



INTERNATIONAL
HELLENIC
UNIVERSITY

DEPARTMENT OF INDUSTRIAL ENGINEERING AND MANAGEMENT

Αυτοματοποιημένη Διαχείριση
Ερευνητικού Έργου στο Εργαστήριο
Μικροσυστημάτων

Automated Research Project Management at
the Microsystems Laboratory

Καφαντάρης Αστέριος

Επιβλέπων καθηγητής: Μιχάλης Κιζήρογλου

Θεσσαλονίκη 2025

Abstract

The aim of the thesis is to develop a streamlined process of updating and displaying research data on a website by dynamically pulling the information from data sheets. In Chapter 2 the epistemic background needed as well as the technologies used in relation to the project are presented. In Chapter 3 the overall architecture of the system and the decision making behind it will be analyzed. A component breakdown is performed in order to further facilitate discussion behind the decisions taken. Chapter 4 describes the development of the system with the help of code examples and some practical explanations. Chapter 5 focuses on how the site was tested in order to meet expectations. Test cases, validation tools and their results in comparison to the projections are presented. In chapter 6, the challenges met throughout the method development are discussed, along with identified solutions. Finally in Chapter 7 the conclusions drawn, along with suggestions for further improvements are presented.

Περίληψη

Ο στόχος της διπλωματικής είναι η ανάπτυξη μιας βελτιστοποιημένης διαδικασίας ενημέρωσης και προβολής ερευνητικών δεδομένων σε μια ιστοσελίδα, με δυναμική ανάκτηση των πληροφοριών από φύλλα δεδομένων. Στο Κεφάλαιο 2, παρουσιάζεται το απαραίτητο επιστημονικό υπόβαθρο, καθώς και οι τεχνολογίες που χρησιμοποιήθηκαν στην εργασία. Στο Κεφάλαιο 3, αναλύεται η συνολική αρχιτεκτονική του συστήματος και η λογική πίσω από τις σχεδιαστικές επιλογές. Επιπρόσθετα πραγματοποιείται ανάλυση των δομικών στοιχείων του συστήματος για να διευκολυνθεί η συζήτηση σχετικά με τις αποφάσεις που λήφθηκαν κατά την δημιουργία του. Το Κεφάλαιο 4 περιγράφει την ανάπτυξη του συστήματος με τη βοήθεια παραδειγμάτων κώδικα και μερικών πρακτικών εξηγήσεων. Το Κεφάλαιο 5 αναλύει τις δοκιμές του ιστότοπου, οι οποίες πραγματοποιήθηκαν για να εξασφαλιστεί ότι πληροί όλες τις τεχνικές προδιαγραφές και τις λειτουργικές ανάγκες του χρήστη . Παρουσιάζονται τα εργαλεία με τα οποία έγιναν οι επαληθεύσεις, περιπτώσεις δοκιμών και τα αποτελέσματά τους σε σύγκριση με τις προβλέψεις. Το Κεφάλαιο 6 αφορά τις προκλήσεις που αντιμετώπισα κατά την ανάπτυξη του προγράμματος και τις λύσεις που βρήκα για αυτά τα προβλήματα. Τέλος, στο Κεφάλαιο 7, παρουσιάζονται τα συμπεράσματα που προέκυψαν κατά τη δημιουργία του συστήματος, μαζί με προτάσεις για περαιτέρω βελτιώσεις και μελλοντικές επεκτάσεις του.

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές ευχαριστίες μου στον επιβλέποντα καθηγητή κ.Κιζήρογλου για την βοήθεια του καθ' όλη την διάρκεια της εκπόνησης της διπλωματικής εργασίας.

Πίνακας περιεχομένων

Abstract	2
Περίληψη	3
Ευχαριστίες	4
Πίνακας περιεχομένων	5
Επεξήγηση ακρωνύμων	7
1 Εισαγωγή	8
1.1 Επιστημονική περιοχή της διπλωματικής	8
1.2 Σημασία της Εφαρμογής	8
1.3 Συνήθεις Μεθοδολογίες Υλοποίησης	9
2 Ανασκόπηση των Τεχνολογιών που χρησιμοποιήθηκαν και Θεωρητικό υπόβαθρο	10
2.1 Επισκόπηση των εργαλείων και τεχνολογιών που χρησιμοποιήθηκαν	10
2.1.1 Hypertext Preprocessor (PHP)	10
2.1.2 HTML	11
2.1.3 Bootstrap	11
2.1.4 JQuery και JavaScript	11
2.1.5 Dropbox	12
2.1.6 WinSCP	12
2.1.7 XAMPP	12
2.1.8 Visual Studio Code	12
2.2 Ενσωμάτωση δεδομένων από αρχεία CSV σε συστήματα ιστού	13
2.3 Στατικές και Δυναμικές Αρχιτεκτονικές Ιστοσελίδων	13
3 Αρχιτεκτονική και Σχεδίαση Συστήματος	16
3.1 Επισκόπηση του Συστήματος	16
3.2 Φιλοσοφία Σχεδίασης	16
3.3 Περιγραφή των Βασικών Μερών του Συστήματος	17
3.3.1 Αρχείο Συντονισμού	17
3.3.2 Αρχείο Παραγωγής Δυναμικού Περιεχομένου	18

3.3.3 Αρχείο Προκαθορισμένου Προτύπου Διάταξης	18
3.4 Σύνοψη Λειτουργίας Συστήματος και Κρίσιμα Σημεία Διασύνδεσης.....	18
4 Ανάλυση του Κώδικα	21
4.1 Αρχείο Συντονισμού.....	21
4.2 Αρχείο Προκαθορισμένου Προτύπου Διάταξης	24
4.3 Αρχεία Παραγωγής Περιεχομένου	29
4.3.1 Ανάκτηση δεδομένων (Data Fetching)	29
4.3.2 Επικύρωση και Αποθήκευση	30
4.3.3 Ανάλυση και Δόμηση Δεδομένων(Data Parsing and Structuring)	31
4.3.4 Απόδοση(Rendering).....	32
4.3.5 Εξαιρέσεις	47
5 Δοκιμές και Αποτελέσματα, Προκλήσεις και Λύσεις	90
5.1 Αποτελέσματα Λειτουργικών Δοκιμών	90
5.1.1 Δοκιμές ανάκτησης και επεξεργασίας δεδομένων	90
5.1.2 Δοκιμές δημιουργίας στατικών σελίδων	91
5.1.3 Δοκιμές Ασφαλείας	92
5.2 Αποτελέσματα Μη Λειτουργικών Δοκιμών	92
5.2.1 Δοκιμές Απόδοσης.....	92
5.2.2 Δοκιμές Ανθεκτικότητας.....	94
5.2.3 Δοκιμές Χρηστικότητας και Αποκριτικότητας	96
5.3 Προκλήσεις και Λύσεις.....	97
5.3.1 Ενσωμάτωση Dropbox και πρόσβαση σε αρχεία	97
5.3.2 Δημιουργία Προτύπου χωρίς πλαίσιο ανάπτυξης	97
5.3.3 Απόδοση HTML μέσα από PHP Scripts	97
6 Συμπεράσματα και Προτάσεις Αναβάθμισης	102
6.1 Συνολική Σύνοψη	102
6.2 Η Υβριδική Αρχιτεκτονική του Συστήματος	102
6.3 Προτάσεις Βελτίωσης	103
7 Βιβλιογραφία	105

Επεξήγηση ακρωνύμων

Την πρώτη φορά που τα χρησιμοποιείτε στο κείμενο πρέπει να τα αναπτύσσετε πλήρως, μετά τα χρησιμοποιείτε ως ακρόνυμα και οφείλει ο αναγνώστης να έρχεται σε αυτό το σημείο για υπενθύμιση.

Π.χ.

CSV	Comma-Separated Values
CMS	Content Management Systems, Συστήματα Διαχείρισης Περιεχομένου
HTML	HyperText Markup Language
PHP	Hypertext Preprocessor
CSS	Cascading Style Sheets
FTP	File Transfer Protocol
XSS	Cross-Site Scripting
UTF	UCS Transformation Format
URL	Uniform Resource Locator
HTTP	Hypertext Transfer Protocol
XAMPP	Cross-Platform Apache MySQL PHP Perl
RSS	Really Simple Syndication

1 Εισαγωγή

1.1 Επιστημονική περιοχή της διπλωματικής

Στη σύγχρονη κοινωνία οι ανάγκες και οι απαιτήσεις του ανθρώπου όσων αφορά τους τομείς της τεχνολογίας και της πληροφορίας ολοένα και αυξάνονται πόσο μάλλον σε ακαδημαϊκά περιβάλλοντα. Αυτή η διπλωματική εργασία αντιμετωπίζει την αυξανόμενη ζήτηση για δυναμικές και παράλληλα ευκόλως συντηρήσιμες και φιλικές προς τον χρήστη και ιδιοκτήτη πλατφόρμες σε ακαδημαϊκά περιβάλλοντα, όπου η συχνή και αποτελεσματική κοινοποίηση των διαφόρων εργασιών και ανακοινώσεων είναι κρίσιμη για την ομαλή λειτουργία τους. Τα επιστημονικά πεδία που εμπλέκονται στην περάτωση αυτής της εργασίας, είναι των πληροφοριακών συστημάτων, της τεχνολογίας λογισμικού (software engineering) και της οπτικοποίησης δεδομένων (data visualization).

1.2 Σημασία της Εφαρμογής

Τα ακαδημαϊκά και ερευνητικά εργαστήρια έρχονται συχνά αντιμέτωπα με προκλήσεις όσων αφορά την έγκυρη και διαρκή ενημέρωση των πληροφοριακών τους συστημάτων. Γεγονός που αποτελεί τροχοπέδη στην επαγγελματική και ακαδημαϊκή ζωή ερευνητών, φοιτητών και καθηγητών. Η παρούσα εφαρμογή αυτοματοποιεί αυτή τη διαδικασία ενημέρωσης με την δημιουργία στατικών ιστοσελίδων οι οποίες είναι συνδεδεμένες με αρχεία “Comma-Seperated Values”, εξασφαλίζοντας έτσι ότι οποιαδήποτε αλλαγή στα δεδομένα αυτών των αρχείων θα αντικατοπτρίζεται άμεσα στη δημόσια ιστοσελίδα του εργαστηρίου. Με αυτό τον τρόπο μειώνεται ο φόρτος εργασία και η πιθανότητα για ανθρώπινο σφάλμα, κατά την διαδικασία ενημέρωσης της ιστοσελίδας, ενώ παράλληλα αυξάνεται η ταχύτητα και η συνέπεια με την οποία πραγματοποιείται.

1.3 Συνήθεις Μεθοδολογίες Υλοποίησης

Το πιο σύνηθες με τέτοια συστήματα είναι να υλοποιούνται με τη χρήση συστημάτων διαχείρισης περιεχομένου (Content Management Systems,CMS) ή δυναμικών πλαισίων ιστού όπως το Django ή το Laravel ή ακόμα και με χειροκίνητη επεξεργασία HyperText Markup Language (HTML). Τέτοιες προσεγγίσεις όμως φέρουν τις δικές τους δυσκολίες όπως είναι οι πολύπλοκες ρυθμίσεις που απαιτούνται κατά τη διαμόρφωση του διακομιστή (server),η συνεχή παρέμβαση και τεχνική υποστήριξη και σε ορισμένες περιπτώσεις αυτές οι μέθοδοι κρίνονται υπερβολικά σύνθετες ή και απαιτητικές για τις πραγματικές ανάγκες των εργαστηρίων. Για αυτούς τους λόγους σε αυτή τη διπλωματική εργασία πάρθηκε η απόφαση να αξιοποιηθούν σενάρια διακομιστή (server-side scripts) χαμηλής πολυπλοκότητας, προτύπων (templates) και ανάλυσης δεδομένων (data parsing) CSV για την δημιουργία ευανάγνωστων και εύχρηστων στατικών σελίδων HTML, προσφέροντας μια πιο απλή και αυτοματοποιημένη λύση χωρίς να θυσιάζεται η απόδοση, η ταχύτητα και η ευελιξία.

2 Ανασκόπηση των Τεχνολογιών που χρησιμοποιήθηκαν και Θεωρητικό υπόβαθρο

Σκοπός του κεφαλαίου είναι η ανάλυση σε βάθος των τεχνολογιών, των συστημάτων και των μεθόδων που χρησιμοποιήθηκαν σε αυτή τη διπλωματική καθώς και τους λόγους για τους οποίους επιλέχθηκαν.

2.1 Επισκόπηση των εργαλείων και τεχνολογιών που χρησιμοποιήθηκαν

Η επιλογή, των τεχνολογιών ιστού που χρησιμοποιήθηκαν, έγινε με κύριο γνώμονα την απλότητα τους, την εύκολη συντήρηση τους σε βάθος χρόνου και τη συμβατότητα τους με περιβάλλοντα φιλοξενίας στατικών ιστοσελίδων.

2.1.1 Hypertext Preprocessor (PHP)

PHP είναι μια γλώσσα προγραμματισμού, που ειδικεύεται στον server-side προγραμματισμό. Επιλέχθηκε λόγω της ενσωμάτωσης της με τη γλώσσα HTML, την ευρεία γκάμα εργαλείων που διαθέτει για χειρισμούς αρχείων και για την συμβατότητα της με σχεδόν όλους τους διακομιστές ιστού. Η ευέλικτη φύση της γλώσσας και το γεγονός ότι δεν χρειάζεται μια πλήρη βάση δεδομένων για τη σωστή λειτουργία της, αλλά και το χαμηλό λειτουργικό της κόστος την έκαναν την ιδανική επιλογή όσων αφορά την χρήση των αυτοματοποιημένων scripts που απαιτούνται.[2]

Στο πλαίσιο της εργασίας:

- Βοηθάει στην ανάκτηση των αρχείων CSV από το Dropbox μέσω της εντολής:cURL
- Επεξεργάζεται δεδομένα με χρήση της εντολής fgetsv
- Εισάγει δυναμικά τα δεδομένα σε πρότυπα HTML
- Συμβάλλει στην δημιουργία των στατικών ιστοσελίδων HTML

2.1.2 HTML

Η HTML είναι μια γλώσσα σήμανσης (markup language) που χρησιμοποιείται για την δημιουργία και την δομή ιστοσελίδων. Η HTML ορίζει το περιεχόμενο, αλλά και την δομή μιας σελίδας μέσω χρήσης “ετικετών” (tags), όπως επίσης μπορεί και να συνδέει σε άλλες σελίδες με την χρήση υπερσυνδέσμων. Μειονέκτημα της είναι ότι από μόνη της η HTML δεν μπορεί να προσφέρει δυναμική λειτουργικότητα.[3]

2.1.3 Bootstrap

Το Bootstrap είναι ένα front-end framework (πλαίσιο ανάπτυξης της πλευράς του χρήστη), το οποίο βασίζεται σε HTML, Cascading Style Sheets (CSS) και JavaScript, που χρησιμοποιήθηκε για την διάταξη και την μορφοποίηση της ιστοσελίδας. Το Bootstrap χρησιμοποιήθηκε λόγω των πλεγμάτων διάταξης, που προσφέρει, που χωρίζουν την οθόνη σε 12 στήλες με σκοπό την καλύτερη ανταπόκριση της ιστοσελίδας σε διαφορετικά μεγέθη οθονών. Επιπρόσθετα διασφαλίζει την οπτική συνέπεια των σελίδων χάρις την ομοιομορφία που παρέχει στην εμφάνιση και την τυπογραφία της, ενώ παράλληλα βοηθάει στην εύκολη ενσωμάτωση εικόνων, πλαισίων και γραφικών στοιχείων μειώνοντας έτσι την ανάγκη για ειδικά προσαρμοσμένο CSS.[4]

2.1.4 JQuery και JavaScript

Η JQuery είναι βιβλιοθήκη του JavaScript που επιτρέπει στους προγραμματιστές να γράφουν λιγότερο κώδικα για κοινές εργασίες, όπως είναι ο χειρισμός HTML στοιχείων ή ο χειρισμός γεγονότων, που σε συνδυασμό με την συμβατότητα της μεταξύ διαφορετικών περιηγητών, συμβάλλει στην μείωση του τεχνικού κόστους της ιστοσελίδας. Η JavaScript είναι μια δυναμική γλώσσα προγραμματισμού η οποία θεωρείται πλέον απαραίτητη για ανάπτυξη δυναμικών ιστοσελίδων ή ιστοσελίδων με δυναμικά χαρακτηριστικά στον σύγχρονο ιστό . Είναι συμβατή με όλους τους μοντέρνους περιηγητές και παράλληλα πολύ ευέλικτη καθώς δεν χρειάζεται μεταγλώττιση. Καθώς οι παραγόμενες σελίδες είναι στατικές, στην εργασία χρησιμοποιήθηκαν οι JavaScript επεκτάσεις που διαθέτει το Bootstrap και η JQuery για να επιτρέψουν την δυνατότητα δυναμικής πλοήγησης στον χρήστη, να βοηθήσουν στην αποκριτική συμπεριφορά της ιστοσελίδας και να βελτιώσουν την γενική αλληλεπίδραση του χρήστη με την ιστοσελίδα.[5][6]

2.1.5 Dropbox

Το Dropbox είναι μια διαδικτυακή υπηρεσία αποθήκευσης αρχείων που επιτρέπει στους χρήστες να αποθηκεύουν τα αρχεία τους στους διακομιστές της με αποτέλεσμα να μπορούν να έχουν πρόσβαση σε αυτά από οποιαδήποτε συσκευή.[7] Στην εργασία χρησιμοποιείται ως μέσο αποθήκευσης των αρχείων CSV αντί για μια πολύπλοκη βάση δεδομένων, όπου τα αρχεία κατεβάζουν τα δεδομένα από αυτό χρησιμοποιώντας τον άμεσο σύνδεσμο του (Σύνδεσμος ο οποίος τελειώνει με dl=1)[8]

2.1.6 WinSCP

Το WinSCP είναι ένα πρόγραμμα μεταφοράς αρχείων για το λειτουργικό σύστημα Windows που επιτρέπει στους χρήστες να ανταλλάσσουν αρχεία με ασφάλεια μεταξύ του τοπικού τους υπολογιστή και ενός απομακρυσμένου διακομιστή. Στην εργασία χρησιμοποιήθηκε για την μεταφορά αρχείων μέσω του File Transfer Protocol (FTP) στον διακομιστή της ιστοσελίδας, με σκοπό την δοκιμή του κώδικα σε διαδικτυακό περιβάλλον.[9]

2.1.7 XAMPP

Το XAMPP είναι ένα πακέτο λογισμικού που εγκαθιστά εύκολα έναν τοπικό εξυπηρετητή ιστού στον υπολογιστή δημιουργώντας περιβάλλον παρόμοιο με αυτό ενός πραγματικού διαδικτυακού εξυπηρετητή ιστού που χρησιμοποιείται για δοκιμές εφαρμογών PHP[10]. Το όνομα του προέρχεται από:

X-Cross-Platform (Τρέχει σε όλα τα λογισμικά)

A-Apache (Ο εξυπηρετητής ιστού)

M-MySQL ή MariaDB (Βάση δεδομένων)

P-PHP (Γλώσσα προγραμματισμού)

P-Perl (Γλώσσα προγραμματισμού)

2.1.8 Visual Studio Code

Το Visual Studio Code είναι ένας δωρεάν επεξεργαστής κώδικα που έχει αναπτυχθεί από την Microsoft. Μπορεί και υποστηρίζει πολλές γλώσσες προγραμματισμού, το οποίο είναι κρίσιμο για την συγκεκριμένη εργασία, αφού ο κώδικας γράφτηκε χρησιμοποιώντας και τη γλώσσα HTML αλλά και τη γλώσσα PHP. Επιπρόσθετα η δυνατότητα εγκατάστασης επεκτάσεων για

πρόσθετες δυνατότητες και ο ενσωματωμένος αποσφαλματωτής που διαθέτει το κάνουν ένα εύχρηστο και αξιόπιστο εργαλείο ως προς την ανάπτυξη κώδικα.[11]

2.2 Ενσωμάτωση δεδομένων από αρχεία CSV σε συστήματα ιστού

Τα αρχεία CSV αποτελούν έναν απλό αλλά ταυτόχρονα αποτελεσματικό τρόπο αποθήκευσης δεδομένων σε μορφή πίνακα. Η ευρεία χρήση τους οφείλεται στην συμβατότητα τους με οποιοδήποτε πρόγραμμα επεξεργασίας κειμένου (Microsoft Excel, Google Sheets κ.α.), στο χαμηλό κόστος συστήματος, μιας και δεν απαιτείται διακομιστής δεδομένων για την χρήση και αξιοποίηση τους, αλλά και στην ευκολία με την οποία μπορούν να χρησιμοποιηθούν από προγράμματα που είναι γραμμένα με τις γλώσσες προγραμματισμού Python ή PHP[12]. Στο πλαίσιο της εργασίας, τα αρχεία CSV χρησιμοποιούνται ως κύρια πηγή δεδομένων για την αυτόματη δημιουργία του περιεχομένου των ιστοσελίδων, τα αρχεία αυτά φιλοξενούνται εξ αποστάσεως (remote host) στην εφαρμογή Dropbox και ανακτώνται ύστερα από αίτημα, το οποίο γίνεται κατά την διαδικασία δημιουργίας της ιστοσελίδας. Κάθε μέρος της ιστοσελίδας χρησιμοποιεί ένα αρχείο CSV που εξυπηρετεί τις ανάγκες του, με τις γραμμές να αντιπροσωπεύουν μία ανεξάρτητη μονάδα πληροφορίας και τις στήλες να περιλαμβάνουν συγκεκριμένα πεδία που σχετίζονται με πληροφορίες σχετικές με την μονάδα, για παράδειγμα, η γραμμή είναι οι ολοκληρωμένες εργασίες που, έχουν γίνει από μέλη του ακαδημαϊκού εργαστηρίου, και μια από τις στήλες είναι το όνομα του εκάστοτε συγγραφέα της, οπότε με την προσθήκη καινούργιας γραμμής αναφερόμαστε σε μία άλλη εργασία της οποίας το όνομα του συγγραφέα θα είναι σε μία από τις στήλες. Τα πεδία των στηλών ορίζονται από τις επικεφαλίδες (headers) τους στη πρώτη γραμμή του αρχείου ενώ, παράλληλα γίνονται διαρκείς έλεγχοι για την επικύρωση των δεδομένων κάθε γραμμής, με στόχο την αποτελεσματική αντιμετώπιση κενών ή και κατεστραμμένων πεδίων αλλά και ως προς αποφυγή ενός ανθρώπινου σφάλματος. Τα βασικά πλεονεκτήματα αυτής της προσέγγισης είναι η δυνατότητα επεκτασιμότητας που προσδίδει στο σύστημα, η ευκολία με την οποία ενσωματώνονται τα αρχεία CSV στην αυτοματοποιημένη διαδικασία δημιουργίας των σελίδων και το ότι επιτρέπει να γίνεται η κεντρική διαχείριση των πληροφοριών χωρίς την χρήση κάποιας πολύπλοκης βάσης δεδομένων. Έτσι το αποτέλεσμα είναι η δημιουργία μιας ελαφριάς και προσβάσιμης λύσης που ακόμα και μέλη μιας ομάδας χωρίς κάποιο τεχνικό υπόβαθρο μπορούν να συνεισφέρουν άφοβα στην ενημέρωση και επεξεργασία των αρχείων μέσω οποιασδήποτε εφαρμογής επεξεργασίας κειμένου προτιμούν.

2.3 Στατικές και Δυναμικές Αρχιτεκτονικές Ιστοσελίδων

Στατικές ιστοσελίδες είναι οι ιστοσελίδες που, αποτελούνται από προκαθορισμένα αρχεία HTML, CSS και JavaScript, με αποτέλεσμα η δομή και η μορφή των σελίδων να εμφανίζεται

ίδια σε κάθε επισκέπτη. Τα πλεονεκτήματα μιας τέτοιας προσέγγισης είναι:

1. Ταχύτητα

Καθώς δεν απαιτείται επεξεργασία των αρχείων από την πλευρά του διακομιστή, οι σελίδες φορτώνονται άμεσα

2. Ασφάλεια

Η έλλειψη δυναμικών δυνατοτήτων αφήνει λιγότερα κενά ασφαλείας

3. Απλότητα

Μπορούν να φιλοξενηθούν ακόμα και σε απλούς διακομιστές και αφού όλο το περιεχόμενο είναι ήδη καθορισμένο στον πηγαίο κώδικα, δεν απαιτείται η χρήση κάποιας βάσης δεδομένων.

Δυναμικές χαρακτηρίζονται οι σελίδες οι οποίες δημιουργούν το περιεχόμενο τους σε πραγματικό χρόνο, συνήθως με την χρήση scripts από την πλευρά του διακομιστή. Τα πλεονεκτήματα που έχει η χρήση τους είναι τα εξής:

1. Άμεση Ενημέρωση Περιεχομένου

Ο κώδικας της ιστοσελίδας σε συνδυασμό με συστήματα διαχείρισης περιεχομένου που χρησιμοποιούνται πολλές φορές σε δυναμικές ιστοσελίδες καθιστούν την ενημέρωση του περιεχομένου ευκολότερη και πιο άμεση, ειδικά σε σύγκριση με τις στατικές ιστοσελίδες.

2. Υψηλή διαδραστικότητα με τον χρήστη

Οι δυναμικές ιστοσελίδες μπορούν να υποστηρίξουν πιο πολύπλοκες λειτουργίες στο κώδικα τους, όπως συστήματα σύνδεσης χρήστη με τα οποία του δίνεται η δυνατότητα να απολαμβάνει προσωποποιημένο περιεχόμενο χάρις τις βάσεις δεδομένων που χρησιμοποιούνται για την αποθήκευση των πληροφοριών του και άλλες λειτουργίες που μπορούν να βελτιώσουν την εμπειρία του με την ιστοσελίδα όπως φίλτρα σύνθετης αναζήτησης δεδομένων, δυνατότητα δημοσίευσής σχόλιων κ.α.

Το μεγάλο μειονέκτημα της αρχιτεκτονικής στατικών ιστοσελίδων είναι η δυσκολία με την οποία μπορούν να ενημερωθούν ειδικά αν το περιεχόμενο πρέπει να αλλάζει συχνά δημιουργεί ένα μεγάλο πονοκέφαλο στον διαχειριστή τους. Αντιθέτως τα μειονεκτήματα της αρχιτεκτονικής

δυναμικών ιστοσελίδων είναι η αυξημένη ανάγκη συντήρησης και το κόστος αυτής, καθώς η διαχείριση της απαιτεί την διαχείριση της από άτομα με το απαραίτητο τεχνικό υπόβαθρο, θέματα ασφάλειας, λόγω του ότι οι δυνατότητες δυναμικού περιεχομένου αφήνουν τρωτά σημεία στον κώδικα της ιστοσελίδας, αλλά και ζητήματα που αφορούν την φιλοξενία της ιστοσελίδας στον κατάλληλο διακομιστή που θα μπορεί να υποστηρίξει τις δυναμικές λειτουργίες της. Αυτή η εργασία υλοποιεί ένα υβριδικό σύστημα που σκοπό έχει την εκμετάλλευση στο έπακρο των πλεονεκτημάτων αυτών των δύο μεθόδων, εξαλείφοντας παράλληλα τα μειονεκτήματά τους με την δυναμική δημιουργία στατικών σελίδων. Επιτρέποντας έτσι στον διαχειριστή να ενημερώνει ευκολά και απλά το περιεχόμενο της σελίδας χωρίς το αυξημένο τεχνικό κόστος και τα ρίσκα ασφαλείας.[13][14]

3 Αρχιτεκτονική και Σχεδίαση Συστήματος

Στο παρόν κεφάλαιο γίνεται γενική επισκόπηση του συστήματος, αναλύεται η φιλοσοφία της σχεδίασης του και περιγράφονται τα βασικά μέρη του.

3.1 Επισκόπηση του Συστήματος

Ο στόχος πίσω από τον σχεδιασμό του συστήματος είναι η μετατροπή των δεδομένων από τα αρχεία CSV σε στατικές ιστοσελίδες HTML ελαχιστοποιώντας την ανάγκη για χειροκίνητες επεμβάσεις στον κώδικα διατηρώντας όμως, την ευκολία ενημέρωσης και υψηλή αναγνωσιμότητα. Η ροή εργασίας περιλαμβάνει 3 στάδια:

- **Ανάκτηση δεδομένων:** Κατεβάζει την πιο πρόσφατη έκδοση του CSV αρχείου που, περιέχει τα δεδομένα που θα χρειαστεί για την δημιουργία του περιεχόμενου από το Dropbox, με την χρήση του πακέτου εντολών cURL.
- **Δημιουργία δυναμικού περιεχομένου:** Τα δεδομένα τροφοδοτούνται στα αρχεία δημιουργίας περιεχομένου όπου, μετά από την κατάλληλη επεξεργασίας τους με χρήση της γλώσσας προγραμματισμού PHP γίνονται HTML blocks έτοιμα για να χρησιμοποιηθούν για την δημιουργία των στατικών σελίδων.
- **Δημιουργία στατικών σελίδων:** Τα HTML blocks εισάγονται στο προκαθορισμένο πρότυπο HTML και έτσι παράγονται οι στατικές σελίδες.

Αυτή η διαδικασία εκτελείται αυτόματα κατά την εκτέλεση του PHP script συντονισμού όπου με την επιτυχή ολοκλήρωση της εμφανίζει ένα μήνυμα στον χρήστη προς ενημέρωσή του.

3.2 Φιλοσοφία Σχεδίασης

Η φιλοσοφία πίσω από την σχεδίαση είχε ως γνώμονα την απλότητα, την επεκτασιμότητα και την συντηρησιμότητα του συστήματος. Σε ακαδημαϊκά περιβάλλοντα οι τεχνικές γνώσεις ανάμεσα στους χειριστές των ιστοσελίδων να διαφέρουν, για αυτό και είναι υψίστης σημασίας το εργαλείο να είναι εύχρηστο, να ενημερώνεται εύκολα και να μπορεί να προσαρμόζεται στις μελλοντικές ανάγκες που μπορεί να προκύψουν χωρίς να υπάρχει η απαίτηση για εκτεταμένες προγραμματιστικές γνώσεις. Για αυτούς τους λόγους η αρχή, που αποτέλεσε την βάση της αρχιτεκτονικής του συστήματος, είναι ο διαχωρισμός αρμοδιοτήτων (Seperation of Concerns) δηλαδή, κάθε αρχείο να έχει ένα σαφή και συγκεκριμένο ρόλο. Το αρχείο συντονισμού συντονίζει την διαδικασία δημιουργίας των σελίδων, τα αρχεία CSV αποτελούν τα δεδομένα που θα γίνουν το περιεχόμενο των σελίδων, τα αρχεία παραγωγής δυναμικού περιεχομένου κάνουν

αυτό ακριβώς που αναφέρει το όνομα τους χρησιμοποιώντας τα δεδομένα των αρχείων CSV και τέλος το αρχείο προτύπου διατάξεως είναι υπεύθυνο για την διάταξη των και την εμφάνιση που θα έχει το περιεχόμενο της σελίδας[15]. Με αυτό τον διαχωρισμό καθίσταται εύκολο:

- Να προστεθούν νέες σελίδες στον ιστότοπο, αφού το μόνο που χρειάζεται για την επέκταση του είναι ένα καινούργιο σετ αρχείου CSV και αρχείου δημιουργίας περιεχομένου.
- Να γίνουν ομοιόμορφες αλλαγές στην μορφή και την εμφάνιση της ιστοσελίδας μέσω του αρχείου προτύπου διατάξεως.
- Να γίνει ενημέρωση των δεδομένων με αλλαγές στα αρχεία CSV.
- Να ελαχιστοποιούνται τα σφάλματα και το εύρος της ζημιάς που μπορεί να προκαλέσουν, καθώς περιορίζονται στο τμήμα που προκλήθηκε το πρόβλημα αφήνοντας τα υπόλοιπα ανεπηρέαστα, διευκολύνοντας έτσι την αντιμετώπιση τους και την συντήρηση του ιστότοπου.
- Τέλος επιτρέπει στους προγραμματιστές ή χειριστές της ιστοσελίδας να εργάζονται παράλληλα σε διαφορετικά τμήματα της μειώνοντας έτσι χρόνο ανάπτυξης και ενημέρωσης της.

3.3 Περιγραφή των Βασικών Μερών του Συστήματος

Τα βασικά αρχεία του συστήματος είναι το αρχείο συντονισμού, τα αρχεία δημιουργίας περιεχομένου και το αρχείο προκαθορισμένου προτύπου διάταξης.

3.3.1 Αρχείο Συντονισμού

Το αρχείο συντονισμού λειτουργεί ως ο κύριος εκτελέσιμος μηχανισμός του συστήματος, ο ρόλος του είναι να συντονίζει την διαδικασία δημιουργίας των HTML σελίδων. Φορτώνει το αρχείο προκαθορισμένου προτύπου διάταξης και έπειτα εκτελεί το κατάλληλο αρχείο δημιουργίας περιεχομένου για κάθε σελίδα του ιστότοπου. Στη συνέχεια παίρνει το δυναμικό περιεχόμενο που παράγεται και το ενσωματώνει στο αρχείο προκαθορισμένου προτύπου διάταξης αντικαθιστώντας την μεταβλητή κράτησης θέσεως. Στο τέλος αποθηκεύει το νέο αρχείο HTML, αντικαθιστώντας παλαιότερη έκδοση αν υπάρχει, στον διακομιστή που φιλοξενούνται και τα υπόλοιπα αρχεία του ιστότοπου. Κάνοντας έτσι τις σελίδες άμεσα προσβάσιμες μέσω του περιηγητή ιστού, χωρίς την ανάγκη για δυναμική επεξεργασία.

3.3.2 Αρχείο Παραγωγής Δυναμικού Περιεχομένου

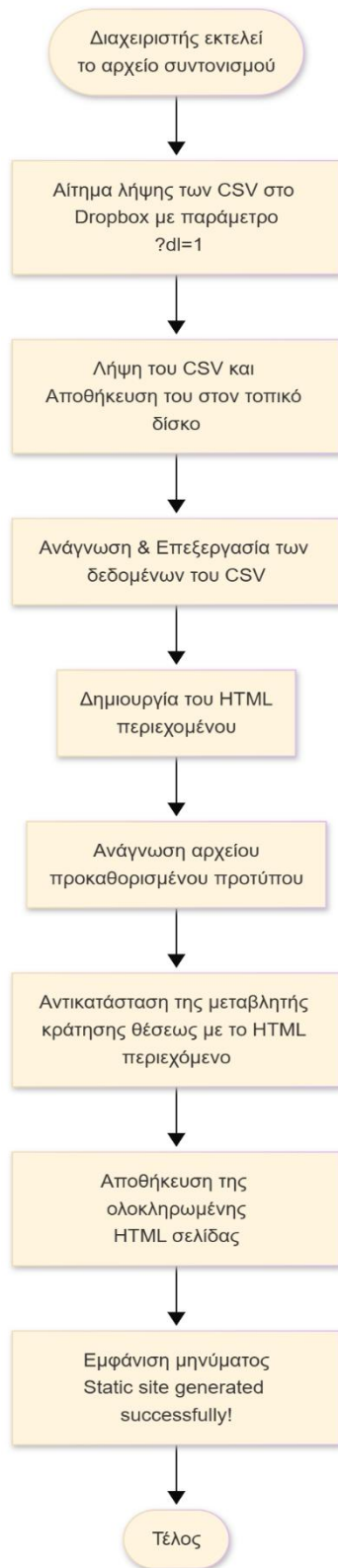
Το αρχείο παραγωγής δυναμικού περιεχομένου αναλαμβάνει την παραγωγή δυναμικού περιεχομένου βασισμένο στα δεδομένα του ανάλογου αρχείου CSV. Η διαδικασία ξεκινάει κατεβάζοντας το αρχείο CSV από το Dropbox μέσω του πακέτου εντολών cURL. Έπειτα τα δεδομένα αναλύονται μέσω της συνάρτησης `fgetcsv` της γλώσσα προγραμματισμού PHP και δημιουργούνται πίνακες στους οποίους τοποθετούνται τα δεδομένα ενώ παράλληλα γίνεται έλεγχος της εγκυρότητας τους, λαμβάνοντας υπόψιν και ελλιπή ή κατεστραμμένα πεδία. Το HTML περιεχόμενο παράγεται με ενσωμάτωση εικόνων και συνδέσμων με την κατάλληλη διαχείριση ελλιπών δεδομένων όπου χρειάζεται, ταυτοχρόνως χρησιμοποιείται η συνάρτηση `htmlspecialchars` ως μέτρο αποφυγής Cross-Site Scripting (XSS) επιθέσεων. Στο τέλος της διαδικασίας μορφοποιεί τις πληροφορίες σε HTML blocks έτοιμα για χρήση από το αρχείο προκαθορισμένου προτύπου διάταξης.

3.3.3 Αρχείο Προκαθορισμένου Προτύπου Διάταξης

Το αρχείο προκαθορισμένου προτύπου διάταξης αποτελεί το πρότυπο της δομής πάνω στο οποίο βασίζονται όλες οι παραγόμενες στατικές σελίδες HTML του ιστότοπου. Η αρχιτεκτονική του αρχείου βασίζεται στη λογική της εισαγωγής ψευδοστοιχείου, δηλαδή στην ύπαρξη ενός προκαθορισμένου κενού σημείου στον κώδικα το οποίο συμβολίζεται με μια ετικέτα και αντικαθίσταται από το παραγόμενο δυναμικό περιεχόμενο. Στο αρχείο βρίσκεται το πλαίσιο ανάπτυξης Bootstrap, για να εξασφαλίσει την αποκριτική σχεδίαση για κινητές αλλά και σταθερές συσκευές, τα απαραίτητα JavaScript scripts που έχουν να κάνουν με την εμφάνιση της ιστοσελίδας όπως επίσης και η κοινή μπάρα πλοήγησης των χρηστών που υπάρχει σε κάθε σελίδα. Έτσι εξασφαλίζεται οπτική συνέπεια στην εμφάνιση των σελίδων μειώνοντας την επανάληψη μέσα στον ίδιο τον κώδικα αφού ότι αφορά την εμφάνιση τους βρίσκεται σε ένα και μόνο αρχείο. Αυτό έχει ως αποτέλεσμα την εύκολη τροποποίηση της εμφάνισης, όταν απαιτείται και βοηθά στην αυτοματοποιημένη εισαγωγή του περιεχομένου κρατώντας και υψηλές δυνατότητες επεκτασιμότητας.

3.4 Σύνοψη Λειτουργίας Συστήματος και Κρίσιμα Σημεία Διασύνδεσης

Για την καλύτερη κατανόηση του συστήματος ακολουθεί διάγραμμα ροής που συνοψίζει την λειτουργία του:



Σχήμα 3.1 Διάγραμμα ροής της λογικής και της σειράς των βημάτων που ακολουθεί το σύστημα για τη δημιουργία μιας στατικής ιστοσελίδας

Στο παραπάνω διάγραμμα ροής φαίνονται οι ενέργειες που πραγματοποιούνται με την εκτέλεση του αρχείου συντονισμού από τον διαχειριστή. Ο πυρήνας της αυτοματοποίησης βασίζεται στη δυνατότητα της γλώσσας προγραμματισμού PHP να ανακτά δυναμικά τα δεδομένα από το Dropbox. Αυτό επιτυγχάνεται με χρήση δημόσιων συνδέσμων για άμεση λήψη από το αρχείο συντονισμού, με την προσθήκη της παραμέτρου “?dl=1” στο τέλος του συνδέσμου που δημιουργείται από το Dropbox, όπου χωρίς την χρήση της παραμέτρου ο σύνδεσμος θα οδηγούσε σε σελίδα προεπισκόπησης αντί για το ίδιο το αρχείο. Έπειτα η PHP χρησιμοποιεί την βιβλιοθήκη cURL για να “προσομοιώσει” ένα φυλλομετρητή και να κατεβάσει τα περιεχόμενα του συνδέσμου.

4 Ανάλυση του Κώδικα

Ο σκοπός του κεφαλαίου είναι η ανάλυση του κώδικα του συστήματος και των επιμέρους μερών του.

4.1 Αρχείο Συντονισμού

Παρακάτω ακολουθεί ο πλήρης κώδικας του αρχείου συντονισμού με σχόλια για καλύτερη εξήγηση των εντολών που χρησιμοποιήθηκαν.

```
<?php // Ετικέτα δήλωσης της αρχής ενός PHP script
$template = file_get_contents('template.html');//Διαβάζει το περιεχόμενο του αρχείου
template.html και το αποθηκεύει στην μεταβλητή $template

// Δημιουργία πίνακα με τις σελίδες που θα δημιουργηθούν
$pages = [
    'index' => 'index_source.php',
    'completed' => 'completedprojectsSite_source.php',
    'active' => 'curprojectsSite_source.php',
    'info' => 'infoSite_source.php',
    'team' => 'teamMembersSite_source.php',
    'printer' => 'printer_source.php',
    'workshop' => 'workshopSite_source.php',
    'announcements' => 'announcements_source.php'
];
// Ελέγχει αν υπάρχει ο φάκελος static και αν δεν υπάρχει τον δημιουργεί
if (!is_dir('static')) {
    mkdir('static', 0755);
}
// Βρόχος δημιουργίας στατικών σελίδων
foreach ($pages as $name => $source) //Επαναλαμβάνει για κάθε σελίδα στον πίνακα
$pages
{
    ob_start(); //Ενεργοποίηση προσωρινής αποθήκευσης εξόδου(output buffering)
    include $source; //Περιλαμβάνει και εκτελεί τα αρχεία παραγωγής περιεχομένου
    $content = ob_get_clean(); //Αποθηκεύει τα περιεχόμενα της προσωρινής αποθήκευσης
    εξόδου και την απενεργοποιεί
    $fullPage = str_replace('{{CONTENT}}', $content, $template); //Αντικαθιστά την ετικέτα
    {{CONTENT}} στο αρχείο προκαθορισμένου προτύπου διάταξης με το δημιουργημένο
    περιεχόμενο
    file_put_contents("static/$name.html", $fullPage); //Αποθηκεύει το τελικό HTML αρχείο
    στον φάκελο static
}
```

```
echo "Static site generated successfully!\n"; //Εμφανίζει το μήνυμα επιτυχίας όταν
ολοκληρωθεί η διαδικασία δημιουργίας σελίδων
//Ετικέτα δήλωσης τέλους PHP
?>
```

Σχήμα 4.1 Ο πλήρης κώδικας του αρχείου συντονισμού

Ο κώδικας ξεκινάει με την εντολή “<?php” που δηλώνει την αρχή ενός PHP script. Η επόμενη εντολή έχει ως σκοπό να διαβάσει το περιεχόμενο του αρχείου προκαθορισμένου πρότυπου διάταξης (*template.html*) και να το αποθηκεύσει στην μεταβλητή *\$template*. Αυτό πραγματοποιείται με χρήση της εντολής “*file_get_contents*” η οποία ανοίγει ένα αρχείο και επιστρέφει το περιεχόμενο του ως συμβολοσειρά.

```
<?php // Ετικέτα δήλωσης της αρχής ενός PHP script
$template = file_get_contents('template.html');//Διαβάζει το περιεχόμενο του αρχείου
template.html και το αποθηκεύει στην μεταβλητή $template
```

Σχήμα 4.2 Η αρχή του κώδικα του αρχείου συντονισμού

Έπειτα ορίζονται οι σελίδες που θα δημιουργηθούν με την δημιουργία ενός συνειρμικού πίνακα ο οποίος ονομάζεται “*\$pages*”. Τα στοιχεία της πρώτης στήλης του πίνακα ονομάζονται κλειδιά, τα οποία αποτελούν και το όνομα των αρχείων εξόδου και τα στοιχεία της δεύτερης στήλης του πίνακα ονομάζονται τιμές, όπου αντιστοιχούν τα αρχεία πηγής που περιέχουν το περιεχόμενο της σελίδας :

```
// Δημιουργία πίνακα με τις σελίδες που θα δημιουργηθούν
$pages = [
    'index' => 'index source.php',
    'completed' => 'completedprojectsSite source.php',
    'active' => 'curprojectsSite source.php',
    'info' => 'infoSite_source.php',
    'team' => 'teamMembersSite source.php',
    'printer' => 'printer source.php',
    'workshop' => 'workshopSite source.php',
    'announcements' => 'announcements_source.php'
];
```

Σχήμα 4.3 Το τμήμα του κώδικα που κατασκευάζει τον πίνακα με τις σελίδες που θα δημιουργηθούν

Στη συνέχεια ελέγχει αν υπάρχει ο φάκελος *static* και αν όχι τον δημιουργεί. Αυτό επιτυγχάνεται με την χρήση της συνάρτησης “*is_dir('static')*” που ελέγχει αν υπάρχει ο φάκελος με το όνομα “*static*”. Κανονικά αυτή η συνάρτηση επιστρέφει “*TRUE*” αν ο φάκελος υπάρχει και “*FALSE*” αν ο φάκελος δεν υπάρχει για αυτό αξιοποιείται ο τελεστής άρνησης ώστε αν η συνάρτηση “*is_dir*” επιστρέψει “*FALSE*”, δηλαδή ο φάκελος δεν υπάρχει, θα εκτελεστεί η συνθήκη “*if*”. Η συνάρτηση “*mkdir*” δημιουργεί έναν νέο φάκελο με όνομα το πρώτο όρισμα που βάζουμε (*static*) και στο δεύτερο όρισμα τον αριθμό 0755. Αυτός ο αριθμός ορίζει τα δικαιώματα

(permissions) του φακέλου, τα οποία καθορίζονται ως οκταδικός αριθμός, εξ' ου και το πρόθεμα 0 στον αριθμό 755. Το 7 αναφέρεται στον ιδιοκτήτη-δημιουργό του φακέλου, ο οποίος έχει δικαιώματα ανάγνωσης, εγγραφής και εκτέλεσης στο φάκελο, ο αριθμός 5 αναφέρεται σε ομάδα χρηστών που τους ανήκει ο φάκελος αλλά δεν είναι αυτοί που τον δημιούργησαν και έχουν δικαιώματα ανάγνωσης και εκτέλεσης τέλος το δεύτερο 5 αναφέρεται σε οποιονδήποτε άλλο χρήστη του συστήματος δεν ανήκει στις πρώτες 2 κατηγορίες και έχουν δικαιώματα ανάγνωσης και εκτέλεσης. Ο λόγος που μόνο ο ιδιοκτήτης έχει δικαιώματα εγγραφής είναι για την καλύτερη δυνατή ασφάλεια του συστήματος.

```
// Ελέγχει αν υπάρχει ο φάκελος static και αν δεν υπάρχει τον δημιουργεί  
if (!is_dir('static')) {  
    mkdir('static', 0755);  
}
```

Σχήμα 4.4 Το τμήμα του κώδικα που ελέγχει αν υπάρχει η δημιουργεί τον φάκελο static

Ακολουθώντας έχουμε τον βρόχο επανάληψης, που επεξεργάζεται κάθε σελίδα. Ο βρόχος ξεκινά με την εντολή “foreach” (\$pages as \$name => \$source), ο σκοπός της οποίας είναι να επαναλάβει τις εντολές που ακολουθούν για κάθε σελίδα στον πίνακα “\$pages” και να δημιουργήσει το αντίστοιχο αρχείο HTML. Για κάθε στοιχείο του πίνακα, εκχωρεί από τον συνειρμικό πίνακα “\$pages” το αντίστοιχο κλειδί στην μεταβλητή “\$name” και την αντίστοιχη τιμή στην μεταβλητή “\$source”. Επόμενη είναι η συνάρτηση “ob_start” που ενεργοποιεί την προσωρινή αποθήκευση εξόδου έτσι ώστε όλες οι επόμενες εξοδοί να μην στέλνονται απευθείας στον διακομιστή, δίνοντας μας την δυνατότητα να τις επεξεργαστούμε κατάλληλα. Στη συνέχεια με την εντολή “include” εκτελείται το αρχείο παραγωγής περιεχομένου και τα αποτελέσματα του αποθηκεύονται στην προσωρινή μνήμη την οποία ενεργοποιήσαμε προηγουμένως. Κατόπιν στην γραμμή “\$content = ob_get_clean()” αποθηκεύεται το περιεχόμενο της προσωρινής μνήμης στην μεταβλητή “\$content” και την απενεργοποιεί. Μετέπειτα στη γραμμή “\$fullPage = str_replace('{{CONTENT}}', \$content, \$template)” αντικαθίσταται η προσωρινή ετικέτα “{{CONTENT}}” στο αρχείο προκαθορισμένου προτύπου διάταξης με το περιεχόμενο της σελίδας. Η συνάρτηση “str_replace” αναζητά την συμβολοσειρά “{{CONTENT}}” στο αρχείο “\$template” και την αντικαθιστά με το περιεχόμενο της μεταβλητής “\$content”.

```
// Βρόχος δημιουργίας στατικών σελίδων  
foreach ($pages as $name => $source) //Επαναλαμβάνει για κάθε σελίδα στον πίνακα  
$pages  
{  
    ob_start(); //Ενεργοποίηση προσωρινής αποθήκευσης εξόδου(output buffering)  
    include $source; //Περιλαμβάνει και εκτελεί τα τα αρχεία παραγωγής περιεχομένου  
    $content = ob_get_clean(); //Αποθηκεύει τα περιεχόμενα της προσωρινής αποθήκευσης  
εξόδου και την απενεργοποιεί  
}
```

```

    $fullPage = str_replace('{{CONTENT}}', $content, $template); //Αντικαθιστά την ετικέτα
    {{CONTENT}} στο αρχείο προκαθορισμένου προτύπου διάταξης με το δημιουργημένο
    περιεχόμενο
    file_put_contents("static/$name.html", $fullPage); //Αποθηκεύει το τελικό HTML αρχείο
    στον φάκελο static
}

```

Σχήμα 4.5 Το τμήμα του κώδικα που δημιουργεί τις στατικές σελίδες

Στο τέλος εμφανίζει το μήνυμα επιτυχίας όταν τελειώσει η διαδικασία.

```

echo "Static site generated successfully!\n"; //Εμφανίζει το μήνυμα επιτυχίας όταν
ολοκληρωθεί η διαδικασία δημιουργίας σελίδων
//Ετικέτα δήλωσης τέλους PHP
?>

```

Σχήμα 4.6 Το τελευταίο τμήμα του κώδικα του αρχείου συντονισμού

4.2 Αρχείο Προκαθορισμένου Προτύπου Διάταξης

Παρακάτω ακολουθεί ο πλήρης κώδικας του αρχείου προκαθορισμένου προτύπου διάταξης με σχόλια για καλύτερη εξήγηση των εντολών που εντολών που χρησιμοποιήθηκαν.

```

<!DOCTYPE html> <!--Ορισμός έκδοσης HTML-->
<html lang="en"> <!--Ορίζει την αρχή του HTML εγγράφου και ότι η γλώσσα είναι αγγλικά-->
<head><!--Αρχή βρόγχου head-->
    <!--Ορίζει τον τίτλο της σελίδας-->
    <title>Microengineering</title>
    <!--Ορίζει την κωδικοποίηση των χαρακτήρων της σελίδας με το πρωτόκολλο-->
    <meta charset="utf-8">
    <!--Κάνει την σελίδα αποκριτική και την προσαρμόζει για κινητές συσκευές-->
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!--Φορτώνει το Bootstrap Framework-->
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
    <!--Φορτώνει την βιβλιοθήκη JQuery-->
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
    <!--Φορτώνει τα JavaScript στοιχεία του Bootstrap-->
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
</head> <!--Τέλος βρόγχου head -->
<body> <!--Αρχή βρόγχου body-->

    <!--Πλοήγηση-Δημιουργία σταθερής μπάρας πλοήγησης-->
    <nav class="navbar navbar-inverse navbar-static-top">
        <!--Περιέχει και κεντράρει το περιεχόμενο της πλοήγησης-->

```

```

<div class="container">
  <!--Δημιουργεί το κουμπί "hamburger" (☰) για αναπτυσσόμενη πλοήγηση σε μικρές
οθόνες-->
  <div class="navbar-header">
    <!--Ιδιότητες του κουμπιού -->
    <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1">
      <!--Δημιουργούν τις 3 γραμμές -->
      <span class="sr-only">Toggle navigation</span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
      <span class="icon-bar"></span>
    </button>
  </div>

  <!-- Συγκεντρώνει τους συνδέσμους πλοήγησης και άλλα στοιχεία για εμφάνιση/απόκρυψη
-->
  <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
    <!--Δημιουργεί την κύρια λίστα πλοήγησης-->
    <ul class="nav navbar-nav" style="font-size: 16px; white-space: nowrap;">
      <li><a href="index.html">Home</a></li>
      <li><a href="completed.html">Completed Projects</a></li>
      <li><a href="active.html">Active Projects</a></li>
      <li><a href="info.html">Useful Info</a></li>
      <li><a href="team.html">Team Members</a></li>
      <li><a href="printer.html">Printing Facility</a></li>
      <li><a href="workshop.html">Workshop</a></li>
      <li><a href="announcements.html">Announcements</a></li>
    </ul>
  </div>
</div>
</nav>

<!-- Ετικέτα που αντικαθίσταται από το δυναμικό περιεχόμενο -->
{{CONTENT}}
<!--Πηγή του πρότυπου εμφάνισης-->
<a style="color:#b5bec9;font-size:0.9em;float:right;"
href="https://github.com/mavroudisv/plain-academic">Website Template: Plain
Academic</a>
</body> <!--Τέλος βρόγχου body-->
</html> <!--Τέλος βρόγχου html -->

```

Σχήμα 4.7 Ο πλήρης κώδικας του αρχείου προκαθορισμένου προτύπου διάταξης

Ο κώδικας ξεκινάει με την “<!DOCTYPE html>” που δηλώνει ότι το αρχείο είναι τύπου HTML5 και συνεχίζει με την “<html lang=“en”>” που ορίζει την αρχή του εγγράφου και δηλώνει ότι η γλώσσα που χρησιμοποιείται είναι η αγγλική βοηθώντας έτσι τους διακομιστές και τις μηχανές αναζήτησης να κατανοήσουν την γλώσσα του περιεχομένου.

```
<!DOCTYPE html> <!--Ορισμός έκδοσης HTML-->
<html lang="en"> <!--Ορίζει την αρχή του HTML εγγράφου και ότι η γλώσσα είναι αγγλικά-->
```

Σχήμα 4.8 Η αρχή του κώδικα του αρχείου προκαθορισμένου προτύπου διάταξης

Έπειτα ξεκινάει η ενότητα “<head>” η οποία περιέχει τα μεταδεδομένα και τις πληροφορίες του αρχείου, τα στοιχεία εντός της ενότητάς δεν εμφανίζονται απευθείας στην σελίδα αλλά είναι απαραίτητα για την λειτουργία της. Το “<title>Microengineering</title>” ορίζει τον τίτλο της σελίδας και με την “<meta charset=“utf-8”>” ορίζεται η κωδικοποίηση των χαρακτήρων της με το πρωτόκολλο UCS Transformation Format 8, που επιτρέπει την ορθή εμφάνιση διεθνών χαρακτήρων όπως είναι οι ελληνικοί. Στη συνέχεια η “<meta name=“viewport” content=“width=device-width, initial-scale=1”>” κάνει την σελίδα αποκριτική και την προσαρμόζει για κινητές συσκευές, ρυθμίζοντας το πλάτος της σελίδας ώστε να αντιστοιχεί στο πλάτος της συσκευής (device-width) και ορίζοντας την αρχική κλίμακα (initial-scale=1). Οι επόμενες 3 εντολές φορτώνουν το Bootstrap CSS framework, την βιβλιοθήκη JQuery και τα JavaScript εξαρτήματα του Bootstrap αντιστοίχως.

```
<head><!--Αρχή βρόγχου head-->
<!--Ορίζει τον τίτλο της σελίδας-->
<title>Microengineering</title>
<!--Ορίζει την κωδικοποίηση των χαρακτήρων της σελίδας με το πρωτόκολλο-->
<meta charset="utf-8">
<!--Κάνει την σελίδα αποκριτική και την προσαρμόζει για κινητές συσκευές-->
<meta name="viewport" content="width=device-width, initial-scale=1">
<!--Φορτώνει το Bootstrap Framework-->
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
<!--Φορτώνει την βιβλιοθήκη JQuery-->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
<!--Φορτώνει τα JavaScript στοιχεία του Bootstrap-->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
</head> <!--Τέλος βρόγχου head -->
```

Σχήμα 4.9 Το τμήμα του κώδικα του αρχείου προκαθορισμένου προτύπου διάταξης που περιέχει το βρόγχο “<head>”

Ακολούθως ξεκινάει ο βρόχος “<body>” που περιέχει το κυρίως περιεχόμενο της σελίδας. Η πρώτη γραμμή του βρόχου είναι η “<nav class=“navbar navbar-inverse navbar-static-top”>” η οποία δημιουργεί την μπάρα πλοήγησης, μαύρου χρώματος (navbar-inverse), την κρατάει σταθερή στην κορυφή της σελίδας (navbar-static-top) και περιέχει τους συνδέσμους για τις

διάφορες σελίδες του ιστότοπου. Συνεχίζοντας με την “<div class=“container”> η οποία περιέχει, κεντράρει και παρέχει συνεπές πλάτος στο περιεχόμενο της πλοήγησης. Ακολούθως η “<div class=“navbar-header”>” είναι απαραίτητη για την σωστή λειτουργία της αποκριτικής μπάρας πλοήγησης σε κινητές συσκευές καθώς δημιουργεί το κουμπί “hamburger” (☰) που, εμφανίζεται όταν η πλοήγηση “κρύβεται” λόγω του μεγέθους της οθόνης. Μετέπειτα το κουμπί και οι ιδιότητες του ορίζονται στη γραμμή “<button type=“button” class=“navbar-toggle collapsed” data-toggle=“collapse” data-target=“#bs-example-navbar-collapse-1”>” όπου δηλώνεται ο τύπος του κουμπιού (type=“button”), ορίζεται ότι το κουμπί ελέγχει την εμφάνιση της πλοήγησης (class=“navbar-toggle”), το μενού θα είναι αρχικά κλειστό και θα χρειαστεί κλικ για να ανοίξει (collapsed), ενεργοποιεί την λειτουργία σύμπτυξης/ανάπτυξης του μενού από το Bootstrap (data-toggle=“collapse”) και σχετίζει το κουμπί με το μενού πλοήγησης που έχει την ταυτότητα “bs-example-navbar-collapse-1” το οποίο θα δούμε παρακάτω (data-target=“#bs-example-navbar-collapse-1”). Στο τέλος του τμήματος κώδικα που αναφέρεται στο κουμπί παρέχεται κείμενο που είναι ορατό μόνο στις οθόνες ανάγνωσης για την διευκόλυνση ατόμων με προβλήματα όρασης (Toggle navigation) και οι 3 γραμμές “” που χρησιμοποιούνται για την οπτική απεικόνιση του κουμπιού δημιουργώντας τις 3 γραμμές του κουμπιού “hamburger” (☰). Επόμενη είναι η γραμμή “<div class=“collapse navbar-collapse” id=“bs-example-navbar-collapse-1”>” που είναι το “κουτί” που περιέχει τους συνδέσμους πλοήγησης και ελέγχεται από το κουμπί “hamburger”, αυτό επιτυγχάνεται χάρις την σύνδεση των δύο, μέσω των παραμέτρων “id” και “data-target” αντίστοιχα. Στη συνέχεια χρησιμοποιώντας την “<ul class=“nav navbar-nav” style=“font-size: 16px; white-space: nowrap;”>” δημιουργείται η κύρια λίστα πλοήγησης. Η κλάση “nav navbar-nav” αφορά την διάταξη των συνδέσμων της λίστας πλοήγησης και πιο συγκεκριμένα τους κάνει να εμφανίζονται οριζόντια σε υπολογιστές και κατακόρυφα σε κινητές συσκευές όταν ανοίγει το “hamburger” μενού. Οι επόμενες παράμετροι αφορούν το στυλ της λίστας πλοήγησης, ορίζοντας το μέγεθος της γραμματοσειράς των συνδέσμων στα 16 pixels (font-size: 16px) και αποτρέποντας την αναδίπλωση του κειμένου (white-space: nowrap), που μπορεί να συμβεί όταν σε μικρές οθόνες το κείμενο του συνδέσμου “σπάει” σε 2 γραμμές για να φαίνεται στην οθόνη. Έτσι η χρήση της παραμέτρου “nowrap” εγγυάται ότι ο σύνδεσμος θα παραμείνει σε μία γραμμή ακόμη και αν δεν χωράει, εμφανιζόμενος οριζόντια αξιοποιώντας μια μπάρα κύλισης, κρατώντας έτσι τη διάταξη της σελίδας άθικτη ακόμα και σε μικρές οθόνες. Οι επόμενες 8 γραμμές είναι οι σύνδεσμοι από τους οποίους αποτελείται η κύρια λίστα πλοήγησης. Στη συνέχεια έχουμε την προσωρινή ετικέτα {{CONTENT}} που αντικαθίσταται από το δυναμικό περιεχόμενο κάθε σελίδας. Στο τέλος του αρχείου με την “Website Template: Plain Academic” προστίθεται ένας μικρός γκρι σύνδεσμος στο κάτω μέρος δεξιά μέρος της σελίδας που περιέχει την πηγή του προτύπου και με τις ετικέτες “</body>” και “<html>” κλείνουν οι βρόγχοι “body” και “html” αντιστοίχως.

```

<body> <!--Αρχή βρόγχου body-->

<!--Πλοήγηση-Δημιουργία σταθερής μπάρας πλοήγησης-->
<nav class="navbar navbar-inverse navbar-static-top">
  <!--Περιέχει και κεντρώνει το περιεχόμενο της πλοήγησης-->
  <div class="container">
    <!--Δημιουργεί το κουμπί "hamburger" (☰) για αναπτυσσόμενη πλοήγηση σε μικρές
οθόνες-->
    <div class="navbar-header">
      <!--Ιδιότητες του κουμπιού -->
      <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
data-target="#bs-example-navbar-collapse-1">
        <!--Δημιουργούν τις 3 γραμμές -->
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
    </div>

    <!-- Συγκεντρώνει τους συνδέσμους πλοήγησης και άλλα στοιχεία για εμφάνιση/απόκρυψη
-->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <!--Δημιουργεί την κύρια λίστα πλοήγησης-->
      <ul class="nav navbar-nav" style="font-size: 16px; white-space: nowrap;">
        <li><a href="index.html">Home</a></li>
        <li><a href="completed.html">Completed Projects</a></li>
        <li><a href="active.html">Active Projects</a></li>
        <li><a href="info.html">Useful Info</a></li>
        <li><a href="team.html">Team Members</a></li>
        <li><a href="printer.html">Printing Facility</a></li>
        <li><a href="workshop.html">Workshop</a></li>
        <li><a href="announcements.html">Announcements</a></li>
      </ul>
    </div>
  </div>
</nav>

<!-- Ετικέτα που αντικαθίσταται από το δυναμικό περιεχόμενο -->
{{CONTENT}}
<!--Πηγή του πρότυπου εμφάνισης-->
<a style="color:#b5bec9;font-size:0.9em;float:right;"
href="https://github.com/mavroudisv/plain-academic">Website Template: Plain
Academic</a>
</body> <!--Τέλος βρόγχου body-->

```

```
</html> <!--Τέλος βρόγχου html -->
```

Σχήμα 4.10 Το τμήμα του κώδικα του αρχείου προκαθορισμένου προτύπου που περιέχει το βρόγχο “<body>”

4.3 Αρχεία Παραγωγής Περιεχομένου

Ο ιστότοπος αποτελείται από 8 σελίδες με καθεμία από αυτές να έχει το δικό της αρχείο παραγωγής περιεχομένου. Οι σελίδες αυτές είναι οι: Home, Completed Projects, Active Projects, Useful Info, Team Members, Printing Facility, Workshop, Announcements. Τα 4 από τα 8 αρχεία του πηγαίου κώδικα (Active Projects, Useful Info, Team Members, Printing Facility) ακολουθούν ένα κοινό μοτίβο υλοποίησης το οποίο μπορεί να αναλυθεί σε 4 στάδια.

4.3.1 Ανάκτηση δεδομένων (Data Fetching)

Τα αρχεία ξεκινάνε ορίζοντας το Uniform Resource Locator (URL) του αρχείου του Dropbox από το οποίο θα πάρουν δεδομένα βάζοντας στο τέλος του αρχείου την κατάληξη dl=1 ώστε να εξασφαλίσουν, ότι ο σύνδεσμος θα κατεβάσει το αρχείο αντί να ανοίξει την προεπισκόπηση του. Στη συνέχεια ξεκινάει την λήψη του CSV με χρήση των εντολών cURL, ξεκινώντας με την “\$ch=curl_init()” όπου, “\$ch” είναι ένα αντικείμενο (handle) που θα χρησιμοποιηθεί για τις επόμενες cURL λειτουργίες και το “curl_init()” η συνάρτηση που ξεκινάει τη συνεδρία cURL. Έπειτα με χρήση της συνάρτησης “curl_setopt()” ορίζονται οι επιλογές για τη cURL, όπου στην πρώτη χρησιμοποιείται η “CURLOPT_URL” για να ορίσουμε το URL από το οποίο θέλουμε να κατεβάσουμε το αρχείο, που σε κάθε περίπτωση είναι η μεταβλητή “\$csvUrl”, που έχει οριστεί στην αρχή του αρχείου. Στη δεύτερη γίνεται χρήση της εντολής “CURLOPT_RETURNTRANSFER” που όταν είναι 1, δηλαδή αληθής, η cURL θα επιστρέψει τα δεδομένα ως συμβολοσειρά αντί να τα εμφανίσει απευθείας στην οθόνη. Στην τρίτη αξιοποιείται η “CURLOPT_FOLLOWLOCATION” που όταν είναι 1 εγγυάται ότι η cURL θα ακολουθήσει οποιαδήποτε Hypertext Transfer Protocol (HTTP) ανακατεύθυνση που μπορεί να χρησιμοποιεί το Dropbox. Κατόπιν το “curl_exec()” εκτελεί την συνεδρία και επιστρέφει τα αποτελέσματα ως συμβολοσειρά που αποθηκεύονται στην μεταβλητή “\$csvData” (\$csvData = curl_exec(\$ch);). Στο τέλος γίνεται έλεγχος σφαλμάτων χρησιμοποιώντας την συνάρτηση “curl_errno()” ως μέρος δομής ελέγχου “if” και κλείνει τη συνεδρία με χρήση της συνάρτησης “curl_close()”. Στην περίπτωση που έχει υπάρξει σφάλμα η “curl_error()” παίρνει το μήνυμα λάθους, η συνάρτηση “error_log()” γράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων του διακομιστή και η συνάρτηση “die()” τερματίζει τη διαδικασία και εμφανίζει το μήνυμα λάθους.

```
<?php
```

```
// Σύνδεσμος αρχείου CSV στο Dropbox για Direct Download
```

```
$csvUrl = "To URL του αρχείου CSV από το Dropbox";
```

```

//Αρχικοποιεί την cURL συνεδρία
$ch = curl_init();

//Ορίζει τις επιλογές για την cURL συνεδρία
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
curl_setopt($ch, CURLOPT_URL, $csvUrl);
// Επιστροφή δεδομένων ως συμβολοσειρές
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// Ακολουθεί HTTP ανακατευθύνσεις
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
$data = curl_exec($ch);

//Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Τερματισμός διαδικασίας
    die("Error: Unable to fetch the CSV file from Dropbox.");
}
//Κλείνει την cURL συνεδρία
curl_close($ch);

```

Σχήμα 4.11 Το τμήμα του κώδικα των αρχείων παραγωγής περιεχομένου στο οποίο γίνεται η ανάκτηση των δεδομένων

4.3.2 Επικύρωση και Αποθήκευση

Στη συνέχεια ξεκινάει η διαδικασία της αποθήκευσης των κατεβασμένων δεδομένων του CSV σε ένα τοπικό αρχείο του διακομιστή. Συγκεκριμένα χρησιμοποιείται η συνάρτηση “file_put_contents()” η οποία δημιουργεί ένα νέο αρχείο (ή αντικαθιστά υπάρχον), γράφει τα δεδομένα που της μεταφέρονται μέσα σε αυτό και κλείνει το αρχείο αυτόματα. Έχει σαν πρώτο όρισμα το όνομα του αρχείου που θα αποθηκευτούν τα δεδομένα (\$localCsvFile) και σαν δεύτερο όρισμα τα δεδομένα που θα γραφτούν, που είναι τα δεδομένα του CSV που κατέβηκε (\$data). Έπειτα γίνεται έλεγχος για την επιτυχία της διαδικασίας, όπου η συνάρτηση επιστρέφει τον αριθμό των bytes που γράφτηκαν στο αρχείο αν όλα πήγαν καλά ή FALSE αν υπήρξε σφάλμα (if (file_put_contents(\$localCsvFile, \$data) === false)). Αν αποτύχει η αποθήκευση το σφάλμα καταγράφεται στο αρχείο καταγραφής σφαλμάτων του διακομιστή με την συνάρτηση “error_log()”, εμφανίζει μήνυμα λάθους στο χρήστη (echo 'Error: Unable to save the CSV file.') και τερματίζει την εκτέλεση του script (exit).

```

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'Το όνομα του τοπικού αρχείου';
// Αποθήκευσε τα downloaded data στον τοπικό φάκελο
if (file_put_contents($localCsvFile, $data) === false) {

```

```

// Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων
error_log('Failed to save the downloaded CSV to local storage. ');
// Εμφανίζει μήνυμα σφάλματος
echo 'Error: Unable to save the CSV file.';
//Τερματισμός εκτέλεσης
exit;
}

```

Σχήμα 4.12 Το τμήμα του κώδικα των αρχείων παραγωγής περιεχομένου στο οποίο γίνεται η επικύρωση και η αποθήκευση των δεδομένων

4.3.3 Ανάλυση και Δόμηση Δεδομένων(Data Parsing and Structuring)

Σε αυτό το στάδιο τα δεδομένα του αρχείου CSV διαβάζονται και μετατρέπονται σε μια δομημένη μορφή για να μπορέσουν να χρησιμοποιηθούν ευκολότερα στη συνέχεια . Η διαδικασία ξεκινάει με χρήση της δομής ελέγχου “if ((\$handle = fopen(\$localCsvFile, 'r')) !== false)”, που χρησιμοποιεί την συνάρτηση “fopen()” για να ελέγξει αν το άνοιγμα του αρχείου ήταν επιτυχές πριν προχωρήσει στην ανάγνωση. Η συνάρτηση fopen(), επιστρέφει ένα δείκτη προς το αρχείο αν το άνοιγμα ήταν επιτυχές ή FALSE. Σε περίπτωση αποτυχίας εμφανίζεται στον χρήστη το μήνυμα “Error: Unable to open the local CSV file.” και τερματίζεται η εκτέλεση του αρχείου. Σε περίπτωση επιτυχίας διαβάζει την πρώτη γραμμή του CSV αρχείου που περιέχει τις επικεφαλίδες (headers) των στηλών, αξιοποιώντας την συνάρτηση “fgetcsv(\$handle, 1000, ',')” όπου παίρνει σαν πρώτη παράμετρο το δείκτη του αρχείου, σαν δεύτερη ορίζει το μέγιστο μήκος γραμμής στους 1000 χαρακτήρες και σαν τρίτη ορίζει τον διαχωριστή των πεδίων “ , ”. Έπειτα αρχικοποιείται ενός κενού πίνακας που θα χρησιμοποιηθεί για να αποθηκευτούν τα δεδομένα (π.χ. \$projects = []). Κατόπιν δημιουργείται βρόχος επανάληψης “while” ο οποίος τρέχει μέχρι να τελειώσουν οι γραμμές του αρχείου CSV, δηλαδή μέχρι η συνάρτηση “fgetcsv()” να επιστρέψει FALSE (while ((\$data = fgetcsv(\$handle, 1000, ',')) !== false)). Μέσα στον βρόγχο εξασφαλίζεται ότι κάθε γραμμή δεδομένων έχει τον ίδιο αριθμό στηλών με τις επικεφαλίδες χρησιμοποιώντας την δομή ελέγχου “if (count(\$headers) === count(\$data))” στην οποία μετριούνται πόσες επικεφαλίδες υπάρχουν (count(\$headers)) και πόσες τιμές έχει η τρέχουσα γραμμή δεδομένων (count(\$data)). Αν έχουν ακριβώς τον ίδιο αριθμό στοιχείων, τότε δημιουργείται συσχετιστικός πίνακας που χρησιμοποιεί τις επικεφαλίδες (\$headers) ως κλειδιά και τις τιμές της γραμμής δεδομένων (\$data) ως τιμές του κενού πίνακα που είχε δημιουργηθεί (\$projects = array_combine(\$headers, \$data)), ενώ αν δεν έχουν τον ίδιο αριθμό στοιχείων, το σφάλμα καταγράφεται στο αρχείο καταγραφής σφαλμάτων του διακομιστή με το μήνυμα "Skipping row with mismatched columns: " και ο πίνακας “\$data” μετατρέπεται σε συμβολοσειρά ενώνοντας τα στοιχεία με κόμμα (error_log("Skipping row with mismatched columns: ". implode(',', \$data))). Έχει χρησιμοποιηθεί μέρος του κώδικα της ενότητας Active Projects ως παράδειγμα παρακάτω:

```

//Ανάγνωση και επεξεργασία του CSV
// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    //Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν κενό πίνακα
    $teamMembers = [];

    // Διάβασε κάθε σειρά του φακέλου CSV
    while (($data = fgetcsv($handle, 1000, ',')) !== false) {
        // Παρέλειψε σειρές με ανόμοιες στήλες
        if (count($headers) === count($data)) {
            // Δημιουργία συσχετιστικού πίνακα
            $teamMembers[] = array_combine($headers, $data);
        } else {
            // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
            error_log("Skipping row with mismatched columns: " . implode(',', $data));
        }
    }
    // Κλείσε το φάκελο
    fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo "Error: Unable to open the local CSV file.";
    exit;
}
?>

```

Σχήμα 4.13 Τμήμα του κώδικα του αρχείου παραγωγής περιεχομένου της ενότητας Active Projects στο οποίο φαίνεται η διαδικασία ανάλυσης και δόμησης των δεδομένων

4.3.4 Απόδοση(Rendering)

Σε αυτό το στάδιο, τα δομημένα δεδομένα (στην μορφή των associative arrays) μετατρέπονται σε HTML κώδικα “γεμίζοντας τα κενά” του αρχείου προκαθορισμένου προτύπου. Η κύρια λογική βασίζεται σε βρόγχο επανάληψης “foreach”, ο οποίος επαναλαμβάνεται για κάθε μία από τις εγγραφές του πίνακα δεδομένων. Στην αρχή του βρόγχου, γίνεται έλεγχος για το αν υπάρχουν τα απαραίτητα πεδία του CSV χρησιμοποιώντας δομή ελέγχου “if” και την συνάρτηση “isset()”. Η συνάρτηση “isset()” ελέγχει αν οι καθορισμένες μεταβλητές/κλειδιά έχουν οριστεί και δεν είναι null. Κάθε τιμή που προέρχεται από τον πίνακα (και κατά συνέπεια από το CSV) εμφανίζεται μέσω της “echo”, που αποτελεί κατασκευή γλώσσας (language construct) της PHP και χρησιμοποιείται για την εξαγωγή δεδομένων στην οθόνη. Παράλληλα με την “echo” χρησιμοποιούνται οι συναρτήσεις “htmlspecialchars()” και “nl2br()”. Η συνάρτηση

“htmlspecialchars()” αντικαθιστά τους ειδικούς χαρακτήρες που χρησιμοποιεί η HTML για την δομή της με οντότητες HTML (HTML entities), εξασφαλίζοντας ότι τα δεδομένα θα εμφανίζονται πάντα σωστά ακόμα και όταν περιέχουν τους χαρακτήρες που χρησιμοποιεί η HTML ενώ ταυτόχρονα προστατεύει τη σελίδα από XSS επιθέσεις. Η συνάρτηση “nl2br()” χρησιμοποιείται ειδικά για κείμενο που περιέχει αλλαγές γραμμής μετατρέποντας τις σε ετικέτες HTML. Όλα τα δεδομένα εντάσσονται σε σημασιολογικά κατάλληλες HTML ετικέτες κάνοντας τον κώδικα πιο προσβάσιμο, βιώσιμο και εύκολο στην συντήρηση.

Παρακάτω παραθέτω τους πλήρεις κώδικες των ενότητων Active Projects, Useful Info, Team Members, Printing Facility με σχόλια για την επεξήγηση των λειτουργιών τους.

4.3.4.1 Active Projects

Η ενότητα Active Projects βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/active.html> και περιέχει πληροφορίες σχετικά με τις τρέχουσες εργασίες των φοιτητών, το CSV που του τροφοδοτεί δεδομένα χωρίζεται σε 5 πεδία: Title, Image, Objective, Description, Publications

Ακολουθεί ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας:

```
<?php
// Σύνδεσμος Dropbox για το CSV
$csvUrl =
'https://www.dropbox.com/scl/fi/itgurkhu9na95gk35sxsc/projects.csv?rlkey=321358z8ialqvych
oi109g75d&st=xri078gg&dl=1';
// Αρχικοποιεί την cURL συνεδρία
$ch = curl_init();

// Ορίζει τις επιλογές για την cURL συνεδρία
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
curl_setopt($ch, CURLOPT_URL, $csvUrl);
// Επιστροφή δεδομένων ως συμβολοσειρές
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// Ακολουθεί HTTP ανακατευθύνσεις
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
$data = curl_exec($ch);

// Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Τερματισμός διαδικασίας
    die("Error: Unable to fetch the CSV file from Dropbox.");
}
```

```

//Κλείνει την cURL συνεδρία
curl_close($ch);

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'curprojects.csv';

// Αποθήκευσε τα κατεβασμένα δεδομένα στον τοπικό φάκελο
if (file_put_contents($localCsvFile, $data) === false) {
    // Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων
    error_log('Failed to save the downloaded CSV to local storage. ');
    // Εμφανίζει μήνυμα σφάλματος
    echo 'Error: Unable to save the CSV file.';
    //Τερματισμός εκτέλεσης
    exit;
}
//Ανάγνωση και επεξεργασία του CSV
// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    //Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν πίνακα που να κρατάει τις εργασίες
    $projects = [];
    // Διάβασε κάθε σειρά του φακέλου CSV
    while (($data = fgetcsv($handle, 1000, ',')) !== false) {
        // Παρέλειψε σειρές με ανόμοιες στήλες
        if (count($headers) === count($data)) {
            // Δημιουργία συσχετιστικού πίνακα
            $projects[] = array_combine($headers, $data);
        } else {
            // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
            error_log("Skipping row with mismatched columns: " . implode(',', $data));
        }
    }
    // Κλείσε το φάκελο
    fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo "Error: Unable to open the local CSV file.";
    // Τερματισμός διαδικασίας
    exit;
}
?>

```

<!-- Περιεχόμενο HTML -->

```

<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της
σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Δημιουργία κλάσης active-project και χρησιμοποιεί γραμματοσειρά
18px εντός της-->
        <div class="active-project" style="font-size: 18px;">
            <!-- Επικεφαλίδα σελίδας -->
            <h2>Active Projects</h2>
            <!-- Βρόχος επανάληψης για κάθε εργασία στον πίνακα $projects -->
            <?php foreach ($projects as $project): ?>
                <?php
                // Ελέγχει ότι τα απαραίτητα πεδία Title, Image, Objective Description υπάρχουν και
δεν είναι null
                if (!isset($project['Title'], $project['Image'], $project['Objective'],
$project['Description'])) {
                    // Εάν κάποιο από αυτά δεν υπάρχει ή είναι null τότε γίνεται καταγραφή σφάλματος
στο αρχείο καταγραφής σφαλμάτων
                    error_log("Skipping incomplete project: " . print_r($project,true));
                    // Παραλείπει την τρέχουσα εργασία και πάει στην επόμενη
continue;
                }
                ?>
                <!-- Εμφανίζει τον τίτλο(Title) της εργασίας με έντονα γράμματα(<b>) -->
                <p><b><?php echo htmlspecialchars($project['Title']); ?></b></p>
                <!-- Εμφανίζει την εικόνα(Image) της εργασίας ορίζοντας το πλάτος και παρέχει
εναλλακτικό κείμενο για λόγους προσβασιμότητας -->
                "><br>

                <!-- Εμφανίζει τον σκοπό της εργασίας(Objective) -->
                <p>Purpose: <?php echo nl2br(htmlspecialchars($project['Objective']));
?></p>
                <!-- Εμφανίζει την περιγραφή της εργασίας(Description) -->
                <p>Description: <?php echo nl2br(htmlspecialchars($project['Description']));
?></p>

                <!-- Εμφανίζει τις δημοσιεύσεις -->
                <p>Publications and Dissertations:
                <!-- Δημιουργεί μια λίστα με κουκκίδες -->
                <ul>
                <?php

```

```

        // Χωρίζει το κείμενο του πεδίου "Publications" σε ξεχωριστές συμβολοσειρές
        // βάση του χαρακτήρα ";" και τις αποθηκεύει σε έναν νέο πίνακα
        $publications = explode(';', $project['Publications']);
        // Βρόχος επανάληψης ο οποίος επαναλαμβάνετε για κάθε δημοσίευση στον
        // καινούργιο πίνακα $publications
        foreach ($publications as $pub) {
            // Ελέγχει αν η δημοσίευση περιέχει τον χαρακτήρα "|"
            if (strpos($pub, '|') !== false) {
                // Χωρίζει την συμβολοσειρά σε 2 μέρη: Κείμενο($text και URL($url)
                list($text, $url) = explode('|', $pub, 2);
                // Μπαίνει ως στοιχείο στη λίστα με κουκκίδες με σύνδεσμο ($url) που ανοίγει
                // σε νέα καρτέλα(target="_blank") με κείμενο συνδέσμου($text)
                echo '<li><a href="' . htmlspecialchars(trim($url)) . '" target="_blank">'
                . htmlspecialchars(trim($text)) . '</a></li>';
            }
            else {
                // Αν δεν βρεθεί ο χαρακτήρας "|" εμφανίζει απλά
                // το κείμενο
                echo '<li>' . htmlspecialchars($pub) . '</li>';
            }
        }
    ?>
</ul>
</p>

<br>
<hr>
<br>
<!--Τέλος βρόγχου επανάληψης -->
<?php endforeach; ?>
<!--Κλείνονται ανοιχτές διαιρέσεις του κώδικα -->
</div>
</div>
</div>

```

Σχήμα 4.14 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας

4.3.4.2 Useful Info

Η ενότητα Useful Info βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/info.html> και περιέχει χρήσιμες πληροφορίες σχετικά με τις εξετάσεις των φοιτητών, το CSV που του τροφοδοτεί δεδομένα χωρίζεται σε 2 πεδία: Semester, Date

Ακολουθεί ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας:

```
<?php
```

```
// Σύνδεσμος Dropbox για το CSV
```

```
$csvUrl =
```

```
'https://www.dropbox.com/scl/fi/qsqdxhq93ytsxgeuwlqa2/exam_dates.csv?rlkey=n0570hv0vvd  
fqctev0dw2k114&dl=1';
```

```
// Αρχικοποιεί την cURL συνεδρία
```

```
$ch = curl_init();
```

```
// Ορίζει τις επιλογές για την cURL συνεδρία
```

```
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
```

```
curl_setopt($ch, CURLOPT_URL, $csvUrl);
```

```
// Επιστροφή δεδομένων ως συμβολοσειρές
```

```
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
```

```
// Ακολουθεί HTTP ανακατευθύνσεις
```

```
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
```

```
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
```

```
$data = curl_exec($ch);
```

```
// Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
```

```
if (curl_errno($ch)) {
```

```
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
```

```
    error_log('Curl error: ' . curl_error($ch));
```

```
    // Μήνυμα σφάλματος
```

```
    echo 'Error: Unable to fetch the CSV file.';
```

```
    // Τερματισμός διαδικασίας
```

```
    exit;
```

```
}
```

```
// Κλείνει την cURL συνεδρία
```

```
curl_close($ch);
```

```
// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
```

```
$localCsvFile = 'exam_dates.csv';
```

```
// Αποθήκευσε τα κατεβασμένα δεδομένα στον τοπικό φάκελο
```

```
if (file_put_contents($localCsvFile, $data) === false) {
```

```
    // Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων  
    error_log('Failed to save the downloaded CSV to local storage.');
```

```
    // Μήνυμα σφάλματος
```

```
    echo 'Error: Unable to save the CSV file.';
```

```
    // Τερματισμός διαδικασίας
```

```
    exit;
```

```
}
```

```

//Ανάγνωση και επεξεργασία του CSV
// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    //Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν πίνακα που να κρατάει τις ημερομηνίες των εξετάσεων
    $examDates = [];
    // Διάβασε κάθε σειρά του φακέλου CSV
    while (($data = fgetcsv($handle, 1000, ',')) !== false) {
        // Παρέλειψε σειρές με ανόμοιες στήλες
        if (count($headers) === count($data)) {
            // Δημιουργία συσχετιστικού πίνακα
            $examDates[] = array_combine($headers, $data);
        }
        else {
            // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
            error_log("Skipping row with mismatched columns: " . implode(',', $data));
        }
    }
    // Κλείσε το φάκελο
    fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo 'Error: Unable to open the local CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}
?>
<!-- Περιεχόμενο HTML -->
<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της
σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Δημιουργία κλάσης useful-info και χρησιμοποιεί γραμματοσειράς 18px εντός της -->
        <div class="useful-info" style="font-size: 18px;">
            <!-- Εμφανίζει επικεφαλίδα -->
            <h2>Useful Information</h2>
            <!-- Αλλαγή γραμμής(break) -->
            <br>
            <!-- Δεύτερη κεφαλίδα -->
            <h3>Dissertation Examination Dates</h3>
            <?php
            // Δημιουργεί κενό πίνακα $groupedExams

```

```

$groupedExams = [];
// Βρόχος επανάληψης που διασχίζει όλα τα στοιχεία του πίνακα $examDates
foreach ($examDates as $exam) {
    // Ελέγχεται αν τα υποχρεωτικά πεδία υπάρχουν στην τρέχουσα εγγραφή
    if (isset($exam['Semester'], $exam['Date'])) {
        // Αν υπάρχουν η τιμή του πεδίου Semester χρησιμοποιείται ως κλειδί στον πίνακα
        $groupedExams και χρησιμοποιώντας το κλειδί βρίσκει ή δημιουργεί(αν δεν υπάρχει), υπό-
        πίνακα στον οποίο προσθέτει την τιμή "$exam['Date']" στο τέλος του
        $groupedExams[$exam['Semester']][] = $exam['Date'];
    }
}
// Βρόχος επανάληψης που επαναλαμβάνετε για κάθε εξάμηνο στον πίνακα
$groupedExams
foreach ($groupedExams as $semester => $dates): ?>
    <!-- Εμφανίζει το όνομα του εξαμήνου -->
    <h4><b><?php echo htmlspecialchars($semester); ?></b></h4>
    <!-- Δημιουργείται λίστα με κουκκίδες -->
    <ul>
        <!-- Βρόχος επανάληψης που επαναλαμβάνετε για κάθε ημερομηνία στον πίνακα
$dates -->
        <?php foreach ($dates as $date): ?>
            <!-- Δημιουργείται στοιχείο λίστας(<li>) και εμφανίζει την ημερομηνία -->
            <li><?php echo htmlspecialchars($date); ?></li>
        <!-- Τέλος βρόγχου επανάληψης -->
        <?php endforeach; ?>
        <!-- Κλείνει η λίστα με κουκκίδες -->
    </ul>
    <!-- Τέλος βρόγχου επανάληψης -->
    <?php endforeach; ?>
    <!-- Κλείνονται ανοιχτές διαιρέσεις του κώδικα -->
</div>
</div>
</div>

```

Σχήμα 4.15 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Useful Info

4.3.4.3 Team Members

Η ενότητα Team Members βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/team.html> και περιέχει πληροφορίες σχετικά με τα μέλη του εργαστηρίου, το CSV που του τροφοδοτεί δεδομένα χωρίζεται σε 3 πεδία: Name, Description, Photo

Ακολουθεί ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας:

```
<?php
```

```
// Σύνδεσμος Dropbox για το CSV
```

```
$csvUrl =
```

```
'https://www.dropbox.com/scl/fi/ebti6o1nvyghdq29he35x/team_members.csv?rlkey=zd2jne9lmmv8g5hxtb8n48hia&raw=1';
```

```
// Αρχικοποιεί την cURL συνεδρία
```

```
$ch = curl_init();
```

```
// Ορίζει τις επιλογές για την cURL συνεδρία
```

```
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
```

```
curl_setopt($ch, CURLOPT_URL, $csvUrl);
```

```
// Επιστροφή δεδομένων ως συμβολοσειρές
```

```
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
```

```
// Ακολουθεί HTTP ανακατευθύνσεις
```

```
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
```

```
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
```

```
$data = curl_exec($ch);
```

```
// Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
```

```
if (curl_errno($ch)) {
```

```
    // Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων
```

```
    error_log('Curl error: ' . curl_error($ch));
```

```
    // Μήνυμα σφάλματος
```

```
    echo 'Error: Unable to fetch the CSV file.';
```

```
    // Τερματισμός διαδικασίας
```

```
    exit;
```

```
}
```

```
// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
```

```
$localCsvFile = 'team_members.csv';
```

```
// Αποθήκευσε τα downloaded data στον τοπικό φάκελο
```

```
if (file_put_contents($localCsvFile, $data) === false) {
```

```
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
```

```
    error_log('Failed to save the downloaded CSV to local storage.');
```

```
    // Μήνυμα σφάλματος
```

```
    echo 'Error: Unable to save the CSV file.';
```

```
    // Τερματισμός διαδικασίας
```

```
    exit;
```

```
}
```

```
// Ανάγνωση και επεξεργασία του CSV
```

```
// Άνοιξε το φάκελο CSV για να το διαβάσεις
```

```

if (($handle = fopen($localCsvFile, 'r')) !== false) {
    //Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν πίνακα που να κρατάει τα μέλη της ομάδας
    $teamMembers = [];
    // Διάβασε κάθε σειρά του φακέλου CSV
    while (($data = fgetcsv($handle, 1000, ',')) !== false) {
        // Παρέλειψε σειρές με ανόμοιες στήλες
        if (count($headers) === count($data)) {
            // Δημιουργία συσχετιστικού πίνακα
            $teamMembers[] = array_combine($headers, $data);
        } else {
            // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
            error_log("Skipping row with mismatched columns: " . implode(',', $data));
        }
    }
    // Κλείσε το φάκελο
    fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo "Error: Unable to open the local CSV file.";
    // Τερματισμός διαδικασίας
    exit;
}
?>

<!-- Περιεχόμενο HTML -->
<!-- Δημιουργία κλάσης team-container -->
<div class="team-container">
<!--Βρόχος επανάληψης που επαναλαμβάνεται για κάθε μέλος της ομάδας στον πίνακα
"$teamMembers" -->
<?php foreach ($teamMembers as $member): ?>
    <?php
    // Έλεγχος για το αν υπάρχουν τα απαραίτητα πεδία και δεν είναι null
    if (!isset($member['Name'], $member['Description'], $member['Photo'])) {
        // Σε περίπτωση αποτυχίας καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
        error_log("Skipping incomplete team member: " . print_r($member, true));
        // Παραλείπει την τρέχουσα εγγραφή
        continue;
    }
    ?>
    <!-- Δημιουργία κλάσης team-member -->
    <div class="team-member">
        <!-- Εμφανίζει την εικόνα, φορτώνοντας την από το φάκελο images και χρησιμοποιεί το
        όνομα του μέλους ως εναλλακτικό κείμενο για λόγους προσβασιμότητας -->

```

```

        margin: 20px 0; /* Ορίζει περιθώρια 20px πάνω κάτω και 0px
αριστερά        δεξιά */
    }
    /* Στυλ του ονόματος του team member */
    .team-member h3 {
        font-size: 20px; /* Ορίζει μέγεθος γραμματοσειράς 20 px */
        margin: 5px 0; /* Ορίζει 5px περιθώρια πάνω κάτω και 0px δεξιά αριστερά */
    }

    /* Στυλ του bio του team member */
    .team-member p {
        font-size: 16px; /* Ορίζει μέγεθος γραμματοσειράς */
        color: #555; /* Σκούρο γκρι χρώμα κειμένου */
    }

    /* Στυλ των συνδέσμων */
    .team-member a {
        display: inline-block; /* Κάνει του συνδέσμου στοιχεία block
που                μπορούν να έχουν περιθώρια */
        margin: 5px; /* Ορίζει 5px περιθώριο γύρω από κάθε σύνδεσμο */
        text-decoration: none; /* Αφαιρεί την υπογράμμιση από
τους                συνδέσμους */
        color: #007bff; /* Μπλε χρώμα για τους συνδέσμους */
    }

    .team-member a:hover {
        text-decoration: underline; /* Προσθέτει υπογράμμιση όταν ο χρήστης περάσει το
ποντίκι του πάνω από τον σύνδεσμο */
    }
<!--Τέλος CSS Styling →
</style>
<!--Τέλος βρόγχου επανάληψης →
<?php endforeach; ?>

```

Σχήμα 4.16 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Team Members

4.3.4.4 Printing Facility

Η ενότητα Printing Facility βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/printer.html> και περιέχει πληροφορίες για την εγκατάσταση εκτύπωσης του εργαστηρίου, το CSV που του τροφοδοτεί δεδομένα χωρίζεται σε 4 πεδία: Title, Description, Email, Image

Ακολουθεί ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας:

```

<?php
// Σύνδεσμος Dropbox για το CSV
$csvUrl =
'https://www.dropbox.com/scl/fi/2o8drl2wqjh5v62r32cvm/printer.csv?rlkey=wpry6brpt0mivy4
zgnsibb23s&dl=1';

//Αρχικοποιεί την cURL συνεδρία
$ch = curl_init();
//Ορίζει τις επιλογές για την cURL συνεδρία
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
curl_setopt($ch, CURLOPT_URL, $csvUrl);
// Επιστροφή δεδομένων ως συμβολοσειρές
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// Ακολουθεί HTTP ανακατευθύνσεις
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
$data = curl_exec($ch);

//Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Μήνυμα σφάλματος
    echo 'Error: Unable to fetch the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

//Κλείνει την cURL συνεδρία
curl_close($ch);

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'printing_facility.csv';

// Αποθήκευσε τα κατεβασμένα δεδομένα στον τοπικό φάκελο
if (file_put_contents($localCsvFile, $data) === false) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Failed to save the downloaded CSV to local storage. ');
    // Μήνυμα σφάλματος
    echo 'Error: Unable to save the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

//Ανάγνωση και επεξεργασία του CSV

```

```

// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    // Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν πίνακα που να κρατάει τα δεδομένα του printing facility
    $facilityData = [];
    // Διάβασε κάθε σειρά του φακέλου CSV
    while (($data = fgetcsv($handle, 1000, ',')) !== false) {
        // παρέλειψε σειρές με ανόμοιες στήλες
        if (count($headers) === count($data)) {
            // Δημιουργία συσχετιστικού πίνακα
            $facilityData[] = array_combine($headers, $data);
        } else {
            // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
            error_log("Skipping row with mismatched columns: " . implode(',', $data));
        }
    }
    // Κλείσε τον φάκελο
    fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo "Error: Unable to open the local CSV file.";
    exit;
}
?>

<!-- Περιεχόμενο HTML -->
<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Ελέγχει αν ο πίνακας $facilityData είναι κενός και αν δεν είναι συνεχίζει -->
        <?php if (!empty($facilityData)): ?>
            <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε εγκατάσταση εκτύπωσης στον πίνακα $facilityData -->
            <?php foreach ($facilityData as $facility): ?>
                <?php
                // Έλεγχος για το αν υπάρχουν τα απαραίτητα πεδία και δεν είναι null
                if (!isset($facility['Title'], $facility['Description'], $facility['Email'], $facility['Image'])) {
                    // Σε περίπτωση αποτυχίας καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
                    error_log("Skipping incomplete printing facility data: " . print_r($facility, true));

```

```

        // Παραλείπει την τρέχουσα εγγραφή
        continue;
    }
    ?>
    <!-- Εμφανίζει τον τίτλο -->
    <h2><?php echo htmlspecialchars($facility['Title']); ?></h2>
    <!-- Δημιουργεί την κλάση active-project και χρησιμοποιεί γραμματοσειρά 18px εντός
της -->
    <div class="active-project" style="font-size: 18px;">
        <!-- Εμφανίζει την εικόνα -->
        
        <!-- Αλλαγή γραμμής -->
        <br>
        <!-- Εμφανίζει την περιγραφή -->
        <p><?php echo nl2br(htmlspecialchars($facility['Description'])); ?></p>
        <p>
            To book or order a print, please contact us at:
            <!-- Δημιουργία συνδέσμου με μαύρο κείμενο(style), που θα ανοίγει στο
προεπιλεγμένο πρόγραμμα ηλεκτρονικής αλληλογραφίας του χρήστη και εμφανίζει την διεύθυνση
e-mail από τον πίνακα $facility -->
            <a style="color:#000;" href="mailto:<?php echo
htmlspecialchars($facility['Email']); ?>">
                <!-- Εμφανίζει την διεύθυνση e-mail -->
                <?php echo htmlspecialchars($facility['Email']); ?>
            </a>
        </p>
    </div>
    <!-- Τέλος βρόγχου επανάληψης -->
    <?php endforeach; ?>
    <!-- Σε περίπτωση που ο πίνακας $facilityData είναι κενός -->
    <?php else: ?>
        <h2>Printing Facility</h2>
        <p>Error: No facility data available.</p>
    <!-- Τέλος δομής ελέγχου -->
    <?php endif; ?>
    <!-- Κλείνει τις ανοιχτές διαιρέσεις του κώδικα -->
    </div>
</div>

```

Σχήμα 4.17 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Printing Facility

4.3.5 Εξαιρέσεις

Ενώ τα προηγούμενα αρχεία παραγωγής περιεχομένου χρησιμοποιούν το ίδιο μοτίβο υλοποίησης, άλλα το επεκτείνουν ή απαιτούν ειδική επεξεργασία λόγω της φύσης των δεδομένων τους ή της διαφορετικής δομής του αρχείου CSV.

4.3.5.1 Completed Projects

Η ενότητα Printing Facility βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/completed.html> και περιέχει πληροφορίες για τις ολοκληρωμένες εργασίες των φοιτητών, το CSV που του τροφοδοτεί δεδομένα χωρίζεται σε 4 πεδία: Year, Author, Title, Link

Ακολουθεί ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας:

```
<?php
// Σύνδεσμος Dropbox για το CSV
$csvUrl =
'https://www.dropbox.com/scl/fi/lxq2o48dhu15fzcgi0f3h/completed_projects.csv?rlkey=a10ft0
2mdknkwnx2wtwsn69e5&dl=1';
// Αρχικοποιεί την cURL συνεδρία
$ch = curl_init();
// Ορίζει τις επιλογές για την cURL συνεδρία
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
curl_setopt($ch, CURLOPT_URL, $csvUrl);
// Επιστροφή δεδομένων ως συμβολοσειρές
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// Ακολουθεί HTTP ανακατευθύνσεις
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
$data = curl_exec($ch);

// Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Μήνυμα σφάλματος
    echo 'Error: Unable to fetch the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

// Κλείνει την cURL συνεδρία
```

```

curl_close($ch);

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'completed_projects.csv';

// Αποθηκεύει τα κατεβασμένα δεδομένα στον τοπικό φάκελο
if (file_put_contents($localCsvFile, $data) === false) {
    // Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων
    error_log('Failed to save the downloaded CSV to local storage. ');
    // Μήνυμα σφάλματος
    echo 'Error: Unable to save the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

// Ανάγνωση και επεξεργασία του CSV
// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    // Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν πίνακα που να κρατάει τις ολοκληρωμένες εργασίες
    $completedProjects = [];
    // Διάβασε κάθε σειρά του φακέλου CSV
    while (($data = fgetcsv($handle, 1000, ',')) !== false) {
        // Παρέλειψε σειρές με ανόμοιες στήλες
        if (count($headers) === count($data)) {
            // Δημιουργία συσχετιστικού πίνακα
            $project = array_combine($headers, $data);
            // Έλεγχος για το αν υπάρχουν τα απαραίτητα πεδία και δεν είναι null
            if (empty($project['Year']) || empty($project['Author']) || empty($project['Title']) ||
                empty($project['Link'])) {
                // Σε περίπτωση αποτυχίας παραλείπει την τρέχουσα εγγραφή
                continue;
            }

            // Ομαδοποιεί τις εργασίες ανά έτος
            // Παίρνει την τιμή του έτους από τη τρέχον εργασία και την αποθηκεύει την στην
            // μεταβλητή "$year"
            $year = $project['Year'];
            // Ελέγχεται αν το έτος δεν υπάρχει ήδη ως κλειδί στον πίνακα
            if (!isset($completedProjects[$year])) {
                // Αν όντως δεν υπάρχει δημιουργείται ένας νέος κενός πίνακας για αυτό το έτος
                $completedProjects[$year] = [];
            }
        }
    }
}

```

```

        // Προστίθεται ολόκληρος ο συσχετιστικός πίνακας που αντιπροσωπεύει μια εργασία
        στον πίνακα του αντίστοιχου έτους .
        $completedProjects[$year][] = $project;

    }
    else {
        // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
        error_log("Skipping row with mismatched columns: " . implode(',', $data));
    }
}
// Κλείσε το φάκελο
fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo "Error: Unable to open the local CSV file.";
    exit;
}
?>
<!-- Περιεχόμενο HTML -->
<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της
σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Δημιουργεί την κλάση old-project και εντός της χρησιμοποιεί γραμματοσειρά 18px -->
        <div class="old-project" style="font-size: 18px;">
            <!-- Εμφανίζει επικεφαλίδα -->
            <h2>Completed Student Projects</h2>

            <?php
            // Ταξινομεί τα χρόνια σε αύξουσα σειρά(Παλαιότερο προς νεότερο)
            ksort($completedProjects);
            // Βρόχος επανάληψης που για κάθε χρονιά(κλειδί) παίρνει τον πίνακα με τις εργασίες
            εκείνης της χρονιάς(τιμή)
            foreach ($completedProjects as $year => $projects) {
                // Εμφανίζει το έτος με έντονα γράμματα(strong), αλλάζει γραμμή(<br>) και ανοίγει
                λίστα με κουκκίδες για τις εργασίες αυτού του έτους(<ul>)
                echo "<strong>$year</strong><br><ul>";
                // Βρόχος επανάληψης για κάθε εργασία στο τρέχον έτος
                foreach ($projects as $project) {
                    // Αποθηκεύει τον τίτλο της εργασίας στη μεταβλητή $title
                    $title = htmlspecialchars($project['Title']);
                    // Αποθηκεύει τον συγγραφέα της εργασίας στη μεταβλητή $author
                    $author = htmlspecialchars($project['Author']);

```

```

// Αποθηκεύει τον σύνδεσμο της εργασίας στη μεταβλητή $link
$link = htmlspecialchars($project['Link']);
// Δημιουργεί αντικείμενο της λίστας με κουκκίδες(<li>), δημιουργεί
σύνδεσμο(<a>), που πηγαίνει στην διεύθυνση $link, ανοίγει σε νέα καρτέλα(target="_blank")
και εμφανίζει ως κείμενο τις τιμές των μεταβλητών $title, $author, $link
echo "<li><a href=\"\$link\" target=\"_blank\"> $year, $author, $title
</a></li>";
}
// Κλείνει την λίστα με κουκκίδες μετά από όλες τις εργασίες του έτους
echo "</ul>";
}
?>
<!-- Κλείνει τις ανοιχτές διαιρέσεις του κώδικα -->
</div>
</div>
</div>

```

Σχήμα 4.18 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Completed Projects

Το αρχείο Completed Projects επεκτείνει το βασικό μοτίβο υλοποίησης, εισάγοντας λογική ομαδοποίησης και ταξινόμησης των δεδομένων του CSV στο στάδιο ανάλυσης και δόμησης δεδομένων. Σε αντίθεση με τα προηγούμενα αρχεία που παρουσιάζουν τα δεδομένα με την σειρά που εμφανίζονται στο αρχείο CSV, το αρχείο Completed Projects κατασκευάζει έναν δισδιάστατο συνθετικό πίνακα, ομαδοποιώντας κάθε ολοκληρωμένη εργασία φοιτητών με βάση το έτος ολοκλήρωσης νέος. Αυτό επιτυγχάνεται, παίρνοντας την τιμή του έτους από τη τρέχον εργασία και αποθηκεύοντας την στην μεταβλητή “\$year” (\$year = \$project[‘Year’]). Κατόπιν ελέγχεται αν το έτος δεν υπάρχει ήδη ως κλειδί στον πίνακα (if (!isset(\$completedProjects[\$year]))) και αν όντως δεν υπάρχει δημιουργείται νέος νέος κενός πίνακας για αυτό το έτος (\$completedProjects[\$year] = []), για παράδειγμα αν το 2022 δεν υπάρχει, δημιουργεί το “\$completedProjects[‘2022’] = []”. Στο τέλος νέος διαδικασίας ομαδοποίησης των εργασιών προστίθεται ολόκληρος ο συσχετιστικός πίνακας που αντιπροσωπεύει μια εργασία στον πίνακα του αντίστοιχου έτους.

Για παράδειγμα αν το CSV περιείχε:

Year	Author	Title	Link
2023	Γιώργος	Project A	Link1
2023		Project B	Link2
2022	Γιάννης	Project C	Link3
2022	Δήμητρα	Project D	Link4

Τότε το αποτέλεσμα νέος επεξεργασίας θα ήταν:

```
$completedProjects = [  
    '2023' => [  
        ['Year' => '2023', 'Author' => 'Γιώργος', 'Title' => 'Project A', 'Link' => 'Link1']  
    ],  
    '2022' => [  
        ['Year' => '2022', 'Author' => 'Γιάννης', 'Title' => 'Project C', 'Link' => 'Link3']  
        ['Year' => '2022', 'Author' => 'Δήμητρα', 'Title' => 'Project D', 'Link' => 'Link4']  
    ]  
];
```

Δημιουργούνται 2 ομάδες για νέος εργασίες 1 ομάδα για όσες έγιναν το 2023 και 1 ομάδα για όσες έγιναν το 2022 όπου τοποθετούνται οι ανάλογες εργασίες πλην νέος εργασίας στη δεύτερη γραμμή λόγω του κενού πεδίου “Author”.

```
// Ομαδοποιεί τις εργασίες ανά έτος  
// Παίρνει την τιμή του έτους από τη τρέχον εργασία και την αποθηκεύει στην  
μεταβλητή “$year”  
$year = $project['Year'];  
// Ελέγχεται αν το έτος δεν υπάρχει ήδη ως κλειδί στον πίνακα  
if (!isset($completedProjects[$year])) {  
    // Αν όντως δεν υπάρχει δημιουργείται νέος νέος κενός πίνακας για αυτό το έτος  
    $completedProjects[$year] = [];  
}  
// Προστίθεται ολόκληρος ο συσχετιστικός πίνακας που αντιπροσωπεύει μια εργασία  
στον πίνακα του αντίστοιχου έτους .  
$completedProjects[$year][] = $project;
```

Σχήμα 4.19 Το τμήμα του κώδικα, το οποίο είναι υπεύθυνο για την ομαδοποίηση των εργασιών ανά έτος

Με αυτή την προσέγγιση το απλό σύνολο δεδομένων που προσφέρει το CSV μετατρέπεται σε μία δομημένη ιεραρχία βάση των χρονολογιών. Στο στάδιο της απόδοσης ο πίνακας “\$completedProjects” ταξινομείται με βάση τα κλειδιά του(έτη) σε αύξουσα σειρά, ώστε οι εργασίες να εμφανίζονται από την παλαιότερη προς τη νεότερη(*ksort(\$completedProjects)*). Επιπρόσθετα λόγω αυτής της νέας δομής η απόδοση προσαρμόζεται οργανικά σε αυτή με τη χρήση δύο βρόγχων επανάληψης εκ των οποίων ο ένας είναι εμφωλευμένος μέσα στον άλλον. Ο πρώτος(εξωτερικός) βρόχος(*foreach (\$completedProjects as \$year => \$projects)*) επαναλαμβάνεται για κάθε χρονιά μέσα στον πίνακα “\$completedProjects” ενώ ο

δεύτερος(*foreach (\$projects as \$project)*)(εσωτερικός) επαναλαμβάνεται για κάθε εργασία μέσα στο τρέχον έτος.

```
// Ταξινομεί τα χρόνια σε αύξουσα σειρά(Παλαιότερο προς νεότερο)
ksort($completedProjects);
// Βρόχος επανάληψης που για κάθε χρονιά(κλειδί) παίρνει τον πίνακα με τις εργασίες
εκείνης της χρονιάς(τιμή)
foreach ($completedProjects as $year => $projects) {
    // Εμφανίζει το έτος με έντονα γράμματα(strong), αλλάζει γραμμή(<br>) και ανοίγει
λίστα με κουκκίδες για τις εργασίες αυτού του έτους(<ul>)
    echo "<strong>$year</strong><br><ul>";
    // Βρόχος επανάληψης για κάθε εργασία στο τρέχον έτος
    foreach ($projects as $project) {
        // Αποθηκεύει τον τίτλο της εργασίας στη μεταβλητή $title
        $title = htmlspecialchars($project['Title']);
        // Αποθηκεύει τον συγγραφέα της εργασίας στη μεταβλητή $author
        $author = htmlspecialchars($project['Author']);
        // Αποθηκεύει τον σύνδεσμο της εργασίας στη μεταβλητή $link
        $link = htmlspecialchars($project['Link']);
        // Δημιουργεί αντικείμενο της λίστας με κουκκίδες(<li>),δημιουργεί
σύνδεσμο(<a>), που πηγαίνει στην διεύθυνση $link, ανοίγει σε νέα καρτέλα(target="_blank")
και εμφανίζει ως κείμενο τις τιμές των μεταβλητών $title,$author,$link
        echo "<li><a href=\"\$link\" target=\"_blank\"> $year, $author, $title
</a></li>";
    }
    // Κλείνει την λίστα με κουκκίδες μετά από όλες τις εργασίες του έτους
    echo "</ul>";
}
```

Σχήμα 4.20 Το τμήμα του κώδικα από το στάδιο της απόδοσης, στο οποίο γίνεται η ταξινόμηση και η χρήση των 2 βρόχων επανάληψης

4.3.5.2 Announcements

Η ενότητα Announcements βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/announcements.html> και περιέχει ανακοινώσεις, το CSV που του τροφοδοτεί δεδομένα χωρίζεται σε 5 πεδία: Date and Time, Title, Announcement, Image, Links

Ακολουθεί ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας:

```
<?php
// Σύνδεσμος Dropbox για το CSV
$csvUrl =
'https://www.dropbox.com/scl/fi/5bs5056hspxo3r42sgrgb/announcements.csv?rlkey=ltqx9edb3
to5pcr9y30gy5d9&dl=1';
```

```

//Αρχικοποιεί την cURL συνεδρία
$ch = curl_init();
//Ορίζει τις επιλογές για την cURL συνεδρία
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
curl_setopt($ch, CURLOPT_URL, $csvUrl);
// Επιστροφή δεδομένων ως συμβολοσειρές
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// Ακολουθεί HTTP ανακατευθύνσεις
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
$data = curl_exec($ch);

//Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Μήνυμα σφάλματος
    echo 'Error: Unable to fetch the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

//Κλείνει την cURL συνεδρία
curl_close($ch);

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'announcements.csv';

// Αποθήκευσε τα downloaded data στον τοπικό φάκελο
if (file_put_contents($localCsvFile, $data) === false) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Failed to save the downloaded CSV to local storage. ');
    // Μήνυμα σφάλματος
    echo 'Error: Unable to save the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

//Ανάγνωση και επεξεργασία του CSV
// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    //Διάβασε τη γραμμή των επικεφαλίδων
    $headers = fgetcsv($handle, 1000, ',');
    // Αρχικοποίησε έναν πίνακα που να κρατάει τα δεδομένα των ανακοινώσεων

```

```

$announcements = [];
// Διάβασε κάθε σειρά του φακέλου CSV
while (($row = fgetcsv($handle, 1000, ',')) !== false) {
    // Παρέλειψε σειρές με ανόμοιες στήλες
    if (count($headers) === count($row)) {
        // Δημιουργείται συσχετιστικός πίνακας
        $announcement = array_combine($headers, $row);
        // Επεξεργασία των συνδέσμων
        // Ελέγχει αν το πεδίο Links υπάρχει στο CSV και δεν είναι κενό
        if (!empty($announcement['Links'])) {
            // Αν δεν είναι κενό χωρίζει το κείμενο σε ξεχωριστά στοιχεία με διαχωριστικό το
            // σύμβολο ";"
            $links = explode(';', $announcement['Links']);
            // Δημιουργείται ένας κενός πίνακας όπου θα αποθηκευτούν οι επεξεργασμένοι
            // σύνδεσμοι
            $processedLinks = [];
            // Βρόχος επανάληψης για να επεξεργαστεί κάθε σύνδεσμος ξεχωριστά
            foreach ($links as $link) {
                // Ελέγχει αν ο σύνδεσμος περιέχει τον χαρακτήρα "|"
                if (strpos($link, '|') !== false) {
                    // Χωρίζει τον σύνδεσμο σε 2 μέρη: Το πρώτο αποθηκεύεται στη μεταβλητή $text
                    // και το δεύτερο στην μεταβλητή $url
                    list($text, $url) = explode('|', $link, 2);
                    // Προσθέτει τον επεξεργασμένο σύνδεσμο στον πίνακα
                    $processedLinks[] = [
                        // Αφαιρεί τα κενά από την αρχή και το τέλος της συμβολοσειράς $text
                        'text' => trim($text),
                        // Αφαιρεί τα κενά από την αρχή και το τέλος της συμβολοσειράς $url
                        'url' => trim($url)
                    ];
                }
            }
            // Μετά το τέλος του βρόγχου αποθηκεύει τον πίνακα με τους επεξεργασμένους
            // συνδέσμους στο πεδίο ProcessedLinks του πίνακα $announcement
            $announcement['ProcessedLinks'] = $processedLinks;
        }
        // Προσθέτει τον πίνακα $announcement (που πλέον περιέχει τους επεξεργασμένους
        // συνδέσμους) στον κύριο πίνακα $announcements
        $announcements[] = $announcement;
    }
}
// Κλείνει τον φάκελο
fclose($handle);
} else {
    // Εμφανίζει μήνυμα σφάλματος

```

```

    echo "Error: Unable to open the local CSV file.";
    // Τερματίζει την διαδικασία
    exit;
}

// Ταξινομεί τις ανακοινώσεις βάζοντας τις νεότερες πρώτες
// Ταξινομεί τον πίνακα $announcements με βάση την συνάρτηση function($a, $b)
usort($announcements, function($a, $b) {
    // Η συνάρτηση strtotime μετατρέπει τις ημερομηνίες σε σημάνσεις χρόνου για σύγκριση όπου
    // ο τελεστής <=> επιστρέφει τις τιμές -1,0,1 και η σειρά $b πριν από $a εξασφαλίζει φθίνουσα
    // ταξινόμηση
    return strtotime($b['Date and Time']) <=> strtotime($a['Date and Time']);
});
?>

<!-- HTML περιεχόμενο -->
<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της
σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Δημιουργεί την κλάση announcements που έχει μέγεθος γραμματοσειράς 18px -->
        <div class="announcements" style="font-size: 18px;">
            <!-- Ορίζει την επικεφαλίδα -->
            <h2>Announcements</h2>
            <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε ανακοίνωση στον πίνακα
$announcements -->
            <?php foreach ($announcements as $announcement): ?>
                <!-- Δημιουργεί ένα "δοχείο" για κάθε ανακοίνωση με κάθετη απόσταση 40px από την
επόμενη. -->
                <div class="announcement-item" style="margin-bottom: 40px;">
                    <!-- Δημιουργεί την κλάση announcement-content και ορίζει το στυλ της -->
                    <div class="announcement-content" style="display: flex; align-items: flex-start;
gap: 30px; padding: 20px; border: 1px solid #ddd; border-radius: 10px; box-shadow: 2px 2px
10px rgba(0,0,0,0.1); background: #fff;">
                        <!-- Ελέγχει αν υπάρχει το πεδίο Image -->
                        <?php if (!empty($announcement['Image'])): ?>
                            <!-- Εμφανίζει την εικόνα, την οποία παίρνει από τον φάκελο images με το
όνομα που έχει μέσα το CSV, με την συνάρτηση alt προσθέτει εναλλακτικό κείμενο για λόγους
προσβασιμότητας και ορίζει το στυλ της με την συνάρτηση style -->
                            " style="width: 150px;
height: 150px; object-fit: cover; border-radius: 10px; flex-shrink: 0;">
                        <?php endif; ?>

```

```

        <!-- Δημιουργεί στοιχείο με στυλ flex 1 που θα πάρει όλο τον διαθέσιμο χώρο
που απομένει μετά την εμφάνιση της εικόνας -->
        <div style="flex: 1;">
            <!-- Δημιουργεί την κλάση announcement-header με απόσταση 15px από το
κείμενο που ακολουθεί -->
            <div class="announcement-header" style="margin-bottom: 15px;">
                <!-- Δημιουργεί κλάση announcement-date και ορίζει το στυλ της -->
                <span class="announcement-date" style="color: #666; font-size: 14px;">
                    <!-- Εμφανίζει την ημερομηνία -->
                    <?php echo htmlspecialchars($announcement['Date and Time']); ?>
                </span>
                <!-- Δημιουργεί κλάση announcement-title και ορίζει το στυλ της -->
                <h3 class="announcement-title" style="margin: 5px 0 10px 0; font-size:
20px; color: #333;">
                    <!-- Εμφανίζει τον τίτλο -->
                    <?php echo htmlspecialchars($announcement['Title']); ?>
                </h3>
            </div>
            <!-- Ορίζει κλάση announcement-text και ορίζει το στυλ της -->
            <div class="announcement-text" style="font-size: 16px; color: #555; line-
height: 1.5;">
                <!-- Εμφανίζει το κείμενο της ανακοίνωσης -->
                <?php echo $announcement['Announcement']; ?>
            </div>
            <!-- Ελέγχει αν υπάρχουν επεξεργασμένοι σύνδεσμοι για την τρέχουσα
ανακοίνωση -->
            <?php if (!empty($announcement['ProcessedLinks'])): ?>
                <!-- Δημιουργεί κλάση announcement-links και ορίζει το στυλ της -->
                <div class="announcement-links" style="margin-top: 15px;">
                    <!-- Εμφανίζει το κείμενο "Related Links" με έντονα γράμματα και
περιθώριο 8px από το κείμενο που ακολουθεί -->
                    <p style="font-weight: bold; margin-bottom: 8px;">Related
Links:</p>
                    <!-- Ανοίγει μια λίστα με κουκκίδες χωρίς εξωτερικό κενό και με
εσωτερικό περιθώριο 20px από τα αριστερά -->
                    <ul style="margin: 0; padding-left: 20px;">
                        <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε σύνδεσμο στο
πεδίο ProcessedLinks του πίνακα $announcement -->
                        <?php foreach ($announcement['ProcessedLinks'] as $link): ?>
                            <!-- Δημιουργεί ένα στοιχείο λίστας με περιθώριο 5px από κάτω -->
                            <li style="margin-bottom: 5px;">
                                <!-- Δημιουργεί ένα σύνδεσμο με URL από το $link['url'], που
ανοίγει σε νέα καρτέλα(target="_blank"), με μπλε χρώμα(#007bff) και χωρίς υπογράμμιση(text-
decoration: none) -->

```

```

        <a href="<?php echo htmlspecialchars($link['url']); ?>"
target="_blank" style="color: #007bff; text-decoration: none;">
        <!-- Κείμενο του συνδέσμου από το $link['text'] -->
        <?php echo htmlspecialchars($link['text']); ?>
        </a>
    </li>
    <!-- Τέλος βρόγχου -->
    <?php endforeach; ?>
    <!-- Κλείνει την λίστα με κουκκίδες -->
    </ul>
    <!-- Κλείνει τις ανοιχτές διαιρέσεις του προγράμματος -->
    </div>
    <!-- Τέλος δομής ελέγχου -->
    <?php endif; ?>
</div>
</div>
</div>
<!-- Τέλος βρόγχου -->
<?php endforeach; ?>
<!-- Ελέγχει αν δεν υπάρχουν ανακοινώσεις -->
<?php if (empty($announcements)): ?>
    <!-- Αν δεν υπάρχουν εμφανίζει αυτό το μήνυμα -->
    <p>No announcements at this time.</p>
    <!-- Τέλος δομής ελέγχου -->
    <?php endif; ?>
</div>
</div>
</div>
<!-- CSS Styling -->
<style>

.announcement-item {
    margin-bottom: 30px; /* Εφαρμόζει κάτω περιθώριο 30px μεταξύ κάθε ανακοίνωσης */
}
.announcement-content {
    transition: transform 0.3s ease; /* Προετοιμάζει το στοιχείο για το εφέ που ακολουθεί με
επιβράδυνση 0.3 δευτερόλεπτα στο τέλος της κίνησης */
}
.announcement-content:hover { /* Όταν ο χρήστης περνάει το ποντίκι πάνω από μια
ανακοίνωση */
    transform: translateY(-3px); /* Μετακινεί την ανακοίνωση επάνω κατά 3px */
    box-shadow: 2px 5px 15px rgba(0,0,0,0.1); /* Προσθέτει μια πιο έντονη σκιά */
}
.announcement-links a:hover { /* Όταν ο χρήστης περνάει το ποντίκι του πάνω από ένα
σύνδεσμο */

```

```

    text-decoration: underline; /* Προσθέτει υπογράμμιση στο κείμενο του συνδέσμου */
}
/* Τέλος CSS Styling */
</style>

```

Σχήμα 4.21 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Announcements

Το αρχείο παραγωγής περιεχομένου της ενότητας Announcements επεκτείνει και αυτό το βασικό μοτίβο υλοποίησης εισάγοντας μετασχηματισμό δεδομένων και χρονολογική ταξινόμηση στον κώδικα του. Αναλύει το πεδίο “Links”, μετατρέποντας το από μια απλή συμβολοσειρά σε ένα συσχετιστικό πίνακα του οποίου κάθε στοιχείο είναι ένας νέος πίνακας που περιέχει ξεχωριστά το κείμενο και το URL κάθε συνδέσμου(*\$announcement['ProcessedLinks']*). Αυτό επιτυγχάνεται χωρίζοντας το κείμενο του πεδίου “Links” σε κομμάτια βάση του χαρακτήρα “;”(*\$links = explode(';', \$announcement['Links'])*) ενώ παράλληλα κατασκευάζει ένα κενό πίνακα στον οποίο θα αποθηκευτούν οι σύνδεσμοι αφότου επεξεργαστούν(*\$processedLinks = []*). Στη συνέχεια χρησιμοποιείται βρόχος επανάληψης ώστε να επεξεργαστεί κάθε σύνδεσμος ξεχωριστά. Οι σύνδεσμοι ελέγχονται για τον αν περιέχουν το σύμβολο “|” και ύστερα χωρίζονται σε δύο μέρη βάση αυτού: Κείμενο(*\$text*) και URL(*\$url*). Έπειτα τα δύο αυτά μέρη αποθηκεύονται στον πίνακα “\$processedLinks” ο οποίος με την σειρά του αποθηκεύεται στο πεδίο “ProcessedLinks” του συσχετιστικού πίνακα “\$announcement”. Στο τέλος ο συσχετιστικός πίνακας “\$announcement” προστίθεται στον κύριο πίνακα “\$announcements” και κλείνεται το αρχείο CSV.

Για παράδειγμα αν το CSV περιείχε:

Date and Time	Title	Announcement	Image	Links
01/01/2025 10:00	Title A	Example Announcement	announcementlogo.jpg	Example https://example.com
15/10/2025 18:00	Example B	Example B announcement	announcelogo.jpg	Program https://example.com/program

Τότε το αποτέλεσμα της επεξεργασίας θα ήταν:

```

$announcements = [
    [
        'Date and Time' => '01/09/2025 10:00',
        'Title' => 'Workshop Registration',
        'Announcement' => 'Example Announcement.',
        'Image' => 'announcelogo.jpg',
        'Links' => 'Example|https://example.com' => [
            [

```

```

        'text' => 'Example',
        'url' => 'https://example.com'
    ]
],
[
    'Date and Time' => '15/10/2025 18:00',
    'Title' => 'Example B',
    'Announcement' => 'Example B announcement',
    'Image' => 'announcelogo.jpg',
    'Links' => 'Program|https://example.com/program' => [
        [
            'text' => 'Program',
            'url' => 'https://example.com/program'
        ]
    ]
]
]

```

Δημιουργείται πίνακας που περιέχει τους συνδέσμους μέσα στον κύριο πίνακα “\$announcements”.

```

// Επεξεργασία των συνδέσμων
// Ελέγχει αν το πεδίο Links υπάρχει στο CSV και δεν είναι κενό
if (!empty($announcement['Links'])) {
    // Αν δεν είναι κενό χωρίζει το κείμενο σε ξεχωριστά στοιχεία με διαχωριστικό το
    σύμβολο ";"
    $links = explode(';', $announcement['Links']);
    // Δημιουργείται ένας κενός πίνακας όπου θα αποθηκευτούν οι επεξεργασμένοι
    σύνδεσμοι
    $processedLinks = [];
    // Βρόχος επανάληψης για να επεξεργαστεί κάθε σύνδεσμος ξεχωριστά
    foreach ($links as $link) {
        // Ελέγχει αν ο σύνδεσμος περιέχει τον χαρακτήρα "|"
        if (strpos($link, '|') !== false) {
            // Χωρίζει τον σύνδεσμο σε 2 μέρη: Το πρώτο αποθηκεύεται στη μεταβλητή $text
            και το δεύτερο στην μεταβλητή $url
            list($text, $url) = explode('|', $link, 2);
            // Προσθέτει τον επεξεργασμένο σύνδεσμο στον πίνακα
            $processedLinks[] = [
                // Αφαιρεί τα κενά από την αρχή και το τέλος της συμβολοσειράς $text
                'text' => trim($text),
                // Αφαιρεί τα κενά από την αρχή και το τέλος της συμβολοσειράς $url
                'url' => trim($url)
            ]
        }
    }
}

```

```

    }
    }
}
// Μετά το τέλος του βρόγχου αποθηκεύει τον πίνακα με τους επεξεργασμένους
// συνδέσμους στο πεδίο ProcessedLinks του πίνακα $announcement
$announcement['ProcessedLinks'] = $processedLinks;
}
// Προσθέτει τον πίνακα $announcement(που πλέον περιέχει τους επεξεργασμένους
// συνδέσμους) στον κύριο πίνακα $announcements
$announcements[] = $announcement;
}
}

```

Σχήμα 4.22 Το τμήμα του κώδικα υπεύθυνο για την επεξεργασία των συνδέσμων

Πριν το στάδιο απόδοσης της ενότητας Announcements, όλες οι ανακοινώσεις ταξινομούνται χρονολογικά σε φθίνουσα σειρά με βάση το πεδίο “Date and Time” του CSV ώστε η πιο πρόσφατη να είναι πάντα στην κορυφή. Χρησιμοποιείται η συνάρτηση “usort()” που ταξινομεί έναν πίνακα με βάση μια συνάρτηση σύγκρισης που ορίζει ο χρήστης. Στην προκειμένη περίπτωση, ο πίνακας “\$announcements” είναι ο πίνακας και η “function(\$a, \$b)” είναι η συνάρτηση που ορίζει πως θα γίνει η σύγκριση(usort(\$announcements, function(\$a, \$b))). Στη συνέχεια αξιοποιείται η συνάρτηση “strtotime()” που μετατρέπει μια συμβολοσειρά ημερομηνίας σε χρονική σήμανση Unix(αριθμός δευτερολέπτων από τις 1/1/1970) και ο ειδικός τελεστής σύγκρισης “<=>”, που επιστρέφει -1 αν το αριστερό μέρος του τελεστή είναι μικρότερο από το δεξί, 0 αν και τα δύο μέρη είναι ίσα και 1 αν το αριστερό μέρος είναι μεγαλύτερο του δεξιού. Έτσι όλες οι συμβολοσειρές στον πίνακα “\$announcements” συγκρίνονται μεταξύ τους ανά ζεύγη και ταξινομούνται καταλλήλως.

```

// Ταξινομεί τις ανακοινώσεις βάζοντας τις νεότερες πρώτες
// Ταξινομεί τον πίνακα $announcements με βάση την συνάρτηση function($a, $b)
usort($announcements, function($a, $b) {
    // Η συνάρτηση strtotime μετατρέπει τις ημερομηνίες σε σημάνσεις χρόνου για σύγκριση όπου
    // ο τελεστής <=> επιστρέφει τις τιμές -1,0,1 και η σειρά $b πριν από $a εξασφαλίζει φθίνουσα
    // ταξινόμηση
    return strtotime($b['Date and Time']) <=> strtotime($a['Date and Time']);
});

```

Σχήμα 4.23 Το τμήμα του κώδικα υπεύθυνο για την ταξινόμηση των ανακοινώσεων

Τέλος στο στάδιο της απόδοσης γίνεται χρήση δύο βρόγχων επανάληψης εκ των οποίων ο ένας είναι εμφωλευμένος μέσα στον άλλον. Ο πρώτος επαναλαμβάνεται για κάθε ανακοίνωση στον πίνακα “\$announcements” (foreach (\$announcements as \$announcement)) ενώ ο δεύτερος επαναλαμβάνεται για κάθε σύνδεσμο στο πεδίο “ProcessedLinks” του πίνακα “\$announcement”(foreach (\$announcement['ProcessedLinks'] as \$link)).

```

<?php foreach ($announcements as $announcement): ?>

```

```

        <!-- Δημιουργεί ένα "δοχείο" για κάθε ανακοίνωση με κάθετη απόσταση 40px από
την επόμενη. -->
        <div class="announcement-item" style="margin-bottom: 40px;">
            <!-- Δημιουργεί την κλάση announcement-content και ορίζει το στυλ της -->
            <div class="announcement-content" style="display: flex; align-items: flex-start;
gap: 30px; padding: 20px; border: 1px solid #ddd; border-radius: 10px; box-shadow: 2px 2px
10px rgba(0,0,0,0.1); background: #fff;">
                <!-- Ελέγχει αν υπάρχει το πεδίο Image -->
                <?php if (!empty($announcement['Image'])): ?>
                    <!-- Εμφανίζει την εικόνα, την οποία παίρνει από τον φάκελο images με το
όνομα που έχει μέσα το CSV, με την συνάρτηση alt προσθέτει εναλλακτικό κείμενο για λόγους
προσβασιμότητας και ορίζει το στυλ της με την συνάρτηση style -->
                    " style="width: 150px;
height: 150px; object-fit: cover; border-radius: 10px; flex-shrink: 0;">
                    <?php endif; ?>
                    <!-- Δημιουργεί στοιχείο με στυλ flex 1 που θα πάρει όλο τον διαθέσιμο χώρο
που απομένει μετά την εμφάνιση της εικόνας -->
                    <div style="flex: 1;">
                        <!-- Δημιουργεί την κλάση announcement-header με απόσταση 15px από το
κείμενο που ακολουθεί -->
                        <div class="announcement-header" style="margin-bottom: 15px;">
                            <!-- Δημιουργεί κλάση announcement-date και ορίζει το στυλ της -->
                            <span class="announcement-date" style="color: #666; font-size: 14px;">
                                <!-- Εμφανίζει την ημερομηνία -->
                                <?php echo htmlspecialchars($announcement['Date and Time']); ?>
                            </span>
                            <!-- Δημιουργεί κλάση announcement-title και ορίζει το στυλ της -->
                            <h3 class="announcement-title" style="margin: 5px 0 10px 0; font-size:
20px; color: #333;">
                                <!-- Εμφανίζει τον τίτλο -->
                                <?php echo htmlspecialchars($announcement['Title']); ?>
                            </h3>
                        </div>
                        <!-- Ορίζει κλάση announcement-text και ορίζει το στυλ της -->
                        <div class="announcement-text" style="font-size: 16px; color: #555; line-
height: 1.5;">
                            <!-- Εμφανίζει το κείμενο της ανακοίνωσης -->
                            <?php echo $announcement['Announcement']; ?>
                        </div>
                        <!-- Ελέγχει αν υπάρχουν επεξεργασμένοι σύνδεσμοι για την τρέχουσα
ανακοίνωση -->
                        <?php if (!empty($announcement['ProcessedLinks'])): ?>
                            <!-- Δημιουργεί κλάση announcement-links και ορίζει το στυλ της -->
                            <div class="announcement-links" style="margin-top: 15px;">

```

```

        <!-- Εμφανίζει το κείμενο "Related Links" με έντονα γράμματα και
        περιθώριο 8px από το κείμενο που ακολουθεί -->
        <p style="font-weight: bold; margin-bottom: 8px;">Related
Links:</p>

        <!-- Ανοίγει μια λίστα με κουκκίδες χωρίς εξωτερικό κενό και με
        εσωτερικό περιθώριο 20px από τα αριστερά -->
        <ul style="margin: 0; padding-left: 20px;">
            <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε σύνδεσμο στο
            πεδίο ProcessedLinks του πίνακα $announcement -->
            <?php foreach ($announcement['ProcessedLinks'] as $link): ?>
                <!-- Δημιουργεί ένα στοιχείο λίστας με περιθώριο 5px από κάτω -->
                <li style="margin-bottom: 5px;">
                    <!-- Δημιουργεί ένα σύνδεσμο με URL από το $link['url'], που
                    ανοίγει σε νέα καρτέλα(target="_blank"), με μπλε χρώμα(#007bff) και χωρίς υπογράμμιση(text-
                    decoration: none) -->
                    <a href="<?php echo htmlspecialchars($link['url']); ?>"
target="_blank" style="color: #007bff; text-decoration: none;">
                        <!-- Κείμενο του συνδέσμου από το $link['text'] -->
                        <?php echo htmlspecialchars($link['text']); ?>
                    </a>
                </li>
            <!-- Τέλος βρόγχου -->
            <?php endforeach; ?>
        <!-- Κλείνει την λίστα με κουκκίδες -->
        </ul>
    <!-- Κλείνει τις ανοιχτές διαιρέσεις του προγράμματος -->
    </div>
<!-- Τέλος δομής ελέγχου -->
<?php endif; ?>
</div>
</div>
</div>
<!-- Τέλος βρόγχου -->
<?php endforeach; ?>

```

Σχήμα 4.24 Το τμήμα του κώδικα από το στάδιο της απόδοσης στάδιο της απόδοσης, στο οποίο γίνεται χρήση δύο βρόγχων επανάληψης εκ των οποίων ο ένας είναι εμφωλευμένος μέσα στον άλλον

4.3.5.3 Home

Η ενότητα Home βρίσκεται αναρτημένη υπό τη διεΨ:

<https://microengineering.iem.ihu.gr/kafantaris/static/index.html> και αφορά την αρχική σελίδα

την οποία επισκέπτεται ο χρήστης με την είσοδο του στον ιστότοπο, το CSV που της τροφοδοτεί δεδομένα αποτελείται από 3 πεδία: General Info, Courses, Other Links.

Ακολουθεί ο πλήρης κώδικας του:

```
<?php
// Σύνδεσμος Dropbox για το CSV
$csvUrl =
"https://www.dropbox.com/scl/fi/tu0mlpg7qfw705ztk4vvt/home.csv?rlkey=pg94bx5z4g5ggg3u
zs3kijnqrr&dl=1";

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'home.csv';

// Αρχικοποιεί την cURL συνεδρία
$ch = curl_init();
// Ορίζει τις επιλογές για την cURL συνεδρία
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
curl_setopt($ch, CURLOPT_URL, $csvUrl);
// Επιστροφή δεδομένων ως συμβολοσειρές
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// Ακολουθεί HTTP ανακατευθύνσεις
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
$data = curl_exec($ch);

// Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Μήνυμα σφάλματος
    echo 'Error: Unable to fetch the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}
// Κλείνει την cURL συνεδρία
curl_close($ch);

// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
$localCsvFile = 'home.csv';

// Αποθήκευσε τα κατεβασμένα δεδομένα στον τοπικό φάκελο
if (file_put_contents($localCsvFile, $data) === false) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Failed to save the downloaded CSV to local storage.');
```

```

// Μήνυμα σφάλματος
echo 'Error: Unable to save the CSV file.';
// Τερματισμός διαδικασίας
exit;
}

// Μετατροπή των δεδομένων του CSV σε πίνακα
// Χωρίζει την συμβολοσειρά $data σε πίνακα γραμμών χρησιμοποιώντας το χαρακτήρα "\n" ως
διαχωριστικό
$rows = explode("\n", trim($data));
// Αρχικοποιεί έναν κενό πίνακα που θα κρατάει τα επεξεργασμένα δεδομένα
$csvdata = [];
// Βρόχος επανάληψης που επαναλαμβάνεται για κάθε γραμμή του πίνακα $rows
foreach ($rows as $row) {
    // Αναλύει μια γραμμή του CSV και την επιστρέφει σαν πίνακα με πεδία
    $fields = str_getcsv($row);
    // Ελέγχει αν η γραμμή περιέχει τουλάχιστον 2 πεδία(κλειδί και τιμή)
    if (count($fields) >= 2) {
        // Εξάγεται το πρώτο πεδίο της γραμμής και αποθηκεύεται στην μεταβλητή $key
        $key = trim($fields[0]);
        // Εξάγεται το δεύτερο πεδίο της γραμμής και αποθηκεύεται στην μεταβλητή $value
        $value = trim($fields[1]);
        // Ελέγχει αν το κλειδί είναι "General Info"
        if ($key === "General Info") {
            // Αν το κλειδί είναι "General Info" χωρίζει την τιμή σε υποτιμές με βάση το ελληνικό
            ερωτηματικό ";";
            $csvdata[$key] = array_map('trim', explode(";", $value));
            // Αν το κλειδί είναι "Courses" ή "Other Links"
        } elseif ($key === "Courses" || $key === "Other Links") {
            // Χωρίζει την τιμή σε υποτιμές με βάση το ελληνικό ερωτηματικό ";";
            $csvdata[$key] = array_map('trim', explode(";", $value));
        }
    }
}

// Εξάγει το πεδίο "General Info" από το πίνακα $csvdata και το αποθηκεύει στη μεταβλητή
$generalInfo
$generalInfo = $csvdata["General Info"] ?? [];

// Ελέγχει αν το κλειδί "Courses" υπάρχει στο πίνακα $csvdata και αν υπάρχει εκτελεί την
array_map() για επεξεργασία
$courses = isset($csvdata["Courses"]) ? array_map(function($entry) {
    // Επεξεργασία κάθε εισόδου, χωρίζοντας την συμβολοσειρά $entry σε 2
    κομμάτια χρησιμοποιώντας το χαρακτήρα "|"
    $parts = explode("|", $entry, 2);

```

```

    // Αν ο χωρισμός ήταν επιτυχής επιστρέφει [URL, Course Name] ενώ αν αποτύχει εμφανίζει
    μήνυμα σφάλματος
    return count($parts) === 2 ? [trim($parts[1]), trim($parts[0])] : ["#", "Invalid Course
    Format: " . htmlspecialchars($entry)];
}, $csvdata["Courses"]) : [];

```

```

// Ελέγχει αν το κλειδί "Other Links" υπάρχει στο πίνακα $csvdata και αν υπάρχει εκτελεί την
array_map() για επεξεργασία
$otherLinks = isset($csvdata["Other Links"]) ? array_map(function($entry) {
    // Επεξεργασία κάθε εισόδου, χωρίζοντας την συμβολοσειρά $entry σε 2 κομμάτια
    χρησιμοποιώντας το χαρακτήρα "|"
    $parts = explode("|", $entry, 2);
    // Αν ο χωρισμός ήταν επιτυχής επιστρέφει [URL, Course Name] ενώ αν αποτύχει εμφανίζει
    μήνυμα σφάλματος
    return count($parts) === 2 ? [trim($parts[1]), trim($parts[0])] : ["#", "Invalid Link
    Format: " . htmlspecialchars($entry)];
}, $csvdata["Other Links"]) : [];
?>

```

```

<!-- HTML περιεχόμενο -->
<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της
    σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Δημιουργεί την κύρια στήλη που καταλαμβάνει τα 8/12 της οθόνης, έχει μέγεθος
        γραμματοσειράς 20px και έχει ύψος το 100% του viewport -->
        <div class="col-md-8" style="font-size: 20px; height: 100vh;">
            <!-- Εμφανίζει την εικόνα χρησιμοποιώντας την κλάση img-responsive για
            προσαρμόζεται στο μέγεθος της οθόνης και προσφέρει εναλλακτικό κείμενο για λόγους
            προσβασιμότητας χάρις την συνάρτηση alt -->
            <br>
            <!-- Εμφανίζει την επικεφαλίδα -->
            <h2>Courses:</h2>
            <!-- Δημιουργεί κλάση lessons με μέγεθος γραμματοσειράς 20px-->
            <div class="lessons" style="font-size: 20px;">
                <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε μάθημα
                στον πίνακα $courses -->
                <?php foreach ($courses as $course): ?>
                    <!-- Δημιουργεί σύνδεσμο, που περιέχει το URL του μαθήματος($course[0]), το
                    όνομα του μαθήματος ($course[1]) και ανοίγει τον σύνδεσμο σε νέα καρτέλα(target="_blank") -
                    ->
                    <a href="<? = htmlspecialchars($course[0]) ?>" target="_blank"><? =
                    htmlspecialchars($course[1]) ?></a>

```

```

        <!-- Αλλαγή γραμμής μετά από κάθε σύνδεσμο -->
        <br>
        <!-- Τέλος βρόγχου επανάληψης -->
        <?php endforeach; ?>
    </div>
    <!-- Εμφανίζει κεφαλίδα -->
    <h2>Other Links:</h2>
    <div>
        <!-- Βρόχος επανάληψης για κάθε σύνδεσμο στον
πίνακα $otherLinks -->
        <?php foreach ($otherLinks as $link): ?>
            <!-- Δημιουργεί σύνδεσμο, που περιέχει το URL($link[0]), το κείμενο του
συνδέσμου (link[1]) και ανοίγει τον σύνδεσμο σε νέα καρτέλα(target="_blank")-->
            <p><a href="<? = htmlspecialchars($link[0]) ?>" target="<? =
htmlspecialchars($link[1]) ?>"></a></p>
            <!-- Τέλος βρόγχου επανάληψης -->
            <?php endforeach; ?>
        <!-- Κλείνει τις ανοιχτές διαιρέσεις του κώδικα -->
        </div>
    </div>

    <!-- Πλάγια μπάρα με το πεδίο General Info που πιάνει τα 4/12 της οθόνης με μέγεθος
γραμματοσειράς 18px -->
    <div class="col-md-4" style="font-size: 18px;">
        <!-- Δημιουργεί επικεφαλίδα με τίτλο General Info και μέγεθος γραμματοσειράς 24px -->
        <div style="font-size: 24px;"><b>General Info</b></div><br>
        <!-- Παίρνει όλα τα στοιχεία του πίνακα $generalInfo, τα ενώνει σε μια ενιαία συμβολοσειρά
και χρησιμοποιεί την ετικέτα αλλαγής γραμμής <br> ως διαχωριστικό μεταξύ των στοιχείων
<p><?php echo implode("<br>", $generalInfo); ?></p>
        <!-- Κλείνει τις ανοιχτές διαιρέσεις του κώδικα -->
        </div>
    </div>
</div>

```

Σχήμα 4.25 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Home

Σε αντίθεση με τις ενότητες Completed Projects και Announcements, που επεκτείνουν τη βασική λογική μετασχηματίζοντας και ομαδοποιώντας τα δεδομένα, το αρχείο παραγωγής περιεχομένου της ενότητας Home αντλεί δεδομένα από ένα μη παραδοσιακό, ημιδομημένο αρχείο CSV που λειτουργεί ως ένα είδος αρχείου ρυθμίσεων απαιτώντας ένα εντελώς διαφορετικό μοντέλο ανάλυσης και επεξεργασίας δεδομένων. Ενώ τα υπόλοιπα αρχεία του συστήματος αναμένουν αρχεία CSV σε μορφή πίνακα, το αρχείο CSV της ενότητας Home οργανώνει τις πληροφορίες ως μια σειρά από ζεύγη κλειδιών-τιμών. Αυτή η θεμελιώδης διαφορά στην πηγή των δεδομένων καθιστά αναγκαία μια διαφορετική προσέγγιση στην ανάγνωση τους. Στο στάδιο ανάλυσης και δόμησης δεδομένων ο κώδικας παρακάμπτει εντελώς τις συναρτήσεις “fgetcsv()” και αντ’ αυτού

χρησιμοποιεί ένα συνδυασμό των συναρτήσεων “explode()” και “str_getcsv()” για να αποδομήσει το περιεχόμενο και να το δομήσει σε έναν συσχετιστικό πίνακα, όπου τα κλειδιά αντιστοιχούν στις τιμές τους. Χρησιμοποιεί την “\$rows = explode("\n", trim(\$data))” για να χωρίσει την συμβολοσειρά “\$data” σε πίνακα γραμμών χρησιμοποιώντας το χαρακτήρα “\n” ως διαχωριστικό. Στη συνέχεια κάθε γραμμή του πίνακα \$rows(foreach (\$rows as \$row)) αναλύεται και επιστρέφεται σαν πίνακας με πεδία(\$fields = str_getcsv(\$row)). Έπειτα αφότου ελεγχθεί ότι υπάρχουν τα δύο πεδία στη γραμμή, το πρώτο πεδίο εξάγεται και αποθηκεύεται στη μεταβλητή “\$key” και το δεύτερο εξάγεται και αποθηκεύεται στη μεταβλητή “\$value”. Κατόπιν το σύστημα εφαρμόζει εξατομικευμένη επεξεργασία(field-specific processing), καθώς κάθε κλειδί ακολουθεί το δικό του μοναδικό μοντέλο δεδομένων που πρέπει να αποκωδικοποιηθεί. Αν το κλειδί είναι General Info(if (\$key === "General Info")), τότε χωρίζει την συμβολοσειρά “\$value” που αντιστοιχεί στο κλειδί σε κομμάτια με βάση το ελληνικό ερωτηματικό “;”(\$csvdata[\$key] = array_map('trim', explode(";", \$value))) και ύστερα το εξάγει και το αποθηκεύει στη μεταβλητή “\$generalInfo” (\$generalInfo = \$csvdata["General Info"] ?? []). Αν το κλειδί είναι Courses ή Other Links(elseif (\$key === "Courses" || \$key === "Other Links")), τότε γίνονται δύο διαχωρισμοί της συμβολοσειράς. Ο πρώτος όπως και με το κλειδί General Info χωρίζει την συμβολοσειρά σε κομμάτια με βάση το ελληνικό ερωτηματικό “;”(\$csvdata[\$key] = array_map('trim', explode(";", \$value))), ο δεύτερος χωρίζει τα κομμάτια σε δύο κομμάτια, με βάση τον χαρακτήρα “|”, τα οποία θα αποτελέσουν το κείμενο και το URL ενός συνδέσμου(\$parts = explode("|", \$entry, 2)).

Για παράδειγμα αν στο CSV υπάρχει η εξής γραμμή: “Courses”, “[1.1]Πρώτο Μάθημα|https://firstclass.gr;[2.1]Δεύτερο Μάθημα|https://secondclass.gr”

Τότε μετά τον πρώτο διαχωρισμό ο πίνακας \$courses θα έχει αυτή τη δομή:

```
$courses = [
  0 => “[1.1]Πρώτο Μάθημα|https://firstclass.gr”,
  1 => “[2.1]Δεύτερο Μάθημα|https://secondclass.gr”
];
```

Και μετά το δεύτερο διαχωρισμό θα έχει αυτή τη δομή:

```
$courses = [
  0 => [
    0 => “https://firstclass.gr”, // URL
    1 => “[1.1]Πρώτο Μάθημα” // Τίτλος
  ],
  1 => [
    0 => “https://secondclass.gr”, // URL
    1 => “[2.1]Δεύτερο Μάθημα” // Τίτλος
  ]
];
```

```
]
];
```

```
// Μετατροπή των δεδομένων του CSV σε πίνακα
// Χωρίζει την συμβολοσειρά $data σε πίνακα γραμμών χρησιμοποιώντας το χαρακτήρα "\n" ως
διαχωριστικό
$rows = explode("\n", trim($data));
// Αρχικοποιεί έναν κενό πίνακα που θα κρατάει τα επεξεργασμένα δεδομένα
$csvdata = [];
// Βρόχος επανάληψης που επαναλαμβάνεται για κάθε γραμμή του πίνακα $rows
foreach ($rows as $row) {
    // Αναλύει μια γραμμή του CSV και την επιστρέφει σαν πίνακα με πεδία
    $fields = str_getcsv($row);
    // Ελέγχει αν η γραμμή περιέχει τουλάχιστον 2 πεδία(κλειδί και τιμή)
    if (count($fields) >= 2) {
        // Εξάγεται το πρώτο πεδίο της γραμμής και αποθηκεύεται στην μεταβλητή $key
        $key = trim($fields[0]);
        // Εξάγεται το δεύτερο πεδίο της γραμμής και αποθηκεύεται στην μεταβλητή $value
        $value = trim($fields[1]);
        // Ελέγχει αν το κλειδί είναι "General Info"
        if ($key === "General Info") {
            // Αν το κλειδί είναι "General Info" χωρίζει την τιμή σε υποτιμές με βάση το ελληνικό
            ερωτηματικό ";";
            $csvdata[$key] = array_map('trim', explode(";", $value));
            // Αν το κλειδί είναι "Courses" ή "Other Links"
        } elseif ($key === "Courses" || $key === "Other Links") {
            // Χωρίζει την τιμή σε υποτιμές με βάση το ελληνικό ερωτηματικό ";";
            $csvdata[$key] = array_map('trim', explode(";", $value));
        }
    }
}

// Εξάγει το πεδίο "General Info" από το πίνακα $csvdata και το αποθηκεύει στη μεταβλητή
$generalInfo
$generalInfo = $csvdata["General Info"] ?? [];

// Ελέγχει αν το κλειδί "Courses" υπάρχει στο πίνακα $csvdata και αν υπάρχει εκτελεί την
array_map() για επεξεργασία
$courses = isset($csvdata["Courses"]) ? array_map(function($entry) {
    // Επεξεργασία κάθε εισόδου, χωρίζοντας την συμβολοσειρά $entry σε 2
    κομμάτια χρησιμοποιώντας το χαρακτήρα "|"
    $parts = explode("|", $entry, 2);
    // Αν ο χωρισμός ήταν επιτυχής επιστρέφει [URL, Course Name] ενώ αν αποτύχει εμφανίζει
    μήνυμα σφάλματος
```

```

    return count($parts) === 2 ? [trim($parts[1]), trim($parts[0])] : ["#", "Invalid Course
Format: " . htmlspecialchars($entry)];
}, $csvdata["Courses"]) : [];

// Ελέγχει αν το κλειδί "Other Links" υπάρχει στο πίνακα $csvdata και αν υπάρχει εκτελεί την
array_map() για επεξεργασία
$otherLinks = isset($csvdata["Other Links"]) ? array_map(function($entry) {
    // Επεξεργασία κάθε εισόδου, χωρίζοντας την συμβολοσειρά $entry σε 2 κομμάτια
    χρησιμοποιώντας το χαρακτήρα "|"
    $parts = explode("|", $entry, 2);
    // Αν ο χωρισμός ήταν επιτυχής επιστρέφει [URL, Course Name] ενώ αν αποτύχει εμφανίζει
    μήνυμα σφάλματος
    return count($parts) === 2 ? [trim($parts[1]), trim($parts[0])] : ["#", "Invalid Link
Format: " . htmlspecialchars($entry)];
}, $csvdata["Other Links"]) : [];

```

Σχήμα 4.26 Το τμήμα του κώδικα, στο οποίο γίνεται η μετατροπή των δεδομένων του CSV σε πίνακα και η εξαγωγή των πεδίων General Info, Courses, Other Links

Το στάδιο της απόδοσης είναι παρόμοιο με το βασικό μοτίβο υλοποίησης, χρησιμοποιώντας βρόγχους επανάληψης “foreach” για να εμφανίσει το περιεχόμενο. Εκεί που διαφέρει το αρχείο παραγωγής περιεχομένου της ενότητας Home είναι στο ότι χωρίζει την οθόνη σε κομμάτια με την κύρια μπάρα στην οποία εμφανίζονται τα πεδία Courses και Other Links να καταλαμβάνει τα 8/12 της οθόνης και την πλάγια μπάρα που περιέχει το πεδίο General Info να καταλαμβάνει τα 4/12 της. Άλλη μια ιδιαιτερότητα είναι η χρήση της συνάρτησης “echo implode” για την εμφάνιση των πληροφοριών της πλάγιας μπάρας έτσι ώστε να επιτευχθεί η κατακόρυφη διάταξη των στοιχείων που ήταν και το επιθυμητό αποτέλεσμα.

```

<!-- HTML περιεχόμενο -->
<!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
περιεχόμενο -->
<div class="container">
    <!-- Δημιουργεί μια γραμμή που περιέχει στήλες που βοηθάει στο να κάνει τη διάταξη της
σελίδας πιο αποκριτική -->
    <div class="row">
        <!-- Δημιουργεί την κύρια στήλη που καταλαμβάνει τα 8/12 της οθόνης, έχει μέγεθος
γραμματοσειράς 20px και έχει ύψος το 100% του viewport -->
        <div class="col-md-8" style="font-size: 20px; height: 100vh;">
            <!-- Εμφανίζει την εικόνα χρησιμοποιώντας την κλάση img-responsive για
προσαρμόζεται στο μέγεθος της οθόνης και προσφέρει εναλλακτικό κείμενο για λόγους
προσβασιμότητας χάρις την συνάρτηση alt -->
            <br>
            <!-- Εμφανίζει την επικεφαλίδα -->
            <h2>Courses:</h2>
            <!-- Δημιουργεί κλάση lessons με μέγεθος γραμματοσειράς 20px-->

```

```

<div class="lessons" style="font-size: 20px;">
  <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε μάθημα
στον
        πίνακα $courses -->
  <?php foreach ($courses as $course): ?>
    <!-- Δημιουργεί σύνδεσμο, που περιέχει το URL του μαθήματος($course[0]), το
όνομα του μαθήματος (course[1]) και ανοίγει τον σύνδεσμο σε νέα καρτέλα(target="_blank") -
->
    <a href="<? = htmlspecialchars($course[0]) ?>" target="_blank"><? =
htmlspecialchars($course[1]) ?></a>
    <!-- Αλλαγή γραμμής μετά από κάθε σύνδεσμο -->
    <br>
    <!-- Τέλος βρόγχου επανάληψης -->
  <?php endforeach; ?>
</div>
<!-- Εμφανίζει κεφαλίδα -->
<h2>Other Links:</h2>
<div>
  <!-- Βρόχος επανάληψης για κάθε σύνδεσμο στον
πίνακα
        $otherLinks -->
  <?php foreach ($otherLinks as $link): ?>
    <!-- Δημιουργεί σύνδεσμο, που περιέχει το URL($link[0]), το κείμενο του
συνδέσμου (link[1]) και ανοίγει τον σύνδεσμο σε νέα καρτέλα(target="_blank")-->
    <p><a href="<? = htmlspecialchars($link[0]) ?>" target="_blank"> <? =
htmlspecialchars($link[1]) ?> </a></p>
    <!-- Τέλος βρόγχου επανάληψης -->
  <?php endforeach; ?>
  <!-- Κλείνει τις ανοιχτές διαιρέσεις του κώδικα -->
</div>
</div>

<!-- Πλάγια μπάρα με το πεδίο General Info που πιάνει τα 4/12 της οθόνης με μέγεθος
γραμματοσειράς 18px -->
<div class="col-md-4" style="font-size: 18px;">
  <!-- Δημιουργεί επικεφαλίδα με τίτλο General Info και μέγεθος γραμματοσειράς 24px -->
  <div style="font-size: 24px;"><b>General Info</b></div><br>
  <!-- Παίρνει όλα τα στοιχεία του πίνακα $generalInfo, τα ενώνει σε μια ενιαία συμβολοσειρά
και χρησιμοποιεί την ετικέτα αλλαγής γραμμής <br> ως διαχωριστικό μεταξύ των στοιχείων
  <p><?php echo implode("<br>", $generalInfo); ?></p>
  <!-- Κλείνει τις ανοιχτές διαιρέσεις του κώδικα -->
</div>
</div>
</div>

```

Σχήμα 4.27 Το στάδιο της απόδοσης του κώδικα της ενότητας Home

4.3.5.4 Workshop

Η ενότητα Workshop βρίσκεται αναρτημένη υπό τη διεύθυνση:

<https://microengineering.iem.ihu.gr/kafantaris/static/workshop.html> και περιέχει πληροφορίες για το ετήσιο συνέδριο, το CSV που της τροφοδοτεί δεδομένα αποτελείται από 3 πεδία: General Info, Courses, Other Links.

Ακολουθεί ο πλήρης κώδικας του:

```
<?php
```

```
// Σύνδεσμος Dropbox για το CSV
```

```
$csvUrl =
```

```
'https://www.dropbox.com/scl/fi/wm0fsdeufub0cgidfgqrd/workshop.csv?rlkey=hxux0wq04qc3wx48l520q1e82&dl=1';
```

```
// Αρχικοποιεί την cURL συνεδρία
```

```
$ch = curl_init();
```

```
// Ορίζει τις επιλογές για την cURL συνεδρία
```

```
// Ορισμός URL από το οποίο κατεβαίνει το αρχείο
```

```
curl_setopt($ch, CURLOPT_URL, $csvUrl);
```

```
// Επιστροφή δεδομένων ως συμβολοσειρές
```

```
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
```

```
// Ακολουθεί HTTP ανακατευθύνσεις
```

```
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
```

```
// Εκτέλεση συνεδρίας και επιστροφή ως συμβολοσειρά
```

```
$data = curl_exec($ch);
```

```
// Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
```

```
if (curl_errno($ch)) {
```

```
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
```

```
    error_log('Curl error: ' . curl_error($ch));
```

```
    // Μήνυμα σφάλματος
```

```
    echo 'Error: Unable to fetch the CSV file.';
```

```
    // Τερματισμός διαδικασίας
```

```
    exit;
```

```
}
```

```
// Κλείνει την cURL συνεδρία
```

```
curl_close($ch);
```

```
// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
```

```
$localCsvFile = 'workshop.csv';
```

```
// Αποθήκευση τα κατεβασμένα δεδομένα στον τοπικό φάκελο
```

```
if (file_put_contents($localCsvFile, $data) === false) {
```

```

// Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
error_log('Failed to save the downloaded CSV to local storage.');
```

// Μήνυμα σφάλματος
echo 'Error: Unable to save the CSV file.';
// Τερματισμός διαδικασίας
exit;

}

// Επεξεργασία των δεδομένων του CSV
// Χωρίζει την συμβολοσειρά σε πίνακα γραμμών χρησιμοποιώντας το χαρακτήρα "\n" ως
διαχωριστικό
\$rows = explode("\n", \$data);
// Αρχικοποιεί ένα κενό πίνακα που θα κρατάει τα επεξεργασμένα δεδομένα
\$csvdata = [];
// Βρόχος επανάληψης που επαναλαμβάνεται για κάθε γραμμή του πίνακα \$rows
foreach (\$rows as \$row) {
// Διαχωρίζει τις γραμμές της συμβολοσειράς \$rows σε ξεχωριστά πεδία με βάση των
χαρακτήρα "," και χρησιμοποιεί εισαγωγικά για περικύκλωση τιμών
\$csvdata[] = str_getcsv(\$row, ",", ""); // Διαχειρίζεται με ακρίβεια τις τιμές που περιέχονται
σε εισαγωγικά
}

// Ελέγχει ότι το αρχείο CSV έχει τουλάχιστον 10 γραμμές(αναμένει συγκεκριμένη δομή)
if (count(\$csvdata) < 10) {
// Αν οι γραμμές είναι λιγότερες από 10 τερματίζει την διαδικασία και εμφανίζει το ανάλογο
μήνυμα
die("Invalid CSV format.");
}

// Δημιουργία βοηθητικής συνάρτησης που δέχεται 2 παραμέτρους(το \$text που έχει το κείμενο
προς επεξεργασία(μπορεί να είναι URL ή κείμενο με URL) και το \$defaultText που είναι το
προεπιλεγμένο κείμενο για τον σύνδεσμο) και επιστρέφει HTML κώδικα για ασφαλείς
συνδέσμους
function formatLink(\$text, \$defaultText = "Click here") {
// Ελέγχει αν το κείμενο(\$text) περιέχει τον χαρακτήρα "|"
if (strpos(\$text, '|') !== false) {
// Χωρίζει το κείμενο σε 2 μέρη βάση του χαρακτήρα "|" και αποθηκεύει το πρώτο μέρος
στη μεταβλητή \$label και το δεύτερο μέρος στη μεταβλητή \$url
list(\$label, \$url) = explode('|', \$text, 2);
// Ελέγχει αν το \$url είναι έγκυρη διεύθυνση URL
if (filter_var(\$url, FILTER_VALIDATE_URL)) {
// Δημιουργεί σύνδεσμο με URL το \$url, κείμενο το \$label και τον ανοίγει σε νέα
καρτέλα
return '' .
htmlspecialchars(\$label) . '';

```

    }
    // Αν το $text δεν περιέχει τον χαρακτήρα "|" αλλά είναι έγκυρο URL
    } elseif (filter_var($text, FILTER_VALIDATE_URL)) {
        // Δημιουργεί σύνδεσμο με URL το $text και κείμενο συνδέσμου το προεπιλεγμένο κείμενο
        $defaultText και τον ανοίγει σε νέα καρτέλα
        return '<a href=' . htmlspecialchars($text) . " target='_blank'>'.
        htmlspecialchars($defaultText) . '</a>';
    }
    // Αν το $text δεν είναι σε μορφή κείμενο|URL ούτε είναι έγκυρο URL, τότε επιστρέφει το
    κείμενο χωρίς να δημιουργήσει σύνδεσμο
    return htmlspecialchars($text);
}

// Εξαγωγή περιεχομένου από το CSV
// Παίρνει την τιμή από την δεύτερη στήλη της πρώτης γραμμής και την αποθηκεύει στην
μεταβλητή $eventTitle(Αν η τιμή είναι null/δεν έχει οριστεί χρησιμοποιεί κενή συμβολοσειρά)
$eventTitle = $csvdata[0][1] ?? "";
// Παίρνει την τιμή από την δεύτερη στήλη της δεύτερης γραμμής και την αποθηκεύει στην
μεταβλητή $eventDate(Αν η τιμή είναι null/δεν έχει οριστεί χρησιμοποιεί κενή συμβολοσειρά)
$eventDate = $csvdata[1][1] ?? "";
// Παίρνει την τιμή από την δεύτερη στήλη της τρίτης γραμμής και την αποθηκεύει στην
μεταβλητή $introduction(Αν η τιμή είναι null/δεν έχει οριστεί χρησιμοποιεί κενή συμβολοσειρά)
$introduction = $csvdata[2][1] ?? "";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της τέταρτης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάσει του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$topics = isset($csvdata[3][1]) ? explode(";", $csvdata[3][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της πέμπτης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάσει του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$news = isset($csvdata[4][1]) ? explode(";", $csvdata[4][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της έκτης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$registration = isset($csvdata[5][1]) ? formatLink($csvdata[5][1], "Register Here") : "";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της έβδομης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$abstractSubmission = isset($csvdata[6][1]) ? formatLink($csvdata[6][1], "Submit Abstract")
: "";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της όγδοης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$programme = isset($csvdata[7][1]) ? formatLink($csvdata[7][1], "View Programme") : "";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της ένατης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάσει του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$pastEvents = isset($csvdata[8][1]) ? explode(";", $csvdata[8][1]) : [];

```

```

// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της δέκατης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάση του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$ sponsors = isset($csvdata[9][1]) ? explode(";", $csvdata[9][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της ενδέκατης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$contactInfo = isset($csvdata[10][1]) ? formatLink($csvdata[10][1], "Contact Us") : ";

?>
<!-- HTML περιεχόμενο -->
<!-- Ανοίγει το βρόγχο body -->
<body>
    <!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το
    περιεχόμενο -->
    <div class="container">
        <!-- Εμφανίζει την μεταβλητή $eventTitle ως επικεφαλίδα υψηλότερης τάξης (h1) -->
        <h1><?php echo htmlspecialchars($eventTitle); ?></h1>
        <!-- Εμφανίζει την μεταβλητή $eventDate ως επικεφαλίδα μικρότερης τάξης (h3) -->
        <h3><?php echo htmlspecialchars($eventDate); ?></h3>
        <!-- Χρησιμοποιείται ακόμα μικρότερης τάξης επικεφαλίδα (h4) για να εμφανίσει την
        μεταβλητή $introduction και δημιουργείται κλάση για το στυλ της-->
        <h4 class="intro-text"><?php echo htmlspecialchars($introduction); ?></h4>
        <!-- CSS Styling -->
        <style>
            .intro-text {
                font-family: 'Lato', sans-serif; /* Ορίζει την γραμματοσειρά Lato ως πρώτη επιλογή
                και αν δεν είναι διαθέσιμη οποιαδήποτε γραμματοσειρά sans-serif */
                font-size: 16px; /* Ορίζει το μέγεθος της γραμματοσειράς 16px */
            }
        /* Τέλος CSS Styling */
        </style>
        <!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
        <h2>Scientific topics include but are not limited to:</h2>
        <!-- Δημιουργία μιας λίστας με κουκκίδες -->
        <ul>
            <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε στοιχείο στον πίνακα $topics --
            >
            <?php foreach ($topics as $topic)
                // Εκτυπώνει την μεταβλητή $topic ως στοιχείο της λίστας με κουκκίδες
                { echo "<li>" . htmlspecialchars($topic) . "</li>"; } ?>
            <!-- Κλείνει την λίστα με κουκκίδες -->
            </ul>
            <!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
            <h2>News</h2>
            <div>

```

```

<!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε στοιχείο στον πίνακα $news --
>
<?php foreach ($news as $newsItem): ?>
    <!-- Ελέγχει αν το στοιχείο του πίνακα $news περιέχει τον χαρακτήρα "|" -->
    <?php if (strpos($newsItem, '|') !== false): ?>
        <!-- Εάν τον περιέχει προσθέτει το στοιχείο στον νέο πίνακα $newsWithLinks -->
        <?php $newsWithLinks[] = $newsItem; ?>
    <!-- Εάν η είδηση δεν περιέχει τον χαρακτήρα "|" (είναι απλό κείμενο) -->
    <?php else: ?>
        <!-- Εμφανίζει άμεσα την είδηση μέσα στην παράγραφο <p> -->
        <p><?php echo htmlspecialchars($newsItem); ?></p>
    <!-- Τέλος δομής ελέγχου -->
    <?php endif; ?>
<!-- Τέλος βρόγχου επανάληψης -->
<?php endforeach; ?>
</div>
<!-- Ελέγχει αν ο πίνακας $newsWithLinks δεν είναι κενός -->
<?php if (!empty($newsWithLinks)): ?>
    <!-- Δημιουργεί μια λίστα με κουκκίδες -->
    <ul>
        <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε στοιχείο στον πίνακα $newsWithLinks -->
        <?php foreach ($newsWithLinks as $newsItem): ?>
            <!-- Δημιουργεί στοιχείο λίστας με την βοήθεια της συνάρτησης formatLink() -->
            <li><?php echo formatLink($newsItem, "Read more"); ?></li>
        <!-- Τέλος βρόγχου επανάληψης -->
        <?php endforeach; ?>
    <!-- Τέλος λίστας με κουκκίδες -->
    </ul>
<!-- Τέλος δομής ελέγχου -->
<?php endif; ?>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Registration</h2>
<div>
    <?php
        // Ανακτά τα δεδομένα από την 6η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
        // συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
        // συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
        $registrationParts = explode(";", $csvdata[5][1] ?? "");
        // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία
        // που δημιουργήθηκαν από τη συνάρτηση explode()
        foreach ($registrationParts as $part) {
            // Ελέγχει αν η συμβολοσειρά $part περιέχει το χαρακτήρα "|"
            if (strpos($part, '|') !== false) {

```

```

        // Αν τον περιέχει δημιουργεί σύνδεσμο με την βοήθεια της συνάρτησης
formatLink()
        echo "<p>" . formatLink($part, "More Info") . "</p>";
    } else {
        // Αν δεν τον περιέχει εμφανίζει το κείμενο της συμβολοσειράς $part
        echo "<p>" . htmlspecialchars($part) . "</p>";
    }
}
?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Abstract Submission</h2>
<div>
    <?php
        // Ανακτά τα δεδομένα από την 7η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
        // συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
        // συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
        $abstractParts = explode(";", $csvdata[6][1] ?? "");
        // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία που δημιουργήθηκαν
        // από τη συνάρτηση explode()
        foreach ($abstractParts as $part) {
            // Εμφανίζει την συμβολοσειρά $part αφαιρώντας τα κενά από την αρχή και το τέλος
            // της(trim) και μετατρέποντας τους χαρακτήρες αλλαγής σειράς (\n) σε ετικέτες HTML
            echo "<p>" . nl2br(htmlspecialchars(trim($part))) . "</p>";
        }
    ?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Programme</h2>
<div>
    <?php
        // Ανακτά τα δεδομένα από την 8η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
        // συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
        // συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
        $programmeParts = explode(";", $csvdata[7][1] ?? "");
        // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία
        // που δημιουργήθηκαν από τη συνάρτηση explode()
        foreach ($programmeParts as $part) {
            // Εμφανίζει την συμβολοσειρά $part αφαιρώντας τα κενά από την αρχή και το τέλος
            // της(trim) και μετατρέποντας τους χαρακτήρες αλλαγής σειράς (\n) σε ετικέτες HTML
            echo "<p>" . nl2br(htmlspecialchars(trim($part))) . "</p>";
        }
    ?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->

```

```

<h2>Past Events</h2>
<div>
  <?php
    // Ανακτά τα δεδομένα από την 9η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
    // συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
    // συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
    $pastEventsParts = explode(";", $csvdata[8][1] ?? "");
    // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία
    // που δημιουργήθηκαν από τη συνάρτηση explode()
    foreach ($pastEventsParts as $sevent) {
      // Ελέγχει αν η συμβολοσειρά $sevent περιέχει το χαρακτήρα "|"
      if (strpos($sevent, '|') !== false) {
        // Αν τον περιέχει δημιουργεί σύνδεσμο με την βοήθεια της συνάρτησης
        formatLink()
        echo "<p>" . formatLink($sevent, "View past event") . "</p>";
      } else {
        // Αν δεν τον περιέχει εμφανίζει το κείμενο της συμβολοσειράς $sevent
        echo "<p>" . htmlspecialchars($sevent) . "</p>";
      }
    }
  ?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Sponsors</h2>
<!-- Δημιουργεί κλάση sponsors -->
<div class="sponsors">
  <?php
    // Ανακτά τα δεδομένα από τη 10η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
    // συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
    // συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
    $sponsorsParts = explode(";", $csvdata[9][1] ?? "");
    // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία που δημιουργήθηκαν
    // από τη συνάρτηση explode()
    foreach ($sponsorsParts as $sponsor) {
      // Ελέγχει αν η συμβολοσειρά $sevent περιέχει το χαρακτήρα "|"
      if (strpos($sponsor, '|') !== false) {
        // Χωρίζει την συμβολοσειρά σε 2 μέρη: Όνομα χορηγού και URL
        list($named, $url) = explode('|', $sponsor, 2);
        // Δημιουργεί τη διαδρομή για την εικόνα του χορηγού, αντικαθιστώντας τα κενά με
        // κάτω παύλες(_), μετατρέποντας τα σε πεζά και προσθέτοντας το path images/ στην αρχή και την
        // επέκταση .jpg στο τέλος του
        $imageSrc = "images/" . strtolower(str_replace(' ', '_', $named)) . ".jpg";
        // Δημιουργεί ένα σύνδεσμο με URL το $url που ανοίγει σε νέα καρτέλα. Ο σύνδεσμος
        // περιέχει μια εικόνα με πηγή το $imageSrc, εναλλακτικό κείμενο το όνομα του χορηγού ($named)
        // και με κλάση sponsor-image για να διαμορφώσει το στυλ του

```

```

        echo '<a href="'. htmlspecialchars($url) . '" target="_blank">
            
            </a>';
    }
}
?>
</div>
<!-- CSS Styling -->
<style>
    .sponsors {
        display: flex; /* Χρησιμοποιεί το μοντέλο διάταξης flexbox για ευέλικτη διάταξη των
στοιχείων */
        flex-wrap: wrap; /* Επιτρέπει στις εικόνες να μεταφέρονται
στην επόμενη γραμμή αν χωράνε */
        gap: 20px; /* Προσθέτει 20px κενό μεταξύ των εικόνων */
    }

    .sponsor-image {
        max-width: 400px; /* Περιορίζει το μέγιστο πλάτος των εικόνων
στα 400px */
        height: auto; /* Διατηρεί τις αναλογίες της εικόνας */
        transition: transform 0.3s; /* Προετοιμάζει το στοιχείο για το εφέ hover που
ακολουθεί με επιβράδυνση 0.3 δευτερόλεπτα στο τέλος της κίνησης */
    }

    .sponsor-image:hover {
        transform: scale(1.1); /* Όταν ο χρήστης περάσει το ποντίκι πάνω από την εικόνα,
μεγεθύνεται κατά 10% */
    }
}
/* Τέλος CSS Styling */
</style>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2 class="contact-header">Contact Information</h2>
<!-- Δημιουργεί κλάση contact-container -->
<div class="contact-container">
    <!-- Δημιουργεί κλάση contact-details -->
    <div class="contact-details">
        <!-- Εικονίδιο φακέλου από τη βιβλιοθήκη Font Awesome -->
        <i class="fa fa-envelope"></i>
        <!-- Εμφανίζει τις πληροφορίες επικοινωνίας από τη
μεταβλητή $contactInfo -->
        <span><?php echo $contactInfo; ?></span>
    </div>
    <!-- Δημιουργεί την κλάση contact-details -->

```

```

<div class="contact-details">
<!-- Εικονίδιο κατασκευαστή χαρτών από τη βιβλιοθήκη Font Awesome -->
  <i class="fa fa-map-marker"></i>
  <!-- Εμφανίζει αυτό το μήνυμα -->
  <span>You can find us at the Laboratory of Microsystems, Department of Industrial
Engineering & Management, International Hellenic University, Greece 57400</span>
</div>
<!-- Δημιουργεί την κλάση map-container -->
<div class="map-container">
  <!-- Ενσωματώνει τον χάρτη Google Maps, με πλάτος που γεμίζει τη κλάση του ύψος
350 pixels και αφαιρεί το περίγραμμα της ετικέτας iframe -->
  <iframe id="map"
    src="https://www.google.com/maps/embed?pb=!1m14!1m8!1m3!1d1513.390870
6703046!2d22.803731837959297!3d40.656741346896084!3m2!1i1024!2i768!4f13.1!3m3!1m
2!1s0x0%3A0xb9b8a179c4d37e55!2sAlexandrio+Technologiko+Ekpedeftiko+Idrima+Thessa
lonikis!5e0!3m2!1sen!2sus!4v1452062855477"
    width="100%" height="350" frameborder="0">
  </iframe>
</div>
</div>
<!-- CSS Styling -->
<style>
.contact-header {
  text-align: center; /* Κεντράρει το header */
  font-size: 28px; /* Ορίζει το μέγεθος της γραμματοσειράς στα 28px */
  font-weight: bold; /* Έντονη γραμματοσειρά */
  color: #333; /* Σκούρο γκρι χρώμα */
  margin-bottom: 15px; /* Προσθέτει κενό κάτω από την επικεφαλίδα */
}

.contact-container {
  background-color: #f9f9f9; /* Ανοιχτό γκρι φόντο */
  padding: 20px; /* Προσθέτει εσωτερικό περιθώριο 20px */
  border-radius: 10px; /* Στρογγυλεμένες γωνίες */
  box-shadow: 2px 2px 15px rgba(0, 0, 0, 0.1); /* Σκιά */
  text-align: center; /* Κεντράρει το κείμενο */
  max-width: 700px; /* Μέγιστο πλάτος */
  margin: auto; /* Κεντράρισμα οριζόντια */
}

.contact-details {
  font-size: 18px; /* Ορίζει το μέγεθος της γραμματοσειράς στα 18px */
  color: #333; /* Σκούρο γκρι χρώμα */
  font-weight: bold; /* Έντονη γραμματοσειρά */
  margin: 10px; /* Ορίζει εξωτερικό περιθώριο 10px */
}

```

```

    display: flex; /* Χρησιμοποιεί το μοντέλο διάταξης flexbox για ευέλικτη διάταξη των
στοιχείων */
    align-items: center; /* Κεντράρει το κείμενο */
    justify-content: center; /* Οριζόντια στοίχιση στο κέντρο */
    gap: 10px; /* Ορίζει την απόσταση μεταξύ εικονιδίου και κειμένου στα 10px */
}

.contact-details i {
    font-size: 22px; /* Ορίζει το μέγεθος της γραμματοσειράς στα 22px */
    color: #007bff; /* Μπλε χρώμα */
}

.map-container {
    margin-top: 15px; /* Κενό πάνω από το χάρτη στα 15px */
    border-radius: 10px; /* Στρογγυλεμένες γωνίες */
    overflow: hidden; /* Αποκρύπτει οτιδήποτε ξεπερνά τα όρια */
}

#map {
    border: none; /* Χωρίς περίγραμμα */
    border-radius: 10px; /* Στρογγυλεμένες γωνίες */
}
/* Τέλος CSS Styling */
</style>
<!-- Κλείνει ανοιχτή διαίρεση κώδικα -->
</div>
<!-- Κλείνει το βρόγχο body -->
</body>

```

Σχήμα 4.28 Ο πλήρης κώδικας του αρχείου παραγωγής περιεχομένου της ενότητας Workshop

Σε αντίθεση με όλα τα υπόλοιπα αρχεία του συστήματος, το αρχείο παραγωγής περιεχομένου της ενότητας Workshop δεν βασίζεται καθόλου στο κοινό μοτίβο επεξεργασίας πινάκων CSV. Αντιπροσωπεύει μια πλήρως εξατομικευμένη και μοναδική υλοποίηση, σχεδιασμένη ειδικά για να αντιμετωπίσει την εντελώς διαφορετική δομή των δεδομένων του συνέδριου, αποτελώντας ουσιαστικά μια αυτόνομη σελίδα μέσα στο κύριο ιστότοπο. Το αρχείο παραγωγής περιεχομένου της ενότητας Workshop περιέχει μη δομημένο, ελεύθερης μορφής κείμενο οργανωμένο σε ενότητες, όπου κάθε γραμμή αντιπροσωπεύει ένα διαφορετικό τμήμα περιεχομένου. Όπως και στο αρχείο παραγωγής περιεχομένου της ενότητας Home χρησιμοποιείται συνδυασμός των συναρτήσεων “explode()” και “str_getcsv()” με σκοπό την αποδόμηση του περιεχομένου και την ταξινόμηση του σε ένα πίνακα με πρόσβαση μέσω δείκτη (Index-based), όπου κάθε γραμμή αποτελεί έναν υποπίνακα.

// Επεξεργασία των δεδομένων του CSV

```

// Χωρίζει την συμβολοσειρά σε πίνακα γραμμών χρησιμοποιώντας το χαρακτήρα "\n" ως
διαχωριστικό
$rows = explode("\n", $data);
// Αρχικοποιεί ένα κενό πίνακα που θα κρατάει τα επεξεργασμένα δεδομένα
$csvdata = [];
// Βρόχος επανάληψης που επαναλαμβάνεται για κάθε γραμμή του πίνακα $rows
foreach ($rows as $row) {
    // Διαχωρίζει τις γραμμές της συμβολοσειράς $rows σε ξεχωριστά πεδία με βάση των
    χαρακτήρα "," και χρησιμοποιεί εισαγωγικά για περικύκλωση τιμών
    $csvdata[] = str_getcsv($row, ",", ""); // Διαχειρίζεται με ακρίβεια τις τιμές που περιέχονται
    σε εισαγωγικά
}

```

Σχήμα 4.29 Το τμήμα του κώδικα, στο οποίο γίνεται η αποδόμηση του περιεχομένου και η ταξινόμηση του σε έναν πίνακα με πρόσβαση μέσω δείκτη

Στη συνέχεια ελέγχεται ότι το αρχείο CSV έχει τουλάχιστον δέκα γραμμές και δημιουργείται εξατομικευμένη βοηθητική συνάρτηση “formatLink()”, η οποία αναλαμβάνει να αναγνωρίζει και να μετατρέπει την ειδική σύνταξη Κείμενο|URL σε έτοιμους HTML συνδέσμους. Η συνάρτηση δέχεται ως εισόδους μια συμβολοσειρά (\$text) και ένα προεπιλεγμένο κείμενο που θα χρησιμοποιηθεί μόνο σε περίπτωση που κάποιος σύνδεσμος δεν περιέχει δικό του κείμενο. Έπειτα με την συνάρτηση “strpos()” ελέγχεται αν η συμβολοσειρά περιέχει το χαρακτήρα “|” και αν τον περιέχει την χωρίζει με την συνάρτηση “explode()” σε δύο μέρη βάσει αυτού: το κείμενο του συνδέσμου(\$label) και τη διεύθυνση URL του(\$url). Κατόπιν τα δύο αυτά μέρη χρησιμοποιούνται για να δημιουργηθεί ο σύνδεσμος. Αν δεν τον περιέχει και η συμβολοσειρά αποτελεί έγκυρη URL, τότε επιστρέφει σύνδεσμο με URL τη συμβολοσειρά και κείμενο του συνδέσμου το προεπιλεγμένο κείμενο. Στο τέλος για οποιαδήποτε άλλη περίπτωση απλώς επιστρέφει το κείμενο της συμβολοσειράς χωρίς να δημιουργεί σύνδεσμο.

```

// Δημιουργία βοηθητικής συνάρτησης που δέχεται 2 παραμέτρους(το $text που έχει το κείμενο
προς επεξεργασία(μπορεί να είναι URL ή κείμενο με URL) και το $defaultText που είναι το
προεπιλεγμένο κείμενο για τον σύνδεσμο) και επιστρέφει HTML κώδικα για ασφαλείς
συνδέσμους
function formatLink($text, $defaultText = "Click here") {
    // Ελέγχει αν το κείμενο($text) περιέχει τον χαρακτήρα "|"
    if (strpos($text, '|') !== false) {
        // Χωρίζει το κείμενο σε 2 μέρη βάση του χαρακτήρα "|" και αποθηκεύει το πρώτο μέρος
        στη μεταβλητή $label και το δεύτερο μέρος στη μεταβλητή $url
        list($label, $url) = explode('|', $text, 2);
        // Ελέγχει αν το $url είναι έγκυρη διεύθυνση URL
        if (filter_var($url, FILTER_VALIDATE_URL)) {
            // Δημιουργεί σύνδεσμο με URL το $url, κείμενο το $label και τον ανοίγει σε νέα
            καρτέλα
            return '<a href="' . htmlspecialchars($url) . '" target="_blank">'.
            htmlspecialchars($label) . '</a>';
        }
    }
}

```

```

    }
    // Αν το $text δεν περιέχει τον χαρακτήρα "|" αλλά είναι έγκυρο URL
    } elseif (filter_var($text, FILTER_VALIDATE_URL)) {
        // Δημιουργεί σύνδεσμο με URL το $text και κείμενο συνδέσμου το προεπιλεγμένο κείμενο
        $defaultText και τον ανοίγει σε νέα καρτέλα
        return '<a href=' . htmlspecialchars($text) . " target='_blank'>".
        htmlspecialchars($defaultText) . '</a>';
    }
    // Αν το $text δεν είναι σε μορφή κείμενο|URL ούτε είναι έγκυρο URL, τότε επιστρέφει το
    κείμενο χωρίς να δημιουργήσει σύνδεσμο
    return htmlspecialchars($text);
}

```

Σχήμα 4.30 Το τμήμα του κώδικα, στο οποίο δημιουργείται η βοηθητική συνάρτηση “formatLink()”

Στον πίνακα που δημιουργείται για το αρχείο παραγωγής περιεχομένου της ενότητας Workshop κάθε γραμμή αντιστοιχεί σε ένα διαφορετικό τμήμα περιεχομένου (π.χ. η πρώτη γραμμή περιέχει τον τίτλο της διοργάνωσης, η δεύτερη την ημερομηνία κ.λπ.) και το περιεχόμενο κάθε γραμμής διαφέρει ως προς τη μορφή και την πολυπλοκότητα του καθώς μπορεί να απαρτίζεται από απλό κείμενο, εκτενείς παραγράφους ή συνδυασμό κειμένου και υπερσυνδέσμων. Για αυτό και το τελευταίο βήμα της διαδικασίας επεξεργασίας των δεδομένων του αρχείου CSV είναι η εξαγωγή του περιεχομένου του με βάση τον δείκτη (Index-based Access). Σε αντίθεση με τα υπόλοιπα αρχεία CSV, όπου η σημασία των δεδομένων προκύπτει από τις ετικέτες των επικεφαλίδων τους, εδώ η σημασία προκύπτει από τη θέση της κάθε γραμμής εντός του αρχείου. Παρόλο που αυτή η προσέγγιση είναι λιγότερο “αυτόματη”, λόγω της “χειροκίνητης” ανάθεσης σημασίας στα δεδομένα εντός του κώδικα, είναι και η μοναδική βιώσιμη λύση για τη διαχείριση του όγκου και της πολυμορφίας των πληροφοριών του αρχείου CSV. Αυτό έχει ως αποτέλεσμα ο κώδικας να “γνωρίζει” εκ των προτέρων, βάσει του δείκτη, ποιο τμήμα περιεχομένου αντιπροσωπεύει η κάθε γραμμή, αποθηκεύοντάς το στην αντίστοιχη μεταβλητή, και απλοποιώντας την διαδικασία απόδοσης που ακολουθεί

```

// Εξαγωγή περιεχομένου από το CSV
// Παίρνει την τιμή από την δεύτερη στήλη της πρώτης γραμμής και την αποθηκεύει στην
μεταβλητή $eventTitle (Αν η τιμή είναι null/δεν έχει οριστεί χρησιμοποιεί κενή συμβολοσειρά)
$eventTitle = $csvdata[0][1] ?? "";
// Παίρνει την τιμή από την δεύτερη στήλη της δεύτερης γραμμής και την αποθηκεύει στην
μεταβλητή $eventDate (Αν η τιμή είναι null/δεν έχει οριστεί χρησιμοποιεί κενή συμβολοσειρά)
$eventDate = $csvdata[1][1] ?? "";
// Παίρνει την τιμή από την δεύτερη στήλη της τρίτης γραμμής και την αποθηκεύει στην
μεταβλητή $introduction (Αν η τιμή είναι null/δεν έχει οριστεί χρησιμοποιεί κενή συμβολοσειρά)
$introduction = $csvdata[2][1] ?? "";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της τέταρτης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάση του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα

```

```

$topics = isset($csvdata[3][1]) ? explode(";", $csvdata[3][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της πέμπτης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάσει του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$news = isset($csvdata[4][1]) ? explode(";", $csvdata[4][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της έκτης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$registration = isset($csvdata[5][1]) ? formatLink($csvdata[5][1], "Register Here") : ";";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της έβδομης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$abstractSubmission = isset($csvdata[6][1]) ? formatLink($csvdata[6][1], "Submit Abstract")
: ";";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της όγδοης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$programme = isset($csvdata[7][1]) ? formatLink($csvdata[7][1], "View Programme") : ";";
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της ένατης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάσει του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$pastEvents = isset($csvdata[8][1]) ? explode(";", $csvdata[8][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της δέκατης γραμμής. Αν υπάρχει χωρίζει την
συμβολοσειρά βάσει του ";" και δημιουργεί πίνακα, ενώ αν δεν υπάρχει δημιουργεί κενό πίνακα
$sponsors = isset($csvdata[9][1]) ? explode(";", $csvdata[9][1]) : [];
// Ελέγχει αν υπάρχει τιμή στην δεύτερη στήλη της ενδέκατης γραμμής. Αν υπάρχει δημιουργεί
σύνδεσμο χρησιμοποιώντας την βοηθητική συνάρτηση formatLink και ορίζει το προεπιλεγμένο
κείμενο της, ενώ αν δεν υπάρχει χρησιμοποιεί κενή συμβολοσειρά
$contactInfo = isset($csvdata[10][1]) ? formatLink($csvdata[10][1], "Contact Us") : ";";

```

Σχήμα 4.31 Το τμήμα του κώδικα στο οποίο γίνεται η εξαγωγή του περιεχομένου από το CSV

Η διαδικασία απόδοσης για το αρχείο παραγωγής περιεχομένου της ενότητας Workshop διαφέρει από αυτή των υπόλοιπων σελίδων. Σε αντίθεση με τα άλλα αρχεία, όπου οι πίνακες CSV ακολουθούν μια αυστηρή, προκαθορισμένη δομή(με επικεφαλίδες στη πρώτη γραμμή και ομοιογενείς εγγραφές στις επόμενες), το συγκεκριμένο αρχείο χαρακτηρίζεται από μια ελεύθερη, ενιαία δομή εγγράφου. Αυτή η θεμελιώδης διαφορά καθιστά την χρήση του κοινού μοτίβου των βρόχων επανάληψης "foreach", που βασίζεται στην ύπαρξη πολλαπλών, όμοιων εγγραφών μη πρακτική. Για το λόγο αυτό, η προσέγγιση που υιοθετήθηκε βασίστηκε στην άμεση αντιστοίχιση μεταξύ της γραμμικής δομής του CSV και της ιεραρχικής δομής της γλώσσας προγραμματισμού HTML. Κάθε γραμμή του αρχείου CSV αντιστοιχίζεται σε μια συγκεκριμένη ενότητα της σελίδας, η οποία εισάγεται με μια κατάλληλη επικεφαλίδα με ετικέτα <h>(<h1>, <h2> κ.λπ.). Στη συνέχεια, το περιεχόμενο της κάθε γραμμής υποβάλλεται στις απαραίτητες τεχνικές επεξεργασίας (διαχωρισμός, δημιουργία συνδέσμων, δημιουργία λιστών) προκειμένου να παραχθεί το τελικό σώμα κειμένου που ακολουθεί την επικεφαλίδα. Αυτή η μοναδική προσέγγιση έχει ως αποτέλεσμα ο κώδικας να αντιμετωπίζει κάθε ενότητα <h> ως ένα

αυτοτελές κομμάτι, το οποίο αποδίδει το ένα μετά το άλλο στη σελίδα, και όχι ως μέρος μιας ενοποιημένης διαδικασίας απόδοσης ολόκληρου του περιεχομένου HTML, όπως συμβαίνει στα υπόλοιπα αρχεία παραγωγής περιεχομένου.

```
<!-- HTML περιεχόμενο -->
<!-- Ανοίγει το βρόγχο body -->
<body>
  <!-- Δημιουργεί ένα κεντραρισμένο πλαίσιο με αποκριτικό πλάτος στο οποίο θα μπει το περιεχόμενο -->
  <div class="container">
    <!-- Εμφανίζει την μεταβλητή $eventTitle ως επικεφαλίδα υψηλότερης τάξης (h1) -->
    <h1><?php echo htmlspecialchars($eventTitle); ?></h1>
    <!-- Εμφανίζει την μεταβλητή $eventDate ως επικεφαλίδα μικρότερης τάξης (h3) -->
    <h3><?php echo htmlspecialchars($eventDate); ?></h3>
    <!-- Χρησιμοποιείται ακόμα μικρότερης τάξης επικεφαλίδα (h4) για να εμφανίσει την μεταβλητή $introduction και δημιουργείται κλάση για το στυλ της-->
    <h4 class="intro-text"><?php echo htmlspecialchars($introduction); ?></h4>
    <!-- CSS Styling -->
    <style>
      .intro-text {
        font-family: 'Lato', sans-serif; /* Ορίζει την γραμματοσειρά Lato ως πρώτη επιλογή και αν δεν είναι διαθέσιμη οποιαδήποτε γραμματοσειρά sans-serif */
        font-size: 16px; /* Ορίζει το μέγεθος της γραμματοσειράς 16px */
      }
    /* Τέλος CSS Styling */
    </style>
    <!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
    <h2>Scientific topics include but are not limited to:</h2>
    <!-- Δημιουργία μιας λίστας με κουκκίδες -->
    <ul>
      <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε στοιχείο στον πίνακα $topics --
    >
      <?php foreach ($topics as $topic)
        // Εκτυπώνει την μεταβλητή $topic ως στοιχείο της λίστας με κουκκίδες
        { echo "<li>" . htmlspecialchars($topic) . "</li>"; } ?>
      <!-- Κλείνει την λίστα με κουκκίδες -->
    </ul>
    <!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
    <h2>News</h2>
    <div>
      <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε στοιχείο στον πίνακα $news --
    >
      <?php foreach ($news as $newsItem): ?>
        <!-- Ελέγχει αν το στοιχείο του πίνακα $news περιέχει τον χαρακτήρα "|" -->
        <?php if (strpos($newsItem, '|') !== false): ?>
```

```

        <!-- Εάν τον περιέχει προσθέτει το στοιχείο στον νέο πίνακα $newsWithLinks -->
        <?php $newsWithLinks[] = $newsItem; ?>
    <!-- Εάν η είδηση δεν περιέχει τον χαρακτήρα "|" (είναι απλό κείμενο) -->
    <?php else: ?>
        <!-- Εμφανίζει άμεσα την είδηση μέσα στην παράγραφο <p> -->
        <p><?php echo htmlspecialchars($newsItem); ?></p>
    <!-- Τέλος δομής ελέγχου -->
    <?php endif; ?>
<!-- Τέλος βρόγχου επανάληψης -->
<?php endforeach; ?>
</div>
<!-- Ελέγχει αν ο πίνακας $newsWithLinks δεν είναι κενός -->
<?php if (!empty($newsWithLinks)): ?>
    <!-- Δημιουργεί μια λίστα με κουκκίδες -->
    <ul>
        <!-- Βρόχος επανάληψης που επαναλαμβάνεται για κάθε στοιχείο στον πίνακα $newsWithLinks -->
        <?php foreach ($newsWithLinks as $newsItem): ?>
            <!-- Δημιουργεί στοιχείο λίστας με την βοήθεια της συνάρτησης formatLink() -->
            <li><?php echo formatLink($newsItem, "Read more"); ?></li>
        <!-- Τέλος βρόγχου επανάληψης -->
        <?php endforeach; ?>
    <!-- Τέλος λίστας με κουκκίδες -->
    </ul>
<!-- Τέλος δομής ελέγχου -->
<?php endif; ?>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Registration</h2>
<div>
    <?php
        // Ανακτά τα δεδομένα από την 6η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
        // συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
        // συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
        $registrationParts = explode(";", $csvdata[5][1] ?? "");
        // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία
        // που δημιουργήθηκαν από τη συνάρτηση explode()
        foreach ($registrationParts as $part) {
            // Ελέγχει αν η συμβολοσειρά $part περιέχει το χαρακτήρα "|"
            if (strpos($part, '|') !== false) {
                // Αν τον περιέχει δημιουργεί σύνδεσμο με την βοήθεια της συνάρτησης
                // formatLink()
                echo "<p>" . formatLink($part, "More Info") . "</p>";
            } else {
                // Αν δεν τον περιέχει εμφανίζει το κείμενο της συμβολοσειράς $part
                echo "<p>" . htmlspecialchars($part) . "</p>";
            }
        }
    </?php>

```

```

    }
  }
  ?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Abstract Submission</h2>
<div>
  <?php
    // Ανακτά τα δεδομένα από την 7η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
    συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
    συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
    $abstractParts = explode(";", $csvdata[6][1] ?? "");
    // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία που δημιουργήθηκαν
    από τη συνάρτηση explode()
    foreach ($abstractParts as $part) {
      // Εμφανίζει την συμβολοσειρά $part αφαιρώντας τα κενά από την αρχή και το τέλος
      της(trim) και μετατρέποντας τους χαρακτήρες αλλαγής σειράς (\n) σε ετικέτες HTML
      echo "<p>" . nl2br(htmlspecialchars(trim($part))) . "</p>";
    }
  ?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Programme</h2>
<div>
  <?php
    // Ανακτά τα δεδομένα από την 8η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
    συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
    συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
    $programmeParts = explode(";", $csvdata[7][1] ?? "");
    // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία
    που δημιουργήθηκαν από τη συνάρτηση explode()
    foreach ($programmeParts as $part) {
      // Εμφανίζει την συμβολοσειρά $part αφαιρώντας τα κενά από την αρχή και το τέλος
      της(trim) και μετατρέποντας τους χαρακτήρες αλλαγής σειράς (\n) σε ετικέτες HTML
      echo "<p>" . nl2br(htmlspecialchars(trim($part))) . "</p>";
    }
  ?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Past Events</h2>
<div>
  <?php
    // Ανακτά τα δεδομένα από την 9η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
    συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
    συμβολοσειρά αν τα δεδομένα δεν υπάρχουν

```

```

$pastEventsParts = explode(";", $csvdata[8][1] ?? "");
// Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία που δημιουργήθηκαν
από τη συνάρτηση explode()
foreach ($pastEventsParts as $sevent) {
    // Ελέγχει αν η συμβολοσειρά $part περιέχει το χαρακτήρα "|"
    if (strpos($sevent, '|') !== false) {
        // Αν τον περιέχει δημιουργεί σύνδεσμο με την βοήθεια της συνάρτησης
formatLink()
        echo "<p>" . formatLink($sevent, "View past event") . "</p>";
    } else {
        // Αν δεν τον περιέχει εμφανίζει το κείμενο της συμβολοσειράς $part
        echo "<p>" . htmlspecialchars($sevent) . "</p>";
    }
}
?>
</div>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2>Sponsors</h2>
<!-- Δημιουργεί κλάση sponsors -->
<div class="sponsors">
    <?php
        // Ανακτά τα δεδομένα από τη 10η γραμμή, 2η στήλη του πίνακα CSV και χωρίζει την
συμβολοσειρά σε μέρη χρησιμοποιώντας τον χαρακτήρα ";" ως διαχωριστικό. Επιστρέφει κενή
συμβολοσειρά αν τα δεδομένα δεν υπάρχουν
        $sponsorsParts = explode(";", $csvdata[9][1] ?? "");
        // Βρόχος επανάληψης που επαναλαμβάνεται για όλα τα στοιχεία που δημιουργήθηκαν
από τη συνάρτηση explode()
        foreach ($sponsorsParts as $sponsor) {
            // Ελέγχει αν η συμβολοσειρά $part περιέχει το χαρακτήρα "|"
            if (strpos($sponsor, '|') !== false) {
                // Χωρίζει την συμβολοσειρά σε 2 μέρη: Όνομα χορηγού και URL
                list($named, $url) = explode('|', $sponsor, 2);
                // Δημιουργεί τη διαδρομή για την εικόνα του χορηγού, αντικαθιστώντας τα κενά με
κάτω παύλες(_), μετατρέποντας τα σε πεζά και προσθέτοντας το path images/ στην αρχή και την
επέκταση .jpg στο τέλος του
                $imageSrc = "images/" . strtolower(str_replace(' ', '_', $named)) . ".jpg";
                // Δημιουργεί ένα σύνδεσμο με URL το $url που ανοίγει σε νέα καρτέλα. Ο σύνδεσμος
περιέχει μια εικόνα με πηγή το $imageSrc, εναλλακτικό κείμενο το όνομα του χορηγού ($named)
και με κλάση sponsor-image για να διαμορφώσει το στυλ του
                echo '<a href="' . htmlspecialchars($url) . '" target="_blank">
                    
                </a>';
            }
        }
    }
}

```

```

    ?>
</div>
<!-- CSS Styling -->
<style>
    .sponsors {
        display: flex; /* Χρησιμοποιεί το μοντέλο διάταξης flexbox για ευέλικτη διάταξη των
στοιχείων */
        flex-wrap: wrap; /* Επιτρέπει στις εικόνες να μεταφέρονται
στην επόμενη γραμμή αν χωράνε */
        gap: 20px; /* Προσθέτει 20px κενό μεταξύ των εικόνων */
    }

    .sponsor-image {
        max-width: 400px; /* Περιορίζει το μέγιστο πλάτος των εικόνων
στα 400px */
        height: auto; /* Διατηρεί τις αναλογίες της εικόνας */
        transition: transform 0.3s; /* Προετοιμάζει το στοιχείο για το εφέ hover που
ακολουθεί με επιβράδυνση 0.3 δευτερόλεπτα στο τέλος της κίνησης */
    }

    .sponsor-image:hover {
        transform: scale(1.1); /* Όταν ο χρήστης περάσει το ποντίκι πάνω από την εικόνα,
μεγεθύνεται κατά 10% */
    }
/* Τέλος CSS Styling */
</style>
<!-- Δημιουργία επικεφαλίδας δεύτερης τάξης(h2) -->
<h2 class="contact-header">Contact Information</h2>
<!-- Δημιουργεί κλάση contact-container -->
<div class="contact-container">
    <!-- Δημιουργεί κλάση contact-details -->
    <div class="contact-details">
        <!-- Εικονίδιο φακέλου από τη βιβλιοθήκη Font Awesome -->
        <i class="fa fa-envelope"></i>
        <!-- Εμφανίζει τις πληροφορίες επικοινωνίας από τη
μεταβλητή $contactInfo -->
        <span><?php echo $contactInfo; ?></span>
    </div>
    <!-- Δημιουργεί την κλάση contact-details -->
    <div class="contact-details">
        <!-- Εικονίδιο κατασκευαστή χαρτών από τη βιβλιοθήκη Font Awesome -->
        <i class="fa fa-map-marker"></i>
        <!-- Εμφανίζει αυτό το μήνυμα -->
        <span>You can find us at the Laboratory of Microsystems, Department of Industrial
Engineering & Management, International Hellenic University, Greece 57400</span>

```

```
</div>
<!-- Δημιουργεί την κλάση map-container -->
<div class="map-container">
  <!-- Ενσωματώνει τον χάρτη Google Maps, με πλάτος που γεμίζει τη κλάση του ύψος
  350 pixels και αφαιρεί το περίγραμμα της ετικέτας iframe -->
  <iframe id="map"
    src="https://www.google.com/maps/embed?pb=!1m14!1m8!1m3!1d1513.390870
6703046!2d22.803731837959297!3d40.656741346896084!3m2!1i1024!2i768!4f13.1!3m3!1m
2!1s0x0%3A0xb9b8a179c4d37e55!2sAlexandrio+Technologiko+Ekpedeftiko+Idrima+Thessa
lonikis!5e0!3m2!1sen!2sus!4v1452062855477"
    width="100%" height="350" frameborder="0">
  </iframe>
</div>
</div>
```

Σχήμα 4.32 Το στάδιο της απόδοσης του κώδικα της ενότητας Workshop

5 Δοκιμές και Αποτελέσματα, Προκλήσεις και Λύσεις

Στο παρόν κεφάλαιο παρουσιάζονται οι δοκιμές, που διεξάχθηκαν ώστε να διασφαλιστεί η ορθή λειτουργία του συστήματος και τα αποτελέσματα τους με σκοπό να αποδειχθεί ότι το σύστημα λειτουργεί αξιόπιστα, πληροί τις λειτουργικές και μη λειτουργικές απαιτήσεις που ορίστηκαν στο Κεφάλαιο 3 και να τεκμηριωθούν οι επιδόσεις και οι ικανότητές του. Οι δοκιμές έγιναν σε πρώτο στάδιο με τη βοήθεια του λογισμικού XAMPP και έπειτα σε παραγωγικό διακομιστή(production server) και επικεντρώνονται τόσο στη λειτουργική ορθότητα όσο και σε πτυχές ασφάλειας, απόδοσης και χρηστικότητα. Επιπρόσθετα στο κεφάλαιο αναλύονται οι βασικές προκλήσεις που εμφανίστηκαν κατά την υλοποίηση του συστήματος και κατά την διάρκεια των δοκιμών, καθώς και οι λύσεις που χρησιμοποιήθηκαν για την αντιμετώπισή τους.

5.1 Αποτελέσματα Λειτουργικών Δοκιμών

Παρουσιάζονται οι λειτουργικές δοκιμές που πραγματοποιήθηκαν και τα αποτελέσματα τους

5.1.1 Δοκιμές ανάκτησης και επεξεργασίας δεδομένων

Σενάριο Δοκιμής	Ενέργεια	Αναμενόμενο Αποτέλεσμα	Πραγματικό Αποτέλεσμα	Κατάσταση
Έγκυρο URL αρχείου CSV	Εκτέλεση των script με σωστούς συνδέσμους Dropbox	Τα αρχεία CSV κατεβαίνουν και αποθηκεύονται τοπικά	Επιτυχής λήψη και αποθήκευση	ΕΠΙΤΥΧΙΑ
Μη έγκυρο URL αρχείου CSV	Εκτέλεση των script με λανθασμένους συνδέσμους Dropbox	Εμφάνιση σφάλματος και ορθή διακοπή λειτουργίας	Η συνάρτηση “curl_errno()” εντοπίζει το σφάλμα, το καταγράφει στο αρχείο καταγραφής σφαλμάτων και εμφανίζει το μήνυμα	ΕΠΙΤΥΧΙΑ
CSV με εσφαλμένη δομή	Εισαγωγή αρχείου CSV με λάθος αριθμό στηλών	Το script παραλείπει την εσφαλμένη γραμμή, συνεχίζει κανονικά.	Ο έλεγχος “count(\$headers) == count(\$data)” λειτουργεί, η γραμμή παραλείπεται και το σφάλμα καταγράφεται στο αρχείο	ΕΠΙΤΥΧΙΑ

			καταγραφής σφαλμάτων	
Επεξεργασία σύνθετων συνδέσμων	CSV με εγγραφή “Text URL; Text2 URL2”	Δημιουργία πολλαπλών συνδέσμων που μπορεί να κλικάρει ο χρήστης	Οι συναρτήσεις “explode() και “strpos()” λειτουργούν σωστά, δημιουργώντας ετικέτες <a>	ΕΠΙΤΥΧΙΑ

5.1.2 Δοκιμές δημιουργίας στατικών σελίδων

Σενάριο Δοκιμής	Ενέργεια	Αναμενόμενο Αποτέλεσμα	Πραγματικό Αποτέλεσμα	Κατάσταση
Εκτέλεση της διαδικασίας δημιουργίας στατικών ιστοσελίδων	Εκτέλεση του αρχείου συντονισμού	Δημιουργία/Ανα νέωση όλων των αρχείων HTML στο φάκελο “static/” που φυλάσσονται τα αρχεία	Όλες οι σελίδες HTML δημιουργήθηκαν χωρίς σφάλματα με την πρώτη εκτέλεση του αρχείου συντονισμού. Στην δεύτερη εκτέλεση του αρχείου συντονισμού όλες οι σελίδες HTML ανανεώθηκαν χωρίς σφάλματα	ΕΠΙΤΥΧΙΑ
Πλοήγηση Ιστοσελίδας	Ύπαρξη πλοήγησης σε όλες τις σελίδες HTML	Οι σελίδες εμφανίζονται σωστά χωρίς “χαλασμένους” συνδέσμους	Η πλοήγηση λειτουργεί σωστά, όλοι οι σύνδεσμοι έχουν τις σωστές διευθύνσεις	ΕΠΙΤΥΧΙΑ
Ενημέρωση Περιεχομένου	Αλλαγή δεδομένων στα αρχεία CSV και εκ νέου εκτέλεση του αρχείου συντονισμού	Τα νέα δεδομένα εμφανίζονται στις HTML σελίδες	Οι ιστοσελίδες ανανεώνονται με τα νέα δεδομένα	ΕΠΙΤΥΧΙΑ

5.1.3 Δοκιμές Ασφαλείας

Γίνεται εξομοίωση απόπειρας εμφύτευσής κακόβουλου κώδικα, προσθέτοντας την ακόλουθη γραμμή κώδικα JavaScript (`<script>alert('XSS')</script>`) στα πεδία των CSV.

Αποτέλεσμα: Η γραμμή κώδικα εμφανίζεται ως απλό κείμενο στην ιστοσελίδα, χωρίς να εκτελεστεί. Αποδεικνύοντας ότι η συνάρτηση “`htmlspecialchars()`” εμποδίζει επιτυχώς τις επιθέσεις.

5.2 Αποτελέσματα Μη Λειτουργικών Δοκιμών

Παρουσιάζονται οι μη λειτουργικές δοκιμές που πραγματοποιήθηκαν και τα αποτελέσματά τους

5.2.1 Δοκιμές Απόδοσης

Ως μέρος των δοκιμών απόδοσης του συστήματος μετρήθηκε ο χρόνος, που απαιτείται για την ολοκλήρωση της διαδικασίας δημιουργίας των στατικών σελίδων. Για την απόκτηση των αποτελεσμάτων πραγματοποιήθηκαν 50 εκτελέσεις του αρχείου συντονισμού υπό ίδιες συνθήκες, καταγράφοντας αυτόματα το χρόνο εκτέλεσης.

Για την χρονομέτρηση δημιουργήθηκε ένα βοηθητικό script του οποίου παρατίθεται ο κώδικας:

```
<?php
// performance_test.php
$execution_times = [];
$number_of_runs = 50;

for ($i = 0; $i < $number_of_runs; $i++) {
    // Καταγραφή χρόνου έναρξης
    $start_time = microtime(true);

    // Εκτέλεση του κύριου script
    include 'generate_static.php';

    // Καταγραφή χρόνου λήξης
    $end_time = microtime(true);

    // Υπολογισμός διάρκειας και αποθήκευση
    $execution_time = round(($end_time - $start_time), 2); // Χρόνος σε δευτερόλεπτα (s)
    $execution_times[] = $execution_time;
```

```

// Προαιρετική εκτύπωση προόδου
echo "Run " . ($i+1) . " / $number_of_runs: " . $execution_time . " ms\n";
}

// Αποθήκευση των αποτελεσμάτων σε CSV για ανάλυση
$file = fopen('performance_results.csv', 'w');
fputcsv($file, ['Run', 'Time (ms)']);
foreach ($execution_times as $index => $time) {
    fputcsv($file, [$index + 1, $time]);
}
fclose($file);

// Υπολογισμός και εμφάνιση στατιστικών
$average_time = array_sum($execution_times) / count($execution_times);
$min_time = min($execution_times);
$max_time = max($execution_times);

// Εμφάνιση μηνυμάτων
echo "\n=== Performance Results ===\n";
echo "Average Execution Time: " . round($average_time, 2) . " ms\n";
echo "Minimum Time: " . $min_time . " ms\n";
echo "Maximum Time: " . $max_time . " ms\n";
echo "Standard Deviation: " . round(calculate_std_dev($execution_times), 2) . " ms\n";

// Βοηθητική συνάρτηση για τον υπολογισμό της τυπικής απόκλισης
function calculate_std_dev($array) {
    $variance = 0.0;
    // Υπολογισμός του μέσου όρου
    $average = array_sum($array) / count($array);
    // Υπολογισμός της διακύμανσης
    foreach ($array as $value) {
        // Για κάθε τιμή, βρίσκουμε πόσο διαφέρει από τον μέσο όρο
        // και το υψώνουμε στο τετράγωνο (για να αφαιρέσουμε αρνητικά πρόσημα)
        $variance += pow(($value - $average), 2);
    }
    // Βήμα 3: Βρίσκουμε τη μέση τιμή των τετραγώνων των διαφορών
    // και παίρνουμε την τετραγωνική ρίζα για να πάρουμε την τυπική απόκλιση
    return sqrt($variance / count($array));
}
?>

```

Σχήμα 5.1 Ο πλήρης κώδικας του βοηθητικού script

Μετά από τις 50 εκτελέσεις τα αποτελέσματα ήταν τα εξής:

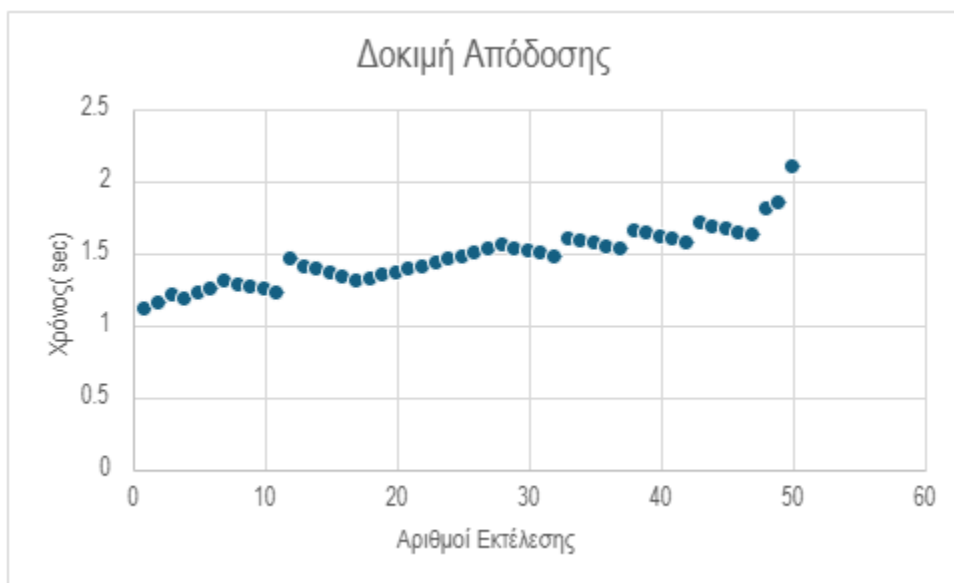
Μέσος Χρόνος Εκτέλεση: 1.45 δευτερόλεπτα

Ελάχιστος Χρόνος Εκτέλεσης: 1.10 δευτερόλεπτα

Μέγιστος Χρόνος Εκτέλεσης: 2.10 δευτερόλεπτα

Τυπική Απόκλιση: 0.25 δευτερόλεπτα

Παρακάτω παρατίθεται και γράφημα με την οπτική αναπαράσταση των αποτελεσμάτων:



Σχήμα 5.2 Διάγραμμα με τα αποτελέσματα των μετρήσεων

Το γράφημα και η μικρή τυπική απόκλιση των 0.25 δευτερολέπτων αποδεικνύουν την σταθερότητα του συστήματος ως προς την απόδοση του. Επίσης παρατηρείται ότι οι πρώτες εκτελέσεις του συστήματος είναι ελαφρώς πιο γρήγορες πιθανόν λόγω της αποθήκευσης των αρχείων στην προσωρινή μνήμη του διακομιστή με την απόδοση να σταθεροποιείται μετά τις πρώτες 10-15 εκτελέσεις.

5.2.2 Δοκιμές Ανθεκτικότητας

1.Δοκιμές αποτυχίας δικτύου

Στα πλαίσια των δοκιμών ανθεκτικότητας του συστήματος, έγιναν αρχικά δοκιμές αποτυχίας δικτύου. Ως μέρος των δοκιμών οι κώδικες των αρχείων παραγωγής περιεχομένου τροποποιήθηκαν προσωρινά με ένα λανθασμένο URL για τα αρχεία που λαμβάνουν από το Dropbox. Συγκεκριμένα η αρχή των αρχείων παραγωγής περιεχομένου μετατράπηκε σε:

```
// Σύνδεσμος Dropbox για το CSV
```

```
$csvUrl = 'https://www.dropbox.com/invalid-url-test-123454321.csv; // Λανθασμένο URL;
```

```
// Τοπικό path για να αποθηκεύσει το ανανεωμένο CSV
```

```

$localCsvFile = 'curprojects.csv';

// Κατέβασε το ανανεωμένο CSV από το Dropbox
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $csvUrl);
curl_setopt($ch, CURLOPT_TIMEOUT, 10); // Ορίζεται χρονικό όριο σύνδεσης 10
δευτερόλεπτα
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
$data = curl_exec($ch);

//Έλεγχος σφαλμάτων και τερματισμός της συνεδρίας αν υπάρχουν
if (curl_errno($ch)) {
    // Καταγραφή σφάλματος στο αρχείο καταγραφής σφαλμάτων
    error_log('Curl error: ' . curl_error($ch));
    // Μήνυμα σφάλματος
    echo 'Error: Unable to fetch the CSV file.';
    // Τερματισμός διαδικασίας
    exit;
}

//Κλείνει την cURL συνεδρία
curl_close($ch);

```

Σχήμα 5.3 Το αλλαγμένου τμήμα του κώδικα στο πλαίσιο των δοκιμών αποτυχίας δικτύου

Το αποτέλεσμα ήταν η ανίχνευση του σφάλματος μέσω της “cURL”, η καταγραφή του στο αρχείο σφαλμάτων, ο τερματισμός της εκτέλεσης του αρχείου και η εμφάνιση του μηνύματος σφάλματος στο χρήστη χωρίς να δημιουργηθούν ημιτελή HTML αρχεία. Ίδιο ήταν και το αποτέλεσμα όταν ορίστηκε μικρό χρονικό όριο σύνδεσης της τάξεως του 1 δευτερολέπτου με την προσθήκη των συναρτήσεων “curl_setopt(\$ch, CURLOPT_TIMEOUT, 1);” και “curl_setopt(\$ch, CURLOPT_CONNECTTIMEOUT, 1);” στους κώδικες των αρχείων παραγωγής περιεχομένου για να γίνει προσομοίωση αργής απόκρισης από τον διακομιστή.

2.Δοκιμές δικαιωμάτων αρχείων

Στη συνέχεια των δοκιμών ανθεκτικότητας γίνονται δοκιμές δικαιωμάτων αρχείων, στο πλαίσιο των οποίων αναιρούνται δικαιώματα από αρχεία της διαδικασίας πριν την εκτέλεση της.

Στην πρώτη δοκιμή μετατράπηκαν τα αρχεία CSV σε αρχεία μόνο για ανάγνωση(read-only) και το σύστημα ανταποκρίθηκε σωστά εμφανίζοντας και το αναμενόμενο μήνυμα “Error: Unable to save the CSV file.” βάσει των τροποποιημένων δικαιωμάτων. Από κάτω παρατίθεται το τμήμα του κώδικα υπεύθυνο για την εύρεση του σφάλματος:

```

// Αποθηκεύει τα κατεβασμένα δεδομένα στον τοπικό φάκελο

```

```

if (file_put_contents($localCsvFile, $data) === false) {
    // Αν αποτύχει καταγράφει το σφάλμα στο αρχείο καταγραφής σφαλμάτων
    error_log('Failed to save the downloaded CSV to local storage.');
```

// Μήνυμα σφάλματος
echo 'Error: Unable to save the CSV file.';
// Τερματισμός διαδικασίας
exit;

Σχήμα 5.3 Το τμήμα του κώδικα υπεύθυνο για την εύρεση του σφάλματος αν τα αρχεία CSV είναι μόνο για ανάγνωση

Στην δεύτερη δοκιμή αφαιρέθηκαν και τα δικαιώματα ανάγνωσης και το σύστημα και το σύστημα πάλι ανταποκρίθηκε σωστά εμφανίζοντας το αναμενόμενο μήνυμα “Error: Unable to open the local CSV file.” βάσει των τροποποιημένων δικαιωμάτων. Από κάτω παρατίθεται το τμήμα του κώδικα υπεύθυνο για την εύρεση του σφάλματος:

```

//Ανάγνωση και επεξεργασία του CSV
// Άνοιξε το φάκελο CSV για να το διαβάσεις
if (($handle = fopen($localCsvFile, 'r')) !== false) {
    //Επεξεργασία δεδομένων ...
} else {
    // Εμφανίζει μήνυμα σφάλματος
    echo "Error: Unable to open the local CSV file.";
    exit;
}

```

Σχήμα 5.4 Το τμήμα του κώδικα υπεύθυνο για την εύρεση του σφάλματος αν τα αρχεία CSV δεν έχουν κανένα δικαίωμα

5.2.3 Δοκιμές Χρηστικότητας και Αποκριτικότητας

Τελευταίες δοκιμές στο πλαίσιο των μη λειτουργικών δοκιμών είναι οι δοκιμές χρηστικότητας και αποκριτικότητας, που πραγματοποιήθηκαν μέσω χειροκίνητης δοκιμής των ιστοσελίδων σε διαφορετικά μεγέθη οθονών (Desktop, Tablet, Κινητή Συσκευή) και σε διάφορους περιηγητές ιστού (Google Chrome, Firefox, Microsoft Edge, Safari, Opera). Μέσω των δοκιμών επιβεβαιώθηκε, ότι οι ιστοσελίδες προσαρμόζονται σωστά σε όλα τα μεγέθη οθονών χάρις το σύστημα πλέγματος του Bootstrap, η πλοήγηση λειτουργεί σωστά στις κινητές συσκευές χάρις το μενού “hamburger” και ότι η εμφάνιση και η λειτουργικότητα των ιστοσελίδων παραμένει ίδια σε όλους τους δοκιμασμένους περιηγητές ιστού.

5.3 Προκλήσεις και Λύσεις

Κατά τη διάρκεια της ανάπτυξης αλλά και των δοκιμών του συστήματος, αντιμετωπίστηκαν διάφορες προκλήσεις που απαιτούσαν αναθεώρηση του κώδικα και εφαρμογή εναλλακτικών λύσεων. Η ανάλυση αυτών των προκλήσεων και των λύσεων τους παρουσιάζει την εξελικτική πορεία της εργασίας και τονίζει τις σημαντικές αποφάσεις που πάρθηκαν γύρω από τον σχεδιασμό της.

5.3.1 Ενσωμάτωση Dropbox και πρόσβαση σε αρχεία

Το σύστημα στηρίζεται σε μεγάλο βαθμό στη λήψη των αρχείων CSV από το Dropbox. Οι πρώτες εκδοχές του κώδικα του συστήματος χρησιμοποιούσαν την συνάρτηση “file_get_contents()”, λόγω της απλότητας και της ευχρηστίας της, για την λήψη των αρχείων CSV, ωστόσο οι ανακατευθύνσεις που χρησιμοποιεί το Dropbox ως μέτρο ασφαλείας των αρχείων του είχαν ως αποτέλεσμα την επιστροφή κενών αποτελεσμάτων ή και πολλές φορές την αποτυχία της λήψεως των αρχείων. Συνεπώς υιοθετήθηκε η χρήση της βιβλιοθήκης cURL, η οποία παρείχε καλύτερη αναφορά σφαλμάτων και μπορούσε να διαχειριστεί τις ανακατευθύνσεις χάρις την παράμετρο “CURLOPT_FOLLOWLOCATION”. Επιπρόσθετα το σύστημα αποθηκεύει τοπικά ένα αντίγραφο του CSV, κάθε φορά που πραγματοποιεί την διαδικασία δημιουργίας των σελίδων, έτσι ώστε να μπορεί να λειτουργεί ακόμη και όταν οι υπηρεσίες του Dropbox είναι προσωρινά μη διαθέσιμες.

5.3.2 Δημιουργία Προτύπου χωρίς πλαίσιο ανάπτυξης

Σε αντίθεση με σύγχρονα πλαίσια ανάπτυξης που διαθέτουν μηχανές προτύπων, το παρόν σύστημα υλοποιήθηκε μόνο με χρήση της γλώσσας PHP. Αυτό είχε ως αποτέλεσμα να επαναλαμβάνεται κώδικας για τις επικεφαλίδες, την πλοήγηση, την εμφάνιση και τα υποσέλιδα, κάνοντας τον κώδικα πιο δυσνόητο και πιο δύσκολο στην συντήρηση του. Η λύση ήταν η δημιουργία του αρχείου καθορισμού προτύπου, κρατώντας τα επαναλαμβανόμενα τμήματα του κώδικα σε ένα μόνο αρχείο, διατηρώντας την ομοιόμορφη εμφάνιση των σελίδων και την απλότητα στην διαχείριση και την συντήρηση του συστήματος.

5.3.3 Απόδοση HTML μέσα από PHP Scripts

Η λύση της προηγούμενης πρόκλησης δημιούργησε ταυτόχρονα μία άλλη, καθώς πλέον τα αρχεία παραγωγής περιεχομένου έπρεπε να παράγουν “καθαρό” HTML κώδικα, ο οποίος να μπορεί να ενσωματωθεί στο αρχείο προκαθορισμένου προτύπου. Αυτή η ανάμειξη μεταξύ μεγάλων τμημάτων HTML και PHP κώδικα εντός των αρχείων παραγωγής περιεχομένου οδηγούσε δυσανάγνωστο και δύσκολο στη συντήρηση κώδικα. Συνεπώς πάρθηκε απόφαση να

γίνει εναλλαγή ανάμεσα σε λειτουργία PHP και έξοδο HTML, χρησιμοποιώντας τη γλώσσα PHP για τη λογική(βρόγχους, συνθήκες) και τη γλώσσα HTML για τη δομή, επιτυγχάνοντας μια δόμηση όπου το HTML και η λογική PHP συνυπάρχουν χωρίς να μπερδεύονται , βελτιώνοντας σημαντικά την συντηρησιμότητα και επεκτασιμότητα του συστήματος σε βάθος χρόνου.

Παραθέτω το πριν και το μετά ενός αποσπάσματος του κώδικα του αρχείου παραγωγής περιεχομένου της ενότητας Announcements:

ΠΡΙΝ(Ένας αλληπάλληλος καταγισμός από συναρτήσεις echo)

```
<?php foreach ($announcements as $announcement): ?>
    <?php
    // Ξεκινάει ένα τεράστιο echo που περιέχει ΟΛΟ τη συμβολοσειρά HTML
    echo "<div class='announcement-item' style='margin-bottom: 40px;'>";
    echo "<div class='announcement-content' style='display: flex; align-items: flex-start; gap:
30px; padding: 20px; border: 1px solid #ddd; border-radius: 10px; box-shadow: 2px 2px
10px rgba(0,0,0,0.1); background: #fff;'>";

    // 1. ΕΛΕΓΧΟΣ για εικόνα ΜΕΣΑ στη συμβολοσειρά
    if (!empty($announcement['Image'])) {
        echo "<img src='images/' . $announcement['Image'] . \" alt=\" . $announcement['Title'] .
\" style='width: 150px; height: 150px; object-fit: cover; border-radius: 10px; flex-shrink:
0;'>";
    }

    // 2. Το κύριο περιεχόμενο
    echo "<div style='flex: 1;'>";
    echo "<div class='announcement-header' style='margin-bottom: 15px;'>";
    echo "<span class='announcement-date' style='color: #666; font-size: 14px;'>\" .
$announcement['Date and Time'] . "</span>";
    echo "<h3 class='announcement-title' style='margin: 5px 0 10px 0; font-size: 20px; color:
#333;'>\" . $announcement['Title'] . "</h3>";
    echo "</div>"; // κλείνει το announcement-header

    echo "<div class='announcement-text' style='font-size: 16px; color: #555; line-height:
1.5;'>\" . $announcement['Announcement'] . "</div>";

    // 3. ΕΛΕΓΧΟΣ για links - Ακόμα πιο πολύπλοκο!
    if (!empty($announcement['ProcessedLinks'])) {
        echo "<div class='announcement-links' style='margin-top: 15px;'>";
        echo "<p style='font-weight: bold; margin-bottom: 8px;'>Related Links:</p>";
        echo "<ul style='margin: 0; padding-left: 20px;'>";

        foreach ($announcement['ProcessedLinks'] as $link) {
            echo "<li style='margin-bottom: 5px;'>";
```

```

        echo "<a href='\"' . $link['url'] . '\" target='_blank' style='color: #007bff; text-
decoration: none;'>\" . $link['text'] . "</a>";
        echo "</li>";
    }

    echo "</ul>";
    echo "</div>"; // κλείνει το links
}

echo "</div>"; // κλείνει το flex: 1;
echo "</div>"; // κλείνει το announcement-content
echo "</div>"; // κλείνει το announcement-item
?>
<?php endforeach; ?>

```

Σχήμα 5.5 Τμήμα του κώδικα του αρχείου παραγωγής περιεχομένου της ενότητας
Announcements με βάση τον μη δομημένο τρόπο γραφής του κώδικα

Χρησιμοποιώντας αυτού του είδους τη προσέγγιση έχει ως αποτέλεσμα:

- Να χάνεται η ιεραρχία της δομής, αφού ο χρήστης πρέπει κυριολεκτικά να μετρήσει τα “echo “</div>”” για να βρει ποιο κλείνει ποιο.
- Μίξη της λογικής και της εξόδου, εφόσον οι δομές ελέγχου “if” και οι βρόχοι “foreach” βρίσκονται “κρυμμένοι” μέσα στις συμβολοσειρές των συναρτήσεων “echo”
- Δυσκολία στην αλλαγή και αναβάθμιση, καθώς για οποιαδήποτε αλλαγή στο στυλ ή αναβάθμιση στον κώδικα θα χρειαστεί να βρεις το σωστό σημείο μέσα στην συμβολοσειρά
- Μπορούν να γίνουν πιο εύκολα λάθη στη συγγραφή του κώδικα, τα οποία θα είναι και πιο δύσκολο να βρεθούν και να διορθωθούν, λόγω της δυσνόητης φύσης του κώδικα.

META(Η δομημένη προσέγγιση)

```

<?php foreach ($announcements as $announcement): ?>
    <!-- 1. KYPIO CONTAINER -->
    <div class="announcement-item" style="margin-bottom: 40px;">
        <div class="announcement-content" style="display: flex; align-items: flex-start; gap:
30px; padding: 20px; border: 1px solid #ddd; border-radius: 10px; box-shadow: 2px 2px
10px rgba(0,0,0,0.1); background: #fff;">
            <!-- 2. EIKONA -->
            <?php if (!empty($announcement['Image'])): ?>
                " style="width: 150px;
height: 150px; object-fit: cover; border-radius: 10px; flex-shrink: 0;">
            <?php endif; ?>
        <!-- ΤΕΛΟΣ ΕΙΚΟΝΑΣ -->
    </div>
</div>

```

```

<!-- 3. ΠΕΡΙΕΧΟΜΕΝΟ -->
<div style="flex: 1;">
  <!-- 3.1 Επικεφαλίδα-->
  <div class="announcement-header" style="margin-bottom: 15px;">
    <span class="announcement-date" style="color: #666; font-size: 14px;">
      <?php echo htmlspecialchars($announcement['Date and Time']); ?>
    </span>
    <h3 class="announcement-title" style="margin: 5px 0 10px 0; font-size: 20px;
color: #333;">
      <?php echo htmlspecialchars($announcement['Title']); ?>
    </h3>
  </div>
  <!-- ΤΕΛΟΣ Επικεφαλίδας -->
  <!-- 3.2 ΚΕΙΜΕΝΟ ΑΝΑΚΟΙΝΩΣΗΣ -->
  <div class="announcement-text" style="font-size: 16px; color: #555; line-height:
1.5;">
    <?php echo $announcement['Announcement']; ?>
  </div>

  <!-- 3.3 LINKS -->
  <?php if (!empty($announcement['ProcessedLinks'])): ?>
    <div class="announcement-links" style="margin-top: 15px;">
      <p style="font-weight: bold; margin-bottom: 8px;">Related Links:</p>
      <ul style="margin: 0; padding-left: 20px;">
        <?php foreach ($announcement['ProcessedLinks'] as $link): ?>
          <li style="margin-bottom: 5px;">
            <a href="<?php echo htmlspecialchars($link['url']); ?>"
target="_blank" style="color: #007bff; text-decoration: none;">
              <?php echo htmlspecialchars($link['text']); ?>
            </a>
          </li>
        <?php endforeach; ?>
      </ul>
    </div>
  <?php endif; ?>
  <!-- ΤΕΛΟΣ LINKS -->
</div>
<!-- ΤΕΛΟΣ ΠΕΡΙΕΧΟΜΕΝΟΥ -->
</div>
<!-- ΤΕΛΟΣ ΑΝΑΚΟΙΝΩΣΗΣ -->
<?php endforeach; ?>

```

Σχήμα 5.6 Το ίδιο τμήμα κώδικα με το Σχήμα 5.5 αλλά με χρήση της νέας δομημένης προσέγγισης

Αυτή η προσέγγιση παρέχει:

- Πολύ καθαρή ιεραρχία, καθώς ο χρήστης μπορεί να δει απευθείας ότι το “announcement-item” περιέχει το “announcement-content”, που περιέχει την εικόνα και ένα τμήμα κώδικα “div” που περιέχει το περιεχόμενο. Επίσης οι ετικέτες ανοίγουν και κλείνουν σε ευθυγράμμιση, κάνοντας τον κώδικα πιο ευανάγνωστο.
- Η ευκρίνεια του κώδικα προσφέρει ευκολότερη αποσφαλμάτωση (debugging), αφού το οποιοδήποτε λάθος είναι πιο εύκολο να βρεθεί.
- Επίσης πιο εύκολη γίνεται και η επεξεργασία του κώδικα, διότι οποιαδήποτε αλλαγή στο στυλ των στοιχείων δεν επηρεάζει την PHP λογική.

6 Συμπεράσματα και Προτάσεις Αναβάθμισης

Στο παρόν κεφάλαιο παρουσιάζεται μια συνολική αξιολόγηση της εργασίας που πραγματοποιήθηκε, συνοψίζοντας πρώτα τα κύρια επιτεύγματα και τη συνεισφορά της και προτείνοντας, στο τέλος συγκεκριμένες κατευθύνσεις για μελλοντική βελτίωση και επέκταση της, ώστε να ενισχυθεί περαιτέρω η λειτουργικότητα και η αξία του εργαλείου που αναπτύχθηκε.

6.1 Συνολική Σύνοψη

Η παρούσα διπλωματική εργασία ανέπτυξε με επιτυχία ένα πλήρως αυτοματοποιημένο σύστημα διαχείρισης περιεχομένου για την ιστοσελίδα του Εργαστηρίου Μικροσυστημάτων, το οποίο υλοποιήθηκε με χρήση της αρχιτεκτονικής της δυναμικής δημιουργίας στατικών σελίδων, δημιουργώντας ένα ελαφρύ και αποδοτικό σύστημα, που παράγει αυτόματα στατικά αρχεία χωρίς να βασίζεται σε πολύπλοκα συστήματα δημιουργίας περιεχομένου όπως το Wordpress, το Joomla κ.α. Αυτού του τύπου η αρχιτεκτονική εξασφάλισε εξαιρετικές ταχύτητες φόρτωσης των σελίδων, ενισχυμένη ασφάλεια και ευκολότερη φιλοξενία του συστήματος από τους διακομιστές του πανεπιστημιακού ιδρύματος. Η επιλογή εργαλείων, όπως η γλώσσα προγραμματισμού PHP και το πλαίσιο ανάπτυξης Bootstrap, για την υλοποίηση του συστήματος, αποδεικνύουν ότι ακόμη και με τεχνολογίες που είναι καθιερωμένες εδώ και χρόνια στον χώρο της πληροφορικής, μπορούν να παραχθούν σύγχρονες και λειτουργικές εφαρμογές έτοιμες να αντιμετωπίσουν προβλήματα του σήμερα. Το σύστημα διαχωρίζει αποτελεσματικά την εμφάνιση και το περιεχόμενο των σελίδων, επιτρέποντας στον διαχειριστή τους να ενημερώνει τα δεδομένα τους μέσω των αρχείων CSV, χωρίς ο ίδιος να χρειαστεί να έχει γνώσεις προγραμματισμού ή να αγγίζει τον κώδικα. Επιπρόσθετα η ευέλικτη δομή του συστήματος, που χαρακτηρίζεται από την αυτονομία των ενοτήτων του ιστοτόπου, κάνει την προσθήκη νέων λειτουργιών και την μελλοντική επέκταση του συστήματος ευκολότερη, ώστε να συνεχίσει να ανταποκρίνεται με επιτυχία σε οποιεσδήποτε μελλοντικές ανάγκες προκύψουν. Συμπερασματικά το τελικό προϊόν δεν είναι απλά μια θεωρητική μελέτη, αλλά μια πλήρως λειτουργική πλατφόρμα, έτοιμη για παραγωγική χρήση, η οποία είναι εύκολη στη διαχείριση, την συντήρηση και την επέκταση της.

6.2 Η Υβριδική Αρχιτεκτονική του Συστήματος

Το κύριο επίτευγμα αυτής της εργασίας είναι ο σχεδιασμός και η υλοποίηση ενός συστήματος που δημιουργεί δυναμικά στατικές ιστοσελίδες. Ο παρακάτω πίνακας αναλύει τα δυναμικά και στατικά στοιχεία του συστήματος για μεγαλύτερη σαφήνεια:

Στοιχείο Συστήματος	Τύπος	Περιγραφή
Διαδικασία Δημιουργίας (Build)	Δυναμικό	Η διαδικασία δημιουργίας των σελίδων, όπου το αρχείο

		συντονισμού εκτελείται, τραβάει τα δεδομένα από τα αρχεία CSV, τα επεξεργάζεται και δημιουργεί τις σελίδες
Πηγές Δεδομένων(CSV)	Δυναμικό	Τα αρχεία CSV στο Dropbox(το σύστημα αντλεί δυναμικά τα δεδομένα από αυτά)
Επεξεργασία και Εμπλουτισμός Δεδομένων	Δυναμικό	Η λογική που χρησιμοποιείται σε πολλές από τις σελίδες όπως, η ταξινόμηση των ανακοινώσεων, η διάσπαση των συνδέσμων σε Κείμενο URL, η ομαδοποίηση των εργασιών των φοιτητών στην ενότητα Completed Projects ανά έτος κ.α.
Τελικές HTML σελίδες	Στατικό	Τα αποτελέσματα της διαδικασίας δημιουργίας, τα οποία είναι HTML αρχεία που μπορούν να εξυπηρετηθούν απευθείας από τον διακομιστή (server) χωρίς να χρειαστούν κάποια άλλη επεξεργασία
Πρότυπο(Template)	Στατικό	Η δομή και η εμφάνιση των σελίδων
Λειτουργίες Πλευράς Χρήστη(Client-Side)	Στατικό	Οποιαδήποτε αλληλεπίδραση γίνεται μετά την φόρτωση της σελίδας στον εξυπηρετητή ιστού(browser) του χρήστη(π.χ. η απόκριση του Bootstrap menu)

6.3 Προτάσεις Βελτίωσης

Ακολουθούν προτάσεις για την περαιτέρω βελτίωση του συστήματος:

1.Αυτόματη Ενημέρωση

Μια πρόταση για την βελτίωση του συστήματος είναι η αυτοματοποίηση της διαδικασίας ενημέρωσης είτε με την χρήση ενός webhook, είτε με την χρήση ενός cron job. Το webhook είναι ένας μηχανισμός ειδοποίησης, που επιτρέπει σε μία εφαρμογή να ενημερώνει αυτόματα μία άλλη εφαρμογή όταν συμβαίνει ένα γεγονός. Στην πράξη όταν συμβαίνει το γεγονός που έχουμε ορίσει(π.χ. αλλαγή ενός CSV αρχείου στο Dropbox), η πλατφόρμα(Dropbox) στέλνει το

αίτημα(webhook) σε μία διεύθυνση URL που έχουμε ορίσει. Με την σειρά του αυτό το αίτημα ενεργοποιεί ένα αρχείο στο διακομιστή μας, το οποίο με την σειρά του θα εκτελεί το αρχείο συντονισμού. Αυτή η πρώτη μέθοδος αν και πιο σύνθετη από τη δεύτερη, θα επέτρεπε την αυτόματη ανανέωση του ιστοτόπου μέσα σε λίγα δευτερόλεπτα από την αλλαγή των αρχείων CSV, διατρέχοντας βέβαια κάποιους κινδύνους όσων αφορά την ασφάλεια και τα μέτρα που θα έπρεπε να παρθούν για να διασφαλιστεί. Η δεύτερη μέθοδος αφορά την χρήση cron job, που ουσιαστικά προγραμματίζει την εκτέλεση του αρχείου συντονισμού σε καθορισμένα χρονικά διαστήματα(π.χ. κάθε μία ώρα). Αυτή η μέθοδος είναι πιο απλή στην ρύθμιση της και πιο ασφαλής αφού δεν εξαρτάται από εξωτερικές εφαρμογές όπως το Dropbox, αλλά δεν ανανεώνει άμεσα τα δεδομένα όπως η προηγούμενη, αφού το αρχείο εκτελείται σε προγραμματισμένα διαστήματα και το αρχείο συντονισμού εκτελείται τακτικά ακόμα και αν δεν υπάρχουν αλλαγές στα δεδομένα, το οποίο οδηγεί στην άσκοπη σπατάλη πόρων. Συμπερασματικά και οι δύο μέθοδοι έχουν τα υπέρ και τα κατά τους και θα μπορούσαν και οι δύο να βελτιώσουν περαιτέρω το σύστημα.

2.Δημιουργία ροής Really Simple Syndication (RSS feed)

Η πρόταση αυτή αφορά την δημιουργία ροής RSS για τις ανακοινώσεις, όπου κατά την διάρκεια δημιουργίας των σελίδων θα δημιουργείται αυτόματα ένα αρχείο eXtensible Markup Language(XML) στον κατάλογο ρίζας(root directory) του ιστότοπου, το οποίο θα περιέχει τις πιο πρόσφατες ανακοινώσεις σε μορφή XML. Έπειτα οι χρήστες θα μπορούν να “εγγραφούν” στη ροή, χρησιμοποιώντας έναν αναγνώστη RSS, ο οποίος θα ελέγχει αυτόματα για νέες ανακοινώσεις και θα ειδοποιεί τον χρήστη, χωρίς αυτός να χρειάζεται να επισκέπτεται συνεχώς την ιστοσελίδα. Με αυτόν τον τρόπο θα ειδοποιούνται άμεσα για οποιαδήποτε νέα ανακοίνωση διασφαλίζοντας την άμεση ενημέρωσή τους με τον ελάχιστο δυνατό “κόπο” από μεριάς τους.

3.Ενσωμάτωση εσωτερικού μηχανισμού αναζήτησης

Αυτή η πρόταση έχει ως στόχο την δημιουργία εσωτερικού μηχανισμού αναζήτησης, που θα επιτρέπει εύρεση πληροφοριών μέσω λέξεων-κλειδιών. Η πρόταση αυτή θα βασιζόταν πλήρως στη δημιουργία ενός ευρετηρίου αναζήτησης κατά την διάρκεια δημιουργίας των σελίδων, το οποίο σύλλεγε και θα ενοποιούσε τα δεδομένα από όλα τα αρχεία CSV και θα τα αποθήκευε σε μορφή JSON. Έπειτα με την χρήση ειδικής βιβλιοθήκης θα φόρτωνε το ευρετήριο και θα γινόταν εκτέλεση των ερωτημάτων απευθείας στο πρόγραμμα περιήγησης και θα παρουσιαζόντουσαν τα αποτελέσματα σε πραγματικό χρόνο, χωρίς να απαιτείται επικοινωνία με τον διακομιστή. Λόγω της αρχιτεκτονικής του συστήματος, αυτή η προσέγγιση αν και περίπλοκη θα διατηρούσε τα πλεονεκτήματα της στατικής αρχιτεκτονικής των σελίδων και θα βελτίωνε παράλληλα την εμπειρία του χρήστη.

7 Βιβλιογραφία

- [1] Open Knowledge Foundation. Retrieved From <https://okfn.org/>
- [2] The PHP Group. PHP: Hypertext Preprocessor Manual. Retrieved From <https://www.php.net/manual/en/>
- [3] MDN Web Docs.HTML: HyperText Markup Language. Retrieved From <https://developer.mozilla.org/en-US/docs/Web/HTML>
- [4] Bootstrap Team. Bootstrap 3.3.6 Documentation. Retrieved From <https://getbootstrap.com/docs/3.3/>
- [5] jQuery Foundation. jQuery Official Documentation. Retrieved From <https://jquery.com/>
- [6] MDN Web Blogs. JavaScript language overview. Retrieved from https://developer.mozilla.org/en-US/docs/Web/JavaScript/Language_overview
- [7] IFTT(2024). What is Dropbox: A beginner's guide. Retrieved From <https://ifttt.com/explore/what-is-dropbox>
- [8] Dropbox Inc. How to force a shared link to download or render. Retrieved From <https://help.dropbox.com/share/force-download>
- [9] WinSCP.net. Introducing WinSCP. Retrieved From <https://winscp.net/eng/docs/introduction>
- [10] Apache Friends. XAMPP Hosting. Retrieved From <https://www.apachefriends.org/hosting.html>
- [11] Visual Studio Code. Visual Studio Code documentation. Retrieved From <https://code.visualstudio.com/docs>
- [12] Anne Bonner. What is a CSV file: A comprehensive guide. Retrieved From <https://flatfile.com/blog/what-is-a-csv-file-guide-to-uses-and-benefits/>
- [13] LTS Group(2023). Static vs Dynamic Website: The Key Differences and Which To Use. Retrieved From <https://ltsgroup.tech/blog/static-vs-dynamic-website/>
- [14] Tobias Mildenerberger(2024). Static Websites vs. Dynamic Websites: A Comparison Retrieved From <https://contabo.com/blog/static-websites-vs-dynamic-websites-a-comparison/>
- [15] Roger S.Pressman. Software Engineering. A Practitioner’s Approach.