# Technical Report

**Introduction**

Κύριος στόχος είναι η δημιουργία μίας πλατφόρμας τύπου e-learning που θα δίνει σε μαθητές ανά τον κόσμο τη δυνατότητα να εντρυφήσουν σε ένα πλήθος μαθημάτων. Για την πρόσβαση στο περιεχόμενο της ιστοσελίδας απαιτείται εγγραφή , ενώ δίνεται η δυνατότητα τροποποίησης του περιεχομένου της ιστοσελίδας , ανάλογα με τον ρόλο του κάθε χρήστη. Επομένως προηγείται μια διαδικασία ταυτοποίησης σε κάθε προσπάθεια πρόσβασης στο βασικό περιεχόμενο της ιστοσελίδας.

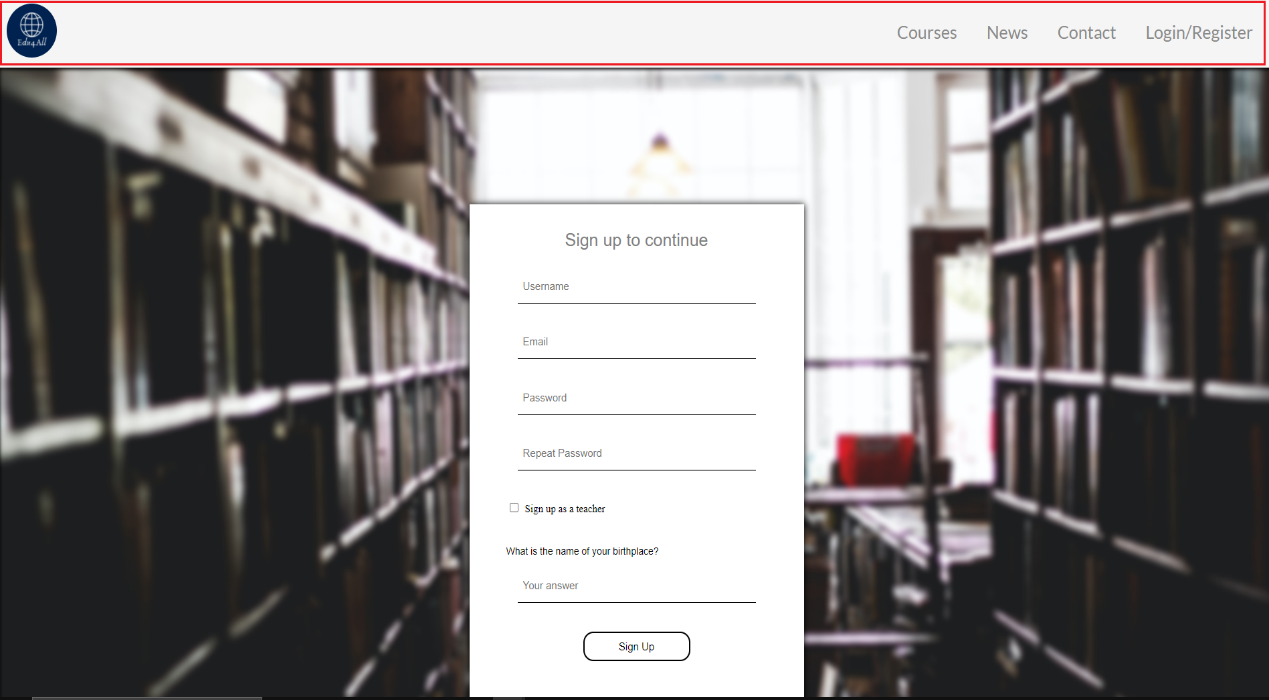
**Responsiveness**

Οι παρακάτω σελίδες έχουν σχεδιαστεί ώστε να είναι responsive προς όλες τις συσκευές ανεξαρτήτους μεγέθους οθόνης.

**Α. Register Page**

Η πρόσβαση στο αρχείο register.php γίνεται μέσω ενός hyperlink που βρίσκεται στην login.php αρχείο και χρησιμοποιείται για την εγγραφή ενός νέου χρήστη και την αποθήκευση των στοιχείων του σε μία βάση δεδομένων.

**Navigation Bar**



Περιλαμβάνει το βασικό μενού πλοήγησης στην ιστοσελίδα

* **Εικόνα (Image)**

πρόκειται για την αρχική σελίδα (Homepage) που εμφανίζεται όταν κάποιος ανοίξει την ιστοσελίδα

* **Courses**

περιλαμβάνει το σύνολο των μαθημάτων που είναι διαθέσιμα για προβολή

* **News**

περιλαμβάνει ένα σύνολο άρθρων σχετικά με την εκπαίδευση

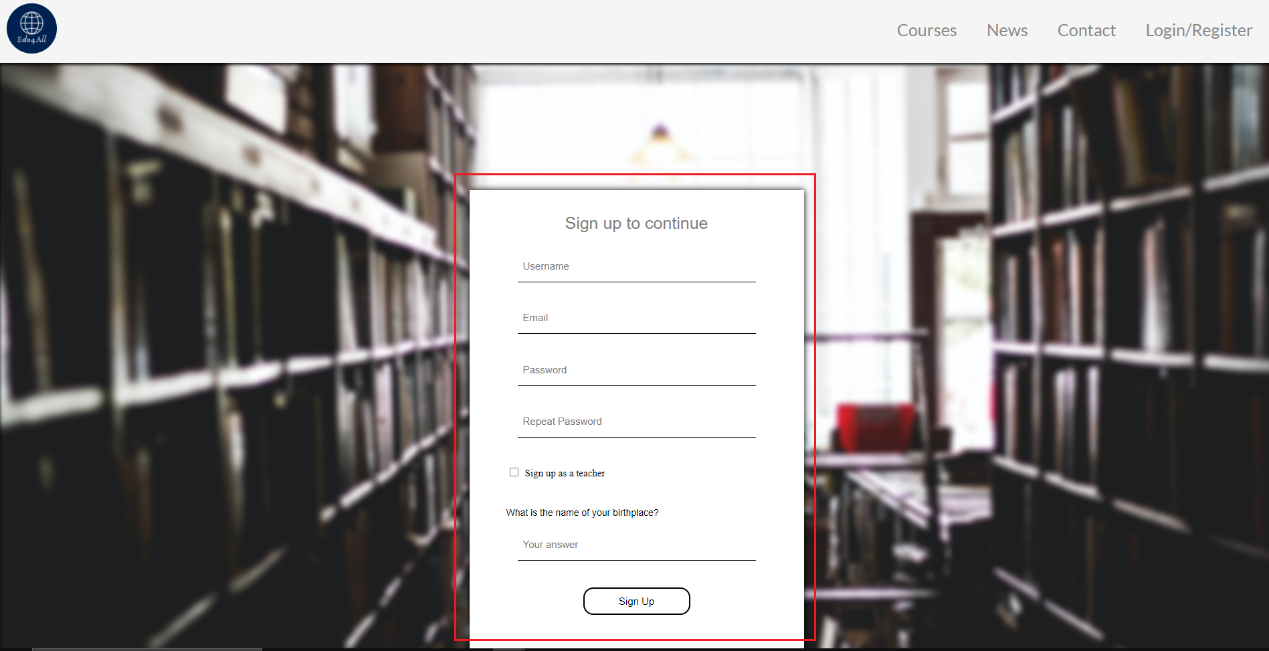
* **Contact**

Δίνει την δυνατότητα επικοινωνίας των χρηστών με τους developers

* **Login/Register**

Hyperlink στο login.php αρχείο

**Register Form**

****

Περιλαμβάνει τα εξής πεδία :

* **Username:** τύπος “text”, υποχρεωτική συμπλήρωση.
* **Email**: τύπος “email”, υποχρεωτική συμπλήρωση.
* **Password / Repeat Password**: τύπος “password”, υποχρεωτική συμπλήρωση.
* **Sign up as a teacher**: δίνει τη δυνατότητα δύο επιπέδων χρηστών:

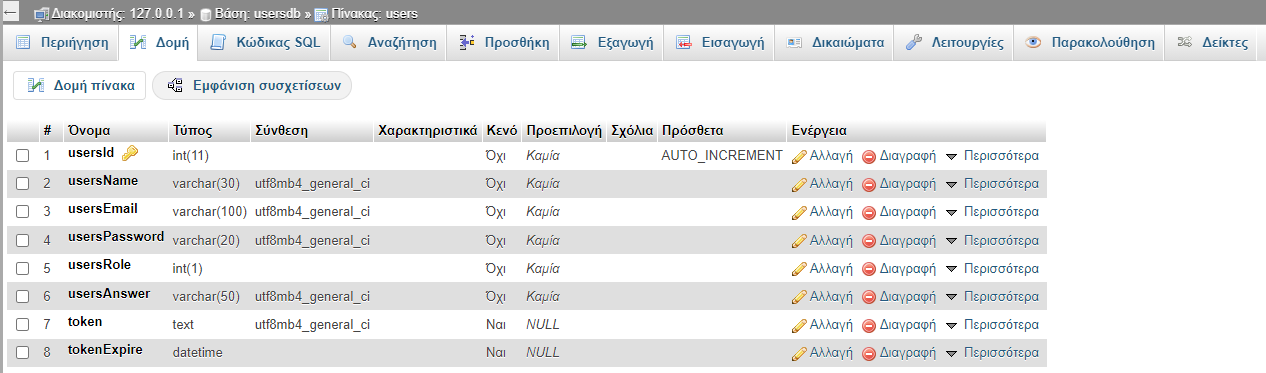
1. Student: μπορεί μόνο να προβάλλει το περιεχόμενο του Courses.
2. Teacher: μπορεί να προβάλλει αλλά και να μεταβάλει το περιεχόμενο του Courses, έχει την δυνατότητα προσθήκης ενός νέου μαθήματος.

* **Your answer**: τύπος “text», υποχρεωτική συμπλήρωση, μπορεί να χρησιμοποιηθεί ως authentication factor σε περίπτωση που ένας χρήστης ξεχάσει τον κωδικό του.
* **Sign up**: χρησιμοποιείται για την ολοκλήρωση της εγγραφής

Ύστερα από την συμπλήρωση των απαραίτητων πεδίων του Register form , τα δεδομένα αυτά χρησιμοποιούνται για την δημιουργία ενός νέου λογαριασμού και αποθηκεύονται στη βάση δεδομένων υπεύθυνη για το σύνολο των εγγεγραμμένων χρηστών.

**SQL Database**

Για την δημιουργία της βάσης δεδομένων των χρηστών χρησιμοποιείται SQL για την δημιουργία μίας σχεσιακής βάσης δεδομένων. Συγκεκριμένα , δημιουργώ έναν νέο πίνακα ‘users’ στον οποίο εισάγω τα δεδομένα που έχουν συμπληρωθεί στην Register form.



**Table ‘users’:**

* **usersId**: int (11) AUTO INCREMENTED, χρησιμοποιείται ως primary key
* **usersName:** χρησιμοποιείται για την αποθήκευση του $\_POST[‘username’]
* **usersEmail:** χρησιμοποιείται για την αποθήκευση του $\_POST[‘email’]
* **usersPassword:** χρησιμοποιείται για την αποθήκευση του $\_POST[‘password’]
* **usersRole:** ελέγχεται αν το checkbox ‘Sign up as a teacher’ είναι τσεκαρισμένο, εάν όντως είναι αποθηκεύεται η τιμή 1, αλλιώς η τιμή 0
* **usersAnswer:** χρησιμοποιείται για την αποθήκευση του $\_POST[‘answer’]
* **token:** χρησιμοποιείται σε περίπτωση που ο χρήστης έχει ξεχάσει τον κωδικό του (προεπιλογή NULL)
* **tokenExpire:** σηματοδοτεί την διάρκεια ισχύος του token (προεπιλογή NULL)

Εφόσον έχουν συμπληρωθεί τα απαραίτητα πεδία , με το πάτημα του button ‘Sign up’ , υλοποιείται η δημιουργία μίας νέας εγγραφής στον πίνακα ‘users’ (register.inc.php).

Στο register.inc.php , αρχικά , πραγματοποιείται έλεγχος εάν η πρόσβαση σε αυτήν την .php σελίδα έγινε μέσω του Submit-button(‘Sign up’). Στη συνέχεια λαμβάνονται τα δεδομένα του Register form και πραγματοποιείται η σύνδεση με την βάση δεδομένων των χρηστών(db.inc.php).Προτού δημιουργηθεί μία νέα εγγραφή πραγματοποιούνται κάποιοι έλεγχοι, εάν το E-mail υπάρχει ήδη αποθηκευμένο στη βάση δεδομένων κάτω από άλλον χρήστη , εάν τα πεδία ‘Password’ και ‘Repeat Password’ είναι όμοια, καθώς και αν ο κωδικός του χρήστη είναι τουλάχιστον 8 ψηφία ενώ παράλληλα ενημερώνει το χρήστη για το error που προέκυψε. Εάν περάσει όλους τους ελέγχους , δημιουργείται μία νέα εγγραφή στον table ‘users’ και μεταφέρεται στην login.php.

**Πειραματισμός**

Αρχικά , δημιούργησα δύο διαφορετικούς πίνακες , έναν πίνακα ‘users’ που περιλαμβάνει όλα τα προαναφερθέντα πεδία πλην των token και tokenExpire , και έναν πίνακα ‘forgotUsers’ που περιλάμβανε πεδία usersId , token και tokenExpire και αφορούσε χρήστες που είχαν ξεχάσει τον κωδικό τους. Στη τελική , θεώρησα πιο αποδοτικό να τα συμπεριλάβω σε έναν πίνακα ώστε η ενημέρωση του κωδικού και ο έλεγχος των token και tokenExpire να είναι ευκολότερος (μπορούν να υλοποιηθούν όλα με queries σε έναν table μόνο).

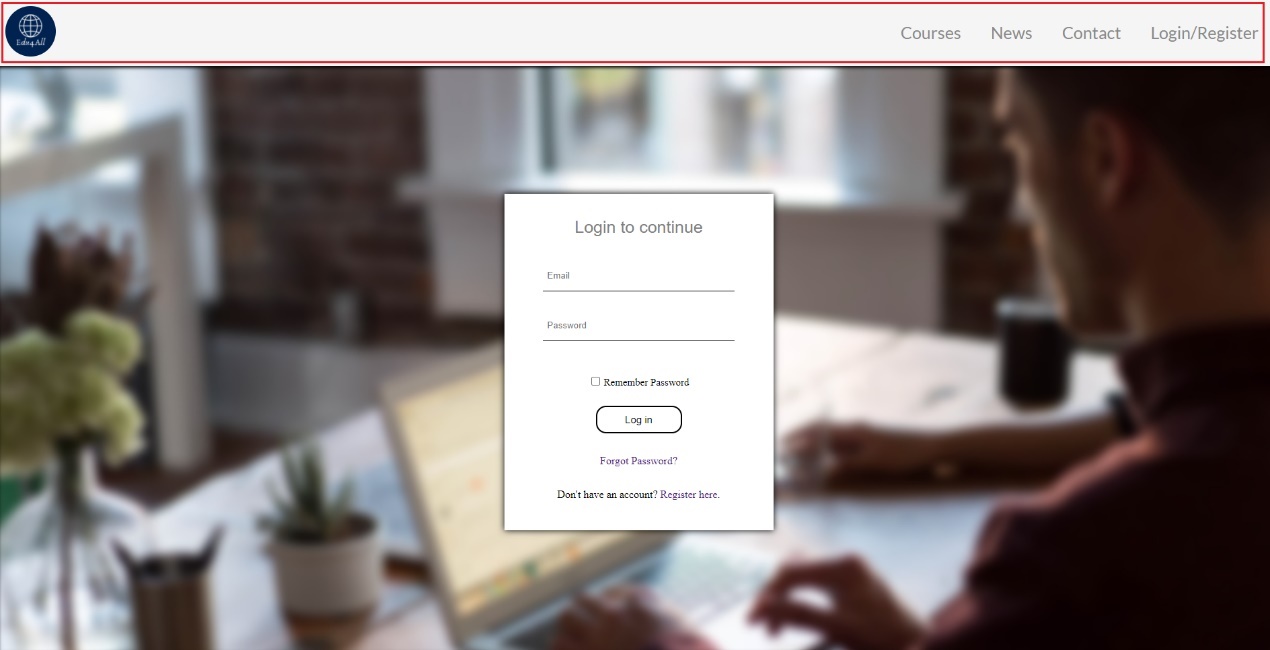
**Μελλοντική Επέκταση**

Ως μελλοντική επέκταση , θα μπορούσαν οι κωδικοί να αποθηκεύονται encrypted-hashed ως ένα επιπλέον μέτρο προστασίας ευαίσθητων δεδομένων.

**Β. Login Page**

Η πρόσβαση στο αρχείο login.php γίνεται μέσω του Homepage.php και χρησιμοποιείται για την είσοδο ενός εγγεγραμμένου χρήστη στην ιστοσελίδα.

**Navigation Bar**



Περιλαμβάνει το βασικό μενού πλοήγησης στην ιστοσελίδα

* **Εικόνα (Image)**

πρόκειται για την αρχική σελίδα (Homepage) που εμφανίζεται όταν κάποιος ανοίξει την ιστοσελίδα

* **Courses**

περιλαμβάνει το σύνολο των μαθημάτων που είναι διαθέσιμα για προβολή

* **News**

περιλαμβάνει ένα σύνολο άρθρων σχετικά με την εκπαίδευση

* **Contact**

Δίνει την δυνατότητα επικοινωνίας των χρηστών με τους developers

* **Login/Register**

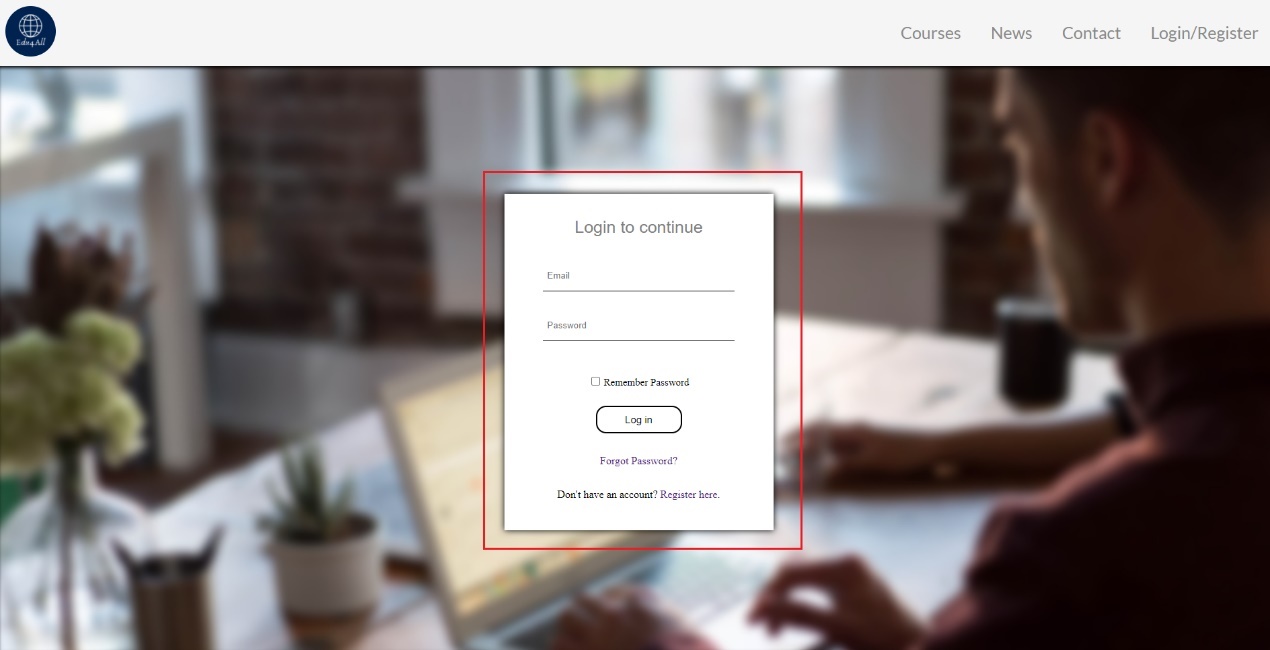
Το php αρχείο που προβάλλεται στην εικόνα

**Login Menu**

Ο χρήστης πρέπει να συμπληρώσει το Email και Password του λογαριασμού με τα οποία είχε κάνει την εγγραφή του.

Τα δύο πεδία Email και Password είναι υποχρεωτικά ,ενώ η μορφή συμπλήρωσής τους είναι συγκεκριμένη ( Email : [example@a.gr](mailto:example@a.gr) , Password δέχεται όλα τα σύμβολα τα οποία ,στη συνέχεια, αποκρύπτει κατά την συμπλήρωση τους ).

Επιπλέον δίνεται η δυνατότητα Remember Password που δημιουργεί δύο Cookies , ένα για το email και ένα για το password τα οποία έχουν διάρκεια ζωής ίση με μία μέρα. Εάν αυτά τα Cookies υπάρχουν , τότε ορίζουν τα values των δύο πεδίων.



Εφόσον έχουν συμπληρωθεί τα απαραίτητα πεδία , με το πάτημα του button ‘Log in’, υλοποιείται η είσοδος του χρήστη στην ιστοσελίδα(μετάβαση στο login.inc.php)

Στο login.inc.php , αρχικά , πραγματοποιείται έλεγχος εάν η πρόσβαση σε αυτήν την .php σελίδα έγινε μέσω του Submit-button(‘Log in’). Στη συνέχεια λαμβάνονται τα δεδομένα του Login form και πραγματοποιείται η σύνδεση με την βάση δεδομένων των χρηστών(db.inc.php).Αρχικά ελέγχεται εάν υπάρχει το συγκεκριμένο mail στον ‘users’ table ,και στη συνέχεια εάν τα πεδία ‘usersPassword’ και $\_POST[‘password’] είναι ίδια. Εάν η ταυτοποίηση είναι επιτυχής , τότε αρχίσει ένα session και δημιουργούνται δύο νέες $\_SESSION vars ,

* $\_SESSION[‘loggedIn’]=true , ελέγχεται εάν είναι true προκειμένου να έχει ο χρήστης πρόσβαση στις μπλοκαρισμένες καρτέλες (Courses , News , Contact)
* $\_SESSION[‘userEmail’]=$email , χρησιμοποιείται για να δούμε εάν πρόκειται για συνδεδεμένο Καθηγητή ή Μαθητή (οι καθηγητές έχουν την δυνατότητα να προσθέτουν μαθήματα)
* Ακολουθεί το direct στο Homepage.php

Εάν υπάρξει σφάλμα ( δεν περάσει τους ελέγχους) κατάλληλο μήνυμα τυπώνεται στον χρήστη.

**Footer Section (ΒΑΛΤΕ FOOTER ΑΥΤΟ ΠΟΥ ΚΑΝΑΤΕ ΕΣΕΙΣ ΜΕ ΤΑ ΤΡΙΑ ΠΛΑΙΣΙΑ)!!**



Το πρώτο πεδίο δίνει περαιτέρω πληροφορίες σχετικά με τους developers (About us), το δεύτερο περιλαμβάνει links τα οποία οδηγούν σε δύο html αρχεία που περιλαμβάνουν οδηγίες για μαθητές και καθηγητές, ενώ το τελευταίο πεδίο περιλαμβάνει πληροφορίες επικοινωνίας . Και τα τρία πλαίσια υλοποιούν ένα animation ( EventListener (‘mouseover’) ) αποκαλύπτοντας το περιεχόμενό τους.

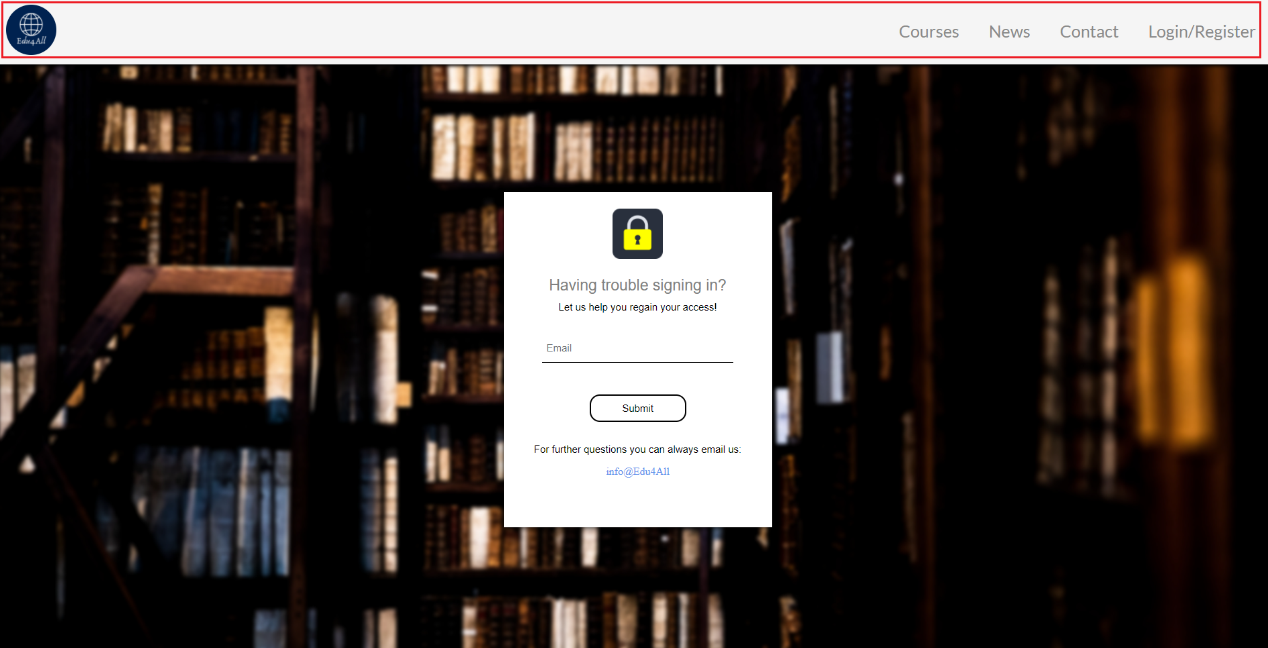
**Πειραματισμός**

Αρχικά, υλοποίησα σύνδεση με βάση το Username και Password του χρήστη , ωστόσο πιο πιθανό είναι να θυμάται ένας χρήστης το email που έκανε εγγραφή έναντι του Username του, επομένως προτίμησα την επιλογή του Email .

**Μελλοντική Επέκταση**

Σύνδεση μέσω Microsoft account , Facebook (OAuthentication).

**C. Forgot Page**



**Navigation Bar**

Περιλαμβάνει το βασικό μενού πλοήγησης στην ιστοσελίδα

* **Εικόνα (Image)**

πρόκειται για την αρχική σελίδα (Homepage) που εμφανίζεται όταν κάποιος ανοίξει την ιστοσελίδα

* **Courses**

περιλαμβάνει το σύνολο των μαθημάτων που είναι διαθέσιμα για προβολή

* **News**

περιλαμβάνει ένα σύνολο άρθρων σχετικά με την εκπαίδευση

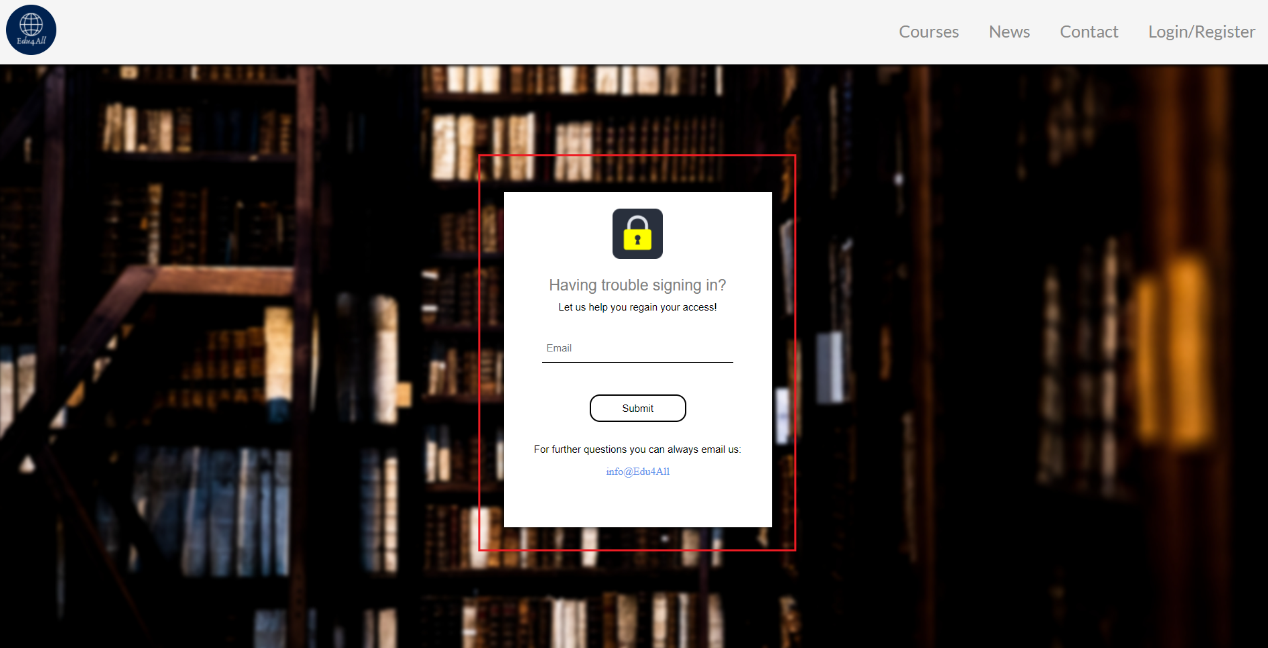
* **Contact**

Δίνει την δυνατότητα επικοινωνίας των χρηστών με τους developers

* **Login/Register**

Το php αρχείο του Login Page

**Forgot Form**



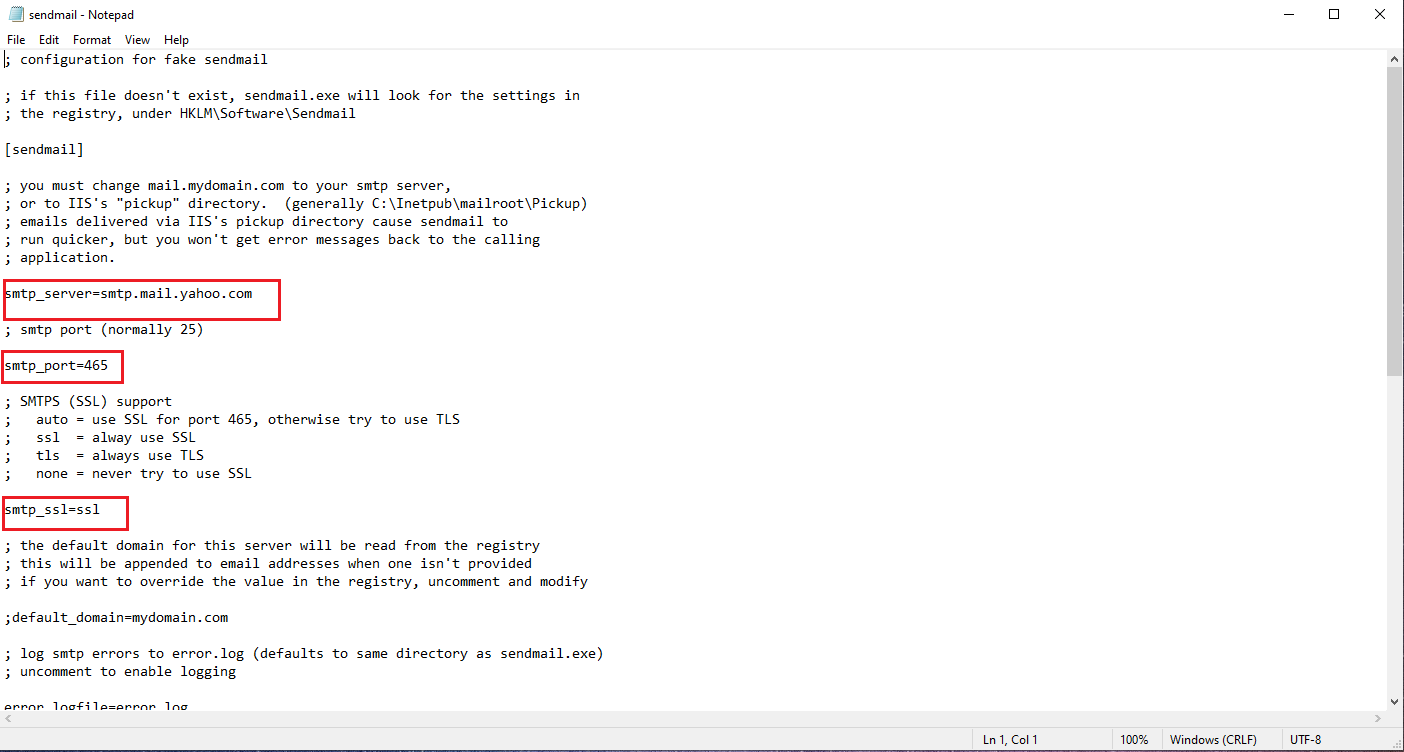
Περιλαμβάνει τo πεδίο:

* **Email**: τύπος “email”, υποχρεωτική συμπλήρωση.

Εφόσον έχουν συμπληρωθεί το πεδίο Email, με το πάτημα του button ‘Submit’, μεταβαίνει στο forgottenpassword.inc.php όπου υλοποιείται η αποστολή ενός email στο email του εγγεγραμμένου χρήστη για την αλλαγή του κωδικού.

Συγκεκριμένα, χρησιμοποιείται το SMTP του Yahoo προκειμένου να αποστέλλεται αυτόματα ένα email.

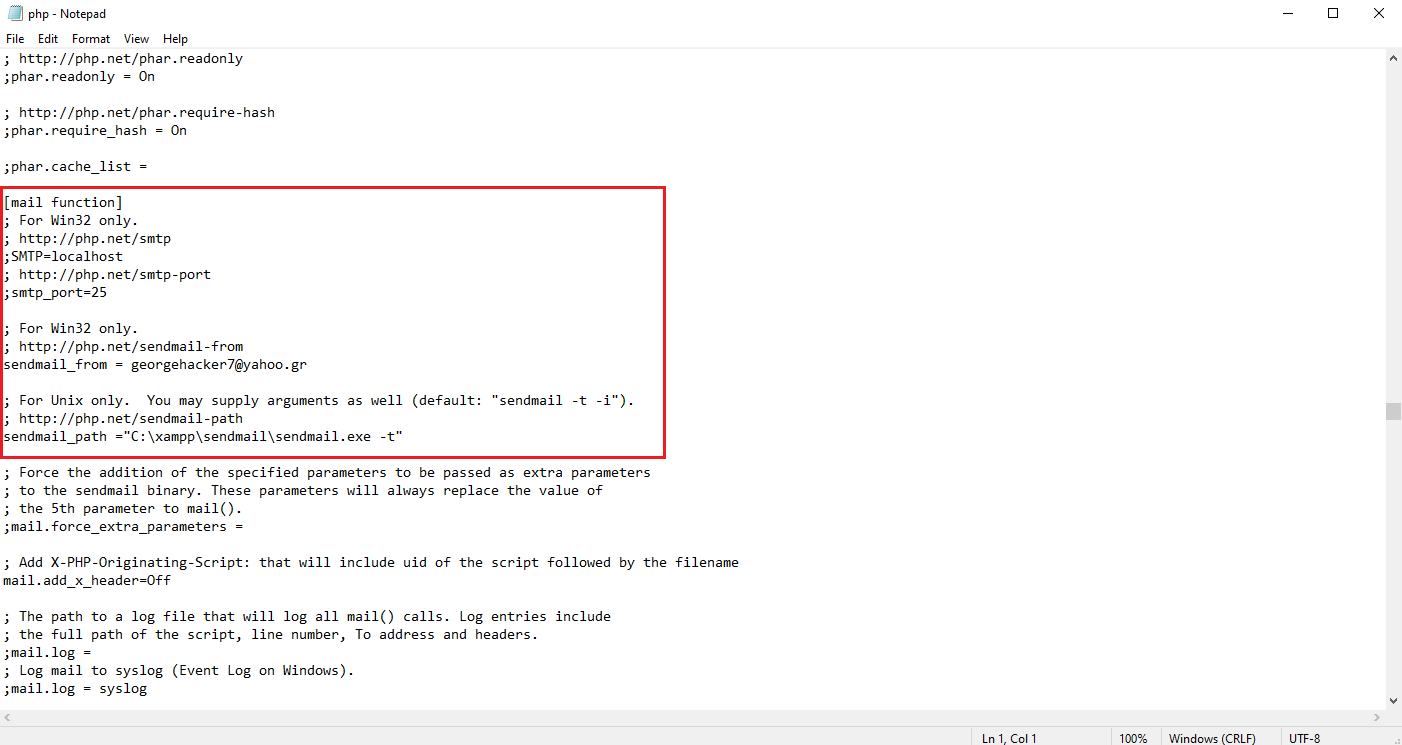
**C:\xampp\sendmail\sendmail.ini**



+auth\_username: ……

+auth\_password: ……

**C:\xampp\php\php.ini**

****

Αρχικά ,στο forgottenpassword.inc.php πραγματοποιείται έλεγχος εάν η πρόσβαση σε αυτήν την .php σελίδα έγινε μέσω του Submit-button(‘Submit’). Στη συνέχεια λαμβάνονται τα δεδομένα του Forgot form και πραγματοποιείται η σύνδεση με την βάση δεδομένων των χρηστών(db.inc.php). Εάν το mail υπάρχει στον πίνακα ‘users’(υπάρχει εγγεγραμμένος χρήστης) τότε κάνουμε generate ένα τυχαίο token (τύπου text) και ενημερώνουμε τα πεδία token και tokenExpire αυτού (tokenExpire : (τωρινή χρονική στιγμή) + 10 λεπτά ).

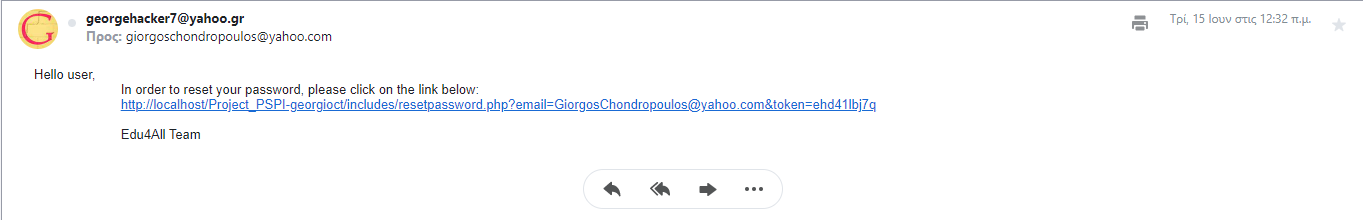
Στη συνέχεια αποστέλλεται ένα email στο email του χρήστη που έκανε το submit , που περιλαμβάνει ως $\_GET vars το email και token του(resetpassword.php).

Στο resetpassword.php ,αρχικά ελέγχεται εάν οι $\_GET vars έχουν λάβει τιμές(σωστός τρόπος πρόσβασης).Στη συνέχεια , αφού πάρει τις δύο αυτές τιμές κάνει τον απαραίτητο έλεγχο του tokenExpire(το token έχει χρόνο ισχύος από τη στιγμή που θα γίνει το Submit +10 λεπτά , εάν περάσει το χρονικό διάστημα αυτό τότε δεν ισχύει).Εάν είναι εντός του χρονικού ορίου , κάνει generate έναν τυχαίο νέο κωδικό , ενημερώνει τον πίνακα ‘users’ και τον προβάλλει στο χρήστη μέσω ενός alert μηνύματος , το οποίο θα κλείσει μόλις ο χρήστης πατήσει το OK.Τέλος , μεταφέρεται στο login.php , με μία $\_GET var , που δείχνει εάν το reset ήταν επιτυχές ή όχι.

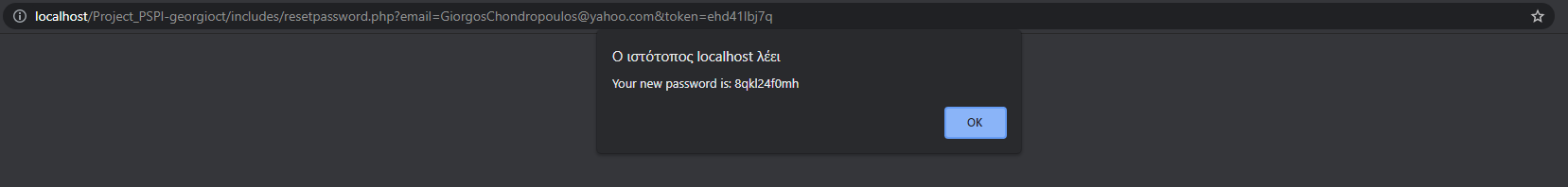
**Πριν το submit:**



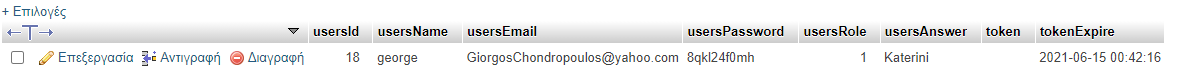
**Αποστολή email:**



**Απεσταλμένο Link:**



**Ενημερωμένη Βάση:**



**Πειραματισμός**

Θα μπορούσε να χρησιμοποιηθεί το πεδίο usersAnswer ως authentication factor , ωστόσο με την αποστολή email εγγυόμαστε την ανανέωση του κωδικού και την απόκτηση πρόσβασης του χρήστη(ο χρήστης μπορεί να έχει ξεχάσει την απάντηση που είχε δώσει όταν έκανε εγγραφή).

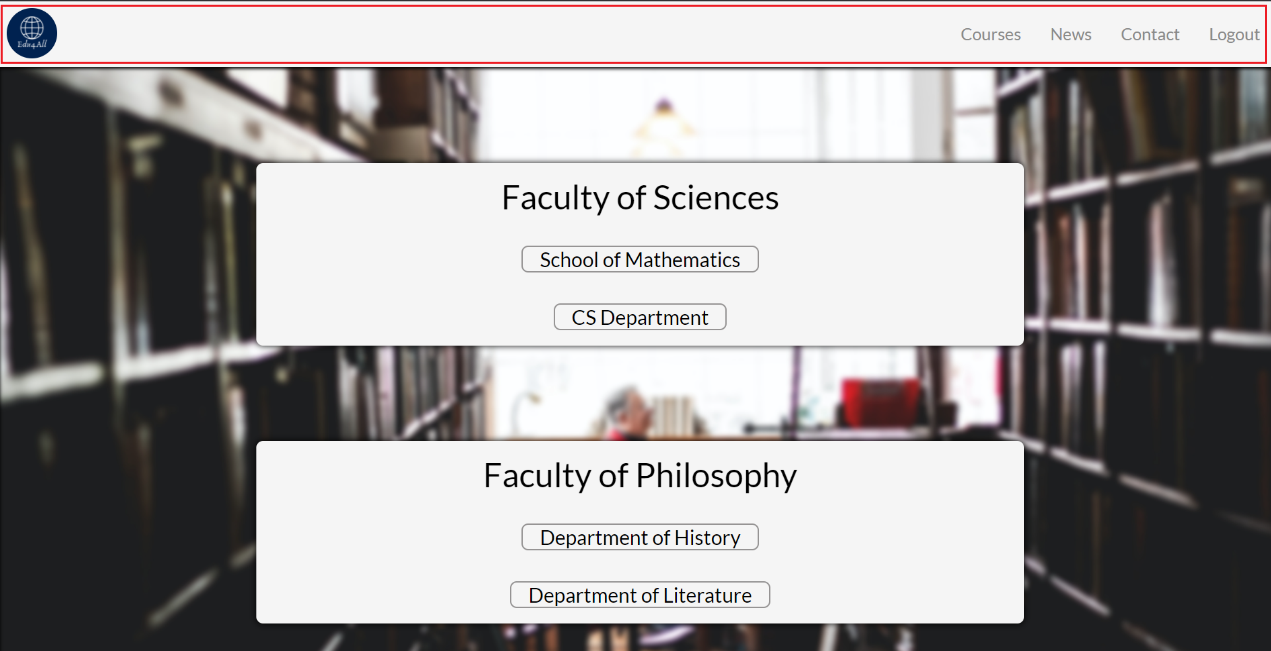
**Μελλοντική Επέκταση**

Δυνατότητα επιλογής Reset με χρήση της ερώτησης που είχε γίνει κατά την εγγραφή έναντι απευθείας αποστολής email.

**D. Courses**

Το περιεχόμενο των Courses (και όλων των redirect του) είναι διαθέσιμο εφόσον ο χρήστης έχει κάνει Login με τα απαραίτητα στοιχεία. Εάν ο χρήστης δεν είναι συνδεδεμένος και προσπαθήσει να αποκτήσει πρόσβαση (μέσω path) , μεταφέρεται στο login.php.

**Courses.php (ΒΑΛΤΕ ΩΣΤΕ ΝΑ ΦΑΙΝΕΤΑΙ ΚΑΙ ΤΟ SEARCH BAR)!!**

****

Περιλαμβάνει το βασικό μενού πλοήγησης στην ιστοσελίδα

* **Εικόνα (Image)**

πρόκειται για την αρχική σελίδα (Homepage) που εμφανίζεται όταν κάποιος ανοίξει την ιστοσελίδα

* **Courses**

περιλαμβάνει το σύνολο των μαθημάτων που είναι διαθέσιμα για προβολή

* **News**

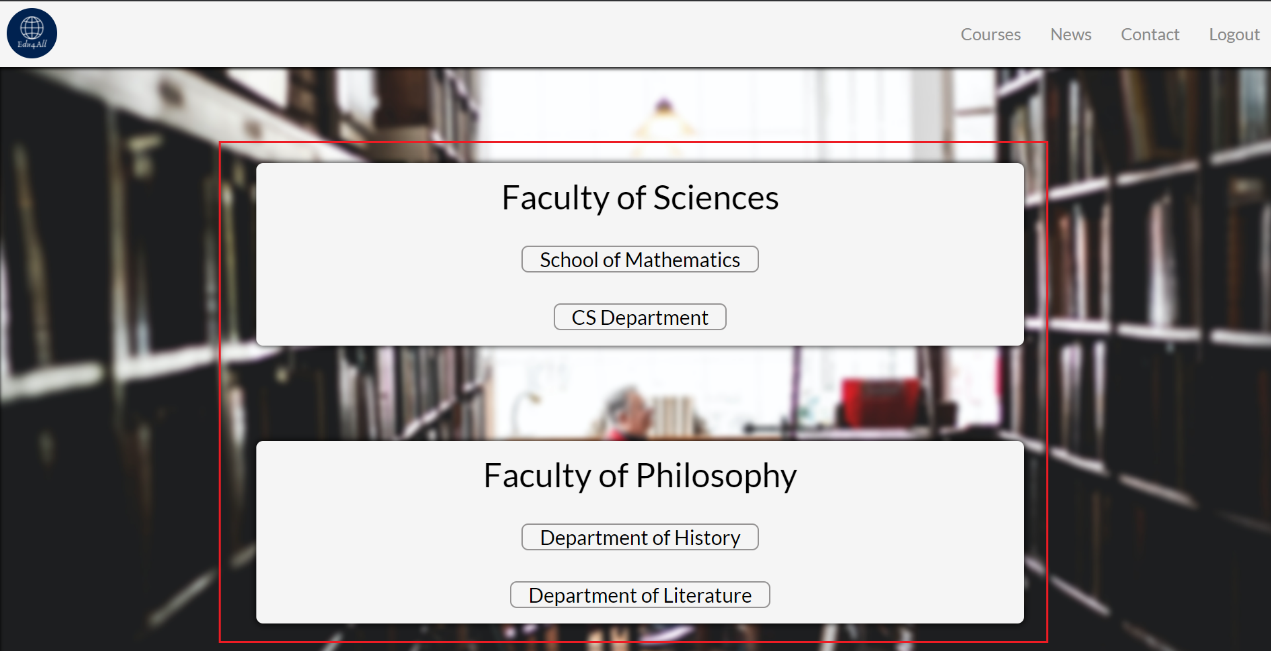
περιλαμβάνει ένα σύνολο άρθρων σχετικά με την εκπαίδευση

* **Contact**

Δίνει την δυνατότητα επικοινωνίας των χρηστών με τους developers

* **Logout**

Τρέχει ένα script το οποίο διαγράφει το τρέχον session (logout.inc.php) , αποκλείοντας με αυτόν τον τρόπο την πρόσβαση σε Courses , News , Contact και ανακατευθύνει τον χρήστη στην αρχική σελίδα.



Τα μαθήματα που παρέχονται μοιράζονται σε δύο Faculties :

1. Faculty of Sciences
2. Faculty of Philosophy

Ενώ παρέχονται ,σε αρχικό στάδιο, 4 διαφορετικά Departments στα οποία ένα μάθημα μπορεί να ενταχθεί :

1. School of Mathematics
2. Computer Science Department
3. Department of History
4. Department of Literature

Με βάση τα δύο επίπεδα χρηστών , ένας καθηγητής έχει την δυνατότητα να προσθέσει ένα μάθημα. Προκειμένου να γίνει αυτό , συνδεόμαστε με την βάση δεδομένων των χρηστών , και χρησιμοποιώντας τη $\_SESSION[‘userEmail’] ελέγχουμε εάν το πεδίο usersRole του συνδεδεμένου χρήστη είναι 1:Teacher (οπότε κάνουμε echo το button για προσθήκη μαθήματος) ή 0:Student(δεν εμφανίζουμε τίποτα καινούργιο).

**Μελλοντική Επέκταση**

Δυνατότητα προσθήκης περισσοτέρων Faculties όπως και περισσοτέρων Departments, ώστε να μην υπάρχει ο περιορισμός των επιλογών αυτών.

**Πειραματισμός**

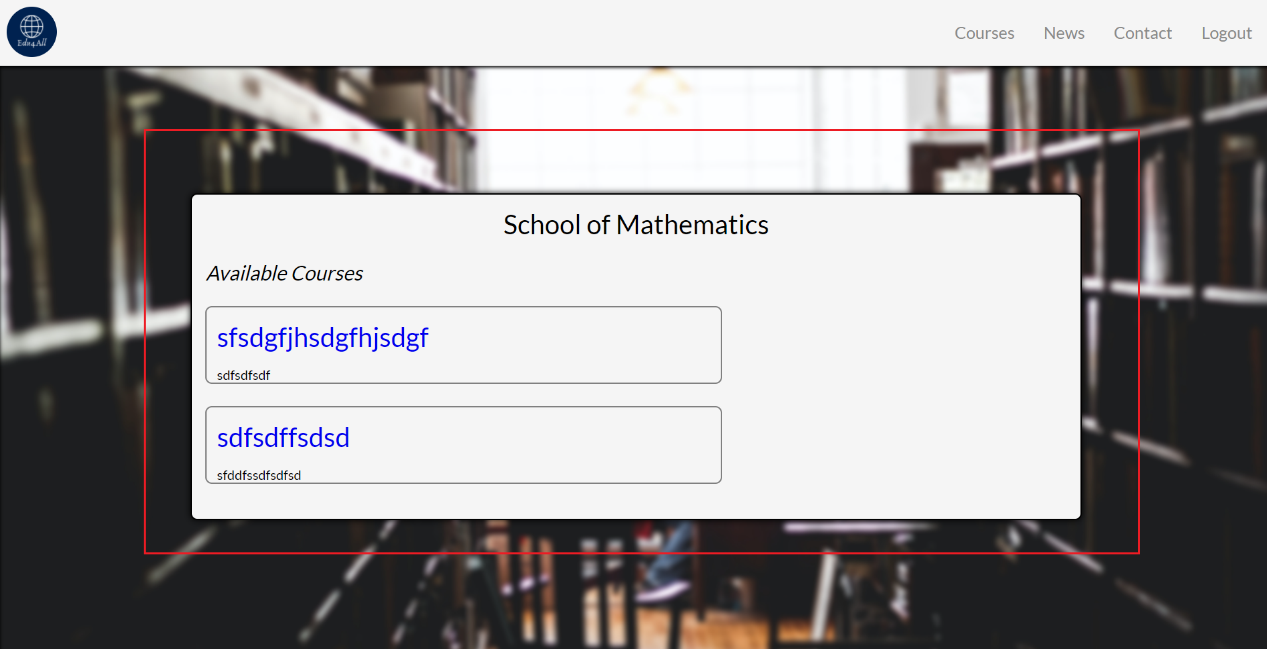
Αρχικά , όλα τα μαθήματα προβάλλονταν κάτω από το Department που ανήκαν, ωστόσο προκειμένου να είναι πιο ευδιάκριτο το περιεχόμενο και ο κώδικας έγινε χωρισμός αυτού σε δύο μέρη(Courses.php και Courses.inc.php)

**Courses.inc.php (ΒΑΛΤΕ ΩΣΤΕ ΝΑ ΦΑΙΝΟΝΤΑΙ ΤΑ ΜΑΘΗΜΑΤΑ, ΕΠΙΛΟΓΗ SCHOOL OF MATHEMATICS)!!**

Με την επιλογή ενός Department μεταφερόμαστε στο Courses.inc.php , στο οποίο ανάλογα με την επιλογή μας γίνεται η προβολή των κατάλληλων μαθημάτων.

Περιλαμβάνει το μενού πλοήγησης του Courses.php .

ΠΧ. Επιλογή School of Mathematics



Με την επιλογή ‘School of Mathematics’ στο Courses.php , μεταφερόμαστε στο Courses.inc.php με μία μεταβλητή $\_GET[‘Subject’] , που χρησιμοποιείται για τον ορισμό του Department.Ανάλογα με την τιμή της global $\_GET[‘Subject’] ορίζεται ο τίτλος καθώς και το διαθέσιμο περιεχόμενο. Συνδεόμαστε με την βάση δεδομένων των Courses (db.connect.php) και κάνουμε fetch τα Title και Introduction πεδία των μαθημάτων που ανήκουν στο ορισμένο Department.

**Μελλοντική Επέκταση**

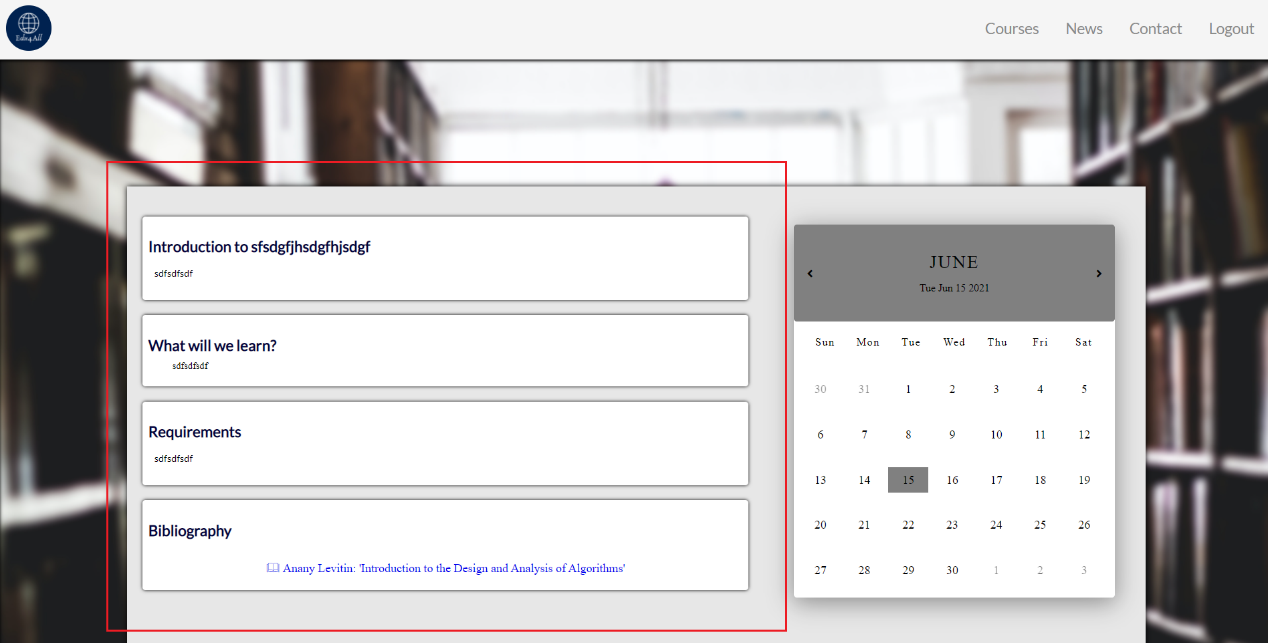
Προσθήκη των στοιχείων των διδασκόντων στον κενό χώρο ώστε κάθε μαθητής να γνωρίσει σε κάποιον βαθμό τον διδάσκοντα του.

**Courses\_final.php (ΕΠΙΛΟΓΗ APPLIED MATHEMATICS)**

Προβάλλει το περιεχόμενο όλων των πεδίων του επιλεγμένου μαθήματος (Title , Introduction , Chapters , Requirements , Bibliography).

Περιλαμβάνει το μενού πλοήγησης του Courses.php .

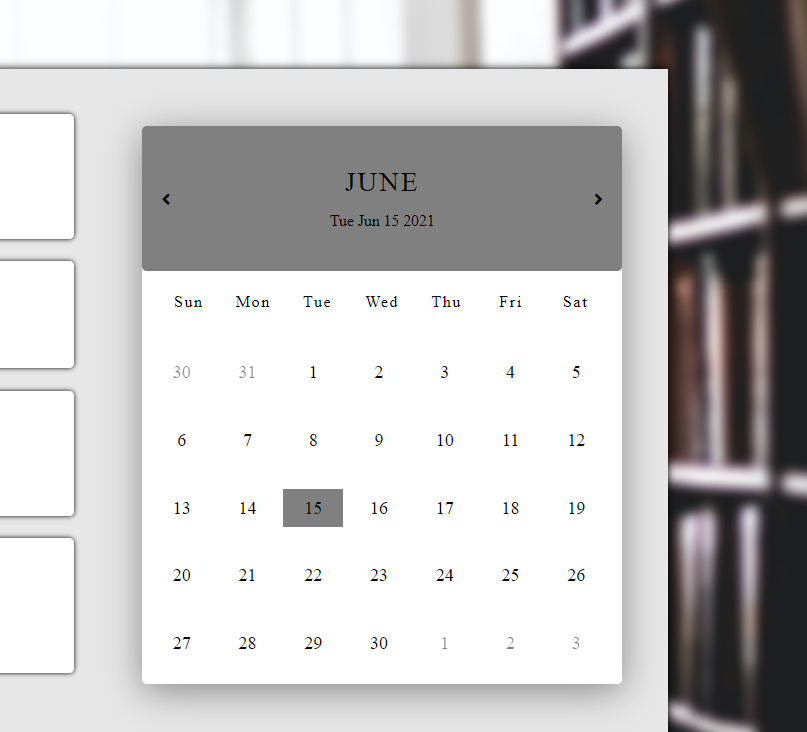
ΠΧ. Επιλογή Applied Mathematics:



Με την επιλογή ενός μαθήματος από το Courses.inc.php , περνάει μία global $\_GET[‘Title’] variable η οποία αναφέρεται στο πεδίο Title του συγκεκριμένου μαθήματος.

Με αυτόν τον τρόπο , το Courses\_final.php συνδέεται με την βάση δεδομένων των Courses και με κατάλληλο SQL query (Χρησιμοποιώντας την $\_GET variable) κάνει fetch τα πεδία του επιλεγμένου μαθήματος τα οποία στη συνέχεια προβάλλει.

**Calendar**



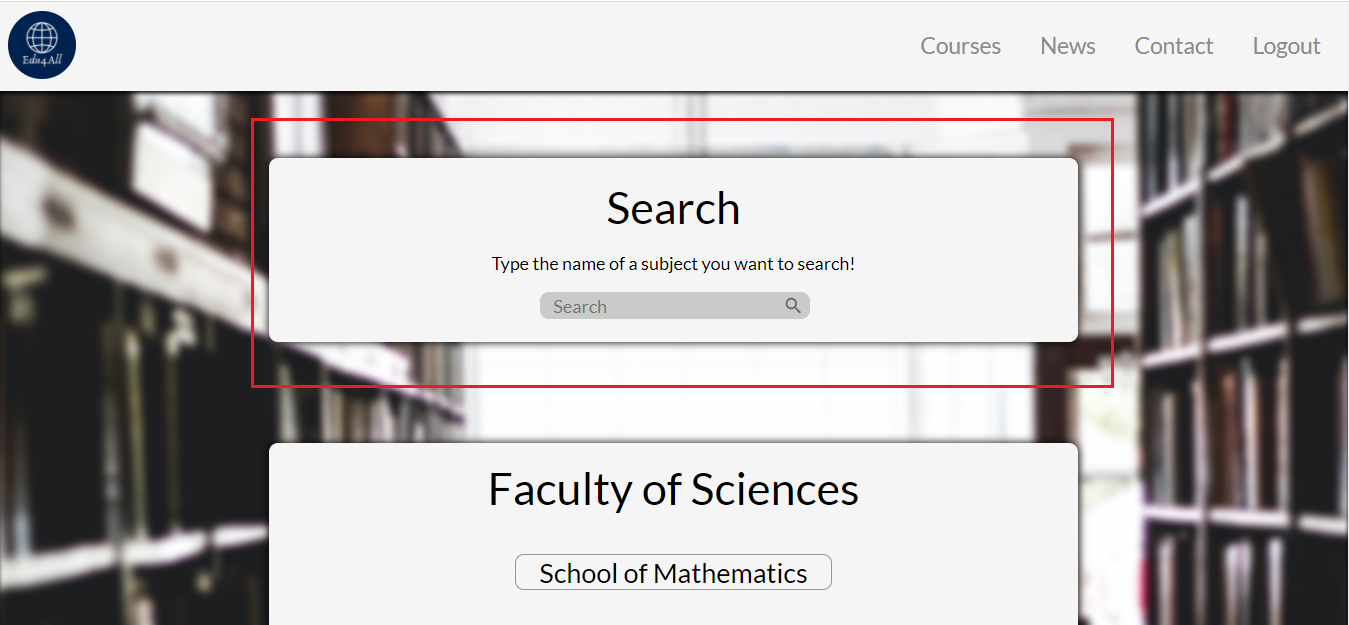
Το αρχικό Html αρχείο περιλαμβάνει τις μέρες της εβδομάδας , καθώς και τις ημερομηνίες (1 -30).Με χρήση της JavaScript( setCalendra.js) λαμβάνεται η τοπική ημερομηνία ,(μέσω της getDate() ) ενώ γίνονται οι απαραίτητες μετατροπές ώστε να προβληθούν σωστά οι μέρες που ανήκουν σε συγκεκριμένο μήνα . Τέλος με χρήση της innerHTML γίνεται η προβολή του μήνα και της ημερομηνίας στο div class=”date” . Για την πλοήγηση στους διάφορους μήνες του χρόνο γίνεται η χρήση έτοιμης υλοποίησης **(**[**https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.13.0/css/all.min.css**](https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.13.0/css/all.min.css)**)**.Για τον καθορισμό των διαστάσεων ( height , weight , margin , padding ) έγινε χρήση των μονάδων **Vh** και **Vw** ώστε να εξασφαλιστεί η responsive λειτουργία του.

* Πρέπει να τονιστεί ότι το calendar δεν υποστηρίζει αλλαγή ημερών παρά μόνο προβάλλει την τωρινή ημερομηνία.

**Μελλοντική Επέκταση**

Προσθήκη event holder στο ημερολόγιο , ενώ παράλληλα μπορεί να γίνει χρήση ενός API (πχ Wiktionary) ώστε να παρέχονται οι ορισμοί λέξεων τις οποίες μπορεί να ψάξει ο χρήστης.

**Courses\_Search.php**



Το Search-Bar είναι ενσωματωμένο στο Courses.php και δίνει την δυνατότητα στο χρήστη να αναζητήσει ένα συγκεκριμένο μάθημα. Με την εισαγωγή ενός ονόματος και το πάτημα του “Search Button” , μεταφερόμαστε στο Courses\_final.php , ενώ με $\_Post [‘search’] λαμβάνουμε την επιλογή του χρήστη. Εάν ο τρόπος πρόσβασης είναι σωστός (μέσω του Search Button ) και ο συγκεκριμένος τίτλος μαθήματος υπάρχει αποθηκευμένος στην db των Courses(προηγείται η σύνδεση μέσω του db.connect.php) τότε μεταφέρεται ο χρήστης στο Courses\_final.php περνώντας ως $\_GET[‘Title’] το μάθημα που αντιστοιχεί στο ‘search’ που έδωσε. Εάν το μάθημα δεν διατίθεται τότε εμφανίζει στο χρήστη το κατάλληλο μήνυμα και τον μεταφέρει στο Courses.php .

**Πειραματισμός**

Δοκίμασα να δημιουργήσω ένα πιο ανεκτικό search-bar , το οποίο δεν χρειάζεται ακριβώς τον τίτλο του μαθήματος( συμπεριλαμβανομένου των Uppercase και Lowercase περιπτώσεων) , ωστόσο το αποτέλεσμα δεν ήταν το επιθυμητό (προκύπταν μαθήματα τα οποία δεν είχαν απαραίτητα σχέση , παρά μόνο πλησίαζαν τα ονόματά τους) , οπότε επέλεξα να το διατηρήσω πιο απλό , με βάση τα τωρινά πεδία της βάσης.

**Μελλοντική Επέκταση**

Σε περίπτωση όπου δεν υπάρχει το μάθημα , να εμφανίζει μαθήματα παρόμοιου περιεχομένου , που ίσως να ενδιαφέρουν τον χρήστη.