# Εκπαιδευτικά Περιβάλλοντα Διαδικτύου

**Μέρος Β Ευάγγελος Δήμος 4038**

***ΟΔΗΓΙΕΣ ΠΡΟΣΒΑΣΗΣ***

Tutors:

* Login= dimvlax@gmail.com pwd=12345v
* Login= billchatzi@gmail.com pwd=12345c

Students:

* Login= nikmulo@csd.auth.gr pwd=12345m
* Login= gianlado@csd.auth.gr pwd=12345l

***ΛΙΝΚ***

[http](http://username.webpages.auth.gr/)s[://evangdimos.webpages.auth.gr](http://username.webpages.auth.gr/)/4038partB/

***ΚΩΔΙΚΑΣ***

Login (index.php)

Αυτό το αρχείο κώδικα αναπτύσσει μια σελίδα πιστοποίησης για έναν χρήστη. Πρώτα ελέγχει εάν ο χρήστης είναι ήδη συνδεδεμένος, και αν ναι, τον ανακατευθύνει στην αρχική σελίδα. Στη συνέχεια, εάν έχει υποβληθεί η φόρμα σύνδεσης, λαμβάνει τα δεδομένα σύνδεσης από τη φόρμα, συνδέεται στη βάση δεδομένων και εκτελεί ένα ερώτημα για τον έλεγχο του χρήστη. Αν τα δεδομένα σύνδεσης είναι σωστά, αποθηκεύει τα στοιχεία συνεδρίας και ανακατευθύνει τον χρήστη στην αρχική σελίδα. Αν υπάρχει σφάλμα κατά την εκτέλεση του ερωτήματος SQL ή κάποιο σφάλμα κατά τη σύνδεση στη βάση δεδομένων, εμφανίζει το σχετικό μήνυμα λάθους. Τέλος, η σελίδα περιέχει ένα HTML φόρμα για την είσοδο του χρήστη, παρουσιάζοντας τυχόν μηνύματα σφάλματος που προκύπτουν κατά την προσπάθεια σύνδεσης.

Logout (logout.php)

Αυτό το αρχείο υλοποιεί μια διαδικασία αποσύνδεσης χρήστη. Κατά την εκτέλεση, πρώτα καθαρίζει όλες τις μεταβλητές της συνεδρίας μέσω της συνάρτησης unset($\_SESSION), και στη συνέχεια καταστρέφει τη συνεδρία μέσω της συνάρτησης session\_destroy(). Έπειτα, ανακατευθύνει τον χρήστη στη σελίδα σύνδεσης μέσω της εντολής header('Location: index.php'). Τέλος, χρησιμοποιεί την εντολή exit για να εξέλθει από το σενάριο. Το κουμπί Logout που βρίσκεται στις 5 βασικές σελίδες του ιστοχώρου συνδέεται με αυτό το αρχείο για την πραγματοποίηση της αποσύνδεσης.

Homepage (homepage.php)

Αυτό το αρχείο κώδικα περιέχει μια σελίδα με την κεφαλίδα ενός ιστότοπου. Καταρχάς, ελέγχει εάν ο χρήστης δεν έχει συνδεθεί, και αν όχι, τον ανακατευθύνει στη σελίδα σύνδεσης. Στη συνέχεια, η σελίδα περιλαμβάνει τίτλο, μεταδεδομένα για την κωδικοποίηση και την προβολή σε φορητές συσκευές, καθώς και έναν σύνδεσμο προς ένα αρχείο CSS για τη μορφοποίηση της σελίδας. Ακολουθεί η κεφαλίδα της σελίδας που περιλαμβάνει έναν τίτλο και το μενού πλοήγησης (sidebar menu) με διάφορες επιλογές. Στο μενού περιλαμβάνεται και ένα κουμπί για διαχείριση χρηστών, το οποίο είναι διαθέσιμο μόνο για τους χρήστες με ρόλο "Tutor". Τέλος, η σελίδα περιέχει το περιεχόμενο της αρχικής σελίδας, με κείμενο καλωσορίσματος και περιγραφής του ιστότοπου, καθώς και μια εικόνα που προσθέτει ενδιαφέρον.

Announcements (announcements.php)

Το παραπάνω αρχείο περιέχει τη σελίδα ανακοινώσεων. Ξεκινάει με έναν έλεγχο για τη σύνδεση του χρήστη και την ανακατεύθυνση στη σελίδα σύνδεσης εάν ο χρήστης δεν έχει συνδεθεί. Στη συνέχεια, περιέχει τίτλο, μεταδεδομένα και σύνδεσμο με το αρχείο CSS για τη μορφοποίηση. Ακολουθεί η κεφαλίδα της σελίδας με το μενού πλοήγησης και το κουμπί αποσύνδεσης. Στη συνέχεια, παρουσιάζονται οι ανακοινώσεις, με δυνατότητα προσθήκης, διαγραφής και επεξεργασίας για τους χρήστες με ρόλο "Tutor". Τέλος, υπάρχει ένας σύνδεσμος που επιστρέφει στην κορυφή της σελίδας.

addAnnouncement.php και editAnnouncement.php

Το πρώτο αρχείο αναφέρεται στην προσθήκη ανακοίνωσης. Ξεκινά με τον έλεγχο εάν η φόρμα υποβλήθηκε και εάν η μέθοδος αίτησης είναι POST. Στη συνέχεια, λαμβάνονται τα δεδομένα από τη φόρμα και πραγματοποιείται σύνδεση στη βάση δεδομένων. Έπειτα, ελέγχεται εάν όλα τα πεδία έχουν συμπληρωθεί και εισάγονται τα δεδομένα στη βάση. Αν η εισαγωγή ολοκληρωθεί επιτυχώς, γίνεται ανακατεύθυνση στη σελίδα announcements.php. Στο τέλος, παρέχεται ένα μήνυμα σφάλματος, εάν υπάρξει κάποιο πρόβλημα κατά την εισαγωγή.

Αντίστοιχα, το παρακάτω αρχείο αναφέρεται στην επεξεργασία ανακοίνωσης. Ξεκινάει με έναν έλεγχο για τη μέθοδο αίτησης (GET ή POST). Αν η μέθοδος είναι GET, γίνεται έλεγχος για την ύπαρξη της παραμέτρου id στο URL και ανάκτηση των στοιχείων της ανακοίνωσης. Στην περίπτωση της POST, εκτελείται ερώτημα SQL για την ενημέρωση των στοιχείων της ανακοίνωσης. Στο τέλος, παρέχεται μήνυμα σφάλματος και υποστήριξης.

Communication (communication.php)

Αυτό το αρχείο κώδικα υλοποιεί μια διαδικτυακή φόρμα επικοινωνίας μέσω ηλεκτρονικού ταχυδρομείου. Αρχικά, ξεκινά με την εκκίνηση της συνεδρίας (session) και την εισαγωγή απαραίτητων βιβλιοθηκών PHPMailer (Τοποθέτησα σε ένα φάκελο με όνομα PHPMailer όλα τα αρχεία. Η δημιουργία του φακέλου έγινε από ένα άλλο project στο github όπως αναγράφεται και στα σχόλια του κώδικα). Στη συνέχεια, ελέγχει εάν ο χρήστης είναι συνδεδεμένος και αν δεν είναι, τον ανακατευθύνει στη σελίδα σύνδεσης. Στη συνέχεια, λαμβάνει τα δεδομένα που υποβλήθηκαν μέσω της φόρμας, ελέγχει την ορθότητά τους και εάν είναι εντάξει, αποθηκεύει τα μηνύματα στη βάση δεδομένων και αποστέλλει email σε όλους τους διδάσκοντες με τα συγκεκριμένα μηνύματα. Τέλος, εμφανίζει τη φόρμα επικοινωνίας στη σελίδα HTML, συμπεριλαμβανομένων δύο επιλογών για αποστολή email, είτε μέσω της web φόρμας είτε με χρήση ηλεκτρονικής διεύθυνσης.

Documents (documents.php)

Αυτό το αρχείο κώδικα υλοποιεί μια σελίδα "Έγγραφα Μαθήματος". Αρχικά, περιλαμβάνει την εκκίνηση μιας συνεδρίας (session) και ελέγχει εάν ο χρήστης έχει συνδεθεί. Σε περίπτωση που δεν έχει συνδεθεί, ανακατευθύνει τον χρήστη στη σελίδα σύνδεσης. Έπειτα, καθορίζει τον πίνακα της συνεδρίας ως "documents" και συνδέεται με τη βάση δεδομένων. Στη συνέχεια, παρέχει ένα περιεχόμενο HTML που περιλαμβάνει μια περιοχή εμφάνισης εγγράφων. Αν ο χρήστης είναι καθηγητής, εμφανίζει ένα κουμπί για προσθήκη νέου εγγράφου. Έπειτα, εκτελεί ένα SQL ερώτημα για ανάγνωση εγγράφων από τη βάση δεδομένων και τα εμφανίζει με τίτλο, περιγραφή και σύνδεσμο για λήψη. Τέλος , περιλαμβάνει ένα κουμπί επιστροφής στην κορυφή της σελίδας.

addDocument.php και editDocument.php

Το πρώτο αρχείο προσθέτει ένα νέο έγγραφο στη βάση δεδομένων. Αρχικά, ελέγχει αν η φόρμα υποβλήθηκε και η μέθοδος αιτήματος είναι POST. Στη συνέχεια, αποθηκεύει τα δεδομένα του εγγράφου από τη φόρμα και εκτελεί ένα SQL ερώτημα για να το εισάγει στη βάση δεδομένων. Αν η εισαγωγή πραγματοποιηθεί επιτυχώς, ανακατευθύνει τον χρήστη στη σελίδα εγγράφων.

Το άλλο αρχείο επεξεργάζεται ένα υπάρχον έγγραφο. Αρχικά, ελέγχει αν η φόρμα υποβλήθηκε με μέθοδο GET και αν υπάρχει η παράμετρος id στο URL. Στη συνέχεια, επιλέγει το έγγραφο από τη βάση δεδομένων με βάση το id. Αν η φόρμα υποβληθεί με μέθοδο POST, αποθηκεύει τα νέα δεδομένα του εγγράφου και εκτελεί ένα SQL ερώτημα για να ενημερώσει τα δεδομένα στη βάση δεδομένων. Αν η ενημέρωση πραγματοποιηθεί επιτυχώς, ανακατευθύνει τον χρήστη στη σελίδα εγγράφων. Τα δύο αρχεία χρησιμοποιούν τη συνάρτηση `uploadFile` από το αρχείο `functions.php` για το ανέβασμα του αρχείου στον server. Επίσης, χρησιμοποιούν τη συνάρτηση `connectToDb` για τη σύνδεση στη βάση δεδομένων. Τέλος, εμφανίζουν το μήνυμα σφάλματος εάν κάποιο πεδίο είναι κενό.

Homework (homework.php)

Το αρχείο παρουσιάζει μια σελίδα που επιτρέπει στους χρήστες να προβάλλουν εργασίες και στους διαχειριστές να προσθέτουν, να επεξεργάζονται και να διαγράφουν εργασίες. Αρχικά, ελέγχει αν ο χρήστης είναι συνδεδεμένος και, αν όχι, τον ανακατευθύνει στη σελίδα σύνδεσης. Στη συνέχεια, φορτώνει τα δεδομένα των εργασιών από τη βάση δεδομένων και τα εμφανίζει. Η σελίδα περιλαμβάνει ένα sidebar menu με συνδέσμους προς άλλες σελίδες του ιστότοπου, όπως η αρχική σελίδα, οι ανακοινώσεις, η επικοινωνία, τα έγγραφα του μαθήματος και οι εργασίες. Επιπλέον, εμφανίζεται και ένα κουμπί αποσύνδεσης. Εάν ο χρήστης είναι διαχειριστής, εμφανίζεται επίσης ένας σύνδεσμος για τη διαχείριση των χρηστών. Στο κύριο περιεχόμενο της σελίδας, εμφανίζονται οι υπάρχουσες εργασίες, συμπεριλαμβανομένων των στόχων, της εκφώνησης, των παραδοτέων και της ημερομηνίας παράδοσης. Εάν ο χρήστης είναι διαχειριστής, υπάρχουν σύνδεσμοι για τη διαγραφή και την επεξεργασία κάθε εργασίας. Σε περίπτωση που δεν υπάρχουν εργασίες, εμφανίζεται ένα μήνυμα ενημέρωσης. Τέλος, υπάρχει ένα κουμπί επιστροφής στην κορυφή της σελίδας.

addHomework.php και editHomework.php

Το πρώτο αρχείο αντιπροσωπεύει μια σελίδα για την προσθήκη νέας εργασίας. Αρχικά, ελέγχει αν ο χρήστης είναι συνδεδεμένος στη συνεδρία, και αν όχι, τον ανακατευθύνει στη σελίδα σύνδεσης. Στη συνέχεια, λαμβάνει τα δεδομένα από τη φόρμα που υποβλήθηκε και εκτελεί διάφορους ελέγχους στα δεδομένα. Αν όλα τα πεδία είναι συμπληρωμένα και η ημερομηνία έχει τη σωστή μορφή, τότε προχωρά στην εισαγωγή της εργασίας στη βάση δεδομένων. Εάν υπάρχει κάποιο σφάλμα στην υποβολή της φόρμας ή σε κάποιο από τα δεδομένα, εμφανίζει ένα μήνυμα λάθους. Στη συνέχεια, παρουσιάζει τη φόρμα για την προσθήκη νέας εργασίας, η οποία περιλαμβάνει πεδία για τους στόχους της εργασίας, τα παραδοτέα, την ημερομηνία παράδοσης και το αρχείο εκφώνησης της εργασίας. Τέλος, προβάλλει ένα κουμπί υποβολής και ένα κουμπί επιστροφής στη λίστα με τις εργασίες.

Το περιεχόμενο του δεύτερου αρχείου αφορά την επεξεργασία μιας εργασίας. Πρώτα γίνεται έλεγχος της μεθόδου αίτησης και ανάκτηση του id της εργασίας από το URL σε περίπτωση που η μέθοδος είναι GET. Στη συνέχεια, επιλέγονται οι πληροφορίες της εργασίας από τη βάση δεδομένων και αποθηκεύονται σε μεταβλητές. Εάν η μέθοδος είναι POST, τότε αποθηκεύονται τα νέα δεδομένα της εργασίας από τη φόρμα και γίνεται ενημέρωση των δεδομένων της εργασίας στη βάση δεδομένων. Υπάρχει έλεγχος για κενά πεδία ή μη έγκυρη μορφή ημερομηνίας πριν την ενημέρωση της εργασίας στη βάση δεδομένων. Σε περίπτωση σφάλματος, εμφανίζεται το αντίστοιχο μήνυμα. Αν η ενημέρωση πραγματοποιηθεί επιτυχώς, ο χρήστης ανακατευθύνεται στη σελίδα εργασιών.

Users (users.php addUser.php editUser.php)

Το παραπάνω αρχείο αποτελεί τη σελίδα διαχείρισης χρηστών (users.php). Ξεκινάει με την έναρξη μιας συνεδρίας PHP και συμπεριλαμβάνει το αρχείο functions.php που περιέχει χρήσιμες συναρτήσεις. Έπειτα γίνεται έλεγχος εάν ο χρήστης έχει συνδεθεί. Αν όχι, τον ανακατευθύνει στη σελίδα σύνδεσης (login.php). Μετά γίνεται σύνδεση στη βάση δεδομένων και ορισμός του πίνακα χρηστών ως τον τρέχοντα πίνακα εργασίας και εμφάνιση της HTML σελίδας με τη λίστα των χρηστών. Ακολουθεί η εκτέλεση ερωτήματος SQL για την επιλογή όλων των χρηστών από τον πίνακα users. Τέλος γίνεται εμφάνιση των αποτελεσμάτων σε έναν πίνακα HTML με τα πεδία του χρήστη (όνομα, επώνυμο, email, κωδικός, ρόλος) και δημιουργία συνδέσμων για επεξεργασία και διαγραφή κάθε χρήστη και παροχή συνδέσμου για προσθήκη νέου χρήστη στο τέλος της σελίδας.

Το δεύτερο αρχείο αναφέρεται στην προσθήκη χρήστη. Αυτό το τμήμα επεξεργάζεται την προσθήκη νέου χρήστη στη βάση δεδομένων. Όταν ο χρήστης υποβάλλει τη φόρμα με τα δεδομένα του νέου χρήστη, γίνεται έλεγχος για κενά πεδία και εάν ο κωδικός ήδη υπάρχει. Στη συνέχεια, τα δεδομένα εισάγονται στη βάση δεδομένων και ο χρήστης ανακατευθύνεται στη σελίδα των χρηστών.

Το τρίτο αρχείο αναφέρεται στην επεξεργασία χρήστη. Αυτό το τμήμα ασχολείται με την επεξεργασία των πληροφοριών ενός χρήστη. Όταν ο χρήστης επιλέξει να επεξεργαστεί έναν υπάρχοντα χρήστη, τα δεδομένα του χρήστη φορτώνονται στη φόρμα επεξεργασίας. Όταν ο χρήστης υποβάλλει τη φόρμα, γίνεται έλεγχος για κενά πεδία και έπειτα γίνεται ενημέρωση των δεδομένων του χρήστη στη βάση δεδομένων.

Delete (delete.php)

Αυτό το αρχείο αναλαμβάνει τη διαγραφή μιας εγγραφής από τη βάση δεδομένων με βάση το ID ή το loginame που περνάει μέσω του URL. Ανάλογα με το πώς περνιέται το ID ή το loginame, εκτελεί το κατάλληλο ερώτημα SQL για διαγραφή και στη συνέχεια ανακατευθύνει τον χρήστη στην αντίστοιχη σελίδα μετά τη διαγραφή. Το αρχείο αυτό είναι κοινό για τη διαγραφή ανακοινώσεων, έγγραφων, εργασιών και χρηστών. Αυτό σημαίνει πως αναλαμβάνει όλες τις διαγραφές και δεν υπάρχει διαφορετική διαγραφή για κάθε ένα από αυτά.

Functions.php

Αυτό το αρχείο περιέχει πολλές χρήσιμες συναρτήσεις για την υλοποίηση κάποιων λειτουργιών στον κώδικα:

* connectToDb(&$conn): Συνάρτηση για τη σύνδεση στη βάση δεδομένων MySQL. Αν η σύνδεση αποτύχει, εμφανίζεται ένα μήνυμα σφάλματος.
* validateDate($date, $format = 'Y-m-d'): Συνάρτηση που ελέγχει εάν μια ημερομηνία έχει τη σωστή μορφή, βάσει ενός συγκεκριμένου μοτίβου (προεπιλεγμένα το 'Y-m-d').
* stringIsNullOrWhitespace($text): Συνάρτηση που ελέγχει εάν ένα κείμενο είναι κενό ή περιέχει μόνο κενά.
* uploadFile($pwd): Συνάρτηση που χρησιμοποιείται για τη μεταφόρτωση ενός αρχείου στον διακομιστή. Επιστρέφει τον αποθηκευμένο διαδρόμο του αρχείου.
* hidePwd($pwd): Συνάρτηση που χρησιμοποιείται για να αποκρύψει τον κωδικό πρόσβασης, αντικαθιστώντας κάθε χαρακτήρα με '\*'.
* emailExists($email): Συνάρτηση που ελέγχει εάν ένα email υπάρχει ήδη στη βάση δεδομένων. Επιστρέφει 1 αν το email υπάρχει και 0 αν δεν υπάρχει.

Style.css , Αρχεία DOC και Φωτογραφία png

Εννοείται ότι υπάρχει και το αρχείο με το οποίο δίνουμε στυλ στην ιστοσελίδα μας που περιέχει πολλές κλάσεις ανάλογα τις περιπτώσεις. Βρίσκεται στο φάκελο Style. Στο φάκελο IMAGES βρίσκεται επίσης και η φωτογραφία η οποία εμφανίζεται στην Αρχική Σελίδα. Τέλος, υπάρχουν και τα αρχεία doc τα οποία κατεβαίνουν όταν πατάς τον κατάλληλο σύνδεσμο στην ιστοσελίδα. Βρίσκονται στο φάκελο uploads.

***DATABASE***

Student4038.sql

Πρόσβαση στη βάση δεδομένων -----> Username = **evangdimos** password = **12345**

Οι πίνακες που υπάρχουν στη βάση δεδομένων `student4038` και η λειτουργία τους είναι:

* Πίνακας `announcements`:
  + Αυτός ο πίνακας χρησιμοποιείται για την αποθήκευση ανακοινώσεων.
  + Κάθε ανακοίνωση έχει ένα μοναδικό αριθμό αναγνώρισης (`id`), ημερομηνία, θέμα και μήνυμα.
  + Ο πίνακας αυτός επιτρέπει στους χρήστες (είτε φοιτητές είτε καθηγητές) να δημοσιεύουν ανακοινώσεις σχετικές με το μάθημα.
* Πίνακας `homework`:
  + Αυτός ο πίνακας αποθηκεύει πληροφορίες σχετικά με τις εργασίες που δίνονται στους φοιτητές.
  + Κάθε εργασία έχει ένα μοναδικό αριθμό αναγνώρισης (`id`), στόχους, τοποθεσία (π.χ. URL ή φάκελος), απαιτούμενα αρχεία και ημερομηνία παράδοσης.
  + Ο πίνακας αυτός βοηθά τους φοιτητές να οργανώνουν και να παρακολουθούν τις εργασίες τους.
* Πίνακας `documents`:
  + Αυτός ο πίνακας αποθηκεύει τα έγγραφα που σχετίζονται με το μάθημα, όπως παρουσιάσεις, αρχεία PDF, κτλ.
  + Κάθε έγγραφο έχει ένα μοναδικό αριθμό αναγνώρισης (`id`), τίτλο, περιγραφή και τοποθεσία (π.χ. URL ή φάκελος).
  + Αυτός ο πίνακας επιτρέπει την εύκολη πρόσβαση σε διάφορα έγγραφα και υλικά μαθήματος.
* Πίνακας `users`:
  + Αυτός ο πίνακας περιέχει πληροφορίες για τους χρήστες του συστήματος.
  + Κάθε χρήστης έχει ένα μοναδικό όνομα σύνδεσης (`loginame`), όνομα, επώνυμο, κωδικό πρόσβασης και ρόλο (είτε φοιτητής είτε καθηγητής).
  + Αυτός ο πίνακας διαχειρίζεται τα στοιχεία των χρηστών και τους επιτρέπει την πρόσβαση στο σύστημα με τις κατάλληλες δικαιώματα.

# Ευχαριστώ πολύ για την προσοχή σας!