



ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΟΣ

ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ

πρώην Εφαρμογών Πληροφορικής στη Διοίκηση και στην Οικονομία(ΕΠΔΟ)

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΙΤΛΟΣ:

Σύστημα Διαχείρισης Εργασιών σε Php/MySql

Σπουδαστές:

Γεωργιάδου Κωνσταντίνα AM 13139

Παραράς Πέτρος AM 13342

Εισηγητές:

Γαρμπής Αριστογιάννης

Φαλιάγκα Έυη

Περιεχόμενα

ΕΙΣΑΓΩΓΗ	4
1. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ.....	6
1.1. Αναγκαιότητα	6
1.2. Σκοπός	6
1.3. Γενική Περιγραφή.....	7
1.4. Περιγραφή Απαιτήσεων.....	8
1.4.1. Βασικές Απαιτήσεις.....	8
1.4.2. Ειδικές Απαιτήσεις	9
1.4.3. Προδιαγραφές Συστήματος	10
Σύστημα.....	10
Διαχείριση Εργασιών.....	10
Διαχείριση Αιτήσεων.....	10
Διαχείριση Καθηγητή	11
Διαχείριση Φοιτητή.....	11
2. ΣΧΕΔΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ.....	12
2.1. Εισαγωγή στο πληροφοριακό σύστημα.....	12
2.2. Βάση Δεδομένων.....	13
2.3. Διαγράμματα UML	16
3. ΤΕΧΝΟΛΟΓΙΕΣ ΚΑΙ ΕΡΓΑΛΙΑ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ.....	22
3.1 ΧΑΜΡΡ	22
4. ΣΧΕΔΙΑΣΜΟΣ USER INTERFACE.....	31
5. ΑΞΙΟΛΟΓΗΣΗ ΕΦΑΡΜΟΓΗΣ ΚΑΙ ΠΡΟΒΛΗΜΑΤΑ	39
5.1 Προβλήματα	39

5.2 Αξιολόγηση εφαρμογής	40
6. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ (User Guide)	41
7. ΒΙΒΛΙΟΓΡΑΦΙΑ - ΙΣΤΟΓΡΑΦΙΑ.....	54
ΠΑΡΑΡΤΗΜΑ.....	55

ΕΙΣΑΓΩΓΗ

Στα πλαίσια ολοκλήρωσης της σχολής του κάθε φοιτητής καλείται να φέρει εις πέρας μια εργασία, η οποία αντικατοπτρίζει όλα όσα διδάχθηκε κατά τη διάρκεια της φοίτησής του. Η εργασία αυτή αποκαλείται Πτυχιακή Εργασία και μπορεί να αποτελείται μόνο από το θεωρητικό σκέλος, ή και από πρακτικό σκέλος το οποίο διαφοροποιείται ανάλογα με την ιδιότητα της κάθε σχολής.

Την εργασία αυτή τη διεκδικεί κάθε φοιτητής που έχει συμπληρώσει κάποιο συγκεκριμένο αριθμό Διδακτικών Μονάδων. Έχει ολοκληρώσει, δηλαδή, με επιτυχία τις εξετάσεις ορισμένων μαθημάτων, ένδειξη που δηλώνει ότι είναι σε θέση να ανταπεξέλθει στις απαιτήσεις της συγκεκριμένης εργασίας, καθώς θα πρέπει να παρουσιάσει τη γνώση που έλαβε στα χρόνια της φοίτησής του στη Σχολή. Η πτυχιακή μπορεί να ανατεθεί είτε σε μονομελή είτε σε διμελή ομάδα.

Η επιλογή του θέματος συνηθίζεται να γίνεται από μία λίστα με προτάσεις θεμάτων ανά καθηγητή, που αναρτάται από τη σχολή, είτε στο διαδικτυακό σύνδεσμο της, είτε στη γραμματεία της. Δεν είναι λίγες βέβαια οι περιπτώσεις που η ανάθεση της εργασίας γίνεται έπειτα από άμεση επικοινωνία του καθηγητή –φοιτητή, χωρίς να προλάβει να δημοσιοποιηθεί στην παραπάνω λίστα.

Η συγκεκριμένη εργασία είναι και αυτή μία Πτυχιακή Εργασία διμελούς ομάδας. Στα πλαίσια αυτής καλούμαστε να δημιουργήσουμε ένα Πληροφοριακό Σύστημα το οποίο θα συμβάλει στην απομακρυσμένη εξυπηρέτηση προπτυχιακών φοιτητών για την ανάθεση πτυχιακών εργασιών, καθώς και για τον απόλυτο έλεγχο από την πλευρά της σχολής.

Για να επιτευχθεί η απομακρυσμένη εξυπηρέτηση, αποφασίσαμε η εφαρμογή να είναι διαδικτυακή. Ο λόγος που έγινε αυτή η επιλογή είναι γιατί στην τεχνολογία λογισμικού, μια εφαρμογή Ιστού είναι μια εφαρμογή που προσεγγίζεται μέσω ενός φυλλομετρητή ιστοσελίδων (web browser) πάνω από το δίκτυο. Ένας φυλλομετρητής ιστοσελίδων είναι ένα λογισμικό που επιτρέπει στον χρήστη του να προβάλλει, και να αλληλεπιδρά με, κείμενα, εικόνες, βίντεο, μουσική, παιχνίδια και άλλες πληροφορίες συνήθως αναρτημένες σε μια ιστοσελίδα ενός ιστότοπου στον Παγκόσμιο Ιστό ή σε ένα τοπικό δίκτυο. Το κείμενο και οι εικόνες σε μια ιστοσελίδα μπορεί να περιέχουν συνδέσμους προς άλλες ιστοσελίδες του ίδιου ή διαφορετικού ιστότοπου. Ο φυλλομετρητής επιτρέπει στον χρήστη την γρήγορη και εύκολη πρόσβαση σε πληροφορίες που βρίσκονται σε διάφορες ιστοσελίδες και ιστότοπους εναλλάσσοντας τις ιστοσελίδες μέσω συνδέσμων (links). Η κύρια γλώσσα που χρησιμοποιείται από τις εφαρμογές Ιστού και τους φυλλομετρητές είναι η γλώσσα μορφοποίησης HTML για την προβολή των ιστοσελίδων. Για την ανάπτυξη εφαρμογών που χρειάζονται περισσότερες διαδραστικά χαρακτηριστικά και δυνατότητες χρησιμοποιούνται και άλλες γλώσσες προγραμματισμού παράλληλα με την HTML όπως η JavaScript η Java και η PHP.

Οι εφαρμογές Ιστού είναι δημοφιλείς επειδή δεν υπάρχει σύγχρονο λειτουργικό σύστημα που προορίζεται για υπολογιστές γραφείου που να μην έχει προεγκατεστημένο έναν ή περισσότερους φυλλομετρητές. Οι φυλλομετρητές ουσιαστικά αποτελούν λογισμικό πελάτη του δικτυακού πρωτοκόλλου επιπέδου εφαρμογών HTTP. Για κάθε web browser διατίθενται, επίσης, και αρκετά πρόσθετα στοιχεία (add-ons), με στόχο την επαύξηση των δυνατοτήτων τους, τη βελτίωση της χρηστικότητας τους και την προστασία του χρήστη σε θέματα ασφάλειας. Καθώς και την ευκολία της χρησιμοποίησης μιας μηχανής αναζήτησης Ιστού ως χρήστη. Η δυνατότητα να ενημερωθούν και να διατηρηθούν οι εφαρμογές Ιστού χωρίς τη διανομή και εγκατάσταση του λογισμικού ενδεχομένως σε χιλιάδες υπολογιστές χρηστών είναι ένας βασικός λόγος για τη δημοτικότητά τους.

Για να γίνει πιο κατανοητή η λειτουργία του συγκεκριμένου Πληροφοριακού Συστήματος καθώς και για τον προσανατολισμό στο σχεδιασμό του θα κάνουμε χρήση των UML Διαγραμμάτων. Η χρήση των διαγραμμάτων αυτών γίνεται γιατί η UML, σαν πρότυπη γλώσσα μοντελοποίησης στη μηχανική λογισμικού, μπορεί να χρησιμοποιηθεί σε διάφορες φάσεις ανάπτυξης, από την ανάλυση απαιτήσεων ως τον έλεγχο ενός ολοκληρωμένου συστήματος, και αποτελείται από ένα σύνολο προσυμφωνημένων όρων, συμβόλων και διαγραμμάτων.

Ένα πρότυπο σχεδίασης ορίζεται ως μία αποδεδειγμένα καλή λύση που έχει εφαρμοστεί με επιτυχία στην επίλυση ενός επαναλαμβανόμενου προβλήματος σχεδίασης συστημάτων λογισμικού. Τα πρότυπα σχεδίασης ορίζονται τόσο σε επίπεδο μακροσκοπικής σχεδίασης όσο και σε επίπεδο υλοποίησης, ενώ με τη χρήση τους ένας προγραμματιστής αντικαθιστά πρακτικώς μεγάλα τμήματα του κώδικα του με μαύρα κουτιά. Πρόκειται για αφαιρέσεις υψηλού επιπέδου που αποτελούν πλήρη υποσυστήματα, κατάλληλα ρυθμισμένα για την επίλυση συγκεκριμένων προβλημάτων και έτοιμα για χρήση. Έχουν οριστεί διάφορες κατηγορίες προτύπων, για διαφορετικά προβλήματα, και κάθε κατηγορία περιλαμβάνει πολλαπλά στοιχεία. Έτσι υπάρχουν κατασκευαστικά πρότυπα, δομικά πρότυπα, συμπεριφορικά πρότυπα κλπ. Στη δική μας περίπτωση τα πρότυπα που θα σχεδιάσουμε είναι κατά κύριο λόγο δομικά πρότυπα που θα μας βοηθήσουν στη δομή της βάσης αλλά και κατασκευαστικά διότι θα μας βοηθήσουν στην κατανόηση και κατασκευής ολόκληρου του συστήματος.

1. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

1.1. Αναγκαιότητα

Για την ανάθεση της εργασίας, όπως αναφέραμε παραπάνω, αρκεί η επικοινωνία του φοιτητή με τον καθηγητή, είτε με επικοινωνία μέσω ηλεκτρονικής αλληλογραφίας, είτε με προσωπική επαφή.

Τα μεγαλύτερα προβλήματα που μπορούμε να συναντήσουμε στα πλαίσια της υφιστάμενης διαδικασίας είναι ότι πολλές φορές το αρχείο αποτελείται από περιορισμένα θέματα που δεν είχαν ανταπόκριση πριν την ανάρτηση αυτού. Σε αυτήν την περίπτωση, όσοι φοιτητές δεν επικοινωνήσαν με τους καθηγητές ανεξάρτητα από τα πλαίσια της διαδικασίας θα πρέπει να περιοριστούν στα προτεινόμενα θέματα, τα οποία δεν αντιπροσωπεύουν πάντα τα ενδιαφέροντα των φοιτητών. Επίσης η αναζήτηση του αρχείου είναι δύσκολη ανάμεσα σε πληθώρα αναρτήσεων στο πέρασμα του χρόνου. '

Η προσωπική επικοινωνία μεταξύ φοιτητή και καθηγητή είναι πολύ σημαντική και επιβραβεύσιμη, αυτό όμως μπορεί να δράσει αρνητικά στο σημείο της αναζήτησης κατάλληλων θεμάτων, αφού κάποια θέματα δεν θα έχουν ποτέ την δυνατότητα να τα επιλέξουν οι φοιτητές. Άλλωστε μια σχολή με περιορισμένες δυνατότητες προς τους φοιτητές δεν είναι πάντα πρώτη επιλογή ενός μελλοντικού φοιτητή.

Για την αποφυγή όλων των παραπάνω προτάθηκε η δημιουργία μιας οργανωμένης ηλεκτρονικής εξέλιξης της εκπόνησης των εργασιών από τον καθηγητή προς τους φοιτητές. Με αυτόν τον τρόπο παρέχονται περισσότερες επιλογές σύμφωνα με τα ενδιαφέροντά τους, ενώ ταυτόχρονα διασφαλίζεται ο απόλυτος έλεγχος και οργάνωση από την πλευρά της σχολής.

1.2. Σκοπός

Σκοπός της εφαρμογής είναι ο σχεδιασμός και η ανάπτυξη ενός οργανωμένου διαδικτυακού πληροφοριακού συστήματος για την ανάρτηση και απόδοση πτυχιακών εργασιών σε φοιτητές, χωρίς την απαραίτητη φυσική παρουσία αυτών. Πρόκειται ουσιαστικά για μια εφαρμογή όπου ο φοιτητής καλείτε να εκδηλώσει ενδιαφέρον για μια πτυχιακή εργασία και ο αρμόδιος καθηγητής εγκρίνει την αίτηση ή όχι. Στην ουσία η επιλογή των ομάδων γίνεται από την πλευρά των καθηγητών.

Πρόκειται για μια κλασική Web εφαρμογή: Ο χρήστης, με τη βοήθεια του Browser του συνδέεται στον Application Server (ο οποίος είναι ένας Web Server με τη δυνατότητα να εμφανίζει δυναμικές σελίδες) της εφαρμογής. Για την αποθήκευση των στοιχείων της εφαρμογής γίνεται επικοινωνία με μια βάση δεδομένων, η οποία και κρατά όλες τις μόνιμες πληροφορίες.

1.3. Γενική Περιγραφή

Το σύστημα αυτό είναι μια κλασική Web εφαρμογή και απευθύνεται σε γενικές γραμμές σε δυο μέρη:

- Τους φοιτητές: Οι φοιτητές μπορούν να επιλέξουν μια διαθέσιμη εργασία, αφού συνδεθούν ή εγγραφούν στο σύστημα.
- Τους καθηγητές: Οι καθηγητές εισέρχονται στο σύστημα. Αναρτούν τις εργασίες στο δικτυακό τόπο, μπορούν να δουν τις νέες αιτήσεις για τις διαθέσιμες εργασίες και να απορρίψουν ή να εγκρίνουν κάθε αίτηση. Υπάρχει η δυνατότητα επεξεργασίας του αναρτημένου θέματος.

Ο χρήστης, με τη βοήθεια του Browser του συνδέεται στον Application Server (ο οποίος είναι ένας Web Server με τη δυνατότητα να εμφανίζει δυναμικές σελίδες) της εφαρμογής. Για την αποθήκευση των στοιχείων της εφαρμογής γίνεται επικοινωνία με μια βάση δεδομένων, η οποία και κρατά όλες τις μόνιμες πληροφορίες. Σε παρακάτω κεφάλαιο φαίνεται αναλυτικά η δομή της εφαρμογής.

Πρέπει να σημειωθεί εδώ ότι όλη η λογική του προγράμματος βρίσκεται στην πλευρά του Application Server. Η εφαρμογή που περιέχεται στον Application Server μεταγλωττίζεται όταν ζητηθεί από τον χρήστη και το αποτέλεσμα είναι μια σειρά από απλές HTML σελίδες τις οποίες ο Application Server αποστέλλει στον web browser. Το να μην υπάρχει η λογική του προγράμματος στις σελίδες που αποστέλλονται στον χρήστη εξυπηρετεί αρκετούς σκοπούς, όπως:

- Δεν ενδιαφέρει το είδος του browser που χρησιμοποιεί ο χρήστης. Υπάρχουν δεκάδες διαφορετικοί browsers οι οποίοι έχουν διαφορετικές δυνατότητες: Από τους γνωστούς browsers Internet Explorer και Mozilla που υποστηρίζουν σχεδόν όλες τις δυνατότητες που μπορεί να φανταστεί κανείς (αλλά ακόμα και έτσι υπάρχουν σελίδες οι

οποίες δε μπορεί να εμφανιστούν στον ένα ή τον άλλο), μέχρι browsers για κινητά τηλέφωνα.

- Δεν ενδιαφέρει η ταχύτητα του υπολογιστή του χρήστη – άλλωστε αυτός ο υπολογιστής μπορεί να μην είναι καν υπολογιστής, αλλά ένα κινητό τηλέφωνο μιας και όλοι οι υπολογισμοί γίνονται στην πλευρά της βάσης.
- Ο κώδικας δεν αποστέλλεται στους χρήστες. Εξαίρεση αποτελεί η Javascript που χρησιμοποιείται σε μερικά σημεία της εφαρμογής, μονάχα όμως για βοηθητικούς σκοπούς και τη βελτίωση της ποιότητας υπηρεσίας. Οι υπολογισμοί και τα δεδομένα γίνονται και αποθηκεύονται στη βάση.

Ο Web Server και η βάση δεδομένων μπορεί να βρίσκονται στον ίδιο, σε διαφορετικούς υπολογιστές ή ακόμα και να είναι διαμοιρασμένοι σε πολλούς υπολογιστές ώστε να είναι γρηγορότερη η απόκριση του συστήματος αλλά και να υπάρχει περισσότερη αξιοπιστία. Σε παρακάτω κεφάλαιο θα δούμε πιο αναλυτικά τις τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση της συγκεκριμένης εφαρμογής.

1.4. Περιγραφή Απαιτήσεων

Το συγκεκριμένο Πληροφοριακό Σύστημα απευθύνεται σε μια κατηγορία χρηστών, φοιτητές, οι οποίοι κατά μέσο όρο έχουν βασικές γνώσεις χρήσης Ηλεκτρονικού Υπολογιστή, και στους καθηγητές, οι οποίοι, λόγω της δουλειάς τους, η εξοικείωση με τους υπολογιστές είναι αναπόφευκτη. Παρ' όλα αυτά, θεωρώντας ότι αυτό θα είναι το βασικό σύστημα διαχείρισης πτυχιακών θα πρέπει να έχει κάποιες προδιαγραφές, τις οποίες αναλύουμε παρακάτω.

1.4.1. Βασικές Απαιτήσεις

- 1) **Φιλικό περιβάλλον.** Στόχος η ύπαρξη μιας φιλικής προς το χρήστη διεπαφής η οποία να μην αποτρέπει τη χρήση του συστήματος και να βοηθά τους μη έμπειρους χρήστες,

- 2) **Έλλειψη ανάγκης για ειδικό λογισμικό σύνδεσης.** Η εφαρμογή θα πρέπει να σχεδιαστεί σε ένα περιβάλλον στο οποίο δεν θα χρειάζεται ειδικό λογισμικό για τη χρήση της. Η έλλειψη ειδικού λογισμικού ελαχιστοποιεί το κόστος για τη χρήση του συστήματος. Για το λόγο αυτό η εφαρμογή πρέπει να σχεδιαστεί ως Εφαρμογή Ιστού.
- 3) **Γρήγορη πρόσβαση στην εφαρμογή.** Επειδή η εφαρμογή είναι δικτυακή, επηρεάζεται από την απόδοση του δικτύου. Στόχος είναι η ελαχιστοποίηση των δεδομένων που ανταλλάσσονται ώστε η επικοινωνία να είναι όσο το δυνατόν γρηγορότερη. Όταν μια διεπαφή είναι αργή επηρεάζει αρνητικά τους χρήστες και μειώνεται σημαντικά η χρηστικότητα της.

1.4.2. Ειδικές Απαιτήσεις

Όπως αναφέραμε παραπάνω η εφαρμογή απευθύνεται σε δυο μέρη. Για τη σωστότερη δόμηση των λειτουργιών που απαιτείται από τους χρήστες αρκεί να κάνουμε μία ανάλυση διαδικασιών. Όσο πιο αναλυτικά παραθέσουμε τις λειτουργικές απαιτήσεις του συστήματος, τόσο καλύτερα θα καλύψουμε τους χρήστες αυτού. Στην αναλυτική καταγραφή απαιτήσεων διακρίναμε τρεις βασικές λειτουργίες του συστήματος:

- Διαχείριση Αιτήσεων
- Διαχείριση Εργασιών
- Διαχείριση Ανάθεσης εργασιών

Αναλύοντάς τες περισσότερο διαπιστώσαμε ότι η κάθε λειτουργία αποτελείται από περεταιίρω διεργασίες. Για την δημιουργία μιας αίτησης προαπαιτείται η εμφάνιση της λίστας των εργασιών, η κατηγορία η οποία εντάσσονται, ο καθηγητής που τις εισάγει καθώς και την κατάσταση της αίτησης. Αν είναι δηλαδή διαθέσιμη ή όχι.

Για την δημιουργία ή επεξεργασία μιας εργασίας, ο καθηγητής καλείται να ενημερώνει το σύστημα για την κατηγορία που την εντάσσει, καθώς και ένα αρχείο στο οποίο θα συμπεριλαμβάνονται πληροφορίες για το θέμα. Το όνομά του καθηγητή εμφανίζεται αυτόματα από το σύστημα αφού έχει ήδη κάνει είσοδο στην εφαρμογή με τους κωδικούς του.

Τέλος για την διαδικασία ανάθεσης εργασιών, η συγκεκριμένη διαδικασία απευθύνεται αποκλειστικά στον καθηγητή, αρκεί ο καθηγητής να επιλέξει έγκριση ή απόρριψη. Η δυνατότητα αυτή εμφανίζεται στις αιτήσεις που βρίσκονται στη «Λίστα Αιτήσεων», επιλογή που εμφανίζεται στην κεντρική σελίδα του καθηγητή.

1.4.3. Προδιαγραφές Συστήματος

Σύστημα

- A1. Το σύστημα έχει μηχανισμό ταυτοποίησης των χρηστών του για να παρέχει τις κατάλληλες επιλογές ανά κατηγορία.
- A2. Για κάθε ενέργεια πέραν της πλοήγησης απαιτείται είσοδος στο σύστημα.
- A3. Για κάθε είσοδο στο σύστημα χρειάζεται η πληκτρολόγηση του ονόματος χρήστη (username) και του κωδικού πρόσβασης (password).
- A4. Το σύστημα αποθηκεύει δεδομένα για τους χρήστες. Ο κάθε χρήστης καλείται , στα πλαίσια της εγγραφής του, να συμπληρώσει τα παρακάτω χαρακτηριστικά: Ονοματεπώνυμο (κείμενο), username(κείμενο), password(κείμενο), e-mail(κείμενο), κατηγορία (Καθηγητής, Φοιτητής).
- A5. Δεν υπάρχει περιορισμός στους χαρακτήρες-αριθμούς του κωδικού πρόσβασης.

Διαχείριση Εργασιών

- A6. Κάθε εργασία ανήκει σε ένα καθηγητή.
- A7. Κάθε εργασία αποτελείται από τα εξής πεδία: τίτλος θέματος (κείμενο), Περιγραφή (κείμενο), Κατηγορία (λίστα), Αρχείο (αρχείου τύπου doc), Κατάσταση (αυτόματο από το σύστημα), Καθηγητής (πληροφορία από τη βάση Δεδομένων).
- A8. Στους μη ταυτοποιημένους χρήστες δίνεται η δυνατότητα προβολής των εργασιών.
- A9. Φίλτρο αναζήτησης της κάθε αίτησης είναι η κατηγορία (οικονομικά, διοίκηση κα) και ο καθηγητής.
- A10. Η κατάσταση της κάθε εργασίας αλλάζει, από τη στιγμή που έχει εγκριθεί μια αίτηση σε μη διαθέσιμη.
- A11. Κάθε εργασία που διαγράφεται παύει να είναι ορατή στο σύστημα.

Διαχείριση Αιτήσεων

- A12. Κάθε αίτηση αποτελείται από τον μοναδικό κωδικό της κάθε αίτησης (αυτόματο από το σύστημα), το όνομα του μοναδικού φοιτητή (πληροφορία από τη βάση δεδομένων) , την

ημερομηνία που έχει σταλεί η αίτηση καθώς και την επιλογή που θα ορίσει ο καθηγητής ώστε να εμφανιστεί και η κατάστασή της (0 ή 1) .

A13. Κάθε αίτηση αντιπροσωπεύει ένα φοιτητή και απευθύνεται σε ένα καθηγητή.

A14. Σε κάθε εργασία που βρίσκεται σε κατάσταση «Μη διαθέσιμη» δεν υπάρχει η δυνατότητα επιλογής της.

Διαχείριση Καθηγητή

A15. Κάθε καθηγητής έχει τη δυνατότητα να ελέγχει και να διαχειρίζεται την ανάθεση των δικών του εργασιών.

A16. Η εκπόνηση της εργασίας από τον καθηγητή προς τον φοιτητή γίνεται με την επιλογή “έγκριση”, ενώ η απόρριψη της αίτησης με την επιλογή “απόρριψη”.

A17. Κάθε καθηγητής έχει τη δυνατότητα να καταχωρήσει πολλές εργασίες.

A18. Υπάρχει η επιλογή επεξεργασίας ή διαγραφής της κάθε αίτησης.

A19. Κάθε καθηγητής χρησιμοποιεί τους κωδικούς πρόσβασης της σχολής για την ταυτοποίηση που απαιτεί το σύστημα.

Διαχείριση Φοιτητή

A20. Κάθε φοιτητής έχει δυνατότητα δημιουργίας λογαριασμού.

A21. Δεν υπάρχει περιορισμός στον αριθμό αιτήσεων που μπορεί να κάνει, ούτε στον αριθμό των καθηγητών.

A22. Για την κατάσταση των αιτήσεων που έχουν καταχωρηθεί καθώς και τα αποτελέσματα αυτών εμφανίζονται στην κεντρική σελίδα- προφίλ του φοιτητή.

A23. Κάθε φοιτητής μπορεί να κάνει λήψη αρχείου για περαιτέρω πληροφορίες.

Η προσθήκη και διαχείριση καθηγητών στο σύστημα γίνεται από τους διαχειριστές με κάποιο εξωτερικό πρόγραμμα και δεν ενδιαφέρει τη συγκεκριμένη εφαρμογή.

Στη συνέχεια παρουσιάζονται αναλυτικά τα χαρακτηριστικά των οντοτήτων που απαιτούνται, στο σχεδιασμό της βάσης δεδομένων.

2. ΣΧΕΔΙΑΣΜΟΣ ΕΦΑΡΜΟΓΗΣ

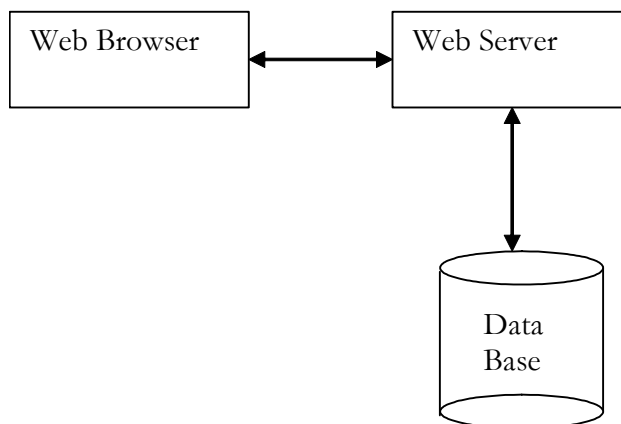
2.1. Εισαγωγή στο πληροφοριακό σύστημα

Η παραπάνω εφαρμογή είναι ένα ολοκληρωμένο Πληροφοριακό Σύστημα που καλύπτει ένα κομμάτι της οργάνωσης της σχολής. Πριν αναλύσουμε όμως τη δομή που αποτελείται το συγκεκριμένο πληροφοριακό σύστημα, καλό θα ήταν να αναφέρουμε τι ορίζεται Πληροφοριακό Σύστημα.

”Πληροφοριακό σύστημα είναι ένα σύνολο από συνιστώσα στοιχεία που αλληλεπιδρούν για να παράγουν πληροφορία.” (<http://users.uom.gr/~kat/ap1/notes/Intro.pdf>, σελ.1)

Ένα πληροφοριακό σύστημα, δηλαδή, είναι μια μορφή επικοινωνίας του συστήματος στο οποίο τα δεδομένα αντιπροσωπεύουν και υποβάλλονται σε επεξεργασία ως μια μορφή κοινωνικής μνήμης. Ένα πληροφοριακό σύστημα μπορεί επίσης να θεωρηθεί ως ημι-επίσημη γλώσσα που υποστηρίζει τις ανθρώπινες λήψεις αποφάσεων και δράσης.

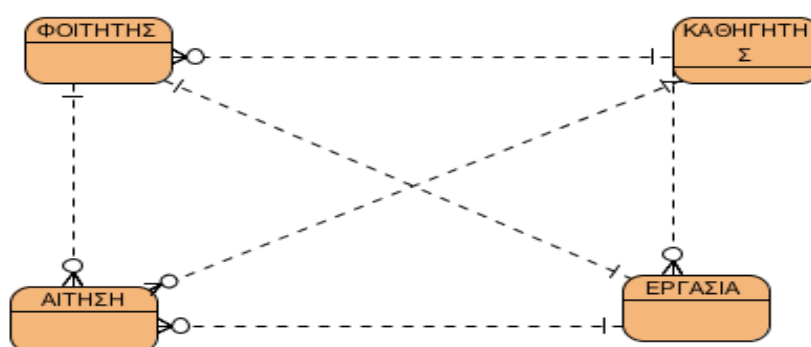
Όπως αναφέραμε παραπάνω το συγκεκριμένο Πληροφοριακό Σύστημα εξυπηρετεί την οργάνωση κάθε σχολής στα πλαίσια της ανάθεσης πτυχιακών εργασιών. Αποδείξαμε ότι για την ευκολότερη λειτουργία του συστήματος η εφαρμογή θα πρέπει να στηθεί στην πλευρά του server. Η αρχιτεκτονική του συστήματος εμφανίζεται στο παρακάτω σχήμα.



Εικόνα 1: Η αρχιτεκτονική της εφαρμογής

Για να κατανοήσουμε πλήρως την αρχιτεκτονική της συγκεκριμένης εφαρμογής θα ξεκινήσουμε την ανάλυση της από αυτό που βλέπει το σύστημα και στη συνέχεια αυτό που βλέπει ο χρήστης.

Από την γενική περιγραφή, λοιπόν της εφαρμογής διακρίνονται 4 οντότητες. Στο παρακάτω διάγραμμα μπορούμε να διακρίνουμε τη συσχέτιση μεταξύ των οντοτήτων. Η μία εργασία που θα καταλήξει να έχει ένας φοιτητή θα είναι από ένα μόνο καθηγητή, έπειτα από πολλές αιτήσεις που θα χρειαστεί να κάνει. Ο κάθε καθηγητής θα δημιουργήσει πολλές εργασίες, έτσι ώστε οι φοιτητές να μπορούν να κάνουν πολλές αιτήσεις. Η κάθε αίτηση δημιουργείται αποκλειστικά από ένα φοιτητή για μία συγκεκριμένη εργασία ενός καθηγητή. Ενώ η κάθε εργασία ενός καθηγητή θα καταλήξει στα χέρια ενός φοιτητή ανάμεσα σε πληθώρα αιτήσεων.



Εικόνα 2: Διάγραμμα Οντοτήτων-Συσχετίσεων

2.2. Βάση Δεδομένων

Από τις λειτουργικές απαιτήσεις για τις αποθήκες δεδομένων, προκύπτει η ανάγκη για τους παρακάτω πίνακες δεδομένων:

Πίνακας καθηγητών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	username	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
2	password	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
3	name	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
4	email	varchar(255)	utf8_unicode_ci		Όχι	Καμία	

Πρωτεύον κλειδί: username

Πίνακας κατηγοριών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	<u>ID</u>	int(11)			Όχι	Καμία	AUTO_INCREMENT
2	name	varchar(255)	utf8_unicode_ci		Όχι	Καμία	

Πρωτεύον κλειδί: ID

Πίνακας εργασιών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	<u>ID</u>	int(11)			Όχι	Καμία	AUTO_INCREMENT
2	description	text	utf8_unicode_ci		Όχι	Καμία	
3	categoryID	int(11)			Όχι	Καμία	
4	title	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
5	status	tinyint(4)			Όχι	Καμία	
6	username	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
7	filename	varchar(255)	utf8_unicode_ci		Όχι	Καμία	

Πρωτεύον κλειδί: ID

Δευτερεύον κλειδί: categoryID, συσχετίζεται με το ID του πίνακα κατηγοριών

Δευτερεύον κλειδί: username, συσχετίζεται με το username του πίνακα καθηγητών

Το status δείχνει την κατάσταση της εργασίας και παίρνει τις ακόλουθες τιμές:

- 0: διαθέσιμη
- 1: μη διαθέσιμη

Η πληροφορία αυτή μπορεί να εξαχθεί και από τον πίνακα αιτήσεων, αλλά χρησιμοποιείται για λόγους ευκολίας.

Πίνακας φοιτητών

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετα
1	username	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
2	password	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
3	name	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
4	email	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
5	essayID	int(11)			Όχι	0	

Πρωτεύον κλειδί: username

Το essayID, συσχετίζεται με το ID του πίνακα essay, όταν ανατεθεί κάποια εργασία στον φοιτητή, παίρνει τιμή ίση με το ID της εργασίας για την οποία έγινε η αίτηση. Η πληροφορία αυτή βρίσκεται και στον πίνακα αιτήσεων, αλλά χρησιμοποιείται το essayID για λόγους ευκολίας.

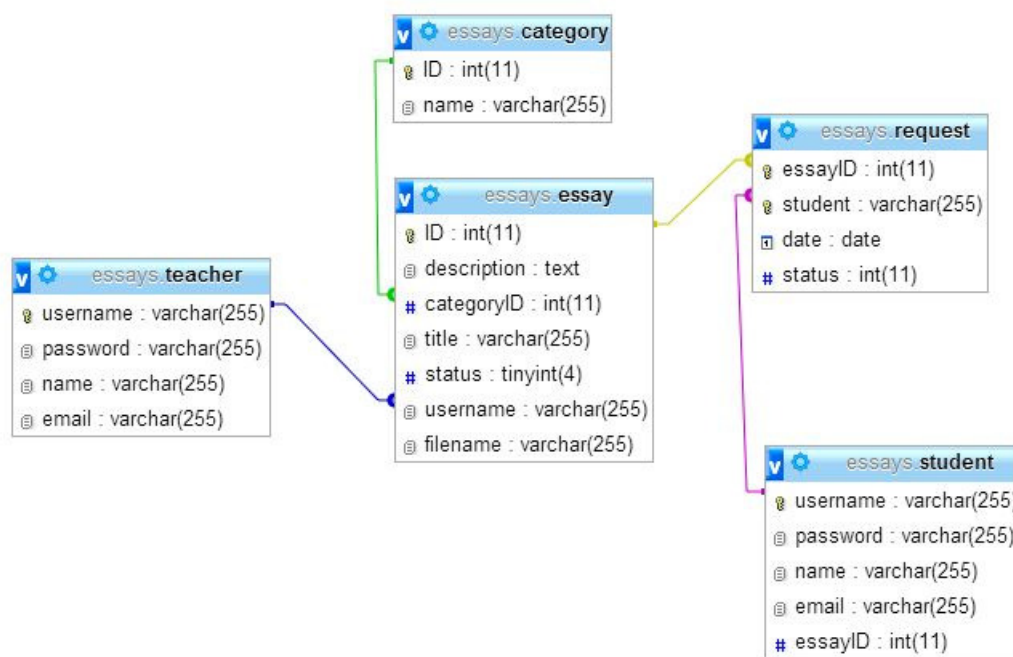
Πίνακας αιτήσεων

1	essayID	int(11)			Όχι	Καμία	
2	student	varchar(255)	utf8_unicode_ci		Όχι	Καμία	
3	date	date			Όχι	Καμία	
4	status	int(11)			Όχι	0	

Κλειδί: συνδυασμός essayID και student, που αντιστοιχίζονται στο ID του πίνακα εργασιών και το username του πίνακα φοιτητών. Η ημερομηνία είναι η ημερομηνία δημιουργίας της αίτησης. Το status παίρνει τις ακόλουθες τιμές:

- 0: εκκρεμής αίτηση
- 1: αίτηση που απορρίφθηκε
- 2: αίτηση που έγινε δεκτή

Στη συνέχεια παρουσιάζουμε το σχήμα της βάσης δεδομένων:



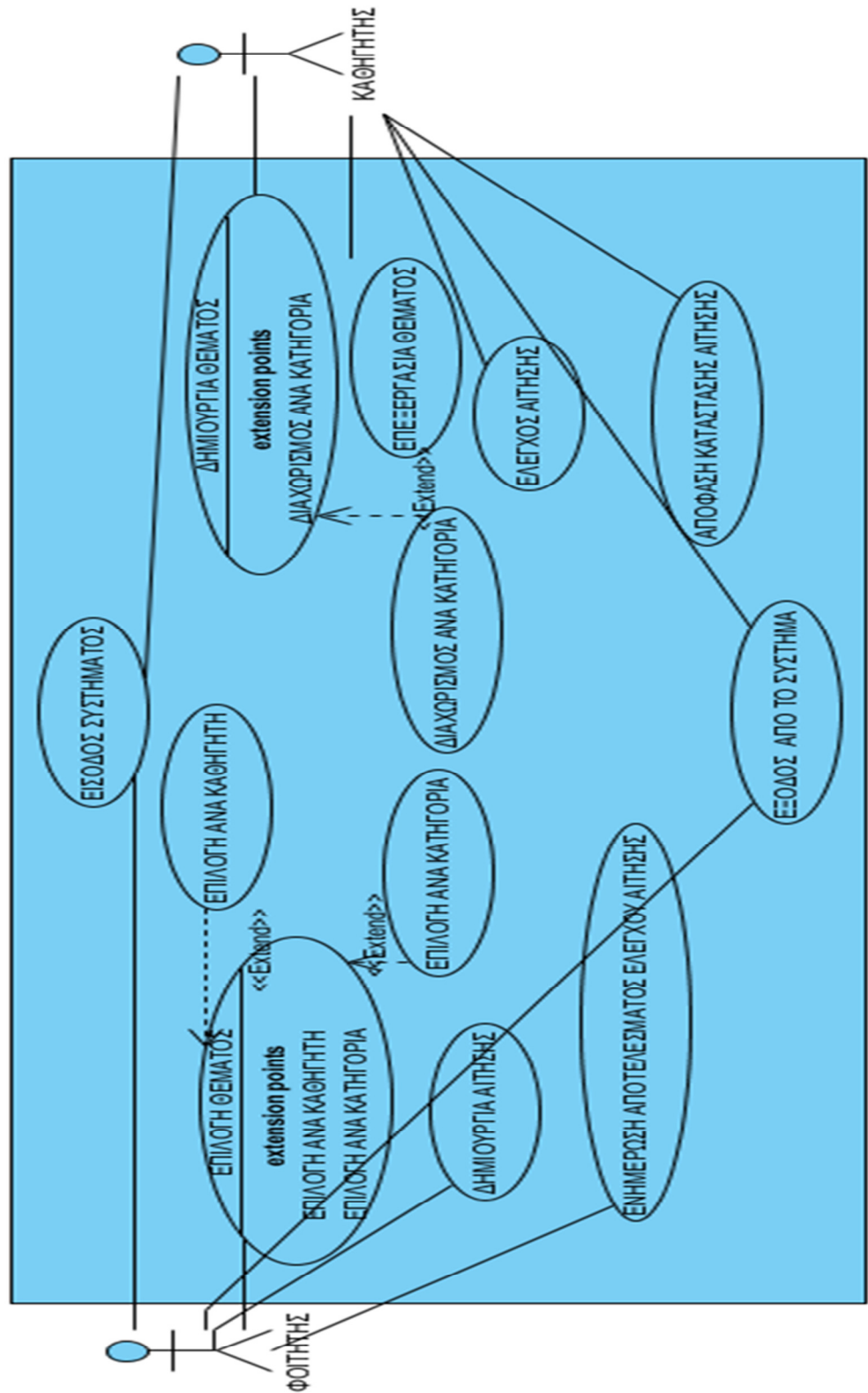
Εικόνα 3: Βάση Δεδομένων

2.3. Διαγράμματα UML

Στο παρακάτω διάγραμμα μπορούμε να κατανοήσουμε τις λειτουργίες που μπορεί να κάνει η κάθε οντότητα, καθώς και οι εναλλακτικές επιλογές που του προσφέρονται από το σύστημα για κάθε κατηγορία χρηστών.

Ο φοιτητής αφού κάνει είσοδο στο σύστημα, έχει τη δυνατότητα να επιλέξει τα θέματα που τον ενδιαφέρουν κάνοντας αναζήτηση από τη λίστα θεμάτων είτε με φίλτρο αναζήτησης τον καθηγητή που την προτείνει, είτε με την κατηγορία που εντάσσεται το κάθε θέμα (πχ Οικονομικά, Διοίκηση, Πληροφορική κα). Αφού δημιουργήσει την αίτηση, αναμένει για την έγκριση ή την απόρριψη της επιλογής του από τον διδάσκοντα καθηγητή. Η ενημέρωση των αποτελεσμάτων γίνεται στην αρχική οθόνη του φοιτητή.

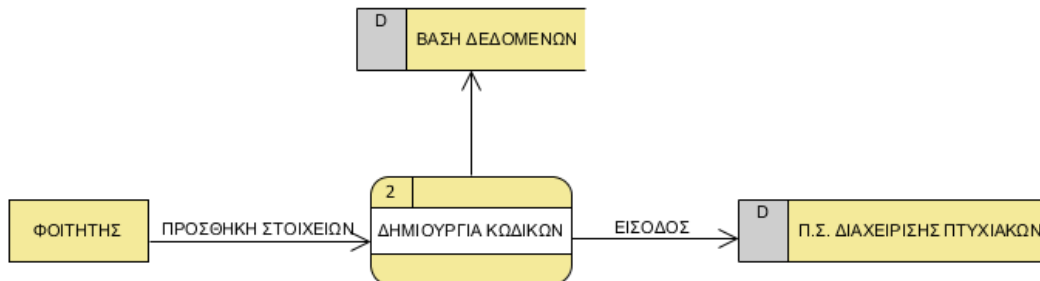
Ο καθηγητής αφού κάνει είσοδο στο σύστημα έχει τη δυνατότητα να προσθέσει κάποιο καινούργιο θέμα, δηλώνοντάς του κατηγορία, όπως και να επεξεργαστεί ένα παλαιότερο. Επίσης μπορεί να ελέγξει τις αιτήσεις του και στη συνέχεια να αναθέσει ή όχι σε κάποιον φοιτητή το θέμα που έχει αναρτηθεί.



Εικόνα 4: Διάγραμμα Περίπτωσης Χρήσης (use case)

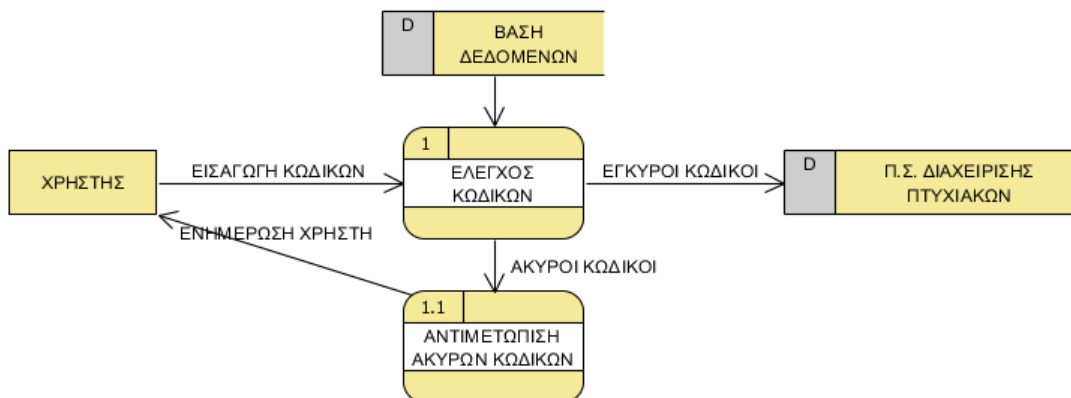
Στο παρακάτω διάγραμμα φαίνεται η λογική λειτουργία του συστήματος μέσα από τους χρήστες.

Για τον φοιτητή



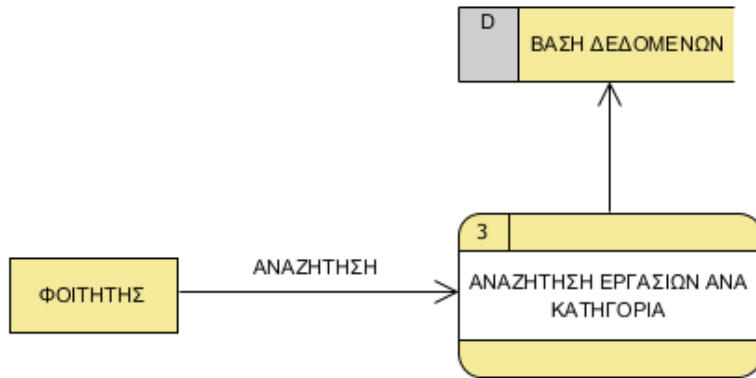
Εικόνα 5: Εγγραφή φοιτητή

Στην παραπάνω εικόνα φαίνεται η διαδικασία εγγραφής νέου χρήστη, δημιουργώντας το προσωπικό του προφίλ με τη συμπλήρωση συγκεκριμένων πεδίων. Ενώ στην εικόνα 6 φαίνεται η σύνδεση στη βάση με τους κωδικούς που έχει προεπιλέξει.

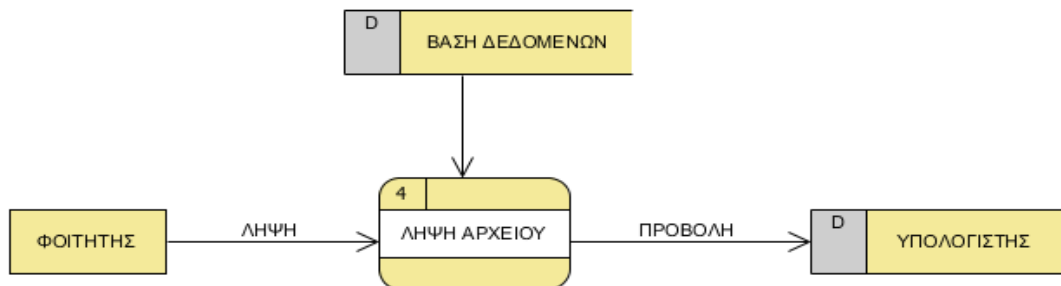


Εικόνα 6: Είσοδος στο σύστημα

Εδώ μπορούμε να διακρίνουμε ότι με τη συμπλήρωση των κωδικών, το σύστημα με τη βοήθεια της βάσης δεδομένων, ελέγχει την ακεραιότητα των κωδικών. Αν οι κωδικοί είναι σωστοί, ο χρήστης εισέρχεται στο Πληροφοριακό Σύστημα. Σε αντίθετη περίπτωση επιστρέφει στην αρχική, ενημερώνοντας το χρήστη και ζητώντας του να τους ξαναπληκτρολογήσει. Στην εικόνα 7 φαίνεται η διαδικασία αναζήτησης εργασιών ανά κατηγορία ή ανά καθηγητή. Στη συγκεκριμένη διαδικασία το σύστημα λαμβάνοντας πληροφορίες από τη βάση εμφανίζει τη λίστα των εργασιών που είναι αναρτημένες έπειτα από φιλτράρισμα (ανά καθηγητή ή ανά κατηγορία) που έχει ορίσει ο φοιτητής.

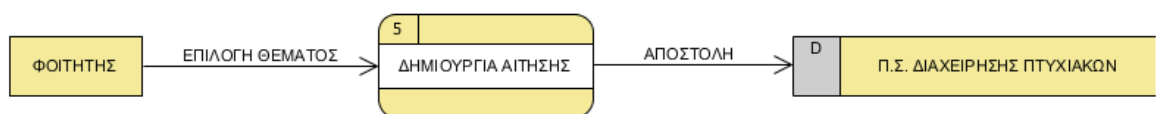


Εικόνα 7: Αναζήτηση εργασιών



Εικόνα 8: Λήψη αρχείου

Στην εικόνα 8 εμφανίζεται η διαδικασία κατά την οποία ο χρήστης κάνει λήψη το αρχείο το οποίο συμπεριλαμβάνει πληροφορίες για την περαιτέρω ενημέρωση του για το κάθε θέμα. Η διαδικασία αυτή θα μπορούσε να θεωρηθεί συμπληρωματική της διαδικασίας αναζήτησης εργασιών, παρ' όλα αυτά επειδή είναι προαιρετική την ορίζουμε σαν ξεχωριστή διαδικασία.



Εικόνα 9: Δημιουργία αίτησης

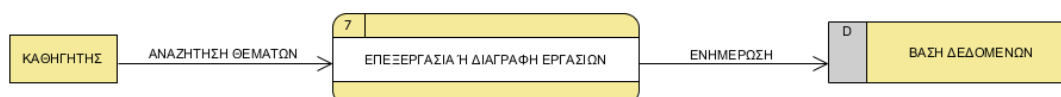
Η διαδικασία αυτή αντικατοπτρίζει τη δημιουργία μιας αίτησης από το φοιτητή. Στην πραγματικότητα ο φοιτητής δημιουργεί μια αίτηση πατώντας το κουμπί «Επιλογή» και το σύστημα ενημερώνει τον καθηγητή για την αίτηση μέσω του Πληροφοριακού Συστήματος.

Για τον καθηγητή

Για την είσοδο του καθηγητή στο σύστημα (εικόνα 6) αρκεί να συμπληρώσει τους κωδικούς που του παρέχονται από την σχολή. Στη συνέχεια εισέρχεται στο προφίλ του καθηγητή που του δίνονται κάποιες επιλογές. Μια από τις επιλογές αυτές είναι η καταχώρηση νέου θέματος (εικόνα 10). Στη συγκεκριμένη διαδικασία ο καθηγητής συμπληρώνει την ανάλογη φόρμα ορίζοντας, μεταξύ άλλων και την κατηγορία του θέματος. Μπορεί επίσης να ανεβάσει και ένα αρχείο τύπου Word στο οποίο θα έχει περαιτέρω σχόλια ή απαιτήσεις για το θέμα. Έπειτα από την συμπλήρωση των πεδίων επιλέγεται η καταχώρηση και στη συνέχεια η αποθήκευση στη βάση δεδομένων.

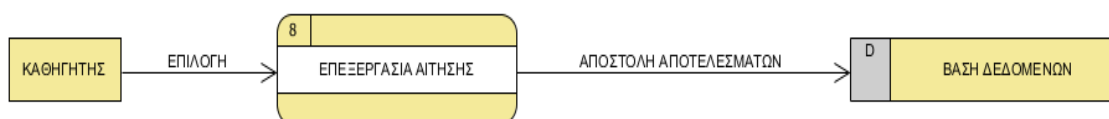


Εικόνα 10: Καταχώρηση Θέματος



Εικόνα 11: Επεξεργασία Εργασιών

Η παραπάνω διαδικασία προστέθηκε για την διευκόλυνση του καθηγητή σε περίπτωση που διαπιστώσει κάποιο λάθος στο κείμενο ή ακόμα αν επιθυμεί τη διαγραφή του θέματος. Έπειτα από οποιαδήποτε ενέργεια επεξεργασίας στα θέματα, η διαδικασία αυτή ενημερώνει τη βάση δεδομένων.



Εικόνα 12: Επεξεργασία Αίτησης

Στην επιλογή αυτή ο καθηγητής καλείται να εγκρίνει ή να απορρίψει την αίτηση του κάθε φοιτητή. Αφού ολοκληρωθεί η επιλογή από το χρήστη-καθηγητή γίνεται η αποστολή των αποτελεσμάτων μέσω της διαδικασίας στη βάση δεδομένων.

3. ΤΕΧΝΟΛΟΓΙΕΣ ΚΑΙ ΕΡΓΑΛΙΑ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ

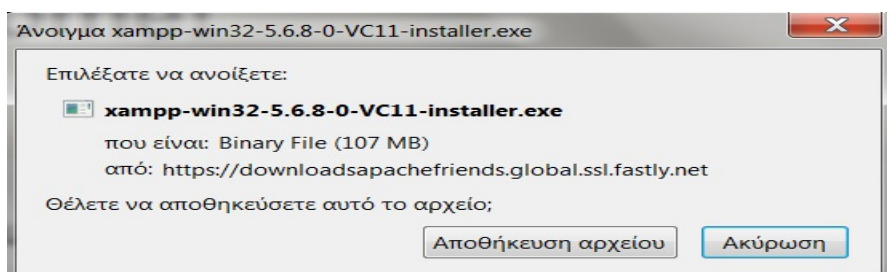
3.1 ΧΑΜΡΡ



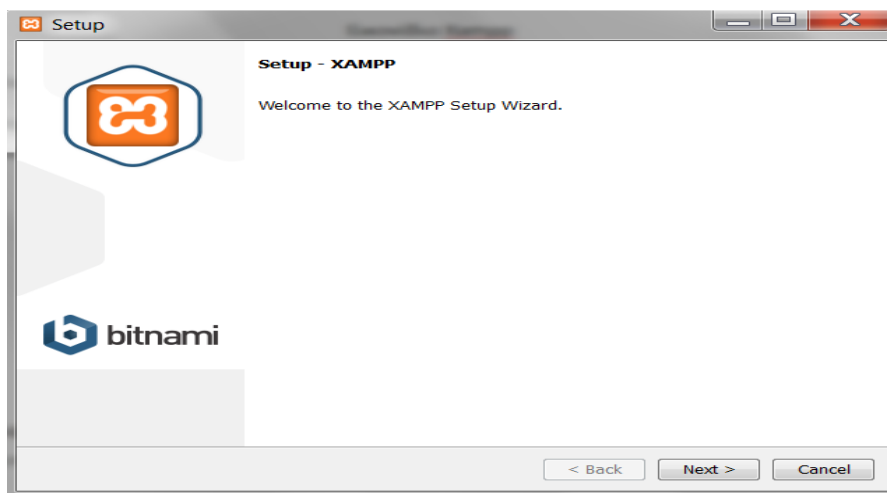
Εικονίδιο Χαμρρ

Τα παρακάτω βήματα περιγράφουν την εγκατάσταση του Χαμρρ σε τοπικό μηχάνημα.

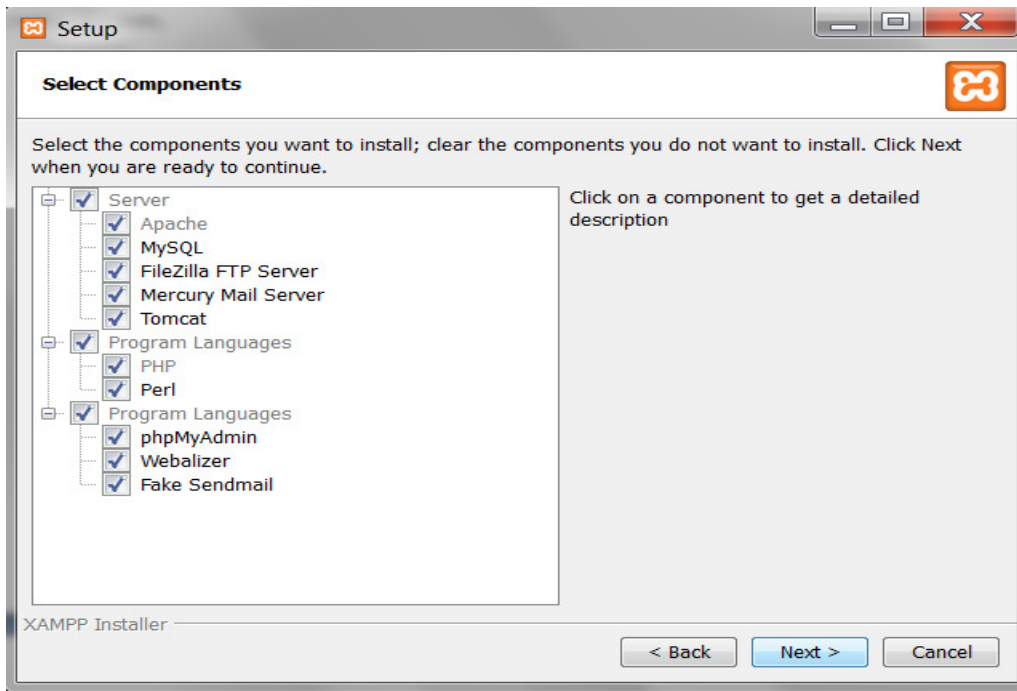
Βήμα 1ο: Πηγαίνουμε στην διεύθυνση <https://www.apachefriends.org/index.html> και κατεβάζουμε την αντίστοιχη έκδοση για τα Windows.



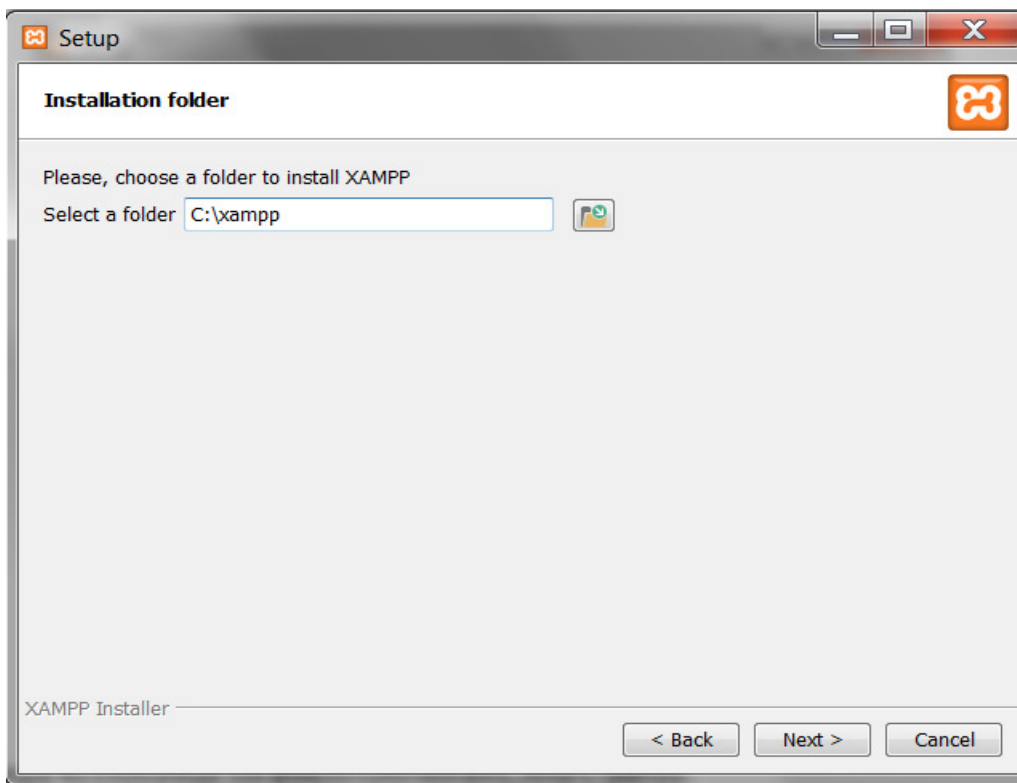
Βήμα 2ο: Αφού αποθηκεύσουμε το .exe αρχείο στον τοπικό δίσκο, πατάμε διπλό κλικ για να ξεκινήσει η εγκατάσταση.



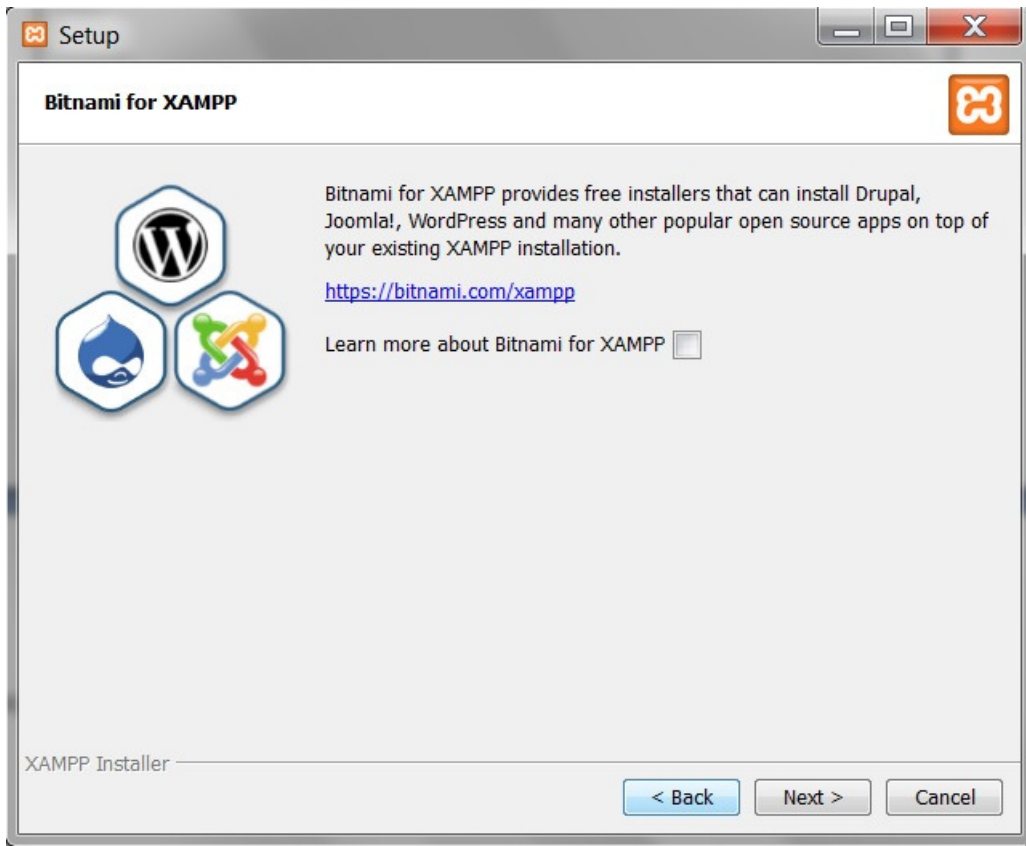
Βήμα 3ο: Πατάμε Next.



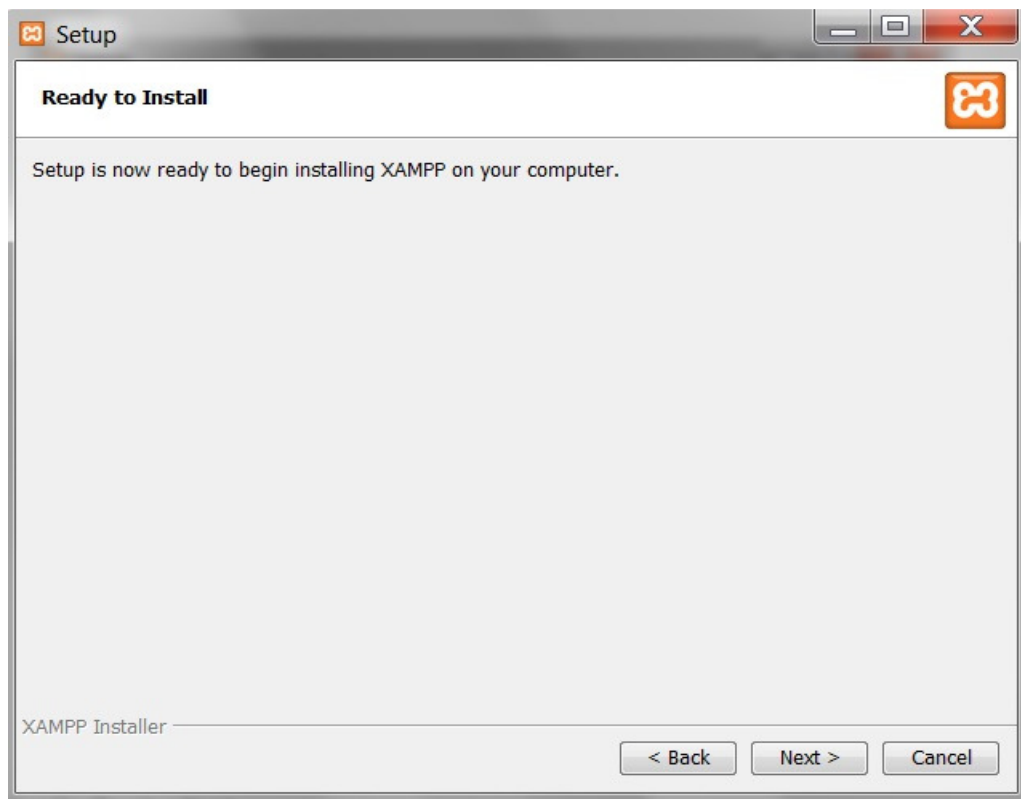
Βήμα 3ο: Επιλέγουμε τα περιεχόμενα που θέλουμε να έχουμε στην εφαρμογή και στη συνέχεια πατάμε Next.



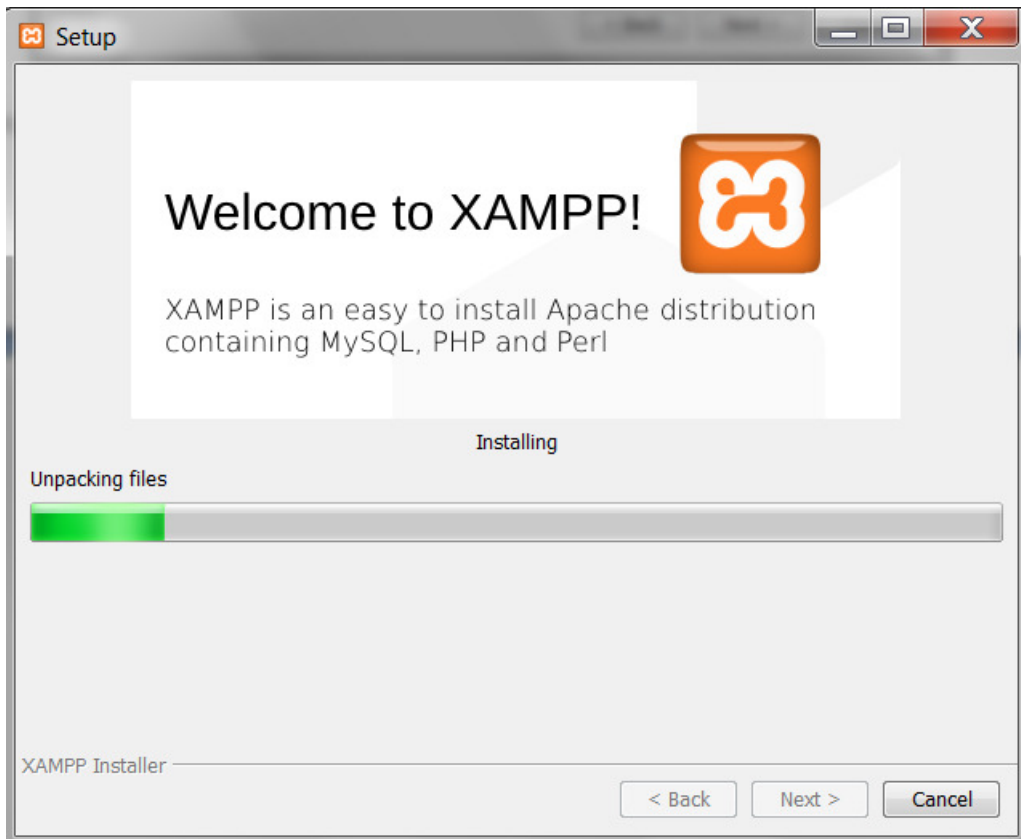
Βήμα 4ο: Επιλέγουμε τον φάκελο εγκατάστασης, έστω C:\xampp και πατάμε Next.



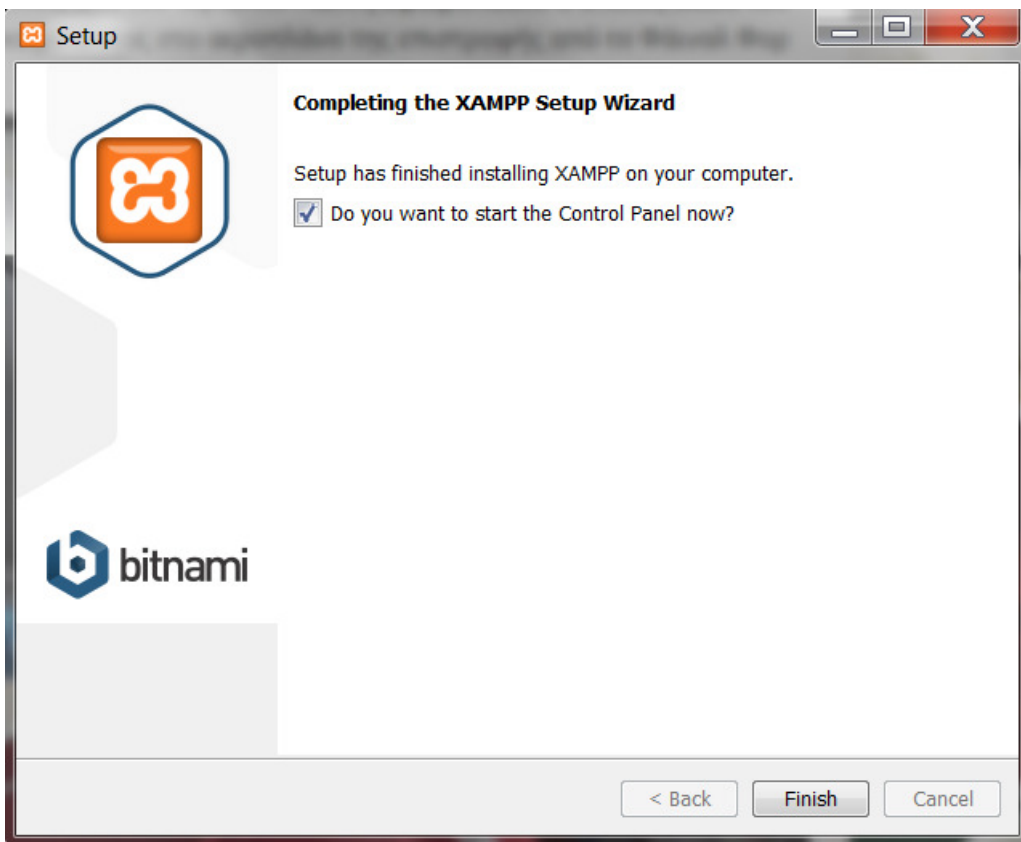
Βήμα 5ο: Πατάμε Next



Βήμα 6ο: Πατάμε Next



Βήμα 7ο: Αφού ολοκληρωθεί η εγκατάσταση πατάμε Finish



Βήμα 8ο:Αφού ολοκληρώθηκε η εγκατάσταση του Χαμpp πηγαίνουμε στην διαδρομή C:\xampp\setup_xampp.bat τρέχουμε μια φορά το αρχείο και πατάμε ένα οποιοδήποτε πλήκτρο.

```

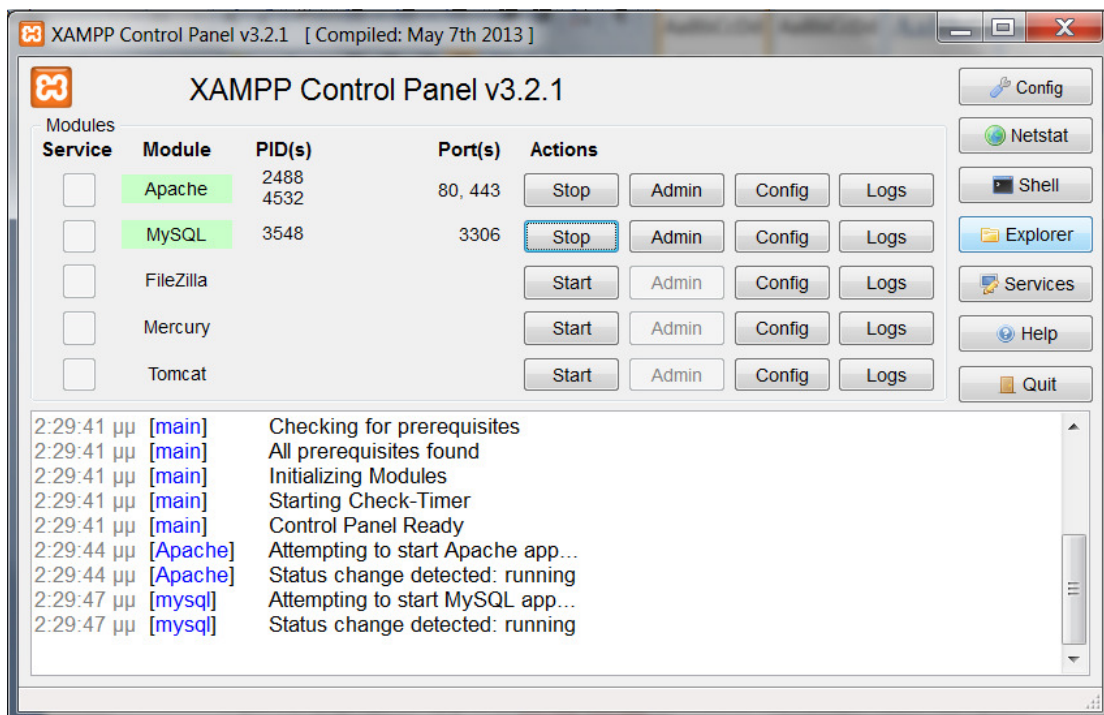
C:\Windows\system32\cmd.exe
[ XAMPP ]: Test php.exe with php\php.exe -n -d output_buffering=0 --version ...
PHP 5.6.8 (cli) (built: Apr 15 2015 15:07:09)
Copyright (c) 1997-2015 The PHP Group
Zend Engine v2.6.0, Copyright (c) 1998-2015 Zend Technologies
[ XAMPP ]: Test for the php.exe successfully passed. Good!
##### END XAMPP TEST SECTION #####
#####

#####
# ApacheFriends XAMPP setup win32 Version #
#-----#
# Copyright (c) 2002-2015 ApacheFriends 5.6.8 #
# #
#-----#
# Authors: Kay Vogelgesang <kvo@apachefriends.org> #
# Carsten Wiedmann <webmaster@wiedmann-online.de> #
#####

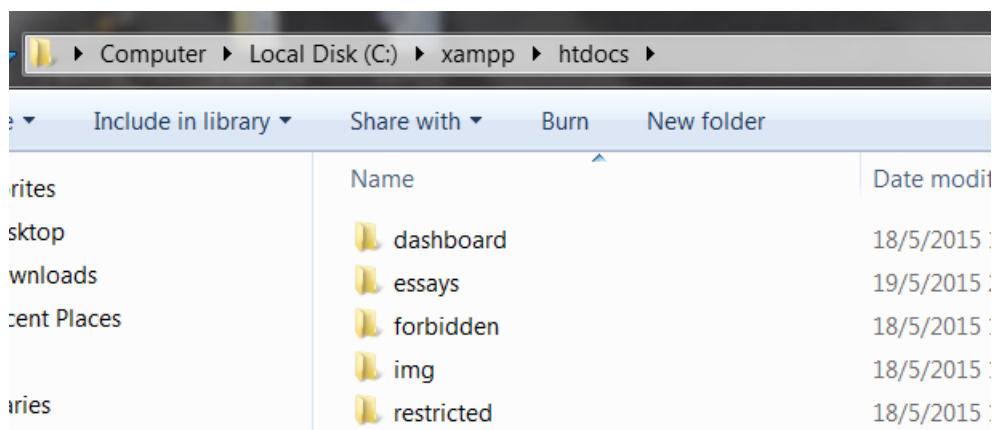
Sorry, but ... nothing to do!

Press any key to continue . . . .
  
```

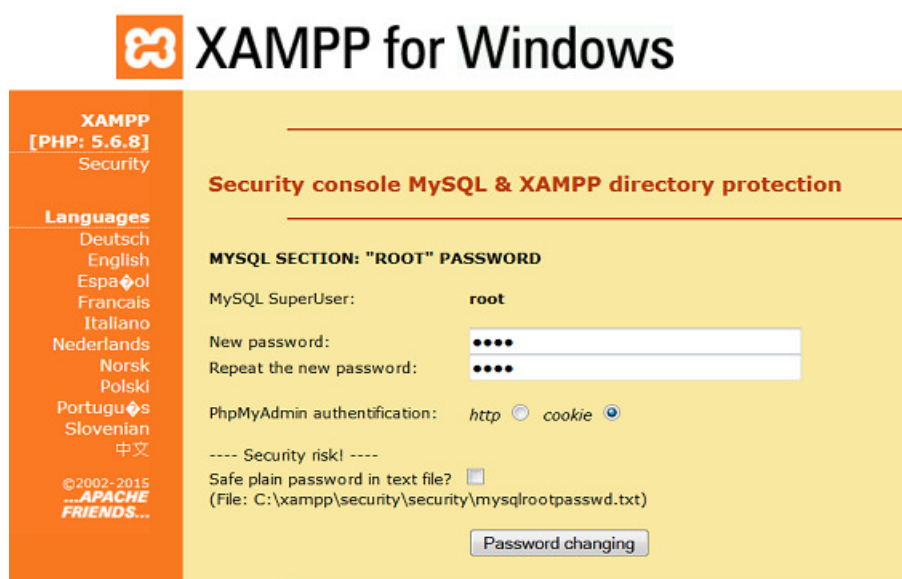
Βήμα 9ο:Τρέχουμε το αρχείο C:\xampp\xampp-control.exe και πατάμε "Start" σε Apache και MySQL.



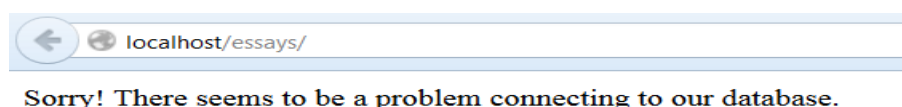
Βήμα 10ο:Αφού έχουμε φτιάξει έναν φάκελο με όνομα "essays", με όλα τα αρχεία της εργασίας ,τον αντιγράφουμε μέσα στο C:\xampp\htdocs\




Βήμα11ο:Με έναν φυλλομετρητή (browser) πηγαίνουμε στην σελίδα <http://localhost/security/index.php> ορίζουμε το password του root για τη σύνδεση στη MySQL και πατάμε "Password changing".



Βήμα 12ο: Ανοίγουμε ένα browser και πάμε στη σελίδα <http://localhost/essays/>.Θα μας δώσει το site που φτιάξαμε αλλά με μηνύματα λάθους γιατί δε θα μπορεί να συνδεθεί στη βάση δεδομένων.Θα πρέπει λοιπόν να κάνουμε εισαγωγή της βάσης στη MySQL.Ο πιο απλός τρόπος είναι μεσω phpmyadmin.



Πηγαίνουμε λοιπόν στην διεύθυνση <http://localhost/rhpmysadmin> και συνδεόμαστε με Όνομα χρήστη "root" και κωδικό πρόσβασης, αυτόν που δώσαμε στο προηγούμενο βήμα.



Καλωσήρθατε στο rhpmysAdmin

Γλώσσα - Language

Ελληνικά - Greek

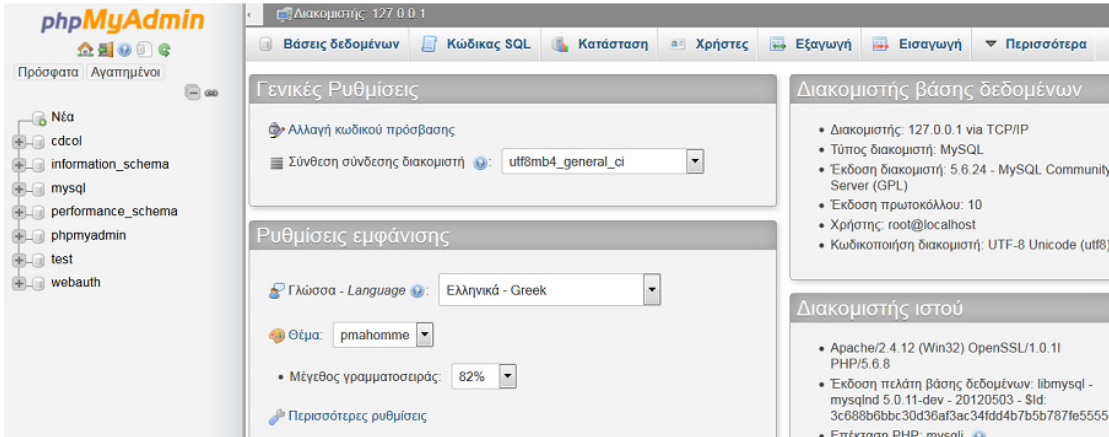
Σύνδεση

Όνομα χρήστη:

Κωδικός πρόσβασης:

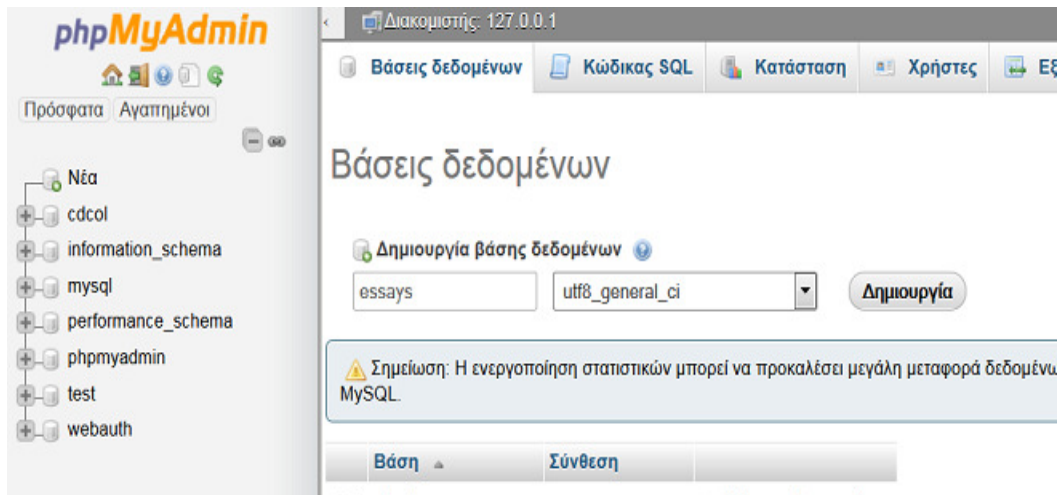
Εκτέλεση

Βήμα 13ο: Έχοντας συνδεθεί, δημιουργούμε μια νέα βάση δεδομένων με όνομα "essays".



The screenshot shows the rhpmysAdmin interface with the following sections:

- Γενικές Ρυθμίσεις**:
 - Αλλαγή κωδικού πρόσβασης
 - Σύνθεση σύνδεσης διακομιστή: utf8mb4_general_ci
- Ρυθμίσεις εμφάνισης**:
 - Γλώσσα - Language: Ελληνικά - Greek
 - Θέμα: pmahomme
 - Μέγεθος γραμματοσειράς: 82%
- Διακομιστής βάσης δεδομένων**:
 - Διακομιστής: 127.0.0.1 via TCP/IP
 - Τύπος διακομιστή: MySQL
 - Έκδοση διακομιστή: 5.6.24 - MySQL Community Server (GPL)
 - Έκδοση πρωτοκόλλου: 10
 - Χρήστης: root@localhost
 - Κωδικοποίηση διακομιστή: UTF-8 Unicode (utf8)
- Διακομιστής ιστού**:
 - Apache/2.4.12 (Win32) OpenSSL/1.0.11 PHP/5.6.8
 - Έκδοση πελάτη βάσης δεδομένων: libmysql - mysqlnd 5.0.11-dev - 20120503 - Sld: 3c688b6bbc30d36af3ac34fdd4b7b5b787fe5555
 - Επέκταση PHP: mysqli



Βήμα 14ο:Αφού δημιουργήσαμε την βάση ,την επιλέγουμε και κάνουμε "Εισαγωγή" του αρχείου essays.sql ,που έχουμε φτιάξει, για να δημιουργηθούν οι πίνακες της βάσης μας, από την επιλογή "Αναζήτηση", επιλέγοντας το αρχείο που είναι αποθηκευμένο στον τοπικό υπολογιστή μας και πατώντας το κουμπί της "Εκτέλεσης".

Εισαγωγή στη βάση δεδομένων «essays»

Αρχείο για εισαγωγή:

Το αρχείο μπορεί να συμπιεστεί (gzip, bzip2, zip) ή να αποσυμπιεστεί.

Το όνομα του συμπιεσμένου αρχείου πρέπει να έχει επέκταση: **.[format].[compression]**. Παράδειγμα: **.sql.zip**

Περιηγηθείτε στον υπολογιστή σας: essays.sql (Μέγιστο μέγεθος: 2,048KB)

Μπορείτε να σύρετε και να αφήσετε ένα αρχείο σε οποιαδήποτε σελίδα.

Σύνολο χαρακτήρων του αρχείου:

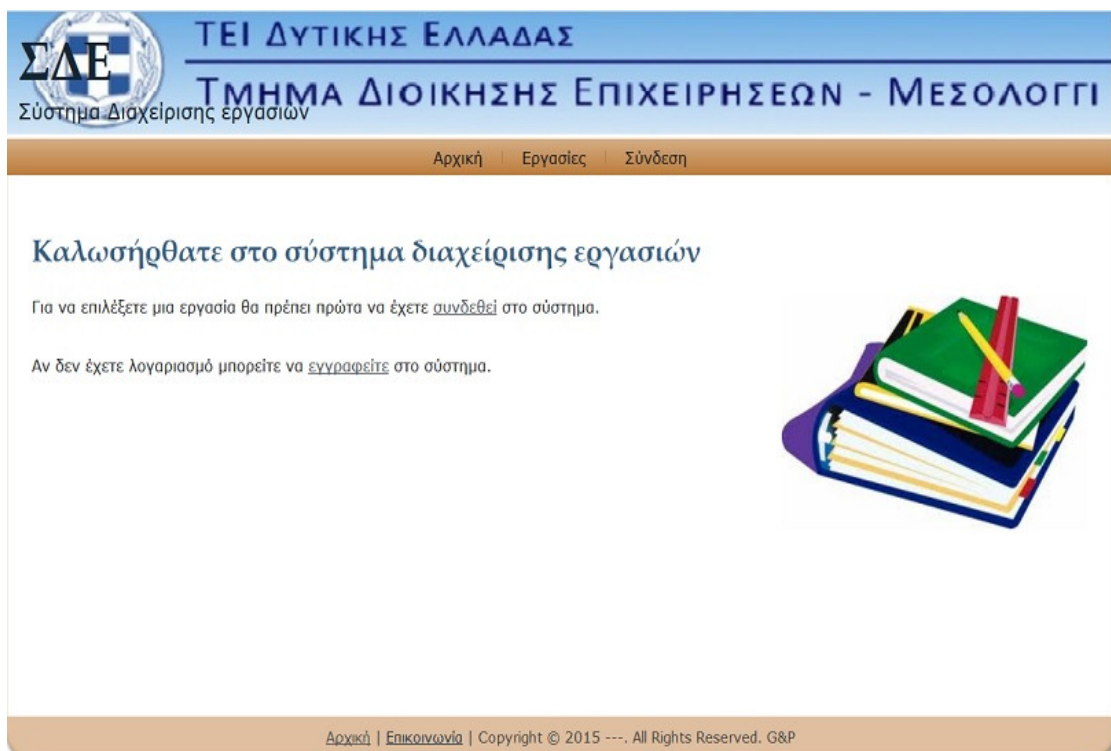
Για να είναι επιτυχής η εισαγωγή της βάσης πρέπει να πάρουμε το παρακάτω μήνυμα

✓ Η εισαγωγή ολοκληρώθηκε επιτυχώς, 21 ερωτήματα εκτελέστηκαν. (essays.sql)

Βήμα 15ο:Η βάση μας πλέον είναι έτοιμη, το μόνο που πρέπει να τσεκάρουμε είναι το password που συνδέει την βάση να είναι το σωστό. Πηγαίνουμε λοιπόν στο αρχείο C:\xampp\htdocs\essays\includes\config.php και το ελέγχουμε.

```
config.php
1 <?php
2 ob_start();
3 session_start();
4
5 define('DBHOST','localhost');
6 define('DBUSER','root');
7 define('DBPASS','1234');
8 define('DBNAME','essays');
9
10
11 $conn = @mysql_connect (DBHOST, DBUSER, DBPASS);
12 $conn = @mysql_select_db (DBNAME);
13 if(!$conn){
14     die( "Sorry! There seems to be a problem connect
```

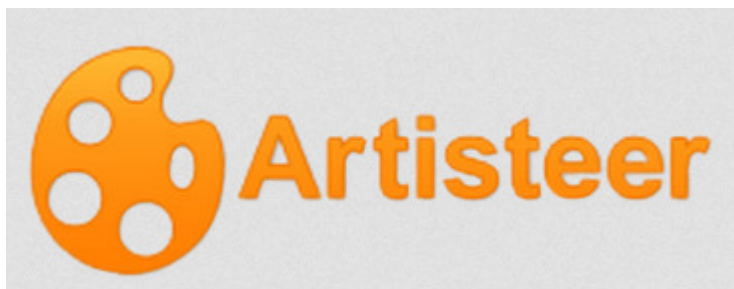
Βήμα 16ο:Τέλος ανοίγουμε ένα browser και πηγαίνουμε στην σελίδα <http://localhost/essays> ,τόρα θα πρέπει να μπορούμε να συνδεθούμε και να δούμε την εφαρμογή.



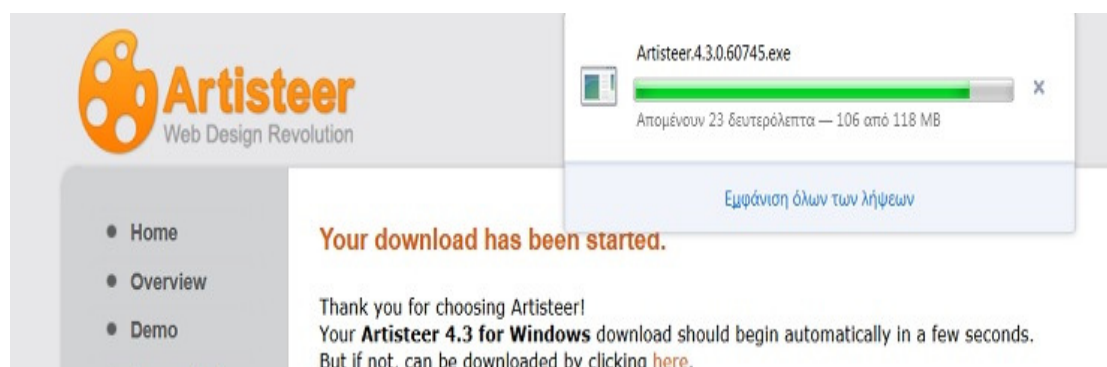
The screenshot shows a web application interface. At the top, there is a header with the logo of the University of Western Greece (ΣΔΕ) and the text "ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ" and "ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ". Below the header, there is a navigation menu with "Αρχική", "Εργασίες", and "Σύνδεση". The main content area features a heading "Καλωσήρθατε στο σύστημα διαχείρισης εργασιών" and two paragraphs of text: "Για να επιλέξετε μια εργασία θα πρέπει πρώτα να έχετε [συνδεθεί](#) στο σύστημα." and "Αν δεν έχετε λογαριασμό μπορείτε να [εγγραφείτε](#) στο σύστημα." To the right of the text is an illustration of a stack of books with a red pen and a yellow pencil. At the bottom, there is a footer with "Αρχική | [Επικοινωνία](#) | Copyright © 2015 ----. All Rights Reserved. G&P".

4. ΣΧΕΔΙΑΣΜΟΣ USER INTERFACE

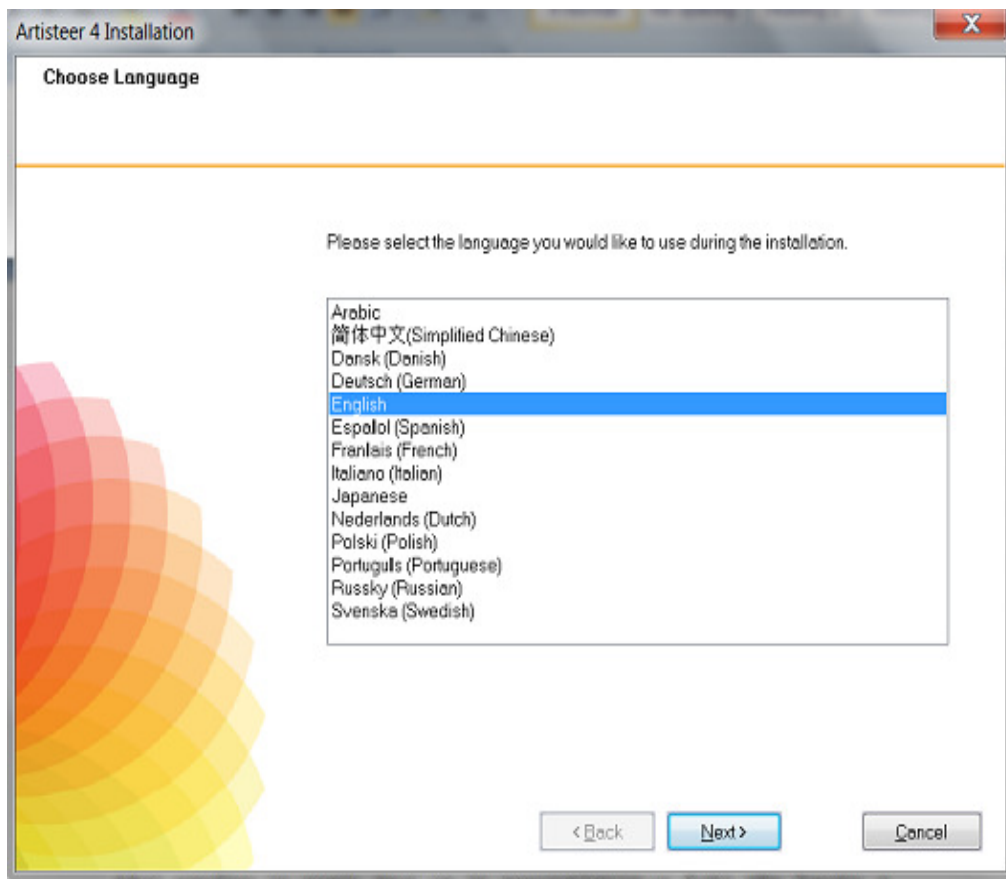
Για τον σχεδιασμό του User Interface χρησιμοποιήθηκε το Artisteer.



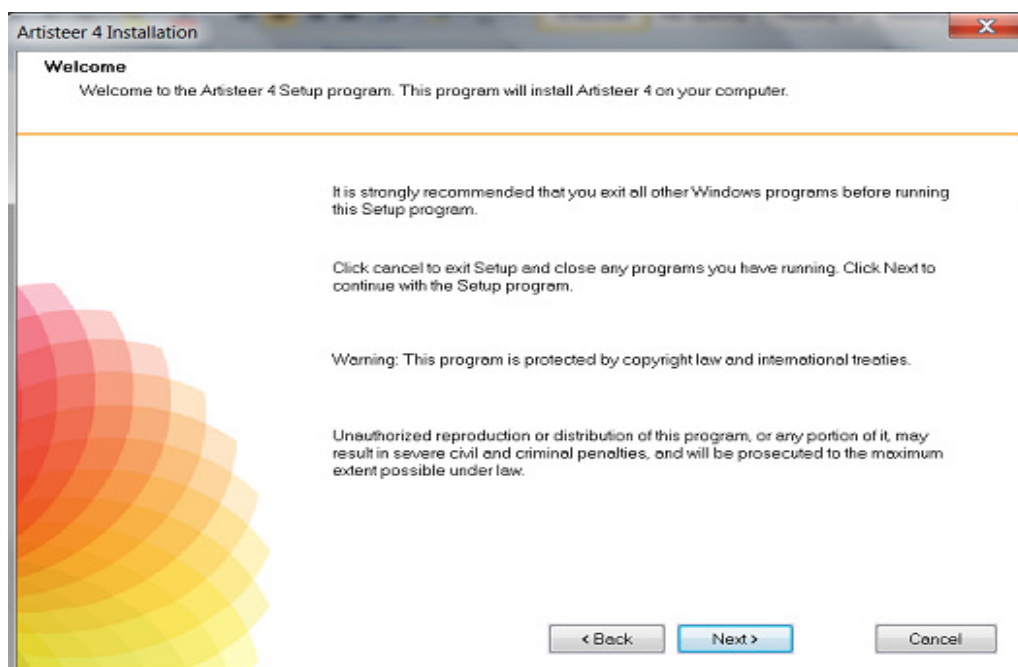
Για να το κατεβάσουμε πηγαίνουμε στην σελίδα <http://www.artisteer.com/> και επιλέγουμε την ελεύθερη έκδοση.



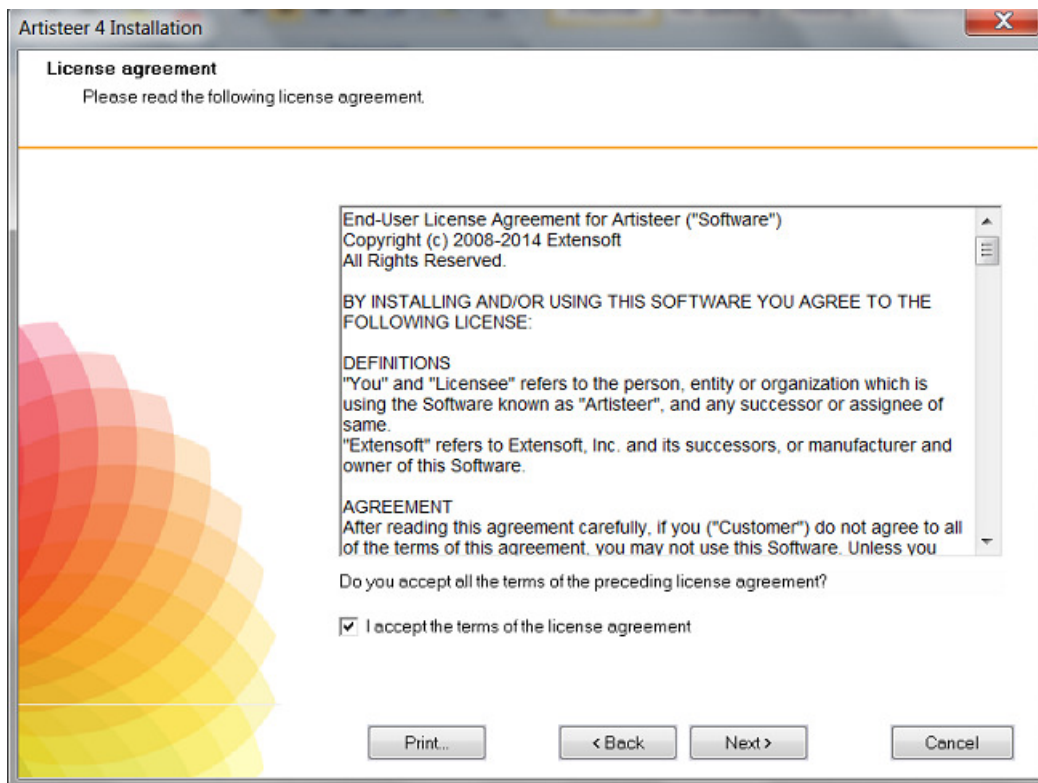
Βήμα 1ο:Αφού κατεβήκε το αρχείο παμε να το εγκαταστήσουμε,με διπλό κλικ ξεκινάει η εγκατάσταση.



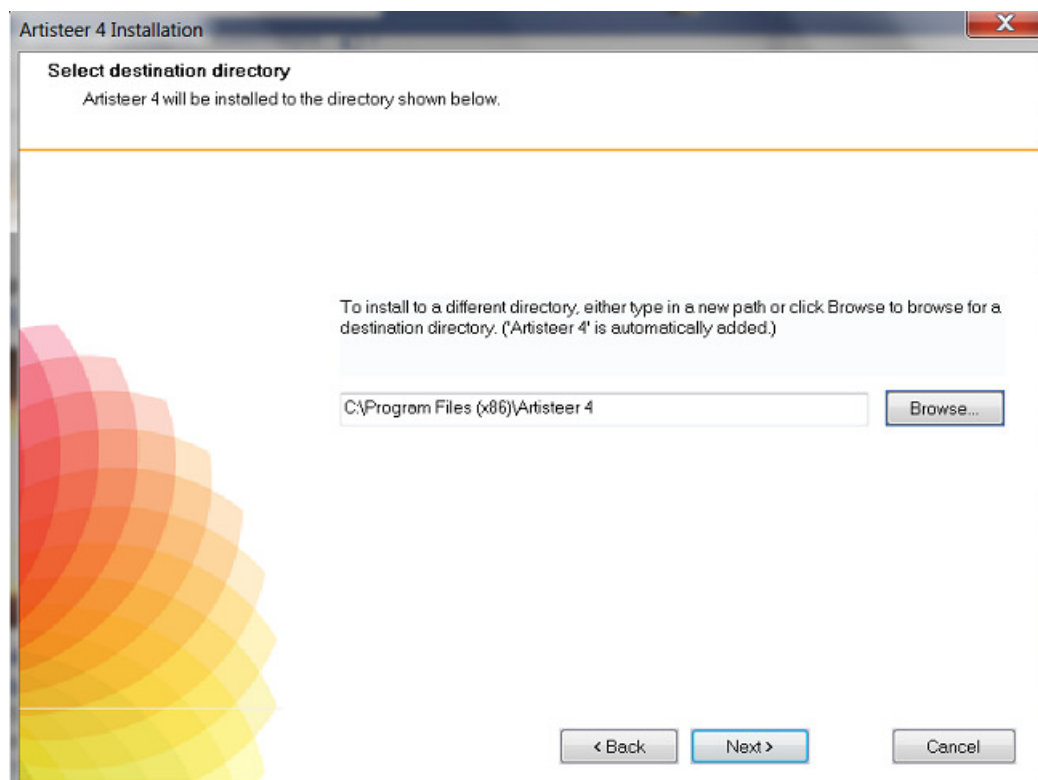
Βήμα 2ο: Πατάμε Next



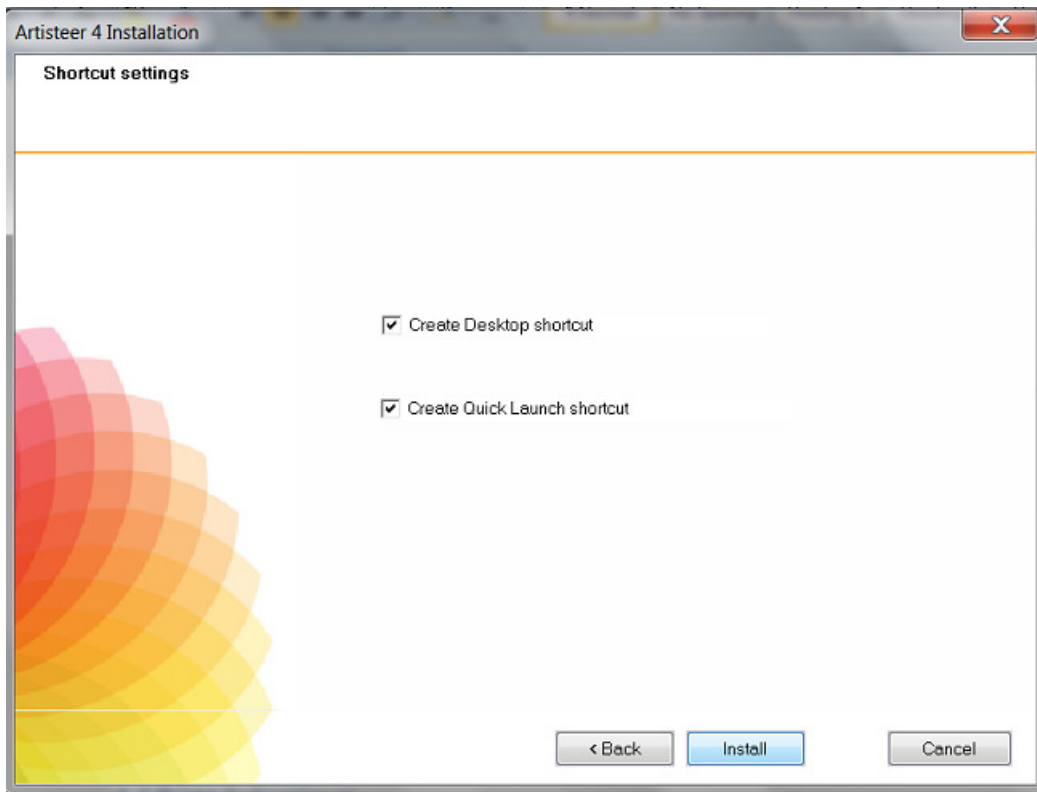
Βήμα 3ο: Πατάμε Next



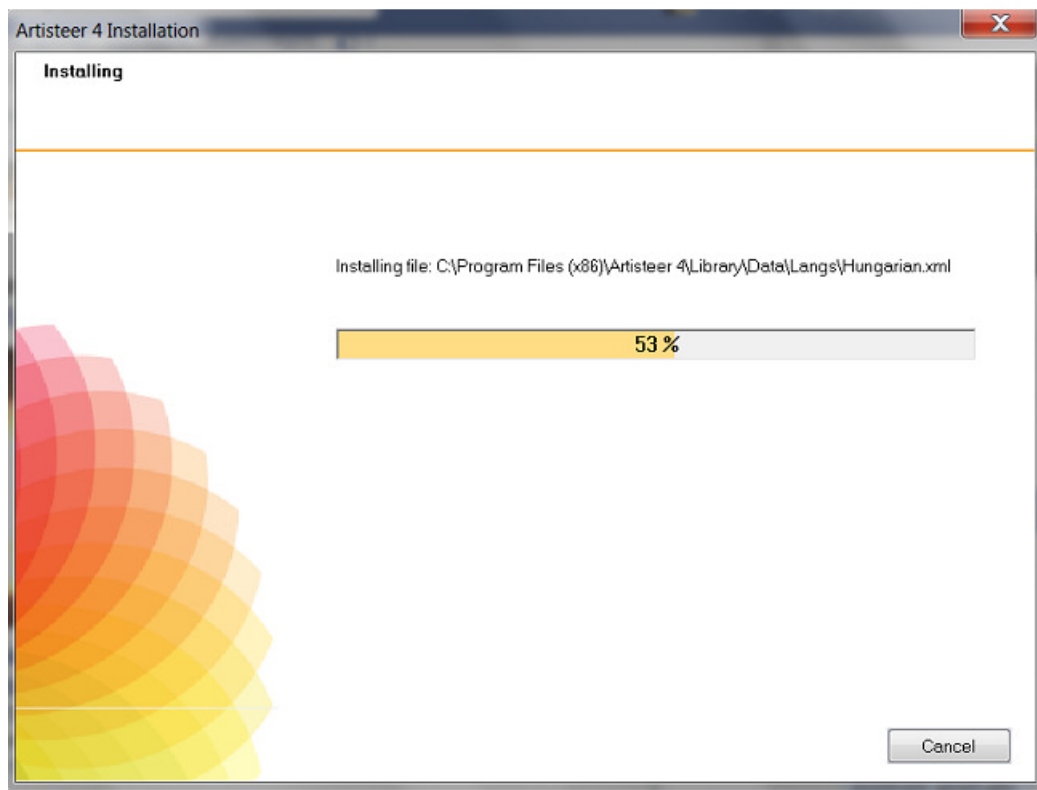
Βήμα 4ο:Αφού επιλέξουμε για την άδεια χρήσης πατάμε Next



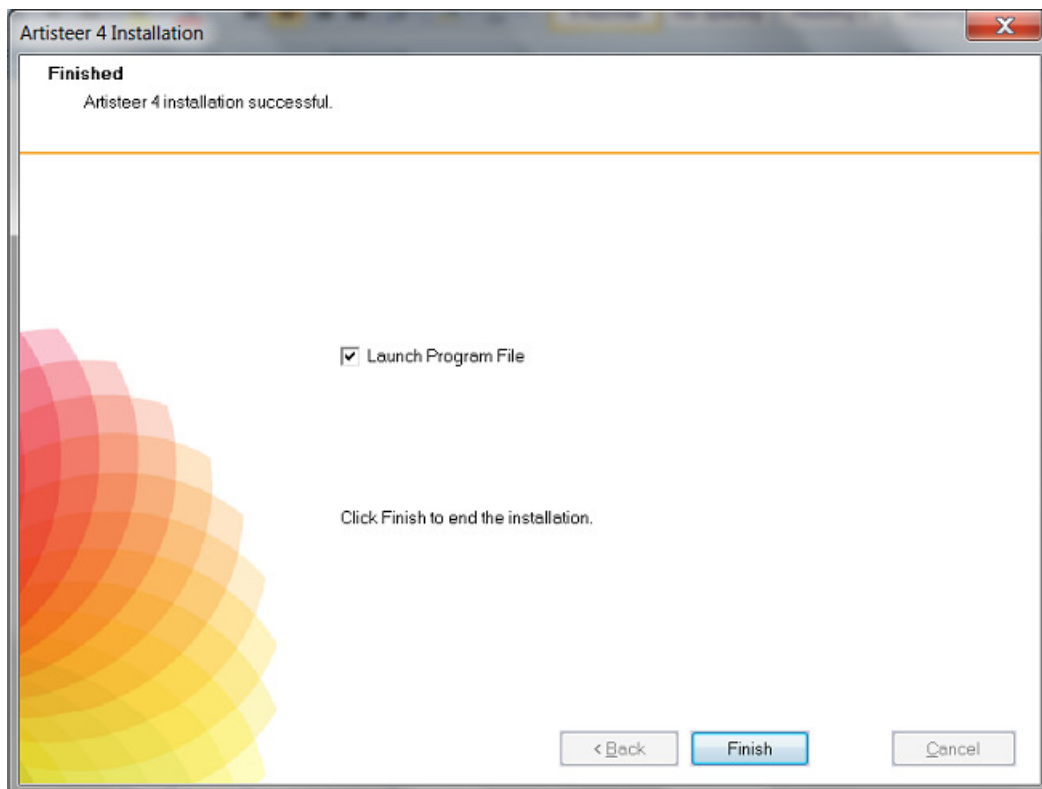
Βήμα 5ο:Επιλέγουμε τον φάκελο προορισμού και πατάμε Next



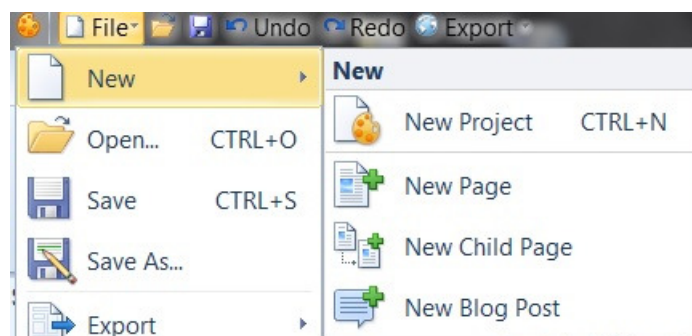
Βήμα 6ο:Επιλέγουμε αν θέλουμε συντομεύσεις του προγράμματος και πατάμε Install.



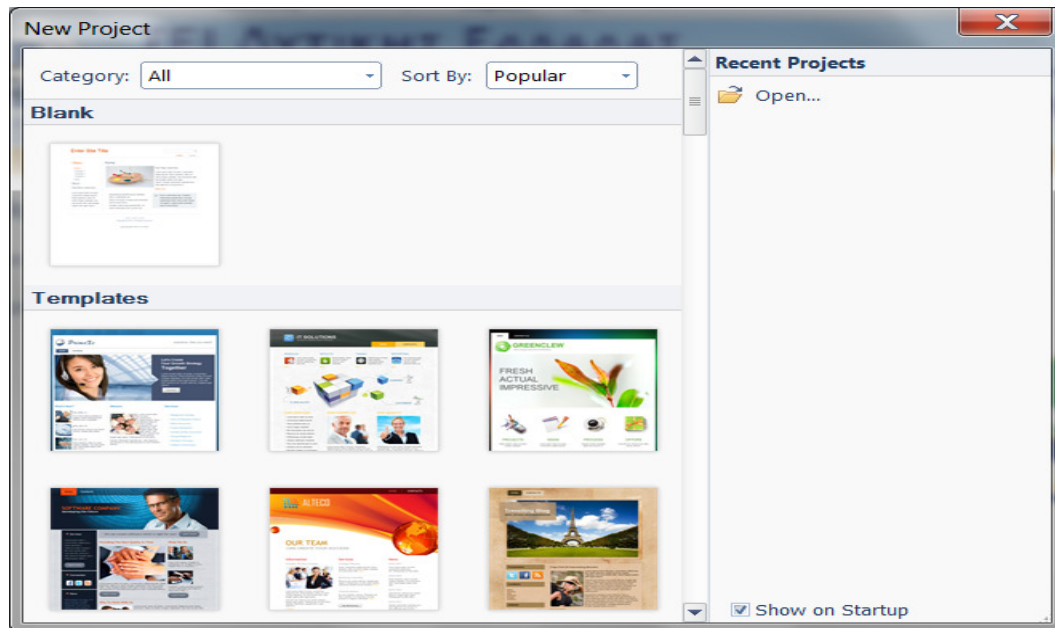
Βήμα 7ο:Αφου ολοκληρωθεί η εγκατάσταση πατάμε Finish.



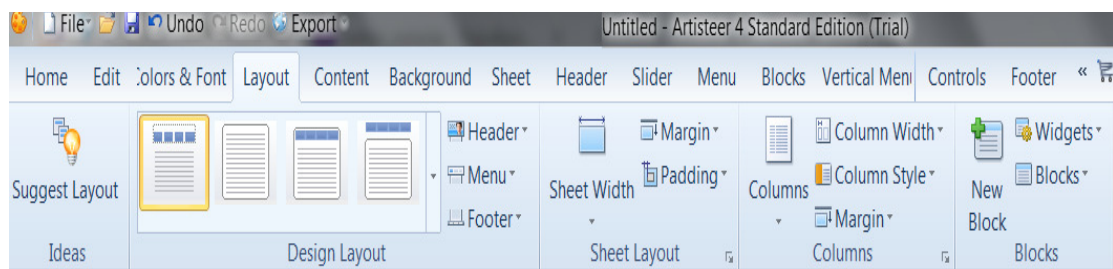
Για να δημιουργήσουμε ένα νέο σχέδιο επιλέγουμε File -> New -> New Project



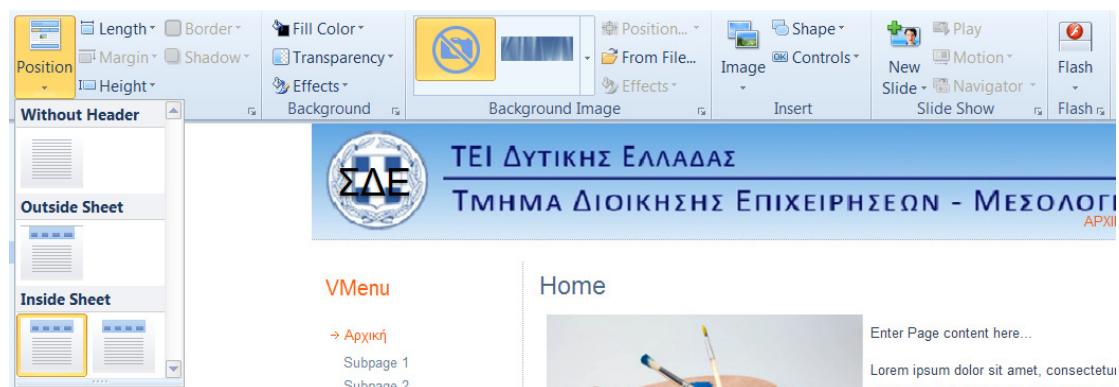
Στην επόμενη σελίδα επιλέγουμε Blank Project



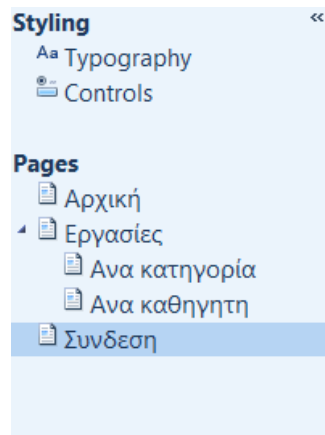
Αρχικά επιλέγουμε το σχέδιο που θέλουμε για την σελίδα μας απο την επιλογή Layout



Για την διαμόρφωση της κεφαλίδας επιλέγουμε απο το μενού Header τις επιλογές που θέλουμε. Την επιλογή Position για την θέση της κεφαλίδας, για να προσθέσουμε μια εικόνα την επιλογή Image κ.ο.κ.

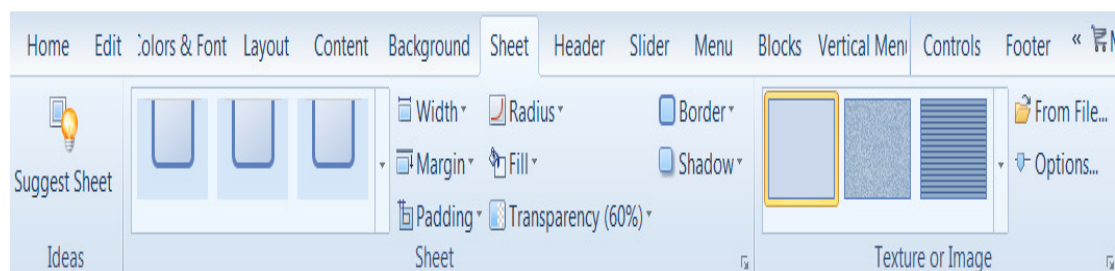


Για την διαμόρφωση του υποσέλιδου επιλέγουμε το Footer κάνοντας και εκεί τις απαραίτητες αλλαγές που θέλουμε.

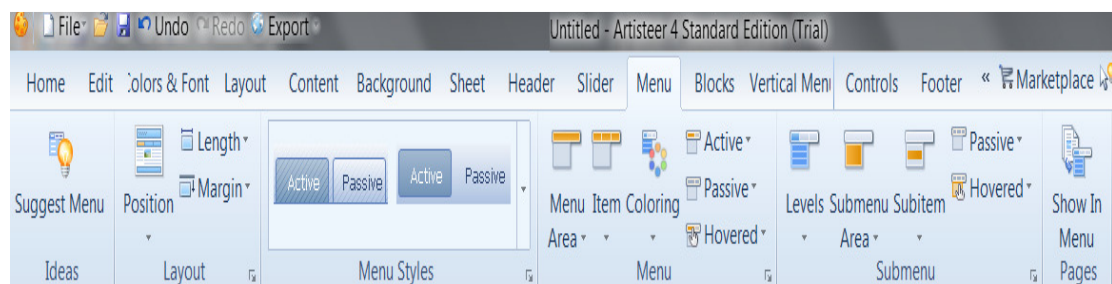


Στην πλαινή μπάρα,προσθέτουμε τις σελίδες που θέλουμε .Μπορούμε να επιλέξουμε την κάθε σελίδα και να προσθέσουμε οτι περιεχόμενο θέλουμε,αυτο βέβαια μπορούμε να το κάνουμε και μετά μέσα στον κώδικα.

Απο το μενού Sheet επιλέγουμε το φύλλο που θέλουμε.



Για την διαμόρφωση του μενού επιλέγουμε απο την κατηγορία Menu τις αντίστοιχες αλλαγές που θέλουμε.Για την θέση του μενού,τον τύπο του κ.ο.κ



Ο όρος διεπαφή χρήστη (user interface) είναι το σύνολο των συστατικών ενός συστήματος το οποίο επιτρέπει αμφίδρομη επικοινωνία μεταξύ συστήματος και χρήστη.Ο σχεδιασμός της διεπαφής χρήστη σε ένα σύστημα αποτελεί ένα απο τα βασικότερα στοιχεία που θα καθορίσουν την επιτυχία της εφαρμογής.Μια δύσχρηστη διεπαφή μπορεί να απορριφθεί απο τον χρήστη και ας δουλεύει σωστά.Για τον σχεδιασμό λοιπόν της διεπαφής πρέπει να ληφθούν κάποιοι παράγοντες.

- ✓ ποιοί θα χρησιμοποιήσουν την εφαρμογή
- ✓ οι δυνατότητες αυτών που θα την χρησιμοποιήσουν
- ✓ σκοπός της εφαρμογής

Στην συγκεκριμένη εργασία αυτοί που θα χρησιμοποιήσουν την εφαρμογή θα είναι φοιτητές και καθηγητές για την διαχείριση πτυχιακών εργασιών, οι οποίοι είναι καλοί γνώστες του αντικειμένου. Παρ' όλα 'αυτά δημιουργήσαμε μια απλή εφαρμογή με μενού, όπου οι χρήστες δεν χρειάζεται να γνωρίζουν εντολές. Η γραφική διεπαφή χρήστη αυξάνει τη λειτουργικότητα, την ευκολία στη χρήση, την αποτελεσματικότητα και την ταχύτητα του διαλόγου μεταξύ ανθρώπου και μηχανής.

Από τα βασικότερα στοιχεία τα οποία πρέπει με προσοχή να συνεκτιμώνται και να λαμβάνονται υπόψη κατά τη δημιουργία της διεπαφής χρήσης μιας εφαρμογής, είναι όσα σχετίζονται με την επιλογή χρωμάτων των στοιχείων της εφαρμογής και τη σχεδίαση των μηνυμάτων λαθών. Το χρώμα θεωρείται βασικό συστατικό στη σχεδίαση της οθόνης μιας εφαρμογής. Η σωστή χρήση του χρώματος μπορεί να αποδειχθεί ένας αποτελεσματικός μηχανισμός για την επικοινωνία, την προσοχή και τον καθορισμό των σημείων που θέλει να αναδείξει μία εφαρμογή. Το άπρο χρώμα που χρησιμοποιήθηκε στο φόντο της εφαρμογής είναι από τα πιο δημοφιλή χρώματα για τις ηλικίες τις οποίες απευθύνεται, καθώς φαντάζει φιλικό στα μάτια τους. Για να έχει συνοχή χρησιμοποιήθηκε και το καφέ, ένα χρώμα συνηθισμένο και γήινο. Δεν προτιμήσαμε το ρίσκο των έντονων χρωμάτων για την τυχόν αποτροπή του χρήστη από την χρησιμοποίηση της εφαρμογής. Βασικό στοιχείο μιας εφαρμογής είναι και το πως παρουσιάζει στον χρήστη τα τυχόν λάθη που θα προκύψουν από την εφαρμογή της

5. ΑΞΙΟΛΟΓΗΣΗ ΕΦΑΡΜΟΓΗΣ ΚΑΙ ΠΡΟΒΛΗΜΑΤΑ

5.1 Προβλήματα

Στην παραπάνω εφαρμογή μπορεί να συναντήσουμε κάποια προβλήματα κατά τη διάρκεια της χρήσης της.

- Η εφαρμογή είναι σχεδιασμένη για μονομερή ομάδα, έτσι δεν υπάρχει δυνατότητα να εγκριθεί αίτηση εκδήλωσης ενδιαφέροντος από 2 φοιτητές ταυτόχρονα.
- Κατά την εγγραφή ενός νέου χρήστη τα στοιχεία που απαιτεί το σύστημα είναι το username, password, όνομα και e-mail. Κατά συνέπεια κάθε επισκέπτης, ανεξάρτητα αν είναι φοιτητής της εκάστοτε σχολής ή όχι έχει τη δυνατότητα να δημιουργήσει λογαριασμό και να κάνει χρήση της εφαρμογής. Στα πλαίσια όμως της ενσωμάτωσης της εφαρμογής στο κεντρικό σύστημα της οργάνωσης της σχολής οι κωδικοί μπορεί παραλαμβάνονται από την γραμματεία του τμήματος.
- Ένα ακόμα πρόβλημα που εμφανίζεται να υπάρχει στην εφαρμογή είναι ότι κάθε χρήστης που έχει ξεχάσει τον κωδικό πρόσβασης του, δεν είναι σε θέση να τον επαναφέρει. Αυτό συμβάλει στην ανούσια συμφόρηση του συστήματος με ανενεργούς λογαριασμούς.
- Ένα ακόμα πρόβλημα που εμφανίζεται στο σύστημα, το οποίο αφορά κυρίως τους καθηγητές, είναι ότι δεν υπάρχει δυνατότητα πολλαπλής επιλογής των αιτήσεων για απόρριψη. Έτσι ο καθηγητής για να απορρίψει 20 αιτήσεις θα πρέπει να πατήσει 20 φορές απόρριψη.

Μετά την ενσωμάτωση της εφαρμογής στο κεντρικό σύστημα οργάνωσης του Ιδρύματος είναι πιθανό να συναντήσουμε κάποια από τα παρακάτω προβλήματα.

- Μεγάλος αριθμός χρηστών

Κάθε Σχολή αποτελείται από πολλούς φοιτητές, κατά συνέπεια οι Εξυπηρετητές θα πρέπει να είναι σε θέση να υποστηρίξουν πιθανή ταυτόχρονη χρήση της εφαρμογής.

- Μεγάλος όγκος αρχείων

Όπως αναφέρθηκε παραπάνω ο καθηγητής θα πρέπει να ανεβάσει ένα αρχείο για κάθε πτυχιακή που θα περιλαμβάνει όλες τις λεπτομέρειες που χρειάζεται. Κάθε καθηγητής έχει την δυνατότητα να ανεβάσει πολλές πτυχιακές άρα ο όγκος των αρχείων που αντιστοιχεί ανα καθηγητή είτε θα πρέπει να είναι συγκεκριμένος είτε η βάση του συστήματος να μπορεί να το υποστηρίξει.

- Πολλές δηλώσεις ανά φοιτητή

Κάθε φοιτητής μπορεί να κάνει πάνω από μία αίτηση εκδήλωσης ενδιαφέροντος πτυχιακής. Αυτό μπορεί να δράσει καταχρηστικά φτάνοντας σε σημείο ο φοιτητής να δηλώνει μεγάλο αριθμό αιτήσεων χωρίς να είναι απαραίτητα αυτές που τον ενδιαφέρουν

πραγματικά. Κατά συνέπεια αυτό δίνει μεγάλο όγκο δουλειάς στους καθηγητές, αφού αυτοί θα είναι που θα επιλέξουν τελικά.

5.2 Αξιολόγηση εφαρμογής

Κύριος στόχος της εργασίας είναι η ανάπτυξη εφαρμογής ιστού με πλήρες γραφικό περιβάλλον χρησιμοποιώντας τεχνολογίες **PHP, MYSQL, HTML& JAVASCRIPT**.

Η παραπάνω εφαρμογή πρόκειται για μια οργανωμένη ηλεκτρονική εξέλιξη της εκπόνησης των εργασιών από τον καθηγητή προς τους φοιτητές. Η χρησιμότητα της εφαρμογής εμφανίζεται όταν ο φοιτητής στα πλαίσια της ολοκλήρωσης της σχολής του καλείται να διαλέξει την κατάλληλη πτυχιακή εργασία ώστε να αναδείξει τη γνώση που έλαβε κατά τη διάρκεια της φοίτησης του, καθώς κι ο καθηγητής τον κατάλληλο φοιτητή ανάμεσα σε πληθώρα αιτήσεων εκδήλωσης ενδιαφέροντος.

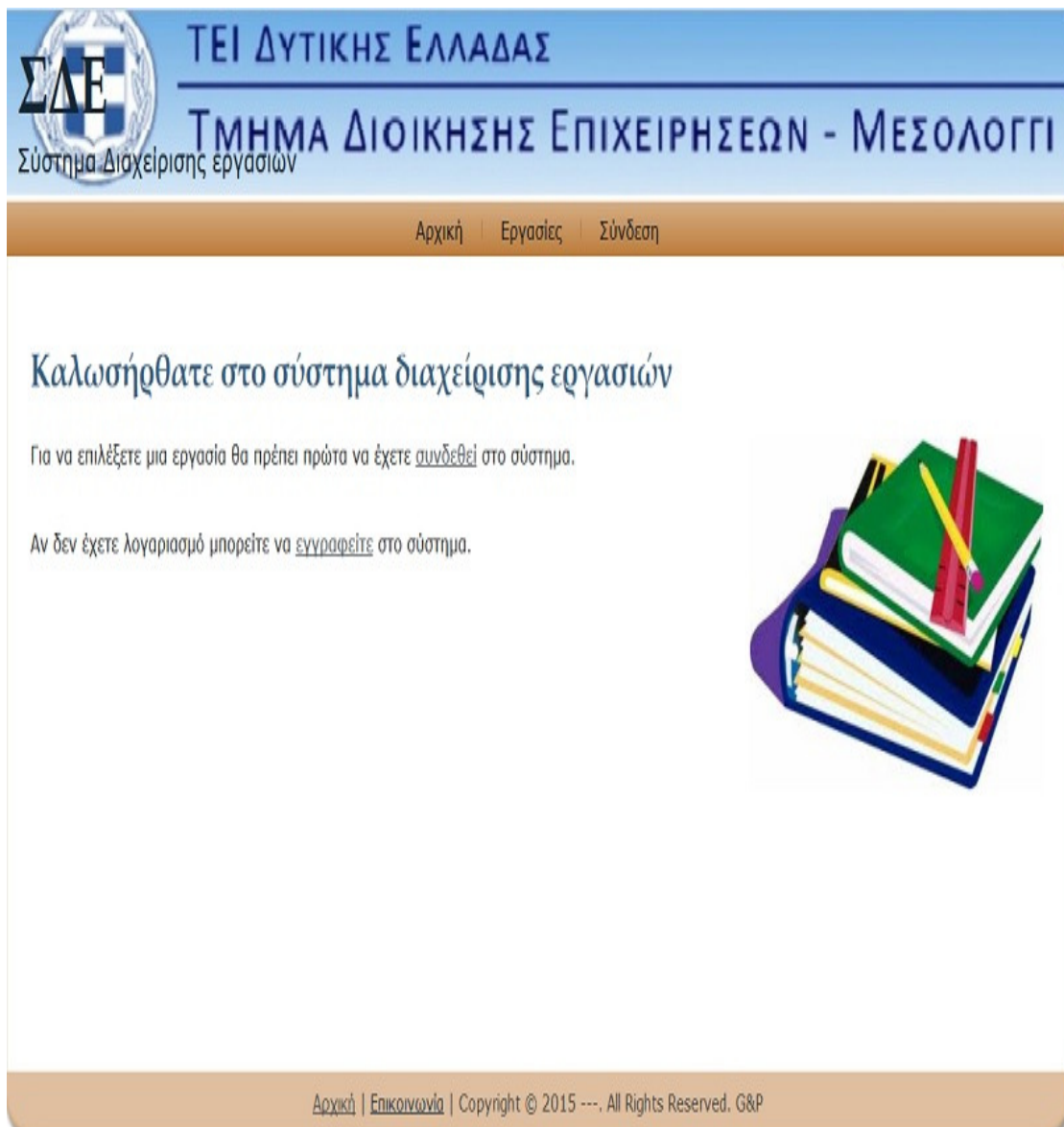
Η συγκεκριμένη εφαρμογή βρίσκεται σε πρώιμο στάδιο καθώς για να γίνει χρηστική θα πρέπει αρχικά να διορθωθούν τα προαναφερθείσα προβλήματα και στη συνέχεια να συνδεθεί με τη βάση δεδομένων του εκπαιδευτικού ιδρύματος έτσι ώστε να γίνει κομμάτι αυτού. Στη συνέχεια θα πρέπει να υποστεί βελτίωση σύμφωνα με τις ανάγκες της εκάστοτε σχολής. Η αναβάθμιση της εφαρμογής μπορεί επίσης να εξαρτηθεί και από την οργάνωση της βάσης. Έπειτα από το συγχρονισμό των χρηστών θα είναι εφικτό, ο καθηγητής να έχει πρόσβαση στις βαθμολογίες των μαθητών στην διάρκεια της φοίτησής τους, ή ακόμα μια αυτόματη πρόταση μπορεί να παρέχεται από το σύστημα για την καταλληλότερη πτυχιακή, δημιουργώντας κριτήρια σύμφωνα με τα ενδιαφέροντα του κάθε φοιτητή, όπως για παράδειγμα, συμμετοχή σε σεμινάρια ή επιλογή προαιρετικών μαθημάτων.

6. ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ (User Guide)

Έχοντας,λοιπόν εγκαταστήσει το ΧΑΜΡΡ,θα προχωρήσουμε βλέποντας πως λειτουργεί η εφαρμογή ανα κατηγορία χρήστη και ανά περίπτωση χρήσης.

Για τον καθηγητή

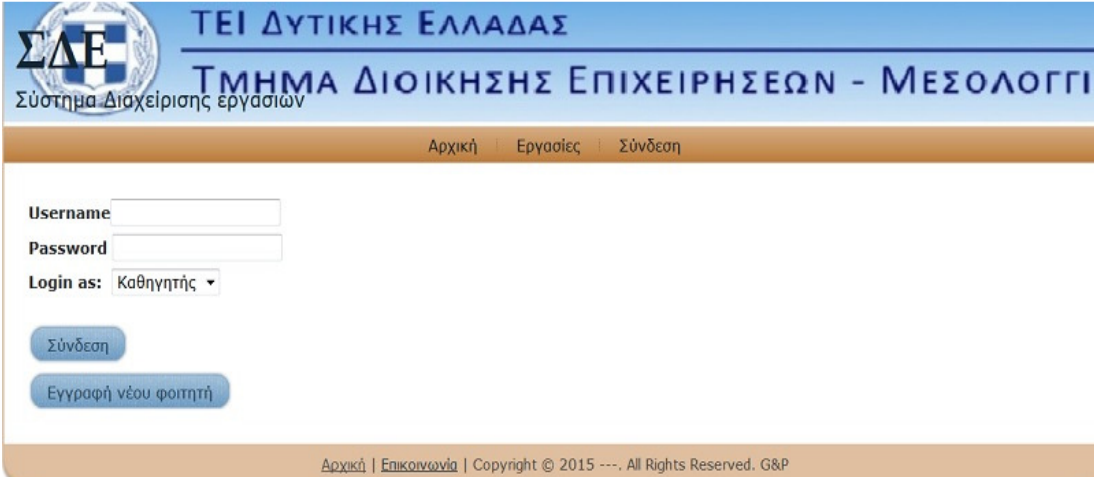
Στην κεντρική σελίδα και στην βασική γραμμή επιλογών υπάρχουν οι επιλογές "Αρχική", "Εργασίες" με την δυνατότητα αναζήτησης της εργασίας ανά καθηγητή και άνα κατηγορία και "Σύνδεση" για την εισαγωγή στο σύστημα.Στο κεντρικό τμήμα της εφαρμογής υπάρχει και η επιλογή της "Εγγραφής" και της "Επικοινωνίας".



Εικόνα κεντρικής σελίδας εφαρμογής

Εισαγωγή

Για να συνδεθεί ο καθηγητής στην εφαρμογή θα πρέπει να εισάγει το όνομα χρήστη και τον κωδικό του, στα αντίστοιχα πεδία username και password, που έχει αποκτήσει από τον διαχειριστή του συστήματος.



ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Σύνδεση

Username

Password

Login as: Καθηγητής ▾

Σύνδεση

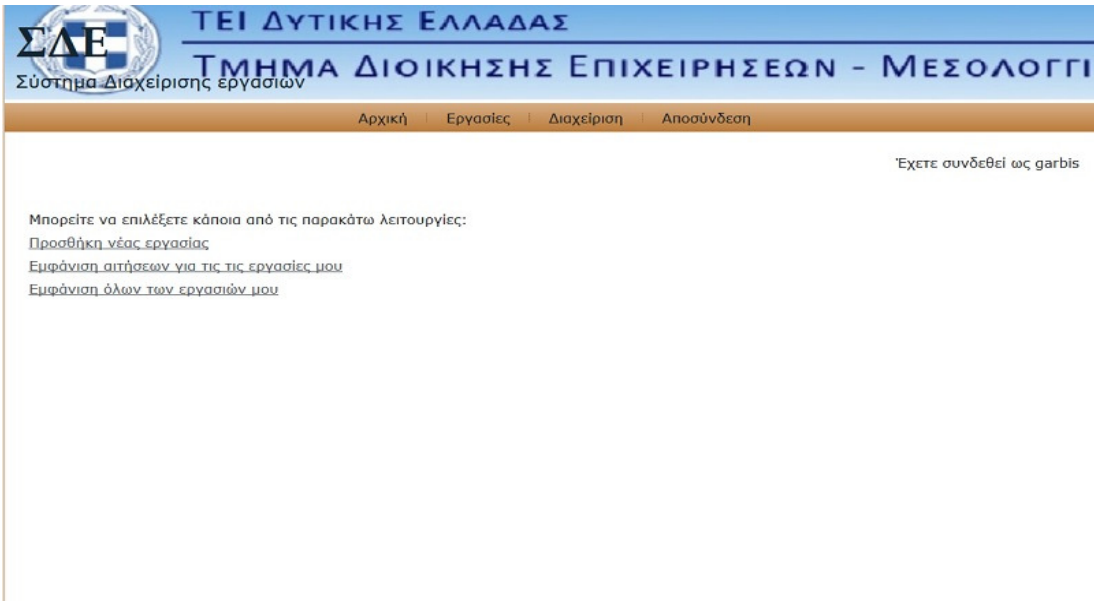
Εγγραφή νέου φοιτητή

Αρχική | Επικοινωνία | Copyright © 2015 ---. All Rights Reserved. G&P

Εικόνα σελίδας log in

Προσθήκη Εργασίας

Στην παρακάτω σελίδα υπάρχουν οι επιλογές "Αρχική" που σε πηγαίνει στην κεντρική σελίδα, "Εργασίες" με την δυνατότητα αναζήτησης της εργασίας ανά καθηγητή και ανά κατηγορία, "Διαχείριση", εδώ είναι οι βασικές λειτουργίες του καθηγητή, οι οποίες βρίσκονται και στο κεντρικό τμήμα της σελίδας και η "Αποσύνδεση".



ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Έχετε συνδεθεί ως garbis

Μπορείτε να επιλέξετε κάποια από τις παρακάτω λειτουργίες:

- [Προσθήκη νέας εργασίας](#)
- [Εμφάνιση αιτήσεων για τις τις εργασίες μου](#)
- [Εμφάνιση όλων των εργασιών μου](#)

Εικόνα κεντρικής σελίδας διαχείρισης καθηγητή

Πιο συγκεκριμένα, με αυτήν την επιλογή ο καθηγητής προσθέτει την εργασία, δίνοντας τον τίτλο και μια σύντομη περιγραφή στα αντίστοιχα πεδία. Επίσης έχει την δυνατότητα επιλογής της κατηγορίας ανάλογα με το αντικείμενο της εργασίας καθώς και το ανέβασμα ενός αρχείου με το τι ζητάει ο καθηγητής για την επίτευξη της πατώντας το κουμπί "Αναζήτηση" επιλέγοντας το αντίστοιχο αρχείο που είναι αποθηκευμένο στο σκληρό δίσκο του προσωπικού του υπολογιστή. Τέλος για να ολοκληρωθεί η διαδικασία επιλέγει το κουμπί της "Αποθήκευσης".

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Προσθήκη Εργασίας

Τίτλος:

Περιγραφή:

Κατηγορία: ΠΛΗΡΟΦΟΡΙΚΗ

Αρχείο: Αναζήτηση... Δεν επιλέχθηκε αρχείο.

Αποθήκευση

Εικόνα κεντρικής σελίδας προσθήκης εργασίας

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Προσθήκη Εργασίας

Τίτλος: ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ

Περιγραφή: Τι είναι τα δίκτυα Η/Υ, σκοπός, αρχιτεκτονική, τοπολογίες κ.α.

Κατηγορία: ΠΛΗΡΟΦΟΡΙΚΗ

Αρχείο: Αναζήτηση... Δίκτυα.docx

Αποθήκευση

Εικόνα προσθήκης εργασίας

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Η εργασία προστέθηκε Έχετε συνδεθεί ως garbis

Αναζήτηση εργασιών με βάση κάποιον καθηγητή

Καθηγητές: ΓΑΡΜΠΗΣ Μόνο διαθέσιμες Αναζήτηση

Διαθέσιμες εργασίες	Τίτλος	Περιγραφή	Κατηγορία	Κατάσταση	Αρχείο
	office	office	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή
	ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ	Τι είναι τα Δίκτυα Η/Υ, σκοπός, αρχιτεκτονική, τοπολογίες κ.α.	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή

Εικόνα αποθήκευσης εργασίας

Λήψη Αρχείου

Σε αυτήν την σελίδα μπορούμε να δούμε τις εργασίες ανα καθηγητή και ποιές είναι μόνο διαθέσιμες αν επιλέξουμε την αντίστοιχη επιλογή. Υπάρχει επίσης η δυνατότητα λήψης του αρχείου καθώς και η διαγραφή της εργασίας.

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Η εργασία προστέθηκε Έχετε συνδεθεί ως garbis

Αναζήτηση εργασιών με βάση κάποιον καθηγητή

Καθηγητές: ΓΑΡΜΠΗΣ Μόνο διαθέσιμες Αναζήτηση

Διαθέσιμες εργασίες

Τίτλος	Περιγραφή	Κατηγορία	Κατάσταση	Αρχείο
office	office	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή
ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ	Τι είναι τα Δίκτυα Η/Υ, σκοπός, αρχιτεκτονική, τοπολογίες κ.α.	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή

Ανοιγμα 55522a93c2c93.docx

Επιλέξατε να ανοίξετε:

55522a93c2c93.docx
που είναι: Microsoft Office Word Document (9,8 KB)
από: http://localhost

Τι να κάνει ο Firefox με αυτό το αρχείο;

Ανοιγμα με Microsoft Office Word (προεπιλογή)

Αποθήκευση αρχείου

Να γίνεται αυτόματα από εδώ και πέρα για αρχεία αυτού του είδους.

OK Ακύρωση

Εικόνα λήψης αρχείου

Διαγραφή Αρχείου

The screenshot shows the website interface for the Department of Administration and Business Operations at the TUI West Greece. A modal dialog box is displayed in the center, asking for confirmation to delete a file titled "ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ". The dialog has "OK" and "Ακύρωση" (Cancel) buttons. In the background, a search results table is visible with columns for Title, Description, Category, Status, and Archive. The table lists two entries, both with the title "ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ" and status "Μη διαθέσιμη".

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

ΣΔΕ
Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Έχετε συνδεθεί ως garbis

Θέλετε σίγουρα να διαγράψετε την εργασία με τίτλο "ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ";

OK Ακύρωση

Αναζήτηση εργασιών με

Καθηγητές: ΓΑΡΜΠΗΣ Μόνο διαθέσιμες

Τίτλος	Περιγραφή	Κατηγορία	Κατάσταση	Αρχειο
office	office	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή
ΔΙΚΤΥΑ ΥΠΟΛΟΓΙΣΤΩΝ	Τι είναι τα Δίκτυα Η/Υ, σκοπός, αρχιτεκτονική, τοπολογίες κ.α.	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή

Εικόνα ερώτησης διαγραφής αρχείου

The screenshot shows the same website interface after the deletion process. A green message box at the top states "Η εργασία διαγράφηκε" (The task has been deleted). The search results table is now empty. The "Αναζήτηση" (Search) button is highlighted in blue.

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

ΣΔΕ
Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Διαχείριση | Αποσύνδεση

Έχετε συνδεθεί ως garbis

Η εργασία διαγράφηκε

Αναζήτηση εργασιών με βάση κάποιον καθηγητή

Καθηγητές: ΓΑΡΜΠΗΣ Μόνο διαθέσιμες [Αναζήτηση](#)

Τίτλος	Περιγραφή	Κατηγορία	Κατάσταση	Αρχειο
office	office	ΠΛΗΡΟΦΟΡΙΚΗ	Μη διαθέσιμη	Λήψη Διαγραφή

Εικόνα διαγραφής αρχείου

Διαχείριση αιτήσεων

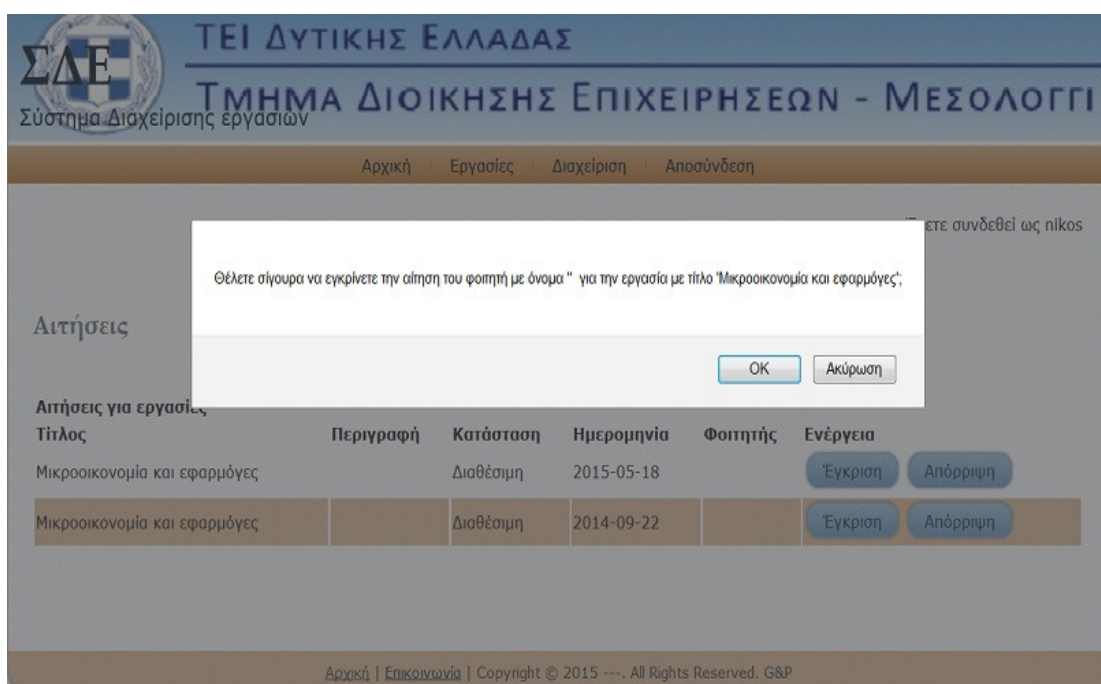
Στην επιλογή "Εμφάνιση αιτήσεων για τις εργασίες μου", εμφανίζονται όλες οι αιτήσεις που έχουν κάνει οι φοιτητές. Ο καθηγητής μπορεί να την εγκρίνει επιλέγοντας το κουμπί "Έγκριση" και στην συνέχεια το κουμπί "Οκ" ή να την απορρίψει επιλέγοντας το κουμπί "Απόρριψη" και εν συνέχεια το κουμπί "Οκ" όπως φαίνεται στις παρακάτω εικόνες.



The screenshot shows the user interface of the 'ΣΔΕ' (Σύστημα Διαχείρισης εργασιών) system. The header includes the logo of the T.E.I. of Western Greece and the Department of Business Administration - Messolonghi. The main content area displays a table of requests for work.

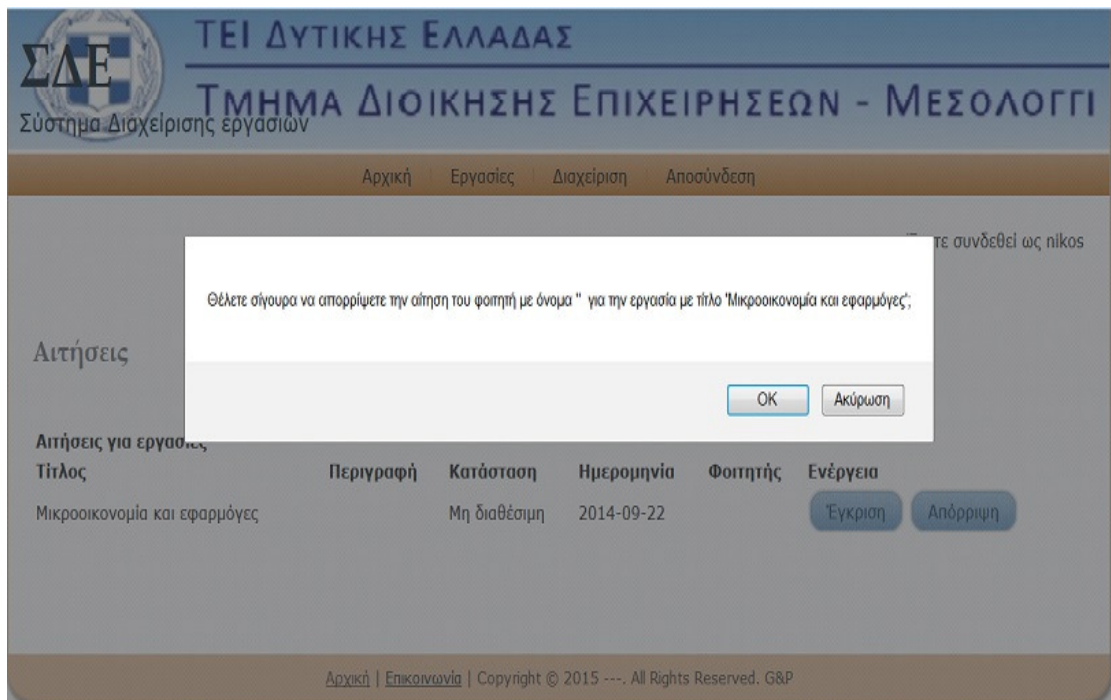
Αιτήσεις για εργασίες	Περιγραφή	Κατάσταση	Ημερομηνία	Φοιτητής	Ενέργεια
Μικροοικονομία και εφαρμογές		Διαθέσιμη	2015-05-18		<input type="button" value="Έγκριση"/> <input type="button" value="Απόρριψη"/>
Μικροοικονομία και εφαρμογές		Διαθέσιμη	2014-09-22		<input type="button" value="Έγκριση"/> <input type="button" value="Απόρριψη"/>

Εικόνα εμφάνισης αιτήσεων



The screenshot shows the same web application interface as above, but with a confirmation dialog box overlaid. The dialog box contains the text: "Θέλετε σίγουρα να εγκρίνετε την αίτηση του φοιτητή με όνομα " για την εργασία με τίτλο 'Μικροοικονομία και εφαρμογές';". Below the text are two buttons: "OK" and "Ακύρωση".

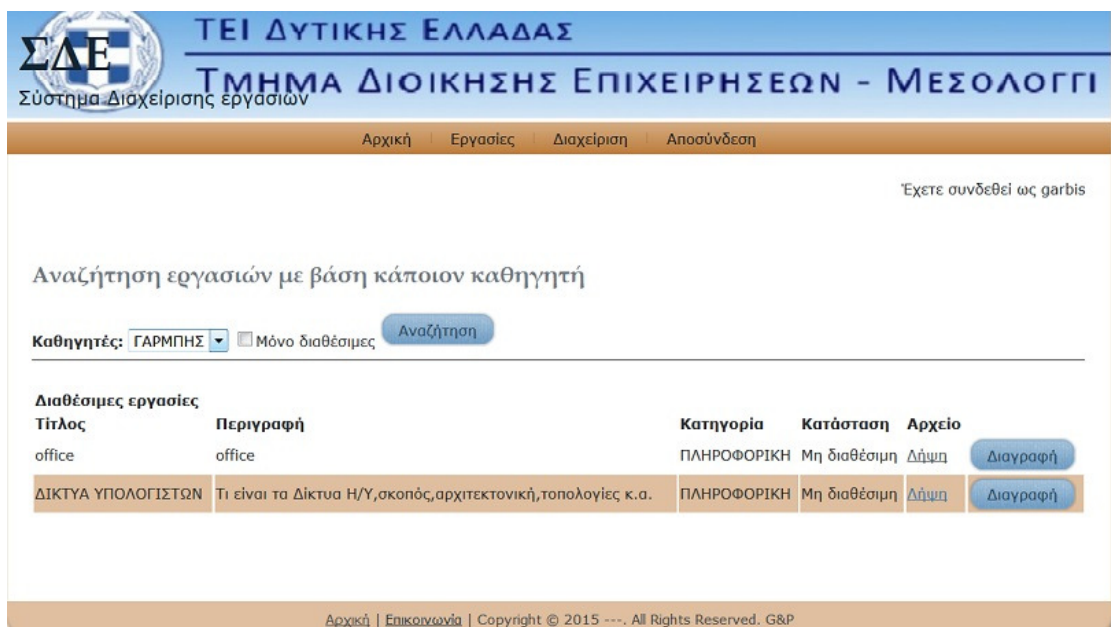
Εικόνα ερώτησης έγκρισης εργασίας



Εικόνα ερώτησης απόρριψης εργασίας

Εμφάνιση όλων των εργασιών

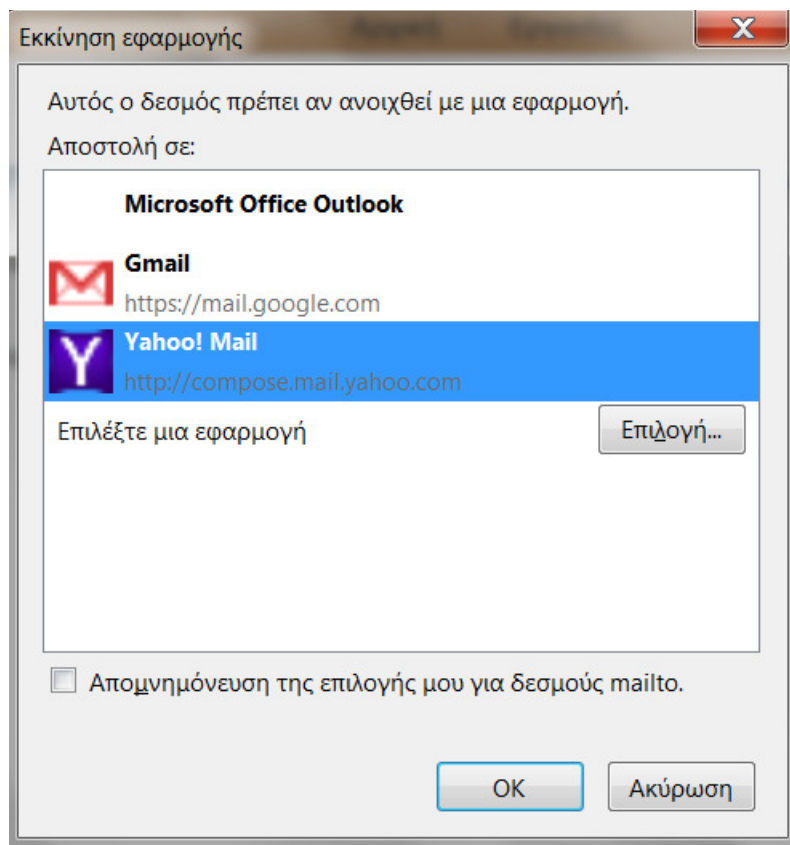
Για να δει ο καθηγητής όλες τις εργασίες του επιλέγει τον σύνδεσμο "Εμφάνιση όλων των εργασιών μου"



Εικόνα εμφάνισης εργασιών

Επικοινωνία

Για να επικοινωνήσει ο καθηγητής με τον διαχειριστή του συστήματος επιλέγει τον σύνδεσμο της "Επικοινωνίας" και ανοίγει το ακόλουθο παράθυρο.



Εικόνα επικοινωνίας

Έξοδος

Για να βγει από το σύστημα ο καθηγητής επιλέγει το κουμπί "Αποσύνδεση".

Για τον φοιτητή

Εγγραφή

Στην περίπτωση του φοιτητή πρέπει να γίνει "εγγραφή νέου φοιτητή" για να εισαχθεί στο σύστημα. Τα πεδία που χρειάζονται για την εγγραφή είναι ένα όνομα, το username, το password και ένα Email πατώντας, τέλος, το κουμπί της Αποστολής.

Εικόνα εγγραφής νέου φοιτητή

Εικόνα ολοκλήρωσης δημιουργίας λογαριασμού

Επιλογή εργασίας ανά καθηγητή

Απο το μενού "Εργασίες" ο φοιτητής μπορεί να επιλέξει την εργασία του είτε ανα κατηγορία είτε ανα καθηγητή βλέποντας και ποιες εργασίες είναι μόνο διαθέσιμες κλικάροντας την αντίστοιχη επιλογή.

ΣΔΕ
Σύστημα Διαχείρισης εργασιών

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Αρχική | Εργασίες | Αποσύνδεση

Έχετε συνδεθεί ως mika

Αναζήτηση εργασιών με βάση κάποιον καθηγητή

Καθηγητές: Επιλογή.. Μόνο διαθέσιμες Αναζήτηση

Επιλογή..
Φαλιόγκα
ΓΑΡΜΠΗΣ
ΝΙΚΟΣ

Αρχική | Επικοινωνία | Copyright © 2015 ---. All Rights Reserved. G&P

Εικόνα αναζήτησης εργασίας ανα καθηγητή

ΣΔΕ
Σύστημα Διαχείρισης εργασιών

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ

Αρχική | Εργασίες | Αποσύνδεση

Έχετε συνδεθεί ως mika

Αναζήτηση εργασιών με βάση κάποιον καθηγητή

Καθηγητές: ΓΑΡΜΠΗΣ Μόνο διαθέσιμες Αναζήτηση

Διαθέσιμες εργασίες		Κατηγορία	Κατάσταση	Αρχείο	
Τίτλος	Περιγραφή				
office	office	ΠΛΗΡΟΦΟΡΙΚΗ	Διαθέσιμη	Δήψη	Επιλογή

Αρχική | Επικοινωνία | Copyright © 2015 ---. All Rights Reserved. G&P

Εικόνα εμφάνισης διαθέσιμων εργασιών με βάση τον καθηγητή

Επιλογή Εργασίας ανά κατηγορία

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ
Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Αποσύνδεση

Έχετε συνδεθεί ως mika

Αναζήτηση εργασιών με βάση την κατηγορία

Κατηγορία: Επιλογή.. Μόνο διαθέσιμες Αναζήτηση

- Επιλογή..
- ΠΛΗΡΟΦΟΡΙΚΗ
- ΔΙΟΙΚΗΣΗ
- ΟΙΚΟΝΟΜΙΑ

Αρχική | Επικοινωνία | Copyright © 2015 ---. All Rights Reserved. G&P

Εικόνα αναζήτησης ανα κατηγορία

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ
Σύστημα Διαχείρισης εργασιών

Αρχική | Εργασίες | Αποσύνδεση

Έχετε συνδεθεί ως mika

Αναζήτηση εργασιών με βάση την κατηγορία

Κατηγορία: ΠΛΗΡΟΦΟΡΙΚΗ Μόνο διαθέσιμες Αναζήτηση

Διαθέσιμες εργασίες				
Τίτλος	Περιγραφή	Καθηγητής	Κατάσταση	Αρχειο
office	office	ΓΑΡΜΠΗΣ	Διαθέσιμη	Δήψη

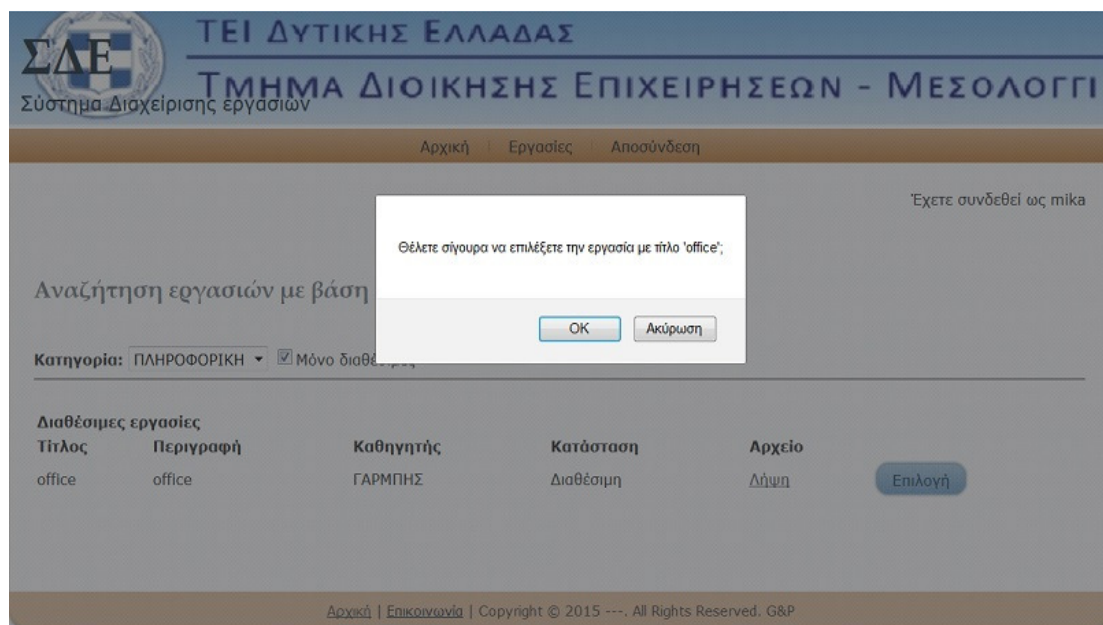
Επιλογή

Αρχική | Επικοινωνία | Copyright © 2015 ---. All Rights Reserved. G&P

Εικόνα εμφάνισης διαθέσιμων εργασιών με βάση την κατηγορία

Εκδήλωση ενδιαφέροντος εργασίας

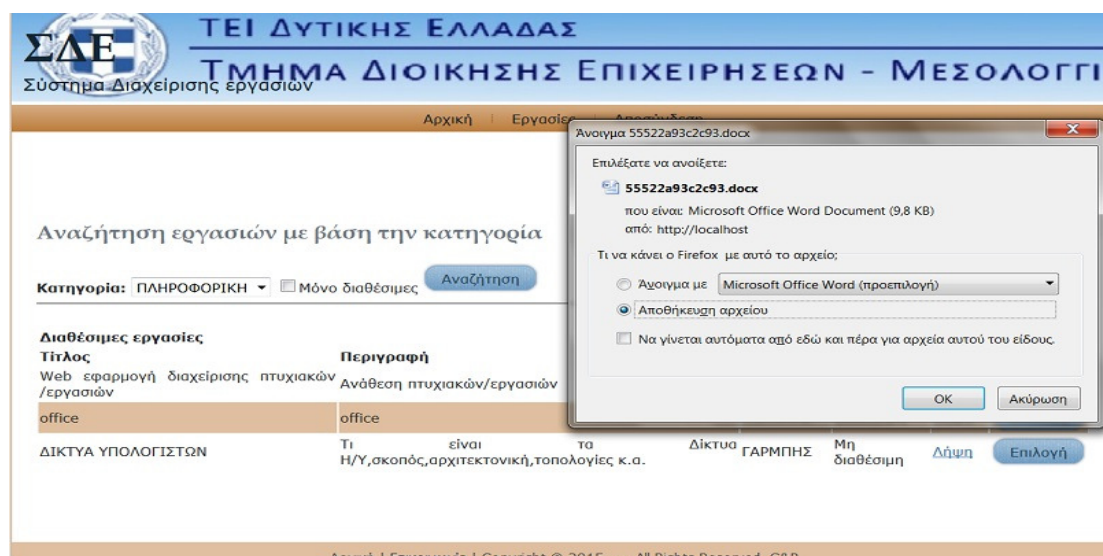
Για την αίτηση ενδιαφέροντος της εργασίας ο φοιτητής κλικάρει το κουμπί της "Επιλογής" και εν συνεχεία "Οκ".



Εικόνα ερώτησης επιλογής εργασίας

Λήψη αρχείου

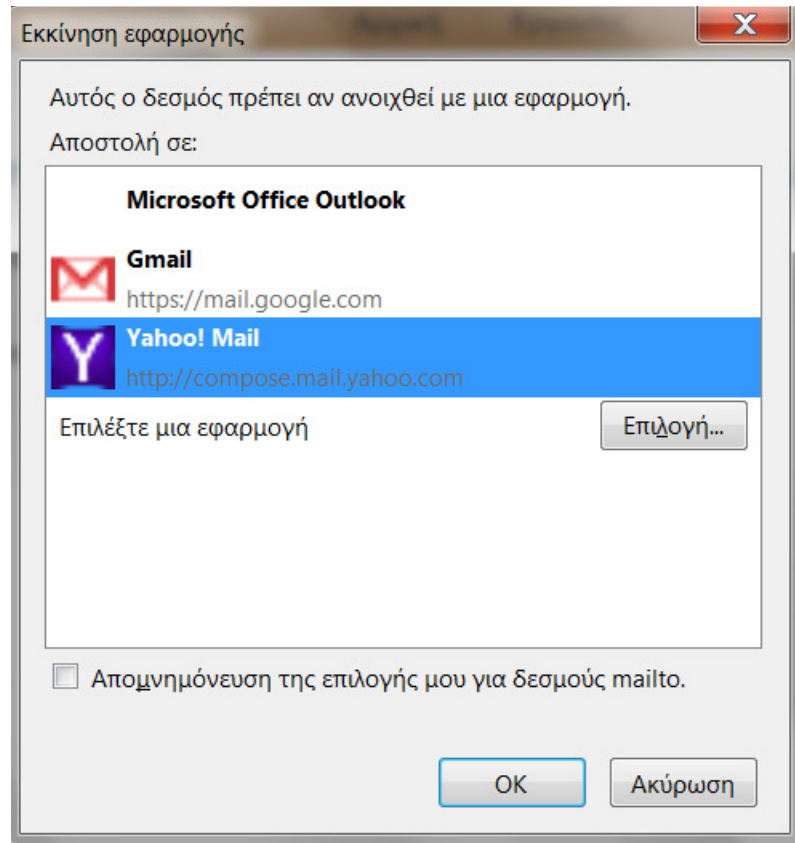
Επίσης υπάρχει και η επιλογή λήψης του αρχείου της εργασίας κλικάροντας τον σύνδεσμο "Λήψη" και στην συνέχεια το "Οκ".



Εικόνα επιλογής λήψης αρχείου

Επικοινωνία

Για να επικοινωνήσει και ο φοιτητής με τον διαχειριστή επιλέγει τον σύνδεσμο "Επικοινωνία" και ανοίγει το αντίστοιχο παράθυρο.



Έξοδος

Τέλος για να βγει απο το σύστημα ο φοιτητής επιλέγει το κουμπί της αποσύνδεσης.

7. ΒΙΒΛΙΟΓΡΑΦΙΑ - ΙΣΤΟΓΡΑΦΙΑ

1. HTML 5: <http://www.w3schools.com/html5/default.asp>
2. PHP: <http://php.net/manual>
3. CSS: <http://www.w3schools.com/css/>
4. JavaScript: <http://www.w3schools.com/js/>
5. SQL: <http://www.w3schools.com/sql/default.asp>
6. MySQL: <http://dev.mysql.com/>
7. JQuery Datetime Picker: <http://trentrichardson.com/examples/timepicker/>
8. <http://www.w3schools.com> - <http://www.freestuff.gr/forums/viewtopic.php?t=26378> -
9. <http://el.wikipedia.org/wiki/CSS> - <http://www.wlearn.gr/index.php/css-84>
10. <http://xoops.gr/modules/publisher/item.php?itemid=12> - <http://localhost/phpmyadmin/>
11. <http://www.dw-world.de/dw/article/0,,3488159,00.html>
12. Luke Welling, Laura Thomson, Τρίτη Έκδοση, 2009, Ανάπτυξη Εφαρμογών με Php και MySql, Εκδόσεις Μ. Γκιούρδας,
13. David J.Eck, Πέμπτη Έκδοση 2006, Introduction to Programming using Java, Creative Commons
14. Rebecca Murphey, 2012, jQuery Fundamentals, [Creative Commons Attribution-Share Alike 3.0 United States license](http://creativecommons.org/licenses/by-sa/3.0/)
15. Reenskaug Trygve, 1995, Working with objects, Hardbound
16. Κωνσταντίνος Μάρκελλος, Πηνελόπη Μαρκέλλου, Μαρία Ρήγκου, Σπύρος Συρμακέσης, Αθανάσιος Τσακαλίδης, 2006, e-Επιχειρηματικότητα, από την ιδέα στην υλοποίηση, Ελληνικά γράμματα

ΠΑΡΑΡΤΗΜΑ

Παρακάτω παρουσιάζουμε τον κώδικα SQL της βάσης δεδομένων:

```
-- phpMyAdmin SQL Dump
-- version 3.5.2.2
-- http://www.phpmyadmin.net
--
-- Φιλοξενητής: 127.0.0.1
-- Χρόνος δημιουργίας: 02 Μαρ 2013 στις 11:39:52
-- Έκδοση διακομιστή: 5.5.27
-- Έκδοση PHP: 5.4.7
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";

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

--
-- Βάση: `essays`
--
-----

-- Δομή πίνακα για τον πίνακα `category`
--
CREATE TABLE IF NOT EXISTS `category` (
  `ID` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci AUTO_INCREMENT=3 ;

-----

--
-- Δομή πίνακα για τον πίνακα `essay`
--
```

```

CREATE TABLE IF NOT EXISTS `essay` (
  `ID` int(11) NOT NULL AUTO_INCREMENT,
  `description` text COLLATE utf8_unicode_ci NOT NULL,
  `categoryID` int(11) NOT NULL,
  `title` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `status` tinyint(4) NOT NULL,
  `username` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `filename` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`ID`),
  KEY `FK_categories` (`categoryID`),
  KEY `FK_username` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci AUTO_INCREMENT=7 ;

```

--

-- Δομή πίνακα για τον πίνακα `request`

--

```

CREATE TABLE IF NOT EXISTS `request` (
  `essayID` int(11) NOT NULL,
  `student` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `date` date NOT NULL,
  `status` int(11) NOT NULL DEFAULT '0',
  UNIQUE KEY `essayID` (`essayID`,`student`),
  KEY `student` (`student`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

```

--

-- Δομή πίνακα για τον πίνακα `student`

--


```

CREATE TABLE IF NOT EXISTS `student` (
  `username` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `password` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `email` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `essayID` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

-----

--
-- Δομή πίνακα για τον πίνακα `teacher`
--

CREATE TABLE IF NOT EXISTS `teacher` (
  `username` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `password` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  `email` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

--
-- Περιορισμοί για άχρηστους πίνακες
--

-- Περιορισμοί για πίνακα `essay`
--

ALTER TABLE `essay`

  ADD CONSTRAINT `essay_ibfk_1` FOREIGN KEY (`categoryID`) REFERENCES `category`
  (`ID`),

  ADD CONSTRAINT `essay_ibfk_2` FOREIGN KEY (`username`) REFERENCES `teacher`
  (`username`);

--

-- Περιορισμοί για πίνακα `request`
--

ALTER TABLE `request`

```

```

ADD CONSTRAINT `request_ibfk_1` FOREIGN KEY (`essayID`) REFERENCES `essay` (`ID`),
ADD CONSTRAINT `request_ibfk_2` FOREIGN KEY (`student`) REFERENCES `student`
(`username`);

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

```

Οι βασικές αρχές που έχουν τηρηθεί κατά την ανάπτυξη της εφαρμογής είναι οι εξής:

- Έγινε προσπάθεια διαχωρισμού των χαρακτηριστικών εμφάνισης του site από τη λειτουργικότητά του. Για το λόγο αυτό, έχει χρησιμοποιηθεί αρχείο css, που περιγράφει το πώς θα εμφανίζονται τα διάφορα στοιχεία της HTML. Επίσης, όλες οι κλήσεις στη βάση δεδομένων έχουν υλοποιηθεί με συναρτήσεις, ώστε να χωριστεί σε ένα βαθμό η λειτουργικότητα του ιστότοπου από την επικοινωνία με τη βάση.
- Τα επαναχρησιμοποιούμενα στοιχεία του ιστότοπου (header, footer) έχουν οριστεί σε ξεχωριστά αρχεία, ώστε να μην επαναλαμβάνεται συνεχώς ο ίδιος κώδικας και να είναι ευκολότερο να γίνουν αλλαγές.
- Οι βασικές οντότητες που περιγράφηκαν και στη βάση δεδομένων, μοντελοποιούνται με χρήση αντικειμενοστραφούς προγραμματισμού. Η χρήση αντικειμένων για την ανάκτηση και αποθήκευση δεδομένων κάνει το κώδικα ευκολότερο στην ανάγνωση και διόρθωση.

Με βάση τα παραπάνω, ο ιστότοπος έχει την παρακάτω δομή:

- files: κατάλογος με τα βοηθητικά αρχεία των εργασιών. Τα αρχεία ανεβαίνουν στον κατάλογο αυτό έπειτα από συμπλήρωση της σχετικής φόρμας από κάποιον καθηγητή
- images: κατάλογος με τις εικόνες της εφαρμογής
- Includes: κατάλογος με βοηθητικά αρχεία
 - Includes/config.php : Αρχείο ρυθμίσεων και ορισμού των βασικών κλάσεων
 - Includes/functions.php : Αρχείο ορισμού όλων των συναρτήσεων του ιστότοπου
- style.css: Το αρχείο css με το template της εφαρμογής
- script.js: αρχείο javascript με βοηθητικές μεθόδους για το template της εφαρμογής
- Footer.php: footer του δημόσιου ιστότοπου
- Header.php: header του δημόσιου ιστότοπου
- Index.php: Αρχική σελίδα
- login.php: Σελίδα για τη σύνδεση στον ιστότοπο
- register.php: Σελίδα για τη δημιουργία νέου λογαριασμού φοιτητή
- essays.php: Σελίδα που εμφανίζει τη λίστα των εργασιών (ανά καθηγητή ή ανά κατηγορία)
- addressay.php: Σελίδα για την ανάρτηση νέας εργασίας
- student.php: Η "κεντρική" σελίδα για τους φοιτητές. Μπορούν να δουν την κατάσταση της αίτησής τους
- teacher.php: Η "κεντρική" σελίδα για τους καθηγητές

- requests.php: Χρησιμοποιείται από τους καθηγητές για την εμφάνιση των αιτήσεων που εκκρεμούν.

Στη συνέχεια θα γίνει αναλυτική περιγραφή των παραπάνω αρχείων:

Επεξήγηση config.php

Στο αρχείο αυτό βρίσκονται οι ορισμοί των κλάσεων που χρησιμοποιούνται στην εφαρμογή και μερικές βοηθητικές συναρτήσεις.

Γίνεται η σύνδεση με τη βάση και αρχικοποιείται το PHP SESSION .

```
<?php
ob_start();
session_start();

define('DBHOST','localhost');
define('DBUSER','root');
define('DBPASS','root');
define('DBNAME','essays');

$conn = @mysql_connect (DBHOST, DBUSER, DBPASS);
$conn = @mysql_select_db (DBNAME);
if (!$conn){
    die( "Sorry! There seems to be a problem connecting to our database.");
}
mysql_query("SET NAMES UTF8");
```

Προκειμένου να γίνει η ανάπτυξη του κώδικα ευκολότερα, προτιμήθηκε η χρήση κλάσεων που περιγράφουν τις βασικές οντότητες του συστήματος. Οι κλάσεις αυτές αντιστοιχούν στους πίνακες της βάσης που ορίστηκαν σε προηγούμενη ενότητα. Πρόκειται για τις κλάσεις Teacher, Student, Essay, Category, που αντιστοιχούν στις οντότητες Καθηγητής, Φοιτητής, Εργασία, Κατηγορία. Ο ορισμός τους δίνεται παρακάτω:

```
class Teacher
{
    public $username="N/A";

    public $password="N/A";

    public $name="N/A";

    public $email="N/A";
}

class Student
{
    public $username="N/A";

    public $password="N/A";

    public $name="N/A";

    public $email="N/A";
}

class Essay
{
    public $ID="N/A";
```

```

        public $description="N/A";
        public $categoryID="N/A";
        public $category="N/A";
    public $title="N/A";
        public $status="N/A";
        public $filename="N/A";
        public $teacher="N/A";
        public $teacherUsername="N/A";
    }
class Category
{
    public $ID="N/A";
    public $name="N/A";
}

```

Επεξήγηση functions.php

Το αρχείο αυτό περιέχει τις συναρτήσεις σύνδεσης και επεξεργασίας των πινάκων της βάσης δεδομένων. Καλείται (με Include) από το αρχείο config.php

Στη συνέχεια, παρουσιάζουμε ενδεικτικά διάφορα είδη συναρτήσεων που υπάρχουν στο αρχείο.

Συνάρτηση ταυτοποίησης των στοιχείων του χρήστη:

```

function login($user, $pass, $type){
    $user = strip_tags(mysql_real_escape_string($user));
    $pass = strip_tags(mysql_real_escape_string($pass));
    $type = strip_tags(mysql_real_escape_string($type));

    $pass = md5($pass);

    $sql = "SELECT * FROM student WHERE username = '$user' AND password = '$pass'";

    if($type=='teacher')
        $sql = "SELECT * FROM teacher WHERE username = '$user' AND password
= '$pass'";

    $result = mysql_query($sql) or die('Query failed. ' . mysql_error());

    if (mysql_num_rows($result) == 1) {

        $_SESSION['authorized'] = $type;
        $_SESSION['username'] = $user;

        header('Location: index.php');
        exit();
    } else {

        $_SESSION['error'] = 'Τα στοιχεία που δώσατε δεν είναι έγκυρα';
    }
}

```

```
}  
}
```

Για λόγους ασφαλείας, γίνεται χρήση των συναρτήσεων `mysql_real_escape_string` και `strip_tags` σε όλες τις μεταβλητές που προκύπτουν από εισαγωγή στοιχείων από το χρήστη (φόρμες, `http get request` κλπ). Οι συναρτήσεις αυτές αφαιρούν «επικίνδυνους» χαρακτήρες από τη μεταβλητή, όπως εισαγωγικά, ερωτηματικά και λοιπά. Με τον τρόπο αυτό παρέχεται προστασία σε επιθέσεις που ονομάζονται SQL injection. Οι επιθέσεις αυτές εκμεταλλεύονται τη σύνταξη της SQL για να εισφέρουν εντολές που μπορούν να διαβάσουν ή να τροποποιήσουν μια βάση δεδομένων, ή να αλλάξουν την έννοια του αρχικού query.

Για παράδειγμα, στην περίπτωση μας η `login.php` έχει δύο πεδία στα οποία επιτρέπεται οι χρήστες να εισάγουν ένα όνομα χρήστη και έναν κωδικό πρόσβασης. Ο κώδικας πίσω από τη σελίδα θα δημιουργήσει ένα ερώτημα SQL για να ελέγξει τον κωδικό πρόσβασης βάσει του καταλόγου των ονομάτων των χρηστών

```
SELECT * FROM members WHERE username = '$user' AND password = '$pass'
```

Αν αυτό το ερώτημα επιστρέφει αποτελέσματα, τότε παρέχεται πρόσβαση. Ωστόσο, εάν ο κακόβουλος χρήστης εισάγει ένα έγκυρο όνομα χρήστη και δώσει κάποιο έγκυρο κωδικό (`'password' OR '1'='1'`), στο πεδίο του κωδικού πρόσβασης, τότε το ερώτημα που προκύπτει, θα είναι κάπως έτσι:

```
SELECT *  
FROM members  
WHERE members.Username = 'Username'  
AND members.Password = 'password' OR '1'='1'
```

Στο παραπάνω παράδειγμα, αν επιτραπεί η μεταβλητή `$pass` να πάρει την τιμή string `'1'='1'`, τότε το query θα επιστρέφει πάντα αποτελέσματα, επιτρέποντας έτσι την πρόσβαση.

Συνάρτηση η οποία τυπώνει στη σελίδα μηνύματα, μέσα από το Session:

```
function messages() {  
    $message = '';  
    if(isset($_SESSION['success']) && $_SESSION['success'] != '') {  
        $message = '<div class="msg-ok">'. $_SESSION['success']. '</div>';  
        $_SESSION['success'] = '';  
    }  
}
```

```

    if( isset($_SESSION['error']) && $_SESSION['success'] != '' ) {
        $message = '<div class="msg-error">'.$_SESSION['error'].'</div>';
        $_SESSION['error'] = '';
    }
    echo "$message";
}

```

Η συνάρτηση αυτή καλείται στην αρχή κάθε σελίδας για να τυπώσει μηνύματα που παράχθηκαν από την προηγούμενη ενέργεια του χρήστη. Επειδή τα μηνύματα παράγονται σε προηγούμενη κλήση στο site, επιλέχθηκε το SESSION προκειμένου να περάσουν από τη μια σελίδα στην άλλη.

Συνάρτηση η οποία επιστρέφει λίστα από αντικείμενα τύπου Category:

```

function getCategories()
{
    $category_list=array();
    $sql="SELECT * FROM `category` ";

    $result = mysql_query($sql);
    $i=0;
    while($row = mysql_fetch_array($result))
    {
        $category=new Category();
        $category->ID=$row['ID'];
        $category->name=$row['name'];

        $category_list[$i]=$category;
        $i++;
    }

    return $category_list;
}

```

Συνάρτηση η οποία επιστρέφει τη λίστα των εργασιών ανά κατηγορία. Υπό περίπτωση, επιστρέφει μονάχα τις διαθέσιμες (status=0)

```

function getEssaysByCategoryID($categoryID,$available=false)
{
    $categoryID = mysql_real_escape_string($categoryID);
    $essay_list=array();
    $sql="SELECT essay.*,teacher.name FROM essay,teacher where
essay.username=teacher.username AND categoryID = $categoryID ";
    if($available)
        $sql = $sql . ' and status=0 ';

    $result = mysql_query($sql);
    $i=0;
    while($row = mysql_fetch_array($result))
    {
        $essay=new Essay();
        $essay->ID=$row['ID'];
        $essay->description=$row['description'];
    }
}

```

```

        $essay->title=$row['title'];
        $essay->filename=$row['filename'];
        $essay->teacher=$row['name'];
        $essay->teacherUsername=$row['username'];
        $essay->status=$row['status'];

        $essay_list[$i]=$essay;
        $i++;
    }

    return $essay_list;
}

```

Συνάρτηση η οποία επιστρέφει τη λίστα των εργασιών ανά καθηγητή. Υπό περίπτωση, επιστρέφει μονάχα τις διαθέσιμες (status=0)

```

function getEssaysByTeacher($username,$available)
{
    $username = mysql_real_escape_string($username);
    $available = mysql_real_escape_string($available);

    $essay_list=array();
    $sql="SELECT essay.*,category.name FROM essay,category where
essay.categoryID=category.ID AND username = '$username' ";
    if($available)
        $sql=$sql . ' and status=0 ';

    $result = mysql_query($sql);
    $i=0;
    while($row = mysql_fetch_array($result))
    {
        $essay=new Essay();
        $essay->ID=$row['ID'];
        $essay->description=$row['description'];
        $essay->title=$row['title'];
        $essay->filename=$row['filename'];
        $essay->teacher=$row['name'];
        $essay->teacherUsername=$row['username'];
        $essay->status=$row['status'];
        $essay->categoryID=$row['categoryID'];
        $essay->category=$row['name'];

        $essay_list[$i]=$essay;
        $i++;
    }

    return $essay_list;
}

```

Συνάρτηση η οποία προσθέτει στη βάση δεδομένων μια εγγραφή τύπου Essay:

```

function addEssay($title,$description , $categoryID,$username,$filename) {

    $title = mysql_real_escape_string($title);
    $description = mysql_real_escape_string($description);
    $categoryID = mysql_real_escape_string($categoryID);
    $username = mysql_real_escape_string($username);
}

```

```

        $filename = mysql_real_escape_string($filename);

        $sql="INSERT INTO essay
(title,description,categoryID,username,filename,status) VALUES
('$title','$description','$categoryID','$username','$filename',0)";
        $result = mysql_query($sql) or die(mysql_error());

        if($result)
            return mysql_insert_id();
        else
            return 0;
    }

```

Συνάρτηση η οποία διαγράφει από τη βάση δεδομένων μια εγγραφή τύπου Essay:

```

function deleteEssay($id,$username){
    $id = mysql_real_escape_string($id);
    $username = mysql_real_escape_string($username);

    $sql="delete from essay where ID=$id AND username='$username'";
    $result = mysql_query($sql) or die(mysql_error());
    return mysql_affected_rows();
}

```

Συνάρτηση η οποία χρησιμοποιείται για την απόρριψη αίτησης (status=1):

```

function denyRequest($id,$username){
    $id = mysql_real_escape_string($id);
    $username = mysql_real_escape_string($username);

    $sql="update request set status=1 where essayID=$id AND
student='$username'";
    $result = mysql_query($sql) or die(mysql_error());
}

```

Συνάρτηση η οποία χρησιμοποιείται για την έγκριση αίτησης (request.status=2, student.essayID=ID της εργασίας, essay.status=1):

```

function acceptRequest($id,$username)
{
    $id = mysql_real_escape_string($id);
    $username = mysql_real_escape_string($username);

    //close request
    $sql="update request set status=2 where essayID=$id AND
student='$username'";
    $result = mysql_query($sql) or die(mysql_error());

    //update essay status
    $sql="update essay set status=1 where ID=$id ";
    $result = mysql_query($sql) or die(mysql_error());
}

```



```

//update student status
$sql="update student set essayID=$id where username='$username'";
$result = mysql_query($sql) or die(mysql_error());
}

```

Επεξήγηση header.php

Το αρχείο αυτό χρησιμοποιείται για το κοινό κομμάτι κάθε ιστοσελίδας (header) στην διαχειριστική εφαρμογή. Αρχικά δηλώνεται ο HTML Header της σελίδας, με δηλώσεις για τα αρχεία css και js που χρησιμοποιούνται στον ιστότοπο.

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="en-US"
xml:lang="en">
<head>
  <!--
    Created by Artisteer v2.3.0.21098
    Base template (without user's data) checked by http://validator.w3.org :
    "This page is valid XHTML 1.0 Transitional"
  -->
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=EmulateIE7" />
  <title>ΣΔΕ </title>

  <script type="text/javascript" src="script.js"></script>

  <link rel="stylesheet" href="style.css" type="text/css" media="screen"
/>
  <!--[if IE 6]><link rel="stylesheet" href="style.ie6.css"
type="text/css" media="screen" /><![endif]-->
  <!--[if IE 7]><link rel="stylesheet" href="style.ie7.css"
type="text/css" media="screen" /><![endif]-->

```

Κατά τη διαγραφή μιας εργασίας, ο ιστότοπος πρέπει να ζητά επιβεβαίωση. Συνεπώς, όταν γίνεται ένα HTTP GET Request που αφορά τη διαγραφή μιας εγγραφής, καλείται η αντίστοιχη συνάρτηση που ζητά επιβεβαίωση από το χρήστη. Η επιβεβαίωση γίνεται μέσω της συνάρτησης confirm της JavaScript:

```

<script language="JavaScript" type="text/javascript">
function deleteEssay(id,title,username){
  if (confirm("Θέλετε σίγουρα να διαγράψετε την εργασία με τίτλο '" + title
+ "';")) {
    window.location.href = 'essays.php?teacher=' + username + '&delid=' +
id;
  }
}

```

```
}
```

Αντίστοιχος κώδικας υπάρχει για κάθε αλλαγή κατάστασης από χρήστη (επιλογή εργασίας, έγκριση/απόρριψη αίτησης).

```
function selectEssay(id,title, username){
    if(username=="")
        alert("Για να επιλέξετε εργασία πρέπει πρώτα να συνδεθείτε στο
σύστημα");
    else if (confirm("Θέλετε σίγουρα να επιλέξετε την εργασία με τίτλο '" +
title + "';"))
    {
        window.location.href = 'student.php?essay=' + id;
    }
}

function acceptRequest(id,title,username,name){
    if (confirm("Θέλετε σίγουρα να εγκρίνετε την αίτηση του φοιτητή με
όνομα '" + name +"' για την εργασία με τίτλο '" + title + "';") )
    {
        window.location.href = 'requests.php?username=' +
username + '&essayid=' + id + '&requestaccept=1' ;
    }
}

function denyRequest(id,title,username,name){
    if (confirm("Θέλετε σίγουρα να απορρίψετε την αίτηση του φοιτητή
με όνομα '" + name +"' για την εργασία με τίτλο '" + title + "';") )
    {
        window.location.href = 'requests.php?username=' +
username + '&essayid=' + id + '&requestaccept=0' ;
    }
}
```

Στη συνέχεια ξεκινά το HTML BODY και ορίζεται το κεντρικό μενού που είναι κοινό για όλες τις σελίδες.

```
</script>
</head>
<body>
<div id="art-page-background-simple-gradient">
</div>
<div id="art-main">
<div class="art-Sheet">
<div class="art-Sheet-tl"></div>
<div class="art-Sheet-tr"></div>
<div class="art-Sheet-bl"></div>
<div class="art-Sheet-br"></div>
<div class="art-Sheet-tc"></div>
<div class="art-Sheet-bc"></div>
<div class="art-Sheet-cl"></div>
<div class="art-Sheet-cr"></div>
<div class="art-Sheet-cc"></div>
```

```

<div class="art-Sheet-body">
  <div class="art-Header">
    <div class="art-Header-jpeg"></div>
    <div class="art-Logo">
      <h1 id="name-text" class="art-Logo-name"><a
href="#">ΣΔΕ</a></h1>
      <div id="slogan-text" class="art-Logo-text">Σύστημα
Διαχείρισης εργασιών</div>
    </div>
  </div>

```

Στην συνέχεια ορίζεται το μενού. Το μενού δεν είναι πάντα το ίδιο. Διαφέρει αν ο χρήστης είναι ανώνυμος, φοιτητής ή καθηγητής.

Τα κοινά στοιχεία για όλους είναι τα εξής:

- Αρχική (index.php)
- Εργασίες
 - Ανά κατηγορία (essays.php)
 - Ανά καθηγητή (essays.php?teacher)

```

<div class="art-nav">
  <div class="l"></div>
  <div class="r"></div>
  <div class="art-nav-center">
    <ul class="art-menu">
      <li>
        <a href="index.php"
class="{ActiveItem}"><span class="l"></span><span class="r"></span><span
class="t">Αρχική</span></a>
      </li>
      <li>
        <a href="#"><span class="l"></span><span
class="r"></span><span class="t">Εργασίες</span></a>
      <ul>
        <li><a
href="essays.php?teacher">Ανά Καθηγητή</a></li>
        <li><a href="essays.php">Ανά
Κατηγορία</a></li>
      </ul>
      </li>
    </ul>
  </div>

```

Αν είναι καθηγητής, εμφανίζεται το εξής υπομενού:

- Διαχείριση
 - Προσθήκη νέας εργασίας (addressay.php)
 - Νέες αιτήσεις (requests.php)
 - Οι εργασίες μου (essays.php?teacher='username')

```
<?php
```

```

        if(logged_in()=='teacher'){
$username = $_SESSION['username'];
                                ?>
                                <li>
                                <a href="#"><span class="l"></span><span
class="r"></span><span class="t">Διαχείριση</span></a>
                                <ul>
                                <li><a
href="addressay.php">Προσθήκη νέας εργασίας</a></li>
                                <li><a
href="requests.php">Νέες αιτήσεις</a></li>
                                <li><a
href="essays.php?teacher=?php echo $username; ?>">Οι εργασίες μου</a></li>
                                </ul>
                                </li>
                                <?php
                                }

```

Αν ο χρήστης είναι ανώνυμος, εμφανίζεται σύνδεσμος για τη σύνδεση στο σύστημα: (login.php) ενώ αν έχει ήδη συνδεθεί (εμφανίζεται σύνδεσμος για την αποσύνδεση (index.php?logout)).

```

                                if(logged_in()==''){
                                ?>
                                <li>
                                <a href="login.php"><span
class="l"></span><span class="r"></span><span class="t">Σύνδεση</span></a>
                                </li>
                                <?php
                                } else {
                                ?>
                                <li>
                                <a href="index.php?logout"><span
class="l"></span><span class="r"></span><span
class="t">Αποσύνδεση</span></a>
                                </li>
                                <?php
                                }
                                ?>

                                </ul>
                                </div>
                                </div>
                                <div class="art-contentLayout">
                                <div class="art-content">
                                <div class="art-Post">
                                <div class="art-Post-body">
                                <div
class="art-Post-inner">
                                <div class="art-PostContent">

```

Επεξήγηση footer.php


```

{
include('includes/header.php');
    messages();
?>

```

Καλείται η συνάρτηση `messages()` για την εμφάνιση μηνυμάτων προς τον χρήστη. Γίνεται έλεγχος αν ο χρήστης έχει συνδεθεί στο σύστημα. Αν έχει συνδεθεί, τότε γίνεται ανακατεύθυνση είτε στην `teacher.php` είτε στην `student.php`, ανάλογα με το τι επιστρέφει η συνάρτηση `logged_in()`. Αν δεν έχει συνδεθεί, εμφανίζονται πληροφορίες και σύνδεσμοι για τη σύνδεση ή τη δημιουργία λογαριασμού.

```

    <h2>Καλωσήρθατε στο σύστημα διαχείρισης εργασιών</h2>

    <p> Για να επιλέξετε μια εργασία θα πρέπει πρώτα να έχετε <a
href="login.php" >συνδεθεί</a> στο σύστημα. </p>

    <br/>

    <p> Αν δεν έχετε λογαριασμό μπορείτε να <a href="register.php" >
εγγραφείτε</a> στο σύστημα. </p>

    <br/> <br/> <br/> <br/>

    <br/> <br/> <br/> <br/> <br/>

    <br/> <br/> <br/> <br/> <br/>

<?php
    include('includes/footer.php');
}
else if($usertype=='teacher'){
    header('Location: teacher.php');
    exit();
}
else if($usertype=='student'){
    header('Location: student.php');
    exit();
}
?>

```

Επεξήγηση *essays.php*

Η σχεδίαση της σελίδας ακολουθεί την εξής λογική:

- Εμφανίζεται μια φόρμα για την εμφάνιση των εργασιών είτε ανά κατηγορία είτε ανά καθηγητή. Η διακλάδωση ανάμεσα στις δυο περιπτώσεις γίνεται με τη χρήση της http GET παραμέτρου `?teacher`
- Κάνοντας Post στη φόρμα ο χρήστης, εμφανίζονται οι αντίστοιχες εργασίες
- Υπάρχει δυνατότητα εμφάνισης μονάχα των διαθέσιμων εργασιών (είτε ανά κατηγορία είτε ανά φοιτητή).

Η σελίδα κάνει include `header.php` και `footer.php` στην αρχή και στο τέλος αντίστοιχα. Αν ο χρήστης έχει συνδεθεί, κρατείται το `username` του. Αν η κλήση HTTP GET της σελίδας έχει ως παράμετρο `'delid'`, τότε γίνεται διαγραφή της εργασίας με `ID= $_GET['delid'];`

```
<?php
require('./includes/config.php');
require('./includes/functions.php');
if(logged_in()!='')
    $username = $_SESSION['username'];
else
    $username = '';
include('includes/header.php');
if(isset($_GET['delid'])) {
    $id=$_GET['delid'];
    $affected_rows = deleteEssay($id,$username);
    if($affected_rows==1){
        $_SESSION['success'] = 'Η εργασία διαγράφηκε';
    }
    else
        $_SESSION['error'] = 'Η εργασία δεν ήταν δυνατό να
διαγραφεί';
}
```

Καλείται η συνάρτηση `messages()` για την εμφάνιση μηνυμάτων προς τον χρήστη.

```
messages();
```

Γίνεται έλεγχος αν ο χρήστης έχει επιλέξει το checkbox ώστε να εμφανίζονται μόνο οι διαθέσιμες εργασίες

```
if(isset($_GET['available']) || isset($_POST['available']))
    $available=true;
else
    $available=false;
```

Στη συνέχεια γίνεται έλεγχος αν θα εμφανιστούν οι εργασίες ανά καθηγητή. Επιστρέφεται η λίστα καθηγητών από τη βάση.

```
if(isset($_GET['teacher']) || isset($_POST['teacher']))
{
    $teachers = getTeachers();
    $teacherUsername="";
    $essays=array();
}
```

Η παράμετρος teacher πρέπει να έχει το Username του καθηγητή του οποίου οι εργασίες θα εμφανιστούν

```
if(isset($_POST['teacher']) && $_POST['teacher']!=' ){
    $teacherUsername=$_POST['teacher'];
}
else if(isset($_GET['teacher']) && $_GET['teacher']!=' ){
    $teacherUsername=$_GET['teacher'];
}
```


Καλείται η συνάρτηση `getEssaysByTeacher` για να επιστραφούν από τη βάση ένα array με αντικείμενα τύπου `Essay` που αντιστοιχούν στον καθηγητή με `username= teacherUsername`. Αν `$available==true`, τότε επιστρέφονται μόνο οι διαθέσιμες.

```
$essays=getEssaysByTeacher($teacherUsername,$available);
```

Ακολουθεί η φόρμα στην οποία ο χρήστης μπορεί να επιλέξει καθηγητή:

```
<form action="" method="post">
    <p><strong>Καθηγητές: </strong>

        <select name="teacher">
            <option value="">Επιλογή..</option>
        <?php
            foreach ( $teachers as $teacher){

                if($teacherUsername == $teacher->username)
                    echo "<option selected='selected'
value='$teacher->username'>$teacher->name</option>";
                else
                    echo "<option value='$teacher-
>username'>$teacher->name</option>";
            }
        ?>
        </select>

        <input type="checkbox" name="available" value="1" <?php
if($available) echo "checked" ?> >Μόνο διαθέσιμες

        <input type="submit" name="submit" value="Αναζήτηση" class="art-
button" /></p>
    </form>

    <hr/>
```

Και στη συνέχεια, διατρέχεται το array `$essays` και εμφανίζεται ο πίνακας με τις εργασίες:

```

if(count($essays)>0) {
    ?>
    <br/>
    <table class="table" width="100%">
        <tr>
            <th colspan='4' ><strong>Διαθέσιμες
εργασίες </strong></th>
        </tr>
        <tr>
            <th><strong>Τίτλος</strong></th>
            <th><strong>Περιγραφή</strong></th>
            <th><strong>Κατηγορία</strong></th>
            <th><strong>Κατάσταση</strong></th>
            <th><strong>Αρχείο</strong></th>
            <th></th>
        </tr>
        <?php
        $i=0;
        foreach ( $essays as $essay){
            if($i%2==0)
                echo "<tr>";
            else
                echo "<tr bgcolor='#DFBF9F'>";
            echo " <td>$essay->title</td>
                <td>$essay->description</td>
                <td>$essay->category</td>";
            if($essay->status==0)
                echo "<td>Διαθέσιμη</td>";
            else
                echo "<td>Μη διαθέσιμη</td>";
            echo "<td><a target='_blank'
href='files/$essay->filename'>Λήψη</a></td>";

```

Εφόσον ο χρήστης δεν είναι ανώνυμος, γίνεται έλεγχος αν είναι καθηγητής ή φοιτητής:

- Αν είναι καθηγητής και η εργασία είναι δική του, μπορεί να την διαγράψει

```

                                if (logged_in()=="teacher"  && $essay->
>teacherUsername == $username )

                                echo "<td><input type=\"button\"
value=\"Διαγραφή\" class=\"art-button\"
onclick=\"javascript:deleteEssay('$essay->ID','$essay->title', '$username');\" > </td></tr>";

                                else

```

- αν είναι φοιτητής, τότε μπορεί να την επιλέξει.

```

                                echo "<td><input type=\"button\"
value=\"Επιλογή\" class=\"art-button\"
onclick=\"javascript:selectEssay('$essay->ID','$essay->title', '$username');\" > </td></tr>";

                                $i++;

                                }

                                ?>

                                </table>

                                <?php

                                }

}

```

Εδώ ολοκληρώνεται ο κώδικας που τρέχει στην περίπτωση που οι εργασίες θα εμφανιστούν ανά καθηγητή. Διαφορετικά, ελέγχεται αν έχει δοθεί κατηγορία από το χρήστη και επιστρέφεται η αντίστοιχη λίστα εργασιών.

```

else {

    $categories = getCategories();

    $categoryID=0;

    $essays=array();

    if(isset($_POST['category_id']) && $_POST['category_id']>0 ){

        $categoryID=$_POST['category_id'];

        $essays=getEssaysByCategoryID($categoryID,$available);

    }

    if(logged_in()!=''){

        ?>

```

```

        <div align="right">Εχετε συνδεθεί ως <?php echo
$username; ?> </div> <br/><br/>

        <?php
        }
        ?>

```

Ακολουθεί η φόρμα στην οποία ο χρήστης μπορεί να επιλέξει κατηγορία:

```

<h3>Αναζήτηση εργασιών με βάση την κατηγορία</h3>

<form action="" method="post">
<p><strong>Κατηγορία:</strong>

        <select name="category_id">
                <option value="">Επιλογή..</option>

        <?php
        foreach ( $categories as $category){
                if($categoryID == $category->ID)
                        echo "<option selected='selected'
value='$category->ID'>$category->name</option>";
                else
                        echo "<option value='$category-
>ID'>$category->name</option>";
                }
        ?>
        </select>

<input type="checkbox" name="available" value="1" <?php if($available) echo
"checked" ?> >Μόνο διαθέσιμες

        <input type="submit" name="submit" value="Αναζήτηση" class="art-
button" /></p>

</form>

<hr/>

```

Και στη συνέχεια, διατρέχεται το array \$essays και εμφανίζεται ο πίνακας με τις εργασίες:

```

if(count($essays)>0) {
        ?>
        <br/>

```

```

        <table class="table" width="100%">
            <tr>
                <th colspan="4" ><strong>Διαθέσιμες
εργασίες </strong></th>
            </tr>
            <tr>
                <th><strong>Τίτλος</strong></th>
                <th><strong>Περιγραφή</strong></th>
                <th><strong>Καθηγητής</strong></th>
                <th><strong>Κατάσταση</strong></th>
                <th><strong>Αρχείο</strong></th>
                <th></th>
            </tr>
            <tr>
                <td><strong>εργασίες </strong></td>
                <td><strong>Τίτλος</strong></td>
                <td><strong>Περιγραφή</strong></td>
                <td><strong>Καθηγητής</strong></td>
                <td><strong>Κατάσταση</strong></td>
                <td><strong>Αρχείο</strong></td>
                <td></td>
            </tr>
        </table>
        <?php
        $i=0;
        foreach ( $essays as $essay){
            if($i%2==0)
                echo "<tr>";
            else
                echo "<tr bgcolor='#DFBF9F'>";
            echo " <td>$essay->title</td>
                <td>$essay->description</td>
                <td>$essay->teacher</td>";
            if($essay->status==0)
                echo "<td>Διαθέσιμη</td>";
            else
                echo "<td>Μη διαθέσιμη</td>";
            echo "<td><a target='_blank'
href='files/$essay->filename'>Λήψη</a></td>";
        }
    }
}

```

Επεξήγηση addressay.php

Η σελίδα αυτή είναι διαθέσιμη μόνο σε καθηγητές, οπότε γίνεται ο σχετικός έλεγχος αρχικά.

```
<?php
require('./includes/config.php');
require('./includes/functions.php');

if(logged_in()!='teacher'){
    logout();
}

$username = $_SESSION['username'];
```

Στη συνέχεια, γίνεται έλεγχος αν πρόκειται για POST (Η σελίδα κάνει Post στον εαυτό της τα στοιχεία της φόρμας. Πρόκειται για φόρμα τύπου " multipart/form-data" καθώς περιέχεται και αρχείο.

Τα στοιχεία που λαμβάνονται από τη φόρμα είναι :

- title
- description
- categoryID
- Η δομή \$_FILES['userfile'] που περιέχει τις πληροφορίες για το αρχείο:
 - \$fileName = \$_FILES['userfile']['name']; (όνομα αρχείου που ανέβασε ο χρήστης)
 - \$tmpName = \$_FILES['userfile']['tmp_name']; (το fullpath του προσωρινού αρχείου που ανέβηκε στο server)
 - \$fileSize = \$_FILES['userfile']['size']; (το μέγεθος του αρχείου σε bytes)
 - \$fileType = \$_FILES['userfile']['type']; (ο τύπος του αρχείου)

```
if(isset($_POST['submit'])){
    $title = $_POST['title'];
    $description = $_POST['description'];
    $categoryID = $_POST['categoryID'];
    if( $_FILES['userfile']['size'] > 0)
    {
        $fileName = $_FILES['userfile']['name'];
        $tmpName = $_FILES['userfile']['tmp_name'];
```

```
$fileSize = $_FILES['userfile']['size'];  
$fileType = $_FILES['userfile']['type'];
```

Στη συνέχεια, βρίσκουμε το extension του αρχείου, δημιουργούμε ένα τυχαίο filename με το extension του αρχείου, και ορίζεται που θα αποθηκευτεί τελικά το αρχείο (target_path).

```
$extension = end(explode(".", $_FILES['userfile']['name']));  
$filename = uniqid() . '.' . $extension;  
$target_path = "files/" . $filename ;
```

Τελικά, αν πετύχει η μεταφορά του αρχείου από την προσωρινή θέση στο target_path, ενημερώνεται και η βάση με τη νέα εργασία με τη χρήση της συνάρτησης addEssay. Η addEssay επιστρέφει το ID της νέας εργασίας .

```
if(move_uploaded_file($_FILES['userfile']['tmp_name'], $target_path))  
{  
    $id = addEssay($title,$description , $categoryID,$username,$filename);  
    if($id>0){  
        $_SESSION['success'] = 'Η εργασία προστέθηκε';  
        header("Location: essays.php?teacher=$username");  
    }  
}
```

Το \$_SESSION['success'] χρησιμοποιείται τελικά από τη messages για την εμφάνιση μηνύματος για την επιτυχημένη ανάρτηση της εργασίας. Η επεξεργασία των δεδομένων ολοκληρώνεται με την ανακατεύθυνση στη σελίδα essays.php.

Ακολουθεί ο html κώδικας για τη δημιουργία της φόρμας. Η λίστα των κατηγοριών δημιουργείται δυναμικά από τη βάση δεδομένων

```
include("../includes/header.php");  
$categories = getCategories();  
>>  
<div id="content">
```

```

<h1>Προσθήκη Εργασίας</h1>

<form action="" method="post" enctype="multipart/form-data">

<table>

    <tr><td>Τίτλος: </td><td> <input name="title" type="text" size="50"
/></td></tr>

    <tr><td>Περιγραφή:</td><td> <textarea name="description" cols="50"
rows="10"></textarea> </td></tr>

    <tr><td>Κατηγορία: </td><td>

        <select name="categoryID">

            <?php

                foreach ( $categories as $category){

                    echo "<option value='$category-
>ID'$>$category->name</option>";

                }

            ?>

        </select>

    </td></tr>

```

Το αρχείο μπορεί να έχει μέγιστο μέγεθος 2000000 Bytes:

```

    <tr><td>Αρχείο:</td> <td><input name="userfile" type="file"
id="userfile"> <input type="hidden" name="MAX_FILE_SIZE"
value="2000000"></td></tr>

</table>

<input type="submit" name="submit" value="Αποθήκευση" class="button" />

</form>

<br/><br/><br/>

<?php

include('./includes/footer.php');

?>

```

Επεξήγηση *requests.php*

Η σελίδα αυτή είναι διαθέσιμη μονάχα σε καθηγητές, οπότε γίνεται ο σχετικός έλεγχος αρχικά.


```

<?php
require('./includes/config.php');
require('./includes/functions.php');
if(logged_in()=='teacher')
    $username = $_SESSION['username'];
else{
    logout();
}
include('includes/header.php');

```

Στη συνέχεια, γίνεται έλεγχος αν ο καθηγητής έχει ήδη ζητήσει την έγκριση ή απόρριψη κάποιας αίτησης, πατώντας στον σχετικό σύνδεσμο παρακάτω στη σελίδα. Αν ναι, καλείται αντίστοιχα η `acceptRequest` ή η `denyRequest`.

```

if(isset($_GET["username"]) && isset($_GET["essayid"]) &&
isset($_GET["requestaccept"])){
    if($_GET["requestaccept"]==1){
        acceptRequest($_GET["essayid"],$_GET["username"]);
    }
    else if ($_GET["requestaccept"]==0){
        denyRequest($_GET["essayid"],$_GET["username"]);
    }
}

```

Ακολουθεί ο κώδικας που εμφανίζει ένα πίνακα με όλες τις αιτήσεις που εκκρεμούν.

```

<?php
if(count($requests)>0) {
?>
<br/>
<table class="table" width="100%">
    <tr>

```

```
εργασίες </strong></th> <th colspan='4' ><strong>Αιτήσεις για
```

```
</tr>
<tr>
<th><strong>Τίτλος</strong></th>
<th><strong>Περιγραφή</strong></th>
<th><strong>Κατάσταση </strong></th>
<th><strong>Ημερομηνία </strong></th>
<th><strong>Φοιτητής</strong></th>
<th><strong>Ενέργεια</strong></th>
</tr>
<?php
$i=0;
foreach ( $requests as $request){
    $student =
    getStudent($request['student']);
    if($i%2==0)
        echo "<tr>";
    else
        echo "<tr bgcolor='#DFBF9F'>";
    echo " <td>" . $request['title'] . "</td>
        <td>"
        . $request['description'] . "</td>";
    if($request['status']==0)
        echo "<td>Διαθέσιμη</td>";
    else
        echo "<td>Μη διαθέσιμη</td>";
    echo " <td>" . $request['date'] .
"</td>";
    echo " <td><a href='mailto:$student-
>email'>" . $student->name . "</a></td>";
    echo "<td><input type=\"button\"
value=\"Εγγραφή\" class=\"art-button\"
onclick=\"javascript:acceptRequest('" . $request['essayID'] . "','"
.$request['title'] . "','" . $student->username . "','" . $student->name
.'"');\> > ";
```

```

                                echo "<input type=\"button\"
value=\"Απόρριψη\" class=\"art-button\"
onclick=\"javascript:denyRequest('\" . $request['essayID'] . '\",'\"
.$request['title'] . '\",'\" . $student->username . '\",'\" . $student->name
.\");\" > </td></tr>";

                                $i++;

                                }

                                ?>

                                </table>

                                <?php
                                }

                                else echo "<strong>Δεν έχετε εκκρεμείς αιτήσεις</strong>";

                                ?>

                                <br/><br/><br/>

                                <?php

                                include('includes/footer.php');

                                ?>

```

Επεξήγηση student.php

Η σελίδα αυτή είναι διαθέσιμη μόνο σε φοιτητές, οπότε γίνεται ο σχετικός έλεγχος αρχικά.

```

<?php

require('./includes/config.php');

require('./includes/functions.php');

if(logged_in() != 'student'){

    logout();

}

include('includes/header.php');

```

Στη συνέχεια, γίνεται έλεγχος αν ο φοιτητής έχει ήδη αναλάβει κάποια εργασία, ή έχει κάνει αίτηση.

```

$username = $_SESSION['username'];

$myEssay = getUserEssay($username);

$myRequestedEssay = getRequestedEssay($username);

```

Στη σελίδα αυτή οδηγείται ο φοιτητής και αν επιλέξει κάποια εργασία. Συνεπώς πρέπει να γίνει έλεγχος ότι δεν έχει ήδη κάποια εργασία, ή ότι η αίτηση είναι έγκυρη. Οι έλεγχοι αυτοί γίνονται παρακάτω:

```

if(isset($_GET['essay']) && $_GET['essay']>0 ){

    $essayID = $_GET['essay'];

    if($myEssay->ID!="N/A"){ //user already has an essay

        $_SESSION['error'] = 'Έχετε ήδη επιλέξει εργασία';

    }

    else if($myRequestedEssay->ID!="N/A"){ //user already has
an essay

        $_SESSION['error'] = 'Έχετε ήδη ζητήσει την
εκπόνηση εργασίας';

    }

    else {

        if(!requestEssay($username,$essayID))

            $_SESSION['error'] = 'Υπήρξε πρόβλημα κατά
την επεξεργασία της αίτησης';

        else

            $myRequestedEssay =
getRequestedEssay($username);

    }

}

```

Αν έχει ήδη αναλάβει μια εργασία, τυπώνονται σε πίνακα τα στοιχεία της εργασίας:

```

<div align="right">Έχετε συνδεθεί ως <?php echo $_SESSION['username'];
?> </div> <br/>

<?php

if($myEssay->ID!="N/A"){?>

<p>Η αίτησή σας για την εκπόνηση της παρακάτω εργασίας έγινε
δεκτή.</p>

<table width="100%">

```

```

<tr>
    <th colspan='4' ><strong>Στοιχεία εργασίας </strong></th>
</tr>
<tr>
    <th><strong>Τίτλος</strong></th>
    <th><strong>Περιγραφή</strong></th>
    <th><strong>Καθηγητής</strong></th>
    <th><strong>Αρχείο</strong></th>
    <th></th>
</tr>
<tr bgcolor="DFBF9F">
    <td><?php echo $myEssay->title  ?></td>
    <td><?php echo $myEssay->description  ?></td>
    <td><?php echo $myEssay->teacher  ?></td>
    <td><a target='_blank' href='files/<?php echo $myEssay->filename?>'>Λήψη</a></td>
</tr>
</table>

```

Διαφορετικά, αν εκκρεμεί κάποια αίτηση του, τυπώνονται σε πίνακα τα στοιχεία της εργασίας:

```

<?php
    } else if ($myRequestedEssay->ID!="N/A") {
    ?>
    <p>Η αίτησή σας για την εκπόνηση της παρακάτω εργασίας εκκρεμεί.</p>
    <table width="100%">
    <tr>
        <th colspan='4' ><strong>Στοιχεία εργασίας </strong></th>
    </tr>
    <tr>
        <th><strong>Τίτλος</strong></th>
        <th><strong>Περιγραφή</strong></th>
        <th><strong>Καθηγητής</strong></th>
        <th><strong>Αρχείο</strong></th>

```

```

                <th></th>
            </tr>
            <tr bgcolor="DFBF9F">
                <td><?php echo $myRequestedEssay->title ?></td>
                <td><?php echo $myRequestedEssay-
>description ?></td>
                <td><?php echo $myRequestedEssay->teacher
?></td>
                <td><a target='_blank' href='files/<?php
echo $myEssay->filename?>'>Λήψη</a></td>
            </tr>
        </table>

```

Διαφορετικά , τυπώνεται μήνυμα ότι έχει κάνει αίτηση για κάποια εργασία ακόμα και ολοκληρώνεται η σελίδα με το Footer.

```

<?php
    } else
        echo "<p>Δεν έχετε κάνει αίτηση για κάποια εργασία
ακόμα</p>"
    ?>
    <br/><br/><br/><br/><br/><br/><br/><br/>
<?php
include('includes/footer.php');
?>

```

Επεξήγηση *teacher.php*

Η σελίδα αυτή είναι διαθέσιμη μόνο σε καθηγητές, οπότε γίνεται ο σχετικός έλεγχος αρχικά.

```

<?php
require('./includes/config.php');
require('./includes/functions.php');
if(logged_in()!='teacher'){
    logout();
}

```

```
}  
  
$username = $_SESSION['username'];  
  
include('includes/header.php');  
  
messages();  
  
?>
```

Στη συνέχεια, εμφανίζονται σχετικοί σύνδεσμοι:

```
<div align="right">Εχετε συνδεθεί ως <?php echo $username; ?>  
</div> <br/> <br/>  
  
<p>Μπορείτε να επιλέξετε κάποια από τις παρακάτω λειτουργίες:</p>  
  
<p><a href="addressay.php">Προσθήκη νέας εργασίας</a></p>  
  
<p><a href="requests.php?teacher=<?php echo $username; ?>">Εμφάνιση  
αιτήσεων για τις τις εργασίες μου</a></p>  
  
<p><a href="essays.php?teacher=<?php echo $username; ?>">Εμφάνιση  
όλων των εργασιών μου</a></p>  
  
<br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/><br/>  
</p>  
  
<?php  
  
include('includes/footer.php');  
  
?>
```