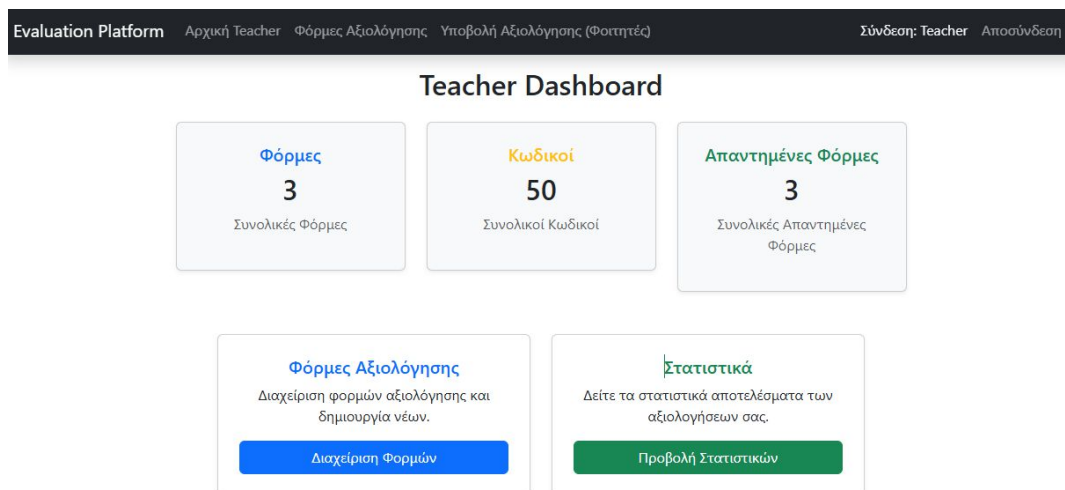


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
«Πλατφόρμα συλλογής και ανάλυσης αξιολογήσεων
διδασκόντων από φοιτητές»



Φοιτητής

ΔΗΜΗΤΡΙΟΣ ΖΑΦΕΙΡΙΟΥ

514037

Επιβλέπων

Δρ. Κυριάκος Τσακμάκης

Ιούνιος 2025

Πλατφόρμα συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές

Κωδικός: 25152

Φοιτητής: ΖΑΦΕΙΡΙΟΥ ΔΗΜΗΤΡΙΟΣ

Εισηγητής: Δρ Κυριάκος Τσιακμάκης

Ημερομηνία ανάληψης Π.Ε. 08-03-2025

Ημερομηνία περάτωσης Π.Ε. 16-05-2025

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Ζαφειρίου Δημήτριου που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Περίληψη

Η πλατφόρμα αυτή σχεδιάστηκε για να διευκολύνει τη διαδικασία συλλογής και ανάλυσης ανώνυμων αξιολογήσεων διδασκόντων από φοιτητές, με στόχο τη βελτίωση της ποιότητας διδασκαλίας στα ακαδημαϊκά ιδρύματα. Μέσω φιλικού περιβάλλοντος οι φοιτητές μπορούν να παρέχουν αξιολογήσεις χρησιμοποιώντας μοναδικούς κωδικούς ώστε να εξασφαλίσουν την εμπιστευτικότητα και την εγκυρότητα των δεδομένων. Οι διδάσκοντες έχουν τη δυνατότητα να δημιουργούν φόρμες αξιολόγησης, να επιλέγουν ερωτήσεις από προκαθορισμένες δεξαμενές και να παρακολουθούν στατιστικά αποτελέσματα μέσω διαδραστικών γραφημάτων. Το σύστημα υποστηρίζει την ανάλυση μέσω όρων ανά ερώτηση και κατηγορία και παράλληλα καταγράφει και τα σχόλια των φοιτητών. Η πλατφόρμα αναπτύχθηκε με χρήση Python (Flask), MySQL και Bootstrap.

« Platform for collecting and analyzing teacher evaluations from students»

Abstract

This platform was designed to facilitate the process of collecting and analyzing anonymous teacher evaluations from students, with the aim of improving the quality of teaching in academic institutions. Through a friendly environment, students can provide evaluations using unique codes, ensuring the confidentiality and validity of the data. Teachers have the ability to create evaluation forms, select questions from predefined pools and monitor statistical results through interactive graphs. The system supports the analysis of averages per question and category, while also recording student comments. The platform was developed using Python (Flask), MySQL and Bootstrap.

Ευχαριστίες

Θέλω να ευχαριστήσω τους φίλους μου και την οικογένεια μου για την συμπαράσταση και κατανόησή τους και τον επιβλέποντα μου για τη βοήθεια του σε όλη τη διαδικασία του έργου.

Περιεχόμενα

Περίληψη	iv
Abstract	v
Ευχαριστίες	vi
Περιεχόμενα	vii
Κατάλογος Σχημάτων	viii
Κεφάλαιο 1ο: Εισαγωγή	9
1.1 Εισαγωγή	9
1.2 Η δομή της εργασίας.....	11
Κεφάλαιο 2ο: Ορισμένες πλατφόρμες συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές 12	
2.1 Blue by Explorance [1]	12
2.2 Watermark Course Evaluations & Surveys [2].....	14
2.3 SmartEvals [3]	16
Κεφάλαιο 3ο: Γλώσσες προγραμματισμού και χρήσεις	18
3.1 Γλώσσα Προγραμματισμού Backend: Python.....	18
3.2 Framework Backend: Flask	19
3.3 Βάση Δεδομένων: MySQL	22
3.4 Frontend Βιβλιοθήκη: Bootstrap	22
3.5 Βιβλιοθήκη Γραφημάτων: Chart.js.....	23
Κεφάλαιο 4ο: Η πλατφόρμα συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές 25	
4.1 Οι διαδικασίες.....	25
4.2 Η διαδικτυακή πλατφόρμα	28
4.2.1 Ρόλος: Teacher – Διδάσκοντας	29
4.2.2 Ρόλος: Student – Φοιτητής.....	41
4.2.3 Ρόλος: Admin – Διαχειριστής	45
4.3 Περιγραφή συστήματος μέσω κώδικα.....	47
4.4 Η βάση του συστήματος	53
4.4.1 Περιγραφή των πινάκων και των πεδίων της βάσης δεδομένων.....	54
Κεφάλαιο 5ο: Αποτίμηση Αποτελεσμάτων και Μελλοντικές Κατευθύνσεις.....	57

ΒΙΒΛΙΟΓΡΑΦΙΑ.....	58
ΠΑΡΑΡΤΗΜΑ - Κώδικες.....	59

Κατάλογος Σχημάτων

Εικόνα 2.1: Blue: The Feedback Analytics Platform	13
Εικόνα 2.2 Watermark Course Evaluations & Surveys	15
Εικόνα 2.3 SmartEvals Evaluations.....	16
Εικόνα 3.1: Πλεονεκτήματα flask python.....	20
Εικόνα 4.1: Διάγραμμα Ροής Υποβολής Αξιολόγησης (Student Flow)	25
Εικόνα 4.2: Διάγραμμα Ροής Διαχείρισης Αξιολογήσεων (Teacher Flow)	26
Εικόνα 4.3: Διαδικασία Χρηστών Admin.....	27
Εικόνα 4.4: Σύνδεση	28
Εικόνα 4.5: Εγγραφή	29
Εικόνα 4.6: Αρχική Σελίδα το Διδάσκοντα - Teacher Dashboard.....	30
Εικόνα 4.7: Σελίδα Λίστα Φορμών Αξιολόγησης	32
Εικόνα 4.8: Δημιουργία Νέας Φόρμας 1 - Teacher.....	32
Εικόνα 4.9: Δημιουργία Νέας Φόρμας 2 - Teacher.....	33
Εικόνα 4.10: Επεξεργασία Φόρμας - Οριστικοποιημένη - Teacher	34
Εικόνα 4.11: Επεξεργασία Φόρμας – Μη Οριστικοποιημένη - Teacher.....	36
Εικόνα 4.12: Σελίδα: Κωδικοί Αξιολόγησης - Teacher	37
Εικόνα 4.13: Σελίδα: Στατιστικά Αξιολόγησης - Teacher	38
Εικόνα 4.14: Οπτικοποίηση του Μέσου Όρου ανά Κατηγορία για τις αξιολογήσεις ενός μαθήματος - Teacher.....	39
Εικόνα 4.15: Αναλυτικά Στατιστικά Αποτελέσματα Αξιολόγησης	40
Εικόνα 4.16: Υποβολή Αξιολόγησης.....	41
Εικόνα 4.17: Στοιχεία Φόρμας Αξιολόγησης και κουμπί για Συμπλήρωση Αξιολόγησης	42
Εικόνα 4.18: Φόρμα Αξιολόγησης.....	43
Εικόνα 4.19: Φόρμα Αξιολόγησης – συνέχεια	44
Εικόνα 4.20: Φόρμα Αξιολόγησης – Ειδοποίηση σε περίπτωση μη συμπλήρωσης-απάντησης ερώτησης.....	44
Εικόνα 4.21: Ειδοποίηση σε περίπτωση εισαγωγής χρησιμοποιούμενου κωδικού κατά την υποβολή αξιολόγησης.....	45
Εικόνα 4.22: Admin Dashboard.....	45
Εικόνα 4.23: Λίστα Ερωτήσεων.....	46
Εικόνα 4.24: Νέα Ερώτηση- Επεξεργασία Ερώτησης	46

Κεφάλαιο 1ο: Εισαγωγή

1.1 Εισαγωγή

Η αξιολόγηση των διδασκόντων από τους φοιτητές είναι μια σημαντική διαδικασία που συμβάλλει στη βελτίωση της ποιότητας της εκπαίδευσης. Μέσα από αυτή τη διαδικασία, οι φοιτητές έχουν την ευκαιρία να εκφράσουν τη γνώμη τους για το πώς διδάσκεται ένα μάθημα, πόσο κατανοητός και αποτελεσματικός είναι ο διδάσκοντας και πώς θα μπορούσε να βελτιωθεί το όλο. Οι διδάσκοντες μπορούν να εντοπίσουν δυνατά και αδύναμα σημεία στη διδασκαλία τους κάτι που τους βοηθά να προσαρμόσουν τις μεθόδους και τις πρακτικές τους.

Τέτοιου είδους αξιολογήσεις πραγματοποιούνται σε πολλά πανεπιστήμια και εκπαιδευτικούς οργανισμούς σε όλο τον κόσμο. Είναι ένας τρόπος να λαμβάνονται αποφάσεις βασισμένες σε πραγματικές ανάγκες και εμπειρίες των φοιτητών. Όταν χρησιμοποιούνται σωστά όλα αυτά τότε τα αποτελέσματα των αξιολογήσεων οδηγούν σε βελτίωση των μαθημάτων και καλύτερη επικοινωνία ανάμεσα σε φοιτητές και διδάσκοντες και γενικά σε μια πιο ποιοτική εκπαίδευση.

Τα τελευταία χρόνια, η τεχνολογία έχει βοηθήσει σημαντικά στην υλοποίηση τέτοιων συστημάτων αξιολόγησης. Παλαιότερα οι αξιολογήσεις γίνονταν με χαρτιά και έντυπα ερωτηματολόγια κάτι που απαιτούσε πολύ χρόνο και κόπο για να συγκεντρωθούν και να αναλυθούν τα δεδομένα. Σήμερα οι αξιολογήσεις γίνονται ψηφιακά μέσω διαδικτυακών πλατφορμών γρήγορα με ακρίβεια και ευκολία στη διαχείριση των αποτελεσμάτων.

Οι σύγχρονες πλατφόρμες αξιολόγησης χρησιμοποιούν τεχνολογίες όπως οι γλώσσες προγραμματισμού Python, JavaScript και PHP για την ανάπτυξη των εφαρμογών. Για την αποθήκευση των δεδομένων χρησιμοποιούνται βάσεις δεδομένων όπως MySQL και PostgreSQL και τα αποτελέσματα παρουσιάζονται με τη χρήση εργαλείων γραφημάτων και στατιστικής ανάλυσης όπως το Chart.js. Αυτά τα εργαλεία επιτρέπουν τη δημιουργία εύχρηστων και ευανάγνωστων αναφορών που βοηθούν τόσο τους διδάσκοντες όσο και τη διοίκηση των ιδρυμάτων.

Η αξιολόγηση μέσω ψηφιακών πλατφορμών έχει το πλεονέκτημα της ανωνυμίας δίνοντας τη δυνατότητα στους φοιτητές να εκφράζουν ελεύθερα τη γνώμη τους χωρίς φόβο ή δισταγμό. Οι διδάσκοντες έχουν άμεση πρόσβαση στα στατιστικά αποτελέσματα βλέποντας συγκεντρωτικά τις αξιολογήσεις τους αλλά και τα συγκεκριμένα σχόλια που μπορεί να τους βοηθήσουν να κατανοήσουν τις ανάγκες των φοιτητών καλύτερα.

Τέτοιες πλατφόρμες βρίσκουν εφαρμογή όχι μόνο στα πανεπιστήμια αλλά και σε άλλους οργανισμούς, όπως κέντρα επαγγελματικής κατάρτισης, ιδιωτικές σχολές και εταιρείες που οργανώνουν σεμινάρια και εκπαιδευτικά προγράμματα. Οπουδήποτε υπάρχει ανάγκη για βελτίωση της εκπαίδευσης και

ανατροφοδότηση από τους εκπαιδευόμενους, τέτοια συστήματα μπορούν να προσφέρουν ουσιαστικές λύσεις.

Στη συνέχεια θα παρουσιάσουμε την πλατφόρμα που σχεδιάσαμε και αναπτύξαμε η οποία έχει ως στόχο να καλύψει αυτές τις ανάγκες. Θα δούμε αναλυτικά τα χαρακτηριστικά της, τις τεχνολογίες που χρησιμοποιήθηκαν και τα οφέλη που προσφέρει τόσο στους φοιτητές όσο και στους διδάσκοντες.

Η πλατφόρμα που αναπτύξαμε έχει σχεδιαστεί για να καλύψει πλήρως τις ανάγκες συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές με στόχο την απλότητα, την ασφάλεια και την αξιοπιστία. Το σύστημα επιτρέπει την καταγραφή ανώνυμων αξιολογήσεων και διασφαλίζει την ειλικρίνεια των απαντήσεων και την προστασία των προσωπικών δεδομένων των φοιτητών.

Μέσω της πλατφόρμας οι διδάσκοντες μπορούν να δημιουργούν δυναμικές φόρμες αξιολόγησης, επιλέγοντας ερωτήσεις από μια οργανωμένη δεξαμενή 40 ερωτήσεων. Οι ερωτήσεις καλύπτουν τέσσερις βασικούς τομείς: το περιεχόμενο του μαθήματος, τον διδάσκοντα, τις υποδομές/εγκαταστάσεις και τον ίδιο τον φοιτητή. Η κάθε φόρμα μπορεί να διαμορφωθεί ανάλογα με τις ανάγκες του μαθήματος και να συνοδεύεται από τίτλο, περιγραφή και χρονικό διάστημα υποβολής.

Η πλατφόρμα υποστηρίζει τη δημιουργία μοναδικών κωδικών συμμετοχής για τους φοιτητές, οι οποίοι χρησιμοποιούνται μία μόνο φορά για την υποβολή των αξιολογήσεων. Αυτό διασφαλίζει ότι κάθε φοιτητής συμμετέχει μόνο μία φορά σε κάθε αξιολόγηση, ενώ παράλληλα παραμένει ανώνυμος. Μετά την υποβολή, ο κωδικός μαρκάρεται αυτόματα ως χρησιμοποιημένος και δεν μπορεί να επαναχρησιμοποιηθεί.

Η ανάλυση των αποτελεσμάτων γίνεται μέσω διαδραστικών γραφημάτων και στατιστικών αναφορών. Οι διδάσκοντες μπορούν να δουν άμεσα τα αποτελέσματα ανά ερώτηση και κατηγορία και τα σχόλια που έχουν υποβάλει οι φοιτητές. Το σύστημα παρέχει τη δυνατότητα υπολογισμού μέσου όρου για κάθε ερώτηση και κατηγορία, δίνοντας μια σαφή εικόνα για τα δυνατά και αδύναμα σημεία της διδασκαλίας.

Η ανάπτυξη της πλατφόρμας πραγματοποιήθηκε με χρήση της γλώσσας Python και του framework Flask για το backend, εξασφαλίζοντας σταθερότητα και ευκολία επεκτασιμότητας. Για την αποθήκευση των δεδομένων χρησιμοποιήθηκε η βάση MySQL, ενώ η διεπαφή χρήστη σχεδιάστηκε με το Bootstrap για ένα μοντέρνο και φιλικό περιβάλλον. Τα γραφήματα υλοποιήθηκαν με το Chart.js, επιτρέποντας τη δυναμική απεικόνιση των αποτελεσμάτων με απλό και κατανοητό τρόπο.

Ιδιαίτερη έμφαση δόθηκε στην ασφάλεια της πλατφόρμας. Η χρήση μοναδικών κωδικών αποτρέπει την κατάχρηση του συστήματος και η απουσία προσωπικών δεδομένων στις αξιολογήσεις ενισχύει την ανωνυμία των φοιτητών. Το σύστημα ελέγχει την εγκυρότητα κάθε κωδικού πριν από την υποβολή της αξιολόγησης, εξασφαλίζοντας ότι δεν υπάρχουν διπλοεγγραφές.

Η πλατφόρμα που υλοποιήσαμε αποτελεί ένα ισχυρό εργαλείο για τη συστηματική αξιολόγηση του εκπαιδευτικού έργου. Παρέχει τόσο στους διδάσκοντες όσο και στη διοίκηση του ιδρύματος πολύτιμες πληροφορίες που μπορούν να αξιοποιηθούν για τη συνεχή βελτίωση της διδασκαλίας και την ενίσχυση της φοιτητικής εμπειρίας.

1.2 Η δομή της εργασίας

Η εργασία οργανώνεται σε έξι κύρια κεφάλαια, ξεκινώντας με την εισαγωγή, όπου παρουσιάζεται το αντικείμενο μελέτης και η αναγκαιότητα υλοποίησης ενός τέτοιου συστήματος. Στην εισαγωγή αναλύεται επίσης η δομή της εργασίας.

Στο δεύτερο κεφάλαιο γίνεται ανασκόπηση υπάρχουσών πλατφορμών συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές. Παρουσιάζονται παραδείγματα γνωστών συστημάτων, αναλύονται τα χαρακτηριστικά τους, οι χρήσεις τους, καθώς και τα πλεονεκτήματα και μειονεκτήματά τους.

Το τρίτο κεφάλαιο επικεντρώνεται στις τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της πλατφόρμας. Αναλύεται η γλώσσα προγραμματισμού Python, το framework Flask για την ανάπτυξη του backend, το σύστημα διαχείρισης βάσης δεδομένων MySQL, η βιβλιοθήκη Bootstrap για τον σχεδιασμό του frontend και η Chart.js για την παρουσίαση των στατιστικών αποτελεσμάτων με γραφήματα.

Στο τέταρτο κεφάλαιο παρουσιάζεται αναλυτικά η πλατφόρμα που αναπτύχθηκε. Αναλύονται οι διαδικασίες που υποστηρίζει το σύστημα και η δομή του με βάση τους τρεις βασικούς ρόλους χρηστών: διδάσκοντα, φοιτητής και διαχειριστής. Παρουσιάζονται επίσης σημαντικά αποσπάσματα κώδικα και η αρχιτεκτονική της βάσης δεδομένων, συμπεριλαμβανομένης της περιγραφής των πινάκων και των πεδίων που χρησιμοποιούνται.

Το πέμπτο κεφάλαιο αναφέρεται σε ειδικά θέματα που σχετίζονται με την πλατφόρμα ή με ζητήματα που προέκυψαν κατά την ανάπτυξη και υλοποίηση της εφαρμογής.

Τέλος, στο έκτο κεφάλαιο παρουσιάζονται τα συμπεράσματα που προέκυψαν από την υλοποίηση της πλατφόρμας και προτείνονται μελλοντικές κατευθύνσεις και βελτιώσεις για την περαιτέρω ανάπτυξη του συστήματος.

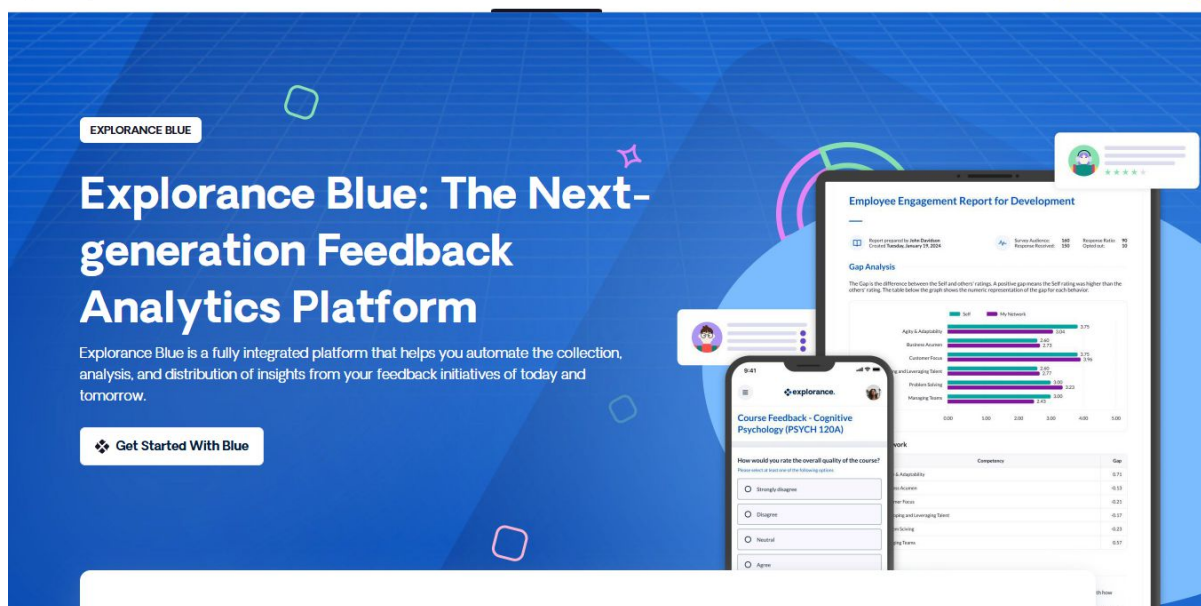
Η εργασία ολοκληρώνεται με τη βιβλιογραφία και το παράρτημα, όπου παρατίθενται οι πηγές που χρησιμοποιήθηκαν και το βασικό τμήμα του κώδικα της εφαρμογής.

Κεφάλαιο 2ο: Ορισμένες πλατφόρμες συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές

Τα συστήματα συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές αποτελούν ένα από τα σημαντικότερα εργαλεία διασφάλισης και βελτίωσης της ποιότητας της εκπαίδευσης. Μέσα από αυτά τα συστήματα, οι φοιτητές έχουν την ευκαιρία να πούνε την άποψή τους για την ποιότητα της διδασκαλίας, την αποτελεσματικότητα των διδασκόντων, το εκπαιδευτικό υλικό και τις υποδομές, συμβάλλοντας ουσιαστικά στη βελτίωση του ακαδημαϊκού περιβάλλοντος. Παράλληλα, οι διδάσκοντες και οι διοικήσεις των ιδρυμάτων μπορούν να εντοπίσουν αδυναμίες, να ενισχύσουν τα θετικά σημεία και να σχεδιάσουν δράσεις βελτίωσης με βάση πραγματικά δεδομένα. Αυτά τα συστήματα βασίζονται σε σύγχρονες τεχνολογίες πληροφορικής και χρησιμοποιούν διαδικτυακές εφαρμογές, βάσεις δεδομένων και στατιστικά εργαλεία ανάλυσης. Προσφέρουν ευκολία στη διαχείριση μεγάλου όγκου δεδομένων και η χρήση διαδραστικών γραφημάτων και δυναμικών αναφορών διευκολύνει την κατανόηση των αποτελεσμάτων. Παρέχουν τη δυνατότητα στους φοιτητές να συμμετέχουν ανώνυμα και άμεσα μέσω ηλεκτρονικών συσκευών μειώνοντας έτσι τις καθυστερήσεις και τα λάθη που υπήρχαν με τα παραδοσιακά έντυπα ερωτηματολόγια. Η αξιοποίηση τέτοιων συστημάτων είναι πλέον αναγκαία σε κάθε σύγχρονο ακαδημαϊκό περιβάλλον, καθώς συμβάλλουν στη διαρκή αναβάθμιση της εκπαιδευτικής διαδικασίας και στην ενίσχυση της φοιτητικής εμπειρίας, ενισχύοντας παράλληλα τη διαφάνεια και την αξιοπιστία της διαδικασίας αξιολόγησης.

2.1 Blue by Explorance [1]

Το Blue by Explorance είναι ένα από τα πιο διαδεδομένα συστήματα συλλογής και ανάλυσης αξιολογήσεων στον χώρο της εκπαίδευσης. Πρόκειται για μια ολοκληρωμένη διαδικτυακή πλατφόρμα που έχει σχεδιαστεί ειδικά για τη διαχείριση αξιολογήσεων διδασκόντων και μαθημάτων σε πανεπιστήμια και εκπαιδευτικούς οργανισμούς. Το Blue προσφέρει εργαλεία που καλύπτουν όλη τη διαδικασία της αξιολόγησης, από τη δημιουργία ερωτηματολογίων μέχρι την ανάλυση των αποτελεσμάτων με τη χρήση προηγμένων στατιστικών εργαλείων και γραφημάτων. Η πλατφόρμα είναι πλήρως παραμετροποιήσιμη κάτι που την καθιστά κατάλληλη για διαφορετικούς τύπους οργανισμών και προγραμμάτων σπουδών.



Εικόνα 2.1: Blue: The Feedback Analytics Platform

[<https://www.explorance.com/products/blue>]

Το Blue χρησιμοποιείται κυρίως από πανεπιστήμια και κολέγια για τη συλλογή αξιολογήσεων από φοιτητές σχετικά με τα μαθήματα, τους διδάσκοντες και τις παρεχόμενες υπηρεσίες. Μέσα από αυτό το σύστημα, οι φοιτητές μπορούν να συμπληρώνουν ερωτηματολόγια ηλεκτρονικά, εύκολα και γρήγορα, χρησιμοποιώντας υπολογιστές, tablets ή κινητά τηλέφωνα. Τα αποτελέσματα συγκεντρώνονται αυτόματα και παρέχονται σε μορφή αναφορών προς τη διοίκηση και τους διδάσκοντες. Η πλατφόρμα χρησιμοποιείται επίσης για αξιολογήσεις προγραμμάτων σπουδών, εσωτερικές διαδικασίες βελτίωσης και ανατροφοδότηση σε εκπαιδευτικές υπηρεσίες.

Η χρησιμότητα του Blue είναι ιδιαίτερα σημαντική καθώς βοηθά στη συλλογή μεγάλου όγκου δεδομένων με τρόπο οργανωμένο και αποδοτικό. Τα αποτελέσματα των αξιολογήσεων μπορούν να αξιοποιηθούν για τη βελτίωση των μαθημάτων, την ενίσχυση των διδασκαλικών μεθόδων και την καλύτερη κατανόηση των αναγκών των φοιτητών. Διευκολύνει τη διαδικασία λήψης αποφάσεων σε διοικητικό επίπεδο και παρέχει αναλυτικές και στοχευμένες αναφορές που βασίζονται σε πραγματικά δεδομένα.

Τα πλεονεκτήματα της πλατφόρμας Blue περιλαμβάνουν τον υψηλό βαθμό παραμετροποίησης, την εύκολη ενσωμάτωση με άλλα συστήματα (όπως Learning Management Systems - LMS), την παραγωγή σύνθετων στατιστικών αναφορών και τη δυνατότητα δημιουργίας πολύπλοκων ερωτηματολογίων. Η πλατφόρμα παρέχει επίσης εργαλεία ανάλυσης δεδομένων με τη χρήση γραφημάτων και dashboard, προσφέροντας στους χρήστες μια ολοκληρωμένη εικόνα των αποτελεσμάτων.

Το σημαντικότερο μειονέκτημα του Blue είναι το υψηλό κόστος απόκτησης και συντήρησης, κάτι που το καθιστά απρόσιτο για μικρότερα εκπαιδευτικά ιδρύματα ή οργανισμούς με περιορισμένους οικονομικούς πόρους. Λόγω της πολυπλοκότητας του, απαιτείται εκπαίδευση των χρηστών για την πλήρη αξιοποίηση των δυνατοτήτων του και σε ορισμένες περιπτώσεις η διαδικασία ρύθμισης και παραμετροποίησης μπορεί να αποδειχθεί χρονοβόρα.

Σε σύγκριση με τη δική μας πλατφόρμα το Blue προσφέρει περισσότερες δυνατότητες παραμετροποίησης και υποστήριξη για μεγάλης κλίμακας εκπαιδευτικούς οργανισμούς. Η πλατφόρμα μας έχει το πλεονέκτημα της απλότητας, της φιλικότητας προς τον χρήστη και του χαμηλότερου κόστους ανάπτυξης και λειτουργίας. Το σύστημά μας είναι ιδανικό για ιδρύματα που επιθυμούν μια εύκολη και άμεση λύση για τη συλλογή αξιολογήσεων χωρίς να απαιτούνται πολύπλοκες ρυθμίσεις και πρόσθετα λειτουργικά κόστη. Επιπρόσθετα η δική μας πλατφόρμα προσφέρει όλα τα απαραίτητα εργαλεία για τη συλλογή, ανάλυση και παρουσίαση των αξιολογήσεων με διαδραστικά γραφήματα και αναφορές καλύπτοντας πλήρως τις ανάγκες μικρών και μεσαίων εκπαιδευτικών μονάδων.

2.2 Watermark Course Evaluations & Surveys [2]

Το Watermark Course Evaluations & Surveys είναι μια προηγμένη διαδικτυακή πλατφόρμα που ειδικεύεται στη συλλογή και ανάλυση αξιολογήσεων διδασκόντων και μαθημάτων. Χρησιμοποιείται ευρέως από πανεπιστήμια, κολέγια και άλλους εκπαιδευτικούς οργανισμούς που επιδιώκουν να βελτιώσουν την ποιότητα των εκπαιδευτικών τους υπηρεσιών. Η πλατφόρμα έχει σχεδιαστεί για να προσφέρει ένα ολοκληρωμένο σύστημα διαχείρισης ερωτηματολογίων, συλλογής δεδομένων και παραγωγής αναφορών, με έμφαση στην αυτοματοποίηση των διαδικασιών και την εύκολη πρόσβαση στα αποτελέσματα .

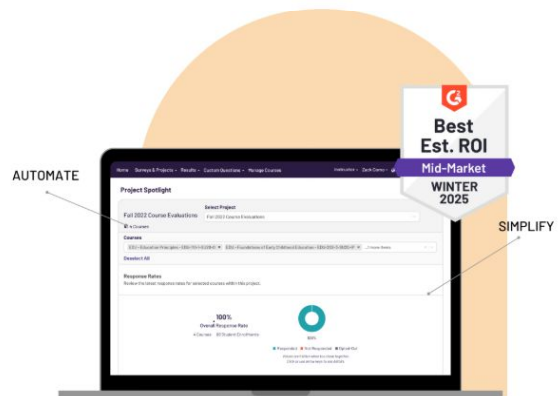
Η κύρια χρήση του Watermark αφορά τη συλλογή αξιολογήσεων από φοιτητές για μαθήματα, διδάσκοντες και γενικά την εκπαιδευτική εμπειρία. Χρησιμοποιείται για την αξιολόγηση προγραμμάτων σπουδών, την παρακολούθηση εσωτερικών διαδικασιών ποιότητας και την παροχή ανατροφοδότησης προς τις διοικήσεις και τους υπευθύνους προγραμμάτων. Χάρη στη δυνατότητα ενσωμάτωσης με συστήματα διαχείρισης μάθησης (LMS), όπως το Blackboard και το Canvas, οι φοιτητές μπορούν να συμπληρώνουν τις αξιολογήσεις τους απευθείας μέσω των εκπαιδευτικών πλατφορμών που ήδη χρησιμοποιούν.



Quality feedback, deeper insights, instant results

Make student feedback the focus of your campus picture. Simplify the course evaluation process and convert results into actionable next steps.

[Book a meeting →](#)



Εικόνα 2.2 Watermark Course Evaluations & Surveys

[<https://www.watermarkinsights.com/solutions/course-evaluation-institutional-surveys/>]

Η χρησιμότητα του Watermark είναι ιδιαίτερα σημαντική για οργανισμούς που επιδιώκουν τη διαρκή βελτίωση των υπηρεσιών τους μέσω της αξιολόγησης και της συστηματικής ανάλυσης των αποτελεσμάτων. Παρέχει εργαλεία για τη δημιουργία αναλυτικών αναφορών και γραφημάτων, ενώ επιτρέπει τη σύγκριση δεδομένων ανά τμήμα, διδάσκοντα ή μάθημα, προσφέροντας πολύτιμες πληροφορίες για λήψη τεκμηριωμένων αποφάσεων.

Τα πλεονεκτήματα της πλατφόρμας Watermark περιλαμβάνουν την πλήρη αυτοματοποίηση των διαδικασιών συλλογής δεδομένων, την εύκολη πρόσβαση στα αποτελέσματα μέσω δυναμικών dashboards, και την ισχυρή ενσωμάτωση με άλλες πλατφόρμες εκπαίδευσης. Προσφέρει εξειληγμένα εργαλεία ανάλυσης δεδομένων και δυνατότητες εξατομικευμένων αναφορών, ενώ υποστηρίζει και την αποστολή υπενθυμίσεων στους φοιτητές για τη συμπλήρωση των ερωτηματολογίων.

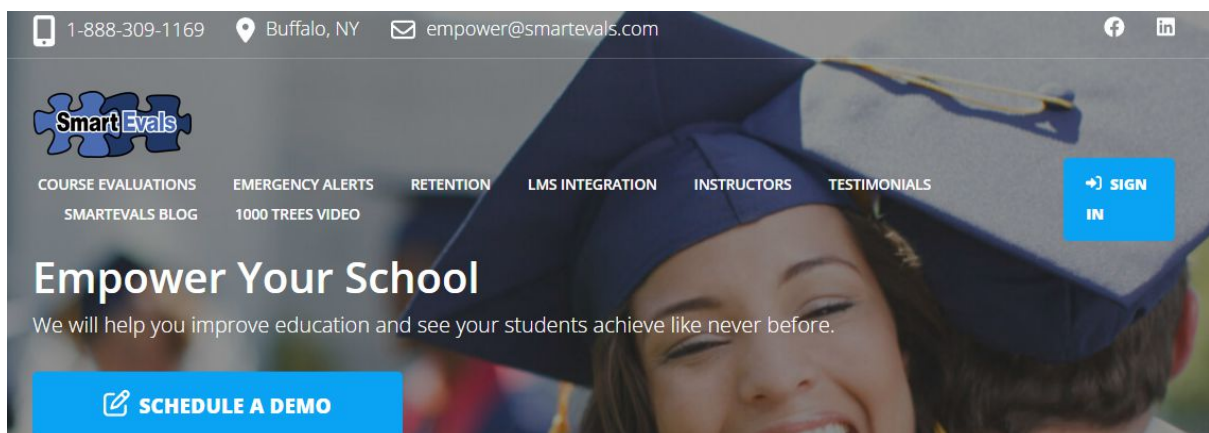
Το βασικό μειονέκτημα του Watermark είναι το υψηλό κόστος απόκτησης και συνδρομής, που το καθιστά δύσκολα προσιτό για μικρότερους εκπαιδευτικούς φορείς. Η μεγάλη ποικιλία λειτουργιών και παραμετροποιήσεων αν και ισχυρό πλεονέκτημα για μεγάλους οργανισμούς, μπορεί να καθυστερεί την αρχική εγκατάσταση και να απαιτεί εξειδικευμένη εκπαίδευση για την πλήρη αξιοποίηση του συστήματος.

Σε σύγκριση με τη δική μας πλατφόρμα, το Watermark προσφέρει πιο εξελιγμένες δυνατότητες ανάλυσης και διασύνδεσης με άλλα συστήματα, αλλά αυτό συνοδεύεται από σημαντικό οικονομικό και διαχειριστικό κόστος. Η πλατφόρμα που αναπτύξαμε είναι πιο απλή και ευέλικτη, ιδανική για οργανισμούς που αναζητούν άμεση υλοποίηση χωρίς υψηλές απαιτήσεις σε πόρους και χρόνο εκμάθησης. Το σύστημά μας παρέχει όλα τα βασικά εργαλεία συλλογής και ανάλυσης αξιολογήσεων με τρόπο φιλικό προς τον χρήστη και με χαμηλότερο κόστος και είναι μια πρακτική λύση για μικρές και μεσαίες εκπαιδευτικές μονάδες.

2.3 SmartEvals [3]

Το SmartEvals είναι μια σύγχρονη πλατφόρμα συλλογής και ανάλυσης αξιολογήσεων που εστιάζει στη διευκόλυνση της διαδικασίας αξιολόγησης διδασκόντων και μαθημάτων. Σχεδιάστηκε με έμφαση στην απλότητα χρήσης και την υψηλή συμμετοχή των φοιτητών. Η πλατφόρμα προσφέρει ένα φιλικό και εύχρηστο περιβάλλον τόσο για τους φοιτητές που υποβάλλουν αξιολογήσεις όσο και για τους διδάσκοντες και τα στελέχη διοίκησης που αναλύουν τα αποτελέσματα. Χρησιμοποιείται ευρέως σε πανεπιστήμια και κολέγια, προσφέροντας μια ολοκληρωμένη λύση για τη συλλογή σχολίων και τη δημιουργία αναφορών που βοηθούν στη βελτίωση της ποιότητας της εκπαίδευσης.

Η κύρια χρήση του SmartEvals είναι η συλλογή αξιολογήσεων για τα μαθήματα και τους διδάσκοντες, επιτρέποντας στους φοιτητές να υποβάλλουν εύκολα τις απαντήσεις τους μέσω ηλεκτρονικών συσκευών. Το σύστημα δίνει τη δυνατότητα δημιουργίας ερωτηματολογίων με διάφορους τύπους ερωτήσεων και παρέχει εργαλεία για την αποστολή υπενθυμίσεων στους φοιτητές, αυξάνοντας τα ποσοστά συμμετοχής. Προσφέρει αναλυτικές αναφορές που βοηθούν τους διδάσκοντες να κατανοήσουν τις ανάγκες και τις απόψεις των φοιτητών και να βελτιώσουν τη διδακτική τους πρακτική.



Εικόνα 2.3 SmartEvals Evaluations

[<https://info.smartevals.com/>]

Η χρησιμότητα του SmartEvals είναι εμφανής στην ταχύτητα και την απλότητα της διαδικασίας αξιολόγησης. Οι διοικήσεις των ιδρυμάτων μπορούν να παρακολουθούν άμεσα τα ποσοστά συμμετοχής, να βλέπουν συγκεντρωτικά αποτελέσματα και να λαμβάνουν χρήσιμες αναλύσεις για τη λήψη αποφάσεων. Η πλατφόρμα παρέχει επίσης τη δυνατότητα ανάλυσης τάσεων με την πάροδο του χρόνου, βοηθώντας τα ιδρύματα να εντοπίζουν μακροπρόθεσμες βελτιώσεις ή προβλήματα.

Τα πλεονεκτήματα του SmartEvals περιλαμβάνουν την πολύ καλή εμπειρία χρήστη, την εύκολη και γρήγορη διαδικασία αξιολόγησης και την ενίσχυση της συμμετοχής μέσω στοχευμένων ειδοποιήσεων. Επιπλέον, το σύστημα υποστηρίζει τη δημιουργία ευέλικτων ερωτηματολογίων και παρέχει κατανοητές και οπτικά ελκυστικές αναφορές. Ένα ακόμα σημαντικό πλεονέκτημα είναι η δυνατότητα ενσωμάτωσης με υπάρχοντα εκπαιδευτικά συστήματα και πλατφόρμες μάθησης.

Το κυριότερο μειονέκτημα του SmartEvals είναι ότι, παρά την απλότητά του, οι προηγμένες δυνατότητες ανάλυσης δεδομένων είναι περιορισμένες σε σχέση με άλλες μεγαλύτερες πλατφόρμες, όπως το Blue ή το Watermark. Η πλατφόρμα είναι εύχρηστη αλλά κάποιες πιο σύνθετες ανάγκες ανάλυσης ή εξειδικευμένες αναφορές απαιτούν επιπλέον εργαλεία ή πρόσθετο κόστος.

Σε σύγκριση με τη πλατφόρμα μας το SmartEvals δίνει έμφαση στην υψηλή συμμετοχή φοιτητών και την απλότητα, χαρακτηριστικά που συναντώνται και στο δικό μας σύστημα. Η δική μας πλατφόρμα έχει το πλεονέκτημα του πλήρους ελέγχου από τον οργανισμό που τη χρησιμοποιεί χωρίς εξαρτήσεις από τρίτες εταιρείες και χωρίς πρόσθετα κόστη συνδρομών. Το δικό μας σύστημα δίνει ιδιαίτερη έμφαση στην ανάλυση των δεδομένων μέσω διαδραστικών γραφημάτων και πλήρους παρουσίασης των σχολίων, παρέχοντας μια συνολική και προσαρμοσμένη εικόνα των αποτελεσμάτων σύμφωνα με τις ανάγκες του κάθε εκπαιδευτικού οργανισμού.

Κεφάλαιο 3ο: Γλώσσες προγραμματισμού και χρήσεις

3.1 Γλώσσα Προγραμματισμού Backend: Python

Η Python είναι μια υψηλού επιπέδου γλώσσα προγραμματισμού γενικού σκοπού που σχεδιάστηκε από τον Guido van Rossum και κυκλοφόρησε για πρώτη φορά το 1991. Χαρακτηρίζεται από τη σαφή σύνταξή της που βασίζεται σε εσοχές και υποστηρίζει πολλαπλά παραδείγματα προγραμματισμού, όπως αντικειμενοστραφή, διαδικαστικό και λειτουργικό [4-8].

Η Python χρησιμοποιείται ευρέως σε διάφορους τομείς της πληροφορικής και της επιστήμης, όπως:

- Ανάπτυξη Ιστοσελίδων και Εφαρμογών Ιστού: Με τη βοήθεια πλαισίων όπως το Flask και το Django.
- Ανάλυση Δεδομένων και Επιστήμη Δεδομένων: Χρησιμοποιείται για την επεξεργασία, ανάλυση και οπτικοποίηση δεδομένων.
- Μηχανική Μάθηση και Τεχνητή Νοημοσύνη: Με βιβλιοθήκες όπως το TensorFlow και το scikit-learn.
- Αυτοματοποίηση Εργασιών και Σενάρια (Scripting): Για την αυτοματοποίηση επαναλαμβανόμενων εργασιών.
- Ανάπτυξη Επιτραπέζιων Εφαρμογών: Με εργαλεία όπως το Tkinter.
- Εκπαίδευση και Διδασκαλία Προγραμματισμού: Λόγω της απλής σύνταξής της, είναι ιδανική για αρχάριους.

Η ευελιξία της Python την καθιστά κατάλληλη για:

- Εφαρμογές Ιστού: Δημιουργία δυναμικών ιστοσελίδων και υπηρεσιών.
- Ανάλυση και Οπτικοποίηση Δεδομένων: Επεξεργασία μεγάλων όγκων δεδομένων και δημιουργία γραφημάτων.
- Ανάπτυξη Παιχνιδιών: Με τη χρήση βιβλιοθηκών όπως το Pygame.
- Εφαρμογές για Κινητά: Αν και λιγότερο συχνή, υπάρχουν εργαλεία που επιτρέπουν την ανάπτυξη εφαρμογών για κινητές συσκευές.
- Επιστημονικές Εφαρμογές: Χρησιμοποιείται σε τομείς όπως η βιοπληροφορική και η αστρονομία.

Η γλώσσα προγραμματισμού Python παρουσιάζει σημαντικά πλεονεκτήματα που την έχουν καταστήσει ιδιαίτερα δημοφιλή τα τελευταία χρόνια. Το βασικό της πλεονέκτημα είναι η απλή και ευανάγνωστη σύνταξή της, η οποία διευκολύνει τόσο τους νέους όσο και τους έμπειρους προγραμματιστές να γράφουν και να συντηρούν κώδικα με ευκολία. Η Python διαθέτει μια τεράστια κοινότητα υποστήριξης και ένα πλήθος από έτοιμες βιβλιοθήκες και εργαλεία, καλύπτοντας σχεδόν κάθε ανάγκη στον χώρο του προγραμματισμού. Η ανεξαρτησία από πλατφόρμες και η εύκολη ενσωμάτωση με άλλες γλώσσες και η χρήση της σε πληθώρα τομέων, από την ανάπτυξη ιστοσελίδων μέχρι την τεχνητή νοημοσύνη και την ανάλυση δεδομένων, την καθιστούν μία από τις πιο ευέλικτες γλώσσες προγραμματισμού.

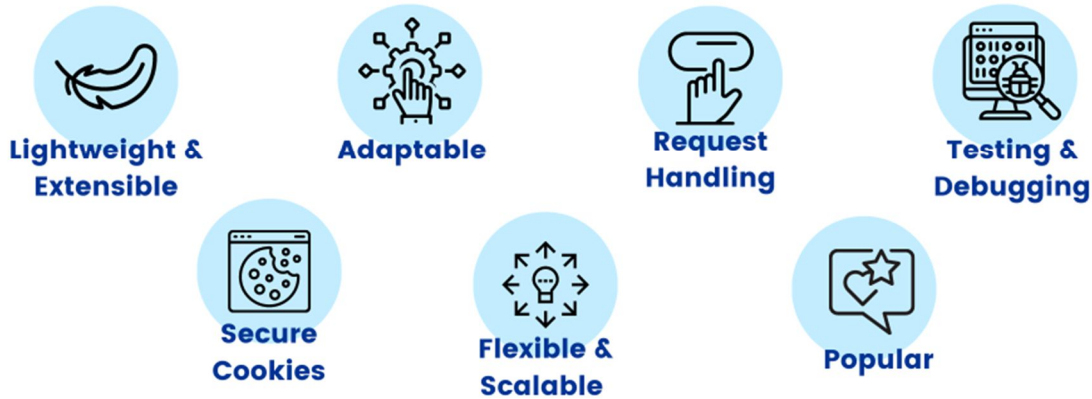
Ένα από τα βασικά της προβλήματα είναι η σχετικά χαμηλή ταχύτητα εκτέλεσης ειδικά σε σύγκριση με γλώσσες όπως η C ή η Java, γεγονός που την καθιστά λιγότερο κατάλληλη για εφαρμογές που απαιτούν υψηλές επιδόσεις σε πραγματικό χρόνο. Επίσης η Python καταναλώνει περισσότερη μνήμη λόγω της δυναμικής διαχείρισης τύπων δεδομένων, κάτι που μπορεί να επηρεάσει την αποδοτικότητα σε συστήματα περιορισμένων πόρων. Η διαχείριση της πολυνηματικότητας (multithreading) περιορίζεται από τον Global Interpreter Lock (GIL), που δεν επιτρέπει την πλήρη αξιοποίηση των επεξεργαστών σε εφαρμογές που βασίζονται σε παράλληλη εκτέλεση. Ενώ υπάρχουν λύσεις για ανάπτυξη εφαρμογών κινητών συσκευών η Python δεν θεωρείται η ιδανική επιλογή για αυτόν τον σκοπό.

3.2 Framework Backend: Flask

Το Flask είναι ένα ελαφρύ και ευέλικτο πλαίσιο ανάπτυξης διαδικτυακών εφαρμογών που βασίζεται στη γλώσσα προγραμματισμού Python. Σχεδιάστηκε για να είναι απλό στη χρήση και κατάλληλο για προγραμματιστές που θέλουν να δημιουργήσουν γρήγορα και εύκολα web εφαρμογές. Σε αντίθεση με άλλα, πιο περίπλοκα frameworks, το Flask προσφέρει μόνο τα βασικά εργαλεία για την ανάπτυξη μιας εφαρμογής και δίνει τη δυνατότητα στον προγραμματιστή να προσθέτει επιπλέον λειτουργίες ανάλογα με τις ανάγκες του έργου. Αυτό το χαρακτηριστικό το κάνει ιδανικό για μικρές και μεσαίες εφαρμογές, αλλά και για έργα που απαιτούν ευελιξία και προσαρμοστικότητα [9-12].



Flask Key Features



> hackr.io

Εικόνα 3.1: Πλεονεκτήματα flask python

[<https://cdn.hackr.io/uploads/posts/attachments/1675527683IH32uPoaem.png>]

Η χρησιμότητα του Flask είναι μεγάλη, καθώς επιτρέπει την άμεση ανάπτυξη web εφαρμογών χωρίς περίπλοκες ρυθμίσεις. Οι βασικές λειτουργίες του καλύπτουν τη δημιουργία διαδρομών για τις σελίδες της εφαρμογής, τη διαχείριση αιτημάτων από τον χρήστη και την παραγωγή δυναμικού περιεχομένου μέσω templates. Το Flask χρησιμοποιείται ευρέως για την ανάπτυξη απλών ιστοσελίδων, APIs, διαχειριστικών εργαλείων και εφαρμογών που συνδέονται με βάσεις δεδομένων, προσφέροντας μια γρήγορη και αποδοτική λύση σε όσους δεν χρειάζονται ένα πλήρως δομημένο και περίπλοκο περιβάλλον.

Η απλότητα και η ευκολία χρήσης αποτελούν σημαντικά πλεονεκτήματα του Flask. Χάρη στην καθαρή και κατανοητή δομή του, ακόμα και αρχάριοι προγραμματιστές μπορούν να το μάθουν εύκολα και να δημιουργήσουν λειτουργικές εφαρμογές σε σύντομο χρονικό διάστημα. Η μεγάλη κοινότητα χρηστών και η πληθώρα διαθέσιμων παραδειγμάτων και επεκτάσεων διευκολύνουν ακόμα περισσότερο την ανάπτυξη εφαρμογών. Παρόλα αυτά, η απλότητα του μπορεί σε κάποιες περιπτώσεις να θεωρηθεί και μειονέκτημα. Για παράδειγμα, σε πολύπλοκες και μεγάλης κλίμακας εφαρμογές, το Flask δεν παρέχει έτοιμες λύσεις για την οργάνωση του κώδικα και την ασφάλεια, με αποτέλεσμα να απαιτείται περισσότερη δουλειά από τον προγραμματιστή για να διασφαλίσει την ορθή λειτουργία και τη σταθερότητα της εφαρμογής.

Το Flask είναι ένα ιδανικό εργαλείο για έργα που απαιτούν ευελιξία, γρήγορη ανάπτυξη και απλότητα, προσφέροντας παράλληλα τη δυνατότητα να επεκταθεί και να καλύψει πιο σύνθετες ανάγκες όταν αυτό

κριθεί απαραίτητο. Με την επιλογή του Flask για το backend της πλατφόρμας μας, εξασφαλίσαμε γρήγορη ανάπτυξη, εύκολη συντήρηση του κώδικα και ένα σταθερό περιβάλλον που ανταποκρίνεται πλήρως στις απαιτήσεις του έργου.

Για να ξεκινήσει κανείς να χρησιμοποιεί το Flask, θα πρέπει να έχει ήδη εγκατεστημένη τη γλώσσα προγραμματισμού Python στον υπολογιστή του. Αφού επιβεβαιωθεί ότι η Python είναι εγκατεστημένη, το επόμενο βήμα είναι να εγκατασταθεί το ίδιο το Flask. Αυτό γίνεται πολύ εύκολα μέσω της γραμμής εντολών. Ανοίγουμε τη γραμμή εντολών και πληκτρολογούμε την εντολή `pip install Flask`. Η εντολή αυτή κατεβάζει και εγκαθιστά αυτόματα το Flask στο σύστημά μας.

Αφού εγκατασταθεί το Flask, μπορούμε να προχωρήσουμε στη δημιουργία της πρώτης μας απλής διαδικτυακής εφαρμογής. Δημιουργούμε έναν νέο φάκελο στον υπολογιστή μας, όπου θα τοποθετήσουμε τα αρχεία της εφαρμογής. Μέσα σε αυτόν τον φάκελο, δημιουργούμε ένα αρχείο με όνομα `app.py`. Αυτό το αρχείο θα περιέχει τον βασικό κώδικα της εφαρμογής μας.

Στο αρχείο `app.py` γράφουμε τον παρακάτω κώδικα:

```
from flask import Flask

app = Flask(__name__)

@app.route('/')
def home():
    return 'Καλώς ήρθατε στην πρώτη σας Flask εφαρμογή!'

if __name__ == '__main__':
    app.run(debug=True)
```

Ο παραπάνω κώδικας δημιουργεί μια πολύ απλή εφαρμογή. Όταν ο χρήστης επισκέπτεται τη διεύθυνση `http://127.0.0.1:5000/` στον browser του, θα δει το μήνυμα "Καλώς ήρθατε στην πρώτη σας Flask εφαρμογή!". Η γραμμή `@app.route('/')` δηλώνει ότι η συγκεκριμένη συνάρτηση `home` θα εκτελείται όταν ο χρήστης επισκέπτεται την αρχική σελίδα της εφαρμογής. Η τελευταία γραμμή `app.run(debug=True)` εκκινεί την εφαρμογή και την θέτει σε λειτουργία.

Για να ξεκινήσουμε την εφαρμογή μας, ανοίγουμε ξανά τη γραμμή εντολών, πηγαίνουμε στον φάκελο όπου βρίσκεται το αρχείο `app.py` και εκτελούμε την εντολή `python app.py`. Αν όλα έχουν γίνει σωστά, στην οθόνη θα εμφανιστεί μήνυμα ότι ο διακομιστής εκτελείται και μπορούμε πλέον να επισκεφτούμε την εφαρμογή μας μέσω του browser.

Με αυτόν τον απλό τρόπο, δημιουργήσαμε και τρέξαμε την πρώτη μας εφαρμογή με Flask. Από αυτό το σημείο και μετά, μπορούμε να επεκτείνουμε την εφαρμογή μας προσθέτοντας περισσότερες σελίδες, λειτουργίες και δυναμικό περιεχόμενο, δημιουργώντας έτσι ολοκληρωμένες διαδικτυακές εφαρμογές.

3.3 Βάση Δεδομένων: MySQL

Η MySQL είναι ένα από τα πιο δημοφιλή συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) παγκοσμίως. Χρησιμοποιείται ευρέως τόσο σε μικρές όσο και σε μεγάλες εφαρμογές λόγω της σταθερότητας, της υψηλής απόδοσης και της ευκολίας χρήσης που έχει. Πρόκειται για μια ανοιχτού κώδικα λύση όπου επιτρέπει στους προγραμματιστές να αποθηκεύουν, να διαχειρίζονται και να ανακτούν δεδομένα με οργανωμένο και αποδοτικό τρόπο, χρησιμοποιώντας τη γλώσσα ερωτημάτων SQL (Structured Query Language) [13-14].

Η χρησιμότητα της MySQL στο πλαίσιο της πλατφόρμας αξιολόγησης είναι καθοριστική, καθώς αναλαμβάνει την αποθήκευση όλων των κρίσιμων δεδομένων του συστήματος. Σε αυτήν αποθηκεύονται τα στοιχεία των χρηστών, οι φόρμες αξιολόγησης, οι ερωτήσεις και οι απαντήσεις των φοιτητών, καθώς και οι μοναδικοί κωδικοί που χρησιμοποιούνται για την υποβολή των αξιολογήσεων. Με αυτόν τον τρόπο, η MySQL εξασφαλίζει την ακεραιότητα και την ταχύτατη ανάκτηση των δεδομένων και δίνει άμεση απόκριση σε όλα τα αιτήματα της πλατφόρμας.

Η MySQL χρησιμοποιήθηκε για την υλοποίηση της συγκεκριμένης πλατφόρμας διότι προσφέρει σημαντικά πλεονεκτήματα. Είναι ιδιαίτερα γρήγορη σε αναζητήσεις και διαχείριση δεδομένων, υποστηρίζει μεγάλο αριθμό ταυτόχρονων συνδέσεων και παρέχει εύκολα εργαλεία για δημιουργία αντιγράφων ασφαλείας και αποκατάσταση δεδομένων. Η απλότητα στη χρήση της, μέσω γραφικών εργαλείων διαχείρισης όπως το phpMyAdmin, διευκολύνει την ανάπτυξη και τη συντήρηση της βάσης δεδομένων. Ένα ακόμα πλεονέκτημα είναι η μεγάλη υποστήριξη από την κοινότητα, που προσφέρει συνεχή αναβαθμίσεις, διορθώσεις ασφαλείας και τεκμηρίωση.

Σε πολύ μεγάλης κλίμακας εφαρμογές με πολύπλοκες σχέσεις δεδομένων, η απόδοσή της μπορεί να μειωθεί αν δεν έχει γίνει σωστός σχεδιασμός και βελτιστοποίηση των ερωτημάτων.

3.4 Frontend Βιβλιοθήκη: Bootstrap

Το Bootstrap είναι μια από τις πιο δημοφιλείς και διαδεδομένες βιβλιοθήκες σχεδίασης διεπαφών χρήστη (UI) για την ανάπτυξη ιστοσελίδων και εφαρμογών. Δημιουργήθηκε από την ομάδα του Twitter

και κυκλοφόρησε ως έργο ανοιχτού κώδικα. Σήμερα χρησιμοποιείται σε χιλιάδες έργα σε όλο τον κόσμο λόγω της ευκολίας που προσφέρει στον σχεδιασμό μοντέρνων, ελκυστικών και responsive ιστοσελίδων, χωρίς να απαιτείται από τον προγραμματιστή βαθιά γνώση CSS ή σύνθετων τεχνικών σχεδίασης [15-16].

Η κύρια χρησιμότητα του Bootstrap είναι ότι παρέχει έτοιμα δομικά στοιχεία και στυλιστικά πρότυπα, τα οποία μπορούν να χρησιμοποιηθούν για την εύκολη δημιουργία ιστοσελίδων που προσαρμόζονται αυτόματα σε διαφορετικά μεγέθη οθόνης, όπως υπολογιστές, tablets και κινητά τηλέφωνα. Αυτό το χαρακτηριστικό, γνωστό και ως responsive design, είναι ιδιαίτερα σημαντικό στη σημερινή εποχή, όπου οι χρήστες χρησιμοποιούν διαφορετικές συσκευές για την πρόσβαση στο διαδίκτυο. Μέσω του Bootstrap, είναι δυνατή η ταχεία ανάπτυξη σελίδων με μοντέρνα εμφάνιση, χωρίς να απαιτείται μεγάλος χρόνος σχεδίασης ή πολύπλοκος προγραμματισμός CSS.

Η πλατφόρμα αξιολόγησης εκμεταλλεύτηκε στο έπακρο τις δυνατότητες του Bootstrap για την ανάπτυξη ενός καθαρού, ευχάριστου και λειτουργικού περιβάλλοντος χρήστη. Όλες οι βασικές σελίδες, όπως οι φόρμες αξιολόγησης, τα dashboards, τα στατιστικά και οι φόρμες εισαγωγής δεδομένων, σχεδιάστηκαν με την υποστήριξη του Bootstrap, εξασφαλίζοντας ότι η πλοήγηση είναι εύκολη και η εμφάνιση των δεδομένων καθαρή και οργανωμένη.

Τα βασικά πλεονεκτήματα του Bootstrap είναι η ταχύτητα ανάπτυξης, η συμβατότητα με όλους τους σύγχρονους browsers, η μεγάλη ποικιλία έτοιμων components όπως κουμπιά, φόρμες, πίνακες και κάρτες, και η ενσωμάτωση μοντέρνων τεχνικών σχεδίασης. Διαθέτει ένα εύχρηστο σύστημα διάταξης (Grid System), το οποίο βοηθάει στη δημιουργία σελίδων με καλή οργάνωση και σωστή προσαρμογή σε όλες τις αναλύσεις οθόνης.

Η υπερβολική εξάρτηση από τα έτοιμα στυλ του Bootstrap μπορεί να οδηγήσει σε ομοιόμορφες και συχνά επαναλαμβανόμενες εμφανίσεις ιστοσελίδων. Σε περιπτώσεις που απαιτείται ιδιαίτερα εξατομικευμένος σχεδιασμός, είναι συχνά αναγκαίο να παρακάμπτεται το Bootstrap και να γίνεται εκτενής χρήση προσαρμοσμένων CSS γεγονός που μειώνει τα πλεονεκτήματα της απλότητας.

Το Bootstrap αποτέλεσε μια εξαιρετική επιλογή για το frontend της πλατφόρμας αξιολόγησης καθώς βοήθησε στη δημιουργία ενός μοντέρνου και εύχρηστου περιβάλλοντος για τους τελικούς χρήστες διατηρώντας παράλληλα την ευκολία στη συντήρηση και την επεκτασιμότητα του συστήματος.

3.5 Βιβλιοθήκη Γραφημάτων: Chart.js

Το Chart.js είναι μια δημοφιλής βιβλιοθήκη ανοιχτού κώδικα που χρησιμοποιείται για την οπτικοποίηση δεδομένων σε διαδικτυακές εφαρμογές. Είναι γραμμένη σε JavaScript και επιτρέπει την

εύκολη δημιουργία δυναμικών και διαδραστικών γραφημάτων μέσα σε ιστοσελίδες. Χρησιμοποιεί το στοιχείο `<canvas>` της HTML5, μέσω του οποίου αποδίδονται τα γραφικά, προσφέροντας υψηλή ποιότητα απεικόνισης και καλή απόδοση ακόμη και με μεγάλα σύνολα δεδομένων [17-18].

Η βιβλιοθήκη Chart.js είναι ιδιαίτερα χρήσιμη σε εφαρμογές που απαιτούν την παρουσίαση στατιστικών στοιχείων και την ανάλυση δεδομένων με γραφικό τρόπο. Μέσω των γραφημάτων, οι χρήστες μπορούν να κατανοήσουν καλύτερα τα αποτελέσματα και τις τάσεις που προκύπτουν από τα δεδομένα, χωρίς να χρειάζεται να αναλύουν μεγάλους πίνακες αριθμών. Η βιβλιοθήκη υποστηρίζει μια μεγάλη ποικιλία τύπων γραφημάτων, όπως ραβδογράμματα, κυκλικά διαγράμματα, γραφήματα γραμμών, ραντάρ και πολικών διαγραμμάτων, καλύπτοντας έτσι διαφορετικές ανάγκες παρουσίασης.

Στην πλατφόρμα αξιολόγησης, το Chart.js χρησιμοποιήθηκε για την παρουσίαση των στατιστικών αποτελεσμάτων των αξιολογήσεων. Μέσω αυτής της βιβλιοθήκης, οι διδάσκοντες έχουν τη δυνατότητα να δουν άμεσα και οπτικά τις μέσες βαθμολογίες ανά ερώτηση, ανά κατηγορία, καθώς και τα συνολικά ποσοστά ικανοποίησης των φοιτητών. Τα γραφήματα παρέχουν διαδραστική λειτουργικότητα, επιτρέποντας την εμφάνιση λεπτομερειών κατά την πλοήγηση του χρήστη επάνω σε κάθε στοιχείο του γραφήματος.

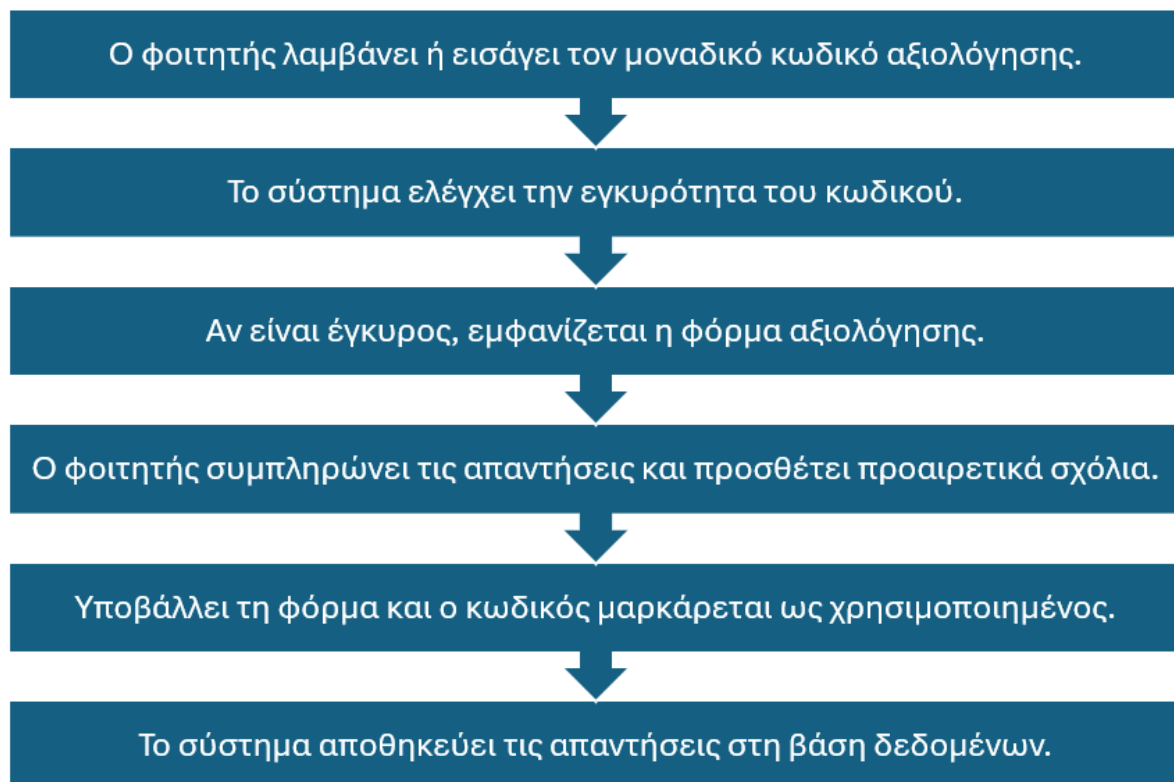
Τα βασικά πλεονεκτήματα του Chart.js είναι η ευκολία χρήσης, η ενσωμάτωσή του σε ιστοσελίδες χωρίς πολύπλοκες διαδικασίες και η άμεση απόδοση οπτικά ελκυστικών και καλοσχεδιασμένων γραφημάτων. Η βιβλιοθήκη προσφέρει επίσης πολλές δυνατότητες προσαρμογής, ώστε ο προγραμματιστής να μπορεί να διαμορφώσει την εμφάνιση των γραφημάτων σύμφωνα με τις ανάγκες της εφαρμογής. Η βιβλιοθήκη έχει πολύ καλή τεκμηρίωση και υποστηρίζεται από μια μεγάλη κοινότητα χρηστών.

Όπως κάθε εργαλείο το Chart.js έχει και ορισμένα μειονεκτήματα. Για πιο περίπλοκες ανάγκες ανάλυσης και παρουσίασης δεδομένων, ενδέχεται να απαιτούνται επιπλέον προσθήκες ή χειροκίνητος προγραμματισμός. Σε περιπτώσεις που διαχειριζόμαστε πολύ μεγάλα σύνολα δεδομένων ή πολύπλοκες διαδραστικές λειτουργίες, η απόδοση της βιβλιοθήκης μπορεί να επηρεαστεί.

Η χρήση του Chart.js στην πλατφόρμα αξιολόγησης αποδείχθηκε εξαιρετικά χρήσιμη καθώς επέτρεψε την εύκολη παρουσίαση των στατιστικών αποτελεσμάτων με τρόπο κατανοητό και οπτικά ευχάριστο.

Κεφάλαιο 4ο: Η πλατφόρμα συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές

4.1 Οι διαδικασίες



Εικόνα 4.1: Διάγραμμα Ροής Υποβολής Αξιολόγησης (Student Flow)

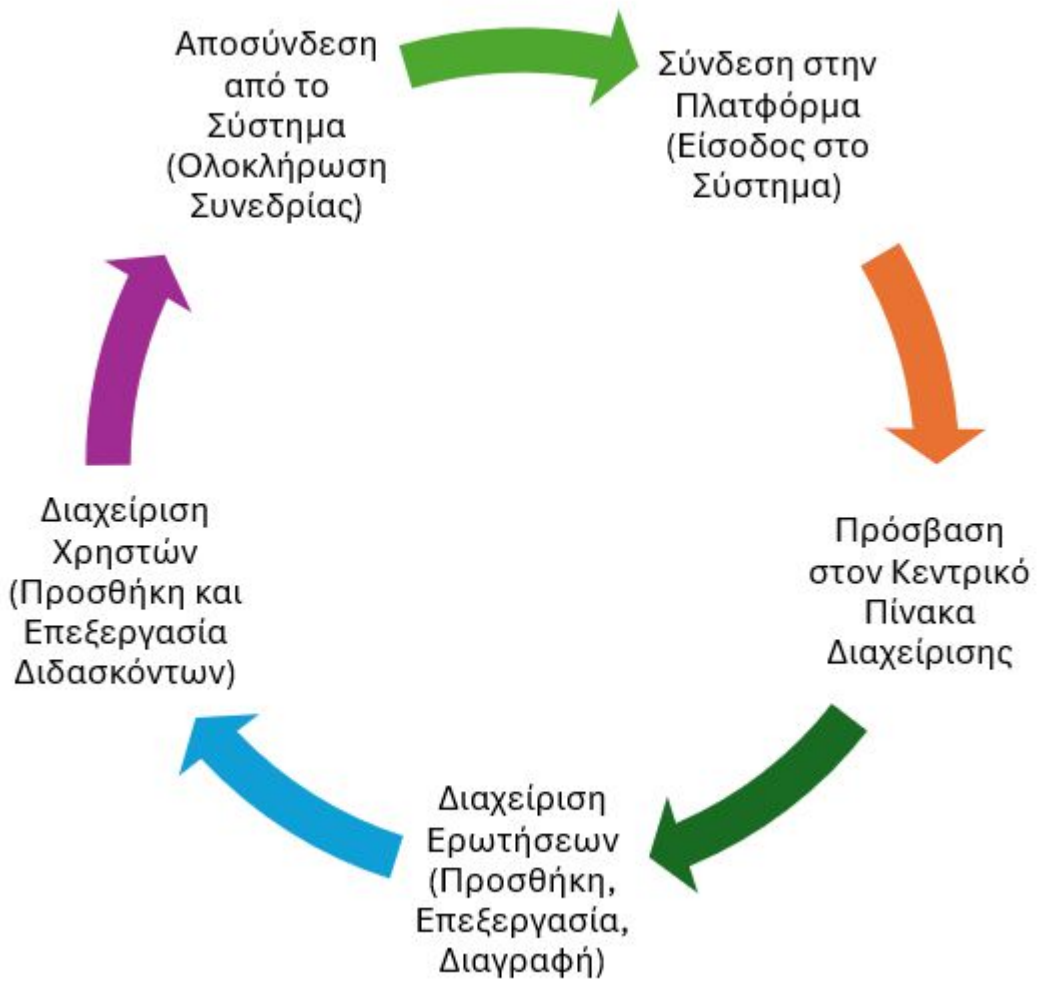
Η διαδικασία υποβολής αξιολόγησης από τον φοιτητή ξεκινά με την εισαγωγή ενός μοναδικού κωδικού που του έχει δοθεί. Το σύστημα ελέγχει την εγκυρότητα και τη διαθεσιμότητα του κωδικού, διασφαλίζοντας ότι δεν έχει ήδη χρησιμοποιηθεί. Αν ο κωδικός είναι έγκυρος, ο φοιτητής οδηγείται στη φόρμα αξιολόγησης και καλείται να απαντήσει στις διαθέσιμες ερωτήσεις και να προσθέσει προαιρετικά κάποιο σχόλιο. Μετά την ολοκλήρωση της διαδικασίας, η αξιολόγηση υποβάλλεται, ο κωδικός μαρκάρεται ως χρησιμοποιημένος και τα δεδομένα αποθηκεύονται με ασφάλεια στη βάση. Η διαδικασία ολοκληρώνεται με την εμφάνιση σχετικού μηνύματος επιτυχούς υποβολής.



Εικόνα 4.2: Διάγραμμα Ροής Διαχείρισης Αξιολογήσεων (Teacher Flow)

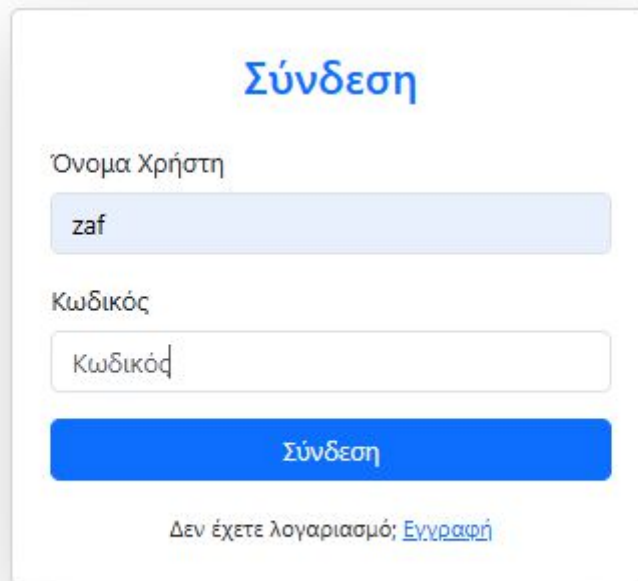
Η διαδικασία διαχείρισης αξιολογήσεων από τον διδάσκοντα ξεκινά με τη σύνδεσή του στο σύστημα και την πρόσβαση στον προσωπικό του πίνακα διαχείρισης. Ο διδάσκων μπορεί να δημιουργήσει νέες φόρμες αξιολόγησης, επιλέγοντας τις κατάλληλες ερωτήσεις από την διαθέσιμη δεξαμενή και ορίζοντας τον τίτλο, την περιγραφή και τις ημερομηνίες ισχύος της φόρμας. Μετά την οριστικοποίηση της φόρμας, έχει τη δυνατότητα να δημιουργήσει παρτίδες μοναδικών κωδικών για τους φοιτητές. Παράλληλα, μπορεί να παρακολουθεί τα στατιστικά των αξιολογήσεων μέσω διαδραστικών γραφημάτων και αναφορών, εξάγοντας χρήσιμα συμπεράσματα για τη διδασκαλία του.

Ο διαχειριστής του συστήματος, αφού πραγματοποιήσει είσοδο, αποκτά πρόσβαση στον κεντρικό πίνακα διαχείρισης, από όπου μπορεί να εκτελέσει όλες τις βασικές λειτουργίες του συστήματος. Έχει τη δυνατότητα να διαχειριστεί τη δεξαμενή των ερωτήσεων, προσθέτοντας νέες, επεξεργαζόμενος ή διαγράφοντας όσες δεν χρησιμοποιούνται. Μπορεί να παρακολουθεί τα στατιστικά αποτελέσματα όλων των αξιολογήσεων που έχουν πραγματοποιηθεί, προκειμένου να εξάγει χρήσιμες αναφορές για τη συνολική εικόνα της εκπαιδευτικής διαδικασίας. Ο διαχειριστής έχει επίσης πλήρη έλεγχο στη διαχείριση των χρηστών, προσθέτοντας νέους διδάσκοντες ή ενημερώνοντας τα στοιχεία των υπαρχόντων. Αφού ολοκληρώσει τις εργασίες του, μπορεί να αποσυνδεθεί και να κλείσει με ασφάλεια τη συνεδρία του.



Εικόνα 4.3: Διαδικασία Χρηστών Admin

4.2 Η διαδικτυακή πλατφόρμα



The image shows a login form titled "Σύνδεση" (Login). It contains two input fields: "Όνομα Χρήστη" (Username) with the value "zaf" and "Κωδικός" (Password) with the value "Κωδικός". Below the fields is a blue button labeled "Σύνδεση" (Login). At the bottom, there is a link that says "Δεν έχετε λογαριασμό; [Εγγραφή](#)" (Don't have an account? [Register](#)).

Εικόνα 4.4: Σύνδεση

Η οθόνη σύνδεσης (Εικόνα 4.4) αποτελεί την κύρια είσοδο των χρηστών στο σύστημα αξιολόγησης. Μέσω αυτής, οι διαχειριστές (admin) και οι διδάσκοντες (teachers) μπορούν να εισάγουν τα προσωπικά τους διαπιστευτήρια και να αποκτήσουν πρόσβαση στις λειτουργίες της πλατφόρμας.

Η φόρμα σύνδεσης περιλαμβάνει δύο βασικά πεδία:

- Email: Όπου ο χρήστης εισάγει τη διεύθυνση email που έχει δηλωθεί στο σύστημα.
- Κωδικός Πρόσβασης: Το πεδίο για την εισαγωγή του προσωπικού κωδικού ασφαλείας.

Υπάρχει επίσης κουμπί για την υποβολή των στοιχείων και έλεγχος εγκυρότητας για την αποφυγή λανθασμένων εισόδων. Σε περίπτωση εσφαλμένων στοιχείων, εμφανίζεται κατάλληλο μήνυμα λάθους.

Η οθόνη εγγραφής (Εικόνα 4.5) επιτρέπει τη δημιουργία νέου λογαριασμού στο σύστημα, συνήθως για διδάσκοντες ή διαχειριστές (ανάλογα με τα δικαιώματα που επιτρέπονται από τον admin). Η φόρμα εγγραφής περιλαμβάνει τα απαραίτητα πεδία για τη δημιουργία ενός νέου χρήστη:

- Όνομα και Επώνυμο: Για την καταγραφή των προσωπικών στοιχείων.
- Email: Για τη μοναδική ταυτοποίηση του χρήστη και μελλοντική επικοινωνία.
- Κωδικός Πρόσβασης και Επιβεβαίωση Κωδικού: Για τη δημιουργία και επιβεβαίωση ασφαλούς κωδικού πρόσβασης.

Το σύστημα εφαρμόζει κανόνες ασφαλείας για τον έλεγχο ισχυρών κωδικών και την αποφυγή διπλών εγγραφών με το ίδιο email. Μετά την επιτυχή συμπλήρωση της φόρμας, ο νέος χρήστης μπορεί να συνδεθεί άμεσα στο σύστημα και να ξεκινήσει να χρησιμοποιεί τις διαθέσιμες λειτουργίες.

The image shows a registration form titled "Εγγραφή" (Registration) in blue. It contains the following fields and elements:

- Όνομα Χρήστη** (Username): Input field containing "zaf".
- Όνομα** (Name): Input field containing "Όνομα".
- Επώνυμο** (Surname): Input field containing "Επώνυμο".
- Κωδικός** (Code): Input field containing "Κωδικός".
- Ρόλος** (Role): Dropdown menu with "Teacher" selected and a downward arrow.
- Εγγραφή** (Registration): A prominent green button.
- Below the button, there is a link: "Έχετε ήδη λογαριασμό; [Σύνδεση](#)".

Εικόνα 4.5: Εγγραφή

4.2.1 Ρόλος: Teacher – Διδάσκοντας

Η Εικόνα 4.6 παρουσιάζει την αρχική σελίδα του διδάσκοντα - Teacher Dashboard. Πρόκειται για την κεντρική σελίδα διαχείρισης από την οποία ο διδάσκων μπορεί να έχει μια συνοπτική εικόνα των δραστηριοτήτων του συστήματος και να πλοηγηθεί εύκολα στις βασικές λειτουργίες της πλατφόρμας.

Στο πάνω μέρος της σελίδας εμφανίζεται η γραμμή πλοήγησης με τις βασικές επιλογές:

- Αρχική Teacher: Επιστροφή στην κεντρική σελίδα του διδάσκοντα.
- Φόρμες Αξιολόγησης: Διαχείριση και δημιουργία νέων φορμών αξιολόγησης.

Στη δεξιά πλευρά της γραμμής εμφανίζεται το status σύνδεσης ως Teacher και η επιλογή Αποσύνδεση.

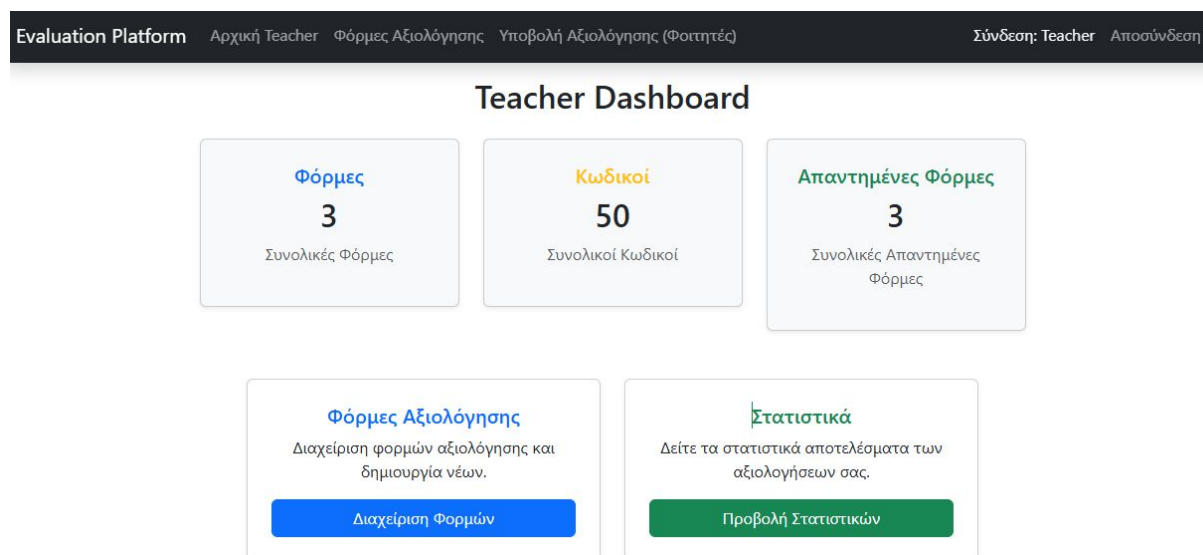
Στο κεντρικό τμήμα της σελίδας, υπάρχουν τρία συνοπτικά πλαίσια με σημαντικές πληροφορίες:

- Φόρμες (3): Δείχνει το συνολικό αριθμό των φορμών αξιολόγησης που έχει δημιουργήσει ο διδάσκων.
- Κωδικοί (50): Εμφανίζει το συνολικό αριθμό των κωδικών που έχουν παραχθεί για συμμετοχή στις αξιολογήσεις.
- Απαντημένες Φόρμες (3): Δείχνει πόσες από τις φόρμες αξιολόγησης έχουν ήδη απαντηθεί από τους φοιτητές.

Κάτω από τα συνοπτικά στατιστικά, υπάρχουν δύο βασικά κουμπιά ενεργειών:

- Διαχείριση Φορμών (μπλε κουμπί): Μεταφέρει τον χρήστη στη σελίδα όπου μπορεί να δημιουργήσει νέες φόρμες ή να επεξεργαστεί τις υπάρχουσες.
- Προβολή Στατιστικών (πράσινο κουμπί): Οδηγεί στη σελίδα με τα στατιστικά αποτελέσματα των αξιολογήσεων, όπου ο διδάσκων μπορεί να δει τα αναλυτικά αποτελέσματα και τα γραφήματα.

Η διάταξη της σελίδας είναι καθαρή και λειτουργική με έμφαση στην άμεση ενημέρωση του διδάσκοντα για την κατάσταση των αξιολογήσεων και την εύκολη πρόσβαση στις κύριες λειτουργίες του συστήματος.



Εικόνα 4.6: Αρχική Σελίδα το Διδάσκοντα - Teacher Dashboard

Η εικόνα 4.7 παρουσιάζει τη σελίδα Λίστα Φορμών Αξιολόγησης, στην οποία ο διδάσκων μπορεί να δει συγκεντρωτικά όλες τις φόρμες αξιολόγησης που έχει δημιουργήσει. Κάθε φόρμα εμφανίζεται με τη μορφή ξεχωριστής κάρτας, όπου περιέχονται βασικές πληροφορίες για το μάθημα και επιλογές διαχείρισης.

Στο πάνω μέρος της σελίδας υπάρχει το πράσινο κουμπί Δημιουργία Νέας Φόρμας, που επιτρέπει στον διδάσκοντα να δημιουργήσει μια καινούρια φόρμα αξιολόγησης.

Κάθε κάρτα περιέχει τις παρακάτω πληροφορίες και λειτουργίες:

- Τίτλος Μαθήματος: Παρουσιάζεται με έντονα μπλε γράμματα, ώστε να ξεχωρίζει.
- Περιγραφή Μαθήματος: Μικρό κείμενο που διευκρινίζει σε ποιο μάθημα αφορά η φόρμα.
- Ημερομηνία Έναρξης και Λήξης: Οι ημερομηνίες κατά τις οποίες οι φοιτητές μπορούν να υποβάλουν την αξιολόγησή τους.
- Κατάσταση Φόρμας: Μπορεί να είναι είτε Οριστικοποιημένη (σε πράσινο χρώμα), είτε Πρόχειρη (σε πορτοκαλί χρώμα), ανάλογα με το αν έχει κλείσει η διαδικασία επεξεργασίας της φόρμας.

Κάτω από αυτές τις πληροφορίες, υπάρχουν τέσσερα κουμπιά που παρέχουν τις βασικές ενέργειες διαχείρισης για κάθε φόρμα:

- Επεξεργασία (μπλε κουμπί): Επιτρέπει την τροποποίηση της φόρμας αν δεν έχει οριστικοποιηθεί ακόμα.
- Διαγραφή (κόκκινο κουμπί): Διαγράφει τη φόρμα, εφόσον δεν υπάρχουν ήδη απαντήσεις.
- Κωδικοί (κίτρινο κουμπί): Οδηγεί στη σελίδα διαχείρισης των κωδικών συμμετοχής, όπου ο διδάσκων μπορεί να δημιουργήσει ή να δει τους υπάρχοντες κωδικούς.
- Στατιστικά (γαλάζιο κουμπί): Παρουσιάζει τα στατιστικά αποτελέσματα της συγκεκριμένης φόρμας αξιολόγησης, δίνοντας τη δυνατότητα προβολής αναφορών και γραφημάτων.

Η σελίδα αυτή παρέχει μια πλήρη και εύχρηστη επισκόπηση όλων των φορμών, επιτρέποντας στον διδάσκοντα να παρακολουθεί την κατάσταση των αξιολογήσεων, να πραγματοποιεί τις απαραίτητες ενέργειες και να αντλεί πολύτιμα συμπεράσματα από τα στατιστικά αποτελέσματα.

Λίστα Φορμών Αξιολόγησης

+ Δημιουργία Νέας Φόρμας

The screenshot shows three course cards under the heading 'Λίστα Φορμών Αξιολόγησης'. At the top is a green button with a plus sign and the text 'Δημιουργία Νέας Φόρμας'. The first card is for 'Μάθημα Ρομποτική' with start date 14-05-2025 and end date 15-05-2025, status 'Οριστικοποιημένη'. The second card is for 'Μάθημα Ηλεκτρονικά Κυκλώματα' with start date 02-05-2025 and end date 25-05-2025, status 'Πρόχειρη'. The third card is for 'Μάθημα Δομημένος Προγραμματισμός' with start date 24-05-2025 and end date 26-05-2025, status 'Πρόχειρη'. Each card has four buttons: 'Επεξεργασία' (blue), 'Διαγραφή' (red), 'Κωδικοί' (yellow), and 'Στατιστικά' (cyan).

Εικόνα 4.7: Σελίδα Λίστα Φορμών Αξιολόγησης

Δημιουργία Νέας Φόρμας

Τίτλος

Περιγραφή

Ημερομηνία Έναρξης

Ημερομηνία Λήξης

Ερωτήσεις (ανά Πεδίο)

Course

- Το περιεχόμενο του μαθήματος ήταν κατανοητό.
- Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι.
- Το μάθημα ανταποκρίθηκε στις προσδοκίες μου.
- Οι ασκήσεις/παραδείγματα βοήθησαν στην κατανόηση της ύλης.
- Το υλικό του μαθήματος ήταν επαρκές και ποιοτικό.
- Η αξιολόγηση του μαθήματος ήταν δίκαιη.
- Το μάθημα συνέβαλε στην ανάπτυξη δεξιοτήτων.
- Ο φόρτος εργασίας ήταν λογικός για το μάθημα.
- Οι διαλέξεις ήταν καλοσχεδιασμένες.
- Το μάθημα ενίσχυσε το ενδιαφέρον μου για το αντικείμενο.
- Το πρόγραμμα των μαθημάτων τηρήθηκε επακριβώς.
- Υπήρχε επαρκής επικοινωνία μέσω της ηλεκτρονικής πλατφόρμας.
- Οι επιπλέον δραστηριότητες συνέβαλαν στη μάθηση.
- Η εμπειρία μου από το μάθημα ήταν θετική.
- Θα πρότεινα το μάθημα αυτό σε άλλους φοιτητές.

Εικόνα 4.8: Δημιουργία Νέας Φόρμας 1 - Teacher

Οι εικόνες 4.8 και 4.9 παρουσιάζουν τη σελίδα Δημιουργία Νέας Φόρμας, στην οποία ο διδάσκων μπορεί να δημιουργήσει μια νέα φόρμα αξιολόγησης για κάποιο από τα μαθήματά του. Πρόκειται για ένα σημαντικό εργαλείο που επιτρέπει στον χρήστη να καθορίσει τις λεπτομέρειες της αξιολόγησης και να επιλέξει τις ερωτήσεις που θα συμπεριληφθούν.

Στο πάνω μέρος της φόρμας υπάρχουν πεδία εισαγωγής βασικών πληροφοριών:

- Τίτλος: Το όνομα της φόρμας αξιολόγησης που βοηθά στον εύκολο εντοπισμό της.
- Περιγραφή: Συνοπτική περιγραφή του σκοπού ή του περιεχομένου της φόρμας.
- Ημερομηνία Έναρξης: Η ημερομηνία από την οποία θα είναι διαθέσιμη η φόρμα για συμπλήρωση από τους φοιτητές.
- Ημερομηνία Λήξης: Η τελευταία ημέρα κατά την οποία οι φοιτητές θα μπορούν να υποβάλουν τις απαντήσεις τους.

Στο κάτω μέρος της φόρμας εμφανίζεται η ενότητα Ερωτήσεις (ανά Πεδίο). Εδώ, ο διδάσκων μπορεί να επιλέξει από μια προτεινόμενη δεξαμενή ερωτήσεων σχετικών με το μάθημα για να διαμορφώσει την αξιολόγηση σύμφωνα με τις ανάγκες του. Οι ερωτήσεις αφορούν το πεδίο Course και καλύπτουν διάφορες πτυχές της διδασκαλίας και του μαθήματος όπως την κατανόηση της ύλης, την επάρκεια του υλικού, τη σαφήνεια των στόχων, τη συμπεριφορά των διδασκόντων και την ενεργό συμμετοχή των φοιτητών. Ο διδάσκων μπορεί να επιλέξει όσες ερωτήσεις θεωρεί κατάλληλες για την αξιολόγηση του συγκεκριμένου μαθήματος και δημιουργεί έτσι μια προσαρμοσμένη φόρμα που ανταποκρίνεται στις πραγματικές ανάγκες του.

Η σελίδα αυτή συμβάλλει στην οργάνωση και τυποποίηση της διαδικασίας αξιολόγησης επιτρέποντας στους διδάσκοντες να επιλέξουν στοχευμένα ερωτήσεις και να ορίσουν συγκεκριμένο χρονικό πλαίσιο για τη διεξαγωγή της αξιολόγησης.

Teacher

- Ο διδάσκων ήταν καλά προετοιμασμένος.
- Ο διδάσκων ενθάρρυνε την ενεργή συμμετοχή.
- Ο διδάσκων απαντούσε με σαφήνεια στις ερωτήσεις.
- Ο διδάσκων ήταν διαθέσιμος για βοήθεια εκτός διαλέξεων.
- Ο διδάσκων ενίσχυε την κριτική σκέψη.
- Ο διδάσκων χρησιμοποιούσε κατάλληλα παραδείγματα.
- Η μεταδοτικότητα του διδάσκοντα ήταν υψηλή.
- Ο διδάσκων ενσωμάτωνε σύγχρονες τεχνολογίες στη διδασκαλία.

Facilities

- Οι αίθουσες διδασκαλίας ήταν κατάλληλες.
- Ο εξοπλισμός στις αίθουσες ήταν επαρκής.
- Οι υπολογιστικές υποδομές ήταν λειτουργικές.
- Η πρόσβαση σε εκπαιδευτικό υλικό ήταν εύκολη.
- Το εργαστηριακό περιβάλλον ήταν κατάλληλο.
- Η καθαριότητα των εγκαταστάσεων ήταν επαρκής.
- Οι χώροι μελέτης ήταν προσβάσιμοι και επαρκείς.
- Οι αίθουσες είχαν επαρκή φωτισμό και αερισμό.
- Οι υποδομές κάλυπταν τις ανάγκες ατόμων με αναπηρία.

Student

- Επέδειξα συνέπεια στην παρακολούθηση του μαθήματος.
- Συμμετείχα ενεργά στις δραστηριότητες του μαθήματος.
- Επιδίωξα να κατανοήσω σε βάθος την ύλη.
- Ανέλαβα υπεύθυνα τις υποχρεώσεις του μαθήματος.
- Αξιοποίησα τους διαθέσιμους πόρους του μαθήματος.
- Συνεργάστηκα αποτελεσματικά με άλλους φοιτητές.
- Επέδειξα προσωπική βελτίωση κατά τη διάρκεια του μαθήματος.
- Αξιολόγησα αντικειμενικά τις δραστηριότητες του μαθήματος.

Δημιουργία

Ακύρωση

Εικόνα 4.9: Δημιουργία Νέας Φόρμας 2 - Teacher

Επεξεργασία Φόρμας

Τίτλος
Μάθημα Ρομποτική

Περιγραφή
Μάθημα Ρομποτική

Ημερομηνία Έναρξης
14-05-2025

Ημερομηνία Λήξης
15-05-2025

Η φόρμα έχει οριστικοποιηθεί. Μπορείτε να αλλάξετε μόνο Τίτλο, Περιγραφή και Ημερομηνίες.

Αποθήκευση Ακύρωση

Εικόνα 4.10: Επεξεργασία Φόρμας - Οριστικοποιημένη - Teacher

Η εικόνα 4.10 παρουσιάζει τη σελίδα Επεξεργασία Φόρμας, μέσω της οποίας ο διδάσκων μπορεί να τροποποιήσει ορισμένα στοιχεία μιας φόρμας αξιολόγησης. Στην προκειμένη περίπτωση η φόρμα έχει ήδη οριστικοποιηθεί κάτι που δηλώνεται και με το χαρακτηριστικό μπλε ενημερωτικό πλαίσιο στο κάτω μέρος, το οποίο αναφέρει:

«Η φόρμα έχει οριστικοποιηθεί. Μπορείτε να αλλάξετε μόνο Τίτλο, Περιγραφή και Ημερομηνίες.»

Στη φόρμα εμφανίζονται προσυμπληρωμένα τα πεδία με τις υπάρχουσες πληροφορίες:

- Τίτλος: Το όνομα της φόρμας αξιολόγησης, το οποίο μπορεί να τροποποιηθεί.
- Περιγραφή: Σύντομη περιγραφή της φόρμας, επίσης τροποποιήσιμη.
- Ημερομηνία Έναρξης: Μπορεί να αλλάξει, καθορίζοντας από ποια ημέρα θα είναι ενεργή η φόρμα.
- Ημερομηνία Λήξης: Ορίζεται η ημερομηνία λήξης της διαθεσιμότητας της φόρμας.

Οι ερωτήσεις της φόρμας δεν εμφανίζονται εδώ και δεν είναι επεξεργάσιμες καθώς η φόρμα έχει οριστικοποιηθεί και δεν επιτρέπονται αλλαγές στο περιεχόμενο των ερωτήσεων μετά την οριστικοποίηση.

Στο κάτω μέρος, υπάρχουν δύο επιλογές ενεργειών:

- Αποθήκευση (μπλε κουμπί): Εφαρμόζει τις αλλαγές στον τίτλο, την περιγραφή και τις ημερομηνίες.
- Ακύρωση (γκρι κουμπί): Επιστρέφει στη λίστα των φορμών χωρίς να αποθηκευτούν οι αλλαγές.

Αυτή η σελίδα δίνει τη δυνατότητα στον διδάσκοντα να κάνει βασικές διορθώσεις ή να παρατείνει την περίοδο υποβολής της αξιολόγησης, διατηρώντας παράλληλα τη σταθερότητα του περιεχομένου της φόρμας.

Η εικόνα 4.11 παρουσιάζει τη σελίδα Επεξεργασία Φόρμας, στην οποία ο διδάσκων μπορεί να τροποποιήσει μια φόρμα αξιολόγησης πριν αυτή οριστικοποιηθεί. Η φόρμα αυτή αφορά το μάθημα Ηλεκτρονικά Κυκλώματα, όπως δηλώνεται στον τίτλο και στην περιγραφή.

Ο διδάσκων έχει τη δυνατότητα να επεξεργαστεί τα παρακάτω πεδία:

- Τίτλος: Το όνομα της φόρμας, το οποίο μπορεί να αλλάξει ελεύθερα.
- Περιγραφή: Περιγράφει το περιεχόμενο της αξιολόγησης ή του μαθήματος.
- Ημερομηνία Έναρξης και Ημερομηνία Λήξης: Ορίζουν το χρονικό διάστημα κατά το οποίο η φόρμα θα είναι διαθέσιμη προς συμπλήρωση από τους φοιτητές.

Στη συνέχεια, εμφανίζονται οι διαθέσιμες ερωτήσεις χωρισμένες σε πεδία, με δυνατότητα επιλογής μέσω checkbox:

- Course (Μάθημα): Περιέχει ερωτήσεις σχετικές με το περιεχόμενο, την οργάνωση και την ποιότητα του μαθήματος. Ο διδάσκων μπορεί να επιλέξει ποιες από αυτές θέλει να συμπεριλάβει στην αξιολόγηση.
- Teacher (Διδάσκων): Περιλαμβάνει ερωτήσεις σχετικές με τη συμπεριφορά, την προετοιμασία και τη διδασκαλία του καθηγητή. Ο διδάσκων μπορεί επίσης να επιλέξει ποιες από αυτές θα συμπεριληφθούν.

Η δυνατότητα επιλογής ερωτήσεων ανά πεδίο επιτρέπει την πλήρη προσαρμογή της φόρμας στις ανάγκες του κάθε μαθήματος. Ο διδάσκων μπορεί να δημιουργήσει στοχευμένες αξιολογήσεις, δίνοντας έμφαση στα σημεία που θεωρεί σημαντικά για τη βελτίωση της διδασκαλίας και της μαθησιακής διαδικασίας.

Μετά την ολοκλήρωση της επεξεργασίας, ο διδάσκων μπορεί να αποθηκεύσει τις αλλαγές ή να επιστρέψει πίσω ακυρώνοντας την επεξεργασία.

Επεξεργασία Φόρμας

Τίτλος

Μάθημα Ηλεκτρονικά Κυκλώματα

Περιγραφή

Αφορά το Μάθημα Ηλεκτρονικά Κυκλώματα

Ημερομηνία Έναρξης

02-05-2025

Ημερομηνία Λήξης

25-05-2025

Ερωτήσεις (ανά Πεδίο)

Course

- Το περιεχόμενο του μαθήματος ήταν κατανοητό.
- Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι.
- Το μάθημα ανταποκρίθηκε στις προσδοκίες μου.
- Οι ασκήσεις/παραδείγματα βοήθησαν στην κατανόηση της ύλης.
- Το υλικό του μαθήματος ήταν επαρκές και ποιοτικό.
- Η αξιολόγηση του μαθήματος ήταν δίκαιη.
- Το μάθημα συνέβαλε στην ανάπτυξη δεξιοτήτων.
- Ο φόρτος εργασίας ήταν λογικός για το μάθημα.
- Οι διαλέξεις ήταν καλοσχεδιασμένες.
- Το μάθημα ενίσχυσε το ενδιαφέρον μου για το αντικείμενο.
- Το πρόγραμμα των μαθημάτων τηρήθηκε επακριβώς.
- Υπήρχε επαρκής επικοινωνία μέσω της ηλεκτρονικής πλατφόρμας.
- Οι επιπλέον δραστηριότητες συνέβαλαν στη μάθηση.
- Η εμπειρία μου από το μάθημα ήταν θετική.
- Θα πρότεινα το μάθημα αυτό σε άλλους φοιτητές.

Teacher

- Ο διδάσκων ήταν καλά προετοιμασμένος.
- Ο διδάσκων ενθάρρυνε την ενεργή συμμετοχή.
- Ο διδάσκων απαντούσε με σαφήνεια στις ερωτήσεις.
- Ο διδάσκων ήταν διαθέσιμος για βοήθεια εκτός διαλέξεων.
- Ο διδάσκων ενίσχυε την κριτική σκέψη.
- Ο διδάσκων χρησιμοποιούσε κατάλληλα παραδείγματα.

Εικόνα 4.11: Επεξεργασία Φόρμας – Μη Οριστικοποιημένη - Teacher

Κωδικοί Αξιολόγησης

+ 50 Νέοι Κωδικοί

1-2-Y8QDR Διαθέσιμο	1-2-H5CSN Χρησιμοποιήθηκε 14-05-2025	1-2-T0IGT Διαθέσιμο	1-2-Q9RUB Χρησιμοποιήθηκε 14-05-2025
1-2-HMV41 Χρησιμοποιήθηκε 14-05-2025	1-2-K8A7R Διαθέσιμο	1-2-J2R60 Διαθέσιμο	1-2-O6DUY Διαθέσιμο
1-2-XN3V7 Διαθέσιμο	1-2-2G9X3 Διαθέσιμο	1-2-B745V Διαθέσιμο	1-2-LS529 Διαθέσιμο
1-2-QZTY3 Διαθέσιμο	1-2-93X0D Διαθέσιμο	1-2-5MQNW Διαθέσιμο	1-2-DW0WJ Διαθέσιμο
1-2-4P4FL Διαθέσιμο	1-2-LUDDQ Διαθέσιμο	1-2-MY39Y Διαθέσιμο	1-2-D1HOL Διαθέσιμο

Εικόνα 4.12: Σελίδα: Κωδικοί Αξιολόγησης - Teacher

Η εικόνα 4.12 παρουσιάζει τη σελίδα Κωδικοί Αξιολόγησης, όπου ο διδάσκων μπορεί να διαχειριστεί τους μοναδικούς κωδικούς που χρησιμοποιούνται από τους φοιτητές για την υποβολή των αξιολογήσεων. Η σελίδα παρέχει μια καθαρή και οργανωμένη προβολή όλων των κωδικών που έχουν εκδοθεί για συγκεκριμένη φόρμα αξιολόγησης.

Στην κορυφή της σελίδας υπάρχει το μπλε κουμπί +50 Νέοι Κωδικοί, μέσω του οποίου ο διδάσκων μπορεί να παράγει επιπλέον κωδικούς σε παρτίδες των 50, για να καλύψει τις ανάγκες μεγαλύτερου αριθμού φοιτητών.

Κάθε κωδικός εμφανίζεται μέσα σε ξεχωριστό πλαίσιο, μαζί με την κατάσταση του:

- Οι κωδικοί που αναγράφουν Διαθέσιμο με πράσινο χρώμα, δεν έχουν ακόμα χρησιμοποιηθεί και είναι έτοιμοι για διανομή στους φοιτητές.
- Οι κωδικοί που συνοδεύονται από την ένδειξη Χρησιμοποιήθηκε και την αντίστοιχη ημερομηνία (π.χ. 14-05-2025) έχουν ήδη χρησιμοποιηθεί από κάποιο φοιτητή για την υποβολή αξιολόγησης.

Η μορφή των κωδικών ακολουθεί το πρότυπο 1-2-XXXXX, όπου τα πρώτα δύο τμήματα μπορεί να σχετίζονται με το ID της φόρμας ή του χρήστη, και το τελευταίο τμήμα είναι ένας τυχαίος μοναδικός συνδυασμός χαρακτήρων.

Η σελίδα αυτή δίνει στον διδάσκοντα άμεση εποπτεία σχετικά με τη χρήση των κωδικών και του επιτρέπει να διαχειρίζεται αποτελεσματικά τη διαδικασία αξιολόγησης, εξασφαλίζοντας ότι κάθε φοιτητής μπορεί να συμμετάσχει μόνο μία φορά με έναν μοναδικό κωδικό.



Εικόνα 4.13: Σελίδα: Στατιστικά Αξιολόγησης - Teacher

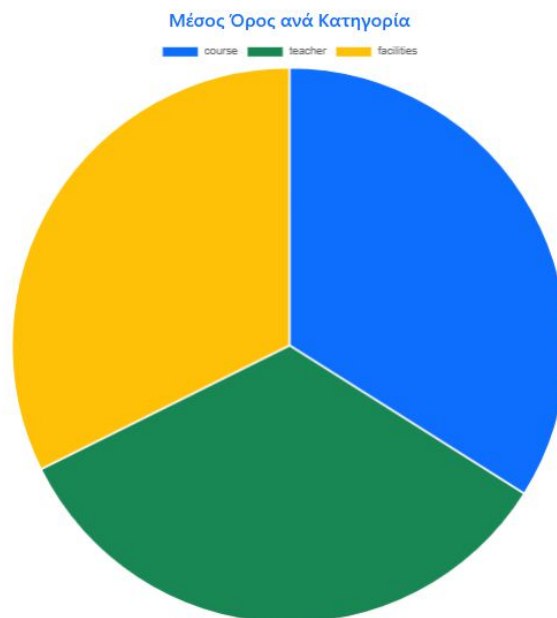
Η εικόνα 4.13 παρουσιάζει τη σελίδα Στατιστικά Αξιολόγησης για το μάθημα Ρομποτική, όπου εμφανίζονται τα συγκεντρωτικά αποτελέσματα των αξιολογήσεων που υποβλήθηκαν από τους φοιτητές. Η αξιολόγηση πραγματοποιήθηκε από τις 14-05-2025 έως τις 15-05-2025, με συμμετοχή 3 φοιτητών.

Στο πάνω μέρος της σελίδας αναγράφονται οι βασικές πληροφορίες για το μάθημα και το χρονικό διάστημα της αξιολόγησης, ενώ υπάρχει και κουμπί Εκτύπωση Στατιστικών, που επιτρέπει την εξαγωγή των αποτελεσμάτων σε μορφή κατάλληλη για εκτύπωση ή αρχειοθέτηση.

Το κύριο μέρος της σελίδας περιλαμβάνει ένα γράφημα τύπου οριζόντιας ράβδου (bar chart), όπου παρουσιάζεται ο Μέσος Όρος ανά Ερώτηση. Για κάθε ερώτηση που περιλαμβάνόταν στη φόρμα αξιολόγησης, εμφανίζεται ο μέσος όρος της βαθμολογίας που έδωσαν οι φοιτητές σε κλίμακα από 1 έως 5.

Οι ερωτήσεις καλύπτουν βασικούς τομείς όπως η κατανόηση του περιεχομένου του μαθήματος, η σαφήνεια των στόχων, η επάρκεια του εκπαιδευτικού υλικού, η συμβολή του μαθήματος στην ανάπτυξη δεξιοτήτων, η συμπεριφορά και η προετοιμασία του διδάσκοντα, καθώς και η καταλληλότητα των υποδομών.

Το γράφημα προσφέρει άμεση οπτική ανατροφοδότηση στον διδάσκοντα σχετικά με τα δυνατά σημεία και τις περιοχές που χρήζουν βελτίωσης, βάσει της άποψης των φοιτητών.



Μέσος Όρος ανά Κατηγορία

Κατηγορία	Μέσος Όρος
Course	4.20
Teacher	4.17
Facilities	4.00

Εικόνα 4.14: Οπτικοποίηση του Μέσου Όρου ανά Κατηγορία για τις αξιολογήσεις ενός μαθήματος - Teacher

Η παραπάνω εικόνα παρουσιάζει την οπτικοποίηση του Μέσου Όρου ανά Κατηγορία για τις αξιολογήσεις ενός μαθήματος, χρησιμοποιώντας κυκλικό διάγραμμα (pie chart) και συγκεντρωτικό πίνακα.

Το κυκλικό διάγραμμα χωρίζεται σε τρία έγχρωμα τμήματα, το καθένα από τα οποία αντιπροσωπεύει μία από τις βασικές κατηγορίες αξιολόγησης:

- Μπλε (Course): Αναφέρεται στο περιεχόμενο και την ποιότητα του μαθήματος.
- Πράσινο (Teacher): Αξιολογεί τον διδάσκοντα, την προετοιμασία του και τη διδακτική του ικανότητα.
- Κίτρινο (Facilities): Σχετίζεται με τις υποδομές και τις συνθήκες των αιθουσών διδασκαλίας.

Κάτω από το διάγραμμα εμφανίζεται πίνακας με τα αριθμητικά αποτελέσματα:

Κατηγορία Μέσος Όρος

Course 4.20

Teacher 4.17

Facilities 4.00

Από τον πίνακα φαίνεται ότι οι φοιτητές έδωσαν την υψηλότερη βαθμολογία στην κατηγορία Course, γεγονός που δείχνει ότι έμειναν περισσότερο ικανοποιημένοι από το περιεχόμενο και την οργάνωση του μαθήματος. Ακολουθεί η κατηγορία Teacher, όπου επίσης καταγράφεται υψηλή ικανοποίηση. Η κατηγορία Facilities παρουσιάζει το χαμηλότερο μέσο όρο, αν και παραμένει σε υψηλά επίπεδα, γεγονός που υποδηλώνει ότι οι υποδομές ήταν επαρκείς αλλά θα μπορούσαν να βελτιωθούν.

Μέσος Όρος ανά Ερώτηση

Ερώτηση	Κατηγορία	Μέσος Όρος
Το περιεχόμενο του μαθήματος ήταν κατανοητό.	Course	4.00
Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι.	Course	4.67
Το μάθημα ανταποκρίθηκε στις προσδοκίες μου.	Course	4.00
Οι ασκήσεις/παραδείγματα βοήθησαν στην κατανόηση της ύλης.	Course	4.33
Το υλικό του μαθήματος ήταν επαρκές και ποιοτικό.	Course	4.33
Η αξιολόγηση του μαθήματος ήταν δίκαιη.	Course	4.33
Το μάθημα συνέβαλε στην ανάπτυξη δεξιοτήτων.	Course	4.00
Ο φόρτος εργασίας ήταν λογικός για το μάθημα.	Course	4.33
Οι διαλέξεις ήταν καλοσχεδιασμένες.	Course	3.67
Το μάθημα ενίσχυσε το ενδιαφέρον μου για το αντικείμενο.	Course	4.33
Ο διδάσκων ήταν καλά προετοιμασμένος.	Teacher	4.33
Ο διδάσκων ενθάρρυνε την ενεργή συμμετοχή.	Teacher	4.00
Οι αίθουσες διδασκαλίας ήταν κατάλληλες.	Facilities	4.33
Ο εξοπλισμός στις αίθουσες ήταν επαρκής.	Facilities	3.67

Σχόλια Φοιτητών

Καλό
Ωραία

Εικόνα 4.15: Αναλυτικά Στατιστικά Αποτελέσματα Αξιολόγησης

Η εικόνα 4.15 παρουσιάζει τα Αναλυτικά Στατιστικά Αποτελέσματα Αξιολόγησης, όπου εμφανίζεται ο μέσος όρος βαθμολογίας για κάθε ερώτηση ξεχωριστά, καθώς και τα σχόλια των φοιτητών.

Στην ενότητα Μέσος Όρος ανά Ερώτηση, παρατίθεται ένας πίνακας με τις εξής στήλες:

- Ερώτηση: Το κείμενο της ερώτησης που απάντησαν οι φοιτητές.
- Κατηγορία: Η κατηγορία στην οποία ανήκει η ερώτηση (Course, Teacher, Facilities).
- Μέσος Όρος: Η μέση βαθμολογία που συγκέντρωσε η ερώτηση, σε κλίμακα από 1 έως 5.

Από τον πίνακα παρατηρείται ότι η υψηλότερη βαθμολογία (4.67) συγκεντρώνει η ερώτηση «Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι», γεγονός που δείχνει ότι οι φοιτητές βρήκαν τη στοχοθέτηση του μαθήματος εξαιρετικά ξεκάθαρη. Αντίθετα, η χαμηλότερη βαθμολογία (3.67) αφορά την ερώτηση «Οι διαλέξεις ήταν καλοσχεδιασμένες.» και «Ο εξοπλισμός στις αίθουσες ήταν επαρκής.», κάτι που υποδεικνύει περιθώρια βελτίωσης τόσο στον τρόπο παρουσίασης των διαλέξεων όσο και στις υλικοτεχνικές υποδομές.

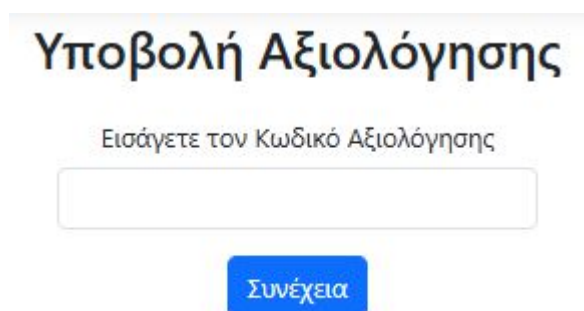
Στο κάτω μέρος της εικόνας εμφανίζεται η ενότητα Σχόλια Φοιτητών, στην οποία καταγράφονται τα ελεύθερα σχόλια που άφησαν οι συμμετέχοντες στην αξιολόγηση.

Η συγκεκριμένη ανάλυση προσφέρει στον διδάσκοντα πολύτιμα στοιχεία για την αξιολόγηση της αποτελεσματικότητας της διδασκαλίας του, τη βελτίωση του μαθήματος και των συνθηκών διδασκαλίας. Μέσω αυτής της διαδικασίας ενισχύεται η ποιότητα της εκπαιδευτικής εμπειρίας τόσο για τον διδάσκοντα όσο και για τους φοιτητές.

4.2.2 Ρόλος: Student – Φοιτητής

Ο φοιτητής πρέπει να μπει στο

<http://127.0.0.1:5000/evaluation/submit>



Υποβολή Αξιολόγησης

Εισάγετε τον Κωδικό Αξιολόγησης

Συνέχεια

Εικόνα 4.16: Υποβολή Αξιολόγησης

Η σελίδα υποβολής αξιολόγησης αποτελεί το αρχικό βήμα της διαδικασίας, όπου ο φοιτητής καλείται να εισάγει τον μοναδικό κωδικό αξιολόγησης που του έχει δοθεί. Μέσω αυτής της διαδικασίας διασφαλίζεται η ανωνυμία και η μοναδικότητα της συμμετοχής. Αν ο κωδικός είναι έγκυρος και αχρησιμοποίητος, ο φοιτητής οδηγείται στη φόρμα αξιολόγησης.

Υποβολή Αξιολόγησης

Εισάγετε τον Κωδικό Αξιολόγησης

Συνέχεια

Στοιχεία Φόρμας Αξιολόγησης
Μάθημα: Μάθημα Ρομποτική
Ημερομηνίες: Από 14-05-2025 έως 15-05-2025

Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων
Διεθνές Πανεπιστήμιο της Ελλάδος

Συμπλήρωση Αξιολόγησης

Εικόνα 4.17: Στοιχεία Φόρμας Αξιολόγησης και κουμπί για Συμπλήρωση Αξιολόγησης

Στην Εικόνα 4.17 εμφανίζονται πληροφορίες σχετικά με τη φόρμα αξιολόγησης, όπως ο τίτλος του μαθήματος, η περιγραφή και οι ημερομηνίες έναρξης και λήξης της αξιολόγησης. Ο φοιτητής έχει τη δυνατότητα να ελέγξει τις λεπτομέρειες πριν προχωρήσει στη συμπλήρωση. Παρέχεται επίσης ένα κουμπί για την έναρξη της αξιολόγησης.

Φόρμα Αξιολόγησης (Κωδικός: 1-2-2G9X3)

Επεξήγηση Βαθμολογίας:

1 - Διαφωνώ απόλυτα | 2 - Διαφωνώ | 3 - Ούτε συμφωνώ/ούτε διαφωνώ | 4 - Συμφωνώ | 5 - Συμφωνώ απόλυτα

Course

Το περιεχόμενο του μαθήματος ήταν κατανοητό.

1 2 3 4 5

Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι.

1 2 3 4 5

Το μάθημα ανταποκρίθηκε στις προσδοκίες μου.

1 2 3 4 5

Οι ασκήσεις/παραδείγματα βοήθησαν στην κατανόηση της ύλης.

1 2 3 4 5

Το υλικό του μαθήματος ήταν επαρκές και ποιοτικό.

1 2 3 4 5

Η αξιολόγηση του μαθήματος ήταν δίκαιη.

1 2 3 4 5

Το μάθημα συνέβαλε στην ανάπτυξη δεξιοτήτων.

1 2 3 4 5

Ο φόρτος εργασίας ήταν λογικός για το μάθημα.

1 2 3 4 5

Οι διαλέξεις ήταν καλοσχεδιασμένες.

1 2 3 4 5

Το μάθημα ενίσχυσε το ενδιαφέρον μου για το αντικείμενο.

1 2 3 4 5

Εικόνα 4.18: Φόρμα Αξιολόγησης

Στην εικόνα 4.18 έχουμε την κύρια φόρμα αξιολόγησης όπου ο φοιτητής καλείται να απαντήσει στις ερωτήσεις που έχουν επιλεγεί από τον διδάσκοντα. Οι ερωτήσεις είναι οργανωμένες ανά κατηγορία και ο φοιτητής μπορεί να αξιολογήσει κάθε ερώτηση με βαθμολογία από 1 έως 5 αστεράκια. Παρέχεται επίσης πεδίο σχολίων για προαιρετικές παρατηρήσεις.

Ο φόρτος εργασίας ήταν λογικός για το μάθημα.

1 2 3 4 5

Οι διαλέξεις ήταν καλοσχεδιασμένες.

1 2 3 4 5

Το μάθημα ενίσχυσε το ενδιαφέρον μου για το αντικείμενο.

1 2 3 4 5

Teacher

Ο διδάσκων ήταν καλά προετοιμασμένος.

1 2 3 4 5

Ο διδάσκων ενθάρρυνε την ενεργή συμμετοχή.

1 2 3 4 5

Facilities

Οι αίθουσες διδασκαλίας ήταν κατάλληλες.

1 2 3 4 5

Ο εξοπλισμός στις αίθουσες ήταν επαρκής.

1 2 3 4 5

Σχόλια (προαιρετικά)

Γράψτε τα σχολιά σας εδώ...

Υποβολή Αξιολόγησης

Εικόνα 4.19: Φόρμα Αξιολόγησης – συνέχεια

Η συνέχεια της φόρμας αξιολόγησης, όπου παρουσιάζονται επιπλέον ερωτήσεις ή κατηγορίες, ανάλογα με τη δομή της φόρμας. Ο φοιτητής μπορεί να ολοκληρώσει την αξιολόγηση και να την υποβάλει μόλις συμπληρώσει όλες τις απαντήσεις.

Φόρμα Αξιολόγησης (Κωδικός: 1-2-2G9X3)

Επεξήγηση Βαθμολογίας:

1 - Διαφωνώ απόλυτα | 2 - Διαφωνώ | 3 - Ούτε συμφωνώ/ούτε διαφωνώ | 4 - Συμφωνώ | 5 - Συμφωνώ απόλυτα

Course

Το περιεχόμενο του μαθήματος ήταν κατανοητό.

1 2 3 4 5

Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι.

1 2 3 4 5

Το μάθημα ανταποκρίθηκε στις προσδοκίες μου.

1 2 3 4 5

Οι ασκήσεις/παραδείγματα βοήθησαν στην κατανόηση της ύλης.

1 2 3 4 5

Εικόνα 4.20: Φόρμα Αξιολόγησης – Ειδοποίηση σε περίπτωση μη συμπλήρωσης-απάντησης ερώτησης

Ο κωδικός έχει ήδη χρησιμοποιηθεί.



Υποβολή Αξιολόγησης

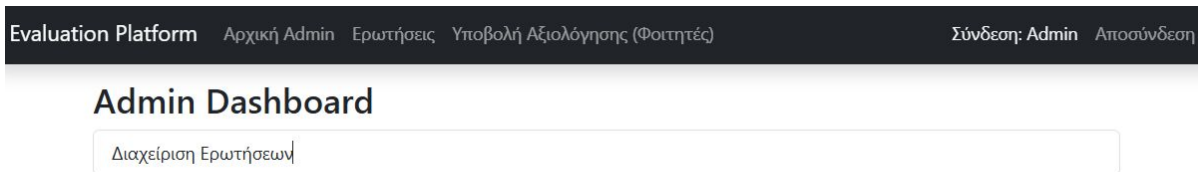
Εισάγετε τον Κωδικό Αξιολόγησης

Συνέχεια

Εικόνα 4.21: Ειδοποίηση σε περίπτωση εισαγωγής χρησιμοποιούμενου κωδικού κατά την υποβολή αξιολόγησης

Αν ο φοιτητής επιχειρήσει να υποβάλει την αξιολόγηση χωρίς να έχει απαντήσει σε όλες τις απαιτούμενες ερωτήσεις, εμφανίζεται προειδοποιητικό μήνυμα. Οι ερωτήσεις που δεν έχουν απαντηθεί επισημαίνονται οπτικά (συνήθως με κόκκινο χρώμα) ώστε να διευκολύνεται η διόρθωση.

4.2.3 Ρόλος: Admin – Διαχειριστής



Εικόνα 4.22: Admin Dashboard

Στην Εικόνα 4.22 παρουσιάζεται η κεντρική σελίδα διαχείρισης για τον διαχειριστή του συστήματος. Παρέχει γρήγορη πρόσβαση στη διαχείριση των ερωτήσεων αξιολόγησης μέσω ενός απλού και καθαρού μενού. Ο διαχειριστής μπορεί να μεταβεί άμεσα στη διαχείριση ερωτήσεων πατώντας στον σχετικό σύνδεσμο ή χρησιμοποιώντας την αναζήτηση.

Στην Εικόνα 4.23 παρουσιάζεται η σελίδα προβολής όλων των διαθέσιμων ερωτήσεων του συστήματος. Οι ερωτήσεις εμφανίζονται σε πίνακα, όπου φαίνεται το ID, το κείμενο της ερώτησης, ο τύπος (π.χ., course, teacher) και οι διαθέσιμες ενέργειες. Ο διαχειριστής μπορεί να επεξεργαστεί ή να διαγράψει οποιαδήποτε ερώτηση, εφόσον δεν χρησιμοποιείται ήδη σε φόρμα αξιολόγησης. Παρέχεται και κουμπί για την προσθήκη νέας ερώτησης.

Λίστα Ερωτήσεων

Νέα Ερώτηση

ID	Κείμενο	Τύπος	Ενέργειες
1	Το περιεχόμενο του μαθήματος ήταν κατανοητό.	course	Επεξεργασία Διαγραφή
2	Οι στόχοι του μαθήματος ήταν σαφώς καθορισμένοι.	course	Επεξεργασία Διαγραφή
3	Το μάθημα ανταποκρίθηκε στις προσδοκίες μου.	course	Επεξεργασία Διαγραφή
4	Οι ασκήσεις/παραδείγματα βοήθησαν στην κατανόηση της ύλης.	course	Επεξεργασία Διαγραφή
5	Το υλικό του μαθήματος ήταν επαρκές και ποιοτικό.	course	Επεξεργασία Διαγραφή
6	Η αξιολόγηση του μαθήματος ήταν δίκαιη.	course	Επεξεργασία Διαγραφή
7	Το μάθημα συνέβαλε στην ανάπτυξη δεξιοτήτων.	course	Επεξεργασία Διαγραφή
8	Ο φόρτος εργασίας ήταν λογικός για το μάθημα.	course	Επεξεργασία Διαγραφή
9	Οι διαλέξεις ήταν καλοσχεδιασμένες.	course	Επεξεργασία Διαγραφή
10	Το μάθημα ενίσχυσε το ενδιαφέρον μου για το αντικείμενο.	course	Επεξεργασία Διαγραφή
11	Ο διδάσκων ήταν καλά προετοιμασμένος.	teacher	Επεξεργασία Διαγραφή
12	Ο διδάσκων ενθάρρυνε την ενεργή συμμετοχή.	teacher	Επεξεργασία Διαγραφή
13	Ο διδάσκων απαντούσε με σαφήνεια στις ερωτήσεις.	teacher	Επεξεργασία Διαγραφή

Εικόνα 4.23: Λίστα Ερωτήσεων

Στην εικόνα 4.22 φαίνεται η φόρμα εισαγωγής νέας ερώτησης στο σύστημα. Ο διαχειριστής εισάγει το κείμενο της ερώτησης και επιλέγει τον τύπο της (course, teacher, facilities). Μετά την ολοκλήρωση, η ερώτηση αποθηκεύεται στη βάση και γίνεται διαθέσιμη για επιλογή κατά τη δημιουργία φορμών αξιολόγησης.

Νέα Ερώτηση

Κείμενο Ερώτησης

Τύπος

[Αποθήκευση](#) [Ακύρωση](#)

Επεξεργασία Ερώτησης

Κείμενο Ερώτησης

Τύπος

[Αποθήκευση](#) [Ακύρωση](#)

Εικόνα 4.24: Νέα Ερώτηση- Επεξεργασία Ερώτησης

Ο διαχειριστής μπορεί να τροποποιήσει το κείμενο της ερώτησης και να αλλάξει τον τύπο της. Αφού πραγματοποιήσει τις επιθυμητές αλλαγές, μπορεί να αποθηκεύσει τις τροποποιήσεις ή να ακυρώσει τη διαδικασία.

4.3 Περιγραφή συστήματος μέσω κώδικα

Θα περιγράψουμε τα βασικά κομμάτια κώδικα από την μεριά του controller και όχι της html.

Τον κώδικα μπορείτε να τον βρείτε στο Παράρτημα της εργασίας.

```
from flask import Flask, render_template, request, redirect, url_for, session, flash
from flask_mysql import MySQL
from datetime import datetime
from config import Config
import random, string
from functools import wraps

app = Flask(__name__)
app.config.from_object(Config)
mysql = MySQL(app)
```

Αυτό το τμήμα αφορά την αρχικοποίηση της εφαρμογής Flask και την προετοιμασία των βασικών βιβλιοθηκών που χρησιμοποιούνται στο έργο.

1. Εισαγωγή Βιβλιοθηκών

- flask: Το βασικό framework που χρησιμοποιείται για την υλοποίηση της web εφαρμογής.
- flask_mysql: Χρησιμοποιείται για την επικοινωνία της εφαρμογής με βάση δεδομένων MySQL.
- datetime: Χρησιμοποιείται για τον χειρισμό και τη μορφοποίηση ημερομηνιών.
- config.Config: Εισάγεται το αρχείο παραμέτρων Config, όπου είναι αποθηκευμένες οι ρυθμίσεις της εφαρμογής (π.χ. σύνδεση στη βάση δεδομένων, κλειδιά ασφαλείας).
- random, string: Χρησιμοποιούνται για τη δημιουργία τυχαίων κωδικών αξιολόγησης.
- functools.wraps: Βοηθάει στη δημιουργία διακοσμητών (decorators), όπως για τον έλεγχο πρόσβασης χρηστών.

2. Αρχικοποίηση Flask App

- `app = Flask(__name__)`: Δημιουργεί την εφαρμογή Flask.
- `app.config.from_object(Config)`: Εφαρμόζει τις ρυθμίσεις που περιέχονται στο αρχείο `config.py`.
- `mysql = MySQL(app)`: Αρχικοποιεί τη σύνδεση με τη βάση δεδομένων MySQL, ώστε να μπορούν να εκτελούνται queries μέσα από την εφαρμογή.

Συνάρτηση Login & Διαχείριση Συνεδρίας Χρηστών

```
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        cur = mysql.connection.cursor()
        cur.execute("SELECT * FROM users WHERE username=%s AND password=%s", (username, password))
        user = cur.fetchone()
        if user:
            session['user_id'] = user[0]
            session['username'] = user[1]
            session['role'] = user[5]
            return redirect(url_for(f'dashboard_{session['role']}'))
        flash('Λάθος στοιχεία.', 'danger')
    return render_template('login.html')
```

Η παραπάνω συνάρτηση υλοποιεί τη λειτουργία εισόδου χρηστών. Ελέγχει αν το αίτημα είναι POST και διαβάζει τα στοιχεία σύνδεσης. Εκτελεί ερώτημα στη βάση για να επιβεβαιώσει την εγκυρότητα των στοιχείων. Αν βρεθεί χρήστης, αποθηκεύει πληροφορίες στη συνεδρία (session) και ανακατευθύνει στο κατάλληλο dashboard ανάλογα με τον ρόλο (admin ή teacher). Αν τα στοιχεία είναι λάθος, εμφανίζει μήνυμα λάθους.

Διαχείριση Φορμών Αξιολόγησης (Teacher Dashboard)

```
@app.route('/dashboard/teacher')
@login_required('teacher')
def dashboard_teacher():
    cur = mysql.connection.cursor()
    user_id = session['user_id']
```

```

cur.execute("SELECT COUNT(*) FROM evaluations WHERE created_by=%s", (user_id,))
forms_count = cur.fetchone()[0]

cur.execute("""
    SELECT COUNT(*) FROM evaluation_codes ec
    JOIN evaluations e ON ec.evaluation_id = e.id
    WHERE e.created_by=%s
""", (user_id,))
codes_count = cur.fetchone()[0]

cur.execute("""
    SELECT COUNT(DISTINCT ec.id)
    FROM evaluation_codes ec
    JOIN evaluations e ON ec.evaluation_id = e.id
    WHERE e.created_by=%s AND ec.used=1
""", (user_id,))
answered_forms_count = cur.fetchone()[0]

return render_template('dashboard_teacher.html',
    forms_count=forms_count,
    codes_count=codes_count,
    answered_forms_count=answered_forms_count)

```

Αυτό το τμήμα κώδικα υπολογίζει και εμφανίζει στατιστικά στοιχεία για τον διδάσκοντα στο dashboard. Παρέχει πληροφορίες για τον αριθμό των φορμών αξιολόγησης που έχει δημιουργήσει, τους παραχθέντες κωδικούς και τις απαντημένες φόρμες. Τα δεδομένα εμφανίζονται δυναμικά στην αρχική σελίδα του διδάσκοντα.

Υποβολή Αξιολόγησης από Φοιτητή

```

@app.route('/evaluation/submit/form/<int:code_id>', methods=['GET', 'POST'])
def evaluation_form_submit(code_id):
    cur = mysql.connection.cursor()

    cur.execute("SELECT code, evaluation_id, used FROM evaluation_codes WHERE id=%s", (code_id,))
    code_data = cur.fetchone()
    if not code_data or code_data[2] == 1:
        flash('Ο κωδικός δεν είναι έγκυρος ή έχει χρησιμοποιηθεί.', 'danger')
        return redirect(url_for('evaluation_submit_code'))

```

```

evaluation_id = code_data[1]

if request.method == 'POST':
    answers = request.form
    comments = answers.get('comments', "").strip()

    for question_id, rating in answers.items():
        if question_id.startswith('q_'):
            question_id_clean = question_id.replace('q_', '')
            cur.execute("""
                INSERT INTO answers (evaluation_code_id, question_id, rating)
                VALUES (%s, %s, %s)
            """, (code_id, question_id_clean, rating))

    if comments:
        cur.execute("UPDATE evaluation_codes SET comments=%s WHERE id=%s", (comments, code_id))

    cur.execute("UPDATE evaluation_codes SET used=1, used_at=NOW() WHERE id=%s", (code_id,))
    mysql.connection.commit()
    flash('Η αξιολόγηση υποβλήθηκε επιτυχώς. Ευχαριστούμε!', 'success')
    return redirect(url_for('evaluation_submit_code'))

cur.execute("""
    SELECT q.id, q.text, q.type
    FROM evaluation_questions eq
    JOIN questions q ON eq.question_id = q.id
    WHERE eq.evaluation_id=%s
    ORDER BY q.type
""", (evaluation_id,))
questions = cur.fetchall()

grouped_questions = {'course': [], 'teacher': [], 'facilities': [], 'student': []}
for q in questions:
    grouped_questions[q[2]].append(q)

return render_template('evaluation_form_student.html',
    code=code_data[0],
    grouped_questions=grouped_questions)

```

Εδώ διαχειριζόμαστε την υποβολή απαντήσεων από τον φοιτητή. Ο φοιτητής χρησιμοποιεί μοναδικό κωδικό συμμετοχής, απαντά στις ερωτήσεις και υποβάλλει τη φόρμα. Οι απαντήσεις και τυχόν σχόλια αποθηκεύονται στη βάση και ο κωδικός μαρκάρεται ως χρησιμοποιημένος εξασφαλίζοντας ότι δεν μπορεί να χρησιμοποιηθεί ξανά.

Στατιστικά Αποτελεσμάτων (Statistics Route)

```
@app.route('/teacher/evaluation/stats/<int:evaluation_id>')
@login_required('teacher')
def evaluation_stats(evaluation_id):
    cur = mysql.connection.cursor()

    cur.execute("SELECT title, start_date, end_date FROM evaluations WHERE id=%s", (evaluation_id,))
    eval_info = cur.fetchone()

    cur.execute("""
        SELECT q.text, q.type, AVG(a.rating)
        FROM answers a
        JOIN questions q ON a.question_id = q.id
        JOIN evaluation_codes ec ON a.evaluation_code_id = ec.id
        WHERE ec.evaluation_id = %s
        GROUP BY q.id
        ORDER BY q.type
    """, (evaluation_id,))
    stats = cur.fetchall()

    cur.execute("""
        SELECT q.type, AVG(a.rating)
        FROM answers a
        JOIN questions q ON a.question_id = q.id
        JOIN evaluation_codes ec ON a.evaluation_code_id = ec.id
        WHERE ec.evaluation_id = %s
        GROUP BY q.type
    """, (evaluation_id,))
    category_averages = cur.fetchall()

    cur.execute("""
        SELECT COUNT(DISTINCT ec.id)
```

```

FROM evaluation_codes ec
JOIN answers a ON ec.id = a.evaluation_code_id
WHERE ec.evaluation_id = %s
""" , (evaluation_id,))
student_count = cur.fetchone()[0]

cur.execute("""
SELECT comments
FROM evaluation_codes
WHERE evaluation_id = %s AND comments IS NOT NULL AND comments != ""
""" , (evaluation_id,))
comments = [row[0] for row in cur.fetchall()]

return render_template('statistics.html',
    eval_info=eval_info,
    stats=stats,
    category_averages=category_averages,
    student_count=student_count,
    comments=comments)

```

Η παραπάνω συνάρτηση συλλέγει και παρουσιάζει στατιστικά δεδομένα για μια συγκεκριμένη φόρμα αξιολόγησης. Υπολογίζει μέσους όρους απαντήσεων ανά ερώτηση και κατηγορία, μετράει τον αριθμό των φοιτητών που συμμετείχαν και προβάλλει τα σχόλια που άφησαν. Τα δεδομένα αποδίδονται γραφικά στη σελίδα στατιστικών για εύκολη αξιολόγηση των αποτελεσμάτων.

Δημιουργία Μοναδικών Κωδικών Συμμετοχής

```

@app.route('/teacher/evaluation/codes/generate/<int:evaluation_id>', methods=['POST'])
@login_required('teacher')
def generate_codes(evaluation_id):
    cur = mysql.connection.cursor()
    for _ in range(50):
        random_code = ''.join(random.choices(string.ascii_uppercase + string.digits, k=5))
        code = f'{evaluation_id}-{session['user_id']}-{random_code}'
        cur.execute("INSERT INTO evaluation_codes (evaluation_id, code) VALUES (%s, %s)", (evaluation_id, code))
    mysql.connection.commit()

```

```
flash('Δημιουργήθηκαν 50 νέοι κωδικοί.', 'success')  
return redirect(url_for('evaluation_codes', evaluation_id=evaluation_id))
```

Αυτή η συνάρτηση επιτρέπει στον διδάσκοντα να δημιουργεί γρήγορα 50 νέους κωδικούς συμμετοχής για μια φόρμα αξιολόγησης. Οι κωδικοί δημιουργούνται με τυχαίους συνδυασμούς χαρακτήρων και αριθμών και συνδέονται με τη συγκεκριμένη φόρμα. Μετά τη δημιουργία ο διδάσκων μπορεί να τους δει και να τους διανείμει στους φοιτητές.

4.4 Η βάση του συστήματος

Η βάση δεδομένων αποτελεί ένα από τα σημαντικότερα συστατικά μέρη του συστήματος, καθώς σε αυτή αποθηκεύονται όλα τα απαραίτητα δεδομένα που σχετίζονται με τους χρήστες, τις φόρμες αξιολόγησης, τις ερωτήσεις, τις απαντήσεις και τους μοναδικούς κωδικούς συμμετοχής. Η σωστή σχεδίαση της βάσης δεδομένων εξασφαλίζει την ομαλή λειτουργία της πλατφόρμας, την ασφάλεια των δεδομένων και την αποτελεσματική εξαγωγή στατιστικών πληροφοριών. Για την ανάπτυξη της βάσης δεδομένων χρησιμοποιήθηκε το σύστημα διαχείρισης βάσεων δεδομένων MySQL, το οποίο είναι ιδιαίτερα αξιόπιστο, εύκολο στη διαχείριση και υποστηρίζει πολλά δεδομένα.

Η βάση περιλαμβάνει όλους τους απαραίτητους πίνακες για την υποστήριξη της λειτουργικότητας της πλατφόρμας. Ο πίνακας users περιέχει τα στοιχεία των χρηστών, όπως το όνομα, το επώνυμο, το email, τον ρόλο τους στο σύστημα και τα στοιχεία εισόδου. Η διάκριση των ρόλων επιτρέπει την εφαρμογή διαφορετικών δικαιωμάτων και λειτουργιών για διαχειριστές και διδάσκοντες. Ο πίνακας questions περιέχει τη δεξαμενή ερωτήσεων που χρησιμοποιούνται στις φόρμες αξιολόγησης, ενώ υπάρχει πρόβλεψη για την κατηγοριοποίηση των ερωτήσεων ώστε να είναι εύκολη η επιλογή τους από τους διδάσκοντες κατά τη δημιουργία μιας φόρμας.

Ο πίνακας evaluations καταγράφει τις φόρμες αξιολόγησης που δημιουργούνται από τους διδάσκοντες. Για κάθε φόρμα καταχωρούνται ο τίτλος, η περιγραφή, οι ημερομηνίες έναρξης και λήξης, καθώς και η κατάστασή της (π.χ. αν έχει οριστικοποιηθεί ή όχι). Αυτός ο σχεδιασμός επιτρέπει την εύκολη διαχείριση και παρακολούθηση των διαθέσιμων φορμών στο σύστημα.

Ο πίνακας codes καταγράφει όλους τους μοναδικούς κωδικούς συμμετοχής που δημιουργούνται για κάθε φόρμα αξιολόγησης. Κάθε κωδικός συνδέεται με μία φόρμα και περιέχει πληροφορίες για το αν έχει χρησιμοποιηθεί ή όχι. Με αυτόν τον τρόπο, διασφαλίζεται ότι κάθε φοιτητής συμμετέχει μόνο μία φορά σε κάθε αξιολόγηση διατηρώντας παράλληλα την ανωνυμία της διαδικασίας.

Ο πίνακας answers περιέχει τις απαντήσεις που δίνουν οι φοιτητές στις φόρμες αξιολόγησης. Για κάθε απάντηση αποθηκεύονται ο κωδικός συμμετοχής, η ερώτηση στην οποία αφορά, η τιμή της απάντησης και η ημερομηνία υποβολής. Ακόμη υπάρχει και πεδίο για την αποθήκευση σχολίων, δίνοντας τη δυνατότητα στους φοιτητές να εκφράσουν πιο αναλυτικά τις απόψεις τους για τη διδασκαλία ή το μάθημα.

Η οργάνωση της βάσης δεδομένων ακολουθεί τις αρχές της απλότητας και της αποτελεσματικής αναζήτησης πληροφοριών. Οι βασικές λειτουργίες όπως η δημιουργία, η ανάκτηση, η ενημέρωση και η διαγραφή δεδομένων εκτελούνται γρήγορα και χωρίς προβλήματα. Το σύστημα διαχειρίζεται με ασφάλεια τα δεδομένα αξιολογήσεων, επιτρέποντας την εύκολη εξαγωγή στατιστικών στοιχείων, τη δημιουργία αναφορών και την οπτικοποίηση των αποτελεσμάτων.

Η επιλογή της MySQL ως βάση δεδομένων εξασφαλίζει υψηλή απόδοση και αξιοπιστία, ενώ παράλληλα υποστηρίζεται από ευρύ φάσμα εργαλείων για backup, επαναφορά δεδομένων και ασφάλεια. Η σχεδίαση της βάσης μπορεί να επεκταθεί μελλοντικά αν προκύψουν νέες ανάγκες όπως η υποστήριξη περισσότερων ρόλων χρηστών, η προσθήκη νέων τύπων αξιολογήσεων ή η υλοποίηση συστήματος ειδοποιήσεων.

4.4.1 Περιγραφή των πινάκων και των πεδίων της βάσης δεδομένων

Πίνακας: users

Περιέχει τα στοιχεία των χρηστών του συστήματος (διαχειριστές και διδάσκοντες).

- id: Αύξων αριθμός (Πρωτεύον κλειδί).
- firstname: Όνομα χρήστη.
- lastname: Επώνυμο χρήστη.
- email: Διεύθυνση email (μοναδική για κάθε χρήστη).
- password: Κωδικός πρόσβασης (αποθηκευμένος με hash).
- role: Ρόλος χρήστη στο σύστημα (admin ή teacher).

Πίνακας: questions

Περιέχει τις ερωτήσεις που χρησιμοποιούνται στις φόρμες αξιολόγησης.

- id: Αύξων αριθμός (Πρωτεύον κλειδί).
- text: Το κείμενο της ερώτησης.
- type: Κατηγορία ερώτησης (theory, teacher, facilities, student).

- active: Κατάσταση ερώτησης (1 για ενεργή, 0 για ανενεργή).

Πίνακας: evaluations

Περιέχει τις φόρμες αξιολόγησης που δημιουργούνται από τους διδάσκοντες.

- id: Αύξων αριθμός (Πρωτεύον κλειδί).
- title: Τίτλος φόρμας αξιολόγησης.
- description: Περιγραφή της φόρμας.
- start_date: Ημερομηνία έναρξης της αξιολόγησης.
- end_date: Ημερομηνία λήξης της αξιολόγησης.
- finalized: Κατάσταση φόρμας (1 για οριστικοποιημένη, 0 για επεξεργάσιμη).
- created_by: ID χρήστη που δημιούργησε τη φόρμα.

Πίνακας: evaluation_questions

Συσχέτιση των ερωτήσεων με τις φόρμες αξιολόγησης.

- id: Αύξων αριθμός (Πρωτεύον κλειδί).
- evaluation_id: ID της φόρμας αξιολόγησης.
- question_id: ID της ερώτησης.

Πίνακας: codes

Περιέχει τους μοναδικούς κωδικούς συμμετοχής που χρησιμοποιούν οι φοιτητές για την υποβολή αξιολογήσεων.

- id: Αύξων αριθμός (Πρωτεύον κλειδί).
- code: Ο μοναδικός κωδικός συμμετοχής.
- evaluation_id: ID της φόρμας αξιολόγησης στην οποία αντιστοιχεί.
- used: Κατάσταση χρήσης (1 αν έχει χρησιμοποιηθεί, 0 αν είναι διαθέσιμος).

Πίνακας: answers

Περιέχει τις απαντήσεις των φοιτητών στις ερωτήσεις των φορμών.

- `id`: Αύξων αριθμός (Πρωτεύον κλειδί).
- `code_id`: ID του μοναδικού κωδικού συμμετοχής.
- `question_id`: ID της ερώτησης.
- `answer_value`: Η απάντηση του φοιτητή (τιμή από 1 έως 5).
- `comment`: Προαιρετικό σχόλιο του φοιτητή.
- `created_at`: Ημερομηνία και ώρα υποβολής της απάντησης.

Κεφάλαιο 5ο: Αποτίμηση Αποτελεσμάτων και Μελλοντικές Κατευθύνσεις

Παρουσιάστηκε και αναλύθηκε η διαδικτυακή πλατφόρμα συλλογής και ανάλυσης αξιολογήσεων διδασκόντων από φοιτητές, με στόχο τη βελτίωση της ποιότητας της εκπαιδευτικής διαδικασίας. Μέσα από την υλοποίηση του συστήματος καταφέραμε την δημιουργία ενός λειτουργικού περιβάλλοντος, όπου οι φοιτητές μπορούν να συμμετέχουν σε διαδικασίες αξιολόγησης με απλό και ανώνυμο τρόπο και οι διδάσκοντες έχουν άμεση πρόσβαση σε χρήσιμα στατιστικά δεδομένα για τη βελτίωση της διδασκαλίας τους.

Το πρώτο βασικό συμπέρασμα είναι ότι η πλατφόρμα ανταποκρίθηκε στις ανάγκες τόσο των φοιτητών όσο και των διδασκόντων, παρέχοντας ένα εύχρηστο και ευέλικτο εργαλείο για τη συλλογή αξιολογήσεων. Οι διαδικασίες ήταν απλοποιημένες κάτι που διευκόλυνε τη χρήση του συστήματος ακόμη και από άτομα χωρίς ιδιαίτερη τεχνολογική εξοικείωση.

Δεύτερο σημαντικό συμπέρασμα αποτελεί η θετική συμβολή του συστήματος στην άμεση παρουσίαση και κατανόηση των αποτελεσμάτων μέσω γραφικών απεικονίσεων. Αυτό διευκολύνει σημαντικά τους διδάσκοντες να εξάγουν συμπεράσματα σχετικά με την ποιότητα της διδασκαλίας τους και να εντοπίζουν τα σημεία που χρειάζονται βελτίωση.

Τρίτο συμπέρασμα είναι ότι η χρήση εργαλείων όπως το Flask, η MySQL και το Chart.js απέδειξε πως ακόμη και με απλά και δωρεάν εργαλεία μπορούν να υλοποιηθούν αξιόπιστα συστήματα που καλύπτουν πλήρως τις ανάγκες μικρών και μεσαίων ακαδημαϊκών οργανισμών.

Όσον αφορά τις προτάσεις βελτίωσης, η πρώτη αφορά την προσθήκη δυνατότητας εξαγωγής των στατιστικών σε μορφή PDF ή Excel, ώστε οι διδάσκοντες να μπορούν εύκολα να αποθηκεύουν και να μοιράζονται τα αποτελέσματα των αξιολογήσεων.

Η δεύτερη πρόταση αφορά την επέκταση της πλατφόρμας για την υποστήριξη ειδοποιήσεων μέσω email προς τους φοιτητές, υπενθυμίζοντάς τους να συμμετάσχουν στις αξιολογήσεις πριν από την ημερομηνία λήξης, με στόχο την αύξηση της συμμετοχής.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] <https://www.explorance.com/products/blue>
- [2] <https://www.watermarkinsights.com/solutions/course-evaluation-institutional-surveys/>
- [3] <https://info.smartevals.com/>
- [4] [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
- [5] <https://www.python.org/>
- [6] <https://www.w3schools.com/python/>
- [7] <https://realpython.com/>
- [8] <https://www.geeksforgeeks.org/python-programming-language-tutorial/>
- [9] <https://www.stxnext.com/blog/flask-vs-django-comparison>
- [10] <https://www.mytaskpanel.com/what-is-flask/>
- [11] <https://careerfoundry.com/en/blog/web-development/what-is-flask/>
- [12] <https://www.geeksforgeeks.org/python-introduction-to-web-development-using-flask/>
- [13] <https://www.w3schools.com/MySQL/default.asp>
- [14] <https://www.remoteplatz.ch/en/blog/advantages-and-disadvantages-of-mysql>
- [15] <https://getbootstrap.com/>
- [16] <https://owdt.com/article/the-pros-and-cons-of-using-bootstrap-for-front-end-development/>
- [17] <https://www.chartjs.org/>
- [18] https://www.tutorialspoint.com/chartjs/chartjs_introduction.htm

ΠΑΡΑΡΤΗΜΑ - Κώδικες

```
from flask import Flask, render_template, request, redirect, url_for, session, flash
from flask_mysql import MySQL
from datetime import datetime
from config import Config
import random, string
from functools import wraps

app = Flask(__name__)
app.config.from_object(Config)
mysql = MySQL(app)

def login_required(role=None):
    def decorator(f):
        @wraps(f)
        def decorated_function(*args, **kwargs):
            if 'user_id' not in session:
                return redirect(url_for('login'))
            if role and session.get('role') != role:
                flash('Μη εξουσιοδοτημένη πρόσβαση.', 'danger')
                return redirect(url_for('login'))
            return f(*args, **kwargs)
        return decorated_function
    return decorator

@app.route('/')
def index():
    return redirect(url_for('login'))

@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        cur = mysql.connection.cursor()
        cur.execute("SELECT * FROM users WHERE username=%s AND password=%s", (username, password))
        user = cur.fetchone()
        if user:
            session['user_id'] = user[0]
            session['username'] = user[1]
            session['role'] = user[5]
            return redirect(url_for(f"dashboard_{session['role']}"))
        flash('Λάθος στοιχεία.', 'danger')
    return render_template('login.html')

@app.route('/register', methods=['GET', 'POST'])
def register():
```

```

if request.method == 'POST':
    username = request.form['username']
    password = request.form['password']
    firstname = request.form['firstname']
    lastname = request.form['lastname']
    role = request.form['role']
    cur = mysql.connection.cursor()
    cur.execute("INSERT INTO users (username, password, firstname, lastname, role) VALUES (%s, %s, %s, %s, %s)",
                (username, password, firstname, lastname, role))
    mysql.connection.commit()
    flash('Εγγραφή επιτυχής. Συνδεθείτε.', 'success')
    return redirect(url_for('login'))
return render_template('register.html')

@app.route('/logout')
def logout():
    session.clear()
    return redirect(url_for('login'))

@app.route('/dashboard/admin')
@login_required('admin')
def dashboard_admin():
    return render_template('dashboard_admin.html')

@app.route('/dashboard/teacher')
@login_required('teacher')
def dashboard_teacher():
    cur = mysql.connection.cursor()
    user_id = session['user_id']

    # Αριθμός Φορμών
    cur.execute("SELECT COUNT(*) FROM evaluations WHERE created_by=%s", (user_id,))
    forms_count = cur.fetchone()[0]

    # Αριθμός Κωδικών
    cur.execute("""
        SELECT COUNT(*) FROM evaluation_codes ec
        JOIN evaluations e ON ec.evaluation_id = e.id
        WHERE e.created_by=%s
    """, (user_id,))
    codes_count = cur.fetchone()[0]

    # Αριθμός Απαντημένων Φορμών (όχι απαντήσεις, αλλά ολοκληρωμένες αξιολογήσεις)
    cur.execute("""
        SELECT COUNT(DISTINCT ec.id)
        FROM evaluation_codes ec
        JOIN evaluations e ON ec.evaluation_id = e.id
        WHERE e.created_by=%s AND ec.used=1
    """, (user_id,))

```

```

        answered_forms_count = cur.fetchone()[0]

        return render_template('dashboard_teacher.html',
                               forms_count=forms_count,
                               codes_count=codes_count,
                               answered_forms_count=answered_forms_count)

@app.route('/admin/questions')
@login_required('admin')
def questions_list():
    cur = mysql.connection.cursor()
    cur.execute("SELECT * FROM questions")
    questions = cur.fetchall()
    return render_template('questions_list.html', questions=questions)

@app.route('/admin/question/create', methods=['GET', 'POST'])
@login_required('admin')
def question_create():
    if request.method == 'POST':
        text = request.form['text']
        q_type = request.form['type']
        cur = mysql.connection.cursor()
        cur.execute("INSERT INTO questions (text, type) VALUES (%s, %s)", (text, q_type))
        mysql.connection.commit()
        flash('Η ερώτηση προστέθηκε.', 'success')
        return redirect(url_for('questions_list'))
    return render_template('question_form.html', action="create")

@app.route('/admin/question/edit/<int:id>', methods=['GET', 'POST'])
@login_required('admin')
def question_edit(id):
    cur = mysql.connection.cursor()
    if request.method == 'POST':
        text = request.form['text']
        q_type = request.form['type']
        cur.execute("UPDATE questions SET text=%s, type=%s WHERE id=%s", (text, q_type, id))
        mysql.connection.commit()
        flash('Η ερώτηση ενημερώθηκε.', 'success')
        return redirect(url_for('questions_list'))
    cur.execute("SELECT * FROM questions WHERE id=%s", (id,))
    question = cur.fetchone()
    return render_template('question_form.html', action="edit", question=question)

@app.route('/admin/question/delete/<int:id>', methods=['POST'])
@login_required('admin')
def question_delete(id):
    # Έλεγχος αν χρησιμοποιείται σε κάποια φόρμα
    cur = mysql.connection.cursor()

```

```

cur.execute("SELECT COUNT(*) FROM evaluation_questions WHERE question_id=%s", (id,))
if cur.fetchone()[0] > 0:
    flash('Η ερώτηση δεν μπορεί να διαγραφεί, χρησιμοποιείται ήδη.', 'danger')
else:
    cur.execute("DELETE FROM questions WHERE id=%s", (id,))
    mysql.connection.commit()
    flash('Η ερώτηση διαγράφηκε.', 'success')
return redirect(url_for('questions_list'))

@app.route('/teacher/evaluations')
@login_required('teacher')
def evaluations_list():
    cur = mysql.connection.cursor()
    cur.execute("SELECT * FROM evaluations WHERE created_by=%s", (session['user_id'],))
    evaluations = cur.fetchall()
    return render_template('evaluations_list.html', evaluations=evaluations)

@app.route('/teacher/evaluation/create', methods=['GET', 'POST'])
@login_required('teacher')
def evaluation_create():
    cur = mysql.connection.cursor()

    if request.method == 'POST':
        title = request.form['title']
        description = request.form['description']
        start_date = datetime.strptime(request.form['start_date'], '%d-%m-%Y').strftime('%Y-%m-%d')
        end_date = datetime.strptime(request.form['end_date'], '%d-%m-%Y').strftime('%Y-%m-%d')
        selected_questions = request.form.getlist('questions')

        cur.execute("""
            INSERT INTO evaluations (title, description, start_date, end_date, created_by, finalized)
            VALUES (%s, %s, %s, %s, %s, 0)
            """, (title, description, start_date, end_date, session['user_id']))

        evaluation_id = cur.lastrowid

        for qid in selected_questions:
            cur.execute("INSERT INTO evaluation_questions (evaluation_id, question_id) VALUES (%s, %s)",
                (evaluation_id, qid))

        mysql.connection.commit()
        flash('Η φόρμα δημιουργήθηκε επιτυχώς.', 'success')
        return redirect(url_for('evaluations_list'))

# Φέρνουμε όλες τις ερωτήσεις και τις ομαδοποιούμε
cur.execute("SELECT * FROM questions ORDER BY type")
questions = cur.fetchall()

grouped_questions = {'course': [], 'teacher': [], 'facilities': [], 'student': []}
for q in questions:

```

```

        grouped_questions[q[2]].append(q)

    return render_template('evaluation_form.html',
                          action="create",
                          grouped_questions=grouped_questions)

@app.route('/teacher/evaluation/edit/<int:id>', methods=['GET', 'POST'])
@login_required('teacher')
def evaluation_edit(id):
    cur = mysql.connection.cursor()

    # Ανάκτηση της φόρμας
    cur.execute("SELECT * FROM evaluations WHERE id=%s AND created_by=%s", (id, session['user_id']))
    evaluation = cur.fetchone()
    if not evaluation:
        flash('Η φόρμα δεν βρέθηκε.', 'danger')
        return redirect(url_for('evaluations_list'))

    is_finalized = evaluation[6] == 1

    if request.method == 'POST':
        title = request.form['title']
        description = request.form['description']
        start_date = datetime.strptime(request.form['start_date'], '%d-%m-%Y').strftime('%Y-%m-%d')
        end_date = datetime.strptime(request.form['end_date'], '%d-%m-%Y').strftime('%Y-%m-%d')

        cur.execute("""
            UPDATE evaluations
            SET title=%s, description=%s, start_date=%s, end_date=%s
            WHERE id=%s AND created_by=%s
            """, (title, description, start_date, end_date, id, session['user_id']))

        # Οριστικοποίηση
        if not is_finalized and 'finalize' in request.form:
            cur.execute("UPDATE evaluations SET finalized=1 WHERE id=%s", (id,))
            is_finalized = True # Αλλάζει για την τρέχουσα εμφάνιση

        # Αν δεν έχει οριστικοποιηθεί, ενημερώνουμε και τις ερωτήσεις
        if not is_finalized:
            selected_questions = request.form.getlist('questions')
            cur.execute("DELETE FROM evaluation_questions WHERE evaluation_id=%s", (id,))
            for qid in selected_questions:
                cur.execute("INSERT INTO evaluation_questions (evaluation_id, question_id) VALUES (%s, %s)", (id, qid))

    mysql.connection.commit()
    flash('Η φόρμα ενημερώθηκε.', 'success')
    return redirect(url_for('evaluations_list'))

```

```

# Ανάκτηση επιλεγμένων ερωτήσεων και όλων των ερωτήσεων ομαδοποιημένων
cur.execute("SELECT question_id FROM evaluation_questions WHERE evaluation_id=%s", (id,))
selected_questions = [row[0] for row in cur.fetchall()]

cur.execute("SELECT * FROM questions ORDER BY type")
questions = cur.fetchall()

# Ομαδοποίηση στο backend για απλότητα στο template
grouped_questions = {'course': [], 'teacher': [], 'facilities': [], 'student': []}
for q in questions:
    grouped_questions[q[2]].append(q)

return render_template('evaluation_form.html',
                       action="edit",
                       evaluation=evaluation,
                       grouped_questions=grouped_questions,
                       selected_questions=selected_questions,
                       is_finalized=is_finalized)

@app.route('/teacher/evaluation/delete/<int:id>', methods=['POST'])
@login_required('teacher')
def evaluation_delete(id):
    cur = mysql.connection.cursor()
    cur.execute("DELETE FROM evaluations WHERE id=%s AND created_by=%s", (id, session['user_id']))
    mysql.connection.commit()
    flash('Η φόρμα διαγράφηκε.', 'success')
    return redirect(url_for('evaluations_list'))

# Στατιστικά Αποτελέσματα
@app.route('/teacher/evaluation/stats/<int:evaluation_id>')
@login_required('teacher')
def evaluation_stats(evaluation_id):
    cur = mysql.connection.cursor()

    # Τίτλος και Ημερομηνίες της Φόρμας
    cur.execute("""
        SELECT title, start_date, end_date
        FROM evaluations
        WHERE id=%s
    """, (evaluation_id,))
    eval_info = cur.fetchone()

    # Μέσος Όρος για κάθε Ερώτηση
    cur.execute("""
        SELECT q.text, q.type, AVG(a.rating)
        FROM answers a
        JOIN questions q ON a.question_id = q.id
        JOIN evaluation_codes ec ON a.evaluation_code_id = ec.id
    """)

```

```

WHERE ec.evaluation_id = %s
GROUP BY q.id
ORDER BY q.type
"", (evaluation_id,))
stats = cur.fetchall()

# Μέσος Όρος ανά Κατηγορία
cur.execute("""
SELECT q.type, AVG(a.rating)
FROM answers a
JOIN questions q ON a.question_id = q.id
JOIN evaluation_codes ec ON a.evaluation_code_id = ec.id
WHERE ec.evaluation_id = %s
GROUP BY q.type
"", (evaluation_id,))
category_averages = cur.fetchall()

# Πλήθος Φοιτητών που Υποβάλαν Αξιολόγηση
cur.execute("""
SELECT COUNT(DISTINCT ec.id)
FROM evaluation_codes ec
JOIN answers a ON ec.id = a.evaluation_code_id
WHERE ec.evaluation_id = %s
"", (evaluation_id,))
student_count = cur.fetchone()[0]

# Σχόλια
cur.execute("""
SELECT comments
FROM evaluation_codes
WHERE evaluation_id = %s AND comments IS NOT NULL AND comments != ''
"", (evaluation_id,))
comments = [row[0] for row in cur.fetchall()]

return render_template('statistics.html',
                       eval_info=eval_info,
                       stats=stats,
                       category_averages=category_averages,
                       student_count=student_count,
                       comments=comments)

# Σελίδα Προβολής Κωδικών
@app.route('/teacher/evaluation/codes/<int:evaluation_id>')
@login_required('teacher')
def evaluation_codes(evaluation_id):
    cur = mysql.connection.cursor()

# Pagination Parameters

```

```

page = int(request.args.get('page', 1))
per_page = 50
offset = (page - 1) * per_page

# Get Total Count for Pagination
cur.execute("SELECT COUNT(*) FROM evaluation_codes WHERE evaluation_id=%s", (evaluation_id,))
total_codes = cur.fetchone()[0]
total_pages = (total_codes + per_page - 1) // per_page

# Fetch Paged Results
cur.execute("""
    SELECT code, used, used_at
    FROM evaluation_codes
    WHERE evaluation_id=%s
    ORDER BY id DESC
    LIMIT %s OFFSET %s
""", (evaluation_id, per_page, offset))
codes = cur.fetchall()

return render_template('evaluation_codes.html',
                      codes=codes,
                      evaluation_id=evaluation_id,
                      page=page,
                      total_pages=total_pages)

# Δημιουργία 50 νέων κωδικών και ανανέωση προβολής
@app.route('/teacher/evaluation/codes/generate/<int:evaluation_id>', methods=['POST'])
@login_required('teacher')
def generate_codes(evaluation_id):
    cur = mysql.connection.cursor()
    for _ in range(50):
        random_code = ''.join(random.choices(string.ascii_uppercase + string.digits, k=5))
        code = f"{evaluation_id}-{session['user_id']}-{random_code}"
        cur.execute("INSERT INTO evaluation_codes (evaluation_id, code) VALUES (%s, %s)", (evaluation_id, code))
    mysql.connection.commit()
    flash('Δημιουργήθηκαν 50 νέοι κωδικοί.', 'success')
    return redirect(url_for('evaluation_codes', evaluation_id=evaluation_id))

@app.route('/evaluation/submit', methods=['GET', 'POST'])
def evaluation_submit_code():
    if request.method == 'POST':
        code_input = request.form['code'].strip().upper()

        cur = mysql.connection.cursor()
        cur.execute("""
            SELECT ec.id, ec.evaluation_id, ec.used, e.title, e.start_date, e.end_date
            FROM evaluation_codes ec

```

```

        JOIN evaluations e ON ec.evaluation_id = e.id
        WHERE ec.code = %s
    """ , (code_input,))
    code_data = cur.fetchone()

    if not code_data:
        flash('Ο κωδικός δεν βρέθηκε.', 'danger')
        return redirect(url_for('evaluation_submit_code'))
    if code_data[2] == 1:
        flash('Ο κωδικός έχει ήδη χρησιμοποιηθεί.', 'danger')
        return redirect(url_for('evaluation_submit_code'))

    # Περνάμε και τις πληροφορίες της φόρμας για προβολή
    return render_template('evaluation_enter_code.html',
                           code_found=True,
                           code_info=code_data)

return render_template('evaluation_enter_code.html', code_found=False)

@app.route('/evaluation/submit/form/<int:code_id>', methods=['GET', 'POST'])
def evaluation_form_submit(code_id):
    cur = mysql.connection.cursor()

    # Ανάκτηση στοιχείων κωδικού και φόρμας
    cur.execute("SELECT code, evaluation_id, used FROM evaluation_codes WHERE id=%s", (code_id,))
    code_data = cur.fetchone()
    if not code_data or code_data[2] == 1:
        flash('Ο κωδικός δεν είναι έγκυρος ή έχει χρησιμοποιηθεί.', 'danger')
        return redirect(url_for('evaluation_submit_code'))

    evaluation_id = code_data[1]

    if request.method == 'POST':
        answers = request.form
        comments = answers.get('comments', '').strip()

        # Αποθήκευση απαντήσεων
        for question_id, rating in answers.items():
            if question_id.startswith('q_'):
                question_id_clean = question_id.replace('q_', '')
                cur.execute("""
                    INSERT INTO answers (evaluation_code_id, question_id, rating)
                    VALUES (%s, %s, %s)
                """, (code_id, question_id_clean, rating))

        # Αποθήκευση σχολίων αν υπάρχουν
        if comments:

```

```

cur.execute("""
    UPDATE evaluation_codes
    SET comments=%s
    WHERE id=%s
    """, (comments, code_id))

# Σημειώνουμε ότι ο κωδικός χρησιμοποιήθηκε
cur.execute("""
    UPDATE evaluation_codes
    SET used=1, used_at=NOW()
    WHERE id=%s
    """, (code_id,))

mysql.connection.commit()
flash('Η αξιολόγηση υποβλήθηκε επιτυχώς. Ευχαριστούμε!', 'success')
return redirect(url_for('evaluation_submit_code'))

# Φέρνουμε τις ερωτήσεις της φόρμας
cur.execute("""
    SELECT q.id, q.text, q.type
    FROM evaluation_questions eq
    JOIN questions q ON eq.question_id = q.id
    WHERE eq.evaluation_id=%s
    ORDER BY q.type
    """, (evaluation_id,))
questions = cur.fetchall()

# Ομαδοποιούμε τις ερωτήσεις
grouped_questions = {'course': [], 'teacher': [], 'facilities': [], 'student': []}
for q in questions:
    grouped_questions[q[2]].append(q)

return render_template('evaluation_form_student.html',
                       code=code_data[0],
                       grouped_questions=grouped_questions)

if __name__ == '__main__':
    app.run(debug=True)

```