



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών

Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων



Πτυχιακή Εργασία

Τίτλος: Δημιουργία ιστοσελίδας με χρήση HTML 5, μιας εταιρίας και παρουσίαση της ίδιας όσο και των προϊόντων της

Πάυλος Αντωνάκος (ΑΜ: 1163)

Επιβλέπων Καθηγητής :Νικόλαος Παπαδάκης

Ημερομηνία Παρουσίασης :2/7/2014

Abstract

The markup languages were initially created for the need of presentation of document with distinguishable way, giving the suitable directives on the program of presentation with regard to the way of depiction of document. They were created and developed for different reasons each time (eg the mission of documents via networks, the description of particular electronic documents characteristics etc). The HTML undeniably is the most popular markup language which adopted these characteristics and developed through the various publications in the last, the HTML 4.01. The development is continued with the creation of the new model of HTML 5.0, which is the effort of a constructors team in the beginning and later the W3C organism . It contains enough new elements and possibilities, while also at this moment that it is written, there is a permanent addition of new characteristics.

Σύνοψη

Οι γλώσσες σήμανσης αρχικά δημιουργήθηκαν για την ανάγκη παρουσίασης ενός εγγράφου με διακριτό τρόπο, δίνοντας τις κατάλληλες οδηγίες για το πρόγραμμα παρουσίασης σχετικά με τον τρόπο απεικόνισης του εγγράφου. Δημιουργήθηκαν και εξελίχθηκαν για διαφορετικούς λόγους κάθε φορά (π.χ την αποστολή εγγράφων μέσω δικτύων , την περιγραφή ιδιαίτερων χαρακτηριστικών ηλεκτρονικών εγγράφων κλπ). Η HTML αναμφισβήτητα είναι η πιο δημοφιλής γλώσσα σήμανσης η οποία υιοθέτησε αυτά τα χαρακτηριστικά και εξελίχθηκε μέσα από τις διάφορες εκδόσεις της στην τελευταία την HTML 4.01. Η εξέλιξη συνεχίζεται με την δημιουργία του νέου προτύπου της HTML 5.0 η οποία είναι η προσπάθεια μιας ομάδας κατασκευαστών στην αρχή και μετέπειτα του οργανισμού W3C. Περιέχει αρκετά νέα στοιχεία και δυνατότητες, ενώ και αυτή τη στιγμή που γράφεται υπάρχει μια διαρκής προσθήκη νέων χαρακτηριστικών.

Πίνακας Περιεχομένων

Κεφάλαιο 1- Εισαγωγή	10
1.1. Γλώσσες σήμανσης (Markup Languages)	10
1.2. Ιστορικά στοιχεία.....	10
1.3. Χαρακτηριστικά.....	13
1.3.1. Τύποι Σήμανσης.....	14
1.3.2. Ανεξαρτησία δεδομένων.....	17
1.3.3. Τύπος Εγγράφου	17
1.4. Χρήσεις	17
Κεφάλαιο 2- HTML 5.....	19
2.1. Εισαγωγή	19
2.2. Περιγραφή – Εξέλιξη της HTML	19
2.3. Δομή της HTML5	21
2.3.1. Σύνταξη.....	22
2.3.2. Μεταδεδομένα	23
2.3.3. Σενάρια - Αρχεία.....	24
2.3.4. Στοιχεία HTML5.....	25
2.3.5. Ενότητες.....	25
2.3.6. Ομαδοποίηση περιεχομένου	27
2.3.7. Σημασιολογικά στοιχεία κειμένου	29
2.3.8. Επεξεργασμένο κείμενο	31
2.3.9. Ενσωματωμένο περιεχόμενο.....	31
2.3.10. Canvas.....	33
2.3.11. Φόρμες	37
2.3.12. Πίνακες	41
2.3.13. Σύνδεσμοι	43

2.4. Διεπαφές προγραμματισμού (APIs).....	45
2.4.1. Communication APIs.....	45
2.4.2. Διεπαφή συστήματος αρχείων (FilesystemAPIs)	48
2.4.3. GeolocationAPI.....	49
Κεφάλαιο 3- Άλλες τεχνολογίες	52
3.1. Εισαγωγή	52
3.2. Τεχνολογίες ανάπτυξης δικτυακών τόπων.....	52
3.2.1. XHTML	52
3.2.2. PHP	52
3.2.3. JSP (JavaServer Pages)	53
3.2.4. Javascript.....	53
3.2.5. Αρχεία στυλ CSS	54
3.2.6. Ajax.....	55
3.2.7. MySQL	55
3.2.8. Oracle.....	57
3.3. Πλαίσια ανάπτυξης διαδικτυακών εφαρμογών.....	59
3.4. Συστήματα CMS	62
Κεφάλαιο 4- Υλοποίηση.....	66
4.1. Αρχική σελίδα (index.html)	67
4.1.1. Επικεφαλίδα	67
4.1.2. Κύριο μέρος	68
4.1.3. Υποσέλιδο	72
4.2. Σελίδα “aboutus.html”	72
4.3. Σελίδα “herbaltea.html”	73
4.4. Σελίδα “flavors.html”	75
4.5. Σελίδα “contact.html”	80
4.6. Δευτερεύουσες σελίδες.....	82

Κεφάλαιο 5- Συμπεράσματα, μελλοντικές επεκτάσεις.....	86
Βιβλιογραφία	88

Πίνακας Εικόνων

Εικόνα 1-1 Ιστορική εξέλιξη στις κύριες γλώσσες σήμανσης.....	11
Εικόνα 2-1 Δέντρο DOM.....	22
Εικόνα 2-2 HTML 5 - Σχέδιο εγγράφου.....	26
Εικόνα 2-3 Χρήση στοιχείων κειμένου	28
Εικόνα 2-4 Αριθμημένες λίστες.....	28
Εικόνα 2-5 Περιγραφική λίστα	29
Εικόνα 2-6 Σημασιολογικά στοιχεία κειμένου	30
Εικόνα 2-7 Σύστημα συντεταγμένων canva	35
Εικόνα 2-8 Παράδειγμα σχεδιασμού σε καμβά.....	36
Εικόνα 2-9 Φόρμα HTML5	40
Εικόνα 2-10 - Φόρμες μηνύματα λάθους.....	40
Εικόνα 2-11 Αποτέλεσμα χρήσης href	44
Εικόνα 2-12 Αποτέλεσμα χρήσης της area, map	44
Εικόνα 2-13 Παράδειγμα εφαρμογής FilesystemAPI.....	49
Εικόνα 2-14 Εντοπισμός θέσης	51
Εικόνα 3-1 Ajax (πηγή: http://www.w3schools.com/ajax/ajax_intro.asp).....	55
Εικόνα 3-2 Αρχιτεκτονική MySQL	56
Εικόνα 3-3 Αρχιτεκτονική Oracle	57
Εικόνα 3-4 Μοντέλο MVC	60
Εικόνα 3-5 Αρχιτεκτονική 3-tier	61
Εικόνα 3-6 Πλαίσια ανάπτυξης ιστοτόπων	62
Εικόνα 4-1 Δημιουργία εικόνας διαβάθμισης.....	68
Εικόνα 4-2 Επικεφαλίδα	68
Εικόνα 4-3 Εικόνα παρουσίασης	69
Εικόνα 4-4 Εισαγωγικό κείμενο	69
Εικόνα 4-5 Ενότητα παρουσίασης.....	70
Εικόνα 4-6 Ενότητα πληροφοριών	72
Εικόνα 4-7 Υποσέλιδο	72

Εικόνα 4-8 Σελίδα "Σχετικά με εμάς"	73
Εικόνα 4-9 Αναδυόμενα παράθυρα πληροφορίας	74
Εικόνα 4-10 Λίστες στην σελίδα herbusherbaltea	75
Εικόνα 4-11 Πεδίο ορισμού ποσότητας	76
Εικόνα 4-12 Φόρμα αποστολής παραγγελίας	78
Εικόνα 4-13 Φόρμα επικοινωνίας	81
Εικόνα 4-14 Σελίδα με βίντεο	82
Εικόνα 4-15 Σελίδα με ανακοινώσεις	82
Εικόνα 4-16 Σελίδα με τις πληροφορίες παραγωγής	83
Εικόνα 4-17 Σελίδα με το δίκτυο συνεργατών	83

Λίστα Πινάκων

Πίνακας 2-1 Νέα στοιχεία της HTML 5	25
---	----

Σκοπός της εργασίας

Η παρούσα εργασία, έχει σαν σκοπό την διερεύνηση και την δημιουργία ενός δικτυακού τόπου μέσω του νέου προτύπου στις γλώσσες σήμανσης, της HTML5.

Αρχικά θα γίνει αναφορά στις γλώσσες σήμανσης και στην ιστορική εξέλιξη τους. Θα περιγραφούν συνοπτικά τα κύρια χαρακτηριστικά, αλλά και η χρήση τους.

Στο δεύτερο κεφάλαιο θα αναλυθεί διεξοδικά η γλώσσα HTML5. Θα αναφερθούμε στα κυριότερα στοιχεία της, ιδιαίτερα δε σε εκείνα τα σημεία τα οποία αποτελούν τα καινούργια χαρακτηριστικά στη γλώσσα.

Στο τρίτο κεφάλαιο θα προσεγγίσουμε νέες τεχνολογίες δημιουργίας ιστοτόπων και θα παρουσιάσουμε τα τεχνικά χαρακτηριστικά τους.

Το τέταρτο κεφάλαιο αποτελεί την υλοποίηση του ιστοτόπου. Τέλος θα προσπαθήσουμε να εξάγουμε συμπεράσματα, σχετικά με την διαδικασία υλοποίησης και θα αναφερθούμε σε μελλοντικές επεκτάσεις.

Εισαγωγή

Γλώσσες σήμανσης (Markup Languages)

Μια γλώσσα σήμανσης είναι ένα σύστημα για τον σχολιασμό και τη παρουσία ενός εγγράφου με διακριτό τρόπο. Η ονομασία προήρθε από τη διαδικασία αναθεώρησης εγγράφων από τους συντάκτες. Οι οδηγίες ή διορθώσεις επισημαίνονταν με μπλε μολύβι στα χειρόγραφα των συγγραφέων. Τέτοια παραδείγματα στοιχειοθεσίας συναντώνται στην troff¹ και LaTeX², αλλά και στις ετικέτες σήμανσης αυτών των γλωσσών.

Η σήμανση καθοδηγεί το λογισμικό να προβεί στην εμφάνιση του κειμένου, σύμφωνα με τις οδηγίες που ενσωματώνονται. Μερικές γλώσσες σήμανσης, όπως η HTML, έχουν προκαθορισμένη σημασιολογική παρουσίαση, πράγμα που σημαίνει ότι οι προδιαγραφές τους ορίζουν, πως τα δεδομένα πρόκειται να παρουσιαστούν. Άλλες δεν υποστηρίζουν αυτή τη δυνατότητα όπως για παράδειγμα η XML.

Η HTML είναι μια ευρέως χρησιμοποιούμενη γλώσσα σήμανσης υπέρ – κειμένου, δηλαδή εγγράφων του διαδικτύου. Αποτελεί μια εκδοχή της SGML, παρ' όλο που δεν απόλυτα συμβατή με αυτή. Η SGML αποτελείται από κανόνες σήμανσης, οι οποίοι συναντώνται στη βιομηχανία των εκδόσεων μεταξύ των συγγραφέων, εκδοτών και εκτυπωτικών συστημάτων.

Ιστορικά στοιχεία

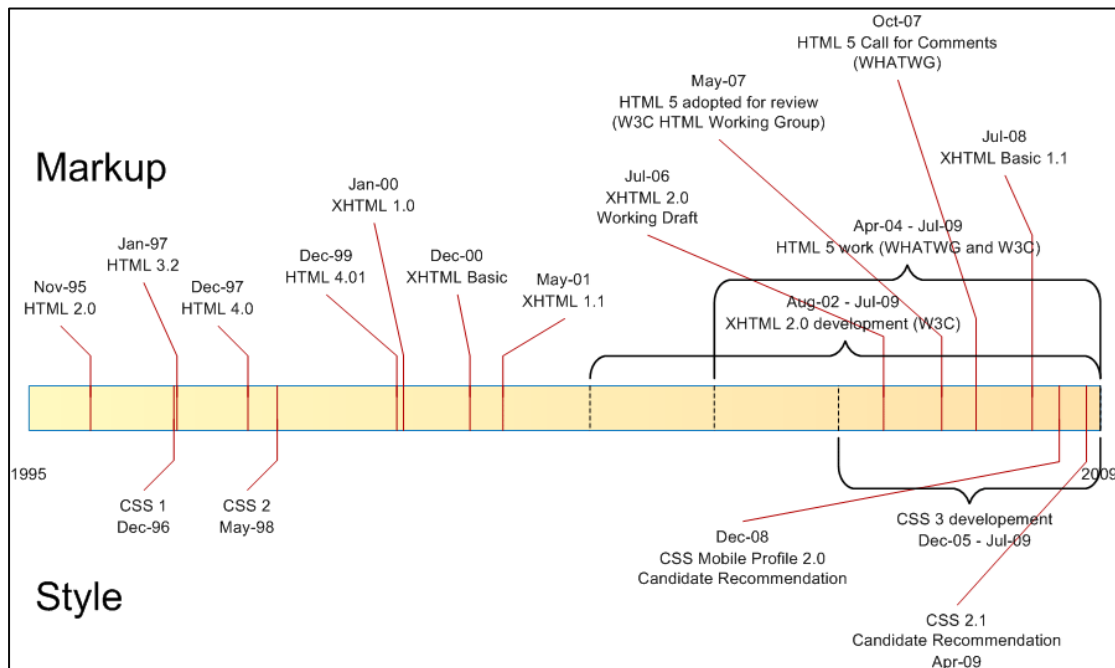
Η πρώτη γνωστή δημόσια παρουσίαση μιας γλώσσας σήμανσης στην επιστήμη των υπολογιστών, έγινε από τον William W. Tunnicliffe σε ένα συνέδριο το 1967, αν και ο ίδιος ονόμασε την παρουσίαση ως «γενική κωδικοποίηση». Στη δεκαετία του 1970 ο Tunnicliffe πραγματοποίησε την ανάπτυξη ενός προτύπου που ονομάστηκε GenCode και εφαρμόστηκε στην εκδοτική βιομηχανία. Αργότερα, ήταν ο πρώτος πρόεδρος της επιτροπής του Διεθνούς Οργανισμού Τυποποίησης που δημιούργησε τη SGML, το πρώτο πρότυπο περιγραφικής γλώσσας σήμανσης.

Παρ' όλα αυτά, ένας ερευνητής της IBM, ο Charles Goldfarb εμφανίζεται συχνότερα ως ο πατέρας των γλωσσών σήμανσης. Ο Goldfarb, ήρθε αντιμέτωπος με τη βασική ιδέα το 1969, όταν ανέπτυξε ένα σύστημα διαχείρισης εγγράφων για δικηγορικές

¹Το Troff, είναι το κύριο συστατικό ενός συστήματος επεξεργασίας εγγράφων το οποίο αναπτύχθηκε από την AT&T για το λειτουργικό σύστημα Unix (Corderoy, n.d.)

²Το LaTeX είναι ένα σύστημα στοιχειοθεσίας. Περιλαμβάνει συστατικά για την δημιουργία τεχνικής και επιστημονικής τεκμηρίωσης (Anon., n.d.).

εταιρίες και βοήθησε στην ανάπτυξη της IBMGMLαργότερα τον ίδιο χρόνο. HGMLπρωτοπαρουσιάστηκε το 1974.



Εικόνα 0-1 Ιστορική εξέλιξη στις κύριες γλώσσες σήμανσης

Το 1975 έγινε αρμόδιος σχεδίασης προϊόντων για την IBM στο ερευνητικό κέντρο της εταιρίας. Εκεί έπεισε τα στελέχη της IBM για την ανάπτυξη και εμπορική διάδοση της GML ως μέρος του προϊόντος «DocumentCompositionFacility», που αφορούσε ένα σύστημα διαχείρισης εγγράφων.

Η SGML, βασίστηκε στη GML και GenCode και αναπτύχθηκε από τον Goldfarb το 1974. Ο ίδιος έγινε τελικά πρόεδρος της επιτροπής SGML, η οποία κυκλοφόρησε από το ISO, σαν ISO 8879 τον Οκτώβριο του 1986.

Ένα άλλο δημοφιλές πρότυπο σήμανσης είναι το TeX, το οποίο δημιουργήθηκε από τον Donald Knuth τη δεκαετία του 80. Το TeX πρόκειται για μια γλώσσα σήμανσης, η οποία περιγράφει ένα λεπτομερές σχέδιο του κειμένου, αλλά και των γραμματοσειρών και χρησιμοποιήθηκε κυρίως σε μαθηματικά βιβλία. Χρησιμοποιείται κυρίως στη σύγχρονη μορφή του LaTeX το οποίο αποτελεί το ηλεκτρονικό ανάλογο του TeX και χρησιμοποιείται για ακαδημαϊκούς λόγους (π.χ. εμφάνιση μαθηματικών τύπων σε κείμενα)

Η πρώτη γλώσσα που έκανε διακριτή τη διαφορά μεταξύ της δομής και παρουσίασης, ήταν η Scribe, η οποία αναπτύχθηκε από τον Brian Reid και

περιγράφηκε το 1980 στο διδακτορικό του. Η Scribeka καινοτόμησε σε πολλούς τομείς, αλλά κυρίως παρουσίασε την ιδέα του διαχωρισμού των στυλ από την περιγραφή. Επηρέασε επίσης την ανάπτυξη της GML (αργότερα SGML).

Στις αρχές της δεκαετίας του 1980, παρουσιάστηκε η ιδέα, ότι η σήμανση θα έπρεπε να εστιάζει στη δομή ενός εγγράφου και να αφήνει την παρουσίαση σε μία άλλη διαδικασία. Αυτή η θεώρηση οδήγησε στην δημιουργία της Πρότυπης Γενικευμένης Γλώσσας Σήμανσης (SGML: Standard Generalized Markup Language), η οποία αποτέλεσε πρόγονο της HTML και του LaTeX.

Η SGML περιέγραψε τη σύνταξη για την συμπερίληψη της σήμανσης στα έγγραφα, καθορίζοντας ποιες ετικέτες (tags) επιτρέπονται αλλά και σε ποια θέση. Αυτό επέτρεψε στους δημιουργούς να υιοθετήσουν τις ετικέτες που ήθελαν, δημιουργώντας τις δικές τους γλώσσες σήμανσης. Υπό αυτό τον όρο, ουσιαστικά η SGML αποτέλεσε μια μετά-γλώσσα σήμανσης, από την οποία κληρονόμησαν χαρακτηριστικά οι επόμενες. Το 1986 η SGML κυκλοφόρησε σαν Διεθνές Πρότυπο, από τον Παγκόσμιο Οργανισμό Πιστοποίησης (ISO 8879).

Το 1991, η ανάγκη για μια απλή μορφή της SGML, οδήγησε τους Sir Tim Berners-Lee, να χρησιμοποιήσουν την SGML για να δημιουργήσουν την HTML. Η απλότητα αυτή, ίσως ήταν και ο σημαντικότερος παράγοντας επέκτασης της HTML, η οποία αποτελεί την πιο χρησιμοποιούμενη γλώσσα σήμανσης σήμερα.

Μία άλλη προσπάθεια πολύ αργότερα, τον Ιανουάριο του 2000 περιελάμβανε την δημιουργία μιας μετά – γλώσσας σήμανσης από το Παγκόσμιο Οργανισμό Ιστού την Εκτεταμένη Γλώσσα Σήμανσης (XML: eXtensible Markup Language). Η XML αναπτύχθηκε από τη Κοινοπραξία του Παγκόσμιου Ιστού (W3C: World Wide Consortium), στα πλαίσια μια επιτροπής που Πρόεδρος ήταν ο Jon Bosak. Η κύρια επιδίωξη της XML, ήταν η απλούστευση της SGML. Παρέμεινε όμως μια μετά-γλώσσα όπως η SGML, παρέχοντας τη δυνατότητα στους χρήστες να δημιουργήσουν τις δικές τους ετικέτες, σύμφωνα με τις ανάγκες τους. Η XML χρησιμοποιήθηκε κυρίως επειδή διατήρησε τη συμβατότητα με την SGML και κάθε XML έγγραφο, μπορούσε να γραφεί με τέτοιο τρόπο ώστε να είναι και SGML έγγραφο.

Από τον Ιανουάριο του 2000 όλες οι προδιαγραφές της HTML βασίστηκαν στην XML, αντί για την SGML, χρησιμοποιώντας την ονομασία Εκτεταμένη Γλώσσα

Σήμανσης Υπερκειμένου (XHTML: eXtendedHyperTextMarkupLanguage). Οι προδιαγραφές όριζαν ότι τα έγγραφα HTML, θα έπρεπε να είναι καλώς μορφοποιημένα με αυστηρότερο τρόπο και κανόνες (Πομπόρτσης, 2009).

Από τότε έχουν κυκλοφορήσει αρκετές εκδόσεις, τις οποίες θα δούμε αναλυτικότερα στα επόμενα κεφάλαια.

Χαρακτηριστικά

Ένα κοινό χαρακτηριστικό πολλών γλωσσών σήμανσης, είναι η ανάμιξη του κειμένου ενός εγγράφου με οδηγίες σήμανσης που ενσωματώνονται στο ίδιο το αρχείο. Ο κώδικας που ακολουθεί αποτελεί ένα τέτοιο παράδειγμα.

```
<h1>Επικεφαλίδα</h1>
<p>
    Παράγραφος <i>Πλάγια Εγγραφή</i> κείμενο, κείμενο, κείμενο.
</p>
```

Το κείμενο μεταξύ αυτών των οδηγιών είναι το πραγματικό κείμενο του εγγράφου. Οι κωδικοί **<h1>**, **<p>** είναι οδηγίες, για την μορφοποίηση και σημασιολογική σήμανση του κειμένου. Συγκεκριμένα ο κωδικός **<h1><h1>** χρησιμοποιείται για να δηλώσει ότι «αυτό είναι το πρώτο επίπεδο επικεφαλίδας», ενώ ο **<p></p>** για να δηλώσει την αρχή και το τέλος μιας παραγράφου. Η ετικέτα **<i></i>** είναι ένα παράδειγμα σήμανσης παρουσίασης, γιατί χρησιμοποιείται για να καθορίσει ένα ιδιαίτερο χαρακτηριστικό στο κείμενο.

Οι ετικέτες αναφέρονται και σαν στοιχεία της γλώσσας και δεν ακολουθούν κάποιο κανόνα δημιουργίας, ούτε αυτός ο τρόπος περιγράφεται. Παρ' όλα αυτά οι δημιουργοί αυτών των συνόλων, φροντίζουν να υιοθετούν κατανοητά ονόματα για τα στοιχεία. Εντός του κειμένου κάθε στοιχείο θα πρέπει να εισάγεται με μια σήμανση αρχής και να κλείνει με μία σήμανση τέλους. Η σήμανση αρχής δηλώνεται με το όνομα του στοιχείου με αγκύλες ανοίγματος, ενώ η σήμανση τέλους αποτελείται από αγκύλες ανοίγματος και τον χαρακτήρα εγκοπής (slash).

Τα ιδιαίτερα χαρακτηριστικά των γλωσσών σήμανσης, ουσιαστικά αποτελούν στο μεγαλύτερο βαθμό χαρακτηριστικά της SGML. Αυτά αφορούν κυρίως:

- Το τύπο της σήμανσης.
- Τον καθορισμό του τύπου εγγράφου.
- Την ανεξαρτησία της γλώσσας από τα συστήματα εφαρμογής.

1.2.1. Τύποι Σήμανσης

HSGML καθόρισε την περιγραφική σήμανση, κατά την οποία χρησιμοποιούνται κώδικες παροχής ονομάτων, για την κατηγοριοποίηση των τμημάτων του εγγράφου. Αυτοί οι κώδικες καθορίζουν για παράδειγμα, ότι η επόμενη καταγραφή είναι μια παράγραφος. Όμως στη πορεία οι απόγονοι της SGML καθόρισαν νέους τύπους ηλεκτρονικής σήμανσης όπως:

- Σήμανση παρουσίασης
- Διαρθρωτική σήμανση
- Διαδικαστική σήμανση
- Σημασιολογική σήμανση

1.2.1.1. Σήμανση παρουσίασης

Παρόλο που οι προδιαγραφές της γλώσσας HTML καθορίζουν ότι η σήμανση μπορεί να χρησιμοποιηθεί για τη δημιουργία μιας ιστοσελίδας, αυτός που είναι υπεύθυνος για την επιλογή των στοιχείων της γλώσσας σήμανσης είναι ο ίδιος ο συγγραφέας.

Αν ο συγγραφέας χρησιμοποιεί λογισμικό που επιτρέπει την επεξεργασία ενός εγγράφου HTML οπτικά και όχι σε επίπεδο κώδικα, τότε είναι προτιμότερο να χρησιμοποιείται σήμανση παρουσίασης. Για παράδειγμα η εμφάνιση μιας επικεφαλίδας μπορεί να επιτευχθεί, μορφοποιώντας το κείμενο κατάλληλα (μεγαλύτερη γραμματοσειρά, έντονα γράμματα κλπ), παρά να εφαρμοστεί ένα στυλ επικεφαλίδας. Η σήμανση για μία τέτοια επικεφαλίδα θα μπορούσε να έμοιαζε:

```
<p><font size="+7" color="#000000"><b>Επικεφαλίδα</b></font></p>
```

Παρόμοια η δημιουργία μιας λίστας σε σήμανση παρουσίασης και σε σήμανση δομής θα ήταν:

Σήμανση	Κώδικας	Εμφάνιση
Παρουσίασης	<pre><p> -Στοιχείο 1
 -Στοιχείο 2
 -Στοιχείο 3
 </p></pre>	<pre>- Στοιχείο 1 - Στοιχείο 2 - Στοιχείο 3</pre>

Δομής	<pre> Στοιχείο 1 Στοιχείο 2 Στοιχείο 3 </pre>	<ul style="list-style-type: none"> • Στοιχείο 1 • Στοιχείο 2 • Στοιχείο 3
-------	---	--

Η σήμανση παρουσίασης παρουσιάζει πολλά μειονεκτήματα όπως:

- Δεν είναι αποδοτική ως προς την συντήρηση του κώδικα. Για κάθε μελλοντική αλλαγή θα πρέπει να γίνεται επεξεργασία κάθε σελίδας.
- Δεν είναι φιλική ως προς τις μηχανές αναζήτησης κατά την ανίχνευση του site. Για παράδειγμα αν θέλουμε να εμφανίσουμε ένα τίτλο στην κορυφή της σελίδας ενός htmlεγγράφου, θα μπορούσαμε να χρησιμοποιήσουμε κάποια ετικέτα μορφοποίησης, όπως η **** και να μεγαλώσουμε τη γραμματοσειρά. Με αυτή όμως τη προσέγγιση, ένας webcrawler κατά την ανίχνευση της σελίδας δε θα καταλάβει τη διαφορά στο κείμενο. Αν όμως χρησιμοποιήσουμε τις διαθέσιμες ετικέτες **<H(x)>**, οι οποίες ορίζουν τίτλους, τότε ο webcrawler θα καταλάβει ποιος είναι ο τίτλος της σελίδας και σε τι αυτή αναφέρεται.
- Η υποστήριξη για τα στοιχεία που αφορούν τη σήμανση παρουσίασης, πολλές φορές εγκαταλείπεται από τις επόμενες εκδόσεις των γλωσσών σήμανσης, με αποτέλεσμα οι καινούργιοι φυλλομετρητές να μην μπορούν να εμφανίζουν αυτά τα στοιχεία σωστά.

1.2.1.2. Διαρθρωτική σήμανση

Η διαρθρωτική σήμανση χρησιμοποιείται για την περιγραφή της δομής ενός κειμένου. Η HTML5 χρησιμοποιεί δομικά στοιχεία όπως:

- **Header:** Κεφαλίδα της σελίδας
- **Head:** Τμήμα που περιέχει οδηγίες και μεταδεδομένα
- **Body:** Το κυρίως περιεχόμενο.
- **Footer:** Υποσέλιδο

Κάθε μία ενότητα από τις παραπάνω μπορεί να χρησιμοποιήσει δικές της εσωτερικές δομές οι οποίες χαρακτηρίζονται από στοιχεία όπως: λίστες, πίνακες, τμήματα div κλπ.

1.2.1.3. Διαδικαστική σήμανση

Η σήμανση είναι ενσωματωμένη στο κείμενο και παρέχει οδηγίες για τα προγράμματα που πρόκειται να το επεξεργαστούν. Τέτοιες παραδείγματα, είναι η troff, LaTeX και PostScript. Ο επεξεργαστής σαρώνει το κείμενο από την αρχή έως το τέλος, ακολουθώντας τις οδηγίες που βρίσκει. Τα γενικά χαρακτηριστικά της διαδικαστικής σήμανσης είναι:

- Καθορίζει τον τρόπο που θα επεξεργαστεί το κείμενο.
- Παρέχει οδηγίες για τον τρόπο επεξεργασίας
- Ασχολείται με τη μορφοποίηση και παρουσίαση.
- Είναι αποτελεσματική καθώς δε χρειάζεται περπατάει επεξεργασία.

1.2.1.4. Σημασιολογική σήμανση

Δεν υπάρχει επίσημη προδιαγραφή που να χαρακτηρίζεται ως σημασιολογική σήμανση. Τα χαρακτηριστικά όμως της HTML «πιέζουν» προς αυτή τη κατεύθυνση. Η σημασιολογική HTML αναφέρεται σε μια πρακτική, για τη δημιουργία εγγράφων, τα οποία θα περιέχουν μόνο την προτεινόμενη από το συγγραφέα έννοια, χωρίς καμία αναφορά για το πώς αυτή παρουσιάζεται (Πομπόρτσης, 2009).

Χρησιμοποιείται για τη σήμανση μερών του εγγράφου, με ετικέτες οι οποίες δηλώνουν την σημασία του κειμένου. Υπάρχει μια λεπτή γραμμή ανάμεσα στη σημασιολογική και στη διαρθρωτική σήμανση. Για παράδειγμα οι επικεφαλίδες H(x) από τη μεριά του συγγραφέα είναι διαρθρωτική σήμανση αφού αποτελεί το τίτλο της σελίδας ο οποίος μπαίνει στην αρχή του κειμένου, για μια μηχανή αναζήτησης όμως έχει εννοιολογική σημασία γιατί αποτελεί το σήμα κατατεθέν του κειμένου. Τα χαρακτηριστικά της είναι:

- Αναγνώριση των λογικών στοιχείων ενός κειμένου.
- Δεν καθορίζει τις διαδικασίες που θα εφαρμοστούν στο κείμενο και για αυτό τον λόγο απαιτείται άλλη διαδικασία για αυτό. Για παράδειγμα περιέχει την πληροφορία για το μέγεθος του κειμένου, αλλά δεν επεξεργάζεται την αλλαγή.

1.2.2. Ανεξαρτησία δεδομένων

Η ανεξαρτησία δεδομένων, διασφαλίζει ότι τα έγγραφα που θα περιγραφούν με μία γλώσσα σήμανσης, θα μπορούν να μεταφέρονται από ένα περιβάλλον υλικού και λογισμικού σε ένα άλλο, χωρίς απώλεια πληροφορίας (Πομπόρτσης, 2009).

1.2.3. Τύπος Εγγράφου

Ένα άλλο χαρακτηριστικό είναι ο ορισμός τύπου εγγράφου (DTD: DocumentTypeDefinition). Ουσιαστικά ο τύπος εγγράφου χρησιμοποιείται για την κατάταξη των εγγράφων σε τύπους. Με αυτό τον τρόπο το πρόγραμμα που θα «διαβάσει» το έγγραφο, συνήθως ο συντακτικός αναλυτής, θα γνωρίζει τις διαδικασίες που θα πρέπει να ακολουθήσει για την επεξεργασία του.

Χρήσεις

Αν και η ιδέα της γλώσσας σήμανσης ξεκίνησε με τα έγγραφα κειμένου, υπάρχει μια αυξανόμενη χρήση και σε άλλους τομείς που αφορούν την παρουσίαση διαφόρων τύπων πληροφορία όπως :

- Μουσικές λίστες
- Διανυσματικά γραφικά
- Διαδικτυακές υπηρεσίες
- Διεπιφάνειες χρήστη

Οι περισσότερες από αυτές τις εφαρμογές χρησιμοποιούν την XML επειδή όπως είδαμε είναι δυνατή η δημιουργία ετικετών ανάλογα με τις ανάγκες χρήσης. Ενδεικτικά μερικές από αυτές τις γλώσσες σήμανσης είναι:

- **BNML (BusinessNarrativeMarkupLanguage):** Χρησιμοποιείται για την σήμανση ηλεκτρονικών συμβολαίων.
- **BPML (BusinessProcessModelingLanguage):** Αποτελεί μια περιγραφική γλώσσα, για την παρουσίαση των δραστηριοτήτων μιας επιχείρησης.
- **CFML (ColdFusionMarkupLanguage):** Είναι μια γλώσσα σεναρίων, για ανάπτυξη εφαρμογών web που εκτελείτε στην εικονική μηχανή της Java, στο πλαίσιο .NET, αλλά και στην μηχανή εφαρμογών της Google.
- **MathML (MathematicalMarkupLanguage):** Χρησιμοποιείται για την περιγραφή μαθηματικών εξισώσεων.
- **CML (ChemicalMarkupLanguage):** Περιγράφει μοριακές δομές, αντιδράσεις, ανάλυση φάσματος, χημική κρυσταλλογραφία κλπ.

- **ABC:**Γλώσσα σήμανσης για την περιγραφή των μουσικών κειμένων (παρτιτούρες) σαν απλό κείμενο.
- **MediaWiki:**Αποτελεί «ελαφριά» έκδοση γλώσσας σήμανσης και χρησιμοποιείται για την δημιουργία σελίδων που διατηρούν γνωσιακές βάσεις.

Εκτός από τις παραπάνω υπάρχει ένας μεγάλος αριθμός γλωσσών σήμανσης³. Αυτές γενικότερα κατηγοριοποιούνται ανάλογα με την χρήση τους σε:

- XML
- Γενικής Χρήσης
- Εγγράφων
- «Ελαφριάς» χρήσης
- Διεπιφάνειας
- Γραφικών
- Υπηρεσιών WEB

³ Μια πλήρης λίστα μπορεί να βρεθεί στην ηλεκτρονική διεύθυνση της Wikipedia:http://en.wikipedia.org/wiki/List_of_markup_languages

HTML 5

Εισαγωγή

Η HTML5 είναι το νέο πρότυπο για την HTML. Σχεδόν όλοι οι νέοι φυλλομετρητές υποστηρίζουν τα περισσότερα από τα καινούργια στοιχεία που εισάγει. Η HTML5 αναπτύσσεται με την υποστήριξη του Παγκόσμιου Συνεταιρισμού Διαδικτύου (W3C:WorldWideWebConsortium) και της ομάδας Εφαρμοσμένης Τεχνολογίας Υπερκειμένου Διαδικτύου (WHATWG: WebHypertextApplicationTechnologyWorkingGroup).

Μερικά από τα καινούργια χαρακτηριστικά της HTML5 περιλαμβάνουν:

- Ανανεωμένα και πρόσθετα στοιχεία ετικετών.
- Μείωση της ανάγκης για εξωτερικά πρόσθετα (plugins).
- Καλύτερη διαχείριση σφαλμάτων.
- Προσανατολισμός στην σήμανση, ελαχιστοποιώντας την ανάγκη για συγγραφή σεναρίων (π.χ javascript).
- Ανεξαρτησία από τη συσκευή εφαρμογής (π.χ ένα html5 έγγραφο, θα μπορεί να φορτωθεί το ίδιο σε έναν υπολογιστή, αλλά και σε μια κινητή συσκευή).

Περιγραφή – Εξέλιξη της HTML

Η γλώσσα σήμανσης υπερκειμένου HTML, εκδόθηκε γύρω στο 1990. Την περίοδο 1993, 1994 υπήρχε μόνο ένας φυλλομετρητής ο NCSAMosaic και οι εξυπηρετητές WEB, ήταν ελάχιστοι. Από τότε ο όγκος του διαδικτύου έχει αυξηθεί εκθετικά, ιδιαίτερα με την έλευση των ευρυζωνικών συνδέσεων.

Η πρώτη έκδοση ήταν η 1.0 και αναπτύχθηκε από τους Tim Berners-Lee. Κατά τη διάρκεια της εργασίας του στο CERN, δούλεψε σε πολλούς υπολογιστές και η εύρεση πληροφορίας σε όλους αυτούς ήταν μια χρονοβόρα διαδικασία. Σκέφτηκε λοιπόν, ότι θα πρέπει να υπάρχει ένας τρόπος, να μπορεί να μεταφέρεται αυτή η πληροφορία από τον έναν υπολογιστή στον άλλο. Αυτή η θεώρηση ενός συστήματος μεταφοράς της πληροφορίας, μέσω τεχνολογιών δικτύου και των αντίστοιχων πρωτοκόλλων, ήταν ουσιαστικά η αρχή της θεμελιώδους γλώσσας του διαδικτύου. Η αρχική έκδοση δεν είχε τις ίδιες δυνατότητες με αυτή που χρησιμοποιούμε σήμερα. Βασιζόταν βέβαια

στην ίδια λογική, αλλά ήταν ένα μικρό υποσύνολο, αν σκεφτούμε ότι αποτελούταν από 22 ετικέτες μόνο.

Η HTML 2.0 εμφανίστηκε το 1995. Δεν ήταν ιδιαίτερα διαφορετική από την 1.0, αλλά ήρθε να προσφέρει καινούργιες δυνατότητες.

Σε αυτό το σημείο το διαδίκτυο άρχισε να γίνεται ολοένα και πιο δημοφιλές και πολλοί θέλησαν να δημιουργήσουν τις δικές τους σελίδες. Για αυτό το λόγο χρειάστηκαν περισσότερες ετικέτες και χαρακτηριστικά για τον εμπλουτισμό αυτών των σελίδων. Παράλληλα ο φυλλομετρητής Netscape είχε κατακτήσει τα πρωτεία, και δημιουργήθηκαν ετικέτες που μπορούσαν να δουλέψουν μόνο σε αυτόν. Για όλους αυτούς τους λόγους παρουσιάστηκε ένα προσχέδιο της HTML 3.0. Όμως οι περισσότεροι φυλλομετρητές, δεν υιοθέτησαν όλα τα νέα χαρακτηριστικά για αυτό το λόγο η εξέλιξη του προσχεδίου εγκαταλείφθηκε.

Εφόσον εγκαταλείφθηκε η έκδοση 3.0 και οι ανάγκες των κατασκευαστών σελίδων μεγάλωνε, η ανάγκη για ένα πρότυπο ήταν επιτακτική. Το 1994 το W3C, σχηματίστηκε για την προτυποποίηση της HTML. Έτσι λοιπόν ξεκίνησε η έκδοση 3.2, η οποία αποτέλεσε τη νέα εξέλιξη της HTML. Η έκδοση 3.2 τελικά έγινε επίσημο πρότυπο τον Ιανουάριο του 1997(Vossen&Hagemann, 2007).

Οι μεγαλύτερες όμως αλλαγές ήρθαν με την έκδοση 4.0, η οποία δημοσιεύθηκε ως σύσταση από το W3C. Ήταν σαφώς διαφορετική από την προηγούμενη και περιελάμβανε αρκετές ετικέτες και ιδιότητες που αφορούσαν το στυλ του κειμένου, αλλά και των συνδέσμων. Όμως αυτό ήταν ταυτόχρονα αρνητικό, γιατί οι κατασκευαστές σπαταλούσαν πολύ χρόνο για να ρυθμίζουν χειροκίνητα αυτές τις ιδιότητες σε όλες τις σελίδες ενός ιστοτόπου. Αυτό που κατανοήθηκε, ήταν ότι το στυλ και οι ιδιότητες, θα έπρεπε να διαχωριστούν και η HTML να προσανατολιστεί στην λογική οργάνωση του εγγράφου. Έτσι λοιπόν αυτή η ανάγκη γέννησε μια ξεχωριστή γλώσσα που ονομάστηκε Αλληλουχία Φύλλων Στυλ (CSS: CascadeStylingSheets). Η HTML 4.0 όμως παρείχε και άλλα χαρακτηριστικά όπως σενάρια, παράθυρα, ενσωματωμένα αντικείμενα, περίπλοκους πίνακες, σύνθετες φόρμες και βελτιωμένα χαρακτηριστικά για άτομα με αναπηρίες. Τελικά έγινε επίσημο πρότυπο τον Απρίλιο του 1998.

Στη συνέχεια ακολούθησε η έκδοση 4.01, που ουσιαστικά ήταν μια αναθεώρηση της 4.0. Συμπεριλάμβανε την προσθήκη μερικών χαρακτηριστικών⁴ όπως για παράδειγμα την προσθήκη της ιδιότητας **name** για την ετικέτα **<form>**. Ταυτόχρονα διορθώθηκαν κάποια λάθη της προηγούμενης έκδοσης.

Η HTML5 δεν αποτελεί μόνο την τελευταία έκδοση της HTML. Αποτελεί ένα σύνολο σχετικών τεχνολογιών για τη δημιουργία μοντέρνων σελίδων με πλούσιο περιεχόμενο.

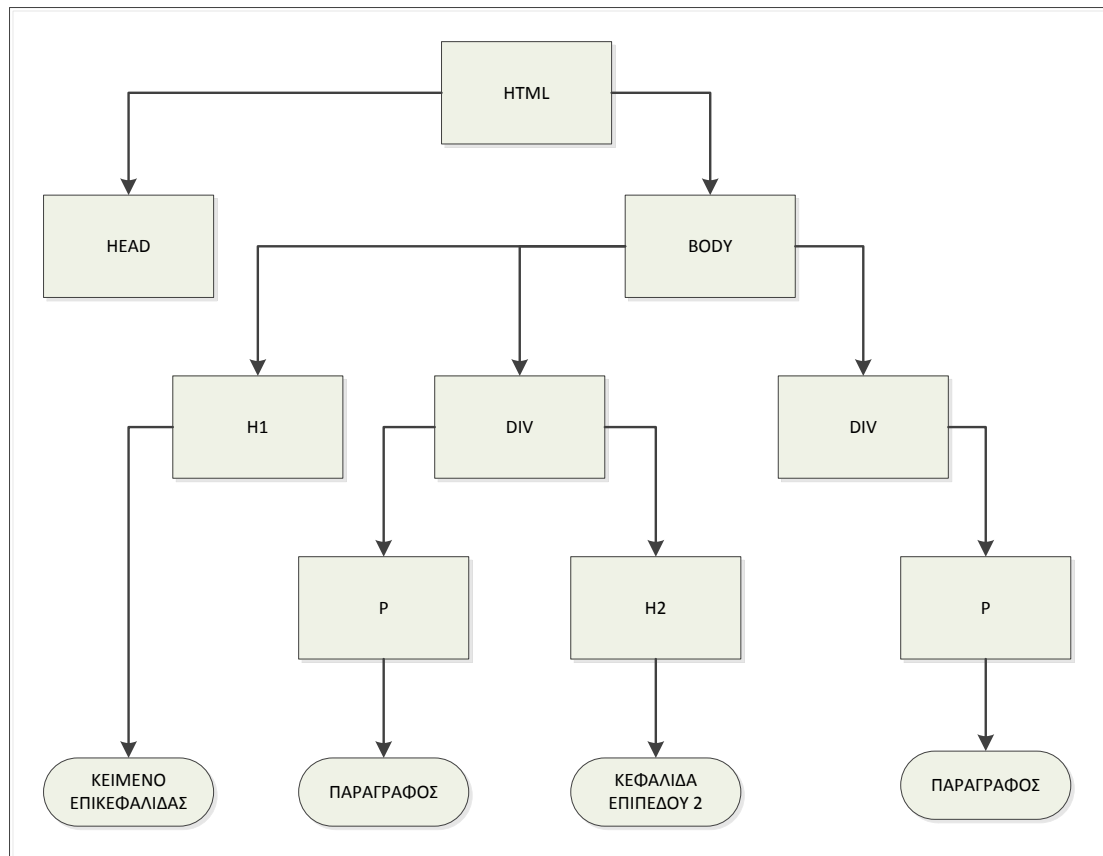
Ο πυρήνας της HTML5, καθορίζει τα συστατικά που χρησιμοποιούνται για την σήμανση ενός κειμένου, με προσανατολισμό την σπουδαιότητα. Τα αρχεία CSS, από την άλλη επιτρέπουν τον έλεγχο της εμφάνισης του κειμένου σήμανσης, όπως αυτό δηλαδή θα παρουσιάζεται στο χρήστη. Η JavaScript, επιτρέπει τον χειρισμό των περιεχομένων σε ένα HTML κείμενο, ανταποκρινόμενο στην αλληλεπίδραση του χρήστη και εκμεταλλεύεται κάποια νέα χαρακτηριστικά της HTML5.

Επειδή η διαδικασία προτυποποίησης είναι μια χρονοβόρα διαδικασία, οι τεχνολογίες που αναφέρονται στο νέο πρότυπο, καθορίζονται μέσω ενός μεγάλου αριθμού μικρότερων σε μέγεθος προτύπων. Το θετικό από αυτή την προσέγγιση είναι ότι τα επί μέρους πρότυπα αναπτύσσονται ανεξάρτητα και πιο γρήγορα. Το αρνητικό όμως είναι ότι δεν είναι εύκολη η ταυτόχρονη παρακολούθηση όλων αυτών των προτύπων και πως σχετίζονται και επηρεάζονται το ένα από το άλλο (Freeman, 2011).

Δομή της HTML5

Το μοντέλο αντικειμένων εγγράφου (DOM: DocumentObjectModel), αναπαριστά όλα τα αντικείμενα σε μία HTML σελίδα. Είναι ανεξάρτητο από τη γλώσσα και τη πλατφόρμα και επιτρέπει την ενημέρωση του περιεχομένου και στυλ της σελίδας όταν αυτή «ανακτηθεί» από τον φυλλομετρητή. Το DOM είναι προσβάσιμο μέσω JavaScript. Για την ανάκτηση μιας HTML σελίδας, οι περισσότεροι φυλλομετρητές χρησιμοποιούν ένα εσωτερικό μοντέλο παρόμοιο με το DOM. Οι κόμβοι κάθε εγγράφου οργανώνονται σε δένδροειδή μορφή, το οποίο καλείται δέντρο DOM. Ο κορυφαίος κόμβος ονομάζεται αντικείμενο «Document». Κάθε έγγραφο HTML ή XML αναπαρίσταται από το αντικείμενο «Document» (McLaughlin, 2011).

⁴ Στην διεύθυνση <http://www.w3.org/TR/REC-html40/appendix/changes.html>, υπάρχει αναλυτική λίστα με τις αναθεωρημένες αλλαγές και προσθήκες.



Εικόνα 0-1 Δέντρο DOM

Η διεύθυνση του εγγράφου είναι σε απόλυτη URL. Όταν το έγγραφο δημιουργείται μέσω κώδικα, π.χ ενός σεναρίου η διεύθυνση του είναι η ίδια με αυτή του σεναρίου.

2.2.1. Σύνταξη

Κάθε έγγραφο αποτελείται από τα στοιχεία που φαίνονται στον κώδικα που ακολουθεί.

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/2002/06/xhtml2" xml:lang="gr">
<head>
  <title>Τίτλοςσελίδας</title>
</head>
<body id="theBody">
  <p>Μιαπαράγραφος</p>
</body>
  
```

Η πρώτη γραμμή σε κάθε έγγραφο είναι ένα ειδικός κώδικας, που λέγεται doctype. Ο ρόλος του είναι να χαρακτηρίζει ότι το έγγραφο που ακολουθεί είναι ένα έγγραφο HTML5. Σε σχέση με τις παλαιότερες εκδόσεις η περιγραφή του έχει απλοποιηθεί αρκετά. Μάλιστα ο μόνος λόγος που υπάρχει στην HTML5 είναι για λόγους συμβατότητας ως προς τους παλαιότερους φυλλομετρητές.

Το **html** είναι το αρχικό στοιχείο (rootelement) για όλες τις σελίδες. Με αυτό ορίζονται διάφορες ιδιότητες του εγγράφου, όπως η γλώσσα ή το είδος της σελίδας καθώς και άλλα χαρακτηριστικά. Το **head** ορίζει την επικεφαλίδα και μπορεί να περιέχει ένα τίτλο με την ετικέτα **title**, όπως φαίνεται στο κώδικα, ή μεταδεδομένα και λοιπές πληροφορίες. Το **body** περιλαμβάνει το περιεχόμενο της σελίδας και μπορεί να περιέχει κείμενο, εικόνες, γραφικά και μια σειρά ετικετών ή οδηγιών.

Τα στοιχεία που αποτελούν τον «κορμό» δηλαδή τα html, head και body είναι προαιρετικά και αυτό αποτελεί μια «χαλαρή συμπεριφορά» της HTML5. Η HTML5 επίσης αγνοεί τα κεφαλαία/μικρά συνεπώς οι ετικέτες `<p></p>` και `<P></P>` είναι ακριβώς οι ίδιες.

Οι επικεφαλίδες που δεν έχουν εμφωλευμένο περιεχόμενο, μπορούν να παραλείψουν την κάθετοή να μπει αυτή και στις δύο άκρες. Συνεπώς οι ετικέτες `
`, `
`, `
` είναι ταυτόσημες και έχουν σαν αποτέλεσμα την αλλαγή γραμμής.

Μία άλλη αλλαγή είναι ότι οι ιδιότητες των ετικετών δεν χρειάζονται πια εισαγωγικά, εκτός αν δηλώνεται κάποιος ειδικός χαρακτήρας. Για παράδειγμα η ετικέτα: ``, χρησιμοποιεί εισαγωγικά στην ιδιότητα `alt` επειδή υπάρχει ο χαρακτήρας διάστημα, ενώ η ιδιότητα `src` δεν χρησιμοποιεί.

2.2.2. Μεταδεδομένα

Ενώ το στοιχείο **html** αποτελεί το ριζικό στοιχείο, τα επόμενα συστατικά της επικεφαλίδας αποτελούν τα μεταδεδομένα.

```
<!DOCTYPE html>
<html>
<head>
  <title>Τίτλος εγγράφου</title>
  <base href="http://www.example.com/index.html">
</head>
<body>
<p>Visit the <a href="otherpage.html">otherpage</a>.</p>
</body>
</html>
```

Τα μεταδεδομένα είναι:

- **head:** Η επικεφαλίδα
- **title:** Ο τίτλος της σελίδας
- **base:** Ορίζει την διεύθυνση του HTMLεγγράφου έτσι ώστε να χρησιμοποιηθεί σαν απόλυτη αναφορά και οι υπόλοιπες να σχετιστούν αναφορικά με αυτή. Στο παράδειγμα του κώδικα, η αναφορά στην «otherpage.html», θα γίνει αφού προστεθεί η διεύθυνση βάσης, δηλαδή `http://www.example.com/otherpage.html`.
- **link:** Χρησιμοποιείται για να επιτρέψει την σύνδεση άλλων πόρων στο τρέχων έγγραφο. Για παράδειγμα με τη ετικέτα **link** μπορούμε να αναφερθούμε σε ένα CSS αρχείο, που βρίσκεται σε άλλο κατάλογο.
- **meta:** Ορίζει τις πληροφορίες σχετικά με το κείμενο κυρίως για της μηχανές αναζήτησης, όπως λέξεις κλειδιά περιγραφή κλπ. Το κείμενο αυτό δεν είναι ορατό στο τελικό χρήστη. Χρησιμοποιείται κυρίως για τεχνικές βελτιστοποίησης για μηχανές αναζήτησης (SEO: SearchEngineOptimization).

2.2.3. Σενάρια - Αρχεία

Τα σενάρια χρησιμοποιούνται από τους κατασκευαστές για τη προσθήκη αλληλεπίδρασης στις σελίδες τους. Η ετικέτα **script** καλείται με το τύπο του σεναρίου που προστίθεται δυναμικά στη σελίδα. Μπορεί να είναι για παράδειγμα τύπου javascript. Η προσθήκη μπορεί να γίνει είτε με σύνδεσμο σε εξωτερικό αρχείο, είτε εσωτερικά του αρχείου που καλείται η ετικέτα **script**. Επίσης μια άλλη δυνατότητα είναι η εισαγωγή ομάδας δεδομένων. Σε αυτή την περίπτωση η ετικέτα είναι τύπου «text».

```
<script src="game-engine.js"></script>
<script type="text/x-game-map">
    .....U.....e
    o.....A.....e
    ....A....AAA.....e
    .A..AAA...AAAAA...e
</script>
```


2.2.4. Στοιχεία HTML5

Τα στοιχεία είναι οι ετικέτες που αφορούν την σήμανση ενός εγγράφου HTML5. Μπορούμε γενικότερα να τις κατατάξουμε σε δύο κατηγορίες. Ανάλογα με την εργασία που επιτελούν μπορούμε να τις χωρίσουμε σε διάφορες κατηγορίες, όπως κειμένου, φορμών, πινάκων κλπ. Οι περισσότερες ετικέτες προέρχονται από την HTML 4.01 και έχουν τον ίδιο ακριβώς σκοπό, ενώ κάποιες έχουν αποκτήσει και άλλες ιδιότητες.

Τα νέα στοιχεία της HTML5 παρουσιάζονται στον πίνακα που ακολουθεί, ανά κατηγορία χρήσης.

Πίνακας 0-1 Νέα στοιχεία της HTML 5

Κατηγορία	Στοιχεία
Στοιχεία για τη δόμηση της σελίδας	<article>, <aside>, <figcaption>, <figure>, <footer>, <header>, <hgroup>, <nav>, <section>, <details>, <summary>
Στοιχεία για το κείμενο	<mark>, <time>, <wbr>
Φόρμες και διαδραστικότητα	<datalist>, <keygen>, <meter>, <progress>, <command>, <menu>, <output>
Πολυμέσα και πρόσθετα	<audio>, <video>, <source>, <embed>
Canvas	<canvas>
Γλωσσική υποστήριξη	<bdo>, <rp>, <rt>, <ruby>

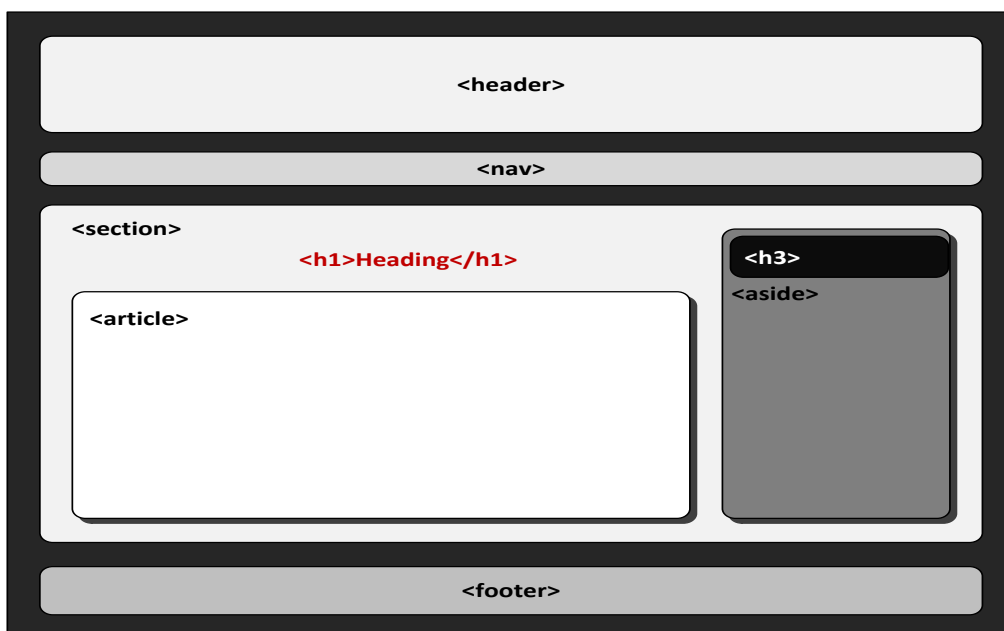
2.2.5. Ενότητες

Οι ενότητες έχουν να κάνουν με το σχέδιο που θα έχει η σελίδα. Αφορούν δηλαδή το επίπεδο της παρουσίασης, αλλά και της εσωτερικής δομής. Οι αντίστοιχες ενότητες χαρακτηρίζονται από τις ετικέτες:

- **body**: Αυτή η ενότητα μπορεί να ανταποκρίνεται σε διάφορα γεγονότα που συμβαίνουν και με τους αντίστοιχους χειριστές να δημιουργούνται οι αντίστοιχες ενέργειες.
- **article**: Αναπαριστά μία σύνθεση σε ένα έγγραφο, η οποία μπορεί να είναι ένα άρθρο, ένα σχόλιο, ένα νήμα ή κάποιο άλλο ανεξάρτητο στοιχείο του περιεχομένου.

- **section:** Ορίζει μία γενική ενότητα στο κείμενο. Χρησιμοποιείται κυρίως για την θεματική κατηγοριοποίηση του περιεχομένου.
- **nav:** Χρησιμοποιείται για τον ορισμό τμημάτων που έχουν εξωτερικούς ή εσωτερικούς συνδέσμους δηλαδή για πλοήγηση.
- **aside:** Είναι καινούργια ετικέτα και χρησιμοποιείται για την παράθεση κειμένου σχετικού με το κυρίως κείμενο, και βρίσκεται συνήθως σε πλευρική θέση.
- **hgroup:** Αποτελεί το συστατικό για των ορισμό των επικεφαλίδων μιας ενότητας. Η τάξη της επικεφαλίδας οριοθετείται μεταξύ h1-h6, με h1 το υψηλότερο επίπεδο και h6 το χαμηλότερο.
- **header:** Καθορίζει την επικεφαλίδα του εγγράφου.
- **footer:** Καθορίζει το υποσέλιδο του εγγράφου.
- **address:** Αποτελεί τα στοιχεία επικοινωνίας ανά ενότητα. Αν αυτό περιέχεται στο body τότε αντιστοιχεί για όλο το έγγραφο. Μπορεί όμως να οριστεί μέσα σε ένα section, οπότε ισχύει μόνο για αυτό.

Η εικόνα που ακολουθεί δείχνει ένα τυπικό σχέδιο, το οποίο έχει δημιουργηθεί μέσω των αντίστοιχων ετικετών, που αφορούν τις ενότητες.



Εικόνα 0-2HTML 5 - Σχέδιο εγγράφου

2.2.6. Ομαδοποίηση περιεχομένου

Σε αυτή την κατηγορία ανήκουν οι ετικέτες που αφορούν την οργάνωση τη παρουσίαση και τη διάταξη του κειμένου. Μπορούμε γενικότερα να τις κατατάξουμε σε δύο κατηγορίες. Στις ετικέτες που αφορούν γενικότερα τον τρόπο παρουσίασης του κειμένου, σε ετικέτες που αφορούν τον τρόπο οργάνωσης και σε ετικέτες που αφορούν λίστες.

2.2.6.1. Οργάνωση κειμένου

Οι αντίστοιχες ετικέτες αφορούν τον τρόπο που παρουσιάζεται το κείμενο στο τελικό χρήστη και είναι:

- **p**: Αναπαριστά μια παράγραφο κειμένου.
- **hr**: Χωρίζει θεματικά δύο ή περισσότερες παραγράφους.
- **pre**: Χρησιμοποιείται για την εμφάνιση κειμένου με ασυνήθιστη εμφάνιση, όπως για παράδειγμα κείμενο κώδικα.
- **blockquote**: Εμφανίζει μια ενότητα μέσα εισαγωγικά. Χρησιμοποιείται κυρίως για να καθορίσει ότι ένα τμήμα που αναφέρεται από άλλη πηγή.

Ο κώδικας που ακολουθεί δείχνει ένα παράδειγμα χρήσης των στοιχείων. Στο παράδειγμα επίσης φαίνεται ότι η χρήση του `blockquote` έχει αλλάξει σε σχέση με την HTML 4.01 που η χρήση του ήταν για την παράθεση ενός αποσπάσματος.

```
<body>
```

```
<p>
```

Αυτή είναι μια παράγραφος η οποία έχει δύο θεματικές ενότητες.

Η πρώτη θεματική ενότητα αφορά τη γλώσσα προγραμματισμού c.

Ο κώδικας που χρησιμοποιείται για την εκτύπωση του "Hello World" σε c είναι:

```
<pre>
```

```
printf ("Hello World");
```

```
</pre>
```

```
<hr/>
```

Η δεύτερη θεματική ενότητα αφορά την γλώσσα προγραμματισμού c++. Ο αντίστοιχος κώδικας σε c++ είναι:

```
<pre>
```

```
std::cout << "Hello World";
```

```
</pre>
```

```
</p>
```

```
<blockquote>
```

Σύμφωνα με τον ιστότοπο <http://www.java2s.com/Code/C/CatalogC.htm>, η printf μπορεί να χρησιμοποιηθεί και σε ένα πρόγραμμα c++!

```
</blockquote>
```

Το παράδειγμα έχει σαν έξοδο την εικόνα που ακολουθεί:

Αυτή είναι μια παράγραφος η οποία έχει δύο θεματικές ενότητες. Η πρώτη θεματική ενότητα αφορά τη γλώσσα προγραμματισμού c. Ο κώδικας που χρησιμοποιείται για την εκτύπωση του "Hello World" σε c είναι:

```
printf ("Hello World");
```

Η δεύτερη θεματική ενότητα αφορά την γλώσσα προγραμματισμού c++. Ο αντίστοιχος κώδικας σε c++ είναι:

```
std::cout << "Hello World";
```

Σύμφωνα με τον ιστότοπο <http://www.java2s.com/Code/C/CatalogC.htm>, η printf μπορεί να χρησιμοποιηθεί και σε ένα πρόγραμμα c++!

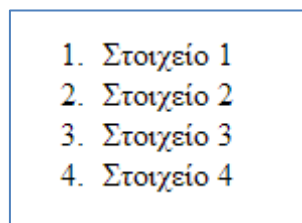
Εικόνα 0-3 Χρήση στοιχείων κειμένου

2.2.6.2. Λίστες

Οι λίστες περιέχουν ένα ή περισσότερα στοιχεία. Υπάρχουν οι αριθμημένες λίστες που δηλώνονται με την ετικέτα **ol** και οι μη αριθμημένες λίστες που δηλώνονται με την ετικέτα **ul**. Αντίστοιχα τα στοιχεία των λιστών περιέχονται στην ετικέτα **li**. Το παράδειγμα που ακολουθεί

```
<ol>
<li>Στοιχείο 1</li>
<li>Στοιχείο 2</li>
<li>Στοιχείο 3</li>
<li>Στοιχείο 4</li>
</ol>
```

έχει σαν αποτέλεσμα:



Εικόνα 0-4 Αριθμημένες λίστες

Εκτός από τις παραπάνω λίστες υπάρχουν και οι περιγραφικές λίστες. Αυτές ορίζονται με την ετικέτα **dl**. Οι ενότητες αυτών των λιστών αντίστοιχα με την ετικέτα

```
<dl>
<dt>Επικεφαλίδα 1</dt>
<dl>Στοιχείο 1.1</dl>
<dl>Στοιχείο 1.2</dl>
<dt>Επικεφαλίδα 2</dt>
<dl>Στοιχείο 2.1</dl>
<dl>Στοιχείο 2.1</dl>
</dl>
```

εμφανίζει:

Επικεφαλίδα 1
Στοιχείο 1.1
Στοιχείο 1.2
Επικεφαλίδα 2
Στοιχείο 2.1
Στοιχείο 2.1

Εικόνα 0-5 Περιγραφική λίστα

2.2.6.3. Άλλα στοιχεία

Στα στοιχεία ομαδοποίησης ανήκουν άλλες δύο ετικέτες. Ημία είναι η **div** που χρησιμοποιείται για τον ορισμό ενός τμήματος στο έγγραφο. Χρησιμεύει για το ορισμό ενός κοινού στυλ, στο τμήμα που ορίζει. Η άλλη είναι η **figure** που χρησιμοποιείται για την ομαδοποίηση διαφορετικών στοιχείων, όπως εικόνες, διαγράμματα, κείμενο, κώδικα κλπ. Ο τίτλος της ενότητας μπορεί να οριστεί επίσης με την ετικέτα **figcaption**.

2.2.7. Σημασιολογικά στοιχεία κειμένου

Τα στοιχεία αυτά, προσδίδουν την ανάλογη σημασία στο κείμενο σε σχέση πάντα με την ετικέτα που θα χρησιμοποιηθεί. Αυτές είναι:

- **code**: Εμφανίζει το κείμενο έτσι ώστε να μοιάζει με κώδικα μιας γλώσσας προγραμματισμού
- **em**: Εμφανίζει το κείμενο με πλάγιους και κάπως αραιά γραμμένους χαρακτήρες
- **dfn**: υποδηλώνει έναν όρο
- **samp**: Προσδιορίζει αποτέλεσμα ενός κώδικα
- **kbd**: υποδεικνύει το κείμενο που πρέπει να πληκτρολογήσει ο χρήστης
- **var**: μια μεταβλητή του κώδικα μιας γλώσσας προγραμματισμού
- **cite**: Παραθέτει μια αναφορά

Η χρήση και η έξοδος, των σημασιολογικών στοιχείων φαίνεται στο παράδειγμα που ακολουθεί.

```
<!DOCTYPE HTML>
<html lang="el">
<head>
<meta charset="utf-8" />
</head>
<body>
<p>
    Αναλύεται το ακόλουθο παράδειγμα κώδικα:
</p>
<hr/>
<code>
    chars[]=('H','e','l','l','o',' ','W','o','r','l','d');
    printf("%s",s);
</code>
<hr/>
Οι μεταβλητή <var>s</var>, είναι ένας πίνακας τύπου χαρακτήρα.
<hr/>
<em>
    Η ανάθεση της τιμής του αλφαριθμητικού μπορεί να γίνει απ' ευθείας με απόδοση
    όλων των στοιχείων κατά τη δήλωση του πίνακα!!
</em>
<hr/>
<dfn>
    Το %s, δηλώνει ότι η τιμή που θα εκτυπωθεί είναι αλφαριθμητικό.
</dfn>
<hr/>
Το αποτέλεσμα του κώδικα θα είναι:
<samp>
    Hello World
</samp>
<hr/>
<cite title="HTML & XHTML: The Definitive Guide. Published by O'Reilly Media, Inc.; fifth
edition (August 1, 2002)">
HTML & XHTML: The Definitive Guide. Published by O'Reilly Media, Inc.; fifth edition (August
1, 2002)</cite>
```

Αναλύεται το ακόλουθο παράδειγμα κώδικα:
<pre>char s[]=('H','e','l','l','o',' ','W','o','r','l','d'); printf("%s",s);</pre>
Οι μεταβλητή <i>s</i> , είναι ένας πίνακας τύπου χαρακτήρα.
<i>Η ανάθεση της τιμής του αλφαριθμητικού μπορεί να γίνει απ' ευθείας με απόδοση όλων των στοιχείων κατά τη δήλωση του πίνακα!!</i>
<i>Το %s, δηλώνει ότι η τιμή που θα εκτυπωθεί είναι αλφαριθμητικό.</i>
Το αποτέλεσμα του κώδικα θα είναι: Hello World
<i>HTML & XHTML: The Definitive Guide. Published by O'Reilly Media, Inc.; fifth edition (August 1, 2002)</i>

Εικόνα 0-6 Σημασιολογικά στοιχεία κειμένου

Παρατηρούμε ότι ενώ μπορούμε να παραθέσουμε μια αναφορά με τη `blockquote`, η `cite` προσδίδει σημασιολογικό χαρακτήρα στην αναφορά και έτσι θα αντιμετωπιστεί από τις μηχανές αναζήτησης.

2.2.8. Επεξεργασμένο κείμενο

Οι ετικέτες που χαρακτηρίζουν την κατάσταση ενός κειμένου είναι η **ins** και η **del**. Χρησιμοποιούνται για να περιγράψουν αλλαγές στο κείμενο. Η πρώτη χαρακτηρίζει το κείμενο προστιθέμενο και το εμφανίζει υπογραμμισμένο. Η δεύτερη το χαρακτηρίζει διορθωμένο και το εμφανίζει υπογεγραμμένο.

2.2.9. Ενσωματωμένο περιεχόμενο

Περιεχόμενο εκτός κειμένου, όπως εικόνες, βίντεο, αντικείμενα, μπορούν να ενσωματωθούν με την χρήση των αντίστοιχων ετικετών που είναι:

2.2.9.1. Εισαγωγή εικόνας

Η ετικέτα `img` χρησιμοποιείται για την εισαγωγή εικόνας σε μια σελίδα. Οι ιδιότητες που υποστηρίζει η ετικέτα είναι:

- **alt**: Εναλλακτικό κείμενο. Αν δεν βρεθεί η εικόνα τότε χρησιμοποιείται αυτό το κείμενο.
- **src**: Τοποθεσία της εικόνας.
- **width** , **height**: Το πλάτος και το ύψος της εικόνας.

Αντίστοιχα ο κώδικας για την εισαγωγή μιας εικόνας σε μια σελίδα είναι:

```

```

2.2.9.2. Εισαγωγή εσωτερικού πλαισίου .

Ο κώδικας:

```
<iframe src="somepage.htm" sandbox=""></iframe>
```

χρησιμοποιείται για την ενσωμάτωση ενός εγγράφου `html` σε ένα άλλο. Υποστηρίζει την ιδιότητα `sandbox`, που είναι νέα στην HTML 5 και παίρνει τις τιμές: `""`, `allow-same-origin`, `allow-top-navigation`, `allow-forms` και `allow-scripts`, οι οποίες καθορίζουν το επίπεδο αλληλεπίδρασης, αλλά και ασφάλειας μεταξύ του πλαισίου, και του αρχικού εγγράφου.

2.2.9.3. Ενσωμάτωση περιεχομένου

Οι ετικέτες αυτής της ομάδας επιτρέπουν την ενσωμάτωση εξωτερικού περιεχομένου όπως: εφαρμογή, αντικείμενο, βίντεο ή αρχείο ήχου. Αυτές είναι:

- **embed**: Επιτρέπει την ενσωμάτωση μιας εξωτερικής εφαρμογής.
- **object**: Ενσωματώνει ένα εξωτερικό αντικείμενο. Για τον ορισμό των παραμέτρων με τις οποίες θα κληθεί το αντικείμενο χρησιμοποιείται η `param`.
- **video**: Νέα ετικέτα στην HTML5, η οποία χρησιμοποιείται για την ενσωμάτωση ενός βίντεο.
- **audio**: Νέα ετικέτα στην HTML5, η οποία χρησιμοποιείται για την ενσωμάτωση ενός αρχείου ήχου.
- **map**: Δεν ενσωματώνει κάποιο περιεχόμενο, αλλά χρησιμοποιείται για να ορίσει μια περιοχή σε μια εικόνα, στην οποία θα υπάρχουν σύνδεσμοι. Για τον καθορισμό των συντεταγμένων που θα έχει ο κάθε σύνδεσμος χρησιμοποιείται η ετικέτα **area**.

Οι ετικέτες `video`, `audio` χρησιμοποιούν την ιδιότητα `source`, με την οποία μπορούν να οριστούν εναλλακτικοί πόροι για συγκεκριμένα αντικείμενα. Για παράδειγμα μπορεί να ορίσει τους `codecs` που θα χρησιμοποιηθούν για την αναπαραγωγή ενός αρχείου ήχου, όπως φαίνεται και στον κώδικα που ακολουθεί.

```
<audio controls>
<source src="example.ogg" type="audio/ogg">
<source src="example.mp3" type="audio/mpeg">
Ο περιηγητής δεν υποστηρίζει αυτό τον τύπο.
</audio>
```

Στη περίπτωση αυτή αν δεν υπάρχει κανένας από τους δύο κωδικοποιητές ήχου (`ogg`, `mp3`) στο σύστημα του χρήστη τότε θα εμφανιστεί το μήνυμα ότι ο περιηγητής δεν υποστηρίζει αυτό το τύπο αρχείου.

Εδώ επίσης θα πρέπει να αναφέρουμε ότι η HTML5 έχει μεθόδους DOM, ιδιότητες και συμβάντα για τα στοιχεία `video`, `audio`. Αυτοί οι μέθοδοι περιλαμβάνουν την: αναπαραγωγή, παύση, εναλλαγή και οι ιδιότητες αντίστοιχα: την διάρκεια, την ένταση κλπ.

Η προσπέλαση αυτών των μεθόδων μπορεί να γίνει μέσω του DOM με τη χρήση Javascript. Ένα τέτοιο παράδειγμα αποτελεί ο κώδικας που ακολουθεί, και επιδεικνύει την χρήση των μεθόδων play(), pause().

```
<!DOCTYPE html>
<html>
<body>

<div style="text-align:center">
<button onclick="playPause()">Play/Pause</button>
<br>
<video id="video1" width="420">
<source src="mov_bbb.mp4" type="video/mp4">
<source src="mov_bbb.ogv" type="video/ogg">
Δεν υποστηρίζεταιHTML5 video.
</video>
</div>

<script>
var myVideo=document.getElementById("video1");

function playPause()
{
  if (myVideo.paused)
    myVideo.play();
  else
    myVideo.pause();
}
</script>
</body>
</html>
```

2.2.10. Canvas

Ένα άλλο νέο στοιχείο ενσωμάτωσης περιεχομένου, ορίζεται με το στοιχείο **canvas**. Αυτή η ετικέτα δημιουργεί μια ορθογώνια περιοχή που μπορεί να χρησιμοποιηθεί για τον σχεδιασμό γραφικών όπως κύκλους, ορθογώνια και διάφορα σχέδια. Επίσης μπορούν να δημιουργηθούν μια σειρά από οπτικά εφέ, αλλά και εφέ μετασχηματισμού.

Οκαμβάζειναι μια περιοχή γραφικών άμεσης λειτουργίας, η οποία μπορεί να ελεγχθεί με τη χρήση JavaScript. Ο όρος άμεση λειτουργία αναφέρεται στον τρόπο με τον οποίο ο καμβάς επανασχεδιάζει από την αρχή κάθε πλαίσιο της οθόνης (frame), χρησιμοποιώντας API κλήσεις μέσω της JavaScript. Ο δημιουργός θα πρέπει να ρυθμίζει τη οθόνη πριν τον σχεδιασμό κάθε πλαισίου, έτσι ώστε να εμφανίζονται τα σωστά εικονοστοιχεία. Αυτή η τεχνολογία διαφέρει από τις άλλες όπως Flash, Silverlight που λειτουργούν σε κατάσταση έκτακτης λειτουργίας. Σε αυτή την

λειτουργία, μία λίστα με αντικείμενα διατηρούνται από τη μηχανή γραφικών και αυτά εμφανίζονται σύμφωνα με τις ιδιότητες που έχουν (θέση x,y, χρώμα, διαφάνεια κλπ). Αυτό έχει σαν πλεονέκτημα ο δημιουργός να αποφεύγει τις χαμηλού επιπέδου λειτουργίες, από την άλλη όμως περιορίζει τον λεπτομερή χειρισμό αυτών των αντικειμένων(Fulton & Fulton, 2011).

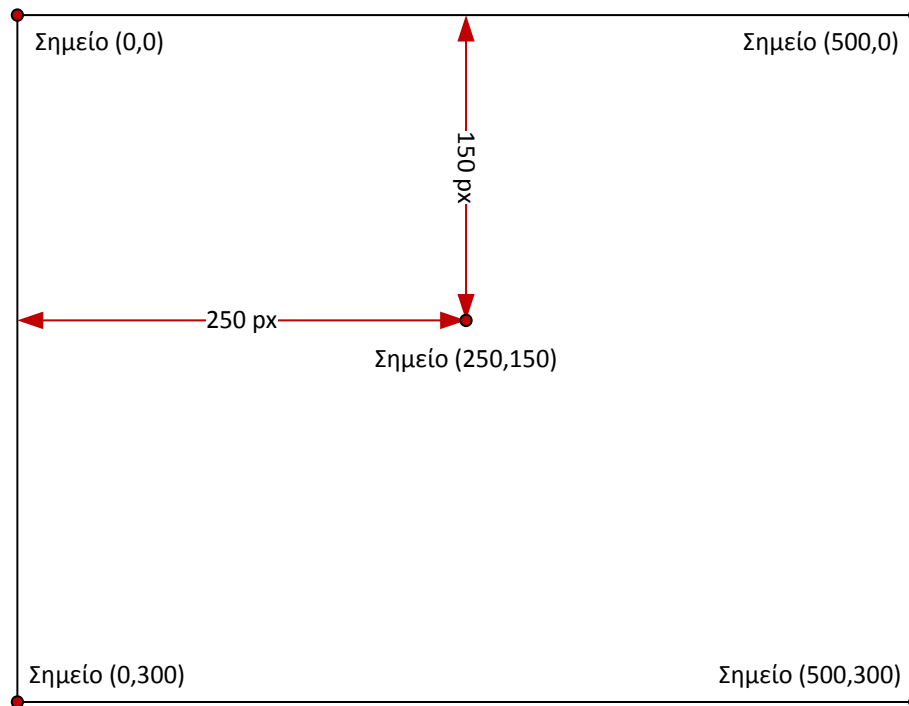
Ο βασικός καμβάς στην HTML5 περιλαμβάνει ένα 2Dπλαίσιο που επιτρέπει την δημιουργία ποικίλων σχημάτων, την εξομάλυνση κειμένου (rendertext) και την εμφάνιση εικόνων απ' ευθείας στη καθορισμένη περιοχή. Υποστηρίζεται επίσης μια σειρά λειτουργιών όπως εφαρμογή χρώματος, περιστροφές, διαφάνειες, έλεγχο εικονοστοιχείων.

Εκτός όμως από την εμφάνιση και εξομάλυνση, μια σειρά λειτουργιών όπως απόκριση σε γεγονότα εισόδων του χρήστη, χρονοιστές, αντικείμενα, κλάσεις, υποστήριξη ήχου, μαθηματικές συναρτήσεις επιτρέπουν τη δημιουργία ολοκληρωμένων εφαρμογών.

Για την εμφάνιση σχεδίων ή εφαρμογών σε canvas, χρειάζεται η δημιουργίαJavaScript κώδικα. Πρώτα απ' όλα θα πρέπει να χρησιμοποιηθεί η `document.getElementById("drawingCanvas")`, αφού η μέθοδος `getElementById()` χρησιμοποιείται για την προσπέλαση οποιουδήποτε στοιχείου της σελίδας. Στη συνέχεια ο κώδικας χρησιμοποιεί την μέθοδο `getContext()`, για να ανακτήσει ένα δυσδιάστατο περιεχόμενο σχεδίασης με την εντολή `varcontext = canvas.getContext("2d");`

Κάθε canvasξεκινά με ένα άδειο παραλληλόγραμμο, σε μια θέση στη σελίδα που θα ορίσουμε. Μπορούμε στη συνέχεια να διαχειριστούμε αυτό το αντικείμενο σε όποιο σημείο θελήσουμε: για παράδειγμα, όταν η σελίδα φορτώνεται, όταν ο επισκέπτης πατήσει κάποιο κουμπί ή με κάποιο άλλο γεγονός.

Ο canvasπου δημιουργείται χρησιμοποιεί ένα σύστημα συντεταγμένων για τον καθορισμό των σημείων του περιεχομένου. Το σύστημα αυτό παρουσιάζεται στην επόμενη εικόνα, όπου φαίνονται τα σημεία σε μία ορθογώνια περιοχή μεγέθους 500x300 pixel(MacDonald, 2011).



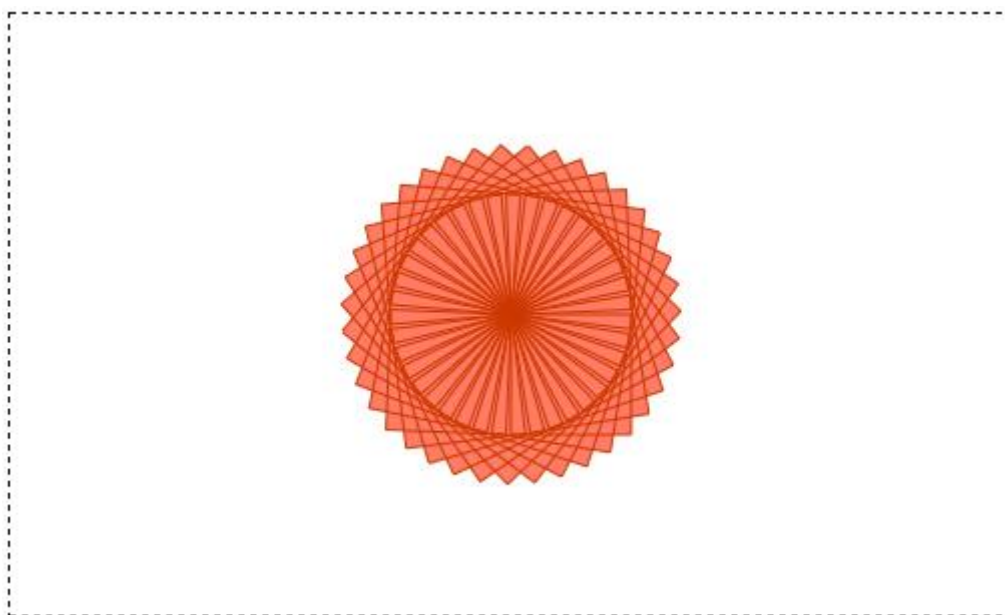
Εικόνα 0-7 Σύστημα συντεταγμένων canvas

Τα στοιχεία βασικής σχεδίασης σε ένα καμβά είναι:

- **Ευθείες γραμμές:** Το πιο απλό σχέδιο είναι οι ευθείες γραμμές. Χρειάζονται τρεις ενέργειες για αυτό. Η μέθοδος `moveTo()` που μετακινεί το δείκτη στο σημείο από το οποίο θα ξεκινήσει η γραμμή, τη μέθοδο `lineTo()`, για την σχεδίαση μιας γραμμής από το σημείο που βρισκόμαστε στο σημείο που ορίζεται στη `lineTo()` και τέλος η `stroke()` για την εμφάνιση της γραμμής.
- **Μονοπάτια και σχήματα:** Για τα σχήματα, χρησιμοποιείται πάλι η μέθοδος `lineTo()`, με τη μόνη διαφορά ότι ορίζεται ένα μονοπάτι από γραμμές για τον σχεδιασμό. Η δημιουργία σχημάτων θα μπορούσε να γίνει και χωρίς μονοπάτι, αλλά με αυτό τον τρόπο δεν υπάρχει δυνατότητα «ορισμού» του σχήματος και περεταίρω επεξεργασίας, όπως για παράδειγμα ο χρωματισμός του. Το μονοπάτι ξεκινά με την μέθοδο `beginPath()` και κλείνει αντίστοιχα με την `closePath()`.
- **Καμπύλες γραμμές:** Ο σχεδιασμός καμπύλων σχημάτων, μπορεί να επιτευχθεί με τη χρήση των: `arc()`, `arcTo()`, `bezierCurveTo()` και `quadraticCurveTo()`. Άλλες αυτές οι μέθοδοι, ζωγραφίζουν καμπύλες γραμμές με διαφορετικό τρόπο χρησιμοποιώντας μαθηματικούς υπολογισμούς.

Αντίστοιχες μέθοδοι χρησιμοποιούνται για άλλες παραμέτρους όπως:

- **Χρωματισμός:** Για τον χρωματισμό της γραμμής που χαράσσεται χρησιμοποιείται η μέθοδος `strokeStyle()`, η οποία ορίζει ένα χρώμα μορφής `rgb`. Αντίστοιχα για το «γέμισμα» ενός σχήματος χρησιμοποιείται η `fillStyle()` σε συνδυασμό με την `fill()`.
- **Διαφάνεια:** Για την διαφάνεια ενός σχήματος, χρησιμοποιείται πάλι η `fillStyle()`, αλλά το χρώμα δεν ορίζεται με την `rgb`, αλλά με την `rgba` που παίρνει τέσσερις παραμέτρους. Οι τρεις κοινές με την `rgb` και αφορούν την ποσόστωση των τριών βασικών χρωμάτων, ενώ η τέταρτη αφορά το άλφα κανάλι και παίρνει τιμές από 0..1.
- **Μετασχηματισμός:** Με τον μετασχηματισμό, ουσιαστικά αλλάζουμε το σύστημα συντεταγμένων του καμβά. Ο μετασχηματισμός χρησιμοποιείται σε περίπτωση που θέλουμε να αποδώσουμε κίνηση σε ένα σχέδιο, χωρίς να το επανασχεδιάζουμε κάθε φορά σε καινούργια θέση. Ειδικά αν το σχέδιο αποτελείται από πολλά σχήματα αυτό είναι ιδιαίτερα βολικό. Για τον μετασχηματισμό χρησιμοποιείται η μέθοδος `translate()`.



Εικόνα 0-8 Παράδειγμα σχεδιασμού σε καμβά

Η εικόνα 2-10 είναι ένα σχήμα που έχει προκύψει από τον ακόλουθο κώδικα που περιέχει τα στοιχεία που αναφέραμε. Ο κώδικας χρησιμοποιεί τη μέθοδο `rotate`, για την περιστροφή των συντεταγμένων κάθε φορά που σχεδιάζεται ένα τετράγωνο. Εφόσον μια πλήρη περιστροφή (ένας κύκλος) ορίζεται σε 2 π , αντίστοιχα σε κάθε

πέρασμα του βρόγχου περιστρέφουμε το σύστημα κατά 2 xpi / copies, όπου copies ο αριθμός των τετραγώνων.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Canvas</title>
<style>
canvas {border: 1px dashed black;}
</style>
<script>
window.onload = function() {
  var canvas = document.getElementById("drawingCanvas");
  var context = canvas.getContext("2d"); context.translate(250, 150);
  // Draw 10 squares.
  var copies = 40;
  for (var i=1; i<copies; i++) {
    context.rotate(2 * Math.PI * 1/(copies-1));
    // Draw the square.
    context.rect(0, 0, 60, 60);
    context.strokeStyle = "rgb(205,55,0)";
    context.fillStyle = "rgba(255,36,0,0.6)";
  }
  context.fill();
  context.stroke();
};
</script>
</head>
<body>
<canvas id="drawingCanvas" width="500" height="300"></canvas>
</body>
</html>
```

2.2.11. Φόρμες

Μια φόρμα HTML, αποτελείται από στοιχεία εισόδου όπως: πεδία κειμένου, κουμπιά επιλογής, λίστες, κουμπιά αποστολής στοιχείων αλλά και άλλα. Οι φόρμες χρησιμοποιούνται για την αποστολή στοιχείων στον εξυπηρετητή, για αποθήκευση και μελλοντική επεξεργασία.

Οι φόρμες αποτελούν στοιχείο της HTML, από το ξεκίνημά της. Πολλά εργαλεία τρίτων κατασκευαστών όπως το JQuery, προσπάθησαν να δώσουν στις φόρμες μια πιο επαγγελματική όψη. Στην HTML5 όμως προστέθηκαν μερικά χαρακτηριστικά για τη δημιουργία πιο επαγγελματικών φορμών(Harris, 2011).

Τα αντίστοιχα στοιχεία που χρησιμοποιούνται στις φόρμες είναι:

- **datalist:** Επιτρέπει σε έναν προγραμματιστή να επισυνάψει μια λίστα από προτεινόμενες τιμές σε μια λίστα κειμένου. Μόλις ξεκινήσει η πληκτρολόγηση τότε εμφανίζεται μια αναδιπλούμενη λίστα με τις αντίστοιχες προτάσεις και ο χρήστης μπορεί να επιλέξει κάποια από αυτές.
- **fieldset:** Χρησιμοποιείται για την ομαδοποίηση των στοιχείων μιας φόρμας. Για την προσθήκη τίτλου στα ομαδοποιημένα στοιχεία της φόρμας χρησιμοποιείται η ετικέτα **<legend>**.
- **keygen:** Δημιουργεί ένα κλειδί, με το οποίο γίνεται κωδικοποίηση των δεδομένων για να μεταφερθούν με ασφάλεια στον εξυπηρετητή.
- **label:** Χρησιμοποιείται για την αντιστοίχιση ενός στοιχείου HTML σε ένα στοιχείο φόρμας. Με αυτό τον τρόπο μπορούν να εφαρμοστούν στυλ εμφάνισης στο στοιχείο της φόρμας μέσω ενός CSSαρχείου.
- **meter:** Με την ετικέτα αυτή μπορεί να οριοθετηθεί ένα αριθμητικό πεδίο. Για την οριοθέτηση χρησιμοποιεί ιδιότητες όπως η μέγιστη (max) και η ελάχιστη (min) τιμή. Επίσης μπορεί να οριστεί μια βέλτιστη τιμή (optimum).
- **output:** Περιγράφει μια ενότητα η οποία μπορεί να αλλαχτεί με τη χρήση ενός σεναρίου, όπως javascript. Αν για παράδειγμα αλλάξει η τιμή ενός πεδίου, τότε μπορεί να εμφανίζεται ένα μήνυμα δίπλα στο πεδίο ότι η τιμή άλλαξε.
- **progress:** Μετράει την πρόοδο που έχει γίνει στη συμπλήρωση μιας φόρμας. Όταν ένα στοιχείο συμπληρώνεται, τότε μπορείτε να καλείται ένα σενάριο javascript και να αυξάνει η ιδιότητα value ανάλογα με το ποσοστό ολοκλήρωσης.

Μαζί με τα νέα στοιχεία, δημιουργήθηκαν ή επαναπροσδιορίστηκαν νέες και παλιές ιδιότητες όπως:

- **autofocus:** Μπορεί να εφαρμοστεί σε οποιοδήποτε στοιχείο της φόρμας, το οποίο θα είναι το πρώτο στοιχείο που θα επικεντρωθεί η είσοδος για τον χρήστη. Συνήθως αφορά το πρώτο στοιχείο σε μια φόρμα.
- **pattern:** Χρησιμοποιεί μια κανονική έκφραση για να ταιριάζει την είσοδο του χρήστη. Επίσης πρέπει να οριστεί και ένα τίτλος ο οποίος αποτελεί μια συμβουλή προς τον χρήστη (tip).
- **placeholder:** Χρησιμοποιείται αντί του **<label>**, για να χαρακτηρίσει ένα πεδίο κειμένου. Όταν κάποιος εστιάσει στο πεδίο τότε το κείμενο του placeholder εξαφανίζεται.

- **required:** Ορίζει αν είναι υποχρεωτική η συμπλήρωση ενός πεδίου.

Στην HTML5 έχουν οριστεί και μια σειρά τύπων για τα πεδία μιας φόρμας όπως:

- **color:** Επιτρέπει στον χρήστη να επιλέξει ένα χρώμα, σε δεκαεξαδική μορφή.
- **date:** Ορίζει ότι η τιμή θα είναι τύπου ημερομηνίας.
- **datetime:** Η τιμή θα είναι σε μορφή ημερομηνίας και ώρας.
- **time:** Αντίστοιχα θα πρέπει να γίνει εισαγωγή ώρας.
- **email:** Το πεδίο αφορά την ηλεκτρονική διεύθυνση ενός χρήστη.
- **number.** Το πεδίο είναι αριθμητικό
- **search:** Στο πεδίο αυτό θα εισαχθεί ένα αλφαριθμητικό για αναζήτηση.
- **tel:** Αφορά τηλεφωνικό αριθμό.
- **url:** Αποτελεί διαδικτυακή διεύθυνση.

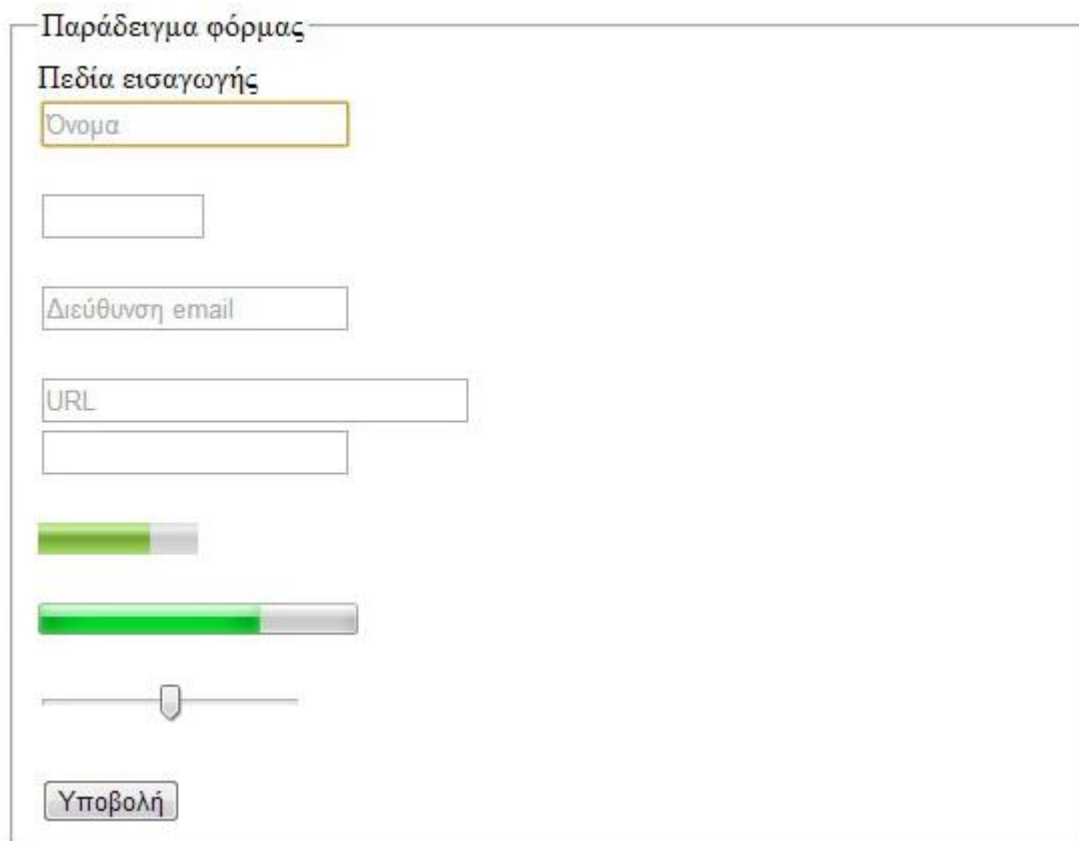
Ένα απλό παράδειγμα δημιουργίας μιας φόρμας μας δίνει ο κώδικας που ακολουθεί, ενώ τα αντίστοιχα αποτελέσματα φαίνονται στην εικόνα 2-3.

```

<!DOCTYPE HTML>
<html lang = "el">
<head>
<meta charset = "UTF-8" />
</head>
<body>
<form action = "">
<fieldset>
  <legend>Παράδειγμα φόρμας</legend>
  <caption>Πεδία εισαγωγής</caption><br>
  <input type="text" size="20" autofocus required placeholder="Όνομα"/><br><br>
  <input type="text" size="8" pattern="\d{3}-\d{4}" title="ddd-dddd"/><br><br>
  <input type="email" size="20" name="email" required placeholder="Διεύθυνση
email"/><br><br>
  <input type="url" size="30" name="url" placeholder="URL"/><br>
  <input type="text" name="choices" list="data-list">
  <datalist id="data-list">
    <option value="Επιλογή 1">
    <option value="Επιλογή 2">
    <option value="Επιλογή 3">
    <option value="Επιλογή 4">
    <option value="Επιλογή 5">
  </datalist><br><br>
  <meter min="0" max="10" value="7"></meter><br><br>
  <progress max="100" value="70">30%</progress><br><br>
  <input type="range" id="range" name="range"><br><br>
  <input type="submit"/>
</fieldset>
</form>
</body>
</html>

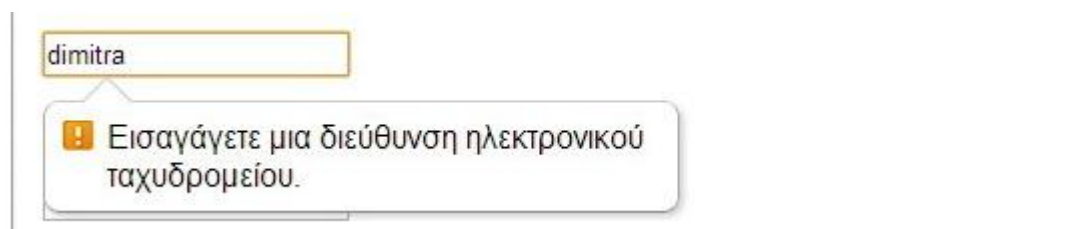
```

Η πρώτη γραμμή «formaction» καθορίζει τον τρόπο με τον οποίο θα αποσταλούν τα δεδομένα στον εξυπηρετητή (POST ή GET). Επίσης η φόρμα μας έχει εστιάσει στο πεδίο όνομα, επειδή έχει χρησιμοποιηθεί η ιδιότητα autofocus. Μετά τα πεδία έχουμε δημιουργήσει αντίστοιχα αριθμητικό πεδίο (meter), μπάρα προόδου (progress) και πεδίο ορίου (range).



Εικόνα 0-9 Φόρμα HTML5

Τα πεδία που ακολουθούν έχουν δηλωθεί “required”, συνεπώς αν δεν συμπληρωθούν θα βγει το αντίστοιχο μήνυμα λάθους, αν βέβαια υποστηρίζει το λογισμικό πλοήγησης αυτή την ιδιότητα. Βλέπουμε ότι στην περίπτωση που πληκτρολογήσουμε λάθος διεύθυνση email, η υποβολή της φόρμας δεν γίνεται και εμφανίζεται αντίστοιχο μήνυμα προειδοποίησης χωρίς να γράψουμε καθόλου κώδικα.



Εικόνα 0-10 - Φόρμες μηνύματα λάθους

2.2.12. Πίνακες

Το μοντέλο πίνακα στην HTML γενικότερα αποτελείται από τα στοιχεία του πίνακα και τις ιδιότητες αυτών των στοιχείων. Ο πίνακας αποτελείται από κελιά στοιχισμένα σε ένα δυσδιάστατο πλέγμα με θέσεις και συντεταγμένες (x,y). Το πλέγμα είναι πεπερασμένο και μπορεί να είναι άδειο ή να έχει μία ή περισσότερες θέσεις. Τα κελιά μπορούν να είναι κελιά δεδομένων είτε κελιά επικεφαλίδων. Τα κελιά δεδομένων αντιστοιχούν στα στοιχεία **td** και τα κελιά επικεφαλίδων στα στοιχεία **th**.

Μια γραμμή είναι ένα σύνολο θέσεων για μια συγκεκριμένη τιμή του y. Οι γραμμές αντιστοιχούν σε **tr** στοιχεία. Μία στήλη είναι ένα σύνολο θέσεων για μια συγκεκριμένη τιμή του x. Αντίστοιχα μπορούν να ορισθούν και οι ομάδες γραμμών και στηλών. Άλλες ετικέτες που αφορούν τους πίνακες είναι:

- **thead**: Ορίζουμε επικεφαλίδα
- **tfoot**: Ορίζουμε υποσέλιδο
- **tbody**: ορίζουμε τις γραμμές που αποτελούν το κυρίως περιεχόμενο του πίνακα.
- **caption**: Ορίζει μια περιγραφή για τον πίνακα

Εξίσου σημαντικές όμως είναι και οι ιδιότητες των **td** και **th** στοιχείων όπως η **colspan** της **th** που χρησιμοποιείται για την οριζόντια συγχώνευση κελιών και αντίστοιχα η **rowspan** της **td** για την κατακόρυφη συγχώνευση κελιών.

Η χρήση των πινάκων γενικότερα μπορεί να κατηγοριοποιηθεί σε δύο κατηγορίες:

Η πρώτη κατηγορία που θέλει τον πίνακα, να χρησιμοποιείται ως στοιχείο παρουσίασης, δηλαδή ενός τρόπου στοιχειοθέτησης των δεδομένων. Βέβαια σε αυτή τη περίπτωση είναι προτιμότερο να χρησιμοποιείται η ετικέτα **div**, γιατί με αυτό τον τρόπο η δομή ανακτάται γρηγορότερα από το σύστημα πλοήγησης. Η χρήση του πίνακα αντί της **div**, γίνεται χρησιμοποιώντας μόνο τις ετικέτες **<table>**, **<tr>** και **<td>**.

Αντίθετα η δεύτερη κατηγορία χρησιμοποιεί και άλλα στοιχεία σήμανσης για την παρουσίαση δεδομένων στον τελικό χρήστη.

Ο κώδικας που ακολουθεί

```

<table><caption>Παράδειγμα HTML5 πίνακα</caption>
  <thead>
    <tr>
      <th>ΣτήληΑ</th>
      <th>ΣτήληΒ</th>
      <th>ΣτήληΓ</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="rowgroup" colspan="3">Πρώτηομάδα</th>
    </tr>
    <tr>
      <th>Γραμμή 1</th>
      <td>ΚελίΒ1 </td>
      <td>ΚελίΓ1 </td>
    </tr>
    <tr>
      <th>Γραμμή 2</th>
      <td>ΚελίΒ2</td>
      <td>ΚελίΓ2</td>
    </tr>
  </tbody>
  <tbody>
    <tr>
      <th scope="rowgroup" colspan="3">Δεύτερηομάδα</th>
    </tr>
    <tr>
      <th>Γραμμή 1</th>
      <td>ΚελίΒ1</td>
      <td>ΚελίΓ1</td>
    </tr>
    <tr>
      <th>Γραμμή 2</th>
      <td>ΚελίΒ2</td>
      <td>ΚελίΓ2</td>
    </tr>
  </tbody>
</table>

```

μας δίνει τον πίνακα

Παράδειγμα HTML5 πίνακα		
Στήλη Α	Στήλη Β	Στήλη C
Πρώτη ομάδα		
Γραμμή 1	Κελί Β1	Κελί Γ1
Γραμμή 2	Κελί Β2	Κελί Γ2
Δεύτερη ομάδα		
Γραμμή 1	Κελί Β1	Κελί Γ1
Γραμμή 2	Κελί Β2	Κελί Γ2

2.2.13. Σύνδεσμοι

Οι σύνδεσμοι σχηματίζονται με τα στοιχεία `a`, `area` και `link` και αναπαριστούν μια σύνδεση μεταξύ δύο πόρων.

Με την ετικέτα `<a>` δημιουργείται ένας σύνδεσμος προς μια άλλη σελίδα, μέσω της ιδιότητας `href`. Αντίστοιχα με την ετικέτα `<area>` ορίζεται μια περιοχή σε μια εικόνα ως σύνδεσμος σε συνδυασμό με την ετικέτα `map`. Με την `<link>` ορίζεται η σχέση μεταξύ του τρέχοντος εγγράφου και ενός εξωτερικού αρχείου. Χρησιμοποιείται κυρίως όπως είδαμε για την προσθήκη κάποιου αρχείου CSS ή Javascript.

Οι ιδιότητες των συνδέσμων φαίνονται στον πίνακα που ακολουθεί:

Ιδιότητα	Τιμή	Περιγραφή
href	URL	Ορίζει το URL του εγγράφου
hreflang	κωδικός γλώσσας	Ορίζει την γλώσσα του εγγράφου που δείχνει το URL της ιδιότητας <i>href</i>
media	all, braille, print, projection, screen, speech	Ορίζει την συσκευή (οθόνη, εκτυπωτής, προτζέκτορας, ομιλία) στην οποία θα εμφανιστεί το περιεχόμενο του εγγράφου
rel	alternate, archives, author, first, help, icon, index, last, license, next pingback, prefetch prev, search, stylesheet, sidebar, tag, up	Ορίζει την σχέση μεταξύ του τρέχον εγγράφου με το έγγραφο που δείχνει το URL της ιδιότητας <i>href</i>
sizes	αριθμός any	Ορίζει τα μεγέθη των εξωτερικών πηγών. Χρησιμοποιείται μόνο όταν rel="icon"
type	τύπος αρχείου, όπως: text/css text/javascript image/gif	Ορίζει τον τύπο του περιεχομένου του εγγράφου που δείχνει το URL της ιδιότητας <i>href</i>

Τα αντίστοιχα παραδείγματα για τις ετικέτες παρουσιάζονται στον κώδικα που ακολουθεί:

```
<!DOCTYPE HTML>
<html lang="el">
<head>
  <meta charset="utf-8" />
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  Σύνδεσμος στο in.gr: <br /><a href="http://www.in.gr" target="_blank" >in.gr σύνδεσμος</a>
</body>
</html>
```

Στην τέταρτη γραμμή ορίζεται το αρχείο στυλ για την εμφάνιση. Ενώ στο body ορίζουμε έναν υπερσύνδεσμο σε έναν εξωτερικό ιστότοπο. Το αποτέλεσμα που λαμβάνουμε είναι:

Σύνδεσμος προς το in.gr:
[in.gr Σύνδεσμος](#)

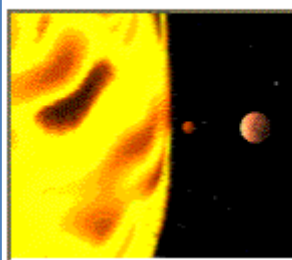
Εικόνα 0-11 Αποτέλεσμα χρήσης href

Αντίστοιχα η ετικέτα area χρησιμοποιείται ως εξής:

```
<!DOCTYPE HTML>
<html lang="el">
<head>
  <meta charset="utf-8" />
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <p>Επέλεξε έναν από τα τρία ουράνια σώματα:</p>
  
  <map name="planetmap">
    <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm" />
    <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm" />
    <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm" />
  </map>
</body>
</html>
```

Το αποτέλεσμα θα είναι η εικόνα που ακολουθεί, η οποία έχει τμηματοποιηθεί με την χρησιμοποίηση των ιδιοτήτων shape, coords. Έτσι η αν ο χρήστης πατήσει στο δεξιό ουράνιο σώμα θα ακολουθήσει τον σύνδεσμο που περιγράφεται από την αναφορά «venus.htm».

Επέλεξε έναν από τα τρία ουράνια σώματα:



Εικόνα 0-12 Αποτέλεσμα χρήσης της area, map

Διεπαφές προγραμματισμού (APIs)

Η διεπαφή προγραμματισμού εφαρμογών, ή αλλιώς διασύνδεση προγραμματισμού εφαρμογών (APIs: ApplicationProgrammingInterface), παρέχει τις απαραίτητες βιβλιοθήκες προκειμένου να εκτελούνται λειτουργίες ή ανταλλαγή δεδομένων. Αυτές οι βιβλιοθήκες αποτελούν την διεπαφή για τις κλήσεις συστήματος με τέτοιο τρόπο, ώστε να αποκρύπτονται οι χαμηλού επιπέδου λεπτομέρειες από τον προγραμματιστή, με στόχο την ασφάλεια και την προστασία του συστήματος.

Η HTML5 έχει πρόσβαση σε αντίστοιχες διεπαφές μέσω javascript. Τέτοιες διεπαφές είναι:

- Communication APIs
- FileSystem APIs
- Geolocation APIs

2.3.1. Communication APIs

Όλες οι εφαρμογές WEB, που εκτελούνται, επηρεάζονται από έναν μηχανισμό ασφαλείας του φυλλομετρητή. Αυτό σημαίνει ότι ο φυλλομετρητής δεν επιτρέπει την εκτέλεση ύποπτου κώδικα που θα μπορούσε να βλάψει το τοπικό μηχάνημα. Για αυτό τον λόγο άλλωστε, οι εφαρμογές WEB, δεν μπορούν να δημιουργήσουν, τροποποιήσουν ή να διαβάζουν αρχεία από το τοπικό σύστημα. Επίσης δεν μπορούν να έχουν πρόσβαση ή να εκτελέσουν σενάρια σε σελίδες ή τοποθεσίες που δεν χρησιμοποιούν το ίδιο πρωτόκολλο, αριθμό πόρτα και όνομα τομέα (Freeman, 2011).

Αυτό ο περιορισμός στην HTML5 μπορεί να αρθεί με συγκεκριμένες τεχνικέςόπως:

2.3.1.1. XHR&XHR2 με CORS

ΤοXHRείναι ένα API, που υποστηρίζει σύγχρονες αιτήσεις, δηλαδή η εκτέλεση του κώδικα γίνεται μετά από την απάντηση του καλούντα. Το XHR2 υποστηρίζει επιπλέον τον χειρισμό της επιφόρτωσης (upload) και της μεταφόρτωσης (download) μέσω ενός συμβάντος προόδου.ΤοXHR2 με CORS (CORS: Cross-OriginResourceSharing), επιτρέπει να γίνει το αίτημα μόνο όταν ο εξυπηρετητής το επιτρέπει.

Το κομμάτι του κώδικα για την πλευρά του πελάτη (client) είναι:

```
var client = new XMLHttpRequest();
client.onreadystatechange = function () {
  if (this.readyState == 4 && this.status == 200) {
    alert(Δοκιμήxhr2 μεcors: ' + this.responseText);
  }
};
client.open('GET', 'http://example.com/cors.php');
client.send();
```

Αν ο εξυπηρετητής απαντήσει με μία επικεφαλίδα CORS τότε το μήνυμα θα μεταδοθεί κανονικά. Βέβαια για να δοθεί η απάντηση θα πρέπει να υπάρχει και ο αντίστοιχος κώδικας στον εξυπηρετητή, δηλαδή στην διεύθυνση <http://www.example.com/cors.php>

```
<?php
header('Access-Control-Allow-Origin: *');
?>
@απάντηση
```

2.3.1.2. WebMessaging

Σε αυτή την περίπτωση χρησιμοποιείται ένα παλαιότερο API, το `postMessage` με το οποίο επιτρέπεται η αποστολή αλφαριθμητικών (μηνυμάτων), σε διαφορετικούς τομείς και προελεύσεις (domains and origins).

Για παράδειγμα μία σελίδα, που δέχεται γεγονότα `onmessage`, μπορεί να περιλαμβάνει τον ακόλουθο κώδικα:

```
window.onmessage = function (event) {
  if (event.origin == 'mytrustedsite.com') {
    alert('my trusted site said: ' + event.data);
  }
};
```

Με την χρήση ενός `<iframe>`, που περιέχει κώδικα και χρησιμοποιεί τον `<iframe>` κόμβο DOM (DOMnode), μπορούμε να θέσουμε στο `<iframe>`:

```
iframe.contentWindow.postMessage("γειαεκεί", location.host);
```

Αυτό μας δίνει την δυνατότητα όπως είδαμε αποστολής του αλφαριθμητικού “γεια εκεί”, ανάμεσα σε τομείς εμπιστοσύνης.

2.3.1.3. WebSockets

Χρησιμοποιείται για την αποστολή μηνυμάτων προς και από τον εξυπηρετητή. Η επικοινωνία μπορεί να γίνει ανάμεσα σε διαφορετικούς τομείς.

Με την χρήση των WebSockets, μπορούν οι απλές εφαρμογές να εγκατασταθούν σε έναν εξυπηρετητή, ενώ σε έναν άλλο να χρησιμοποιηθούν άλλες πιο απαιτητικές υπηρεσίες όπως για παράδειγμα πολυμέσα ροής (streaming).

Η αποστολή μηνυμάτων γίνεται μόνο στην περίπτωση που οι υποδοχές (sockets)⁵, είναι ανοιχτές. Το μοντέλο επικοινωνίας μοιάζει ως εξής:

```
var ws = new WebSocket('ws://somesite.com/updates');
ws.onmessage = function (event) {
  alert(event.data);
};
ws.onopen = function () {
  ws.send('yay! we connected!');
};
```

2.3.1.4. ServerSentEvents

Επιτρέπει την αποστολή μηνυμάτων από τον εξυπηρετητή προς τον πελάτη και όχι το αντίστροφο. Ο πελάτης μπορεί μόνο να «ακούει» μηνύματα. Αυτό το API, χρησιμοποιεί το μοντέλο onmessage. Κατασκευάζεται χρησιμοποιώντας το αντικείμενο EventSource(Sharp, 2011).

Η δημιουργία κάποιου γεγονότος μπορεί να γίνει ως εξής:

```
var es = new EventSource('/sse');
es.onopen = function () {
  console.log('opened stream');
};
es.onmessage = function (event) {
  console.log('new message: ' + event.data);
};
```

ToSSE (ServerSentEvent), συνδέεται αυτόματα, κατά την δημιουργία του αντικείμενου και ενεργοποιεί το γεγονός onopen.

⁵ Socket είναι το ένα άκρο, από έναν επικοινωνιακό διάλογο διπλής κατεύθυνσης, μεταξύ δύο προγραμμάτων που εκτελούνται στο δίκτυο. Περιλαμβάνει το πρωτόκολλο, την διεύθυνση και τον αριθμό θύρας του άκρου (Αnon., 2013).

2.3.2. Διεπαφή συστήματος αρχείων (FilesystemAPIs)

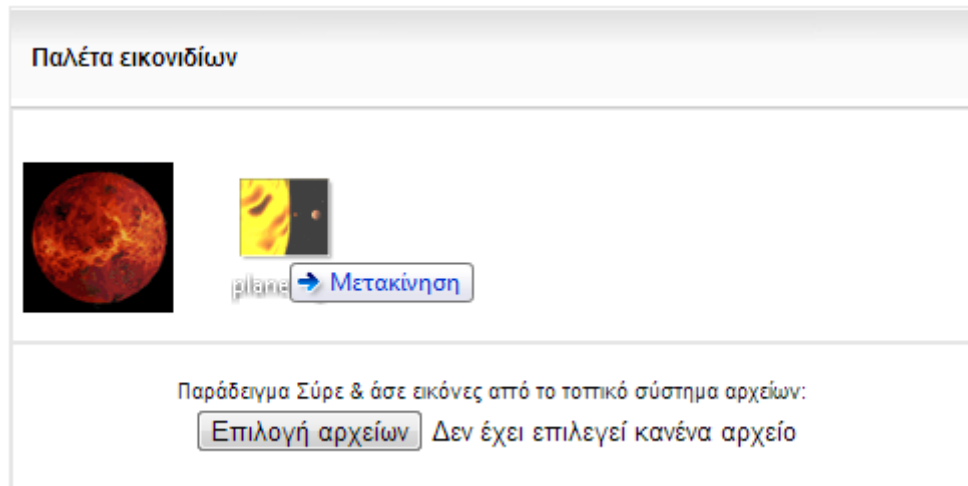
Το HTML5 σύστημα αρχείων API, δίνει τη δυνατότητα στις εφαρμογές να γράφουν και να αποθηκεύουν αρχεία μέσω της Javascript. Οι δυνατότητες που προσφέρονται μέσω της διασύνδεσης είναι:

- Ανάγνωση και διαχείριση αρχείων με τη χρήση των File/Blob, FileList, FileReader.
- Δημιουργία και εγγραφή με τις εντολές BlobBuilder, FileWriter.
- Πρόσβαση σε καταλόγους και συστήματα αρχείων με DirectoryReader, FileEntry/DirectoryEntry, LocalFileSystem.

Ένα παράδειγμα εφαρμογής είναι το ακόλουθο (Leah, 2013). Ουσιαστικά χρησιμοποιείται η λειτουργία σύρε & άσε, για την επιλογή αρχείων από το τοπικό σύστημα αρχείων.

```
function imagesSelected(myFiles) {
  for (var i = 0, f; f = myFiles[i]; i++) {
    var imageReader = new FileReader();
    imageReader.onload = (function(aFile) {
      return function(e) {
        var span = document.createElement('span');
        span.innerHTML = [<img class="images" src="", e.target.result,"" title="", aFile.name,
"/>].join("");
        document.getElementById('thumbs').insertBefore(span, null);
      };
    })(f);
    imageReader.readAsDataURL(f);
  }
}

function dropIt(e) {
  imagesSelected(e.dataTransfer.files);
  e.stopPropagation();
  e.preventDefault();
}
```

Εικόνα 0-13 Παράδειγμα εφαρμογής FilesystemAPI

Οι επιλογές αποθήκευσης του API, αφορούν μόνο την πλευρά του πελάτη. Μια τυπική διαδικτυακή εφαρμογή, μπορεί να χρησιμοποιήσει χώρο αποθήκευσης είτε προσωρινά είτε μόνιμα.

Η προσωρινή αποθήκευση μπορεί να πραγματοποιηθεί εύκολα χωρίς στην συγκατάθεση του χρήστη. Επίσης δεν απαιτείται η εκχώρηση ειδικών δικαιωμάτων. Η αποθήκευση αυτού του τύπου συνίσταται για χρήση κρυφής μνήμης (cache).

Η μόνιμη αποθήκευση χρησιμοποιείται για την αποθήκευση δεδομένων με τέτοιο τρόπο ώστε αυτά να είναι διαχειρίσιμα και για τις επόμενες προσβάσεις. Μία τέτοια λειτουργία είναι η αποθήκευση cookies, που διασφαλίζει ότι ο ιστότοπος «θυμάται» τις προτιμήσεις του χρήστη. Βέβαια ακόμα και η μόνιμη αποθήκευση μπορεί να διαγραφεί από τον χρήστη (Bidelman, 2011).

Σε αντίθεση με την προσωρινή αποθήκευση ισχύουν:

- Ο φυλλομετρητής ζητάει την άδεια του χρήστη για να αποθηκεύσει τα δεδομένα.
- Το μέγεθος των δεδομένων είναι σταθερό και δεν μπορεί να ζητηθεί έξτρα χώρος.

2.3.3. GeolocationAPI

Το GeolocationAPI, χρησιμοποιείται για τον προσδιορισμό της γεωγραφικής θέσης ενός χρήστη. Αποτελεί υψηλού επιπέδου API, συνεπώς ο χρήστης δεν χρειάζεται να ξέρει λεπτομέρειες για τον τρόπο που γίνεται ο εντοπισμός.

Το GeolocationAPI, υποστηρίζεται σχεδόν από όλους τους μοντέρνους φυλλομετρητές τόσο σε υπολογιστές, αλλά και σε κινητές συσκευές. Ο εντοπισμός μπορεί να επιτευχθεί μέσω πολλών τρόπων, με το καθένα να έχει συγκεκριμένη ακρίβεια ως προς τον ακριβή προσδιορισμό. Οι τρόποι αυτοί συνοπτικά είναι:

- **IPΔιεύθυνση:** Είναι δυνατός ο εντοπισμός της θέσης μέσω της διεύθυνση IP του χρήστη. Κάθε χώρα, περιοχή, πόλη έχει συγκεκριμένο εύροςIPδιευθύνσεων. Συνεπώς κάποιος μπορεί να προσδιορίσει τη θέση ενός χρήστη, μέσω αυτού του τρόπου, όχι όμως με μεγάλη ακρίβεια.
- **Σήμα Wi-Fi:** Σε συνδυασμό με τον εντοπισμό μέσω IPδιεύθυνσης μπορεί να δώσει μεγαλύτερη ακρίβεια. Στηρίζεται στο γεγονός, ότι τα Wi-Fi δίκτυα έχουν ευρετηριαστεί από αντίστοιχες υπηρεσίες και βάση αυτής της πληροφορίας είναι γνωστό το στίγμα τους. Έτσι από την ένταση του σήματος, το όνομα του δικτύου και το χρόνο αντίχτυσης μπορεί να υπολογιστεί η θέση του χρήστη από την γνωστή τοποθεσία.
- **Σήμα GPS:** Είναι η μέθοδος με την υψηλότερη ακρίβεια, χρησιμοποιείται όμως κυρίως από κινητές συσκευές. Χρησιμοποιεί ένα σύστημα δορυφόρων για τον εντοπισμό της θέσης ενός σημείου στη γη, χρησιμοποιώντας τριγωνισμό.

Οι τρεις βασικές συναρτήσεις που χρησιμοποιεί το API είναι:

- **getCurrentPosition:** Χρησιμοποιείται για να προσδιορίσει τις γεωγραφικές συντεταγμένες του χρήστη. Παίρνει δύο ορίσματα: το ένα για την επιτυχή αναζήτηση (locationSuccess) και το άλλο για την περίπτωση αποτυχίας (locationFail). Η συνάρτηση επιστρέφει ένα αντικείμενο θέσης, μέσω του οποίου μπορούμε να λάβουμε αρκετές πληροφορίες σε σχέση με τη θέση όπως: γεωγραφικό μήκος/πλάτος/ύψος, ακρίβεια, ταχύτητα, ώρα.
- **watchPosition:** Παρακολουθεί τις συντεταγμένες του χρήστη σε πραγματικό χρόνο. Επιστρέφει ένα αναγνωριστικό χρήστη (ID)
- **clearWatch:** Σταματάει την παρακολούθηση της θέσης του χρήστη. Παίρνει σαν παράμετρο το αναγνωριστικό που έχει επιστραφεί με τη watchPosition.

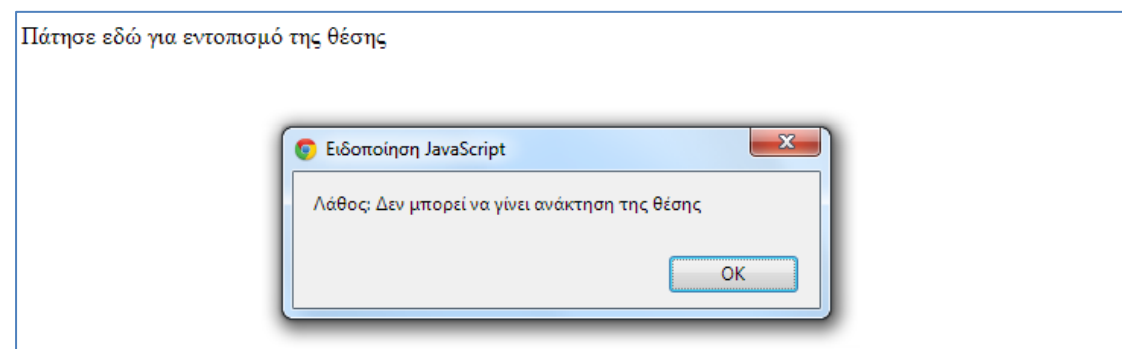
Χρησιμοποιώντας αντίστοιχα τις παραπάνω εντολές μπορούμε να προσδιορίσουμε τη γεωγραφική μας θέση με τον κάτωθι τρόπο:

```
if (navigator.geolocation) {
  var timeoutVal = 10 * 1000 * 1000;
  navigator.geolocation.getCurrentPosition(
    displayPosition,
    displayError,
    { enableHighAccuracy: true, timeout: timeoutVal, maximumAge: 0 }
  );
}
```

Αντίστοιχα η συνάρτηση `displayError` ορίζεται ως εξής:

```
function displayError(error) {  
  var errors = {  
    1: 'Δεν επιτρέπεται η πρόσβαση',  
    2: 'Δεν μπορεί να γίνει ανάκτηση της θέσης',  
    3: 'Το χρονικό όριο του αιτήματος εξαντλήθηκε'  
  };  
  alert("Λάθος: " + errors[error.code]);  
}
```

Στην περίπτωση τη δική μας δεν ήταν δυνατός ο εντοπισμός της θέσης, αφού στον υπολογιστή δεν είχαμε κάποιο `gps` wifiδέκτη.



Εικόνα 0-14 Εντοπισμός θέσης

Άλλες τεχνολογίες

Εισαγωγή

Για την ανάπτυξη ενός δικτυακού τόπου, υπάρχουν αρκετές προσφερόμενες τεχνολογίες, οι οποίες μπορούν να χρησιμοποιηθούν για την επίτευξη του στόχου. Μπορεί να είναι ολοκληρωμένα περιβάλλοντα ανάπτυξης, είτε ένα σύνολο εργαλείων που συνεργάζονται μεταξύ τους.

Τεχνολογίες ανάπτυξης δικτυακών τόπων.

Οι βασικότερες τεχνολογίες ανάπτυξης παρουσιάζονται στη συνέχεια.

3.1.1. XHTML

Η XHTML αναπτύχθηκε για να υπηρετήσει ένα σημαντικό σκοπό στο διαδίκτυο. Πολλές σελίδες περιείχαν «κακό» κώδικα HTML. Αυτό συνέβαινε για λόγους όπως ετικέτες που δεν έκλειναν, η έκλειναν με λάθος σειρά, αλλά και ιδιότητες οι οποίες δεν ήταν σωστές. Η XHTML δημιουργήθηκε για να αντιμετωπίσει αυτά τα λάθη. Ορίζοντας κανόνες που θα έπρεπε να ακολουθήσει η HTML, όπως όλες οι ετικέτες θα πρέπει να κλείνουν και μάλιστα εμφωλιασμένες η μία στην άλλη με τη σωστή σειρά, κατάφερε να βελτιώσει τον κώδικα HTML. Αυτή η βελτίωση παρήγαγε ένα καθαρότερο κώδικα με καλύτερα οπτικά αποτελέσματα για τους επισκέπτες των δικτυακών τόπων.

Μεταξύ του Αυγούστου του 2002 και του Ιουλίου του 2006, το W3C κυκλοφόρησε οκτώ προσχέδια της XHTML 2.0, μιας νέας έκδοσης XHTML η οποία επανασχεδιάστηκε με σκοπό να είναι μια ξεκάθαρη έκδοση, χωρίς την υποστήριξη συμβατότητας προς τα πίσω. Αυτή η έλλειψη συμβατότητας, προκάλεσε μεγάλη διαμάχη στην κοινότητα των κατασκευαστών. Ένα ένατο προσχέδιο της XHTML 2.0, αναμενόταν να εμφανιστεί το 2009, όμως στις 2 Ιουλίου του 2009, το W3C αποφάσισε να μην προχωρήσει σε περαιτέρω ανάπτυξη. Έτσι η XHTML 2.0 και τα σχετικά κείμενα κυκλοφόρησαν από το W3C σαν σημειώσεις τον Δεκέμβριο του 2010.

3.1.2. PHP

Η php, είναι μια γλώσσα σεναρίου η οποία έχει σχεδιαστεί για την δημιουργία εφαρμογών διαδικτύου. Χρησιμοποιείται όμως και σαν γενικού σκοπού γλώσσα προγραμματισμού.

Ο ρηκώδικας ερμηνεύεται από έναν εξυπηρετητή webμέσω μιας μονάδας επεξεργασίας, η οποία παράγει την προκύπτουσα ιστοσελίδα. Οι εντολές ρhp, μπορούν να ενσωματωθούν άμεσα σε ένα έγγραφο HTML.

Ηρhpσυνεργάζεται άψογα με την βάση δεδομένων MySQLπου θα αναφερθούμε στη συνέχεια. Επίσης δίνεται σαν συνολικό πακέτοσε περιβάλλον Linux, το οποίο καλείται LAMPκαι περιέχει όλα τα προαπαιτούμενα για την δημιουργία ενός δυναμικού ιστοτόπου. Αυτά είναι ο εξυπηρετητής Apache, η MySQLκαι φυσικά η ρhp. Αντίστοιχα στα windows, υπάρχει το xamppτο οποίο θα χρησιμοποιήσουμε για τις ανάγκες κατασκευής του ιστοτόπου μας.Επίσης ρhpσενάρια, θα χρησιμοποιήσουμε για την αποστολή των φορμών επικοινωνίας του ιστοτόπου.

3.1.3. JSP (JavaServer Pages)

Είναι μια τεχνολογία που βοηθά στην δημιουργία δυναμικών σελίδων, που βασίζονται στην HTML, στην XMLή σε οποιοδήποτε άλλο τύπο εγγράφου. Για την ανάπτυξη και εκτέλεση μιας σελίδας jsp, απαιτείται η χρήση ενός συμβατού εξυπηρετητή που υποστηρίζει την αντίστοιχη κλάση Javaμε την οποία επεκτείνονται οι δυνατότητές του. Ο πιο γνωστός εξυπηρετητής που υποστηρίζει αυτή τη δυνατότητα είναι το ApacheTomcatενώ υπάρχει και ο Jetty(Anon., 2014).

Το μοντέλο JSP αρχιτεκτονικά μπορεί να παρασταθεί ως μια υψηλού επιπέδου αφαίρεση από κλάσεις Java (jvaservlets). Υιοθετεί το μοντέλο 2, ένα σύνθετο σχέδιο το οποίο διαχωρίζει την εμφάνιση του περιεχομένου, από τη λογική που χρησιμοποιείται για την λήψη και διαχείριση του περιεχομένου. Ουσιαστικά είναι μια εναλλακτική προσέγγιση του MVCμοντέλου. Όμως το model 2, θα μπορούσαμε να πούμε ότι αποτελείται από δυο συνιστώσες: της εμφάνισης και της διαχείρισης, ενώ τοMVCπροχωράει ένα βήμα παραπέρα στο κομμάτι της διαχείρισης.

3.1.4. Javascript

Η Javascriptείναι μια γλώσσα προγραμματισμού, η οποία αποτελεί μέρος των μοντέρνων προγραμμάτων πλοήγησης και επιτρέπει την εκτέλεση σεναρίων στη μεριά του πελάτη για αλληλεπίδραση με τον επισκέπτη της ιστοσελίδας, έλεγχο του φυλλομετρητή, ασύγχρονη επικοινωνία και αλλαγή του περιεχομένου που εμφανίζεται. Χρησιμοποιείται όμως και στη πλευρά του εξυπηρετητή κυρίως για υλοποίηση παιχνιδιών και δημιουργία εφαρμογών επιφάνειας εργασίας(Anon., 2014).

Έχει παρόμοια σύνταξη με την γλώσσα προγραμματισμού και παρ' όλο ότι το όνομά της έχει αναφορά στη Javaσημασιολογικά δεν έχουν μεγάλη σχέση.

Η Javascriptέχει επηρεάσει για τη δημιουργία του ανοιχτού προτύπου JSON, το οποίο περιγράφει τον τρόπο για την μετάδοση δεδομένων σε κατάλληλη μορφή. Επίσης το jQueryείναι μια βιβλιοθήκη Javascriptπου έχει σχεδιαστεί για την απλοποίηση της δημιουργίας σεναρίων στην πλευρά του πελάτη.

Η Javascriptμαζί με τη χρήση της τεχνολογίας AJAX, επιτρέπει στους δημιουργούς την κατασκευή πιο επαγγελματικών σελίδων με πλούσιο περιεχόμενο. Χαρακτηριστικό της Javascript, είναι ότι η επεξεργασία του κώδικα συνήθως δεν πραγματοποιείται στον διακομιστή, αλλά τοπικά από το πρόγραμμα περιήγησης(Meyer, 2011). Στο προσχέδιο της HTML5 δεν γίνεται κάποια σχετική αναφορά σχετικά με τη Javascript, όμως σύμφωνα με το W3C, σχεδιάζεται η αντικατάσταση του JavaScriptAPIσε σχέση με την HTML4, XML1 και το επίπεδο 2 DOM εγγράφων.

3.1.5. Αρχεία στυλ CSS

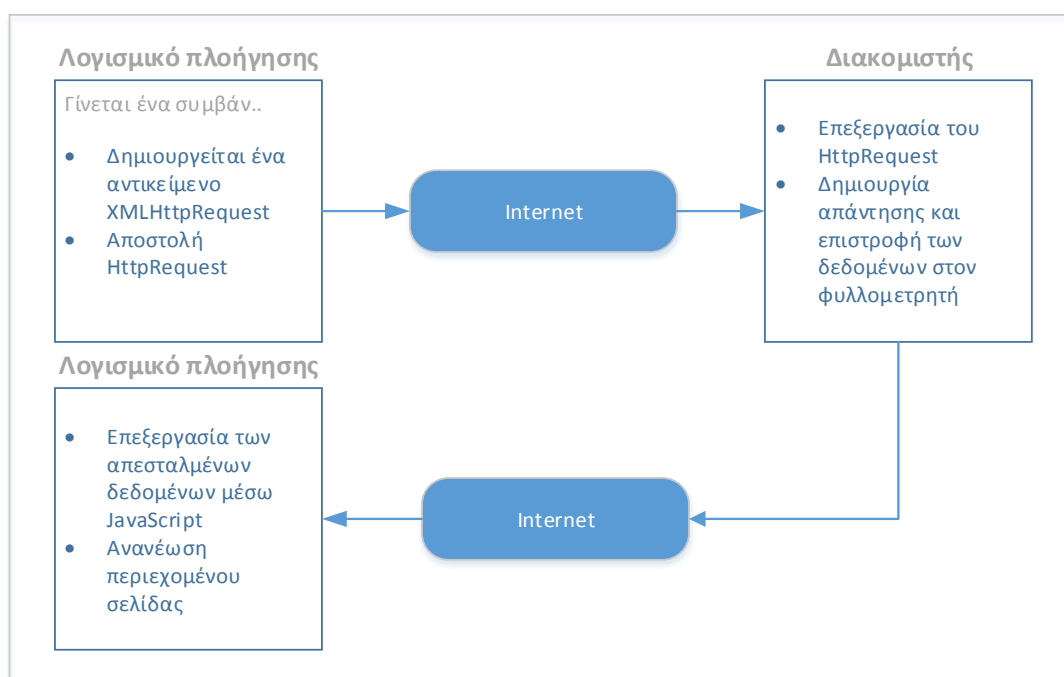
Τα αρχεία αυτά χρησιμοποιούν τη γλώσσα CSSγια να περιγράψουν την όψη και τη μορφοποίηση ενός εγγράφου, που έχει γραφτεί σε μία γλώσσα σήμανσης. Σχεδιάστηκε αρχικά για να διαχωρίσει το περιεχόμενο από την παρουσίαση και περιλαμβάνει στοιχεία για το σχέδιο του εγγράφου και την μορφοποίηση του κειμένου. Η τελευταία έκδοση είναι η CSS3 και αποτελείται από ξεχωριστές ενότητες. Κάθε ενότητα προσθέτει νέες δυνατότητες ή επεκτείνει δυνατότητες όπως αυτές ορίστηκαν στην CSS 2.Συνοπτικά αυτές οι ενότητες περιλαμβάνουν:

- Επιλογείς (selectors)
- Μοντέλο κουτιού (box model)
- Παρασκήνια και περιθώρια
- Εφέ κειμένου
- 2D/3D μεταμορφώσεις
- Κίνηση
- Πολλαπλές στήλες
- Διεπιφάνεια χρήστη

Τα φύλλα στυλ, μπορούν να ενσωματωθούν σε ένα έγγραφο html, αλλά μπορούν να αποτελούν και ξεχωριστό αρχείο.

3.1.6. Ajax

Η τεχνολογία Ajax αποτελεί μία ομάδα αλληλένδετων τεχνικών ανάπτυξης, που χρησιμοποιούνται στην πλευρά του πελάτη για την δημιουργία ασύγχρονων εφαρμογών Web. Μέσω αυτής της τεχνολογίας οι διαδικτυακές εφαρμογές μπορούν να στείλουν αλλά και να ανακτήσουν δεδομένα από ένα διακομιστή ασύγχρονα, χωρίς να χρειάζεται η ανανέωση της υπάρχουσας σελίδας, αλλά μόνο ένα μέρος αυτής. Τα δεδομένα μπορούν να ανακτηθούν χρησιμοποιώντας το αντικείμενο XMLHttpRequest. Η περιγραφή των δεδομένων γίνεται μέσω του μοντέλου JSON.



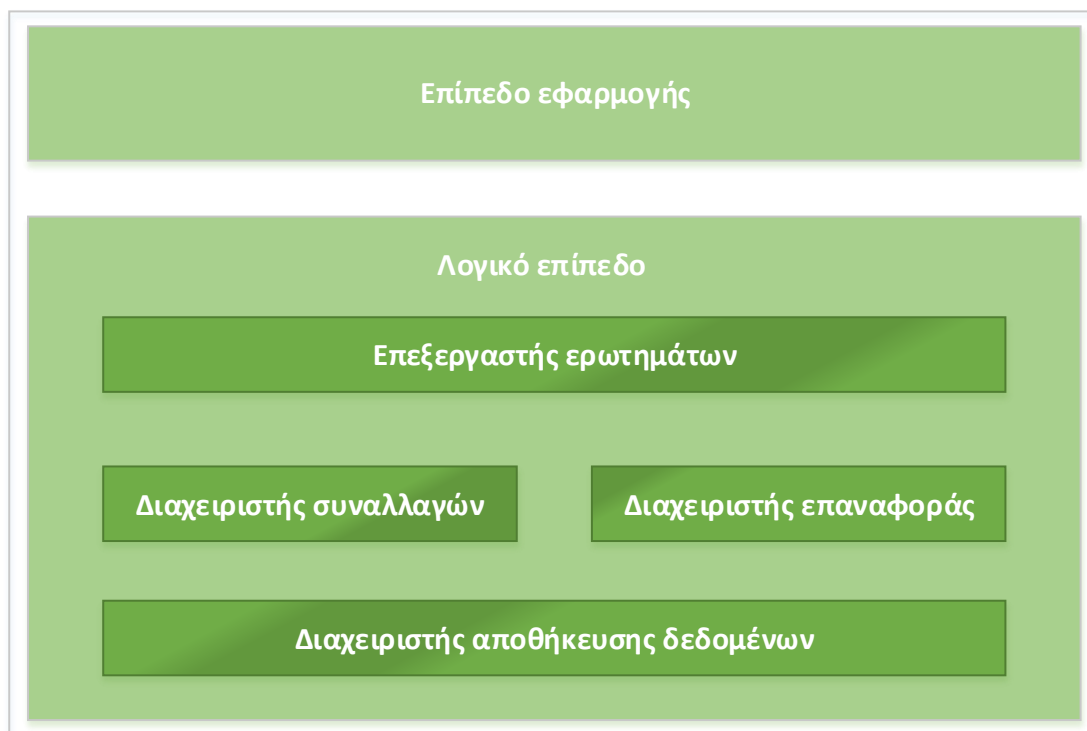
Εικόνα 0-1Ajax (πηγή: http://www.w3schools.com/ajax/ajax_intro.asp)

Όπως είπαμε η Ajax δεν είναι μια ενιαία τεχνολογία, αλλά αποτελείται από μία ομάδα. Έτσι η HTML και CSS μπορούν να χρησιμοποιηθούν για την σήμανση και τις πληροφορίες στυλ, ενώ η JavaScript για αλλαγή του περιεχομένου της σελίδας.

3.1.7. MySQL

Η MySQL είναι το δεύτερο πιο ευρύτατα χρησιμοποιούμενο σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Η ομάδα ανάπτυξης της MySQL, έχει κάνει τον κώδικα διαθέσιμο, κάτω από τους όρους την GNU. Η MySQL αναπτύχθηκε από μία σουηδική εταιρία, ενώ τώρα έχει περάσει στα χέρια της Oracle.

Είναι μια δημοφιλής επιλογή για χρήση σε δικτυακές εφαρμογές και έχει πολλές πλατφόρμες ανάπτυξης την έχουν υιοθετήσει. Η αρχιτεκτονική της βάσης παρουσιάζεται στο επόμενο σχήμα (Vuong, 2013).



Εικόνα 0-2 Αρχιτεκτονική MySQL

Στο επίπεδο εφαρμογής οι χρήστες και οι πελάτες αλληλεπιδρούν με την MySQL. Υπάρχουν τρία συστατικά σε αυτό το επίπεδο:

- **Διαχειριστές:** Χρησιμοποιούν διάφορα περιβάλλοντα και εργαλεία όπως π.χ. `mysqladmin` για την πλήρη διαχείριση των βάσεων.
- **Πελάτες:** Οι πελάτες επικοινωνούν με την βάση μέσω κατάλληλων διεπαφών και εργαλείων όπως π.χ. τα `MySQLAPIs`.
- **Χρήστες:** Οι απλοί χρήστες επικοινωνούν μέσω της διεπιφάνειας “`mysql`” που ουσιαστικά είναι μία διεπιφάνεια ερωτημάτων. Χρησιμοποιείται για να επιτρέψουν στους χρήστες την έκδοση ερωτημάτων, αλλά και τη λήψη των αποτελεσμάτων.

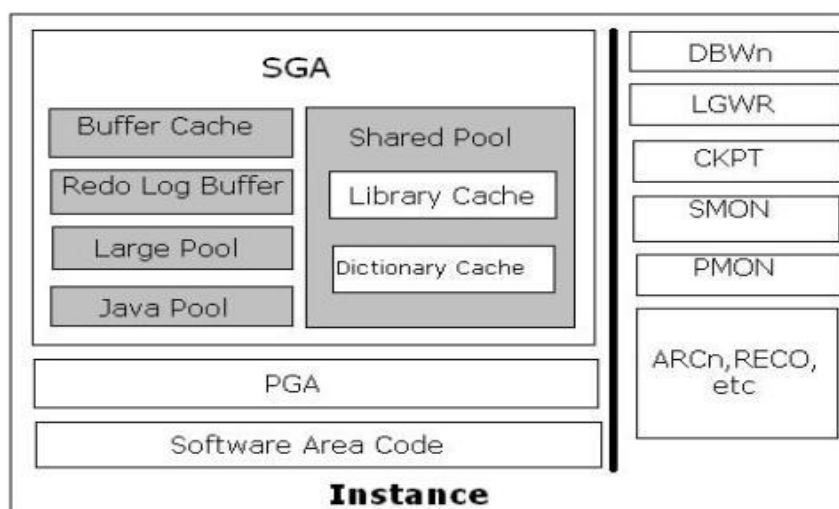
Το λογικό επίπεδο της αρχιτεκτονικής διαιρείται σε τέσσερα υπο-συστήματα, τα οποία συνεργάζονται για να διαχειριστούν τα αιτήματα προς τον διακομιστή:

- **Επεξεργαστής ερωτημάτων:** Ο επεξεργαστής αποτελείται από υπό-μονάδες που σκοπό έχουν την ανάκτηση, επεξεργασία, βελτιστοποίηση και εκτέλεση των ερωτημάτων.
- **Διαχειριστής συναλλαγών:** Είναι υπεύθυνος για κλείδωμα των εγγραφών, την διαχείριση πολλαπλών προσβάσεων ταυτόχρονα και για την πρόληψη αλλοίωσης των δεδομένων
- **Διαχειριστής επαναφοράς:** Είναι υπεύθυνος για την καταγραφή κάθε λειτουργίας σε αρχεία ημερολογίου (logs). Αποθηκεύει επίσης τις εντολές των συναλλαγών για την ανάκτηση του συστήματος σε προηγούμενη λειτουργική κατάσταση.
- **Διαχειριστής αποθήκευσης:** Η κύρια λειτουργία του είναι η αποδοτική αποθήκευση των δεδομένων.

Η MySQL, χρησιμοποιεί πρόσθετες μηχανές αποθήκευσης, οι οποίες είναι διαχωρισμένες από τον πυρήνα. Έχει περισσότερες από 20 μηχανές με τις πιο κοινές να είναι η InnoDB και η MyISAM. (Anon., 2014)

3.1.8. Oracle

Η βάση δεδομένων Oracle, είναι ένα αντικειμενοστραφές σύστημα διαχείρισης βάσεων δεδομένων που έχει δημιουργηθεί από την εταιρία Oracle. Βασικά, υπάρχουν δύο κύριες συνιστώσες σε μία βάση Oracle: στιγμιότυπο και η βάση δεδομένων. Ένα στιγμιότυπο αποτελείται από ορισμένες δομές μνήμης και διεργασίες παρασκηνίου, ενώ η βάση δεδομένων αφορά τους πόρους του δίσκου (Anon., 2008).



Εικόνα 0-3 Αρχιτεκτονική Oracle

Η SGA είναι οι πρωτεύουσες δομές μνήμης. Αυτή η περιοχή αποτελείται από:

- **Buffercache:** Χρησιμοποιείται για να αποθηκεύει τα μπλοκ των δεδομένων που ανακτώνται από τα αρχεία.
- **SharedPool:** Αποτελείται από την προσωρινή βιβλιοθήκη και την προσωρινό λεξικό. Η βιβλιοθήκη χρησιμοποιείται για την αποθήκευση πληροφορίας που αφορούν αιτήματα SQL, ενώ το λεξικό αποθηκεύει πληροφορίες σχετικές με τους πίνακες, στήλες, ευρετήρια, δικαιώματα κλπ.
- **RedoLogBuffer:** Όταν εκτελείται μία διεργασία από τους χρήστες (select, insert, update, delete), τότε αυτή καταγράφεται. Στην περίπτωση που δεν εκτελεστεί με επιτυχία η διεργασία, με αυτό τον τρόπο γίνεται ανάκτηση.
- **Largepool:** Είναι μια προαιρετική περιοχή μνήμης στην SGA. Χρησιμοποιείται για την διαχείριση των I/O διαδικασιών, αλλά και για την επέκταση της κοινόχρηστης μνήμης.
- **Java pool:** Χρησιμοποιείται για την συντακτική ανάλυση των εντολών Java.
- **ProgramGlobalArea:** Παρ' όλο που τα αποτελέσματα μιας εντολής SQL αποθηκεύονται στην προσωρινή βιβλιοθήκη, η τιμή της μεταβλητής σύνδεση αποθηκεύεται στην PGA γιατί πρέπει να είναι ιδιωτική για λόγους ασφάλειας.
- **SoftwareAreaCode:** Περιέχει τον κώδικα εκτέλεσης της Oracle.

Αντίστοιχα οι διαδικασίες παρασκηνίου αφορούν:

- **DBWn:** Γράφει τα δεδομένα από το προσωρινό χώρο αποθήκευσης στα αρχεία.
- **LGWR:** Γράφει τα δεδομένα από το προσωρινό χώρο επαναφοράς σε αρχείο ημερολογίου (log)
- **CKPT:** Πυροδοτεί το DBWn για να ξεκινήσει τη διαδικασία εγγραφής.
- **SMON:** Χρησιμοποιείται για την ανάκτηση των δεδομένων από τον προσωρινό χώρο επαναφοράς σε περίπτωση αποτυχίας.
- **PMON:** Επαναφέρει το σύστημα μετά από ανάκτηση ελευθερώνοντας παράλληλα πόρους.

Η καθ' αυτή βάση δεδομένων αποτελείται από μικρότερες λογικές μονάδες για την διαχείριση, αποθήκευση και ανάκτηση των δεδομένων. Αυτές οι λογικές μονάδες είναι:

- **Tablespace:** Είναι μία περιοχή αποθήκευσης όπου τα πραγματικά δεδομένα κρατούνται.
- **Segment:** Ένα tablespace χωρίζεται σε τμήματα για την αποθήκευση ομότυπων αντικειμένων.
- **Extent:** Ένα segment χωρίζεται σε επεκτάσεις που περιέχουν μπλοκ δεδομένων.
- **DataBlock:** Είναι η μικρότερη μονάδα αποθήκευσης σε μία βάση Oracle.

Πλαίσια ανάπτυξης διαδικτυακών εφαρμογών

Ένα πλαίσιο ανάπτυξης διαδικτυακών εφαρμογών (WAF: **WebApplicationFramework**) είναι ένα πλαίσιο λογισμικού το οποίο έχει σχεδιαστεί για την ανάπτυξη δυναμικών ιστοστόπων, δικτυακών εφαρμογών και υπηρεσιών. Παρέχουν μια σειρά βιβλιοθηκών για πρόσβαση σε βάσεις δεδομένων, πρότυπα διεπιφάνειας και εργαλεία διαχείρισης (Anon., 2014).

Υπάρχουν κυρίως δυο διαφορετικές τεχνολογίες ως προς την αρχιτεκτονική αυτών των συστημάτων. Τα περισσότερα από αυτά, βασίζονται στο μοντέλο MVC. Σύμφωνα με αυτό το μοντέλο, υπάρχει διαχωρισμός των επί μέρους λειτουργιών του πλαισίου.

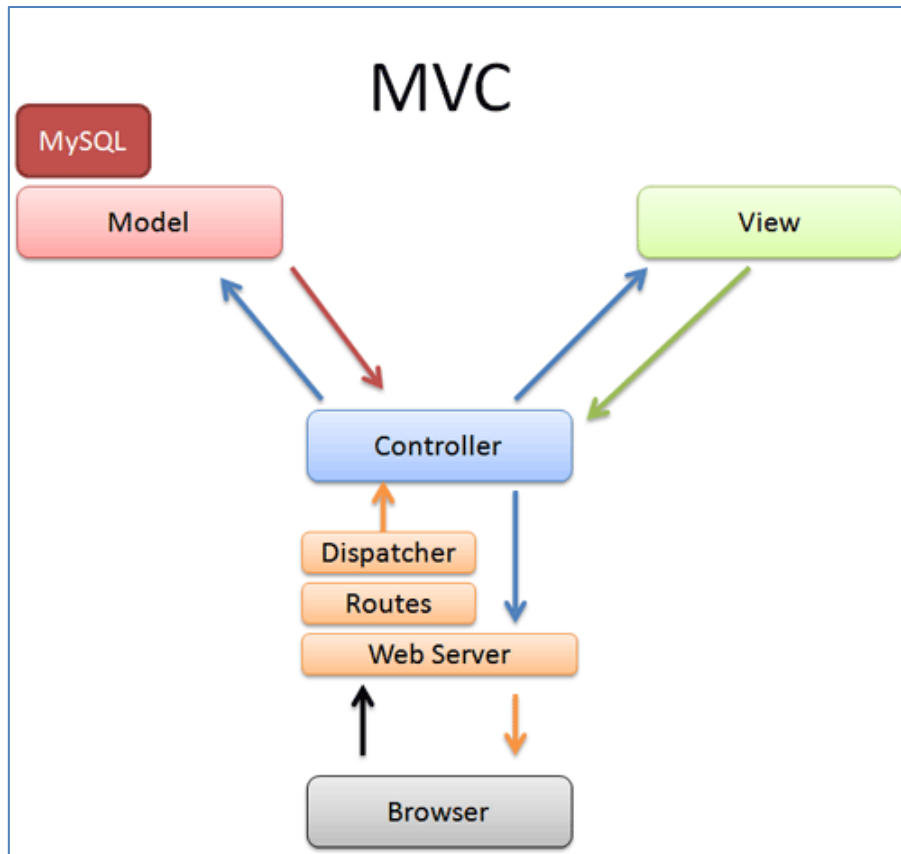
Το μοντέλο MVC αποτελείται όπως φαίνεται από την εικόνα 1 από τις εξής ενότητες:

- **Model:** Αναπαριστά τα δεδομένα της εφαρμογής
- **View:** Παράγει μια παρουσίαση των δεδομένων του μοντέλου
- **Controller:** Διαχειρίζεται και κατευθύνει τα αιτήματα των χρηστών

Η διαδικασία που πραγματοποιείται όταν δημιουργείται ένα αίτημα, δηλαδή η ανάκτηση μιας σελίδας συνοπτικά είναι:

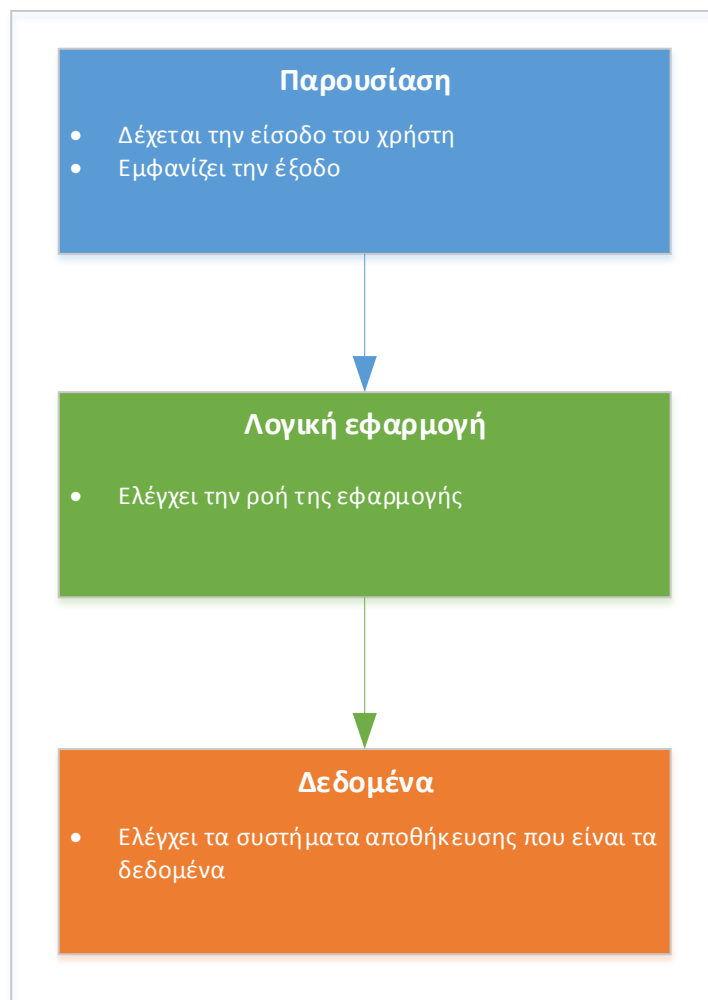
- Ο φυλλομετρητής στέλνει ένα αίτημα για την ανάκτηση μιας σελίδας.
- Ο εξυπηρετητής (WebServer) λαμβάνει το αίτημα και το δρομολογεί.
- Στη συνέχεια ο controller δίνει τις κατάλληλες εντολές, για να ανακτηθεί η αντίστοιχη πληροφορία, μέσω του model.

- Το model αναλαμβάνει να προσπελάσει την βάση για την ανάκτηση της πληροφορίας.



Εικόνα 0-4 Μοντέλο MVC

Η άλλη αρχιτεκτονική βασίζεται στην οργάνωση τριών επιπέδων (Three-tier organization). Σε αυτή, οι εφαρμογές δομούνται γύρω από τρία φυσικά επίπεδα: τον πελάτη (client), την εφαρμογή και την βάση δεδομένων. Η βάση δεδομένων είναι μια οποιαδήποτε σχεσιακή, ενώ η εφαρμογή τρέχει σε έναν εξυπηρετητή και επικοινωνεί χρησιμοποιώντας HTTP.



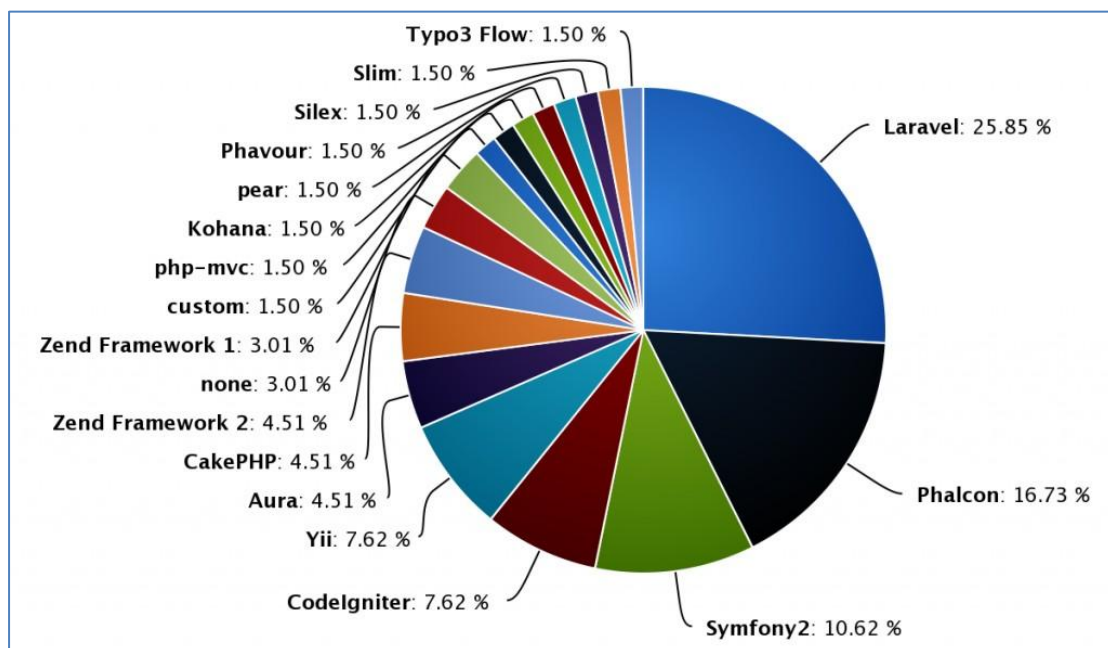
Εικόνα 0-5 Αρχιτεκτονική 3-tier

Μερικά από τα σημαντικότερα πλαίσια είναι:

- **Symfony:** Είναι ένα phpπλαίσιο για την ανάπτυξη δυναμικών ιστοτόπων. Ακολουθεί το μοντέλο MVC και αποτελεί λογισμικό ανοιχτού κώδικα. Θεωρείται το ευκολότερο από τα άλλα πλαίσια, έχει πολύ καλή τεκμηρίωση και είναι αρκετά δημοφιλές.
- **Zend:** Είναι ανοικτού κώδικα πλαίσιο για την ανάπτυξη διαδικτυακών εφαρμογών χρησιμοποιώντας php. Είναι πλήρως αντικειμενοστραφές και υιοθετεί στην παρούσα έκδοση τα περισσότερα από τα χαρακτηριστικά της php 5.3.
- **Phalcon:** Είναι ένα ανοικτού κώδικα, πλήρης στοίβας πλαίσιο για την PHP 5 γραμμένος ως C-επέκταση, βελτιστοποιημένο για υψηλή απόδοση. Είναι από τα πιο δημοφιλή πλαίσια.

- **Laravel:** Ακολουθεί και αυτό το μοντέλο MVC και σύμφωνα με την έρευνα που απεικονίζεται στην εικόνα, αναφέρεται ως το πιο δημοφιλές πλαίσιο για το 2013.
- **CodeIgniter:** Σύμφωνα με τον δημιουργό είναι ένα ισχυρό πλαίσιο PHP με ένα πολύ μικρό ίχνος, που χτίστηκε για την δημιουργία πλούσιων εφαρμογών web. Είναι αρκετά δημοφιλές και λόγω του μικρού ίχνους που αφήνει, χρησιμοποιείται σε sharedhosted περιβάλλοντα.

Η εικόνα που ακολουθεί παρουσιάζει αυτά τα πλαίσια σύμφωνα με την δυναμικότητά τους.



Εικόνα 0-6 Πλαίσια ανάπτυξης ιστοτόπων (πηγή: <http://www.sitepoint.com/best-php-frameworks-2014/>)

Συστήματα CMS

Με την ανάπτυξη του διαδικτύου υπήρξε ανάγκη δημιουργίας αποδοτικότερων εργαλείων δημιουργίας ιστοσελίδων. Η φράση “catch-all” επινοήθηκε για να περιγράψει το ευρύ φάσμα των συστημάτων που επέτρεπαν στους χρήστες την δημιουργία, επεξεργασία και διαχείριση του περιεχομένου μιας ιστοσελίδας.

Στις αρχές του 1990, υπήρξε η πρώτη δυνατότητα ενημέρωσης του διαδικτυακού περιεχομένου με προϊόντα που είχαν αναπτυχθεί από την Microsoft και την Lotus, ενώ το πρώτο υποτυπώδες σύστημα διαχείρισης περιεχομένου ήταν το StoryServer

από την Vignette γύρω στο 1996. Τα χρόνια που ακολούθησαν παρουσιάστηκαν αρκετές προτάσεις προετοιμάζοντας για την ανάπτυξη που θα είχαν σήμερα τα CMS.

Παρ' όλα αυτά μεταξύ του 2000 και 2005 υπήρξε ένα μαζικό κύμα συγχωνεύσεων και εξαγορών μεταξύ των εταιρειών που δραστηριοποιούνταν στο συγκεκριμένο χώρο. Τα συστήματα που είχαν δημιουργηθεί εγκαταλείφθηκαν με αποτέλεσμα οι χρήστες να μείνουν χωρίς υποστήριξη.

Η αγορά όμως γρήγορα επανέκαμψε και το 2007 υπήρχαν ουσιαστικά τρεις τύποι συστημάτων:

- **Αποσύνδεσης:** Εξυπηρετούσαν την σχεδίαση του ιστοτόπου σε τοπικό επίπεδο και στη συνέχεια γίνονταν μεταφόρτωση του περιεχομένου στο διαδίκτυο. Απαιτούσαν την εγκατάσταση κάποιας εφαρμογής τοπικά.
- **Απ' ευθείας σύνδεσης:** Αυτά τα συστήματα δεν χρειαζόντουσαν εγκατάσταση κάποιου προγράμματος, αφού το πρόγραμμα διαχείρισης ήταν από μόνο του ένας δικτυακός τόπος που μπορούσε να συνδεθεί με όνομα χρήστη και συνθηματικά.
- **Υβριδικά συστήματα:** Υποστήριζαν και τις δύο λειτουργίες (on-line, off-line).

Σήμερα τα CMS έχουν γίνει εξαιρετικά πολύπλοκα, επιτρέποντας στους χρήστες να κάνουν σχεδόν τα πάντα, χωρίς την συγγραφή ούτε μιας γραμμής κώδικα. Οι νέες εξελίξεις έχουν ενσωματώσει πρακτικές μάρκετινγκ όπως συστήματα αποστολής μαζικών μηνυμάτων μέσω ενσωματωμένου ηλεκτρονικού ταχυδρομείου, παρακολούθηση στατιστικών στοιχείων, δημιουργία δημοσκοπήσεων και μια σειρά άλλων δυνατοτήτων.

Ένα Σύστημα Διαχείρισης Περιεχομένου χωρίζει το σχεδιασμό και τη διαχείριση της ιστοσελίδας από την προβολή του περιεχομένου. Εάν προσπαθήσουμε να δημιουργήσουμε ένα στατικό δικτυακό τόπο θα ανακαλύψουμε την δυσκολία ενημέρωσής του με νέες σελίδες. Ακόμα πιο επίπονη διαδικασία είναι η προσθήκη διαφορετικών μενού σε αυτές τις νέες σελίδες. Ένα Σύστημα Διαχείρισης Περιεχομένου αναλαμβάνει την αυτοματοποίηση αυτών των διαδικασιών και αφήνει τον δημιουργό να επικεντρωθεί αυστηρά στο περιεχόμενο.

Σύμφωνα με την Wikipedia:

“Τα Συστήματα Διαχείρισης Περιεχομένου (ΣΔΠ, Content Management Systems, CMS) είναι διαδικτυακές εφαρμογές που επιτρέπουν την online τροποποίηση του περιεχομένου ενός δικτυακού τόπου. Οι διαχειριστές μέσω του διαδικτύου ενημερώνουν το περιεχόμενο στο ΣΔΠ, το οποίο είναι εγκατεστημένο σ' ένα διακομιστή. Οι αλλαγές αυτές γίνονται αυτόματα διαθέσιμες πάλι μέσω του διαδικτύου, σε όλους τους επισκέπτες και χρήστες του δικτυακού τόπου”

Τα εργαλεία αυτού του συστήματος χρησιμοποιούνται για να παρέχουν:

- **Ασφάλεια στην πρόσβαση:** Με την δημιουργία χρηστών και δικαιωμάτων μπορούμε να καθορίσουμε το επίπεδο πρόσβασης. Έτσι μπορούμε να αναθέσουμε συγκεκριμένες αρμοδιότητες σε ένα άτομο ή να απαγορέψουμε ανάλογα κάποιες. Για παράδειγμα μπορούμε να δημιουργήσουμε ένα χρήστη ο οποίος θα μπορεί να τροποποιήσει ένα άρθρο, αλλά δεν θα μπορεί να το σβήσει.
- **Εύκολη διαχείριση των δεδομένων:** Εφόσον όλα τα δεδομένα αποθηκεύονται σε μια βάση δεδομένων, η πρόσβαση και η διαχείριση αυτών γίνεται αρκετά εύκολα. Η επικαιροποίηση των δεδομένων συνιστά μια απλή διαδικασία
- **Γεννήτρια αναφορών και στατιστικών:** Η οργάνωση των δεδομένων, μας δίνει την δυνατότητα παροχής στατιστικών στοιχείων. Επίσης η εκτύπωση αναφορών γίνεται εύκολα μέσω ερωτημάτων στην βάση.
- **Υποστήριξη για πολυμέσα:** Υπάρχει η δυνατότητα διαχείρισης οποιουδήποτε αρχείου όπως απλά έγγραφα, ταινίες, εικόνες.

Τα συστήματα διαχείρισης περιεχομένου, συνήθως αποτελούνται από δύο μέρη:

- **back-end:** Αποτελεί το σύστημα διαχείρισης του Joomla, που παρέχει όλα εκείνα τα εργαλεία για την δημιουργία του ιστοτόπου.
- **front-end:** Είναι το τμήμα που βλέπει ο επισκέπτης, δηλαδή ο ίδιος ο ιστότοπος. Ουσιαστικά το front-end είναι το αποτέλεσμα των ενεργειών που γίνονται στο back-end.

Οι συνηθέστερες ενέργειες που γίνονται στο back-end είναι:

- **Ενεργοποίηση συστήματος εισόδου:** Χρησιμοποιείται για την δημιουργία μιας οθόνης εισόδου για την διαπίστευση του χρήστη κατά την επίσκεψή του στον δικτυακό τόπο. Επίσης δίνει την δυνατότητα εγγραφής ενός χρήστη. Ο

εγγεγραμμένος χρήστης μπορεί να χρήζει ιδιαίτερων δικαιωμάτων, όπως για παράδειγμα η επίσκεψη σε σελίδες που δεν μπορεί να δει ο απλός επισκέπτης του ιστοτόπου.

- **Ενεργοποίηση συστήματος δημοφιλέστερων ή νεότερων άρθρων:** Παρέχει την δυνατότητα εμφάνισης των δημοφιλέστερων στοιχείων σε έναν δικτυακό τόπο (άρθρων, εικόνων, ταινιών κλπ).
- **Δημιουργία ψηφοφοριών:** Παρέχει την δυνατότητα δημοσκοπήσεων.
- **Παροχή πεδίου αναζήτησης:** Παρέχει την δυνατότητα αναζήτησης των περιεχομένων του ιστοτόπου.
- **Πολύγλωσση υποστήριξη:** Παρέχει την δυνατότητα δημιουργίας ιστοσελίδων σε πολλές γλώσσες.

Υλοποίηση

Η υλοποίηση στα πλαίσια της πτυχιακής εργασίας, θα γίνει σε τοπικό επίπεδο και για αυτό τον λόγο θα γίνει η εγκατάσταση του πακέτου `xampp` για να καλύψουμε τις σχετικές απαιτήσεις οι οποίες είναι:

- **WebServerApache**: Για την προσπέλαση του δικτυακού τόπου είναι απαραίτητη η εγκατάσταση ενός εξυπηρετητή WEB όπως είναι ο Apache.
- **PHP**: Για την λειτουργία του ιστοτόπου όσο αφορά τα δυναμικά στοιχεία (αποστολή φορμών επικοινωνίας) της απαραίτητη είναι η εγκατάσταση της `php`.

Τα εργαλεία που θα χρησιμοποιηθούν είναι:

- **Πρόγραμμα επεξεργασίας εικόνας**: Θα χρησιμοποιηθεί το `photoshop` για την δημιουργία μερικών από τις εικόνες παρουσίασης (`banners`).
- **Επεξεργαστής κειμένου (editor)**: Για εισαγωγή του κώδικα θα χρησιμοποιηθεί ο `notepad++`.
- **Διάφορα δικτυακά εργαλεία**: Αυτά αφορούν εργαλεία διακόσμησης κυρίως, όπως παραγωγή κουμπιών, παρασκηνίων κλπ.

Πριν προχωρήσουμε στην διαδικασία της υλοποίησης θα πρέπει να ορίσουμε κάποιες προδιαγραφές σχετικά με αυτόν. Δηλαδή το είδος του περιεχομένου και την οργάνωση της πληροφορίας. Ο δικτυακός μας τόπος, αφορά την παρουσίαση μιας εταιρίας. Έτσι αυτός θα πρέπει να έχει ένα κεντρικό σύστημα επιλογών, για την εύκολη πλοήγηση του χρήστη. Οι επιλογές θα είναι:

Σελίδες στο μενού επιλογών:

- **Αρχή(index.html)**: Εμφανίζει την αρχική σελίδα του ιστοτόπου
- **Σχετικά με εμάς (aboutus.html)**: Θα εμφανίζει πληροφορίες σχετικά με την εταιρία.
- **HerbusHerbalTea (herbaltea.html)**: Πληροφορίες σχετικά με το προϊόν
- **Γεύσεις (flavos.html)**: Παρουσίαση των εναλλακτικών γεύσεων και δυνατότητα εκδήλωσης ενδιαφέροντος παραγγελίας από πελάτες.
- **Επικοινωνία (contact.html)**: Φόρμα επικοινωνίας με τους επισκέπτες του ιστοτόπου.

Υπόλοιπες σελίδες:

- **Πολυμέσα (videos.html):** Σελίδα με βίντεο παρουσίασης των προϊόντων.
- **Ανακοινώσεις (press.html):** Ανακοινώσεις στον τύπο.
- **Παραγωγή (production.html):** Πληροφορίες για την παραγωγή των προϊόντων.
- **Δίκτυο Συνεργατών (network.html):** Σημεία πώλησης.

Αρχική σελίδα (index.html)

Η αρχική σελίδα αποτελείται από τρία τμήματα, τα οποία ορίζονται στο αρχείο /css/style.css και είναι: η επικεφαλίδα, το κύριο μέρος και το υποσέλιδο.

4.0.1. Επικεφαλίδα

Η επικεφαλίδα περιέχει το λογότυπο, το μενού πλοήγησης και υλοποιείται με τον ακόλουθο κώδικα HTML5 και CSS.

```

                                HTML
<header class="header">
  <h1 id="logo"><a href="#">Intergamma</a></h1>
  <nav id="navigation">
    <ul>
      <li class="active"><a href="index.html">Αρχή</a></li>
      <li><a href="#">Σχετικά με μας</a></li>
      .....
      <li><a href="contact.html">Επικοινωνία</a></li>
    </ul>
  </nav>
</header>

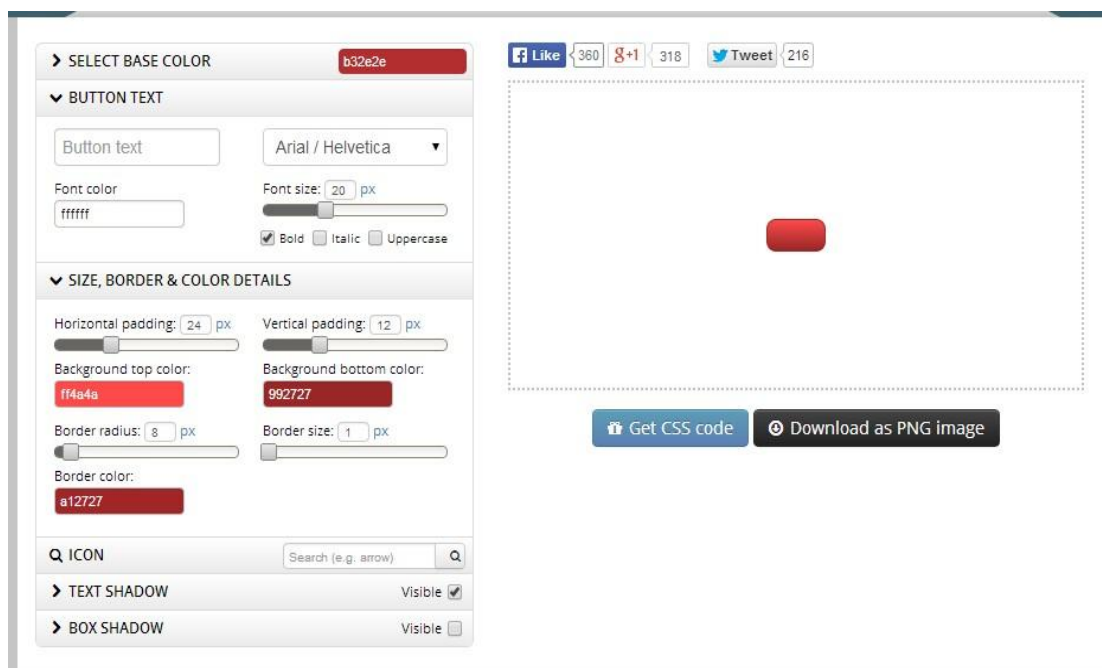
                                CSS
.header { padding: 32px 32px 16px; }
#logo { width: 249px; float: left; }
#logo a { height: 48px; display: block; background: url(images/logo.png) no-repeat 0 0; font-size: 0;
line-height: 0; text-indent: -4000px; }
#navigation { height: 35px; line-height: 35px; float: right; position: relative; z-index: 20; }
#navigation ul { list-style: none; list-style-position: outside; font-size: 13px; text-shadow:
rgba(255,255,255,0.5) 0px 1px 1px; }
.....
#navigation ul li.first ul li a { border-radius: 0px; -moz-border-radius: 0px; -webkit-border-radius:
0px; -o-border-radius: 0px; }

```

Στην κλάση .header ορίζονται οι διαστάσεις της επικεφαλίδας και τα αντίστοιχα id selector όπως το #logo, τα στοιχεία πλοήγησης. Με αυτό τον τρόπο στο παραπάνω κώδικα HTML5, όταν δηλώνουμε το id, σε μία ετικέτα το πρόγραμμα πλοήγησης, λαμβάνει τις ιδιότητες εμφάνισης από το αρχείο CSS. Βλέπουμε ότι το τμήμα της

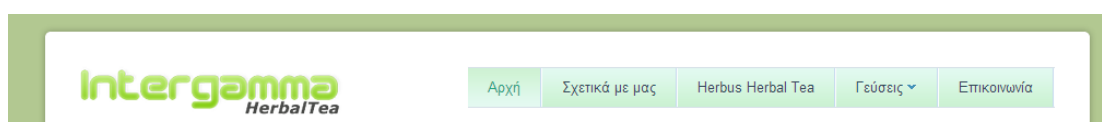
πλοήγησης ορίζονται αρκετές ιδιότητες ανάλογα με το αν το στοιχείο της λίστας έχει επιλεγθεί, αποτελεί πρώτο ή δεύτερο επίπεδο κοκ.

Επίσης για τον χρωματισμό και την διακόσμηση των επιλογών των μενού, έχουν επιλεγθεί αρχεία εικόνων ρηγμε διαβάθμιση χρώματος. Η κατασκευή των εικόνων έγινε μέσω της υπηρεσίας που προσφέρεται στη σελίδα <http://buttonoptimizer.com/>



Εικόνα 0-1 Δημιουργία εικόνας διαβάθμισης

Το αντίστοιχο αποτέλεσμα για την επικεφαλίδα, φαίνεται στην εικόνα που ακολουθεί.



Εικόνα 0-2 Επικεφαλίδα

4.0.2. Κύριο μέρος

Αποτελείται από κύριο μέρος που περιλαμβάνει τη εικόνα παρουσίασης και τρεις ενότητες (sections) όπως ορίζονται στην HTML5: το εισαγωγικό κείμενο, την περιοχή παρουσίασης και το τμήμα πληροφοριών. Ο κώδικας για την εικόνα παρουσίασης είναι:

```
HTML
<div class="imageshow">
  
</div>

CSS
.imageshow { width: 892px; height: 231px; margin: 0 auto; border-radius: 5px; -moz-border-radius: 5px; -webkit-border-radius: 5px; -o-border-radius: 5px; overflow: hidden; margin-bottom: 27px; }
```

Στο css ορίζουμε τα χαρακτηριστικά της όπως ύψος, πλάτος, περιθώρια και μορφή περιθωρίων, ενώ στο αρχείο htmlορίζουμε την θέση της εικόνας για εμφάνιση.Το αποτέλεσμα φαίνεται στην εικόνα που ακολουθεί:



Εικόνα 0-3Εικόνα παρουσίασης

Η ενότητα του εισαγωγικού κειμένου ορίζεται ως εξής:

```
HTML
<section class="intro">
<h3><a href="#">ΗΕταιρίαμας</a></h3>
<p>
    Η INTERGAMMA ΑΕ είναι μία νέα Ελληνική εμπορική εταιρεία που ξεκίνησε
    την δραστηριότητα το 2008 στην Αθήνα, με μια γκάμα εισαγόμενων προϊόντων
    όπως ζαχαρώδη, παιχνίδια, είδη οικιακής χρήσης και πολλές άλλες κατηγορίες,
    .....
    .....
</p>
</section>
CSS
section { padding: 0 32px 27px; }
.intro h3 { padding-bottom: 14px; }
.intro p { text-align: justify; padding-bottom: 10px; font-size: 13px;}
```

Το εισαγωγικό κείμενο όπως εμφανίζεται στην αρχική σελίδα είναι:

Η Εταιρία μας

Η INTERGAMMA ΑΕ είναι μία νέα Ελληνική εμπορική εταιρεία που ξεκίνησε την δραστηριότητα το 2008 στην Αθήνα, με μια γκάμα εισαγόμενων προϊόντων όπως ζαχαρώδη, παιχνίδια, είδη οικιακής χρήσης και πολλές άλλες κατηγορίες, τα οποία διακινούσε σε ένα ιδιαίτερα ευρύ δίκτυο πωλήσεων σε όλη την Ελλάδα. Σήμερα η εταιρεία επεκτείνεται στο λανσάρισμα δικών της προϊόντων παραγωγής στη Ελληνική αγορά και ετοιμάζεται για εξαγωγές των προϊόντων αυτών σε χώρες του εξωτερικού.

Η αγάπη και η γνώση για τα βότανα αλλά παράλληλα και ο σεβασμός για τον άνθρωπο , οδήγησαν την εταιρεία να καταλήξει στην παραγωγή ενός καθαρά ελληνικού αφεψήματος με στόχο και σκοπό να γευτεί ο καταναλωτής τις ευεργετικές ιδιότητες των βοτάνων μέσα από μια μοντέρνα και φιλική προς το περιβάλλον συσκευασία των 500ml. Η δημιουργία των σημερινών εγκαταστάσεων έχει γίνει σύμφωνα με τις απαιτήσεις της διεθνούς αγοράς και με βάση τη διεθνή νομοθεσία αλλά και πρακτική αναφορικά με την υγιεινή παραγωγή των προϊόντων.

Το Herbous Herbal Tea παράγεται με όλο σχεδιασμό και εφαρμογή συστήματος διασφάλισης ποιότητας κατά ISO και μελέτης ανάλυσης των κρίσιμων σημείων επικινδυνότητας κατά hazard (HACCP). Τον Απρίλιο του 2012 ξεκίνησε η διανομή και διάθεση του προϊόντος σε όλη την Ελλάδα σε περίπτερα και φιλικά και σήμερα βρίσκεται στα ράφια μεγάλων αλυσίδων.

Εικόνα 0-4Εισαγωγικό κείμενο

Τα γενικά χαρακτηριστικά μιας ενότητας ορίζονται μία φορά με την περιγραφή section και αντίστοιχα στις κάτω γραμμές δημιουργούμε τις κλάσεις. Τα χαρακτηριστικά του κειμένου για την ενότητα, όπως η επικεφαλίδα, παράγραφος, στοίχιση, μέγεθος γραμματοσειρά και αποστάσεις περιγράφονται στο CSS με την κλάση “intro”. Αντίστοιχα για την ενότητα presentation ισχύει:

```

HTML

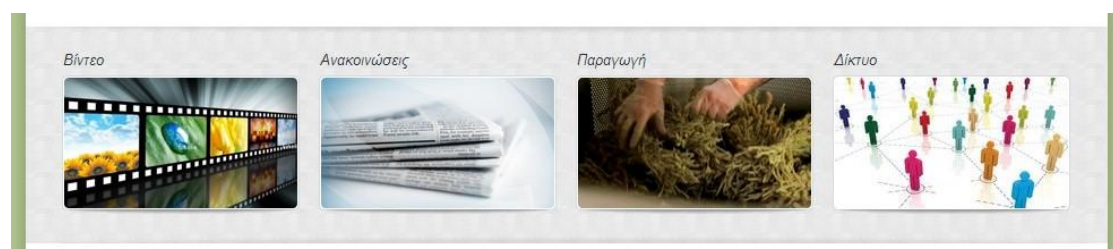
<section class="presentation">
<span class="shadow-t"></span>
<div class="cl">&nbsp;</div>
<div class="entries">
  <div class="entry">
    <h3>Βίντεο</h3>
    <a href="videos.html"></a>
    <span class="shadow"></span>
  </div>
  .....
</div>
</section>

CSS

.presentation { background: url(images/box.png) repeat 0 0; padding: 19px 32px 29px 32px;
position: relative; height: 100%; }
.presentation span.shadow-t { width: 956px; height: 11px; background: url(images/box-shadow-
t.png) no-repeat 0 0; position: absolute; top: 0; left: 0; display: block;}
.presentation span.shadow-b { width: 956px; height: 11px; background: url(images/box-shadow-
b.png) no-repeat 0 0; position: absolute; bottom: -4px; left: 0; display: block;}
.presentation h3 { position: absolute; left: 0px; top: -25px; font-size: 13px;font-style:italic}

```

Βλέπουμε ότι χαρακτηριστικά όπως η σκιά της περιοχής, το χρώμα παρασκηνίου ορίζονται στο CSS και καλούνται από τον HTML κώδικα μέσω των αντίστοιχων κλάσεων. Η ενότητα παρουσίασης όπως έχουμε αναφέρει περιλαμβάνει τα βίντεο, τις ανακοινώσεις, πληροφορίες για την παραγωγή και το δίκτυο συνεργατών. Ουσιαστικά αποτελεί έναν εναλλακτικό τρόπο πλοήγησης σε δευτερεύουσες σελίδες του ιστοτόπου.



Εικόνα 0-5 Ενότητα παρουσίασης

Τέλος η ενότητα των πληροφοριών υλοποιείται ως εξής:

```
HTML

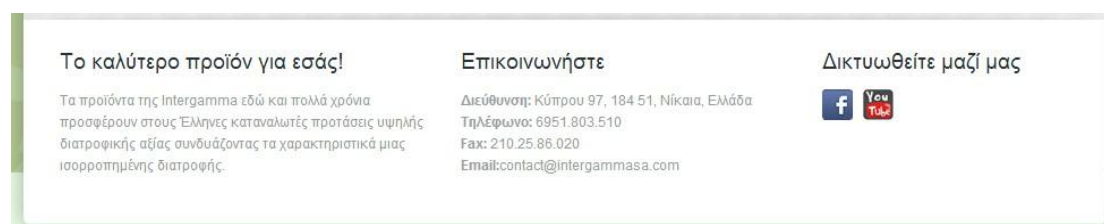
<section class="informations">
<divclass="widget">
  <h3>Το καλύτερο προϊόν για εσάς!</h3>
  <p>Τα προϊόντα της Intergamma εδώ και πολλά χρόνια προσφέρουν στους Έλληνες
  καταναλωτές προτάσεις υψηλής διατροφικής αξίας συνδυάζοντας τα χαρακτηριστικά
  μιας ισορροπημένης διατροφής.
  </p>
</div>
<div class="widget contact-widget">
  <h3>Επικοινωνήστε</h3>
  <ul>
  <li><strong>Διεύθυνση: </strong>Κύπρου 97, 184 51, Νίκαια, Ελλάδα</li>
  <li><strong>Τηλέφωνο:</strong>6951.803.510</li>
  .....
  </ul>
</div>
<div class="widget socials-widget">
  <h3>Δικτυωθείτε μαζί μας</h3>
  <p><a href="https://www.facebook.com/HerbalTeaIntergamma" class="facebook-
  ico">facebook</a>
  <a href="https://www.youtube.com/user/Herbalthegreektea?feature=watch"
  class="youtube-ico">youtube</a>
  </p>
</div>
</section>

CSS

.informations { border-top: 3px solid #fff; background: url(images/informations.png) repeat-x 0 0;
padding-top: 23px; }
.informations h3 { padding-bottom: 14px; }
.informations p { padding-bottom: 15px; }
.informations .widget { float: left; width: 335px; }
.informations .widget ul { list-style: none; list-style-position: outside; }
.informations .contact-widget { width: 300px; }
.informations .widget + .widget { padding-left: 19px; }
.informations .socials-widget { width: 219px; }
.informations .socials-widget a { width: 27px; height: 27px; background: url(images/socials.png)
no-repeat 0 0; float: left; margin-right: 9px; font-size: 0; line-height: 0; text-indent: -4000px; }
.informations .socials-widget a:hover { opacity: 0.7;
-moz-opacity: 0.7;
filter:alpha(opacity=70); }
.informations .socials-widget a.facebook-ico { background-position: 0 0; }
.informations .socials-widget a.youtube-ico { background-position: -36px 0; }
```

Παρατηρούμε αποτελείται από τρεις στήλες που περιέχουν κατά σειρά πληροφορίες, στοιχεία επικοινωνίας, και στοιχεία κοινωνικής δικτύωσης. Οι πληροφορίες αφορούν καθαρό κείμενο, τα στοιχεία επικοινωνίας μία λίστα. Η περιοχή κοινωνικής δικτύωσης περιέχει τις αντίστοιχες αναφορές στα δίκτυα που έχει παρουσία η εταιρία.

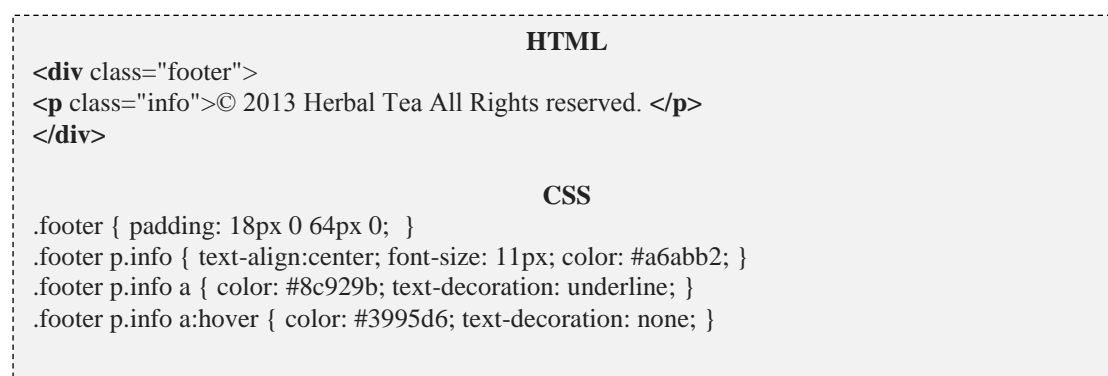
Το οπτικό αποτέλεσμα αντίστοιχα είναι:



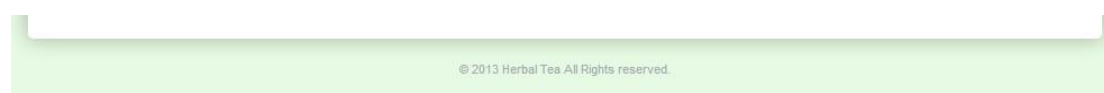
Εικόνα 0-6 Ενότητα πληροφοριών

4.0.3. Υποσέλιδο

Το υποσέλιδο περιέχει τις πληροφορίες πνευματικής ιδιοκτησίας και περιέχει απλά μία δήλωση στο κάτω μέρος της σελίδας. Ο κώδικας είναι:



Ενώ το αποτέλεσμα στην σελίδα μας:



Εικόνα 0-7 Υποσέλιδο

Σελίδα “aboutus.html”

Η σελίδα που παρουσιάζει κάποια στοιχεία σχετικά με την εταιρία ακολουθεί την δομή της αρχικής σελίδας όπως και όλες οι σελίδες, με μικρές διαφορές. Αυτές είναι ότι αποτελείται από δύο ενότητες: μία με το εισαγωγικό κείμενο και αναφορά στις εγκαταστάσεις και μία με ιστορικά δεδομένα.

Επίσης έχει αφαιρεθεί η ενότητα παρουσίασης, για να διατηρηθεί σε λογικά πλαίσια το ύψος της σελίδας και να μπορούν να επικεντρωθούν οι επισκέπτες στο κείμενο.

Σε επίπεδο κώδικα δεν υπάρχει κάτι αξιοσημείωτο, εκτός από το στυλ του μεγάλου αρχικού γράμματος που υλοποιήθηκε μέσω CSS και είναι:

HTML

```
<section class="aboutus">
.....
</section>
```

CSS

```
.aboutus p { font-size: 1em; line-height: 1.25em; margin: 0; padding: 10px; text-align: justify;}
.aboutus p + p {}
.aboutus p:first-letter { font-size: 4.5em; line-height: 1em; font-weight: 400; float: left; margin: 0 0.107em 0 0;}
.aboutus p:first-line { font-weight: 800;}
.aboutus p + p:first-letter { font-size: 1em; line-height: 1.25em; float: none; margin: 0;}
.aboutus p + p:first-line { font-weight: 400;}
```

Όπως φαίνεται και στην εικόνα που ακολουθεί με το οπτικό αποτέλεσμα, ο κώδικας CSS ευθύνεται για το εφέ του κεφαλαίου γράμματος στην αρχή των ενοτήτων. Ουσιαστικά επιλέγουμε διαφορετική διαμόρφωση και στοίχιση για την πρώτη παράγραφο και επαναφέρουμε τις ρυθμίσεις για τις επόμενες παραγράφους.



The screenshot shows the website for Intergamma Herbal Tea. At the top left is the logo 'Intergamma Herbal Tea'. To the right is a navigation menu with buttons for 'Αρχή', 'Σχετικά με μας', 'Herbus Herbal Tea', 'Γεύσεις', and 'Επικοινωνία'. The main content area features a large green globe logo with the text 'Intergamma Σχετικά με εμάς'. Below this is a section titled 'Λίγα λόγια για την εταιρία μας.' followed by a paragraph of text starting with 'Η INTERGAMMA ΑΕ είναι μία νέα Ελληνική εμπορική εταιρεία...'. There is also a section titled 'Ιστορικά δεδομένα.' with a small image and text starting with 'Ολα άρχισαν το 1924 όταν ο μπάρμπα Κώστας Τενεδιός...'. The website has a clean, green-themed design.

Εικόνα 0-8 Σελίδα "Σχετικά με εμάς"

Σελίδα "herbaltea.html"

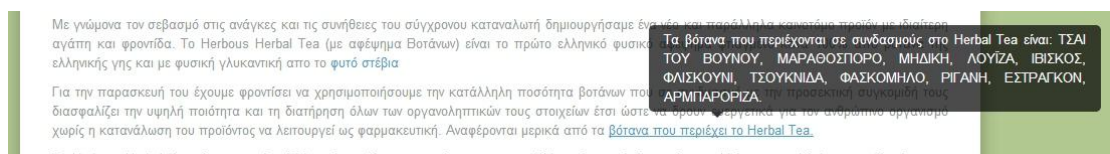
Η σελίδα Herbus Herbal Tea περιέχει πληροφορίες σχετικά με το τσάι, την παραγωγή του και τα πλεονεκτήματά του. Σε αυτή την σελίδα χρησιμοποιήσαμε δύο νέα

στοιχεία διακόσμησης, αλλά και το νέο χαρακτηριστικό της html5 που αφορά προσαρμοσμένα χαρακτηριστικά για την δημιουργία αναδυόμενων παραθύρων πληροφόρησης (tooltips).

```
HTML
<a href="#" class="tooltip" data-tooltip="Τα βότανα που περιέχονται σε συνδυασμούς στο Herbal Tea είναι: ΤΣΑΙ ΤΟΥ ΒΟΥΝΟΥ, ΜΑΡΑΘΟΣΠΟΡΟ, ΜΗΔΙΚΗ, ΛΟΥΪΖΑ, ΙΒΙΣΚΟΣ, ΦΛΙΣΚΟΥΝΙ, ΤΣΟΥΚΝΙΔΑ, ΦΑΣΚΟΜΗΛΟ, ΡΙΓΑΝΗ, ΕΣΤΡΑΓΚΟΝ, ΑΡΜΠΑΡΟΡΙΖΑ."
<span>βότανα που περιέχει το HerbalTea.</span>
</a>

CSS
.tooltip{display: inline; position: relative;}
.tooltip:hover:after{background: #333; background: rgba(0,0,0,.7); border-radius: 5px; bottom: 26px; color: #fff; content: attr(data-tooltip); left: 20%; padding: 5px 15px; position: absolute; z-index: 98; width: 420px;}
.tooltip:hover:before{border: solid; border-color: #333 transparent; border-width: 6px 6px 0 6px; bottom: 20px; content: ""; left: 50%; position: absolute; z-index: 99;}
```

Για τα tooltips, δημιουργήσαμε την αντίστοιχη κλάση την οποία καλούμε από το html αρχείο. Εδώ βλέπουμε ότι ορίζουμε μία νέα ιδιότητα της κλάσης με το όνομα “data-tooltip” στην οποία αναθέτουμε το αλφαριθμητικό που θα περιέχει το αναδυόμενο παράθυρο. Τα χαρακτηριστικά της ιδιότητας ορίζονται στο CSS με την γραμμή content: attr(data-tooltip). Το κείμενο που είναι μέσα στις ετικέτες ορίζουν αντίστοιχα τον σύνδεσμο στο κείμενο της σελίδας μας. Έτσι με το πέρασμα του ποντικιού πάνω από το σύνδεσμο εμφανίζεται το tooltip όπως φαίνεται στην εικόνα.



Εικόνα 0-9 Αναδυόμενα παράθυρα πληροφορίας

Η δεύτερη αλλαγή έχει να κάνει με την παρουσίαση των λιστών αρίθμησης.

HTML

```
<div id="tlist">
<ol>
  <li><p><em>Δεν έχει ανταγωνισμό</em> και σύγκριση με κανένα προϊόν της αγοράς.
Είναι νέα κατηγορία προϊόντος<p></li>
.....
</ol>
```

CSS

```
#tlist { width:880px; }
#tlist ol { font-style:italic; font-size:18px; color:#999ea2; padding-left: 40px;}
#tlist ol li { }
#tlist ol li p {padding:8px; font-style:normal; font-family:Arial; font-size:13px; border-left: 1px
solid #999; }
#tlist ol li p em { display:block; font-weight:bold;}
```

Με το αναγνωριστικό #tlist δημιουργούμε νέα χαρακτηριστικά για τις λίστες της σελίδας μας. Ορίζουμε αποστάσεις και γραμματοσειρές για κάθε στοιχείο της ταξινομημένης λίστας μας. Οι αριθμοί θα εμφανίζονται με πλάγιο στυλ, ενώ το κείμενο μέσα σε ετικέτες θα είναι με έντονο στυλ ενώ η δήλωση display:block τα εμφανίζει σε μία γραμμή.

Το Herbous Herbal Tea είναι το πρώτο Ελληνικό προϊόν που παρόμοιο του στην Ελληνική αγορά δεν υπάρχει αλλά και σε ολόκληρη την Ευρώπη, οι πολλές δυνατότητες και η δυναμική του το καθιστά από μόνο του μια νέα και μοναδική κατηγορία προϊόντος. Είναι μοναδικό γιατί:

1. *Δεν λανσάρουμε ακόμα ένα τσάι στην αγορά αλλά ένα αφέψημα.*
Το Herbous Herbal Tea παράγεται ακριβώς όπως παλιά. Είναι αφέψημα, που σημαίνει απλή εμβάπτιση σε ζεστό νερό για ορισμένο χρόνο. Μάλιστα έχουμε προσέξει να είναι σε αυστηρά όρια η θερμοκρασία και ο χρόνος εμβάπτισης έτσι ώστε να έχουμε το καλύτερο δυνατό αποτέλεσμα. Αυτή η διαδικασία είναι μικρότερης απόδοσης σε αρωματικά και άλλα στοιχεία των βοτάνων. Για το λόγο αυτό είμαστε υποχρεωμένοι να χρησιμοποιούμε, συγκριτικά, μεγαλύτερες ποσότητες για να κάνουμε το αφέψημά μας. Η μέθοδος αυτή, όμως, είναι πιο ασφαλής καθώς τα διαθρεπτικά στοιχεία λαμβάνονται μέσα από τις κυτταρικές μεμβράνες των φυτών (βοτάνων) που λειτουργούν σαν φίλτρα και μας αποδίδουν μόνο τα θετικά τους στοιχεία. Ακόμη το διαλυτικό μέσο είναι άριστης ποιότητας νερό από πηγές στους πρόποδες του όρους Βέρμιο.
2. *Δεν έχει ανταγωνισμό*
και σύγκριση με κανένα προϊόν της αγοράς. Είναι νέα κατηγορία προϊόντος
3. *Είναι το μοναδικό προϊόν*
με βότανα που προσφέρει ευεργετικές ιδιότητες για τον άνθρωπο και όλες συγκεντρωμένες σε ένα μπουκάλι.
4. *Στηρίζουμε την Ελληνική οικονομία και*
ενισχύουμε την αγροτική οικονομία γιατί όλα τα βότανα συλλέγονται και καλλιεργούνται στην Ελλάδα.

Εικόνα 0-10 Λίστες στην σελίδα herbusherbaltea

Σελίδα “flavors.html”

Η σελίδα flavors.html, περιέχει πληροφορίες σχετικά με τις ποικιλίες του προϊόντος, αλλά δίνει και τη δυνατότητα παραγγελίας ή εκδήλωσης ενδιαφέροντος. Για τη δημιουργία της σελίδας, έχουν υιοθετηθεί όλα τα χαρακτηριστικά από τις προηγούμενες σελίδες, αλλά έχει προστεθεί μία φόρμα που υιοθετεί στοιχεία από την HTML5, κάνει επικύρωση μέσω javascript και για την αποστολή έχει χρησιμοποιηθεί η Php.

Η φόρμα ορίζεται με την δήλωση:

```
HTML
<form id="order" class="order-form" action="#" method="post" name="order"
oninput="x.value=parseInt(quantity.value)">
<div class="field">
  <label for="quantity">Ποσότητα:</label>200
  <input type="range" id="quantity" name="quantity"
    min="200" max="1000" step="5" value="600">1000 =
  <output name="x" for="quantity"></output>
</div>
</form>
```

Με την μέθοδο post δηλώνουμε τον τρόπο με τον οποίο θα αποσταλούν τα δεδομένα μας ενώ η συμβάν oninput ανανεώνει τη τιμή της ποσότητας. Η ποσότητα είναι ένα πεδίο της φόρμας που παίρνει τιμές από 200-1000 οι οποίες μεταβάλλονται από έναν δρομέα. Με την μεταβολή του δρομέα η τιμή διαφοροποιείται και εμφανίζεται.



Εικόνα 0-11 Πεδίο ορισμού ποσότητας

Με τον ίδιο τρόπο ορίζονται και τα υπόλοιπα πεδία στη φόρμα μας:

HTML

```
<div class="field">
<label>Γεύσεις:</label>
<label for = "chklemon">Λεμόνι</label><input type = "checkbox" id = "chklemon" value =
"lemon" class="txtcb"/>
<label for = "chkestragon">Εστραγκόν</label><input type = "checkbox" id = "chkestragon"
value = "estragon" class="txtcb"/>
<label for = "chkrodi">Ρόδι</label><input type = "checkbox" id = "chkpomegrande" value =
"pomegrande" class="txtcb"/>
<label for = "chkestragon">Μήλο</label><input type = "checkbox" id = "chkapple" value =
"apple" class="txtcb"/>
</div>
<div class="field">
<label>Σημείο Λιανικής:</label>
<input list="suppliers" name="supplier">
<datalist id="suppliers">
  <option value="Αθήνα">
  <option value="Θεσσαλονίκη">
  <option value="Πάτρα">
  <option value="Ηράκλειο">
</datalist>
</div>
<div class="field">
<label for="email">E-mail:</label><input type="email" id="email" name="email" class="txt">
</div>
<div class="field">
.....
</div>
<button id="send">Αποστολή</button>
```

Το πεδίο γεύσεις είναι τύπου checkbox και έχει μία δήλωση `<input>` ανά γεύση. Το πεδίο σημείο λιανικής, είναι μία λίστα που μπορεί να πάρει μόνο μία τιμή από αυτές που δηλώνονται μέσω της ετικέτας `<option>`. Το email είναι πεδίο κειμένου.

Η αποστολή της φόρμας γίνεται μόλις πατηθεί το κουμπί «Αποστολή». Ουσιαστικά μόλις πατηθεί το κουμπί, γίνονται οι έλεγχοι μέσω Javascript και καλείται το αρχείο σεναρίου της ρηρόπως θα δούμε.


```

$("#send").on("click", function(){
    var emailval = $("#email").val();
    var phoneval = $("#phone").val();
    var mailvalid = validateEmail(emailval);
    var phonevalid= validatePhone(phoneval);

    if(mailvalid == false) {
        $("#email").addClass("error");
    }
    else if(mailvalid == true){
        $("#email").removeClass("error");
    }

    if(phonevalid == false) {
        $("#phone").addClass("error");
    }
    .....
    .....
    if(mailvalid == true &&phonevalid == true && msglen >= 4) {
        $("#send").replaceWith("<em>ηφόρμααποστέλλεται...</em>");

        $.ajax({
            type: 'POST',
            url: 'send_order.php',

```

Αποτελείται από δύο συναρτήσεις για τον έλεγχο της ορθής καταχώρισης του email, και του τηλεφώνου. Στο συμβάν onclick, δηλαδή όταν γίνεται αποστολής της φόρμας, λαμβάνονται οι τιμές της φόρμας και καταχωρούνται σε αντίστοιχες μεταβλητές. Στη συνέχεια καλούνται οι συναρτήσεις οι οποίες επιστρέφουν True/False ανάλογα με το αν η τιμή του πεδίου ταιριάζει με μια συγκεκριμένη έκφραση. Στην περίπτωση του τηλεφώνου η έκφραση `reg = /\d{10}$/`; δηλώνει δεκαψήφιο τηλεφωνικό νούμερο. Κατόπιν αν η συνάρτηση έχει επιστρέψει False τότε το συγκεκριμένο πεδίο, μπαίνει σε κατάσταση “error” και εμφανίζεται με κόκκινο χρώμα (βλ. εικόνα 4-12). Αν όλοι οι έλεγχοι πραγματοποιηθούν χωρίς σφάλμα τότε καλείται το σενάριο ‘`phpsend_order.php`’ για την τελική αποστολή.

Στην αρχή κάνουμε ανάκτηση των δεδομένων

```

<?php
    $usermail = $_POST['email'];
    $content = nl2br($_POST['msg']);
    $quantity = $_POST['quantity'];
    $phone = $_POST['phone'];
    $supplier = $_POST['supplier'];

```

Κατόπιν δημιουργείτε η δομή του μηνύματος και ελέγχεται η τιμή του πεδίου σημείο λιανικής. Ανάλογα με την τιμή, δηλώνεται και ο αντίστοιχος παραλήπτης.

```

$subject = "Εκδήλωση ενδιαφέροντος";
$headers = "From: " . strip_tags($usermail) . "\r\n";
$headers .= "Reply-To: " . strip_tags($usermail) . "\r\n";
$headers .= "MIME-Version: 1.0\r\n";
$headers .= "Content-Type: text/html;charset=utf-8 \r\n";

$msg = "<html><body style='font-family:Arial,sans-serif;'>";
$msg .= "<h2>Παραγγελία</h2>\r\n";
$msg .= "<p><strong>Από:</strong> ".$usermail."</p>\r\n";
$msg .= "<p><strong>Μήνυμα:</strong> ".$content."</p>\r\n";
$msg .= "<p><strong>Ποσότητα:</strong> ".$quantity."</p>\r\n";
$msg .= "<p><strong>Τηλέφωνο:</strong> ".$phone."</p>\r\n";
$msg .= "</body></html>";

if ($supplier == 'Αθήνα') {$sendto = "athens@hbt.gr";}
if ($supplier == 'Θεσσαλονίκη') {$sendto = "thessaloniki@hbt.gr";}
if ($supplier == 'Πάτρα') {$sendto = "patra@hbt.gr";}
if ($supplier == 'Ηράκλειο') {$sendto = "irakleio@hbt.gr";}

```

Τέλος γίνεται αποστολή του μηνύματος και επιστροφή της αντίστοιχης TRUE/FALSE τιμής, ανάλογα με το αν αυτή έγινε με επιτυχία.

```

if (@mail($sendto, $subject, $msg, $headers)) {
    echo "true";
} else {
    echo "false";
}
?>

```

Σελίδα “contact.html”

Σε αυτή την σελίδα μπορεί κάποιος να επικοινωνήσει για γενικά θέματα. Συνεπώς υπάρχει και εδώ η αντίστοιχη φόρμα επικοινωνίας. Για την επαλήθευση των δεδομένων της φόρμας, χρησιμοποιήθηκαν οι τρεις νέες ιδιότητες της HTML5: η autofocus, η required και η placeholder.

```

<form name="contact_form" id="cf" method="post" action="/scripts/send_message.php">
  <h1>Επικοινωνήστε μαζί μας!</h1>
  <label><span>Όνομα*:</span><input type="text" placeholder="Συμπληρώστε το
    Ονοματεπώνυμό σας" name="name" id="name" required autofocus>
  </label>
  <label><span>Πόλη*:</span>
    <input type="text" placeholder="Συμπληρώστε την πόλη σας" name="city"
      id="city" required>
  <input class="submitButton" type="submit" name="Submit" value="Αποστολή">
</form>

```


Με την autofocus, ο κέρσορας εστιάζει στο πεδίο Όνομα κατά την φόρτωση της σελίδας, ενώ η required καθιστά το πεδίο υποχρεωτικό. Τέλος η placeholder, εμφανίζει ένα βοηθητικό κείμενο για την ενημέρωση του χρήστη.

Το σενάριο send_message.php είναι παρόμοιο με αυτό της φόρμας παραγγελίας, αλλά διαφοροποιείται ως προς το αριθμό και είδος των πεδίων.

```
<?php
$name=$_POST['name'];
$city=$_POST['city'];
$phone=$_POST['phone'];
$useremail=$_POST['email'];
$content=$_POST['msg'];

$headers = "From:" . $useremail . "\r\n";
$headers .= "Content-type: text/plain; charset=UTF-8" . "\r\n";

$sendto = "info@hbt.gr";
$subject = "Μήνυμα από τη φόρμα επικοινωνίας";
$message = " Name: " . $name . "\r\n City: " . $city . "\r\n Phone: " . $phone . "\r\n Email: " . $useremail . "\r\n Μήνυμα: " . $content;

if (@mail($sendto,$subject,$message, $headers))
{
    echo true;
}else{
    echo false;
}
?>
```

Το αποτέλεσμα των ανωτέρω φαίνεται στην εικόνα που ακολουθεί:

The image shows a contact form with the following elements:

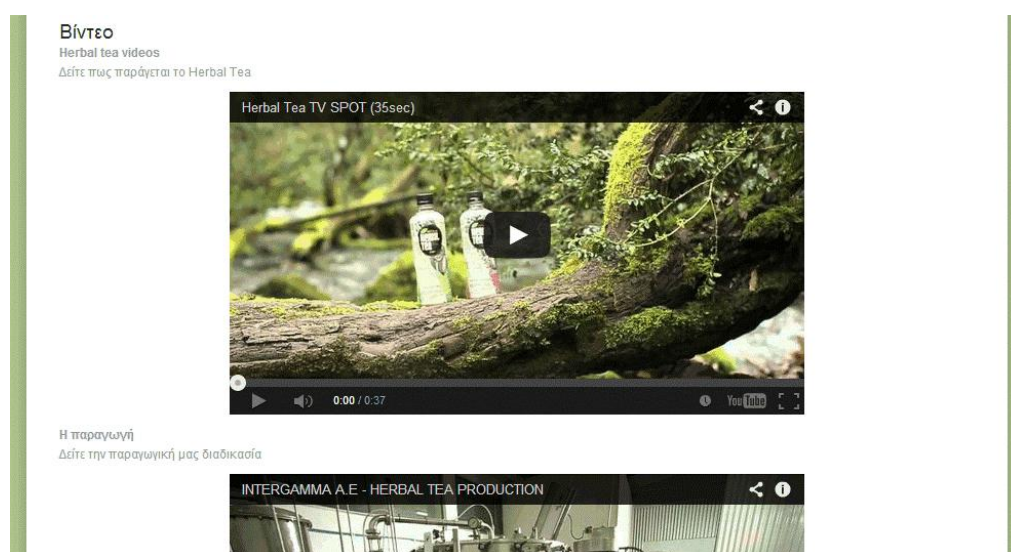
- Title:** Επικοινωνήστε μαζί μας!
- Subtitle:** Συμπληρώνοντας την ακόλουθη φόρμα
- Fields:**
 - Όνομα*:** Συμπληρώστε το Ονοματεπώνυμό σας
 - Πόλη*:** Συμπληρώστε την πόλη σας
 - Τηλέφωνο:** Συμπληρώστε ένα τηλέφωνο επικοινωνίας
 - E-Mail*:** Συμπληρώστε την ηλεκτρονική σας διεύθυνση
 - Μήνυμα*:** A large text area for the message.
- Button:** Αποστολή

Εικόνα 0-13 Φόρμα επικοινωνίας

Δευτερεύουσες σελίδες

Είναι οι σελίδες που εμφανίζονται με επιλογή από την περιοχή παρουσίασης και αφορούν δευτερεύουσες πληροφορίες σχετικά με το προϊόν και την εταιρία. Αυτές είναι:

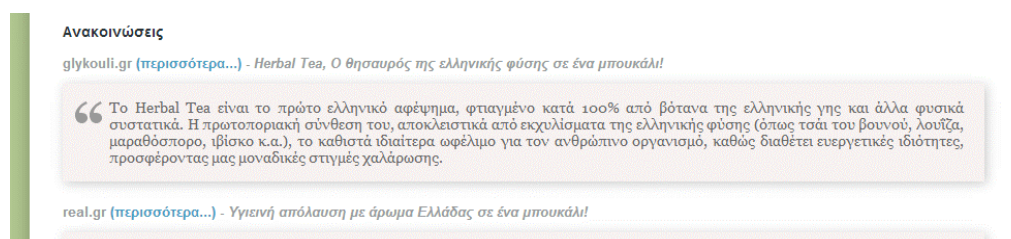
Βίντεο: Εμφανίζει τα βίντεο της εταιρίας, από το youtube. Η κλήση των βίντεο, γίνεται μέσω του κώδικα `<iframe width="560" height="315" src="http://www.youtube.com/embed/hCF2NELXMLw" frameborder="0" allowfullscreen></iframe>`.



Εικόνα 0-14 Σελίδα με βίντεο

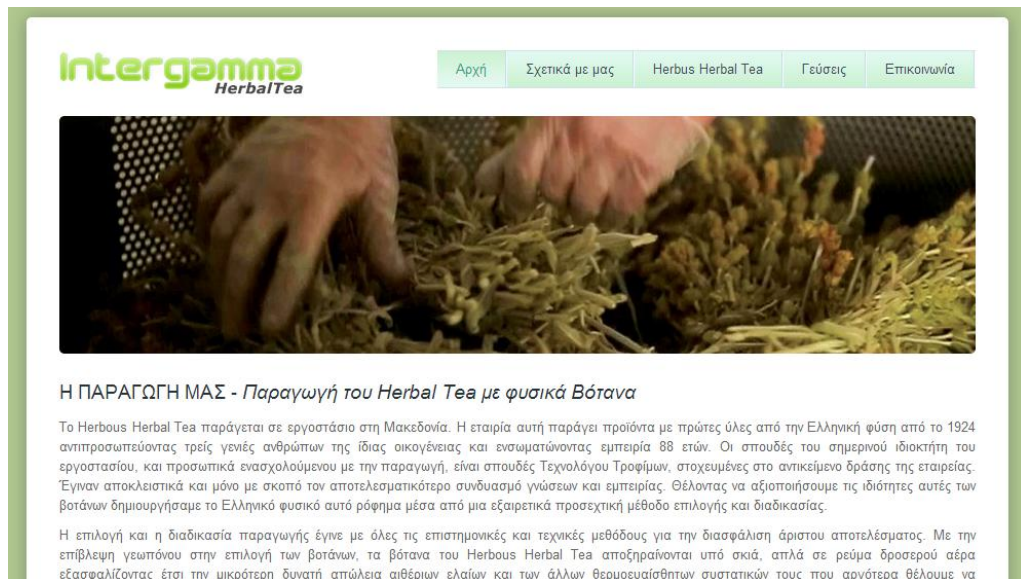
- **Ανακοινώσεις:** Περιέχει ανακοινώσεις για τα προϊόντα από τον ηλεκτρονικό τύπο. Για την εμφάνιση του σχολίου έχει γίνει η χρήση της ετικέτας `<blockquote></blockquote>`.

`<blockquote>`
Το HerbalTea είναι το πρώτο ελληνικό αφέψημα, φτιαγμένο κατά 100% από βότανα της ελληνικής γης και άλλα φυσικά συστατικά. Η πρωτοποριακή σύνθεση του, αποκλειστικά από εκχυλίσματα της ελληνικής φύσης
`</blockquote>`



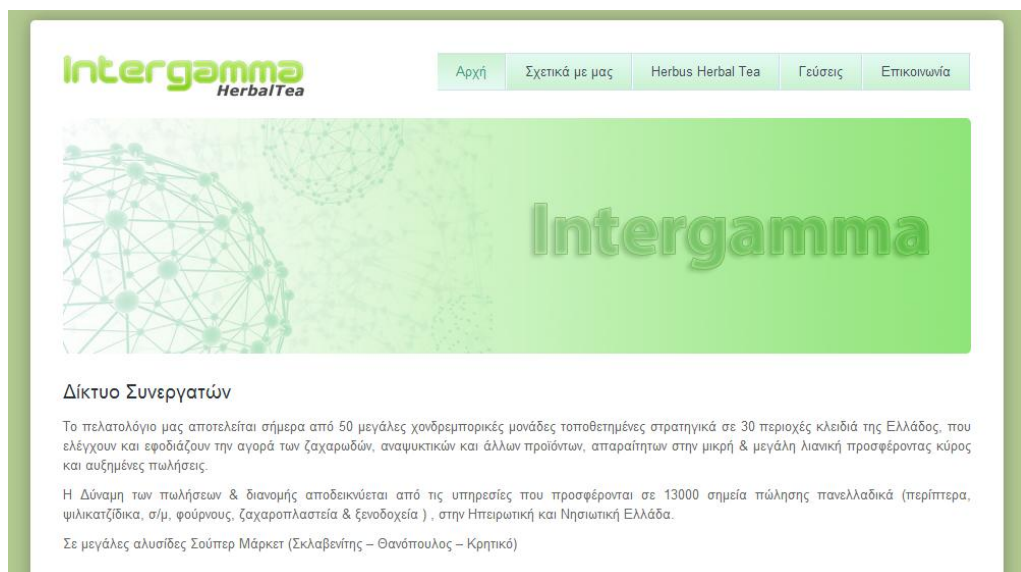
Εικόνα 0-15 Σελίδα με ανακοινώσεις

- **Παραγωγή:** Περιέχει πληροφορίες για την διαδικασία παραγωγής του προϊόντος.



Εικόνα 0-16 Σελίδα με τις πληροφορίες παραγωγής

- **Δίκτυο συνεργατών:** Περιέχει αντίστοιχες πληροφορίες με τους συνεργάτες και τα σημεία πώλησης.



Εικόνα 0-17 Σελίδα με το δίκτυο συνεργατών

Δημιουργία λογοτύπων και εικόνων

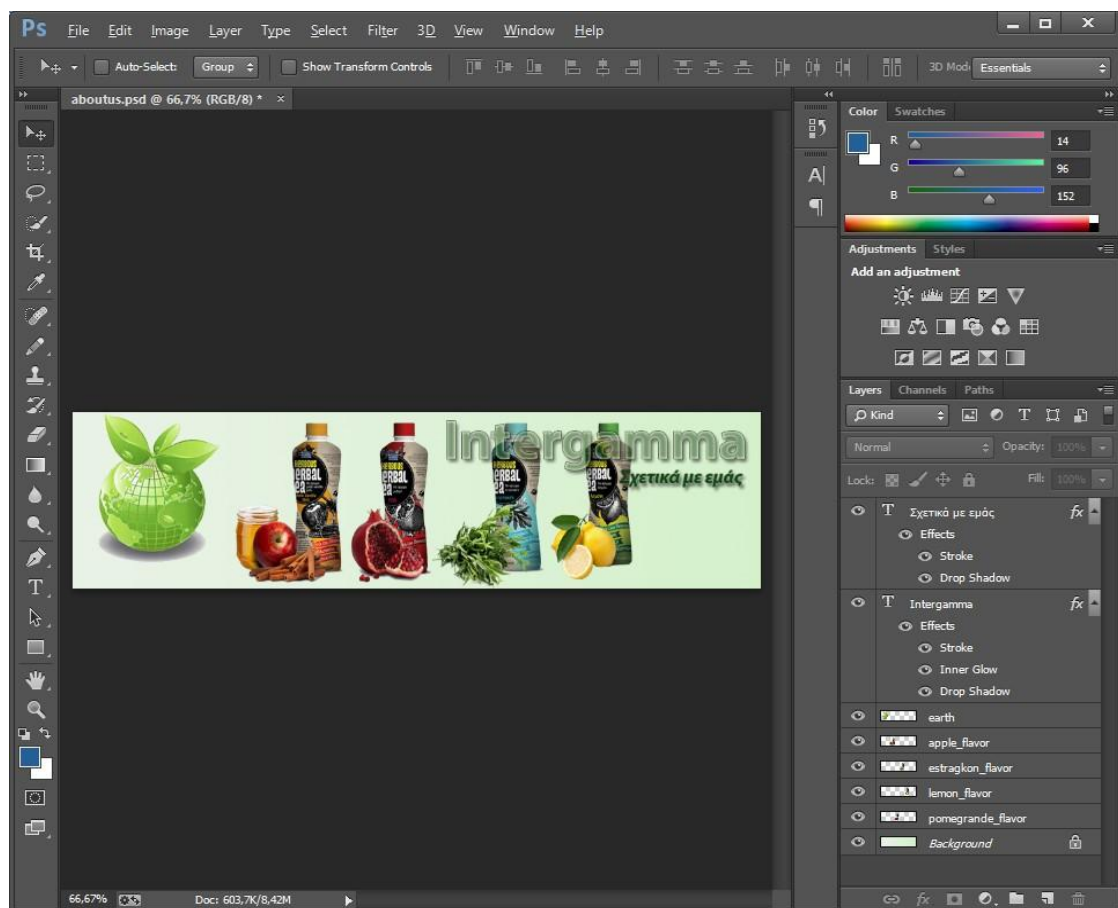
Για την δημιουργία των σελίδων, χρησιμοποιήθηκαν αρκετά γραφικά στοιχεία. Μερικά από αυτά όπως το παρασκήνιο, τα χρώματα των κουμπιών δημιουργήθηκαν με δωρεάν εργαλεία στο διαδίκτυο. Κάποιες από τις εικόνες όμως δημιουργήθηκαν με

τη χρήση του photoshop. Αυτές ήταν οι εικόνες στις επικεφαλίδες των αντίστοιχων σελίδωνόπως:

- aboutus.jpg
- flavors.jpg
- network.jpg

Στο photoshop χρησιμοποιήθηκε η τεχνική των επιπέδων (layers) για κάθε στοιχείο μιας εικόνας και κάποια από αυτά τα στοιχεία επαναχρησιμοποιήθηκαν για την παραγωγή κάποιας άλλης.

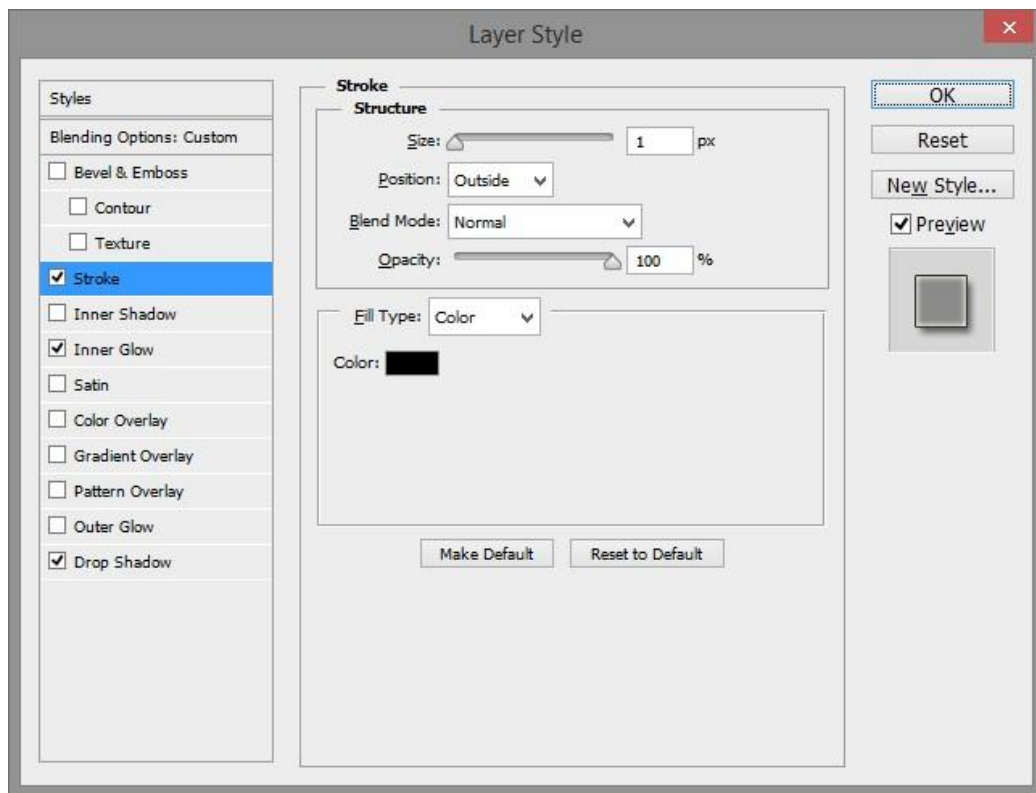
Στην εικόνα που ακολουθεί βλέπουμε τα στοιχεία που χρησιμοποιήθηκαν για την παραγωγή των εικόνων aboutus.jpg και flavors.jpg.



Τα επίπεδα που έχουν χρησιμοποιήθηκαν είναι αντίστοιχα:

- earth: Η εικόνα με την πράσινη γη
- apple_flavor: Η εικόνα με το μπουκάλι με τη γεύση μήλο. Αντίστοιχα οι άλλες γεύσεις μπήκαν στα επίπεδα lemon_flavor, estragkon_flavor, pomegrande_flavor και συνδέθηκαν μεταξύ τους.

- Intergamma: Περιέχει το αντίστοιχο κείμενο και έγινε η σχετική επεξεργασία και προσθήκη εφέ. Έτσι θέσαμε τα εφέ strokeγια το περίγραμμα, InnerGlowγια το γέμισμα και dropshadowγια την δημιουργία εφέ σκιάς.



- Σχετικά με εμάς: Περιέχει το αντίστοιχο κείμενο με την περιγραφή και περιέχει και αυτό περίπου τα ίδια εφέ.

Η παραγωγή των αντίστοιχων εικόνων που χρησιμοποιήθηκαν έγινε αποκρύπτοντας κάθε φορά τα επίπεδα που δεν χρειαζόμαστε στην τελική φωτογραφία.

Συμπεράσματα, μελλοντικές επεκτάσεις

Η εξέλιξη της τεχνολογίας στο διαδίκτυο, οδήγησε στην ανάγκη κατασκευής νέων προτύπων και έτσι την περίοδο που το επίσημο W3C ξεκίνησε το προσχέδιο της XHTML2, ταυτόχρονα ανεξάρτητοι κατασκευαστές δημιουργούν μία κοινότητα ανάπτυξης και ξεκινούν το προσχέδιο της HTML5 το οποίο παραδίδεται σε κάποια στιγμή στο W3C.

Θα μπορούσαμε να πούμε ότι η HTML 5.0 είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης για το διαδίκτυο σε συνδυασμό με τη χρήση JavaScript και CSS3. Επιπροσθέτως παρέχει αρκετές ευκολίες στους κατασκευαστές για ποικίλες υλοποιήσεις από τις πιο απλές έως τις πιο περίπλοκες συν το γεγονός ότι διατηρεί την συμβατότητα προς τα πίσω.

Η HTML5 είναι ένα ισχυρό εργαλείο, στα χέρια των κατασκευαστών για την δημιουργία εντυπωσιακών ιστοτόπων εκμεταλλευόμενη τα νέα χαρακτηριστικά της. Σε μεγάλο βαθμό έχει περιορίσει την ανάγκη πρόσθετων τεχνολογιών για την δημιουργία τους, αλλά δεν μπορεί να τα καταφέρει εξίσου καλά σε όλα τα επίπεδα.

Κατά την υλοποίηση πολλά από αυτά τα χαρακτηριστικά υιοθετήθηκαν, όπως η ενσωμάτωση των νέων ετικετών, η δημιουργία φορμών καθώς και ο έλεγχός τους. Η χρήση όμως και άλλων τεχνικών, όπως javascript και CSS αρχείων, κρίθηκε απαραίτητη για την αρτιότητα του αποτελέσματος.

Η διαδικασία κατασκευής ενός ιστοτόπου, δεν είναι μία απλή διαδικασία, αλλά αποτέλεσμα πολλών παραμέτρων και δεξιοτήτων. Θα πρέπει ο δημιουργός, να προσέξει για την καλή διατύπωση των κειμένων, τη σωστή δομή, αλλά και για την σωστή αισθητική του ιστοτόπου.

Κατά την διάρκεια δημιουργίας του δικτυακού τόπου στα πλαίσια της πτυχιακής εργασίας, προσπαθήσαμε να ικανοποιήσουμε αυτές τις απαιτήσεις. Εκτός από το περιεχόμενο, αρκετή έμφαση δόθηκε στην παρουσίαση μέσω των αρχείων στυλ εμφάνισης, αλλά και στην δομή.

Σίγουρα υπάρχουν αρκετά περιθώρια βελτίωσης, αλλά και επέκτασης των δυνατοτήτων του ιστοτόπου. Καθώς η χρήση της HTML5 ενδείκνυται κυρίως για την χρήση στατικών ιστοσελίδων, θα μπορούσε μελλοντικά να χρησιμοποιηθεί και δυναμικό περιεχόμενο για την βελτίωση των λειτουργιών.

Μια σημαντική βελτίωση είναι η διαχείριση του ιστορικού των παραγγελιών. Απαραίτητη προϋπόθεση για να γίνει αυτό είναι η χρήση βάσης δεδομένων για την αποθήκευση και διατήρηση των παραγγελιών. Ο χρήστης θα μπορεί να συνδεθεί στην περιοχή πελάτη, όπου θα του δίνεται η δυνατότητα παρακολούθησης της εκτέλεσης της παραγγελίας. Στην περίπτωση αυτή η εκτέλεση αυτών των λειτουργιών θα πρέπει να βασιστεί σε σενάρια ρημαζί με ένα RDBMS σύστημα.

Ένας δικτυακός τόπος, αποτελεί διαφήμιση για την εταιρία και τα προϊόντα που παράγει και όχι απλά ένα σημείο παρουσίας. Η δυνατότητα προσέλκυσης νέων αλλά και παλαιών χρηστών είναι απαραίτητη προϋπόθεση και στρατηγική μάρκετινγκ. Η δημιουργία μιας ενότητας ημερολογίου (blog), που θα ανανεωνόταν καθημερινά, με συμβουλές διατροφής και υγείας θα μπορούσε να βοηθήσει στην επισκεψιμότητα, συνεπώς και στην προβολή της εταιρίας. Επίσης η ολοένα και αυξανόμενη χρήση των κοινωνικών δικτύων από μεγάλη μερίδα ανθρώπων, οδηγεί στην ανάγκη προβολής μέσα από αυτά. Η ενότητα ημερολογίου, μπορεί κάλλιστα να εξυπηρετήσει αυτό το σκοπό με την αυτόματη δημοσίευση των καθημερινών άρθρων στα κοινωνικά δίκτυα, αυξάνοντας ακόμα περισσότερο την προβολή.

Βιβλιογραφία

- [1]. Anon., 2001. *XHTML Mobile Profile*. [Ηλεκτρονικό]
Available at: <http://www.openmobilealliance.org/tech/affiliates/wap/wap-277-xhtmlmp-20011029-a.pdf>
[Πρόσβαση 19 04 2014].
- [2]. Anon., 2008. *BASIC ORACLE ARCHITECTURE*. [Ηλεκτρονικό]
Available at: <http://monishsb.wordpress.com/2008/05/24/basic-oracle-architecture/>
[Πρόσβαση 21 04 2014].
- [3]. Anon., 2010. *XHTML - Print*. [Ηλεκτρονικό]
Available at: <http://www.w3.org/TR/xhtml-print/xhtml-print.pdf>
[Πρόσβαση 19 04 2014].
- [4]. Anon., 2013. *Network socket*. [Ηλεκτρονικό]
Available at: http://en.wikipedia.org/wiki/Network_socket
[Πρόσβαση 19 04 2014].
- [5]. Anon., 2014. *Documentation*. [Ηλεκτρονικό]
Available at: <http://docs.phalconphp.com/en/latest/index.html>
[Πρόσβαση 19 04 2014].
- [6]. Anon., 2014. *Javascript*. [Ηλεκτρονικό]
Available at: <http://el.wikipedia.org/wiki/JavaScript>
[Πρόσβαση 18 04 2014].
- [7]. Anon., 2014. *jQuery*. [Ηλεκτρονικό]
Available at: <http://en.wikipedia.org/wiki/JQuery>
[Πρόσβαση 18 04 2014].
- [8]. Anon., 2014. *MySQL Architecture*. [Ηλεκτρονικό]
Available at: http://www.datadisk.co.uk/html_docs/mysql/architecture.htm
[Πρόσβαση 19 04 2014].
- [9]. Anon., 2014. *PHP*. [Ηλεκτρονικό]
Available at: <http://el.wikipedia.org/wiki/PHP>
[Πρόσβαση 18 04 2014].

- [10]. Anon., 2014. *Web application framework*. [Ηλεκτρονικό]
Available at: http://en.wikipedia.org/wiki/Web_application_framework
[Πρόσβαση 18 04 2014].
- [11]. Anon., n.d. *LaTeX - A document preparation system*. [Ηλεκτρονικό]
Available at: <http://www.latex-project.org>
[Πρόσβαση 19 04 2014].
- [12]. Bidelman, E., 2011. *Using the HTML5 Filesystem API*. 1η Έκδοση επιμ.
Sebastopol: O'Reilly.
- [13]. Corderoy, R., n.d. *The Text Processor for Typesetters*. [Ηλεκτρονικό]
Available at: <http://www.troff.org>
[Πρόσβαση 19 04 2014].
- [14]. Freeman, A., 2011. *The Definitive Guide to HTML 5*. 1η Έκδοση επιμ.
s.l.:Apress.
- [15]. Fulton, S. & Fulton, J., 2011. *HTML5 Canvas*. Sebastopol: O'Reilly.
- [16]. Harris, A., 2011. Part 4: New and modified form elements. Στο: *HTML5 for dummies*. New Jersey: Willey Publishing, Inc, pp. 58-70.
- [17]. Leah, B., 2013. *HTML5 - code example of File API - drag and drop hard drive files to a webpage!*. [Ηλεκτρονικό]
Available at:
https://www.ibm.com/developerworks/mydeveloperworks/blogs/bobleah/entry/html5_code_example_of_file_api_drag_drop_hard_drive_files_to_a_webpage28?lang=en
[Πρόσβαση 19 04 2014].
- [18]. MacDonald, M., 2011. *HTML5 The Missing Manual*. Cambridge: O'Reilly.
- [19]. McLaughlin, B., 2011. *What is HTML 5*. s.l.:O'Reilly.
- [20]. Meyer, J., 2011. *HTML5 and Javascript Projects*. New York: APress.
- [21]. Sharp, R., 2011. *Methods of communication*. [Ηλεκτρονικό]
Available at: <http://html5doctor.com/methods-of-communication/>
[Πρόσβαση 19 04 2014].

- [22]. Vossen, G. & Hagemann, S., 2007. Chapter 1 - A Brief History of the Web.
Στο: *Unleashing Web 2.0*. s.l.:Morgan Kaufmann, pp. 1-68.
- [23]. Vuong, T., 2013. *MySQL Architectures and Concepts(s)*. [Ηλεκτρονικό]
Available at: <http://www.slideshare.net/TuyenVuongDinh/mysql-architectures>
[Πρόσβαση 19 04 2014].
- [24]. Πομπόρτσης, Α., 2009. *Εισαγωγή στις γλώσσες σήμανσης*. 1η Έκδοση επιμ.
Θεσσαλονίκη: Τζιόλα.