



ΤΕΧΝΟΛΟΓΙΚΟ
ΕΚΠΑΙΔΕΥΤΙΚΟ
ΙΔΡΥΜΑ
ΤΕΙ ΗΠΕΙΡΟΥ

ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ & ΟΙΚΟΝΟΜΙΑΣ (Σ.Δ.Ο)
ΤΜΗΜΑ ΤΗΛΕΠΛΗΡΟΦΟΡΙΚΗΣ & ΔΙΟΙΚΗΣΗΣ

SCHOOL OF MANAGEMENT AND ECONOMICS
*DEPARTMENT OF COMMUNICATIONS,
INFORMATICS AND MANAGEMENT*

Τ.Ε.Ι. Ηπείρου
Τμήμα Τηλεπληροφορικής & Διοίκησης
Ιωάννης Σ. Γκουτσίδης
On-line Εφαρμογή Βαθμολογίας Διδασκομένων
Εκπαιδευτικών Ιδρυμάτων
“ eGrade ”

Σεπτέμβριος 2004

**ON-LINE ΕΦΑΡΜΟΓΗ ΒΑΘΜΟΛΟΓΙΑΣ
ΔΙΔΑΣΚΟΜΕΝΩΝ ΕΚΠΑΙΔΕΥΤΙΚΩΝ
ΙΔΡΥΜΑΤΩΝ**

“ eGrade ”

Πτυχιακή Εργασία

Μέρος των απαιτήσεων του Τμήματος Τηλεπληροφορικής και Διοίκησης

Ιωάννης Σ. Γκουτσίδης

Σεπτέμβριος 2004

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα εργασία εκπονήθηκε στο τμήμα Τηλεπληροφορικής & Διοίκησης του Τ.Ε.Ι. Ηπείρου.

Η ανάθεση του θέματος και η γενική επίβλεψη πραγματοποιήθηκε από τον επιστημονικό συνεργάτη του τμήματος, Δρα. Νικόλαο Κουρκουμέλη, τον οποίο ευχαριστώ για την καθοδήγηση και την βοήθεια που προσέφερε καθ' όλη τη διάρκεια της συγγραφής αυτής της εργασίας. Ευχαριστώ, επίσης τα μέλη της συμβουλευτικής επιτροπής, τον Αναπλ. Καθηγητή κ. Ευριπίδη Γλαβά και τον Αναπλ. Καθηγητή κ. Νικόλαο Αντωνιάδη για το ενδιαφέρον που επέδειξαν. Ευχαριστώ ακόμη, τα μέλη-διαχειριστές του Κέντρου Ελέγχου Δικτύων του Τ.Ε.Ι. Ηπείρου κ. Δημήτριο Βασιλειάδη, Γεώργιο Ρίζο και Λεωνίδα Τσιαντή, για την υποστήριξη, την οποία μου παρείχαν όσον αφορά την εύρυθμη και απρόσκοπτη λειτουργία της ηλεκτρονικής εφαρμογής βαθμολογίας eGrade στους Web Servers του τμήματος και πιο συγκεκριμένα στην ηλεκτρονική διεύθυνση: <http://myweb.teleinfom.teiep.gr/ggkoutsidis/>.

Τέλος, να ευχαριστήσω τους υπεύθυνους της εταιρίας παραγωγής λογισμικού Cardisoft A.E. κ. Κωνσταντίνο Μυλωνά (Τεχνικό Εφαρμογών) και Γεώργιο Παπαζίδη (Τμήμα Πωλήσεων), για όλες τις τεχνικές πληροφορίες που μου παρείχαν, όσον αφορά τις παραπλήσιες με την εν λόγω, εφαρμογές, τις οποίες διαθέτει και διανέμει η εταιρία τους στην ελληνική αγορά πληροφορικής, συμπεριλαμβανομένου και του Τ.Ε.Ι. Ηπείρου.

ΔΗΛΩΣΗ ΠΕΡΙ ΛΟΓΟΚΛΟΠΗΣ

Όλες οι προτάσεις, οι οποίες παρουσιάζονται σε αυτό το κείμενο και οι οποίες ανήκουν σε άλλους αναγνωρίζονται από τα εισαγωγικά και υπάρχει η σαφής δήλωση του συγγραφέα. Τα υπόλοιπα γραφόμενα είναι επινόηση του γράφοντος, ο οποίος φέρει και την καθολική ευθύνη για αυτό το κείμενο και δηλώνω υπεύθυνα ότι δεν υπάρχει λογοκλοπή για το κείμενο αυτό.

Όνοματεπώνυμο: Ιωάννης Σ. Γκουτσίδης

Υπογραφή:

Ημερομηνία: Σεπτέμβριος 2004

ΠΡΟΛΟΓΟΣ

Μια ηλεκτρονική εφαρμογή βαθμολόγησης παρέχει τη δυνατότητα σε ένα διδάσκοντα να ενημερώνεται σχετικά με τους διδασκόμενους και τα μαθήματά τους. Παρέχεται μια πληθώρα διαφορετικών μορφών εμφάνισης των αποτελεσμάτων.

Η γενική μορφή μιας τέτοιας εφαρμογής δε διαφέρει πολύ από αυτές των παραδοσιακών-χειρόγραφων μεθόδων, μιας και ουσιαστικά ενοποιούνται πληροφορίες σχετικά με τους διδασκόμενους και τα μαθήματά τους. Χρησιμοποιείται για την καταγραφή των αποτελεσμάτων στις εξετάσεις κάθε διδασκόμενου και για την παρακολούθηση των συνολικών τους επιδόσεων. Επιπλέον τα αποτελέσματα του εκάστοτε διδασκόμενου, στο σύνολο των μαθημάτων ή μεμονωμένα, υπολογίζονται και παρουσιάζονται αυτόματα.

Η εφαρμογή βαθμολόγησης eGrade επιτρέπει την παρουσίαση των αποτελεσμάτων εξετάσεων δια μέσω του Internet και είναι διαθέσιμη στους διδασκόμενους και το διδακτικό προσωπικό του εκάστοτε εκπαιδευτικού ιδρύματος.

Στόχος της εφαρμογής είναι η εξυπηρέτηση των εργαζομένων-φοιτητών, καθώς επίσης και η αποσυμφόρηση των γραμματειών των Τμημάτων και του Ιδρύματος από τηλεφωνήματα σχετικά με τη βαθμολογία. Σε καμία περίπτωση δεν υποκαθιστά την επίσημη βαθμολογία της γραμματείας, είναι μόνον για ενημερωτικούς σκοπούς.

Το απόρρητο της βαθμολογίας διασφαλίζεται αφενός με την πιστοποίηση του διδάσκοντα μέσω κωδικού πρόσβασης και αφετέρου με την πιστοποίηση του φοιτητή. Έτσι ο κάθε διδασκόμενος έχει πρόσβαση μόνο στις δικές του βαθμολογίες.

Η χρήση της εφαρμογής κρίνεται απλή, καθώς πραγματοποιείται μόνο με τη χρήση μιας εφαρμογής πλοήγησης (Web Browser) και τη βοήθεια των σχετικών εντολών επιλογής από αντίστοιχα μενού.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1. Εισαγωγή	1
1.1. Λόγοι Χρήσης Εφαρμογής Βαθμολόγησης	1
2. Γενικές Αρχές	4
2.1. Βασική Αρχιτεκτονική On-line Βάσεων Δεδομένων	4
2.2. Αποθήκευση & Ανάκτηση Δεδομένων.....	6
2.3. Βασικές Έννοιες Βάσεων Δεδομένων	8
2.4. Τεχνικές Σχεδιασμού Βάσεων Δεδομένων	9
2.5. Αρχιτεκτονική Διαδικτυακών Βάσεων Δεδομένων	11
3. Η Εφαρμογή Βαθμολόγησης eGrade	13
3.1. Λειτουργία Της Εφαρμογής	13
3.2. Η Βάση Δεδομένων	15
3.3. Τα Αρχεία PHP.....	18
3.4. Βήμα προς Βήμα - Διαχειριστής	33
3.5. Βήμα προς Βήμα - Διδάσκοντας	38
3.6. Βήμα προς Βήμα - Διδασκόμενος.....	46
Επίλογος	52
Παράρτημα	54
1. Κώδικας της Εφαρμογής.....	54
Βιβλιογραφία	80
Ημερολόγιο Πορείας της Εργασίας	81

1. Εισαγωγή

1.1. Λόγοι Χρήσης Εφαρμογής Βαθμολόγησης

Μερικοί καλοί λόγοι για τη χρήση μιας εφαρμογής βαθμολόγησης είναι οι εξής:

1. Εύκολη διαχείριση των πληροφοριών-δεδομένων

Η αναφορά των βαθμολογιών των φοιτητών δεν είναι το μόνο πράγμα που μπορεί να κάνει μια εφαρμογή διαχείρισης βαθμολογίας. Το βασικό πλεονέκτημα των ηλεκτρονικών εφαρμογών διαχείρισης βαθμολογίας είναι ότι με την ορθή καταχώρηση των βαθμολογιών πραγματοποιούνται διάφορες λειτουργίες με σχεδόν μηδενικό έργο. Ο χρήστης μπορεί με μεγάλη ευκολία να υπολογίσει τους μέσους όρους σε επίπεδο τμημάτων, να συντάξει λίστες μαθημάτων που δεν έχουν εξεταστεί, να δημιουργήσει ομάδες φοιτητών οι οποίες επιδέχονται επιπλέον παρακολούθησης σε συγκεκριμένα μαθήματα, και να ενημερώσει αυτόματα μέσους όρους και σύνολα την ίδια στιγμή όπου καταχωρούνται νέα δεδομένα ή ενημερώνονται ήδη υπάρχοντα. Είναι πολύ ευκολότερο να διαπιστωθεί που υπάρχει η ανάγκη της επαναδιδασκαλίας/επιπλέον παρακολούθησης όταν γίνεται χρήση τέτοιων εφαρμογών, οι οποίες βοηθούν στην εξακρίβωση των αναγκών. Οι βαθμολογίες καθώς επίσης και οι διάφορες σημειώσεις πρέπει μόνο να καταχωρηθούν μία φορά, αλλά μπορούν άμεσα να ανακτηθούν κάτω από μια ποικιλία διαφορετικών μορφοποιήσεων.

2. Καλύτερος καταμερισμός του χρόνου

Αυτού του είδους οι εργασίες απαιτούν σαφώς μεγαλύτερο χρόνο για την περάτωσή τους όταν πραγματοποιούνται με τις κλασικές χειρόγραφες μεθόδους. Ακόμη και με το γεγονός ότι ο χρόνος για την εισαγωγή των δεδομένων είναι σχεδόν ο ίδιος είτε γίνεται χειρόγραφα είτε σε κάποια εφαρμογή υπολογιστή, η πραγματική μείωση χρόνου διαφαίνεται όταν παρουσιαστεί η ανάγκη για το χειρισμό των δεδομένων και τη δημιουργία αναφορών για αυτά. Το 1993 οι Vockell και Fiore (βλ. <http://search.epnet.com/direct.asp?an=9705041248&db=tfh>) απέδειξαν ότι χρειάστηκαν 74 λεπτά για την προετοιμασία μιας σειράς αναφορών προόδου

και μιας χειρόγραφης στατιστικής ανάλυσης για μια τάξη 30 διδασκομένων. Για την ίδια εργασία απαιτήθηκαν μόνο 10 λεπτά με τη βοήθεια ενός ηλεκτρονικού υπολογιστή. Οι χρόνοι συμπεριελάμβαναν την καταχώρηση των βαθμολογιών, αλλά όχι την αρχική διαμόρφωση των καταλόγων βαθμολόγησης. Από τότε, υπολογιστές και λογισμικό έγιναν ακόμη πιο αποδοτικοί και πολύπλοκοι, σε αντίθεση με τις συμβατικές μεθόδους του χαρτιού. Φυσικό επόμενο αποτελεί το γεγονός ότι σήμερα οι ίδιες εργασίες πραγματοποιούνται σαφώς πιο αποτελεσματικά και πιο γρήγορα.

3. Επαγγελματική εμφάνιση

Οι αναφορές που δημιουργούνται με τη χρήση ηλεκτρονικών εφαρμογών βαθμολόγησης είναι σαφείς στη μορφοποίησή τους και παρουσιάζουν τα δεδομένα με επαγγελματική εμφάνιση. Συχνά η αξία της λεπτομέρειας είναι μεγαλύτερη σε σχέση με αυτή που μπορεί να παρουσιαστεί σε μια χειρόγραφη αναφορά. Η επαγγελματική αναφορά των ηλεκτρονικά παραγόμενων αναφορών δίνει σε κηδεμόνες και διδάσκοντες μια πιο οργανωμένη και συγκεντρωτική εικόνα της κατάστασης.

4. Η κινητικότητα και οι επιδόσεις των διδασκομένων αυξάνονται

Οι Rockman, Pershing και Ware (βλ. <http://rockman.com/projects/cfet/PPandE92.pdf>) αναφέρουν ότι σε τμήματα στα οποία οι διδάσκοντες χρησιμοποίησαν ηλεκτρονικές εφαρμογές βαθμολόγησης με σκοπό τη συχνή ενημέρωση των διδασκομένων αναφορικά με τις επιδόσεις τους, οι τελευταίοι παρουσίασαν αύξηση στις επιδόσεις και τη γενικότερη κινητικότητά τους. Από τη στιγμή που γίνουν κατανοητές οι βασικές αρχές λειτουργίας μιας εφαρμογής βαθμολόγησης, είναι πρακτικά απλή η διαδικασία δημιουργίας μιας ακριβούς αναφοράς, η οποία περιγράφει τις επιδόσεις των διδασκομένων. Με μια τόσο άμεση ενημέρωση, οι διδασκόμενοι είναι σε θέση να εστιάσουν τις προσπάθειές τους εκεί ακριβώς που απαιτείται.

5. Μείωση του φόρτου εργασίας

Οι εφαρμογές βαθμολόγησης παρέχουν ενημερωμένες αναφορές για οποιονδήποτε διδασκόμενο. Αυτό πραγματοποιείται πλέον με μεγάλη ευκολία και σε πολύ λίγο χρόνο. Επιπλέον υπάρχει η πιθανότητα μετακίνησης ενός διδασκόμενου, κάποιος διδάσκοντας να χρειάζεται τέτοιου είδους πληροφορίες για τις ανάγκες ενός συνεδρίου, ή κάποιος κηδεμόνας να θέλει

να ενημερωθεί για την πρόοδο του διδασκομένου. Με τη χρήση μιας ηλεκτρονικής εφαρμογής βαθμολόγησης, η όλη διαδικασία είναι υπόθεση μερικών «κλικ» του ποντικιού.

6. Τεκμηρίωση των εργασιακών επιδόσεων

Η τεκμηρίωση αποτελεί ένα από τα σημαντικότερα επαγγελματικά καθήκοντα εκ μέρους του διδάσκοντα και απαιτεί ιδιαίτερα επίπονη και λεπτομερειακή εργασία.

Απαιτείται η αρχειοθέτηση των επιδόσεων των διδασκομένων, των επαφών με τους κηδεμόνες και των στρατηγικών επεμβάσεων στη διαδικασία της διδασκαλίας. Με άλλα λόγια πρέπει να αρχειοθετείται και να τεκμηριώνεται κάθε πτυχή μιας εργασίας και να ακολουθεί σύγκριση των αποτελεσμάτων με τα καθορισμένα πρότυπα.

Η χρήση ηλεκτρονικών εφαρμογών βαθμολόγησης διευκολύνει τη διαδικασία της τεκμηρίωσης. Είναι εύκολη η παροχή τεκμηρίωσης υπό τη μορφή αναφορών προόδου, διορθωτικών λιστών, λιστών με μαθήματα που δεν έχουν εξεταστεί, σημειώσεων κ.λπ. Είναι τέλος εύκολο για τους κηδεμόνες να ενημερωθούν για την πρόοδο των διδασκομένων, μιας και μπορούν να δημιουργηθούν αναφορές για τις εβδομαδιαίες επιδόσεις τους και οι αναφορές αυτές να συνοδεύονται από αποδεικτικό ανάγνωσης, το οποίο θα υπογράφεται από τον κηδεμόνα.

Πρακτικά μια ηλεκτρονική εφαρμογή βαθμολόγησης αποδεικνύεται «αυτό-τεκμηριούμενη».

Οι ηλεκτρονικές εφαρμογές βαθμολόγησης παρέχουν μια δυναμική μέθοδο οργάνωσης και άμεσης ανάκτησης εκπαιδευτικών δεδομένων. Καλά γραμμένες εφαρμογές είναι σχεδιασμένες ώστε να απλοποιούν τις χρονοβόρες διαδικασίες, οι οποίες υπεισέρχονται αρνητικά στην όλη διαδικασία της διδασκαλίας.

2. Γενικές Αρχές

2.1. Βασική Αρχιτεκτονική On-line Βάσεων Δεδομένων

Στην πιο βασική του μορφή το Internet λειτουργεί βάση της αρχιτεκτονικής client/server. Με άλλα λόγια ένας κεντρικός εξυπηρετητής (Server) και μια εφαρμογή client είναι υπεύθυνοι για μια σειρά από συνεχείς επεξεργασίες στοιχείων. Αυτό διαφέρει από μια τυπική εφαρμογή που λειτουργεί ανεξαρτήτως σύνδεσης.

- **O Client**

Οι εφαρμογές που δημιουργούνται με τη MySQL και την PHP κάνουν χρήση μιας απλής εφαρμογής client: του Web Browser. Αυτή βέβαια δεν είναι η μοναδική (κύρια) πιθανότητα για εφαρμογές που εκτελούνται στο Internet (Internet Applications). Η πρωταρχική γλώσσα των browsers είναι η HTML (HyperText Markup Language). Η HTML παρέχει ένα πλήθος ετικετών (tags), οι οποίες περιγράφουν τον τρόπο εμφάνισης των περιεχομένων μιας σελίδας web. Η χρήση scripting γλωσσών (όπως ASP & PHP) επιτυγχάνουμε την παρουσίαση δυναμικού περιεχομένου που προσαρμόζεται στις ανάγκες του χρήστη.

- **O Server**

Όλος σχεδόν ο φόρτος επεξεργασίας σε μια web application λαμβάνει χώρα στον server. Μια συγκεκριμένη εφαρμογή, ο web server, είναι υπεύθυνος για την επικοινωνία με τον web browser. Ένας web server σχεσιακών βάσεων δεδομένων είναι υπεύθυνος για την αποθήκευση οποιασδήποτε πληροφορίας απαιτείται από την εφαρμογή. Τελικά, απαιτείται μια γλώσσα για την επικοινωνία του web server με τον database server, η οποία θα χρησιμοποιηθεί επίσης για προγραμματιστικές εργασίες στις πληροφορίες που έρχονται προς και από τον web server.

- **Το Λειτουργικό Σύστημα**

Τα πλέον ενδεδειγμένα λειτουργικά συστήματα για διαδικτυακές εφαρμογές (Web Applications), είναι τα Windows 2000/2003 Server και Unix/Linux. Αυτά καλύπτουν περισσότερο από το 90% των web servers ανά τον κόσμο.

Κατά τη διάρκεια ανάπτυξης της παρούσας υλοποίησης έγινε χρήση ενός Microsoft Windows 2003 Server Enterprise Edition. Η επιλογή έγινε με βάση

τη διαθεσιμότητα της βασικής πλατφόρμας και του μεριδίου της αγοράς εργασίας.

- **O Web Server**

Ο web server είναι μια υπηρεσία η οποία, αναμένει αιτήσεις από χρήστες του Web, στέλνει απαντήσεις σε αυτές και εξυπηρετεί τις web σελίδες για τις οποίες είναι υπεύθυνος. Στην πραγματικότητα, η κατάσταση είναι πιο πολύπλοκη λόγω της «24/7» φύσης του Web, όπου η σταθερότητα του web server αποτελεί θέμα μείζονος σημασίας.

Δύο web servers κατακλύζουν σήμερα την αγορά: οι Microsoft IIS και Apache HTTP Server.

- *Microsoft Internet Information Server*

Ο IIS είναι στενά συνδεδεμένος με το περιβάλλον των Windows και είναι το βασικό στοιχείο για την τεχνολογία σελίδων web ASP (Active Server Pages). Η PHP στην έκδοση 4.x ενοποιείται αποδοτικά με τον IIS.

- *Apache HTTP Server*

Ο Apache είναι ο πιο διαδεδομένος web server. Βρίσκεται κάτω από την GNU Public άδεια χρήσης (βλ. <http://www.gnu.org/>). Στην ανάπτυξη του παρούσας υλοποίησης έγινε χρήση του Apache HTTP Server 2.0.49 για πλατφόρμα Windows. Η επιλογή βασίστηκε στην καλύτερη ενοποίηση του με τη PHP.

- **Η Γλώσσα Προγραμματισμού**

Η PHP συγκαταλέγεται στην κατηγορία των scripting γλωσσών που δρουν ως «Middleware». Αυτές οι γλώσσες συνεργάζονται στενά με τον web server για τη μεταγλώττιση των αιτήσεων από το World Wide Web, την επεξεργασία τους, την αλληλεπίδρασή τους με άλλες εφαρμογές που εκτελούνται στον web server για την εκπλήρωση των αιτήσεων και τέλος για να υποδείξουν στον web server το ακριβές περιεχόμενο που θα αποσταλεί στον web browser του client.

Η γλώσσα PHP κάνει χρήση της μηχανής Zend (βλ. <http://www.zend.com/>) κάτω από την Q Public (βλ. <http://www.opensource.org/licenses/gtpl.php>) άδεια χρήσης. Η βασική αυτή μηχανή scripting παρέχει στη γλώσσα,

απόδοση, αξιοπιστία και ένα εύκολο στη χρήση περιβάλλον συγγραφής κώδικα.

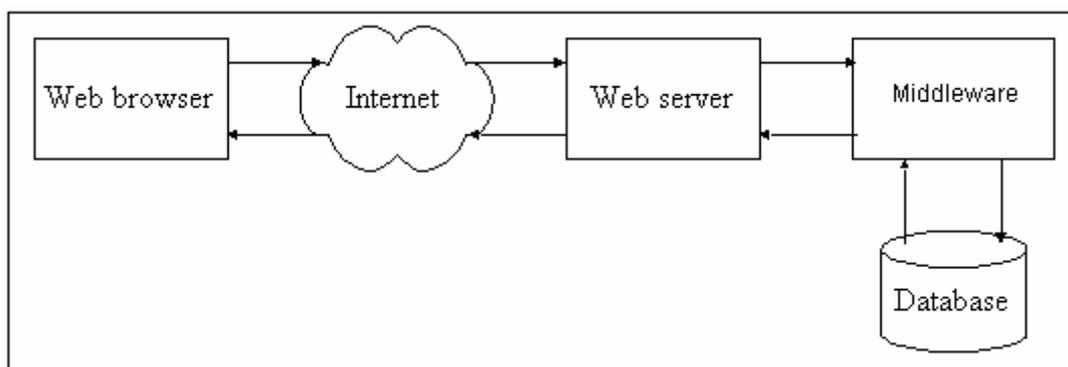
Κατά τη διάρκεια ανάπτυξης την παρούσας υλοποίησης έγινε χρήση της γλώσσας PHP στην έκδοση 4.3.8 για πλατφόρμα Windows, η οποία εγκαταστάθηκε ως module στον Apache HTTP Server.

- **Το Σύστημα Διαχείρισης Βάσεων Δεδομένων**

Τα συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational DataBase Management Systems – RDBMSs), παρέχουν έναν ιδανικό τρόπο για την αποθήκευση και την επεξεργασία πολύπλοκων πληροφοριών. Η πλειοψηφία των βάσεων δεδομένων κάνει χρήση της γλώσσας SQL (Structured Query Language).

Εμπορικά συστήματα είναι τα Oracle, Sybase, Microsoft SQL Server κ.α. Υπάρχουν όμως και «open-source» συστήματα όπως τα Postgres, MySQL και Borland Interbase.

Κατά την υλοποίηση της παρούσας εφαρμογής χρησιμοποιήθηκε το πακέτο MySQL Server 4.0.18 για πλατφόρμα Windows.



Σχήμα 2.1.1. Γραφική αναπαράσταση διαδικτυακών εφαρμογών

2.2. Αποθήκευση & Ανάκτηση Δεδομένων

Η χρήση μιας βάσης δεδομένων ενδείκνυται σε σχέση με ένα αρχείο κειμένου (Flat File) διότι:

- Όταν ένα αρχείο μεγαλώνει σε μέγεθος, γίνεται πιο αργό στην επεξεργασία.

- Η αναζήτηση μιας εγγραφής/ομάδας εγγραφών σε ένα αρχείο είναι δύσκολη. Αν οι εγγραφές είναι ταξινομημένες, είναι δυνατή η χρήση μιας μεθόδου δυαδικής αναζήτησης (Binary Search) από κοινού με ένα αρχείο καθορισμένου πλάτους στην αναζήτηση σε ένα βασικό πεδίο. Αν είναι επιθυμητή η εύρεση εγγραφών βάση κάποιου προτύπου (για παράδειγμα, η εύρεση όλων των πελατών με πόλη διαμονής την Άρτα), θα πρέπει να γίνει ανάγνωση όλων των εγγραφών και έλεγχος σε καθεμιά ξεχωριστά.
- Οι συναλλαγές με ταυτόχρονη πρόσβαση μπορούν να γίνουν προβληματικές. Κατά τη διάρκεια κλειδώματος του αρχείου μπορεί να δημιουργηθεί καθυστέρηση (bottle-neck). Με αρκετή κίνηση (traffic) στον ιστοχώρο (web site), ένα μεγάλο πλήθος χρηστών μπορεί να περιμένει εωσότου το αρχείο ξεκλειδώσει για να κάνουν οποιαδήποτε εργασία. Η αναμονή αυτή είναι ανεπιθύμητη.
- Τα αρχεία αυτού του είδους είναι σειριακά (sequential). Αυτό σημαίνει ότι κατά την εισαγωγή/διαγραφή εγγραφών που βρίσκονται στη μέση του αρχείου με τυχαία προσπέλαση (random access), υπάρχουν δυσκολίες- πρέπει να ολοκληρωθεί η μεταφορά όλου του αρχείου στη μνήμη, να εφαρμοστούν οι αλλαγές και να αποθηκευτεί όλο το αρχείο ξανά πίσω στο δίσκο. Με ένα μεγάλο μεγέθους αρχείο, αυτό γίνεται ιδιαίτερα αργό και δύσκολο.
- Πέρα από τους περιορισμούς που προσφέρονται από τα δικαιώματα των αρχείων, δεν υπάρχει κάποιος εύκολος τρόπος εκχώρησης διαφορετικών επιπέδων πρόσβασης στα δεδομένα.

Τα συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMSs) έρχονται να λύσουν τα προαναφερθέντα προβλήματα των απλών αρχείων προσφέροντας:

- Πιο γρήγορη πρόσβαση στα δεδομένα.
- Πιο εύκολες επερωτήσεις (queries) για την ανάκτηση ομάδων δεδομένων, οι οποίες καλύπτουν συγκεκριμένα κριτήρια.
- Ενσωματωμένους μηχανισμούς για το χειρισμό ταυτόχρονων συνδέσεων.
- Μη σειριακή προσπέλαση σε δεδομένα.
- Ενσωματωμένα συστήματα διαχείρισης δικαιωμάτων. Ο MySQL Server διαθέτει ιδιαίτερες δυνατότητες στον τομέα αυτό (βλ. <http://dev.mysql.com/doc/mysql/en/index.html>).

2.3. Βασικές Έννοιες Βάσεων Δεδομένων

- **Πίνακες (Tables)**

Οι σχεσιακές βάσεις δεδομένων στηρίζονται σε πίνακες. Ένας πίνακας διαθέτει όνομα, ένα πλήθος στηλών, η καθεμιά από τις οποίες αντιπροσωπεύει ένα διαφορετικό στοιχείο πληροφορίας, και ένα πλήθος γραμμών, το οποίο αντιπροσωπεύει τις εγγραφές.

- **Στήλες (Columns)**

Κάθε στήλη σε έναν πίνακα έχει διαφορετικό όνομα και περιέχει διαφορετικά στοιχεία. Κάθε στήλη έχει έναν τύπο δεδομένων. Οι στήλες καλούνται πεδία ή/και ιδιότητες.

- **Γραμμές (Rows)**

Κάθε γραμμή σε έναν πίνακα αντιπροσωπεύει μια διαφορετική εγγραφή. Λόγω της συνοπτικής μορφοποίησής τους, όλες διαθέτουν τα ίδια πεδία (fields). Οι γραμμές καλούνται και εγγραφές.

- **Τιμές (Values)**

Κάθε εγγραφή αποτελείται από μια ομάδα διαφορετικών τιμών, που αντιστοιχούν στα πεδία. Κάθε τιμή πρέπει να διαθέτει του τύπο δεδομένων που έχει οριστεί στο πεδίο της.

- **Κλειδιά (Keys)**

Πρέπει να υπάρχει ένας τρόπος, ώστε κάθε εγγραφή να αναγνωρίζεται μοναδικά. Ένας αρκετά διαδεδομένος τρόπος για να επιτευχθεί αυτό είναι να δημιουργηθεί ένα επιπλέον πεδίο στον πίνακα, το οποίο θα διαθέτει ξεχωριστή (μοναδική) τιμή για κάθε εγγραφή (π.χ. ο αριθμός τραπεζικού λογαριασμού για κάθε πελάτη είναι διαφορετικός). Το πεδίο ενός πίνακα, το οποίο χαρακτηρίζει μοναδικά κάθε εγγραφή ονομάζεται *Πρωτεύον Κλειδί (Primary Key)* του πίνακα. Ένα κλειδί μπορεί να αποτελείται από πολλά πεδία, τα οποία παρόλα αυτά συχνά δεν εγγυώνται τη μοναδικότητα κάθε εγγραφής.

Οι βάσεις δεδομένων συνήθως αποτελούνται από πολλούς πίνακες και χρησιμοποιούν ένα κλειδί ως αναφορά από έναν πίνακα σε κάποιον άλλο.

- **Σχήματα (Schemas)**

Η πλήρης ομάδα από τα σχέδια των πινάκων μιας βάσης δεδομένων ονομάζεται σχήμα της βάσης δεδομένων. Ένα σχήμα απεικονίζει τους πίνακες με τα πεδία τους, τους τύπους δεδομένων των πεδίων και εμφανίζει το πρωτεύον κλειδί, καθώς επίσης και όσα ξένα κλειδιά τυχόν υπάρχουν. Το σχήμα δεν περιλαμβάνει δεδομένα, αλλά μπορεί να εισαχθούν για λόγους κατανόησης. Το σχήμα μπορεί να παρουσιαστεί είτε με ένα διάγραμμα οντοτήτων-συσχετίσεων (Entity Relationship Diagram-ER Diagram), είτε σε μια φόρμα κειμένου.

- **Σχέσεις (Relationships)**

Τα κλειδιά αναπαριστούν μια σχέση ανάμεσα σε δύο πίνακες. Στις σχεσιακές βάσεις δεδομένων υπάρχουν τρεις τύποι σχέσεων. Είναι ταξινομημένοι βάση του πλήθους των αντικειμένων που υπάρχουν σε κάθε πλευρά της σχέσης και είναι οι εξής:

- Ένα-προς-ένα (1-1): Όταν υπάρχει ένα αντικείμενο εκατέρωθεν της σχέσης.
- Ένα-προς-πολλά (1-N): Όταν μία εγγραφή του ενός πίνακα συνδέεται με πολλές εγγραφές του άλλου. Σε αυτές τις σχέσεις ο πίνακας που περιλαμβάνει τις πολλές εγγραφές θα διαθέτει ένα ξένο κλειδί προς τον πίνακα με τη μία εγγραφή.
- Πολλά-προς-πολλά (N-N): Όταν πολλές εγγραφές ενός πίνακα συνδέονται με πολλές εγγραφές ενός άλλου.

2.4. Τεχνικές Σχεδιασμού Βάσεων Δεδομένων

- **Κανονικοποίηση (Normalization)**

Η διαδικασία με την οποία αποφεύγονται οι δυσλειτουργίες προσθήκης-μεταβολής-διαγραφής (Insert-Modify-Delete Anomalies) σε μία βάση δεδομένων ονομάζεται κανονικοποίηση.

Η κανονικοποίηση περιλαμβάνει τις διάφορες «Κανονικές Μορφές» (Normal Forms). Συνήθως χρησιμοποιούνται οι α', β' και γ' κανονικές μορφές, αλλά η διαδικασία μπορεί να προχωρήσει βαθύτερα με ακόμα δύο κανονικές μορφές. Οι τρεις πρώτες κανονικές μορφές ονομάζονται κανονική μορφή κατά «Boyce/Codd», ενώ οι άλλες δύο κατά «Domain/Key». Συνήθως αν μία βάση

δεδομένων φτάσει στην τρίτη κανονική μορφή θεωρείται, σε γενικές γραμμές, λειτουργική. Αναλυτικότερα ισχύουν τα παρακάτω:

○ 1^η Κανονική Μορφή:

Τα δεδομένα πρέπει να είναι σε μορφή πίνακα και να συμφωνούν με τα εξής κριτήρια:

§ Κάθε πεδίο πρέπει να περιέχει μια συγκεκριμένη (atomic) τιμή. Πρέπει δηλ. να υπάρχει μία τιμή σε κάθε πεδίο. Στο πεδίο δεν μπορούν να υπάρχουν πίνακες ή/και γενικά οποιαδήποτε τεχνική παρουσίασης πολλαπλών τιμών.

§ Κάθε πεδίο διαθέτει μοναδικό όνομα.

§ Ο πίνακας πρέπει να διαθέτει μια ομάδα τιμών, η οποία χαρακτηρίζει μοναδικά κάθε εγγραφή (αυτό είναι γνωστό και ως πρωτεύον κλειδί του πίνακα).

§ Δύο εγγραφές δεν μπορεί να είναι πανομοιότυπες.

§ Δεν επιτρέπονται συνεχόμενες ομάδες δεδομένων.

○ 2^η Κανονική Μορφή:

Για να την προσαρμογή ενός πίνακα σε αυτή τη μορφή πρέπει να αφαιρεθούν οι εγγραφές που είναι μερικώς εξαρτώμενες από πρωτεύοντα κλειδιά πολλών πεδίων στον πίνακά τους. Η 2^η Κανονική Μορφή δεν μπορεί να εφαρμοστεί σε πίνακες που διαθέτουν πρωτεύοντα κλειδιά με ένα πεδίο

○ 3^η Κανονική Μορφή:

Ένας πίνακας που βρίσκεται σε αυτή τη μορφή δεν πρέπει να διαθέτει πεδία, τα οποία δεν αντιπροσωπεύουν άμεσα το πρωτεύον κλειδί του. Με αυτό τον τρόπο διασφαλίζονται οι περιπτώσεις επαναλαμβανόμενων δεδομένων.

• Αποφυγή αποθήκευσης συνεχόμενων ίδιων δεδομένων, διότι δημιουργείται:

○ Κατασπατάληση αποθηκευτικού χώρου.

○ Δυσλειτουργίες ενημέρωσης όπως:

§ Δυσλειτουργία προσθήκης εγγραφών

§ Δυσλειτουργία μεταβολής δεδομένων

§ Δυσλειτουργία διαγραφής εγγραφών

• Χρήση μοναδικών τιμών στα πεδία των πινάκων (Atomic Column Values)

Σε κάθε πεδίο κάθε εγγραφής αποθηκεύεται ένα μόνον αντικείμενο.

- Επιλογή των πραγματικά μοναδικών πεδίων ως πρωτεύοντα κλειδιά.
- Συνυπολογισμός των επερωτήσεων που θα γίνουν στη βάση δεδομένων.
- Αποφυγή δημιουργίας πεδίων, τα περισσότερα από τα οποία θα μείνουν κενά.

2.5. Αρχιτεκτονική Διαδικτυακών Βάσεων Δεδομένων

- **Web Server**

- Απαιτείται ένας σύνδεσμος επικοινωνίας μεταξύ των δύο σταθμών.
- Ένας web browser (client) αποστέλλει μία αίτηση (request) προς τον web server (server).
- Ο server στέλνει πίσω μία απάντηση (response).

Το σενάριο αυτό καλύπτει ικανοποιητικά συναλλαγές μεταξύ στατικών ιστοσελίδων.

- **Web Database Server**

- Ο web browser ενός χρήστη αποστέλλει μια αίτηση HTTP για μια συγκεκριμένη ιστοσελίδα (π.χ. συμπληρώνοντας μια φόρμα με αποδέκτη το αρχείο A).
- Ο web server λαμβάνει την αίτηση για το αρχείο A, το ανακτά και το περνά στη μηχανή της PHP (PHP Engine) για επεξεργασία.
- Η PHP Engine ξεκινά την ανάλυση του script. Εντός του script υπάρχει κάποια εντολή, με την οποία πραγματοποιείται σύνδεση με τη βάση δεδομένων και εκτελείται κάποιο query. Η PHP ανοίγει μια σύνδεση με τον MySQL server και αποστέλλει το κατάλληλο query.
- Ο MySQL server λαμβάνει το query και το επεξεργάζεται, αποστέλλοντας τα αποτελέσματα πίσω στην PHP Engine.
- Η PHP Engine τερματίζει την εκτέλεση του script, το οποίο συνήθως περιλαμβάνει και μορφοποίηση των αποτελεσμάτων σε HTML. Έπειτα επιστρέφει το HTML αποτέλεσμα στον web server.
- Ο web server στέλνει το HTML πίσω στον web browser, όπου ο χρήστης ένα αναγνώσιμο response για το αρχικό του request.

Οι υπηρεσίες Web Server, PHP και MySQL Server συνήθως εκτελούνται στον ίδιο υπολογιστή αν και για λόγους ασφάλειας και αξιοπιστίας ο MySQL Server μπορεί να

εκτελείται σε διαφορετικό. Πάντως σε επίπεδο ανάπτυξης η τεχνική/φιλοσοφία παραμένει ίδια ή διαφοροποιείται ελάχιστα, σε σχέση πάντα με την πρώτη περίπτωση.

3. Η Εφαρμογή Βαθμολόγησης eGrade

3.1. Λειτουργία Της Εφαρμογής

Η ηλεκτρονική εφαρμογή βαθμολόγησης eGrade, στο σύνολό της, πλαισιώνεται από τρεις διαφορετικές οντότητες.

Οι οντότητες αυτές αντιπροσωπεύουν κατ' ουσία τα διαφορετικά επίπεδα χρηστών, τα οποία μπορούν να αλληλεπιδράσουν με την εφαρμογή. Πρόκειται για τα εξής επίπεδα:

1. Διδασκόμενος

Στην κατηγορία αυτή ανήκουν όλοι οι διδασκόμενοι του εκάστοτε εκπαιδευτικού ιδρύματος, οι οποίοι έχουν ολοκληρώσει επιτυχώς τη διαδικασία εγγραφής τους στο σύστημα. Ο διδασκόμενος επιλέγει ένα μοναδικό και αναντικατάστατο «Όνομα Χρήστη», και μέσω της διεύθυνσης e-mail που δηλώνει κατά την εγγραφή του, λαμβάνει από το σύστημα έναν «Κωδικό Πρόσβασης». Για τη σύνδεση του διδασκόμενου στο σύστημα απαιτούνται οι δύο προαναφερθείσες πληροφορίες. Μετά την επιτυχή σύνδεσή του, ο διδασκόμενος μπορεί να μεταβάλει προσωπικές του πληροφορίες (π.χ. Όνομα, Επώνυμο, Διεύθυνση e-mail κ.λπ.).

Βασικό μέλημά του πλέον είναι η καταχώρησή του στα διδασκόμενα μαθήματα, για τα οποία θέλει να ενημερώνεται για τη βαθμολογία του. Λόγω του ήδη υπάρχοντος κύκλου σπουδών η επιλογή των προς καταχώρηση μαθημάτων είναι ανεξάρτητη από το τυπικό εξάμηνο, στο οποίο φοιτεί ο διδασκόμενος, ο οποίος φέρει την πλήρη ευθύνη των μαθημάτων που θα καταχωρήσει. Άπαξ και ολοκληρωθεί η διαδικασία καταχώρησης των μαθημάτων από έναν διδασκόμενο, δεν υπάρχει η δυνατότητα διόρθωσης ή/και διαγραφής των καταχωρήσεων αυτών. Την ευθύνη για τέτοιου είδους λειτουργίες φέρει η ομάδα «Διαχειριστών», η οποία θα περιγραφεί παρακάτω.

2. Διδάσκοντες

Στην εν λόγω κατηγορία χρηστών εντάσσεται το εκπαιδευτικό προσωπικό του εκάστοτε εκπαιδευτικού ιδρύματος.

Η διαδικασία εγγραφής ενός διδάσκοντα διαφέρει ελάχιστα από αυτή του διδασκόμενου, μιας και σε αυτή την περίπτωση αποστέλλεται αυτόματα στο

σύστημα μια επιπλέον αίτηση για την ένταξη του συγκεκριμένου χρήστη στην κατηγορία των διδασκόντων. Μόλις επιβεβαιωθεί η ταυτότητα του διδάσκοντος, ο τελευταίος μπορεί να συνδεθεί στο σύστημα με το «Όνομα Χρήστη» που έχει επιλέξει και τον «Κωδικό Πρόσβασης» που του έχει αποσταλεί.

Ο χρήστης μπορεί να μεταβάλλει προσωπικά του στοιχεία με όμοιο τρόπο, όπως αυτός του διδασκόμενου.

Κύρια αρμοδιότητα του διδάσκοντα είναι η δημιουργία των μαθημάτων, τα οποία διδάσκει. Σε αυτά τα μαθήματα ο διδάσκοντας, έχει πλήρη πρόσβαση, υπό την έννοια ότι μπορεί ακόμα και να διαγράψει ένα μάθημά του, αν αυτό κριθεί απαραίτητο.

Μετά την ολοκλήρωση της δήλωσης-καταχώρησης των διδασκόμενων μαθημάτων, ο διδάσκοντας είναι σε θέση να καταχωρήσει τις αντίστοιχες βαθμολογίες για τους διδασκόμενους, οι οποίοι έχουν δηλώσει τα εκάστοτε μαθήματα. Υπάρχουν δύο τρόποι αποστολής της βαθμολογίας και είναι οι εξής:

i. Φόρμα

Με αυτή τη μέθοδο στην οθόνη του διδάσκοντα εμφανίζονται όλοι οι διδασκόμενοι, οι οποίοι έχουν δηλώσει το εν λόγω μάθημα, είτε για πρώτη φορά, είτε έχουν βαθμολογία μικρότερη του 5.00. Ο διδάσκοντας συμπληρώνει τις αντίστοιχες βαθμολογίες, οι οποίες αν συνεχία αποστέλλονται στο σύστημα.

ii. Αρχείο Excel

Αποτελεί πλέον κοινή πρακτική πολλών διδασκόντων να διατηρούν ένα υποτυπώδες βαθμολόγιο με τα ονοματεπώνυμα και τους αριθμούς μητρώου των διδασκόμενων σε ένα απλό «Φύλλο Εργασίας» της εφαρμογής Microsoft Excel. Ο διδάσκοντας πλέον, αφού συμπληρώσει πλήρως το εν λόγω αρχείο και επιλέξει από την εφαρμογή eGrade το μάθημα στο οποίο αυτό αντιστοιχεί, μπορεί να αποστείλει αυτούσιο το αρχείο ηλεκτρονικά. Η επεξεργασία των απεσταλμένων δεδομένων επαφίεται στο σύστημα, το οποίο είναι σε θέση να διαχωρίσει τις διάφορες εγγραφές και να κάνει την ορθή καταχώρηση των αντίστοιχων αποτελεσμάτων για τον εκάστοτε διδασκόμενο.

3. Διαχειριστές

Πρόκειται για την ομάδα χρηστών που αποτελεί τη γενική αρχή λειτουργίας και χρήσης του συστήματος.

Στη δικαιοδοσία της υπάγονται όλες οι ειδικές λειτουργίες συντήρησης και ελέγχου του συστήματος. Οι χρήστες της ομάδας αυτής μπορούν να δημιουργήσουν τα διάφορα «Τμήματα» για κάθε Εκπαιδευτικό Ίδρυμα. Επίσης έχουν δικαιοδοσία να παρεμβαίνουν σε οποιαδήποτε μεταβολή κριθεί απαραίτητη, όπως π.χ. στην αλλαγή των καταχωρημένων μαθημάτων ενός διδασκόμενου, μιας και ο τελευταίος δεν διαθέτει ανάλογα δικαιώματα.

Είναι επιφορτισμένοι με τη διαδικασία πιστοποίησης της ταυτότητας ενός διδάσκοντα, ώστε αυτός να εισαχθεί στην αντίστοιχη ομάδα, όπου λαμβάνει και τα απαραίτητα δικαιώματα.

Η διαδικασία εγγραφής κάποιου στην εν λόγω ομάδα είναι αποκομμένη από το σύστημα eGrade, καθαρά για λόγους ασφαλείας. Κατά την πρώτη εκκίνηση του συστήματος υπάρχει ένας εξ' ορισμού χρήστης σε αυτή την ομάδα με «Όνομα Χρήστη» Admin, και αυτός αποφασίζει για το ποιοι επιπλέον χρήστες μπορούν να εισαχθούν στην εν λόγω ομάδα.

Οι χρήστες της ομάδας «Διαχειριστές» δεν εμφανίζονται σε καμία λειτουργία του συστήματος (π.χ. στην καταχώρηση βαθμολογίας), δεν μπορούν να δηλώσουν μαθήματα, μιας και δεν είναι διδασκόμενοι, και φυσικά δεν μπορούν να εκτελέσουν λειτουργίες του τύπου «Προβολή Βαθμολογίας».

Ο ρόλος τους είναι καθαρά συντονιστικός και συμβουλευτικός για την εύρυθμη και απρόσκοπτη λειτουργία του όλου συστήματος.

3.2. Η Βάση Δεδομένων

Η λειτουργία της εφαρμογής eGrade βασίζεται στην ύπαρξη ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων. Το προϊόν που επελέγη είναι το MySQL Server, στην εμπορική έκδοση 4.0.18.

Η βάση δεδομένων που έχει δημιουργηθεί στο συγκεκριμένο σύστημα διαχείρισης περιλαμβάνει τους εξής πίνακες:

1. Users

Πρόκειται για τον πίνακα, στον οποίο καταχωρούνται τα στοιχεία όλων των χρηστών που μπορούν να συνδεθούν στο σύστημα. Διατηρούνται οι εξής πληροφορίες:

- Κωδικός χρήστη (user_id)
- Αριθμός μητρώου χρήστη, μόνο αν είναι διδασκόμενος (user_registernumber)
- Όνομα χρήστη (user_username)
- Κωδικός πρόσβασης χρήστη (user_password)
- Ομάδα χρήστη (user_securitylevel)
 - i. 1, αν είναι διδασκόμενος
 - ii. 2, αν είναι διδάσκοντας
 - iii. 3, αν είναι διαχειριστής
- Πραγματικό όνομα χρήστη (user_firstname)
- Επώνυμο χρήστη (user_lastname)
- Πατρώνυμο χρήστη (user_fathername)
- Διεύθυνση κατοικίας χρήστη (user_address)
- Πόλη κατοικίας χρήστη (user_city)
- Ταχ. Κώδικας χρήστη (user_postalcode)
- Τηλέφωνο χρήστη (user_phone)
- E-mail διεύθυνση χρήστη (user_email)
- Παρατηρήσεις χρήστη, για τυχόν επιπλέον σημειώσεις (user_notes)

2. Departments

Εδώ αποθηκεύονται στοιχεία σχετικά με τα διάφορα τμήματα, τα οποία υπάγονται σε κάθε εκπαιδευτικό Ίδρυμα. Οι πληροφορίες αυτές είναι οι εξής:

- Κωδικός τμήματος (department_id)
- Όνομα τμήματος (department_name)
- Προϊστάμενος τμήματος (department_manager)

3. Assignments

Στον παρόν πίνακα διατηρούνται τα διδασκόμενα μαθήματα με τις παρακάτω πληροφορίες:

- Κωδικός μαθήματος (assignment_id)

- Όνομα μαθήματος (assignment_name)

4. StudentsAssignments

Ο εν λόγω πίνακας χρησιμοποιείται ώστε να διατηρούνται οι σχέσεις μεταξύ των διδασκομένων και μαθημάτων. Με άλλα λόγια εδώ αποθηκεύονται πληροφορίες σχετικά με το ποια μαθήματα παρακολουθεί ο κάθε διδασκόμενος. Υπάρχουν τα εξής πεδία:

- Κωδικός αντιστοίχισης μαθήματος-διδασκόμενου (studentassignment_id)
- Κωδικός διδασκόμενου (studentassignment_userid)
- Κωδικός μαθήματος (studentassignment_assignmentid)

5. Results

Εδώ διατηρούνται οι βαθμολογίες για τον κάθε διδασκόμενο. Τα διαθέσιμα στοιχεία είναι τα εξής:

- Κωδικός βαθμολογίας (result_id)
- Κωδικός διδασκόμενου (result_userid)
- Κωδικός μαθήματος (result_assignmentid)
- Βαθμολογία μαθήματος (result_grade)

6. Menu

Στον πίνακα αυτό αποθηκεύονται οι διάφορες επιλογές πλοήγησης μέσα στην ιστοσελίδα για τους χρήστες. Εδώ βρίσκουμε τις εξής πληροφορίες:

- Κωδικός επιλογής (menu_id)
- Όνομα επιλογής που εμφανίζεται στην ιστοσελίδα (menu_caption)
- Όνομα ιστοσελίδας, όπου μεταφέρεται ο χρήστης (menu_name)

Όλοι οι παραπάνω πίνακες υπόκεινται στον εξ' ορισμού τύπο μηχανής αποθήκευσης της MySQL, ο οποίος καλείται MyISAM. Πρόκειται για πολύ γρήγορους στην αναζήτηση πίνακες, οι οποίοι αποθηκεύονται στο δίσκο και χρησιμοποιούν για τις λειτουργίες τους τη δομή B-tree.

Το σύστημα διαχείρισης βάσεων δεδομένων της MySQL μπορεί να υποστηρίζει μεγάλες βάσεις δεδομένων, με πολλούς πίνακες και αντίστοιχες εγγραφές. Ενδεικτικά αναφέρεται ότι το εν λόγω σύστημα υποστηρίζει βάσεις δεδομένων με περισσότερες από 50.000.000 εγγραφές και περισσότερους από 60.000 πίνακες.

Όσον αφορά τα ευρετήρια των πινάκων, υποστηρίζονται έως και 32 ανά πίνακα. Κάθε ευρετήριο μπορεί να απαρτίζεται από έως και 16 πεδία με συνολικό μέγεθος στα 500 bytes.

Επίσης αναφορικά με τους ταυτόχρονους χρήστες σε ένα τέτοιου είδους ηλεκτρονικό σύστημα, το οποίο βασίζεται σε MySQL, αυτοί δεν ξεπερνούν τους 50. Το προαναφερθέν πλήθος χρηστών αναφέρεται βέβαια σε συγκεκριμένες λειτουργίες (π.χ. ανάγνωσης ή/και εγγραφής) καθώς επίσης και σε low-end servers, υπολογιστικά συστήματα δηλαδή με σχετικά μέτριων επιδόσεων hardware. Σε απλά read-only queries το πλήθος των ταυτόχρονων χρηστών δύναται να φτάσει τους 1000.

Σχετικά με την υποστήριξη που παρέχει η γλώσσα PHP όσον αφορά το σύστημα διαχείρισης βάσεων δεδομένων της MySQL, αυτή παρέχεται εγγενώς. Η PHP πέραν του γεγονότος ότι περιλαμβάνει αρθρώματα (modules) για την υποστήριξη SQL βάσεων δεδομένων, αλλά και το πρότυπο ODBC (Open DataBase Connectivity), υποστηρίζει με έτοιμες, στον πυρήνα, της συναρτήσεις το σύστημα της MySQL, χωρίς επιπλέον παραμετροποίηση είτε του συστήματος διαχείρισης των βάσεων δεδομένων, είτε της γλώσσας αυτής καθ' αυτής.

3.3. Τα Αρχεία PHP

Τα αρχεία που απαρτίζουν την εφαρμογή είναι στο σύνολό τους 25. Αναλυτικά τα αρχεία που περιλαμβάνονται είναι τα εξής:

1. **add_assignment.php**

Μέγεθος: 1,36 KB

Πλήθος γραμμών: 45

Περιγραφή: Είναι η σελίδα μέσω της οποίας ο διδάσκοντας κάνει την καταχώρηση των μαθημάτων που αυτός διδάσκει.

2. **add_department.php**

Μέγεθος: 1,86 KB

Πλήθος γραμμών: 56

Περιγραφή: Είναι η σελίδα μέσω της οποίας ένας διαχειριστής κάνει την καταχώρηση των επιμέρους τμημάτων ενός εκπαιδευτικού ιδρύματος.

Κώδικας: Με τον κώδικα που ακολουθεί καταχωρούνται στις μεταβλητές \$name και \$manager τα περιεχόμενα των αντίστοιχων πεδίων της φόρμας εισαγωγής τμήματος. Μια βασική διαδικασία είναι η αποκοπή των χαρακτήρων backslash (\), από το δοθέν string, πριν από την καταχώρηση των δεδομένων στη βάση δεδομένων. Επιπλέον διενεργείται ένας έλεγχος από την εφαρμογή, και όχι από το σύστημα διαχείρισης βάσεων δεδομένων MySQL, για την ύπαρξη διπλότυπων εγγραφών, ώστε να αποφευχθεί και αυτή η περίπτωση.

```
$name=$_POST['name'];
$manager=$_POST['manager'];
$name=stripslashes($name);
$manager=stripslashes($manager);
$sql_name_check=mysql_query("select department_name from departments where
department_name='$name'");
$name_check=mysql_num_rows($sql_name_check);
if ($name_check>0)
{
    echo "Το όνομα τμήματος χρησιμοποιείται από άλλον τμήμα! Παρακαλώ εισάγετε
διαφορετικά στοιχεία!";
    unset($name);
    print_add_department();
    exit();
}
```

3. add_result.php

Μέγεθος: 3,49 KB

Πλήθος γραμμών: 100

Περιγραφή: Είναι η σελίδα της πρώτης περίπτωσης αποστολής βαθμολογίας από τον διδάσκοντα. Πρόκειται για τη φόρμα με τους διδασκομένους που έχουν δηλώσει το μάθημα, για πρώτη φορά, ή έχουν βαθμολογία σε προηγούμενη εξέταση, μικρότερη του 5,00. Στην εν λόγω φόρμα συμπληρώνονται οι ενημερωμένες βαθμολογίες, οι οποίες εν συνεχεία αποστέλλονται στο σύστημα.

Κώδικας: Αρχικά επιβεβαιώνεται η μέθοδος αποστολής της βαθμολογίας. Στη συγκεκριμένη περίπτωση, αφού η μεταβλητή grademethod φέρει την τιμή 0, η αποστολή θα γίνει με τη χρήση φόρμας. Τα στοιχεία της φόρμας, με άλλα

λόγια, οι διδασκόμενοι, οι οποίοι θα εμφανιστούν, εξαρτώνται από το ερώτημα SQL που ακολουθεί και το οποίο έχει ως αποτέλεσμα για την εφαρμογή, την εμφάνιση όλων των διδασκομένων που έχουν καταχωρήσει το συγκεκριμένο μάθημα και φέρουν σε αυτό βαθμολογία μικρότερη του 5,00. Υπενθυμίζεται εδώ, ότι διδασκόμενοι οι οποίοι δεν έχουν εξεταστεί ουδέποτε στο εν λόγω μάθημα, φέρουν βαθμολογία 0,00.

Επιπλέον, η μεταβλητή \$result, στην οποία εκχωρείται το αποτέλεσμα του ερωτήματος, το οποίο αποστέλλεται από το σύστημα διαχείρισης της βάσης δεδομένων προς την εφαρμογή, ελέγχεται και στην περίπτωση στην οποία δεν έχει εκχωρηθεί τιμή σε αυτή, τότε το σύστημα τυπώνει το ανάλογο μήνυμα σφάλματος-αποτυχίας.

```
if ($_REQUEST['grademethod']==0)
{
    echo "<form action=\"add_result.php\" method=\"POST\">";
    echo "<table align=center border=0>";
    $sql="select * from studentsassignments,users,results where ";
    $sql.="studentsassignments.studentassignment_assignmentid=$lessonid and ";
    $sql.="users.user_id=studentsassignments.studentassignment_userid and ";
    $sql.="results.result_assignmentid=studentsassignments.studentassignment_assignmentid
and ";
    $sql.="results.result_userid=users.user_id and ";
    $sql.="results.result_grade<5.00";
    $result=mysql_query($sql);
    if (!$result)
    {
        die ("Σφάλμα κατά την εκτέλεση του ερωτήματος");
    }
}
```

4. add_user.php

Μέγεθος: 6,7 KB

Πλήθος γραμμών: 148

Περιγραφή: Πρόκειται για τη φόρμα, την οποία πρέπει να συμπληρώσει κάθε χρήστης (διδασκόμενος ή/και διδάσκοντας), ώστε να αποκτήσει τις απαραίτητες πληροφορίες για να συνδεθεί στο σύστημα.

Κώδικας: Κατά τη διάρκεια αποστολής των στοιχείων, τα οποία έχει εισάγει ένας νέος για το σύστημα χρήστης, πραγματοποιείται μια σειρά ελέγχων,

αναφορικά με την εγκυρότητα των απεσταλμένων στοιχείων. Τέτοιοι έλεγχοι έχουν σκοπό τη διασφάλιση των δεδομένων και έχουν να κάνουν αρχικά με την εγκυρότητα της διεύθυνσης e-mail του χρήστη. Αυτό είναι πολύ σημαντικό καθότι η συγκεκριμένη διεύθυνση θα χρησιμοποιηθεί για την αποστολή του αρχικού κωδικού πρόσβασης του χρήστη, στο σύστημα. Επιπλέον πραγματοποιείται ένας έλεγχος εγκυρότητας για το δηλωθέντα αριθμό τηλεφώνου του χρήστη, ο οποίος όπως η διεύθυνση e-mail θα πρέπει απαραίτητα να περιέχει τους χαρακτήρες @ και ., θα πρέπει να περιλαμβάνει 10 αριθμητικά ψηφία και φυσικά όχι γράμματα.

```
// Έλεγχος εγκυρότητας email
$expression = "^[_A-Za-z0-9-]+@[_A-Za-z-]+(\.[A-Za-z]+)(\.[A-Za-z]+)*$";
if(!ereg("$expression",$_POST['email']))
{
    echo "Η διεύθυνση email δεν είναι έγκυρη";
    print_add_user();
    exit();
}

// Έλεγχος εγκυρότητας τηλεφώνου
if ((ereg("[a-zA-Zα-ωΑ-Ω]",$_POST['phone']) || (strlen($_POST['phone'])!=10)))
{
    echo "Ο αριθμός τηλεφώνου δεν είναι έγκυρος";
    print_add_user();
    exit();
}
```

Βέβαια υπάρχει και ένα κομμάτι κώδικα γραμμένο σε JavaScript, με τη βοήθεια του οποίου, κατά τη διάρκεια καταχώρησης των δεδομένων από έναν διδάσκοντα απενεργοποιείται αυτομάτως το πεδίο που αντιστοιχεί στον αριθμό μητρώου, μιας και η εν λόγω χρήστες δεν διαθέτει τέτοιου είδους χαρακτηριστικά.

```

<script Language="JavaScript">
var _T = "locked";
var _F = "unlocked";
function lockIt(_P)
{
var _L = document.frmMyForm.lck.value;
document.frmMyForm.registernumber.disabled=(document.frmMyForm.lck.value==_L==_F)
?_T:_F==_T;
}
function isDis() { return (document.frmMyForm.lck.value==_T); }
</script>
echo "<form id=\"frmMyform\" name=\"frmMyForm\" action=\"add_user.php\"
method=\"POST\">";
echo "<tr><td><strong>Αριθμός Μητρώου:</strong></td><td><input type=\"text\"
id=\"registernumber\" name=\"registernumber\" name=\" onfocus=\"if(isDis())blur();\"
maxlength=\"4\" size=\"4\"> (Απαραίτητο ΜΟΝΟ για τους φοιτητές)</td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"lck\" value=\"unlocked\"><td></tr>";

```

5. choose_assignment.php

Μέγεθος: 1,62 KB

Πλήθος γραμμών: 52

Περιγραφή: Είναι η φόρμα όπου γίνεται, από το διδάσκοντα, η επιλογή του μαθήματος, για το οποίο ο τελευταίος θέλει να αποστείλει τις αντίστοιχες βαθμολογίες.

Κώδικας: Ο μηχανισμός που παρουσιάζεται παρακάτω χρησιμοποιείται ευρέως στο σύστημα και σαν σκοπό έχει την εφαρμογή διάφορων ερωτημάτων στη βάση δεδομένων. Βάσει του μηχανισμού αυτού τα δεδομένα αποστέλλονται στην εφαρμογή, η οποία εν συνεχεία τα μορφοποιεί αναλόγως, εισάγοντας τα σε πίνακες για το καλύτερο δυνατό οπτικό αποτέλεσμα.

```

function print_choose_assignment()
{
echo "<form action=\"choose_assignment.php\" method=\"post\">";
echo "<table border=0 align=center>";
echo "<tr>";
echo "<th></th>";

```

```

echo "<th>Όνομα Μαθήματος</th>";
echo "</tr>";
$result=mysql_query("select * from assignments");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
    $id=$line['assignment_id'];
    $name=$line['assignment_name'];
    echo "<tr>";
    echo "<td><input type=\"radio\" name=\"radio\" value=$id></td>";
    echo "<td>$name</td>";
    echo "</tr>";
}
echo "<tr><td colspan=2 align=center>";
echo "Τρόπος εισαγωγής: ";
echo "<select name=\"grade_method\">";
echo "<option value=\"0\">Φόρμα</option>";
echo "<option value=\"1\">Αρχείο Excel</option>";
echo "</select>";
echo "<input type=\"hidden\" name=\"posted\" value=\"1\">";
echo "</table>";
echo "<p align=center>";
echo "<td><input type=\"submit\" name=\"submit\" value=\"Επιλογή\"></td>";
echo "</p>";
echo "</form>";
html_footer();
}

```

6. db_egrade.inc.php

Μέγεθος: 0,4 KB

Πλήθος γραμμών: 11

Περιγραφή: Στο εν λόγω αρχείο περιέχονται οι πληροφορίες σχετικά με τη σύνδεση της εφαρμογής στο σύστημα διαχείρισης βάσεων δεδομένων και κατ' επέκταση στη βάση δεδομένων αυτή καθ' αυτή.

Λόγω της κρισιμότητας του αρχείου, αυτό δεν πρέπει να είναι στο ίδιο φυσικό κατάλογο με τα υπόλοιπα αρχεία της εφαρμογής, αλλά σε κάποιο άλλο, το οποίο δεν είναι προσβάσιμο μέσω του Internet.

7. delete_assignment.php

Μέγεθος: 1,48 KB

Πλήθος γραμμών: 48

Περιγραφή: Σε αυτή τη σελίδα παρουσιάζονται όλα τα διαθέσιμα μαθήματα. Επιλέγοντας κάποιο (ένα κάθε φορά), ο διδάσκοντας ή/και ο διαχειριστής μπορεί να το διαγράψει από το σύστημα. Κατά τη διαγραφή του βέβαια, διαγράφονται και όσα στοιχεία (βαθμολογίες), σχετίζονται με αυτό.

8. delete_department.php

Μέγεθος: 1,59 KB

Πλήθος γραμμών: 55

Περιγραφή: Σε αυτή τη σελίδα παρουσιάζονται όλα τα διαθέσιμα τμήματα για το εκπαιδευτικό ίδρυμα. Επιλέγοντας κάποιο (ένα κάθε φορά), ο διαχειριστής μπορεί να το διαγράψει από το σύστημα. Κατά τη διαγραφή του βέβαια, διαγράφονται και όσα στοιχεία (διδάσκοντες, μαθήματα, διδασκόμενοι, βαθμολογίες), σχετίζονται με αυτό.

9. delete_user.php

Μέγεθος: 1,56 KB

Πλήθος γραμμών: 58

Περιγραφή: Σε αυτή τη σελίδα παρουσιάζονται όλοι οι διαθέσιμοι χρήστες που υπάρχουν στο σύστημα, ανεξαρτήτως ομάδας χρηστών. Επιλέγοντας κάποιον (έναν κάθε φορά), ο διαχειριστής μπορεί να τον διαγράψει από το σύστημα. Κατά τη διαγραφή του βέβαια, διαγράφονται και όσες πληροφορίες σχετίζονται με αυτόν.

10. edit_user.php

Μέγεθος: 3,9 KB

Πλήθος γραμμών: 77

Περιγραφή: Πρόκειται για τη φόρμα όπου κάθε χρήστης, ανεξάρτητα από την ομάδα στην οποία ανήκει, μπορεί να μεταβάλλει τα προσωπικά του στοιχεία. Μπορεί να μεταβάλλει οποιαδήποτε πληροφορία αφορά το προφίλ του με μοναδική εξαίρεση το όνομα χρήστη, το οποίο για λόγους ασφαλείας και εύρυθμης λειτουργίας της βάσης δεδομένων, παραμένει αναντικατάστατο.

11. egrade_fns.php

Μέγεθος: 0,11 KB

Πλήθος γραμμών: 5

Περιγραφή: Ο ρόλος του συγκεκριμένου αρχείου είναι περισσότερο βοηθητικός, μιας και το εν λόγω αρχείο καλεί τις διαθέσιμες από το σύστημα συναρτήσεις, έτσι ώστε η εφαρμογή να μπορεί να λειτουργήσει. Καλείται σχεδόν σε κάθε αρχείο της εφαρμογής.

12. includes.inc

Μέγεθος: 4,42 KB

Πλήθος γραμμών: 163

Περιγραφή: Το εν λόγω αρχείο περιέχει την υλοποίηση διάφορων συναρτήσεων, η χρήση των οποίων είναι απαραίτητη για την εύρυθμη λειτουργία της διαδικασίας αποστολής της βαθμολογίας με τη χρήση αρχείου Excel. Οι συναρτήσεις αυτές σχετίζονται με τη μετατροπή χαρακτήρων Unicode σε χαρακτήρες String (αναγνωρίσιμους από την PHP), τη λήψη δεδομένων από φύλλα εργασίας του Excel, την προετοιμασία των πινάκων της PHP για την καταχώρηση των στοιχείων από το αρχείο του Excel κ.λπ.

13. index.php

Μέγεθος: 0,44 KB

Πλήθος γραμμών: 21

Περιγραφή: Είναι η αρχική σελίδα της εφαρμογής, στην οποία μεταφέρεται ο χρήστης μετά από την επιτυχημένη σύνδεσή του στο σύστημα. Εδώ παρατίθενται όλες οι επιλογές πλοήγησης, τις οποίες ο χρήστης μπορεί να χρησιμοποιήσει, ώστε να διεκπεραιώσει όλες τις επιθυμητές εργασίες του.

14. login.php

Μέγεθος: 1,63 KB

Πλήθος γραμμών: 51

Περιγραφή: Είναι η σελίδα, η οποία προτρέπει το χρήστη να εισάγει το «Όνομα Χρήστη» καθώς επίσης και τον «Κωδικό Πρόσβασης», ώστε να συνδεθεί στο σύστημα. Αν πρόκειται για νέο χρήστη, ο οποίος δεν διαθέτει να προαναφερθέντα στοιχεία, τότε υπάρχει η δυνατότητα μεταφοράς του στη σελίδα δημιουργίας «Νέου Χρήστη».

Κώδικας: Η πιστοποίηση του χρήστη πραγματοποιείται με τη δημιουργία ενός SQL ερωτήματος και την αποστολή του στη βάση δεδομένων, αφού προηγηθεί η συμπλήρωση των απαραίτητων πεδίων καθώς επίσης και η αποκοπή των χαρακτήρων backslash (\). Αφού ο χρήστης πιστοποιηθεί,

πραγματοποιείται η αρχικοποίηση ενός νέου session για την πιστοποίησή του σε όλο το σύστημα.

```
function connect_user()
{
if ((!isset($_POST['posted'])) || ($_POST['posted']!=1))
{
    print_connect_user();
    exit;
}
if (!filled_out($_POST))
{
    print_connect_user();
    exit();
}
$username=$_POST['username'];
$password=$_POST['password'];
$username=stripslashes($username);
$password=stripslashes($password);
$result=mysql_query("select * from users where user_username='$username' and
user_password='$password'");
if (!$result)
{
    die("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_error());
}
if (mysql_num_rows($result)==0)
{
    print_connect_user();
    exit();
}
$line=mysql_fetch_array($result,MYSQL_BOTH);
$_SESSION['userid']=$line['user_id'];
$_SESSION['securitylevel']=$line['user_securitylevel'];
$_SESSION['username']=$line['user_username'];
header ("Location:index.php");
}
```

15. logout.php

Μέγεθος: 0,14 KB

Πλήθος γραμμών: 7

Περιγραφή: Η σελίδα αυτή τερματίζει τη εκάστοτε σύνοδο (session), του χρήστη με το σύστημα, ενημερώνοντας τον ταυτόχρονα ότι έχει αποσυνδεθεί από αυτό.

16. pages_fns.php

Μέγεθος: 1,31 KB

Πλήθος γραμμών: 48

Περιγραφή: Εδώ περιέχονται διάφορες συναρτήσεις, η χρήση των οποίων απαιτείται για τη σωστή λειτουργία της εφαρμογής. Οι συναρτήσεις αυτές έχουν να κάνουν κυρίως με πληροφορίες που εμφανίζονται στατικά στις ιστοσελίδες της εφαρμογής, όπως π.χ. η κεφαλίδα και το υποσέλιδο σε κάθε σελίδα, ή το μενού με τις επιλογές πλοήγησης της εφαρμογής.

Κώδικας: Η δημιουργία σταθερής κεφαλίδας, υποσέλιδου καθώς επίσης και του μενού επιλογών για την πλοήγηση του χρήστη στο σύστημα, υπό τη μορφή συναρτήσεων, για να επιτευχθεί η επαναχρησιμοποίηση του κώδικα σε διάφορες σελίδες παρουσιάζεται εδώ.

```
function html_header($title)
{
    echo "<html>";
    echo "<head>";
    echo "<title>$title</title>";
    echo "<style>";
    echo "body { font-family: Arial, Helvetica, sans-serif; font-size: 13px; text-align=center}";
    echo "li, td { font-family: Arial, Helvetica, sans-serif; font-size: 13px }";
    echo "hr { color: #333333; width=300; text-align=center}";

    echo "a { color: #000000 }";
    echo "</style>";
    echo "</head>";
    echo "<body>";
    echo "<p align=center><img src=\"images/logo.jpg\" alt=\"eGrade\" height=100 width=700></p>";
    echo "<hr>";
}
function html_footer()
{
    echo "<hr>";
}
```

```
echo "<p align=center>(C) Giannis Gkoutisidis - 2004</p>";
echo "</body>";
echo "</html>";
}
function print_menu()
{
$result=mysql_query("select * from menu order by menu_id");
if (!$result)
{
    echo "Σφάλμα κατά την εκτέλεση του ερωτήματος!";
    exit();
}
echo "<p align=center>";
echo "| ";
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
    $caption=$line['menu_caption'];
    $url="http://".$_SERVER['SERVER_NAME']."/ggkoutisidis/" . "index.php?page=".$line['menu_name'];
    echo "<a href=\"".$url.\">$caption<a>";
    echo " | ";
}
echo "</p>";
}
```

17. register_assignment.php

Μέγεθος: 2,81 KB

Πλήθος γραμμών: 70

Περιγραφή: Η εν λόγω σελίδα εμφανίζει στο χρήστη, και πιο συγκεκριμένα στο διδασκόμενο, τα διαθέσιμα μαθήματα, τα οποία μπορεί να καταχωρήσει, ώστε να ενημερώνεται για τις βαθμολογίες του σε αυτά. Η επιλογή των μαθημάτων είναι πολλαπλή, δηλ. ο χρήστης μπορεί να επιλέξει με από μια σελίδα όλα τα μαθήματα που τον ενδιαφέρουν. Παρέχεται βέβαια και η δυνατότητα στον χρήστη να επιλέξει επιπλέον μαθήματα σε μια δεύτερη σύνδεσή του στο σύστημα, χωρίς όμως να εμφανίζονται όσα μαθήματα έχει καταχωρήσει σε προηγούμενη χρονική στιγμή.

18. send_result.php

Μέγεθος: 7,26 KB

Πλήθος γραμμών: 197

Περιγραφή: Είναι η σελίδα της δεύτερης περίπτωσης αποστολής βαθμολογίας από τον διδάσκοντα. Πρόκειται για ένα σύνολο βημάτων μέχρι την ολοκλήρωση της όλης διαδικασίας, βάσει της οποίας απαιτείται η επιλογή του αρχείου Excel από το μέσο αποθήκευσης του διδάσκοντα, η αποστολή του στο σύστημα, και στην περίπτωση που αυτό κριθεί σωστό, η καταχώρησή του στη βάση δεδομένων του συστήματος. Ο διδάσκοντας τέλος ενημερώνεται για τις βαθμολογίες, οι οποίες έχουν καταχωρηθεί.

19. show_result.php

Μέγεθος: 1,83 KB

Πλήθος γραμμών: 47

Περιγραφή: Στη σελίδα αυτή παρουσιάζονται στο διδασκόμενο οι βαθμολογίες των μαθημάτων, τα οποία έχει καταχωρήσει. Σε όσα μαθήματα δεν έχει εξεταστεί ούτε μια φορά εμφανίζεται η ενδεικτική βαθμολογία 0,00.

20. users_fns.php

Μέγεθος: 2,9 KB

Πλήθος γραμμών: 96

Περιγραφή: Εδώ περιέχονται διάφορες συναρτήσεις, η χρήση των οποίων απαιτείται για τη σωστή λειτουργία της εφαρμογής. Οι συναρτήσεις αυτές έχουν να κάνουν κυρίως με την αφανή αλληλεπίδραση των χρηστών με το σύστημα. Περιέχονται δηλ. συναρτήσεις όπως π.χ. η δημιουργία του αρχικού κωδικού πρόσβασης του χρήστη, ο έλεγχος των δικαιωμάτων του εκάστοτε χρήστη, οι επιπλέον διαδικασίες που πρέπει να εκτελεστούν με τη διαγραφή ενός χρήστη κ.λπ.

Κώδικας: Ο έλεγχος για το αν τα απαραίτητα πεδία έχουν συμπληρωθεί από τον χρήστη επιτυγχάνεται ως εξής:

```
function filled_out($form_vars) // Έλεγχος συμπληρωμένων πεδίων
{
    foreach ($form_vars as $key=>$value)
    {
        if (!isset($key) || ($value==""))
        {
            return false;
        }
    }
}
```

```
        return true;
    }
}
```

Επίσης η δημιουργία των αρχικών κωδικών πρόσβασης για κάθε νέο χρήστη υπόκειται στην ύπαρξη ενός αλγορίθμου αλφαριθμητικών με μέγεθος 10 χαρακτήρων, οι οποίοι μπορούν να περιλαμβάνουν πεζούς χαρακτήρες καθώς επίσης και αριθμούς. Η παραγωγή των εν λόγω κωδικών επιτυγχάνεται με τη χρήση του παρακάτω κώδικα.

```
function generate_password($digits) // Γεννήτρια κωδικού πρόσβασης
{
    $salt="abcdefghijklmnopqrstuvwxyz0123456789";
    srand((double)microtime()*1000000);
    $pass="";
    $i=0;
    while ($i<$digits)
    {
        $num=rand()%33;
        $tmp=substr($salt,$num,1);
        $pass=$pass.$tmp;
        $i++;
    }
    return $pass;
}
```

Όπως έχει προαναφερθεί, η λειτουργία διαγραφής ενός διδασκόμενου από το σύστημα έχει ως άμεσο αποτέλεσμα την ταυτόχρονη διαγραφή και όλων των σχετικών βαθμολογίων του εν λόγω διδασκόμενου από αυτό. Η λειτουργία που μόλις περιγράφηκε πλαισιώνεται από τον εξής κώδικα:

```
function delete_user_fns($id)
{
    $sql=array("select user_securitylevel from users where user_id=$id",
              "select * from studentsassignments where studentassignment_userid=$id",
              "delete from studentsassignments where studentassignment_userid=$id",
              "delete from results where result_userid=$id",
              "delete from users where user_id=$id")
}
```

```

);
$result=mysql_query($sql[0]);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!");}
$line=mysql_fetch_array($result,MYSQL_BOTH);
if ($line['user_securitylevel']==1)
{
$result=mysql_query($sql[1]);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!");}
$rows=mysql_num_rows($result);
if ($rows>0) // Έχει καταχωρηθεί έστω κι ένα μάθημα από τον φοιτητή
{
$result=mysql_query($sql[2]);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
$result=mysql_query($sql[3]);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
$result=mysql_query($sql[4]);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
}
else {$result=mysql_query($sql[4]);}
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
}
else {$result=mysql_query($sql[4]);}
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
}
}

```

Μια ακόμη συνάρτηση, η οποία χρίζει ιδιαίτερης μνείας είναι η `check_security`. Με τη χρήση της εν λόγω συνάρτησης επιτυγχάνεται ο έλεγχος της ομάδας στην οποία ανήκει ο κάθε χρήστης και η πιστοποίησή του όσον αφορά το αν ο συγκεκριμένος χρήστης μπορεί να προσπελάσει τη σελίδα που καλεί ή όχι. Ο κώδικας έχει ως εξής.

```

function check_security($securitylevel)
{
if (!isset($_SESSION['userid']))
{
header("Location:login.php");
exit();
}
elseif ((isset($_SESSION['securitylevel'])) || ($_SESSION['securitylevel']<$securitylevel))

```

```
{
    echo "Δεν έχετε τα απαραίτητα δικαιώματα!";
    exit();
}
```

21. `\parser\dataprovider.php`

Μέγεθος: 3, 74 KB

Πλήθος γραμμών: 131

Περιγραφή: Πρόκειται για ένα αρχείο, με τη βοήθεια του οποίου προσδιορίζονται τα βασικά στοιχεία του προς αποστολή αρχείου Excel στο σύστημα, όπως το μέγεθός του, τα blocks από τα οποία απαρτίζεται και η γενική γραμμογράφηση που ακολουθεί.

22. `\parser\debug.php`

Μέγεθος: 6,64 KB

Πλήθος γραμμών: 249

Περιγραφή: Η χρήση του εν λόγω αρχείου είναι απαραίτητη για τη συντήρηση και το γενικό έλεγχο του συγκεκριμένου κομματιού της εφαρμογής. Βάσει αυτού του αρχείου συλλέγονται οι διάφορες πληροφορίες για όλες τις εργασίες που λαμβάνουν χώρα κατά την αποστολή ενός αρχείου Excel στο σύστημα μέχρι την τελική καταχώρηση των δεδομένων στην βάση δεδομένων, καθώς επίσης και για τα διάφορα σφάλματα που ίσως ανακύψουν.

23. `\parser\exceldate.php`

Μέγεθος: 1,67 KB

Πλήθος γραμμών: 86

Περιγραφή: Είναι το αρχείο με το οποίο γίνεται η αντιστοίχιση των ημερομηνιών, όταν αυτές υπόκεινται σε επεξεργασία από διαφορετικά λειτουργικά συστήματα όπως Windows, Linux, Macintosh κ.α. Επειδή η εφαρμογή είναι «cross-platform» (ανεξάρτητη λειτουργικού συστήματος), απαιτείται η χρήση τέτοιου είδους κώδικα για την εύρυθμη λειτουργία της εφαρμογής.

24. `\parser\excelfont.php`

Μέγεθος: 3,57 KB

Πλήθος γραμμών: 121

Περιγραφή: Ο κώδικας στο συγκεκριμένο αρχείο χρησιμοποιείται για το χειρισμό των Unicode χαρακτήρων, οι οποίοι είναι διαθέσιμοι στην εφαρμογή Microsoft Excel, και οι οποίοι απαιτείται τελικά να μετατραπούν σε απλούς χαρακτήρες, έτσι ώστε να γίνουν συμβατοί αφενός μεν με τους τύπους δεδομένων που μπορούν να εισαχθούν σε έναν πίνακα της PHP, και αφετέρου σε έναν πίνακα βάσης δεδομένων της MySQL.

25. \parser\excelparser.php

Μέγεθος: 30,19 KB

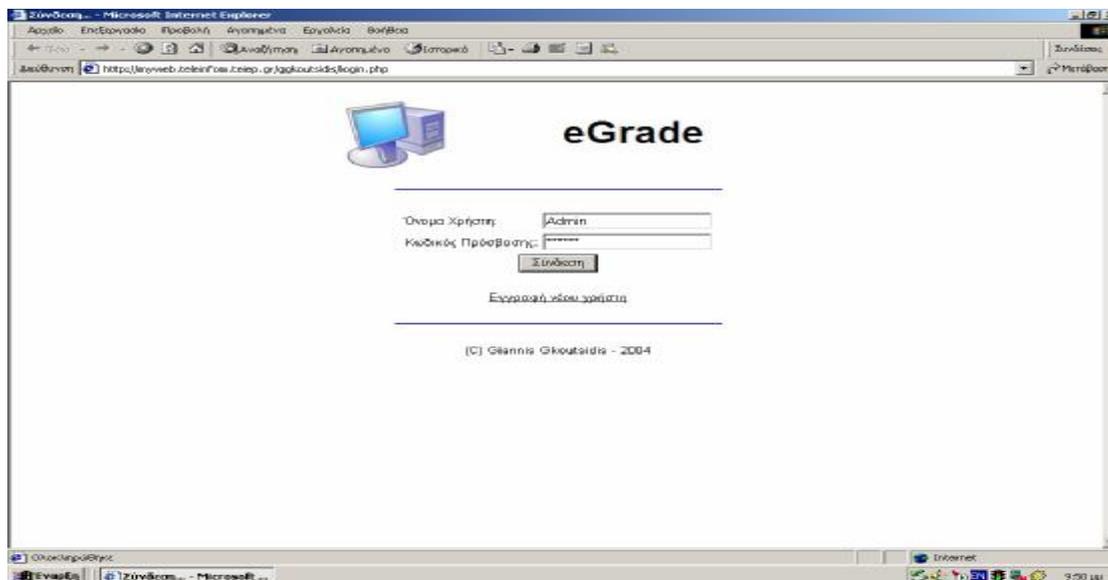
Πλήθος γραμμών: 1.093

Περιγραφή: Πρόκειται για ένα αρχείο-κλάση της PHP, το οποίο αναλαμβάνει τη διαδικασία ελέγχου ενός δυαδικού αρχείου Excel, όσον αφορά την ορθότητά του. Αφού το απαιτούμενο αρχείο κριθεί κατάλληλο, τότε ακολουθεί η καταχώρηση όλων των πληροφοριών που περιέχει σε μια βάση δεδομένων της MySQL.

3.4. Βήμα προς Βήμα - Διαχειριστής

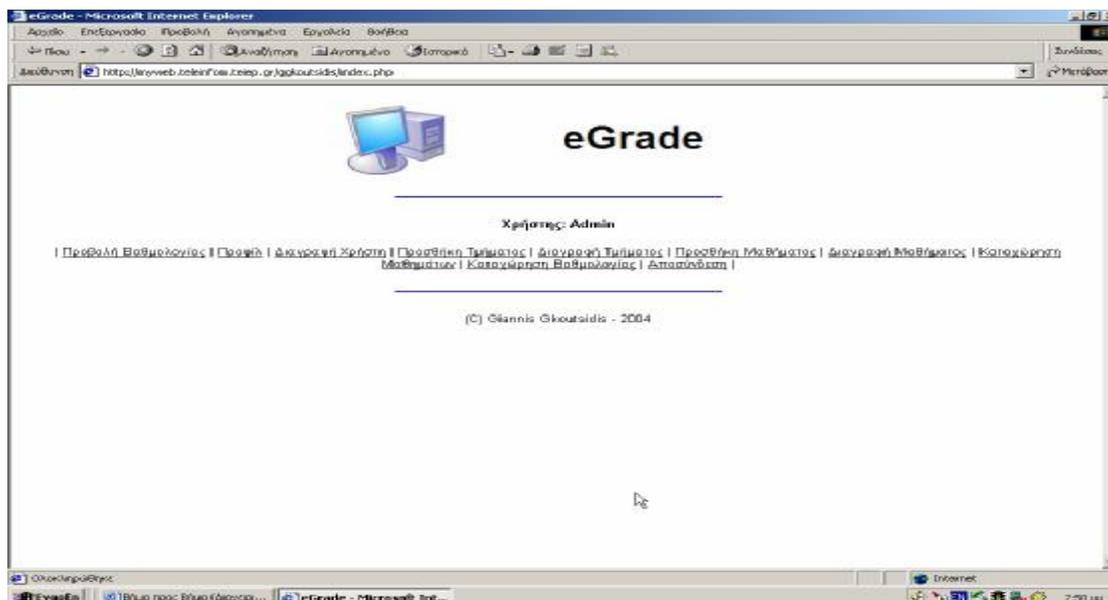
Όπως προαναφέρθηκε σε προηγούμενο κεφάλαιο η διαδικασία εισαγωγής ενός χρήστη στην ομάδα των Διαχειριστών επαφίεται στη δικαιοδοσία ενός ήδη υπάρχοντα χρήστη στη συγκεκριμένη ομάδα και όχι στο σύστημα αυτό καθ' αυτό για λόγους ασφαλείας.

Κάθε χρήστης, σε οποιαδήποτε ομάδα κι αν ανήκει, πρέπει υποχρεωτικά να συνδεθεί στο σύστημα με το Όνομα Χρήστη και τον Κωδικό Πρόσβασης, που του έχουν ανατεθεί, ώστε να διεκπεραιώσει τις εργασίες του, όπως φαίνεται παρακάτω.



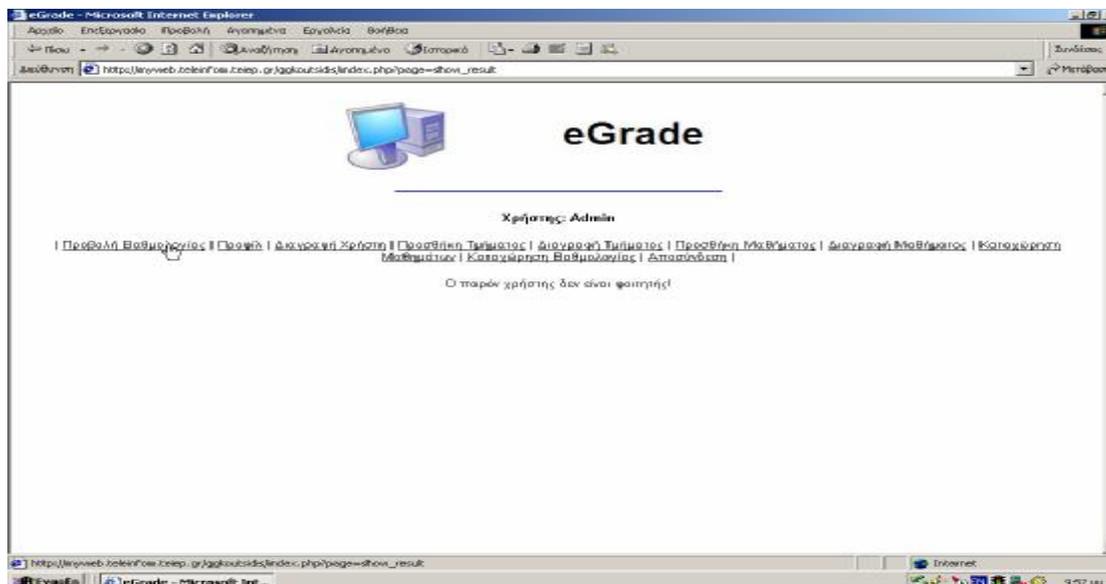
Σχήμα 3.4.1. Οθόνη σύνδεσης στο σύστημα

Αφού ολοκληρωθεί η διαδικασία πιστοποίησης του χρήστη από το σύστημα, ο χρήστης μεταφέρεται στην κεντρική σελίδα του συστήματος.



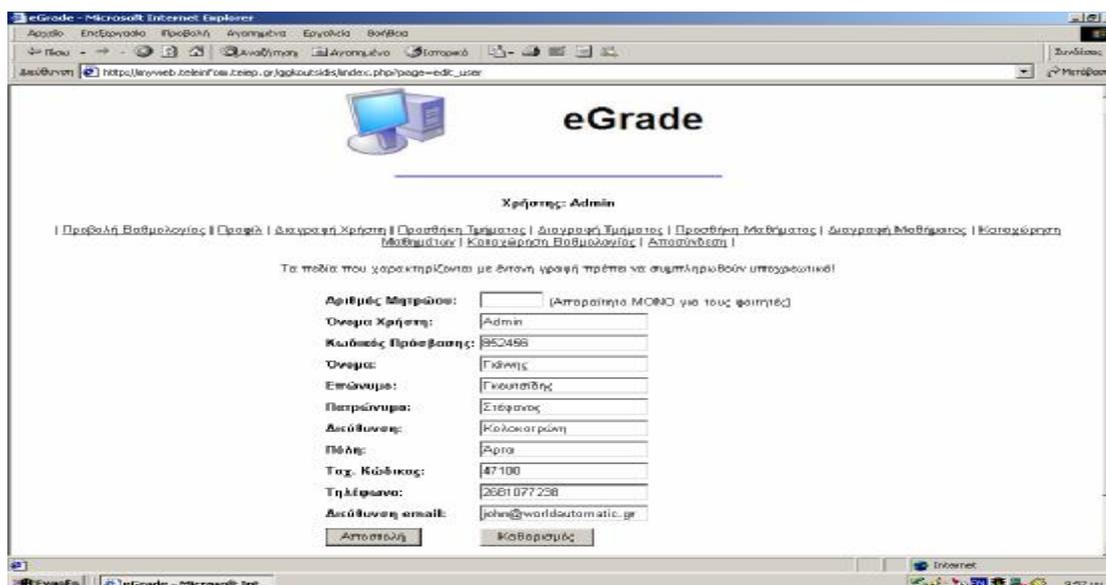
Σχήμα 3.4.2. Κεντρική σελίδα εφαρμογής

Ο χρήστης Admin δεν έχει δικαιώματα σε λειτουργίες όπως π.χ. την Προβολή Βαθμολογίας, καθότι δεν αποτελεί διδασκόμενο. Οπότε σε μια ενδεχόμενη αίτηση προς την εφαρμογή για προβολή της βαθμολογίας του θα λάβει το αντίστοιχο μήνυμα σφάλματος από την τελευταία.



Σχήμα 3.4.3. Προβολή βαθμολογίας του χρήστη Admin

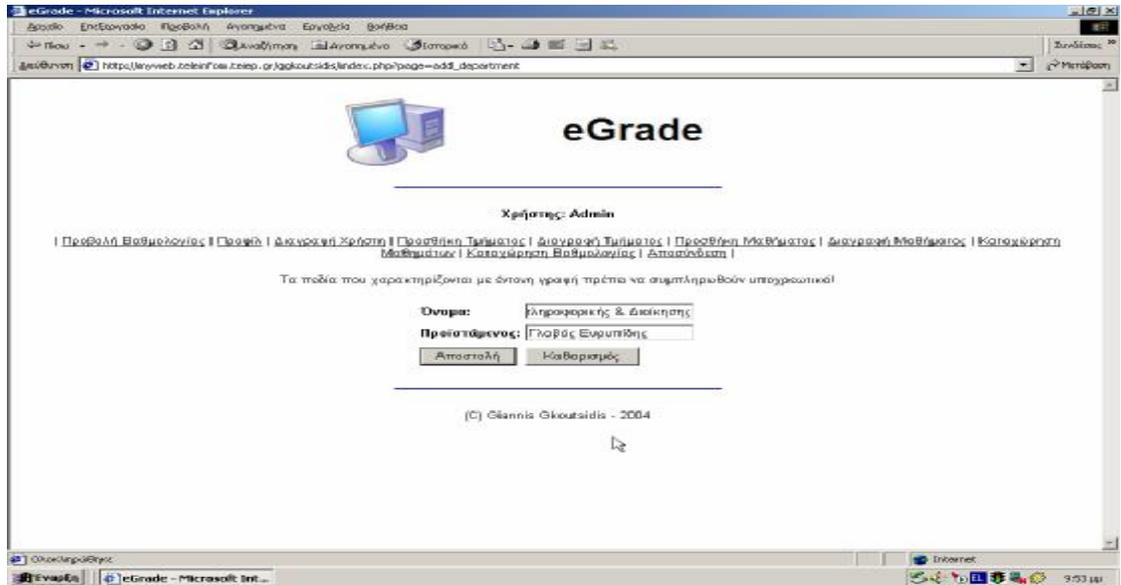
Ο κάθε χρήστης μπορεί, μέσα από τις σελίδες της εφαρμογής, να μεταβάλει προσωπικά του στοιχεία, από το μενού επιλογών Προφίλ, όπως φαίνεται παρακάτω.



Σχήμα 3.4.4. Μεταβολή προσωπικών στοιχείων χρήστη

Ανάμεσα στις διάφορες δικαιοδοσίες, με τις οποίες έχει επιφορτιστεί η ομάδα των Διαχειριστών, αναφέρονται οι εξής:

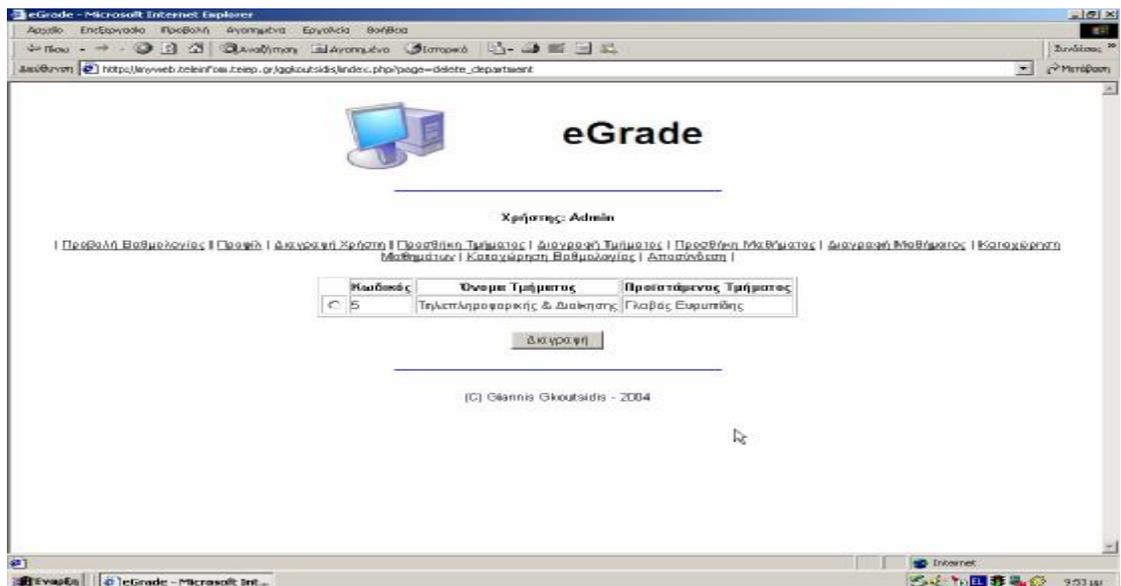
- Προσθήκη τμήματος



Σχήμα 3.4.5. Προσθήκη τμήματος

Από την εν λόγω επιλογή ένας Διαχειριστής μπορεί εισάγοντας το Όνομα Τμήματος και το Ονοματεπώνυμο του Προϊσταμένου να δημιουργήσει ένα νέο τμήμα, στο οποίο θα δημιουργηθούν εκ των υστέρων νέα μαθήματα.

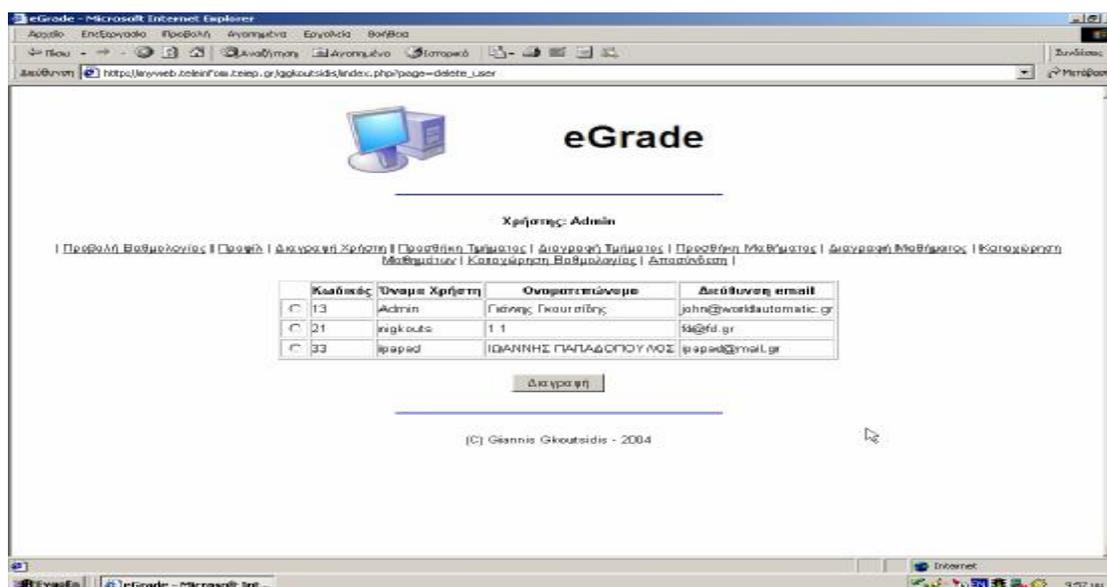
- Διαγραφή τμήματος



Σχήμα 3.4.6. Διαγραφή τμήματος

Στην παρούσα σελίδα εμφανίζονται όλα τα διαθέσιμα τμήματα. Η επιλογή ενός από αυτά σε συνδυασμό με την εντολή Διαγραφή, έχει σαν αποτέλεσμα τη διαγραφή του επιλεγμένου τμήματος από το σύστημα.

- Διαγραφή χρήστη



Σχήμα 3.4.7. Διαγραφή χρήστη

Μέσω της παρούσας σελίδας πραγματοποιείται η διαγραφή ενός χρήστη. Ο Διαχειριστής έχει στη διάθεσή του τα εξής στοιχεία για κάθε χρήστη, ο οποίος είναι εγγεγραμμένος στο σύστημα:

- Κωδικό χρήστη
- Όνομα χρήστη
- Ονοματεπώνυμο
- Διεύθυνση e-mail

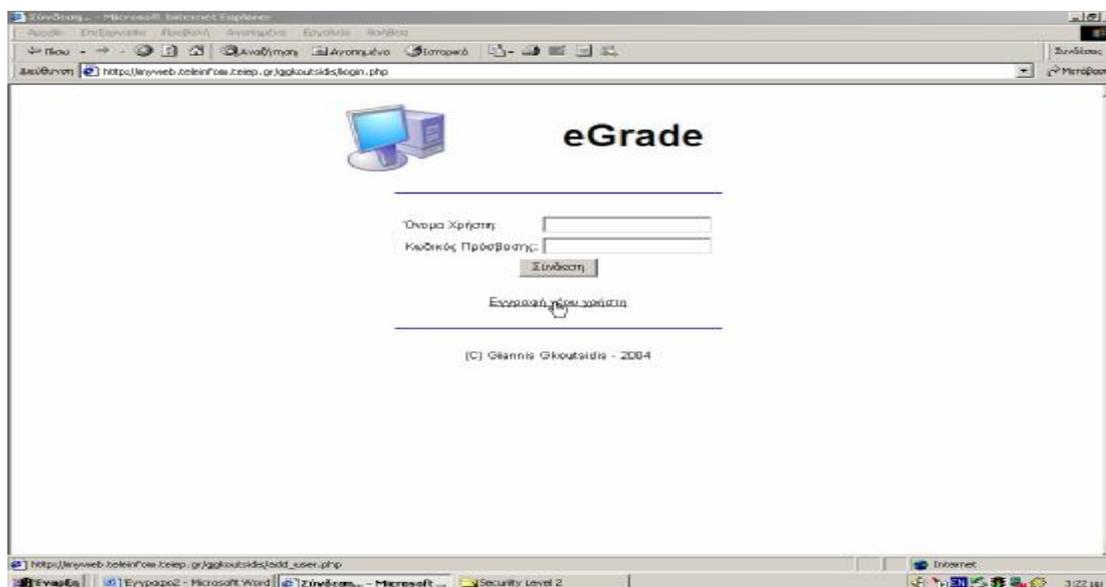
Με την επιλογή ενός χρήστη και την εντολή Διαγραφή, ο επιλεγμένος χρήστης διαγράφεται από το σύστημα, καθώς επίσης και οποιαδήποτε βαθμολογία έχει καταχωρηθεί για αυτόν.

Οι χρήστες που βρίσκονται στην κατηγορία Διαχειριστές, μπορούν να έχουν στη διάθεσή τους στοιχεία, τα οποία τους παρέχουν πρόσβαση στο Σύστημα Διαχείρισης Βάσεων Δεδομένων και κατ' επέκταση στη Βάση Δεδομένων του συστήματος eGrade, οπότε μπορούν να ολοκληρώσουν εργασίες, οι οποίες δεν είναι άμεσα διαθέσιμες από τις σελίδες του συστήματος, όπως π.χ. η μεταβολή των καταχωρημένων μαθημάτων ενός διδασκόμενου κ.λπ.

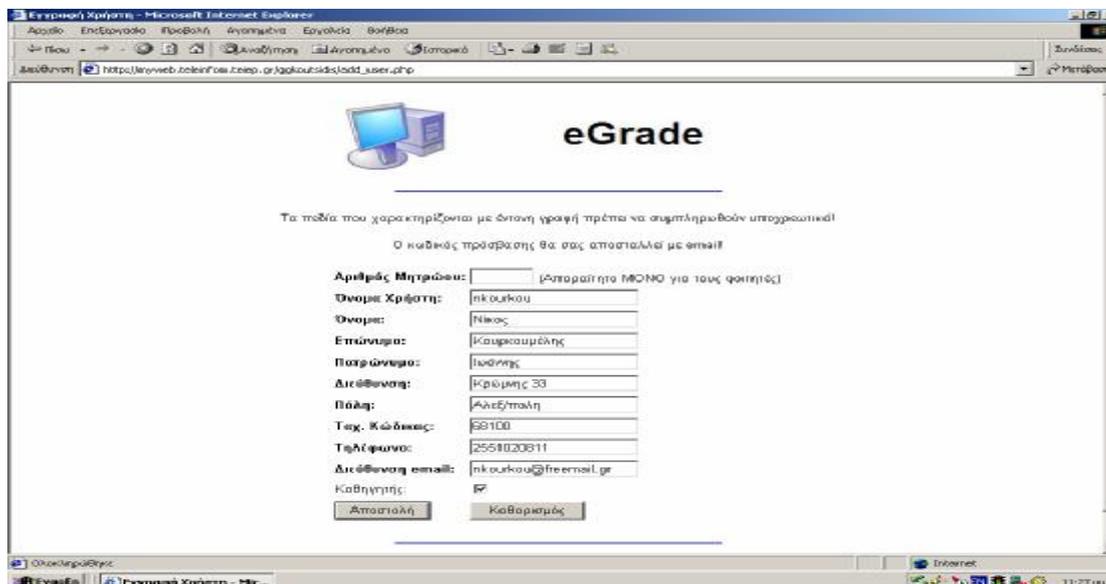
3.5. Βήμα προς Βήμα - Διδάσκοντας

Με την ολοκλήρωση της δημιουργίας των διάφορων Τμημάτων ενός Εκπαιδευτικού Ιδρύματος, το σύστημα είναι σε θέση να «υποδεχτεί» τους διάφορους διδάσκοντες των τμημάτων αυτών, οι οποίοι είναι επιφορτισμένοι, σε πρώτη φάση, με τη διαδικασία δημιουργίας των διδασκόμενων μαθημάτων, και σε τελική φάση με την καταχώρηση των βαθμολογιών.

Η διαδικασία εγγραφής ενός διδάσκοντα πραγματοποιείται μέσω της επιλογής Νέος Χρήστης από τη σελίδα σύνδεσης του συστήματος, όπως φαίνεται παρακάτω.



Σχήμα 3.5.1. Μετάβαση στη σελίδα δημιουργίας νέου χρήστη



Σχήμα 3.5.2. Εισαγωγή των στοιχείων του νέου χρήστη

Με την εισαγωγή των προσωπικών στοιχείων του χρήστη και την ενεργοποίηση της επιλογής Καθηγητής, μιας και ο νέος αυτός χρήστης είναι διδάσκοντας, πραγματοποιείται η διαδικασία αποστολής των στοιχείων αυτών προς το σύστημα.

Παρ.: Όπως αναφέρεται ρητά και μέσα από τη σελίδα δημιουργίας του νέου χρήστη, όταν αυτός αποτελεί διδάσκοντα δεν απαιτείται να συμπληρωθεί το πεδίο Αριθμός Μητρώου, μιας και δεν είναι απαραίτητο από το σύστημα.

Με την αποστολή των στοιχείων στο σύστημα, ο χρήστης ενημερώνεται για την ολοκλήρωση της διαδικασίας.

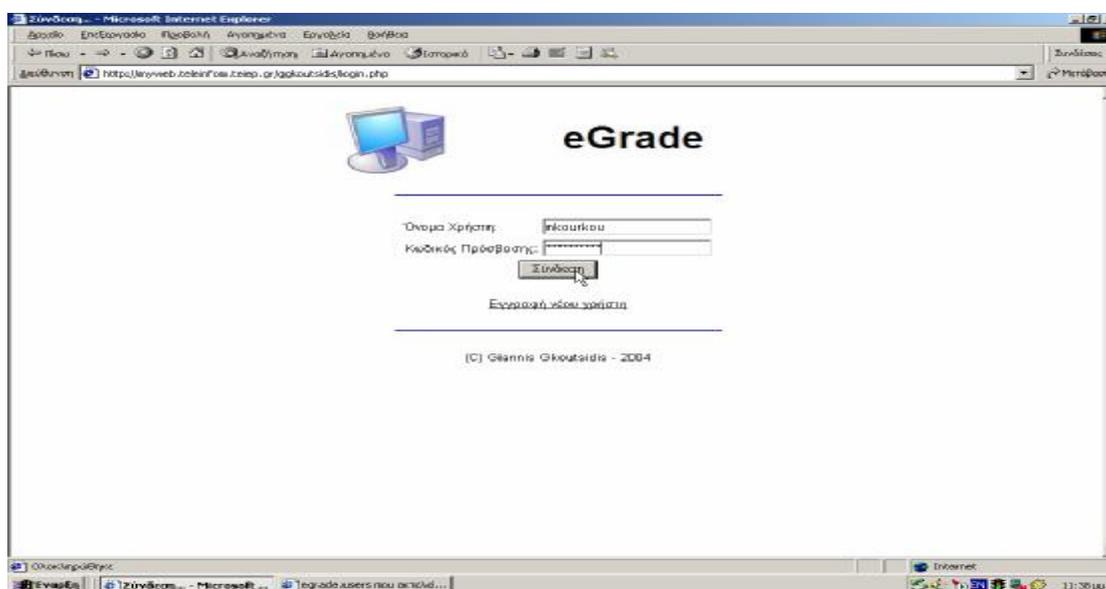


Σχήμα 3.5.3. Ολοκλήρωση της διαδικασίας εγγραφής νέου χρήστη

Ο χρήστης πλέον καλείται να ελέγξει τη δηλωθείσα διεύθυνση e-mail, ώστε να λάβει τον αρχικό Κωδικό Πρόσβασης για τη σύνδεσή του στο σύστημα.

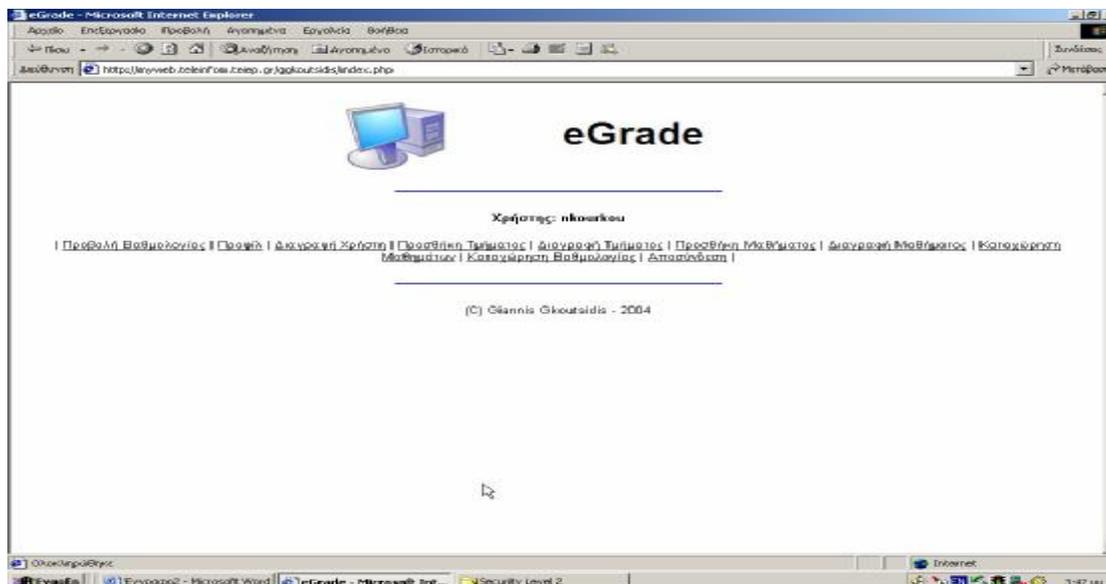
Παρ.: Οι λειτουργίες που έχει στη διάθεσή του ένας διδάσκοντας δεν γίνονται άμεσα διαθέσιμες, μιας και η αίτησή του πρέπει να ελεγχθεί από έναν Διαχειριστή, ώστε να πιστοποιηθεί η ταυτότητα του νέου χρήστη, ο οποίος τελικά μεταβαίνει στην ομάδα χρηστών Διδάσκοντες. Μέχρι να ολοκληρωθεί η προαναφερθείσα διαδικασία το σύστημα αναγνωρίζει το νέο αυτόν χρήστη ως διδασκόμενο.

Με την ολοκλήρωση της μετάβασης του χρήστη στην ομάδα Διδάσκοντες, ενεργοποιούνται όλες οι διαθέσιμες λειτουργίες της εν λόγω ομάδας και έτσι ο χρήστης μπορεί να συνδεθεί στο σύστημα.



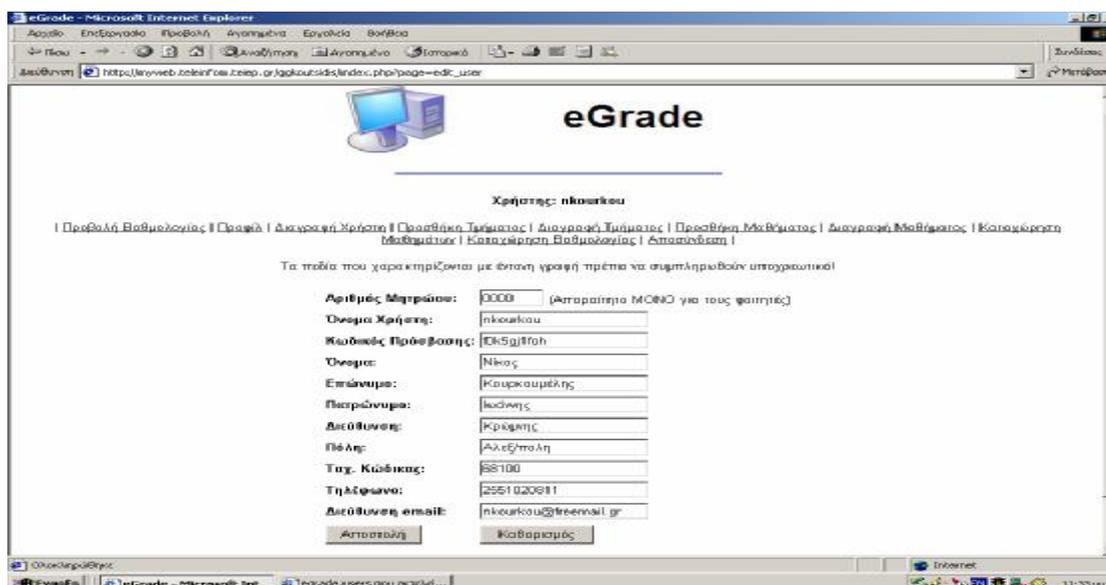
Σχήμα 3.5.4. Σύνδεση διδάσκοντα στο σύστημα

Μετά την επιτυχή σύνδεση του χρήστη-διδάσκοντα στο σύστημα, ο τελευταίος μεταφέρεται στην κεντρική σελίδα επιλογών του συστήματος.



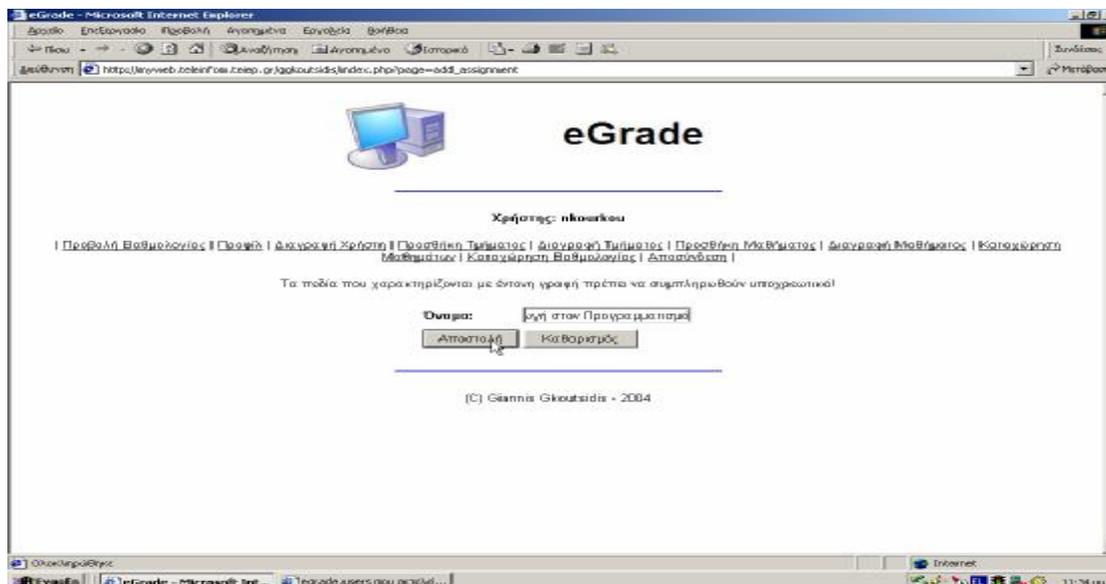
Σχήμα 3.5.5. Κεντρική σελίδα συστήματος

Ο χρήστης, με τη βοήθεια των εντολών του μενού, μπορεί να μεταβάλλει τα προσωπικά του στοιχεία από την επιλογή Προφίλ.



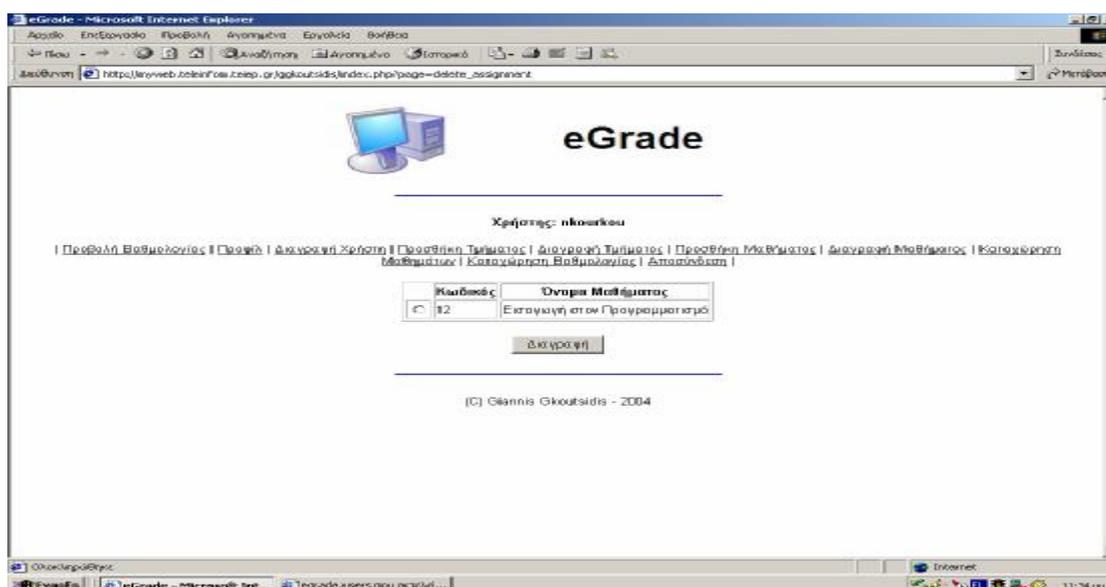
Σχήμα 3.5.6. Μεταβολή προσωπικών στοιχείων χρήστη

Βασικό μέλημα ενός χρήστη-διδάσκοντα αποτελεί η δημιουργία/διαγραφή των διδασκόμενων μαθημάτων για τα οποία είναι υπεύθυνος.



Σχήμα 3.5.7. Δημιουργία νέου μαθήματος

Από το μενού επιλογών και την εντολή Προσθήκη Μαθήματος ο χρήστης μπορεί να εισάγει το όνομα για το νέο μάθημα και να αποστείλει τα στοιχεία αυτά προς το σύστημα. Με τον τρόπο αυτό πραγματοποιείται η δημιουργία του μαθήματος, το οποίο εν συνεχεία μπορούν να δηλώσουν-καταχωρήσουν οι διάφοροι διδασκόμενοι. Ένας διδάσκοντας βέβαια μπορεί κάλλιστα να διαγράψει κάποιο μάθημα.



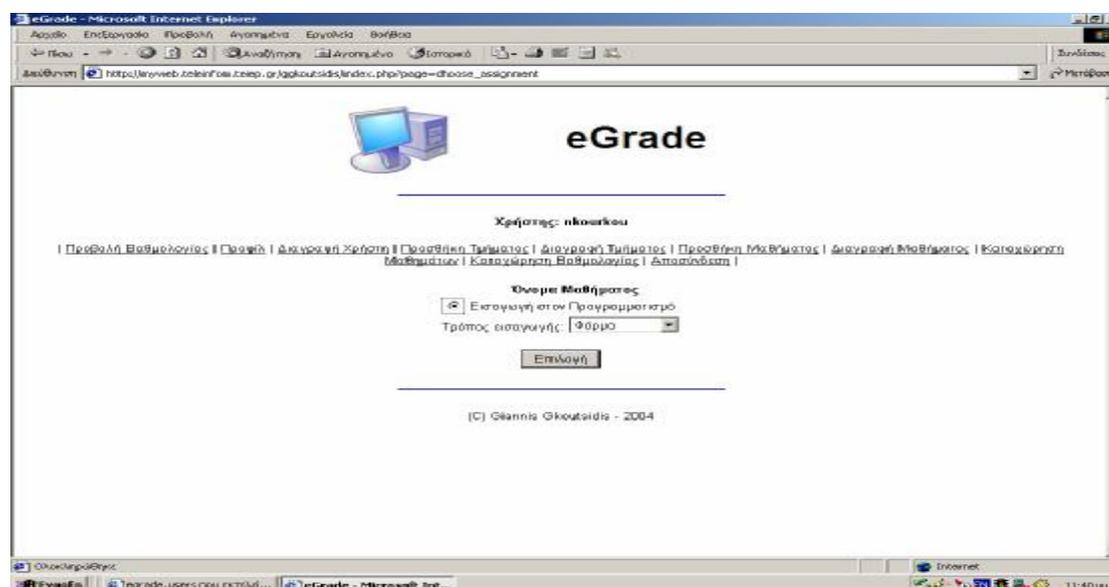
Σχήμα 3.5.8. Διαγραφή μαθήματος

Στην παρούσα σελίδα, στην οποία ο χρήστης μεταβαίνει με τη βοήθεια της εντολής Διαγραφή Μαθήματος, εμφανίζονται τα διαθέσιμα μαθήματα, τα οποία έχουν

καταχωρηθεί στο σύστημα. Ο χρήστης-διδάσκοντας έχει στη διάθεσή του τον Κωδικό καθώς επίσης και το Όνομα κάθε μαθήματος, το οποίο μπορεί και να διαγράψει.

Αφού ολοκληρωθεί η διαδικασία καταχώρησης των μαθημάτων από την πλευρά των διδασκόμενων, οι χρήστες-διδάσκοντες είναι πλέον σε θέση να καταχωρήσουν τις αντίστοιχες βαθμολογίες. Η όλη διαδικασία μπορεί να πραγματοποιηθεί με δύο τρόπους, οι οποίοι βέβαια για το σύστημα έχουν το ίδιο ακριβώς αποτέλεσμα. Η διαφοροποίηση των δύο μεθόδων αποστολής βαθμολογίας έγκειται στον τρόπο με τον οποίο ο κάθε διδάσκοντας έχει οργανωμένα τα βαθμολογία του.

Κοινό σημείο των δύο αυτών μεθόδων αποτελεί το γεγονός ότι ο διδάσκοντας πρέπει απαραίτητα να επιλέξει το μάθημα για το οποίο πρόκειται να γίνει η αποστολή βαθμολογίας.



Σχήμα 3.5.9. Επιλογή μαθήματος για αποστολή βαθμολογίας

Με τη βοήθεια της εντολής Καταχώρηση Βαθμολογίας, ο χρήστης μεταβαίνει στη σελίδα επιλογής αφενός μεν του μαθήματος για το οποίο θα πραγματοποιηθεί η αποστολή της βαθμολογίας, αφετέρου δε της μεθόδου της αποστολής.

Οι διαθέσιμες επιλογές είναι οι εξής:

- Φόρμα

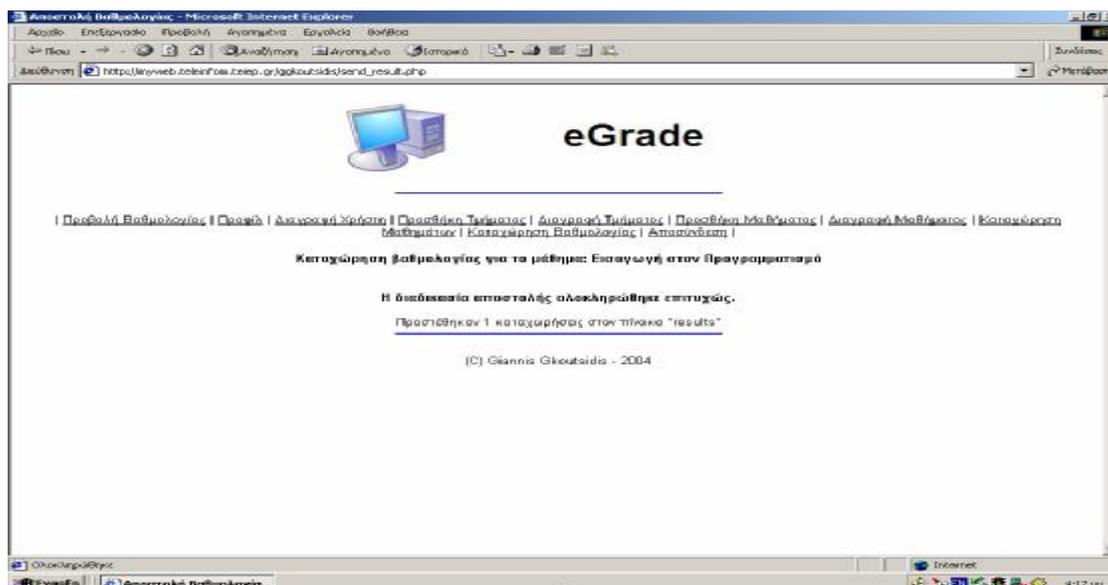


Σχήμα 3.5.10. Αποστολή βαθμολογίας με χρήση φόρμας

Με τη συγκεκριμένη μέθοδο, στην οθόνη του χρήστη-διδάσκοντα, εμφανίζονται όλοι οι διδασκόμενοι, οι οποίοι έχουν δηλώσει-καταχωρήσει το επιλεγμένο μάθημα και είτε δεν έχουν εξεταστεί ποτέ, είτε έχουν βαθμολογία σε προηγούμενη εξέταση, μικρότερη του 5,00. Ο χρήστης-διδάσκοντας έχει στη διάθεσή του τον κωδικό καθώς επίσης και το ονοματεπώνυμο του κάθε διδασκόμενου για το οποίο υποχρεούται να συμπληρώσει την αντίστοιχη βαθμολογία στο ανάλογο πεδίο. Έπεται η αποστολή της εν λόγω φόρμας δεδομένων προς το σύστημα, το οποίο είναι σε θέση να καταχωρήσει τις βαθμολογίες αυτές στη βάση δεδομένων, ώστε να ενημερώνεται για αυτές ο εκάστοτε διδασκόμενος.

- Αρχείο Excel

σημείο έχει ολοκληρωθεί η διαδικασία ελέγχου για την ορθότητα του αρχείου, καθώς επίσης και των δεδομένων που αυτό περιέχει. Ο χρήστης είναι πλέον σε θέση να αποστείλει τα δεδομένα στο σύστημα.

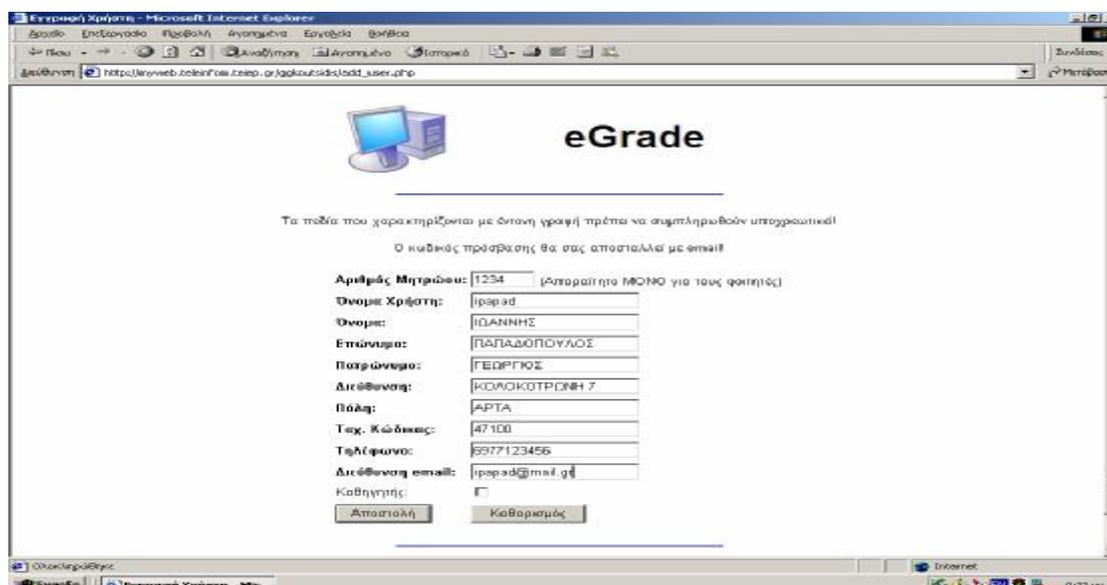


Σχήμα 3.5.13. Ολοκλήρωση της αποστολής βαθμολογίας

Με την ολοκλήρωση της διαδικασίας αποστολής του αρχείου Excel, ο χρήστης-διδάσκοντας, ενημερώνεται για το πλήθος των καταχωρήσεων που έχουν πραγματοποιηθεί στη βάση δεδομένων του συστήματος, οπότε και έχει ολοκληρωθεί η όλη διαδικασία.

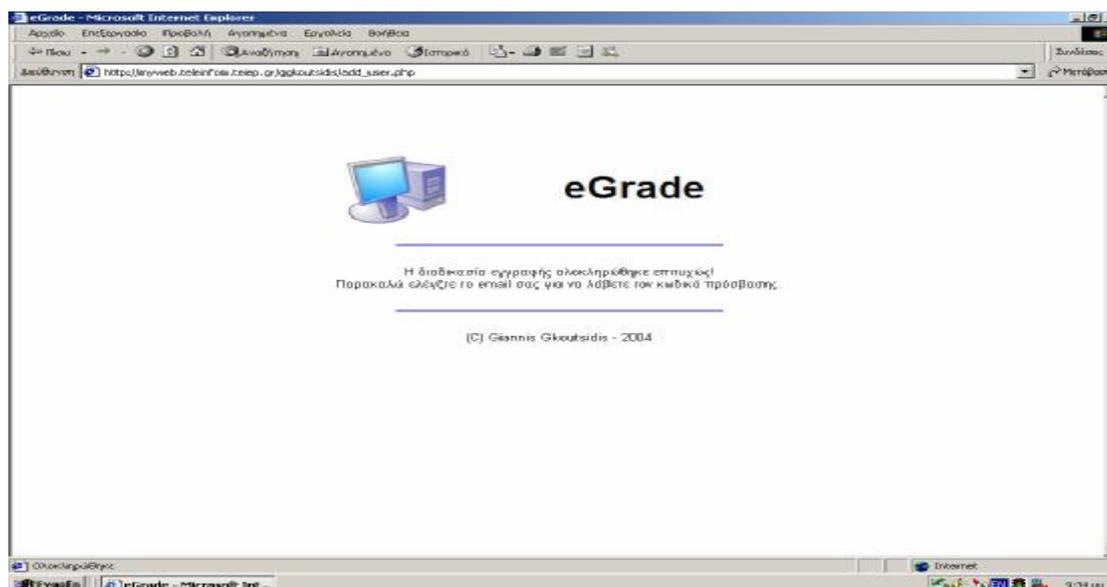
3.6. Βήμα προς Βήμα - Διδασκόμενος

Αφού ολοκληρωθούν οι διαδικασίες δημιουργίας των Τμημάτων από τους Διαχειριστές και των Μαθημάτων από τους Διδάσκοντες, οι Διδασκόμενοι είναι πλέον σε θέση να καταχωρήσουν τα μαθήματα για τα οποία επιθυμούν να ενημερώνονται σχετικά με τη βαθμολογία τους. Απαραίτητη προϋπόθεση βέβαια αποτελεί η ολοκλήρωση της διαδικασίας εγγραφής τους στο σύστημα.



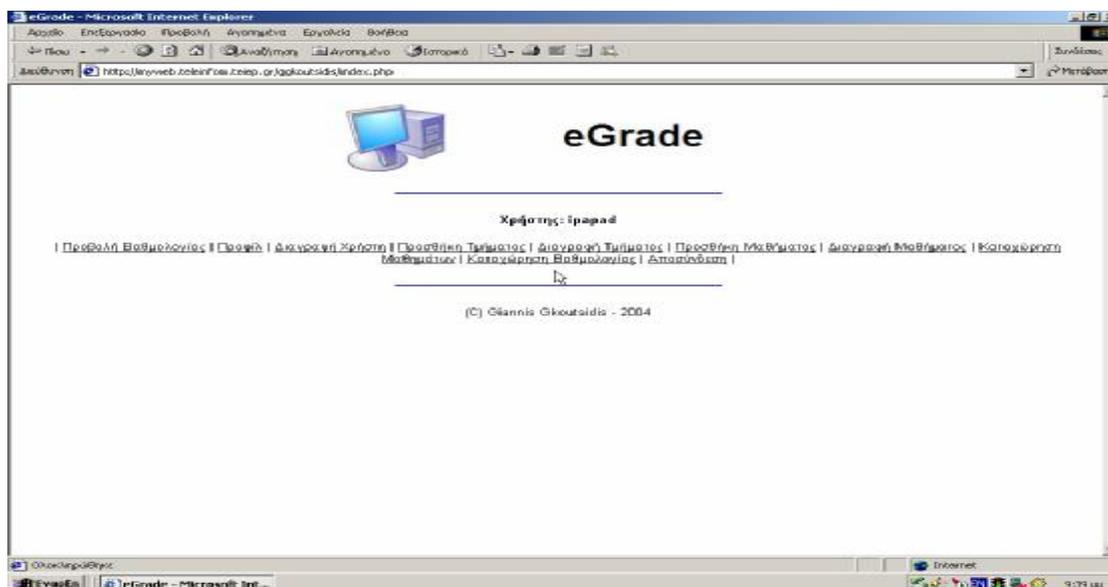
Σχήμα 3.6.1. Συμπλήρωση των προσωπικών στοιχείων του χρήστη-διδασκόμενου

Ο νέος χρήστης καλείται να συμπληρώσει πέραν των προσωπικών του στοιχείων και τον Αριθμό Μητρώου του. Αυτό είναι απαραίτητο για την ορθή ολοκλήρωση της αποστολής βαθμολογίας με τη χρήση αρχείου Excel. Ο χρήστης-διδασκόμενος δεν καλείται να ενεργοποιήσει την επιλογή Καθηγητής, μιας και δεν απαιτείται.



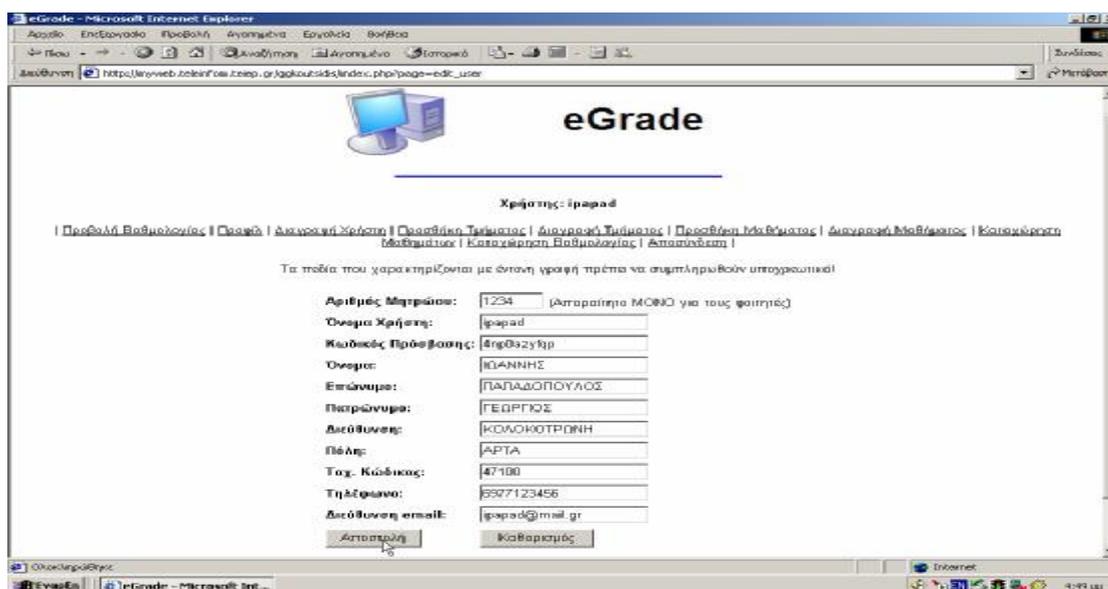
Σχήμα 3.6.2. Ολοκλήρωση εγγραφής χρήστη-διδασκόμενου

Μετά την αποστολή των στοιχείων του, ο χρήστης καλείται να ελέγξει τη δηλωθείσα διεύθυνση e-mail, ώστε να λάβει τον αρχικό κωδικό πρόσβασης που απαιτείται για τη σύνδεσή του στο σύστημα.



Σχήμα 3.6.3. Κεντρική οθόνη επιλογών συστήματος

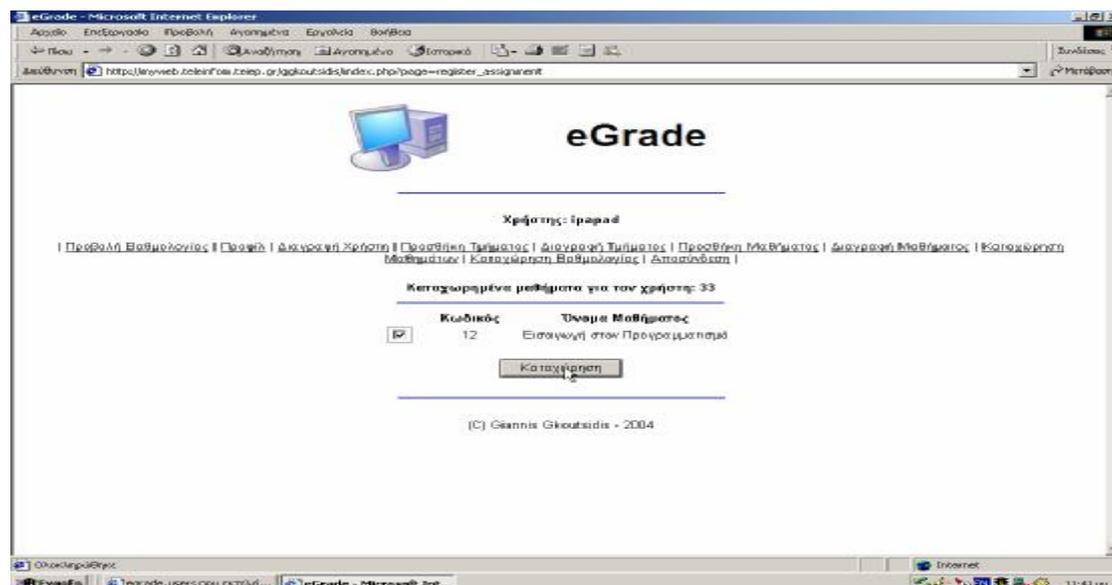
Αφού ο χρήστης-διδασκόμενος πιστοποιηθεί από το σύστημα μέσω της οθόνης σύνδεσης, μεταβαίνει στην κεντρική σελίδα επιλογών του συστήματος.



Σχήμα 3.6.4. Μεταβολή των προσωπικών στοιχείων του χρήστη-διδασκόμενου

Από αυτό το σημείο ο χρήστης-διδασκόμενος μπορεί να μεταβάλει τα προσωπικά του στοιχεία με τη βοήθεια της εντολής Προφίλ (όπως συμβαίνει με όλους τους χρήστες του συστήματος).

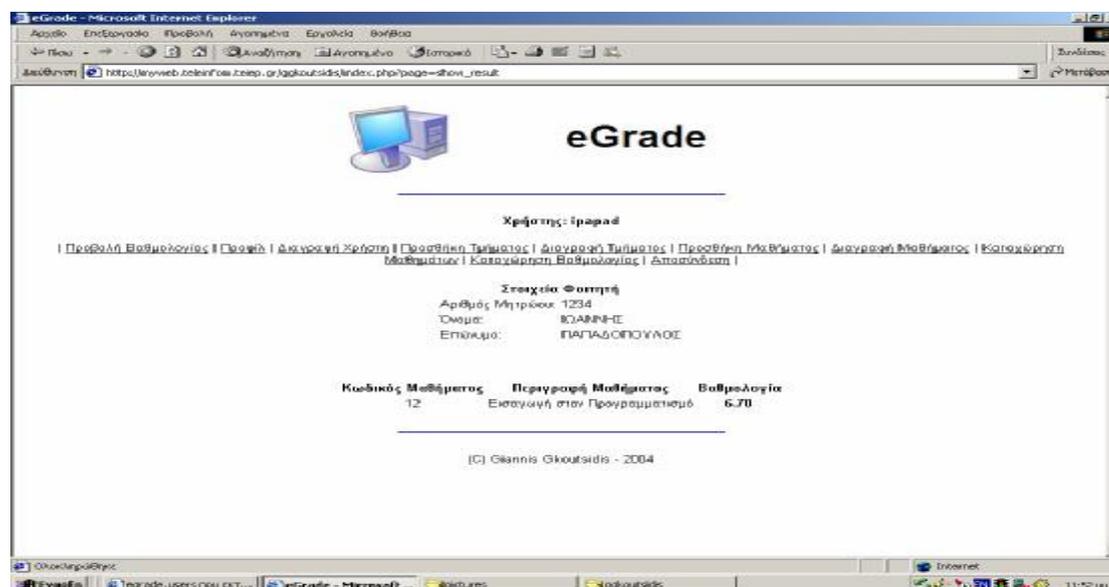
Με τη χρήση της εντολής Καταχώρηση Μαθημάτων ο χρήστης-διδασκόμενος μεταβαίνει στη σελίδα καταχώρησης των επιθυμητών μαθημάτων για τα οποία θέλει να ενημερώνεται.



Σχήμα 3.6.5. Καταχώρηση των προς παρακολούθηση της βαθμολογίας μαθημάτων

Ο χρήστης-διδασκόμενος καλείται να επιλέξει τα μαθήματα για τα οποία απαιτεί την ενημέρωσή του αναφορικά με τις βαθμολογίες. Δίνεται η δυνατότητα πολλαπλής επιλογής μαθημάτων. Σε περίπτωση που κάποιος διδασκόμενος κάνει μια δεύτερη σύνδεση στο σύστημα και εκκινήσει τη διαδικασία καταχώρησης μαθημάτων για ακόμη μια φορά, τότε η λίστα των προς καταχώρηση μαθημάτων που θα εμφανιστεί δεν θα περιλαμβάνει τα ήδη δηλωθέντα μαθήματα.

Με την ολοκλήρωση και της παραπάνω διαδικασίας ο χρήστης-διδασκόμενος είναι πλέον σε θέση να ενημερωθεί για τις βαθμολογίες των μαθημάτων που έχει καταχωρήσει, αρκεί βέβαια να έχουν πραγματοποιηθεί οι αντίστοιχες καταχωρήσεις από πλευράς διδασκόντων.

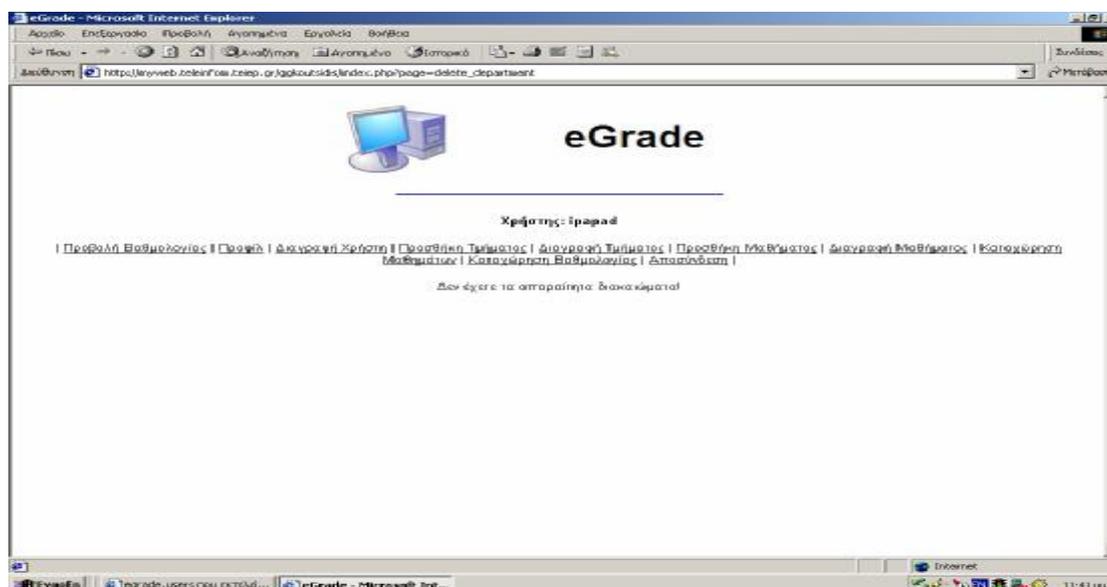


Σχήμα 3.6.6. Παρουσίαση βαθμολογίας χρήστη-διδασκόμενου

Με τη εντολή Προβολή Βαθμολογίας ο χρήστης-διδασκόμενος μεταβαίνει στη σελίδα παρουσίασης των βαθμολογιών του. Στην εν λόγω σελίδα γίνονται γνωστά ο Αριθμός Μητρώου, το Όνομα και το Επώνυμο του διδασκόμενου, καθώς επίσης και οι βαθμολογίες στα καταχωρημένα μαθήματά του. Αν ο διδασκόμενος δεν έχει εξεταστεί σε κάποιο μάθημα που έχει δηλώσει πριν θα λάβει την ενδεικτική βαθμολογία 0,00, για το συγκεκριμένο μάθημα.

Παρ.: Ο εκάστοτε διδασκόμενος διαθέτει πρόσβαση στα δικά του καταχωρημένα μαθήματα και μόνο σε αυτά. Η όλη διαδικασία ολοκληρώνεται με τη χρήση κατάλληλων sessions, τα οποία διαχειρίζονται από το σύστημα.

Ο χρήστης-διδασκόμενος επιπλέον δεν έχει δικαιώματα σε λειτουργίες όπως π.χ. Διαγραφή Μαθήματος, όπως φαίνεται και στο σχ. 25.



Σχήμα 3.6.7. Απόρριψη χρήστη-διδασκόμενου από τη διαγραφή μαθήματος

ΕΠΙΛΟΓΟΣ

Η ηλεκτρονική εφαρμογή βαθμολογίας eGrade δεν αποτελεί εμπορικό προϊόν στην παρούσα έκδοσή της. Πέραν αυτού υπάρχει μια πληθώρα βελτιώσεων, η υλοποίηση/εφαρμογή των οποίων μπορούν να προσδώσουν μια εντελώς διαφορετική δυναμική στην όλη εφαρμογή. Η βελτιώσεις αυτές αφορούν όχι μόνον την εφαρμογή αυτή καθ' αυτή (αναφερόμενοι κυρίως στις διάφορες λειτουργίες που αυτή εξυπηρετεί), αλλά και στις λειτουργίες εγκατάστασης και συντήρησης ενός τέτοιου συστήματος.

Όσον αφορά τις λειτουργίες εγκατάστασης και συντήρησης της εν λόγω εφαρμογής, θα μπορούσαν να υλοποιηθούν οι εξής βελτιώσεις:

1. Εργαλείο εγκατάστασης της εφαρμογής

Το συγκεκριμένο εργαλείο-εφαρμογή θα διευκολύνει το χρήστη-διαχειριστή, ώστε να ολοκληρώσει επιτυχώς την αρχική εγκατάσταση του συστήματος, αλλά και την παραμετροποίηση των πρώτων ρυθμίσεων αυτού. Πρέπει βέβαια να επισημανθεί το γεγονός ότι η εφαρμογή ακολουθεί cross-platform αρχιτεκτονική, η οποία εμπεριέχει και την πιθανότητα εγκατάστασης της εφαρμογής σε servers με διαφορετικά λειτουργικά συστήματα. Δεδομένου του γεγονότος αυτού, γίνεται αντιληπτό ότι και το εργαλείο της εγκατάστασης θα πρέπει να ακολουθεί μια λογική, η οποία να του προσδίδει τη δυνατότητα ευελιξίας, όταν αυτό εγκαθίσταται σε servers που εκτελούν Windows, Linux κ.λπ.

2. Εργαλείο λήψης/επαναφοράς αντιγράφων ασφαλείας δεδομένων

Με τη βοήθεια του εργαλείου αυτού θα πραγματοποιείται λήψη και αντίστοιχα επαναφορά δεδομένων, τα οποία προέρχονται από τη βάση δεδομένων. Τέτοιου είδους λειτουργίες βέβαια υλοποιούνται και από τις διάφορες εφαρμογές διαχείρισης βάσεων δεδομένων, όπως π.χ. το EMS MySQL Manager, το οποίο απευθύνεται στο σύστημα διαχείρισης βάσεων δεδομένων της MySQL (βλ. <http://www.mysqlmanager.com>). Η ενοποίηση μιας τέτοιας λειτουργίας όμως με το όλο σύστημα θα κάνει τις εν λόγω λειτουργίες ακόμα πιο αποδοτικές και ευέλικτες, μιας και θα εκτελούνται στο

σύνολό τους από το περιβάλλον διεπαφής της εφαρμογής και όχι με τη χρήση εργαλείων τρίτων κατασκευαστών.

Αναφορικά με τις εσωτερικές λειτουργίες της εφαρμογής και κατ' επέκταση την όλη λειτουργία της, θα μπορούσαν να εφαρμοστούν τα εξής:

1. Εισαγωγή του κύκλου σπουδών

Η δυνατότητα εισαγωγής του κύκλου σπουδών και των διάφορων μαθημάτων-αλυσίδων του εκάστοτε τμήματος. Αυτό θα είχε ως αποτέλεσμα τη δημιουργία ενός πιο αποδοτικού συστήματος, αφού κατά τη δήλωση των μαθημάτων από το διδασκόμενο, ο τελευταίος θα μπορούσε να ενημερώνεται αυτόματα για το ποια μαθήματα μπορεί ή όχι να δηλώσει και κατ' επέκταση να παρακολουθήσει.

2. Χρήση γραφημάτων

Η χρήση γραφημάτων αποτελεί μια αρκετά σημαντική βελτίωση για την εφαρμογή, μιας και η εμφάνιση συγκεντρωτικών στοιχείων αναφορικά με τις βαθμολογίες των διδασκομένων, σε συνδυασμό με την εφαρμογή κατάλληλων φίλτρων για τις διάφορες αναζητήσεις, αποτελεί την πλέον αποδοτική λύση για την παρακολούθηση της συνολικής προόδου.

3. Ενότητα: Πτυχιακές Εργασίες

Θα μπορούσε να δημιουργηθεί μια ξεχωριστή ενότητα μαθημάτων, η οποία θα καλύπτει τις Πτυχιακές Εργασίες. Στην ενότητα αυτή κάθε διδασκόμενος θα μπορεί να εισάγει τα στοιχεία της εργασίας του (Θέμα, Επιβλέπον Καθηγητής, Πρόοδος, Βαθμολογία), ώστε να παρακολουθείται πιο διεξοδικά η διαδικασία περάτωσης της πτυχιακής εργασίας.

ΠΑΡΑΡΤΗΜΑ

1. Κώδικας της Εφαρμογής

- **add_assignment.php**

```

<?
include_once ("egrade_fns.php");
function print_add_assignment()
{
echo "<p align=center>Τα πεδία που χαρακτηρίζονται με έντονη γραφή πρέπει να
συμπληρωθούν υποχρεωτικά!</p>";
echo "<form action=\"add_assignment.php\" method=\"POST\">";
echo "<table align=center>";
echo "<tr><td><strong>Όνομα:</strong></td><td><input type=\"text\" name=\"name\"
maxlength=\"60\"></td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"posted\" value=\"1\"><td></tr>";
echo "<tr><td><input type=\"submit\" name=\"submit\"
value=\"Αποστολή\"></td><td><input type=\"reset\" name=\"reset\"
value=\"Καθαρισμός\"></td></tr>";
echo "</table>";
echo "</form>";
html_footer();
}
function add_assignment()
{
if ((!isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_add_assignment();
exit();
}
if (!filled_out($_POST))
{
echo "Δεν έχουν συμπληρωθεί όλα τα απαραίτητα στοιχεία!";
print_add_assignment();
exit();
}
$name=$_POST['name'];
$class=$_POST['class'];
$name=stripslashes($name);
$result=mysql_query("insert into assignments (assignment_name) values ('$name')");
if (!$result)
{
echo "Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_error();
exit();
}
else
{
header ("Location:index.php");
}
}
}
@session_start();
check_security(2);

```

```
add_assignment();
?>
```

- **add_department.php**

```
<?
include_once ("egrade_fns.php");
function print_add_department()
{
echo "<p align=center>Τα πεδία που χαρακτηρίζονται με έντονη γραφή πρέπει να
συμπληρωθούν υποχρεωτικά!</p>";
echo "<form action=\"add_department.php\" method=\"POST\">";
echo "<table align=center>";
echo "<tr><td><strong>Όνομα:</strong></td><td><input type=\"text\" name=\"name\"
maxlength=\"60\"></td></tr>";
echo "<tr><td><strong>Προϊστάμενος:</strong></td><td><input type=\"text\"
name=\"manager\" maxlength=\"60\"></td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"posted\" value=\"1\"><td></tr>";
echo "<tr><td><input type=\"submit\" name=\"submit\"
value=\"Αποστολή\"></td><td><input type=\"reset\" name=\"reset\"
value=\"Καθαρισμός\"></td></tr>";
echo "</table>";
echo "</form>";
html_footer();
}
function add_department()
{
if ((!isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_add_department();
exit;
}
if (!filled_out($_POST))
{
echo "Δεν έχουν συμπληρωθεί όλα τα απαραίτητα στοιχεία!";
print_add_department();
exit();
}
$name=$_POST['name'];
$manager=$_POST['manager'];
$name=stripslashes($name);
$manager=stripslashes($manager);
$sql_name_check=mysql_query("select department_name from departments where
department_name='$name'");
$name_check=mysql_num_rows($sql_name_check);
if ($name_check>0)
{
echo "Το όνομα τμήματος χρησιμοποιείται από άλλον τμήμα! Παρακαλώ εισάγετε
διαφορετικά στοιχεία!";
unset($name);
print_add_department();
exit();
}
$result=mysql_query("insert into departments (department_name,department_manager)
values ('$name','$manager)");
if (!$result)
{
echo "Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_error();
exit();
}
}
```

```

else
{
header ("Location:index.php");
}
}
@session_start();
check_security(3);
add_department();
?>

```

- **add_result.php**

```

<?
include_once ("egrade_fns.php");
function print_add_result()
{
$lessonid=$_SESSION['lesson_id'];
$sql="select assignment_name from assignments where assignment_id=$lessonid";
$result=mysql_query($sql);
if (!$result)
{
die ("Σφάλμα κατά την εκτέλεση του ερωτήματος");
}
$line=mysql_fetch_array($result,MYSQL_BOTH);
$lessonname=$line['assignment_name'];
echo "<p align=center><strong>Καταχώρηση βαθμολογίας για το μάθημα:
$lessonname</strong></p>";
if ($_REQUEST['grademethod']==0)
{
echo "<form action=\"add_result.php\" method=\"POST\">";
echo "<table align=center border=0>";
$sql="select * from studentsassignments,users,results where ";
$sql.="studentsassignments.studentassignment_assignmentid=$lessonid and ";
$sql.="users.user_id=studentsassignments.studentassignment_userid and ";
$sql.="results.result_assignmentid=studentsassignments.studentassignment_assignmentid and ";
$sql.="results.result_userid=users.user_id and ";
$sql.="results.result_grade<5.00"; // Καλά ε, το έστειλα το μηχανημα!!
$result=mysql_query($sql);
if (!$result)
{
die ("Σφάλμα κατά την εκτέλεση του ερωτήματος");
}
$numrows=mysql_num_rows($result);
if ($numrows>0)
{
echo "<tr>";
echo "<th>Κωδικός</th>";
echo "<th>Όνοματεπώνυμο</th>";
echo "<th>Βαθμολογία</th>";
echo "</tr>";

while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$id=$line['user_id'];
$name=$line['user_firstname']." ".$line['user_lastname'];
echo "<tr>";
echo "<td align=center>$id</td>";
echo "<td align=center>$name</td>";
echo "<td align=center><input type=\"text\" name=\"grade[]\" maxlength=\"5\" size=\"5\"</td>";
echo "<tr><td><input type=\"hidden\" name=\"id[]\" value=\"$id\"><td></tr>";
echo "</tr>";
}
}
}

```

```

}
echo "<tr><td><input type=\"hidden\" name=\"posted\" value=\"1\"><td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"numrows\" value=\"$numrows\"><td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"lessonid\" value=\"$lessonid\"><td></tr>";
echo "<tr><td colspan=3 align=center><input type=\"submit\" name=\"submit\"
value=\"Αποστολή\"></td></tr>";
echo "</table>";
echo "</form>";
}
else
{
echo "<p align=center>Δεν υπάρχουν φοιτητές που να έχουν καταχωρήσει το επιλεγμένο μάθημα
ή όλοι οι φοιτητές έχουν βαθμολογία μεγαλύτερη του 5.00!</p>";
}
}
else
{
header("Location:send_result.php");
}
html_footer();
}
function add_result()
{
if ((isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_add_result();
exit();
}
foreach ($_POST['grade'] as $value)
{
if ((!is_numeric(str_replace(",",".", $value))) || (str_replace(",",".", $value)>10))
{
html_header("Μη έγκυρες βαθμολογίες");
print_menu();
echo "<p align=center>Οι βαθμολογίες δεν είναι έγκυρες!</p>";
html_footer();
exit();
}
}
for ($i=0;$i<$_POST['numrows'];$i++)
{
$lessonid=$_SESSION['lesson_id'];
$id=$_POST['id'][$i];
$grade=str_replace(",",".", $_POST['grade'][$i]);
$sql="update results set result_grade='$grade' where result_userid='$id' and
result_assignmentid='$lessonid'";
$result=mysql_query($sql);
}
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
else {header ("Location:index.php");}
}
@session_start();
check_security(2);
add_result();
?>

```

- **add_user.php**

```
<html>
```

```

<head>
<script Language="JavaScript">
// Author: JS-Examples - http://www.js-examples.com
// Courtesy of SimplytheBest.net - http://simplythebest.net/scripts/
var _T = "locked";
var _F = "unlocked";
function lockIt(_P)
{
var _L = document.frmMyForm.lck.value;
document.frmMyForm.registernumber.disabled=(document.frmMyForm.lck.value==_L==_F)?_T:
_F==_T;
}
function isDis() { return (document.frmMyForm.lck.value==_T); }
</script>
</head>
<?
include_once ("egrade_fns.php");
function print_add_user()
{
html_header("Εγγραφή Χρήστη");
echo "<p align=center>Τα πεδία που χαρακτηρίζονται με έντονη γραφή πρέπει να συμπληρωθούν
υποχρεωτικά!</p>";
echo "<p align=center>Ο κωδικός πρόσβασης θα σας αποσταλλεί με email!</p>";
echo "<form id=\"frmMyform\" name=\"frmMyForm\" action=\"add_user.php\"
method=\"POST\">";
echo "<table align=center>";
echo "<tr><td><strong>Αριθμός Μητρώου:</strong></td><td><input type=\"text\"
id=\"registernumber\" name=\"registernumber\" name=\"\" onfocus=\"if(isDis())blur();\"
maxlength=\"4\" size=\"4\"> (Απαραίτητο ΜΟΝΟ για τους φοιτητές)</td></tr>";
echo "<tr><td><strong>Όνομα Χρήστη:</strong></td><td><input type=\"text\"
name=\"username\" maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Όνομα:</strong></td><td><input type=\"text\" name=\"firstname\"
maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Επώνυμο:</strong></td><td><input type=\"text\" name=\"lastname\"
maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Πατρώνυμο:</strong></td><td><input type=\"text\"
name=\"fathername\" maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Διεύθυνση:</strong></td><td><input type=\"text\" name=\"address\"
maxlength=\"60\"></td></tr>";
echo "<tr><td><strong>Πόλη:</strong></td><td><input type=\"text\" name=\"city\"
maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Ταχ. Κώδικας:</strong></td><td><input type=\"text\"
name=\"postalcode\" maxlength=\"6\"></td></tr>";
echo "<tr><td><strong>Τηλέφωνο:</strong></td><td><input type=\"text\" name=\"phone\"
maxlength=\"10\"></td></tr>";
echo "<td><strong>Διεύθυνση email:</strong></td><td><input type=\"text\" name=\"email\"
maxlength=\"60\"></td></tr>";
echo "<tr><td><strong>Καθηγητής:</strong></td><td><input type=\"checkbox\" name=\"teacher\" value=\"1\"
onclick=\"lockIt(_T)\" id=\"teacher\"></td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"posted\" value=\"1\"><td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"lck\" value=\"unlocked\"><td></tr>";
echo "<tr><td><input type=\"submit\" name=\"submit\" value=\"Αποστολή\"></td><td><input
type=\"reset\" name=\"reset\" value=\"Καθαρισμός\"></td></tr>";
echo "</table>";
echo "</form>";
html_footer();
}
function add_user()
{
// Έλεγχος αποστολής φόρμας

```

```

if ((isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_add_user();
exit;
}
// Έλεγχος συμπληρωμένων πεδίων
if (!filled_out($_POST))
{
echo "Δεν έχουν συμπληρωθεί όλα τα απαραίτητα στοιχεία!";
print_add_user();
exit();
}
// Έλεγχος εγκυρότητας email
$expression = "^[A-Za-z0-9-]+@[A-Za-z-]+(\.[A-Za-z]+)(\.[A-Za-z]+)*$";
if (!ereg("$expression", $_POST['email']))
{
echo "Η διεύθυνση email δεν είναι έγκυρη";
print_add_user();
exit();
}
// Έλεγχος εγκυρότητας τηλεφώνου
if ((ereg("[a-zA-Zα-ωΑ-Ω]", $_POST['phone']) || (strlen($_POST['phone'])!=10)))
{
echo "Ο αριθμός τηλεφώνου δεν είναι έγκυρος";
print_add_user();
exit();
}
if (isset($_POST['teacher']))
{
$notes='Αίτηση για αύξηση επιπέδου πρόσβασης σε 2';
}
// Εκχώρηση συμπληρωμένων πεδίων σε μεταβλητές
$registernumber=$_POST['registernumber'];
$username=$_POST['username'];
$firstname=$_POST['firstname'];
$lastname=$_POST['lastname'];
$fathername=$_POST['fathername'];
$address=$_POST['address'];
$city=$_POST['city'];
$postalcode=$_POST['postalcode'];
$phone=$_POST['phone'];
$email=$_POST['email'];
$password=generate_password(10);
// Διαγραφή τυχόν εισηγμένων χαρακτήρων διαφυγής
$registernumber=stripslashes($registernumber);
$username=stripslashes($username);
$firstname=stripslashes($firstname);
$lastname=stripslashes($lastname);
$fathername=stripslashes($fathername);
$address=stripslashes($address);
$city=stripslashes($city);
$postalcode=stripslashes($postalcode);
$phone=stripslashes($phone);
$email=stripslashes($email);
// Έλεγχος μοναδικότητας εγγραφής
if ($registernumber)
{
$sql_registernumber_check=mysql_query("select user_registernumber from users where
user_registernumber='$registernumber'");
$registernumber_check=mysql_num_rows($sql_registernumber_check);
}

```

```

}
$sql_username_check=mysql_query("select user_username from users where
user_username='$username'");
$sql_email_check=mysql_query("select user_email from users where user_email='$email'");
$username_check=mysql_num_rows($sql_username_check);
$email_check=mysql_num_rows($sql_email_check);
if (($username_check>0) || ($email_check>0) || ($registernumber_check>0))
{
if ($username_check>0)
{
echo "Το όνομα χρήστη χρησιμοποιείται από άλλον χρήστη! Παρακαλώ εισάγετε διαφορετικά
στοιχεία!";
unset($username);
}
if ($email_check>0)
{
echo "Η διεύθυνση email χρησιμοποιείται από άλλον χρήστη! Παρακαλώ εισάγετε διαφορετικά
στοιχεία!";
unset($email);
}
if (($registernumber_check>0) && (!isset($_POST['teacher'])))
{
echo "Ο αριθμός μητρώου χρησιμοποιείται από άλλον χρήστη! Παρακαλώ εισάγετε διαφορετικά
στοιχεία!";
unset($registernumber);
}
print_add_user();
exit();
}
$result=mysql_query("insert into users
(user_registernumber,user_username,user_password,user_firstname,user_lastname,user_fathernam
e,user_address,user_city,user_postalcode,user_phone,user_email,user_notes) values
('$registernumber','$username','$password','$firstname','$lastname','$fathername','$address','$city',
'$postalcode','$phone','$email','$notes')");
if (!$result)
{
echo "Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_error();
}
else
{
ini_set("sendmail_from","webmaster@egrade.net");
$subject="Εγγραφή στο σύστημα eGrade";
$headers="";
$message="Αγαπητέ/ή $firstname,\r\n\r\n";
$message.="Η διαδικασία εγγραφής στο ηλεκτρονικό σύστημα βαθμολογίας eGrade
ολοκληρώθηκε επιτυχώς!\r\n\r\n";
$message.="Τα στοιχεία που απαιτούνται για τη σύνδεσή σας στο σύστημα eGrade είναι τα
εξής:\r\n";
$message.="Username:$username\r\n";
$message.="Password:$password\r\n";
$message.="Για να συνδεθείς στο σύστημα eGrade αντέγραψε το παρακάτω URL σε ένα νέο
παράθυρο του browser σου.\r\n";
$message.="http://www.worldautomatic.gr/ggkouts/login.php\r\n";
$message.="Είμαστε στη διάθεσή σας για οποιαδήποτε πληροφορία.\r\n\r\n";
$message.="Φίλικά,\r\n";
$headers.="From: webmaster@".$_SERVER['SERVER_NAME']."\r\n";
$headers.="Reply-To: webmaster@".$_SERVER['SERVER_NAME']."\r\n";
$headers.="X-Mailer: PHP/" .phpversion();
mail($email,$subject,$message,$headers);
html_header("eGrade");

```

```

echo "<p align=center>";
echo "Η διαδικασία εγγραφής ολοκληρώθηκε επιτυχώς!<br>";
echo "Παρακαλώ ελέγξτε το email σας για να λάβετε τον κωδικό πρόσβασης.";
echo "</p>";
html_footer();
}
}
add_user();
?>
</html>

```

- **choose_assignment.php**

```

<?
include_once ("egrade_fns.php");
function print_choose_assignment()
{
echo "<form action='choose_assignment.php' method='post'>";
echo "<table border=0 align=center>";
echo "<tr>";
echo "<th></th>";
echo "<th>Όνομα Μαθήματος</th>";
echo "</tr>";
$result=mysql_query("select * from assignments");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$id=$line['assignment_id'];
$name=$line['assignment_name'];
echo "<tr>";
echo "<td><input type='radio' name='radio' value=$id></td>";
echo "<td>$name</td>";
echo "</tr>";
}
echo "<tr><td colspan=2 align=center>";
echo "Τρόπος εισαγωγής: ";
echo "<select name='grade_method'>";
echo "<option value='0'>Φόρμα</option>";
echo "<option value='1'>Αρχείο Excel</option>";
echo "</select>";
echo "<input type='hidden' name='posted' value='1'>";
echo "</table>";
echo "<p align=center>";
echo "<td><input type='submit' name='submit' value='Επιλογή'></td>";
echo "</p>";
echo "</form>";
html_footer();
}
function choose_assignment()
{
if ((isset($_POST['posted'])) || ($_POST['posted']!=1)){print_choose_assignment();exit();}
if (!isset($_POST['radio']))
{
header("Location:index.php?page=choose_assignment");
exit();
}
$id=$_POST['radio'];
$_SESSION['lesson_id']=$id;
$method=$_POST['grade_method'];

```

```

header ("Location:index.php?page=add_result&lessonid=$id&grademethod=$method");
}
@session_start();
check_security(2);
choose_assignment();
?>

```

- **db_egrade.inc.php**

```

<?
$db_user = "ggkouts";
$db_pass = "852456";
$db_name = "egrade";
$db_host = "127.0.0.1";
$db_port = "3306";
$link=mysql_connect($db_host,$db_user,$db_pass);
if (!$link) {die ("Σφάλμα κατά τη σύνδεση με τον MySQL Server! ".mysql_error());}
$db=mysql_select_db($db_name,$link);
if (!$db) {die ("Σφάλμα κατά την επιλογή της βάση δεδομένων! ".mysql_error());}
?>

```

- **delete_assignment.php**

```

<?
include_once ("egrade_fns.php");
function print_delete_assignment()
{
echo "<form action=\"delete_assignment.php\" method=\"post\">";
echo "<table border=1 align=center>";
echo "<tr>";
echo "<th></th>";
echo "<th>Κωδικός</th>";
echo "<th>Όνομα Μαθήματος</th>";
echo "</tr>";
$result=mysql_query("select * from assignments");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$id=$line['assignment_id'];
$name=$line['assignment_name'];
echo "<tr>";
echo "<td><input type=\"radio\" name=\"radio\" value=$id></td>";
echo "<td>$id</td>";
echo "<td>$name</td>";
echo "</tr>";
}
echo "<input type=\"hidden\" name=\"posted\" value=\"1\">";
echo "</table>";
echo "<p align=center>";
echo "<td><input type=\"submit\" name=\"submit\" value=\"Διαγραφή\"></td>";
echo "</p>";
echo "</form>";
html_footer();
}
function delete_assignment()
{
if ((isset($_POST['posted'])) || ($_POST['posted']!=1)){print_delete_assignment();exit();}

```

```

if (!isset($_POST['radio']))
{
header("Location:index.php?page=delete_assignment");
exit();
}
$id=$_POST['radio'];
$result=mysql_query("delete from assignments where assignment_id=$id");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτημάτος!".mysql_error());}
header ("Location:index.php");
}
@session_start();
check_security(2);
delete_assignment();
?>

```

- **delete_department.php**

```

<?
include_once ("egrade_fns.php");
function print_delete_department()
{
echo "<form action=\"delete_department.php\" method=\"post\">";
echo "<table border=1 align=center>";
echo "<tr>";
echo "<th></th>";
echo "<th>Κωδικός</th>";
echo "<th>Όνομα Τμήματος</th>";
echo "<th>Προϊστάμενος Τμήματος</th>";
echo "</tr>";
$result=mysql_query("select * from departments");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!");}
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$id=$line['department_id'];
$name=$line['department_name'];
$manager=$line['department_manager'];
echo "<tr>";
echo "<td><input type=\"radio\" name=\"radio\" value=$id></td>";
echo "<td>$id</td>";
echo "<td>$name</td>";
echo "<td>$manager</td>";
echo "</tr>";
}
echo "<input type=\"hidden\" name=\"posted\" value=\"1\">";
echo "</table>";
echo "<p align=center>";
echo "<td><input type=\"submit\" name=\"submit\" value=\"Διαγραφή\"></td>";
echo "</p>";
echo "</form>";
html_footer();
}
function delete_department()
{
if ((isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_delete_department();
exit();
}
if (!isset($_POST['radio']))

```

```

{
header("Location:index.php?page=delete_department");
exit();
}
$id=$_POST['radio'];
$result=mysql_query("delete from departments where department_id=$id");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτημάτων!");}
header ("Location:index.php");
}
@session_start();
check_security(3);
delete_department();
?>

```

- **delete_user.php**

```

<?
include_once ("egrade_fns.php");
function print_delete_user()
{
echo "<form action=\"delete_user.php\" method=\"post\">";
echo "<table border=1 align=center>";
echo "<tr>";
echo "<th></th>";
echo "<th>Κωδικός</th>";
echo "<th>Όνομα Χρήστη</th>";
echo "<th>Όνοματεπώνυμο</th>";
echo "<th>Διεύθυνση email</th>";
echo "</tr>";
$result=mysql_query("select * from users");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!");}
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$id=$line['user_id'];
$username=$line['user_username'];
$firstname=$line['user_firstname'];
$lastname=$line['user_lastname'];
$email=$line['user_email'];
echo "<tr>";
echo "<td><input type=\"radio\" name=\"radio\" value=$id></td>";
echo "<td>$id</td>";
echo "<td>$username</td>";
echo "<td>$firstname $lastname</td>";
echo "<td>$email</td>";
echo "</tr>";
}
echo "<input type=\"hidden\" name=\"posted\" value=\"1\">";
echo "</table>";
echo "<p align=center>";
echo "<td><input type=\"submit\" name=\"submit\" value=\"Διαγραφή\"></td>";
echo "</p>";
echo "</form>";
html_footer();
}
function delete_user()
{
if ((isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_delete_user();

```

```

exit();
}
if (!isset($_POST['radio']))
{
header ("Location:index.php?page=delete_user");
exit();
}
$id=$_POST['radio'];
delete_user_fns($id);
header ("Location:index.php");
}
@session_start();
check_security(3);
delete_user();
?>

```

- **edit_user.php**

```

<?
include_once ("egrade_fns.php");
function edit_user()
{
if (!isset($_POST['posted'])) {print_edit_user();}
else
{
$userid=$_SESSION['userid'];
$registernumber=$_POST['registernumber'];
$username=$_POST['username'];
$password=$_POST['password'];
$firstname=$_POST['firstname'];
$lastname=$_POST['lastname'];
$fathername=$_POST['fathername'];
$address=$_POST['address'];
$city=$_POST['city'];
$postalcode=$_POST['postalcode'];
$phone=$_POST['phone'];
$email=$_POST['email'];
$sql="update users set ";
$sql.="user_registernumber='$registernumber','";
$sql.="user_username='$username','";
$sql.="user_password='$password','";
$sql.="user_firstname='$firstname','";
$sql.="user_lastname='$lastname','";
$sql.="user_fathername='$fathername','";
$sql.="user_address='$address','";
$sql.="user_city='$city','";
$sql.="user_postalcode='$postalcode','";
$sql.="user_phone='$phone','";
$sql.="user_email='$email' ";
$sql.="where user_id=$userid";
$result=mysql_query($sql);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
header("Location:index.php");
}
}
function print_edit_user()
{
$userid=$_SESSION['userid'];
$result=mysql_query("select * from users where user_id=$userid");

```

```

if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
$line=mysql_fetch_array($result,MYSQL_BOTH);
$registernumber=$line['user_registernumber'];
$username=$line['user_username'];
$password=$line['user_password'];
$firstname=$line['user_firstname'];
$lastname=$line['user_lastname'];
$fathername=$line['user_fathername'];
$address=$line['user_address'];
$city=$line['user_city'];
$postalcode=$line['user_postalcode'];
$phone=$line['user_phone'];
$email=$line['user_email'];
echo "<p align=center>Τα πεδία που χαρακτηρίζονται με έντονη γραφή πρέπει να συμπληρωθούν
υποχρεωτικά!</p>";
echo "<form action=\"edit_user.php\" method=\"POST\">";
echo "<table align=center>";
echo "<tr><td><strong>Αριθμός Μητρώου:</strong></td><td><input type=\"text\"
name=\"registernumber\" value=$registernumber maxlength=\"4\" size=\"4\"> (Απαραίτητο
MONO για τους φοιτητές)</td></tr>";
echo "<tr><td><strong>Όνομα Χρήστη:</strong></td><td><input readonly type=\"text\"
name=\"username\" value=$username maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Κωδικός Πρόσβασης:</strong></td><td><input type=\"text\"
name=\"password\" value=$password maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Όνομα:</strong></td><td><input type=\"text\" name=\"firstname\"
value=$firstname maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Επώνυμο:</strong></td><td><input type=\"text\" name=\"lastname\"
value=$lastname maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Πατρώνυμο:</strong></td><td><input type=\"text\"
name=\"fathername\" value=$fathername maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Διεύθυνση:</strong></td><td><input type=\"text\" name=\"address\"
value=$address maxlength=\"60\"></td></tr>";
echo "<tr><td><strong>Πόλη:</strong></td><td><input type=\"text\" name=\"city\" value=$city
maxlength=\"30\"></td></tr>";
echo "<tr><td><strong>Ταχ. Κώδικας:</strong></td><td><input type=\"text\"
name=\"postalcode\" value=$postalcode maxlength=\"6\"></td></tr>";
echo "<tr><td><strong>Τηλέφωνο:</strong></td><td><input type=\"text\" name=\"phone\"
value=$phone maxlength=\"10\"></td></tr>";
echo "<td><strong>Διεύθυνση email:</strong></td><td><input type=\"text\" name=\"email\"
value=$email maxlength=\"60\"></td></tr>";
echo "<tr><td><input type=\"hidden\" name=\"posted\" value=\"1\"><td></tr>";
echo "<tr><td><input type=\"submit\" name=\"submit\" value=\"Αποστολή\"><td><input
type=\"reset\" name=\"reset\" value=\"Καθαρισμός\"></td></tr>";
echo "</table>";
echo "</form>";
}
@session_start();
check_security(1);
edit_user();
?>

```

- **egrade_fns.php**

```

<?
include_once ("db_egrade.inc.php");
include_once ("users_fns.php");
include_once ("pages_fns.php");
?>

```

- **includes.inc**

```

<?
function print_error($msg)
{
print <<<END
<tr>
<td colspan=5><font color=red><b>Σφάλμα: </b></font>$msg</td>
<td><font color=red><b>Απορρίφθηκε</b></font></td>
</tr>
END;
}
function getHeader( $exc, $data )
{
// string
$ind = $data['data'];
if( $exc->sst[unicode][$ind] )
return convertUnicodeString ( $exc->sst['data'][$ind]);
else
return $exc->sst['data'][$ind];
}
function convertUnicodeString( $str )
{
for( $i=0; $i<strlen($str)/2; $i++ )
{
$no = $i*2;
$hi = ord( $str[$no+1] );
$lo = $str[$no];
if( $hi != 0 )
continue;
elseif( ! ctype_alnum( $lo ) )
continue;
else
$result .= $lo;
}
return $result;
}
function uc2html($str) {
$ret = "";
for( $i=0; $i<strlen($str)/2; $i++ ) {
$charcode = ord($str[$i*2])+256*ord($str[$i*2+1]);
$ret .= '&#'.$charcode;
}
return $ret;
}
function get( $exc, $data )
{
switch( $data['type'] )
{
// string
case 0:
$ind = $data['data'];
if( $exc->sst[unicode][$ind] )
return uc2html($exc->sst['data'][$ind]);
else
return $exc->sst['data'][$ind];
// integer
case 1:
return (integer) $data['data'];
// float

```

```

case 2:
return (float) $data['data'];
case 3:
return gmdate("m-d-Y",$exc->xls2tstamp($data[data]));
default:
return "";
}
}
function fatal($msg = "") {
echo '[Fatal error]';
if( strlen($msg) > 0 )
echo ": $msg";
echo "<br>\nΤο script τερματίστηκε<br>\n";
if( $_opened ) @fclose($fh);
exit();
}
function getTableData ( $ws, $exc ) {
global $excel_file, $db_table;
global $db_host, $db_name, $db_user, $db_pass;
$data = $ws['cell'];
echo <<<FORM
<form action="" method="POST" name="db_export">
<table border="0" cellspacing="1" cellpadding="2" align="center" bgcolor="#666666">
<tr bgcolor="#f1f1f1">
FORM;
// Form fieldnames
if ( !$_POST['useheaders'] ) {
for ( $j = 0; $j <= $ws['max_col']; $j++ ) {
$field = "field" . $j;
echo <<<HEADER
<td>
<input type="checkbox" name="fieldcheck[$j]" value="$j" checked title="Check to proceed this
field">
<input type="text" name="fieldname[$j]" value="$field" title="Field name">
</td>
HEADER;
}
}
else {
for ( $j = 0; $j <= $ws['max_col']; $j++ ) {
$field = getHeader ( $exc, $data[0][$j] );
$field = ereg_replace ( "^[0-9]+", "", $field );
if ( empty ( $field ) )
$field = "field" . $j;
echo <<<HEADER
<td>
<input type="checkbox" name="fieldcheck[$j]" value="$j" checked title="Check to proceed this
field">
<input type="text" name="fieldname[$j]" value="$field" title="Field name">
</td>
HEADER;
}
}
echo "</tr>";
foreach( $data as $i => $row ) { // Output data and prepare SQL instructions
if ( $i == 0 && $_POST['useheaders'] )
continue;
echo "<tr bgcolor=\"#ffffff\">";
for ( $j = 0; $j <= $ws['max_col']; $j++ ) {
$cell = get ( $exc, $row[$j] );

```

```

echo "<td>$cell</td>";
}
echo "</tr>";
$i++;
}
echo <<<FORM2
</table><br>
<table align="center" width="390">
<tr><td><input type="hidden" name="db_table" value="results"></td></tr>
<tr><td><input type="hidden" name="db_drop" unchecked></td></tr>
<!--
<tr><td colspan="2"><i>Uncheck this option to add data into the existing table.<br><font
color="red">
Note that if you have mismatch in fieldnames in database and fieldnames in outputting data will be
errors!</td></tr>
-->
<tr><td><input type="hidden" name="db_host" value="$db_host"><input type="hidden"
name="db_name" value="$db_name"></td></tr>
<tr><td><input type="hidden" name="db_user" value="$db_user"><input type="hidden"
name="db_pass" value="$db_pass"></td></tr>
<tr><td><input type="hidden" name="excel_file" value="$excel_file"><input type="hidden"
name="step" value="2"></td></tr>
<tr><td align=center><input type="submit" name="submit" value="Αποστολή"></td></tr>
</form>
</table>
<br>&nbsp;
<div align="right">
</div>
FORM2;
}
function prepareTableData ( $exc, $ws, $fieldcheck, $fieldname ) {
$data = $ws['cell'];
foreach( $data as $i=>$row ) { // Output data and prepare SQL instructions
if ( $i == 0 && $_POST['useheaders'] )
continue;
$SQL[$i] = "";
for ( $j = 0; $j <= $ws['max_col']; $j++ ) {
if ( isset($fieldcheck[$j]) ) {
$SQL[$i] .= $fieldname[$j];
$SQL[$i] .= "=";
$SQL[$i] .= addslashes ( get ( $exc, $row[$j] ) );
$SQL[$i] .= "\"";
$SQL[$i] .= ",";
}
}
$SQL[$i] = rtrim($SQL[$i], ',');
$i++;
}
return $SQL;
}
?>

```

- **index.php**

```

<?
include_once ("egrade_fns.php");
function print_index()
{
html_header("eGrade");

```

```

$username=$_SESSION['username'];
echo "<p align=center><strong>Χρήστης: $username</strong></p>";
echo "<p align=left>";
print_menu();
echo "</p>";
if (isset($_REQUEST['page']))
{
$page=$_REQUEST['page'];
include ($page.".php");
}
html_footer();
}
@session_start();
check_security(1);
print_index();
?>

```

- **login.php**

```

<?
include_once("egrade_fns.php");
function print_connect_user()
{
html_header("Σύνδεση...");
echo "<form action='login.php' method='post'>";
echo "<table align=center>";
echo "<tr><td>Όνομα Χρήστη:</td><td><input type='text' name='username'></td></tr>";
echo "<tr><td>Κωδικός Πρόσβασης:</td><td><input type='password' name='password'
maxlength='30'></td></tr>";
echo "<tr><td colspan='2' align='center'><input type='submit' name='submit'
value='Σύνδεση' maxlength='30'></td></tr>";
echo "<tr><td><input type='hidden' name='posted' value='1'></td></tr>";
echo "</table>";
echo "</form>";
echo "<p align=center><a href='add_user.php'>Εγγραφή νέου χρήστη</a></p>";
html_footer();
}
function connect_user()
{
if ((isset($_POST['posted'])) || ($_POST['posted']!=1))
{
print_connect_user();
exit;
}
if (!filled_out($_POST))
{
print_connect_user();
exit();
}
$username=$_POST['username'];
$password=$_POST['password'];
$username=stripslashes($username);
$password=stripslashes($password);
$result=mysql_query("select * from users where user_username='$username' and
user_password='$password'");
if (!$result)
{
die("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_error());
}
}

```

```

if (mysql_num_rows($result)==0)
{
print_connect_user();
exit();
}
$line=mysql_fetch_array($result,MYSQL_BOTH);
$_SESSION['userid']=$line['user_id'];
$_SESSION['securitylevel']=$line['user_securitylevel'];
$_SESSION['username']=$line['user_username'];
header ("Location:index.php");
}
session_start();
connect_user();
?>

```

- **logout.php**

```

<?
include ("egrade_fns.php");
session_start();
$_SESSION['userid']="";
session_destroy();
header ("Location:login.php");
?>

```

- **pages_fns.php**

```

<?
function html_header($title)
{
echo "<html>";
echo "<head>";
echo "<title>$title</title>";
echo "<style>";
echo "body { font-family: Arial, Helvetica, sans-serif; font-size: 13px; text-align=center }";
echo "li, td { font-family: Arial, Helvetica, sans-serif; font-size: 13px }";
echo "hr { color: #3333cc; width=300; text-align=center }";
echo "a { color: #000000 }";
echo "</style>";
echo "</head>";
echo "<body>";
echo "<p align=center><img src=\"images/logo.jpg\" alt=\"eGrade\" height=100
width=700></p>";
echo "<hr>";
}
function html_footer()
{
echo "<hr>";
echo "<p align=center>(C) Giannis Gkoutisidis - 2004</p>";
echo "</body>";
echo "</html>";
}
function print_menu()
{
$result=mysql_query("select * from menu order by menu_id");
if (!$result)
{
echo "Σφάλμα κατά την εκτέλεση του ερωτήματος!";

```

```

exit();
}
echo "<p align=center>";
echo "| ";
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$caption=$line['menu_caption'];
$url="http://".$_SERVER['SERVER_NAME']."/ggkoutsidis/" . "index.php?page=".$line['menu_name'];
echo "<a href=\"\$url\">$caption<a>";
echo "| ";
}
echo "</p>";
}
function MessageBox($msg)
{
echo "<script language=\"JavaScript\">\n<!--\nalert(\"$msg\");\n/-->\n</script>\n";
}
?>

```

- **register_assignment.php**

```

<?
include_once ("egrade_fns.php");
function print_register_assignment()
{
$userid=$_SESSION['userid'];
$result=mysql_query("select * from users where user_id='$userid'");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος".mysql_error);}
$line=mysql_fetch_array($result,MYSQL_BOTH);
if ($line['user_securitylevel']!=1) {die ("Ο παρόν χρήστης δεν είναι φοιτητής!");}
echo "<form action=\"register_assignment.php\" method=\"post\">";
echo "<table border=0 align=center>";
echo "<tr><td colspan=3 align=center><strong>Καταχωρημένα μαθήματα για τον χρήστη:
$userid</strong></td></tr>";
echo "<tr><td colspan=3 align=center><hr></td></tr>";
echo "<tr>";
echo "<th></th>";
echo "<th>Κωδικός</th>";
echo "<th>Όνομα Μαθήματος</th>";
echo "</tr>";
$tempresult=mysql_query("select studentassignment_assignmentid from studentsassignments
where studentassignment_userid=$userid");
if (mysql_num_rows($tempresult)==0) {$sql="select * from assignments";}
else
{
$tempstring="";
while ($templine=mysql_fetch_array($tempresult,MYSQL_BOTH))
{$tempstring.=$templine[0].",";}
$templength=strlen($tempstring)-1;
$tempstring=substr($tempstring,0,$templength);
$sql="select * from assignments where assignments.assignment_id not in ($tempstring)";
}
$result=mysql_query($sql);
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!".mysql_error());}
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$id=$line['assignment_id'];
$name=$line['assignment_name'];

```

```

echo "<tr>";
echo "<td><input type=\"checkbox\" name=\"assign[ ]\" value=$id></td>";
echo "<td align=center>$id</td>";
echo "<td align=center>$name</td>";
echo "</tr>";
}
echo "<input type=\"hidden\" name=\"posted\" value=\"1\">";
echo "</table>";
echo "<p align=center>";
echo "<td><input type=\"submit\" name=\"submit\" value=\"Καταχώρηση\"></td>";
echo "</p>";
echo "</form>";
html_footer();
}
function register_assignment()
{
$userid=$_SESSION['userid'];
if ((isset($_POST['posted'])) || ($_POST['posted']!=1)){print_register_assignment();exit();}
if (!isset($_POST['assign']))
{
header("Location:index.php?page=register_assignment");
exit();
}
foreach ($_POST['assign'] as $value)
{
$result=mysql_query("insert into studentsassignments
(studentassignment_userid,studentassignment_assignmentid) values ($userid,$value)");
$result2=mysql_query("insert into results (result_userid,result_assignmentid) values
($userid,$value)");
}
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτημάτων!".mysql_error());}
if (!$result2) {die ("Σφάλμα κατά την εκτέλεση του ερωτημάτων!".mysql_error());}
header ("Location:index.php");
}
@session_start();
check_security(1);
register_assignment();
?>

```

- **send_result.php**

```

<?
include_once ("./users_fns.php");
@session_start();
check_security(2);
ini_set("allow_call_time_pass_reference","1");
include_once ("./includes.inc");
include_once ("./egrade_fns.php");
include_once ("./parser/excelparser.php");
$lessonid=$_SESSION['lesson_id'];
$result=mysql_query("select assignment_name from assignments where
assignment_id='$lessonid'");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτημάτων!");}
$line=mysql_fetch_array($result,MYSQL_BOTH);
$lessonname=$line['assignment_name'];
if ( !isset($_POST['step']) )
$_POST['step'] = 0;
?>
<html>

```

```

<head>
<style>
.style1 { color: #FF0000}
</style>
</head>
<body>
<?
html_header("Αποστολή Βαθμολογίας");
print_menu();
// Outputting fileselect form (step 0)
echo "<p align=center><strong>Καταχώρηση βαθμολογίας για το μάθημα:
".$lessonname."</strong></p>";
if ( $_POST['step'] == 0 )
echo <<<FORM
<table width="100%" border="0" align="center">
<tr><td align=center>
Επιλέξτε το προς αποστολή αρχείο Excel
</td>
</tr>
<tr>
<td>
<table border="0" align="center">
<form name="exc_upload" method="post" action="" enctype="multipart/form-data">
<tr><td align=right>Όνομα αρχείου:</td><td><input type="file" size=30
name="excel_file"></td></tr>
<tr><td><input type="hidden" name="useheaders" value="checked"></td></tr>
<tr><td colspan="2" align="center">
<input type="hidden" name="step" value="1">
<input type="button" value="Επόμενο" onClick="
javascript:
if( (document.exc_upload.excel_file.value.length==0))
{ alert('Πρέπει πρώτα να καθοριστεί το όνομα αρχείου'); return; }; submit();
"></td></tr>
</form>
</table>
</td>
</tr>
<tr>
<td align="right">
</td>
</tr>
</table>
FORM;
// Processing excel file (step 1)
if ( $_POST['step'] == 1 ) {
echo "<br>";
// Uploading file
$excel_file = $_FILES['excel_file'];
if( $excel_file )
$excel_file = $_FILES['excel_file']['tmp_name'];
if( $excel_file == " " ) fatal("Δεν υπάρχει αρχείο προς αποστολή");
move_uploaded_file( $excel_file, 'upload/' . $_FILES['excel_file']['name']);
$excel_file = 'upload/' . $_FILES['excel_file']['name'];
$fh = @fopen( $excel_file,'rb');
if( !$fh ) fatal("Δεν υπάρχει αρχείο προς αποστολή");
if( filesize($excel_file)==0 ) fatal("Δεν υπάρχει αρχείο προς αποστολή");
$fc = fread( $fh, filesize($excel_file) );
@fclose($fh);
if( strlen($fc) < filesize($excel_file) )
fatal("Σφάλμα κατά την ανάγνωση του αρχείου");

```

```

// Check excel file
$exc = new ExcelFileParser ();
$res = $exc->ParseFromString($fc);
switch ($res) {
case 0: break;
case 1: fatal("Σφάλμα κατά το άνοιγμα του αρχείου");
case 2: fatal("Το μέγεθος του αρχείου είναι πολύ μικρό");
case 3: fatal("Σφάλμα κατά την ανάγνωση της επικεφαλίδας του αρχείου");
case 4: fatal("Σφάλμα κατά την ανάγνωση του αρχείου");
case 5: fatal("Το παρόν αρχείο δεν είναι Excel, ή είναι παλαιότερης έκδοσης");
case 6: fatal("Το αρχείο είναι κατεστραμμένο");
case 7: fatal("Δεν βρέθηκαν δεδομένα Excel στο αρχείο");
case 8: fatal("Μη υποστηριζόμενη έκδοση αρχείου");
default: fatal("Άγνωστο σφάλμα");
}
// Pricessing worksheets
$ws_number = count($exc->worksheet['name']);
if( $ws_number < 1 ) fatal("Δεν υπάρχουν φύλλα δεδομένων στο αρχείο.");
$ws_number = 1; // Setting to process only the first worksheet
for ($ws_n = 0; $ws_n < $ws_number; $ws_n++) {
$ws = $exc -> worksheet['data'][$ws_n]; // Get worksheet data
if ( !$exc->worksheet['unicode'][$ws_n] )
$db_table = $ws_name = $exc -> worksheet['name'][$ws_n];
else {
$ws_name = uc2html( $exc -> worksheet['name'][$ws_n] );
$db_table = convertUnicodeString ( $exc -> worksheet['name'][$ws_n] );
}
echo "<div align='center'>Φύλλο δεδομένων: <b>$ws_name</b></div><br>";
echo "<span class='style1'>ΠΡΟΣΟΧΗ: Τα πεδία που πρέπει να εισαχθούν είναι 2 (δύο) και
πρέπει να περιλαμβάνουν τον κωδικό του φοιτητή και τη βαθμολογία του MONO με ονόματα
result_userid και result_grade αντίστοιχα!</span>";
$max_row = $ws['max_row'];
$max_col = $ws['max_col'];
if ( $max_row > 0 && $max_col > 0 )
getTableData ( &$ws, &$exc ); // Get structure and data of worksheet
else fatal("Κενό φύλλο δεδομένων");
}
}
if ( $_POST['step'] == 2 ) { // Adding data into mysql (step 2)
echo "<br>";
extract ($_POST);
// Database connect check
if ( !$link = @mysql_connect ($db_host, $db_user, $db_pass) )
fatal("Σφάλμα κατά τη σύνδεση με τη βάση δεδομένων.");
if ( !$connect = mysql_select_db ($db_name ) )
fatal("Το όνομα της βάσης δεδομένων δεν είναι έγκυρο.");
if ( empty ($db_table) )
fatal("Κενό όνομα πίνακα.");
if ( !isset ($fieldcheck) )
fatal("Δεν έχουν επιλεγεί πεδία.");
if ( !is_array ($fieldcheck) )
fatal("Δεν έχουν επιλεγεί πεδία.");
$sql .= "CREATE TABLE IF NOT EXISTS $db_table ( ";
foreach ($fieldcheck as $fc)
if ( empty ( $fieldname[$fc] ) )
fatal("Κενό όνομα πεδίου για το επιλεγμένο πεδίο $fc.");
else {
// Prepare table structure
if ( empty ( $fieldname[$fc] ) )
$fieldname[$fc] = "field" . $fc;

```

```

$tbl_SQL .= $fieldname[$fc] . " text NOT NULL,";
}
$tbl_SQL = rtrim($tbl_SQL, ',');
$tbl_SQL .= ") TYPE=MyISAM";
$fh = @fopen ($excel_file,'rb');
if( !$fh ) fatal("Δεν υπάρχει αρχείο προς αποστολή");
if( filesize($excel_file)==0 ) fatal("Δεν υπάρχει αρχείο προς αποστολή");
$fc = fread( $fh, filesize($excel_file) );
@fclose($fh);
if( strlen($fc) < filesize($excel_file) )
fatal("Σφάλμα κατά την ανάγνωση του αρχείου");
$exc = new ExcelFileParser;
$res = $exc->ParseFromString($fc);
switch ($res) {
case 0: break;
case 1: fatal("Σφάλμα κατά το άνοιγμα του αρχείου");
case 2: fatal("Το μέγεθος του αρχείου είναι πολύ μικρό");
case 3: fatal("Σφάλμα κατά την ανάγνωση της επικεφαλίδας του αρχείου");
case 4: fatal("Σφάλμα κατά την ανάγνωση του αρχείου");
case 5: fatal("Το παρόν αρχείο δεν είναι Excel, ή είναι παλαιότερης έκδοσης");
case 6: fatal("Το αρχείο είναι κατεστραμμένο");
case 7: fatal("Δεν βρέθηκαν δεδομένα Excel στο αρχείο");
case 8: fatal("Μη υποστηριζόμενη έκδοση αρχείου");
default: fatal("Άγνωστο σφάλμα");
}
// Pricessing worksheets
$ws_number = count($exc->worksheet['name']);
if( $ws_number < 1 ) fatal("Δεν υπάρχουν φύλλα δεδομένων στο αρχείο.");
$ws_number = 1; // Setting to process only the first worksheet
for ($ws_n = 0; $ws_n < $ws_number; $ws_n++) {
$ws = $exc -> worksheet['data'][$ws_n]; // Get worksheet data
$max_row = $ws['max_row'];
$max_col = $ws['max_col'];
if ( $max_row > 0 && $max_col > 0 ) {$SQL = prepareTableData ( &$exc, &$ws, $fieldcheck,
$fieldname );}
else {fatal("Κενό φύλλο δεδομένων");}
}
if (empty ( $SQL ))
fatal("Σφάλμα κατά την εξαγωγή του πίνακα");
// Output data into database
$sql_pref = "INSERT INTO " . $db_table . " SET result_assignmentid='$lessonid','";
$serr = "";
for ($nmb=1;$nmb<count($SQL);$nmb++)
{
$sql=$SQL[$nmb];
$sql = $sql_pref . $sql;
if ( !mysql_query ($sql) ) {
$serr .= "<b>Σφάλμα SQL στο</b> :<br>$sql <br>";
}
}
$nmb=$nmb-1;
if ( empty ($serr) ) {
echo <<<SUCC
<div align="center">
<b>Η διαδικασία αποστολής ολοκληρώθηκε επιτυχώς.</b><br><br>
Προστέθηκαν $nmb καταχωρήσεις στον πίνακα "$db_table"<br>
</div>
SUCC;
}
else
echo "<br><br><font color='red'>$serr</font>";

```

```
@unlink ($excel_file);
}
html_footer();
?>
```

- **show_result.php**

```
<?
function show_result($userid)
{
echo "<p align=center>";
$result=mysql_query("select * from users where user_id='$userid'");
if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος".mysql_error);}
$line=mysql_fetch_array($result,MYSQL_BOTH);
if ($line['user_securitylevel']!=1) {die ("Ο παρόν χρήστης δεν είναι φοιτητής!");}
$registernumber=$line['user_registernumber'];
$firstname=$line['user_firstname'];
$lastname=$line['user_lastname'];
echo "<table align=center>";
echo "<tr><td colspan=2 align=center><strong>Στοιχεία Φοιτητή</strong></td></tr>";
echo "<tr><td>Αριθμός Μητρώου:</td><td>$registernumber</td></tr>";
echo "<tr><td>Όνομα:</td><td>$firstname</td></tr>";
echo "<tr><td>Επώνυμο:</td><td>$lastname</td></tr>";
echo "</table>";
echo "<br><br><br>";
$sql="select * from results,assignments where results.result_userid=$userid and
results.result_assignmentid=assignments.assignment_id";
$result=mysql_query($sql);
if (!$result) {die ("Σφάλμα κατά τη εκτέλεση του ερωτήματος!".mysql_error());}
else if (mysql_num_rows($result)==0) {echo "Δεν υπάρχει βαθμολογία για τον φοιτητή!";exit();}
echo "<table align=center>";
echo "<th>Κωδικός Μαθήματος</th>";
echo "<th>Περιγραφή Μαθήματος</th>";
echo "<th>Βαθμολογία</th>";
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$assignmentid=$line['assignment_id'];
$assignmentname=$line['assignment_name'];
$grade=$line['result_grade'];
echo "<tr>";
echo "<td align=center>$assignmentid</td>";
echo "<td align=center>$assignmentname</td>";
echo "<td align=center><strong>$grade</strong></td>";
echo "</tr>";
}
echo "</table>";
echo "</p>";
}
@session_start();
$userid=$_SESSION['userid'];
check_security(1);
show_result($userid);
?>
```

- **users_fns.php**

```
<?
include_once("egrade_fns.php");
function filled_out($form_vars) //Έλεγχος συμπληρωμένων πεδίων
```

```

{
foreach ($form_vars as $key=>$value)
{
if (!isset($key) || ($value==""))
{
return false;
}
return true;
}
}

function generate_password($digits) // Γεννήτρια κωδικού πρόσβασης
{
$salt="abcdefghijklmnopqrstuvwxyz0123456789";
srand((double)microtime()*1000000);
$pass="";
$i=0;
while ($i<$digits)
{
$num=rand()%33;
$tmp=substr($salt,$num,1);
$pass=$pass.$tmp;
$i++;
}
return $pass;
}

function check_security($securitylevel)
{
if (!isset($_SESSION['userid']))
{
header("Location:login.php");
exit();
}
elseif ((isset($_SESSION['securitylevel']) || ($_SESSION['securitylevel']<$securitylevel))
{
echo "Δεν έχετε τα απαραίτητα δικαιώματα!";
exit();
}
}

function returnname($selectname,$table,$column,$value) //Ένα πάρα πολύ καλό παράδειγμα για
function overloading!!!
{
$sql="select * from $table";
if (($column) && ($value)) {$sql.=" where $column=$value";}
$result=mysql_query($sql);
if (!$result) {echo "Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_error();exit();}
else
{
echo "<select name=\"\$selectname\">";
while ($line=mysql_fetch_array($result,MYSQL_BOTH))
{
$value1=$line[0];
$name=$line[1];
echo "<option value=\"\$value1\">$name</option>";
}
echo "</select>";
}
}

```

```
function delete_user_fns($id)
{
    $sql=array("select user_securitylevel from users where user_id=$id",
    "select * from studentsassignments where studentassignment_userid=$id",
    "delete from studentsassignments where studentassignment_userid=$id",
    "delete from results where result_userid=$id",
    "delete from users where user_id=$id"
    );
    $result=mysql_query($sql[0]);
    if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!");}
    $line=mysql_fetch_array($result,MYSQL_BOTH);
    if ($line['user_securitylevel']==1)
    {
        $result=mysql_query($sql[1]);
        if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος!");}
        $rows=mysql_num_rows($result);
        if ($rows>0) // Έχει καταχωρηθεί έστω κι ένα μάθημα από τον φοιτητή
        {
            $result=mysql_query($sql[2]);
            if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
            $result=mysql_query($sql[3]);
            if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
            $result=mysql_query($sql[4]);
            if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
        }
        else {$result=mysql_query($sql[4]);}
        if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
    }
    else {$result=mysql_query($sql[4]);}
    if (!$result) {die ("Σφάλμα κατά την εκτέλεση του ερωτήματος! ".mysql_query());}
    }
    ?>
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

Ø Ηλεκτρονικές πηγές

1. Apache Web Server 2.0.49
<http://httpd.apache.org/docs-2.0/>
2. PHP 4.3.8
<http://www.php.net/manual/el/>
3. MySQL Database Server & Client 4.0.18
<http://dev.mysql.com/doc/mysql/en/index.html>
4. Zend Technologies, LTD.
<http://www.zend.com>
5. Άδεια χρήσης λογισμικού GNU
<http://www.gnu.org>
6. Άδεια χρήσης λογισμικού Q Public
<http://www.opensource.org/licenses/qtpl.php>

Ø Έντυπες Πηγές

1. Vockell, Edward L., and Douglas J. Fiore. "Electronic Gradebooks: What Current Programs Can Do For Teachers". Clearing House, January/February 1993.
2. Rockman, Saul, James Pershing and William Ware. "Productivity, Professionalism, and Empowerment: Given a Computer for Every Teacher". October 1992.
3. Tobias Ratschiller and Till Gerken, with contributions by Zend Technologies, LTD. "Web Application Development with PHP 4.0". New Riders Publishing, July 2000.
4. Luke Welling and Laura Tomson. "PHP and MySQL Web Development". Sams Publishing, March 2001.
5. Jay Greenspan and Brad Bulger. "MySQL/PHP Database Applications". IDG Books WorldWide/M&T Books Publishing, February 2002.

ΗΜΕΡΟΛΟΓΙΟ ΠΟΡΕΙΑΣ ΤΗΣ ΕΡΓΑΣΙΑΣ

Ημερομηνία	Δραστηριότητα
06.04.2004	Ανάθεση θέματος
20.04.2004	Πρώτη συνάντηση με επιβλέποντα καθηγητή για τον καθορισμό των απαιτήσεων του εγγράφου
21.04.2004	Αποστολή σχετικής βιβλιογραφίας από τον καθηγητή
19.05.2004	Αποστολή εγγράφου προς διόρθωση
28.05.2004	Εφαρμογή διορθώσεων που απέστειλε ο καθηγητής για το έγγραφο
31.05.2004	Συνάντηση με επιβλέποντα καθηγητή για το καθορισμό των απαιτήσεων της εφαρμογής
03.06.2004	Συνάντηση με τα μέλη του Κ.Ε.Δ. (Τ.Ε.Ι. Ηπείρου) για την ενεργοποίηση του ιστοχώρου
12.07.2004	Upload της εφαρμογής στο δικτυακό τόπο
14.07.2004	Διορθώσεις επιβλέποντα καθηγητή για την εφαρμογή
02.08.2004	Εφαρμογή των διορθώσεων
06.08.2004	Επικοινωνία με επιβλέποντα καθηγητή για τον τελικό έλεγχο της εφαρμογής
19.08.2004	Επικοινωνία με επιβλέποντα καθηγητή για τον τελικό έλεγχο του εγγράφου
02.09.2004	Παρουσίαση Πτυχιακής Εργασίας