



Τμήμα Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΙΤΛΟΣ

Κατασκευή Responsive Ιστότοπου βασισμένο σε CSS
Framework

Ελπίδα Κυριτσοπούλου Α.Μ. 3032

Επιβλέπων Καθηγητής: Ιωάννης Τζήμας

Πάτρα – Μάρτιος 2021

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Πάτρα,22/03/2021.....

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. Εισηγητής: Ιωάννης Τζήμας

2.

3.

Αφιέρωση

Θα ήθελα να αφιερώσω τη συγκεκριμένη εργασία στη μαμά μου που είναι πάντα δίπλα μου και με στηρίζει στις αποφάσεις μου, όποιες και αν είναι αυτές.

Ευχαριστώ Μαμά.

Ευχαριστίες

Με την ολοκλήρωση της παρούσας εργασίας, αισθάνομαι την υποχρέωση να ευχαριστήσω τους ανθρώπους που γνώρισα και συνεργάστηκα μαζί τους και έπαιξαν σημαντικό ρόλο στην πραγματοποίησή της.

Θα ήθελα, επίσης, να ευχαριστήσω την οικογένειά μου για τη στήριξη που μου πρόσφεραν σε όλη τη διάρκεια των σπουδών μου.

Τέλος, ένα μεγάλο ευχαριστώ στον επιβλέποντα καθηγητή κ. Ιωάννη Τζήμα για την εμπιστοσύνη και τη βοήθεια που μου παρείχε.

Ελπίδα Κυριτσοπούλου

Περίληψη

Σκοπός της πτυχιακής εργασίας είναι η ενασχόληση και εξοικείωση με τις web τεχνολογίες που συνιστούν τον πυρήνα του Web Development. Το Web Development είναι ένας ραγδαία αναπτυσσόμενος κλάδος και οι τεχνολογίες (HTML και CSS) που χρησιμοποιούνται εξελίσσονται εξίσου ραγδαία.

Ευρύτερος στόχος είναι να τεθούν οι βάσεις για την περαιτέρω εκμάθηση νέων τεχνολογιών και εργαλείων του τομέα, κάτι που θα έχει ως αποτέλεσμα την ικανότητα ενασχόλησης με σύγχρονα, σύνθετα και εξαιρετικά ενδιαφέροντα projects. Τέλος, επιδιώκουμε την κατασκευή ιστοσελίδας.

Συγκεκριμένα, στη παρούσα πτυχιακή εργασία, δημιουργήθηκε μια ιστοσελίδα για το Εργαστήριο Μοριακής Βιολογίας και Ανοσολογίας του τμήματος της Φαρμακευτικής του Πανεπιστημίου Πατρών. Απευθύνεται σε φοιτητές, διδάσκοντες και σε όποιον άλλο θέλει να ενημερωθεί σχετικά με το συγκεκριμένο εργαστήριο καθώς και για τις δραστηριότητες του. Η ιστοσελίδα περιέχει πληροφορίες σχετικά με το έργο του εργαστηρίου και περιλαμβάνει όλους του εκπαιδευτικούς που το απαρτίζουν. Οι επισκέπτες της σελίδας θα είναι σε θέση να μαθαίνουν τα νέα του εργαστηρίου και να ενημερώνονται για τα τρέχοντα ζητήματά του.

Για την κατασκευή της ιστοσελίδας έγινε εκτεταμένη χρήση των HTML, CSS και του Bootstrap framework , με στόχο την ανάπτυξη ενός responsive website. Τέλος, γίνεται λεπτομερής αναφορά στον τρόπο και τις τεχνολογίες υλοποίησης της ιστοσελίδας.

Abstract

The aim of this thesis project is the use and familiarity of the web services which are the main part of web development. Web development is an increasingly popular like of work and the functionalities, technologies, and programming languages (such as HTML and CSS) are constantly under development.

A wider aim of the specific thesis is to set the foundation of the continuous learning and development of such functionalities and tools of web development, which can have, as a result, the ability to use new, complex and extremely interesting projects. Lastly, the main goal of the thesis is to build a fully functional website.

To be more specific, during the current project, a new website for the Laboratory of Molecule, Biology and Immunology of the Department of Pharmacy of the University of Patras was developed. The website addresses students, teachers and anyone who is interested and wants to stay informed regarding the laboratory as well as their activities. The website includes information regarding the mission as well as details and information of the professors that collaborate with the laboratory. The visitors of the website will be able to see the news of the laboratory and get updates when it comes to current activities.

For the development of the website there was an extensive use of the programming languages of HTML, CSS and Bootstrap Framework, in order to build a responsive website. Last but not least, there is a description included which focuses on how and which functionalities are used in order for the website to be build.

Περιεχόμενα

| | |
|--|----|
| Ευχαριστίες..... | 3 |
| Περίληψη..... | 4 |
| Abstract..... | 5 |
| Περιεχόμενα..... | 6 |
| Εικόνες και Πίνακες..... | 8 |
| ΕΙΣΑΓΩΓΗ..... | 9 |
| ΚΕΦ. 1 : ΙΣΤΟΣΕΛΙΔΕΣ..... | 10 |
| 1.1. Ιστοσελίδα - Εισαγωγή..... | 10 |
| 1.2. Στατικές Ιστοσελίδες (Static Webpage)..... | 12 |
| 1.3. Δυναμικές Ιστοσελίδες (Dynamic Webpage)..... | 14 |
| 1.4. Responsive Design[9]..... | 16 |
| ΚΕΦ 2. ΓΛΩΣΣΕΣ ΣΗΜΑΝΣΗΣ..... | 18 |
| Εισαγωγή..... | 18 |
| 2.1. HTML (HyperText Markup Language)..... | 19 |
| 2.1.1. Ιστορία..... | 19 |
| 2.1.2. Ιστορικό εκδόσεων του προτύπου..... | 20 |
| 2.1.3. Τι είναι η HTML..... | 21 |
| 2.1.4. Δομή ενός αρχείου HTML..... | 22 |
| 2.1.5. Βασικές ετικέτες..... | 23 |
| 2.1.6. Δείγμα Κώδικα HTML..... | 25 |
| 2.2. HTML 5 (HyperText Markup Language 5)..... | 26 |
| 2.2.1. Νέες ετικέτες..... | 28 |
| 2.2.2. Μειονεκτήματα στην HTML5..... | 29 |
| 2.3. Βασικές διαφορές HTML και HTML 5..... | 31 |
| 2.4. CSS (Cascading Style Sheets)[24]..... | 32 |
| Εισαγωγή..... | 32 |
| 2.4.1. Η Ιστορία της CSS..... | 34 |
| 2.4.2. Σύνταξη..... | 34 |
| 2.4.2.1. Διαφορά μεταξύ Ιδιοτήτων και Κλάσεων..... | 36 |
| 2.4.3. Επιλογείς..... | 37 |
| 2.5. CSS 3 (Cascading Style Sheets 3)..... | 44 |
| 2.5.1. Modules..... | 45 |
| 2.5.2. Media Queries..... | 46 |
| 2.5.3. Transitions..... | 47 |

| | |
|--|----|
| 2.5.4. Νέοι επιλογείς..... | 48 |
| 2.5.5. Νέες Ιδιότητες..... | 50 |
| 2.6. Σύγκριση CSS και CSS3[29] | 51 |
| ΚΕΦ. 3. BOOTSTRAP | 56 |
| Εισαγωγή..... | 56 |
| 3.1. Χαρακτηριστικά - Εγκατάσταση | 56 |
| 3.2. Δομή και Λειτουργία | 57 |
| 3.2.1. Containers | 58 |
| 3.2.2. Πλέγμα (Grid) | 58 |
| 3.2.3. Επιλογές Πλέγματος (Grid Options)..... | 59 |
| 3.2.4. Εμφώλευση Στηλών | 60 |
| 3.2.5. Media Queries | 60 |
| 3.3. Ανάλυση παραμέτρων | 61 |
| ΚΕΦ. 4. CSS Frameworks | 63 |
| Εισαγωγή..... | 63 |
| 4.2 Παραδείγματα CSS Frameworks..... | 63 |
| 4.3 Οι λόγοι που χρησιμοποίησα το Bootstrap..... | 66 |
| ΚΕΦ. 5. Υλοποίηση – Κώδικας | 67 |
| 5.1. Αρχική Σελίδα – index.html | 68 |
| 5.2. Φάκελος Αρχείων CSS | 74 |

Εικόνες και Πίνακες

| | |
|---|----|
| Εικόνα 1 Πρώτη HTML σελίδα..... | 10 |
| Εικόνα 2 Διεύθυνση ιστοσελίδας..... | 11 |
| Εικόνα 3. Στατική ιστοσελίδα..... | 12 |
| Εικόνα 4 Δυναμική ιστοσελίδα..... | 14 |
| Εικόνα 5 Responsive Design | 17 |
| Εικόνα 6 HTML..... | 19 |
| Εικόνα 7 official logo HTML 5 | 26 |
| Εικόνα 8 CSS - Cascading Style Sheets..... | 32 |
| Εικόνα 9 CSS 3 logo | 44 |
| Εικόνα 10 Bootstrap logo | 56 |
| Εικόνα 11 Bootstrap CDN..... | 57 |
| Εικόνα 12 Δομή αρχείων Bootstrap | 57 |
| Εικόνα 13 Επιλογές Πλέγματος στο Bootstrap..... | 59 |
| Εικόνα 14 Παράδειγμα εμφώνλευσης στηλών..... | 60 |
| Εικόνα 15 Bootstrap media queries | 60 |
| | |
| Πίνακας 1 Πίνακας σύγκρισης CSS και CSS3 | 55 |

ΕΙΣΑΓΩΓΗ

Το Web development αφορά στη διαδικασία που περιλαμβάνει την ανάπτυξη ενός website. Το web development μπορεί να ποικίλλει από την ανάπτυξη ενός απλού στατικού site απλού κειμένου, μέχρι πολύπλοκων διαδικτυακών εφαρμογών ηλεκτρονικού εμπορίου και εφαρμογών κοινωνικής δικτύωσης. Υπάρχουν τρία είδη ειδίκευσης στο web development. Ο front end developer, ο back end developer, και ο full stack developer. Οι πρώτοι, είναι υπεύθυνοι για την εμφάνιση και το οπτικό κομμάτι που τρέχει στον browser, ενώ οι back end developers ασχολούνται με τους servers. Στην παρούσα διπλωματική, δίνεται έμφαση στο front end κομμάτι, και οι τεχνολογίες που αποτελούν τον πυρήνα του είναι η τριάδα, HTML, CSS συνεπικουρούμενος με frameworks και libraries, όπως το bootstrap. Αυτά ακριβώς θα γίνει προσπάθεια να αναλυθούν και να χρησιμοποιηθούν στα επόμενα κεφάλαια.

Στο Κεφάλαιο 1 γίνεται μία εισαγωγή στο τι είναι μία ιστοσελίδα ο διαχωρισμός σε στατική και δυναμική ιστοσελίδα αλλά και μία εισαγωγή στο Responsive Design.

Στο Κεφάλαιο 2 γίνεται μια εισαγωγή στις γλώσσες σήμανσης HTML, HTML5, CSS & CSS3, στην ιστορία τους, την εξέλιξή τους, με έμφαση στα δομικά χαρακτηριστικά τους και τις δυνατότητές τους. Στη συνέχεια αναφέρονται οι διαφορές της HTML με την HTML5 αλλά και οι διαφορές της CSS με τη CSS3.

Στο Κεφάλαιο 3 παρουσιάζεται το Bootstrap και ο ρόλος του στο styling, ως framework. Ο τρόπος εγκατάστασης και κάποια βασικά χαρακτηριστικά του συγκεκριμένου framework.

Στο Κεφάλαιο 4 παρουσιάζονται κάποιες άλλες επιλογές CSS Frameworks, τα θετικά και τα αρνητικά σε σχέση με το Bootstrap. Και ο λόγος που τελικά επιλέχθηκε το συγκεκριμένο Framework.

Τέλος, στο Κεφάλαιο 5 παρουσιάζεται αναλυτικά η υλοποίηση σε κώδικα και η χρήση των τεχνολογιών που παρουσιάστηκαν στα προηγούμενα κεφάλαια.

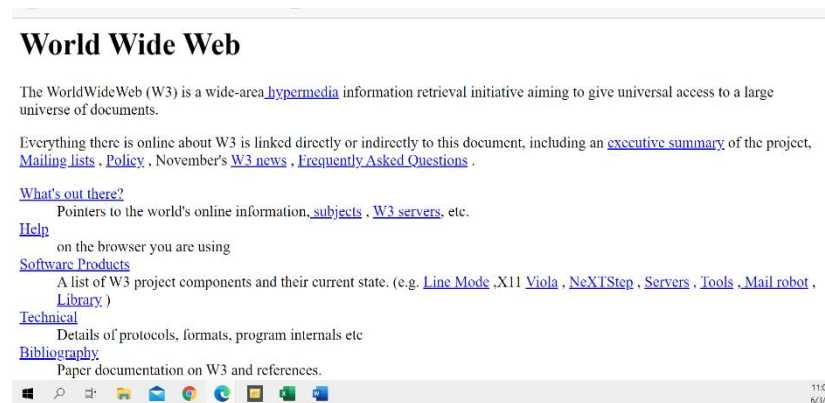
ΚΕΦ. 1 : ΙΣΤΟΣΕΛΙΔΕΣ

1.1. Ιστοσελίδα - Εισαγωγή

Ιστοσελίδα (web page) είναι ένα είδος εμπλουτισμένου εγγράφου του παγκόσμιου ιστού που περιλαμβάνει πληροφορίες που έχουν τη μορφή κειμένου, εικόνας, ήχου, βίντεο και υπερκειμένου.

Η πρώτη ιστοσελίδα [1] δημιουργήθηκε από τον δημιουργό του Παγκόσμιου Ιστού (World Wide Web) Tim Berners-Lee, στις 6 Αυγούστου του 1991 στο CERN. Η οποία «ανέβηκε» ξανά online για να γιορτάσουν τα 20 χρόνια του Παγκόσμιου Ιστού. Ήταν η πρώτη στατική ιστοσελίδα με απλό κείμενο και υπερσυνδέσμους που οδηγούσαν στις άλλες σελίδες του ιστοτόπου.

Μπορείτε να επισκεφτείτε την πρώτη ιστοσελίδα πηγαίνοντας στον παρακάτω σύνδεσμο: <http://info.cern.ch/hypertext/WWW/TheProject.html>. [2]



Εικόνα 1 Πρώτη HTML σελίδα

Η ιστοσελίδα συνήθως αναφέρεται σε αυτό που είναι ορατό, αλλά ο όρος επίσης μπορεί να αναφέρεται σε ένα έγγραφο υπολογιστή, το οποίο συνήθως είναι γραμμένο στην γλώσσα σήμανσης HTML ή σε οποιαδήποτε άλλη γλώσσα σήμανσης κειμένου. Για την εμφάνιση των ιστοσελίδων, οι φυλλομετρητές συντονίζουν τα διάφορα αρχεία που περιέχουν τα στοιχεία πόρων του ιστού, για την συγκεκριμένη ιστοσελίδα, όπως είναι τα φύλλα στυλ (stylesheets), scripts και εικόνες. Η μορφή μίας τυπικής ιστοσελίδας παρέχει υπερκείμενο, το οποίο περιλαμβάνει μία μπάρα πλοήγησης ή μία πλαϊνή μπάρα μενού που συνδέεται με άλλες ιστοσελίδες μέσω υπερσυνδέσμων, που συχνά αναφέρονται ως σύνδεσμοι (links).

Με την βοήθεια του πρωτόκολλο HTTP (Hypertext Transfer Protocol) επιτυγχάνεται η πρόσβαση και η μεταφορά μίας ιστοσελίδας. Οι ιστοσελίδες αλληλοσυνδέονται και μπορεί ο χρήστης να μεταβεί από τη μία στην άλλη κάνοντας «κλικ», επιλέγοντας δηλαδή συνδέσμους που υπάρχουν στο κείμενο ή στις φωτογραφίες της ιστοσελίδας.

Οι σύνδεσμοι προς άλλες σελίδες εμφανίζονται συνήθως υπογραμμισμένοι και με μπλε χρώμα για να είναι γρήγορα ξεκάθαρο στον επισκέπτη ότι πρόκειται για σύνδεσμο προς άλλη ιστοσελίδα, χωρίς όμως πάντα να είναι αυτό απαραίτητο.

Η δημιουργία πολλών ιστοσελίδων μαζί, με κοινό θέμα και ένα μενού με υπερσυνδέσμους που οδηγούν στις υπόλοιπες ιστοσελίδες, οι οποίες όλες φιλοξενούνται στο ίδιο Domain Name και Server του Παγκόσμιου Ιστού, ονομάζεται Ιστότοπος ή Ιστοχώρος (Web Site). Μπορούμε να επισκεφτούμε έναν Ιστότοπο χρησιμοποιώντας την διεύθυνση της ιστοσελίδας, η οποία βασίζεται στην υπηρεσία WWW και χρησιμοποιεί το πρωτόκολλο http. Όταν θέλουμε λοιπόν να επισκεφτούμε μία ιστοσελίδα αρκεί να θυμόμαστε το domain name της και να χρησιμοποιήσουμε το url: `http://WWW.domainname.TLD` [3] , για παράδειγμα αν θέλουμε να επισκεφτούμε την ιστοσελίδα της Google μπαίνουμε στο `http://www.google.com`.

Κάθε ιστοσελίδα που περιέχεται σε δικτυακό τόπο, έχει τη δική της διεύθυνση στον Παγκόσμιο Ιστό, και αν θέλουμε να «επισκεφτούμε» μία ιστοσελίδα, πρέπει να ξέρουμε τη διεύθυνσή της. Η διεύθυνση αυτή καλείται URL (Uniform Resource Locator)[4] – Ενιαίος Προσδιοριστής Πόρου ή απλούστερα διεύθυνση ιστοσελίδας. Μία διεύθυνση ιστοσελίδας είναι μοναδική και έχει συνήθως την εξής μορφή:



Εικόνα 2 Διεύθυνση Ιστοσελίδας

Η κατασκευή ιστοσελίδων είναι κάτι που μπορεί να γίνει πολύ εύκολα με προγράμματα που κυκλοφορούν ελεύθερα, αλλά υπάρχουν και αυτοματοποιημένοι μηχανισμοί

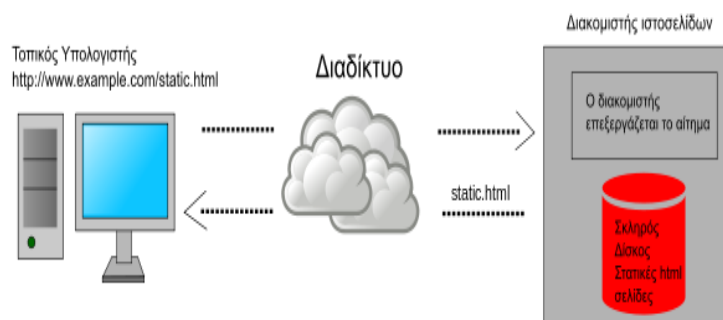
κατασκευής ιστοσελίδων που επιτρέπουν σε απλούς χρήστες να δημιουργήσουν εύκολα και γρήγορα προσωπικές ή και εμπορικές ιστοσελίδες. Από την άλλη μεριά υπάρχουν και πολλές εταιρίες, που εξειδικεύονται στη δημιουργία ελκυστικών και λειτουργικών ιστοσελίδων που έχουν σαν στόχο να οδηγήσουν τους επισκέπτες στην αγορά κάποιου προϊόντος, στην επικοινωνία με τον ιδιοκτήτη του ιστοτόπου ή απλά στο ανέβασμα του εταιρικού προφίλ μιας επιχείρησης. Οι ιστοσελίδες χωρίζονται σε κατηγορίες. Οι δύο βασικότερες και μεγαλύτερες είναι οι:

1. Στατικές ιστοσελίδες (static webpage)
2. Δυναμικές ιστοσελίδες (dynamic webpage)[5]

Οι διαφορές τους δεν είναι στην εμφάνιση της ιστοσελίδας αλλά στον τρόπο λειτουργίας και τις δυνατότητές τους, όπως θα φανεί στη συνέχεια.

1.2. Στατικές Ιστοσελίδες (Static Webpage)

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



Εικόνα 3. Στατική Ιστοσελίδα

Οι στατικές ιστοσελίδες χαρακτηρίζονται από την μονιμότητα του περιεχομένου τους και της διάταξής τους (layout), τα οποία μπορούν να αλλάξουν/να τροποποιηθούν μόνο με αίτημα για αναβάθμιση (update) από τον προγραμματιστή/διαχειριστή της σελίδας αυτής.

Οι ιστοσελίδες αυτές είναι απλά HTML έγγραφα που περιέχουν απλό κείμενο, εικόνες και συνδέσμους. Λόγω του ότι το περιεχόμενό τους είναι σταθερό, είναι κατάλληλες για ιστοσελίδες που δεν υπάρχει η ανάγκη για συχνές αλλαγές ή τροποποίηση του περιεχομένου τους. Η τροποποίηση των δεδομένων (κείμενα, φωτογραφίες κλπ) σε μία στατική ιστοσελίδα γίνεται μόνο με την παρέμβαση του διαχειριστή χειροκίνητα, με την βοήθεια κάποιου ειδικού προγράμματος κατασκευής ιστοσελίδων

Με άλλα λόγια, κάθε στατικό site αποτελεί ένα σύνολο ιστοσελίδων με σταθερό περιεχόμενο, οι οποίες ενδέχεται να περιέχουν εκτενείς πληροφορίες (όπως στοιχεία επικοινωνίας, τιμές, προϊόντα/υπηρεσίες, γραφικά), αλλά η τροποποίησή τους είναι δύσκολη και απαιτεί χρόνο και χρήμα.

Πλεονεκτήματα στατικής ιστοσελίδας

- χαμηλό κόστος κατασκευής και συντήρησης, αφού μια στατική σελίδα, δεν απαιτεί μεγάλο χώρο σε φιλοξενία από webserver.
- Δεν χρειάζονται προγραμματιστικές δεξιότητες για να δημιουργήσει κάποιος μια στατική σελίδα.
- Η σελίδα μπορεί να βρίσκεται με εύκολο τρόπο κλωνοποιημένη σε περισσότερους από έναν εξυπηρετητές.
- Δεν χρειάζεται ειδικό λογισμικό στον εξυπηρετητή ιστοσελίδων για την δημοσίευση στατικών σελίδων.
- Αναπτύσσεται/σχεδιάζεται γρηγορότερα από ότι μια δυναμική.
- Οι σελίδες φορτώνουν πιο γρήγορα από ότι οι δυναμικές.
- Είναι Seo friendly.[6] Το SEO (Search Engine Optimization) που στα Ελληνικά σημαίνει Βελτιστοποίηση Σελίδας για τις μηχανές αναζήτησης, είναι πιο φιλικές για τις μηχανές αναζήτησης και αυτό συμβαίνει γιατί στις στατικές υπάρχει η δυνατότητα εισαγωγής στον κώδικα HTML, των meta-tags τα οποία αναγνωρίζονται από τις μηχανές αναζήτησης, και τα οποία βοηθούν τις μηχανές αναζήτησης να κατατάσσουν την στατική σελίδα πιο πάνω από μία δυναμική

που δεν έχει ακριβώς αυτήν την δυνατότητα. Αυτό είναι πολύ σημαντικό διότι αυξάνει την επισκεψιμότητα ενός ιστοχώρου, άρα και το κέρδος της.

- Μεγαλύτερη Ασφάλεια Δεδομένων.

Μειονεκτήματα στατικής ιστοσελίδας

- Χρειάζονται προγραμματιστικές γνώσεις για την ανανέωση του περιεχομένου
- Δεν είναι εύκολη η διαδραστικότητα με τον χρήστη.
- Η διαχείριση μεγάλου αριθμού στατικών ιστοσελίδων δεν είναι εύκολη χωρίς αυτόματα εργαλεία.

1.3.Δυναμικές Ιστοσελίδες (Dynamic Webpage)

Δυναμική ιστοσελίδα (dynamic web page) είναι μια ιστοσελίδα η οποία δημιουργείται δυναμικά την στιγμή της πρόσβασης σε αυτή ή την στιγμή που ο χρήστης αλληλεπιδρά με τον εξυπηρετητή ιστοσελίδων.



Εικόνα 4 Δυναμική Ιστοσελίδα

Οι δυναμικές ιστοσελίδες θεωρούνται δομικό στοιχείο της νέας γενιάς του παγκόσμιου ιστού (Web 2.0) όπου η πληροφορία διαμοιράζεται σε πολλαπλές ιστοσελίδες. Οι δυναμικές ιστοσελίδες, σε αντίθεση με τις στατικές, προσαρμόζουν το περιεχόμενό τους και την εμφάνισή τους σύμφωνα με την καταχώρηση/αλληλεπίδραση ή τις αλλαγές του τελικού χρήστη στο περιβάλλον προγραμματισμού (χρήστης, ώρα, τροποποιήσεις στη βάση δεδομένων κτλ.).

Το περιεχόμενο μπορεί να αλλάζει στον υπολογιστή του τελικού χρήστη με τη χρήση των γλωσσών προγραμματισμού που εκτελούνται στον υπολογιστή του χρήστη (JavaScript, VBScript). Ο, δε, διαχειριστής μπορεί να κάνει οποιαδήποτε προσθήκη,

αλλαγή, αφαίρεση και διόρθωση χρειάζεται, αφού οι δυναμικές ιστοσελίδες είναι εφαρμογές – προγράμματα και όχι απλά HTML έγγραφα όπως οι στατικές.

Για να διαβαστούν τα περιεχόμενα μία δυναμικής ιστοσελίδας χρειάζεται κάτι παραπάνω από έναν φυλλομετρητή όπως είναι στις στατικές.

Για να μεταφραστούν και να εμφανιστούν στις οθόνες μας μέσω των φυλλομετρητών, τα περιεχόμενα τους μεταφράζονται στους Servers και από εκεί αποστέλλονται μέσω του διακομιστή Apache [7]. Τα περιεχόμενα είναι γραμμένα σε γλώσσες προγραμματισμού που εκτελούνται στον εξυπηρετητή (Perl, PHP, ASP, JSP, ColdFusion, .NET κτλ).

Πίσω από δυναμικά websites κρύβονται πάντα βάσεις δεδομένων (databases) όπου εκεί αποθηκεύονται δεδομένα και πληροφορίες του εν λόγω website. Και με εντολή του χρήστη, το website επικοινωνεί με την database, από όπου θα αντλήσει το περιεχόμενο που απαιτήσε να δει ο χρήστης. Εξαιτίας των βάσεων δεδομένων που χρησιμοποιούνται, είναι πολύ εύκολο για κάποιον που δεν γνωρίζει από γλώσσες προγραμματισμού να μπορέσει να προσθέσει ή να αφαιρέσει κάποια περιεχόμενα από την ιστοσελίδα του, γιατί το μόνο που χρειάζεται είναι να έχει τη γνώση να διαχειρίζεται το περιεχόμενα μίας βάσης δεδομένων. Όλα τα υπόλοιπα γίνονται αυτοματοποιημένα από το πρόγραμμα.

Για την δημιουργία ενός δυναμικού ιστοτόπου χρειάζονται πολλές γραμμές κώδικα προγραμματισμού κάτι που το κάνει δύσκολο στην κατασκευή, όμως είναι εύκολο αργότερα στην διαχείριση και στην ανανέωση των περιεχομένων του. Εδώ έχουμε και το μεγάλο τους πλεονέκτημα που είναι η χρήση των CMS[8]. Για να μπορεί ένας απλός χρήστης να διαχειρίζεται τα περιεχόμενα μίας δυναμικής ιστοσελίδας εύκολα και χωρίς να χρειάζονται γνώσεις προγραμματισμού υπάρχουν τα Content Management System ή αλλιώς CMS τα οποία είναι μηχανισμοί διαχείρισης περιεχομένων. Τα CMS έχουν μία περιοχή διαχείρισης της σελίδας, όπου ο διαχειριστής μπορεί να διαχειρίζεται όλα τα περιεχόμενα του ιστοτόπου του.

Πλεονεκτήματα δυναμικής ιστοσελίδας:

- Δυναμική παρουσίαση στο Διαδίκτυο
- Δεν απαιτείται η βοήθεια ειδικού προγραμματιστή για τη διαχείριση της ιστοσελίδας
- Εύκολη διαχείριση χρηστών, διαχειριστών και μελών μίας ιστοσελίδας

- Παρέχουν πολλές δυνατότητες απο ότι οι στατικές σελίδες.
- Πολύ πιο εύκολη η ενημέρωσή της.
- Συχνή ανανέωση περιεχομένου και καθόλου περιορισμός στις σελίδες που μπορεί να έχει κάποιος
- Πολύ φιλικές ως προς τις μηχανές αναζήτηση

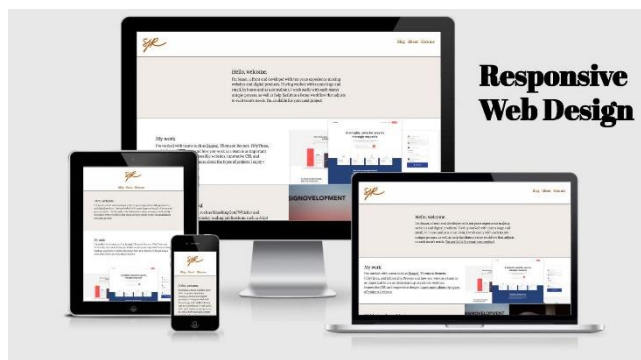
Μειονεκτήματα δυναμικής ιστοσελίδας:

- Υψηλό κόστος Κατασκευής και συντήρησης
- Μικρότερη ασφάλεια δεδομένων (online Βάση Δεδομένων)
- Απαιτείται μεγάλος χώρος φιλοξενίας σε κάποιον Web Server
- Αναπτύσσεται και σχεδιάζεται πιο αργά από μία στατική ιστοσελίδα
- Η σελίδα «φορτώνεται» πιο αργά λόγω του περιεχομένου της

Οι δυναμικές ιστοσελίδες είναι χρήσιμες για επιχειρήσεις που θέλουν να κατασκευάσουν ένα ηλεκτρονικό κατάστημα, όπου χρειάζεται η συχνή ανανέωση των περιεχομένων της. Όπως επίσης και για ξενοδοχεία που θέλουν οι πελάτες να μπορούν να κάνουν ηλεκτρονικές κρατήσεις. Και επίσης, όλα οι ειδησεογραφικοί ιστότοποι όπου χρειάζονται συχνά να ανανεώνουν το περιεχόμενό τους.

1.4. Responsive Design[9]

Τα τελευταία χρόνια η χρήση των tablets και των κινητών συσκευών για την πλοήγηση στο διαδίκτυο αυξήθηκε αρκετά σε σχέση με παλαιότερα που χρησιμοποιούσαμε αποκλειστικά σταθερό ηλεκτρονικό υπολογιστή. Κατά συνέπεια, δημιουργήθηκαν προβλήματα στην εμφάνιση των ιστοσελίδων που οφείλονται σε διάφορες ασυμβατότητες ανάμεσα σε διαφορετικές φορητές συσκευές, καθώς επίσης και στην πληθώρα διαφορετικών διαστάσεων στις οθόνες τους. Για να ξεπεραστούν τα προβλήματα αυτά χρησιμοποιούμε την τεχνική του Responsive Web Design για την κατασκευή ιστοσελίδων με στόχο να προσφέρουμε στον επισκέπτη την καλύτερη δυνατή και πιο ευχάριστη πλοήγηση αλλά και την εκμετάλλευση όλων των δυνατοτήτων που προσφέρει το κάθε μέσο που χρησιμοποιεί.



Εικόνα 5 Responsive Design

Το Responsive Web Design είναι ένας σχετικά καινούριος όρος που πήρε την ονομασία του από την λέξη "Respond" . Στη γλώσσα μας η λέξη μεταφράζεται ως απάντηση – ανταπόκριση. Μπορούμε, δηλαδή, να πούμε ότι χρησιμοποιούμε αυτήν την τεχνική ώστε η ιστοσελίδα να ανταποκρίνεται βάση μεταβλητών τις οποίες ανιχνεύει στη συσκευή του χρήστη. Ως εκ τούτου, σημαίνει ότι μια ιστοσελίδα ανταποκρίνεται & διαμορφώνεται ανάλογα με την ανάλυση της οθόνης στην οποία εμφανίζεται η ιστοσελίδα.

Στο web, θα ορίζουμε το responsive web design ως τη διαδικασία σχεδιασμού και κατασκευής ιστοσελίδων οι οποίες ανιχνεύουν διάφορες μεταβλητές από το εξωτερικό και εσωτερικό περιβάλλον και ανταποκρίνονται ανάλογα προς τον επισκέπτη. Στόχος της καινοτομίας αυτής είναι η δημιουργία μίας «έξυπνης» ιστοσελίδας η όποια θα προσαρμόζει το μέγεθος και τα βασικά χαρακτηριστικά της (κείμενο , εικόνες, μενού) με βάση τις διαστάσεις της οθόνης της συσκευής του κάθε χρήστη.

Αυτό σημαίνει ότι αν κάποιος ανοίξει μια ιστοσελίδα η οποία είναι Responsive από ένα smartphone, η ιστοσελίδα θα διαμορφωθεί με τέτοιο τρόπο ώστε να μην χρειάζεται να κάνει μεγέθυνση ή να κάνει πλάγιο σκρολ για να διαβάσει με ευκολία το περιεχόμενο της εκάστοτε ιστοσελίδας αφού η ιστοσελίδα ανιχνεύει τη συσκευή του επισκέπτη και τις διαστάσεις της οθόνης.

Για παράδειγμα, όταν ο επισκέπτης χρησιμοποιεί οθόνη με σχετικά μεγάλες διαστάσεις (π.χ. πάνω από 1000px), η προβολή της σελίδας γίνεται υπό κανονικές συνθήκες εμφανίζοντας όσον το δυνατόν περισσότερα αντικείμενα. Σε οθόνες με μικρότερες διαστάσεις, τα αντικείμενα τοποθετούνται συνήθως το ένα κάτω από το άλλο και μερικές φορές αποκρύπτονται εντελώς.

Πλεονεκτήματα:

- Το πιο βασικό πλεονέκτημα είναι ότι έχουμε μια ιστοσελίδα η οποία συμπεριφέρεται σαν 2 σε 1. Μπορούμε να επισκεφθούμε μια ιστοσελίδα χωρίς πρόβλημα είτε από έναν προσωπικό υπολογιστή είτε από μια κινητή συσκευή έχοντας την καλύτερη δυνατή εμπειρία πλοήγησης.
- Analytics: Μπορούμε να έχουμε μια πλήρη αναφορά των επισκέψεων της ιστοσελίδας μας για διαφορετικού είδους συσκευές.
- Σύνδεσμοι: Κοινοί σύνδεσμοι (links) που έχουμε στην ιστοσελίδα ανεξάρτητα από το είδος της συσκευής.
- SEO: Ένα url συμπεριλαμβάνει όλα τα links που θέλουμε να έχουμε.
- Μεγαλύτερη ευκολία στη συντήρηση και μείωση των σφαλμάτων. Είναι καλύτερα να συντηρείς μία ιστοσελίδα, παρά τρεις (μία για προσωπικό υπολογιστή, μία για tablet και μία για κινητό τηλέφωνο).
- Μείωση του χρόνου σχεδίασης της δομής της ιστοσελίδας για όσον αφορά διαφορετικές συσκευές.
- Διευκόλυνση του επισκέπτη της ιστοσελίδας: Καθώς ο επισκέπτης της ιστοσελίδας δεν χρειάζεται να κάνει συνεχώς μεγέθυνση για να διαβάζει το περιεχόμενο της ιστοσελίδας, του προσφέρεται ένα πιο ευχάριστο, καλαίσθητο και γρήγορο περιβάλλον για να την εξερευνήσει.

ΚΕΦ 2. ΓΛΩΣΣΕΣ ΣΗΜΑΝΣΗΣ

Εισαγωγή

Οι γλώσσες σήμανσης είναι μία τεχνική περίπλοκη με την οποία λέμε τι πρέπει να κάνει ένας υπολογιστής. Δίνουμε, δηλαδή, στον υπολογιστή οδηγίες με μικρά βήματα ώστε αυτός να τα κάνει με την σειρά.

Αποκτώντας την πρόσβαση σε μία ιστοσελίδα με ένα πρόγραμμα περιήγησης, η γραμμή σελίδας HTML θα αναλυθεί ή θα διαβαστεί από τον διακομιστή του ιστού κατά γραμμή και θα εκτελέσει τον κώδικα όταν έρθει σε γλώσσα προγραμματισμού.

Μία στατική και απλή σελίδα HTML μετατρέπεται σε δυναμική μέσα από τον προγραμματισμό. Όλοι μπορούν να αλληλεπιδρούν με τον ιστότοπο και να κάνουν χρήση της ιστοσελίδας σε όποιον υπολογιστή έχει πρόσβαση στο διαδίκτυο. Επιτρέπει, επίσης, την επεξεργασία μέσω του διαχειριστικού πχ. Βιβλίο επισκεπτών, φόρουμ ή και την υποβολή φόρμας.

Ολόκληρος ο προγραμματισμός ιστού επιτυγχάνεται με τις γλώσσες σήμανσης που μπορούν να περιλαμβάνουν τεχνολογίες όπως CSS και HTML.

Ωστόσο, ο περισσότερος προγραμματισμός ιστοσελίδων πραγματοποιείται χρησιμοποιώντας γλώσσες προγραμματισμού ιστού από την πλευρά του διακομιστή. Αυτός ο κώδικας εκτελείται στον διακομιστή και στη συνέχεια δίνει στατικές πληροφορίες πίσω στο πρόγραμμα περιήγησης ιστού.

2.1. HTML (HyperText Markup Language)



Εικόνα 6 HTML

Η HTML (HyperText Markup Language, Γλώσσα Σήμανσης του Υπερκειμένου) αποτελεί υποσύνολο της γλώσσας SGML (Standard Generalized Markup Language) που επινοήθηκε από την IBM προκειμένου να λυθεί το πρόβλημα της μη τυποποιημένης εμφάνισης κειμένων στα διάφορα υπολογιστικά συστήματα.

2.1.1. Ιστορία

Δημιουργός της HTML γλώσσας είναι ο Tim Berners-Lee, ο ίδιος που δημιούργησε και τον Παγκόσμιο Ιστό (WWW). Το 1980, όταν ο Lee εργαζόταν ακόμα στο CERN, για τις ανάγκες των ερευνητών να χρησιμοποιούν και να διαμοιράζονται έγγραφα, επινόησε το σύστημα ENQUIRE[10]. Έτσι κατασκεύασε ένα πρωτότυπό του.

Αργότερα, το 1989, πρότεινε ένα νέο σύστημα το οποίο βασίστηκε στο διαδίκτυο και πλέον θα χρησιμοποιούσε υπερκείμενο, φτιάχνοντας έτσι την προδιαγραφή της HTML και αργότερα στα τέλη του 1990, έγραψε τον φυλλομετρητή και το λογισμικό του server. Η δημιουργία της HTML βασίστηκε στην SGML.

Η HTML ορίστηκε και επίσημα από τον οργανισμό IETF [11] ως μια υλοποίηση της SGML. Το 1993, με μία δημοσίευση της πρώτης πρότασης για μια προδιαγραφή της HTML, η οποία προδιαγραφή περιελάμβανε και έναν Ορισμό τύπου εγγράφου (DTD) της SGML, ορίζει την σωστή γραμματική του εγγράφου. Ο Ορισμός τύπου εγγράφου (DTD) είναι στην ουσία ένα σύνολο δηλώσεων σήμανσης που ορίζουν έναν τύπο εγγράφου για μία γλώσσα σήμανσης που ανήκει στην οικογένεια της SGML, ορίζει δηλαδή τη δομή του εγγράφου με μια λίστα αποδεκτών στοιχείων και χαρακτηριστικών.

2.1.2. Ιστορικό εκδόσεων του προτύπου

Το χρονικό εκδόσεων HTML:

24 Νοεμβρίου 1995

Η HTML 2.0 δημοσιεύτηκε ως IETF RFC 1866.[12]

Τον Ιούνιο του 2000, τα ιδιαίτερα χαρακτηριστικά της χαρακτηρίστηκαν outdated από το RFC 2854.

Ιανουάριος 1997

Η HTML 3.2[13] δημοσιεύτηκε ως Σύσταση από τον οργανισμό W3C. Επρόκειτο για την πρώτη έκδοση που αναπτύχθηκε και καθορίστηκε ως πρότυπο αποκλειστικά από το W3C, ιδιαίτερα αφότου το IETF τερμάτισε το Working Group για την HTML τον Σεπτέμβριο του 1996.[14]

Δημοσιεύτηκε η HTML 4.0 [15] από το W3C, ως Σύσταση. Παρείχε τρεις εκδοχές:

- την Strict, στην οποία δεν επιτρέπεται η χρήση outdated elements,
- την Transitional, στην οποία επιτρέπονται τα outdated elements,

- την Frameset, στην οποία επιτρέπονται μόνο frame elements.

Η HTML 4 είναι μια εφαρμογή του SGML, σύμφωνα με το πρότυπο ISO 8879 - SGML.[16]

Απρίλιος 1998

Εκδίδεται και πάλι η HTML 4.0 [17], με μικροαλλαγές, δίχως να αλλάξει ο αριθμός της version.

Δεκέμβριος 1999

Δημοσιεύεται ως Σύσταση του W3C η HTML 4.01[18]. Παρείχε τις ίδιες 3 version με την HTML 4.0. Τα πλέον πρόσφατα λάθη και παραλείψεις (errata) δημοσιεύτηκαν στις 12 Μαΐου του 2001.[19]

Μάιος 2000

Με το ISO/IEC 15445:2000[20] [21]καθορίζεται το πρότυπο της ISO HTML, και βασίστηκε στην HTML 4.01 Strict, ως international πρότυπο ISO/IEC.

Από τα μέσα του 2008, η HTML 4.01 και το ISO/IEC 15445:2000 αποτελούν τις πλέον πρόσφατες version της HTML. Παράλληλα αναπτύχθηκε η XHTML, που εδράζεται στην XML, με την οποία ασχολήθηκε η Working Group του W3C για την HTML από την αρχή μέχρι τα μέσα της δεκαετίας του 2000.

2.1.3. Τι είναι η HTML

Η HTML είναι η πρώτη και η πιο διαδεδομένη γλώσσα περιγραφής ιστοσελίδας την οποία αναγνωρίζουν πλέον όλοι οι Browsers και βλέπουμε το τελικό αποτέλεσμα των ιστοσελίδων που παράγονται από αυτήν.

Αποτελεί μία απλή γλώσσα περιγραφής που χρησιμοποιείται για τη δημιουργία εγγράφων υπερκειμένου (hypertext), τα οποία είναι δυνατόν να μεταφέρονται σε διαφορετικές πλατφόρμες λειτουργικών συστημάτων.

Η HTML δεν είναι γλώσσα προγραμματισμού αλλά είναι η κύρια markup language για websites, και ως προς τα στοιχεία της, πρόκειται για τα βασικά δομικά στοιχεία των ιστοσελίδων. Έχει, δε, τυποποιηθεί από τον παγκόσμιο οργανισμό τυποποίησης τεχνολογιών του Web, τον World Wide Web Consortium (W3C).

Τα αντικείμενα ενός εγγράφου HTML αναπαριστούνται και αλληλεπιδρούν μεταξύ τους μέσω μιας cross-platform και language – independent σύμβαση η οποία ονομάζεται Document Object Model (DOM).

2.1.4. Δομή ενός αρχείου HTML

Η HTML γράφεται υπό μορφή HTML elements, που αποτελούνται από ετικέτες (tags) τις οποίες τις χρησιμοποιεί για να δώσει συγκεκριμένες οδηγίες στον browser για την μορφοποίηση κειμένου, για την δημιουργία συνδέσμων (links) μετάβασης ανάμεσα των σελίδα, για την εισαγωγή εικόνων, ήχου κ.α. τα πάντα σε αυτή την γλώσσα περιγράφονται με τις ετικέτες οι οποίες έχουν αρχή και τέλος, καθώς επίσης δεν γίνεται διάκριση μεταξύ πεζών και κεφαλαίων.

Για έναν φυλλομετρητή, για να μπορέσει να καταλάβει ότι ένα αρχείο είναι HTML, πρέπει να υπάρχουν μέσα στο έγγραφο κάποιες βασικές ετικέτες. Αυτές είναι οι εξής:

- Ετικέτα html
- Ετικέτα head
- Ετικέτα body

Όταν ένας Web Browser ανοίγει ένα αρχείο HTML τα στοιχεία (tags) μεταφράζονται σε κατάλληλα χαρακτηριστικά με αποτελέσματα στην εμφάνιση και στην λειτουργικότητα της συγκεκριμένης σελίδας.

Ο ρόλος του browser είναι να διαβάζει τα HTML documents, και να συνθέσει web pages, και έτσι, τις ετικέτες που θα διαβάσει δε θα τις εμφανίσει, παρά θα τις χρησιμοποιήσει για να παρουσιάσει το content του webpage.

Οι ετικέτες γράφονται ανάμεσα στα σύμβολα < ... > και σχεδόν πάντα όλες οι ετικέτες αποτελούνται από μια ετικέτα με αρχή και τέλος και ανάμεσα σε αυτές υπάρχει το κείμενο που θέλουμε να χαρακτηρίζεται από αυτές. Την ετικέτα τέλους την ξεχωρίζουμε από τον χαρακτήρα / πριν το όνομα της ετικέτας.

Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), και η πρώτη αναφέρεται ως *open tag* και η δεύτερη ως *close tag*. Ανάμεσα στις ετικέτες, μπορεί να τοποθετηθεί κείμενο, πίνακες, εικόνες κλπ.

Τα HTML elements χρησιμοποιούνται για τη δόμηση όλων των websites. Η HTML προβλέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, ενώ δύναται να εμφανίζονται και interactive forms. Επίσης, υπάρχει η δυνατότητα να ενσωματώνονται JavaScript και όχι μόνο, scripts, προκειμένου να μετατρέψουν τις ιστοσελίδες από statics σε interactive.

Για να γράψουμε κώδικα HTML χρειαζόμαστε έναν οποιονδήποτε editor όπως πχ. Notepad και δίνοντας κατάληξη .html Οι σελίδες στο Internet πιθανώς να έχουν διαφορετικές καταλήξεις όπως .htm , η οποία είναι επίσης μια αποδεκτή επέκταση καθώς και άλλες επεκτάσεις όπως .jps (Java Server Pages) , .asp (Microsoft Active Server Pages) ή .php (PHP) οι οποίες μεταγλωτίζονται στο διακομιστή και παράγεται στην έξοδο κώδικας Html.

2.1.5. Βασικές ετικέτες

Ετικέτα <html>

Η ετικέτα αυτή περικλείει όλο τον κώδικα της σελίδας και δηλώνει στον φυλλομετρητή ότι ο κώδικας αυτός είναι γραμμένος σε HTML.

Ετικέτα <head>

Η ετικέτα αυτή περιλαμβάνει διάφορες πληροφορίες σχετικά με το έγγραφο HTML όπως το τίτλο της σελίδας καθώς και συνδέσμους (links) για την εισαγωγή CSS , JavaScript και άλλων αρχείων που είναι απαραίτητα για την μορφοποίηση της ιστοσελίδας. Το κομμάτι του κώδικα αυτού παραμένει κρυφό και δεν εμφανίζεται από το φυλλομετρητή (Web Browser).

Ετικέτα <body>

Η ετικέτα αυτή αποτελεί το σώμα του εγγράφου HTML και περιλαμβάνει όλο το κείμενο, τους συνδέσμους τις εικόνες κ.α , δηλαδή όλο των κώδικα που περικλείει η ετικέτα <html> χωρίς την ετικέτα <head>.

Ετικέτα <div>

Η ετικέτα div ορίζει ένα τμήμα στο HTML έγγραφο. Ένα τμήμα ομαδοποιεί τα περιεχόμενα που είναι μέσα στις ετικέτες <div> ...</div> ώστε να μπορούμε να ορίζουμε ένα ενιαίο στυλ (χρώμα, γραμματοσειρά, στοίχιση) με την ιδιότητα style. Συνήθως πριν και μετά το τμήμα div ο browser αφήνει μια γραμμή κενή.

Ετικέτα <p>

Η ετικέτα <p> ομαδοποιεί το κείμενο σε μια παράγραφο, αφήνοντας αυτόματα μια κενή γραμμή πριν την αρχή της παραγράφου και μια μετά το τέλος αυτής. Καλό είναι να αποφεύγουμε να χρησιμοποιούμε την ετικέτα αυτή όταν μπορούμε να ομαδοποιήσουμε το κείμενο με άλλες ετικέτες.

**Ετικέτα
**

Η ετικέτα
 αναγκάζει τον browser να σταματήσει να τυπώνει στην τρέχουσα γραμμή και έτσι το αμέσως επόμενο κείμενο ξεκινά από το αριστερό περιθώριο της επόμενης γραμμής. Επίσης, δεν προσθέτει επιπλέον χώρο πάνω ή κάτω από τη νέα γραμμή και δεν αλλάζει τη γραμματοσειρά ή το στυλ του κειμένου.

Ετικέτα <h1.....h6>

Η ετικέτα αυτή αποτελεί την επικεφαλίδα της ιστοσελίδας και εμφανίζεται με κεφαλαία γράμματα στην HTML. Οι επικεφαλίδες χωρίζονται σε επίπεδα με βάση την αξία τους με την ετικέτα <h1> να αποτελεί ετικέτα πρώτου επιπέδου, <h2> ετικέτα δευτέρου επιπέδου κτλ.

**Ετικέτα **

Με την ετικέτα δημιουργούμε στοιχεία (items) σε μία λίστα η οποία μπορεί να είναι ταξινομημένη (order lists) ή όχι (unorder lists).

Ετικέτα <link>

Με την ετικέτα <link> ορίζουμε την σχέση μεταξύ του τρέχον εγγράφου με ένα εξωτερικό αρχείο, το οποίο συνήθως είναι αρχείο CSS ή JavaScript. Περιλαμβάνεται στην ετικέτα <head>.

Ετικέτα <a>

Η ετικέτα <a> περιλαμβάνει συνδέσμους προς άλλες σελίδες χρησιμοποιώντας την ιδιότητα href. Οι πιο σημαντικές ιδιότητες της ετικέτας είναι η href η οποία ορίζει το URL του συνδέσμου και η target η οποία ορίζει που θα ανοίξει το URL (στο ίδιο tab ή σε νέο tab). Επίσης οι σύνδεσμοι μπορούν να είναι είτε εσωτερικοί (αρχική σελίδα , νεα , επικοινωνία) , είτε εξωτερικοί προς άλλες σελίδες όπως www.google.gr

Ετικέτα <textarea>

Με την ετικέτα <textarea> εισάγουμε Περιοχή Κειμένου στην Φόρμα μας. Στην Περιοχή Κειμένου ο επισκέπτης μπορεί να γράψει κείμενο χωρίς περιορισμό στον αριθμό των χαρακτήρων. Επίσης μπορούμε να ορίσουμε τις διαστάσεις της περιοχής κειμένου είτε με τις ιδιότητες cols και rows , είτε με CSS.

2.1.6. Δείγμα Κώδικα HTML

```
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>This is a Heading</h1>
```

```
<p>This is a paragraph.</p>  
</body>  
</html>
```

Εξήγηση παραδείγματος:

- Με τις ετικέτες `<html>...</html>` ο φυλλομετρητής καταλαβαίνει ότι το έγγραφο μας είναι ένα html έγγραφο. Το κείμενο ανάμεσα στις ετικέτες περιγράφει την ιστοσελίδα.
- Το κείμενο που βρίσκεται ανάμεσα στις ετικέτες `<head>..... </head>` περιέχει τις πληροφορίες για την σελίδα που βρισκόμαστε, όπως γλώσσα συγγραφής, τίτλο σελίδας (αυτός που εμφανίζεται στην καρτέλα ενός φυλλομετρητή), δήλωση κώδικα CSS, JavaScript και άλλα.
- Το κείμενο που βρίσκεται ανάμεσα στις ετικέτες `<body>.....</body>` είναι το ορατό κομμάτι μίας HTML σελίδας. Ότι θέλουμε να εμφανίζεται στην ιστοσελίδα μας το περιλαμβάνουμε ανάμεσα σε αυτές ετικέτες.
- Το κείμενο που βρίσκεται ανάμεσα στις ετικέτες `<title>.....</title>` είναι ο τίτλος που αναφερθήκαμε παραπάνω, δηλαδή είναι ο τίτλος που εμφανίζεται στην ανοιχτή καρτέλα ενός φυλλομετρητή.

2.2. HTML 5 (HyperText Markup Language 5)



Εικόνα 7 official logo HTML 5

Η HTML5 είναι η τελευταία έκδοση της γνωστής γλώσσας προγραμματισμού HTML για την κατασκευή ιστοσελίδων. Από την πρώτη HTML έχουν αλλάξει πολλά στον παγκόσμιο ιστό, και έτσι η ανάγκη για standards και πλήρη συμβατότητα οδήγησε το World Wide Web Consortium (W3C) και το Web Hypertext Application Technology

Working Group (WHATWG) να συνεργαστούν και να δημιουργήσουν την νέα αυτή γλώσσα.

Σε αυτήν την έκδοση δίνεται μεγάλη έμφαση στην δημιουργία responsible ιστοσελίδων ,σελίδων που προσαρμόζονται σε διαφορετικές συσκευές. Για τους χρήστες κινητών συσκευών (smartphones), που πιθανόν πλοηγούνται στην ιστοσελίδα με περιορισμένο bandwidth και είναι αυτοί στους οποίους απευθύνεται κυρίως η τεχνική της responsive σχεδίασης, θα πρέπει η ιστοσελίδα όχι μόνο να ανταποκρίνεται στον περιορισμένο «χώρο» της οθόνης τους, αλλά και να την φορτώνει όσο το δυνατόν ταχύτερα.

Επίσης, προσφέρει πληθώρα απο νέες δυνατότητες που απλοποιούν και διευκολύνουν στην ανάπτυξη ενός ιστοτόπου. Προστίθενται νέες ετικέτες (tags) που προσφέρουν πιο ουσιαστικό κώδικα στις μηχανές αναζήτησης και νέες καινοτομίες που απλοποιούν την υλοποίηση διαφόρων εργασιών αναιρώντας την χρήση κώδικα JavaScript και plug-ins όπως Adobe Flash κτλ. Σκοπός είναι η δημιουργία περισσότερο λιτού απλοποιημένου και γρήγορου στη φόρτωση κώδικα .

Η HTML5 είναι ακόμα υπό ανάπτυξη, αλλά μερικοί browsers υποστηρίζουν από τώρα κάποια χαρακτηριστικά της και κάποιες λειτουργίες της.

Κάποιοι βασικοί κανόνες που έχουν οριστεί για την HTML5 είναι:

- Βάση για τα νέα χαρακτηριστικά να είναι οι HTML, CSS, DOM, και η JavaScript
- Ελαχιστοποίηση των plugins (όπως το Flash)
- Καλύτερη λειτουργία εντοπισμού λαθών
- Περισσότερο markup για να αντικατασταθεί το scripting
- Πλήρη συμβατότητα ανεξαρτήτως συσκευής

Τα νέα χαρακτηριστικά της HTML5 είναι:

- Το στοιχείο canvas για το drawing
- Τα στοιχεία video και audio για αναπαραγωγή πολυμέσων
- Νέα στοιχεία περιεχομένου όπως τα footer, header, nav και section
- Νέα στοιχεία δημιουργίας φόρμας όπως τα calendar, date, time, email, url και search

2.2.1. Νέες ετικέτες

Οι προδιαγραφές HTML5 περιλαμβάνουν μια σειρά νέων σημασιολογικών στοιχείων, η αξιοποίηση των οποίων προσφέρει συγκεκριμένο νόημα στα διάφορα τμήματα μιας ιστοσελίδας, όπως μια κεφαλίδα (header), ένα υποσέλιδο (footer), γραμμές εργαλείων πλοήγησης κ.τ.λ. Σε προηγούμενες εκδόσεις της HTML, για τη δημιουργία τέτοιων στοιχείων χρησιμοποιούνταν ετικέτες <div> και με κατάλληλη χρήση ιδιοτήτων id ή/και class γινόταν η διάκριση μεταξύ τους. Το πρόβλημα που προκύπτει είναι ότι οι ετικέτες <div> δεν έχουν καμία σημασιολογική έννοια καθώς δεν υπάρχουν αυστηροί κανόνες που να ορίζουν πως και που χρησιμοποιούνται οι ιδιότητες id και class, γεγονός που καθιστά εξαιρετικά δύσκολο για τον περιηγητή να προσδιορίσει και να αναλύσει τη σημασιολογική δομή του κώδικα της ιστοσελίδας.

Στη συνέχεια, παρουσιάζονται ενδεικτικά οι σημαντικότερες νέες ετικέτες με μια σύντομη περιγραφή της χρήσης τους:

<**header**>: Η ετικέτα «header» ορίζει την επικεφαλίδα ενός εγγράφου. Το περιεχόμενο της επικεφαλίδας αποτελείται συνήθως από γραφικά (πχ. Λογότυπο), κείμενο ή μενού πλοήγησης.

<**footer**>: Η ετικέτα «footer» ορίζει το υποσέλιδο ενός εγγράφου. Το υποσέλιδο, χρησιμοποιείται συνήθως για την παροχή πληροφοριών, στοιχείων επικοινωνίας, copyrights κ.τ.λ.

<**nav**>: Η ετικέτα «nav» ορίζει μια περιοχή που περιέχει το μενού πλοήγησης του εγγράφου (navigation menu).

<**section**>: Η ετικέτα «section» χρησιμοποιείται για την θεματική ομαδοποίηση ίδιου περιεχόμενου, συνήθως κάτω από μία κεφαλίδα.

<**article**>: Η ετικέτα «article» ορίζει ένα αυτοτελές μέρος του εγγράφου (πχ. ένα άρθρο), το οποίο μπορεί να ξαναχρησιμοποιηθεί μόνο του.

<**aside**>: Η ετικέτα «aside» ορίζει ένα τμήμα με μικρότερη σχέση με το υπόλοιπο έγγραφο και διαφορετική ροή. Το περιεχόμενο του μπορεί να τοποθετηθεί για παράδειγμα σαν πλάγια μπάρα.

<hgroup>: Η ετικέτα «hgroup» ομαδοποιεί ένα σύνολο από επικεφαλίδες ενός εγγράφου. Συχνά, ερχόμαστε αντιμέτωποι με μια επικεφαλίδα κάτω από την οποία εμφανίζεται μια άλλη με μικρότερα γράμματα για να δηλώσει ότι είναι υποσύνολο ή παιδί της πρώτης.

<details>: Η ετικέτα «details» ορίζει επιπλέον πληροφορίες για ένα έγγραφο, τις οποίες ο επισκέπτης μπορεί να τις εμφανίζει ή να τις αποκρύπτει. Η επικεφαλίδα είναι πάντα ορατή, ενώ το κείμενο που βρίσκεται μέσα στην ετικέτα «details» εμφανίζεται μόνο όταν ο χρήστης πατήσει πάνω στην επικεφαλίδα.

<figure>: Η ετικέτα «figure» αναπαριστά ένα αυτοδύναμο κομμάτι κειμένου που αντιμετωπίζετε ως ένα κομμάτι στη ροή του εγγράφου.

<canvas>: Η ετικέτα «canvas» χρησιμοποιείται για την απεικόνιση γραφικών με την χρήση scripts, για την απεικόνιση γραφημάτων κ.α.

<video>, **<audio>**: Οι ετικέτες «video» και «audio» χρησιμοποιούνται για την αναπαραγωγή των αντίστοιχων πολυμέσων, χωρίς την χρήση βοηθητικών προγραμμάτων (plug-ins).

2.2.2. Μειονεκτήματα στην HTML5

Με τη πρώτη εμφάνιση της html5 ήρθαν τα πάνω κάτω σε ολόκληρο το διαδίκτυο. Αυτό οφείλετε στα νέα χαρακτηριστικά και τις νέες εφαρμογές που εμπεριέχει η νέα html. Είναι όμως τόσο τέλεια όσο τη περιγράφουν ή και αυτή έχει κάποιες ατέλειες; Η αλήθεια είναι ότι και αυτή έχει κάποια μειονεκτήματα.

Υπάρχουν κάποια παραπάνω χαρακτηριστικά και εφαρμογές που κάνουν τη νέα γλώσσα ελκυστική και ανταγωνιστική σε σχέση με τις άλλες γλώσσες, ωστόσο υπάρχουν κάποια προβλήματα στην ασφάλεια, προκλήσεις συγχρονισμού, περιορισμοί στη χωρητικότητα τοπικής αποθήκευσης δεδομένων και κάποια άλλα που θα αναφερθούμε στη συνέχεια.

1. Ασφάλεια

Το προφανές πρόβλημα που υπάρχει στην ασφάλεια είναι ότι ο κάθε χρήστης έχει και το παραμικρό έλεγχο στον κώδικα. Πιο συγκεκριμένα όταν σε έναν browser υπάρχει ένα καλό εργαλείο αποσφαλμάτωσης, το λεγόμενο debugging, ο κάθε hacker θα μπορεί να έχει πρόσβαση σε προσωπικά δεδομένα του καθενός. (Δηλαδή το εργαλείο αυτό έχει εξειδικευμένο λογισμικό που παρέχει στο χρήστη τη δυνατότητα να εκτελέσει το κώδικα βήμα προς βήμα.) Για παράδειγμα στον firefox, όποιος διαθέτει το firebug, που είναι ένα javascriptdebugger θα μπορούσε να δει πως δουλεύουν πολλά sites μόνο εισάγοντας breakpoints σε γραμμές του κώδικα, βλέποντας πως λειτουργεί βήμα προς βήμα. Βέβαια αυτό αν το δούμε από εκπαιδευτική άποψη είναι πολύ χρήσιμο αλλά από πλευρά ασφάλειας είναι καταστροφικό.

2. Η περιορισμένη τοπική αποθήκευση δεδομένων

Οι τοπικές βάσεις δεδομένων που είναι μέσα στα προγράμματα περιήγησης (browser) είναι αυτά που κάνουν πιο εύκολες και πιο απλές τις web εφαρμογές σε συνάρτηση με τα δεδομένα που είναι κρυφή μνήμη του υπολογιστή. Στην html5 υπάρχει η δυνατότητα αποθήκευσης των δεδομένων αλλά δεν μπορούμε να μεταφέρουμε τα δεδομένα σ' ένα άλλο υπολογιστή ή να τα αντιγράψουμε όπως επίσης να κάνεις και backup γιατί είναι κρυμμένα στο browser. Αυτές λοιπόν οι βάσεις δεδομένων ενώ είναι αποθηκευμένες στο πρόγραμμα περιήγησης ο χρήστης δεν μπορεί να έχει τον έλεγχο. Βέβαια υπάρχουν και οι τελευταίες εκδόσεις των browsers που επιτρέπουν στους χρήστες να δουν τις βάσεις δεδομένων που έχουν δημιουργηθεί στον υπολογιστή του αλλά και πάλι οι πληροφορίες που του δίνει είναι περιορισμένες. Παράδειγμα ο safari της Apple αφήνει και όταν διαγράφεται μια βάση δεδομένων αλλά και πάλι δεν σου αφήνει τη προβολή της ούτε τη μεταφορά της σε άλλο μηχάνημα. Μπορεί κάποιος βέβαια κάλος προγραμματιστής να το αποκωδικοποιήσει αν μελετήσει τη δομή τους γιατί δεν είναι όπως ένας πίνακας σ' ένα αρχείο word ή excel για να το διαβάσει κανείς μ' ένα επεξεργαστή κειμένου. Αυτό καθιστά τα δεδομένα λιγότερο χρηστικά από ότι στις desktop εφαρμογές.

3. Ο συγχρονισμός των offline εφαρμογών

Με την τοπική αποθήκευση των δεδομένων που αναφερθήκαμε και πιο πάνω βοήθησε να χρησιμοποιούνται οι web εφαρμογές και όταν είναι offline. Το πρόβλημα που υπάρχει εδώ είναι ο συγχρονισμός. Όσο μία Web εφαρμογή είναι συνδεδεμένη στο internet αποθηκεύει δεδομένα όταν όμως αλλάξουμε browser ή υπολογιστή τότε

αποθηκεύονται ξανά τα ίδια δεδομένα με πριν με αποτέλεσμα να έχουμε διπλά δεδομένα και συνεπώς πρόβλημα στο συγχρονισμό. Μπορεί να γίνει και πιο χειρότερο όταν τα τα ρολόγια από το ένα μηχάνημα στο άλλο να είναι ασύγχρονα μεταξύ τους και έτσι δεν θα είναι σίγουρο για το πότε έγινε η σωστή αποθήκευση.

4. Οι ασυμβατότητες στο format

Μπορεί οι νέες ετικέτες audioκαι videona είναι ένα από τα καλύτερα χαρακτηριστικά της HTML 5 ,όμως δεν δουλεύουν εξίσου απλά όπως για παράδειγμα με την ετικέτα εικόνων γιατί τα formatvideoκαι audioείναι πολλά και δεν υποστηρίζονται όλα .

2.3. Βασικές διαφορές HTML και HTML 5

Η HTML είναι η κύρια γλώσσα σήμανσης που χρησιμοποιείται για την προβολή ιστοσελίδων στο Διαδίκτυο. Το HTML 5 [22][23]αποτελεί μια σημαντική αναθεώρηση της HTML για την περιγραφή του περιεχομένου και της εμφάνισης των ιστοσελίδων. Ως εκ τούτου, αυτή είναι η θεμελιώδης διαφορά μεταξύ HTML και HTML5.

- Ο ήχος και το βίντεο δεν ήταν μέρος προηγούμενων προδιαγραφών έκδοσης HTML, ωστόσο, και οι δύο αποτελούν αναπόσπαστο μέρος των προδιαγραφών HTML5.
- Τα διανυσματικά γραφικά αποτελούν επίσης αναπόσπαστο μέρος της HTML5, όπως SVG και καμβά ενώ τα διανυσματικά γραφικά είναι δυνατά σε HTML με τη βοήθεια διαφόρων τεχνολογιών όπως VML, Silver-light, Flash, κλπ.
- Σε HTML, η προσωρινή μνήμη του προγράμματος περιήγησης μπορεί να χρησιμοποιηθεί ως προσωρινή αποθήκευση. Στην περίπτωση HTML5, χρησιμοποιείται η προσωρινή μνήμη εφαρμογών, η βάση δεδομένων SQL και η διαδικτυακή αποθήκευση.
- Η υποστήριξη αποθήκευσης εκτός σύνδεσης δεν είναι καλή σε HTML, ενώ το HTML5 υποστηρίζει την αποθήκευση εκτός σύνδεσης πολύ καλά.
- Οι υποδοχές Web δεν είναι διαθέσιμες σε HTML. Ωστόσο, οι υποδοχές ιστού είναι διαθέσιμες σε HTML5 και παρέχουν πλήρη αμφίδρομη επικοινωνία. Αυτή είναι μια άλλη διαφορά μεταξύ HTML και HTML5.
- Το HTML δεν επιτρέπει JavaScript να εκτελείται σε πρόγραμμα περιήγησης ιστού, εκτελείται σε ίδιο νήμα με τη διεπαφή του προγράμματος περιήγησης. Το HTML5 επιτρέπει τη λειτουργία JavaScript στο παρασκήνιο, η οποία είναι δυνατή λόγω του API του JS Web worker στο HTML5.

- Το HTML λειτουργεί ομαλά με όλα τα παλιά προγράμματα περιήγησης ενώ τα περισσότερα από τα σύγχρονα προγράμματα περιήγησης ιστού έχουν αρχίσει να υποστηρίζουν τις προδιαγραφές HTML5. Αυτά τα προγράμματα περιήγησης περιλαμβάνουν: Firefox, Mozilla, Chrome, Opera, Safari κ.λπ.
- Η HTML5 υποστηρίζει τη γεωγραφική τοποθέτηση, σε αντίθεση με την HTML.

2.4. CSS (Cascading Style Sheets)[24]



Εικόνα 8 CSS - Cascading Style Sheets

Εισαγωγή

Η CSS (Cascading Style Sheets)[25] είναι μια γλώσσα περιγραφής στυλ που χρησιμοποιείται για να περιγράψει την εμφάνιση και την μορφοποίηση (presentation semantics) ενός εγγράφου που είναι γραμμένο σε μία σημασιολογική γλώσσα.. Είναι, δηλαδή, μία γλώσσα style sheet για την περιγραφή της παρουσίασης ενός εγγράφου που έχει γραφτεί σε μία markup language, π.χ. HTML. Αποτελεί μία βασική τεχνολογία του World Wide Web και ένα χρήσιμο εργαλείο που μας επιτρέπει να αλλάζουμε την εμφάνιση και την διάταξη (layout) των ιστοσελίδων, προσφέροντας πληθώρα απο νέες δυνατότητες που δεν ήταν εφικτές με την χρήση της HTML.

Η πιο κοινή εφαρμογή της γλώσσας αυτής είναι σε έγγραφα που είναι γραμμένα σε HTML, όπου καθορίζει την εμφάνιση αυτών (μιας ιστοσελίδας και γενικότερα ενός ιστοτόπου). Αν και ο συντάκτης του εγγράφου συνδέει τυπικά το έγγραφο με κάποιο CSS, ο αναγνώστης μπορεί να χρησιμοποιήσει κάποιο διαφορετικό CSS για να προβληθεί το έγγραφο αυτό, δίνοντας έτσι την δυνατότητα στους χρήστες να μπορούν να επιλέγουν ακόμη και τον τρόπο της δομής και της εμφάνισης ενός εγγράφου.

Η CSS διαχωρίζει την παρουσίαση και το περιεχόμενο, πχ τη διάταξη, τα χρώματα και τις γραμματοσειρές. Με αυτόν τον τρόπο, βελτιώνεται η προσβασιμότητα του content, παρέχεται πιο μεγάλη ευελιξία και έλεγχος στα χαρακτηριστικά της παρουσίασης, και

επιτρέπει σε πολλαπλά websites να μοιράζονται μορφοποίηση διαμορφώνοντας το αντίστοιχο CSS σε ξεχωριστό αρχείο .css.

Οι προδιαγραφές της CSS τηρούνται από τον διεθνή οργανισμό προτύπων W3C (World Wide Web Consortium). Πέραν της HTML, και άλλες markup languages υποστηρίζουν το CSS, όπως η XHTML, που υποστηρίζεται και ως XML, SVG και XUL.

Η σύνταξη θεωρείται απλή καθώς χρησιμοποιεί συγκεκριμένες λέξεις κλειδιά για να προσδιορίσει τα ονόματα διαφόρων ιδιοτήτων (χρώμα, μέγεθος, διάστιχο, κλπ). Ένα έγγραφο CSS αποτελείται από μια λίστα κανόνων. Ο κάθε κανόνας απευθύνεται σε ένα ή και περισσότερα στοιχεία. Ο κάθε κανόνας αποτελείται από μια λίστα προσδιορισμού ιδιοτήτων στην οποία καθορίζονται οι ιδιότητες δομής και μορφοποίησης των στοιχείων για τα οποία απευθύνεται ο κανόνας αυτός.

Η καλύτερη προσβασιμότητα στα περιεχόμενα που μας παρέχει η χρήση του CSS δεν ωφελεί μόνο τους σχεδιαστές αλλά και τις μηχανές αναζήτησης όπως το Google, Yahoo, Bing κλπ., διότι τους παρέχεται πιο ξεκάθαρο το περιεχόμενο χωρίς να “μπερδεύονται” με την μορφοποίηση. Άξια αναφοράς είναι επίσης και η ταχύτητα που επιτυγχάνουμε στη φόρτωση σελίδων οι οποίες χρησιμοποιούν CSS. Όταν επισκεπτόμαστε μία σελίδα ο φυλλομετρητής μας αποθηκεύει το CSS στην μνήμη κι έτσι όταν η σελίδα ξαναφορτωθεί δε χρειάζεται να ξανακατεβάσουμε όλες τις ρυθμίσεις μορφοποίησής της.

Συνοψίζοντας, μπορούμε να πούμε ότι, η χρήση της CSS πραγματοποιείται με την εφαρμογή κανόνων στο περιεχόμενο της HTML, προσφέροντας έναν πιο δομημένο και ευανάγνωστο κώδικα. Ένα σημαντικό πλεονέκτημα της, είναι παροχή κανόνων που εφαρμόζονται σε ομάδα-ες στοιχείων της HTML και έτσι οι τυχόν αλλαγές εφαρμόζονται σε όλη την ομάδα και δεν χρειάζεται να πηγαίνουμε σε κάθε στοιχείο ξεχωριστά. Επιπλέον, μας δίνεται η δυνατότητα της χρήσης ενός αρχείου CSS σε πολλαπλές ιστοσελίδες γλυτώνοντας πολλές ώρες συγγραφής κώδικα. Τα αρχεία css φορτώνονται μόνο μια φορά απο τον browser και διατηρούνται σε την διάρκεια της πλοήγησης μειώνοντας παράλληλα τον όγκο των δεδομένων που ένας φυλλομετρητής (browser) καλείται να “κατεβάσει”.

Η σύνδεση ενός αρχείου css επιτυγχάνεται με την χρήση της παρακάτω εντολής στην <head> ετικέτα του HTML κώδικα:

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

2.4.1. Η Ιστορία της CSS

Ο Håkon Wium Lie ήταν ο πρώτος επικεφαλής τεχνικός της εταιρείας Opera Software και συνδημιουργός των προτύπων του CSS.

Η ανάπτυξη του HTML, του CSS και του DOM είχε λάβει χώρα σε μία ομάδα, το HTML Review Editorial Board (ERB). Στις αρχές του 1997, η ERB χωρίστηκε σε τρεις ομάδες εργασίας: Ομάδα εργασίας HTML, υπό την προεδρία του Dan Connolly του W3C. Ομάδα εργασίας DOM, υπό την προεδρία του Lauren Wood από το SoftQuad και η ομάδα εργασίας CSS, υπό την προεδρία του Chris Lilley του W3C.

Η ομάδα εργασίας CSS άρχισε να ασχολείται με θέματα που δεν είχαν αντιμετωπιστεί με το επίπεδο CSS 1, με αποτέλεσμα τη δημιουργία του επιπέδου CSS 2 στις 4 Νοεμβρίου 1997. Δημοσιεύτηκε ως σύσταση του W3C στις 12 Μαΐου 1998. Το επίπεδο CSS 3, που ξεκίνησε το 1998, βρίσκεται ακόμη υπό ανάπτυξη από το 2014.

Το 2005, οι ομάδες εργασίας CSS αποφάσισαν να επιβάλουν αυστηρότερα τις απαιτήσεις για τα πρότυπα. Αυτό σήμαινε ότι ήδη δημοσιευμένα πρότυπα όπως το CSS 2.1, το CSS 3 Selectors και το CSS 3 Text τραβήχτηκαν από την υπογήφια σύσταση σε επίπεδο εργασίας.

Παραλλαγές

Το CSS έχει διάφορα επίπεδα και προφίλ. Κάθε επίπεδο CSS βασίζεται στο τελευταίο, τυπικά προσθέτοντας νέα χαρακτηριστικά και συνήθως χαρακτηρίζεται ως CSS1, ως CSS2, ως CSS3 και ως CSS4.

2.4.2. Σύνταξη

Η σύνταξη της CSS είναι απλή και αποτελείται από μια λίστα κανόνων(rules). Κάθε rule ή σύνολο rules αποτελείται από έναν ή και πιο πολλούς selectors και ένα declaration block (μπλοκ δήλωσης).

Αναλυτικά, οι κανόνες αποτελούνται από 2 μέρη: τον επιλογέα (selector) ο οποίος αφορά στο τι θα μορφοποιηθεί και τις ιδιότητες (properties) οι οποίες αφορούν το πως θα μορφοποιηθεί. Η σύνταξη είναι η εξής:

```
επιλογέας {  
  
ιδιότητα1: τιμή1;  
  
ιδιότητα2: τιμή2;
```

ιδιότητα3: τιμή3;

...

}

Selector

Στη CSS, οι selector δηλώνουν ποιο μέρος της σήμανσης ισχύει για κάθε πρότυπο , συνδυάζοντας ετικέτες και χαρακτηριστικά στο ίδιο το markup.

Οι selector ισχύουν για τα ακόλουθα:

- όλα τα στοιχεία συγκεκριμένου τύπου, π.χ. οι κεφαλίδες δευτέρου επιπέδου της html με στοιχεία που καθορίζονται από τα χαρακτηριστικά:
- id που δηλώνει ένα αναγνωριστικό μοναδικό μέσα στο έγγραφο
- class που δηλώνει ένα αναγνωριστικό που μπορεί να σχολιάσει πολλά στοιχεία σε ένα έγγραφο
- ανάλογα με τον τρόπο με τον οποίο τοποθετούνται σε σχέση με άλλους στο document tree.

Οι κλάσεις και τα αναγνωριστικά ID συμβολίζονται από κεφαλαία πεζά, ξεκινούν με γράμματα και μπορούν να περιλαμβάνουν αλφαριθμητικούς, χαρακτήρες και παύλες .

Μια κλάση μπορεί να ισχύει για οποιοδήποτε αριθμό στοιχείων. Ένα αναγνωριστικό ID μπορεί να εφαρμοστεί σε ένα μόνο στοιχείο.

Οι ψευδοκλάσεις χρησιμοποιούνται σε selector της CSS για να επιτρέπουν τη μορφοποίηση με βάση πληροφορίες που δεν περιέχονται στο document tree (δέντρο εγγράφων).

Ένα παράδειγμα μιας ευρέως χρησιμοποιούμενης ψευδο-κατηγορίας είναι: hover, το οποίο αναγνωρίζει το περιεχόμενο μόνο όταν ο χρήστης "δείχνει" στο ορατό στοιχείο, συνήθως κρατώντας το δείκτη του ποντικιού πάνω του. Συνδέεται σε έναν selector πως σε ένα: hover ή #elementid: hover. Μια ψευδο-τάξη ταξινομεί τα στοιχεία του εγγράφου, όπως: link ή :visited, ενώ ένα ψευδο-στοιχείο κάνει μια επιλογή που μπορεί να αποτελείται από μερικά στοιχεία, όπως :: first-line ή :: first-letter.

Για τους επιλογείς δύνανται να γίνουν πολλοί συνδυασμοί για να επιτευχθεί μεγάλη εξειδίκευση (specificity) και ευελιξία. Πολλαπλοί selector μπορούν να ενωθούν σε μια

λίστεα σε απόσταση για να καθορίσουν στοιχεία ανά τοποθεσία, τύπο στοιχείου, id, κλάση ή οποιονδήποτε συνδυασμό αυτών. Η σειρά των επιλογέων είναι σημαντική. Για παράδειγμα, το `div .myClass {color: red;}` ισχύει για όλα τα στοιχεία της κλάσης `myClass` που βρίσκονται μέσα σε στοιχεία `div`, ενώ `.myClass div {color: red;}` ισχύει για όλα τα στοιχεία `div` που είναι στοιχεία της κλάσης `myClass`.

2.4.2.1. Διαφορά μεταξύ Ιδιοτήτων και Κλάσεων

Πριν ξεκινήσουμε να αναλύουμε τους κανόνες της CSS είναι σημαντικό να κατανοήσουμε την διαφορά των ιδιοτήτων (ids) και των κλάσεων (Classes). Όπως αναφέραμε οι κανόνες αποτελούνται από επιλογείς (selectors). Οι επιλογείς προσδιορίζουν το τι θα αλλάζει σε ένα HTML αρχείο.

Το περιεχόμενο ενός HTML αρχείου αποτελείται από ετικέτες (tags) που χρησιμοποιούνται για την δόμηση του περιεχομένου. Αυτές οι ετικέτες (tags) αποτελούν τους επιλογείς (selectors) της CSS, η οποία καλείται για να μορφοποιήσει το περιεχόμενο. Οι δυνατότητες όμως των ετικετών είναι περιορισμένες και δεν μπορούν να καλύψουν όλες τις δυνατές κατηγορίες περιεχομένου της οποίες επιθυμούμε να μορφοποιήσουμε.

Έτσι, εισάγεται η έννοια των ιδιοτήτων (ids) και κλάσεων (Classes) οι οποίες έχουν ως στόχο να διαφοροποιήσουν ίδιες ετικέτες (tags) μεταξύ τους. Για παράδειγμα, μπορούμε να έχουμε `<div class="main content">` και `<div class="sidebar content">` για να εφαρμόσουμε μέσω CSS διαφορετική μορφοποίηση. Επίσης, μπορούμε να έχουμε και `<div id="footer">` για να ομαδοποιήσουμε μια ετικέτα που έχει ένα χαρακτηριστικό περιεχόμενο. Επομένως, χρησιμοποιούνται ως επιλογείς (selectors) από την CSS.

Παρακάτω παρουσιάζεται η διαφορά μεταξύ τους:

Οι ιδιότητες (ID's) είναι μοναδικές:

- κάθε ετικέτα (tag) μπορεί να έχει ΜΟΝΟ ένα id
- κάθε σελίδα έχει ΜΟΝΟ μια ετικέτα με το συγκεκριμένο id

Οι Κλάσεις (Classes) δεν είναι μοναδικές:

- Πολλές ετικέτες (tags) μπορούν να ανήκουν στην ίδια κλάση

- Κάθε ετικέτα μπορεί να ανήκει σε περισσότερες απο μια κλάσεις.

2.4.3. Επιλογείς

Οι επιλογείς της CSS είναι:

- *

Όταν ο επιλογέας είναι ένας χαρακτήρας αστερίσκου, τότε οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS εφαρμόζονται σε κάθε στοιχείο της σελίδας μας. Όπως είναι κατανοητό, συνήθως δεν είναι και πολύ χρήσιμος επιλογέας από μόνος του, και χρησιμοποιείται κυρίως σε συνδυασμό με άλλους.

- στοιχείο

Όταν ο επιλογέας αποτελείται απλά από το όνομα ενός html tag, τότε οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS εφαρμόζονται σε κάθε τέτοιο στοιχείο html. Για παράδειγμα, ο επιλογέας p θα εφαρμοστεί σε οτιδήποτε στη σελίδα μας περιλαμβάνεται εντός των tags <p>...</p>, ο επιλογέας table θα εφαρμοστεί σε όλους τους πίνακες στη σελίδα μας, ο επιλογέας img θα αφορά όλες τις εικόνες στη σελίδα κοκ. Προφανώς όταν θέλουμε να εφαρμόσουμε κάποιες ιδιότητες CSS σε ολόκληρη τη σελίδα, χρησιμοποιούμε ως επιλογέα body μιας και όλο το ορατό τμήμα της σελίδας περιέχεται εντός των tags <body>...</body>.

- .όνομα_κλάσης

Όταν ο επιλογέας μας περιλαμβάνει μια τελεία (.) στην αρχή του, τότε ο browser ψάχνει όσα στοιχεία στη σελίδα μας περιλαμβάνουν την ιδιότητα class και εφαρμόζει τις ιδιότητες που θα γράψουμε στον κανόνα CSS αυτό σε οποιοδήποτε στοιχείο περιλαμβάνει την κλάση «όνομα_κλάσης» στην ιδιότητα class του. Φυσικά ως όνομα_κλάσης μπορούμε να γράψουμε οτιδήποτε αποτελείται από γράμματα, αριθμούς, παύλες και χαρακτήρες underscore (_) και να ξεκινάει με γράμμα. Αξίζει να σημειωθεί ότι μπορεί το ίδιο στοιχείο να ανήκει σε περισσότερες από μια κλάσεις, διαχωρισμένες με κενά μέσα στην class html attribute του. Πχ <p class="emphasis bodytext">...</p>.

Για παράδειγμα, ο παρακάτω κανόνας CSS:

```
.emphasis {
```

```
color: red;
```

```
}
```

θα κάνει κόκκινα τα γράμματα και στο στοιχείο `<p class="emphasis">blah blah</p>`,

και στο στοιχείο `<div class="emphasis otherclass">blah blah</div>`.

- #όνομα_id

Όταν ο επιλογέας μας περιλαμβάνει ένα χαρακτήρα δέσσης (#) στην αρχή του, τότε ο browser εφαρμόζει τις ιδιότητες που θα γράψουμε στο στοιχείο το οποίο περιλαμβάνει την ιδιότητα `id="όνομα_id"`.

Εκτός από τους κλασσικούς επιλογείς η CSS μας παρέχει μια σειρά από σύνθετους επιλογείς που μας επιτρέπουν να απομονώνουμε κείμενο και να δημιουργούμε ενδιαφέρουσες μορφοποιήσεις.

Παρακάτω παρουσιάζεται μια σειρά από τους βασικούς σύνθετους επιλογείς:

- στοιχείο1 , στοιχείο2

Χρησιμοποιείται όταν θέλουμε να εφαρμόσουμε την ίδια μορφοποίηση σε περισσότερα από ένα στοιχεία.

πχ. a, p, h1 {

```
color: blue;
```

```
}
```

- στοιχείο1 στοιχείο2 στοιχείο x

Χρησιμοποιείται για την εφαρμογή μορφοποίησης στο στοιχείο που είναι `<<παιδί>>` του στοιχείου (n-1) , που είναι `<<παιδί>>` του στοιχείου (n-2) κ.ο.κ

πχ. li a {

```
color: red;
```

```
}
```

δίνει κόκκινο χρώμα γραμματοσειράς σε όλες τις ετικέτες (tag) a τα οποία ανήκουν στην ετικέτα (tag) li και αποτελούν απογονοί του.

- στοιχείο1 + στοιχείο2

Όταν χρησιμοποιείται, η μορφοποίηση εφαρμόζεται μόνο στο πρώτο στοιχείο2 του κάθε στοιχείου1 στο html κώδικα.

```
πχ. li + p {
color: red;
}
```

δίνει κόκκινο χρώμα γραμματοσειράς στην πρώτη και μόνο παράγραφο p κάθε λίστας li.

Ψευδοκλάσεις

Στην CSS έχουμε και τις ψευδοκλάσεις. Ορισμένες φορές τις χρησιμοποιούμε (pseudo-classes), τα οποία μας επιτρέπουν να επιλέγουμε στοιχεία τα οποία δεν αποτελούν html elements, αλλά κομμάτια τους ή συγκεκριμένες καταστάσεις τους.

Ουσιαστικά αποτελούν κάποιες λέξεις-κλειδιά που γράφουμε μετά από έναν επιλογέα του τύπου στοιχείο και ξεκινούν με :

- a:link

Χρησιμοποιείται αποκλειστικά για στοιχεία a (Δηλαδή ως εξής: a:link) και αφορά τους συνδέσμους που ο χρήστης δεν έχει ακόμη επισκεφθεί.

- a:visited

Επίσης χρησιμοποιείται αποκλειστικά για στοιχεία a (Δηλαδή ως εξής: a:visited) και αφορά τους συνδέσμους που ο χρήστης έχει επισκεφθεί.

- στοιχείο:active

Αφορά τα στοιχεία τύπου <στοιχείο> τη στιγμή που ο χρήστης έχει πατημένο το ποντίκι πάνω σε αυτά. Πχ ο επιλογέας a:active εφαρμόζεται σε συνδέσμους την ώρα που ο χρήστης έχει πατημένο το ποντίκι πάνω τους.

- στοιχείο:hover

Από τις πιο συχνά χρησιμοποιούμενες ψευδο-κλάσεις. Αφορά τα στοιχεία τύπου <στοιχείο> τη στιγμή που ο χρήστης έχει το δείκτη του ποντικιού πάνω σε κάποιο από

αυτά (χωρίς να πατάει κάποιο πλήκτρο). Μπορεί να μας βοηθήσει να δημιουργήσουμε διάφορα όμορφα εφέ, τα οποία παλιότερα ήταν εφικτά μόνο με javascript.

- στοιχείο:focus

Χρησιμοποιείται κυρίως για στοιχεία φορμών και εφαρμόζεται στα στοιχεία τύπου <στοιχείο> που εκείνη τη στιγμή έχουν «focus», παραδείγματος χάριν, ένα πεδίο κειμένου στο οποίο ο χρήστης έκανε κλικ για να εισάγει κείμενο.

Ομοίως, τα ψευδο-στοιχεία αποτελούν κάποιες λέξεις-κλειδιά που γράφουμε μετά από έναν επιλογέα του τύπου στοιχείο και ξεκινούν με ::

- στοιχείο::after

Ψευδό-στοιχείο που χρησιμοποιείται για την εισαγωγή κειμένου μετά από το περιεχόμενο που υπάρχει μέσα στο στοιχείο τύπου <στοιχείο>.

- στοιχείο::before

Ψευδό-στοιχείο που χρησιμοποιείται για την εισαγωγή κειμένου πριν από το περιεχόμενο που υπάρχει μέσα στο στοιχείο τύπου <στοιχείο>.

- στοιχείο::selection

Ψευδό-στοιχείο που χρησιμοποιείται για την εισαγωγή χρώματος γραμματοσειράς και φόντου του κειμένου που υπάρχει μέσα στο στοιχείο τύπου <στοιχείο> , όταν το ποντίκι κάνει <<κλικ>> επιλογής στο κείμενο.

Τέλος, έχουμε τις ιδιότητες.

- color

Αφορά το χρώμα του κειμένου, αλλά αν δεν οριστεί χρώμα περιγράμματος (μέσω της ιδιότητας border-color), ο browser χρησιμοποιεί αυτό που ορίστηκε στην ιδιότητα color. Τα χρώματα μπορούν να εισαχθούν είτε σε μορφή RGB (πχ color: rgb(255,128,30);), είτε σε μορφή hex (πχ color: #ff801e;) είτε με τη μορφή κάποιου keyword (πχ color:orange;).

- font-size

- Αφορά το μέγεθος της γραμματοσειράς. Οι τιμές που δέχεται μπορούν να είναι εκφρασμένες σε ένα μεγάλο πλήθος μονάδων μεγέθους, από τις οποίες οι πιο ευρέως διαδεδομένες είναι τα pixels (πχ font-size: 12px;) και οι στιγμές (πχ font-size:10pt).

- font-family

Η ιδιότητα αυτή μας επιτρέπει να ορίσουμε ένα πλήθος γραμματοσειρών που θα χρησιμοποιηθούν για το κείμενο, κατά σειρά προτίμησης. Ουσιαστικά μέσω αυτής ορίζουμε τη γραμματοσειρά του κειμένου, αλλά μας επιτρέπει να ορίσουμε και εναλλακτικές επιλογές, ώστε αν η γραμματοσειρά που ορίσαμε δεν υπάρχει στον υπολογιστή του χρήστη, να μην μας καταστρέψει ο browser την εμφάνιση της σελίδας επιλέγοντας όποια αυτός νομίζει, αλλά την επόμενη επιλογή μας. Παράδειγμα χρήσης της ιδιότητας: font-family: Calibri, Trebuchet MS, Verdana, sans-serif;

- font-style

Σε αντίθεση με αυτό που θα περίμενε κανείς από μια ιδιότητα με αυτό το όνομα, δηλαδή την εφαρμογή πολλών και διαφόρων «εφέ» στο κείμενο, στην πραγματικότητα αφορά μόνο την περίπτωση όπου το κείμενο θα είναι πλάγιο. Για άλλα εφέ, χρησιμοποιούνται άλλες ιδιότητες, οι οποίες θα εξεταστούν παρακάτω. Οι πιθανές τιμές της είναι normal, italic και oblique.

- font-weight

Αφορά το «βάρος» της γραμματοσειράς και στην πράξη χρησιμοποιείται για να ορίσει αν το κείμενο μας θα είναι έντονο ή όχι, μιας και οι περισσότερες γραμματοσειρές που χρησιμοποιούνται στο web διατίθενται μόνο σε δύο βάρη:

Κανονικό και έντονο, σε αντίθεση με πιο εξειδικευμένες γραμματοσειρές που χρησιμοποιούνται από γραφίστες, οι οποίες πολλές φορές διατίθενται σε διάφορα βάρη. Οπότε οι τιμές που συνήθως χρησιμοποιούνται σε αυτή την ιδιότητα είναι οι normal και bold, η λειτουργία των οποίων είναι προφανής.

- text-decoration

Μας επιτρέπει να εφαρμόσουμε στο κείμενο μας διάφορα εφέ, συμπεριλαμβανόμενης και της υπογράμμισης. Οι τιμές που δέχεται είναι οι εξής:

- none: Καμία διακόσμηση

- underline: Υπογράμμιση

- overline: Γραμμή πάνω από το κείμενο (ουσιαστικά το αντίθετο της υπογράμμισης)

- line-through: Διαγράμμιση

Οι παραπάνω τιμές μπορούν να χρησιμοποιηθούν και συνδυαστικά, όταν επιθυμούμε να εφαρμόσουμε πάνω από ένα τέτοιο εφέ στο κείμενο μας. Παραδείγματος χάριν `text-decoration: underline overline;`

- `text-align`

Μας επιτρέπει να καθορίσουμε τη στοίχιση του κειμένου μας. Οι πιθανές τιμές είναι `left`, `center`, `right` και `justify`.

- `Font`

Μας επιτρέπει να καθορίσουμε αρκετές ιδιότητες κειμένου με μια μόνο ιδιότητα (συγκεκριμένα τις `font-size`, `font-family`, `font-weight` και `font-style` από όσες αναφέρθηκαν παραπάνω, καθώς και κάποιες άλλες που δεν αναφέρθηκαν). Αν κάποια ιδιότητα δεν συμπεριληφθεί, ο browser θα «υποθέσει» την προεπιλεγμένη τιμή της.

- `background-color`

Χρησιμοποιείται για να ορίσει χρώμα φόντου στα στοιχεία που αφορά ο επιλογέας. Το χρώμα μπορεί να γραφεί σε οποιαδήποτε από τις μορφές που περιεγράφηκαν για την ιδιότητα `color`.

- `background-image`

Χρησιμοποιείται για να ορίσει μια εικόνα φόντου.

- `background-position`

Η ιδιότητα αυτή μας επιτρέπει να ορίσουμε τη θέση που θα τοποθετηθεί η εικόνα φόντου που ορίσαμε με την ιδιότητα `background-image` τόσο κάθετα, όσο και οριζόντια. Συνήθως ως τιμές της χρησιμοποιούμε κάποια `keywords`, τα οποία είναι τα εξής:

- `left`, `center`, `right` : όσον αφορά την οριζόντια θέση της εικόνας
- `top`, `center`, `bottom` : όσον αφορά την κάθετη θέση της εικόνας

- `border-color`

Ρυθμίζει το χρώμα περιγράμματος. Το χρώμα μπορεί να γραφεί σε οποιαδήποτε από τις μορφές που περιγράφηκαν για την ιδιότητα `color`. Αν δεν οριστεί αυτή η ιδιότητα, χρησιμοποιείται το χρώμα που ορίστηκε στην ιδιότητα `color`.

- border-width

Ρυθμίζει το πάχος του περιγράμματος σε κάποια από τις μονάδες μέτρησης που μπορούν να χρησιμοποιηθούν στο CSS, συνηθέστερα σε pixels. Παράδειγμα: border-width: 10px;

- border

Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να ορίσουμε με μία μόνο ιδιότητα όλες τις παραπάνω (όπως είναι η ιδιότητα font για τα χαρακτηριστικά κειμένου). Παραδείγματα χρήσης:

```
border: 10px solid #999999;
```

```
border: 2px dashed;
```

- padding

Ορίζει το κενό που θα υπάρχει μεταξύ των ορίων ενός στοιχείου και των περιεχομένων του. Είναι πολύ σημαντικό να ορίζουμε padding σε στοιχεία στα οποία έχουμε ορίσει κάποιο περίγραμμα, ώστε να μην «κολλάνε» τα περιεχόμενα τους με το περίγραμμα, κάτι που φαίνεται ιδιαίτερα άσχημο και ερασιτεχνικό. Επίσης, καλό είναι να μην είστε ιδιαίτερα φειδωλοί με το padding. Περισσότερο padding δίνει πιο επαγγελματική εμφάνιση (μέχρι κάποιων ορίων φυσικά). Για να ορίσουμε διαφορετικό padding ανά πλευρά, μπορούμε ομοίως με το border, είτε να χρησιμοποιήσουμε τις ιδιότητες padding-top, padding-right, padding-bottom, padding-left, είτε να εισάγουμε και τις τέσσερις τιμές στην ιδιότητα

padding με τη σειρά [top] [right] [bottom] [left], είτε [top,bottom] [right,left].

Παραδείγματα:

```
padding: 8px;
```

```
padding: 2px 6px;
```

```
padding: 0px 6px 6px 6px;
```

- margin

Το αντίθετο ουσιαστικά του padding. Ορίζει τον χώρο μεταξύ των ορίων ενός στοιχείου και όσων το περιβάλλουν. Είναι ιδιαίτερα σημαντικό να ορίζουμε margin σε εικόνες, ώστε να έχουν απόσταση από τα περιεχόμενα τους, μιας και είναι ιδιαίτερα

αντισταθμιστικό να «κολλάνε» με το κείμενο. Φυσικά και εδώ μπορούμε να χρησιμοποιήσουμε είτε τις ιδιότητες `margin-top`, `margin-right`, `margin-bottom`, `margin-left` για να ορίσουμε διαφορετικές τιμές `margin` ανά πλευρά, είτε να εισάγουμε τις τέσσερις διαφορετικές τιμές `margin` με τη σειρά `[top] [right] [bottom] [left]`, ή `[top,bottom] [right,left]`.

Παραδείγματα:

```
margin: 4px;
```

```
margin: 2px 4px;
```

```
margin: 0px 8px 2px 2px;
```

- `width, height`

Όπως είναι προφανές, οι δύο αυτές ιδιότητες ορίζουν το πλάτος και το ύψος ενός στοιχείου, είτε σε ποσοστό (το οποίο υπολογίζεται βάσει του στοιχείου που το περιέχει), είτε σε κάποια μονάδα μήκους. Παραδείγματα:

```
width: 100px;
```

```
width: 90%;
```

```
height: 300px;
```

```
height: 100%;
```

Υπάρχουν και οι ιδιότητες `min-width`, `min-height`, `max-width`, `max-height` οι οποίες ορίζουν τα όρια στα οποία μπορούν να κινούνται οι διαστάσεις ενός στοιχείου, όταν δεν θέλουμε να ορίσουμε συγκεκριμένες διαστάσεις.

2.5. CSS 3 (Cascading Style Sheets 3)



Εικόνα 9 CSS 3 logo

Η CSS3 είναι το τελευταίο πρότυπο για την CSS και είναι πλήρως συμβατή με προηγούμενες εκδόσεις των CSS. Είναι ακόμα υπό ανάπτυξη, αλλά μερικοί browsers υποστηρίζουν από τώρα κάποια χαρακτηριστικά της και κάποιες λειτουργίες της.

Προσφέρει πληθώρα απο νέα χαρακτηριστικά και δυνατότητες μορφοποίησης που πριν δεν ήταν δυνατόν να υλοποιηθούν χωρίς την χρήση JavaScript ή άλλων γλωσσών προγραμματισμού. Ένα απο τα σημαντικότερα πλεονεκτήματά της είναι η δυνατότητα δημιουργίας *responsive* ιστοσελίδων, ιστοσελίδων που προσαρμόζονται σε διαφορετικά περιβάλλοντα με την συγγραφή μιας και μόνο σελίδας. Αυτό έχει ως αποτέλεσμα να μειώνεται αισθητά ο χρόνος κατασκευής καθώς παράγεται λιγότερος και πιο κατανοητός κώδικας. Επιπλέον, στην CSS3 μας δίνεται η δυνατότητα δημιουργίας 3D γραφικών που έως τώρα εμφανιζόντουσαν σαν εικόνες και την εφαρμογή διάφορων <εφέ> δημιουργώντας ένα πιο ευχάριστο περιβάλλον πλοήγησης.

Η χρήση της CSS3, της τελευταίας έκδοσης της CSS, γίνεται αποκλειστικά με τη χρήση HTML5[26]. Ο συγκεκριμένος συνδυασμός τεχνολογιών έχει γίνει ιδιαίτερα δημοφιλής τα τελευταία χρόνια από τους προγραμματιστές και τους σχεδιαστές ιστοσελίδων, λόγω της ευκολίας που παρέχει στη σχεδίαση όμορφων και εντυπωσιακών εφαρμογών.

Η HTML5 και η CSS3 αποτελούν επανάσταση στο χώρο του web development και web design, μιας και εισάγουν αρκετά καινούρια χαρακτηριστικά τα οποία αν συνδυαστούν μεταξύ τους μπορούν να προσφέρουν εκπληκτικές δυνατότητες σε διάφορους τομείς. Με την CSS3 λοιπόν, που είναι η γλώσσα υλοποίησης των γραφικών των ιστοσελίδων, οι σελίδες αποκτούν πολύ διαφορετική όψη με ωραία γραφικά τα οποία φορτώνουν γρήγορα την ίδια στιγμή. Επιπλέον υπάρχει ενσωματωμένη μια λειτουργία στο CSS3 που χρησιμοποιώντας μόνο CSS κώδικα μας δίνεται η δυνατότητα να τοποθετήσουμε εφέ που θυμίζουν λίγο το flash.

2.5.1. Modules

Η CSS3 έχει χωριστεί σε «μονάδες» (modules). Περιέχει την "παλιά προδιαγραφή CSS" (το οποίο έχει χωριστεί σε μικρότερα κομμάτια). Επιπλέον, προστίθενται νέες ενότητες. Μερικές από τις πιο σημαντικές ενότητες CSS3 είναι:

- Οι Selectors

- Το Box Model
- Το φόντο και τα σύνορα (Background and Borders)
- Τιμές εικόνας και αντικατάσταση περιεχομένου
- Επιδράσεις στο κείμενο
- Μετασχηματισμοί 2D/3D
- Animations
- Πολλαπλής Διάταξης Στήλες
- User Interface

2.5.2. Media Queries

Μια από τις σημαντικότερες καινοτομίες της CSS3 είναι η κατασκευή *responsive* ιστοσελίδων. Αυτό επιτυγχάνεται με την χρήση των *Media queries*, όπου μας δίνεται η δυνατότητα να προσαρμόζουμε την ιστοσελίδα στο μέγεθος του περιβάλλοντος που βρισκόμαστε χωρίς να αλλάζουμε το περιεχόμενο του CSS3 αρχείου που συγγράψαμε.

Η δομή των *media queries* :

```
@media not|only mediatype and (media feature) {
  CSS code;
}
```

Η σύνδεση τους στο HTML κώδικα πραγματοποιείται ως εξής:

Να επισημάνουμε πως έχουμε την δυνατότητα να χρησιμοποιήσουμε διαφορετικά αρχεία

CSS για κάθε συσκευή.

```
<link rel="stylesheet" media="mediatype and|not|only (media feature)
href="mystylesheet.css">
```

Παρακάτω δίνονται παραδείγματα της χρήσης των *media queries*:

Max-width

Ο ακόλουθος κώδικας εφαρμόζεται για συσκευές με μέγιστη γωνία θέασης 600px

```
@media screen and (max-width: 600px) {  
  .class {  
    background: #ccc; }  
}
```

Παρατηρούμε πως μέσα στα media queries εισάγουμε κώδικα για να προσαρμόσουμε κατάλληλα τα στοιχεία στο νέο περιβάλλον.

Min-width

Ο ακόλουθος κώδικας εφαρμόζεται για συσκευές με γωνία θέασης μεγαλύτερη των 900px

```
@media screen and (min-width: 600px) {  
  .class {  
    background: #666; }  
}
```

Συνδυασμός των media queries (multiple queries)

```
@media screen and (min-width: 600px) and (max-width: 900px) {  
  .class {  
    background: #333; }  
}
```

2.5.3. Transitions

Τα CSS3 Transitions [27] παρέχουν τη δυνατότητα δημιουργίας animation των στοιχείων της ιστοσελίδας, κι αυτό επιτυγχάνεται λόγω της ιδιότητας τους να μπορούν να αλλάζουν τις τιμές των στοιχείων με βάση το χρόνο. Ακόμη, με τα CSS3 transitions μπορούμε να χρωματίσουμε τη γραμματοσειρά του κειμένου ή να αλλάξουμε το μέγεθος των γραμμάτων της και γενικά να τροποποιήσουμε τα κείμενα μας όπως επιθυμούμε.

Οι ιδιότητες CSS και η περιγραφή τους είναι :

- **transition-property:** Θέτουμε την ιδιότητα στην οποία θα γίνει η αλλαγή τιμής.
- **transition-duration:** Θέτουμε το χρόνο το οποίο θα χρειαστεί η μετάβαση μας να ολοκληρωθεί (σε δευτερόλεπτα).
- **transition-timing function:** Θέτουμε το ρυθμό μεταβολής της τιμής σε σχέση με το χρόνο που ορίσαμε. Πέρα από την επιλογή `ease` που χρησιμοποιήσαμε στο παραπάνω παράδειγμα υπάρχουν και: `ease-out`, `ease-in`, `linear` και η `cubic-bezier`.
- **transition-delay:** Θέτουμε την χρονοκαθυστέρηση που θα έχει πριν ξεκινήσει την μεταβολή της τιμής.

Οι παραπάνω ιδιότητες όπως και πολλές ακόμα στα CSS3 δεν υποστηρίζονται ως CSS standards από πολλούς browsers και έτσι ο κάθε κατασκευαστής έχει προσθέσει τα δικά του προθέματα (prefixes). Αυτά είναι `-webkit-` για τους Safari και Chrome, `-moz-` για τον Firefox και `-o-` για τον Opera. Βασικός κανόνας όταν χρησιμοποιούμε transitions, αλλά και το CSS3 γενικότερα, είναι να χρησιμοποιούμε τα κατάλληλα prefixes ώστε να εμφανίζονται τα εφέ που θέλουμε σωστά σε όλους τους browsers μιας και δεν υπάρχει ακόμα υποστήριξη από όλους τους κατασκευαστές. Η χρήση των CSS3 transitions είναι λίγο δύσκολη για την ώρα λόγω αυτής της ασυμβατότητας, όμως είναι αρκετά γρηγορότερη η υλοποίηση της και απαιτεί σαφώς λιγότερη υπολογιστική ισχύ κατά την εκτέλεση της συγκριτικά με άλλη.

2.5.4. Νέοι επιλογείς

Το CSS3 προσφέρει νέους τρόπους με τους οποίους μπορούμε να γράψουμε κανόνες CSS με νέους επιλογείς CSS, καθώς και έναν νέο συνδυαστή και μερικά νέα ψευδο-στοιχεία.

Τρεις νέοι επιλογείς[28] χαρακτηριστικών:

- **Η αρχή του στοιχείου αντιστοιχεί ακριβώς στο** στοιχείο `[foo ^ = "bar"]` Το στοιχείο έχει ένα χαρακτηριστικό που ονομάζεται `foo` που αρχίζει με `"bar"` π.χ.
- **Το χαρακτηριστικό που τελειώνει αντιστοιχεί ακριβώς στο** στοιχείο `[foo $ = "bar"]` Το στοιχείο έχει ένα χαρακτηριστικό που ονομάζεται `foo` και τελειώνει με `"bar"` π.χ.
- **Το χαρακτηριστικό περιέχει το** στοιχείο **αντιστοίχισης** `[foo * = "bar"]` Το στοιχείο έχει ένα χαρακτηριστικό που ονομάζεται `foo` που περιέχει τη συμβολοσειρά `"bar"` π.χ.

16 νέες ψευδο-κατηγορίες:

- **:ρίζα**
 - Το ριζικό στοιχείο του εγγράφου. Σε HTML αυτό είναι πάντα.
- **: n-παιδί (n)**
 - Χρησιμοποιήστε αυτό για να ταιριάξετε τα ακριβή στοιχεία του παιδιού ή χρησιμοποιήστε μεταβλητές για να λάβετε εναλλασσόμενες αντιστοιχίσεις.
- **: nth-last-child (n)**
 - Ταίριασμα των ακριβών στοιχείων παιδιών που μετράνε από το τελευταίο.
- **: n-o-τύπου (n)**
 - Ταιριάζουν τα αδέσποτα στοιχεία με το ίδιο όνομα πριν από αυτό στο δέντρο εγγράφων.
- **: n-τελευταίου τύπου (n)**
 - Ταίριασμα των αδελφών στοιχείων με το ίδιο όνομα που μετράνε από κάτω.
- **:τελευταίο παιδί**
 - Ταιριάζει με το τελευταίο παιδικό στοιχείο του γονέα.
- **: πρώτος τύπου**
 - Ταίριασμα του πρώτου αδελφικού στοιχείου αυτού του τύπου.
- **: τελευταίο του τύπου**
 - Ταιριάζει με το τελευταίο στοιχείο του αδελφού αυτού του τύπου.
- **:μοναχοπαίδι**
 - Αντιστοιχεί το στοιχείο που είναι το μόνο παιδί του γονέα του.
- **: μόνο τύπου**
 - Αντιστοιχεί το στοιχείο που είναι το μόνο του τύπου του.
- **:αδειάζω**
 - Ταίριασμα του στοιχείου που δεν έχει παιδιά (συμπεριλαμβανομένων των κόμβων κειμένου).
- **:στόχος**
 - Ταίριασμα ενός στοιχείου που είναι ο στόχος του αναφερόμενου URI.
- **: ενεργοποιημένη**
 - Ταίριασμα του στοιχείου όταν είναι ενεργοποιημένη.
- **:άτομα με ειδικές ανάγκες**
 - Ταίριασμα του στοιχείου όταν είναι απενεργοποιημένο.

- **:τετραγωνισμένος**
 - Ταίριασμα του στοιχείου όταν αυτό είναι επιλεγμένο (κουμπιά επιλογής ή πλαίσιο ελέγχου).
- **: δεν (ες)**
 - Ταίριασμα όταν το στοιχείο δεν ταιριάζει με τον απλό επιλογή s.

Ένας νέος συνδυαστής:

- **στοιχείοA ~ στοιχείοB**
 - Ταίριασμα όταν το στοιχείο B ακολουθεί κάπου μετά το στοιχείοA, όχι απαραίτητα αμέσως.

2.5.5. Νέες Ιδιότητες

Το CSS3 εισήγαγε επίσης μια σειρά από νέες ιδιότητες CSS. Πολλές από αυτές τις ιδιότητες ήταν να δημιουργηθούν οπτικά στυλ που πιθανόν να συσχετίζονται περισσότερο με ένα γραφικό πρόγραμμα όπως το Photoshop. Ορισμένα από αυτά, όπως η ακτίνα ακτίνων ή η σκιά κουτιού, είναι γύρω από την εισαγωγή του CSS3. Άλλοι, όπως το flexbox ή ακόμα και το CSS Grid, είναι νεότερα στυλ που θεωρούνται ακόμα συχνά CSS3 προσθήκες.

Στο CSS3, το μοντέλο κουτιού δεν έχει αλλάξει. Αλλά υπάρχει μια δέσμη νέων ιδιοτήτων στυλ φόντου.

- **κλιπ φόντου**
 - Αυτή η ιδιότητα ορίζει πώς πρέπει να αποκοπεί η εικόνα φόντου. Η προεπιλογή είναι το πλαίσιο περιγράμματος, αλλά μπορεί να αλλάξει στο πλαίσιο παρεμβολής ή στο πλαίσιο περιεχομένου.
- **αρχική προέλευση**
 - Αυτή η ιδιότητα καθορίζει εάν το φόντο θα πρέπει να τοποθετείται στο πλαίσιο παραγεμίσματος, στο πλαίσιο περιγράμματος ή στο πλαίσιο περιεχομένου.
- **μέγεθος φόντου**

- Αυτή η ιδιότητα σας επιτρέπει να υποδείξετε το μέγεθος της εικόνας φόντου. Σας επιτρέπει να τεντώσετε μικρότερες εικόνες ώστε να ταιριάζουν στη σελίδα.

Αλλαγές στις υπάρχουσες ιδιότητες στυλ φόντου

Υπάρχουν επίσης κάποιες αλλαγές στις υπάρχουσες ιδιότητες στυλ φόντου:

- **background-repeat**
 - Υπάρχουν δύο νέες τιμές για αυτήν την ιδιότητα: `χώρο` και `στρογγυλό`. Διαστήματος `χώρος` η εικόνα πλακιδίων ομοιόμορφα εντός του κουτιού χωρίς να κόβεται. Ο `στρογγυλός` μετασχηματίζει την εικόνα φόντου έτσι ώστε να πλακάει πολλές φορές στο κουτί.
- **προσάρτηση φόντου**
 - Προστίθεται μια νέα τιμή "`τοπική`" έτσι ώστε το φόντο να μετακινηθεί με το περιεχόμενο του στοιχείου όταν το στοιχείο έχει μια γραμμή κύλισης.
- **Ιστορικό**
 - Η ιδιότητα `shorthand` του φόντου προσθέτει τις ιδιότητες `μεγέθους` και `προέλευσης`.

2.6. Σύγκριση CSS και CSS3[29]

Αν και οι CSS και CSS3 είναι δημοφιλείς επιλογές αξίζει να αναφερθούμε σε μερικές από τις πιο σημαντικές διαφορές τους.

- Η κύρια διαφορά μεταξύ CSS και CSS3 είναι ότι το CSS3 διαθέτει `modules` και μπορεί να χωριστεί σε αυτές. Το CSS είναι η βασική έκδοση και δεν υποστηρίζει `responsive design` σε αντίθεση με την CSS3, που είναι η τελευταία έκδοση.
- Η CSS ως παλαιότερη έκδοση είναι πιο αργό από την CSS3.
- Εκτός από αυτά τα CSS3 έχει πολλές δυνατότητες ευθυγράμμισης. Το CSS3 παρέχει ένα εργαλείο `μεγέθους κουτιού` το οποίο επιτρέπει στο χρήστη να πάρει το σωστό μέγεθος οποιουδήποτε στοιχείου χωρίς να κάνει αλλαγές στις

διαστάσεις ή το γέμισμα. Το CSS δεν διαθέτει κανένα εργαλείο μεγέθους κουτιού και ως εκ τούτου ο χρήστης πρέπει να χρησιμοποιεί τις τυπικές διαδικασίες που ορίζονται για την ευθυγράμμιση του κειμένου.

- Οι κινούμενες εικόνες και οι τρισδιάστατοι μετασχηματισμοί είναι καλύτεροι στην CSS3. Τα στοιχεία μπορούν να μετακινηθούν στην οθόνη με τη βοήθεια φλας και JavaScript. Χρησιμοποιώντας αυτό, τα στοιχεία θα μπορούν επίσης να αλλάξουν το μέγεθος και το χρώμα τους. Όλα τα είδη μεταβάσεων, μετασχηματισμών και κινούμενων σχεδίων μπορούν να γίνουν χρησιμοποιώντας CSS3. Το CSS δεν παρέχει 3D animation και μετασχηματισμούς.
- Η CSS παρέχει βασικά σχήματα χρωμάτων και τυπικά χρώματα. Το CSS3 υποστηρίζει χρώματα RGBA, HSLA, HSL και ντεγκραντέ. Υποστηρίζει επίσης στρογγυλεμένες γωνίες εικόνων για πλαίσια κειμένου.
- Τα μπλοκ κειμένου πολλαπλών στηλών μπορούν να οριστούν στο CSS3. Το CSS υποστηρίζει μόνο μπλοκ κειμένου.

Πίνακας σύγκρισης CSS και CSS3

Η κύρια σύγκριση μεταξύ CSS και CSS3 φαίνεται στον παρακάτω πίνακα:

| Η βάση σύγκρισης μεταξύ CSS έναντι CSS3 | CSS | CSS3 |
|--|---|--|
| <ul style="list-style-type: none"> • Συμβατότητα | <ul style="list-style-type: none"> • Το CSS1 δεν είναι συμβατό με το CSS3. Ο κύριος στόχος της ήταν να παρέχει διάφορες δυνατότητες μορφοποίησης. Πρόσθεσαν επίσης δυνατότητες τοποθέτησης για κείμενα και αντικείμενα. Αλλά όλα αυτά σταδιακά ενημερώθηκαν σε CSS3. Ως εκ τούτου μπορούμε να πούμε ότι το CSS έχει αυξηθεί σε CSS3. | <ul style="list-style-type: none"> • Το CSS3 είναι συμβατό με το CSS1. Δεν θα κάνει κανέναν κωδικό γραμμένο στο CSS1 ως μη έγκυρο. Κάνει την εμφάνιση και την αίσθηση μιας ιστοσελίδας ακόμη καλύτερη. Φορτώνουν γρηγορότερα και ο χρόνος που απαιτείται για τη δημιουργία μιας σελίδας είναι ακόμη μικρότερος. |

| | | |
|--|---|---|
| <ul style="list-style-type: none"> • Στρογγυλεμένες γωνίες και κλίσεις | <ul style="list-style-type: none"> • Πριν από την κυκλοφορία του CSS3, οι προγραμματιστές συνήθιζαν να σχεδιάζουν εικόνες που μοιάζουν με στρογγυλεμένες γωνίες σε διαφορετικές δομές και κλίσεις φόντου. Η διαδικασία περιελάμβανε τον προγραμματιστή να σχεδιάσει το συγκεκριμένο περίγραμμα, να ανεβάσει αυτό το σχέδιο μέσω του διακομιστή, να τοποθετήσει την εικόνα στην ιστοσελίδα και στο τέλος το CSS έπρεπε να τοποθετήσει σωστά αυτό το περίγραμμα. | <ul style="list-style-type: none"> • Από την στιγμή που κυκλοφόρησε το CSS3, ο προγραμματιστής πρέπει απλώς να γράψει τον κώδικα όπως <code>".roundBorder {border-radius: 10px;}"</code>. Τάδα! Εγινε. Ο χρήστης δεν χρειάζεται να τοποθετήσει τον κωδικό στο διακομιστή και να εκτελέσει τις άλλες δραστηριότητες. Οι ντεγκραντέ μπορούν να ρυθμιστούν χρησιμοποιώντας κώδικα όπως <code>".gradBG {background: linear-gradient (λευκό, μαύρο);}"</code> |
| <ul style="list-style-type: none"> • Εφέ κίνησης και κειμένου | <ul style="list-style-type: none"> • Τα κινούμενα σχέδια σε CSS γράφτηκαν σε JavaScript και JQuery. Το CSS δεν είχε χαρακτηριστικά στο επίπεδο σχεδίασης και τα στοιχεία της σελίδας δεν θα μπορούσαν επίσης να έχουν ειδικά εφέ όπως σκίαση κειμένου, ενότητες κειμένου κ.λπ. | <ul style="list-style-type: none"> • Χρησιμοποιώντας CSS3, ένας προγραμματιστής μπορεί να προσθέσει σκιά κειμένου για να διευκολύνει την ανάγνωση. Μπορούν επίσης να προσθέσουν οπτικά εφέ για να σπάσουν γραμμές και μεγαλύτερες λέξεις, ώστε να ταιριάζουν σωστά μέσα στις στήλες και τα περιτυλίγματα λέξεων. Άλλα χαρακτηριστικά περιλαμβάνουν επίσης μια συνεχή αλλαγή μεγέθους και χρώματος κειμένου. Ο χρόνος αλλαγής μπορεί να οριστεί. Ακόμη και μια ενέργεια όπως το ποντίκι του ποντικιού μπορεί να ρυθμιστεί για την αλλαγή. |

| | | |
|---|---|---|
| <ul style="list-style-type: none"> • Τόπος αγώνων | <ul style="list-style-type: none"> • Το CSS επιτρέπει στον χρήστη να: <ul style="list-style-type: none"> 1) Ορισμός διαφορετικών λιστών για ταξινομημένες λίστες2) Ορισμός διαφορετικών λιστών για μη ταξινομημένες λίστες3) Ορισμός εικόνας για δείκτη στοιχείου λίστας4) Προσθήκη χρωμάτων φόντου στη λίστα και στοιχεία λίστας. Οι διαφορετικοί δείκτες στοιχείων λίστας είναι στυλ λίστας- τύπος. Αυτά μπορούν να οριστούν ως κύκλος, τετράγωνο κ.λπ. | <ul style="list-style-type: none"> • Για να χρησιμοποιήσετε μια λίστα στο CSS3, η ιδιότητα «οθόνη» πρέπει να έχει καθορισμένο στοιχείο λίστας. Το στοιχείο λίστας έχει έναν μετρητή και επηρεάζεται άμεσα από τις ιδιότητες αύξησης μετρητών και επαναφοράς μετρητή. Το CSS3 δεν υποστηρίζει το σύστημα αρίθμησης και επομένως ενδέχεται να το αγνοήσει. Η ιδιότητα εικόνας στυλ λίστας στο CSS3 επιτρέπει τη ρύθμιση μιας εικόνας έναντι του δείκτη στοιχείου λίστας. Μόλις η εικόνα είναι διαθέσιμη, θα ρυθμιστεί ως δείκτης τύπου στυλ λίστας. Έχει επίσης την ιδιότητα θέσης στυλ λίστας που θα καθορίζει τη θέση του πλαισίου σήμανσης σε ένα κύριο πλαίσιο. Μπορεί να οριστεί είτε εντός είτε εκτός του κουτιού. |
| <ul style="list-style-type: none"> • Ψευδο-τάξεις | <ul style="list-style-type: none"> • Οι ψευδο-τάξεις χρησιμοποιούνται για να ορίσουν ειδικά την κατάσταση ενός στοιχείου. • Σύνταξη: • επιλογέας: <code>pseudo-class { property: value; }</code> Παρέχει διαφορετικές ιδιότητες όπως <code>Hover on ()</code>, <code>Simple tooltip hover ()</code>. Η: ψευδοκλάση πρώτου παιδιού αντιστοιχεί στο | <ul style="list-style-type: none"> • Ψευδο-τάξεις στο CSS3 μοιάζουν πολύ με το CSS. Έχουν όμως μερικά επιπλέον χαρακτηριστικά που το καθιστούν ευκολότερο και διάσημο στη χρήση. Σε αυτά περιλαμβάνονται: <ul style="list-style-type: none"> 1): <code>root target</code> του βασικού στοιχείου του εγγράφου. 2): <code>nth-child (n)</code> χρησιμοποιεί |

| | | |
|--|--|---|
| | <p>πρώτο παιδί οποιουδήποτε στοιχείου.</p> | <p>αριθμητικές τιμές στο (n) για να στοχεύσει θυγατρικά στοιχεία σε σχέση με τη θέση τους στο γονέα. Για παράδειγμα, θα μπορούσατε να το χρησιμοποιήσετε για να προσθέσετε εναλλακτικά χρώματα φόντου σε σχόλια ιστολογίου 3): κενά στοιχεία στόχου που δεν έχουν κείμενο ή παιδιά, όπως κενά στοιχεία όπως <p></p></p> |
|--|--|---|

Πίνακας 1 Πίνακας σύγκρισης CSS και CSS3

Συμπέρασμα

Οι παραπάνω διαφορές μεταξύ CSS και CSS3 δείχνουν τον τρόπο με τον οποίο το CSS μετατράπηκε σταδιακά σε CSS3. Οι ομαλές μεταβάσεις, ο καθαρός σχεδιασμός και η ταχύτερη απόδοση έχουν φέρει το CSS στο σημείο που είναι σήμερα. Το CSS μπορεί να χρησιμοποιηθεί για όλη την ανάπτυξη εφαρμογών ιστού. Το CSS3 υποστηρίζει τώρα όλα τα προγράμματα περιήγησης και ως εκ τούτου χρησιμοποιείται παντού. Με την πάροδο του χρόνου, το CSS4 πρόκειται να εισαχθεί σύντομα. Μέχρι τότε το CSS3 είναι διαθέσιμο για όλους τους τρέχοντες χρήστες με τις μικρές βελτιώσεις στο παρόν πλαίσιο. Ως εκ τούτου, το Cascading Style Sheet θα παραμείνει στη βιομηχανία λογισμικού και θα βοηθήσει τους χρήστες να δημιουργήσουν διαδραστικές και τις πιο κομψές εφαρμογές και σελίδες ιστού.

ΚΕΦ. 3. BOOTSTRAP



Εικόνα 10 Bootstrap logo

Εισαγωγή

Η Bootstrap είναι μια δωρεάν και open-source δομή front-end για γρήγορη και ευκολότερη ανάπτυξης ιστοσελίδων. Περιέχει πρότυπα βασισμένα σε HTML και CSS που βοηθούν στη ρύθμιση της εμφάνισης μιας ιστοσελίδας στην τυπογραφία, στις φόρμες, στα κουμπιά, στην πλοήγηση, και σε πολλά άλλα στοιχεία. Επίσης περιέχει προαιρετικές επεκτάσεις javascript. Σε αντίθεση με πολλά παλαιότερα πρότυπα, η Bootstrap ασχολείται μόνο με το front-end κομμάτι της δημιουργίας ιστοσελίδας. Το Bootstrap αναπτύχθηκε από τους Mark Otto και Jacob Thornton[30] στα μέσα του 2010 για λογαριασμό του Twitter ως ένα Framework για την εξασφάλιση μιας ενιαίας αισθητικής στις διάφορες λειτουργίες του. Το Αύγουστο του 2011 εκδόθηκε σαν έργο ανοικτού κώδικα (open source project) και το φεβρουάριο του 2011 έγινε το πιο δημοφιλές έργο στο GitHub[31]. Από τότε έχει γνωρίσει αρκετές ενημερώσεις με το Bootstrap 2 να δημοσιεύεται τον Ιανουάριο του 2012, το Bootstrap 3 να δημοσιεύεται τον Αύγουστο του 2013 και το Bootstrap 4 να δημοσιεύεται τον Ιανουάριο του 2018.

3.1. Χαρακτηριστικά - Εγκατάσταση

Το Bootstrap δεν έχει απόλυτη υποστήριξη για HTML5 και CSS3, αλλά είναι compatible με όλους τους browsers. Από την edition 2.0 προβλέπεται responsive design των websites. Τουτέστιν, η διάταξη των websites προσαρμόζεται δυναμικά, σύμφωνα με τα χαρακτηριστικά της συσκευής στην οποία χρησιμοποιείται (PC, tablets, smartphones). Το Bootstrap είναι open-source framework και βρίσκεται στο GitHub,

με συνέπεια, οι developers να δύνανται να συνεισφέρουν στην ανάπτυξη και εξέλιξή του.

Το Bootstrap διαθέτει διάφορους τρόπους χρήσης ανάλογα με επίπεδο του χρήστη. Μπορεί να χρησιμοποιήσει τα ήδη έτοιμα στοιχεία που παρέχει το bootstrap είτε να χρησιμοποιήσει τα κατάλληλα tags στο κώδικα HTML για την μορφοποίηση της ιστοσελίδας. Επίσης, το Bootstrap είναι full customazable το οποίο δίνει την δυνατότητα στον χρήστη να παραμετροποιήσει και να δημιουργήσει δικιά του έκδοση. Η εγκατάσταση του Bootstrap μπορεί να πραγματοποιηθεί με δύο τρόπους , είτε κατεβάζοντας τις precompiled και minified εκδόσεις των CSS, JavaScript και fonts είτε με την χρήση του Bootstrap CDN.

```
Bootstrap CDN
The folks over at MaxCDN graciously provide CDN support for Bootstrap's CSS and JavaScript. Just use these Bootstrap CDN links.

<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAqycuAPWgG32mVxrk708N3l6Ukb6Wdb270P7tGLTQQtcIUcEytNGlM2nChLaTBHW9MSrVtLGScLJuC76WjdmV9SvRlR7ooP92bWgngJ0IwXwL/Sp" crossorigin="anonymous">

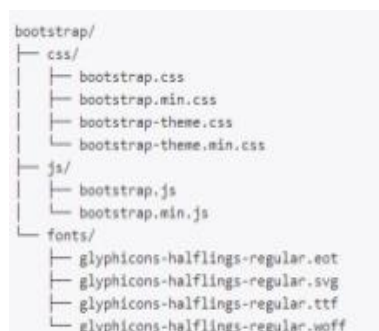
<!-- Optional theme -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap-theme.min.css" integrity="sha384-rhY0NiRvXV4nD0JutInGAsL3Jc7UwjduM95VrLr7ooP92bWgngJ0IwXwL/Sp" crossorigin="anonymous">

<!-- Latest compiled and minified JavaScript -->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-TcS1q1b82qyJSHHJ0MaKkfuWVZkUfNc3A712mCOWp69nGCD8w8N1cP07Txa" crossorigin="anonymous"></script>
```

Εικόνα 11 Bootstrap CDN

3.2. Δομή και Λειτουργία

Το Bootstrap είναι διαθέσιμο για κατέβασμα σε δύο μορφές που αποτελούνται απο φακέλους και αρχεία σε compiled και minified εκδόσεις. Παρακάτω παρουσιάζεται η δομή των αρχείων τα οποία μπορούν να χρησιμοποιηθούν σαν drop-in σε οποιοδήποτε έργο.



Εικόνα 12 Δομή αρχείων Bootstrap

3.2.1. Containers

Ο κώδικας του Bootstrap πρέπει να περικλείεται από μια ετικέτα (tag) τύπου `<div>` που να ανήκει στην κλάση με όνομα `container`.

Δίνονται 2 επιλογές για `containers`:

Χρησιμοποιούμε την κλάση `.container` για ένα `responsive grid` με συγκεκριμένο μέγιστο πλάτος ανεξάρτητο από την ανάλυση της οθόνης:

```
<div class="container">
```

```
...
```

```
</div>
```

ή εναλλακτικά την κλάση `.container-fluid` για πλήρες ανάπτυξη σε όλο το πλάτος ανάλογα με την κάθε οθόνη:

```
<div class="container-fluid">
```

```
...
```

```
</div>
```

3.2.2. Πλέγμα (Grid)

Το bootstrap εξορισμού χρησιμοποιεί πλέγμα (grid) 12 στηλών (12 column grid) με προκαθορισμένες κλάσεις για εύκολη διαχείριση του layout.

Οι κανόνες που χρησιμοποιούμε είναι οι εξής:

- Οι σειρές (rows) πρέπει να είναι μέσα σ' έναν `container` για “πιάνουν” το σωστό χώρο μέσα σ' αυτούς με τα σωστά κενά (padding).
- Χρησιμοποιούμε τις σειρές για να έχουμε οριζόντιες ομάδες (groups) από στήλες (columns).
- Το περιεχόμενο μας θα πρέπει να είναι μέσα στις στήλες και μόνο οι στήλες θα πρέπει να είναι παιδιά (children) των γραμμών (rows).

- Υπάρχουν προκαθορισμένες κλάσεις όπως `.row`, `.col-xs-4` για γρήγορες δημιουργίες grid layout.
- Οι στήλες δημιουργούν τα κενά (πατούρες) με την ιδιότητα απο css - padding. Το padding είναι αρνητικό στην πρώτη και τελευταία στήλη πάντα για να στοιχίζεται σωστά σε σχέση με το περιεχόμενο μέσα στο grid που δηλώνετε απο την κλάση **.row**.
- Οι στήλες του κάθε grid ορίζονται μέσα προκαθορισμένες κλάσεις που αντιπροσωπεύουν και τα μεγέθη - πλάτος που μπορεί να έχει η κάθε στήλη. Έτσι π.χ μια στήλη με πλάτος 3 στηλών την καλούμε : **.col-xs-3**
- Αν στο άθροισμα κάθε γραμμής έχουμε πάνω απο 12 στήλες οι έξτρα στήλες “πέφτουν” απο κάτω στο grid.
- Το pattern **col-device-colnum** αντιπροσωπεύει τις εξής προκαθορισμένες κλάσεις: device για την ανάλυση οθόνης συσκευής που έχει πρόσβαση στη σελίδα μας κάθε φορά π.χ lg->large , xs->extra small κτλ ενώ το **colnum** είναι το μέγεθος (απο 1 έως και 12).

3.2.3. Επιλογές Πλέγματος (Grid Options)

Οι επιλογές του Grid φαίνονται στον ακόλουθο πίνακα:

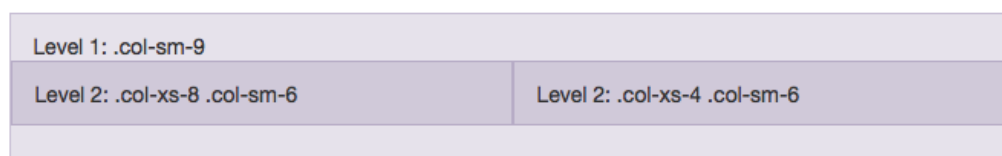
| | Extra small devices Phones (<768px) | Small devices Tablets (≥768px) | Medium devices Desktops (≥992px) | Large devices Desktops (≥1200px) |
|------------------------|--|-----------------------------------|-------------------------------------|-------------------------------------|
| Grid behavior | Horizontal at all times | | | |
| Container width | None (auto) | 750px | 970px | 1170px |
| Class prefix | <code>.col-xs-</code> | <code>.col-sm-</code> | <code>.col-md-</code> | <code>.col-lg-</code> |
| # of columns | 12 | | | |
| Column width | Auto | ~62px | ~81px | ~97px |
| Gutter width | 30px (15px on each side of a column) | | | |
| Nestable | Yes | | | |
| Offsets | Yes | | | |

Εικόνα 13 Επιλογές Πλέγματος στο Bootstrap

3.2.4. Εμφώλευση Στηλών

Για να μπορέσουμε να εμφωλεύσουμε στήλες μέσα σε στήλες πάνω στο default μας πλέγμα χρειάζεται να ξαναορίσουμε μια `div` ετικέτα (tag) που να ανήκει στην κλάση `.row` και κατόπιν να προσθέσουμε το εμφωλευμένο μας πλέγμα (grid) με τις στήλες μας κανονικά.

Έτσι αν π.χ στο πλέγμα μας υπάρχει ήδη μια στήλη με πλάτος 9 στηλών και θέλουμε να προσθέσουμε μέσα ένα άλλο πλέγμα (εμφωλευμένο) το οποίο θα έχει πλάτος 8 και 4 στηλών αντίστοιχα (είναι σημαντικό το άθροισμα των εσωτερικών στηλών να μας δίνει πάντα 12 στο σύνολο, για να μην “σπάει” το πλέγμα). Το σκούρο μώβ του πίνακα είναι το εμφωλευμένο πλέγμα.



Εικόνα 14 Παράδειγμα εμφώνλευσης στηλών

3.2.5. Media Queries

Για την κατασκευή responsive ιστοσελίδων το Bootstrap κάνει χρήση των media queries λαμβάνοντας υπόψιν όλα τα πιθανά περιβάλλοντα πλοήγησης

```
Media queries
We use the following media queries in our Less files to create the key breakpoints in our grid system.

/* Extra small devices (phones, less than 768px) */
/* No media query since this is the default in Bootstrap */

/* Small devices (tablets, 768px and up) */
@media (min-width: @screen-sm-min) { ... }

/* Medium devices (desktops, 992px and up) */
@media (min-width: @screen-md-min) { ... }

/* Large devices (large desktops, 1200px and up) */
@media (min-width: @screen-lg-min) { ... }
```

Εικόνα 15 Bootstrap media queries

Η σύνδεση τους στο HTML κώδικα πραγματοποιείται ως εξής:

```
<link rel="stylesheet" media="mediatype and|not|only (media feature)
href="mystylesheet.css">
```

3.3. Ανάλυση παραμέτρων

Για να χρησιμοποιήσουμε μια από τις πολλές έτοιμες ιδιότητες της Bootstrap αρκεί να προσθέσουμε το όνομα της αντίστοιχης κλάσης στο στοιχείο που θέλουμε να επεξεργαστούμε. Αναλυτικά μερικές από τις σημαντικότερες κλάσεις:

- **.active:** προσθέτει ένα χρωματιστό φόντο σε ένα στοιχείο όταν είναι “ενεργό”. Μπορεί να χρησιμοποιηθεί σε διάφορα στοιχεία όπως πίνακες, υπερσυνδέσμους στην μπάρα περιήγησης, buttons, αντικείμενα λίστας και άλλα.
- **.alert:** δημιουργεί ένα μήνυμα προειδοποίησης
- **.btn:** δημιουργεί ένα απλό κουμπί, με γκρι φόντο και στρογγυλεμένες γωνίες
- **.caption:** προσθέτει υπότιτλο σε μια εικόνα
- **.carousel:** δημιουργεί μια αλληλουχία εικόνων σε στυλ καρουσέλ
- **.checkbox:** χρησιμοποιείται στο στοιχείο γονιό ενός επιλογέα στυλ checkbox
- **.col-*-#:** δημιουργεί τη δομή για ένα στοιχείο ώστε να αλλάζει μέγεθος ανάλογα με το μέγεθος της οθόνης της συσκευής. Όπου * βάζουμε xs, sm, md, lg ή xl ανάλογα με το μέγεθος της οθόνης. Όπου # βάζουμε τον αριθμό στηλών που θέλουμε να καταλαμβάνει το στοιχείο στη σελίδα, από 1 έως 12
- **.collapse:** δηλώνει ένα στοιχείο το οποίο κρύβεται και εμφανίζεται όταν θέλουμε
 - **.container:** χρησιμοποιείται για να δηλώσουμε ένα στοιχείο που θα περιέχει άλλα, το οποίο έχει συγκεκριμένο μέγεθος ανάλογα με την οθόνη και επίσης ίσες αποστάσεις αριστερά και δεξιά από την άκρη της σελίδας
- **.disabled:** απενεργοποιεί ένα στοιχείο
 - **.dropdown:** δημιουργεί μια αναπτυσσόμενη λίστα όπου ο χρήστης μπορεί να επιλέξει μια τιμή από προκαθορισμένες επιλογές
 - **.form-control:** χρησιμοποιείται σε περιοχές φόρμας όπου συμπληρώνει στοιχεία ο χρήστης και ελέγχει το μέγεθος των στοιχείων ανάλογα με το μέγεθος οθόνης
 - **.glyphicon:** δημιουργεί ένα εικονίδιο. Η Bootstrap έχει μια βιβλιοθήκη με 260 εικονίδια
 - **.hidden:** κάνει το στοιχείο αόρατο δίνοντας του την ιδιότητα και τιμή display:none

- **.icon-bar**: χρησιμοποιείται στη μπάρα πλοήγησης για να δημιουργήσει ένα μενού τύπου hamburger (3 οριζόντιες μπάρες)
- **.img-circle**: μετατρέπει μια εικόνα σε κύκλο
- **.img-rounded**: στρογγυλοποιεί τις γωνίες μιας εικόνας
- **.item**: προστίθεται σε κάθε αντικείμενο που ανήκει σε καρουσέλ
- **.jumbotron**: προσθέτει στο στοιχείο ένα γκρι φόντο με στρογγυλεμένες γωνίες και μεγαλώνει τη γραμματοσειρά ώστε να καλέσει την προσοχή του χρήστη στο στοιχείο
- **.lead**: μεγαλώνει τη γραμματοσειρά και το ύψος γραμμής σε μια παράγραφο
- **.modal**: δηλώνει ότι το περιεχόμενο του στοιχείου ανήκει σε modal και στρέφει την προσοχή του χρήστη σε αυτό
- **.navbar**: δημιουργεί μια μπάρα περιήγησης
- **.nav nav-tabs**: δηλώνει ένα μενού στη μπάρα περιήγησης που έχει κουμπιά-κελιά
- **.navbar-brand**: προστίθεται για να δηλώσει το λογότυπο σε μια σελίδα
- **.navbar-btn**: στοιχίζει κατακόρυφα ένα κουμπί μέσα στη μπάρα περιήγησης
- **.navbar-collapse**: κρύβει τη μπάρα περιήγησης και την αντικαθιστά με ένα εικονίδιο hamburger σε μικρότερες συσκευές (πχ κινητά τηλέφωνα)
- **.navbar-fixed-top**: κάνει τη μπάρα περιήγησης να κολλήσει στο πάνω μέρος της οθόνης και να είναι πάντα εμφανής στο χρήστη
- **.navbar-left**: στοιχίζει όλα τα στοιχεία της μπάρας περιήγησης στα αριστερά
- **.popover**: δημιουργεί ένα στοιχείο που εμφανίζεται όταν ο χρήστης πατήσει σε ένα στοιχείο
- **.progress-bar**: δημιουργεί μια μπάρα προόδου
- **.row**: χρησιμοποιείται σε στοιχεία που θα περιέχουν responsive στήλες
- **.table**: προσθέτει βασική στοίχιση και περιθώρια σε πίνακες

ΚΕΦ. 4. CSS Frameworks

Εισαγωγή

Τα CSS Frameworks είναι στην ουσία βιβλιοθήκες. Βοηθούν τον προγραμματιστή να δημιουργήσει πιο εύκολα και πιο γρήγορα βασικά στοιχεία που δομούν την αρχική σελίδα μία ιστοσελίδας(πχ navbars, footer, slider, 3 column layout). Ωστόσο η εξοικονόμηση χρόνου δεν είναι το μοναδικό όφελος που υπάρχει με τη χρήση των CSS Frameworks. Ομάδες με πολλούς προγραμματιστές, ειδικά μεγαλύτερες, μπορούν πραγματικά να εκτιμήσουν τα πρότυπα που παρουσιάζουν τα Frameworks. Αντί για κάθε προγραμματιστή να φέρει τη δική του οπτική σε ένα έργο, τα Frameworks τυποποιούν τις διατάξεις και επιτρέπουν σε ένα προγραμματιστή να διαβάσει εύκολα τον κώδικα του άλλου προγραμματιστή. Αυτό επιτρέπει ένα ομαλότερο κύκλο ανάπτυξης με λιγότερα σφάλματα και ευκολότερη επικοινωνία της ομάδας.

Είναι σημαντικό να τονιστεί ότι δε συνιστάται η μετάβαση σε Frameworks ως αρχάριος. Μερικές φορές οι άνθρωποι θέλουν απλώς να ξεκινήσουν όσο πιο γρήγορα μπορούν, και βλέπουν τα Frameworks ως έναν τρόπο να ξεκινήσουν τους στόχους τους. Παρόλο που αυτό ισχύει για την γρήγορη και σωστή δημιουργία κάποιου έργου, μπορεί να είναι επιβλαβές για τη μακροχρόνια καριέρα του αρχάριου προγραμματιστή, καθώς λείπουν οι βασικές δεξιότητες που μπορούν να εφαρμοστούν γενικά.

4.2 Παραδείγματα CSS Frameworks

Το responsive design είναι απλά html5, css3 και λίγο javascript. Έχουν αναπτυχθεί διάφορα πλαίσια(frameworks) εργασίας για τους προγραμματιστές και τους σχεδιαστές τα οποία διευκολύνουν την ανάπτυξη responsive sites. Παρακάτω περιγράφω κάποια εργαλεία που μπορούν να βοηθήσουν την σχεδίαση:

- **Grid960**

Το 960.gs βασίζεται στην τοποθέτηση όλων των στοιχείων του ιστότοπού σας σε κοντέινερ πλάτους 960 εικονοστοιχείων (έξυπνο κορίτσι) και στη διαίρεση αυτού του κοντέινερ σε στήλες 12, 16 ή 24 ίσου μεγέθους. **Γιατί 960 εικονοστοιχεία;** Επειδή το 960px είναι ένα πλάτος που ταιριάζει σε μεγάλο αριθμό πλατφορμών στις οποίες μπορούμε να περιηγηθούμε στον Ιστό. Επιτρέπει ουσιαστικά σε μια οθόνη πλάτους 1024 εικονοστοιχείων να

προβάλλει τον ιστότοπο με ακρίβεια και χωρίς οριζόντια κύλιση, που αντιστοιχεί στο πλάτος του χρώματος του προγράμματος περιήγησης, των γραμμών κύλισης και ενός τμήματος γεμίσματος για ευκρίνεια. Υπάρχει πάντα ένα περιθώριο 10 εικονοστοιχείων τοποθετημένο στα δεξιά και αριστερά της κύριας στήλης του περιεχομένου, πράγμα που σημαίνει ότι τα μικρότερα προγράμματα περιήγησης θα είναι πάντα σε θέση να διαβάσουν το πιο μακρινό αριστερό περιεχόμενο χωρίς το κείμενο, αλλά στο παράθυρο του προγράμματος περιήγησης.

- **Foundation**

Το Foundation (από την εταιρία ZURB), είναι ένα σχετικά ελαφρύ πλαίσιο εργασίας (front-end framework) για δημιουργία responsive web site. Βασίζεται σε ένα πλέγμα 12 στηλών, των οποίων το πλάτος μπορεί να αλλάξει εύκολα, δημιουργώντας έτσι ότι layout επιθυμούμε. Χρησιμοποιεί την ιδιότητα του CSS “box-sizing: border-box”, ώστε να μην επηρεάζονται τα περιθώρια (borders και padding) των στηλών, και να μπορούμε να σχεδιάζουμε χωρίς περίπλοκα μαθηματικά για τον υπολογισμό των pixels. Το foundation περιλαμβάνει πάρα πολλά στοιχεία και στυλ, για την επιτάχυνση της φάσης κατασκευής, διασφαλίζοντας έτσι την δια λειτουργικότητα του σε όλες σχεδόν τις συσκευές.

Έχει κάποια ασυνήθιστα χαρακτηριστικά από άλλα πλαίσια CSS:

Πλήρες εργαλείο: Είναι τεχνικά λάθος να πούμε ότι το Ίδρυμα είναι ένα πλαίσιο CSS. Ενωώ, είναι, αλλά έχει κατασκευαστεί ως μια μεγάλη και αρθρωτή συλλογή εργαλείων που στοχεύει στην επίλυση σχεδόν όλων των ειδών προβλημάτων front-end.

Ένα από τα μειονεκτήματα είναι:

1. **Μικρή (er) κοινότητα:** Η κοινότητα του Ίδρύματος είναι πολύ μικρότερη από αυτήν του Bootstrap και αν προσπαθείτε κάτι εξωτικό και κολλήσετε, οι πιθανότητες εύρεσης σχετικής βοήθειας είναι χαμηλότερες. Ωστόσο, θα το προσθέσω για όλους τους πρακτικούς σκοπούς. υπάρχει αρκετή κοινότητα εκεί έξω. Είναι απλώς ότι είναι αρκετές τάξεις μεγέθους μικρότερες από αυτές του Bootstrap, οπότε ενδέχεται να μην βρείτε λύσεις αμέσως.

- **Bulma**

Το Bulma είναι ένας σχετικά νεοεισερχόμενος στο πεδίο μάχης των πλαισίων CSS και έχει πάρει ένα όνομα για τον εαυτό της σε σύντομο χρονικό διάστημα. Η ελκυστικότητά του έγκειται σε μια αυστηρή προσέγγιση μόνο για CSS (δεν υπάρχουν στοιχεία JavaScript) και κομψές προεπιλογές, κάτι που πολλοί προγραμματιστές με καλό σχεδιασμό για να αντιμετωπίσουν πρόβλημα όταν εργάζονται με το Bootstrap

Υπάρχουν πολλοί λόγοι για να χρησιμοποιήσετε το Bulma. Ένας από αυτούς είναι το γεγονός ότι είναι αρκετά δημοφιλές. Δεν είναι πιο δημοφιλές από το Bootstrap, ωστόσο από την αρχή το Bulma είχε 30k + αστέρια στο Github.

- **UIkit**

Αυτό που έρχεται στο μυαλό όταν σκεφτόμαστε το [UIkit](#) είναι ο μινιμαλισμός. Ο μινιμαλισμός όχι στα χαρακτηριστικά (στην πραγματικότητα, προσφέρει ίσως τα περισσότερα χαρακτηριστικά όλων των πλαισίων), αλλά στο σχεδιασμό. Ωστόσο υπάρχει ένα σημαντικό μειονέκτημα. Όπως το bootstrap, έτσι και το UIkit λειτουργεί με Javascript ενώ υπάρχει η δυνατότητα χρήσης JQuery για χειρισμό DOM, η χρήση ενός εικονικού πλαισίου DOM όπως η React είναι αδύνατη.

- **Tailwind CSS:**

Είναι ένα εξαιρετικά προσαρμόσιμο, χαμηλού επιπέδου, πλαίσιο πρώτων βοηθημάτων CSS που σας παρέχει όλα τα δομικά στοιχεία που χρειάζεστε για να δημιουργήσετε ξεχωριστά σχέδια χωρίς ενοχλητικά στυλ που πρέπει να αγωνιστείτε για να παρακάμψετε. Σε αντίθεση με άλλα πλαίσια CSS (Bootstrap ή Materialize CSS) δεν συνοδεύεται από προκαθορισμένα στοιχεία. Αντ' αυτού, λειτουργεί σε χαμηλότερο επίπεδο και σας προσφέρει ένα σύνολο βοηθητικών τάξεων CSS. Χρησιμοποιώντας αυτές τις τάξεις, μπορείτε να δημιουργήσετε γρήγορα προσαρμοσμένο σχεδιασμό εύκολα. Το Tailwind CSS σας επιτρέπει να δημιουργήσετε το δικό σας μοναδικό σχέδιο.

Λόγοι για να χρησιμοποιήσετε το Tailwind:

1. Δεν επιβάλλει σχεδιαστικές αποφάσεις που πρέπει να παλέψετε για να αναιρέσετε
2. Προσφέρει την αρχική εφαρμογή ενός προσαρμοσμένου σχεδίου με τη δική του ποιότητα
3. Έρχεται με ένα μενού προκαθορισμένων widget για την κατασκευή του ιστοτόπου σας

- **Materialize CSS**

Είναι ένα πλαίσιο απόκρισης front-end που βασίζεται στον σχεδιασμό υλικού με συλλογές στοιχείων UI με ελάχιστα εφέ στα οποία οι χρήστες μπορούν εύκολα να προσελκύσουν. Το Materialize ανταποκρίνεται πλήρως σε tablet και κινητά. Είναι εύκολο να μάθει, καθώς παρέχεται και εξαιρετική τεκμηρίωση. Αυτό το πλαίσιο έχει μεγάλη υποστήριξη από την κοινότητα και εξαιρετικά θετικά σχόλια. Το Materialize CSS σας επιτρέπει να προσαρμόσετε τις επιλογές σας με ένα εντυπωσιακό σύνολο συλλογών χρωμάτων.

Λόγοι για να χρησιμοποιήσετε το Materialize CSS:

1. Η σελίδα τεκμηρίωσης του Materialize είναι πολύ περιεκτική και πολύ εύκολη στην αρχή.
2. Η σελίδα συστατικών του Materialize περιλαμβάνει κάρτες, κουμπιά, πλοήγηση και πολλά άλλα πρόσθετα χαρακτηριστικά.

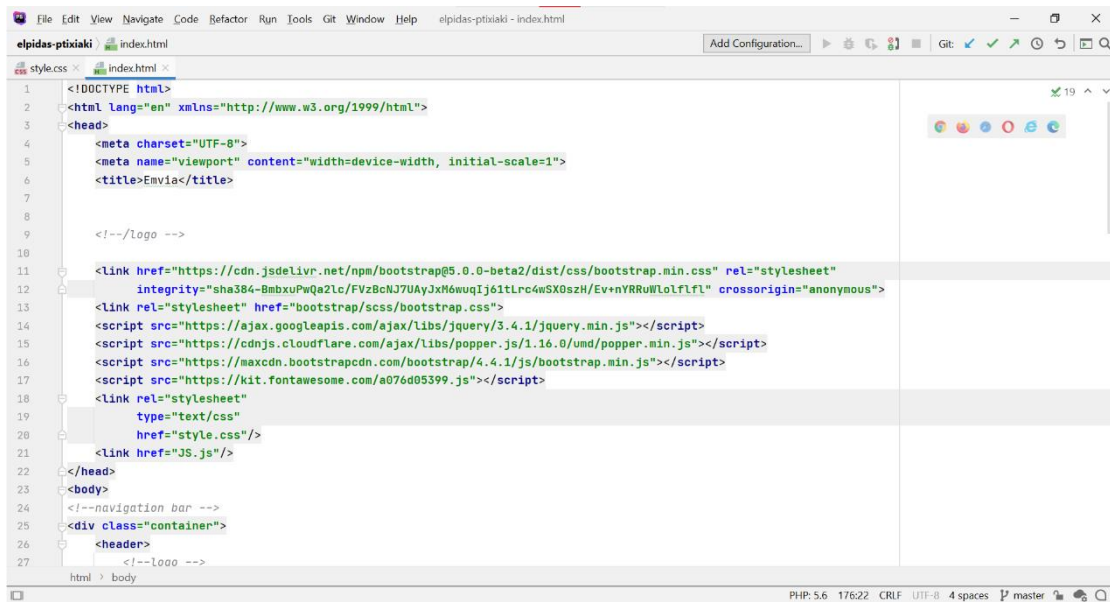
4.3 Οι λόγοι που χρησιμοποίησα το Bootstrap

Μια πρωτοβουλία του Twitter, **το Bootstrap** είναι το καλύτερο πλαίσιο CSS στον κόσμο με μεγάλη κοινοτική υποστήριξη. Αυτό το πλαίσιο είναι ενσωματωμένο σε HTML, SASS και javascript. Προς το παρόν, το Bootstrap 5.0 είναι η τελευταία έκδοση με μεγαλύτερη ανταπόκριση με τάξεις χρησιμότητας και νέα στοιχεία. Απευθύνεται στην απόκριση, την πρώτη διεπαφή για φορητές συσκευές που την καθιστά χρησιμοποιήσιμη για οποιαδήποτε συσκευή και φιλική για προγραμματιστές. Το Bootstrap υποστηρίζει όλα τα σύγχρονα προγράμματα περιήγησης. Το καλύτερο πλεονέκτημα του bootstrap είναι, αυτό το πλαίσιο διαθέτει εξαιρετικά στοιχεία javascript με προσαρμοσμένα αρχεία ή CDN. Είναι το πρώτο framework που προώθησε τη φιλοσοφία του “mobile-first”.

Πιο συγκεκριμένα το Bootstrap προσφέρει πολλά παραδείγματα και μια προκαθορισμένη διάταξη για να ξεκινήσετε. Με το Bootstrap, οι προγραμματιστές μπορούν εύκολα να συρράψουν διαφορετικά στοιχεία και διατάξεις για να δημιουργήσουν ένα νέο και εντυπωσιακό σχεδιασμό σελίδας. Πολλές λεπτομερείς τεκμηρίωση παρέχονται με αυτές τις διατάξεις, έτσι ώστε οι χρήστες να μπορούν να τις κατανοήσουν εύκολα. Το Bootstrap βασίζεται στην άδεια MIT, επομένως η χρήση είναι δωρεάν, η διανομή δωρεάν, ώστε να μπορείτε να αναπτύξετε και να συμβάλλετε και στην κοινότητα. Η σελίδα Github του Bootstrap [GitHub](#) αποτελείται από περισσότερες από 19.000 δεσμεύσεις και 2000 συνεργάτες.

Επίσης είναι ιδανικό για νέους χρήστες της CSS, καθώς μπορεί κάθε χρήστης να ξεκινήσει να χρησιμοποιεί το Bootstrap χωρίς εμπόδιο. Για προγραμματιστές που έχουν λίγη γνώση JavaScript χωρίς να χρειαστεί να γράψουν καμία γραμμή JS. Αλλά και για προγραμματιστές back-end που θέλουν κάποιες αλλαγές στο περιβάλλον εργασίας του χρήστη ακόμη και αν είναι αρχάριος σε HTML και CSS.

ΚΕΦ. 5. Υλοποίηση – Κώδικας



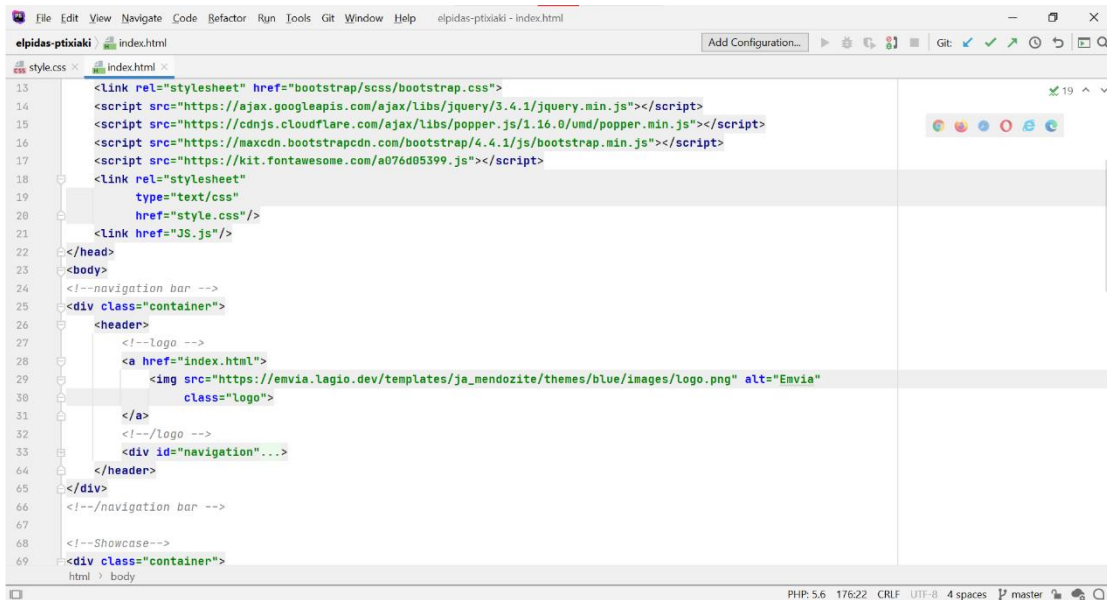
```
1 <!DOCTYPE html>
2 <html lang="en" xmlns="http://www.w3.org/1999/html">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Emvia</title>
7
8
9   <!--/Logo -->
10
11 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta2/dist/css/bootstrap.min.css" rel="stylesheet"
12   integrity="sha384-BmbxuPwQa2lc/FVzBcNJ7UAyJxM6wqIj61tLrc4wSX0szH/Ev+nYRRUWlolflfl" crossorigin="anonymous">
13 <link rel="stylesheet" href="bootstrap/scss/bootstrap.css">
14 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
15 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
16 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>
17 <script src="https://kit.fontawesome.com/a076d05399.js"></script>
18 <link rel="stylesheet"
19   type="text/css"
20   href="style.css"/>
21 <link href="JS.js"/>
22 </head>
23 <body>
24 <!--navigation bar -->
25 <div class="container">
26 <header>
27 <!--Logo -->
```

Εικόνα 5.1: index.html - Βιβλιοθήκες

5.1. Αρχική Σελίδα – index.html

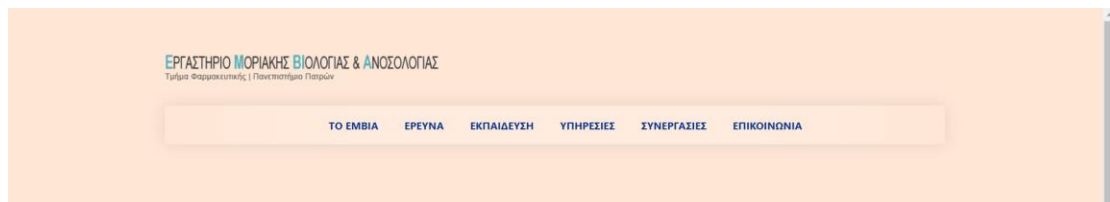
Στις κάτωθι Εικόνες δίνεται σε μορφή κώδικα η υλοποίηση της αρχικής σελίδας. Αξίζει να σημειωθεί η συμπερίληψη των απαραίτητων αρχείων κώδικα CSS.

Με τη βοήθεια της CSS και των κατάλληλων εντολών δημιουργήθηκε το επιθυμητό αποτέλεσμα το οποίο είναι:



```
13 <link rel="stylesheet" href="bootstrap/scss/bootstrap.css">
14 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
15 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
16 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>
17 <script src="https://kit.fontawesome.com/a076d05399.js"></script>
18 <link rel="stylesheet"
19       type="text/css"
20       href="style.css"/>
21 <link href="JS.js"/>
22 </head>
23 <body>
24 <!--navigation bar -->
25 <div class="container">
26 <header>
27 <!--Logo -->
28 <a href="index.html">
29 
31 </a>
32 <!--/Logo -->
33 <div id="navigation"...>
34 </div>
35 </header>
36 </div>
37 <!--/navigation bar -->
38 <!--Showcase-->
39 <div class="container">
40 <div id="showcase" class="grid">
41 <div class="bg-image"></div>
42 <div class="content-wrap">
43 <a href="News.html" class="btn">Περισσότερα...</a>
44 </div>
45 </div>
46 </div>
47 </div>
```

Εικόνα 5.2 Κεντρικό Μενού - Logo



Εικόνα 5.3 Κύριο Μενού και Logo

Στη συνέχεια έφτιαξα τα κύρια κομμάτια που δομούν την ιστοσελίδα.

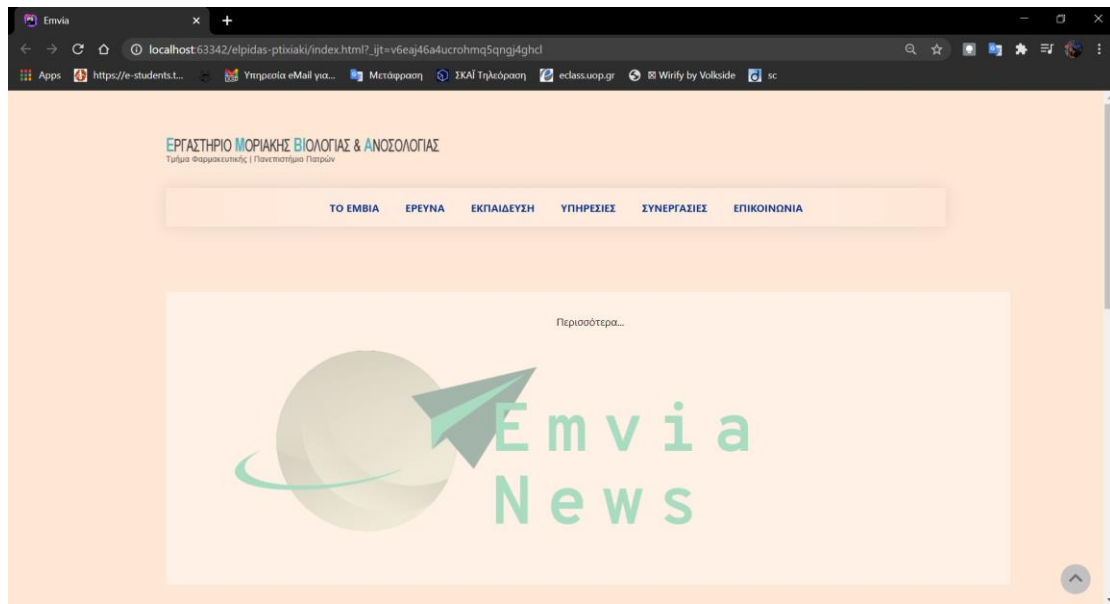
Τα οποία είναι :

1.



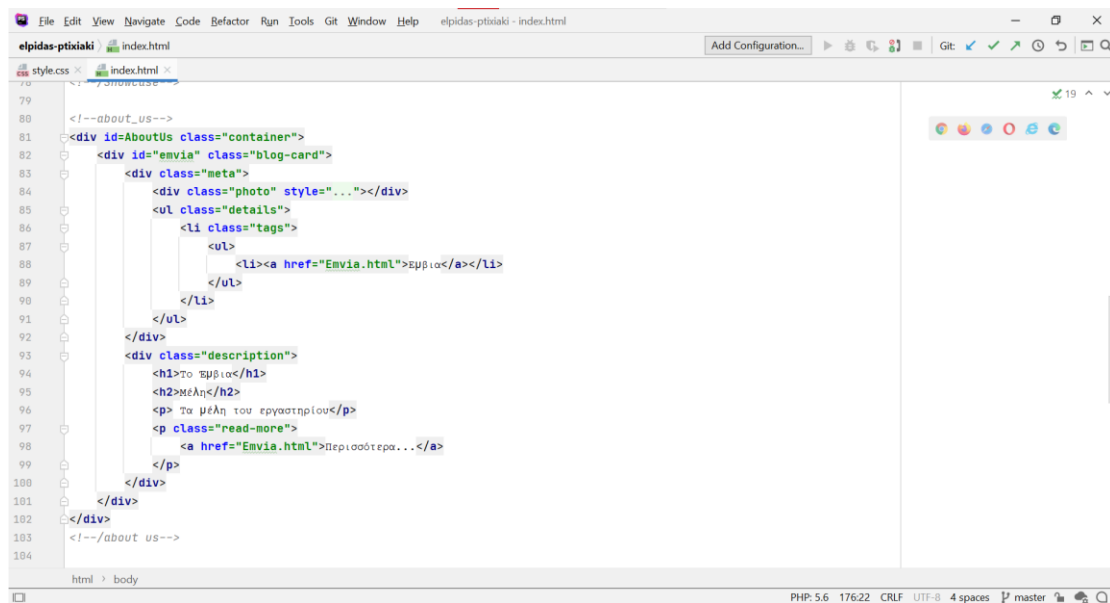
```
67 <!--Showcase-->
68 <div class="container">
69 <div id="showcase" class="grid">
70 <div class="bg-image"></div>
71 <div class="content-wrap">
72 <a href="News.html" class="btn">Περισσότερα...</a>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
```

Εικόνα 5.4 News

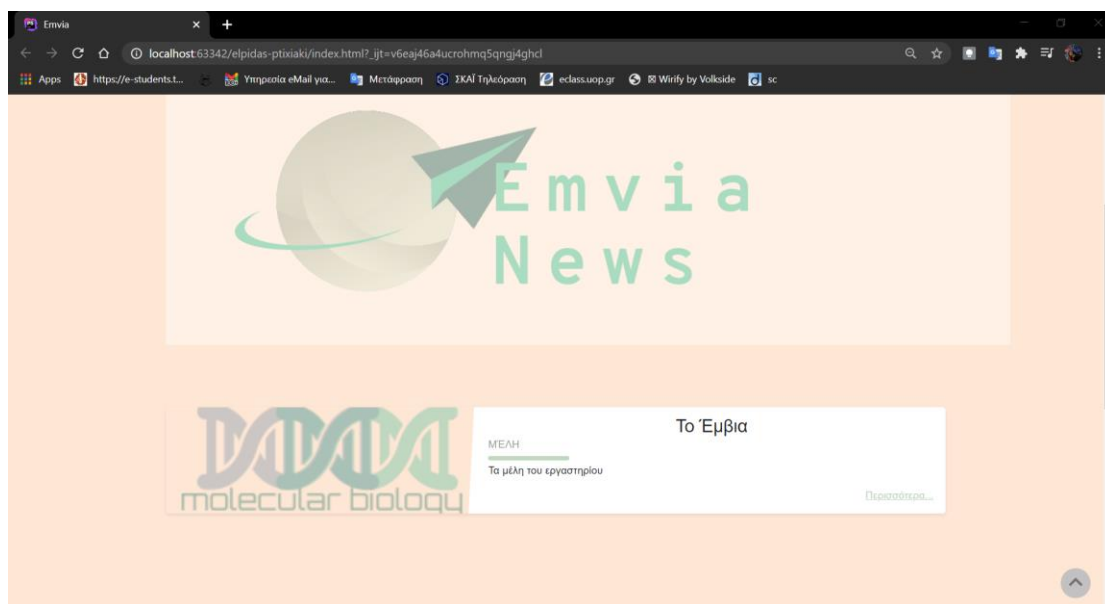


Εικόνα 5.5

2.



Εικόνα 5.6 Το Έμβια



Εικόνα 5.7

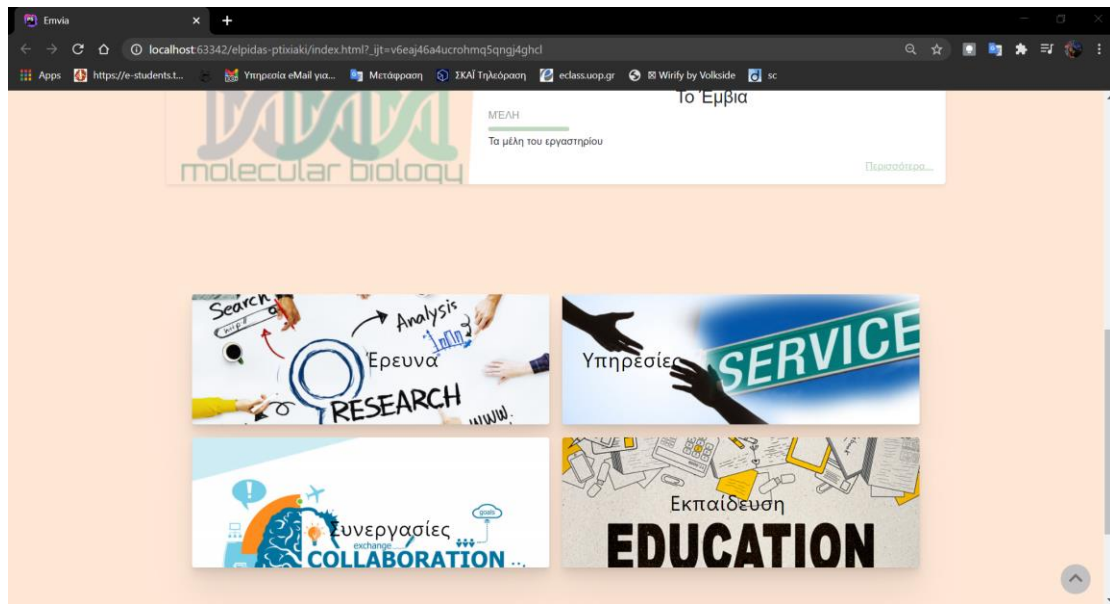
3.

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - index.html
elpidas-ptixiaki index.html
style.css index.html
105 <!--card-->
106 <div class="container">
107 <div class="content">
108 <div class="wrapper">
109 <a href="Research.html">
110 <div id="ereyna" class="box vintage">
111 
112 <h2>Ερευνα</h2>
113 </div>
114 </a>
115 </div>
116 <div class="wrapper">
117 <a href="Services.html">
118 <div id="yphresies" class="box w-content">
119 
120 <div class="frame">
121 <h2>Υπηρεσίες</h2>
122 </div>
123 </div>
124 </a>
125 </div>
126 <div class="wrapper">
127 <a href="Collaborations.html">
128 <div id="synergasies" class="box postcard">
129 
130 <h2>Συνεργασίες</h2>
131 </div>
132 </a>
133 </div>
134 </div>
135 <div class="wrapper">
136 <a href="Education.html">
137 <div id="ekpaideysh" class="box zoom-in">
138 
139 <h2>Εκπαίδευση</h2>
140 </div>
141 </a>
142 </div>
143 </div>
144 </div>
145 </div>
html > body
PHP: 5.6 176:22 CRLF UTF-8 4 spaces master
```

Εικόνα 5.8 Έρευνα & Υπηρεσίες

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - index.html
elpidas-ptixiaki index.html
style.css index.html
120 <div class="frame">
121 <h2>Υπηρεσίες</h2>
122 </div>
123 </div>
124 </a>
125 </div>
126 <div class="wrapper">
127 <a href="Collaborations.html">
128 <div id="synergasies" class="box postcard">
129 
130 <h2>Συνεργασίες</h2>
131 </div>
132 </a>
133 </div>
134 </div>
135 <div class="wrapper">
136 <a href="Education.html">
137 <div id="ekpaideysh" class="box zoom-in">
138 
139 <h2>Εκπαίδευση</h2>
140 </div>
141 </a>
142 </div>
143 </div>
144 </div>
145 </div>
html > body
PHP: 5.6 176:22 CRLF UTF-8 4 spaces master
```

Εικόνα 5.9 Συνεργασίες & Εκπαίδευση



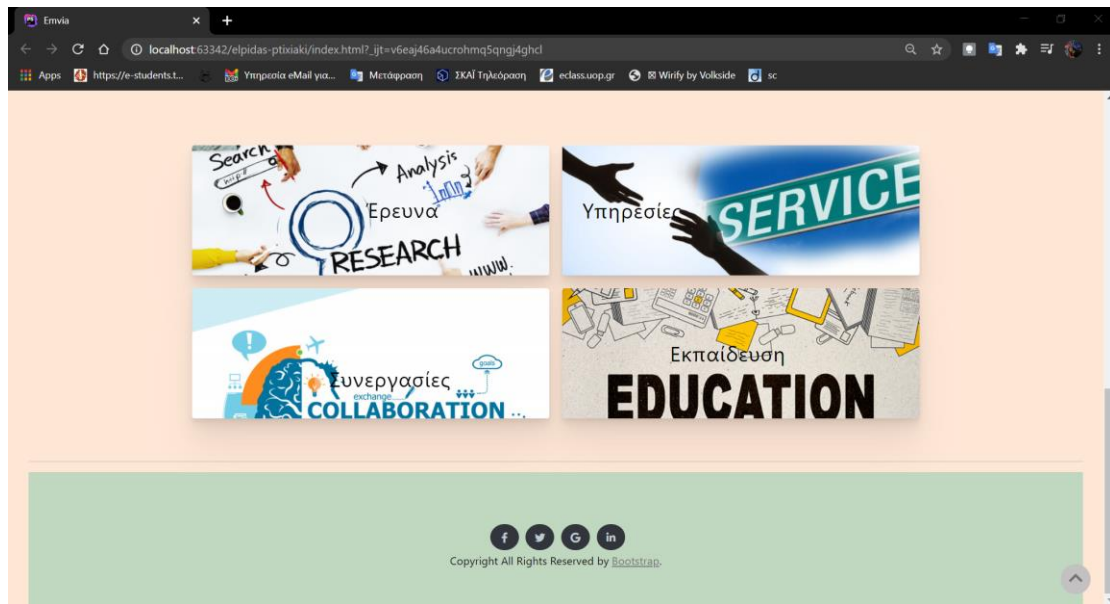
Εικόνα 5.10

4.

```

150 <!-- Site footer -->
151 <footer id="footer" class="site-footer font-small black">
152   <div class="container">
153     <div class="row">
154       <div class="col-md-12 col-sm-6 col-xs-12">
155         <ul class="social-icons">
156           <li><a class="facebook" href="#"><i class="fab fa-facebook-f"></i></a></li>
157           <li><a class="twitter" href="#"><i class="fab fa-twitter"></i></a></li>
158           <li><a class="google" href="#"><i class="fab fa-google"></i></a></li>
159           <li><a class="linkedin" href="#"><i class="fab fa-linkedin-in"></i></a></li>
160         </ul>
161       </div>
162       <div class="col-md-12 col-sm-6 col-xs-12">
163         <p class="copyright-text">Copyright All Rights Reserved by
164           <a href="#">Bootstrap</a>.
165         </p>
166       </div>
167     </div>
168   </div>
169 </div>
170
171 <a class="toTop" href="#">
172   <i class="fas fa-chevron-up" aria-hidden="true"></i>
173 </a>
174 </div>
175 </footer>
  
```

Εικόνα 5.11 Footer



Εικόνα 5.12

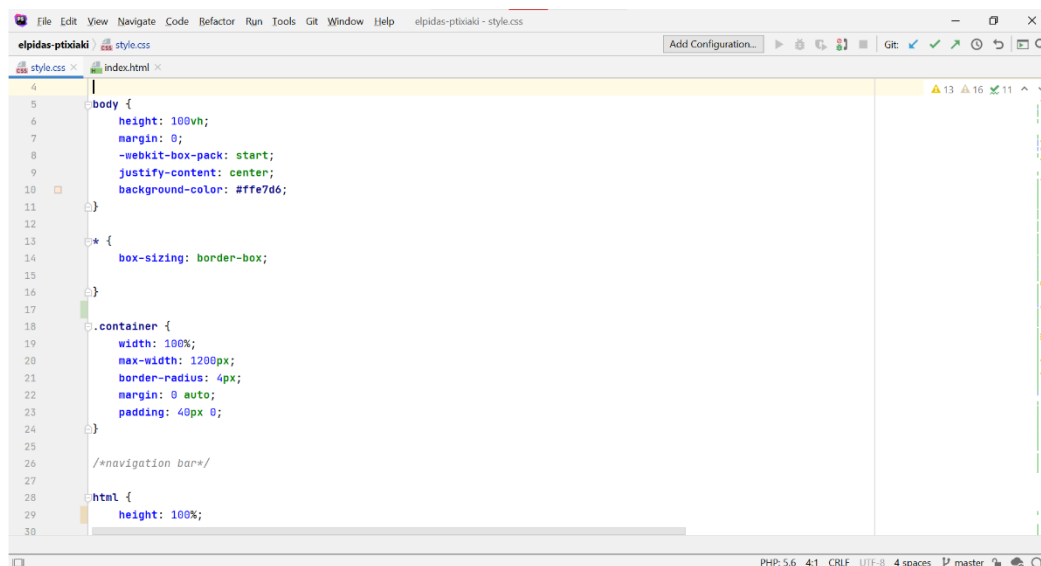
5.2. Φάκελος Αρχείων CSS

Στο φάκελο αυτό, συμπεριλαμβάνονται τα αρχεία bootstrap.css και style.css. Πρόκειται, ουσιαστικά, για τους κανόνες styling – κώδικας CSS, που θα εφαρμοστούν στην ιστοσελίδα, στην ολότητά της.

Το αρχείο bootstrap.css περιλαμβάνει τον κώδικα CSS που παρέχεται από το Bootstrap framework.

Το αρχείο styles.css περιλαμβάνει τον κώδικα CSS που θα δώσει τη χαρακτηριστική εμφάνιση της σελίδας (φόντο, χρώμα, background κλπ.).

Αξίζει ιδιαίτερος να τονιστεί, η υλοποίηση Responsive Design, μέσω της πρόβλεψης διαφορετικών κανόνων CSS, ανάλογα με το εύρος της οθόνης.



```
1 |
2 |
3 |
4 |
5 | body {
6 |     height: 100vh;
7 |     margin: 0;
8 |     -webkit-box-pack: start;
9 |     justify-content: center;
10 |    background-color: #ffe7d6;
11 | }
12 |
13 | * {
14 |     box-sizing: border-box;
15 | }
16 |
17 |
18 | .container {
19 |     width: 100%;
20 |     max-width: 1200px;
21 |     border-radius: 4px;
22 |     margin: 0 auto;
23 |     padding: 40px 0;
24 | }
25 |
26 | /*navigation bar*/
27 |
28 | html {
29 |     height: 100%;
30 | }
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
25
26 /*navigation bar*/
27
28 html {
29     height: 100%;
30 }
31
32
33 .logo {
34     position: relative;
35     left: 0;
36 }
37
38 nav {
39     position: sticky;
40     max-width: 100%;
41     height: 100px;
42     max-height: 100%;
43     mask-image: linear-gradient(
44         90deg,
45         rgba(255, 255, 255, 0) 0%,
46         #ffffff 25%,
47         #ffffff 75%,
48         rgba(255, 255, 255, 0) 100%
49     );
50     margin: 0 auto;
51     padding: 20px 0;

```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
49
50     margin: 0 auto;
51     padding: 20px 0;
52 }
53
54 .main-nav .logo {
55     width: 110px;
56 }
57
58 nav ul {
59     text-align: center;
60     background: linear-gradient(
61         90deg,
62         rgba(255, 255, 255, 0) 0%,
63         rgba(255, 255, 255, 0.2) 25%,
64         rgba(255, 255, 255, 0.2) 75%,
65         rgba(255, 255, 255, 0) 100%
66     );
67     box-shadow: 0 0 25px rgba(0, 0, 0, 0.1),
68     inset 0 0 1px rgba(255, 255, 255, 0.6);
69 }
70
71 nav ul li {
72     display: inline-block;
73 }
74
75 nav ul li a {

```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css index.html
73 }
74
75 nav ul li a {
76     padding: 10px;
77     /*font-family: "Open Sans", bold;*/
78     font-style: initial;
79     text-transform: uppercase;
80     color: rgb(2, 49, 143);
81     font-size: 16px;
82     font-weight: bold;
83     text-decoration: none;
84     display: block;
85 }
86
87 nav ul li a:hover {
88     box-shadow: 0 0 10px rgb(233, 233, 233),
89     inset 0 0 1px rgba(255, 255, 255, 0.6);
90     background: #c0d7c0;
91     color: rgb(0, 0, 0);
92 }
93
94 .dropdown {
95     position: relative;
96     display: inline-block;
97 }
98
99 .dropdown-content {
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css index.html
98
99 .dropdown-content {
100     display: none;
101     position: absolute;
102     background-color: #d7b8a2;
103     min-width: 160px;
104     box-shadow: 0px 8px 16px 0px rgba(0, 0, 0, 0.2);
105     padding: 12px 16px;
106     z-index: 1;
107 }
108
109 .dropdown:hover .dropdown-content {
110     display: block;
111 }
112
113 /*Showcase*/
114
115 #showcase {
116     min-height: 450px;
117     color: #fff;
118     text-align: center;
119 }
120
121 #showcase .bg-image {
122     position: absolute;
123     background: #ffffff url("logo_transparent.png");
124     background-position: center;
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
125 background-repeat: no-repeat;
126 width: 1300px;
127 max-width: 100%;
128 height: 450px;
129 max-height: 100%;
130 z-index: -1;
131 opacity: 0.4;
132 }
133
134 #showcase .content-wrap,
135 #section-b {
136 padding: 2em 1em 1em;
137 }
138
139
140 .btn {
141 position: absolute;
142 max-width: 10%;
143 max-height: 100%;
144 margin-bottom: 0;
145 left: 50%;
146 display: grid;
147 color: #000000;
148 text-decoration: none;
149 padding: 0em 0em;
150
151 }
PHP: 5.6 4:1 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
152
153 .btn:hover {
154 background: #c0d7c0;
155 color: #000000;
156 }
157
158 /*Showcase*/
159 /*!*card!**/
160
161 .header {
162 font-size: 24px;
163 text-align: center;
164 }
165
166 .content {
167 display: flex;
168 flex-wrap: wrap;
169 margin-top: 60px;
170 padding: 0 30px;
171 }
172
173 .wrapper {
174 width: 50%;
175 height: 100%;
176 padding: 10px;
177 }
178
PHP: 5.6 4:1 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
179 .name {
180     position: relative;
181     font-size: 10px;
182     display: inline-block;
183 }
184
185 .name:after {
186     content: "";
187     position: absolute;
188     width: calc(100% + 10px);
189     height: 1px;
190     background-color: #000;
191     bottom: -4px;
192     left: 0;
193 }
194
195 .box {
196     position: relative;
197     max-width: 100%;
198     max-height: 200px;
199     border-radius: 4px;
200     overflow: hidden;
201     box-shadow: 0 1.4px 1.7px rgba(0, 0, 0, 0.017), 0 3.3px 4px rgba(0, 0, 0, 0.024), 0 6.3px 7.5px rgba(0, 0, 0, 0.03), 0 11.2px 13.4px rgba(0, 0, 0, 0.037);
202 }
203
204 .box .hide {
205     opacity: 0;
}
PHP: 5.6 4.1 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
230     right: 20px;
231 }
232
233 .box p {
234     bottom: 0;
235     font-size: 14px;
236     letter-spacing: 1px;
237 }
238
239 .box:hover {
240     transition: all 0.3s ease-in-out;
241 }
242
243 .box:hover:before {
244     transition: all 0.3s ease-in-out;
245 }
246
247 .box img {
248     position: relative;
249     width: 100%;
250     height: 90%;
251     z-index: 1;
252     transition: all 0.3s ease-in-out;
253 }
254
255 .box img:hover {
256     transition: all 0.3s ease-in-out;
}
.box.frame
PHP: 5.6 21028 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css index.html
203
204 .box .hide {
205     opacity: 0;
206 }
207
208 .box .frame {
209     position: absolute;
210     border: 1px solid #fff;
211     z-index: 2;
212     top: 50%;
213     left: 50%;
214     transform: translate(-50%, -50%);
215 }
216
217 .box h2, .box p {
218     position: absolute;
219     color: #000000;
220     z-index: 2;
221     width: 100%;
222     transition: opacity 0.2s, transform 0.3s;
223 }
224
225 .box h2 {
226     font-weight: 400;
227     margin-bottom: 0;
228     letter-spacing: 1px;
229     font-size: xx-large;
}
.box.frame
PHP: 5.6 210:28 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css index.html
257
258
259 .box img:after {
260     content: "";
261     position: absolute;
262     background-color: rgba(0, 0, 0, 0.6);
263     width: 100%;
264     height: 100%;
265     top: 0;
266     left: 0;
267     opacity: 0;
268 }
269
270 .box img:hover {
271     transition: all 0.3s ease-in-out;
272 }
273
274 .zoom-in h2 {
275     top: 50%;
276     transform: translatey(-50%);
277     text-align: center;
278     margin: 0;
279     font-size: xx-large;
280 }
281
282 .zoom-in p {
283     text-align: center;
}
.box.frame
PHP: 5.6 210:28 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css index.html
284
285     top: calc(50% + 40px);
286     transition: all 0.3s ease;
287     transform: scale(1.8);
288     opacity: 0;
289 }
290
291 .zoom-in:hover img {
292     transform: scale(1.1);
293     filter: grayscale(100%);
294 }
295
296 .zoom-in:hover p {
297     transform: scale(1);
298     transition: all 0.3s ease;
299     opacity: 1;
300 }
301
302 .w-content .frame {
303     width: calc(100% - 100px);
304     height: calc(100% - 100px);
305     top: 50%;
306     left: 50%;
307     transform: translate(-50%, -50%);
308     border-color: transparent;
309 }
310
311 .w-content h2 {
}
.box.frame
PHP: 5.6 210:28 CRLF UTF-8 4 spaces master
```



```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
386 width: 100%;
387 height: 100%;
388 background: #008E6B;
389 z-index: 1;
390 left: 0;
391 opacity: 0;
392 transition: all 0.3s ease-in-out;
393 }
394
395 .blury-card img {
396 z-index: 2;
397 }
398
399 .blury-card:hover .frame {
400 width: calc(100% - 120px);
401 height: calc(100% - 120px);
402 opacity: 1;
403 }
404
405 .blury-card:hover img {
406 opacity: 0.7;
407 }
408
409 .blury-card:hover:before {
410 opacity: 1;
411 transition: all 0.3s ease-in-out;
412 }
.box.frame
PHP: 5.6 210:28 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
413
414 .vintage h2 {
415 top: 10%;
416 left: 9%;
417 transform: translate3d(0, 60px, 0);
418 text-align: center;
419 font-size: xx-large;
420 }
421
422 .vintage p {
423 opacity: 0;
424 bottom: 0;
425 transform: translate3d(0, -10px, 0);
426
427 letter-spacing: 1px;
428 text-align: center;
429 }
430
431
432 .vintage:before {
433 content: "";
434 position: absolute;
435 width: 100%;
436 height: 100%;
437 background: linear-gradient(to bottom, rgba(72, 76, 97, 0) 0%, rgba(72, 76, 97, 0.8) 75%);
438 z-index: 2;
439 bottom: -100%;
440 }
.box.frame
PHP: 5.6 210:28 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
437 background: linear-gradient(to bottom, rgba(72, 76, 97, 0) 0%, rgba(72, 76, 97, 0.8) 75%);
438 z-index: 2;
439 bottom: -100%;
440 left: 0;
441 }
442
443 .vintage:hover:before {
444 bottom: 0;
445 }
446
447 .vintage:hover h2 {
448 bottom: 40px;
449 transform: translate3d(0, 20px, 0);
450 }
451
452 .vintage:hover p {
453 opacity: 1;
454 transform: translate3d(0, -30px, 0);
455 }
456
457 .zoom-out .frame {
458 width: calc(100% - 100px);
459 height: calc(100% - 100px);
460 top: 50%;
461 left: 50%;
462 transform: translate(-50%, -50%);
463 width: calc(100% - 40px);
464 }
.box.frame
PHP: 5.6 210:28 CRLF UTF-8 4 spaces master
```



```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
614 background: #fff;
615 position: relative;
616 z-index: 1;
617 }
618
619 .blog-card .description h1,
620 .blog-card .description h2 {
621 font-family: Poppins, sans-serif;
622 }
623
624 .blog-card .description h1 {
625 text-align: center;
626 line-height: 1;
627 margin: 0;
628 font-size: 1.7rem;
629 }
630
631 .blog-card .description h2 {
632 text-align: left;
633 font-size: 1rem;
634 font-weight: 300;
635 text-transform: uppercase;
636 color: #a2a2a2;
637 margin-top: 5px;
638 }
639
640 .blog-card .description .read-more {
        .blog-card .description .read-more a:after
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
641 text-align: right;
642 }
643
644 .blog-card .description .read-more a {
645 color: #c0d7c0;
646 display: inline-block;
647 position: relative;
648 }
649
650 .blog-card .description .read-more a:after {
651 font-family: FontAwesome;
652 margin-left: -10px;
653 opacity: 0;
654 vertical-align: middle;
655 transition: margin 0.3s, opacity 0.3s;
656 }
657
658 .blog-card .description .read-more a:hover:after {
659 margin-left: 5px;
660 opacity: 1;
661 }
662
663 .blog-card p {
664 position: relative;
665 margin: 1rem 0 0;
666 }
667
        .blog-card .description .read-more a:after
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
668 .blog-card p:first-of-type {
669 margin-top: 1.25rem;
670 }
671
672 .blog-card p:first-of-type:before {
673 content: "";
674 position: absolute;
675 height: 7px;
676 background: #c0d7c0;
677 width: 125px;
678 top: -0.75rem;
679 border-radius: 3px;
680 }
681
682 .blog-card:hover .details {
683 left: 0%;
684 }
685
686 @media (min-width: 640px) {
687 .blog-card {
688 flex-direction: row;
689 max-width: 100%;
690 height: 100%;
691 }
692
693 .blog-card .meta {
694 flex-basis: 40%;
        .blog-card .description .read-more a:after
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
695 height: auto;
696 }
697
698 .blog-card .description {
699 flex-basis: 60%;
700 }
701
702 .blog-card .description:before {
703 transform: skew(-3deg);
704 content: "";
705 background: #fff;
706 width: 100%;
707 position: absolute;
708 left: -10px;
709 top: 0;
710 bottom: 0;
711 z-index: -1;
712 }
713
714 .blog-card.alt {
715 flex-direction: row-reverse;
716 }
717
718 .blog-card.alt .description:before {
719 left: inherit;
720 right: -10px;
721 transform: skew(3deg);
}
.blog-card .description .read-more a:after
PHP: 5.6 65045 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
722 }
723
724 .blog-card.alt .details {
725 padding-left: 25px;
726 }
727 }
728
729 /*/news*/
730 /*Footer Elements */
731
732 .toTop {
733 top: auto;
734 bottom: 20px;
735 right: 20px;
736 left: auto;
737 position: fixed;
738 font-size: 24px;
739 height: 46px;
740 width: 46px;
741 line-height: 40px;
742 font-weight: bold;
743 color: #000000;
744 background-color: #c3c1c1;
745 border-radius: 50px;
746 text-align: center;
747 }
}
.blog-card .description .read-more a:after
PHP: 5.6 65045 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
749 .a.toTop {
750 text-decoration: none;
751 }
752
753 .a.toTop:hover {
754 color: #c0d7c0;
755 background-color: #c3c1c1;
756 }
757
758 .site-footer {
759 position: relative;
760 bottom: 0;
761 width: 100%;
762 text-align: center;
763 background-color: #c0d7c0;
764 padding: 40px 0 20px;
765 font-size: 17px;
766 line-height: 24px;
767 color: #000000;
768 }
769
770 .site-footer hr {
771 border-top-color: #bbb;
772 opacity: 0.5;
773 }
774
775 .site-footer hr.small {
}
.blog-card .description .read-more a:after
PHP: 5.6 65045 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
776 margin: 20px 0;
777
778
779 .site-footer h6 {
780 color: #fff;
781 font-size: 16px;
782 text-transform: uppercase;
783 margin-top: 5px;
784 letter-spacing: 2px;
785 }
786
787 .site-footer a {
788 color: #737373;
789 }
790
791 .site-footer a:hover {
792 color: #3366cc;
793 text-decoration: none;
794 }
795
796 .footer-links {
797 padding-left: 0;
798 list-style: none;
799 }
800
801 .footer-links li {
802 display: block;
.blog-card .description .read-more a.after
PHP: 5.6 65045 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptixiaki - style.css
elpidas-ptixiaki style.css
style.css x index.html x
803
804
805 .footer-links a {
806 color: #737373;
807 }
808
809 .footer-links a:active,
810 .footer-links a:focus,
811 .footer-links a:hover {
812 color: #3366cc;
813 text-decoration: none;
814 }
815
816 .footer-links.inline li {
817 display: inline-block;
818 }
819
820 .site-footer .social-icons {
821 text-align: center;
822 }
823
824 .site-footer .social-icons a {
825 height: 40px;
826 line-height: 40px;
827 margin-left: 6px;
828 margin-right: 0;
829 border-radius: 100%;
.blog-card .description .read-more a.after
PHP: 5.6 65045 CRLF UTF-8 4 spaces master
```

```
File Edit View Navigate Code Refactor Run Tools Git Window Help elpidas-ptisiaki - style.css
elpidas-ptisiaki style.css
style.css x index.html x Add Configuration...
830 background-color: #33353d;
831 }
832
833 .copyright-text {
834     margin: 0;
835 }
836
837 @media (max-width: 991px) {
838     .site-footer [class="col-"] {
839         margin-bottom: 30px;
840     }
841 }
842
843 @media (max-width: 800px) {
844     .site-footer {
845         padding-bottom: 0;
846     }
847
848     .site-footer .copyright-text,
849     .site-footer .social-icons {
850         text-align: center;
851     }
852 }
853
854 .social-icons {
855     padding-left: 0;
856     margin-bottom: 0;
857 }
858
859 .blog-card .description .read-more a.after
```

PHP: 5.6 65045 CRLF UTF-8 4 spaces master

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] iefimerida.gr Online Εφημερίδα, 2013, <http://www.iefimerida.gr/news/> αυτή-είναι-η-πρώτη-ιστοσελίδα-που-κατασκευάστηκε-στην-ιστορία-εικόνα (η πρώτη ιστοσελίδα που κατασκευάστηκε στην Ιστορία)
- [2] <http://info.cern.ch/hypertext/WWW/TheProject.html>
- [3]Modern PHP: New Features and Good Practices 1st Edition, Josh Lockhart, O'Reilly Media; 1 edition (March 1, 2015)
- [4]URL (Uniform Resource Locator). Ανακτήθηκε από <https://tools.ietf.org/html/rfc3305>[6]Search Engine Optimization, Wikipedia, https://en.wikipedia.org/wiki/Search_engine_optimization
- [5]World Wide Web Consortium, 2015, <http://www.w3.org/standards/webdesign>[3]Modern PHP: New Features and Good Practices 1st Edition, Josh Lockhart, O'Reilly Media; 1 edition (March 1, 2015)
- [6]Search Engine Optimization, Wikipedia, https://en.wikipedia.org/wiki/Search_engine_optimization
- [7]Apache HTTP Server, 2015, <http://httpd.apache.org/>
- [8]Content Management System, Wikipedia, https://en.wikipedia.org/wiki/Content_management_system
- [9]Responsive Web Design <http://www.artabout.gr/blog/responsive-web-design> , <http://alistapart.com/article/responsive-web-design>
- [10]Tim Berner-Lee, Wikipedia, https://el.wikipedia.org/wiki/Τιμ_Μπέρνερς_Λι
- [11]Internet Engineering Task Force, Τακτική Δύναμη Μηχανικών Internet, <https://el.wikipedia.org/wiki/IETF>
- [12]"[rfc:1866 Request for Comments: 1866]". tools.ietf.org, November 1995.
- [13]Raggett, Dave (1998). Raggett on HTML 4. [14]"IETF HTML WG". September 12, 1996 .
- [14]"IETF HTML WG". September 12, 1996 .
- [15]"HTML 4.0 Specification". World Wide Web Consortium. December 18, 1997.
- [16]"HTML 4 - 4 Conformance: requirements and recommendations". W3C Superseded Recommendation.
- [17]"HTML 4.0 Specification". World Wide Web Consortium. April 24, 1998.
- [18]"HTML 4.01 Specification". World Wide Web Consortium. December 24, 1999[19]"HTML 4.01 Errata". May 12, 2001.
- [19]"HTML 4.01 Errata". May 12, 2001.
- [20]ISO (2000). "ISO/IEC 15445:2000 – Information technology – Document description and processing languages – HyperText Markup Language (HTML)

- [21]ISO (2000). "ISO/IEC 15445:2000 – Information technology – Document description and processing languages – HyperText Markup Language (HTML)".
- [22]<https://el.fondoperlaterra.org/html-vs-html5-472#menu-1>
- [23]<https://en.wikipedia.org/wiki/CSS>
- [24]"Επισκόπηση HTML". [Www.tutorialspoint.com](http://www.tutorialspoint.com)
- [25]CSS
http://nefeli.lib.teicrete.gr/browse/stef/epp/2010/PerrakiKonstantina,KapetanakiErato/attacheddocument-1291719192-666797-13436/Perraki_Kapetanaki2010.pdf[25]L. Sikos (2011), Web Standards: Mastering HTML5, CSS3, and XML, Apress
- [26]L. Sikos (2011), Web Standards: Mastering HTML5, CSS3, and XML, Apress
- [27]CSS3 Transitions http://www.w3schools.com/css3/css3_transitions.asp
- [28]<https://eyewated.com/%CE%B7-%CE%B4%CE%B9%CE%B1%CF%86%CE%BF%CF%81%CE%AC-%CE%BC%CE%B5%CF%84%CE%B1%CE%BE%CF%8D-css2-%CE%BA%CE%B1%CE%B9-css3/>[28]<https://www.educba.com/css-vs-css3/>
- [29] <https://www.educba.com/css-vs-css3/>
- [30]. Mark Otto και Jacob Thornton. Ανακτήθηκε από <http://getbootstrap.com/>
- [31]"About - GitHub". Ανακτήθηκε από <https://github.com/about>