



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ**

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ
ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

**Διαμορφώσιμο, Βασιζόμενο στον
Παγκόσμιο Ιστό Σύστημα
Διαχείρισης Βαθμολογιών
(Configurable, Web-Based Grade
Management System)**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Καμπουράκη Ηλία

Επιβλέπων: Κυριάκος Κρητικός (Αν. Καθηγητής)

**Μέλη εξεταστικής επιτροπής: Γκουμόπουλος Χρήστος (Αν. Καθηγητής), Κωστούλας
Θεόδωρος (Αν. Καθηγητής)**

Σάμος, Οκτώβριος 2023

Περίληψη

Η διαχείριση των μαθημάτων μπορεί να αποτελέσει μια χρονοβόρα και περίπλοκη διαδικασία, ειδικά όταν ο διδάσκων έχει να διαχειριστεί μεγάλο πλήθος μαθημάτων. Με τη βοήθεια, όμως, των ηλεκτρονικών υπολογιστών, δίνεται η δυνατότητα (εν μέρει) αυτοματοποίησης της διαδικασίας αυτής μέσω της αποθήκευσης, διαχείρισης, βαθμολόγησης και εξαγωγής των αναλυτικών ή τυπικών βαθμολογιών για το κάθε μάθημα.

Τα παραπάνω επιτυγχάνονται με τη χρήση του συστήματος λογισμικού που αναπτύχθηκε στην παρούσα διπλωματική. Το σύστημα αυτό που είναι βασισμένο στον Παγκόσμιο Ιστό, συνεισφέρει στη καλύτερη διαχείριση της βαθμολογίας των μαθημάτων που έχει αναλάβει ο διδάσκων, καθώς και στη μείωση του χρόνου βαθμολόγησης και εξαγωγής των βαθμολογιών φοιτητών. Ένα από τα σημαντικότερα πλεονεκτήματα που προσφέρει είναι πως επιτρέπει τον προσδιορισμό και την διαχείριση ιεραρχίας βαθμολογιών για ένα μάθημα, καλύπτοντας με ευέλικτο τρόπο τις διάφορες μεθόδους αξιολόγησης φοιτητών που μπορεί να εφαρμοστούν για το θεωρητικό και εργαστηριακό μέρος του μαθήματος. Επομένως, κάθε κλαδί της ιεραρχίας μπορεί να περιλαμβάνει την αξιολόγηση/βαθμολόγηση για επιμέρους μέρη της θεωρίας ή του εργαστηρίου του μαθήματος, όπως είναι οι ασκήσεις, οι πρόοδοι και οι τελικές γραπτές εξετάσεις.

Λέξεις κλειδιά: βαθμολόγηση, αξιολόγηση, διαχείριση μαθημάτων, εφαρμογή ιστού, RESTful API

Summary

Course management can be a time-consuming and complex process, especially when a professor has to manage a big amount of courses. However, with the assistance of computer machines, the capability to (partially) automate this process can be realised through the storage and management of courses, the grading of its students as well as the extraction of the respective analytical or typical grades.

The above are accomplished via the use of a software system that has been developed in the context of the current thesis. This system, which is based on the World Wide Web, contributes to the better management of course grading (as undertaken by the course professor) as well as to the reduction of grading and grade extraction time. One of its biggest advantages offered is that it allows defining and managing a grading hierarchy for a course, covering in a flexible manner the various student assessment methods that can be applied on the theoretical and laboratorial section of the course. Thus, every branch of the hierarchy can include the assessment of the various theoretical or laboratorial course parts, such as exercises, projects, midterm exams and final, written exams.

Keywords: grading, assessment, course management, web application, RESTful API

Πίνακας Περιεχομένων

1.	Εισαγωγή.....	9
1.1	Αντικείμενο της διπλωματικής εργασίας	9
1.2.	Δομή αναφοράς.....	9
2.	Σχετικές Εργασίες.....	11
2.1.	Ανάλυση σχετικών συστημάτων.....	11
2.2.	Σύγκριση με το προτεινόμενο σύστημα	12
3.	Ανάπτυξη Συστήματος	14
3.1.	Διαδικασία Ανάπτυξης Συστήματος	14
3.2.	Απαιτήσεις συστήματος.....	15
3.2.1	Λειτουργικές απαιτήσεις	15
3.2.2	Μη λειτουργικές απαιτήσεις	17
3.3.	Σχεδίαση Συστήματος	18
3.3.1.	Αρχιτεκτονική Συστήματος	19
3.3.2.	Σχεδίαση διεπαφής χρήστη	21
3.3.3.	Διάγραμμα περιβάλλοντος.....	22
3.3.4.	Διάγραμμα δραστηριοτήτων	23
3.3.5.	Διάγραμμα περιπτώσεων χρήσης	25
3.3.6.	Διάγραμμα ακολουθίας.....	25
3.4.	Υλοποίηση Συστήματος	27
3.4.1.	Διάγραμμα πακέτων	27
3.4.2.	Τεχνολογίες & Εργαλεία Back-end Συστήματος.....	28
3.4.3.	Τεχνολογίες & Εργαλεία Front-end Συστήματος.....	31
3.4.4.	Εργαλεία Ανάπτυξης Συστήματος.....	32
3.5.	Βαθμός Ικανοποίησης Απαιτήσεων.....	34
4.	Επίδειξη Συστήματος	40
4.1.	Διαδικασία Εγκατάστασης & Εκτέλεσης Συστήματος	40
4.2.	Περιπτώσεις χρήσης	41
4.2.1.	Αρχική Σελίδα, Εγγραφή και Σύνδεση	41
4.2.2.	Διαχείριση Διδασκαλιών και Σπουδαστών.....	43

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

4.2.3.	Επεξεργασία στοιχείων μαθήματος και βαθμολόγηση σπουδαστή	45
4.2.4.	Εξαγωγή βαθμολογίας.....	47
5.	Συμπεράσματα & Μελλοντική Εργασία	49
5.1.	Συμπεράσματα.....	49
5.2.	Μελλοντική εργασία.....	49
	ΒΙΒΛΙΟΓΡΑΦΙΑ.....	50

Πίνακας Εικόνων

Εικόνα 1 - Διάγραμμα Αρχιτεκτονικής του Συστήματος.....	20
Εικόνα 2 - Εξειδικευμένο Διάγραμμα Αρχιτεκτονικής του Συστήματος	21
Εικόνα 3 - Διάγραμμα Περιβάλλοντος	23
Εικόνα 4 - Διάγραμμα Διαδικασιών	24
Εικόνα 5 - Διάγραμμα Περιπτώσεων Χρήσης.....	25
Εικόνα 6 - Διάγραμμα Ακολουθίας	26
Εικόνα 7 - Διάγραμμα Πακέτων	28
Εικόνα 8 - Δέντρο Οντοτήτων	32
Εικόνα 9 - Αρχική Σελίδα.....	41
Εικόνα 10 - Φόρμα Εγγραφής.....	42
Εικόνα 11 - Φόρμα Σύνδεσης.....	43
Εικόνα 12 - Λίστα μαθημάτων που έχουν ανατεθεί στον διδάσκων	44
Εικόνα 13 - Εκφόρτωση Διδασκαλιών από αρχείο.....	44
Εικόνα 14 - Διαχείριση φοιτητών/σπουδαστών μιας διδασκαλίας	45
Εικόνα 15 - Ανέβασμα σπουδαστών.....	45
Εικόνα 16 - Επεξεργασία στοιχείων μαθήματος.....	46
Εικόνα 17 - Βαθμολόγηση σπουδαστών.....	47
Εικόνα 18 - Επιλογή εξαγωγής Τυπικής ή Αναλυτικής βαθμολογίας	47
Εικόνα 19 - Τυπική Βαθμολογία.....	48
Εικόνα 20 - Αναλυτική Βαθμολογία.....	48

Λίστα Πινάκων

Πίνακας 1 - Βαθμός Ικανοποίησης Λειτουργικών Απαιτήσεων.....	34
Πίνακας 2 - Βαθμός Ικανοποίησης Μη Λειτουργικών Απαιτήσεων.....	38

1. Εισαγωγή

1.1 Αντικείμενο της διπλωματικής εργασίας

Η διαδικασία της διαχείρισης των σπουδαστών και των διδασκαλιών έχει αποκτήσει μεγάλο ενδιαφέρον τα τελευταία χρόνια με την εισαγωγή των υπολογιστών στα εκπαιδευτικά ιδρύματα. Τα εκπαιδευτικά ιδρύματα χρησιμοποιούν όλο και περισσότερο τους υπολογιστές με σκοπό την αυτοματοποίηση και την απλοποίηση της συγκεκριμένης διαδικασίας, χρησιμοποιώντας συστήματα λογισμικού που παρέχουν υποστήριξη και εν μέρη αυτοματοποίηση. Όμως, λόγω του μεγάλου αριθμού των εκπαιδευτικών ιδρυμάτων και των διαφορετικών αναγκών τους, οι οποίες έγκειται εν μέρει στην ετερογένεια των τμημάτων και των σχολών που τα απαρτίζουν αλλά και στην ετερογένεια των διαδικασιών που ακολουθούν, κάθε ίδρυμα αναπτύσσει ή αγοράζει ένα δικό του σύστημα καθώς και το παραμετροποιεί και το διαμορφώνει με βάση τις δικές του ανάγκες. Επίσης, σε ορισμένες περιπτώσεις, ιδρύματα δύναται να χρησιμοποιούν παραπάνω από ένα λογισμικό για την διαχείριση αυτή.

Η παρούσα διπλωματική εργασία προτείνει ένα διαμορφώσιμο σύστημα λογισμικού, βασιζόμενο στον παγκόσμιο ιστό, για την διαχείριση των μαθημάτων, των σπουδαστών και των βαθμολογιών τους. Στο σύστημα αυτό παρέχονται οι δυνατότητες:

- της εισαγωγής και επεξεργασίας όλων των μαθημάτων που διδάσκει ο χρήστης (διδάσκων),
- της δημιουργίας και επεξεργασίας των διδασκαλιών ανά μάθημα,
- της εισαγωγής και επεξεργασίας των σπουδαστών σε διδασκαλίες μαθημάτων,
- της βαθμολόγησης των σπουδαστών σε μια διδασκαλία μαθήματος,
- της εξαγωγής της βαθμολογίας όλων των σπουδαστών σε μια διδασκαλία μαθήματος.

Όλες οι παραπάνω δυνατότητες έχουν ως αποτέλεσμα την ελάττωση του χρόνου που απαιτείται για την διαδικασία της διαχείρισης των διδασκαλιών και την διαδικασία της βαθμολόγησης των σπουδαστών καθώς και την απλοποίηση των παραπάνω διαδικασιών με τη χρήση ενός μόνο συστήματος. Επιπλέον, λόγω του βαθμού ευελιξίας που παρέχει το σύστημα λογισμικού ως προς την διαμόρφωση των αξιολογήσεων των σπουδαστών, το σύστημα αυτό μπορεί να καλύψει όλα τα πιθανά σενάρια αξιολόγησης/βαθμολόγησης.

1.2. Δομή αναφοράς

- ΚΕΦΑΛΑΙΟ 2^ο – «Σχετικές Εργασίες»: Στο κεφάλαιο αυτό παρουσιάζονται και αναλύονται εργασίες σχετικές με το θέμα της παρούσας διπλωματικής εργασίας.
- ΚΕΦΑΛΑΙΟ 3^ο – «Ανάπτυξη Συστήματος»: Στο κεφάλαιο αυτό πραγματοποιείται η ανάλυση των φάσεων ανάπτυξης του συστήματος λογισμικού που κατασκευάστηκε στα πλαίσια της παρούσας εργασίας.
- ΚΕΦΑΛΑΙΟ 4^ο – «Επίδειξη Συστήματος»: Στο κεφάλαιο αυτό περιγράφεται η διαδικασία εγκατάστασης και εκτέλεσης του συστήματος καθώς και επιδεικνύονται ορισμένες

- περιπτώσεις χρήσης του.
- ΚΕΦΑΛΑΙΟ 5^ο – «Συμπεράσματα & Μελλοντική Εργασία»: Παρουσιάζονται τα συμπεράσματα που προέκυψαν από την ανάπτυξη του συστήματος καθώς και μελλοντικές βελτιώσεις που θα μπορούσαν να πραγματοποιηθούν σε αυτό.

2. Σχετικές Εργασίες

Στο κεφάλαιο αυτό θα γίνει η ανάλυση ορισμένων εργασιών που είναι σχετικές με το αντικείμενο και τη συνεισφορά της διπλωματικής εργασίας.

2.1. Ανάλυση σχετικών συστημάτων

Classter¹

Ξεκινάμε την ανάλυσή μας με ένα ήδη αναπτυγμένο και σε λειτουργία σύστημα, το Classter. Αυτό το σύστημα είναι βασισμένο στο παγκόσμιο ιστό (World Wide Web – WWW) και στο υπολογιστικό νέφος (cloud). Προσφέρει μια πλήρως λειτουργική λύση για τις καθημερινές λειτουργίες εκπαιδευτικών οργανισμών, όπως την δημιουργία εργασιών και την βαθμολογία αυτών για κάθε μάθημα, το παρουσιολόγιο, τις καρτέλες σπουδαστών κ.α. Το σύστημα αυτό παρέχεται με συνδρομή ενώ διαθέτει και εφαρμογή κατάλληλη για κινητές συσκευές.

Error! Reference source not found.²

Η πλατφόρμα eClass είναι ένα ολοκληρωμένο σύστημα λογισμικού διαχείρισης μαθημάτων που αναπτύχθηκε από την αστική εταιρεία μη κερδοσκοπικού χαρακτήρα GUnet με σκοπό την ασύγχρονη τηλεκπαίδευση. Η πλατφόρμα αυτή χρησιμοποιείται από όλα σχεδόν τα ελληνικά εκπαιδευτικά ιδρύματα. Υποστηρίζει την οργάνωση, αποθήκευση και παρουσίαση του ψηφιακού εκπαιδευτικού υλικού που προσφέρεται στους σπουδαστές (πχ. σημειώσεις). Επίσης, επιτρέπει την βαθμολόγηση και την εξέταση των σπουδαστών με ερωτήσεις πολλαπλής επιλογής ή γραπτού κειμένου. Τέλος, υποστηρίζει την διαχείριση βαθμολογιών για κάθε μάθημα.

Error! Reference source not found.³

Η πλατφόρμα Moodle είναι και αυτή με τη σειρά της ένα ολοκληρωμένο σύστημα διαχείρισης μαθημάτων που δημιουργήθηκε το 1999 από τον Martin Dougiamas. Παρέχεται δωρεάν ως ελεύθερο λογισμικό και μπορεί να εκτελεστεί σε οποιοδήποτε σύστημα υποστηρίζει PHP ενώ παράλληλα μπορεί να συνδυαστεί με πολλούς τύπους βάσεων δεδομένων. Διαθέτει ποικιλία εργαλείων για διάφορες σχετικές λειτουργίες: διαχείριση μαθημάτων, εγγράφων, υποστήριξη για φόρουμ και συζητήσεις, υποστήριξη αξιολόγησης μέσω εργασιών και βαθμολόγιο.

Martín Llamas-Nistal, Manuel J. Fernández-Iglesias, Juan González-Tato, and Fernando A. Mikic-Fonte. 2013. Blended e-assessment. *Comput. Educ.* 62, C (March 2013), 72–87.

Στη συγκεκριμένη εργασία προτείνεται ένα σύστημα αξιολόγησης των γραπτών εξετάσεων και των εργασιών σε ηλεκτρονική μορφή. Η δυνατότητα αυτή επιτυγχάνεται με τη χρήση κωδικών που μοιράζονται στους σπουδαστές και τοποθετούνται στην αρχική σελίδα του γραπτού, το οποίο

¹ <https://www.classter.com/>

² <https://www.openeeclass.org/>

³ <https://moodle.com/>

στη συνέχεια σαρώνεται και αποθηκεύεται στον εξυπηρετητή. Στη συνέχεια ο καθηγητής έχει τη δυνατότητα να βαθμολογήσει και να αιτιολογήσει κάθε ερώτημα της γραπτής εξέτασης που επιθυμεί για κάθε σπουδαστή. Επίσης, υπάρχει η δυνατότητα πρόσβασης του γραπτού, μετά την βαθμολόγηση, από τους σπουδαστές και δυνατότητα σχολιασμού μερών ασκήσεων. Στο σύστημα αυτό δεν προσφέρεται η ιεραρχική διαχείριση της συνολικής βαθμολογίας ενός μαθήματος διότι αφορά αποκλειστικά την διαδικασία της βαθμολόγησης των γραπτών εξετάσεων.

Lukyamuzi, A. (2020). A Flexible Student Assessment System for Continuous Performance Monitoring. Busitema University, Uganda.

Στη συγκεκριμένη εργασία προτείνεται άλλο ένα σύστημα αξιολόγησης, το οποίο έχει τη δυνατότητα να αποθηκεύει τις πληροφορίες των σπουδαστών καθώς και τους συνολικούς βαθμούς απόδοσης ανά μάθημα και βάση αυτών να δημιουργεί καρτέλες για τον κάθε σπουδαστή μετά το τέλος μιας καθορισμένης περιόδου. Επίσης, το σύστημα αυτό μπορεί να δίνει στατιστικά στοιχεία σχετικά με την απόδοση στα μαθήματα των σπουδαστών των εκπαιδευτικών ιδρυμάτων σε σχέση με το χρόνο και το γενικό πλήθος σπουδαστών σε σύγκριση με άλλα εκπαιδευτικά ιδρύματα που χρησιμοποιούν το συγκεκριμένο σύστημα. Ακόμη, λόγω των διαφορετικών επιλογών βαθμολόγησης μαθημάτων από διάφορα εκπαιδευτικά ιδρύματα, το συγκεκριμένο σύστημα διαθέτει επιπλέον επίπεδα παραμετροποίησης, προσπαθώντας να καλύψει περισσότερες επιλογές από περισσότερα εκπαιδευτικά ιδρύματα, για την βαθμολόγηση των μαθημάτων ανάλογα με τις απαιτήσεις του κάθε ιδρύματος. Το επίπεδο παραμετροποίησης, όμως, είναι μικρότερο σε σχέση με το σύστημα της παρούσας εργασίας.

2.2. Σύγκριση με το προτεινόμενο σύστημα

Τα περισσότερα από τα συστήματα που αναλύθηκαν στην προηγούμενη ενότητα έχουν τα παρακάτω κοινά στοιχεία: προσφέρουν τη δυνατότητα της διαχείρισης των διδασκαλιών και των σπουδαστών καθώς και της βαθμολόγησης των σπουδαστών. Ορισμένα παρέχουν παραπάνω δυνατότητες, όπως υποστήριξη για συζητήσεις (moodle, cluster), ενώ κάποια έχουν ως κύριο σκοπό κυρίως τη βαθμολόγηση γραπτών εξετάσεων (Blended e-assessment [1]).

Το σύστημα της παρούσας διπλωματικής παρέχει και αυτό με τη σειρά του την δυνατότητα διαχείρισης των μαθημάτων, προσθήκης διδασκαλιών μαθημάτων καθώς και βαθμολόγησης των σπουδαστών ανά διδασκαλία μαθήματος. Επίσης, παρέχει τη δυνατότητα της παραμετροποίησης της βαθμολογίας του μαθήματος, ενώ η βαθμολογία υπολογίζεται αυτοματοποιημένα με ιεραρχικό τρόπο ανάλογα με την προσδιορισμένη ιεραρχία μεθόδων αξιολόγησης του μαθήματος. Η προσθήκη μαθημάτων μπορεί να γίνεται μεμονωμένα ή με μαζικό τρόπο μέσω της μεταφόρτωσης κατάλληλων αρχείων με αποτέλεσμα την σημαντική μείωση του χρόνου της εισαγωγής των οντοτήτων αυτού του είδους. Η προσθήκη σπουδαστών στις διδασκαλίες πραγματοποιείται με την μεταφόρτωση των κατάλληλων αρχείων στις διδασκαλίες των μαθημάτων, ενώ οι διδασκαλίες των μαθημάτων εισάγονται από τον χρήστη για κάθε μάθημα.

Κατά την γνώμη μας, το προτεινόμενο σύστημα παρέχει δύο σημαντικά πλεονεκτήματα σε σχέση με τα άλλα συστήματα που αναλύθηκαν προηγουμένως. Το πρώτο πλεονέκτημα αφορά την παραμετροποίηση της αξιολόγησης λόγω της μεγάλης ευελιξίας που παρέχει, καλύπτοντας όλα τα

δυνατά σενάρια αξιολόγησης που μπορεί να συντελούνται σε οποιοδήποτε τμήμα οποιουδήποτε ιδρύματος. Ειδικότερα, ο διδάσκων έχει την ευχέρεια να δημιουργήσει ένα δένδρο ή ιεραρχία αξιολόγησης που μπορεί να καλύπτει και το θεωρητικό και το εργαστηριακό μέρος ενός μαθήματος (εφόσον το τελευταίο υπάρχει) σε οποιοδήποτε βάθος, καλύπτοντας κάθε τρόπο μεμονωμένης αξιολόγησης που μπορεί να χρησιμοποιηθεί στο κάθε ένα από τα δύο αυτά μέρη. Το δεύτερο πλεονέκτημα αφορά τον αυτοματοποιημένο υπολογισμό της συνολικής βαθμολογίας ενός σπουδαστή με βάση τις επιμέρους βαθμολογίες του μόνο στα φύλλα της ιεραρχίας αξιολόγησης του μαθήματος. Με αυτό τον τρόπο, εξασφαλίζεται όχι μόνο η αυτοματοποίηση του υπολογισμού αλλά και η μικρότερη δυνατή αλλά επαρκής προσπάθεια από τον διδάσκων ως προς την παροχή των επιμέρους βαθμολογιών.

3. Ανάπτυξη Συστήματος

Στο κεφάλαιο αυτό θα γίνει η ανάλυση των φάσεων / σταδίων ανάπτυξης του συστήματος λογισμικού μας. Αρχικά, η πρώτη ενότητα προσδιορίζει ποιες είναι οι φάσεις αυτές. Έπειτα, οι επόμενες ενότητες αναλύουν την εργασία που επιτελέστηκε και τα αποτελέσματά της σε κάθε μία από τις φάσεις αυτές, αντίστοιχα.

3.1. Διαδικασία Ανάπτυξης Συστήματος

Όπως όλα τα βιομηχανικά προϊόντα έτσι και το λογισμικό ως προϊόν έχει τον κύκλο ζωής του, ο οποίος διέπεται από την διαδικασία ανάπτυξής του.

Η διαδικασία ανάπτυξης ενός συστήματος λογισμικού, ανεξαρτήτως δημιουργού, περνάει από αρκετά στάδια: από την σύλληψη της ιδέας (ως προς την βασική του λειτουργικότητα) και των απαιτήσεων του λογισμικού έως την απόσυρση του, όπου συνήθως αντικαθίσταται από νεότερο λογισμικό που εκμεταλλεύεται πληρέστερα τις νεότερες τεχνολογικές εξελίξεις. Όλα αυτά τα στάδια απαρτίζουν το κύκλο ζωής λογισμικού. Μια χονδρική διάκριση των σταδίων/φάσεων αυτών είναι η εξής:

- **Προσδιορισμός απαιτήσεων.** Με την ανάλυση και τον προσδιορισμό των απαιτήσεων, ορίζουμε τι πρέπει να κάνει το σύστημα αλλά όχι πως θα το κάνει. Επίσης, καλύπτουμε και μη λειτουργικές απαιτήσεις που μπορούν να αφορούν, πχ., την αναμενόμενη απόδοση του συστήματος (λογισμικού).
- **Σχεδίαση.** Η σχεδίαση πραγματοποιείται με γνώμονα τις απαιτήσεις που εξάχθηκαν. Αφορά τον προσδιορισμό της δομής/αρχιτεκτονικής του συστήματος καθώς και την μοντελοποίηση διαφόρων πλευρών του (στατικών ή δυναμικών, δομής ή συμπεριφοράς) με την μορφή ορισμένων διαγραμμάτων (πχ. διάγραμμα αρχιτεκτονικής ή διάγραμμα καταστάσεων).
- **Υλοποίηση.** Είναι η δραστηριότητα που περιλαμβάνει τον προγραμματισμό/υλοποίηση του συστήματος με οδηγό την σχεδιάσή του. Επομένως, στο στάδιο αυτό παράγεται το τελικό προϊόν του λογισμικού.
- **Επικύρωση.** Η δραστηριότητα που συντελείται σε αυτό το στάδιο συνοδεύει τον προγραμματισμό και εστιάζει στον έλεγχο αν η συμπεριφορά του λογισμικού και η λειτουργικότητά του είναι η αναμενόμενη, δηλαδή αυτή που έχει προδιαγραφεί. Κατά την διάρκεια αυτής της δραστηριότητας, ανιχνεύονται και διορθώνονται τυχόν υπάρχοντα λάθη καθώς και ελέγχονται κάποια από τα ποιοτικά χαρακτηριστικά του λογισμικού, όπως η αποδοτικότητα και η αξιοπιστία.
- **Διατήρηση.** Η φάση αυτή περιλαμβάνει την έναρξη της λειτουργίας του λογισμικού από τους πελάτες καθώς και τη συντήρηση και αναβάθμισή του, εφόσον υπάρξει κάποιο θέμα κατά τη διάρκεια της χρήσης του ή κάποια μεταβολή στις απαιτήσεις του.

Στις επόμενες ενότητες αναλύουμε τις εργασίες που έγιναν σε κάθε μία από τις προαναφερόμενες

φάσεις και τα αποτελέσματά της εκτός από τις φάσεις της επικύρωσης και διατήρησης. Για την φάση της επικύρωσης, πραγματοποιήθηκαν εκτενείς χειρωνακτικές δοκιμές του λογισμικού υπό ανάπτυξη που οδήγησαν στον εντοπισμό διαφόρων λαθών και την επιδιόρθωσή τους. Η τελευταία φάση ουσιαστικά δεν εκτελέστηκε λόγω της φύσης του έργου του λογισμικού αυτού (πρωτότυπο με την μορφή ανοικτού κώδικα). Όμως, έχει ήδη λάβει την μορφή ενός Github έργου⁴, το οποίο προσφέρεται στην παγκόσμια κοινότητα των προγραμματιστών προς επέκταση και βελτίωση ώστε να πάρει έπειτα τον δρόμο προς χρήση σε εκπαιδευτικά και ακαδημαϊκά ιδρύματα ως ένα ώριμο και ποιοτικό προϊόν λογισμικού.

3.2. Απαιτήσεις συστήματος

Το πρώτο βασικό βήμα για την ανάπτυξη ενός λογισμικού είναι η καταγραφή των απαιτήσεων του πελάτη ή εν γένει των ενδιαφερόμενων μερών (stakeholders). Αυτό το πρόβλημα λύνει η Μηχανική Απαιτήσεων (Requirements Engineering). Η Μηχανική Απαιτήσεων είναι η διαδικασία που εγκαθιδρύει τις υπηρεσίες που απαιτεί ο πελάτης από ένα σύστημα καθώς και τους περιορισμούς υπό τους οποίους το σύστημα αυτό θα λειτουργεί [3]. Οι απαιτήσεις συστήματος είναι στην ουσία οι (λεπτομερείς) περιγραφές των υπηρεσιών και των περιορισμών που παράγονται από την διαδικασία της Μηχανικής Απαιτήσεων.

Για την καλύτερη περιγραφή αυτών των απαιτήσεων, τις διακρίνουμε σε λειτουργικές απαιτήσεις και μη λειτουργικές απαιτήσεις.

Οι λειτουργικές απαιτήσεις περιγράφουν τις υπηρεσίες που το σύστημα πρέπει να παρέχει καθώς και τις αντιδράσεις του συστήματος σε συγκεκριμένες εισόδους ή/και σε συγκεκριμένες καταστάσεις / γεγονότα.

Οι μη λειτουργικές απαιτήσεις περιγράφουν κάθε απαίτηση που δεν ανήκει στις λειτουργικές απαιτήσεις. Πιο συγκεκριμένα, είναι οι περιορισμοί στην ποιότητα των υπηρεσιών ή στις λειτουργίες που προσφέρονται από το σύστημα, όπως περιορισμοί χρόνου, περιορισμοί χώρου, ασφάλειας κλπ. [3]

Παρακάτω παραθέτουμε τις απαιτήσεις που τέθηκαν για το σύστημα λογισμικού της διπλωματικής εργασίας, κατηγοριοποιημένες με βάση τις δύο προαναφερόμενες κατηγορίες και ομαδοποιημένες ανάλογα την οντότητα που διαχειριζόμαστε (ειδικά για την κατηγορία των λειτουργικών απαιτήσεων). Οι οντότητες αυτές είναι τα μαθήματα, οι διδάσκοντες/χρήστες, οι διδασκαλίες και οι σπουδαστές.

3.2.1 Λειτουργικές απαιτήσεις

- Χρήστες:

⁴ <https://github.com/ilias24k/grade-managment-system>

1. Το σύστημα πρέπει να υποστηρίζει τη δημιουργία λογαριασμού χρήστη – διδάσκων
 2. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή να βλέπει τη λίστα με τους διδάσκοντες-χρήστες
 3. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή την διαγραφή χρηστών (διδασκόντων) από το σύστημα.
 4. Το σύστημα πρέπει να επιτρέπει την τροποποίηση των στοιχείων των χρηστών.
- Μαθήματα
 1. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή την εισαγωγή μαθημάτων, μέσω αρχείων JSON, με κάτω όρια και βάρη για κάθε μέρος βαθμολογίας ενός μαθήματος. Τα κάτω όρια επιτρέπουν να προσδιοριστεί αν ο βαθμός ενός φοιτητή σε ένα μέρος του μαθήματος είναι προβιβάσιμος ή όχι. Ενώ τα βάρη αποτελούν την σχετική βαρύτητα που έχει ένα μέρος βαθμολόγησης σε σχέση με τα άλλα για ένα μάθημα.
 2. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή την ανάθεση μαθημάτων σε εγγεγραμμένους στο σύστημα χρήστες-διδάσκοντες.
 3. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή να βλέπει την λίστα των μαθημάτων και σε ποιους διδάσκοντες αυτά έχουν ανατεθεί.
 4. Το σύστημα πρέπει να παρέχει τη δυνατότητα στον διδάσκων να καθορίσει για ένα μάθημα αν θα έχει δύο βασικά μέρη (θεωρία & εργαστήριο) ή μόνο ένα (θεωρία) μέρος καθώς και τους τρόπους αξιολόγησης φοιτητών ανά μέρος με την αντίστοιχη βαρύτητά τους με έναν ιεραρχικό τρόπο.
 5. Το σύστημα πρέπει να υποστηρίζει την μεταφόρτωση μαθημάτων από αρχεία τύπου JSON από τον διαχειριστή του συστήματος. Σε περίπτωση που προκύψει κάποιο σφάλμα, ακυρώνεται η εκφόρτωση και εμφανίζεται μήνυμα λάθους.
 - Διδασκαλίες
 1. Το σύστημα πρέπει να επιτρέπει τη δημιουργία διδασκαλιών για το κάθε μάθημα (χειρωνακτική & μαζική μέσω αρχείου προς μεταφόρτωση) που έχει αναθέσει στον διδάσκων ο διαχειριστής του συστήματος - με χρονολογία και αριθμό εξαμήνου.
 2. Το σύστημα πρέπει να επιτρέπει στον διδάσκων να βλέπει και να επεξεργάζεται τα στοιχεία μιας εκφορτωμένης διδασκαλίας, εφόσον ο διδάσκων έχει ανατεθεί σε αυτή, όπως αυτά παρέχονται από το αντίστοιχο αρχείο εκφόρτωσης (τα βάρη της δομής του μαθήματος, τα κάτω όρια και τη χρονική διάρκεια που θα κρατιέται ένας προβιβάσιμος βαθμός)
 3. Το σύστημα πρέπει να παρέχει στον διδάσκων τη δυνατότητα διαχείρισης (διάβασμα & ανανέωση) της όλης ιεραρχίας αξιολόγησης/βαθμολόγησης.
 4. Το σύστημα πρέπει να παρέχει τη δυνατότητα στον διδάσκων να ανανεώνει διδασκαλίες για ένα μάθημα το οποίο του έχει ανατεθεί.
 5. Το σύστημα πρέπει να επιτρέπει μόνο στον διαχειριστή να αφαιρεί διδασκαλίες

6. Το σύστημα πρέπει να επιτρέπει το πάγωμα μιας διδασκαλίας χειροκίνητα (από επιλογή του αντίστοιχου κουμπιού στο γραφικό περιβάλλον), από τον διδάσκων του μαθήματος ή τον διαχειριστή.
- Σπουδαστές
 1. Το σύστημα πρέπει να επιτρέπει στον διδάσκων την μαζική εισαγωγή σπουδαστών σε συγκεκριμένη διδασκαλία μαθήματος, από αρχείο τύπου EXCEL.
 2. Το σύστημα πρέπει να επιτρέπει στον διδάσκων να βλέπει όλες τις πληροφορίες (αριθμό μητρώου, ονοματεπώνυμο, email) για τους σπουδαστές που είναι εγγεγραμμένοι σε ένα μάθημα ή διδασκαλία.
 3. Το σύστημα πρέπει να επιτρέπει στον διδάσκων να δει το σύνολο των σπουδαστών ανά μάθημα και ανά διδασκαλία. Ειδικότερα, θα πρέπει να εμφανίζονται όλοι οι φοιτητές που έχουν εγγραφεί ποτέ στο μάθημα (χωρίς διπλότυπα), μαζί με τις χρονολογίες των διδασκαλιών που έχουν εγγραφεί.
 4. Το σύστημα θα πρέπει να υποστηρίζει τη δυνατότητα υπολογισμού των τελικών, συνολικών βαθμών του κάθε σπουδαστή με βάση τα αντίστοιχα βάρη στην διδασκαλία του μαθήματος και το βαθμό επιδόσεων του σπουδαστή στα μέρη της διδασκαλίας του μαθήματος που έχει βαθμολογηθεί (στα φύλλα της ιεραρχίας αξιολόγησης της διδασκαλίας).
 5. Το σύστημα πρέπει να υποστηρίζει την δυνατότητα της τροποποίησης των βαθμολογιών των φοιτητών από τον διδάσκων.
 6. Το σύστημα πρέπει να διατηρεί ιστορικό με τις βαθμολογίες των σπουδαστών ανά διδασκαλία. Το ιστορικό θα εμφανίζεται ύστερα από αίτηση του διδάσκοντος για οποιοδήποτε σπουδαστή διδασκαλίας του.
 7. Το σύστημα πρέπει να παρέχει τη δυνατότητα στο διδάσκων να εξάγει τις τυπικές και τις αναλυτικές βαθμολογίες όλων των σπουδαστών σε μορφή EXCEL.

3.2.2 Μη λειτουργικές απαιτήσεις

1. Αξιοπιστία

- 1.1. Το σύστημα θα πρέπει να έχει τη δυνατότητα να επανέλθει σε πλήρη λειτουργία μετά από κάποια βλάβη μέσα σε συγκεκριμένο χρόνο που εξαρτάται από τον διαχειριστή του συστήματος.
- 1.2. Το σύστημα δεν πρέπει να εμφανίζει κρίσιμα σφάλματα.

2. Απόδοση

2.1. Η ταχύτητα απάντησης του συστήματος στις εντολές του χρήστη πρέπει να είναι σχεδόν άμεση και ικανοποιητική για τους χρήστες.

3. Ευχρηστία

- 3.1. Το σύστημα θα πρέπει να είναι ιδιαίτερα εύκολο στην εκμάθηση και στη χρήση.
- 3.2. Θα πρέπει να παρέχονται οδηγίες για τη σειρά των ενεργειών που μπορεί να εκτελέσει ο χρήστης προς επιτέλεση των βασικών λειτουργιών που προσφέρονται από το σύστημα.
- 3.3. Θα πρέπει να εμφανίζονται κατάλληλα μηνύματα λάθους στους χρήστες όταν αυτοί διαπράττουν κάποιο σχετικό σφάλμα, πχ. εισόδου δεδομένων, ώστε να το διορθώσουν άμεσα.

4. Φορητότητα και Συμβατότητα

- 4.1. Το σύστημα θα πρέπει να είναι βασισμένο στον Παγκόσμιο Ιστό και είναι διαθέσιμο σε όποια συσκευή έχει τους απαραίτητους πόρους για να φιλοξενήσει έναν φυλλομετρητή.
- 4.2. Η διαχείριση του συστήματος μπορεί να πραγματοποιείται απομακρυσμένα.
- 4.3. Θα πρέπει να υλοποιηθεί υποκείμενη REST υπηρεσία που να υλοποιεί τις βασικές λειτουργίες του συστήματος και να έχει το υψηλότερο επίπεδο ωριμότητας REST.

5. Ασφάλεια

- 5.1. Το σύστημα πρέπει να κωδικοποιεί τους κωδικούς του χρήστη στη βάση δεδομένων για μεγαλύτερη ασφάλεια
- 5.2. Το σύστημα παρέχει να παρέχει ταυτοποίηση χρηστών και εξουσιοδότηση βασισμένη σε ρόλους (όπου θα πρέπει να υποστηριχθούν οι ρόλοι του διδάσκοντος και του διαχειριστή).

3.3. Σχεδίαση Συστήματος

Ο σχεδιασμός του λογισμικού είναι μια πολύπλοκη διαδικασία που μπορεί να χρειαστεί κάποια ενημέρωση (ως προς τα σχεδιαστικά προϊόντα που παράγει) εφόσον εμφανιστούν ορισμένα θέματα κατά την ανάπτυξη/υλοποίησή του. Για αυτό και θα πρέπει να γίνεται με προσοχή, κοιτάζοντας εμπεριστατωμένα όλες τις σχετικές πλευρές. Διότι οι σχεδιαστικές αλλαγές ή λάθη έχουν μεγαλύτερο κόστος σε σχέση με προγραμματιστικά λάθη.

Σε αυτό το κεφάλαιο θα παρουσιαστούν ορισμένα σχεδιαστικά μοντέλα: αρχιτεκτονικής, περιβάλλοντος, περιπτώσεων χρήσης, δραστηριοτήτων και ακολουθίας. Σε αυτά τα διαγράμματα αποτυπώνονται οι σχεδιαστικές αποφάσεις που πάρθηκαν καθώς και η συμπεριφορά του συστήματος.

3.3.1. Αρχιτεκτονική Συστήματος

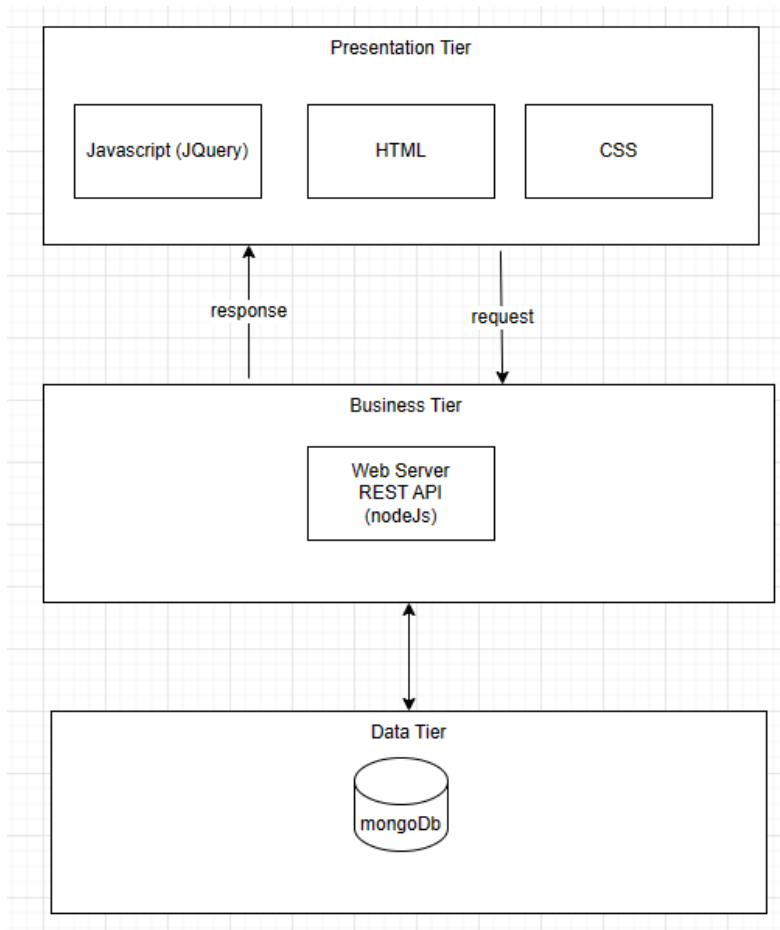
Το σύστημα που περιγράφεται στην διπλωματική αυτή σχεδιάστηκε με βάση την αρχιτεκτονική τριών επιπέδων, η οποία εφαρμόζεται κατά κόρον στις εφαρμογές ιστού (Εικόνα 1). Η συγκεκριμένη σχεδίαση προσφέρει τη δυνατότητα στον προγραμματιστή να επεξεργάζεται, να προσθέτει ή να αφαιρεί κάποιο κομμάτι του λογισμικού χωρίς να πρέπει να αλλάξει η υπάρχουσα δομή του συστήματος. Η αρχιτεκτονική τριών επιπέδων αποτελείται από τα επίπεδα παρουσίασης, επιχειρησιακής λογικής και δεδομένων.

Το επίπεδο παρουσίασης (Presentation Tier) είναι το επίπεδο όπου το λογισμικό αλληλοεπιδρά με τον χρήστη, συλλέγοντας δεδομένα από αυτόν και παρουσιάζοντας την πληροφορία διαφόρων σχετικών οντοτήτων. Πρόκειται για το επίπεδο στο οποίο έχουν πρόσβαση οι χρήστες του συστήματος, το οποίο περιλαμβάνει το σύνολο των ιστοσελίδων του συστήματος με το γραφικό τους περιβάλλον. Οι γραφικές ιστοσελίδες συνήθως αναπτύσσονται χρησιμοποιώντας τεχνολογίες ιστού, όπως HTML, CSS και Javascript.

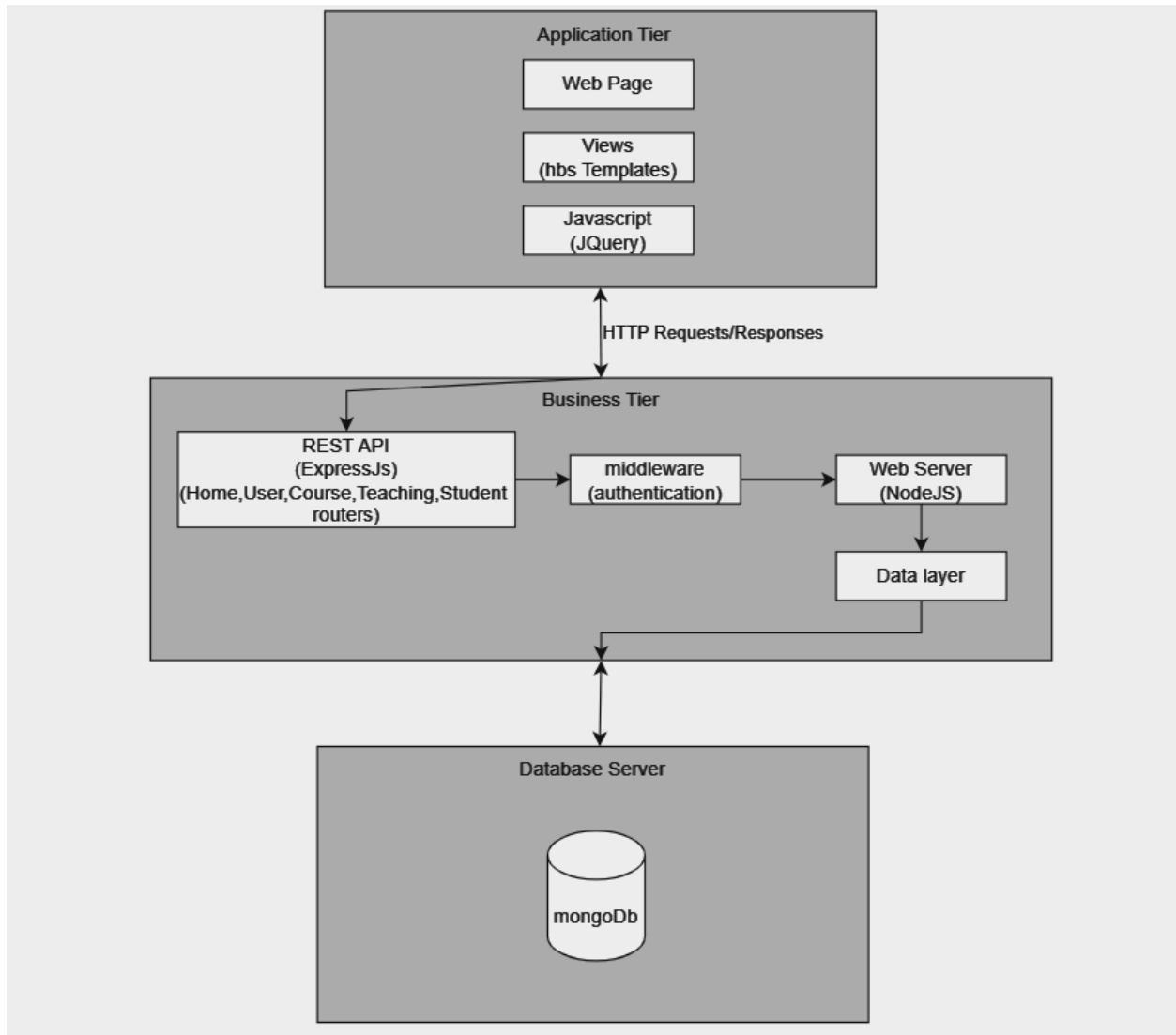
Το επίπεδο επιχειρησιακής λογικής (Business Logic Tier), είναι η καρδιά του συστήματος. Το επίπεδο αυτό έχει τη δυνατότητα να εισάγει, να διαγράφει και να τροποποιεί τα δεδομένα των οντοτήτων του συστήματος (με την βοήθεια του υποκείμενου επιπέδου των δεδομένων) ανάλογα με την αιτούμενη λειτουργία καθώς και να υλοποιεί τις βασικές λειτουργίες του συστήματος. Αναπτύσσεται συνήθως χρησιμοποιώντας προγραμματιστικές γλώσσες, όπως Python, Java, PHP και Javascript.

Το επίπεδο δεδομένων (Data Tier) περιλαμβάνει βάσεις ή αποθετήρια δεδομένων. Οι βάσεις μπορεί να είναι σχεσιακές (SQL), όπως MySQL & Oracle, ή μη σχεσιακές, όπως η MongoDB. Ουσιαστικά, τοποθετούνται και λειτουργούν σε αυτό το επίπεδο Συστήματα Διαχείρισης Βάσεων Δεδομένων που διαχειρίζονται τις βάσεις αυτές. Τα συστήματα αυτά διαθέτουν διεπαφή ή διεπαφές που επιτρέπουν την διαχείριση των δεδομένων στις βάσεις τους από το επίπεδο της επιχειρησιακής λογικής.

Τα προαναφερόμενα επίπεδα φαίνονται στα δύο διαγράμματα αρχιτεκτονικής που ακολουθούν. Το πρώτο επιδεικνύει την γενική αρχιτεκτονική των εφαρμογών ιστού (και άρα και της δικής μας με πολύ λίγες λεπτομέρειες) ενώ το δεύτερο χτίζει πάνω στο πρώτο για να παρουσιάσει την λεπτομερή αρχιτεκτονική του συστήματός μας.



Εικόνα 1 - Διάγραμμα Αρχιτεκτονικής του Συστήματος



Εικόνα 2 - Εξειδικευμένο Διάγραμμα Αρχιτεκτονικής του Συστήματος

3.3.2. Σχεδίαση διεπαφής χρήστη

Η σχεδίαση και υλοποίηση της διεπαφής του χρήστη (UI) βασίζεται στην ακόλουθη λογική: Ο διαχειριστής συνδέεται στο σύστημα και ανεβάζει τα μαθήματα, τα οποία ανατίθενται σε έναν διδάσκων. Στη συνέχεια, με την είσοδο του διδάσκων – χρήστη στο λογισμικό θα του εμφανίζεται μια σελίδα (Home) με οδηγίες βήμα προς βήμα για την ορθή λειτουργία του λογισμικού. Επίσης, θα παρουσιάζεται μενού με τις επιλογές Home, Course και Students.

Στη σελίδα Course, ο διδάσκων-χρήστης μπορεί να εισάγει τη λίστα των διδασκαλιών στα μαθήματα που του έχουν ανατεθεί. Επίσης, μπορεί, αφού έχουν εισαχθεί διδασκαλίες, να τις επεξεργαστεί καθώς και να προσθέσει τη λίστα με τους σπουδαστές που παρακολουθούν την κάθε διδασκαλία.

Επιπλέον, μπορεί να ορίσει τα μέρη και τα βάρη των διδασκαλιών, και στη συνέχεια να βαθμολογήσει τον κάθε σπουδαστή με βάση αυτά τα μέρη και βάρη.

Τέλος, ο διδάσκων μπορεί να εξάγει είτε την αναλυτική είτε την τυπική βαθμολογία για όλους τους σπουδαστές για μια διδασκαλία σε μορφή αρχείου excel.

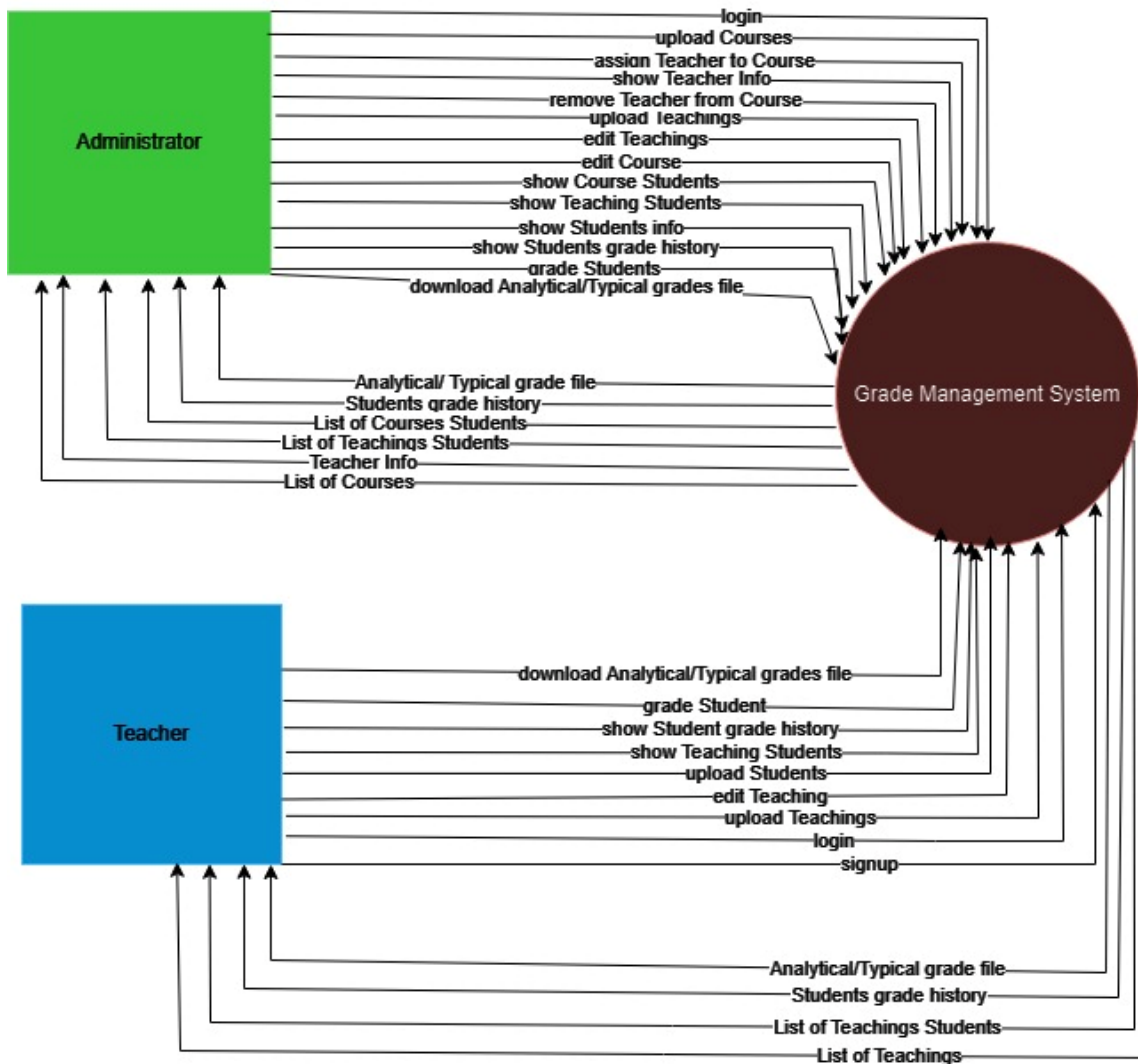
Στη σελίδα Students, ο διδάσκων- χρήστης μπορεί να δει τη λίστα με όλους τους σπουδαστές των διδασκαλιών ενός μαθήματος του καθώς και παράλληλα του δίνεται η δυνατότητα να δει τις βαθμολογίες των σπουδαστών ανά διδασκαλία εφόσον οι σπουδαστές έχουν βαθμολογηθεί σε αυτήν.

Τονίζουμε, επιπλέον, πως οι προαναφερόμενες σελίδες που επιτρέπουν την πραγματοποίηση συγκεκριμένων λειτουργιών του συστήματος εμφανίζονται και στον διαχειριστή του συστήματος, όντας ένας υπερ-χρήστης του συστήματος αυτού.

Παραθέτουμε και αναλύουμε συντόμως παρακάτω διάφορα σχεδιαστικά διαγράμματα που προσδιορίζουν διάφορες πτυχές της εφαρμογής ιστού που αναπτύχθηκε.

3.3.3. Διάγραμμα περιβάλλοντος

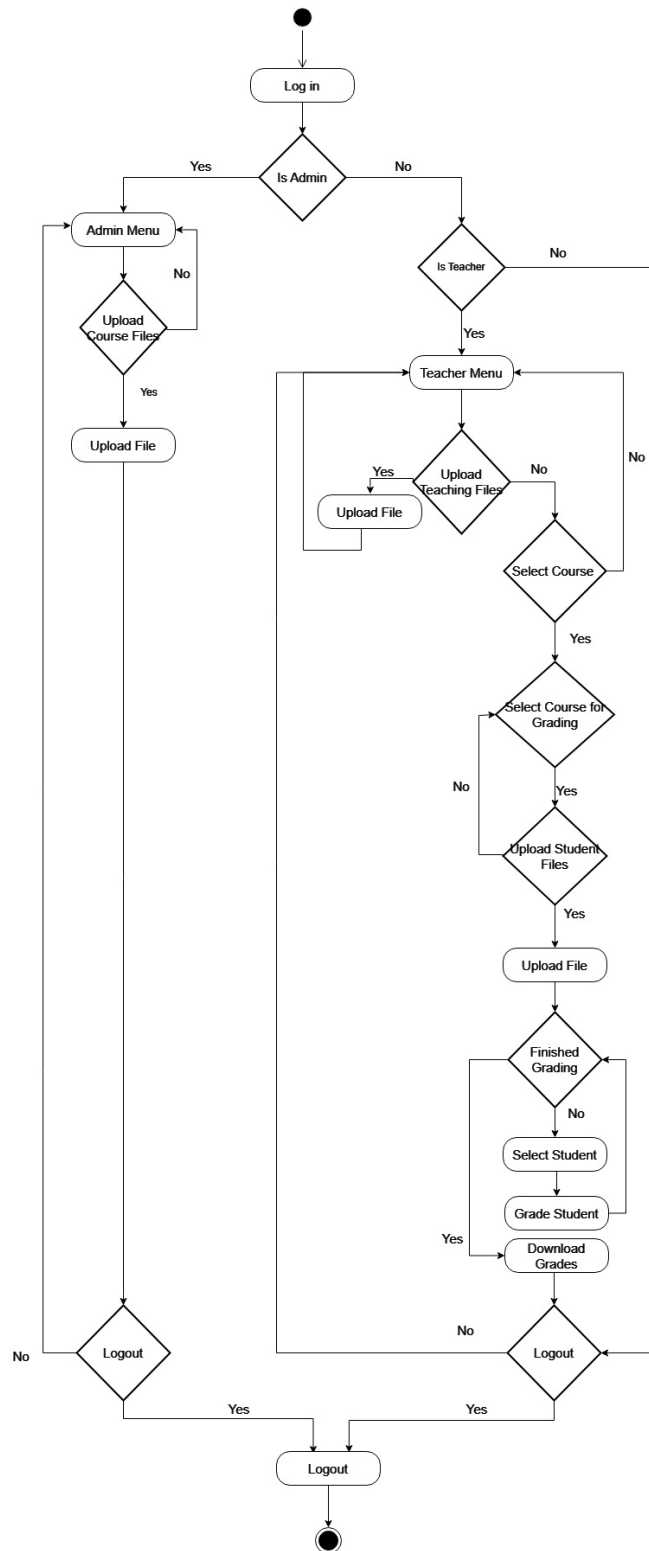
Ξεκινώντας από το διάγραμμα περιβάλλοντος παρατηρούμε τη ροή των δεδομένων / αλληλεπιδράσεις υψηλού επιπέδου μεταξύ του συστήματός μας και των βασικών του ειδών χρηστών / ρόλων. Συγκεκριμένα βλέπουμε όλες τις λειτουργίες που μπορεί να κάνουν ο διαχειριστής και ο διδάσκων – χρήστης.



Εικόνα 3 - Διάγραμμα Περιβάλλοντος

3.3.4. Διάγραμμα δραστηριοτήτων

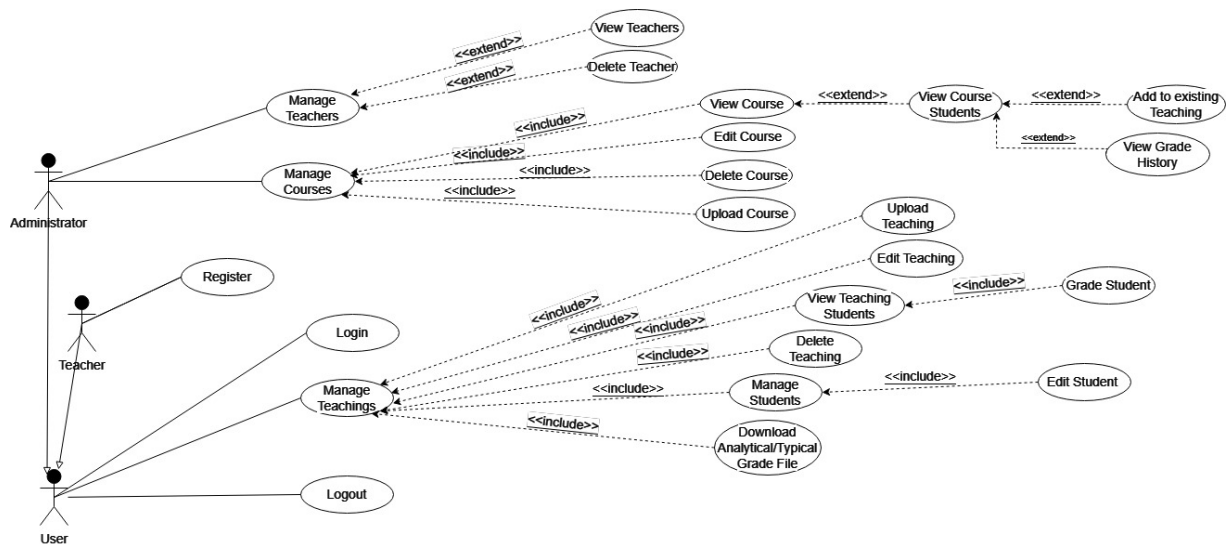
Παρακάτω βλέπουμε το διάγραμμα δραστηριοτήτων, στο οποίο περιγράφονται οι δραστηριότητες που μπορεί να ακολουθούνται από τη στιγμή της σύνδεσης ενός χρήστη (διδάσκοντος ή διαχειριστή) μέχρι και την αποσύνδεσή του.



Εικόνα 4 - Διάγραμμα Διαδικασιών

3.3.5. Διάγραμμα περιπτώσεων χρήσης

Στο επόμενο διάγραμμα, αυτό των περιπτώσεων χρήσης, παρατηρούμε τις βασικές λειτουργίες του λογισμικού, με την μορφή περιπτώσεων χρήσης, που μπορεί να καταναλώσουν ο διδάσκων και ο διαχειριστής του συστήματος. Στα σενάρια/περιπτώσεις χρήσης που φαίνονται στο διάγραμμα, οι συμμετέχοντες (διαχειριστής ή διδάσκων) εισέρχονται στο σύστημα. Επιπλέον, ο διαχειριστής αναθέτει τα μαθήματα στον διδάσκων, και εκείνος με τη σειρά του μπορεί να εκφορτώνει τις διδασκαλίες και τους σπουδαστές. Επίσης, ο διαχειριστής μπορεί να δει τις πληροφορίες του διδάσκων, ενώ ο διδάσκων προχωράει στη βαθμολόγηση των σπουδαστών και στην μεταφόρτωση των αρχείων με τις βαθμολογίες των σπουδαστών.

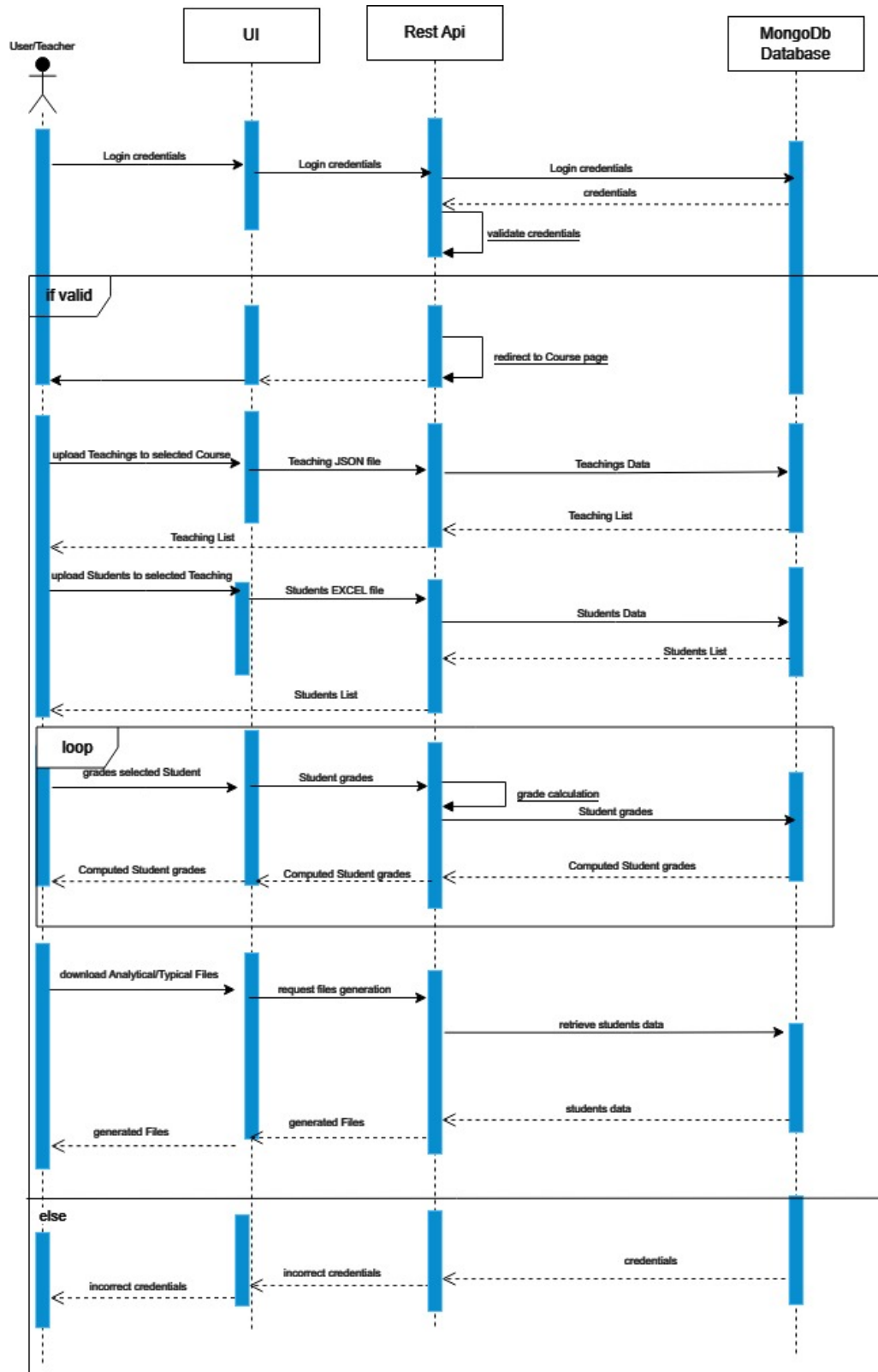


Εικόνα 5 - Διάγραμμα Περιπτώσεων Χρήσης

3.3.6. Διάγραμμα ακολουθίας

Στο διάγραμμα ακολουθίας παρατηρούμε ένα σενάριο χρήσης, το οποίο καλύπτει πολλές περιπτώσεις χρήσης. Σε αυτό προδιαγράφεται πως ο χρήστης αρχικά συνδέεται στο σύστημα, ανεβάζει το αρχείο με τις διδασκαλίες στα μαθήματα που του έχουν ανατεθεί από τον διαχειριστή. Στη συνέχεια ανεβάζει στην επιλεγμένη διδασκαλία τους σπουδαστές και συνεχίζει στην βαθμολόγηση τους. Τέλος, εκφορτώνει την τυπική ή αναλυτική βαθμολογία της διδασκαλίας.

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών



Εικόνα 6 - Διάγραμμα Ακολουθίας

3.4. Υλοποίηση Συστήματος

Σε αυτό την ενότητα παρουσιάζονται οι λεπτομέρειες υλοποίησης καθώς και οι τεχνολογίες και τα εργαλεία πάνω στα οποία βασίστηκε η υλοποίηση της εφαρμογής, όπως επίσης και οι λόγοι για τους οποίους επιλέχθηκαν (αυτά τα εργαλεία & τεχνολογίες).

Η ανάπτυξη του συστήματος χωρίστηκε σε δύο μέρη: το μετωπιαίο (front-end) και το νωτιαίο (back-end). Το front-end αφορά το γραφικό περιβάλλον μιας ιστοσελίδας/ιστοτόπου και όλα τα γραφικά στοιχεία με τα οποία αυτό μπορεί να αλληλοεπιδράσει με τους χρήστες, όπως κουμπιά, μενού, φόρμες εισαγωγής κειμένου κλπ. Η ανάπτυξη του front-end έγινε κυρίως με τη χρήση Javascript και τη γλώσσα προτύπων handlebars σε συνδυασμό με HTML και CSS. Επίσης, χρησιμοποιήθηκε εκτενώς η βιβλιοθήκη JQuery της Javascript ενώ χρησιμοποιήθηκε η τεχνολογία Ajax, όπου αυτό ήταν εφικτό και κατάλληλο, για την άντληση δυναμικού περιεχομένου στην μεριά του πελάτη.

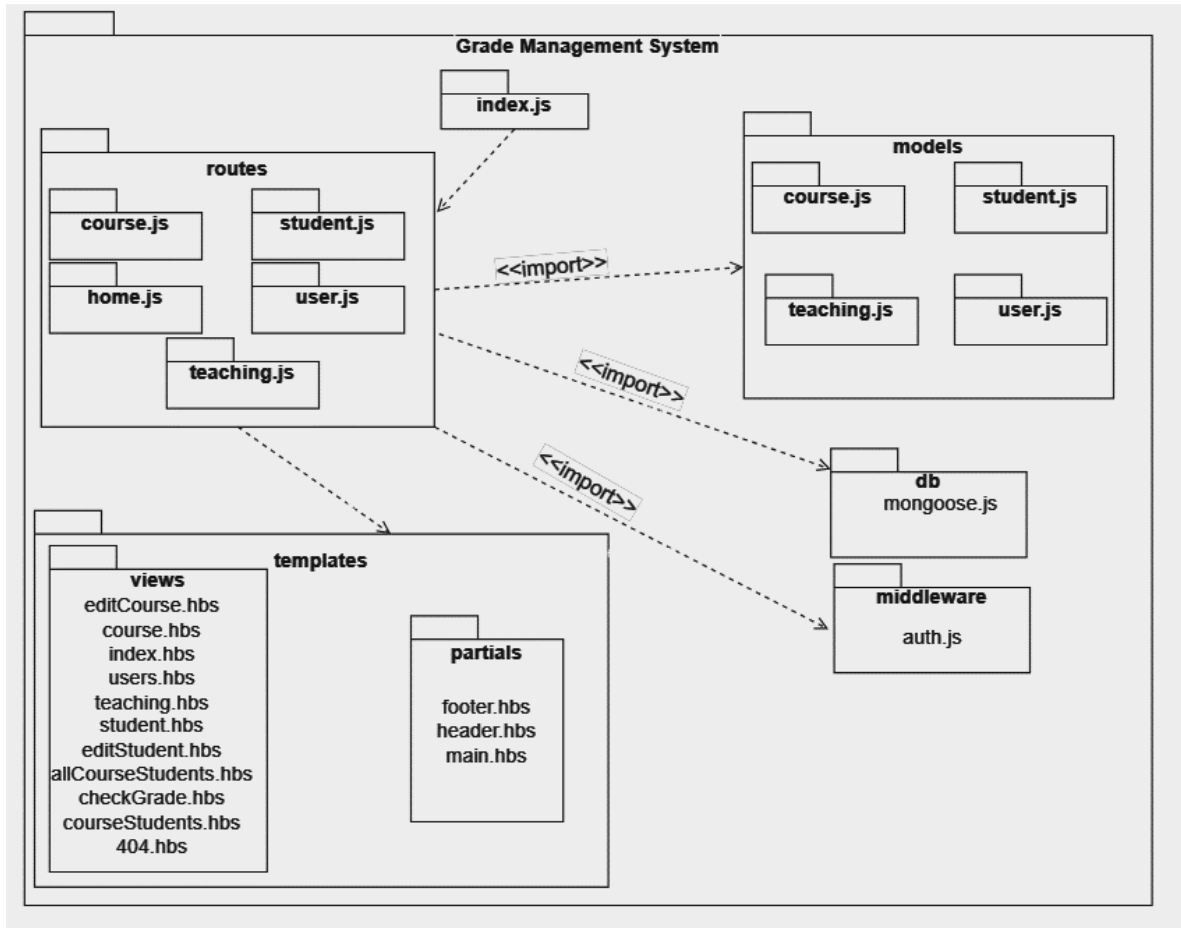
Όσον αφορά το back-end, αυτό αποτελεί το μέσο με το οποίο πραγματοποιείται η επικοινωνία με τη βάση δεδομένων ενώ υλοποιεί τις βασικές λειτουργίες του συστήματος. Η ανάπτυξη του back-end πραγματοποιήθηκε: (α) με την γλώσσα Javascript χρησιμοποιώντας την τεχνολογία Node.js και ειδικότερα το πλαίσιο της Express για την ανάπτυξη REST υπηρεσιών και (β) το Mongoose για την διασύνδεση και αλληλεπίδραση με την μη σχεσιακή βάση δεδομένων MongoDB.

Τα εργαλεία και περιβάλλοντα που χρησιμοποιήθηκαν για την ανάπτυξη του (όλου) λογισμικού καθώς και για τη δραστηριότητα/φάση της επικύρωσης/δοκιμής είναι τα Postman (γραφική εφαρμογή για έλεγχο λειτουργικότητας του API), Robo3T (γραφικό περιβάλλον διαχείρισης βάσεων δεδομένων), & Visual Studio Code (περιβάλλον ανάπτυξης λογισμικού).

Ο κώδικας της εφαρμογής ιστού που υλοποιήθηκε είναι διαθέσιμος στο GitHub στον ακόλουθο σύνδεσμο: <https://github.com/ilias24k/grade-managment-system>

3.4.1. Διάγραμμα πακέτων

Στο διάγραμμα πακέτων παρατηρούμε τα πακέτα των αρχείων του συστήματος, καθώς και τις όποιες εξαρτήσεις έχουν μεταξύ τους.



Εικόνα 7 - Διάγραμμα Πακέτων

3.4.2. Τεχνολογίες & Εργαλεία Back-end Συστήματος

Ξεκινώντας από την πλευρά του εξυπηρετητή (server), επιλέχθηκε το περιβάλλον εκτέλεσης Node.js βασισμένο στην Javascript λόγω της σταθερότητας, ασφάλειας και της μεγάλης κοινότητας που το χρησιμοποιεί και προσφέρει στην συντήρησή του. Το τελευταίο είναι ιδιαίτερα σημαντικό διότι επιδεικνύει την σημαντικότητα και δημοτικότητα (popularity) του περιβάλλοντος αυτού, το οποίο εξελίσσεται με μεγάλη ταχύτητα. Παράλληλα, αφού και στο front-end τμήμα του λογισμικού χρησιμοποιείται η Javascript, η επιλογή του Node.js ήταν ιδανική αφού δεν απαιτείται η εκμάθηση/γνώση κάποιας έξτρα γλώσσας για την ανάπτυξη του συστήματος – αυτό επιταχύνει την ανάπτυξη της εφαρμογής. Άλλος ένας ιδιαίτερα σημαντικός λόγος επιλογής του Node.js είναι ότι λειτουργεί ασύγχρονα. Αυτό σημαίνει ότι αντί να δημιουργεί ένα νέο νήμα (thread) για κάθε πελάτη, προσφέρει ένα μοντέλο όπου δημιουργεί ένα συμβάν για κάθε αίτημα. Αυτά τα αιτήματα επεξεργάζονται ανεξάρτητα και χωρίς αποκλεισμό, επιτρέποντας έτσι την ομαλή λειτουργία του εξυπηρετητή χωρίς τον αναγκαστικό χρόνο αναμονής των πελατών για την ολοκλήρωση κάθε αιτήματός τους. Σε περίπτωση επιλογής συγχρονισμένης (synchronous) πλατφόρμας, η ύπαρξη μεγάλου αριθμού από ταυτόχρονους πελάτες και αιτήσεις θα είχε ως αποτέλεσμα την

υπερφόρτωση του εξυπηρετητή και άρα την περιορισμένη διαθεσιμότητά του. Επιπλέον, θα καθιστούσε επιτακτική την ανάγκη για την παροχή επιπλέον πόρων στον εξυπηρετητή για να ανταποκριθεί στον κυμαινόμενο φόρτο εργασίας.

Άλλο ένα πλεονέκτημα του Node.js είναι πως προσφέρει διάφορα πλαίσια ανάπτυξης εφαρμογών ιστού και APIs, όπως είναι το πλαίσιο Express.js που χρησιμοποιήθηκε. Το πλαίσιο αυτό επιτρέπει την εύκολη δημιουργία του back-end περιβάλλοντος για την επικοινωνία πελάτη-εξυπηρετητή με τη μορφή ενός RESTful API ενώ παράλληλα διαθέτει πληθώρα βιβλιοθηκών για τη διευκόλυνση της συνολικής υλοποίησης, όπως βιβλιοθήκες για την επικοινωνία με τη βάση δεδομένων. Οπότε, με αυτό το πλαίσιο δημιουργήθηκαν οι διαδρομές (routes) και τα τελικά σημεία (endpoints) του συστήματος.

Οι διαδρομές σε ένα REST API είναι τα URI⁵ μονοπάτια που επιτρέπουν την πρόσβαση στα διαθέσιμα τελικά σημεία. Τα τελικά σημεία σε ένα REST API είναι τα URIs που αντιστοιχούν στις διάφορες οντότητες του συστήματος. Τα URIs αυτά σε συνδυασμό με την εκάστοτε HTTP μέθοδο προσδιορίζουν επακριβώς την λειτουργία διαχείρισης που πρέπει να πραγματοποιηθεί σε μια οντότητα του συστήματος.

Τα πακέτα/modules του Node.js χρησιμοποιούνται για να επεκτείνουν τις δυνατότητες του Node.js και να προσθέσουν έξτρα λειτουργικότητα σε ένα πρόγραμμα. Αυτά τα πακέτα παρέχουν διάφορες λειτουργίες και βιβλιοθήκες που διευκολύνουν την ανάπτυξη της εφαρμογής.

Άλλα πακέτα/modules (εκτός από το Express.js) που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής στο Node.js είναι τα παρακάτω:

- Το πακέτο **multer**. Το πακέτο αυτό επιτρέπει το ανέβασμα των αρχείων του χρήστη στον εξυπηρετητή μέσω φόρμας (απλής ή πολλαπλών μερών - form ή multipart/ form-data).
- Τα πακέτα **xlsx** και **exceljs**, τα οποία επιτρέπουν την δημιουργία, διάβασμα και επεξεργασία αρχείων τύπου Excel. Με αυτά τα πακέτα διαβάζονται οι σπουδαστές και δημιουργούνται τα τελικά αρχεία με τις αναλυτικές και τυπικές βαθμολογίες των σπουδαστών.
- Το πακέτο **hbs**. Το **hbs** είναι μια βιβλιοθήκη που χρησιμοποιεί τη γλώσσα μοτίβων (template language) handlebars, η οποία κάνει πιο εύκολη την ενσωμάτωση των προσδιορισμών της στο πλαίσιο Express. Η γλώσσα handlebars διευκολύνει τη συγγραφή του περιεχομένου των ιστοσελίδων (HTML) για το κομμάτι του front-end.
- Το πακέτο **nodemon**. Το συγκεκριμένο πακέτο ήταν ιδιαίτερα χρήσιμο αφού βοηθάει στην ανάπτυξη του λογισμικού, παρακολουθώντας συνεχώς τυχόν αλλαγές που συμβαίνουν

⁵ Ένα URI (Uniform Resource Identifier) είναι μια συμβολοσειρά χαρακτήρων που προσδιορίζει μια τοποθεσία ή ένα πόρο και επιτρέπει την αλληλεπίδραση με αυτά σε ένα δίκτυο, τοπικά ή στο Διαδίκτυο, χρησιμοποιώντας ειδικά πρωτόκολλα όπως το HTTP ή το FTP.

στα αρχεία του συστήματος. Μετά από κάθε αλλαγή που πραγματοποιείται στα αρχεία αυτά, επανεκκινεί τον εξυπηρετητή. Αυτό επιτρέπει να παρατηρήσουμε άμεσα τις νέες αλλαγές που πραγματοποιούνται και να τις ελέγξουμε έπειτα διενεργώντας τις απαραίτητες δοκιμές.

- Η βιβλιοθήκη **bcrypt** με την βοήθεια της οποίας κωδικοποιούμε τους κωδικούς των χρηστών.
- Η βιβλιοθήκη **cookie-parser**, η οποία μας βοηθάει στην αυθεντικοποίηση των χρηστών μέσω πληροφοριών που υπάρχουν στα cookies.
- Η βιβλιοθήκη **mongoose**. Η βιβλιοθήκη αυτή διασυνδέει μια εφαρμογή ιστού με υποκείμενη NoSQL βάση δεδομένων (MongoDB) ώστε να διευκολύνει την εφαρμογή στην διαχείριση των δεδομένων της βάσης ενώ επιτρέπει την δημιουργία σχήματος (schema) για την πλήρη μοντελοποίηση των δεδομένων (της εφαρμογής).

Για τη βάση δεδομένων, λόγω των απαιτήσεων της εργασίας, επιλέχθηκε να χρησιμοποιηθεί βάση δεδομένων τύπου NoSQL. Οι NoSQL βάσεις δεδομένων είναι σχεδιασμένες για αποθήκευση μεγάλου όγκου δεδομένων, ενώ οι πιο κοινοί τύποι αρχείων που χρησιμοποιούνται είναι:

1. Έγγραφα (Documents): Αυτός είναι ο τύπος που χρησιμοποιείται στις βάσεις δεδομένων τύπου document-oriented (προσανατολισμένες στα έγγραφα), όπως η MongoDB. Τα έγγραφα είναι δομημένα κατά JSON ή BSON (Binary JSON).
2. Στήλες (Columns): Αυτός είναι ο τύπος που χρησιμοποιείται στις βάσεις δεδομένων τύπου column-oriented (προσανατολισμένες στις στήλες), όπως η Apache Cassandra. Τα δεδομένα αποθηκεύονται σε στήλες και όχι σε γραμμές όπως στις σχεσιακές βάσεις.
3. Κλειδιά-τιμές (Key-Value): Αυτός είναι ο τύπος που χρησιμοποιείται στις βάσεις δεδομένων τύπου key-value, όπως η Redis. Κάθε έγγραφο αποθηκεύεται με ένα κλειδί που αντιστοιχεί σε μια τιμή.
4. Γράφοι (Graphs): Αυτός είναι ο τύπος που χρησιμοποιείται στις βάσεις δεδομένων γράφων, όπως η Neo4j. Τα δεδομένα αποθηκεύονται ως γράφοι, οι οποίοι περιλαμβάνουν κόμβους που συνδέονται με ακμές.

Ο λόγος της επιλογής βάσης τύπου NoSQL είναι λόγω των απαιτήσεων της εργασίας (εισαγωγή διαφόρων δεδομένων σε JSON/XML) καθώς και της ταχύτητας εισαγωγής πολλών εγγραφών στην βάση.

Από τα διάφορα είδη συστημάτων διαχείρισης βάσεων δεδομένων προσανατολισμένα στα έγγραφα επιλέχθηκε το MongoDB διότι φαίνεται να είναι το πιο δημοφιλές και καλύτερο (δείτε πχ. σύγκριση με την Couchbase εδώ: <https://www.mongodb.com/mongodb-vs-couchbase>). Το MongoDB αποθηκεύει τα δεδομένα σε μορφή JSON με κάθε έγγραφο να έχει το δικό της μοναδικό

αναγνωριστικό, το οποίο δημιουργείται αυτόματα. Το τελευταίο διευκολύνει πολύ την εύρεση της κάθε εγγραφής (πχ. φοιτητή) για επεξεργασία και για τις συσχετίσεις με τις άλλες συλλογές (πχ. διδασκαλίες).

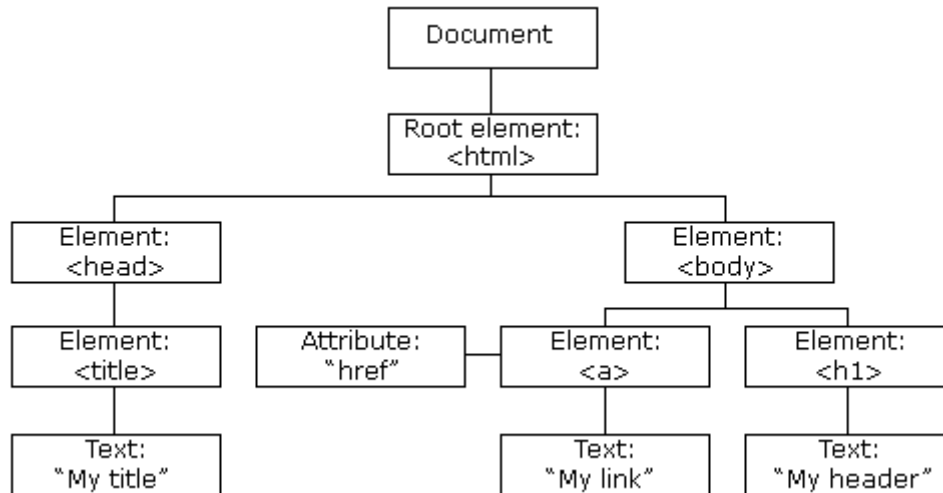
3.4.3. Τεχνολογίες & Εργαλεία Front-end Συστήματος

Όπως είπαμε και σε προηγούμενη παράγραφο, το front-end τμήμα του λογισμικού αποτελείται από το σύνολο των ιστοσελίδων με τις οποίες οι χρήστες μπορούν να αλληλοεπιδράσουν.

Για την ευκολότερη ανάπτυξη των σελίδων (views) της εφαρμογής ιστού χρησιμοποιήθηκε η γλώσσα μοτίβων HBS (Handlebars). Η συγκεκριμένη γλώσσα δίνει τη δυνατότητα της εγγραφής καθαρότερου και πιο κατανοητού κώδικα μέσω τη χρήσης μοτίβων (templates). Τα μοτίβα περιέχουν HTML κώδικα και εκφράσεις (expressions), οι οποίες με τη σειρά τους περιέχουν μεταβλητές ή βοηθητικές συναρτήσεις ενώ η εισαγωγή των εκφράσεων γίνεται με τα σύμβολα `{{}}`. Μέσα στα τελευταία μια μεταβλητή επισημαίνεται με `{{ variable }}`. Επίσης, μπορούν να χρησιμοποιηθούν helpers (στοιχεία ροής ελέγχου), όπως `if`, `each` σε δεδομένα, τα οποία αντιστοιχίζονται σε προϋποθέσεις/συνθήκες, οι οποίες εφόσον ικανοποιούνται οδηγούν στην εμφάνιση αντίστοιχων, κατάλληλων αποτελεσμάτων. Οι helpers είναι ένας τρόπος να προσθέσουμε προσαρμοσμένη λειτουργικότητα στα μοτίβα. Για παράδειγμα, μια επανάληψη `each` μπορεί να εμφανίσει όλα τα περιεχόμενα μιας λίστας με JSON δεδομένα.

Οι Ajax κλήσεις που χρησιμοποιήθηκαν, είναι έξτρα αιτήματα που στέλνει ο πελάτης με δεδομένα JSON στον εξυπηρετητή (πχ. για την δυναμική αντίδραση σε γεγονότα όπως το άνοιγμα μιας λίστας όπου τα δεδομένα της λίστας αντλούνται δυναμικά από τον εξυπηρετητή). Πρέπει να χρησιμοποιούνται με προσοχή για να μην επιβαρύνεται σημαντικά ο εξυπηρετητής με πολλαπλές αιτήσεις ανά ιστοσελίδα.

Το jQuery είναι μια βιβλιοθήκη της Javascript που έχει σχεδιαστεί για να απλοποιεί τη διαχείριση δεδομένων του HTML DOM (Document Tree Of Objects) καθώς και τον χειρισμό συμβάντων, όπως το πάτημα κουμπιού. Το DOM αποτελεί ένα δεντρικό μοντέλο αντικειμένων (στοιχείων σελίδας) (δείτε την Εικόνα 8) που κατασκευάζεται αυτόματα κάθε φορά που φορτώνεται μία σελίδα.



Εικόνα 8 - Δέντρο Οντοτήτων

Τέλος, χρησιμοποιήθηκε το framework **bootstrap** και συγκεκριμένα οι πίνακες που διαθέτει για την προβολή και την επεξεργασία των δεδομένων. Το bootstrap αποτελεί μια συλλογή από τμήματα κώδικα που περιλαμβάνουν HTML, CSS, και Javascript με σκοπό την πιο γρήγορη και πιο εύκολη συγγραφή κώδικα για πλήρως ανταποκρινόμενες (responsive) σελίδες. Πιο συγκεκριμένα, βοηθάει στη δημιουργία σελίδων που προσαρμόζονται αυτόματα σε διάφορες συσκευές και οθόνες, όπως υπολογιστές, τάμπλετ και κινητά τηλέφωνα. Αυτό σημαίνει ότι οι χρήστες μπορούν να προβάλλουν και να αλληλεπιδρούν με την εφαρμογή σε διάφορες συσκευές χωρίς να υπάρχει απώλεια λειτουργικότητας ή εμφάνισης.

3.4.4. Εργαλεία Ανάπτυξης Συστήματος

Εφόσον επιλέχθηκαν τα απαραίτητα εργαλεία & τεχνολογίες για την υλοποίηση των δύο βασικών μερών του συστήματος, το επόμενο βήμα είναι η υλοποίηση του μέσω της χρήσης κατάλληλων εργαλείων ανάπτυξης και δοκιμής.

Αρχικά, επιλέχθηκε το Visual Studio Code⁶. Το Visual Studio Code είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης λογισμικού που κατασκευάστηκε από τη Microsoft. Αποτελεί το διασημότερο περιβάλλον ανάπτυξης λογισμικού καθώς υποστηρίζει σχεδόν όλες τις κύριες γλώσσες προγραμματισμού, όπως Java, C, C++, Javascript, Python καθώς και πολλές άλλες. Επίσης, προσφέρει πολλές επεκτάσεις (extensions) προς βοήθεια στους προγραμματιστές, είναι

⁶ <https://code.visualstudio.com/>

απλό στη χρήση, επιτρέπει ενέργειες, όπως αποσφαλμάτωση (debugging) και δοκιμή (testing), ενώ προσφέρεται δωρεάν, πράγμα που το κάνει προσβάσιμο σε όλους.

Στη συνέχεια εγκαταστήσαμε το εργαλείο Postman⁷. Το Postman αποτελεί το πιο διαδεδομένο εργαλείο για την χειρωνακτική δοκιμή REST API παγκοσμίως. Περιέχει μια γραφική διεπαφή μέσα από την οποία μπορεί ο προγραμματιστής να αποστέλλει αιτήματα σε ένα API και να βλέπει τις απαντήσεις που επιστρέφονται. Οπότε, εν γένει, ο προγραμματιστής μπορεί να δοκιμάσει όσες λειτουργίες έχουν υλοποιηθεί στο API αυτό, όπως αποστολή αρχείων, τροποποίηση δεδομένων κλπ.

Άλλο ένα εργαλείο που χρησιμοποιήθηκε είναι το Robo 3T⁸. Το Robo 3T είναι ένα διαδεδομένο γραφικό περιβάλλον που χρησιμοποιείται για την επίβλεψη βάσεων δεδομένων του συστήματος διαχείρισης MongoDB. Μέσα από αυτό το εργαλείο παρέχονται οι δυνατότητες δημιουργίας, διαγραφής και τροποποίησης των εγγράφων και των συλλογών που υπάρχουν σε μία βάση.

Τέλος, χρησιμοποιήθηκε η εφαρμογή GitHub Desktop⁹. Το GitHub Desktop είναι μια εφαρμογή που επιτρέπει την επικοινωνία με το GitHub χρησιμοποιώντας γραφικό περιβάλλον αντί για γραμμή εντολών ή φυλλομετρητή. Το Github είναι μια εταιρία που προσφέρει μία υπηρεσία (διαχείρισης) αποθετηρίων Git σε σύννεφο (cloud). Το Git είναι ένα κατανεμημένο σύστημα ελέγχου εκδόσεων ανοιχτού κώδικα που δημιουργήθηκε από τον Linus Torvalds το 2005. Το σύστημα αυτό οργανώνει τα έργα λογισμικού σε κεντροποιημένα αποθετήρια πηγαίου κώδικα (όπως αυτά που φιλοξενούνται από το GitHub). Το κάθε αποθετήριο εμπεριέχει όλο τον πηγαίο κώδικα και το ιστορικό επεξεργασίας του για ένα έργο λογισμικού ώστε να είναι διαθέσιμα στον κάθε προγραμματιστή. Με αυτό τον τρόπο, ο προγραμματιστής μέσω του Git μπορεί να αντιγράψει με ασφάλεια τον πηγαίο κώδικα, να πραγματοποιήσει τις όποιες αλλαγές χρειάζεται και στη συνέχεια να υποβάλλει τις αλλαγές αυτές πίσω στον κύριο πηγαίο κώδικα του αποθετηρίου. Με λίγα λόγια, το Git έχει ως σκοπό να βοηθάει τους προγραμματιστές στο να αποθηκεύουν και να επεξεργάζονται τον κώδικά τους με ομαδικό τρόπο καθώς και να μπορούν να παρατηρούν και να ελέγχουν τυχών αλλαγές που έχουν πραγματοποιηθεί από ανεξάρτητα, τρίτα μέρη (πχ. τα μέλη μιας κοινότητας).

⁷ <https://www.postman.com/>

⁸ <https://robomongo.org/>

⁹ <https://desktop.github.com/>

3.5. Βαθμός Ικανοποίησης Απαιτήσεων

Σε αυτό το κεφάλαιο θα παρουσιαστεί ο βαθμός ικανοποίησης των λειτουργικών και μη λειτουργικών απαιτήσεων από το προσφερόμενο σύστημα, όπως αυτές αναλύθηκαν στο κεφάλαιο 3.1

Πίνακας 1 - Βαθμός Ικανοποίησης Λειτουργικών Απαιτήσεων

Οντότητα	Λειτουργική Απαίτηση	Βαθμός Ικανοποίησης	Συνοπτική Επεξήγηση
Χρήστες			
	1. Το σύστημα πρέπει να υποστηρίζει τη δημιουργία λογαριασμού χρήστη – διδασκων	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει την δημιουργία λογαριασμού χρήστη-διδασκων
	2. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή να βλέπει τη λίστα με τους διδασκοντες-χρήστες	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διαχειριστή να βλέπει τη λίστα με τους διδασκοντες
	3. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή την διαγραφή χρηστών (διδασκόντων) από το σύστημα.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διαχειριστή την διαγραφή χρηστών από το σύστημα
	4. Το σύστημα πρέπει να επιτρέπει την τροποποίηση των στοιχείων των χρηστών	Δεν ικανοποιείται	Το σύστημα δεν επιτρέπει την τροποποίηση των στοιχείων των χρηστών
Μαθήματα			
	1. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή την εισαγωγή μαθημάτων, μέσω αρχείων JSON, με κάτω όρια και βάρη για κάθε μέρος βαθμολογίας ενός μαθήματος. Τα κάτω όρια επιτρέπουν να προσδιοριστεί αν ο βαθμός ενός φοιτητή σε ένα μέρος του μαθήματος είναι προβιβάσιμος ή όχι. Ενώ τα βάρη αποτελούν την σχετική βαρύτητα που έχει ένα μέρος βαθμολόγησης σε	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει την εκφόρτωση μαθημάτων από αρχεία JSON, με εμφάνιση μηνύματος λάθους σε περίπτωση σφάλματος. Το αρχείο πρέπει να είναι συγκεκριμένης διαμόρφωσης

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

	σχέση με τα άλλα για ένα μάθημα.		
	2. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή την ανάθεση μαθημάτων σε εγγεγραμμένους στο σύστημα χρήστες-διδάσκοντες.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διαχειριστή την ανάθεση μαθημάτων σε εγγεγραμμένους στο σύστημα χρήστες-διδάσκοντες
	3. Το σύστημα πρέπει να επιτρέπει στον διαχειριστή να βλέπει την λίστα των μαθημάτων και σε ποιους διδάσκοντες αυτά έχουν ανατεθεί.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διαχειριστή να βλέπει την λίστα των μαθημάτων και σε ποιους διδάσκοντες αυτά έχουν ανατεθεί.
	4. Το σύστημα πρέπει να παρέχει τη δυνατότητα στον διδάσκων να καθορίσει για ένα μάθημα αν θα έχει δύο βασικά μέρη (θεωρία & εργαστήριο) ή μόνο ένα (θεωρία) μέρος καθώς και τους τρόπους αξιολόγησης φοιτητών ανά μέρος με την αντίστοιχη βαρύτητά τους με έναν ιεραρχικό τρόπο.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διδάσκων να καθορίσει για ένα μάθημα αν θα έχει δύο βασικά μέρη (θεωρία & εργαστήριο) ή μόνο ένα (θεωρία) μέρος καθώς και τους τρόπους αξιολόγησης φοιτητών ανά μέρος με την αντίστοιχη βαρύτητά τους με έναν ιεραρχικό τρόπο
	5. Το σύστημα πρέπει να υποστηρίζει την εκφόρτωση μαθημάτων από αρχεία τύπου JSON από τον διαχειριστή του συστήματος. Σε περίπτωση που προκύψει κάποιο σφάλμα, ακυρώνεται η εκφόρτωση και εμφανίζεται μήνυμα λάθους	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα υποστηρίζει την εκφόρτωση μαθημάτων από αρχεία τύπου JSON από τον διαχειριστή του συστήματος. Σε περίπτωση που προκύψει κάποιο σφάλμα, ακυρώνεται η εκφόρτωση και εμφανίζεται μήνυμα λάθους
Διδασκαλίες			
	1. Το σύστημα πρέπει να επιτρέπει τη δημιουργία διδασκαλιών (χειρωνακτική/μαζική) για το κάθε μάθημα που έχει αναθέσει στον διδάσκων ο διαχειριστής του συστήματος - με χρονολογία και αριθμό εξαμήνου.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει την εκφόρτωση διδασκαλιών από αρχεία JSON, με εμφάνιση μηνύματος λάθους σε περίπτωση σφάλματος. Το αρχείο πρέπει να είναι συγκεκριμένης διαμόρφωσης

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

	2. Το σύστημα πρέπει να επιτρέπει στον διδάσκων να βλέπει και να επεξεργάζεται τα στοιχεία μιας εκφορτωμένης διδασκαλίας, εφόσον ο διδάσκων έχει ανατεθεί σε αυτή, όπως αυτά παρέχονται από το αντίστοιχο αρχείο εκφόρτωσης (τα βάρη της δομής του μαθήματος, τα κάτω όρια και τη χρονική διάρκεια που θα κρατιέται ένας προβιβάσιμος βαθμός)	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διδάσκων να βλέπει και να επεξεργάζεται τα στοιχεία μιας εκφορτωμένης διδασκαλίας
	3. Το σύστημα πρέπει να παρέχει στον διδάσκων τη δυνατότητα διαχείρισης (διάβασμα & ανανέωση) της όλης ιεραρχίας αξιολόγησης/βαθμολόγησης.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διδάσκων τη διαχείριση (διάβασμα & ανανέωση) της όλης ιεραρχίας αξιολόγησης/βαθμολόγησης
	4. Το σύστημα πρέπει να παρέχει τη δυνατότητα στον διδάσκων να ανανεώνει διδασκαλίες από ένα μάθημα	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διδάσκων να προσθέτει και να ανανεώνει διδασκαλίες από ένα μάθημα
	5. Το σύστημα πρέπει να επιτρέπει μόνο στον διαχειριστή να αφαιρεί διδασκαλίες	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει μόνο στον διαχειριστή να αφαιρεί διδασκαλίες
	6. Το σύστημα πρέπει να επιτρέπει το πάγωμα μιας διδασκαλίας χειροκίνητα (από επιλογή του αντίστοιχου κουμπιού στο γραφικό περιβάλλον)	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει το πάγωμα μιας διδασκαλίας χειροκίνητα
Φοιτητές			
	1. Το σύστημα πρέπει να επιτρέπει στον διδάσκων την μαζική εισαγωγή σπουδαστών σε συγκεκριμένη διδασκαλία μαθήματος, από αρχείο τύπου EXCEL	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει τη μαζική εισαγωγή σπουδαστών από αρχείο EXCEL. Το αρχείο πρέπει να είναι συγκεκριμένης διαμόρφωσης
	2. Το σύστημα πρέπει να	Ικανοποιεί την	Το σύστημα επιτρέπει στον

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

	επιτρέπει στον διδάσκων να βλέπει όλες τις πληροφορίες (αριθμό μητρώου, ονοματεπώνυμο, email) για τους σπουδαστές που είναι εγγεγραμμένοι σε ένα μάθημα ή διδασκαλία	απαίτηση πλήρως	διδάσκων να βλέπει όλες τις πληροφορίες (αριθμό μητρώου, ονοματεπώνυμο, email) για τους σπουδαστές που είναι εγγεγραμμένοι σε ένα μάθημα ή διδασκαλία
	3. Το σύστημα πρέπει να επιτρέπει στον διδάσκων να δει το σύνολο των σπουδαστών ανά μάθημα και ανά διδασκαλία. Πιο συγκεκριμένα θα πρέπει να εμφανίζονται όλοι οι φοιτητές που έχουν εγγραφτεί ποτέ στο μάθημα (χωρίς διπλότυπα), μαζί με τις χρονολογίες των διδασκαλιών που έχουν εγγραφτεί.	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διδάσκων να δει το σύνολο των σπουδαστών ανά μάθημα και ανά διδασκαλία
	4. Το σύστημα θα πρέπει να υποστηρίζει τη δυνατότητα υπολογισμού των τελικών, συνολικών βαθμών του κάθε σπουδαστή με βάση τα αντίστοιχα βάρη στην διδασκαλία του μαθήματος και τον βαθμό επιδόσεων του σπουδαστή στα μέρη της διδασκαλίας του μαθήματος που έχει βαθμολογηθεί (στα φύλλα της ιεραρχίας αξιολόγησης της διδασκαλίας)	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα υποστηρίζει τη δυνατότητα υπολογισμού των τελικών, συνολικών βαθμών του κάθε σπουδαστή με βάση τα αντίστοιχα βάρη στην διδασκαλία του μαθήματος και τον βαθμό επιδόσεων του σπουδαστή στα μέρη της διδασκαλίας του μαθήματος που έχει βαθμολογηθεί
	5. Το σύστημα θα πρέπει να υποστηρίζει τη δυνατότητα τροποποίησης του βαθμού από τον διδάσκων	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα υποστηρίζει τη δυνατότητα τροποποίησης του βαθμού από τον διδάσκων
	6. Το σύστημα πρέπει να διατηρεί ιστορικό με τις βαθμολογίες των σπουδαστών ανά διδασκαλία. Το ιστορικό θα εμφανίζεται ύστερα από	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα διατηρεί ιστορικό με τις βαθμολογίες σπουδαστών ανά διδασκαλία.

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

	αίτηση του διδάσκοντος για οποιοδήποτε σπουδαστή διδασκαλίας του		
	7. Το σύστημα πρέπει να παρέχει τη δυνατότητα στο διδάσκων να εξάγει τις τυπικές και τις αναλυτικές βαθμολογίες όλων των σπουδαστών σε μορφή EXCEL	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα επιτρέπει στον διδάσκων να εξάγει τις τυπικές και τις αναλυτικές βαθμολογίες όλων των σπουδαστών σε μορφή EXCEL

Πίνακας 2 - Βαθμός Ικανοποίησης Μη Λειτουργικών Απαιτήσεων

Κατηγορία	Μη Λειτουργική Απαίτηση	Περιγραφή	Βαθμός	Συνοπτική Επεξήγηση
Αξιοπιστία				
	1.1	Επαναφορά σε πλήρη λειτουργία μετά από βλάβη	Ικανοποιεί την απαίτηση μερικώς	Το σύστημα μπορεί να αποκατασταθεί και να επανέλθει σε πλήρη λειτουργία μέσα σε συγκεκριμένο χρονικό διάστημα. Αυτό εξαρτάται από τον διαχειριστή του συστήματος (πόσο γρήγορα θα εντοπίσει και θα διορθώσει το σχετικό πρόβλημα/βλάβη).
	1.2	Μη εμφάνιση κρίσιμων σφαλμάτων	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα δεν εμφανίζει κρίσιμα σφάλματα, είναι σταθερό και αξιόπιστο.
Απόδοση				
	2.1	Διαχείριση απομακρυσμένα	Ικανοποιεί την απαίτηση πλήρως	Η διαχείριση του συστήματος είναι δυνατή απομακρυσμένα.
	2.2	Ταχύτητα απάντησης σε εντολές χρήστη	Ικανοποιεί την απαίτηση πλήρως	Η ταχύτητα απάντησης του συστήματος στις εντολές των χρηστών είναι άμεση και ικανοποιητική.
Ευχρηστία				
	3.1	Ευκολία στην	Ικανοποιεί	Το σύστημα είναι εύκολο στην

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

		εκμάθηση και χρήση	την απαίτηση πλήρως	εκμάθηση και χρήση από τους χρήστες.
	3.2	Παροχή οδηγιών για ενέργειες χρηστών	Ικανοποιεί την απαίτηση πλήρως	Παρέχονται οδηγίες για τη σειρά των ενεργειών που μπορεί να εκτελέσει ο χρήστης.
	3.3	Εμφάνιση κατάλληλων μηνυμάτων λάθους	Ικανοποιεί την απαίτηση πλήρως	Εμφανίζονται κατάλληλα μηνύματα λάθους στους χρήστες όταν αυτοί διαπράττουν σφάλματα.
	3.4	Παροχή υποκείμενης REST υπηρεσίας με επίπεδο 3	Ικανοποιεί την απαίτηση πλήρως	Παρέχεται υποκείμενη REST υπηρεσία που υλοποιεί τις βασικές λειτουργίες του συστήματος με επίπεδο 3.
Φορητότητα & Συμβατότητα				
	4.1	Διαθεσιμότητα σε οποιαδήποτε συσκευή	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα είναι διαθέσιμο σε οποιαδήποτε συσκευή που έχει τους απαραίτητους πόρους για να το φιλοξενήσει.
	4.2	Διαχείριση απομακρυσμένα	Ικανοποιεί την απαίτηση πλήρως	Η διαχείριση του συστήματος μπορεί να γίνει απομακρυσμένα.
	4.3	Παροχή υποκείμενης REST υπηρεσίας με επίπεδο ωριμότητας 3	Ικανοποιεί την απαίτηση πλήρως	Παρέχεται υποκείμενη REST υπηρεσία που υλοποιεί τις βασικές λειτουργίες του συστήματος με επίπεδο 3.
Ασφάλεια				
	5.1	Κωδικοποίηση κωδικών χρηστών στη βάση δεδομένων	Ικανοποιεί την απαίτηση πλήρως	Οι κωδικοί των χρηστών κωδικοποιούνται στη βάση δεδομένων για αυξημένη ασφάλεια.
	5.2	Ταυτοποίηση και εξουσιοδότηση βασισμένη σε ρόλους	Ικανοποιεί την απαίτηση πλήρως	Το σύστημα παρέχει ταυτοποίηση και εξουσιοδότηση βασισμένη σε ρόλους, όπως διδάσκοντες και διαχειριστές.

4. Επίδειξη Συστήματος

Σε αυτό το κεφάλαιο περιγράφεται η διαδικασία εγκατάστασης και εκτέλεσης του συστήματος καθώς και επιδεικνύονται ορισμένες περιπτώσεις χρήσης του.

4.1. Διαδικασία Εγκατάστασης & Εκτέλεσης Συστήματος

Στο συγκεκριμένο σημείο ακολουθούν αναλυτικές οδηγίες για την εγκατάσταση του συστήματος σε ένα τοπικό υπολογιστή. Η διαδικασία εγκατάστασης είναι αρκετά εύκολη. Αρχικά θα πρέπει να εγκαταστήσουμε τη βάση δεδομένων.

Τα βήματα εγκατάστασης της βάσης δεδομένων είναι τα παρακάτω:

1. Κατεβάζουμε το συμπιεσμένο αρχείο από τον παρακάτω σύνδεσμο: <https://www.mongodb.com/try/download/community> ανάλογα με τα χαρακτηριστικά του υπολογιστή μας (πχ. λειτουργικό σύστημα).
2. Στη συνέχεια αποσυμπιέζουμε το εκφορτωμένο αρχείο σε ένα νέο φάκελο με ονομασία mongodb και τον μετακινούμε στον home φάκελο (directory). Για παράδειγμα: "C:\Users\ilias\mongodb" όπου ilias είναι το όνομα του βασικού χρήστη του συστήματος εφόσον βρισκόμαστε σε περιβάλλον Windows.

Έπειτα, πρέπει να εγκαταστήσουμε το Node.js εκτελώντας τον installer που πρέπει να εκφορτωθεί από τον ακόλουθο σύνδεσμο: <https://nodejs.org/en/> (επιλέγουμε και πάλι τον σωστό installer με βάση τα χαρακτηριστικά του υπολογιστή μας).

Στη συνέχεια θα πρέπει να εγκαταστήσουμε την εφαρμογή ιστού με βάση τα εξής βήματα:

- κάνουμε clone τον πηγαίο κώδικα της εφαρμογής από το GitHub σύνδεσμο <https://github.com/ilias24k/grade-managment-system> μέσω του αντίστοιχου κουμπιού στο GitHub Desktop ή αντίστοιχης εντολής Git από την κονσόλα (`git clone https://github.com/ilias24k/grade-managment-system`).
- τρέχουμε στη κονσόλα την εντολή `npm install` (στον ριζικό φάκελο του πηγαίου κώδικα) ώστε να εγκαταστήσουμε τις απαραίτητες βιβλιοθήκες και πακέτα που χρειάζονται για να λειτουργεί σωστά το λογισμικό της εφαρμογής. Το Node.js εγκαθιστά αυτόματα και τον NPM (Node Package Manager) (διαχειριστή πακέτων). Αυτό μπορεί να επαληθευτεί ανοίγοντας ένα παράθυρο τερματικού και πληκτρολογώντας της εντολές `"npm -v"` και `"node -v"`, όπου θα πρέπει να εμφανίζονται οι αντίστοιχες πρόσφατες εκδόσεις.

Η διαδικασία εκτέλεσης της εφαρμογής μας είναι κι αυτή εύκολη και αποτελείται από τα εξής βήματα:

- Ανοίγουμε ένα παράθυρο στη γραμμή εντολών και τρέχουμε σε αυτό την εντολή `dbinstall/bin/mong- --dbpath` όπου το `dbpath` είναι το μονοπάτι στο σύστημα αρχείων του υπολογιστή, που βρίσκονται τα αρχεία της βάσης δεδομένων, ώστε να ξεκινήσει η

λειτουργία του συστήματος της βάσης δεδομένων. Για παράδειγμα “dbpath=C:/Users/ilias/mongodb-data” όπου ilias είναι το όνομα του βασικού χρήστη του συστήματος

- Σε ένα άλλο παράθυρο γραμμής εντολών τρέχουμε την εντολή `nodemon grade-management-system`

Για να ελέγξουμε πως η εφαρμογή ιστού λειτουργεί κανονικά, ανοίγουμε τον φυλλομετρητή στη διεύθυνση `localhost:4000`, η οποία είναι η διεύθυνση όπου διατίθεται τοπικά η εφαρμογή αυτή.

4.2. Περιπτώσεις χρήσης

Σε αυτό το κεφάλαιο θα επιδείξουμε αρκετές περιπτώσεις χρήσης του συστήματος από τη δημιουργία νέου χρήστη μέχρι την βαθμολογία των σπουδαστών μιας διδασκαλίας ενός μαθήματος. Συγκεκριμένα το κεφάλαιο χωρίζεται σε τέσσερις ενότητες, όπου κάθε ενότητα αφορά μια ή περισσότερες περιπτώσεις χρήσης του συστήματος.

4.2.1. Αρχική Σελίδα, Εγγραφή και Σύνδεση

Σε αυτή την ενότητα, θα καλύψουμε τις περιπτώσεις χρήσης που σχετίζονται με την αρχική σελίδα της εφαρμογής, καθώς και την εγγραφή και τη σύνδεση των χρηστών.

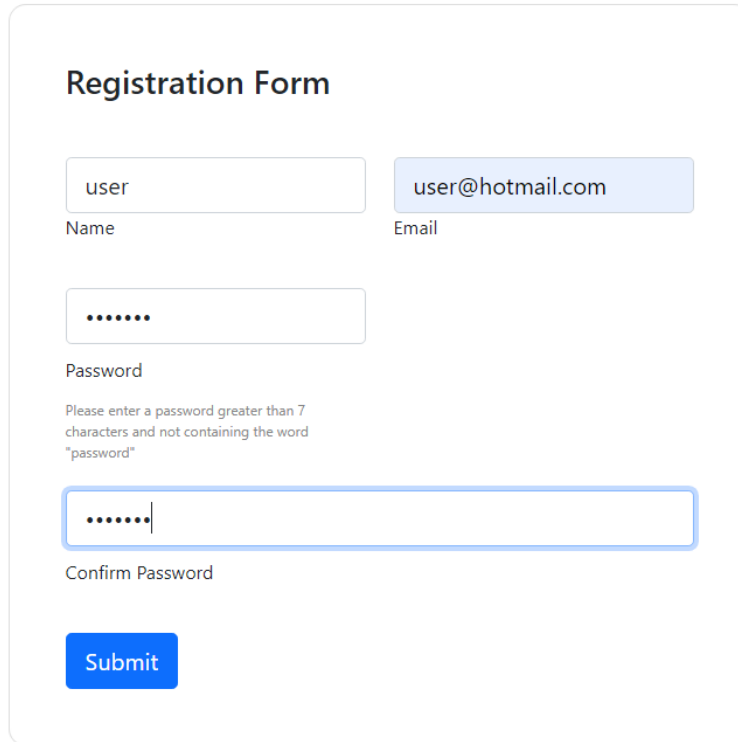
Αρχική σελίδα με τις οδηγίες. Αρχικά ο χρήστης βλέπει την αρχική σελίδα με οδηγίες για την εγγραφή και τη σύνδεση. Από το πάνω δεξιά μενού θα πρέπει έπειτα να διαλέξει είτε να εγγραφεί ή να ταυτοποιηθεί..



Εικόνα 9 - Αρχική Σελίδα

Εγγραφή χρήστη. Ένας νέος χρήστης εγγράφεται μέσω φόρμας που εμφανίζεται εφόσον επιλέξει

την επιλογή Signup από το προαναφερόμενο μενού.



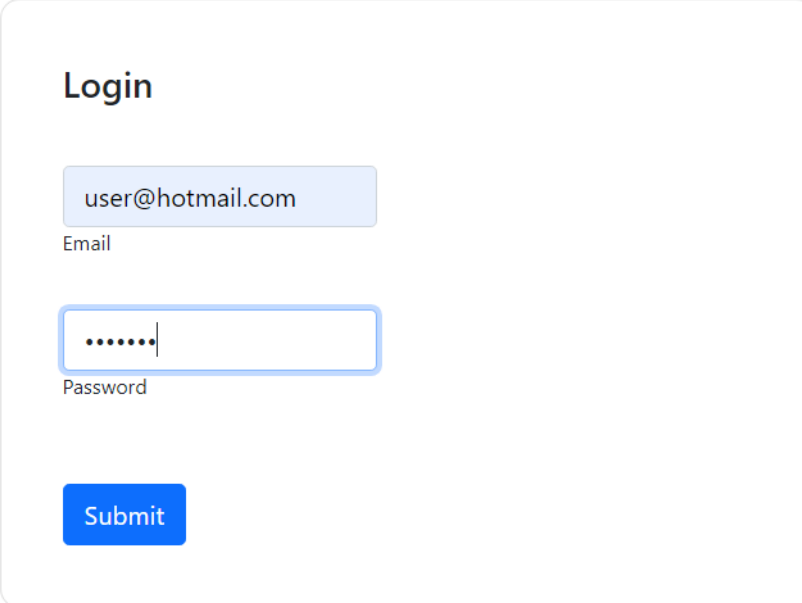
The image shows a registration form titled "Registration Form". It contains the following fields and elements:

- Name:** A text input field containing the text "user".
- Email:** A text input field containing the text "user@hotmail.com".
- Password:** A text input field containing six dots, representing a password.
- Confirm Password:** A text input field containing six dots and a vertical cursor line, representing a confirmation password.
- Submit:** A blue button with the text "Submit".

Below the Password field, there is a note: "Please enter a password greater than 7 characters and not containing the word 'password'".

Εικόνα 10 - Φόρμα Εγγραφής

Σύνδεση χρήστη. Στη συνέχεια, εφόσον η εγγραφή του ήταν επιτυχής, ο χρήστης συνδέεται στο σύστημα μέσω της φόρμας σύνδεσης/ταυτοποίησης με τα εγγεγραμμένα στοιχεία του.



The image shows a login form titled "Login". It contains two input fields: one for "Email" with the text "user@hotmail.com" and one for "Password" with six dots. Below the fields is a blue "Submit" button.

Εικόνα 11 - Φόρμα Σύνδεσης

4.2.2. Διαχείριση Διδασκαλιών και Σπουδαστών

Σε αυτή την υπο-ενότητα, θα αναλύσουμε τις περιπτώσεις χρήσης που αφορούν τη διαχείριση διδασκαλιών και των σπουδαστών εντός της εφαρμογής.

Διαχείριση διδασκαλιών. Στην παρακάτω εικόνα βλέπουμε τα μαθήματα που έχουν ανατεθεί στον διδάσκων, αφού αυτός έχει συνδεθεί στο σύστημα.

Courses

Course Name	Grade Maintain Time	Action
Java	5	View Teachings View Edit
Physics	1	View Teachings View Edit

Εικόνα 12 - Λίστα μαθημάτων που έχουν ανατεθεί στον διδάσκων

Ο διδάσκων έπειτα μπορεί να προχωρήσει στην μεταφόρτωση διδασκαλιών μαθήματος μέσω της επιλογής View Teachings και στη συνέχεια Upload Teachings.

Teachings

[+Upload Teachings](#)

Semester	Year	Duration	Teacher	Action
----------	------	----------	---------	--------

[Submit](#)

Μετά την εκφόρτωση, εμφανίζονται στον χρήστη οι διδασκαλίες που εκφορτώθηκαν, όπως φαίνεται στο παρακάτω σχήμα.

Teachings

[+Upload Teachings](#)

Semester	Year	Duration	Teacher	Action
winter	2023	2	user	Select Duration of Grade: -- select an option -- View Students
winter	2024	3	user	Select Duration of Grade: -- select an option -- View Students

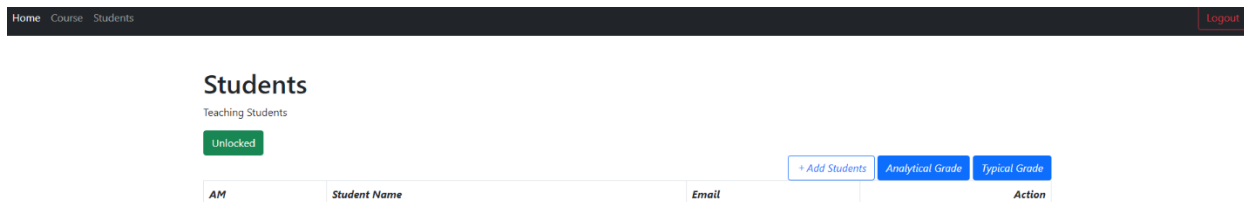
[Submit](#)

Εικόνα 13 - Εκφόρτωση Διδασκαλιών από αρχείο

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

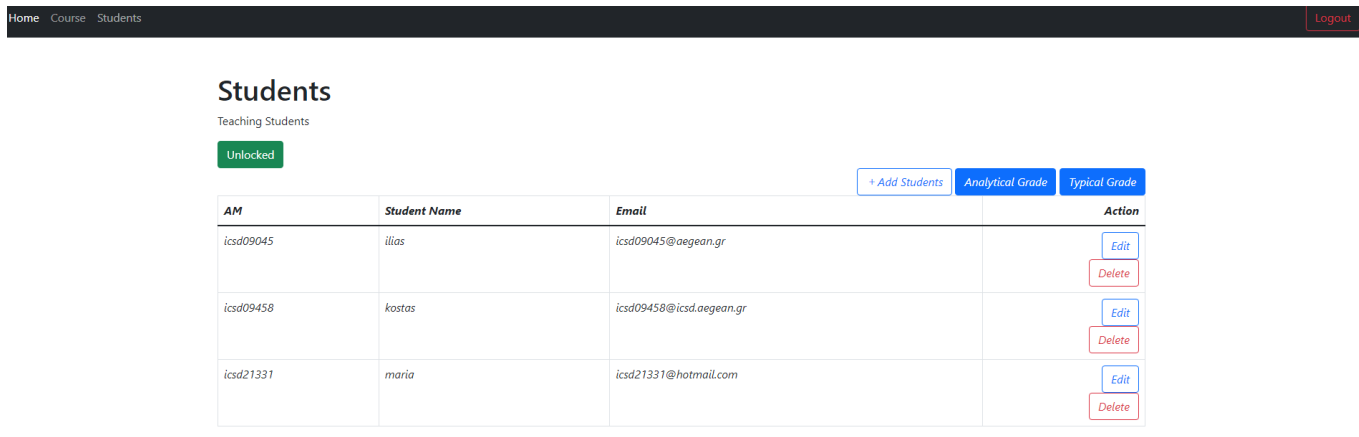
Διαχείριση σπουδαστών. Ο διδάσκων μπορεί να μεταφορτώσει σπουδαστές/φοιτητές με το πάτημα του View Students κουμπιού σε μια διδασκαλία και στη συνέχεια το πάτημα του Add Students κουμπιού στην επόμενη σελίδα.

Στο παρακάτω σχήμα φαίνεται το περιεχόμενο όταν πραγματοποιηθεί η επιλογή View Students.



Εικόνα 14 - Διαχείριση φοιτητών/σπουδαστών μιας διδασκαλίας

Εφόσον πραγματοποιηθεί η μεταφόρτωση των σπουδαστών, τότε αυτοί εμφανίζονται, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 15 - Ανέβασμα σπουδαστών

Από εκεί και πέρα, ο διδάσκων μπορεί να τροποποιεί ή να διαγράφει σπουδαστές πατώντας τα αντίστοιχα κουμπιά στην κατάλληλη γραμμή του εμφανιζόμενου πίνακα.

4.2.3. Επεξεργασία στοιχείων μαθήματος και βαθμολόγηση σπουδαστή

Επεξεργασία στοιχείων μαθήματος. Ο διδάσκων μπορεί να εισάγει νέα στοιχεία καθώς και να ενημερώνει παλαιά σε ένα μάθημα μέσω της εμφάνισης των μαθημάτων από το μενού Courses με

το πάτημα του κουμπιού Edit στην κατάλληλη γραμμή του μαθήματος.

General

Name	Weight	Bound	Edit
Theory	60	5	Edit
Lab	40	5	Edit

Theory

id	name	weight	bound	action
1	proodos	50	5	Edit Bound Delete
2	teliki	40	5	Edit Bound Delete
3	test	10	5	Edit Bound Delete

[+ Add Row](#)

Lab

id	name	weight	bound	action
1	askisi1	25	5	Edit Bound Delete
2	askisi2	25	5	Edit Bound Delete
3	omadiki	50	5	Edit Bound Delete

[+ Add Row](#)

[Submit](#)

Εικόνα 16 - Επεξεργασία στοιχείων μαθήματος

Βαθμολόγηση σπουδαστή. Η βαθμολόγηση σπουδαστή γίνεται με την επιλογή του κουμπιού Edit στον σπουδαστή. Η επιλογή εμφανίζεται ύστερα από την επιλογή View Teachings σε ένα μάθημα, στο μενού Courses και View Students σε διδασκαλία στο επόμενο μενού. Από ότι φαίνεται στην παρακάτω εικόνα, ο διδάσκων θα πρέπει να παρέχει βαθμούς για τον σπουδαστή μόνο στα φύλλα της ιεραρχίας αξιολόγησης.

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

Home Course StudentsLogout

Theory

id	name	weight %	insert grade	grade	final grade
1	proodos	50	<input type="text" value="6"/>		
2	teliki	50	<input type="text" value="6"/>		
3	test	10	<input type="text" value="2"/>		

Lab

id	name	weight %	insert grade	grade	final grade
1	askisi1	25	<input type="text" value="7"/>		
2	askisi2	25	<input type="text" value="3"/>		
3	omadiki	50	<input type="text" value="5"/>		
4	test	10	<input type="text" value="6"/>		

Εικόνα 17 - Βαθμολόγηση σπουδαστών

4.2.4. Εξαγωγή βαθμολογίας

Η εξαγωγή της συνολικής βαθμολογίας (αναλυτικής ή τυπικής), των σπουδαστών μιας διδασκαλίας πραγματοποιείται από την επιλογή View Teachings σε ένα μάθημα, στο μενού Courses, στη συνέχεια την επιλογή View Students σε μια διδασκαλία και τέλος την επιλογή του αντίστοιχου κουμπιού για την εξαγωγή της τυπικής ή αναλυτικής βαθμολογίας, αντίστοιχα.

Students

Teaching Students

Unlocked

+ Add StudentsAnalytical GradeTypical Grade

AM	Student Name	Email	Action
icsd09045	ilias	icsd09045@aegean.gr	<div style="display: flex; flex-direction: column; gap: 5px;">EditDelete</div>
icsd09458	kostas	icsd09458@icsd.aegean.gr	<div style="display: flex; flex-direction: column; gap: 5px;">EditDelete</div>
icsd21331	maria	icsd21331@hotmail.com	<div style="display: flex; flex-direction: column; gap: 5px;">EditDelete</div>

Εικόνα 18 - Επιλογή εξαγωγής Τυπικής ή Αναλυτικής βαθμολογίας

Διπλωματική εργασία: Διαμορφώσιμο, Βασιζόμενο στον Παγκόσμιο Ιστό Σύστημα Διαχείρισης Βαθμολογιών

Παρακάτω παραθέτουμε την τυπική βαθμολογία όπως αυτή εμφανίζεται όταν ανοίξουμε το αντίστοιχο εκφορτωμένο αρχείο από το πρόγραμμα Excel.

A	B	C	D	E
name	email	finalGradeTheory	finalGradeLab	total
ilias	icsd09045@aegean.gr	7	4.85	2.425
kostas	icsd09458@icsd.aegean.gr	2.5	2	1
maria	icsd21331@hotmail.com	5.5	5.25	5.375

Εικόνα 19 - Τυπική Βαθμολογία

Παρομοίως, μια αναλυτική βαθμολογία θα εμφανίζεται πάλι με την μορφή ενός Excel αρχείου.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
name	email		proodos	teliki	test	askisi1	askisi2	omadiki	test		total lab	total theo	total	
ilias	icsd09045@aegean.g		3	4	0	1.75	0	2.5	0.6		7	4.85	2.425	
kostas	icsd09458@icsd.aegi		2.5	0	0	0	1.5	0	0.5		2.5	2	1	
maria	icsd21331@hotmail.		0	5	0.5	0	1.25	3	1		5.5	5.25	5.375	

Εικόνα 20 - Αναλυτική Βαθμολογία

5. Συμπεράσματα & Μελλοντική Εργασία

Σε αυτό το κεφάλαιο θα αναφερθούν τα συμπεράσματα και μελλοντικές βελτιώσεις που μπορούν να υπάρξουν στη εφαρμογή που αναπτύχθηκε στο πλαίσιο της τρέχουσας διπλωματικής εργασίας.

5.1. Συμπεράσματα

Μέσω της παρούσας διπλωματικής εργασίας υλοποιήθηκε μια διαδικτυακή εφαρμογή, η οποία επιχειρεί να εισάγει διευκολύνσεις για τις τυπικές και χρονοβόρες διαδικασίες που συναντώνται στον εκπαιδευτικό τομέα στην τριτοβάθμια εκπαίδευση, ειδικότερα ως προς την διαχείριση των διαφορετικών ειδών βαθμολογιών σε ένα μάθημα. Παρουσιάστηκαν λεπτομερώς τα στάδια σχεδίασης και υλοποίησης καθώς και αναλύθηκαν όλες οι λειτουργικές και μη απαιτήσεις του συστήματος/εφαρμογής προς κατασκευή. Η εφαρμογή αυτή δεν αποτελεί ένα ολοκληρωμένο προϊόν, όπως είναι τα συστήματα ηλεκτρονικής τάξης, αλλά μπορεί να θεωρηθεί ως μια χρήσιμη επέκτασή τους λόγω της πιο ευέλικτης διαχείρισης του τρόπου βαθμολόγησης ενός μαθήματος και των διδασκαλιών του.

Μέσω της εκπόνησης της εργασίας αυτής, αποκτήθηκε σημαντική εμπειρία στη διαχείριση και ανάπτυξη εφαρμογών με το σύστημα μη σχεσιακών βάσεων δεδομένων MongoDB, καθώς και στην ανάπτυξη REST APIs για το backend της εφαρμογής μέσω της χρήσης του Node.js & Express.js.

5.2. Μελλοντική εργασία

Κατά την ανάπτυξη της εφαρμογής, δόθηκε περισσότερη έμφαση στο back-end παρά στο front-end. Μπορούν να αλλάξουν μελλοντικά κάποια στοιχεία της διεπαφής του χρήστη ώστε να την κάνουν ακόμη πιο εύχρηστη και διαισθητική. Επίσης, θα μπορούσε να γίνει μεταφορά του συστήματος στο νέφος για να υποστηρίζονται και σενάρια δυναμικής κλιμάκωσής του ενώ θα μπορούσε να γίνει ακόμη και παραγωγή έκδοσης της εφαρμογής σε περιβάλλοντα κινητών συσκευών για ακόμα ευκολότερη και πιο γρήγορη εξυπηρέτηση του χρήστη. Εκτός των προηγούμενων, η εφαρμογή θα μπορούσε να υποστηρίξει τη δημιουργία περισσότερων ξεχωριστών ρόλων για τους χρήστες, όπως αυτόν του σπουδαστή. Επιπλέον, θα μπορούσαν να προστεθούν εργαλεία συνεργασίας και αλληλεπίδρασης, όπως συνομιλία ή συστήματα κοινής εργασίας, για διευκόλυνση της συνεργασίας μεταξύ χρηστών. Τέλος, θα μπορούσε να γίνει ενοποίηση της εφαρμογής, με την μορφή επέκτασης, με ένα ολοκληρωμένο σύστημα διαχείρισης ηλεκτρονικής τάξης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Martín Llamas-Nistal, Manuel J. Fernández-Iglesias, Juan González-Tato, and Fernando A. Mikic-Fonte. 2013. Blended e-assessment. *Comput. Educ.* 62, C (March 2013), 72–87.
2. Lukyamuzi, A. (2020). A Flexible Student Assessment System for Continuous Performance Monitoring. Busitema University, Uganda.
3. Κυριάκος Κρητικός. Διαφάνειες μαθήματος τεχνολογίας λογισμικού - Ενότητες Μοντελοποίησης Συστήματος, Αρχιτεκτονικής Σχεδίαση και Μηχανικής Απαιτήσεων.