



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΙΩΑΝΝΙΝΩΝ

ΣΧΟΛΗ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Web Design. Ανάπτυξη ιστοσελίδας με WordPress και οι διαφορές
Front-End και Back-End Developer**

Πολύζος Ευάγγελος

ΑΜ:1207

Επιβλέπων: Γιαννακέας Νικόλαος

Άρτα, Απρίλιος 2022

**Web Design. Website development with WordPress and the differences
Front-End and Back-End Developer**

Εγκρίθηκε από τριμελή εξεταστική επιτροπή

Άρτα, Απρίλιος 2022

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. Επιβλέπων καθηγητής

Νικόλαος Γιαννακέας,

Επίκουρος Καθηγητής

2. Μέλος επιτροπής

Τζάλλας Αλέξανδρος,

Επίκουρος Καθηγητής

3. Μέλος επιτροπής

Δημήτριος Δημόπουλος,

Πανεπιστημιακός Υπότροφος

Ο Προϊστάμενος του Τμήματος

Ευριπίδης Γλαβάς,

Καθηγητής, Α' Βαθμίδας

Υπογραφή

© Πολύζος, Ευάγγελος, 2022.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Δήλωση μη λογοκλοπής

Δηλώνω υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν. 2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα μεταπτυχιακή εργασία είναι εξ ολοκλήρου αποτέλεσμα δικής μου ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία.

Πολύζος, Ευάγγελος

Υπογραφή

ΕΥΧΑΡΙΣΤΙΕΣ

Οι πρώτοι άνθρωποι που θέλω να ευχαριστήσω με αφορμή αυτή την εργασία είναι οι γονείς και τα αδέρφια μου, καθώς στις στιγμές των σπουδών μου και σε όλη τη διάρκεια των μαθητικών μου χρόνων ήταν δίπλα μου σε ότι και αν χρειάστηκα και πίστεψαν σε εμένα. Επιπλέον θα ήθελα να ευχαριστήσω τον επιβλέπων καθηγητή μου, κύριο Νικόλαο Γιαννακέα γιατί η βοήθεια που μου παρείχε ήταν κάτι περισσότερο από πολύτιμη, όποτε και αν αυτό χρειάστηκε. Τέλος, θα ήθελα να ευχαριστήσω τον φίλο μου Πεππέ Σπύρο που μου επέτρεψε να αναδείξω τα προϊόντα του και ευελπιστώ σε μια συνεργασία στο άμεσο μέλλον. Θα ήθελα να ευχαριστήσω τον καθένα ξεχωριστά που έβαλε το λιθαράκι του στο να τελειώσω τις σπουδές μου και που μου έδωσε ώθηση να συνεχίσω σε κάθε δυσκολία.

ΠΕΡΙΛΗΨΗ

Στην παρούσα εργασία μελετήθηκαν η σχεδίαση Ιστοσελίδων και αναπτύχθηκε ένα ηλεκτρονικό κατάστημα με Σύστημα Διαχείρισης Περιεχομένου (WordPress). Αρχικά, πραγματοποιήθηκε μια ιστορική αναδρομή στο Διαδίκτυο, και μια ανάλυση της εξέλιξης όλων των τεχνολογιών που γνωρίζουμε μέχρι σήμερα ως Παγκόσμιο ιστό. Επίσης μελετήθηκε ο ρόλος του προγραμματιστή προσκηνίου εφαρμογών (Front End Developer), οι γλώσσες που χρησιμοποιεί ένας Front End σχεδιαστής Ιστοσελίδων, η ανατομία και η σύνταξη τους. Κατ' αντιστοιχία μελετήθηκε ο ρόλος του προγραμματιστή του παρασκηνίου εφαρμογών (Back End Developer) στον σχεδιασμό Ιστοσελίδων, οι γλώσσες και τα πρωτοκόλλα που χρησιμοποιεί ένας Back End σχεδιαστής, αλλά και οι ευθύνες που τον βαρύνουν. Παρουσιάστηκαν και αναλύθηκαν οι τεχνικές σχεδίασης UI / UX, ο ρόλος των σχεδιαστών και οι ευθύνες που φέρουν, καθώς αναλύθηκαν και οι αμοιβές των προγραμματιστών του διαδικτύου γενικότερα. Καταλήγοντας παρουσιάστηκαν τα κυριότερα CMS της αγοράς όπως το Drupal, το Joomla, καθώς και το WordPress όπου δημιουργήθηκε και το ηλεκτρονικό κατάστημα της εργασίας. Στο τελευταίο κεφάλαιο γίνεται ενδελεχής ανάλυση της δημιουργίας και του σχεδιασμού του ηλεκτρονικού καταστήματος.

Λέξεις-κλειδιά: Web Design, Front-End, CSS, HTML, Back-End, Python, Java, API, UI/UX, CMS, WordPress, Elementor, WooCommerce

ABSTRACT

In the present work the differences between the web designers were studied and an online store was developed with the WordPress Content Management System. A history review of the Internet and how everything we know today as the Web has started. Next, the role of the Front-End Developer was studied, along with the languages used by a Front-End Web Designer as well as their anatomy and syntax. Furthermore, the role of the Back End Developer in the design of the Websites was studied, along with the languages and protocols used by a Back End designer, as well as the responsibilities of a background designer. Next, UI / UX design techniques were presented and analyzed, the role of designers and the responsibilities they bear as well as the remuneration of internet developers in general. Concluding, the main CMS of the market were presented, such as Drupal, Joomla and WordPress, where the online store of the current study was created. The last chapter provides an extensive analysis of the creation and design of the online store.

Keywords: Web Design, Front-End, CSS, HTML, Back-End, Python, Java, API, UI / UX, CMS, WordPress, Elementor, WooCommerce

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Δήλωση μη λογοκλοπής.....	5
ΕΥΧΑΡΙΣΤΙΕΣ	6
ΠΕΡΙΛΗΨΗ.....	7
ABSTRACT.....	8
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	9
1. Ανάπτυξη Διαδικτύου.....	12
1.1 Ιστορική Αναδρομή του Web Design	13
1.2 Η σημαντικότητα του σχεδιασμού και ανάπτυξης Ιστοσελίδων	15
1.2.1 Ευκολότερη πλοήγηση.....	16
1.2.2 Μηχανές αναζήτησης (SEO).....	16
1.2.3 Οπτικό Περιεχόμενο.....	17
1.2.4 Επωνυμία.....	17
1.3 Σχεδιαστικά στοιχεία ενός Ιστού.....	17
2. Front-End Σχεδίαση.....	19
2.1 Ο ρόλος του Προγραμματιστή.....	20
2.2 Γλώσσα Front-End σχεδίασης HTML (Hypertext Markup Language)	20
2.2.1 Ανατομία της HTML.....	21
2.2.2 Ανατομία ενός αρχείου HTML.....	22
2.3 Γλώσσα Front-End σχεδίασης CSS (Cascading Style Sheets).....	23
2.3.1 Σύνταξη και ενότητες της CSS.....	23
2.4 Γλώσσα σχεδίασης JavaScript.....	25
2.4.1 Που χρησιμοποιείται η JavaScript.....	25
2.4.2 Πλεονεκτήματα της JavaScript.....	25
2.4.3 Ανατομία της Javascript.....	26

3. Back-End Σχεδίαση	28
3.1 Δεξιότητες για έναν Back -End Developer.....	28
3.2 Ο Ρόλος και οι ευθύνες ενός Back-End Σχεδιαστή.....	29
3.3 Γλώσσες Back-End σχεδίασης.....	30
3.3.1 Η Γλώσσα Java	30
3.3.2 Η Γλώσσα Python.....	31
3.3.3 Η Γλώσσα SQL.....	31
3.4 Τύποι APIs.....	32
3.4.1 Τύποι API πρωτοκόλλων	33
4. UI/UX Σχεδίαση και οι αμοιβές των Developers.....	35
4.1 Τι είναι η UI σχεδίαση.....	35
4.2 Τι είναι η UX σχεδίαση	36
4.3 Πως τα UI/UX συνεργάζονται	36
4.4 Ποιες οι ευθύνες του UI σχεδιαστή	37
4.5 Μέσες αμοιβές των Developer (UI/UX, Frontend, Backend, Full Stack).....	37
5. Σύστημα Διαχείρισης Περιεχομένου (CMS)	40
5.1 Πως δουλεύει ένα CMS	41
5.2 Γιατί να χρησιμοποιήσουμε CMS	42
5.3 Το Σύστημα Διαχείρισης Περιεχομένου Joomla	43
5.3.1 Γιατί να χρησιμοποιήσει κάποιος Joomla και ποιοι το έχουν επιλέξει	45
5.3.2 Joomla στατιστικά κατά το 2021	45
5.3.3 Joomla vs WordPress.....	46
5.4 Το Σύστημα Διαχείρισης Περιεχομένου Drupal.....	47
5.4.1 Λειτουργικότητα Drupal.....	48
5.4.2 Drupal vs WordPress	49
5.4.3 Πόσο ευκολόχρηστα και ασφαλή είναι τα δύο συστήματα	50
6. Ηλεκτρονικό Κατάστημα με WordPress	51

6.1 WordPress Dashboard και Back End σχεδίαση.....	55
6.2 Γενικές Ρυθμίσεις και προσθήκη θέματος.....	55
6.3 Δημιουργία Κατηγοριών Σελίδων και Menu.....	57
6.3.1 Δημιουργία σελίδων κατηγοριών και μενού.....	59
6.4 WooCommerce Tool.....	61
6.4.1 Φόροι μεταφορικά και πληρωμές.....	62
6.5 Simple και Variable Products.....	68
6.6 Elementor Theme Builder και WP Forms.....	71
6.6.1 Κεφαλίδα και Υποσέλιδο και Responsive Design.....	76
6.6.2 WP Forms.....	77
7. Συμπεράσματα.....	78
Βιβλιογραφία.....	79

Κεφάλαιο 1^ο

1. Ανάπτυξη Διαδικτύου

Η ανάπτυξη διαδικτύου είναι, κατά κύριο λόγο, η κατασκευή και η συντήρηση των Ιστοσελίδων. Υπάρχουν προγραμματιστές που εργάζονται για την ανάπτυξη Ιστού, χρησιμοποιώντας διάφορες γλώσσες, οι οποίες εξαρτώνται από το είδος της πλατφόρμας αλλά και των εργασιών στις οποίες δουλεύουν. Οι δεξιότητες ανάπτυξης ιστού, έχουν μεγάλη ζήτηση σε όλο τον κόσμο καθώς είναι σχετικά εύκολη η εκμάθηση και η υλοποίηση ενός έργου, ακόμη και χωρίς κάποια αναγνωρισμένη πιστοποίηση, ενώ η επιλογή αυτού του επαγγέλματος φαίνεται ιδανική καθώς οικονομικά το καθιστά αξιόλογο για βιοποριστικούς σκοπούς (marionletendart, 2018).



Εικόνα 1 - Web Development

Η διαδικασία ανάπτυξης ιστοσελίδων δεν περιλαμβάνει όμως μόνο την κατασκευή αλλά και την συντήρηση, τον προγραμματισμό της ταχύτητας, αλλά και την ασφάλεια που επιζητεί ο πελάτης. Επίσης, μπορεί να περιλαμβάνει τις ενέργειες που χρειάζονται για την κατάταξη στις μηχανές αναζήτησης (Search Engine Optimization). Επομένως, η ανάπτυξη

ιστοσελίδας κυμαίνεται από την δημιουργία σελίδων απλού κειμένου, έως εφαρμογές κοινωνικών δικτύων και ηλεκτρονικές εφαρμογές, τα γνωστά e-shop. Η ιεραρχία ανάπτυξης ιστοσελίδων έχει ως εξής:

1. Κωδικοποίηση από την πλευρά του εξυπηρετητή (Server)
2. Κωδικοποίηση από την πλευρά του χρήστη (Client)
3. Τεχνολογία των βάσεων δεδομένων (What Does Web Development Mean?, 2020)

1.1 Ιστορική Αναδρομή του Web Design

Ο πρώτος ιστότοπος στον χώρο του Internet δημιουργήθηκε από τον Tim-Berners Lee στο CERN ενώ ο ίδιος δημιούργησε και την περίφημη γλώσσα HTML με την οποία κωδικοποίησε και ιστότοπο. Στα πρώτα χρόνια ανάπτυξης Ιστοσελίδων ο μοναδικός τρόπος εκχώρησης πληροφοριών ήταν η συνάρτηση <table> της γλώσσας και δεν υπήρχαν τα χρώματα, οι εικόνες, τα γραφικά και τα 3D εφέ που γνωρίζουμε σήμερα. Ο παγκόσμιος Ιστός συνέχισε να εξελίσσεται και μερικά χρόνια αργότερα δημιουργήθηκε η πρώτη μηχανή αναζήτησης η ALIWEB. Η ALIWEB οργάνωνε τα link στις κατηγορίες υπολογιστών, ψυχαγωγίας, έρευνας και αγορών κ.α. Επίσης πρόσθεσε το χρώμα και διευκόλυνε έτσι τους χρήστες να βρίσκουν πιο εύκολα αυτό που αναζητούσαν.

Ο παγκόσμιος Ιστός μεγάλωνε διαρκώς, μαζί και η επιθυμία για την χρήση των σελίδων σε επιχειρησιακούς σκοπούς. Περίπου στο 1993 οι σελίδες ξεκίνησαν να προσελκύουν τους χρήστες με διάφορες προσκλήσεις όπως “Κλικ εδώ για είσοδο” ή δημοφιλέστερο σε όλους μας Sign Up. Το 1994 το περιοδικό Hotwire έφτιαξε το πρώτο Web banner ad. Οι Ιστοσελίδες εξερράγησαν σε δημοτικότητα την δεκαετία του 90. Υπήρχε μία μόνο σελίδα το 1991 ενώ υπήρχαν 2.738 το 1994, την χρονιά όπου ξεκίνησε την Amazon ο Τζεφ Μπέζος. (Little, 2021).

1995 – 2000

Η εξέλιξη του σχεδιασμού που συνέβη από το 1995 έως το 1998 θα άλλαζε σημαντικά την ιστορία του Web Design. Έγινε εισαγωγή της JavaScript, μιας γλώσσας προγραμματισμού που πρόσθεσε κίνηση στις στατικές σελίδες με διάφορα εφέ. Η γλώσσα HTML 2.0 που

κυκλοφόρησε το 1995, έλυνε τα χέρια των σχεδιαστών, υποστηρίζοντας διάφορα θεαματικά εφέ και πίνακες.

Το 1996 ένα μεγάλο κεφάλαιο στην ιστορία της σχεδίασης και ανάπτυξης Ιστοσελίδων - ένα από τα καλύτερα δίδυμα στην ιστορία ανάπτυξης Ιστοσελίδων- θα έρθει για να δώσει μεγαλύτερη κατεύθυνση και σχεδίαση των στοιχείων της HTML, η CSS. Η CSS είναι μια style γλώσσα, όπου τελικά θα αντικαταστήσει τους πίνακες της HTML, με κάποιες εξαιρέσεις που απαιτούσαν πραγματική διάταξη πίνακα. (Little, 2021)

2000 – 2006

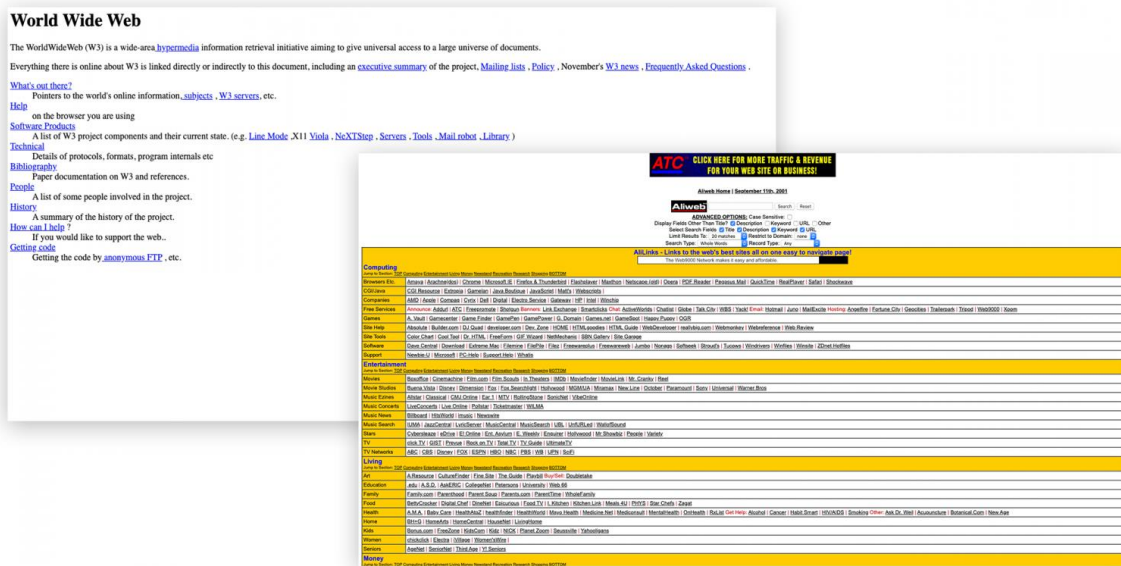
Μέχρι το 2000 όλα τα παραπάνω έχουν σχεδιαστεί για την εξυπηρέτηση σε περιηγητές επιτραπέζιων υπολογιστών (PCs). Στις αρχές του 2000 οι προγραμματιστές ξεκίνησαν να πειραματίζονται στην δημιουργία των σελίδων και σε άλλες συσκευές όπως κινητά και tablet. Όμως επειδή οι περιηγητές και οι συσκευές εξελισσόταν συνεχώς, οι ενημερώσεις έπρεπε να γίνονται ξεχωριστά, κάτι που καθιστά την διαδικασία χρονοβόρα. Όσο περνούσε ο καιρός η ανάγκη για συστήματα διαχείρισης περιεχομένου (CMS) μεγάλωνε, καθώς επέτρεπε την δυναμική σχεδίαση περιεχομένων. Το πιο δημοφιλές σύστημα διαχείρισης περιεχομένου στον κόσμο είναι το WordPress. Το WordPress καλύπτει το 35.9% του διαδικτύου, ας πούμε το 1/3 του παγκόσμιου Ιστού. (Little, 2021)

2007-2010

Η μεγαλύτερη τεχνολογική εξέλιξη εκείνη την χρονιά (2007) ήταν η παρουσίαση του πρώτου iPhone. Ήταν το πρώτο κινητό τηλέφωνο, που έφερε ένα πρωτότυπο λογισμικό σε κινητή συσκευή, όπως είχε πει ο Steve Jobs τότε στην παρουσίαση του. Το πρόγραμμα περιήγησης Safari πρόσθεσε, ήταν ο πρώτος πλήρης και λειτουργικός περιηγητής σε HTML σε κινητή συσκευή. (Little, 2021). Τον Δεκέμβριο του 2008 ήρθε σε κυκλοφορία το Google Chrome και στις αρχές του 2010 ξεπέρασε σε δημοτικότητα τον Internet Explorer και έφτασε στο peak του το 2018 όπου είχε περισσότερο από το 66% μεριδίου στην παγκόσμια αγορά (Web Design History Timeline, n.d.).

Με την πάροδο του χρόνου, στο Web Design είχαμε την εξέλιξη επίπεδης σχεδίασης (φωτεινά χρώματα, 2D). Οι δύο μεγάλοι κολοσσοί, Microsoft και Apple, ακολούθησαν αυτό το μοντέλο μέσω του Windows Media Center και iOS 7 αντίστοιχα. Ο καθένας μας μπορεί να παρατηρήσει αυτή την αλλαγή αν κοιτάξει την διαφορά των εικονιδίων του

iPhone 5 με το iPhone 7. (Little, 2021). Ο Jordan Walke ανακοίνωσε την δημιουργία βιβλιοθήκης JavaScript React, όπου έχει στόχο στην διευκόλυνση δημιουργίας διεπαφών χρήστη αλλά και την δημιουργία εφαρμογών κινητών συσκευών. Την ανάπτυξη και συντήρηση της React χειρίζεται μια ανεξάρτητη ομάδα προγραμματισμών του Facebook (Web Design History Timeline, χ.χ.).



Εικόνα 1 - Αριστερά το πρώτο website, δεξιά η πρώτη μηχανή αναζήτησης aliweb

1.2 Η σημαντικότητα του σχεδιασμού και ανάπτυξης Ιστοσελίδων

Το 2021 θα μείνει για πάντα ορόσημο στην Ιστορία, εξαιτίας αυτής της πρωτοφανής υγειονομικής κρίσης που βίωσε ο πλανήτης. Το γεγονός όμως αυτό για μερικούς επαγγελματίες ίσως να αποδειχθεί τελικά κερδοφόρο. Εξαιτίας λοιπόν των περιοριστικών μέτρων, μερικές επιχειρήσεις για να πουλήσουν τα προϊόντα τους (ρούχα, είδη υγιεινής, φαγητό) αναγκάστηκαν να δημιουργήσουν ηλεκτρονικά μαγαζιά (e-shop) ώστε να μπορέσουν να σταθούν στην αγορά.

Αυτά τα ηλεκτρονικά μαγαζιά ή απλά Ιστοσελίδες, δημιουργήθηκαν από ανάγκη για μερικούς, αλλά και λόγω του διαθέσιμου χρόνου που υπήρχε εξαιτίας αυτής της πανδημίας. Οι επιχειρήσεις αναγκάστηκαν να βρουν τρόπους να προσεγγίσουν και πάλι το κοινό τους, έπρεπε επομένως να σχεδιάσουν άριστα τις σελίδες τους στο Διαδίκτυο. Στην

συνέχεια θα μελετήσουμε μερικά σημεία που είναι απαραίτητα για έναν καλό σχεδιασμό Ιστοσελίδων. (Lentz, 2020)

Μια καλή σχεδίαση δίνει την πρώτη εντύπωση. Σύμφωνα με την Google ο χρόνος που χρειάζεται ένας επισκέπτης για να αξιολογήσει μια Ιστοσελίδα είναι λιγότερος από 0,05 δευτερόλεπτα για αυτό μια κακή σχεδίαση μπορεί να απομακρύνει τους πελάτες, με το ποσοστό που τις απορρίπτει να ανέρχεται στο 94%. Τα καλά σχέδια πρέπει να είναι ορθά οργανωμένα, φωτεινά, όμορφα, να μην έχει μεγάλα μπλοκ κώδικα και να υπάρχει σωστή τοποθέτηση των λιστών, στην back-end σχεδίαση της Ιστοσελίδας. (Lentz, 2020)

Με έναν όμορφα σχεδιασμένο Ιστότοπο οι πελάτες προσελκύνονται ευκολά και γρήγορα σε αντίθεση αν είχαν έναν άσχημα σχεδιασμένο. Από εταιρείες που βελτιστοποίησαν την Ιστοσελίδα τους για κινητά τηλέφωνα παρατηρήθηκε αύξηση 62% σε πωλήσεις, ενώ 64% παρατηρήθηκε στα tablet αντίστοιχα. (ibis, n.d.)

1.2.1 Ευκολότερη πλοήγηση

Μια σελίδα διαδικτύου θα πρέπει να έχει εύκολα προσβάσιμες πληροφορίες, γρήγορη ταχύτητα στην φόρτωση και πληθώρα επιλογών, όπως πλαίσια αναζήτησης και φίλτρα, όπου ο επισκέπτης μπορεί εύκολα να βρει αυτό που αναζητεί. Ο προγραμματιστής του Ιστοτόπου, είναι υποχρεωμένος να δοκιμάζει τακτικά την σελίδα και να επιβλέπει τυχών σφάλματα για να επιτυγχάνεται γρήγορα η επίλυση τους και η εύκολη πλοήγηση στους επισκέπτες. (ALfred, n.d.)

1.2.2 Μηχανές αναζήτησης (SEO)

Η βελτιστοποίηση των SEO είναι μια θεμελιώδης πλευρά στην ανάπτυξη του διαδικτύου. Με βάση την πληθώρα Ιστοσελίδων που υπάρχει στο Internet, οι προγραμματιστές των μηχανών αναζήτησης έπρεπε να βρουν έναν τρόπο καταγραφής. Σε αυτόν τον τομέα σημαντικό ρόλο έχει το πως είναι σχεδιασμένη μια σελίδα. Εξετάζεται αν υπάρχουν λέξεις κλειδιά, η σύνδεση των link μεταξύ τρίτων, η ποιότητα της εικόνας. Αυτό σημαίνει ότι ένας Ιστότοπος έχει όλα τα κριτήρια που θα τον εντάξει σε υψηλά σημεία κατάταξης των αναζητήσεων. Ένας προγραμματιστής Ιστού αν έχει δημιουργήσει μια σελίδα με πολλά ερωτήματα (όπως, ανέφερε τυχών προβλήματα) την κάνει πιο ελκυστική, αποκτά περισσότερα clicks και έτσι βρίσκεται ψηλά στις μηχανές αναζήτησης. (ALfred, n.d.)

1.2.3 Οπτικό Περιεχόμενο

Μια Ιστοσελίδα είναι απαραίτητο να περιέχει οπτικοακουστικό υλικό και αυτό γιατί οι πελάτες κατανοούν καλύτερα τα προϊόντα που προσφέρονται. Παλιότερα πριν την εισαγωγή των γλωσσών styling (CSS) οι Ιστοσελίδες περιείχαν μόνο κείμενα και η πώληση των προϊόντων και των υπηρεσιών ήταν δύσκολή, ενώ πλέον ο πελάτης έχει σαφή εικόνα αυτών που αναζητεί καθώς μπορεί να τα δει ή να τα ακούσει. Ένα κακό σε όλο αυτό είναι ότι η κατάχρηση του οπτικού περιεχομένου έχει ως αποτέλεσμα ακριβώς το αντίθετο, να μην μπορεί δηλαδή ο χρήστης να κατανοήσει εύκολα το περιεχόμενο της Ιστοσελίδας με αποτέλεσμα αυτή να πέφτει στην κατάταξη στις μηχανές αναζήτησης (ALfred, n.d.).

1.2.4 Επωνυμία

Όταν μια Ιστοσελίδα ανεβαίνει στο διαδίκτυο και είναι ενεργή για τον χρήστη αυτόματα η επιχείρηση-οργανισμός γίνεται ευρέως γνωστός-η. Οι προγραμματιστές δημιουργούν μια επωνυμία που θα εμφανίζεται σε όλες τις μηχανές αναζήτησης (Google, Bing, Yahoo) όπως ίδιο είναι και το λογότυπο. Σε περίπτωση που αλλάξει η επωνυμία οι προγραμματιστές είναι υποχρεωμένοι να κάνουν την ενημέρωση στις SEO όπου και θα γίνει η ενημέρωση που απαιτείται. Ο Ιστότοπος μπορεί να ενημερώσει τους πελάτες του, μέσω των social media, για κάποια επικείμενη αλλαγή πριν αυτή πραγματοποιηθεί (ALfred, χ.χ.).

1.3 Σχεδιαστικά στοιχεία ενός Ιστού

Τώρα που έχουν ήδη αναφερθεί μερικοί από τους λόγους, για τους οποίους είναι σημαντικός ο σχεδιασμός μια Ιστοσελίδας, θα εξετάσουμε μερικά επιπλέον στοιχεία που κάνουν έναν σχεδιασμό καλύτερο. Παρακάτω θα δούμε μια λίστα από ενέργειες που βοηθάνε σε μια αξιόλογη Ιστοσελίδα.

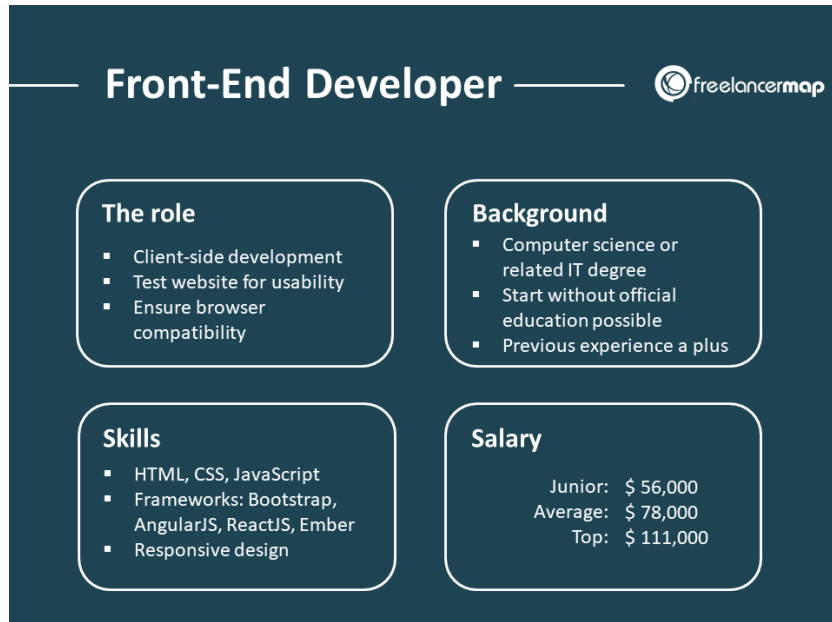
- Κουμπιά Call – To – Action, η αλλιώς κουμπιά παρότρυνσης για δράση που τοποθετούνται κατάλληλα στον Ιστού όπου ο χρήστης να είναι έτοιμος να αναλάβει δράση.

- Καθιέρωση κανόνα 60-30-10 των χρωμάτων όπου το 60% είναι το κυρίαρχο χρώμα, το 30% το δευτερεύον χρώμα και το 10% το χρώμα έμφασης. (DECORation)
- Στυλ γραμματοσειράς, μια συγκεκριμένη επιλογή στυλ και γραμματοσειράς μπορεί να διατηρήσει την συνέπεια σε μια σελίδα.
- Εικόνες υψηλής ποιότητας που μπορούν να αναδείξουν την ομορφιά της σελίδας. (Lentz, 2020)

Κεφάλαιο 2^ο

2. Front-End Σχεδίαση

Το «front-end» είναι το «προσκήνιο» μέρος της Ιστοσελίδας. Περιλαμβάνει την εμφάνιση με όλα όσα αλληλοεπιδρούν οι επισκέπτες (κουμπιά, φόρμες, πίνακες, εικόνες). Το προσκήνιο συνδέσει το γραφικό περιβάλλον με τις ενέργειες που γίνονται πίσω από την Ιστοσελίδα (στο παρασκήνιο - back-end). Εάν έχουμε έναν υπολογιστή όλα τα μέρη που μπορούμε να δούμε απευθείας όπως πληκτρολόγιο, ποντίκι μπορούμε να πούμε ότι αποτελούν το προσκήνιο, ενώ το hardware, μνήμες RAM, σκληρός δίσκος αποτελούν το υλικό. Με αυτόν τον τρόπο λειτουργεί μια σελίδα Ιστού και σε αυτό το κεφάλαιο θα περιγράψει η σχεδίαση του προσκηνίου και οι γλώσσες με την οποία επιτυγχάνεται αυτή (Marinov, 2020).



Εικόνα 2 - Front End Σχεδίαση

2.1 Ο ρόλος του Προγραμματιστή

Τώρα που γνωρίζουμε τον ρόλο του Προγραμματιστή στην σχεδίαση front-end θα μελετηθούν οι γλώσσες προγραμματισμού που χρησιμοποιεί. Οι γλώσσες προγραμματισμού είναι συνήθως οι HTML, CSS, JavaScript. Δουλειά όμως ενός προγραμματιστή σχεδίασης είναι όχι μόνο να χτίσει την σελίδα στο εξωτερικό της αλλά και να παρακολουθεί πως αλληλοεπιδρά ο χρήστης με αυτήν. Οι υποχρεώσεις ενός προγραμματιστή σχεδίασης είναι:

1. Ανάπτυξη με HTML,CSS,JS
2. Συντήρηση και βελτιστοποίηση διεπαφών χρήστη (user interface)
3. Δημιουργία εργαλείων για την λειτουργικότητα της σελίδας σε άλλες συσκευές ή άλλους περιηγητές.
4. Διαχείριση ροής λογισμικού
5. Εφαρμογή onpage SEO
6. Μετατροπή UX σε πραγματικότητα σχεδίων UI/ UX
7. Βελτίωση στην ταχύτητα φόρτωσης της σελίδας
8. Εξασφάλιση διαδικτυακής προσβασιμότητας (Marinon, 2020).

2.2 Γλώσσα Front-End σχεδίασης HTML (Hypertext Markup Language)

Η Γλώσσα Σήμανσης Κειμένου είναι η πρώτη γλώσσα που θα μάθει κάποιος για τη σύνταξη της δομής κειμένου και των στοιχείων. (Marinon, 2020)

Το κείμενο θα μπορούσε να δομηθεί σε σύνολο παραγράφων, σε αρίθμηση, σε εικόνες και πίνακες. Η HTML δηλαδή, χρησιμοποιείται για να κάνει το κείμενο να φαίνεται με έναν τρόπο συγκεκριμένο.

Ένα παράδειγμα που μπορεί να δοθεί είναι το παρακάτω:

Το όνομα μου είναι Βαγγέλης

Σε γλώσσα HTML και για να επισημανθεί ότι αυτή η πρόταση θα μπει σε παράγραφο πρέπει να περιληφθεί μέσα στο tag <p> και θα παρουσιαζόταν όπως παρακάτω:

<p> Το όνομα μου είναι Βαγγέλης </p>.

Το <p> ονομάζεται ετικέτα ανοίγματος, όπου σημαίνει ότι αρχίζει και τίθεται σε ισχύ το στοιχείο (p).

Το </p> ονομάζεται ετικέτα κλεισίματος, είναι ίδιο ακριβώς με την ετικέτα ανοίγματος μόνο που πριν το στοιχείο p μπαίνει και η κάθετος / και στην προκειμένη περίπτωση δηλώνει ότι τελειώνει η παράγραφος.

Στοιχείο ονομάζεται η ετικέτα ανοίγματος, η ετικέτα κλεισίματος και το περιεχόμενο. (MDN Web Docs, 2021)

2.2.1 Ανατομία της HTML

Όπως αναφέρθηκε και στο παραπάνω παράδειγμα αλλά και όπως φαίνεται στην φωτογραφία η ανατομία της γλώσσας HTML είναι η εξής:

```
<body>
  <h1>Hello World</h1>
  <p>Αυτό είναι ένα <i>κείμενο</i> </p>
  <hr>
  Αυτό είναι <b>ένα άλλο</b> κείμενο.
  <p>
```

1. Ετικέτα ανοίγματος (p)
2. Ετικέτα κλεισίματος (/p)
3. Περιεχόμενο του στοιχείου
4. Το στοιχείο

Στην εικόνα παραπάνω παρουσιάζεται ένα ακόμη tag το <i> όπου αυτό τοποθετείται αν θέλουμε να προσθέσουμε στοιχεία μέσα σε άλλα στοιχεία. Στο συγκεκριμένο παράδειγμα το στοιχείο *κείμενο* θα εμφανιστεί με πλάγια γράμματα (MDN Web Docs, 2021).

2.2.2 Ανατομία ενός αρχείου HTML

- `<!DOCTYPE html>`. Όταν πρωτοξεκίνησε η γλώσσα τα doctypes χρησιμοποιήθηκαν για να λειτουργούν σαν σύνδεσμοι σε ένα σύνολο κανόνων που θα μπορούσε να σημαίνει και αυτόματο έλεγχο σφαλμάτων για να είναι σωστή η σελίδα HTML. Σήμερα απλά το χρησιμοποιούμε για να βεβαιωθούμε ότι το αρχείο μας συμπεριφέρεται σωστά. Αφού λοιπόν γράφουμε ένα δομημένο κείμενο, ξεκινάμε πάντα με το tag αυτό για να γίνουν όλες οι απαραίτητες ενημερώσεις!
- `<html></html>`. Ότι γράφεται ανήκει μέσα στο tag αυτό, μπορούμε να το ονομάσουμε και ριζικό tag.
- `<head></head>`. Περιλαμβάνει κάποιες πληροφορίες που αποδίδονται στον φυλλομετρητή μας.
- `<meta charset = "UTF-8">`. Είναι ένα από τα στοιχεία που γράφεται μέσα στο `<head>` και χρησιμοποιείται για να ορίσει το σύνολο των χαρακτήρων που χρησιμοποιεί το έγγραφο.
- `<title></title>`. Είναι ο τίτλος της σελίδας όπως θα εμφανίζεται στον φυλλομετρητή αλλά και όπως όταν αυτός θα ορίζεται σελιδοδείκτης.
- `<body></body>`. Εδώ περιέχεται όλο το περιεχόμενο που θέλουμε να εμφανίζεται. Αυτό είναι από εικόνες μέχρι βίντεο και διάφορα παιχνίδια. Είναι το κυρίως πιάτο της γλώσσας.

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the page </title>
<meta charset="UTF-8">
</head>
<body>
  <h1>Hello World</h1>
  <p>Αυτό είναι ένα <i>κείμενο</i> </p>
  <hr>
  Αυτό είναι <b>ένα άλλο</b> κείμενο.
  <p>
  <a href="https://www.google.gr/">Google</a>
  <br>
  </p>
</body>
</html>
```

Στην παραπάνω εικόνα έχουμε και ένα άλλο tag, το `<a href...>`, το οποίο σημαίνει αναφορά υπερ-κειμένου και χρησιμοποιείται για τους συνδέσμους «links» που θέλουμε να προσθέσουμε στον Ιστό μας. Στο συγκεκριμένο παράδειγμα πατώντας πάνω στην λέξη Google ανοίγει η αρχική σελίδα της Google. Μεγάλη προσοχή απαιτείται μετά το tag href διότι δεν πρέπει να παραλειφθεί ποτέ το πρωτόκολλο `https://`, καθώς θα εμφανίζονται μνήματα σφάλματος ή δεν θα λειτουργεί καθόλου ο σύνδεσμος (MDN Web Docs, 2021).

2.3 Γλώσσα Front-End σχεδίασης CSS (Cascading Style Sheets)

Η γλώσσα CSS χρησιμοποιείται για τη διαμόρφωση των στοιχείων σε γλώσσες σήμανσης όπως είναι η HTML. Όπως είπαμε και στο πρώτο κεφάλαιο η γλώσσα CSS αναπτύχθηκε το 1966 για τον απλούστατο λόγο ότι δίνει όλη την έμφαση (γραμματοσειρές, φόντο) στον Ιστότοπο. Η HTML δεν μπορούσε να το κάνει αυτό καθώς κάθε Ιστότοπος είχε διαφορετικό στυλ και έπρεπε να ξανά γραφτεί ο κώδικας από την αρχή, έτσι η ένταξη της CSS στον προγραμματισμό η ζωή των webmasters έγινε πιο εύκολη. Δεν είναι απαραίτητο ένας Ιστότοπος να έχει κώδικα CSS, μπορεί να έχει μόνο HTML αλλά θα είναι εντελώς γυμνός. (B, 2021)

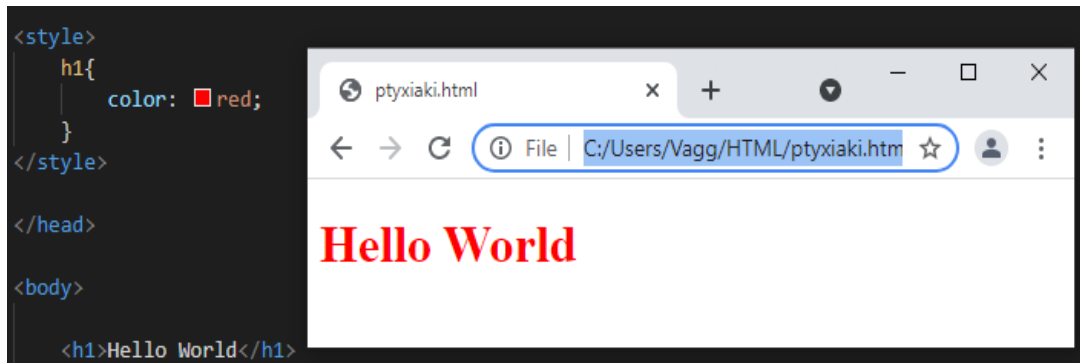
2.3.1 Σύνταξη και ενότητες της CSS

Η γλώσσα CSS βασίζεται σε κανόνες που καθορίζουν στοιχεία στην Ιστοσελίδα μας. Η CSS δεν είναι δύσκολο να γραφτεί καθώς δεν είναι ακριβώς μια γλώσσα προγραμματισμού, δεν περιέχει «if statements» και χρησιμοποιείται για styling. Αν θέλουμε για παράδειγμα να δώσουμε χρώμα στο κείμενο μας, «Hello world», που φαίνεται στις εικόνες παραπάνω αυτό μπορεί να επιτευχθεί με τρεις τρόπους.

Ο πρώτος τρόπος είναι να το γραφτεί inline. Δηλαδή μέσα στην HTML και μετά το tag `<h1>` να κωδικοποιηθεί ως εξής: `<h1 style="color = red">Hello World</h1>`.

Αυτός ο τρόπος δεν χρησιμοποιείται συχνά, διότι όταν θα χρειαστεί να μεγαλώσει ο κώδικας, αν επιθυμεί ο προγραμματιστής να αλλάξει για παράδειγμα το χρώμα πρέπει να τροποποιήσει εκείνο το κομμάτι του καμμάτι κώδικα, γεγονός που είναι χρονοβόρο. Επομένως, εδώ αναφέρεται για διδακτικούς σκοπούς.

Ένας άλλος τρόπος είναι να το τοποθετηθεί μέσα στο tag `<style>`. Δεν ανακατεύει την μορφοποίηση και την παρουσίαση που γίνεται αν γραφτεί όλο σε μια γραμμή. Με το tag `style` θα γραφόταν:



The image shows a code editor on the left with the following HTML code:

```
<style>
  h1{
    color: red;
  }
</style>

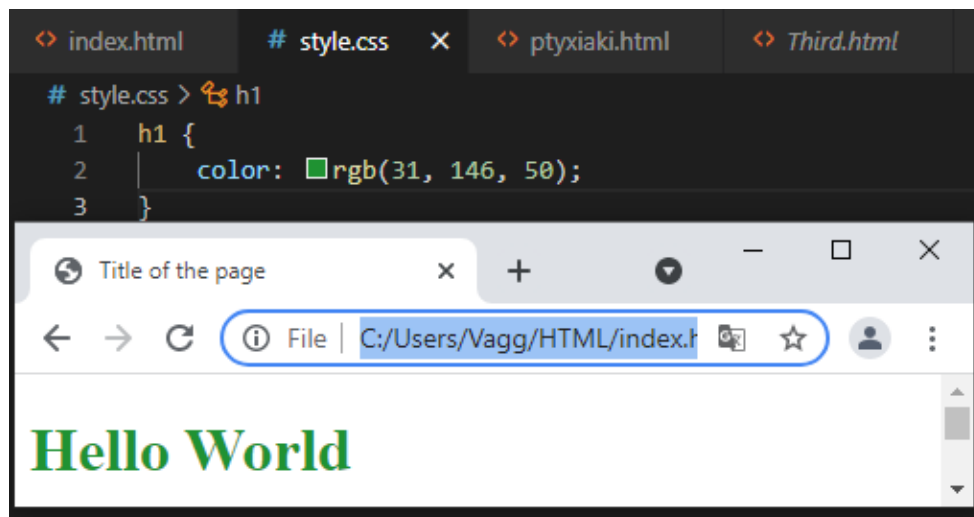
</head>

<body>

  <h1>Hello World</h1>
```

On the right, a browser window displays the result: "Hello World" in red text. The browser's address bar shows the file path: `C:/Users/Vagg/HTML/ptyxiaki.htm`.

και θα επιτυγχάναμε το αποτέλεσμα που φαίνεται. Βέβαια ούτε αυτός ο τρόπος είναι ο βέλτιστος, διότι ξανά σε 300 γραμμές κώδικα και 10 αρχεία, αν επιθυμήσει ο προγραμματιστής να το τροποποιήσει πρέπει να τα ανοίξει όλα τα αρχεία και να ψάξει που είναι το καθένα.



The image shows a code editor with two files open: `style.css` and `ptyxiaki.html`. The `style.css` file contains:

```
# style.css > h1
1  h1 {
2  |   color: rgb(31, 146, 50);
3  }
```

The `ptyxiaki.html` file is not visible in the snippet. Below the code editor, a browser window displays the result: "Hello World" in green text. The browser's address bar shows the file path: `C:/Users/Vagg/HTML/index.f`.

Ο καλύτερος τρόπος για να δοθεί έμφαση και ζωή στην σελίδα επιτυγχάνεται αν αυτές οι λίγες γραμμές κώδικα γραφτούν σε ξεχωριστό αρχείο με την κατάληξη `.css`. Συνηθίζεται το όνομα αυτού του αρχείου να είναι το `style.css`.

Δημιουργείται το αρχείο `style.css` και εκεί μετά γράφουμε τον κώδικα χωρίς τα `style` tags. Επιπρόσθετα, πρέπει να αφαιρεθεί από τον κώδικα στο `html` αρχείο τα `style` tags και στην θέση τους να εισαχθεί η εντολή `<link rel="stylesheet" href="style.css">`. Έτσι το αρχείο θα καταλάβει με αυτή την εντολή ότι το `style` μας έχει μεταφερθεί πλέον στο ξεχωριστό

style.css αρχείο. Με τον συγκεκριμένο τρόπο και κάτω από την τελευταία αγκύλη δύναται να συνεχιστεί ο κώδικας με άλλα tags όπως το body και να οριστεί χρώμα στο background.

2.4 Γλώσσα σχεδίασης JavaScript

Η γλώσσα JavaScript είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που χρησιμοποιείται για να κάνει τις Ιστοσελίδες πιο δυναμικές. Δύναται να χρησιμοποιηθεί και από τον server και από τον client ενώ η γλώσσα προσφέρει δυνατότητες που η HTML και η CSS δεν τις έχουν. Όσες Ιστοσελίδες είναι προγραμματισμένες με JavaScript έχουν δυναμική εικόνα και καλύτερη σύνδεση με τους χρήστες. (Still, 2021)

Εν ολίγης ένας προγραμματιστής Ιστού χρησιμοποιεί HTML για την βασική δομή της σελίδας, την CSS για την μορφοποίηση και την διάταξη, την ωραιοποίηση με τις εικόνες ενώ η JavaScript βελτιώνει αυτά τα στοιχεία δίνοντας τους κίνηση.

2.4.1 Που χρησιμοποιείται η JavaScript

1. Μετατρέπει πιο δραστικά τις Ιστοσελίδες βελτιώνοντας την εμφάνιση των εικόνων, δημιουργώντας menu, την κίνηση των εικόνων και τις αλλαγές στα χρώματα των κουμπιών όταν τα επιλέγει ο χρήστης.
2. Οι προγραμματιστές χρησιμοποιούν πλαίσια της JavaScript όπως React όταν φτιάχνονται οι εφαρμογές των κινητών. Οι δυνατότητες αυτές περιλαμβάνουν διάφορα κουμπιά, όπως αναζήτησης και άλλα.
3. Είναι η κατάλληλη γλώσσα για εφαρμογές παιχνιδιών. Συνήθως χρησιμοποιείται η Node.js μια βιβλιοθήκη τη γλώσσας για τη δημιουργία βασικών web server. (Still, 2021)

2.4.2 Πλεονεκτήματα της JavaScript

- Αλληλεπίδραση: Η JavaScript δημιουργεί καλύτερες διεπαφές χρήστη με αποτέλεσμα να γίνεται καλύτερα η αλληλεπίδραση μεταξύ των χρηστών.
- Ταχύτητα: Η JavaScript εκτελείται απευθείας μέσα στον περιηγητή, μειώνοντας την ζήτηση, επομένως η ταχύτητα φόρτωσης είναι μεγαλύτερη.
- Ευκολία: Ένας προγραμματιστής JavaScript δεν είναι απαραίτητο να διαθέσει άπλετο χρόνο για την εκμάθηση της καθώς είναι σχετικά εύκολη και ευκολόχρηστη.

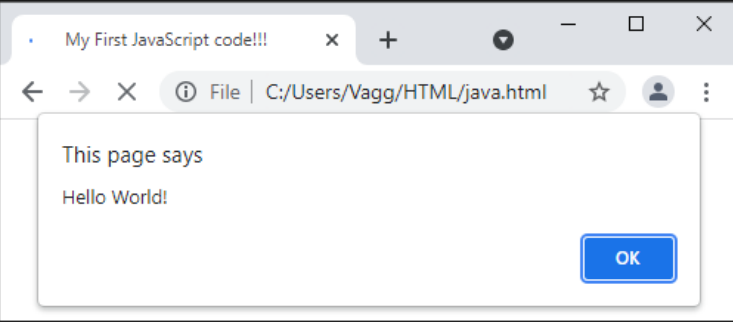
- Ενσωμάτωση: Η JavaScript λειτουργεί άριστα με τις υπόλοιπες γλώσσες καθώς μπορεί να ενσωματωθεί εύκολα μέσα σε διαφορετικές εφαρμογές.
- Δημοτικότητα: Είναι μια από τις πιο δημοφιλείς γλώσσες καθώς οι σελίδες στο σύνολο τους χρησιμοποιούν κάποιο κομμάτι της.

Δεν υπάρχουν ουσιαστικά μειονεκτήματα της γλώσσας, αλλά ένα αρκετά βασικό είναι η ενδεχόμενη παραβίαση ασφαλείας καθώς η JavaScript είναι ορατή προς όλους. Δεν υπάρχει όμως κάποιος σοβαρός λόγος για να μην ασχοληθεί κάποιος με τη συγκεκριμένη γλώσσα, αφού είναι η πιο δημοφιλής και είναι εύκολη για έναν νέο Developer που θέλει να κάνει τα πρώτα του βήματα στον προγραμματισμό του Διαδικτύου. (Still, 2021)

2.4.3 Ανατομία της Javascript

Αυτό που βοηθά το πρόγραμμα περιήγησης να ξεχωρίσει τον κώδικα JavaScript από όλο το υπόλοιπο πρόγραμμα είναι το tag `<script>`. Το πιο απλό πρόγραμμα που θα φτιάξει κάποιος στην JS θα έχει την μορφή της παρακάτω εικόνας:

```
1 <html>
2 <head>
3   <title>My First JavaScript code!!!</title>
4   <script>
5     alert("Hello World!");
6   </script>
7 </head>
8 <body>
9 </body>
10</html>
```

A screenshot of a web browser window. The browser's address bar shows the file path 'C:/Users/Vagg/HTML/java.html'. The page title is 'My First JavaScript code!!!'. An alert dialog box is displayed in the center of the browser, with the text 'This page says Hello World!' and an 'OK' button.

Όπως μπορούμε να διακρίνουμε η λέξη που εμφανίζει ο φυλλομετρητής μας είναι γραμμένη μέσα στα tag `script`! Με παρόμοιο τρόπο που κάναμε και στην CSS λειτουργεί και η JavaScript, δημιουργώντας δηλαδή ένα αρχείο ξεχωριστό με την κατάληξη .js και γράφουμε την εντολή της γραμμής 5 εκεί μέσα. Εκτελούμε μέσα στο σώμα (body) της HTML την εντολή `<script src="main.js"></script>` και όταν εκτελεστεί στο Visual Studio το πρόγραμμα θα έχει εμφανίσει το ίδιο αποτέλεσμα, με λιγότερο χρόνο απόκρισης και λιγότερο κώδικα σε ένα μοναδικό αρχείο. (Hartman, 2021)

Εάν θέλουμε να προσθέσουμε σχόλια στο πρόγραμμα μας στην JavaScript περικλείουμε το σχόλιο με κάθετο και αστερίσκο στην αρχή, ενώ στο τέλος των σχολίων τοποθετούμε αντίστροφα αστερίσκο και κάθετο, με την σειρά που τα αναφέραμε:

/ διάφορα σχόλια που θέλουμε να κάνουμε...*/*

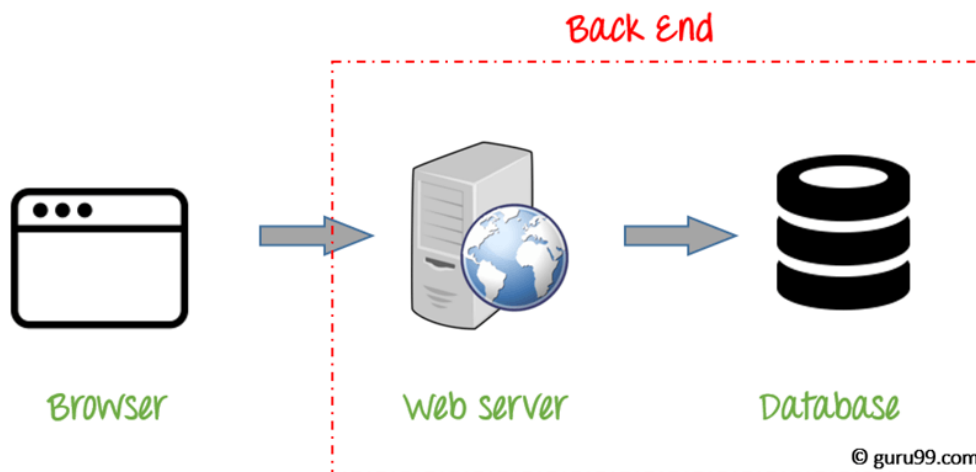
Η JavaScript δεν είναι όπως η HTML και η CSS, είναι μια δυναμική γλώσσα προγραμματισμού και δεν χρησιμοποιείται μόνο για την ανάπτυξη Ιστοσελίδων. Έχει συναρτήσεις και αντικείμενα όπως η Java η C και η C++.

Με λίγα λόγια όταν ένας χρήστης πατήσει ένα κουμπί της HTML εκτελείται στο παρασκήνιο ένας κώδικας - script της γλώσσας και ο χρήστης λαμβάνει αυτό το script ως αποτέλεσμα.

Κεφάλαιο 3^ο

3. Σχεδίαση Back-End

Η Back - End είναι η σχεδίαση από την πλευρά του διακομιστή, ενώ επικεντρώνεται στις βάσεις δεδομένων, στο scripting και στις εφαρμογές που εκτελούνται στο παρασκήνιο ενός Ιστοτόπου. Οι κώδικες και οι γλώσσες που χρησιμοποιούν οι back-end προγραμματιστές βοηθάνε τους φυλλομετρητές να αλληλοεπιδρούν με διάφορες βάσεις δεδομένων. (Martin, 2021). Όπως αναφέρθηκε και στο πρώτο κεφάλαιο οτιδήποτε αφορά σε γραμματοσειρές και σχέδια είναι αρμοδιότητα της front-end σχεδίασης, ενώ ότι έχει σχέση με το προσκήνιο, δηλαδή το περιεχόμενο μιας σελίδας αφορά τον διακομιστή που επικοινωνεί με βάσεις δεδομένων.



Εικόνα 3 - Back End Σχεδίαση

3.1 Δεξιότητες για έναν Back-End Developer

Οι δεξιότητες που πρέπει να έχει ένας προγραμματιστής είναι η καλή γνώση των γλωσσών, των βάσεων δεδομένων, των διακομιστών και του API

1. Ένας προγραμματιστής Ιστού back-end σχεδίασης πρέπει να έχει τεχνογνωσία γλωσσών όπως PHP, Java, Python. (Martin, 2021)
2. Ένας Developer πρέπει να κατέχει την γνώση για δημιουργία και επεξεργασία της βάσης δεδομένων, κυρίως πρέπει να ξέρει MySQL, SQLServer. (Martin, 2021)
3. Κάθε εφαρμογή που εκτελείται σε έναν υπολογιστή εκτελείται σε έναν διακομιστή (server) ο οποίος λαμβάνει αρχεία html, css, Javascript κλπ. Είναι συνεπώς απαραίτητο στοιχείο και στην ανάπτυξη Ιστοσελίδων. Μερικά παραδείγματα από αυτούς τους διακομιστές είναι οι Apache, Nginx. (InterviewBit, 2021)
4. Το Application Programming Interface ή διεπαφή των προγραμματιστικών διαδικασιών είναι ένα μέσο όπου τα λογισμικά υπολογιστών μπορούν και συνομιλούν μεταξύ τους. Όσο συμβαίνει αυτή η επικοινωνία ο server παρέχει το API και ο πελάτης το χρησιμοποιεί. Τα APIs λαμβάνουν μια εντολή από τον χρήστη, τον στέλνουν στον server και στη συνέχεια ο server στέλνει μια απάντηση στην εντολή. (InterviewBit, 2021)

Σε αυτό το σημείο πρέπει να τονιστεί ότι η επικοινωνία αυτή γίνεται μέσω των HTTP μεθόδων και οι προγραμματιστές πρέπει να έχουν αρκετά καλή γνώση σχετικά με τα APIs. Οι γλώσσες που χρησιμοποιούνται για τη δημιουργία είναι η Python ή η βιβλιοθήκη NodeJS ενώ στην συνέχεια οι Front-End developers χρησιμοποιούν αυτά τα APIs για να βελτιώσουν την εμπειρία χρήστη. (InterviewBit, 2021)

3.2 Ο Ρόλος και οι ευθύνες ενός Back-End Σχεδιαστή

Ο ρόλος όμως και οι ευθύνες που έχει ένα προγραμματιστής Ιστοσελίδων δεν είναι μόνο αυτά που προ αναφέρθηκαν, υπάρχουν μερικές ακόμη αρμοδιότητες:

- Συνεργάζεται με τους Front-End Developers για την ανάπτυξη αλγορίθμων από την πλευρά του διακομιστή για την μεταφορά δεδομένων στις web εφαρμογές.
- Συνεργάζεται με μηχανικούς για να αναπτυχθεί καλύτερα η εμπειρία χρήστη.
- Ελέγχει αν η εφαρμογή – σελίδα είναι το ίδιο γρήγορα κάθε φορά που αλλάζει ο αριθμός των επισκέψεων.

- Πειράζει και ελέγχει συχνά την βάση δεδομένων για την καλύτερη δυνατή αποθήκευση.
- Δημιουργεί και χειρίζεται τα APIs
- Εργάζεται στην ασφάλεια και στις καλύτερες πρακτικές της ενώ συντάσσει κώδικα και βιβλιοθήκες για μελλοντική χρήση. (InterviewBit, 2021)

3.3 Γλώσσες Back-End σχεδίασης

Με βάση όσα έχουν περιγραφεί μέχρι τώρα ο προγραμματισμός μιας Back-End σχεδίασης μπορεί να είναι αντικειμενοστραφής. Η αντικειμενοστραφής τεχνική εστιάζει στην δημιουργία αντικειμένων και οι γλώσσες που το κάνουν αυτό είναι οι Java, Python και άλλες. Η τεχνική της λειτουργικότητας σημαίνει ότι οι γλώσσες έχουν τη δυνατότητα να εκτελούν τις εντολές σε οποιαδήποτε σειρά. Μια δημοφιλής λειτουργική γλώσσα που χρησιμοποιείται και στις βάσεις δεδομένων είναι η SQL. (krystal, 2021)

3.3.1 Η Γλώσσα Java

Η Java είναι, αν όχι η πιο δημοφιλής, μία από τις πιο δημοφιλής γλώσσες προγραμματισμού στον κόσμο και αυτό γιατί μπορεί να την χρησιμοποιήσει κάποιος προγραμματιστής για οποιαδήποτε εφαρμογή. Το σημαντικότερο εργαλείο που κάνει την Java πιο εύλικτη από άλλες γλώσσες, εξίσου καλές, είναι το Java Virtual Machine. Η Java Virtual Machine μπορεί να εκτελεί τον κώδικα σε κάθε υπολογιστή, ανεξάρτητα αν αυτός έχει μεταγλωττιστεί αλλού.

Η Java αν και είναι τόσο διαδεδομένη σαν γλώσσα και σχετικά εύκολη θα έλεγε κάποιος που έχει εργαστεί με αυτήν, είναι λιγότερο δημοφιλής σε επίπεδα μάθησης από την Python. Αυτό συμβαίνει γιατί απαιτεί περισσότερο κώδικα και γιατί η Python είναι πιο κοντά στην ανθρώπινη λογική, επομένως είναι πιο εύκολο για κάποιον που ξεκινάει τώρα να ασχολείται με τον προγραμματισμό να μάθει γλώσσα Python. (krystal, 2021)

Πράγματα που μπορείς να κάνεις με την γλώσσα Java:

- 1 Ανάπτυξη κινητών εφαρμογών
- 2 Ανάπτυξη Ιστοσελίδας
- 3 Επεξεργασία εικόνων και GUI (Graphical User Interface)
- 4 Συνδεσιμότητα και επεξεργασία των βάσεων δεδομένων

Εταιρείες που χρησιμοποιούν την Java:

- 1 Airbnb
- 2 Uber
- 3 LinkedIn
- 4 eBay (krystal, 2021)

3.3.2 Η Γλώσσα Python

Ίσως η ταχύτερα αναπτυσσόμενη γλώσσα προγραμματισμού που χρησιμοποιείται για την ανάπτυξη Ιστοσελίδων και όχι μόνο, μπορεί εύκολα να την μάθει κανείς καθώς υπάρχουν πολλοί οδηγοί στον διαδίκτυο. Η σύνταξη της είναι απλή και κατανοητή, ενώ το μικρό μήκος του κώδικα της κάνει παραγωγική στην Back-End σχεδίαση. (krystal, 2021)

Πράγματα που μπορείς να κάνεις με την γλώσσα Python:

- 1 Απλή ανάπτυξη Ιστοσελίδας
- 2 Γρήγορος αυτοματισμός
- 3 Cross platform scripting δηλαδή δημιουργία σεναρίων (script) που θα εκτελούνται σε διαφορετική πλατφόρμα από αυτήν που προορίζεται.

Εταιρίες που χρησιμοποιούν Python

- 1 Uber
- 2 Spotify
- 3 Netflix (krystal, 2021)

3.3.3 Η Γλώσσα SQL

Η SQL είναι μια γλώσσα ερωτημάτων που χρησιμοποιείται για αλληλεπίδραση μεταξύ των βάσεων δεδομένων. Η σύνταξη της είναι σχετικά απλή, είναι μια γλώσσα δηλωτική και είναι ιδανική για έναν αρχάριο. (krystal, 2021) Για να την κατανοήσουμε πιο εύκολα, οι χρήστες μιας πλατφόρμας όπως το Instagram, έχουν τα δεδομένα τους, τα οποία αποθηκεύονται σε βάσεις δεδομένων. Οι προγραμματιστές μπορούν να επεξεργαστούν και να δουλέψουν πάνω σε αυτές τις βάσεις δεδομένων με την γλώσσα SQL ή αλλιώς sequel.



Εικόνα 4 - MySQL

Συνοψίζοντας, η SQL παρέχει πρόσβαση, δημιουργία αλλά και επεξεργασίας μιας βάσης δεδομένων. Όσοι ασχολούνται ή εργάζονται επαγγελματικά με την ανάλυση δεδομένων, Big Data, η SQL είναι ιδανική επιλογή. Χρησιμοποιείται στον χρηματοοικονομικό κλάδο και σε τραπεζικές εφαρμογές, στις μουσικές εφαρμογές αλλά και σε πλατφόρμες κοινωνικής δικτύωσης καθώς όλα όσα προαναφέρθηκαν χρησιμοποιούν από πίσω τεράστιες βάσεις δεδομένων για την αποθήκευση και την επεξεργασία τους. (Stefanuk, 2021)

3.4 Τύποι APIs

Υπάρχουν 4 τύποι APIs (Application Programming Interface) που συναντούμε στην σχεδίαση των Ιστοσελίδων:

- Τα ανοιχτά APIs ή Public είναι διαθέσιμα σε προγραμματιστές και σε εξωτερικούς χρήστες και επικεντρώνονται σε δεδομένα. Τα Open APIs μπορεί να απαιτούν εγγραφή ή χρήση κλειδιού.
- Τα API των συνεργατών, τα οποία δεν είναι διαθέσιμα δημόσια όπως τα Public αλλά απαιτούν πρόσβαση σε αυτά. Είναι τα πιο ορατά και χρησιμοποιούνται πέρα από τα όρια και τις ανάγκες της εκάστοτε επιχείρησης. Συνήθως εκτίθενται σε προγραμματιστές που μπορούν να τα διαχειριστούν self-service. Όταν τα Public APIs είναι ανοιχτά, υπάρχει μια συγκεκριμένη διαδικασία επικύρωσης για να αποκτηθεί η πρόσβαση στα API των συνεργατών.
- Ιδιωτικά APIs, όπου είναι κρυμμένα για τους εξωτερικούς χρήστες και είναι διαθέσιμα μόνο για συστήματα στο εσωτερικό. Σε αντίθεση με τα δύο παραπάνω που είδαμε τα ιδιωτικά APIs δεν χρησιμοποιούνται για κατανάλωση εκτός αλλά

χρησιμοποιούνται σε ομάδες ανάπτυξης μέσα στην επιχείρηση για καλύτερη παραγωγή

- Composite APIs, όπου συνδυάζουν πολλαπλά δεδομένα. Επιτρέπουν τους προγραμματιστές να έχουν πρόσβαση σε πολλά σημεία με λίγες ενέργειες. Τα composite API είναι χρήσιμα γιατί σε μερικές ιδιαίτερες περιπτώσεις θέλουμε να συλλέξουμε πληροφορίες από πολλά δεδομένα ή υπηρεσίες με μια μόνο εργασία, κάτι το οποίο επιτυγχάνουν (Castellani, 2020).

3.4.1 Τύποι API πρωτοκόλλων

Για την αξιοποίηση αυτών των τύπων API, ακολουθούνται ορισμένα πρωτόκολλα από τους προγραμματιστές developers. Ένα πρωτόκολλο παρέχει συγκεκριμένους κανόνες και καθορίζει αποδεκτούς τύπους δεδομένων.

REST API

Το REST API είναι ένα API για web υπηρεσίες. Τα REST APIs αποτελούν βασικό μέρος μεγάλων και καθημερινών εφαρμογών μας όπως το Netflix, Amazon και άλλων. Για να είναι ένα API restful θα πρέπει να ακολουθεί τους εξής κανόνες:

1. Stateless, είναι από μόνο του ανεξάρτητο και ακολουθεί την αρχιτεκτονική πελάτη-διακομιστή.
2. Κοινή Διεπαφή, πελάτης και διακομιστής δηλαδή να επικοινωνούν μεταξύ τους μέσω HTTP πρωτόκολλων.
3. Οι αλλαγές που γίνονται στον server δεν πρέπει να επηρεάζουν τον πελάτη – client, πρέπει να είναι ανεξάρτητα ο πελάτης με τον διακομιστή.
4. Μνήμη Cache, από κάθε πελάτη να αποθηκεύονται οι απαντήσεις έτσι βελτιώνεται η εμπειρία χρήστη και καθιστά τις απαντήσεις αυτές αποτελεσματικές και ταχύτερες,
5. Πολυεπίπεδη αρχιτεκτονική και με το κάθε επίπεδο να συμβάλει σε μια σαφή ιεραρχία.

SOAP API

Το SOAP API είναι ένα πρωτόκολλο ίδιο σχεδόν με το REST και χρησιμοποιείται και αυτό κυρίως για web services. Χρησιμοποιήθηκε την δεκαετία του 1990 και ήταν το

πρώτο πρωτόκολλο που καθιέρωσε τον τρόπο με τον οποίο οι εφαρμογές χρησιμοποιούν το διαδίκτυο και τις υπηρεσίες. Το SOAP όμως έγινε αναγνωρίσιμο με αρκετά αυστηρούς κανόνες κάτι που σήμαινε υψηλούς πόρους και απαιτήσεις. Στις μέρες μας και εξαιτίας των υψηλών απαιτήσεων που έχει ένα SOAP API, οι προγραμματιστές προτιμούν την ανάπτυξη με REST.

RPC API

Το RPC είναι ένα απομακρυσμένο πρωτόκολλο καθώς είναι και το παλιότερο και το πιο απλό από τους τύπους API. Στόχος του RPC ήταν ο πελάτης να εκτελεί κώδικα σε έναν διακομιστή. Ένας client σε έναν server δηλαδή. Το πρωτόκολλο είναι ίδιο σχεδόν με το REST όμως έχουν μια βασική διαφορά, τα RPC είναι στενά συνδεδεμένα μεταξύ τους και αυτό καθιστά δύσκολη την επεξεργασία και την συντήρηση.

Συμπερασματικά, το REST API αξιοποιεί πλήρως όλα τα πρότυπα του παγκόσμιου Ιστού, είναι πιο απλό από τα πρότυπα που βασίζεται το SOAP και αν και ίδιο με το RCP, επιτρέπει μια πολυσταθμική αρχιτεκτονική που καθιστά εύκολη την διατήρηση. (Castellani, 2020)

Κεφάλαιο 4^ο

4. UI/UX Σχεδίαση και οι αμοιβές των Προγραμματιστών

Την σχεδίαση UI/UX (UI/UX design) συναντάται συχνά στον χώρο της σχεδίασης Ιστού. Συναντώνται συχνά οι δύο όροι μαζί, όπως τους φαίνονται γραμμένοι, όμως στην πραγματικότητα σημαίνουν δύο τελείως διαφορετικά πράγματα. Το UI αναφέρεται στην διεπαφή χρήστη γνωστό και ως User Interface ενώ το UX στην εμπειρία χρήστη, γνωστό ως User Experience. Τα δύο αυτά στοιχεία είναι σημαντικά για την σχεδίαση των Ιστοσελίδων και των εφαρμογών και αν και είναι διαφορετικά μεταξύ τους, το ένα ακολουθεί το άλλο.

4.1 Τι είναι η UI σχεδίαση

Όπως αναφέρθηκε ανωτέρω, η σχεδίαση διεπαφής χρήστη δεν είναι τόσο εύκολο να ερμηνευτεί καθώς έχει πάρει διάφορες εκδοχές στον τεράστιο όγκο του Ιστού. Η καλύτερη εκδοχή είναι ότι η διεπαφή χρήστη είναι η αλληλεπίδραση μεταξύ του χρήστη και μιας συσκευής τεχνολογίας, ένα tablet πολύ απλά. Στον σχεδιασμό της διεπαφής χρήστη δίνεται έμφαση στο στυλ και την εμφάνιση του προϊόντος, για αυτό δίνεται μεγάλη προσοχή στον σχεδιασμό αυτών των στοιχείων. (LAMPRECHT, 2021)

Ένας σχεδιαστής θα σκεφτεί το κάθε κουμπί και την κίνηση του, τα χρώματα και τις εικόνες που το πλαισιώνουν. Είναι εξίσου σημαντικό όσο και η σχεδίαση εμπειρίας χρήστη και πρέπει οι προγραμματιστές να κάνουν αυτή την σχεδίαση ελκυστική και αντάξια του συνολικού σχεδιασμού. Στον σχεδιασμό των Ιστοσελίδων η UI είναι οτιδήποτε έχει να κάνει με σχεδιασμό αυτών που βλέπουμε σε μια σελίδα στο διαδίκτυο, όπως το πάτημα των κουμπιών, τους πίνακες, τα πλαίσια, τις εικόνες και άλλα. (UX Planet, 2019)

4.2 Τι είναι η UX σχεδίαση

Το UX σημαίνει εμπειρία χρήστη, το πως ο χρήστης εκλαμβάνει την γνωριμία του με μια σελίδα ή με μια εφαρμογή. Η εμπειρία χρήστη είναι η εμπειρία που έχει ο χρήστης, αν δηλαδή αισθάνεται βολικά ή όχι με αυτό που έχει σχεδιάσει ο προγραμματιστής της διεπαφής. Ενώ οι σχεδιαστές διεπαφής χρήστη καθορίζουν την εμφάνιση, οι σχεδιαστές UX καθορίζουν το πως θα λειτουργεί η UI. Αυτός είναι και ο λόγος που τόσος κόσμος μπερδεύεται με τις δύο αυτές έννοιες. Εάν μια διεπαφή είναι ευχάριστη, γρήγορη, έξυπνη και δεν κουράζει τότε η εμπειρία του χρήστη-πελάτη θα είναι ικανοποιητική, εάν συμβεί το αντίθετο, δηλαδή να είναι μια διεπαφή αργή, παλιά, κουραστική τότε δεν θα αρέσει στον κόσμο και η δημοτικότητα της εφαρμογής ή της Ιστοσελίδας θα μειωθεί. Σε αυτό το κομμάτι δουλεύουν και θέλουν να το αποφύγουν οι σχεδιαστές UX. (UX Planet, 2019)

4.3 Πως τα UI/UX συνεργάζονται

Στην παρακάτω ενότητα θα δούμε τις ευθύνες και τις αρμοδιότητες που έχει ένας σχεδιαστής UX. Αρχικά, ένας σχεδιαστής UX έχει στρατηγική και πορεύεται με βάση αυτήν, αναλύει και γίνεται έρευνα στους πελάτες που θέλει να αναφερθεί, προσπαθεί να είναι πρωτότυπες οι ιδέες του, συγκρίνει άλλα πράγματα με την δική του σχεδίαση και κάνει το ανάλογο developing. Πιο συγκεκριμένα:

- i. Ανάλυση των ανταγωνιστών και πελατών
- ii. Ανάλυση της στρατηγικής του προϊόντος
- iii. Ανάπτυξη περιεχομένου
- iv. Δίνει έμφαση στην πρωτοτυπία
- v. Δοκιμάζει συνεχώς και επαναλαμβάνει τις διαδικασίες
- vi. Εξασκείτε πάνω στο πλάνο για την ιεραρχία της σχεδίασης.
- vii. Συνεργάζεται με τους UI και τους Web σχεδιαστές
- viii. Ενσωματώνει τους στόχους της σχεδίασης

Σε ένα τελικό συμπέρασμα θα λέγαμε πως ο σκοπός είναι να συνδεθούν οι στόχοι της επιχείρησης με τις ανάγκες των πελατών, ώστε να εξασφαλίζεται μια ευχάριστη και όμορφη διαδικασία που θα εξυπηρετεί και τις δύο πλευρές. (LAMPRECHT, 2021)

4.4 Ποιες οι ευθύνες του UI σχεδιαστή

Εάν σας αρέσει η εμφάνιση, το στυλ, η σχεδίαση εικόνων ή πλαισίων τότε θα ήσασταν κατάλληλος για την σχεδίαση διεπαφής χρήστη. Ένας σχεδιαστής διεπαφής χρήστη αναλύει τους πελάτες, όπως άλλωστε και ο σχεδιαστής UX, κάνει έρευνα του σχεδιασμού, εργάζεται στην ανάπτυξη των γραφικών, υλοποιεί 3D στοιχεία και συνεργάζεται σαφώς με τους προγραμματιστές. Πιο συγκεκριμένα:

- i. Ανάλυση πελατών
- ii. Έρευνα του σχεδιασμού
- iii. Ανάπτυξη γραφικών. συμπεριλαμβανομένων animation και 3D
- iv. Στοχεύει σε πρωτότυπα UI
- v. Συνεργάζεται με τους UI και τους Web σχεδιαστές

Όπως καταλαβαίνουμε ο ρόλος του σχεδιαστή UI είναι σημαντικός και καίριος για κάθε εφαρμογή – Ιστοσελίδα, καθώς αλληλοεπιδρά με τους πελάτες. Όπως είδαμε παραπάνω ενώ το UI δεν χρειάζεται κάποια γνώση κωδικοποίησης, όσο περνάει ο καιρός και εξελίσσεται ο χώρος του Internet οι απαιτήσεις θα μεγαλώνουν συνεχώς και ίσως καταστεί απαραίτητο εργαλείο η γνώση κώδικα. Θα μπορούσαμε από την άλλη να πούμε πως αν κάποιος έχει καλή τεχνογνωσία CSS ή React μπορεί να δημιουργήσει animations ή διεπαφές που θα ήταν απαραίτητες για πολλές διαδικτυακές εφαρμογές αλλά και σελίδες ιστού. (LAMPRECHT, 2021)

4.5 Μέσες αμοιβές των Developer (UI/UX, Frontend, Backend, Full Stack)

Ο μισθός των Developer, ανάλογα την κατηγορία όπου ανήκει διαμορφώνεται. Οι παράγοντες που διαμορφώνουν τον μισθό είναι η χώρα που εργάζεται κάποιος, η εμπειρία και η γνώση που κατέχει ο σχεδιαστής, το έργο και ο χρόνος που θα διαρκέσει, ο τίτλος και η ιδιοσυγκρασία της επιχείρησης αλλά σαφώς και η προϋπηρεσία που διαθέτει ο σχεδιαστής.

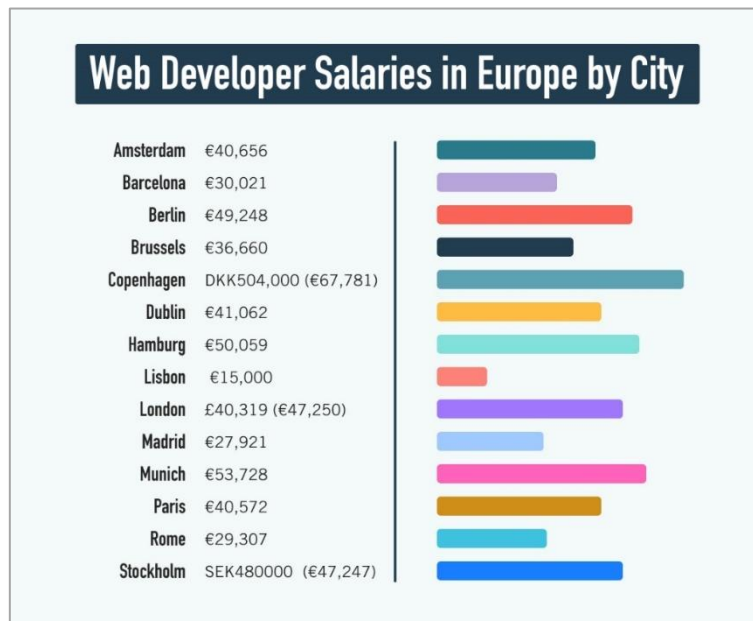
UI/UX

Μέσος ετήσιος μισθός, σύμφωνα με πηγές από την κεντρική Ευρώπη για UI/UX Designer ξεκινάει από 28-33 χιλιάδες ευρώ, ενώ για έναν εξειδικευμένο σχεδιαστή κυμαίνεται στα 50 με 80 χιλιάδες ευρώ. (LAMPRECHT, 2021)

Frontend Developer

- Αυστραλία - 58.000\$
- Γερμανία - 51.000 ευρώ
- Ηνωμένο Βασίλειο - 45.000\$
- Ηνωμένες Πολιτείες Αμερικής - 70.000\$

Όπως είπαμε αυτά τα δεδομένα διαφέρουν, ανάλογα την εμπειρία, το project, το επιχειρηματικό όραμα και τους στόχους της επιχείρησης – εταιρείας.



Εικόνα 5 – Μισθοί ανά πόλη στην Ευρώπη

Όπως θα παρατηρεί ο καθένας, υπάρχουν σε μερικές περιοχές τεράστιες διαφορές σε σχέση με άλλες, αλλά αυτό αλλάζει ανάλογα τη χώρα (ασφάλιση) και τον χώρο εργασίας (bonus, υπερωρίες). (Stevens, 2021)

Backend+ Full stack Developer

Ο λόγος που δεν διαχωρίζονται, όπως έγινε και παραπάνω είναι γιατί δεν υπάρχει ουσία. Εάν κάποιος ανατρέξει στα δεδομένα της εργασίας από την αρχή θα διαπιστώσει ότι Frontend, Backend, Full stack, είναι κυρίως η κωδικοποίηση. Η διαφορά μεταξύ τους είναι στην δουλειά που έχει να κάνει ο καθένας. Επομένως, στην παραπάνω εικόνα εάν ένας Front σχεδιαστής στο Βερολίνο έπαιρνε 50.000 ευρώ ετησίως, αν δήλωνε Backend θα έπαιρνε 60-65.000 ευρώ και αν δήλωνε Full stack θα έπαιρνε 70-75.000 ευρώ. Να τονιστεί εδώ ότι ένας Full stack σχεδιαστής απλά γνωρίζει να κάνει την δουλειά και του Frontend και του Backend, ξέρει δηλαδή να προγραμματίζει και browser και server και βάσεις δεδομένων, συνεπώς και λόγω των γνώσεων του θα αμειφθεί με περισσότερα χρήματα!

Συμπερασματικά, όπως όμως τονίσαμε οι μισθοί στον τομέα του σχεδιασμού Ιστού (Front, Back, Full, UI/UX, WordPress, CMS) αλλάζουν και οι παράγοντες που είναι υπεύθυνοι για αυτό είναι αρκετοί. Για να δοθεί ένα ξεκάθαρο παράδειγμα για αυτό, ένας Frontend σχεδιαστής που ξέρει άπταιστα γλώσσα Javascript, με ην βιβλιοθήκη node.js και τα frameworks που προστίθενται τα τελευταία χρόνια, έχει ισοδύναμες γνώσεις με έναν που δηλώνει Backend developer σε υψηλό επίπεδο προγραμματισμού.

Κεφάλαιο 5^ο

5. Σύστημα Διαχείρισης Περιεχομένου (CMS)

Σε αυτή την εργασία θα ασχοληθούμε με το Σύστημα Διαχείρισης Περιεχομένου καθώς θα δημιουργήσουμε ένα ηλεκτρονικό κατάστημα σε WordPress. Πριν προχωρήσουμε ωστόσο στην διαδικασία αυτή θα αναλύσουμε τι είναι ακριβώς το CMS, ποια είναι τα θετικά και τα αρνητικά του, πόσο εύκολο είναι στην χρήση του και ποια είναι τα πιο δημοφιλή CMSs ως τώρα στην κοινότητα.



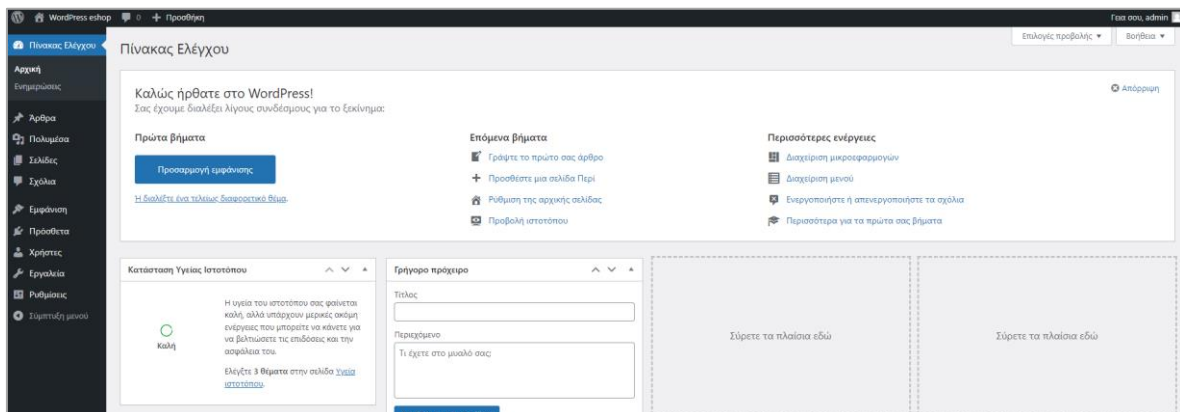
Εικόνα 6 - CMS

Σύστημα Διαχείρισης Περιεχομένου λοιπόν είναι μια εφαρμογή λογισμικού που επιτρέπει στον οποιοδήποτε, καθώς δεν απαιτεί απαιτητικές γνώσεις προγραμματισμού, να διαχειρίζεται το περιεχόμενο του όπως εικόνες, βίντεο, κείμενα κ.α.. Οι αλλαγές αυτές γίνονται μέσω κάποιου html editors και μοιάζουν σαν την επεξεργασία μέσω της ζωγραφικής ή ακόμη καλύτερα σαν του Microsoft Word. Με το σύστημα διαχείρισης

περιεχομένου προσαρμόζεται η σχεδίαση του Ιστότοπου, μέσω έτοιμων προτύπων (templates) και επεκτάσεων (plug-in) που στην συνέχεια παραμετροποιούνται κατά το δοκούν. (Fitzgerald, n.d.)

5.1 Πως δουλεύει ένα CMS

Στην παρακάτω Εικόνα 8 παρουσιάζεται ο Πίνακας διαχείρισης (Dashboard) του WordPress, από όπου πραγματοποιούνται όλες οι τροποποιήσεις. Θα παρουσιαστεί παρακάτω με λεπτομέρεια, όταν θα αναλυθεί η υλοποίηση του ηλεκτρονικού καταστήματος (eshop) και η προσθήκη των θεμάτων και των plugins. Πριν έρθουν τα CMS στο προσκήνιο ο τρόπος που κάποιος έφτιαχνε τον Ιστότοπο του είναι όμοιος με αυτόν που είδαμε στα κεφάλαια ένα και δύο, με την σύνταξη HTML, CSS και JavaScript.



Εικόνα 8 - Πίνακας Ελέγχου CMS (WordPress)

Ουσιαστικά, το CMS αποτελείται από μια εφαρμογή διαχείρισης του περιεχομένου (CMA) και μια εφαρμογή παράδοσης του περιεχομένου (CDA), αυτές οι δύο εφαρμογές εκτελούν όλες τις λειτουργίες ώστε να μπορεί ο χειριστής της σελίδας να εστιάσει στο Frontend σχεδιασμό. Αντί λοιπόν εκκινήσουμε με μια κενή σελίδα HTML αρχείου, ανοίγουμε το πρόγραμμα επεξεργασίας κειμένου πραγματοποιώντας τις απαραίτητες ενέργειες. (Fitzgerald, n.d.)

Στα αριστερά, όπου βρίσκεται ο πίνακας διαχείρισης, υπάρχουν πάρα πολλές λειτουργίες, όπου ο χρήστης μπορεί να χρησιμοποιήσει. Αν επιθυμεί για παράδειγμα να τροποποιήσει το θέμα, να προσθέσει ή να διαγράψει σελίδες και περιεχόμενα, να προσθέσει επεκτάσεις, όλα αυτά υλοποιούνται με την εφαρμογή CMA, όπου στη συνέχεια το CDA το λαμβάνει

και το εμφανίζει στον επισκέπτη. Όλη αυτή η αυτοματοποιημένη διαδικασία γίνεται στο παρασκήνιο. (Fitzgerald, n.d.)

5.2 Γιατί να χρησιμοποιήσουμε CMS

Έχουν ήδη αναφερθεί μερικά από τα πλεονεκτήματα της διαχείρισης περιεχομένου, όμως παρακάτω θα παρουσιαστούν λίγο πιο αναλυτικά για να γίνουν αντιληπτές καλύτερα κάποιες πτυχές. Εν γένει, ένα σύστημα διαχείρισης περιεχομένου:

- Δεν απαιτεί κωδικοποίηση: Η κωδικοποίηση για να δημιουργήσει κάποιος τον δικό του Ιστότοπο είναι και χρονοβόρα και απαιτεί τις απαραίτητες γνώσεις, ακόμη και θέληση. Για να χρησιμοποιήσει κάποιος τον απλό τρόπο HTML→CSS→JS χρειάζεται να ξοδέψει αρκετές ώρες για να μάθει αυτές τις γλώσσες, ενδεχομένως και επιπλέον χρήματα. Το σύστημα διαχείρισης περιεχομένου έρχεται για να απλοποιήσει την υλοποίηση καθώς οτιδήποτε αλλαγές πραγματοποιούνται μέσα στο Dashboard και δεν χρειάζεται καθόλου γνώση κώδικα. Συνεπώς, είναι εύκολο για κάποιον αν μελετήσει λίγες μέρες να καταφέρει να φτιάξει και να συντηρήσει το δικό του online κατάστημα, την δική του Ιστοσελίδα ή το portfolio του.
- Διευκολύνει την Συνεργασία κατά την υλοποίηση: Με ένα CMS ο καθένας μπορεί να έχει πρόσβαση στην σελίδα. Δηλαδή πολλά άτομα μαζί μπορούν να συνεργάζονται ταυτόχρονα ώστε να βγει το καλύτερο αποτέλεσμα. Ένα CMS λοιπόν καθιστά δυνατή την ταυτόχρονη υλοποίηση ώστε η παραγωγή να προχωράει με ταχύτατους ρυθμούς.
- Έχει ρόλους και δικαιώματα χρηστών, έτσι εάν ένας προγραμματιστής έχει δημιουργήσει μια σελίδα, μπορεί ως διαχειριστής να χειρίζεται και να επεξεργάζεται όλο το σύστημα, αλλά μπορεί να έχει ορίσει και έναν έναν συνιδιοκτήτη (co-founder) το οποίοι να του ορίσει την διαχείριση από κοινού π.χ. για το 95% των πλήρων δικαιωμάτων του συστήματος. Αυτό σημαίνει ότι αυτός θα μπορεί να επεξεργαστεί, να προσθέσει, να διαγράψει κ.α.. Μπορεί να του επιτρέψει να τροποποιήσει για παράδειγμα συγκεκριμένες σελίδες και όχι όλες τις σελίδες του Ιστοτόπου. Αυτό γίνεται με ένα επιπρόσθετο plugin που ονομάζεται User Role Editor, όπου δημιουργείται ο χρήστης που θέλουμε να δώσουμε δικαιώματα και ανάλογα την δουλειά που αναθέτουμε σε κάποιον το παραμετροποιούμε. Αυτό για

να πραγματοποιηθεί με JavaScript και χωρίς την βοήθεια του CMS απαιτεί την δημιουργία και επεξεργασία πολλών συνθηκών για τα δικαιώματα χρηστών.

- Το σύστημα διαχείρισης περιεχομένου και οι επεκτάσεις που παρέχει βοηθούν την σελίδα να είναι φιλική προς τις μηχανές αναζήτησης (SEO). Με αυτά τα πρόσθετα η σελίδα μπορεί να προσαρμόζει τους τίτλους της, επιλέγει παρεμφερή url προς τα SEO, δημιουργεί χάρτες XML γνωστό ως sitemap στο WordPress για την τοποθέτηση του site στις αναζητήσεις της Google και να δημιουργούμε redirects και να βελτιώνουμε τους χρόνους φόρτωσης της σελίδας.
- Υπάρχουν και οι επεκτάσεις ασφαλείας ή το πιστοποιητικό SSL που είναι αυτό που πιστοποιεί ότι ο επισκέπτης θα πλοηγηθεί με ασφαλή λειτουργία και την προστασία των πνευματικών δικαιωμάτων του.
- Προσχεδιασμένα Themes και Templates που παρέχουν την ευχέρεια για αλλαγές on the fly στην σελίδα μας, εξοικονομώντας χώρο και χρόνο που θα χρειαζόμασταν για να το σχεδιάσουμε όλο εξαρχής. Γίνεται όμως κάποιος να μη θέλει να χρησιμοποιήσει κανένα έτοιμο ή επι πληρωμή θέμα και να φτιάξει την σελίδα όπως επιθυμεί.
- Εύκολη πρόσβαση: είναι το σημαντικότερο που παρέχει οποιοδήποτε σύστημα CMS και σε οποιαδήποτε συσκευή. Όλα είναι στον πίνακα ελέγχου ή το Dashboard και οποιαδήποτε αλλαγή συμβαίνει από εκεί. Στην κουβέντα για το πιο CMS είναι καλύτερο καθώς υπάρχουν πολλές πτυχές, και θα επιχειρηθεί προσπάθεια ανάλυσης τον πιο δημοφιλή. (Fitzgerald, χ.χ.)

Υπάρχουν πολλά ειδών CMS. Στα επόμενα κεφάλαια θα αναλυθούν τα πιο σημαντικά, καθώς και το CMS που χρησιμοποιήθηκε στην παρούσα εργασία και του λόγους για τους οποίους επιλέχθηκε. Στην επόμενη ενότητα μελετηθεί το Joomla, το Drupal και το WordPress.

5.3 Το Σύστημα Διαχείρισης Περιεχομένου Joomla!

Όπως έχει αναφερθεί στην αρχή του κεφαλαίου, το Σύστημα Διαχείρισης Περιεχομένου βοηθά να δημιουργηθούν Ιστότοποι και εφαρμογές χωρίς την ιδιαίτερη γνώση προγραμματισμού, εύκολα και γρήγορα! Ένα από αυτά τα συστήματα ονομάζεται Joomla ! και είναι ένα από τα πιο διαδεδομένα στον κόσμο.

Το Joomla ξεκίνησε το 2005 και εξελίχθηκε σε ένα από τα πιο διαδεδομένα CMS με περισσότερες από 99+ εκατομμύρια λήψεις ως σήμερα. Το Joomla έχει προγραμματιστεί σε γλώσσα PHP και χρησιμοποιεί την SQL για την διαχείριση των βάσεων δεδομένων. Υπάρχουν πολλά πακέτα φιλοξενίας στο διαδίκτυο για να εκκινήσει κάποιος να αναπτύσσει έναν ιστότοπο με το Joomla. Το Joomla χρησιμοποιεί τεχνική MVC που σημαίνει Model-View Controller δηλαδή όταν επεξεργάζεται ένα αίτημα, αναλύει το URL και καθορίζεται έπειτα το στοιχείο που θα επεξεργαστεί. Το MVC περιέχει τα δεδομένα του στοιχείου και έχει αποκλειστική ευθύνη να ενημερώνεται η βάση δεδομένων όταν αυτό χρειαστεί.

Παρακάτω είναι μερικά από τα βασικά χαρακτηριστικά του:

1. Είναι πολύγλωσσο, υποστηρίζει 75 γλώσσες και εξυπηρετεί τις ανάγκες όσων δεν έχουν την δυνατότητα να είναι γνώστες αγγλικών.
2. Στην γλώσσα των Ιστοσελίδων τα τελευταία χρόνια έχει προστεθεί η λέξη Responsive, η οποία σημαίνει ότι ανταποκρίνεται σε κάθε συσκευή (PC, phone, tablet) έτσι από όπου και αν μπαίνετε στην σελίδα σας το Joomla ανταποκρίνεται σε αυτό το κομμάτι άψογα.
3. Είναι πολύ εύκολο στην χρήση, όπως όλα τα CMS είναι η αλήθεια με συνεχής ενημερώσεις και δεν θα υπάρξει κανένα πρόβλημα στην διαδικασία δημιουργίας της προσωπικής σας σελίδας.
4. Η ασφάλεια παίζει σημαντικότατο ρόλο στις σελίδες διαχείρισης περιεχομένου, πόσο μάλλον αν μιλάμε για ηλεκτρονικά καταστήματα, συνεπώς το Joomla παρέχει δύο παράγοντες ταυτοποίησης και έτσι είναι αδύνατο έως ακατόρθωτο να πέσει κάποιος θύμα hacking, αλλά επειδή στις μέρες μας το δίκτυο έχει αλλάξει, εάν έπεφτε κάπως θύμα hacking μπορούσε να ανακτήσει όλα τα δεδομένα και την σελίδα του πίσω.
5. Υπάρχει το Joomla Forum όπου παρέχεται βοήθεια σε τυχόν προβλήματα αλλά γίνονται και συζητήσεις για καινοτόμες ιδέες που έχει κάποιος. Επιπλέον, είναι ένα φιλικό περιβάλλον και γνωρίζεις πολλά νέα άτομα που η επικοινωνία μπορεί να σας βοηθήσει καλύτερα στην δημιουργία των σελίδων σας. (Chahal, 2021)

5.3.1 Γιατί να χρησιμοποιήσει κάποιος Joomla και ποιοι το έχουν επιλέξει

Το σύστημα διαχείρισης περιεχομένου Joomla είναι εντελώς δωρεάν, εκτός λίγων εξαιρέσεων που είναι επί πληρωμή. Είναι ανοιχτού κώδικα πράγμα που σημαίνει ότι δεν θα αντιμετωπίσει κάποιος ιδιαίτερα προβλήματα, μπορεί κάποιος να δημιουργήσει ότι θέλει και όπως το θέλει. Η κοινότητα που έχει δημιουργηθεί για το Joomla έχει αναπτύξει ιδέες και διάφορες επεκτάσεις (θέματα κλπ) που δίνουν τη δυνατότητα για την άψογη επεξεργασία του Ιστοτόπου χωρίς ιδιαίτερες γνώσεις προγραμματισμού. Υπάρχει ένας αριθμός εταιρειών που η ενασχόληση τους είναι αποκλειστικά με Joomla καθώς παρέχει όλη την άνεση και την ευκολία που είπαμε. Το Joomla είναι από τις μεγαλύτερες πλατφόρμες για κυβερνητικούς οργανισμούς στον κόσμο ενώ χρησιμοποιείται για ηλεκτρονικό εμπόριο, εκπαιδευτικές γνώσεις, προσωπικές Ιστοσελίδες, portfolio και άλλα. Εν κατακλείδι, το σύστημα διαχείρισης περιεχομένου Joomla, όπως και όλα τα CMS άλλωστε, δεν απαιτούν γνώσεις προγραμματισμού και κάποιος οτιδήποτε ηλικία και να έχει μπορεί να ξεκινήσει σχεδιάζοντας την δική του σελίδα. Δημοφιλής και σημαντικοί οργανισμοί, κυβερνητικοί και μη χρησιμοποιούν Joomla ανάμεσα τους το Πανεπιστήμιο του Harvard, η LINUX, το κέντρο πληροφοριών Ηνωμένων Εθνών και το ομοσπονδιακό εφετείο των ΗΠΑ. (RocketTheme, n.d.)

5.3.2 Joomla στατιστικά κατά το 2021

Η πιο δημοφιλής επιλογή CMS στον κόσμο τη δεδομένη χρονική στιγμή είναι το WordPress με ένα ποσοστό πολύ υψηλό της τάξης του 60% του Ιστού, ακολουθεί το Joomla, πίσω από το Joomla το Drupal και συνεχίζουν άλλα, μικρότερης εμβέλειας συστήματα διαχείρισης περιεχομένου.

1. Μεγαλώνει σε δημοτικότητα. Αν και είναι αρκετά πίσω σε ανταγωνιστικότητα από το WP, συνεχίζει με μικρά βήματα να έχει ένα μικρό ποσοστό σε εφαρμογές και σε σελίδες.
2. Είναι το δεύτερο δημοφιλέστερο σύστημα διαχείρισης περιεχομένου στον κόσμο.
3. Κατέβηκε σε υπολογιστές χρηστών 123 εκατομμύρια φορές, 26 εκατομμύρια περισσότερες από το 2020.
4. Το 2,2% των σελίδων στο Internet τρέχουν σε Joomla, ακούγεται λίγο αλλά ο όγκος του Ιστού είναι αχανής στις μέρες μας.
5. 2,5 εκατομμύρια ιστότοποι χρησιμοποιούν Joomla

6. Περισσότερο από το 9% των σελίδων των επιχειρήσεων χρησιμοποιούν Joomla
7. Έχει καλύτερη ασφάλεια και προστασία από το WordPress καθώς σε αναγκάζει να βάλεις SSL, κάτι που δεν ισχύει στα άλλα CMS (Galov, 2021)

5.3.3 Joomla vs WordPress

Τα δύο αυτά συστήματα διαχείρισης περιεχομένου όπως προαναφέρθηκε είναι αυτό-φιλοξενούμενα (self-hosted) και βρίσκονται στο χώρο για πάνω από μια δεκαετία.



Εικόνα 10 - WordPress Logo



Εικόνα 11 – Joomla Logo

Το CMS WordPress κυκλοφόρησε αρχικά για την δημιουργία Blogs, έπειτα πήρε αυτή τη μεγάλη έκταση ως ένα σύστημα διαχείρισης περιεχομένου που καταλαμβάνει το 43% του διαδικτυακού Ιστού. Κυκλοφόρησε το 2005, εξίσου παλιό με το WP, πίσω όμως σε δημοτικότητα από το WP κατέχει το 3% σχεδόν, των ιστοσελίδων του διαδικτύου.

Σύμφωνα με τα στατιστικά και τα Google Trends τα δύο αυτά συστήματα διαχείρισης περιεχομένου είχαν την ίδια απήχηση έως το 2010, όπου και άρχισε να κυριαρχεί το WordPress.

Πλεονεκτήματα WordPress

- Θεωρείται το πιο εύκολο στην χρήση σύστημα διαχείρισης περιεχομένου στον κόσμο
- Καταλληλότερο για blogging, το WP είναι καλύτερο για αναρτήσεις σελίδων, ενώ το Joomla περιλαμβάνει ένα άρθρο που απλά μπορείς να χειριστείς τις ενότητες
- Επεκτάσεις και θέματα. Διαθέτει την μεγαλύτερη και ίσως την ομορφότερη γκάμα θεμάτων και templates στην αγορά
- Χαμηλότερο κόστος στις εργασίες ανάπτυξης και στη δημιουργία Ιστοσελίδων

Πλεονεκτήματα Joomla

- Καλύτερη διαχείριση χρηστών Joomla και καλύτερος έλεγχος πρόσβασης
- Ευελιξία για τους διαφορετικούς τύπους δεδομένων που συναντάμε στις ενότητες και στα στοιχεία του Joomla
- Υποστήριξη πολλών γλωσσών σε αντίθεση με το WP που θέλει πρόσθετη επέκταση για την ίδια δουλειά
- Επιτρέπει την χρησιμοποίηση πολλών προτύπων για τα κομμάτια περιεχομένου ενώ στο WP χρησιμοποιούμε μόνο ένα θέμα (Jackson, Kinsta, 2021)

Εν κατακλείδι, πρέπει να τονιστεί ότι στο ερώτημα "Ποιο σύστημα διαχείρισης περιεχομένου να επιλέξω", η απάντηση είναι ανάλογη με το είδος της εργασίας που θέλεις να πραγματοποιήσει. Πρώτο σε προτιμήσεις χρηστών είναι το WordPress, διότι είναι πιο φιλικό, εκκινούν γρήγορα και μικρό κόστος τα Project σε αυτό και σε σχέση με άλλα CMS διαθέτει μεγαλύτερη κοινότητα και πιο επαγγελματική υποστήριξη.

5.4 Το Σύστημα Διαχείρισης Περιεχομένου Drupal

Το Σύστημα Διαχείρισης Περιεχομένου Drupal είναι ευέλικτο και βασίζεται στην LAMP stack. Σε LAMP (Linux, Apache, MySQL, PHP) τρέχουν ορισμένες δημοφιλής εφαρμογές Ιστού με βασικό σενάριο την PHP, πολλές εφαρμογές σε JavaScript και CSS. Το Drupal μπορεί να τρέξει εξίσου καλά και σωστά σε Windows ή Mac OS αντί για Linux (Drupal, 2021).



Εικόνα 12 - Drupal Logo

Το Drupal είναι ένα λογισμικό ανοιχτού κώδικα όπως τα υπόλοιπα με πλεονεκτήματα όπως χαμηλό κόστος και υψηλή ασφάλεια. Είναι δωρεάν, επομένως δίνει τη δυνατότητα σε οποιονδήποτε να το αποκτήσει και να το χρησιμοποιήσει. Είναι γεμάτο με αρκετούς τύπους περιεχομένων, πολυμέσα και κείμενα, ενώ υπάρχουν πολλά εργαλεία επεξεργασίας, όπως είναι ένα CMS. Το Drupal είναι επίσης μια πλατφόρμα ανάπτυξης

σελίδων και ακολουθεί τα πρότυπα του αντικειμενοστραφή προγραμματισμού και της HTML., ενώ έχει πολλά θέματα για να επιλέξει κάποιος που θα το χρησιμοποιήσει για την δουλειά του και μια μεγάλη γκάμα προσθέτων. (drupalizeme, n.d.)

Το Drupal δίνει τις εξής δυνατότητες σε όποιον χρήστη θέλει να σχεδιάσει με αυτό:

- Είναι Site Builder, δηλαδή, μπορεί να δημιουργήσει διαφορά πρότυπα και προσαρμοσμένες λίστες. Μπορεί επίσης να σχεδιαστεί η πλοήγηση σε site με διαφορά menus.
- Είναι Front-End Developing, δίνει τη δυνατότητα για να σχεδιάσει κάποιος ολοκληρωτικά την Ιστοσελίδα του η το portfolio του επεξεργάζονται τεχνολογίες όπως HTML CSS και JavaScript. Σε μερικά templates θα συναντήσει ο χρήστης και PHP.
- Είναι καθαρά ένα Developing tool καθώς μπορεί κάποιος προγραμματιστής Ιστού με όλες τις γνώσεις που έχει να χρησιμοποιήσει τις βασικές αρχές του Developing που είναι οι PHP,MySQL,HTML, JavaScript, CSS για να βελτιώσει και να αναβαθμίσει τον Ιστοτοπο. (drupalizeme, n.d.)

5.4.1 Λειτουργικότητα Drupal

Το Drupal έχει πολλές λειτουργίες που χρειάζεται ένας Ιστότοπος να ανταπεξέλθει στην αγορά. Μερικές από αυτές είναι:

- Λειτουργικότητα Drupal
- Διαχείριση Περιεχομένου (CM)
- Ευέλικτο σύστημα πλοήγησης
- Αναζήτηση
- Φόρμες επικοινωνίας
- Πολύγλωσσο
- Διαχείριση χρηστών
- Responsive Design
- Προσιτότητα (Koga, 2018)

5.4.2 Drupal vs WordPress

Το Drupal υπάρχει στην αγορά από το 2000 παρόλα αυτά το WordPress έχει αποδειχθεί μονοπώλιο για την κατασκευή Ιστοσελίδων. Το Drupal κατέχει το 2,3 όλων των Ιστοσελίδων και έχει μερίδιο αγοράς στα Συστήματα Διαχείρισης Περιεχομένου 4,6%.



Εικόνα 13 - Drupal vs WordPress

Μεγάλοι Ιστότοποι που χρησιμοποιούν Drupal είναι η πολιτεία του Κολοράντο, το Πανεπιστήμιο του Κολοράντο καθώς και το nasa.gov.

Πλεονεκτήματα WordPress

- Θεωρείται το πιο εύκολο στην χρήση σύστημα διαχείρισης περιεχομένου στον κόσμο
- Καταλληλότερο για blogging, το WP είναι καλύτερο για αναρτήσεις σελίδων, ενώ το Joomla περιλαμβάνει ένα άρθρο που απλά μπορείς να χειριστείς τις ενότητες
- Επεκτάσεις και θέματα. Διαθέτει την μεγαλύτερη και ίσως την ομορφότερη γκάμα θεμάτων και templates στην αγορά
- Χαμηλότερο κόστος στις εργασίες ανάπτυξης και στη δημιουργία Ιστοσελίδων

Πλεονεκτήματα Drupal

- Προσαρμοσμένοι τύποι περιεχομένου που θεωρούνται από πολλούς πιο ευέλικτοι σε σχέση με τις συναρτήσεις που προσφέρει το WordPress.
- Το Drupal παρέχει τη δυνατότητα των πολύγλωσσων λειτουργιών σε αντίθεση με το WordPress που πρέπει να χρησιμοποιηθούν επιπλέον πρόσθετα.

- Θεωρείται πως το σύστημα ταξινόμησης του Drupal είναι πιο εξελιγμένο σε σχέση με τον αντίπαλο του WordPress. (Jackson, kinsta, 2021)

5.4.3 Πόσο ευκολόχρηστα και ασφαλή είναι τα δύο συστήματα

Η χρήση του WordPress είναι αρκετά απλή καθώς από το πουθενά μέσα σε λίγες μόνο ώρες μπορείς να χτίσεις το δικό σου website. Δεν απαιτεί γνώση προγραμματισμού και οι αλλαγές που συμβαίνουν εκείνη την στιγμή (on the fly), είναι απλές και εύκολες. Από την άλλη πλευρά το Drupal περιέχει προσαρμοσμένα θέματα που απαιτούν εξειδίκευση και σίγουρα απαιτείται μελέτη ή εκ των προτέρων γνώσεις για να εκκινήσει ένα έργο. Η διεπαφή του Drupal είναι αρκετά καλή και κατανοητή παρόλα τα παραπάνω ενώ σε ένα γενικό πλαίσιο θα περιμέναμε το Drupal να προσφέρει μια καλύτερη και πιο ευκολόχρηστη εμπειρία περιεχομένου.

Το WordPress έχει 53.000 πρόσθετα (plugins) σε αντίθεση με το Drupal που έχει μόλις 39.000 αρθρώματα (modules) όπως ονομάζονται στο Drupal. Επίσης, το WP προσφέρει 5.000 δωρεάν θέματα σε αντίθεση με το Drupal το οποίο προσφέρει 250.

WordPress Security

Η ασφάλεια με το σύστημα περιεχομένου WordPress δεν θα πρέπει να ανησυχεί κανέναν και για να γίνει αυτό πιο γενικό, κανένα CMS δεν αντιμετωπίζει μεγάλο πρόβλημα με την ασφάλεια του. Το WordPress είναι ασφαλές από μόνο του, όμως βασίζεται πάρα πολύ σε κώδικα τρίτων και αυτό οδηγεί σε κενά ασφαλείας πολλές των περιπτώσεων, αυτό από μόνο του το καθιστά ευάλωτο σε αντίθεση με το Drupal. Κατά γενική ομολογία όμως θεωρείται ασφαλές σύστημα.

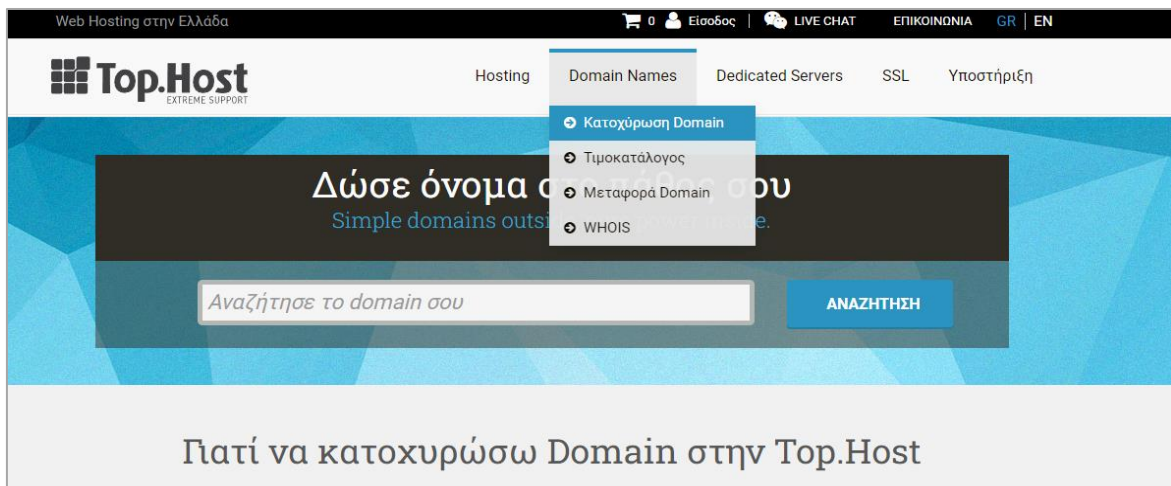
Drupal Security

Μετά από έρευνα που διεξήχθη για τα συστήματα και την ασφάλεια του καθενός, μόνο το 2% των παραβιασμένων Ιστότοπων ήταν Drupal, ενώ το μερίδιο του στην αγορά είναι 4.7% γεγονός που σημαίνει ότι είναι απόλυτα αξιόπιστο στα θέματα ασφαλείας. Αρκετοί κυβερνητικοί σκοποί χρησιμοποιούν Drupal μόνο για αυτό το λόγο, ενώ δημοσιεύει και αναφορές ασφαλείας κάτι που το ανεβάζει ένα σκαλί παραπάνω στην αξιοπιστία των πελατών. (Jackson, kinsta, 2021)

Κεφάλαιο 6^ο

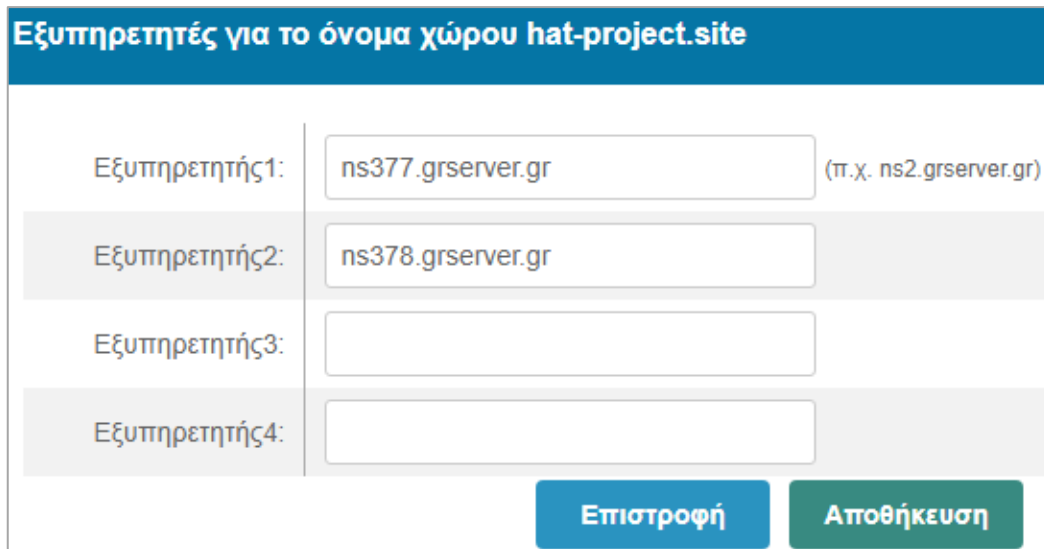
6. Ηλεκτρονικό Κατάστημα με WordPress

Στην εργασία που εκπονήθηκε θα αναδειχθεί πως δύναται πολύ απλά και εύκολα να δημιουργηθεί ένα ηλεκτρονικό κατάστημα – eshop με την βοήθεια του CMS WordPress. Το πρώτο πράγμα που πρέπει να έχει ένα ηλεκτρονικό κατάστημα είναι ένα *domain name*. Αρχικά επιλέγουμε εταιρεία βάση του κόστους και τις αξιοπιστίας. Για την ηλεκτρονικό κατάστημα της εργασίας επελέγη η ελληνική εταιρεία Top.Host, διότι υπάρχει δυνατότητα επικοινωνίας σε LiveChat και στα ελληνικά.



Εικόνα 14 - Κατοχύρωση Domain Name

Στην εικόνα 14 φαίνεται η είσοδος στην Top.Host. Επιλέγοντας το πεδίο Domain Names - > Κατοχύρωση Domain και στη συνέχεια στο πλαίσιο αναζήτησης εισάγεται το επιθυμητό όνομα για τον Ιστότοπο. Στην περίπτωση της εργασίας είναι το WWW.HAT-PROJECT.SITE. Σε περίπτωση που αυτό ήδη χρησιμοποιείται εξάγεται αυτόματα στο site καρτέλα με παρόμοια ονόματα ή απλά πληκτρολογείτε ένας άλλο όνομα. Προσοχή απαιτείται διότι, άλλη χρέωση θα έχει η αγορά του Domain μας με μια κατάληξη. site όπως το δικό μου, άλλη με.gr και άλλη με.com.



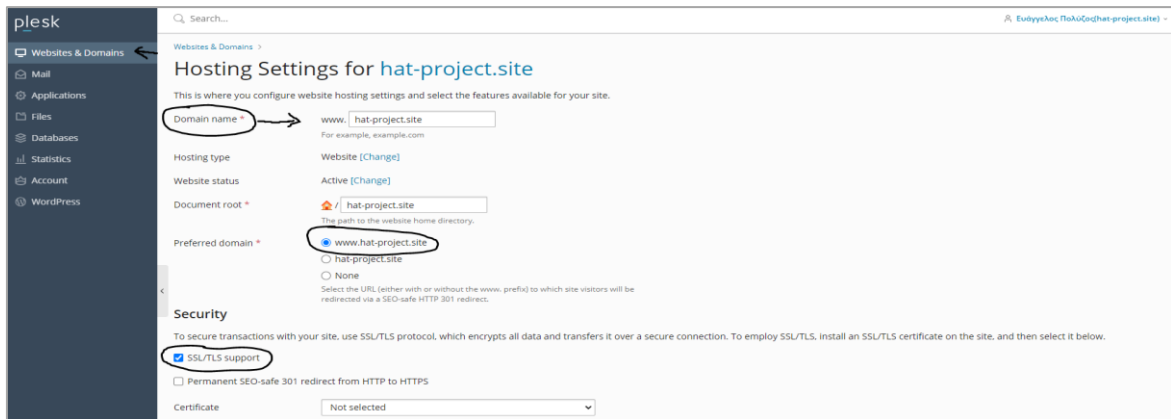
Εξυπηρετητές για το όνομα χώρου hat-project.site	
Εξυπηρετητής1:	<input type="text" value="ns377.grserver.gr"/> (π.χ. ns2.grserver.gr)
Εξυπηρετητής2:	<input type="text" value="ns378.grserver.gr"/>
Εξυπηρετητής3:	<input type="text"/>
Εξυπηρετητής4:	<input type="text"/>
<input type="button" value="Επιστροφή"/> <input type="button" value="Αποθήκευση"/>	

Εικόνα 15 - Εξυπηρετητές

Καθώς τελειώνει αυτό το βήμα πρέπει να επιλεγεί και ένα πακέτο δηλαδή ένα Host. Τα πακέτα αυτά είναι ανάλογα με την εργασία για την οποία απαιτούνται, και με την χρονική τους διάρκεια (3 μήνες, 6 μήνες, 1 έτος κ.ο.κ). Επιλέγη τρίμηνο πακέτο διότι προς το παρόν το απαιτείται για τους εκπαιδευτικούς σκοπούς της εργασίας. Εν κατακλείδι, όταν αγοράσουμε Domain Name, Host και το SSL το οποίο θα αναπτυχθεί στην συνέχεια, γίνεται λήψη email όπως φαίνεται στην Εικόνα 15.

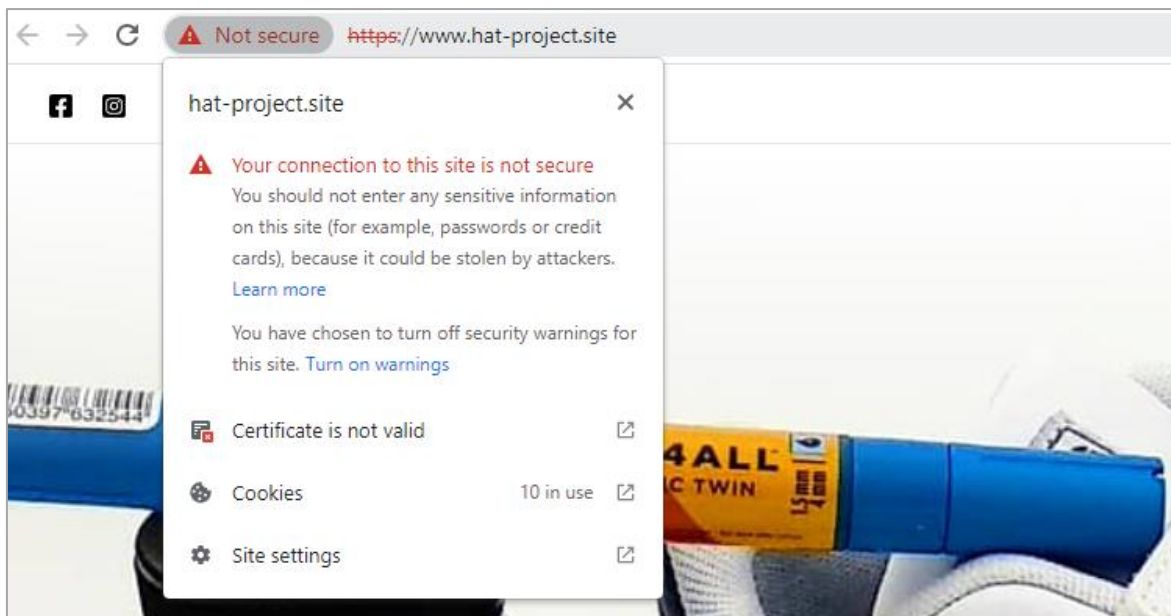
Οι εξυπηρετητές απαιτούν το Domain name για να καταλάβει ο πάροχος πως το όνομα συνδέεται με το πακέτο. Όταν συνδεθούμε στην σελίδα της εταιρείας, πηγαίνοντας στο πακέτο που θα το εμφανίσει πρέπει να πληκτρολογηθούν οι *nameservers* που εμφανίζονται ακριβώς από κάτω από το πακέτο και στη συνέχεια το παράθυρο που φαίνεται στην φωτογραφία. Στην συνέχεια, διασταυρώνονται τα στοιχεία που έρχονται στο email όπως αυτά της εικόνας 15. Επιλέγεται αποθήκευση, και μετά από περίπου 24 ώρες, διότι τόσο διαρκεί η διαδικασία, δίνεται η δυνατότητα να κατασκευαστεί η Ιστοσελίδα.

Από τα πρώτα βήματα που πραγματοποιήθηκαν για την σελίδα που θέλουμε να δημιουργήσουμε είναι η σύνδεση στον πάροχο και η επιλογή Plesk login. Όπως φαίνεται στην εικόνα το menu Plesk είναι το παραπάνω, στην καρτέλα Websites and Domains επιλέγεται το όνομα που έχει δοθεί στην αρχή για να ενοικιαστεί το Domain. Έπειτα επιλέγεται το όνομα στο menu Preferred Domain και σαφώς επιλέγουμε και το SSL/TSL Support όπως φαίνεται στο κάτω μέρος της Εικόνας 16.



Εικόνα 16- Plesk panel

Πρέπει να σημειωθεί ότι δεν έχει μισθωθεί SSL για τη συγκεκριμένη σελίδα, καθώς θεωρήθηκε ότι είναι αποκλειστικά για εκπαιδευτικούς σκοπούς και δεν τίθεται κανένα θέμα κλοπής στοιχείων, κωδικών και λογαριασμών.



Εικόνα 17 - SSL

Τι είναι το SSL

Το πιστοποιητικό SSL είναι ένα ήδος σύνδεσης πελάτη-web server έτσι ώστε να παρέχει πάνω από όλα ασφάλεια στην πλοήγηση του πελάτη και εμπιστοσύνη. Με το πιστοποιητικό SSL διασφαλίζεται η ασφαλή ανταλλαγή των δεδομένων μεταξύ τους. Στην εικόνα που βλέπετε, έτσι εμφανίζεται αν κάποιος Ιστότοπος ΔΕΝ έχει SSL

ενεργοποιημένο ή έχει λήξει για αυτό λοιπόν προσοχή σε τι σελίδες πραγματοποιείται τις συναλλαγές σας και την ανταλλαγή δεδομένων.

Τελειώνοντας με όλα τα παραπάνω προχωράμε στην εγκατάσταση του WordPress. Στο menu του plesk υπάρχει ένα εικονίδιο του WP, το πατάμε και μας οδηγεί στην επιλογή εγκατάστασης του WordPress. Η παρακάτω Εικόνα 17 δείχνει πως είναι μια εγκατάσταση του WordPress.

Προσοχή θα πρέπει να δοθεί διότι πρέπει στο path να χρησιμοποιηθεί το https μπροστά και στη συνέχεια το όνομα που έχουμε δώσει στο eshop μας. Οι επιλογές έπειτα είναι όπως φαίνονται στην Εικόνας 18, δημιουργώντας κωδικό για να δοθεί πρόσβαση στο Dashboard του WordPress, επιλέγεται Install και υπάρχει μια αναμονή έως ότου ολοκληρωθεί.

Random values will be generated if fields are left blank.

General

Installation path ://

Website title

Plugin/theme set

Website language

Version

WordPress Administrator

Username

Password

Email

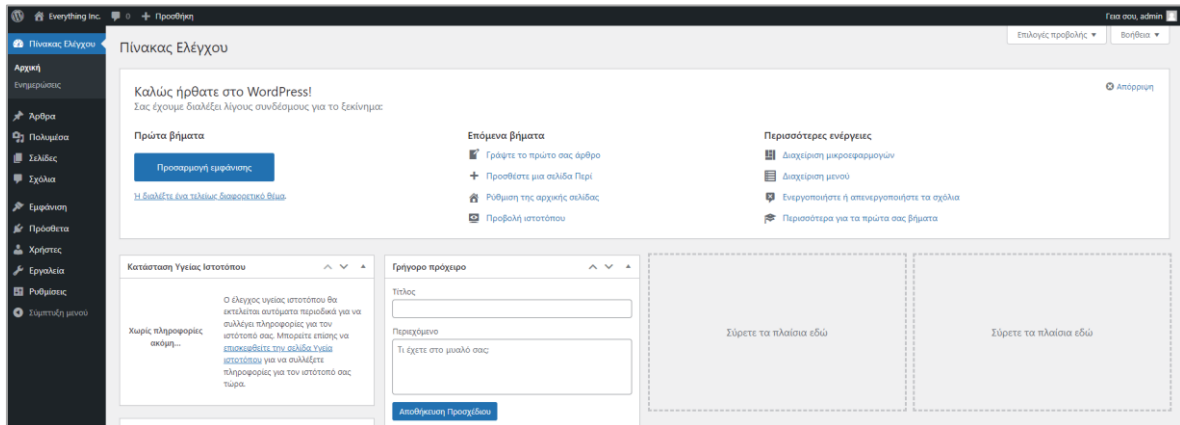
Database

Automatic Update Settings

Εικόνα 18 - WordPress Installation

6.1 WordPress Dashboard και Back End σχεδίαση

Στην Εικόνα 19 φαίνεται ο πίνακας ελέγχου του WordPress! Στην αριστερή πλευρά ή πλευρική στήλη φαίνεται το menu, από όπου πραγματοποιούνται όλες οι ενέργειες για την κατασκευή της σελίδας, του blog μας, ενός απλού Ιστότοπου και οτιδήποτε άλλο.



Εικόνα 19 - WordPress Dashboard

Το menu στα αριστερά περιλαμβάνει:

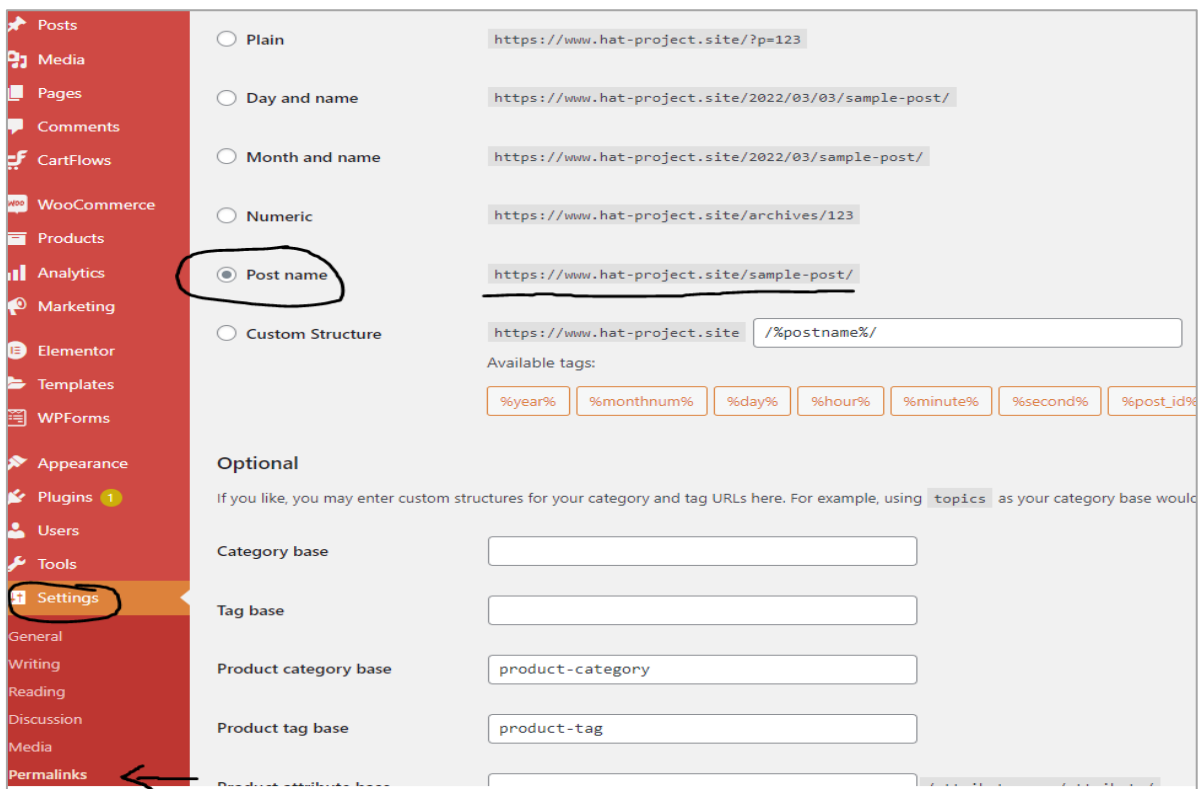
1. **Άρθρα**, όπου γίνεται η συγγραφή των άρθρων ή μιας στήλης
2. **Πολυμέσα**, η επεξεργασία βίντεο και κινούμενων εικόνων που θα θελήσει κάποιος να προσθέσει
3. **Σελίδες**, η επεξεργασία των σελίδων του site μας (Home, About, Contact pages)
4. **Σχόλια**
5. **Εμφάνιση**, εκεί γίνεται η επεξεργασία του περιεχομένου (content) του site μας
6. **Πρόσθετα**, τα λεγόμενα plugins που θα χρειαστούμε για τη δημιουργία (Elementor, WooCommerce)
7. **Εργαλεία**, διάφορα εργαλεία
8. **Ρυθμίσεις**, οι βασικές ρυθμίσεις του Ιστοχώρου μας

6.2 Γενικές Ρυθμίσεις και προσθήκη θέματος

Για την μετάβαση στις ρυθμίσεις του WordPress επιλέγεται αριστερά «ρυθμίσεις» ή «Settings», ανάλογα με την γλώσσα που έχει επιλέξει ο χρήστης. πηγαίνουμε στο General και από εκεί μπορούμε να αλλάξουμε τον τίτλο του Site, το Administration Email, τον

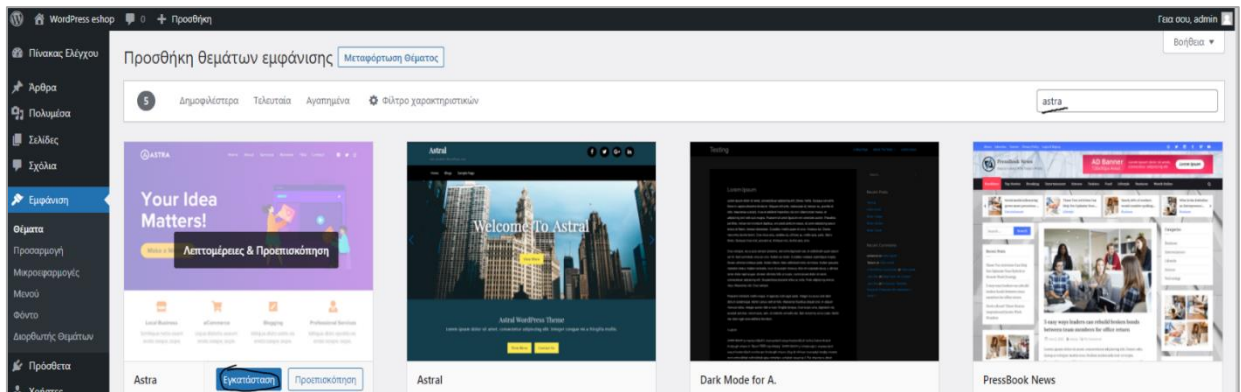
ρόλο κάθε νέου χρήστη (subscriber, author, customer, editor), την ημέρα που ξεκινάει η εβδομάδα, την γλώσσα του Site και άλλα πολλά.

Εν συνεχεία, μεταβαίνουμε στην επιλογή Settings → Permalinks που όπως θα δούμε και στην επόμενη Εικόνα 20, στα common settings επιλέγουμε το Post Name, δηλαδή όνομα άρθρου και αυτό γίνεται για να μην εμφανίζεται άσχημα η διεύθυνση της σελίδας



Εικόνα 20- General Settings and Permalinks

Όπως βρίσκεται κάποιος στο Dashboard μεταβαίνει στο menu εμφάνισης για να επιλεγεί το θέμα. Λόγω του γεγονότος ότι κατασκευάστηκε ηλεκτρονικό κατάστημα έγινε αντίστοιχα έρευνα για ένα ιδανικό θέμα που έχει όλα τα πρότυπα ενός eshop καταλήγοντας σε θέμα που ονομάζεται Astra. Στην Εικόνα 21 που φαίνεται παρακάτω δεξιά πάνω στην οθόνη κάνουμε αναζήτηση του θέματος, επιλέγεται το θέμα και στη συνέχεια κάνουμε εγκατάσταση.



Εικόνα 21- Προσθήκη θέματος

Καθώς πραγματοποιούνται αυτές οι επιλογές που γίνονται στο παρασκήνιο και πραγματοποιείτε η εγκατάσταση μόλις ολοκληρωθούν μπορούμε να αρχίσουμε να δημιουργούμε το site. Όταν εγκατασταθεί το θέμα, μεταβαίνουμε στην επιλογή «Customize» και εκκινείτε η διαδικασία σχεδίασης του eshop.

Okay, just one last step...

Tell us a little bit about yourself

Evangelos [Redacted]

Beginner Myself/My Company

I agree to receive your newsletters and accept the data privacy statement.

Advanced Options

Import Customizer Settings

Import Widgets

Import Content

Share Non-Sensitive Data

Submit & Build My Website →

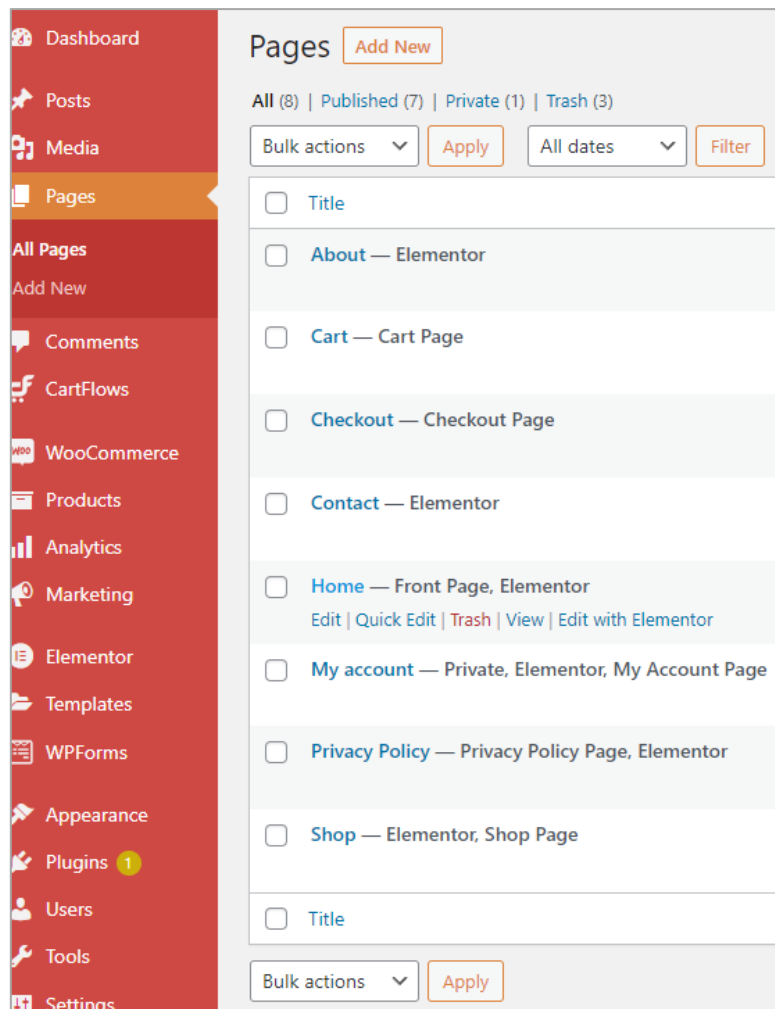
Εικόνα 22 - Εγκατάσταση και Import του θέματος

6.3 Δημιουργία Κατηγοριών Σελίδων και Menu

Αφού έχουν πραγματοποιηθεί όλα τα παραπάνω βήματα εκκινούνται οι διαδικασίες παραμετροποίησης με σκοπό να γίνουν οι αλλαγές και η επιθυμητή σχεδίαση. Δίνεται η δυνατότητα τώρα να ξεκινήσουμε με κάποια starter templates, κάτι σαν οδηγούς δηλαδή

που θα μας βοηθήσουν να κάνουμε τη σχεδίαση μας εύκολη και γρήγορη. Για το σκοπό αυτό μεταβαίνουμε στα Plugins → Elementor στην αναζήτηση και πραγματοποιούμε εγκατάσταση το εργαλείο που θα μας εμφανίσει πρώτο. Στο σημείο αυτό στην οθόνη μας και αφού το θέμα μας έχει εγκατασταθεί, επιλέγουμε την καρτέλα Appearance → Starter Template και τον Elementor, όπου είναι ένα πρόσθετο Page Builder στοιχείο και επιτρέπει να γίνονται οι όποιες αλλαγές εκείνη την στιγμή. Έχοντας λοιπόν όλα τα templates, επιλέγουμε το starter site που μας αρέσει περισσότερο και πατάμε **Import**. Ανακεφαλαιώνοντας, πρώτα πηγαίνουμε Customize μετά την εγκατάσταση του θέματος, έπειτα επιλέγεται Elementor, και κάποιο starter site που έχει έτοιμα και πατάμε Import. Ο Elementor Pro που είναι επί πληρωμή και έχει πολλές παραπάνω δυνατότητες από τον απλό, ωστόσο μπορεί κάποιος να κάνει ευκολά τα πάντα και με τον απλό Page Builder.

Στα αριστερά στον πίνακα ελέγχου υπάρχει η επιλογή Pages. Με την επιλογή Pages εμφανίζονται όλες οι σελίδες που υπάρχουν έτοιμες στο θέμα που εισήχθη εξ αρχής. Συνήθως αυτές οι starter pages είναι η σελίδες Αρχική (Home), Πολιτική ασφάλειας (Privacy Policy) και μια σελίδα υπόδειγμα (sample page) ως πρόχειρη σελίδα για να την τροποποιηθεί κατά το δοκούν. Με click στην επιλογή «Add New» δημιουργήθηκε η πρώτη σελίδα στο eshop. Μόλις δημιουργηθεί η σελίδα αυτή την επιλέγουμε για να ανοίξει ο Editor, μεταβαίνουμε στην επιλογή Customize που βρίσκεται στην κορυφή της οθόνης και ακολουθούμε τα εξής βήματα. Customize → Homepage Settings στην αριστερή πλευρική μας στήλη → επιλέγουμε το static page στο homepage displays → Homepage → Home έτσι καταφέρνουμε να έχουμε την Home Page ως αρχική σελίδα του eshop μας. τον ίδιο τρόπο φτιάχνουμε και τις υπόλοιπες σελίδες που φαίνονται όπως στην Εικόνα 23.

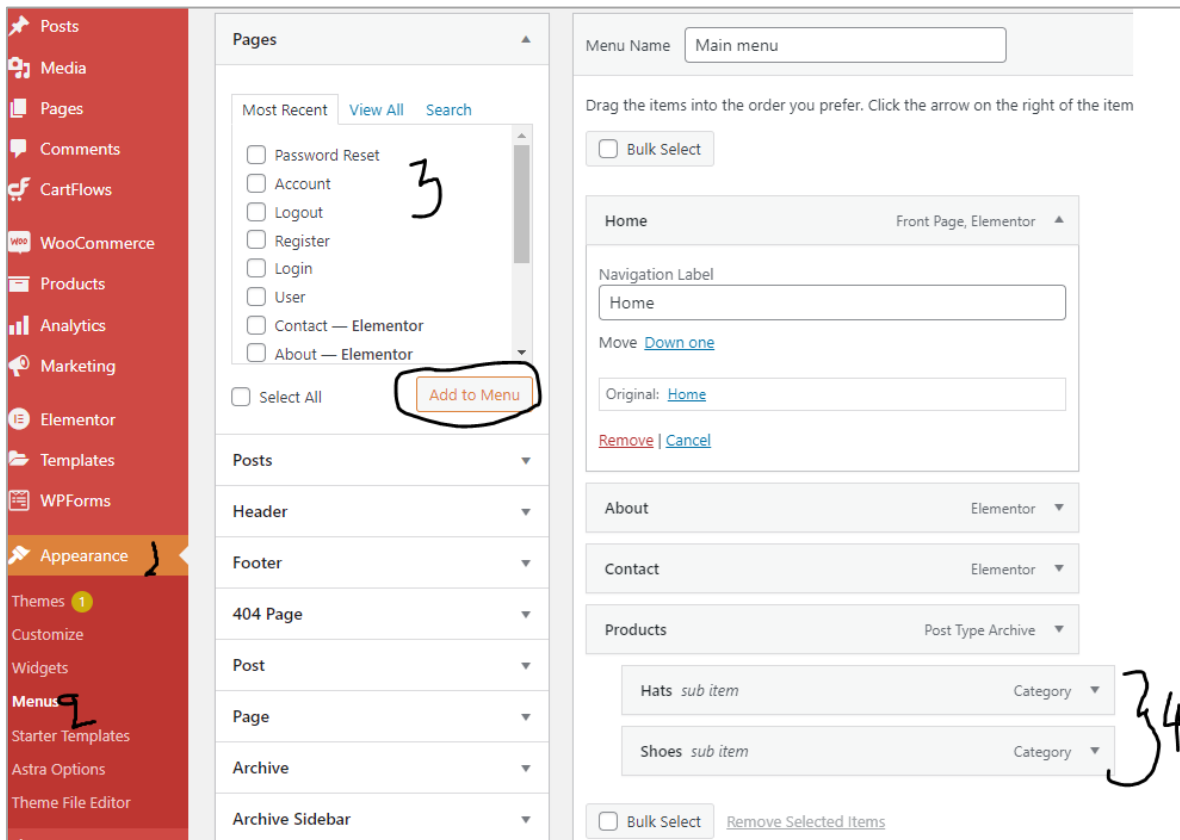


Εικόνα 23 - Σελίδες του θέματος μας

6.3.1 Δημιουργία σελίδων κατηγοριών και μενού

Για να προσθέσουμε νέα σελίδα στο πάνω μέρος της οθόνης μας όπως προαναφέρθηκε επιλέγουμε «**Add new**», ενώ στη συνέχεια δίνεται η δυνατότητα μέσω του Editor να αποδοθεί ο τίτλος που επιθυμούμε να έχει η σελίδα, ενώ για την καλύτερη δυνατή επεξεργασία επιλέγουμε «**edit with elementor**», όπου εκεί διαμορφώνεται το τελικό αποτέλεσμα. Στην Εικόνα 23 φαίνεται επίσης ότι καθώς δημιουργείται μια σελίδα, έστω και κενή, πηγαίνοντας στην σελίδα έχει τις επιλογές Edit, Quick Edit, Trash, View και Edit with elementor απευθείας. Από τις επιλογές αυτές πραγματοποιούνται όλες οι ενημερώσεις και οι αλλαγές. Κάθε σελίδα μπορεί να γίνει προσαρμοσμένη (custom) επεξεργασία, δηλαδή όλη να είναι σκέψη και δημιουργήμα του προγραμματιστή ή να γίνει μια ανακατάταξη σε ένα ήδη υπάρχων πρότυπο (template).

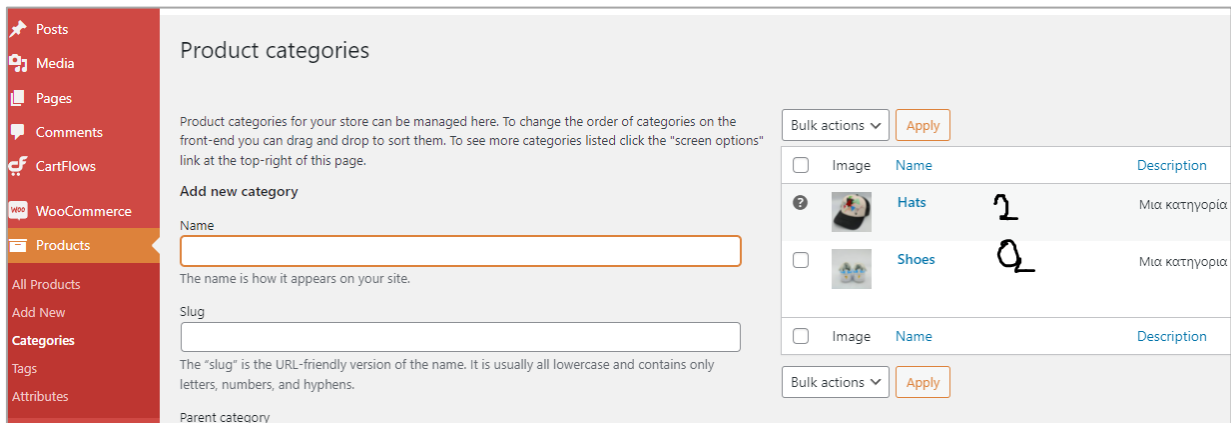
Είναι λίγα τα πράγματα που επεξεργάστηκαν χωρίς την βοήθεια του Elementor καθώς είναι ένα εργαλείο που δίνει πολλές επιλογές και δίνει μεγάλη ευελιξία στο χρήστη.



Εικόνα 24 - Create, Add, Remove Menus

Στο Eshop που δημιουργήθηκε χρησιμοποιήσα λίγα μενού για να είναι εύκολα κατανοητά και διαχειρίσιμα. Μεταβήκαμε λοιπόν αριστερά στο Dashboard ->Appearance -> Menus. Εκεί δημιουργήθηκαν όσα απαιτούνταν για την προσθήκη/διαγραφή των Menu που ήταν επιθυμητά για να εμφανίζονται στην αρχική σελίδα. Όπως φαίνεται έχει γραφεί ο αριθμός 3. Εκεί υπάρχουν όλα τα menu που προέρχονται από το θέμα μας. Επιλέγονται όσα χρειάζονται να εμφανιστούν και στη συνέχεια επιλέγεται «Add to menu». Στο δεξί μέρος της οθόνης και πάνω από τον αριθμό 4, είναι τα Menu που επελέγησαν και που θα εμφανίζονται στην αρχική σελίδα (Home Page) του E-shop μου. Όπως φαίνεται αν επιλεγεί μια από τις παραπάνω επιλογές, για παράδειγμα «Home», δύναται να διαγραφεί επιλέγονται στο Remove ή να αποσταλεί μια θέση παρακάτω πατώντας το «Down One». Μπορεί επίσης να αλλάξει η σειρά με Drag and Drop. Όπως φαίνεται έχει προσθέσει ο αριθμός 4 σε δύο κατηγορίες. Αυτές οι κατηγορίες δημιουργούνται από την καρτέλα

Products, πατώντας Products → Categories → και δημιουργούνται οι κατηγορίες ενδιαφέροντος για τα προϊόντα.



Εικόνα 25 - Κατηγορίες Προϊόντων

Καθώς δημιουργήθηκαν οι κατηγορίες, μεταβαίνουμε ξανά στο Menu και έπειτα στα αριστερά στο Sidebar στην καρτέλα «Product Categories» δίνεται η δυνατότητα να προσθέσουμε στο Menu τις δύο κατηγορίες που δημιουργήσαμε. Τις σέρνουμε «Drag» ελαφρώς δεξιά για να εισαχθούν μέσα στην καρτέλα Products όπως φαίνεται στην εικόνα 24 στον αριθμό 4 και στη συνέχεια Save menu κάτω δεξιά.

6.4 WooCommerce Tool

Για την ένταξη του ενός πρόσθετου με λειτουργικότητα ηλεκτρονικού καταστήματος, απαιτείται η μετάβαση αριστερά στην στήλη και στην καρτέλα Plugins, και η αναζήτηση του εργαλείου WooCommerce. Το WooCommerce είναι ένα από τα πιο χρήσιμα plugins καθώς από εκεί γίνονται όλες οι διαδικασίες για την αγοραπωλησία των προϊόντων μας, οι φόροι, οι πληρωμές, οι αποστολές και λοιπά. Όταν εγκατασταθεί το εργαλείο, επιλέγουμε το WooCommerce αριστερά στην στήλη στον πίνακα ελέγχου και μας κατευθύνει σε ένα νέο παράθυρο Home, όπου εκεί δηλώνονται οι πληροφορίες του καταστήματος, (π.χ. σε τι κατηγορία υπάγεται αυτό το κατάστημα, ρούχα, καλλυντικά κ.α., πόσα προϊόντα θα εισαχθούν ή για ποιον δημιουργείται το κατάστημα). Η επόμενη καρτέλα είναι η καρτέλα Orders, όπου εκεί έρχονται οι παραγγελίες που γίνονται από το Store. Ακολουθεί η καρτέλα Customers όπου φαίνονται οι πελάτες που κάνουν εγγραφή στο κατάστημα μας, η

καρτέλα Reports όπου έχουμε αναλυτικές αναφορές για τους πελάτες, τις παραγγελίες, τους φόρους και το απόθεμα.

The image shows the 'General' tab of the WooCommerce Settings. The 'Store Address' section is highlighted with handwritten numbers 1 through 6 and an arrow pointing to the right. The fields are: Address line 1 (Athinas 15), Address line 2 (empty), City (Ioannina), Country / State (Greece — Epirus), Postcode / ZIP (45500). Under 'General options', Selling location(s) is 'Sell to all countries', Shipping location(s) is 'Ship to all countries you sell to', and Default customer location is 'No location by default'.

Εικόνα 26 - WooCommerce Settings

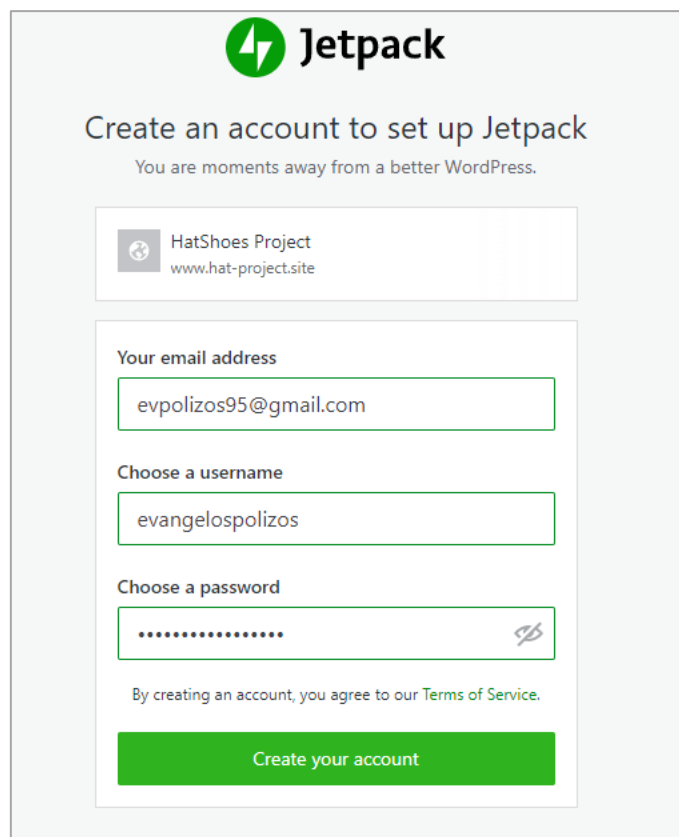
Η πρώτη καρτέλα αναφέρεται στις γενικές ρυθμίσεις του WooCommerce, όπως η διεύθυνση του καταστήματος, η πόλη, ο ταχυδρομικός κωδικός, οι περιοχές αποστολής των εμπορευμάτων, η ενεργοποίηση του υπολογισμού του φόρου και το νόμισμα επιλογής. Η δεύτερη καρτέλα αφορά τα προϊόντα και πως αυτά μετριοούνται (κιλά ή γραμμάρια), ποια θα είναι η σελίδα του καταστήματος μας (Shop page) και η επιλογή αναφορών για τα προϊόντα και από που ποιους χρήστες (subscribers ή μη).

6.4.1 Φόροι μεταφορικά και πληρωμές

Η Τρίτη, η τέταρτη και η πέμπτη καρτέλα αναφέρεται στους φόρους στα μεταφορικά και στις πληρωμές. Πιο συγκεκριμένα, στην καρτέλα Tax γίνονται όλες οι αλλαγές σχετικά με την φορολογία των προϊόντων. Ένας τρόπος, αρκετά χρονοβόρος, είναι μέσα από την καρτέλα Tax να γίνουν όλες οι αλλαγές στους φόρους χειροκίνητα. Εκεί εμφανίζεται ένα

μενού όπου μπορεί να επιλεγεί ο τρόπος εμφάνισης των τιμών στον πελάτη (δηλαδή με φόρο ή χωρίς), θα μπορεί να προσθέτει και να υπολογίζει τον φόρο ανάλογα με αυτά που πουλάει το κατάστημα. Για παράδειγμα, αν θέλουμε να πουλήσουμε ένα καπέλο πηγαίνουμε στην καρτέλα Tax → Standard Rates → βάζουμε τον συντελεστή 24% που είναι για τις αγοραπωλησίες εντός της χώρας μας, για το εξωτερικό έπρεπε να καταχωρούσαμε διαφορετικό συντελεστή, ανάλογα την χώρα που στέλνουμε, αλλά και να φτιάξουμε τυχόν μειωμένους (13%) ή καθόλου συντελεστές ΦΠΑ. Αυτό βέβαια είναι ανάλογα και με τα προϊόντα, μπορεί η φορολογία για τα ρούχα στη χώρα μας να είχε μειωμένο συντελεστή και όχι τον σταθερό 24%.

Για να γλιτώσουμε αυτή τη διαδικασία υπάρχει ένα Plugin που ονομάζεται **Jetpack** και που βοηθάει καθώς όλοι οι φόροι προθέτονται αυτόματα στο κατάστημα μας. Πηγαίνοντας λοιπόν στην καρτέλα με τα Plugins → Add new → WooCommerce Shipping & Tax → Activate και Install. Το επόμενο βήμα είναι να κάνουμε έναν λογαριασμό στο συγκεκριμένο εργαλείο όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 27 – Jetpack

Ενεργοποιώντας το Plugin, η καρτέλα Tax θα εμφανίζεται όπως στην εικόνα 28.

The screenshot shows the 'Tax' settings in WooCommerce. The 'Automated taxes' section is highlighted with a bracket, showing the 'Enable automated taxes' dropdown menu. Below it, there are radio buttons for 'Prices entered with tax' (Yes/No). Other settings include 'Calculate tax based on' (Customer shipping address), 'Shipping tax class' (Standard), 'Rounding' (unchecked), 'Additional tax classes' (Μειωμένος συντελεστής, Μηδενικός συντελεστής), 'Display prices in the shop' (Excluding tax), and 'Display prices during cart and checkout' (Excluding tax).

Εικόνα 28 - Taxes

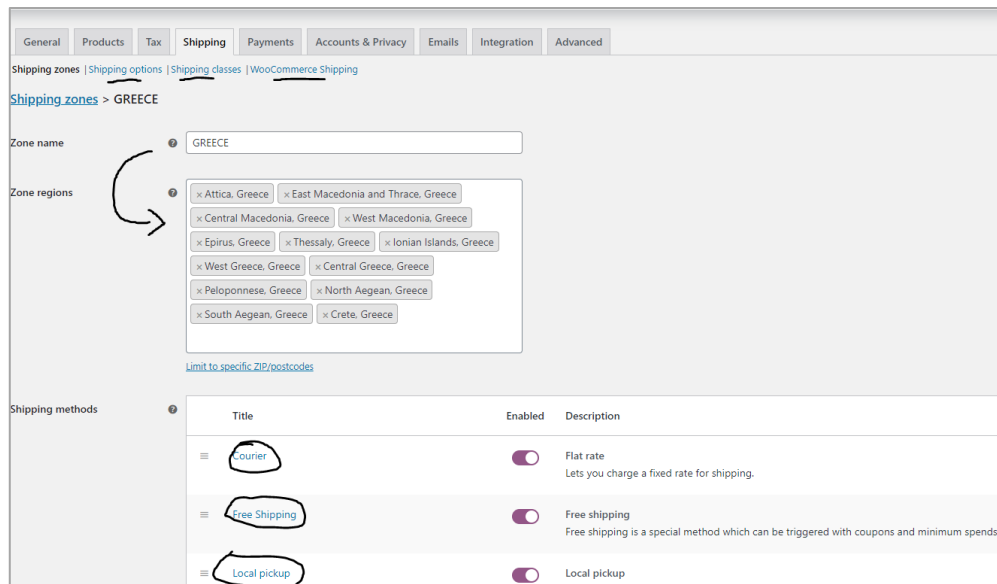
Τα υπογραμμισμένα είναι οι σύνδεσμοι που προαναφέρθηκαν, όπου μπορούν να αλλάξουν χειροκίνητα ανάλογα με την χώρα που πουλάμε και το προϊόν μας. Όπως βλέπουμε έχει δημιουργηθεί η επιλογή Automated taxes → Enable Automated Taxes και είναι μόνο αυτή η επιλογή διαθέσιμη καθώς το επιλέξαμε! Επίσης όλα γίνονται πλέον φορολογήσιμα και στο κυκλωμένο πλαίσιο υπάρχει μόνο η επιλογή Including Tax καθώς το έχει αυτόματα το Jetpack, έτσι οι τιμές στα προϊόντα εμφανίζονται χωρίς τον φόρο και αυτός θα εμφανιστεί στην τελική παραγγελία στο Checkout.

Όπως ακριβώς μεταβήκαμε στην καρτέλα Tax, στην καρτέλα Shipping για να προσθέτουμε τα μεταφορικά. Όπως φαίνονται στην Εικόνα 28 δύναται να δημιουργήσουμε από μόνοι μας τα μεταφορικά έξοδα των αποστολών. Συγκεκριμένα για το παρόν κατάστημα δημιουργήθηκαν εγγραφές για τις μεταφορές εντός της Ελλάδος, σε κάθε περιφέρεια, με τον ίδιο ΦΠΑ με 3 επιλογές.

1. Courier
2. Free Shipping

3. Παραλαβή από το κατάστημα

Πηγαίνοντας στην καρτέλα Shipping επιλέγεται Add Shipping Zone -> Ελλάδα -> τις περιφέρειες που επιθυμούμε -> Save. Στην συνέχεια επεξεργάζεται η ζώνη Ελλάδα που προσθέσαμε και εμφανίζονται όπως στην Εικόνα 29.



Εικόνα 29 - Shipping Zone Greece

Στην καρτέλα Shipping Zones έχουμε τοποθετήσει την χώρα που θέλουμε να στείλουμε τα πακέτα μας (μπορεί κάποιος να έβαζε Ευρώπη ή Αμερική) και κάτω στα κυκλωμένα πλαίσια βλέπουμε τις 3 επιλογές που αναφέρθηκαν προηγουμένως. Η κάθε μια από αυτές τις επιλογές έχει τα δικά της χαρακτηριστικά και συγκεκριμένα η επιλογή για Free μεταφορικά έχει ελάχιστο ποσό τα 100 ευρώ.

Free shipping Settings

Title

Free shipping requires...

Minimum order amount

Coupons discounts Apply minimum order rule before coupon discount

Εικόνα 30 - Free Shipping

Το ίδιο γίνεται σε οτιδήποτε άλλο προσθέσουμε σαν επιλογή, στην καρτέλα Courier δηλαδή υπάρχει χρέωση αποστολής 5 ευρώ, ενώ στην καρτέλα παραλαβή από το κατάστημα υπάρχει μηδενική χρέωση. Ένα άλλο πρόσθετο που συχνά χρησιμοποιείται λέγεται WooCommerce Advanced Free Shipping και δουλεύει ως εξής.

Δημιουργούμε τις ίδιες μεθόδους που δημιουργήσαμε και εξ αρχής, απλά στις επιλογές μπορούμε να επιλέξουμε δωρεάν αποστολές αν υπάρχουν 3 προϊόντα στο καλάθι μας ή αν υπάρχει παραγγελία με κόστος 150 και ανάλογα τα κιλά του πακέτου μας. Στην δική μας περίπτωση δεν χρησιμοποιήθηκε το εν λόγω Plugin, ώστε αφενός για να μην υπερφορτωθεί το ηλεκτρονικό κατάστημα, και αφετέρου διότι δεν χρειάζεται εφόσον το κάθε τι που γίνεται από τον πελάτη περνάει πρώτα την επίβλεψη του δημιουργού της σελίδας ή εκείνου που τη συντηρεί, έτσι ώστε να μην υπάρχει σύγχυση με μια παραγγελία σχετικά με τα μεταφορικά. Εάν κάποιος επιλέξει να κάνει μια παραγγελία και βάλει τις επιλογές που δώσαμε στις ζώνες μεταφοράς η παραγγελία του θα έχει την παρακάτω μορφή της εικόνας 31.

Checkout

Billing details

First name * Last name *

Company name (optional)

Country / Region *

Street address *

Apartment, suite, unit, etc. (optional)

Town / City *

State / County (optional)

Your order

Product	Subtotal
FearTheDeer	165,00 €
Hat × 3	
Subtotal	165,00 €
Shipping	5,00 €
VAT	39,60 €
Total	209,60 €

Shipping options:
1 Courier: 5,00 €
9 Free Shipping
3 Local pickup

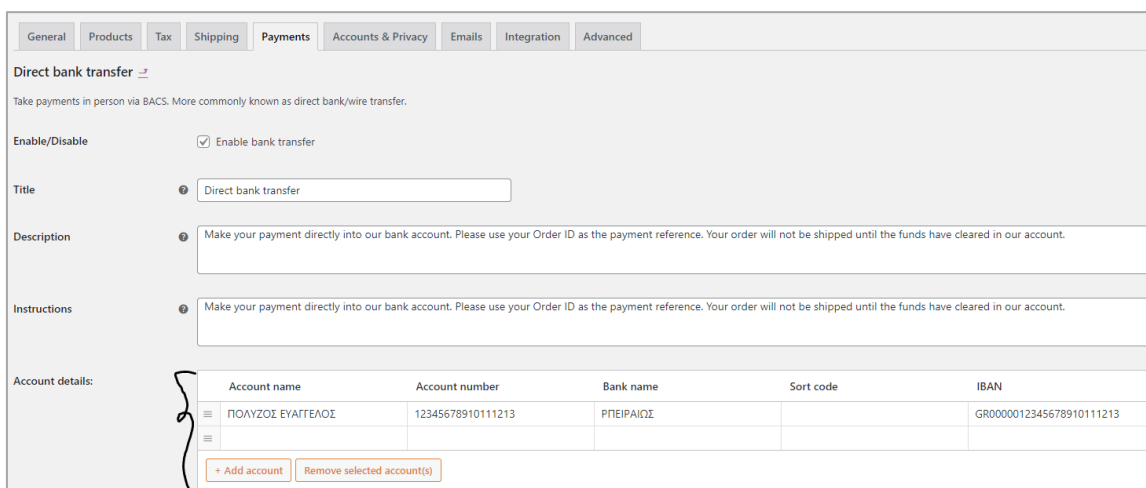
Εικόνα 31 - Checkout Shipping Methods

Όπως παρατηρούμε δίνονται οι πληροφορίες σε όλα τα πλαίσια που ζητούνται (τόπος διαμονής, ταχυδρομικός κώδικας, τηλέφωνο και διεύθυνση) και δεξιά βλέπουμε πως εμφάνισε τις επιλογές που έχουμε δημιουργήσει στο Shipping Zone. Το plugin που αναφέρθηκε παραπάνω μπορεί απλά να αλλάξει την επιλογή Free Shipping εάν η παραγγελία είναι πάνω από τα 100 ευρώ. Στην επιλογή Courier έχουμε 5 ευρώ μεταφορικά έξοδα αποστολής ενώ στην παραλαβή από το κατάστημα δεν έχουμε κάποια χρέωση.

Στην καρτέλα «πληρωμές» (Payments) πραγματοποιούνται οι ρυθμίσεις για τον τρόπο με τον οποίο ο πελάτης μπορεί να πληρώσει την παραγγελία του. Καθώς μπαίνουμε μέσα στην καρτέλα μας έχει αυτόματα 3 πληρωμές:

- 1 Μεταφορά μέσω τράπεζας
- 2 Πληρωμή με επιταγή
- 3 Αντικαταβολή

Για την περίπτωση μας επιλέχθηκε προσωρινά και για τις ανάγκες της εργασίας η πρώτη και η Τρίτη επιλογή.



The screenshot shows the 'Payments' tab in the WooCommerce admin interface. The 'Direct bank transfer' method is selected. The 'Enable/Disable' section has 'Enable bank transfer' checked. The 'Title' field contains 'Direct bank transfer'. The 'Description' and 'Instructions' fields both contain the text: 'Make your payment directly into our bank account. Please use your Order ID as the payment reference. Your order will not be shipped until the funds have cleared in our account.' The 'Account details' section is expanded, showing a table with one account entry:

Account name	Account number	Bank name	Sort code	IBAN
ΠΟΛΥΖΟΣ ΕΥΑΓΓΕΛΟΣ	12345678910111213	ΡΠΕΙΡΑΙΩΣ		GR0000012345678910111213

Below the table are buttons for '+ Add account' and 'Remove selected account(s)'.

Εικόνα 32 - Bank transfer Payment

Στην Εικόνα 32, φαίνεται πως γίνεται μια συναλλαγή μέσω μεταφοράς τράπεζας. Πρέπει να επιλεγεί «Enable bank transfer» και στη συνέχεια μεταβαίνουμε στη τράπεζα για να μεταφέρουμε τα χρήματα της παραγγελίας, χρησιμοποιώντας και το Order Id.

Ένας άλλος τρόπος είναι με αντικαταβολή, με τα ίδια βήματα, επιλέγοντας «Enable cash on delivery» προσφέροντας επιλογή με ποιες μεθόδους αποστολής «Shipping» θα γίνεται

η αντικαταβολή και όλοι όπως θα έχουμε κάνει είναι ή παραλαβή από το κατάστημα ή μέσω της Courier που θα μεταφέρει το δέμα.

6.5 Simple και Variable Products

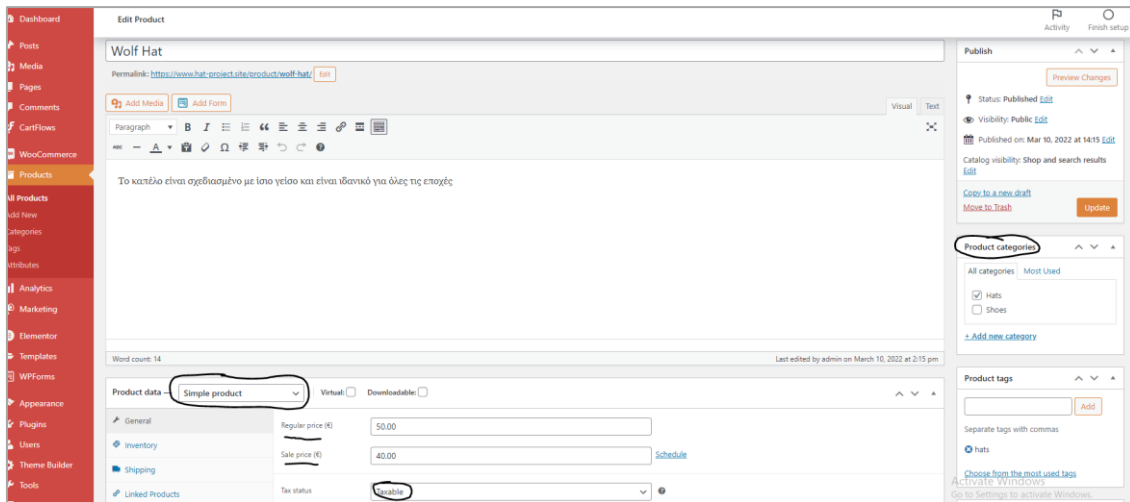
Στην ενότητα αυτή θα παρουσιαστεί πως καταχωρούνται προϊόντα στο κατάστημα. Στο eshop αυτή τη στιγμή έχουν προστεθεί αρκετά από τα προϊόντα. Όλη αυτή η διαδικασία γίνεται με έναν πολύ απλό τρόπο, καθώς πραγματοποιούνται από τον πίνακα ελέγχου του WordPress, μεταβαίνοντας στην αριστερά πλευρική στήλη, και επιλέγοντας την καρτέλα για τα προϊόντα (Products). Εκεί εμφανίζονται όλα τα προϊόντα που έχει το θέμα προ καταχωρημένα ή όσα νέα καταχωρήθηκαν για το κατάστημα. Τα προ καταχωρημένα διαγράφηκαν.

Οι κατηγορίες που απαιτούνται περισσότερο είναι:

1. Απλά Προϊόντα (Simple Product)
2. Μεταβλητά Προϊόντα (Variable Product)

Τα Απλά Προϊόντα όπως υποδηλώνεται από το όνομα είναι τα προϊόντα που δεν έχουν παραλλαγές (Variations) όπως είναι στην δική μας περίπτωση τα καπέλα. Είναι προϊόντα χωρίς διαφορετικά νούμερα και χωρίς μεγέθη που σημαίνει ότι υπάγονται σε αυτήν την κατηγορία. Από την άλλη, τα Μεταβλητά Προϊόντα είναι αυτά που μεταβάλλονται σε μέγεθος σε τιμή σε ποικιλία σε χρώματα και στην δική μας περίπτωση είναι τα παπούτσια.

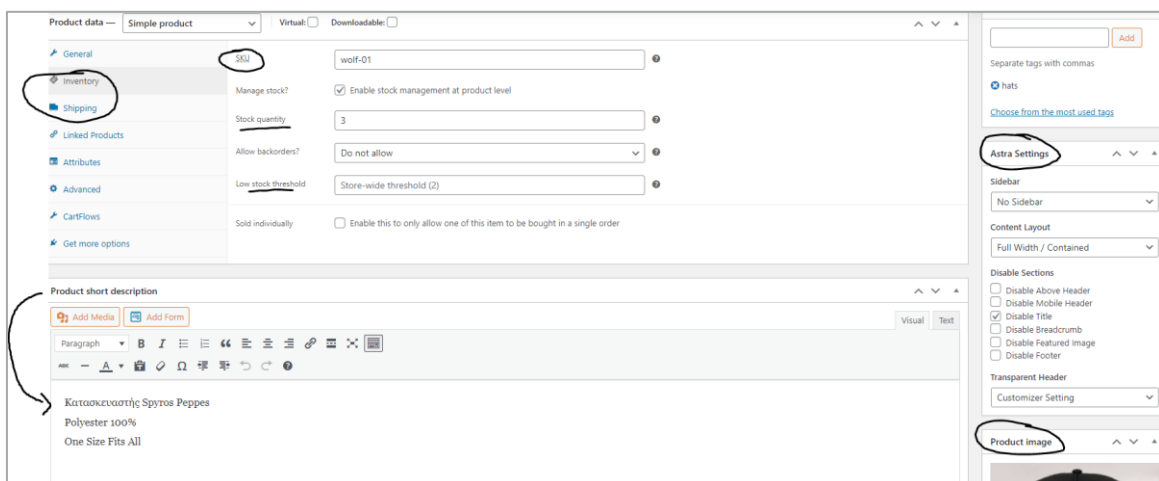
Έχοντας προσπελάσει την καρτέλα των προϊόντων, μεταβαίνουμε στην επιλογή «Add New» από όπου υπάρχει η δυνατότητα να καταχωρηθεί το πρώτο προϊόν. Πρώτα καταχωρείτε ο τίτλος του προϊόντος (Wolf Hat), στην συνέχεια μια περιγραφή όπως φαίνεται στο πλαίσιο της Εικόνας 33, ενώ έπειτα επιλέγουμε στο κυκλωμένο πλαίσιο και την επιλογή Simple Product.



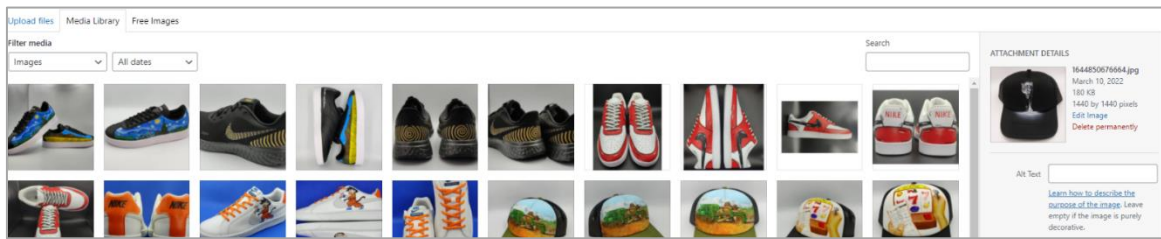
Εικόνα 33 - Add Simple Product

Στα πιο κάτω πεδία δύναται να καταχωρήσουμε ένα εύρος τιμών, ενώ τέλος στο μενού δεξιά επιλέγουμε την αντίστοιχη κατηγορία που ανήκει το προϊόν (καπέλα, παπούτσια).

Στην συνέχεια αλλάζουμε καρτέλα και προχωρά με στις δύο επόμενες Inventory και Shipping. Στην καρτέλα Inventory καταχωρούμε το πεδίο Stock Keeping Unit ή έναν κωδικό που θέλουμε εμείς να ξεχωρίσουμε τον αριθμό του προϊόντος. Παρατηρούμε ότι υπάρχει η δυνατότητα να επιλέξουμε το Stock Quantity και να εμφανίζει μήνυμα στον πελάτη για τα εναπομείναντα κομμάτια (Low Stock). Επιπροσθέτως, δεξιά έχουμε κάποιες επιλογές εμφάνισης του κάθε προϊόντος που τις έχει το θέμα που εγκαταστήσαμε, όπως μια μοναδική φωτογραφία του προϊόντος, ή πιο κάτω από την επιλογή αυτή δίδεται η επιλογή Add gallery images, όπου καταχωρούνται μαζικά φωτογραφίες για να έχει στη διάθεση του ο πελάτης.



Αυτές οι μεταφορτώσεις γίνονται πατώντας κάτω δεξιά στην οθόνη μας Add Product Image:



Εικόνα 34 - Μεταφόρτωση Εικόνας Προϊόντος

Η δεύτερη κατηγορία προϊόντων είναι τα Variable Products όπου ακολουθούμε ακριβώς την ίδια διαδικασία με κάποιες μικρές παραλλαγές. Πρώτα πηγαίνουμε στην καρτέλα Attributes → Add και η διαδικασία που ακολουθούμε είναι η εξής: α) δίνουμε ένα όνομα στο χαρακτηριστικό μας, (π.χ. Χρώμα), β) δίνουμε τιμές που χωρίζονται μεταξύ τους με «|» και γ) πατάμε SAVE. Έπειτα, μεταβαίνουμε στην άλλη καρτέλα Variations (εδώ υπάρχουν οι τιμές που καθορίστηκαν από το χαρακτηριστικό «SIZE» αλλά και οι τιμές από το χαρακτηριστικό «COLOR»).

Εικόνα 36 - Πρόσθεση Χαρακτηριστικών σε Variable product

Product data — Variable product	
General	Default Form Values: No default SIZE...
Inventory	Add variation
Shipping	
Linked Products	
Attributes	
Variations	#1863 40
Advanced	#1864 41
CartFlows	#1865 44
Save changes Cancel	

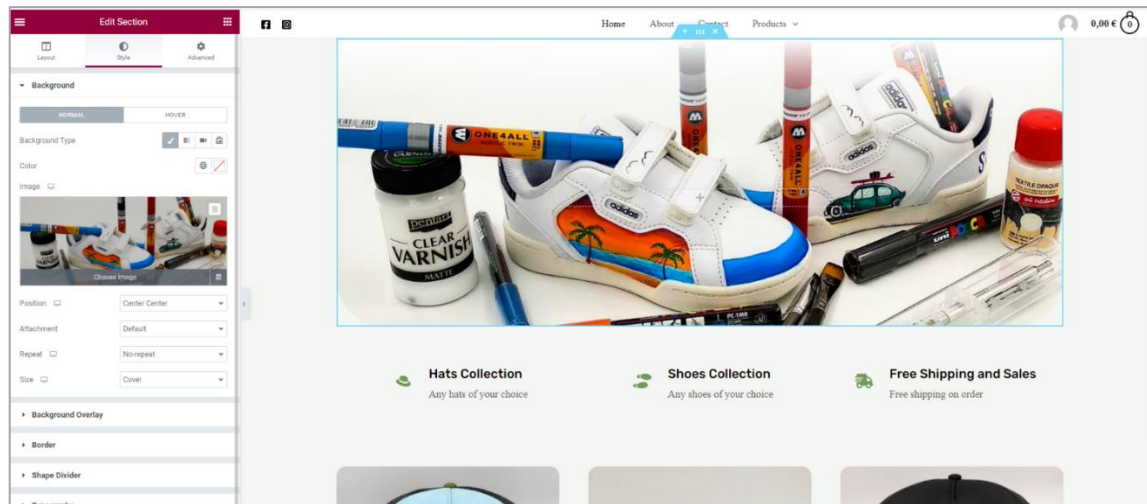
Εικόνα 37 – Παραλλαγές του προϊόντος

6.6 Elementor Theme Builder και WP Forms

Ένα από τα πιο διαδεδομένα builder για τις ιστοσελίδες είναι το Elementor Tool. Σαφώς δεν είναι το μοναδικό, καθώς υπάρχουν και άλλα πρόσθετα με παρόμοια λειτουργικότητα, όπως το Thrive Theme ή το Divi. Η παρούσα εργασία έκανε χρήση του Elementor για όλες τις μορφοποιήσεις του καταστήματος. Οι παραμετροποιήσεις που απαιτείται κάποιος να πραγματοποιήσει σε ένα Site μέσω αυτού του εργαλείου είναι αρκετά εύκολες και δεν είναι απαραίτητο να υπάρχει γνώση κάποιας γλώσσας προγραμματισμού (αν και υπάρχει και αυτή η επιλογή για κάποιον πιο απαιτητικό χρήστη!).

Οι παραμετροποιήσεις αυτές μπορούν να πραγματοποιηθούν είτε στα αρχικά στάδια της υλοποίησης, είτε στο τέλος και αφού έχει προστεθεί όλη η λειτουργικότητα και έχουν καταχωρηθεί όλα τα δεδομένα. Ενδεικτικά η διαδικασία που πρέπει να ακολουθηθεί έχει ως εξής: Ξεκινώντας από τον πίνακα ελέγχου / Dashboard στην αριστερή στήλη και στην καρτέλα Pages και μεταβαίνουμε στην σελίδα Home. Επιλέγοντας αυτό θα κατευθυνθούμε στην σελίδα του WordPress Builder. Το WordPress Builder είναι ένας απλός Editor όπου εκεί υπάρχει η επιλογή Edit with Elementor. Επιλέγουμε αυτή την επιλογή και μεταφερόμαστε στο θέμα που έγινε εισαγωγή (import) στην ενότητα 6.3.

Οι παρακάτω φωτογραφίες είναι αποτέλεσμα όλων των αλλαγών που έγιναν ώστε να έρθει το Eshop μας στην τελική του μορφή.



Εικόνας 38: Η τελική μορφή του eshop

New Projects Coming Soon

Watch straps , T-Shirts , lighters , e-Cigarette battery

Need help in choosing the right products?

[Ask For Help](#)

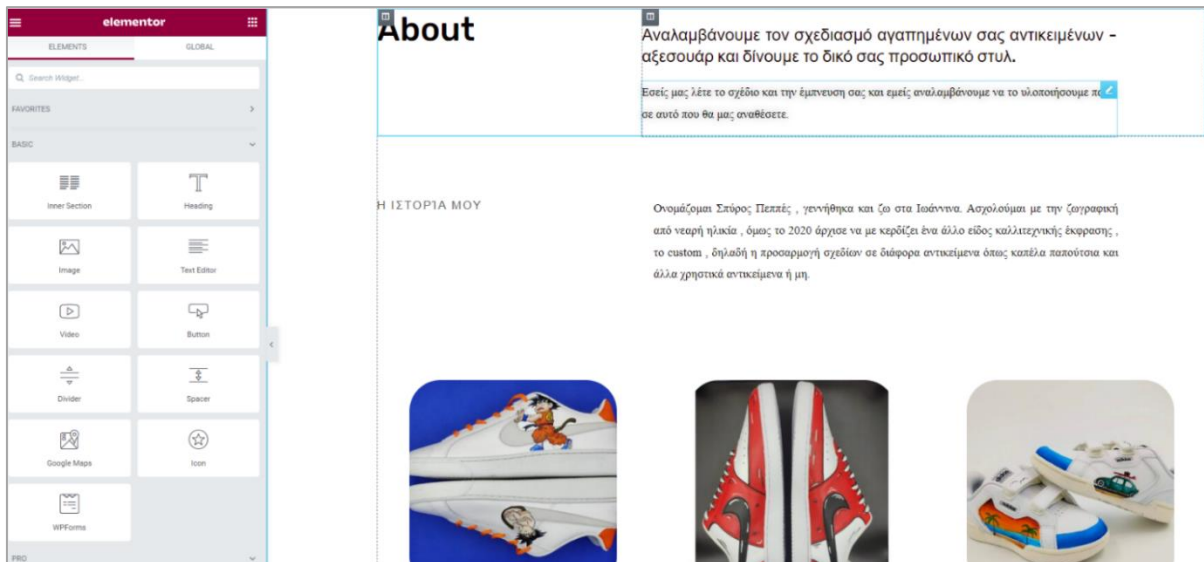
Είμαστε στην διάθεση σας για οτιδήποτε προκύψει. Καλέστε μας ή επικοινωνήστε μαζί μας μέσω της φόρμας επικοινωνίας που βρίσκετε στην καρτέλα Contact για:

- Αλλαγές προϊόντων
- Λάθος παράδοση ή παραλαβή
- Πρόβλημα που θα προκύψει με τις οικονομικές συναλλαγές

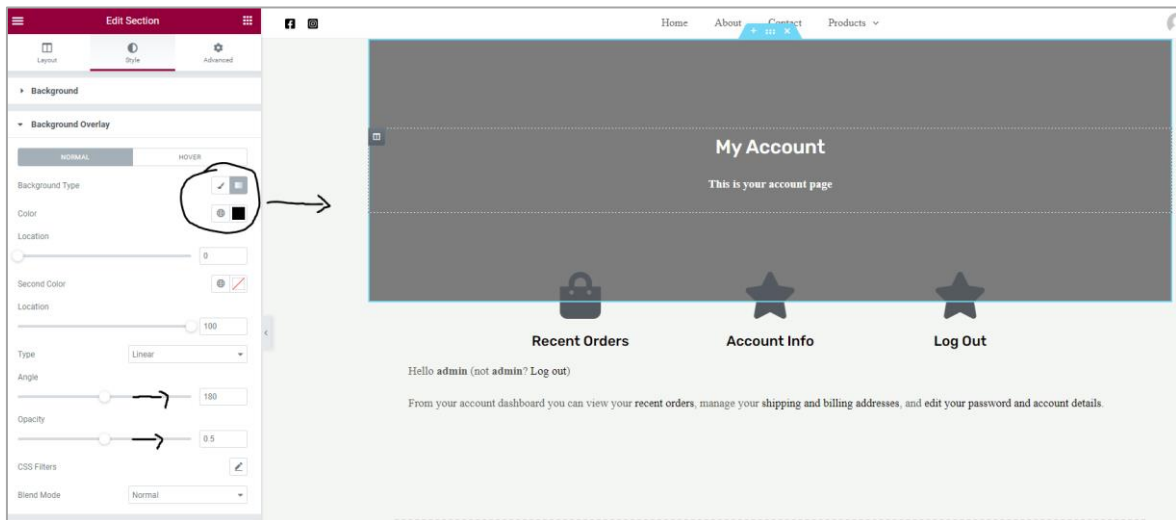
For how to clean your products see the steps

1. Πλένετε στο χέρι και σε χαμηλή θερμοκρασία
2. Τρίβετε απαλά με ένα σφουγγάρι εάν υπάρχει κάποιος λεκές

Εικόνας 39: Ανακοίνωση για νέα προτζεκτς



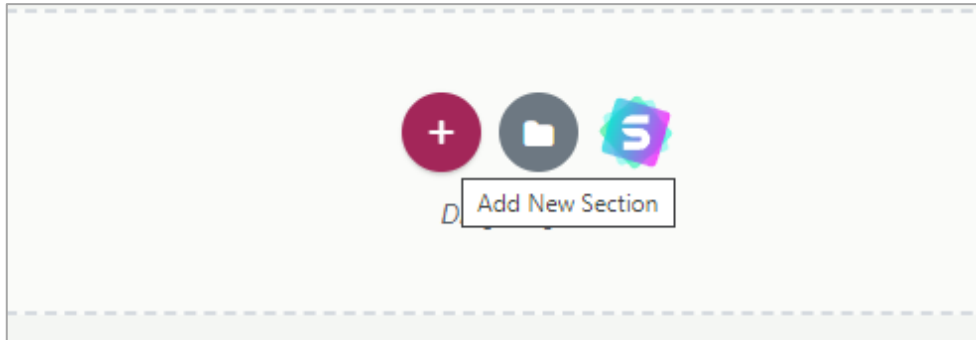
Εικόνας 40: Η σελίδα για τους συντελεστές



Εικόνας 41: Η σελίδα του χρήστη

Το ηλεκτρονικό κατάστημα μπορεί κανείς να προσπελάσει στον ιστό πληκτρολογώντας www.hat-project.site. Αν κατέχει δικαιώματα υπερχρήστη μπορεί να συνεχίσει να επεξεργάζεται την μορφή του. Όταν ο χρήστη εισέλθει στην σελίδα Home Page και επιλέξει «Edit with Elementor», αριστερά θα εμφανιστεί μια μπάρα επιλογών όπου πραγματοποιούνται όλες οι ενέργειες για την επεξεργασία της σελίδας μας.

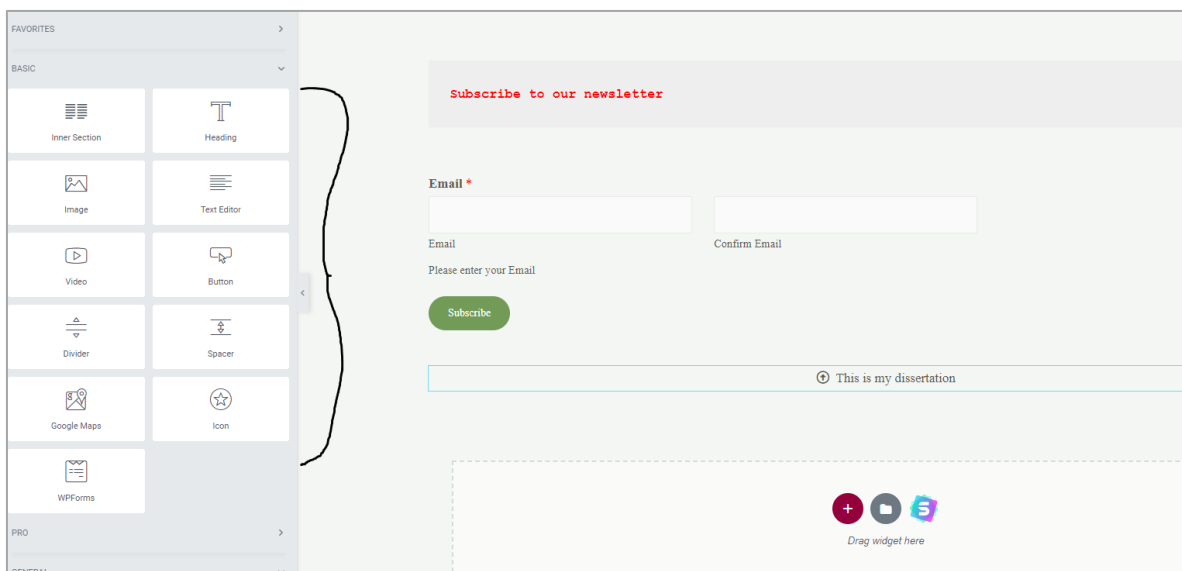
Πιο συγκεκριμένα, υπάρχει ένα πλαίσιο που μας δίνει την δυνατότητα να επιλέξουμε την δομή της κάθε ενότητας του Page, επιλέγοντας στο + →Select structure και ανάλογα με τον αριθμό των στηλών επιθυμούμε να υπάρχουν σε κάθε ενότητα.



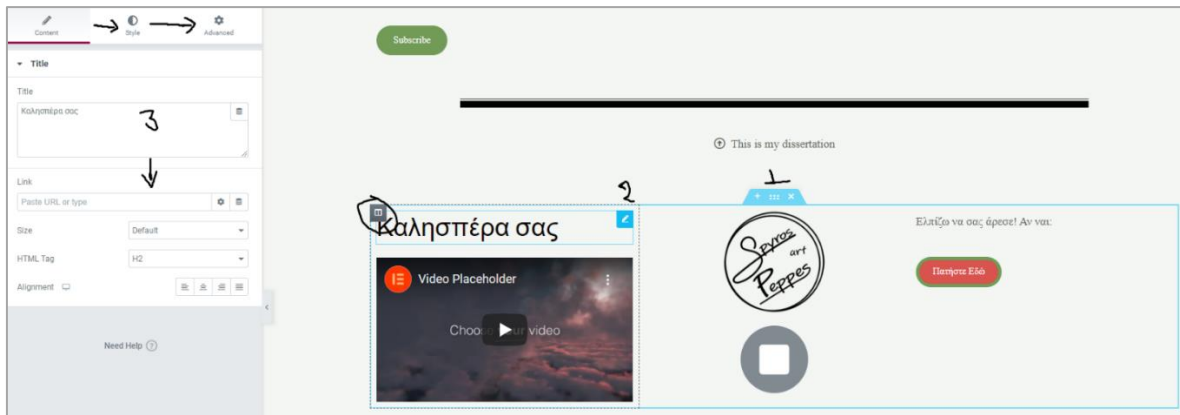
Εικόνας 42: Εισαγωγή Widget

Μόλις επιλέξουμε την δομή μας (αριθμός στηλών) ακριβώς στα αριστερά υπάρχουν οι επιλογές του σχεδιασμού. Στο πλαίσιο της εργασίας ασχοληθήκαμε μόνο μόνο με τις καρτέλες General και Basic καθώς οι υπόλοιπες απαιτούν το Elementor Pro, το οποίο προσφέρεται επί πληρωμή σε επαγγελματίες σχεδιαστές σελίδων.

Εάν θέλουμε να βάλουμε τρεις στήλες στην αρχική σελίδα μας, επιλέγουμε Structure 3 και στην συνέχεια μέσα από τις καρτέλες Basic ή General βρίσκουμε όλα τα δωρεάν εργαλεία που μπορούμε να χρησιμοποιήσουμε.



Εικόνας 42: Βασικές Ρυθμίσεις



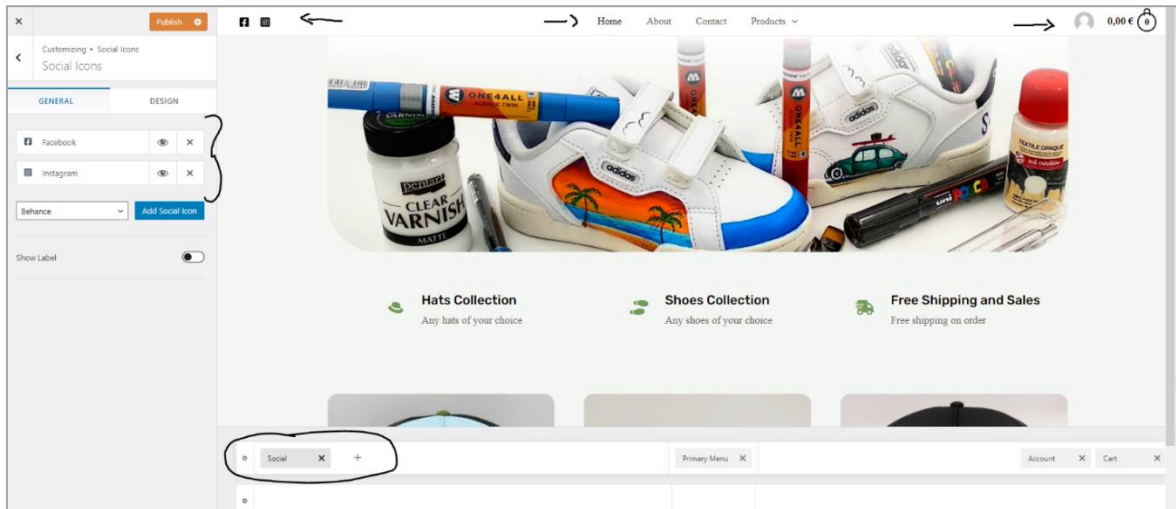
Εικόνας 43: διαχείριση εμφάνισης

Δεδομένου ότι έχουμε επιλέξει τρεις στήλες για επεξεργασία, πηγαίνουμε αριστερά στην οθόνη μας και επιλέγουμε περιοχόμενο της αρεσκίας μας σε κάθε μια από τις στήλες με σκοπό να υπάρχει ένα καλό αισθητικό αποτέλεσμα. Η επεξεργασία φαίνεται στην εικόνα 43 και έγινε με επιλογή δομής τριών στηλών και με έναν απλό επεξεργαστή κειμένου:

- Στον σχεδιασμένο αριθμό 1 έχει τρεις επιλογές. Από αριστερά προς τα δεξιά έχουμε την επιλογή Add section, Edit section και delete section. Πατώντας Add και Delete προστίθεται ή διαγράφεται ενότητα αντίστοιχα. Η επιλογή Edit την αναφέρεται σε ολόκληρη την ενότητα και όχι στα επιμέρους στοιχεία που έχουν καταχωρηθεί μέσα σε αυτή. Για τα επιμέρους στοιχεία και για το κάθε ένα από αυτά ξεχωριστά αναφέρεται ο αριθμός 2. Ο αριθμός 2 δηλαδή έχει ένα μπλε σύνδεσμο για να είναι δυνατή η επεξεργασία του κειμένου «καλησπέρα σας». Αντίστοιχοι σύνδεσμοι έχουν και τα υπόλοιπα περιεχόμενα όπως το video, το λογότυπο κ.ο.κ.
- Ο αριθμός 3 αναφέρετε στην επεξεργασία της επικεφαλίδας του κειμένου «καλησπέρα σας» όπου τοποθετήθηκε από την καρτέλα Basic → Heading με drag&drop. Με τον ίδιο τρόπο δημιουργήθηκαν οι επικεφαλίδες της αρχικής σελίδας. Επίσης στα αριστερά στο πλαίσιο με αριθμό 3 πραγματοποιούνται όλες οι αλλαγές στοίχισης (στο κέντρο ή δεξιά), οι αλλαγές χρωμάτων στην καρτέλα Style που συμβολίζεται με το βέλος, αλλά και κάποια πιθανή κίνηση του κειμένου μας που γίνεται από την καρτέλα Advanced.

6.6.1 Κεφαλίδα και Υποσέλιδο και Responsive Design

Ένα σημαντικό κομμάτι στην δημιουργία ιστοτόπων είναι οι κεφαλίδες και τα υποσέλιδα. Αυτά μπορούμε να τα επεξεργαστούμε μέσω του Customize του θέματος. Μεταβαίνοντας λοιπόν στον πίνακα ελέγχου → Appearance → Customize εμφανίζεται η σελίδα που έχει μέχρι στιγμής δημιουργηθεί, ενώ στα αριστερά υπάρχουν οι επιλογές, Header Builder και Footer Builder.



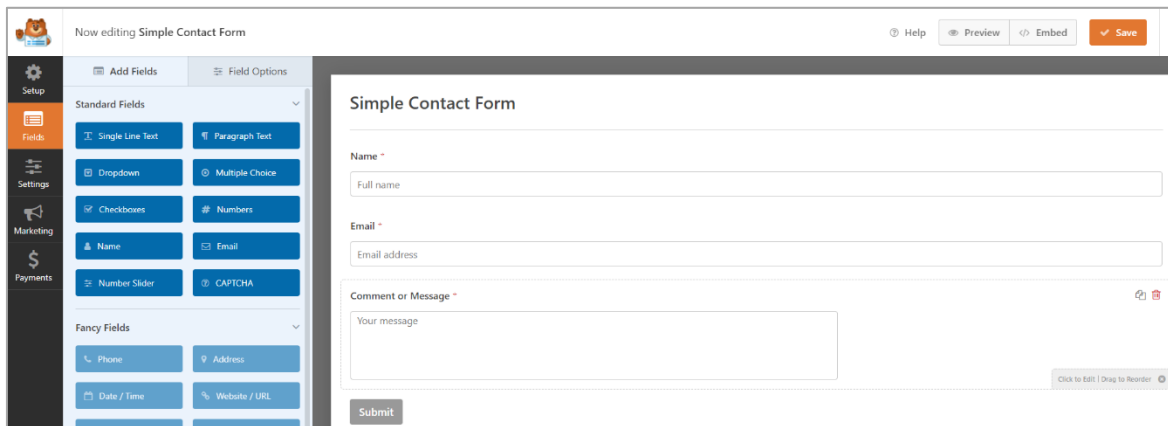
Εικόνας 44: Διαχειριστής κεφαλίδας

Όπως διακρίνουμε, η επικεφαλίδα μας αποτελείται από τα social media icons, τις καρτέλες Home, About, Contact, Products και δεξιά το καλάθι των αγορών και ο λογαριασμός μας. Όλα αυτά προστίθενται από το κάτω μέρος της οθόνης στο κυκλωμένο πλαίσιο και στο + και έχουμε και διάφορες άλλες επιλογές να προσθέσουμε όπως πλευρικές στήλες, κουμπί αναζήτησης ή διάφορα άλλα κουμπιά που θα μεταφέρουν τον επισκέπτη σε μια άλλη καρτέλα ή σε έναν σύνδεσμο. Επίσης, μπορούμε από τα αριστερά όπως είδαμε και στα προηγούμενα να σχεδιάσουμε εκ νέου τα default κουμπιά, για παράδειγμα τα social media μπορούμε να τους μεταβάλουμε το σχήμα, να αλλάξουμε το εικονίδιο καθώς έχει το θέμα μας μια γκάμα επιλογών αλλά και να αλλάξουμε τα χρώματα.

Με τον ίδιο ακριβώς τρόπο δημιουργείται και το υποσέλιδο (Footer). Το υποσέλιδο δηλαδή που φαίνεται στο τέλος του καταστήματος, έχοντας εξίσου διάφορες επιλογές και widgets που θα δώσουν έμφαση και ζωντάνια σε μια σελίδα. Στο πλαίσιο της εργασία εκεί τοποθετήθηκε το copyright του καταστήματος με ένα επιπλέον κείμενο.

6.6.2 WP Forms

Κλείνοντας, θα ήταν σκόπιμο να περιγράψει ένα επιπλέον πρόσθετο που χρησιμοποιήθηκε, το πρόσθετο «WP forms lite». Το Plugin αυτό μας δίνει την ευκαιρία να δημιουργήσουμε φόρμες επικοινωνίας και επιλογές Subscribe ή Newsletter.



Εικόνας 45: Φόρμα επικοινωνίας

Το WPForms βρίσκεται στον πίνακα ελέγχου στα αριστερά, το οποίο επιλέγοντας το εμφανίζεται όπως ακριβώς στην εικόνα 45. Τα μπλε κουμπιά στα αριστερά μπορούμε «σέρνοντας» τα ή επιλέγοντας τα να τα τοποθετήσουμε στα δεξιά και να σχηματιστεί μια φόρμα. Στην περίπτωση την δική μας, χρησιμοποιήθηκε μια κλασική φόρμα επικοινωνίας με πεδία το όνομα του χρήστη, ένα email και ένα μήνυμα που επιθυμεί ο χρήστης να αποσταλεί στον ιδιοκτήτη ή τον admin της σελίδας. Το κάθε πεδίο αν το επιλέξουμε με το ποντίκι και μεταβούμε στην επιλογή Field Options μπορούμε να το επεξεργαστούμε κατά το δοκούν. Ενδεικτικά δύναται να δημιουργηθούν υποχρεωτικά κάποια πεδία (επισημαίνονται με αστερίσκο), να βάλουμε confirmation (δηλαδή να πληκτρολογεί ο χρήστης το email του δύο φορές ώστε να γίνεται επιβεβαίωση) ή να αλλάξει τις διαστάσεις στα κείμενα του πεδίου. Με την εγκατάσταση του WPForms, μπορούμε να τοποθετήσουμε σε ένα section με τον Elementor την φόρμα στην σελίδα μας ή και στην αρχική σελίδα. Πολλές ιστοσελίδες έχουν στην αρχική τους την σελίδα επικοινωνίας. Με τον ίδιο ακριβώς τρόπο δημιουργήθηκε η καρτέλα Contact στην φόρμα επικοινωνίας αλλά και στην αρχική σελίδα το newsletter.

Κεφάλαιο 7^ο

7. Συμπεράσματα

Συμπερασματικά, στο πλαίσιο της παρούσας εργασίας έγινε αντιληπτό ότι από το 1991 και έπειτα η εξέλιξη του διαδικτύου είναι αλματώδης. Η βελτίωση στις μηχανές αναζήτησης έφερε μαζί με τον ανταγωνισμό και την περαιτέρω εξέλιξη των σελίδων και βέβαια και των ηλεκτρονικών καταστημάτων. Δημιουργήθηκαν πολλές θέσεις εργασίας πάνω στην επιστήμη της πληροφορικής και την επικοινωνία με την χρήση των ιστολογίων και των ιστοτόπων. Ο θέσεις εργασίας προγραμματιστών προσκηνίου και παρασκηνίου είναι πλέον περιζήτητες σε παγκόσμια κλίμακα. Επιπλέον, στον τομέα των γραφικών και των γραφιστικών συναντάμε στις μέρες μας διάφορες σχεδιάσεις όπως η UI και UX και σιγά σιγά οι αμοιβές όλο και αυξάνονται.

Η ανάγκη για την διερεύνηση πρωτοκόλλων και η συνταγή της επιτυχίας των API's έχουν εκτοξεύσει τις δυνατότητες των σχεδιαστών, μέσω των γλωσσών προγραμματισμού όπως η JavaScript, η Python αλλά και η style CSS να βελτιώνουν και να εξελίσσουν συνεχώς τα framework τους. Επιπροσθέτως, η ταχύτατη ανάπτυξη των CMS συστημάτων όπως το WordPress, δίνει την δυνατότητα σε εργασιακές εξελίξεις για κάποιους ανθρώπους που δεν έχουν τη δυνατότητα να σπουδάσουν, καθώς δεν είναι εργαλεία που χρήζουν ιδιαίτερη μαθησιακή γνώση. Σκοπός της εργασίας μου ήταν να μελετηθεί η εξέλιξη του σχεδιασμού των Ιστοσελίδων από το μακρινό 1990 μέχρι και σήμερα, να εμπεδωθεί η ανάγκη να διαχωρίζονται οι σχεδιαστές σε διαφορετικές κατηγορίες, διότι ο όγκος και ο φόρτος εργασίας είναι μεγάλος. Επίσης να διερευνηθεί το γεγονός ότι υπάρχουν πολλές γλώσσες προγραμματισμού με τις οποίες για την υλοποίηση ηλεκτρονικών καταστημάτων. Το WordPress είναι ένα εργαλείο εύκολο στη χρήση του με όλες τις αλλαγές του να γίνονται εκείνη τη στιγμή, σαν να χρησιμοποιεί κάποιος έναν word editor, χωρίς κώδικες.

Βιβλιογραφία

- ALfred, χ.χ. *Dot Com Infoway (dci)*. [Ηλεκτρονικό]
Available at: <https://www.dotcominfoway.com/blog/web-design-and-development-guide/>
- Anon., 2019. *UX Planet*. [Ηλεκτρονικό]
Available at: <https://uxplanet.org/what-is-ui-vs-ux-design-and-the-difference-d9113f6612de>
- Anon., 2020. *What Does Web Development Mean?*. [Online]
Available at: <https://www.techopedia.com/definition/23889/web-development>
- Anon., 2021. *Drupal*. [Ηλεκτρονικό]
Available at: https://www.drupal.org/docs/user_guide/en/understanding-drupal.html
- Anon., 2021. *InterviewBit*. [Ηλεκτρονικό]
Available at: <https://www.interviewbit.com/blog/backend-developer-skills/>
- Anon., 2021. *MDN Web Docs*. [Ηλεκτρονικό]
Available at: https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics
- Anon., χ.χ. *DECORation*. [Ηλεκτρονικό]
Available at: <https://decoration.gr/ta-mistika-tis-diakosmisis-o-kanonas-60-30-10/>
- Anon., χ.χ. *drupalizeme*. [Ηλεκτρονικό]
Available at: <https://drupalize.me/what-is-drupal>
- Anon., χ.χ. *RocketTheme*. [Ηλεκτρονικό]
Available at: <https://rockettheme.com/docs/joomla/platform>
- Anon., χ.χ. *Web Design History Timeline*. [Ηλεκτρονικό]
Available at: <https://www.webdesignmuseum.org/web-design-history>
- B, A., 2021. *HOSTINGER TUTORIALS*. [Ηλεκτρονικό]
Available at: <https://www.hostinger.com/tutorials/what-is-css>
- Castellani, S., 2020. *Axway*. [Ηλεκτρονικό]
Available at: <https://blog.axway.com/amplify-products/api-management/different-types-apis>
- Chahal, P., 2021. *TemplateToaster*. [Ηλεκτρονικό]
Available at: <https://blog.template toaster.com/what-is-joomla/>
- Fitzgerald, A., χ.χ. *Hubspot*. [Ηλεκτρονικό]
Available at: <https://blog.hubspot.com/blog/tabid/6307/bid/7969/what-is-a-cms-and-why-should-you-care.aspx>
- Galov, N., 2021. *Hosting tribunal*. [Ηλεκτρονικό]
Available at: <https://hostingtribunal.com/blog/joomla-statistics/>
- Hartman, J., 2021. *Guru99*. [Ηλεκτρονικό]
Available at: <https://www.guru99.com/introduction-to-javascript.html>

- ibis, χ.χ. *The Importance of Good Web Design*. [Ηλεκτρονικό]
Available at: <https://ibisstudio.com/the-importance-of-good-web-design/>
- Jackson, B., 2021. *kinsta*. [Ηλεκτρονικό]
Available at: <https://kinsta.com/blog/wordpress-vs-drupal/>
- Jackson, B., 2021. *Kinsta*. [Ηλεκτρονικό]
Available at: <https://kinsta.com/blog/joomla-vs-wordpress/>
- Koga, A., 2018. *evolving web*. [Ηλεκτρονικό]
Available at: <https://evolvingweb.ca/blog/what-drupal>
- krystal, 2021. *Learn to CODE with me*. [Ηλεκτρονικό]
Available at: <https://learntocodewith.me/posts/backend-development/>
- LAMPRECHT, E., 2021. *CareerFoundry*. [Ηλεκτρονικό]
Available at: <https://careerfoundry.com/en/blog/ux-design/the-difference-between-ux-and-ui-design-a-laymans-guide/>
- Lentz, C., 2020. *Results Repeat*. [Ηλεκτρονικό]
Available at: <https://resultsrepeat.com/why-good-web-design-is-important/>
- Little, C., 2021. *The history of web design*. [Ηλεκτρονικό]
Available at: <https://tillerdigital.com/blog/the-history-of-web-design/>
- Marinov, V., 2020. *freelancermap*. [Ηλεκτρονικό]
Available at: <https://www.freelancermap.com/blog/career-insights-front-end-developer/>
- marionletendart, 2018. *What is Web Development?*. [Online]
Available at: <https://blog.openclassrooms.com/en/2018/03/28/web-development-definition/>
- Martin, M., 2021. *Guru99*. [Ηλεκτρονικό]
Available at: <https://www.guru99.com/what-is-backend-developer.html>
- Stefanuk, A., 2021. *Learn to CODE with me*. [Ηλεκτρονικό]
Available at: <https://learntocodewith.me/posts/sql-guide/>
- Stevens, E., 2021. *CareerFoundry*. [Ηλεκτρονικό]
Available at: <https://careerfoundry.com/en/blog/web-development/how-much-will-i-earn-as-a-web-developer/>
- Still, J., 2021. *INSIDER*. [Ηλεκτρονικό]
Available at: <https://www.businessinsider.com/what-is-javascript>