

ΤΕΧΝΟΛΟΓΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΗΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πτυχιακή εργασία

ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΗΣ
ΓΙΑ ΤΟ ΤΕΧΝΟΛΟΓΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΚΥΠΡΟΥ

Παναγιώτης Γεωργιάδης

Σύμβουλος καθηγητής
Κος Χρίστος Μακαρούνας

Λεμεσός 2016

Πνευματικά δικαιώματα

Copyright © Παναγιώτης Γεωργιάδης, 2016

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του Τεχνολογικού Πανεπιστημίου Κύπρου δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον καθηγητή μου Κο. Χρίστο Μακαρούνα που με εμπιστεύτηκε για την υλοποίηση αυτής της πτυχιακής εργασίας που στάθηκε δίπλα μου και με βοήθησε με κάθε δυνατό τρόπο. Ιδιαίτερες ευχαριστίες θα ήθελα επίσης να απευθύνω στην μητέρα μου και στον πατέρα μου για την στήριξη που μου πρόσφεραν όλα αυτά τα χρόνια των σπουδών μου και συνέβαλαν στα μέγιστα έτσι ώστε να φτάσω μέχρι εδώ.

ΠΕΡΙΛΗΨΗ

Ο σκοπός αυτής της πτυχιακής εργασίας είναι η δημιουργία μιας εφαρμογής για το Τεχνολογικό Πανεπιστήμιο Κύπρου, η οποία θα προσφέρει πληροφορίες που αφορούν το πανεπιστήμιο σε νεοεισερχόμενους φοιτητές αλλά και σε τελειόφοιτους μαθητές Λυκείων και Τεχνικών σχολών.

Ο στόχος της εφαρμογής αυτής είναι να πείσει κυρίως τους τελειόφοιτους μαθητές να επιλέξουν το Τεπακ ως το μέρος για την συνέχιση της ακαδημαϊκής τους καριέρας. Αυτό θα γίνεται μέσω της παρουσίασης βασικών πληροφοριών που αφορούν το πανεπιστήμιο, τα διάφορα τμήματα του και τις διάφορες του υπηρεσίες (εστιατόριο, γυμναστήριο κλπ.). Επίσης μέσω αυτής της εφαρμογής ο χρήστης θα μπορεί να δει διάφορες φωτογραφίες κτηρίων, αιθουσών και εργαστηρίων του πανεπιστημίου, αλλά και χάρτες που υποδεικνύουν τις τοποθεσίες των διαφόρων αυτών κτηρίων.

Η εφαρμογή αποφασίστηκε να υλοποιηθεί σε μορφή ιστοσελίδας, καθώς στην σύγχρονη εποχή η χρήση των ιστοσελίδων αλλά και γενικά του διαδικτύου αποτελούν μια πολύ αποτελεσματική μορφή παρουσίασης δεδομένων και πληροφοριών. Μελετήθηκαν σύγχρονοι τρόποι σχεδιασμού μιας ιστοσελίδας και το πως μπορεί η περιήγηση σε αυτή και η αναζήτηση των πληροφοριών να γίνεται με μια διαδικασία εύκολη, απλή και γρήγορη για τον χρήστη.

Για την υλοποίηση αυτής της ιστοσελίδας χρησιμοποιήθηκαν οι γλώσσες προγραμματισμού και τεχνολογίες ιστού HTML5, CSS3, JavaScript, το περιβάλλον εργασίας Bootstrap, αλλά και το εργαλείο κατασκευής λογισμικών NetBeans.

Λέξεις κλειδιά: Τεχνολογικό Πανεπιστήμιο Κύπρου, ιστοσελίδα, διαδίκτυο, τεχνολογίες ιστού, σχεδίαση.

ABSTRACT

The purpose of this final year project is to create an application for Cyprus University of Technology that will provide information about the university to new students or to upcoming graduates of high schools.

The purpose of this application is to persuade students to choose Cyprus University of Technology as the place to continue their studies. This will be happening by presenting basic information about the university and information about its departments and other services that it offers (restaurant, gym etc). Also, with this application the user will be able to see photos of the buildings, the lecture rooms and the laboratories of the university along with maps with the location of them.

For this application, it was decided to be created as a website because today, websites and internet in general are offering great advantages and is a very effective method of presenting information and data around the world.

Many modern practices of web design have been studied, so that the navigation through the website will be an easy, fast and friendly procedure and overall, how to improve the user experience.

For the implementation of this website, we used web development technologies like HTML5, CSS3, JavaScript, Bootstrap and the software development platform Netbeans.

Keywords: Cyprus University of Technology, website, internet, web development technologies, web design

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ.....	iv
ABSTRACT	v
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	vi
ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ	xi
ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ.....	xiv
ΑΠΟΔΟΣΗ ΟΡΩΝ.....	xv
ΕΙΣΑΓΩΓΗ.....	xvi
1 Διαδίκτυο και βασικές έννοιες	1
1.1 Διαδίκτυο	1
1.1.1 Εισαγωγή	1
1.1.2 Ορισμός	1
1.2 Παγκόσμιος Ιστός.....	1
1.3 Ιστοσελίδες.....	2
1.3.1 Εισαγωγή	2
1.3.2 Web browser.....	2
1.3.3 Κατασκευή Ιστοσελίδων.	2
1.3.3.1 Διαδικασία.....	2
1.3.3.2 Τρόποι κατασκευής.....	3
1.3.4 Σημασία και χρησιμότητα	3
2 Τεχνολογίες Ιστού – Γλώσσες προγραμματισμού	6
2.1 HTML5.....	6
2.1.1 Εισαγωγή – Ορισμός HTML.....	6
2.1.2 Δομή της HTML.....	6
2.2 CSS3.....	8

2.3	Δυναμικές γλώσσες προγραμματισμού ιστού	8
2.3.1	JavaScript – Ορισμός.....	8
2.3.2	Πλεονεκτήματα JavaScript.....	9
2.3.3	JQuery.....	9
3	Responsive Design	10
3.1	Σημερινές τάσεις τεχνολογίας	10
3.2	Ανάγκες για responsive design.....	10
3.3	Πλεονεκτήματα responsive design	10
4	Εργαλεία κατασκευής NetBeans και περιβάλλον Bootstrap	12
4.1	NetBeans.....	12
4.2	Bootstrap.....	14
4.2.1	Εισαγωγή	14
4.2.2	Πλεονεκτήματα – Χρησιμότητα.....	14
5	Δημιουργία Αρχικής Σελίδας	15
5.1	Εισαγωγή – Αρχικές Δηλώσεις στον κώδικα HTML.....	15
5.2	Οθόνη Καλωσορίσματος	17
5.2.1	Σκοπός	17
5.2.2	Κώδικας HTML.....	17
5.2.3	Κώδικας CSS	18
5.2.3.1	Κώδικας HTML	20
5.2.3.2	Κώδικας CSS.....	21
5.2.3.3	Κώδικας JavaScript.....	21
5.2.4	Πάνω μέρος οθόνης καλωσορίσματος – Μπάρα πλοήγησης.....	22
5.2.4.1	Κώδικας HTML	22
5.2.4.2	Κώδικας CSS.....	23
5.2.5	Εικόνα φόντου (Background Image).....	24

5.3	Οθόνη για γενικές πληροφορίες Τεπακ.....	25
5.3.1	Grid System Bootstrap	25
5.3.2	Κώδικας HTML οθόνης με γενικές πληροφορίες	27
5.3.3	Κώδικας CSS οθόνης με γενικές πληροφορίες	28
5.4	Οθόνη για το τι περιλαμβάνει το website.....	29
5.4.1	Κώδικας HTML.....	30
5.4.2	Κώδικας CSS	31
5.5	Σχεδίαση σελίδας για άλλες συσκευές – εφαρμογή responsive design.....	32
5.5.1	Media queries	32
5.5.2	Κινητά τηλέφωνα	32
5.5.2.1	Εμφάνιση Οθόνης	33
5.5.2.2	Κώδικας HTML για μπάρα πλοήγησης κινητών τηλεφώνων.....	34
5.5.2.3	Κώδικας CSS.....	35
5.5.3	Συσκευές tablet.....	36
5.5.3.1	Εμφάνιση Οθονών.....	36
5.5.3.2	Κώδικας CSS.....	36
6	Δημιουργία Κεντρικής Σελίδας.....	37
6.1	Αρχική Οθόνη Κεντρικής Σελίδας	37
6.1.1	Πλαίσιο με εναλλαγή Διαφανειών	38
6.1.1.1	Κώδικας HTML	38
6.1.1.2	Κώδικας CSS.....	39
6.1.2	Μπάρα πλοήγησης.....	39
6.1.2.1	Κώδικας HTML	40
6.1.2.2	Κώδικας CSS.....	41
6.2	Οθόνες με Βασικές Πληροφορίες Τεπάκ	42
6.2.1	Κώδικας HTML.....	43

6.2.2	Κώδικας JavaScript για εμφάνιση της δεύτερης μπάρας πλοήγησης	44
6.2.3	Κώδικας CSS	44
6.3	Σχεδίαση σελίδας για άλλες συσκευές – responsive design.....	45
7	Δημιουργία σελίδων τμημάτων Τεπακ	47
7.1	Οθόνη περιγραφής τμήματος.....	47
7.1.1	Κώδικας HTML.....	48
7.1.2	Κώδικας CSS	49
7.2	Φωτογραφίες	49
7.2.1	Κώδικας HTML.....	50
7.2.2	Κώδικας CSS	51
7.2.3	Παράθυρο διαλόγου modal	52
7.3	Χάρτες	56
7.3.1	Κώδικας HTML και CSS	57
7.3.2	Κώδικας JavaScript – Χρήση Google Maps API.....	58
7.4	Σχεδίαση σελίδων για άλλες συσκευές – responsive design.....	59
8	Χρήση Animation	61
8.1	Εισαγωγή - Χρησιμότητα Animation.....	61
8.2	Ιδιότητες CSS3 για animation	62
8.3	Παραδείγματα.....	62
8.3.1	Ιδιότητα transition και transition-duration	63
8.3.2	Ιδιότητα animation και keyframes.....	64
9	Δημιουργία Footer.....	67
9.1	Πρακτικές δημιουργίας footer	67
9.2	Δημιουργία footer – κώδικας HTML και CSS	67
9.3	Εμφάνιση Footer σε μικρότερες συσκευές.....	70
	ΣΥΜΠΕΡΑΣΜΑΤΑ/ ΑΠΟΤΕΛΕΣΜΑΤΑ/ ΕΠΙΛΟΓΟΣ.....	72

ΒΙΒΛΙΟΓΡΑΦΙΑ..... 73

ΚΑΤΑΛΟΓΟΣ ΔΙΑΓΡΑΜΜΑΤΩΝ

Διάγραμμα 1: Παράδειγμα κώδικα εγγράφου HTML.....	7
Διάγραμμα 2: Περιβάλλον εργασίας NetBeans.....	12
Διάγραμμα 3: Περιβάλλον εργασίας NetBeans (2).....	13
Διάγραμμα 4: Δυνατότητες NetBeans.....	14
Διάγραμμα 5: Κώδικας επικεφαλίδας head.....	15
Διάγραμμα 6: Οθόνη Καλωσορίσματος.....	17
Διάγραμμα 7: Κώδικας HTML.....	17
Διάγραμμα 8: Κώδικας CSS για οθόνη καλωσορίσματος.....	18
Διάγραμμα 9: Δείκτης/Σύνδεσμος.....	20
Διάγραμμα 10: Κώδικας CSS για τον δείκτη/σύνδεσμο.....	21
Διάγραμμα 11: Συναρτήσεις JQuery.....	21
Διάγραμμα 12: Μπάρα πλοήγησης αρχικής σελίδας - Κώδικας HTML.....	22
Διάγραμμα 13: Κώδικας CSS για μπάρα πλοήγησης.....	23
Διάγραμμα 14: Κώδικας CSS για εικόνα φόντου.....	24
Διάγραμμα 15: Οθόνη Γενικών πληροφοριών.....	25
Διάγραμμα 16: Grid system bootstrap.....	26
Διάγραμμα 17: Παράδειγμα κώδικα Grid System Bootstrap.....	27
Διάγραμμα 18: Κώδικας HTML για οθόνη με γενικές πληροφορίες.....	27
Διάγραμμα 19: Κώδικας CSS για οθόνη γενικών πληροφοριών.....	28
Διάγραμμα 20: Εμφάνιση οθόνης για το τι περιλαμβάνει το website.....	29
Διάγραμμα 21: Κώδικας HTML για κομμάτι πληροφόρησης για website.....	30
Διάγραμμα 22: Κώδικας CSS για κομμάτι πληροφόρησης για website.....	31
Διάγραμμα 23: Εμφάνιση Οθονών Αρχικής σελίδας σε κινητές συσκευές.....	33
Διάγραμμα 24: Μπάρα πλοήγησης - κινητά τηλέφωνα.....	34

Διάγραμμα 25: Κώδικας CSS για κινητά τηλέφωνα.....	35
Διάγραμμα 26: Εμφάνιση Οθονών Αρχικής σελίδας σε συσκευές tablet.....	36
Διάγραμμα 27: Κώδικας CSS για συσκευές tablet.....	36
Διάγραμμα 28: Αρχική Οθόνη Κεντρικής Σελίδας.....	37
Διάγραμμα 29: Κώδικας HTML για Carousel.....	38
Διάγραμμα 30: Κώδικας CSS για Carousel.....	39
Διάγραμμα 31: Εμφάνιση λίστας/καταλόγου.....	40
Διάγραμμα 32: Κώδικας HTML - μπάρα πλοήγησης κεντρικής σελίδας.....	40
Διάγραμμα 33: Κώδικας CSS - μπάρα πλοήγησης κεντρικής σελίδας.....	41
Διάγραμμα 34: Δείγμα Οθόνης με βασικές πληροφορίες Τεπακ.....	42
Διάγραμμα 35: Κώδικας HTML για βασικές πληροφορίες Τεπακ.....	43
Διάγραμμα 36: JavaScript για εμφάνιση δεύτερης μπάρας πλοήγησης.....	44
Διάγραμμα 37: CSS για οθόνη με βασικές πληροφορίες Τεπακ.....	44
Διάγραμμα 38: Οθόνες κεντρικής σελίδας - κινητές συσκευές.....	46
Διάγραμμα 39: Κώδικας CSS για κινητές συσκευές - οθόνες κεντρικής σελίδας.....	46
Διάγραμμα 40: Οθόνη περιγραφής τμήματος.....	47
Διάγραμμα 41: Κώδικας HTML οθόνης περιγραφής τμήματος.....	48
Διάγραμμα 42: Κώδικας CSS - οθόνη περιγραφής τμήματος.....	49
Διάγραμμα 43: Τμήμα οθόνης σελίδας με φωτογραφικό υλικό.....	50
Διάγραμμα 44: HTML Κώδικας για τμήμα οθόνης με φωτογραφίες.....	50
Διάγραμμα 45: Κώδικας CSS για εμφάνιση φωτογραφιών.....	52
Διάγραμμα 46: Παράθυρο Διαλόγου για εμφάνιση φωτογραφιών.....	53
Διάγραμμα 47: Κώδικες HTML και CSS για Παράθυρο Διαλόγου.....	53
Διάγραμμα 48: Κώδικας JavaScript για εμφάνιση modal.....	54
Διάγραμμα 49: Κώδικας για χειρισμό κουμπιών Next και Previous στο modal.....	56
Διάγραμμα 50: Οθόνη με Χάρτες.....	57

Διάγραμμα 51: Οθόνη με χάρτες - Κώδικας HTML και CSS.....	57
Διάγραμμα 52: Κώδικας JavaScript για Google Maps API	58
Διάγραμμα 53: Εμφάνιση σελίδας Τμημάτων σε κινητά τηλέφωνα.....	60
Διάγραμμα 54: Κώδικας CSS - οθόνες κινητών συσκευών	60
Διάγραμμα 55: Παράδειγμα 1 με χρήση animation	63
Διάγραμμα 56: Παράδειγμα 2: με χρήση animation	64
Διάγραμμα 57: Κώδικας JQuery για έλεγχο εμφάνισης animation.....	65
Διάγραμμα 58: Κώδικας JQuery για ομαλή μετάβαση σε σημεία εντός μια σελίδας.....	66
Διάγραμμα 59: Οθόνη Footer	67
Διάγραμμα 60: Κώδικας HTML για Footer	68
Διάγραμμα 61: Κώδικας CSS για Footer	69
Διάγραμμα 62: Σελίδες Help και About.....	70
Διάγραμμα 63: Εμφάνιση Footer σε κινητά τηλέφωνα και κώδικας CSS	71
Διάγραμμα 64: Σελίδες Help και About σε μικρότερες συσκευές.....	71

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

ΤΕΠΑΚ:	Τεχνολογικό Πανεπιστήμιο Κύπρου
WWW	World Wide Web
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
API	Application Programming Interface

ΑΠΟΔΟΣΗ ΟΡΩΝ

WWW	Παγκόσμιος Ιστός
Website	Ιστότοπος
Web browser	Περιηγητής Ιστού
Website	Ιστότοπος
CSS	Διαδοχικά Φύλλα Στυλ
HTML	Γλώσσα Σήμανσης Υπερκειμένου
Tags	Ετικέτες
Animation	Σχεδιοκίνηση
Responsive design	Δυναμική προσαρμογή
Grid System	Σύστημα πλεγμάτων
Tablet	Υπολογιστής ταμπλέτας
Page Orientation	Προσανατολισμός σελίδας
Portrait or landscape	Κατά ύψος ή κατά πλάτος
Footer	Υποσέλιδο

ΕΙΣΑΓΩΓΗ

Όπως αναφέρθηκε και πιο πάνω, η εφαρμογή αποφασίστηκε να γίνει σε μορφή ιστοσελίδας λόγω των πολλών πλεονεκτημάτων που προσφέρει αυτή η προσέγγιση, τα οποία θα αναλυθούν εκτενέστερα στην συνέχεια.

Επίσης στην συνέχεια παρουσιάζονται οι διάφορες έννοιες που αφορούν το διαδίκτυο και τις τεχνολογίες του, αλλά και οι γλώσσες προγραμματισμού και τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη και την υλοποίηση αυτής της ιστοσελίδας.

Ακόμα ένα σημαντικό κομμάτι που λήφθηκε σοβαρά υπόψη, ήταν ο όρος της «δυναμικής προσαρμογής» μιας ιστοσελίδας, ή αλλιώς όπως είναι και ευρύτερα γνωστός, responsive design. Σύμφωνα με αυτή την προσέγγιση, μια ιστοσελίδα κατασκευάζεται με τέτοιο τρόπο έτσι ώστε να προσαρμόζεται και να παρουσιάζεται με εύκολο, γρήγορο και ευανάγνωστο τρόπο σε κάθε είδους συσκευή, ανεξάρτητα από το μέγεθος της οθόνης στην οποία καλείται να παρουσιαστεί η σελίδα στον χρήστη. Πιο κάτω, γίνεται μια περιγραφή του πως αυτή η τεχνική μπορεί να καταστεί χρήσιμη και γιατί είναι απαραίτητη η εφαρμογή της στις σύγχρονες μεθόδους κατασκευής ιστοσελίδων.

Επιπλέον, γίνεται ανάλυση των διαφόρων σύγχρονων τεχνικών σχεδιασμού που εφαρμόστηκαν στην δημιουργία της σελίδας αυτής και το πως συμβάλλουν στο να κάνουν την εφαρμογή πιο φιλική και αποτελεσματική προς τον χρήστη. Γίνεται επίσης επεξήγηση του κώδικα που γράφτηκε και παρουσιάζονται τα διάφορα κομμάτια της ιστοσελίδας και οι επιλογές που θα έχει στην διάθεση του ο χρήστης.

1 Διαδίκτυο και βασικές έννοιες

1.1 Διαδίκτυο

1.1.1 Εισαγωγή

Με την ραγδαία ανάπτυξη της τεχνολογίας που έχει επιτευχθεί τα τελευταία χρόνια, το Διαδίκτυο αποτελεί ένα καθημερινό κομμάτι στην ζωή μας, καθώς λόγω των τεράστιων δυνατοτήτων και πλεονεκτημάτων του, αποτελεί ένα από τα κυριότερα μέσα με τα οποία ο άνθρωπος επικοινωνεί, λαμβάνει δεδομένα και επεξεργάζεται πληροφορίες.

1.1.2 Ορισμός

Το Διαδίκτυο είναι ένα σύστημα από διάφορα δίκτυα υπολογιστών που μπορεί να βρίσκονται τοποθετημένα σε οποιαδήποτε σημεία του πλανήτη, και επικοινωνούν μεταξύ τους χρησιμοποιώντας συγκεκριμένες ομάδες πρωτοκόλλων.

Δύο ή περισσότεροι υπολογιστές που είναι μεταξύ τους συνδεδεμένοι αποτελούν ένα δίκτυο. Τα πολλαπλά δίκτυα υπολογιστών σε ολόκληρο τον κόσμο που είναι διασυνδεδεμένα μεταξύ τους, σχηματίζουν αυτό που αποκαλούμε διαδίκτυο.

Η σύνδεση αυτή γίνεται με την βοήθεια τηλεπικοινωνιακών καλωδίων ή και ασύρματα. Όπως αναφέραμε και πιο πάνω οι επικοινωνία αυτή γίνεται με την χρήση διαφόρων πρωτοκόλλων (τόσο σε λογισμικό όσο και υλικό επίπεδο) τα οποία είναι υπεύθυνα για την ασφαλή και γρήγορη μεταφορά των δεδομένων από υπολογιστή σε υπολογιστή, αλλά και για την εμφάνιση και παρουσίαση τους στον χρήστη.

Αξίζει να σημειωθεί ότι το διαδίκτυο δεν αποτελείται μόνο από συνδεδεμένους ηλεκτρονικούς υπολογιστές, αλλά και από άλλες συσκευές όπως για παράδειγμα κινητά τηλέφωνα, τηλεοράσεις, με τον αριθμό και την ποικιλία συσκευών να αναμένεται να αυξηθεί όσο περνούν τα χρόνια.

1.2 Παγκόσμιος Ιστός

Ο Παγκόσμιος Ιστός (www – world wide web όπως είναι και πιο γνωστός) είναι μια από τις πιο δημοφιλείς και ευρύτερα χρησιμοποιούμενες υπηρεσίες του διαδικτύου. Μέσω αυτής της υπηρεσίας και των τεχνολογιών ιστού που εφαρμόζονται σήμερα, οι χρήστες του διαδικτύου μπορούν να αναζητούν και να έχουν εύκολη και γρήγορη πρόσβαση σε διάφορα δεδομένα

και πληροφορίες που χρειάζονται. Αυτό γίνεται με την χρήση του πρωτοκόλλου HTTP, το οποίο είναι υπεύθυνο για την σωστή παρουσίαση των δεδομένων αυτών στην οθόνη του χρήστη.

1.3 Ιστοσελίδες

1.3.1 Εισαγωγή

Οι ιστοσελίδες είναι ψηφιακά έγγραφα τα οποία είναι μέρος του Παγκόσμιου Ιστού. Τα έγγραφα αυτά μπορεί να είναι αποθηκευμένα σε οποιαδήποτε ψηφιακή μορφή, όπως κείμενο, ήχο, εικόνα ή βίντεο. Ο χρήστης μπορεί να έχει εύκολα πρόσβαση στην ιστοσελίδα της επιλογής του με την χρήση του κατάλληλου λογισμικού (web browser), αρκεί να γνωρίζει την διεύθυνση της στον Παγκόσμιο Ιστό. Πολλές ιστοσελίδες μαζί αποτελούν έναν ιστότοπο (website) και ανήκουν στην ίδια διεύθυνση.

1.3.2 Web browser

Ο web browser ή αλλιώς φυλλομετρητής ή περιηγητής ιστού, είναι το πρόγραμμα που επιτρέπει στον χρήστη του διαδικτύου να εμφανίζει στην οθόνη του με εύκολο και γρήγορο τρόπο τις διάφορες ιστοσελίδες που αυτός επιθυμεί. Το λογισμικό αυτό χρησιμοποιεί το πρωτόκολλο HTTP που σε συνεργασία με τα υπόλοιπα πρωτόκολλα του διαδικτύου, είναι υπεύθυνο για την τελική μεταφορά των δεδομένων μιας ιστοσελίδας στον πελάτη (χρήστη). Μέσω του web browser, ο χρήστης μπορεί να μεταβαίνει από μια σελίδα σε άλλη, να αλληλεπιδρά με το περιεχόμενο και τις πληροφορίες που βρίσκονται σε αυτές, αλλά και να επιλέγει τις δικές του ρυθμίσεις για το πως θέλει να του εμφανίζονται αυτά τα διάφορα στοιχεία. Αυτό βέβαια εξαρτάται και από τον τρόπο που λειτουργεί και τις δυνατότητες που προσφέρει ο κάθε web browser ξεχωριστά. Οι κυριότεροι που χρησιμοποιούνται σήμερα είναι ο Google Chrome, ο Mozilla Firefox, ο Internet Explorer, ο Opera και ο Safari.

1.3.3 Κατασκευή Ιστοσελίδων.

1.3.3.1 Διαδικασία

Η κατασκευή των ιστοσελίδων είναι η διαδικασία κατά την οποία δημιουργείται το περιεχόμενο το οποίο θα παρουσιαστεί στον χρήστη όταν αυτός επιλέξει να έχει πρόσβαση σε ένα ιστότοπο.

Η διαδικασία αυτή περιλαμβάνει και απαιτεί την χρήση διαφόρων τεχνικών και ικανοτήτων όπως για παράδειγμα τον κατάλληλο σχεδιασμό του περιεχομένου της ιστοσελίδας και τον τρόπο με τον οποίο θα παρουσιάζεται το περιεχόμενο στους χρήστες (φωτογραφίες, κείμενο, ήχος, χρήση κινούμενων εικόνων κλπ). Ο κυριότερος στόχος αυτών των τεχνικών είναι να βελτιστοποιηθεί η αλληλεπίδραση μεταξύ ανθρώπου-υπολογιστή, έτσι ώστε τα μηνύματα και οι πληροφορίες που περιέχονται σε μια ιστοσελίδα να μεταδίδονται στον χρήστη με τον όσο πιο δυνατό εύκολο και κατανοητό τρόπο, αλλά και ταυτόχρονα να δημιουργούν ένα όμορφο και ευχάριστο περιβάλλον περιήγησης για τον χρήστη.

1.3.3.2 Τρόποι κατασκευής

Υπάρχουν διάφορες τεχνολογίες που υπάρχουν και χρησιμοποιούνται σήμερα για τον σχεδιασμό και την κατασκευή ιστοσελίδων. Μερικές από αυτές περιλαμβάνουν τις γλώσσες HTML5, CSS3, JavaScript, JQuery(περισσότερη ανάλυση στην συνέχεια) τις οποίες και χρησιμοποιήσαμε για την κατασκευή της εφαρμογής που αφορά αυτή την πτυχιακή εργασία.

Επίσης υπάρχουν πολλά διαθέσιμα προγράμματα και αυτόνομοι μηχανισμοί με τα οποία οι χρήστες τους μπορούν να δημιουργήσουν με πιο αυτόματο τρόπο περιεχόμενα ιστοσελίδων. Στα πλαίσια όμως αυτής της εργασίας δεν χρησιμοποιήσαμε κάποιο από αυτά τα έτοιμα προγράμματα καθώς συνήθως δεν προσφέρουν τον κατάλληλο βαθμό ελευθερίας που αφορά το πως θα γίνει η σχεδίαση του περιεχομένου της ιστοσελίδας από τους σχεδιαστές.

Ακόμη, η ανάπτυξη της εφαρμογής χωρίς την χρήση κάποιου αυτοματοποιημένου προγράμματος, είχε σκοπό στο να βοηθήσει στην περαιτέρω ανάπτυξη των γνώσεων που αφορούν τις γλώσσες σήμανσης και προγραμματισμού ιστοσελίδων.

1.3.4 Σημασία και χρησιμότητα

Τα πλεονεκτήματα της χρήσης μιας ιστοσελίδας η γενικά ενός ιστότοπου, είναι τεράστια στην σύγχρονη εποχή, γι' αυτό και αποφασίστηκε η κατασκευή μιας διαδικτυακής εφαρμογής στα πλαίσια αυτής της εργασίας.

- Εύκολη πρόσβαση:

Αρχικά, η πρόσβαση σε μια ιστοσελίδα, αλλά και η μετάβαση από μια ιστοσελίδα σε άλλη είναι μια διαδικασία εύκολη και γρήγορη για τον χρήστη και δεν απαιτεί κάποιες ιδιαίτερες γνώσεις για να γίνει αυτό. Επίσης μπορεί αυτό να γίνει με οποιαδήποτε συσκευή, αρκεί ο χρήστης απλά να γνωρίζει την διεύθυνση της σελίδας στον Παγκόσμιο

Ιστό, να έχει σύνδεση στο διαδίκτυο και να χρησιμοποιήσει ένα πρόγραμμα web browser της επιλογής του με το οποίο θα πετύχει την πρόσβαση σε αυτή.

- Αποτελεσματικό μέσο επικοινωνίας:

Το διαδίκτυο αποτελεί σήμερα ένα από τα κυριότερα μέσα ενημέρωσης, επικοινωνίας, ψυχαγωγίας και άντλησης πληροφοριών και λαμβάνοντας υπόψη ότι περισσότερο μέρος του πληθυσμού μπορεί εύκολα να έχει σύνδεση σε αυτό, τότε η χρήση μιας ιστοσελίδας με σκοπό την πληροφόρηση και ενημέρωση αποτελεί αναμφισβήτητα ένα αποδοτικό τρόπο για να γίνει αυτό.

- Διαθεσιμότητα:

Μια ιστοσελίδα μπορεί να είναι διαθέσιμη στον χρήστη οποιαδήποτε στιγμή της ημέρας και το κυριότερο, με τις δυνατότητες που προσφέρουν σήμερα οι ασύρματες συνδέσεις μπορεί να έχει την επισκεφτεί ακόμα και όταν δεν βρίσκεται στο σπίτι του, με την χρήση για παράδειγμα μιας κινητής συσκευής.

- Συμβατικότητα:

Ακόμη ένα πλεονέκτημα είναι το γεγονός ότι δεν προϋποθέτει ο χρήστης να έχει κάποιο συγκεκριμένο λειτουργικό σύστημα εγκατεστημένο για να μπορεί να έχει πρόσβαση στο διαδίκτυο, καθώς οι σημερινοί web browsers είναι διαθέσιμοι σε όλα τα λειτουργικά συστήματα.

- Πολύ λίγο κόστος χρήσης:

Για να χρησιμοποιήσει κάποιος μια ιστοσελίδα αλλά και γενικά η πρόσβαση στο διαδίκτυο δεν απαιτεί ο χρήστης να έχει κάποια συγκεκριμένο υλικό στον υπολογιστή του (ή γενικά σε οποιαδήποτε συσκευή) ούτε κάποιο πολύπλοκο λογισμικό εγκατεστημένο. Επιπλέον, η ιστοσελίδα δεν βρίσκεται αποθηκευμένη στη συσκευή του χρήστη, αλλά κατεβαίνει από κάποιο εξωτερικό υπολογιστή του διαδικτύου και αποθηκεύεται προσωρινά στον web browser, κάτι που συμβάλλει έτσι ώστε η κατανάλωση χώρου και πόρων στο σύστημα του χρήστη να είναι σχεδόν αμελητέα.

- Κόστος και δυνατότητες στην κατασκευή:

Οι σύγχρονες τεχνολογίες ιστού που υπάρχουν προσφέρουν πολλές δυνατότητες στον κατασκευαστή μιας διαδικτυακής εφαρμογής καθώς μπορεί να διαμορφώσει και να εμπλουτίσει το περιεχόμενο της ιστοσελίδας όπως αυτός επιθυμεί, με τρόπους που θα

δημιουργήσουν ένα περιβάλλον όμορφο και ευχάριστο για τον χρήστη. Επίσης υπάρχουν σήμερα πάρα πολλοί διαθέσιμοι οδηγοί εκμάθησης όλων αυτών των τεχνολογιών, κάτι που αυξάνει τις επιλογές και διευκολύνσεις που έχει στην διάθεση του ο δημιουργός.

- Εύκολη συντήρηση:

Ένα ακόμη πλεονέκτημα που λήφθηκε σοβαρά υπόψη, είναι το γεγονός ότι η συντήρηση μιας ιστοσελίδας και γενικά οι οποιοσδήποτε μελλοντικές αναβαθμίσεις χρειαστούν να γίνουν δεν απαιτούν ιδιαίτερο κόπο και χρόνο. Και αυτό γιατί αφού η ιστοσελίδα είναι αποθηκευμένη σε κάποιο εξωτερικό υπολογιστή, μόνο μια τοπική αναβάθμιση χρειάζεται, χωρίς να είναι αναγκαίες οι αλλαγές σε κάθε υπολογιστή ξεχωριστά που γίνεται η χρήση της εφαρμογής κάτι που εξοικονομεί κόστος και χρόνο.

2 Τεχνολογίες Ιστού – Γλώσσες προγραμματισμού

Σε αυτό το σημείο θα αναφερθούμε στις τεχνολογίες που χρησιμοποιήσαμε για την ανάπτυξη αυτής της εφαρμογής και το πώς συνέβαλαν σε αυτό.

2.1 HTML5

2.1.1 Εισαγωγή – Ορισμός HTML

Η HTML είναι μια γλώσσα σήμανσης υπερκειμένου που ο σκοπός της είναι να καθορίζει την δομή και την μορφή με την οποία παρουσιάζονται οι ιστοσελίδες στον web browser.

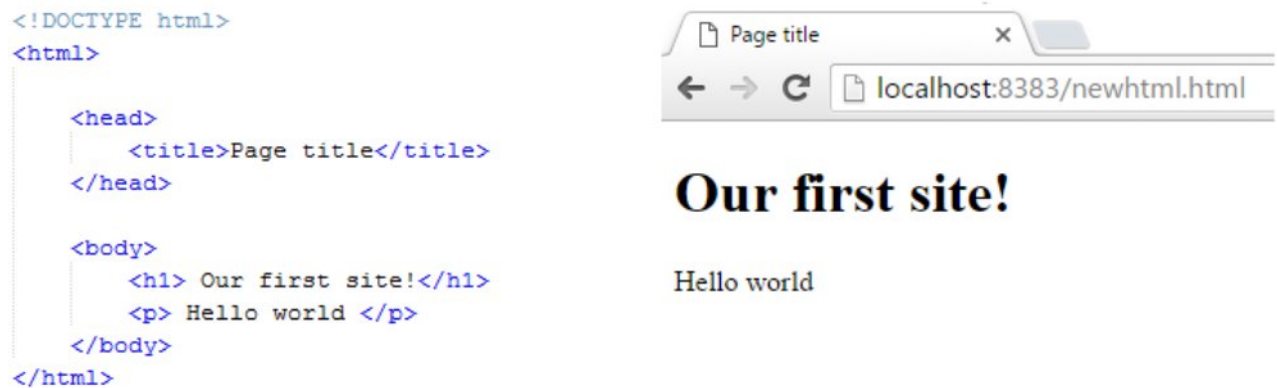
Η γλώσσα σήμανσης(markup language) είναι μια γλώσσα που περιέχει κείμενο το οποίο χαρακτηρίζεται από διάφορους κανόνες και σύμβολα με στόχο να δώσουν οδηγίες σε ένα υπολογιστικό σύστημα για τον τρόπο με τον οποίο θα παρουσιάζονται τα διάφορα στοιχεία ενός εγγράφου. Αυτό που τις διαχωρίζει από τις συνηθισμένες γλώσσες προγραμματισμού είναι ότι σε αντίθεση με αυτές, δεν εκτελούν διαδικασίες που αφορούν την επεξεργασία δεδομένων ή την λήψη αποφάσεων για το τι θα εκτελεστεί σε ένα πρόγραμμα, απλά περιγράφουν εντολές και δίνουν οδηγίες για την μορφή ενός εγγράφου. Η HTML αποτελεί σήμερα την πιο διάσημη γλώσσα σήμανσης που χρησιμοποιείται για την παρουσίαση ιστοσελίδων, με την HTML5 να είναι η πιο πρόσφατη έκδοση της.

2.1.2 Δομή της HTML

Η δομή της HTML αποτελείται από τα tags(ετικέτες) τα οποία είναι ειδικά στοιχεία που περιγράφουν και ελέγχουν την δομή του εγγράφου με την χρήση συμβόλων, βοηθώντας έτσι τον web browser να ερμηνεύσει σωστά το περιεχόμενό του, ενώ επίσης δίνουν και κάποιες γενικές πληροφορίες που αφορούν μια ιστοσελίδα.

Τα tags περιγράφονται από τα σύμβολα '<' (ετικέτα αρχής) και '>' (ετικέτα τέλους) και μέσα τους περιλαμβάνεται τυποποιημένο το όνομα του στοιχείου που περιγράφουν (για παράδειγμα κείμενο, επικεφαλίδα, εικόνα. Βίντεο κλπ.), ενώ μεταξύ των tags βρίσκεται το περιεχόμενο. Για παράδειγμα: <p> Αυτή είναι μια παράγραφος. </p> , με το σύμβολο p να υποδεικνύει ότι πρόκειται για παράγραφο. Επίσης τα στοιχεία αυτά μπορούν να έχουν διάφορες ιδιότητες(attributes) που να χαρακτηρίζουν την συμπεριφορά τους.

Ένα μικρό παράδειγμα ενός εγγράφου HTML5, μαζί με το αποτέλεσμα του φαίνεται στην εικόνα που ακολουθεί:



Διάγραμμα 1: Παράδειγμα κώδικα εγγράφου HTML

Αριστερά, βλέπουμε τον κώδικα σε HTML και δεξιά, την μορφή που έχει η σελίδα στον web browser.

Ο κώδικας αρχίζει με την δήλωση `<!DOCTYPE html>` η οποία είναι μια δήλωση στον web browser που αφορά το είδος του εγγράφου που είναι γραμμένη η σελίδα. Κάθε έγγραφο HTML αρχίζει και τελειώνει με τα tags `<html>` και `</html>` που δηλώνουν ότι ο κώδικας που βρίσκεται σε αυτό το έγγραφο είναι κώδικας html.

Στην συνέχεια, ο κώδικας αποτελείται από δύο βασικές ενότητες, την επικεφαλίδα (`<head></head>`) και το κύριο μέρος (`<body></body>`). Στην επικεφαλίδα περιγράφονται κάποιες βασικές πληροφορίες που αφορούν την ιστοσελίδα, όπως για παράδειγμα ο τίτλος της σελίδας που καθορίζεται από το tag `<title></title>` (εμφανίζεται στο πάνω μέρος του παραθύρου του web browser, όπως φαίνεται και στο παράδειγμα αυτό). Το κύριο μέρος αποτελεί την κύρια δομή του εγγράφου στο οποίο περιέχονται τα βασικά στοιχεία που χαρακτηρίζουν την σελίδα. Στην περίπτωση αυτή, το κύριο μέρος αποτελείται από δύο tags το `<h1>` που περιγράφει μια επικεφαλίδα και το `<p>` που περιγράφει μια παράγραφο, όπως φαίνεται και στο αποτέλεσμα στον web browser.

Το πλήθος των πιθανών tags που μπορούν να χρησιμοποιηθούν αυξάνονται όλο και περισσότερο με το πέρασμα των χρόνων , προσφέροντας έτσι καινούργιες δυνατότητες στον δημιουργό μιας ιστοσελίδας.

2.2 CSS3

Η CSS είναι γλώσσα φύλλων στυλ, που χρησιμοποιείται για να οριστεί το στυλ που θα έχει ένα HTML έγγραφο και τον τρόπο με τον οποίο θα εμφανιστούν και θα παρουσιαστούν τα διάφορα στοιχεία του. Ο κύριος σκοπός της είναι να προσδίδει μια ομορφιά και ένα ξεχωριστό στυλ που θα εμπλουτίζει την κάθε ιστοσελίδα και τα στοιχεία της, δίνοντας την έτσι την δυνατότητα να ξεφύγει από την απλότητα που περιγράφεται μέσω της HTML και να δημιουργώντας ένα πιο ευχάριστο περιβάλλον για τον χρήστη.

Μερικά από τα στυλ που μπορούμε να ορίσουμε μέσω της CSS είναι το χρώμα και το μέγεθος των γραμματοσειρών, την στοίχιση, την ύπαρξη περιγραμμάτων, την απόσταση μεταξύ γραμμμάτων και λέξεων και πολλά άλλα.

Ένα από τα πιο σημαντικά πλεονεκτήματα της CSS είναι ότι επιτρέπει τον ορισμό του κώδικά της σε ξεχωριστό αρχείο από αυτό της HTML με τα δύο αρχεία να επικοινωνούν μεταξύ τους κατάλληλα χωρίς να χρειάζεται η ύπαρξη τους στο ίδιο αρχείο κάτι που θα έκανε τον κώδικα δυσανάγνωστο και την συντήρηση δυσκολότερη.

Η CSS3 αποτελεί την πιο πρόσφατη έκδοση της CSS, που με την κυκλοφορία της ήρθε να προσφέρει καινούργιες δυνατότητες και ευκαιρίες για την περιγραφή του στυλ διαφόρων στοιχείων και κυρίως τη ευκαιρία για δημιουργία animations εντός μιας ιστοσελίδας.

2.3 Δυναμικές γλώσσες προγραμματισμού ιστού

2.3.1 JavaScript – Ορισμός

Η JavaScript είναι μια διερμηνευμένη γλώσσα προγραμματισμού που χρησιμοποιείται για να προσδίδει στις ιστοσελίδες περισσότερη διαδραστικότητα και αλληλεπαφή μεταξύ χρήστη και της σελίδας. Είναι μια γλώσσα η οποία έχει πάρει διάφορα στοιχεία από άλλες γλώσσες προγραμματισμού (όπως για παράδειγμα την αντικειμενοστραφή και συναρτησιακό στυλ προγραμματισμού) και εκτελείτε στην πλευρά του πελάτη, δηλαδή κατευθείαν στον web browser του χρήστη χωρίς να χρειάζεται να λαμβάνει δεδομένα από κάποιο εξωτερικό υπολογιστή του διαδικτύου.

Διαφέρει με την HTML και την CSS στο γεγονός ότι περιέχει συναρτήσεις και επεξεργάζεται δεδομένα με σκοπό να παρουσιάσει ένα αποτέλεσμα. Ο στόχος της είναι να καθορίσει την συμπεριφορά της ιστοσελίδας όταν ο χρήστης αλληλεπιδρά με αυτή. Για παράδειγμα όταν ο

χρήστης πατήσει ένα κουμπί, πληκτρολογήσει κάτι, μεταβεί σε ένα καινούργιο σημείο στην σελίδα και γενικά όταν εκτελέσει μια πράξη, μέσω της JavaScript παρέχεται η δυνατότητα να καλεστεί κάποια συνάρτηση η οποία αλλάζει δυναμικά το περιεχόμενο του HTML εγγράφου.

2.3.2 Πλεονεκτήματα JavaScript

Όπως αναφέρθηκε και πιο πάνω, ο κώδικας JavaScript μπορεί να εκτελεστεί κατευθείαν στον web browser του χρήστη χωρίς να χρειάζεται η επικοινωνία με κάποιο τρίτο, εξωτερικό υπολογιστή. Αυτό κάνει την φόρτωση μιας ιστοσελίδας πιο γρήγορη και έτσι ο χρήστης δεν χρειάζεται να περιμένει για την εμφάνιση ενός καινούργιου περιεχομένου.

Παρόμοια και με την CSS, η JavaScript μειώνει την επιβάρυνση του HTML κώδικα, καθώς η εκτέλεση των λειτουργιών που περιλαμβάνει διαχωρίζεται από τον καθορισμό της δομής της σελίδας.

Επίσης, με τις δυνατότητες που προσφέρει η JavaScript, δημιουργείται ένα πιο ευχάριστο και φιλικό περιβάλλον προς τον χρήστη, αυξάνοντας το επίπεδο διαδραστικότητας και επαφής μεταξύ του χρήστη και της ιστοσελίδας.

2.3.3 JQuery

Η JQuery είναι μια βιβλιοθήκη της JavaScript, που περιέχει συναρτήσεις και χρήση διαφόρων μεταβλητών που απλοποιούν σε μεγάλο βαθμό τον προγραμματισμό σε JavaScript. Αυτό σημαίνει ότι με την χρήση της JQuery μειώνεται η ποσότητα του κώδικα που πρέπει να γραφτεί, κάτι που επιτυγχάνεται με τις διάφορες συναρτήσεις που έχει και τον τρόπο που χειρίζεται τα διάφορα στοιχεία του HTML εγγράφου για την αλληλεπίδραση του με την ιστοσελίδα. Επίσης τα ονόματα των συναρτήσεων και των μεταβλητών που χρησιμοποιούνται διακρίνονται για την σαφήνεια τους και για την ευκολία χρήσης τους κάτι που συμβάλλουν στο διευκόλυνση της συγγραφής κώδικα από τον προγραμματιστή.

Αυτές αποτελούν τις κύριες γλώσσες προγραμματισμού και σήμανσης που χρησιμοποιήσαμε οι οποίες βοηθούν αποτελεσματικά στην ομαλή διεκπεραίωση της δημιουργίας ιστοσελίδων. Περισσότερη ανάλυση όλων αυτών των γλωσσών θα γίνει και στην συνέχεια, όπου θα επεξηγηθούν και τα διάφορα στοιχεία και οι συναρτήσεις που χρησιμοποιήθηκαν για την κατασκευή αυτού του website, καθώς και ο τρόπος που συνδέονται μεταξύ τους τα διάφορα αρχεία.

3 Responsive Design

3.1 Σημερινές τάσεις τεχνολογίας

Η τεχνολογία με το πέρασμα των χρόνων αναπτύσσεται όλο και περισσότερο, καθώς χρόνο με τον χρόνο, δημιουργούνται καινούργιες συσκευές τόσο σε επίπεδο επιτραπέζιων υπολογιστών όσο και σε επίπεδο κινητών συσκευών. Οι υπολογιστικές δυνατότητες που κατέχουν σήμερα τα κινητά τηλέφωνα αλλά και άλλες μοντέρνες συσκευές είναι τεράστιες σε σχέση με τα παλιά τα χρόνια και σε συνδυασμό με την φορητότητα και τις πολλές χρήσιμες εφαρμογές που προσφέρουν, έχουν οδηγήσει στην αύξηση της χρήσης τους δραστικά.

3.2 Ανάγκες για responsive design

Λόγω των πιο πάνω συνθηκών, έχει παρατηρηθεί τα τελευταία χρόνια μια μεγάλη αύξηση στην χρήση του διαδικτύου από κινητές συσκευές. Έτσι μέσα από αυτό, προέρχεται και η ανάγκη για εφαρμογή της τεχνικής που ονομάζεται responsive design για τον σχεδιασμό των ιστοσελίδων. Σύμφωνα με αυτή την τακτική μια ιστοσελίδα σχεδιάζεται με τέτοιο τρόπο έτσι ώστε το περιεχόμενό της, να διαμορφώνεται και να προσαρμόζεται ανάλογα με την συσκευή και το μέγεθος της οθόνης που καλείται να εμφανιστεί η σελίδα.

3.3 Πλεονεκτήματα responsive design

Τα πλεονεκτήματα αυτής της τεχνικής σχεδιασμού είναι πολλά. Ο κύριος της στόχος είναι μια ιστοσελίδα να είναι ευανάγνωστη και ευπαρουσίαστη προς τον χρήστη από όποια συσκευή και να επιλέξει να την επισκεφθεί, χωρίς να χρειάζεται να κάνει πλάγια μετακίνηση ή κάποια μεγέθυνση σε αυτή.

Σημαντικό ρόλο παίζει αυτό και στον τομέα των επιχειρήσεων, καθώς επέρχεται περισσότερο κέρδος στις επιχειρήσεις όταν ο χρήστης μπορεί με ευκολία να έχει πρόσβαση στην ιστοσελίδα της επιχείρησης από οποιαδήποτε συσκευή. Και αυτό γιατί σε αντίθετη περίπτωση οι χρήστες θα ήταν δυσαρεστημένοι με το περιεχόμενο, κάτι που θα πολύ πιθανό να σήμαινε χάσιμο επισκεπτών για την σελίδα.

Μια άλλη πιθανή λύση στο πρόβλημα που δημιουργείται λόγω των σημερινών αναγκών θα ήταν να δημιουργηθεί μια ξεχωριστή εφαρμογή για κάθε διαθέσιμη συσκευή και πλατφόρμα που υπάρχει. Αυτή η λύση όμως θα ήταν αρκετά χρονοβόρα, με περισσότερο κόστος και καθόλου αποτελεσματική ενώ επίσης είναι αδύνατο να προβλεφθεί η κυκλοφορία καινούργιων συσκευών και οι καινούργιες ανάγκες που πιθανόν να προκύψουν.

Έτσι, το να δημιουργηθεί μια εφαρμογή και να είναι προσαρμόσιμη ανάλογα, αποτελεί μια πιο ιδανική λύση, κάτι που θα διευκολύνει επίσης οποιαδήποτε μελλοντική συντήρηση χρειαστεί να γίνει από τον δημιουργό.

4 Εργαλεία κατασκευής NetBeans και περιβάλλον Bootstrap

4.1 NetBeans

Για την δημιουργία αυτού του website χρησιμοποιήσαμε το εργαλείο κατασκευής λογισμικού NetBeans. Το λογισμικό αυτό παρέχει πολλές δυνατότητες στον χρήστη του και ειδικά στην περίπτωση της κατασκευής ιστοσελίδων.

Αρχικά, προσφέρει την δυνατότητα για την δημιουργία αρχείων και συγγραφής κώδικα όλων των ειδών των πιο σύγχρονων γλωσσών προγραμματισμού και ιστού.

Επίσης περιέχει ένα μοντέρνο επεξεργαστή κειμένου που βοηθά με τον κατάλληλο χρωματισμό στο να διακρίνονται οι διάφορες ιδιότητες και στοιχεία του κώδικα όπως για παράδειγμα κλάσεις, συναρτήσεις, μεταβλητές, με τρόπο όπου να διευκολύνει τον δημιουργό, όπως φαίνεται και στο δεξιό μέρος της πιο κάτω εικόνας. Στο αριστερό μέρος παρατηρούμε τον διαχωρισμό των αρχείων που γίνεται από το σύστημα ανάλογα και τους φακέλους που δημιουργεί ο χρήστης, κάτι που συμβάλλει στη καλύτερη διαχείριση των αρχείων ειδικά στην περίπτωση όπου ο αριθμός τους είναι αρκετά μεγάλος.



Διάγραμμα 2: Περιβάλλον εργασίας NetBeans

Ακόμη, παρέχει την λειτουργία της αυτόματης καταχώρησης κλάσης ή μιας ιδιότητας. Όταν δηλαδή ο χρήστης πρόκειται να πληκτρολογήσει κάτι στον επεξεργαστή, το σύστημα του προσφέρει επιλογές που έχει στην διάθεση του να διαλέξει. Επιπλέον, του εμφανίζει κάποιες βασικές πληροφορίες που αφορούν την ιδιότητα που θέλει να χρησιμοποιήσει, ενώ τον ειδοποιά για τυχόν συντακτικά λάθη που μπορεί να προκύψουν κατά την γραφή του κώδικα μαζί με πιθανές κατάλληλες λύσεις για επιδιόρθωση τους.

The screenshot shows the NetBeans IDE with a CSS file open. The code includes a comment about the 'margin' property and several CSS rules. A tooltip is displayed over the 'margin' property in the '.header1' rule, listing its possible values: 'margin', 'margin-bottom', 'margin-left', 'margin-right', and 'margin-top'. The tooltip also indicates that the property is of type 'Box'.

```

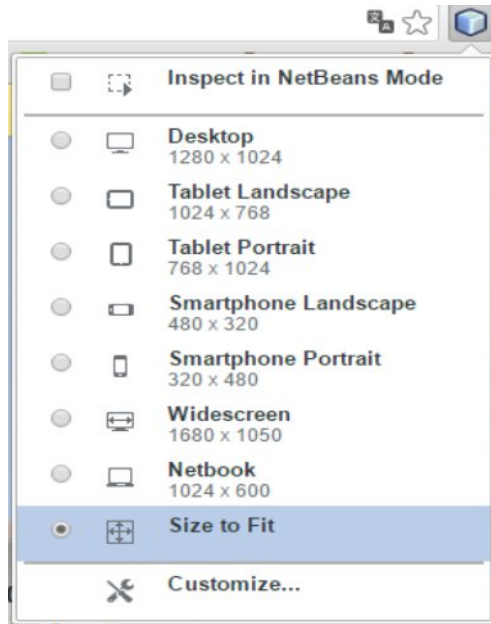
22     background-color: #d6d6c2;
23 }
24
25 /*nav bar styling*/
26 .navbar-inverse {
27     background-color: #333;
28     border: none;
29     border-radius: 4px;
30     -webkit-animation: 0.5s ease 0s;
31     -webkit-animation: 0.5s ease 0s;
32     animation-name: none;
33     animation-duration: 0.5s;
34 }
35
36 .header1 { padding-left: 10px; }
37 .header1 { margin: 10px; }
38 .glyphicon-home { font-size: 2em; }
39 .header1_text { font-size: 1.2em; font-weight: bold; }
40
41 .navbar-right li a {
42     color: white;
43     letter-spacing: 1px;
44     font-family: "Montserrat", sans-serif;
45 }
46
47 .header1:hover {
48     color: #d6d6c2 !important;

```

Διάγραμμα 3: Περιβάλλον εργασίας NetBeans (2)

Το NetBeans, με την τελευταία του έκδοση, εκμεταλλεύεται όλες τις καινούργιες δυνατότητες που προσφέρει η HTML5 και με την χρήση του, υποστηρίζει την τεχνική του responsive design έτσι ώστε οι ιστοσελίδες που δημιουργούνται να προσαρμόζονται σε όλα τα είδη οθονών. Παρέχει επίσης ένα ειδικό μηχανισμό ο οποίος λειτουργεί στον web browser Google Chrome με τον οποίο μπορούμε να βλέπουμε απευθείας σε αυτόν, τον τρόπο που εμφανίζεται η ιστοσελίδα και τις αλλαγές που γίνονται βάση του κώδικα μας.

Επίσης η τεχνική του responsive design, υποστηρίζεται με την παροχή της δυνατότητας να αλλάξουμε τις διαστάσεις του web browser μας για να συμβαδίζει με το μέγεθος της οθόνης της συσκευής που θέλουμε να ελέγξουμε την εμφάνιση μιας ιστοσελίδας σε αυτή.



Διάγραμμα 4: Δυνατότητες NetBeans

4.2 Bootstrap

4.2.1 Εισαγωγή

Το Bootstrap είναι μια συλλογή από βιβλιοθήκες ανοιχτού κώδικα που χρησιμοποιούνται για την δημιουργία ιστοσελίδων. Συμπεριλαμβάνει διάφορα στοιχεία και κλάσεις που χρησιμοποιούνται ειδικά στον κώδικα των HTML εγγράφων για να ορίζουν την δομή κάποιων κομματιών του περιεχομένου, άλλα και το στυλ τους. Επίσης περιέχει κλάσεις και βιβλιοθήκες σε JavaScript για να αυξάνει το επίπεδο διαδραστικότητας των ιστοσελίδων.

4.2.2 Πλεονεκτήματα – Χρησιμότητα

Ένα από τα τεράστια πλεονεκτήματα του που είναι και ο κύριος λόγος που το χρησιμοποιήσαμε στα πλαίσια αυτής της πτυχιακής είναι το γεγονός ότι συμβάλλει στην επίτευξη των στόχων της τεχνικής του responsive design, καθώς οι κλάσεις και τα στοιχεία που περιέχει βοηθούν το περιεχόμενο και την δομή των ιστοσελίδων να διαμορφώνονται με τον κατάλληλο τρόπο έτσι ώστε να είναι προσαρμόσιμα σε κάθε είδους μεγέθους οθόνης.

Επίσης διευκολύνει με τις κλάσεις που περιέχει την ανάπτυξη των ιστοσελίδων από την μεριά του δημιουργού, ενώ επίσης η εκμάθηση του δεν είναι μια διαδικασία που απαιτεί πολύ χρόνο. Επιπλέον, το Bootstrap είναι συμβατό με όλους τους σημερινούς διαθέσιμους web browsers, κάτι που κάνει την χρήση του ακόμη πιο βολική.

5 Δημιουργία Αρχικής Σελίδας

Σε αυτό το μέρος θα αναλυθεί η σχεδίαση της αρχικής σελίδας, η οποία είναι η πρώτη σελίδα που εμφανίζεται στον χρήστη όταν επιλέξει να επισκεφτεί αυτό το website. Επίσης θα περιγραφούν και πολλές από τις βασικές αρχές που αφορούν την συγγραφή κώδικα HTML και CSS που χρησιμοποιήσαμε και για την δημιουργία και των επόμενων σελίδων.

5.1 Εισαγωγή – Αρχικές Δηλώσεις στον κώδικα HTML

Αρχικά θα εξετάσουμε τον κώδικα της επικεφαλίδας head, τις διάφορες δηλώσεις που υπάρχουν σε αυτό και τον τρόπο που συνδέουμε την HTML με τα άλλα αρχεία.

```
7 <html lang="el">
8   <head>
9     <title>CUT Info</title>
10    <link rel="shortcut icon" href="images/favicon.ico">
11    <meta charset="UTF-8">
12    <meta name="viewport" content="width=device-width, initial-scale=1.0">
13
14    <!-- Latest compiled and minified CSS -->
15    <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">
16    <!-- links for CSS files -->
17    <link rel="stylesheet" type="text/css" href="CSS_files/home.css">
18    <link rel="stylesheet" type="text/css" href="CSS_files/home_responsive.css" media="screen and (max-width: 750px)">
19
20    <!-- jQuery library -->
21    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
22    <!-- Latest compiled JavaScript -->
23    <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>
24    <!-- links for js files -->
25    <script src="js_files/animation_scroll.js"></script>
26  </head>
```

Διάγραμμα 5: Κώδικας επικεφαλίδας head

Αρχικά ο κώδικας ξεκινά με το στοιχείο <html> με την ιδιότητα lang να έχει την τιμή el κάτι που δηλώνει ότι σε αυτό το έγγραφο χρησιμοποιείται η Ελληνική γλώσσα, και στην συνέχεια με τον ορισμό του στοιχείου <title> που δηλώνει τον τίτλο που θα έχει η σελίδα που θα εμφανίζεται στο πάνω μέρος του web browser.

Με το στοιχείο <link> ορίζουμε ένα εξωτερικό σύνδεσμο, δηλαδή ένα αρχείο που βρίσκεται κάπου αλλού αποθηκευμένο και θα επικοινωνεί με αυτό τον HTML κώδικα. Αυτό γίνεται με την ιδιότητα href η οποία η τιμή της αντιπροσωπεύει την τοποθεσία που θα πρέπει να αναφερθεί ο κώδικας για να βρει αυτά τα αρχεία. Με την ιδιότητα rel ορίζουμε ένα εικονίδιο το οποίο χαρακτηρίζει την σελίδα και βρίσκεται δίπλα από τον τίτλο της.

Στην συνέχεια έχουμε το στοιχείο <meta> και τον κώδικα charset="UTF-8" πληροφορούμε τον web browser για το σύνολο των χαρακτήρων και των συμβόλων που χρησιμοποιούνται στην σελίδα. Επίσης χρησιμοποιώντας τον κώδικα:

```
name="viewport" content="width=device-width, initial-scale=1.0">
```

δηλώνουμε στον web browser να ρυθμίζει το πλάτος της σελίδας έτσι ώστε να συμβαδίζει με το μέγεθος της οθόνης που εμφανίζεται η σελίδα, ενώ επίσης θέτουμε το επίπεδο μεγέθυνσης σε 1, έτσι ώστε ο χρήστης να μην χρειάζεται να μεγεθύνει την σελίδα για να διαβάσει το περιεχόμενο της.

Ακόμη, όπως προαναφέραμε με το στοιχείο <link> συνδέουμε τον κώδικα HTML με άλλα εξωτερικά αρχεία, κάτι που γίνεται και με τις βιβλιοθήκες του Bootstrap. Συγκεκριμένα για τα αρχεία Bootstrap συμπεριλαβαίναμε τα αρχεία αυτά χρησιμοποιώντας ένα δίκτυο μεταφοράς περιεχομένου που συμβάλλει στην γρήγορη μεταφορά δεδομένων στο διαδίκτυο, λαμβάνοντας υπόψη την γεωγραφική θέση του χρήστη, κάνοντας έτσι πιο γρήγορη την μεταφορά δεδομένων και κατά συνέπεια την γρηγορότερη φόρτωση των σελίδων. Για τα υπόλοιπα αρχεία απλά αναφέραμε την τοποθεσία τους βάση του φακέλου που βρίσκονται στο τοπικό σύστημα. Τα αρχεία CSS ξεχωρίζουν με τις ιδιότητες rel="stylesheet" type="text/css", ενώ για τα αρχεία JavaScript χρησιμοποιείται το στοιχείο <script> και η ιδιότητα <src> για να δηλώσουμε την τοποθεσία τους.

Οι δηλώσεις και οι ορισμοί αυτοί ισχύουν σε κάθε έγγραφο HTML με την διαφορά να έγκειται στο ποια αρχεία συνδέονται κάθε φορά με τον HTML κώδικα για κάθε σελίδα που υπάρχει.

5.2 Οθόνη Καλωσορίσματος

5.2.1 Σκοπός



Διάγραμμα 6: Οθόνη Καλωσορίσματος

Ο σκοπός αυτής της σελίδας είναι να καλωσορίζει τον χρήστη στο website με ένα μήνυμα στο οποίο αναγράφεται συνοπτικά το τι περιέχει το website που μόλις έχουν επισκεφτεί. Στόχος είναι να δημιουργήσει το ενδιαφέρον στον χρήστη και να τον κάνει να θέλει να εξερευνήσει περισσότερα.

5.2.2 Κώδικας HTML

```
<div class="container-fluid intro text-center">  
  <div class="intro_msg">  
    <h1>Καλωσορίσατε στο CUT Info</h1>  
    <p> Ένα μέρος όπου μπορείτε να βρείτε πολλές χρήσιμες πληροφορίες για το  
      Τεχνολογικό Πανεπιστήμιο Κύπρου, όπως φωτογραφίες, χάρτες και άλλα!</p>  
    <a href="mainpage_gr.html" class="btn btn-default btn-lg mybutton">ΕΙΣΟΔΟΣ</a>  
  </div>  
</div>  
  
<!-- link for user to scroll down -->  
<div class="container-fluid navigation text-center">  
  <p class="learn_more">Δείτε περισσότερα</p>  
  <a href="#more_info">  
    <span class="glyphicon glyphicon-chevron-down" id="down_arrow"></span>  
  </a>  
</div>
```

Διάγραμμα 7: Κώδικας HTML

Με τον παραπάνω κώδικα δημιουργούμε το μήνυμα καλωσορίσματος για τον χρήστη. Με το στοιχείο `<div>` ορίζουμε ένα τμήμα του εγγράφου το οποίο ξεχωρίζει από τα υπόλοιπα. Μέσα σε αυτό ορίσαμε τα ονόματα των κλάσεων που θα έχει. Οι κλάσεις είναι ιδιότητες των στοιχείων που μπορούν να χρησιμοποιούνται σε πολλά στοιχεία για να χαρακτηρίζουν την συμπεριφορά τους. Η κλάση `container-fluid` αποτελεί κλάση του Bootstrap και δηλώνει ότι το συγκεκριμένο τμήμα θα παίρνει την έκταση όλης της οθόνης. Η άλλη επιλογή κλάσης από το Bootstrap που θα μπορούσαμε να έχουμε στην συγκεκριμένη περίπτωση είναι η κλάση `container` η οποία παίρνει συγκεκριμένο, προκαθορισμένο μέρος της οθόνης.

Με το στοιχείο `<h1>` ορίζουμε την επικεφαλίδα και με το `<p>` την παράγραφο που την ακολουθεί. Στην συνέχεια τοποθετούμε ένα κουμπί το οποίο όταν ο χρήστης πατήσει θα τον μεταφέρει στην επόμενη σελίδα του website. Το στυλ του καθορίζεται με τις κλάσεις `btn`, `btn-default`, `btn-lg` του Bootstrap και την κλάση `mybutton` που ορίζουμε εμείς. Η τιμή της ιδιότητας `href` καθορίζει την τοποθεσία που θα μεταβεί η ιστοσελίδα όταν πατηθεί το κουμπί, ενώ με το στοιχείο `<a>` δηλώνουμε στο έγγραφο HTML ότι αυτό το τμήμα του κώδικα αποτελεί σύνδεσμο.

Αξίζει επίσης να σημειωθεί ότι το μέρος του κειμένου που βρίσκεται μεταξύ των συμβόλων `<!-- και -->` είναι σχόλια που χρησιμοποιούνται για να διευκολύνουν τον προγραμματιστή στο να κρατά σημειώσεις για το τι αφορούν ορισμένα σημεία κώδικα.

5.2.3 Κώδικας CSS

```
.intro{
  padding: 90px 50px;
}

.intro h1{
  font-family: "Montserrat", sans-serif;
  letter-spacing: 0.6px;
  font-size: 50px;
}

.intro_msg p{
  margin: 50px auto;
  font-size: 21px;
  font-weight: bold;
  width: 60%;
  font-family: 'Nunito', sans-serif;
}

.mybutton {
  background: none;
  border-color: black;
  border-width: 1.5px;
  border-radius: 0px;
  font-size: 17px;
  padding: 13px 26.5px;
  -webkit-transition-duration: 0.6s;
  transition-duration: 0.6s;
}

.mybutton:hover , .mybutton:focus{
  background-color: #FFE571;
  border-radius: 30px;
  box-shadow: 6px 6px 25px grey;
}
```

Διάγραμμα 8: Κώδικας CSS για οθόνη καλωσορίσματος

Με την χρήση της CSS και συγκεκριμένα του αρχείου home.css διαμορφώνουμε την εμφάνιση με την οποία θα παρουσιάζονται τα διάφορα στοιχεία του HTML εγγράφου. Αυτό γίνεται βάση των κλάσεων που ορίσαμε να έχουν τα στοιχεία HTML και βάση τις τιμές που δίνουμε στις ιδιότητες τους στις διάφορες δηλώσεις που κάνουμε στον κώδικα CSS. Οι δηλώσεις ξεχωρίζουν μεταξύ τους με το σύμβολο ‘;’. Για παράδειγμα στην πιο πάνω περίπτωση, η κλάση intro αντιπροσωπεύει ολόκληρο το κομμάτι που αφορά το μήνυμα καλωσορίσματος και με την δήλωση padding: 90px 50px; καθορίζεται η απόσταση που θα έχει το περιεχόμενο του στοιχείου σε σχέση με τα όρια του. Η τιμή 90px δηλώνει την απόσταση από την πάνω και την κάτω πλευρά, ενώ η τιμή 50px την απόσταση από την δεξιά και αριστερή μεριά.

Στην συνέχεια διαμορφώνουμε την εμφάνιση της επικεφαλίδας και της παραγράφου επιλέγοντας τα στοιχεία του HTML κώδικα ανάλογα(.intro h και .intro_msg p αντίστοιχα). Με τις ιδιότητες font-family και font-size ορίζουμε το είδος της γραμματοσειράς και το μέγεθος των γραμμάτων αντίστοιχα, ενώ με την ιδιότητα font-weight και την τιμή bold δηλώνουμε ότι τα γράμματα θα είναι σε έντονη μορφή(bold). Στην περίπτωση της ιδιότητας font-family, ορίζουμε περισσότερες τιμές έτσι ώστε στην περίπτωση που κάποια γραμματοσειρά δεν υποστηρίζεται από κάποιο web browser η ιδιότητα παίρνει την αμέσως επόμενη τιμή που ακολουθεί. Με την ιδιότητα letter-spacing ορίζεται η απόσταση μεταξύ των γραμμάτων στην επικεφαλίδα <h1>.

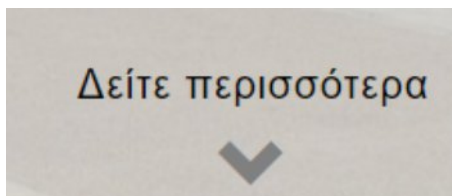
Για την παράγραφο που ακολουθεί ορίζουμε το πλάτος που θα καταλαμβάνει σε στην οθόνη και συγκεκριμένα χρησιμοποιώντας την ιδιότητα width και δίνοντας σε αυτή την τιμή 60%. Τέλος με την ιδιότητα margin ορίζεται η απόσταση που θα έχει κάποιο στοιχείο σε σχέση με τα άλλα γειτονικά του στοιχεία στην σελίδα. Αυτή η ιδιότητα διαφέρει με την ιδιότητα padding στο ότι η δεύτερη αφορά την εσωτερική απόσταση που έχει ένα στοιχείο σε σχέση με τα όρια του, ενώ το margin αφορά την απόσταση του σε σχέση με άλλα, εξωτερικά στοιχεία. Πολλές φορές όμως και οι δύο ιδιότητες μπορούν να έχουν το ίδιο αποτέλεσμα. Στην περίπτωση αυτή, με την δήλωση margin: 50px auto ορίσαμε την απόσταση που θα έχει η παράγραφος από την επικεφαλίδα και το κουμπί που ακολουθεί. Με την τιμή 50px ορίσαμε την πάνω και κάτω απόσταση, ενώ με την τιμή auto δηλώνουμε ότι η απόσταση από αριστερά και δεξιά θα είναι ίση και συγκεκριμένα αυτό, τοποθετεί την παράγραφο στο κέντρο της οθόνης.

Επίσης, με την κλάση `text-center` που ορίσαμε στον κώδικα HTML δηλώνουμε ότι το κείμενο που ακολουθεί θα έχει κεντρική στοίχιση.

Έπειτα με τις δηλώσεις για την κλάση `mybutton` ορίζουμε την εμφάνιση που θα έχει το κουμπί που φαίνεται στην οθόνη. Με την ιδιότητα `background` και την τιμή `none`, δηλώνουμε ότι δεν θα υπάρχει κάποιο φόντο για το κουμπί αυτό, ενώ με τις διάφορες παραλλαγές της ιδιότητας `border` καθορίζουμε την εμφάνιση του περιγράμματος που υπάρχει γύρω από το κουμπί, όπως για παράδειγμα το χρώμα του (`border-color`), το πάχος του περιγράμματος (`border-width`) και το αν θα έχουν κάποια κλίση οι γωνίες τους (`border-radius`). Δείκτης για μετακίνηση προς τα κάτω

5.2.3.1 Κώδικας HTML

Όπως βλέπουμε στο διάγραμμα 7 μετά από το μήνυμα καλωσορίσματος υπάρχει προς το τέλος της οθόνης ένα τόξο το οποίο αν ο χρήστης πατήσει θα μετακινηθεί προς την συνέχεια της σελίδας. Όταν ο χρήστης περάσει το δρομέα του ποντικιού πάνω από αυτό το τόξο θα του εμφανιστεί το μήνυμα «Δείτε περισσότερα», όπως φαίνεται στην παρακάτω εικόνα. Ο στόχος είναι να παρακινήσει τον χρήστη να συνεχίσει στην συνέχεια της σελίδας για να δει το περιεχόμενο που ακολουθεί.



Διάγραμμα 9: Δείκτης/Σύνδεσμος

Όπως και στην περίπτωση με το μήνυμα καλωσορίσματος, ορίζουμε ένα τμήμα `<div>` με την κλάση `container-fluid` για να επεκτείνεται σε όλο το πλάτος της οθόνης. Επίσης ορίζουμε το κείμενο που θα εμφανίζεται με το στοιχείο `<p>` και στην συνέχεια ορίζουμε τον σύνδεσμο που θα μεταφέρει τον χρήστη στην συνέχεια της σελίδας. Το εικονίδιο τοποθετείται χρησιμοποιώντας ένα από τα `glyphicons`, τα οποία είναι έτοιμα εικονίδια που προσφέρει το Bootstrap. Το εικονίδιο αυτό περικλείεται γύρω από το στοιχείο `span` το οποίο χρησιμοποιείται για να ξεχωρίσουμε κάποιο τμήμα του εγγράφου από τα άλλα. Επίσης με την ιδιότητα `id` ορίζουμε μια τιμή (`id="down_arrow"`) η οποία είναι μοναδική για αυτό το στοιχείο, όπου σε αντίθεση με την κλάση (ιδιότητα `class`), δεν μπορεί να χρησιμοποιηθεί για άλλο στοιχείο μέσα στην σελίδα.

5.2.3.2 Κώδικας CSS

```
.learn_more{
  visibility: hidden;
  color: black;
  font-size: 20px;
  letter-spacing: 0.7px;
  font-family: 'Nunito', sans-serif;
  margin: 8px auto;
}

.navigation{
  margin-top: 14px;
}

.navigation .glyphicon {
  text-decoration: none;
  font-size: 35px;
  color: gray;
}
```

Διάγραμμα 10: Κώδικας CSS για τον δείκτη/σύνδεσμο

Η κλάση `learn_more` αντιπροσωπεύει το κείμενο «Δείτε περισσότερα» το οποίο με την ιδιότητα `visibility` και την τιμή `hidden` στην αρχή δεν φαίνεται στην οθόνη, μέχρι να περάσει ο χρήστης τον δρομέα του ποντικιού πάνω από το εικονίδιο, κάτι που υλοποιείται με JavaScript που θα δούμε στην συνέχεια. Στην συνέχεια διαμορφώνουμε την εμφάνιση αυτού του μικρού κειμένου όπως και την τοποθεσία του στην οθόνη αυτού και του εικονιδίου.

5.2.3.3 Κώδικας JavaScript

Με κώδικα JavaScript που βρίσκεται στο αρχείο `animation_scroll` υλοποιούμε το κομμάτι του κώδικα που ενεργοποιείται όταν ο χρήστης περάσει τον δρομέα του ποντικιού του πάνω από το εικονίδιο/τόξο με κατεύθυνση προς τα κάτω.

```
$("#down_arrow").hover(function(){
  $(".learn_more").removeClass("hide_text");
  $(".learn_more").addClass("show_text");
}, function(){
  $(".learn_more").addClass("hide_text");
});
```

Διάγραμμα 11: Συναρτήσεις JQuery

Με την χρήση της συνάρτησης της JQuery `hover`, επιτυγχάνουμε τον πιο πάνω σκοπό. Συγκεκριμένα επιλέγουμε το στοιχείο `#down_arrow` και ορίζουμε σε αυτό την συνάρτηση `hover`. Συγκεκριμένα όταν ο δείκτης του ποντικιού περνά πάνω από το εικονίδιο τότε ενεργοποιείται ο κώδικας που βρίσκετε ανάμεσα στο πρώτο ζευγάρι των συμβόλων `{ }` ενώ

όταν ο δείκτης του ποντικιού φεύγει από το εικονίδιο ενεργοποιείται το δεύτερο κομμάτι της συνάρτησης. Με τις χρήσεις των συναρτήσεων `addClass` και `removeClass` προσθέτουμε και αφαιρούμε αντίστοιχα τις κλάσεις που χρειάζονται τα στοιχεία για να πραγματοποιείται το animation με το οποίο εμφανίζεται και εξαφανίζεται το κείμενο «Δείτε περισσότερα». Περισσότερα όμως για τα animations αυτά θα εξεταστούν στην συνέχεια σε επόμενο κεφάλαιο.

5.2.4 Πάνω μέρος οθόνης καλωσορίσματος – Μπάρα πλοήγησης

Στο πάνω μέρος της οθόνης, όπως φαίνεται και στο Διάγραμμα 6, έχουμε μια μπάρα πλοήγησης όπου αριστερά υπάρχει το εικονίδιο του website αυτού μαζί με το όνομα του website, ενώ στα δεξιά υπάρχει ένας κατάλογος με τον οποίο ο χρήστης θα μπορεί να επιλέγει την γλώσσα της επιλογής του.

5.2.4.1 Κώδικας HTML

```
<!-- navigation bar of the page -->
<nav class="navbar navbar-inverse nav_desk">
  <div class="container-fluid">
    <div class="navbar-header">
      
      <a class="navbar-brand" href="index.html">CUIT Info</a>
    </div>

    <ul class="nav navbar-nav navbar-right">
      <li class="dropdown">
        <a class="dropdown-toggle" data-toggle="dropdown" data-target=".dropdown-menu" href="#">
          <span class="glyphicon glyphicon-globe globeicon"></span> Language <span class="caret"></span>
        </a>
        <ul class="dropdown-menu">
          <li><a href="index_en.html"><span style="margin-left: 20px;"> English</span></a></li>
          <li><a href="index.html"><span class="glyphicon glyphicon-ok"></span> Greek</a></li>
        </ul>
      </li>
    </ul>
  </div>
</nav>
```

Διάγραμμα 12: Μπάρα πλοήγησης αρχικής σελίδας - Κώδικας HTML

Με την χρήση των κλάσεων `navbar` `navbar-inverse` του Bootstrap αλλά και των υπόλοιπων κλάσεων που ακολουθούν δημιουργούμε μια μπάρα πλοήγησης η οποία τοποθετείται στο πάνω μέρος της οθόνης.

Το τμήμα που περιέχεται μέσα στην κλάση `navbar-header` καθορίζει την επικεφαλίδα της μπάρας που βρίσκεται στο πάνω μέρος αριστερά της οθόνης που αποτελείται από το κείμενο που αντιπροσωπεύει το όνομα του website και το εικονίδιο του website που δηλώνεται με το στοιχείο `img`.

Στην συνέχεια έχουμε το τμήμα με τις κλάσεις `navbar-nav` και `navbar-right` που αντιπροσωπεύει το δεξί κομμάτι της μπάρας. Με την κλάση `navbar-right` στοιχίζεται το περιεχόμενο αυτό στα δεξιά της μπάρας. Έπειτα έχουμε την κλάση `dropdown` με την οποία δημιουργείται ένα τμήμα που όταν ο χρήστης πατήσει πάνω του θα ανοίξει μπροστά του ένας κατάλογος που θα περιέχει τις επιλογές που θα έχει στην διάθεση του ο χρήστης για την επιλογή γλώσσας.

Για να ανοίξει ο κατάλογος/λίστα ο χρήστης πατά πάνω στον σύνδεσμο που δημιουργείται με την κλάση `dropdown-toggle` που αποτελείται από την λέξη `Language` και ένα σύμβολο ενός τόξου με κατεύθυνση προς τα κάτω που δημιουργείται με την κλάση `caret`. Με την ιδιότητα `data-toggle` και την τιμή `dropdown` δηλώνουμε ότι αυτό που θα ανοίξει όταν ο χρήστης πατήσει πάνω του αποτελεί κατάλογο και με την ιδιότητα `data-target` ορίζουμε το όνομα της κλάσης του καταλόγου που θα ανοίξει.

Ο κατάλογος δηλώνεται με την κλάση `dropdown-menu` μέσα στο στοιχείο `` το οποίο είναι υπεύθυνο για την δημιουργία λιστών, με τα στοιχεία `` να αποτελούν τα κομμάτια της λίστας (επιλογή για αγγλική και επιλογή για ελληνική γλώσσα).

5.2.4.2 Κώδικας CSS

```
#logo{
  float:left;
  height: 50px;
  width:50px;
}

.navbar-inverse{
  padding: 5px;
  background: none;
  border: 0;
}

.navbar-header a{
  font-family: "Montserrat", sans-serif;
  padding-left: 25px;
  color: black !important;
  font-weight: bold;
  letter-spacing: 1px;
  font-size: 13pt;
}

.navbar-nav li a{
  font-family: "Montserrat", sans-serif;
  color: black !important;
  font-size: 13pt;
}
```

Διάγραμμα 13: Κώδικας CSS για μπάρα πλοήγησης

Πιο πάνω βλέπουμε τον κώδικα με τον οποίο ρυθμίζει το στυλ και την εμφάνιση της μπάρας πλοήγησης. Η ιδιότητα `#logo` αντιπροσωπεύει την εικόνα που βρίσκεται στα αριστερά της μπάρας και με τις ιδιότητες που δηλώνουμε την θέτουμε να είναι βρίσκεται αριστερά από το κείμενο που ακολουθεί (`float: left;`) ενώ επίσης δηλώνουμε και το μέγεθος που καταλαμβάνει στην οθόνη.

Επίσης, στην κλάση `navbar-inverse` θέτουμε την ιδιότητα `border` να είναι ίση με 0 έτσι ώστε η μπάρα πλοήγησης να μην έχει κάποιο περίγραμμα, επιτυγχάνοντας έτσι να φαίνεται σαν να είναι μέρος του υπόλοιπου περιεχομένου της οθόνης αυτής.

5.2.5 Εικόνα φόντου (Background Image)

Τέλος ολόκληρος ο κώδικας HTML που αφορά την οθόνη καλωσορίσματος βρίσκεται μέσα στο τμήμα με την κλάση `backgroundImg`, στην οποία κάνουμε χρησιμοποιώντας CSS τις απαραίτητες δηλώσεις έτσι ώστε να τοποθετηθεί μια εικόνα πίσω στο φόντο της οθόνης. Παρακάτω βλέπουμε τον κώδικα CSS με τον οποίο επιτυγχάνεται αυτό:

```
.backgroundImg{  
    height: auto;  
    background-image: url("../images/tepak_bck0_33.JPG");  
    background-repeat: no-repeat;  
    background-position: center center;  
    background-size: 100% 100%;  
}
```

Διάγραμμα 14: Κώδικας CSS για εικόνα φόντου

Με την ιδιότητα `height` και την τιμή `auto` ορίζουμε το ύψος του τμήματος αυτού να είναι αυτόματο, να προσαρμόζεται δηλαδή πάντα ανάλογα με τον χώρο που καταλαμβάνουν τα περιεχόμενα του.

Στην συνέχεια βλέπουμε την κλάση `background` και τις διάφορες παραλλαγές της που ορίζουν την εικόνα φόντου και την τοποθεσία της.

Με την ιδιότητα `background-image` ορίζουμε το ποια θα είναι η εικόνα που θα χρησιμοποιηθεί για αυτό το σκοπό, και την τοποθεσία που βρίσκεται αποθηκευμένη.

Με την ιδιότητα `background-repeat` και την τιμή `no-repeat` η εικόνα δεν επαναλαμβάνεται.

Με την ιδιότητα `background-position` θέτουμε το που θα τοποθετηθεί η εικόνα. Χωρίς να ορίσουμε κάτι η εικόνα θα τοποθετηθεί στην πάνω-αριστερή πλευρά, και για αυτό τον λόγο θέτουμε την τιμή `center center` για να τοποθετηθεί στο κέντρο του τμήματος που αντιπροσωπεύει.

Τέλος, με την ιδιότητα `background-size` και την τιμή `100% 100%` θέτουμε την εικόνα να καλύπτει ολόκληρο το μέρος του τμήματος που βρίσκεται τόσο σε ύψος όσο και σε πλάτος.

5.3 Οθόνη για γενικές πληροφορίες Τεπακ

Σε αυτή την οθόνη φαίνονται κάποιες γενικές συνοπτικές πληροφορίες για το Τεπακ. Ο χρήστης μπορεί να ενημερωθεί γενικά για το τι είναι το Τεπακ, και στην συνέχεια να συνεχίσει παρακάτω, για να δει τι περιλαμβάνει ακριβώς αυτό το website. Σε αυτό το σημείο αναλύσαμε το πως λειτουργεί το Grid System του Bootstrap, το οποίο είναι ένα σύστημα που μας βοηθά να δημιουργούμε όμορφες και ευέλικτες για κάθε συσκευή διατάξεις. Στην συνέχεια θα δούμε περισσότερες λεπτομέρειες για αυτό, καθώς και το πως το χρησιμοποιήσαμε για την δημιουργία αυτού του website.



Διάγραμμα 15: Οθόνη Γενικών πληροφοριών

5.3.1 Grid System Bootstrap

Το Bootstrap περιλαμβάνει ένα πολύ χρήσιμο σύστημα το οποίο μας επιτρέπει να δημιουργούμε διατάξεις χωρίζοντας το περιεχόμενο μιας σελίδας σε γραμμές και στήλες που αλλάζουν δυναμικά ανάλογα με το μέγεθος της οθόνης που εμφανίζονται.

Το σύστημα αυτό αποτελείται κυρίως από συγκεκριμένες ομάδες περιεχομένου (μια από τις κλάσεις container και container-fluid που είδαμε στο υποκεφάλαιο 5.2.2), από γραμμές και από στήλες.

Οι γραμμές αντιπροσωπεύονται από την κλάση row με την οποία ομαδοποιούνται σε μια οριζόντια γραμμή οι στήλες του περιεχομένου και στοιχίζονται ανάλογα. Κάθε καινούργια κλάση row δηλώνει ότι έχουμε μια καινούργια γραμμή.

Πιο κάτω βλέπουμε μια εικόνα που δείχνει το πως χωρίζονται οι στήλες μέσα σε μια σελίδα:

Διάγραμμα 16: Grid system bootstrap

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Πηγή 1: Bootstrap Grid System, w3schools.com

Ο μέγιστος αριθμός στηλών που μπορεί να έχει μια γραμμή είναι δώδεκα. Σε περίπτωση όμως που χρειαζόμαστε λιγότερες από δώδεκα στήλες, μπορούμε να ομαδοποιήσουμε πολλές στήλες σε μια, όπως φαίνεται και στην παραπάνω εικόνα. Αν ξεπεράσουμε το όριο των 12 στηλών σε μια γραμμή, τότε αυτόματα θα δημιουργηθεί καινούργια γραμμή.

Το Bootstrap χρησιμοποιεί συγκεκριμένες κλάσεις για να καθορίζεται η διάταξη των γραμμών και των στηλών ανάλογα με το μέγεθος της οθόνης που εμφανίζεται μια ιστοσελίδα. Για παράδειγμα, στην περίπτωση που έχουμε μια γραμμή με τέσσερις στήλες, για μεγάλες οθόνες επιθυμούμε να εμφανίζονται αυτές οι στήλες σε μια γραμμή, ενώ για τις μικρές οθόνες λόγω του ότι ο χώρος είναι μικρότερος επιθυμούμε οι στήλες να στοιβάζονται η μια πάνω στην άλλη.

Η βασική σύνταξη μιας τέτοιας κλάσης είναι η εξής: `col - * - *`

Το πρώτο '*' που συναντούμε αναφέρεται στο είδος της συσκευής που επιθυμούμε να εφαρμοστεί αυτή η κλάση. Οι τέσσερις πιθανές τιμές αντιπροσωπεύουν ένα είδος μεγέθους οθόνης: `xs` για μικρές οθόνες κινητών τηλεφώνων, `sm` για συσκευές tablet, `md` για οθόνες ηλεκτρονικών υπολογιστών (επιτραπέζιων ή φορητών) και `lg` για μεγάλες οθόνες. Η κλάση αυτή εφαρμόζεται σε συσκευές μεγαλύτερες ή ίσες του μεγέθους της οθόνης της συσκευής που αναφέρεται η τιμή αυτή.

Το δεύτερο '*' αναφέρεται στο πόσες στήλες θα ομαδοποιηθούν για εκείνο το τμήμα της γραμμής.

Παράδειγμα:

Έστω ότι θέλουμε να δημιουργήσουμε μια διάταξη που να καταλαμβάνει όλη την έκταση της οθόνης και να αποτελείται από μια γραμμή και τρεις στήλες για συσκευές με μέγεθος οθόνης

tablet και πάνω, και μια στήλη για οθόνες κινητών τηλεφώνων και κάτω. Τότε ο κώδικας μας θα έχει ως ακολούθως:

```
<div class="container-fluid">
  <div class="row">
    <div class="col-sm-4"> 1st column </div>
    <div class="col-sm-4"> 2nd column </div>
    <div class="col-sm-4"> 3rd column </div>
  </div>
</div>
```

Διάγραμμα 17: Παράδειγμα κώδικα Grid System Bootstrap

Άλλες δυνατότητες που μας προσφέρονται από το Bootstrap είναι να μετακινούμε στήλες δεξιά χρησιμοποιώντας την κλάση offset, αυξάνοντας έτσι το αριστερό περιθώριο μεταξύ δύο στηλών ή να αλλάζουμε την σειρά μεταξύ στηλών χρησιμοποιώντας τις κλάσεις push(για μετακίνηση τους δεξιά) και pull(για μετακίνηση αριστερά).

5.3.2 Κώδικας HTML οθόνης με γενικές πληροφορίες

```
<!-- More info about Tepak -->
<div class="container-fluid tepak_info" id="more_info">
  <div class="slideanim text-center">
    <h1>
      Τεχνολογικό Πανεπιστήμιο Κύπρου
    </h1>
    <hr class="hrstyle1">
    <div class="row">
      <div class="col-sm-5 col-sm-push-7">
        
      </div>
      <div class="col-sm-6 col-sm-pull-4">
        <p>
          Τοποθετημένο στην Λεμεσό, το Τεχνολογικό Πανεπιστήμιο Κύπρου (γνωστό και ως Τεπακ) είναι εν
          πανεπιστήμιο της Κύπρου και με την συνεχή ανάπτυξη του με το πέρασμα των χρόνων στοχεύει
          στους φοιτητές του μόρφωση υψηλού επιστημονικού, τεχνολογικού και επαγγελματικού επιπέδου.
        </p>
      </div>
    </div>
  </div>
</div> <!-- End of slideanim class -->
</div>
```

Διάγραμμα 18: Κώδικας HTML για οθόνη με γενικές πληροφορίες

Με αυτό τον κώδικα δημιουργήσαμε την δομή της οθόνης αυτής. Αρχικά δημιουργήσαμε ένα πλαίσιο το οποίο καλύπτει ολόκληρη την οθόνη με την κλάση container-fluid στο οποίο μέσα θα είχαμε τον τίτλο με μια επικεφαλίδα <h1>. Στην συνέχεια βάση του Grid System του Bootstrap δημιουργήσαμε μια γραμμή με δύο στήλες όπου στην μια θα περιέχεται το κείμενο

και στην άλλη η εικόνα. Για μικρές συσκευές οι δύο στήλες θα σχηματίζουν μια κάθετη γραμμή όπου θα στοιβάζονται η μια πάνω στην άλλη, με το κείμενο να προηγείται.

Για να δηλώσουμε την εικόνα χρησιμοποιούμε το στοιχείο `` με το οποίο ορίζουμε εικόνες στα HTML έγγραφα. Με την ιδιότητα `src` δηλώνουμε την τοποθεσία που βρίσκεται αποθηκευμένη η εικόνα για να την βρει ο web browser και με την ιδιότητα `alt` δηλώνουμε ένα μικρό κείμενο το οποίο θα εμφανίζεται σε περίπτωση που για κάποιο λόγο δεν φορτωθεί η εικόνα στην σελίδα. Στην συνέχεια βλέπουμε τις κλάσεις `img-responsive` και `img-circle` του Bootstrap, όπου η μια χρησιμοποιείται για να σιγουρεύει ότι η φωτογραφία θα εμφανίζεται σωστά σε όλα τα μεγέθη οθονών και δεν θα εξέχει προς τα έξω (αναγκάζοντας τον χρήστη να μετακινηθεί πλάγια στην οθόνη του για να δει την εικόνα), ενώ η δεύτερη σχηματίζει ένα κυκλικό περίγραμμα γύρω από την φωτογραφία.

Επίσης, με το στοιχείο `<hr>` τοποθετούμε μια διαχωριστική γραμμή για να δίνουμε την αίσθηση του διαχωρισμού των δύο τμημάτων της οθόνης.

5.3.3 Κώδικας CSS οθόνης με γενικές πληροφορίες

Στην συνέχεια βλέπουμε τον CSS κώδικα που χρησιμοποιήθηκε για να ορίσουμε την εμφάνιση που θα έχουν τα στοιχεία της παραπάνω οθόνης:

```
.tepak_info{
  height: auto;
  background-color: #FFE571;
}
.tepak_info h1{
  font-family: "Montserrat", sans-serif;
  margin-top: 8%;
  font-size: 46px;
  text-shadow: 2px 2px 6px grey;
}
hr.hrstyle1{
  width: 26%;
  height: 2.5px;
  margin: 45px auto;
  border: 0 none;
  background-image: linear-gradient(to right, rgba(0,0,0,0), rgba(0,0,0,0.75), rgba(0,0,0,0));
}
.tepak_info p{
  margin: 12% auto;
  width: 95%;
  font-family: 'Nunito', sans-serif;
  font-size: 16pt;
}
.tepak_img{
  display: block;
  width: 50%;
  margin: 10% auto;
  box-shadow: 5px 5px 20px #333332;
```

Διάγραμμα 19: Κώδικας CSS για οθόνη γενικών πληροφοριών

Μια νέα ιδιότητα που χρησιμοποιούμε είναι η ιδιότητα `text-shadow` που λειτουργεί με παρόμοιο τρόπο με την ιδιότητα `box-shadow` με την διαφορά αυτή η ιδιότητα χρησιμοποιείται για να σκιάζει κείμενο.

Επίσης παρατηρούμε και τον κώδικα για τον χειρισμό της εμφάνισης της διαχωριστικής γραμμής με την δήλωση `hr.hrstyle{ ... }`, όπου μεταξύ άλλων ορίσαμε το πλάτος, το ύψος, την ακριβή θέση της γραμμής.

5.4 Οθόνη για το τι περιλαμβάνει το website

Στην συνέχεια όταν ο χρήστης μετακινηθεί πιο κάτω στη σελίδα θα δει το τμήμα όπου αναγράφεται το τι να περιμένει ο χρήστης να περιέχει αυτό το website. Συνοπτικά, περιγράφεται στον χρήστη το τι περιλαμβάνει γενικά αυτό το website έτσι ώστε να του δοθεί μια γενική ιδέα για το είδος των πληροφοριών που υπάρχουν στην συνέχεια.



Διάγραμμα 20: Εμφάνιση οθόνης για το τι περιλαμβάνει το website

5.4.1 Κώδικας HTML

```
<!-- How it works section -->
<div class="container-fluid infoheader slideanim text-center">
  <h2>Τι περιλαμβάνει</h2>
  <hr class="hr_howitworks">
</div>

<div class="row row_section">
  <div class="slideanim">
    <div class="col-sm-6 col-sm-push-6">
      <h3>Πληροφορίες</h3>
      <hr class="hrstyle2">
      <p>Βρείτε πολλές πληροφορίες για το Τεπακ, που αφορούν την ιστορία του,
        την αποστολή του και πολλά άλλα.</p>
    </div>
    <div class="col-sm-6 col-sm-pull-6">
      
    </div>
  </div>
</div>

<div class="row row_section row2">
  <div class="slideanim">
    <div class="col-sm-6 section2">
      <h3>Φωτογραφίες και Χάρτες</h3>
      <hr class="hrstyle2">
      <p>Ένα μέρος όπου μπορείτε να βρείτε πολλές φωτογραφίες και χάρτες των τμημάτων
        και κτηρίων του Τεπακ.</p>
    </div>
    <div class="col-sm-6">
      
    </div>
  </div>
</div>

<div class="row row_section">
  <div class="slideanim">
    <div class="col-sm-6 col-sm-push-6">
      <h3>Σκεφτείτε και αποφασίστε</h3>
      <hr class="hrstyle2">
      <p>Ένα χρήσιμο εργαλείο σε φοιτητές του Τεπακ, αλλά κυρίως σε νέους που σκέφτονται να
        συνεχίσουν τις σπουδές τους στο Τεπακ.</p>
    </div>
    <div class="col-sm-6 col-sm-pull-6">
      <span class="glyphicon glyphicon-education education"></span>
    </div>
  </div>
</div>
```

Διάγραμμα 21: Κώδικας HTML για κομμάτι πληροφόρησης για website

Στην περίπτωση αυτής της οθόνης αφού πρώτα δημιουργήσαμε την επικεφαλίδα και ένα διαχωριστικό με το στοιχείο `<hr>`, στην συνέχεια δημιουργήσαμε τρεις γραμμές με δύο στήλες στην κάθε μια. Για τις οθόνες κινητών τηλεφώνων και κάτω οι δύο αυτές στήλες θα στοιβάζονται η μια πάνω στην άλλη.

Στην μια στήλη θα βρίσκεται το κείμενο που θα πληροφορεί τον χρήστη και στην άλλη ένα εικονίδιο που θα αντιπροσωπεύει αυτό που αναγράφεται στο κείμενο. Με τον τρόπο που χωρίσαμε την κάθε γραμμή φροντίσαμε έτσι ώστε η κάθε στήλη να παίρνει τον μισό χώρο της κάθε γραμμής χρησιμοποιώντας την κλάση col-sm-6 και τις κλάσεις push και pull για να αλλάζουμε την σειρά τους για τα μεγέθη οθονών που επιθυμούσαμε (col-sm-push-6 και col-sm-pull-6).

5.4.2 Κώδικας CSS

```

.row_section{
  padding: 65px 0px 65px 0px;
  background-color: #e6e6e6;
}

.row2{
  background-color: whitesmoke;
}

.row_section h3{
  text-transform: uppercase;
  letter-spacing: 1.85px;
  word-spacing: 7px;
  text-shadow: 1px 1px 3.5px #7a7a52;
  font-size: 37px;
  font-family: 'Century Gothic';
}

.row_section p{
  line-height: 1.40;
  letter-spacing: 0.5px;
  width: 60%;
  margin: 45px auto;
  font-size: 23px;
  font-family: 'Nunito', sans-serif;
}

hr.hrstyle2{
  width: 23%;
  margin: 30px auto;
  height: 5px;
  border: 0 none;
  color: #E9BD45;
  background-color: #E9BD45;
}

#img1, #img2, .education{ margin: 6.5% auto; }
#img1{ width: 25%; }
#img2{ width: 30%; }

.education{
  font-size: 180px;
  color: black;
}

```

Διάγραμμα 22: Κώδικας CSS για κομμάτι πληροφόρησης για website

Πιο πάνω βλέπουμε τον κώδικα CSS με τον οποίο ορίσαμε την εμφάνιση που θα έχουν οι διατάξεις της πιο πάνω οθόνης. Με την χρήση των κατάλληλων ιδιοτήτων δημιουργήσαμε τον επιθυμητό χώρο μεταξύ των στηλών(κυρίως με την ιδιότητα padding), το χρώμα του φόντου για την κάθε γραμμή, όπως και την επιθυμητή εμφάνιση των γραμμάτων και των επικεφαλίδων της κάθε γραμμής. Εδώ παρατηρούμε τρεις καινούργιες ιδιότητες, την ιδιότητα text-transform με την οποία αλλάζουμε τα γράμματα της επικεφαλίδας σε κεφαλαία, την ιδιότητα word-spacing με την οποία ορίζουμε μια απόσταση μεταξύ των λέξεων του στοιχείου και την ιδιότητα line-height που το ύψος της κάθε γραμμής στο κείμενο της παραγράφου.

Επίσης θέσαμε το κατάλληλο πλάτος στα εικονίδια και στην περίπτωση όπου το εικονίδιο προερχόταν από την κλάση `glyphicon` ελέγξαμε το μέγεθος του με την χρήση της ιδιότητας `font-size`.

5.5 Σχεδίαση σελίδας για άλλες συσκευές – εφαρμογή **responsive design**

5.5.1 **Media queries**

Η εφαρμογή των `media queries` είναι μια νέα τεχνική που εισήγαγε η καινούργια έκδοση της CSS, η CSS3. Σύμφωνα με αυτή την τεχνική μπορούμε να ορίζουμε διαφορετικούς κανόνες CSS για διαφορετικές συσκευές ή όταν το website εμφανίζεται από διαφορετικά μεγέθη οθονών. Αυτό γίνεται με το να θέτουμε κάποιες συνθήκες οι οποίες αν πληρούνται τότε εφαρμόζονται οι κανόνες που αντιστοιχούν.

Με αυτό τον τρόπο επιτυγχάνονται οι στόχοι της τεχνικής του `responsive design` καθώς μπορούμε να ορίζουμε διαφορετικούς κανόνες CSS για μεγέθη οθονών ή για διαφορετικές είδη προβολών μιας σελίδας, κάνοντας έτσι τις διάφορες σελίδες μας είναι ευανάγνωστες σε όλα τα είδη συσκευών.

Μερικές από τις συνθήκες που μπορούμε να ελέγχουμε είναι το πλάτος της οθόνης, το ύψος της, το `orientation` της σελίδας (`landscape` ή `portrait`) ή την ανάλυση της οθόνης.

Για την περίπτωση της δημιουργίας του website αυτού συνδυάσαμε τους κανόνες που εφαρμόσαμε μέσω των `media queries` και των δυνατοτήτων που μας προσφέρει το Bootstrap για να προσαρμόζονται οι διάφορες σελίδες μας σε όλα τα είδη συσκευών και οθονών.

Αυτό το πετύχαμε ορίζοντας τις ανάλογες δηλώσεις κώδικα CSS στο ίδιο το αρχείο CSS είτε δηλώνοντας ένα ξεχωριστό αρχείο μέσω του εγγράφου HTML.

Η σύνταξη έχει ως εξής: `@media not|only mediatype and (expressions) { Κανόνες CSS }`

Όπου `mediatype` αναφέρεται στο είδος της συσκευής, και η λέξη `expression` αντιπροσωπεύει μια συνθήκη που θέλουμε να εξετάσουμε αν ισχύει. Αν ολόκληρη η έκφραση ισχύει τότε θα εφαρμοστούν οι κανόνες CSS που βρίσκονται μέσα στις αγκύλες.

5.5.2 **Κινητά τηλέφωνα**

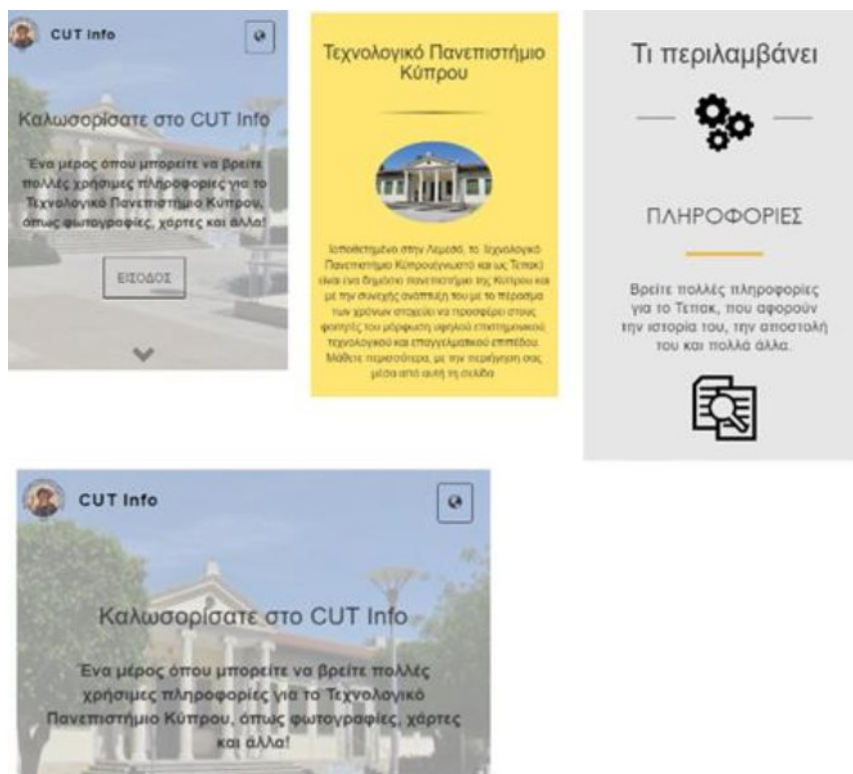
Για την περίπτωση των κινητών συσκευών ορίσαμε μέσω του εγγράφου HTML ένα κανόνα κατά τον οποίο καλείται ένα επιπλέον CSS αρχείο όταν ισχύει η έκφραση που υπάρχει στο `media query`.

Η δήλωση όπως φαίνεται και στο Διάγραμμα 5 είναι η εξής:

```
href="CSS_files/home_responsive.css" media="screen and (max-width: 750px)
```

Συμφώνα με την δήλωση αυτή, για όλους τους τύπους οθονών(screen), αν το μέγεθος του πλάτους της οθόνης είναι από 750 pixels (δηλαδή πρόκειται κυρίως για κινητά τηλέφωνα) και κάτω τότε θα φορτώνεται το αρχείο home_responsive και θα εφαρμόζονται οι κανόνες CSS που βρίσκονται σε αυτό. Το πλεονέκτημα με αυτή την μέθοδο είναι ότι το αρχείο αυτό φορτώνεται μόνο όταν ισχύει η έκφραση που υπάρχει στο media query και όχι σε άλλες περιπτώσεις. Επίσης συμβάλλει και στην καλύτερη μελλοντική συντήρησης της συσκευής καθώς οποιεσδήποτε αλλαγές που αφορούν κώδικα CSS για κινητές συσκευές θα γίνονται μόνο σε αυτό το αρχείο.

5.5.2.1 Εμφάνιση Οθόνης



Διάγραμμα 23: Εμφάνιση Οθονών Αρχικής σελίδας σε κινητές συσκευές

5.5.2.2 Κώδικας HTML για μπάρα πλοήγησης κινητών τηλεφώνων

```
<nav class="navbar navbar-inverse nav_resp">
  <div class="container-fluid">
    <div class="navbar-header">
      
      <a class="navbar-brand" href="index.html">CUT Info</a>
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
        <span class="sr-only">Toggle navigation</span>
        <span class="glyphicon glyphicon-globe"></span>
      </button>
    </div>
    <div class="navbar-collapse collapse">
      <ul class="nav navbar-nav nav_right">
        <li><a href="index_en.html"><span style="margin-left: 20px;"> ENGLISH</span></a></li>
        <li><a href="index.html"><span class="glyphicon glyphicon-ok"></span> GREEK</a></li>
      </ul>
    </div>
  </div>
</nav>
```

Διάγραμμα 24: Μπάρα πλοήγησης - κινητά τηλέφωνα

Για τα κινητά τηλέφωνα δηλώσαμε στο έγγραφο HTML μια διαφορετική μπάρα πλοήγησης κατά την οποία αντί το κείμενο 'Language' μαζί με το εικονίδιο για επιλογή γλώσσας εμφανίζεται μόνο το εικονίδιο υπό μορφή κουμπιού και όταν ο χρήστης το πατήσει εμφανίζονται οι δύο επιλογές.

Το κουμπί δημιουργείται με την κλάση navbar-toggle και την ιδιότητα data-toggle να έχει την τιμή collapse κάτι που δηλώνει ότι το περιεχόμενου του στοιχείου στο οποίο αναφέρεται (data-target=".navbar-collapse") κρύβεται και εμφανίζεται μόνο όταν πατηθεί το κουμπί.

5.5.2.3 Κώδικας CSS

```
.backgroundImg{
  background-image: url("../images/tepak_bck_sm.jpg");
}

.nav_desk{ display:none; }
.nav_resp{ display: block; }

.navbar-header a{ padding-left: 10px; }
.navbar-nav li a{ font-size: 11pt;}

.nav_right{ float: right; }
.intro{
  padding: 0px;
  margin-top: 10%;
}
.intro h1{
  letter-spacing: 0.3px;
  font-size: 24px;
}
.intro_msg p{
  margin: 30px auto;
  font-size: 17px;
  width: 90%;
}
.mybutton {
  font-size:15px;
  padding: 15px 16px;
}
.navigation{ padding-top: 40px;}

.learn_more{display: none;}

.navigation .glyphicon { font-size: 30px; }

.row_section{
  padding: 15px 5px 25px;
}
.row_section h3{
  letter-spacing: 0px;
  word-spacing: 0px;
  text-shadow: none;
  font-size: 27px;
}
.row_section p{
  line-height: 1.3;
  width: 80%;
  margin: 25px auto;
  font-size: 18px;
}
hr.hrstyle2{
  width: 30%;
  margin: 30px auto;
}
#img1, #img2{ width: 25%;}
#img1, #img2, .education{ margin: 0 auto; }
.education{ font-size: 90px; }
```

Διάγραμμα 25: Κώδικας CSS για κινητά τηλέφωνα

Πιο πάνω βλέπουμε μέρος του κώδικα CSS για τον έλεγχο της εμφάνισης των οθονών αυτής της σελίδας, για κινητά τηλέφωνα. Ο έλεγχος για το ποια μπάρα πλοήγησης θα εμφανιστεί γίνεται με την ιδιότητα `display`, όπου παίρνει την τιμή `none` για αυτή που δεν θέλουμε να εμφανίζεται και `block` για αυτή που επιθυμούμε να εμφανίζεται. Με τις υπόλοιπες ιδιότητες κάνουμε τις κατάλληλες δηλώσεις έτσι ώστε το μέγεθος των γραμμμάτων, ο χώρος που καταλαμβάνει ένα κείμενο ή μια εικόνα στην οθόνη και η τοποθεσία των διάφορων στοιχείων και η απόσταση τους σε σχέση με άλλα στοιχεία να είναι η κατάλληλη.

Μια σημαντική δήλωση που ορίσαμε επίσης, είναι η αλλαγή της εικόνας φόντου με την ιδιότητα `background-image` λόγω του ότι για την περίπτωση των κινητών τηλεφώνων επιθυμούμε να εμφανίζεται μια διαφορετική φωτογραφία πίσω στο φόντο η οποία να ταιριάζει με τις διαστάσεις της οθόνης. Επίσης όταν το `orientation` της συσκευής είναι `landscape` τότε εφαρμόζονται κάποιοι επιπλέον κανόνες βάση του `media query @media (orientation: landscape){.. }`.

5.5.3 Συσκευές tablet

5.5.3.1 Εμφάνιση Οθονών



Διάγραμμα 26: Εμφάνιση Οθονών Αρχικής σελίδας σε συσκευές tablet

5.5.3.2 Κώδικας CSS

```
@media (min-width: 751px) and (max-width: 950px) {  
  .intro h1{  
    letter-spacing: 0.6px;  
    font-size: 45px;  
  }  
  
  .intro_msg p{  
    margin: 50px auto 40px;  
    font-size: 20px;  
    width: 70%;  
  }  
  
  .mybutton {  
    font-size:15px;  
  }  
  
  .tepak_info h1{  
    margin-top: 8%;  
    font-size: 37px;  
  }  
  
  .tepak_info p{  
    text-align: left;  
    margin: 7% auto 10%;  
    width: 100%;  
    font-size: 13pt;  
  }  
  
  #tepak_img{  
    display: block;  
    margin-top: 15%;  
    width: 60%;  
    box-shadow: none;  
  }  
  
  .infoheader{  
    padding: 50px 0px 7px 0px;  
  }  
  
  .infoheader h2{  
    font-size: 38px;  
  }  
  
  .row_section{  
    padding: 28px 0px 28px 0px;  
  }  
  
  .row2{ padding-left: 8%;}  
  
  .row_section h3{  
    text-transform: uppercase;  
    letter-spacing: 1.3px;  
    word-spacing: 0px;  
    font-size: 30px;  
  }  
}
```

Διάγραμμα 27: Κώδικας CSS για συσκευές tablet

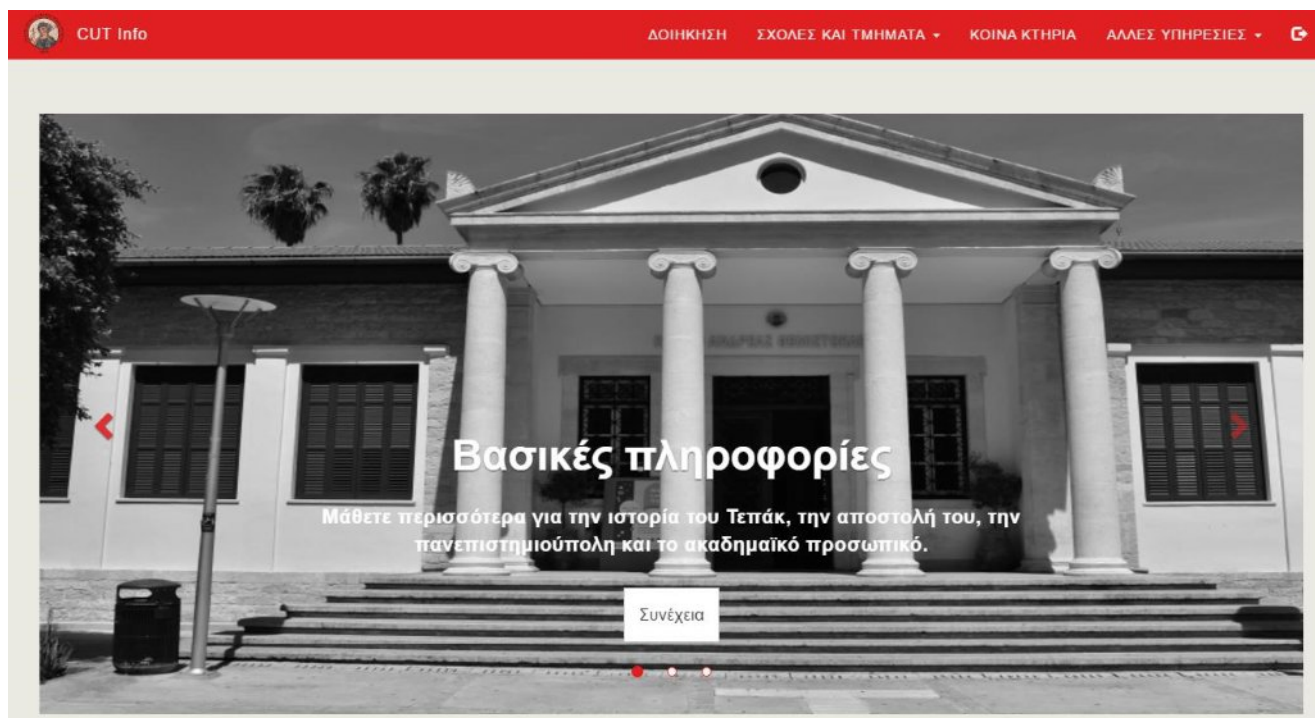
Για την περίπτωση των συσκευών tablet χρησιμοποιήσαμε στο ίδιο αρχείο την έκφραση με media query που φαίνεται πιο πάνω έτσι ώστε οι κανόνες αυτοί να εφαρμόζονται μόνο σε περίπτωση όπου το πλάτος της οθόνης είναι μεταξύ 750 pixel και 950pixel που είναι συνήθως οι διαστάσεις οθονών tablet ((min-width: 751px) and (max-width: 950px)).

6 Δημιουργία Κεντρικής Σελίδας

Σε αυτή τη σελίδα ο χρήστης θα μπορεί να μάθει για κάποιες βασικές πληροφορίες που αφορούν το Τεπακ και συγκεκριμένα για την ιστορία του, τους στόχους του, για την πανεπιστημιούπολη και για το ακαδημαϊκό προσωπικό. Επίσης ο χρήστης θα μπορεί μέσω αυτής της σελίδας να μεταβεί σε άλλες σελίδες που αφορούν το Τεπακ και να μάθει περισσότερα για τα τμήματα, τις σχολές και για άλλες εγκαταστάσεις του Τεπακ.

6.1 Αρχική Οθόνη Κεντρικής Σελίδας

Σε αυτή την οθόνη υπάρχει αρχικά η μπάρα πλοήγησης όπου υπάρχουν οι σύνδεσμοι για να μεταβαίνει ο χρήστης στις υπόλοιπες σελίδες. Στην συνέχεια υπάρχει ένα πλαίσιο με μια ομαλή και ευχάριστη εναλλαγή διαφανειών στις οποίες παρουσιάζονται συνοπτικά το περιεχόμενο που ακολουθεί στις επόμενες σελίδες και τις πληροφορίες που υπάρχουν σε αυτές.



Διάγραμμα 28: Αρχική Οθόνη Κεντρικής Σελίδας

6.1.1 Πλαίσιο με εναλλαγή Διαφανειών

6.1.1.1 Κώδικας HTML

Η δημιουργία αυτού του πλαισίου με τις εναλλασσόμενες διαφάνειες έγινε χρησιμοποιώντας ένα στοιχείο του Bootstrap, το carousel.

```
<div id="myCarousel" class="carousel slide" data-ride="carousel" data-interval="4000">

  <!-- Indicators -->
  <ol class="carousel-indicators">
    <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
    <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
  </ol>

  <!-- Wrapper for slides -->
  <div class="carousel-inner" role="listbox">

    <div class="item active">
      <div class="carousel-caption">
        <h2>Βασικές πληροφορίες</h2>
        
        <p>
          Μάθετε περισσότερα για την ιστορία του Τεπάκ, την αποστολή του,
        </p>
        <a href="#main_content" class="btn btn-default btn-md carousel_button">
        </div>
      </div>

    <div class="item">
      <div class="carousel-caption">
        <h2>Σχολές και Τμήματα</h2>
        <p>Υπάρχουν αυτή τη στιγμή 6 διαφορετικές σχολές στο Τεπάκ, κάθε μια με τα τμήματα της.
        </div>
      </div>

    <div class="item">
      <div class="carousel-caption">
        <h2>Άλλες Εγκαταστάσεις και Υπηρεσίες</h2>
        <p>Δείτε επίσης περισσότερα για άλλες
        </div>
      </div>

  </div> <!-- End of carousel items -->

  <!-- Left and right controls -->
  <a class="left carousel-control" href="#myCarousel" data-slide="prev">
    <span class="glyphicon glyphicon-chevron-left">
    <span class="sr-only">Previous</span>
  </a>
  <a class="right carousel-control" href="#myCarousel" data-slide="next">
    <span class="glyphicon glyphicon-chevron-right">
    <span class="sr-only">Next</span>
  </a>

</div> <!-- End of Carousel -->
```

Διάγραμμα 29: Κώδικας HTML για Carousel

Πιο πάνω φαίνεται ο κώδικας με τον οποίο δημιουργούμε ένα carousel στο HTML έγγραφο. Με την κλάση carousel δηλώνουμε ότι αυτό το κομμάτι/πλαίσιο της σελίδας θα περιλαμβάνει ένα carousel και η κλάση slide προσθέτει ένα animation κάθε φορά που γίνεται εναλλαγή στις διαφάνειες. Η ιδιότητα data-slide και η τιμή carousel φροντίζει έτσι ώστε να ξεκινά η εναλλαγή των διαφανειών μόλις φορτωθεί η σελίδα, ενώ η ιδιότητα data-interval και η τιμή της καθορίζει το πόσο συχνά θα γίνεται η εναλλαγή των διαφανειών milliseconds). Η κλάση carousel-indicators αντιπροσωπεύει τους μικρούς δείκτες που φαίνονται στο κάτω μέρος του carousel που δείχνουν το πόσες διαφάνειες υπάρχουν. Στο τμήμα carousel-inner υπάρχουν ξεχωριστά οι διαφάνειες του carousel που η κάθε μια να δηλώνεται με την κλάση item και το περιεχόμενο της να βρίσκεται κάτω από την κλάση carousel-caption. Τέλος με την κλάση carousel-control και τις κλάσεις left και right δηλώνουμε τα δύο κουμπιά που βρίσκονται στις

άκρες πλάγια του carousel με κατεύθυνση αριστερά και δεξιά αντίστοιχα, τα οποία μπορεί να πατά ο χρήστης για να μεταβαίνει από την μια διαφάνεια στην άλλη

6.1.1.2 Κώδικας CSS

```
.carousel .item{
  margin-top: -100px;
  height: 730px;
}

/* background images to the carousel items*/
.item:nth-of-type(1){
  background-image: url("../images/carousel_tepak.JPG");
  background-repeat: no-repeat;
  background-position: center center;
  background-size: 95% 95%;
  -webkit-filter: grayscale(100%);
  filter: grayscale(100%);
}

.item:nth-of-type(2){
  background-image: url("../images/carousel_dep.jpg");
  background-repeat: no-repeat;
  background-position: center center;
  background-size: 95% 95%;
  -webkit-filter: grayscale(100%);
  filter: grayscale(100%);
}

.item:nth-of-type(3){
  background-image: url("../images/facilities.jpg");
  background-repeat: no-repeat;
  background-position: center center;
  background-size: 95% 95%;
  -webkit-filter: grayscale(95%);
  filter: grayscale(95%);
}

.carousel-control.right, .carousel-control.left {
  background-image: none;
  color: #df2020;
}

.carousel-caption{ padding-bottom: 50px; }
.carousel-caption h2 {
  padding-bottom: 15px;
  color: white;
  font-weight: bold;
  font-size: 32pt;
  font-family: "Montserrat", sans-serif;
}
.carousel-caption p{
  padding-bottom: 20px;
  color: white;
  font-weight: bold;
  letter-spacing: 0.5px;
  font-size: 20px;
  text-shadow: none !important;
  font-family: 'Nunito', sans-serif;
}

.carousel-indicators{ margin-bottom: 30px; }
.carousel-indicators li, .carousel-indicators li.active{ margin-left: 10px; }
.carousel-indicators li:first-child { margin-left: 0px; }

.carousel-indicators li {
  border-color: #df2020;
  background-color: whitesmoke;
}
```

Διάγραμμα 30: Κώδικας CSS για Carousel

Πιο πάνω βλέπουμε τον κώδικα με τον οποίο ελέγχουμε την εμφάνιση στα διάφορα στοιχεία του carousel. Για την κάθε διαφάνεια ορίσαμε μια εικόνα φόντου με την ιδιότητα background-image, η οποία θα φαίνεται μαυρόασπρη χρησιμοποιώντας την ιδιότητα grayscale. Στην συνέχεια ορίσαμε το πως θα φαίνονται τα διάφορα στοιχεία, όπως για παράδειγμα το κείμενο μέσα στις διαφάνειες, οι επικεφαλίδες, τα κουμπιά για εναλλαγή διαφανειών και η απόσταση που θα έχουν μεταξύ τους. Αυτό έγινε με την χρήση των κατάλληλων ιδιοτήτων που ελέγχουν το χρώμα των γραμμάτων, το μέγεθος τους, το είδος της γραμματοσειράς κλπ.

6.1.2 Μπάρα πλοήγησης

Η μπάρα πλοήγησης, αποτελείται όπως φαίνεται και στο Διάγραμμα 28 από ένα γενικό κατάλογο και τις επιπλέον λίστες για τα τμήματα και τις υπόλοιπες υπηρεσίες του Τεπακ

(Διάγραμμα 31) που θα εμφανιστεί όταν ο χρήστης περάσει τον δείκτη του ποντικιού από πάνω τους.



Διάγραμμα 31: Εμφάνιση λίστας/καταλόγου

6.1.2.1 Κώδικας HTML

Ο κώδικας HTML για αυτή την μπάρα πλοήγησης έχει πολύ παρόμοια δομή με αυτή που εξετάσαμε στο υποκεφάλαιο 5.2.5.1. Αποτελείται από την επικεφαλίδα που βρίσκεται στο αριστερό τμήμα, και τους υπόλοιπους συνδέσμους μαζί με τις επιπλέον λίστες που βρίσκονται στο δεξί τμήμα της μπάρας.

```
<nav class="navbar navbar-inverse">
  <div class="container-fluid" id="page_start">

    <div class="navbar-header">
      
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="mainpage_gr.html">CUT Info</a>
    </div>

    <div class="navbar-collapse collapse">
      <ul class="nav navbar-nav navbar-right">
        <li><a href="Others/common/administration_gr.html">Διοίκηση</a></li>
        <li class="dropdown">
          <a class="dropdown-toggle" data-toggle="dropdown" href="#">Σχολες και Τμήματα <span class="caret"></span></a>
          <ul class="dropdown-menu">
            <li class="faculty">Σχολή Γεωτεχνικών Επιστημών και Διαχείρισης Περιβάλλοντος</li>
            <li><a href="Departments/Biotechnology/Biotechnology_gr.html">Τμήμα Γεωπονικών Επιστημών</a>
            <li><a href="Departments/Environmental/Environmental_gr.html">Τμήμα Επιστήμης και Τεχνολογίας Περιβάλλοντος</a>
            <li class="faculty">Σχολή Διοίκησης και Οικονομίας</li>
            <li><a href="Departments/Economics/HotelManagement_gr.html">Τμήμα Διοίκησης Ξενοδοχείων</a>
            <li><a href="Departments/Economics/Finance_gr.html">Τμήμα Εμπορίου, Χρηματοοικονομικών</a>
            <li><a href="Departments/Economics/Management_gr.html">Διατμηματικό Πρόγραμμα Διοίκησης</a>
            <li class="faculty">Σχολή Επικοινωνίας και Μέσων Ενημέρωσης</li>
          </ul>
        </li>
      </ul>
    </div>
  </div>
</nav>
```

Διάγραμμα 32: Κώδικας HTML - μπάρα πλοήγησης κεντρικής σελίδας

6.1.2.2 Κώδικας CSS

```
/*nav bar styling*/
.navbar-inverse{
  background-color: #df2020;
  border: none;
  border-radius: 0;

  .navbar-right li a, .navbar-right ,.navbar-brand {
    color: whitesmoke !important;
    letter-spacing: 1px;
    font-family: "Montserrat",sans-serif !important;
  }

  .navbar-header a:hover{
    color: #d6d6c2 !important;
    -webkit-transition-duration: 0.45s;
    transition-duration: 0.45s;
  }

  .navbar-nav li{ text-transform: uppercase; }

  .navbar-right li a:hover, .navbar-right li.active a,
  -webkit-transition-duration: 0.3s;
  transition-duration: 0.3s;
  color: #df2020 !important;
  background-color: whitesmoke !important;
}

.logoutIcon{ font-size: 15px; }

#logo{
  float:left;
  height: 47px;
  width: 47px;
}

/* styling for the dropdown menu */
@media (min-width: 750px){
  .dropdown-menu li a{
    color: black !important;
    text-transform: none !important;
    font-size: 11pt;
  }

  .faculty{
    color: #cc4400;
    padding-left: 3.5px;
  }

  .open .dropdown-toggle {
    background-color: whitesmoke !important;
    color: #df2020 !important;
  }
}

@media (min-width: 1100px){
  .dropdown-menu {
    opacity: 0.2;
    -webkit-transform-origin-y: top;
    -webkit-transform: scale(1, 0);
    display: block; /* its an inline el
    -webkit-transition-duration: 0.2s;
    transition-duration: 0.2s;
  }

  .dropdown:hover .dropdown-menu {
    -webkit-transform: scale(1, 1);
    transform: scale(1, 1);
    opacity:1;
    display: block;
  }
}
```

Διάγραμμα 33: Κώδικας CSS - μπάρα πλοήγησης κεντρικής σελίδας

Πιο πάνω απεικονίζεται ο κώδικας με τον οποίο ελέγξαμε την εμφάνιση της μπάρας πλοήγησης αυτής της σελίδας. Με την χρήση των κατάλληλων ιδιοτήτων θέσαμε τα κατάλληλα χρώματα και είδη γραμματοσειρών, έτσι ώστε τα διάφορα κομμάτια της λίστας να ξεχωρίζουν μεταξύ τους. Επίσης ορίσαμε ένα χρώμα για ολόκληρη την μπάρα για να ξεχωρίζει από το υπόλοιπο τμήμα της σελίδας. Για την εμφάνιση των επιπλέον καταλόγων ορίσαμε τους κανόνες μέσα στα κατάλληλα media queries έτσι ώστε οι κανόνες αυτοί να εφαρμόζονται μόνο στις συσκευές με μέγεθος οθόνης tablet και πάνω (@media (min-width: 750px)) και τα animations που αφορούν τον τρόπο εμφάνισης τους να εφαρμόζονται μόνο σε πιο μεγάλες οθόνες (@media (min-width: 1100px)). Περισσότερα για τα Animations θα εξεταστούν σε επόμενο κεφάλαιο.

6.2 Οθόνες με Βασικές Πληροφορίες Τεπάκ


Σε αυτό το τμήμα της σελίδας ο χρήστης θα μπορεί να δει βασικές πληροφορίες που αφορούν το Τεπακ. Με το που θα φτάνει σε αυτό το σημείο της σελίδας θα εμφανίζεται στο πάνω μέρος της οθόνης μια νέα μπάρα πλοήγησης με την οποία ο χρήστης θα έχει την δυνατότητα να περιηγείται σε αυτές τις πληροφορίες μέσω της σελίδας. Ο κώδικας HTML για αυτή την μπάρα είναι παρόμοιος με τις περιπτώσεις για δημιουργία μπάρα πλοήγησης που είδαμε πιο πάνω. Χρησιμοποιήσαμε επίσης την ιδιότητα `data-spy="scroll"` του Bootstrap με την οποία συνδέουμε αυτή την μπάρα πλοήγησης με την υπόλοιπη σελίδα έτσι ώστε το χρώμα φόντου των συνδέσμων της να αλλάζει ανάλογα με την θέση που βρίσκεται ο χρήστης στην σελίδα.

ΙΣΤΟΡΙΑ ΟΡΑΜΑ ΚΑΙ ΑΠΟΣΤΟΛΗ ΠΑΝΕΠΙΣΤΗΜΙΟΥΠΟΛΗ ΑΚΑΔΗΜΑΙΚΟ ΠΡΟΣΩΠΙΚΟ


Ιστορία

Το Τεχνολογικό Πανεπιστήμιο Κύπρου (ΤΕΠΑΚ) είναι ένα νέο δημόσιο Πανεπιστήμιο, που ιδρύθηκε με νόμο του κράτους το 2004 και δέχθηκε τους πρώτους προπτυχιακούς φοιτητές το Σεπτέμβριο του 2007. Έδρα του Πανεπιστημίου είναι η Λεμεσός όπου λειτουργούν και οι έξι Σχολές του και ο συναγωνισμός για είσοδο σε αυτό είναι πολύ έντονος από τον πρώτο χρόνο λειτουργίας του.

Είναι ένα σχετικά καινούργιο πανεπιστήμιο όπου η ίδρυση του αποτέλεσε μια προσπάθεια να καλυφθούν τα κενά που είχε ο εκπαιδευτικός τομέας της Κύπρου προσφέροντας προπτυχιακές και μεταπτυχιακές σπουδές σε επίπεδο που δεν προσφέρεται από άλλο ακαδημαϊκό ίδρυμα στην Κύπρο.



Όραμα και Αποστολή



Το ΤΕΠΑΚ φιλοδοξεί να εξελιχθεί σε ένα σύγχρονο πρωτοποριακό Πανεπιστήμιο με διεθνή αναγνώριση ικανό να προσφέρει εκπαίδευση και έρευνα υψηλού επιπέδου σε κλάδους αιχμής που σήμερα έχουν μεγάλη οικονομική, τεχνική και επιστημονική απόδοση.

Με τον προσανατολισμό του στην εφαρμοσμένη έρευνα, το Τεχνολογικό Πανεπιστήμιο Κύπρου φιλοδοξεί να καταστεί σημαντικός αρωγός της πολιτείας και της κοινωνίας μας στην αντιμετώπιση προβλημάτων που την απασχολούν σε όλους τους τομείς της επιστήμης, της τεχνολογίας και της γνώσης που υπηρετούνται σε αυτό.

Διάγραμμα 34: Δείγμα Οθόνης με βασικές πληροφορίες Τεπακ

6.2.1 Κώδικας HTML

```
<!-- Main Content - information about the University -->
<div class="container-fluid content" id="main_content">

  <div id="history" class="row info">
    <div class="slideanim_m">
      <h1>Ιστορία</h1>

      <div class="col-sm-7 text_history">
        <p>
          Το Τεχνολογικό Πανεπιστήμιο Κύπρου (ΤΕΠΑΚ) είναι ένα νέο δημόσιο Πανεπιστήμιο, που ιδρύθηκε
          Έδρα του Πανεπιστημίου είναι η Λεμεσός όπου λειτουργούν και οι έξι Σχολές του και ο συναγωγ
        </p>
        <p>
          Είναι ένα σχετικά καινούργιο πανεπιστήμιο όπου η ίδρυση του αποτέλεσε μια προσπάθεια να καλυ
          προσφέροντας προπτυχιακές και μεταπτυχιακές σπουδές σε επίπεδο που δεν προσφέρεται από άλλο
        </p>
      </div>
    </div>
    <div class="col-sm-5 img_content img_content1 iconanim"></div>
  </div>

  <div id="mission" class="row bg-whitesmoke">
    <div class="slideanim_m">
      <h1>Όραμα και Αποστολή</h1>

      <div class="col-sm-7 col-sm-push-5 text_vision">
        <p>
          Το ΤΕΠΑΚ φιλοδοξεί να εξελιχθεί σε ένα σύγχρονο πρωτοποριακό Πανεπιστήμιο με διεθνή αναγνώριση
        </p>
      </div>
    </div>
  </div>
```

Διάγραμμα 35: Κώδικας HTML για βασικές πληροφορίες Τεπακ

Χρησιμοποιώντας και πάλι το Grid System του Bootstrap, δημιουργήσαμε μια διάταξη με 4 γραμμές με δύο στήλες στην κάθε μια, όπου εναλλάξ βρίσκεται από την μια μεριά το κείμενο με τις πληροφορίες και από την άλλη ένα εικονίδιο που να αντιστοιχεί στο κείμενο που περιγράφεται (Διάγραμμα 34). Η κάθε γραμμή έχει το δικό της id έτσι ώστε να συνδέεται με την δεύτερη μπάρα πλοήγησης της σελίδας.

6.2.2 Κώδικας JavaScript για εμφάνιση της δεύτερης μπάρας πλοήγησης

```
window.onscroll = function() {myFunction()};

function myFunction() {
  if (document.body.scrollTop > 550 || document.documentElement.scrollTop > 550)
  {
    $(".navbar-default").addClass("navbar-fixed-top nav_slide");
    $(".uparrow").addClass("slide_content");
    if (document.body.scrollTop > 2340 || document.documentElement.scrollTop > 2340)
    {
      $(".uparrow").removeClass("slide_content");
    }
  }
  else
  {
    $(".navbar-default").removeClass("navbar-fixed-top nav_slide");
    $(".uparrow").removeClass("slide_content");
  }
}
}
```

Διάγραμμα 36: JavaScript για εμφάνιση δεύτερης μπάρας πλοήγησης

Σύμφωνα με τον πιο πάνω κώδικα υπάρχει μια συνάρτηση η οποία ενεργοποιείται όταν ο χρήστης μετακινηθεί προς τα κάτω στην οθόνη. Στην συνέχεια, αυτή η συνάρτηση με την συνθήκη (`document.body.scrollTop > 550 || document.documentElement.scrollTop > 550`) ελέγχει αν ο χρήστης έχει περάσει το όριο που θέσαμε (550) και αν ναι, τότε προσθέτει στην δεύτερη μπάρα πλοήγησης τις κατάλληλες κλάσεις έτσι ώστε να εμφανιστεί(ή να εξαφανιστεί) στην οθόνη του χρήστη αυτή η μπάρα, αλλά και ένα τόξο με κατεύθυνση προς τα πάνω (που θα βρίσκεται στο κάτω μέρος της οθόνης) το οποίο θα αποτελεί σύνδεσμο για να μετακινηθεί ο χρήστης στην αρχή της σελίδας.

6.2.3 Κώδικας CSS

```
.content{
  height: auto;
  margin-top: 80px;
  background-color: white;
}

.content h1{
  padding-bottom: 70px;
  text-align: center;
  font-family: "Montserrat", sans-serif;
  font-size: 32pt;
  text-shadow: 2px 2px 8.5px grey;
}

.content p{
  width: 85%;
  margin: 12px auto;
  font-size: 12.5pt;
  font-family: 'Nunito', sans-serif;
  letter-spacing: 0.4px;
  text-indent: 25px;
}

.bg-whitesmoke, .info { padding: 80px 0px 100px 0px; }

.img_content{
  height: 200px;
  background-repeat: no-repeat;
  background-position: center center;
  background-size: 100% 100%;
}

.img_content1{ background-image: url("../images/history.svg"); }
.img_content2{ background-image: url("../images/vision.svg"); }
.img_content3{ background-image: url("../images/campus.svg"); margin-top: 28px;}
.img_content4{ background-image: url("../images/staff.svg"); margin-top: 28px;}

.bg-whitesmoke {
  background-color: whitesmoke;
  border-top: 1px solid #d6d6c2;
  border-bottom: 1px solid #d6d6c2;
}
```

Διάγραμμα 37: CSS για οθόνη με βασικές πληροφορίες Τεπακ

6.3 Σχεδίαση σελίδας για άλλες συσκευές – responsive design

Όπως και στην περίπτωση της αρχικής σελίδας, το αρχείο CSS που είναι υπεύθυνο για την εφαρμογή κανόνων CSS καλείται μέσω media query και εφαρμόζεται μόνο στην περίπτωση όπου το πλάτος της οθόνης της συσκευής είναι κάτω από 750 pixel.

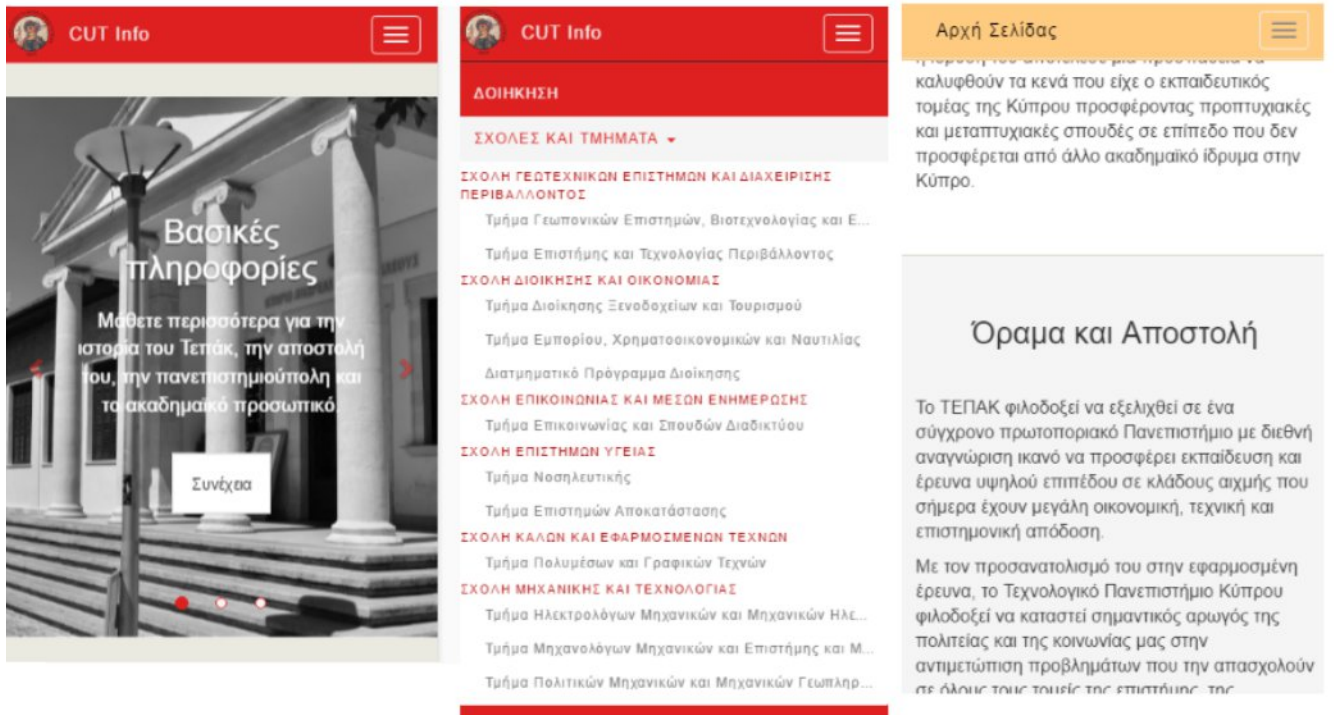
Παρόμοια και με την περίπτωση της αρχικής σελίδας, η μπάρα πλοήγησης με την χρήση της κλάσης `navbar-collapse` κρύβεται στα κινητά τηλέφωνα και εμφανίζεται σε κάθετη διάταξη όταν ο χρήστης πατήσει το κουμπί/μενού που φαίνεται στα δεξιά της οθόνης (που δημιουργείται με την κλάση `navbar-toggle`). Επιπλέον, με την χρήση των δηλώσεων `overflow: hidden` και `text-overflow: ellipsis !important` σιγουρέψαμε ότι τα στοιχεία των καταλόγων που περιλαμβάνουν τα τμήματα του Τεπακ και τις άλλες υπηρεσίες δεν θα εξέρχουν από τις διαστάσεις της οθόνης έτσι ώστε να μην χρειάζεται να κάνει δεξιά μετακίνηση ο χρήστης.

Για τις εικόνες φόντου των διαφανειών του carousel, χρησιμοποιήσαμε δίνοντας άλλες τιμές στην κλάση `background-image` διαφορετικές φωτογραφίες που να ταιριάζουν στις διαστάσεις των κινητών τηλεφώνων. Σημειώνεται ότι για τις φωτογραφίες αυτές χρησιμοποιήσαμε εικονίδια σε μορφή αρχείου SVG, μορφή με την οποία μπορούμε να δημιουργούμε εικονίδια για ιστοσελίδες τα οποία χρειάζονται πολύ λίγο χώρο για να αποθηκευτούν στους φακέλους του τοπικού συστήματος, ενώ επίσης αλλάζει το μέγεθος τους πολύ εύκολα και φαίνονται ολοκάθαρα και σε πιο μικρές συσκευές.

Ακόμη, αλλάξαμε τον τρόπο εμφάνισης των στοιχείων του carousel έτσι ώστε να στοιχίζονται κατάλληλα και να γίνονται πιο ευανάγνωστα για κινητές συσκευές. Η ιδιότητα `nth-of-type(n)` που χρησιμοποιήθηκε σε αυτή την περίπτωση επιλέγει κάθε n στοιχείο ενός συγκεκριμένου τύπου, σε αυτή την περίπτωση, του τύπου `item` που αντιπροσωπεύει τις διαφάνειες του carousel.

Επίσης για τα τμήματα της σελίδας όπου θα εμφανίζονται οι γενικές πληροφορίες για το Τεπακ λόγω του ότι επιθυμούσαμε να μην εμφανίζονται τα εικονίδια που συνοδεύουν το κείμενο, θέσαμε την τιμή της ιδιότητας `background-image` για τις κλάσεις αυτές σε `none` και την δήλωση `.img_content{display: none;}` Αυτό θα κάνει τις εικόνες αυτές να μην φορτώνονται έτσι ώστε να μην χρειάζεται να κατεβάζει δεδομένα άσκοπα η σελίδα.

Για τις οθόνες tablet η δομή της σελίδας παραμένει η ίδια και για οποιοσδήποτε αλλαγές έγιναν ορίστηκαν οι ανάλογοι κανόνες CSS στο ίδιο αρχείο με την χρήση media queries.



Διάγραμμα 38: Οθόνες κεντρικής σελίδας - κινητές συσκευές

```

.navbar-header a{ margin: 0; padding-left: 10px; font-size: 12pt; }
.navbar-right li a{font-size: 13px !important;}

.dropdown-menu{
    background-color: white !important;
}

.faculty{
    overflow: hidden;
    text-overflow:ellipsis !important;
    color: #b30000;
    font-size: 8.5pt;
    padding-left: 3px;
}

.dropdown-menu li a{
    color: grey !important;
    overflow: hidden;
    text-overflow:ellipsis !important;
    text-transform: none !important;
    font-size: 9pt !important;
}

.header2{
    display: block;
    margin-left: 18px !important;
    color: black !important;
}

.item:nth-of-type(1){
    background-image: url("../images/carousel_tepak_xs.jpg");
    background-size: 100% 100%;
}

.item:nth-of-type(2){
    background-image: url("../images/carousel_dep_xs.jpg");
    background-size: 100% 100%;
}

.item:nth-of-type(3){
    background-image: url("../images/facilities_xs.jpg");
    background-size: 100% 100%;
}

.carousel-caption h2 {
    font-weight: normal;
    padding-bottom: 8px;
    font-size: 22pt;
}

.carousel-caption p{
    letter-spacing: 0px;
    font-size: 17.5px;
    font-weight: normal;
    text-shadow: none !important;
}

.content{
    height: auto;
}

.content p{
    width: 100%;
    margin: 9px auto;
    font-size: 16px;
    letter-spacing: 0px;
    text-indent: 0px;
}

.content h1{
    padding-bottom: 30px;
    font-size: 27px;
    text-shadow: none;
}

.img_content{ display: none;}
.img_content1, .img_content2, .img_content3, .img_content4 { background-image: none;}

.bg-whitesmoke, .info { padding: 35px 0px 45px 0px; }

.carousel-indicators{ margin-bottom: 10px; }

```

Διάγραμμα 39: Κώδικας CSS για κινητές συσκευές - οθόνες κεντρικής σελίδας

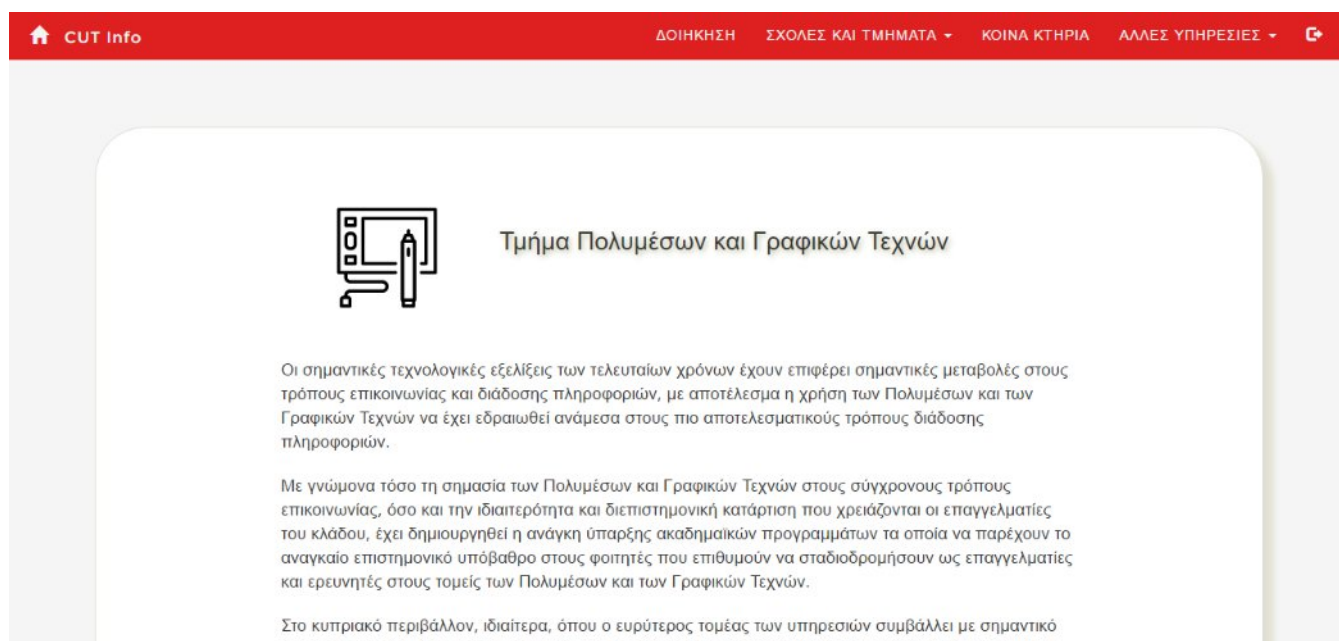
7 Δημιουργία σελίδων τμημάτων Τεπακ

Σε αυτές τις σελίδες ο χρήστης θα μπορεί να βρει κάποιες βασικές πληροφορίες για το τμήμα ή την υπηρεσία του Τεπακ που τον ενδιαφέρει. μαζί με φωτογραφικό υλικό και χάρτες των κτηρίων. Οι σελίδες αυτές χωρίστηκαν σε τρία τμήματα, ένα για το κάθε κομμάτι που προαναφέραμε. Κάθε τμήμα της σελίδας ξεχωρίζει από το άλλο εμφανισιακά, με την χρήση περιγράμματος και σκίασης πίσω από το τμήμα που βρίσκεται το κάθε περιεχόμενο.

Επίσης, η μπάρα πλοήγησης που υπάρχει στο πάνω μέρος της οθόνης σχεδιάστηκε με τον ίδιο τρόπο όπως και στην κεντρική σελίδα, με την διαφορά ότι το εικονίδιο που υπάρχει στο αριστερό της τμήμα είναι διαφορετικό, για να δηλώνει ότι πατώντας αυτό τον σύνδεσμο ο χρήστης θα μπορεί να επιστρέφει στην κεντρική σελίδα.

7.1 Οθόνη περιγραφής τμήματος

Σε αυτή την οθόνη θα περιγράφεται συνοπτικά ο ρόλος του τμήματος, κάποιες γενικές πληροφορίες για αυτό, το τι μπορεί να προσφέρει στον φοιτητή καθώς και τις ευκαιρίες που προσφέρει στους απόφοιτους. Επίσης στο τέλος θα δίνεται και ένας σύνδεσμος που θα μεταφέρει τον χρήστη στην επίσημη ιστοσελίδα του Τεπακ για να μάθει αν τον ενδιαφέρει περισσότερες πληροφορίες για τα μαθήματα του τμήματος, το προσωπικό κλπ.



Διάγραμμα 40: Οθόνη περιγραφής τμήματος

7.1.1 Κώδικας HTML

```
<div class="container content" id="summary">

  <div class="row content_header">

    <div class="col-md-7 col-md-push-4 slideh">
      <h2>Τμήμα Πολυμέσων και Γραφικών Τεχνών</h2>
    </div>

    <div class="col-md-2 col-md-pull-5">
      
    </div>

  </div>

  <div class="content_text slide">

    <p>
      Οι σημαντικές τεχνολογικές εξελίξεις των τελευταίων χρόνων έχουν επιφέρει σημαντικές μεταβολές
      στους τρόπους επικοινωνίας και διάδοσης πληροφοριών, με αποτέλεσμα η χρήση των Πολυμέσων και των I
    </p>
    <p>
      Με γνώμονα τόσο τη σημασία των Πολυμέσων και Γραφικών Τεχνών στους σύγχρονους τρόπους επικοινωνίας
      την ιδιαιτερότητα και διεπιστημονική κατάρτιση που χρειάζονται οι επαγγελματίες του κλάδου, έχει
      η ανάγκη ύπαρξης ακαδημαϊκών προγραμμάτων τα οποία να παρέχουν το αναγκαίο επιστημονικό υπόβαθρο
      που επιθυμούν να σταδιοδρομήσουν ως επαγγελματίες και ερευνητές στους τομείς των Πολυμέσων και των
    </p>
    <p>
      Στο κυπριακό περιβάλλον, ιδιαίτερα, όπου ο ευρύτερος τομέας των υπηρεσιών συμβάλλει με σημαντικό
      διαμέρισμα του εθνικού μας εισοδήματος, και απασχολεί σημαντικό αριθμό εργαζόμενων, επιβάλλεται
      εργοδότηση άρτια καταρτισμένων επιστημόνων με εξειδίκευση στις εφαρμογές των Πολυμέσων και των Γραφικών
      Με πλήρη επίγνωση των σύγχρονων αναγκών, το Τμήμα προσφέρει υψηλού επιπέδου πανεπιστημιακό πτυχίο
      Το Τμήμα ευελπιστεί να δημιουργήσει το κατάλληλο περιβάλλον, το οποίο να παρέχει την αναγκαία
    </p>
    <p>
      Για περισσότερες πληροφορίες: <a href="http://www.cut.ac.cy/mga/">http://www.cut.ac.cy/mga/</a></p>
  </div> <!-- end of class content_text -->

</div> <!-- end of class container content -->
```

Διάγραμμα 41: Κώδικας HTML οθόνης περιγραφής τμήματος

Για το πάνω μέρος της οθόνης δημιουργήσαμε μια διάταξη με την χρήση του grid συστήματος του Bootstrap όπου δημιουργήσαμε μια γραμμή με δυο στήλες, με την μια να αποτελεί τον τίτλο του τμήματος και την άλλη ένα μικρό εικονίδιο που αντιπροσωπεύει το τμήμα. Αυτό θα ισχύει για οθόνες μεγάλων ηλεκτρονικών υπολογιστών και πάνω (κλάση md) όπου αλλάξαμε και την σειρά τους ανάλογα (κλάσεις push και pull) ενώ για μικρότερες από αυτές οθόνες οι στήλες θα στοιβάζονται η μια πάνω στην άλλη. Στην συνέχεια κάτω την κλάση content_text υπάρχει το κείμενο με την περιγραφή του τμήματος

7.1.2 Κώδικας CSS

```
.content{
  height: auto;
}

.content_header h2, .images_intro h2, .map_section h2,
font-family: "Montserrat", sans-serif;
text-shadow: 2px 2px 7px #b6b696;
}

.content_header h2{ font-size: 25px; width: 80%;}

#dep_img, #others_img{
  display: block;
  margin-left: auto;
  margin-right: auto;
  width: 75%;
  height: 100px;
}

.content_text p:first-of-type{
  margin-top: 50px;
}

.content_text p:last-of-type{
  font-style: italic;
  margin-top: 30px;
  font-size: 16px;
}

.content_text p{
  width: 70%;
  margin: 20px auto;
  clear: both;
  font-family: 'Nunito', sans-serif;
  font-size: 17px;
}
```

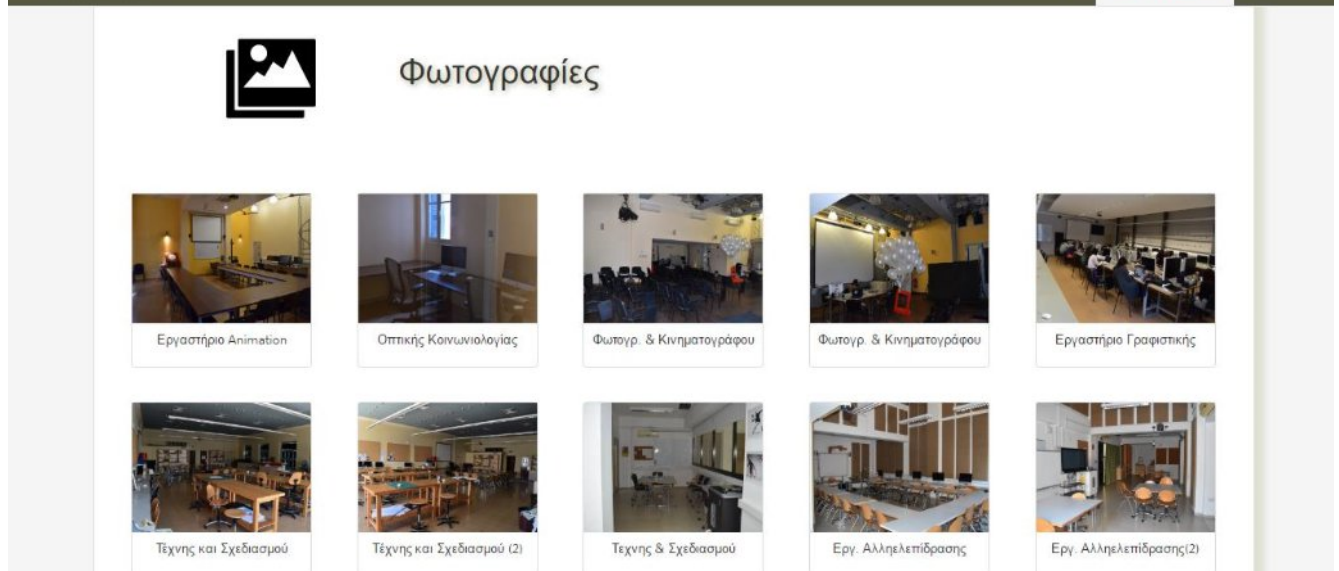
Διάγραμμα 42: Κώδικας CSS - οθόνη περιγραφής τμήματος

Με τους πιο πάνω κανόνες ελέγχεται η εμφάνιση των στοιχείων στην οθόνη όπου γίνεται η περιγραφή του τμήματος. Με τις δηλώσεις `first-of-type` και `last-of-type` ορίζουμε την εμφάνιση της πρώτης και της τελευταίας παραγράφου στο κείμενο περιγραφής. Επίσης, σε αυτό το σημείο ορίσαμε και την εμφάνιση άλλων εικονιδίων που ακολουθούν σε άλλες σελίδες (`#others_img`) αλλά και επικεφαλίδες σε άλλα τμήματα της σελίδας (`images_intro h2` και `map_section h2`).

7.2 Φωτογραφίες

Στην συνέχεια, ο χρήστης θα μπορεί να μετακινηθεί προς τα κάτω στην σελίδα όπου θα μπορεί να δει φωτογραφικό υλικό με φωτογραφίες από τις αίθουσες και τα εργαστήρια του τμήματος καθώς και μια μικρή περιγραφή του ονόματος τους. Ο χρήστης έτσι θα μπορεί να πάρει μια γενική ιδέα για τις εγκαταστάσεις του τμήματος που τον ενδιαφέρει και να δει τους χώρους όπου πιθανόν να εργάζεται στο μέλλον.

Μια δεύτερη μπάρα πλοήγησης εμφανίζεται όταν ο χρήστης μετακινηθεί προς τα κάτω στην σελίδα με την οποία ο χρήστης θα μπορεί να περιηγείται στα τρία τμήματα της σελίδας (Περιγραφή, Φωτογραφίες και Χάρτες), με παρόμοιο τρόπο όπως και στην κεντρική σελίδα.



Διάγραμμα 43: Τμήμα οθόνης σελίδας με φωτογραφικό υλικό

7.2.1 Κώδικας HTML

```
<div class="container photos" id="photos">
  <div class="row images_intro">
    <div class="col-xs-4 col-sm-2 col-sm-offset-1 iconanim_h">
      
    </div>
    <div class="col-xs-8 col-sm-4 slideanim_h">
      <h2>#φωτογραφίες</h2>
    </div>
  </div>
  <div class="images_section">
    <div class="thumbnail image">
      <picture>
        <source srcset="pol1_sm.JPG" media="(max-width: 550px)">
        
      </picture>
      <div class="desc">Εργαστήριο Animation</div>
    </div>
    <div class="thumbnail image">
      <picture>
        <source srcset="pol2_sm.JPG" media="(max-width: 550px)">
        
      </picture>
      <div class="desc">Οπτικής Κοινωνιολογίας</div>
    </div>
  </div>
</div>
```

Διάγραμμα 44: HTML Κώδικας για τμήμα οθόνης με φωτογραφίες

Στο πάνω μέρος της οθόνης δημιουργείται και πάλι η επικεφαλίδα με τον τίτλο (Φωτογραφίες σε αυτή την περίπτωση) και το εικονίδιο που την αντιπροσωπεύει. Στην συνέχεια παρουσιάζονται οι φωτογραφίες, όπου ο ορισμός τους έγινε μέσα στο τμήμα με την

κλάση `image_section`. Από εκεί και πέρα κάθε φωτογραφία βρίσκεται μέσα σε ένα τμήμα με την κλάση `image` και την κλάση `thumbnail` του Bootstrap που χρησιμοποιείται για να δημιουργεί περιγράμματα γύρω από φωτογραφίες ή τμήματα που έχουν μέσα φωτογραφίες.

Στην συνέχεια χρησιμοποιήσαμε ένα καινούργιο στοιχείο της HTML το στοιχείο `picture` με το οποίο μπορούμε να ορίζουμε διαφορετικές φωτογραφίες, για διαφορετικές συσκευές ή διαφορετικά είδη οθονών. Για παράδειγμα, όταν ο χρήστης φορτώνει την σελίδα από μια μεγάλη οθόνη υπολογιστή όπου υπάρχουν περισσότερες υπολογιστικές δυνατότητες τότε επιθυμούμε στην σελίδα να φορτώνονται μεγαλύτερες σε διαστάσεις φωτογραφίες. Από την άλλη όμως, όταν ο χρήστης επισκέπτεται την σελίδα από μια συσκευή κινητού τηλεφώνου τότε αν χρησιμοποιούμε σε αυτή την περίπτωση τις ίδιες φωτογραφίες με την προηγούμενη περίπτωση, τότε θα φορτώνονται επιπλέον αχρείαστα δεδομένα που πιθανόν να κάνουν την φόρτωση της σελίδας πιο αργή. Έτσι, για πιο μικρές οθόνες, χρησιμοποιήσαμε μικρότερες φωτογραφίες.

Για την περίπτωση αυτών των φωτογραφιών, χρησιμοποιήσαμε φωτογραφίες τύπου `JPG` ο οποίος είναι ο καταλληλότερος τύπος για απεικόνιση φωτογραφιών στις οποίες εμφανίζεται τοπίο και περιέχουν έντονη λεπτομέρεια, περισσότερα χρώματα και γενικά περισσότερη πολυπλοκότητα.

Σύμφωνα με την σύνταξη του στοιχείο `<picture>`, όπως φαίνεται και στο διάγραμμα 44, περιέχει δύο στοιχεία, το στοιχείο `<source>` και το στοιχείο ``. Στο πρώτο δηλώνεται η φωτογραφία που θα φορτώνεται στην περίπτωση κινητών συσκευών με την χρήση της ιδιότητας `srcset`, κάτι που ελέγχεται με την χρήση `media query` (`media="(max-width: 550px)`), ενώ αν αυτή η συνθήκη δεν ισχύει (δηλαδή το πλάτος της οθόνης είναι μεγαλύτερο από `550 pixel`) τότε θα φορτώνεται η φωτογραφία που βρίσκεται στο στοιχείο ``.

7.2.2 Κώδικας CSS

Στις περιπτώσεις όπου είχαμε πολλές φωτογραφίες για ένα τμήμα, τότε χρησιμοποιήσαμε την κλάση `image` για να εμφανίζονται πέντε φωτογραφίες στην κάθε γραμμή με το κατάλληλο πλάτος η κάθε μια και με την κατάλληλη απόσταση μεταξύ τους. Παρακάτω βλέπουμε τον κώδικα με τον οποίο επιτυγχάνεται αυτό, όπως και ο έλεγχος της εμφάνισης των υπόλοιπων στοιχείων αυτής της οθόνης.

```

/*container photos section*/
.photos{ height: auto; }
#photos_icon{
  display: block;
  width: 80%;
  height: 90px;
}

/* photos display styling, and modal
.images_section{ margin-top: 70px; }
.image{
  padding: 0px 0px 10px 0px;
  width: 16%;
  float: left;
  margin: 0 2% 3% 2%;
}

.image img{
  display:block;
  cursor: pointer;
  width: 100%;
  height: 130px;
}
.desc{
  font-size:13px;
  padding: 8px;
  font-family: 'Nunito', sans-serif;
  text-align: center;
}

.image_facilities{ padding: 0px 0px 0px 0px;}

.image_others:first-of-type, .image_others:nth-of-type(5), .image_others:nth-of-type(9){
  margin-left: 12%;
}

```

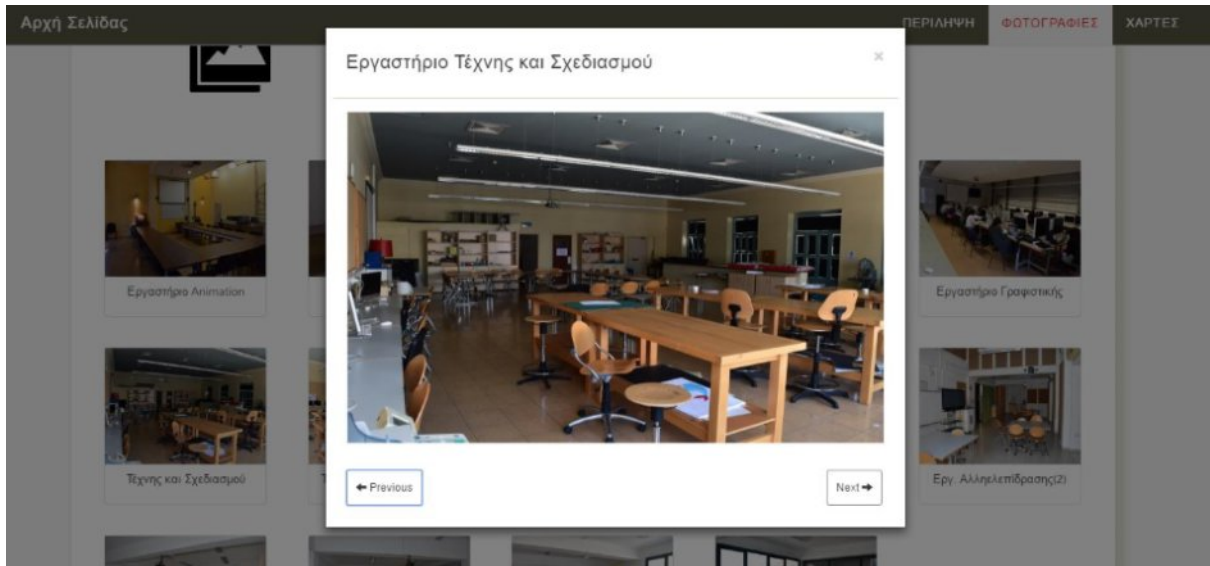
Διάγραμμα 45: Κώδικας CSS για εμφάνιση φωτογραφιών

Σημειώνεται επίσης ότι για τις περιπτώσεις όπου πιθανόν να μην είχαμε πολλές φωτογραφίες σε μια σελίδα προσθέσαμε τις κλάσεις `image_others` για να δημιουργούνται τέσσερις αντί πέντε φωτογραφίες σε κάθε γραμμή μετακινώντας αριστερά τις κατάλληλες φωτογραφίες, όπως φαίνεται στην παραπάνω εικόνα. Επίσης για τις φωτογραφίες όπου θέλαμε να μην υπάρχει περιγραφή στο κάτω μέρος χρησιμοποιήσαμε την κλάση `img_facilities`.

7.2.3 Παράθυρο διαλόγου modal

Μια άλλη δυνατότητα που προσφέρεται στον χρήστη είναι να μπορεί να πατά πάνω στις φωτογραφίες και να του εμφανίζεται στην οθόνη μια πιο μεγάλη έκδοση αυτών των φωτογραφιών μέσω ενός πλαισίου με κουμπιά `Next` και `Previous` για να μπορεί να περιηγείται μέσα στην συλλογή των φωτογραφιών. Για τον σκοπό αυτό, χρησιμοποιήσαμε ένα στοιχείο του Bootstrap, το `modal` το οποίο είναι ένα παράθυρο διαλόγου που εμφανίζεται στην οθόνη όταν ο χρήστης, στην περίπτωση αυτή, πατήσει πάνω στην φωτογραφία.

Στην συνέχεια βλέπουμε το παράθυρο αυτό και την μορφή που έχει όταν εμφανίζεται στην οθόνη του χρήστη, όπως και τους κώδικες σε HTML (αριστερά) και CSS (δεξιά) που ελέγχουν την δομή και την εμφάνιση του.



Διάγραμμα 46: Παράθυρο Διαλόγου για εμφάνιση φωτογραφιών

```

<!-- Modal -->
<div id="myModal" class="modal fade" role="dialog">

  <div class="modal-dialog">

    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-dismiss="modal">&times;</button>
        <p id="modal_title" class="slideanim_h"></p>
      </div>
      <div class="modal-body">
        <img id="modal_img" alt="modal_img">
        <div class="row">
          <div class="col-sm-2">
            <button type="button" id="prev_img" class="btn btn-default btn-sm slid
              <span class="glyphicon glyphicon-arrow-left"></span> Previous
            </button>
          </div>
          <div class="col-sm-2 col-sm-offset-8">
            <button type="button" id="next_img" class="btn btn-default btn-sm slid
              Next <span class="glyphicon glyphicon-arrow-right"></span>
            </button>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

.modal-content{
  width: 100%;
  height: auto;
  padding: 0px;
}

#modal_title{
  padding: 0px;
  letter-spacing: 0px;
  font-size: 15px;
}

.next, .prev{
  padding: 8px;
  border: 0.5px solid grey;
}

.prev{ float: left; }
.next{ float: right; }

#modal_img{
  width: 100%;
  height: auto;
  padding-bottom: 10px;
}

```

Διάγραμμα 47: Κώδικες HTML και CSS για Παράθυρο Διαλόγου

Για να καλεστεί το modal χρησιμοποιούμε δύο ιδιότητες που φαίνονται στο Διάγραμμα 44, τις `data-toggle="modal"` που δηλώνει ότι αυτό που πρόκειται να ανοίξει με αυτό τον σύνδεσμο είναι modal και την `data-target="#myModal"` για να αναφέρεται στο id του modal που επιθυμούμε να ανοίξει. Στην συνέχεια, η κλάση `modal` δηλώνει ότι αυτό το τμήμα περιέχει modal με την κλάση `fade` να ορίζει το animation εμφάνισης του, ενώ η κλάση `modal-dialog` ορίζει τις απαραίτητες διαστάσεις και περιθώρια του παραθύρου αυτού. Στο

τμήμα της κλάσης modal-content υπάρχει το περιεχόμενο του παραθύρου. Η κλάση modal-header ορίζει την επικεφαλίδα του modal που στην περίπτωση αυτή αποτελείται από την περιγραφή της φωτογραφίας και ένα κουμπί εξόδου από το modal (class="close" και data-dismiss="modal"), ενώ η κλάση modal-body αποτελεί το κυρίως μέρος του modal όπου περιλαμβάνει την φωτογραφία και τα κουμπιά Next και Previous.

```
$(document).ready(function(){

    // Get the modal
    var modal = document.getElementById('myModal');

    // Get the id of the modal img
    var modalImg = document.getElementById('modal_img');
    // Get all images and create an array with them
    var images_class = document.getElementsByClassName("image")
    var images = $(images_class).find('img');

    var newsrc;          var nextsrc;
    var newheader;      var prevsrc;
                        var img_index;

    /* code for finding the clicked image and put the 'src'
    attribute into the variable 'newsrc' */
    var i;
    for (i = 0; i < images.length; i++) {
        images[i].onclick = function(){
            newsrc = $(this).attr('src');
            newheader = $(this).attr('alt');
            img_index = $.inArray(this,images);
        };
    }

    /* when the modal is open assign the newsrc value to the src of the modal img*/
    $(modal).on('show.bs.modal', function (){
        $(modalImg).attr('src',newsrc);
        document.getElementById("modal_title").innerHTML = newheader;
        $(document.getElementById("modal_title")).addClass("modal_header_anim");
        if (img_index === (images.length-1))
        {
            $('#next_img').css("display","none");
        }
        else{
            $('#next_img').css("display","block");
            $('#next_img').addClass("button_next_anim");
        }
        if (img_index === 0)
        {
            $('#prev_img').css("display","none");
        }
        else{
            $('#prev_img').css("display","block");
            $('#prev_img').addClass("button_prev_anim");
        }
    });
});
```

Διάγραμμα 48: Κώδικας JavaScript για εμφάνιση modal

Πιο πάνω βλέπουμε τον κώδικα σε JavaScript σε συνδυασμό με συναρτήσεις της JQuery για τον χειρισμό των λειτουργιών που αφορούν το modal. Αρχικά δημιουργούμε κάποιες μεταβλητές οι οποίες θα αναφέρονται σε στοιχεία του HTML έγγραφου, όπως την φωτογραφία στην σελίδα, την εικόνα ή και την επικεφαλίδα του modal. Αυτό γίνεται χρησιμοποιώντας της μεθόδους `getElementById` και `getElementsByClassName` οι οποίες επιστρέφουν το στοιχείο που έχει την τιμή `id` ή κλάσης αντίστοιχα, που αναφέρεται μεταξύ των παρενθέσεων. Στην συνέχεια βρίσκουμε όλες τις εικόνες με το στοιχείο `img` κάτω από την κλάση `image` και τις βάζουμε μέσα στον πίνακα `images`.

Έπειτα, όταν ο χρήστης πατήσει πάνω σε κάποια εικόνα θα καλεστεί η συνάρτηση στην οποία αποθηκεύονται στις κατάλληλες μεταβλητές, οι τιμές της τοποθεσίας που είναι αποθηκευμένη η εικόνα (`scr`), της περιγραφής της (μέσω της ιδιότητας `alt`), καθώς και ο δείκτης που δείχνει την θέση της εικόνας μέσα στον πίνακα (μεταβλητές `newsrsc`, `newheader` και `img_index` αντίστοιχα).

Και αυτό έτσι ώστε όταν καλεστεί να ανοίξει το modal με την προκαθορισμένη συνάρτηση (`$(modal).on('show.bs.modal', function () { ... })`), οι τιμές αυτές να μεταφέρονται στις τιμές των στοιχείων του modal έτσι ώστε κάθε φορά να εμφανίζεται στο παράθυρο διαλόγου η φωτογραφία που πάτησε ο χρήστης μαζί με την περιγραφή της.

Επίσης βάση την θέση της εικόνας στον πίνακα ελέγχεται αν είναι η πρώτη ή η τελευταία εικόνα στην σειρά, έτσι ώστε να καθοριστεί να θα εμφανιστούν και τα δύο κουμπιά `Next` και `Previous` η αν θα εμφανιστεί μόνο ένα από τα δύο.

```

/* code when the 'next' button is clicked */
$(modal).on('click', ".next", function () {
    img_index = img_index + 1;
    nextsrc = $($ (images).eq(img_index)).attr('src');
    newheader = $($ (images).eq(img_index)).attr('alt');
    $(modalImg).attr('src',nextsrc);
    document.getElementById("modal_title").innerHTML = newheader;
    if (img_index === (images.length-1))
    {
        $('#next_img').css("display","none");
    }
    $('#prev_img').css("display","block");
});

/* code when the 'prev' button is clicked */
$(modal).on('click', ".prev", function () {
    img_index = img_index - 1;
    prevsrc = $($ (images).eq(img_index)).attr('src');
    newheader = $($ (images).eq(img_index)).attr('alt');
    $(modalImg).attr('src',prevsrc);
    document.getElementById("modal_title").innerHTML = newheader;
    if (img_index === 0)
    {
        $('#prev_img').css("display","none");
    }
    $('#next_img').css("display","block");
});

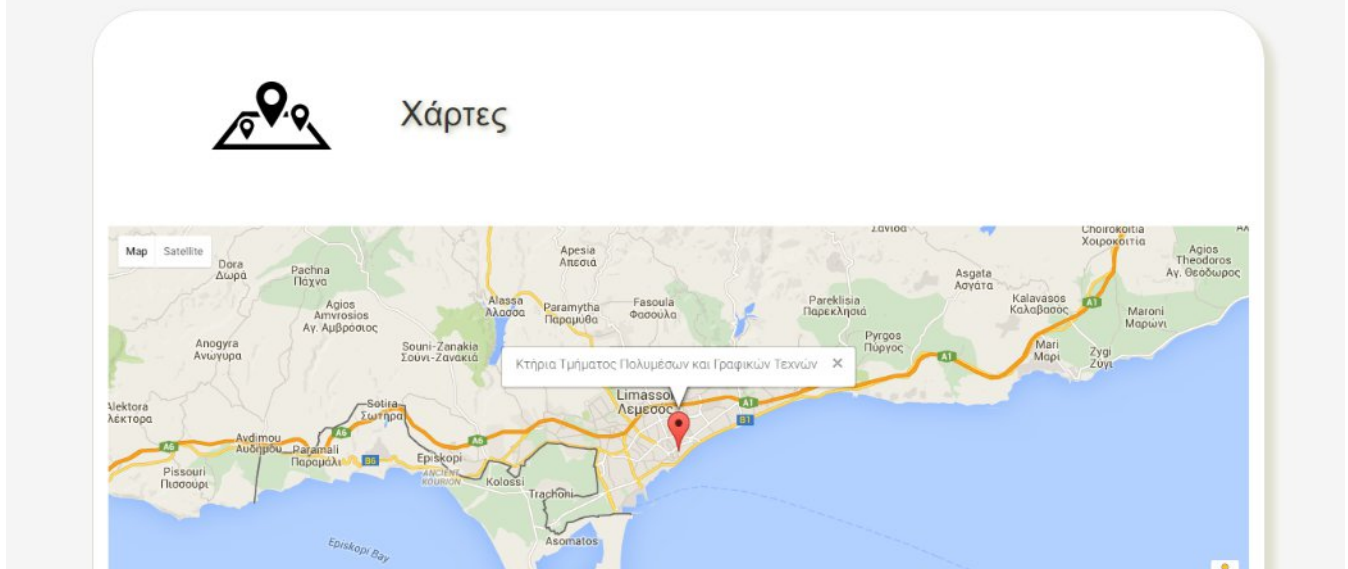
```

Διάγραμμα 49: Κώδικας για χειρισμό κουμπιών Next και Previous στο modal

Πιο πάνω, βλέπουμε τον κώδικα που εκτελείται όταν ο χρήστης πατήσει ένα από τα κουμπιά Next και Previous (\$(modal).on('click', ".next" ή ".prev", function ()) για να δει την επόμενη ή προηγούμενη φωτογραφία στην λίστα ανάλογα. Κάθε φορά που στο modal καλείτε να εμφανιστεί μια καινούργια φωτογραφία μαζί με την περιγραφή της, αυξάνεται ή μειώνεται ο δείκτης της θέσης της εικόνας στον πίνακα ανάλογα, έτσι ώστε να αλλάξουν οι τιμές των μεταβλητών που καθορίζουν τις τιμές των στοιχείων του modal. Επίσης και πάλι βάση του δείκτη, ελέγχεται αν ο χρήστης έφτασε στην πρώτη ή τελευταία φωτογραφία, έτσι ώστε να καθοριστεί αν θα εξαφανιστούν προσωρινά τα κουμπιά Previous και Next αντίστοιχα (καθώς αν για παράδειγμα φτάσει στην τελευταία φωτογραφία δεν θα υπάρχει επόμενη, έτσι δεν υπάρχει νόημα να υπάρχει για εκείνη την φωτογραφία το κουμπί Next).

7.3 Χάρτες

Μετά από το κομμάτι της σελίδας με το φωτογραφικό υλικό, ο χρήστης θα μπορεί να δει ένα χάρτη μέσω του οποίου μπορεί να δει τις τοποθεσίες των διαφόρων κτηρίων του τμήματος ή άλλων υπηρεσιών του Τεπακ.



Διάγραμμα 50: Οθόνη με Χάρτες

7.3.1 Κώδικας HTML και CSS

```

<div class="container map_section" id="maps">
  <div class="row map_intro">
    <div class="col-xs-4 col-sm-2 col-sm-offset-1 iconanim_h">
      
    </div>
    <div class="col-xs-8 col-sm-4 slideanim_h">
      <h2>Χάρτες</h2>
    </div>
  </div>
  <div id="googleMap"></div>
</div> <!-- End of end of map section -->

```

```

.map_section{
  height: auto;
  padding-top: 0px;
  padding-bottom: 20px;
}

#map_icon{
  display: block;
  width: 100%;
  height: 120px;
}

.map_section h2{ font-size: 33px; }

#googleMap{
  margin-top: 20px;
  height: 450px;
}

```

Διάγραμμα 51: Οθόνη με χάρτες - Κώδικας HTML και CSS

Πιο πάνω βλέπουμε τον κώδικα HTML και CSS που ελέγχει την δομή και την εμφάνιση αυτού του τμήματος της σελίδας. Μετά από την επικεφαλίδα και το εικονίδιο της, ορίσαμε ένα τμήμα με id ίσο με googleMap στο οποίο θα εμφανίζεται ο χάρτης που θα φορτώνεται μέσω JavaScript.

7.3.2 Κώδικας JavaScript – Χρήση Google Maps API

```
var myCenter = new google.maps.LatLng(34.678749, 33.045191);
var myCenter2 = new google.maps.LatLng( 34.678994, 33.045344);

function initialize()
{
    /* set the variable parameters for the map */
    var mapProp = {
        center:myCenter,
        zoom:11,
        mapTypeId:google.maps.MapTypeId.ROADMAP
    };

    /* Load the map into the specified html element (googleMap) */
    var map=new google.maps.Map(document.getElementById("googleMap"),mapProp);

    /* Create a market to add to the map in the specified position */
    var marker=new google.maps.Marker({ position:myCenter });
    var marker2=new google.maps.Marker({ position:myCenter2 });

    /* Creates an informational window to open when user clicks on the marker */
    var infowindow = new google.maps.InfoWindow({ content:"Κτήριο Ρούσος" });
    infowindow.open(map,marker);
    google.maps.event.addListener(marker, 'click', function(){ infowindow.open(map,marker); });

    var infowindow2 = new google.maps.InfoWindow({content:"Κτήρια Τμήματος Πολυμέσων και Γραφικών Τεχνών" });
    infowindow2.open(map,marker2);
    google.maps.event.addListener(marker2, 'click', function(){ infowindow2.open(map,marker2); });

    marker.setMap(map);
    marker2.setMap(map);
}

google.maps.event.addDomListener(window, 'load', initialize);
```

Διάγραμμα 52: Κώδικας JavaScript για Google Maps API

Χρησιμοποιώντας ένα σύνολο μεθόδων που παρέχονται από μια βιβλιοθήκη JavaScript, το Google Maps API, δημιουργήσαμε χάρτη με σημάδια για τις τοποθεσίες των διαφόρων κτηρίων των τμημάτων του Τεπακ.

Η βιβλιοθήκη JavaScript φορτώθηκε στην σελίδα μέσω της ακόλουθης δήλωσης που βρίσκεται στην επικεφαλίδα <head> του κάθε HTML εγγράφου:

```
<script src="http://maps.googleapis.com/maps/api/js?key=AIzaSyCheLXi4aVKbR8K0e-CaE9y-8MEzCBohLY"></script>
```

Αρχικά δημιουργούμε τις μεταβλητές (myCenter, myCenter2 κλπ) που θα αντιστοιχούν στις συντεταγμένες των κτηρίων που θέλουμε να βάλουμε στο χάρτη. Στην συνέχεια δημιουργούμε ένα αντικείμενο που θα παίρνει ως παραμέτρους μια από αυτές τις μεταβλητές για να την θέσει ως κέντρο του χάρτη, την παράμετρο zoom για να καθορίζει το επίπεδο εστίασης και την παράμετρο mapTypeId με την οποία καθορίζεται το είδος του χάρτη (στην περίπτωση αυτή έχουμε ένα κανονικό δυσδιάστατο χάρτη). Έπειτα δημιουργούμε ένα

αντικείμενο map το οποίο θα δημιουργήσει ένα χάρτη στο τμήμα της σελίδας που θέσαμε στο HTML έγγραφο (με id=googleMap), στο οποίο περνούμε τις παραμέτρους που δημιουργήσαμε πιο πάνω.

Στην συνέχεια δημιουργούμε σημάδια που αντιπροσωπεύουν με το αντίστοιχο εικονίδιο πάνω στον χάρτη τις συντεταγμένες των κτηρίων (βάση των μεταβλητών my Center), και δημιουργήσαμε αυτά τα σημάδια στον χάρτη με την συνάρτηση setMap().

Επίσης, με τις μεταβλητές infowindow δημιουργούμε ένα κείμενο που θα περιέχει τον τίτλο του κτηρίου το οποίο συνδέουμε με τον αντίστοιχο του σημάδι που δημιουργήσαμε πιο πάνω. Με την χρήση της μεθόδου addListener, το κείμενο αυτό εμφανίζεται ή εξαφανίζεται ανάλογα, όταν ο χρήστης πατήσει πάνω στα αντίστοιχα σημάδια/εικονίδια στον χάρτη,

Τέλος με την χρήση της μεθόδου addDomListener εκτελείται η συνάρτηση initialize() για την δημιουργία του χάρτη και των στοιχείων του, όταν φορτωθεί η σελίδα.

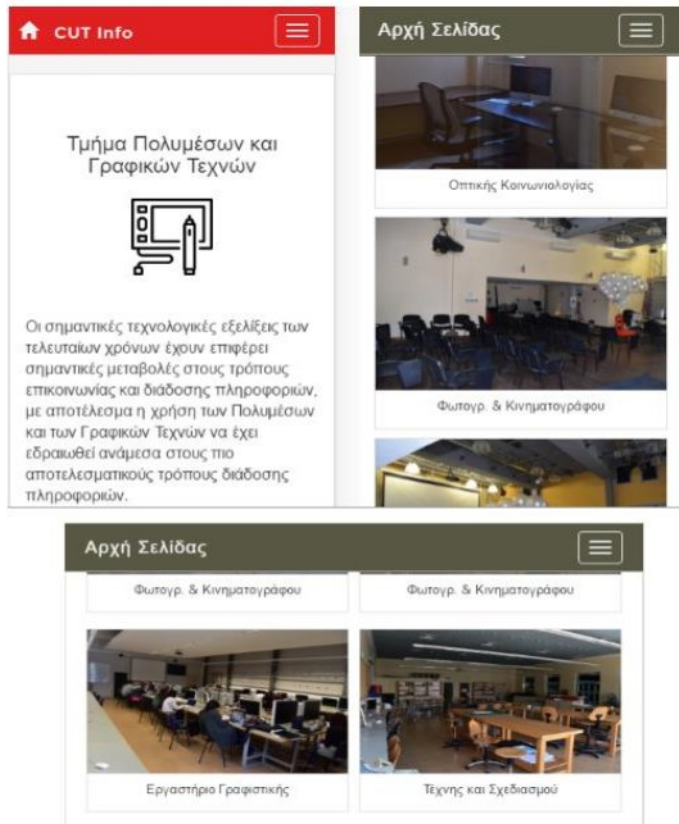
7.4 Σχεδίαση σελίδων για άλλες συσκευές – responsive design

Όπως και στις υπόλοιπες περιπτώσεις, οι κανόνες CSS για συσκευές με μεγέθη οθονών κινητών τηλεφώνων ορίστηκαν σε άλλο αρχείο, και πάλι με την χρήση media query.

Σε αυτές τις συσκευές οι κύριες αλλαγές που έχουν γίνει εκτός από την κατάλληλη μορφοποίηση των στοιχείων για να ανταποκρίνεται η εμφάνιση τους σε τέτοια μεγέθη οθονών είναι η φόρτωση διαφορετικών φωτογραφιών όπως αναλύσαμε πιο πάνω με την χρήση του στοιχείο picture και το γεγονός ότι στο τμήμα της σελίδας με το φωτογραφικό υλικό, θα φαίνεται κάθε φορά μια εικόνα ανά γραμμή (θέτοντας τους καταλληλους κανόνες στις κλάσεις image και image_others). Για τις περιπτώσεις οριζοντίου(landscape) orientation, λόγω του ότι η οθόνη θα είναι πιο πλατιά θα φαίνονται δύο φωτογραφίες ανά γραμμή.

Επίσης η επικεφαλίδα με τον τίτλο του τμήματος και το εικονίδιο του θα εμφανίζονται η μια στήλη πάνω στην άλλη, βάση των δηλώσεων που κάναμε στο HTML έγγραφο με την χρήση του grid συστήματος του Bootstrap. Ακόμη το κείμενο θα παίρνει όλη την έκταση της οθόνης, όπως και ο χάρτης στο τρίτο τμήμα της σελίδας.

Για τις οθόνες tablet η δομή και εμφάνιση της σελίδας παραμένει η ίδια και για οποιεσδήποτε μικρές αλλαγές έγιναν ορίστηκαν οι ανάλογοι κανόνες CSS στο ίδιο αρχείο με την χρήση media queries.



Διάγραμμα 53: Εμφάνιση σελίδας Τμημάτων σε κινητά τηλέφωνα

```

.images_intro h2{
  font-size: 30px;
  text-shadow: none;
}

/* photos display styling, and modal */

.images_section{ margin-top: 30px; }

.image_others:first-of-type, .image_others:nth-of-type(5),
.image_others:nth-of-type(9){ margin-left: 0%; }

.image, .image_others{
  padding: 0px 0px 6px 0px;
  border-radius: 0;
  width: 100%;
  margin: 0% 0% 3% 0%;
}
image img{ height: 180px; }

.map_section{
  height: auto;
  padding-top: 10px;
}

#map_icon{
  width: 100%;
}

.map_section h2{
  font-size: 30px;
  text-shadow: none;
}

#googleMap{
  width: 100%;
  margin-top: 10px;
  height: 400px;
}

.content, .photos, .map_section, .content_cb{
  padding-top: 40px;
  box-shadow: none;
  border-radius: 0px;
}

#dep_img{
  margin: 8% auto;
  width: 33%;
  height: 80px;
}

.content_header h2, .content_cb h1{
  font-size: 21px;
  text-align: center;
  text-shadow: none;
}

.content_header h2{
  width: 100%;
}

.content_text p:first-of-type{
  margin-top: 20px;
}

.content_text p{
  width: 100%;
  margin: 10px auto;
  font-size: 15.5px;
}

```

Διάγραμμα 54: Κώδικας CSS - οθόνες κινητών συσκευών

8 Χρήση Animation

8.1 Εισαγωγή - Χρησιμότητα Animation

Εκτός από τον καθορισμό της εμφάνισης και της μορφής των διαφόρων στοιχείων, έμφαση δόθηκε επίσης και στην δημιουργία διαφόρων animation για κάποια από τα στοιχεία των σελίδων. Animation των στοιχείων είναι η δημιουργία κίνησης στα στοιχεία, είτε κατά την εμφάνιση τους στην οθόνη, είτε κατά την εκτέλεση μιας πράξης από τον χρήστη.

Η χρήση animation στις ιστοσελίδες αποτελεί ένα ενδιαφέρον και πολύτιμο εργαλείο για τον σχεδιαστή ιστοσελίδων. Σε γενικές γραμμές, ενσωματώνοντας τα στις σελίδες μας, δημιουργείται ένα πιο ευχάριστο περιβάλλον για τον χρήστη, κάτι που κάνει την περιήγηση του στις διάφορες σελίδες μας, λόγω των διαφόρων κινήσεων που υπάρχουν, να μοιάζει περισσότερο με συνθήκες πραγματικού κόσμου.

Όπως προαναφέραμε, τα animation μπορούν είτε να δημιουργούνται από μόνα τους χωρίς να απαιτείται κάποια πράξη από τον χρήστη και γίνονται κυρίως κατά την φόρτωση της σελίδας, είτε να είναι μικρές κινήσεις των στοιχείων, ή αλλαγή στη εμφάνιση τους που προκαλείται όταν ο χρήστης αλληλεπιδρά με την ιστοσελίδα. Δυο είναι οι κύριοι ρόλοι που έχουν τα animation μέσα σε μια ιστοσελίδα. Αρχικά μπορούν να καθοδηγούν τον χρήστη για το πως μπορεί να χρησιμοποιήσει την ιστοσελίδα ή για το πως μπορεί να περιηγηθεί μέσα σε αυτή. Έπειτα, μπορούν να χρησιμοποιηθούν και ως διακοσμητικό μέσο για την ιστοσελίδα βελτιώνοντας την εμφάνιση του περιεχομένου της και αυξάνοντας έτσι το ενδιαφέρον του χρήστη για αυτή.

Υπάρχουν όμως και κάποιοι παράγοντες που λήφθηκαν υπόψη κατά την δημιουργία των animation για τις σελίδες μας. Αρχικά, θα πρέπει να υπάρχει ένα μέτρο για το πόσο πολύπλοκα θα είναι. Και αυτό γιατί όταν ένα animation αποτελείται από πολύπλοκες και υπερβολικές κινήσεις τότε θα καταλήξει να γίνεται κουραστικό για τον χρήστη, προκαλώντας έτσι μια δυσάρεστη παρά ευχάριστη εμπειρία για αυτόν. Γενικά, τα animations θα πρέπει να έχουν ένα ξεκάθαρο λόγο ύπαρξης, να είναι ομαλά, να υπηρετούν με τον καταλληλότερο τρόπο τον χρήστη και να μην στέκονται εμπόδιο στην παρουσίαση της ουσίας του περιεχομένου μιας ιστοσελίδας.

8.2 Ιδιότητες CSS3 για animation

Για την δημιουργία των animation χρησιμοποιήσαμε εκτός από κώδικα JavaScript, και κάποιους κανόνες και ιδιότητες που περιλαμβάνει η CSS3.

Μια από αυτές τις ιδιότητες είναι η ιδιότητα transform η οποία μας επιτρέπει να αλλάζουμε κάποια χαρακτηριστικά ενός στοιχείου όπως για παράδειγμα το μέγεθος του, το σχήμα του, την κλίση του και την τοποθεσία του. Οι αλλαγές αυτές μπορούν να γίνονται είτε με δισδιάστατες(2D Transforms) είτε με τρισδιάστατες(3D Transforms) μετατροπές.

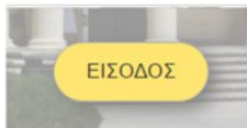
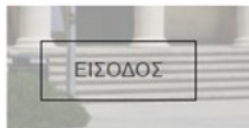
Επίσης, μια άλλη ιδιότητα είναι τα transitions (μεταβάσεις), ιδιότητα με την οποία μπορούμε να αλλάζουμε τις τιμές άλλων ιδιοτήτων ενός στοιχείου σταδιακά.

Ακόμη, χρησιμοποιήσαμε την ιδιότητα animation για να δημιουργούμε animations στα διάφορα στοιχεία μιας σελίδας, αλλάζοντας δηλαδή σταδιακά την εμφάνιση τους από μια κατάσταση σε άλλη. Η χρήση αυτής της ιδιότητας φαίνεται να είναι παρόμοια με την ιδιότητα transition, όμως διαφέρουν στο ότι ένα animation μπορεί να γίνει και κατά την φόρτωση της σελίδας, ενώ η ιδιότητα transition χρησιμοποιείται συνήθως όταν ένα στοιχείο αλλάζει η εμφάνιση του λόγω μιας πράξης που έκανε ο χρήστης. Επίσης ένα animation μπορεί να εκτελεστεί πολλές φορές, με διαφορετική φορά(δηλαδή ξεκινώντας από το τέλος) και γενικά μας προσφέρει περισσότερες δυνατότητες.

8.3 Παραδείγματα

Στην συνέχεια θα δούμε περισσότερες λεπτομέρειες για αυτές τις ιδιότητες μέσω παραδειγμάτων από animation που δημιουργήσαμε για το website αυτό.

8.3.1 Ιδιότητα transition και transition-duration



```
.mybutton {  
  background: none;  
  border-color: black;  
  border-width: 1.5px;  
  border-radius: 0px;  
  font-size: 17px;  
  padding: 13px 26.5px;  
  -webkit-transition-duration: 0.6s;  
  transition-duration: 0.6s;  
}  
  
.mybutton:hover , .mybutton:focus {  
  background-color: #FFE571;  
  border-radius: 30px;  
  box-shadow: 6px 6px 25px grey;  
}
```

Διάγραμμα 55: Παράδειγμα 1 με χρήση animation

Πιο πάνω βλέπουμε το πως αλλάζει η εμφάνιση του κουμπιού «Είσοδος» στην αρχική σελίδα, όταν ο χρήστης θα περάσει τον δρομέα του ποντικιού πάνω από το κουμπί. Οι ιδιότητες που θα αλλάζουν φαίνονται κάτω από τον κώδικα της μεθόδου hover και focus(.mybutton:hover , .mybutton:focus) οι οποίες ελέγχουν το τι θα αλλάξει στην εμφάνιση ενός στοιχείου όταν ο δρομέας του ποντικιού του χρήστη περάσει πάνω από το στοιχείο αυτό ή όταν το στοιχείο είναι πατημένο αντίστοιχα. Συγκεκριμένα το χρώμα του φόντου του κουμπιού θα αλλάξει σε ελαφρύ πορτοκαλί με την δήλωση background-color: #FFE571;, η κλίση των γωνιών θα αλλάξει ενώ επίσης θα δημιουργείται σκιά πίσω από το κουμπί με την ιδιότητα box-shadow.

Με την χρήση όμως τις ιδιότητας transition δημιουργείται μια διάρκεια στην μετάβαση από την μια κατάσταση εμφάνισης στην άλλη κάτι που κάνει το animation να φαίνεται πιο ομαλό. Συγκεκριμένα χρησιμοποιούμε την ιδιότητα transition-duration και θέτουμε την τιμή της να είναι 0,6s. Το πρόθεμα -webkit- στην δεύτερη δήλωση χρησιμοποιείται για να υποστηρίζεται η ιδιότητα από άλλους web browsers.

8.3.2 Ιδιότητα animation και keyframes

```
<div id="history" class="row info">
  <div class="slideanim_m">
    <h1>Ιστορία</h1>

    <div class="col-sm-7 text_history">
      <p>
        Το Τεχνολογικό Πανεπιστήμιο Κύπρου (ΤΕΠΑΚ) είναι ένα νέο δημόσιο Πανεπιστήμιο, που ιδρύθηκε με νόμο
        Έδρα του Πανεπιστημίου είναι η Λεμεσός όπου λειτουργούν και οι έξι Σχολές του και ο συναγωνισμός γι
      </p>
      <p>
        Είναι ένα σχετικά καινούργιο πανεπιστήμιο όπου η ίδρυση του αποτέλεσε μια προσπάθεια να καλυφθούν τα
        προσφέροντας προπτυχιακές και μεταπτυχιακές σπουδές σε επίπεδο που δεν προσφέρεται από άλλο ακαδημαϊ
      </p>
    </div>
  </div>
  <div class="col-sm-5 img_content img_content1 iconanim"></div>
</div>

.slideanim_m{ visibility:hidden; }

.slide_content{
  animation-name: slide_content;
  -webkit-animation-name: slide_content;
  animation-duration: 2.1s;
  -webkit-animation-duration: 2.1s;
  visibility: visible;
}

@keyframes slide_content {
  0% {
    opacity: 0;
    transform: translate3d(0px,130px,0px);
  }
  100% {
    opacity: 1;
    transform: translate3d(0px,0px,0px);
  }
}
@-webkit-keyframes slide_content {
  0% {
    opacity: 0;
    -webkit-transform: translate3d(0px,130px,0px);
  }
  100% {
    opacity: 1;
    -webkit-transform: translate3d(0px,0px,0px);
  }
}

.iconanim{ visibility: hidden; }

.icon_slide{
  animation-name: icon_slide;
  -webkit-animation-name: icon_slide;
  animation-duration: 1.8s;
  -webkit-animation-duration: 1.8s;
  visibility: visible;
  animation-timing-function: ease-out;
  -webkit-animation-timing-function: ease-out;
}

@keyframes icon_slide {
  0% {
    opacity: 0;
    transform: scale3d(0.2,0.2,1);
  }
  100% {
    opacity: 1;
    transform: scale3d(1,1,1);
  }
}
@-webkit-keyframes icon_slide {
  0% {
    opacity: 0;
    -webkit-transform: scale3d(0.2,0.2,1);
  }
  100% {
    opacity: 1;
    -webkit-transform: scale3d(1,1,1);
  }
}
```

Διάγραμμα 56: Παράδειγμα 2: με χρήση animation

Το πιο πάνω παράδειγμα είναι από την κεντρική σελίδα, με τις γενικές πληροφορίες για το Τεπακ. Το animation αυτό δημιουργείται όταν ο χρήστης μετακινηθεί στο τμήμα της σελίδας με τις γενικές πληροφορίες και καθορίζει το πως θα εμφανιστεί στην οθόνη το κείμενο και το εικονίδιο που το αντιπροσωπεύει.

Παραθέτουμε και πάλι τον κώδικα HTML για αυτό το σημείο της οθόνης όπου φαίνονται οι κλάσεις `slideanim_m` και `iconanim` που ορίσαμε για τα στοιχεία που αφορούν το κείμενο και το εικονίδιο αντίστοιχα. Μέσω της CSS, δηλώσαμε την ιδιότητα `visibility` αυτών των κλάσεων να είναι ίση με `hidden` έτσι ώστε αρχικά να μην εμφανίζονται τα στοιχεία αυτά. Στην συνέχεια χρησιμοποιώντας JQuery ελέγξαμε το πότε θα προσθέτονται οι κλάσεις στα στοιχεία που θα δημιουργούν το animation.

```
$(window).scroll(function() {
    var winTop = $(window).scrollTop();
    var pos;

    $(".slideanim_m").each(function(){
        pos = $(this).offset().top;

        if (pos < winTop + 500) {
            $(this).addClass("slide_content");
        }
    });

    $(".iconanim").each(function(){
        pos = $(this).offset().top;

        if (pos < winTop + 500) {
            $(this).addClass("icon_slide");
        }
    });
});
```

Διάγραμμα 57: Κώδικας JQuery για έλεγχο εμφάνισης animation

Συγκεκριμένα ο κώδικας αυτός χρησιμοποιεί δύο μεταβλητές, την `winTop` η οποία ελέγχει το πόσο μετακινήθηκε ο χρήστης προς τα κάτω στην σελίδα και την `pos` η οποία ελέγχει την θέση του στοιχείου που θέλουμε να προστεθεί σε αυτό η καινούργια κλάση. Όταν ο χρήστης φτάσει στο κατάλληλο σημείο τότε με την χρήση της κλάσης `addClass` προστίθεται η ανάλογη κλάση στο στοιχείο.

Μετά από αυτό, μέσω των πιο πάνω δηλώσεων που φαίνονται στο Διάγραμμα 56 δημιουργείται το animation. Με την ιδιότητα `animation-name` ορίζουμε το όνομα του animation και την διάρκεια του με την ιδιότητα `animation-duration`. Επίσης το στοιχείο γίνεται πλέον εμφανίσιμο με την ιδιότητα `visibility` και την τιμή `visible`. Στην συνέχεια το animation εκτελείται χρησιμοποιώντας τους κανόνες `keyframes`. Αυτοί οι κανόνες καθορίζουν την εμφάνιση των στοιχείων σε συγκεκριμένες στιγμές κατά την διάρκεια του animation, έτσι ώστε να αλλάζει η εμφάνιση των στοιχείων σταδιακά και ομαλά. Στην περίπτωση αυτή χρησιμοποιήσαμε την ιδιότητα `transform` με την μέθοδο `translate3d(x,y,z)` που καθορίζει την θέση του στοιχείου στον τρισδιάστατο χώρο και την ιδιότητα `opacity` που καθορίζει το πόσο διαφανές είναι. Για την περίπτωση του κειμένου (`slide_content`) όπως παρατηρούμε και από τον κώδικα, στην αρχική του κατάσταση, δηλαδή στο 0% το στοιχείο ξεκινά 130 pixel κάτω από τον κάθετο άξονα και με `opacity 0` (δηλαδή είναι εντελώς αδιαφανές) και σταδιακά προχωρά προς τα πάνω μέχρι να φτάσει στην τελική του

κατάσταση, δηλαδή στο 100%. Για την περίπτωση των εικονιδίων (icon_slide) χρησιμοποιήσαμε την μέθοδο scale3d(x,y,z) της ιδιότητας transform για να ξεκινά το εικονίδιο με μέγεθος μικρότερο του κανονικού του μέχρι να φτάνει σιγά σιγά στις κανονικές του διαστάσεις. Ακόμη, με την χρήση της ιδιότητας animation-timing-function ορίσαμε την ταχύτητα με την οποία εξελίσσεται το animation και συγκεκριμένα με την τιμή ease-out θα έχει κανονική ροή με το τέλος να είναι αργό.

Άλλα animation που χρησιμοποιήσαμε ήταν για παράδειγμα η εμφάνιση του μηνύματος καλωσορίσματος όταν φορτώνεται η αρχική σελίδα, ή η εμφάνιση της δεύτερης μπάρας πλοήγησης στην κεντρική και στις σελίδες των σχολών και τμημάτων με την οποία ο χρήστης μπορεί να περιηγηθεί μέσα στις σελίδες.

Επίσης, χρησιμοποιήσαμε κώδικα JQuery για να δημιουργούμε ομαλές μεταβάσεις μέσα στις σελίδες, όταν ο χρήστης μεταβαίνει μέσω συνδέσμου από ένα σημείο της σελίδας σε άλλο.

```
$(document).ready(function(){
  // Add smooth scrolling to all links in navbar + footer link
  $(".nav_second a, .header2, .navigation a, .upnav a[href='#page_start'],
  .carousel_button, .footer a[href='#page_start']").on('click', function(event) {

  // Prevent default anchor click behavior
  event.preventDefault();

  // Store hash
  var hash = this.hash;

  // Using jQuery's animate() method to add smooth page scroll
  // The optional number (700) specifies the number of milliseconds it takes to scroll to the specified area
  $('html, body').animate({
    scrollTop: $(hash).offset().top
  }, 700, function(){

    // Add hash (#) to URL when done scrolling (default click behavior)
    window.location.hash = hash;
  });
});
```

Διάγραμμα 58: Κώδικας JQuery για ομαλή μετάβαση σε σημεία εντός μια σελίδας

Σύμφωνα με τον παραπάνω κώδικα, όταν ο χρήστης πατήσει σε κάποιο σύνδεσμο καλείται μια συνάρτηση στην οποία χρησιμοποιήσαμε την ιδιότητα hash και την μέθοδο animate() για να μεταβαίνει ο χρήστης στο σημείο που επέλεξε. Συγκεκριμένα η ιδιότητα hash αποθηκεύει στην μεταβλητή hash που επιλέξαμε την τοποθεσία στην οποία θα μεταβεί ο χρήστης και με την μέθοδο animate() και την ιδιότητες scrollTop και top εκτελείται το animation που θα κάνει την ομαλή αυτή μετακίνηση από το ένα σημείο της σελίδας στο άλλο.

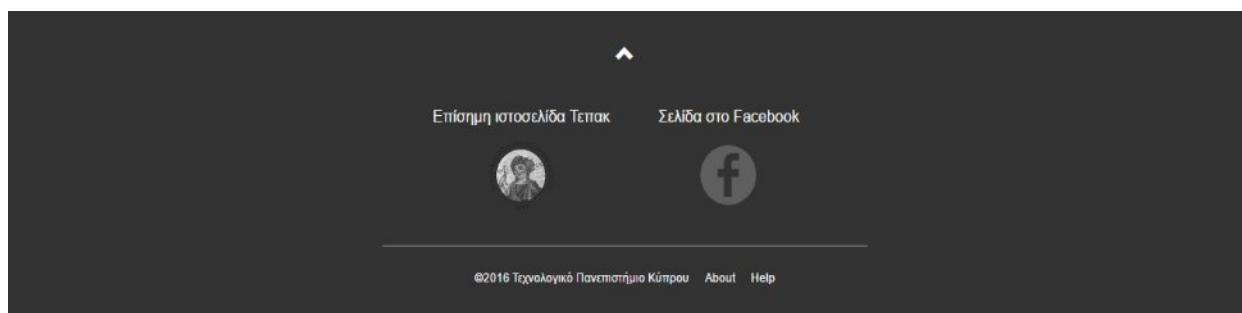
9 Δημιουργία Footer

9.1 Πρακτικές δημιουργίας footer

Το footer παρόλο που δεν αποτελεί το πιο σημαντικό κομμάτι μιας ιστοσελίδας, είναι κάτι που δεν θα πρέπει να αγνοηθεί καθώς είναι το κομμάτι μέσω του οποίου οι χρήστες μπορούν να βρουν κάποιες επιπλέον πληροφορίες που χρειάζονται.

Και για αυτό τον λόγο υπάρχουν κάποιες πρακτικές που θα ήταν χρήσιμο να εφαρμόζονται κατά την δημιουργία του. Αρχικά, πρέπει να είναι έχει μια απλή σχεδίαση και να συμβαδίζει η εμφάνιση του με το υπόλοιπο κομμάτι της ιστοσελίδας. Επίσης πρέπει να είναι σωστά οργανωμένο, να υπάρχει η κατάλληλη απόσταση μεταξύ των συνδέσμων που περιλαμβάνει και ο χρήστης να μπορεί να διαβάζει και να βρίσκει εύκολα τις πληροφορίες που χρειάζεται. Οι πολύπλοκες διατάξεις και η ύπαρξη επιπλέον και αχρείαστων πληροφοριών είναι κάτι που συστήνεται να αποφεύγεται έτσι ώστε να μην καταντήσει η παρουσία του footer ενοχλητική για τον χρήστη. Γενικά ο footer είναι καλό να παρέχει τις απαιτούμενες πληροφορίες με τον καλύτερο και πιο αποτελεσματικό τρόπο σε όσο πιο μικρό χώρο είναι δυνατό.

9.2 Δημιουργία footer – κώδικας HTML και CSS



Διάγραμμα 59: Οθόνη Footer

Το footer, είναι το τελευταίο τμήμα όλων των σελίδων αυτού του website και περιλαμβάνει κάποιες γενικές πληροφορίες για αυτό και χρήσιμους συνδέσμους για άλλες ιστοσελίδες που έχουν σχέση με το Τεπακ.

Όπως φαίνεται και στην πιο πάνω οθόνη, αρχικά τοποθετήσαμε ένα εικονίδιο ενός τόξου με κατεύθυνση προς τα πάνω (με την χρήση glyphicon από το Bootstrap) το οποίο ο χρήστης θα μπορεί να πατήσει για να μεταβεί στην αρχή της σελίδας. Έπειτα ακολουθεί μια διάταξη με

την χρήση του grid system του Bootstrap, όπου σε μια γραμμή και δύο στήλες θα φαίνονται σύνδεσμοι για τις δύο εξωτερικές ιστοσελίδες που θα μπορεί να επισκεφθεί αν επιθυμεί ο χρήστης. Τέλος δημιουργήσαμε μια οριζόντια λίστα στην οποία υπάρχουν οι σύνδεσμοι με επιπλέον πληροφορίες για το website, όπως επίσης και ένα κείμενο με την ειδοποίηση στην οποία αναγράφονται τα πνευματικά δικαιώματα της ιστοσελίδας. Τα δύο αυτά τμήματα του footer διαχωρίζονται μεταξύ τους με γραμμή διαχώρισης που δημιουργήθηκε χρησιμοποιώντας το στοιχείο `<hr>`. Πιο κάτω φαίνεται ο αντίστοιχος HTML κώδικας.

```

<!-- Footer -->
<div class="footer container-fluid text-center">
  <a href="#page_start" title="To Top">
    <span class="glyphicon glyphicon-chevron-up"></span>
  </a>
  <div class="row linkrow">
    <div class="col-sm-2 col-sm-offset-4 link1 slideanim_f">
      <a href="http://www.cut.ac.cy/?languageId=1">
        <h4>Επίσημη ιστοσελίδα Τεπακ</h4>
        
      </a>
    </div>
    <div class="col-sm-2 link2 slideanim_f">
      <a href="https://www.facebook.com/CyprusUniversityofTechnology">
        <h4>Σελίδα στο Facebook</h4>
        
      </a>
    </div>
  </div>
  <hr class="hr_footer">

  <ul class="navfooter">
    <li>©2016 Τεχνολογικό Πανεπιστήμιο Κύπρου</li>
    <li><a href="about.html">About</a></li>
    <li><a href="help.html">Help</a></li>
  </ul>
</div>
<!-- End of footer -->

```

Διάγραμμα 60: Κώδικας HTML για Footer

Ακολουθεί ο κώδικας CSS με τον οποίο καθορίσαμε την εμφάνιση των στοιχείων του Footer. Συγκεκριμένα ορίσαμε το χρώμα του φόντου του Footer να είναι μαύρο και τα γράμματα να έχουν χρώμα παραλλαγής του άσπρου έτσι ώστε να ξεχωρίζουν. Επίσης ορίσαμε τις κατάλληλες αποστάσεις που θα έχουν μεταξύ τους τα στοιχεία, ενώ θέσαμε το μέγεθος των γραμμάτων να είναι μικρό καθώς αποτελούν δευτερεύον πληροφορίες. Δεν θα πρέπει όμως να είναι υπερβολικά μικρό το κείμενο έτσι ώστε να είναι ευανάγνωστο προς τον χρήστη για να βρίσκει εύκολα και γρήγορα αυτό που επιθυμεί. Στην συνέχεια, με την δήλωση `display: inline-block` ορίσαμε τα στοιχεία της λίστας να είναι στοιχισμένα οριζόντια. Ακόμη, χρησιμοποιήσαμε την ιδιότητα `text-decoration` με την τιμή `none` για να μην υπογραμμίζεται το κείμενο όταν ο χρήστης περνά με το ποντίκι του πάνω από τους συνδέσμους, αλλά να αλλάζει μόνο το χρώμα.

```

.footer{
  background:#333;
  padding: 25px;
  text-align: center;
}

.footer .glyphicon {
  font-size: 20px;
  color: white;
  padding: 10px;
}

.footer .glyphicon:hover{
  color: orange;
  -webkit-transition-duration: 0.6s;
  transition-duration: 0.6s;
}

.linkrow{ margin-top: 2%; }

hr.hr_footer{
  width: 40%;
  height: 1px;
  margin: 20px auto;
  background-color: #e6e6e6;
  border-radius: 90px;
  opacity: 0.3;
}

.footer h4{
  color: whitesmoke;
  font-family: 'Nunito', sans-serif;
  font-size: 16px;
}

/* footer icon styling */
.link1 img{
  display:block;
  margin: 1% auto;
  -webkit-filter: grayscale(100%);
  filter: grayscale(100%);
}

.link2 img{
  display:block;
  margin: 20px auto 20px;
  width: 60px; height: 60px;
  -webkit-filter: grayscale(100%);
  filter: grayscale(100%);
}

.link1 img:hover, .link2 img:hover{
  -webkit-filter: grayscale(0%);
  filter: grayscale(0%);
  -webkit-transition-duration: 0.6s;
  transition-duration: 0.6s;
}

.link1 a:hover, .link2 a:hover{
  text-decoration: none;
}

.navfooter{ padding-left: 0px;}

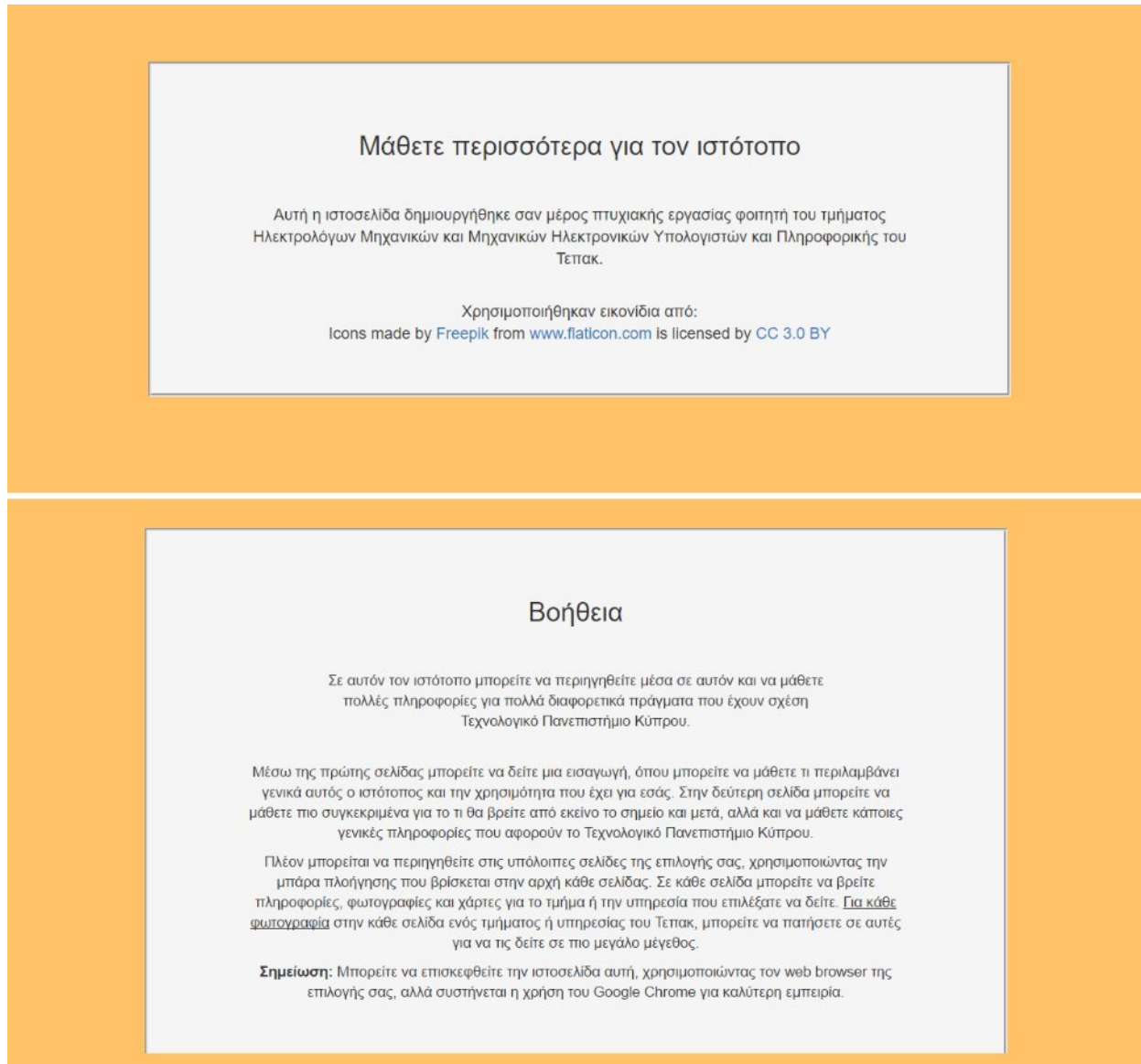
.navfooter a:link,
.navfooter a:visited,
.navfooter li
{
  display: inline-block;
  margin-left: 1%;
  color: whitesmoke;
  font-family: 'Nunito', sans-serif;
  font-size: 12px;
}

.navfooter li:first-of-type{
  margin-left: 0px;
}

.navfooter li a:hover{
  color: orange;
  text-decoration: none;
  -webkit-transition-duration: 0.7s;
  transition-duration: 0.7s;
}

```

Διάγραμμα 61: Κώδικας CSS για Footer



Διάγραμμα 62: Σελίδες Help και About

Πιο πάνω φαίνονται οι οθόνες των σελίδων Βοήθεια(Help) και Μάθετε περισσότερα(About) με την μια σελίδα να δίνει οδηγίες χρήσης και την άλλη πληροφορίες που αφορούν την δημιουργία του website, αντίστοιχα. Τα κείμενα αυτών των σελίδων είναι στοιχισμένα στο κέντρο, με πορτοκαλί φόντο για να συμβαδίζει με τα χρώματα των προηγούμενων σελίδων.

9.3 Εμφάνιση Footer σε μικρότερες συσκευές

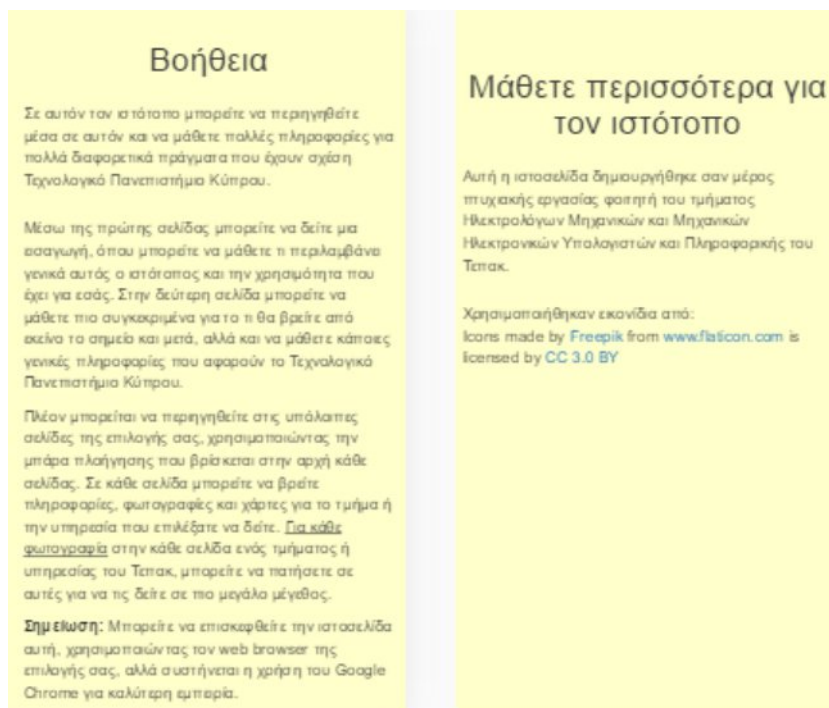
Πιο κάτω φαίνεται η εμφάνιση του Footer σε μικρότερες συσκευές και συγκεκριμένα στα κινητά τηλέφωνα , σε portrait orientation καθώς και ο κώδικας σε CSS που ελέγχει την εμφάνιση αυτή. Στην συνέχεια βλέπουμε επίσης την εμφάνιση των σελίδων Βοήθεια(Help) και Μάθετε περισσότερα(About).



Διάγραμμα 63: Εμφάνιση Footer σε κινητά τηλέφωνα και κώδικας CSS

Όπως ορίσαμε στο έγγραφο HTML σύμφωνα με το grid system του Bootstrap, στις συσκευές με μικρότερο μέγεθος το τμήμα που αφορά τους εξωτερικούς συνδέσμους που περιλαμβάνει το footer θα φαίνεται με δύο στήλες με την μια να ακολουθεί την άλλη, σε κάθετη διάταξη. Επίσης με την δήλωση `display: block` τοποθετούμε τα στοιχεία της λίστας στο τέλος σε κάθετη διάταξη αυτή τη φορά. Γενικά κάναμε τις απαραίτητες αλλαγές στους κανόνες έτσι ώστε τα στοιχεία να προσαρμόζονται καλύτερα σε μια πιο κάθετη διάταξη.

Για την περίπτωση της προβολής της σελίδας σε συσκευές tablet, η εμφάνιση του footer είναι ίδια με κάποιες μικρές αλλαγές να έχουν γίνει έτσι ώστε να γίνεται η κατάλληλη προσαρμογή.



Διάγραμμα 64: Σελίδες Help και About σε μικρότερες συσκευές

ΣΥΜΠΕΡΑΣΜΑΤΑ/ ΑΠΟΤΕΛΕΣΜΑΤΑ/ ΕΠΙΛΟΓΟΣ

Στα πλαίσια αυτής της πτυχιακής εργασίας δημιουργήσαμε ένα website με το οποίο θα μπορούν οι επισκέπτες του να μαθαίνουν για κάποιες σημαντικές πληροφορίες που αφορούν το Τεπακ μέσω κειμένου, φωτογραφικού υλικού αλλά και με χάρτες μέσα σε ένα περιβάλλον φιλικό και ευχάριστο προς τον χρήστη.

Αναλύσαμε την σημαντικότητα που έχει η δημιουργία ιστοσελίδων σε σχέση με άλλα είδη εφαρμογών, καθώς στην σημερινή εποχή η χρήση τους είναι πολύ έντονα διαδεδομένη λόγω των τεράστιων ωφελειών και πλεονεκτημάτων που προσφέρουν.

Μελετήσαμε επίσης διάφορες σύγχρονες μεθόδους σχεδιασμού ιστοσελίδων και εμβαθύνουμε τις γνώσεις μας σε θέματα τεχνολογιών ιστού για ανάπτυξη και δημιουργία εφαρμογών αυτού του είδους, στις οποίες η εμφάνιση τους θα προσαρμόζεται ανάλογα σε όλα τα είδη συσκευών και μεγεθών οθονών.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Refsnes Data, “W3Schools Online Web Tutorials,” 1998. [Online]. Available: <http://www.w3schools.com/>.
- [2] “Διαδίκτυο,” *Wikipedia*. .
- [3] “Γνωριμία με το Διαδίκτυο και τις υπηρεσίες του.” [Online]. Available: http://hermes.di.uoa.gr/exe_activities/diakiktio/_____.html.
- [4] “Ιστοσελίδα,” *Wikipedia*. .
- [5] “Παγκόσμιος Ιστός,” *Wikipedia*. .
- [6] Jennifer Kyrnin, “What are Markup Languages?,” *December 10*, , 2014. [Online]. Available: <http://webdesign.about.com/od/htmlxhtmltutorials/p/what-are-markup-languages.htm>.
- [7] “CSS,” *Wikipedia*. .
- [8] “HTML,” *Wikipedia*. .
- [9] “Διαδικτυακή εφαρμογή,” *Wikipedia*. .
- [10] “Web browser,” *Wikipedia*. .
- [11] NetBeans IDE, “NetBeans IDE Features.” [Online]. Available: <https://netbeans.org/features/html5/>.
- [12] “Bootstrap,” *Wikipedia*. .
- [13] tutorialspoint, “Client-side JavaScript Advantages of JavaScript.”
- [14] tutorialspoint, “jQuery Overview.” [Online]. Available: <http://www.tutorialspoint.com/jquery/jquery-overview.htm>.
- [15] Ακης Καργιοφύλλης, “Γενικά για την HTML,” 2005. [Online]. Available: <http://www.wlearn.gr/index.php/home-html-30>.
- [16] “Responsive design,” *Wikipedia*. .
- [17] Niels van Hove, “What are CSS Media Queries and how to implement them.” [Online]. Available: <http://cssmediaqueries.com/what-are-css-media-queries.html>.

- [18] Jessica Meher, “15 of the Best Website Homepage Design Examples,” *January 21*, 2015. [Online]. Available: <http://blog.hubspot.com/blog/tabid/6307/bid/34006/15-Examples-of-Brilliant-Homepage-Design.aspx#sm.0010bsitu1at2erqxq51wdg3grlr9>.
- [19] Code My Views, “15 Tips for Creating a Great Website Footer.”
- [20] Carrie Cousins, “An Introduction to Animation in Web Design,” 2015. [Online]. Available: <https://designshack.net/articles/graphics/an-introduction-to-animation-in-web-design/>.
- [21] James Cole, “Using Icons in Web Development: The move to SVG,” 2014. [Online]. Available: <https://newsignature.com/articles/using-icons-in-web-development-the-move-to-svg>.
- [22] Pearl Chen, “Built-in Browser Support for Responsive Images,” 2014. [Online]. Available: <http://www.html5rocks.com/en/tutorials/responsive/picture-element/>.
- [23] Eric Portis, “Responsive Images Done Right,” 2014. [Online]. Available: <https://www.smashingmagazine.com/2014/05/responsive-images-done-right-guide-picture-srcset/>.
- [24] Brad Frost, “Responsive Navigation Patterns,” 2012. [Online]. Available: <http://bradfrost.com/blog/web/responsive-nav-patterns/>.
- [25] JOSH BYERS, “Mobile Responsive Design 101,” *March 7*, 2013. [Online]. Available: <http://www.copyblogger.com/mobile-responsive-design-101/>.
- [26] Shay Howe, “Positioning Content HTML and CSS,” 2016. [Online]. Available: <http://learn.shayhowe.com/html-css/positioning-content/>.