασης της ιστοσελίδας που φιλοξενεί. Η διαθεσιμότητα τέτοιων υπηρεσιών φιλοξενίας ιστοσελίδων (web hosting) είναι τόσο στην Ελλάδα, όσο και στο εξωτερικό και συνήθως αναφέρονται ως "πακέτα" κάθε ένα από το οποίο έχει διαφορετικά τεχνικά χαρακτηριστικά, τα οποία συνήθως αναφέρονται πολύ αναλυτικά, ενώ η κάθε εταιρία παρουσιάζει και συγκριτικούς πίνακες μεταξύ των πακέτων που προσφέρει. Κριτήρια για να γίνει η επιλογή της κατάλληλης υπηρεσίας, είναι οι ανάγκες της κάθε ιστοσελίδας αλλά και η τιμή του πακέτου φιλοξενίας. Δύο από τα πιο βασικά χαρακτηριστικά ενός πακέτου φιλοξενίας ιστοσελίδων είναι το Disk Space, δηλαδή ο διαθέσιμος χώρος για τα δεδομένα της ιστοσελίδας και το Bandwidth, δηλαδή ο μέγιστος αριθμός της διακίνησης δεδομένων της ιστοσελίδας²¹.

²⁰ Ajax: A New Approach to Web Applications, 2005, <http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications/> tutor.edu.gr, 2015, <http://www.tutor.edu.gr/index.php/ajax/ajax-intro> Jesse James Garrett (18 February 2005). "Ajax: A New Approach to Web Applications". Ullman, Chris (March 2007). *Beginning Ajax*. wrox. ISBN 978-0-470-10675-4. Archived

²¹ Web hosting service, 2015, http://en.wikipedia.org/wiki/Web_hosting_service

ΚΕΦΑΛΑΙΟ 3

ΣΧΕΔΙΟ ΔΡΑΣΗΣ ΓΙΑ ΤΗΝ ΕΚΠΟΝΗΣΗ ΤΗΣ ΕΡΓΑΣΙΑΣ

Σε αυτό το κεφάλαιο θα γίνει παρουσίαση των τεχνολογιών που χρησιμοποιήθηκαν σαν βάση για να εκπονηθεί η συγκεκριμένη πτυχιακή, δηλαδή την γλώσσα XML και τα Συστήματα Διαχείρισης Περιεχομένου.

3.1 ΠρωτόκολλοXML (eXtensible Markup Language)

Η γλώσσα XML (eXtensible Markup Language) αποτελεί σήμερα ένα ευρέως διαδεδομένο πρότυπο για την περιγραφή και την ανταλλαγή δεδομένων. Η XML έχει αναπτυχθεί και συντηρείται από το W3C (World Wide Web Consortium), και αποτελεί το υποσύνολο ενός προτύπου κατά ISO, της γλώσσας SGML (Standard Generalized Markup Language).

Η γλώσσα XML περιγράφει μια κατηγορία από αντικείμενα δεδομένα, τα οποία λέγονται XML έγγραφα και κατά κάποιον τρόπο τη συμπεριφορά των προγραμμάτων των υπολογιστών που τα επεξεργάζονται. Τα έγγραφα XML αποτελούνται από μονάδες αποθήκευσης, τις οντότητες (entities), που περιλαμβάνουν αναλυμένα λεκτικά (parsed) και μη αναλυμένα λεκτικά (unparsed) δεδομένα. Τα αναλυμένα λεκτικά δεδομένα (parsed) αποτελούνται από χαρακτήρες, κάποιοι από τους οποίους είναι δεδομένα χαρακτήρων και ορισμένοι άλλοι κωδικοποιούν για να περιγράψουν τη διάταξη και τη λογική δομή του εγγράφου, δηλαδή δημιουργούν markup.

Η κύρια διαφορά μεταξύ της HTML και της XML είναι ότι η πρώτη επικεντρώνεται στην εμφάνιση των δεδομένων, ενώ η δεύτερη στοχεύει στην μεταφορά και αποθήκευση των δεδομένων.

Άλλες διαφορές είναι ότι η XML έχει δομημένο κείμενο, ενώ το κείμενο της HTML είναι μορφοποιημένο. Η δομή της XML είναι καθορισμένη από τον χρήστη, ενώ η HTML έχει προκαθορισμένη μορφή. Η XML έχει αποδοτικότερη διασύνδεση υπερκειμένου, αντίθετα με την HTML η οποία έχει περιορισμένη διασύνδεση υπερκειμένου. Τέλος, η XML διαθέτει ένα ευέλικτο και προσαρμοσμένο κείμενο.

Πλέον, η XML αποτελεί το πρότυπο για την αποθήκευση δεδομένων που ανταλλάσσονται μεταξύ των εφαρμογών και οφείλει την επιτυχία της στα παρακάτω χαρακτηριστικά της:

- Η ανεξαρτησία της από υλικό και λογισμικό.
- Διατίθεται δωρεάν.
- Αναπτύσσεται συνεχώς.
- Η υποστήριξη ανεξαρτησίας από τα δεδομένα και ο διαχωρισμός των περιεχομένων από τον τρόπο εμφάνισής και τον χειρισμό τους. Επομένως, καθίσταται ευκολότερη η λεκτική ανάλυσή τους (parsing).
- Η υποστήριξη της από πλήθος εφαρμογών και εργαλείων.
- Η διαθεσιμότητα έτοιμων τρόπων σύνδεσης των κειμένων XML με σύγχρονα προγραμματιστικά περιβάλλοντα, όπως για παράδειγμα το Document Object Model (DOM) και το Simple API for XML (SAX).
- Η επεκτασιμότητα και η ανεξαρτησία της από πλατφόρμες, καθιστώντας την απρόσβλητη από τεχνολογικές αλλαγές.
- Η αναγνωσιμότητα των εγγράφων της από ανθρώπους και από μηχανές. Παρόλο που δεν προορίζονται για ανάγνωση, αν κριθεί απαραίτητο, προσφέρεται η δυνατότητα αυτή στον χρήστη.
- Η συμβατότητα της με Unicode, και επομένως μπορεί να χειριστεί την πληροφορία η οποία είναι γραμμένη σε οποιαδήποτε ανθρώπινη γλώσσα. Ταυτόχρονα, υποστηρίζονται διεθνείς και τοπικές προσαρμογές.
- Η αναπαράσταση, από τους δημιουργούς των εγγράφων, πολύπλοκων δομών δεδομένων, όπως για παράδειγμα λίστες, εγγραφές και δένδρα και η περιγραφή κάθε τύπου δεδομένων, εμπεριέχοντας μαθηματικούς τύπους, οδηγίες διαμόρφωσης λογισμικού, μουσική, αποδείξεις και οικονομικές αναφορές.

Παρόλο που ο αρχικός σχεδιασμός της ήταν για την υποστήριξη αλφαριθμητικών εφαρμογών, όπως για παράδειγμα εφαρμογών βιβλιοθηκών, τραπεζών, κ.α., πλέον έχει υιοθετηθεί ευρέως και σε εφαρμογές που εμπλέκουν μη παραδοσιακά δεδομένα όπως για παράδειγμα χωρικά, πολυμέσων κ.α.

Ενδεικτικά, αναφέρεται ότι η XML είναι η βάση του RDF (Resource Description Framework), του Σημασιολογικού Ιστού (Semantic Web), της WML (Wireless Markup Language), της MathML, της SVG (Scalar Vector Graphics), της KML (Keyhole Markup Language), της RSS (Rich Site Summary), της CML (Chemical Markup Language), της SAOP (Simple Object Access Protocol), της

XHTML (eXtensible HTML), της X3D/VRML (Virtual Reality Markup Language) και της GML (Geography Markup Language).

Τέλος, η γλώσσα XML συνοδεύεται από μια διαρκώς αναπτυσσόμενη οικογένεια εργαλείων που έχει συμβάλει στην ευρεία διάδοση και επιτυχία του προτύπου, όπως για παράδειγμα XMLSchemas, XSL, XSLT, XLink, RDF, OM, XPath, XQuery κ.α., τα οποία καθιστούν ευκολότερη μια σειρά λειτουργιών, όπως η μοντελοποίηση δεδομένων στη γλώσσα XML σύμφωνα με δομημένα σχήματα, η διαχείριση κειμένων XML (εισαγωγή, διαγραφή και ενημέρωση των στοιχείων, ετικετών και γνωρισμάτων ενός κειμένου), η υποβολή ερωτήσεων σε εκφραστικές γλώσσες και η ανάκτηση δεδομένων από μεγάλα κείμενα XML, η ευρετηριοποίηση κειμένων XML, η μορφοποίηση και οπτικοποίηση τους, ο εμπλουτισμός της σημασιολογίας τους κ.α.²².

3.1.1 Γενική περιγραφή εγγράφου XML

Στην γλώσσα XML, τα δεδομένα επισημαίνονται χρησιμοποιώντας ετικέτες (tags), που είναι ονόματα τα οποία περιβάλλονται από τα σύμβολα < >. Αυτές οι ετικέτες χρησιμοποιούνται σε ζεύγη έτσι ώστε να οροθετήσουν τους χαρακτήρες δεδομένων.

Με τις ετικέτες επιτυγχάνεται το markup και μια ετικέτα που αρχίζει με markup, δηλαδή δεδομένα XML, είναι διαφορετική από μια ετικέτα που τερματίζει την επισήμανση κατά την κεκλιμένη / που ακολουθεί τον χαρακτήρα <. Ο προσδιορισμός των ετικετών από τον χρήστη είναι ελεύθερος, και έχει την δυνατότητα να τους δίνει όποια σημασία αυτός ορίζει κάθε φορά.

Η γλώσσα XML στηρίζεται στην αναπαράσταση των δεδομένων μέσω αμιγούς κειμένου. Η βασική μονάδα του συντακτικού της XML είναι το στοιχείο (element). Κάθε στοιχείο (element) μπορεί να περιλαμβάνει ορισμένες ιδιότητες (attributes), που είναι ζεύγη ονόματος- τιμής στις ετικέτες αρχής. Το στοιχείο είναι ένα τμήμα κειμένου το οποίο περιβάλλεται από ταιριαστές ετικέτες (tags).

Ένα παράδειγμα ενός στοιχείου (element) είναι το παρακάτω: <συγγραφέας>Γόλκιν</συγγραφέας>. Αυτό το στοιχείο φιλοξενεί το επίθετο ενός φυσικού προσώπου. Η ετικέτα (tag) του στοιχείου είναι το <συγγραφέας>, η οποία

²² XML Language Specification. <http://www.w3.org/XML/>
XML tutorial, <http://www.w3schools.com/xml/default.asp>
Σαμψών, Δ. (2003). *Η γλώσσα σήμανσης XML*. Πανεπιστήμιο Πειραιώς.

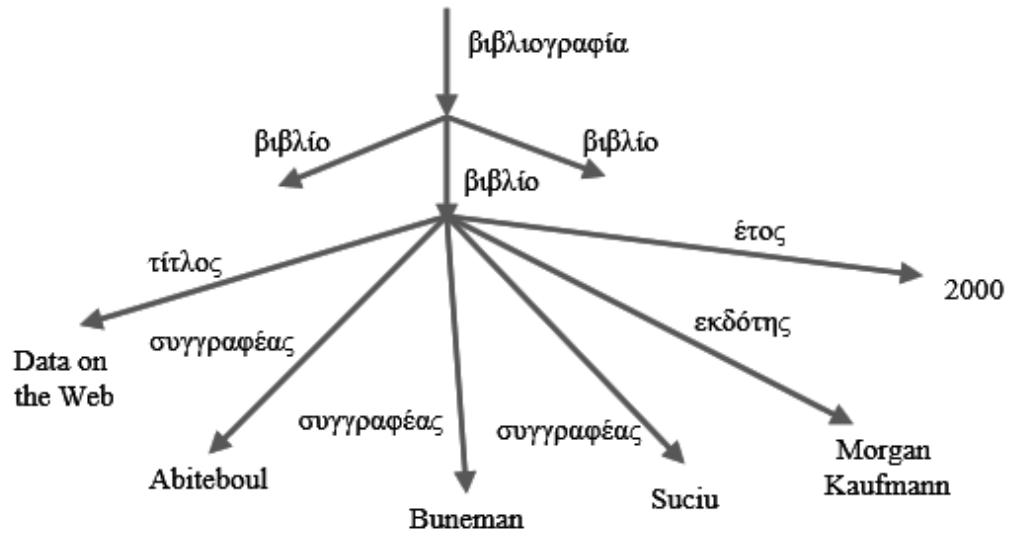
ταιριάζει και κλείνει με το </συγγραφέας>. Μέσα από το όνομα της ετικέτας, μεταφέρεται η πληροφορία- σημασιολογία ότι το φυσικό πρόσωπο είναι ο συγγραφέας. Τα στοιχεία σε ένα κείμενο XML έχουν την δυνατότητα να είναι φωλιασμένα. Επίσης, μπορούν να είναι κενά, οπότε και υπάρχει η δυνατότητα σύμπτυξης τους.

Παράδειγμα ενός κενού στοιχείου αποτελεί το παρακάτω: <συγγραφέας></συγγραφέας> και η συμπτυγμένη μορφή του είναι η εξής: <συγγραφέας></>ή<συγγραφέας/>.

Κάθε κείμενο XML έχει ένα στοιχείο, μέσα στο οποίο φωλιάζουν όλα τα υπόλοιπα στοιχεία, είναι μοναδικό και ονομάζεται ρίζα του κειμένου. Το στοιχείο ρίζας, είναι το πρώτο μετά τη δήλωση XML. Τα στοιχεία τοποθετούνται το ένα μέσα στο άλλο για να σχηματιστούν ιεραρχίες, με το στοιχείο ρίζας στην κορυφή αυτής της ιεραρχίας. Με αυτήν την ιδιότητα της ενθυλάκωσης (nesting) μπορούν να κατασκευαστούν στοιχεία με πολύπλοκη εσωτερική δομή. Ένα παράδειγμα κειμένου XML, το οποίο περιγράφει μια συλλογή βιβλίων, όπως για παράδειγμα τις βιβλιογραφικές αναφορές ενός άρθρου είναι το παρακάτω:

```
<βιβλιογραφία >
<βιβλίο >
<τίτλος > Αρχοντας των δαχτυλιδιων </ τίτλος >
<συγγραφέας > Τόλκιν </ συγγραφέας >
<εκδότης >Κέδρος</ εκδότης >
<έτος >2002</ έτος >
</ βιβλίο >
<βιβλίο >
<τίτλος>Data on the Web</τίτλος>
<συγγραφέας>Abiteboul</ συγγραφέας >
<συγγραφέας >Buneman</ συγγραφέας >
<συγγραφέας >Suciu</ συγγραφέας >
<εκδότης >Morgan Kaufmann</ εκδότης >
<έτος >2000</ έτος >
</ βιβλίο >
</ βιβλιογραφία >
```

Η αναπαράσταση ενός κειμένου XML μπορεί να γίνει και σε μορφή διαγράμματος . Αυτό το διάγραμμα έχει στην απλή του μορφή μια ιεραρχική δομή-δομή δένδρου και ονομάζεται διάγραμμα XML.



Εικόνα 16 : Διάγραμμα XML

Όπως έχει ήδη αναφερθεί, η γλώσσα XML επιτρέπει την ανάθεση γνωρισμάτων (attributes) στα στοιχεία. Ένα τέτοιο παράδειγμα αποτελεί το παρακάτω:

```
<βιβλίο κόστος =“40” νόμισμα =“ Ευρώ ”>
<τίτλος>DataontheWeb</ τίτλος>
<συγγραφέας >Abiteboul</ συγγραφέας >
...
</ βιβλίο >
```

Με βάση αυτό, το στοιχείο βιβλίο ανατίθεται δύο γνωρίσματα: το γνώρισμα κόστος με τιμή 40 και το γνώρισμα νόμισμα με τιμή Ευρώ. Αντιθέτως από τα στοιχεία, οι τιμές των γνωρισμάτων αναγράφονται πάντα εντός διπλών εισαγωγικών. Τα γνωρίσματα είναι ένας εναλλακτικός τρόπος αναπαράστασης δεδομένων σε ένα κείμενο XML.

Επίσης, με την XML επιτρέπεται η ανάθεση μοναδικών ταυτοτήτων στα στοιχεία ενός κειμένου, η οποία πραγματοποιείται σε ένα συγκεκριμένο γνώρισμα του στοιχείου, το οποίο ονομάζεται id. Αυτό το γνώρισμα αξιοποιείται σαν αναφορά του στοιχείου από άλλα στοιχεία του κειμένου. Η αναφορά επιτυγχάνεται μέσω του γνωρισματος idref. Για την καλύτερη κατανόηση, ακολουθεί ένα παράδειγμα:

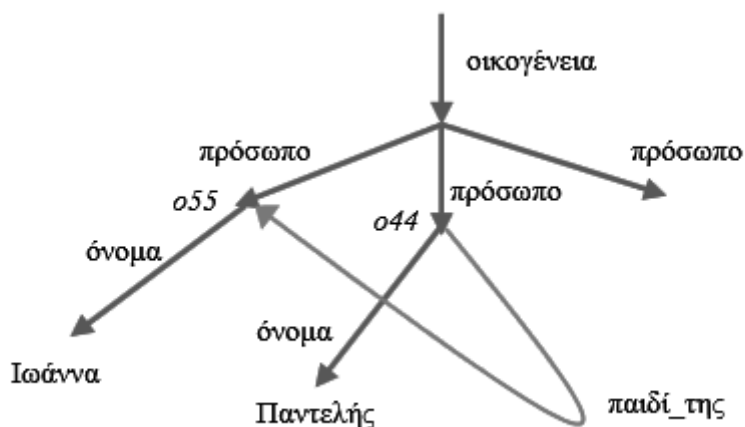
```
<οικογένεια>
<πρόσωπο id=“ο55”>
<όνομα>Ιωάννα </όνομα>
</πρόσωπο>
```

```

<πρόσωπο id="ο44">
<όνομα>Παντελής</όνομα>
<παιδί_της idref="ο55"/>
</πρόσωπο >
...
</ οικογένεια >

```

Σε αυτό το παράδειγμα, παρουσιάζονται δύο πρόσωπα μιας οικογένειας, η Ιωάννα και ο Παντελής, με ταυτότητες (id), ο55 και ο44 αντιστοίχως. Στο γνώρισμα idref στο στοιχείο παιδί_της συσχετίζονται τα δύο πρόσωπα. Η σημασιολογία, η οποία μεταφέρεται από την ετικέτα αυτού του στοιχείου, απεικονίζει ότι ο Παντελής είναι παιδί της Ιωάννας. Η χρήση ταυτοτήτων και αναφορών αποτελούν τον μηχανισμό της γλώσσας XML για την επέκταση των ιεραρχικών δομών-δένδρων σε γράφους, δηλαδή σε δικτυακές δομές.



Εικόνα 17: Διάγραμμα XML με αναφορές

```

<?XML version="1.0" ?>
<!--
A first program using XML
-->
<holidays XMLns:altgr="http://www.alternativegreece.gr"
XMLns:ginn="http://www.guestinn.com">
  <cruises>
    <voyage type="on a large ship" price="normal"
destination="scheduled"/>
    <voyage type="on a private yacht" price="high"
destination="anywhere"/>
  </cruises>
  <altgr:alternative_tourism>
    <altgr type="rafting" />
    <altgr type="ski" />
    <participation price="20-25 euro">There are special prices for

```



```

students</participation>
    <ages type="young" />
</altgr:alternative_tourism>
<ginn:alternative_tourism>
    <ginn type="residence in traditional villages" />
    <participation price="50-60 euro" />
    <ages type="all ages" />
</ginn:alternative_tourism>
</holidays>

```

Το έγγραφο XML το οποίο παρουσιάζεται στο παραπάνω παράδειγμα ξεκινάει με μια προαιρετική δήλωση XML, η οποία το καθορίζει ως έγγραφο XML. Η παράμετρος πληροφοριών (version) προσδιορίζει την έκδοση της XML, η οποία χρησιμοποιείται στο έγγραφο. Τα σχόλια XML αρχίζουν με <!-- και τελειώνουν με --> και έχουν την δυνατότητα τοποθέτησης σχεδόν οπουδήποτε σε ένα έγγραφο XML.

Γίνεται κατανοητό ότι με την ιδιότητα της ενθυλάκωσης υπάρχει η δυνατότητα κατασκευής στοιχείων με πολύπλοκη δομή, όπως για παράδειγμα το στοιχείο cruises.

Με την χρήση των χώρων ονομάτων (namespaces) της XML, παρέχεται ένα μέσο μοναδικού προσδιορισμού των στοιχείων XML και αποφεύγονται οι διενέξεις ονομάτων στην περίπτωση ύπαρξης ενός στοιχείου σε δύο διαφορετικά κείμενα με το ίδιο όνομα και αυτά τα κείμενα θα συγχωνευτούν. Τα στοιχεία διαφοροποιούνται μέσω των προθεμάτων χώρων ονομάτων, τα οποία προσδιορίζουν τον χώρο ονομάτων όπου ανήκει ένα στοιχείο και μιας σύμβασης, που συνήθως είναι ένας καθολικός δείκτης πόρων (Universal Resource Locator – URL) και προσδιορίζει που είναι ορισμένος ο χώρος ονομάτων. Τα URLs δεν είναι απαραίτητο να αναφέρονται σε πραγματικές ιστοσελίδες ή να είναι καταλλήλως διαμορφωμένα. Απλά αναπαριστούν μια ακολουθία χαρακτήρων για να διαφοροποιήσουν τα ονόματα. Για παράδειγμα, το <altgr:alternative_tourism> προσδιορίζει το στοιχείο alternative_tourism με το πρόθεμα χώρου ονομάτων altgr και URL το <http://www.alternativegreece.gr>²³.

²³ World Wide Web Consortium, 2015, <http://www.w3.org/>

Sall, K.B., 2002. XML Family of Specifications. A Practical Guide. Addison Wesley.

H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML Howto program, Prentice Hall, 2001

3.1.2 Ορθότητα XML εγγράφου

Για να θεωρηθεί ένα έγγραφο XML ορθό θα πρέπει να είναι ταυτόχρονα

- Καλοσχηματισμένο (well-formed), δηλαδή να υπακούει σε όλους τους συντακτικούς κανόνες της XML. Αν ένα έγγραφο δεν είναι ορθό συντακτικά, τότε δεν θεωρείται έγγραφο XML, η ανάλυση (parsing) παύει και ο αναλυτής (parser) δίνει ένα σφάλμα.
- Έγκυρο (valid), δηλαδή να ικανοποιεί ένα σύνολο κανόνων ορισμένοι από τον χρήστη. Αυτό επιτυγχάνεται με τη χρήση ενός βοηθητικού κειμένου, το οποίο προσδιορίζει τις έγκυρες ετικέτες, το φώλιασμα των στοιχείων και τα γνωρίσματα αυτών κ.α. και δομείται με βάση τις προδιαγραφές που ορίζουν τα δύο παρακάτω πρότυπα: το Document Type Definition (DTD) και το XML Schema Definition (XSD)²⁴.

3.1.2.1 DTD (Document Type Definition)

Η γλώσσα XML κάνει χρήση ενός μοντέλου περιγραφής των εγγράφων XML, το οποίο ονομάζεται Document Type Definition (DTD). Η XML το χρησιμοποιεί για την περιγραφή των δεδομένων.

Το DTD είναι ένας μηχανισμός ο οποίος κληρονομήθηκε από την γλώσσα SGML και χρησιμοποιείται για την περιγραφή οποιουδήποτε αντικειμένου σε ένα έγγραφο XML, όπως είναι τα στοιχεία και τα χαρακτηριστικά. Είναι ένα σύνολο από κανόνες που αφορούν τα tags της XML. Ειδικότερα, ορίζει κανόνες για το ποια είναι τα επιτρεπτά ονόματα στοιχείων και τα επιτρεπτά υποστοιχεία (subelements) και χαρακτηριστικά (attributes) για κάθε συγκεκριμένο στοιχείο καθώς και το φώλιασμά τους. Παράδειγμα ενός εγγράφου DTD είναι το παρακάτω. Στα έγγραφα με μορφοποίηση XML τα σχόλια περιέχονται μεταξύ των <!-- και -->.

```
<!Elements name (#PCDATA | fname | lname )*>
<!Elementsfname (#PCDATA )><!--Η δεσμευμένη λέξη PCDATA
σημαίνει parsed character data και σημαίνει ότι τα συγκεκριμένο στοιχείο μπορεί
να περιέχει κείμενο. Χρησιμοποιείται συνήθως για τα στοιχεία «φύλλα».-- >
<!Elements lname (#PCDATA )>
<!Elements address (street, region; , country)>
<!ATTLIST address preferred (true | false) 'false'>
<!Elements street (#PCDATA)><!--Το στοιχείο street είναι ένα στοιχείο
```

²⁴ XML Language Specification. <http://www.w3.org/XML/>
XML tutorial, <http://www.w3schools.com/xml/default.asp>
Σαμψών, Δ. (2003). *Η γλώσσα σήμανσης XML*. Πανεπιστήμιο Πειραιώς.

```

<<φύλλο>>διότι δεν έχει παιδιά -->
  <!Elements region (#PCDATA)>
  <!Elements country (#PCDATA)>
  <!Elements tel (#PCDATA)>
  <!ATTLIST tel preferred (true | false) 'false'>
  <!Elements email (#PCDATA)>
  <!ATTLIST email href CDATA #REQUIRED
  preferred (true | false) 'false'>

```

Το παραπάνω DTD έγγραφο μπορεί να συνοδεύει το εξής έγγραφο XML:

```

<?XML version="1.0"?>
<!DOCTYPE address-book SYSTEM "address-book. dtd">
< address-book >
  <entry>
    <name>Γιάννης Παπαδόπουλος</name>
    <address>
      <street>Ικάρου 82</street>
      <region>Ηράκλειο</region>
      <country>Ελλάδα</country>
    </address>
    <tel preferred="true">210458796547</tel>
    <email href=whatever@wherever.com/>
  </entry>
</address-book>

```

Όσον αφορά τη δομή και τη σύνταξη των εγγράφων DTD:

- Όλα τα ονόματα των αντικειμένων αρχίζουν με γράμμα και απαγορεύεται ο χαρακτήρας κενού διαστήματος.
- Η XML υποστηρίζει πλήρως το πρότυπο κωδικοποίησης χαρακτήρων Unicode, επομένως επιτρέπεται η χρήση ελληνικών χαρακτήρων ως έγκυροι χαρακτήρες ονομάτων για τα στοιχεία του εγγράφου.
- Οι χαρακτήρες *, ?, και + λέγονται χαρακτήρες ύπαρξης και δείχνουν αν και πώς τα στοιχεία στην λίστα των παιδιών επαναλαμβάνονται. Αν δεν υπάρχει δείκτης ύπαρξης σε ένα στοιχείο, αυτό δείχνει ότι το στοιχείο αυτό εμφανίζεται μόνο μια φορά. Στοιχείο με τον χαρακτήρα + πρέπει να εμφανίζεται μια τουλάχιστον φορά. Στοιχείο με τον χαρακτήρα * πρέπει να εμφανίζεται καμία ή περισσότερες φορές. Στοιχείο με τον χαρακτήρα ? δείχνει ότι μπορεί να εμφανιστεί το πολύ μια φορά.
- Οι χαρακτήρες , και | ονομάζονται συζευκτικές, αφού συμβάλουν στον διαχωρισμό των παιδιών ενός στοιχείου μέσα στο μοντέλο περιεχομένου. Ο χαρακτήρας , σημαίνει ότι τα δύο στοιχεία αριστερά και δεξιά του κόμματος

πρέπει να εμφανίζονται με την ίδια σειρά μέσα στο έγγραφο. Ο χαρακτήρας | σημαίνει ότι μόνο το ένα από τα στοιχεία πρέπει να εμφανίζεται στο έγγραφο.

Κάθε τύπος έγκυρου εγγράφου XML που είναι διαφορετικό από τα υπόλοιπα χρησιμοποιεί το δικό του DTD. Για να είναι εφικτή η επαλήθευση της εγκυρότητας ενός XML για συμμόρφωση με τους κανόνες που ορίζει το συνοδευτικό DTD. Με τα ίδια προγράμματα πρέπει να ελέγχονται και τα ίδια τα έγγραφα DTD για συνέπεια με τους κανόνες που ορίζονται στο πρότυπο DTD.

Τα προγράμματα validators χωρίζονται σε αυτά που λειτουργούν με γραμμή εντολών, όπως για παράδειγμα το XSV, αυτά που διαθέτουν API όπως είναι τα Apache Xerces, IBM Schema Quality Checker και Microsoft MSXML4.0 και τέλος με γραφικό περιβάλλον όπως το XML Spy και το TurboXML²⁵.

3.1.2.2 XSD (XML Schemas Definitions)

Τα κείμενα DTD παρουσιάζουν αρκετές αδυναμίες όσον αφορά την προδιαγραφή κειμένων XML και για αυτό αντικαταστάθηκαν από τα κείμενα XSD. Οι βασικές αδυναμίες των κειμένων DTD αφορούν το γεγονός ότι δεν είναι γραμμένα σε γλώσσα XML και δεν υποστηρίζουν τύπους δεδομένων δηλαδή, δεν προδιαγράφουν τους τύπους των τιμών που φιλοξενεί κάθε στοιχείο ή γνώρισμα.

Τα κείμενα XSD προδιαγράφουν πληρέστερα τα κείμενα XML, αφού υποστηρίζουν τύπους δεδομένων, επιτρέπουν τη δημιουργία νέων τύπων δεδομένων από τους χρήστες και υιοθετούν αντικειμενοστραφείς έννοιες, όπως την κληρονομικότητα και τον πολυμορφισμό. Τα XML Schemas αποτελούν σύσταση του οργανισμού W3C και στην ουσία είναι έγγραφα XML τα οποία προσδιορίζουν τον τρόπο με τον οποίο πρέπει να δομούνται κάποια άλλα έγγραφα XML.

Το πλεονέκτημά τους σε σχέση με τα DTDs (Document Type Definitions) είναι ότι υπάρχει η δυνατότητα να υποστούν χειρισμούς, όπως για παράδειγμα

²⁵ Σαμψών, Δ. (2003). *Ηγλώσσασήμεανσης XML*. Πανεπιστήμιο Πειραιώς.

Damien Foggon, Daniel Maharry, Chris Ullman and Karli Watson. *Programming Microsoft .NET XML Web Services*, Microsoft Press, 2004.

Mathew Macdonald. *Microsoft .NET Distributed Applications: Integrating XML Web Services and .NET Remoting*, 2001

David Carlson. *Modeling XML applications with UML*, Addison-Wesley, 2000

H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. *XML How to program*, Prentice Hall, 2001

μπορούν να προστεθούν ή να διαγραφούν στοιχεία, όπως σε οποιοδήποτε άλλο έγγραφο XML.

Τα XML Schemas εγγυώνται ότι ο ορθός τύπος δεδομένων αποθηκεύεται σε κάθε στοιχείο, όπως για παράδειγμα το περιεχόμενο του στοιχείου Date_of_birth θα είναι τύπου date και όχι ακέραιος ή κάποια ακολουθία χαρακτήρων (string). Επίσης, παρέχεται η δυνατότητα στον χρήστη να ορίσει δικούς του τύπους δεδομένων, εκτός από τους ήδη υπάρχοντες στο σύστημα τύπων XML Schema Definition (XSD). Κάνουν ευκολότερη τη μετάβαση ανάμεσα στις πλατφόρμες, αφού το σύστημα τύπων XSD είναι ανεξάρτητο από πλατφόρμες και επομένως δεν δημιουργείται πρόβλημα αν ένας .NET εξυπηρετητής στείλει μια τιμή ακεραίου (integer) μήκους 32 bit σε έναν Visual Basic 6.0 πελάτη, για τον οποίο ο τύπος ακέραιος (integer) έχει μήκος 16 bit.

Παρακάτω παρατίθεται ένα έγγραφο XML με το αντίστοιχο XML Schema στο οποίο υπακούει στο σχήμα.

ΈγγραφοXML book.XML

```
<?XML version="1.0" ?>
<!-- αρχείο XML που επισημαίνει δεδομένα βιβλίου-->
<book>
  <title>C# How to program</title>
  <serial_no>123456789</serial_no>
</book>
```

XML Schema bookSchema.xsd

```
<?XML version="1.0" encoding="utf-8" ?>
<xs:schema XMLNs:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="book">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="title" type="xs:string"/>
      <xs:element name="serial_no" type="xs:integer"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```

Όπως ήδη έχει αναφερθεί, τα DTD είναι ένα χαρακτηριστικό της XML που κληρονομήθηκε από τον πρόγονο της γλώσσας, την SMGL. Για αυτόν τον λόγο, τα DTD δεν υποστηρίζουν ενδογενώς τους χώρους ονοματοδοσίας. Επομένως, το W3C εξετάζει την αντικατάσταση των DTD με τη νέα σύσταση των XML Schemas. Ένα XML Schema είναι ένα XML έγγραφο με επέκταση αρχείου .xsd το οποίο εκφράζει

τους κανόνες που πρέπει να ακολουθεί ένα XML έγγραφο. Ανάλογα με την λειτουργία των DTD, κάθε καλά έγκυρο έγγραφο XML μπορεί να συνοδεύεται εναλλακτικά από ένα XML Schema.

Εξαιτίας των πολλών πλεονεκτημάτων του XML Schema έναντι του DTD, είναι χρήσιμη η μετατροπή των διαθέσιμων .dtd αρχεία .xsd. Τα κυριότερα πλεονεκτήματα των XML Schemas έναντι των DTD είναι τα παρακάτω:

- Η χρησιμοποιούμενη γλώσσα για την σύνταξη των XSD αρχείων ακολουθεί σύνταξη ανάλογη με την σύνταξη των XML εγγράφων. Αντιθέτως, τα έγγραφα DTD χρησιμοποιούν σύνταξη ασύμβατη με των XML εγγράφων.
- Τα XML Schemas υποστηρίζουν τους χώρους ονοματοδοσίας (namespaces). Ένα namespace ορίζει το σύνολο στο οποίο ανήκουν τα ονόματα των στοιχείων σε ένα έγγραφο XML.
- Υποστηρίζουν ενδογενώς περισσότερους τύπους δεδομένων (τουλάχιστον 44) από ότι τα DTD (10 τύπους δεδομένων). Επίσης, τα XML Schemas επιτρέπουν τον ορισμό νέων σύνθετων τύπων, αυξάνοντας τις δυνατότητες περιγραφής δεδομένων και κανόνων που τα διέπουν.
- Υποστηρίζουν κανονικές εκφράσεις (regular expressions), οι οποίες κατά τον ορισμό νέων σύνθετων τύπων είναι πολύ χρήσιμες.
- Προσφέρονται δυνατότητες συμπύκνωσης των tags (inlinedelement declarations), μειώνοντας της πολυπλοκότητας και καθιστώντας ευκολότερη σάρωση και ανάγνωση του εγγράφου κανόνων .xsd.
- Υπάρχει η δυνατότητα ορισμού στοιχείων με μηδενικό (null) περιεχόμενο ή ακόμα και δύο ή περισσότερων στοιχείων που έχουν το ίδιο όνομα αλλά διαφορετικό περιεχόμενο.
- Μπορούν να χρησιμοποιηθούν σε ένα έγγραφο περισσότερα του ενός Schemas, με την προϋπόθεση να ανήκουν όλα στο ίδιο namespace.
- Ένα Schema με το στοιχείο <include> μπορεί να χρησιμοποιεί στοιχεία τα οποία είναι δηλωμένα σε ένα άλλο Schema και με το στοιχείο <redefine> μπορεί να επανακαθορίζει τα στοιχεία αυτά.
- Η επεκτασιμότητα ενός Schema μεγαλώνει αν προστεθούν στο αρχείο.xsd οι τεχνολογίες XSLT/Xpath.
- Επιτρέπεται στα XML έγγραφα η χρησιμοποίηση πρόσθετων στοιχείων εκτός από αυτά που ορίζει το XML Schema, σύμφωνα με τις ανάγκες του

δημιουργού περιεχομένου. Σε αυτή την περίπτωση, δηλώνονται τα στοιχεία <any> και <anyAttribute> στο .xsd αρχείο²⁶.

3.1.3 XML Parsers

Αν και τα έγγραφα XML είναι αρχεία κειμένου, τότε η ανάκτηση δεδομένων από αυτά μέσα από τεχνικές σειριακής προσπέλασης αρχείων δεν είναι πρακτική ούτε αποτελεσματική, ειδικά όταν τα δεδομένα πρέπει να προστεθούν ή να αφαιρεθούν δυναμικά. Αυτό το κενό καλύπτει η διαδικασία της λεκτικής ανάλυσης (parsing), η οποία αναλύει το κείμενο στα επιμέρους στοιχεία του, στα οποία περιλαμβάνονται η αρχή ετικέτας, το τέλος ετικέτας, το κείμενο και οι ιδιότητες.

Στην διαδικασία της λεκτικής ανάλυσης εμπλέκεται πάντα ένας λεκτικός αναλυτής (parser), ο οποίος παίζει κεντρικό ρόλο στην αντιστοίχιση και αλληλοσύνδεση της δεντρικής δομής του κειμένου XML με την αντίστοιχη- συνήθως αντικειμενοστραφή δομή στην χρησιμοποιούμενη γλώσσα, δεδομένου ότι το κείμενο XML είναι ένας συρμός χαρακτήρων που ανταλλάσσεται μέσω μίας επικοινωνιακής σύνδεσης, ενός αρχείου στον δίσκο κ.α. Οι πιο γνωστοί λεκτικοί αναλυτές είναι ο DOM (Document Object Model) και ο SAX (Simple API for XML).

Το Μοντέλο Αντικειμένου Εγγράφου (Document Object Model- DOM) αποθηκεύει τα δεδομένα εγγράφου ως δομές δένδρων στη μνήμη αυτόματα, χωρίς την παρέμβαση του χρήστη στην εξέλιξη της ανάλυσης. Βοηθά τον χρήστη στην θεώρηση της δεντρικής μορφή του XML κειμένου με αντικειμενοστραφή τρόπο, αφού το δένδρο DOM αναπαριστά κάθε στοιχείο του εγγράφου XML ως ένα κόμβο στο δένδρο και να επέμβει σε αυτό με συγκεκριμένες διεπαφές. Επομένως, με το XML DOM δίνεται στον χρήστη η ευκαιρία δημιουργίας ενός XML κειμένου, πλοήγησης σε αυτό και πρόσθεσης, αφαίρεσης και μεταβολής στοιχείων του.

Το Simple API for XML (SAX) μοιάζει με σειριακό αναγνώστη (SAX reader) του κειμένου XML, χαρακτηρίρα προς χαρακτηρίρα, ο οποίος αναγνωρίζοντας τα διάφορα μέρη του αρχείου, δημιουργεί αντίστοιχα συμβάντα (events), στα οποία το πρόγραμμα του χρήστη πρέπει να ανταποκριθεί με βάση τις επιθυμίες του. Μόλις ο

²⁶ Damien Foggon, Daniel Maharry, Chris Ullman and Karli Watson. Programming Microsoft .NET XML Web Services, Microsoft Press, 2004.

Mathew Macdonald. Microsoft .NET Distributed Applications: Integrating XML Web Services and .NET Remoting, 2001

David Carlson. Modeling XML applications with UML, Addison-Wesley, 2000
<http://www.w3.org/tr/xmlschema-0>

χειριστής συμβάντων (event handler), που έχει δημιουργήσει ο χρήστης, αντιμετωπίσει το συμβάν, ο λεκτικός αναλυτής συνεχίζει μέχρι το επόμενο σημείο.

Επομένως, με τον λεκτικό αναλυτή SAX, ο χρήστης βρίσκεται στο κατώτερο προγραμματιστικά περιβάλλον που του δίνει πλήρη πρόσβαση στο έγγραφο XML. Η ένταξη των επιθυμιών του μέσω μιας εφαρμογής είναι πιο επίπονη σε σχέση με το DOM, αλλά οι διαθέσιμες δυνατότητες και η ταχύτητα υπερτερούν. Ο χρήστης δεν περιμένει να δημιουργηθεί και να του προσφερθεί η πλήρης δομή του XML σύμφωνα με το DOM μοντέλο, αλλά ειδοποιείται και μπορεί να ανταποκριθεί στα σχετικά συμβάντα on the fly, καθώς διαβάζεται το XML κείμενο και χωρίς να περιμένει το τέλος της ανάγνωσης αυτής²⁷.

3.1.4 Διαχείριση και μορφοποίηση κειμένων XML

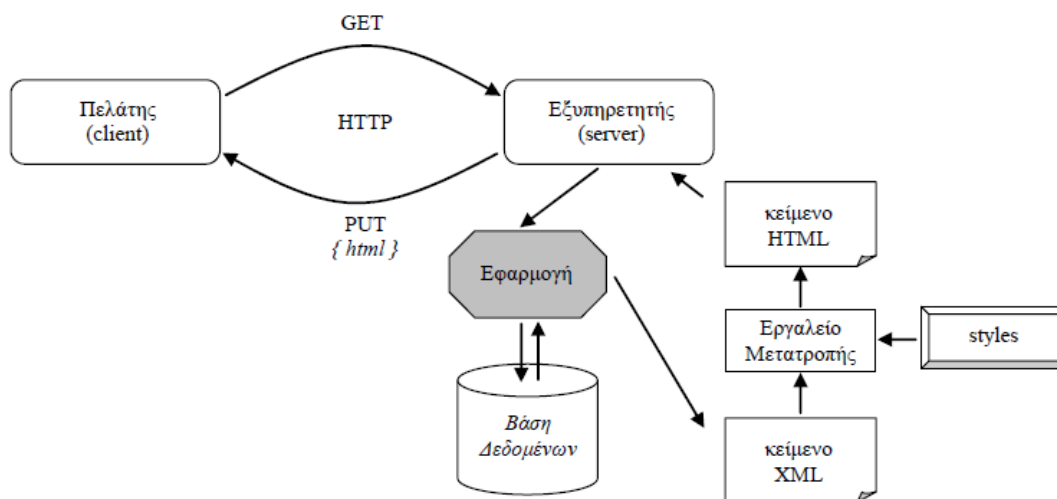
Η διαχείριση των δεδομένων ενός κειμένου XML πραγματοποιείται με τη χρήση μιας γλώσσας ερωτήσεων. Στο παρελθόν έχουν προταθεί πολλές γλώσσες διαχείρισης κειμένων XML όπως για παράδειγμα: XPath, XQuery, Lorel, UnQL, XML-QL, XQL κ.α., κάθε μια από τις οποίες έχει τις ιδιαιτερότητές της και υποστηρίζει καλύτερα ή χειρότερα ορισμένες από τις διαδικασίες διαχείρισης.

Για το στυλ και την μορφοποίηση των εγγράφων XML, δηλαδή τον τρόπο παρουσίασης του εγγράφου μέσα από μια εφαρμογή, όπως για παράδειγμα ένας browser ή ένας επεξεργαστής κειμένου, υπάρχουν τεχνολογίες που καθιστούν τα έγγραφα XML ευκολότερα παραμετροποιήσιμα όσον αφορά την εμφάνισή τους.

Ένας τρόπος ελέγχου της μορφοποίησης των XML εγγράφων είναι η ενσωμάτωση στο έγγραφο XML κάποια HTML tags τα οποία συμβάλουν στην μορφοποίηση στοχεύοντας στην επιθυμητή παρουσίαση του εγγράφου μέσα από έναν web browser. Αν μπορούσαμε να έχουμε μια αυστηρή έκδοση της HTML που υπακούει σε όλους τους κανόνες δομής της XML, τότε είναι εφικτή η δημιουργία της μορφοποίησης που θέλουμε, διατηρώντας την απαραίτητη αυστηρή δομή της XML.

²⁷ Damien Foggon, Daniel Maharry, Chris Ullman and Karli Watson. Programming Microsoft .NET XML Web Services, Microsoft Press, 2004.
Mathew Macdonald. Microsoft .NET Distributed Applications: Integrating XML Web Services and .NET Remoting, 2001
David Carlson. Modeling XML applications with UML, Addison-Wesley, 2000
H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML How to program, Prentice Hall, 2001

Μια τέτοια προσπάθεια εκφράζεται από την εξελιγμένη γλώσσα σήμανσης, την XHTML²⁸.

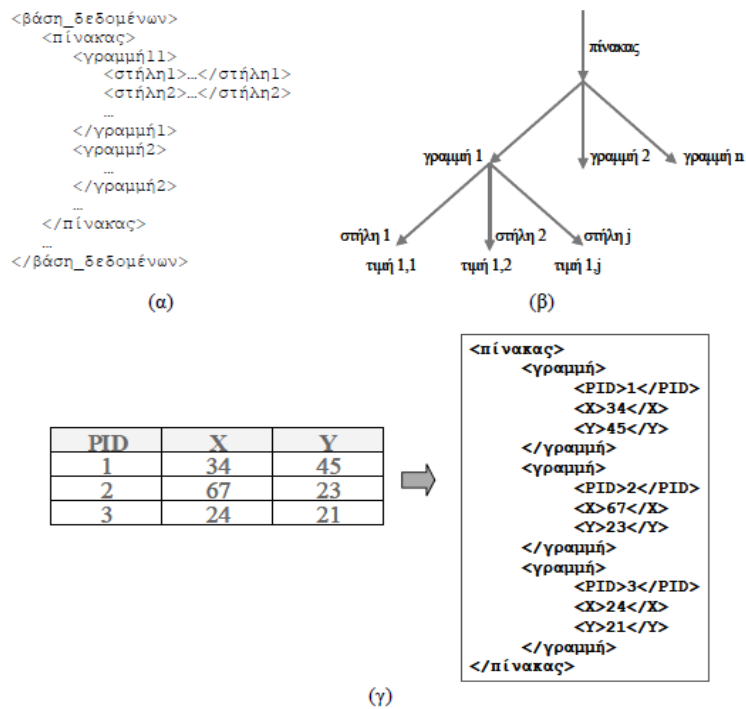


Εικόνα 18: Παράδειγμα αιτήματος πελάτη σε εξυπηρετητή, που εμπλέκει το μετασχηματισμό κειμένου XML σε κείμενο HTML.

3.1.4.1 XPath(XMLPathLanguage)

Η γλώσσα XPath (XML Path Language) εφαρμόζει μια σύνταξη που προσομοιώνει ιεραρχικά μονοπάτια. Η γλώσσα αυτή αξιοποιεί μια βιβλιοθήκη συναρτήσεων, με τις οποίες επιτυγχάνει την ανάκτηση δεδομένων από κείμενα XML και την περαιτέρω επεξεργασία τους. Να σημειωθεί, ότι η βιβλιοθήκη των συναρτήσεων είναι επεκτάσιμη.

²⁸ Σαμψών, Δ. (2003). *Η γλώσσα σήμανσης XML*. Λ. Πανεπιστήμιο Πειραιώς.



Εικόνα 19: Απεικόνιση μιας σχέσης (πίνακα) σε κείμενο XML: Η δομή του κειμένου XML (α), το αντίστοιχο διάγραμμα XML (β), και ένα παράδειγμα απεικόνισης (γ).

Η XPath διαχειρίζεται ένα μόνο κείμενο XML, το οποίο και αντιμετωπίζει σαν ένα δένδρο με κόμβους. Υπάρχουν διάφορα είδη κόμβων στο μοντέλο της γλώσσας, όπως οι κόμβοι κειμένου, στοιχείων, γνωρισμάτων, κ.ά.. Οι τιμές που επιστρέφονται μετά την εκτέλεση μιας ερώτησης αντιμετωπίζονται επίσης σαν κόμβοι.

Θα χρησιμοποιήσουμε το ακόλουθο έγγραφο XML στα κάτωθι παραδείγματα:

```

<?XML version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>

  <book category="CHILDREN">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>

  <book category="WEB">
    <title lang="en">XQuery Kick Start</title>
    <author>James McGovern</author>
    <author>Per Bothner</author>

```

```

    <author>Kurt Cagle</author>
    <author>James Linn</author>
    <author>Vaidyanathan Nagarajan</author>
    <year>2003</year>
    <price>49.99</price>
  </book>

  <book category="WEB">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>

```

- **Επέλεξε όλους τους τίτλους:** Το ακόλουθο παράδειγμα επιλέγει όλους τους κόμβους με το στοιχείο «τίτλος» **/bookstore/book/title**
- **Επέλεξε τον τίτλο του πρώτου βιβλίου:** Το ακόλουθο παράδειγμα επιλέγει τον πρώτο κόμβο που θα βρει με το στοιχείο «τίτλος» **/bookstore/book[1]/title**
- **Επέλεξε όλες τις τιμές:** Το ακόλουθο παράδειγμα επιλέγει το κείμενο από όλους τους κόμβους με το στοιχείο «τιμές» **/bookstore/book/price[text()]**
- **Επέλεξε όλες τις τιμές πάνω από 35:** Το ακόλουθο παράδειγμα επιλέγει όλους τους κόμβους με το στοιχείο «τιμές» που είναι πάνω από την τιμή 35 **/bookstore/book[price>35]/price**
- **Επέλεξε όλους τους τίτλους που οι τιμές είναι πάνω από 35:** Το ακόλουθο παράδειγμα επιλέγει όλους τους κόμβους με το στοιχείο «τιμές» που είναι πάνω από την τιμή 35 αλλά θα εμφανίσει τους τίτλους τους **/bookstore/book[price>35]/title**²⁹.

3.1.4.2 XSL (XML Style Language)

Η γλώσσα XSL (XML Style Language) σχεδιάστηκε για τον μετασχηματισμό ενός κειμένου XML σε κείμενο HTML. Με άλλα λόγια, αποτελεί μια γλώσσα μορφοποίησης ενός κειμένου XML για την παρουσίασή του σε μια ιστοσελίδα του Παγκόσμιου Ιστού.

Η XSL εφαρμόζεται επίσης για τη μετατροπή κειμένων XML σε άλλα κείμενα XML. Το μοντέλο της XSL ακολουθεί το ιεραρχικό. Μια ερώτηση XSL συντίθεται

²⁹ Sall, K.B., 2002. XML Family of Specifications. A Practical Guide. Addison Wesley. W3schools, 2015, http://www.w3schools.com/xpath/xpath_examples.asp

από κανόνες προτύπων (template rules) και έχουν τη μορφή: κανόνας πρότυπο = συνθήκη + πρότυπο.

Η εκτέλεση της ερώτησης έχει αναδρομική ισχύ. Συγκεκριμένα, ακολουθεί τα παρακάτω βήματα:

- a. Ξεκινάει από το στοιχείο (κόμβο) ρίζα
- b. Ελέγχει μια συνθήκη (pattern) στον κόμβο αυτό
- c. Αν αυτή ικανοποιηθεί, εκτελεί το αντίστοιχο πρότυπο (template)
- d. Το πρότυπο δημιουργεί ένα κείμενο XML σαν αποτέλεσμα, και
- e. Η διαδικασία επαναλαμβάνεται αναδρομικά στα(φωλιασμένα) παιδιά της ρίζας.

Για την επεξεργασία των εγγράφων XML χρησιμοποιούνται και οι τεχνολογίες XSLT και XSLFO (XSL Formating Objects), οι οποίες μαζί με την σύσταση xpath(XML Path Language) υπάγονται στην σύσταση XSL (XML Stylesheet Language) του W3C. Η σύσταση XSL αφορά στο στυλ των εγγράφων XML και μας δίνει την δυνατότητα να μετατρέπουμε έγγραφα XML σε άλλα έγγραφα XML με διαφορετικό DTD ή XML Schema, όπως επίσης και σε έγγραφα διαφορετικών μορφών (HTML, text και άλλα). Ακόμη η XSL βοηθά όταν θέλουμε να δημοσιεύσουμε έναν μεγάλο όγκο από έγγραφα είτε να αναδιοργανώσουμε κάποια έγγραφα XML ώστε να δημιουργήσουμε πίνακες περιεχομένων ή άλλες χρηστικές δομές αναπαράστασης των πληροφοριών.

Η σύσταση XSLT αφορά στον μετασχηματισμό ενός εγγράφου XML, ενώ η σύσταση XSLFO στην μορφοποίησή του για παρουσίαση. Την ίδια λειτουργία με την XSLFO επιτελεί και η τεχνολογία CSS (Cascading Style Sheets) ωστόσο η CSS είναι πολύ δημοφιλέστερη και μονοπωλεί το ενδιαφέρον των κατασκευαστών λογισμικού και παροχών περιεχομένου³⁰.

XSLT

Τα αρχεία XSLT ονομάζονται και XSLT stylesheets και έχουν σύνταξη παρόμοια με αυτή της XML. Κάθε XSLT stylesheets είναι ένα καλά δομημένο (well formed) XML έγγραφο. Τα αρχεία XSL stylesheets βοηθούν πολύ σε κάθε προσπάθεια δημοσίευσης περιεχομένου που υποστηρίζει XML. Ένα χαρακτηριστικό

³⁰ H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML How to program, Prentice Hall, 2001
Sall, K.B., 2002. XML Family of Specifications. A Practical Guide. Addison Wesley.

παράδειγμα είναι η δημιουργία της μορφοποίησης ενός website από περιεχόμενο μιας βάσης δεδομένων βασισμένης σε XML. Στην περίπτωση αυτή ένα αρχείο με οδηγίες μορφοποίησης XSLT βοηθά στην μετατροπή του εγγράφου .XML σε ένα έγγραφο .HTML. Η συνεισφορά του XSLT επεκτείνεται όταν θέλουμε να έχουμε πολλές διαφορετικές εκδόσεις του περιεχομένου ενός website όπως για παράδειγμα αρχεία .HTML, αρχεία απλού κειμένου.txt (plain text), αρχεία για εμφάνιση από φορητές συσκευές μέσω WAP (WML), κ .α.

Για την αναγνώριση των XSLT stylesheets και την εκτέλεση των XML μετασχηματισμών που αυτά ορίζουν απαιτείται κάποιο πρόγραμμα XSLT processor. Τέτοια προγράμματα είναι ενσωματωμένα σε μεγαλύτερα πακέτα επεξεργασίας εγγράφων XML, όπως είναι το XML Spy. XSLT processors επίσης ενσωματώνονται και σε browsers, όπως για παράδειγμα ο Internet Explorer 6, ο οποίος διαθέτει το πρόγραμμα MSXML parser.

Γενικότερα το βασικό ή δομικό στοιχείο σε ένα XSLT stylesheet είναι το element `<xsl:stylesheet>`. Διαθέτει σαν υποστοιχεία μια σειρά από elements `<xsl:template>....<xsl:template>`. Μέσα σε αυτά τα `<xsl:template>` elements εκφράζονται οι μετασχηματισμοί που πρέπει να γίνουν στο έγγραφο στο έγγραφο XML. Κάθε template περιέχει ένα attribute με όνομα match. Το match είναι μια διαδρομή για το στοιχείο του XML εγγράφου στο οποίο αναφέρεται το template. Παράλληλα με το match, κάθε template υπάρχει μια δήλωση `<xsl:apply-templates/>` η οποία είναι μια κλήση που πρέπει να γίνει στο stylesheet κατά την ιεραρχική σάρωση των στοιχείων του πηγαίου XML εγγράφου από τον αναλυτή.

Τα βασικά σύμβολα που χρησιμοποιούνται για την σύνταξη των διαδρομών είναι τα σύμβολα / (στοιχείο ρίζα και άμεσος απόγονος), // (όλοι οι απόγονοι), * (όλοι οι άμεσοι απόγονοι) και | (επιλογή ενός εκ των δύο στοιχείων που βρίσκονται εκατέρωθεν του συμβόλου). Οι διαδρομές αντιστοιχίζονται τόσο σε elements όσο και σε attributes του πηγαίου XML εγγράφου. Η διαδρομή που αναφέρεται σε ένα στοιχείο url με attribute το protocol που έχει τιμή mailto γράφεται ως `match="url[@protocol='mailto']"`, δηλαδή το attribute στο οποίο γίνεται η αναφορά μπαίνει σε αγκύλες και πριν από το όνομα του τοποθετείται το σύμβολο.

Μπορούμε να χρησιμοποιήσουμε μια από τις έτοιμες συναρτήσεις της XSL, οι οποίες παίρνουν μηδέν ή περισσότερα ορίσματα και επιστρέφουν κάτι σχετικό με το πηγαίο έγγραφο. Για παράδειγμα η συνάρτηση `text()` επιστρέφει τα περιεχόμενα ενός στοιχείου.

Παράλληλα με τις έτοιμες συναρτήσεις μπορούμε να ορίσουμε και δικές μας προσθέτοντας στο XSLT stylesheet ένα στοιχείο `<xsl:fuctions >` ως εξής:

```
<xsl:fuctionsns="myns" type="text/javascript">
fuctiontoday()
{
returnDate(), toString()
}
</ xsl:fuctions >
```

Το πρόγραμμα parser ξεκινά την σάρωση του πηγαίου XML εγγράφου από την ρίζα προς τους τελευταίους κόμβους ιεραρχικά, με βάση έναν αλγόριθμο DFS (Depth First Search). Σε κάθε βήμα της σάρωσης, ο αναλυτής προσπαθεί να αντιστοιχίσει τον τρέχοντα κόμβο ένα από τα πρότυπα που υπάρχουν στο XSLT stylesheet. Αν υπάρχει μια τέτοια αντιστοιχία, ο αναλυτής δημιουργεί τους κόμβους που του λέει το stylesheet στο παραγόμενο δέντρο (παραγόμενο XML έγγραφο). Όταν ο αναλυτής συναντήσει το στοιχείο `<xsl:apply-templates>`, τότε προχωρά στα παιδιά του τρέχοντος κόμβου και επαναλαμβάνει την διαδικασία, δηλαδή προσπαθεί να τα αντιστοιχίσει με κάποιο πρότυπο. Αν ο αναλυτής δεν βρει καμία αντιστοιχία για ένα στοιχείο τότε εφαρμόζει το ενσωματωμένο πρότυπο του αναλυτή. Το ενσωματωμένο πρότυπο περιέχει συνήθως μόνο μια κλήση `<xsl:apply-templates/>` με χαρακτηριστικό `match="* | /"` που σημαίνει ότι ο αναλυτής οδηγείται στα παιδιά του στοιχείου για το οποίο δεν είχε αρχικά βρει αντιστοιχία στο XSLT stylesheet. Έτσι η διαδικασία δεν τερματίζεται, παρά μόνο όταν ολοκληρωθεί η σάρωση όλου του πηγαίου XML εγγράφου.

Επίσης κάποια στοιχεία έχουν σαν μοναδικό παιδί τους ένα κείμενο και για το κείμενο αυτό δεν έχει οριστεί κάποιο template στο XSLT. Τότε πάλι εκτελείται ένα ενσωματωμένο template του αναλυτή που υπαγορεύει την παρουσίαση του κειμένου στο τρέχον σημείο του παραγόμενου δέντρου.

Τα attributes που ορίζονται στο παραγόμενο XML δέντρο είναι είτε ρητά δηλωμένα στο XSLT είτε υπολογίζονται κατά την δημιουργία του παραγόμενου δέντρου. Η δεύτερη περίπτωση γίνεται με την δήλωση `<xsl:attributename="...">.....</xsl:attribute>` για τον υπολογισμό της τιμής του attribute name κατά την δημιουργία του παραγόμενου δέντρου.

Για την δημιουργία νέων αντικειμένων είναι αναγκαίο να υπάρχει μια σειρά από εντολές XSLT. Για παράδειγμα υπάρχουν οι εντολές `<xsl:element>` και

<xsl:attribute> για την δημιουργία ενός στοιχείου ή ενός χαρακτηριστικού στο παραγόμενο δέντρο και υπολογισμό της τιμής του κατά την δημιουργία του δέντρου.

Όπως έχουμε πει υπάρχουν δύο τρόποι για να παράγουμε το τελικό επιθυμητό HTML έγγραφο. Είτε να εφαρμόσουμε εμείς εσωτερικά στον αναλυτή μας το XSLT και να παράγουμε το HTML είτε να δώσουμε στον τελικό χρήστη το πηγαίο XML έγγραφο και το XSLT και ο μετασχηματισμός να εκτελεστεί στην πλατφόρμα του χρήστη. Στην δεύτερη αυτή περίπτωση πρέπει να είμαστε προσεκτικοί με θέματα συμβατότητας ώστε το XSLT μας να καλύπτει τις προδιαγραφές του αναλυτή του χρήστη.

Σχετικά με την προτεραιότητα εφαρμογής των προτύπων του XSLT, ισχύουν τα εξής: Τα πρότυπα με τις πιο συγκεκριμένες διαδρομές προηγούνται αυτών που είναι λιγότερο συγκεκριμένα. Αν αυτό το κριτήριο δεν αρκεί για την επιλογή του επόμενου template, τότε μεγαλύτερη προτεραιότητα έχει το template που βρίσκεται τελευταίο στο XSLT stylesheet. Όταν θέλουμε να επιβάλλουμε σε ένα XML έγγραφο με ένα αρχικό XML Schema να ακολουθεί ένα άλλο XML Schema, τότε πρέπει να δημιουργήσουμε ένα XSLT stylesheet για μετατροπή μεταξύ των δύο schemas και εν συνέχεια να εκτελέσουμε αυτό το XSLT stylesheet μέσα από έναν browser ή έναν XSLT αναλυτή. Εκεί θα πρέπει στην κορυφή του εγγράφου να γίνει μια δήλωση για αναφορά στο XSLT που χρησιμοποιούμε. Η δήλωση αυτή έχει μορφή:

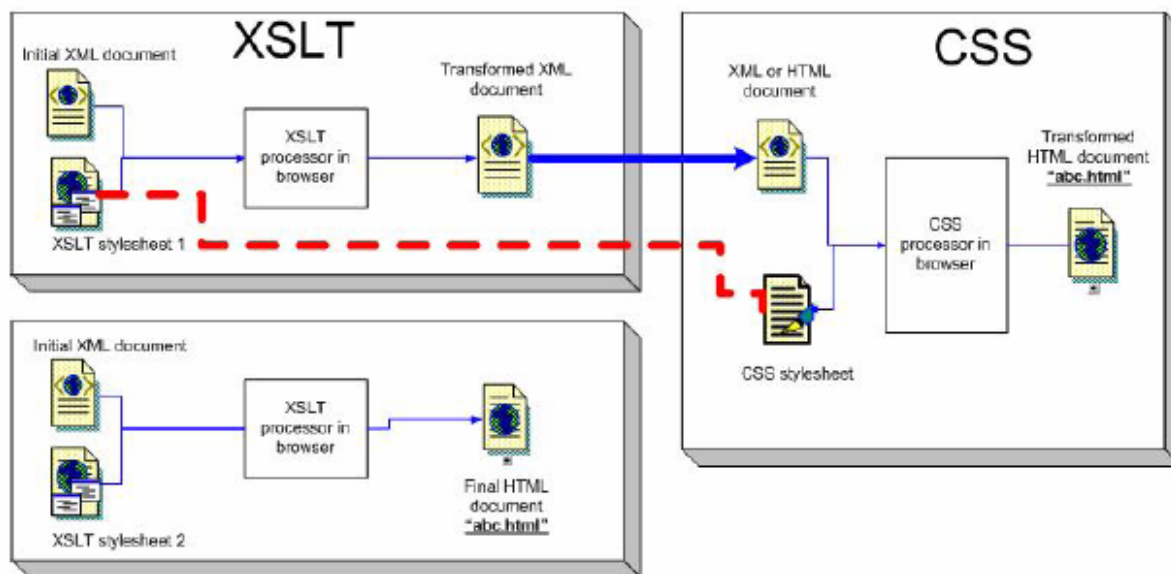
```
<?XML-stylesheethref="..."type="text/xsl"?>.
```

Ένα XSLT stylesheet μπορεί επίσης να περιέχει μια αναφορά σε ένα CSS stylesheet για την τελική μορφοποίηση του παραγόμενου εγγράφου XML. Έτσι με την χρήση των templates στα XSLT stylesheet έχουμε δύο επιλογές:

- Να χρησιμοποιήσουμε XSLT, το οποίο εφαρμόζεται πάνω στο επιθυμητό XML έγγραφο και να παράγουμε το τελικό HTML ή XHTML έγγραφο.
- Να χρησιμοποιήσουμε XSLT που έχει αναφορά σε κάποιο CSS και να παράγουμε ένα XML έγγραφο. Το τελικό αυτό XML έγγραφο μορφοποιείται με βάση τις οδηγίες μορφοποίησης του CSS στο οποίο υπάρχει αναφορά³¹.

Τα παραπάνω παρουσιάζονται και σχηματικά ως εξής:

³¹ Damien Foggon, Daniel Maharry, Chris Ullman and Karli Watson. Programming Microsoft .NET XML Web Services, Microsoft Press, 2004.
Mathew Macdonald. Microsoft .NET Distributed Applications: Integrating XML Web Services and .NET Remoting, 2001
David Carlson. Modeling XML applications with UML, Addison-Wesley, 2000
H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML How to program, Prentice Hall, 2001



Εικόνα 20: Σύγκριση των XSLT και CSS

XSLFO

Τα XSLFO (XML Formatting Objects) είναι αρχεία XSL που όμως έχουν διαφορετικά tags μέσα σε κάθε `<xsl:template>` element από αυτά που υπάρχουν μέσα σ' ένα XSLT stylesheet. Τα νέα tags που χρησιμοποιούνται στα XSLFO βασίζονται στην λογική των CSS stylesheets. Κανένας από τους γνωστούς browser δεν υποστηρίζει το XSLFO αλλά οι περισσότεροι υποστηρίζουν πλέον εναλλακτικά το XSLFO και το CSS2. Η συντριπτική πλειοψηφία των εγγράφων βασίζονται στην HTML και την XHTML για την εμφάνιση τους στο Web. Με αυτή τη λογική το XSLFO μπορεί να αποτελεί μια τελειώς νέα λογική, αλλά αυτό που χρησιμοποιείτε στην πράξη και θα συνεχίσει να υποστηρίζεται στο μέλλον είναι ο συνδυασμός XSLT και XHTML, ως εξέλιξη της απλής HTML³²

3.1.4.3 XQUERY

Η XQuery είναι μια γλώσσα ερωτημάτων(query) που έχει σχεδιαστεί για να μετατρέπει τα ερωτήματα και τα δεδομένα που συλλέγει σε μορφή XML. Η γλώσσα αυτή αναπτύχθηκε από την XML ομάδα της διεθνούς επιτροπής W3C.

³² Damien Foggon, Daniel Maharry, Chris Ullman and Karli Watson. Programming Microsoft .NET XML Web Services, Microsoft Press, 2004.

H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML How to program, Prentice Hall, 2001

Το βασικό πλεονέκτημα της XQuery είναι πως μας δίνει τη δυνατότητα να εξάγουμε και να χειριζόμαστε τα δεδομένα από έγγραφα XML ή οποιαδήποτε πηγή δεδομένων που μπορεί να θεωρηθεί ως XML, όπως σχεσιακές βάσεις δεδομένων ή έγγραφα γραφείου.

Αντίθετα με την XSLT, η XQuery δεν αναπαριστά ερωτήματα της XML αλλά μοιάζει περισσότερο με SQL. Ερωτήματα που είναι οργανωμένα σε παραστάσεις που αποτελούνται από τέσσερις ενότητες: for ,let, where και return που εν συντομία αναφέρονται και ως “FLWR”. Αναλυτικά η κάθε μια ενότητα:

- **For:** δίνει μια σειρά από μεταβλητές που διαφέρουν από τα αποτελέσματα των αναπαραστάσεων XPath. Όταν καθορίζονται περισσότερες από μια μεταβλητές, τα αποτελέσματα περιλαμβάνουν το καρτεσιανό γινόμενο των πιθανών τιμών που μπορεί να λάβουν οι μεταβλητές, κάνοντας τον όρο for κάτι σαν το from σ’ ένα SQL ερώτημα.

- **Let:** Επιτρέπει να αντιστοιχούνται περίπλοκες παραστάσεις σε ονόματα μεταβλητών προκειμένου να γίνει πιο απλή η αναπαράσταση.

- **Where:** Ακριβώς όπως και ο όρος where της SQL, κάνει επιπλέον ελέγχους για τις συνδεδεμένες εγγραφές από την ενότητα for.

- **Return:** Επιτρέπει την κατασκευή αποτελεσμάτων στην XML.

Αν και η XQuery αρχικά σχεδιάστηκε ως μια γλώσσα επερωτήσεων για μεγάλες συλλογές εγγράφων XML, είναι επίσης ικανή να μετατρέψει μεμονωμένα έγγραφα. Ως εκ τούτου, οι δυνατότητές της συμπίπτουν με την XSLT, η οποία σχεδιάστηκε για να επιτρέψει ρητά έγγραφα εισόδου XML ώστε να μετατραπεί σε HTML ή άλλες μορφές. Τα XSLT 2.0 and XQuery πρότυπα αναπτύχθηκαν από χωριστές ομάδες εργασίας στο πλαίσιο του W3C, που εργάζονται από κοινού για να εξασφαλιστεί μια κοινή προσέγγιση κατά περίπτωση. Μοιράζονται το ίδιο μοντέλο δεδομένων (XDM), σύστημα τύπου, και βιβλιοθήκη συναρτήσεων, και οι δύο περιλαμβάνουν XPath 2.0 ως sublanguage.

Οι δύο γλώσσες, ωστόσο, έχουν τις ρίζες τους σε διαφορετικές παραδόσεις και εξυπηρετούν τις ανάγκες των διαφορετικών κοινοτήτων. Η XSLT είχε κυρίως σχεδιαστεί ως μια style sheet language., του οποίου πρωταρχικός στόχος ήταν να καταστεί η XML πιο εύκολη και ευανάγνωστη για τον χρήστη XQuery ήταν κυρίως σχεδιαστεί ως μια βάση δεδομένων γλώσσα επερωτήσεων στην παράδοση της SQL. Επειδή οι δύο γλώσσες προέρχονται από διαφορετικές κοινότητες, η XSLT

είναι ισχυρότερη κατά το χειρισμό της αφήγησης εγγράφων με πιο ευέλικτη δομή, ενώ η XQuery είναι ισχυρότερη στα δεδομένα του χειρισμού, για παράδειγμα, κατά την εκτέλεση σχεσιακών ενώνων.

Διάφορες επιστημονικές μελέτες που έχουν γίνει κατά καιρούς, έχουν δείξει ότι η XQuery είναι πιο εύκολη στην ανάγνωσή της, ειδικά για κάποιον που έχει ασχοληθεί με SQL σε σχέση με την XSLT. Λόγω της ευελιξίας που προσφέρει η XQuery καθίσταται πιο εύχρηστη από την XSLT, η οποία περιλαμβάνει ένθετα στοιχεία. Ακόμα, η σύνταξη XQuery είναι πιο κατάλληλη για ενσωμάτωση σε παραδοσιακές γλώσσες προγραμματισμού όπως Java ή C#.

Αντίθετα, η XSLT είναι πιο ισχυρή από την XQuery για τις εφαρμογές επειδή επιτρέπει τις μικροαλλαγές σε ένα έγγραφο, όπως για παράδειγμα, την διαγραφή όλων των στοιχείων με ένα συγκεκριμένο όνομα. Μια δυνατότητα που λείπει από την XQuery είναι ο πολυμορφισμός και η δυνατότητα δυναμικής δέσμευσης. Η απουσία αυτής της ικανότητας αρχίζει να γίνεται αισθητή όταν γράφονται μεγάλες εφαρμογές, ή κατά τη σύνταξη κώδικα που έχει σχεδιαστεί για να επαναχρησιμοποιηθεί σε διαφορετικά περιβάλλοντα. Η XSLT παρέχει δύο συμπληρωματικούς μηχανισμούς σε αυτόν τον τομέα, πρώτον τη δυναμική αντιστοίχιση των κανόνων ενός Template, και δεύτερον την ικανότητα να παρακάμπτει τους κανόνες που χρησιμοποιούν `xsl:import`, που καθιστούν δυνατό να γράψουν εφαρμογές με πολλαπλά layers.

Τέλος, παρόλο που αυτές οι δυνατότητες κάνουν την XSLT πιο ισχυρή δεν μειώνουν τη δυναμική της XQuery. Αντιθέτως, προσδίδουν στην γλώσσα το πλεονέκτημα να είναι πιο εύκολη στην ανάλυσή της αλλά και να εντοπίζει τα σφάλματα ευκολότερα κατά τη μεταγλώττισής της³³.

3.1.5 Βάσεις Δεδομένων και XML

Η γλώσσα XML, παρόλο που αναπτύχθηκε αρχικά για να υποστηρίξει εφαρμογές διαδικτύου, είναι στενά συνδεδεμένη με την τεχνολογία συστημάτων βάσεων δεδομένων.

Ένα κείμενο XML είναι ουσιαστικά μια συλλογή σχετιζόμενων δεδομένων και πολλές φορές αντιμετωπίζεται σαν μια Βάση Δεδομένων. Ένα κείμενο XML, ως βάση δεδομένων, συγκεντρώνει κάποια πλεονεκτήματα και μειονεκτήματα σε σχέση με τα αρχεία των κλασσικών βάσεων δεδομένων.

³³ W3C XML Query (XQuery), 2014, <http://www.w3.org/XML/Query>

Ένα από τα πλεονεκτήματα, είναι ότι ένα κείμενο XML έχει την ιδιότητα της αυτοπεριγραφής, αφού μέσα από τις ετικέτες και τα γνωρίσματα, η γλώσσα XML περιγράφει τα δεδομένα του κειμένου. Όμως, η σημασιολογία της περιγραφής αυτής είναι περιορισμένη.

Επίσης, όσον αφορά την μεταφερσιμότητα του κειμένου XML, η γλώσσα XML είναι αναγνώσιμη από ανθρώπους και μηχανές και είναι ανεξάρτητη του υλικού ή του λογισμικού του υπολογιστικού συστήματος, αφού τα κείμενα XML υιοθετούν την κωδικοποίηση κατά Unicode. Τέλος, τα κείμενα XML μπορούν να περιγράψουν ιεραρχικές και δικτυακές δομές με απλό τρόπο.

Από την άλλη, ένα κείμενο XML, ως βάση δεδομένων, εμφανίζει αρκετά μειονεκτήματα: τα στοιχεία του είναι διατεταγμένα, σε αντίθεση με τα λογικά μοντέλα βάσεων δεδομένων, όπως για παράδειγμα το σχεσιακό, όπου η έννοια της διάταξης των δεδομένων δεν υφίσταται. Για παράδειγμα, τα παρακάτω δύο κείμενα XML δεν είναι ισοδύναμα:

Κείμενο 1ο:

```
<πρόσωπο>  
<όνομα>Messi</όνομα>  
<επώνυμο>Lionel</επώνυμο>  
</πρόσωπο>
```

Κείμενο 2ο:

```
<πρόσωπο>  
<επώνυμο>Messi</επώνυμο>  
<όνομα>Lionel</όνομα>  
</πρόσωπο>
```

Ένα άλλο μειονέκτημα των κειμένων XML αφορά τις συντακτικές ιδιαιτερότητες της γλώσσας τους. Στην XML, αντιθέτως με τις βάσεις δεδομένων, επιτρέπεται η ανάμειξη σχολίων και στοιχείων σε ένα κείμενο. Για παράδειγμα, το παρακάτω κείμενο συμβαδίζει με τους κανόνες της γλώσσας:

```
<διάλεξη>Βάσεις Δεδομένων και XML  
<ομιλητής>LionelMessi</ομιλητής>  
</διάλεξη>
```

Επίσης, η γλώσσα XML επιτρέπει την υιοθέτηση διαφορετικών δομικών μονάδων για την έκφραση της αυτής κατάστασης, δηλαδή η καταγραφή δεδομένων σε ένα κείμενο XML μπορεί να γίνει με εναλλακτική χρήση γνωρισμάτων ή στοιχείων, όπως στο παράδειγμα που ακολουθεί, τα ακόλουθα τρία κείμενα μεταφέρουν την ίδια πληροφορία:

Κείμενο 1ο:

```
<πρόσωπο>  
<όνομα>Διονύσης</όνομα>  
<ηλικία>5</ηλικία>  
</πρόσωπο>
```

Κείμενο 2ο:

```
<πρόσωπο όνομα="Διονύσης" ηλικία="5"/>
```

Κείμενο 3ο:

```
<πρόσωπο ηλικία="5">  
<όνομα>Διονύσης</όνομα>  
</πρόσωπο>
```

Όπως αναφέρθηκε στην προηγούμενη Ενότητα, η γλώσσα XML περιβάλλεται από μια συλλογή εργαλείων. Ορισμένα από τα εργαλεία αυτά έχουν σαν στόχο τη διαχείριση των δεδομένων, που φιλοξενούνται σε κείμενα XML. Επομένως, καθώς ένα κείμενο XML αντιμετωπίζεται σαν μια βάση δεδομένων, τα παραπάνω εργαλεία μπορεί να θεωρηθεί ότι συνθέτουν το αντίστοιχο Σύστημα Διαχείρισης Βάσης Δεδομένων (ΣΔΒΔ)³⁴.

3.1.6 Πλεονεκτήματα και Μειονεκτήματα της XML

Οι παλιότερες μορφές μετάδοσης, είτε με χρήση δομημένων κειμένων (CSV) είτε με χρήση ροής δεδομένων (CORBA) αποδείχθηκαν δύσκολες στην χρήση τους, αφού τα CSV δεν είχαν πάντα την ίδια κωδικοποίηση, ενώ το CORBA δεν είχε την δυνατότητα υποστήριξης με τον ίδιο τρόπο σε κάθε γλώσσα προγραμματισμού. Η XML είναι σήμερα η κοινή διάλεκτος για την ανταλλαγή δεδομένων μεταξύ προγραμμάτων, υπολογιστικών συστημάτων κ.α.

Οι κυριότεροι παράγοντες οι οποίοι οδήγησαν στην επιτυχία της είναι οι εξής:

- Η XML είναι απλό κείμενο, εύκολα αναγνώσιμο από ανθρώπους.
- Η XML ασχολείται μόνο με τη σημασία των δεδομένων, και όχι με την επεξεργασία τους.
- Εύκολη επεξεργασία αρχείων XML, με χρήση βιβλιοθηκών δημιουργημένες για αυτόν τον σκοπό ή με χρήση της ίδιας της XML (XSL).
- Η κάθε γλώσσα προγραμματισμού μπορεί να δημιουργήσει δεδομένα σε XML μορφή, να τα διαβάσει και να τα επεξεργαστεί. Ουσιαστικά, ένα αρχείο XML

³⁴ H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML How to program, Prentice Hall, 2001
Sall, K.B., 2002. XML Family of Specifications. A Practical Guide. Addison Wesley.

είναι ένα καλά δομημένο αρχείο κειμένου και το μόνο που το περιορίζει είναι η ύπαρξη ενός tag όπου να συμπεριλαμβάνεται η δομή των δεδομένων.

- Η XML διευκολύνει την μετάδοση αντικειμενοστραφών πληροφοριών.

Υπάρχουν φυσικά και περιπτώσεις που η χρήση XML δεν ενδείκνυται:

- Η κωδικοποίηση σε XML οδηγεί σε μεγάλα αρχεία, αφού όλα τα δεδομένα είναι μέσα σε ετικέτες. Συνήθως πάνω από τα 2/3 ενός αρχείου XML καταλαμβάνεται από ετικέτες.
- Η κωδικοποίηση των δεδομένων σε απλό κείμενο είναι χρονοβόρα και πολλές φορές απαγορευτική για επιστημονικές εφαρμογές ή για εφαρμογές όπου η απόδοση έχει μεγάλη σημασία³⁵.

3.2 Ελεύθερο Λογισμικό

Το ελεύθερο λογισμικό είναι λογισμικό το οποίο μπορεί να χρησιμοποιηθεί, να αντιγραφεί, να μελετηθεί, να τροποποιηθεί και να αναδιανεμηθεί χωρίς κανένα απολύτως περιορισμό. Η ελευθερία από τέτοιους περιορισμούς είναι το βασικό στοιχείο στην ιδέα του «ελεύθερου λογισμικού». Ακόμα το συναντάμε με τα ονόματα ανοιχτό λογισμικό ή λογισμικό ανοιχτού κώδικα. Το αντίθετο του ελεύθερου λογισμικού να είναι το ιδιόκτητο λογισμικό ή το λογισμικό που πωλείται για κέρδος, όπως για παράδειγμα το εμπορικό λογισμικό.

Με βάση την ισχύουσα νομοθεσία περί πνευματικής ιδιοκτησίας, η ελεύθερη αντιγραφή, διανομή και τροποποίηση του λογισμικού απαγορεύεται και για αυτό το λόγο, οι εκδόσεις ελεύθερου λογισμικού κάνουν χρήση ειδικής άδειας (free software license) και έτσι παραχωρείται το δικαίωμα αντιγραφής, τροποποίησης και αναδιανομής του λογισμικού αμέσως στους χρήστες.

Σύμφωνα με το Ίδρυμα Ελεύθερου Λογισμικού, οι άδειες χρήσης ελεύθερου λογισμικού πρέπει να περιλαμβάνουν τις εξής ελευθερίες:

- Ελευθερία χρήσης του προγράμματος για οποιονδήποτε σκοπό.
- Ελευθερία μελέτης και τροποποίησης του προγράμματος.
- Ελευθερία αντιγραφής του προγράμματος.
- Ελευθερία βελτίωσης του προγράμματος και επανέκδοσης του, προς το συμφέρον της κοινότητας των χρηστών.

³⁵Πλεονεκτήματα και Μειονεκτήματα της XML, 2015,
http://www0.dmst.aueb.gr/louridas/lectures/dais/web_intro/ar01s22.HTML

Το ελεύθερο λογισμικό είναι αρκετά χρήσιμο στην τεχνολογική κοινότητα, με βασικότερο πλεονέκτημα την δυνατότητα περαιτέρω έρευνας και επεξεργασίας προγραμμάτων μέσω της λογικής της ανάπτυξης του λογισμικού ανοιχτού κώδικα.

Επίσης, τα προγράμματα Ελεύθερου Λογισμικού μπορούν να προσαρμοστούν στις ανάγκες κάθε οργανισμού λόγω της παραμετροποίησης, να αποτελέσουν σημαντικό εκπαιδευτικό εργαλείο ή εργαλείο για απόκτηση προγραμματιστικής εμπειρίας από αυτούς που αναπτύσσουν κώδικα. Τέλος, η χρήση ανοιχτού κώδικα δεν περιορίζει τον οργανισμό σε μια σχέση εξάρτησης από εταιρίες, αφού το κόστος των προγραμμάτων ανοιχτού κώδικα είναι τις περισσότερες φορές μηδενικό.

Όπως είναι φυσικό, υπάρχουν αρκετά μειονεκτήματα, με βασικότερο την δύσκολη υποστήριξη των προγραμμάτων ανοιχτού κώδικα, αφού δεν υπάρχει επίσημη εταιρία με καταρτισμένους τεχνικούς και πολλές φορές υπάρχει ελλιπής τεκμηρίωση στα προγράμματα ανοιχτού κώδικα. Τα αντίστοιχα εμπορικά προγράμματα προσφέρουν καλύτερη υποστήριξη, τεκμηρίωση και συνεργασία.

Επίσης, όταν ένα πρόγραμμα εγκαταλειφθεί από τους δημιουργούς του, ο οργανισμός μένει πρωτίστως χωρίς υποστήριξη και έπειτα χωρίς μελλοντικές προοπτικές. Τέλος, ένας οργανισμός με την εγκατάσταση ενός λογισμικού ανοιχτού κώδικα χρειάζεται να έχει αρκετά καταρτισμένο τμήμα πληροφορικής προκειμένου να αντιμετωπίζει τις δυσκολίες που θα παρουσιαστούν³⁶.

3.3 Συστήματα Διαχείρισης Περιεχομένου (Content Management System CMS)

Τα Συστήματα Διαχείρισης Περιεχομένου (Content Management System) είναι διαδικτυακές εφαρμογές οι οποίες επιτρέπουν την online τροποποίηση του περιεχομένου ενός δικτυακού τόπου. Οι διαχειριστές, μέσα από το διαδίκτυο ενημερώνουν το περιεχόμενο στο CMS, το οποίο είναι εγκατεστημένο σ' ένα εξυπηρετητή διαδικτύου. Αυτές οι αλλαγές, είναι αυτόματα διαθέσιμες πάλι μέσω του διαδικτύου, σε όλους τους επισκέπτες και χρήστες του δικτυακού τόπου.

Ένα σύστημα διαχείρισης περιεχομένου υποστηρίζει την δημιουργία, διαχείριση, διανομή, έκδοση και ανεύρεση επιχειρησιακών πληροφοριών. Για παράδειγμα, μπορεί να καλύψει ολόκληρο τον κύκλο ζωής των ιστοσελίδων ενός ιστοχώρου με την διάθεση απλών εργαλείων που χρησιμοποιούνται από την

³⁶Ελεύθερο λογισμικό/ Λογισμικό ανοιχτού κώδικα, 2015, <https://ellak.gr/>
Free Software foundation, 2015, <http://www.fsf.org/>

δημιουργία περιεχομένου, μέχρι την έκδοση και τελικά την αρχειοθέτησή του. Η χρήση του μπορεί να επεκταθεί και πέρα από την διαχείριση ενός ιστοχώρου.

Επειδή ο όρος CMS έχει ευρύτερη σημασία, έχει καθιερωθεί μια εναλλακτική ονομασία που χρησιμοποιείται όταν ένα Σύστημα διαχείρισης Περιεχομένου χρησιμοποιείται αποκλειστικά για διαδικτυακές εφαρμογές. Έτσι με τον όρο διαχείριση διαδικτυακού Περιεχομένου (Web Content Management) αναφερόμαστε στα συστήματα που επικεντρώνονται συνήθως στο διαδικτυακό περιεχόμενο ενός εταιρικού ιστοχώρου (website) ή ενός ενδοδικτύου (intranet).

Το Σύστημα διαχείρισης διαδικτυακού Περιεχομένου, είναι μια δικτυακή εφαρμογή σχεδιασμένη για να διευκολύνει χρήστες, όχι απαραίτητα προγραμματιστές, με την προσθήκη, την επεξεργασία, την δημοσίευση περιεχομένου και γενικά την διαχείριση ενός ισόχωρου χωρίς την χρήση development tools ή γνώση διαφόρων γλωσσών προγραμματισμού.

Ένα ολοκληρωμένο CMS είναι απαραίτητο να μπορεί να διαχειρίζεται όλες τις δυναμικές πληροφορίες του site και να προσφέρει υπηρεσίες που εξυπηρετούν πλήρως τις ανάγκες των διαχειριστών του. Κάποια από τα πλεονεκτήματα και τα χαρακτηριστικά ενός ολοκληρωμένου CMS είναι τα παρακάτω:

- Η γρήγορη ενημέρωση, διαχείριση και αρχειοθέτηση του περιεχομένου του δικτυακού τόπου.
- Η ενημέρωση του περιεχομένου από οπουδήποτε.
- Η εύκολη χρήση και η άμεση γνώση του τελικού αποτελέσματος, όπως συμβαίνει με τους κειμενογράφους.
- Οι μη υποχρεωτικές ειδικές τεχνικές γνώσεις από τους διαχειριστές του.
- Η δυνατότητα αναζήτησης του περιεχομένου που καταχωρείται και η αυτόματη δημιουργία αρχείου.
- Η ασφάλεια και η προστασία σχεδιασμού του site από λανθασμένες ενέργειες, οι οποίες πιθανόν να δημιουργήσουν προβλήματα στην εμφάνιση του.
- Ο διαχωρισμός του περιεχομένου από το σχεδιασμό και την πλοήγηση του δικτυακού τόπου.
- Η αλλαγή σχεδιασμού ή τρόπου πλοήγησης χωρίς απαραίτητα την ενημέρωση όλων των σελίδων από τον χρήστη.
- Η αυτόματη δημιουργία συνδέσμων μεταξύ των σελίδων και η αποφυγή προβλημάτων ανύπαρκτων σελίδων (404 error page).

- Ο μικρότερος φόρτος στο εξυπηρετητή και η χρήση λιγότερου χώρου, αφού δεν υπάρχουν πολλές επαναλαμβανόμενες στατικές σελίδες.
- Τα στατιστικά διαχείρισης μελών (newsletters).
- Η καταχώριση όλου του περιεχομένου στις βάσεις δεδομένων, που μπορούν ευκολότερα και γρηγορότερα να προστατευτούν με την τήρηση αντίγραφων ασφάλειας.
- Η εφαρμογή διαχείρισης και προβολής διαφημιστικών banners, δημοσκοπήσεων και παραμετροποίησης (personalisation).
- Η δυνατότητα παρουσίασης του περιεχομένου σε συνεργαζόμενα sites³⁷.

3.4 CMS Ανοιχτού Κώδικα

Σε εφαρμογές ανοικτού κώδικα επιτρέπεται η πλήρης πρόσβαση και η αλλαγή του πηγαίου κώδικα. Με τη χρήση των συγκεκριμένων CMS, πραγματοποιείται σημαντική μείωση του κόστους της εφαρμογής και τις περισσότερες φορές όλη η εφαρμογή είναι στο διαδίκτυο με δυνατότητα στον ενδιαφερόμενο να την κατεβάσει με μηδενικό κόστος. Παρόλα αυτά, απαιτούνται τουλάχιστον κάποιες τεχνικές γνώσεις για να στηθεί η εφαρμογή και να μπει σε λειτουργία. Τα CMS ανοικτού κώδικα υποστηρίζονται από μία κοινότητα χρηστών και προγραμματιστών και συχνά συνοδεύονται από πρόσθετα (plugins) τα οποία δημιουργούνται και προσφέρονται από την κοινότητα.

Κύριο πλεονέκτημα των CMS ανοικτού κώδικα είναι η δυνατότητα της δημιουργίας ενός διαδικτυακού τόπου από την αρχή χωρίς εξειδικευμένες τεχνικές γνώσεις και με χαμηλό κόστος. Κόστος υπάρχει μόνο για την παροχή υπηρεσιών και την υποστήριξη και όχι για το λογισμικό. Επίσης, τα CMS ανοικτού κώδικα εύκολα παραμετροποιούνται με την υποστήριξη και τη βοήθεια των χρηστών του Web-CMS από τα forums του και εύκολα ολοκληρώνονται με τα υπάρχοντα λογισμικά. Όμως, τα CMS ανοικτού κώδικα δεν είναι πάντα κατάλληλα για επίπεδο επιχειρήσεων μεγάλης έκτασης αφού δεν υπάρχει εμπορική υποστήριξη και επομένως πλήρης τεκμηρίωση. Τέλος, η εστίαση στην τεχνική και όχι στην εμπειρία του χρήστη οδηγεί στην έλλειψη χρηστικότητας³⁸.

³⁷ Εισαγωγή στα συστήματα διαχείρισης περιεχομένου, Θεόδωρος Δούβλης, 2015, <http://www.slideshare.net/rodotheos/CMS-421346>

³⁸ *Content Management Bible*, Bob Boiko. John Wiley & Sons, 2005.

ΚΕΦΑΛΑΙΟ 4

ΤΕΧΝΙΚΟ ΜΕΡΟΣ ΤΗΣ ΠΤΥΧΙΑΚΗΣ

4.1 Ανάλυση προβλήματος

Η υλοποίηση του online τουριστικού οδηγού του Μεσολογγίου είχε διπλό σκοπό. Πρώτον σχεδιάστηκε με σκοπό να εξυπηρετήσει τις ανάγκες των χρηστών που θέλουν να ενημερωθούν την πόλη, αλλά και για τις παροχές που τους προσφέρει και δεύτερον έπρεπε να σχεδιαστεί με την γλώσσα XML δηλαδή να μην χρησιμοποιήσουμε καθόλου κάποια βάση δεδομένων αλλά να δουλέψουμε κατά κόρον με το συγκεκριμένο πρότυπο για την περιγραφή και απεικόνιση των δεδομένων που πρέπει να έχει ένας τέτοιος οδηγός. Αρα έπρεπε να συνδυάσουμε την XML με άλλες τεχνολογίες προκειμένου να πάρουμε ένα καλο και λειτουργικό ιστότοπο.

4.2 Σχεδιασμός υλοποίησης

Προκειμένου να σχεδιαστεί ένα τέτοιο σύστημα χρειάστηκε να γίνουν μια σειρά από ενέργειες. Η πρώτη δουλειά ήταν να στήσουμε τον ένα τοπικό εξυπηρετητή για να μπορούμε να τρέξουμε τις γλωσσες σήμανσης που θα χρησιμοποιούσαμε. Έτσι επιλέξαμε τον xampp για να κάνει αυτή τη δουλειά.



Εικόνα 21: XAMPP

Επόμενο βήμα ήταν να κάνουμε ένα σχεδιάγραμμα σχετικά με τα περιεχόμενα που θα έχει το site για να μπορούμε να γράψουμε ή να επεξεργαστούμε τις πληροφορίες σχετικά με τον τόπο και να βρούμε πως ακριβώς θα φτιάξουμε την κάθε σελίδα.

<ul style="list-style-type: none">• Αρχική• Ιστορία	<ul style="list-style-type: none">• Εικόνες• Ξενοδοχεία
--	--

<ul style="list-style-type: none"> ○ Αρχαιότητα ○ Η Αιτωλική Συμπολιτεία ○ Ενετοκρατία ○ Η Έξοδος ○ Σύγχρονη Ελλάδα <ul style="list-style-type: none"> • Αξιοθέατα <ul style="list-style-type: none"> ○ Λιμνοθάλασσα ○ Αρχαιολογικοί Χώροι ○ Κτίρια ○ Εκκλησίες και Μονές ○ Ιστορικά Μνημεία ○ Μουσεία 	<ul style="list-style-type: none"> • Δραστηριότητες <ul style="list-style-type: none"> ○ Στο Νερό ○ Στην Γή ○ Στον Αέρα ○ Παραδοσιακοί Χοροί <ul style="list-style-type: none"> • Τοπικά Προϊόντα <ul style="list-style-type: none"> ○ Αυγοτάραχο ○ Ούζο ○ Ελαιόλαδο ○ Κρασί <ul style="list-style-type: none"> • Χάρτης • Επικοινωνία
---	---

Έπειτα μετά από έρευνα καταλήξαμε ότι ο καλύτερος συνδυασμός για να φτιάξουμε ένα τουριστικό οδηγό με τη χρήση της γλώσσας XML ήταν να βρούμε ένα CMS το οποίο να βασίζεται στην γλώσσα αυτή κ όχι σε κάποια βάση δεδομένων. Οπότε πιο διαδεδομένα CMS όπως το Joomla, το Drupal και το Wordpress απορρίφθηκαν για το λόγο της μη χρησιμοποίησης XML.

Ψάξαμε για xml-based CMS, δοκιμάσαμε αρκετά μεταξύ αυτών το XMLNuke, το Ingeniux , το WebJax, το Kumera.sharpCMS, το YellowTail, το Marix και το Webrender, πριν καταλήξουμε τελικά στο GetSimple CMS



Εικόνα 22: GetSimple CMS

Το GetSimple είναι ένα σύστημα διαχείρισης περιεχομένου ανοιχτού κώδικα, βασισμένο στην XML. Επιλέχθηκε για την απλότητα του στα βασικά στοιχεία που χρειάζεται ένα site, την ξεκάθαρη δομή του, την ευκολία και την ευελιξία στην χρήση, την μεγάλη πληροφόρηση μέσω των φόρουμ του και τις οδηγίες χρήσης, και τέλος την πληθώρα από προσθετα και templates τα οποία είναι εξίσου εύκολο να χρησιμοποιήσεις και να διαχειριστείς.

Το τελευταίο βήμα ήταν να βρούμε themes και προσθετα για το GetSimple τα οποία θα μας βοηθήσουν να εχουμε επιθυμητο αποτέλεσμα.

4.3 Υλοποίηση

4.3.1 XAMPP

Το **XAMPP** είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα και ανεξαρτήτου πλατφόρμας το οποίο περιέχει το εξυπηρετητή ιστοσελίδων http Apache, την βάση δεδομένων MySQL και ένα διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl. Το XAMPP είναι ακρωνύμιο αναφερόμενο 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. Δεν έχει σημασία το λειτουργικό σύστημα και λειτουργεί σε Microsoft Windows, Linux, Solaris, and Mac OS X και χρησιμοποιείται ως πλατφόρμα για την σχεδίαση και ανάπτυξη ιστοσελίδων με τεχνολογίες όπως PHP, JSP και Servlets.

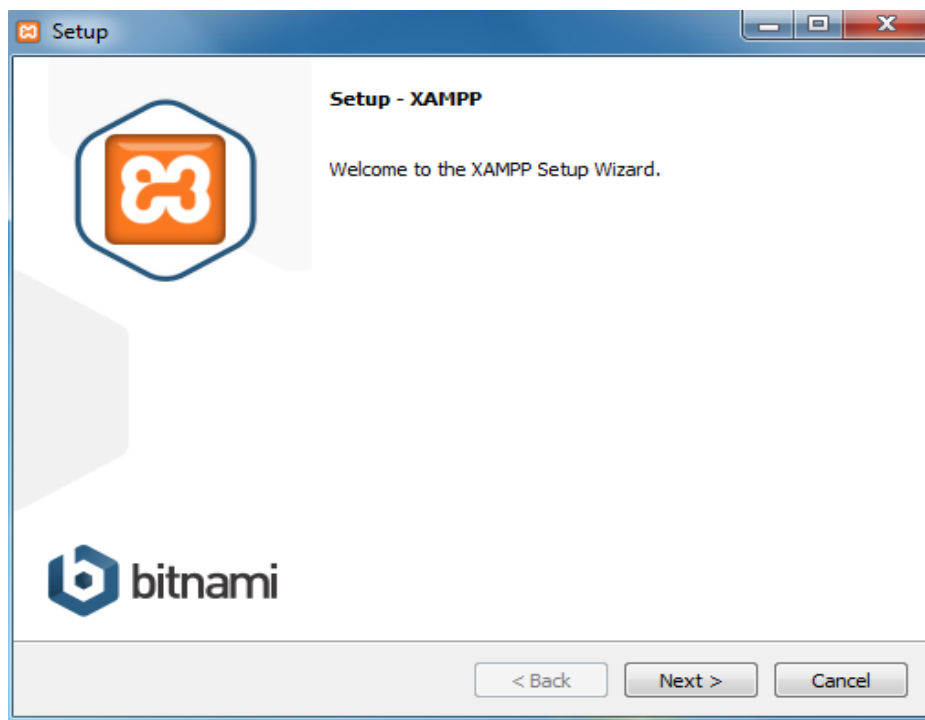
Το XAMPP προϋποθέτει να υπάρχουν λογισμικά συμπίεσης αρχείων όπως zip, tar, 7z ή exe κατά την διάρκεια της εγκατάστασης. Το XAMPP αναβαθμίζεται στις νέες εκδόσεις του εξυπηρετητή ιστοσελίδων http Apache, της βάσης δεδομένων MySQL, της γλώσσας PHP και Perl. Στο XAMPP συμπεριλαμβάνονται επίσης τα πακέτα OpenSSL και το phpMyAdmin.

Αρχικά το XAMPP προόριζονταν ως εργαλείο ανάπτυξης και δοκιμών ιστοσελίδων τοπικά στον υπολογιστή χωρίς να χρειάζεται η σύνδεση στο διαδίκτυο αλλά ορισμένες φορές χρησιμοποιείται και για την φιλοξενία ιστοσελίδων. Με το XAMPP επιτρέπει την δημιουργία και διαχείριση βάσεων δεδομένων MySQL και SQLite.

Μόλις εγκατασταθεί στον τοπικό υπολογιστή το XAMPP διαχειρίζεται τον localhost ως ένα απομακρυσμένο κόμβο, που υποστηρίζει και συνδέεται με το πρωτόκολλο μεταφοράς αρχείων FTP.

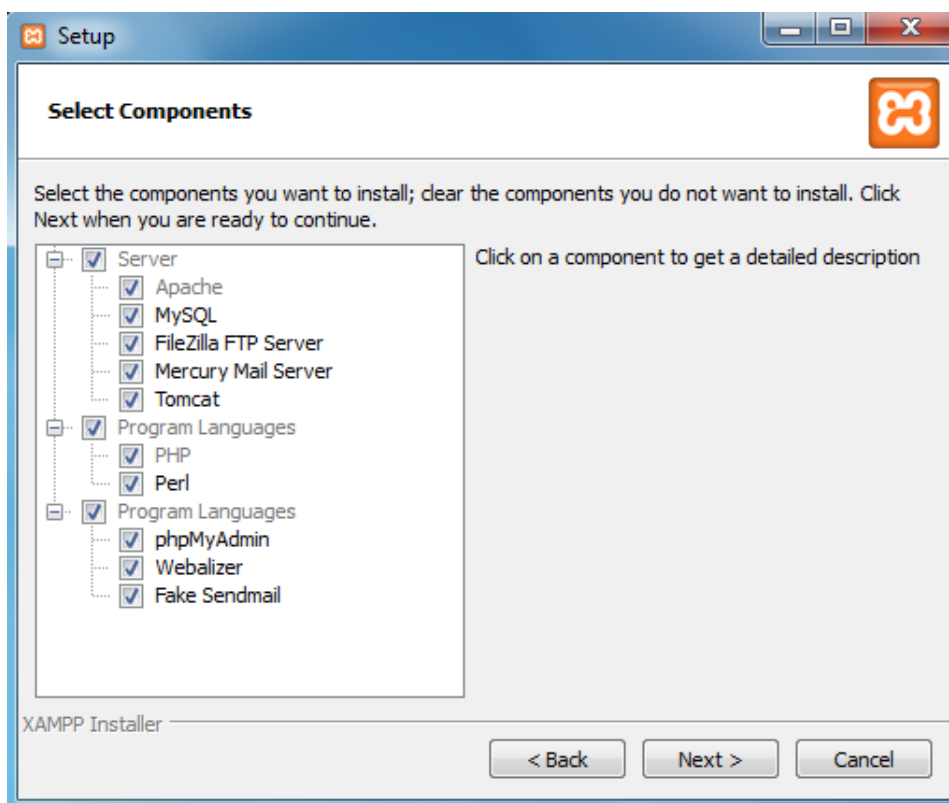
Η εγκατάσταση του xampp είναι απλούστατη. Πρώτα κατεβάζουμε το εκτελέσιμο αρχείο της τελευταίας έκδοσης του από τον εξουσιοδοτημένο ιστοτοπο του. Έπειτα το εγκαθιστούμε με τα εξής ακόλουθα βήματα:

Βήμα 1: Μόλις τρέξουμε το αρχείο που μόλις κατεβάσαμε θα εμφανιστεί η παρακάτω εικόνα και θα ξεκινήσει η διαδικασία της εγκατάστασης.



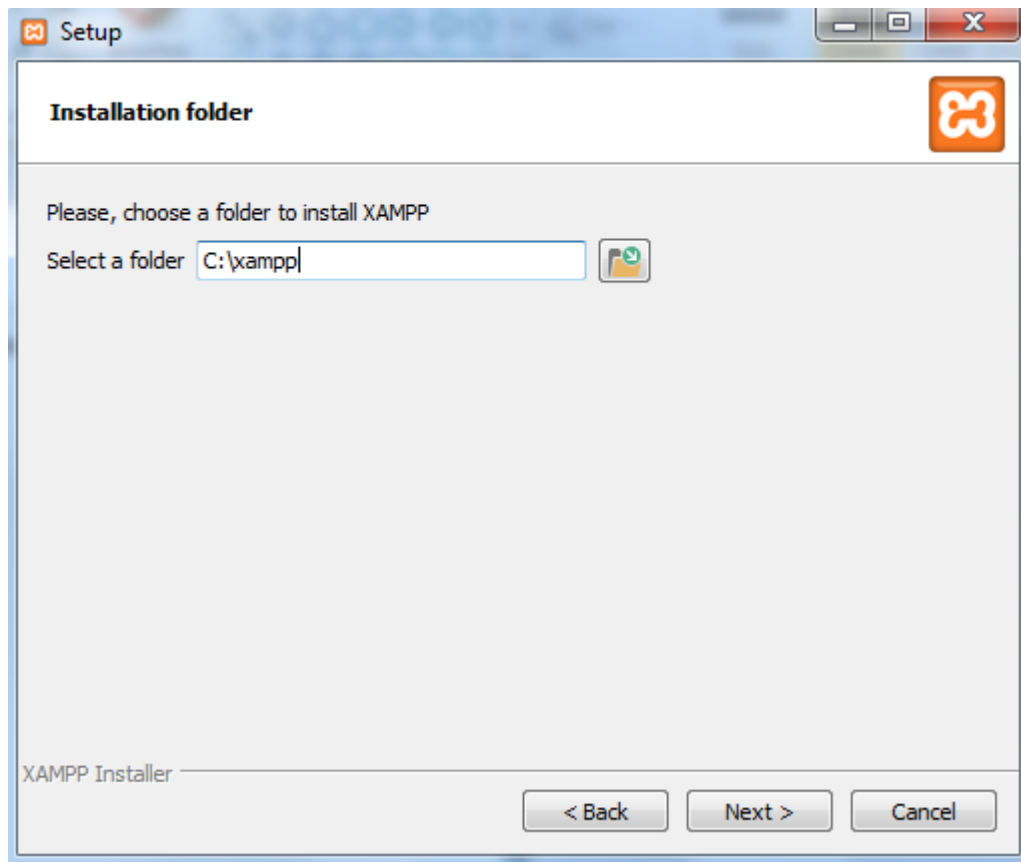
Εικόνα 23: Εγκατάσταση Xampp Βήμα 1

Βήμα 2: Διαλέγουμε όλα τα περιεχόμενα.



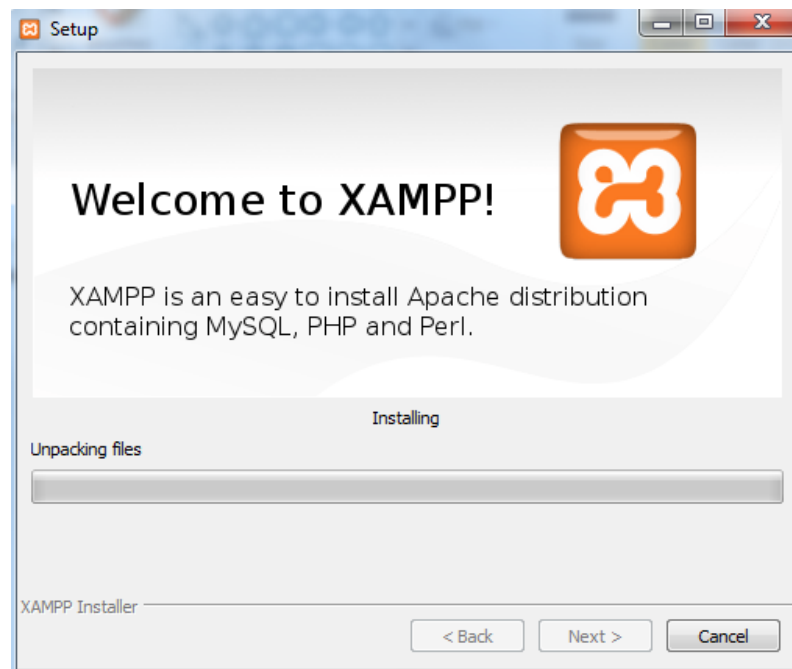
Εικόνα 24: Εγκατάσταση Xampp Βήμα 2

Βήμα 3: Ο φάκελος όπου θα εγκατασταθεί το xampp **δεν** πρέπει να είναι μέσα στο Program Files. Ο installer έχει προρυθμισμένο το φάκελο C:\xampp οπότε θα το εγκαταστήσουμε εκεί.



Εικόνα 25: Εγκατάσταση Xampp Βήμα 3

Βήμα 4: Η διαδικασία εγκατάστασης ξεκίνησε.



Εικόνα 26: Εγκατάσταση Xampp Βήμα 4

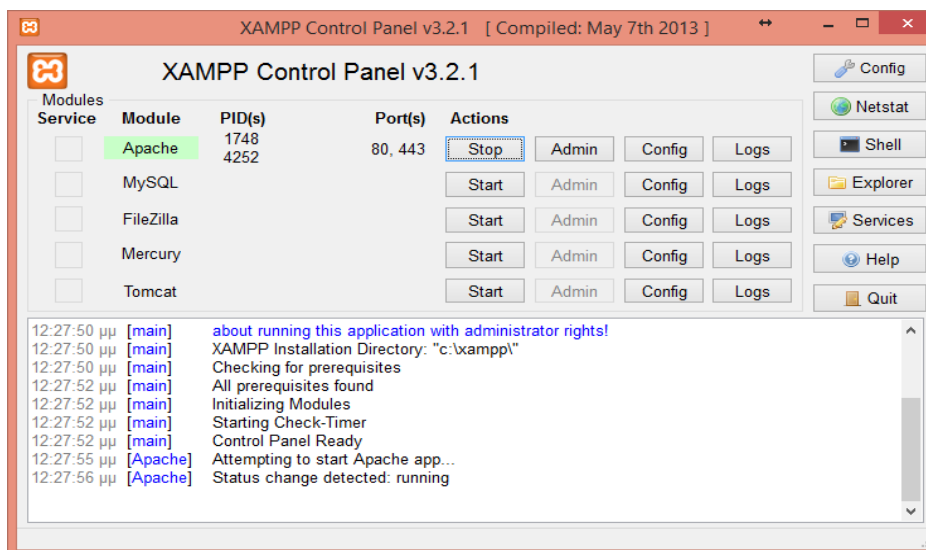
Βήμα 5: Η εγκατάσταση τελειώσε, πατάμε Finish για να ανοίξει το control panel του xampp.



Εικόνα 27: Εγκατάσταση Xampp Βήμα 5

Παρακάτω είναι το control panel του xampp με τον Apache να τρέχει. Για να ξεκινήσει μια υπηρεσία πατάμε Start, για να σταματήσει Stop. Εμείς εφόσον δεν θα χρησιμοποιήσουμε βάση δεδομένων δεν θα χρειαστεί να ενεργοποιήσουμε κάποια άλλη υπηρεσία του XAMPP³⁹.

³⁹ XAMPP Apache + MySQL + PHP + Perl, 2015, <https://www.apachefriends.org/index.HTML>



Εικόνα 28: Control Panel του XAMPP

4.3.2 GetSimple

GetSimple CMS είναι ένα σύστημα διαχείρισης περιεχομένου ανοιχτού κώδικα το οποίο είναι απλό, γρήγορο και εύκολο στη χρήση του. Είναι βασισμένο στις γλωσσές PHP και Javascript. Αντί να χρησιμοποιεί κάποια βάση δεδομένων για να αποθηκεύει πληροφορίες, το GetSimple αποθηκεύει όλα τα δεδομένα σε δομημένα αρχεία XML κ γι' αυτό ανήκει σε μια ομάδα εφαρμογών που μπορούν να τρέξουν χωρίς βάση δεδομένων. Η μνήμη του συστήματος είναι χαμηλή που αυτό το κάνει εύκολο εγκαταστασιμο. Η πρώτη έκδοση του ήταν τον Ιούλιο του 2009. Το όραμα ήταν να περιγράφεται ένας απλος τρόπος να φτιαχτεί και να διατηρηθεί ένας ιστοτοπος χωρίς μεγάλη γνώση HTML και ένα καλο και φιλικό προς χρήστη περιβάλλον.

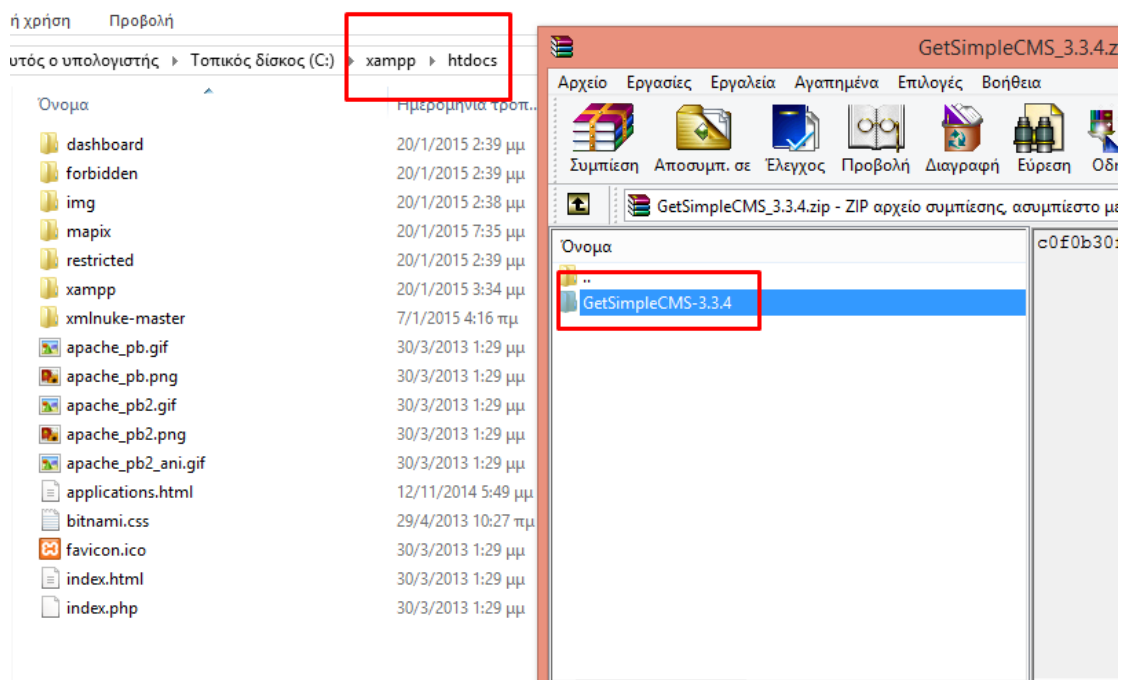
Το GetSimple αποδεικνύεται ότι είναι επεκτάσιμο μιας και πολλά σημαντικά χαρακτηριστικά μπορούν να προστεθούν σαν προσθετα (plugins) όπως είναι η πολυγλωσσική υποστήριξη, η υποστήριξη χρήστη, τα συστήματα σχολιασμού, γκαλερί φωτογραφιών, ιστολογία και αλληλεπίδραση με τα social media.

Χαρακτηριστικά που υπάρχουν στην βασική έκδοση είναι το ενσωματωμένο σύστημα για backup, η αναίρεση πράξεων στο περιβάλλον του administrator, επεξεργαστής κειμένου WYSIWYG και CKEditor, διαχείριση μενού ενός επιπέδου, επιλογή και επεξεργασία θεμάτων, επεξεργαστής επιπλέον μερών, διαχείριση προσθέτων και διαχείριση εικόνων και αρχείων.

Τα μειονεκτήματα του είναι ότι χρειάζεται να βάλεις προσθετα σε πολλές λειτουργίες μιας και η βασική έκδοση δεν τις υποστηρίζει. Γενικώς υπάρχει έλλειψη σε πολλές λειτουργίες προσθέτων όπως το ecommerce.

Εγκατάσταση GetSimple:

Κατεβάζουμε την τελευταία έκδοση GetSimple από το site <http://get-simple.info/>. Ανοίγουμε το αρχείο και μεταφέρουμε τον φάκελο που βλέπουμε μέσα στην περιοχή C:/Xampp/htdocs/ που είναι η ρίζα του localhost μας.



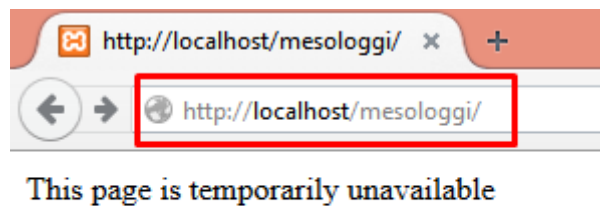
Εικόνα 29:Εγκατασταση GetSimple

Μετονομάζουμε τον φάκελο από GetSimpleCMS-3.3.4 σε mesologgi

Όνομα	Ημερομηνία τροπ...	Τύπος	Μέγεθος
dashboard	20/1/2015 2:39 μμ	Φάκελος αρχείων	
forbidden	20/1/2015 2:39 μμ	Φάκελος αρχείων	
img	20/1/2015 2:38 μμ	Φάκελος αρχείων	
mapix	20/1/2015 7:35 μμ	Φάκελος αρχείων	
mesologgi	8/10/2014 7:34 πμ	Φάκελος αρχείων	
restricted	20/1/2015 2:39 μμ	Φάκελος αρχείων	
xampp	20/1/2015 3:34 μμ	Φάκελος αρχείων	
xmlnuke-master	7/1/2015 4:16 πμ	Φάκελος αρχείων	
apache_pb.gif	30/3/2013 1:29 μμ	Εικόνα GIF	3 KB
apache_pb.png	30/3/2013 1:29 μμ	Εικόνα PNG	2 KB
apache_pb2.gif	30/3/2013 1:29 μμ	Εικόνα GIF	3 KB
apache_pb2.png	30/3/2013 1:29 μμ	Εικόνα PNG	2 KB
apache_pb2_ani.gif	30/3/2013 1:29 μμ	Εικόνα GIF	3 KB
applications.html	12/11/2014 5:49 μμ	Αρχείο HTML	2 KB
bitnami.css	29/4/2013 10:27 πμ	Αρχείο CSS	3 KB
favicon.ico	30/3/2013 1:29 μμ	Αρχείο ICO	8 KB
index.html	30/3/2013 1:29 μμ	Αρχείο HTML	1 KB
index.php	30/3/2013 1:29 μμ	Αρχείο PHP	1 KB

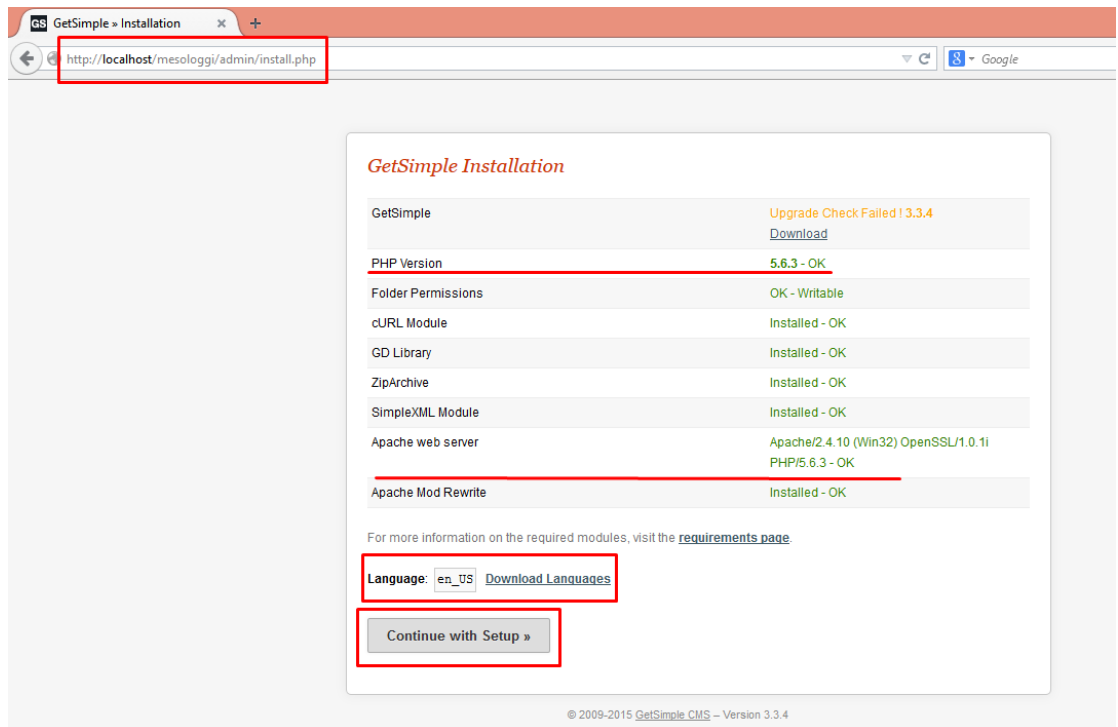
Εικόνα 30:Εγκατασταση GetSimple

Έτσι ανοίγουμε ένα browser και γράφοντας την διεύθυνση του φακέλου μέσω του localhost μας δίνει το εξής αποτέλεσμα:



Εικόνα 31:Εγκατασταση GetSimple

Αυτό σημαίνει ότι πρέπει να εγκαταστήσουμε το GetSimple μέσα από το περιβάλλον του διαχειριστή. Οπότε βάζουμε στον browser μας την διεύθυνση localhost/mesologgi/admin/install.php που είναι το αρχείο που περιέχει την αρχή της εγκατάστασης του GetSimple.

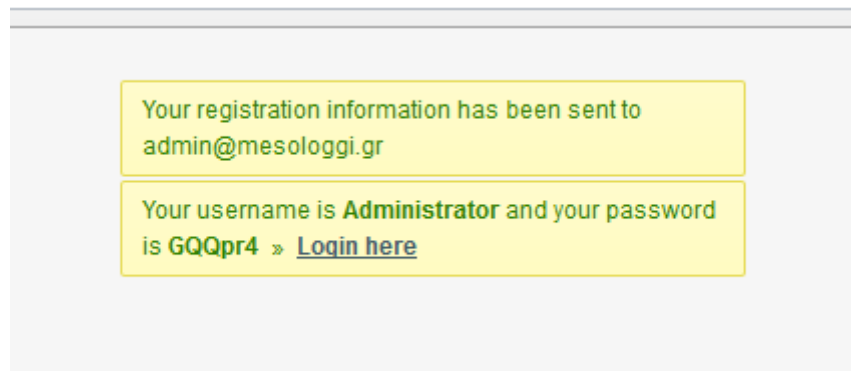


Εικόνα 32:Εγκατασταση GetSimple

Το πρώτο πράγμα που έχουμε να κάνουμε είναι να δηλώσουμε το όνομα του site, το όνομα του διαχειριστή και ένα email επικοινωνίας. Εφόσον δεν θέλουμε να ανεβάσουμε τον ιστοτοπο στο internet αλλά μόνο θα το έχουμε να τρέχει τοπικά στον υπολογιστή μας και δεν υπάρχει πιθανότητα επικοινωνίας με κάποιον επισκέπτη αρχικά δίνουμε ένα μη έγκυρο email.

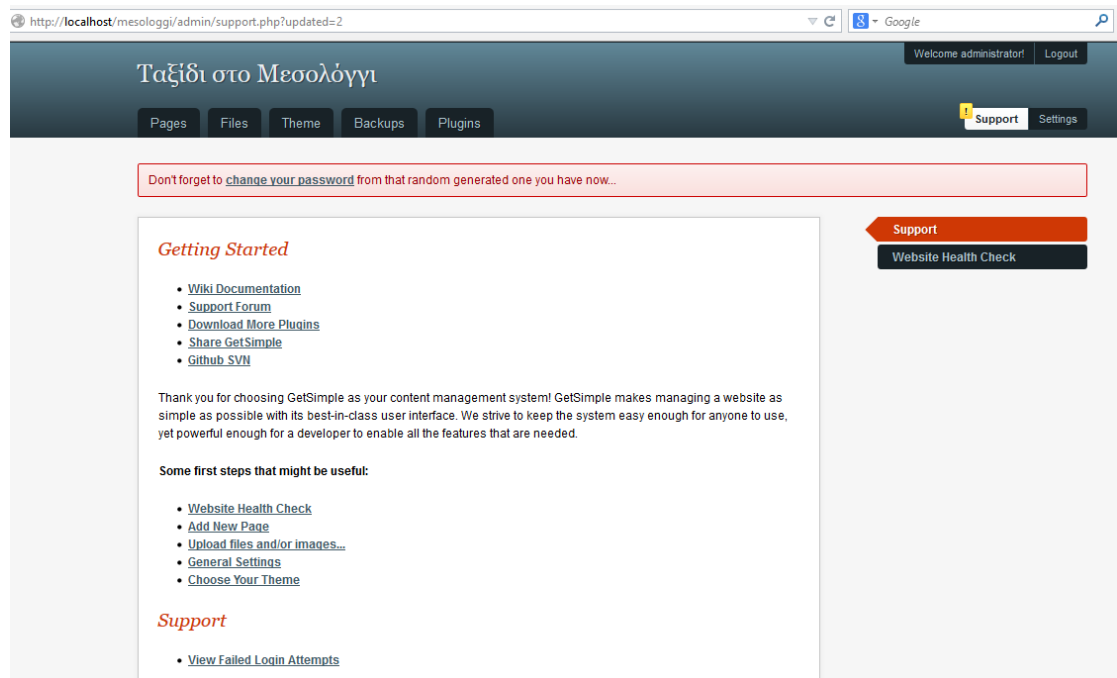
Εικόνα 33:Εγκατασταση GetSimple

Το επόμενο είναι ένα μήνυμα επιτυχής ολοκλήρωσης και ο προσωρινος κωδικός πρόσβασης του διαχειριστή



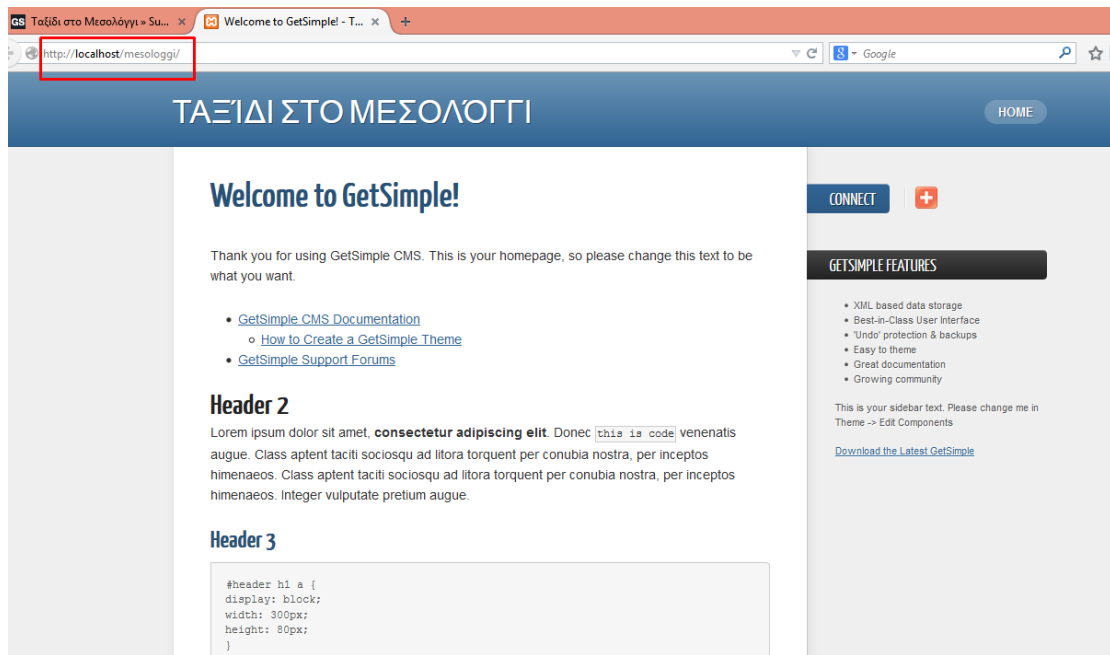
Εικόνα 34:Εγκατασταση GetSimple

Έπειτα με ανανέωση του browser φαίνεται το περιβάλλον του διαχειριστή το οποίο λέγεται και backend.



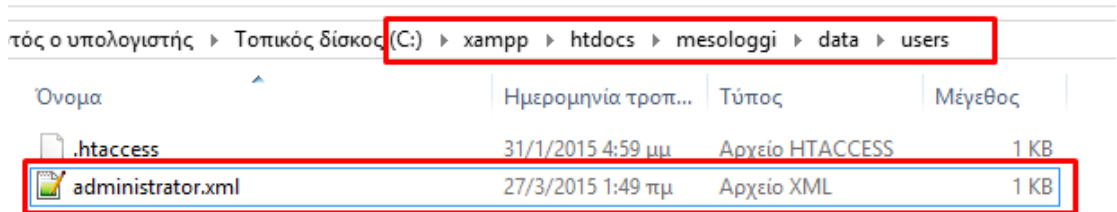
Εικόνα 35: GetSimple Backend

Μετα που αλλάξαμε κωδικό για τον διαχειριστή πληκτρολογούμε ξανά στον browser μας την διεύθυνση του site δηλαδή localhost/mesologgi οπου τώρα βλέπουμε το site με τα αρχικά περιεχόμενα του GetSimple. Αυτό θα βλέπει ο επισκέπτης όταν πληκτρολογεί την διεύθυνση στον browser αν μελλοντικά ανεβεί στο internet και αυτό το λεμέ frontend.

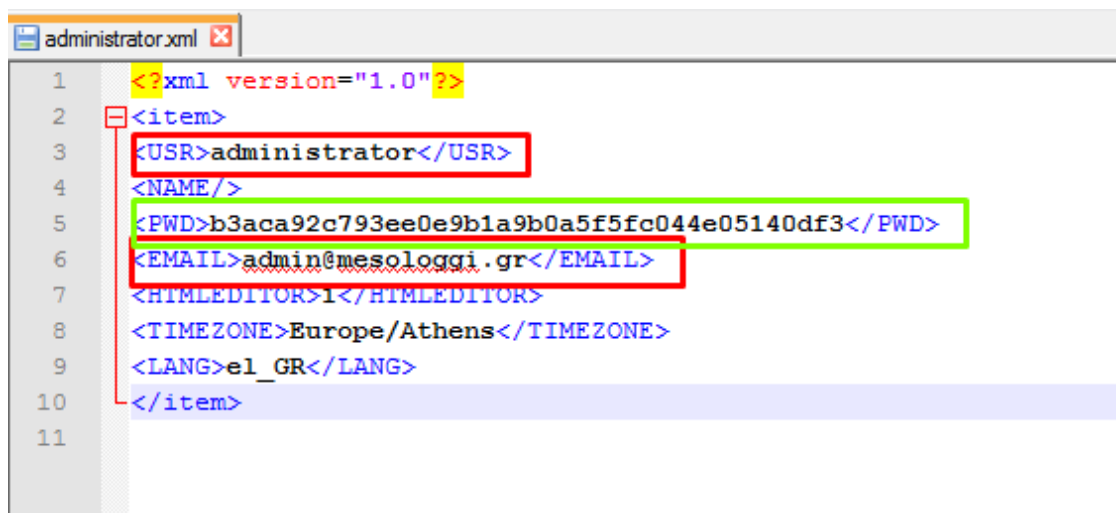


Εικόνα 36: GetSimple Frontend

Τα δεδομένα του διαχειριστή που μόλις φτιάξαμε (όνομα χρήστη, κωδικός και email) αποθηκεύτηκαν σε ένα αρχείο XML στον φάκελο C:\xampp\htdocs\mesologgi\data\users.



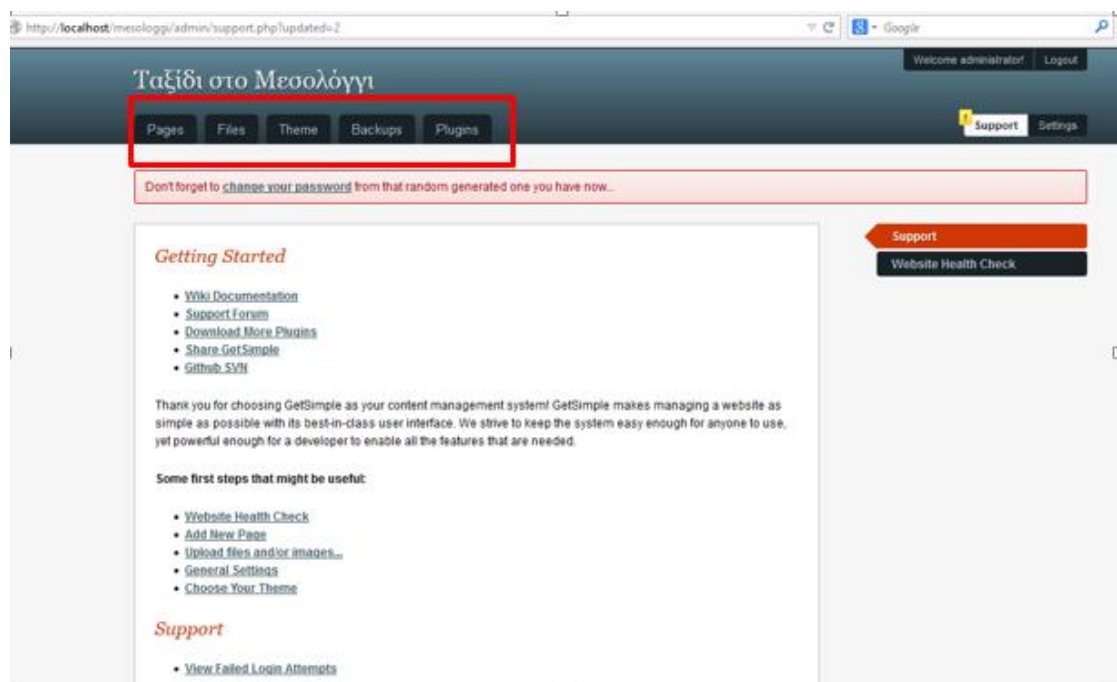
Εικόνα 37: C:\xampp\htdocs\mesologgi\data\users



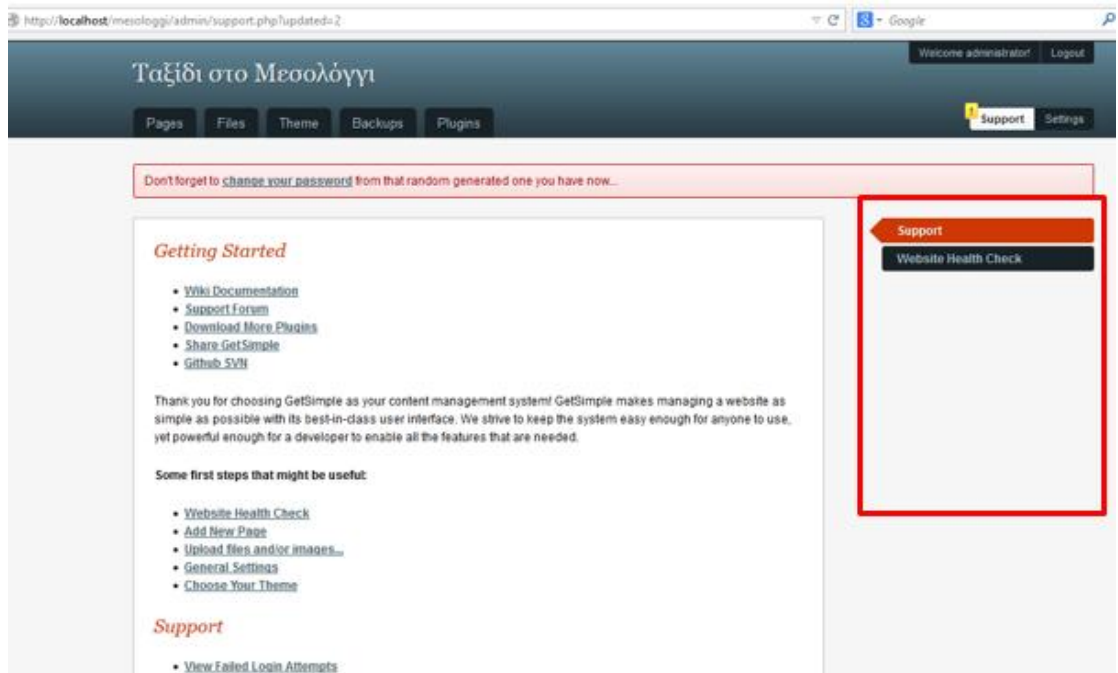
Εικόνα 38: Administrator.xml

Μόλις ανοίξεις το αρχείο καταλαβαίνεις ποιο στοιχείο αντιστοιχεί σε τι. Είναι προφανές ότι τα tags `<USR>administrator</USR>`, `<PWD>b3aca92c793ee0e9b1a9b0a5f5fc044e05140df3</PWD>` και `<EMAIL>admin@mesologgi.gr</EMAIL>` αντιστοιχούν στο όνομα χρήστη, στον κωδικό και στο email. Το όνομα και το email είναι όπως τα δηλώσαμε. Το μοναδικό που έχει αλλάξει είναι ο κωδικός που λόγω ασφαλείας τον δείχνει κρυπτογραφημένο με τον αλγόριθμο SHA-1.

Στο backend βλέπουμε 5 tabs τα οποία θα χρησιμοποιήσουμε. Σε κάθε tab υπάρχει ένα μενού δεξιά το οποίο ανάλογα με το tab που έχει επιλεγεί έχει διαφορές ιδιότητες.

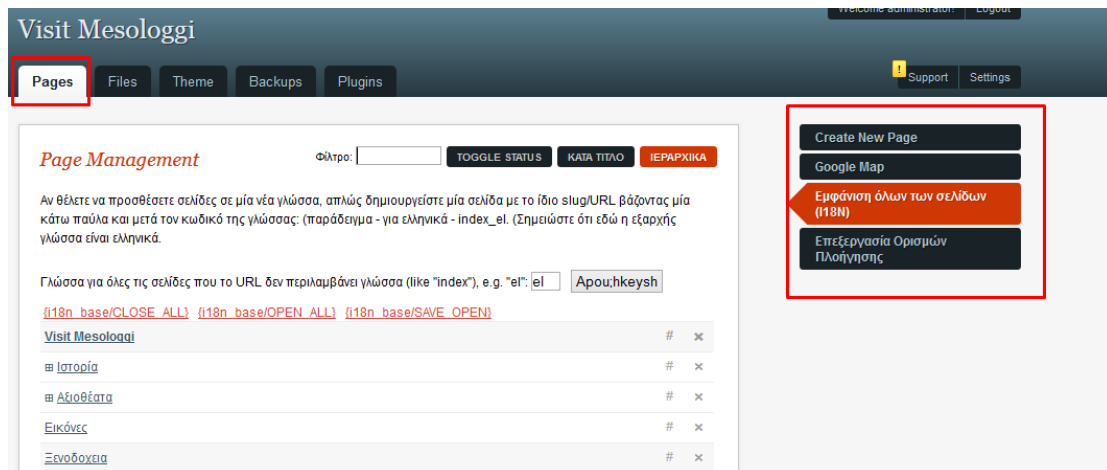


Εικόνα 39: GetSimple Backend Tabs



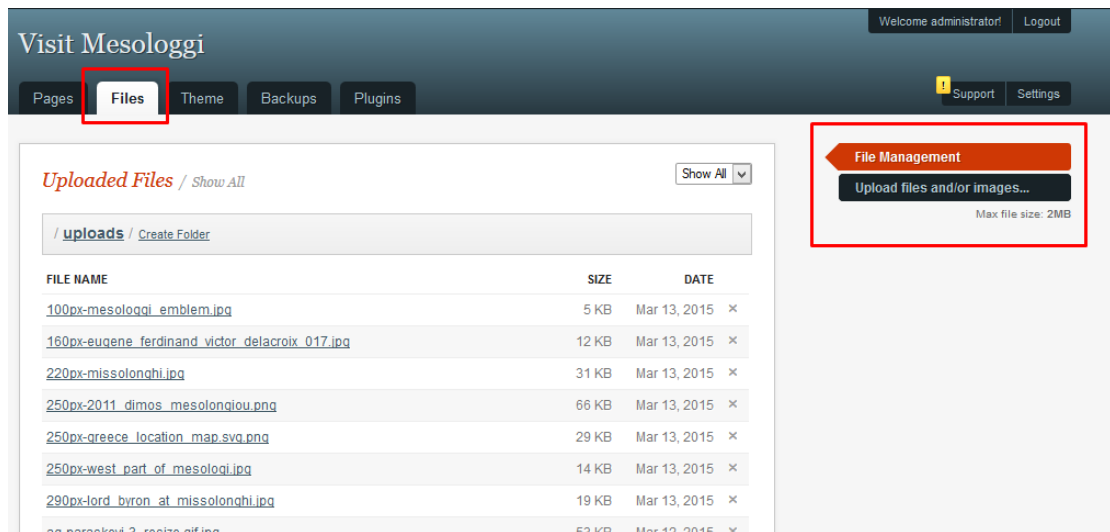
Εικόνα 40: GetSimple Backend Side Menu

- Το pages που εκεί γράφουμε τα άρθρα



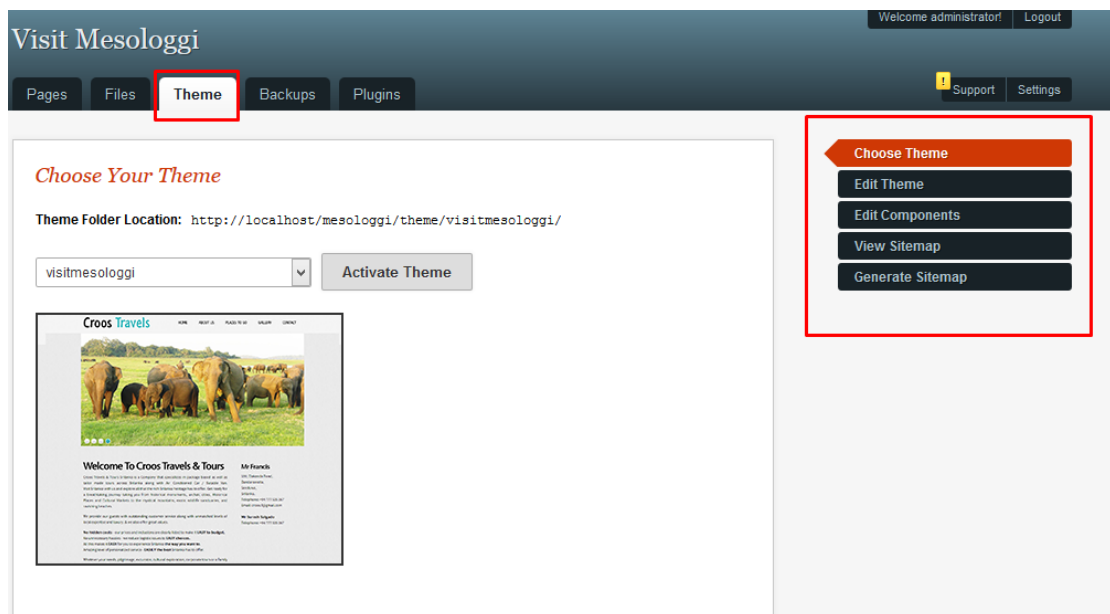
Εικόνα 41: GetSimple Backend Pages Tab

- Το files που εκεί ανεβάζουμε ότι αρχεία θέλουμε όπως πχ φωτογραφίες



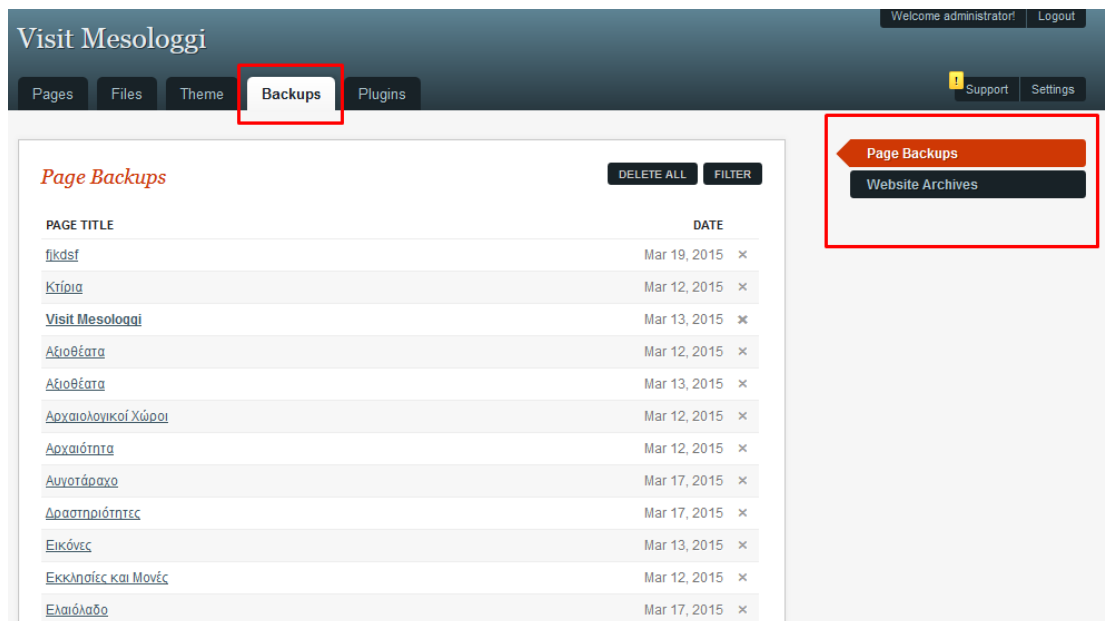
Εικόνα 42: GetSimple Backend Files Tab

- Το theme που εκεί αλλάζουμε και επεξεργαζόμαστε τα θέματα των ιστοτοπων μας



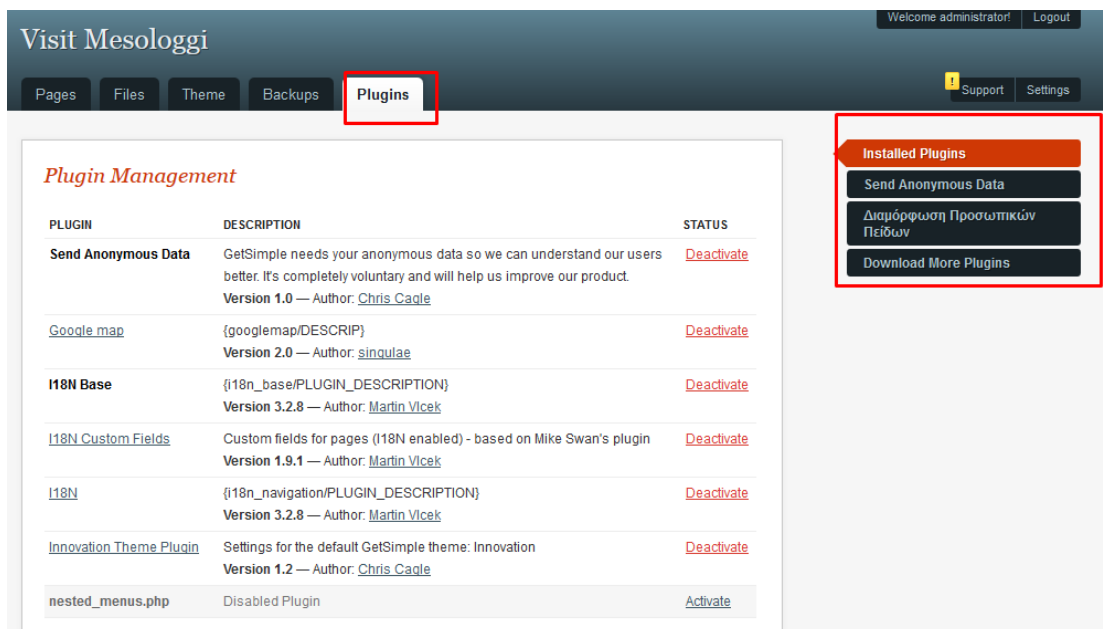
Εικόνα 43: GetSimple Backend Theme Tab

- Το backups που βλέπουμε την προηγούμενη μορφή των σελίδων και των αρχείων μας.



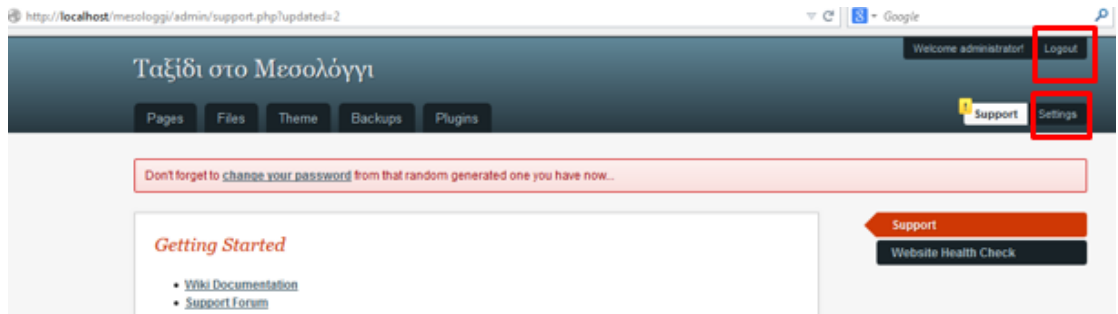
Εικόνα 44: GetSimple Backend Backups Tab

- Το plugins που εγκαθιστούμε κ διαχειριζόμαστε τα προσθετα μας.

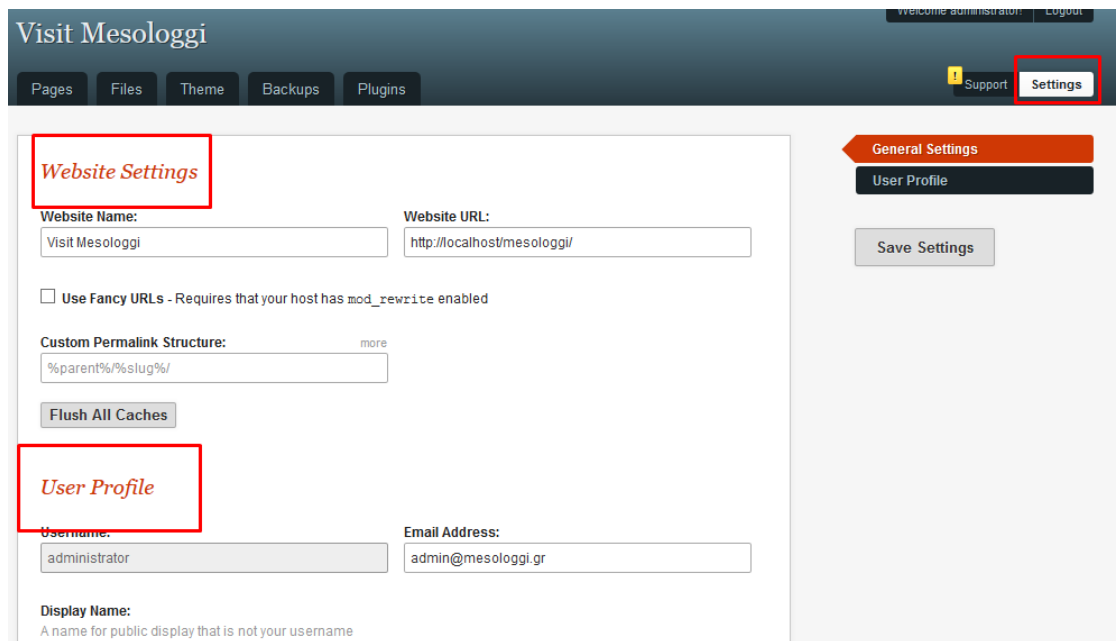


Εικόνα 45: GetSimple Backend Plugins Tab

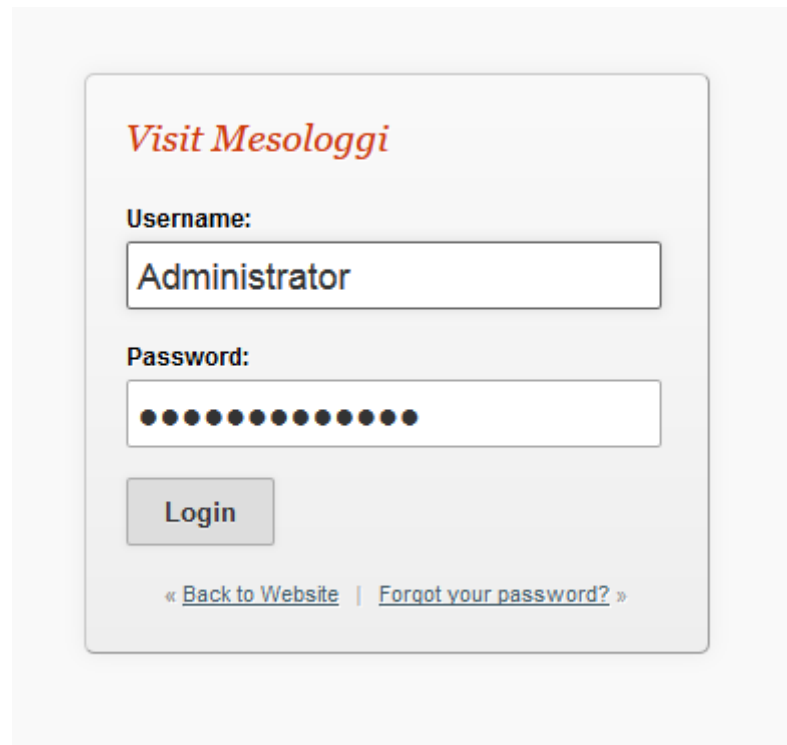
Επίσης είναι το κουμπι logout με το οποίο βγαίνεις από το backend και το κουμπι settings που μπορείς να αλλάξεις γενικά πράγματα στο site όπως το όνομα του, τον κωδικό του διαχειριστή κ.ο.κ



Εικόνα 46: GetSimple Backend Logout & Settings



Εικόνα 47: GetSimple Backend Settings

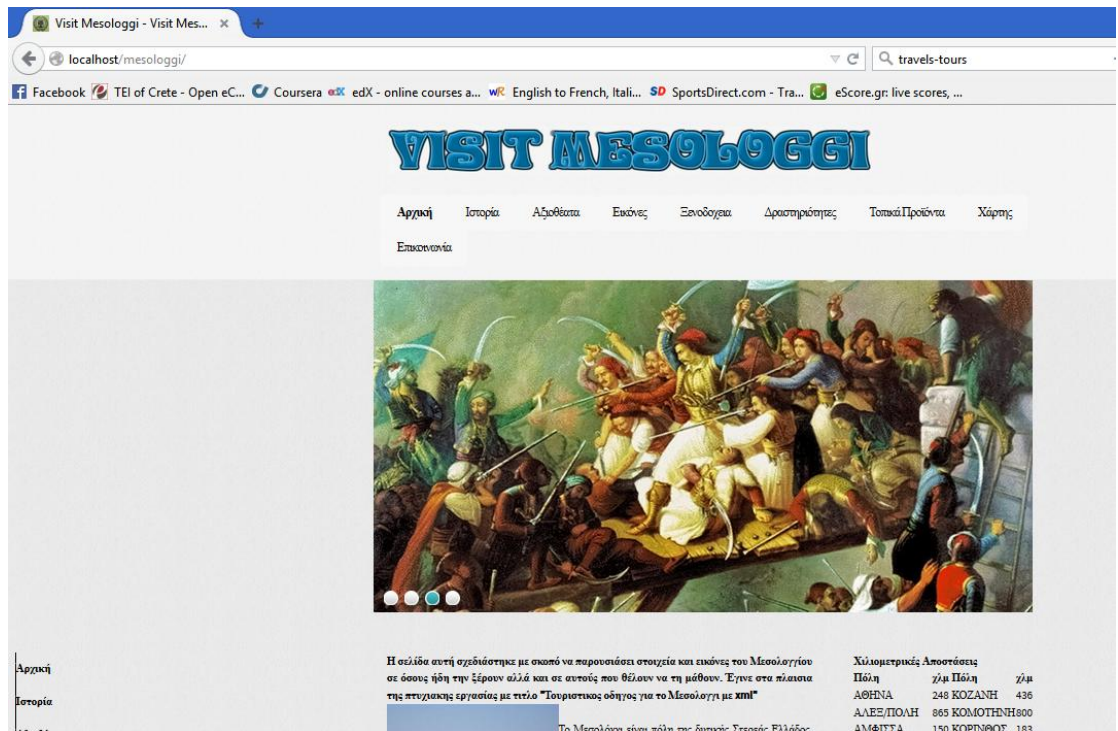


Εικόνα 48: GetSimple Backend Logout

Επιλέξαμε σαν αρχικο theme το Travels and Tours από τα themes του GetSimple το οποίο και το αλλάξαμε σχεδόν ολο μέσω της CSS.



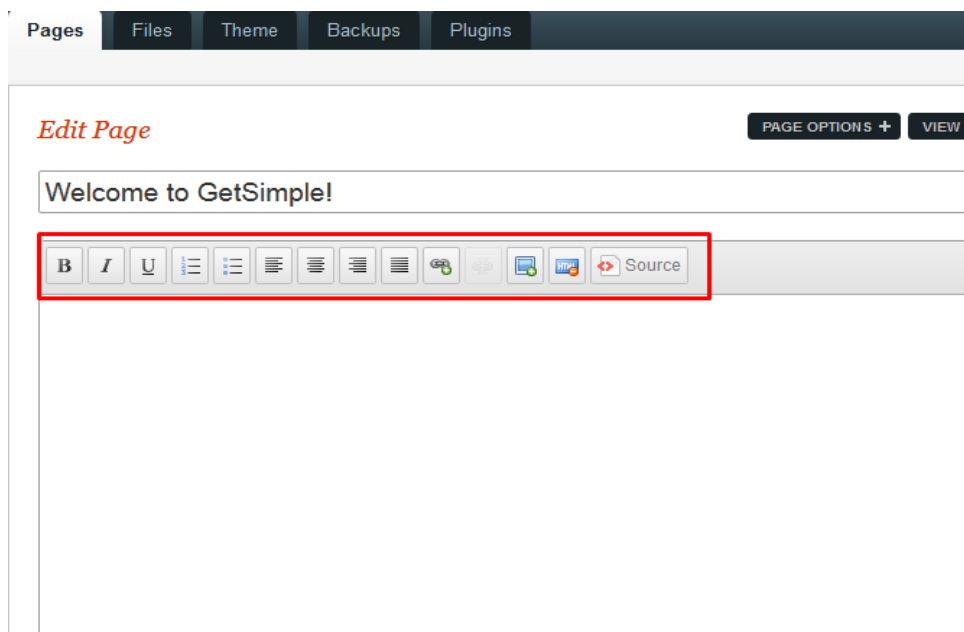
Εικόνα 49: GetSimple Theme Travels and Tours



Εικόνα 50: Το site όπως είναι στην τελική του μορφή

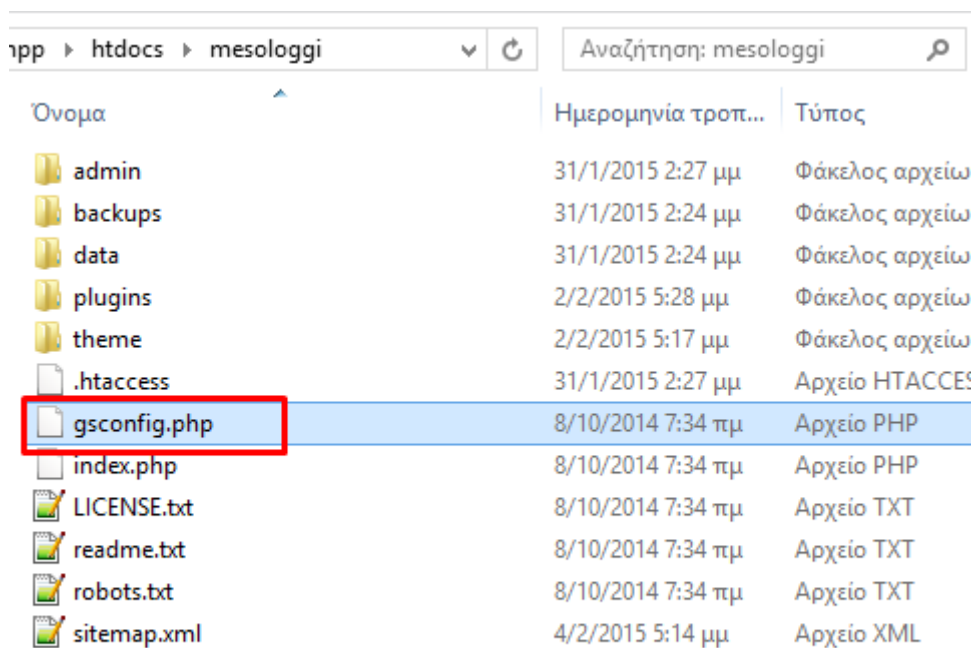
Ένα από τα χαρακτηριστικά του GetSimple είναι ότι μπορούμε να βάλουμε ένα άλλο editor για να γράφουμε τα άρθρα μας στις σελίδες. Αυτό μπορούμε να το κάνουμε αλλάζοντας τις παραμέτρους σε ένα από τα πηγαία αρχεία. Μιας κ το GetSimple είναι ανοιχτού κώδικα έχουμε το δικαίωμα να το πειράξουμε.

Default editor έχουμε τον CKeditor



Εικόνα 51: CKeditor

Πηγαίνουμε στον φάκελο που είναι εγκατεστημένο το site μας και βρίσκουμε το αρχείο gconfig.php



Όνομα	Ημερομηνία τροπο...	Τύπος
admin	31/1/2015 2:27 μμ	Φάκελος αρχείω
backups	31/1/2015 2:24 μμ	Φάκελος αρχείω
data	31/1/2015 2:24 μμ	Φάκελος αρχείω
plugins	2/2/2015 5:28 μμ	Φάκελος αρχείω
theme	2/2/2015 5:17 μμ	Φάκελος αρχείω
.htaccess	31/1/2015 2:27 μμ	Αρχείο HTACCE!
gconfig.php	8/10/2014 7:34 πμ	Αρχείο PHP
index.php	8/10/2014 7:34 πμ	Αρχείο PHP
LICENSE.txt	8/10/2014 7:34 πμ	Αρχείο TXT
readme.txt	8/10/2014 7:34 πμ	Αρχείο TXT
robots.txt	8/10/2014 7:34 πμ	Αρχείο TXT
sitemap.xml	4/2/2015 5:14 μμ	Αρχείο XML

Εικόνα 52:Βρίσκουμε το gconfig.php

Βρίσκουμε την γραμμή που είναι ο WYSIWYG editor

```
0  
1 # WYSIWYG toolbars (advanced, basic or [custom config])  
2 #define('GSEEDITOROOL', 'advanced');  
3
```

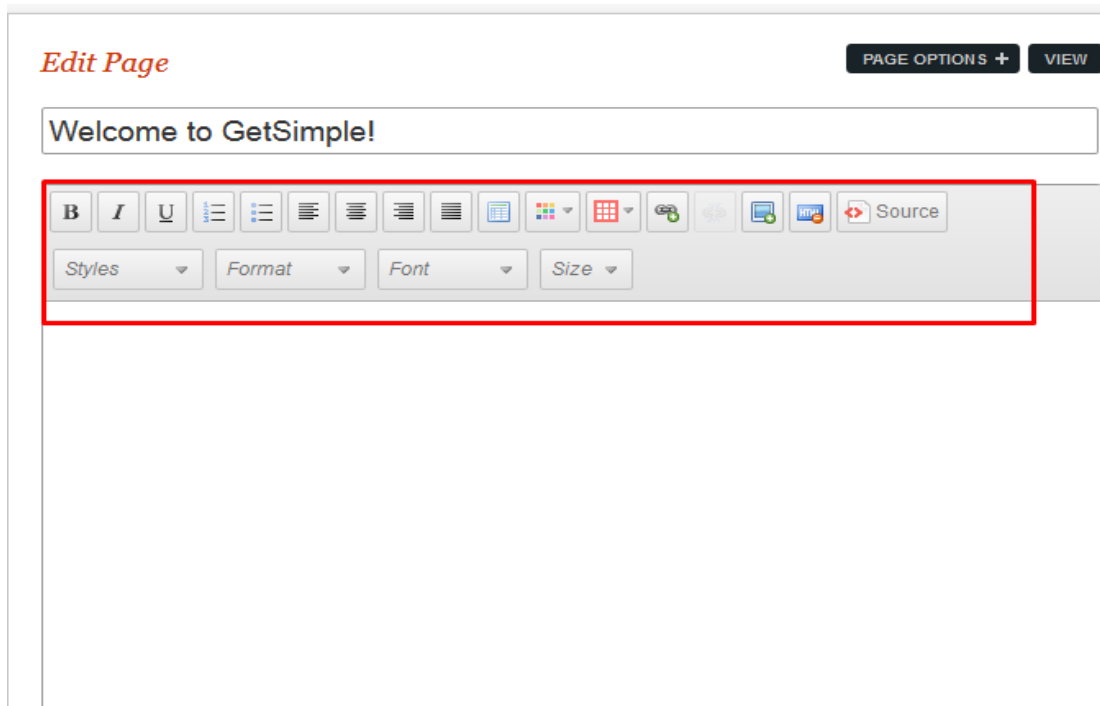
Εικόνα 53: Βρίσκουμε την γραμμή που είναι ο WYSIWYG editor

Και τον ενεργοποιούμε βγάζοντας τον από τα σχόλια

```
# WYSIWYG toolbars (advanced, basic or [custom config])  
define('GSEEDITOROOL', 'advanced');
```

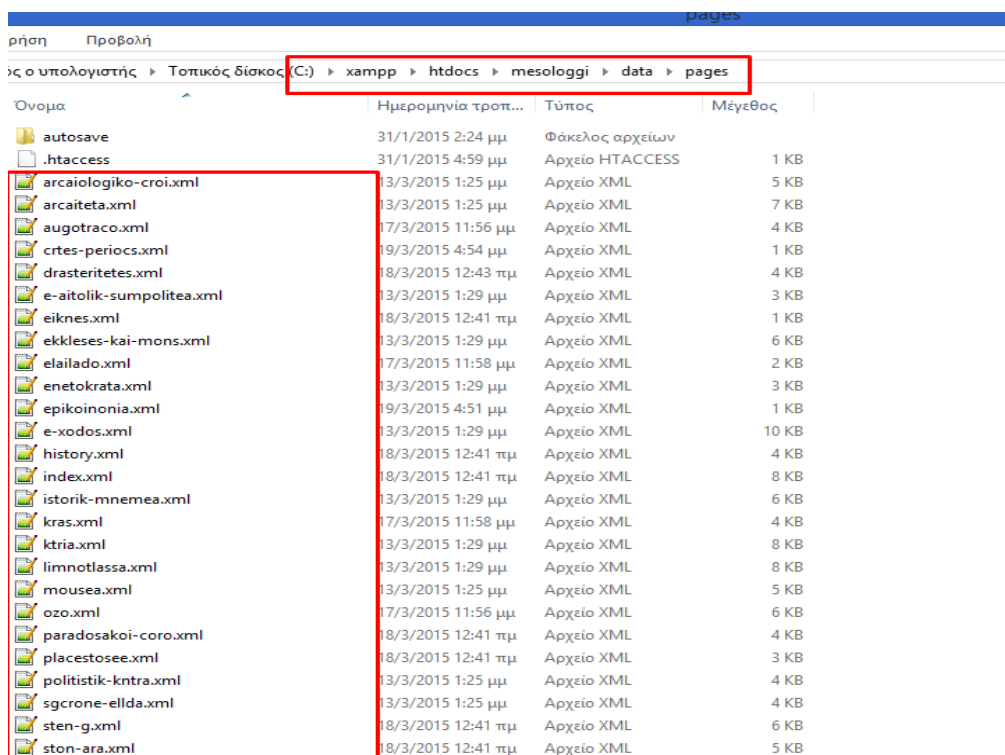
Εικόνα 54:Και τον ενεργοποιούμε βγάζοντας τον από τα σχόλια

Έτσι έχουμε τον WYSIWYG editor ο οποίος έχει περισσότερες επιλογές και μπορούμε να γράφουμε πιο πλούσια κείμενα και πιο εύκολα.

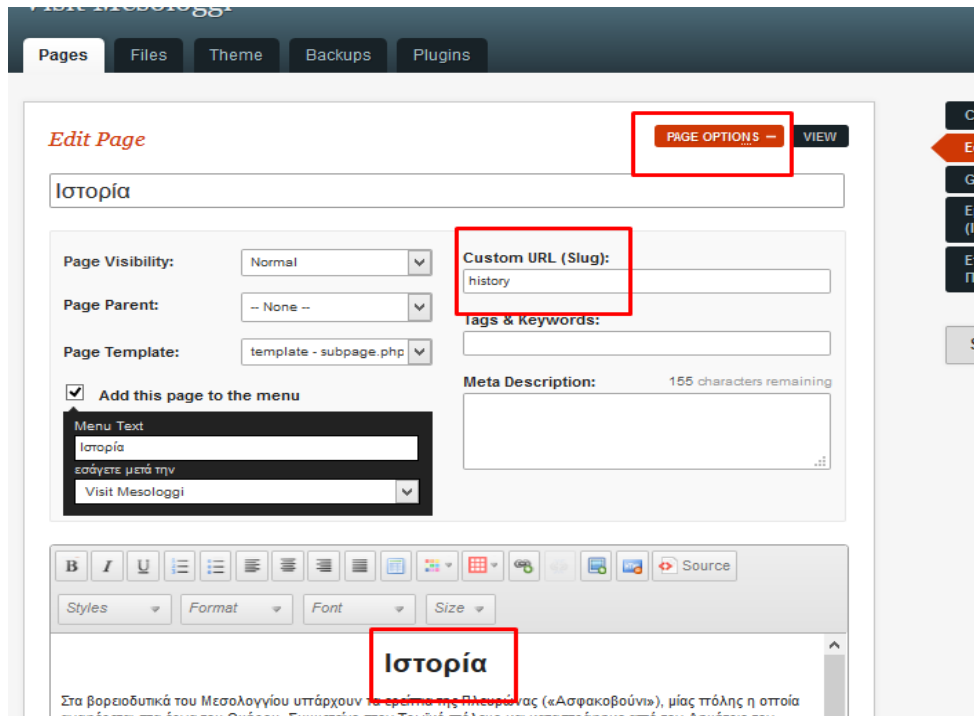


Εικόνα 55:WYSIWYG editor

Αναφέραμε πολλές φορές ότι τα δεδομένα είναι σε μορφή XML. Οπότε αν θέλουμε να δούμε τα δεδομένα σαν αρχεία θα πάμε στον φάκελο C:\xampp\htdocs\mesologgi\data\pages και θα δούμε ότι όλες οι σελίδες έχουν γραφεί σε μορφή XML συγκεκριμένα το όνομα που έχει δοθεί στο πεδίο custom url στο page options.

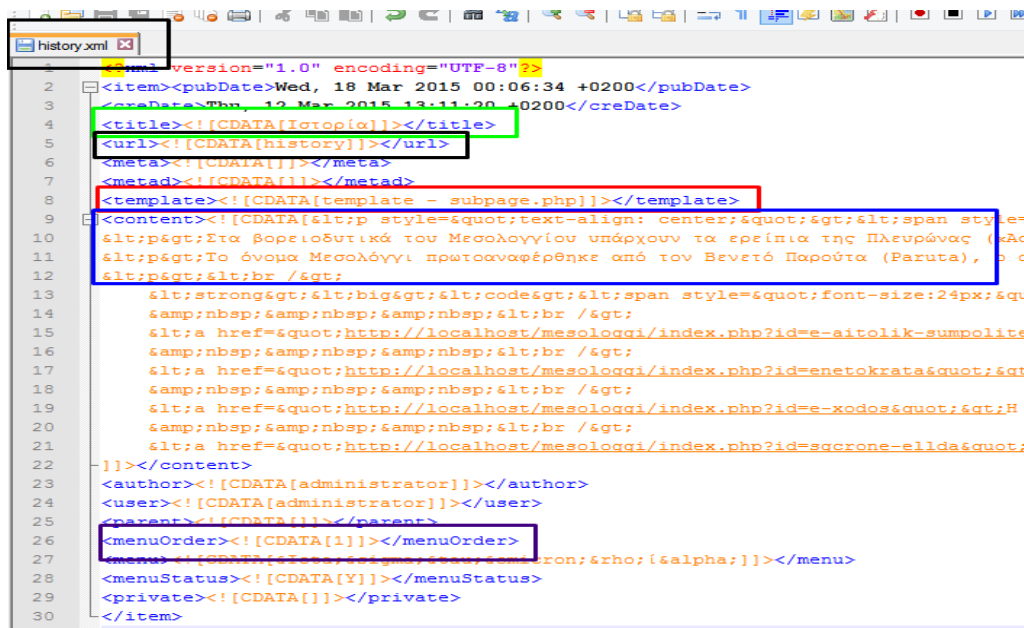


Εικόνα 56:C:\xampp\htdocs\mesologgi\data\pages

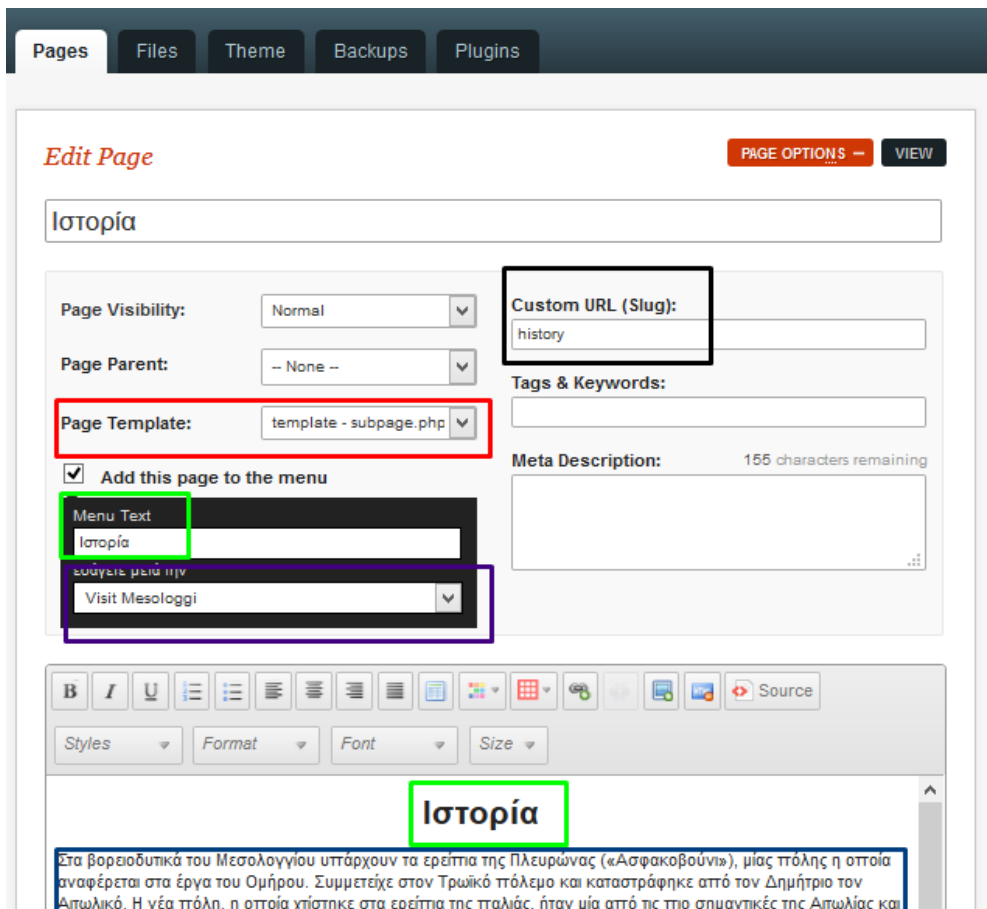


Εικόνα 57:Backend Pages-Ιστορία

Αν ανοίξουμε π.χ. το αρχείο history.xml θα δούμε πως έχουν αναπαρασταθεί τα δεδομένα στη μορφή xml όπως ακριβώς τα έχουμε ορίσει στο backend. Τα tags επίσης διακρίνονται εύκολα ποια στοιχεία περιγράφουν. Έπειτα με την επεξεργασία με CSS έχουν εμφανιστεί στο frontend με την τελική μορφή που θα τα δει ο επισκέπτης. Στις επόμενες τρεις φωτογραφίες έχουν επισημανθεί αντιστοιχηθεί με χρώματα που φαίνονται τα δεδομένα στο xml, πως έχουν δηλωθεί στο backend και την τελική μορφή του frontend.



Εικόνα 58:History.xml



Εικόνα 59:Backend Pages-Ιστορία



Εικόνα 60:Frontend Pages-Ιστορία

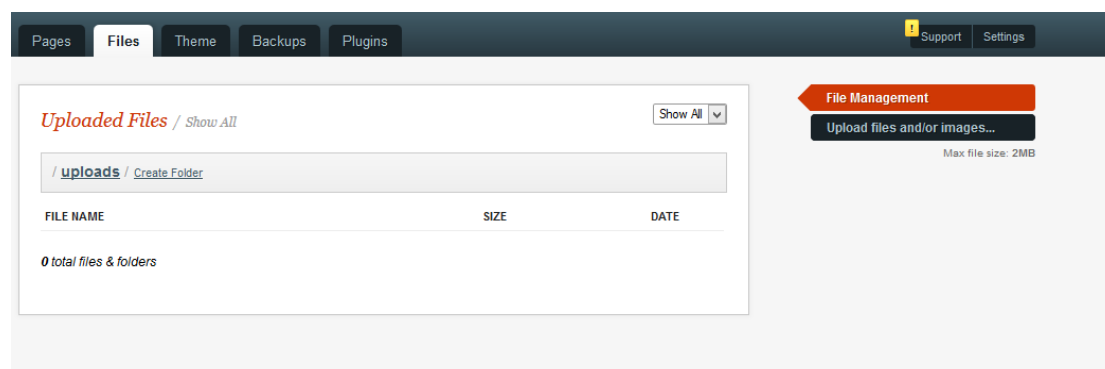
Γενικά το σύνολο των σελίδων και η παρουσίαση τους στο μενού περιγράφονται από ένα άλλο αρχείο xml το pages.xml που βρίσκεται στο C:\xampp\htdocs\mesologgi\data\other

Στο συγκεκριμένο παράδειγμα πήραμε το στοιχείο «κρασί», όπου μας εμφανίζει τα δεδομένα που θα χρησιμοποιήσει το μενού γι' αυτή τη σελίδα.

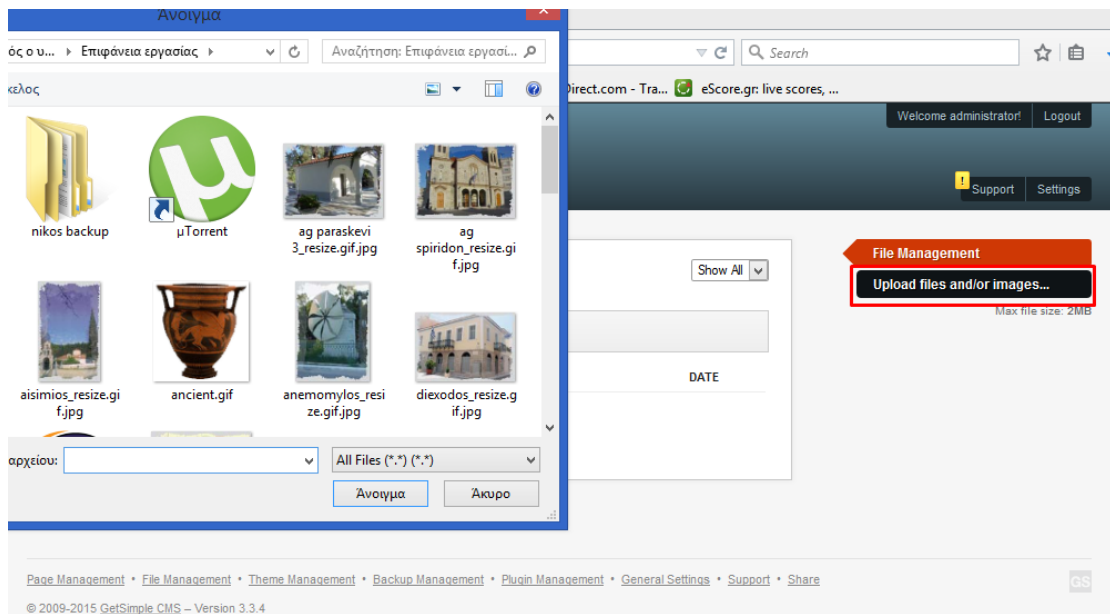
```
<item><pubDate><![CDATA[Tue, 17 Mar 2015 23:56:16 +0200]]></pubDate>
<createDate><![CDATA[Tue, 17 Mar 2015 23:50:01 +0200]]></createDate>
<title><![CDATA[Κρασί]]></title><url><![CDATA[kras]]></url>
<meta><![CDATA[]]></meta><metad><![CDATA[]]></metad>
<template><![CDATA[template - subpage.php]]></template>
<author><![CDATA[administrator]]></author>
<user><![CDATA[administrator]]></user>
<parent><![CDATA[topik-pronta]]></parent>
<menuOrder><![CDATA[3]]></menuOrder>
<menu><![CDATA[&Kappa;&rho;&alpha;&sigma;]]></menu>
<menuStatus><![CDATA[Y]]></menuStatus>
<private><![CDATA[]]></private>
<slug><![CDATA[kras]]></slug>
<filename><![CDATA[kras.xml]]></filename>
</item>
```

Εικόνα 61:pages.xml

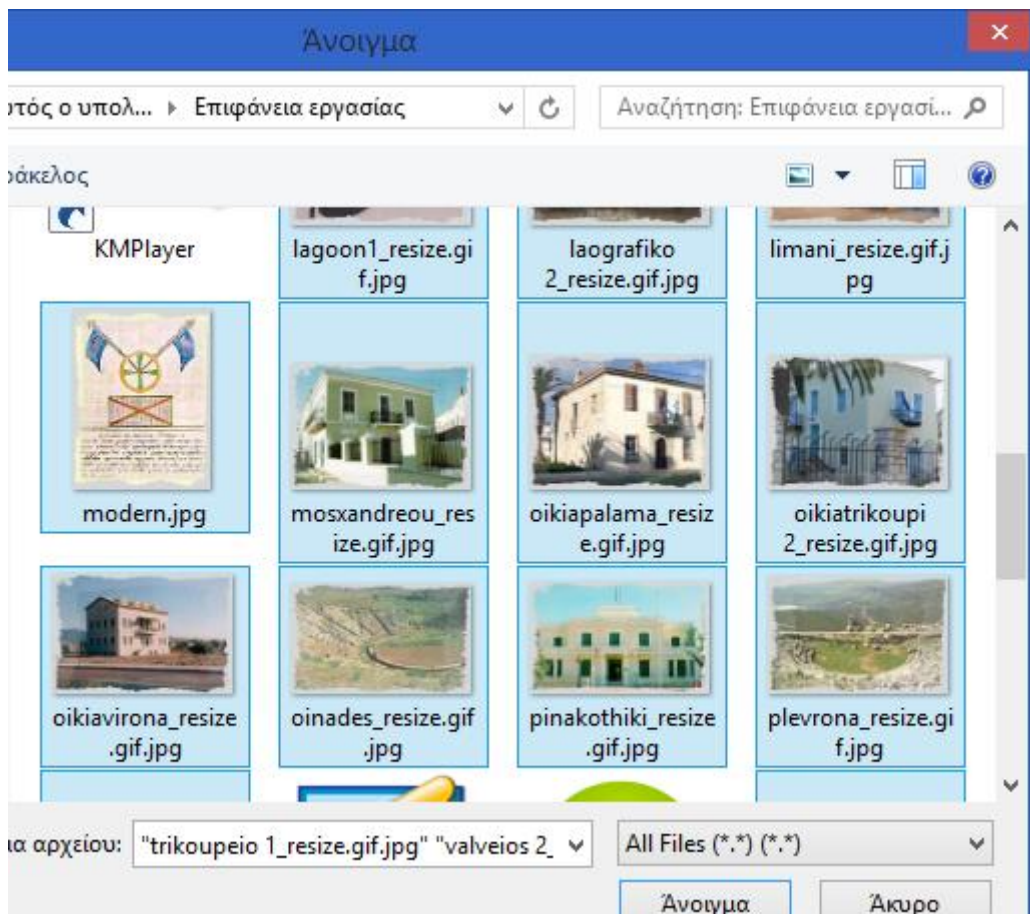
Μπορούμε να προσθέσουμε αρχεία κ εικόνες στο site κ από κει αμέσως στις σελίδες μας.



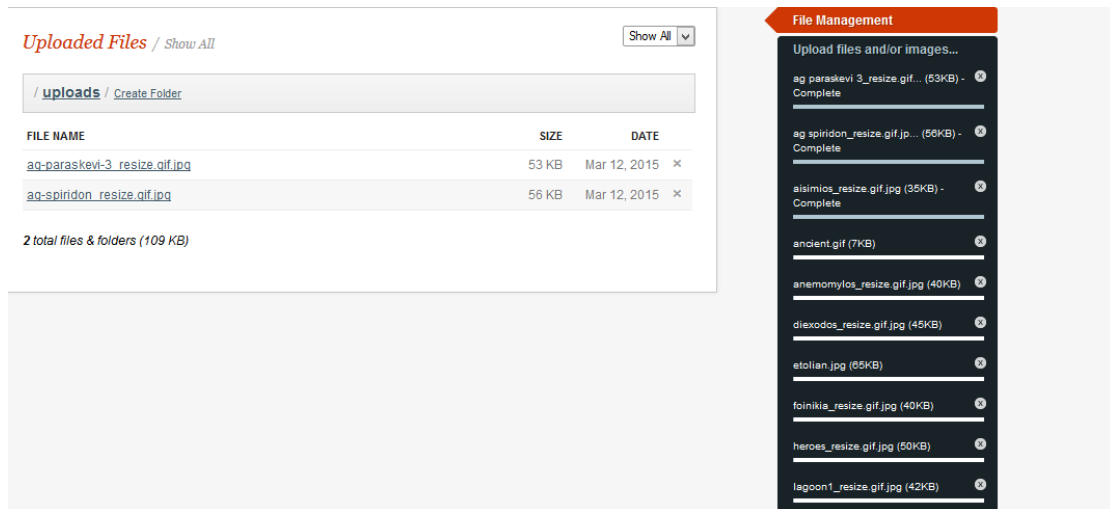
Εικόνα 62:Backend uploads



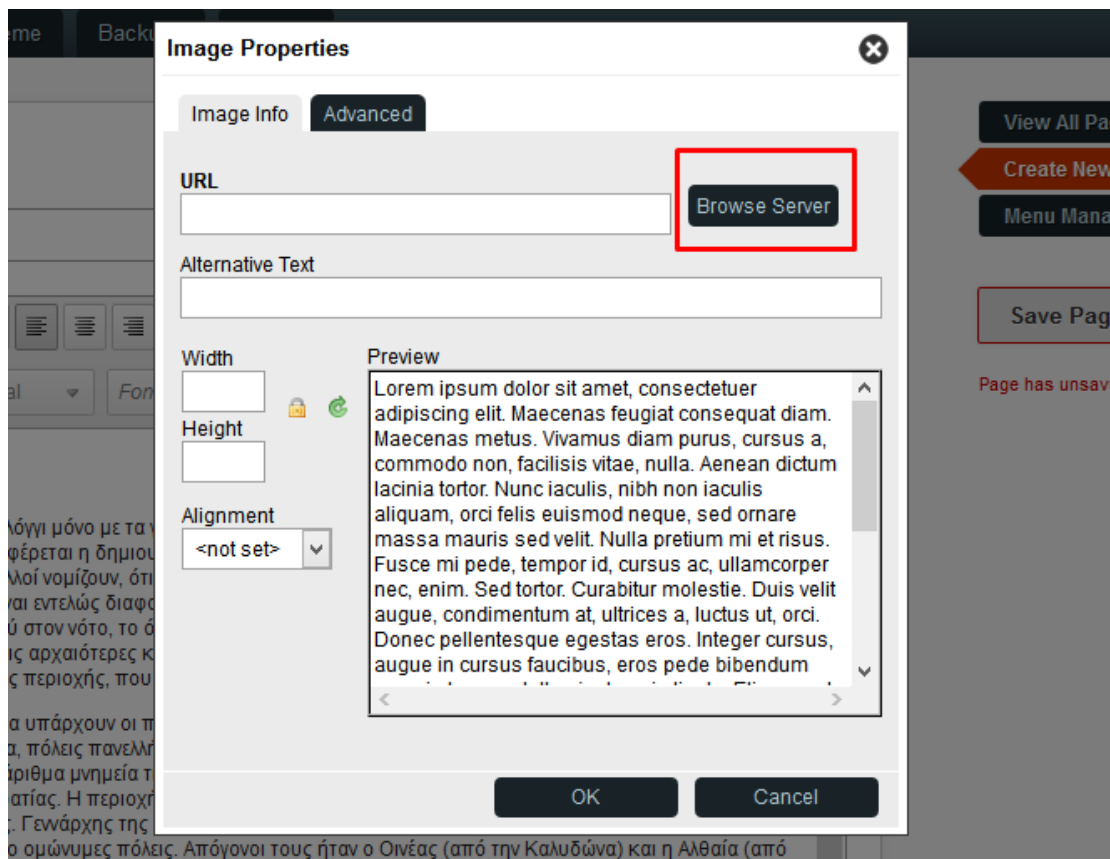
Εικόνα 63:Backend browse uploads



Εικόνα 64:Backend select browse uploads



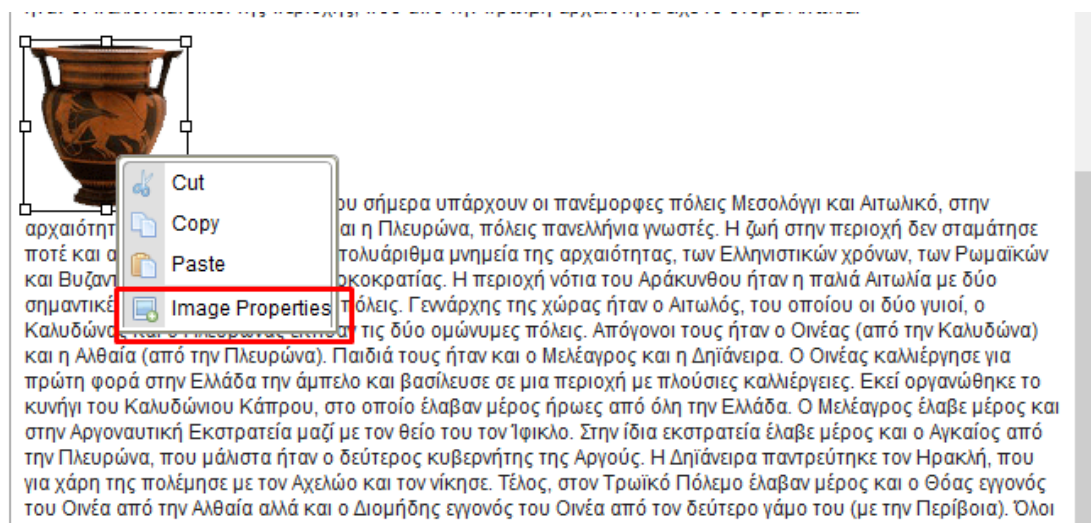
Εικόνα 65:Backend Uploading images



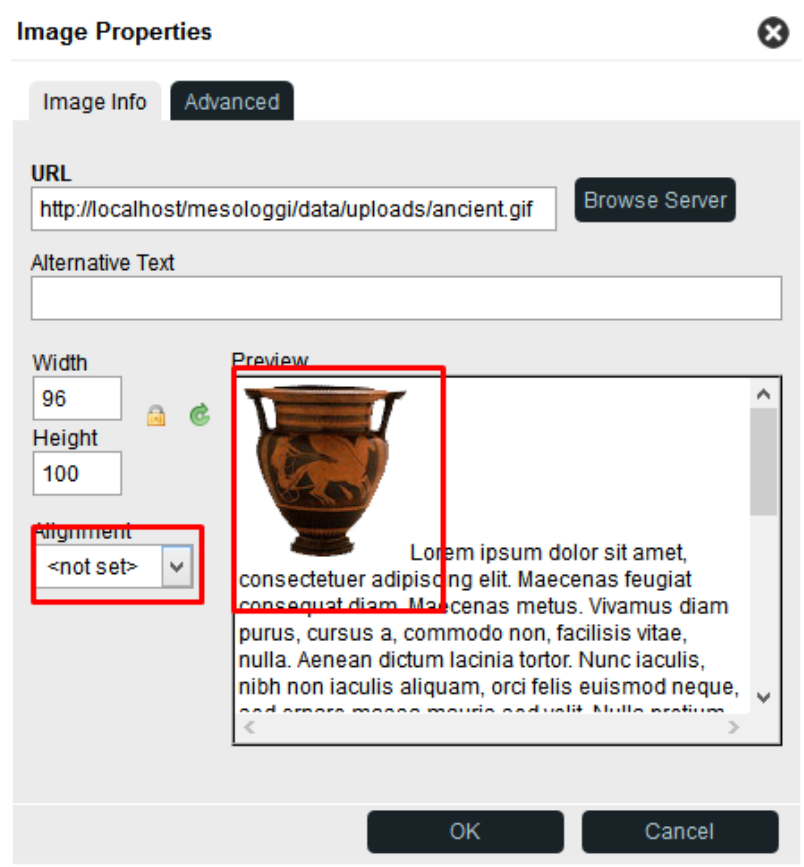
Εικόνα 66:Backend Εισαγωγή εικόνας σε κείμενο



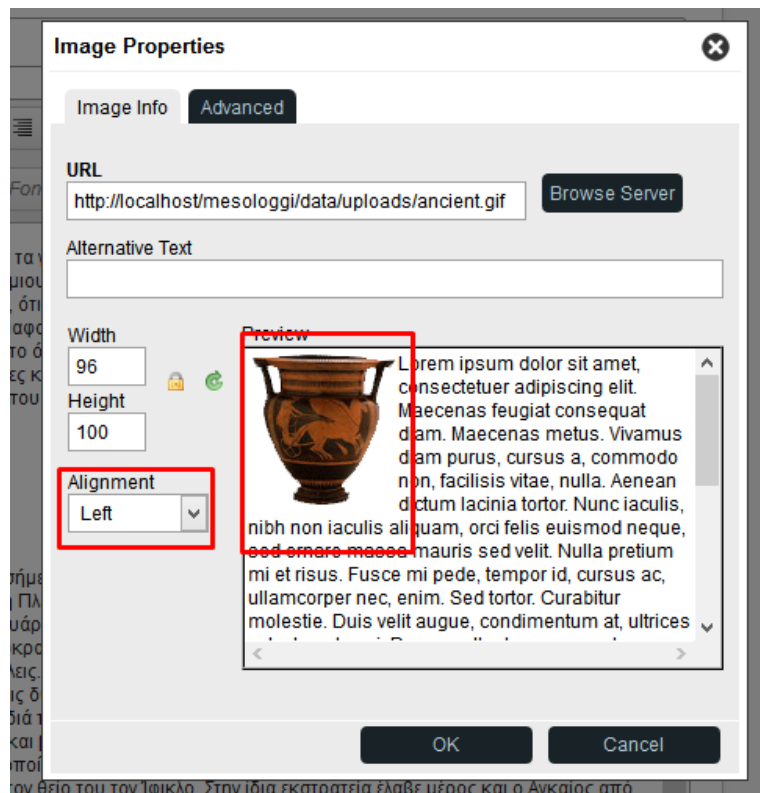
Εικόνα 67:Backend Εισαγωγή εικόνας σε κείμενο



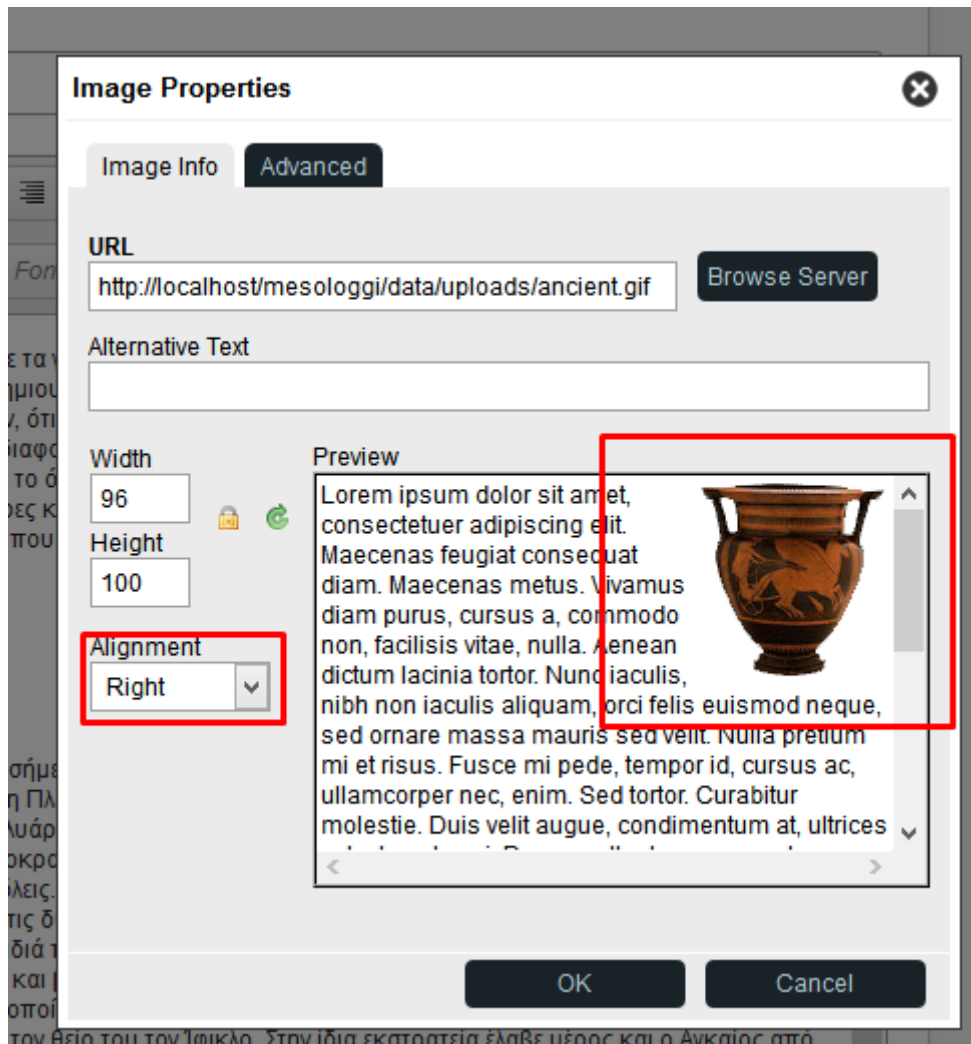
Εικόνα 68:Backend Εισαγωγή εικόνας σε κείμενο



Εικόνα 69:Backend Εισαγωγή εικόνας σε κείμενο



Εικόνα 70:Backend Εισαγωγή εικόνας σε κείμενο



Εικόνα 71:Backend Εισαγωγή εικόνας σε κείμενο

Αρχαιότητα

Πολλοί συνδέουν το Μεσολόγγι μόνο με τα γεγονότα της Επανάστασης του 1821 και την Εξοδο του 1826. Εξ' άλλου στις σχετικές ιστορίες αναφέρεται η δημιουργία της πόλης από πειρατές και ψαράδες Έλληνες και ξένους κατά τον 16ο αιώνα. Έτσι πάρα πολλοί νομίζουν, ότι η πόλη ιδρύθηκε σε μια περιοχή έρημη και ακατοίκητη έως την εποχή εκείνη. Η αλήθεια όμως είναι εντελώς διαφορετική. Η περιοχή που περικλείεται από την λιμνοθάλασσα του Μεσολογγίου και Αιτωλικού στον νότο, το όρος Αράκυνθος στον βορρά, τον Εύηνο στα ανατολικά και τον Αχελώο στα δυτικά είναι μια από τις αρχαιότερες κατοικημένες περιοχές του Ελληνικού χώρου. Οι Κουρήτες και οι Αιτωλοί ήταν οι παλιοί κάτοικοι της περιοχής, που από την πρώτη αρχαιότητα είχε το όνομα Αιτωλία.

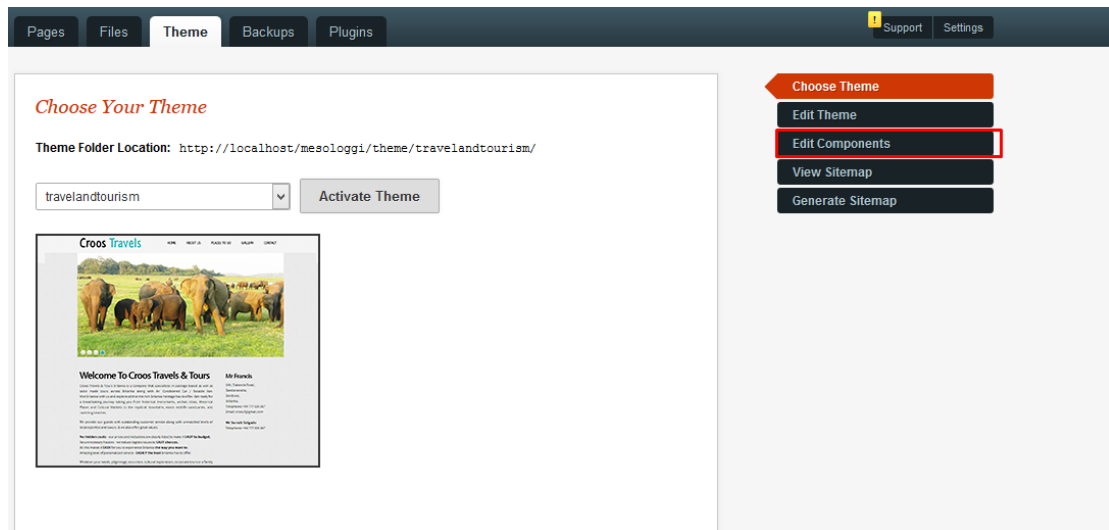
Στην περιοχή όπου σήμερα υπάρχουν οι πανέμορφες πόλεις Μεσολόγγι και Αιτωλικό, στην αρχαιότητα άκμαζαν η Καλυδώνα και η Πλευράνα, πόλεις πανελλήνια γνωστές. Η ζωή στην περιοχή δεν σταμάτησε ποτέ και αυτό μαρτυρείται από τα πολυάριθμα μνημεία της αρχαιότητας, των Ελληνιστικών χρόνων, των Ρωμαϊκών και Βυζαντινών χρόνων και της Τουρκοκρατίας. Η περιοχή νότια του Αράκυνθου ήταν η παλιά Αιτωλία με δύο σημαντικές και πανελλήνια γνωστές πόλεις. Γεννάρχης της χώρας ήταν ο Αιτωλός, του οποίου οι δύο γιοι, ο Καλυδώνας και ο Πλευράνας έδειξαν τις δύο σύγχρονες πόλεις. Απόγονοι τους ήταν ο Ονώνας (από την Καλυδώνα)

Χιλιόμετρικές Αποστάσεις

Πόλη	χλμ Πόλη
ΑΘΗΝΑ	248 ΚΟΖ.
ΑΛΕΞ/ΠΟΛΗ	865 ΚΟΜ
ΑΜΦΙΣΣΑ	150 ΚΟΡΙ
ΑΡΤΑ	123 ΛΑΜ.
ΒΕΡΟΙΑ	498 ΛΑΡ.
ΒΟΛΟΣ	339 ΛΕΙΦ
ΓΡΕΒΕΝΑ	375 ΛΕΥ.
ΔΡΑΜΑ	676 ΝΑΥ.
ΕΔΕΣΣΑ	516 ΞΑΝ
ΗΓΟΥΜΕΝΙΤΣΑ	251 ΠΑΤΙ
ΘΕΣ/ΝΙΚΗ	523 ΠΟΛ.
ΙΩΑΝΝΙΝΑ	198 ΠΡΕΙ
ΚΑΒΑΛΑ	688 ΠΥΡ.
ΚΑΛΑΜΑΤΑ	268 ΣΕΡΦ
ΚΑΡΔΙΤΣΑ	314 ΣΠΑ.
ΚΑΡΠΙΕΝΗΣΙ	156 ΤΡΙΚ
ΚΑΣΤΟΠΙΑ	467 ΤΡΙΠ

Εικόνα 72:Τελική μορφή κειμένου στο frontend

Επίσης μπορούμε να βάλουμε επιπλέον components στα theme προσθέτοντας μερικές γραμμές κώδικα στον πηγαίο κώδικα php του site.



Εικόνα 73:Edit Components στο tab Theme



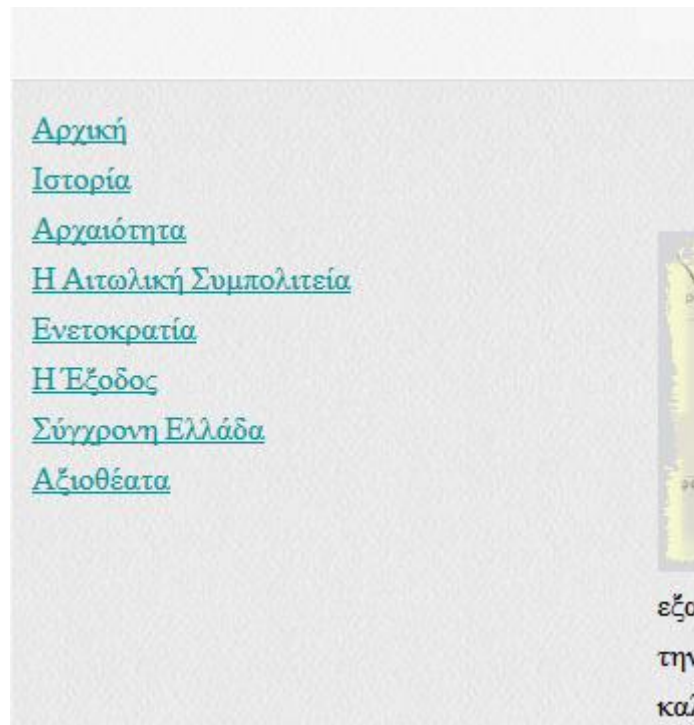
Εικόνα 74:Συμπληρώνουμε τον κώδικα για το πλευρικό μενού

```
# Include the header template
include('header.inc.php');
?>

<section id="content">
<div class="grid_4">
<?php get_component('sidebar'); ?>
</div>
<div class="main">

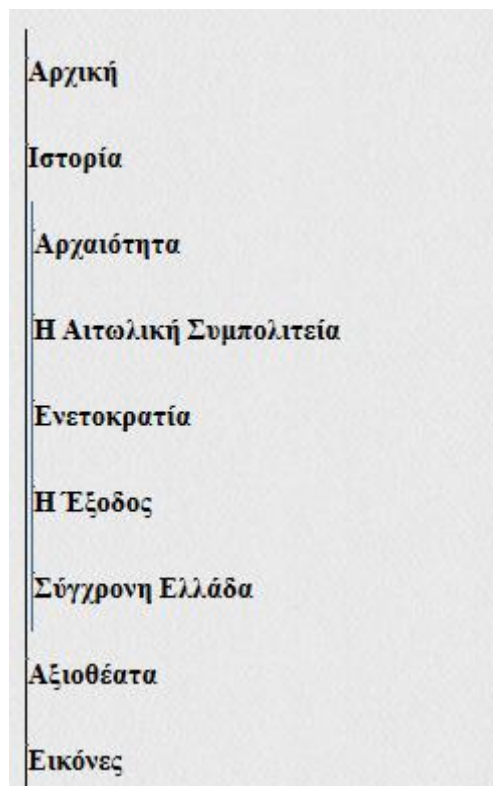
<div class="container_12 sub">
<div class="wrapper">
```

Εικόνα 75:Ενεργοποιούμε το component στον κώδικα της PHP



Εικόνα 76: Εμφανίζεται το μενού στην πλευρική στήλη

Το οποίο μορφοποιήσαμε με CSS κ πήρε την τελική μορφή του με εφέ σαν δέντρο.



Εικόνα 77: Το πλευρικό μενού σε μορφή δέντρου μετά από εμπλουτισμό με CSS

Για να προσθέσουμε παραπάνω χαρακτηριστικά στο site βάλαμε 2 προσθετα.

Το I18N που μας βοηθάει να δώσουμε πολυγλωσσικότητα στις σελίδες μας, πολυεπιπεδα μενού μιας και η αρχική έκδοση δεν μας δίνει αυτή τη δυνατότητα και επίσης breadcrumbs

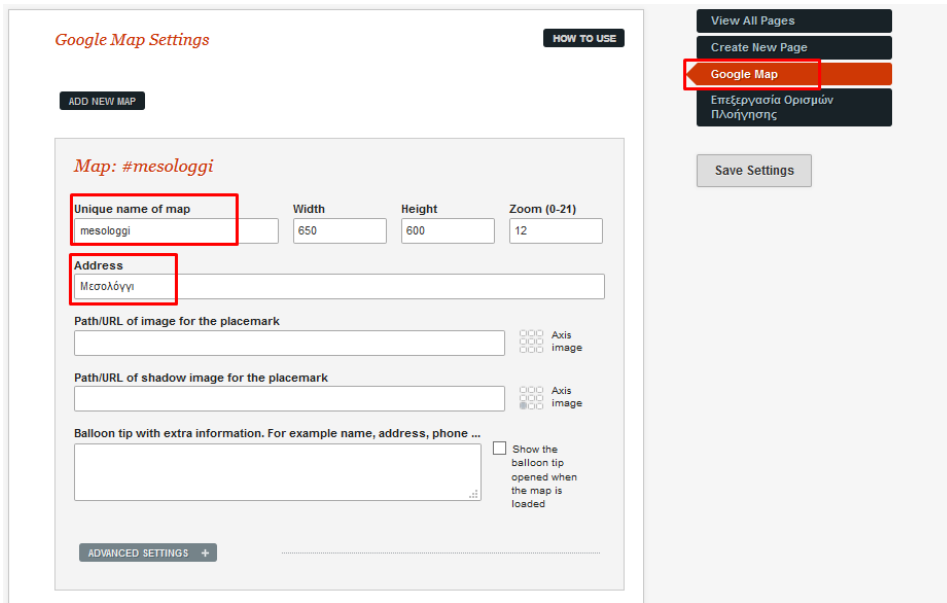
Το google maps το οποίο μας βοήθησε να βάλουμε τον χάρτη της ομώνυμης υπηρεσίας σε μια σελίδα στο site.

PLUGIN	DESCRIPTION	STATUS
Send Anonymous Data	GetSimple needs your anonymous data so we can understand our users better. It's completely voluntary and will help us improve our product. Version 1.0 — Author: Chris Cagle	Deactivate
Google map	{googlemap/DESCRIP} Version 2.0 — Author: singularae	Deactivate
I18N Base	{i18n_base/PLUGIN_DESCRIPTION} Version 3.2.8 — Author: Martin Vicek	Deactivate
I18N Custom Fields	Custom fields for pages (I18N enabled) - based on Mike Swan's plugin Version 1.9.1 — Author: Martin Vicek	Deactivate
I18N	{i18n_navigation/PLUGIN_DESCRIPTION} Version 3.2.8 — Author: Martin Vicek	Deactivate
Innovation Theme Plugin	Settings for the default GetSimple theme: Innovation Version 1.2 — Author: Chris Cagle	Deactivate
nested_menus.php	Disabled Plugin	Activate

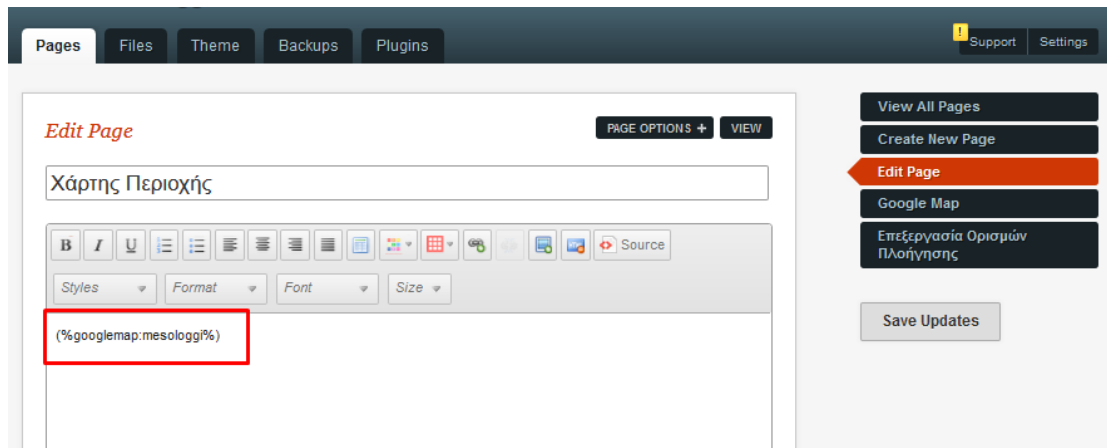
Εικόνα 78:Ενεργοποίηση προσθέτων

Page Name	Slug	Status
Visit Mesologgi	visit-mesologgi	# x
Ιστορία	istoria	# x
Αξιοθέατα	axiotheata	# x
Εικόνες	eikonες	# x
Ξενοδοχεία	xe-nodocheia	# x
Δραστηριότητες	drastiriotites	# x
Τοπικά Προϊόντα	topika-proionta	# x
Χάρτης Περιοχής	charthi-periochi	# x
Επικοινωνία	epikoinonia	# x

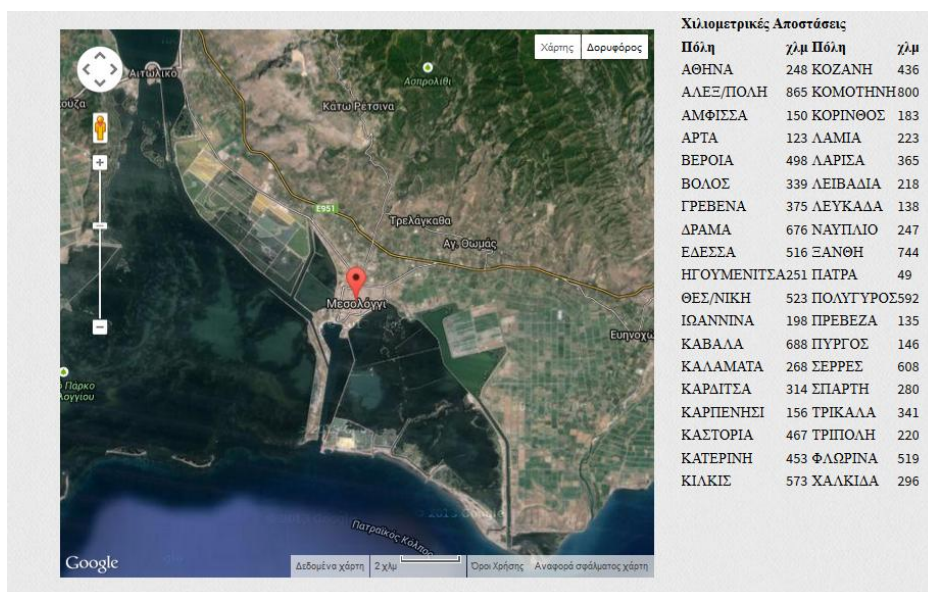
Εικόνα 79:Χρήση προσθέτου I18N στο pages



Εικόνα 80: Ορισμός google map στο backend

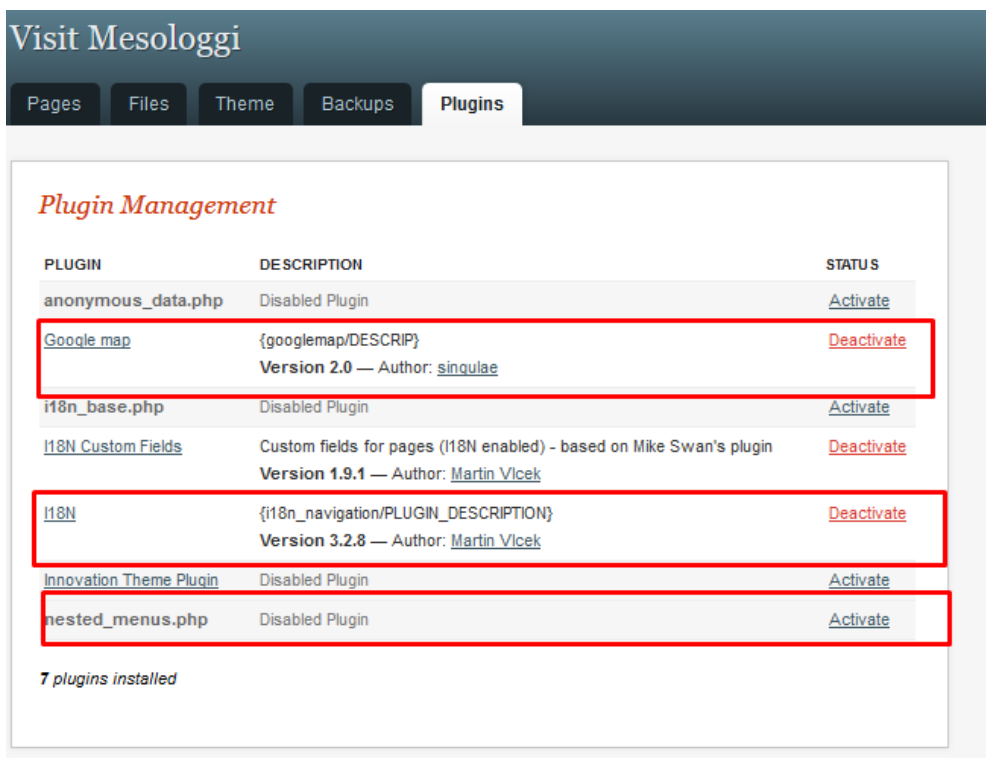


Εικόνα 81: Προσθήκη εντολής στη σελίδα για να εμφανιστεί ο χάρτης στο frontend



Εικόνα 82: Χάρτης στο frontend

Όπως επίσης για τα plugins φαίνονται αν είναι ενεργοποιημένα από ένα έγγραφο xml το plugins.xml που βρίσκεται στον φάκελο C:\xampp\htdocs\mesologgi\data\other. Βλέπουμε ότι τα προσθετα Google map και I18N είναι ενεργοποιημένα ενώ το nested_menus δεν είναι. Αυτό φαίνεται και στο αρχείο plugins.xml.



Εικόνα 83:Plugins Activated



Εικόνα 84:Plugins.xml

Αν αλλάξω το αρχείο plugins.xml θα έχει άμεσο αντίκτυπο κ στο backend.
Π.χ Στο item nested_menus θα αλλάξω την τιμή του enabled από false σε true

Δηλαδή το :

```
<item>
<plugin><![CDATA[nested_menus.php]]></plugin>
<enabled><![CDATA[false]]></enabled>
</item>
```

Θα το κανω:

```
<item>
<plugin><![CDATA[nested_menus.php]]></plugin>
<enabled><![CDATA[true]]></enabled>
</item>
```

Και αυτό θα μου αλλάξει αυτόματα το προσθετο στο backend σε ενεργοποιημένο.



```
<?xml version="1.0" encoding="UTF-8"?>
<channel><item><plugin><![CDATA[anonymous_data.php]]></plugin>
<enabled><![CDATA[false]]></enabled>
</item>
<item><plugin><![CDATA[googlemap.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
<item><plugin><![CDATA[i18n_base.php]]></plugin>
<enabled><![CDATA[false]]></enabled></item>
<item><plugin><![CDATA[i18n_customfields.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
<item><plugin><![CDATA[i18n_navigation.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
<item><plugin><![CDATA[InnovationPlugin.php]]></plugin>
<enabled><![CDATA[false]]></enabled></item>
<item><plugin><![CDATA[nested_menus.php]]></plugin>
<enabled><![CDATA[false]]></enabled></item>
</channel>
```

Εικόνα 85:Plugin enabled:False

Visit Mesologi

Pages Files Theme Backups **Plugins**

Plugin Management

PLUGIN	DESCRIPTION	STATUS
anonymous_data.php	Disabled Plugin	Activate
Google map	{googlemap/DESCRIP} Version 2.0 — Author: singulae	Deactivate
i18n_base.php	Disabled Plugin	Activate
i18N Custom Fields	Custom fields for pages (i18N enabled) - based on Mike Swan's plugin Version 1.9.1 — Author: Martin Vlcek	Deactivate
i18N	{i18n_navigation/PLUGIN_DESCRIPTION} Version 3.2.8 — Author: Martin Vlcek	Deactivate
Innovation Theme Plugin	Disabled Plugin	Activate
nested_menus.php	Disabled Plugin	Activate

7 plugins installed

Εικόνα 86:Plugin Disabled

```

xml x plugins.xml x
<?xml version="1.0" encoding="UTF-8"?>
<channel><item><plugin><![CDATA[anonymous_data.php]]></plugin>
<enabled><![CDATA[false]]></enabled>
</item>
<item><plugin><![CDATA[googlemap.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
<item><plugin><![CDATA[i18n_base.php]]></plugin>
<enabled><![CDATA[false]]></enabled></item>
<item><plugin><![CDATA[i18n_customfields.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
<item><plugin><![CDATA[i18n_navigation.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
<item><plugin><![CDATA[InnovationPlugin.php]]></plugin>
<enabled><![CDATA[false]]></enabled></item>
<item><plugin><![CDATA[nested_menus.php]]></plugin>
<enabled><![CDATA[true]]></enabled></item>
</channel>

```

Εικόνα 87:Plugin enabled:True

Visit Mesologgi

Pages Files Theme Backups **Plugins**

Plugin Management

PLUGIN	DESCRIPTION	STATUS
anonymous_data.php	Disabled Plugin	Activate
Google map	{googlemap/DESCRIP} Version 2.0 — Author: singulae	Deactivate
i18n_base.php	Disabled Plugin	Activate
i18N Custom Fields	Custom fields for pages (i18N enabled) - based on Mike Swan's plugin Version 1.9.1 — Author: Martin Vlcek	Deactivate
i18N	{i18n_navigation/PLUGIN_DESCRIPTION} Version 3.2.8 — Author: Martin Vlcek	Deactivate
Innovation Theme Plugin	Disabled Plugin	Activate
Nested Menus	Provides a set of template tags to return or print the menu as a set of nested arrays, i.e. child items under parent. Version 1.4 — Author: Chris Bloom	Deactivate

7 plugins installed

Εικόνα 88:Plugin Enabled

Οι λειτουργίες του CMS GetSimple ακόμα και στη βασική έκδοση είναι πολλές. Με τα κατάλληλα προσθετα πολλαπλασιάζονται. Επεμβαίνοντας στον κώδικα PHP και στο CSS και προσθέτοντας δικα μας javascript μπορούμε να φτιάξουμε ένα πλήρως λειτουργικό και όμορφο site.

ΚΕΦΑΛΑΙΟ 5

ΑΠΟΤΕΛΕΣΜΑΤΑ

5.1 Συμπεράσματα

Μέσα από αυτή την εργασία μπορέσαμε να δημιουργήσουμε ένα ολοκληρωμένο οδηγό για το Μεσολόγγι με τη μορφή ενός δυναμικού ιστότοπου, φτιαγμένο με την τεχνολογία xml. Βλέποντας ότι οι περισσότεροι ιστοτοποι λειτουργούν με βάσεις δεδομένων καταφέραμε να δώσουμε ένα εξίσου ωραίο αποτέλεσμα χωρίς να εμπλακούμε καθόλου με αυτές. Χρησιμοποιώντας ένα νέο για μας εργαλείο μάθαμε τα οφέλη τα πλεονεκτήματα και τα μειονεκτήματα των ιστοτοπων που έχουν γραφτεί σε xml έναντι των κλασσικών ιστοτοπων φτιαγμένους με CMS.

Εχοντας πλέον ολοκληρώσει τον ιστοτοπο και εχοντας ψάξει και δουλέψει με το συγκεκριμένο εργαλείο φτάνουμε στο συμπέρασμα οτι δεν έχει τίποτα να ζηλέψει από τα υπόλοιπα δημοφιλή CMS που χρησιμοποιούν βάσεις δεδομένων. Βέβαια μας βοήθησε το γεγονός ότι δεν είχαμε να επεξεργαστούμε μεγάλο όγκο δεδομένων κ ότι έπρεπε να φτιάξουμε μια σελίδα η οποία θα είναι καθαρά για διαφήμιση και προώθηση. Επομένως σε τέτοιες απλές σελίδες ξέρουμε πλέον ότι μπορούμε να εχουμε και αυτή την λύση χωρίς να σηκώνουμε βαριά CMS σε διακομιστες οι οποίοι πρέπει να υποστηρίζουν βάση δεδομένων. Το ότι αποθηκεύει τις παλιότερες αναρτήσεις μας βοηθάει παρα πολύ στο να βρίσκουμε πληροφορίες που διαγράψαμε τυχαία. Τέλος το ότι μπορείς να πάρεις backup με ένα απλοί copy paste από το σημείο που εχεις ανεβάσει το site είναι παρα πολύ χρήσιμο μιας και δεν θα χάσεις τα δεδομένα που έχουν αποθηκευτεί σε κάποια βάση αλλά θα τα αντιγράψεις μαζί με τα υπόλοιπα αρχεία του site.

Ενας άλλος στόχος μας ήταν μέσα από την πτυχιακή, πέρα από το να μάθουμε πώς να στήνουμε ένα website με xml από το μηδέν, ήταν και η προβολή του όμορφης πόλης του Μεσολογγίου. Οι φωτογραφίες, οι πληροφορίες για τις δραστηριότητες αλλά και κάποια πράγματα για την ιστορία του και τα αξιοθέατα που μπορείς να δεις, που μπορείς να μείνεις και τι να δοκιμάσεις, ελπίζουν στο να κάνουν κάποιους ανθρώπους να γνωρίσουν την πόλη και να την επισκεφτούν μελλοντικά.

Τέλος η ενασχόληση μας με αυτό το τεράστιο κομμάτι της επιστήμης της Πληροφορικής μας απέδωσε χρήσιμες εμπειρίες που θα μας βοηθήσουν σίγουρα στο μέλλον.

5.2 Μελλοντική εργασία

Η εφαρμογή που περιεγράφηκε παραπάνω, περιέχει αρκετές από τις βασικές λειτουργίες και υπηρεσίες που μπορεί να προσφέρει ένα website. Ωστόσο, πάντα θα μπορούν να υπάρξουν βελτιώσεις και προσθήκες, αισθητικές και λειτουργικές, με σκοπό το αποτέλεσμα να είναι ακόμα καλύτερο.

Μερικές πιθανές επεκτάσεις και προσθήκες θα μπορούσαν να είναι οι παρακάτω:

- Δυναμικό σύστημα κρατήσεων: Πλήρες λειτουργικό σύστημα όπου οι διαθεσιμότητες των παροχών κράτησης θα ενημερώνονται δυναμικά.
- Επιπλέον, θα ήταν καλο ο χρήστης να μπορεί να κάνει τις συναλλαγές που αφορούν τις κρατήσεις μέσω πιστωτικής κάρτας ή PayPal.
- Πρόσβαση στο backend για όποιον ξενοδόχο θέλει να συμμετέχει στο δυναμικο σύστημα κρατήσεων για να ενημερώνει και να ενημερώνεται για την πληρότητα των δωματίων.
- Προσθήκη διαφημίσεων τοπικών επιχειρήσεων.
- Δυναμική ενημέρωση για καιρο σε συνεργασία με εξειδικευμένα site.
- Μπαρα ειδήσεων για την περιοχή.
- Πολυγλωσσικότητα: Δυνατότητα επέκτασης της εφαρμογής σε διάφορες γλώσσες όπως αγγλικά.
- Ανίχνευσιμότητα από τις μηχανές αναζήτησης: Καταχώρηση του διαδικτυακού τόπου σε μηχανές αναζήτησης με τις κατάλληλες ρυθμίσεις SEO.
- Social Media: Δημιουργία και επέκταση της ιστοσελίδας στα social media.
- Κουμπιά για άμεση αλληλεπίδραση με τα social media(π.χ. like για facebook, follow για twitter κ.ο.κ.)
- Κριτική χρηστών: Δημιουργία μια μπάρας με αστέρια όπου οι επισκέπτες θα μπορούν να ψηφίζουν με π.χ. πολύ κακό το 1 και πολύ καλό το 5 το κάθε δημοσιευμένο άρθρο.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Benatallah, B.; Casati, F.; Toumani, F. (2004). "Web service conversation modeling: A cornerstone for e-business automation". IEEE Internet Computing
- [2] Jacobs, Ian; Walsh, Norman (15 December 2004). "URI/Resource Relationships". Architecture of the World Wide Web, Volume One.
- [3] Robins, Kevin & Frank Webster. (1999). Times of the Technoculture: From the Information Society to the Virtual Life, London: Routledge.
- [4] Hauben, M. (1994) 'Netizens an Anthology'
- [5] The Information Revolution", J. R. Okin. ISBN 0-9763857-4-0. Ed. Ironbound Press, 2005.
- [6] Harris, Andy (2001). JavaScript Programming for the Absolute Beginner. Premier Press. ISBN 0-7615-3410-5.
- [7] Goodman, Danny; Eich, Brendan (2001). JavaScript Bible. John Wiley & Sons. ISBN 0-7645-3342-8.
- [8] Jesse James Garrett (18 February 2005). "Ajax: A New Approach to Web Applications".
- [9] Ullman, Chris (March 2007). Beginning Ajax. wrox. ISBN 978-0-470-10675-4.
- [10] Σαμψών, Δ. (2003). Η γλώσσα σήμανσης XML. Πανεπιστήμιο Πειραιώς
- [11] H. M. Deitel, P. J. Deitel, T. R. Nieto, T. M. Lin and P. Sadhu. XML How to program, Prentice Hall, 2001
- [12] Cerami, S. St Laurent and E. Building XML Applications. New York: McGraw-Hill, 1999.
- [13] Bosak, Jon Tim Bray. XMLandthe Second-Generation Web. Scientific American, 1999.
- [14] "Introduction to XML". Course Slides. Pierre Geneves. October 2012.
- [15] Damien Foggon, Daniel Maharry, Chris Ullman and Karli Watson. Programming Microsoft .NET XML Web Services, Microsoft Press, 2004.
- [16] Mathew Macdonald. Microsoft .NET Distributed Applications: Integrating XML Web Services and .NET Remoting, 2001Sall, K.B., 2002.
- [17] XML Family of Specifications. A Practical Guide. Addison Wesley.
- [18] Content Management Bible, Bob Boiko. John Wiley & Sons, 2005.

Διαδικτυακές Πηγές

- [1] Wikipedia, 2015, <http://en.wikipedia.org>
- [2] Ελληνική Wikipedia, 2015, <http://el.wikipedia.org>
- [3] W3 Schools, 2015, <http://www.w3schools.com/>
- [4] World Wide Web Consortium, 2015, <http://www.w3.org>
- [5] GetSimple , 2015, get-simple.info
- [6] XAMPP Apache + MySQL + PHP + Perl, 2015, <https://www.apachefriends.org/index.HTML>
- [7] Apache HTTP Server, 2015, <http://httpd.apache.org/>
- [8] Dictionary Merriam Webster, 2015, <http://www.merriam-webster.com>
- [9] ΕΝΗΜΕΡΩΣΗ ΜΕΣΩ ΔΙΑΔΙΚΤΥΟΥ «ΨΗΦΙΖΟΥΝ» ΟΙ ΕΥΡΩΠΑΙΟΙ, 6 Ιουνίου 2012, <http://www.newsbomb.gr/media-agb/story/139241/enimerosi-meso-diadiktyoy-psifizoyn-oi-eyropaioi>
- [10] Ενημέρωση μέσω τηλεόρασης – ενημέρωση μέσω διαδικτύου σημειώσατε 1, 12 Ιανουάριου 2012, <http://www.videotex.gr/%CE%B5%CE%BD%CE%B7%CE%BC%CE%AD%CF%81%CF%89%CF%83%CE%B7-%CE%BC%CE%AD%CF%83%CF%89-%CE%B4%CE%B9%CE%B1%CE%B4%CE%B9%CE%BA%CF%84%CF%8D%CE%BF%CF%85-%CE%B5%CE%BD%CE%B7%CE%BC%CE%AD%CF%81%CF%89%CF%81%CE%B7/>
- [11] A Privacy Manifesto for the Web 2.0 Era, 2015, <https://gigaom.com/2008/01/08/a-privacy-manifesto-for-the-web-20-era/>
- [12] Εισαγωγή στην προσβασιμότητα στο διαδίκτυο (WebAIM), 2015, <http://microo.net/article.php?intro-to-web-accessibility>
- [13] http://users.sch.gr/tsibinos/internet_history/history_1.HTML
- [14] Microsoft Technet, 2014, <https://technet.microsoft.com/en-us/library/cc958978.aspx>
- [15] Γενικά σχετικά με την HTML, 2015, <http://www.it.uom.gr/project/HTML2/tut0.HTML>

- [16] Μαθήματα Κατασκευής Web Σελίδων με τη Γλώσσα HTML, 2005, <http://www.eeei.gr/odhgos/HTMLfaq.htm>
- [17] ΚΕΝΤΡΟ ΠΛΗ.ΝΕ.Τ. Ν. ΦΛΩΡΙΝΑΣ Η Γλώσσα Προγραμματισμού XHTML, 2015, <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-XHTML.HTML>
- [18] Wlearn, 2014, <http://www.wlearn.gr/index.php/CSS-84>
- [19] PHP Manual, 2015, <http://php.net/manual/en/index.php>
- [20] The greek technology team, 2011, <http://www.techteam.gr/wiki/PHP>
- [21] Ajax: A New Approach to Web Applications, 2005, <http://www.adaptivepath.com/ideas/ajax-new-approach-web-applications/>
- [22] tutor.edu.gr, 2015, <http://www.tutor.edu.gr/index.php/ajax/ajax-intro>
- [23] Πλεονεκτήματα και Μειονεκτήματα της XML, 2015, http://www0.dmst.aueb.gr/louridas/lectures/dais/web_intro/ar01s22.HTML
- [24] Free Software foundation, 2015, <http://www.fsf.org/>
- [25] Ελεύθερο λογισμικό/ Λογισμικό ανοιχτού κώδικα, 2015, <https://ellak.gr/>
- [26] Εισαγωγή στα συστήματα διαχείρισης περιεχομένου, Θεόδωρος Δούβλης, 2015, <http://www.slideshare.net/rodotheos/CMS-421346>