

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

**Μεταπτυχιακό Δίπλωμα Ειδίκευσης
στην Εφαρμοσμένη Πληροφορική**



ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ
Μεταπτυχιακό στην
Εφαρμοσμένη
Πληροφορική

Ανάπτυξη Πρόσθετου (Plugin) σε WordPress για δημιουργία και διαχείριση Προγραμμάτων Διατροφής

Development of a Wordpress plugin for nutrition programs

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ



Σιώπης Π. Στυλιανός
ΑΕΜ: 135

Επιβλέπων
Dr. Τσιμπίρης Αλκιβιάδης

ΣΕΡΡΕΣ - 3 - 2021

Πρόλογος

Ο προγραμματισμός είναι τρόπος σκέψης. Ξεκινάει με έντονο ενθουσιασμό και δημιουργικότητα. Έχει αρκετές δυσκολίες, αρκετές ώρες αναζήτησης για πιθανά λάθη και βελτιώσεις. Με σωστό οργανόγραμμα φτάνεις στο αποτέλεσμα που θέλεις. Και καταλαβαίνεις ότι τίποτα δεν τελειώνει εδώ. Πάντα θα βελτιώνεσαι, πάντα θα μαθαίνεις ...

Στην οικογένεια μου

Περίληψη στα Ελληνικά

Οι περιπτώσεις διαταραχών διατροφής αυξάνονται καθημερινά με όλο και περισσότερους ανθρώπους να χρειάζονται εξειδικευμένη βοήθεια και υποστήριξη. Το wordpress είναι το πιο διαδεδομένο CMS αυτή τη στιγμή, διατίθεται δωρεάν και είναι open-source. Η δημιουργία plugin για το wordpress ενισχύει αυτή τη δυναμική που έχει. Σκοπός της εργασίας αυτής είναι η δημιουργία Πρόσθετου (Plugin) σε Wordpress για τη δημιουργία προγραμμάτων διατροφής. Το πρόσθετο θα δημιουργεί μια νέα βάση δεδομένων σε ήδη υπάρχον δικτυακό τόπο που είναι σχεδιασμένο σε Wordpress. Η γλώσσα προγραμματισμού θα είναι η php στην οποία είναι σχεδιασμένο και το Wordpress. Ο χρήστης θα έχει τη δυνατότητα να δημιουργεί το δικό του πελατολόγιο όπου θα μπορεί να καταχωρεί τα ανθρωπομετρικά στοιχεία καθώς και το διαιτολόγιο του κάθε πελάτη. Ο χρήστης θα μπορεί να δημοσιεύει διαιτολόγια με τη χρήση shortcode.

Λέξεις-Κλειδιά: πρόσθετο Wordpress, πρόγραμμα διατροφής, διαιτολόγιο, ανθρωπομετρία, BMI, BMR, συντελεστής δραστηριότητας, ισοδύναμα τροφίμων, θρεπτική αξία

Summary

Cases of eating disorders are increasing daily with more and more people in need of specialized help and support. Wordpress is the most popular cms at the moment, it is available for free and it is open-source. Creating a plugin for wordpress enhances this dynamic. The purpose of this work is to create a Plugin in Wordpress to create nutrition programs. The plugin will create a new database on an existing Wordpress-designed website. The programming language will be php in which Wordpress is designed. The user will be able to create his own client registration where he will be able to enter the anthropometric data as well as the diet of each customer. The user will be able to publish diet programs using shortcode.

Keywords: Wordpress plugin, nutrition programs, diet, anthropometry, BMI, BMR, BMI, BMR, activity factor, food equivalents, nutritional value

Πίνακας περιεχομένων

ΚΕΦΑΛΑΙΟ 1^ο Εισαγωγή	1
1.1 Αντικείμενο της διπλωματικής.....	1
1.2 Οργάνωση του τόμου	1
ΚΕΦΑΛΑΙΟ 2^ο Περιγραφή του θέματος.....	3
2.1 Ανάλυση του αντικειμένου	3
2.2 Προεπισκόπηση, σχετικές εργασίες.....	4
2.3 Συνεισφορά της εργασίας	4
ΚΕΦΑΛΑΙΟ 2^ο Ανάλυση και σχεδίαση	5
3.1 Θεωρητική ανάλυση	5
3.2 Διαδικασίες επίλυσης.....	5
ΚΕΦΑΛΑΙΟ 4^ο Υλοποίηση	7
4.1 Πλατφόρμες και προγραμματιστικά εργαλεία	7
4.2 Λεπτομέρειες υλοποίησης.....	8
4.2.1 Φάκελος foodplan.....	10
4.2.1.1 Φάκελος <i>admin</i>	12
4.2.1.2 Φάκελος <i>includes</i>	18
4.2.1.3 Κλάση <i>Foodplan</i>	21
4.2.1.4 Κλάση <i>Foodplan_Post_Types</i>	23
4.2.1.5 Φάκελος <i>languages</i>	42
4.2.1.6 Φάκελος <i>public</i>	47
4.2.1.7 Φάκελος <i>templates</i>	49
4.2.1.8 Φάκελος <i>vendor</i>	50
4.2.1.9 Αρχείο <i>foodplan.php</i>	51
4.2.1.10 Αρχείο <i>LICENSE.txt</i>	54

4.2.1.11	Αρχείο README.txt	55
4.2.1.12	Αρχείο uninstall.php	56
4.2.2	Ισοδύναμα τροφίμων	57
4.2.2.1	Γάλα.....	58
4.2.2.2	Φρούτα.....	59
4.2.2.3	Λαχανικά	60
4.2.2.4	Αμυλούχα.....	61
4.2.2.5	Κρέας	62
4.2.2.6	Λίπη	62
4.2.3	Ανάπτυξη web εφαρμογής	63
4.2.4	Εγκατάσταση πρόσθετου και παραδείγματα.....	65
ΚΕΦΑΛΑΙΟ 5^ο Επίλογος.....		71
5.1	Σύνοψη και συμπεράσματα	71
5.2	Μελλοντικές επεκτάσεις.....	71
ΚΕΦΑΛΑΙΟ 6^ο Βιβλιογραφία		72

ΚΕΦΑΛΑΙΟ 1^ο

Εισαγωγή

Μετά τη βιομηχανική επανάσταση ήρθε και η επανάσταση της πληροφορίας. Ολοένα και περισσότεροι άνθρωποι βρίσκονται καθηλωμένοι στο χώρο του διαδικτύου. Ο σχεδιασμός και η ανάπτυξη ιστοσελίδων γίνεται ολοένα και πιο απαιτητικός. Η εξειδίκευση σε συγκεκριμένους τομείς της πληροφορικής καθώς και ο συνδυασμός επιστημών είναι απαραίτητα εργαλεία για να μπορέσεις να ακολουθήσεις αυτούς τους απαιτητικούς ρυθμούς ανάπτυξης.

1.1 Αντικείμενο της διπλωματικής

Η παρούσα διπλωματική βασίζεται στο συνδυασμό δύο επιστημών της Πληροφορικής και της Διαιτολογίας. Το αντικείμενο της διπλωματικής είναι η δημιουργία πρόσθετου λογισμικού (plugin) σε σύστημα διαχείρισης περιεχομένου (cms) το Wordpress. Το πρόσθετο αυτό δίνει μία επιπλέον δυναμική στον χρήστη του Wordpress. Πιο συγκεκριμένα, επιτρέπει στο χρήστη να δημιουργεί μία βάση δεδομένων για τη διαχείριση πελατολογίου και δεδομένων ανθρωπομετρίας καθώς και τη δημιουργία θερμοδομημένων προγραμμάτων διατροφής με βάση τα ισοδύναμα τροφίμων.

1.2 Οργάνωση του τόμου

Η βασική δομή της διπλωματικής μας χωρίζεται σε 6 κεφάλαια. Στο **1^ο Κεφάλαιο** ξεκινάμε εισαγωγικά με το αντικείμενο της διπλωματικής ώστε να μπορεί ο αναγνώστης με μια πρώτη ματιά να αντιληφθεί με το τι ασχοληθήκαμε.

Στο **2^ο Κεφάλαιο** ασχοληθήκαμε με τη περιγραφή του θέματος και δίνουμε στον αναγνώστη να καταλάβει την ιδιότητα των πρόσθετων (plugins) για το Wordpress, την έρευνα που κάναμε για το αν υπάρχει κάτι αντίστοιχο σε αυτό που δημιουργήσαμε και το πόσο βοήθησε η εργασία αυτή τόσο σε κοινωνικό όσο και σε προσωπικό επίπεδο.

Στο **3^ο Κεφάλαιο** αναφερόμαστε σε γενικές γραμμές στη διαδικασία που ακολουθήσαμε για να δημιουργήσουμε το πρόσθετό μας. Περιλαμβάνει τη θεωρητική ανάλυση του θέματος με μικρή αναφορά στα εργαλεία που χρησιμοποιήσαμε και τη λειτουργικότητα του plugin.

Στο **4^ο Κεφάλαιο** αναφερόμαστε στην υλοποίηση του πρόσθετου. Το κεφάλαιο αυτό είναι το ουσιαστικότερο για κάποιον που θέλει να μάθει ακριβώς τη διαδικασία και τα βήματα που ακολουθήσαμε για να δημιουργήσουμε αυτό το πρόσθετο. Περιλαμβάνει όλες τις λεπτομέρειες της υλοποίησης για να κατανοήσει ο αναγνώστης τα προγραμματιστικά εργαλεία και τις πλατφόρμες που χρησιμοποιήσαμε.

Το **5^ο Κεφάλαιο** είναι ο επίλογος της διπλωματικής όπου αναφερόμαστε στο τι αποκομίσαμε από την διαδικασία αυτή και πως θα διαχειριστούμε αυτή τη γνώση στο μέλλον.

Τέλος, στο **6^ο Κεφάλαιο** αναφερόμαστε στη βιβλιογραφία και τις ιστοσελίδες που μας βοήθησαν στην υλοποίηση της διπλωματικής.

ΚΕΦΑΛΑΙΟ 2^ο

Περιγραφή του θέματος

2.1 Ανάλυση του αντικειμένου

Το WordPress είναι ένα από τα πιο διαδεδομένα συστήματα διαχείρισης περιεχομένου με εκατομμύρια χρήστες. Εάν υπάρχει ένας βασικός κανόνας στην ανάπτυξη του WordPress, είναι ο εξής: **Μην αγγίζετε τον πυρήνα του WordPress.** Αυτό σημαίνει ότι δεν επεξεργαζόμαστε βασικά αρχεία WordPress για να προσθέσουμε λειτουργικότητα στον ιστότοπό μας. Αυτό συμβαίνει επειδή το WordPress αντικαθιστά τα βασικά αρχεία με κάθε ενημέρωση. Οποιαδήποτε λειτουργικότητα προστίθεται ή τροποποιείται πρέπει να γίνει χρησιμοποιώντας **πρόσθετα** (plugin) (Introduction to Plugin Development, 2021).

Τα πρόσθετα σας επιτρέπουν να επεκτείνετε σε μεγάλο βαθμό τη λειτουργικότητα του WordPress χωρίς να αγγίζετε τον ίδιο τον πυρήνα του WordPress. Είναι πακέτα κώδικα που επεκτείνουν τη βασική λειτουργικότητα του. Αποτελούνται από κώδικα PHP και μπορούν να περιλαμβάνουν άλλα στοιχεία όπως εικόνες, CSS και JavaScript.

Με βάση αυτά τα πρότυπα θα σχεδιάσουμε και το δικό μας πρόσθετο ώστε να μπορεί να χρησιμοποιείται σε οποιοδήποτε ιστότοπο Wordpress. Το πρόσθετο αυτό θα δίνει τη δυνατότητα στον χρήστη να δημιουργεί μία βάση δεδομένων πελατολογίου και στοιχείων ανθρωπομετρίας που είναι χρήσιμη για τον διαιτολόγο ή επαγγελματία υγείας. Επίσης, ο χρήστης θα μπορεί να δημιουργεί θερμοδομετρημένα προγράμματα διατροφής σχεδιάζοντας τη διαιτητική εντολή (θερμιδική αξία και θρεπτικά συστατικά) με βάση τα ισοδύναμα τροφίμων, όπως θα αναλυθούν σε παρακάτω κεφάλαιο.

2.2 Προεπισκόπηση, σχετικές εργασίες

Μόνο στο επίσημο site του Wordpress (wordpress.org) υπάρχουν περίπου 60.000 πρόσθετα. Με μια γρήγορη αναζήτηση με λέξεις - κλειδιά «διατροφή και διαίτα» παρατηρούμε ότι δεν υπάρχει μεγάλος όγκος πρόσθετων πάνω σε αυτή τη θεματολογία. Ιδιαίτερα στο σχεδιασμό θερμιδομετρημένων προγραμμάτων διατροφής δεν βρήκαμε κάτι αντίστοιχο.

2.3 Συνεισφορά της εργασίας

Το πρόσθετο που δημιουργήσαμε δίνει τη δυνατότητα στον χρήστη να οργανώσει τη δουλειά του και το πελατολόγιο του μέσα από το προσωπικό του ιστότοπο. Επίσης μπορεί να χρησιμοποιηθεί για διαφημιστικούς και προωθητικούς λόγους μιας ιστοσελίδας. Τέλος, κάνοντας μία προσωπική αναφορά, κερδίσαμε αρκετά μέσα από τη διαδικασία αυτή μαθαίνοντας HTML, CSS, JavaScript, bootstrap, php, Wordpress, Wordpress Plugin Development και δεσμευόμαστε για το σχεδιασμό αρκετών plugin στο μέλλον.

ΚΕΦΑΛΑΙΟ 3^ο

Ανάλυση και σχεδίαση

3.1 Θεωρητική ανάλυση

Για τον σχεδιασμό του πρόσθετου χρησιμοποιήσαμε αρχικά τοπικό server μέσω του προγράμματος XAMPP και ως συντάκτη (editor) για τον κώδικά μας χρησιμοποιήσαμε τον Visual Studio Code. Ολοκληρώνοντας το plugin δημιουργήσαμε μία WordPress ιστοσελίδα foodplanwp.com για να μπορούμε να παρουσιάσουμε τη λειτουργικότητα του.

Για να έχουμε σωστά οργανωμένο τον κώδικά μας χρησιμοποιήσαμε το Wordpress Boilerplate Plugin Generator, ένα πρότυπο εξειδικευμένο για το σχεδιασμό πρόσθετων. Το εργαλείο αυτό είναι ένα τυποποιημένο, καλά οργανωμένο, σχεδιασμένο σύμφωνα με τον αντικειμενοστραφή προγραμματισμό, πρότυπο για τη δημιουργία υψηλής ποιότητας πρόσθετων (plugin) για Wordpress (THE WORDPRESS PLUGIN BOILERPLATE, 2021).

Η τεκμηρίωση (Documentation) του πρότυπου αυτού είναι αυτή που απαιτείται από το ίδιο το Wordpress. Η οργάνωση των αρχείων και των φακέλων του πρόσθετου είναι έτσι σχεδιασμένη ώστε να γίνεται κατανοητός ο κώδικας και από άλλους προγραμματιστές πρόσθετων ιδιαίτερα όταν το πρόσθετο αυτό γίνεται αρκετά σύνθετο. Θα αναλύσουμε περισσότερο τα αρχεία για τη δημιουργία του πρόσθετου στο επόμενο κεφάλαιο.

3.2 Διαδικασίες επίλυσης

Αφού σχεδιάσαμε τη «ραχοκοκαλιά» του plugin είναι καιρός να προσθέσουμε τον κώδικά μας για να δώσουμε λειτουργικότητα στο plugin μας. Το πρόσθετο αυτό δημιουργεί ένα επιπλέον μενού για τον χρήστη (admin menu) με τη ονομασία Foodplan. Μέσα στο μενού αυτό προσθέτουμε 4 υπομενού.

1. Υπομενού **Foodplan** που είναι μια στατική σελίδα παρουσίασης του πρόσθετου σε περιβάλλον του χρήστη
2. Υπομενού **Ισοδύναμα** που είναι μια στατική σελίδα που περιλαμβάνει μία αναλυτική λίστα των ισοδυνάμων τροφίμων για να διευκολύνει το χρήστη στη σωστή θερμιδομέτρηση των προγραμμάτων διατροφής.
3. Υπομενού **Πελατολόγιο** που είναι ένας προσαρμοσμένος τύπος δημοσίευσης (custom post type) και είναι ορατός μόνο στο περιβάλλον του χρήστη (public => false) και δημιουργεί μία βάση δεδομένων για το πελατολόγιο του χρήστη. Εδώ προσθέσαμε αρκετά προσαρμοσμένα πεδία (custom fields) που θα αναλυθούν στο επόμενο κεφάλαιο.
4. Υπομενού **Διαιτολόγιο** που είναι ένας προσαρμοσμένος τύπος δημοσίευσης (custom post type) και είναι ορατός δημόσια (public => true) και επιτρέπει στο χρήστη να δημιουργεί θερμιδομετρημένα προγράμματα διατροφής σύμφωνα με τα ισοδύναμα τροφίμων. Με τη χρήση **shortcode** που προσθέτει ο χρήστης μπορεί να εξάγει τα παραπάνω δεδομένα. Η διαδικασία αναλύεται με λεπτομέρεια στο παρακάτω κεφάλαιο.

ΚΕΦΑΛΑΙΟ 4^ο

Υλοποίηση

4.1 Πλατφόρμες και προγραμματιστικά εργαλεία

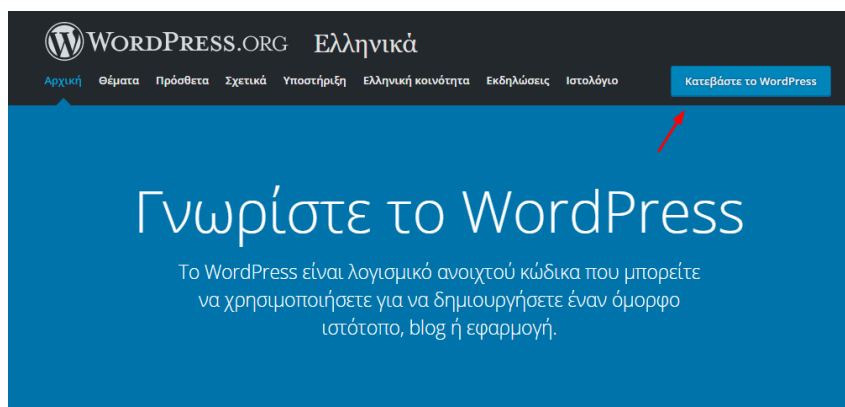
Χρησιμοποιήσαμε τα παρακάτω προγραμματιστικά εργαλεία για την υλοποίηση της εργασίας μας:

1. **Wordpress:** Το CMS στο οποίο εγκαταστήσαμε το plugin
2. **XAMPP:** Για τη δημιουργία τοπικού server
3. **Visual Studio Code:** ο editor που χρησιμοποιήσαμε για την επεξεργασία του plugin
4. **Wordpress Boilerplate Plugin Generator:** πρότυπο σχεδιασμού plugin
5. **Γλώσσα προγραμματισμού php:** Η γλώσσα στην οποία γράψαμε το plugin
6. **Γλώσσα σήμανσης HTML:** σε πολλά σημεία του plugin περιλαμβάνονται γραμμές κώδικα HTML
7. **Γλώσσα σχεδιασμού CSS:** αρκετές γραμμές κώδικα CSS για να φτάσουμε στο σχεδιαστικό αποτέλεσμα του πρόσθετου.
8. **Γλώσσα προγραμματισμού Javascript:** όλες οι εξισώσεις που χρησιμοποιήθηκαν δημιουργήθηκαν με Javascript. Επίσης, σε κάποια σημεία του κώδικά μας χρησιμοποιήσαμε jQuery και jQuery UI που ανήκουν στη βιβλιοθήκη της Javascript.
9. **Συλλογή εργαλείων Bootstrap:** Χρησιμοποιήσαμε αρκετές κλάσεις Bootstrap στον κώδικά μας για καλύτερο σχεδιαστικό αποτέλεσμα.
10. **Γραφιστικά Εργαλεία (Adobe Illustrator & Photoshop):** Για τον σχεδιασμό του λογότυπου, favicon και των εικόνων που χρησιμοποιήσαμε στα παραδείγματά μας.

Παρακάτω θα αναλύσουμε με λεπτομέρεια τον τρόπο που χρησιμοποιήσαμε όλα τα παραπάνω.

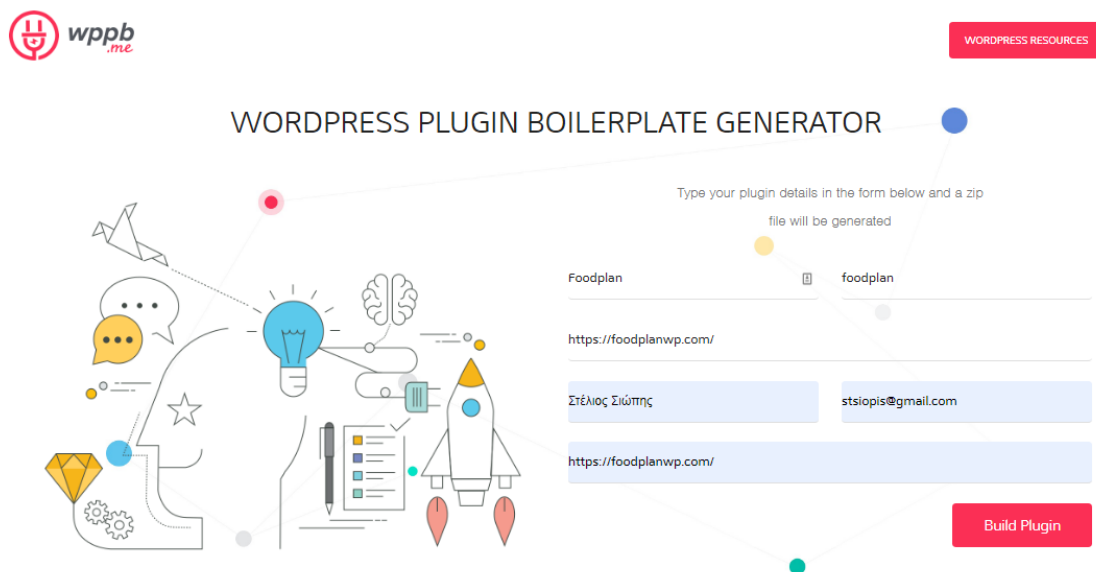
4.2 Λεπτομέρειες υλοποίησης

Ξεκινήσαμε αρχικά να δημιουργούμε το πρόσθετό μας σε τοπικό server. Με τη βοήθεια του προγράμματος XAMPP (XAMPP, 2021) εγκαταστήσαμε τοπικά το Wordpress (εικόνα 4.1) (Γνωρίστε το Wordpress, 2021). Όταν ολοκληρώσαμε το plugin δημιουργήσαμε μία Wordpress ιστοσελίδα σε online server που διαχειριζόμαστε για να μπορέσουμε να παρουσιάσουμε τη λειτουργικότητα του πρόσθετου στο διαδίκτυο.



Εικόνα 4.1

Δημιουργήσαμε αρχικά τη δομή του plugin με τη βοήθεια του Wordpress Boilerplate Plugin Generator. (εικόνα 4.2) (WORDPRESS PLUGIN BOILERPLATE GENERATOR, 2021)

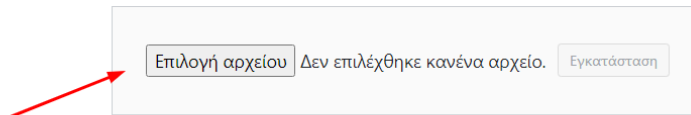


Εικόνα 4.2

Κατεβάσαμε το αρχείο foodplan.zip που προέκυψε από τον παραπάνω Generator και το εγκαταστήσαμε στην Wordpress ιστοσελίδα που δημιουργήσαμε σε local server. (εικόνα 4.3)

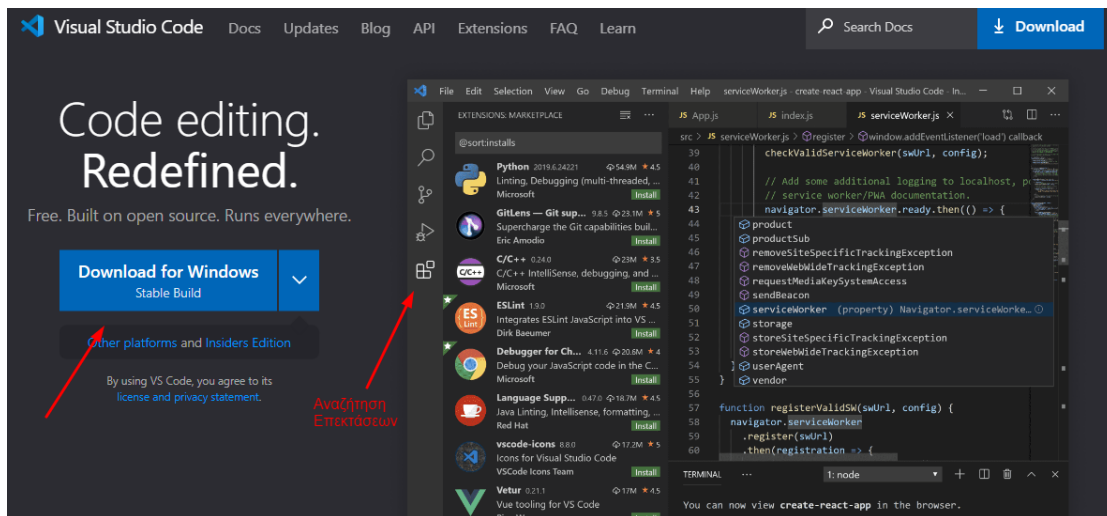
Προσθήκη προσθέτων Μεταφόρτωση Πρόσθετου

Αν έχετε ένα πρόσθετο σε μορφή .zip, μπορείτε να εγκαταστήσετε ή να το αναβαθμίσετε μεταφορτώντας το εδώ.



Εικόνα 4.3

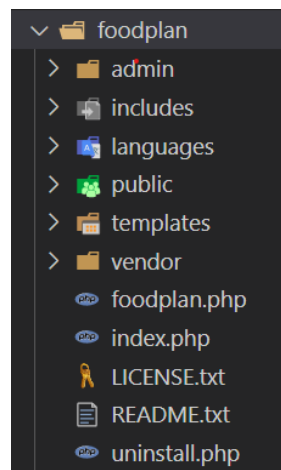
Χρησιμοποιήσαμε ως συντάκτη (editor) για τον κώδικά μας τον Visual Studio Code. Είναι ένας δωρεάν editor με βασικές επεκτάσεις για το Wordpress (extensions) όπως Wordpress Snippets, Wordpress Development Toolkit. (εικόνα 4.4) (Code editing, 2021)



Εικόνα 4.4

4.2.1 Φάκελος foodplan

Ο φάκελος Foodplan του πρόσθετου έχει μια συγκεκριμένη δομή σύμφωνα με τα πρότυπα σχεδιασμού plugin του Wordpress. (εικόνα 4.5) (Best Practices, 2021)



Εικόνα 4.5

Πιο συγκεκριμένα περιέχει τους παρακάτω υποφακέλους:

1. **Φάκελος admin:** εδώ δημιουργούμε όλο το περιεχόμενο του κώδικα που αναφέρετε στο περιβάλλον του χρήστη.
2. **Φάκελος includes:** περιλαμβάνει τις κλάσεις που δημιουργήσαμε και το μεγαλύτερο περιεχόμενο του κώδικα σε αυτό το πρόσθετο
3. **Φάκελος languages:** στον οποίο προσθέτουμε τις γλώσσες που θέλουμε να μεταφράσουμε το πρόσθετο
4. **Φάκελος public:** στον οποίο περιέχονται όλα τα αρχεία που είναι σε επίπεδο δημοσίευσης

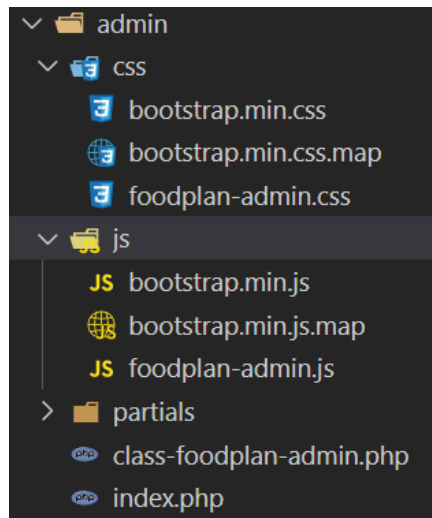
Στους παραπάνω φακέλους εμείς προσθέσαμε τον **φάκελο templates και vendor** στους οποίους τοποθετούμε εξωτερικά αρχεία που χρησιμοποιεί το πρόσθετο. Επίσης, το πρόσθετο περιλαμβάνει τα παρακάτω βασικά αρχεία:

1. Το αρχείο **foodplan.php** που περιλαμβάνει τα βασικά χαρακτηριστικά του πρόσθετου.
2. Το αρχείο **index.php** που συνήθως είναι ένα άδειο αρχείο με ένα απλό σχόλιο php και χρησιμεύει κυρίως στην ασφάλεια του πρόσθετου.
3. Το αρχείο **LICENSE.txt** όπου περιλαμβάνει την άδεια που χρησιμοποιούμε για τη χρήση του πρόσθετου.
4. Το αρχείο **README.txt** που είναι ένα αρχείο με βασικές πληροφορίες για το πρόσθετο.
5. Το αρχείο **uninstall.php** που χρησιμοποιείται όταν θέλει ο χρήστης να διαγράψει το πρόσθετο

Όλο το περιεχόμενο των φακέλων και των αρχείων θα αναλυθεί με λεπτομέρεια παρακάτω.

4.2.1.1 Φάκελος *admin*

Ο φάκελος αυτός περιλαμβάνει όλα τα αρχεία που χρησιμοποιούνται για το περιβάλλον του χρήστη (εικόνα 4.6).

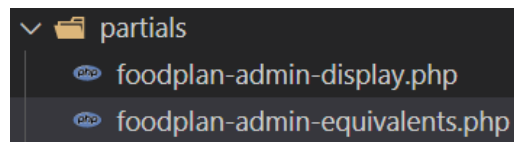


Εικόνα 4.6

Πιο συγκεκριμένα περιέχει τους παρακάτω υποφακέλους και αρχεία:

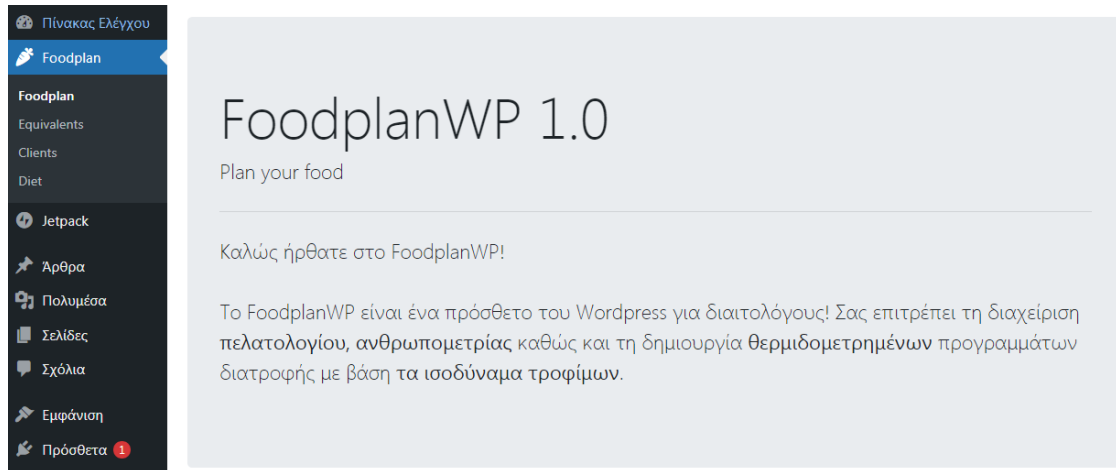
Τον φάκελο `css` και `js` όπου προσθέτουμε όλη το περιεχόμενο `css` και `javascript`. Όπως παρατηρείται στην εικόνα 6 ήδη έχουμε προσθέσει στο `foodplan` τη δυνατότητα να χρησιμοποιεί **bootstrap**. Ο τρόπος που το κάναμε αυτό θα αναφερθεί πιο αναλυτικά παρακάτω.

Ο φάκελος `partials` περιέχει δύο αρχεία `php` που δημιουργούν τις αντίστοιχες σελίδες στον πίνακα ελέγχου (εικόνα 4.7). Παρακάτω αναλύουμε το περιεχόμενο των 2 αυτών αρχείων.



Εικόνα 4.7

foodplan-admin-display.php: Είναι η αρχική σελίδα που βλέπει ο χρήστης μόλις ανοίγει το Foodplan στον πίνακα ελέγχου. Η σελίδα αυτή περιλαμβάνει bootstrap και ένα απλό ενημερωτικό κείμενο για το plugin (εικόνα 4.8).



Εικόνα 4.8

foodplan-admin-equivalents.php: Η σελίδα αυτή περιλαμβάνει μια λίστα με τα ισοδύναμα τροφίμων. Στη σελίδα αυτή χρησιμοποιήσαμε επιπλέον **jQuery UI** (jQueryUI: tabs, 2021) για να διαμορφώσουμε τη λίστα μας. Μικρό παράδειγμα από τον κώδικα της συγκεκριμένης σελίδας φαίνεται στην εικόνα 4.9.

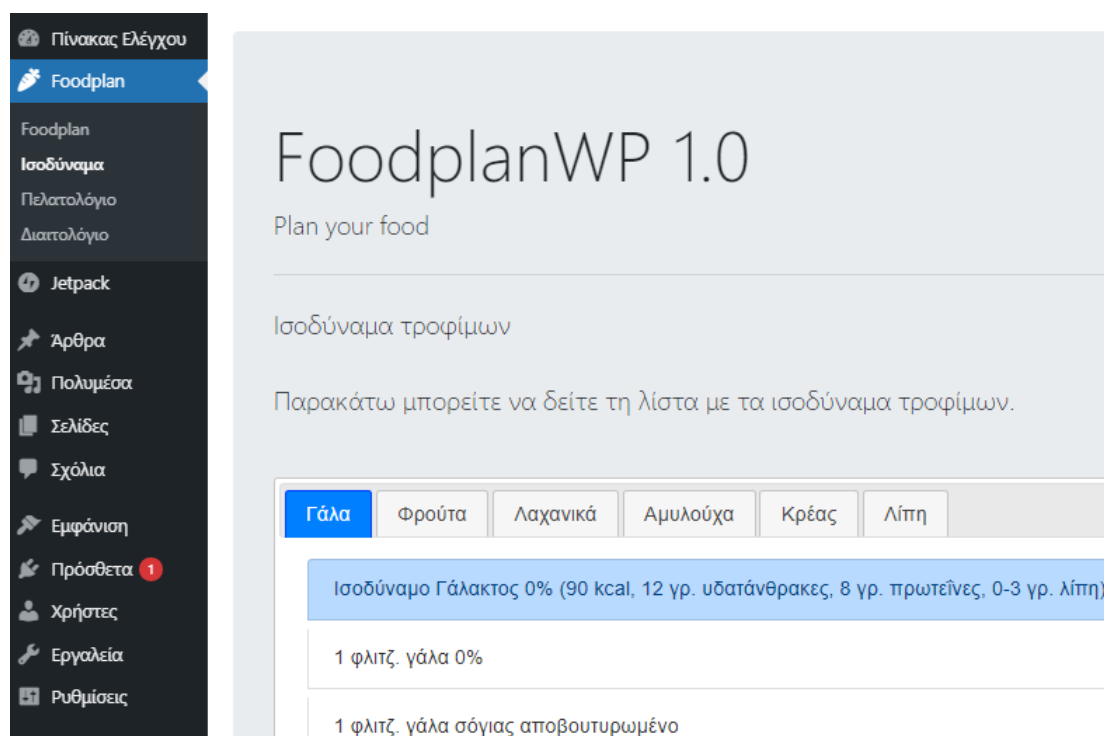
```
<script src="https://code.jquery.com/jquery-1.12.4.js"></script>
<script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>

<script>
$( function() {
  $( "#tabs" ).tabs();
} );
</script>

<div class="jumbotron">
<h1 class="display-4">FoodplanWP 1.0</h1>
<p class="lead">Plan your food</p>
<hr class="my-4">
<p class="lead">Ισοδύναμα τροφίμων <br> <br>Παρακάτω μπορείτε να δείτε σε μορφή λίστας τα
ισοδύναμα τροφίμων.<br><br> </p>
```

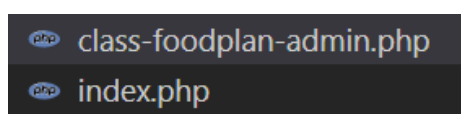
Εικόνα 4.9

Τελικά φτάσαμε στο οπτικό αποτέλεσμα χρήστη όπως φαίνεται στην εικόνα 4.10.



Εικόνα 4.10

Τέλος, ο φάκελος admin περιέχει 2 επιπλέον αρχεία php. Το αρχείο class-foodplan-admin.php και το αρχείο index.php (εικόνα 4.11). Παρακάτω αναλύουμε το περιεχόμενο των 2 αυτών αρχείων.



Εικόνα 4.11

Το αρχείο **class-foodplan-admin.php** είναι το πιο σημαντικό αρχείο στον φάκελο admin. Εδώ δημιουργήσαμε μία βασική κλάση **class Foodplan_Admin**. Η κλάση αυτή περιλαμβάνει αρκετές συναρτήσεις (functions) που δημιουργούν αντίστοιχα το οπτικό αποτέλεσμα που έχουμε στο περιβάλλον του χρήστη. Οι συναρτήσεις αυτές είναι οι παρακάτω:

public function enqueue_style (Code Reference: enqueue_style, 2021) και **public function enqueue_scripts** (Code Reference: enqueue_scripts, 2021): εδώ προσθέτουμε όλα τα αρχεία που χρησιμοποιούμε για css και javascript. Επίσης όπως φαίνεται στην εικόνα 4.12 προσθέσαμε και bootstrap (bootstrap, 2021) για το περιβάλλον του χρήστη.

```
public function enqueue_styles() {  
    /** ...  
    wp_enqueue_style( $this->plugin_name, plugin_dir_url( __FILE__ ) . 'css/  
    foodplan-admin.css', array(), $this->version, 'all' );  
    wp_enqueue_style( 'bootstrap-css', plugin_dir_url( __FILE__ ) . 'css/bootstrap.min.  
    css', array(), $this->version, 'all' );  
}  
/** ...  
public function enqueue_scripts() {  
    /** ...  
    wp_enqueue_script( $this->plugin_name, plugin_dir_url( __FILE__ ) . 'js/  
    foodplan-admin.js', array( 'jquery' ), $this->version, false );  
    wp_enqueue_script( 'bootstrap-js', plugin_dir_url( __FILE__ ) . 'js/bootstrap.min.  
    js', array( 'jquery' ), $this->version, false );  
}
```

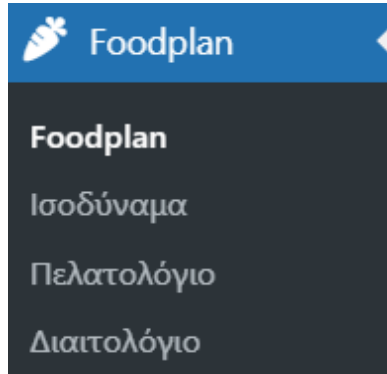
Εικόνα 4.12

function fp_admin_menus: η συνάρτηση αυτή προσθέτει το μενού του πρόσθετου στον πίνακα ελέγχου. Στην εικόνα 4.13 φαίνονται και τα 4 υπομενού που δημιουργήσαμε για το plugin. (Administration Menus, 2021)

```
function fp_admin_menus() {  
    add_menu_page( '', 'Foodplan', 'manage_options', 'foodplan/mainsettings.php', array(  
    $this, 'fp_login_page' ), 'dashicons-carrot', 2 );  
  
    add_submenu_page( 'foodplan/mainsettings.php', 'Equivalentents', 'Equivalentents',  
    'manage_options', '', array( $this, 'fp_equivalentents_page' ) );  
  
    add_submenu_page( 'foodplan/mainsettings.php', '', 'Clients', 'manage_options',  
    'edit.php?post_type=fp-clients');  
  
    add_submenu_page( 'foodplan/mainsettings.php', '', 'Diet', 'manage_options', 'edit.  
    php?post_type=diet');  
}
```

Εικόνα 4.13

Το οπτικό αποτέλεσμα που έχουμε από την παραπάνω συνάρτηση στον πίνακα ελέγχου φαίνεται στην εικόνα 4.14. Επίσης, φαίνεται και το σύμβολο που χρησιμοποιήσαμε στην ετικέτα του πρόσθετου (**dashicons-carrot**).



Εικόνα 4.14

function fp_login_page και **function fp_equivalents_page**: οι συναρτήσεις αυτές συνδέουν τις 2 σελίδες που έχουμε στον φάκελο admin/partials με το μενού που δημιουργήσαμε. (εικόνα 4.15)

```
function fp_login_page() {  
    require_once 'partials/foodplan-admin-display.php';  
}  
  
function fp_equivalents_page() {  
    require_once 'partials/foodplan-admin-equivalents.php';  
}
```

Εικόνα 4.15

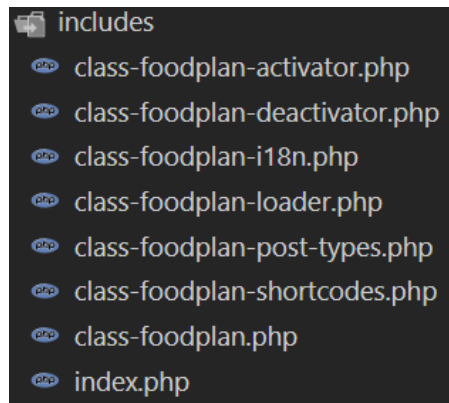
Το αρχείο **index.php** το οποίο χρησιμοποιείται συχνά από τον Generator συνήθως είναι ένα αρχείο με 1 απλό σχόλιο php (εικόνα 4.16) και προσφέρει μία επιπλέον ασφάλεια στο πρόσθετό μας. Η χρήση του αρχείου αυτού έγκειται στο γεγονός ότι αν κάποιος θα ήθελε να προσβάλει την σελίδα μας μέσω του πρόσθετού μας αρχικά θα έψαχνε για κάποιο αρχείο τύπου index.php που στη προκειμένη περίπτωση είναι άδειο. Η χρήση του αρχείου αυτού είναι **προαιρετική** και θα μπορούσαμε αν θέλαμε να το διαγράψουμε.

```
1 <?php // Silence is golden
```

Εικόνα 4.16

4.2.1.2 Φάκελος *includes*

Στον φάκελο αυτό δημιουργούμε όλες τις κλάσεις για το πρόσθετό μας για να του δώσουμε την λειτουργικότητα που θέλουμε. Η ιεραρχία του φακέλου φαίνεται παρακάτω (εικόνα 4.16). Στον φάκελο αυτό δημιουργήσαμε **8 αρχεία php (κλάσεις)**. Παρακάτω αναλύουμε το περιεχόμενο τους.



Εικόνα 4.16

1. **class-foodplan-activator.php**: Στο αρχείο αυτό προσθέσαμε μία συνάρτηση (**public static function activate**) η οποία φορτώνει μόνο όταν ενεργοποιείται το πρόσθετο και ουσιαστικά προσθέτει τους προσαρμοσμένους τύπους δημοσίευσης (custom post types) που δημιουργήσαμε (εικόνα 4.17). Τέλος, χρησιμοποιούμε μια επιπλέον συνάρτηση (**flush_rewrite_rules**) η οποία επιτρέπει την αυτόματη εκκαθάριση των κανόνων επανεγγραφής του WordPress. Η συνάρτηση αυτή είναι απαραίτητη για την προβολή και δημοσίευση των custom post types. (Activation / Deactivation Hooks, 2021)

```
public static function activate() {  
    require_once FOODPLAN_BASE_DIR . 'includes/class-foodplan-post-types.php';  
  
    // Register CPT  
    $plugin_post_type = new Foodplan_Post_Types(FOODPLAN_NAME, FOODPLAN_VERSION);  
    $plugin_post_type->init();  
  
    // Flush permalinks  
    flush_rewrite_rules();  
}
```

Εικόνα 4.17

2. **class-foodplan-deactivator.php:** Στο αρχείο αυτό προσθέσαμε μία συνάρτηση (**public static function deactivate**) η οποία διαγράφει τα custom post types που δημιουργήσαμε όταν το πρόσθετό μας απενεργοποιηθεί (εικόνα 4.18). Επίσης, χρησιμοποιούμε πάλι τη συνάρτηση (**flush_rewrite_rules**). (Activation / Deactivation Hooks, 2021)

```
public static function deactivate() {  
  
    // Un-register CPT  
  
    unregister_post_type('fp-clients');  
    unregister_post_type('diet');  
  
    // Flush Rewrite Rules  
    flush_rewrite_rules();  
  
}
```

εικόνα 4.18

3. **class-foodplan-i18n.php:** Το αρχείο αυτό χρησιμοποιείται για τη μετάφραση του πρόσθετου. Στο αρχείο αυτό δημιουργήσαμε μία κλάση **class Foodplan_i18n** που περιλαμβάνει μία συνάρτηση **public function load_plugin_textdomain()** που «φορτώνει» το αρχείο της μετάφρασης που βρίσκεται στο φάκελο languages (εικόνα 4.19). (How to Internationalize Your Plugin, 2021) Η διαδικασία μετάφρασης του πρόσθετου περιγράφεται με λεπτομέρεια σε ξεχωριστή ενότητα παρακάτω.

```
class Foodplan_i18n {  
  
    /**  
     * Load the plugin text domain for translation.  
     *  
     * @since 1.0.0  
     */  
  
    public function load_plugin_textdomain() {  
  
        load_plugin_textdomain(  
            'foodplan',  
            false,  
            dirname( dirname( plugin_basename( __FILE__ ) ) ) . '/languages/'  
        );  
  
    }  
  
}
```

εικόνα 4.19

4. **class-foodplan-loader.php:** Το αρχείο αυτό φορτώνει όλα τα actions και filters του πρόσθετου.
5. **class-foodplan-post-types.php:** Το αρχείο αυτό περιλαμβάνει τον περισσότερο κώδικα που γράψαμε για το πρόσθετο. Για το λόγο αυτό θα το αναλύσουμε σε ξεχωριστή ενότητα παρακάτω.
6. **class-foodplan-shortcodes.php:** Το αρχείο αυτό περιλαμβάνει μία συνάρτηση public function diet_list (\$atts, \$content) που είναι υπεύθυνη για τη δημιουργία του περιεχομένου του **shortcode [diet_list]** που δημιουργήσαμε (εικόνα 20). (Shortcodes, 2021)

```
// Shortcode for diet List
public function diet_list( $atts, $content ) {

    $paged = (get_query_var('paged') ? get_query_var('paged') : 1);

    $loop_args = array(
        'post_type'      => 'diet',
        'posts_per_page' => 3,
        'paged'          => $paged
    );

    ob_start();

    $loop = new WP_Query( $loop_args );
```

εικόνα 4.20

7. **class-foodplan.php:** Περιλαμβάνει μία κλάση με όνομα **class Foodplan**. Η κλάση αυτή είναι η πιο σημαντική για το πρόσθετο και θα την αναλύσουμε σε ξεχωριστή ενότητα.
8. **index.php:** Όπως έχουμε αναφέρει και σε προηγούμενους φακέλους είναι ένα άδειο αρχείο με ένα απλό σχόλιο php και χρησιμοποιείται για λόγους ασφάλειας του πρόσθετου.

4.2.1.3 Κλάση Foodplan

Το αρχείο class-foodplan.php περιέχει τη πιο σημαντική κλάση του πρόσθετου (**class Foodplan**). Η κλάση αυτή έχει αρκετές συναρτήσεις (functions) απαραίτητες για την εύρυθμη λειτουργία του plugin. Αυτές είναι:

1. **private function load_dependencies():** η συνάρτηση αυτή «φορτώνει» όλες τις εξαρτημένες κλάσεις που προαναφέραμε (εικόνα 4.21).

```
private function load_dependencies() {  
  
    /**  
     * The class responsible for orchestrating the actions and filters of the  
     * core plugin.  
     */  
    require_once plugin_dir_path( dirname( __FILE__ ) ) . 'includes/  
class-foodplan-loader.php';  
  
    /**  
     * The class responsible for defining internationalization functionality  
     * of the plugin.  
     */  
    require_once plugin_dir_path( dirname( __FILE__ ) ) . 'includes/class-foodplan-i18n.  
php';  
}
```

εικόνα 21

2. **private function define_admin_hooks():** Η συνάρτηση αυτή καθορίζει όλες τις συναρτήσεις (hooks = άγκιστρα) που σχετίζονται με το περιβάλλον του χρήστη και όπως φαίνεται στην εικόνα και το admin menu (εικόνα 4.22). (Hooks, 2021)

```
private function define_admin_hooks() {  
  
    $plugin_admin = new Foodplan_Admin( $this->get_plugin_name(), $this->get_version() )  
    ;  
  
    $this->loader->add_action( 'admin_enqueue_scripts', $plugin_admin, 'enqueue_styles'  
    );  
  
    $this->loader->add_action( 'admin_enqueue_scripts', $plugin_admin,  
    'enqueue_scripts' );  
  
    // Admin menu  
    $this->loader->add_action( 'admin_menu', $plugin_admin, 'fp_admin_menus' );  
}
```

εικόνα 22

3. **public function define_post_type_hooks():** Η συνάρτηση αυτή καθορίζει όλες τις συναρτήσεις που δημιουργήσαμε στην κλάση class-foodplan-post-types.php (εικόνα 4.23).

```
public function define_post_type_hooks() {  
  
    $plugin_post_types = new Foodplan_Post_Types($this->get_plugin_name(),  
    $this->get_version());  
  
    $this->loader->add_action( 'init', $plugin_post_types , 'init' );  
  
    $this->loader->add_filter( 'the_content' , $plugin_post_types, 'content_single_diet' )  
    ;  
  
    //Metaboxes ...  
    //Save Metaboxes ...  
  
}
```

εικόνα 4.23

4. **public function define_shortcode_hooks():** Η συνάρτηση αυτή καθορίζει όλα τα shortcode που δημιουργούμε με το πρόσθετό μας (εικόνα 4.24).

```
/**  
 * Defining all shortcodes for the plugin  
 */  
public function define_shortcode_hooks() {  
  
    $plugin_shortcodes = new Foodplan_Shortcodes(  
        $this->get_plugin_name(),  
        $this->get_version()  
    );  
  
    //Adding Shortcode  
    add_shortcode( 'diet_list', array( $plugin_shortcodes, 'diet_list' ) );  
  
}
```

εικόνα 4.24

4.2.1.4

Κλάση *Foodplan_Post_Types*

Το αρχείο **class-foodplan-post-types.php** περιλαμβάνει μία κλάση (**class *Foodplan_Post_Types***) μέσα στην οποία δημιουργούμε όλες τις συναρτήσεις (functions) που θέλουμε να δώσουμε στο plugin. Η κλάση αυτή (εικόνα 4.25) περιλαμβάνει περίπου 1000 γραμμές κώδικα και θα τις αναλύσουμε με λεπτομέρεια παρακάτω.

```
24 class Foodplan_Post_Types {
25
26 > // The ID, the Version, the initialization of the class...
58 > // Hooked into 'init' action hook...
68 > // Custom Post Type: Clients...
577 > // Custom Post Type: Diet...
978 > // Filter content for CPT: Diet...
1007 }
```

εικόνα 4.25

Παρακάτω αναφέρονται αριθμητικά όλες οι συναρτήσεις (functions) της κλάσης αυτής:

1. **public function init ()** : η συνάρτηση αυτή «φορτώνει» τα 2 custom post types που δημιουργήσαμε (εικόνα 26). Στη function αυτή μπορούμε να προσθέτουμε μελλοντικά οποιοδήποτε custom post type θέλουμε να δημιουργεί το πρόσθετό μας.

```
public function init() {
    $this->fp_register_fp_clients();
    $this->fp_register_fp_diet();
}
```

εικόνα 4.26

2. **public function fp_register_fp_diet():** Η function αυτή εγκαθιστά το custom post type: Diet (εικόνα 4.27) δηλαδή τη δημιουργία θερμιδομετρημένων διαιτολογίων από τον χρήστη με τη βοήθεια των διαβητικών ισοδυνάμων (Registering Custom Post Types, 2021). Για να γίνει κατανοητή αυτή η διαδικασία θα κάνουμε μία πλήρη αναφορά σε επόμενη ενότητα στο σύστημα ισοδυνάμων μετά την ολοκλήρωση της περιγραφής του πρόσθετου.

```
public function fp_register_fp_diet() {  
  
    $labels = [  
        'name'           => _x( 'Diets', 'post type general name',  
        'Diets' ),  
        'singular_name' => _x( 'Diet', 'post type singular name',  
        'Diet' ),  
        'menu_name'     => _x( 'Diets', 'admin menu', 'Diets' ),  
        'name_admin_bar' => _x( 'Diet', 'add new on admin bar', 'Diet' ),  
        'add_new'       => _x( 'Add New', 'Diet', 'Diet' ),  
        'add_new_item'  => __( 'Add New Diet', 'Diet' ),  
        'new_item'      => __( 'New Diet', 'Diet' ),  
        'edit_item'     => __( 'Edit Diet', 'Diet' ),  
        'view_item'    => __( 'View Diet', 'Diet' ),  
        'all_items'     => __( 'All Diet', 'Diet' ),  
        'search_items'  => __( 'Search Diet', 'Diet' ),  
        'parent_item_colon' => __( 'Parent Diet:', 'Diet' ),  
        'not_found'     => __( 'No Diet found.', 'Diet' ),  
        'not_found_in_trash' => __( 'No Diet found in Trash.', 'Diet' )  
    ];  
  
    $args = [  
        "label" => __( "Diet"),  
        "labels" => $labels,  
    ]  
}
```

εικόνα 4.27

Με την παραπάνω function δίνουμε τη δυνατότητα στο χρήστη να καταχωρεί νέες δημοσιεύσεις σε μορφή διαιτολογίου. Τα post αυτά θα είναι **δημόσια** ("public" => true) (εικόνα 4.28) οπότε ο χρήστης θα μπορεί να τα δημοσιεύσει στον δικτυακό του χώρο με τη χρήση shortcode.

```
$args = [  
    "label" => __( "Diet"),  
    "labels" => $labels,  
    "description" => "",  
    "public" => true,  
    "show_ui" => true,  
]
```

εικόνα 4.28

Επίσης, για το post αυτό χρησιμοποιήσαμε κάποια βοηθήματα (**supports**) του Wordpress (εικόνα 4.29) (Code Reference: post_type_supports, 2021).

```
"supports" => [ "title", 'editor', 'author', 'thumbnail' ],
"taxonomies" => [],
"can_export"=> true,
```

εικόνα 4.29

- 3. public function fp_meals_custom_box():** Η συνάρτηση δημιουργεί ένα προσαρμοσμένο πεδίο Διαιτητική Εντολή (εικόνα 4.30) (Custom Meta Boxes, 2021). Το πεδίο αυτό θα μας δίνει τη δυνατότητα να προσθέτουμε τις θερμίδες καθώς και 3 μακροθρεπτικά συστατικά στο διαιτολόγιο **τους υδατάνθρακες, τις πρωτεΐνες και τα λίπη.**

```
// Custom field: Dietary Order

public function fp_meals_custom_box() {
    $screens = [ 'Diet' ];
    foreach ( $screens as $screen ) {
        add_meta_box(
            'fp_meals_id',
            __('Διαιτητική Εντολή'),
            array ( $this, 'fp_meals_custom_box_html' ),
            $screen,
            'normal'
        );
    }
}
```

εικόνα 4.30

4. **public function fp_meals_custom_box_html(\$post):** Η συνάρτηση αυτή δημιουργεί το οπτικό αποτέλεσμα σε περιβάλλον χρήστη για το παραπάνω προσαρμοσμένο πεδίο (custom field) (Custom Meta Boxes, 2021). Περιλαμβάνει όλον τον κώδικα HTML, css, javascript και bootstrap για το πεδίο αυτό (εικόνα 4.31).

```
public function fp_meals_custom_box_html( $post) {
    wp_nonce_field(basename(__FILE__), 'fp_meals_nonce');
    $fp_meals_stored_meta = get_post_meta( $post->ID);

    ?>
    <style>...
</style>

    <div class="container" id="table-equev"> ...
</div>
    <br>
    <div class="container" id="equev-results"> ...
</div>

    <script>...
</script>

    <?php
}
```

εικόνα 4.31

Στη παραπάνω function δημιουργήσαμε 2 “container” κλάσεις bootstrap. Το πρώτο περιλαμβάνει τον πίνακα Ισοδυνάμων (εικόνα 4.32). Εδώ ο χρήστης μπορεί να καταχωρεί τον αριθμό των ισοδυνάμων που περιέχει το διαιτολόγιο που σχεδιάζει.

Ομάδες Τροφίμων	Θερμίδες
Γάλα άπαχο	0
Γάλα ημιάπαχο	3

εικόνα 4.32

Το δεύτερο «container» περιλαμβάνει τη θρεπτική αξία του διαιτολογίου (εικόνα 4.33).

Θρεπτική Αξία Διαιτολογίου	
Θερμίδες (kcal)	1375
Υδατάνθρακες (γρ.)	104
Πρωτεΐνες (γρ.)	73
Λίπη (γρ.)	71

εικόνα 4.33

Χρησιμοποιήσαμε βασικές συναρτήσεις css για να διαμορφώσουμε το περιβάλλον του χρήστη (εικόνα 4.34).

```
.container {  
  font-weight: bold;  
  border-radius: 25px;  
  padding: 1em;  
  background: bisque;  
  margin-bottom: 1em;  
  border-radius: 25px;  
  text-align: center;  
  align-content: center;  
  margin: auto;  
}
```

εικόνα 4.34

Επίσης χρησιμοποιήθηκαν και κάποιες κλάσεις bootstrap για καλύτερο σχεδιαστικό αποτέλεσμα και για να προσαρμόζεται η φόρμα μας σε όλες τις οθόνες (εικόνα 4.35).

```
<div class="container" id="table-equev">
  <div class="col" >
    <div class="col-12 form-group">
      <div class="row">
        <div class="col-12">
          <label><?php _e('Πίνακας Ισοδυνάμων Διαιτολογίου',
            'foodplan');?></label>
        </div>
      </div>
      <hr>
    </div>
  </div>
</div>
```

εικόνα 4.35

Για την επίτευξη των αποτελεσμάτων τόσο για τη καταχώρηση των ισοδυνάμων όσο και για τα αποτελέσματα της θρεπτικής αξίας του διαιτολογίου χρησιμοποιήθηκε javascript (εικόνα 4.36).

```
function change1() {

  times1 = document.getElementById('eq1').value;
  times2 = document.getElementById('eq2').value;
  times3 = document.getElementById('eq3').value;
  times4 = document.getElementById('eq4').value;
  times5 = document.getElementById('eq5').value;
  times6 = document.getElementById('eq6').value;
  times7 = document.getElementById('eq7').value;
  times8 = document.getElementById('eq8').value;
  times9 = document.getElementById('eq9').value;
  times10 = document.getElementById('eq10').value;
  times11 = document.getElementById('eq11').value;

  //Υπολογισμος SUM
  kcal = (times1 * 90) + (times2 * 120) + (times3 * 150) +
    (times4 * 60) + (times5 * 25) + (times6 * 80) + (times7 * 35)
    + (times8 * 55) + (times9 * 75) + (times10 * 100) + (times11
    * 45);
}
```

εικόνα 4.36

5. **public function fp_meals_save_postdata (\$post_id):** Βασικό μέρος της όλης διαδικασίας είναι και η αποθήκευση των δεδομένων μας που γίνεται με τη συνάρτηση αυτή (εικόνα 4.37). Η function αυτή καταχωρεί τα δεδομένα που προσθέσαμε στα προσαρμοσμένα πεδία σε εμφωλευμένη βάση δεδομένων του Wordpress (Custom Meta Boxes, 2021).

```
public function fp_meals_save_postdata ( $post_id ) {
    $is_autosave = wp_is_post_autosave($post_id);
    $is_revision = wp_is_post_revision($post_id);
    $is_valid_nonce = ( isset( $ _POST [ 'fp_meals_nonce' ] ) &&
    wp_verify_nonce( $ _POST [ 'fp_meals_nonce' ], basename( __FILE__ ) ) ) ?
    'true' : 'false';

    if( $is_autosave || $is_revision || !$is_valid_nonce ){
        return;
    }

    if( isset( $ _POST [ 'eq1' ] ) ){
        update_post_meta( $post_id, 'eq1', sanitize_text_field( $ _POST
        [ 'eq1' ] ) );
    }
}
```

εικόνα 4.37

6. **public function fp_register_fp_clients() :** Η συνάρτηση αυτή δημιουργεί το custom post types: Clients δηλαδή το πελατολόγιο που θα μπορεί να δημιουργεί ο χρήστης (εικόνα 4.38) (Custom Meta Boxes, 2021).

```
public function fp_register_fp_clients() {

    $labels = [
        'name' => _x( 'Clients', 'post type general name', 'Clients'
        ),
        'singular_name' => _x( 'Client', 'post type singular name', 'Client' )
        ,
        'menu_name' => _x( 'Clients', 'admin menu', 'Clients' ),
        'name_admin_bar' => _x( 'Client', 'add new on admin bar', 'Client' ),
        'add_new' => _x( 'Add New', 'Client', 'Client' ),
        'add_new_item' => __( 'Add New Client', 'Client' ),
        'new_item' => __( 'New Client', 'Client' ),
    ];
}
```

εικόνα 4.38

Τα δεδομένα που προκύπτουν από τον παραπάνω τύπο δημοσίευσης αποθηκεύονται σε εμφωλευμένη βάση δεδομένων του Wordpress. Τα δεδομένα αυτά δεν είναι δημόσια (“public” => false) (εικόνα 4.39) πράγμα που κάνει πιο δύσκολο σε κάποιον να τα εξαγάγει από το Web ή να έχει πρόσβαση σε αυτά παρά μόνο αν καταφέρει να έχει πρόσβαση στη βάση δεδομένων του server μας ή στον πίνακα ελέγχου της ιστοσελίδα μας. Με αυτόν τον τρόπο το custom post type Clients είναι μια ξεχωριστή εφαρμογή για τον χρήστη που του επιτρέπει να καταχωρεί το πελατολόγιο του μαζί με δείκτες ανθρωπομετρίας καθώς και το διαιτολόγιο του κάθε πελάτη.

```
$args = [  
    "label" => __( "Clients" ),  
    "labels" => $labels,  
    "description" => "",  
    "public" => false,  
    "show_ui" => true,  
    "has_archive" => false,  
    "show_in_menu" => false,  
    "show_in_nav_menus" => true,  
    "exclude_from_search" => false,  
    "capability_type" => "post",
```

εικόνα 4.39

Σε κάθε custom post type που δημιουργούμε, το Wordpress μας δίνει τη δυνατότητα να έχουμε κάποια πρότυπα – υποστηρικτές (**supports**) περιεχομένου. Στο συγκεκριμένο custom post type: Clients χρησιμοποιήσαμε 3 από αυτά τα supports (title, thumbnail, author) (Code Reference: post_type_supports, 2021) (εικόνα 4.40).

```
"supports" => [ 'title', 'thumbnail', 'author' ],  
"taxonomies" => [],
```

εικόνα 4.40

Δημιουργήσαμε και αρκετά προσαρμοσμένα πεδία καταχώρησης δεδομένων (custom fields) και θα αναλυθούν στις επόμενες συναρτήσεις (functions).

7. **public function fp_clients_pers_custom_box()** : Η συνάρτηση αυτή (function) δημιουργεί ένα προσαρμοσμένο πεδίο (εικόνα 4.41) που περιέχει **τα προσωπικά στοιχεία** του πελάτη (Custom Meta Boxes, 2021).

```
public function fp_clients_pers_custom_box() {
    $screens = [ 'fp-clients' ];
    foreach ( $screens as $screen ) {
        add_meta_box(
            'fp_clients_pers_id',
            __( 'Προσωπικά Στοιχεία' ),
            array ( $this, 'fp_clients_pers_custom_box_html' ),
            $screen,
            'normal'
        );
    }
}
```

εικόνα 4.41

8. **public function fp_clients_pers_custom_box_html(\$post)**: Η function αυτή (εικόνα 4.42) περιλαμβάνει όλο τον HTML, css και javascript κώδικα που προσθέσαμε για το πεδίο **Προσωπικά Στοιχεία** (Custom Meta Boxes, 2021).

```
public function fp_clients_pers_custom_box_html( $post ) {
    wp_nonce_field( basename( __FILE__ ), 'fp_clients_pers_nonce' );
    $fp_pers_stored_meta = get_post_meta( $post->ID );

    ?>

    <style>...
    </style>

    <div class="container">...
    </div>

    <?php
}
```

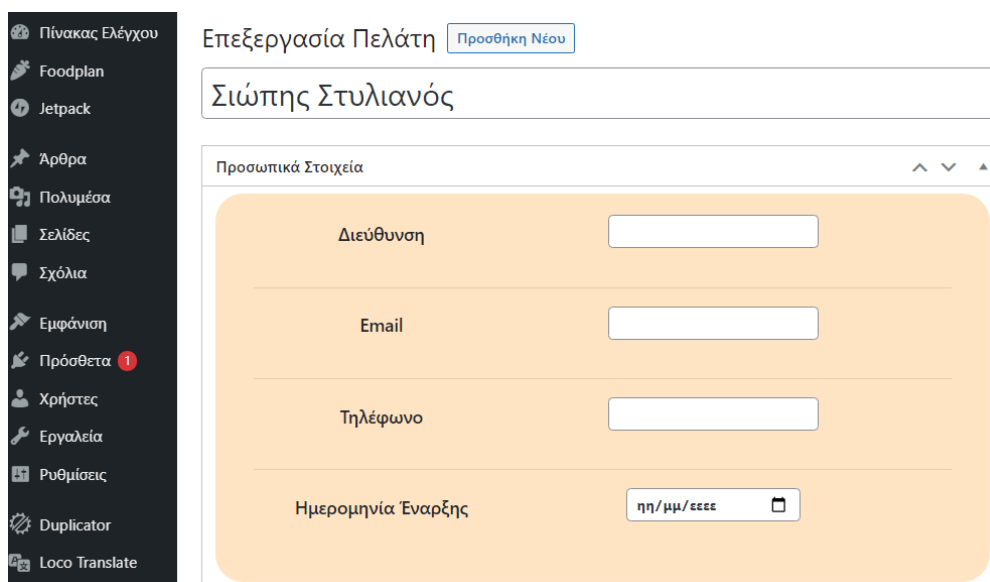
εικόνα 4.42

Στη συνάρτηση αυτή χρησιμοποιήσαμε βασικές ιδιότητες css. Επίσης, χρησιμοποιήσαμε συγκεκριμένες κλάσεις bootstrap ώστε να δημιουργήσουμε ένα πιο φιλικό περιβάλλον για το χρήστη (εικόνα 4.43).

```
<div class="container">
  <div class="col">
    <div class="col-12 form-group">
      <div class="row">
        <div class="col-4 col-form-label">
          <h6><label for="street"><?php _e('Διεύθυνση',
            'fp_domain'); ?></label></h6>
        </div>
        <div class="col-8">
          <input type="text" name="street_id" id="street_id"
            value="<?php if(!empty($fp_pers_stored_meta
              ['street_id'])) echo esc_attr($fp_pers_stored_meta
                ['street_id'][0]); ?>">
        </div>
      </div>
    </div><hr>
```

εικόνα 4.43

Αφού ολοκληρώσαμε τον κώδικα σχεδιασμού (HTML, CSS, bootstrap) έχουμε το παρακάτω οπτικό αποτέλεσμα σε περιβάλλον χρήστη (εικόνα 4.44).



εικόνα 4.44

9. **public function fp_clients_pers_save_postdata(\$post_id) :** Η συνάρτηση (function) αυτή (εικόνα 4.45) χρησιμοποιείται για την αποθήκευση του προσαρμοσμένου πεδίου Προσωπικά Στοιχεία (Custom Meta Boxes, 2021).

```
public function fp_clients_pers_save_postdata( $post_id ) {
    $is_autosave = wp_is_post_autosave($post_id);
    $is_revision = wp_is_post_revision($post_id);
    $is_valid_nonce = (isset($_POST['fp_clients_pers_nonce']) &&
    wp_verify_nonce($_POST['fp_clients_pers_nonce'], basename(__FILE__)))?
    'true' : 'false';
    if($is_autosave || $is_revision || !$is_valid_nonce){
        return;
    }

    if(isset($_POST['street_id'])){
        update_post_meta($post_id, 'street_id', sanitize_text_field($_POST
        ['street_id']));
    }
}
```

εικόνα 4.45

10. **public function fp_clients_metrics_custom_box():** η συνάρτηση αυτή (εικόνα 4.46) δημιουργεί ένα προσαρμοσμένο πεδίο (custom field) που περιέχει τα ανθρωπομετρικά χαρακτηριστικά του πελάτη (**Ανθρωπομετρία**) (Custom Meta Boxes, 2021).

```
// Custom field: Anthropometry

public function fp_clients_metrics_custom_box() {
    $screens = [ 'fp-clients' ];
    foreach ( $screens as $screen ) {
        add_meta_box(
            'fp_clients_metrics_id',
            __( 'Ανθρωπομετρία' ),
            array ( $this, 'fp_clients_metrics_custom_box_html' ),
            $screen,
            'normal'
        );
    }
}
```

εικόνα 4.46

11. public function fp_clients_metrics_custom_box_html(\$post): Η συνάρτηση αυτή (εικόνα 4.47) δημιουργεί όλο το περιεχόμενο του παραπάνω custom field και περιλαμβάνει όλο τον HTML, css και javascript κώδικα που προσθέσαμε για αυτό το πεδίο (Custom Meta Boxes, 2021).

```
public function fp_clients_metrics_custom_box_html( $post ) {
    wp_nonce_field(basename(__FILE__), 'fp_clients_metrics_nonce');
    $fp_metrics_stored_meta = get_post_meta( $post->ID);
    $fp_sex_stored_meta = get_post_meta( $post->ID,
    'fp_clients_metrics_custom_box', true );

    ?>

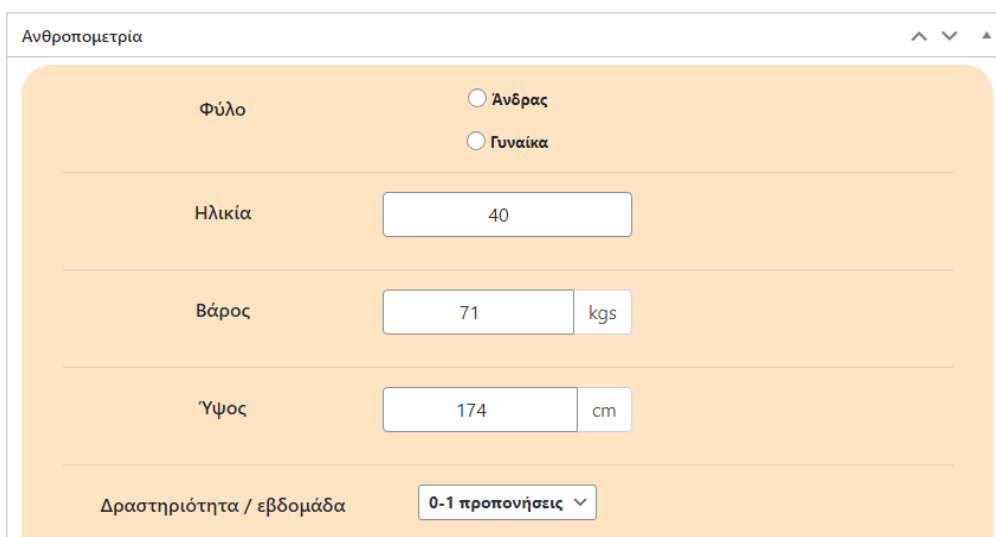
    <script>...
    </script>

    <div class="container">...
    </div>

    <?php
}
```

εικόνα 4.47

Τα προσαρμοσμένα πεδία που προσθέσαμε είναι το φύλο, το βάρος, το ύψος, η ηλικία και η δραστηριότητα (εικόνα 4.48).

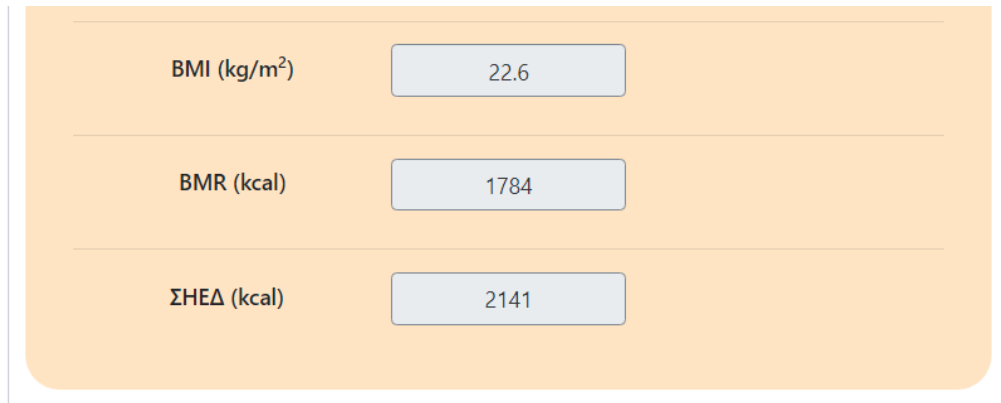


The image shows a web form titled "Ανθρωπομετρία" (Anthropometry) with a light orange background. It contains the following fields:

- Φύλο** (Gender): Radio buttons for "Ανδρας" (Male) and "Γυναίκα" (Female).
- Ηλικία** (Age): A text input field containing the number "40".
- Βάρος** (Weight): A text input field containing "71" and a unit dropdown menu set to "kgs".
- Ύψος** (Height): A text input field containing "174" and a unit dropdown menu set to "cm".
- Δραστηριότητα / εβδομάδα** (Activity / week): A dropdown menu set to "0-1 προπονήσεις" (0-1 sessions).

εικόνα 4.48

Επίσης προστέθηκαν κάποιοι ανθρωπομετρικοί δείκτες (εικόνα 4.49) όπως το BMI (Δείκτης Μάζας Σώματος), το BMR (Βασικός Μεταβολικός Ρυθμός) και ΣΗΕΔ (Συνολική Ημερήσια Ενεργειακή Δαπάνη) που προκύπτουν από τη προσθήκη των παραπάνω δεδομένων που καταχωρούμε και με τη χρήση javascript και θα αναλυθούν παρακάτω.



BMI (kg/m ²)	22.6
BMR (kcal)	1784
ΣΗΕΔ (kcal)	2141

εικόνα 4.49

Ο **Δείκτης Μάζας Σώματος (BMI)** είναι ο πιο απλός τρόπος για τον προσδιορισμό της κατάστασης του βάρους σύμφωνα με την παρακάτω εξίσωση: $BMI = \text{Βάρος (kg)} / \text{Υψος}^2 \text{ (m)}$ (Παπανικολάου Γ., 2005). Το συγκεκριμένο αποτέλεσμα προκύπτει με μια function σε γλώσσα javascript (εικόνα 4.50).

```
function fp_bmi() {  
  
    var weight = document.getElementById("weight").value;  
    var height = document.getElementById("height").value;  
    console.log(weight);  
    console.log(height);  
  
    ll = weight / (height / 100) / (height / 100);  
    zz = ll.toFixed(1);  
    document.getElementById("bmi").value = zz;  
    fp_bmr()  
}
```

εικόνα 4.50

Επίσης σημαντικός μαθηματικός τύπος είναι ο **Δείκτης Βασικού Μεταβολικού Ρυθμού (BMR)**. Ο BMR προσδιορίζει την ενέργεια που καταναλώνουμε ημερησίως (χωρίς την ενέργεια που δαπανάται λόγω φυσικής δραστηριότητας) και προκύπτει σύμφωνα με την παρακάτω εξίσωση (Παπανικολάου Γ., 2005):

Για τους άντρες: $BMR = 88 + (13.4 \times B) + (4.8 \times Y) - (5.7 \times H)$

Για τις γυναίκες: $BMR = 448 + (9.2 \times B) + (3.1 \times Y) - (4.3 \times H)$

όπου B= Βάρος σε kg, Y= Ύψος σε cm, H= Ηλικία

Το συγκεκριμένο αποτέλεσμα προκύπτει με μια function σε γλώσσα javascript. (εικόνα 4.51).

```
function fp_bmr() {
  var bmr;
  var weight = document.getElementById("weight").value;
  var height = document.getElementById("height").value;
  var age = document.getElementById("age").value;
  var sex = document.getElementById("sex_id").value;
  console.log(weight);
  console.log(height);
  console.log(age);
  console.log(sex);
  if (document.getElementById("sex_id").checked) {
    bmr = 88 + ( 13.4 * weight ) + ( 4.8 * height ) - ( 5.7 * age
  ) }
  else {
    bmr = 448 + ( 9.2 * weight ) + ( 3.1 * height ) - ( 4.3 * age )
  }
}
```

εικόνα 4.51

Ο **Συντελεστής Δραστηριότητας (ΣΔ)** είναι ένας δείκτης που εκφράζει την ημερήσια σωματική δραστηριότητα και χρησιμοποιείται για την εκτίμηση της Συνολικής Ενεργειακής Δαπάνης ενός ατόμου σε συνδυασμό με τον Βασικό Μεταβολικό Ρυθμό (BMR). (Williams, M.H., 2003) Ένας εύκολος τρόπος κατηγοριοποίησης του ΣΔ είναι ο παρακάτω:

0-1 φορές προπόνηση / εβδομάδα: $\Sigma\Delta = 1,2$

1-3 φορές προπόνηση / εβδομάδα: $\Sigma\Delta = 1,35$

3-5 φορές προπόνηση / εβδομάδα: $\Sigma\Delta = 1,55$

καθημερινή προπόνηση: $\Sigma\Delta = 1,725$

πρωταθλητισμός: $\Sigma\Delta = 1,9$

Ο συγκεκριμένος συντελεστής δημιουργήθηκε με κώδικα HTML με tag <select> (εικόνα 4.52).

```
<select name="activity" id="activity"
onchange="fp_shed()">
  <option value="nothing" <?php selected(
$activity_from_db, "nothing" ) ?><?php _e('0-1
προπονήσεις', 'fp_domain'); ?>
</option>
  <option value="mild" <?php selected(
$activity_from_db, "mild" ) ?><?php _e('1-3
προπονήσεις', 'fp_domain'); ?>
</option>
  <option value="moderate" <?php selected(
$activity_from_db, "moderate" ) ?><?php _e('3-5
προπονήσεις', 'fp_domain'); ?>
</option>
```

εικόνα 4.52

Η **Συνολική Ημερήσια Ενεργειακή Δαπάνη (ΣΗΕΔ)** είναι η ενέργεια που καταναλώνουμε καθημερινά, δηλαδή η ποσότητα θερμίδων που χρειαζόμαστε για να διατηρούμε το βάρος μας. Η ΣΗΕΔ προκύπτει από την παρακάτω απλή εξίσωση:

$ΣΗΕΔ = ΣΔ \times BMR$ (Williams, M.H., 2003). Το συγκεκριμένο αποτέλεσμα προκύπτει με μια function σε γλώσσα javascript. (εικόνα 4.53).

```
function fp_shed() {
  var shed;
  var bmr = document.getElementById("bmr").value;
  var activity = document.getElementById("activity");
  var activities = activity.options[activity.selectedIndex].value;
  console.log(bmr);
  console.log(activity);
  if (activities == "nothing") {shed = bmr * 1.2
} else if (activities == "mild") {shed = bmr * 1.35
} else if (activities == "moderate") {shed = bmr * 1.55
} else if (activities == "high") {shed = bmr * 1.725
} else {shed = bmr * 1.9
}
}
shedfixed = shed.toFixed(0);
document.getElementById("shed").value = shedfixed;
}
```

εικόνα 4.53

12. **public function fp_clients_metrics_save_postdata(\$post_id)** : Η function αυτή (εικόνα 4.54) χρησιμοποιείται για την αποθήκευση των δεδομένων του custom field Ανθρωπομετρία (Custom Meta Boxes, 2021).

```
public function fp_clients_metrics_save_postdata( $post_id ) {
    $is_autosave = wp_is_post_autosave($post_id);
    $is_revision = wp_is_post_revision($post_id);
    $is_valid_nonce = (isset($_POST['fp_clients_metrics_nonce']) &&
        wp_verify_nonce($_POST['fp_clients_metrics_nonce'],
            'fp_clients_metrics_custom_box'))? 'true' : 'false';

    if($is_autosave || $is_revision || !$is_valid_nonce){
        return;
    }

    if(isset($_POST['bmi'])){
        update_post_meta($post_id, 'bmi', sanitize_text_field($_POST['bmi']));
    }
}
```

εικόνα 4.54

13. **public function fp_diet_custom_box()**: Η function αυτή (εικόνα 4.55) δημιουργεί ένα προσαρμοσμένο πεδίο: **Διαιτολόγιο** που δίνει τη δυνατότητα στον χρήστη να επιλέγει διαιτολόγιο το οποίο προκύπτει από το προσαρμοσμένα τύπο δημοσίευσης (custom post type) diet (Custom Meta Boxes, 2021)..

```
public function fp_diet_custom_box() {
    $screens = [ 'fp-clients' ];
    foreach ( $screens as $screen ) {
        add_meta_box(
            'fp_diet_id', // Unique ID
            __('Εβδομαδιαίο Διαιτολόγιο'), // Box title
            array ($this, 'fp_diet_custom_box_html'), // Content callback,
            // must be of type callable
            $screen // Post type
        );
    }
}
```

εικόνα 4.55

14. public function fp_diet_custom_box_html(\$post): Η συνάρτηση αυτή (εικόνα 4.56) περιλαμβάνει όλο τον HTML, css και javascript κώδικα που προσθέσαμε για αυτό το πεδίο (Custom Meta Boxes, 2021).

```
public function fp_diet_custom_box_html( $post ) {
    wp_nonce_field(basename(__FILE__), 'fp_diet_nonce');
    $fp_diet_stored_meta = get_post_meta($post->ID);

    $breakf = array('post_type' => 'diet');
    $query4 = new WP_Query( $breakf );

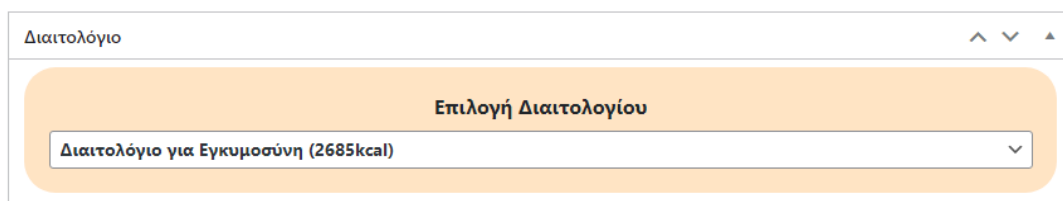
    ?>

    <div class="container">...
    </div>

    <?php
}
```

εικόνα 4.56

Στο πεδίο αυτό ο χρήστης έχει τη δυνατότητα να επιλέξει 1 διαιτολόγιο με βάση τον τίτλο που έχουμε δώσει σε αυτό π.χ. Διαιτολόγιο για Εγκυμοσύνη καθώς και με βάση τις θερμίδες του διαιτολογίου που έχει προσδιορίσει ο χρήστης στο Custom Post Type: Diet (εικόνα 4.57).



Διαιτολόγιο

Επιλογή Διαιτολογίου

Διαιτολόγιο για Εγκυμοσύνη (2685kcal)

εικόνα 4.57

Οι δύο αυτές δυνατότητες επιλογής διαιτολογίου προκύπτουν από τον HTML κώδικα με τη χρήση ενός tag <select> με while loop (εικόνα 4.58).

```
<select name="fp_diet_break1" id="fp_diet_break1"
style="text-align: center;" class="widefat">
  <option value = "0" disabled selected><?php _e('Επιλέξτε
διαιτολόγιο...', 'fp-domain');?></option>

  <?php while ( $query4->have_posts() ) :
  $query4->the_post();?>
  <?php $break1_db = get_post_meta( get_the_ID(), 'kcal',
true); ?>
  <?php $post_title = get_the_title(); ?>

  <option value="<?php echo $post_title; ?>" <?php
selected( $break1_from_db, $post_title ); ?>>
  <?php echo $post_title; ?> (<?php echo $break1_db; ?
>kcal)
  </option>

  <?php endwhile; wp_reset_postdata();?>
```

εικόνα 4.58

15. public function fp_diet_save(\$post_id): Η συνάρτηση αυτή χρησιμεύει για την αποθήκευση των δεδομένων για το custom field: Διαιτολόγιο (εικόνα 4.59) (Custom Meta Boxes, 2021).

```
public function fp_diet_save($post_id){
    $is_autosave = wp_is_post_autosave($post_id);
    $is_revision = wp_is_post_revision($post_id);
    $is_valid_nonce = (isset($_POST['fp_diet_nonce']) && wp_verify_nonce($_POST
['fp_diet_nonce'], basename(__FILE__)))? 'true' : 'false';

    if($is_autosave || $is_revision || !$is_valid_nonce){
        return;
    }
    if(isset($_POST['fp_diet_break1']))){
        $thesave = $_POST['fp_diet_break1'];
        update_post_meta($post_id, 'fp_diet_break1', $thesave);
    }
}
```

εικόνα 4.59

Εδώ ολοκληρώσαμε όλα τα προσαρμοσμένα πεδία (custom filed) που προσθήσαμε στο προσαρμοσμένο τύπο δημοσίευσης (custom post type) clients.

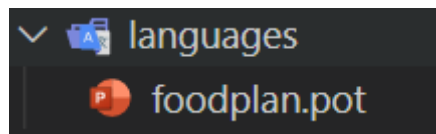
16. public function content_single_diet (\$the_content): η συνάρτηση αυτή (εικόνα 4.60) φιλτράρει το περιεχόμενο του custom post type Diet και χρησιμοποιείται για το αποτέλεσμα του shortcode.

```
// Filter content for CPT: Diet
public function content_single_diet( $the_content ) {
    // filter contents for just Diet
    if ( in_the_loop() && is_singular( 'diet' ) ) {
        ob_start();
        ?>
        <?php echo get_the_content(); ?>
        <?php include FOODPLAN_BASE_DIR . 'templates/diet-content.php';?>
        <?php
        return ob_get_clean();
    }
    return $the_content;
}
```

εικόνα 4.60

4.2.1.5 Φάκελος *languages*

Ο φάκελος αυτός χρησιμοποιείται για να γίνει η μετάφραση του πρόσθετου. Στον φάκελο αυτό υπάρχει αρχικά ένα άδειο αρχείο `foodplan.pot` (εικόνα 4.61). Το αρχείο αυτό χρησιμοποιείται από το πρόσθετό μας με τη βοήθεια μια κλάσης `class Foodplan_i18n` που έχουμε προαναφέρει (σελ.18).



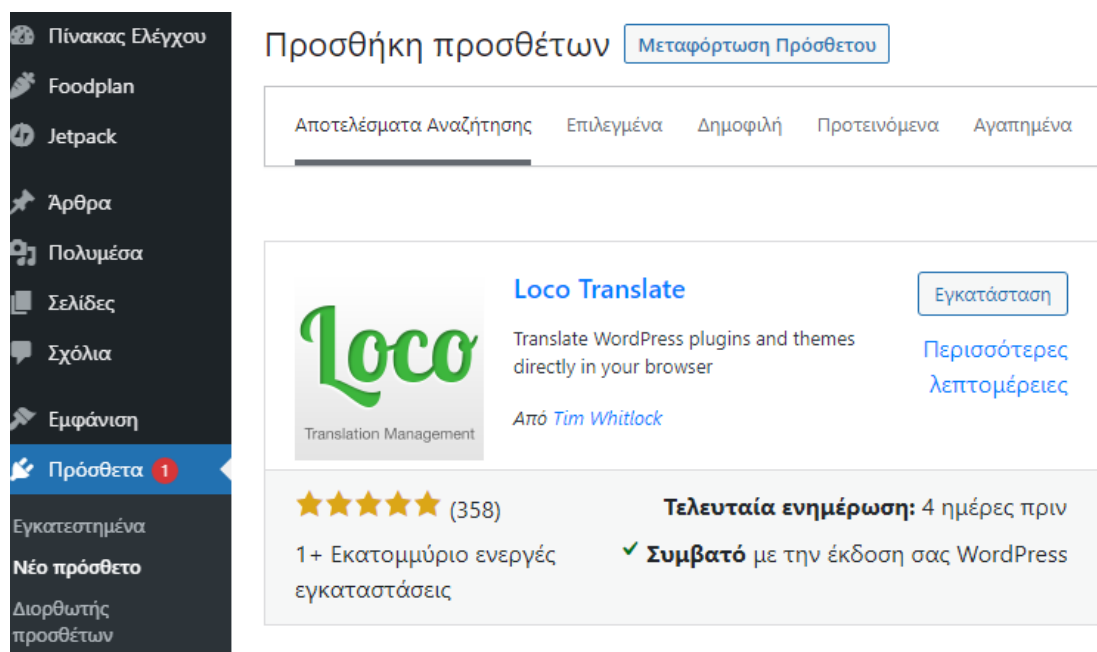
εικόνα 4.61

Ο τρόπος με τον οποίο θέτουμε μία λέξη προς μετάφραση στο Wordpress γίνεται με τη βοήθεια ειδικών συναρτήσεων. Αυτές που χρησιμοποιήσαμε στο δικό μας πρόσθετο είναι οι συναρτήσεις `__()` και `_e()` που επιστρέφουν ή επαναλαμβάνουν τη μεταφρασμένη συμβολοσειρά. Η χρήση τους είναι προφανής: μία συμβολοσειρά, μία μετάφραση. Επίσης χρησιμοποιήσαμε και τη συνάρτηση `_x()` όταν η λέξη που θέτουμε προς μετάφραση μπορεί υπάρχει και σε άλλα σημεία του κώδικά μας (εικόνα 4.62) (How to Internationalize Your Plugin, 2021).

```
_x( 'Clients', 'admin menu', 'foodplan' ),  
_x( 'Client', 'add new on admin bar', 'foodplan' ),  
_x( 'Add New', 'Client', 'foodplan' ),  
__( 'Add New Client', 'foodplan' ),  
__( 'New Client', 'foodplan' ),  
__( 'Edit Client', 'foodplan' ),
```

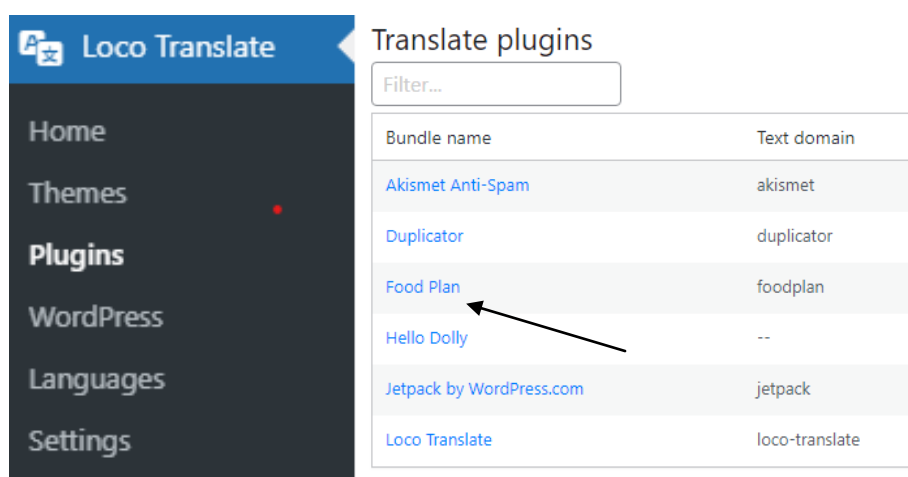
εικόνα 4.62

Για τη μετάφραση του πρόσθετου χρησιμοποιήσαμε ένα επιπλέον πρόσθετο στον ιστότοπό μας με ονομασία Loco translate (εικόνα 4.63) (Πρόσθετα: Loco translate, 2021).



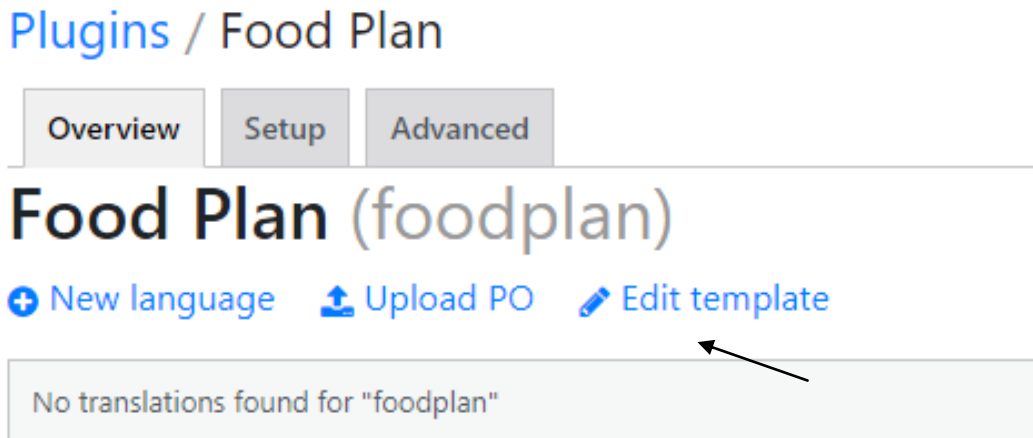
εικόνα 4.63

Εγκαταστήσαμε και ενεργοποιήσαμε το πρόσθετο αυτό στον πίνακα ελέγχου. Παρατηρούμε στο admin menu ότι δημιουργήθηκε μία επιπλέον ετικέτα Loco Translate. Επιλέγουμε από την ετικέτα αυτή την επιλογή plugin και επιλέγουμε το foodplan για να το μεταφράσουμε (εικόνα 4.64).



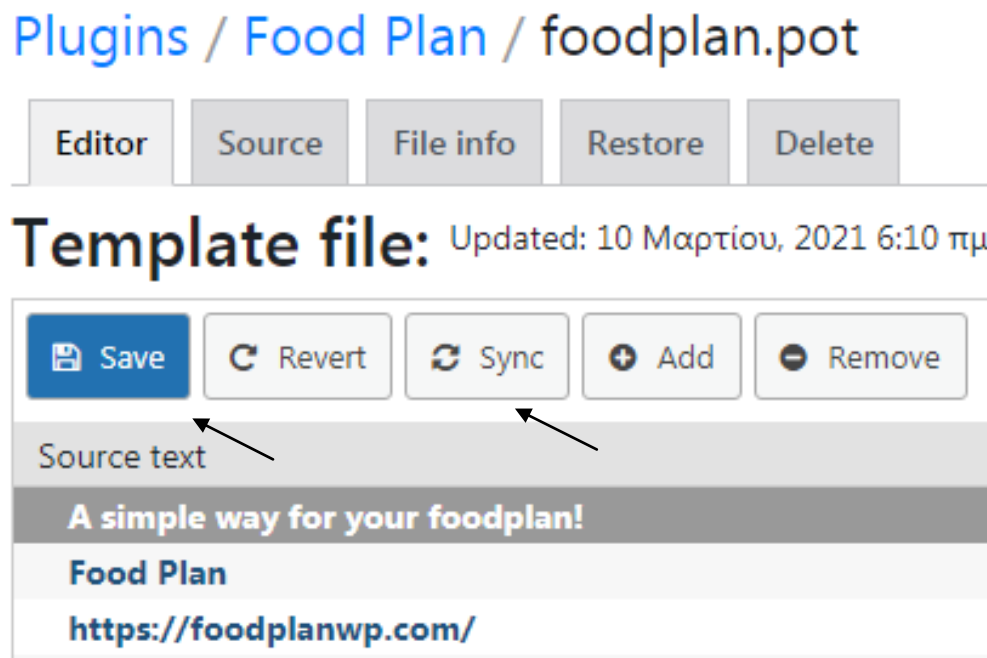
εικόνα 4.64

Συνεχίζοντας με το Loco Translate δημιουργούμε ένα template στο οποίο θα καταχωρούνται οι λέξεις που έχουμε θέσει στο πρόσθετό μας προς μετάφραση (εικόνα 4.65).



εικόνα 4.65

Δημιουργήσαμε το template και για να συγχρονίσουμε το πρόσθετό μας με το Loco Translate πατάμε στο κουμπί sync και μετά save (εικόνα 4.66).



εικόνα 4.66

Με αυτόν τον τρόπο αν επιστρέψουμε πίσω στον κώδικά μας θα παρατηρήσουμε ότι οι εγγραφές που είναι προς μετάφραση έχουν καταχωρηθεί αυτόματα στο αρχείο foodplan.pot. Η μετάφραση στα ελληνικά των καταχωρήσεων αυτών γίνεται με την επιλογή «προσθήκη γλώσσας» (εικόνα 4.65). Η επιλογή αυτή μας μεταφέρει σε ένα καινούριο παράθυρο (εικόνα 4.67) όπου επιλέγουμε την γλώσσα : Ελληνικά και τον φάκελο στον οποίο θα αποθηκευτεί το αρχείο foodplan-el.po που θα μεταφράζει τις καταχωρήσεις μας στα ελληνικά. Με τον ίδιο τρόπο μπορούμε να μεταφράσουμε το πρόσθετό μας σε οποιαδήποτε άλλη γλώσσα θέλουμε.

Plugins / Food Plan / New language

Overview Setup **Advanced**

Notice: This file may be overwritten or deleted when you update WordPress

Initializing new translations in "foodplan"

4 strings found in foodplan.pot

1. Choose a language:

WordPress language: Custom language:

EA

2. Choose a location:

[? What's this?](#)

Custom: languages/loco/plugins/foodplan-el.po

System: languages/plugins/foodplan-el.po

Author: plugins/foodplan/languages/foodplan-el.po

[Start translating](#)

εικόνα 4.67

Πατώντας το πλήκτρο Έναρξη μετάφρασης (Start translating) το πρόσθετο Loco Translate δημιουργεί το προς μετάφραση αρχείο foodplan-el.po (εικόνα 4.68).

Plugins / Food Plan / foodplan-el.po

EA Greek Updated: 14 Μαρτίου, 2021 9:35 πμ – 96% translated, 77 strings (3 untranslated)

Save Revert Sync Auto Filter translations

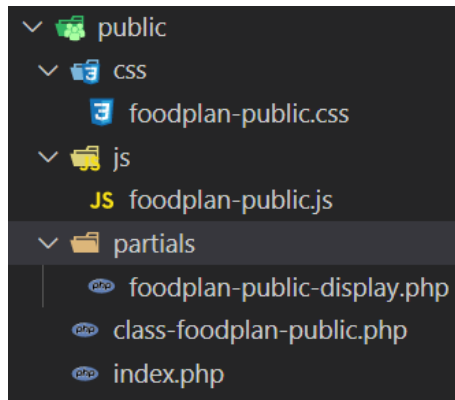
Source text	Translation
Add New Diet	Προσθήκη Νέου Διατολογίου
Client Add New	Προσθήκη Νέου
Diet Add New	Προσθήκη Νέου
Adress	Διεύθυνση
All Clients	Όλοι οι Πελάτες
All Diet	Όλα τα Διαιτολόγια
BMI (kg/m2)	ΒΜΙ (kg/m2)

εικόνα 4.68

Στο φάκελο languages του πρόσθετού μας παρατηρούμε ότι έχει προστεθεί το παραπάνω αρχείο και με τον τρόπο αυτό ολοκληρώνουμε τη μετάφραση του πρόσθετού μας στα ελληνικά.

4.2.1.6 Φάκελος *public*

Ο φάκελος αυτός περιλαμβάνει όλα τα αρχεία και τις κλάσεις που είναι δημόσια και θα χρησιμοποιούνται για το οπτικό περιβάλλον της ιστοσελίδας μας. Η δομή του (εικόνα 4.69) είναι παρόμοια με τον φάκελο `admin`.



εικόνα 4.69

Περιλαμβάνει τον φάκελο `css` με βασικό αρχείο `foodplan-public.css` το οποίο περιέχει όλη τη `css` που χρησιμοποιήσαμε σε επίπεδο `public`. Επίσης, χρησιμοποιήσαμε και `flexbox css` (εικόνα 4.70) που μας δίνει περισσότερες δυνατότητες σχεδιασμού πέραν από τις βασικές σε προηγούμενα παραδείγματα.

```
/* Flexbox stuff */  
  
.meal-entry-content {  
  display: flex;  
  align-content: center;  
  align-items: center;  
  align-self: center;  
  padding: 1.5em;  
  flex-direction: column;  
  text-align: center;  
  font-family: Tahoma, Verdana, sans-serif;  
  font-style: oblique;  
  font-size: 24px;  
  font-weight: normal;  
  line-height: 1.7;  
  text-shadow: 1px 1px 2px rgb(113, 200, 221);  
}
```

εικόνα 4.70

Επίσης, ο φάκελος js περιέχει ένα αρχείο javascript. Στο συγκεκριμένο πρόσθετο δεν χρησιμοποιήσαμε κωδικα javascript για «public» περιεχόμενο οπότε το αρχείο αυτό είναι άδειο. Το ίδιο ισχύει και για τον φάκελο partials.

Τέλος, υπάρχει ένα αρχείο **class-foodplan-public.php** που περιέχει μία κλάση class Foodplan_public. Οι βασικές συναρτήσεις (functions) της κλάσης αυτής είναι η **public function enqueue_styles()** και **public function enqueue_scripts()** που ουσιαστικά «φορτώνουν» τα αρχεία css και js σε επίπεδο «public». Τέλος, υπάρχει και ένα αρχείο **index.php** με ένα απλό σχόλιο php για λόγους ασφάλειας του πρόσθετου.

4.2.1.7 Φάκελος *templates*

Ο φάκελος αυτός περιέχει ένα αρχείο `diet-content.php` που δημιουργήσαμε και ουσιαστικά αποτελεί το περιεχόμενο που θέλουμε να αποδώσουμε στο site μας από το custom post type `Diet` που δημιουργήσαμε σε επίπεδο `admin`. Το αρχείο αυτό είναι το αρχείο – γέφυρα που μεταφέρει ουσιαστικά τη πληροφορία που θα του υποδείξουμε από τον `admin` στο `public` επίπεδο. Πιο συγκεκριμένα, μεταφέρουμε τη πληροφορία του περιεχομένου του `editor` (δηλαδή το διαιτολόγιο) του custom post type `Diet` καθώς και τη θρεπτική αξία του διαιτολογίου (εικόνα 4.71).

```
<div class="meal-entry-content">
  <?php

  $meta_fields = array(
    'kcal' => __( 'Θερμίδες', 'foodplan' ),
    'carb' => __( 'gr Υδατάνθρακες', 'foodplan' ),
    'pro' => __( 'gr Πρωτεΐνες', 'foodplan' ),
    'fat' => __( 'gr Λίπη', 'foodplan' ),
  );

  $meta_icons = array(
    'kcal' => '<i class="fab fa-nutritionix"></i>',
  );

  $html = '';

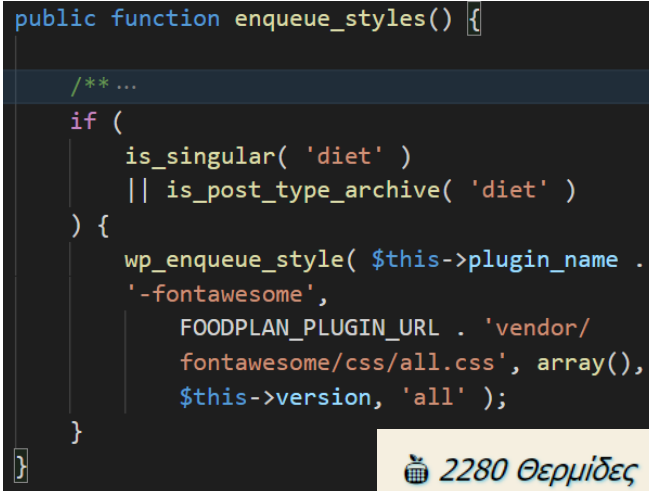
  foreach ( $meta_fields as $meta_key => $label ) {
    // first : meta value
    $value = esc_html(
      get_post_meta(
        get_the_ID(),
        $meta_key,
```

εικόνα 4.71

4.2.1.8 Φάκελος *vendor*

Στον φάκελο αυτό προσθέτουμε οποιοδήποτε εξωτερικό αρχείο θέλουμε να χρησιμοποιήσουμε για το πρόσθετό μας. Στο πρόσθετο αυτό χρησιμοποιήσαμε **παραδειγματικά** έναν φάκελο *fontawesome* όπου μας δίνει τη δυνατότητα να χρησιμοποιήσουμε γραμματοσειρές και εικονίδια που υπάρχουν διαθέσιμα μόνο στο web (Fontawesome, 2021). Τα αρχεία αυτά χρησιμοποιούνται μέσω της συνάρτησης **public function enqueue_styles ()** στη κλάση *class Foodplan_public* (εικόνα 4.72).

```
public function enqueue_styles() {  
  
    /** ...  
  
    if (  
        is_singular( 'diet' )  
        || is_post_type_archive( 'diet' )  
    ) {  
        wp_enqueue_style( $this->plugin_name .  
            '-fontawesome',  
                FOODPLAN_PLUGIN_URL . 'vendor/  
                fontawesome/css/all.css', array(),  
                $this->version, 'all' );  
    }  
}
```



εικόνα 4.72

Ολοκληρώσαμε το περιεχόμενο των υποφακέλων του plugin και συνεχίζουμε στα επιμέρους μονήρη αρχεία του πρόσθετου.

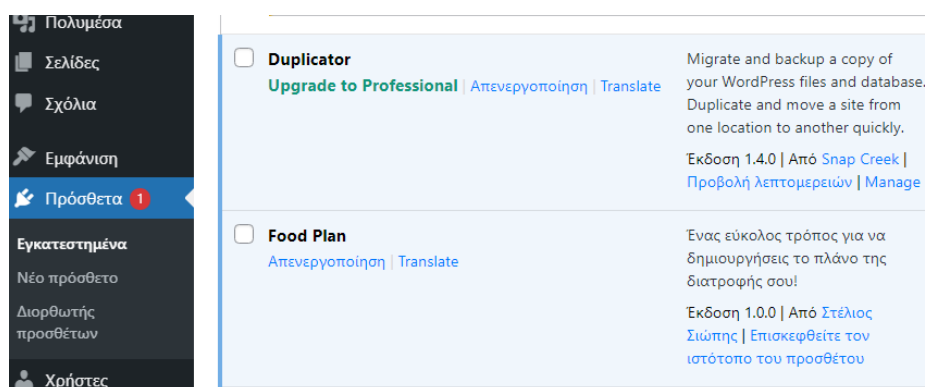
4.2.1.9 Αρχείο *foodplan.php*

Το αρχείο αυτό είναι ο «ακρογωνιαίος λίθος» του Foodplan. Περιλαμβάνει αρχικά όλα τα στοιχεία του πρόσθετού μας, όπως το όνομα, την έκδοση, τον συγγραφέα κτλ. Αυτά βρίσκονται σε μορφή σχολίων php (εικόνα 4.73) (Header Requirements, 2021).

```
* @wordpress-plugin
* Plugin Name:      Food Plan
* Plugin URI:      https://foodplanwp.com/
* Description:     A simple way for your foodplan!
* Version:         1.0.0
* Author:          Stelios Siopis
* Author URI:     https://foodplanwp.com/
* License:         GPL-2.0+
* License URI:    http://www.gnu.org/licenses/gpl-2.0.txt
* Text Domain:    foodplan
* Domain Path:    /languages
*/
```

εικόνα 4.73

Σχεδιάζοντας με τον Generator όλες αυτές οι πληροφορίες υπάρχουν ήδη στον φάκελο *foodplan* και μπορεί εύκολα ο συγγραφέας να τις επεξεργαστεί και να κερδίσει χρόνο για τη δημιουργία του πρόσθετου. Στη προκειμένη περίπτωση το μόνο που διαμορφώσαμε είναι η περιγραφή (Description). Όλα αυτά τα σχόλια αναγνωρίζονται από το Wordpress στο περιβάλλον του χρήστη στα πρόσθετα (εικόνα 4.74).



εικόνα 4.74

Το αρχείο αυτό περιλαμβάνει σημαντικό μερίδιο για την ασφάλεια του πρόσθετου. Σε πρώτο επίπεδο ασφάλειας αν κάποιος θα ήθελε να έχει πρόσβαση σε ένα ιστότοπο μέσω του πρόσθετου μας συνήθως θα έψαχνε μια άμεση πρόσβαση στο βασικό αρχείο του κώδικά μας, δηλαδή το foodplan.php. Αυτό μπορεί να αποφευχθεί αν κάποιος θα ήθελε να έχει άμεση πρόσβαση στο αρχείο αυτό (εικόνα 4.74) (Plugin Security, 2021).

```
// If this file is called directly, abort.
if ( ! defined( 'WPINC' ) ) {
    die;
}
```

εικόνα 4.74

Περιέχει τις βασικότερες συναρτήσεις του πρόσθετου (functions) που είναι:

1. **function activate_foodplan():** Η συνάρτηση αυτή ενεργοποιεί το πρόσθετό μας και «φορτώνει» την κλάση Foodplan_Activator (εικόνα 4.75).

```
/**
 * The code that runs during plugin activation.
 * This action is documented in includes/class-foodplan-activator.php
 */
function activate_foodplan() {
    require_once plugin_dir_path( __FILE__ ) . 'includes/class-foodplan-activator.php';
    Foodplan_Activator::activate();
}
```

εικόνα 4.75

2. **function deactivate_foodplan():** Η συνάρτηση αυτή απενεργοποιεί το πρόσθετο και «φορτώνει» την κλάση Foodplan_Deactivator (εικόνα 4.76).

```
/**
 * The code that runs during plugin deactivation.
 * This action is documented in includes/class-foodplan-deactivator.php
 */
function deactivate_foodplan() {
    require_once plugin_dir_path( __FILE__ ) . 'includes/class-foodplan-deactivator.php';
    Foodplan_Deactivator::deactivate();
}
```

εικόνα 4.76

3. **function run_foodplan ():** Η συνάρτηση αυτή δίνει την εντολή να εκτελεστεί το πρόσθετο (εικόνα 4.77).

```
/**
 * Begins execution of the plugin.
 *
 * Since everything within the plugin is registered via hooks,
 * then kicking off the plugin from this point in the file does
 * not affect the page life cycle.
 *
 * @since 1.0.0
 */
function run_foodplan() {
    $plugin = new Foodplan();
    $plugin->run();
}
run_foodplan();
```

εικόνα 4.77

Τέλος, κύρια λειτουργία του αρχείου αυτού είναι ότι «φορτώνει» τη βασική κλάση του foodplan (εικόνα 4.78) που αναλύσαμε στον φάκελο includes.

```
/**
 * The core plugin class that is used to define internationalization,
 * admin-specific hooks, and public-facing site hooks.
 */
require plugin_dir_path( __FILE__ ) . 'includes/class-foodplan.php';
```

εικόνα 4.78

4.2.1.10 Αρχείο LICENSE.txt

Το αρχείο LICENSE.txt είναι ένα αρχείο κειμένου που περιέχει ένα αντίγραφο της άδειας στην οποία υπόκειται το πρόσθετό μας και είναι γνωστή ως GPL-2 (εικόνα 4.79) (GNU Public License, 2021). Το αρχείο αυτό είναι απαραίτητο για την έκδοση του πρόσθετου στην επίσημη σελίδα του Wordpress για πρόσθετα (Plugins, 2021).

```
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
```

εικόνα 4.79

4.2.1.11 Αρχείο README.txt

Το αρχείο README.txt που περιλαμβάνει επιπλέον πληροφορίες για το πρόσθετό μας σε μορφή κειμένου. Εδώ μπορούμε να προσθέτουμε οδηγίες για τον χρήστη για να χρησιμοποιεί με τον καλύτερο τρόπο το πρόσθετό μας (Detailed Plugin Guidelines, 2021). Το αρχείο αυτό περιλαμβάνει μια μικρή περίληψη του πρόσθετου καθώς και όποιες πληροφορίες θέλουμε να μεταφέρουμε στον χρήστη μέσω αρχείου text. (εικόνα 4.80).

```
=== FoodplanWP ===
Contributors: Stsiopis
Donate link: https://foodplanwp.com/
Tags: comments, spam
Requires at least: 3.0.1
Tested up to: 3.4
Stable tag: 4.3
License: GPLv2 or later
License URI: http://www.gnu.org/licenses/gpl-2.0.html

== Short description ==

FoodplanWP is a Wordpress plugin for dietitians! Allows you to manage clientele, anthropometry as well as create calorie-counted nutrition programs based on food equivalents.
```

εικόνα 4.80

Το αρχείο αυτό απαιτείται για μια πιο επαγγελματική παρουσίαση ενός πρόσθετου καθώς και για την έκδοσή του στην επίσημη σελίδα του WordPress για πρόσθετα (Plugins, 2021). Για την έκδοσή του Foodplan στη σελίδα αυτή θα πρέπει να προσθέσουμε το αρχείο αυτό σε ένα επικυρωτή του Wordpress (Readme Validator, 2021) και να ακολουθήσουμε μία συγκεκριμένη διαδικασία που απαιτείται από το Wordpress (Developer Information, 2021).

4.2.1.12 Αρχείο *uninstall.php*

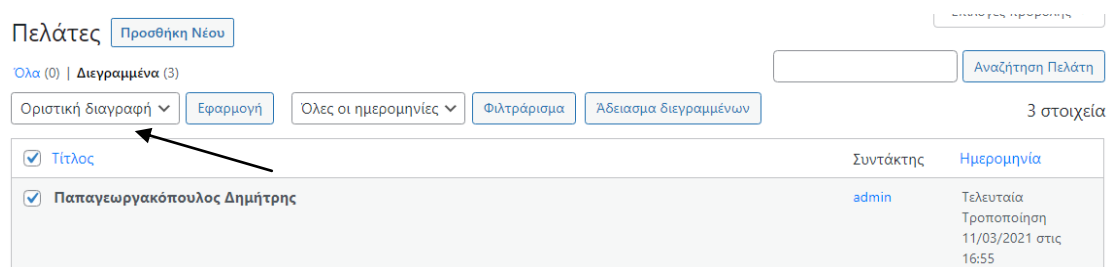
Το αρχείο `uninstall.php` είναι απαραίτητο γιατί διαγράφει οτιδήποτε έχει δημιουργήσει το πρόσθετό μας στη βάση δεδομένων του Wordpress σε περίπτωση διαγραφής του πρόσθετου από το περιβάλλον του χρήστη (Uninstall Methods, 2021). Στο συγκεκριμένο πρόσθετο κάνοντας μία αναζήτηση στη βάση δεδομένων της ιστοσελίδας μας που φιλοξενεί το πρόσθετο μέσω του phpMyAdmin παρατηρούμε ότι οι φάκελοι του Wordpress στη βάση μας που επηρεάζονται είναι ο φάκελος `postmeta` και ο φάκελος `posts` (εικόνα 4.81).



79 αποτελέσματα στο	<code>wp_</code>	<code>_postmeta</code>	Περιήγηση	Διαγραφή
15 αποτελέσματα στο	<code>wp_</code>	<code>_posts</code>	Περιήγηση	Διαγραφή

εικόνα 4.81

Επειδή οι φάκελοι αυτοί είναι προς διαχείριση του χρήστη και δεν χρειάζεται να αφαιρούμε δεδομένα αυτόματα είτε επειδή ο χρήστης μπορεί να ξαναεγκαταστήσει το πρόσθετο είτε επειδή θέλει να διαγράψει μόνος του τα δεδομένα που ο ίδιος έχει καταχωρήσει επιλέξαμε να μην προσθέσουμε επιπλέον κώδικα για τη διαγραφή των δεδομένων αυτών γεγονός που είναι σύμφωνο με τις αρχές του Wordpress. Οπότε αν κάποιος χρήστης θέλει να διαγράψει τόσο το πελατολόγιο του όσο και τα διαιτολόγια που δημιούργησε θα πρέπει να τα διαγράψει από το κάθε custom post type ξεχωριστά χρησιμοποιώντας την επιλογή οριστικής διαγραφής (εικόνα 4.82).



<input checked="" type="checkbox"/>	Τίτλος	Συντάκτης	Ημερομηνία
<input checked="" type="checkbox"/>	Παπαγεωργακόπουλος Δημήτρης	admin	Τελευταία Τροποποίηση 11/03/2021 στις 16:55

εικόνα 4.82

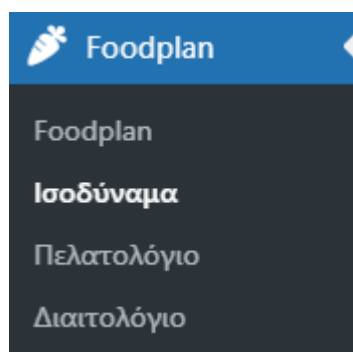
Εδώ φτάσαμε στο τέλος της περιγραφής του πρόσθετου. Παρακάτω θα αναφερθούμε στα ισοδύναμα τροφίμων, στη δημιουργία του `foodplanwp.com` ενός wordpress site σε δικό μας server, πως εγκαθιστούμε το πρόσθετό μας και πιο το οπτικό αποτέλεσμα του πρόσθετου στον ιστότοπο μαζί με κάποια παραδείγματα.

4.2.2 Ισοδύναμα τροφίμων

Το σύστημα διαβητικών ισοδυνάμων αρχικά δημιουργήθηκε για να εξυπηρετήσει τους διαβητικούς ασθενείς και στη συνέχεια για τον γενικό πληθυσμό. Πρακτικά, ο διαιτολόγος χρησιμοποιεί το σύστημα αυτό για να δημιουργήσει ένα θερμιδομετρημένο διαιτολόγιο με τον προσδιορισμό 3 μακροθρεπτικών συστατικών: **τους υδατάνθρακες, τις πρωτεΐνες και τα λίπη** (Gibney M et al., 2007). Το σύστημα των ισοδυνάμων διακρίνει τα τρόφιμα σε έξι ομάδες.

- Γάλα
- Φρούτα
- Λαχανικά
- Αμυλούχα
- Κρέας
- Λίπη

Με βάση αυτές τις 6 ομάδες τροφίμων έχουν κατηγοριοποιηθεί όλα τα τρόφιμα σε συγκεκριμένες ποσότητες σύμφωνα με τις θερμίδες που αποδίδουν κατά μέσο όρο και τα 3 μακροθρεπτικά συστατικά που προαναφέρθηκαν. Στο πρόσθετό μας περιλαμβάνουμε μία πλήρη λίστα των ισοδυνάμων για να μπορεί ο χρήστης να εξάγει σωστά τη διαιτητική εντολή του διαιτολογίου που θα δημιουργήσει. Πιο συγκεκριμένα τη λίστα αυτή μπορούμε να τη βρούμε στο υπομενού του πρόσθετου Ισοδύναμα (εικόνα 4.83).



Εικόνα 4.83

Παρακάτω δίνεται ένα μέρος της λίστας αυτής καθώς και κάποιες επεξηγήσεις για την κάθε ομάδα τροφίμων.

4.2.2.1 Γάλα

Η ομάδα αυτή έχει 3 είδη ισοδυνάμων με βάση τη περιεκτικότητα του τροφίμου σε λίπος (άπαχο, ημιάπαχο, πλήρες). (εικόνα 4.84)

Ισοδύναμο Γάλακτος 0% (90 kcal, 12 γρ. υδατάνθρακες, 8 γρ. πρωτεΐνες, 0-3 γρ. λίπη)
1 φλιτζ. γάλα 0%
1 φλιτζ. γάλα σόγιας αποβουτυρωμένο
200γρ. γιαούρτι 0%
Ισοδύναμο Γάλακτος 2% (120 kcal, 12 γρ. υδατάνθρακες, 8 γρ. πρωτεΐνες, 5 γρ. λίπη)
1 φλιτζ. γάλα 1,5%
200γρ. γιαούρτι 2%
1 φλιτζ. κεφίρ, αριάνι, ξινόγαλο 2%
Ισοδύναμο Γάλακτος 3,5% (150 kcal, 12 γρ. υδατάνθρακες, 8 γρ. πρωτεΐνες, 5 γρ. λίπη)
1 φλιτζ. γάλα 3,5%
1/2 φλιτζ. γάλα εβαπορέ πλήρες (κουτί)
1 φλιτζ. γάλα κατσίκας
200γρ. γιαούρτι από πλήρες γάλα

Εικόνα 4.84

4.2.2.2 Φρούτα

Η ομάδα αυτή προσδιορίζει την ποσότητα των φρούτων που αποδίδουν κατά μέσο όρο 60 θερμίδες (εικόνα 4.85).

Ισοδύναμο Φρούτου (60 kcal, 15 γρ. υδατάθρακες, 0 γρ. πρωτεΐνες, 0 γρ. λίπη)
1 μεγάλο ακτινίδιο
$\frac{3}{4}$ φλιτζ. ανανάς
$\frac{1}{3}$ φλιτζ. ανανάς κονσέρβα
1 μικρό (~ 110 g) αχλάδι
3 μεγάλα (~ 200 g) βερίκοκα
$\frac{1}{2}$ μεγάλο (~ 330 g) γκρέιπφρουτ
2 μέτρια (~ 140 g) δαμάσκηνα
1 και $\frac{1}{4}$ φλιτζ. (~ 380 g) καρπούζι
12 μεγάλα (~ 85 g) κεράσια
$\frac{1}{2}$ μικρό (~ 160 g) μάνγκο
2 μικρά (~ 220 g) μανταρίνι
1 μικρό (~ 120 g) μήλο

Εικόνα 4.85

4.2.2.3 Λαχανικά

Η ομάδα αυτή προσδιορίζει την ποσότητα των λαχανικών που αποδίδουν κατά μέσο όρο 25 θερμίδες. (εικόνα 4.86)

Ισοδύναμο Λαχανικών (25 kcal, 5 γρ. υδατάθρακες, 2 γρ. πρωτεΐνες, 0 γρ. λίπη)
1/2 φλιτζ. βρασμένα λαχανικά και 1 φλιτζ. ωμά λαχανικά
αγγούρι
μανιτάρια
αγκινάρα
μαρούλι
αντίδια
μελιτζάνες
βρούβες
μπρόκολο
βλίτα
μπάμιες

Εικόνα 4.86

4.2.2.4 Αμυλούχα

Η ομάδα αυτή προσδιορίζει την ποσότητα των αμυλούχων τροφίμων που αποδίδουν κατά μέσο όρο 80 θερμίδες. (εικόνα 4.87)

Ισοδύναμο Αμύλου (80 kcal, 5 γρ. υδατάθρακες, 2 γρ. πρωτεΐνες, 0 γρ. λίπη)
Ψωμί
1 φέτα (30 g) λευκό (όλων των τύπων)
1 φέτα (30 g) ολικής άλεσης
ψωμάκια ατομικά (~30 g)
½ τεμάχιο ψωμί hot dog
½ τεμάχιο σταφιδόψωμο (30 γρ)
½ τεμάχιο πίτα (π.χ. για σουβλάκι, τύπου αραβική)
2 τεμάχια φρυγανιές (μικρές)
½ τεμάχιο κουλούρι Θεσσαλονίκης (μέτριο)
2 μεγάλα (~20 g) κριτσίνια
4 τεμάχια (~20 g) κράκερ
Δημητριακά προϊόντα

Εικόνα 4.87

4.2.2.5 Κρέας

Η ομάδα αυτή έχει 4 είδη ισοδυνάμων με βάση τη περιεκτικότητα του κρέατος σε λίπος (πολύ άπαχο, άπαχο, μέτριο σε λιπαρά, υψηλό σε λιπαρά). (εικόνα 4.88)

Ισοδύναμο πολύ άπαχου κρέατος (35 kcal, 0 γρ. υδατάνθρακες, 7 γρ. πρωτεΐνες, 0-1 γρ. λίπη)
30 g λευκού κρέατος από κοτόπουλο και γαλοπούλα (χωρίς το δέρμα), μπακαλιάρου, γαλέου, γλώσσας, τόνου (φρέσκου ή κονσέρβας σε νερό), μυδιών, καβουριού, αστακού, γαρίδων, πίνων, πάπιας, φασιανού (χωρίς δέρμα), ελαφιού, στρουθοκαμήλου, τυριού cottage άπαχου, ζαμπόν γαλοπούλας άπαχου.
Ισοδύναμο άπαχου κρέατος (55 kcal, 0 γρ. υδατάνθρακες, 7 γρ. πρωτεΐνες, 3 γρ. λίπη)
30 g γαλοπούλας (σκούρο κρέας, χωρίς δέρμα), κοτόπουλου (λευκό κρέας με δέρμα), πάπιας (χωρίς δέρμα), κουνελιού, μπτον φιλέ, κόντρα, μπριζόλας χοιρινής με κόκαλο (αφού αφαιρεθεί και το ελάχιστο ορατό λίπος), ψαρανεφριού, χοιρομεριού άπαχου (πόδι, πλευρά, ώμος), ρέγκας, σαρδέλας (κονσέρβα), σολομού (φρέσκου ή κονσερβοποιημένου), τόνου (σε λάδι, αποστραγγισμένου), συκωτιού, καρδιάς 6 μέτρια στρείδια, ¼ φλιτζανιού τυρί cottage 4.5% λίπος.
Ισοδύναμο medium fat κρέατος (35 kcal, 0 γρ. υδατάνθρακες, 7 γρ. πρωτεΐνες, 0-1 γρ. λίπη)
30 g κοτόπουλου και γαλοπούλας (σκούρο κρέας με δέρμα), μοσχαρίσιου κιμά με λίπος 15%, μπριζόλας, βρασμένου χοιρομεριού, κιμά χοιρινού, παιδάκια ψητά, λιπαρών ψαριών που δεν αναφέρονται στις προηγούμενες ομάδες, ανθότυρου, φέτας, μοτσαρέλας, 1 αυγό
Ισοδύναμο high fat κρέατος (35 kcal, 0 γρ. υδατάνθρακες, 7 γρ. πρωτεΐνες, 0-1 γρ. λίπη)
30 g χοιρινά παιδάκια, λουκάνικων, σαλαμιού, τηγανιτών ψαριών, τυριών όπως γραβιέρα, κασέρι, κεφαλογραβιέρα, κεφαλοτύρι, μετσοβόνα, κοπανιστή, παρμεζάνα, cheddar, edam.

Εικόνα 4.88

4.2.2.6 Λίπη

Η ομάδα αυτή προσδιορίζει την ποσότητα των λιπαρών υλών που αποδίδουν κατά μέσο όρο 45 θερμίδες. (εικόνα 4.89)

Γάλα	Φρούτα	Λαχανικά	Αμυλούχα	Κρέας	Λίπη
Ισοδύναμο Λίπους (45 kcal, 0 γρ. υδατάνθρακες, 0 γρ. πρωτεΐνες, 5 γρ. λίπος)					
Ομάδα μονοακόρεστων λιπιδίων					
30 g αβοκάντο					
1 κ.γ. ελαιόλαδο					
10 μικρές ή 5 μεγάλες ελιές					
1 κ.σ. σησαμόσποροι					
2 κ.γ. ταχίνι					
6 κομμάτια αμύγδαλα					

Εικόνα 4.89

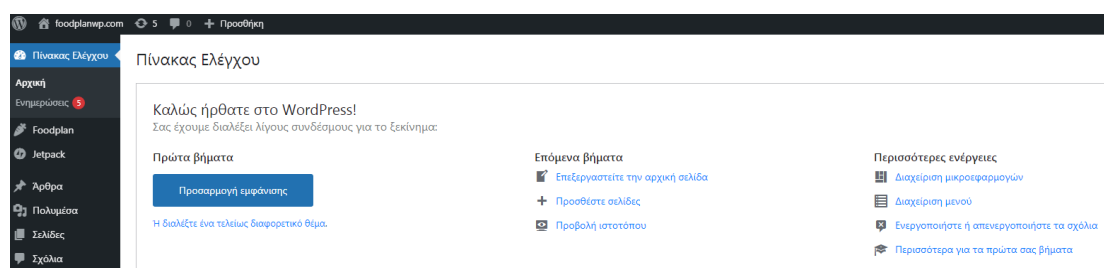
4.2.3 Ανάπτυξη web εφαρμογής

Αρχικά, κλείσαμε το **domain name : foodplanwp.com** για το πρόσθετό μας. Στη διεύθυνση αυτή του web δημιουργήσαμε μία βάση δεδομένων και εγκαταστήσαμε το Wordpress με αυτοματοποιημένα εργαλεία που μας προσφέρει η εταιρεία που μας παρέχει τον server μας (εικόνα 4.90).



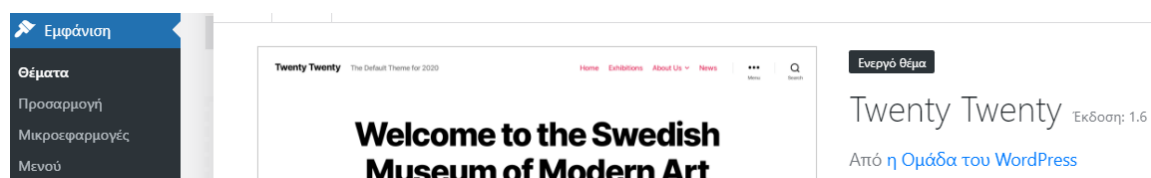
Εικόνα 4.90

Καθώς έγινε η εγκατάσταση του Wordpress στο συγκεκριμένο domain φτάσαμε στον πίνακα ελέγχου της σελίδας (Εικόνα 4.91).



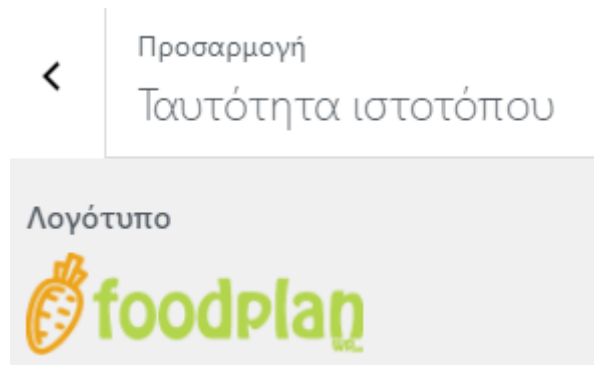
Εικόνα 4.91

Εγκαταστήσαμε 1 δωρεάν θέμα που μας προσφέρει το Wordpress με ονομασία Twenty Twenty από το μενού Εμφάνιση -> Θέματα (εικόνα 4.92) (Θέματα: Twenty Twenty, 2021).



Εικόνα 4.92

Σχεδιάσαμε το λογότυπο (logo) του πρόσθετου και το προσθεσαμε στην Ταυτότητα του Ιστοτόπου στο πεδίο Εμφάνιση -> Προσαρμογή (εικόνα 4.93).



Εικόνα 4.93

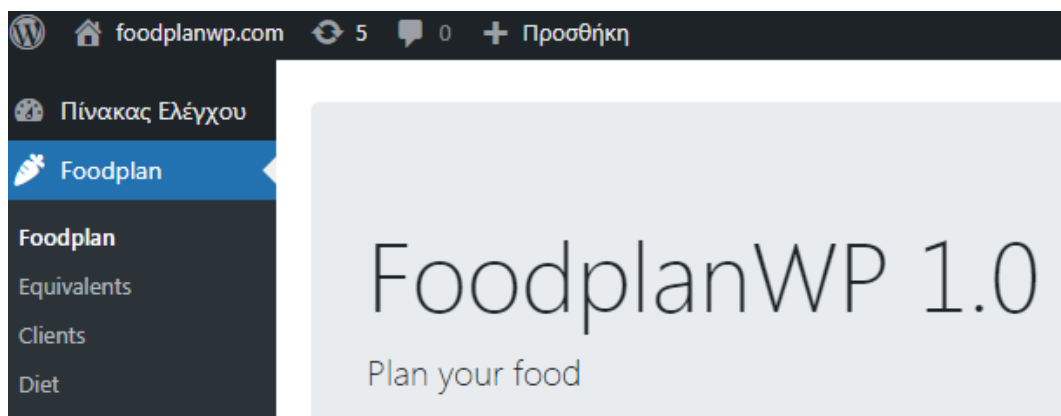
Η ιστοσελίδα αυτή δημιουργήθηκε καθαρά για λόγους παρουσίασης του πρόσθετου και δεν έχει κάποιον ιδιαίτερο σχεδιασμό (design). Περιλαμβάνει την αρχική σελίδα Foodplan (εικόνα 4.94) και μια επιπλέον σελίδα: Διαιτολόγια.



Εικόνα 4.94

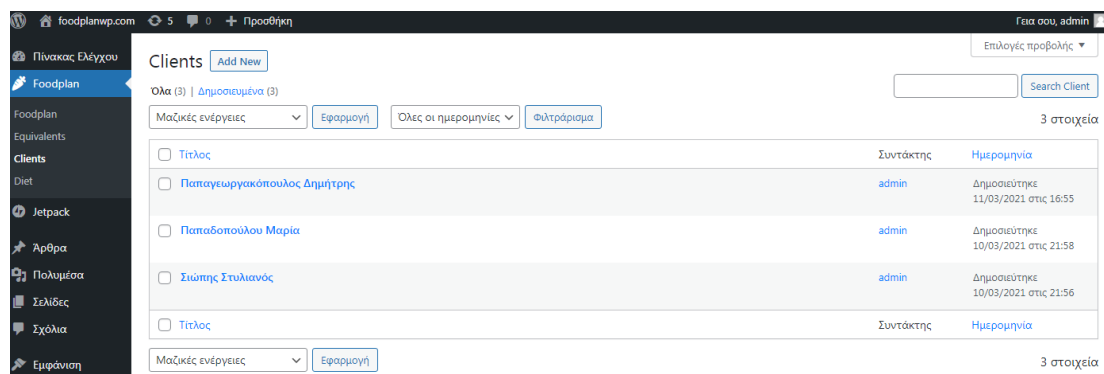
4.2.4 Εγκατάσταση πρόσθετου και παραδείγματα

Εγκαταστήσαμε το πρόσθετο Foodplan και μπορούμε να περιηγηθούμε με κάποια παραδείγματα για να δούμε πως λειτουργεί τόσο σε επίπεδο χρήστη (εικόνα 4.95) όσο και σε επίπεδο web.



Εικόνα 4.95

Στο πεδίο Clients προσθέσαμε κάποια παραδείγματα για να δείξουμε πως φαίνεται ο προσαρμοσμένος τύπος δημοσίευσης (custom post type Clients) στο admin panel. Η εφαρμογή αυτή που δημιουργήσαμε δίνει τη δυνατότητα στον χρήστη να δημιουργεί το πελατολόγιο του και να το διαχειρίζεται μέσα από τον ιστότοπό του (εικόνα 4.96). Η εφαρμογή αυτή μπορεί να χρησιμοποιηθεί και μεμονομένα από το χρήστη σαν **ξεχωριστό πρόγραμμα διαχείρισης πελατολογίου** με τη προϋπόθεση ότι πρέπει να είναι σε περιβάλλον Wordpress.



Εικόνα 4.96

Επίσης του δίνει τη δυνατότητα να κρατάει για κάθε πελάτη δεδομένα ανθρωπομετρίας καθώς και το διαιτολόγιο (εικόνα 4.97) που έχει προταθεί από τον χρήστη στο υπομενού Διαιτολόγια.

Ανθρωπομετρία

Φύλο Άνδρας Γυναίκα

Ηλικία

Βάρος kgs

Ύψος cm

Δραστηριότητα / εβδομάδα

BMI (kg/m²)

BMR (kcal)

ΣΗΕΔ (kcal)

Διαιτολόγιο

Επιλογή Διαιτολογίου

Διαιτολόγιο για Εγκυμοσύνη (2685kcal)

εικόνα 4.97

Στο πεδίο Diet προσθέσαμε τέσσερα παραδείγματα για να δείξουμε πως φαίνεται ο προσαρμοσμένος τύπος δημοσίευσης (custom post type Diet) στο admin panel (εικόνα 4.98).

Diet

Όλα (4) | Δημοσιευμένα (4)

Μαζικές ενέργειες Όλες οι ημερομηνίες

<input type="checkbox"/> Τίτλος	Συντάκτης	Ημερομηνία
<input type="checkbox"/> Διαιτολόγιο για αθλητές	admin	Δημοσιεύτηκε 10/03/2021 στις 06:24
<input type="checkbox"/> Διαιτολόγιο για απώλεια βάρους	admin	Δημοσιεύτηκε 10/03/2021 στις 17:15
<input type="checkbox"/> Διαιτολόγιο για Εγκυμοσύνη	admin	Δημοσιεύτηκε 10/03/2021 στις 17:40
<input type="checkbox"/> Διαιτολόγιο για παιδιά	admin	Δημοσιεύτηκε 10/03/2021 στις 17:42
<input type="checkbox"/> Τίτλος	Συντάκτης	Ημερομηνία

Μαζικές ενέργειες

4 στοιχεία

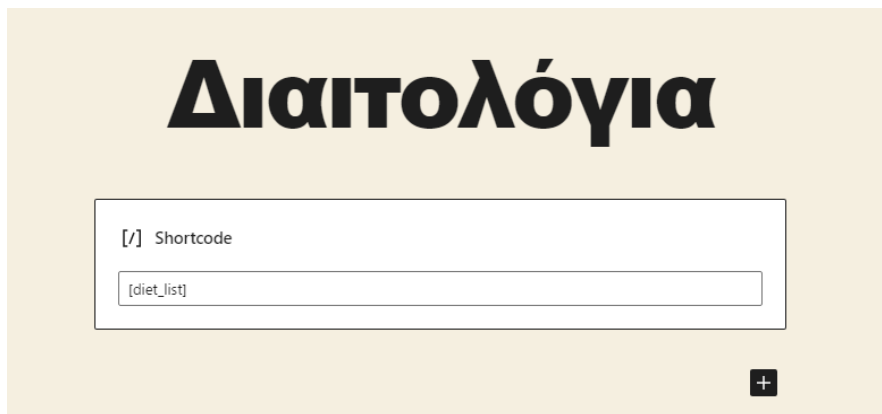
εικόνα 4.98

Η εφαρμογή αυτή που δημιουργήσαμε δίνει τη δυνατότητα στον χρήστη να δημιουργεί θερμιδομετρημένα διαιτολόγια με τη βοήθεια των ισοδυνάμων τροφίμων (εικόνα 4.99).

Ομάδες Τροφίμων	Ισοδύναμα
Γάλα άπαχο	<input type="text" value="0"/>
Γάλα ημιάπαχο	<input type="text" value="3"/>
Γάλα πλήρες	<input type="text"/>
Φρούτα	<input type="text" value="6"/>
Λαχανικά	<input type="text" value="4"/>
Αμυλούχα	<input type="text" value="9"/>
Κρέας πολύ χαμηλό σε λιπαρά	<input type="text"/>
Κρέας χαμηλό σε λιπαρά	<input type="text" value="8"/>
Κρέας μέτριο σε λιπαρά	<input type="text"/>
Κρέας (υψηλό σε λιπαρά)	<input type="text"/>
Λίπη	<input type="text" value="6"/>

εικόνα 4.99

Όταν ολοκληρώσει ο χρήστης τη δημιουργία των διαιτολογίων μπορεί να τα εξάγει στο διαδίκτυο με τη χρήση του shortcode [diet_list]. Ο τρόπος που χρησιμοποιήσαμε τον shortcode στη συγκεκριμένη ιστοσελίδα είναι με τη δημιουργία νέας σελίδας με την ονομασία Διαιτολόγια. Στη σελίδα αυτή προσθέσαμε τον shortcode (εικόνα 4.100).



εικόνα 4.100

Στη σελίδα αυτή φαίνονται τα 4 διαιτολόγια που δημιουργήσαμε μαζί με τη θρεπτική αξία που προσθέσαμε στο περιβάλλον χρήστη (εικόνα 4.101).

Διαιτολόγιο για απώλεια βάρους



●

Πρωινό: 1 ποτήρι γάλα (με 1.5% λιπαρά) και 30 γρ. δημητριακά

Δεκαπαινό: 1 φρούτο εποχής

Μεσημεριανό: 120 γρ κοτόπουλο στήθος, χωρίς πέτσα με σαλάτα

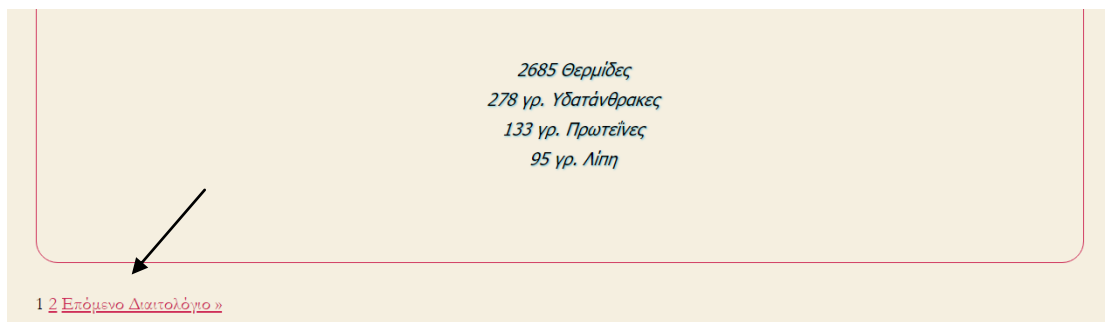
Απογευματινό: 1 φρούτο εποχής

Βραδινό: 1 μέτρια σαλάτα από λαχανικά εποχής (βραστά ή ωμά) μαζί με 60 γρ. κοτόπουλο ψητό

*1415 θερμίδες
174 gr Υδατάνθρακες
76 gr Πρωτεΐνες
42 gr Λίπη*

Εικόνα 4.101

Στη σελίδα αυτή παρατηρούμε ότι υπάρχει σελιδοποίηση ανά 2 διαιτολόγια (εικόνα 4.102) που δημιουργήσαμε μέσα από τον κώδικα του πρόσθετου. Ο επισκέπτης μπορεί να μετακινηθεί στα επόμενα διαιτολόγια όπως φαίνεται κάτω αριστερά στην εικόνα 4.102.



Εικόνα 4.102

Ο επισκέπτης μπορεί να περιηγηθεί στα διαιτολόγια που δημιουργήσαμε μέσω της αναζήτησης που είναι πάνω δεξιά στη σελίδα μας (Εικόνα 4.94). Με αναζήτηση τη λέξη «παιδιά» μας δίνει το αποτέλεσμα: «Διαιτολόγιο για παιδιά» που είναι ένα από τα παραδείγματα που δημιουργήσαμε (Εικόνα 4.103).



Εικόνα 4.103

Επίσης, ο επισκέπτης μπορεί να περιηγηθεί από το ένα διαιτολόγιο στο άλλο με τη βοήθεια βέλων μετακίνησης (εικόνα 4.104).

Διαιτολόγιο για Εγκυμοσύνη

Αντικείμενο: Διατροφή | 10 Σεπτεμβρίου 2021



🟢

Πρωινό: 1 ποτήρι γάλα με 1.5% λιπαρά και 30 γρ. δημητριακά

Δειπνή: 1 φρούτο εποχής

Μεσημεριανό: 120 γρ κοτόπουλο στήθος, μερίς τίσιμα με σάλτσα

Απογευματινό: 1 φρούτο εποχής

Οροδινό: 1 μέτριο σάλτσα από λαχανικά εποχής (φρέσκα ή καψά) με 60 γρ. κινόα/κουάφι φαγό

🍎 2685 Καρμίδες
278 γρ. υδατάνθρακες
123 γρ. πρωτεΐνες
65 γρ. λίπη

📄 Εκτύπωση...

← Διαιτολόγιο για σπώλικα βάρους Διαιτολόγιο για παιδιά →

Εικόνα 4.104

Εδώ ολοκληρώσαμε την παρουσίαση του πρόσθετου τόσο σε επίπεδο χρήστη του Πίνακα Ελέγχου του Wordpress όσο και σε επίπεδο επισκέπτη της ιστοσελίδας foodplanwr.com.

ΚΕΦΑΛΑΙΟ 5^ο

Επίλογος

5.1 Σύνοψη και συμπεράσματα

Με μεθοδικότητα φτάσαμε στο τέλος. Ήταν ένα ταξίδι γεμάτο γνώσεις και αρκετές ώρες αναζήτησης. Κάθε λεπτό που αφιερώνεις στον προγραμματισμό μετράει και σου δίνει επιπλέον εμπειρία για να επιλύσεις δυσκολότερες προκλήσεις και «σπαζοκεφαλιές». Η διπλωματική μου έδωσε ένα κίνητρο να ασχοληθούμε περαιτέρω με το αντικείμενο αυτό και μια δυναμική για το σχεδιασμό ανάπτυξης λογισμικού στο διαδίκτυο.

5.2 Μελλοντικές επεκτάσεις

Ο προγραμματισμός πάντα είναι εξελίξιμος. Η ενασχόληση με τη δημιουργία plugin για Wordpress ήταν προσωπική επιλογή και πάντα θα προσπαθώ για τη βελτίωση τόσο αυτού του project και δεσμεύομαι για τη δημιουργία νέων.

ΚΕΦΑΛΑΙΟ 6^ο

Βιβλιογραφία

1. Introduction to Plugin Development. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/intro/>
2. THE WORDPRESS PLUGIN BOILERPLATE. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <http://wppb.io/>
3. XAMPP. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://www.apachefriends.org/index.html>
4. Γνωρίστε το Wordpress. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://wordpress.org/>
5. WORDPRESS PLUGIN BOILERPLATE GENERATOR. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://wppb.me/>
6. Code editing. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://code.visualstudio.com/>
7. Best Practices. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/plugin-basics/best-practices/>
8. jQueryUI: tabs, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://jqueryui.com/tabs/>
9. Bootstrap, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://getbootstrap.com/>
10. Administration Menus. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/administration-menus/>
11. Activation / Deactivation Hooks, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/plugin-basics/activation-deactivation-hooks/>

12. How to Internationalize Your Plugin. (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/internationalization/how-to-internationalize-your-plugin/>
13. Shortcodes, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/shortcodes/>
14. Hooks, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/hooks/>
15. Registering Custom Post Types, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/registering-custom-post-types/>
16. Code Reference: post_type_supports, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα https://developer.wordpress.org/reference/functions/post_type_supports/
17. Code Reference: post_type_style, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα https://developer.wordpress.org/reference/functions/wp_enqueue_style/
18. (Code Reference: enqueue_scripts, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα https://developer.wordpress.org/reference/functions/wp_enqueue_scripts/
19. Custom Meta Boxes, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/metadata/custom-meta-boxes/>
20. Πρόσθετα: Loco translate, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://el.wordpress.org/plugins/loco-translate/>
21. Fontawesome (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://fontawesome.com/>
22. Header Requirements, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/plugin-basics/header-requirements/>
23. Plugin Security, (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/security/>
24. GNU Public License (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://wordpress.org/about/license/>
25. Uninstall Methods (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/plugin-basics/uninstall-methods/>
26. Detailed Plugin Guidelines (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://developer.wordpress.org/plugins/wordpress-org/detailed-plugin-guidelines/>
27. Developer Information (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://wordpress.org/plugins/developers/>

28. Readme Validator (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://wordpress.org/plugins/developers/readme-validator/>
29. Plugins (2021, 15 Μαρτίου) Διαθέσιμο στην ιστοσελίδα <https://wordpress.org/plugins/>
30. Gibney M., Vorster H., Kok. F (eds). (2007). *Εισαγωγή στη διατροφή του ανθρώπου*. Εκδόσεις Παρισιάνου Α.Ε., Αθήνα.
31. Παπανικολάου, Γ. (2005). *Σύγχρονη Διατροφή και Διαιτολογία* (6^η Έκδοση). Εκδόσεις Θυμάρι, Αθήνα.
32. Williams, M.H. (2003). *Διατροφή & Υγεία, Ευρωστία και Αθλητική Απόδοση* (1^η Έκδοση). Ιατρικές Εκδόσεις Π.Χ. Πασχαλίδης, Αθήνα.

