



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Δημιουργία και διαχείριση του δικτυακού
ιστότοπου του Ερευνητικού Εργαστηρίου
Συστημάτων Υπολογιστών, Δικτύων και
Ασφάλειας (Σ.Υ.Α.Δ.)»

Του φοιτητή
Εξηντάρη Ορέστη
Αρ. Μητρώου: 08/3368

Επιβλέπων:
Περικλής Χατζημίσιος
Καθηγητής

Ημερομηνία: 11/06/21

Τίτλος Π.Ε.: Δημιουργία και διαχείριση του δικτυακού ιστότοπου του Ερευνητικού
Εργαστηρίου Συστημάτων Υπολογιστών, Δικτύων και Ασφάλειας (Σ.Υ.Α.Δ.)

Κωδικός Δ.Ε.: 19064

Ονοματεπώνυμο φοιτητή: Εξηντάρης Ορέστης

Ονοματεπώνυμο εισηγητή: Περικλής Χατζημίσιος

Ημερομηνία ανάληψης Δ.Ε. 10/2019

Ημερομηνία περάτωσης Δ.Ε. 06/2021

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Εξηντάρη Ορέστη που την εκπόνησε/αν. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

ΠΡΟΛΟΓΟΣ

Η Επιλογή αυτής της εργασίας έγινε με σκεπτικό την εξοικείωση μου με πρακτικές Web Development , βελτίωση των ικανοτήτων μου αναφορικά με Γλώσσες Προγραμματισμού όπως PHP/JavaScript, υπό το πλαίσιο της εξ 'ολοκλήρου δημιουργίας ενός ιστότοπου, από το επίπεδο hosting , σχεδίασης και κατασκευής της σελίδας, αξιοποιώντας σύγχρονα εργαλεία και δυνατότητες των CMS , αλλά και διαχείριση και βελτιστοποίηση του εν λόγω ιστότοπου ως Administrator.

ΠΕΡΙΛΗΨΗ

Το θέμα της πτυχιακής εργασίας είναι η κατασκευή ενός ιστότοπου που θα φιλοξενήσει την διαδικτυακή παρουσία ενός Ερευνητικού εργαστηρίου. Το αντικείμενο έρευνας και οι δραστηριότητες του εν λόγω εργαστηρίου είναι αόριστα ορισμένες, με Placeholder κείμενα και πρόσωπα. Ο στόχος του ιστότοπου ήταν σε πρώτο επίπεδο να παρέχει την βασική υποδομή που μπορεί να βρεθεί σε οποιαδήποτε αντίστοιχη σελίδα Ερευνητικής οντότητας εσωτερικού η εξωτερικού, όπως προγραμματισμός εκδηλώσεων, συνόψεις και χώρο παρουσίασης της δραστηριότητας του εργαστηρίου, προσωπικές σελίδες προσωπικού κ.α. Κατόπιν, να προστεθεί λειτουργικότητα και ευχρηστία στην σελίδα με την χρήση πρακτικών ορθής σχεδίασης για λόγους βελτιστοποίησης Search Engine Optimization (SEO), πρακτικές προσβασιμότητας για διευκόλυνση χρηστών διαφορετικών ειδικών ομάδων, και κατασκευή features για την διευκόλυνση του διαχειριστή/μελών του εργαστηρίου. Ο ιστότοπος που δημιουργήθηκε έχει την απαραίτητη δομή για να ικανοποιήσει τις ανάγκες ενός οργανισμού για την διαφήμιση και προβολή του έργου του και των ικανοτήτων του ιδίου και των εμπλεκόμενων μελών καθώς και κάποια εργαλεία που εκμεταλλεύονται τεχνολογίες Representational State Transfer (REST) για να παρέχουν διευκολύνσεις σε χρήστες και στους διαχειριστές / μέλη του ιστότοπου.

«Development and Management of the Website for the Computer Systems, Networks and Security Research Lab (CSSN)»

«Orestis Exintaris»

ABSTRACT

The subject of this B.Sc. thesis is designing and developing a website to host the online presence of a Research Lab, or a relevant organization. For this thesis the specific Research object of the Lab/Organization is loosely defined, thus, the information on the website such as the About section or the staff Information are currently Placeholder. The aim of the current thesis was to construct a space that offers the same capabilities of any other Research Lab page, such as event planning, announcement boards, detailing and highlighting activities and achievements of the lab, Individual Staff member pages etc. Additionally, to add functionality and useability to the website. This was attempted by using practices to improve optimization (Search Engine Optimization - SEO) of the page and accessibility of different user groups. These practices include using proper markup, descriptive hyperlink usage and naming, selecting appropriate color schemes and more. Finally, Representational State Transfer (REST) technologies were used to extend the website's capabilities, offering some tools to enable the site Administrator and/or authorized staff to be more efficient in their tasks. The WordPress REST API was used as well as a custom-made PHP API to explore the possibilities of their positive impact in website design.

Πίνακας Εικόνων

ΕΙΚΟΝΑ 1. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΕΝΟΣ ΣΤΑΤΙΚΟΥ ΙΣΤΟΤΟΠΟΥ.....	8
ΕΙΚΟΝΑ 2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΕΝΟΣ ΔΥΝΑΜΙΚΟΥ ΙΣΤΟΤΟΠΟΥ.....	9
ΕΙΚΟΝΑ 3. ΚΑΤΑΝΕΜΗΜΕΝΑ ΚΑΙ ΚΕΝΤΡΙΚΑ ΣΥΣΤΗΜΑΤΑ VERSION CONTROL	13
ΕΙΚΟΝΑ 4. ΣΥΓΚΡΙΣΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ JOOMLA!, DRUPAL ΚΑΙ WORDPRESS.....	14
ΕΙΚΟΝΑ 5. ΚΑΤΕΒΑΖΟΝΤΑΣ ΤΟ WORDPRESS.....	17
ΕΙΚΟΝΑ 6. ΕΠΙΛΟΓΗ ΓΛΩΣΣΑΣ ΚΑΤΑ ΤΗΝ ΕΓΚΑΤΑΣΤΑΣΗ	18
ΕΙΚΟΝΑ 7. ΣΤΟΙΧΕΙΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΚΑΤΑ ΤΗΝ ΕΓΚΑΤΑΣΤΑΣΗ	19
ΕΙΚΟΝΑ 8. ΕΠΙΒΕΒΑΙΩΣΗ ΣΤΟΙΧΕΙΩΝ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	19
ΕΙΚΟΝΑ 9. ΣΤΟΙΧΕΙΑ ΔΙΑΧΕΙΡΙΣΤΗ ΚΑΤΑ ΤΗΝ ΕΓΚΑΤΑΣΤΑΣΗ	20
ΕΙΚΟΝΑ 10. Η ΘΟΝΗ ΕΙΣΟΔΟΥ ΓΙΑ ΤΟΝ ΠΙΝΑΚΑ ΕΛΕΓΧΟΥ ΔΙΑΧΕΙΡΙΣΤΗ	21
ΕΙΚΟΝΑ 11. Ο ΠΙΝΑΚΑΣ ΕΛΕΓΧΟΥ WORDPRESS	22
ΕΙΚΟΝΑ 12. ΠΡΩΤΗ ΣΕΛΙΔΑ	25
ΕΙΚΟΝΑ 13. ΕΙΣΑΓΩΓΗ ΠΡΩΤΗΣ ΣΕΛΙΔΑΣ	25
ΕΙΚΟΝΑ 14. ΚΑΡΤΕΛΕΣ ΠΡΟΣΩΠΙΚΟΥ	26
ΕΙΚΟΝΑ 15. ΓΚΑΛΕΡΙ ΕΙΚΟΝΩΝ ΜΕ LOGO ΣΥΝΕΡΓΑΖΟΜΕΝΩΝ ΦΟΡΕΩΝ	26
ΕΙΚΟΝΑ 16. ΣΥΝΔΕΣΜΟΙ ΠΡΟΣ ΤΟΥΣ ΚΥΡΙΟΥΣ ΤΟΜΕΙΣ ΤΟΥ ΙΣΤΟΤΟΠΟΥ	26
ΕΙΚΟΝΑ 17. ΠΡΟΣΦΑΤΕΣ ΔΡΑΣΤΗΡΙΟΤΗΤΕΣ	27
ΕΙΚΟΝΑ 18. ΥΠΟΣΕΛΙΔΟ.....	27
ΕΙΚΟΝΑ 19. Ο ΤΟΜΕΑΣ "ΣΧΕΤΙΚΑ" (ABOUT)	28
ΕΙΚΟΝΑ 20. ΓΡΑΦΙΚΕΣ ΑΝΑΠΑΡΑΣΤΑΣΕΙΣ ΣΤΟΙΧΕΙΩΝ/ΣΤΑΤΙΣΤΙΚΩΝ.....	29
ΕΙΚΟΝΑ 21. ΟΙ ΕΡΕΥΝΗΤΙΚΕΣ ΠΕΡΙΟΧΕΣ ΤΟΥ ΕΡΓΑΣΤΗΡΙΟΥ	29
ΕΙΚΟΝΑ 22. ΚΥΡΙΟΣ ΚΑΤΑΛΟΓΟΣ ΕΡΕΥΝΗΤΙΚΩΝ ΤΟΜΕΩΝ	30
ΕΙΚΟΝΑ 23. ΑΤΟΜΙΚΗ ΣΕΛΙΔΑ ΜΕΛΟΥΣ-ΒΙΟΓΡΑΦΙΚΟ	31
ΕΙΚΟΝΑ 24. ΑΤΟΜΙΚΗ ΣΕΛΙΔΑ ΜΕΛΟΥΣ - ΔΗΜΟΣΙΕΥΣΕΙΣ ΚΑΙ ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ	31
ΕΙΚΟΝΑ 25. Η ΣΕΛΙΔΑ ΣΥΝΕΡΓΑΤΩΝ	32
ΕΙΚΟΝΑ 26. ΚΥΡΙΑ ΣΕΛΙΔΑ ΔΡΑΣΤΗΡΙΟΤΗΤΩΝ	33
ΕΙΚΟΝΑ 27. ΑΤΟΜΙΚΗ ΣΕΛΙΔΑ ΔΡΑΣΤΗΡΙΟΤΗΤΑΣ	33
ΕΙΚΟΝΑ 28. ΣΕΛΙΔΑ ΑΝΑΚΟΙΝΩΣΕΩΝ	34
ΕΙΚΟΝΑ 29. ΚΥΡΙΑ ΣΕΛΙΔΑ ΑΝΑΚΟΙΝΩΣΗΣ/ΑΡΘΡΟΥ, 1/2.....	34
ΕΙΚΟΝΑ 30. ΚΥΡΙΑ ΣΕΛΙΔΑ ΑΝΑΚΟΙΝΩΣΗΣ/ΑΡΘΡΟΥ, 2/2.....	35
ΕΙΚΟΝΑ 31. ΔΗΛΩΣΗ JSON ΩΣ ΤΥΠΟΣ ΠΕΡΙΕΧΟΜΕΝΟΥ.....	39
ΕΙΚΟΝΑ 32. ΣΥΝΔΕΣΗ ΣΤΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ	39
ΕΙΚΟΝΑ 33. ΕΡΩΤΗΜΑ ΠΡΟΣ ΤΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ.....	39
ΕΙΚΟΝΑ 34. Η ΣΥΝΑΡΤΗΣΗ ΚΩΔΙΚΟΠΟΙΗΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ ΣΕ ΜΟΡΦΗ JSON	39
ΕΙΚΟΝΑ 35. Η ΣΕΛΙΔΑ "SENSOR DATA" ΠΡΙΝ ΤΗΝ ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ.....	40
ΕΙΚΟΝΑ 36. ΦΟΡΜΑ ΓΙΑ ΤΗΝ ΑΙΤΗΣΗ ΤΟΥ SENSOR ID.....	40
ΕΙΚΟΝΑ 37. ΠΡΟΣΘΟΗΚΗ ΤΟΥ ΛΗΦΘΕΝΤΟΣ SENSOR ID ΣΤΗΝ ΑΙΤΗΣΗ ΠΡΟΣ ΤΟ API.....	40
ΕΙΚΟΝΑ 38. ΑΡΧΙΚΟΠΟΙΗΣΗ CURL	41
ΕΙΚΟΝΑ 39. ΑΠΟΚΩΔΙΚΟΠΟΙΗΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ.....	41
ΕΙΚΟΝΑ 40. Η ΣΕΛΙΔΑ SENSOR DATA ΜΕΤΑ ΤΗΝ ΑΠΑΝΤΗΣΗ	41
ΕΙΚΟΝΑ 41. ΣΥΝΔΕΣΗ ΤΟΥ ΑΡΧΕΙΟΥ ΜΕ ΤΙΣ ΜΕΘΟΔΟΥΣ JAVASCRIPT ΜΑΣ	42
ΕΙΚΟΝΑ 42. ΔΗΜΙΟΥΡΓΙΑ ΚΟΥΜΠΙΟΥ ΚΑΙ DIV ΠΟΥ ΘΑ ΠΕΡΙΕΧΕΙ ΤΟ ΑΠΟΤΕΛΕΣΜΑ.....	43
ΕΙΚΟΝΑ 43. Η ΣΥΝΑΡΤΗΣΗ-ΔΙΑΧΕΙΡΙΣΤΗΣ ΤΟΥ ΠΑΤΗΜΑΤΟΣ ΚΟΥΜΠΙΟΥ	43
ΕΙΚΟΝΑ 44. ΔΗΜΙΟΥΡΓΙΑ ΤΗΣ ΛΙΣΤΑΣ ΚΑΤΗΓΟΡΙΩΝ.....	44
ΕΙΚΟΝΑ 45. ΑΝΑΚΤΩΝΤΑΣ ΤΗΝ ΕΠΙΛΟΓΗ ΤΟΥ ΧΡΗΣΤΗ ΑΠΟ ΤΗΝ ΛΙΣΤΑ	44
ΕΙΚΟΝΑ 46. Η ΣΥΝΑΡΤΗΣΗ CREATEHTML ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΤΗΣ ΔΗΜΟΣΙΕΥΣΗΣ	44
ΕΙΚΟΝΑ 47. ΑΚΑΤΕΡΓΑΣΤΗ ΠΛΗΡΟΦΟΡΙΑ JSON	45
ΕΙΚΟΝΑ 48. Η ΣΕΛΙΔΑ ΠΡΙΝ ΤΗΝ ΕΠΙΛΟΓΗ ΤΟΥ ΧΡΗΣΤΗ	45
ΕΙΚΟΝΑ 49. Η ΣΕΛΙΔΑ ΜΕΤΑ ΤΗΝ ΦΟΡΤΩΣΗ ΔΗΜΟΣΙΕΥΣΕΩΝ	46
ΕΙΚΟΝΑ 50. ΔΗΜΙΟΥΡΓΙΑ ΠΕΡΙΟΧΩΝ ΕΙΣΑΓΩΓΗΣ ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΔΗΜΟΣΙΕΥΣΗΣ	47

ΕΙΚΟΝΑ 51. ΜΟΡΦΟΠΟΙΗΣΗ CSS	47
ΕΙΚΟΝΑ 52. Η ΣΕΛΙΔΑ ΠΡΟΣΘΗΚΗΣ ΔΗΜΟΣΙΕΥΣΗΣ (ΩΣ ADMIN)	48
ΕΙΚΟΝΑ 53. Η ΣΥΝΑΡΤΗΣΗ-ΔΙΑΧΕΙΡΙΣΤΗΣ ΤΟΥ ΠΑΤΗΜΑΤΟΣ ΓΙΑ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΔΗΜΟΣΙΕΥΣΗΣ	48
ΕΙΚΟΝΑ 54. ΑΙΤΗΣΗ "POST" ΠΡΟΣ ΤΟ WORDPRESS API	48
ΕΙΚΟΝΑ 55. ΜΕΤΑΤΡΟΠΗ ΤΟΥ ΑΝΤΙΚΕΙΜΕΝΟΥ ΣΕ JSON STRING ΚΑΙ ΑΠΟΣΤΟΛΗ	49
ΕΙΚΟΝΑ 56. ΠΕΡΝΩΝΤΑΣ ΤΟ ΜΟΝΑΔΙΚΟ ΑΝΑΓΝΩΡΙΣΤΙΚΟ ΧΡΗΣΤΗ ΣΤΟ WORDPRESS API	49
ΕΙΚΟΝΑ 57. ΑΝΑΚΤΗΣΗ ΤΟΥ ΑΝΑΓΝΩΡΙΣΤΙΚΟΥ ΧΡΗΣΤΗ	49
ΕΙΚΟΝΑ 58. Η ΟΛΟΚΛΗΡΩΜΕΝΗ ΑΙΤΗΣΗ AJAX.....	50
ΕΙΚΟΝΑ 59. ΈΛΕΓΧΟΣ ΓΙΑ ΛΑΘΗ ΚΑΙ ΚΑΤΑΣΤΑΣΗ ΤΗΣ ΔΗΜΟΣΙΕΥΣΗΣ	50
ΕΙΚΟΝΑ 60. Η ΚΑΜΠΥΛΗ ΑΝΑΖΗΤΗΣΕΩΝ ΑΝΑ ΛΕΞΕΙΣ/ΚΛΕΙΔΙΑ	58
ΕΙΚΟΝΑ 61. ΔΗΜΟΤΙΚΟΤΗΤΑ ΤΟΥ ΟΡΟΥ ΙΟΤ ΣΕ ΑΝΑΖΗΤΗΣΕΙΣ GOOGLE	60
ΕΙΚΟΝΑ 62. ΔΗΜΟΤΙΚΟΤΗΤΑ ΕΠΟΧΙΑΚΩΝ ΟΡΩΝ	61
ΕΙΚΟΝΑ 63. ΜΕΤΑ-ΠΕΡΙΓΡΑΦΕΣ ΣΕ ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΝΑΖΗΤΗΣΗΣ	64

Πίνακας Περιεχομένων

Πρόλογος.....	iii
Περίληψη.....	iv
Abstract	v
1 Το Διαδίκτυο.....	6
1.1 Εισαγωγή.....	6
1.2 Σχετική Ορολογία.....	6
1.2.1 World-Wide Web.....	6
1.2.2 Web browser.....	6
1.2.3 Web server.....	7
1.2.4 Web site (Ιστότοπος).....	7
1.2.5 Web page (Ιστοσελίδα).....	7
1.3 Στατικές και Δυναμικές Σελίδες.....	8
1.3.1 Στατικές Σελίδες.....	8
1.3.2 Δυναμικές Σελίδες.....	8
2 Συστήματα Διαχείρισης Περιεχομένου.....	10
2.1 Εισαγωγή.....	10
2.2 Τα Διαφορετικά Είδη CMS.....	10
2.2.1 Web CMS.....	10
2.2.2 ECMS.....	11
2.2.3 MCMS.....	11
2.2.4 CCMS.....	11
2.3 Πλεονεκτήματα και μειονεκτήματα των CMS.....	11
2.4 Σύγκριση και Επιλογή CMS.....	12
2.4.1 Κεντρικό Σύστημα Ελέγχου / Centralized Version Control System.....	13
2.4.2 Κατανεμημένο Σύστημα Ελέγχου / Distributed Version Control System.....	13
3 Το Σύστημα διαχείρισης Wordpress.....	15
3.1 Εισαγωγή.....	15
3.2 WordPress.com και WordPress.org.....	15
3.3 Εγκατάσταση του WordPress.....	16
3.3.1 Απαιτήσεις Συστήματος.....	16
3.3.2 Κατεβάζοντας το WordPress.....	16
3.3.3 Εγκατάσταση WordPress.....	17
3.3.4 Ο πίνακας ελέγχου WordPress.....	21

4	Δημιουργώντας τη σελίδα.....	24
4.1	Λειτουργικές απαιτήσεις.....	24
4.1.1	Πρώτη Σελίδα	24
4.1.2	Σχετικά με το εργαστήριο (About).....	28
4.1.3	Research - Έρευνα	30
4.1.4	Staff - Προσωπικό.....	30
4.1.5	Affiliates – Συνεργάτες	31
4.1.6	Events – Δραστηριότητες.....	32
4.1.7	Νέα και Ανακοινώσεις	33
5	Representational State Transfer (REST).....	36
5.1	Εισαγωγή.....	36
5.2	Τι είναι η αρχιτεκτονική REST:	36
5.2.1	Διαχωρισμός Client – Server.	36
5.2.2	Statelessness	37
5.2.3	Επικοινωνία Client – Server	37
5.3	REST API.....	37
5.3.1	JSON	37
5.3.2	Το REST API του WordPress.....	38
5.4	Εφαρμογή #1 - Δημιουργία ενός PHP REST API.....	38
5.4.1	Κατασκευάζοντας το API.....	38
5.4.2	Αναλώνοντας το API.....	40
5.5	Εφαρμογή #2 – Φόρτωση Posts με AJAX Requests.....	42
5.5.1	Αίτηση AJAX	42
5.5.2	Δημιουργία και διασύνδεση του χώρου εργασίας (Workspace)	42
5.5.3	Δημιουργώντας την εφαρμογή.....	43
5.6	Εφαρμογή #3 – Γρήγορη δημιουργία Post από το Front-End της σελίδας	47
5.6.1	Κατασκευάζοντας το Front-End	47
5.6.2	Κατασκευάζοντας το Back-End.....	48
6	Μηχανές αναζήτησης και Βελτιστοποίηση	51
6.1	Εισαγωγή:.....	51
6.2	Ιστορία και Εξέλιξη του SEO.....	51
6.3	Μηχανές Αναζήτησης.....	52
6.3.1	Μηχανές αναζήτησης βασισμένες σε Περιηγητή (Crawler-Based)	52
6.3.2	Πληρωμένα και Οργανικά αποτελέσματα	54
6.4	Πως Βελτιστοποιείται ένας ιστότοπος	54
6.4.1	Κάνοντας τη σελίδα μας προσπελάσιμη.....	54

6.4.2	Έρευνα Λέξεων-Κλειδιών (Keyword Research)	57
6.5	Βελτιστοποίηση σε επίπεδο Σελίδας.....	62
6.5.1	Στοιχεία Περιεχομένου.....	62
6.5.2	Στοιχεία HTML.....	63
6.5.3	Στοιχεία Αρχιτεκτονικής	64
7	Προσβασιμότητα.....	66
7.1	Εισαγωγή.....	66
7.2	Βελτίωση προσβασιμότητας ιστότοπου	67
7.2.1	Επιλογή Κατάλληλου CMS	67
7.2.2	Αποτελεσματική Χρήση Επικεφαλίδων	67
7.2.3	Χρήση Εναλλακτικού Κειμένου για Εικόνες.....	67
7.2.4	Χρήση μοναδικών και περιγραφικών ονομάτων Συνδέσμων	68
7.2.5	Χρήση Χρώματος.....	69
7.2.6	Φόρμες.....	70
7.2.7	Συμβατότητα Πληκτρολογίου.....	72
7.3	Έλεγχος Προσβασιμότητας Σελίδας	73
	Βιβλιογραφία	74

1 ΤΟ ΔΙΑΔΙΚΤΥΟ

1.1 ΕΙΣΑΓΩΓΗ

Το Internet, ή διαδίκτυο, είναι ένα παγκόσμιο σύστημα αλληλοσυνδεδεμένων δικτύων υπολογιστών, που χρησιμοποιεί την σουίτα πρωτοκόλλων “TCP/IP” για την επικοινωνία μεταξύ δικτύων και συσκευών. Πρόκειται για ένα «δίκτυο δικτύων» που αποτελείται από ιδιωτικά, δημόσια, ακαδημαϊκά, εταιρικά και κυβερνητικά δίκτυα τοπικής ή παγκόσμιας εμβέλειας, συνδεδεμένα με μια ευρεία γκάμα ηλεκτρονικών συσκευών. Διαθέτει πληθώρα πληροφοριών και υπηρεσιών, όπως τα έγγραφα HTTP/ιστοσελίδες, και εφαρμογές όπως ηλεκτρονικό ταχυδρομείο, τηλεφωνία και διαμοιρασμό αρχείων.

Το Internet έχει ξεπεράσει τα παραδοσιακά μέσα επικοινωνίας (όπως τηλεφωνία, ραδιόφωνο, τηλεόραση κ.α.) και έχει δημιουργήσει νέες υπηρεσίες όπως διαδικτυακή τηλεφωνία/τηλεόραση, OnLine μουσική, ψηφιακές εφημερίδες και πολλές άλλες. Έντυπες εκδόσεις πλέον έχουν προσαρμοστεί σε ιστοσελίδες, ή έχουν μεταμορφωθεί σε Blog και Web feeds. Το διαδίκτυο έχει διευκολύνει και επιταχύνει νέες μορφές αλληλεπίδρασης, μέσω φόρουμ και υπηρεσιών κοινωνικής δικτύωσης. Και σημαντικότερα για τους σκοπούς αυτής της εργασίας, παρέχει μια πλατφόρμα για ακαδημαϊκές οντότητες και πρόσωπα να προωθήσουν το έργο τους και να συνεργαστούν πιο αποτελεσματικά [1].

1.2 ΣΧΕΤΙΚΗ ΟΡΟΛΟΓΙΑ

1.2.1 World-Wide Web

Παγκόσμιος Ιστός, ή αλλιώς World-Wide Web (WWW) είναι ένα σύστημα πληροφοριών, στο οποίο αρχεία και άλλοι διαδικτυακοί πόροι αναγνωρίζονται από μια διεύθυνση , ή URL , η οποία μπορεί να είναι διασυνδεδεμένη και αναφερόμενη μέσω συνδέσμων. Με βάση αυτή τη διεύθυνση οι πόροι μπορούν να προσπελαύνονται μέσω του διαδικτύου. Συγκεκριμένα, μεταφέρονται μέσω του πρωτοκόλλου HTTP (Hypertext Transfer Protocol), προσπελαύνονται από χρήστες μέσω μίας εφαρμογής περιηγητή (web browser) και εκδίδονται/φιλοξενούνται σε έναν web server [2].

1.2.2 Web browser

Ένας web browser, η περιηγητής ουσιαστικά είναι μια εφαρμογή μέσω της οποίας ένας χρήστης αποκτά πρόσβαση στο διαδίκτυο. Όταν ο χρήστης ζητά μια συγκεκριμένη σελίδα από έναν συγκεκριμένο ιστότοπο, ο περιηγητής «αποσπά» το απαραίτητο περιεχόμενο από τον εκάστοτε server και εμφανίζει την σελίδα στη συσκευή του χρήστη. Ο δημοφιλέστερος web browser την περίοδο συγγραφής αυτής της εργασίας είναι το **Google Chrome**, ακολουθούμενο από το **Safari** [3]. Άλλοι σημαντικοί περιηγητές είναι οι **Mozilla Firefox**, **Microsoft Edge** και **Opera** [4].

1.2.3 Web server

Ένας Web Server είναι λογισμικό, ή το υλικό που χρησιμοποιείται για να «τρέξει» το λογισμικό αυτό, το οποίο είναι υπεύθυνο για να ανταποκρίνεται σε αιτήσεις χρηστών στον παγκόσμιο ιστό. Ένας web server τυπικά μπορεί να περιέχει έναν ή περισσότερους ιστότοπους, και επεξεργάζεται εισερχόμενες αιτήσεις HTTP και άλλων σχετικών πρωτοκόλλων.

Η κύρια λειτουργία του είναι να αποθηκεύσει, επεξεργαστεί, και να μεταβιβάσει ιστοσελίδες στους χρήστες [5]. Η επικοινωνία client-server λαμβάνει χώρα χρησιμοποιώντας HTTP, και οι σελίδες που παραδίδονται είναι συχνότερα αρχεία HTML, που μπορεί να περιλαμβάνουν αρχεία εικόνας, οδηγίες μορφοποίησης (CSS) ή και εκτελέσιμα τμήματα κώδικα πέραν του κειμένου.

Συνήθως, ένας πράκτορας χρήστη (είτε περιηγητής είτε web crawler) επικοινωνεί με αίτημα για έναν συγκεκριμένο πόρο χρησιμοποιώντας HTTP, και ο server ανταποκρίνεται με το σχετικό περιεχόμενο, ή ένα μήνυμα σφάλματος εάν δεν ήταν δυνατή η μεταβίβαση της πληροφορίας [6].

1.2.4 Web site (Ιστότοπος)

Ιστότοπος είναι μια συλλογή ιστοσελίδων και άλλου σχετικού περιεχομένου, που αναγνωρίζεται με ένα κοινό όνομα (domain name), και φιλοξενείται σε τουλάχιστον έναν web server. Όλοι οι δημοσίως προσβάσιμοι ιστότοποι αποτελούν μαζί τον παγκόσμιο ιστό. Υπάρχουν επίσης ιδιωτικοί ιστότοποι που μπορούν να προσπελαστούν μόνο σε ένα ιδιωτικό δίκτυο (Private network), όπως ένας εσωτερικός εταιρικός ιστότοπος για τους υπαλλήλους.

Οι ιστότοποι είναι τυπικά αφιερωμένοι σε ένα σκοπό, ή μία θεματολογία, όπως ειδήσεις, εκπαίδευση, εμπόριο, ψυχαγωγία Κ.Ο.Κ. Χρήση συνδέσμων μεταξύ ιστοσελίδων διευκολύνει την πλοήγηση στον ιστότοπο, και συνήθως ξεκινά σε μια πρώτη σελίδα (Home page) [7].

1.2.5 Web page (Ιστοσελίδα)

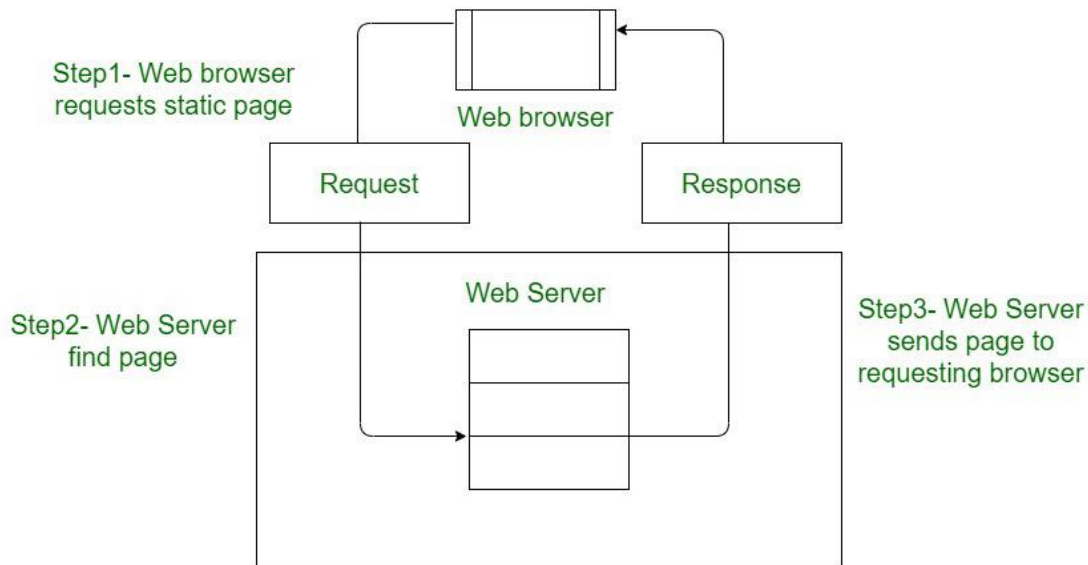
Ιστοσελίδα είναι μια συγκεκριμένη συλλογή πληροφοριών, εμφωλευμένη σε έναν ιστότοπο και προσβάσιμη σε έναν χρήστη μέσω ενός περιηγητή.

Κάθε σελίδα διαθέτει ένα διακριτό URL. Όταν ο χρήστης εισάγει ένα URL στον περιηγητή του, τα στοιχεία της σελίδας ανακτώνται από τον εκάστοτε web server που φιλοξενεί τον ιστότοπο. Ο περιηγητής στη συνέχεια «μεταμορφώνει» τα στοιχεία της σελίδας σε μια οπτική αναπαράσταση στη συσκευή του χρήστη. Οι ιστοσελίδες είναι δύο κυρίων ειδών. Στατικές, και Δυναμικές.

1.3 ΣΤΑΤΙΚΕΣ ΚΑΙ ΔΥΝΑΜΙΚΕΣ ΣΕΛΙΔΕΣ

1.3.1 Στατικές Σελίδες

Στους στατικούς ιστότοπους, οι σελίδες που ανακτώνται από τον server είναι προκατασκευασμένα αρχεία κώδικα, σε μια απλή γλώσσα όπως HTML, CSS, η JavaScript. Οι ιστοσελίδες ανακτώνται από τον server χωρίς αλλαγές, έχοντας ως αποτέλεσμα τη γρηγορότερη φόρτωση των στατικών σελίδων. Δεν υπάρχει αλληλεπίδραση με βάσεις δεδομένων, και το κόστος είναι σχετικά μικρότερο, καθώς ο host δεν χρειάζεται να υποστηρίξει επεξεργασία στο επίπεδο του server (server-side) με διαφορετικές γλώσσες. [8]



Εικόνα 1. Αρχιτεκτονική ενός στατικού Ιστότοπου

Χάρη στην σχετική τους απλότητα, ένας στατικός ιστότοπος δεν χρειάζεται εξειδικευμένη γνώση για να κατασκευαστεί. Μπορεί να γραφεί με HTML/CSS ακόμα και σε ένα απλό αρχείο κειμένου. Είναι ιδιαιτέρως χρήσιμο για άτομα με βασικές γνώσεις Development που θέλουν να δημιουργήσουν έναν ιστότοπο με λίγες σελίδες που απεικονίζουν πληροφορίες.

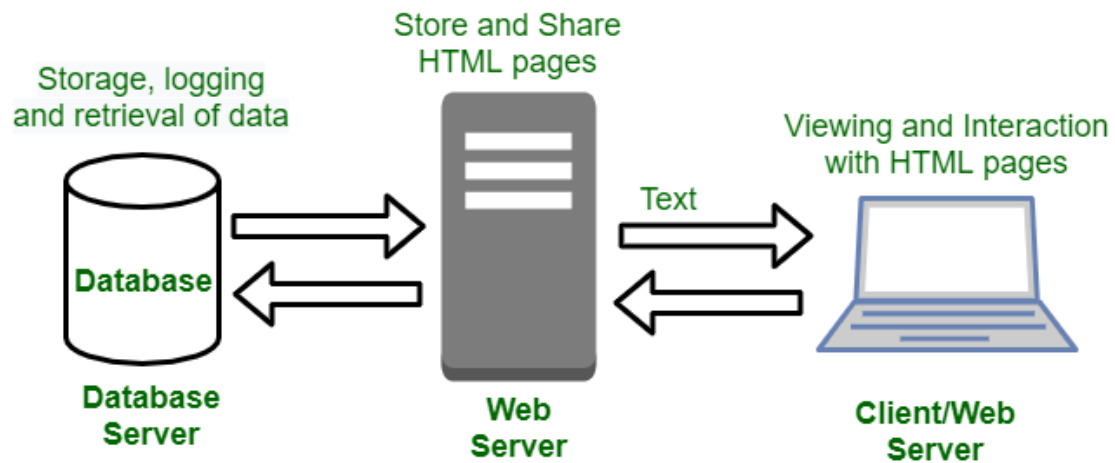
Ένα ακόμα πλεονέκτημα των στατικών ιστότοπων είναι ότι βελτιώνουν τον χρόνο φόρτωσης των σελίδων, καθώς ο περιηγητής δεν κάνει τόσες αιτήσεις στον server όσες σε μια δυναμική σελίδα, οπότε το περιεχόμενο μπορεί να ανακτηθεί γρηγορότερα. Ο χρόνος φόρτωσης είναι μια από τις μετρικές της Google όσον αφορά τη βελτιστοποίηση (SEO), και η μείωση του χρόνου φόρτωσης μιας σελίδας είναι ένας καλός τρόπος να βελτιώσουμε την κατάταξή της στις αναζητήσεις [9].

1.3.2 Δυναμικές Σελίδες

Στους δυναμικούς ιστότοπους, οι ιστοσελίδες που ανακτώνται από τον server βρίσκονται υπό επεξεργασία κατά την ανάκτηση, που σημαίνει ότι δεν είναι προκατασκευασμένες αλλά «χτίζονται» κατά τις απαιτήσεις του χρήστη, με τη βοήθεια server-side γλωσσών προγραμματισμού όπως PHP, Node.js, ASP.NET και πολλές περισσότερες που υποστηρίζονται από τον server. Ως αποτέλεσμα, είναι πιο αργές από στατικές σελίδες αλλά ενημερώσεις και αλληλεπίδραση με βάσεις δεδομένων είναι εφικτές.

Χρησιμοποιούνται πολύ περισσότερο από στατικούς ιστότοπους καθώς οι ενημερώσεις μπορούν να γίνουν πολύ γρήγορα και εύκολα, καθώς είναι πιθανό να γίνει μια μόνο κοινή

αλλαγή που θα εφαρμοστεί σε ολόκληρο τον ιστότοπο αυτόματα ενώ σε έναν στατικό ιστότοπο θα έπρεπε να γίνουν αλλαγές σε κάθε σελίδα [8].



Εικόνα 2. Αρχιτεκτονική ενός Δυναμικού Ιστότοπου

Για ότι προσφέρει ένας στατικός ιστότοπος με γρηγορότερους χρόνους φόρτωσης, οι δυναμικοί ιστότοποι το αντισταθμίζουν με αυξημένη λειτουργικότητα και ευχρηστία. Οι δυναμικοί ιστότοποι είναι ικανοί να αλλάζουν τα δεδομένα τους καθώς ο χρήστης πλοηγείται στις ιστοσελίδες. Παραδείγματος χάριν, μια αρχική σελίδα ενός Blog όπου το πιο πρόσφατο άρθρο εμφανίζεται πρώτο, θα είναι σχεδόν πάντα δυναμική. Οι διαδικασίες προσθήκης του άρθρου στον ιστότοπο και η ανάκτησή τους από την πρώτη σελίδα στηρίζονται σε βάσεις δεδομένων, προσβάσιμες από το backend της σελίδας.

Εάν χρησιμοποιηθεί ένα CMS (Content Management System) ή λογισμικό Website-builder, η δυσκολία κατασκευής μειώνεται κατά πολύ περισσότερο από την σύνταξη ενός αρχείου HTML/CSS για μια στατική σελίδα. Αντί ο ιδιοκτήτης της σελίδας να πρέπει να ζητήσει από τον developer να κάνει αλλαγές στο περιεχόμενο, μπορούν απλά να συνδεθούν στο CMS και να κάνουν τις αλλαγές αυτοπροσώπως.

Για τους σκοπούς της εργασίας, και για το είδος της απαιτούμενης σελίδας, θα αναπτυχθεί ένας δυναμικός ιστότοπος, καθώς πέραν από στατικές πληροφορίες η σελίδα πρέπει να μπορεί να υποστηρίξει την έκδοση/δημοσιοποίηση άρθρων, δημιουργία δράσεων/events και διάφορες άλλες δυνατότητες. Η ανάπτυξη θα γίνει μέσω του WordPress CMS .

2 ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΕΡΙΕΧΟΜΕΝΟΥ

2.1 ΕΙΣΑΓΩΓΗ

Σύστημα διαχείρισης περιεχομένου, η CMS, είναι ένα σύστημα που χρησιμοποιείται για την διαχείριση περιεχομένου ενός ιστότοπου. Πρόκειται για μια υπολογιστική εφαρμογή που επιτρέπει την έκδοση, επεξεργασία, τροποποίηση, οργάνωση και συντήρηση του περιεχομένου μέσω ενός κεντρικού interface. Τυπικά αποτελείται από δύο διαφορετικά στοιχεία [10]:

- Εφαρμογή διαχείρισης περιεχομένου / Content Management Application (CMA)
- Εφαρμογή παράδοσης περιεχομένου / Content Delivery Application (CDA)

Στο επίπεδο του CMA, ένας χρήστης μπορεί να διαχειριστεί την δημιουργία και επεξεργασία κειμένου σε έναν ιστότοπο χωρίς να χρειαστεί υποστήριξη από έναν web developer. Εφαρμόζει μια σχεδιαστική νοοτροπία «What you see is what you get» όπου διευκολύνει τους χρήστες να κάνουν τις απαραίτητες αλλαγές χωρίς εξειδικευμένες γνώσεις. Όλες οι επιθυμητές αλλαγές μπορούν να γίνουν από διαφορετικά τμήματα του admin panel.

Όταν ολοκληρωθεί η διαδικασία επεξεργασίας, η εφαρμογή CDA θα πάρει το περιεχόμενο, και θα «συναρμολογήσει» τον κώδικα ώστε να το παρουσιάσει στους front-end χρήστες [11].

2.2 ΤΑ ΔΙΑΦΟΡΕΤΙΚΑ ΕΙΔΗ CMS

Υπάρχουν διαφορετικά είδη Συστημάτων Διαχείρισης Περιεχομένου για διαφορετικούς τύπους δεδομένων. Η φύση του περιεχομένου καθορίζει τον τύπο του CMS. Οι κύριες κατηγορίες είναι οι εξής:

- Web CMS
- Enterprise CMS
- Mobile CMS
- Component CMS

Οι διαφορετικοί τύποι CMS είναι σχεδιασμένοι για διαφορετικούς σκοπούς, για παράδειγμα υπάρχουν CMS για τη διαχείριση διαδικτυακού περιεχομένου και διαφορετικά για τις ανάγκες οργανισμών/εταιριών. Υπάρχουν επίσης συστήματα που επικεντρώνονται στην mobile βιομηχανία τεχνολογίας και αυτά που διαχειρίζονται το περιεχόμενο σε «μικροσκοπικό» επίπεδο [12].

2.2.1 Web CMS

Ένα Web CMS επιτρέπει στον χρήστη να δημιουργήσει και να τροποποιήσει ιστοσελίδες χωρίς να απαιτεί εξειδικευμένες γνώσεις. Βασίζεται σε μια σειρά αρχέτυπων (templates) τα οποία σχηματίζουν την μορφή ενός ιστότοπου και χρησιμοποιούνται για την παραγωγή περιεχομένου.

Ο συντάκτης του περιεχομένου το εισάγει σε αυτά τα αρχέτυπα, τα οποία κατόπιν δημοσιεύονται στον ιστότοπο.

2.2.2 ECMS

Αυτό το είδος CMS χρησιμοποιείται για την διαχείριση περιεχομένου, πόρων, εγγραφών και λοιπών πληροφοριών που ορίζουν τη δομή και ιεραρχία ενός οργανισμού. Περιλαμβάνει μια ευρεία γκάμα διαδικασιών, εργαλείων και στρατηγικών, όπως αρχειοθέτηση (indexing), ιχνηλάτηση (tracking) και άλλες, με απώτερο σκοπό την διευκόλυνση της διαχείρισης εταιρικών πληροφοριών.

2.2.3 MCMS

Η ραγδαία ανάπτυξη της mobile/κινητής βιομηχανίας τεχνολογίας οδήγησε σε αυξημένη ζήτηση συστημάτων διαχείρισης που εξειδικεύονται σε περιεχόμενο για Smartphone, PDA, κινητά τηλέφωνα και άλλες φορητές συσκευές.

2.2.4 CCMS

Component Content Management Systems είναι ένας εξειδικευμένος τύπος συστημάτων που διαχειρίζεται το περιεχόμενο σε «βαθύτερο» επίπεδο. Αυτό αναφέρεται σε μία μονάδα περιεχομένου, όπως μια παράγραφος κειμένου ή μια εικόνα, σε αντίθεση με ένα ολόκληρο αρχείο, η οποία βρίσκεται αποθηκευμένη στο σύστημα. Μπορεί να επαναχρησιμοποιηθεί μέσα σε ένα αρχείο ή σε μια σειρά αρχείων, και κάθε μονάδα περιεχομένου υπάρχει στο σύστημα και ως ξεχωριστή οντότητα αλλά και ως μέρος ενός μεγαλύτερου συστήματος.

2.3 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΚΑΙ ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΤΩΝ CMS

Στα πλαίσια της πτυχιακής εργασίας, θα χρησιμοποιήσουμε Web CMS, και συγκεκριμένα open-source Web CMS. Η χρήση CMS για την κατασκευή του ιστότοπου έχει πολλά θετικά στοιχεία, όπως:

- Πολλές επιλογές, όπως WordPress, Joomla!, Drupal.
- Μικρός χρόνος Development.
- Μικρό συνολικό κόστος, συμπεριλαμβανομένου εξόδων κατασκευής και διαχείρισης.
- Δεν είναι απαραίτητη η γνώση ή η συγγραφή κώδικα.
- Μεγάλη ποικιλία αρχέτυπων (templates).
- Απλά και εύχρηστα interface.
- Απλή και αποτελεσματική διαδικασία δημιουργίας και έκδοσης περιεχομένου.
- Μεγάλη ποικιλία επεκτάσεων, widgets/plugins, για εύκολη παραμετροποίηση του περιεχομένου.

- Ένας developer μπορεί να αναλάβει τον σχεδιασμό, και να παραδώσει το site στον ιδιοκτήτη χωρίς περαιτέρω ανάγκη συντήρησης/διαχείρισης.

Κάποια αρνητικά στοιχεία χρήσης ενός CMS:

- Όσο πιο λεπτομερή πρέπει να είναι τα περιεχόμενα ενός site, τόσο μειώνεται η αποτελεσματικότητα ενός CMS.
- Παρόλο που ενημερώνονται συνεχώς, κάποια εργαλεία και δυνατότητες των CMS είναι ακόμα περιορισμένες. Πλήρης παραμετροποίηση δεν είναι πάντα δυνατή.
- Είναι απαραίτητη κάποια εκπαίδευση / γνώση του συστήματος και των λειτουργιών του για την αποτελεσματική χρήση και συντήρηση.
- Είναι πολύ εύκολο να «σπάσει» μια σελίδα ή να δημιουργηθούν σοβαρά προβλήματα με μικρά λάθη από τον admin.
- Καθώς τα πιο δημοφιλή CMS είναι open-source, το ρίσκο ασφαλείας είναι αντιστοίχως μεγαλύτερο, ιδιαίτερα εάν δεν ενημερώνεται αρκετά συχνά.
- Χρησιμοποιεί περισσότερους πόρους από μια στατική σελίδα.

2.4 ΣΥΓΚΡΙΣΗ ΚΑΙ ΕΠΙΛΟΓΗ CMS

Το Joomla! είναι ένα από τα πλέον διαδεδομένα συστήματα διαχείρισης περιεχομένου για τη δημιουργία δυναμικών ιστοσελίδων [13]. Είναι συμβατό με MySQL, SQL Server, PostgreSQL βάσεις δεδομένων, και το κύριο χαρακτηριστικό του είναι ότι προσφέρει την μεγαλύτερη ποικιλία εργαλείων λειτουργικότητας, όπως γκαλερί εικόνων, φόρουμ, Blogs, υποστήριξη για άρθρα ειδήσεων κ.α. Παρομοίως, το Drupal επίσης προσανατολίζεται στη δημιουργία δυναμικών ιστότοπων. Έχει υψηλή συμβατότητα με διαφορετικές βάσεις δεδομένων, υψηλές δυνατότητες ασφαλείας, χαμηλούς χρόνους φόρτωσης και μεγάλη ποικιλία διαφορετικών ρόλων χρήστη. Μας επιτρέπει να περιορίζουμε πρόσβαση σε έναν συγκεκριμένο χρήστη έτσι ώστε να είναι ικανοί να τροποποιήσουν τις ιδιότητες μόνο μιας συγκεκριμένης λειτουργίας, ή ακόμα και συγκεκριμένες παραμέτρους της λειτουργίας αυτής.

Το WordPress αρχικά ήταν προσανατολισμένο στη δημιουργία Blogs, αλλά έχει εξελιχθεί για να παρέχει εφαρμογές ιστού και εργαλεία ηλεκτρονικού εμπορίου. Ένα από τα κύρια χαρακτηριστικά του WordPress είναι η εξαιρετική απόδοσή του σε επίπεδο SEO (Search Engine Optimization). Ο λόγος για αυτό είναι ότι το wordpress διαθέτει πολλά πρόσθετα και επεκτάσεις για να βελτιώσουν την κατάταξη μιας σελίδας όσον αφορά τις μηχανές αναζήτησης, περισσότερα από άλλα Web CMS [14]. Επιπροσθέτως, είναι δυνατό να δημιουργηθεί ένα απλό blog εντελώς δωρεάν, υπό το domain της πλατφόρμας (.wordpress.com). Επεκτάσεις για τη διευκόλυνση της βελτιστοποίησης SEO μπορούν να χρησιμοποιηθούν και σε Joomla! και σε Drupal, αλλά με μικρότερη αποτελεσματικότητα απ' ότι στο WordPress.

Υπάρχουν πολλές διαφορές μεταξύ των τριών αυτών δημοφιλέστερων CMS. Υπάρχουν πολλά σενάρια, αλλά πρακτικά ένας ιστότοπος ξεκινά με μικρό μέγεθος και λίγα χαρακτηριστικά / περιεχόμενο. Σταδιακά, με την αύξηση της επισκεψιμότητας, δημιουργία νέου περιεχομένου, αύξηση όγκου πωλήσεων κλπ. , δημιουργείται η ανάγκη για νέα εργαλεία, πρόσθετα και επεκτάσεις. Το WordPress προσφέρει πολλές επεκτάσεις για να προσθέσουμε συγκεκριμένη λειτουργικότητα σε μια σελίδα, αλλά δεν έχει το ίδιο εύρος λειτουργιών όπως η Joomla!. Τέλος,

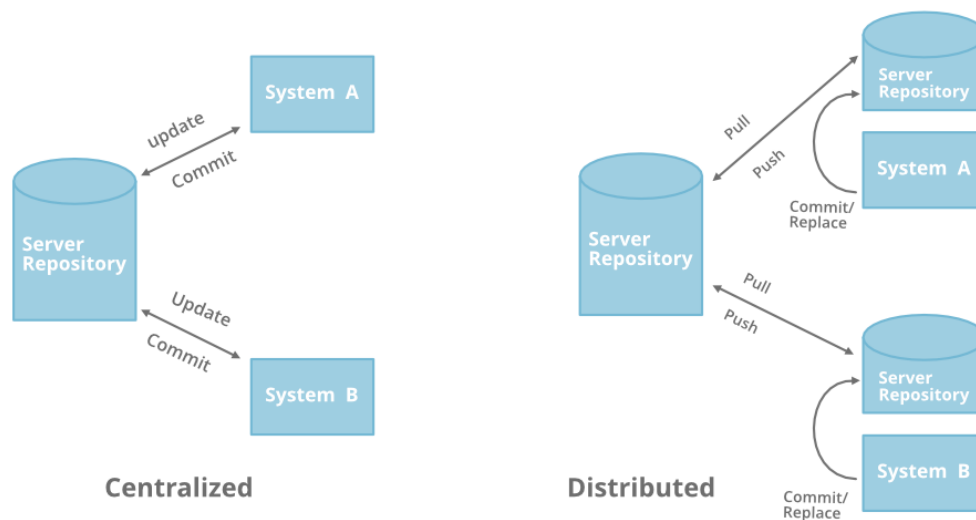
το Drupal είναι το πιο πολύπλοκο Web CMS από την σκοπιά διεύρυνσης των δυνατοτήτων, καθώς μια συγκεκριμένη λειτουργικότητα μπορεί να έχει εξαρτήσεις από διαφορετικές βιβλιοθήκες και συναρτήσεις, που συχνά καθιστά την διαδικασία μακρόσυρτη και ευτελή. Οι χώροι αποθήκευσης (Repositories) είναι πολύ σημαντικοί για την ενημέρωση των open source Web CMS. Αυτοί μπορεί να είναι καταναμημένοι (distributed repositories) όπως στην περίπτωση της Joomla! και του WordPress, είτε κεντρικοί (Centralized Repositories) στην περίπτωση του Drupal.

2.4.1 Κεντρικό Σύστημα Ελέγχου / Centralized Version Control System

Σε αυτό το σύστημα, υπάρχει ένας server και ένας client. Ο server είναι το κύριο repository που περιέχει όλες τις διαφορετικές εκδόσεις του κώδικα. Για εργασία σε οποιοδήποτε project πρώτα ένας χρήστης πρέπει να πάρει τον κώδικα από τον server. Οπότε επικοινωνεί και ανακτά όλο τον κώδικα, ή την τωρινή έκδοση στην συσκευή του. Ουσιαστικά χρειάζεται να πάρουμε μια ενημέρωση από τον server και μετά έχουμε ένα τοπικό αντίγραφο του κώδικα στο σύστημά μας, οπότε και μπορούμε να κάνουμε αλλαγές πριν ξανακάνουμε commit στον server. Όλα είναι κεντρικά σε αυτό το μοντέλο, υπάρχει μόνο ένα repository, και αυτό θα περιέχει όλο το ιστορικό αλλαγών του κώδικα και τα διαφορετικά παρακλάδια.

2.4.2 Καταναμημένο Σύστημα Ελέγχου / Distributed Version Control System

Σε ένα καταναμημένο σύστημα, το μεγαλύτερο μέρος του μηχανισμού είναι παρόμοιο με ένα κεντρικό. Η κύρια διαφορά είναι ότι εκτός από ένα μοναδικό repository, κάθε developer έχει τον δικό του server και θα έχουν ένα αντίγραφο όλου του ιστορικού αλλαγών και τα παρακλάδια του κώδικα στην τοπική τους συσκευή. Ουσιαστικά κάθε developer μπορεί να εργαστεί τοπικά και κυριότερα, “offline”, που είναι πολύ πιο βολικό από μια κεντρική δομή.



Εικόνα 3. Καταναμημένα και Κεντρικά Συστήματα Version Control

Οι μεγαλύτερες κοινότητες χρηστών ανήκουν στο WordPress και στην Joomla!, με αποτέλεσμα μεγαλύτερη δραστηριότητα και καλύτερη υποστήριξη. Η κοινότητα της Drupal συγκριτικά είναι περιορισμένη γιατί είναι πιο πολύπλοκη στην εφαρμογή και διαχείριση. Αντ’ αυτού, η πολυπλοκότητα της Drupal όσον αφορά τους ρόλους χρήστη παρέχει ένα σημαντικό

πλεονέκτημα σε σύγκρισή με τα άλλα 2 CMS. Οι διαφορές φαίνονται καλύτερα στον παρακάτω πίνακα [15].

Feature	Joomla!	Drupal	WordPress
Main content type	Websites, online apps	Blog	Blog, e-commerce, online apps
Extension availability	High	Middle	High
Functionality range	High	Middle	High
Extension repository	Distributed	Centralized	Distributed
Documentation	Excellent	Good	Excellent
User community	Very active	Limited	Very active
Ease of use	Simple	Complex	Simple
User role personalization	Middle	Very High	Middle
Manual SEO positioning	Yes	Yes	Yes
Automatic SEO positioning	Extensions	Modules	Plugins and tools

Εικόνα 4. Σύγκριση Χαρακτηριστικών Joomla!, Drupal και WordPress

Το CMS που θα χρησιμοποιήσουμε για τους σκοπούς της εργασίας είναι το WordPress CMS.

3 ΤΟ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ WORDPRESS

3.1 ΕΙΣΑΓΩΓΗ

Το WordPress είναι ένα open-source CMS, υπό την άδεια GPLv2. Οποιοσδήποτε έχει τη δυνατότητα να χρησιμοποιήσει και να τροποποιήσει το λογισμικό δωρεάν. Πρόκειται για ένα εργαλείο που διευκολύνει τους δημιουργούς διαδικτυακού περιεχομένου στη διαχείριση των ιστότοπων τους χωρίς να απαιτεί προγραμματιστικές γνώσεις. Πρόκειται για το δημοφιλέστερο CMS, με μερίδιο αγοράς 59,7% [16]. Το WordPress δημιουργήθηκε από τους Mike Little και Matt Mullenweg, ως διάδοχος (και ως παρακλάδι κώδικα) ενός απλού λογισμικού blog ονόματι b2/cafeblog. Η πρώτη του έκδοση (Version 0,7) ήταν στις 27 Μαΐου 2003.

3.2 WORDPRESS.COM ΚΑΙ WORDPRESS.ORG

Το WordPress μας επιτρέπει να δημιουργήσουμε μια σελίδα μέσω 2 “οδών”, των wordpress.com και wordpress.org . Η κύρια διαφορά μεταξύ των δύο είναι ποιος ακριβώς είναι ο πραγματικός host. Χρησιμοποιώντας WordPress.org μπορούμε να κατεβάσουμε τον κώδικα και να τον εγκαταστήσουμε σε μια συσκευή τοπικά, η σε οποιοδήποτε server. Αλλά στην περίπτωση του WordPress.com, όλα τα τμήματα της διαδικασίας από hosting μέχρι διαχείριση περιεχομένου μέσω επεκτάσεων και προσθέτων είναι «έτοιμα». Εννοώντας ότι δεν χρειάζεται να στήσουμε web server ή να πληρώσουμε για web hosting, ή να κατεβάσουμε οποιαδήποτε εφαρμογή.

Wordpress.com

- Απαιτεί εγγραφή/δημιουργία λογαριασμού για την δημιουργία ιστότοπου.
- Παρέχει δωρεάν βασική λειτουργικότητα, απαιτεί πληρωμή για πιο εξειδικευμένο Development.
- Δεν επιτρέπεται το ανέβασμα/εγκατάσταση προσθέτων.
- Το δωρεάν πρόγραμμα έχει ως αποτέλεσμα η σελίδα να είναι πάντα Sub-domain του WordPress.

Wordpress.org

- Απαιτεί ξεχωριστό web hosting για να «τρέξει» μια σελίδα.
- Βοηθά στη δημιουργία ενός site, είναι δωρεάν και open-source
- Δυνατότητα εγκατάστασης προσθέτων για επέκταση λειτουργικότητας.
- Δυνατότητα χρήσης custom URL.

Εάν δεν σκοπεύουμε να πληρώσουμε hosting ή να διαχειριστούμε έναν web server, το WordPress.com είναι η κατάλληλη επιλογή, καθώς είναι δωρεάν και η εγκατάσταση μπορεί να γίνει γρήγορα και εύκολα. Εάν θέλουμε ωστόσο να αλλάξουμε την ονομασία του site (να αφαιρέσουμε τον όρο WordPress από το URL) θα υπάρχει κάποια χρέωση, όπως και εάν θέλουμε να εγκαταστήσουμε συγκεκριμένα themes/templates και πρόσθετα. Επίσης, δεν μπορούμε να κάνουμε αλλαγές στον κώδικα PHP στο παρασκήνιο, και στο δωρεάν πρόγραμμα, διαφημίσεις θα εμφανίζονται στη σελίδα μας.

Το WordPress.org είναι μία ιδιο-φιλοξενούμενη (self-hosted) έκδοση και παρέχει επιπρόσθετη δυνατότητα ελέγχου του ιστότοπου και μεγαλύτερη ελευθερία κατά την κατασκευή. Μπορούμε να χρησιμοποιήσουμε domain name που έχουμε αγοράσει, και να ανεβάσουμε custom θέματα και αρχέτυπα, και να εγκαταστήσουμε πρόσθετα και επεκτάσεις.

3.3 ΕΓΚΑΤΑΣΤΑΣΗ ΤΟΥ WORDPRESS

3.3.1 Απαιτήσεις Συστήματος

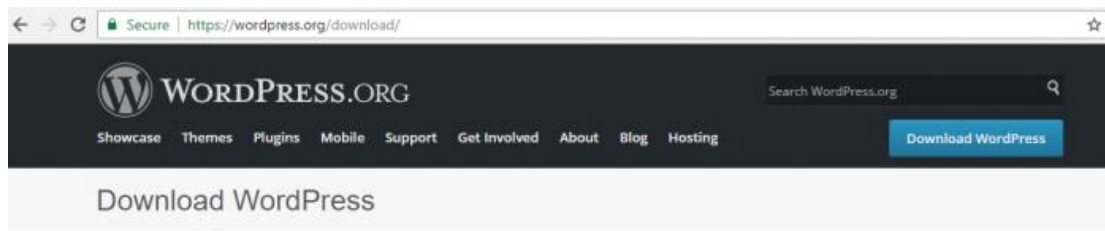
Ένας από τους λόγους που το WordPress είναι τόσο δημοφιλές είναι οι πολύ χαμηλές απαιτήσεις συστήματος για την λειτουργία του στον web server. Πολλές υπηρεσίες web hosting επίσης χρησιμοποιούν αυτόματα πακέτα εγκατάστασης για την απλοποίηση της διαδικασίας.

- Εάν χρησιμοποιούμε Windows, τότε είναι η απαραίτητη η εγκατάσταση WAMP (Windows, Apache, MySQL, PHP).
- Για Linux, είναι απαραίτητη η εγκατάσταση LAMP (Linux, Apache, MySQL, PHP).
- Για MAC, απαιτείται η εγκατάσταση MAMP (Macintosh, Apache, MySQL, PHP).
- Εναλλακτικά μπορούμε να εγκαταστήσουμε XAMPP, που είναι μια Multi-Platform εφαρμογή, και περιλαμβάνει Apache, MariaDB, PHP, και Perl.
- Το WordPress είναι συμβατό με PHP 5.2+ και MySQL 5+.

Οι επίσημες συστάσεις από το WordPress για τον Host (Δεκ. 2020) είναι PHP 7.4 ή νεότερη, MySQL 5.6 / MariaDB 10.1 ή νεότερη, και υποστήριξη HTTPS.

3.3.2 Κατεβάζοντας το WordPress

Μπορούμε να κατεβάσουμε το WordPress απευθείας από την επίσημη σελίδα της πλατφόρμας.



Stable Download

Requirements

Beta Releases

Nightly Builds

Download Counter

Release Archive

Source Code

The latest stable release of WordPress (Version 4.7.3) is available in two formats from the links to your right. If you have no idea what to do with this download, we recommend signing up with one of our [web hosting partners](#) that offers a one-click install of WordPress or [getting a free account on WordPress.com](#).

Download WordPress 4.7.3
zip — 8.3 MB
Download tar.gz — 7.6 MB

Release Notification
We've got a handy mailing list that we send a friendly message to whenever there's a new stable release for you to enjoy.

[Join](#)

What's Next?
With our famous 5-minute installation, setting up WordPress for the first time is simple. We've created a [handy guide](#) to see you through the installation process. If you're upgrading your existing installation, we've got a guide for that, too. And should you run into any trouble along the way, our [support forums](#) are a great resource, where seasoned WordPress experts volunteer their time to help you get the most out of your blog.

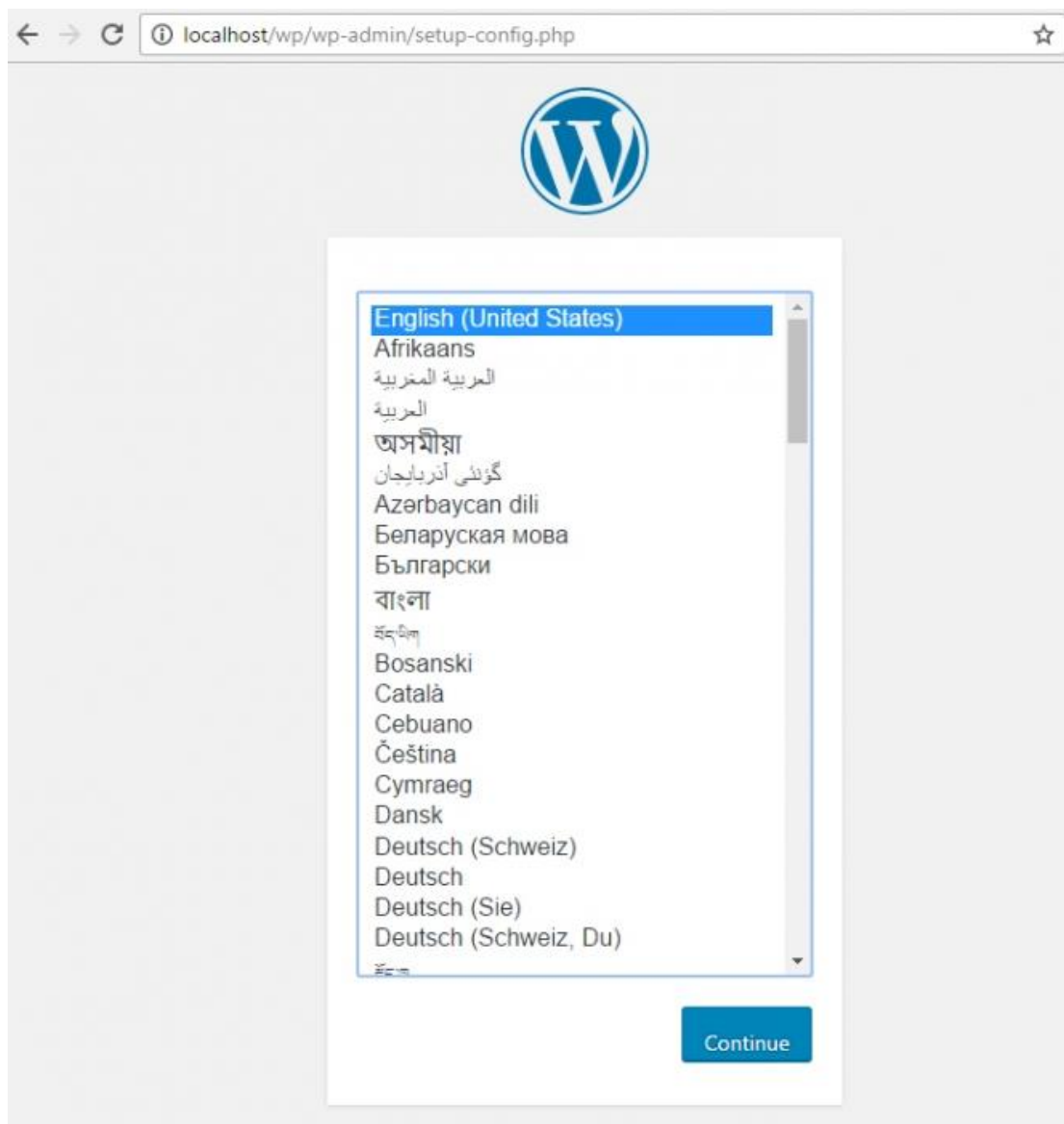
Εικόνα 5. Κατεβάζοντας το WordPress

3.3.3 Εγκατάσταση WordPress

Πρώτα, πρέπει να εξάγουμε το πακέτο του WordPress που έχουμε κατεβάσει, και να το ανεβάσουμε στον Localhost, η στον αντίστοιχο φάκελο του project στον Web Server. Χρησιμοποιώντας έναν browser, πλοηγούμαστε στον φάκελο που βρίσκεται το WordPress (σε περίπτωση Localhost η διεύθυνση θα είναι **http://localhost/wordpress_project_folder**).

Για την εγκατάσταση απαιτείται μια άδεια βάση δεδομένων MySQL, οπότε θα χρειαστεί να τη δημιουργήσουμε εκ των προτέρων. Μετά θα μας ζητηθούν πληροφορίες για την σελίδα, ως εξής:

1. Θα μας ζητηθεί να επιλέξουμε γλώσσα εγκατάστασης



Εικόνα 6. Επιλογή γλώσσας κατά την Εγκατάσταση

2. Σε αυτό το βήμα ο οδηγός εγκατάστασης θα μας παρέχει κάποιες πληροφορίες σχετικά με τα απαιτούμενα στοιχεία πριν προχωρήσουμε.

3. Θα πρέπει να εισάγουμε τα στοιχεία σύνδεσης με την βάση MySQL που έχουμε δημιουργήσει. Θα μας ζητηθούν τα εξής πεδία:
 - Όνομα βάσης δεδομένων.
 - Όνομα χρήστη (Username).
 - Κωδικός (Password).

- Database host. Αυτομάτως θα είναι «localhost»
- Table Prefix. Θα εισάγει πρόθεμα στους πίνακες (tables) της βάσης δεδομένων ώστε να διευκολυνθεί η λειτουργία πολλαπλών site πάνω στην ίδια βάση δεδομένων.

Εικόνα 7. Στοιχεία Βάσης Δεδομένων κατά την Εγκατάσταση

Μόλις εισάγουμε τις απαραίτητες πληροφορίες πατάμε το κουμπί «Submit».

4. Ο οδηγός εγκατάστασης ελέγχει τα στοιχεία που έχουμε εισάγει και μας παρέχει επιβεβαίωση.

Εικόνα 8. Επιβεβαίωση στοιχείων Βάσης Δεδομένων

Εφόσον τα στοιχεία που δώσαμε ήταν σωστά, θα εμφανιστεί το παραπάνω παράθυρο με την κουμπί «Run the Install».

5. Το τελικό βήμα, στο οποίο θα πρέπει να εισάγουμε τα στοιχεία του administrator.
- Τίτλος/όνομα σελίδας
 - Username
 - Password
 - Διεύθυνση E-mail
 - Επιλογή check-box για να επιτραπεί η ευρετηρίαση του site από μηχανές αναζήτησης.

localhost/wp/wp-admin/install.php?language=en_US

Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title

Username
Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password
Strong
Important: You will need this password to log in. Please store it in a secure location.

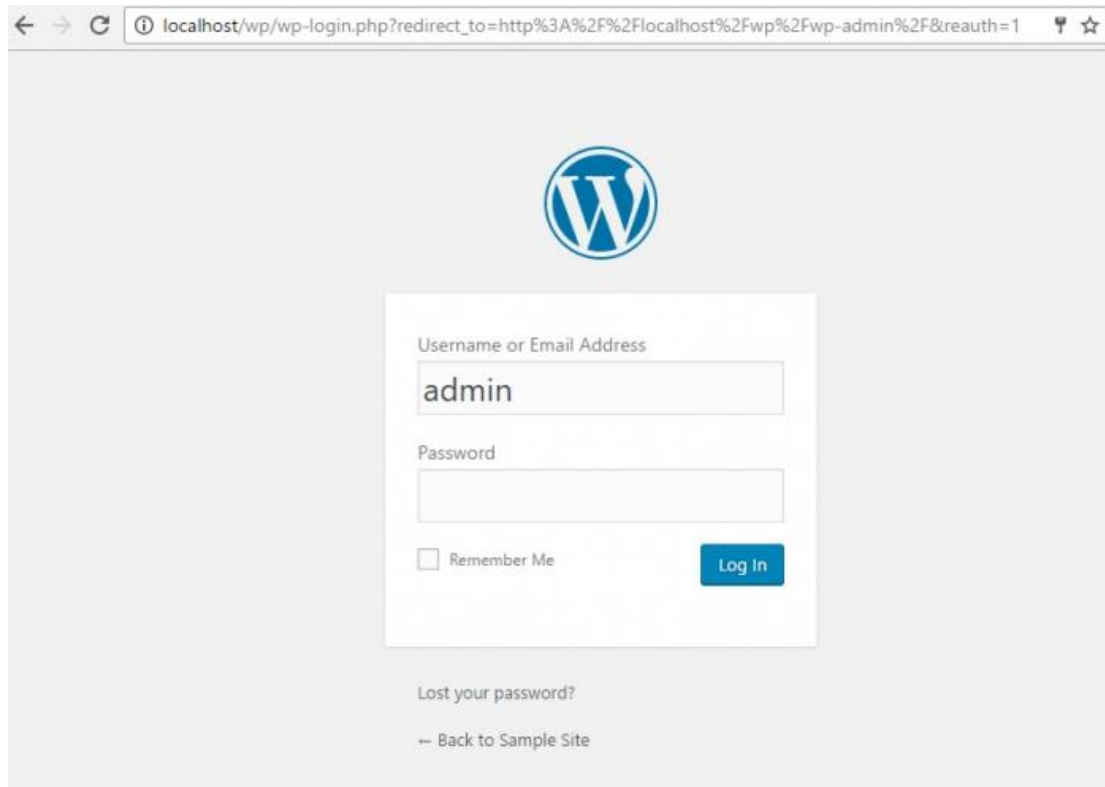
Your Email
Double-check your email address before continuing.

Search Engine Visibility Discourage search engines from indexing this site
It is up to search engines to honor this request.

Εικόνα 9. Στοιχεία Διαχειριστή κατά την Εγκατάσταση

Μόλις συμπληρώσουμε τα στοιχεία πατάμε το κουμπί «Install WordPress».

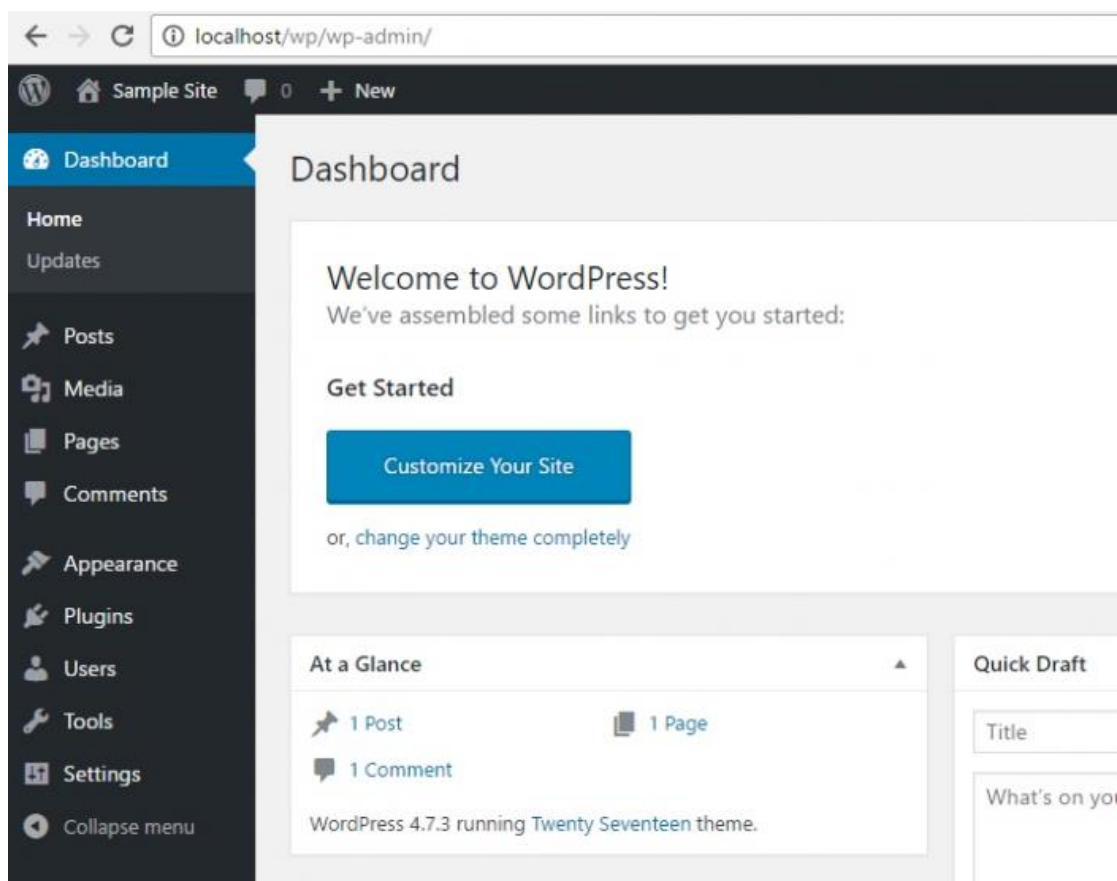
6. Τώρα θα εμφανιστεί η οθόνη για σύνδεση στο admin panel. Εισάγουμε το όνομα χρήστη και τον κωδικό που διαλέξαμε σε προηγούμενο παράθυρο και πατάμε «Log In». Με αυτό ολοκληρώνεται η διαδικασία εγκατάστασης.
- 7.



Εικόνα 10. Η οθόνη εισόδου για τον Πίνακα Ελέγχου Διαχειριστή

3.3.4 Ο πίνακας ελέγχου WordPress

Πίνακας ελέγχου, ή Dashboard, είναι η πρώτη οθόνη που βλέπουμε μετά την σύνδεση στο admin panel. Παρέχει με μια ματιά πληροφορίες για την σελίδα, επιλογές επεξεργασίας και γρήγορη πρόσβαση σε άλλες σελίδες. Αποκαλείται “WP Admin Panel”, και λειτουργεί ως το κέντρο ελέγχου για την κατασκευή και συντήρηση του ιστότοπου μας, και έχει μια πληθώρα χαρακτηριστικών.



Εικόνα 11. Ο Πίνακας Ελέγχου WordPress

- **Top Admin Bar** – Παρέχει συντομεύσεις και συνδέσμους, που αφορούν ενημερώσεις για πρόσθετα, ενημερώσεις του wordpress, σχόλια χρηστών, επιλογές χρήστη κλπ. Είναι διαθέσιμο σε όλες τις σελίδες του WordPress μετά το login.
- **Screen Option** – Η επιλογή αυτή αφορά την παραμετροποίηση της τρέχουσας σελίδας, και μπορεί να περιέχει κάποια checkbox για την εμφάνιση ή απόκρυψη συγκεκριμένων στοιχείων. Ουσιαστικά πρόκειται για μια συλλογή διαφορετικών εργαλείων (widget) που μπορεί να επεξεργασθεί ο χρήστης για να ελέγξει τι είναι ορατό στο Dashboard.
- **Welcome to WordPress** – Ένας τομέας του dashboard με χρήσιμους συνδέσμους και βήματα για να ξεκινήσει κανείς να δουλεύει με το WordPress.
- **At a Glance** – Μια γρήγορη σύνοψη κάποιων δημοσιευμένων σελίδων, άρθρων και άλλων στοιχείων. Επίσης μας ενημερώνει για την τρέχουσα έκδοση του WordPress και πληροφορίες για το ενεργό θέμα / αρχέτυπο.
- **Activity** – Ένα εργαλείο που μας παρέχει γρήγορη πρόσβαση σε πρόσφατες δημοσιεύσεις και σχόλια χρηστών. Μας επιτρέπει να δεχτούμε, αρνηθούμε, να απαντήσουμε, σβήσουμε, να επεξεργαστούμε ένα σχόλιο ή να το χαρακτηρίσουμε ως spam.
- **Quick Draft** – Μας επιτρέπει να γράφουμε και να αποθηκεύσουμε περιεχόμενο από το dashboard. Έρχεται με έναν απλό editor που μας παρέχει βασική λειτουργικότητα

σύνταξης, αποθήκευσης και δημοσίευσης, προσθήκης τίτλων και σημειώσεων η αποθήκευση ως πρόχειρου.

- **WordPress News** – Παρέχει πληροφορίες από το επίσημο Blog του WordPress, σχετικά με νέες εκδόσεις του λογισμικού, ενημερώσεις, νέα και ειδοποιήσεις.

3.3.4.1 *Admin Sidebar*

Στο sidebar βρίσκεται η κυρίως λειτουργικότητα του wordpress. Συμπεριλαμβάνει επιλογές για δημιουργία Post, πολυμέσα, σελίδες, σχόλια, εμφάνιση.

- **Posts** – Αναφέρεται στις εγγραφές δημοσιεύσεων, ως περιεχόμενα σε αντίστροφη χρονολογική σειρά, με το πιο πρόσφατο στην κορυφή. Επίσης, οι δημοσιεύσεις απεικονίζονται και στο RSS feed του Blog. Μπορούμε να ελέγξουμε τον αριθμό των Post που φαίνονται από τις ρυθμίσεις.
- **Pages** – Είναι ένα στατικό στοιχείο περιεχομένου του ιστότοπου. Ένα ευρετήριο σελίδων, και των default του wordpress αλλά και αυτών που έχουμε συντάξει εμείς.
- **Media Library** – Ένας χώρος για την αποθήκευση πολυμεσικών αρχείων (εικόνων, βίντεο κλπ.). Μας επιτρέπει να ανεβάζουμε και να διαχειριζόμαστε πολυμεσικά αρχεία στις σελίδες μας εύκολα και αποτελεσματικά.
- **Comments** – Απεικονίζουν σχόλια από επισκέπτες του ιστότοπου.
- **Appearances** – Χρησιμοποιείται για την παραμετροποίηση της εμφάνισης του ιστότοπου.
- **Plugins** – Ένας τομέας μέσω του οποίου μπορούμε να ενσωματώσουμε επεκτάσεις και πρόσθετο λογισμικό στις σελίδες μας, τα οποία μπορούν να ενισχύσουν τη λειτουργικότητα του web site.
- **User** – Ο τομέας που καθορίζει τις ενέργειες που επιτρέπεται να κάνει ο κάθε χρήστης.
- **Tools** – Μια συλλογή προγραμμάτων για την υλοποίηση εξειδικευμένων εργασιών διαχείρισης.
- **Settings** – Ένας τομέας για τις βασικές ρυθμίσεις της σελίδας.

4 ΔΗΜΙΟΥΡΓΩΝΤΑΣ ΤΗ ΣΕΛΙΔΑ

Το αντικείμενο της πτυχιακής εργασίας είναι η δημιουργία μιας σελίδας Ερευνητικού Εργαστηρίου. Τη στιγμή σύνταξης αυτού του τομέα το αντικείμενο του εργαστηρίου δεν είναι καθορισμένο, ωστόσο οι απαιτούμενες λειτουργίες και λειτουργικότητα είναι γνώστες. Τα κείμενα είναι δοκιμαστικά, και οι εικόνες είναι φωτογραφίες stock, και γενικότερα τα στοιχεία της σελίδας είναι όλα προσωρινά. Η εγκατάσταση και Development πραγματοποιήθηκαν σε υπηρεσία Web-Hosting επί πληρωμή με πάροχο την εταιρία Papaki.com. Η πλατφόρμα διαχείρισης είναι το Plesk Obsidian 18.0.30 (As of Dec. 2020). Χρησιμοποιήθηκε το CMS Wordpress και το Template «Avada».

4.1 ΛΕΙΤΟΥΡΓΙΚΕΣ ΑΠΑΙΤΗΣΕΙΣ

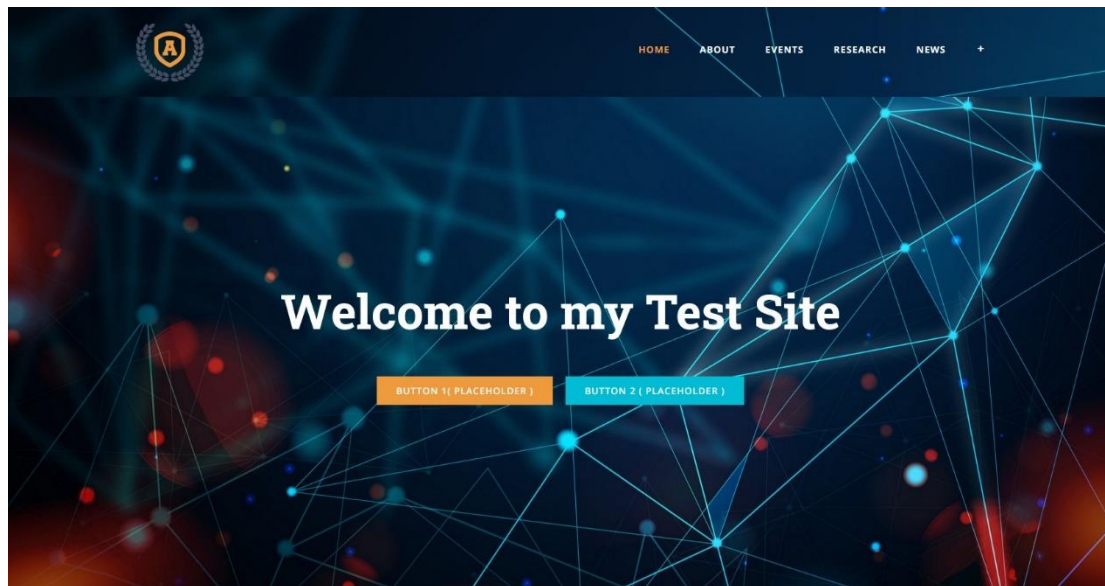
Ο ιστότοπος ενός ερευνητικού εργαστηρίου κατά κύριο λόγο απευθύνεται σε χρήστες του Ακαδημαϊκού χώρου. Ακολούθως, ένας τέτοιος ιστότοπος πρέπει να πληροί κάποιες βασικές προδιαγραφές, ή αλλιώς να διαθέτει συγκεκριμένα χαρακτηριστικά, ως εξής:

- Homepage / Πρώτη σελίδα.
- “About” section / Πληροφορίες για το εργαστήριο.
- Research / Το αντικείμενο του εργαστηρίου.
- Staff / Προσωπικό.
- News and Events/ Νέα και εκδηλώσεις.
- Social media / Πληροφορίες επικοινωνίας.

Πέρα από αυτά τα βασικά χαρακτηριστικά, υπάρχει μια πληθώρα δευτερευόντων χαρακτηριστικών οι οποίες μπορούν να εμπλουτίσουν τη λειτουργικότητα μιας σελίδας ωστόσο δεν είναι απολύτως απαραίτητες για την λειτουργικότητα της σελίδας. Τέτοια χαρακτηριστικά μπορεί να είναι ένας τομέας δημοσιεύσεων, άρθρα μελών προσωπικού, πολυμεσικό περιεχόμενο – γκαλερί εικόνων/βίντεο, τομέας FAQ, Social Media και άλλα.

4.1.1 Πρώτη Σελίδα

Το σημαντικότερο στοιχείο, το πρώτο ερέθισμα που θα λάβει ένας χρήστης που επισκέπτεται την σελίδα. Τυπικά στοιχεία που περιέχει είναι ο τίτλος του εργαστηρίου, ένα Logo, έναν πίνακα περιεχομένων υπό την μορφή Sidebar η Header, μια γρήγορη σύνοψη του αντικειμένου ή/και στόχων του εργαστηρίου, και συνδέσμους στις κυριότερες σελίδες και τομείς του ιστότοπου.



Εικόνα 12. Πρώτη Σελίδα

Η πρώτη μας ματιά στη σελίδα μας δείχνει ένα μήνυμα που καλωσορίζει τον χρήστη, κείμενο σε αντιπαράθεση με μια εικόνα που καταλαμβάνει όλη την οθόνη, το Logo του εργαστηρίου (stock photo/placeholder) πάνω αριστερά και το κυρίως μενού της πλοήγησης πάνω δεξιά.



Short Introduction/ catchy tagline

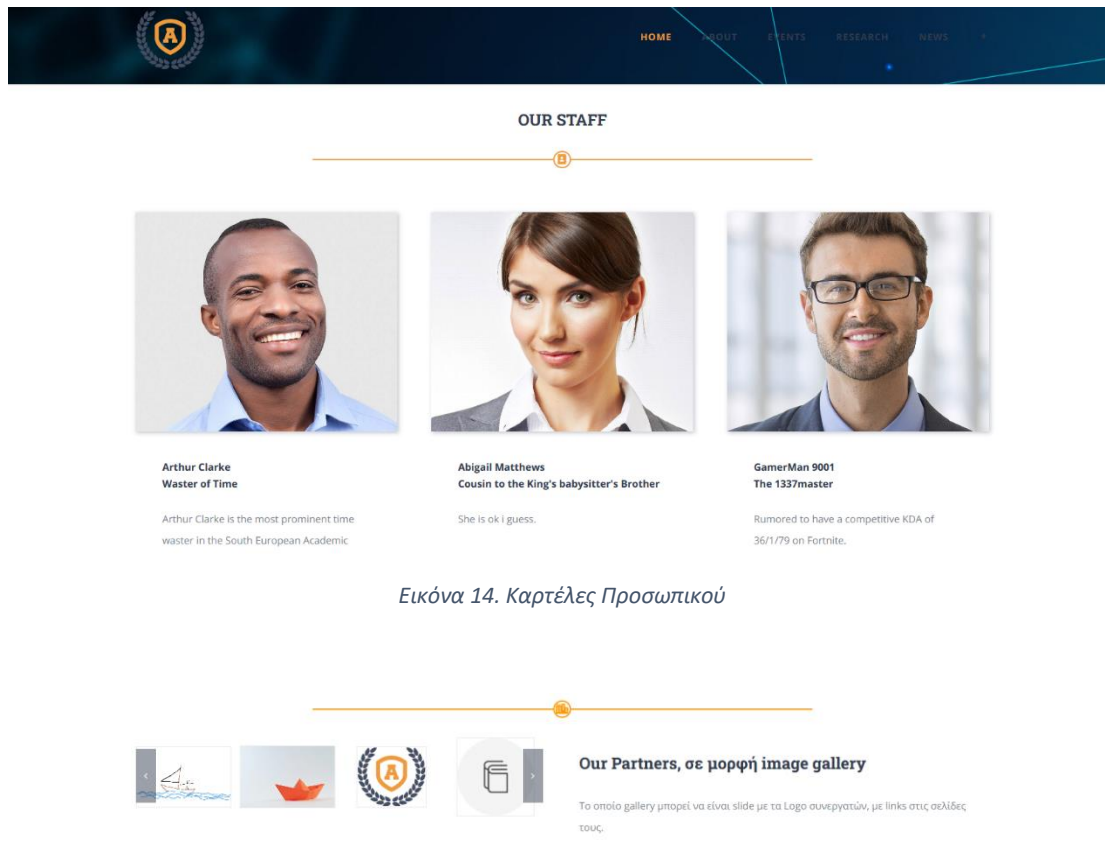
About Us

Title is a link to "about page". This space is intended for a short introduction, with more detail regarding the lab in the About section



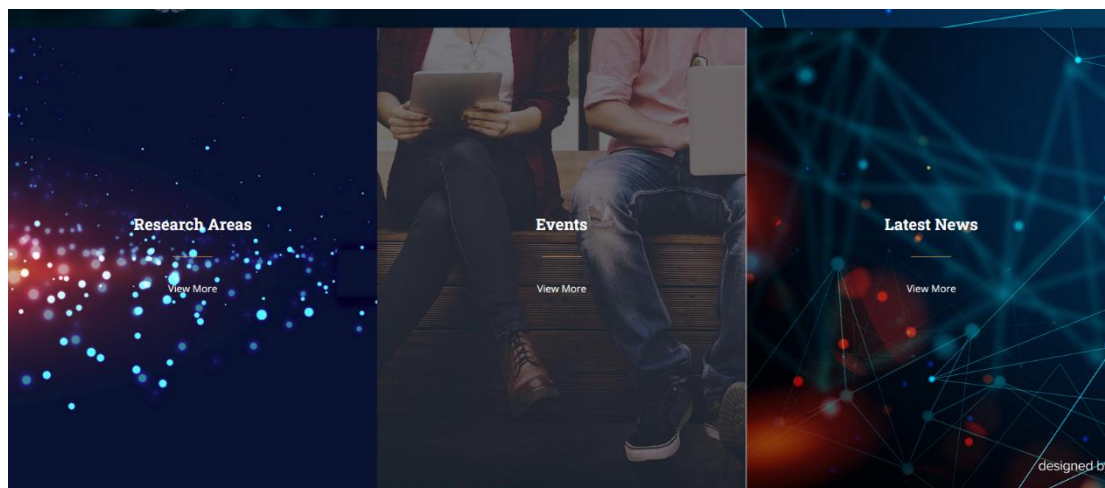
Sample Video

Εικόνα 13. Εισαγωγή πρώτης σελίδας

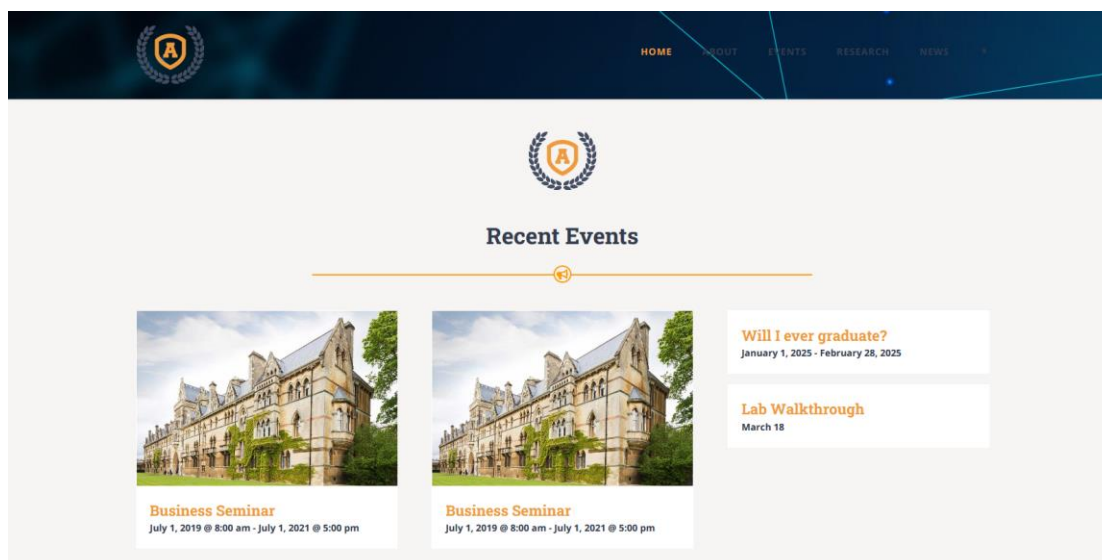


Εικόνα 14. Καρτέλες Προσωπικού

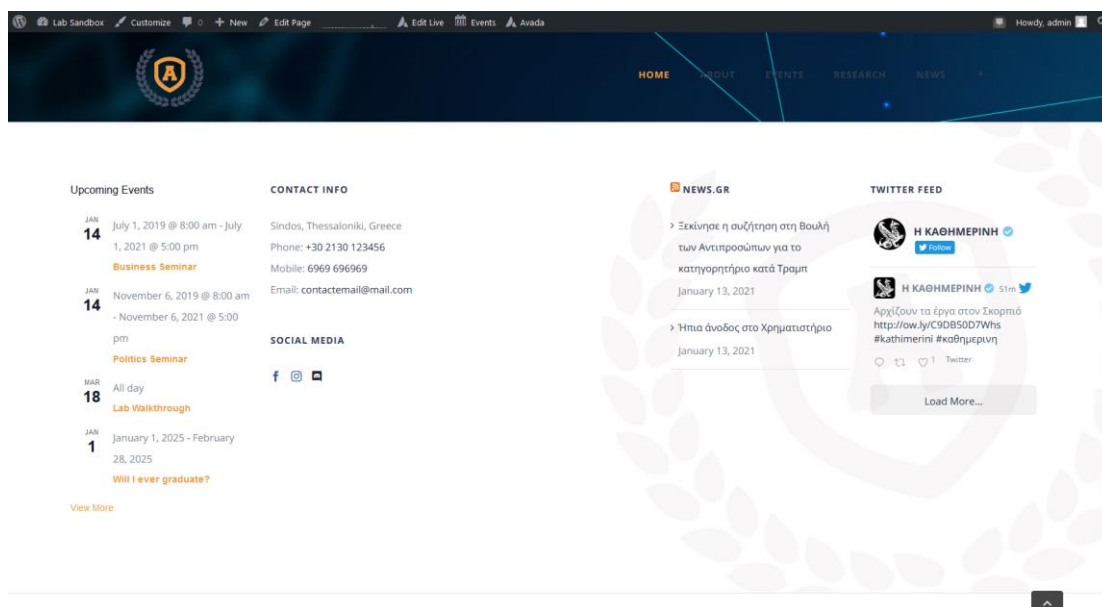
Εικόνα 15. Γκαλερί Εικόνων με Logo Συνεργαζόμενων Φορέων



Εικόνα 16. Σύνδεσμοι προς τους κύριους τομείς του Ιστότοπου



Εικόνα 17. Πρόσφατες Δραστηριότητες



Εικόνα 18. Υποσέλιδο

Συνεχίζοντας προς τα κάτω το κυρίως μενού πλοήγησης παραμένει στην κορυφή και ακολουθεί την κύλιση, και έχουμε έναν τομέα προορισμένο για σύντομη εισαγωγή στο εργαστήριο. Οι πληροφορίες είναι οργανωμένες σε content boxes τα οποία μπορούμε να σύρουμε και να τοποθετήσουμε κατά βούληση σε διάφορα σημεία της σελίδας, και μετά σε αυτά μπορούμε να προσθέσουμε περιεχόμενο, είτε πρόκειται για πολυμέσα, είτε για κώδικα HTML. Χρησιμοποιώντας αυτά τα content boxes, στην πρώτη σελίδα πέραν της εισαγωγής, περιλαμβάνουμε:

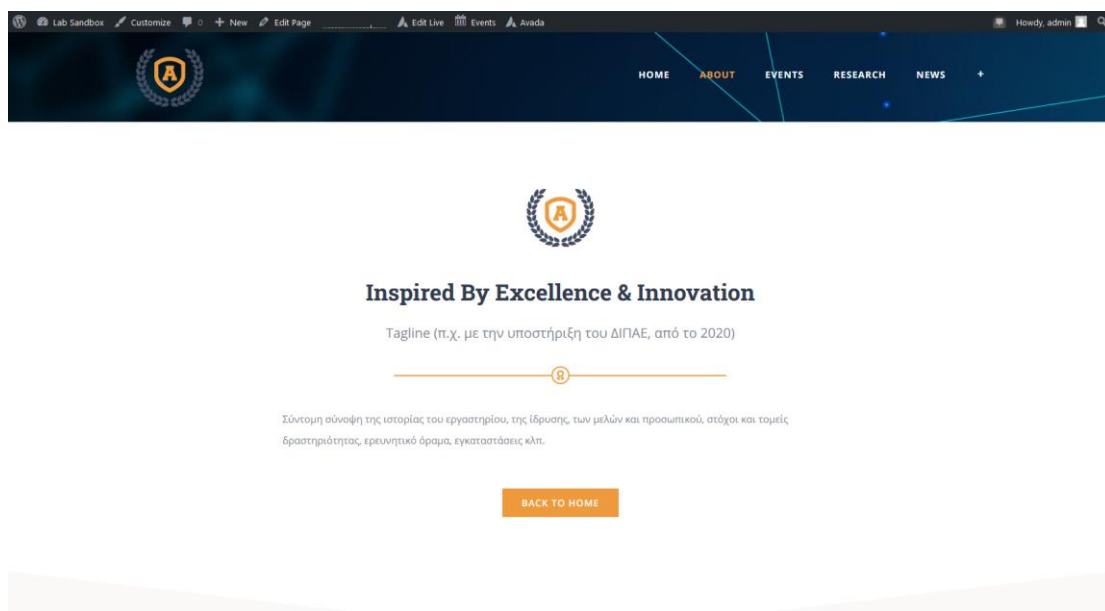
- Καρτέλες Προσωπικού, με σύνδεσμο που οδηγεί σε εξατομικευμένη σελίδα για κάθε μέλος, η οποία θα έχει λεπτομερές βιογραφικό, Contact Info, κ.α.

- Γκαλερί/slider εικόνων που μπορεί να χρησιμοποιηθεί είτε για Logo συνεργατών/συνεργαζόμενων οργανισμών είτε για άλλο σκοπό.
- Μεγάλους συνδέσμους με κείμενο πάνω σε εικόνα προς τους υπόλοιπους κύριους τομείς της σελίδας.
- Τμήμα όπου φαίνονται τα πιο πρόσφατα event, με υποδομή του template και event δημιουργημένα από τον administrator, ταξινομημένα με βάση την ημερομηνία.
- Footer

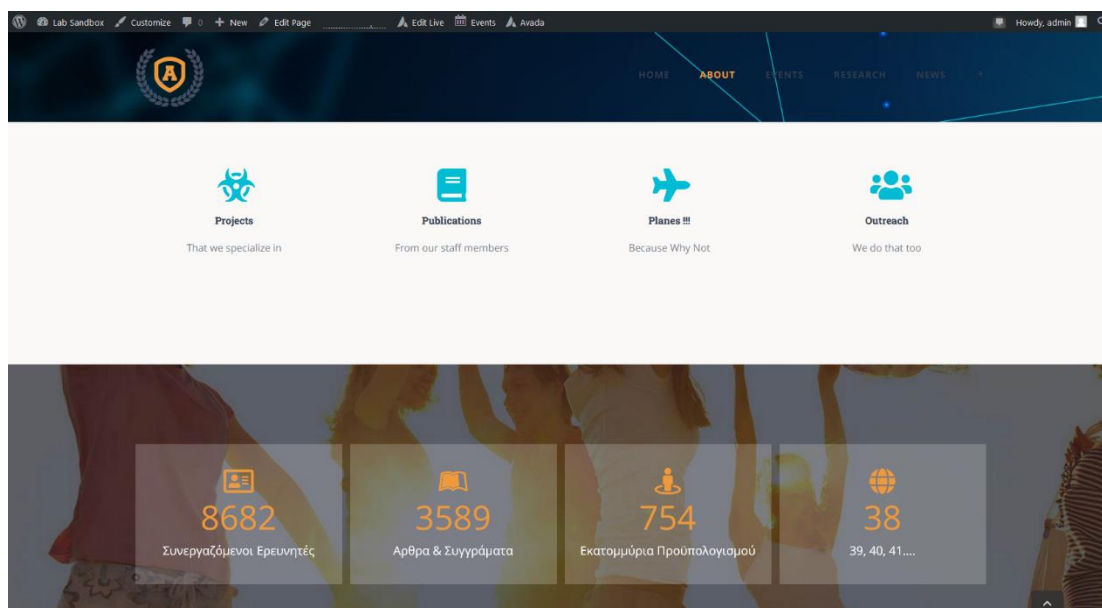
Στο footer έχουν τοποθετηθεί οι πληροφορίες επικοινωνίας, και τα κουμπιά για τα μέσα κοινωνικής δικτύωσης στα οποία δραστηριοποιείται το εργαστήριο. Επίσης υπάρχει ένα RSS feed και ένα Twitter feed, προσωρινά σε λογαριασμούς τρίτων για δοκιμαστικούς λόγους. Τέλος, κάτω δεξιά υπάρχει ένα βέλος το οποίο μας επιστρέφει στην κορυφή της σελίδας, το οποίο είναι λειτουργικότητα του Avada Template.

4.1.2 Σχετικά με το εργαστήριο (About)

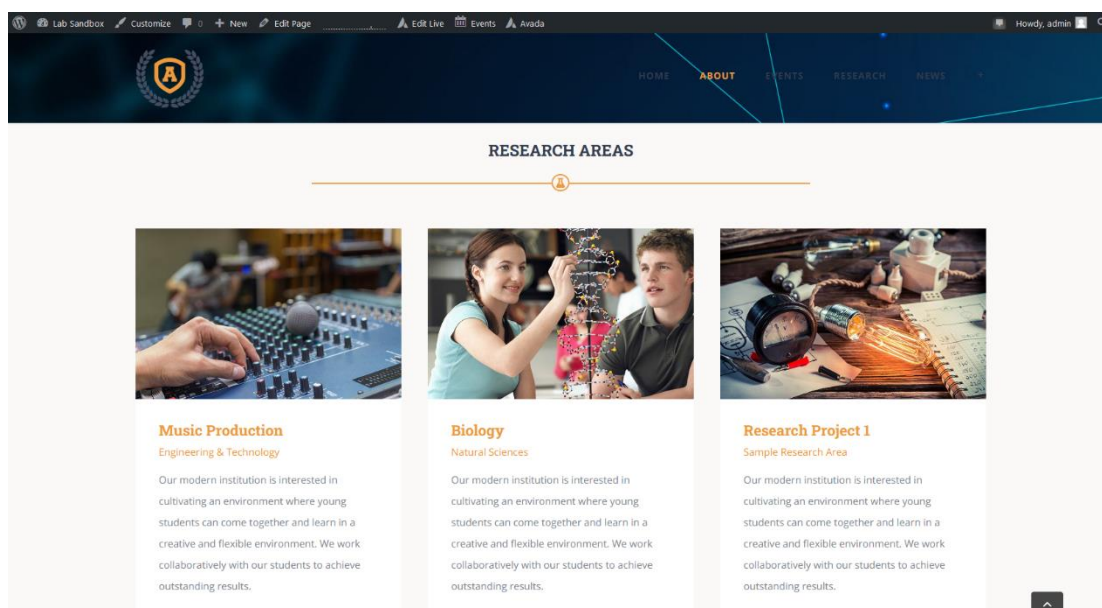
Μία απλή σελίδα που περιγράφει σε μεγαλύτερο βάθος το εργαστήριο. Περιλαμβάνει πληροφορίες για την ιστορία του εργαστηρίου, το αντικείμενο, τις δραστηριότητες και άλλους τομείς. Μπορεί να περιλαμβάνει και ένα FAQ.



Εικόνα 19. Ο τομέας "Σχετικά" (About)



Εικόνα 20. Γραφικές Αναπαράστασεις Στοιχείων/Στατιστικών

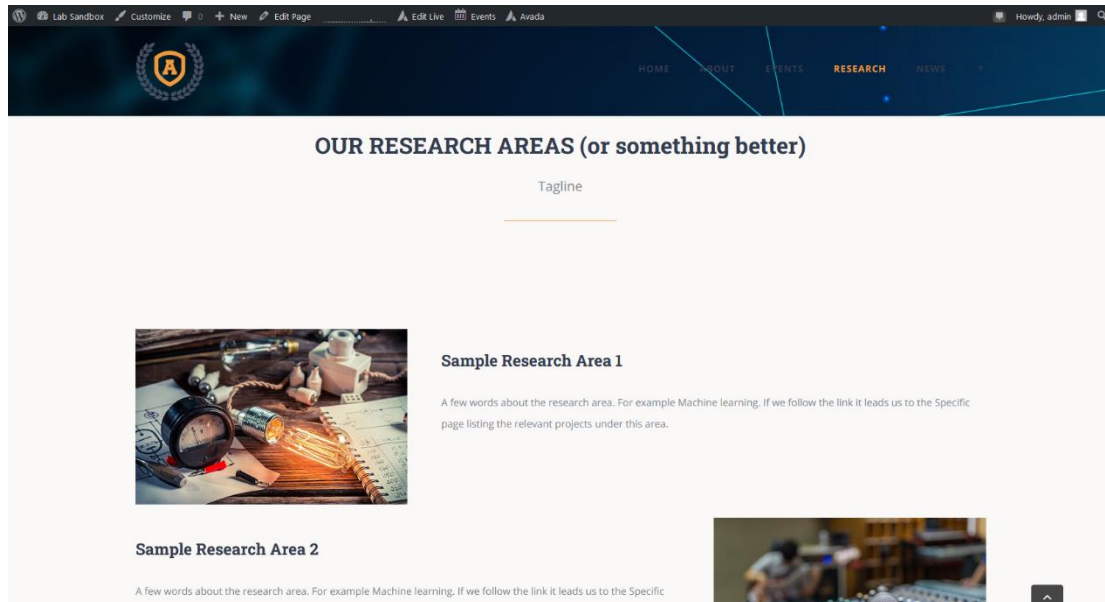


Εικόνα 21. Οι ερευνητικές περιοχές του εργαστηρίου

Σε αυτό τον τομέα της σελίδας πέρα από την λεπτομερή παρουσίαση του εργαστηρίου, χρησιμοποιούμε διάφορα είδη στοιχείων για να απεικονίσουμε στατιστικές για την δραστηριότητα του εργαστηρίου και να προβάλλουμε περαιτέρω το ερευνητικό έργο του. Σε αυτό τον τομέα επίσης θα τοποθετηθεί η υποσελίδα του Προσωπικού, που θα περιέχει όλες τις καρτέλες προσωπικού και συνδέσμους για τις ατομικές σελίδες

4.1.3 Research - Έρευνα

Αυτός ο τομέας περιέχει πληροφορίες για την δραστηριότητα του εργαστηρίου. Τους κύριους τομείς στους οποίους επικεντρώνεται η έρευνα, τα project στα οποία συμμετέχει το προσωπικό του εργαστηρίου, σχετικές δημοσιεύσεις και άρθρα του προσωπικού ή τρίτων. Φιλοξενεί επίσης την υποσελίδα που θα λειτουργήσει ως κατάλογος δημοσιεύσεων.

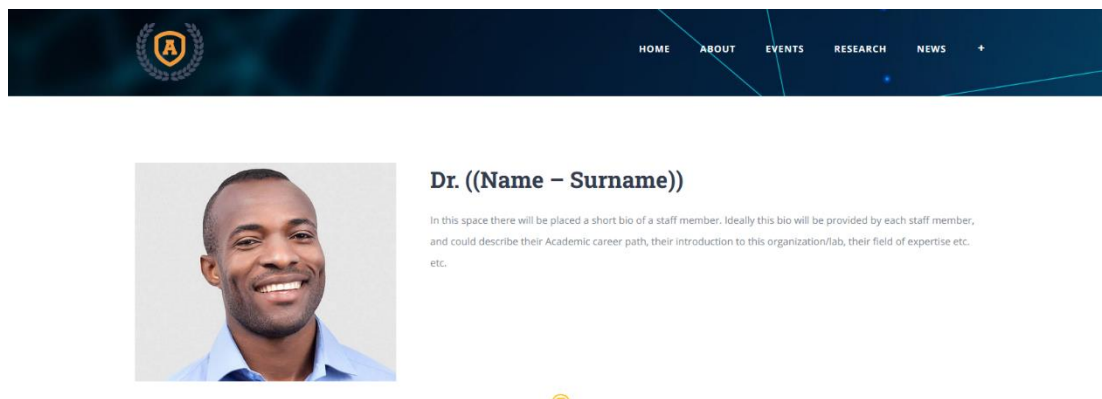


Εικόνα 22. Κύριος κατάλογος ερευνητικών τομέων

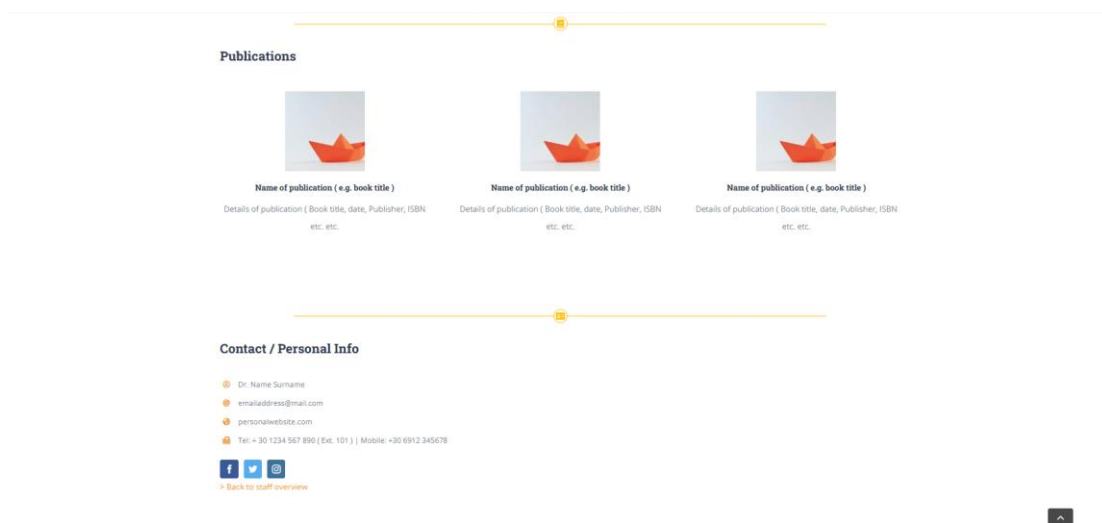
Κάθε ένας από τους ερευνητικούς τομείς (π.χ. ασφάλεια δικτύων) θα οδηγεί σε υποσελίδα στην οποία θα βρίσκονται όλα τα project που ανήκουν σε αυτή την ομάδα. Παράδειγμα ο ερευνητικός τομέας 1 περιλαμβάνει αυτή την στιγμή το πλασματικό Project 1, το οποίο είναι οργανωμένο σε διαφορετικές καρτέλες, με πολυμεσικό υλικό, με υπεύθυνους ερευνητές και με σχετικούς συνδέσμους.

4.1.4 Staff - Προσωπικό

Στην σελίδα μας το προσωπικό είναι υποκατηγορία του About Section. Περιέχει πληροφορίες για τα μέλη του προσωπικού και συνδέσμους για ατομικές υποσελίδες για κάθε μέλος. Κάθε υπο-σελίδα περιέχει ένα σύντομο bio, πληροφορίες επικοινωνίας / μέσων κοινωνικής δικτύωσης και δημοσιεύσεις του μέλους.



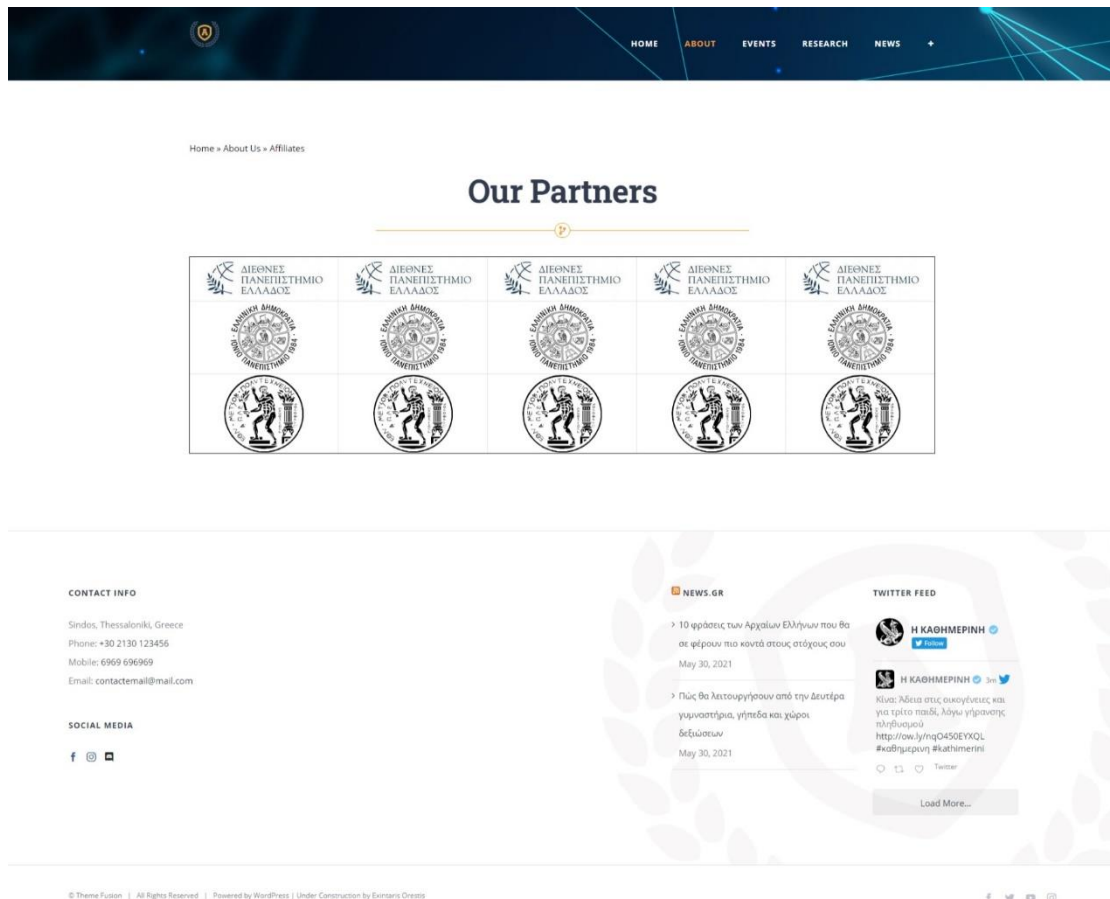
Εικόνα 23. Ατομική Σελίδα Μέλους-Βιογραφικό



Εικόνα 24. Ατομική Σελίδα Μέλους - Δημοσιεύσεις και στοιχεία επικοινωνίας

4.1.5 Affiliates – Συνεργάτες

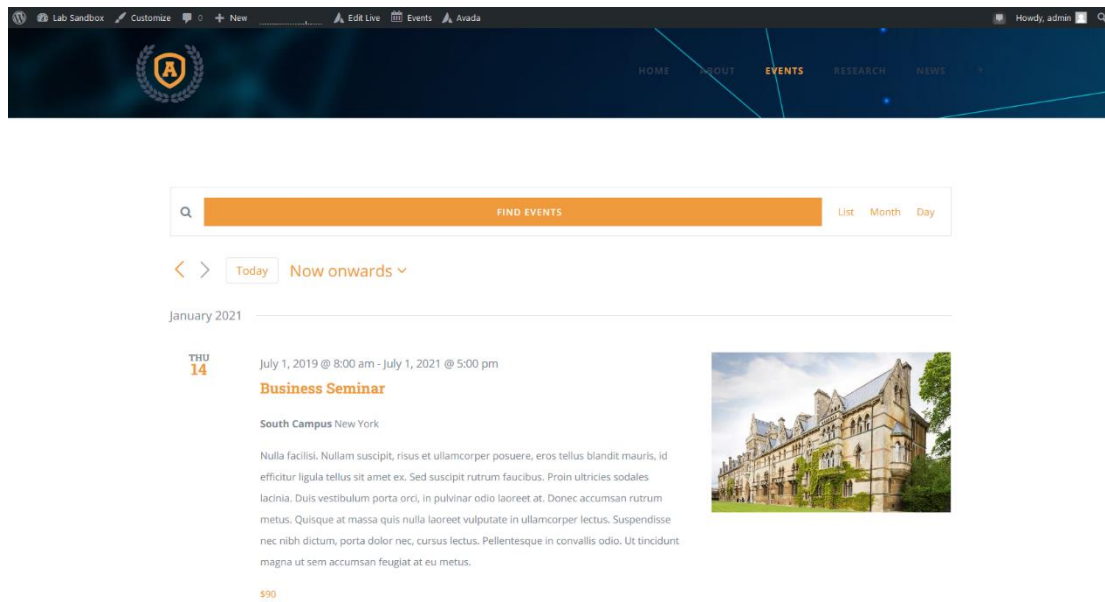
Αυτή η σελίδα περιέχει τους συνεργαζόμενους φορείς και οργανισμούς του εργαστηρίου. Με μια απλή αναπαράσταση εικόνων σε έναν πίνακα HTML, όπου κάθε εικόνα αποτελεί και σύνδεσμο προς τον συγκεκριμένο οργανισμό.



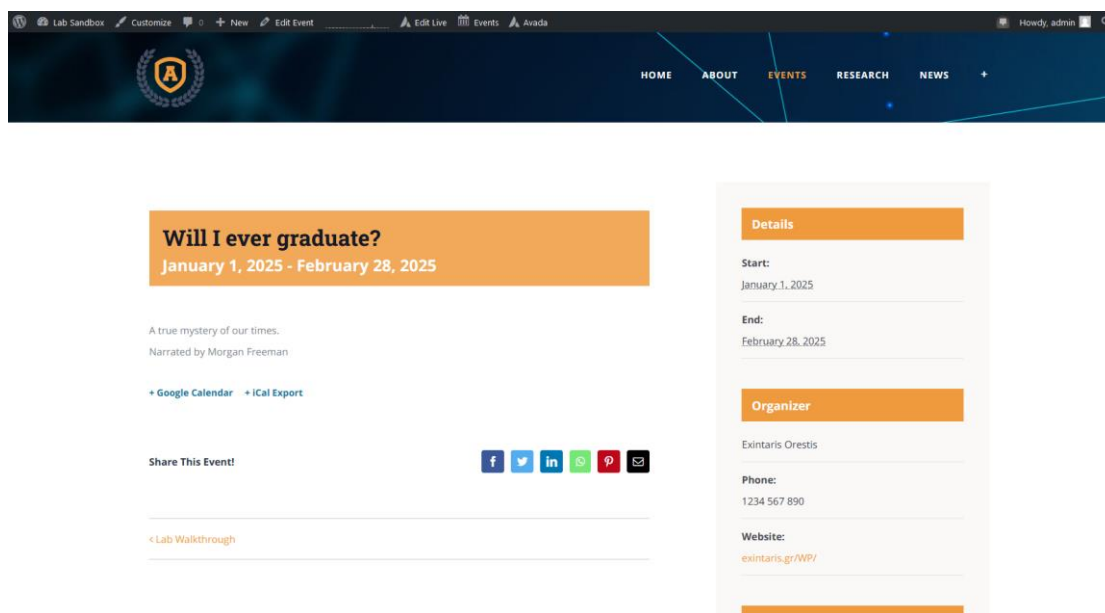
Εικόνα 25. Η σελίδα Συνεργατών

4.1.6 Events – Δραστηριότητες

Στην σελίδα αυτή ο χρήστης μπορεί να αναζητήσει δραστηριότητες με βάση την ημερομηνία, να δει λεπτομέρειες όπως ώρες διεξαγωγής, τοποθεσία, κόστος και άλλες. Επίσης μπορεί να επισκεφτεί τις ατομικές σελίδες για κάθε event, όπου θα βρει λεπτομέρειες για τη διεύθυνση της τοποθεσίας, στοιχεία επικοινωνίας διοργανωτή και να ανακτήσει πληροφορίες σε μορφή αναγνώσιμη από λογισμικό ημερολογίου (π.χ. Outlook) ώστε να μπορεί να εισάγει μια δραστηριότητα στο ημερολόγιο της δικής του συσκευής.



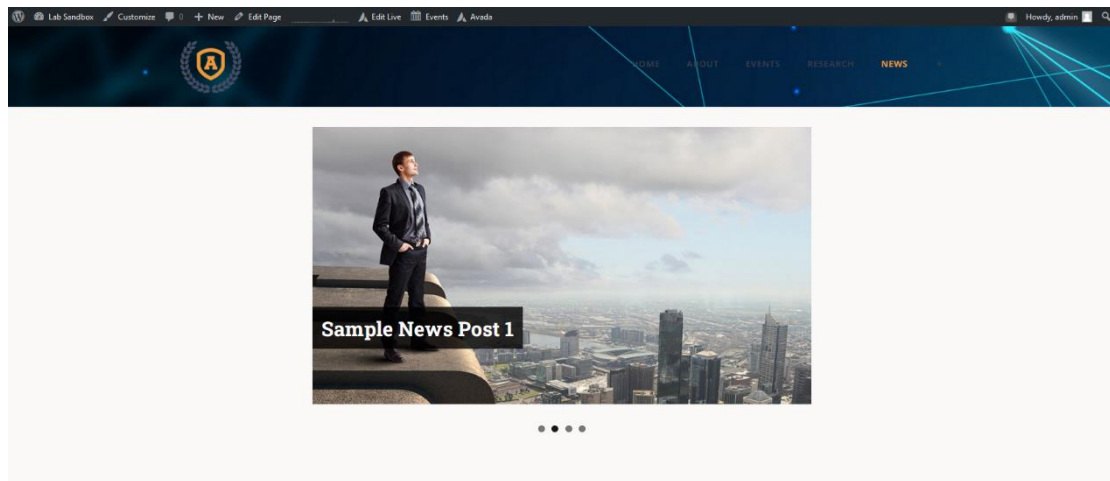
Εικόνα 26. Κύρια Σελίδα Δραστηριοτήτων



Εικόνα 27. Ατομική Σελίδα Δραστηριότητας

4.1.7 Νέα και Ανακοινώσεις

Αυτή η σελίδα περιέχει έναν κατάλογο, μια λίστα από Post με κάποιες βασικές πληροφορίες όπως συντάκτη και ημερομηνία, τίτλο, εικόνα και σύνοψη. Επίσης κάθε είδηση/δημοσίευση έχει σύνδεσμο για μετάβαση σε σχετική υποσελίδα όπου βρίσκεται το πλήρες περιεχόμενο. Στην κύρια σελίδα υπάρχει ένα sidebar με πρόσφατα/δημοφιλή άρθρα/post, ένα Search bar και κουμπιά για κοινοποίηση σε μέσα κοινωνικής δικτύωσης.

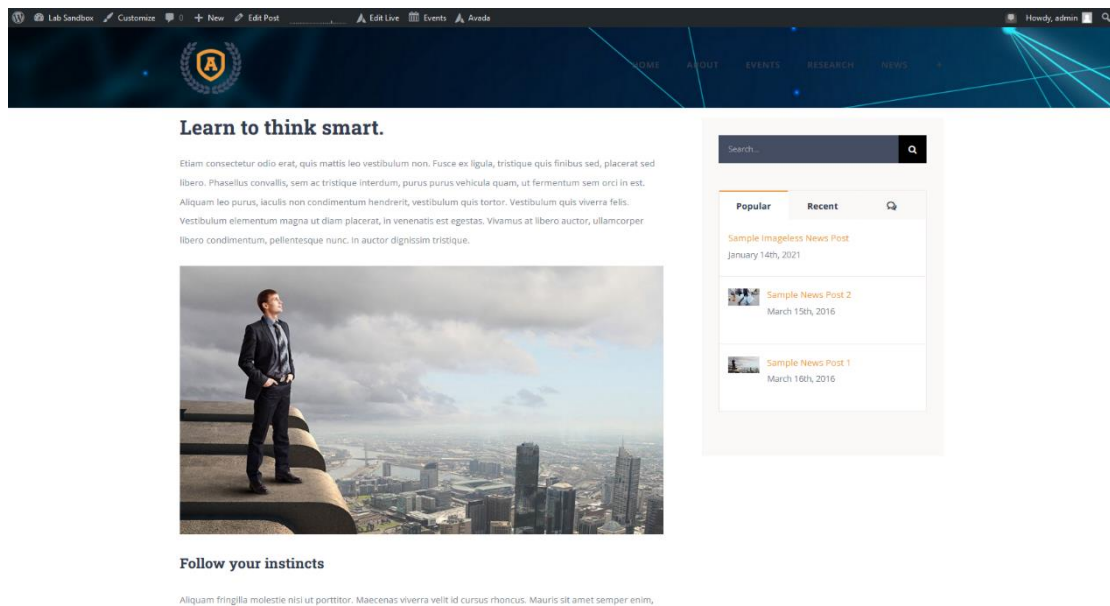


14
01, 2021

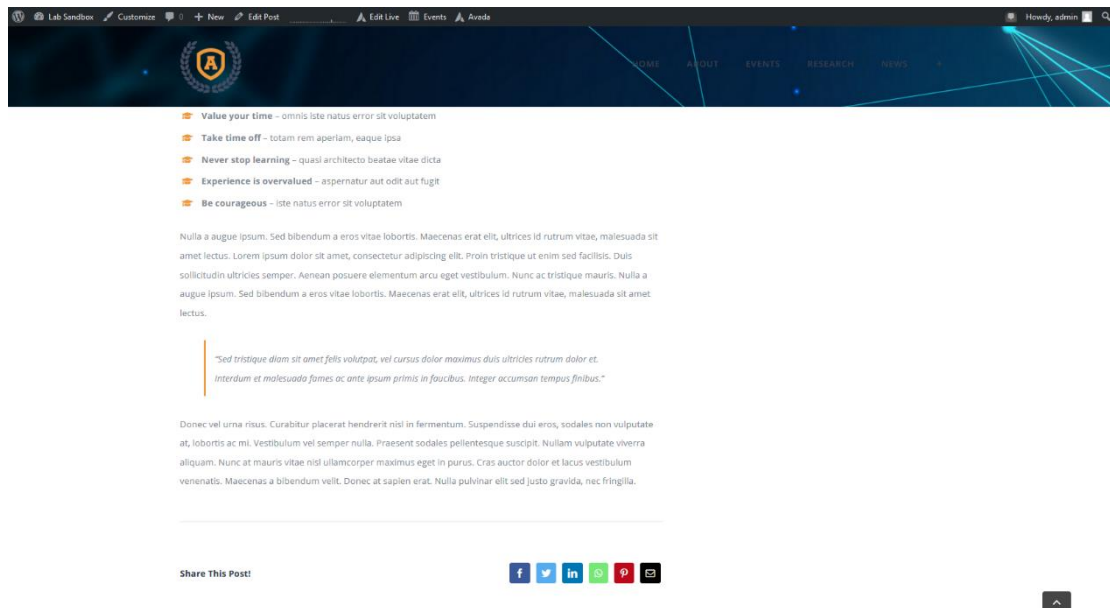
Sample Imageless News Post
By admin | January 14th, 2021 | Categories: Uncategorized

Learn to think smart. Etiam consectetur odio erat, quis mattis leo vestibulum non. Fusce ex ligula, tristique quis finibus sed, placerat sed libero. Phasellus convallis, sem ac tristique interdum, purus purus vehicula quam, ut fermentum sem orci in est. Aliquam leo purus, laculis non condimentum hendrerit, vestibulum quis tortor. Vestibulum quis viverra felis. Vestibulum elementum magna ut diam placerat, in venenatis est egestas. Vivamus at libero auctor, ullamcorper libero [...]

Εικόνα 28. Σελίδα Ανακοινώσεων



Εικόνα 29. Κύρια σελίδα Ανακοίνωσης/Άρθρου, 1/2



Εικόνα 30. Κύρια σελίδα Ανακοίνωσης/Άρθρου, 2/2

5 REPRESENTATIONAL STATE TRANSFER (REST)

5.1 ΕΙΣΑΓΩΓΗ

Στο πλαίσιο της πτυχιακής εργασίας και της κατασκευής σελίδας, θα χρησιμοποιήσουμε την αρχιτεκτονική REST για να προσθέσουμε λειτουργικότητα στον ιστότοπο. Θα προσθέσουμε 3 features αξιοποιώντας το REST API του WordPress αλλά και χτίζοντας το δικό μας API για ξεχωριστές λειτουργίες. Τα features που θα προσθέσουμε θα είναι:

- Με δυνατότητα ενημέρωσης σε πραγματικό χρόνο για αξίες στοιχείων του εργαστηρίου. Εν προκειμένω θα χρησιμοποιηθεί μια πλασματική βάση δεδομένων με αισθητήρες θερμοκρασίας, και ο χρήστης θα μπορεί να ανακτά την τρέχουσα θερμοκρασία ενός αισθητήρα.
- Δυνατότητα φόρτωσης περισσότερων Posts σε μια σελίδα με βάση συγκεκριμένα κριτήρια.
- Δυνατότητα δημιουργίας και δημοσίευσης ανακοινώσεων (WordPress post) από το front end της σελίδας με το πάτημα ενός κουμπιού, με τη χρήση του WordPress REST API.

5.2 ΤΙ ΕΙΝΑΙ Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ REST:

Rest (Representational State Transfer) εκφράζει ένα στυλ αρχιτεκτονικής για την παροχή προδιαγραφών με σκοπό τη διευκόλυνση της επικοινωνίας μεταξύ συστημάτων υπολογιστών στο διαδίκτυο [17]. Συστήματα συμβατά με πρακτικές REST (Restful Systems) έχουν ως κύρια χαρακτηριστικά τον διαχωρισμό client – server και το ότι είναι «stateless».

5.2.1 Διαχωρισμός Client – Server.

Στο αρχιτεκτονικό στυλ της REST η εφαρμογή του client και του server μπορούν να γίνουν ανεξάρτητα και χωρίς ο ένας να γνωρίζει για τον άλλο. Αυτό σημαίνει ότι ο κώδικας στη μεριά του client μπορεί να αλλάξει οποτεδήποτε χωρίς να επηρεάσει τη λειτουργία του server και αντιστρόφως. Από τη στιγμή που κάθε μεριά γνωρίζει τί μορφή πρέπει να έχουν τα μηνύματα που ανταλλάσσουν μεταξύ τους, μπορούν να παραμένουν ευέλικτα και χωριστά. Διαχωρίζοντας το User Interface από περιορισμούς σχετικά με αποθήκευση δεδομένων βελτιώνουμε την ευελιξία του Interface σε διάφορες πλατφόρμες καθώς και την επεκτασιμότητα της δομής στον Server. Ο διαχωρισμός αυτός επιτρέπει σε κάθε ένα από τα 2 αυτά στοιχεία να εξελίσσεται ανεξάρτητα.

5.2.2 Statelessness

Το Stateless πρωτόκολλο απαιτεί ότι δεν θα υπάρχει ενεργό session (application state) στον server. Κατά συνέπεια, κάθε αίτηση περιέχει όλες τις απαραίτητες πληροφορίες για να γίνει κατανοητή. Το πλεονέκτημα του πρωτοκόλλου αυτού είναι η βελτίωση της ορατότητας, αξιοπιστίας και επεκτασιμότητας.

- Η ορατότητα βελτιώνεται γιατί ένα σύστημα δεν χρειάζεται να εξετάσει στοιχεία πέραν της ίδιας της αίτησης για να κατανοήσει το πλήρες φάσμα της.
- Η αξιοπιστία βελτιώνεται γιατί διευκολύνεται η διαδικασία ανάκαμψης από τυχόν σφάλματα.
- Η επεκτασιμότητα βελτιώνεται γιατί δεν υπάρχει πλέον η ανάγκη να διατηρούνται πληροφορίες σύνδεσης / session data για κάθε αίτηση, έχοντας ως αποτέλεσμα ο server να μπορεί να απελευθερώσει πόρους γρηγορότερα. Περαιτέρω, απλοποιείται πολύ η εφαρμογή καθώς ο server δεν χρειάζεται να διαχειριστεί την χρήση πόρων για πολλές διαφορετικές αιτήσεις [18].

5.2.3 Επικοινωνία Client – Server

Με βάση την αρχιτεκτονική REST ένας client θα στείλει αιτήσεις για να ανακτήσει ή τροποποιήσει πληροφορίες, και ένας server απαντά στις αιτήσεις αυτές. Μία αίτηση REST σχετίζεται με λειτουργίες CRUD (Create, Read, Update, Delete), που είναι και οι 4 βασικοί τύποι δυνατοτήτων που θέλουμε να προσφέρει ένα REST API. Αντιστοιχίζοντας τις με τα ρήματα HTTP που θα χρησιμοποιήσουμε ορίζονται ως εξής.

- GET – Χρησιμοποιείται για την ανάκτηση δεδομένων, όμοια με την Read.
- POST – Χρησιμοποιείται για την δημιουργία νέων εγγραφών, όμοια με την Create.
- PUT – Χρησιμοποιείται για την ενημέρωση εγγραφών, όμοια με την Update.
- DELETE - Χρησιμοποιείται για την διαγραφή δεδομένων, ίδια με την Delete.

5.3 REST API

Ένα API (Application Programming Interface) ορίζεται ως: “An interface or communication protocol between a client and a server intended to simplify the building of client-side software” [19]. Ένα API επιτρέπει την επικοινωνία μεταξύ υπηρεσιών/προϊόντων χωρίς να απαιτείται γνώση για την εφαρμογή του κάθε ενός. Παρομοιάζονται συχνά με συμβόλαια, αντιπροσωπεύοντας μια συμφωνία μεταξύ διαφορετικών οντοτήτων. Εάν η οντότητα #1 κάνει μια αίτηση δομημένη με έναν συγκεκριμένο τρόπο, ορίζεται το πώς θα απαντήσει στην αίτηση μια οντότητα #2 [20].

5.3.1 JSON

«JSON (JavaScript Object Notation) είναι ένα ανοικτό μορφότυπο το οποίο χρησιμοποιεί κείμενο, που μπορεί να διαβαστεί από τον άνθρωπο, για τη μετάδοση αντικειμένων δεδομένων που αποτελούνται από ζεύγη χαρακτηριστικών-τιμών και τύπου δεδομένων συστοιχιών (ή οποιασδήποτε άλλης σειριοποιήσιμης τιμής). Πρόκειται για ένα πολύ κοινό μορφότυπο

δεδομένων που χρησιμοποιείται για την ασύγχρονη επικοινωνία περιήγηση-διακομιστή, συμπεριλαμβανομένης της αντικατάστασης για XML σε μερικά συστήματα τύπου AJAX». [21]

Το JSON είναι χτισμένο πάνω σε 2 δομές [22]:

- Μια συλλογή από ζευγάρια ονόματος/αξίας. Σε πολλές γλώσσες εμφανίζεται ως αντικείμενο (object).
- Μια ταξινομημένη λίστα αξιών. Σε πολλές γλώσσες εμφανίζεται ως Array ή Vector.

Όλες οι σύγχρονες γλώσσες προγραμματισμού (π.χ. JavaScript, C#, Python etc.) καθώς και πλατφόρμες εφαρμογών υποστηρίζουν ευρέως την παραγωγή και ανάλυση δεδομένων JSON. [23]

5.3.2 Το REST API του WordPress

Το WordPress Rest API παρέχει σε εφαρμογές έναν τρόπο αλληλεπίδρασης με τον WordPress ιστότοπο μας, μέσω της αποστολής και λήψης δεδομένων σε μορφή JSON αντικειμένων. Παρέχει Rest Endpoints (URL 's) τα οποία αντιπροσωπεύουν όλα τα στοιχεία δεδομένων του Wordpress (Post, pages etc.). Μια εφαρμογή μπορεί να ανταλλάξει δεδομένα JSON σε αυτά τα URL για να ζητήσει πληροφορίες, να τροποποιήσει στοιχεία ή να δημιουργήσει περιεχόμενο στον ιστότοπο. [24]

Το REST API του WordPress συμπεριλήφθηκε στο Wordpress Core στην έκδοση 4.7 το 2016, και είναι σχεδιασμένο για να υποστηρίξει διάφορες εφαρμογές κατασκευασμένες σε WordPress και να μεταμορφώσει το WordPress από Content Management System σε πλατφόρμα εφαρμογών.

5.4 ΕΦΑΡΜΟΓΗ #1 - ΔΗΜΙΟΥΡΓΙΑ ΕΝΟΣ PHP REST API

Στα πλαίσια της εργασίας θα προσθέσουμε λειτουργικότητα στον ιστότοπο με τη δημιουργία ενός Rest API, με την χρήση HTML, PHP και JSON. Έχουμε δημιουργήσει μια πλασματική βάση δεδομένων με αισθητήρες θερμοκρασίας, η οποία έχει σε έναν πίνακα SQL πληροφορίες για το αναγνωριστικό ενός αισθητήρα (Sensor ID), την τοποθεσία στην οποία βρίσκεται ο αισθητήρας (Location), και την τιμή της θερμοκρασίας που καταγράφει (Temperature). Το REST API θα συμπεριλαμβάνει την σύνδεση στη βάση δεδομένων, και θα περιμένει client request, περνώντας έπειτα τις παραμέτρους του request σε ένα SQL query και επιστρέφοντας ένα JSON response. Για να το αναλώσουμε θα δημιουργήσουμε μια ιστοσελίδα με ένα web form, και θα χρησιμοποιήσουμε JSON για την μετάφραση και παρουσίαση της πληροφορίας.

5.4.1 Κατασκευάζοντας το API

Ξεκινώντας θα πρέπει να ορίσουμε στο Header τον τύπο της πληροφορίας που θα πρέπει να περιμένει ο Browser. Σε αυτή την περίπτωση ο τύπος είναι JSON.

```
<?php
header("Content-Type:application/json");
```

Εικόνα 31. Δήλωση JSON ως τύπος περιεχομένου

Έπειτα (με την προϋπόθεση ότι μας έχει γίνει POST το Sensor ID) συνδεόμαστε στη βάση δεδομένων καλώντας το αρχείο PHP που περιέχει τα στοιχεία σύνδεσης.

```
if (isset($_GET['SensorID']) && $_GET['SensorID']!="") {
    include('db.php');
```

Εικόνα 32. Σύνδεση στη βάση δεδομένων

Από τη στιγμή που είμαστε συνδεδεμένοι στη βάση δεδομένων και έχουμε την παράμετρο με βάση την οποία μας ζητούνται πληροφορίες, περνάμε το Sensor ID σε ένα SQL query και διενεργούμε Select στην βάση δεδομένων. Ελέγχουμε πως οι σειρές (rows) της βάσης δεν είναι άδειες, και εάν δεν υπάρχουν σφάλματα , αποθηκεύουμε το αποτέλεσμα και κλείνουμε τη σύνδεση.

```
$sensorid = $_GET['SensorID'];
$result = mysqli_query($con,"SELECT * FROM `Sensors` WHERE SensorID=$sensorid");

if(mysqli_num_rows($result)>0){
    $row = mysqli_fetch_array($result);
    $Location = $row['Location'];
    $Temperature = $row['Temperature'];
    response($sensorid, $Location, $Temperature);
    mysqli_close($con);
}else{
    response(404, "No Record Found", 404);
}
}else{
    response(NULL, "Invalid Request", NULL);
}
```

Εικόνα 33. Ερώτημα προς τη βάση δεδομένων.

Χρησιμοποιήσαμε τα στοιχεία που πήραμε από την βάση για να καλέσουμε μια μέθοδο, την 'response'. Στην μέθοδο αυτή κωδικοποιούμε τις πληροφορίες και επιστρέφουμε αποτέλεσμα JSON. Και το API είναι έτοιμο για ανάλωση.

```
function response($sensorid, $Location, $Temperature){
    $response['SensorID'] = $sensorid;
    $response['Location'] = $Location;
    $response['Temperature'] = $Temperature;

    $json_response = json_encode($response);
    echo $json_response;
}
?>
```

Εικόνα 34. Η συνάρτηση κωδικοποίησης πληροφορίας σε μορφή JSON

5.4.2 Αναλώνοντας το API

Για να αναλώσουμε (consume) το API θα δημιουργήσουμε μια νέα σελίδα στον ιστότοπο. Η σελίδα θα περιέχει ένα απλό web form που θα δέχεται το Sensor ID ως Input και θα επιστρέφει το αποτέλεσμα του SQL query.



Εικόνα 35. Η σελίδα "Sensor Data" πριν την εισαγωγή δεδομένων.

Σε ένα HTML code block έχουμε τοποθετήσει την φόρμα η οποία κάνει POST το Sensor ID:

```
<body>

<form action="" method="post">
<label>Enter Sensor ID:</label><br />
<input type="text" name="SensorID" placeholder="Enter Sensor ID" required/>
<br /><br />
<button type="submit" name="submit">Submit</button>
</form>
```

Εικόνα 36. Φόρμα για την αίτηση του Sensor ID

Στη συνέχεια, ελέγχουμε εάν όντως έχει γίνει POST η παράμετρος που ζητάμε, και την κάνουμε append σε μια μεταβλητή URL, που ορίζει την τοποθεσία του REST API.

```
<?php
if (isset($_POST['SensorID']) && $_POST['SensorID']!="") {
    $sensorid = $_POST['SensorID'];
    $url = "https://exintaris.gr/api.php?SensorID=".$sensorid;
```

Εικόνα 37. Προσθήκη του ληφθέντος Sensor ID στην αίτηση προς το API

Για την σύνδεση και το πέρασμα παραμέτρων στο URL του REST API, θα χρησιμοποιήσουμε CURL. Πρόκειται για μια βιβλιοθήκη (libcurl) που υποστηρίζεται από την PHP και διευκολύνει την σύνδεση και επικοινωνία με διάφορους τύπους server και πρωτοκόλλων, όπως http, https, ftp κ.α.

```

$client = curl_init($url);
curl_setopt($client, CURLOPT_RETURNTRANSFER, true);
$response = curl_exec($client);

```

Εικόνα 38. Αρχικοποίηση CURL

Με την εντολή 'curl_init(\$url)' αρχικοποιούμε ένα curl Session, και παίρνουμε ένα curl_handle το οποίο χρησιμοποιούμε στην επόμενη σειρά. Με την εντολή 'curl_setopt' παραμετροποιούμε την μεταφορά της πληροφορίας. Η παράμετρος 'CURLOPT_RETURNTRANSFER' δηλώνεται ως TRUE, που σημαίνει ότι επιστρέφουμε την αξία ως String για να εκτελεσθεί η μεταφορά με την χρήση της εντολής 'curl_exec' όπου και εκτελούμε το session με βάση το handle (το url) και τα κριτήρια που έχουμε θέσει.

Τέλος, αφού έχουμε λάβει το response (που είναι σε μορφή JSON) το αποκωδικοποιούμε και το παρουσιάζουμε με τη βοήθεια ενός απλού HTML table.

```

$result = json_decode($response);

echo "<table>";
//echo "<tr><td>SensorID: </td><td></td><td>$result->sensorid</td></tr>";
echo "<tr><td>Location: </td><td></td><td>$result->Location</td></tr>";
echo "<tr><td>Temperature: </td><td></td><td>$result->Temperature</td></tr>";
echo "</table>";
}
?>
</body>

```

Εικόνα 39. Αποκωδικοποίηση και παρουσίαση της πληροφορίας

Sensor Data

Enter Sensor ID

Location: Thessaloniki

Temperature: 24

Εικόνα 40. Η σελίδα Sensor Data μετά την απάντηση

Οπότε έχουμε ένα API στον ιστότοπο μας που είναι έτοιμο να απαντήσει σε αιτήσεις πληροφορίας όχι μόνο από τη σελίδα μας, αλλά από οπουδήποτε.

5.5 ΕΦΑΡΜΟΓΗ #2 – ΦΟΡΤΩΣΗ POSTS ΜΕ AJAX REQUESTS

Για τη δεύτερη εφαρμογή, θα αξιοποιήσουμε το REST API του Wordpress. Σε μια ιστοσελίδα News posts θα χρησιμοποιήσουμε ένα AJAX 'GET' Request για να φορτώσουμε περισσότερα POST, με κριτήριο τα tags τους τα οποία θα τα περάσουμε ως παραμέτρους στο URL. Με το πάτημα ενός κουμπιού τα φορτώνουμε σε ένα HTML div .

5.5.1 Αίτηση AJAX

AJAX σημαίνει Asynchronous JavaScript And XML. Πρακτικά, πρόκειται για τη χρήση του αντικειμένου XMLHttpRequest για την επικοινωνία με έναν Server. Μπορεί να ανταλλάξει πληροφορίες σε πολλές διαφορετικές μορφές , συμπεριλαμβανομένων JSON, XML, HTML και αρχείων κειμένου. Το κύριο πλεονέκτημά του είναι η ασύγχρονη φύση του, που σημαίνει πως μπορεί να επικοινωνήσει με τον Server, να ανταλλάξει δεδομένα και να ενημερώσει την ιστοσελίδα χωρίς να χρειάζεται ανανέωσή της. [25]

5.5.2 Δημιουργία και διασύνδεση του χώρου εργασίας (Workspace)

Για να υλοποιήσουμε την εφαρμογή μας θα πρέπει να χρησιμοποιήσουμε JavaScript. Θα δημιουργήσουμε αρχείο που θα περιέχει τον κώδικά μας στο Directory του ενεργού WordPress Theme, και θα το συμπεριλάβουμε (include) από το αρχείο «functions.php».

```
function my_custom_script() {
    wp_enqueue_script('main', get_template_directory_uri().'/js/main.js', NULL, 0.1, true);
    wp_localize_script('main', 'identifier', array(
        'nonce' => wp_create_nonce('wp_rest')
    ));
}
add_action('wp_enqueue_scripts', 'my_custom_script');
```

Εικόνα 41. Σύνδεση του αρχείου με τις μεθόδους JavaScript μας

Σε μία νέα μέθοδο my_custom_script() χρησιμοποιούμε την εντολή «wp_enqueue_script»,

```
wp_enqueue_script( $handle, $src, $deps, $ver, $in_footer ); Όπου:
```

- \$handle: Το όνομα του script.
- \$src: Πλήρες URL του script, είτε το μονοπάτι του σε σχέση με το root directory του Wordpress. Για να αποφύγουμε Hardcoding του URL χρησιμοποιούμε την μέθοδο «get_template_directory_uri()» για να πάρουμε το root του wordpress και προσθέτουμε το μονοπάτι του script.
- \$deps: Ένας πίνακας με άλλα script handle από τα οποία εξαρτάται το main script μας (dependencies). Ορίζεται ως NULL εν προκειμένω καθώς δεν έχουμε dependencies.
- \$ver: Η έκδοση του script.

- `$in_footer`: Δηλώνουμε πως θέλουμε να φορτωθεί το script στο footer της σελίδας.

Και ολοκληρώνουμε τη διασύνδεση (hook) με την εντολή «`add_action`». Οι 2 άλλες γραμμές στην μέθοδο αφορούν το 3^ο Feature/ εφαρμογή και θα εξετασθούν αργότερα. Τώρα που έχει συνδεθεί το script file μας μπορούμε να γράψουμε τον κυρίως κώδικα του Feature.

5.5.3 Δημιουργώντας την εφαρμογή

Αρχικά θα προσθέσουμε ένα κουμπί στην σελίδα η οποία θα φιλοξενήσει την εφαρμογή, με το πάτημα του οποίου θα ενεργοποιείται το AJAX Request, και ένα DIV στο οποίο θα εμφανίζονται τα ζητούμενα Post.

```
<button id="postsButton"> Load additional posts </button>
<div id="postsContainer"></div>
```

Εικόνα 42. Δημιουργία Κουμπιού και Div που θα περιέχει το αποτέλεσμα

Τώρα που έχουμε την βασική υποδομή, μπορούμε να γράψουμε τον κώδικα που θα διαχειριστεί το event και θα ενεργοποιήσει το AJAX Request.

```
var postsButton = document.getElementById("postsButton");
var postsContainer = document.getElementById("postsContainer");

if (postsButton) {

    postsButton.addEventListener("click", function() {
        var Request = new XMLHttpRequest();
        Request.open('GET', 'https://exintaris.gr/wp/wp-json/wp/v2/posts?tags=50&per_page=2&order=desc');
        Request.onload = function() {
            if (Request.status >= 200 && Request.status < 400) {
                var data = JSON.parse(Request.responseText);
                createHTML(data);
                postsButton.remove();
            } else {
                console.log("Error");
            }
        };

        Request.onerror = function() {
            console.log("Connection error");
        };

        Request.send();
    });
}
```

Εικόνα 43. Η συνάρτηση-διαχειριστής του πατήματος κουμπιού

Αρχικά δημιουργούμε 2 μεταβλητές που αναφέρονται στο κουμπί και στο DIV. Ελέγχουμε εάν το κουμπί υπάρχει (καθώς δεν θα βρίσκεται σε κάθε σελίδα του ιστότοπου), και εάν υπάρχει φορτώνουμε ένα Listener για τη διαχείριση του «Click» event. Κάνουμε ένα νέο AJAX ‘GET’ Request με δεδομένα JSON, με παραμέτρους ένα συγκεκριμένο tag/κατηγορία post που θέλουμε να εμφανιστούν, και μορφοποίηση για 2 post ανά σελίδα σε φθίνουσα ταξινόμηση.

5.5.3.1 Προβολή συγκεκριμένων Δημοσιεύσεων με χρήση Tags

Για να γίνει πιο εύχρηστο το feature θα πρέπει να δώσουμε στον χρήστη την δυνατότητα να επιλέξει το είδος των δημοσιεύσεων θέλει να δει. Για τον σκοπό αυτό θα προσθέσουμε ένα Drop-down list στην σελίδα, με επιλογές για προβολή όλων των Δημοσιεύσεων, η συγκεκριμένων κατηγοριών με βάση τα tag τους. Σε αυτό το στάδιο θα χρησιμοποιήσουμε τις κατηγορίες “Announcements” και “Articles”.

```
<label for="tagList">Choose post Category</label>
<select id="tagList" name="tagList">
  <option value="All">All posts</option>
  <option value="51">Announcements</option>
  <option value="52">Articles</option>
</select>
```

Εικόνα 44. Δημιουργία της λίστας κατηγοριών

Θα προσθέσουμε επίσης στον κώδικα μια μεταβλητή που θα περιέχει την αξία του Selected Index της λίστας, και θα υλοποιούμε το AJAX request με βάση την επιλογή του χρήστη.

```
//declared with other global variables
var tagList = document.getElementById("tagList");

//declared in the click Event Handler function
var tag = tagList.options[tagList.selectedIndex].value;
```

Εικόνα 45. Ανακτώντας την επιλογή του χρήστη από την λίστα

Στη συνέχεια, ελέγχουμε εάν το Request έχει διεκπεραιωθεί (αν είναι μέσα στην αποδεκτή εμβέλεια error status, από 200 OK έως 400 bad request error). Εάν έχει προχωρήσει καλούμε την μέθοδο createHTML η οποία δέχεται την απάντηση του server σε μορφή JSON.

```
function createHTML(postsData) {
  var HTMLString = '';
  for (i = 0; i < postsData.length; i++) {
    HTMLString += '<h2>' + postsData[i].title.rendered + '</h2>';
    HTMLString += postsData[i].content.rendered;
  }

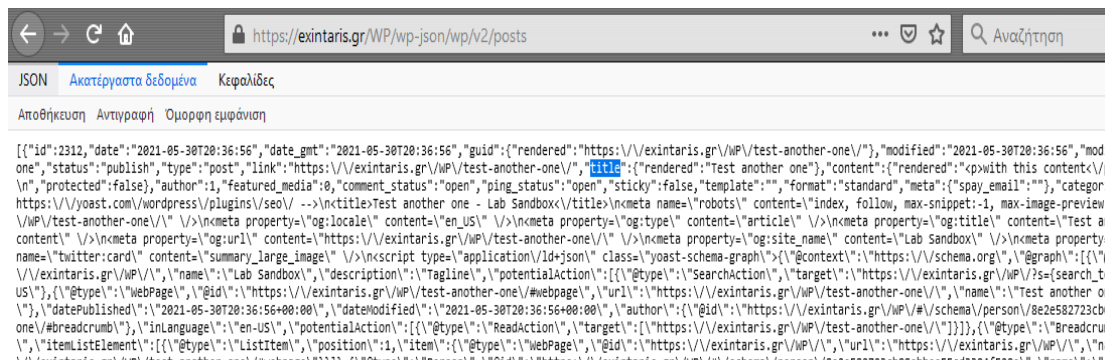
  postsContainer.innerHTML = HTMLString;
}
```

Εικόνα 46. Η συνάρτηση createHTML για την δημιουργία της δημοσίευσης

Για να καταφέρουμε να παρουσιάσουμε τα Post στον χρήστη, θα πρέπει να ξέρουμε που βρίσκεται το περιεχόμενό τους και ο τίτλος τους. Για να το μάθουμε αυτό θα πρέπει να ζητήσουμε πληροφορίες JSON από το Wordpress, με το παρακάτω URL.

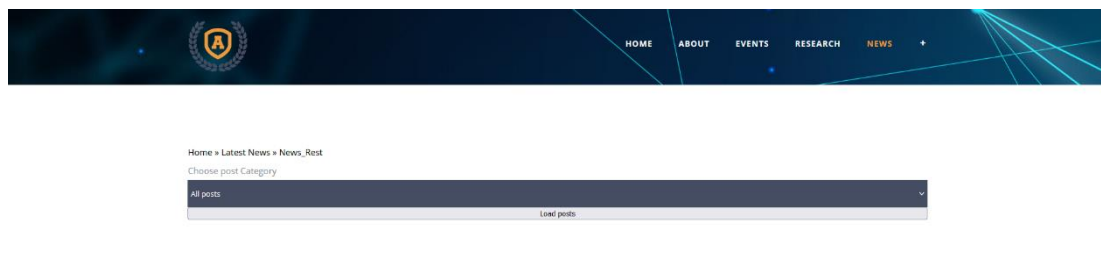
‘https://exintaris.gr/WP/wp-json/wp/v2/posts’

Σε αυτό το URL το Wordpress θα μας επιστρέψει μορφοποιημένα τα στοιχεία όλων των Post (ή ότι επιλέξουμε να κάνουμε query, π.χ. media). Μπορούμε να δούμε επίσης και ακατέργαστα τα δεδομένα για να βρούμε ακριβώς που βρίσκεται η πληροφορία που αναζητούμε.

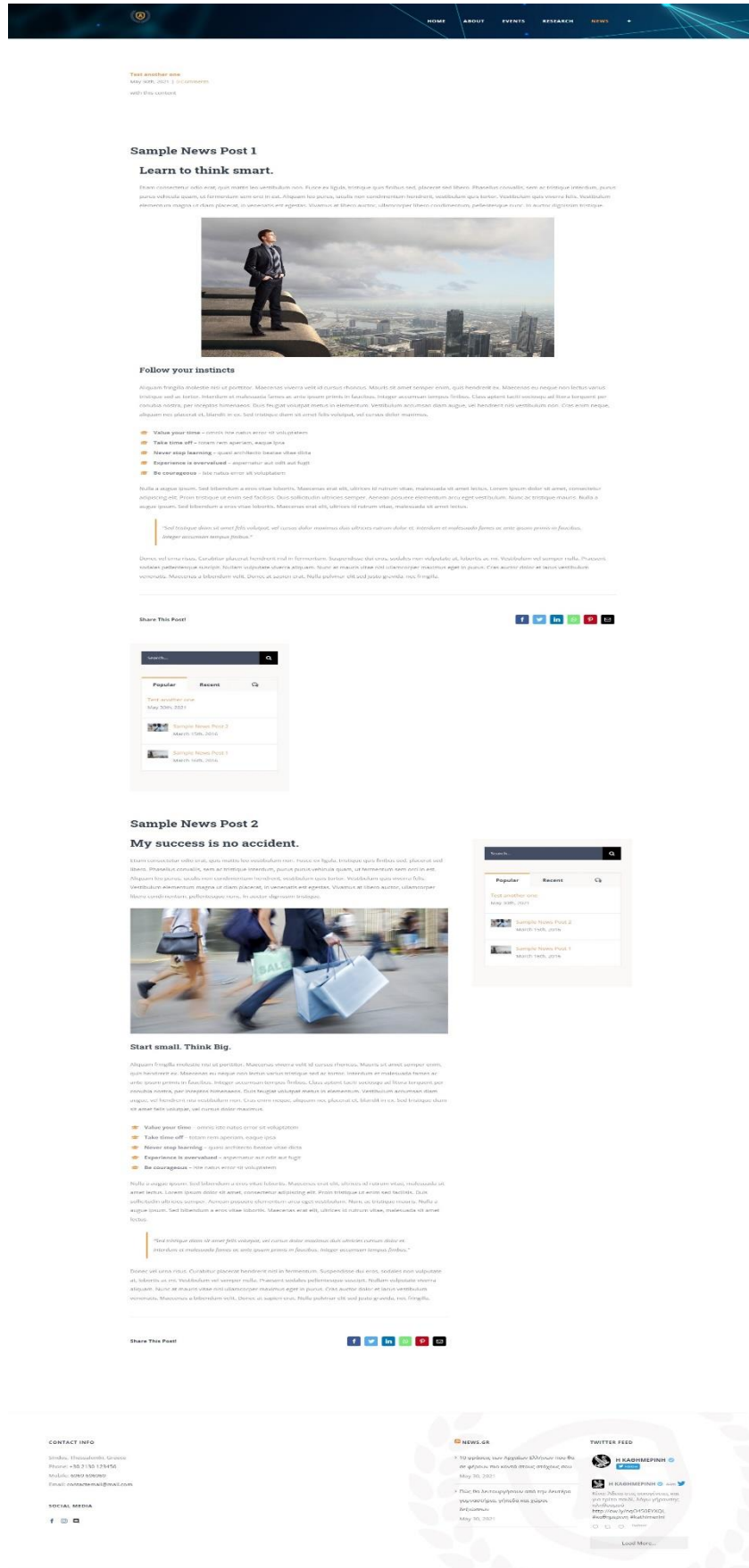


Εικόνα 47. Ακατέργαστη πληροφορία JSON

Εάν αναζητήσουμε για title θα παρατηρήσουμε πως ο τίτλος που χρειαζόμαστε βρίσκεται εμφωλευμένος μέσα σε μια ιδιότητα «rendered», που με τη σειρά της είναι μέσα στο «title». Ομοίως και για το content. Οπότε χρησιμοποιούμε την μέθοδο createHTML για να μορφοποιήσουμε αναλόγως την απάντηση JSON της AJAX Request που είχαμε διαπράξει, και το εισάγουμε στο DIV που έχουμε δημιουργήσει, για κάθε post που επέστρεψε το query.



Εικόνα 48. Η σελίδα πριν την επιλογή του χρήστη



Εικόνα 49. Η σελίδα μετά την φόρτωση Δημοσιεύσεων

5.6 ΕΦΑΡΜΟΓΗ #3 – ΓΡΗΓΟΡΗ ΔΗΜΙΟΥΡΓΙΑ POST ΑΠΟ ΤΟ FRONT-END ΤΗΣ ΣΕΛΙΔΑΣ

Η Τρίτη εφαρμογή που θα κατασκευάσουμε είναι πρακτικά ένα ‘POST’ request στο WordPress REST API, όπου ένας χρήστης με προνόμια Administrator θα μπορεί να δημιουργήσει και να δημοσιεύσει κατευθείαν μια ανακοίνωση/post χωρίς να χρειάζεται να μπει στο back end του WordPress. Θα χρησιμοποιήσουμε το αρχείο script το οποίο συνδέσαμε με το wordpress κατά την κατασκευή της Δεύτερης Εφαρμογής. Ο χώρος εφαρμογής θα είναι μια υποσελίδα ανακοινώσεων, με 2 πεδία για είσοδο κειμένου, και ένα κουμπί Submit.

5.6.1 Κατασκευάζοντας το Front-End

```

1 <?php if (current_user_can(administrator)) : ?>
2 <div class="main_column">
3     <div class="admin_quick_add">
4         <h3>Quick Add Post</h3>
5         <input type="text" name="title" placeholder="Title">
6         <textarea name="content" placeholder="Content"></textarea>
7         <button id="add_button"> Create Post </button>
8     </div>
9 </div>
10
11 <?php endif; ?>

```

Εικόνα 50. Δημιουργία περιοχών εισαγωγής για την δημιουργία Δημοσίευσης

Πρώτα ελέγχουμε την δικαιοδοσία του χρήστη, έπειτα δημιουργούμε τα πεδία Input. Για λόγους εμφάνισης διευκρινίζουμε διαφορετικά class ώστε να μπορούμε να μορφοποιήσουμε τα DIV με CSS. Custom CSS εισάγουμε κατευθείαν από τα options του Avada.

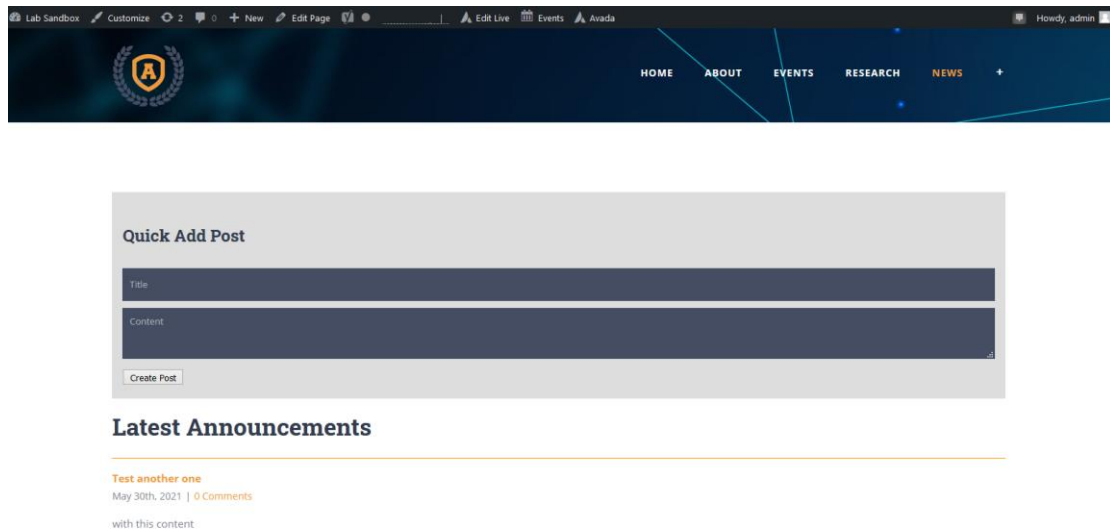
```

.admin_quick_add {
    background-color: #DDD;
    padding: 15px;
    margin-bottom: 15px;
}

.admin_quick_add input,
.admin_quick_add textarea {
    width: 100%;
    border: none;
    padding: 10px;
    margin: 0 0 10px 0;
    box-sizing: border-box;
}

```

Εικόνα 51. Μορφοποίηση CSS



Εικόνα 52. Η σελίδα προσθήκης δημοσίευσης (ως Admin)

5.6.2 Κατασκευάζοντας το Back-End

Αφού έχουμε ετοιμάσει το input area τώρα θα πρέπει να διαχειριστούμε το submit.

```
var QuickAddButton = document.querySelector("#add_button");

//If the button exists on the page, create an event listener for the click event.
if (QuickAddButton) {
  QuickAddButton.addEventListener("click", function(){
    //Create a new variable postData and store the user input in it. Set status to publish to skip draft stage of the post.
    var postData = {
      "title": document.querySelector('.admin_quick_add [name="title"]').value,
      "content": document.querySelector('.admin_quick_add [name="content"]').value,
      "status": "publish"
    }
  })
}
```

Εικόνα 53. Η συνάρτηση-διαχειριστής του πατήματος για την δημιουργία δημοσίευσης

Αρχικά επιλέγουμε το κουμπί και το ορίζουμε σε μια μεταβλητή. Ελέγχουμε για την ύπαρξη του κουμπιού στη σελίδα και εάν υπάρχει δημιουργούμε ένα Event Listener για την διαχείριση του Click. Ορίζουμε ένα αντικείμενο postData που περιέχει τον τίτλο και το περιεχόμενο που εισήγαγε ο χρήστης στα πεδία Input. Προσθέτουμε στο τέλος την ιδιότητα «status». Την ορίζουμε ως «publish» ώστε το post να προσπεράσουμε την φάση του draft και το Wordpress να δημοσιεύσει το Post κατευθείαν.

```
//Send a POST request to the api, notifying it that you want to create a new post.
var createPost = new XMLHttpRequest();
createPost.open('POST', 'https://exintaris.gr/WP/wp-json/wp/v2/posts');
```

Εικόνα 54. Αίτηση "POST" προς το WordPress API

Για να στείλουμε την αίτηση POST θα χρησιμοποιήσουμε το JavaScript object postData. Θα μετατρέψουμε σε αναγνώσιμη μορφή και θα ενημερώσουμε εκ των προτέρων τον Server πως θα πρέπει να ερμηνεύσει την πληροφορία σε JSON.

```
//Parse data to JSON and send it to create the post
createPost.setRequestHeader("Content-Type", "application/json;charset=UTF-8");
createPost.send(JSON.stringify(postData));
```

Εικόνα 55. Μετατροπή του αντικειμένου σε JSON string και αποστολή

Υπάρχει ακόμα ένα βήμα που πρέπει να ολοκληρώσουμε για να διεκπεραιωθεί η αίτησή μας. Το Wordpress, ως βασικό στρώμα ασφάλειας δεν πρόκειται να ανταποκριθεί σε μια αίτηση POST εάν δεν προέρχεται από συνδεδεμένο χρήστη. Ωστόσο, είναι δυνατόν να συνδεθεί κάποιος στο Wordpress αλλά να κάνει αίτηση σε μια σελίδα που δεν του ανήκει. Για να αποφευχθούν τυχόν Cross-Site exploits, το Wordpress απαιτεί επιπροσθέτως έναν μοναδικό κωδικό που αντιστοιχεί στον συνδεδεμένο χρήστη και πρέπει να σταλεί μαζί με το AJAX Request. Οπότε πριν εκτελέσουμε send() θα πρέπει να περάσουμε και αυτό τον κωδικό.

```
//Authentication Check
createPost.setRequestHeader("X-WP-Nonce", identifier.nonce);
```

Εικόνα 56. Περνώντας το μοναδικό αναγνωριστικό χρήστη στο WordPress API

Δεν μπορούμε να μαντέψουμε τον κωδικό για να τον περάσουμε ως παράμετρο, αλλά μπορούμε να χρησιμοποιήσουμε PHP echo , για να «ρίξουμε» τον κωδικό στη σελίδα μας και να τον καλέσουμε μέσω JavaScript. Θα πρέπει να επιστρέψουμε στην μέθοδο “**my_custom_script**” που είχαμε κατασκευάσει στο feature #2, στο αρχείο «functions.php».

```
function my_custom_script() {
    wp_enqueue_script('main', get_template_directory_uri().'/js/main.js', NULL, 0.1, true);
    wp_localize_script('main', 'identifier', array(
        'nonce' => wp_create_nonce('wp_rest')
    ));
}
add_action('wp_enqueue_scripts', 'my_custom_script');
```

Εικόνα 57. Ανάκτηση του αναγνωριστικού χρήστη

Χρησιμοποιούμε την εντολή «**wp_localize_script**», που δέχεται ως παραμέτρους το όνομα του script file, το όνομα του αντικειμένου που θέλουμε να κάνουμε Output/echo (το οποίο θα βαφτίζουμε ‘identifier’), και ως 3^η παράμετρο ένα array με αυτό που θέλουμε να εξάγουμε. Αυτό που θέλουμε είναι το nonce (number only used once), και για να το πάρουμε θα χρησιμοποιήσουμε την μέθοδο Wordpress «wp_create_nonce» με παράμετρο την ονομασία

‘wp_rest’. Επιστρέφοντας στο script μας, πλέον μπορούμε να περάσουμε στο Wordpress τον μοναδικό κωδικό Nonce, εφόσον το αντικείμενο identifier περιέχει την ιδιότητα ‘nonce’.

```
//Send a POST request to the api, notifying it that you want to create a new post.
var createPost = new XMLHttpRequest();
createPost.open('POST', 'https://exintaris.gr/WP/wp-json/wp/v2/posts');

//Authentication Check
createPost.setRequestHeader("X-WP-Nonce", identifier.nonce);

//Parse data to JSON and send it to create the post
createPost.setRequestHeader("Content-Type", "application/json;charset=UTF-8");
createPost.send(JSON.stringify(postData));
```

Εικόνα 58. Η ολοκληρωμένη αίτηση AJAX

Τέλος, θα ελέγξουμε εάν δημιουργήθηκε το post χωρίς σφάλματα και θα καθαρίσουμε τα πεδία input.

```
//Check if the post was created and clear the title and content fields.
createPost.onreadystatechange = function(){
  if (createPost.readyState == 4){
    if(createPost.status == 201){
      document.querySelector('.admin_quick_add [name="title"]').value = '';
      document.querySelector('.admin_quick_add [name="content"]').value = '';
    } else {
      alert("Error while creating the post. Please try again.");
    }
  }
}
```

Εικόνα 59. Έλεγχος για λάθη και κατάσταση της Δημοσίευσης

Οι αιτήσεις AJAX είναι ασύγχρονες, που σημαίνει ότι θα πρέπει να επιτρέψουμε ένα χρονικό περιθώριο για να διεκπεραιωθεί. Για να αναγνωρίσουμε σε ποιο στάδιο βρίσκεται το Request, ελέγχουμε το «readyState».

- 0: request not initialized
- 1: server connection established
- 2: request received
- 3: processing request
- 4: request finished and response is ready

Εφόσον είναι ‘4’, ελέγχουμε το status. 201 σημαίνει «CREATED», άρα το Post/ η ανακοίνωση που θέλαμε να περάσουμε έχει δημιουργηθεί κανονικά και μπορούμε να καθαρίσουμε τα πεδία Input. Η σελίδα που φιλοξενεί αυτό το feature περιέχει ένα widget του Avada που εμφανίζει τα πιο πρόσφατα Post. Οπότε εάν κάνουμε ανανέωση στη σελίδα θα πρέπει να δούμε το Post που μόλις δημιουργήσαμε στην κορυφή.

6 ΜΗΧΑΝΕΣ ΑΝΑΖΗΤΗΣΗΣ ΚΑΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ

6.1 ΕΙΣΑΓΩΓΗ:

Μηχανή αναζήτησης είναι ένα πρόγραμμα το οποίο βοηθά χρήστες του Internet να βρουν συγκεκριμένες πληροφορίες και περιεχόμενα στον Παγκόσμιο Ιστό [26]. Καθώς αυξάνεται η εξάρτηση του σύγχρονου τρόπου ζωής στο Διαδίκτυο, οι μηχανές αναζήτησης είναι ο κύριος παράγοντας για την εξαγωγή των κατάλληλων πληροφοριών την κατάλληλη στιγμή. Σήμερα υπάρχουν εκατοντάδες διαφορετικές μηχανές αναζήτησης, όλες με διακριτά χαρακτηριστικά και δυνατότητες. Google, Bing, Yahoo είναι μόνο κάποιες από της δημοφιλέστερες [27]. Συγκεκριμένα, η Google έχει γίνει ένας απαραίτητος σύνδεσμος μεταξύ των χρηστών του Internet και των πληροφοριών που θέλουν να βρουν στο διαδίκτυο.

Το μεγαλύτερο ποσοστό της κίνησης στο διαδίκτυο εξαρτάται από μηχανές αναζήτησης, και πολλοί χρήστες τις χρησιμοποιούν ως βασικό εργαλείο πλοήγησης, καθώς και ως φίλτρο ενημέρωσης [28] [29]. Καθώς όλο και περισσότερες ιστοσελίδες καταχωρούνται από μηχανές αναζήτησης και δημιουργείται ανταγωνισμός μεταξύ των, είναι προφανές πως η υψηλή κατάταξη και οι πρώτες θέσεις στις σελίδες αποτελεσμάτων γίνονται όλο και περισσότερο ζωτικής σημασίας για μια OnLine οντότητα [30]. SEO (Search Engine Optimization), είναι η διαδικασία βελτιστοποίησης μιας ιστοσελίδας ώστε να καταταχθεί σε υψηλότερες θέσεις, κατά την παρουσίαση αποτελεσμάτων έρευνας μέσω μηχανών αναζήτησης.

6.2 ΙΣΤΟΡΙΑ ΚΑΙ ΕΞΕΛΙΞΗ ΤΟΥ SEO

Οι ιδιοκτήτες σελίδων πρώτη φορά ξεκίνησαν να βελτιστοποιούν το περιεχόμενο τους στα μέσα της δεκαετίας του 1990, όταν και οι πρώτες μηχανές αναζήτησης άρχισαν να αρχειοθετούν τον ιστό. Αρχικά το μόνο που χρειαζόταν να κάνει κάποιος ήταν να καταθέσει τη διεύθυνση της σελίδας, η URL, σε διάφορες μηχανές αναζήτησης οι οποίες θα έστελναν έναν crawler να προσπελάσει την σελίδα, να αποσπάσει συνδέσμους και να επιστρέψει την πληροφορία για ευρετηρίαση [31]. Η αξία μιας υψηλής κατάταξης και ορατότητας ήταν προφανής από πολύ νωρίς, ωθώντας ένα κύμα από τεχνικούς SEO να εξασκήσουν τις ικανότητές τους, είτε με τη χρήση “White-hat” (‘Ηθικών’) είτε “Black-hat” (‘Ανήθικων’) μεθοδολογιών.

Οι πρώιμες εκδόσεις των αλγορίθμων βελτιστοποίησης εξαρτώνταν από πληροφορίες που παρείχαν οι διαχειριστές των σελίδων, όπως λέξεις κλειδιά στα HTML “meta-tags”, η ευρετήρια μηχανών αναζήτησης. Η χρήση metadata αποδείχθηκε αναξιόπιστη, καθώς οι λέξεις-κλειδιά που επέλεγαν οι διαχειριστές μπορεί να μην ήταν αντιπροσωπευτικές του περιεχομένου της σελίδας, με αποτέλεσμα να εμφανίζονται στην αναζήτηση σελίδες με περιεχόμενο πολύ διαφορετικό από αυτό που έψαχνε ο χρήστης. Οι webmasters έκαναν προσπάθειες να επιτύχουν υψηλές κατατάξεις, και πολλές φορές αυτό το πετύχαιναν με τη χρήση αθέμιτων μεθόδων, όπως υπερβολική χρήση keyword, συχνά και άσχετων με το περιεχόμενο. Κάποιες από τις πρώτες μηχανές αναζήτησης, όπως η Altavista και η Infoseek, έκαναν αλλαγές στους αλγορίθμους τους για την αντιμετώπιση αυτής της συμπεριφοράς [32].

Η εξάρτηση των μηχανών αναζήτησης από μετρικές όπως πυκνότητα λέξεων-κλειδιών (“Keyword Density”) οι οποίες ήταν εκτός του ελέγχου τους, προκαλούσε σημαντικά προβλήματα στην κατάταξη των ιστοσελίδων. Οι μηχανές αναζήτησης έπρεπε να

προσαρμοστούν ώστε οι σελίδες αποτελεσμάτων τους να εμφανίζουν πάντα τις πιο σχετικές πληροφορίες. Αυτό απαιτούσε αλλαγή των μεθοδολογιών για βαθμολόγηση ιστότοπων, με λιγότερο βάρος στην πυκνότητα των όρων και χρήση νέων μετρικών. Αυτό οδήγησε στην ανάπτυξη και αύξηση της πολυπλοκότητας των αλγορίθμων με χρήση στοιχείων που ήταν δυσκολότερα να παραποιηθούν από webmasters.

Οι μηχανές αναζήτησης έχουν αναπτύξει ισχυρούς δεσμούς με τη βιομηχανία SEO, παρέχοντας χρηματοδότηση και συμμετέχοντας πολύ συχνά σε συνέδρια ή σεμινάρια, ως καλεσμένοι ομιλητές ή και διοργανωτές. Ταυτόχρονα, έχουν εκδώσει οδηγούς και εκπαιδευτικό υλικό για να βοηθήσουν με τη βελτιστοποίηση ιστότοπων [33] [34]. Παραδείγματος χάριν, η Google διαθέτει έναν οδηγό Sitemap για την ενημέρωση των webmaster σχετικά με την κατάσταση της σελίδας τους και για τυχόν προβλήματα στην ευρετηρίασή τους [35]. Η Bing διαθέτει την σουίτα εργαλείων “Bing Webmaster Tools”, που παρέχει στους webmaster έναν τρόπο να καταθέσουν sitemap, να παρακολουθήσουν την κατάσταση ευρετηρίου της σελίδας τους και να τροποποιήσουν διάφορες παραμέτρους σχετικά με την αλληλεπίδραση σελίδας – μηχανής αναζήτησης.

Το 2015 ανακοινώθηκε ότι η Google στρέφει τις προτεραιότητές της στην αναζήτηση σε φορητές συσκευές, και ως αποτέλεσμα πολλές εταιρίες υιοθετούν νέες στρατηγικές marketing και κατά συνέπεια βελτιστοποίησης [36].

6.3 ΜΗΧΑΝΕΣ ΑΝΑΖΗΤΗΣΗΣ

Μια μηχανή αναζήτησης θεωρείται ως η πηγή, μέσω της οποίας προωθείται μια ιστοσελίδα και κατ’ επέκταση η οντότητα/επιχείρηση/οργανισμός που αντιπροσωπεύει. Οι μηχανές αναζήτησης εξερευνούν τα περιεχόμενα μιας ιστοσελίδας για να συλλέξουν πληροφορίες για αυτήν. Οπότε δημιουργείται η ανάγκη να βελτιστοποιηθεί μια ιστοσελίδα ώστε να γίνει “φιλική” προς τις μηχανές αναζήτησης. Αυτό βοηθά στο να βρεθεί η σελίδα στα κορυφαία αποτελέσματα αναζήτησης. Τα προγράμματα των μηχανών αναζήτησης ονομάζονται “crawlers”, “robots” ή “spiders”. Οι μηχανές αναζήτησης κατατάσσονται σε 2 κύριες κατηγορίες: “Crawler-based” και “Human-powered directories”; με θεμελιώδη διαφορετικό τρόπο λειτουργίας η καθεμία. Από αυτές τις 2 κατηγορίες απορρέουν άλλα δύο είδη μηχανών αναζήτησης, οι Υβριδικές, που συνδυάζουν “crawler-based” αναζήτηση με υπηρεσία ευρετηρίου, και “Meta” μηχανές αναζήτησης, οι οποίες προωθούν την ερώτηση του χρήστη σε πολλές άλλες μηχανές αναζήτησης, και επιστρέφουν τα αποτελέσματα συνδυασμένα σε μια σελίδα. Παραδείγματα τέτοιων σελίδων είναι Dogpile, Highway61, Mamma κ.α. Όσον αφορά τη βελτιστοποίηση ιστοσελίδων και το SEO, τα είδη μηχανών αναζήτησης που μας ενδιαφέρουν είναι οι crawler-based μηχανές.

6.3.1 Μηχανές αναζήτησης βασισμένες σε Περιηγητή (Crawler-Based)

Οι crawler-based μηχανές αναζήτησης τυπικά δουλεύουν σε 3 βήματα [37].

6.3.1.1 Περιήγηση (Crawling)

Ένας web crawler ξεκινά με μια λίστα URL (διευθύνσεων) που έχει να επισκεφτεί, τα αποκαλούμενα “seed”. Καθώς ο crawler επισκέπτεται αυτές τις διευθύνσεις, εντοπίζει όλους τους συνδέσμους στις σελίδες και τους προσθέτει στην λίστα των URL προς επίσκεψη.

Επισκέψεις σε αυτές τις σελίδες γίνονται αναδρομικά σύμφωνα με μια λίστα κανόνων (policies) :

- Selection Policy (Πολιτική επιλογής) – Καθορίζει ποιες σελίδες θα αποθηκευτούν.
- Re-visit Policy (Πολιτική ανά-επίσκεψης) – Καθορίζει πότε θα ελέγχονται οι σελίδες για αλλαγές
- Politeness Policy (Πολιτική ευγενείας) – Καθορίζει πως αποφεύγεται η υπερφόρτωση σελίδων
- Parallelization Policy (Πολιτική παραλληλισμού) – Καθορίζει πως θα συντονίζονται οι crawlers

Εάν ο crawler κάνει και αρχειοθέτηση ιστοσελίδων, αντιγράφει και αποθηκεύει το περιεχόμενο που συναντά κατά την περιήγησή του. Τα αρχεία συνήθως αποθηκεύονται με τρόπο που επιτρέπει την προσπέλαση και ανάγνωσή τους, ωστόσο είναι αποθηκευμένα ως στιγμιότυπα [38].

Το αρχείο είναι σχεδιασμένο για την αποθήκευση και διαχείριση της συλλογής ιστοσελίδων, αποθηκεύει μόνο HTML σελίδες , και αυτές καταχωρούνται ως διακριτά αρχεία. Έχει δομή παραπλήσια μιας βάσης δεδομένων, με διαφορά το ότι ένα τέτοιο αρχείο δεν χρειάζεται όλη τη λειτουργικότητα μιας βάσης δεδομένων. Σε αυτό αποθηκεύεται η πιο πρόσφατη εκδοχή της ιστοσελίδας που επέστρεψε ο crawler [39].

6.3.1.2 Καταχώρηση (Indexing)

Όπως προαναφέρθηκε, οι μηχανές αναζήτησης χρησιμοποιούν προγράμματα που ονομάζονται crawler. Οι crawler προσπελούν και διαβάζουν ιστοσελίδες, τις οποίες αργότερα αποθηκεύουν σε μία κεντρική δομή [40]. Αυτό είναι γνωστό ως “Ευρετηρίαση” και η δομή ως “Ευρετήριο”. Όταν ένας χρήστης εισάγει έναν όρο προς αναζήτηση, συνήθως κάποιες λέξεις-κλειδιά ή φράσεις, η μηχανή αναζήτησης αντλεί αποτελέσματα από το Ευρετήριο (βάση δεδομένων), σχετικά με το ερώτημα του χρήστη. Ο σκοπός της προσπέλασης του διαδικτύου (crawling) και της ευρετηρίασης (indexing) είναι η αποτελεσματικότερη ανταπόκριση σε ερωτήματα χρηστών, ώστε να εντοπίζονται γρήγορα και αποτελεσματικά οι σχετικές πληροφορίες.

6.3.1.3 Κατάταξη (Ranking)

Οι μηχανές αναζήτησης κατατάσσουν τις σελίδες που είναι αποθηκευμένες στο ευρετήριο, και μόνο αυτές. Αυτός είναι ο λόγος που νέες σελίδες που δεν έχουν καταχωρηθεί στο ευρετήριο δεν εμφανίζονται στα αποτελέσματα των αναζητήσεων. Οι crawlers επισκέπτονται νέες , και ήδη καταχωρημένες σελίδες περιοδικά. Ο χρόνος προσπέλασης και το διάστημα μεταξύ επισκέψεων στην ίδια σελίδα ποικίλλει, αναλόγως με τον αλγόριθμο της εκάστοτε μηχανής αναζήτησης [41]. Ο τρόπος που κατατάσσονται οι ιστοσελίδες διαφέρει σε κάθε μηχανή αναζήτησης, ωστόσο εάν μια σελίδα έχει υψηλή κατάταξη σημαίνει ότι διαθέτει τις πλέον σχετικές πληροφορίες σε σχέση με συγκεκριμένες λέξεις-κλειδιά. Η σελίδα με τις πιο σχετικές πληροφορίες για μια αναζήτηση χρήστη θα εμφανιστεί στην κορυφή των αποτελεσμάτων. Οι web developers προσπαθούν να βελτιώνουν την κατάταξη της ιστοσελίδας τους, γιατί υψηλότερη θέση στα αποτελέσματα αποφέρει περισσότερες επισκέψεις και “clicks” στη σελίδα.

6.3.2 Πληρωμένα και Οργανικά αποτελέσματα

Οι υβριδικές μηχανές αναζήτησης, όπως Google και Yahoo παρουσιάζουν δύο τύπους αποτελεσμάτων σχετικά με τις λέξεις-κλειδιά της αναζήτησης. Τα αποτελέσματα αυτά κατηγοριοποιούνται ως “Pay Per Click” (PPC) , η πληρωμένα αποτελέσματα τα οποία προέρχονται από “human-powered directories”, και ως Οργανικά (“Organic”) αποτελέσματα.

Τα πληρωμένα αποτελέσματα συνήθως παρουσιάζονται στην κορυφή η στην άνω δεξιά θέση της σελίδας αποτελεσμάτων. Η σελίδα αυτή ονομάζεται SERP (Search Engine Results Page), και ο τρόπος που αποτελέσματα παρουσιάζονται σε αυτή διαφέρει αναλόγως με την πολιτική της μηχανής αναζήτησης. Με την χρήση PPC, ο διαφημιστής/webmaster πληρώνει ένα ποσό στην μηχανή αναζήτησης για κάθε φορά που η διαφήμιση/σύνδεσμος του δέχεται ένα «κλικ». Το μοντέλο αυτό επιτρέπει σε διαφημιστές να πληρώνουν μόνο όταν ένας χρήστης αλληλοεπιδρά με το περιεχόμενό τους. Οι διαφημιστές πληρώνουν για συγκεκριμένα keywords που θέλουν να συσχετίσουν με το περιεχόμενό τους, όσο περισσότερα keywords τόσο μεγαλύτερο το ποσό που πληρώνουν [42].

Εάν οργανικό αποτέλεσμα είναι προϊόν της διαδικασίας βελτιστοποίησης (SEO), όπου ακολουθούνται στρατηγικές με στόχο την υψηλή κατάταξη της ιστοσελίδας σε μια SERP, χωρίς πληρωμή στις μηχανές αναζήτησης. Ιστοσελίδες με μοναδικό, ενημερωμένο, υψηλής ποιότητας περιεχόμενο με καλή χαρτογράφηση συνδέσμων θα έχουν καλύτερη θέση στη λίστα οργανικών αποτελεσμάτων. Επένδυση σε SEO για καλύτερα οργανικά αποτελέσματα αποφέρει περισσότερη κίνηση μακροπρόθεσμα, καθώς είναι πολύ πιθανότερο οι χρήστες να «πατήσουν» τα φυσικά αποτελέσματα παρά τις διαφημίσεις [43].

6.4 ΠΩΣ ΒΕΛΤΙΣΤΟΠΟΙΕΙΤΑΙ ΕΝΑΣ ΙΣΤΟΤΟΠΟΣ

Όπως αναφέραμε προηγουμένως, SEO είναι η διαδικασία βελτιστοποίησης μιας ιστοσελίδας ώστε να καταταχθεί σε υψηλότερες θέσεις, κατά την παρουσίαση αποτελεσμάτων έρευνας μέσω μηχανών αναζήτησης. Η πιο συγκεκριμένα, η διαδικασία βελτίωσης της ποιότητας και ποσότητας κίνησης μιας ιστοσελίδας, αυξάνοντας την ορατότητα της για τους χρήστες των μηχανών αναζήτησης. Στοχεύει στα «οργανικά» αποτελέσματα και δεν αφορά την απευθείας κίνηση (“direct traffic” – όταν ένας χρήστης γνωρίζει και επισκέπτεται τη σελίδα δίχως μηχανή αναζήτησης για ενδιάμεσο) και την πληρωμένη τοποθέτηση σε SERP.

Ως κομμάτι κάθε στρατηγικής SEO, λαμβάνεται υπόψη ο τρόπος λειτουργίας των μηχανών αναζήτησης, των αλγορίθμων που καθορίζουν τη συμπεριφορά τους, τις τάσεις αναζητήσεων των χρηστών, είτε σε θεματολογία είτε σε προτίμηση μηχανής αναζήτησης, και τις λέξεις-κλειδιά που χρησιμοποιούν σε αυτές τις αναζητήσεις. Η διαδικασία SEO πραγματοποιείται γιατί μια ιστοσελίδα θα δεχτεί περισσότερους επισκέπτες από μια μηχανή αναζήτησης εάν αποκτήσει υψηλή κατάταξη στην σελίδα αποτελεσμάτων (SERP). Αυτοί οι επισκέπτες μετά μπορεί να μετατραπούν σε πελάτες [44].

6.4.1 Κάνοντας τη σελίδα μας προσπελάσιμη

Το πρώτο βήμα στη διαδικασία βελτιστοποίησης είναι να σιγουρευτούμε πως η σελίδα μας είναι ορατή στις μηχανές αναζήτησης, επισκέψιμη, και εμφανίζεται στις SERP. Ένας καλός τρόπος να το μάθουμε είναι να ελέγξουμε πόσες από τις σελίδες είναι αποθηκευμένες στο ευρετήριο. Αυτό μπορεί να γίνει χρησιμοποιώντας τον όρο αναζήτησης “site:domain.com”, όπου domain είναι το όνομα της εκάστοτε ιστοσελίδας, και η αναζήτηση αυτή θα επιστρέψει

αποτελέσματα που η Google έχει στο ευρετήριο της. Ο αριθμός των αποτελεσμάτων ίσως να μην είναι απόλυτα ακριβής ωστόσο μας δίνει μια ιδέα για το ποιες σελίδες του ιστότοπου μας είναι ευρετηριασμένες, καθώς και για το πώς εμφανίζονται. Εάν η σελίδα μας δεν εμφανίζεται στα αποτελέσματα, κάποιοι λόγοι μπορεί να είναι:

- Η σελίδα είναι καινούρια και δεν έχει προσπελαστεί ακόμα από crawlers
- Η σελίδα δεν έχει εξωτερικούς συνδέσμους που να οδηγούν σε αυτή
- Η δομή της σελίδας δυσκολεύει η εμποδίζει τους crawler να την προσπελάσουν
- Η σελίδα εφαρμόζει κανόνες που περιορίζουν τη δραστηριότητα των crawler
- Η σελίδα έχει υποστεί «ποινή» από την Google για ανήθικες τακτικές marketing

Τυπικά θέλουμε μια μηχανή αναζήτησης να βρίσκει όλες τις υποσελίδες του ιστότοπου μας, ωστόσο ορισμένες φορές μπορεί να υπάρχει περιεχόμενο που θέλουμε να κρύψουμε από τους crawler. Αυτό μπορεί να είναι παλιά URL με χαμηλής ποιότητας περιεχόμενο, σελίδες δοκιμαστικές ή υπό κατασκευή κ.α. Για να αποτρέψουμε τα Robot της Google, χρησιμοποιούμε ένα αρχείο που λέγεται robots.txt [45].

6.4.1.1 Το αρχείο Robots.txt

Το αρχείο robots.txt, λέει στους crawler μηχανών αναζήτησης ποιες σελίδες και αρχεία μπορούν και δεν μπορούν να προσπελάσουν στην ιστοσελίδα μας. Αυτό κυρίως χρησιμοποιείται για την αποφυγή υπερφόρτωσης της σελίδας με αιτήματα; Δεν είναι μηχανισμός για να αποκλείσει τις σελίδες από την Google. Εάν θέλουμε να αποκρύψουμε την σελίδα μας από την Google πρέπει να χρησιμοποιήσουμε noindex directives ή να προστατέψουμε τη σελίδα μας απαιτώντας κωδικό πρόσβασης. Ωστόσο το αρχείο αυτό έχει και σημαντικούς περιορισμούς που οφείλει να γνωρίζει ένας webmaster πριν το χρησιμοποιήσει, όπως:

- **Δεν υποστηρίζεται από όλες τις μηχανές αναζήτησης** – Οι οδηγίες στο robots.txt είναι κατευθυντήριες γραμμές, και δεν μπορούν να εγγυηθούν τη συμμόρφωση των crawler με τους κανόνες. Η Google και οι περισσότερες μηχανές αναζήτησης θα ακολουθήσουν τους κανόνες, ωστόσο κάποιοι crawlers μπορεί να τους αγνοήσουν.
- **Διαφορετικοί Crawler ερμηνεύουν τη σύνταξη με διαφορετικό τρόπο** – Παρόλο που οι περισσότεροι crawler ακολουθούν τις οδηγίες, μπορεί να τις ερμηνεύσουν διαφορετικά. Είναι σημαντικό να γνωρίζουμε τη σωστή σύνταξη για διαφορετικούς crawler ώστε οι οδηγίες αυτές να είναι καθολικά κατανοητές.
- **Μια σελίδα που έχει αποκλειστεί από το robots.txt μπορεί ακόμα να ευρετηριαστεί** – Παρόλο που η Google δεν θα προσπελάσει ή αποθηκεύσει τη σελίδα που ορίζει το robots.txt, είναι δυνατόν να βρει και να αποθηκεύσει το URL της εάν αυτό βρίσκεται σε εξωτερική σελίδα ως σύνδεσμος. Αυτό θα έχει ως αποτέλεσμα να εμφανίζεται στην αναζήτηση η διεύθυνση URL και ίσως κι άλλες πληροφορίες όπως anchor text των συνδέσμων στη σελίδα. Ο καλύτερος τρόπος να αποκλειστεί πλήρως μια σελίδα είναι είτε η χρήση κωδικού στους φακέλους του server, το meta tag noindex, η φυσικά η ολική αφαίρεση της σελίδας [46].

Πολλές φορές μια μηχανή αναζήτησης θα μπορεί να βρει τμήματα της σελίδας μας κατά την προσπέλαση, ωστόσο κάποιες υποσελίδες μπορεί να παραμένουν απροσπέλαστες για

διάφορους λόγους. Κάποιοι από τους λόγους αυτούς μπορεί να είναι ότι το περιεχόμενό μας είναι «κρυμμένο» πίσω από φόρμες σύνδεσης χρήστη, ή απαιτεί τη χρήση εργαλείου αναζήτησης στη σελίδα για να εμφανιστεί, είτε ακόμα και επειδή είναι σε μορφή που η μηχανή αναζήτησης δεν μπορεί να διαβάσει (βίντεο/εικόνες κ.α.). Ένας crawler δεν μπορεί να κάνει login στη σελίδα, ούτε και να χρησιμοποιήσει την αναζήτηση, και παρόλο που με τη χρήση μηχανικής μάθησης οι μηχανές αναζήτησης έχουν βελτιωθεί στην αναγνώριση εικόνας/βίντεο και άλλων μορφών πληροφορίας, ακόμα δεν είναι σε βαθμό που θα κατανοήσουν σε ικανοποιητικό βαθμό σε τί αναφέρεται το περιεχόμενο. Τέλος είναι ιδιαίτερα σημαντικό να υπάρχει ένα μονοπάτι συνδέσμων ώστε να μπορούν οι crawlers να πλοηγηθούν από σελίδα σε σελίδα στον ιστότοπο μας. Εάν έχουμε μια σελίδα στην οποία δεν οδηγεί κανένας άλλος σύνδεσμος, η σελίδα είναι πρακτικά αόρατη για την μηχανή αναζήτησης **Invalid source specified.**

6.4.1.2 Δομή Υπερσυνδέσμων (URL)

Η δομή των URL μας μπορεί να επηρεάσει πώς εμφανίζεται το site μας και πώς αξιολογούνται οι σελίδες του. Η Google ευνοεί σελίδες με ξεκάθαρα URL που διευκολύνουν τους χρήστες να πλοηγηθούν στη σελίδα, παρέχοντας ταυτόχρονα περιγραφικό κείμενο που βοηθά τη μηχανή αναζήτησης να κατανοήσει το περιεχόμενο της σελίδας. Υπάρχουν κάποιες βασικές κατευθυντήριες γραμμές όταν αποφασίζουμε τη δομή των URL του site μας. Κάποιες από αυτές είναι η αποφυγή Dynamic URLs όπου αυτό είναι δυνατό, η αποφυγή χρήσης ειδικών χαρακτήρων όπως “#\$/%^\&*” ή μεγάλων αριθμητικών ακολουθιών όπως “36543823545123”.

Προτείνεται επίσης η χρήση ενός “breadcrumb trail” που θα δείχνει τη θέση κάθε σελίδας ανάμεσα στις διάφορες κατηγορίες του site. Για παράδειγμα: «DomainName.gr/first-category/second-category/page». Είναι θεμιτό να χρησιμοποιούμε ευνόητο και περιγραφικό κείμενο στο τέλος κάθε URL, που να περιλαμβάνει και τις λέξεις-κλειδιά μας εάν είναι δυνατόν. Το ιδανικό URL είναι στατικό, ευνόητο, προσφέρει breadcrumb trail και περιγράφει το περιεχόμενο της σελίδας με συναφή keywords [47].

6.4.1.3 Sitemaps

Ένα πολύ πρακτικό εργαλείο για να βοηθήσουμε την μηχανή αναζήτησης να βρει και να αρχειοθετήσει τις σελίδες μας, είναι το Sitemap. Πρόκειται για ένα αρχείο στο οποίο παρέχουμε πληροφορίες σχετικά με τις σελίδες, τα βίντεο και άλλα αρχεία στον ιστότοπο μας, καθώς και για τις σχέσεις μεταξύ τους. Οι μηχανές αναζήτησης όπως η Google το χρησιμοποιούν για να προσπελάσουν πιο αποτελεσματικά την ιστοσελίδα μας. Το Sitemap τους ενημερώνει για το ποιες σελίδες και αρχεία θεωρούμε σημαντικότερα στη σελίδα, τότε μια σελίδα έχει ενημερωθεί τελευταία, πόσο συχνά αλλάζει μια σελίδα, εάν υπάρχουν εναλλακτικές εκδόσεις της σελίδας (π.χ. με διαφορετική γλώσσα) κ.α.. Μια καταχώριση σε sitemap μπορεί να παρέχει πληροφορίες για ένα βίντεο ή μια εικόνα, όπως την διάρκεια του βίντεο σε λεπτά, κατηγορία και θεματολογία, είδος του αντικειμένου μιας εικόνας, τύπος αρχείου και άδεια. Τυπικά χρειαζόμαστε sitemap εάν:

- **Ο ιστότοπος μας είναι πολύ μεγάλος** – Όπου οι web crawlers μπορεί να αγνοήσουν κάποιες νέες ή ενημερωμένες σελίδες.
- **Ο ιστότοπος διαθέτει εκτενές αρχείο σελίδων που δεν συνδέονται αποτελεσματικά μεταξύ τους** – Εάν οι σελίδες του ιστότοπου μας δεν περιέχουν συνδέσμους που τις αλληλοσυνδέουν, μπορούμε να χρησιμοποιήσουμε sitemap για να βοηθήσουμε τις μηχανές αναζήτησης να τις ανακαλύψουν.

- **Ο ιστότοπος είναι καινούριος και δεν οδηγούν σε αυτόν εξωτερικοί σύνδεσμοι** – Εάν άλλες ιστοσελίδες δεν οδηγούν στη δική μας, οι crawler των μηχανών αναζήτησης μπορεί να μην τις ανακαλύψουν.
- **Ο ιστότοπος μας είναι πλούσιος σε πολυμεσικό περιεχόμενο** – Εάν διαθέτουμε πολλές εικόνες, βίντεο, αρχεία ήχου, είτε ακόμα και εάν η σελίδα μας εμφανίζεται σε site όπως Google News, ένα Sitemap μπορεί να δώσει πληροφορίες για το περιεχόμενό για καλύτερα αποτελέσματα κατά την αναζήτηση.

Ένα Sitemap ίσως να μην είναι απαραίτητο εάν το site μας είναι μικρού μεγέθους, εάν είναι σχεδιασμένο με λεπτομερές μονοπάτι συνδέσμων μεταξύ των σελίδων ή εάν δεν έχουμε πολλά πολυμεσικά αρχεία. Ωστόσο, παρόλο που ένα Sitemap δεν εγγυάται ότι όλες οι καταχωρήσεις του θα προσπελαστούν και αποθηκευτούν σε ευρετήριο, δεν έχει καμία αρνητική συνέπεια η ύπαρξή του οπότε και προτείνεται [35].

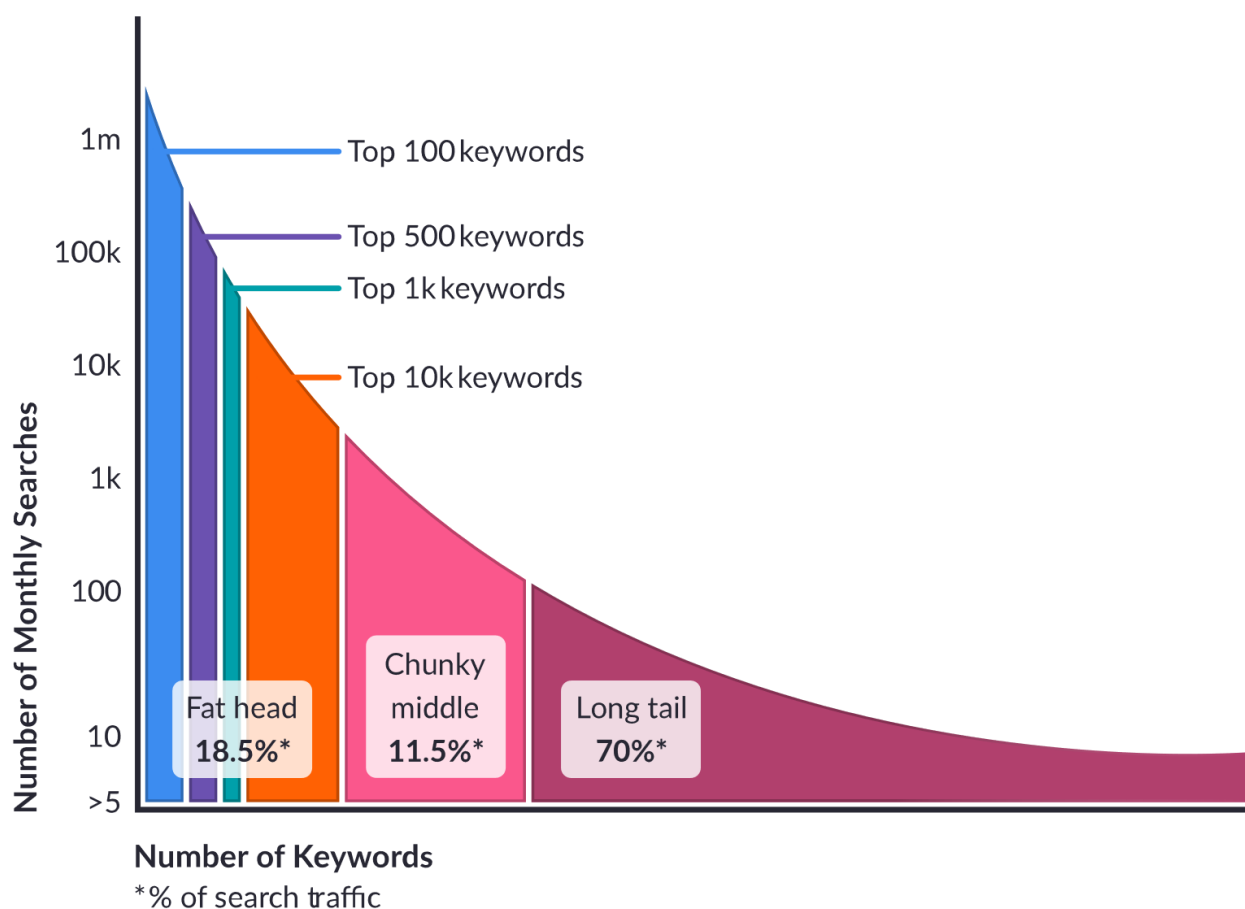
6.4.2 Έρευνα Λέξεων-Κλειδιών (Keyword Research)

Αφού βεβαιωθούμε πως η δομή του ιστότοπου μας είναι άρτια, και το site είναι ορατό και προσπελάσιμο από τις μηχανές αναζήτησης, το επόμενο βήμα είναι το περιεχόμενο της σελίδας. Keyword Research είναι η διαδικασία κατά την οποία αποφασίζουμε ποιες είναι οι λέξεις ή φράσεις που χρησιμοποιούν οι χρήστες στις αναζητήσεις τους, οι οποίες αντιπροσωπεύουν καλύτερα το προϊόν ή περιεχόμενο της σελίδας μας. Η έρευνα αυτή είναι σημαντική για όλα τα βήματα της βελτιστοποίησης, και για αυτό συνήθως είναι το πρώτο βήμα μιας καμπάνιας SEO [48].

6.4.2.1 Τα διαφορετικά είδη Λέξεων-Κλειδιών

Κατασκευάζοντας το site μας, συνήθως έχουμε ήδη μια ιδέα του περιεχομένου που θέλουμε να παρουσιάσουμε, κάποιους όρους που το περιγράφουν και σε ποιες ομάδες χρηστών του Internet απευθύνεται. Εάν, παραδείγματος χάριν, έχουμε ένα blog για ασφάλεια δικτύων/ network security, προφανώς θα θέλουμε να χρησιμοποιήσουμε την φράση “network security” ως μια από τα Keywords μας. Το πρόβλημα με τέτοιες λέξεις κλειδιά είναι ο υψηλός ανταγωνισμός, δηλαδή η ευρεία χρήση τους από πολλές διαφορετικές ιστοσελίδες ως λέξεις κλειδιά ακριβώς επειδή πρόκειται για πολύ γενικευμένους όρους, με αποτέλεσμα να έχουν μειωμένη αποτελεσματικότητα. Συχνά τέτοιες λέξεις κλειδιά αναφέρονται ως “fat head keywords”. Εάν ωστόσο εμβαθύνουμε και γίνουμε πιο συγκεκριμένοι με τα keyword μας, εάν π.χ. χρησιμοποιήσουμε keywords όπως “Network Security Certification”, ή “Network Security Tools” αναλόγως με τα ενδιαφέροντα του target group χρηστών μας, ο ανταγωνισμός για υψηλή κατάταξη αποτελεσμάτων μειώνεται. Αυτές οι λέξεις κλειδιά ονομάζονται “Long-tail keywords”, και τυπικά αποτελούνται από 3 ή περισσότερες λέξεις, άρα είναι πιο συγκεκριμένες και είναι πιθανότερο να τραβήξουν την προσοχή ενός χρήστη. Μπορεί επίσης να έχουν μικρότερο όγκο αναζητήσεων, αλλά για αυτό τον λόγο έχουν και πολύ μικρότερο ανταγωνισμό.

The Search Demand Curve



Εικόνα 60. Η καμπύλη Αναζητήσεων ανά Λέξεις/Κλειδιά

Image Source: <https://moz.com/beginners-guide-to-seo/keyword-research>

Όπως φαίνεται και στην Εικόνα 1, παρόλο που οι fat-head λέξεις-κλειδιά έχουν μακράν περισσότερες αναζητήσεις μηνιαίως, το μεγαλύτερο σημαντικό ποσοστό κίνησης (δηλαδή χρήστες που θα ακολουθήσουν τους συνδέσμους από την αναζήτηση στην ιστοσελίδα) προέρχεται από long-tail keywords. Ένας χρήστης που αναζητά π.χ. τις λέξεις “Network Security” κατά κανόνα δεν έχει συγκεκριμένο σκοπό αναζήτησης ή δεν γνωρίζει ακριβώς τι ψάχνει. Ωστόσο εάν ένας χρήστης αναζητήσει για π.χ. “Network Security Engineer Positions”, τότε έχουμε έναν χρήστη που ξέρει τι θέλει να βρει και είναι έτοιμος να ακολουθήσει συνδέσμοι που ικανοποιούν τα κριτήριά του [49].

6.4.2.2 Συμπεριφορά Αναζήτησης Χρήστη (User Search Behavior)

Όπως είπαμε προηγουμένως, κείμενο στο Keyword Research είναι η γνώση των χρηστών στους οποίους «στοχεύουμε» το περιεχόμενο του site μας. Δηλαδή τί αναζητούν και με ποια μορφή θέλουν να «καταναλώσουν» τα αποτελέσματα. Σε αυτό μας βοηθούν πολύ τα SERP των μηχανών αναζήτησης, ιδιαίτερα της Google, η οποία περιγράφει τις προθέσεις των χρηστών στα “Quality Rater’s Guidelines” [50], και τις διαχωρίζει σε διάφορες κατηγορίες, με τις παρακάτω 5 πρωτεύουσες:

- **Ανάκτηση πληροφοριών** - Ο χρήστης χρειάζεται πληροφορίες για γνωσιακούς σκοπούς, π.χ. «Ποια είναι η πρωτεύουσα της Ελλάδας».
- **Ερωτήσεις Πλοήγησης** – Ο χρήστης θέλει να «βρει» , η να πλοηγηθεί σε ένα συγκεκριμένο σημείο στο διαδίκτυο (π.χ. μια συγκεκριμένη ιστοσελίδα).
- **Αιτήματα Συναλλαγών** – Ο χρήστης θέλει να διεκπεραιώσει κάτι, π.χ. να ακούσει ένα τραγούδι, η να αγοράσει ένα αεροπορικό εισιτήριο
- **Εμπορική Έρευνα** – Ο χρήστης θέλει να κάνει έρευνα αγοράς, π.χ. να βρει διαφορετικές τιμές για ένα προϊόν.
- **Τοπικές Ερωτήσεις** – Ο χρήστης θέλει να βρει η να επισκεφθεί ένα μέρος στην περιοχή του, π.χ. ένα μουσείο ή ένα εστιατόριο.

Όταν αναγνωρίσουμε σε ποια κατηγορία ανήκει ένας χρήστης του target group μας, τότε μπορούμε να μελετήσουμε πώς διαμορφώνονται οι SERP στις αναζητήσεις αυτής της κατηγορίας και να προσαρμόσουμε το περιεχόμενό μας αντιστοίχως [51].

6.4.2.3 *Επιλογή και αξιολόγηση Keywords*

Αφού καταλήξουμε στο είδος του περιεχομένου που θέλουμε στη σελίδα και αναγνωρίσουμε το είδος του χρήστη που ανήκει στο target group μας, είναι καιρός να επιλέξουμε τα Keywords για τα οποία θέλουμε να καταταχθεί η σελίδα. Υπάρχουν διάφορα εργαλεία και πλατφόρμες που διευκολύνουν αυτή τη διαδικασία, όπως το Google Keyword Planner, Google Trends, Google Suggestions, Youtube, Reddit, Google Search Console, διάφορα forum, Amazon κ.α.. Υπάρχουν διάφορα μέσα που μπορούμε να χρησιμοποιήσουμε ωστόσο κατά κανόνα τα πιο αποτελεσματικά είναι 3rd Party Software και εξειδικευμένο λογισμικό, αλλά τα πιο διαδεδομένα (και δωρεάν) είναι τα εργαλεία της Google. Από το Keyword Planner που αποτελεί μέρος του Google Ads, μέχρι την κονσόλα αναζήτησης και τη λειτουργία auto-complete [51].

Όταν επιλέξουμε τα keyword που θέλουμε να χρησιμοποιήσουμε, η έχουμε καταλήξει σε κάποια υποψήφια, περνάμε στο στάδιο της ανάλυσης και αξιολόγησης, ώστε να επιλέξουμε λίγα από αυτά που θα μας εξυπηρετήσουν όσο το δυνατόν καλύτερα, έχοντας υπόψη μας ότι θα πρέπει να βελτιστοποιήσουμε για κάθε Keyword, οπότε μεγάλος αριθμός λέξεων κλειδιών μπορεί να αποτελέσει πρόβλημα.

Υπάρχουν 3 παράγοντες όταν αξιολογούμε Keywords: **Δημοτικότητα**, **Δυσκολία** και **Συνάφεια** [49]. Είναι ιδιαίτερος σημαντικό να ικανοποιούνται και οι 3 παράγοντες για μία λέξη-κλειδί, διότι:

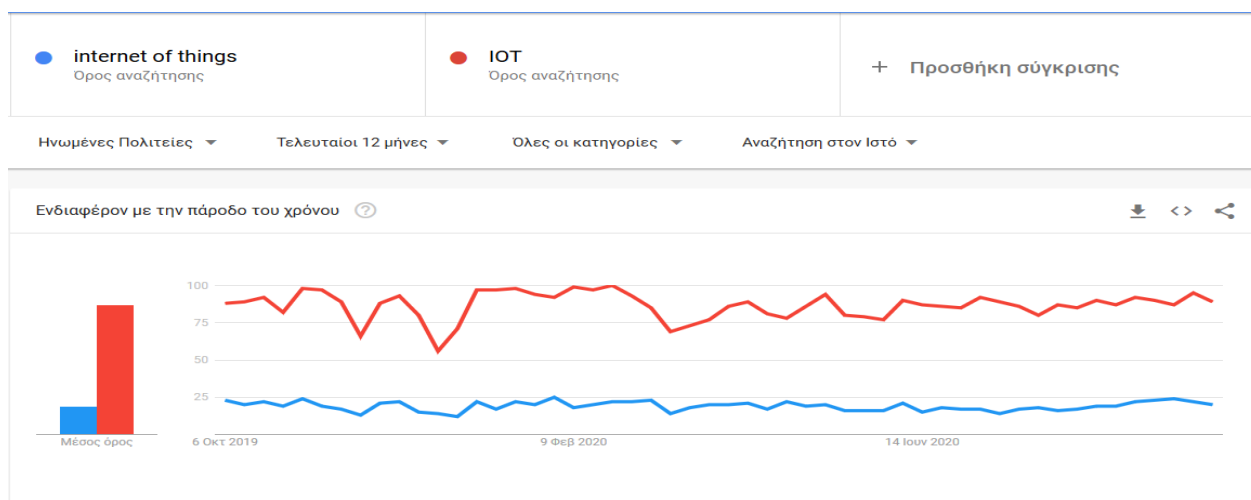
- Εάν ένα keyword έχει υψηλή δημοτικότητα, χαμηλή δυσκολία αλλά και χαμηλή συνάφεια, η Google δεν θα εμφανίσει τη σελίδα μας για αυτό το keyword.
- Εάν η δυσκολία είναι χαμηλή και το keyword έχει υψηλή συνάφεια, αλλά δεν είναι δημοφιλές τότε δεν θα έχουμε επισκεψιμότητα.
- Εάν η δημοτικότητα και η συνάφεια είναι υψηλές, αλλά η δυσκολία είναι υψηλή, θα είναι πολύ δύσκολο να ξεπεράσουμε τους ανταγωνιστές μας στην κατάταξη.

6.4.2.4 *Δημοτικότητα*

Δημοτικότητα εννοούμε τον όγκο των αναζητήσεων για ένα keyword – πόσοι χρήστες αναζητούν τη συγκεκριμένη λέξη. Συνήθως υπολογίζεται ως μηνιαίος μέσος όρος, με βάση τους τελευταίους 12 μήνες. Υπάρχουν 2 βασικές πηγές για πληροφορίες όγκου αναζητήσεων,

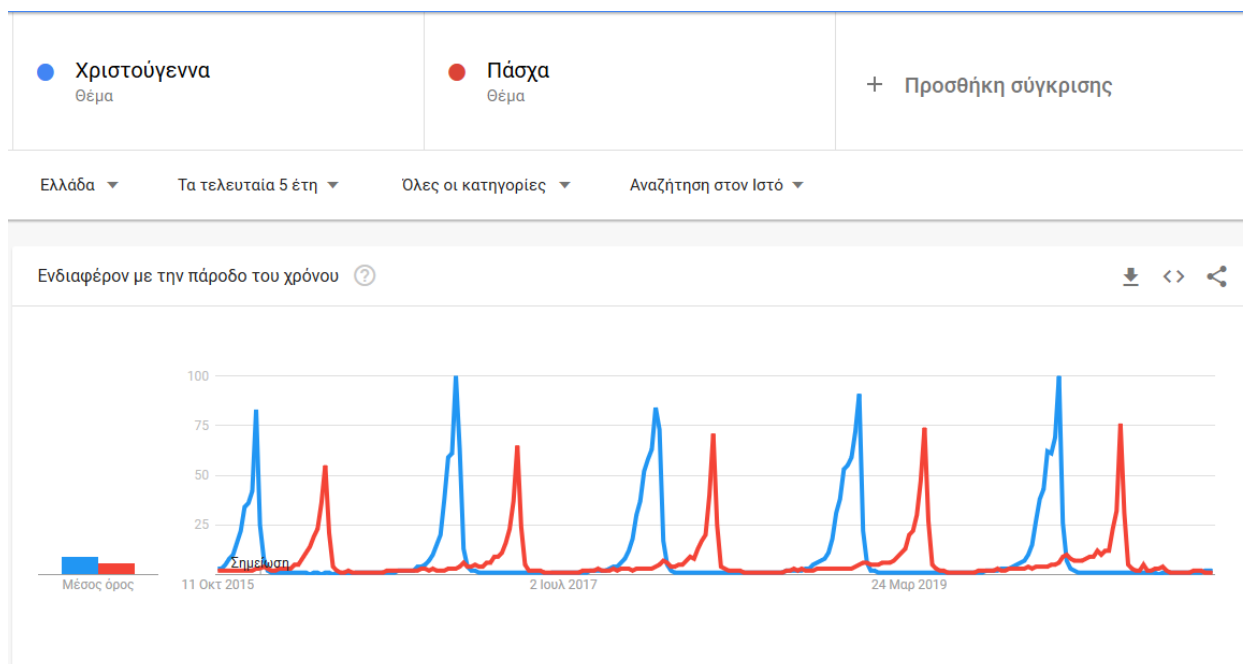
Google data (πληροφορίες για αναζητήσεις από την βάση δεδομένων του Google Keyword Planner) και **Clickstream data** (πληροφορίες για αναζητήσεις βασισμένες στη συμπεριφορά χρηστών του Internet, η οποία συλλέγεται από Plugins, επεκτάσεις περιηγητών κλπ.), και αναλόγως την πηγή, τα αποτελέσματα μπορεί να έχουν κάποια απόκλιση μεταξύ τους.

Επίσης σημαντικό είναι να εξετάζουμε την δημοτικότητα των keyword μακροπρόθεσμα, δηλαδή τάσεις ενδιαφέροντος για μια λέξη κλειδί σε βάθος χρόνου. Το καλύτερο εργαλείο για αυτό είναι το Google Trends. Απλά εισάγουμε μια λέξη κλειδί και το εργαλείο μας επιστρέφει ένα γράφημα της σχέσης ενδιαφέροντος/χρόνου σε μια κλίμακα από 1 έως 100.



Εικόνα 61. Δημοτικότητα του όρου IOT σε αναζητήσεις Google

Για παράδειγμα, στην παραπάνω εικόνα έχουμε μια σύγκριση των Keyword «Internet of Things» και «IOT». Οι όροι αυτοί χρησιμοποιούνται για να περιγράψουν το ίδιο αντικείμενο ωστόσο έχουν μεγάλη διαφορά στον όγκο αναζήτησής τους. Τα Google Trends επίσης μας παρέχουν πληροφορίες για τις τοποθεσίες με μεγάλη δραστηριότητα σχετικών αναζητήσεων, προτείνουν περισσότερα keyword σχετικά με την αναζήτηση, είτε μας ενημερώνουν για το εποχιακό ενδιαφέρον για ένα Keyword [49].



Εικόνα 62. Δημοτικότητα εποχιακών όρων

Στο παραπάνω παράδειγμα, έχουμε τη σύγκριση 2 όρων, «Χριστούγεννα» και «Πάσχα», για αναζητήσεις στην Ελλάδα τα τελευταία 5 έτη. Μπορούμε να δούμε πως η δημοτικότητα αυτών των λέξεων αυξάνεται συγκεκριμένες περιόδους κάθε έτους.

Τέλος, όταν αξιολογούμε τη δημοτικότητα ενός Keyword, πρέπει να λάβουμε υπόψη μας το CTR, ή αλλιώς “Click Through Rate”, δηλαδή το ποσοστό χρηστών που ακολουθούν έναν συγκεκριμένο σύνδεσμο σε σχέση με το σύνολο των χρηστών που θα δουν μια σελίδα, μια διαφήμιση ή ένα email. Χρησιμοποιείται συνήθως για να μετρήσει την αποτελεσματικότητα μιας διαφημιστικής καμπάνιας, είτε για μια ιστοσελίδα, είτε μέσω email [52] [53].

6.4.2.5 Δυσκολία Λέξεων Κλειδιών

Ως δυσκολία keywords αναφέρεται η στατιστική που υπολογίζει πόσο δύσκολο είναι να καταταχθείς από μια μηχανή αναζήτησης για μια συγκεκριμένη λέξη-κλειδί. Όσο υψηλότερη η δυσκολία, τόσο πιο δύσκολο να επιτευχθεί υψηλή κατάταξη σε αποτελέσματα αναζήτησης.

Οι μετρικές δυσκολίας στα διάφορα keyword tools λαμβάνουν υπόψη τους την αξιοπιστία των site που εμφανίζονται στην πρώτη σελίδα των SERP. Εάν υπάρχουν πολλά site με χαμηλή αξιοπιστία στην 1^η σελίδα τότε υπάρχει μεγάλη πιθανότητα να καταταχθεί η σελίδα μας σε καλή θέση. Στις περισσότερες περιπτώσεις η αξιοπιστία των sites υπολογίζεται με βάση 2 παράγοντες:

- **Αριθμός Backlinks** – Πόσες σελίδες περιέχουν συνδέσμους σε αυτό το site.
- **Ποιότητα Backlinks** – Εξαρτάται από την αξιοπιστία και την συνάφεια των σελίδων που περιέχουν links προς το site.

Πρέπει να έχουμε πάντα κατά νου ότι η δυσκολία ενός keyword είναι μόνο ένας παράγοντας αξιολόγησης. Εάν το περιεχόμενό μας είναι σωστά δομημένο και με υψηλή συνάφεια, μπορούμε να ξεπεράσουμε σε κατάταξη sites με μεγαλύτερη αξιοπιστία.

6.5 ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΣΕ ΕΠΙΠΕΔΟ ΣΕΛΙΔΑΣ

On-Site SEO (ή On-Page SEO) είναι η πρακτική βελτιστοποίησης των στοιχείων σε μία ιστοσελίδα, με σκοπό την υψηλή κατάταξη σε μηχανές αναζήτησης και την προσέλκυση σχετικής κίνησης. Το On-Site SEO αναφέρεται στην βελτιστοποίηση και του περιεχομένου αλλά και του πηγαίου κώδικα HTML μιας σελίδας.

Πέρα από το να βοηθά μηχανές αναζήτησης να ερμηνεύσουν το περιεχόμενο μιας σελίδας, η σωστή On-Site βελτιστοποίηση επίσης βοηθά χρήστες να κατανοήσουν γρήγορα το αντικείμενο μιας σελίδας, και εάν ανταποκρίνεται στα κριτήρια αναζήτησής τους. Ουσιαστικά, το αποτελεσματικό on-site SEO βοηθά τις μηχανές αναζήτησης να καταλάβουν τί θα έβλεπε ένας άνθρωπος εάν επισκεπτόταν την εν λόγω σελίδα, ώστε να προσφέρουν αξιόπιστα αυτό που οι επισκέπτες θα θεωρούσαν περιεχόμενο υψηλής ποιότητας για ένα συγκεκριμένο Query (Keyword) [54]. Ο τελικός στόχος της βελτιστοποίησης on-site, είναι να βοηθήσει και τους χρήστες και τις μηχανές αναζήτησης να θεωρήσουν τη σελίδα χρήσιμη, και άξια της υψηλής κατάταξης σε SERP. Τα κυριότερα στοιχεία του On-Page SEO έχουν ως εξής:

1. Περιεχόμενο Υψηλής Ποιότητας
2. Τίτλοι Σελίδων
3. Επικεφαλίδες (Headers)
4. Meta Descriptions
5. Εναλλακτικό κείμενο εικόνων (Alt-text)
6. Δομημένη Μορφοποίηση
7. URL σελίδων
8. Εσωτερική Διασύνδεση
9. Απόδοση σε Mobile Interfaces
10. Ταχύτητα σελίδας

Όλα τα στοιχεία συγκεντρώνονται σε 3 κατηγορίες: Στοιχεία Περιεχομένου, Στοιχεία HTML, και στοιχεία αρχιτεκτονικής [55].

6.5.1 Στοιχεία Περιεχομένου

Η κατηγορία αυτή αναφέρεται στο περιεχόμενο του site μας. Κείμενα, πολυμέσα, πληροφορίες που καθιστούν τη σελίδα μας ελκυστική στους χρήστες και δηλώνουν στις μηχανές αναζήτησης πως το site μας έχει υψηλή αξία.

6.5.1.1 Περιεχόμενο Υψηλής ποιότητας

Το πρώτο βήμα στη δημιουργία περιεχομένου υψηλής ποιότητας είναι η επιλογή συναφών keywords και θεματολογιών. Δηλαδή η εφαρμογή του Keyword Research που έχουμε πραγματοποιήσει. Κατά τη συγγραφή κειμένων πρέπει να ενσωματώσουμε μια ποικιλία keyword με οργανικό τρόπο, συνοδευόμενες με ενδιαφέρον και σχετικό οπτικό/πολυμεσικό περιεχόμενο. Πρέπει να γράφουμε για συγκεκριμένες ομάδες χρηστών, με βάση τα ερωτήματα και προβλήματά τους, και να δημιουργούμε υλικό που θα θέλουν να μοιραστούν ή να κοινοποιήσουν με συνδέσμους. Όλα τα παρακάτω στοιχεία On-Page SEO προέρχονται από το περιεχόμενο της σελίδας, οπότε πρέπει να επενδύσουμε σε αυτό.

6.5.2 Στοιχεία HTML

Η κατηγορία αναφέρεται στα πηγαία στοιχεία του κώδικα HTML της σελίδας, ή σε συγκεκριμένα HTML tags που προτείνεται να χρησιμοποιούμε, με στόχο να καταλαβαίνουν γρηγορότερα οι μηχανές αναζήτησης σε τί ακριβώς αναφέρεται το περιεχόμενο της σελίδας, και να το φορτώνουν γρηγορότερα [56].

6.5.2.1 Τίτλοι

Οι HTML τίτλοι ήταν και παραμένουν το σημαντικότερο σήμα στις μηχανές αναζήτησης για να κατανοήσουν σε τι αναφέρεται μια σελίδα. Οι τίτλοι επίσης μεταφέρουν στους χρήστες πληροφορίες για το περιεχόμενο της σελίδας, οπότε είναι σημαντικό να είναι μοναδικοί και περιγραφικοί.

Τίτλοι ορίζονται μέσω του HTML title tag: <title> My Title </title>. Για να σιγουρευτούμε ότι η σελίδα μας κατατάσσεται για το σωστό κοινό ενδείκνυται να συμπεριλαμβάνουμε τα κύρια keywords για την σελίδα στον τίτλο της – αποφεύγοντας πάντα υπερβολική χρήση [56].

Κάποιες από τις προτεινόμενες πρακτικές για την σύνταξη τίτλων:

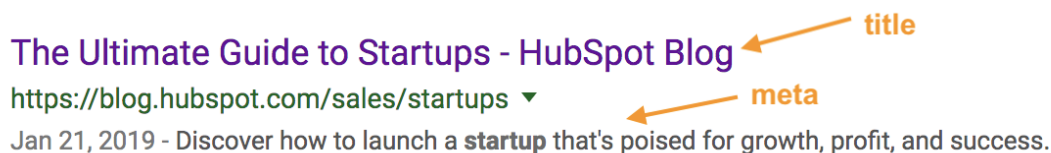
- Κατά προτίμηση λιγότερες από 70 λέξεις (σύμφωνα με ενημέρωση της Google). Μεγαλύτεροι από αυτό θα περικοπούν στις σελίδες αποτελεσμάτων, ενώ τα αποτελέσματα σε κινητές συσκευές δείχνουν έως 78 χαρακτήρες.
- Αποφυγή υπερβολικής χρήσης keywords στον τίτλο. Όχι μόνο είναι μια αντιαισθητική πρακτική προς τους χρήστες, αλλά οι μοντέρνοι αλγόριθμοι των μηχανών αναζήτησης αναγνωρίζουν τότε συμβαίνει και «τιμωρούν» περιεχόμενο με αφύσικες ποσότητες λέξεων-κλειδιών.
- Ο τίτλος πρέπει να είναι σχετικός με τη σελίδα.
- Αποθαρρύνεται η χρήση μόνο κεφαλαίων γραμμάτων.
- Προτείνεται η χρήση της ονομασίας της εταιρίας/οργανισμού στον τίτλο.

6.5.2.2 Κεφαλίδες (Headers)

Τα header, γνωστά και ως body tags, αναφέρονται στο στοιχείο HTML <h1>, <h2>, <h3> κ.α. Αυτά τα tags μας βοηθάνε να οργανώσουμε το περιεχόμενό μας για τους χρήστες που θα επισκεφθούν τη σελίδα, και τις μηχανές αναζήτησης να αναγνωρίσουν ποιο τμήμα του περιεχόμενου είναι σημαντικότερο. Συστήνεται η χρήση keyword , αλλά διαφορετικών από αυτές που χρησιμοποιούνται στον τίτλο. Οι πιο σημαντικές λέξεις-κλειδιά θα πρέπει να τοποθετηθούν στα <h1> , <h2>.

6.5.2.3 Μετα-Περιγραφές

Το Meta Description tag είναι μια σύντομη περιγραφή της σελίδας που εμφανίζεται κάτω από τον τίτλο στα αποτελέσματα αναζήτησης. Παρόλο που δεν είναι ένας επίσημος παράγοντας κατάταξης για μηχανές αναζήτησης, μπορεί να επηρεάσει εάν ένας χρήστης θα «κλικάρει» στη σελίδα, οπότε είναι ιδιαίτερος σημαντικό.



Εικόνα 63. Μετα-Περιγραφές σε αποτελέσματα αναζήτησης

Τα Meta Tags επίσης μπορούν να αντιγραφούν στα μέσα κοινωνικής δικτύωσης όταν το περιεχόμενό μας κοινοποιείται (χρησιμοποιώντας δομημένη μορφοποίηση), ώστε να ενθαρρύνει επισκέψεις και από εκεί. Μία καλή περιγραφή Meta τυπικά αποτελείται από λιγότερους από 160 χαρακτήρες, παρόλο που η Google επιτρέπει μεγαλύτερες περιγραφές – έως 220 χαρακτήρες. Σε κινητές συσκευές οι περιγραφές περικόπτονται μετά των 120 χαρακτήρων. Προφανώς ενδείκνυται να περιέχουν τις λέξεις-κλειδιά σχετικές με το περιεχόμενο της σελίδας, και συνήθως περιέχουν μία ή δύο ολοκληρωμένες προτάσεις, κατά προτίμηση χωρίς αλφαριθμητικούς χαρακτήρες [55].

6.5.2.4 Δομημένη Μορφοποίηση (Structured Markup)

Δομημένη Μορφοποίηση, ή Σήμανση, είναι η διαδικασία ταξινόμησης του κώδικά μας ώστε να το καταστήσουμε ευκολότερο για τις μηχανές αναζήτησης να εντοπίσουν και να κατανοήσουν διαφορετικά στοιχεία του περιεχομένου μας. Η μορφοποίηση αυτή καθορίζει την εμφάνιση του περιεχομένου μας όταν συμπεριλαμβάνεται σε code snippets, όπως όταν κοινοποιείται σε διαφορετικές πλατφόρμες ή όταν εμφανίζεται σε ένα από τα προβεβλημένα παράθυρα ενός SERP στην 1^η σελίδα αποτελεσμάτων.

6.5.2.5 Εναλλακτικό Κείμενο (Alternative Text)

Alt-text ουσιαστικά είναι SEO για τις εικόνες μας. Ενημερώνει τις μηχανές αναζήτησης για το τί απεικονίζεται σε μια εικόνα, και καθώς η Google επιστρέφει πλέον και πολλά αποτελέσματα εικόνας, η σημαντικότητά του έχει αυξηθεί. Μπορεί να επιτρέψει σε χρήστες να ανακαλύψουν την ιστοσελίδα μας και μέσω εικόνων, εάν φυσικά εισάγουμε το κατάλληλο εναλλακτικό κείμενο. Επίσης, είναι πολύ χρήσιμο για χρήστες με ειδικές ανάγκες καθώς είναι αναγνώσιμο από λογισμικό screen reader.

6.5.3 Στοιχεία Αρχιτεκτονικής

Η κατηγορία αυτή αναφέρεται στα στοιχεία που αποτελούν τον ιστότοπο και τις σελίδες μας. Ο τρόπος με τον οποίο κατασκευάζουμε τις σελίδες μπορεί να βοηθήσει την Google και άλλες μηχανές αναζήτησης να προσπελάσουν αποτελεσματικά το site μας.

6.5.3.1 Υπερσύνδεσμοι Σελίδων (URL)

Τα URL της σελίδας μας θα πρέπει να είναι απλά και κατανοητά, και για τους χρήστες και για τις μηχανές αναζήτησης. Είναι επίσης σημαντικά για την ιεραρχία της σελίδας καθώς δημιουργούμε σελίδες και υποσελίδες στο domain. Παραδείγματος χάριν, το παράδειγμα:

«<https://exintaris.gr/blog/article>»

Στο παραπάνω παράδειγμα το “exintaris.gr” είναι το domain, “blog” είναι η υποκατηγορία για το άρθρο, και “article” είναι ο δείκτης του άρθρου στο οποίο βρισκόμαστε. Προτείνεται η χρήση μόνο μίας, ή 2 λέξεων κλειδιών, και η χρήση πρωτοκόλλου HTTPS καθώς η Google το θεωρεί θετικό παράγοντα στην αξιολόγηση κατάταξης.

6.5.3.2 Εσωτερική Διασύνδεση

Η διαδικασία εισαγωγής συνδέσμων (hyperlinks) σε άλλες χρήσιμες σελίδες στο domain μας. Είναι σημαντικό για το SEO διότι οι εσωτερικοί σύνδεσμοι στέλνουν τους αναγνώστες σε άλλες σελίδες στο ίδιο site, επεκτείνοντας τον χρόνο που περνάνε στην ιστοσελίδα και κατ’ επέκταση ενημερώνοντας την Google ότι το site είναι χρήσιμο και αξιόλογο. Επίσης, όσο περισσότερο χρόνο περνάνε οι επισκέπτες στην ιστοσελίδα μας τόσο περισσότερο χρόνο έχει η Google να προσπελάσει και να αποθηκεύσει τις σελίδες, απορροφώντας περισσότερες πληροφορίες που πιθανόν να έχουν αποτέλεσμα υψηλότερη κατάταξη στις SERP.

6.5.3.3 Ανταποκρισιμότητα Κινητών Συσκευών

Η Google και άλλες μηχανές έχουν αρχίσει να ευνοούν σελίδες optimized για κινητές συσκευές – ακόμα και για αναζητήσεις από desktop [57]. Η απόδοση σε Mobile Interfaces είναι ιδιαίτερος σημαντική, όσον αφορά την υπηρεσία Hosting που επιλέγουμε, τον σχεδιασμό της σελίδας, το Theme της ιστοσελίδας και άλλα στοιχεία. Πρέπει να προσανατολιζόμαστε σε επιλογές που είναι αναγνώσιμες και πλοηγήσιμες από κινητές συσκευές. Η Google παρέχει ένα εργαλείο για να εξακριβώσουμε πόσο mobile-friendly είναι η σελίδα μας, ονόματι Google’s Mobile-Friendly Test Tool.

6.5.3.4 Ταχύτητα Ιστοσελίδας

Είτε σε περιβάλλον Desktop, είτε σε Mobile, η σελίδα μας πρέπει να φορτώνει γρήγορα. Η Google ενδιαφέρεται πάνω απ’ όλα για την εμπειρία των χρηστών (User Experience – UX). Εάν η σελίδα φορτώνει αργά, οι χρήστες δεν θα παραμείνουν – και οι μηχανές αναζήτησης θα το καταλάβουν. Επιπροσθέτως, αργές ταχύτητες μπορεί να επηρεάσουν αρνητικά τους ρυθμούς μετατροπής (περιστασιακούς επισκέπτες σε τακτικούς χρήστες/πελάτες) και το ROI (*ρυθμό μετατροπής κόστους διαφημίσεων σε πωλήσεις – κυρίως για σελίδες με κερδοσκοπικό χαρακτήρα) [58].

7 ΠΡΟΣΒΑΣΙΜΟΤΗΤΑ

7.1 ΕΙΣΑΓΩΓΗ

Καθώς ο κόσμος κινείται ολοένα και περισσότερο προς το διαδίκτυο και το ψηφιακό περιεχόμενο, είναι σημαντικό το περιεχόμενο αυτό να είναι προσβάσιμο από όλους, συμπεριλαμβανομένων και ατόμων με ειδικές ανάγκες. Η HTML έχει σχεδιαστεί ώστε να γίνουν οι ιστοσελίδες πιο προσβάσιμες προς το ευρύ κοινό [59], ωστόσο συχνά δεν αξιοποιείται η λειτουργικότητά της για την διευκόλυνση ευαίσθητων ομάδων. Ο στόχος της προσβασιμότητας ιστοσελίδων είναι για τους developer και designer να κάνουν το περιεχόμενό τους προσβάσιμο από όλες τις ομάδες του πληθυσμού.

Υπάρχουν πολλές διαφορετικές ομάδες χρηστών με διαφορετικές ανάγκες η κάθε μία, ωστόσο για χρήστες με ειδικές ανάγκες, μπορούμε να ξεχωρίσουμε 5 κύριες κατηγορίες.

- **Περιορισμοί Όρασης** – Συμπεριλαμβάνει μερική ή ολική αδυναμία αντίληψης συγκεκριμένων χρωματικών συνδυασμών.
- **Περιορισμοί Ακοής** – Χρήστες οι οποίοι έχουν μειωμένη ικανότητα ακοής.
- **Κινητικοί Περιορισμοί/Φυσικές αναπηρίες** – Χρήστες που μπορεί να έχουν δυσκολία στην κίνηση συγκεκριμένων μελών του σώματός τους, συμπεριλαμβανομένων κινήσεων υψηλής ακρίβειας (όπως η χρήση ενός ποντικιού).
- **Ευαισθησία στο Φως** – Παθήσεις όπως επιληψία μπορεί να προκαλέσουν κρίσεις, συχνά ερεθισμένες από έντονα χρώματα που εναλλάσσονται γρήγορα ή φώτα που αναβοσβήνουν.
- **Νοητικοί Περιορισμοί** – Υπάρχουν διάφορες παθήσεις που επηρεάζουν την νοητική ικανότητα, όπως π.χ. δυσλεξία.

Για την αντιμετώπιση αυτών των προβλημάτων πολλοί χρήστες χρησιμοποιούν βοηθητικές τεχνολογίες για να προσπελάσουν το διαδίκτυο. Εφαρμοσμένη τεχνολογία σε αυτό τον τομέα μπορεί να έχει την μορφή εργαλείων και συσκευών που χρησιμοποιούνται από άτομα με ειδικές ανάγκες για την βελτίωση της εμπειρίας τους στο διαδίκτυο. Τέτοια εργαλεία επιτρέπουν σε χρήστες να πραγματοποιήσουν ενέργειες που δεν μπορούσαν προηγουμένως, είτε βελτιώνουν την διεκπεραίωση της ενέργειας αυτής, κάνοντας την απλούστερη ή γρηγορότερη. Για παράδειγμα, άτομα με περιορισμένη όραση ίσως να μην μπορούν να δουν το κείμενο και τις εικόνες σε μια σελίδα. Για αυτό το σκοπό μπορούν να χρησιμοποιήσουν αναγνώστες θόνης (Screen Readers) η τερματικά Braille. Υπότιτλοι ή νοηματική γλώσσα μπορούν να βοηθήσουν χρήστες με περιορισμένη ακοή όταν προσπελούν ηχητικό περιεχόμενο στο διαδίκτυο [60].

Τέτοια εργαλεία θα πρέπει να υπακούν σε συγκεκριμένες κατευθυντήριες γραμμές, ώστε να εγγυάται η αποτελεσματικότητά τους. Έχουν δημοσιευθεί διαφορετικές παραλλαγές Accessibility Guidelines, με πληρέστερες τις οδηγίες του Web Accessibility Initiative (WAI). Πρόκειται για το WCAG, Web Content Accessibility Guidelines. Σχεδιασμένες σε συνεργασία με οργανισμούς σε όλο τον κόσμο με τον στόχο της παροχής ενός ενιαίου προτύπου για διαδικτυακή προσβασιμότητα. Εξηγεί πώς να τροποποιηθεί το OnLine περιεχόμενο ώστε να είναι προσπελάσιμο από άτομα με ειδικές ανάγκες. Ως περιεχόμενο τυπικά αναφέρεται σε

πληροφορίες όπως κείμενο, εικόνες και ήχους, και κώδικα που περιγράφει την δομή και παρουσίαση πληροφοριών σε σελίδες. Τρέχουσα έκδοση είναι η 2.1 και αναμένεται η έκδοση 2.2 εντός του 2021 [61].

7.2 ΒΕΛΤΙΩΣΗ ΠΡΟΣΒΑΣΙΜΟΤΗΤΑΣ ΙΣΤΟΤΟΠΟΥ

Πέρα από την κατασκευή συσκευών και μηχανημάτων προσαρμοσμένα στο να βοηθήσουν ομάδες χρηστών με ειδικές ανάγκες να προσπελάσουν το διαδίκτυο, και πέρα από την ανάπτυξη εφαρμογών, προσθέτων και επεκτάσεων για να βελτιώσουν την εμπειρία αυτών των χρηστών OnLine, υπάρχουν κάποιες βασικές αρχές σχεδίασης και κατασκευής ενός ιστότοπου που μπορούμε να εφαρμόσουμε κατά το Development του ιστότοπου για να βελτιώσουμε σημαντικά την προσβασιμότητά του.

7.2.1 Επιλογή Κατάλληλου CMS

Υπάρχουν πολλά διαθέσιμα συστήματα διαχείρισης, και τα πλέον δημοφιλέστερα όλα προσφέρουν λύσεις για την βελτίωση της προσβασιμότητας μιας σελίδας. Κατόπιν επιλογής CMS πρέπει να επιλέξουμε ένα θέμα/αρχέτυπο με λειτουργικότητα που επίσης βοηθά σε αυτό τον σκοπό, καθώς και αντίστοιχα πρόσθετα και επεκτάσεις με τέτοια λειτουργικότητα.

7.2.2 Αποτελεσματική Χρήση Επικεφαλίδων

Εφαρμογές ανάγνωσης οθόνης (Screen Readers) μπορούν να χρησιμοποιηθούν για την πλοήγηση μιας ιστοσελίδας. Με την χρήση σωστών επικεφαλίδων – δηλαδή HTML elements όπως <h1>, <h2> - στα κατάλληλα σημεία, το περιεχόμενο της σελίδας μπορεί να είναι καλά οργανωμένο και κατά συνέπεια εύκολα κατανοητό και αναγνώσιμο από Screen Readers. Είναι σημαντικό να χρησιμοποιείται η κατάλληλη σειρά κεφαλίδων και να διαχωρίζεται η δομή από την εμφάνιση, με την χρήση CSS. Παραδείγματα καλής χρήσης είναι:

- Αποφυγή της χρήσης του HTML tag <h1> για οτιδήποτε πέρα από τον τίτλο του ιστότοπου και τους τίτλους των ιστοσελίδων.
- Χρήση των tag επικεφαλίδων για την οργάνωση της δομής του περιεχομένου.
- Διατήρηση της συνοχής μεταξύ tags. Δηλαδή να μην προσπερνάμε ένα επίπεδο tags (π.χ. πηγαίνοντας κατευθείαν από <h1> σε <h3>) γιατί οι screen readers θα ερμηνεύσουν ότι λείπει περιεχόμενο από την σελίδα [62].

7.2.3 Χρήση Εναλλακτικού Κειμένου για Εικόνες

Εναλλακτικό κείμενο (Alt text) πρέπει να είναι διαθέσιμο έτσι ώστε οι Screen Readers να μπορούν να καταλάβουν το νόημα μίας εικόνας στο περιεχόμενο. Αυτό είναι ιδιαίτερα σημαντικό για εικόνες με πληροφορίες ή στατιστικά στοιχεία. Εάν η εικόνα απεικονίζει κείμενο, τότε το κείμενο αυτό θα πρέπει να υπάρχει στο Alt text. Εξαίρεση σε αυτό τον κανόνα

είναι όταν μια εικόνα χρησιμοποιείται καθαρά για διακοσμητικούς σκοπούς, είτε για να βελτιώσει την εμφάνιση της σελίδας, είτε εάν απεικονίζει πληροφορίες αλλά οι πληροφορίες αυτές υπάρχουν και σε μορφή κειμένου δίπλα στην εικόνα. Σε αυτές της περιπτώσεις, το πεδίο alt text μπορεί να μένει κενό (alt=""), ώστε οι αναγνώστες οθόνης να το αγνοήσουν. Η παράλειψη του Alt δεν είναι καλή επιλογή γιατί όταν δεν υπάρχει, κάποιος αναγνώστης θα ανακοινώσει το όνομα του αρχείου εικόνας στη θέση του [63], και το ίδιο ισχύει για εικόνες που χρησιμοποιούνται για συνδέσμους [62].

7.2.4 Χρήση μοναδικών και περιγραφικών ονομάτων Συνδέσμων

Όταν συμπεριλαμβάνουμε συνδέσμους (Links) σε μια σελίδα πρέπει να επιλέξουμε προσεκτικά ποιες λέξεις θα αποτελέσουν τον σύνδεσμο αυτόν. Ως γενικό κανόνα ενδείκνυται η χρήση λέξεων κλειδιών. Χρήση keywords σε συνδέσμους:

- Βοηθά κατά την διαδικασία του SEO.
- Δίνει περισσότερες πληροφορίες σε χρήστες με λειτουργική όραση που προσπελαίνουν την ιστοσελίδα.
- Παρέχει πληροφορίες για το πλαίσιο αναφοράς στο οποίο υπάρχει ο σύνδεσμος, σε χρήστες που μπορεί να χρησιμοποιούν βοηθητικές τεχνολογίες.

Οι σύνδεσμοι θα πρέπει να είναι κατανοητοί και εκτός πλαισίου αναφοράς. Ο λόγος είναι ότι το λογισμικό ανάγνωσης συχνά θα μετακινηθεί από σύνδεσμο σε σύνδεσμο, παραλείποντας το υπόλοιπο κείμενο. Θα διαβάσουν ηχητικά κάθε σύνδεσμο κατά σειρά.

Για παράδειγμα, εάν το κείμενο ενός συνδέσμου είναι «Athens», ο αναγνώστης θα διαβάσει «link Athens», και ο χρήστης θα κατανοήσει ότι η σελίδα στην οποία συνδέεται πιθανότατα αφορά την Αθήνα. Εάν αντιθέτως το κείμενο είναι «Click Here», ο αναγνώστης θα διαβάσει «link Click Here», και ο χρήστης δεν θα καταλάβει σε τι αναφέρεται η συνδεόμενη σελίδα.

Ένα ακόμα χαρακτηριστικό που μπορεί να εκμεταλλευτεί ένας Screen Reader, είναι η ικανότητα ταξινόμησης συνδέσμων αλφαβητικά. Αυτό είναι ιδιαίτερος χρήσιμο όταν ο χρήστης έχει ήδη ένα συγκεκριμένο σύνδεσμο κατά νου. Οι πιο σημαντικές λέξεις-κλειδιά είναι καλύτερο να βρίσκονται στην αρχή του συνδέσμου. Παραδείγματος χάριν, ο σύνδεσμος «Staff» θα εμφανιστεί εκεί όπου είναι αναμενόμενο αλφαβητικά. Ο σύνδεσμος «Take a look at our Staff» θα βρίσκεται σε λάθος σημείο [64].

7.2.4.1 Σύνδεσμοι Εικόνας

Για κάποιες ιστοσελίδες, μπορεί να είναι προτιμότερη η χρήση εικόνων για συνδέσμους αντί για χρήση κειμένου. Σε αυτές τις περιπτώσεις ο αναγνώστης θα διαβάσει το Alt text της εικόνας ως το κείμενο συνδέσμου. Το εναλλακτικό κείμενο θα πρέπει να περιέχει και το περιεχόμενο αλλά και τη λειτουργία της εικόνας. Εάν η εικόνα είναι διακοσμητική, μόνο η λειτουργία της είναι απαραίτητη [64]. Για παράδειγμα:

- **Διακοσμητική Εικόνα** - `<p></p>`

- **Λειτουργική εικόνα** - `<p></p>`

7.2.4.2 ARIA

ARIA (Accessible Rich Internet Applications) είναι ένα σύνολο ιδιοτήτων για την βελτίωση της προσβασιμότητας σε στοιχεία που δεν την υποστηρίζουν εκ φύσεως. Πολλά από τα στοιχεία που απαιτούσαν τη χρήση ARIA πλέον έχουν εφαρμοστεί στην HTML 5. [65]

Σε κάποιες περιπτώσεις μπορεί να θέλουμε να έχουμε αόριστο κείμενο σε έναν σύνδεσμο, όπως “[read more]”. Σε αυτές τις περιπτώσεις μπορούμε να χρησιμοποιήσουμε το στοιχείο **aria-label** για να ενημερώσουμε τους αναγνώστες οθόνης για το πλαίσιο αναφοράς του συνδέσμου. Οι περισσότεροι Screen Readers θα διαβάσουν το aria-label αντί για το κείμενο του συνδέσμου. Για παράδειγμα:

```
<p class="right"><a href="/tour/Athens/" aria-label="Athens"> [read more] </a></p>
```

Σε αυτό το παράδειγμα, ένας τυπικός χρήστης θα δει τον σύνδεσμο που αναγράφει [read more], αλλά ένας Screen Reader θα διαβάσει «link Athens». [64]

7.2.5 Χρήση Χρώματος

Όταν σχεδιάζουμε ιστοσελίδες και διαδικτυακό περιεχόμενο πρέπει να σκεφτούμε προσεκτικά την χρήση χρωμάτων. Οι χρωματικές επιλογές μας μπορεί να επηρεάσουν σοβαρά χρήστες με αχρωματοψία, περιορισμένη όραση ή άλλες παθήσεις σχετικές με χρωματική αντίληψη.

7.2.5.1 Χρωματική αντίθεση – Contrast

Χρωματική αντίθεση ή Contrast είναι ο βαθμός της διαφοράς (Contrast Ratio) μεταξύ των χρωμάτων προσκηνίου και παρασκηνίου. Για παράδειγμα η χρήση κίτρινου συνδέσμου (hex code #ffc00) σε μαύρο background (hex code #00000) υπολογίζεται σε βαθμό contrast 13.89:1. Σύμφωνα με τις κατευθυντήριες οδηγίες WCAG 2.0 προτείνονται συγκεκριμένοι επιθυμητοί βαθμοί contrast ως εξής:

- **Contrast Ratio 4.5:1** για κανονικό κείμενο (σώμα ιστοσελίδας, τυπικοί σύνδεσμοι κλπ.).
- **Contrast Ratio 3:1** για μεγάλο κείμενο (**Bold** κείμενο με μέγεθος γραμματοσειράς 18px η μεγαλύτερο, και κανονικό κείμενο 24px η μεγαλύτερο).

Επίσης πρέπει να υπάρχει αντίθεση μεταξύ διαφορετικών στοιχείων σελίδας, όπως κουμπιά και συνδέσμους, ώστε οι χρήστες με περιορισμένη όραση να μπορούν να τα εντοπίσουν και να αλληλοεπιδράσουν μαζί τους. Μπορούμε να εξετάσουμε την χρωματική αντίθεση της σελίδας μας με χρήση συγκεκριμένων εργαλείων, όπως το WebAIM Contrast Checker [66].

7.2.5.2 Χρώμα ως μέσο επικοινωνίας

Πολλές φορές χρησιμοποιούμε διαφορετικούς χρωματισμούς για να περάσουμε διαφορετικά μηνύματα στον χρήστη. Ωστόσο η χρήση χρώματος δεν πρέπει να είναι η μοναδική επιλογή επικοινωνίας. Οι τρόποι με τους οποίους χρησιμοποιείται χρώμα σε ιστοσελίδες είναι την υπόδειξη λειτουργίας, και υπόδειξη νοήματος. Αναλόγως με την ζωτικότητα του χρώματος για την πληροφορία κρίνεται και το εάν θα πρέπει να συμπεριληφθεί στο alt text.

Ένα παράδειγμα υπόδειξης λειτουργίας είναι η χρήση διαφορετικού χρώματος σε συνδέσμους. Παραδοσιακά στο διαδίκτυο χρησιμοποιείται το μπλε χρώμα για Links, ωστόσο εάν κάποιος χρήστης δεν μπορεί να αντιληφθεί το μπλε, δεν θα μπορέσει να εντοπίσει τον σύνδεσμο. Για να σιγουρευτούμε για την προσβασιμότητα των συνδέσμων μας πρέπει να χρησιμοποιήσουμε και μια δευτερεύουσα ένδειξη ανεξάρτητη από χρώμα, όπως υπογράμμιση. Αντιστοίχως αποθαρρύνεται η χρήση χρώματος ως το μοναδικό μέσο υπόδειξης νοήματος. Εάν έχουμε μια σελίδα που αναπαριστά διαφορετικές θερμοκρασίες με μια κλίμακα από πράσινο έως κόκκινο, ένας χρήστης με σχετικούς περιορισμούς όρασης δεν θα μπορέσει να λάβει την πληροφορία που αναπαριστά η σελίδα, οπότε μια λύση είναι να συμπεριλάβουμε και τις αριθμητικές τιμές τις θερμοκρασίας. [67]

7.2.6 Φόρμες

Εάν οι φόρμες μιας σελίδας δεν έχουν άρτιο κώδικα μπορεί να παρουσιάσουν συγκεκριμένες δυσκολίες σε χρήστες που χρησιμοποιούν Screen Readers. Το λογισμικό πρέπει να μπορεί να κατανοήσει διάφορα στοιχεία της φόρμας, όπως:

- Ποια labels ανήκουν σε ποια πεδία
- Ποια πεδία είναι υποχρεωτικά
- Ποια πεδία είναι ομαδοποιημένα (fieldsets)
- Ποια πεδία έχουν σφάλματα επαλήθευσης

7.2.6.1 Μορφολογία

Πρέπει να αποφεύγεται η χρήση tables για την δομή μιας φόρμας. Λάθος χρήση tabular κώδικα μπορεί να δημιουργήσει σφάλματα που καθιστούν την φόρμα μη προσπελάσιμη. Αντ' αυτού, προτείνεται η χρήση ενός συνδυασμού HTML /CSS , με χρήση <div> ως δομικά συστατικά.

Μια προσβάσιμη φόρμα δημιουργείται με καλή χρήση HTML markup. Κάθε πεδίο/input μιας φόρμας πρέπει να έχει ένα label σωστά συνδεδεμένο μαζί του. Ομάδες πεδίων, όπως σει radio ή checkboxes πρέπει να εμφωλεύονται σε στοιχείο fieldset με περιγραφή legend. Παραδείγματος χάριν:

- **Πεδίο Input με label**

```

<form>
  <div> // layout div
    <label for="name">Name</label> // label associated with specific input field
    <input id="name" type="text" />
  </div>
</div>

```

```

<label for="email">Email</label>
<input id="email" type="email" />
</div>
</form>

```

Στο παραπάνω παράδειγμα συσχετίζουμε ένα label με ένα πεδίο input, με την χρήση των στοιχείων **for** και **id**. Οι Screen Readers χρησιμοποιούν αυτόν τον συσχετισμό για να περιγράψουν στον χρήστη ποια πεδία πρέπει να συμπληρωθούν με ποιόν τύπο στοιχείων.

- **Ομαδοποιημένα Checkbox**

```

<form>
  <div> // layout div
    <fieldset> // Fieldset wraps group
      <legend>Colors</legend> // Legend provides a label for group
      <label for="blue">Blue</label>
      <input id="blue" type="checkbox" />
      <label for="green">Green</label>
      <input id="green" type="checkbox" />
      <label for="purple">Purple</label>
      <input id="purple" type="checkbox" />
    </fieldset>
  </div>
</form>

```

Στο παραπάνω παράδειγμα χρησιμοποιούμε το στοιχείο fieldset για να ομαδοποιήσουμε συγγενικά στοιχεία υπό ένα ομαδικό label. Ένα fieldset συχνά χρησιμοποιείται για σελ από checkboxes και radio buttons.

1. Απαιτούμενα Πεδία

```

<form>
  <p>* Required fields</p>
  <div> // layout div
    <label for="name">Name*</label>
    <input id="name" type="text" aria-required="true" />
  </div>
  <div>
    <label for="email">Email*</label>
    <input id="email" type="email" aria-required="true" />
  </div>
</div>

```

```

<label for="phone">Phone Number</label>
<input id="phone" type="text" />
</div>
</form>

```

Στο παραπάνω παράδειγμα χρησιμοποιούμε το στοιχείο “aria-required”, που είναι αυτομάτως ορισμένο ως «false». Αλλάζοντας την αξία σε «true» ενημερώνουμε τους screen readers όσον αφορά ποια πεδία είναι υποχρεωτικά.

2. Επαλήθευση Φόρμας

```

<form>
<p>* Required fields</p>
<div> // layout div
  <label for="name">Name*</label>
  <input id="name" type="text" aria-invalid="true" />
  <p>This field is required</p>
</div>
<div>
  <label for="email">Email*</label>
  <input id="email" type="email" aria-invalid="true" />
  <p>Please enter a valid email address</p>
</div>
<div>
  <label for="phone">Phone Number</label>
  <input id="phone" type="text" />
</div>
</form>

```

Εάν η φόρμα μας χρησιμοποιεί επαλήθευση στοιχείων (είτε κατά την πληκτρολόγηση είτε κατά την κατάθεση στοιχείων) είναι απαραίτητο να ενημερώσουμε τον Screen Reader χρησιμοποιώντας την ιδιότητα «aria-invalid». Αυτομάτως η ιδιότητα είναι «false». Θα πρέπει να την ορίσουμε ως «true» για κάθε πεδίο που παρουσιάζει εσφαλμένα στοιχεία. Αυτό πρέπει να γίνει με χρήση JavaScript. Τέλος, αν ένα πεδίο γίνει έγκυρο εκ των υστέρων, καθώς ο χρήστης πληκτρολογεί/εισάγει πληροφορίες, η ιδιότητα aria-invalid πρέπει να οριστεί ξανά ως false.

7.2.7 Συμβατότητα Πληκτρολογίου

Όλη η λειτουργικότητα μιας σελίδας πρέπει να είναι προσβάσιμη μέσω πληκτρολογίου. Αυτό σημαίνει ότι οι χρήστες πρέπει να μπορούν να προσπελάσουν συνδέσμους, κουμπιά, φόρμες και άλλα στοιχεία της σελίδας χρησιμοποιώντας το κλειδί Tab, τα Arrow keys, και παρόμοιες μεθόδους εισόδου. Οι σελίδες δεν πρέπει να απαιτούν τη χρήση ποντικιού [68].

Όταν ο χρήστης φτάσει σε ένα αντικείμενο με τη χρήση Tab, το αντικείμενο γίνεται «συγκεντρωμένο» (focused), και μπορεί να ενεργοποιηθεί, ή να επεξεργασθεί μέσω πληκτρολογίου. Ένας χρήστης πρέπει επίσης να δεχτεί και οπτικό ερέθισμα που να τον ενημερώνει ότι το στοιχείο είναι πλέον focused. Τέτοιες ενδείξεις παρέχονται αυτομάτως από web browsers, και παρόλο που η εμφάνισή τους ποικίλλει, τυπικά εμφανίζονται ως ένα περίγραμμα γύρω από το εν λόγω στοιχείο. Αυτά τα περιγράμματα μπορούν να αποκρυφθούν με τη χρήση CSS, με τις εντολές «outline:0» ή «outline:none» σε στοιχεία που μπορούν να γίνουν focused. Επίσης με την χρήση CSS μπορούμε να κάνουμε τις ενδείξεις αυτές πιο προβεβλημένες και με υψηλότερη χρωματική αντίθεση [69].

7.2.7.1 Πλοήγηση στη σελίδα

Καθώς ένας χρήστης μετακινείται μέσα σε μια ιστοσελίδα, η σειρά με την οποία τα στοιχεία της σελίδας γίνονται focused είναι πολύ σημαντική. Η σειρά πλοήγησης πρέπει να είναι λογική, που συνήθως σημαίνει ότι ακολουθεί την οπτική ροή της σελίδας. Από αριστερά προς δεξιά, από πάνω προς τα κάτω. Για τις περισσότερες σελίδες, αυτό σημαίνει Header πρώτα, μετά κύρια πλοήγηση (top menu bar), κυρίως περιεχόμενο, και μετά footer. Η σειρά αυτή καθορίζεται από τον πηγαίο κώδικα της σελίδας [70].

Σύνδεσμοι, κουμπιά και στοιχεία ελέγχου φόρμας είναι εκ φύσεως προσβάσιμα σε χρήστες πληκτρολογίου, αλλά στοιχεία που δεν είναι διαδραστικά για χρήστες Mouse ή αφής δεν θα πρέπει να ορίζονται ως focusable (χρησιμοποιώντας tabindex π.χ.). Η μετατροπή μη-διαδραστικών στοιχείων σε πλοηγήσιμα στοιχεία μέσω πληκτρολογίου μπορεί να δημιουργήσει σύγχυση σε χρήστες. [69]

7.3 ΈΛΕΓΧΟΣ ΠΡΟΣΒΑΣΙΜΟΤΗΤΑΣ ΣΕΛΙΔΑΣ

Μπορούμε να ελέγξουμε την κατάσταση της σελίδας μας όσον αφορά την προσβασιμότητά με διάφορους τρόπους και εργαλεία.

1. Επαλήθευση HTML – Εάν η εφαρμογή της HTML σε έναν ιστότοπο έχει γίνει λανθασμένα, βοηθητική τεχνολογία μπορεί να αντιμετωπίσει προβλήματα κατά την ανάγνωση του περιεχομένου.
2. Δοκιμή με Πληκτρολόγιο – Χρησιμοποιώντας το πλήκτρο tab για να πλοηγηθούμε στη σελίδα μας θα πρέπει να μπορούμε να αλληλοεπιδράσουμε με όλα τα διαδραστικά στοιχεία της. Εάν αυτό δεν είναι δυνατό, η σελίδα μας έχει κενά προσβασιμότητας.
3. Χρήση εργαλείου Accessibility Checker – Υπάρχουν αρκετά δωρεάν OnLine εργαλεία που μπορούν να ελέγξουν την προσβασιμότητα της σελίδας μας, όπως:
 - WAVE Web Accessibility Tool
 - Accessibility Developer Tools (Google Chrome Add-on)
 - Color Contrast Analyzer (single ratio)
4. Δοκιμή με Screen Reader – Μπορούμε να πλοηγηθούμε στη σελίδα μας μέσω screen reader για να διαπιστώσουμε σημαντικές ελλείψεις, ή να συνεργαστούμε με τακτικό χρήστη λογισμικού Screen Reader.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. **Wikipedia**. Internet. *Wikipedia*. [Ηλεκτρονικό] 2020. <https://en.wikipedia.org/wiki/Internet>.
2. —. World Wide Web. *Wikipedia*. [Ηλεκτρονικό] 2020. https://en.wikipedia.org/wiki/World_Wide_Web.
3. **StatCounter**. Browser Market Share Worldwide. *Statcounter*. [Ηλεκτρονικό] 2020. <https://gs.statcounter.com/>.
4. **Wikipedia**. Web Browser. *Wikipedia*. [Ηλεκτρονικό] 2020. https://en.wikipedia.org/wiki/Web_browser.
5. **Patrick, Killelea**. *Web performance tuning (2nd ed.)*. Beijing : O'Reilly, 2002.
6. **Wikipedia**. Web Server. *Wikipedia*. [Ηλεκτρονικό] 2020. https://en.wikipedia.org/wiki/Web_server.
7. —. Website. *Wikipedia*. [Ηλεκτρονικό] 2020. <https://en.wikipedia.org/wiki/Website>.
8. **Bansal, Ayush**. Static vs Dynamic Website. *Gekksforgeeks*. [Ηλεκτρονικό] 2020. <https://www.geeksforgeeks.org/static-vs-dynamic-website/>.
9. **Ullah, Rebecca**. Static vs Dynamic Websites. *Fasthosts*. [Ηλεκτρονικό] April 2020. <https://www.fasthosts.co.uk/blog/static-vs-dynamic-websites/>.
10. *Web Content Management System*. **Srivastav, Manoj Kumar και Nath, Asoke**. 2016, International Journal of Innovative Research in Advanced Engineering (IJRAE), σ. 7.
11. **Fitzgerald, Anna**. What is a CMS and why should you care. *hubspot.com*. [Ηλεκτρονικό] April 2020. <https://blog.hubspot.com/blog/tabid/6307/bid/7969/what-is-a-cms-and-why-should-you-care.aspx>.
12. **CMS UK**. Types Of Content Management Systems. *cms.co.uk*. [Ηλεκτρονικό] 2021. <http://www.cms.co.uk/types/>.
13. **w3techs.com**. Usage statistics of content management systems. *w3techs*. [Ηλεκτρονικό] 2020. https://w3techs.com/technologies/overview/content_management.
14. **Shivakumar, S.K.** *Enterprise Content and Search Management for Building Digital Platforms*. Hoboken, NJ, USA : John Wiley & Sons, 2016. ISBN 1119206812.
15. **Martinez-Caro, Jose-Manuel, και συν.** *A Comparative Study of Web Content Management Systems*. s.l. : MDPI, 2018.
16. **OpenSourceCMS**. Open-source CMS Market Share. *opensourcecms.com*. [Ηλεκτρονικό] 2020. <https://www.opensourcecms.com/cms-market-share/>.
17. **CodeCademy**. What is Rest. *CodeCademy*. [Ηλεκτρονικό] <https://www.codecademy.com/articles/what-is-rest>.
18. **Fielding, Roy**. *Architectural Styles and the Design of Network-Based Software Architectures*. s.l. : University of California, Irvine, 2000. OCLC 45706361.

19. **McCollin, Rachel.** The complete Guide to Wordpress Rest API Basics. *kinsta*. [Ηλεκτρονικό] 2021. <https://kinsta.com/blog/wordpress-rest-api/>.
20. **RedHat.** What is a rest API. *Redhat.com*. [Ηλεκτρονικό] 2021. <https://www.redhat.com/en/topics/api/what-is-a-rest-api>.
21. **Wikipedia.** JSON. *wikipedia.org*. [Ηλεκτρονικό] 2017. <https://el.wikipedia.org/wiki/JSON>.
22. **JSON.org.** Introducing JSON. *Json.org*. [Ηλεκτρονικό] 2021. <https://www.json.org/json-en.html>.
23. **RESTFULAPI.** Introduction to JSON. *restfulapi.net*. [Ηλεκτρονικό] 2021. <https://restfulapi.net/introduction-to-json/>.
24. **WordPress.** Rest API handbook. *wordpress.org*. [Ηλεκτρονικό] 2021. <https://developer.wordpress.org/rest-api/>.
25. **Mozilla.** What is AJAX. *mozilla.org*. [Ηλεκτρονικό] 2021. https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started.
26. **Herzinger, M.** "The past, present, and future of Web search engines." *31st International Colloquium of Automata, Languages and Programming*, p. 3. 2004.
27. **Gaur D. and Dagar, P.** "SEO techniques: New dimensions for popular search engines." *International Journal on Advanced Computer Theory and Engineering*, 3(4): 32-36. 2014.
28. **Safran, N.** 310 Million Visits: Nearly Half of All Web Site Traffic Comes From Natural Search (Data). [Ηλεκτρονικό] 2013. <http://www.conductor.com/blog/2013/06/data-310-million-visits-nearly-half-of-all-web-site-traffic-comes-from-nat>.
29. **Newman, N., Levy A. L. D.** Reuters Institute Digital News Report 2014 – Tracking the future of news. *Reuters Institute for the Study of Journalism, University of Oxford*. [Ηλεκτρονικό] 2014. <https://reutersinstitute.politics.ox.ac.uk/sites/default/files/Reuters%20Institute%20Digital%20News%20Report%2>.
30. *Sustainable Online News Projects: Redefining Production Norms and Practices*. **Spyridou, P. και Veglis, A.** Thessaloniki : s.n., 2012. Proceedings of the 10th World Media Economics and Management Conference.
31. *"Finding What People Want: Experiences with the WebCrawler"*. **Pinkerton, Brian.** Chicago, USA : s.n., 2007. The Second International WWW Conference.
32. *"Desperately Seeking Surfers"*. **Flynn, Laurie J.** 1996, New York Times.
33. **Google.** "Google's Guidelines on Site Design". *Google*. [Ηλεκτρονικό] 2020. <https://support.google.com/webmasters/answer/35769>.
34. **Bing.** Bing Webmaster Guidelines. *Bing.com*. [Ηλεκτρονικό] 2020. <https://www.bing.com/webmaster/help/webmaster-guidelines-30fba23a>.
35. **Google.** Sitemaps. *Google*. [Ηλεκτρονικό] 2020. https://support.google.com/webmasters/answer/156184?hl=en&ref_topic=4581190.

36. **Broadbent, Andrew.** "By the Data: For Consumers, Mobile is the Internet". *Startupgrind*. [Ηλεκτρονικό] 2017. <https://www.startupgrind.com/blog/mobile-is-the-internet-for-consumers/>.
37. *A study of Search Engine Optimization methods.* **Rogan, G.** 2009, National University of Ireland, Gateway.
38. **Julien, Masanes.** "*Web Archiving*", ISBN 978-3-54046332-0. s.l. : Springer, 2007.
39. "*Review of Web Crawlers with Specification and Working*". **Patil, Yugandhara και Patil, Sonal.** 2016, International Journal of Advanced Research in Computer and Communication Engineering.
40. **Enge, E., και συν.** "*The Art of SEO: Mastering Search Engine Optimization*", ISBN 978-0-596-51886-8. s.l. : O'Reilly Media Inc., 2009.
41. **Ledford, J. L.** "*SEO: Search Engine Optimization Bible*". s.l. : John Wiley & Sons Inc. Publishing USA, ISBN 0470452641, 2009.
42. **McGinley, Corinne.** SEO vs. PPC: When to Optimize and When to Pay for Traffic. *HubSpot*. [Ηλεκτρονικό] February 2020. <https://blog.hubspot.com/marketing/tabid/6307/bid/1514/paid-search-vs-organic-search.aspx>.
43. **Sarcona, Cliff.** Organic vs PPC in 2020: The CTR Results. *Zero Limit Web*. [Ηλεκτρονικό] March 2020. <https://www.zerolimitweb.com/organic-vs-ppc-2020-ctr-results-best-practices/>.
44. *Classifying Web Search Queries in Order to Identify High Revenue Generating Customers.* **A., Ortiz-Cordova και J., Jansen B.** 2012, Journal of the American Society for Information Sciences and Technology.
45. **Britney Muller.** How Search Engines Work: Crawling, Indexing, and Ranking. *moz*. [Ηλεκτρονικό] <https://moz.com/beginners-guide-to-seo/how-search-engines-operate>.
46. **Google.** Introduction to robots.txt. *Google Help Center*. [Ηλεκτρονικό] 2020. <https://support.google.com/webmasters/answer/6062608?hl=en>.
47. **Timothy Carter.** On-Site SEO – A Comprehensive Guide to On-Page Optimization. *seo.co*. [Ηλεκτρονικό] 2020. <https://seo.co/on-site-seo/>.
48. **Brian Dean.** Keyword Research: The Definitive Guide. *backlinko*. [Ηλεκτρονικό] 2020. <https://backlinko.com/keyword-research>.
49. **Pavlik, Vlado.** Keyword Research for SEO. *mangools.com*. [Ηλεκτρονικό] 2020. <https://mangools.com/blog/keyword-research/>.
50. **Google.** *Quality Rater's Guidelines*. 2019.
51. **Britney Muller.** Beginners Guide to Keyword Research. *moz*. [Ηλεκτρονικό] 2020. <https://moz.com/beginners-guide-to-seo/keyword-research>.
52. *American Marketing Association Dictionary.* **American Marketing Association.** 2012.

53. **Lin, Jave.** "Increase your Organic Traffic using CTR Magnet Method now!". [Ηλεκτρονικό] 2016. <https://www.linkedin.com/pulse/increase-your-organic-traffic-using-ctr-magnet-method-jave-lin?published=t>.
54. **Moz.** On-Site SEO. *moz.com*. [Ηλεκτρονικό] 2020. <https://moz.com/learn/seo/on-site-seo>.
55. **Decker, Allie.** The Ultimate Guide to On-Page SEO in 2020. *Hubspot*. [Ηλεκτρονικό] 2020. <https://blog.hubspot.com/blog/tabid/6307/bid/33655/a-step-by-step-guide-to-flawless-on-page-seo-free-template.aspx>.
56. **Search Engine Land team.** Essential Guide to SEO: HTML Code & Search Engine Success Factors. *SearchEngineLand*. [Ηλεκτρονικό] 2020. <https://searchengineland.com/guide/seo/html-code-search-engine-ranking>.
57. **Google.** Rolling out mobile-first Indexing. *webmasters.googleblog.com*. [Ηλεκτρονικό] 2018. <https://webmasters.googleblog.com/2018/03/rolling-out-mobile-first-indexing.html?fbclid=IwAR1xRCLuKhAZi325AhcfTGJgC8iCdR48gfWZF1Sa4LMBQAnOBe2Nsp-KTbg>.
58. **Botify.** SEO Return on Investment: What's the ROI of SEO & How Can I Measure It? *botify*. [Ηλεκτρονικό] 2020. <https://www.botify.com/blog/roi-seo>.
59. **Al-Khalifa, H.** WCAG 2.0 Semi-automatic Accessibility Evaluation System: Design and Implementation. *Computer and Information Science, Vol. 5*. 2012.
60. **Elkabani, Islam, και συν.** *Toward Better Web Accessibility*. 2016.
61. **WAI.** Web Content Accessibility Guidelines (WCAG) Overview. *w3.org*. [Ηλεκτρονικό] 2020. <https://www.w3.org/WAI/standards-guidelines/wcag/>.
62. **Berkeley WebAccess.** Making your Website Accessible. *berkeley.edu*. [Ηλεκτρονικό] 2020. <https://webaccess.berkeley.edu/resources/tips/web-accessibility>.
63. **w3.org.** Decorative Images. *w3.org*. [Ηλεκτρονικό] 2020. <https://www.w3.org/WAI/tutorials/images/decorative/>.
64. **Michigan Tech.** Descriptive Link Text. *mtu.edu*. [Ηλεκτρονικό] 2020. <https://www.mtu.edu/accessibility/training/web/link-text/>.
65. **Mozilla.org.** ARIA. *mozilla.org*. [Ηλεκτρονικό] 2020. <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA>.
66. **WebAIM.** Introduction to Web Accessibility. *webaim.org*. [Ηλεκτρονικό] April 2020. <https://webaim.org/intro/>.
67. **Michigan Tech.** Accessible Technology. *mtu.edu*. [Ηλεκτρονικό] 2020. <https://www.mtu.edu/accessibility/training/web/color/>.
68. **Web Accessibility Initiative.** Keyboard Compatibility. *w3.org*. [Ηλεκτρονικό] 2020. <https://www.w3.org/WAI/perspective-videos/keyboard/>.
69. **WebAim.** Keyboard Accessibility. *webaim.org*. [Ηλεκτρονικό] 2021. <https://webaim.org/techniques/keyboard/>.

70. **University of Washington.** Designing for Keyboard Accessibility. *washington.edu*. [Ηλεκτρονικό] 2020. <https://www.washington.edu/accessibility/checklist/keyboard/>.