



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ  
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

«Προσαρμογή του WordPress για ανάπτυξη εφαρμογών  
λειτουργίας εκπαιδευτικών οργανισμών»

Του φοιτητή  
Χρήστου Ζυγλάκη  
Αρ. Μητρώου: 175007

Επιβλέπων  
Ονοματεπώνυμο Μιχαήλ  
Σαλαμπάσης  
Βαθμίδα Καθηγητής

Ημερομηνία 2025

Τίτλος Δ.Ε. Web εφαρμογή λειτουργίας εκπαιδευτικού οργανισμού

Κωδικός Δ.Ε. **22344**

Όνοματεπώνυμο φοιτητή/τών Χρήστος Ζυγλάκης

Όνοματεπώνυμο εισηγητή Μιχαήλ Σαλαμπάσης

Ημερομηνία ανάληψης Δ.Ε. Νοέμβριος 2022

Ημερομηνία περάτωσης Δ.Ε. Απρίλιος 2025

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

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

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

*«Αφιέρωση»*

*Χρήστος Ζυγλάκης*



# 1. Πρόλογος

Η παρούσα πτυχιακή εργασία εκπονήθηκε στο πλαίσιο της ολοκλήρωσης των σπουδών μου στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων. Στόχος της εργασίας ήταν η μελέτη, ανάπτυξη και υλοποίηση μιας διαδικτυακής πλατφόρμας διαχείρισης και διάθεσης μαθημάτων, αξιοποιώντας τεχνολογίες ανοιχτού κώδικα. Η εκπόνηση της εργασίας αποτέλεσε μια σημαντική ευκαιρία για την εμβάθυνση των γνώσεών μου στις τεχνολογίες ανάπτυξης web εφαρμογών και στη διαχείριση περιεχομένου μέσω του WordPress, καθώς και στην ενοποίηση συστημάτων όπως το WooCommerce και τα Advanced Custom Fields.

## 2. Περίληψη

### Κεφάλαιο 1ο: 1. Εισαγωγή

Στην παρούσα πτυχιακή εργασία αναπτύσσεται μία ολοκληρωμένη διαδικτυακή εκπαιδευτική πλατφόρμα με τη χρήση του συστήματος διαχείρισης περιεχομένου WordPress και του πρόσθετου WooCommerce. Στόχος της πλατφόρμας είναι η παροχή μιας ευέλικτης και λειτουργικής λύσης για τη δημιουργία, διαχείριση και πώληση εκπαιδευτικών μαθημάτων, καθώς και για την οργάνωση της σχέσης μεταξύ καθηγητών και σπουδαστών.

Οι σύγχρονες εκπαιδευτικές ανάγκες, τόσο σε ακαδημαϊκό όσο και σε επαγγελματικό επίπεδο, απαιτούν ψηφιακά συστήματα που να υποστηρίζουν την παροχή μαθημάτων εξ αποστάσεως, τη διαχείριση εκπαιδευτικού υλικού και την παρακολούθηση προόδου σπουδαστών. Η πλατφόρμα που αναπτύχθηκε στα πλαίσια της παρούσας εργασίας ενσωματώνει δυνατότητες ηλεκτρονικού εμπορίου (e-commerce), επιτρέποντας στους χρήστες να αγοράζουν μαθήματα, να παρακολουθούν τις αγορές τους και να έχουν πρόσβαση σε εκπαιδευτικό περιεχόμενο μέσω ειδικής διεπαφής.

Η εργασία εστιάζει τόσο στη θεωρητική ανάλυση των συστημάτων διαχείρισης εκπαιδευτικού περιεχομένου και των τεχνολογιών που χρησιμοποιήθηκαν, όσο και στην πρακτική υλοποίηση εξατομικευμένων λειτουργιών μέσω ανάπτυξης custom plugins και προσαρμογών. Συγκεκριμένα, αναπτύχθηκαν λειτουργίες όπως η αυτόματη δημιουργία προϊόντων WooCommerce από μαθήματα, η διασύνδεση καθηγητών και μαθημάτων, η προσαρμογή της διεπαφής χρήστη και η εμφάνιση σχετικών δεδομένων μέσω ειδικών συντομεύσεων (shortcodes).

Η επιλογή του WordPress ως βασικής πλατφόρμας έγινε με γνώμονα την ευχρηστία, την ευελιξία, τη μεγάλη κοινότητα υποστήριξης και την πληθώρα διαθέσιμων εργαλείων που παρέχει. Επιπλέον, η αξιοποίηση του WooCommerce επιτρέπει την ενσωμάτωση ισχυρών δυνατοτήτων ηλεκτρονικού εμπορίου με ελάχιστο κόστος ανάπτυξης.

Η πτυχιακή εργασία στοχεύει να προσφέρει ένα ολοκληρωμένο παράδειγμα ανάπτυξης εκπαιδευτικής πλατφόρμας, καλύπτοντας όλα τα στάδια σχεδίασης, υλοποίησης και αξιολόγησης, ενώ παράλληλα αναδεικνύει τις τεχνικές προκλήσεις και τις λύσεις που δόθηκαν κατά τη διαδικασία ενσωμάτωσης των διάφορων τεχνολογιών.

# Customizing the WordPress CMS: a case study

Christos Zyglakis

## **3. Abstract**

This thesis presents the development of a comprehensive web-based educational platform using WordPress and WooCommerce. The platform aims to provide a flexible and functional solution for the creation, management, and commercialization of online courses, as well as the organization of relationships between instructors and students.

The system integrates modern e-learning requirements, allowing users to create educational content, sell courses, and manage student progress through a user-friendly interface. The implementation involved custom plugin development, including functionalities such as automatic product creation from course metadata, the association of instructors with their courses, and the display of student and course information within the user account area.

WordPress was selected as the core platform due to its flexibility, usability, and extensive support community. WooCommerce was utilized to integrate e-commerce features, enabling seamless course purchasing and management. Throughout the project, significant challenges were addressed, particularly in the integration of course management with WooCommerce, user-role customization, and dynamic content display through shortcodes.

The result is a fully functional educational platform that demonstrates the practical application of content management systems and e-commerce tools in the field of online learning. The project highlights both the technical solutions implemented and the potential for future expansion and enhancement of the platform.

## 4. Περιεχόμενα

1. Πρόλογος.....	v
2. Περίληψη.....	vi
3. Abstract .....	vii
4. Περιεχόμενα .....	viii
5. Συνομογραφίες.....	xi
Κεφάλαιο 1ο: Θεωρητικό υπόβαθρο.....	1
1.1 Εισαγωγή στα CMS .....	1
1.2 Λίγα λόγια για το Σύστημα Διαχείρισης Περιεχομένου (Content Management System - CMS)..	1
1.3 Το WordPress ως CMS .....	1
1.4 Βασικές Τεχνολογίες του WordPress.....	2
1.4.1 Themes (Θέματα) .....	2
1.4.2 Plugins (Πρόσθετα) .....	2
1.4.3 Shortcodes (Σύντομοι Κωδικοί) .....	3
1.5 Επίλογος.....	3
Κεφάλαιο 2ο Έρευνα σχετικά με CMS και LMS.....	4
2.1 Εισαγωγή.....	4
2.2 Coursera.....	4
2.3 Udemy .....	5
2.4 Domestika .....	6
2.5 Hb Studio.....	7
2.6 Επίλογος .....	7
Κεφάλαιο 3ο : Τεχνολογίες και εργαλεία .....	8
3.1 Εισαγωγή.....	8
3.1.1 Λειτουργικές απαιτήσεις.....	8
3.1.2 Μη λειτουργικές απαιτήσεις .....	8
3.2 Αρχιτεκτονική Συστήματος .....	8
3.3 Λειτουργία του δικού μας WordPress.....	8
3.3.1 Ορισμός Χρηστών και Ρόλων.....	8
3.3.2 Τεχνολογίες που χρησιμοποιήθηκαν .....	9
3.4 Μικρή εξήγηση έργου .....	10
Επίλογος.....	11
Κεφάλαιο 4ο : Υλοποίηση Συστήματος.....	11

4.1 Εισαγωγή.....	11
4.2 Advance custom field (ACF).....	11
4.3 Lesson Layout part (Shortcode).....	12
4.3.1 (ACF - Shortcode).....	13
4.4 Custom plugin for WooCommerce .....	14
4.5 Πρόγραμμα μαθημάτων .....	16
4.6 My account .....	17
4.7 Custom plugin Μαθητές & Μαθήματα .....	18
4.8 Plugin δημιουργία custom integration.....	24
4.9 Επίλογος.....	26
Κεφάλαιο 5ο Συμπεράσματα.....	28
5.1 Περίληψη.....	28
5.2 Συμπεράσματα από τη Λειτουργία του Συστήματος .....	28
5.3 Προβλήματα και Προκλήσεις Κατά την Υλοποίηση.....	28
5.4 Μελλοντικές Επεκτάσεις.....	29



## 5. Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΠΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία



## Κεφάλαιο 1ο : Θεωρητικό υπόβαθρο

### 1.1 Εισαγωγή στα CMS

Τα Συστήματα Διαχείρισης Περιεχομένου (CMS) αποτελούν λογισμικά που επιτρέπουν τη δημιουργία, τροποποίηση και διαχείριση ψηφιακού περιεχομένου μέσω γραφικού περιβάλλοντος, χωρίς να απαιτείται γνώση προγραμματισμού. Τα CMS παρέχουν στους διαχειριστές ιστοτόπων τη δυνατότητα να οργανώνουν και να διαχειρίζονται δομημένα δεδομένα όπως άρθρα, σελίδες, πολυμέσα και χρήστες.

Η αρχιτεκτονική ενός CMS διακρίνεται σε δύο βασικά επίπεδα:

- Το **front-end**, το οποίο εμφανίζει το περιεχόμενο στον τελικό χρήστη.
- Το **back-end**, μέσω του οποίου ο διαχειριστής ελέγχει τη δομή και το περιεχόμενο του ιστότοπου.

### 1.2 Λίγα λόγια για το Σύστημα Διαχείρισης Περιεχομένου (Content Management System - CMS)

Ένα CMS είναι ένα δυναμικό site το οποίο αποτελείται από τα ακόλουθα μέρη:

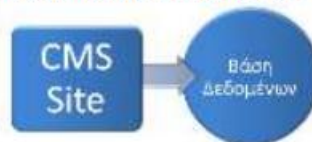
- από μια βάση δεδομένων η οποία λειτουργεί πίσω από το site,
- από το back end κομμάτι το οποίο είναι το περιβάλλον διαχείρισης του site,
- από το front end κομμάτι το οποίο είναι το site που βλέπει ο επισκέπτης.

Ένα CMS αποτελείται από μια σειρά αρχείων συνδεδεμένων μεταξύ του με τέτοιο τρόπο ώστε να αποτελούν από μόνα τους μια έτοιμη δομή. Αυτή η δομή αποθηκεύεται κατά την διάρκεια της εγκατάστασης του CMS σε μια βάση δεδομένων. Εμείς αυτό που κάνουμε είναι να συνδεόμαστε στο back end του site, να μεταβάλλουμε το περιεχόμενο και την λειτουργία του site και να βλέπουμε το τελικό αποτέλεσμα το front end. Ότι μεταβολές και προσθήκες κάνουμε στο site αποθηκεύονται αυτόματα στην βάση δεδομένων.

#### Back End – Front End



#### Βάση δεδομένων και site



### 1.3 Το WordPress ως CMS

Το WordPress αποτελεί ένα από τα δημοφιλέστερα CMS παγκοσμίως, με ποσοστό χρήσης που υπερβαίνει το 40% όλων των ιστοτόπων. Αρχικά δημιουργήθηκε ως πλατφόρμα blogging, όμως εξελίχθηκε σε πλήρες CMS μέσω της χρήσης **θεμάτων (themes)** και **πρόσθετων (plugins)**, τα οποία επιτρέπουν την επέκταση της λειτουργικότητας.

### Κύρια χαρακτηριστικά του WordPress:

- Προσφέρεται δωρεάν.
- Είναι εύκολη η εγκατάσταση του και η συντήρηση του.
- Ευχρηστία και φιλικότητα προς το χρήστη.
- Ευρεία κοινότητα υποστήριξης και πλούσια βιβλιοθήκη πρόσθετων.
- Διαθέτει εκατοντάδες έτοιμα θέματα (Themes) που προσφέρονται δωρεάν ή με κάποιο αντίτιμο.
- Προσφέρει ένα ιδανικό περιβάλλον δημιουργίας ιστολογίων και επέκτασης τους μέσω plugin.

### 1.4 Βασικές Τεχνολογίες του WordPress

Το WordPress αποτελεί ένα από τα πιο διαδεδομένα συστήματα διαχείρισης περιεχομένου (Content Management Systems – CMS), το οποίο επιτρέπει την εύκολη δημιουργία, διαχείριση και δημοσίευση ιστοσελίδων χωρίς την ανάγκη εκτενούς προγραμματιστικής γνώσης. Είναι ένα λογισμικό ανοικτού κώδικα, το οποίο υποστηρίζει εκτεταμένη παραμετροποίηση και επέκταση λειτουργιών μέσω θεμάτων (themes), πρόσθετων (plugins) και σύντομων κώδικων (shortcodes).

#### 1.4.1 Themes (Θέματα)

Τα themes στο WordPress καθορίζουν την εμφάνιση και τη διάταξη του περιεχομένου μιας ιστοσελίδας. Μέσω ενός theme, ο διαχειριστής μπορεί να προσαρμόσει τη γραφιστική ταυτότητα του ιστότοπου, να επιλέξει τοποθετήσεις μενού, υποσέλιδων, διάταξη σελίδων και πολλά άλλα.

Κατά την ανάπτυξη της πλατφόρμας μας, αξιοποιήθηκε custom theme, το οποίο προσαρμόστηκε ώστε να υποστηρίζει τη διαχείριση μαθημάτων και χρηστών με φιλικό περιβάλλον.

#### 1.4.2 Plugins (Πρόσθετα)

Τα plugins αποτελούν λογισμικά επεκτάσεις που προσθέτουν νέες λειτουργίες στο WordPress, χωρίς να απαιτείται τροποποίηση του βασικού κώδικα. Ενδεικτικά, μέσω plugins είναι δυνατή η προσθήκη ηλεκτρονικού καταστήματος (WooCommerce), διαχείρισης πεδίων (Advanced Custom Fields) ή ακόμα και η δημιουργία custom post types.

Plugin όπως :

- WooCommerce → Για την διαχείριση πωλήσεων προϊόντων φυσικών ή άυλων ηλεκτρονικά
- Advanced Custom Fields (ACF) → Για την προσθήκη προσαρμοσμένων πεδίων σε συγκεκριμένα σημεία στον κώδικα.
- Custom Plugin → Δημιουργία δικών μας plugin ώστε να μπορούμε να προσαρμόσουμε όπως εμείς θέλουμε το interface του WordPress είτε να προσθέσουμε δυνατότητες που δεν υπάρχουν σε έτοιμα plugin και άλλες προσαρμοσμένες λειτουργίες.

### **1.4.3 Shortcodes (Σύντομοι Κώδικες)**

Οι σύντομοι κώδικες (shortcodes) στο WordPress αποτελούν έναν απλό τρόπο εισαγωγής δυναμικού περιεχομένου μέσα σε σελίδες ή άρθρα, χωρίς να απαιτείται προγραμματισμός. Ένας shortcode περικλείεται σε αγκύλες (π.χ. [example\_shortcode]) και όταν εκτελείται, εμφανίζει το περιεχόμενο ή τη λειτουργία που αντιστοιχεί.

### **1.5 Επίλογος**

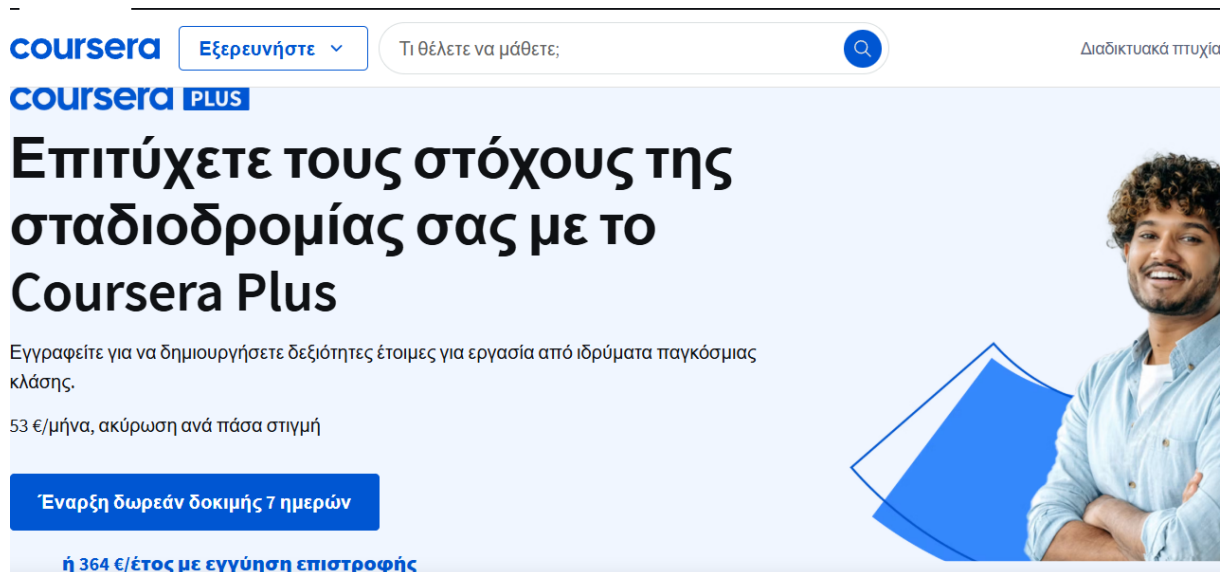
Η ευελιξία του WordPress επιτρέπει την προσαρμογή, επέκταση και δόμηση web εφαρμογών με διάφορους τρόπους, καθιστώντας το ένα ισχυρό εργαλείο για την ανάπτυξη σύγχρονων και δυναμικών ιστοσελίδων. Μέσω της χρήσης plugins, themes, shortcodes και άλλων τεχνικών, οι προγραμματιστές μπορούν να δημιουργήσουν μοναδικές και λειτουργικές λύσεις που ικανοποιούν τις συγκεκριμένες ανάγκες των χρηστών.

## Κεφάλαιο 2ο Έρευνα σχετικά με CMS και LMS

### 2.1 Εισαγωγή

Σε αυτό το κεφάλαιο αναλύουμε και άλλα CMS όπως είναι το WordPress, τα οποία συζητήθηκαν κατά την διάρκεια της πτυχιακής εργασίας ώστε να αναλύσουμε σε βάθος την λειτουργία τους αναδεικνύοντας τα χαρακτηριστικά, τα πλεονεκτήματα και τα μειονεκτήματά τους, να μπορέσουμε να βρούμε ένα πρότυπο συστήματος εκπαιδευτικού περιεχομένου (LMS) . Όπως είναι το Coursera, Udemy, hb Studio και Domestika που ακολουθούν αυτό το πρότυπο.

### 2.2 Coursera



Το Coursera είναι μια διαδικτυακή εκπαιδευτική πλατφόρμα (LMS) που συνεργάζεται με κορυφαία πανεπιστήμια και οργανισμούς παγκοσμίως, προσφέροντας μαθήματα, εξειδικεύσεις και πιστοποιημένα προγράμματα σπουδών σε πληθώρα θεματικών ενοτήτων.

Θετικά :

- ✓ Πιστοποιήσεις από πανεπιστήμια
- ✓ Πλούσια διαδραστικά εργαλεία (quizzes, assignments)
- ✓ Πολύπλοκο reporting & analytics

Αρνητικά :

- ✗ Δεν επιτρέπει εύκολη δημιουργία μαθημάτων από ανεξάρτητους εκπαιδευτές (κλειστό σύστημα)
- ✗ Πολύ υψηλό κόστος ανάπτυξης

## 2.3 Udemy

### Ready to reimagine your career?

Get the skills and real-world experience employers want with Career Accelerators.

Το Udemy είναι μια ανοιχτή εκπαιδευτική αγορά (online course marketplace), όπου εκπαιδευτές από όλο τον κόσμο μπορούν να δημιουργούν και να διαθέτουν μαθήματα για ένα ευρύ κοινό σε διάφορους τομείς, από τεχνολογία μέχρι προσωπική ανάπτυξη.

Θετικά :

- ✓ Δυνατότητα λήψης πιστοποίησης μετά το πέρας κάθε μαθήματος
- ✓ Αλληλεπίδραση μεταξύ μαθητή και καθηγητή
- ✓ Αρκετός έλεγχος στο τι αγοράζουμε. Καθώς αυτό φαίνεται και στο Dashboard

Αρνητικά :

- ✗ Ο κάθε ένας μπορεί να ανεβάσει ένα μάθημα και να "βαπτιστεί" καθηγητής
- ✗ Δεν υπάρχει πρόγραμμα μαθημάτων. Ο κάθε μαθητής μπορεί να ολοκληρώσει το μάθημα που αγόρασε όποτε εκείνος επιθυμεί
- ✗ Δεν μπορείς να παραμετροποιήσεις την εμφάνιση, τα workflows ή τη διαχείριση της πλατφόρμας (είσαι "φιλοξενούμενος" στο Udemy)

## 2.4 Domestika

The screenshot shows the Domestika website interface. At the top, there's a navigation bar with the Domestika logo, links for 'Courses', 'Community', and 'Plus', a search bar, and 'Log in' and 'Join for Free' buttons. Below the navigation, there's a breadcrumb trail: 'Courses > Popular courses'. The main content area is titled 'Popular courses' and features three course cards. Each card has a 'BEST SELLER' badge, a course title, a brief description, the instructor's name, and a 'Buy' button with a price of €0.99 (98% discount from €49.99). The courses are: 'Canva for Beginners: Create Professional Designs' by Clàudia Cánovas, 'Using ChatGPT for Work' by Víctor Mollá, and 'Digital Marketing 101 for Entrepreneurs and Freelancers' by Gabriel Perelman. On the left, there's a sidebar with 'Courses' and 'Categories' sections. The 'Categories' section lists various fields like Illustration, Craft, Marketing & Business, etc.

Το Domestika είναι μια διαδικτυακή πλατφόρμα εκμάθησης που επικεντρώνεται κυρίως σε δημιουργικές δεξιότητες, όπως design, φωτογραφία, illustration, animation και χειροτεχνία.

Ξεκίνησε στην Ισπανία και πλέον διαθέτει μαθήματα σε πολλές γλώσσες, με ιδιαίτερη έμφαση στην υψηλή αισθητική και την ποιοτική παραγωγή.

Θετικά :

- ✓ Πολύ καλαίσθητο UI/UX. Το interface είναι μοντέρνο, καθαρό και φιλικό προς τον χρήστη
- ✓ Τα μαθήματα επιλέγονται αυστηρά. Δεν μπορεί ο καθένας να ανεβάσει περιεχόμενο όπως στο Udemy.
- ✓ Πολλά μαθήματα προσφέρουν υποτίτλους σε πολλές γλώσσες

Αρνητικά :

- Περιορισμός ελέγχου από τον δημιουργό του μαθήματος. Δεν έχει το δικό του dashboard ώστε να βλέπει, να προσθέτει ή να αλλάζει περιεχόμενο ελεύθερα
- Δεν υπάρχει δυνατότητα αλληλεπίδρασης μαθητή-καθηγητή real-time
- Δεν έχει ownership των μαθητών δηλαδή άμεση πρόσβαση στα στοιχεία του μαθητή (email κ.λ.π)

## 2.5 Hb Studio



### WELCOME TO HB STUDIO

*Conceived as an artistic and working home, the Herbert Berghof Studio offers an outlet for practice and growth for the professional theatre artist and an opportunity for the young to establish roots in their intended craft. The Studio's guiding principle is creative freedom, which has as its logical consequence, a responsibility to a noble art.*

Το HB Studio είναι ένα εξειδικευμένο εκπαιδευτικό ίδρυμα με έδρα τη Νέα Υόρκη, το οποίο προσφέρει μαθήματα υποκριτικής και performing arts, τόσο δια ζώσης όσο και διαδικτυακά μέσω της πλατφόρμας HB Studio Online.

Θετικά :

- Live μαθήματα από καθηγητές (στην συγκεκριμένη περίπτωση δάσκαλους χωρού)
- Μαθήματα με διάρκεια εβδομάδων και πατάνε πάνω σε ένα calendar events, το κάθε ένα ξεχωριστό

Αρνητικά :

- Δεν μπορείς να δημιουργήσεις δικό σου courses
- Υψηλό κόστος
- Περιορισμένες δυνατότητες αυτοματοποίησης/διαχείρισης (σε σχέση με Udemy/Coursera)

## 2.6 Επίλογος

Αναλύοντας διάφορα Learning Management Systems (LMS) μπορούμε να διακρίνουμε αρκετές διαφοροποιήσεις και κοινά χαρακτηριστικά που τα καθιστούν ελκυστικά για διαφορετικούς τύπους χρηστών και οργανισμών. Καθεμία από αυτές τις πλατφόρμες προσφέρει μοναδικά χαρακτηριστικά και δυνατότητες, ενώ ταυτόχρονα έχει τα δικά της όρια και αδυναμίες. Η δική μας υλοποίηση, μέσω της πλατφόρμας WordPress και WooCommerce, προσφέρει έναν μοναδικό συνδυασμό ευελιξίας και προσαρμοστικότητας, επιτρέποντας στους δασκάλους να δημιουργούν και να διαχειρίζονται τα μαθήματα τους με πλήρη έλεγχο, ενώ οι μαθητές μπορούν να έχουν ευκολία στην παρακολούθηση και να αγοράζουν μαθήματα μέσω του WooCommerce.

## Κεφάλαιο 3ο : Τεχνολογίες και εργαλεία

### 3.1 Εισαγωγή

Η ανάπτυξη της εκπαιδευτικής πλατφόρμας βασίστηκε σε μια σειρά λειτουργικών και μη λειτουργικών απαιτήσεων, οι οποίες καθορίστηκαν ώστε να υποστηρίξουν τη δημιουργία, διαχείριση και διάθεση εκπαιδευτικών μαθημάτων μέσω διαδικτύου.

#### 3.1.1 Λειτουργικές απαιτήσεις

- Δημιουργία και διαχείριση μαθημάτων από καθηγητές
- Ανάθεση μαθημάτων σε καθηγητές και μαθητές
- Αγορά μαθημάτων από μαθητές μέσω WooCommerce
- Εμφάνιση μαθημάτων και αναθέσεων στον λογαριασμό του χρήστη (My Account)
- Αυτόματη σύνδεση μαθημάτων με WooCommerce προϊόντα

#### 3.1.2 Μη λειτουργικές απαιτήσεις

- Φιλικότητα προς το χρήστη (user-friendly περιβάλλον)
- Επεκτασιμότητα και προσαρμοστικότητα συστήματος
- Ασφάλεια δεδομένων χρηστών

### 3.2 Αρχιτεκτονική Συστήματος

Το σύστημα βασίστηκε στην πλατφόρμα WordPress και επεκτάθηκε με τη χρήση WooCommerce και προσαρμοσμένων προσθετών (custom plugins). Η συνολική αρχιτεκτονική περιλαμβάνει:

- WordPress CMS: Διαχείριση περιεχομένου και χρηστών
- WooCommerce plugin: Διαχείριση αγορών και πληρωμών
- Custom Plugin: Διαχείριση μαθημάτων και μαθητών και εμφάνιση περιεχομένου τόσο στο back end όσο και στο front end του WooCommerce
- Short Codes και Custom Post Types και Advance Custom Fields: Εμφάνιση περιεχομένου
- Themify: Το theme που χρησιμοποιήθηκε για την δημιουργία μίας εύχρηστης και κομψής εμφάνισης.

### 3.3 Λειτουργία του δικού μας WordPress

Ξεκινήσαμε βλέποντας ποια κομμάτια χρειάζονται βελτίωση ώστε να φτάσουμε να κάνουμε μια πλήρως λειτουργική και αξιόπιστη, φιλική προς το χρήστη πλατφόρμα. Το WordPress είναι ένα κορυφαίο σύστημα διαχείρισης περιεχομένου (CMS) ανοιχτού κώδικα, γνωστό για την ισχύ και την ευελιξία του.

#### 3.3.1 Ορισμός Χρηστών και Ρόλων

Το σύστημα υποστηρίζει τρεις βασικούς ρόλους χρηστών:

Ρόλος	Περιγραφή	Δικαιώματα	Ρόλος στο WordPress
-------	-----------	------------	---------------------

<b>Διαχειριστής</b>	Ολοκληρωμένος έλεγχος πλατφόρμας	Δημιουργία/επεξεργασία όλων των μαθημάτων & χρηστών	Administrator
<b>Καθηγητής</b>	Δημιουργία και διαχείριση μαθημάτων	Προσθήκη/επεξεργασία μαθημάτων, διαχείριση δικών του μαθητών	Editor
<b>Μαθητής</b>	Παρακολούθηση και αγορά μαθημάτων	Αγορά μαθημάτων, προβολή λίστας μαθημάτων του	Customer

### 3.3.2 Τεχνολογίες που χρησιμοποιήθηκαν

## Hooks

Το WordPress παρέχει ένα ιδιαίτερα ισχυρό και ευέλικτο μηχανισμό επέκτασης μέσω της χρήσης hooks. Τα hooks αποτελούν σημεία στον πυρήνα του WordPress ή στα πρόσθετα (plugins) και θέματα (themes), όπου ο προγραμματιστής μπορεί να «εισάγει» δικό του κώδικα ή να τροποποιήσει υπάρχουσες λειτουργίες χωρίς να χρειαστεί να επέμβει άμεσα στον κώδικα του πυρήνα (core code).

Υπάρχουν δύο βασικοί τύποι hooks:

- **Actions:** Επιτρέπουν την εκτέλεση custom κώδικα σε συγκεκριμένες στιγμές (π.χ. όταν δημιουργείται ένα νέο post ή όταν ολοκληρώνεται μια αγορά).
- **Filters:** Επιτρέπουν την τροποποίηση δεδομένων πριν αυτά εμφανιστούν ή αποθηκευτούν (π.χ. τροποποίηση του περιεχομένου ενός post πριν την αποθήκευση ή εμφάνιση).

### Πλεονεκτήματα Χρήσης Hooks στο Integration

Επεκτασιμότητα χωρίς επέμβαση στον πυρήνα

- Με τη χρήση hooks, μπορούμε να προσθέσουμε νέες λειτουργίες ή να τροποποιήσουμε υπάρχουσες συμπεριφορές χωρίς να αλλάξουμε τον βασικό κώδικα του WordPress ή των πρόσθετων. Αυτό διασφαλίζει ευκολότερες αναβαθμίσεις και μειωμένο κόστος συντήρησης
- Καλύτερη οργάνωση και επαναχρησιμοποίηση κώδικα
- Οι λειτουργίες που υλοποιούνται μέσω hooks είναι περιεκτικές και σαφείς, καθιστώντας τον κώδικα πιο ευανάγνωστο και πιο εύκολο στη διαχείριση. Ένα καλά δομημένο hook μπορεί να επαναχρησιμοποιηθεί και σε άλλα plugins ή θέματα
- Στο έργο μας, η χρήση hooks (όπως π.χ. το save\_post) επέτρεψε την αυτόματη καταγραφή ή ενημέρωση post/pages. Αν κάποιο page δημιουργήθηκε για πρώτη φορά ή έγινε κάποια αλλαγή σε αυτό είτε διαγράφηκε τότε η καταγραφή μέσω του hooks έχει ενημερωθεί. Χωρίς hooks, μια τέτοια λειτουργία θα απαιτούσε πιο περίπλοκες και δυσκίνητες λύσεις
- Διαλειτουργικότητα μεταξύ διαφορετικών plugins
- Χάρη στα hooks, καταφέραμε να συνδέσουμε το custom plugin μας με το WooCommerce και άλλα πρόσθετα χωρίς συγκρούσεις, εξασφαλίζοντας σταθερότητα και συμβατότητα

## Custom-plugin

Η χρήση των hooks στο integration που προσπαθήσαμε να φτιάξουμε δεν αρκούσε. Η χρήση custom κώδικα με τον οποίο κάναμε ενοποίηση των hooks και των function που φτιάξαμε εμείς, δημιούργησαν το custom plugin για το integration του WordPress. Πετύχαμε την καταγραφή οποιαδήποτε αλλαγής είτε στην βάση είτε στην σελίδα μας. Με αυτό το τρόπο ελέγχουμε αν γίνει κάποια αλλαγή σε κάποιο post, αν εισαχθεί ένα καινούργιο plugin είτε γίνει κάποια αλλαγή σε αυτό, αν εισαχθεί/διαγραφεί ή αλλάξουν οι ρόλοι ενός χρήστη, αν αλλάξουν οι κατηγορίες και όλα τα metadata από κάποιο post. Με την δημιουργία custom plugin έχουμε περισσότερη ευελιξία καθώς μπορούμε να φτιάξουμε ή να βελτιώσουμε πολλά στο WordPress τα οποία δεν μας αρέσουν όπως εμφανίζονται ή θέλουμε να παραμετροποιήσουμε ώστε να φαίνονται διαφορετικά (π.χ. η εμφάνιση χρηστών σε έναν πίνακα).

## Shortcodes

Τα shortcodes αποτελούν έναν ακόμα ισχυρό μηχανισμό επέκτασης του WordPress, επιτρέποντας την ενσωμάτωση δυναμικού περιεχομένου μέσα σε σελίδες, άρθρα ή widgets με απλό και κατανοητό τρόπο. Πρόκειται για σύντομες «εντολές» που περικλείονται σε αγκύλες (π.χ. [my\_custom\_shortcode]) και, κατά την εμφάνιση της σελίδας, αντικαθίστανται με δυναμικά παραγόμενο περιεχόμενο ή λειτουργίες.

- Για κάποιον που δεν είναι προγραμματιστής μπορεί να ενσωματώσει πολύπλοκες λειτουργίες απλά βάζοντας ένα κομμάτι απλού κειμένου κλεισμένο σε αγκύλες
- Προσφέρει ευελιξία στην χρήση καθώς με την ανάπτυξη custom shortcodes η εμφάνιση γίνεται πιο εύκολη όπως και η αναπαραγωγή αυτών των λειτουργιών (ανακατεύθυνση προϊόντος, εμφάνιση δυναμικών λιστών)
- Επαναχρησιμοποίηση ώστε το πρόγραμμα να μην επιβαρύνεται να γίνεται εύκολο στην ανάγνωση χωρίς να πρέπει να γράφουμε ξανά και ξανά το ίδιο κομμάτι κώδικα όταν πριν το έχουμε ξαναχρησιμοποιήσει
- Εκτός από τα custom Shortcodes υπάρχουν και ήδη έτοιμα όπως για χρήση κάποιον layout theme, maps, galery, διάφορα widget προσφέρουν δυνατότητα γρήγορης ενσωμάτωσης δυναμικού περιεχομένου ακόμη και σε πιο σύνθετες διατάξεις σελίδων

## 3.4 Μικρή εξήγηση έργου

Το σύστημα που αναπτύξαμε παρέχει ένα φιλικό και κατανοητό περιβάλλον στο χρήστη τόσο για μαθητές όσο και για καθηγητές. Οι δυνατότητες που έχει κάθε μια πλευρά είναι :

### Μαθητές

- Να περιηγηθούν στις διαθέσιμες σελίδες μαθημάτων
- Να δουν πληροφορίες μαθήματος (περιγραφή, τιμή, πρόγραμμα, προαπαιτούμενα μαθήματα, καθηγητής)
- Να αγοράσουν μαθήματα μέσω WooCommerce (καλάθι, checkout)
- Να δουν τα αγορασμένα μαθήματα στη σελίδα My Account → Τα Μαθήματά μου

### Καθηγητές

- Να δημιουργήσουν νέα μαθήματα μέσω του κουμπιού Δημιουργία Μαθήματος

- Να εισάγουν πληροφορίες μαθήματος (περιγραφή, πρόγραμμα, τιμή) καθώς και να μετατρέψουν το μάθημα σε online product ώστε να μπορούν να το αγοράσουν οι μαθητές (εισάγοντας και τις ημέρες διδασκαλίας)
- Να δουν τα μαθήματά τους μέσω του custom plugin

#### Διαχειριστής – Εποπτεία Συστήματος

- Ο διαχειριστής μπορεί να δει όλα τα μαθήματα, καθηγητές και αναθέσεις. Έχει απεριόριστες δυνατότητες.
- Έχει τη δυνατότητα να επεξεργαστεί ή να δημιουργήσει χειροκίνητα αναθέσεις μαθημάτων.

#### Επίλογος

Ανακεφαλαιώνοντας, χρησιμοποιήσαμε αρκετές τεχνολογίες στην ανάπτυξη της εκπαιδευτικής πλατφόρμας προσφέροντας ευελιξία, αξιοπιστία στην χρήση του, εύκολο και φιλικό προς το χρήστη και γενικά κατηγοριοποιημένα αρκετά περιεχόμενα. Η χρήση hooks αποτελεί έναν από τους βασικούς λόγους που το WordPress παραμένει ένα από τα πιο ισχυρά και ευέλικτα CMS. Τα shortcodes αποτελούν έναν πρακτικό και ισχυρό τρόπο ενσωμάτωσης δυναμικού περιεχομένου και λειτουργιών στο WordPress. Στην πτυχιακή μας, η χρήση τους βελτίωσε την επαναχρησιμοποίηση, απλοποίησε τη διαχείριση και συνέβαλε στην ανάπτυξη μιας ευέλικτης και επεκτάσιμης πλατφόρμας εκπαίδευσης.

## Κεφάλαιο 4ο : Υλοποίηση Συστήματος

### 4.1 Εισαγωγή

Στο παρόν κεφάλαιο παρουσιάζεται αναλυτικά η διαδικασία σχεδίασης και υλοποίησης της πλατφόρμας ηλεκτρονικής μάθησης που αναπτύξαμε, αξιοποιώντας το σύστημα διαχείρισης περιεχομένου WordPress. περιγράφονται τα επιμέρους υποσυστήματα και οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη των κύριων λειτουργιών, όπως η διαχείριση μαθημάτων, η διαχείριση χρηστών (καθηγητών και μαθητών) και η ενσωμάτωση με το WooCommerce για την εμπορική διάθεση των μαθημάτων. Ιδιαίτερη έμφαση δίνεται στις τεχνικές προσαρμογής και επέκτασης του WordPress μέσω της δημιουργίας custom post types, shortcodes, hooks και custom plugins. Αναλύονται επίσης τα εργαλεία και οι τεχνολογίες που χρησιμοποιήθηκαν, οι δυσκολίες που προέκυψαν κατά την υλοποίηση, καθώς και οι τρόποι με τους οποίους επιλύθηκαν.

### 4.2 Advance custom field (ACF)

Προσθέσαμε στο ACF το πεδίο Course\_price τύπου number και default τιμή το 50€. Αυτό το πεδίο εμφανίζεται στο σε ένα οποιοδήποτε μάθημα (Page) τύπου “Lesson”. Εκεί μπορεί ο editor (καθηγητής) να εισάγει την τιμή που επιθυμεί. Αυτή θα αποθηκευτεί και θα εμφανιστεί στην σελίδα του μαθήματος καθώς και όταν μετατραπεί σε WooCommerce προϊόν θα περαστούν στην σελίδα του online προϊόντος η τιμή του, ο τίτλος, η περιγραφή, τα προαπαιτούμενα μαθήματα, οι καθηγητές που το διδάσκουν.

## Κεφάλαιο 3

7 Course\_price \* course\_price # Number

General Validation Presentation Conditional Logic

Field Type  
# Number Browse Fields

Field Label  
Course\_price  
This is the name which will appear on the EDIT page

Field Name  
course\_price  
Single word, no spaces. Underscores and dashes allowed

Default Value  
50  
Appears when creating a new post

Fields

#	Label	Name	Type
1	Title *	title	Text
2	Desc *	desc	Text Area
3	prerequisites	prerequisites	Relationship
4	Teaching	teaching	Relationship
5	GC_ID	gc_id	Email
6	products	products	Post Object
7	Course_price *	course_price	Number

Course\_price \*  
50 €

Themify Custom Panel

Page Options Page Appearance Query Posts Query Portfolios

Page Sub Heading

Sidebar Option  
Default sidebar1 sidebar1 sidebar-left sidebar-none

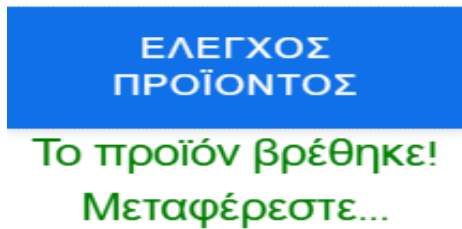
Εικόνα από το page με κατηγορία lessons

### 4.3 Lesson Layout part (Shortcode)

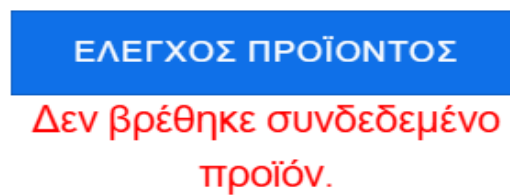
Σε αυτό το σημείο κάτω δεξιά (βέλος 1) έχω εισάγει ένα custom HTML/Text/Shortcode το [custom\_product] το οποίο εμφανίζει ένα Text “Αυτό είναι το συνδεδεμένο προϊόν” και ένα button “Έλεγχος Προϊόντος” παντώντας το κουμπί αυτό παίρνει το id του μαθήματος ψάχνει στην βάση

δεδομένων αν υπάρχει Woocommerce product συνδεδεμένο με αυτό το id και αν ναι εμφανίζει το αντίστοιχο μήνυμα. “ Το προϊόν βρέθηκε! Μεταφέρεστε...” με πράσινα γράμματα και “ Δεν βρέθηκε συνδεδεμένο προϊόν.” με κόκκινα γράμματα. Αφού βρεθεί το προϊόν μας ανακατευθύνει στην αντίστοιχη σελίδα του.

Αυτό είναι το  
συνδεδεμένο προϊόν

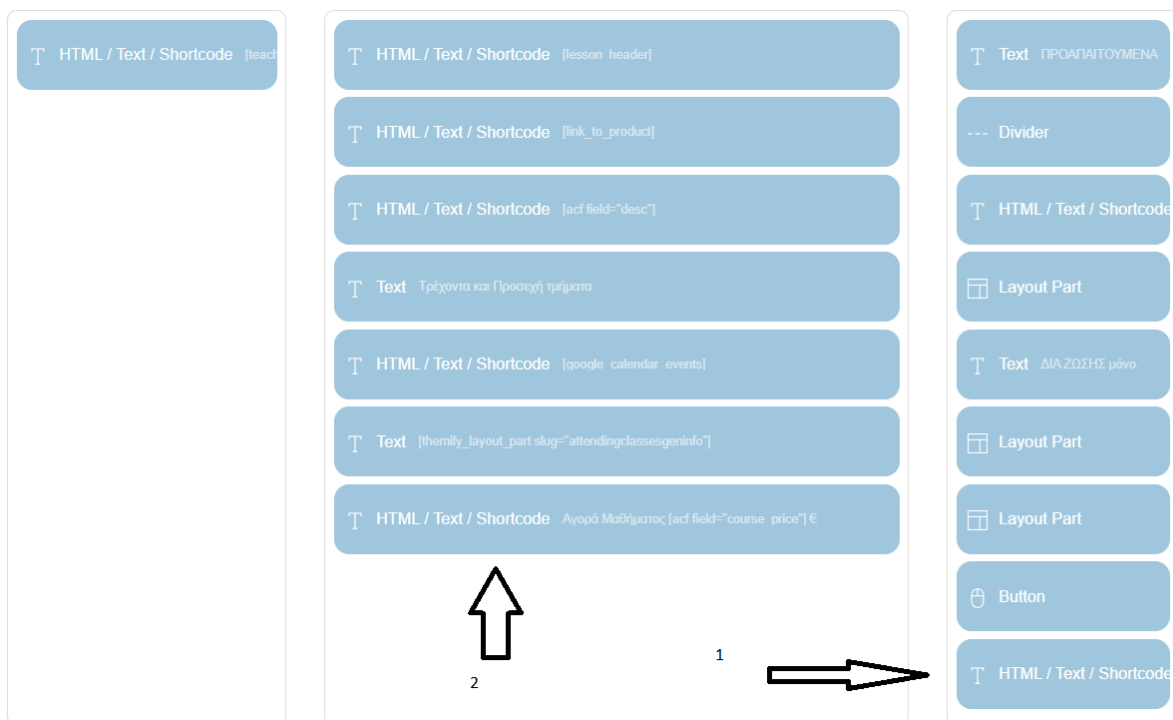


Αυτό είναι το συνδεδεμένο  
προϊόν



#### 4.3.1 (ACF - Shortcode)

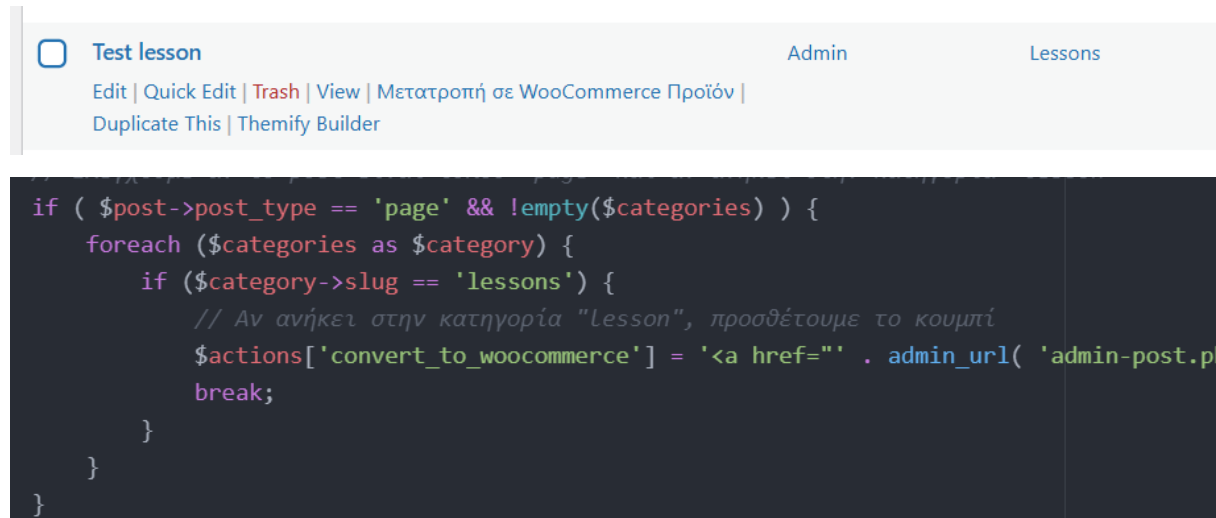
Στο δεύτερο βελάκι έχω εισάγει ένα shortcode [acf field="course\_price"] το οποίο παίρνει την τιμή από το advance custom field lesson και πεδίο course\_price. Αυτό περιέχει την τιμή που έχουμε ορίσει για το κάθε μάθημα και θα εμφανίσει ένα παράδειγμα “Αγορά Μαθήματος 50€” όπου αυτή την τιμή θα καλεστούμε να πληρώσουμε όταν θα θελήσουμε να αγοράσουμε το μάθημα από τη σελίδα του στο WooCommerce.



Εικόνα με βελάκια 1 και 2 για τα custom shortcode

## 4.4 Custom plugin for WooCommerce

Προσθέτει κουμπί για μετατροπή μαθήματος σε προϊόν WooCommerce. Το κουμπί αυτό προστίθεται μόνο σε pages τύπου lessons ώστε να μην μπορεί να μετατρέψει λάθος σελίδα σε προϊόν όπως μια σελίδα καθηγητή. Το εισάγουμε ανάμεσα από το Edit | Quick Edit | Trash | View και προσθέτουμε μια function ώστε όταν πατηθεί το κουμπί να ενεργοποιηθεί. Αν πατήσουμε το “Μετατροπή σε WooCommerce Προϊόν” τότε στον κώδικα αυτό θα ψάξει με βάση το post id αν υπάρχει το αντίστοιχο προϊόν στο WooCommerce, αν υπάρχει θα μας ανακατευθύνει κατευθείαν στην σελίδα με τα products έτσι αποφεύγουμε να ξαναμετατρέψουμε σελίδα σε προϊόν, αλλιώς αν δεν υπάρχει τότε παίρνουμε όλα τα στοιχεία από το page-μάθημα όπως είναι ο τίτλος, η περιγραφή, η τιμή του προϊόντος τα προαπαιτούμενα μαθήματα και τον/τους καθηγητή/τες που το διδάσκουν.



Αφού πάρουμε τα στοιχεία τα αποθηκεύουμε σε meta data ώστε να δημιουργήσουμε(εικόνα 1) το αντίστοιχο WooCommerce Product θα χρειαστεί να κανουμε και την σύνδεση του page με το WooCommerce product (εικόνα 2). Αυτό το επιτυγχάνουμε με metadata όπου χρησιμοποιώ το post id και το “\_related\_woocommerce\_product” για να μπουν στην βάση δεδομένων στο αντιστοιχο product id. Αν διαγραφεί κάποιο product τότε αυτόματα θα διαγραφούν όλα τα metadata καθώς και η συσχέτιση για να μπορούμε να το ξαναδημιουργήσουμε. Όπως αν αλλάξουν πεδία από τίτλο, περιγραφή, προαπαιτούμενα μαθήματα, καθηγητές που το διδάσκουν κ.λ.π στην σελίδα του μαθήματος τότε αυτόματα με ένα Action hook (εικόνα 3) που έχω φτιάξει θα ενεργοποιήσει την συνάρτηση “delete\_product\_meta\_everywhere ” και θα αλλάξουν και για το WooCommerce product ώστε να υπάρχει μια συμφωνία και στα 2.

```
// Δημιουργία νέου προϊόντος WooCommerce
$new_product = array(
    'post_title'    => $course_title,
    'post_name'    => $course_title,
    'post_content' => $course_content,
    'post_status'  => 'publish',
    'post_type'    => 'product',
);

$product_id = wp_insert_post( $new_product );
```

Εικόνα 1 δημιουργία Product

```
// Σύνδεση του μαθήματος με το προϊόν WooCommerce
update_post_meta( $post_id, '_related_woocommerce_product', $product_id );
```

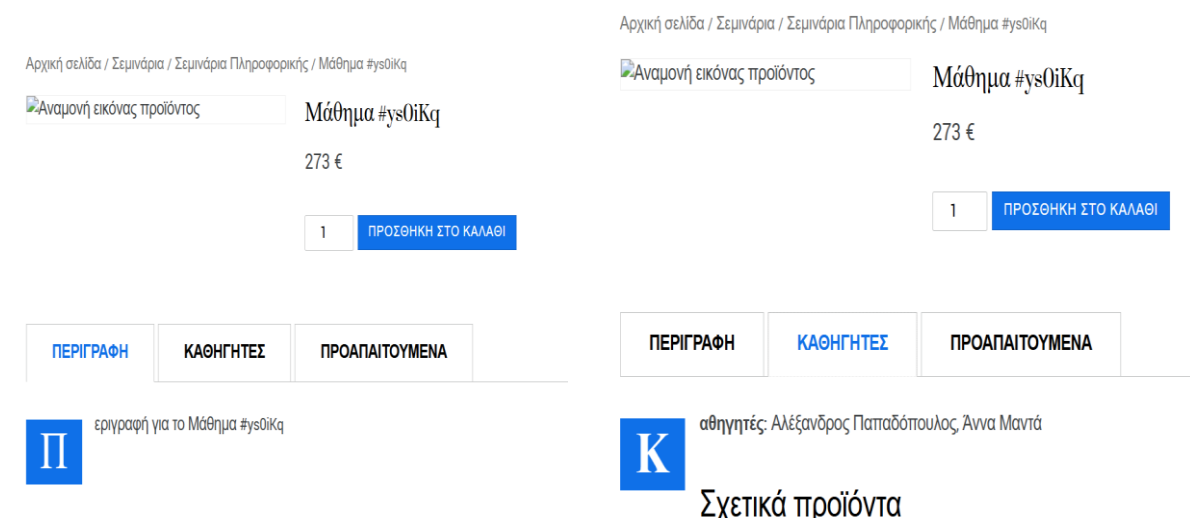
Εικόνα 2 σύνδεση page με WooCommerce product

```
add_action('before_delete_post', 'delete_product_meta_everywhere');
```

Εικόνα 3 Action hook

## WooCommerce Tab

Το WordPress μας αφήνει σχετικά μεγάλη ευελιξία στην παραμετροποίηση σελίδων, plugin κ.λπ. Ένα παράδειγμα είναι το WooCommerce στο οποίο χρησιμοποίησα hook filter για να μπορέσω να τροποποιήσω ήδη υπάρχον post στην συγκεκριμένη περίπτωση WooCommerce product.



Τα στοιχεία που χρειάστηκα να τροποποιήσω σε σχέση με αυτά που υπήρχαν ήδη από την δημιουργία του προϊόντος ήταν :

Αρχικά :

- title
- desc
- course\_price
- \_regular\_price
- \_virtual (εικονικό προϊόν)

Τελική τροποποίηση με εισαγωγή:

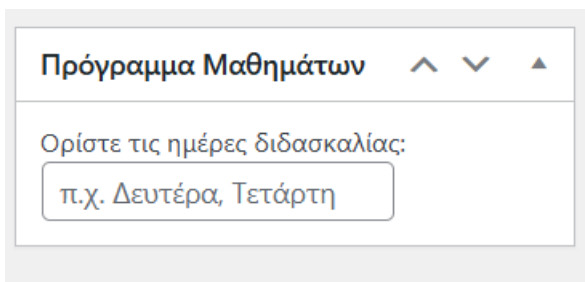
- Prerequisites (προαπαιτούμενα)
- Teaching (καθηγητές)

Με την function `add_custom_course_tabs` εισάγουμε 2 νέα tabs ώστε να φαίνονται οι καθηγητές και τα προαπαιτούμενα μαθήματα.

```
// Δημιουργία του tab "Καθηγητές"
if ($course_teaching) {
    $tabs['course_teaching_tab'] = array(
        'title' => __('Καθηγητές', 'woocommerce'),
        'priority' => 20,
        'callback' => 'course_teaching_tab_content',
    );
}
```

#### 4.5 Πρόγραμμα μαθημάτων

Μετά την δημιουργία του WooCommerce product έχω εισάγει ένα box στο οποίο μπορείς να εισάγεις ημέρες διδασκαλίας των μαθημάτων ώστε να ξέρουν οι μαθητές πότε μπορούν να το παρακολουθήσουν.



```
function save_course_schedule( $post_id ) {
    if ( isset( $_POST['course_schedule'] ) ) {
        update_post_meta( $post_id, '_course_schedule', sanitize_text_field( $_POST['course_schedule'] ) );
    }
}
add_action( 'save_post', 'save_course_schedule' );
```

- Χρησιμοποιώ hook ώστε να αποθηκεύσω τις ημέρες και να αποθηκευτεί σε custom metadata. Δημιουργούμε μια συνάρτηση με όνομα `save_course_schedule`
- Ως παράμετρο παίρνει το `$post_id`, δηλαδή το ID της ανάρτησης (post, course, κ.λπ.) που αποθηκεύεται
- Ελέγχουμε αν υπάρχει στο `$_POST` (δηλαδή στη φόρμα που υποβάλλεται) το πεδίο `course_schedule`
- Το `course_schedule` είναι το custom πεδίο που δημιουργήσαμε για να αποθηκεύονται οι ημέρες, αν υπάρχει τότε κάνουμε αποθήκευση δεδομένων στη βάση.
- Η τιμή του πεδίου είναι το περιεχόμενο που υποβλήθηκε στη φόρμα, αφού πρώτα το έχουμε καθαρίσει (sanitize) με την `sanitize_text_field()` — αυτό αφαιρεί επικίνδυνα ή αχρείαστα δεδομένα και κάνει το input ασφαλέστερο.

#### 4.6 My account

Ο κάθε μαθητής που θα συνδεθεί στο account του και εφόσον έχει αγοράσει κάποιο μάθημα θα εμφανίζονται σε μορφή πίνακα το ένα κάτω από το άλλο τα μαθήματα, η ημερομηνία αγοράς και οι ημέρες διδασκαλίας του μαθήματος. Διαφορετικά θα εμφανίζεται μήνυμα “Δεν έχετε αγοράσει ακόμα μαθήματα.” ή σε περίπτωση που έχει αγοραστεί το μάθημα αλλά δεν υπάρχουν ακόμα οι ημέρες διδασκαλίας τότε θα εμφανίζεται το μήνυμα «Δεν έχουν οριστεί.» . `esc_html($course['schedule'])` .» όπου `course['schedule']`) είναι το custom field που έχουμε φτιάξει για τις ημέρες διδασκαλίας, όπου αν ενημερωθεί αυτό θα ενημερωθεί και το πεδίο.

## Τα Μαθήματά Μου

Μάθημα	Ημερομηνία Αγοράς	Ημέρες Διδασκαλίας
Γλώσσες και Τεχνολογίες Ιστού	14/04/2025	Δευτέρα, Τρίτη

Σε περίπτωση που κάποιος μαθητής (Customer) αγοράσει ένα product τότε δεν θα μπορεί να το ξανααγοράσει. Αυτό επιτυγχάνεται με “add filter woocommerce\_is\_purchasable” ; όπου κρύβουμε το κουμπί της αγοράς προϊόντος για να μην μπορεί να το αγοράσει ξανά εφόσον το έχει ήδη στην κατοχή του. Έτσι με αυτό τον τρόπο επιτυγχάνουμε να μην υπάρχουν διπλές εγγραφές σε μαθήματα.

```

// Παίρνουμε όλες τις ολοκληρωμένες παραγγελίες του χρήστη
$order = wc_get_orders(array(
    'customer_id' => $user_id,
    'status'      => array('completed', 'processing'),
    'limit'      => -1,
));

// Ελέγχουμε αν έχει ήδη αγοράσει το μάθημα
foreach ($order as $order) {
    foreach ($order->get_items() as $item) {
        if ($item->get_product_id() == $product->get_id()) {
            return false; // Δεν μπορεί να το ξαναγοράσει
        }
    }
}

return $purchasable;

```

Ο κώδικας χρησιμοποιεί τη συνάρτηση `wc_get_orders()` για να ανακτήσει όλες τις παραγγελίες που σχετίζονται με τον τρέχοντα χρήστη (μέσω του `$user_id`), τις βάζουμε σε ένα array ώστε να το διατρέξουμε με μια `foreach`, για να ελέγξουμε αν μέσα στο array βρίσκεται το μάθημα που πρέπει να απενεργοποιήσουμε. Παίρνει υπόψη μόνο παραγγελίες με κατάσταση:

- `completed` (ολοκληρωμένες)
- `processing` (υπό επεξεργασία)

Η παράμετρος `'limit' => -1` διασφαλίζει ότι επιστρέφονται όλες οι σχετικές παραγγελίες, χωρίς όριο.

Ακολουθούν δύο επικαλυπτόμενα `foreach`:

- Το πρώτο (`foreach ($order as $order)`) διατρέχει όλες τις παραγγελίες του χρήστη.
- Το δεύτερο (`foreach ($order->get_items() as $item)`) διατρέχει όλα τα προϊόντα (`items`) της κάθε παραγγελίας.

Για κάθε προϊόν της παραγγελίας γίνεται έλεγχος:

```
if ($item->get_product_id() == $product->get_id())
```

Αν το ID του προϊόντος είναι ίδιο με το ID του μαθήματος που εξετάζεται, επιστρέφεται `false`. Αυτό σημαίνει ότι ο χρήστης δεν μπορεί να το ξαναγοράσει. Αν δεν βρεθεί τότε ο χρήστης μπορεί να αγοράσει το μάθημα κανονικά αφού δεν έχει κάποιο με ίδιο ID.

#### 4.7 Custom plugin Μαθητές & Μαθήματα

Σε αυτό το plugin αναπτύχθηκαν:

- Πίνακας εμφάνισης μαθημάτων, μαθητών και email αυτών
- Φιλτράρισμα πίνακα ανάλογα με το ποιο μάθημα ψάχνουμε

- Προσθήκη μαθητή σε μάθημα. Αν κάποιος μαθητής έχει αγοράσει offline το μάθημα περνάμε με αυτόν τον τρόπο την αγορά του και online ώστε να μπορεί να το παρακολουθήσει
- Εισαγωγή μαθητών – μαθημάτων και ανάθεση μαθημάτων στους χρήστες-μαθητές. Αυτό έγινε στα πλαίσια της δοκιμής της βάσης ώστε να υπάρχουν αρκετά δεδομένα ώστε να μπορούμε να τα επεξεργαστούμε και να κάνουμε αρκετές δοκιμές

```
// Προσθήκη νέου admin menu για τη διαχείριση μαθητών & μαθημάτων
function add_students_courses_menu() {
    add_menu_page(
        'Μαθητές & Μαθήματα', // Τίτλος σελίδας
        'Μαθητές & Μαθήματα', // Όνομα στο μενού
        'manage_options', // Δικαιώματα πρόσβασης (μόνο admins)
        'students_courses', // Slug της σελίδας
        'display_students_courses_page', // Callback function για το περιεχόμενο
        'dashicons-welcome-learn-more', // Εικονίδιο στο admin menu
        20 // Θέση στο admin menu
    );
}
add_action('admin_menu', 'add_students_courses_menu');
```

Θα δημιουργήσουμε μια φόρμα στο WordPress Admin Panel, όπου ο διαχειριστής μπορεί να επιλέξει έναν χρήστη και ένα μάθημα και να τον προσθέσει χειροκίνητα στη λίστα των εγγεγραμμένων μαθητών.

```
// Επεξεργασία φόρμας προσθήκης μαθητή
if (isset($_POST['add_student'])) {
    $student_id = intval($_POST['student_id']);
    $course_id = intval($_POST['course_id']);

    if ($student_id && $course_id) {
        $order = wc_create_order(array('customer_id' => $student_id));
        $order->add_product(wc_get_product($course_id), 1); // Προσθήκη του μαθήματος στην παραγγελία
        $order->calculate_totals();
        $order->update_status('completed'); // Ολοκλήρωση παραγγελίας

        echo '<div class="updated"><p><input checked="" type="checkbox"> Ο μαθητής προστέθηκε με επιτυχία στο μάθημα!</p></div>';
    }
}
```

Αυτό για να το πετύχουμε θα χρειαστεί στις 2 φόρμες μαθητή και μαθήματα να εισάγουμε βιβλιοθήκες ώστε να μπορούμε να τις προσαρμόσουμε για να μας εμφανίζει χρήστες με βάση το id τους και το ρόλο τους και μαθήματα με βάση το id τους. Όταν ο διαχειριστής προσθέσει έναν μαθητή, εμφανίζεται επιβεβαίωση. Αν ο μαθητής είναι ήδη εγγεγραμμένος, εμφανίζεται μήνυμα σφάλματος.

```
// Φόρμα προσθήκης μαθητή
?>
<!-- Φόρτωση της βιβλιοθήκης Select2 -->
<link href="https://cdn.jsdelivr.net/npm/select2@4.0.13/dist/css/select2.min.css" rel="stylesheet" />
<script src="https://cdn.jsdelivr.net/npm/jquery@3.6.0/dist/jquery.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/select2@4.0.13/dist/js/select2.min.js"></script>
```

### Διαδικασία εισαγωγής μαθήματος σε μαθητή (Offline πληρωμή)

- Ο διαχειριστής επιλέγει μαθητή και μάθημα και πατάει "Προσθήκη Μαθητή"
- Δημιουργείται μια νέα WooCommerce παραγγελία με το μάθημα ως προϊόν
- Η παραγγελία ολοκληρώνεται αυτόματα (completed) ώστε ο μαθητής να έχει άμεση πρόσβαση
- Ο μαθητής βλέπει το μάθημα στο "Οι Παραγγελίες μου" (My Account > Orders).

### Εμφάνιση μαθητών με φίλτρο

Στην εμφάνιση μαθημάτων, μαθητών και email έχω προσθέσει ένα φίλτρο ώστε αν κάποιος επιθυμεί να δει μόνο ένα συγκεκριμένο μάθημα πόσους μαθητές έχει καθώς και τα email ώστε να επικοινωνήσει μαζί τους. Αυτό γίνεται για να μην χαθούν σε τόσες εγγραφές. Από κάτω φαίνεται η μπάρα επιλογής μαθήματος ως φίλτρο. Σε περίπτωση που δεν επιλεγεί κάποιο μάθημα σαν φίλτρο τότε κανονικά εμφανίζονται όλα τα δεδομένα του πίνακα.

```
// Αν υπάρχει επιλεγμένο μάθημα από το φίλτρο
$selected_course = isset($_GET['course_filter']) ? intval($_GET['course_filter']) : 0;

// Φόρμα φίλτρου μαθημάτων
echo '<form method="GET">';
echo '<input type="hidden" name="page" value="students_courses">';
echo '<label for="course_filter"><strong>Φιλτράρισμα ανά Μάθημα:</strong></label> ';
echo '<select name="course_filter" id="course_filter" onchange="this.form.submit();">';
echo '<option value="0">-- Όλα τα Μαθήματα --</option>';
foreach ($courses as $course) {
    $selected = ($selected_course == $course->ID) ? 'selected' : '';
    echo '<option value="' . esc_attr($course->ID) . '" ' . $selected . '>' . esc_html($course->post_title) . '</option>';
}
echo '</select>';
echo '</form>';

echo '<hr>';
```

Παράλληλα με αυτό το κομμάτι έχω φτιάξει μια υποκατηγορία στο “μαθήματα και μαθητές” ένα υπομενού “οι μαθητές μου”, το οποίο έχει και αυτό ίδια δικαιώματα με το κύριο μενού. Ο πίνακας είναι ίδιος με το προηγούμενο μενού, εμφανίζει τα ίδια δεδομένα η διαφορά είναι ότι υπάρχει ο περιορισμός τα μαθήματα που εμφανίζονται να είναι για το συγκεκριμένο καθηγητή που είναι συνδεδεμένος.

Καθηγητής: Admin		
Μάθημα	Μαθητής	Email

Στο παρακάτω κομμάτι κώδικα αναλύουμε ακριβώς τον τρόπο με τον οποίο ελέγχει ποιος user είναι συνδεδεμένος. Παίρνουμε το id του user που είναι συνδεδεμένος στη σελίδα μας, μετά παίρνουμε από την βάση τις πληροφορίες για τον συγκεκριμένο user έπειτα βρίσκουμε το όνομα του και το εμφανίζουμε Καθηγητής: {όνομα καθηγητή}. Ψάχνουμε έπειτα ποια μαθήματα διδάσκει ο συγκεκριμένος καθηγητής τα αποθηκεύουμε σε ένα array και ύστερα θα τα εμφανίσουμε στον πίνακα. Οι καθηγητές θα βλέπουν μόνο τους μαθητές που έχουν αγοράσει τα δικά τους μαθήματα. Παίρνει τα μαθήματα που διδάσκει ο καθηγητής χρησιμοποιώντας το ACF teaching. Σε περίπτωση που δεν υπάρχουν εγγεγραμμένοι μαθητές στο μάθημα του τότε εμφανίζει μήνυμα "Δεν υπάρχουν εγγεγραμμένοι μαθητές".

```
// Εμφάνιση μαθητών μόνο για τα μαθήματα του editor
function display_editor_students_page() {
    $editor_id = get_current_user_id(); // Παίρνουμε το ID του editor

    echo '<div class="wrap"><h1>Οι Μαθητές μου</h1>';

    $user_id = get_current_user_id();
    $user_info = get_userdata($user_id);
    $user_name = $user_info->display_name;
    echo '<p><strong>Καθηγητής:</strong> ' . esc_html($user_name) . '</p>';

    // Παίρνουμε τα μαθήματα που διδάσκει ο editor
    $courses = get_posts(array(
        'post_type' => 'product',
        'numberposts' => -1,
        'meta_query' => array(
            array(
                'key' => 'teaching', // ACF Relationship Field "teaching"
                'value' => "'" . "$user_name" . "'",
                'compare' => 'LIKE'
            ),
        ),
    ));
}
```

### Εισαγωγή μαθημάτων και μαθητών σαν Testing (γέμισμα της βάσης δεδομένων)

Ο παρακάτω κώδικας προσθέτει:

- Ένα κουμπί με το κείμενο "Εισαγωγή Random Μαθητών"
- Ένα textbox όπου καθορίζεις πόσους τυχαίους μαθητές θέλεις να δημιουργήσεις
- Χρήστες με τυχαία στοιχεία (username, name, email) και ρόλο customer. Το WordPress μας επιτρέπει να χρησιμοποιήσουμε το "wp\_generate\_password" που δημιουργεί έναν τυχαίο κωδικό πρόσβασης από ένα καθορισμένο σύνολο χαρακτήρων.

**Εισαγωγή Random Μαθητών**

Αριθμός μαθητών προς εισαγωγή:

Εισαγωγή Random Μαθημάτων

👤 Ανάθεση Τυχαίων Μαθημάτων σε Ανενεργούς Χρήστες

**Λίστα Μαθητών & Μαθημάτων**

👤 Συνολικοί χρήστες: 1000

🟢 Μαθητές με παραγγελίες: 1000

🔴 Μαθητές χωρίς παραγγελίες: 0

📖 Συνολικά Μαθήματα ("Σεμινάρια Πληροφορικής"): 840

```
// Διαχείριση της προσθήκης τυχαίων μαθητών
if (isset($_POST['generate_random_students'])) {
    $count = intval($_POST['random_student_count']);
    if ($count > 0) {
        for ($i = 0; $i < $count; $i++) {
            $random_username = 'user_' . wp_generate_password(6, false);
            $random_name = 'Student_' . wp_generate_password(3, false);
            $random_email = $random_username . '@example.com';

            // Δημιουργία χρήστη
            $user_id = wp_insert_user(array(
                'user_login' => $random_username,
                'user_pass' => wp_generate_password(),
                'user_email' => $random_email,
                'first_name' => $random_name,
                'role' => 'customer'
            ));
        }
    }
}
```

Με τον ίδιο τρόπο δημιουργούμε random μαθήματα. Δημιουργούμε random μαθήματα με τυχαίο τίτλο τυχαία περιγραφή και τιμή και επειδή τα στοιχεία των προαπαιτούμενων μαθημάτων και οι οι καθηγητές είναι Advance custom field τύπου relationship. Αυτό σημαίνει ότι εμφανίζει ένα πεδίο με όλα τα μαθήματα και ένα πεδίο με όλους τους καθηγητές όλα με βάση το id τους. Ο κώδικας έχει ενημερωθεί ώστε να επιλέγει τυχαία 1 ή 2 μαθήματα ως προαπαιτούμενα και 1 ή 2 καθηγητές από τις αντίστοιχες κατηγορίες "lessons" και "Teacher".

```
<h2>Εισαγωγή Random Μαθημάτων</h2>
<form method="post">
<input type="number" name="random_courses_count" min="1" value="1" required>
<input type="submit" name="generate_random_courses" class="button-primary" value="Δημιουργία Μαθημάτων">
</form>
<?php
if (isset($_POST['generate_random_courses']) && intval($_POST['random_courses_count']) > 0) {
    $count = intval($_POST['random_courses_count']);
    for ($i = 0; $i < $count; $i++) {
        $random_title = 'Μάθημα #' . wp_generate_password(6, false);
        $random_description = 'Περιγραφή για το ' . $random_title;
        $random_price = rand(10, 500);

        $new_course = array(
            'post_title' => $random_title,
            'post_content' => $random_description,
            'post_status' => 'publish',
            'post_type' => 'page',
        );
    }
}
```

Χρησιμοποιούμε την array\_rand ώστε να επιλέξουμε τυχαία μαθήματα.

Αναλυτικά η ροή :

- count(\$lesson\_ids) μετράει πόσα IDs υπάρχουν στον πίνακα
- min(3, count(\$lesson\_ids)) εξασφαλίζει ότι:

Αν υπάρχουν τουλάχιστον 3 μαθήματα, θα επιλεγούν 3

Αν υπάρχουν λιγότερα από 3 μαθήματα, θα επιλεγούν όλα τα διαθέσιμα.

- `array_rand()` επιλέγει τυχαία 1,2 ή 3 κλειδιά από τον πίνακα `$lesson_ids`.
- `update_field()` → Καταχωρεί τα τυχαία IDs ως array στο ACF field.

```
if (!empty($lesson_ids)) {
    $random_prerequisites = array_map(
        fn($key) => $lesson_ids[$key],
        (array) array_rand($lesson_ids, min(3, count($lesson_ids))));
    //var_dump($random_prerequisites);
    update_field('prerequisites', $random_prerequisites, $course_id);
}
```

Τέλος πριν φτιάξουμε και το αντίστοιχο WooCommerce με τον ίδιο τρόπο που το φτιάξαμε στην αρχή θα χρειαστεί να ακολουθήσουμε το ίδιο layout lesson ώστε να υπάρχει μια συμφωνία στα μαθήματα που δημιουργούνται καθώς και να πάρουν αυτόματα την κατηγορία “Lessons” ώστε να εμφανίζονται σωστά. Το συγκεκριμένο layout\_part που χρησιμοποιούμε εμείς είναι με id = 3712 σαφώς σε ένα άλλο project το κάθε layout part έχει το δικό του id οπότε χρησιμοποιούμε το αντίστοιχο.

```
$layout_part_shortcode = '[themify_layout_part id="3712"]';
```

Για να γίνει η ανάθεση των μαθημάτων σε μαθητές οι οποίοι δεν έχουν αγοράσει ούτε ένα μάθημα χρειάζεται να πατήσεις το κουμπί “Ανάθεση Μαθημάτων”. Τα βήματα που θα ακολουθήσουμε είναι :

#### **Εύρεση χρηστών που δεν έχουν παραγγελίες:**

- Θα ψάξουμε στις παραγγελίες του WooCommerce.
- Θα βρούμε χρήστες που δεν έχουν αγοράσει κανένα προϊόν.

#### **Επιλογή τυχαίων μαθημάτων:**

- Θα βρούμε προϊόντα της κατηγορίας "Σεμινάρια Πληροφορικής".
- Θα επιλέξουμε 2 ή 3 τυχαία μαθήματα.

#### **Ανάθεση μαθημάτων στους χρήστες:**

- Θα τα προσθέσουμε ως ολοκληρωμένες παραγγελίες στο όνομά τους με βάση το Id.

Παρακάτω είναι και η εμφάνιση μηνυμάτων σε περίπτωση λάθους ή σωστής ανάθεσης.

```
if (empty($users_without_orders)) {
    echo '<div class="updated"><p>✔ Δεν υπάρχουν χρήστες χωρίς αγορές.</p></div>';
} else {
```

```
if (empty($seminar_products)) {
    echo '<div class="error"><p>⚠ Δεν βρέθηκαν προϊόντα στην κατηγορία "Σεμινάρια Πληροφορικής".</p></div>';
}
```

```
echo '<div class="updated"><p>✔ Τα μαθήματα ανατέθηκαν επιτυχώς στους ανενεργούς χρήστες!</p></div>';
```

#### 4.8 Plugin δημιουργία custom integration

Σκοπός δημιουργίας αυτού του plugin είναι η μαζική συλλογή δεδομένων από το project και τις αλλαγές σε αυτό. Όλα αυτά που δημιουργήσαμε μέσω του plugin έγιναν για να υπάρχει αυτόματη καταγραφή δεδομένων και όχι χειροκίνητη και εστίασαμε στην αποφυγή λαθών.

Αν κάποιος για παράδειγμα δημιουργήσει 10 σελίδες με τα αντίστοιχα product τους τότε θα γίνει και αυτόματη καταγραφή 10 μαθημάτων με τύπο create. Όλες οι αλλαγές καταγράφονται σε έναν πίνακα στην βάση δεδομένων wp\_migration\_logs όπου εκεί μπαίνουν οι καταγραφές σε SQL commands.

Στην καταγραφή μας για να αποφύγουμε συνεχόμενες εγγραφές του τύπου heart beat που έχει το WordPress οι οποίες εκτελούνται κάθε λεπτό έχουμε εξαιρέσει από την καταγραφή με το παρακάτω κώδικα.

```
// Εξαιρέσεις για συγκεκριμένα queries
$excluded_patterns = [
    "transient_doing_cron", // Αποφυγή καταγραφής cron-related entries
    "autosave",           // Αποφυγή autosave
    "heartbeat",           // Αποφυγή WordPress heartbeat
];
```

Όλα τα κομμάτια κώδικα έχουν υλοποιηθεί με action hooks ώστε να εκτελούνται κατά την αλλαγή λεπτομεριών και έτσι να πετύχουμε το σκοπό που έχουμε δημιουργήσει το plugin.

Όλα όσα αποθηκεύονται στον πίνακα wp\_migration\_logs μπορούν να εξαχθούν σαν αρχείο με κατάληξη .sql . Αυτό θα χρησιμοποιηθεί όταν τρέχουν το project παραπάνω από 1 άτομο ώστε να υπάρχει σωστή καταγραφή των δεδομένων και να δουλεύουν πάνω στο ίδιο project.

### **Καταγραφές :**

- Καταγραφή δημιουργίας ή ενημέρωσης posts/pages
- Καταγραφή σχέσης Page -> WooCommerce Product
- Καταγραφή των κατηγοριών του post
- Καταγραφή postmeta (custom fields)
- Καταγραφή διαγραφής posts/pages και postmeta
- Καταγραφή ενεργοποίησης/απενεργοποίησης plugin
- Καταγραφή συσχετισμού προϊόντος με κατηγορίες WooCommerce
- Καταγραφή αλλαγών σε κατηγορίες και tags
- Καταγραφή αλλαγών στον Customizer
- Καταγραφή δημιουργίας χρήστη μαζί με τα usermeta
- Καταγραφή ενημέρωσης χρήστη και μεταδεδομένων

### **Extra PHP αρχείο για την εκτέλεση των Queries**

Εδώ σε αυτό το PHP αρχείο κάνουμε σύνδεση στην βάση αφού έχουμε πάρει το wp\_migration\_logs.sql με τα στοιχεία που έχουμε ορίσει για όνομα βάσης, κωδικούς κ.λπ. Αν δεν είναι σωστά τα στοιχεία θα μας εμφανίσει μήνυμα λάθους αλλιώς θα ξεκινήσει να καταχωρεί στον πίνακα wp\_migration\_logs όλα τα Queries που καταγράψαμε. Αφού τελειώσει θα αρχίσει να εκτελεί ένα ένα τα Queries , αν δεν υπάρχουν Queries στον πίνακα τότε πάλι εμφανίζεται μήνυμα λάθους.

```

// Σύνδεση με τη βάση δεδομένων
$mysqli = new mysqli('localhost', 'root', '', 's2cnew2'); // 's2cnew2'

// Έλεγχος σύνδεσης
if ($mysqli->connect_error) {
    die("Connection failed: " . $mysqli->connect_error);
}

// Ανάγνωση των queries από τον πίνακα wp_migration_logs
$result = $mysqli->query("SELECT query FROM wp_migration_logs");

if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        $query = $row['query'];

        // Εκτέλεση του query
        if ($mysqli->query($query) === TRUE) {
            echo "Query executed successfully: " . $query . "\n";
        } else {
            echo "Error executing query: " . $mysqli->error . "\n";
        }
    }
} else {
    echo "No queries found in wp_migration_logs.\n";
}

// Κλείσιμο σύνδεσης
$mysqli->close();
?>

```

#### 4.9 Επίλογος

Στο παρόν κεφάλαιο παρουσιάστηκε αναλυτικά η διαδικασία υλοποίησης της προτεινόμενης διαδικτυακής εκπαιδευτικής πλατφόρμας, αξιοποιώντας το WordPress ως βασικό σύστημα διαχείρισης περιεχομένου (CMS) σε συνδυασμό με το WooCommerce για τη διαχείριση των συναλλαγών. Εφαρμόστηκαν τεχνικές επεκτασιμότητας και παραμετροποίησης μέσω custom plugin ανάπτυξης, hooks, shortcodes και ειδικών λειτουργιών που διασφαλίζουν τη λειτουργικότητα της πλατφόρμας σύμφωνα με τις απαιτήσεις που τέθηκαν.

Αναπτύξαμε εργαλεία που διευκολύνουν τη διαχείριση μαθημάτων και χρηστών, αυτοματοποιήσαμε τη συσχέτιση καθηγητών με μαθήματα και ενσωματώσαμε μηχανισμούς ελέγχου αγορών, ώστε να αποφεύγονται λάθη όπως η επαναγορά μαθημάτων.

Επιπλέον, σχεδιάστηκαν custom κουμπιά για την ευκολία των διαχειριστών στην εισαγωγή νέων μαθημάτων και στην προβολή σχετικών δεδομένων, ενισχύοντας τη χρηστικότητα και την εμπειρία χρήστη. Η υλοποίηση αυτή αποδεικνύει τη δυνατότητα αξιοποίησης ανοιχτών τεχνολογιών και εργαλείων για τη δημιουργία εξατομικευμένων και λειτουργικών εκπαιδευτικών πλατφορμών

Η ολοκλήρωση της υλοποίησης αποτελεί θεμέλιο για την αξιολόγηση και την περαιτέρω επέκταση της πλατφόρμας στο μέλλον.

## Κεφάλαιο 5ο Συμπεράσματα

### 5.1 Περίληψη

Σκοπός της παρούσας πτυχιακής εργασίας ήταν η ανάπτυξη μιας διαδικτυακής πλατφόρμας διαχείρισης και διάθεσης μαθημάτων, βασισμένης στο σύστημα διαχείρισης περιεχομένου WordPress. Το σύστημα που υλοποιήθηκε επιτρέπει την εύκολη δημιουργία, διαχείριση και εμπορική διάθεση μαθημάτων από καθηγητές, ενώ παράλληλα προσφέρει ένα φιλικό περιβάλλον περιήγησης και αγοράς για τους μαθητές.

Τα κυριότερα χαρακτηριστικά που επιτεύχθηκαν είναι:

- Δημιουργία Advance custom field για την συσχέτιση τιμής με μάθημα.
- Αυτοματοποιημένη μετατροπή μαθήματος σε WooCommerce προϊόν με ένα κουμπί βασισμένο σε Shortcode.
- Εμφάνιση των μαθημάτων που έχει αγοράσει κάθε μαθητής στη σελίδα My Account με χρήση custom plugin.
- Διαχωρισμός ρόλων και λειτουργιών (μαθητής – καθηγητής).
- Εύκολη διαχείριση περιεχομένου μέσω ACF πεδίων και προσαρμοσμένων plugins.

### 5.2 Συμπεράσματα από τη Λειτουργία του Συστήματος

Το σύστημα ανταποκρίνεται πλήρως στις απαιτήσεις που τέθηκαν στην αρχή της ανάπτυξης:

- Ευκολία Χρήσης → Οι χρήστες (μαθητές, καθηγητές) μπορούν εύκολα να εκτελέσουν τις βασικές λειτουργίες τους.
- Ολοκληρωμένη Διαχείριση → Καθηγητές δημιουργούν μαθήματα, μαθητές τα αγοράζουν και τα παρακολουθούν.
- Αξιοπιστία → Οι συνδέσεις μεταξύ μαθημάτων, χρηστών και παραγγελιών λειτουργούν χωρίς σφάλματα.
- Επεκτασιμότητα → Το σύστημα μπορεί να επεκταθεί με νέα πεδία ή λειτουργίες στο μέλλον.

### 5.3 Προβλήματα και Προκλήσεις Κατά την Υλοποίηση

Κατά τη διαδικασία υλοποίησης της εκπαιδευτικής πλατφόρμας, προέκυψαν αρκετές τεχνικές και σχεδιαστικές προκλήσεις που έπρεπε να αντιμετωπιστούν ώστε να διασφαλιστεί η σωστή λειτουργία του συστήματος και η βέλτιστη εμπειρία χρήστη.

- Η ανάγκη για έλεγχο της επαναγοράς μαθημάτων από χρήστες απαιτούσε τη χρήση της `wc_get_orders` και τη διαχείριση παραγγελιών με custom λογική. Η μεγαλύτερη πρόκληση

ήταν να αναγνωρίζεται σωστά ο χρήστης και το προϊόν σε περιπτώσεις πολλαπλών παραγγελιών και να διασφαλίζεται πως το ίδιο μάθημα δεν μπορεί να ξαναγοραστεί.

- Παρότι τα WordPress hooks είναι ιδιαίτερα ισχυρά, η χρήση τους απαιτεί ακρίβεια, καθώς μπορούν να ενεργοποιηθούν σε λάθος στιγμές ή να προκαλέσουν side effects. Ιδιαίτερα με τις ενέργειες όπως save\_post, χρειαζόταν να διασφαλιστεί πως τα δεδομένα αποθηκεύονται μόνο όταν είναι έγκυρα, χωρίς να επηρεάζονται άλλες λειτουργίες του συστήματος. Ειδικά στην δημιουργία custom plugin με χρήση hooks για το migration της σελίδας, οι εντολές έπρεπε να είναι αρκετά ξεκάθαρες για να αποφευχθεί οποιοδήποτε λάθος στην καταγραφή των ενεργειών της σελίδας.
- Η χρήση custom Themify Layout Parts και η ενσωμάτωσή τους αυτόματα σε νέα μαθήματα μέσω custom κώδικα παρουσίασε δυσκολίες σε σχέση με τον συγχρονισμό της εμφάνισης. Επιπλέον, η συμβατότητα με τα WooCommerce templates χρειάστηκε προσαρμογές ώστε να φαίνονται τα μαθήματα στον λογαριασμό του χρήστη (My Account) με τρόπο λειτουργικό και ξεκάθαρο.

## 5.4 Μελλοντικές Επεκτάσεις

Μετά την υλοποίηση της διπλωματικής και αντιμετωπίζοντας κάποια προβλήματα ορίμασαν οι σκέψεις μου σχετικά με την επέκταση της εφαρμογής. Υπάρχουν περιθώρια βελτίωσης και εξέλιξης και οι κατευθύνσεις που προτείνω είναι :

1. Υποστήριξη Live Μαθημάτων → Ενσωμάτωση συστήματος live streaming (π.χ. μέσω Zoom) ώστε να πραγματοποιούνται διαδραστικά μαθήματα
2. Σύστημα Αξιολόγησης και Πιστοποιήσεων → Προσθήκη δυνατότητας αξιολόγησης μαθημάτων (reviews), καθώς και αυτόματη έκδοση πιστοποιητικών παρακολούθησης
3. Προηγμένα Στατιστικά και Αναφορές → Δημιουργία dashboards με αναφορές προόδου μαθητών ("Σε εξέλιξη" / "Ολοκληρώθηκε") και στατιστικά από πωλήσεις μαθημάτων
4. Ειδοποίηση email από τους καθηγητές απευθείας σε μαθητές ή σε ομάδα μαθητών που παρακολουθούν το μάθημα
5. Upload εργασιών → Δυνατότητα δημιουργίας κουμπιού upload ούτως ώστε να μπορούν οι μαθητές να ανεβάζουν εργασίες, ώστε να τις αξιολογούν οι καθηγητές.
6. Προτάσεις μαθημάτων → Μηχανισμός εμφάνισης προτεινόμενων μαθημάτων. Δημιουργία μιας μεθόδου όπου θα κοιτάει τι έχουν αγοράσει οι χρήστες και αν κάποιοι έχουν παρόμοια μαθήματα θα τους προτείνει μαθήματα που πιστεύει ότι θα τον ενδιαφέρουν.

Η υλοποίηση των παραπάνω λειτουργιών θα ενισχύσει τη λειτουργικότητα και τη χρηστικότητα του συστήματος, καθιστώντας το ανταγωνιστικό σε σχέση με παρόμοιες εμπορικές πλατφόρμες (όπως Udemy ή Coursera). Η παρούσα διπλωματική εργασία υλοποίησε ένα ολοκληρωμένο σύστημα διαχείρισης μαθημάτων και εκπαιδευτικών προϊόντων σε περιβάλλον WordPress και WooCommerce. Κατά τη διάρκεια αυτής της πτυχιακής εργασίας, είχα την ευκαιρία να εφαρμόσω θεωρητικές γνώσεις στην πράξη, να έρθω αντιμέτωπος με πραγματικά τεχνικά προβλήματα και να αναπτύξω δεξιότητες στην αρχιτεκτονική του WordPress. Οι γνώσεις που απέκτησα θα με βοηθήσουν μελλοντικά στην εξέλιξη μιας διαδικτυακής εκπαιδευτικής τεχνολογίας.