



ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Διαδικτυακά εργαλεία για τη διαχείριση πτυχιακών
εργασιών του τμήματος πληροφορικής**



Τμήμα Μηχανικών
Πληροφορικής ΑΤΕΙΘ

Του φοιτητή

Μαυροφείδη Αθανάσιου

Αρ. Μητρώου: 063050

Επιβλέπων καθηγητής

Σιδηρόπουλος Αντώνιος

Θεσσαλονίκη 2015

ΠΕΡΙΛΗΨΗ

Στην παρούσα εργασία παρουσιάζουμε μία εφαρμογή διαχείρισης πτυχιακών εργασιών το οποίο σχεδιάστηκε για το Αλεξάνδρειο Τεχνολογικό Ίδρυμα Θεσσαλονίκης στα πλαίσια της πτυχιακής εργασίας του τμήματος Πληροφορικής. Η σχεδίαση αυτής της εφαρμογής αποσκοπεί στη μηχανογράφηση των συγκεκριμένων λειτουργιών διαχείρισης που περιλαμβάνει ο θεσμός των πτυχιακών εργασιών. Κύριες λειτουργίες της εφαρμογής είναι η δυνατότητα εύρεσης πτυχιακών εργασιών με τις πληροφορίες τους, ανάθεσης τους στους φοιτητές από κάποιον αρμόδιο, δημιουργίας τους από τους καθηγητές και επιλογές των πτυχιακών εργασιών από τους φοιτητές. Μέσα σε αυτές τις λειτουργίες προστίθεται και η λειτουργία της γραμματείας που είναι υπεύθυνη για την ύπαρξη των βοηθητικών εγγράφων που αφορούν τον θεσμό των πτυχιακών εργασιών.

ABSTRACT

In this paper we present a management application for the final theses, which was designed for the Alexander Technological Institute of Thessaloniki for the final theses of the IT department. The design of this application is aimed in office automation of certain procedures of service management which are included in the institution of final theses. Main functions of the application are the ability of finding final theses with their information, the assignment of them to possible candidates from someone who is responsible for that, the ability of creating final theses by the teachers and finally the ability for the students to choose their final theses. In addition we have the operation of secretary which is responsible for the existence of supporting documents which are relative to the institution.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ	3
ABSTRACT	4
ΠΕΡΙΕΧΟΜΕΝΑ	5
Ευρετήριο εικόνων.....	6
Ευρετήριο πινάκων.....	6
1.ΕΙΣΑΓΩΓΗ	7
1.1 Στόχος.....	7
1.2 Χρησιμότητα.....	7
1.3 Επισκόπηση.....	7
2.ΑΝΑΛΥΣΗ	8
2.1 Εισαγωγική ανάλυση.....	8
2.1.1 Σκοπός.....	8
2.1.2 Γενική περιγραφή	8
2.2 Ανάλυση χρηστών.....	8
2.2.1 Χαρακτηριστικά – Ικανότητες – Επιθυμίες χρηστών.....	8
2.3 Ανάλυση λειτουργιών	10
3. ΣΧΕΔΙΑΣΗ.....	11
3.1 Απαιτήσεις συστήματος.....	11
3.1.1 Προγράμματα που χρησιμοποιήθηκαν	11
3.1.2 Περιγραφή τεχνολογιών	12
3.2 Περιγραφή της βάσης δεδομένων	17
3.3 Βάση δεδομένων και πίνακες	18
3.4 Η μορφή της εφαρμογής	29
3.5 Η ροή των εργασιών	29
3.6 Λογική κώδικα	31
3.7 Παρουσίαση εφαρμογής.....	34
4. ΑΠΟΤΕΛΕΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ	39
5. ΒΙΒΛΙΟΓΡΑΦΙΑ	40

Ευρετήριο εικόνων

Εικόνα 3.1 Το αντικείμενο XMLHttpRequest.....	15
Εικόνα 3.2 Τα βήματα της εφαρμογής	29
Εικόνα 3.3 Η ροή των εργασιών	30
Εικόνα 3.4 Κώδικας addEditSaveCourse.php	32
Εικόνα 3.5 Κώδικας addNewCourse.php	33
Εικόνα 3.6 Αποδοχή – διαγραφή νέων χρηστών	34
Εικόνα 3.7 Η σελίδα της ανάθεσης εργασιών	35
Εικόνα 3.8 Καταχώρηση ημερομηνιών και ορισμό συνεξεταστών.....	36
Εικόνα 3.9 Επιλογή πτυχιακών εργασιών.....	37
Εικόνα 3.10 Φιλτράρισμα πτυχιακών εργασιών από τη γραμματεία.....	38

Ευρετήριο πινάκων

Πίνακας 3.1 usrTypes.....	18
Πίνακας 3.2 users.....	19
Πίνακας 3.3 courses.....	20
Πίνακας 3.4 theses.....	21
Πίνακας 3.5 prerequisites	22
Πίνακας 3.6 assignments	22
Πίνακας 3.7 choices	23
Πίνακας 3.8 presentations	25
Πίνακας 3.9 necessaryDocs	26
Πίνακας 3.10 semesters.....	27
Διάγραμμα Βάσης Δεδομένων	28

1.ΕΙΣΑΓΩΓΗ

1.1 Στόχος

Ο στόχος της παρούσας εργασίας είναι η δημιουργία λογισμικού διαχείρισης πτυχιακών εργασιών του τμήματος Πληροφορικής του ΑΤΕΙΘ. Βέβαια δεν είναι έτοιμη ακόμη για να ενσωματωθεί στο σύστημα. Το σύστημα υποστηρίζει και απευθύνεται σε φοιτητές, διδάσκοντες, στη γραμματεία και γενικά εξωτερικούς χρήστες (επισκέπτες), που έχουν την δυνατότητα πραγματοποίησης, προσπέλασης και ανάκτησης δεδομένων.

1.2 Χρησιμότητα

Η εργασία υλοποιήθηκε με χρήση της γλώσσας PHP, μιας σύγχρονης και με πολλές δυνατότητες γλώσσας για προγραμματισμό διαδικτύου. Μεγάλο ρόλο στη δημιουργία και οργάνωση του κώδικα έπαιξε το PhpStorm ένα ισχυρό IDE. Επίσης χρησιμοποιήθηκε η MySQL για την διατύπωση δομημένων αιτημάτων και την διαχείριση των δεδομένων της βάσης δεδομένων της εφαρμογής. Επίσης απαιτήθηκε και η χρήση της γλώσσας HyperText Markup Language (HTML).

1.3 Επισκόπηση

Η επισκόπηση αποτελεί έναν οδηγό του συστήματος που αναπτύχθηκε. Περιλαμβάνει τα παρακάτω κεφάλαια:

- 2.Ανάλυση
- 3.Σχεδίαση
- 4.Συμπεράσματα
- 5.Βιβλιογραφία

Η ανάλυση περιλαμβάνει τον προσδιορισμό των αναγκών του συστήματος διαχείρισης πτυχιακών εργασιών που αναπτύχθηκε. Επίσης περιλαμβάνει λεπτομερή ανάλυση των διάφορων κατηγοριών του συστήματος. Επιπλέον, υπάρχει και η ανάλυση των διαφόρων λειτουργιών που είναι διαθέσιμες στους χρήστες.

Η σχεδίαση αφορά την περιγραφή θεμάτων σχετικών με την σχεδίαση του λογισμικού διαχείρισης πτυχιακών εργασιών. Πιο αναλυτικά, αρχικά παρουσιάζεται μια λεπτομερής περιγραφή της βάσης δεδομένων που χρησιμοποιήθηκε. Στη συνέχεια, παρουσιάζεται το σχέδιο της βάσης δεδομένων και η αναλυτική σχεδίαση των πινάκων. Τέλος ακολουθεί το τμήμα που περιγράφει της βασικότερες οθόνες της διεπιφάνειας του συστήματος με επεξήγηση της λειτουργίας τους και των βασικών στοιχείων υλοποίησής τους.

2.ΑΝΑΛΥΣΗ

2.1 Εισαγωγική ανάλυση

2.1.1 Σκοπός

Ο σκοπός αυτής της ενότητας της εργασίας είναι ο προσδιορισμός των και των περιορισμών του συστήματος που υλοποιήθηκε και η ανάλυση των χρηστών του και των κυριότερων λειτουργιών τους που μπορούν να εκτελέσουν.

2.1.2 Γενική περιγραφή

Η συγκεκριμένη εφαρμογή αποτελεί ένα σύστημα διαχείρισης των πτυχιακών εργασιών και αποτελεί προϊόν σχεδίασης που αποσκοπεί στη μηχανοργάνωση των συγκεκριμένων λειτουργιών διαχείρισης που περιλαμβάνει ο θεσμός των πτυχιακών εργασιών.

Αρχικά γίνεται η σχεδίαση της βάσης δεδομένων με προσδιορισμό των πινάκων, των πεδίων τους και του τύπου δεδομένων του καθενός καθώς και οι σχέσεις μεταξύ τους. Κεντρικός άξονας της σχεδίασης είναι η αποτελεσματική υποστήριξη των λειτουργιών που θα επιτελεί το σύστημα.

Στη συνέχεια γίνεται ο σχεδιασμός των διεπιφανειών για κάθε κατηγορία χρήστη που υποστηρίζει το σύστημα, με κριτήριο σχεδιασμού την ευχρηστία του συστήματος σε σχέση με το είδος του χρήστη. Πέντε είναι οι κατηγορίες χρηστών:

Επισκέπτης
Admin
Φοιτητής
Καθηγητής
Γραμματεία

Η πρώτη κατηγορία πρέπει να έχει εργαλεία για την εκτέλεση ανάκλησης σχετικών πληροφοριών απο το σύστημα, ενώ οι υπόλοιπες τρεις έχουν αυξημένα δικαιώματα για την προσθήκη, αφαίρεση, τροποποίηση δεδομένων του συστήματος μέσω αντίστοιχων λειτουργιών.

2.2 Ανάλυση χρηστών

2.2.1 Χαρακτηριστικά – Ικανότητες – Επιθυμίες χρηστών

Όπως συμβαίνει με κάθε προϊόν λογισμικού, αρχικά καταγράφονται οι λειτουργικές απαιτήσεις. Είναι το σημείο όπου καθορίζουμε τους πιθανούς χρήστες και τους τρόπους με τους οποίους είναι δυνατό να αλληλεπιδρούν με την εφαρμογή. Περιγράφουμε λοιπόν τις λειτουργίες που υποστηρίζει η εφαρμογή πράγμα που συνιστά το εννοιολογικό σχέδιο, με απλά λόγια το «τι» της εφαρμογής. Γενικά, η ανάλυση χρηστών αφορά τον προσδιορισμό των τυπικών χρηστών του συστήματος και στην συνέχεια την ανάλυση των χαρακτηριστικών τους από την οποία θα προκύψουν χρήσιμα συμπεράσματα για τον σχεδιασμό του συστήματος με τρόπο που να ικανοποιεί τις ανάγκες των διαφόρων κατηγοριών χρηστών. Πιο συγκεκριμένα καταγράφονται τα ατομικά χαρακτηριστικά, οι

ικανότητες, οι επιθυμίες και οι περιορισμοί για τις διάφορες κατηγορίες χρηστών του συστήματος.

Στο σύστημα υπάρχουν πέντε κατηγορίες χρηστών:

- Επισκέπτες
- Admin
- Φοιτητές
- Καθηγητές
- Γραμματεία

Ο Admin έχει δικαιώματα για όλους τους ρόλους. Μπορεί να κάνει τα πάντα και έχει πρόσβαση παντού μέσα στην εφαρμογή.

Για την κατηγορία των φοιτητών, θεωρείται αυτονόητο η μεγάλη εξοικείωση με την χρήση υπολογιστών και παρόμοιων τέτοιων συστημάτων. Γενικά οι χρήστες θέλουν το περιβάλλον του συστήματος να είναι κατανοητό, με απλούς τρόπους πλοήγησης παρόμοιους με αυτούς που έχουν συνηθίσει μέχρι σήμερα.

Συγκεκριμένα, οι φοιτητές χρησιμοποιούν το σύστημα για την ανάκτηση πληροφοριών, σχετικά με το ποιές εργασίες διενεργούνται, από ποιους φοιτητές, υπό την επίβλεψη ποιων καθηγητών. Επομένως το σύστημα πρέπει να παρέχει στους φοιτητές τη δυνατότητα εκτέλεσης λειτουργιών αναζήτησης για την εμφάνιση των πληροφοριών που τους ενδιαφέρουν.

Για την κατηγορία διδάσκοντες το περιβάλλον είναι επίσης πολύ φιλικό. Μπορεί να διαχειρίζεται τις πτυχιακές που επιβλέπει, να βαθμολογεί εργασίες που επιβλέπει ή είναι συνεξεταστής, να ορίζει ημερομηνία παρουσίασης κλπ.

Το προσωπικό της γραμματείας επίσης θεωρείται αυτονόητο το ότι έχει στοιχειώδεις γνώσεις χρήσης υπολογιστών και κάποια εμπειρία και εξοικείωση με παρόμοια συστήματα. Και εδώ επιθυμία των χρηστών είναι η ευχρηστία και η ευκολία αντίληψης και εκμάθησης του συστήματος. Πιο συγκεκριμένα, στους εξουσιοδοτημένους αυτούς χρήστες θα πρέπει να εξασφαλίζεται η δυνατότητα αναζήτησης πτυχιακών εργασιών ανά έτος ή ανά διδάσκοντα ή ανά φοιτητή. Επίσης, κύρια αρμοδιότητα της γραμματείας είναι να μπορεί να βλέπει την βαθμολογία των φοιτητών καθώς επίσης και τις πτυχιακές που βρίσκονται σε εξέλιξη. Φυσικά, όλες αυτές οι λειτουργίες θα πρέπει να είναι απλές στην εκτέλεση, εύκολες στην εκμάθηση, και κατανοητές διότι αποτελούν τακτικές εργασίες του προσωπικού και φυσικά θα πρέπει να γίνονται με τρόπο που να εξασφαλίζεται η προστασία των δεδομένων από μη εξουσιοδοτημένη πρόσβαση.

2.3 Ανάλυση λειτουργιών

Σ' αυτή την ενότητα θα παρουσιαστούν οι διάφορες διεργασίες της εφαρμογής ανάλογα με τον χρήστη.

Επισκέπτης

Αναζήτηση για παλαιότερες ολοκληρωμένες εργασίες με βάση:

- Ημερομηνία
- Θέμα της πτυχιακής
- Επιβλέποντα καθηγητή
- Φοιτητή

Τα αποτελέσματα μπορούν να εμφανιστούν ταξινομημένα (αύξουσα ή φθίνουσα σειρά) κατά: Ημερομηνία ή θέμα ή επιβλέποντα. Βέβαια η ταξινόμηση μπορεί να γίνει και μετά την προβολή με βάση τα κριτήρια που θέλουμε.

Φοιτητής

- Αναζήτηση της διαθέσιμης πτυχιακής με λέξεις κλειδιά ή με τον επιβλέποντα καθηγητή
- Εμφάνιση πτυχιακών που έχει εκδηλώσει ενδιαφέρον
- Διαχείριση ομάδων: Δημιουργία μιας καινούριας ομάδας και προσθήκη μελών
- Διαχείριση ομάδων: Να συμμετέχει στην ομάδα κάποιου άλλου

Καθηγητής

- Δημιουργία θέματος πτυχιακής (τίτλος, περιγραφή, ομαδική ή όχι, μέγιστο αριθμό μελών, αρχείο περιγραφής)
- Εμφάνιση πτυχιακών που επιβλέπει
- Διαχείριση αναθέσεων που επιβλέπει
- Ορισμό συνεξεταστών
- Ορισμό ημερομηνίας παρουσίασης
- Εμφάνιση βαθμολογίας
- Ολοκλήρωση βαθμολογίας
- Εμφάνιση απενεργοποιημένων θεμάτων
- Αποθήκευση εντύπου βαθμολογίας

- Εκτύπωση εντύπου βαθμολόγησης

Γραμματεία

- Εμφάνιση βαθμολογίας
- Εμφάνιση αναθέσεων πτυχιακών που είναι σε εξέλιξη
- Όλα τα έγγραφα που είναι απαραίτητα για την ανάλυση και ολοκλήρωση της πτυχιακής σε μορφή pdf

Admin

Έχει πρόσβαση παντού και είναι υπεύθυνος για την έγκριση των πτυχιακών εργασιών.

3. ΣΧΕΔΙΑΣΗ

3.1 Απαιτήσεις συστήματος

Στο υποκεφάλαιο αυτό θα γίνει μια σύντομη παρουσίαση των προγραμμάτων που χρησιμοποιήθηκαν για τη σχεδίαση και ανάπτυξη της εφαρμογής, όπως και μια μικρή εισαγωγή και επεξήγηση των τεχνολογιών που χρησιμοποιήθηκαν.

Επίσης θα γίνει μια μικρή αναφορά στον ορισμό «Διαδικτυακές υπηρεσίες» οι οποίες στις μέρες μας χρησιμοποιούνται όλο και περισσότερο από όλους.

3.1.1 Προγράμματα που χρησιμοποιήθηκαν

- Wamp
- PhpStorm
- SQLyog

PhpStorm είναι ένα ολοκληρωμένο cross-platform IDE για την PHP, χτισμένο πάνω στο JetBrains IntelliJ IDEA platform. Παρέχει ένα πρόγραμμα επεξεργασίας για την PHP, την HTML και JavaScript με on-the-fly ανάλυση κώδικα, την πρόληψη σφαλμάτων και αυτοματοποιημένα refactorings για τον κώδικα. Υποστηρίζει PHP 5.3, 5.4, 5.5 και 5.6. Περιλαμβάνει ένα ολοκληρωμένο SQL editor. Το PhpStorm είναι χτισμένο όπως είπαμε πάνω στο JetBrains IntelliJ IDEA platform που είναι γραμμένο σε Java. Όλα τα χαρακτηριστικά που είναι διαθέσιμα στο WebStorm περιλαμβάνονται στο PhpStorm.

WAMP είναι το αρκτικόλεξο για το Windows Apache MySQL PHP/Python/Perl. Αυτό το αρκτικόλεξο αναφέρεται σε ένα set open source εφαρμογών, σε συνδυασμό με τα Microsoft Windows, τα οποία χρησιμοποιούνται σε περιβάλλοντα Web server. Το WAMP παρέχει στους προγραμματιστές τα τέσσερα βασικά στοιχεία ενός διακομιστή web. Ένα λειτουργικό σύστημα, βάση δεδομένων,

διακομιστή web και web-scripting λογισμικού. Αυτά τα στοιχεία είναι Microsoft Windows το λειτουργικό, η MySQL χειρίζεται τη βάση δεδομένων, Apache ο διακομιστής και PHP, Python ή Perl οι δυναμικές γλώσσες scripting.

SQLyog είναι ένα GUI (graphical user interface) εργαλείο για την RDBMS SQL. Είναι διαθέσιμο και δωρεάν και επί πληρωμής. Είναι γραμμένο σε γλώσσα C++ χρησιμοποιώντας το Win32 API. Δουλεύει πάνω στην πλατφόρμα των windows (windowsXP to windows 8).

3.1.2 Περιγραφή τεχνολογιών

Οι τεχνολογίες που χρησιμοποιήθηκαν και θα περιγραφούν εν συντομία στη παρούσα πτυχιακή είναι:

- HTML
- CSS
- JavaScript
- DOM
- AJAX
- JSON
- PHP
- MySQL
- Apache

HTML Είναι η γλώσσα στην οποία είναι γραμμένες οι σελίδες. Επιτρέπει στις εικόνες να συνδυάζονται με το κείμενο και προσφέρει ένα ευρύ φάσμα δυνατοτήτων μορφοποίησης. Ένα από τα πιο σημαντικά χαρακτηριστικά της HTML είναι ο υπέρ-σύνδεσμος (hypertext), που είναι ένα μέρος της ιστοσελίδας που συνδέεται με μια άλλη ιστοσελίδα. Κάνοντας κλικ σε μια υπερ-σύνδεση ανακατευθύνει τον περιηγητή σε άλλη σελίδα. Η HTML βασίζεται στις ετικέτες σήμανσης (markup tags), οι οποίες συχνά αποκαλούνται HTML ετικέτες (HTML tags). Οι ετικέτες αυτές είναι λέξεις-κλειδιά που περιβάλλονται από γωνιακές αγκύλες όπως πχ. <html>. Οι ετικέτες έρχονται σε ζεύγη αρχής . <p> και τέλους </p> όπως επίσης ονομάζονται ετικέτες ανοίγματος και κλεισίματος. Είναι ένα αρχείο που περιέχει διάφορες HTML ετικέτες και σκέτο κείμενο και το οποίο έχει κατάληξη .html, ιστοσελίδα (web page).

CSS Το πρότυπο CSS είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ. Είναι ένα σύνολο από εντολές τις οποίες ο φυλλομετρητής (browser) ακολουθεί ώστε να αλλάξει την προεπιλεγμένη παρουσίαση των HTML στοιχείων ανάλογα με τις τιμές που έχουν οριστεί. Μέσω

του προτύπου CSS επιτυγχάνεται η εύκολη και μαζική μορφοποίηση, δίνοντας τη δυνατότητα στο σχεδιαστή να ξεχωρίζει το περιεχόμενο των σελίδων από το σχεδιασμό, έχοντας ως αποτέλεσμα οι αλλαγές καθώς και οι επανασχεδιασμοί να γίνονται πολύ πιο εύκολα σε μια ιστοσελίδα.

Η εισαγωγή ενός CSS προτύπου σε μια HTML σελίδα μπορεί να γίνει με δύο τρόπους:

Ενσωματώνοντας τον CSS δηλώσεις μέσα στα <head> </head> tags ο οποίος κώδικας θα πρέπει να βρίσκεται ανάμεσα στα <Style type="TEXT/CSS"> </Style> tags. Δημιουργώντας ένα αρχείο στο οποίο θα υπάρχουν όλες οι δηλώσεις το αρχείο θα εισαχθεί στην HTML σελίδα μας μέσα στα <head></head> tags με αυτή τη δήλωση <link rel="stylesheet" type="text/css" href="something.css" />. Το αρχείο αυτό όπως φαίνεται, πρέπει να έχει την κατάληξη «.css».

JavaScript Η Javascript είναι μια διερμηνευτική (interpreted) γλώσσα προγραμματισμού που τρέχει στην πλευρά του φυλλομετρητή (client side), το οποίο την κάνει πιο γρήγορη από κάποια άλλη γλώσσα που τρέχει στον διακομιστή (server side). Φυσικά αυτό έχει ένα μειονέκτημα ότι ο πηγαίος κώδικας είναι προσβάσιμος από όλους τους χρήστες της ιστοσελίδας.

Σχεδιάστηκε για να δώσει στους προγραμματιστές ένα εργαλείο με απλή σύνταξη. Η Javascript, όπως και όλες οι γλώσσες προγραμματισμού, έχει μεταβλητές, αλφαριθμητικά, πίνακες, αντικείμενα συναρτήσεων και όλες τις συνηθισμένες δομές ελέγχου. Διαθέτει λειτουργίες ειδικά για ιστοσελίδες, όπως δυνατότητα διαχείρισης παραθύρων και πλαισίων, αποστολή-λήψη και δημιουργία-κατάργηση cookies, χειρισμός φορμών και άλλων στοιχείων που βρίσκονται στη σελίδα.

Έχει επίσης πολλές ιδιότητες αντικειμενοστραφούς γλώσσας και πολλές ομοιότητες με τις άλλες γλώσσες αφού κτίστηκε πάνω στο πρότυπο των γλωσσών C, C++ και Java. Έχει όμως μία πολύ σημαντική διαφορά, διαχειρίζεται τους τύπους δεδομένων πολύ πιο χαλαρά, για παράδειγμα μια μεταβλητή στην αρχή μπορεί να είναι τύπου *int* και στη συνέχεια να γίνει τύπου *string*. Αξίζει να σημειωθεί στο παρόν στάδιο πως η Java και η Javascript είναι δυο εντελώς διαφορετικές γλώσσες προγραμματισμού.

Με την JavaScript μπορεί να επιτευχθεί η μειωμένη αποστολή δεδομένων στον διακομιστή (server) όπως και το φιλτράρισμα των δεδομένων που θα αποσταλούν, λόγω του ότι δίνεται η δυνατότητα να ελεγχθούν τα δεδομένα ή ακόμη και να αλλάξουν μέσα από την φόρμα πριν την αποστολή τους στον διακομιστή. Επίσης υπάρχει η δυνατότητα να αλλάξουν οι τιμές και ιδιότητες στοιχείων της σελίδα δυναμικά πολύ εύκολα.

Τέλος, η JavaScript υποστηρίζεται από όλους τους browsers και προσφέρει ένα καλύτερο και ελαφρότερο γραφικό περιβάλλον αφού τα σενάρια (scripts) είναι μικρά σε μέγεθος και αποθηκεύονται στη μνήμη του browser μόλις αυτά κατέβουν.

DOM Το Μοντέλο Αντικειμενικού Εγγράφου χρησιμοποιείται για την δυναμική έκθεση των δεδομένων καθώς και για την αλληλεπίδραση με αυτά. Συγκεκριμένα το DOM είναι μια σύμβαση για τον συμβολισμό και την αλληλεπίδραση των αντικειμένων μέσα σε ένα έγγραφο HTML, XHTML και XML. Είναι ανεξάρτητο του λειτουργικού συστήματος και της γλώσσας προγραμματισμού που χρησιμοποιείται. Ένας φυλλομετρητής δεν είναι υποχρεωμένος να χρησιμοποιεί το μοντέλο αυτό για να απεικονίσει ένα έγγραφο HTML. Ωστόσο απαιτείται από τη γλώσσα Javascript όταν πρόκειται να επιθεωρήσει (inspect) ή επεξεργαστεί περαιτέρω το έγγραφο, τα αποτελέσματα αυτά να ενσωματώνονται στον ιστότοπο που παρουσιάζεται στο φυλλομετρητή.

Το DOM αποδομεί και παρουσιάζει ένα HTML έγγραφο σε δενδρική μορφή. Κατά τη διάρκεια αυτής της διαδικασίας απομνημονεύει αρχικά όλο το έγγραφο, για το λόγο αυτό τα προγράμματα που χρησιμοποιούν DOM επιβάλλεται να έχουν μεγάλη μνήμη.

AJAX Για χρόνια, οι σχεδιαστές ιστοσελίδων παραπονούνταν για την αδυναμία να δημιουργήσουν εξελιγμένες διεπαφές που να ανταποκρίνονται, κάτι σαν αυτές που βρίσκουμε στις desktop εφαρμογές. Όλα άρχισαν να αλλάζουν το 2005, όταν ο Jesse James Garrett, επινόησε τον όρο Ajax σε ένα άρθρο *"Ajax: Μια νέα προσέγγιση για εφαρμογές στον Παγκόσμιο Ιστό"*, οι οποίες βασίζονται σε τεχνικές που χρησιμοποιούνται στις ιστοσελίδες της Google.

Ajax ένα ακρωνύμιο για Asynchronous JavaScript και XML, είναι ένα σύνολο αλληλένδετων τεχνικών ανάπτυξης ιστοσελίδων που χρησιμοποιείται στην πλευρά του πελάτη και δημιουργεί web εφαρμογές που δουλεύουν ασύγχρονα.

Στον παραδοσιακό προγραμματισμό με JavaScript, αν θέλει κάποιος να πάρει πληροφορίες από μια βάση δεδομένων ή ένα αρχείο από τον server, ή να στείλει πληροφορίες για το χρήστη σε ένα server, θα πρέπει να κάνει μια HTML φόρμα και με χρήση GET ή POST να στείλει τα δεδομένα στο διακομιστή (server). Ο χρήστης θα χρειαστεί να κάνετε κλικ στο κουμπί "Υποβολή" για να στείλει / λάβει τις πληροφορίες, να περιμένει το διακομιστή να απαντήσει, και στη συνέχεια μια νέα σελίδα θα φορτωθεί με τα αποτελέσματα.

Επειδή ο διακομιστής επιστρέφει μια νέα σελίδα κάθε φορά που ο χρήστης υποβάλλει αίτημα, οι παραδοσιακές web εφαρμογές τρέχουν αργά και τείνουν να είναι λιγότερο φιλικές προς το χρήστη. Με την AJAX, η JavaScript επικοινωνεί απευθείας με το διακομιστή, μέσω του αντικειμένου XMLHttpRequest της JavaScript.

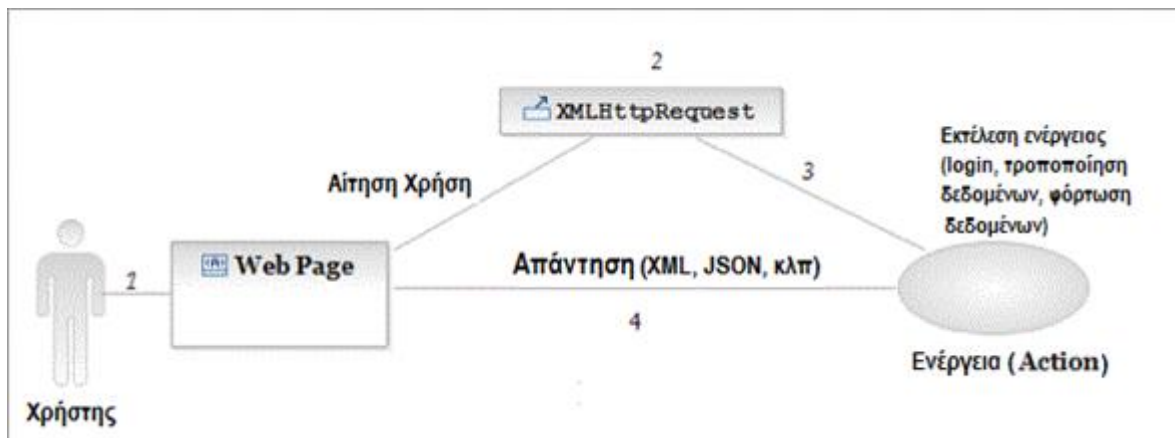
Με αίτηση HTTP, μια ιστοσελίδα μπορεί να υποβάλει μία αίτηση και να πάρει μία απάντηση από έναν web server, χωρίς την επαναφόρτωση της σελίδας. Ο χρήστης θα παραμείνει στην ίδια σελίδα, και δεν θα παρατηρήσει πως σενάρια (scripts) που τρέχουν στο παρασκήνιο αιτούνται σελίδες, ή στέλνουν δεδομένα σε ένα διακομιστή.

Παρά το όνομα της, η χρήση της XML δεν είναι απαραίτητη, αντί αυτού, συχνά χρησιμοποιείται κωδικοποίηση JSON, η οποία έχει χρησιμοποιηθεί στην παρούσα πτυχιακή εργασία.

Το αντικείμενο XMLHttpRequest

Το XMLHttpRequest είναι μία διεπαφή ανάπτυξης εφαρμογών (API) του μοντέλου DOM. Μπορεί να χρησιμοποιηθεί σε μία γλώσσα προγραμματισμού client-side όπως η JavaScript για να στείλει ένα αίτημα HTTP ή HTTPS σε ένα Διακομιστή διαδικτύου και να φορτώσει τα δεδομένα απάντησης του διακομιστή πίσω στη γλώσσα. Από τη στιγμή που τα δεδομένα βρεθούν μέσα στη γλώσσα, είναι διαθέσιμα ως ένα XML έγγραφο, εφόσον η απάντηση είχε έγκυρη σήμανση και ως «απλό κείμενο». Τα δεδομένα απάντησης μπορούν να χρησιμοποιηθούν για να ενημερωθεί το περιεχόμενο της σελίδας χωρίς να χρειάζεται επαναφόρτωση του browser.

Το απλό κείμενο ως απάντηση, όπως χρησιμοποιήθηκε και στην παρούσα πτυχιακή εργασία, μπορεί από τον browser να διαμορφωθεί ως JSON (JavaScript Object Notation) και να αξιολογηθεί από την JavaScript για να δημιουργηθεί ένα αντικείμενο δεδομένων. Το αντικείμενο XMLHttpRequest παίζει ίσως τον πιο σημαντικό ρόλο στην διαδικτυακή τεχνική ανάπτυξης AJAX.



Εικόνα 3.1 Το αντικείμενο XMLHttpRequest

JSON Το JSON είναι ακρώνυμο του JavaScript Object Notation και είναι ένα format μορφοποιήσεις δεδομένων σε javascript όπως έχει περιγραφεί μερικώς λίγο πιο πριν. Το format του είναι πολύ απλό και τυποποιημένο. Έχει απλή ιεραρχική μορφή, ανεξάρτητη από την γλώσσα ή την πλατφόρμα στην οποία χρησιμοποιείται.

Είναι ένα πρότυπο κειμένου το οποίο είναι τελείως ανεξάρτητο από γλώσσες προγραμματισμού, αλλά χρησιμοποιεί πρακτικές (conventions) οι οποίες είναι γνωστές στους προγραμματιστές της οικογένειας προγραμματισμού C, συμπεριλαμβανομένων των C, C++, C#, Java, JavaScript, Perl, Python, και

πολλών άλλων. Αυτές οι ιδιότητες κάνουν το JSON μια ιδανική γλώσσα προγραμματισμού ανταλλαγής δεδομένων.

Το JSON προσφέρεται ως εναλλακτική λύση στην XML. Φυσικά δεν μπορεί να αντικαταστήσει την XML 100% διότι δεν υποστηρίζει schema validation, δεν μπορεί από μόνη της να ενημερώσει σχετικά με την κωδικοποίησή της και δεν έχει την έννοια των attributes, αλλά όπου αυτά τα ελαττώματα μπορούν να αγνοηθούν, η αντικατάσταση θα είναι πολύ εύκολη. Όπως και η XML έτσι και το JSON έχει self-documentated format που περιγράφει την δομή των δεδομένων και δεν ασχολείται με την παρουσίασή τους. Επίσης, παραλείπονται πολλές επαναλήψεις κάποιων πληροφοριών που υπάρχουν στην XML.

PHP PHP (ακρωνύμιο του PHP: Hypertext Preprocessor) είναι μια ευρέως χρησιμοποιούμενη, ανοιχτού κώδικα, γενικού σκοπού scripting γλώσσα προγραμματισμού, η οποία είναι ειδικά κατάλληλη για ανάπτυξη ιστοσελίδων και μπορεί να ενσωματωθεί στην HTML.

Μεγάλο μέρος της σύνταξης του είναι δανεισμένο από την C, Java και την Perl με μια σειρά από μοναδικά PHP-ειδικά χαρακτηριστικά.

Ο στόχος της PHP είναι να επιτρέπει στους web developers να δημιουργούν γρήγορα δυναμικά παραγόμενες σελίδες.

Αυτό που διαχωρίζει την PHP από κάτι σαν client-side Javascript είναι ότι, όταν κάποιος επισκέπτεται την ιστοσελίδα, ο web server επεξεργάζεται τον PHP κώδικα. Βλέπει ποια μέρη θα πρέπει να δείξει στους επισκέπτες (το περιεχόμενο και τις εικόνες) και κρύβει όλα τα υπόλοιπα (τη λειτουργία των αρχείων, τους μαθηματικούς υπολογισμούς, κλπ.). Τότε μεταφράζεται ο PHP κώδικας σε μορφή HTML. Μετά την μετάφραση σε HTML, στέλνει την ιστοσελίδα στον browser του επισκέπτη.

Η PHP διαθέτει ενσωματωμένες εντολές υποστήριξης για ένα μεγάλο αριθμό βάσεων δεδομένων (MySQL Microsoft SQL Server, PostgreSQL και Oracle). Μπορεί να μεταφραστεί και να τρέξει σε οποιοδήποτε λειτουργικό σύστημα (Microsoft Windows, Linux, SD, Solaris, Macintosh OS X, και UNIX servers). Συνεργάζεται χωρίς προβλήματα με πολλούς web servers (Apache, IIS) και μπορεί να επικοινωνήσει με αρκετά πρωτόκολλα (HTTP , POP3, SNMP και LDAP) ώστε να παρέχει τη δυνατότητα ανάγνωσης και εγγραφής σε αρχεία και αποστολής email.

Επίσης, παρόλο που η PHP δεν άρχισε ως αντικειμενοστρεφής (object-oriented) γλώσσα προγραμματισμού, μετά από χρόνια προσπάθειας, προστέθηκαν πολλά αντικειμενοστρεφή χαρακτηριστικά που παρουσιάζονταν στις άλλες γλώσσες προγραμματισμού, όπως:

- Ενθυλάκωση

- Κληρονομικότητα – πολλαπλή κληρονομικότητα
- Πολυμορφισμός
- Κλάσεις (Classes)
- Διεπαφές
- Namespaces
- Υπερφόρτωση μεθόδων.

Η PHP επίσης υποστηρίζει όπως και οι άλλες γλώσσες προγραμματισμού λειτουργίες όπως

- Διαχείριση λαθών.
- Εργασία με αρχεία και το λειτουργικό σύστημα.
- Κανονικές εκφράσεις.
- Διαχείριση συνεδριάσεων (session handling).
- Διαδικτυακές υπηρεσίες ... και άλλα.

PHP κώδικας είναι κλεισμένος σε ειδικά tags αρχής `<?php` και τέλους `?>` που επιτρέπουν τη μετάφραση εντός και εκτός PHP λειτουργίας. Αντί για πολλές εντολές για την εξαγωγή HTML κώδικα (όπως στην C ή Perl), οι PHP σελίδες περιέχουν HTML με ενσωματωμένο κώδικα που κάνει «κάτι».

MySQL Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακής βάσης ανοικτού κώδικα (relational database management system - RDBMS) που χρησιμοποιεί την Structured Query Language (SQL), την πιο γνωστή γλώσσα για την προσθήκη, την πρόσβαση και την επεξεργασία δεδομένων σε μία Βάση Δεδομένων.

Λόγω του ότι είναι ανοικτού κώδικα (open source), οποιοσδήποτε μπορεί να κατεβάσει την MySQL και να την διαμορφώσει βάσει των αναγκών του, σύμφωνα πάντα με την γενική άδεια που υπάρχει. Η MySQL είναι γνωστή κυρίως για την ταχύτητα, την αξιοπιστία, και την ευελιξία που παρέχει.

Η MySQL αυτή τη στιγμή μπορεί να λειτουργήσει σε περιβάλλον Linux, Unix, και Windows και χρησιμοποιείται από τα πιο μεγάλα CMS (Joomla, Wordpress, Drupal κλπ) και όπως και από τεράστια συστήματα που χρησιμοποιούνται καθημερινά (Google, Wikipedia, Facebook, Twitter).

3.2 Περιγραφή της βάσης δεδομένων

Για να καλύψει την λειτουργικότητά της η διαδικτυακή εφαρμογή των πτυχιακών εργασιών και να παρουσιάσει την κατάλληλη πληροφορία διαχειρίζεται δυναμικά δεδομένα. Τα δεδομένα αυτά αποθηκεύονται σε μια βάση δεδομένων με την οποία η εφαρμογή επικοινωνεί. Για το σκοπό αυτό, χρησιμοποιήθηκε το σύστημα

διαχείρισης βάσεων δεδομένων (DBMS) της MySql και αναπτύχθηκε η βάση. Ας δούμε αναλυτικά τους πίνακες της βάσης μας παρακάτω. Στην ενότητα αυτή γίνεται μια αναλυτική περιγραφή των πινάκων της βάσης δεδομένων και των πεδίων που περιλαμβάνει ο καθένας τους. Στη συνέχεια παρατίθεται μια περιγραφή του σχεδιασμού και της λειτουργίας κάθε μιας από τις κύριες οθόνες του συστήματος.

3.3 Βάση δεδομένων και πίνακες

Τα δεδομένα τα οποία καταχωρούν οι χρήστες αποθηκεύονται σε κάποιους πίνακες, οι οποίοι είναι αποθηκευμένοι με τη σειρά τους στη βάση δεδομένων που χρησιμοποιούμε. Αυτοί οι πίνακες είναι οι εξής:

a) Πίνακας **usrTypes**

Περιγραφή: Σε αυτόν τον πίνακα, καταχωρούνται τα id των ρόλων των χρηστών και ο ρόλος του κάθε id.

Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
type	VARCHAR(20)	

Πίνακας 3.1 usrTypes

- Το **id** (Primary Key) είναι ο μοναδικός αριθμός κάθε ρόλου. Για κάθε ρόλο θα αυξάνεται αυτόματα.
- Το **type** είναι το όνομα του κάθε ρόλου.

b) Πίνακας **users**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται όλα τα στοιχεία των χρηστών.

Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
--------------	-------	----------

id	INT	Αυξανόμενος Αριθμός
username	VARCHAR(30)	UNIQUE
password	VARCHAR(64)	
salt	VARCHAR(3)	
type	INT	
email	VARCHAR(64)	
name	VARCHAR(30)	
surname	VARCHAR(30)	
avatarName	CHAR(9)	
activated	BOOLEAN	

Πίνακας 3.2 users

- Το **id** (Primary Key) είναι ο κωδικός αριθμός του κάθε χρήστη. Ο αριθμός αυτός αυξάνεται αυτόματα κάθε φορά που δημιουργείται ένας νέος χρήστης.
- Το **username** είναι το όνομα που επέλεξε ο χρήστης. Πρέπει να είναι μοναδικό στη βάση.
- Το **password** είναι ο κωδικός του χρήστη με encryption. Ο χρήστης γράφει το password το οποίο κρυπτογραφείται με sha256. Μετά δημιουργείται ένα τυχαίο salt με το οποίο κρυπτογραφούμε το ήδη κρυπτογραφημένο password. Αυτό αποθηκεύεται τελικά στο πεδίο.
- Το **salt** είναι η τυχαία ακολουθία που δημιουργείται για την κρυπτογράφηση του password.
- Το **type** (Foreign Key) είναι ξένο κλειδί του πίνακα usrTypes(id).
- Το **email** είναι το email του χρήστη.
- Το **name** είναι το όνομα του χρήστη.
- Το **surname** είναι το επίθετο του χρήστη.
- Το **avatarName** είναι το όνομα του avatar που χρησιμοποιεί η εφαρμογή.

- Το **activated** δείχνει αν είναι ενεργοποιημένος ο χρήστης.

c) Πίνακας **courses**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται τα μαθήματα που είναι προαπαιτούμενα για τις διάφορες πτυχιακές.

Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
courseCode	CHAR(4)	
name	VARCHAR(70)	

Πίνακας 3.3 courses

- Το **id** (Primary Key) είναι ο μοναδικός αριθμός του κάθε μαθήματος.
- Το **courseCode** είναι ο κωδικός που έχει το μάθημα σύμφωνα με τα στοιχεία της σχολής.
- Το **name** είναι ο τίτλος του μαθήματος.

d) Πίνακας **theses**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται τα στοιχεία της κάθε πτυχιακής εργασίας.

Δομή:

Όνομα πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
profId	INT	
title	VARCHAR(100)	
description	TEXT	

isTeamed	BOOLEAN	
totalStudents	INT	DEFAULT'1'
descriptionFile	VARCHAR(60)	
creationDate	DATE	
disabled	BOOLEAN	

Πίνακας 3.4 theses

- Το **id** (Primary Key) είναι ο μοναδικός αριθμός της κάθε πτυχιακής που δημιουργείται. Κάθε φορά που δημιουργείται μια πτυχιακή ο αριθμός αυξάνεται κατά ένα.
- Το **proflid** είναι το id του καθηγητή που δημιουργεί την πτυχιακή και είναι ξένο κλειδί του πίνακα users(id).
- Το **title** είναι ο τίτλος της πτυχιακής εργασίας.
- Το **description** είναι μια σύντομη περιγραφή της πτυχιακής εργασίας.
- Το **isTeamed** είναι το πεδίο που φανερώνει αν η πτυχιακή είναι ομαδική ή όχι.
- Το **totalStudents** είναι ο αριθμός των ατόμων που μπορούν να αναλάβουν την πτυχιακή ως ομάδα. Αυτό το πεδίο έχει default τιμή '1'.
- Το **descriptionFile** είναι η ονομασία του αρχείου που περιέχει όλες τις πληροφορίες σχετικά με την πτυχιακή.
- Το **creationDate** είναι η ημερομηνία που δημιουργήθηκε η πτυχιακή εργασία.
- Το **disabled** είναι το πεδίο που φανερώνει αν η πτυχιακή είναι ενεργή.

e) Πίνακας **prerequisites**

Περιγραφή: Σε αυτόν τον πίνακα είναι καταχωρούνται τα προαπαιτούμενα μαθήματα της κάθε πτυχιακής.

Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
thesisId	INT	
courseId	INT	

Πίνακας 3.5 prerequisites

- Το **id** (Primary Key) είναι ο μοναδικός αριθμός της κάθε εγγραφής που γίνεται για προαπαιτούμενα μαθήματα.
- Το **thesisId** (Foreign Key) είναι το id της πτυχιακής εργασίας που καταχωρείται το προαπαιτούμενο μάθημα. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **courseId** (Foreign Key) είναι το id του μαθήματος που είναι προαπαιτούμενο για την πτυχιακή. Είναι ξένο κλειδί του πίνακα courses(id).

f) Πίνακας assignments

Περιγραφή: Σε αυτόν τον πίνακα καταχωρείται η ανάθεση της κάθε πτυχιακής. Πότε έγινε, ποια είναι και σε ποιον ανατέθηκε.

Δομή:

Όνομα πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
userId	INT	
thesisId	INT	
date	DATE	

Πίνακας 3.6 assignments

- Το **id** (Primary Key) είναι μοναδικός αριθμός για κάθε μια ανάθεση πτυχιακής εργασίας που γίνεται και αυξάνεται αυτόματα.
- Το **userid** (Foreign Key) είναι το id του χρήστη που του ανατίθεται η πτυχιακή εργασία. Είναι ξένο κλειδί του πίνακα users(id).
- Το **thesisId** (Foreign Key) είναι το id της πτυχιακής που ανατίθεται προς υλοποίηση. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **date** είναι η ημερομηνία που γίνεται η ανάθεση η πτυχιακή εργασία.

g) Πίνακας **choices**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται οι 5 επιλογές του φοιτητή για της υποψήφιες πτυχιακές εργασίες.

Δομή:

Όνομα πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
userid	INT	
first	INT	
second	INT	
third	INT	
forth	INT	
fifth	INT	
date	DATE	

Πίνακας 3.7 choices

- Το **id** (Primary Key) είναι μοναδικός αριθμός για κάθε μια αίτηση του φοιτητή και αυξάνεται αυτόματα με κάθε νέα αίτηση.
- Το **userid** (Foreign Key) είναι το id του φοιτητή που κάνει την αίτηση. Είναι ξένο κλειδί του πίνακα users(id).

- Το **first** (Foreign Key) είναι το id της πτυχιακής που έχει ως πρώτη επιλογή ο φοιτητής. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **second** (Foreign Key) είναι το id της πτυχιακής που έχει ως δεύτερη επιλογή ο φοιτητής. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **third** (Foreign Key) είναι το id της πτυχιακής που έχει ως τρίτη επιλογή ο φοιτητής. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **fourth** (Foreign Key) είναι το id της πτυχιακής που έχει ως τέταρτη επιλογή ο φοιτητής. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **fifth** (Foreign Key) είναι το id της πτυχιακής που έχει ως πέμπτη επιλογή ο φοιτητής. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **date** είναι η ημερομηνία που έκανε αίτηση ο φοιτητής για ανάληψη πτυχιακής εργασίας.

h) Πίνακας **presentations**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται οι πληροφορίες σχετικά με τις παρουσιάσεις πτυχιακών εργασιών. Οι εξεταστές, ημερομηνία, ώρα εξέτασης, βαθμοί, αρχείο παρουσίασης κτλ.

Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
thesisId	INT	
examiner1	INT	
examiner2	INT	
date	DATE	
time	TIME	
accepted	BOOLEAN	
grade1	INT	
grade2	INT	

grade3	INT	
presentationFile	VARCHAR(40)	
textFile	VARCHAR(40)	
sourceCodeFile	VARCHAR(40)	

Πίνακας 3.8 presentations

- Το **id** (Primary Key) είναι μοναδικός αριθμός, που αφορά τη κάθε παρουσίαση που δημιουργείται. Αυξάνεται αυτόματα με κάθε νέα παρουσίαση που δημιουργείται.
- Το **thesisId** (Foreign Key) είναι το id της πτυχιακής που θα παρουσιαστεί. Είναι ξένο κλειδί του πίνακα theses(id).
- Το **examiner1** (Foreign Key) είναι το id του πρώτου εξεταστή. Είναι ξένο κλειδί του πίνακα users(id).
- Το **examiner2** (Foreign Key) είναι το id του δεύτερου εξεταστή. Είναι ξένο κλειδί του πίνακα users(id).
- Το **date** είναι η ημερομηνία παρουσίασης που ορίστηκε.
- Το **time** είναι η ώρα παρουσίασης που ορίστηκε.
- Το **accepted** δείχνει αν έχει γίνει η παρουσίαση.
- Το **grade1** είναι ο πρώτος βαθμός που καταχωρήθηκε στη πτυχιακή εργασία.
- Το **grade2** είναι ο δεύτερος βαθμός που καταχωρήθηκε στη πτυχιακή εργασία.
- Το **grade3** είναι ο τρίτος βαθμός που καταχωρήθηκε στη πτυχιακή εργασία.
- Το **presentationFile** είναι το όνομα του αρχείου παρουσίασης (powerPoint).
- Το **textFile** είναι το όνομα του αρχείου του κειμένου της πτυχιακής εργασίας (document).
- Το **sourceCodeFile** είναι το όνομα του αρχείου που περιέχει τον κώδικα της πτυχιακής.

Πίνακας **necessaryDocs**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται όλα τα απαραίτητα σχετικά με την πτυχιακή εργασία έγγραφα.

Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός
description	VARCHAR(50)	
documentName	VARCHAR(40)	

Πίνακας 3.9 necessaryDocs

- Το **id** (Primary Key) είναι μοναδικός αριθμός, που αφορά κάθε ένα έγγραφο που υπάρχει. Αυξάνεται αυτόματα με κάθε νέο έγγραφο που προστίθεται.
- Το **description** είναι η περιγραφή του περιεχομένου του κάθε εγγράφου.
- Το **documentName** είναι το όνομα του αρχείου του κάθε εγγράφου.

i) Πίνακας **semesters**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται οι ημερομηνίες των ενεργών εξαμήνων, καθώς και η περίοδος που γίνεται η ανάθεση των πτυχιακών εργασιών.

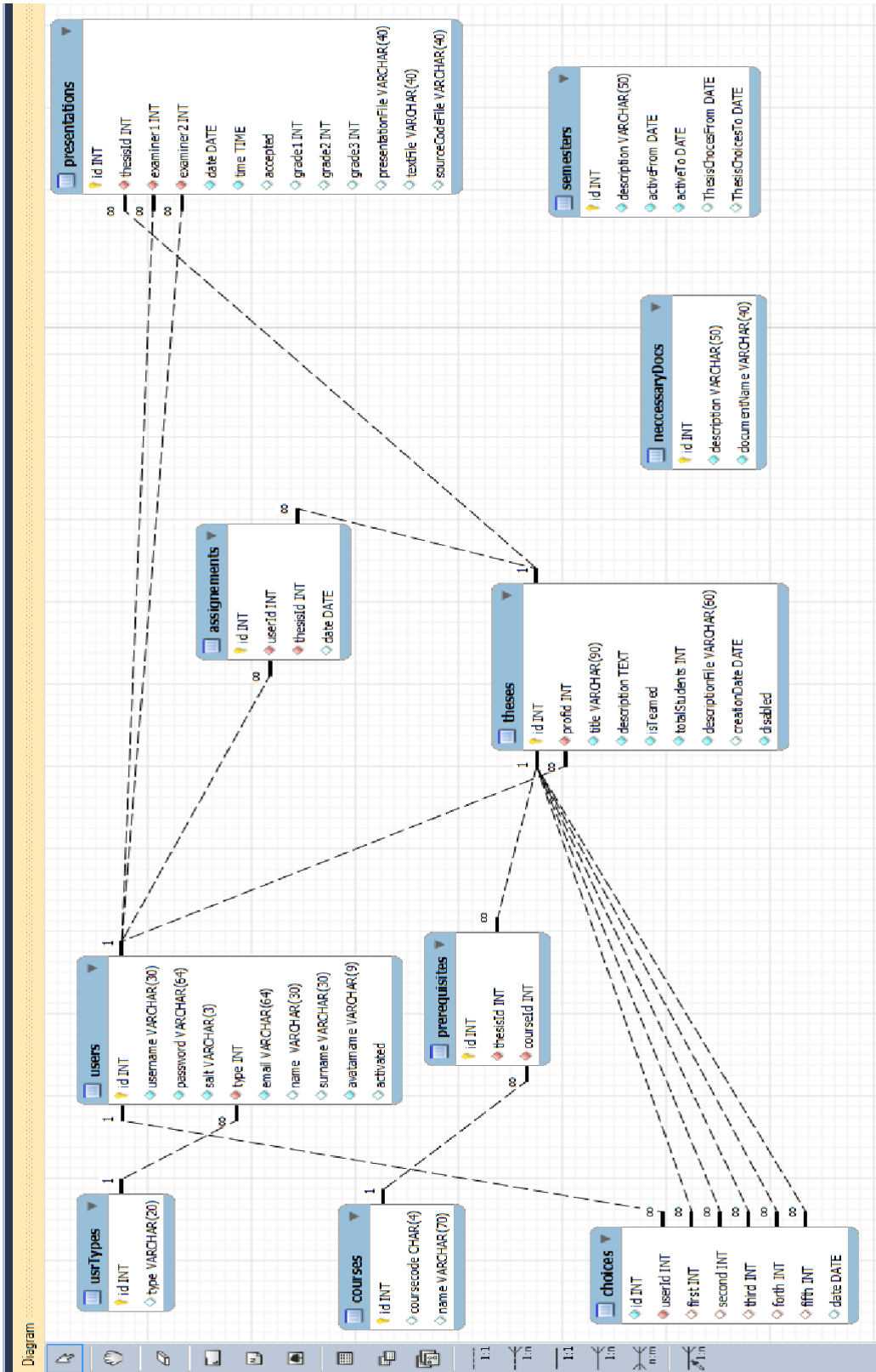
Δομή:

Όνομα Πεδίου	Τύπος	Πρόσθετα
id	INT	Αυξανόμενος Αριθμός

description	VARCHAR(50)	
activeFrom	DATE	
activeTo	DATE	
thesisChoicesFrom	DATE	
thesisChoicesTo	DATE	

Πίνακας 3.10 semesters

- Το **id** (Primary Key) είναι μοναδικός αριθμός, που αφορά το κάθε εξάμηνο. Αυξάνεται αυτόματα με κάθε νέο εξάμηνο που δημιουργείται.
- Το **description** είναι το όνομα του εξαμήνου που δημιουργείται. Π.χ. Εαρινό 2014-2015.
- Το **activeFrom** είναι η ημερομηνία που ξεκινάει το εξάμηνο.
- Το **activeTo** είναι η ημερομηνία που ολοκληρώνεται το εξάμηνο.
- Το **thesisChoicesFrom** είναι η ημερομηνία που ξεκινάνε οι επιλογές των πτυχιακών εργασιών.
- Το **thesisChoicesTo** είναι η ημερομηνία που ολοκληρώνεται η διαδικασία της επιλογής των πτυχιακών εργασιών.



Διάγραμμα Βάσης Δεδομένων

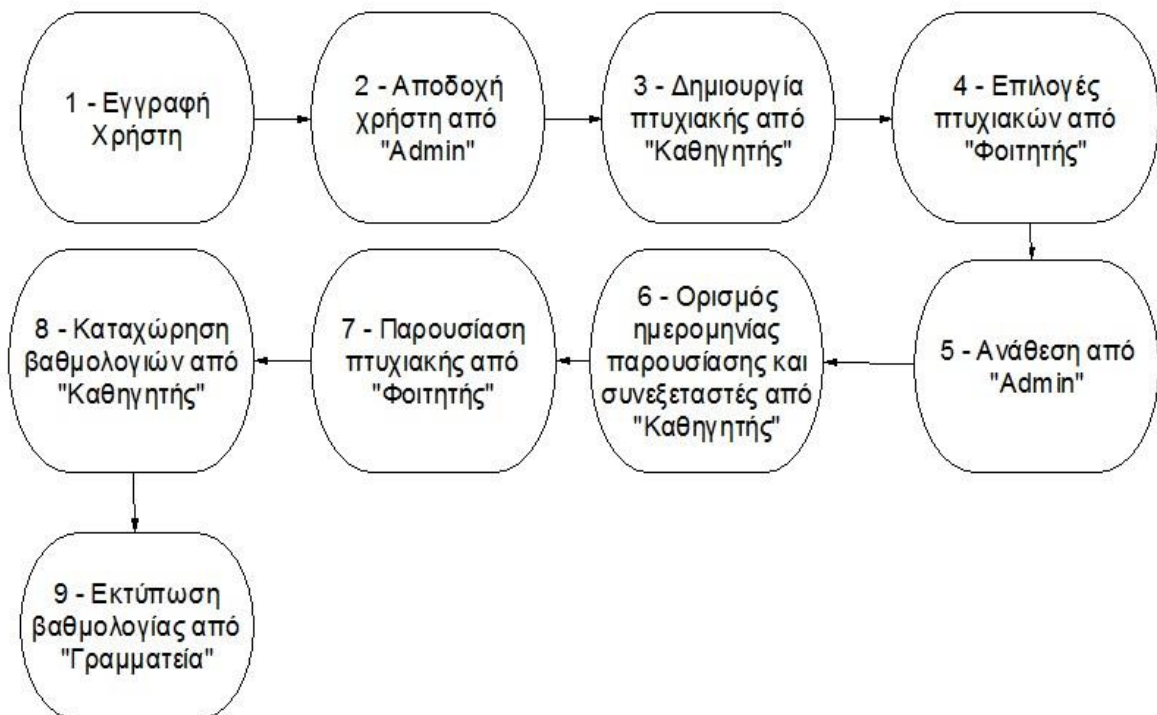
3.4 Η μορφή της εφαρμογής

Για να καληφθεί η λειτουργικότητα της εφαρμογής, όπως αυτή περιγράφηκε παραπάνω σχεδιάστηκαν κατάλληλες σελίδες και τμήματα σελίδων, μεταξύ των οποίων πλοηγείται ο χρήστης αλληλεπιδρώντας με την εφαρμογή ώστε να επιτύχει το επιθυμητό αποτέλεσμα. Η γενική ιδέα για την διεπιφάνεια της εφαρμογής μας συνίσταται στην χρήση συστατικών που τοποθετούνται στις σελίδες της και της προσδίδουν την απαιτούμενη λειτουργικότητα. Συστατικά όπως φόρμες, κουμπιά, μενού επιλογής, πίνακες, γνώριμα στους χρήστες του διαδικτύου, τα οποία δομούνται ακολουθώντας ένα νοητό περίγραμμα (template), ώστε να συνθέσουν μια διεπαφή φιλική στον χρήστη. Οι φόρμες που χρησιμοποιούνται όπου προβλέπεται, αναμένεται είσοδος από τον χρήστη, τα κουμπιά για την αλληλεπίδραση του με την εφαρμογή μιας και επιτρέπουν την πυροδότηση γεγονότων, οι πίνακες για παρουσίαση δεδομένων, ενώ τα μενού επιλογής σε σημεία όπου η είσοδος που αναμένεται από τον χρήστη είναι τυποποιημένη, οπότε του δίνεται η δυνατότητα να επιλέξει παρά να πληκτρολογήσει.

3.5 Η ροή των εργασιών

Σε αυτό το κεφάλαιο θα μιλήσουμε για τη ροή των εργασιών της εφαρμογής για ένα ολοκληρωμένο σενάριο.

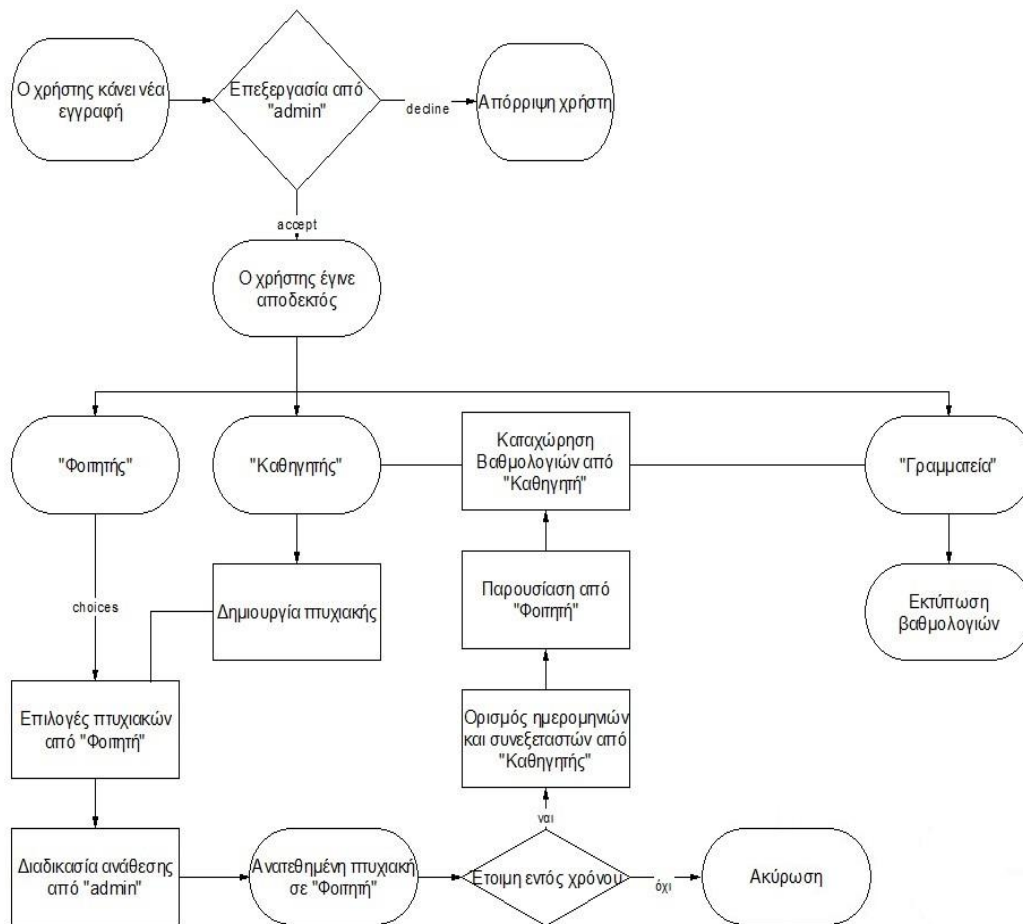
Αρχικά θα αναφερθούμε στα βήματα που ακολουθεί η εφαρμογή. Τα βήματα αυτά φαίνονται στην παρακάτω εικόνα:



Εικόνα 3.2 Τα βήματα της εφαρμογής

Όπως βλέπουμε αρχικά ο χρήστης κάνει νέα εγγραφή. Η αίτηση επεξεργάζεται από τον ρόλο του “Admin” και γίνεται αποδεκτή ή όχι. Ο “Καθηγητής” πρέπει να δημιουργήσει πτυχιακές ώστε να υπάρχουν στο σύστημα για να επιλεγθούν από τους φοιτητές. Μετά ο “Φοιτητής” είναι υπεύθυνος να καταχωρήσει τις επιλογές του. Αυτές οι επιλογές φτάνουν στον “Admin” που είναι ο υπεύθυνος για την ανάθεσή τους. Όταν τελειώσει ο φοιτητής με την πτυχιική ο “Καθηγητής” είναι υπεύθυνος για τον ορισμό ημερομηνιών παρουσίασης και συνεξεταστών. Με το πέρας της παρουσίασης από τον φοιτητή ο “Καθηγητής” είναι πάλι υπεύθυνος για την καταχώρηση των βαθμολογιών. Αυτές οι βαθμολογίες στέλνονται στη γραμματεία που τις εκτυπώνει.

Αναλυτικότερα αυτός ο συσχετισμός των ρόλων φαίνεται από το παρακάτω διάγραμμα:



Εικόνα 3.3 Η ροή των εργασιών

Σε αυτό το διάγραμμα φαίνονται και οι ρόλοι “Φοιτητής”, “Καθηγητής” και “Γραμματεία” και πως συνδέονται μεταξύ τους, από τη δημιουργία της πτυχιακής μέχρι την εκτύπωση της βαθμολογίας της από την γραμματεία.

3.6 Λογική κώδικα

Σε αυτό το κομμάτι θα αναφερθούν λίγα λόγια για τον κώδικα της εφαρμογής. Η εφαρμογή αναπτύχθηκε αρχικά τοπικά στον υπολογιστή. Για την συγγραφή του κώδικα χρησιμοποιήθηκε όπως είπαμε το PHP IDE (Integrated Development Environment) PhpStorm. Για τη δημιουργία και διαχείριση της βάσης χρησιμοποιήθηκε το εργαλείο SQLyog. Κύρια γλώσσα προγραμματισμού ήταν η PHP που χρησιμοποιήθηκε σε συνδυασμό με άλλες γλώσσες όπως SQL, JavaScript, HTML κτλ. Οι αλλαγές στα στοιχεία μέσα στις σελίδες γίνονται ασύγχρονα με τη χρήση της τεχνολογίας JSON (JavaScript Object Notation). Σαν τοπικός server επιλέχθηκε να χρησιμοποιηθεί ο WampServer που δίνει τη δυνατότητα να φιλοξενήσει τις διαδυκτιακές εφαρμογές που χρησιμοποιούν Apache, PHP και SQL. Τώρα θα εξηγήσουμε τη φιλοσοφία κάτω από την οποία λειτουργεί ο κώδικας και πώς συνδέονται όλα μεταξύ τους με ένα συγκεκριμένο παράδειγμα.

Αρχικά έχουμε τις σελίδες που βλέπει ο χρήστης που μερικές είναι οι παρακάτω:

```
addNewCourse.php  
addThesisGrades.php  
showAllCourses.php  
editCourse.php  
addNewCourse.php
```

Σε αυτές τις σελίδες ο χρήστης δημιουργεί κάποια γεγονότα. Για παράδειγμα καθώς ο χρήστης (συγκεκριμένα ο admin) είναι στη σελίδα *addNewCourse.php* και θέλει να προσθέσει ένα μάθημα, δημιουργεί ένα γεγονός και καλείται η function *addCourse()*. Αυτή με τη σειρά της δημιουργεί ένα *XMLHttpRequest* το *xmlhttp = new XMLHttpRequest()*. Με την εντολή *xmlhttp.open("GET", "./addEditRetrivePages/addEditDaveCourse.php?whatToDo=add" + parameters, true)* ορίζουμε τα κατάλληλα ορίσματα για την αίτηση. Έπειτα καλείται η *xmlhttp.send()* η οποία είναι η μέθοδος που θα στείλει την αίτηση στον server. Στον server τρέχει το script που έχουμε ορίσει στην μέθοδο *open()*. Στο συγκεκριμένο παράδειγμα τρέχει το script που βρίσκεται μέσα στο αρχείο *addEditRetrivePages/addEditSaveCourse.php*. Επίσης με την αίτηση έχουμε στείλει και σαν action το *whatToDo=add*. Έτσι ο server τρέχει το script και με το κατάλληλο action μπαίνει στην κατάλληλη *if()* και προκαλεί τις αλλαγές (στη συγκεκριμένη περίπτωση προσθέτει επιτυχώς ένα μάθημα ή όχι) και κάνει τις απαραίτητες αλλαγές στη βάση, για να καταχωρηθεί το μάθημα, με SQL queries. Η SQL εντολή, αν όλα πάνε καλά και δεν υπάρξουν σφάλματα κατά τον έλεγχο του μαθήματος, είναι η *INSERT INTO `courses`(`courseCode`, `name`) VALUES ('\$courseCode', '\$courseName')* που δημιουργεί στον πίνακα *courses* ένα μάθημα με την αντίστοιχη ονομασία και κωδικό που στέλνουμε.

Αυτή είναι η λογική και η φιλοσοφία κάτω από την οποία δουλεύει όλη η εφαρμογή. Ο χρήστης δημιουργεί requests τα οποία στέλνονται στον server ο οποίος με τη

σειρά του, τρέχει κάποια scripts επιστρέφει responses (JSON αντικείμενα) και πραγματοποιεί αλλαγές στη βάση. Αποσπάσματα κώδικα για το παραπάνω παράδειγμα που αναλύθηκε δείχνουν οι παρακάτω εικόνες.

```

$returnedCoursesArray = array();
$whatToDo = $_GET['whatToDo'];
$courseId = -1;

if($whatToDo == 'add'){ // code to add one single course
    $courseCode = $_GET['courseCode'];
    $courseName = $_GET['courseName'];

    $findIfCourseExistsQuery = "SELECT count(*) as found FROM `courses` WHERE `courseCode` = '" . $courseCode . "'";

    if(!mysql_query($findIfCourseExistsQuery, $conn)){
        die('✘ Υπήρξε κάποιο πρόβλημα κατά την εγγραφή του μαθήματος...</br>' . mysql_error());
    }
    else{
        $queryResult = mysql_query($findIfCourseExistsQuery, $conn);
        $count = mysql_fetch_array($queryResult, 0);

        if($count['found'] != 0){
            echo "✘ Το συγκεκριμένο μάθημα υπάρχει ήδη!";
        }
        else{
            $addQuery = "INSERT INTO `courses` (`courseCode`, `name`) VALUES ('$courseCode', '$courseName')";

            if(!mysql_query($addQuery, $conn)){
                die('✘ Υπήρξε κάποιο πρόβλημα κατά την εγγραφή του μαθήματος...</br>' . mysql_error());
            }
            else{
                echo "✔ Το μάθημα <strong>" . $courseCode . "</strong> προστέθηκε επιτυχώς.";
            }
        }
    }
}

```

Εικόνα 3.4 Κώδικας *addEditSaveCourse.php*


```

function addCourse(){
    if(validateForm()){
        var courseCode = document.forms["addNewCourseForm"]["course_code"].value;
        var courseName = document.forms["addNewCourseForm"]["course_name"].value;

        var parameters = "&courseCode=" + courseCode + "&courseName=" + courseName;

        if (window.XMLHttpRequest){// code for IE7+, Firefox, Chrome, Opera, Safari
            xmlhttp = new XMLHttpRequest();
        }
        else{// code for IE6, IE5
            xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
        }
        xmlhttp.onreadystatechange = function(){
            if(xmlhttp.readyState==4 && xmlhttp.status==200){
                if(xmlhttp.responseText.contains("✖")){
                    alert(xmlhttp.responseText);
                    document.forms["addNewCourseForm"]["course_code"].value = "";
                }
                else{
                    document.getElementById("logger").innerHTML += xmlhttp.responseText + "<br>";
                    document.forms["addNewCourseForm"].reset();
                }
            }
        }
        xmlhttp.open("GET", "./addEditRetrievePages/addEditSaveCourse.php?whatToDo=add" + parameters, true);
        xmlhttp.send();
    }
}

```

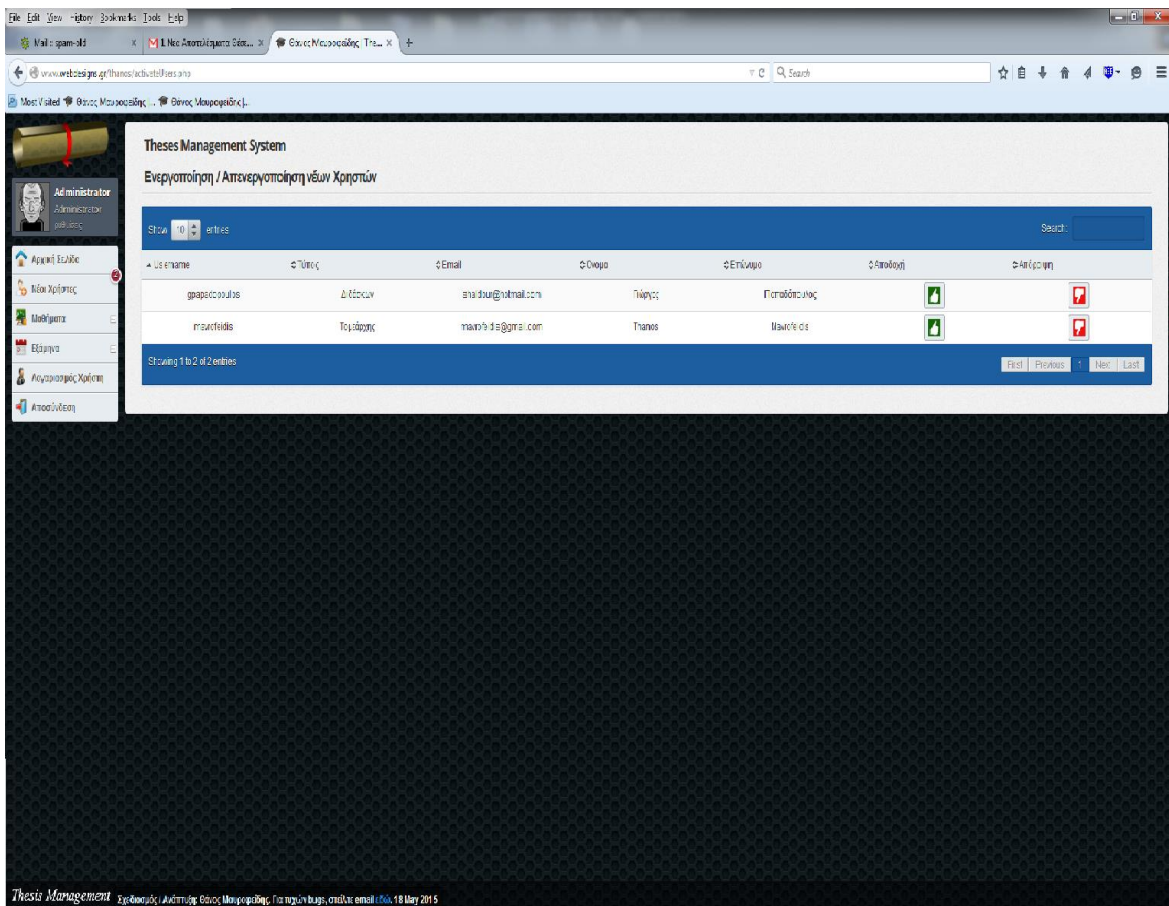
Εικόνα 3.5 Κώδικας *addNewCourse.php*

Σε αυτό το σημείο θα πρέπει να επισημάνουμε ότι για τις ανάγκες της εφαρμογής χρησιμοποιήθηκε το template της Adminica. Το template αγοράστηκε για προσωπική χρήση και ενσωματώθηκε στην πτυχιακή εργασία. Παρέχει διάφορες λειτουργίες που παραμετροποιήθηκαν κατάλληλα για τις ανάγκες της πτυχιακής εργασίας. Περιέχει CSS3 για τη μορφοποίηση, προσφέρει τη δυνατότητα εύκολης ταξινόμησης στους πίνακες και εύκολη επικύρωση στις φόρμες της εφαρμογής. Υπάρχει ο φάκελος include μέσα στον κώδικα της εφαρμογής ο οποίος περιέχει όλα τα κατάλληλα includes για το framework. Είναι συμβατό με πολλούς browsers και χρησιμοποιήθηκε η PHP έκδοση. Από το site της Adminica που είχε πολλά παραδείγματα επιλέχθηκε η μορφοποίηση των διάφορων CSS. Αρχικά στη σελίδα κάνουμε include το includes/core/document_head.php που είναι ένα αρχείο που περιέχει τα βασικά CSS της εφαρμογής.

3.7 Παρουσίαση εφαρμογής

Σε αυτό το κεφάλαιο θα παρουσιαστούν οι βασικές λειτουργίες της εφαρμογής και πώς αυτές αλληλεπιδράνε ώστε να είναι η εφαρμογή λειτουργική. Στην αρχική σελίδα, λοιπόν, παρουσιάζεται φόρμα εισόδου στην υπηρεσία, όπου οι χρήστες πρέπει να εισάγουν το username και το password τους, αν έχουν ήδη λογαριασμό, αλλιώς είτε θα έχουν τον ρόλο των επισκεπτών, είτε θα κάνουν εγγραφή και θα περιμένουν το accept ή reject του admin.

Ο χρήστης – επισκέπτης μπορεί να δει τις πτυχιακές εργασίες και τις πληροφορίες σχετικά με αυτές. Αν θέλει να εγγραφεί ως νέος χρήστης συμπληρώνει τη φόρμα εγγραφής και το αίτημα του φαίνεται στο λογαριασμό του admin που είναι υπεύθυνος για τον χειρισμό των νέων χρηστών. Ο admin από την πλευρά του όταν του έρθει η ενημέρωση ότι έχει νέο χρήστη πρέπει να επιλέξει αν θα τον αποδεχτεί ή θα τον απορρίψει. Η φόρμα η οποία βλέπει ο admin γι αυτόν τον λόγο είναι η παρακάτω.

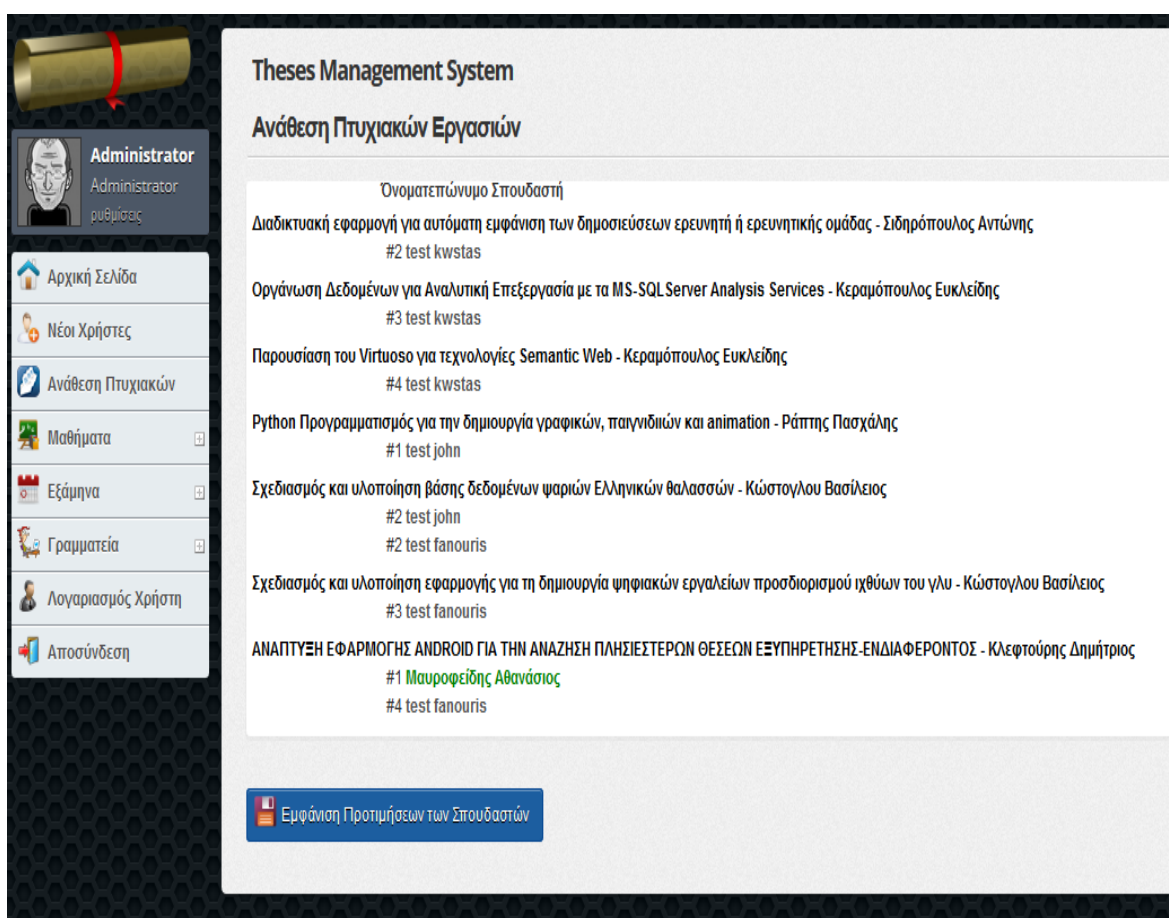


Εικόνα 3.6 Αποδοχή – διαγραφή νέων χρηστών

Αν γίνει αποδεκτός ο χρήστης, τότε καταχωρούνται τα στοιχεία του στη βάση, στέλνεται ενημερωτικό email και μπορεί να περιηγηθεί στην εφαρμογή, ανάλογα με

τον ρόλο που έχει. Σαν επισκέπτης μπορεί να δει τις διάφορες πτυχιακές που έχουν ανατεθεί καθώς και πληροφορίες για αυτές.

Admin: Ο admin πέρα από το να διαχειρίζεται τους νέους χρήστες, είναι υπεύθυνος για να δημιουργεί και να αλλάζει τα μαθήματα, καθώς και για την δημιουργία εξαμήνων και ημερομηνιών για τις δηλώσεις των πτυχιακών εργασιών. Μία άλλη βασική ευθύνη είναι να αναθέτει τις πτυχιακές στους φοιτητές, κάτω από κάποια κριτήρια και σύμφωνα με τις επιλογές του κάθε φοιτητή. Στον admin φαίνονται οι τίτλοι των πτυχιακών και από κάτω τα ονοματεπώνυμά των φοιτητών με τον αριθμό προτίμησης τους. Ο admin επιλέγει τον φοιτητή στον οποίο θα ανατεθεί η πτυχιακή (γίνεται πράσινος) και ανανεώνεται η σελίδα ώστε να σβηστεί το ονοματεπώνυμο του φοιτητή και η πτυχιακή. Η εικόνα είναι η παρακάτω:



The screenshot displays the 'Theses Management System' interface. On the left is a navigation sidebar with a user profile for 'Administrator' and menu items: Αρχική Σελίδα, Νέοι Χρήστες, Ανάθεση Πτυχιακών, Μαθήματα, Εξάμηνα, Γραμματεία, Λογαριασμός Χρήστη, and Αποσύνδεση. The main content area is titled 'Ανάθεση Πτυχιακών Εργασιών' and lists several assignments with their titles and authors. The fourth assignment is highlighted in green, indicating it is assigned to the user. At the bottom, there is a button for 'Εμφάνιση Προτιμήσεων των Σπουδαστών'.

Όνοματεπώνυμο Σπουδαστή
Διαδικτυακή εφαρμογή για αυτόματη εμφάνιση των δημοσιεύσεων ερευνητή ή ερευνητικής ομάδας - Σιδηρόπουλος Αντώνης #2 test kwstas
Οργάνωση Δεδομένων για Αναλυτική Επεξεργασία με τα MS-SQL Server Analysis Services - Κεραμόπουλος Ευκλείδης #3 test kwstas
Παρουσίαση του Virtuoso για τεχνολογίες Semantic Web - Κεραμόπουλος Ευκλείδης #4 test kwstas
Python Προγραμματισμός για την δημιουργία γραφικών, παιχνιδιών και animation - Ράπτης Πασχάλης #1 test john
Σχεδιασμός και υλοποίηση βάσης δεδομένων ψαριών Ελληνικών θαλασσών - Κώστογλου Βασίλειος #2 test john
Σχεδιασμός και υλοποίηση εφαρμογής για τη δημιουργία ψηφιακών εργαλείων προσδιορισμού ιχθύων του γλυ - Κώστογλου Βασίλειος #3 test fanouris
ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ANDROID ΓΙΑ ΤΗΝ ΑΝΑΖΗΤΗ ΠΛΗΣΙΕΣΤΕΡΩΝ ΘΕΣΕΩΝ ΕΞΥΠΗΡΕΤΗΣΗΣ-ΕΝΔΙΑΦΕΡΟΝΤΟΣ - Κλεφτούρης Δημήτριος #1 Μαυροφείδης Αθανάσιος #4 test fanouris

Εικόνα 3.7 Η σελίδα της ανάθεσης εργασιών

Διδάσκων: Ο διδάσκων μπορεί να δημιουργεί τις πτυχιακές του μέσα από την εφαρμογή, να τις διαχειρίζεται, να ορίζει ημερομηνίες παρουσίασης, να καταχωρεί τις βαθμολογίες και να καταχωρεί τα αρχεία της πτυχιακής. Ενδεικτικά βλέπουμε την σελίδα του καθηγητή όταν είναι έτοιμος να ορίσει ημερομηνία παρουσίασης, καθώς και τους συνεξεταστές καθηγητές.

Theses Management System

Καταχώρηση ημερομηνίας παρουσίασης και των επιτηρητών σε κάποια ήδη ανατεθειμένη πτυχιακή εργασία.

Τίτλος	Ημ/νία Δημιουργίας	Είναι Ομαδική?	Αρχείο Πτυχιακής	Λεπτομέρειες
Διαδικτυακή εφαρμογή για αυτόματη εμφάνιση των δημοσιεύσεων ερευνητή ή ερευνητικής ομάδας	06 Aug 2013	Ναι	Πτυχιακι_Sidiropoulos_webpublications.pdf	

Τίτλος	Διαδικτυακή εφαρμογή για αυτόματη εμφάνιση των δημοσιεύσεων ερευνητή ή ερευνητικής ομάδας				
Ημ/νία Δημιουργίας	06 Aug 2013	Ημ/νία Αναθέσεως	26 May 2015	Αρχείο Πτυχιακής	Πτυχιακι_Sidiropoulos_webpublications.pdf
Είναι Ομαδική?	Ναι	Ανατέθηκε στον/στους	test kwstas		
Επιβλέπων Καθηγητής	Απόνης Σιδιρόπουλος	2ος Εξεταστής	<input type="text" value="Επιλέξτε κάποιον καθηγητή"/> <input type="text" value="Επιλέξτε κάποιον καθηγητή"/> Κεραμάπουλος Ευκλείδης Κλεφτούρης Δημήτριος Κώστογλου Βασίλειος Παπαδόπουλος Γιώργος Ράπτης Πασχάλης	3ος Εξεταστής	Επιλέξτε κάποιον καθηγητή
Ώρα Παρουσίασης	undefined	Ημ/νία Παρουσίασης			

Αποθήκευση Αλλαγών Επαναφορά

Εικόνα 3.8 Καταχώρηση ημερομηνιών και ορισμό συνεξεταστών

Ενεργές πτυχιακές είναι οι πτυχιακές που έχουν ανατεθεί αλλά δεν έχουν ολοκληρωθεί. Άρα μια πτυχιακή που δημιουργείται από τον καθηγητή ανατίθεται από τον admin (βλέπει τις αιτήσεις των φοιτητών) στους φοιτητές και τις υπόλοιπες φάσεις (ημερομηνία εξέτασης, συνεξεταστές, βαθμολογίες κτλ.) τις διαχειρίζεται ο καθηγητής.

Φοιτητής: Ο φοιτητής από την πλευρά του βλέπει τις διαθέσιμες πτυχιακές και πραγματοποιεί τις 5 επιλογές του. Αυτές οι επιλογές στέλνονται στον admin που κάνει με τη σειρά του τις αναθέσεις σύμφωνα με τα κριτήρια του τμήματος. Η σελίδα που ο φοιτητής καταχωρεί τις επιλογές φαίνεται παρακάτω. Ο φοιτητής βλέπει στο κάτω μέρος τις επιλογές του και μπορεί να αλλάξει τη προτεραιότητα με τα βελάκια πάνω και κάτω, καθώς και να την αφαιρέσει από τις επιλογές του. Όλα τα παραπάνω γίνονται πριν πατήσει καταχώρηση προτιμήσεων και σταλούν στον admin για ανάθεση.

The screenshot shows a web browser window with the URL www.webdesigns.gr/thanos/saveThesesChoices.php. The page title is "Theses Management System" and the main heading is "Επιλογή Πτυχιακών Εργασιών (maximum 5)".

On the left, there is a sidebar with a user profile for "Φοιτητής Αθανάσιος Μαυροφειδης" and navigation links for "Αρχική Σελίδα", "Φοιτητής", "Λογαριασμός Χρήστη", and "Αποσύνδεση".

The main content area is titled "Διαθέσιμες Πτυχιακές Εργασίες για το συγκεκριμένο εξάμηνο" and lists several thesis topics with their descriptions:

- Σιδερόπουλος Αντώνης**
 - 3D Γλυπτική και δημιουργία χαρακτήρων για ταινίες, παιχνίδια, και κινούμενες εικόνες
 - Μετατροπή συστήματος αρχείων από fat σε ext
 - Διαδικτυακή εφαρμογή για αυτόματη εμφάνιση των δημοσιεύσεων ερευνητή ή ερευνητικής ομάδας
 - Διαδικτυακά εργαλεία για διαχείριση πτυχιακών εργασιών του τμήματος πληροφορικής
- Ράπτης Πασχάλης**
 - Δημιουργία παιχνιδιού τύπου first person shooter με GML 8
 - Πληροφοριακό Σύστημα εταιρείας TAXI
 - Python Προγραμματισμός για την δημιουργία γραφικών, παιχνιδιών και animation
- Κεραμοπούλος Ευκλείδης**
 - Παρουσίαση του Viteoσο για τεχνολογίες Semantic Web
 - Προβολή ατομικού προγράμματος μαθημάτων σε Android
- Κασιφτόρης Δημήτριος**
 - ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ΥΠΟΒΟΛΗΣ ΠΑΡΑΓΓΕΛΙΑΣ ΣΕ BeachBar ΜΕ ΧΡΗΣΗ ΤΗΣ ΠΛΑΤΦΟΡΜΑΣ ANDROID
 - ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ ANDROID ΓΙΑ ΤΗΝ ΑΝΑΖΗΤΗΣΗ ΠΛΗΣΙΕΣΤΕΡΩΝ ΘΕΣΕΩΝ ΕΞΥΠΗΡΕΤΗΣΗΣ-ΕΝΔΙΑΦΕΡΟΝΤΟΣ
- Κώστογλου Βασίλειος**
 - Σχεδιασμός και υλοποίηση εφαρμογής για τη δημιουργία ψηφιακών εργαλείων προσδιορισμού κθύν του γλυ
 - Σχεδιασμός και υλοποίηση βάσης δεδομένων ψαριών Ελληνικών θαλασσών
 - Ανάπτυξη εργαλείων για την ποσοτική εκτίμηση της έντασης του χρώματος στο δέρμα ψαριών με βάση το χρ

Below the list, there is a section "Επιλεγμένες Πτυχιακές Εργασίες κατά σειρά Προτίμησης" with a list of two selected items:

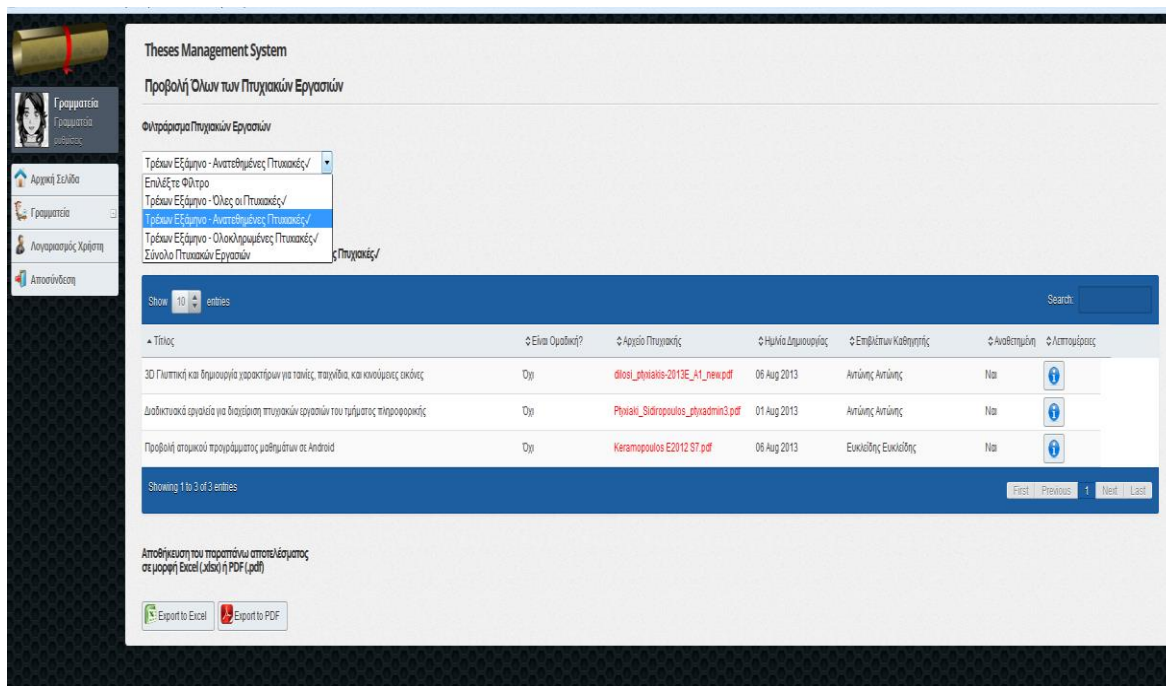
1. Οργάνωση Δεδομένων για Αναλυτική Επεξεργασία με τα MS-SQLServer Analysis Services
2. Οργάνωση Δεδομένων για Αναλυτική Επεξεργασία με χρήση Oracle

A "Καταχώρηση Προτιμήσεων" button is located at the bottom of the list.

At the bottom of the page, there is a footer: "Thesis Management Σχεδιασμός / Ανάπτυξη: Θάνος Μαυροφειδης. Για τυχών bugs, στείλτε email edw. 19 May 2015".

Εικόνα 3.9 Επιλογή πτυχιακών εργασιών

Γραμματεία: Η γραμματεία είναι υπεύθυνη για τον χειρισμό των βοηθητικών εγγράφων που αφορούν τον θεσμό των πτυχιακών εργασιών. Μπορεί να ανεβάζει αρχεία και να τα διαχειρίζεται. Εκτός αυτής της λειτουργίας μπορεί να φιλτράρει τις πτυχιακές και να εμφανίζει τα αποτελέσματα σε μορφή excel για εκτύπωση. Εικόνα αυτής της λειτουργίας βλέπουμε παρακάτω.



Εικόνα 3.10 Φιλτράρισμα πτυχιακών εργασιών από τη γραμματεία

Αυτές ήταν οι βασικές λειτουργίες της εφαρμογής με κάποιες εικόνες ώστε να παρουσιαστεί το interface που συναντά ο εκάστοτε χρήστης. Έτσι σε μία σύνοψη των παραπάνω που αφορά τη ροή των διαδικασιών είναι η εξής: Ο admin ενημερώνει το τρέχον εξάμηνο και ορίζει τις ημερομηνίες των δηλώσεων πτυχιακών. Οι φοιτητές από τη πλευρά τους βλέπουν τις διαθέσιμες πτυχιακές (που έχουν δημιουργηθεί από τους καθηγητές) και πραγματοποιούν τις επιλογές τους. Ο admin τους κάνει ανάθεση τις πτυχιακές όταν λήξει ο χρόνος δηλώσεως τους. Οι φοιτητές κατεβάζουν τα απαραίτητα έγγραφα από τη γραμματεία τα συμπληρώνουν και τα πηγαίνουν για να υπογραφούν. Όταν οι φοιτητές ολοκληρώνουν την πτυχιακή τους εργασία σε συνεννόηση με τον επιβλέπων καθηγητή ορίζουν τις ημερομηνίες εξέτασης και τους συνεξεταστές, τις οποίες τις καταχωρεί ο καθηγητής, καθώς και τα αρχεία της πτυχιακής. Όταν ολοκληρωθεί η παρουσίασή της, ο καθηγητής καταχωρεί τη βαθμολογία της στη βάση.

4. ΑΠΟΤΕΛΕΣΜΑΤΑ ΚΑΙ ΠΡΟΤΑΣΕΙΣ

Στόχος της παρούσας εργασίας ήταν να παρουσιαστεί ένα εργαλείο που θα μπορέσει να βοηθήσει στον θεσμό των πτυχιακών εργασιών. Μέσα από την εφαρμογή επιτεύχθηκε βαθύτερη κατανόηση της ρηρ και έγινε χρήση αρκετών δυνατοτήτων της γλώσσας. Αυτή η εργασία δεν είναι άμεσα λειτουργική, έστω και για τη σχολή πληροφορικής, μιας και λείπουν κάποια κομμάτια ώστε να μπορεί να αντικαταστήσει – συγχωνευτεί με το υπάρχον σύστημα. Κομμάτια όπως συνδεσιμότητας με τον αετό και χρήση αλγορίθμου ανάθεσης πτυχιακών για παράδειγμα. Επικεντρωθήκαμε στην κάλυψη των λειτουργικών προδιαγραφών, οπότε και υπάρχει χώρος για βελτιώσεις και προσθήκες. Άλλωστε τέτοιου είδους εφαρμογές οφείλουν να εξελίσσονται και να προσαρμόζονται ανάλογα με τις ανάγκες. Πέρα λοιπόν από την εξέλιξη της για εφαρμογή στη σχολή Πληροφορικής, θα μπορούσε να υπάρξει ένα βήμα παραπάνω στην εξέλιξη της ώστε να μπορεί να χρησιμοποιηθεί και από άλλα εκπαιδευτικά ιδρύματα. Έτσι θα υπήρχε ένας ενιαίος χώρος ελέγχου – φιλοξενίας και θα μπορούσε ακόμα και ο απλός ο κόσμος να βλέπει διάφορα θέματα με τις πληροφορίες τους, ώστε να ενημερώνεται για το τι έχει ερευνηθεί και κατ' επέκταση ακόμα και το επίπεδο τις παιδείας μας.

5. ΒΙΒΛΙΟΓΡΑΦΙΑ

Raghu Ramakrishnan, Johannes Gehrke (2002), Συστήματα Διαχείρισης Βάσεων Δεδομένων, Εκδόσεις Τζιόλα

Δρ. Μιχάλης Σαλαμπάσης (2008), Εισαγωγή στον προγραμματισμό διαδικτυακών εφαρμογών

Luke Welling, Laura Thomson (2009), Ανάπτυξη web εφαρμογών με PHP και MySQL, Εκδόσεις Μ. Γκιούρδας

<http://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>

<http://viralpatel.net/blogs/listbox-select-all-move-left-right-up-down-javascript/>

<http://stackoverflow.com/>

<http://www.w3schools.com/>

<http://themeforest.net/item/adminica-the-professional-admin-template/160638>

<http://www.webyog.com>

<http://php.net/manual/en/>

<http://www.jetbrains.com/phpstorm/>

<http://www.wikipedia.org>

<http://www.fpdf.org/>

Πτυχιική εργασία Χαραλάμπους Μάριος «Web application for classroom/calendar Administration based on AJAX»