



Τμήμα
Μηχανικών
Πληροφορικής τ.ε.

Τεχνολογικό Εκπαιδευτικό Ίδρυμα
Δυτικής Ελλάδας

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Σχεδιασμός και Ανάπτυξη Ολοκληρωμένης Εφαρμογής
Παγκόσμιου Ιστού για την Διαχείριση των Πτυχιακών
Εργασιών του Τμήματος

Χαρίλαος Δημητρουλόπουλος

Επιβλέπων καθηγητής: Βασίλειος Ταμπακάς

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή
Αντίρριο,

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

- 1.
- 2.
- 3.

Ευχαριστίες

Σε αυτό το σημείο θα ήταν αξιοσημείωτο να αναφέρω τους ανθρώπους που με στήριξαν και με βοήθησαν καθ' όλη την διάρκεια της συγγραφής της παρούσας πτυχιακής εργασίας.

Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή, κ. Βασίλειο Ταμπακά, ο οποίος δέχτηκε να με καθοδηγήσει και να με βοηθήσει με την δική του οπτική γωνία αν και ανέλαβε το ρόλο του επιβλέποντα καθηγητή προς το τέλος της πτυχιακής.

Στο σημείο αυτό θα ήθελα να ευχαριστήσω και τον προηγούμενο επιβλέπων καθηγητή κ. Γιάννη Τζήμα, που με εμπιστεύτηκε για την εκπόνηση της παρούσας πτυχιακής. Οι πολύτιμες συμβουλές του όσων αφορά την δομή και το περιεχόμενο της παρούσας πτυχιακής και η εν γένη συνεργασία αποτέλεσαν για εμένα κίνητρο για περαιτέρω έρευνα στο γνωστικό πεδίο δημιουργίας ενός ιστοχώρου για την διαχείριση των πτυχιακών.

Τέλος, θα ήθελα να ευχαριστήσω και την οικογένειά μου που με στήριξαν και με βοήθησαν τόσο κατά την διάρκεια της συγγραφής της πτυχιακής όσο και καθ' όλη την διάρκεια των σπουδών μου. «Σας ευχαριστώ που με βοηθήσατε να υλοποιήσω έναν από τους στόχους μου»

Περιεχόμενα

Ευχαριστίες.....	2
Περιεχόμενα	3
1 Ο Παγκόσμιος Ιστός και οι Ιστοσελίδες που τον απαρτίζουν.....	9
1.1 Ο Παγκόσμιος Ιστός (Word Wide Web).....	9
1.2 Το Internet	9
1.3 Ιστοσελίδα	10
1.4 Κατηγορίες Ιστοσελίδων	10
1.5 Στατικές Ιστοσελίδες	10
1.6 Δυναμικές Ιστοσελίδες.....	10
2 Γλώσσες Προγραμματισμού.....	11
2.1 HTML.....	11
2.1.1 Βασικές ετικέτες	13
2.1.2 Φόρμες	15
2.2 CSS.....	17
2.3 PHP.....	19
2.3.1 Superglobal variables	20
2.3.2 Εκτύπωση και τελεστής συνένωσης.....	21
2.3.3 Πίνακες	21
2.4 MySQL.....	22
2.5 Javascript	24
2.6 Συνδυασμός των παραπάνω τεχνολογιών	25
2.6.1 Ανάκτηση και προβολή δεδομένων από τη βάση.....	26
2.6.2 Αποθήκευση δεδομένων από φόρμα	27
2.7 Wampserver	28
3 CMS (Content Management System).....	30
3.1 Στόχος του CMS	30
3.2 Δομή του CMS.....	30
3.2.1 Βάση Δεδομένων (Data Base).....	31
3.2.2 Ο πυρήνας (Core)	31
3.2.3 Template.....	31

3.2.4	Plug-ins	31
4	Εφαρμογή Διαχείρισης Πτυχιακών Εργασιών	31
4.1	Σκοπός.....	32
4.2	Ρόλος.....	32
4.3	Ορισμός σχήματος βάσης δεδομένων	33
4.4	Ανάπτυξη εφαρμογής	34
4.4.1	Αρχεία header.php και footer.php.....	35
4.4.2	Αρχείο db_connect.php	35
4.4.3	Αρχείο Style.css	36
4.4.4	Αρχείο Register.php	36
4.4.5	Αρχείο index.php (login)	37
4.4.6	Αρχείο logout.php	39
4.4.7	Αρχείο check_login.php	39
4.4.8	Αρχείο admin.php	40
4.4.9	Αρχείο supervisors.php.....	41
4.4.10	Αρχείο students.php	42
4.4.11	Αρχείο list.php.....	42
4.4.12	Αρχείο by_sv.php	43
4.4.13	Αρχείο view.php.....	44
4.4.14	Αρχείο sv_theses.php	44
4.4.15	Αρχείο create.php	45
4.4.16	Αρχείο confirm.php.....	46
4.4.17	Αρχείο approve.php.....	47
4.4.18	Αρχείο st_thesis.php.....	48
4.4.19	Αρχείο select1.php.....	49
4.4.20	Αρχείο select2.php.....	50
4.4.21	Αρχείο submit.php	51
	Βιβλιογραφία.....	53
	Παράρτημα	55
1.	Αρχείο register.php	55
2.	Αρχείο index.php	57
3.	Αρχείο logout.php.....	59
4.	Αρχείο check_login.php.....	60

5.	Αρχείο admin.php.....	Error! Bookmark not defined.
6.	Αρχείο supervisors.php	61
7.	Αρχείο students.php.....	62
8.	Αρχείο list.php	63
9.	Αρχείο by_sv.php.....	Error! Bookmark not defined.
10.	Αρχείο view.php	65
11.	Αρχείο sv_theses.php.....	67
12.	Αρχείο create.php.....	68
13.	Αρχείο confirm.php	71
14.	Αρχείο approve.php	72
15.	Αρχείο st_thesis.php	74
16.	Αρχείο select1.php	75
17.	Αρχείο select2.php	76
18.	Αρχείο submit.php.....	78
19.	Αρχείο header.php	81
20.	Αρχείο footer.php.....	83
21.	Αρχείο db_connect.php.....	84
22.	Αρχείο style.css.....	85
23.	Αρχείο theses.sql	87

Λίστα Εικόνων

Εικόνα 2.1.1-1 Παγκόσμιος Ιστός	9
Εικόνα 2.1.1-1 HTML.....	12
Εικόνα 2.1.2-1 CSS	17
Εικόνα 2.1.2-1 PHP.....	19
Εικόνα 2.3.3-1 MySQL.....	22
Εικόνα 2.3.3-1 JavaScript	24
Εικόνα 2.3.3-1 Τα βήματα που εκτελούνται όταν ο περιηγητής ζητήσει μια σελίδα php.....	25
Εικόνα 2.6.2-1 Wampserver	28
Εικόνα 4.4.4-1 Φόρμα εγγραφής με συμπληρωμένα τα στοιχεία	37
Εικόνα 4.4.5-1 Είσοδος Χρήστη	38
Εικόνα 4.4.5-2 Φόρμα εισόδου με μήνυμα λάθους μετά από ανεπιτυχή προσπάθεια εισόδου.....	39
Εικόνα 4.4.5-3 Μήνυμα για μη ενεργό λογαριασμό	39
Εικόνα 4.4.8-1 Η σελίδα του διαχειριστή σε προβολή των μη ενεργοποιημένων χρηστών.....	40
Εικόνα 4.4.8-2 Η σελίδα του διαχειριστή σε προβολή των ενεργοποιημένων χρηστών.....	41
Εικόνα 4.4.9-1 Λίστα επιβλεπόντων	41
Εικόνα 4.4.10-1 Λίστα των φοιτητών	42
Εικόνα 4.4.11-1 Λίστα Πτυχιακών	43
Εικόνα 4.4.12-1 Πτυχιακές Συγκεκριμένου καθηγητή.....	43
Εικόνα 4.4.13-1 Προβολή όλων των λεπτομερειών μιας πτυχιακής εργασίας.....	44
Εικόνα 4.4.14-1 Το μενού του επιβλέποντα <<Οι πτυχιακές μου>>	45
Εικόνα 4.4.15-1 Η φόρμα για δημιουργία ή επεξεργασία πτυχιακής από τον επιβλέποντα	46
Εικόνα 4.4.16-1 Αναμονή Έγκρισης(από καθηγητή)	47
Εικόνα 4.4.16-2 Τελική Έγκριση.....	47
Εικόνα 4.4.17-1 Σελίδα προβολής και έγκρισης υποβληθείσας πτυχιακής εργασίας	48
Εικόνα 4.4.18-1 Η σελίδα των φοιτητών <<Η Πτυχιακή μου>>	48
Εικόνα 4.4.19-1 Επιλογή Πτυχιακής - 1ο Βήμα.....	49
Εικόνα 4.4.19-2 Επιλογή έως και 5 πτυχιακές.....	49
Εικόνα 4.4.19-3 Επιλέχθικαν 5 πτυχιακές.....	50
Εικόνα 4.4.20-1 Επιλογή Πτυχιακής - 2ο Βήμα.....	50
Εικόνα 4.4.21-1 Φόρμα υποβολής πτυχιακής.....	51
Εικόνα 4.4.21-2 Μήνυμα σφάλματος λόγο κενής περίληψης	52
Εικόνα 4.4.21-3 Μη επιτυχές ανέβασμα αρχείου	52
Εικόνα 4.4.21-4 Επιτυχές ανέβασμα αρχείου	52

Περίληψη

Στην παρούσα πτυχιακή εργασία παρουσιάζεται η ανάγκη για τον προγραμματισμό καθώς και οι πέντε γλώσσες υψηλού επιπέδου , όπως είναι οι HTML, Css, Php, MySQL, Javascript. Εκτός από την παρουσίαση αυτών γίνεται και η παρουσίαση των πέντε τεχνολογιών παρουσιάζοντας τα ωφελεί από τον συνδυασμό τους. Τέλος, γίνεται βιβλιογραφική αναφορά για τον WampServer που αποτελεί τον πυλώνα για την ανάπτυξη της εφαρμογής διαχείρισης πτυχιακών εργασιών. Σήμερα παρατηρείται μια τάση για ανάπτυξη διαδικτυακών εφαρμογών. Οι εφαρμογές αυτού του είδους είναι πιο εύχρηστες, καθώς δεν απαιτούν εγκατάσταση και επιτρέπουν την αλληλεπίδραση μεταξύ διαφορετικών χρηστών. Στην παρούσα πτυχιακή εργασία δημιουργήθηκε ένα διαδικτυακό σύστημα διαχείρισης πτυχιακών εργασιών. Το σύστημα που αναπτύχθηκε επιτρέπει στους καθηγητές να δημιουργήσουν και να διαχειριστούν πτυχιακές εργασίες, ενώ ταυτόχρονα δίνει τη δυνατότητα στους φοιτητές να περιηγηθούν στις πτυχιακές εργασίες, διαθέσιμες και ολοκληρώμενες, να επιλέξουν την επιθυμητή εργασία κατόπιν έγκρισης από τον επιβλέποντα και να υποβάλουν την τελική της έκδοση, ώστε να είναι διαθέσιμη στους υπόλοιπους χρήστες. Η υλοποίηση έγινε σε PHP-HTML ενώ χρησιμοποιήθηκε η MySQL ως βάση δεδομένων.

Abstract

In this thesis, the need for programming as well as the five high-level languages such as HTML, Css, Php, MySql and JavaScript will be presented. Apart from this, we shall also illustrate the benefits of combining the five technological structures with the languages. Lastly, a bibliography-based refence will be given regarding the WampServer, the pylon for the development of the application for the administration of thesis projects. Nowadays, a trend towards evolving applications on the Internet, is highly noticeable. These types of applications are more user-friendly as they, on the one hand, do not require installation but, on the other hand, simply permit interaction between different users. In this project, on Internet-based system, regulating the administration of thesis projects, has been created. This system allows lecturers to initiate and handle all graduate projects while it simultaneously enables students to read through existing, available and comprehensive projects so as to select their desired one in accordance with their supervisor's agreement.

1 Ο Παγκόσμιος Ιστός και οι Ιστοσελίδες που τον απαρτίζουν

1.1 Ο Παγκόσμιος Ιστός (Word Wide Web)



Εικόνα 2.1.1-1 Παγκόσμιος Ιστός

Η ιστορία του Παγκόσμιου Ιστού είναι σχετικά πρόσφατη και ξεκινάει ειδικότερα από το 1989 με τον εφευρέτη Tim Berners-Lee που αναπτύσσει την τεχνολογία διαμοιρασμού πληροφοριών μέσω υπερσυνδεδεμένων εγγράφων κειμένου[1]. Μετά την υλοποίηση του Παγκόσμιου Ιστού το διαδίκτυο απέκτησε έναν πιο πλούσιο κόσμο με αμέτρητες δυνατότητες που κανείς δεν πίστευε πριν ότι θα μπορούσαν να υλοποιηθούν. Το web (όπως αποκαλείτε στις μέρες μας εν συντομία) είναι ένα δίκτυο υπολογιστών με παγκόσμια εμβέλεια, που δίνει την δυνατότητα στους υπολογιστές να μπορούν να επικοινωνούν μεταξύ τους για την υλοποίηση διάφορων στόχων. Για να επιτευχθεί αυτός ο δεσμός επικοινωνίας χρειαζόμαστε την στήριξη ενός πρωτοκόλλου επονομαζόμενο ως HTTP το οποίο προήλθε από τα αρχικά Hyper Transfer Protocol.

1.2 Το Internet

Αν θα χρησιμοποιούνταν μια μεταφορά για να χαρακτηρίσει το ιντερνέτ, θα το παρομοίαζε σαν ένα ιστό αράχνης, του οποίου οι άκρες θα επεκτεινόταν σε κάθε γωνιά του κόσμου. Τα έντομα πάνω στον ιστό θα ήταν οι χρήστες που ανταλλάσσουν πληροφορίες. Είναι αξιοσημείωτο να

αναφέρουμε πως για συγκεκριμένες λειτουργίες το ιντερνέτ χρειάζεται την βοήθεια του πρωτοκόλλου TCP/IP.

1.3 Ιστοσελίδα

Η ιστοσελίδα, η οποία φιλοξενείται στο web, δεν είναι κάτι διαφορετικό από τον χώρο στον οποίο φυλάσσεται και προβάλλεται η ηλεκτρονική πληροφορία που αναζητήσαμε. Συνήθως αυτές οι σελίδες περιέχουν πληροφορίες ευρύ τύπου. Για την διευκόλυνση των χρηστών οι σελίδες αλληλοεπιδρούν μεταξύ τους με το πάτημα ενός απλού κουμπιού.

1.4 Κατηγορίες Ιστοσελίδων

Οι Ιστοσελίδες χωρίζονται σε δύο κατηγορίες, ανάλογα με τον σκοπό που προσφέρουν τόσο στον χρήστη όσο και στο διαδίκτυο. Χωρίζονται σε Στατικές και Δυναμικές. Εμφανισιακά δεν μπορεί κάποιος να εντοπίσει αισθητές αλλαγές παρά μόνο άμα ελέγξει τις λειτουργίες τους.

1.5 Στατικές Ιστοσελίδες

Με τον όρο στατική ιστοσελίδα εννοούμε ότι το περιεχόμενο της δεν αλλάζει με κάποιον εύκολο και γνωστό τρόπο που θα τον ήξεραν χρήστες, χωρίς τις απαραίτητες γνώσεις προγραμματισμού ιστοσελίδων. Για να γίνει μια τροποποίηση σε μια στατική ιστοσελίδα χρειάζεται κάποιος Web designer. Μια στατική ιστοσελίδα μπορεί να εμπεριέχει στο περιεχόμενο της αρχεία σε μορφή βίντεο. Συνήθως την χρησιμοποιούν Web site που δεν χρειάζεται το περιεχόμενό τους να είναι συχνά ανανεώσιμο, για αυτό το λόγο ονομάζεται στατική. Και αυτό είναι ταυτόχρονα και το μειονέκτημά της.[2]

1.6 Δυναμικές Ιστοσελίδες

Είναι ένα website το οποίο μπορούμε να το τροποποιήσουμε μέσα από έναν φυλλομετρητή (browser). Μας προσφέρει την δυνατότητα να του αλλάξουμε περιεχόμενο σε ζωντανή στιγμή, δηλαδή την ώρα που το βλέπουμε. Απαιτούνται βασικές γνώσεις επεξεργασίας κειμένου από τον

χρήστη. Δυναμικές Ιστοσελίδες χρησιμοποιούμε σε web site που το περιεχόμενο τους αλλάζει συχνά (4 μέρες).[2]

2 Γλώσσες Προγραμματισμού

Με την ανάπτυξη της τεχνολογίας των Η/Υ δημιουργήθηκε η ανάγκη για περισσότερες διεργασίες και ταχύτερα αποτελέσματα στα ανθρώπινα προβλήματα. Ο τρόπος παρέμβασης του ανθρώπου στην μηχανή γίνεται με τον προγραμματισμό. Ο προγραμματισμός δεν είναι τίποτε άλλο από ένα σύνολο εντολών που δίνει ο άνθρωπος - προγραμματιστής στην μηχανή - Η/Υ ώστε να "τρέχει" πολύπλοκους αλγόριθμους για την επίλυση των προβλημάτων.

Στην σύγχρονη κοινωνία οι ανθρώπινες ανάγκες που προκύπτουν συνεχώς αυξάνονται και για το λόγο αυτό απαιτούνται όλο και πιο σύνθετη αλγόριθμοι στα πολύπλοκα προβλήματα. Το γεγονός αυτό οδήγησε στην ανάπτυξη μέσων για την επίλυση τους και έδωσε ώθηση για την ανάπτυξη του πεδίου της γλώσσας προγραμματισμού. Αρχικά οι γλώσσες προγραμματισμού ήταν χαμηλού επιπέδου ωστόσο με το πέρασμα του χρόνου δημιουργήθηκαν πιο σύγχρονες και υψηλότερου επιπέδου γλώσσες. Στη συνέχεια της ανάλυσης θα παρουσιαστούν οι εξής γλώσσες προγραμματισμού: HTML, PHP, CSS, MySQL, Javascript.

2.1 HTML



Εικόνα 2.1.1-1 HTML

Η HTML (HyperText Markup Language) δεν είναι μια γλώσσα προγραμματισμού (για παράδειγμα δεν έχει μεταβλητές και έλεγχο ροής), αλλά είναι μια γλώσσα σήμανσης (markup language) [3]. Δηλαδή είναι ένας ειδικός τρόπος γραφής κειμένου και χρησιμοποιείται για να καθορίσει τον τρόπο εμφάνισης του κειμένου σε μια ιστοσελίδα. Όλες οι ιστοσελίδες αποτελούνται από στοιχεία της HTML. Η HTML χρησιμοποιεί τις ειδικές ετικέτες (tags) να δώσει τις απαραίτητες οδηγίες στον περιηγητή. Ο περιηγητής (browser) είναι ένα πρόγραμμα που διαβάζει και εκτελεί τις εντολές που περιέχονται στο HTML αρχείο, ώστε να εμφανίσει το τελικό αποτέλεσμα στο χρήστη. Οι ετικέτες είναι εντολές που συνήθως ορίζουν την αρχή ή το τέλος μιας λειτουργίας και βρίσκονται πάντα μεταξύ των συμβόλων “<” και “>”, π.χ.
. Οι οδηγίες είναι case insensitive, δηλαδή δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά ή κεφαλαία γράμματα. Οι περισσότερες ετικέτες HTML λειτουργούν ανά ζεύγη, για παράδειγμα <p> και </p>, και ονομάζονται ετικέτα έναρξης και ετικέτα λήξης αντίστοιχα, αν και υπάρχουν ετικέτες που λειτουργούν μόνες τους χωρίς ετικέτα λήξης. Ανάμεσα στις ετικέτες τοποθετείται κείμενο το οποίο διαμορφώνεται για προβολή σύμφωνα με τις ετικέτες (το <i>πλάγιο</i> θα εμφανιστεί ως πλάγιο). Η HTML επιτρέπει επίσης την εμφάνιση άλλων αντικειμένων μέσα στη σελίδα, όπως εικόνες. Επιπλέον παρέχει τα μέσα δημιουργίας δομημένων εγγράφων καθορίζοντας στοιχεία όπως κεφαλίδες, παραγράφους, λίστες (αριθμημένες ή με κουκίδες), πίνακες, συνδέσμους, φόρμες κλπ. Ένα αρχείο HTML έχει συνήθως την κατάληξη htm ή HTML και μπορεί να δημιουργηθεί χρησιμοποιώντας απλώς έναν επεξεργαστή κειμένου.

Η HTML δημιουργήθηκε το 1993 από τον Tim Berners Lee, ο οποίος επίσης δημιούργησε το πρωτόκολλο www (world wide web). Για να μπορούν οι browser να ερμηνεύουν σωστά την HTML έχουν θεσπιστεί κάποιες προδιαγραφές, ώστε κάθε είδος υπολογιστή και περιηγητή να

μπορεί να δείξει με τον ίδιο τρόπο μια ιστοσελίδα. Βέβαια, ακόμη και σήμερα μπορεί να υπάρχουν μικρές διαφορές στην απεικόνιση κάποιας σελίδας από διαφορετικούς browsers. Σήμερα η τελευταία έκδοση είναι η HTML 5, ενώ χρησιμοποιείται ευρέως η HTML 4.

Οι δυνατότητες των ιστοσελίδων επεκτείνονται με την ενσωμάτωση κώδικα όπως η JavaScript, ο οποίος εκτελείται στον περιηγητή και μπορεί να επηρεάζει δυναμικά τη συμπεριφορά των ιστοσελίδων HTML. Οι περιηγητές μπορούν επίσης να ενσωματώνουν στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη της ιστοσελίδας. Θα αναφερθούμε στις δύο αυτές τεχνολογίες ξεχωριστά παρακάτω. Παρακάτω θα αναφέρουμε συνοπτικά τις βασικές ετικέτες της HTML [4].

2.1.1 Βασικές ετικέτες

Όταν θέλουμε να δημιουργήσουμε μια σειρά εντολών σε HTML, ώστε να δημιουργηθεί μια ιστοσελίδα, αρχίζουμε πάντα με την ετικέτα <HTML>. Με αυτό το τρόπο γίνεται φανερό στον περιηγητή ότι από εδώ και πέρα βρίσκεται κώδικας HTML που πρέπει να μεταφραστεί κατάλληλα. Όταν τελειώσει αυτή η σειρά ετικετών HTML ακολουθεί η ετικέτα </HTML> οπότε ο περιηγητής σταματάει τη διαμόρφωση του κειμένου. Συνήθως μία τυπική ιστοσελίδα περιέχει τρεις βασικές ετικέτες: το head, το title και το body. Το <head> τοποθετείται στην αρχή της σελίδας και περικλείει ετικέτες που περιέχουν πληροφορίες προς τον περιηγητή χωρίς να γίνονται ορατές στο χρήστη και όταν τελειώσουν αυτές οι ετικέτες δηλώνεται ομοίως το </head>. Το <title> και </title> είναι οι ετικέτες η οποίες δηλώνουν τον τίτλο της ιστοσελίδας, ο οποίος εμφανίζεται συνήθως στο πάνω μέρος του παραθύρου του περιηγητή και τοποθετείται πάντα μέσα στην head. Τέλος με την ετικέτα <body>, δηλώνουμε ότι το περιεχόμενό της ανήκει στο κυρίως μέρος της ιστοσελίδας. Ένα ελάχιστο έγγραφο HTML φαίνεται παρακάτω:

```
<HTML>
<HEAD>
<TITLE>Τίτλος</TITLE>
</HEAD>
<BODY>
Hello, world!
```

</BODY>

</HTML>

Στη συνέχεια παρουσιάζουμε τις βασικές ετικέτες μορφοποίησης με τις οποίες μας παρέχεται η δυνατότητα να διαμορφώσουμε ανάλογα με τις επιθυμίες μας κάθε φορά το περιεχόμενο της ιστοσελίδας μας. Αν δεν αναφέρεται ρητά, εννοείται ότι υπάρχουν και οι αντίστοιχες εντολές κλεισίματος.

: Έντονη γραφή (bold)

<u>: Υπογράμμιση (underlined)

<i>: Πλάγια γραφή (italics)

<center>: Κεντρική στοίχιση

<left>: Αριστερή στοίχιση

<right>: Δεξιά στοίχιση

: Δίνει έμφαση(highlighted), ίδιο αποτέλεσμα με πλάγια γραφή

<p>: Παράγραφο

: Νέα γραμμή (συνέχιση γραφής στην επόμενη). Δεν κλείνει.

<hr>: Χάραξη μιας οριζόντιας γραμμής στην σελίδα μας. Δεν κλείνει

<h1> - <h6>: επικεφαλίδες, όπου ο αριθμός ορίζει το μέγεθος της επικεφαλίδας.

Το πιο σημαντικό χαρακτηριστικό των ιστοσελίδων που τις διαφοροποιεί από τα απλά κείμενα είναι η ύπαρξη υπερσυνδέσμων. Δηλαδή μέσα από μία ιστοσελίδα μπορεί κάποιος να κάνει ένα πλήθος παραπομπών σε άλλες εσωτερικές ή εξωτερικές ιστοσελίδες. Για την δημιουργία μιας υπερσύνδεσης χρησιμοποιείται ο παρακάτω κώδικας :

```
<a href="www.google.com"> Κείμενο </a>
```

Το κείμενο που περικλείεται από το <a> εμφανίζεται συνήθως μπλε και υπογραμμισμένο και πατώντας πάνω μεταφερόμαστε στο url που ορίζεται στο href. Εδώ παρατηρούμε για πρώτη φορά ότι μια ετικέτα μπορεί να δεχθεί παραμέτρους χρησιμοποιώντας κάποιες ιδιότητες (attributes), στην περίπτωση αυτή το href.

Για να συμπεριλάβουμε μια εικόνα γράφουμε:

```

```

Για την δημιουργία ενός πίνακα θα δώθει ένα παράδειγμα για το πώς μπορεί να οριστούν οι γραμμές και οι στήλες του.

```
<table>
<tr> <td> 1 </td> <td> 2 </td>
</tr>
<tr> <td> 3 </td> <td> 4 </td>
</tr>
</table>
```

Και το αποτέλεσμα είναι:

1	2
3	4

2.1.2 Φόρμες

Μια πολύ σημαντική δυνατότητα που προσφέρει η HTML, την οποία και χρησιμοποιήσαμε εκτεταμένα κατά τη διάρκεια της δημιουργίας της εργασίας αυτής είναι οι φόρμες. Μέσω των φορμών μπορούμε να αντλήσουμε στοιχεία από το χρήστη. Για να δημιουργήσουμε μια τέτοια φόρμα χρειαζόμαστε την οδηγία <form> η λειτουργία της οποίας ρυθμίζεται από τις παραμέτρους method (παίρνει τις τιμές GET και POST) και action. Η παράμετρος action δηλώνει το script στο οποίο θα σταλούν τα δεδομένα της φόρμας για να τα διαχειριστεί. Τα scripts αυτά, π.χ. σε php γλώσσα, επιτρέπουν τη δυναμική παραγωγή σελίδων HTML και θα

αναλυθούν παρακάτω. Η παράμετρος method, καθορίζει με ποιο τρόπο θα λάβει τα δεδομένα το script. Όταν έχει την τιμή GET, τότε τα περιεχόμενα της φόρμας θα προστεθούν στο τέλος του url που θα λάβει το script, π.χ. www.a.com?variable=value. Με την τιμή POST το script θα λάβει τα δεδομένα μέσω κάποιας μεταβλητής η οποία δεν είναι ορατή στο url. Η GET επιτρέπει την αποθήκευση της σελίδας με τα δεδομένα που έχει επιλέξει ο χρήστης, όμως έχει κάποιο περιορισμό στο μέγεθος. Αντίθετα με την POST μπορούμε να αποστείλουμε μεγάλο όγκο δεδομένων και επιπλέον χρησιμοποιείται όταν δεν θέλουμε να φαίνονται τα δεδομένα στο url.

Οι φόρμα μπορεί να περιλαμβάνει πολλούς τύπους πεδίων για είσοδο από το χρήστη. Για αυτό το σκοπό υπάρχει η ετικέτα <input>, η οποία λειτουργεί με τις παραμέτρους type ή/και name. Η name καθορίζει το όνομα που δίνεται στο πεδίο αυτό και καθώς και στην αντίστοιχη μεταβλητή που θα σταλεί στο script. Τα πεδία είναι:

- Text: το προκαθορισμένο (αν δεν δηλώνεται) και πιο συνηθισμένο είδος εισόδου, ένα πεδίο για εισαγωγή κειμένου.
- Checkbox: όπως λέει το όνομα του, ένα «κουτί» που ο χρήστης μπορεί να τσεκάρει.
- Radio: σε αντίθεση με το checkbox, ο χρήστης μπορεί να επιλέξει μόνο ένα radio button από μια ομάδα επιλογών που έχουν το ίδιο name (αλλά διαφορετικό value).
- Hidden: για αποστολή κρυφών τιμών
- Password: εμφανίζει αστερίσκους (*) κρύβοντας το κείμενο που πληκτρολογείτε.
- Submit: κουμπί που υποβάλει τη φόρμα, δηλαδή στέλνει τα δεδομένα στο script.

Άλλες ετικέτες που μπορούν να χρησιμοποιηθούν σε μια φόρμα είναι:

- Select: επιλογή από dropdown μενού. Χρησιμοποιείται σε συνδυασμό με το option
- Textarea: κείμενο πολλών γραμμών.

Παράδειγμα μιας απλής φόρμας για είσοδο χρήστη:

```
<form name="input" action="login.php" method="POST">
Όνομα χρήστη: <input type="text" name="user">
Κωδικός: <input type="password" name="password">
<input type="submit" value="submit">
</form>
```

2.2 CSS



Εικόνα 2.1.2-1 CSS

Το CSS (Cascading Style Sheets) είναι πρότυπο του W3C που ελέγχει τον τρόπο εμφάνισης ενός εγγράφου που έχει γραφτεί σε HTML [3]. Το CSS έχει σκοπό τη μορφοποίηση του περιεχομένου και τη γεωμετρική τοποθέτησή του στη σελίδα. Για πιο ξεκάθαρο κώδικα και για περισσότερες σχεδιαστικές δυνατότητες η χρήση CSS είναι αναγκαία. Σύμφωνα με τις υποδείξεις του οργανισμού W3C (World Wide Web Consortium), ο οποίος δημιουργεί και συντηρεί τα πρότυπα για HTML και CSS, ενθαρρύνεται η χρήση CSS αντί στοιχείων της HTML για τη διαμόρφωση του περιεχομένου.

Οι δυνατότητες που μας δίνει η html για μορφοποίηση τόσο όσον αφορά τη διάταξη της σελίδας όσο και τη μορφοποίηση των χαρακτήρων είναι περιορισμένες. Με μόνο τη χρήση της html θα έπρεπε για την διάταξη της σελίδας να κάνουμε χρήση πολλών ιδιοτήτων καταλήγοντας σε μεγάλο μέγεθος κώδικα, πολύ συχνά επαναλαμβανόμενο σε διαφορετικά έγγραφα html του ίδιου ιστότοπου. Αυτό θα δυσκόλευε την κατανόηση του κώδικα καθώς και τη δυνατότητα πραγματοποίησης μικρών αλλαγών. Γενικότερα ο σκοπός της html δεν είναι να μορφοποιεί το περιεχόμενο αλλά να το επισημαίνει και να αφήνει τη μορφοποίηση του περιεχομένου σε άλλες τεχνολογίες. Το CSS αποτελείται από οδηγίες οι οποίες καθορίζουν στιλιστικά μια ιστοσελίδα όπως χρώματα, στοίχιση και δίνει πολλές περισσότερες δυνατότητες μορφοποίησης και πολύ περισσότερες δυνατές τιμές για αυτές τις επιλογές από ότι οι αντίστοιχες ιδιότητες στοιχείων της html. Επίσης η γλώσσα ενσωματώνει την ιδέα της επικάλυψης (Cascade) αφού υπάρχει η δυνατότητα ιεραρχικής εφαρμογής των κανόνων έτσι ώστε να δίνεται η δυνατότητα για

εξειδικευμένες μορφοποιήσεις σε επιμέρους περιεχόμενο και επίσης η δυνατότητα καθολικής μετατροπής της εμφάνισης με τροποποίηση λίγων σχετικά οδηγιών.

Οι οδηγίες της γλώσσας CSS είναι ανεξάρτητες από το αρχείο το οποίο μορφοποιούν, δηλαδή μπορούν οι ίδιες οδηγίες να εφαρμοστούν σε πολλές σελίδες html. Οι οδηγίες αυτές μπορούν να βρίσκονται σε εξωτερικό αρχείο και να συμπεριλαμβάνονται από το html αρχείο με την ετικέτα

```
<link rel="stylesheet" type="text/css" href="filename.css">
```

Άλλη δυνατότητα είναι αυτές οι οδηγίες να είναι ενσωματωμένες στον κώδικα της html σελίδας με τη χρήση της ετικέτας <style> Οδηγίες CSS</style>. Τέλος μπορούμε να εισάγουμε οδηγίες σε κάθε ετικέτα html ξεχωριστά με χρήση της ιδιότητας style=" οδηγίες CSS" που υποστηρίζεται από τις περισσότερες ετικέτες της html.

Η γλώσσα CSS αποτελείται από μία σειρά κανόνων της μορφής

```
<Επιλογέας> {Δήλωση 1; Δήλωση 2 ... Δήλωση N;}
```

όπου μια Δήλωση αποτελείται από ένα ζεύγος Ιδιότητα: Τιμή

Ο επιλογέας (selector) καθορίζει σε ποια στοιχεία της σελίδας θα εφαρμοστούν οι δηλώσεις. Οι δηλώσεις σε έναν κανόνα ομαδοποιούνται από αγκύλες και μπορούν να περιέχουν παραπάνω από μιας οδηγίες. Ο επιλογέας είναι μια html ετικέτα, όπως για παράδειγμα η <p>, οπότε οι δηλώσεις μορφοποιούν την εμφάνιση όλων των στοιχείων <p> της σελίδας. Επίσης μετά το όνομα της ετικέτας μπορούμε να ορίσουμε μια κλάση (class) ή μια ταυτότητα (id). Το class αναφέρεται σε όλες τις ετικέτες αυτού του τύπου, ενώ το id αντιστοιχεί σε μια συγκεκριμένη ετικέτα.

Η διαμόρφωση της διάταξης της σελίδας μας επιτυγχάνεται με χρήση εμφωλευμένων ετικετών <div> όπου η κάθε μία έχει διαφορετική τιμή για την ιδιότητα id. Η ετικέτα div από μόνη της δεν έχει κάποιο ορατό αποτέλεσμα, απλά χωρίζει το περιεχόμενο σε τμήματα. Ύστερα, με την εφαρμογή κατάλληλων οδηγιών CSS επιτυγχάνεται η σωστή τοποθέτηση των τμημάτων στο χώρο κατά τον επιθυμητό τρόπο.

Παραδείγματα:

Όλες οι παράγραφοι έχουν μπλε κείμενο:

```
p { color: blue;}
```

Οι παράγραφοι της κατηγορίας prasino <p class="prasino"> θα έχουν πράσινο κείμενο:

```
p .prasino { color: green;}
```

Η παράγραφος warning <p id="warning"> θα έχει κόκκινο κείμενο:

```
p #warning {color: red;}
```

2.3 PHP



Εικόνα 2.1.2-1 PHP

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων Web με δυναμικό περιεχόμενο, η οποία δημιουργήθηκε το 1995 [5]. Μια ιστοσελίδα που περιέχει κώδικα PHP υπόκειται σε επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε με αποτέλεσμα τη δημιουργία να παραχθεί κώδικας HTML ο οποίος θα αποστέλλεται αλλά και θα προβάλλεται στο πρόγραμμα περιήγησης των επισκεπτών χρήστη.

Ο PHP κώδικας μπορεί να γραφεί είτε ανεξάρτητα είτε να ενσωματωθεί σε ένα HTML αρχείο. Για να ορίσουμε ότι ο κώδικας είναι σε PHP χρησιμοποιούνται ειδικές ετικέτες, οι οποίες οριοθετούν τις περιοχές κώδικα PHP: η ετικέτα έναρξης `<?php` και τερματισμού `?>`. Όπως προαναφέραμε, η έξοδος της PHP είναι ένα HTML αρχείο. Δηλαδή η βασική εντολή `echo` δεν τυπώνει κάτι στην οθόνη/κονσόλα όπως στις συνηθισμένες γλώσσες προγραμματισμού, αλλά τυπώνει το αποτέλεσμα στο αρχείο που θα σταλεί στον περιηγητή.

Η γλώσσα PHP έχει παρόμοια σύνταξη με τη C/C++, π.χ. ο τερματισμός εντολής γίνεται πάντα με «;», έχει ακριβώς τις ίδιες δομές επιλογής (if, switch) και επανάληψης (while, for). Οπότε δεν θα αναφερθούμε αναλυτικά σε αυτά, αλλά θα επικεντρώσουμε στα ιδιαίτερα χαρακτηριστικά που την ξεχωρίζουν από τις γλώσσες όπως C/C++. Οι μεταβλητές της php έχουν πάντα ως πρόθεμα το σύμβολο `$` [4][5]. Η PHP είναι σχετικά χαλαρή όσον αφορά το θέμα του τύπου δεδομένων, το οποίο σημαίνει ότι εξακριβώνει τους τύπους δεδομένων όταν ανατίθενται δεδομένα σε κάθε μεταβλητή. Αυτό είναι ταυτόχρονα πλεονέκτημα και μειονέκτημα. Από την μία σημαίνει ότι οι μεταβλητές μπορούν να χρησιμοποιούνται με ευέλικτο τρόπο, αποθηκεύοντας ένα αλφαριθμητικό την μία φορά και έναν ακέραιο την άλλη. Από την άλλη, αυτή η προσέγγιση μπορεί να δημιουργήσει προβλήματα σε μεγαλύτερα scripts, εάν αναμένεται ότι μία μεταβλητή αποθηκεύει έναν συγκεκριμένο τύπο δεδομένων.

2.3.1 Superglobal variables

Επιπρόσθετα με τις μεταβλητές που δημιουργούμε εμείς, η PHP διαθέτει αρκετές προκαθορισμένες μεταβλητές οι οποίες αποκαλούνται «υπερ-γενικές» (superglobals). Αυτές οι μεταβλητές είναι πάντα παρούσες και οι τιμές τους είναι διαθέσιμες σε όλα τα script μας. Κάθε μία από τις ακόλουθες superglobal μεταβλητές είναι στη πραγματικότητα μια διάταξη (array) άλλων μεταβλητών:

- `$_GET`: περιέχει τις μεταβλητές που παρέχονται σε ένα script από μια φόρμα μέσω τις μεθόδου GET.
- `$_POST`: περιέχει τις μεταβλητές που παρέχονται σε ένα script από μια φόρμα μέσω τις μεθόδου POST.

- `$_COOKIE`: περιέχει τις μεταβλητές που παρέχονται σε ένα script μέσω ενός cookie.
- `$_SERVER`: περιέχει πληροφορίες όπως οι κεφαλίδες (headers), θέσεις αρχείων (διαδρομές καταλόγων), και θέσεις των scripts.
- `$_SESSION`: περιέχει τις μεταβλητές που είναι δηλωμένες σε μία σύνοδο (session) του περιηγητή με τον server.

2.3.2 Εκτύπωση και τελεστής συνένωσης

Για να ενώσουμε δύο μεταβλητές σε ένα αλφαριθμητικό, μπορούμε να χρησιμοποιούμε τον τελεστή «.».

```
$var1 = "abc";
$var2 = 123;
echo $var1.var2; //τυπώνει abc123.
```

Ενδιαφέρον έχει η συμπεριφορά των απλών και διπλών εισαγωγικών. Χρησιμοποιώντας διπλά εισαγωγικά, αντικαθίστανται τυχόν μεταβλητές με την τιμή τους

```
echo "Var1 is $var1"; //τυπώνει Var1 is abc
echo 'Var1 is $var1'; //τυπώνει Var1 is $var1
```

2.3.3 Πίνακες

Το πιο ενδιαφέρον σημείο είναι ότι η php υποστηρίζει αλφαριθμητικά ως δείκτες πινάκων (associative arrays). Κατά τα άλλα, μπορούμε να έχουμε τη γνωστή αρίθμηση η οποία ξεκινάει από το 0. Π.χ.

```
$a = array();
$a[0] = 1;
$a["b"] = 2;
```

2.4 MySQL



Εικόνα 2.3.3-1 MySQL

Η MySQL είναι ένα ανοιχτού κώδικα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Data Base Management System - RDBMS) και είναι διαθέσιμη από το 1996 [5]. Μια βάση δεδομένων μας επιτρέπει να αποθηκεύουμε, να αναζητούμε, να ταξινομούμε και να διαβάζουμε τα δεδομένα αποτελεσματικά. Αποτελείται από έναν εξυπηρετητή ο οποίος είναι πολυνηματικός και επιτρέπει την ταυτόχρονη πρόσβαση πολλών χρηστών. Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα, και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση. Χρησιμοποιεί την SQL (Structured Query Language) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων.

Οι βασικές εντολές που χρησιμοποιήσαμε εκτεταμένα σε αυτή την εργασία είναι:

```
SELECT * FROM table_name [WHERE conditions]
```

Επιλέγει όλες τις στήλες από τον πίνακα `table_name`. Σχεδόν πάντα συνδυάζεται με ένα σύνολο συνθηκών, για παράδειγμα της μορφής

```
id > 10 AND status = "active"
```

όπου υποθέτουμε ότι ο πίνακας έχει στήλες `id`, `status`, οπότε επιλέγουμε τις γραμμές που συμφωνούν με την παραπάνω λογική συνθήκη. Το `WHERE` μπορεί να έχει αρκετά πολύπλοκη

μορφή για να πετύχουμε αυτό που θέλουμε. Επιπρόσθετα μπορούμε να κάνουμε πολλές ενέργειες, που δεν θα αναλύσουμε, όπως ταξινόμηση ή να εφαρμόσουμε κάποια συνάρτηση (π.χ. άθροισμα) σε όλες τις γραμμές που επιλέγονται.

Επίσης, συχνά χρειάζεται να επιλέξουμε δεδομένα από παραπάνω από έναν πίνακα. Αυτό συνοπτικά γίνεται ως εξής:

```
SELECT * FROM students, supervisors WHERE
```

Ενώνει κάθε γραμμή του πίνακα students με τις γραμμές του πίνακα supervisors, όπου όμως το id του επιβλέποντα είναι το ίδιο με το supervisor_id του φοιτητή.

Εκτός από επιλογή γραμμών, υπάρχουν και οι παρακάτω εντολές:

```
INSERT INTO table name (col1, col2 ...) VALUES (val1, val2...)
```

Πραγματοποιεί εισαγωγή των δεδομένων (val1,val2...) στις αντίστοιχες στήλες.

```
DELETE FROM table_name WHERE condition
```

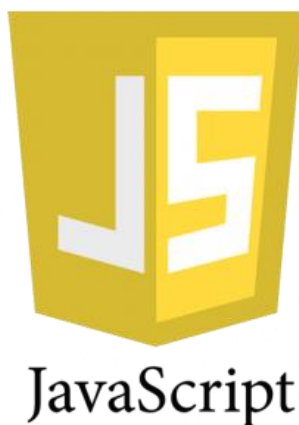
Διαγράφει τις γραμμές που επιλέγονται με βάση τη συνθήκη.

```
UPDATE table_name SET col1=val1, col2=val2... WHERE conditions
```

Τροποποιεί τις στήλες των επιλεγμένων γραμμών.

Τέλος, για να βελτιστοποιηθεί η απόδοση της βάσης δεδομένων και να μειωθεί ο απαραίτητος αποθηκευτικός χώρος, υποστηρίζονται πολλοί τύποι δεδομένων με διαφορετικό μέγεθος ο καθένας, όπως Αριθμητικοί, Κείμενο, Ημερομηνία/Ωρα κλπ.

2.5 Javascript

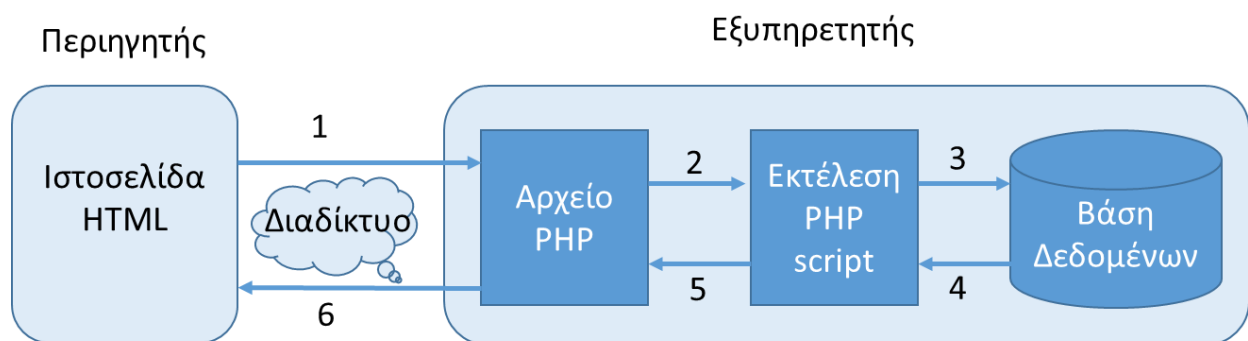


Εικόνα 2.3.3-1 JavaScript

Η Javascript είναι μια διερμηνευμένη (interpreted) γλώσσα προγραμματισμού η οποία εκτελείται από τον περιηγητή (αν και υπάρχουν εφαρμογές με χρήση javascript και στην πλευρά του εξυπηρετητή, π.χ. το Node.js). Χρησιμοποιείται σε δυναμικές ιστοσελίδες, με σκοπό τη δυναμική αλλαγή του προβαλλόμενου περιεχομένου στην ιστοσελίδα, όπως οπτικά εφέ, καθώς και την ασύγχρονη λήψη δεδομένων (Ajax). Η HTML μπορεί να δημιουργήσει μόνο στατικές ιστοσελίδες και για παράδειγμα δεν μπορεί να ανιχνεύσει γεγονότα όπως κλικ. Στην εφαρμογή που αναπτύξαμε υπήρξε περιορισμένη χρήση της javascript, κυρίως για έλεγχο σε φόρμες. Η javascript, αφού εκτελείται στον περιηγητή, μπορεί να ελέγξει την ορθότητα των δεδομένων πριν αυτά να σταλούν στον εξυπηρετητή, μειώνοντας έτσι την ανταλλαγή δεδομένων και τη χρήση πόρων στον εξυπηρετητή. Παρά το όνομά της, δεν έχει σχέση με τη Java, αν και η σύνταξή της είναι C-like. Η Javascript μπορεί να ενσωματωθεί στην html είτε εντός των ετικετών `<script>` είτε συμπεριλαμβάνοντας εξωτερικά αρχεία.

2.6 Συνδυασμός των παραπάνω τεχνολογιών

Με τη βοήθεια της PHP μπορούμε να συνδεθούμε σε έναν οποιοδήποτε MySQL Server στον οποίο έχουμε λογαριασμό, να πάρουμε δεδομένα από ήδη υπάρχουσες βάσεις, να εισάγουμε δεδομένα σε πίνακες βάσεων, να ανανεώσουμε κάποια υπάρχοντα δεδομένα, να φτιάξουμε νέες βάσεις και νέους πίνακες και γενικά να κάνουμε οτιδήποτε γίνεται με μια MySQL βάση δεδομένων.



Εικόνα 2.3.3-1 Τα βήματα που εκτελούνται όταν ο περιηγητής ζητήσει μια σελίδα php

Ο τρόπος που συνεργάζονται όλες αυτές οι τεχνολογίες μαζί φαίνεται στο σχήμα 1.1.

1. Ο περιηγητής του χρήστη στέλνει ένα αίτημα ιστοσελίδας PHP στον Web Server (συνήθως Apache). Αν η σελίδα είναι μόνο HTML, απλά πάμε στο βήμα 6.
2. Εφόσον η σελίδα περιέχει κώδικα PHP στέλνει τον κώδικα στη μηχανή της PHP για περαιτέρω επεξεργασία.
3. Αν ο κώδικας PHP περιέχει ερωτήματα στη βάση δεδομένων, το ερώτημα στέλνεται στον MySQL Server.
4. Το αποτέλεσμα από το ερώτημα στέλνεται πίσω στη μηχανή της PHP για επεξεργασία και ενσωμάτωση στη σελίδα.
5. Ο Web Server συναρμολογεί την σελίδα σε κώδικα HTML.
6. Ο HTML κώδικας στέλνεται στον περιηγητή του πελάτη.

Ας υποθέσουμε ότι ο περιηγητής του χρήστη ζητά από τον εξυπηρετητή τη σελίδα users.php η οποία προβάλλει μια λίστα των χρηστών (Βήμα 1). Ο εξυπηρετητής θα εκτελέσει τον php κώδικα (Βήμα 2) ο οποίος διαβάζει τους χρήστες από τη βάση δεδομένων (Βήμα 3 και 4) και παράγει έναν HTML πίνακα (Βήμα 5) και στέλνει το αποτέλεσμα πίσω στον χρήστη (Βήμα 6).

Πιο αναλυτικά, η επικοινωνία της PHP και της MySQL (Βήμα 3 και 4) περιλαμβάνει:

- 3Α. Σύνδεση στην MySQL χρησιμοποιώντας το hostname, το username και το password.
- 3Β. Επιλογή της βάσης δεδομένων με την οποία θα εργαστούμε.
- 3Γ. Υποβολή SQL ερωτήματος.
4. Ανάγνωση των αποτελεσμάτων

Ουσιαστικά όλες οι λειτουργίες της εφαρμογής μας περιλαμβάνουν είτε προβολή δεδομένων (συνήθως σε πίνακα) είτε αποθήκευση δεδομένων μέσω φόρμας (εισαγωγή νέων ή επεξεργασία υπαρχόντων). Θα δώσουμε παρακάτω τη βασική δομή σε κώδικα php που υλοποιεί αυτές τις λειτουργίες.

2.6.1 Ανάκτηση και προβολή δεδομένων από τη βάση

Σε κώδικα php τα βασικά βήματα είναι:

```
$con = mysqli_connect("localhost", "my_user", "my_password");//Βήμα 3Α
mysqli_select_db($con, "my_db");//Βήμα 3Β
$query = "SELECT username FROM users"
$result = mysqli_query($con, $query); //Βήμα 3Γ
//Βήμα 4. Εδώ παίρνουμε κάθε γραμμή του αποτελέσματος μέσω της
συνάρτησης mysqli_fetch_assoc στη μεταβλητή $row, η οποία είναι ένας
associative πίνακας (δηλαδή χρησιμοποιεί το όνομα της στήλης ως
δείκτη)
echo "<table>";
while ( $row = mysqli_fetch_assoc($result) ){
```

```

Echo "<tr><td>".$row["username"].</td></tr>;
}
Echo "</table>";
mysqli_close($con); //κλείσιμο της σύνδεσης

```

2.6.2 Αποθήκευση δεδομένων από φόρμα

Η προβολή της φόρμας γίνεται απλά μέσω HTML όπως προαναφέραμε. Η php είναι αναγκαία για την επεξεργασία των δεδομένων. Η συνήθης πρακτική είναι να δημιουργείται μια σελίδα php που προβάλλει τη φόρμα και επεξεργάζεται τα δεδομένα (δηλαδή το action στην ετικέτα form είναι το ίδιο αρχείο. Στο παρακάτω παράδειγμα, όταν αρχικά εκτελεστεί ο κώδικας προβάλλεται η φόρμα (else κομμάτι) και όταν πατηθεί το κουμπί «Υποβολή», τα δεδομένα της φόρμας στέλνονται στη ίδια σελίδα, αλλά αυτή τη φορά εκτελείται το κομμάτι κώδικα στο “if”, το οποίο αποθηκεύει την τιμή της μεταβλητής που έδωσε ο χρήστης στη βάση δεδομένων.

Στον παρακάτω κώδικα ελέγχουμε, αν έχει υποβληθεί η φόρμα. Εφόσον πατήθηκε το κουμπί με όνομα submit, θα υπάρχει η αντίστοιχη τιμή στη μεταβλητή \$_POST. Επιπρόσθετα σε αυτόν τον κώδικα εκτελείται και η εντολή για σύνδεση στη βάση και εκτέλεση του ερωτήματος, όπως περιεγράφηκε παραπάνω.

```

<?php
if (isset($_POST["submit"])) {
    $variable1 = $_POST["variable1"];
    $query = "INSERT INTO table1(var1) VALUES ($variable1)";
    echo "Data submitted successfully";
}
else {
?>
<form name="input" action="save.php" method="POST">
Τιμή: <input type="text" name="variable1">

```

```
<input type="submit" name="submit" value="Υποβολή">
</form>
<?php
} //end if
?>
```

2.7 Wampserver



Εικόνα 2.6.2-1 Wampserver

Στην εργασία αυτή το βασικό εργαλείο που χρησιμοποιήθηκε είναι ο Wampserver. Πρόκειται για ένα περιβάλλον ανάπτυξης δικτυακών εφαρμογών, στο οποίο ενσωματώνονται τα τρία πρωτεύοντα εργαλεία που χρησιμοποιούμε για το προγραμματιστικό κομμάτι της εργασίας μας, ο Apache server, η MySQL και η PHP (τα αρχικά του Wamp σημαίνουν Windows-Apache-MySQL-PHP. Αντίστοιχα σε Linux υπάρχει ο Lamp server). Επίσης εμπεριέχει και το PHPMyadmin το οποίο χρησιμοποιείται για την διαχείριση της βάσης δεδομένων μας. Ο Wampserver διατίθεται ελεύθερα από το διαδίκτυο στη διεύθυνση <http://www.wampserver.com>. Ο Wampserver εγκαθίσταται αυτοματοποιημένα και με τις προεπιλεγμένες ρυθμίσεις είναι άμεσα λειτουργικός και διαθέσιμος στο χρήστη. Εκτός των τριών τεχνολογιών που συμπεριλαμβάνει ο WAMP, προφανώς έγινε και εκτεταμένη χρήση της HTML η οποία περιλαμβάνει σε μερικά σημεία σενάρια Javascript.

Αρχικά δημιουργήσαμε στο PHPMyadmin μια βάση δεδομένων με το όνομα «theses», την οποία χρησιμοποιούμε προκειμένου να αποθηκεύσουμε κάθε πληροφορία που χρειαζόμαστε. Εδώ

αφήσαμε τον προεπιλεγμένο λογαριασμό «admin»/«» (χωρίς κωδικό) για πρόσβαση της εφαρμογής στη βάση, αλλά αυτό είναι κάτι που αλλάζει πολύ εύκολα (στο αρχείο db_connect.php). Επίσης δημιουργήσαμε το λογαριασμό του διαχειριστή. Όλες οι υπόλοιπες πληροφορίες μπορούν να παραχθούν μέσω της εφαρμογής.

3 CMS (Content Management System)

Γύρω στα τέλη του 1990 έκανε την εμφάνιση της η πρώτη έκδοση του συστήματος διαχείρισης περιεχόμενου CMS. Ο λόγος που επινοήθηκε και ταυτόχρονα υλοποιήθηκε αυτή η ιδέα ήταν για να μπορεί ένας χρήστης να δημιουργήσει το δικό του ιστοχώρο χωρίς εξειδικευμένες γνώσεις, έτσι ώστε να μπορεί να παρουσιάσει δικτυακά τα προϊόντα τις επιχειρήσεις του. Επειδή στις μέρες μας σχεδόν τα πάντα γίνονται μέσα από το internet και ειδικά η προβολή των προϊόντων μας μέσα από Web sites, που ολοένα και εξελίσσονται, τα CMS έδωσαν την καλύτερη λύση. Έλυσαν τα χέρια των κατασκευαστών Web site μιας και ο χρόνος υλοποιήσεως τους μειώθηκε κατά πολύ.

3.1 Στόχος του CMS

Ο Cms στοχεύει στην διαφοροποίηση κ εύκολη δημιουργία των στοιχείων. Στοιχεία τα οποία χαρακτηρίζουν και ορίζουν έναν ιστοχώρο. Μέσα από το περιβάλλον που διαθέτει το CMS ο χρήστης μπορεί να τροποποιεί το περιεχόμενο της ιστοσελίδας του, τόσο σε κείμενο και εικόνες, όσο σε ήχο και video χωρίς να γνωρίζει προγραμματισμό μιας και είναι όλα αυτοματοποιημένα μέσα από το CMS. Το CMS μπορεί να καλύψει τις ανάγκες του χρήστη μέχρι ένα ικανοποιητικό επίπεδο.

3.2 Δομή του CMS

Το CMS αποτελείται από

- Την Βάση Δεδομένων (Data Base)
- Τον πυρήνα (Core)
- Τα templates
- Τα plug-ins

3.2.1 Βάση Δεδομένων (Data Base)

Την βάση δεδομένων την χρειάζεται για να μπορεί να αποθηκεύει αυτόματα τις παραμετροποίησης και γενικότερα τις ρυθμίσεις που γίνονται στο Web site, καθώς και το menu που δημιουργήσαμε.

3.2.2 Ο πυρήνας (Core)

Ο Core είναι ο πυρήνας του CMS ο οποίος δημιουργεί το interface στον οποίο προβάλλεται το Website καθώς και έχει αναλάβει καθήκοντα υπεύθυνου στην διαχείριση περιεχομένου.

3.2.3 Template

Η λειτουργία του Template είναι να ρυθμίζει την εμφάνιση του περιεχομένου, καθώς και του interface στις προτιμήσεις του χρήστη. Υπάρχουν έτοιμα Template στο περιεχόμενο του CMS για την διευκόλυνση του χρήστη.

3.2.4 Plug-ins

Τέλος τα plug-ins τα οποία μπορούν να προστεθούν στο CMS για συγκεκριμένες χρήσεις.

4 Εφαρμογή Διαχείρισης Πτυχιακών Εργασιών

4.1 Σκοπός

Ο σκοπός της παρούσας πτυχιακής εργασία είναι η ανάπτυξη ενός συστήματος για τη διαχείριση πτυχιακών εργασιών του τμήματος. Οι βασικές λειτουργίες του συστήματος περιλαμβάνουν:

Οι καθηγητές θα πρέπει να μπορούν

- να δημιουργούν, επεξεργάζονται και διαγράφουν θέματα πτυχιακών εργασιών
- να εγκρίνουν την ανάθεση και την ολοκλήρωση των πτυχιακών εργασιών.

Οι φοιτητές θα πρέπει να μπορούν:

- να περιηγούνται στα πιθανά θέματα
- να επιλέγουν την επιθυμητή πτυχιακή εργασία.
- να υποβάλουν το τελικό κείμενο της πτυχιακής εργασίας

4.2 Ρόλος

Από την παραπάνω περιγραφή είναι φανερό ότι θα χρειαστούν οι εξής ρόλοι χρηστών: οι καθηγητές, οι φοιτητές καθώς και οι διαχειριστές. Οι καθηγητές θα μπορούν να δημιουργούν και να επεξεργάζονται τα θέματα των εργασιών. Ακόμα, θα χρειάζεται να εγκρίνουν τις ενέργειες των φοιτητών, όπως την επιλογή μιας εργασίας και την υποβολή της τελικής έκδοσης. Οι φοιτητές από την άλλη μεριά, αρχικά θα πρέπει να μπορούν να βλέπουν τα διαθέσιμα θέματα εργασιών. Κατόπιν θα μπορούν να επιλέξουν και να δηλώσουν ένα από αυτά (κατόπιν εγκρίσεως από τον καθηγητή). Μετά από κάποιο χρονικό διάστημα, όταν ολοκληρώσουν την εργασία θα μπορούν να ανεβάζουν ένα αρχείο με το τελικό κείμενο. Οι ολοκληρωμένες εργασίες θα πρέπει και αυτές να είναι διαθέσιμες για ανάγνωση από τους υπόλοιπους χρήστες του συστήματος. Τέλος, απαιτείται και ένας διαχειριστής για τους λογαριασμούς των χρηστών (θα μπορούσε να είναι τα μέλη της γραμματείας τους τμήματος). Ο ρόλος του διαχειριστή θα είναι να ελέγχει τους χρήστες που εγγράφονται στο σύστημα: δεν πρέπει κάποιος να μπορεί να γραφτεί στο σύστημα αν δεν είναι φοιτητής και προφανώς πρέπει να ελέγχονται και οι λογαριασμοί των καθηγητών μιας και έχουν αυξημένη ευθύνη στη λειτουργία του συστήματος.

4.3 Ορισμός σχήματος βάσης δεδομένων

Το βασικό αντικείμενο στην εφαρμογή μας είναι η πτυχιακή εργασία (πίνακας theses). Αποφασίσαμε ότι τα πεδία που χρειάζονται να περιγράψουν ικανοποιητικά μια πτυχιακή εργασία είναι τα παρακάτω:

- Τίτλος (κείμενο)
- Περιγραφή (Κείμενο)
- Προαπαιτούμενα (Κείμενο)
- Λέξεις-Κλειδιά (Κείμενο)
- Κατάσταση (ακέραιος, θα περιγράψουμε παρακάτω τις δυνατές καταστάσεις)
- Περίληψη (κείμενο – συμπληρώνεται από το φοιτητή)
- Ημερομηνία Έναρξης (ημερομηνία)
- Ημερομηνία Ολοκλήρωσης (ημερομηνία)
- Id – ακέραιος - κλειδί, μοναδικός για κάθε εργασία
- Supervisor_id – ακέραιος, αναφέρεται στον user που είναι ο καθηγητής που δημιούργησε αυτή την εργασία
- Student_id – ακέραιος, αναφέρεται στον user που είναι ο φοιτητής που ανέλαβε αυτή την εργασία.

Ύστερα, έρχονται οι χρήστες (πίνακας users), όπως περιεγράφηκαν παραπάνω. Τα απαιτούμενα πεδία είναι:

- Id - ακέραιος-κλειδί, μοναδικός για κάθε χρήστη
- Όνομα - κείμενο
- Επώνυμο - κείμενο
- Email – κείμενο (θα χρησιμοποιηθεί ως username)
- Κωδικός – κείμενο (σταθερού μήκους 40 Byte, βλ. παρακάτω)
- Τηλέφωνο - κείμενο
- Ιδιότητα - ακέραιος (1 – Φοιτητής, 2 – Επιβλέπων, 3 – Διαχειριστής)

- Ενεργός - ακέραιος (0 – ανενεργός, 1 – ενεργός)
- Αριθμός Μητρώου (AM) – ακέραιος, μόνο για Φοιτητές, κενό για καθηγητές

Τέλος, θα χρειαστούμε έναν ακόμα πίνακα (files) για να κρατάμε τη λίστα των αρχείων του τελικού κειμένου που ανεβάζουν οι φοιτητές. Αυτός ο πίνακας χρειάζεται τα εξής πεδία:

- Id - ακέραιος-κλειδί, μοναδικός για κάθε αρχείο
- thesis_id – ακέραιος, αναφέρεται στην εργασία με την οποία σχετίζεται το αρχείο
- filename - κείμενο

Το σχήμα της βάσης δεδομένων μπορεί να παραχθεί από τον κώδικα MySQL ο οποίος περιέχεται στο παράρτημα.

4.4 Ανάπτυξη εφαρμογής

Η αναλυτική λειτουργία της εφαρμογής που αναπτύξαμε περιγράφεται με εικόνες για την καλύτερη κατανόηση και αναλύονται τα τμήματα κώδικα που επιτελούν κάποια αξιοσημείωτη λειτουργία.

Θα περιγράψουμε αναλυτικά τα Scripts που δημιουργήσαμε σε γλώσσα PHP και HTML για την δημιουργία των απαραίτητων λειτουργιών της εφαρμογής μας.

Περίληπτικά, το user management (διαχείριση χρηστών) επιτυγχάνεται μέσω των αρχείων register.php, index.php και logout.php. Για την καλύτερη οργάνωση του κώδικα δημιουργήθηκαν δύο αρχεία (db_connect.php και check_login.php), τα οποία γίνονται include όπου είναι απαραίτητο. Για το layout της εφαρμογής και την εμφάνιση (χρώματα, μορφοποίηση, κτλ) δημιουργήθηκαν τα αρχεία header.php, footer.php και style.css. Τα αρχεία γίνονται include όπου χρειάζεται. Τα υπόλοιπα αρχεία επιτελούν τις βασικές λειτουργίες της εφαρμογής μας που σχετίζονται με τη διαχείριση των εργασιών (δημιουργία, επιλογή από φοιτητές, υποβολή κλπ).

4.4.1 Αρχεία header.php και footer.php

Αυτές είναι βοηθητικές σελίδες και περιέχουν τον κώδικα για την αρχή και το τέλος της σελίδας. Όλες οι σελίδες που βλέπει ο χρήστης περιλαμβάνουν αυτές τις δύο σελίδες. Επίσης η header συμπεριλαμβάνει το menu, το οποίο τροποποιείται ανάλογα με το είδος χρήστη (π.χ. στο διαχειριστή προβάλλει το μενού του διαχειριστή κλπ).

Στο header.php επίσης δηλώσαμε δύο πίνακες που βοηθούν ώστε να αντιστοιχίζονται οι κωδικοί της κατάστασης της εργασίας και των ρόλων σε κείμενο, για να το προβάλλουμε με φιλικό προς το χρήστη τρόπο.

```
$statuses = [  
    0 => 'Διαθέσιμη',  
    1 => 'Ζητήθηκε',  
    2 => 'Σε εξέλιξη',  
    3 => 'Υποβλήθηκε',  
    4 => 'Ολοκληρωμένη',  
    5 => 'Διακόπηκε',  
    '' => ''];
```

```
$roles = [  
    1 => 'Φοιτητής',  
    2 => 'Επιβλέπων',  
    3 => 'Διαχειριστής'];
```

4.4.2 Αρχείο db_connect.php

Περιέχει τον κώδικα που δημιουργεί μια σύνδεση στη βάση δεδομένων (Login και επιλογή της βάσης theses). Αυτό το αρχείο συμπεριλαμβάνεται σε κάθε σελίδα όπου απαιτείται ανάγνωση ή αποθήκευση δεδομένων. Δημιουργεί τη μεταβλητή \$con που χρησιμοποιούν οι εντολές mysqlí.

4.4.3 Αρχείο Style.css

Περιλαμβάνει τους κανόνες CSS για το στυλ της σελίδας μας. Χρησιμοποιήσαμε στοιχειώδεις κανόνες, ώστε να φαίνεται καλοσχεδιασμένη η σελίδα μας. Αρχικά την εχούμε χωρίσει στις εξής περιοχές με divs:

<div id="container">: περιέχει τα πάντα στη σελίδα.

<div id="header">: περιέχει το επάνω τμήμα, που είναι ίδιο σε κάθε σελίδα

<div id="my_title">: ο τίτλος της σελίδας, ο οποίος με κλικ μας πηγαίνει στην αρχική σελίδα

<div id="menu_container">: περιέχει το μενού

<div id="mainContent">: περιέχει το κυρίως σώμα της κάθε σελίδας.

Επίσης, στο μενού υπάρχει μια λίστα (<ul id="menu">) που περιέχει τον κάθε σύνδεσμο του μενού, για τα οποία επίσης έχουν δημιουργηθεί κανόνες (π.χ. όταν κάποιος περνάει το ποντίκι πάνω από ένα στοιχείο του μενού (hover), αυτό αλλάζει χρώμα). Τέλος, υπάρχει και ένας κανόνας για την εμφάνιση των πινάκων. Όλοι οι κανόνες ουσιαστικά ορίζουν τις αποστάσεις των στοιχείων, ώστε να μην «κολλάνε» μεταξύ τους.

4.4.4 Αρχείο Register.php

Εξετάζοντας τη δομή της εφαρμογής, απαιτείται αρχικά εγγραφή με τα προσωπικά στοιχεία του κάθε χρήστη για τη δημιουργία λογαριασμού. Η σελίδα αυτή δημιουργεί μια φόρμα για να υποβάλει ο χρήστης που επιθυμεί να εγγραφεί τα στοιχεία του (Εικόνα 4.4.4-1):

- Όνομα
- Επώνυμο

- Email (θα χρησιμοποιηθεί ως username)
- Κωδικός
- Ιδιότητα (επιλογή μεταξύ «Φοιτητής» ή «Επιβλέπων»)
- Τηλέφωνο
- AM (μόνο για Φοιτητές)

Κατόπιν γίνεται έλεγχος αν το email δεν υπάρχει ήδη στη βάση δεδομένων, μιας και θα παίζει το ρόλο του μοναδικού ονόματος χρήστη, και κατόπιν εισάγονται τα στοιχεία στη βάση δεδομένων στον πίνακα users με κατάσταση «μη ενεργός» (active = 0). Αξίζει να σημειώσουμε ότι ο κωδικός κρυπτογραφείται με τον αλγόριθμο SHA1 και αποθηκεύεται το κρυπτογράφημα στη βάση δεδομένων και όχι το απλό κείμενο. Συνοπτικά, η συνάρτηση κρυπτογράφησης είναι μια μαθηματική συνάρτηση που έχοντας ως είσοδο μια αυθαίρετου μεγέθους ομάδα δεδομένων δίνει έξοδο ένα καθορισμένου μεγέθους αλφαριθμητικό (string), εδώ για την περίπτωση του SHA1 40 Byte. Η έξοδος δεν μπορεί με κανένα τρόπο να αντιστραφεί και να μας παράγει την αρχική είσοδο.

Πτυχιακές Εργασίες

Είσοδος

Επιτυχής Εγγραφή!

Εγγραφή:

Όνομα:

Επώνυμο:

Email:

Κωδικός:

Ιδιότητα:

Τηλέφωνο:

AM: (για Φοιτητές)

Εικόνα 4.4.4-1 Φόρμα εγγραφής με συμπληρωμένα τα στοιχεία

4.4.5 Αρχείο index.php (login)

Η αρχική μας σελίδα έχει τη λειτουργία της εισόδου (Εικόνα 4.4.5-1). Έχει την κλασική φόρμα με δύο πεδία για email και κωδικό πρόσβασης. Όταν η φόρμα συμπληρωθεί και υποβληθεί, παίρνουμε από τη βάση τα στοιχεία του χρήστη με το email που υποβλήθηκε. Ύστερα εξετάζουμε αν ο (κρυπτογραφημένος) κωδικός είναι ο ίδιος. Αν ναι, εξετάζουμε αν ο χρήστης είναι ενεργοποιημένος. Αν η είσοδος ήταν επιτυχής, αποθηκεύουμε στο \$_SESSION τα στοιχεία του χρήστη και θέτουμε μια λογική μεταβλητή logged ίση με 1, ώστε σε άλλες σελίδες να εξετάζουμε μόνο το \$_SESSION και να γνωρίζουμε ποιος χρήστης ζητά τη σελίδα αυτή. Σε περίπτωση που δεν έχουμε επιτυχή είσοδο, προβάλλεται στο χρήστη το κατάλληλο μήνυμα λάθους («Λάθος email/κωδικός!» ή «Ο λογαριασμός δεν είναι ενεργός! Επικοινωνήστε με το Διαχειριστή για ενεργοποίηση.»).

Πτυχιακές Εργασίες

Είσοδος

Email:

Κωδικός:

Εικόνα 4.4.5-1 Είσοδος Χρήστη

Πτυχιακές Εργασίες

Είσοδος

Εγγραφή

Λάθος email/κωδικός!

Είσοδος

Email:
Κωδικός:

Είσοδος

Εικόνα 4.4.5-2 Φόρμα εισόδου με μήνυμα λάθους μετά από ανεπιτυχή προσπάθεια εισόδου

Πτυχιακές Εργασίες

Είσοδος

Εγγραφή

**Ο λογαριασμός δεν είναι ενεργός!
Επικοινωνήστε με το Διαχειριστή για ενεργοποίηση.**

Είσοδος

Email:
Κωδικός:

Είσοδος

Εικόνα 4.4.5-3 Μήνυμα για μη ενεργό λογαριασμό

4.4.6 Αρχείο logout.php

Όταν ο χρήστης επιλέξει έξοδο από τη σελίδα, εκτελείται η Logout.php η οποία το μόνο που κάνει είναι να καταστρέψει το session (\$_SESSION), οπότε ο χρήστης δεν θα αναγνωρίζεται πια ότι έχει πραγματοποιήσει είσοδο.

4.4.7 Αρχείο check_login.php

Αυτή η σελίδα περιέχει μια συνάρτηση για έλεγχο εάν ο χρήστης έχει το απαιτούμενο επίπεδο πρόσβασης (1 για φοιτητή, 2 για επιβλέποντα, 3 για διαχειριστή), αλλιώς προβάλλει ένα μήνυμα λάθους και τερματίζει την εκτέλεση του php script για την ασφάλεια του συστήματος.

4.4.8 Αρχείο admin.php

Αυτή η σελίδα είναι διαθέσιμη μόνο στο διαχειριστή (χρήστης επιπέδου 3). Από εδώ ο διαχειριστής μπορεί να ενεργοποιεί τους χρήστες που έκαναν αίτηση για λογαριασμό στην εφαρμογή (Εικόνα 4.4.8-1) καθώς και να απενεργοποιεί λογαριασμούς και να τους διαγράφει (Εικόνα 4.4.8-2). Συγκεκριμένα φτιάξαμε μια φόρμα (GET μέθοδος) για να επιλέγει να βλέπει τους ενεργούς ή τους ανενεργούς χρήστες. Κατόπιν προβάλλεται η λίστα των χρηστών, όπου δίπλα σε κάθε χρήστη υπάρχει το κουμπί «ενεργοποίηση» ή «απενεργοποίηση» ή «Διαγραφή». Το κάθε κουμπί είναι μια μικρή φόρμα και όταν πατηθεί υποβάλλεται στην admin.php και εκτελείται η αντίστοιχη ενέργεια.

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Διαχείριση Έξοδος

Διαχείριση

Ενεργός: Ναι Όχι

Όνομα	Επώνυμο	Email	Τηλέφωνο	AM	Ιδιότητα	Ενεργός	Action
Kwstantinos	Dmi	kwstantinos@hotmail.com	6985471235	1859	1	0	<input type="button" value="Ενεργοποίηση"/> <input type="button" value="Διαγραφή"/>

Εικόνα 4.4.8-1 Η σελίδα του διαχειριστή σε προβολή των μη ενεργοποιημένων χρηστών

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Διαχείριση Έξοδος

Διαχείριση

Ενεργός: Ναι Όχι

Όνομα	Επώνυμο	Email	Τηλέφωνο	ΑΜ	Ιδιότητα	Ενεργός	Action
maria	lpl	m@hotmail.com	6986768280	1475	1	1	Απενεργοποίηση
panagiwth	mixalopoulos	panos@hotmail.com	6982227080	1742	1	1	Απενεργοποίηση
dimitris	d	d@hotmail.com	6987541475	1478	1	1	Απενεργοποίηση
katerina	kat	k@hotmail.com	6963	1265	1	1	Απενεργοποίηση
Spyridoula	thom	p@hotmail.com	6986825263	1693	1	1	Απενεργοποίηση
Sofia	Dml	sofia@hotmail.com	6986615281	5728	1	1	Απενεργοποίηση
Nikolaos	Anastasopoulos	nikos@hotmail.com	6987072177	0	2	1	Απενεργοποίηση
Kwstas	Dml	kwstas@hotmail.com	6946318522	1676	1	1	Απενεργοποίηση
Xaris	Dml	xarisgauros@hotmail.com	6986825263	1675	1	1	Απενεργοποίηση
Giannhs	Tzimas	giannis@tz.com	1234	0	2	1	Απενεργοποίηση
a	a	a	a	0	3	1	Απενεργοποίηση

Εικόνα 4.4.8-2 Η σελίδα του διαχειριστή σε προβολή των ενεργοποιημένων χρηστών

4.4.9 Αρχείο supervisors.php

Η λίστα όλων των καθηγητών. Το όνομα κάθε καθηγητή είναι υπερσύνδεσμος που οδηγεί στη λίστα πτυχιακών αυτού του καθηγητή. Ο κώδικας διαβάζει από τον πίνακα users τους χρήστες που είναι καθηγητές και τους προβάλλει σε ένα πίνακα.

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Διαχείριση Έξοδος

Λίστα Επιβλεπόντων

Επιβλέπων	Email	Τηλέφωνο
Giannhs Tzimas	giannis@tz.com	1234
Nikolaos Anastasopoulos	nikos@hotmail.com	6987072177

Εικόνα 4.4.9-1 Λίστα επιβλεπόντων

4.4.10 Αρχείο students.php

Ομοίως με τη λίστα των επιβλεπόντων, υλοποιείται η λίστα όλων των φοιτητών. Αξιοσημείωτο είναι ότι επειδή θέλουμε να προβάσουμε και τον τίτλο της πτυχιακής εργασίας του κάθε φοιτητή εφόσον έχει επιλέξει μια, χρησιμοποιούμε LEFT JOIN του πίνακα των χρηστών που είναι φοιτητές με τις πτυχιακές. Στην περίπτωση που χρησιμοποιούσαμε απλό JOIN (SELECT * FROM users, theses ...) θα είχε ως αποτέλεσμα να παίρναμε μόνο τους φοιτητές που έχουν πτυχιακή. Επίσης φαίνεται και σε τι κατάσταση είναι η πτυχιακή του. Στην περίπτωση που δεν έχει επιλέξει ακόμα μια πτυχιακή, όπως βλέπουμε και παρακάτω, εμφανίζεται το πεδίο του τίτλου κενό.

Πτυχιακές Εργασίες					
Λίστα	Επιβλέποντες	Φοιτητές	Διαχείριση	Έξοδος	
Λίστα Φοιτητών					
ΑΜ	Όνοματεπώνυμο	Email	Τηλέφωνο	Τίτλος	Κατάσταση
1478	dimitris d	d@hotmail.com	6987541475	Δημιουργία Web-Site Διπλωματικών	Ολοκληρωμένη
5728	Sofia Dml	sofia@hotmail.com	6986615281	Ασφάλεια σε Ασύρματα Δίκτυα 802.11	Ολοκληρωμένη
1742	panagiwths mixalopoulos	panos@hotmail.com	6982227080	Ποιότητα Υπηρεσίας (Quality of Service-QOS) σε IP δίκτυα	Ζητήθηκε
1693	Spyridoula thom	p@hotmail.com	6986825263	Εκπαιδευτικές προσομοιώσεις λειτουργικών συστημάτων	Ολοκληρωμένη
1675	Xaris Dml	xarisgauros@hotmail.com	6986825263		
1676	Kwstas Dml	kwstas@hotmail.com	6946318522		
1265	katerina kat	k@hotmail.com	6963		
1475	maria lpl	m@hotmail.com	6986768280		
1859	Kwstantinos Dml	kwstantinos@hotmail.com	6985471235		

Εικόνα 4.4.10-1 Λίστα των φοιτητών

4.4.11 Αρχείο list.php

Τα αρχείο είναι η σελίδα στην οποία κατευθύνεται ο χρήστης όταν εισέρθει και δείχνει τη λίστα όλων των πτυχιακών. Για πιο ξεκάθαρη προβολή, δημιουργήσαμε μια φόρμα με radiobuttons για επιλογή της κατηγορίας πτυχιακών που θέλουμε να δούμε (η οποία υποβάλλεται με GET στον εαυτό της, list.php και εκτελείται μετά η αντίστοιχη επιλογή γραμμών του πίνακα theses). Οι δυνατές επιλογές είναι «Όλες», «Διαθέσιμη», «Ζητήθηκε», «Σε εξέλιξη», «Υποβλήθηκε» και «Ολοκληρωμένη». Ο τίτλος κάθε πτυχιακής είναι υπερσύνδεσμος για προβολή επιπλέον λεπτομερειών (βλ. αρχείο view.php).

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Διαχείριση Έξοδος

Λίστα Πτυχιακών Εργασιών

Κατάσταση: Όλες Διαθέσιμη Ζητήθηκε Σε εξέλιξη Υποβλήθηκε Ολοκληρωμένη

Τίτλος	Επιβλέπων	Κατάσταση
Δημιουργία Web-Site Διπλωματικών	Giannhs Tzimas	Ολοκληρωμένη
Ασφάλεια σε Ασύρματα Δίκτυα 802.11	Giannhs Tzimas	Ολοκληρωμένη
Ποιότητα Υπηρεσίας (Quality of Service-QoS) σε IP δίκτυα	Nikolaos Anastasopoulos	Ζητήθηκε
Σχεδίαση Ασύρματων Δικτύων Ειδικού Σκοπού & Ασύρματων Δικτύων Αισθητήρων	Nikolaos Anastasopoulos	Διαθέσιμη
Υποστήριξη Ποιότητας Υπηρεσίας σε ομότιμα δίκτυα	Nikolaos Anastasopoulos	Διαθέσιμη
Αλγόριθμοι κατανομής εργασιών σε περιβάλλοντα καταμετρημένη υπολογιστική	Nikolaos Anastasopoulos	Διαθέσιμη
Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)	Giannhs Tzimas	Διαθέσιμη
Ενεργοποίηση οικιακής συσκευής με αποστολή SMS μέσω κινητού τηλεφώνου και χρήση μικροελεγκτή 8051	Giannhs Tzimas	Διαθέσιμη
Αυτόματος έλεγχος οικιακού φωτισμού με τη χρήση μικροελεγκτή της οικογένειας 8051.	Nikolaos Anastasopoulos	Διαθέσιμη
Εκπαιδευτικές προσομοιώσεις λειτουργικών συστημάτων	Nikolaos Anastasopoulos	Ολοκληρωμένη
Σχεδιασμός-ανάπτυξη-δημιουργία λογισμικού τήρησης λογαριασμών ισολογισμού και αποτελεσμάτων χρήσεως με ενσωματωμένα Java&PHP	Giannhs Tzimas	Διαθέσιμη
Σχεδιασμός-ανάπτυξη-δημιουργία μιας εφαρμογής τηλεδιάσκεψης μεταξύ συνεργαζόμενων εμπορικών επιχειρήσεων Ελλάδος και εξωτερικού	Giannhs Tzimas	Διαθέσιμη

PDF

Εικόνα 4.4.11-1 Λίστα Πτυχιακών

4.4.12 Αρχείο by_sv.php

Αυτή η σελίδα προβάλλει τη λίστα πτυχιακών ενός συγκεκριμένου καθηγητή για να διευκολύνεται η αναζήτηση. Σε αυτή τη σελίδα καταλήγουμε όταν σε κάποιο σημείο της εφαρμογής πατήσουμε πάνω στο σύνδεσμο στο όνομα του καθηγητή.

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Διαχείριση Έξοδος

Πτυχιακές εργασίες ανά Επιβλέποντα:

Επιβλέπων	Email	Τηλέφωνο
Giannhs Tzimas	giannis@tz.com	1234

Τίτλος	Κατάσταση
Δημιουργία Web-Site Διπλωματικών	Ολοκληρωμένη
Ασφάλεια σε Ασύρματα Δίκτυα 802.11	Ολοκληρωμένη
Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)	Διαθέσιμη
Ενεργοποίηση οικιακής συσκευής με αποστολή SMS μέσω κινητού τηλεφώνου και χρήση μικροελεγκτή 8051	Διαθέσιμη
Σχεδιασμός-ανάπτυξη-δημιουργία λογισμικού τήρησης λογαριασμών ισολογισμού και αποτελεσμάτων χρήσεως με ενσωματωμένα Java&PHP	Διαθέσιμη
Σχεδιασμός-ανάπτυξη-δημιουργία μιας εφαρμογής τηλεδιάσκεψης μεταξύ συνεργαζόμενων εμπορικών επιχειρήσεων Ελλάδος και εξωτερικού	Διαθέσιμη

PDF

Εικόνα 4.4.12-1 Πτυχιακές Συγκεκριμένου καθηγητή

4.4.13 Αρχείο view.php

Εδώ προβάλλονται όλα τα στοιχεία μιας πτυχιακής. Συγκεκριμένα προβάλλονται οι πληροφορίες του καθηγητή, του φοιτητή (αν την έχει δηλώσει) καθώς και όλα τα στοιχεία, τα οποία είναι «Τίτλος», «Περιγραφή», «Λέξεις-Κλειδιά», «Προαπαιτούμενα», «Κατάσταση», «Έναρξη», «Τέλος», «Κείμενο». Στο συγκεκριμένο κομμάτι της πτυχιακής, δίνεται η δυνατότητα στους φοιτητές, καθώς και στους καθηγητές να μπορούν να κατεβάσουν το αρχείο (word, pdf) το οποίο έχει επισυνάψει ο ίδιος ο φοιτητής και είναι διαθέσιμο προς κατέβασμα κλικάροντας τον υπερσύνδεσμο «Λήψη Αρχείου».

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Διαχείριση Έξοδος

Δημιουργία Web-Site Διπλωματικών

Επιβλέπων	Email	Τηλέφωνο
Giannhs Tzimas	giannis@tz.com	1234

AM	Φοιτητής	Email	Τηλέφωνο
1478	dimitris d	d.@hotmail.com	6987541475

Τίτλος	Δημιουργία Web-Site Διπλωματικών
Περιγραφή	Να δημιουργηθεί ένας ιστοχώρος όπου καθηγητές θα μπορούν να δημιουργούν διπλωματικές και οι φοιτητές να μπορούν να τις ζητάνε η να δηλώνουν ενδιαφέρον. Επίσης, αφού τελειώσει ο φοιτητής την διπλωματική του να μπορεί να την υποβάλει στον καθηγητή με το αν
Στόχοι	
Λέξεις-Κλειδιά	php , java , Html, css
Προαπαιτούμενα	Περασμένα μαθήματα : Παγκόσμιου Ιστού(Html,css), Java, Βάσεις Δεδομένων.
Ακαδημαϊκό Έτος	
Κατάσταση	Ολοκληρωμένη
Κατηγορία	
Έναρξη	2016-05-09
Τέλος	2016-05-09
Κείμενο	Λήψη Αρχείου

PDF

Εικόνα 4.4.13-1 Προβολή όλων των λεπτομερειών μιας πτυχιακής εργασίας

4.4.14 Αρχείο sv_theses.php

Αυτό είναι το μενού του καθηγητή «οι πτυχιακές μου». Από εδώ μπορεί να διαχειριστεί τις πτυχιακές. Πιο αναλυτικά, υπάρχει μια λίστα των πτυχιακές εργασιών καθώς και κουμπί για δημιουργία νέας (create.php). Οι δυνατές καταστάσεις της πτυχιακής είναι:

- «Διαθέσιμη»: Όταν δημιουργείται μια νέα πτυχιακή, είναι διαθέσιμη προς επιλογή από κάποιον φοιτητή.
- «Ζητήθηκε»: Όταν τη ζητήσει κάποιος φοιτητής, τότε αλλάζει η κατάσταση από «Διαθέσιμη» σε «Ζητήθηκε». Τότε ο καθηγητής πρέπει να εγκρίνει (confirm.php) την πτυχιακή.
- «Σε εξέλιξη»: Όταν εγκριθεί η ανάθεση της πτυχιακής σε έναν φοιτητή, η κατάσταση αλλάζει σε «Σε εξέλιξη».
- «Υποβλήθηκε»: Όταν ο φοιτητής αποφασίσει να την υποβάλει, τότε αλλάζει η κατάσταση σε «Υποβλήθηκε». Ο καθηγητής πρέπει να εγκρίνει τη πτυχιακή (approve.php).
- «Ολοκληρωμένη»: Η τελική κατάσταση όταν ο καθηγητής εγκρίνει την τελική έκδοση της πτυχιακής.

Πτυχιακές Εργασίες

Οι Πτυχιακές Εργασίες μου

Δημιουργία

Τίτλος	Κατάσταση	Ενέργειες
Ποιότητα Υπηρεσίας (Quality of Service-QoS) σε IP δίκτυα	Ζητήθηκε	
Σχεδίαση Ασυρμάτων Δικτύων Ειδικού Σκοπού & Ασυρμάτων Δικτύων Αισθητήρων	Διαθέσιμη	Επεξεργασία Ανάθεση
Υποστήριξη Ποιότητας Υπηρεσίας σε ομότιμα δίκτυα	Διαθέσιμη	Επεξεργασία Ανάθεση
Αλγόριθμοι κατανομής εργασιών σε περιβάλλοντα κατανεμημένης υπολογιστικής	Διαθέσιμη	Επεξεργασία Ανάθεση
Αυτόματος έλεγχος οικιακού φωτισμού με τη χρήση μικροελεγκτή της οικογένειας 8051.	Διαθέσιμη	Επεξεργασία Ανάθεση
Εκπαιδευτικές προσομοιώσεις λειτουργικών συστημάτων	Ολοκληρωμένη	

Εικόνα 4.4.14-1 Το μενού του επιβλέποντα <<Οι πτυχιακές μου>>

4.4.15 Αρχείο create.php

Εδώ υπάρχει μια φόρμα για τη δημιουργία νέας πτυχιακής. Επίσης η ίδια σελίδα χρησιμοποιείται για επεξεργασία μιας υπάρχουσας πτυχιακής (αν παρέχεται το id της μέσω της μεταβλητής \$_GET), οπότε φορτώνει τα στοιχεία της από τη βάση δεδομένων. Επιπλέον υπάρχει ένα κουμπί διαγραφή το οποίο επιτρέπει στον καθηγητή να διαγράψει τη πτυχιακή. Η σελίδα αυτή είναι προσβάσιμη μόνο από χρήστες επιπέδου 2. Τα πεδία που πρέπει να συμπληρώσει ο επιβλέπων είναι τα «Τίτλος», «Περιγραφή», «Λέξεις-Κλειδιά» και «Προαπαιτούμενα». Στην Εικόνα 4.4.15-

1 η φόρμα έχει ανοίξει για επεξεργασία όπως φαίνεται από την ύπαρξη της επιπρόσθετης λειτουργίας «Διαγραφή».

Πτυχιακές Εργασίες

Λίστα | Επιβλέποντες | Φοιτητές | Οι Πτυχιακές μου | Έξοδος

Στοιχεία Πτυχιακής Εργασίας

Τίτλος: Web Site σε HTML

Περιγραφή:

Στόχοι:

Λέξεις-Κλειδιά:

Απαιτήσεις:

Ακαδημαϊκό Έτος:

Κατηγορία:

- Σχεδιασμός και ανάπτυξη συστήματος
- Συγκριτική επισκόπηση ή μελέτη και πλαίσιο αξιολόγησης
- Ανάλυση και σχεδιασμός μοντέλων
- Θεωρητική μελέτη, ανάπτυξη ή ανάλυση αλγορίθμων

Αποθήκευση | Ακύρω

PDF

Εικόνα 4.4.15-1 Η φόρμα για δημιουργία ή επεξεργασία πτυχιακής από τον επιβλέποντα

4.4.16 Αρχείο confirm.php

Εδώ ενημερώνεται ο καθηγητής ότι κάποιος φοιτητής ζήτησε τη πτυχιακή (Εικόνα 4.4.16-1) και εγκρίνει την ανάθεση. Πατώντας το κουμπί «Έγκριση» ο καθηγητής έχει την δυνατότητα να ενημερωθεί για το ποιος φοιτητής την ζήτησε και έπειτα να προχωρήσει στην τελική του έγκριση. Αξιοσημείωτο είναι πως παρόλο που μεταβούμε στο τελικό βήμα έγκρισης ο καθηγητής ακόμα έχει την δυνατότητα επεξεργασίας του περιεχομένου. Ο καθηγητής μπορεί να δει ποιοι έχουν ζητήσει την συγκεκριμένη πτυχιακή και να επιλέξει μέσα από το ειδικό πλαίσιο "Επιλογή Φοιτητή" σε ποιον θα την αναθέσει.

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Οι Πτυχιακές μου Έξοδος

Οι Πτυχιακές Εργασίες μου

Δημιουργία

Τίτλος	Κατάσταση	Ενέργειες
Ποιότητα Υπηρεσίας (Quality of Service-QoS) σε IP δίκτυα	Ζητήθηκε	
Σχεδίαση Ασυρμάτων Δικτύων Ειδικού Σκοπού & Ασυρμάτων Δικτύων Αισθητήρων	Διαθέσιμη	Επεξεργασία Ανάθεση
Υποστήριξη Ποιότητας Υπηρεσίας σε ομότιμα δίκτυα	Διαθέσιμη	Επεξεργασία Ανάθεση
Αλγόριθμοι κατανομής εργασιών σε περιβάλλοντα κατακευματισμένης υπολογιστικής	Διαθέσιμη	Επεξεργασία Ανάθεση
Αυτόματος έλεγχος οικιακού φωτισμού με τη χρήση μικροελεγκτή της οικογένειας 8051.	Διαθέσιμη	Επεξεργασία Ανάθεση
Εκπαιδευτικές προσομοιώσεις λειτουργικών συστημάτων	Ολοκληρωμένη	

PDF

Εικόνα 4.4.16-1 Αναμονή Έγκρισης(από καθηγητή)

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Οι Πτυχιακές μου Έξοδος

Ανάθεση Πτυχιακής Εργασίας

Ζητήθηκε για δήλωση η παρακάτω πτυχιακή εργασία:

Τίτλος: Σχεδίαση Ασυρμάτων Δικτύων Ειδικού Σκοπού & Ασυρμάτων Δικτύων Αισθητήρων

ΑΜ	Όνοματεπώνυμο	Email	Τηλέφωνο
1265	katerina kat	k@hotmail.com	6963
1475	maria lpl	m@hotmail.com	6986768280

Επιλογή φοιτητή:

Εικόνα 4.4.16-2 Τελική Έγκριση

4.4.17 Αρχείο approve.php

Ομοίως με παραπάνω, εδώ ενημερώνεται ο καθηγητής για το την υποβολή της πτυχιακής από τον φοιτητή και εγκρίνει την ολοκλήρωσή της, οπότε αλλάζει η κατάσταση της πτυχιακής από «Υποβλήθηκε» σε «Ολοκληρωμένη» (Εικόνα 4.4.17-1).

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Οι Πτυχιακές μου Έξοδος

Ολοκλήρωση Πτυχιακής Εργασίας

Υποβλήθηκε η παρακάτω πτυχιακή εργασία:

AM	Όνοματεπώνυμο	Email	Τηλέφωνο
1475	maria lpl	m@hotmail.com	6986768280
Τίτλος	Σχεδίαση Ασυρμάτων Δικτύων Ειδικού Σκοπού & Ασυρμάτων Δικτύων Αισθητήρων		
Έναρξη	2016-05-16		
Τέλος	2016-05-16		
Περίληψη	hahahahaha		
Κείμενο	Λήψη Αρχείου		
Έγκριση			

Εικόνα 4.4.17-1 Σελίδα προβολής και έγκρισης υποβληθείσας πτυχιακής εργασίας

4.4.18 Αρχείο st_thesis.php

Αν ο χρήστης είναι φοιτητής, τότε έχει το μενού «η πτυχιακή μου». Εδώ είτε μπορεί να επιλέξει μια πτυχιακή (select1.php) είτε φαίνεται αυτή που έχει επιλέξει και μπορεί να προβάλει τις λεπτομέρειες (view.php) καθώς και να υποβάλει την τελική έκδοση (submit.php) (Εικόνα 4.4.18-1).

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Η Πτυχιακή μου Έξοδος

Η Πτυχιακή μου

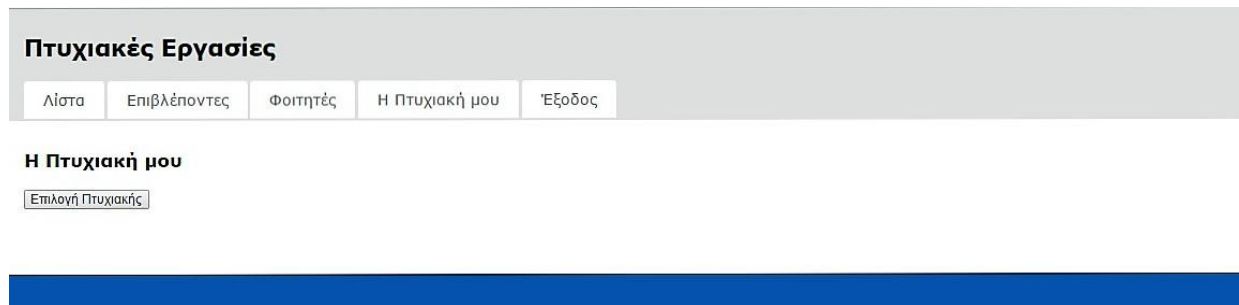
Τίτλος: Αλγόριθμοι κατανομής εργασιών σε περιβάλλοντα καταναμημένης υπολογιστικής
Κατάσταση: Σε εξέλιξη

Προβολή
Υποβολή

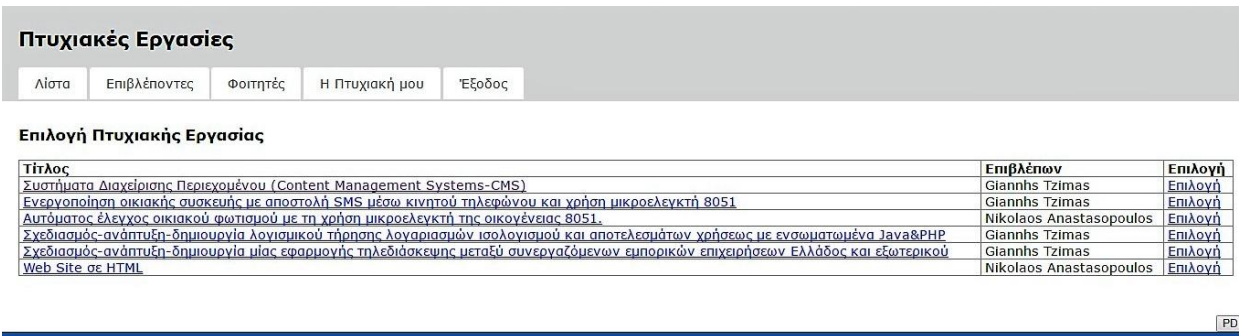
Εικόνα 4.4.18-1 Η σελίδα των φοιτητών <<Η Πτυχιακή μου>>

4.4.19 Αρχείο select1.php

Η σελίδα αυτή υλοποιεί το πρώτο βήμα για την επιλογή πτυχιακής. Ουσιαστικά είναι μια παραλλαγή της list.php, όπου προβάλλει μόνο τις διαθέσιμες πτυχιακές και δίπλα τους ένα κουμπί για επιλογή, το οποίο οδηγεί στο δεύτερο βήμα. Το δεύτερο βήμα είναι η επιλογή πτυχιακής. Ο φοιτητής μπορεί να επιλέξει από μια έως και πέντε πτυχιακές. Αξιοσημείωτο είναι ότι μπορεί να επιλέξει πτυχιακές από διαφορετικό καθηγητή. Τις ίδιες πτυχιακές μπορεί να τις επιλέξει και διαφορετικός φοιτητής, παρά το γεγονός το ότι έχουν επιλεγθεί και από άλλον φοιτητή. Η έγκριση είναι στην κρίση του καθηγητή.



Εικόνα 4.4.19-1 Επιλογή Πτυχιακής - 1ο Βήμα



Εικόνα 4.4.19-2 Επιλογή έως και 5 πτυχιακές

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Η Πτυχιακή μου Έξοδος

Η Πτυχιακή μου

Πτυχιακές εργασίες που ζητήθηκαν:

1. Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)
2. Αυτόματος έλεγχος οικιακού φωτισμού με τη χρήση μικροελεγκτή της οικογένειας 8051.
3. Ενεργοποίηση οικιακής συσκευής με αποστολή SMS μέσω κινητού τηλεφώνου και χρήση μικροελεγκτή 8051
4. Web Site σε HTML
5. Σχεδιασμός-ανάπτυξη-δημιουργία λογισμικού τήρησης λογαριασμών ισολογισμού και αποτελεσμάτων χρήσεως με ενσωματωμένα Java&PHP

PDF

Εικόνα 4.4.19-3 Επιλέχθηκαν 5 πτυχιακές

4.4.20 Αρχείο select2.php

Εδώ προβάλλονται για επιβεβαίωση όλες οι λεπτομέρειες της πτυχιακής που επιλέχθηκε στο προηγούμενο βήμα για να μπορούν να αποφύγουμε τυχών λάθη οι φοιτητές. Αν ο φοιτητής είναι σίγουρος, δηλώνει τη πτυχιακή, η οποία μεταβαίνει στην κατάσταση «Ζητήθηκε» και αναμένει έγκριση από τον καθηγητή.

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Η Πτυχιακή μου Έξοδος

Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)

Επιβλέπων	Email	Τηλέφωνο
Giannhs Tzimas	giannis@tz.com	1234

Τίτλος	Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)
Περιγραφή	Ένα CMS σύστημα χρησιμοποιείται προκειμένου να οργανώνει να διαχειρίζεται και να διανέμει πληροφορία. Ουσιαστικά ένα CMS σύστημα αποτελείται από μία βάση δεδομένων και μία εφαρμογή που έχει την δυνατότητα να οργανώνει και να δίνει πρόσβαση σε όλους τους
Στόχοι	
Λέξεις-Κλειδιά	CMS, Βάσεις Δεδομένων, ψηφιακού περιεχομένου
Προαπαιτούμενα	Περασμένα μαθήματα : Βάσεις Δεδομένων, Ψηφιακή Τηλεώραση
Ακαδημαϊκό Έτος	
Κατηγορία	

Αν επιλέξετε αυτή την Πτυχιακή Εργασία, ο επιβλέπων θα ενημερωθεί για να την εγκρίνει.

Επιλογή

PDF

Εικόνα 4.4.20-1 Επιλογή Πτυχιακής - 2ο Βήμα

4.4.21 Αρχείο submit.php

Όταν φτάσει η στιγμή που ο φοιτητής έχει ολοκληρώσει τη πτυχιακή του, επιλέγει υποβολή από την sv_thesis.php και οδηγείται σε αυτή τη σελίδα όπου συμπληρώνει την περίληψη και ανεβάζει το αρχείο με το τελικό κείμενο (Εικόνα 2.4.21-1). Εδώ ενημερώνεται αυτόματα το πεδίο «Τέλος». Η κατάσταση της εργασίας μεταβαίνει από «Σε εξέλιξη» σε «Υποβλήθηκε» και αναμένει την τελική έγκριση από τον καθηγητή. Στην περίπτωση που δεν γράψει ο φοιτητής το σχετικό κείμενο στο πεδίο περίληψης, εμφανίζεται μήνυμα λάθους χρησιμοποιώντας κώδικα Javascript (Εικόνα 2.4.21-2). Επιπρόσθετα, υπάρχει ειδοποίηση στην οθόνη του φοιτητή στην περίπτωση που το αρχείο δεν ανέβει επιτυχώς (Εικόνα 2.4.21-3), καθώς και μήνυμα επιτυχούς upload του αρχείου στην οθόνη του φοιτητή (Εικόνα 2.4.21-4).

The screenshot shows a web interface for thesis submission. At the top, there is a navigation bar with the title "Πτυχιακές Εργασίες" and several menu items: "Λίστα", "Επιβλέποντες", "Φοιτητές", "Η Πτυχιακή μου", and "Έξοδος". Below this, the main section is titled "Υποβολή Διπλωματικής". It contains three main input areas: "Τίτλος" with a text box containing "Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)", "Περίληψη" with a large text area, and "Κείμενο" with a message "Επιλογή αρχείου Δεν επιλέχθηκε κανένα αρχείο." and two buttons: "Υποβολή" and "Άκυρο". At the bottom left, there is a "PDF" button. The entire form is set against a light gray background with a dark blue footer bar.

Εικόνα 4.4.21-1 Φόρμα υποβολής πτυχιακής

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Η Πτυχιακή μου Έξοδος

Υποβολή Διπλωματικής

Τίτλος: Συστήματα Διαχείρισης Περιεχομένου (Content Management Systems-CMS)

Περίληψη:

Κείμενο: Δεν επιλέχθηκε κανένα αρχείο.

PDF

Ο ιστότοπος 127.0.0.1 λείπει:
Abstract must not be empty!

Εικόνα 4.4.21-2 Μήνυμα σφάλματος λόγο κενής περιλήψης

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Η Πτυχιακή μου Έξοδος

Πρόεκυψε σφάλμα στη λήψη του αρχείου.

PDF

Εικόνα 4.4.21-3 Μη επιτυχές ανέβασμα αρχείου

Πτυχιακές Εργασίες

Λίστα Επιβλέποντες Φοιτητές Η Πτυχιακή μου Έξοδος

Πρόεκυψε σφάλμα στη λήψη του αρχείου.

PDF

Εικόνα 4.4.21-4 Επιτυχές ανέβασμα αρχείου

Εξαγωγή σε PDF

Για την εξαγωγή html κειμένου σε pdf αρχείο, βασιστήκαμε στο php πακέτο html2pdf-4.5.1(το οποίο εγκαταστήσαμε με composer για να εγκαταστήσει κάποια προαπαιτούμενα πακέτα). Για να το χρησιμοποιήσουμε αρχικά φτιάξαμε μια φόρμα με ένα κουμπί που εκκινεί τη μετατροπή σε pdf, το οποίο προσθέσαμε στο κάτω μέρος της σελίδας μας στο footer.php. Η φόρμα έχει δύο κρυφά πεδία, pdfcontent και pdfname, η τιμή των οποίων συμπληρώνεται με javascript ανάλογα με τη σελίδα στην οποία βρισκόμαστε. Συγκεκριμένα, δημιουργήσαμε τη συνάρτηση getPDFContent (αρχείο header.php) η οποία "τραβάει" όλη την html που συμπεριλαμβάνεται στην mainContent div (δηλ. το περιεχόμενο της σελίδας μετά το μενού). Επίσης, ως όνομα για το pdf αρχείο ορίζει το όνομα του αρχείου php το οποίο αποθηκεύουμε. Στο footer.php εκτελείται αυτή η συνάρτηση, ώστε να είναι έτοιμες οι τιμές στη φόρμα, εφόσον πατηθεί το κουμπί PDF. Σε περίπτωση που υποβληθεί η φόρμα, στο header.php δημιουργείται το αρχείο με τη βοήθεια του πακέτου html2pdf.

Βιβλιογραφία

- [1] Paul Deitel, Harvey Deitel, Προγραμματισμός Internet & World Wide Web, Μ. Γκιούρδας, 2011.
- [2] Andrew Tanenbaum, Δίκτυα Υπολογιστών, Κλειδάριθμος,σελ 692, 2011 .
- [3] Elizabeth Castro, Εισαγωγή στην HTML για τον παγκόσμιο ιστό, Κλειδάριθμος, 2003.

[4] www.w3schools.com (τελευταία πρόσβαση 07/05/2016).

[5] Larry Ullman, Εισαγωγή στην PHP για τον παγκόσμιο ιστό, Κλειδάριθμος, 2005.

Παράρτημα

Εδώ παρατίθεται όλος ο κώδικας της εφαρμογής μας.

1. Αρχείο register.php

```
<?php
    session_start();

    $msg="";
    if (isset($_POST['register']))
    {
        include("db_connect.php");
        $email = mysqli_real_escape_string($con, $_POST['email']);
        $password = sha1($_POST['password']); // Encrypts the password.
        $name = mysqli_real_escape_string($con, $_POST['name']);
        $surname = mysqli_real_escape_string($con, $_POST['surname']);
        $tel = mysqli_real_escape_string($con, $_POST['tel']);
        $role = mysqli_real_escape_string($con, $_POST['role']);
        $am = mysqli_real_escape_string($con, $_POST['am']);

        $result = mysqli_query($con, "SELECT * FROM users WHERE email =
        '". $email. "'")
            or die (mysqli_error($con));

        $row = mysqli_fetch_assoc($result);
        if (!empty($row)){
            $msg="Το email υπάρχει ήδη!";
        }
        else {
            $sql = "INSERT INTO `users`
            SET `name`=' $name', `surname`=' $surname', `email`=' $email',
            password=' $password',
            `role`=' $role', `tel`=' $tel', `am`=' $am', `active`=0";

            $result = mysqli_query($con, $sql)
                or die (mysqli_error($con));

            $msg = "Επιτυχής Εγγραφή!";
        }

        unset($_POST['register']);
        mysqli_close($con);
    }
}
```



```

include('header.php');
if (isset($msg)) {
    echo '<div class="err">'. $msg. '</div>';
}
?>
<h3>Εγγραφή:</h3>
<hr />
<form action="register.php" method="post">
    <table>
        <tr>
            <td>Όνομα:</td>
            <td><input type="text" name="name" /></td>
        </tr>
        <tr>
            <td>Επώνυμο:</td>
            <td><input type="text" name="surname" /></td>
        </tr>
        <tr>
            <td>Email:</td>
            <td><input type="text" name="email" /></td>
        </tr>
        <tr>
            <td>Κωδικός:</td>
            <td><input type="password" name="password" /></td>
        </tr>
        <tr>
            <td>Ιδιότητα:</td>
            <td>
                <select name="role">
                    <option value="1">Φοιτητής</option>
                    <option value="2">Επιβλέπων</option>
                </select>
            </td>
        </tr>
        <tr>
            <td>Τηλέφωνο:</td>
            <td><input type="text" name="tel" /></td>
        </tr>
        <tr>
            <td>AM:</td>
            <td><input type="text" name="am" /> (για φοιτητές)</td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit" name="register" value="Εγγραφή"
        </td></tr>
    </table>
</form>

<?php
include('footer.php');
?>

```

2. Αρχείο index.php

```
<?php
    session_start();
    if ( isset($_SESSION['logged']) ){
        header("Location: list.php"); // Goes to main page.
    }

    $msg="";
    if (isset($_POST['login']))
    {
        include("db_connect.php");
        $email = mysqli_real_escape_string($con, $_POST['email']);
        $password = sha1($_POST['password']); // Encrypts the password.

        $result = mysqli_query($con, "SELECT * FROM users WHERE email =
'{$email}'")
            or die (mysqli_error($con)); // MySQL query

        $row = mysqli_fetch_assoc($result);

        if($password == $row['password'])
        {
            if($row['active']){
                $_SESSION['logged'] = 1; // Sets the session.
                $_SESSION['user_id'] = $row['id'];
                $_SESSION['role'] = $row['role'];

                header("Location: list.php"); // Goes to main page.
            }else{
                $msg = "<br><strong>Ο λογαριασμός δεν είναι ενεργός!<br>
/>Επικοινωνήστε με το Διαχειριστή για ενεργοποίηση.</strong>";
            }

        } else { // Invalid username/password.
            $msg = "<br><strong>Λάθος email/κωδικός!</strong>";
        }
        unset($_POST['login']);
        mysqli_close($con);
    }

include('header.php');

if (isset($msg)){
    echo '<div class="err">'. $msg. '</div>';
}

?>
<h3>Είσοδος</h3>
<hr />

<form action="index.php" method="post">
    <table>
        <tr>
            <td>Email:</td>
```

```
        <td><input type="text" name="email" /></td>
    </tr>
    <tr>
        <td>Κωδικός:</td>
        <td><input type="password" name="password" /></td>
    </tr>
    <tr>
        <td></td>
        <td><input type="submit" name="login" value="Είσοδος" /></td>
    </tr>
</table>
</form>
<br />
```

```
<?php
include('footer.php');

?>
```

3. Αρχείο logout.php

```
<?php
session_start();

unset($_SESSION['logged']);

session_unset();

session_destroy();

header("Location: index.php"); // Goes back to login.
?>
```

4. Αρχείο check_login.php

```
<?php
    session_start();

    function check_login($level)
    {
        if (isset($_SESSION['logged'])) {
            if (isset($_SESSION['role']) && $_SESSION['role'] < $level) {
                include('header.php');
                echo '<div class="err">You do not have access to view this
page!</div>';
                include('footer.php');
                exit();
            }
            return;
        } else {
            header('Location: index.php');
        }
    }
?>
```

5. Αρχείο supervisors.php

```
<?php include('check_login.php');
    check_login(1);

    include('header.php');

include('db_connect.php');

$status = 0;

//get all supervisors
$sql = "SELECT * FROM `users` WHERE `role`=2";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$data = array();
while($res = mysqli_fetch_assoc($result)){
    $data []= $res;
}

mysqli_close($con);

?>
<h3>Λίστα Επιβλεπόντων</h3>
<table id="my_table"><tbody>
<tr><th>Επιβλέπων</th><th>Email</th><th>Τηλέφωνο</th></tr>
<?php
//show all supervisors
foreach($data as $item){
    echo "<tr><td><a href='by_sv.php?id={$item['id']}'>{$item['name']}
{$item['surname']}</a></td><td>{$item['email']}</td><td>{$item['tel']}</td></
tr>";
}

echo '</tbody></table>';

include('footer.php');
?>
```

6. Αρχείο students.php

```
<?php include('check_login.php');
    check_login(1);

    include('header.php');

include('db_connect.php');

$status = 0;
//LEFT join: to show all students, and those without thesis
$sql = "SELECT users.*, theses.title, theses.status FROM `users` LEFT JOIN
`theses` ON users.id=theses.student_id WHERE users.role=1";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$data = array();
while($res = mysqli_fetch_assoc($result)){
    $data []= $res;
}

mysqli_close($con);
?>
<h3>Λίστα φοιτητών</h3>
<table id="my_table"><tbody>
<tr><th>ΑΜ</th><th>Όνοματεπώνυμο</th><th>Email</th><th>Τηλέφωνο</th><th>Τίτλο
ς</th><th>Κατάσταση</th></tr>
<?php
foreach($data as $item){
    echo "<tr><td>{$item['am']}</td><td>{$item['name']}
{$item['surname']}</td><td>{$item['email']}</td><td>{$item['tel']}</td><td>{$
item['title']}</td><td>{$statuses[$item['status']]</td></tr>";
}

echo '</tbody></table>';

include('footer.php');
?>
```

7. Αρχείο list.php

```
<?php include('check_login.php');
    check_login(1);

    include('header.php');

include('db_connect.php');

//select theses to view by status
$status = -1;
if (isset($_GET['status'])){
    $status = $_GET['status'];
}

$sql = "SELECT theses.*, users.name, users.surname FROM `theses`, `users`
WHERE theses.supervisor_id =users.id";
if ($status !=-1){
    $sql = $sql." AND theses.`status`=$status";
}

$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$theses = array();
while($res = mysqli_fetch_assoc($result)){
    $theses []= $res;
}

mysqli_close($con);
?>
<h3>Λίστα Πτυχιακών Εργασιών</h3>

<form name="select" action="list.php" method="GET" style="font-size:small">
Κατάσταση: <input type="radio" name="status" value="-1" <?php if ($status ==
-1) echo 'checked="checked"';?>>Όλες
<input type="radio" name="status" value="0" <?php if ($status == 0) echo
'checked="checked"';?>>Διαθέσιμη
<input type="radio" name="status" value="1" <?php if ($status == 1) echo
'checked="checked"';?>>Ζητήθηκε
<input type="radio" name="status" value="2" <?php if ($status == 2) echo
'checked="checked"';?>>Σε εξέλιξη
<input type="radio" name="status" value="3" <?php if ($status == 3) echo
'checked="checked"';?>>Υποβλήθηκε
<input type="radio" name="status" value="4" <?php if ($status == 4) echo
'checked="checked"';?>>Ολοκληρωμένη
<input type="submit" name="select" value="Επιλογή">
</form>
<br>

<table id="my_table" width="100%"><tbody>
<tr><th>Τίτλος</th><th>Επιβλέπων</th><th>Κατάσταση</th></tr>
<?php
foreach($theses as $thesis){
```



```
    echo "<tr><td><a
href='view.php?id={$thesis['id']}'>{$thesis['title']}</a></td>
    <td>{$thesis['name']}
{$thesis['surname']}</td><td>{$statuses[$thesis['status']]</td></tr>";
}
echo '</tbody></table>';

include('footer.php');
?>
```

8. Αρχείο view.php

```
<?php
include('check_login.php');
    check_login(1); //at least student

    include('header.php');

include('db_connect.php');

if (isset($_GET['id'])){
    $id = $_GET['id']; //id of thesis
}

$status = 0;

//get data of thesis #id
$sql = "SELECT * FROM `theses` WHERE `id`=$id";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$thesis = mysqli_fetch_assoc($result);

//get supervisor data
$sql = "SELECT * FROM `users` WHERE `id`={$thesis['supervisor_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$sv = mysqli_fetch_assoc($result);

$st = [];
$files = [];
if ($thesis['student_id'] > 0){
    //if it has student, show his/her data
    $sql = "SELECT * FROM `users` WHERE `id`={$thesis['student_id']}";
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));

    $st = mysqli_fetch_assoc($result);

    //if submitted (3) show files
    if ($thesis['status'] > 2) {
        $sql = "SELECT * FROM `files` WHERE `thesis_id`=" . $thesis['id'];
        $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
        while($res = mysqli_fetch_assoc($result)){
            $files[] = $res;
        }
    }
}

mysqli_close($con);

echo "<h3>{$thesis['title']}</h3>";
```

```

//supervisor data
echo '<table id="my_table"><tbody>';
echo '<tr><th>Επιβλέπων</th><th>Email</th><th>Τηλέφωνο</th></tr>';
echo "<tr><td><a href='by_sv.php?id={\$sv['id']}'>{\$sv['name']}</a></td><td>{\$sv['email']}</td><td>{\$sv['tel']}</td></tr>";
echo '</tbody></table><br>';

if (!empty($st)) {
    //student data if exists
    echo '<table id="my_table"><tbody>';
    echo
'<tr><th>ΑΜ</th><th>Όνοματεπώνυμο</th><th>Email</th><th>Τηλέφωνο</th></tr>';
    echo "&<tr><td>{\$st['am']}</td><td>{\$st['name']}</td><td>{\$st['surname']}</td><td>{\$st['email']}</td><td>{\$st['tel']}</td></tr>";
    echo '</tbody></table><br>';
}

echo '<table id="my_table"><tbody>';
echo "<tr><td><b>Τίτλος</b></td><td>{\$thesis['title']}</td></tr>";
echo "<tr><td><b>Περιγραφή</b></td><td>{\$thesis['description']}</td></tr>";
echo "<tr><td><b>Λέξεις-Κλειδιά</b></td><td>{\$thesis['keywords']}</td></tr>";
echo
"<tr><td><b>Προαπαιτούμενα</b></td><td>{\$thesis['requirements']}</td></tr>";
echo
"<tr><td><b>Κατάσταση</b></td><td>{\$statuses[{\$thesis['status']}]</td></tr>";
//show following only if exist
if ($thesis['start'] > 0) {
    echo "<tr><td><b>Έναρξη</b></td><td>{\$thesis['start']}</td></tr>";
}
if ($thesis['end'] > 0) {
    echo "<tr><td><b>Τέλος</b></td><td>{\$thesis['end']}</td></tr>";
}
if ($thesis['abstract'] > 0) {
    echo "<tr><td><b>Περίληψη</b></td><td>{\$thesis['abstract']}</td></tr>";
}
if (!empty($files)) {
    $types = [0 => 'Κείμενο', 1 => 'Συμπληρωματικά Αρχεία'];
    foreach($files as $file) {
        echo "<tr><td><b>{\$types[{\$file['type']}]</b></td><td><a
href='{\$file['filename']}'>Λήψη Αρχείου</a></td></tr>";
    }
}
echo '</tbody></table>';

include('footer.php');
?>

```

9. Αρχείο sv_theses.php

```
<?php include('check_login.php');
check_login(2); //only professors

include('header.php');

include('db_connect.php');

//get user data
$sql = "SELECT * FROM `users` WHERE `id`={$SESSION['user_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$sv = mysqli_fetch_assoc($result);

//get user's theses
$sql = "SELECT * FROM `theses` WHERE `supervisor_id`={$SESSION['user_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$theses = array();
while($res = mysqli_fetch_assoc($result)){
    $theses []= $res;
}

mysqli_close($con);

echo '<h3>Οι Πτυχιακές μου</h3>';
//create buton
echo '<a href ="create.php"><button>Δημιουργία</button></a><br />';
echo '<table id="my_table" width="100%"><tbody>';
echo '<tr><th>Τίτλος</th><th>Κατάσταση</th><th>Ενέργειες</th></tr>';

//show theses list
foreach($theses as $thesis){
    $actions = "<a href='create.php?id={$thesis['id']}'>Επεξεργασία</a>";
    if ($thesis['status']==1){
        $actions .= " <a href='confirm.php?id={$thesis['id']}'>Εγκριση</a>";
    }
    if ($thesis['status']==3){
        $actions .= " <a href='approve.php?id={$thesis['id']}'>Εγκριση</a>";
    }
    echo "<tr><td><a
href='view.php?id={$thesis['id']}'>{$thesis['title']}</a></td>
<td>{$statures[$thesis['status']]</td><td>{$actions}</td></tr>";
}

echo '</tbody></table>';

include('footer.php');
?>
```

10. Αρχείο create.php

```
<?php
include('check_login.php');
check_login(2); // only professors

if (isset($_POST['Save'])) {

    $title = $_POST['title'];
    $description = $_POST['description'];
    $keywords = $_POST['keywords'];
    $requirements = $_POST['requirements'];

    if ($_POST['id']!=''){
        //update
        include('db_connect.php');
        $sql = "UPDATE `theses` SET `title`='$title',
`description`='$description', `keywords`='$keywords',
`requirements`='$requirements'
        WHERE `id`={$_POST['id']}";
        $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
        mysqli_close($con);
        header('Location: sv_theses.php');
    }
    else{
        //new -> insert
        include('db_connect.php');
        $sql = "INSERT INTO `theses` (`supervisor_id`, `title`,
`description`, `keywords`, `requirements`, `status`)
        VALUES ('{$_SESSION['user_id']}', '$title', '$description',
'$keywords', '$requirements', 0)";
        $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
        mysqli_close($con);
        header('Location: sv_theses.php');
    }
}
elseif (isset($_POST['Delete'])) {
    include('db_connect.php');
    $sql = "DELETE FROM `theses` WHERE id=".$_POST['id'];
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    mysqli_close($con);
    header('Location: sv_theses.php');
}
else {

    if (isset($_GET['id'])) {
        $id = $_GET['id'];
        //load existing to edit
        include('db_connect.php');
        $sql = 'SELECT * FROM `theses` WHERE `id`='.$id;
        $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    }
}
```

```

        $thesis = mysqli_fetch_assoc($result);

        $title = $thesis['title'];
        $description = $thesis['description'];
        $keywords = $thesis['keywords'];
        $requirements = $thesis['requirements'];
        mysqli_close($con);
    } else {
        //create new -> empty
        $id = '';
        $title = '';
        $description = '';
        $keywords = '';
        $requirements = '';
    }
    //show form
}

//form
include('header.php');
?>
<script type="text/javascript">
    function validateForm()
    {
        var x=document.forms["form"]["title"].value;
        if (x==null || x=="")
        {
            alert("Title must not be empty!");
            return false;
        }
        else{
            return true;
        }
    }
</script>

<h3>Στοιχεία Πτυχιακής</h3>
<form id="form" name="form" method="post" action="create.php<?php if($id)
echo '?id='.$id;?>" >
    <input type="hidden" name="id" value="<?php echo $id;?>">
    <input type="hidden" name="supervisor_id" value="<?php echo
$_SESSION['user_id'];?>">
    <table border="0" >
        <tr>
            <td><strong>Τίτλος</strong></td>
            <td><input name="title" type="text" value="<?php echo $title; ?>"
size="128"/></td>
        </tr>
        <tr>
            <td><strong>Περιγραφή</strong></td>
            <td><textarea name="description" id="description" cols="64"
rows="8"><?php echo $description; ?></textarea></td>
        </tr>
        <tr>
            <td><strong>Λέξεις-Κλειδιά</strong></td>
            <td><input name="keywords" type="text" value="<?php echo
$keywords; ?>" size="128"/></td>

```

```

        </tr>
        <tr>
            <td><strong>Απαιτήσεις</strong></td>
            <td><input name="requirements" type="text" value="<?php echo
$requirements; ?>" size="128"/></td>
        </tr>
        <tr>
            <td></td>
            <td>
                <input type="submit" name="Save" value="Αποθήκευση"
onclick="return validateForm()" />
                <?php if($id){
                    echo '<input type="submit" name="Delete" value="Διαγραφή"
onclick="return confirm(\'Οριστική διαγραφή πτυχιακής;\')" />';
                }?>
                <input type="reset" name="Cancel" value="Άκυρο" />
            </td>
        </tr>
    </table>

<?php
    include('footer.php');
?>

```

11. Αρχείο confirm.php

```
<?php
include('check_login.php');
check_login(2);

include('header.php');

include('db_connect.php');

if (isset($_POST['Confirm'])){
    //confirm request
    $id = $_POST['id'];
    $sql = 'UPDATE `theses` SET `status`= 2, `start`=NOW() WHERE
`id`=' . $_POST['id'];
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    mysqli_close($con);
    header('location: sv_theses.php');
}

$id = $_GET['id'];
//get thesis data
$sql = "SELECT * FROM `theses` WHERE `id`=$id";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));
$thesis = mysqli_fetch_assoc($result);

//get student data
$sql = "SELECT * FROM `users` WHERE `id`={$thesis['student_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));
$st = mysqli_fetch_assoc($result);

mysqli_close($con);

echo '<h3>Ανάθεση Πτυχιακής</h3>';
echo 'Ζητήθηκε για δήλωση η παρακάτω πτυχιακή: <br /><br /><b>Τίτλος:</b>
' . $thesis['title'] . '<br /><br />';

//show student data
echo '<table id="my_table"><tbody>';
echo
'<tr><th>ΑΜ</th><th>Όνοματεπώνυμο</th><th>Email</th><th>Τηλέφωνο</th></tr>';
echo "<tr><td>{$st['am']}</td><td>{$st['name']}
{$st['surname']}</td><td>{$st['email']}</td><td>{$st['tel']}</td></tr>";
echo '</tbody></table>';

?>

<form id="form" name="form" method="post" action="confirm.php?id=<?php
echo $id;?>" >
    <input type="hidden" name="id" value="<?php echo $id;?>">
    <input type="submit" name="Confirm" value="Εγκρίση">
</form>

<?php
```



```
include('footer.php');?>
```

12. Αρχείο approve.php

```
<?php include('check_login.php');
check_login(2);

include('header.php');

include('db_connect.php');

if (isset($_POST['Approve'])) {
    //approve request
    $id = $_POST['id'];
    $sql = 'UPDATE `theses` SET `status`= 4 WHERE `id`=' . $_POST['id'];
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    mysqli_close($con);
    header('location: sv_theses.php');
}

$id = $_GET['id'];

$sql = "SELECT * FROM `theses` WHERE `id`=$id";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));
$thesis = mysqli_fetch_assoc($result);

$sql = "SELECT * FROM `users` WHERE `id`={$thesis['student_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));
$st = mysqli_fetch_assoc($result);

$files = [];
$sql = "SELECT * FROM `files` WHERE `thesis_id`=" . $id;
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));
while($res = mysqli_fetch_assoc($result)) {
    $files[] = $res;
}

mysqli_close($con);

echo '<h3>Ολοκλήρωση Πτυχιακής</h3>';
echo 'Υποβλήθηκε η παρακάτω πτυχιακή: <br />';

//student data
echo '<table id="my_table"><tbody>';
echo
'<tr><th>AM</th><th>Όνοματεπώνυμο</th><th>Email</th><th>Τηλέφωνο</th></tr>';
echo "<tr><td>{$st['am']}</td><td>{$st['name']}</td><td>{$st['surname']}</td><td>{$st['email']}</td><td>{$st['tel']}</td></tr>";
echo '</tbody></table>';
```

```

echo '<table id="my_table"><tbody>';
echo "<tr><td><b>Τίτλος</b></td><td>{$thesis['title']}</td></tr>";
echo "<tr><td><b>Έναρξη</b></td><td>{$thesis['start']}</td></tr>";
echo "<tr><td><b>Τέλος</b></td><td>{$thesis['end']}</td></tr>";
echo "<tr><td><b>Περίληψη</b></td><td>{$thesis['abstract']}</td></tr>";
if (!empty($files)) {
    foreach($files as $file) {
        echo "<tr><td><b>Κείμενο</b></td><td><a
href='{$file['filename']}'>Λήψη Αρχείου</a></td></tr>";
    }
}
echo '</tbody></table>';

?>

<form id="form" name="form" method="post" action="approve.php?id=?php
echo $id;?>" >
    <input type="hidden" name="id" value="<?php echo $id;?>">
    <input type="submit" name="Approve" value="Έγκριση">
</form>

<?php
include('footer.php');
?>

```

13. Αρχείο st_thesis.php

```
<?php include('check_login.php');
    check_login(1);

    include('header.php');

include('db_connect.php');

$sql = "SELECT * FROM `theses` WHERE `student_id`={$SESSION['user_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));
$thesis = mysqli_fetch_assoc($result);

mysqli_close($con);

echo '<h3>Η Πτυχιακή μου</h3>';

if (!empty($thesis)){
    echo '<b>Τίτλος: </b>'. $thesis['title']. '<br /><b>Κατάσταση:
</b>'. $statuses[$thesis['status']]. '<br /><br />';
    echo "<a
href='view.php?id={$thesis['id']}'><button>Προβολή</button></a>"; //bad
practise, it works
    if ($thesis['status'] == 2){
        echo "<a
href='submit.php?id={$thesis['id']}'><button>Υποβολή</button></a>"; //bad
practise, it works
    }
}else{
    echo '<a href="select1.php"><button>Επιλογή Πτυχιακής</button></a>';
}

include('footer.php');

?>
```

14. Αρχείο select1.php

```
<?php include('check_login.php');
    check_login(1);

    include('header.php');

include('db_connect.php');

$status = 0;
//get all available theses
$sql = "SELECT theses.*, users.name, users.surname FROM `theses`, `users`
WHERE theses.supervisor_id =users.id AND theses.`status`=$status";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$theses = array();
while($res = mysqli_fetch_assoc($result)){
    $theses []= $res;
}

mysqli_close($con);
?>
<h3>Επιλογή Πτυχιακής</h3>
<table id="my_table" width="100%"><tbody>
<tr><th>Τίτλος</th><th>Επιβλέπων</th><th>Επιλογή</th></tr>
<?php
foreach($theses as $thesis){
    echo "<tr><td><a
href='view.php?id={$thesis['id']}'>{$thesis['title']}</a></td>
    <td>{$thesis['name']} {$thesis['surname']}</td><td><a
href='select2.php?id={$thesis['id']}'>Επιλογή</a></td></tr>";
}

echo '</tbody></table>';

include('footer.php');
?>
```

15. Αρχείο select2.php

```
<?php include('check_login.php');
    check_login(1);

    include('header.php');

include('db_connect.php');

if (isset($_POST['Select'])) {
    //mark this thesis as selected
    $sql = "UPDATE `theses` SET `status`=1,
student_id=".$_SESSION['user_id']."' WHERE `id`='".$_POST['id'];
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    mysqli_close($con);
    header('location: st_thesis.php');
}

//get id of selected thesis in previous step (select1.php)
$id = '';
if (isset($_GET['id'])) {
    $id = $_GET['id'];
}

$status = 0;

$sql = "SELECT * FROM `theses` WHERE `id`=$id";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$thesis = mysqli_fetch_assoc($result);

$sql = "SELECT * FROM `users` WHERE `id`={$thesis['supervisor_id']}";
$result = mysqli_query($con, $sql) OR die (mysqli_error($con));

$sv = mysqli_fetch_assoc($result);

$st = [];
if ($thesis['student_id'] > 0) {
    $sql = "SELECT * FROM `users` WHERE `id`={$thesis['student_id']}";
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));

    $st = mysqli_fetch_assoc($result);
}

mysqli_close($con);

echo "<h3>{$thesis['title']}</h3>";

//supervisor data
echo '<table id="my_table"><tbody>';
```

```

echo '<tr><th>Επιβλέπων</th><th>Email</th><th>Τηλέφωνο</th></tr>';
echo "<tr><td><a href='by_sv.php?id={\$sv['id']}'>{\$sv['name']}
{\$sv['surname']}</a></td><td>{\$sv['email']}</td><td>{\$sv['tel']}</td></tr>";
echo '</tbody></table>';

?>

<?php
echo '<table id="my_table"><tbody>';
echo "<tr><td><b>Τίτλος</b></td><td>{\$thesis['title']}</td></tr>";
echo "<tr><td><b>Περιγραφή</b></td><td>{\$thesis['description']}</td></tr>";
echo "<tr><td><b>Λέξεις-Κλειδιά</b></td><td>{\$thesis['keywords']}</td></tr>";
echo
"<tr><td><b>Προαπαιτούμενα</b></td><td>{\$thesis['requirements']}</td></tr>";
echo '</tbody></table>';
?>

    Αν επιλέξετε αυτή τη πτυχιακή, ο επιβλέπων θα ενημερωθεί για να την
εγκρίνει.
    <form id="form" name="form" method="post" action="select2.php?id=<?php
echo \$id;?>" >
        <input type="hidden" name="id" value="<?php echo \$id;?>">
        <input type="submit" name="Select" value="Επιλογή">
    </form>
<?php
include('footer.php');
?>

```

16. Αρχείο submit.php

```
<?php
include('check_login.php');
check_login(1); //student access
include('header.php');

if (isset($_POST['Submit'])) {

    $abstract = $_POST['abstract'];
    $id = $_POST['id'];

    $uploadOk = 0;
    $msg = '';
    if(isset($_FILES["manuscript"])) {
        $target_dir = "uploads/";
        $input_file = basename($_FILES["manuscript"]["name"]);
        $target_file = $target_dir.$id.'_'.$input_file; //add id as prefix
        $uploadOk = 1;

        // Warning if file already exists
        if (file_exists($target_file)) {
            $msg .= "Το αρχείο υπάρχει ήδη και θα αντικατασταθεί.<br />";
        }

        // Check file size
        $FILE_SIZE_LIMIT = 20000000; //20MB
        if ($_FILES["manuscript"]["size"] > $FILE_SIZE_LIMIT) {
            $msg .= "Το αρχείο είναι μεγαλύτερο από 20MB.<br />";
            $uploadOk = 0;
        }

        // Check if $uploadOk is set to 0 by an error
        if ($uploadOk == 0) {
            $msg .= "Το αρχείο δεν αποθηκεύτηκε.<br />";
            // if everything is ok, try to upload file
        } else {
            if (move_uploaded_file($_FILES["manuscript"]["tmp_name"],
            $target_file)) {
                $msg .= "Το αρχείο $input_file αποθηκεύτηκε επιτυχώς!<br />Η
                πτυχιακή υποβλήθηκε στον επιβλέποντα για έγκριση.";
            } else {
                $msg .= "Προέκυψε σφάλμα στη λήψη του αρχείου.<br />";
                $uploadOk = 0;
            }
        }
    }

    //save file data to db
    if ($uploadOk) {
        include('db_connect.php');
        $sql = "INSERT INTO `files` (`thesis_id`, `filename`) VALUES ($id,
        '$target_file')";
        $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    }
}
```

```

        $sql = "UPDATE `theses` SET `abstract`='$abstract', `end`=NOW(),
`status`=3 WHERE `id`='$id";
        $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
        mysqli_close($con);
    }

```

```

    echo $msg;

```

```

}else {
    //show form
    $id = $_GET['id'];
    //load
    include('db_connect.php');
    $sql = 'SELECT * FROM `theses` WHERE `id`='.$id;
    $result = mysqli_query($con, $sql) OR die (mysqli_error($con));
    $thesis = mysqli_fetch_assoc($result);
    mysqli_close($con);

```

```

    $title = $thesis['title'];

```

```

?>

```

```

<script type="text/javascript">

```

```

    function validateForm()
    {
        var x=document.forms["form"]["abstract"].value;
        if (x==null || x=="")
        {
            alert("Abstract must not be empty!");
            return false;
        }
        else{
            return true;
        }
    }

```

```

</script>

```

```

<h3>Υποβολή Πτυχιακής</h3>

```

```

<form id="form" name="form" method="post" action="submit.php<?php echo
'?id='.$id;?>" enctype="multipart/form-data">

```

```

    <input type="hidden" name="id" value="<?php echo $id;?>">

```

```

    <table border="0" >

```

```

        <tr>

```

```

            <td><strong>Τίτλος</strong></td>

```

```

            <td><input name="title" type="text" value="<?php echo $title; ?>"

```

```

size="128"/></td>

```

```

        </tr>

```

```

        <tr>

```

```

            <td><strong>Περίληψη</strong></td>

```

```

            <td><textarea name="abstract" id="abstract" cols="64"

```

```

rows="8"></textarea></td>

```

```

        </tr>

```

```

        <tr>

```

```

            <td><strong>Κείμενο</strong></td>

```

```

            <td><input type="file" name="manuscript" id="manuscript" /></td>

```



```
        </tr>
        <tr>
            <td></td>
            <td>
                <input type="submit" name="Submit" value="Υποβολή"
onclick="return validateForm()" />
                <input type="reset" name="Cancel" value="Άκυρο" />
            </td>
        </tr>
    </table>

<?php
} //endif

include('footer.php');
?>
```

17. Αρχείο header.php

```
<?php
$statuses = [0 => 'Διαθέσιμη',
            1 => 'Ζητήθηκε',
            2 => 'Σε εξέλιξη',
            3 => 'Υποβλήθηκε',
            4 => 'Ολοκληρωμένη',
            5 => 'Διακόπηκε',
            '' => ''];

$roles = [1 => 'Φοιτητής', 2 => 'Επιβλέπων', 3 => 'Διαχειριστής'];

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<title>Πτυχιακές</title>

<link href="style.css" rel="stylesheet" type="text/css" />

</head>

<body>

<div id="container">
  <div id="header">
    <div id="my_title">
      <a href="index.php" style="text-decoration: none; color:
black;><h2>Πτυχιακές</h2></a>
    </div>
    <div id="menu_container">
      <ul id="menu">
        <?php
          if (isset($_SESSION['logged'])) {
            echo '<li><a href="list.php">Λίστα</a></li>';
            echo '<li><a href="supervisors.php">Επιβλέποντες</a></li>';
            echo '<li><a href="students.php">Φοιτητές</a></li>';

            if ($_SESSION['role'] == 1) {
              echo '<li><a href="st_thesis.php">Η Πτυχιακή μου</a></li>';
            }
            if ($_SESSION['role'] == 2) {
              echo '<li><a href="sv_theses.php">Οι Πτυχιακές
μου</a></li>';
            }
            if ($_SESSION['role'] == 3) {
              echo '<li><a href="admin.php">Διαχείριση</a></li>';
            }
          }
        </?php
      </ul>
    </div>
  </div>
</div>

```

```
        echo '<li><a href="logout.php">Εξοδος</a></li>';
    }else{
        echo '<li><a href="index.php">Είσοδος</a></li>';
        echo '<li><a href="register.php">Εγγραφή</a></li>';
    }

    ?>
</ul>
</div>
</div><!-- end #header -->
<div id="mainContent">
```

18. Αρχείο footer.php

```
<br /><br />  
</div> <!-- end #main content-->  
</div<!-- end #container -->  
</body>  
</html>
```

19. Αρχείο db_connect.php

```
<?php
    $con = mysqli_connect("localhost","root","");
    if (!$con)
    {
        die('Could not connect #1: ' . mysqli_error());
    }

    $db_selected = mysqli_select_db($con, "theses");
    if (!$db_selected) {
        die ('Could not connect #2: ' . mysqli_error());
    }
    mysqli_query($con, 'SET NAMES utf8');
```

?>

20. Αρχείο style.css

```
body {
    font: 100% Verdana, Arial, Helvetica, sans-serif;
    background: #0C60B0;
    margin: 0; /* it's good practice to zero the margin and padding of the
body element to account for differing browser defaults */
    padding: 0;
    text-align: center; /* this centers the container in IE 5* browsers. The
text is then set to the left aligned default in the #container selector */
    color: #000000;
}
#container {
    width: 100%;
    background: #FFFFFF;
    margin: 0 auto; /* the auto margins (in conjunction with a width) center
the page */
    padding: 0;
    border-bottom: 1px solid #000000;
    text-align: left; /* this overrides the text-align: center on the body
element. */
}
#header {
    background: #E0E0E0;
    margin: 0;
    padding: 0;
}
#my_title {
    margin: 0; /* zeroing the margin of the last element in the #header div
will avoid margin collapse - an unexplainable space between divs. If the div
has a border around it, this is not necessary as that also avoids the margin
collapse */
    padding: 5px 0 0 20px;
}
#menu_container {
    padding: 0px 0px 0px 20px ;
    margin: 0;
    height: 41px;
}
ul#menu {
    padding: 0px;
    margin: 0px;
    list-style-type: none;
    height: 41px;
    width: 670px;
}
ul#menu li {
    //display: inline;
    float: left;
    margin: 0px;
    padding: 0px;
}
```

```

    margin-right: 3px;
}

ul#menu li a {
    background-color: #ffffff;
    padding: 10px 20px;
    text-decoration: none;
    color: #696969;
    border-radius: 4px 4px 0 0;
    display: block;
}

ul#menu li a:hover {
    color: white;
    background-color: black;
}

#mainContent {
    margin: 0;
    padding: 10px 20px;
    background: #FFFFFF;
}

#my_table table{
    border-collapse: collapse;
}

td.even{
    background-color: #CCFFFF;
}

#my_table tbody, #my_table th, #my_table td{
    border: 1px solid #A0A0A0;
    border-collapse: collapse;
    cellspacing: 0px;
}

div.err{
    color: red;
    font-weight: bold;
}

```

21. Αρχείο theses.sql

Αυτό το αρχείο είναι μια εξαγωγή του σχήματος της βάσης μας από το phpmyadmin.

```
-- phpMyAdmin SQL Dump
-- version 4.1.14
-- http://www.phpmyadmin.net
--
-- Host: 127.0.0.1
-- Generation Time: Oct 31, 2015 at 07:26 PM
-- Server version: 5.6.17
-- PHP Version: 5.5.12

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

--
-- Database: `theses`
--

-----

--
-- Table structure for table `files`
--

CREATE TABLE IF NOT EXISTS `files` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `thesis_id` int(11) NOT NULL,
  `filename` varchar(64) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-----

--
-- Table structure for table `theses`
--

CREATE TABLE IF NOT EXISTS `theses` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `supervisor_id` int(11) NOT NULL,
  `student_id` int(11) NOT NULL,
  `title` varchar(128) NOT NULL,
  `description` varchar(256) NOT NULL,
  `requirements` varchar(128) NOT NULL,
  `keywords` varchar(128) NOT NULL,
  `status` int(11) NOT NULL,
  `start` date NOT NULL,
```



```

    `end` date NOT NULL,
    `abstract` text NOT NULL,
    PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-----

--
-- Table structure for table `users`
--

CREATE TABLE IF NOT EXISTS `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(16) NOT NULL,
  `surname` varchar(24) NOT NULL,
  `email` varchar(64) NOT NULL,
  `password` char(40) NOT NULL,
  `tel` char(10) NOT NULL,
  `role` tinyint(4) NOT NULL,
  `active` tinyint(4) NOT NULL,
  `am` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```