ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ

Ανάπτυξη λογισμικού υποβοήθησης διδασκαλίας

Γεωργία Νεραντζάκη

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
Υπεύθυνος
Γεράσιμος Ραζής
Ακαδημαϊκός Υπότροφος

Λαμία,2024

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ

Ανάπτυξη λογισμικού υποβοήθησης διδασκαλίας

Νεραντζάκη Γεωργία

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων/σα
Γεράσιμος Ραζής
Ακαδημαϊκός Υπότροφος

Λαμία, 2024

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις [(1)], που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. *Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί* ***χωρίς να τα περικλείω σε εισαγωγικά*** *και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.*

2. *Δέχομαι ότι η αυτολεξεί* ***παράθεση χωρίς εισαγωγικά****, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.*

3. *Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια*

4. *Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.*

Ημερομηνία: 3/10/2024

Ο – Η Δηλ.

(Υπογραφή)

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Ανάπτυξη λογισμικού υποβοήθησης διδασκαλίας

Νεραντζάκη Γεωργία

Τριμελής Επιτροπή:

Γεράσιμος Ραζής, Ακαδημαϊκός υπότροφος

Ιωάννης Αναγνωστόπουλος, Καθηγητής

Κωνσταντίνος Δελήμπασης, Καθηγητής

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά θα ήθελα να ευχαριστήσω θερμά τον κύριο Γεράσιμο Ραζή για την υπομονή και την καθοδήγησή του καθ' όλη την διάρκεια της εκπόνησης της πτυχιακής εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω τον πατέρα μου για κάθε υποστήριξη που μου παρείχε μέχρι στιγμής.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

ΠΕΡΙΛΗΨΗ

Ο σκοπός της παρούσας πτυχιακής εργασίας είναι η ανάπτυξη ενός λογισμικού που αντιμετωπίζει τις υπάρχουσες προκλήσεις της δια ζώσης και εξ αποστάσεως εκπαιδευτικής διαδικασίας, αλλά κυρίως αποσκοπεί στην βελτίωσή της. Συγκεκριμένα, παρέχει ένα ανοιχτού κώδικα λογισμικό για φοιτητές και διδάσκοντες που είναι διαθέσιμο για όλα τα πανεπιστήμια και για όλες τις συσκευές (PC, κινητό τηλέφωνο, tablet), με σκοπό την βελτίωση τόσο της μάθησης όσο και της διδασκαλίας.

Η εφαρμογή προσφέρει πολλά φιλικά προς τον χρήστη εργαλεία, όπως το ανέβασμα/προβολή εγγράφων, εργασιών, βαθμολογιών, εργαλεία που ήδη υπάρχουν στα περισσότερα λογισμικά διαχείρισης εκπαιδευτικού περιεχομένου των πανεπιστημίων αλλά προστίθεται και μια πρωτότυπη λειτουργία που ενθαρρύνει τους φοιτητές να κάνουν, σε πραγματικό χρόνο, ανώνυμα ερωτήσεις, σχόλια και να αξιολογούν κατά πόσο κατανοητό είναι το περιεχόμενο της διάλεξης, επιτρέποντας ταυτόχρονα τον διδάσκοντα να σχηματίσει μια πλήρη εικόνα σχετικά τις δυσκολίες των φοιτητών ώστε να βελτιώσει ή να προσθέσει επιπλέον εκπαιδευτικό περιεχόμενο για να τους βοηθήσει να ξεπεράσουν τα μαθησιακά τους εμπόδια. Οι γλώσσες που χρησιμοποιήθηκαν είναι η Python για το server κομμάτι και η HTML για το client κομμάτι της εφαρμογής. Επιπλέον χρησιμοποιήθηκε frontend framework (Boostrap), backend framework (Flask) και template engine (Jinja).

ABSTRACT

This Thesis is focused to the development of an open source, accessible by anyone and on all devices, web application, with the aim to improvise and make easier both the learning and the teaching process. The app will provide many user-friendly tools like uploading/viewing documents, assignments, grades, tools in most university e-learning websites already exist, but there is also a new feature added that encourages the students to ask questions and to comment anonymously during the lecture assisting the teacher to better understanding students' "weaknesses" and therefore modify or adding extra content to help them overcome their learning obstacles. The app is suitable both for online and traditional education. The application was built with Python programming language for the server side and HTML mark-up language for the client side. Also, frameworks were used like Flask for the back end, Boostrap for the front end and template engine Jinja.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Περιεχόμενα

ΠΕΡΙΛΗΨΗ 7

ABSTRACT 7

1. ΕΙΣΑΓΩΓΗ 10

1.1 Διατύπωση του προβλήματος 10

1.2 Περιγραφή δυνατοτήτων της εφαρμογής 10

2. ΣΧΕΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ 11

3. ΤΕΧΝΟΛΟΓΙΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ 13

3.1 MVC 13

3.2 FRONTEND 14

3.2.1 HTML 14

3.2.2 BOOSTRAP 15

3.3 BACK-END 16

3.3.1 PYTHON 16

3.3.2 FLASK 16

3.3.3 JINJA 17

3.4 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ 18

3.4.1 Σχήμα Βάσης Δεδομένων 18

3.4.2 SQLite 26

4. RBAC 26

5. PROJECT SET UP ΚΑΙ ΔΟΜΗ 27

6. ΠΑΡΟΥΣΙΑΣΗ ΕΓΑΡΜΟΓΗΣ ΚΑΙ ΑΝΤΙΣΤΟΙΧΟΥ ΚΩΔΙΚΑ 30

6.1 ΣΕΛΙΔΑ ΣΥΝΔΕΣΗΣ/ΕΓΓΡΑΦΗΣ 30

6.2 ΔΙΕΠΑΦΗ ΔΙΔΑΣΚΟΝΤΑ 34

6.2.1 Αρχική Σελίδα Διδάσκοντα 34

6.2.2 Σελίδα δημιουργίας μαθήματος 35

6.2.3 Προβολή σελίδας λεπτομερειών/περιγραφής του μαθήματος 37

6.2.4 Σελίδα δημοσίευσης ανακοινώσεων μαθήματος 38

6.2.5 Σελίδα δημοσίευσης υλικού μαθήματος 41

6.2.6 Δημοσίευση εργασίας και προβολής λύσεων των φοιτητών για κάθε εργασία ... 43

6.2.7 Δημοσίευση βαθμών σε κάθε φοιτητή ξεχωριστά 47

6.2.8 Σελίδα δημιουργία διάλεξης 49

6.2.9 Εμφάνιση αποτελεσμάτων ερώτησης αξιολόγησης 51

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.10 Εμφάνιση στατιστικών δεδομένων δια δραστικού εργαλείου 55

6.3 ΔΙΕΠΑΦΗ ΦΟΙΤΗΤΗ..... 61

6.3.1 Αρχική σελίδα Φοιτητή 61

6.3.2 Εγγραφή σε μάθημα..... 63

6.3.3 Προβολή σελίδας ανακοινώσεων 65

6.3.4 Σελίδα προβολής υλικού μαθήματος 66

6.3.5 Προβολή σελίδας βαθμού 68

6.3.6 Προβολή εκφώνησης εργασίας και ανέβασμα αρχείου λύσης 69

6.3.7 Προβολή ερώτησης αξιολόγησης γνώσεων και δια δραστικού εργαλείου 72

6.4. ΑΛΛΑ ΤΜΗΜΑΤΑ ΚΩΔΙΚΑ 76

7. ΕΙΚΟΝΙΚΗ ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ 77

7.1 Παρουσίαση εικονικής διδασκαλίας και ανάλυση αποτελεσμάτων πρώτης ακαδημαϊκής χρονιάς 77

7.2 Αποτελέσματα εικονικής διδασκαλίας την δεύτερη ακαδημαϊκή χρονιά..... 80

7.3 Ανάλυση αποτελεσμάτων μελέτης περίπτωσης..... 82

8. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΙΘΑΝΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΕΙΣ 82

8.1 Συμπεράσματα 82

8.2 Πιθανές βελτιώσεις ή επεκτάσεις 83

Αναφορές..... 84

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

1. ΕΙΣΑΓΩΓΗ

Είναι γεγονός, ότι τις τελευταίες δεκαετίες η τεχνολογία έχει εξελιχθεί ραγδαία σε όλους του τομείς συμπεριλαμβανομένης και της εκπαίδευσης. Πλέον η χρήση των τεχνολογικών μέσων αποτελεί αναπόσπαστο μέρος της εκπαιδευτικής διαδικασίας. Οι φοιτητές και οι καθηγητές βασίζονται κυρίως σε λογισμικά διαχείρισης εκπαιδευτικού υλικού και αξιολόγησης της προόδου των φοιτητών, από τα οποία κάποια είναι αμφίδρομα διαδραστικά, ώστε να μπορούν καθηγητές και μαθητευόμενοι να αλληλοεπιδρούν μεταξύ τους, για να αντιμετωπιστούν οι δυσκολίες που προκύπτουν κατά τη διάρκεια των μαθημάτων. Όμως παραμένουν ορισμένα μειονεκτήματα των εφαρμογών που χρησιμοποιούνται, που αφορούν το κόστος, την προσβασιμότητα καθώς και την ανωνυμία, τα οποία επιλύονται με την ανάπτυξη του εν λόγω λογισμικού, προσφέροντας ταυτόχρονα αμφίδρομη διαδραστική αλληλεπίδραση σε πραγματικό χρόνο μεταξύ διδασκόντων και διδασκομένων.

1.1Διατύπωση του προβλήματος

Τα προβλήματα που παραμένουν έως σήμερα με τη χρήση των υφιστάμενων εφαρμογών, συνοψίζονται σε τρεις τομείς:

- **Κόστος:** Μέχρι στιγμής δεν έχει δημιουργηθεί κάποιο λογισμικό που να προσφέρει πλήρως δωρεάν διαδραστική αλληλεπίδραση μεταξύ διδάσκοντα και φοιτητών κατά την διάρκεια της παράδοσης του μαθήματος.
- **Προσβασιμότητα:** Πολλά εργαλεία εξαιρετικά για διαδραστική μάθηση, θέτουν περιορισμούς όσον αφορά την τοποθεσία από την οποία μπορούν να έχουν πρόσβαση οι φοιτητές και το διδακτικό προσωπικό.
- **Ανωνυμία:** Παρόλο που γενικά χρησιμοποιούνται εφαρμογές από τις οποίες μπορεί κάποιος να διατυπώσει ερωτήσεις και σχόλια σε πραγματικό χρόνο, δεν υπάρχει η δυνατότητα διατύπωσής τους χωρίς να αποκαλύπτεται η ταυτότητα κάποιου.

Η εν λόγω εφαρμογή επιλύει ταυτόχρονα όλα τα παραπάνω προβλήματα.

1.2 Περιγραφή δυνατοτήτων της εφαρμογής

Το λογισμικό προσφέρει τα εργαλεία που παρέχουν και άλλες εφαρμογές διαχείρισης εκπαιδευτικού περιεχομένου και επιπλέον παρέχονται οι παρακάτω δυνατότητες:

Για τον φοιτητή:

- Ο φοιτητής μπορεί να εκφραστεί ελεύθερα και ανώνυμα σε πραγματικό χρόνο, χωρίς κανένα ενδοιασμό προς τους συμφοιτητές που ταυτόχρονα παρακολουθούν το μάθημα.
- Ο φοιτητής μπορεί να ενημερώνει ανώνυμα συνεχώς τον διδάσκοντα κατά την διάρκεια του μαθήματος, αξιολογώντας το επίπεδο κατανόησης της τρέχουσας διάλεξης.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Για τον καθηγητή:

- Ο καθηγητής την ώρα της διάλεξης ενημερώνεται για το ποσοστό συμμετοχής των φοιτητών στο μάθημά του και το πόσο εύληπτο είναι αυτό από το ακροατήριό του.
- Ο καθηγητής σε πραγματικό χρόνο μπορεί να δει απορίες, σχόλια των φοιτητών και αν υπάρχει μεγάλο ποσοστό φοιτητών που εκφράζει την ίδια ή παρόμοια ερώτηση, έχει την δυνατότητα να εστιάσει στην επίλυση της χωρίς να χρειαστεί να διακόψει προσωρινά την διάλεξη.
- Επίσης, είναι εφικτό και μετά το πέρας της διάλεξης να ενημερωθεί σχετικά με τις ερωτήσεις και τα σχόλια των φοιτητών ώστε να προχωρήσει σε ανανέωση του εκπαιδευτικού περιεχομένου.
- Την ώρα του μαθήματος, ο διδάσκοντας έχει την δυνατότητα να παρακολουθεί τα στατιστικά της αξιολόγησης των φοιτητών, τα οποία σχετίζονται με την συμμετοχή τους, το ποσοστό των φοιτητών που δεν κατανόησαν το μάθημα καθώς και το επίπεδο κατανόησης του μέσου φοιτητή που παρακολουθεί.
- Ακόμα και σε περίπτωση που δεχθεί μη κατάλληλα σχόλια ζωντανά, καθώς οι φοιτητές διατηρούν την ανωνυμία τους, μπορεί να διακόψει την πρόσβαση στο εργαλείο για όσο χρονικό διάστημα επιθυμεί.

2.ΣΧΕΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ

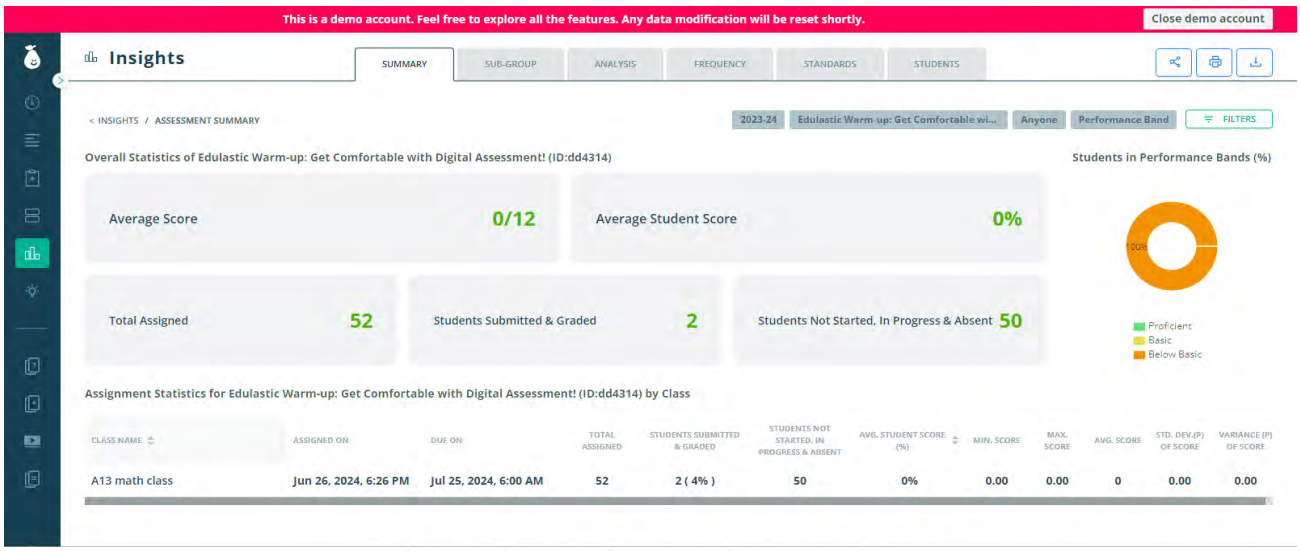
Τις τελευταίες δεκαετίες με την ραγδαία εξέλιξη της τεχνολογίας, ανακαλύπτονται συνεχώς και γίνονται προσπάθειες ικανοποίησης τόσο των αναγκών των μαθητευόμενων όσο και των διδασκόντων κατά τη διάρκεια της διδασκαλίας. Για αυτό κυκλοφορούν ολοένα και περισσότερες εφαρμογές που αποσκοπούν στην υποβοήθηση των εμπλεκόμενων στην εκπαιδευτική διαδικασία ώστε η διδασκαλία να γίνεται ευκολότερη και πιο αποτελεσματική. Κάποιες εφαρμογές που εκπληρώνουν αυτόν τον σκοπό είναι:

A) **Pear Assessment**

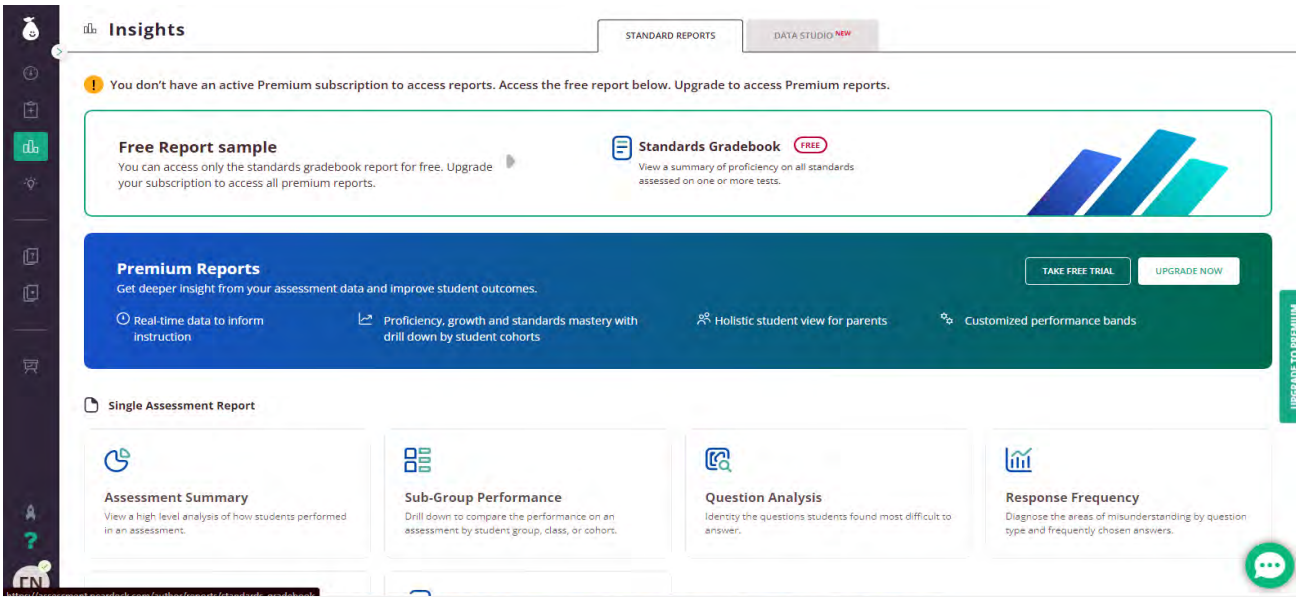
Το Pear Assessment είναι ένα βοηθητικό εργαλείο για αξιολόγηση μαθητών. Επιτρέπει στους δασκάλους να ενημερώνονται για την κατανόηση και την μάθηση των μαθητών, καθώς και να βελτιώνουν τον τρόπο διδασκαλίας έχοντας πρόσβαση σε σχετικά δεδομένα. Εκτός από αυτό, προσφέρει καθοδήγηση μετά από απαίτηση, ενισχύοντας και βελτιώνοντας τις αλληλεπιδράσεις μεταξύ δασκάλου και μαθητή. Όμως δεν είναι αποκλειστικά δωρεάν για όλους τους χρήστες καθώς παρόλο που υπάρχει μια δωρεάν έκδοση, παρέχονται περιορισμένες δυνατότητες στον διδάσκοντα. Για να αποκτήσει πρόσβαση σε περισσότερες λειτουργίες της εφαρμογής χρειάζεται συνδρομή από τον διδάσκοντα. Πιο αναλυτικά, με την δωρεάν έκδοση (Εικόνα 2) ο καθηγητής περιορίζεται μόνο στην δημιουργία εικονικών τάξεων, τεστ αξιολόγησης και εργασιών, ωστόσο για να αποκτήσει πρόσβαση σε στατιστικά προόδου φοιτητών ώστε να ενημερωθεί για την μάθηση τους και να βελτιώσει την διδασκαλία, απαιτείται συνδρομή από τον καθηγητή (Εικόνα 1).

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Σύνδεσμος: https://www.peardeck.com/products/pear-assessment



Εικόνα 1.Πεδίο προβολής στατιστικών (Premium έκδοση)



Εικόνα 2.Σελίδα προβολής στατιστικών (δωρεάν έκδοση)

B) **Wordwall**

Παρέχει κυρίως την δυνατότητα δημιουργίας customized δια δραστικών δραστηριοτήτων (π.χ κουίζ, παιχνίδια λέξεων κλπ). Επιπλέον ο διδάσκοντας μπορεί να παρακολουθεί την απόδοση των μαθητών μέσω dashboard. Παρόλο που προσφέρει τις παραπάνω δυνατότητες η εφαρμογή δεν σχετίζεται με την ενημέρωση του διδάσκοντα σχετικά με την κατανόηση του μαθήματος από τους φοιτητές , ούτε είναι δυνατή η προβολή των απορριών τους παρά μόνο εστιάζει στην προβολή αποτελεσμάτων των τεστ αξιολόγησης. Τέλος η πλατφόρμα παρέχει μια

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

δωρεάν έκδοση αλλά υπάρχει περιορισμός ως προς τον αριθμό των κουίζ που μπορεί να φτιάξει ο καθηγητής με περιορισμένα πρότυπα που μπορεί να χρησιμοποιήσει για να δημιουργήσει τα εν λόγω τεστ. Επομένως για να μπορέσει κάποιος να δημιουργήσει όσα τεστ επιθυμεί χρησιμοποιώντας περισσότερα πρότυπα θα πρέπει να καταβληθεί ένα μηνιαίο ποσό.

Σύνδεσμος:https://wordwall.net/

Γ) **Bright Minds**

Ξεχωρίζει ως μια δωρεάν δυναμική διαδραστική πλατφόρμα εκμάθησης. Οι δάσκαλοι μπορούν να προσαρμόσουν τα μαθήματα ώστε να ανταποκρίνονται στις διαφορετικές ανάγκες των μαθητών. Κύριο χαρακτηριστικό της εφαρμογής είναι τα κουίζ που διατηρούν την προσοχή και το ενδιαφέρον των μαθητευόμενων, δίνοντας τους την δυνατότητα να παρέχουν ανατροφοδότηση βοηθώντας τους δασκάλους να έχουν καλύτερη εικόνα σχετικά με την κατανόηση και την απόδοση των μαθητών. Επιπλέον ο καθηγητής μπορεί να ανεβάσει υλικό μαθήματος, ανακοινώσεις και εργασίες. Αλλά υπάρχουν μειονεκτήματα καθώς πρώτον δεν έχει δικαίωμα εγγραφής στην εφαρμογή οποιοσδήποτε από οποιαδήποτε χώρα, παρά μόνο όσοι διαμένουν στην Ινδία, δεύτερον είναι δυνατή η λήψη της εφαρμογής μόνο στη κινητή συσκευή και όχι στον υπολογιστή, και τέλος η εφαρμογή δεν επιτρέπει την αμφίδρομη αλληλεπίδραση μεταξύ των εμπλεκομένων κατά την παράδοση του μαθήματος [1] [2].

Σύνδεσμος:https://play.google.com/store/apps/details?id=co.iron.heoti&hl=en&gl=US&pli=1

3. ΤΕΧΝΟΛΟΓΙΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ

3.1 MVC

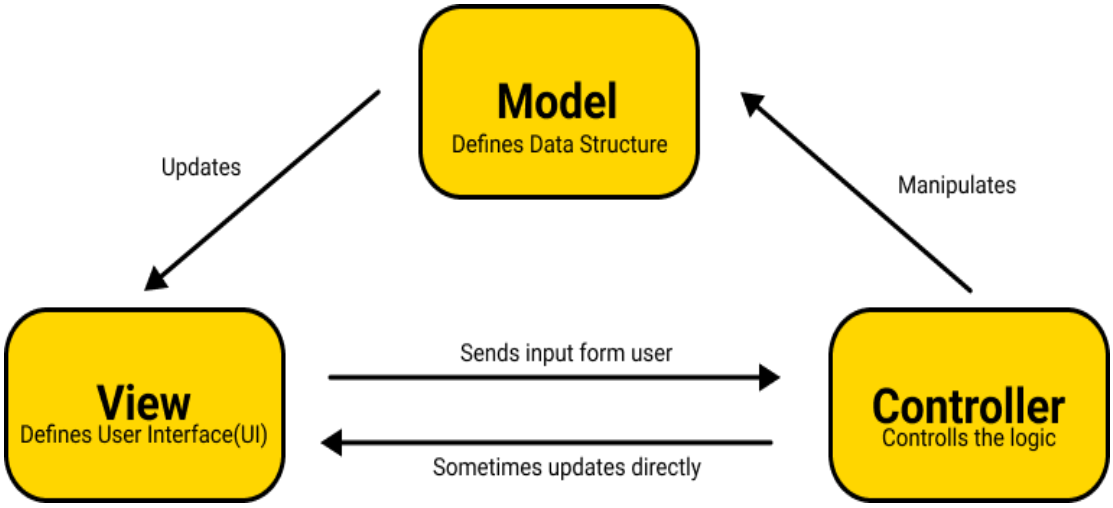
Το Model-view-controller (ή αλλιώς MVC) είναι ένα μοντέλο αρχιτεκτονικής λογισμικού το οποίο είναι χρήσιμο στη δημιουργία περιβαλλόντων αλληλεπίδρασης χρήστη. Το μοντέλο αυτό χωρίζει την εφαρμογή σε τρία διασυνδεδεμένα μέρη ώστε να διαχωριστεί η παρουσίαση της πληροφορίας στον χρήστη από την μορφή που έχει αποθηκευτεί στο σύστημα :

1)Το Model είναι υπεύθυνο για την ανάκτηση/αποθήκευση των δεδομένων στο σύστημα.

2)Το View είναι υπεύθυνο για την αναπαράσταση της πληροφορίας στον χρήστη (γραφική αναπαράσταση της εφαρμογής σε μια συγκεκριμένη μορφή).

3)Το Controller ο οποίος δέχεται την είσοδο του χρήστη και στέλνει εντολές στο αντικείμενο Model και στο View [3].

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43



Εικόνα 3. MVC Διάγραμμα

3.2 FRONTEND

3.2.1 HTML

Η HTML (HyperText Markup Language) είναι standard markup language για την κατασκευή ή σχεδίαση ιστοσελίδων. Εστιάζει κυρίως στην εμφάνιση και στην δομή του περιεχομένου. Με άλλα λόγια είναι ένα σύνολο ετικετών (tags) και ιδιοτήτων (attributes), τα οποία περιγράφουν τις ετικέτες που καθορίζουν την αναπαράσταση και το περιεχόμενο του εγγράφου. Συνήθως συνδυάζεται με CSS (Cascading Style Sheets) και Javascript (scripting language). Δεν είναι ακριβώς γλώσσα προγραμματισμού, αλλά γλώσσα περιγραφής ή αναπαράστασης των στοιχείων μιας ιστοσελίδας ή εφαρμογής. Όλα τα έγγραφα που περιέχουν HTML εντολές έχουν κατάληξη *.html* δηλαδή το όνομα τους είναι σε μορφή *filename.html*.

Ετικέτες: Οι *ετικέτες* αποτελούν την δομή και την μορφή του κειμένου της σελίδας. Ακόμη δίνουν πληροφορίες προς τον web browser για την σελίδα που πρόκειται να παρουσιαστεί, όπως ο τίτλος της σελίδας ή ο συγγραφέας της, κ.α. Οι ετικέτες είναι μέσα σε σύμβολα “< >” και η κάθε ετικέτα έχει ετικέτα αρχής (opening tag) και τέλους (closing tag).
Πχ.<label></label>. Όμως κάποιες δεν έχουν ετικέτες τέλους (self closing) πχ.<meta>. Τέλος οι ετικέτες έχουν και ιδιότητες που δίνουν επιπλέον στοιχεία ή εισάγουν ιδιότητες και κλάσεις από CSS αρχείο π.χ <li class="breadcrumb-item active" aria-current="page"> [4] [5].

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Εικόνα 4. Λογότυπο της HTML

Σύντομη ιστορία της HTML: Η πρώτη έκδοση της HTML έχει γραφεί από τον Tim Berners-Lee τη δεκαετία του 90. Παρόλο που από τότε κυκλοφόρησαν πολλές εκδόσεις της HTML η πιο ευρέως χρησιμοποιούμενη έκδοση την δεκαετία του 2000 ήταν η HTML 4.01. Σήμερα η ανάπτυξη των περισσότερων ιστοσελίδων και εφαρμογών βασίζεται στην HTML 5 (η πιο τελευταία έκδοση) [6].

3.2.2 BOOSTRAP

Το Bootstrap είναι μια συλλογή ανοιχτού κώδικα (front-end framework) για τη ανάπτυξη ιστοσελίδων και διαδικτυακών εφαρμογών. Περιέχει κώδικα HTML σε συνδυασμό με «έτοιμες» κλάσεις CSS για την αναπαράσταση κειμένου, κουμπιών πλοήγησης και άλλων στοιχείων του γραφικού περιβάλλοντος, καθώς και προαιρετικές επεκτάσεις JavaScript [7].

Πλεονεκτήματα χρήσης Boostrap:

- Ταχύτερη ανάπτυξη ιστοσελίδας: επειδή διαθέτει πολλά έτοιμα κομμάτια κώδικα, μπορεί να κάνει πιο γρήγορη την κατασκευή της σελίδας.
- Προηγμένα στοιχεία: Περιέχει elements με «στυλ» για κατασκευή dropdown menus, μπάρες πλοήγησης(navigation bars) , ειδοποιήσεις και πολλά άλλα.
- Εύκολη Χρήση: Διαθέτει εξαιρετική τεκμηρίωση, διευκολύνοντας την χρήση τόσο στους αρχάριους όσο και στους πιο προχωρημένους χρήστες [8].

Η εισαγωγή του Boostrap στο html αρχείο γίνεται ως εξής:

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
<!-- Bootstrap core CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-W
```

Εικόνα 5. Boostrap CDN

Στιγμιότυπα κώδικα όπου χρησιμοποιείται το Boostrap:

```
<p class="h5"><u>Grades</u></p>
<ul class="nav flex-column">
<li class="nav-item">
<div class="col-md-9">
<div class="jumbotron">
```

3.3 BACK-END

3.3.1 PYTHON

Η Python είναι γενικού σκοπού (general-purpose) και υψηλού επιπέδου (high-level), γλώσσα προγραμματισμού. Είναι μέρος των γλωσσών προστακτικού προγραμματισμού (*Imperative programming*) και υποστηρίζει τόσο το *procedural programming* όσο και το *object-oriented programming* προγραμματιστικό υπόδειγμα (*programming paradigm*). Αποτελεί δυναμική γλώσσα προγραμματισμού. Γράφτηκε από τον Guido van Rossum το 1989 και πρωτοκυκλοφόρησε το 1991.Είναι διάδοχος της γλώσσας προγραμματισμού ABC [9].
Πλεονεκτήματα Python:

- Έχει απλή σύνταξη που μιμείται την φυσική γλώσσα ώστε να γίνεται καλύτερη και ευκολότερη η κατανόηση της.
- Είναι ευέλικτη μπορεί για χρησιμοποιηθεί για ποικίλους σκοπούς από την κατασκευή ιστοσελίδων μέχρι και την μηχανική μάθηση.
- Είναι ανοιχτού κώδικα που σημαίνει ότι μπορεί να χρησιμοποιηθεί και για εμπορικούς σκοπούς.

3.3.2 FLASK

Το Flask είναι ένα web framework γραμμένο σε Python. Δεν περιέχει δικούς του μηχανισμούς από μόνο του π.χ επαλήθευση στοιχείων φόρμας αλλά, διαθέτει επεκτάσεις για επιπλέον λειτουργίες. Οι επεκτάσεις αφορούν object-relational mappers, επαλήθευση φόρμας, και για τεχνολογίες επιβεβαίωσης (authentication technologies) και διάφορα άλλα. Το συγκεκριμένο framework επιλέχθηκε για την ευελιξία, την απλότητα του καθώς είναι κατάλληλο για την ανάπτυξη απλών και μικρών εφαρμογών [10].

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

3.3.3 JINJA

«Το Jinja είναι template engine για τη γλώσσα προγραμματισμού Python. Δημιουργήθηκε από τον Armin Ronacher. Το Jinja είναι παρόμοιο με το Django που είναι επίσης template engine, αλλά παρέχει εντολές που μοιάζουν με εκείνες της Python. Είναι μια γλώσσα προτύπου που βασίζεται στην δημιουργία δυναμικού περιεχομένου της ιστοσελίδας.
Η Jinja δίνει την δυνατότητα προσαρμογής ετικετών, φίλτρων (για μορφοποίηση ή μετατροπή τιμών και δοκιμές (για αξιολόγηση συνθηκών). Σε αντίθεση με τη Django, η Jinja επιτρέπει στον template designer να καλεί συναρτήσεις με ορίσματα σε αντικείμενα. Το Jinja είναι η προεπιλεγμένη template engine του Flask » [11]. Είναι κατάλληλο για την δημιουργία δυναμικού περιεχομένου εφαρμογών.

Μερικές από τις λειτουργίες που παρέχει το Jinja περιλαμβάνουν:

- **Δομές ελέγχου**

  ```
  {%if courseteacher%}{%endif%}
  ```

  Οι δομές ελέγχου επιτρέπουν την εμφάνιση περιεχομένου που ικανοποιούν μια συνθήκη

- **For Βρόγχοι**

  ```
  {%for assignmentsolution in assignmentsolutions%}{%endfor%}
  ```

  Τα φίλτρα εμφανίζουν πληθώρα αποτελεσμάτων εφόσον δεν είναι κενή η λίστα εισόδου.

- **Κληρονομικότητα μεταξύ προτύπων**

  ```
  {%extends "teacherhome/layout.html"%}
  ```

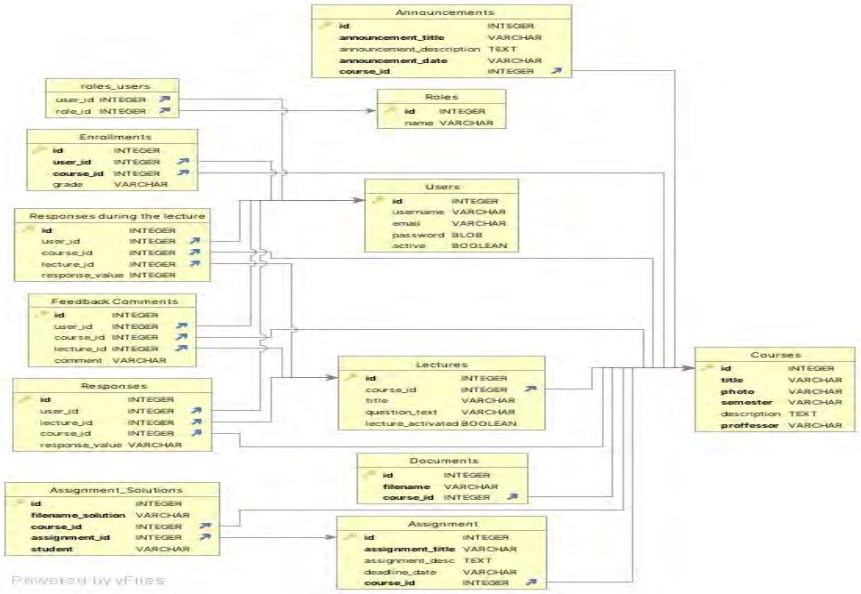
  Πολλά πρότυπα «κληρονομούν» από ένα «πατρικό» πρότυπο , για γρηγορότερη ανάπτυξη προτύπου. Με την «κληρονομικότητα» το θυγατρικό πρότυπο αποκτά πέρα από τα δικά του μοναδικά στοιχεία αλλά και εκείνα του «πατρικού» ή προτύπου βάσης, χωρίς να χρειαστεί να ξαναγραφεί όλος ο κώδικας από την αρχή.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

3.4 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

3.4.1 Σχήμα Βάσης Δεδομένων

Οι πίνακες της Βάσης Δεδομένων είναι οι εξής:



Εικόνα 6. Entity-Relationship Διάγραμμα (DbVisualizer).

Κάθε πίνακας, πέρα από τις ιδιότητες, αποτελείται από 2 είδη κλειδιών:

- **primary key ή πρωτεύον κλειδί**: είναι το μοναδικό αναγνωριστικό για κάθε μοντέλο ή πίνακα.
- **foreign key ή ξένο κλειδί**: Συνδέει ή συσχετίζει τα δεδομένα ενός πίνακα με ενός άλλου (Κυρίως συσχετίζει primary keys άλλων πινάκων ή μοντέλων).

USERS
id - πρωτεύον κλειδί
username - Το όνομα χρήστη
email - Το email χρήστη. Όλα τα email έχουν κατάληξη ‘@elearningweb.app’
password - Ο κωδικός πρόσβασης
active - Είναι τύπου Boolean, δηλώνει αν ο χρήστης είναι ενεργός ή όχι

ROLES
id - πρωτεύον κλειδί
name - Το όνομα του ρόλου

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

ROLES_USERS
user_id - ξένο κλειδί (το αναγνωριστικό του χρήστη, πρωτεύον κλειδί πίνακα Users)
role_id - ξένο κλειδί (το αναγνωριστικό του ρόλου, πρωτεύον κλειδί πίνακα Roles)

COURSES
id - πρωτεύον κλειδί
title - Ο τίλος του μαθήματος (σε μορφή συμβολοσειράς)
photo - Φωτογραφία μαθήματος (σε μορφή συμβολοσειράς, εισάγεται σαν path στην φόρμα δημιουργίας μαθήματος)
semester - Το εξάμηνο του μαθήματος(συμβολοσειρά)
description - Μικρή περιγραφή σχετικά με το μάθημα(συμβολοσειρά)
professor - Το όνομα του καθηγητή που διδάσκει το μάθημα (συμβολοσειρά)

ENROLLMENTS
id - πρωτεύον κλειδί
user_id - ξένο κλειδί (το αναγνωριστικό του χρήστη, πρωτεύον κλειδί πίνακα Users)
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)
grade - Ο βαθμός του φοιτητή στο συγκεκριμένο μάθημα που έχει εγγραφεί (συμβολοσειρά)

ANNOUNCEMENTS
id - πρωτεύον κλειδί
announcement title - Τίτλος ανακοίνωσης (συμβολοσειρά)
announcements description - Περιγραφή ανακοίνωσης (σε μορφή κειμένου)
announcement date - Ημερομηνία που δημοσιεύτηκε η ανακοίνωση (συμβολοσειρά)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

DOCUMENT
id - πρωτεύον κλειδί
filename - Όνομα αρχείου (συμβολοσειρά)
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)

ASSIGNMENT
id - πρωτεύον κλειδί
assignment title - Τίτλος εργασίας (συμβολοσειρά)
assignment description - Περιγραφή εργασίας (σε μορφή κειμένου)
deadline date - Ημερομηνία λήξης προθεσμίας παράδοσης (συμβολοσειρά)

ASSIGNMENT_SOLUTIONS
id - πρωτεύον κλειδί
solution_filename - Όνομα αρχείου που περιέχει την λύση (συμβολοσειρά)
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)
assignment_id - ξένο κλειδί (το αναγνωριστικό της εργασίας, πρωτεύον κλειδί πίνακα Assignment)
student - Όνομα φοιτητή που παρέδωσε την εργασία(συμβολοσειρά)

LECTURES
id - πρωτεύον κλειδί
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)
title - Τίτλος διάλεξης (συμβολοσειρά)
question_text - Ερώτηση αξιολόγησης που σχετίζεται με την διάλεξη
lecture_activated - Τύπου Boolean, είναι κυρίως για την ενεργοποίηση/απενεργοποίηση φόρμας

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

RESPONSES
id - πρωτεύον κλειδί
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)
lecture_id - ξένο κλειδί (το αναγνωριστικό της διάλεξης, πρωτεύον κλειδί πίνακα Lectures)
response_value - Τιμή της απάντησης στην φόρμα αξιολόγησης μετά τη παράδοση μαθήματος (συμβολοσειρά)

RESPONSES DURING THE LECTURES
id - πρωτεύον κλειδί
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)
lecture_id - ξένο κλειδί (το αναγνωριστικό της διάλεξης, πρωτεύον κλειδί πίνακα Lectures)
user_id - ξένο κλειδί (το αναγνωριστικό του χρήστη, πρωτεύον κλειδί πίνακα Users)
response_value - Τιμή της απάντησης στην live form κατά τη διάρκεια της διάλεξης (συμβολοσειρά)

FEEDBACK COMMENTS
id - πρωτεύον κλειδί
user_id - ξένο κλειδί (το αναγνωριστικό του χρήστη, πρωτεύον κλειδί πίνακα Users)
course_id - ξένο κλειδί (το αναγνωριστικό του μαθήματος, πρωτεύον κλειδί πίνακα Courses)
lecture_id - ξένο κλειδί (το αναγνωριστικό της διάλεξης, πρωτεύον κλειδί πίνακα Lectures)
comment - Σχόλιο/ερώτηση/απορία σχετικά με το περιεχόμενο της διάλεξης, υποβάλλεται από τους φοιτητές «ζωντανά» την ώρα της παράδοσης (συμβολοσειρά)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Οι παραπάνω πίνακες έχουν οριστεί σε ξεχωριστά αρχεία με την ονομασία models.py τα οποία για καλύτερη οργάνωση δημιουργήθηκαν σε ξεχωριστούς φακέλους:

- **authentication/models.py:** Στο αρχείο αυτό έχουν οριστεί οι πίνακες που σχετίζονται με την αυθεντικοποίηση του χρήστη (Εικόνα 7).
- **teacherhome/models.py:** Στο αρχείο αυτό (Εικόνες 9-13) έχουν οριστεί οι πίνακες που σχετίζονται με την εκπαιδευτική διαδικασία γενικότερα (πίνακες που αφορούν διαλέξεις, μαθήματα, εγγραφές).

Παρακάτω έχουν προστεθεί στιγμιότυπα που έχουν οριστεί τα μοντέλα της Βάσης Δεδομένων:

```
# -*- encoding: utf-8 -*-

#from flask_login import UserMixin
from app.database import db, login_manager
from app.authentication.util import hash_pass
from flask_security import UserMixin, RoleMixin

from app.teacherhome.models import Course

roles_users = db.Table('roles_users',
        db.Column('user_id', db.Integer(), db.ForeignKey('Users.id')),
        db.Column('role_id', db.Integer(), db.ForeignKey('Roles.id')))

class Users(db.Model, UserMixin):

    __tablename__ = 'Users'

    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(64), unique=True)
    email = db.Column(db.String(64), unique=True)
    password = db.Column(db.LargeBinary)
    active = db.Column(db.Boolean())
    roles = db.relationship('Role', secondary=roles_users, backref='roled')
    courses = db.relationship('Enrollment', back_populates='user')
    user_response = db.relationship('Responses',back_populates ='user')
    user_responses_inlecture = db.relationship('Lectureresponses',back_populates ='user')


    def __init__(self, **kwargs):
        roles = kwargs.pop('roles', [])
        super(Users, self).__init__(**kwargs)
        for role in roles:
            self.roles.append(role)
        for property, value in kwargs.items():
            # depending on whether value is an iterable or not, we must
            # unpack it's value (when **kwargs is request.form, some values
            # will be a 1-element list)
            if hasattr(value, '__iter__') and not isinstance(value, str):
        # will be a 1-element list)
        if hasattr(value, '__iter__') and not isinstance(value, str):

            value = value[0]

        if property == 'password':
            value = hash_pass(value)

        setattr(self, property, value)

def __repr__(self):
    return str(self.username)

create table in database for storing roles
class Role(db.Model, RoleMixin):
    __tablename__ = 'Roles'
    id = db.Column(db.Integer(), primary_key=True)
    name = db.Column(db.String(80), unique=True)


create_test_users(app):
        if Role.query.count() == 0:
            test_roles = [
                Role(name='student'),
                Role(name='teacher'),
                Role(name='admin')
            ]
            for role in test_roles:
                db.session.add(role)
                app.logger.info('%s role created successfully',role.name)
            db.session.commit()

        if Users.query.count() == 0:
            admin_role = Role.query.filter_by(name='admin').first()
            student_role = Role.query.filter_by(name='student').first()
            teacher_role = Role.query.filter_by(name='teacher').first()
            # Add test users when the database is empty
            test_users = [
                Users(username='admin', email='admin@elearningweb.app',password='1234',active = True, roles=[admin_role]),
                Users(username='teacher', email='teacher@elearningweb.app',password='1234',active = True, roles=[teacher_rol
            # Add test users when the database is empty
            test_users = [
                Users(username='admin', email='admin@elearningweb.app',password='1234',active = True, roles=[admin_role]),
                Users(username='teacher', email='teacher@elearningweb.app',password='1234',active = True, roles=[teacher_role]),
                Users(username='student', email='student@elearningweb.app',password='1234',active = True, roles=[student_role])

                # Add more test users as needed
            ]
            for user in test_users:
                db.session.add(user)
                app.logger.info('%s created successfully', user.username)

            db.session.commit()


@login_manager.user_loader
def user_loader(id):
    return Users.query.filter_by(id=id).first()


@login_manager.request_loader
def request_loader(request):
    username = request.form.get('username')
    user = Users.query.filter_by(username=username).first()
    return user if user else None
```

Εικόνα 7.authentication/models.py.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
# app/models.py
from app.database import db
#from app.authentication.models import Users
from datetime import datetime
import time


class Course(db.Model):

    __tablename__ = 'Courses'
    id = db.Column(db.Integer, primary_key=True)
    title = db.Column(db.String(100), nullable=False)
    photo = db.Column(db.String(100), nullable=False)
    semester = db.Column(db.String(50), nullable=False)
    description = db.Column(db.Text, nullable=True)
    proffessor = db.Column(db.String(100), nullable=False)
    enrollments = db.relationship('Enrollment', back_populates='course')
    announcement = db.relationship('Announcement',back_populates ='course_announcement')
    document = db.relationship('Document',back_populates ='course_doc')
    assignment = db.relationship('Assignment',back_populates ='course_assignment')
    assignment_solution = db.relationship('AssignmentSolution',back_populates ='solution_for_course')
    lecture = db.relationship('Lecture',back_populates ='lecture_course')
    response_for_course = db.relationship('Responses',back_populates ='response_course')
    response_for_lecture = db.relationship('Lectureresponses',back_populates ='response_lecture')



    def __init__(self, **kwargs):
        for property, value in kwargs.items():
            # depending on whether value is an iterable or not, we must
            # unpack it's value (when **kwargs is request.form, some values
            # will be a 1-element list)
            if hasattr(value, '__iter__') and not isinstance(value, str):
                # the ,= unpack of a singleton fails PEP8 (travis flake8 test)
                value = value[0]

            setattr(self, property, value)
```

Εικόνα 8.teacherhome/models.py

```python
    def __init__(self, **kwargs):
        for property, value in kwargs.items():
            # depending on whether value is an iterable or not, we must
            # unpack it's value (when **kwargs is request.form, some values
            # will be a 1-element list)
            if hasattr(value, '__iter__') and not isinstance(value, str):
                # the ,= unpack of a singleton fails PEP8 (travis flake8 test)
                value = value[0]

            setattr(self, property, value)

    def __repr__(self):
        return str(self.title)


def create_test_courses(app):
    if Course.query.count() == 0:
        test_courses = [
          Course(title = 'Physics', photo = '/media/images/course-image.jpg', semester = '2024B' , description = 'Introduction',proffessor  = 'teacher' )
        ]
        for course in test_courses:
            db.session.add(course)
            app.logger.info('%s course created successfully',course.title)
        db.session.commit()

class Enrollment(db.Model):
    __tablename__ = 'Enrollments'
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('Users.id'), nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'), nullable=False)
    grade  = db.Column(db.String(2), nullable=True)

    user = db.relationship('Users', back_populates='courses')
    course = db.relationship('Course', back_populates='enrollments')
```

Εικόνα 9.teacherhome/models.py(συνέχεια)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
class Assignment(db.Model):
    __tablename__ = 'Assignment'
    id = db.Column(db.Integer, primary_key=True)
    assignment_title = db.Column(db.String(30),nullable=False)
    assignment_desc = db.Column(db.Text, nullable=True)
    deadline_date =  db.Column(db.String(30),nullable=True)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'), nullable=False)
    course_assignment = db.relationship('Course',back_populates ='assignment')
    student_solution   = db.relationship('AssignmentSolution',back_populates ='solution_for_assignment', cascade= 'all', passive_deletes = True)
    def __repr__(self):
        return str(self.assignment_title)


class AssignmentSolution(db.Model):
    __tablename__ = 'Assignment_Solutions'
    id = db.Column(db.Integer, primary_key=True)
    filename_solution = db.Column(db.String(30),nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'), nullable=False)
    assignment_id = db.Column(db.Integer, db.ForeignKey('Assignment.id', ondelete='CASCADE'), nullable=False)
    student =  db.Column(db.String(30),nullable=False)

    solution_for_course = db.relationship('Course',back_populates ='assignment_solution')
    solution_for_assignment= db.relationship('Assignment',back_populates ='student_solution')

    def __repr__(self):
        return str(self.filename)


class Lecture(db.Model):
    __tablename__ = 'Lectures'
    id = db.Column(db.Integer, primary_key=True)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'))
    title = db.Column(db.String(40),nullable=True)
    question_text = db.Column(db.String(255))
    lecture_activated = db.Column(db.Boolean(),default = False)
    lecture_course = db.relationship('Course',back_populates ='lecture')
    lecture_response = db.relationship('Responses',back_populates ='response_for_lecture')
def __repr__(self):
```

Εικόνα 10.teacherhome/models.py(συνέχεια)

```python
class Enrollment(db.Model):
    __tablename__ = 'Enrollments'
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('Users.id'), nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'), nullable=False)
    grade  = db.Column(db.String(2), nullable=True)

    user = db.relationship('Users', back_populates='courses')
    course = db.relationship('Course', back_populates='enrollments')


class Announcement(db.Model):
    __tablename__ = 'Announcements'
    id = db.Column(db.Integer, primary_key=True)
    announcement_title = db.Column(db.String(100), nullable=False)
    announcement_description = db.Column(db.Text, nullable=True)
    announcement_date = db.Column(db.String(20), nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'), nullable=False)


    course_announcement = db.relationship('Course',back_populates ='announcement')

    def __repr__(self):
        return str(self.announcement_title)


class Document(db.Model):
    __tablename__ = 'Documents'
    id = db.Column(db.Integer, primary_key=True)
    filename = db.Column(db.String(30),nullable=False)
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'), nullable=False)

    course_doc = db.relationship('Course',back_populates ='document')

    def __repr__(self):
        return str(self.filename)
```

Εικόνα 11.teacherhome/models.py(συνέχεια)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
    lecture_response = db.relationship('Responses',back_populates ='response_for_lecture')
def __repr__(self):
        return str(self.title)



class Responses(db.Model):
    __tablename__ = 'Responses'
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('Users.id'))
    lecture_id = db.Column(db.Integer, db.ForeignKey('Lectures.id'))
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'))
    response_value = db.Column(db.String(40),nullable=True)
    response_course = db.relationship('Course',back_populates ='response_for_course')
    user = db.relationship('Users',back_populates ='user_response')
    response_for_lecture = db.relationship('Lecture',back_populates ='lecture_response')

    # def __init__(self, user_id, lecture_id, course_id, response_value):
    #     self.user_id = user_id
    #     self.lecture_id = lecture_id
    #     self.course_id = course_id
    #     self.response_value = response_value


    def __repr__(self):
        return str(self.response_value)



class Lectureresponses(db.Model):
    __tablename__ = 'Responses during the lecture'
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('Users.id'))
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'))
    lecture_id = db.Column(db.Integer, db.ForeignKey('Lectures.id'))
    response_value = db.Column(db.Integer)
```

Εικόνα 12.teacherhome/models.py(συνέχεια)

```
    response_course = db.relationship('Course',back_populates ='response_for_course')
    user = db.relationship('Users',back_populates ='user_response')
    response_for_lecture = db.relationship('Lecture',back_populates ='lecture_response')

    # def __init__(self, user_id, lecture_id, course_id, response_value):
    #     self.user_id = user_id
    #     self.lecture_id = lecture_id
    #     self.course_id = course_id
    #     self.response_value = response_value


    def __repr__(self):
        return str(self.response_value)



class Lectureresponses(db.Model):
    __tablename__ = 'Responses during the lecture'
    id = db.Column(db.Integer, primary_key=True)
    user_id = db.Column(db.Integer, db.ForeignKey('Users.id'))
    course_id = db.Column(db.Integer, db.ForeignKey('Courses.id'))
    lecture_id = db.Column(db.Integer, db.ForeignKey('Lectures.id'))
    response_value = db.Column(db.Integer)
    response_lecture = db.relationship('Course',back_populates ='response_for_lecture')
    user = db.relationship('Users',back_populates ='user_responses_inlecture')

    # def __init__(self, user_id, lecture_id, course_id, response_value):
    #     self.user_id = user_id
    #     self.lecture_id = lecture_id
    #     self.course_id = course_id
    #     self.response_value = response_value


    def __repr__(self):
        return str(self.response_value)
```

Εικόνα 13.teacherhome/models.py(συνέχεια)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

3.4.2 SQLite

Το SQLite είναι μια μηχανή Βάσης Δεδομένων γραμμένη στη γλώσσα προγραμματισμού C. Είναι η πιο ευρέως αναπτυγμένη μηχανή Βάσης Δεδομένων, καθώς χρησιμοποιείται από πολλά προγράμματα περιήγησης ιστού, λειτουργικά συστήματα, κινητά τηλέφωνα, εφαρμογές κλπ. Το SQLite είναι επίσης συνδεδεμένη με πολλές γλώσσες προγραμματισμού. Είναι μια βιβλιοθήκη που οι προγραμματιστές ενσωματώνουν στις εφαρμογές τους. Επιπρόσθετα, ανήκει στις ενσωματωμένες Βάσεις Δεδομένων που περιλαμβάνουν τις Σχεσιακές Βάσεις Δεδομένων (RDBMS, Relational Database Management System). Με την Σχεσιακή Βάση Δεδομένων μπορούμε να οργανώσουμε τα δεδομένα σε γραμμές και στήλες πινάκων, οι οποίοι σχετίζονται μεταξύ τους. Οι πίνακες ορίζονται στον κώδικα ως μοντέλα (models) με τις ιδιότητες τους να αποτελούν τις στήλες τους [12].

4. RBAC

Το RBAC (γνωστό ως Role Based Access Control) είναι τρόπος ανάθεσης δικαιωμάτων πρόσβασης και ενεργειών ανάλογα με τον ρόλο ενός ατόμου που κατέχει σε μια εφαρμογή. Οι κάτοχοι του ίδιου ρόλου έχουν τα ίδια δικαιώματα πρόσβασης και μπορούν να εκτελέσουν τις ίδιες ενέργειες σε σχέση με όσους έχουν διαφορετικούς ρόλους [13]. Στην συγκεκριμένη εφαρμογή οι ρόλοι είναι τρεις: Φοιτητής (τιμή 1 στην βάση δεδομένων), διδάσκοντας (τιμή 2 στην βάση δεδομένων) και διαχειριστής (τιμή 3 στην βάση δεδομένων, να σημειωθεί ότι ο συγκεκριμένος ρόλος δεν έχει αναπτυχθεί περισσότερο στα πλαίσια της εργασίας). Οι φοιτητές έχουν διαφορετικά δικαιώματα πρόσβασης και μπορούν να εκτελούν διαφορετικές ενέργειες σε σχέση με τους καθηγητές και τον διαχειριστή.
Πλεονεκτήματα υλοποίησης RBAC:

- **Μεγαλύτερη ασφάλεια δεδομένων.** Το RBAC συνεισφέρει στην προστασία των δεδομένων από διάφορες παραβιάσεις (Data Breaches) καθώς επιτρέπει στους χρήστες με συγκεκριμένους ρόλους να έχουν πρόσβαση σε συγκεκριμένα δεδομένα και να ενεργούν αναλόγως.
- **Βελτιστοποιημένες λειτουργίες.** Οι παραδοσιακές μέθοδοι πρόσβασης που περιλαμβάνουν την ανάθεση δικαιωμάτων μπορεί να είναι ευαίσθητες σε σφάλματα. Το RBAC βελτιώνει τις λειτουργίες με την ανάθεση ρόλων αντί ατομικών δικαιωμάτων εξασφαλίζοντας με αυτόν τον τρόπο συνέπεια.
- **Βελτιώνεται η συμμόρφωση.** Με το RBAC,γίνεται πιο εύκολη η αντιμετώπιση συνηθισμένων απαιτήσεων. Ελέγχεται η προστασία δεδομένων, η ιδιωτικότητα, και η προσβασιμότητα [14] [13] [15].

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

5. PROJECT SET UP ΚΑΙ ΔΟΜΗ

Δημιουργία virtual environment με την εντολή:

python -m venv env

Το env είναι ένα virtual environment που περιέχει όλες τις βιβλιοθήκες, scripts ώστε να τρέξει η εφαρμογή. Δηλαδή περιέχει όλες τις εξαρτήσεις της εφαρμογής σε ένα τοπικό περιβάλλον. Η εγκατάσταση των flask extensions στο virtual environment γίνεται μέσω της εντολής ***pip install <extension name>*** (Το pip είναι σύστημα διαχείρισης πακέτων).

Η ενεργοποίηση του virtual environment γίνεται με την παρακάτω εντολή:

.\env\Scripts\activate

Οργάνωση του πρότζεκτ:

```
+---elearningwebapp
|   record.log
|   requirements.txt
|   run.py
|
|
+---env
+---app
|   |   config.py
|   |   database.py
|   |   db.sqlite3
|   |   __init__.py
|   |
|   +---authentication
|   |   |   forms.py
|   |   |   models.py
|   |   |   routes.py
|   |   |   util.py
|   |   |   __init__.py
|   |
|   +---home
|   |   |   forms.py
|   |   |   routes.py
|   |   |   __init__.py
|   |
|   +---media
|   |   \---images
|   |
|   +---teacherhome
|   |   |   forms.py
|   |   |   models.py
|   |   |   routes.py
|   |   |   __init__.py
```

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
|   |
|   +---templates
|   |   +---accounts
|   |   |       layout.html
|   |   |       login.html
|   |   |       register.html
|   |   |
|   |   +---home
|   |   |       allcourses.html
|   |   |       announcements.html
|   |   |       assignments.html
|   |   |       coursedetails.html
|   |   |       documents.html
|   |   |       grades.html
|   |   |       index.html
|   |   |       layout.html
|   |   |       likertform.html
|   |   |       liveform.html
|   |   |       liveformlectures
|   |   |       liveformlectures.html
|   |   |       questionaire.html
|   |   |
|   |   +---includes
|   |   |       _navbar.html
|   |   |       _navbarTeacher.html
|   |   |
|   |   \---teacherhome
|   |           announcementsteacher.html
|   |           assignmentsolution
|   |           assignmentsolution.html
|   |           assignmentsteacher.html
|   |           courseteacher.html
|   |           createcourse.html
|   |           documentsteacher.html
|   |           gradesteacher.html
|   |           layout.html
|   |           lecturesteacher.html
|   |           liveformlectures
|   |           liveformlectures.html
|   |           liveformresponses.html
|   |           questionaireresponses.html
|   |           questionaireteacher.html
|   |           teacherindex.html
```

- app: περιέχει το πρότζεκτ της εφαρμογής.
- __init__.py: ένα αρχείο ειδικό που σε αυτό οφείλεται η σωστή λειτουργία των packages που γίνονται import. Επιπλέον στο

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

αρχείο αυτό δημιουργείται οντότητα του Flask app (Flask application instance).

- media: φάκελος που αφορά διάφορα έγγραφα (φωτογραφίες, docx, pdf κλπ) που ανεβαίνουν μέσω της εφαρμογής από τους χρήστες.
- requirements.txt: αρχείο που περιλαμβάνει την λίστα των packages που απαιτούνται για την εφαρμογή.
- record.log: αρχείο που σχετίζεται με όσες αλλαγές συμβαίνουν στο πρόγραμμα και στον κώδικα.
- run.py: Ο σκοπός που δημιουργήθηκε το αρχείο είναι να «τρέξει» την εφαρμογή.
- templates: φάκελος που περιλαμβάνει όλα τα html αρχεία για την απεικόνιση των σελίδων της εφαρμογής.
- templates/teacherhome: κατάλογος που αφορά τις σελίδες στις οποίες θα έχει πρόσβαση ο διδάσκοντας.
- templates/home: κατάλογος που αφορά τις σελίδες στις οποίες θα έχει πρόσβαση ο φοιτητής.
- templates/accounts: σχετίζεται με την απεικόνιση σελίδων σύνδεσης στην εφαρμογή καθώς και την εγγραφή σε αυτή.
- templates/include: φάκελος που περιέχει τα αρχεία με τον κώδικα για templates που αναπαριστούν navigation bars (τα οποία εμφανίζονται σε όλες τις σελίδες).
- teacherhome/__init__.py: Αρχείο στο οποίο ορίζεται το blueprint για τον καθηγητή. Τα blueprints είναι application components τα οποία απλοποιούν και οργανώνουν καλύτερα την εφαρμογή.
- teacherhome/forms.py: Αρχείο στο οποίο έχουν οριστεί και έχουν δημιουργηθεί όλες οι φόρμες (δημοσίευσης ανακοίνωσης, εγγράφου, εκφώνησης της εργασίας και δημιουργίας διάλεξης και βαθμού για κάθε εγεγραμμένο στο μάθημα φοιτητή ξεχωριστά).
- teacherhome/models.py: Σε εκείνο το αρχείο υπάρχουν τα μοντέλα της βάσης δεδομένων, δηλαδή στην ουσία έχουν οριστεί οι πίνακες της βάσης δεδομένων και οι στήλες τους. Οι πίνακες σχετίζονται με δεδομένα που αφορούν μαθήματα ,εγγραφές φοιτητών σε αυτά ,τις διαλέξεις και το περιεχόμενο(έγγραφα, ανακοινώσεις, εργασίες) των μαθημάτων. Τα μοντέλα μπορούν να χρησιμοποιηθούν και σε άλλα αρχεία κάνοντας import το μοντέλο που θέλουμε να χρησιμοποιήσουμε. Π.χ `from app.teacherhome.models import Course, Announcement`
- teacherhome/routes.py: Έχουν οριστεί τα URL και τις αντίστοιχες λειτουργίες στις οποίες μπορεί να έχει πρόσβαση ο διδάσκοντας.
- home/__init__.py: Ορίζει το blueprint για τον φοιτητή.
- home /forms.py: Εδώ έχουν οριστεί και έχουν δημιουργηθεί όλες οι φόρμες (υποβολής λύσης μιας εργασίας, απάντησης σε ερωτήσεις αξιολόγησης που αφορούν τη διάλεξη, εγγραφής και απεγγραφής σε μάθημα)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

- home /routes.py: Έχουν οριστεί τα URL και τις αντίστοιχες λειτουργίες στις οποίες μπορεί να έχει πρόσβαση ο φοιτητής.
- authentication/__init__.py: Αρχικοποιεί το blueprint για την αυθεντικοποίηση του χρήστη.
- authentication/forms.py: Εδώ έχουν οριστεί και έχουν δημιουργηθεί οι φόρμες εγγραφής και σύνδεσης.
- authentication/routes.py: Έχουν οριστεί τα URL και τις αντίστοιχες λειτουργίες για την αυθεντικοποίηση του χρήστη.
- authentication/models.py: Σε εκείνο το αρχείο υπάρχουν τα μοντέλα των χρηστών και των ρόλων τους.
- authentication/util.py: Περιλαμβάνει συναρτήσεις επαλήθευσης και κατακερματισμού κωδικού.

6. ΠΑΡΟΥΣΙΑΣΗ ΕΓΑΡΜΟΓΗΣ ΚΑΙ ΑΝΤΙΣΤΟΙΧΟΥ ΚΩΔΙΚΑ

Στην ενότητα αυτή έχουν προστεθεί στιγμιότυπα από το γραφικό περιβάλλον της εφαρμογής και τα αντίστοιχα τμήματα κώδικα. Ο κώδικές σχετίζονται με την αναπαράσταση της εφαρμογής και το backend κομμάτι της ανάπτυξης του λογισμικού. Σημαντικό να σημειωθεί ότι το λογισμικό μετά την σελίδα σύνδεσης και εγγραφής, διαχωρίζεται σε διεπαφή διδάσκοντα και φοιτητή όπου ο κάθε χρήστης έχει πρόσβαση και εκτελεί τις ανάλογες ενέργειες ανάλογα με τον ρόλο του, όπως φαίνεται στον κάτωθι πίνακα:

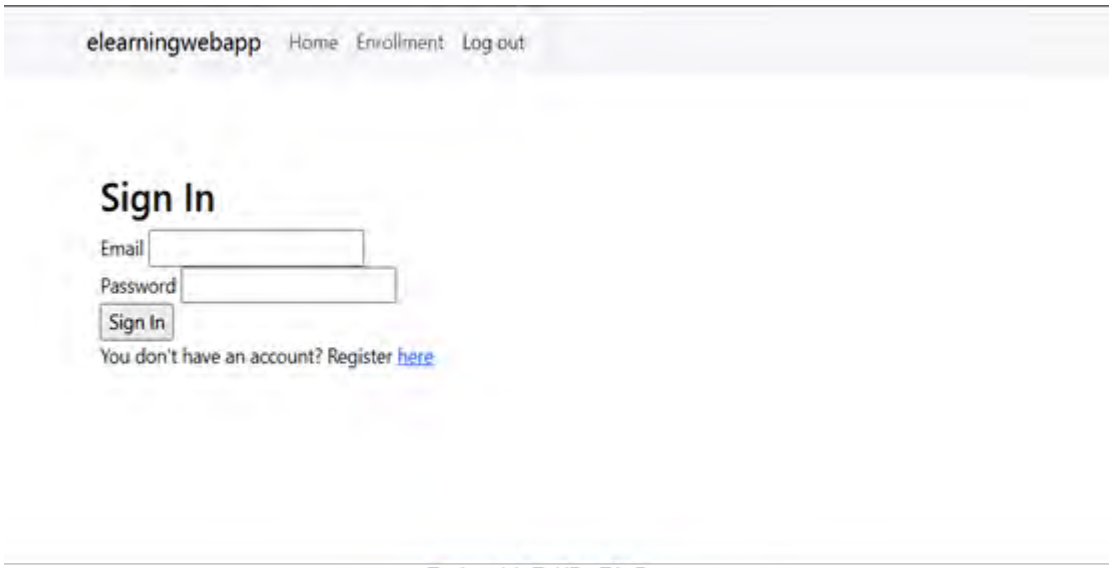
Διεπαφή Καθηγητή	Διεπαφή Φοιτητή
Αρχική Σελίδα Διδάσκοντα	Αρχική Σελίδα Φοιτητή
Δημιουργία μαθήματος	Εγγραφή σε μάθημα
Προβολή λεπτομερειών/περιγραφής μαθήματος	Προβολή ανακοινώσεων μαθήματος
Δημοσίευση ανακοινώσεων μαθήματος	Προβολή υλικού μαθήματος
Δημοσίευση υλικού μαθήματος	Προβολή βαθμού
Δημοσίευση εργασίας και προβολής λύσεων των φοιτητών	Προβολή της εκφώνησης της εργασίας και ανέβασμα λύσης
Δημοσίευση βαθμολογιών σε κάθε φοιτητή	Προβολή ερώτησης αξιολόγησης γνώσεων και δια δραστικού εργαλείου
Δημιουργία διάλεξης	
Προβολή αποτελεσμάτων ερώτησης αξιολόγησης	
Προβολή στατιστικών δεδομένων δια δραστικού εργαλείου	

6.1 ΣΕΛΙΔΑ ΣΥΝΔΕΣΗΣ/ΕΓΓΡΑΦΗΣ

Παρακάτω εμφανίζονται οι σελίδες σύνδεσης (Εικόνα 14), εγγραφής (Εικόνα18) και τα σχετικά τμήματα κώδικα (Εικόνες 15-17, Εικόνες 19-21). Στην σελίδα εγγραφής ο χρήστης συμπληρώνει το μειλ του και τον κωδικό πρόσβασης. Επιπλέον στην εγγραφή ο νέος χρήστης εισάγει το

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

όνομα χρήστη, το συνθηματικό, το μειλ και τέλος επιλέγει και τον ρόλο που θα έχει.



Εικόνα 14. Σελίδα Σύνδεσης

```
class LoginForm(Form):
    username = StringField('Email',
                           id='username_login',
                           validators=[DataRequired()])
    password = PasswordField('Password',
                             id='pwd_login',
                             validators=[DataRequired()])
```

Εικόνα 15. Κώδικας για τη δημιουργία φόρμας σύνδεσης.

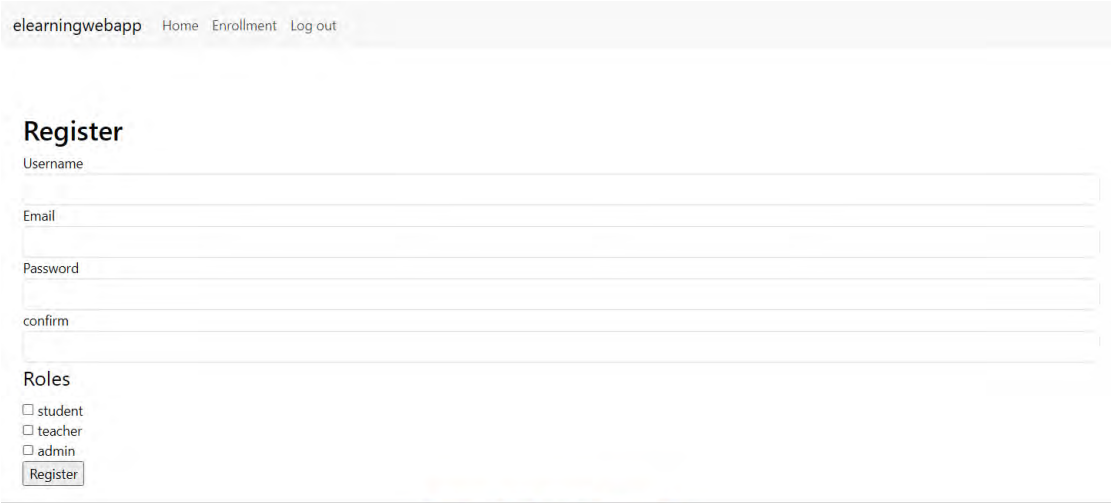
```
{% extends 'accounts/layout.html' %}

{% block content %}
<h2>Sign In</h2>

{% if msg %}
<p style="color: red;">{{ msg }}</p>
{% endif %}

{% with messages = get_flashed_messages() %}
  {% if messages %}
<ul class=flashes>
    {% for message in messages %}
    <li class="error">{{ message }}</li>
    {% endfor %}
</ul>
  {% endif %}
{% endwith %}
<!--<p>Demo User: admin@elearningweb.app - 1234 -test</p>-->
<form method="post">
    {{ form.hidden_tag }}
    <div class="form-group">
        <label for="username">{{ form.username.label }}</label>

        {{ form.username() }}
    </div>
    <div class="form-group">
        <label for="password"> {{ form.password.label }}</label>

        {{ form.password() }}
    </div>


    <div class="form-group">
        <input type="submit" name="login" value="Sign In">
    </div>
</form>
<p>You don't have an account? Register <a class="link-opacity-100" href="{{url_for('authentication_blueprint.register')}}">here</a></p>
{% endblock %}
```

Εικόνα 16. login.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
@blueprint.route('/login', methods=['GET', 'POST'])
def login():
    #logging.Logger.info('Hit Login')
    login_form = LoginForm(request.form)
    if 'login' in request.form:

        # read form data
        username = request.form['username']
        password = request.form['password']

        # Locate user
        user = Users.query.filter_by(email=username).first()

        # Check the password
        if user and verify_pass(password, user.password):
            login_user(user)
            logging.info('%s login successfully', user.username)
            return redirect(url_for('authentication_blueprint.route_default'))
        # Something (user or pass) is not ok
        flash('Invalid password provided', 'error')
        return render_template('accounts/login.html', form=login_form)

    if not current_user.is_authenticated:
        return render_template('accounts/login.html', form=login_form)
    else:
        # Check user role and redirect accordingly
        if user_has_role('admin'):
            return redirect(url_for('home_blueprint.index'))
        elif user_has_role('teacher'):
            return redirect(url_for('teacherhome_blueprint.index'))
        elif user_has_role('student'):
            return redirect(url_for('home_blueprint.index'))
        else:
            return redirect(url_for('home_blueprint.index'))

    return redirect(url_for('home_blueprint.index'))


def user_has_role(role_name):
    if not current_user.is_authenticated:
        return False  # or redirect to login, based on your application's flow

    # Check if any of the user's roles match the role_name
    return any(role.name == role_name for role in current_user.roles)
```

Εικόνα 17. Δημιουργία URL σύνδεσης



Εικόνα 18. Σελίδα εγγραφής

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
class MultiCheckboxField(SelectMultipleField):
    widget = ListWidget(prefix_label=False)
    option_widget = CheckboxInput()

class CreateAccountForm(Form):
    username = StringField('Username',
                           id='username_create',
                           validators=[DataRequired()])
    email = StringField('Email',
                        id='email_create',
                        validators=[DataRequired(), Email()])
    roles = MultiCheckboxField('Roles', coerce=int)
    password = PasswordField('Password',
                             id='pwd_create',
                             validators=[DataRequired()])
    confirm = PasswordField('confirm',
                            id='confirm',
                            validators=[DataRequired()])
```

Εικόνα 19. Δημιουργία φόρμας για εγγραφή χρήστη.

```python
@blueprint.route('/register', methods=['GET', 'POST'])
# @roles_accepted('admin')
def register():
    create_account_form = CreateAccountForm(request.form)
    create_account_form.roles.choices = [(role.id, role.name) for role in Role.query.all()]
    if 'register' in request.form:

        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        roles_selected = create_account_form.roles.data
        #confirm_password= request.form['confirm']

        user = Users.query.filter_by(username=username).first()
        if user:
            return render_template('accounts/register.html',
                                   msg='Username already registered',
                                   success=False,
                                   form=create_account_form)


        user = Users.query.filter_by(email=email).first()
        if user:
            return render_template('accounts/register.html',
                                   msg='Email already registered',
                                   success=False,
                                   form=create_account_form)


        #user = Users(**request.form)
        user = Users(username=username, email=email, password=password)
        user.roles = Role.query.filter(Role.id.in_(roles_selected)).all()
        user.active = True
        db.session.add(user)
        db.session.commit()

        return render_template('accounts/register.html',
                               msg='User created please <a href="/login">login</a>',
                               success=True,
                               form=create_account_form)
```

Εικόνα 20. Δημιουργία URL για την εγγραφή χρήστη

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

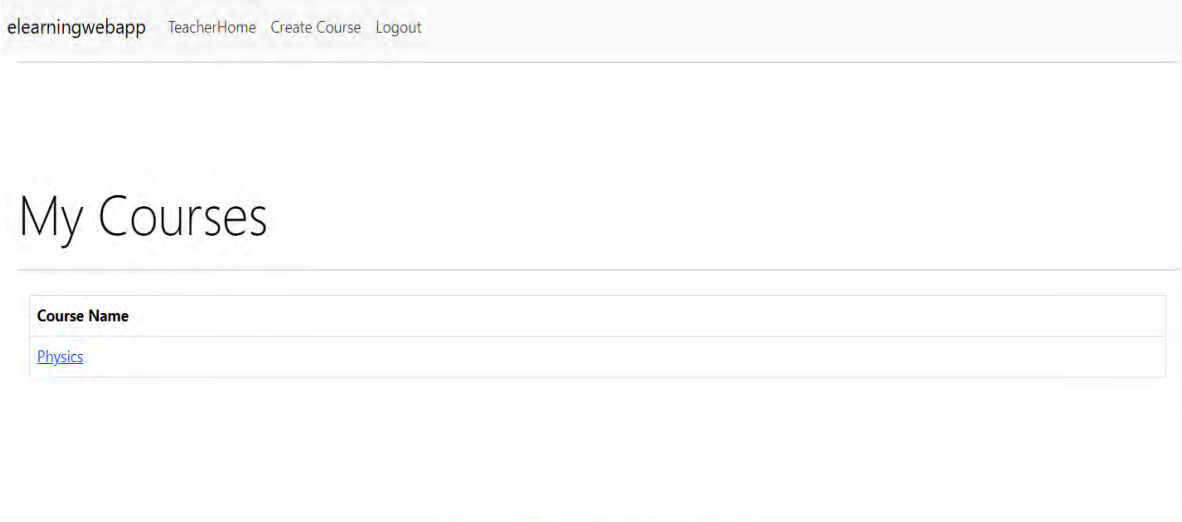
```
{% extends 'accounts/layout.html' %}

{% block content %}
<h2>Register</h2>
<form method="post">
    {{ form.hidden_tag }}
    {{ form.csrf_token }}
    <div>
        {{ form.username.label }}
        {{ form.username(class="form-control") }}
    </div>
    <div>
        {{ form.email.label }}
        {{ form.email(class="form-control") }}
    </div>
    <div>
        {{ form.password.label }}
        {{ form.password(class="form-control") }}
    </div>
    <div>
        {{ form.confirm.label }}
        {{ form.confirm(class="form-control") }}
    </div>
    <fieldset>
        <legend>{{ form.roles.label }}</legend>
        {% for subfield in form.roles %}
        <div>
            <label>{{ subfield }} {{ subfield.label }}</label>
        </div>
        {% endfor %}
    </fieldset>
    <input type="submit" name="register" value="Register">
</form>

{% endblock %}
```

Εικόνα 21.register.html

6.2 ΔΙΕΠΑΦΗ ΔΙΔΑΣΚΟΝΤΑ

6.2.1 Αρχική Σελίδα Διδάσκοντα

Αμέσως μετά την σύνδεση στον λογαριασμό, εμφανίζεται η αρχική σελίδα του διδάσκοντα στην οποία εμφανίζονται όλα τα μαθήματα που διδάσκει (Εικόνα 22). Ακόμη, παρατίθενται τα κομμάτια κώδικα που αντιστοιχούν στην ανάκτηση των μαθημάτων του καθηγητή (Εικόνα 23) και στην γραφική απεικόνιση (Εικόνα 24).



Εικόνα 22. Αρχική σελίδα καθηγητή

```
@blueprint.route('/teacher')
@roles_accepted('admin','teacher')
def index():
    user_id  = current_user.id


    teacher_courses = Course.query.filter_by(proffessor=current_user.username).all()
    return render_template('teacherhome/teacherindex.html', segment ='teacherindex', teacher_courses = teacher_courses)
```

Εικόνα 23. Κώδικας για δημιουργία URL της αρχικής σελίδας

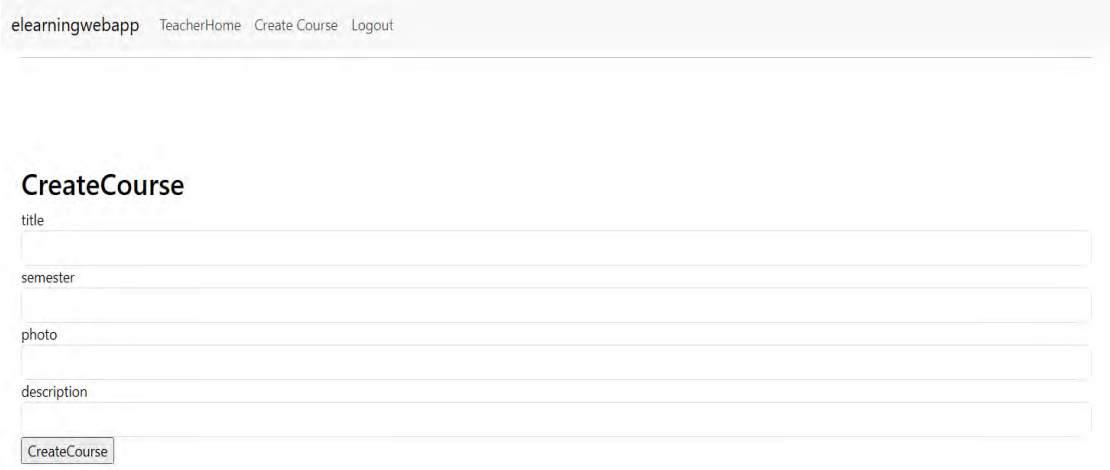
Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```html
{% extends "teacherhome/layout.html" %}
{% block metatags %} {% endblock %}
{% block content %}

<div class="jumbotron">
    <h1 class="display-4">My Courses</h1>

    <hr class="my-4">
    <div class="container">
        <div class="row">
            <div class="col-12">
                <table class="table table-bordered">
                    <thead>
                        <tr>
                            <th scope="col">Course Name</th>


                        </tr>
                    </thead>
                    <tbody>
                        {%for course in teacher_courses%}
                        <tr>
                            <td><a href="{{url_for('teacherhome_blueprint.courseteacher', course_id = course.id)}}">{{course.title}}</a></td>


                        </tr>
                        {%endfor%}

                    </tbody>
                </table>
            </div>
        </div>
    </div>

</div>
{% endblock %}
```

Εικόνα 24. teacherindex.html

6.2.2 Σελίδα δημιουργίας μαθήματος

Στην σελίδα αυτή (Εικόνα 25) ο καθηγητής μπορεί να φτιάξει το μάθημα που θα διδάξει, συμπληρώνοντας τον τίτλο, το εξάμηνο, φωτογραφία και μια σύντομη περιγραφή. Στις Εικόνες 26-28 απεικονίζονται τα τμήματα κώδικα που σχετίζονται με την δημιουργία μαθήματος.



Εικόνα 25. Σελίδα δημιουργίας μαθήματος

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
@blueprint.route('teacher/createcourse', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def createcourse():
    form = CreateCourseForm(request.form)
    if request.method == 'POST' and form.validate():
        title = form.title.data
        semester = form.semester.data
        description = form.description.data
        photo = form.photo.data
        proffessor = current_user.username

        course = Course.query.filter_by(title=title).first()
        if course:
            return render_template('teacherhome/createcourse.html',
                                   msg='Course already created',
                                   success=False,
                                   form=form)



        course = Course(**request.form)
        course.proffessor = current_user.username
        db.session.add(course)
        db.session.commit()

        return render_template('teacherhome/createcourse.html',
                               msg='Course created',
                               success=True,
                               form=form)

    else:
        return render_template('teacherhome/createcourse.html', form=form)
```

Εικόνα 26. Κώδικας για δημιουργία URL που σχετίζεται με την δημιουργία μαθήματος

```python
class CreateCourseForm(Form):
    title = StringField('title',
                        id='title_create',
                        validators=[DataRequired()])
    semester = StringField('semester',
                        id='course_semester',
                        validators=[DataRequired()])
    description = StringField('description',
                        id='course_desc',
                        validators=[DataRequired()])
    photo = StringField('photo',
                        id='photo',
                        validators=[DataRequired()])
```

Εικόνα 27. Δημιουργία φόρμας δημιουργίας μαθήματος.

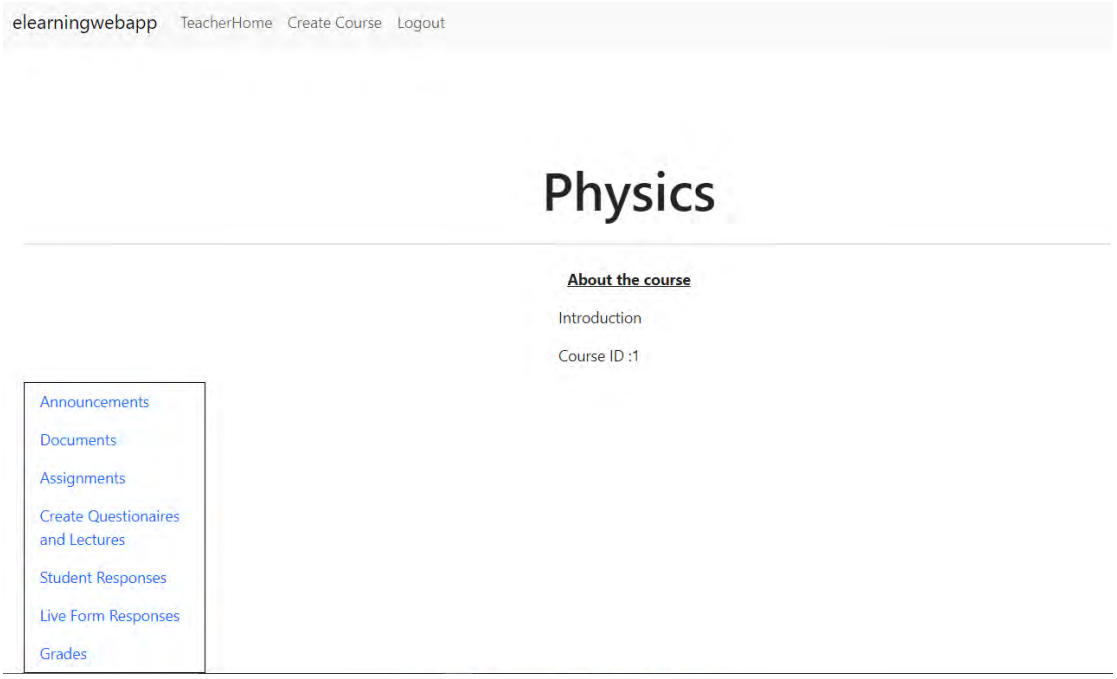
Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
{% extends 'teacherhome/layout.html' %}

{% block content %}
<h2>CreateCourse</h2>
<form method="post">
    {{ form.hidden_tag }}
    {{ form.csrf_token }}
    <div>
        {{ form.title.label }}
        {{ form.title(class="form-control") }}
    </div>
    <div>
        {{ form.semester.label }}
        {{ form.semester(class="form-control") }}
    </div>
    <div>
        {{ form.photo.label }}
        {{ form.photo(class="form-control") }}
    </div>
    <div>
        {{ form.description.label }}
        {{ form.description(class="form-control") }}
    </div>
    <input type="submit" name="createcourse" value="CreateCourse">
</form>
{% endblock %}
```

Εικόνα 28. createcourse.html

6.2.3 Προβολή σελίδας λεπτομερειών/περιγραφής του μαθήματος

Στην αρχική σελίδα του μαθήματος (Εικόνα 29) εμφανίζεται η σύντομη περιγραφή και το id του μαθήματος. Στις εικόνες 30-31 εμφανίζεται το κομμάτι κώδικα υπεύθυνο για αυτή την λειτουργία.



Εικόνα 29. Αρχική σελίδα μαθήματος

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
@blueprint.route('/courseteacher/<int:course_id>')
@roles_accepted('admin','teacher')
def courseteacher(course_id):
    courseteacher = Course.query.get(course_id)
    return render_template('teacherhome/courseteacher.html',courseteacher = courseteacher)
```

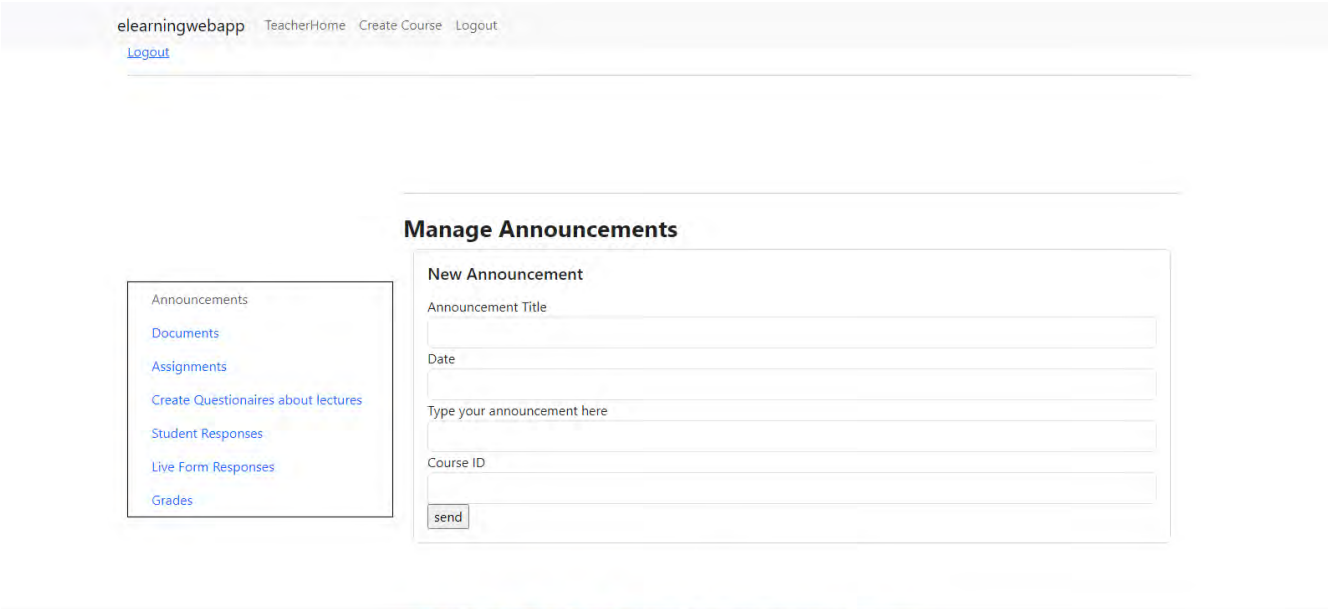
Εικόνα 30. Κώδικας για δημιουργία URL για την αρχική σελίδα μαθήματος

```
{% extends "teacherhome/layout.html" %}
{% block content %}
<div class="jumbotron">

    {%if courseteacher%}
    <h1 class="display-4" align="center" style="font-weight:600">{{courseteacher.title}} </h1>
    <hr class="my-4">
    <div class="container">
        <p class="text-center"><strong><u>About the course</u></strong></p>
        <p style="margin-left:550px">{{courseteacher.description}}</p>
        <p style="margin-left:550px">Course ID :{{courseteacher.id}}</p>
        {%endif%}
    </div>
    <div style="width:15% ;border:1px solid black; height:100% ">
        <ul class="nav flex-column">
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = courseteacher.id)}}">Announcements</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = courseteacher.id)}}">Documents</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = courseteacher.id)}}">Assignments</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = courseteacher.id)}}">Create  Questionaires and Lectures</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = courseteacher.id)}}">Student Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = courseteacher.id)}}">Live Form Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = courseteacher.id)}}">Grades</a>
            </li>
        </ul>
    </div>
</div>

    {% endblock %}
```

Εικόνα 31. courseteacher.html

6.2.4 Σελίδα δημοσίευσης ανακοινώσεων μαθήματος

Στην Εικόνα 32 ο διδάσκοντας μπορεί να δημοσιεύσει ανακοίνωση εισάγοντας τον τίτλο, την ημερομηνία, το κείμενο και το id του μαθήματος. Οι εικόνες 33 και 34 απεικονίζουν τον κώδικα για τη δημοσίευση των ανακοινώσεων.



Εικόνα 32.Σελίδα δημοσίευσης ανακοινώσεων

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
@blueprint.route('/teacher/announcementsteacher/<int:course_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def announcementsteacher(course_id):
    form = CreateAnnouncementForm(request.form)
    course = Course.query.get(course_id)
    if request.method == 'POST' and form.validate():
        announcement_title = form.announcement_title.data
        announcement_description = form.announcement_description.data
        announcement_date = form.announcement_date.data


        course_id = form.course_id.data




        announcement_course = Announcement(
            announcement_title=announcement_title,
            announcement_description=announcement_description,
            announcement_date=announcement_date,
            course_id=course_id
        )

        db.session.add(announcement_course)
        db.session.commit()

        return render_template('teacherhome/announcementsteacher.html',
                               msg='Announcement created',
                               success=True,
                               form=form,course = course)



    else:
        return render_template('teacherhome/announcementsteacher.html', form=form, course = course)
```

Εικόνα 33. Δημιουργία URL για την σελίδα δημοσίευσης ανακοινώσεων.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```html
{% extends "teacherhome/layout.html" %}
{% block metatags %} {% endblock %}
{% block content %}
<div class="container-fluid">
    <div class="row">


      <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
          <ul class="nav flex-column">
              <li class="nav-item">
                  <a class="nav-link disabled" href="#">Announcements</a>
              </li>
              <li class="nav-item">
                  <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
              </li>
              <li class="nav-item">
                  <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = course.id)}}">Assignments</a>
              </li>
              <li class="nav-item">
                  <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires and Lectures</a>
              </li>
              <li class="nav-item">
                  <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
              </li>
              <li class="nav-item">
                  <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
              </li>
              <li class="nav-item">
                  <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
              </li>
          </ul>
      </div>


      <div class="col-md-9">
        <div class="jumbotron">

          <h1 class="display-4" align="center" style="font-weight:600"> </h1>
          <hr class="my-4">
          <h3><strong>Manage Announcements</strong></h3>
```


```html
          <h3><strong>Manage Announcements</strong></h3>
```

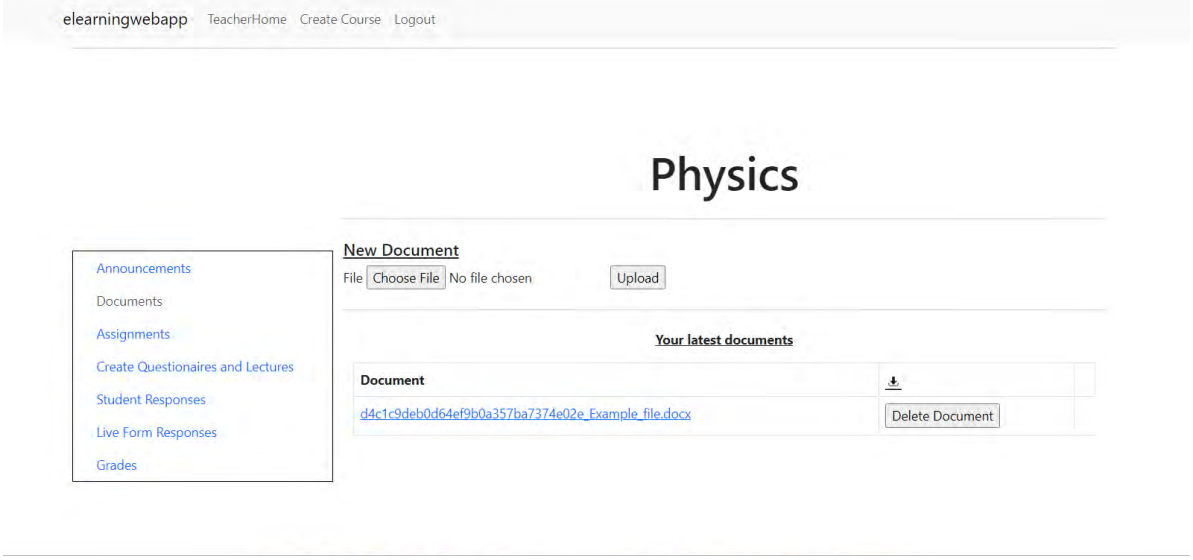

```html
          <div class="container">
            <div class="row">
              <div class="col-12">
                <div class="card">
                    <div class="card-body">
                        <div class="mb-3">
                            <h5 class="card-title">New Announcement</h5>
                        </div>
                        <form method="post">
                            {{ form.hidden_tag }}
                            {{ form.csrf_token }}
                            <div>
                                {{ form.announcement_title.label }}
                                {{ form.announcement_title(class="form-control") }}
                            </div>
                            <div>
                                {{ form.announcement_date.label }}
                                {{ form.announcement_date(class="form-control") }}
                            </div>
                            <div>
                                {{ form.announcement_description.label }}
                                {{ form.announcement_description(class="form-control") }}
                            </div>
                            <div>
                                {{ form.course_id.label }}
                                {{ form.course_id(class="form-control") }}
                            </div>
                            <input type="submit" name="announcementsteacher" value="send">
                        </form>


                    </div>
                </div>

              </div>
            </div>



          </div>
        </div>
      </div>
    </div>
    </div>
{% endblock %}
```

Εικόνα 34. announcementsteacher.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.5 Σελίδα δημοσίευσης υλικού μαθήματος

Ο καθηγητής μπορεί να δημοσιεύσει έγγραφα για το υλικό του μαθήματος (Εικόνα 35) καθώς και να τα διαχειριστεί διαγράφοντας τα. Τα έγγραφα που έχουν δημοσιευθεί εμφανίζονται ακριβώς κάτω από την φόρμα δημοσίευσης. Επιπροσθέτως, το τμήμα κώδικα που εκτελεί τα παραπάνω εμφανίζεται στις εικόνες 36-40.



Εικόνα 35. Σελίδα δημοσίευσης υλικού μαθήματος και διαχείρισης του

```
@blueprint.route('teacher/documentsteacher/<int:course_id>', methods = ['GET','POST'])
@roles_accepted('admin', 'teacher')
def upload_document(course_id):
    form  = FileUploadForm()
    course =Course.query.get(course_id)
    if request.method == 'POST' and form.validate_on_submit():
        uploaded_file  = form.file.data



        directory = os.path.join('media', 'documents', str(course_id))
        absolute_path = os.path.abspath(directory)

        if not os.path.exists(absolute_path):
            os.makedirs(absolute_path)



        unique_filename = uuid4().hex + '_' + uploaded_file.filename
        file_path =os.path.join(absolute_path, secure_filename(unique_filename))
        uploaded_file.save(file_path)

        document = Document(filename=secure_filename(unique_filename), course_id=course_id)
        db.session.add(document)
        db.session.commit()

        return redirect(url_for('teacherhome_blueprint.upload_document', course_id=course_id))

    documents = Document.query.filter_by(course_id=course_id).all()
    return render_template("teacherhome/documentsteacher.html", form=form, documents=documents,course =course)
```

Εικόνα 36. Κώδικας για δημιουργία URL για τη δημοσίευση εγγράφων

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
class FileUploadForm(FlaskForm):
    file = FileField('File', validators=[FileRequired(), FileAllowed(['pdf', 'docx', 'txt'], 'File types allowed: PDF, DOCX, TXT')])
    submit = SubmitField('Submit')
```

Εικόνα 37. Δημιουργία Φόρμας δημοσίευσης εγγράφου

```python
@blueprint.route('teacher/delete_document/<int:course_id>/<int:document_id>', methods=['POST'])
@roles_accepted('admin', 'teacher')
def delete_document(course_id,document_id):
    document = Document.query.filter_by(course_id=course_id).first()
    if document:
        db.session.delete(document)
        db.session.commit()
        flash('The document is deleted', 'success')
    else:
        flash('Could not delete the document', 'error')
    return redirect(url_for('teacherhome_blueprint.upload_document', course_id = course_id))
```

Εικόνα 38. Δημιουργία URL για διαγραφή εγγράφου (πλήκτρο διαγραφής).

```html
<div class="col-md-9">
    <div class="jumbotron">


        <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
        <hr class="my-4">
        <p class="h5"><u>New Document</u></p>
        <form method="POST" enctype="multipart/form-data">
            {{ form.csrf_token }}
            {{ form.file.label }}
            {{ form.file }}
            <input type="submit" name="file" value="Upload">
            </form>
            <hr class="my-4">

            <div class="container"><p class="text-center"><strong><u>Your latest documents</u></strong></p></div>

            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <table class="table table-bordered">
                            <thead>
                                <tr>

                                    <th scope="col">Document</th>

                                    <th scope="col"><img src="https://cdn4.vectorstock.com/i/1000x1000/93/18/download-icon-for-web-app-logo-ui-vector-27559318.
                                  <th></th>

                                </tr>
                            </thead>
                            <tbody>
                                {% for document in documents %}
                                <tr>
                                    <td><a href="#" target="_blank" title="{{ document.filename }}" download>{{ document.filename }}</a></td>

                                    <form method="POST" action="{{ url_for('teacherhome_blueprint.delete_document', course_id=course.id,document_id = document.
                                        {{ form.csrf_token }}
                                        {{ form.hidden_tag() }}

                                        <input type="submit" value="Delete Document" name="submit">
                                    </td>
```

Εικόνα 39. documentsteacher.html

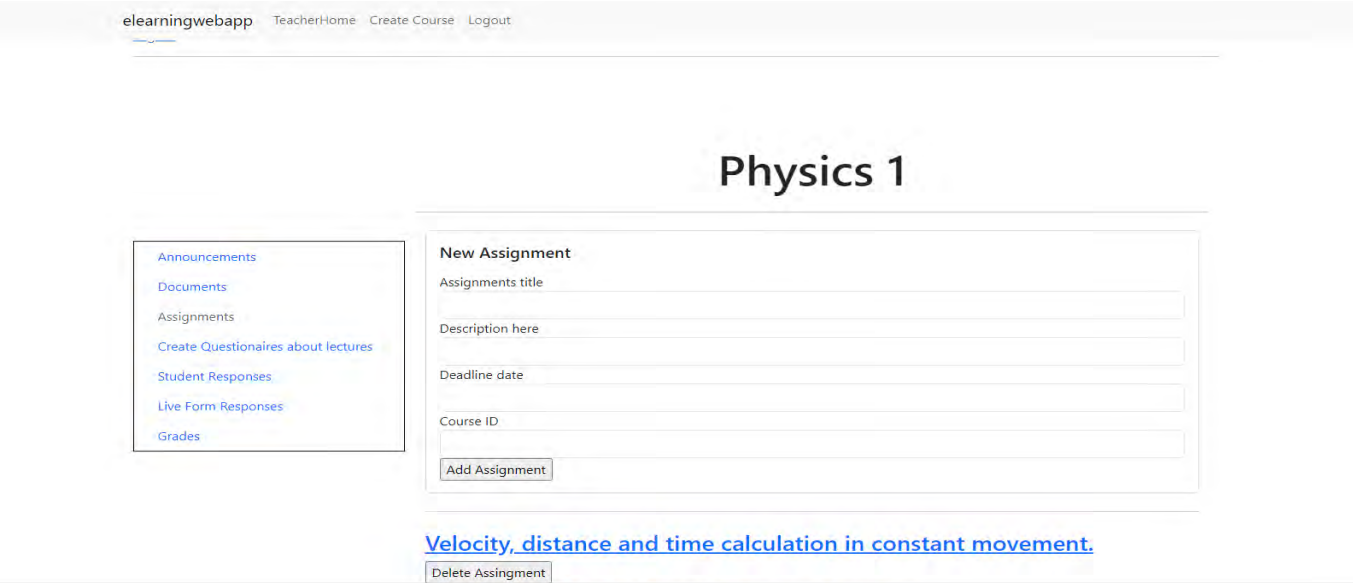
```html
                <div class="container">
                    <div class="row">
                        <div class="col-12">
                            <table class="table table-bordered">
                                <thead>
                                    <tr>

                                        <th scope="col">Document</th>

                                        <th scope="col"><img src="https://cdn4.vectorstock.com/i/1000x1000/93/18/download-icon-for-web-app-logo-ui-vector-27559318.
                                      <th></th>

                                    </tr>
                                </thead>
                                <tbody>
                                    {% for document in documents %}
                                    <tr>
                                        <td><a href="#" target="_blank" title="{{ document.filename }}" download>{{ document.filename }}</a></td>

                                        <form method="POST" action="{{ url_for('teacherhome_blueprint.delete_document', course_id=course.id,document_id = document.
                                            {{ form.csrf_token }}
                                            {{ form.hidden_tag() }}
<td>
                                            <input type="submit" value="Delete Document" name="submit">
                                        </td>
                                        </form>

                                    </tr>
                                    {% endfor %}
                                </tbody>
                            </table>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
  {% endblock %}
```

Εικόνα 40.assignmentsteacher.html (συνέχεια)

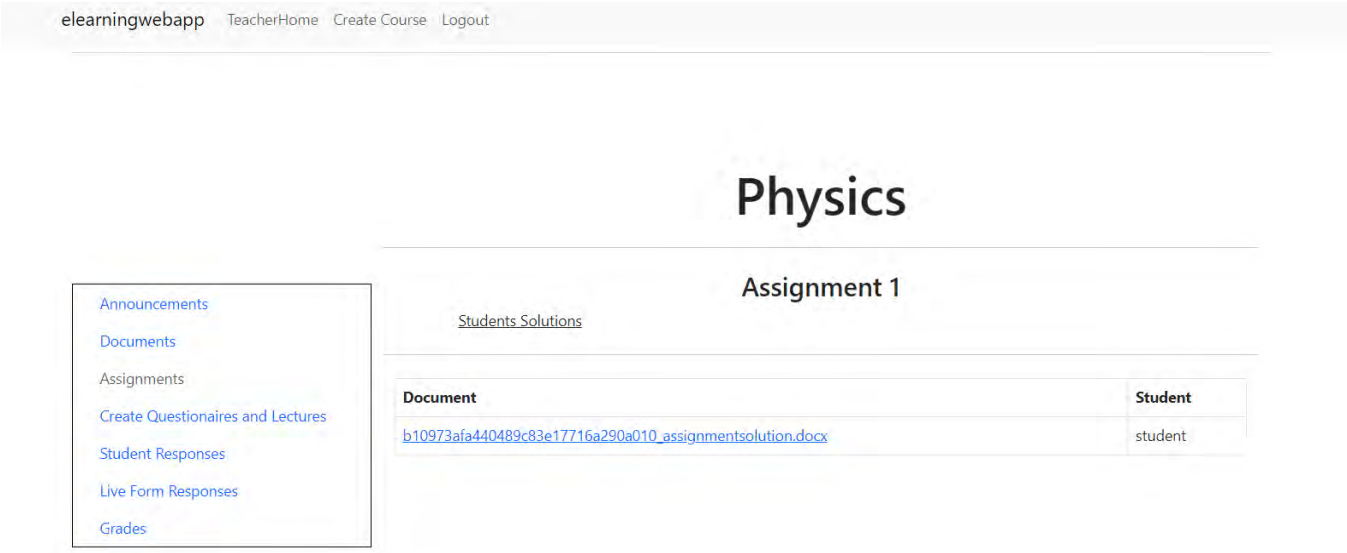
Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.6 Δημοσίευση εργασίας και προβολής λύσεων των φοιτητών για κάθε εργασία

Στην Εικόνα 41 αναπαριστά τη σελίδα που δημοσιεύονται οι εργασίες του μαθήματος, εισάγοντας τον τίτλο της εργασίας, την εκφώνηση/περιγραφή, την ημερομηνία λήξης της προθεσμίας και τέλος το id του μαθήματος. Οι εργασίες που έχουν δημοσιευθεί εμφανίζονται κάτω από την φόρμα δημοσίευσης εργασίας σε μορφή συνδέσμου που οδηγεί στις λύσεις που υπέβαλλαν οι φοιτητές (Εικόνα 42). Στις εικόνες 44-47 ο κώδικας σχετίζεται με την δημοσίευση και διαχείριση της εργασίας και στις εικόνες 43, 48 με την προβολή των λύσεων των φοιτητών.



Εικόνα 41. Σελίδα δημοσίευσης της εργασίας



Εικόνα 42.Σελίδα προβολής λύσεων εργασίας από τους φοιτητές

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
@blueprint.route('teacher/assignmentsolution/<int:course_id>/<int:assignment_id>')
@roles_accepted('admin', 'teacher')
def assignmentsolution(course_id,assignment_id):
    course = Course.query.get(course_id)
    assignment = Assignment.query.get(assignment_id)
    assignmentsolutions = AssignmentSolution.query.filter_by(course_id = course_id,assignment_id = assignment_id ).all()
    return render_template('teacherhome/assignmentsolution.html',assignmentsolutions =assignmentsolutions,course =course,assignment =assignment
```

Εικόνα 43. Κώδικας για δημιουργία URL για προβολή των αρχείων με τις λύσεις των φοιτητών.

```python
@blueprint.route('teacher/assignmentsteacher/<int:course_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def addassignment(course_id):
    form = CreateAssignmentForm(request.form)
    deleteassignmentform = DeleteAssignment()
    course= Course.query.get(course_id)
    assignments = Assignment.query.filter_by(course_id = course_id).all()
    #student_solutions = None
    # if assignments:
     #student_solutions =AssignmentSolution.query.join(Assignment).filter(AssignmentSolution.assignment_id == Assignment.id).all()
    if request.method == 'POST' and form.validate():
        assignment_title = form.assignment_title.data
        assignment_description = form.assignment_description.data
        deadline_date = form.assignment_date.data


        course_id = form.course_id.data




        assignment = Assignment(
            assignment_title=assignment_title,
            assignment_desc=assignment_description,
            deadline_date=deadline_date,
            course_id=course_id
        )

        db.session.add(assignment)
        db.session.commit()

    return render_template('teacherhome/assignmentsteacher.html',
                           msg='Assignment created',
                           success=True,
                           course = course,
                           assignments = assignments,
                           deleteassignmentform =deleteassignmentform,
                           form=form)
```

Εικόνα 44. Δημιουργία URL για το ανέβασμα εκφώνησης εργασίας.

```python
@blueprint.route('/delete_assignment/<int:course_id>/<int:assignment_id>', methods=['POST'])
@roles_accepted('admin', 'teacher')
def delete_assignment(course_id,assignment_id):
    assignment = Assignment.query.filter_by(course_id = course_id).first()
    solutions = AssignmentSolution.query.filter_by(course_id = course_id, assignment_id = assignment_id).all()
    if assignment:
        db.session.delete(assignment)
        for solution in solutions:
            db.session.delete(solution)
        db.session.commit()


        flash('The assignment is deleted', 'success')
    else:
        flash('The assignment does not exist', 'error')
    return redirect(url_for('teacherhome_blueprint.addassignment',course_id =course_id))
```

Εικόνα 45. Δημιουργία URL για τη διαγραφή εργασίας.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
{% extends "teacherhome/layout.html" %}
{% block metatags %} {% endblock %}
{% block content %}
<div class="container-fluid">
    <div class="row">

        <!-- Navigation -->
        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires about
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveform_responses', course_id = course.id)}}">Live Form Responses</a
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>


        <div class="col-md-9">
            <div class="jumbotron">


                <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
                <hr class="my-4">
                <div class="container">
                    <div class="row">
                        <div class="col-12">
                            <div class="card">
                                <div class="card-body">
                                    <div class="mb-3">
                                        <h5 class="card-title">New Assignment</h5>
                                    </div>
                                    <form method="post">
                                        {{ form.hidden_tag }}
                                        {{ form.csrf_token }}
```

Εικόνα 46. assignmentsteachers.html

```
                                    <div class="mb-3">
                                        <h5 class="card-title">New Assignment</h5>
                                    </div>
                                    <form method="post">
                                        {{ form.hidden_tag }}
                                        {{ form.csrf_token }}
                                        <div>
                                            {{ form.assignment_title.label }}
                                            {{ form.assignment_title(class="form-control") }}
                                        </div>
                                        <div>
                                            {{ form.assignment_description.label }}
                                            {{ form.assignment_description(class="form-control") }}
                                        </div>
                                        <div>
                                            {{ form.assignment_date.label }}
                                            {{ form.assignment_date(class="form-control") }}
                                        </div>
                                        <div>
                                            {{ form.course_id.label }}
                                            {{ form.course_id(class="form-control") }}
                                        </div>
                                        <input type="submit" name="assignmentsteacher" value="Add Assignment">
                                    </form>



                                    </div>
                                </div>

                            </div>
                        </div>


                        <hr class="my-4">
                        {%for assignment in assignments%}
                        <h3><u><a href="{{url_for('teacherhome_blueprint.assignmentsolution', course_id=course.id,assignment_id = assignment.id)}}">{{ a


        <form method="POST" action="{{ url_for('teacherhome_blueprint.delete_assignment', course_id=course.id,assignment_id = assignment.id) }}"
            {{ deleteassignmentform.csrf_token }}
            {{ deleteassignmentform.hidden_tag() }}

            <input type="submit" value="Delete Assingment" name="submit">

        </form>
        {%endfor%}
```

Εικόνα 47.assignmentsteacher.html (συνέχεια)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

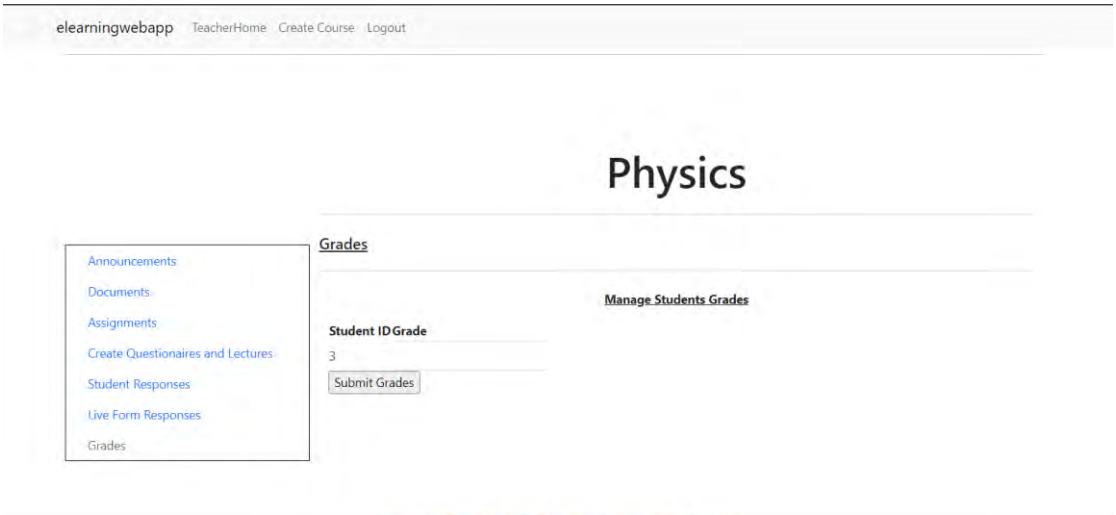
```
{% extends "teacherhome/layout.html" %}
{% block metatags %} {% endblock %}
{% block content %}
<div class="container-fluid">
    <div class="row">

        <!-- Navigation -->
        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires about
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveform_responses', course_id = course.id)}}">Live Form Responses</a
                </li>
                <li class="nav-item">
                    <a class="nav-link " href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>
    <div class="col-md-9">
        <div class="jumbotron">
                    <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
            <hr class="my-4">
            <h3  align="center" style="font-weight:600">{{assignment.assignment_title}} </h3>
        <p style="margin-left:80px">  <u> Students Solutions</u></p>
            <hr class="my-4">
    <div class="container">
            <div class="row">
                <div class="col-12">
                    <table class="table table-bordered">
                        <thead>
                          <tr>
                            <th scope="col">Document</th>
                            </tr>
                        </thead>
                        <tbody>
                            <tr>
                                {% for assignmentsolution in assignmentsolutions %}
        <td><a href="#" target="_blank" title="{{ assignmentsolution.filename_solution }}" download>{{ assignmentsolution.filename_solution }}</a><
                                {% endfor %}
                            </tr>
        </tbody>
                    </table>
                </div>
            </div>
        </div>
      </div>
    </div>
  </div>
{% endblock %}
```

Εικόνα 48.assignemntsolution.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.7 Δημοσίευση βαθμών σε κάθε φοιτητή ξεχωριστά

Στην υποενότητα αυτή παρουσιάζονται η σελίδα δημοσίευσης βαθμολογιών στην οποία εμφανίζονται τα id των φοιτητών ώστε να δημοσιευθούν οι βαθμοί στον κάθε φοιτητή ξεχωριστά (Εικόνα 49) μαζί με τα τμήματα κώδικα (Εικόνες 50-52) που είναι υπεύθυνα για αυτή την διαδικασία.



Εικόνα 49. Σελίδα δημοσίευσης βαθμών

```
class GradeForm(FlaskForm):
    student_grade = StringField('Grade for Student ')
    submit = SubmitField('Submit')
```

Εικόνα 50 Δημιουργία φόρμας καταχώρησης βαθμού.

```
@blueprint.route('teacher/pass_grades/<int:course_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def pass_grades(course_id):
    form = GradeForm()

    course = Course.query.get(course_id)
    enrolled_students = Enrollment.query.filter_by(course_id=course_id).all()

    if form.validate_on_submit():

        for enrollment in enrolled_students:
            student_id = enrollment.user_id
            new_grade = form.student_grade.data
            enrollment.grade = new_grade
            db.session.commit()

        flash('Grades have been successfully updated!', 'success')

    return render_template('teacherhome/gradesteacher.html', form=form,course=course, enrolled_students=enrolled_students)
```

Εικόνα 51. Δημιουργία URL για τη δημοσίευση βαθμού σε κάθε φοιτητή.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

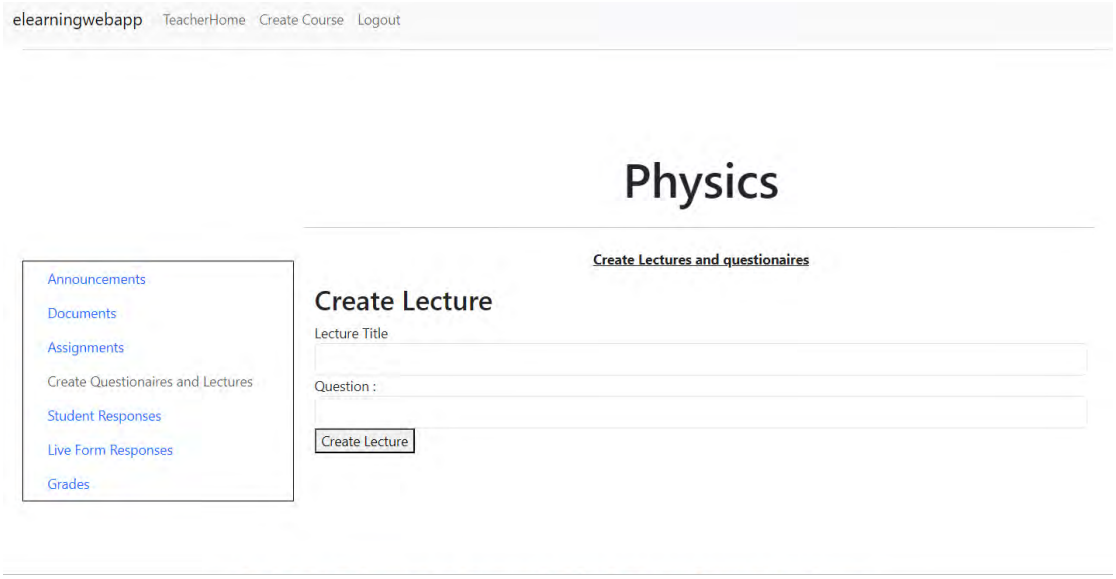
```
{% extends "teacherhome/layout.html" %}
{% block metatags %} {% endblock %}
{% block content %}
<div class="container-fluid">
    <div class="row">

        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = course.id)}}">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires and Lectures</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Grades</a>
                </li>
            </ul>
        </div>


        <div class="col-md-9">
          <div class="jumbotron">


            <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
            <hr class="my-4">
            <p class="h5"><u>Grades</u></p>


            <hr class="my-4">
        <div class="container"><p class="text-center"><strong><u>Manage Students Grades</u></strong></p></div>
          <div class="container">
                <div class="row">
                    <div class="col-12">
                        {% if enrolled_students %}
                        <form method="POST">
                            {{ form.hidden_tag() }}
                            <table>
                                <tr>
                                    <th>Student ID</th>
                                    <th></th>
                                </tr>
                                {% for enrollment in enrolled_students%}
                                <tr>
                                    <td>{{ enrollment.user_id }}</td>
                                    <td>{{ form.student_grade(class="form-control") }}</td>
                                </tr>
                                {% endfor %}
                            </table>
                            <input type="submit" value="Submit Grades">
                        </form>
                        {% else %}
                        <p>No students enrolled yet.</p>
                        {% endif %}
                    </div>
                </div>

            </div>
            </div>

          </div>

        </div>
    {% endblock %}
```

Εικόνα 52. gradesteacher.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.8 Σελίδα δημιουργία διάλεξης

Ο καθηγητής για να δημιουργήσει διάλεξη εισάγει τον τίτλο και την ερώτηση αξιολόγησης γνώσεων (Εικόνα 53) ο κώδικας για την διαδικασία αυτή βρίσκεται στις εικόνες 55 και 55.



Εικόνα 53. Φόρμα για τη δημιουργία διάλεξης και ερώτησης αξιολόγησης.

```
@blueprint.route('teacher/create_lecture/<int:course_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def create_lecture(course_id):
        form = LectureForm(request.form)
        course = Course.query.get(course_id)
        #new_lecture = None
        if request.method == 'POST' and form.validate():

          lecture_title = form.title.data
          lecture  = Lecture.query.filter_by(title = lecture_title).first()
          question_text = form.question_text.data
          lecture_question = Lecture(course_id = course_id , title = lecture_title, question_text = question_text)
          db.session.add(lecture_question)
          db.session.commit()


          return redirect(url_for('teacherhome_blueprint.create_lecture', course_id = course_id))
        #if request.method == 'GET':

        return render_template('teacherhome/questionaireteacher.html', form=form,course =course)
```

Εικόνα 54. URL για την δημιουργία διάλεξης.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
{% extends "teacherhome/layout.html" %}
{% block metatags %} {% endblock %}
{% block content %}
<div class="container-fluid">
    <div class="row">


        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="documentsteacher">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = course.id)}}">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Create Questionaires and Lectures</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.post_grades', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>

        <div class="col-md-9">
            <div class="jumbotron">


                <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
                <hr class="my-4">
                <div class="container"><p class="text-center"><strong><u>Create Lectures and questionaires</u></strong></p></div>

                <div class="container">
                    <div class="row">
                        <div class="col-12">
                            <h2>Create Lecture</h2>
```


```
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.post_grades', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>

        <div class="col-md-9">
            <div class="jumbotron">


                <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
                <hr class="my-4">
                <div class="container"><p class="text-center"><strong><u>Create Lectures and questionaires</u></strong></p></div>

                <div class="container">
                    <div class="row">
                        <div class="col-12">
                            <h2>Create Lecture</h2>
```

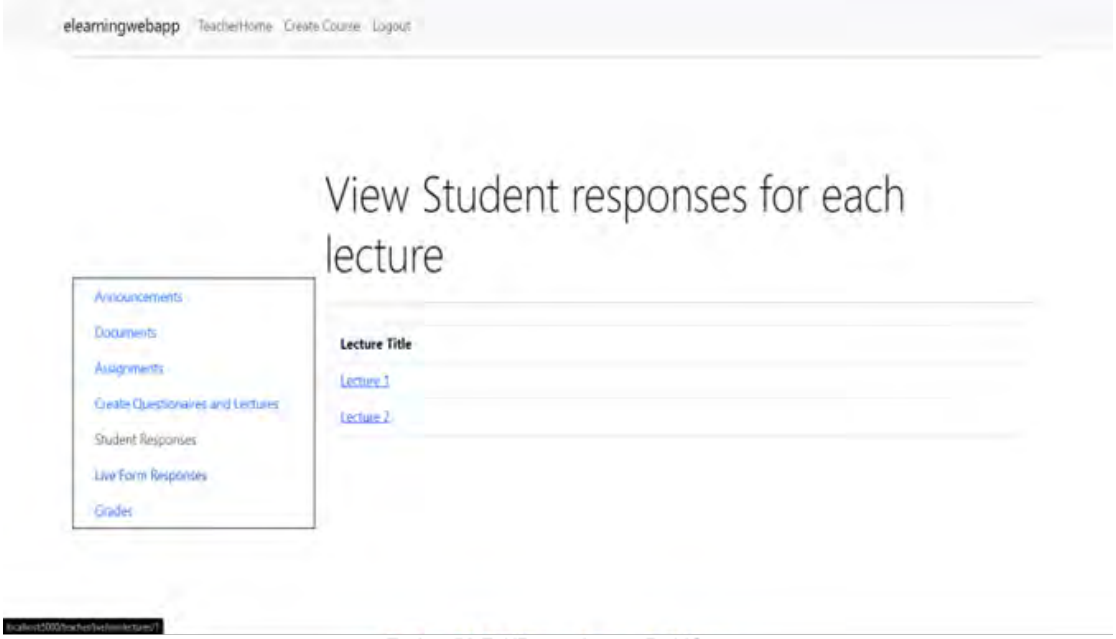

```
                            <form method="POST">
                                {{ form.hidden_tag() }}
                                {{ form.csrf_token() }}
                                <div>
                                    {{ form.title.label() }}
                                    {{ form.title(class="form-control") }}
                                </div>
                                <div>
                                    {{ form.question_text.label() }}
                                    {{ form.question_text(class="form-control") }}
                                </div>

                                <button type="submit">Create Lecture</button>
                            </form>

                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
```

Εικόνα 55.questionairteacher.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.9 Εμφάνιση αποτελεσμάτων ερώτησης αξιολόγησης

Παρακάτω εμφανίζεται η σελίδα εμφάνισης αποτελεσμάτων και στατιστικών δεδομένων ερώτησης αξιολόγησης (Εικόνα 59). Επειδή τα σχετικά δεδομένα και οι ερωτήσεις σχετίζονται με την κάθε διάλεξη αρχικά εμφανίζόνται όλες οι διαλέξεις του μαθήματος που κατεθύνουν στα αντίστοιχα αποτελέσματα (Εικόνα 56).



Εικόνα 56.Σελίδα εμφάνισης διαλέξεων

Στις εικόνες 57 και 58 παρουσιάζεται ο κώδικας για την ανάκτηση και προβολή των διαλέξεων του μαθήματος και στις εικόνες 60-61 τα τμήματα κώδικα για την προβολή στατιστικών δεδομένων και απαντήσεων.

```
@blueprint.route('teacher/lectures/<int:course_id>')
@roles_accepted('admin', 'teacher')
def lectures(course_id):
    course = Course.query.get(course_id)
    lectures = Lecture.query.filter_by(course_id = course_id).all()
    return render_template('teacherhome/lecturesteacher.html',  lectures = lectures, course =course )
```

Εικόνα 57.URL για την ανάκτηση όλων των διαλέξεων του μαθήματος.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
{% extends "teacherhome/layout.html" %}
{% block content %}
<div class="container-fluid">
  <div class="row">

    <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
        <ul class="nav flex-column">
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = course.id)}}">Assignments</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires and Lectures</a>
            </li>
            <li class="nav-item">
                <a class="nav-link disabled" href="#">Student Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
            </li>
        </ul>
    </div>

    <div class="col-md-9">
        <div class="jumbotron">
            <h1 class="display-4">View Student responses for each lecture</h1>

            <hr class="my-4">
            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <table class="table table-bordered">
                            <thead>
                                <tr>
                                    <th scope="col">Lecture Title</th>
```

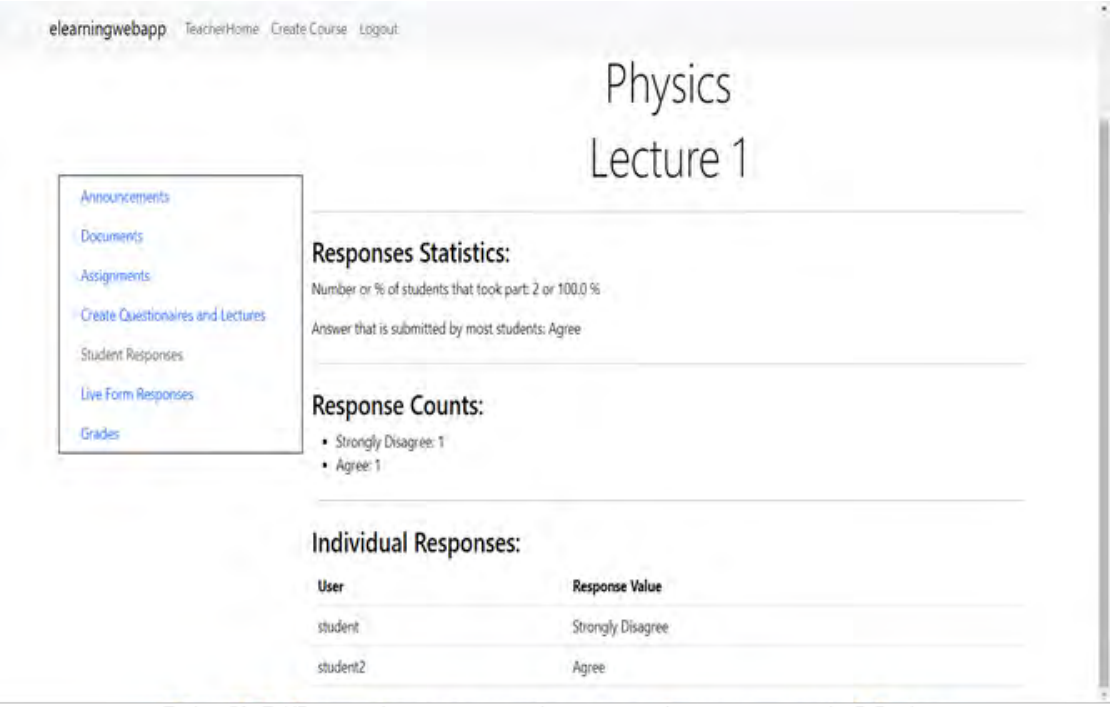

```
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
            </li>
        </ul>
    </div>

    <div class="col-md-9">
        <div class="jumbotron">
            <h1 class="display-4">View Student responses for each lecture</h1>

            <hr class="my-4">
            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <table class="table table-bordered">
                            <thead>
                                <tr>
                                    <th scope="col">Lecture Title</th>
```


```
                                </tr>
                            </thead>
                            <tbody>
                                {%for lecture in lectures%}
                                <tr>
                                    <td><a href="{{url_for('teacherhome_blueprint.questionaire_responses', course_id = course.id, lecture_id = lecture.id)}}

                                </tr>
                                {%endfor%}

                            </tbody>
                        </table>
                    </div>
                </div>
            </div>

        </div>
</div>
{% endblock %}
```

Εικόνα 58. lecturesteacher.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Στην παρακάτω σελίδα (Εικόνα 59) τα δεδομένα κατατάσσονται ως εξής: Στο πεδίο των στατιστικών (Responses Statistics) εμφανίζεται το ποσοστό του συνόλου των εγεγγραμένων φοιτητών που απάντησαν στην ερώτηση αξιολόγησης και η απάντηση που έχει επιλεχθεί από την πλειοψηφία. Στο επόμενο πεδίο (Response Counts) είναι οι απαντήσεις που επιλέχθηκαν από τους φοιτητές μαζί με τον αριθμό των ατόμων που τις επέλεξαν και στο τελευταίο πεδίο εμφανίζονται όλες οι απαντήσεις και τα ονόματα χρήστη των συμμετέχοντων.



Εικόνα 59. Σελίδα απαντήσεων των φοιτητών και των αντίστοιχων στατιστικών δεδομένων

```
@blueprint.route('teacher/lectures/<int:course_id>')
@roles_accepted('admin', 'teacher')
def lectures(course_id):
    course = Course.query.get(course_id)
    lectures = Lecture.query.filter_by(course_id = course_id).all()
    return render_template('teacherhome/lecturesteacher.html',  lectures = lectures, course =course )


@blueprint.route('/questionaire_responses/<int:course_id>/<int:lecture_id>')
@roles_accepted('admin', 'teacher')
def questionaire_responses(course_id, lecture_id):
    course = Course.query.get(course_id)
    lecture = Lecture.query.filter_by(course_id=course_id, id=lecture_id).first()
    responses = Responses.query.filter_by(course_id=course_id, lecture_id=lecture_id).all()
    enrollments = Enrollment.query.filter_by(course_id=course_id).all()
    percentage_of_students = None
    response_values = [response.response_value for response in responses if response.response_value is not None]
    response_count = len(response_values)
    enrolled_students = len(enrollments)
    if enrolled_students == 0:
        flash('No enrolled students yet!','info')
    else:
        percentage_of_students  = (100 * response_count)/enrolled_students
    #mean_response = sum(response_values) / response_count if response_count > 0 else 0
    mode_response = max(set(response_values), key=response_values.count) if response_count > 0 else 0
    option_counts = Counter(response_values)
    return render_template('teacherhome/questionaireresponses.html',course = course, lecture=lecture, responses=responses,
                           response_count=response_count, mode_response=mode_response,option_counts =option_counts,percentage_of_students  = percentage
```

Εικόνα 60.URL για την εμφάνιση των απαντήσεων των φοιτητών και των αντίστοιχων στατιστικών.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

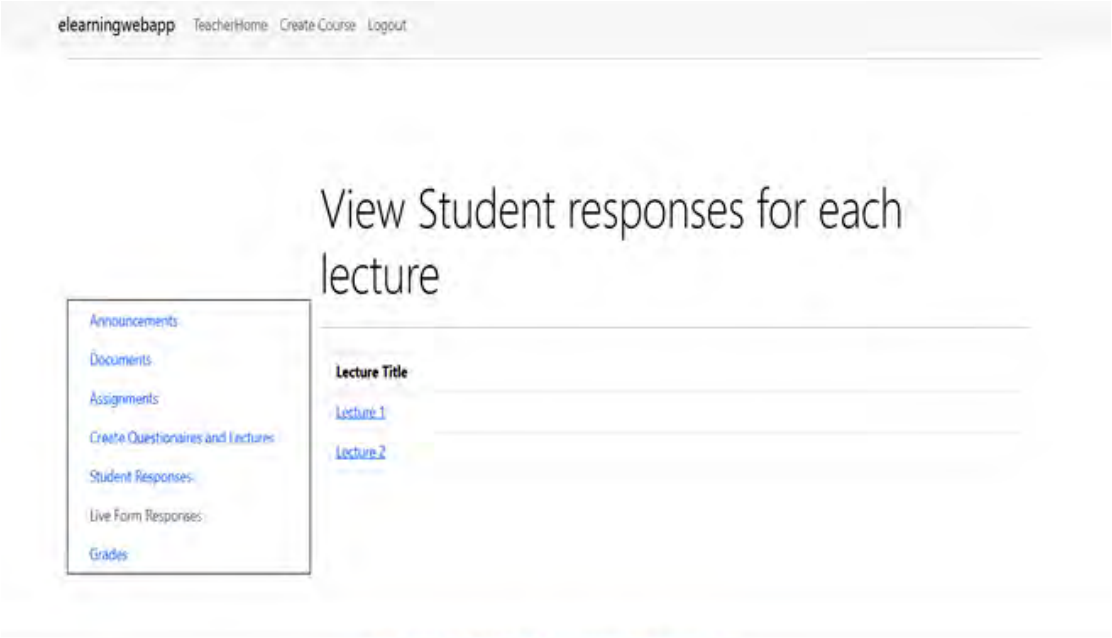
```
{% extends "teacherhome/layout.html" %}
{% block content %}
<div class="container-fluid">
    <div class="row">

        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcements/teacher', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = course.id)}}">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires and Lectures</
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Student Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.liveformlectures', course_id = course.id)}}">Live Form Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>
        <div class="col-md-9">
            <div class="jumbotron">
                <div class="col-md-12">
                    <h1 class="display-4" align="center">{{ course.title }}</h1>
                    <h2 class="display-4" align="center">{{ lecture.title }}</h2>
                    <hr class="my-4">
                    <h3>Responses Statistics:</h3>
                    <p>Number or % of students that took part: {{ response_count }} or {{ percentage_of_students }} %</p>

                    <p>Answer that is submitted by most students: {{ mode_response }}</p>

                    <hr class="my-4">
                    <h3>Response Counts:</h3>
                    <ul>
                        {% for option, count in option_counts.items() %}
                    </li>
            </ul>
        </div>
        <div class="col-md-9">
            <div class="jumbotron">
                <div class="col-md-12">
                    <h1 class="display-4" align="center">{{ course.title }}</h1>
                    <h2 class="display-4" align="center">{{ lecture.title }}</h2>
                    <hr class="my-4">
                    <h3>Responses Statistics:</h3>
                    <p>Number or % of students that took part: {{ response_count }} or {{ percentage_of_students }} %</p>

                    <p>Answer that is submitted by most students: {{ mode_response }}</p>

                    <hr class="my-4">
                    <h3>Response Counts:</h3>
                    <ul>
                        {% for option, count in option_counts.items() %}
                        <li>{{ option }}: {{ count }}</li>
                        {% endfor %}
                    </ul>
                    <hr class="my-4">
                    <h3>Individual Responses:</h3>
                    <table class="table">
                        <thead>
                            <tr>
                                <th>User</th>
                                <th>Response Value</th>
                            </tr>
                        </thead>
                        <tbody>
                            {% for response in responses %}
                            <tr>
                                <td>{{ response.user.username }}</td>
                                <td>{{ response.response_value }}</td>
                            </tr>
                            {% endfor %}
                        </tbody>
                    </table>
                </div>
            </div>
        </div>
        {% endblock %}
```

Εικόνα 61. questionaireresponses.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.2.10 Εμφάνιση στατιστικών δεδομένων δια δραστικού εργαλείου

Παρακάτω εμφανίζεται η σελίδα εμφάνισης στατιστικών δεδομένων διαδραστικού εργαλείου (Εικόνα 65). Παρομίως με προηγουμένως, επειδή τα σχετικά δεδομένα και οι απορίες/σχόλια σχετίζονται με την κάθε διάλεξη πρώτα εμφανίζονται όλες οι διαλέξεις του μαθήματος που οδηγούν στα αντίστοιχα αποτελέσματα (Εικόνα 62).



Εικόνα 62. Σελίδα εμφάνισης διαλέξεων

Στην εικόνα 63 και 64 παρουσιάζεται ο κώδικας για την ανάκτηση και προβολή των διαλέξεων του μαθήματος, ενώ στις εικόνες 66-70 τα τμήματα κώδικα για την προβολή στατιστικών δεδομένων, αποτελεσμάτων και την δημιουργία πλήκτρων ενεργοποίησης/απενεργοποίησης του δια δραστικού εργαλείου.

```
@blueprint.route('/liveformlectures/<int:course_id>')
#@login_required
@roles_accepted('admin', 'student')
def liveformlectures(course_id):

    course = Course.query.get(course_id)
    assignment = Assignment.query.get(course_id)
    lectures = Lecture.query.filter_by(course_id = course_id).all()
    lecture = Lecture.query.get(course_id)
    return render_template('home/liveformlectures.html',  lectures = lectures, course =course,assignment =assignment, lecture = lecture )
```

Εικόνα 63. URL για ανάκτηση των διαλέξεων του μαθήματος.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

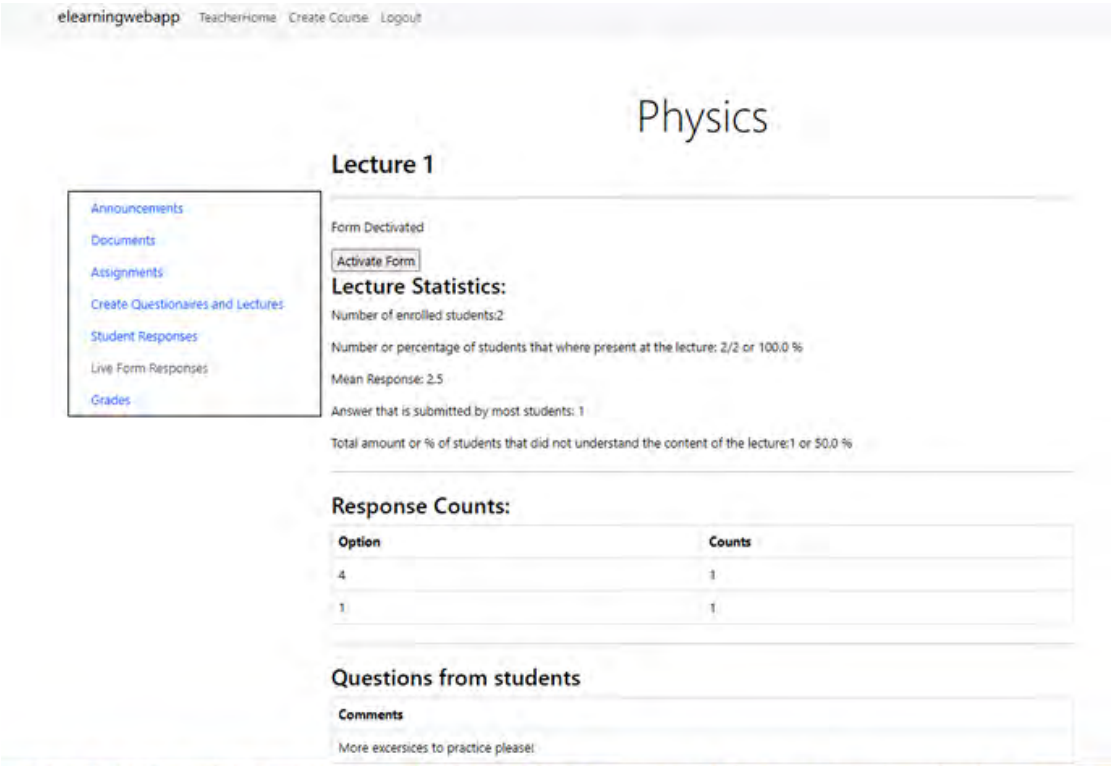
```
{% extends "teacherhome/layout.html" %}
{% block content %}
<div class="container-fluid">
  <div class="row">

    <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
        <ul class="nav flex-column">
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.mkassignment', course_id = course.id)}}">Assignments</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires and Lectures</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link disabled" href="#">Live Form Responses</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
            </li>
        </ul>
    </div>


    <div class="col-md-9">
        <div class="jumbotron">
            <h1 class="display-4">View Student responses for each lecture</h1>

            <hr class="my-4">
            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <table class="table table-bordered">
                            <thead>
                                <tr>
                                    <th scope="col">Lecture Title</th>
```

```
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
            </li>
        </ul>
    </div>


    <div class="col-md-9">
        <div class="jumbotron">
            <h1 class="display-4">View Student responses for each lecture</h1>

            <hr class="my-4">
            <div class="container">
                <div class="row">
                    <div class="col-12">
                        <table class="table table-bordered">
                            <thead>
                                <tr>
                                    <th scope="col">Lecture Title</th>


                                </tr>
                            </thead>
                            <tbody>
                                {%for lecture in lectures%}
                                <tr>
                                    <td><a href="{{url_for('teacherhome_blueprint.liveform_responses', course_id = course.id, lecture_id = lecture.id)}}">{{


                                </tr>

                                {%endfor%}

                            </tbody>
                        </table>
                    </div>
                </div>
            </div>

        </div>
</div>
{% endblock %}
```

Εικόνα 64. liveformlectures.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Στην Εικόνα 65 που προβάλλονται τα στατιστικά αποτελέσματα, τα δεδομένα κατανέμονται ως εξής: στο πεδίο των στατιστικών (Lecture Statistics) εμφανίζεται το σύνολο των εγγεγραμμένων στο μάθημα φοιτητών και το ποσοστό συμμετοχής τους στην διάλεξη, ο μέσος βαθμός αξιολόγησης από τον διδασκόμενο, η βαθμολογία που έχει επιλεχθεί από την πλειοψηφία των συμμετεχόντων και τέλος το ποσοστό των παρόντων στην διάλεξη που δήλωσαν ότι δεν έχουν κατανοήσει το περιεχόμενο του μαθήματος εκείνη την στιγμή. Στο επόμενο πεδίο (Response Counts) είναι οι βαθμολογίες των φοιτητών με τον αριθμό των διδασκομένων που τις επέλεξαν και στο τελευταίο πεδίο εμφανίζονται οι απορίες και τα σχόλια των φοιτητών. Να σημειωθεί ότι ο διδάσκοντας επιλέγει πότε θα έχουν πρόσβαση οι φοιτητές στο δια δραστικό εργαλείο ενεργοποιώντας ή απενεργοποιώντας το εργαλείο (πλήκτρο Activate Form).



Εικόνα 65. Σελίδα προβολής αποτελεσμάτων αξιολόγησης διάλεξης, στατιστικών δεδομένων και ερωτήσεων/σχολίων φοιτητών

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
@blueprint.route('teacher/activate_liveform/<int:course_id>/<int:lecture_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def activate_liveform(course_id,lecture_id):
    # course = Course.query.get(course_id)
    lecture = Lecture.query.get(lecture_id)

    lecture.lecture_activated = True
    db.session.commit()
    flash('Form activated successfully !', 'success')
    return redirect(url_for('teacherhome_blueprint.liveform_responses',course_id = course_id,lecture_id = lecture_id))
```

Εικόνα 66. Συνάρτηση για το πλήκτρο ενεργοποίησης της φόρμας.

```python
@blueprint.route('teacher/deactivate_liveform/<int:course_id>/<int:lecture_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'teacher')
def deactivate_liveform(course_id,lecture_id):
    course = Course.query.get(course_id)
    lecture = Lecture.query.get(lecture_id)
    lecture.lecture_activated = False
    db.session.commit()
    flash('Form deactivated successfully !', 'success')
    return redirect(url_for('teacherhome_blueprint.liveform_responses', course_id=course_id, lecture_id=lecture_id))
```

Εικόνα 67. Συνάρτηση για το πλήκτρο απενεργοποίησης της φόρμας

```python
class ActivateForm(FlaskForm):
    lecture_id = HiddenField('Lecture ID', validators=[DataRequired()])
    submit = SubmitField('Submit')


class DeactivateForm(FlaskForm):
    lecture_id = HiddenField('Lecture ID', validators=[DataRequired()])
    submit = SubmitField('Submit')
```

Εικόνα 68. Δημιουργία πλήκτρων ενεργοποίησης (Activate Form) και απενεργοποίησης (Deactivate Form) φόρμας.

```python
@blueprint.route('teacher/liveform_responses/<int:course_id>/<int:lecture_id>')
@roles_accepted('admin', 'teacher')
def liveform_responses(course_id,lecture_id):
    course = Course.query.get(course_id)
    # lecture = Lecture.query.get(course_id)
    feedback_from_students = FeedbackComment.query.filter_by(course_id = course_id,lecture_id = lecture_id).all()
    lecture = Lecture.query.filter_by(course_id=course_id, id=lecture_id).first()
    activateform = ActivateForm()
    deactivateform = DeactivateForm()
    liveresponses =  Lectureresponses.query.filter_by(course_id=course_id, lecture_id = lecture_id).all()
    enrollments = Enrollment.query.filter_by(course_id=course_id).all()
    percentage_of_students = None
    response_values = [response.response_value for response in liveresponses if response.response_value is not None]
    response_count = len(response_values)
    enrolled_students = len(enrollments)
    if enrolled_students == 0:
        flash('No enrolled students yet!','info')
    else:
        percentage_of_students  = (100 * response_count)/enrolled_students
    mean_response = sum(response_values) / response_count if response_count > 0 else 0
    mode_response = max(set(response_values), key=response_values.count) if response_count > 0 else 0
    low_responses_count = sum(1 for response in response_values if response <= 3)
    low_responses_percentage = (100 * low_responses_count)/response_count if response_count > 0 else 0
    option_counts = Counter(response_values)
    return render_template('teacherhome/liveformresponses.html',course = course, liveresponses=liveresponses,
                           response_count=response_count, mode_response=mode_response,mean_response =mean_response,option_counts =option_counts, lecture
```

Εικόνα 69. URL για την εμφάνιση των αποτελεσμάτων, των στατιστικών και των απορριών των φοιτητών.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
{% extends "teacherhome/layout.html" %}
{% block content %}
<div class="container-fluid">
    <div class="row">


        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.announcementsteacher', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.upload_document', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.addassignment', course_id = course.id)}}">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.create_lecture', course_id = course.id)}}">Create Questionaires and Lectures
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.lectures', course_id = course.id)}}">Student Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Live Form Responses</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('teacherhome_blueprint.pass_grades', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>
        <div class="col-md-9">
            <div class="jumbotron">
                <div class="col-md-12">
                    <h1 class="display-4" align="center">{{ course.title }}</h1>
                    <h2>{{lecture.title}}</h2>
                    <hr class="my-4">
                    {%if lecture.lecture_activated == False%}
                    <p>Form Dectivated</p>
                    <form method="POST" action="{{ url_for('teacherhome_blueprint.activate_liveform', course_id=course.id, lecture_id = lecture.id) }}">
                        {{ activateform.csrf_token }}
                        {{ activateform.hidden_tag() }}

                        <input type="submit" value="Activate Form" name="submit">

                        <input type="submit" value="Activate Form" name="submit">

                    </form>
                    {%endif%}

                    {%if lecture.lecture_activated == True%}
                    <p>Form Activated</p>
                    <form method="POST" action="{{ url_for('teacherhome_blueprint.deactivate_liveform', course_id=course.id, lecture_id = lecture.id) }}">
                        {{ deactivateform.csrf_token }}
                        {{ deactivateform.hidden_tag() }}

                        <input type="submit" value="Deactivate Form" name="submit">

                    </form>
                    {%endif%}

                    <h3>Lecture Statistics:</h3>
                    <p>Number of enrolled students:{{enrolled_students}}</p>
                    <p>Number or percentage of students that where present at the lecture: {{ response_count }}/{{enrolled_students}} or {{percentage_of_stu
                    <p>Mean Response: {{ mean_response }}</p>
                    <p>Answer that is submitted by most students: {{ mode_response }}</p>
                    <p>Total amount or % of students that did not understand the content of the lecture:{{low_responses_count}} or {{low_responses_percentag
                    <hr class="my-4">
                    <h3>Response Counts:</h3>
                    <table class="table table-bordered">
                        <thead><tr><th scope="col">Option</th><th scope="col">Counts</th></tr></thead>
                        <tbody>


                            {% for option, count in option_counts.items() %}
                            <tr>
                                <td>{{ option }}</td>
                                <td> {{ count }}</td>
                            </tr>
                            {% endfor %}


                        </tbody>
                    </table>
                    <hr class="my-4">
                    <h3>Questions from students</h3>

                    <table class="table table-bordered">
                        <thead><tr><th scope="col">Comments</th></tr></thead>
```

Εικόνα 70. liveformresponses.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
<p>Number of enrolled students:{{enrolled_students}}</p>
<p>Number or percentage of students that where present at the lecture: {{ response_count }}/{{enrolled_students}} or {{percentage_of_stu
<p>Mean Response: {{ mean_response }}</p>
<p>Answer that is submitted by most students: {{ mode_response }}</p>
<p>Total amount or % of students that did not understand the content of the lecture:{{low_responses_count}} or {{low_responses_percentag
<hr class="my-4">
<h3>Response Counts:</h3>
<table class="table table-bordered">
    <thead><tr><th scope="col">Option</th><th scope="col">Counts</th></tr></thead>
    <tbody>


        {% for option, count in option_counts.items() %}
        <tr>
            <td>{{ option }}</td>
            <td> {{ count }}</td>
        </tr>
        {% endfor %}


    </tbody>
</table>
<hr class="my-4">
<h3>Questions from students</h3>

<table class="table table-bordered">
    <thead><tr><th scope="col">Comments</th></tr></thead>
    <tbody>


        {%for feedback in feedback_from_students%}
        <tr>
            <td>{{ feedback_comment }}</td>

        </tr>

        {%endfor%}

    </tbody>
</table>


{% endblock %}
```

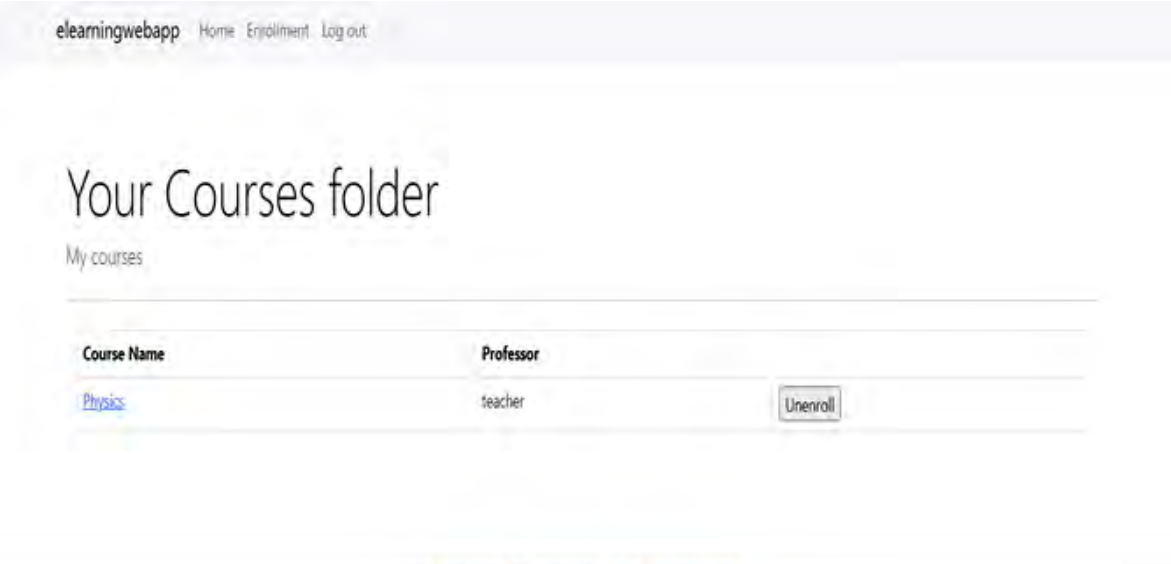
Εικόνα 71. liveformresponses.html(συνέχεια)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.3 ΔΙΕΠΑΦΗ ΦΟΙΤΗΤΗ

6.3.1 Αρχική σελίδα Φοιτητή

Αμέσως μετά την σύνδεση, εμφανίζεται η αρχική σελίδα του φοιτητή στην οποία εμφανίζονται όλα τα μαθήματα που παρακολουθεί καθώς και το πλήκτρο απεγγραφής από το μάθημα. Στις Εικόνες 73-75 εμφανίζεται ο κώδικας που ευθύνεται για το πλήκτρο απεγγραφής, την ανάκτηση των μαθημάτων που έχει εγγραφεί φοιτητής και την απεικόνιση της εν λόγω σελίδας.



Εικόνα 72. Αρχική σελίδα φοιτητή

```
class Unenroll(FlaskForm):
    enrollment_id = HiddenField('Enrollment  ID', validators=[DataRequired()])
    submit = SubmitField('Submit')
```

Εικόνα 73. Φόρμα για απεγγραφή (Unenroll) από μάθημα.

```
@blueprint.route('/index')
#@login_required
@roles_accepted('admin', 'student')
def index():
    form = Unenroll()
    if current_user.is_authenticated:
        enrolled_courses = Course.query.join(Enrollment).filter(Enrollment.user_id == current_user.id).all()

    else:
        enrolled_courses = []
    return render_template('home/index.html', segment='index',form=form,enrolled_courses=enrolled_courses)
```

Εικόνα 74. URL για την εμφάνιση των μαθημάτων στην αρχική σελίδα του φοιτητή.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

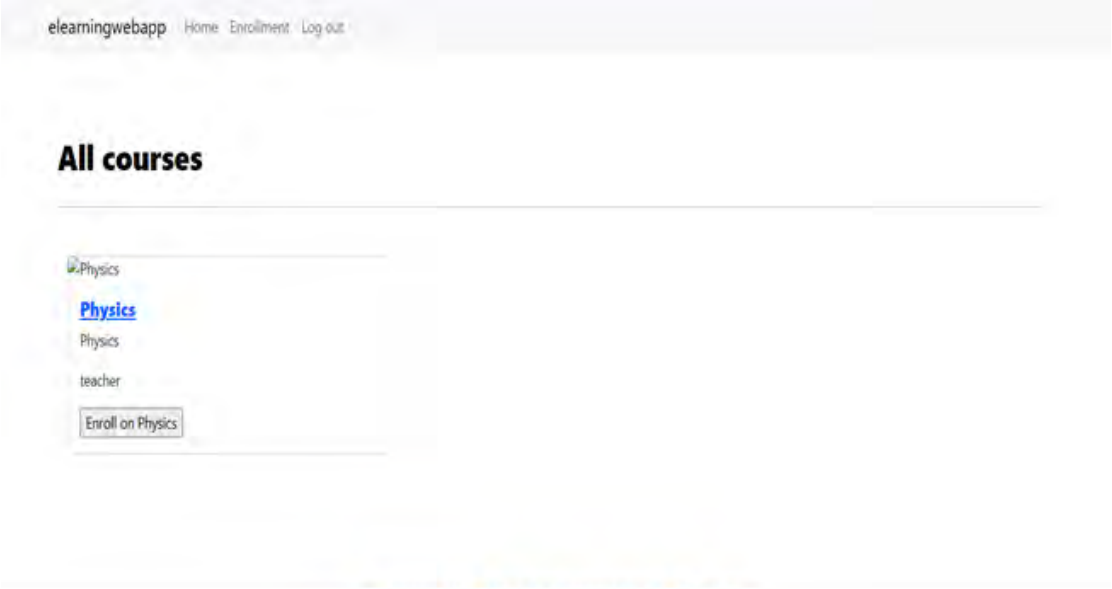
```html
<div class="jumbotron">

    <h1 class="display-4"> Your Courses folder  </h1>
    <p class="lead">My courses</p>


    <hr class="my-4">
    <div class="container">
        <div class="row">
            <div class="col-12">
                <table class="table table-bordered">
                    <thead>
                        <tr>
                            <th scope="col">Course Name</th>
                            <th scope="col">Professor</th>
                            <th scope="col"></th>

                        </tr>
                    </thead>

                    <tbody>
                        {% for  course in enrolled_courses %}
                        <tr>


                            <td><a href="{{url_for('home_blueprint.coursedetails', course_id = course.id)}}">{{course.title}}</a></td>
                            <td>{{course.professor}}</td>

                            <form method="POST" action="{{ url_for('home_blueprint.unenroll_course', course_id=course.id) }}">
                                {{ form.csrf_token }}
                                {{ form.hidden_tag() }}
                            <td>
                                <input type="submit" value="Unenroll" name="submit">
                            </td>
                            </form>



                        </tr>
                        {% endfor %}
                    </tbody>
                </table>
            </div>
```

Εικόνα 75. index.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.3.2 Εγγραφή σε μάθημα

Παρακάτω εμφανίζονται όλα τα μαθήματα που διδάσκονται για να εγγραφεί ο φοιτητής (Εικόνα 76). Τα τμήματα κώδικα που παρουσιάζονται στις εικόνες 77-79 είναι υπεύθυνα για την ανάκτηση και εμφάνιση των μαθημάτων που διδάσκονται και την διαδικασία της εγγραφής.



Εικόνα 76. .Σελίδα εγγραφής σε μάθημα

```
@blueprint.route('/allcourses')
@roles_accepted('admin', 'student')
def all_courses():
    all_courses = Course.query.all()
    form = Enroll()
    return render_template('home/allcourses.html', all_courses=all_courses, form=form)
```

Εικόνα 77. URL για την εμφάνιση όλων των μαθημάτων που υπάρχουν στην βάση δεδομένων καθώς και της φόρμας εγγραφής σε αυτά.

```
@blueprint.route('/enroll_course/<int:course_id>', methods=['POST'])
@roles_accepted('admin', 'student')
def enroll_course(course_id):
    if request.method == 'POST':
        if Enrollment.query.filter_by(user_id=current_user.id, course_id=course_id).first():
            flash('You are already enrolled in this course', 'info')
            return redirect(url_for('home_blueprint.all_courses', course_id=course_id))
    enrollment = Enrollment(user_id=current_user.id, course_id=course_id)
    db.session.add(enrollment)
    db.session.commit()
    flash('You enrolled successfully','success')
    return redirect(url_for('home_blueprint.all_courses',course_id = course_id))
```

Εικόνα 78. URL για την διαδικασία εγγραφής.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

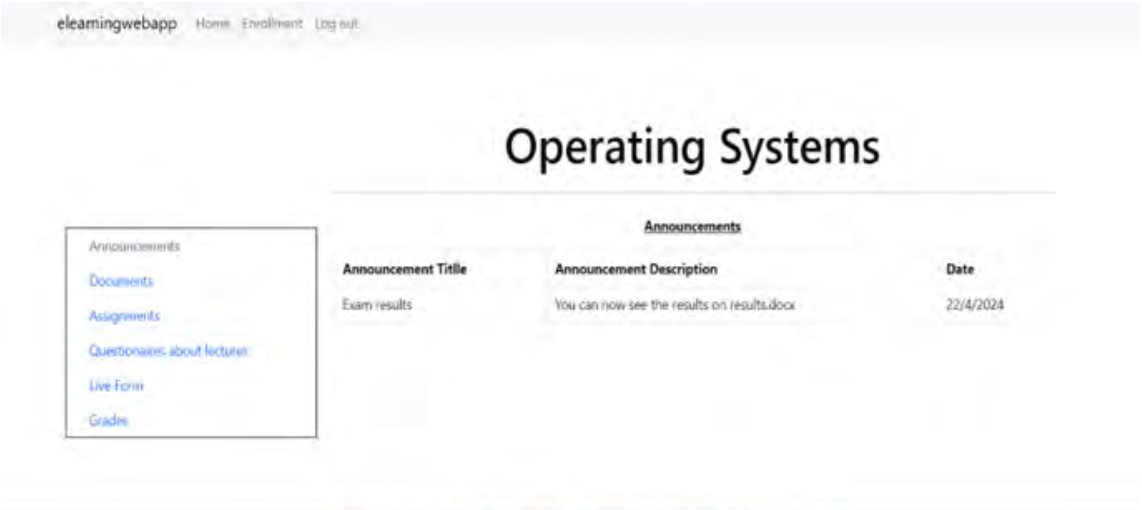
```
{% extends "home/layout.html" %}
{% block content %}
<p class="lead" style="font-weight:800; font-size:35px;">All courses</p>
<hr class="my-4">
<div class="container mt-5">
    <div class="row">

        {% for course in all_courses %}


        <form method="POST" action="{{ url_for('home_blueprint.enroll_course', course_id=course.id) }}">
            {{ form.hidden_tag() }}
            {{ form.course_id(value=course.id) }}
            <div class="col-md-4 mb-4">
                <div class="card">
                    <img src="{{ course.photo }}" class="card-img-top" alt="{{ course.title }}">
                    <div class="card-body">
                        <h5 class="card-title" style="font-weight:800"><a href="/coursedetails"> {{ course.title }}</a></h5>
                        <p class="card-text">{{ course.title }}</p>
                        <p class="card-text">{{ course.proffessor }}</p>
                        <input type="submit" value="Enroll on {{ course.title }}" name="submit_{{ course.id }}">

                    </div>
                </div>
            </div>

        </form>



        {% endfor %}



    </div>
</div>
{% endblock %}
```

Εικόνα 79. allcourses.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.3.3 Προβολή σελίδας ανακοινώσεων

Εδώ προβάλλονται όλες οι ανακοινώσεις του μαθήματος (Εικόνα 80). Ο κώδικας που αντιστοιχεί στην προβολή των ανακοινώσεων εμφανίζεται στις εικόνες 81 και 82.



Εικόνα 80. Σελίδα προβολής ανακοινώσεων μαθήματος

```
@blueprint.route('/announcements/<int:course_id>')
@roles_accepted('admin', 'student')
def announcements(course_id):
    course = Course.query.get(course_id)
    announcements = Announcement.query.filter_by(course_id=course_id).all()
    assignment = Assignment.query.get(course_id)
    lecture = Lecture.query.get(course_id)
    return render_template('home/announcements.html', course=course,announcements = announcements,assignment =assignment,lecture = lecture)
```

Εικόνα 81. URL για την προβολή όλων των ανακοινώσεων του μαθήματος.

```
        </div>


        <div class="col-md-9">
            <div class="jumbotron">


                <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
                <hr class="my-4">
                <div class="container"><p class="text-center"><strong><u>Announcements</u></strong></p></div>

                <div class="container">
                    <div class="row">
                        <div class="col-12">
                            <table class="table table-bordered">
                                <thead>
                                    <tr>
                                        <th scope="col">Announcement Titlle</th>
                                        <th>Announcement Description</th>
                                        <th scope="col">Date</th>
                                    </tr>
                                </thead>
                                <tbody>
                                    {%for announcement in announcements%}
                                    <tr>
                                        <td>{{announcement.announcement_title}}</td>
                                        <td>{{announcement.announcement_description}}</td>
                                        <td>{{announcement.announcement_date}}</td>
                                    </tr>
                                    {%endfor%}
                                </tbody>
                            </table>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
{% endblock %}
```

Εικόνα 82. announcements.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.3.4 Σελίδα προβολής υλικού μαθήματος

Παρακάτω είναι η σελίδα εμφάνισης εγγράφων μαθήματος (Εικόνα 83) και ο σχετικός κώδικας (Εικόνες 84-86).

Εικόνα 83. Σελίδα προβολής μαθήματος

```
@blueprint.route('/view_documents/<int:course_id>')
@roles_accepted('admin', 'student')
def view_documents(course_id):

    viewdocuments = Document.query.filter_by(course_id = course_id).all()
    course  = Course.query.get(course_id)
    lecture = Lecture.query.get(course_id)
    return render_template('home/documents.html',viewdocuments = viewdocuments, course = course,lecture = lecture)
```

Εικόνα 84. URL που ανακτά όλα τα έγγραφα που συσχετίζονται με το μάθημα.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```html
{% extends "home/layout.html" %}
{% block content %}
<div class="container-fluid">
 <div class="row">


  <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
      <ul class="nav flex-column">
          <li class="nav-item">
              <a class="nav-link" href="{{url_for('home_blueprint.announcements', course_id = course.id)}}">Announcements</a>
          </li>
          <li class="nav-item">
              <a class="nav-link disabled" href="#">Documents</a>
          </li>
          <li class="nav-item">
              <a class="nav-link" href="{{url_for('home_blueprint.view_assignment', course_id = course.id)}}">Assignments</a>
          </li>
          <li class="nav-item">
              <a class="nav-link" href="{{url_for('home_blueprint.view_lectures', course_id = course.id)}}">Questionaires about lectures</a>
          </li>
          <li class="nav-item">
              <a class="nav-link" href="{{url_for('home_blueprint.liveformlectures', course_id = course.id)}}">Live Form</a>
          </li>
          <li class="nav-item">
              <a class="nav-link" href="{{url_for('home_blueprint.view_grade', course_id = course.id)}}">Grades</a>
          </li>
      </ul>
  </div>


  <div class="col-md-9">
    <div class="jumbotron">


      <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
      <hr class="my-4">
      <div class="container"><p class="text-center"><strong><u>Documents</u></strong></p></div>

      <div class="container">
        <div class="row">
          <div class="col-12">
            <table class="table table-bordered">
              <thead>
```

Εικόνα 85. documents.html

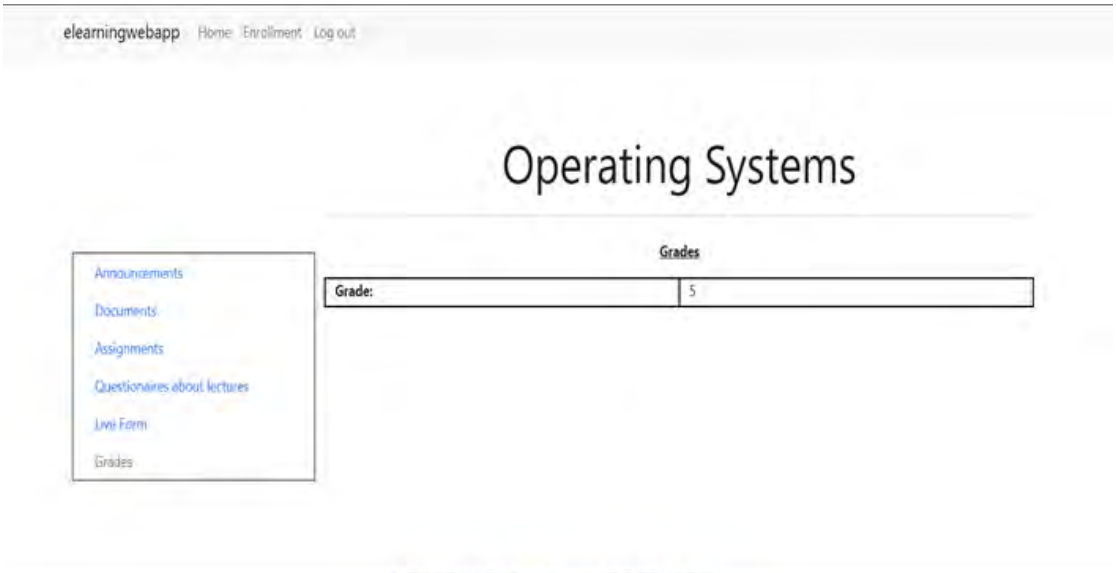
```html
  <div class="col-md-9">
    <div class="jumbotron">


      <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
      <hr class="my-4">
      <div class="container"><p class="text-center"><strong><u>Documents</u></strong></p></div>

      <div class="container">
        <div class="row">
          <div class="col-12">
            <table class="table table-bordered">
              <thead>
                <tr>

                  <th scope="col">Document</th>


                </tr>
              </thead>
              <tbody>
                  <tr>
                      {% for viewdocument in viewdocuments %}

                      <td><a href="#" target="_blank" title="{{ viewdocument.filename }}" download>{{ viewdocument.filename }}</a></td>


                      {% endfor %}
                  </tr>


              </tbody>
            </table>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
{% endblock %}
```

Εικόνα 86. documents.html(συνέχεια)

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.3.5 Προβολή σελίδας βαθμού

Στην εικόνα 87 εμφανίζεται η σελίδα στην οποία κάθε φοιτητής μπορεί να δει την βαθμολογία του στο μάθημα και στις εικόνες 88-89 παρουσιάζεται ο κώδικας για την προβολή βαθμού.



Εικόνα 87. Σελίδα προβολής βαθμού

```
{% extends "home/layout.html" %}
{% block content %}
<div class="container-fluid">
  <div class="row">

    <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
        <ul class="nav flex-column">
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.announcements', course_id = course.id)}}">Announcements</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.view_documents', course_id = course.id)}}">Documents</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.view_assignment', course_id = course.id)}}">Assignments</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.view_lectures', course_id = course.id)}}">Questionaires about lectures</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.liveformlectures', course_id = course.id)}}">Live Form</a>
            </li>
            <li class="nav-item">
                <a class="nav-link disabled" href="#">Grades</a>
            </li>
        </ul>
    </div>

    <div class="col-md-9">
      <div class="jumbotron">


        <h1 class="display-4" align="center" style="font-weight:400">{{course.title}}</h1>
        <hr class="my-4">
        <div class="container"><p class="text-center"><strong><u>Grades</u></strong></p></div>

       <div class="container" style="border:1px solid black;">
                <div class="row" >
                    {% if grade %}
                  <div class="col-sm" style="border:1px solid black; height:30px;">
                   <strong>Grade:</strong>
                  </div>
                  <div class="col-sm" style="border:1px solid black; height:30px;">
```

Εικόνα 88. grades.html

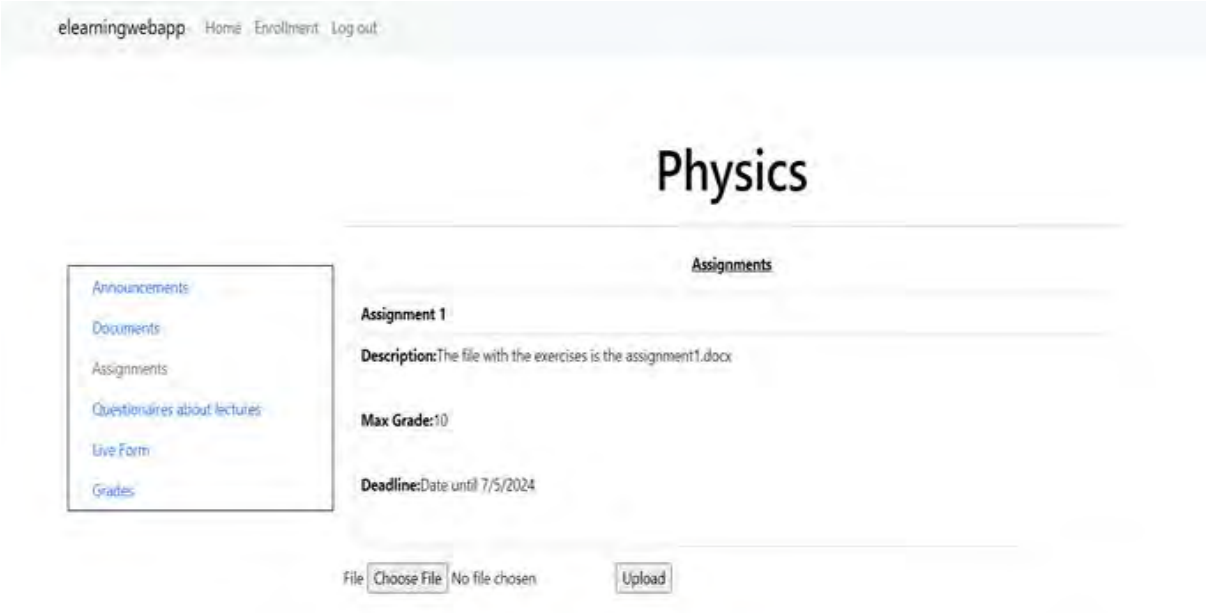
Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.view_lectures', course_id = course.id)}}">Questionaires about lectures</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="{{url_for('home_blueprint.liveformlectures', course_id = course.id)}}">Live Form</a>
            </li>
            <li class="nav-item">
                <a class="nav-link disabled" href="#">Grades</a>
            </li>
        </ul>
    </div>

    <div class="col-md-9">
      <div class="jumbotron">


        <h1 class="display-4" align="center" style="font-weight:400">{{course.title}}</h1>
        <hr class="my-4">
        <div class="container"><p class="text-center"><strong><u>Grades</u></strong></p></div>

        <div class="container" style="border:1px solid black;">
                <div class="row" >
                    {% if grade %}
                  <div class="col-sm" style="border:1px solid black; height:30px;">
                    <strong>Grade:</strong>
                  </div>
                  <div class="col-sm" style="border:1px solid black; height:30px;">
                    {{grade.grade}}
                  </div>
                    {% endif %}
                </div>
              </div>
            </div>
    </div>
  </div>
</div>
{% endblock %}
```

Εικόνα 89. grades.html(συνέχεια)

6.3.6 Προβολή εκφώνησης εργασίας και ανέβασμα αρχείου λύσης

Στην παρακάτω σελίδα (Εικόνα 90) εμφανίζεται η εκφώνηση της εργασίας με τη μέγιστη βαθμολογία, την ημερομηνία λήξης της προθεσμίας καθώς και ο κώδικας που είναι υπεύθυνος για αυτή την λειτουργία (91-93).



Εικόνα 90. Σελίδα προβολής εργασίας

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```python
@blueprint.route('/view_assignment/<int:course_id>', methods=['GET','POST'])
@roles_accepted('admin','student')
def view_assignment(course_id):

    #assignment = Assignment.query.get(assignment_id)
    assignments = Assignment.query.filter_by(course_id=course_id).all()
    form = SolutionUploadForm()
    course = Course.query.get(course_id)
    lecture = Lecture.query.get(course_id)
    if assignments is None:

        return "Assignments not found", 404


    enrollment = Enrollment.query.filter_by(user_id=current_user.id, course_id=course_id).first()

    if enrollment is None:

        return "You are not enrolled in this course", 403

    # if request.method == 'POST' and form.validate_on_submit():
    #     uploaded_file = form.file.data

    #     # Create the directory if it doesn't exist
    #     directory = '/media/studentsolutions'
    #     if not os.path.exists(directory):
    #         os.makedirs(directory)

    #     file_path = os.path.join(directory, secure_filename(uploaded_file.filename))
    #     uploaded_file.save(file_path)


    #     document = AssignmentSolution(filename_solution=secure_filename(uploaded_file.filename), course_id=course_id)
    #     db.session.add(document)
    #     db.session.commit()

    #     return redirect(url_for('teacherhome_blueprint.upload_document', course_id=course_id))

    return render_template('home/assignments.html',assignments=assignments, course = course, form = form,lecture = lecture)
```

Εικόνα 91. URL για την προβολή εκφώνησης εργασίας και της φόρμας υποβολής της λύσης (SolutionUploadForm).

```python
@blueprint.route('/upload_solution/<int:course_id>/<int:assignment_id>', methods=['GET','POST'])
@roles_accepted('admin','student')
def upload_solution(course_id,assignment_id):
    form = SolutionUploadForm()
    assignment = Assignment.query.get(assignment_id)
    if request.method == 'POST' and form.validate_on_submit():
        uploaded_file = form.solution.data


        directory = os.path.join('media', 'uploads', str(course_id))
        absolute_path = os.path.abspath(directory)

        if not os.path.exists(absolute_path):
            os.makedirs(absolute_path)



        unique_filename = uuid4().hex + '_' + uploaded_file.filename
        file_path = os.path.join(absolute_path, secure_filename(unique_filename))
        uploaded_file.save(file_path)


        solution = AssignmentSolution(filename_solution=secure_filename(unique_filename), course_id=course_id, assignment_id = assignment_id, student = current_
        db.session.add(solution)
        db.session.commit()

        return redirect(url_for('home_blueprint.view_assignment', course_id=course_id))
```

Εικόνα 92. URL για το ανέβασμα αρχείου με την λύση.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

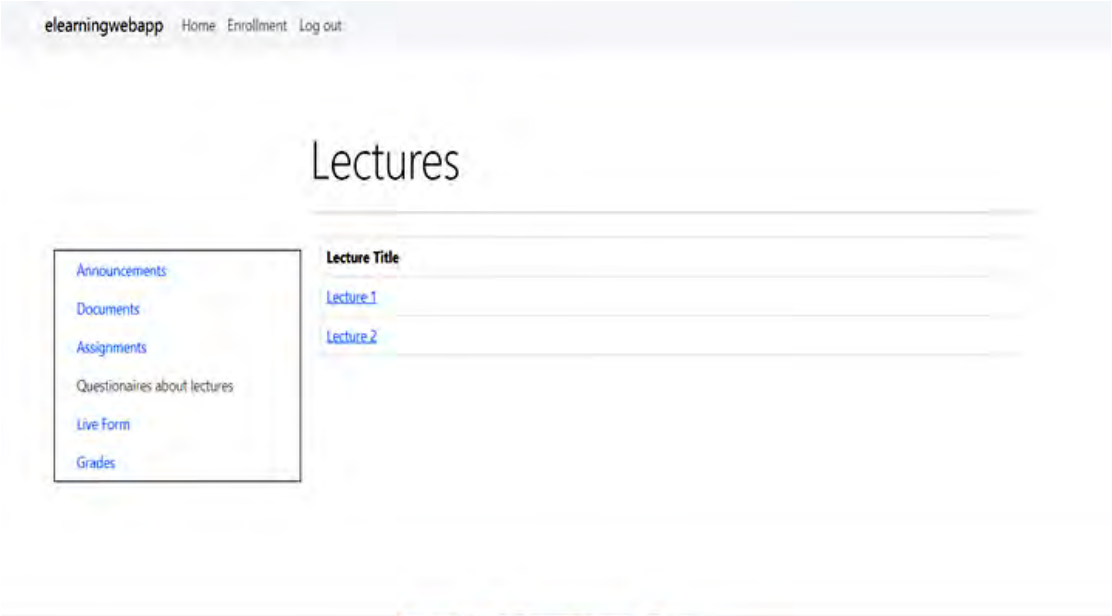
```
{% block content %}
<div class="container-fluid">
    <div class="row">


        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.announcements', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_documents', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_lectures', course_id = course.id)}}">Questionaires about lectures</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.liveformlectures', course_id = course.id)}}">Live Form</a>
                </li>

                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_grade', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>


        <div class="col-md-9">
            <div class="jumbotron">
                <h1 class="display-4" align="center" style="font-weight:600">{{course.title}}</h1>
                <hr class="my-4">
                <div class="container"><p class="text-center"><strong><u>Assignments</u></strong></p></div>

                {%for assignment in assignments%}
                <div class="container">
                    <div class="row">
                        <div class="col-12">
                            <table class="table table-bordered">
                                <thead>
                <div class="row">
                    <div class="col-12">
                        <table class="table table-bordered">
                            <thead>
                                <tr>
                                    <th scope="col">{{assignment.assignment_title}}</th>

                                </tr>
                            </thead>
                            <tbody>

                                <tr>

                                    <td>
                                        <p><strong>Description:</strong><span>{{assignment.assignment_desc}} </span></p><br>
                                        <p><strong>Max Grade:</strong><span>10</span></p><br>
                                        <p><strong>Deadline:</strong><span>Date until {{assignment.deadline_date}} </span></p><br>

                                    </td>

                                </tr>
                            </tbody>
                        </table>
                    </div>
                </div>

            </div>

            <form method="POST" action="{{ url_for('home_blueprint.upload_solution', course_id=course.id, assignment_id = assignment.id) }}" enctype="multipart/
                {{ form.csrf_token }}
                {{ form.solution.label }}
                {{ form.solution }}
                <input type="submit" name="file" value="Upload">
            </form>

        </div>
    </div>
    {%endfor%}
  </div>
</div>
{% endblock %}
```

Εικόνα 93. assignments.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

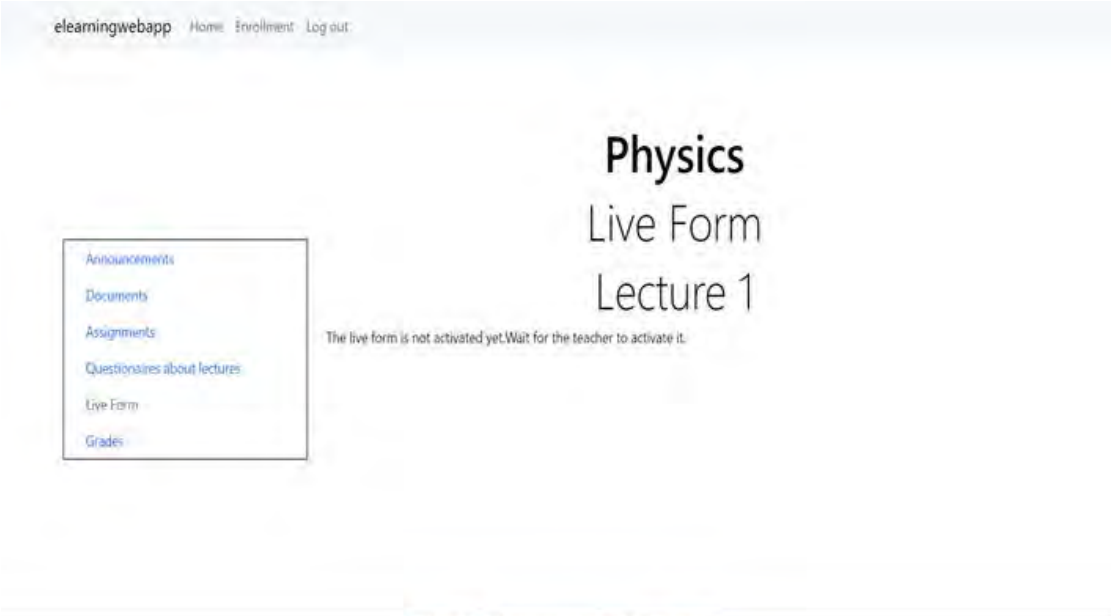
6.3.7 Προβολή ερώτησης αξιολόγησης γνώσεων και δια δραστικού εργαλείου

Στην υποενότητα αυτή εμφανίζεται το διαδραστικό εργαλείο (Εικόνα 96) και η ερώτηση αξιολόγησης γνώσεων (Εικόνα 97). Επειδή οι αξιολογήσεις, οι απορίες/σχόλια και οι απαντήσεις των φοιτητών στην ερώτηση σχετίζονται με την κάθε διάλεξη πρώτα εμφανίζονται όλες οι διαλέξεις του μαθήματος (Εικόνα 94).



Εικόνα 94. Σελίδα εμφάνισης διαλέξεων

Όταν ο καθηγητής έχει απενεργοποιήσει την πρόσβαση στο δια δραστικό εργαλείο εμφανίζεται η παρακάτω σελίδα στους φοιτητές:

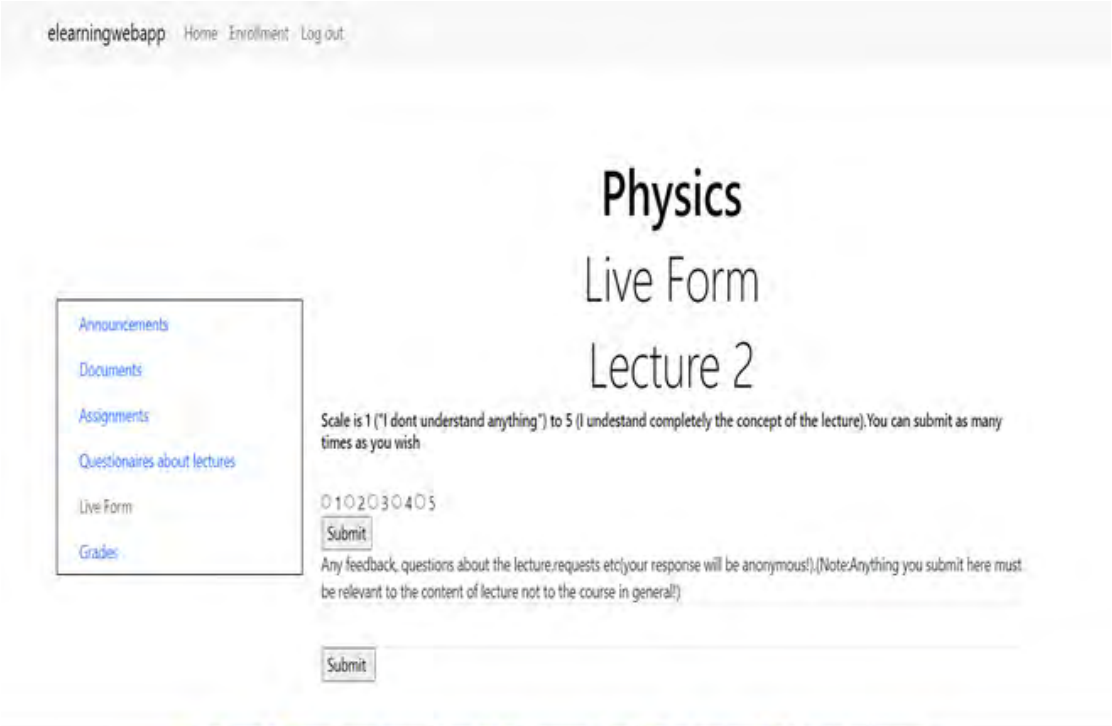


Εικόνα 95. Απενεργοποιημένη φόρμα

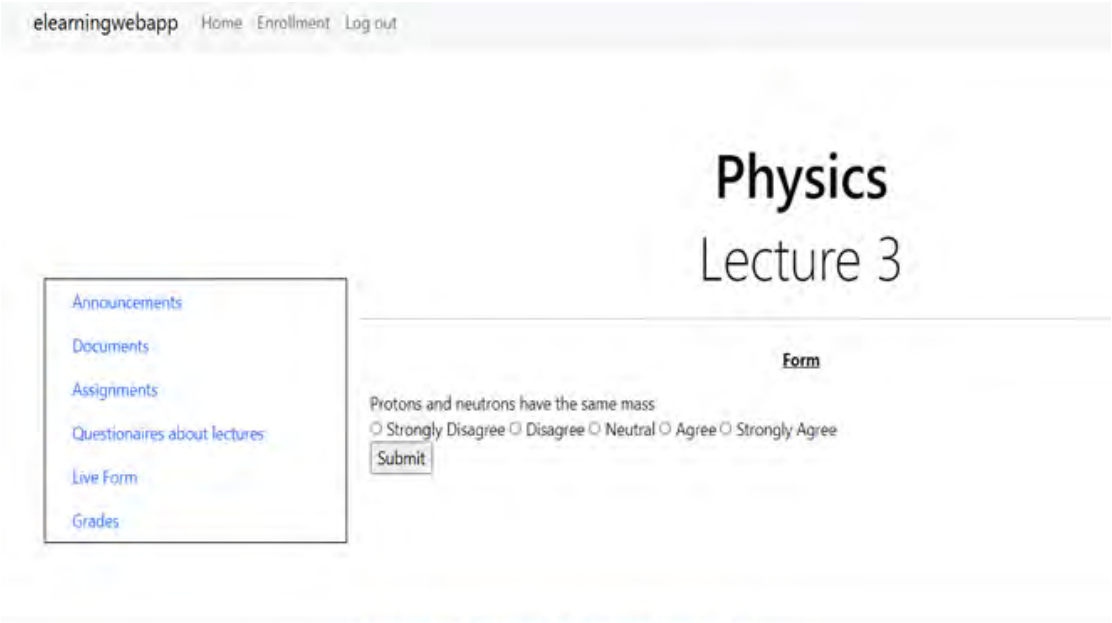
Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Το νέο εργαλείο αποτελείται από δύο μέρη:

- Την φόρμα τύπου Likert για την αξιολόγηση, σε πραγματικό χρόνο, κατανόησης περιεχομένου του μαθήματος. Η φόρμα είναι κλίμακας με τιμές που διακυμαίνονται από το 1 έως το 5 οι οποίες υποδεικνύουν αναλογικά το επίπεδο κατανόησης της διάλεξης.
- Την φόρμα υποβολής απορριών και σχολίων κατά την διάρκεια της διάλεξης.



Εικόνα 96. Εμφάνιση εργαλείου αξιολόγησης διαλέξεων και υποβολής ερωτήσεων



Εικόνα 97. Ερώτηση αξιολόγησης γνώσεων

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Στην εικόνα 98 παρουσιάζεται ο κώδικας για την ανάκτηση των διαλέξεων του μαθήματος.

```
@blueprint.route('/liveformlectures/<int:course_id>')
#@login_required
@roles_accepted('admin', 'student')
def liveformlectures(course_id):

    course = Course.query.get(course_id)
    assignment = Assignment.query.get(course_id)
    lectures = Lecture.query.filter_by(course_id = course_id).all()
    lecture = Lecture.query.get(course_id)
    return render_template('home/liveformlectures.html',  lectures = lectures, course =course,assignment =assignment, lecture = lecture )
```

Εικόνα 98. URL για προβολή των διαλέξεων.

Στις Εικόνες 99-101 εμφανίζονται τα τμήματα κώδικα για την προβολή των στατιστικών δεδομένων και των ανώνυμων ερωτήσεων.

```
@blueprint.route('/live_response/<int:course_id>/<int:lecture_id>', methods=['GET', 'POST'])
@roles_accepted('admin', 'student')
def live_response(course_id, lecture_id):
    course = Course.query.get(course_id)
    lecture = Lecture.query.get(lecture_id)
    form = FeedbackForm()
    response_value = None

    # if not lecture.lecture_activated:
    #   flash('Form is not active', 'info')
    if request.method == 'POST':
        response_value = request.form.get(f'response_{lecture.id}')
        existing_response = Lectureresponses.query.filter_by(user_id=current_user.id, course_id=course_id,lecture_id = lecture_id).first()
        if existing_response:
          db.session.delete(existing_response)
          db.session.commit()
        new_response = Lectureresponses(
            user_id=current_user.id,
            course_id=course_id,
            response_value=response_value,
            lecture_id = lecture_id)
        db.session.add(new_response)
        db.session.commit()
    if form.validate_on_submit():
        feedback_comment = form.comment_field.data
        new_comment = FeedbackComment(
            user_id=current_user.id,
            course_id=course_id,
            comment=feedback_comment,
            lecture_id = lecture_id
           )
        db.session.add(new_comment)
        db.session.commit()

    return render_template('home/liveform.html', course=course,lecture = lecture,response_value =response_value, form = form)
```

Εικόνα 99.URL για την προβολή της φόρμας ερωτήσεων (FeedbackForm) και την ανάκτηση των απαντήσεων των φοιτητών

```
class FeedbackForm(FlaskForm):
    comment_field = StringField('Any feedback, questions about the lecture,requests etc(your response will be anonymous!).(Note:Anything you submit her
    submit = SubmitField('Submit')
```

Εικόνα 100.Φόρμα για τις απορίες/σχόλια

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
{% extends "home/layout.html" %}

{% block content %}
<div class="container-fluid">
    <div class="row">


        <div class="col-md-3" style="border: 1px solid black; height: 100%; margin-top:130px;">
            <ul class="nav flex-column">
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.announcements', course_id = course.id)}}">Announcements</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_documents', course_id = course.id)}}">Documents</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_assignment', course_id = course.id)}}">Assignments</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_lectures', course_id = course.id)}}">Questionaires about lectures</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link disabled" href="#">Live Form </a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="{{url_for('home_blueprint.view_grade', course_id = course.id)}}">Grades</a>
                </li>
            </ul>
        </div>


        <div class="col-md-9">
            <div class="jumbotron">
                <h1 class="display-4" align="center" style="font-weight:600">{{course.title}} </h1>
                <h4 class="display-4" align="center" >Live Form</h4>
                <h5 class="display-4" align="center" >{{lecture.title}} </h5>

                <div class="container">
                    <div class="row">
                        <div class="col-12">



                        </div>
            <div class="container">
                <div class="row">
                    <div class="col-12">



                    </div>

                    {% if lecture.lecture_activated %}
                    <p class="h6">Scale is 1 ("I dont understand anything") to 5 (I undestand completely the concept of the lecture).You can submit as m
                    <form id="response_form" method="POST">

                        {% if course %}
                        <div>
                            <label></label><br>

                            <input type="radio" name="response_{{ lecture.id }}" value="1"> 1
                            <input type="radio" name="response_{{ lecture.id }}" value="2"> 2
                            <input type="radio" name="response_{{ lecture.id }}" value="3"> 3
                            <input type="radio" name="response_{{ lecture.id }}" value="4"> 4
                            <input type="radio" name="response_{{ lecture.id }}" value="5"> 5
                        </div>
                        {% endif %}
                        <input type="submit" value="Submit">
                    </form>

                    <form method="POST">
                        {{ form.hidden_tag() }}

                        <div>

                            {{ form.comment_field.label }}
                            {{ form.comment_field(class="form-control") }}

                        </div>
                        <input type="submit" value="Submit ">
                    </form>
                    {% else %}
                    <p>The live form is not activated yet.Wait for the teacher to activate it.</p>
                    {% endif %}
                </div>
            </div>
        </div>
```

Εικόνα 101. liveform.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

6.4. ΑΛΛΑ ΤΜΗΜΑΤΑ ΚΩΔΙΚΑ

Σε όλα τα πρότυπα HTML στην αρχή υπάρχουν τα εξής κομμάτια κώδικα:

{%extends"home/layout.html"%} (Εικόνα 102) στον κώδικα των templates του φοιτητή και *{%extends"teacherhome/layout.html"%}* (Εικόνα 103) στον κώδικα των templates του καθηγητή. Αυτά υπάρχουν γιατί όλες οι σελίδες αποτελούν επέκταση του layout. Τα δύο τελευταία είναι κομμάτια κώδικα που καθορίζουν την μορφή και την εμφάνιση που θα έχουν όλες οι σελίδες της εφαρμογής. Σε αυτά εισάγουμε το Boostrap.

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="{{ app_data['description'] }}">
    <meta name="keywords" content="{{ app_data['keywords'] }}">
    <meta name="author" content="{{ app_data['author'] }}">
    <link rel="icon" href="../../../../favicon.ico">


    <title>{{ app_data['html_title'] }}</title>

    <!-- Bootstrap  CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-GLhlTQ8iRABdZLl6O3oVMWSktQOp6b7In1Zl3/Jr59b6EGGoI1aFkw7cmDA6
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-w76AqPfDkMBDXo30jS1Sgez6pr3x5MlQ1ZAGC+nuZB+EYdgRZgiwxhTBTkF7CXvN" cross
</head>

<body>

    {% include 'includes/_navbar.html' %}

    <main role="main" class="container">
        <div class="container">
            <div class="starter-template">
                <br style="padding-top: 100px;">
                {% with messages = get_flashed_messages() %}
                {% if messages %}
                <ul class=flashes>
                    {% for message in messages %}
                    <li>{{ message }}</li>
                    {% endfor %}
                </ul>
                {% endif %}
                {% endwith %}
                {% block content %}
                {% endblock %}
            </div>
        </div>
    </main><!-- /.container -->
    <!-- Bootstrap  JavaScript
    ================================================== -->
    <!-- Placed at the end of the document so the pages load faster -->
    <script src="https://code.jquery.com/jquery-3.6.3.min.js" integrity="sha256-pvPw+upLPUjgMXY0G+8O0xUf+/Im1MZjXxxgOcBQBXU=" crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js" integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3" crossorigin="an
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.min.js" integrity="sha384-mQ93GR66B00ZXjt0YO5KlohRA5SY2XofN4zfuZxLkoj1gXtW8ANNCe9d5Y3eG5eD" crossorigin=

</body>
</html>
```

Εικόνα 102. home/layout.html

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="{{ app_data['description'] }}">
    <meta name="keywords" content="{{ app_data['keywords'] }}">
    <meta name="author" content="{{ app_data['author'] }}">
    <link rel="icon" href="../../../../favicon.ico">
    <meta http-equiv="refresh" content="30">
    <title>{{ app_data['html_title'] }}</title>

    <!-- Bootstrap  CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-GLhlTQ8iRABdZLl6O3oVMWSkt
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js" integrity="sha384-w76AqPfDkMBDXo30jS1Sgez6pr3x5MlQ1ZAG
</head>

  <body>

    {% include 'includes/_navbarTeacher.html' %}

  <main role="main" class="container">
    <div class="container">
        <div class="starter-template">
            {% if current_user.is_authenticated %}
            <!-- Show current users email -->
            <b>Current user</b>: {{current_user.email}}
            <!-- Current users roles -->
            | <b>Role</b>: {% for role in current_user.roles%}
            {{role.name}}
            {% endfor %}
            <br>
            <br>

            <!-- Show if user is not logged in -->
            {% else %}
            <a href="/register">Sign up</a> | <a href="/login">Sign in</a>
            {% endif %}
            <hr style="padding-top: 100px;">
            {% with messages = get_flashed_messages() %}
            {% if messages %}
            <ul class=flashes>
                {% for message in messages %}
                <li>{{ message }}</li>
                {% endfor %}
            </ul>
            {% endif %}
            {% endwith %}

            {% block content %}
            {% endblock %}
        </div>
    </div>

  </main><!-- /.container -->

  <!-- Bootstrap JavaScript -->

  <script src="https://code.jquery.com/jquery-3.6.3.min.js" integrity="sha256-pvPw+upLPUjgMXY0G+8O0xUf+/Im1MZjXxxgOcBQBXU=" crossorigin="anonymous"></scri
  <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js" integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFu
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.min.js" integrity="sha384-mQ93GR66B00ZXjt0YO5KlohRA5SY2XofN4zfuZxLkoj
</body>
</html>
```

Εικόνα 103. teacherhome/layout.html

7. ΕΙΚΟΝΙΚΗ ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ

7.1 Παρουσίαση εικονικής διδασκαλίας και ανάλυση αποτελεσμάτων πρώτης ακαδημαϊκής χρονιάς

Μελετήθηκε μια εικονική περίπτωση διδασκαλίας (case study) στην οποία υποθετικά χρησιμοποιήθηκε η εφαρμογή σε ένα μάθημα για δύο διαδοχικά ακαδημαϊκά έτη.

Την πρώτη χρονιά, που το λογισμικό χρησιμοποιήθηκε για πρώτη φορά από τους φοιτητές και τον διδάσκοντα, οι φοιτητές έκαναν εγγραφή στο μάθημα, παρακολούθησαν τις διαλέξεις, είχαν πρόσβαση σε εκπαιδευτικό υλικό, ανακοινώσεις, βαθμούς και υπέβαλαν λύσεις των εργασιών τους. Επιπλέον οι φοιτητές αξιολόγησαν το περιεχόμενο της διάλεξης, υπέβαλλαν απορίες και ερωτήσεις μέσα από το εργαλείο στο οποίο είχαν πρόσβαση μόνο κατά την διάρκεια της διάλεξης (Εικόνες 104 -107).

Τη δεύτερη χρονιά, ο καθηγητής αφού αξιολόγησε τις αδυναμίες των φοιτητών μέσω των αποριών και των σχολίων που κατέθεσαν, έκανε βελτιώσεις στο περιεχόμενο των διαλέξεων και στον τρόπο παράδοσης του μαθήματος, βασιζόμενος στις ερωτήσεις που υποβλήθηκαν από τους φοιτητές κατά την πρώτη καθώς και την δεύτερη χρονιά. Επιπρόσθετα μετά την παράδοση της κάθε διάλεξης έδωσε έμφαση στις ερωτήσεις που υποβλήθηκαν κατά την διάρκεια του μαθήματος ώστε να εντοπίσει τις δυσκολίες των φοιτητών και να

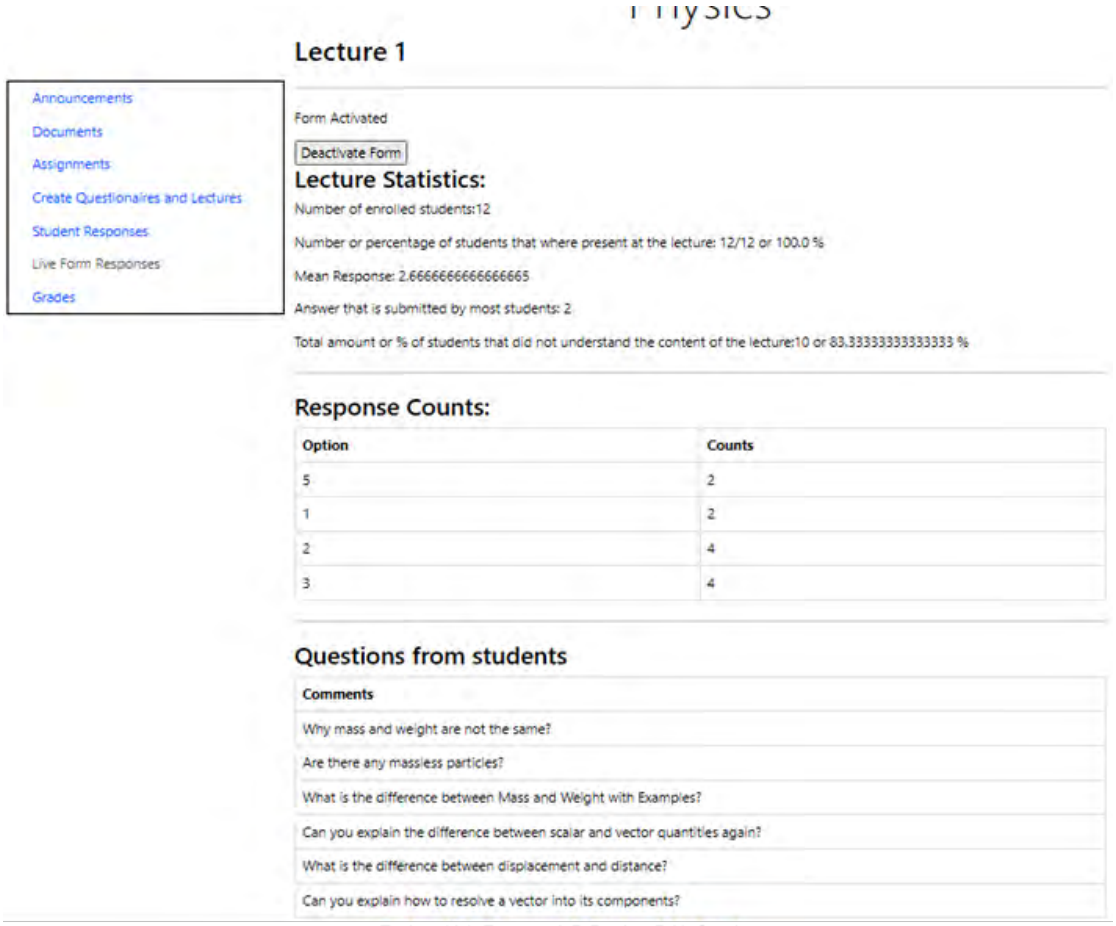
Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

προβεί σε ανάλογες βελτιώσεις, ανανεώνοντας το εκπαιδευτικό υλικό του συγκεκριμένου μαθήματος.

Η ανάλυση των αποτελεσμάτων της υποθετικής μελέτης περίπτωσης έδειξε ότι, την πρώτη χρονιά, υπήρχε μεγάλο ενδιαφέρον των φοιτητών για τις διαλέξεις (κυρίως μέχρι τις πρώτες 6), με το 100% των εγγεγραμμένων φοιτητών να χρησιμοποιούν ενεργά την φόρμα αλλά διατυπώνοντας πολλές απορίες και ερωτήσεις, με το ποσοστό των ατόμων που δεν κατανόησαν το περιεχόμενο του μαθήματος να κυμαίνεται σε σχετικά υψηλά επίπεδα, τουλάχιστον το 77% των φοιτητών και ο μέσος φοιτητής να μην κατανοεί το μάθημα (Εικόνες 107-109). Αυτό είχε ως αποτέλεσμα μετά το πέρας των 6 διαλέξεων η συμμετοχή των φοιτητών να μειώνεται, να αποθαρρυνθούν οι ίδιοι σχετικά με την ενασχόληση τους με το μάθημα να υπάρξουν και πολλές αποτυχίες στην τελική εξέταση του εξαμήνου (Εικόνα 107).

Οι τελικές βελτιώσεις περιλάμβαναν την ανανέωση των διαφανειών παρουσίασης, και την ενσωμάτωση επιπλέον παραδειγμάτων, ασκήσεων, εκπαιδευτικών βίντεο, ηλεκτρονικών βιβλίων που ζητήθηκαν από τους φοιτητές, προκειμένου οι διαλέξεις να είναι πιο εύληπτες και κατανοητές από τους φοιτητές.

Μερικά στιγμιότυπα αποτελεσμάτων της πρώτης χρονιάς:



Εικόνα 104. Στατιστικά δεδομένα διάλεξης 1

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Physics

Lecture 2

- Announcements
- Documents
- Assignments
- Create Questionaires and Lectures
- Student Responses
- Live Form Responses
- Grades

Form Activated

Deactivate Form

Lecture Statistics:

Number of enrolled students:12

Number or percentage of students that where present at the lecture: 12/12 or 100.0 %

Mean Response: 2.0

Answer that is submitted by most students: 1

Total amount or % of students that did not understand the content of the lecture:10 or 83.33333333333333 %

Response Counts:

Option	Counts
4	1
5	1
2	5
1	5

Questions from students

Comments
what is μ in the friction formula f = μ N?
Which type of surface produces more friction?
Can you explain how to resolve a vector into its components?
Could you explain the solution in exersice 2.1?
What is the difference between static friction and kinetic friction?
What does the coefficient of friction represent, and how is it determined?

Εικόνα 105. Στατιστικά δεδομένα διάλεξης 2.

Lecture 3

- Announcements
- Documents
- Assignments
- Create Questionaires and Lectures
- Student Responses
- Live Form Responses
- Grades

Form Activated

Deactivate Form

Lecture Statistics:

Number of enrolled students:12

Number or percentage of students that where present at the lecture: 12/12 or 100.0 %

Mean Response: 2.3333333333333335

Answer that is submitted by most students: 1

Total amount or % of students that did not understand the content of the lecture:10 or 83.33333333333333 %

Response Counts:

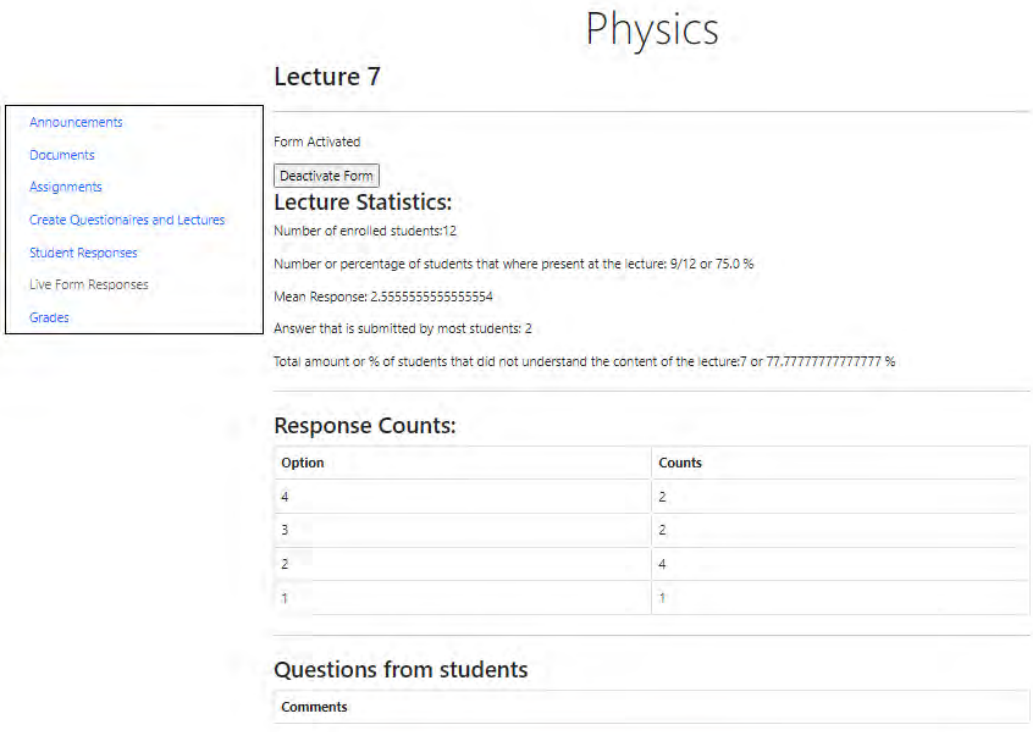
Option	Counts
4	2
3	4
1	4
2	2

Questions from students

Comments
How is mass related to inertia?
What are some real-world examples of inertia?
How does inertia affect the motion of objects in different scenarios, like in space vs. on Earth?
What is rotational inertia and how does it differ from linear inertia?
What forces are required to overcome an object's inertia?
Can you explain slide 20 again ?
Why do objects with greater mass have more inertia?
How is mass related to inertia?

Εικόνα 106. Στατιστικά δεδομένα διάλεξης 3.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43



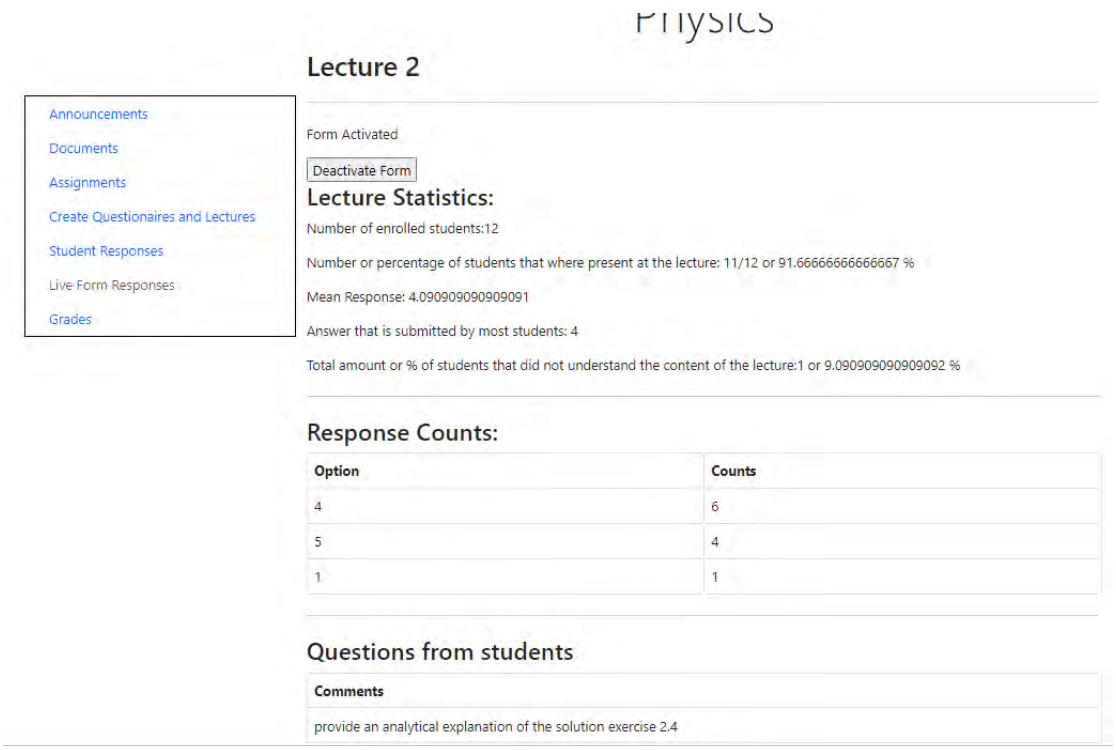
Εικόνα 107. Στατιστικά δεδομένα διάλεξης 7.

7.2 Αποτελέσματα εικονικής διδασκαλίας την δεύτερη ακαδημαϊκή χρονιά

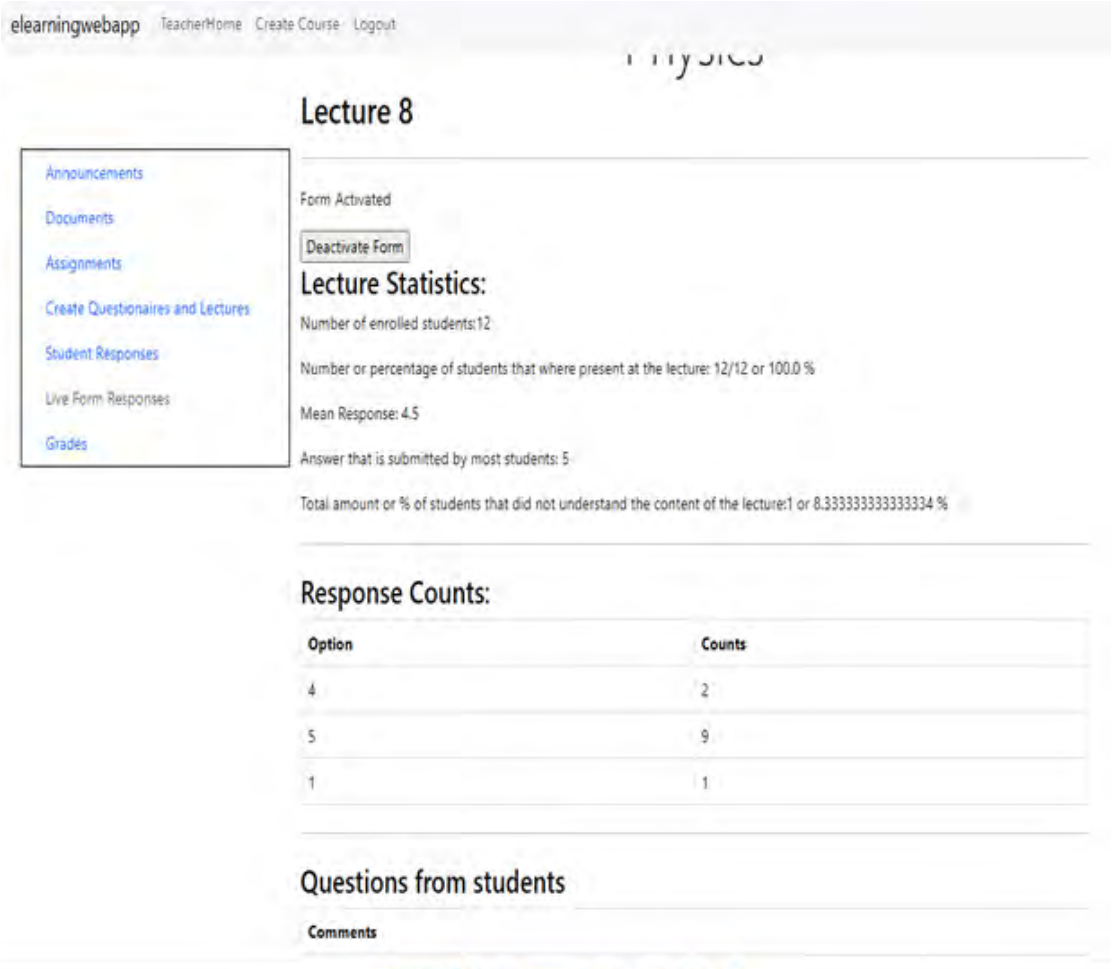
Την δεύτερη χρονιά, αφού ο διδάσκοντας έχει καταλήξει σε καλύτερα συμπεράσματα σχετικά με τις δυσκολίες και την πρόοδο των φοιτητών και έχει βελτιώσει το περιεχόμενο του υλικού των διαλέξεων, έχει παρατηρηθεί αύξηση και διατήρηση του ενδιαφέροντος των φοιτητών με τουλάχιστον το 90% των εγγεγραμμένων φοιτητών να χρησιμοποιούν το εργαλείο σε όλες τις διαλέξεις, ρωτώντας λιγότερες απορίες και ερωτήσεις, με το ποσοστό των ατόμων που δεν κατανόησαν το περιεχόμενο του μαθήματος ή της κάθε διάλεξης να μειώνεται γύρω στο 8-9% και ο μέσος φοιτητής να αξιολογεί τη διάλεξη ως αρκετά κατανοητή (Εικόνες 108-109).

Μερικά στιγμιότυπα αποτελεσμάτων της δεύτερης χρονιάς:

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43



Εικόνα 108. Στατιστικά δεδομένα διάλεξης 2.



Εικόνα 109. Στατιστικά της Διάλεξης 8

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

7.3 Ανάλυση αποτελεσμάτων μελέτης περίπτωσης

Συμπερασματικά, με βάση τα αποτελέσματα που εξάχθηκαν από την εικονική μελέτη περίπτωσης, μπορεί κάποιος να καταλήξει ότι η εφαρμογή είναι εξαιρετικά χρήσιμη τόσο για τους διδασκόμενους όσο και στους εκπαιδευτικούς. Με την προσθήκη λειτουργιών, όπως η δυνατότητα ανώνυμης υποβολής ερωτήσεων και σχολίων σε πραγματικό χρόνο, η εφαρμογή βοηθά τόσο τους καθηγητές όσο τους φοιτητές να ξεπερνούν τα όσα εμπόδια και ενδοιασμούς που προκύπτουν κατά την διδασκαλία όπως φαίνεται στην παρακάτω ανάλυση των αποτελεσμάτων.

Κατά τη διάρκεια του πρώτου ακαδημαϊκού έτους:

- Η συμμετοχή των φοιτητών μειώνονταν με την πάροδο των διαλέξεων.
- Αυξανόταν ο αριθμός των φοιτητών που δυσκολευόταν να κατανοήσουν το μάθημα.
- Ο αριθμός των ερωτήσεων-σχολίων τους ήταν σχετικά υψηλός.
- Το ποσοστό αποτυχίας στις εξετάσεις του μαθήματος ήταν μεγάλο.

Κατά τη διάρκεια του δεύτερου ακαδημαϊκού έτους:

- Η συμμετοχή των φοιτητών ήταν αυξημένη καθ’ όλη την διάρκεια των διαλέξεων, συγκριτικά με την προηγούμενη χρονιά.
- Μειώθηκε ο αριθμός των φοιτητών που δυσκολευόταν να κατανοήσουν το μάθημα.
- Ο αριθμός των ερωτήσεων-σχολίων τους ήταν μειωμένος.
- Το ποσοστό αποτυχίας στις εξετάσεις του μαθήματος ήταν πολύ μικρότερο.
- Τα παραπάνω ήταν ακόλουθα των βελτιώσεων του εκπαιδευτικού περιεχομένου από τον διδάσκοντα αφού είχε αναλύσει-εκτιμήσει τα στατιστικά δεδομένα της προηγούμενης χρονιάς.

8. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΙΘΑΝΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΕΙΣ

8.1 Συμπεράσματα

Συνεπώς, με βάση και τα δεδομένα που συγκεντρώθηκαν από την μελέτη περίπτωσης, μπορεί κάποιος να καταλήξει ότι η συγκεκριμένη εφαρμογή παρέχει σημαντικά οφέλη τόσο στους φοιτητές όσο και στους εκπαιδευτικούς. Με την εισαγωγή καινοτόμων λειτουργιών, όπως η δυνατότητα ανώνυμης υποβολής ερωτήσεων και σχολίων κατά τη διάρκεια της διάλεξης, η εφαρμογή βοηθά τόσο τους διδάσκοντες όσο τους διδασκόμενους να ξεπερνούν τα όσα εμπόδια και ενδοιασμούς που προκύπτουν κατά την εκπαιδευτική διαδικασία.
Ειδικότερα, βοηθά τους διδάσκοντες να αντιλαμβάνονται σε πραγματικό χρόνο καλύτερα τις αδυναμίες των σπουδαστών και να προχωρήσουν σε περεταίρω διευκρινίσεις αν αυτό είναι εφικτό.
Σε δεύτερο χρόνο, αναλύοντας τα στατιστικά δεδομένα των φοιτητών έχουν την δυνατότητα να προσαρμόζουν το εκπαιδευτικό περιεχόμενο και την μέθοδο διδασκαλίας βελτιώνοντας τα αναλόγως.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Αυτό αναμένεται να βελτιώσει σε μεγάλο βαθμό την ποιότητα της διδασκαλίας, να προσελκύσει τους φοιτητές καθώς και να διευκολύνει την μάθηση προσφέροντας ένα ευέλικτο, λειτουργικό και φιλικό προς το χρήστη περιβάλλον τόσο για τους φοιτητές όσο και για τους διδάσκοντες. Συμπερασματικά, θα μπορούσαμε να πούμε ότι η εφαρμογή εκπλήρωσε τον σκοπό για τον οποίο αναπτύχθηκε.

8.2 Πιθανές βελτιώσεις ή επεκτάσεις

Στην συγκεκριμένη εφαρμογή, όπως και κάθε άλλη επιδέχεται βελτιώσεις και επεκτάσεις. Κάποιες από αυτές είναι:

1. Προσθήκη thread μηνυμάτων. Πιο επεξηγηματικά θα μπορούσε να δοθεί η δυνατότητα στον φοιτητή ή στον καθηγητή να «ανοίξει» μια συζήτηση (thread παρόμοιο με το StackOverflow που είναι από τα πιο γνωστά , ορατό σε όλους τους φοιτητές που έχουν εγγραφεί στο μάθημα) με συγκεκριμένο θέμα που αφορά το μάθημα, στην οποία θα μπορούν να συμμετέχουν όσοι φοιτητές ενδιαφέρονται στέλνοντας μηνύματα, ρωτώντας απορίες ή λύνοντας τες. Παρόλο που θα μπορούν να έχουν πρόσβαση όλοι οι χρήστες, θα μπορούσε να δοθεί δυνατότητα στον φοιτητή να διατηρήσει την ανωνυμία εφόσον το επιθυμεί.
2. Προσθήκη ειδοποιήσεων. Θα μπορούσε να προστεθεί στην εν λόγω εφαρμογή η λειτουργία της εμφάνισης των ειδοποιήσεων. Κάθε φορά που γίνεται κάποια νέα δραστηριότητα σχετικά με το μάθημα(π.χ νέα ανακοίνωση, ανέβασμα εργασιών, εγγράφων, ενεργοποίηση ή απενεργοποίηση φόρμας διαλέξεων κλπ) θα εμφανίζεται σχετική νέα ειδοποίηση στην συσκευή ή μόλις γίνει είσοδος στον λογαριασμό του από τον φοιτητή ή τον καθηγητή.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

Αναφορές

[1] DunbarEducation, «4 Must-Have Apps for Teachers,» [Ηλεκτρονικό]. Available: https://www.dunbareducation.com/blog/2024/01/4-must-have-apps-for-teachers?source=google.com.

[2] EducationCorner, «35 of the BEST Educational Apps for Teachers (Updated 2024),» 24 April 2024. [Ηλεκτρονικό]. Available: https://www.educationcorner.com/35-incredible-classroom-apps/.

[3] Wikipedia, «MVC,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller.

[4] ip.gr, «Τι είναι η HTML?,» [Ηλεκτρονικό]. Available: https://www.ip.gr/Web_Development/%CF%84%CE%B9-%CE%B5%CE%AF%CE%BD%CE%B1%CE%B9-%CE%B7-html-61.html.

[5] wlearn.gr, «Η γλώσσα HTML,» [Ηλεκτρονικό]. Available: https://www.wlearn.gr/index.php/html-html-32.

[6] wasinghton.edu, «A Brief History of HTML,» [Ηλεκτρονικό]. Available: https://www.washington.edu/accesscomputing/webd2/student/unit1/module3/html_history.html#:~:text=The%20first%20version%20of%20HTML,HTML%20as%20an%20XML%20language.

[7] Wikipedia, «Bootstrap,» [Ηλεκτρονικό]. Available: https://el.wikipedia.org/wiki/Bootstrap#:~:text=%CE%A4%CE%BF%20Bootstrap%20%CE%B5%CE%AF%CE%BD%CE%B1%CE%B9%20%CE%BC%CE%B9%CE%B1%20%CF%83%CF%85%CE%BB%CE%BB%CE%BF%CE%B3%CE%AE,%CE%BA%CE%B1%CE%B8%CF%8E%CF%82%20%CE%BA%CE%B1%CE%B9%20%CF%80%CF%81%CE%BF%CE%B1%CE%B.

[8] «Clarity,» [Ηλεκτρονικό]. Available: https://www.clarity-ventures.com/blog/benefits-of-using-bootstrap-for-web-design.

[9] Wikipedia, «Python (programming language),» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Python_(programming_language).

[10] Wikipedia, «Flask (web framework),» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Flask_(web_framework).

[11] Wikipedia, «Jinja (template engine),» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Jinja_(template_engine).

[12] Wikipedia, «SQLite,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/SQLite.

[13] Smartpaces, «Benefits of Role Based Access Control,» [Ηλεκτρονικό]. Available: https://www.smartspaces.app/blog/role-based-access-control/.

[14] Okta, «What Is Role-Based Access Control (RBAC)?,» 2024. [Ηλεκτρονικό]. Available: https://www.okta.com/identity-101/what-is-role-based-access-control-rbac/.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43

[15] B. Herzberg, «Dataversity,» 16 October 2023. [Ηλεκτρονικό]. Available: https://www.dataversity.net/4-benefits-of-role-based-access-control-rbac-and-how-to-implement-it/.

[16] Ben-Herzberg Dataversity, «4 Benefits of Role-Based Access Control (RBAC) and How to Implement It,» 2023.

[17] Clarity, «Clarity,» [Ηλεκτρονικό]. Available: https://www.clarity-ventures.com/blog/benefits-of-using-bootstrap-for-web-design.

[18] T. H. Education. [Ηλεκτρονικό]. Available: https://www.timeshighereducation.com/campus/anonymous-polling-platforms-boost-student-confidence-engagement-and-inclusivity.

Institutional Repository - Library & Information Centre - University of Thessaly
09/08/2026 21:33:01 EEST - 216.73.216.43