

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ ΠΡΟΕΓΓΡΑΦΗΣ
ΦΟΙΤΗΤΩΝ ΣΕ ΜΑΘΗΑΤΑ ΠΕΡΙΟΡΙΣΜΕΝΗΣ
ΕΠΙΛΟΓΗΣ**

Μάιος 2023

Ατομική Διπλωματική Εργασία

**ΥΛΟΠΟΙΗΣΗ ΣΥΣΤΗΜΑΤΟΣ ΠΡΟΕΓΓΡΑΦΗΣ ΦΟΙΤΗΤΩΝ ΣΕ
ΜΑΘΗΑΤΑ ΠΕΡΙΟΡΙΣΜΕΝΗΣ ΕΠΙΛΟΓΗΣ**

Κωνσταντίνος Αναστασίου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2023

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Υλοποίηση Συστήματος Προεγγραφής Φοιτητών σε Μαθήματα Περιορισμένης
Επιλογής**
Κωνσταντίνος Αναστασίου

Επιβλέπων Καθηγητής
Ελπίδα Κεραυνού Παπαηλιού

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των
απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του
Πανεπιστημίου Κύπρου

Μάιος 2023

Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής μου εργασίας, ολοκληρώνεται ο κύκλος σπουδών μου στο Προπτυχιακό Πρόγραμμα της Επιστήμης της Πληροφορικής και ανοίγεται μια πόρτα σε νέες εμπειρίες.

Πρώτα, θα ήθελα να ευχαριστήσω όλους τους ανθρώπους που συνέβαλαν και ήταν μέρος της προσπάθειας αυτής.

Στη συνέχεια, θα ήθελα να ευχαριστήσω θερμά την επιβλέπουσα καθηγήτρια μου κα Ελπίδα Κεραυνού-Παπαηλιού, πρώην πρόεδρο του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου, για την ευκαιρία που μου έδωσε για την εκπόνηση της διπλωματικής μου εργασίας, καθώς επίσης για όλη την βοήθεια που έδωσε σε εμένα και τους συμφοιτητές μου κατά την διάρκεια της θητείας της ως Πρόεδρος Τμήματος, καθώς η συνεισφορά της στο Τμήμα Πληροφορικής είναι μεγάλη.

Επιπρόσθετα, θα ήθελα να ευχαριστήσω την οικογένεια μου, που παρόλες τις δυσκολίες που συναντήσαμε κατά την διάρκεια των σπουδών μου, δεν σταμάτησε να με υποστηρίζει και να είναι στο πλευρό μου. Δεν θα μπορούσα να μην ευχαριστήσω ονομαστικά την μητέρα μου, η οποία παρόλο που δεν βρίσκεται στην ζωή, μου έδωσε την δύναμη να ολοκληρώσω τις σπουδές μου.

Τέλος, θα ήθελα να ευχαριστήσω όλους τους φίλους και συμφοιτητές για την βοήθεια τους, αλλά και για τις όμορφες στιγμές που περάσαμε παρέα στο Πανεπιστήμιο Κύπρου

Περίληψη

Σε αυτή την εργασία εξηγείται η προσπάθεια επίλυσης του προβλήματος που δημιουργείται κατά την διάρκεια των εγγραφών κατά την έναρξη των εξαμήνων στο Τμήμα Πληροφορικής Πανεπιστημίου Κύπρου. Συγκεκριμένα, αρκετοί φοιτητές επιθυμούν να εγγραφούν σε κάποιο από τα μαθήματα περιορισμένης επιλογής, είτε λόγω κάποιας κατεύθυνσης που θέλουν να ακολουθήσουν, είτε από ενδιαφέρον, με αποτέλεσμα να μην υπάρχουν διαθέσιμες θέσεις για όλους και να υπάρχουν φοιτητές που δεν μπορούν να το παρακολουθήσουν. Η λύση η οποία αναλύεται είναι η δημιουργία μιας διαδικτυακής εφαρμογής μέσω της οποίας οι φοιτητές θα μπορούν να συνδέονται και θα καταχωρούν τις προτιμήσεις τους και τις προθέσεις τους στις εγγραφές των μαθημάτων του επόμενου εξαμήνου. Έτσι, θα υπάρχει η πληροφόρηση και η γνώση της πρόθεσης εγγραφής των φοιτητών, με αποτέλεσμα να δίδεται η για βελτίωση του προγραμματισμού του τμήματος, και θα δώσει την ευκαιρία σε περισσότερους φοιτητές να εγγραφούν στα μαθήματα περιορισμένης επιλογής της προτίμησής τους.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
1.1	Εισαγωγή	1
1.2	Κίνητρο	2
1.3	Δομή Διπλωματικής Εργασίας	3
Κεφάλαιο 2	Ανάλυση Απαιτήσεων και Προδιαγραφών	4
2.1	Εισαγωγή	4
2.2	Μεθοδολογία Ανάπτυξης	5
2.3	Προσδιορισμός Αναγκών και απαιτήσεων	6
2.3.1	Υποθέσεις – Σταθερές Συστήματος – Επεξήγηση	6
2.3.1	Προσδιορισμός Τυπικών Χρηστών	8
2.4	Ανάλυση Αναγκών	9
2.4.1.	Σενάρια Χρήσης	9
2.5	Καθορισμός Απαιτήσεων	10
2.5.1	Λειτουργικές Απαιτήσεις	10
2.5.2.	Μη Λειτουργικές Απαιτήσεις	10
2.6	Περιορισμοί	14
Κεφάλαιο 3	Προδιαγραφές Συστήματος.....	12
3.1	Αρχιτεκτονική	15
3.2	Τεχνολογίες και εργαλεία	16
3.2.1	Βάση Δεδομένων MongoDB	16
3.2.2	Node.js	17
3.2.3	ExpressJS	17
3.2.4	NPM – Node Package Manager	17
3.2.5	CSS – Cascading Style Sheets	18
3.2.6	MS Visual Studio Code	18

Κεφάλαιο 4	Υλοποίηση Συστήματος.....	4
4.1	Βάση Δεδομένων	19
4.2	API & Web Server	21
4.2.1	Οργάνωση Εργασίας Node. JS	21
4.2.2	View engine	23
Κεφάλαιο 5	Λειτουργίες Συστήματος	21
5.1	Φοιτητής	24
5.1.1	Σύνδεση	24
5.1.2	Αρχική Σελίδα	25
5.1.3	Πλοήγηση μαθημάτων	26
5.1.4	Σελίδα Μαθήματος	27
5.1.5	Εξειδικεύσεις	28
5.1.6	Ρυθμίσεις λογαριασμού	28
5.2	Διαχειριστής – Admin	29
5.2.1	Αρχική Σελίδα Διαχειριστή	29
5.2.2	Μαθήματα ανά Εξάμηνο	30
5.2.3	Μαθήματα – Admin	30
5.2.4	Όλα τα μαθήματα	31
5.2.5	Σελίδα μαθήματος – Admin	31
Κεφάλαιο 6	Συμπεράσματα	30
6.1	Συμπεράσματα	33
6.2	Μελλοντική Εργασία	34
Βιβλιογραφία		36

Κεφάλαιο 1

Εισαγωγή

1.1 Εισαγωγή	1
1.2 Κίνητρο	2
1.3 Δομή Διπλωματικής Εργασίας	3

1.1 Εισαγωγή

Στις μέρες μας, η τεχνολογία παίζει ένα αρκετά σημαντικό ρόλο στην ζωή μας και στην καθημερινότητα μας. Ο άνθρωπος, καταβάλλει συνεχώς προσπάθειες βελτίωσης της ποιότητας ζωής του αλλά και του εργασιακού του περιβάλλοντος. Για τον λόγο αυτό, ο ρυθμός ανάπτυξης της τεχνολογίας αυξάνεται εκθετικά και κάθε μέρα υπάρχει κάποια τεχνολογική ανακάλυψη. Έτσι, αφού η τεχνολογική ανάπτυξη επηρεάζει την καθημερινή μας ζωή δεν μπορεί να μην επηρεάσει και τα πανεπιστήμια, καθώς είναι τα εκπαιδευτικά ιδρύματα και πολλές από τις έρευνες λαμβάνουν μέρος σε αυτά.

Η παρούσα Ατομική Διπλωματική Εργασία επιχειρεί να επιλύσει ένα πρόβλημα που αφορά τις εγγραφές των φοιτητών του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου, χρησιμοποιώντας την τεχνολογία και πιο συγκεκριμένα τις τεχνολογίες διαδικτύου.

Το τμήμα Πληροφορικής έχει ως στόχο την καλύτερη και ομαλότερη λειτουργία για τους φοιτητές του. Στην αρχή κάθε εξαμήνου όμως, δημιουργούνται προβλήματα ως προς την διεξαγωγή των εγγραφών σε μαθήματα. Το κυριότερο πρόβλημα στο τμήμα Πληροφορικής είναι με τα μαθήματα περιορισμένης επιλογής. Από την ομάδα αυτή των

μαθημάτων, ο φοιτητής καλείται να παρακολουθήσει έξι (6) για να μπορέσει να αποκτήσει τον τίτλο σπουδών του με επιτυχία.

Σε αρκετές περιπτώσεις, πολλοί φοιτητές επιθυμούν να παρακολουθήσουν το ίδιο μάθημα, με αποτέλεσμα να μην υπάρχουν θέσεις για να ικανοποιήσουν όλους τους φοιτητές, και αρκετοί να αναγκάζονται να παρακολουθήσουν μαθήματα που να μην ανήκουν στο πλάνο τους – είτε γιατί δεν ενδιαφέρονται με αυτό τον τομέα, είτε γιατί πρέπει να παρακολουθήσουν συγκεκριμένο μάθημα για την απόκτηση εξειδίκευσης.

Η εξειδίκευση δεν είναι υποχρεωτική στο Τμήμα Πληροφορικής του Πανεπιστημίου Κύπρου, ωστόσο δίνεται η ευχέρειά στους φοιτητές να αποκτήσουν μία όταν έχουν παρακολουθήσει 3 μαθήματα περιορισμένης επιλογής από την ομάδα μαθημάτων για την απόκτηση της εξειδίκευσης και το θέμα της Ατομικής διπλωματικής τους εργασίας να είναι σχετικό με την εξειδίκευση.

Οι εξειδικεύσεις που είναι διαθέσιμες στο τμήμα είναι οι ακόλουθες:

- Εξειδίκευση Δικτύων Υπολογιστών
- Εξειδίκευση Θεμελιώσεων Πληροφορικής
- Εξειδίκευση Μεγάλα Δεδομένα και Διαδικτυακός Υπολογισμός
- Εξειδίκευση Υπολογισμός Πραγματικού Κόσμου
- Εξειδίκευση Τεχνητή Νοημοσύνη
- Εξειδίκευση Τεχνολογία Λογισμικού
- Εξειδίκευση Ψηφιακά και Ενσωματωμένα Συστήματα

1.2 Κίνητρο

Στόχος λοιπόν, είναι η ανάπτυξη μιας διαδικτυακής εφαρμογής στην οποία οι φοιτητές θα μπορούν να εκφράσουν το ενδιαφέρον τους για τα μαθήματα περιορισμένης επιλογής που προσφέρονται στο επόμενο εξάμηνο. Έτσι, η το τμήμα Πληροφορικής θα έχει την γνώση και την πληροφορία ως προς το ποιες είναι οι επιλογές και οι προτιμήσεις των φοιτητών, και θα μπορούν να προσδιορίσουν τα μαθήματα που θα προσφερθούν στο επόμενο εξάμηνο καθώς και τον αριθμό θέσεων στους φοιτητές, έτσι ώστε να

υλοποιήσουν τους σχεδιασμούς τους και να εγγραφούν στα μαθήματα περιορισμένης επιλογής της προτίμησής τους.

1.3 Δομή Διπλωματικής Εργασίας

Κεφάλαιο 1: Εισαγωγή

Σε αυτό το κεφάλαιο δίνεται μια σύντομη εισαγωγή του θέματος της Ατομικής Διπλωματικής Εργασίας, καθώς επίσης εξηγείται το κίνητρο και ο στόχος της εργασίας αυτής.

Κεφάλαιο 2: Ανάλυση Απαιτήσεων και Προδιαγραφών

Στο κεφάλαιο αυτό δίνεται μια λεπτομερής περιγραφή των απαιτήσεων και των προδιαγραφών του συστήματος με βάση την ανάγκη του. Στη συνέχεια δίνεται μια περιγραφή του, καθώς και κάποια σενάρια χρήσης του από τους χρήστες.

Κεφάλαιο 3: Προδιαγραφές Συστήματος

Εξηγούνται η αρχιτεκτονική και οι τεχνολογίες (εργαλεία, λογισμικό) που χρησιμοποιήθηκαν κατά την διάρκεια της εκπόνησης της εργασίας.

Κεφάλαιο 4: Υλοποίηση Συστήματος

Αναλύεται η δομή οργάνωσης του συστήματος για τα συστατικά του. Καθώς και η διαδικασία υλοποίησης.

Κεφάλαιο 5: Λειτουργίες Συστήματος

Παρουσιάζονται οι λειτουργίες του συστήματος και παρατίθενται εικόνες των σελίδων που υλοποιήθηκαν.

Κεφάλαιο 6: Συμπεράσματα

Δίνει ένα γενικό συμπέρασμα για το σύστημα που υλοποιήθηκε, πως θα βοηθήσει καθώς επίσης και πιθανό μελλοντικό έργο που μπορεί να γίνει.

Κεφάλαιο 2

Ανάλυση Απαιτήσεων και Προδιαγραφών

2.1 Εισαγωγή	4
2.2 Μεθοδολογία Ανάπτυξης	5
2.3 Προσδιορισμός Αναγκών και απαιτήσεων	6
2.3.1 Υποθέσεις – Σταθερές Συστήματος – Επεξήγηση	6
2.3.1 Προσδιορισμός Τυπικών Χρηστών	8
2.4 Ανάλυση Αναγκών	9
2.4.1. Σενάρια Χρήσης	9
2.5 Καθορισμός Απαιτήσεων	10
2.5.1 Λειτουργικές Απαιτήσεις	10
2.5.2. Μη Λειτουργικές Απαιτήσεις	10
2.6 Περιορισμοί	14

2.1 Εισαγωγή

Κάθε διαδικτυακή εφαρμογή (σύστημα) αποτελείται από 3 δομικά στοιχεία. Το πρώτο είναι ο εξυπηρετητής -υπολογιστής (1) ο οποίος δίνει την δυνατότητα στον χρήστη (2) να μπορεί να ολοκληρώσει μια εργασία – ανάγκη. Το τρίτο δομικό στοιχείο είναι η διεπαφή – interface (3) που δίνει στον χρήστη (2) να έρθει σε επικοινωνία με τον εξυπηρετητή (1) για να εκτελέσει την εργασία του.

Ο στόχος του συστήματος είναι να βοηθήσει τον χρήστη να φέρει εις πέρας την επιθυμία (ανάγκη) του. Γι' αυτό το λόγο οι δημιουργοί των συστημάτων - εφαρμογών πρέπει να υπολογίσουν και να δημιουργήσουν την διεπαφή βασιζόμενοι στα άλλα 2 δομικά στοιχεία. Από την μία πρέπει να κατανοήσουν τους περιορισμούς του συστήματος, και από την άλλη πρέπει να καταλάβουν τις ανάγκες που έχει ο χρήστης.

2.2 Μεθοδολογία Ανάπτυξης

Η μεθοδολογία ανάπτυξης (μοντέλο κύκλου ζωής) μιας τεχνολογικής εφαρμογής είναι ο τρόπος με τον οποίο θα καθοριστεί η ανάπτυξη της εφαρμογής. Μέσω της μεθοδολογίας δημιουργείται η σειρά με την οποία θα εκτελεστούν οι εργασίες κατά τον σχεδιασμό, την ανάπτυξη και συντήρηση της εφαρμογής.

Για την εκπόνηση της εργασίας αυτής αποφασίστηκε όπως ακολουθηθεί το μοντέλο κύκλου ζωής του καταρράκτη. Η μεθοδολογία αυτή χωρίζει τις φάσεις καθορισμού απαιτήσεων και ανάπτυξης, επικύρωσης και εξέλιξης. Αυτό έχει ως αποτέλεσμα να ξεκινά η κάθε φάση μετά το πέρας της προηγούμενης. Σε περίπτωση που χρειάζεται κάποια αλλαγή σε προηγούμενη φάση, τότε όλες οι ενδιάμεσες φάσεις πρέπει να ολοκληρωθούν ξανά, μέχρι να φτάσουμε στην τελική φάση.

Οι φάσεις του μοντέλου του καταρράκτη καθορίζονται ως εξής:

1. Καθορισμός Προβλήματος

Καταγράφεται το πρόβλημα το οποίο η ομάδα δημιουργίας λογισμικού (εφαρμογών) καλείται να επιλύσει.

2. Ανάλυση και καθορισμός απαιτήσεων

Καταγράφονται όλες οι ανάγκες και οι απαιτήσεις του τελικού χρήστη και επιπλέον καθορίζονται οι περιορισμοί και οι προδιαγραφές.

3. Σχεδίαση

Καθορίζονται η αρχιτεκτονική του λογισμικού, το υλικό, τα υλικά του συστήματος και οι σχέσεις μεταξύ τους.

4. Υλοποίηση

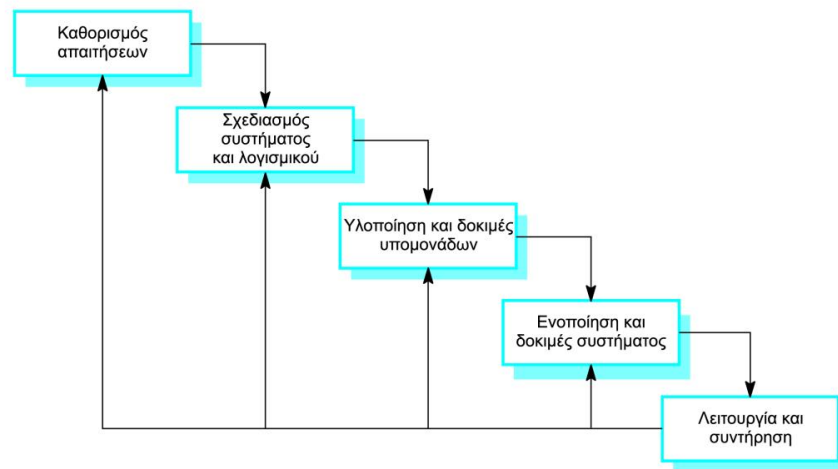
Σε αυτό το στάδιο γίνεται η συγγραφή κώδικα και η επαλήθευση.

5. Λειτουργία και συντήρηση

Εδώ διορθώνονται τυχόν σφάλματα που μπορεί να προκύψουν κατά την διάρκεια λειτουργίας του συστήματος, καθώς και η βελτιστοποίηση των υπηρεσιών του.

Στην περίπτωση της εργασίας αυτής, το πρόβλημα έχει οριστεί ως το θέμα που προκύπτει κάθε εξάμηνο κατά την διαδικασία εγγραφής στα μαθήματα.

Στο επόμενο υποκεφάλαιο θα αναλυθεί το στάδιο της ανάλυσης και του καθορισμού των απαιτήσεων.



Εικόνα 1. Μεθοδολογία ανάπτυξης καταρράκτη

2.3 Προσδιορισμός Αναγκών

2.3.1 Υποθέσεις – Σταθερές Συστήματος – Επεξήγηση

Στην παρούσα εργασία γίνονται κάποιες υποθέσεις έτσι ώστε να μπορεί να γίνει η ανάπτυξη του και η λειτουργία του. Το όριο του αριθμού προτιμήσεων ανά φοιτητή ορίζεται στις **3 επιλογές**, καθώς με βάση τον οδηγό σπουδών του τμήματος, ο μέγιστος αριθμός μαθήματος που καλούνται οι φοιτητές να εγγραφούν είναι 2 ανά εξάμηνο και αφορά τεταρτοετής φοιτητές. Επίσης, ο κάθε φοιτητής έχει μεγάλη πιθανότητα να εγγραφεί σε τουλάχιστον ένα από τα μαθήματα περιορισμένης επιλογής της προτίμησής του, με αποτέλεσμα να χρειάζεται ακόμη μια εναλλακτική λύση κατά τις εγγραφές μαθημάτων.

Επιπρόσθετα, στην παρούσα εργασία χρησιμοποιούνται όροι οι οποίοι χρήζουν την επεξήγηση για την καλύτερη κατανόηση της από τον αναγνώστη και τον χρήστη. Όροι όπως ζήτηση, πληρότητα και διαθέσιμες θέσεις εξηγούνται πιο κάτω.

1. Πληρότητα

Είναι η αναλογία των προτιμήσεων των φοιτητών ως προς τον αριθμό θέσεων που ορίζεται από το τμήμα Πληροφορικής για το κάθε μάθημα.

2. Ζήτηση

Ζήτηση ορίζεται ως ο αριθμός των φοιτητών που εκφράζουν την πρόθεσή τους να εγγραφούν στο μάθημα και καθορίζεται με βάση το ποσοστό πληρότητας. Επίσης σε παρένθεση δίνεται παράδειγμα μέγιστης πληρότητας σε μάθημα 25 ατόμων.

- a. Μάθημα με χαμηλή ζήτηση χαρακτηρίζεται ένα μάθημα του οποίου το ποσοστό πληρότητας του δεν υπερβαίνει το 50%. (12,5/25)
- b. Μάθημα με μέτρια ζήτηση χαρακτηρίζεται το μάθημα που το ποσοστό πληρότητας του υπερβαίνει το 50% και δεν υπερβαίνει το 80%. (20/25)
- c. Μάθημα με υψηλή/μεγάλη ζήτηση χαρακτηρίζεται το μάθημα του οποίου το ποσοστό πληρότητας είναι πάνω από 80% και κάτω από το 140%. (35/25)
- d. Μαθήματα των οποίων το ποσοστό πληρότητας τους είναι μεγαλύτερο του 140% είναι τα μαθήματα υπερβολικής ζήτησης και χρήζουν άμεσης λήψης μέτρων από το τμήμα. (36+/25)

3. Μέτρα που προτείνεται να ληφθούν

Με βάση την ζήτηση που ορίστηκε πιο πάνω, οι χειριστές του συστήματος θα λαμβάνουν προτάσεις για το κάθε μάθημα. Οι προτάσεις είναι οι εξής:

- Στα μαθήματα χαμηλής ζήτησης το τμήμα θα πρέπει να εξετάσει την αντικατάστασή του με κάποιο άλλο μάθημα που να έχει περισσότερο ενδιαφέρον από αυτό, καθώς κατά τις εγγραφές αν δεν μαζέψει τον ελάχιστο αριθμό φοιτητών που ορίζεται από το Πανεπιστήμιο θα πρέπει να ακυρωθεί.
- Σε μαθήματα με μέτρια ζήτηση δεν θα παρουσιάζεται κάποια πρόταση, καθώς εικάζεται ότι θα έχει περισσότερους φοιτητές από το ελάχιστο και θα πληρωθούν οι περισσότερες από τις θέσεις του μαθήματος.
- Τα μαθήματα με μεγάλη/υψηλή ζήτηση είναι αρκετά σημαντικά, διότι υπάρχει μεγάλη πιθανότητα να είναι πλήρης κατά τις εγγραφές και να μην δοθεί η ευκαιρία σε όλους τους φοιτητές να εγγραφούν. Με βάση την πληρότητα, το σύστημα θα εισηγείται την εξέταση της αύξησης των διαθέσιμων θέσεων κατά 5 θέσεις για παράδειγμα, έτσι ώστε να εγγραφούν περισσότεροι φοιτητές. Επίσης, όταν ένα μάθημα πλησιάζει την υπερβολική ζήτηση, δηλαδή είναι πάνω από 130% θα δίνεται η πρόταση δημιουργίας ενός ακόμα ακροατηρίου, αφού εικάζεται ότι με

αυτό το ποσοστό θα μπορέσουν να δημιουργηθούν 2 τμήματα με πάνω από τον ελάχιστο αριθμό φοιτητών.

- Σε μαθήματα με υπερβολική ζήτηση το σύστημα θα προτείνει την δημιουργία ενός ακόμα ακροατηρίου, αλλά θα συστήνει την προσφορά του εν λόγω μαθήματος και το επόμενο εξάμηνο.

Ο κύκλος της λειτουργίας του συστήματος είναι ο ακόλουθος:

1. Ο διαχειριστής ενεργοποιεί τα μαθήματα που θα προσφερθούν στο επόμενο εξάμηνο.
2. Για συγκεκριμένο χρονικό περιθώριο που θα καθοριστεί από το τμήμα Πληροφορικής.
3. Με το πέρας του περιθωρίου υποβολής των προτιμήσεων, ο διαχειριστής προβαίνει στην εξαγωγή των δεδομένων και την ανάλυσή τους.

2.3.2 Προσδιορισμός Τυπικών Χρηστών

1. Φοιτητές

Οι βασικοί χρήστες αυτής της εφαρμογής είναι οι φοιτητές. Αυτοί θα μπορούν να συνδέονται στο σύστημα, να βλέπουν τα μαθήματα που θα προσφερθούν για το επόμενο εξάμηνο και να καταχωρούν τις προτιμήσεις τους στα μαθήματα περιορισμένης επιλογής που θα επιθυμούσαν να παρακολουθήσουν. Επιπρόσθετα, οι φοιτητές θέλουν να λαμβάνουν και κάποιες προτάσεις μαθημάτων, έτσι ώστε να μπορούν να αποφασίσουν ευκολότερα ποια μαθήματα επιθυμούν/ θα πρέπει να παρακολουθήσουν το επόμενο εξάμηνο.

2. Διαχειριστές συστήματος

Οι διαχειριστές του συστήματος επιθυμούν να μάθουν αναλυτικά για τα μαθήματα που θα είναι διαθέσιμα για παρακολούθηση το επόμενο εξάμηνο. Επίσης, θα πρέπει να γνωρίζουν εάν υπάρχει κάποιο δημοφιλές μάθημα και θα πρέπει να αυξηθεί το μέγεθος των ακροατηρίων ή να προστεθεί ακόμη ένα ακροατήριο ή να διαθέσουν αυτό το μάθημα και τα δύο εξάμηνα, ούτως ώστε να ικανοποιηθεί η προτίμηση όσων περισσότερων φοιτητών γίνεται. Επιπρόσθετα, θα πρέπει να γνωρίζουν αν υπάρχει κάποιο μάθημα που δεν αντιμετωπίζει μεγάλο ενδιαφέρον, και ίσως χρειαστεί να μειώσουν τις θέσεις του

ακροατηρίου (άρα θα χρειαστεί μικρότερη αίθουσα διδασκαλίας) ή να σκεφτούν την περίπτωση αντικατάστασής του με κάποιο άλλο μάθημα, που ίσως έχει περισσότερη απήχηση. Στη συνέχεια, οι διαχειριστές πρέπει να μπορούν να ενεργοποιήσουν/απενεργοποιήσουν μαθήματα, και να κάνουν αλλαγές στο σύστημα. Τέλος, θα πρέπει να μπορούν να εξάγουν τα αποτελέσματα από το εν λόγω σύστημα.

2.4 Ανάλυση Αναγκών

Με βάση τα πιο πάνω, η ανάγκη που προκύπτει από το πρόβλημα αυτό, είναι η δημιουργία ενός συστήματος, μέσω του οποίου θα γίνει η προσπάθεια μερικής ή ολικής επίλυσής του.

Από την οπτική των φοιτητών το σύστημα θα πρέπει να μπορεί να δεχτεί τις προτιμήσεις τους και να μπορεί να τους δώσει τυχόν προτάσεις. Από την πλευρά των διαχειριστών της εφαρμογής, θα πρέπει να μπορεί να εξάγει αποτελέσματα για να τα χρησιμοποιήσουν και να προβούν σε δράσεις εφόσον χρειάζεται.

2.4.1 Σενάρια Χρήσης

Τα σενάρια χρήσης είναι ένας τρόπος χρήσης και σχεδίασης ενός συστήματος. Είναι «ιστορίες» που αφηγούνται μια εργασία ενός χρήστη που πρέπει να γίνεται από το σύστημα που πρόκειται να υλοποιηθεί. Αυτό βοηθάει στην καλύτερη ανάλυση των αναγκών αλλά και επίσης στην καλύτερη αξιολόγηση του αποτελέσματος.

Σενάριο 1: Φοιτητής – δεν επιθυμεί να ακολουθήσει εξειδίκευση

Ο χρήστης αυτός συνδέεται στο σύστημα και βλέπει την λίστα των μαθημάτων που θα προσφέρονται στο επόμενο εξάμηνο και επιλέγει τα μαθήματα.

Σενάριο 2: Φοιτητής που ενδιαφέρεται για προτάσεις με βάση κριτήρια

Ο χρήστης, συνδέεται στο σύστημα και μελετά όλες τις σελίδες προσεκτικά. Παρατηρεί τα μαθήματα που θα προσφερθούν και σκέφτεται ποια θα επιλέξει. Επιλέγει την εξειδίκευση που προτιμάει να εξασφαλίσει και πηγαίνοντας στην αρχική σελίδα, παρατηρεί τις προτάσεις μαθημάτων που του δίνει το σύστημα. Αφού έχει μελετήσει τις

επιλογές του και έχει αποφασίσει τις προτεραιότητες του, καταχωρεί τις προτιμήσεις του στο σύστημα.

Σενάριο 3: Ο Διαχειριστής του συστήματος

Ο διαχειριστής συνδέεται στο σύστημα και μεταβαίνει στην σελίδα των μαθημάτων και ενεργοποιεί τα μαθήματα που θα προσφερθούν το επόμενο εξάμηνο, και ελέγχει τον αριθμό αρχικών θέσεων του μαθήματος. Σε μετέπειτα χρόνο, συνδέεται και μεταβαίνει στην σελίδα όπου μπορεί να δει τις προτιμήσεις των μαθημάτων και κάποιες προτάσεις. Αφού λήξει η περίοδος υποβολής προτιμήσεων μεταβαίνει και απενεργοποιεί όλα τα μαθήματα που ήταν ενεργοποιημένα.

Σενάριο 4: Μεγάλη ζήτηση σε μάθημα – Διαχειριστής

Ο διαχειριστής συνδέεται στο σύστημα, μελετά τις προτιμήσεις των φοιτητών στα μαθήματα, παρατηρεί τις προτάσεις που το δίνει το σύστημα, και ενημερώνει τον αριθμό θέσεων ή λαμβάνει τα κατάλληλα μέτρα σε περίπτωση που χρειάζεται. Με το πέρας της περιόδου επιλογής από τους φοιτητές εξάγει τα αποτελέσματα για μελλοντική χρήση.

2.5 Καθορισμός απαιτήσεων

2.5.1 Λειτουργικές απαιτήσεις

Λειτουργικές απαιτήσεις είναι όλες οι δηλώσεις που σχετίζονται με την συμπεριφορά του συστήματος. Καθορίζει το τι πρέπει να κάνει το σύστημα έτσι ώστε να ανταποκριθεί στις ανάγκες και τις προσδοκίες του χρήστη.

Λειτουργικές Απαιτήσεις			
A/A	Απαίτηση	Τύπος Χρήστη	Περιγραφή
1.	Προβολή ενεργών μαθημάτων	Φοιτητής, Διαχειριστής	Ο χρήστης θα πρέπει να μπορεί να μεταβεί στην σελίδα των μαθημάτων και να δει όλα τα μαθήματα που είναι διαθέσιμα εκείνη την στιγμή

Λειτουργικές Απαιτήσεις			
A/A	Απαίτηση	Τύπος Χρήστη	Περιγραφή
2	Προβολή προτάσεων επιλογής μαθημάτων	Φοιτητής	Ο φοιτητής θα μπορεί να δει κάποιες προτάσεις σχετικά με τα διαθέσιμα μαθήματα που θα διατεθούν.
3	Να μπορεί να δει πόσες και ποιες ενεργές επιλογές έχει κάνει	Φοιτητής	Ο φοιτητής θα πρέπει να μπορεί να ενημερωθεί σχετικά με το πόσες ενεργές επιλογές έχει κάνει, καθώς και ποια μαθήματα επέλεξε.
4	Να μπορεί να δει τις εξειδικεύσεις και ποια μαθήματα ανήκουν σε κάθε μια	Φοιτητής	Ο χρήστης πρέπει να είναι σε θέση να δει τις εξειδικεύσεις και τα μαθήματα που υπάγονται στην κάθε μια.
5	Να καταχωρήσει προτίμηση για ένα μάθημα	Φοιτητής	Ο φοιτητής εφόσον δεν έχει ξεπεράσει το όριο των προτιμήσεων πρέπει να μπορεί να θέσει προτιμήσεις σε μαθήματα
6	Να ενημερωθεί ότι το μάθημα έχει υψηλή ζήτηση	Φοιτητής	Όταν ο φοιτητής προβεί να δει μάθημα του οποίου η ένδειξη ενδιαφέροντος πλησιάζει ή έχει υπερβεί τον αριθμό όλων των θέσεων (πληρότητα πάνω από 80%), η εφαρμογή θα παρουσιάζει μήνυμα που να ειδοποιεί τον φοιτητή ότι αυτό το μάθημα είναι αρκετά δημοφιλές και ότι αν επιλέξει αυτό το μάθημα υπάρχει περίπτωση να μην καταφέρει να εγγραφεί στις εγγραφές εξαμήνου.

Λειτουργικές Απαιτήσεις			
A/A	Απαίτηση	Τύπος Χρήστη	Περιγραφή
7	Να παρουσιάζονται εναλλακτικές προτάσεις σε μαθήματα με ζήτηση πέραν των διαθέσιμων θέσεων	Φοιτητής	Σε περίπτωση που έχει μεγαλύτερη απαίτηση από την πληρότητα του μαθήματος (εξέφρασαν περισσότεροι φοιτητές την προτίμηση τους από τις διαθέσιμες θέσεις), το σύστημα να προτείνει εναλλακτικά μαθήματα σε συναφές πεδίο.
8	Να ενημερώνονται όλοι οι φοιτητές ότι για το μάθημα που έχουν στη λίστα προτιμήσεων τους έχουν δείξει ενδιαφέρον αρκετοί φοιτητές	Φοιτητής	Όταν περισσότερο από 80% των διαθέσιμων θέσεων έχει καλυφθεί, ο φοιτητής που έχει το μάθημα ήδη στην λίστα προτιμήσεων του, να λαμβάνει ενημερωτικό μήνυμα ότι υπάρχει μεγάλη ζήτηση για το συγκεκριμένο μάθημα.
9	Ενεργοποίηση ή απενεργοποίηση μαθήματος	Διαχειριστής	Ο διαχειριστής πρέπει να μπορεί να ενεργοποιεί ή να απενεργοποιεί μαθήματα (ανάλογα με το ποια μαθήματα θα προσφερθούν το επόμενο εξάμηνο)
10	Να δει πόσες φορές έχει καταχωρηθεί στην λίστα προτιμήσεων ένα μάθημα καθώς και ποιοι το έχουν στην λίστα τους	Διαχειριστής	Ένας διαχειριστής θα έχει την δυνατότητα να δει ποιοι φοιτητές έχουν καταχωρήσει ένα μάθημα στην λίστα προτιμήσεών τους, τις πληροφορίες τους και συνολικά πόσες φορές βρίσκεται στις λίστες φοιτητών.

Λειτουργικές Απαιτήσεις			
A/A	Απαίτηση	Τύπος Χρήστη	Περιγραφή
11	Να λαμβάνει εβδομαδιαίο μήνυμα για την ενημέρωση της πληρότητας των διαθέσιμων μαθημάτων	Διαχειριστής	Κατά την διάρκεια που θα είναι διαθέσιμο το σύστημα στους φοιτητές, ο διαχειριστής να λαμβάνει ενημερωτικό εβδομαδιαίο email, το οποίο θα περιλαμβάνει τον αριθμό των φοιτητών που έδειξαν προτίμηση σε κάθε ενεργό μάθημα.
12	Να ενημερώνεται όταν ένα μάθημα πλησιάζει στην πληρότητα	Διαχειριστής	Όταν ένα μάθημα έχει καταχωρηθεί στην λίστα προτιμήσεων φοιτητών σε περισσότερο από το 80% των θέσεων του μαθήματος, ο διαχειριστής να λαμβάνει ειδοποίηση μέσω ηλεκτρονικού μηνύματος που να τον ενημερώνει για την ζήτηση του μαθήματος.
13	Να λαμβάνει προτάσεις με μέτρα που είναι πιθανών να πρέπει να λάβει σε περίπτωση μεγάλης ζήτησης	Διαχειριστής	Όταν ένα μάθημα έχει πλησιάζει ή έχει υπερβεί τον αριθμό θέσεων, ο διαχειριστής να λαμβάνει προτάσεις για μέτρα τα οποία μπορεί να λάβει ανάλογα με το ποσοστό ζήτησης.
14	Να μπορεί να εξάγει τα δεδομένα από τις προτιμήσεις των φοιτητών	Διαχειριστής	Ο διαχειριστής να μπορεί να εξάγει τα δεδομένα των προτιμήσεων των φοιτητών (π.χ. .xlsx) για μελλοντική χρήση.

2.5.2 Μη Λειτουργικές Απαιτήσεις

Οι μη λειτουργικές απαιτήσεις είναι οι περιορισμοί ή οι απαιτήσεις που αφορούν το σύστημα.

Μη Λειτουργικές Απαιτήσεις		
A/A	Απαίτηση	Περιγραφή
Απόδοση		
1	Ταχύτητα απόκρισης (response time) και εκτέλεσης	Οι λειτουργίες του συστήματος πρέπει να εκτελούνται χωρίς καθυστερήσεις, ανεξαρτήτως του φόρτου που έχει ο εξυπηρετητής (server) την κάθε στιγμή.
Ευχρηστία		
2	Εύκολο σύστημα προς χρήση για όλους τους χρήστες	Ο σχεδιασμός των διεπαφών πρέπει να είναι εύκολος και κατανοητός από τον χρήστη.
Αξιοπιστία		
3	Λειτουργία χωρίς σφάλματα που να επηρεάσουν την πλοήγηση του χρήστη	Το σύστημα χρειάζεται να λειτουργεί χωρίς σφάλματα, διότι αυτό θα επηρεάσει τον χρήστη και ίσως χαθούν δεδομένα.
Ασφάλεια		
4	Αποτροπή μη εξουσιοδοτημένης πρόσβασης	Η εφαρμογή πρέπει να αποτρέπει την μη εξουσιοδοτημένη πρόσβαση

2.6 Περιορισμοί

Σε όλα τα συστήματα υπάρχουν κάποιοι περιορισμοί, έτσι ώστε να εξασφαλιστεί η εγκυρότητα των περιεχομένων του. Καταρχάς, για την ορθή λειτουργία του πρέπει να υπάρχει έλεγχος πρόσβασης – σύνδεση στο σύστημα (login), ούτως ώστε να αποτρέπεται η ανεπιθύμητη χρήση του συστήματος από αγνώστους που δεν είναι εξουσιοδοτημένοι. Ακόμη, πρέπει να περιορίζει τις λειτουργίες με βάση τον ρόλο του κάθε χρήστη, για να εξασφαλιστεί η ορθή λειτουργία του συστήματος. Τέλος, για την λειτουργία του συστήματος χρειάζεται μια βάση δεδομένων και ένας εξυπηρετητής για να χειρίζεται τα αιτήματα (requests) των χρηστών.

Κεφάλαιο 3

Προδιαγραφές Συστήματος

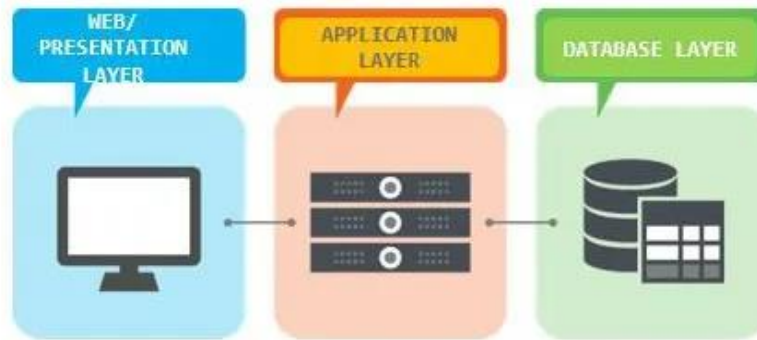
3.1 Αρχιτεκτονική	15
3.2 Τεχνολογίες και εργαλεία	16
3.2.1 Βάση Δεδομένων MongoDB	16
3.2.2 Node.js	17
3.2.3 ExpressJS	17
3.2.4 NPM – Node Package Manager	17
3.2.5 CSS – Cascading Style Sheets	18
3.2.6 MS Visual Studio Code	18

3.1 Αρχιτεκτονική

Η αρχιτεκτονική του Συστήματος Προεγγραφής Φοιτητών σε μαθήματα Περιορισμένης Επιλογής είναι βασισμένη στην αρχιτεκτονική 3-tier client-server architecture, όπως φαίνεται και στην Εικόνα 2. Στα αριστερά είναι το επίπεδο του χρήστη, ο οποίος αποκτά πρόσβαση στην σελίδα μέσω του φυλλομετρητή (Web Browser - παρ. Google Chrome, Mozilla Firefox, Safari) κάποιας συσκευής του (υπολογιστής, κινητό τηλέφωνο κ.α.), αλλά και το επίπεδο του εξυπηρετητή (web server) ο οποίος λαμβάνει και χειρίζεται τα αιτήματα του πελάτη. Επίσης ο server παραμένει συνέχεια ενεργός και «ακούει» σε συγκεκριμένη πόρτα (port) για να μπορεί να εξυπηρετήσει τον client οποιαδήποτε στιγμή.

Στη συνέχεια υπάρχει το application layer το οποίο είναι το API (application process interface) και η δουλειά του είναι να επεξεργάζεται τα αιτήματα που στέλνει ο client και να δίνει στον server απάντηση την οποία ο εξυπηρετητής θα επιστρέψει πίσω στον πελάτη.

Τέλος, βρίσκεται το database layer στο οποίο υπάρχουν όλα τα δεδομένα της εφαρμογής αποθηκευμένα. Το application layer ζητάει από την βάση δεδομένων πληροφορίες που υπάρχουν αποθηκευμένα για να τα επεξεργαστεί και να αποστείλει απάντηση στον client.



Εικόνα 2. 3-tier client-server architecture

3.2 Τεχνολογίες και Εργαλεία

3.2.1 MongoDB

Η MongoDB είναι μια μη-σχεσιακή βάση δεδομένων ανοικτού κώδικα, η οποία αποθηκεύει τα δεδομένα σε μορφή αρχείων JSON. Αποτελεί μια ελαφριά βάση δεδομένων η οποία δεν χρειάζεται μεγάλη προετοιμασία για να αποθηκεύσει τα δεδομένα. Αντιπροσωπεύει το database layer από το προηγούμενο υποκεφάλαιο. Η δομή της Βάσης Δεδομένων από κάτω προς τα πάνω είναι:



1. Fields (πεδία)

Αποτελούν την πληροφορία που αποθηκεύουμε σε αυτή.

2. Documents (αρχεία)

Αποτελούνται από μια ομάδα πεδίων (αποθηκεύονται σε μορφή αρχείου JSON και αντιστοιχεί στις γραμμές (rows) σε μία σχεσιακή βάση δεδομένων).

3. Collections (συλλογές)

Αποτελεί την συλλογή των αρχείων με το ίδιο θέμα (αντίστοιχο με τα tables σε σχεσιακές βάσεις δεδομένων).

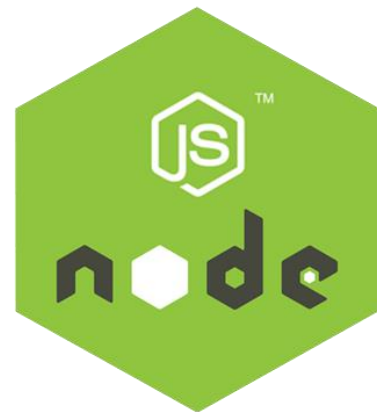
Σημ. Μαζί με την εγκατάσταση της βάσης δεδομένων γίνεται και η εγκατάσταση της εφαρμογής MongoDB Compass η οποία δίνει την δυνατότητα στον χρήστη να παρατηρεί την βάση δεδομένων και να κάνει μερικές λειτουργίες χωρίς την χρήση της Γραμμής Εντολής.

3.2.2. Node.JS

Το Node.JS είναι ένα διαπλατφορμικό περιβάλλον εκτέλεσης ανοικτού κώδικα (open source cross-platform runtime environment) το οποίο αποτελεί το κομμάτι του εξυπηρετητή στο web layer και ολόκληρο το application layer από την αρχιτεκτονική που αναλύθηκε στο προηγούμενο υποκεφάλαιο.

Είναι χτισμένο σε περιβάλλον JavaScript το οποίο εκτελείται αποκλειστικά από την πλευρά του εξυπηρετητή. Με την χρήση πολλαπλών JavaScript αρχείων επιτυγχάνουμε την ομαλή λειτουργία ενός web server και ενός api.

Για την σωστή χρήση του Node.js χρειάζεται να εγκατασταθεί και το npm (Node package manager) – θα αναλυθεί στην πορεία.



3.2.3 ExpressJS

Το ExpressJS είναι ένα web application framework ανοικτού κώδικα το οποίο χρησιμοποιείται από το Node.JS σχεδόν συνέχεια στις διαδικτυακές εφαρμογές. Παρέχει την δυνατότητα διαχείρισης HTTP server, και ο σκοπός του είναι να διευκολύνει την κατασκευή εφαρμογών ιστού, σελίδων, υπηρεσιών REST.

3.2.4 NPM – Node Package Manager

Ο Node Package Manager είναι ένα software registry το οποίο περιέχει πάνω από 800.000 πακέτα. Εγκαθίσταται αυτόματα με την εγκατάσταση του Node.JS και εξυπηρετεί στο να εγκαταστήσει πακέτα τα οποία θα χρησιμοποιηθούν στην εφαρμογή.

Τα σημαντικότερα πακέτα – βιβλιοθήκες που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας αυτής είναι:

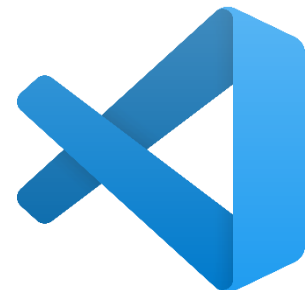
- ExpressJS – επεξηγήθηκε στο 3.2.3
- Mongoose – δημιουργεί σύνδεση μεταξύ του Node.JS και της βάσης δεδομένων MongoDB
- Pug JS – είναι μια μηχανή προτύπων σχεδιασμένη να μετατρέπει δεδομένα και να αποστέλλεται στον πελάτη ένα αρχείο HTML για την απεικόνιση της ιστοσελίδας. Εδώ η οργάνωση των περιεχομένων των tags γίνεται με την χρήση εσοχών στον κώδικα.

3.2.5 CSS – Cascading Style Sheets

Το CSS περιγράφει τον τρόπο που θα απεικονιστούν τα αρχεία HTML στην οθόνη του πελάτη.

3.2.6. MS Visual Studio Code

Το Visual Studio Code είναι ένα εργαλείο ανάπτυξης κώδικα το οποίο δημιουργήθηκε από την Microsoft. Προσφέρει ένα ευρύ φάσμα επιλογών συγγραφής προγραμματιστικών γλωσσών – και μη – αλλά επίσης διευκολύνει τον προγραμματιστή στο να εντοπίσει τυχόν λάθη στον κώδικα του.



Κεφάλαιο 4

Υλοποίηση Συστήματος

4.1 Βάση Δεδομένων	19
4.2 API & Web Server	21
4.2.1 Οργάνωση Εργασίας Node. JS	21
4.2.2 View engine	23

4.1 Βάση Δεδομένων

Για την υλοποίηση της εφαρμογής δημιουργήθηκε μία MongoDB βάση δεδομένων, η οποία αποτελείται από τις ακόλουθες συλλογές (collections):

- Courses
- People

Η συλλογή Courses περιέχει τις πληροφορίες που χρειάζονται και ανάλογα με τον τύπο του κάθε χρήστη, παρουσιάζονται τα ανάλογα πεδία. Τα πεδία που υπάρχουν στη συλλογή αυτή είναι τα ακόλουθα:

- courseID: Sting
- course_name: String
- semester_offered: Array
- credits: String
- Objectives: String
- Seats: Integer Number
- avail_seats: Integer Number
- active: Boolean

Για κάθε έγγραφο που εισάγεται στην βάση δεδομένων, όπως σε κάθε βάση, δημιουργείται ένα `_id` το οποίο περιέχει το μοναδικό αναγνωριστικό του συγκεκριμένου δεδομένου.

```

_id: ObjectId('64753edb95fb14faf7a2b8b5')
courseID: "ΕΠΛ344"
course_name: "Internet Technologies"
▼ semester_offered: Array
  0: "Χειμερινό"
  1: "Εαρινό"
credits: "7,5"
Objectives: "Introduction to Internet and the World-Wide Web Technologies. Emphasis..."
seats: 25
avail_seats: 24
active: true

```

Εικόνα 3. παράδειγμα εγγραφου στην συλλογή Courses

Η συλλογή People περιέχει τα ακόλουθα πεδία:

- userID: String
- username: String
- first_name: String
- last_name: String
- type: String
- preferences: Array
- password: String
- sp_pref: String
- entry_year: Number

```

_id: ObjectId('64750d0195fb14faf7a2b8a7')
userID:
username: "canast07"
first_name: "Κωνσταντίνος"
last_name: "Αναστασίου"
type: "student"
▼ preferences: Array
  0: "ΕΠΛ421"
  1: "ΕΠΛ433"
password:
sp_pref: "Εξειδίκευση Τεχνητή Νοημοσύνη"
entry_year: 2017

```

Εικόνα 4. Παράδειγμα εγγράφου στην συλλογή People

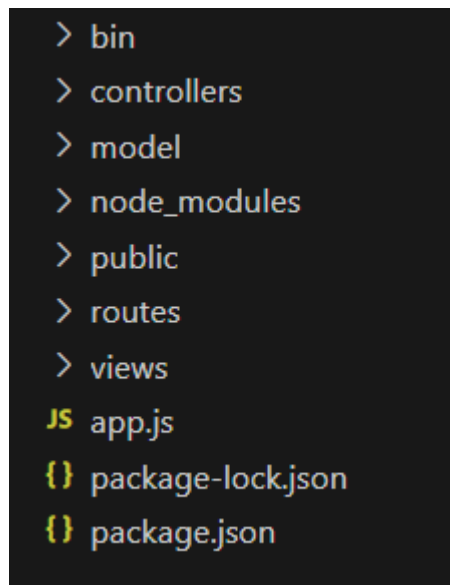
Σε αυτά τα δεδομένα εκτελούνται διάφορα queries από το api για να μπορέσουν να απαντηθούν τα αιτήματα του πελάτη.

4.2 API & Web Server

4.2.1 Οργάνωση Εργασίας Node. JS

Όπως εξηγήθηκε στο προηγούμενο κεφάλαιο, στο περιβάλλον NodeJS υπάρχουν όλα τα μέρη που δεν εκτελούνται στη συσκευή του πελάτη – πλην του database layer. Η οργάνωση του, όπως φαίνεται και στην εικόνα 5 αναλύεται ως ακολούθως:

Το αρχείο package.json και package-lock.json αποτελούν τα αρχεία που περιέχουν τις αναγκαίες πληροφορίες για να ξεκινήσει το πρόγραμμα. Με την εκτέλεση της εντολής `npm start` στην Γραμμή Εντολών το αρχείο package.json περιέχει την πληροφορία και δίνει εντολή να ξεκινήσει η εκτέλεση του αρχείου `www` που βρίσκεται μέσα στο φάκελο `bin`. Μέσα σε αυτό το αρχείο γίνεται ο ορισμός του server, και καθορίζεται η πόρτα που ακούει, έτσι ώστε να μπορεί να λαμβάνει τα αιτήματα από τον client. Επίσης, στο αρχείο αυτό καθορίζεται και ποιο αρχείο εκτελεί τον ρόλο του api και περιέχει τις αναγκαίες εντολές για την εκτέλεση της εφαρμογής.



Εικόνα 5. Οργάνωση του Node.JS project

Όταν ο πελάτης αποστέλλει κάποιο αίτημα, μέσα από το αρχείο `app.js` θα αναλυθεί το αίτημα και θα εκτελεστούν οι ανάλογες εντολές. Για παράδειγμα αν έρθει αίτημα ‘GET’ το αίτημα αυτό θα γίνει κατευθυνθεί στην κατάλληλη συνάρτηση (function) και θα

εκτελέσει τα κατάλληλα api calls για να δοθεί η τελική εντολή render η οποία μετατρέπει τα . pug αρχεία σε κώδικα HTML και το αποστέλνει στον φυλλομετρητή του χρήστη.

Σημαντικό κομμάτι στην λειτουργία της εφαρμογής είναι η επικοινωνία του application layer με τον database layer – δηλαδή, η επικοινωνία της εφαρμογής με την βάση δεδομένων. Για να επιτευχθεί αυτό χρησιμοποιήθηκε το πακέτο Mongoose το οποίο περιέχει τις απαραίτητες εντολές και διαδικασίες για την σωστή επικοινωνία των δύο συστατικών, συνήθως μέσω της πόρτας 27017.

Η σε κάθε query που γίνεται στην βάση δεδομένων επιστρέφεται ένα αρχείο JSON το οποίο επεξεργάζεται με την βοήθεια της βιβλιοθήκης mongoose. Για να μπορούν να σταλούν τα queries στην βάση δεδομένων με το σωστό collection, χρειάζεται να καθορίσουμε τα μοντέλα, τα οποία βρίσκονται στον φάκελο models. Με λίγα λόγια, δημιουργείται ένα schema το οποίο καθορίζει τα ονόματα και τον τύπο των πεδίων και στην συνέχεια τα χρησιμοποιεί ως μοντέλα για να σταλεί το query στην βάση δεδομένων στο collection.

<pre>//People model + Schema const peopleSchema = new mongoose.Schema({ userID: String, username: String, first_name: String, last_name: String, type: String, preferences: Array, password: String, sp_pref: String, entry_year: Number }); const People = mongoose.model('People', peopleSchema, 'people');</pre>	<pre>//Course model + Schema const courseSchema = new mongoose.Schema({ courseID: String, course_name: String, semester_offered: Array, credits: String, Objectives: String, seats: Number, avail_seats: Number, active: Boolean }); const Course = mongoose.model('Course', courseSchema);</pre>
<pre>//Specialisation model + Schema const spSchema = new mongoose.Schema({ sp_name: String, requirements: Array }); const SP = mongoose.model('SP', spSchema, 'specialisations');</pre>	

4.2.2 View engine

Το μέρος της απεικόνισης της απάντησης του server και η αποστολή των HTML αρχείων χειρίζεται από το view engine, το οποίο αποτελείται αρχεία ενός πρότυπου αρχείου (template). Υπάρχουν διάφοροι τύποι αρχείων πρωτοτύπων, και για την εργασία αυτή επιλέχθηκε ο τύπος αρχείου .pug. Τα αρχεία αυτά έχουν την ίδια μορφή με ένα HTML αρχείο, με την διαφορά ότι δεν χρειάζονται να περιλαμβάνουμε τα tags κλεισίματος (π.χ. `</div>`), καθώς αυτό καθορίζεται από τις εσοχές του κειμένου. Για παράδειγμα, ο παρακάτω κώδικας HTML

```
<div>
  <p> Κείμενο</p>
</div>
```

Θα γραφτεί με τον ακόλουθο τρόπο σε ένα αρχείο .pug.

```
div
  p Κείμενο
```

Κεφάλαιο 5

Λειτουργίες Συστήματος

5.1 Φοιτητής	24
5.1.1 Σύνδεση	24
5.1.2 Αρχική Σελίδα	25
5.1.3 Πλοήγηση μαθημάτων	26
5.1.4 Σελίδα Μαθήματος	27
5.1.5 Εξειδικεύσεις	28
5.1.6 Ρυθμίσεις λογαριασμού	28
5.2 Διαχειριστής – Admin	29
5.2.1 Αρχική Σελίδα Διαχειριστή	29
5.2.2 Μαθήματα ανά Εξάμηνο	30
5.2.3 Μαθήματα – Admin	30
5.2.4 Όλα τα μαθήματα	31
5.2.5 Σελίδα μαθήματος – Admin	31

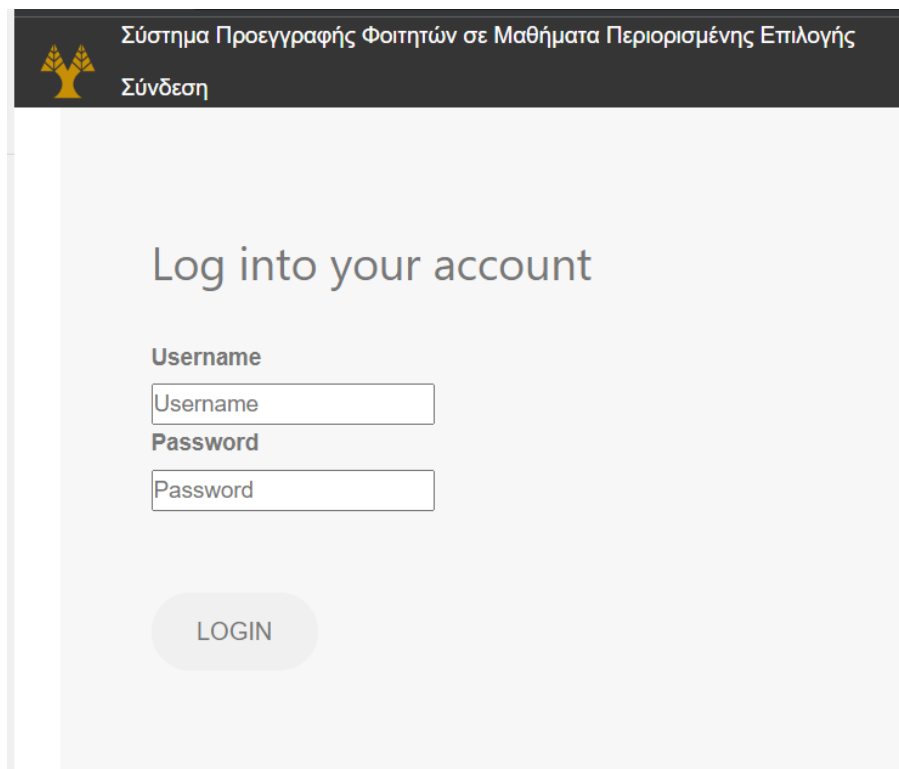
5.1 Φοιτητής

Για τους 2 τύπους χρηστών υπάρχουν ξεχωριστές λειτουργίες που μπορούν να εκτελέσουν. Ο φοιτητής αφού συνδεθεί έχει την ευκαιρία να δει τις προσωπικές του πληροφορίες, να επιλέξει συγκεκριμένη κατεύθυνση που προτιμάει, να ακολουθήσει και να θέσει προτιμήσεις σε μαθήματα που επιθυμεί να εγγραφεί.

5.1.1 Σύνδεση

Ο κάθε χρήστης που επιθυμεί να χρησιμοποιήσει την πλατφόρμα αυτή, χρειάζεται να συνδεθεί με τον λογαριασμό του. Για τον λόγο αυτό, η πρώτη οθόνη που βλέπει κάποιος

όταν εισάγει το link της εφαρμογής, είναι η οθόνη σύνδεσης, στην οποία ο χρήστης καλείται να εισάγει το username του και τον κωδικό του. Σε περίπτωση που εισάγει λάθος δεδομένα, γίνεται ανακατεύθυνση σε σελίδα που αναφέρει ότι τα στοιχεία εισόδου είναι λανθασμένα, διαφορετικά ανάλογα με τον τύπο του (φοιτητής ή διαχειριστής) κατευθύνεται στην κατάλληλη Αρχική Σελίδα.



The image shows a web application interface for logging in. At the top, there is a dark header bar with a yellow tree icon on the left and the text 'Σύστημα Προεγγραφής Φοιτητών σε Μαθήματα Περιορισμένης Επιλογής' and 'Σύνδεση' on the right. Below the header, the main content area has a light gray background. It features the heading 'Log into your account' in a large, dark font. Underneath, there are two input fields: 'Username' and 'Password', each with a label above it. Below these fields is a rounded rectangular button labeled 'LOGIN'.

Εικόνα 6. Σελίδα Σύνδεσης

5.1.2 Αρχική Σελίδα

Με την επιτυχημένη εισαγωγή στοιχείων, ο χρήστης κατευθύνεται στην αρχική σελίδα του, στην οποία μπορεί να δει το username του, τον αριθμό των προτιμήσεων που έχει καταχωρήσει, καθώς και μερικές επιλογές μαθημάτων που θα μπορούσε να επιλέξει. Οι προτάσεις των μαθημάτων γίνονται με βάση την κατεύθυνση που δήλωσε ο φοιτητής ότι προτιμάει να ακολουθήσει. Τα μαθήματα σε αυτή την λίστα είναι υπερσύνδεσμοι και παραπέμπουν τον χρήστη στην σελίδα του μαθήματος.



Καλώς Ορίσατε canast07

Εδώ μπορείτε να θέσετε τις προτιμήσεις σας για τα μαθήματα περιορισμένης επιλογής.

Αριθμός καταχωρημένων προτιμήσεων: 2

Προτάσεις επιλογής Μαθημάτων

Με βάση την καταχωρημένη σας προτίμηση εξειδίκευσης σκεφτείτε τα ακόλουθα μαθήματα:

ΕΠΛ412 - Τεστ 412

ΕΠΛ433 - Τεστ 433

ΕΠΛ434 - Τεστ 434

ΕΠΛ442 - Τεστ 442

ΕΠΛ444 - Τεστ 444

ΕΠΛ445 - Τεστ 445

ΕΠΛ447 - Τεστ 447

ΕΠΛ448 - Τεστ 448

Εικόνα 7. Αρχική Σελίδα Φοιτητή

4.1.3 Πλοήγηση Μαθημάτων

Από την μπάρα πλοήγησης στο πάνω μέρος της οθόνης του φοιτητή μπορεί να πατήσει στην επιλογή **Μαθήματα**, και κατευθύνει στην σελίδα που ο χρήστης μπορεί να βρει τα μαθήματα που θα είναι διαθέσιμα προς παρακολούθηση. Η κάθε επιλογή από την λίστα αποτελεί σύνδεσμο και στέλνει τον χρήστη στην σελίδα του κάθε μαθήματος ανάλογα.



Μαθήματα Περιορισμένης Επιλογής

ΕΠΛ344 - Internet Technologies
ΕΠΛ421 - Προγραμματισμός Συστημάτων
ΕΠΛ433 - Τεστ 433
ΕΠΛ434 - Τεστ 434
ΕΠΛ442 - Τεστ 442
ΕΠΛ444 - Τεστ 444
ΕΠΛ445 - Τεστ 445
ΕΠΛ447 - Τεστ 447
ΕΠΛ448 - Τεστ 448

Εικόνα 8. Σελίδα διαθέσιμων μαθημάτων περιορισμένης επιλογής

4.1.4 Σελίδα Μαθημάτων

Όταν ο χρήστης επιλέξει τον σύνδεσμο οποιουδήποτε μαθήματος, μεταφέρεται στην σελίδα με τις πληροφορίες του μαθήματος, στην οποία μπορεί να βρει πληροφορίες για το μάθημα. Επιπρόσθετα, σε αυτή τη σελίδα ο χρήστης έχει την δυνατότητα καταχώρησης/κατάργησης της προτίμησής του στο συγκεκριμένο μάθημα. Αν ο χρήστης δεν έχει υπερβεί το όριο των 3 επιλογών μαθημάτων, τότε θα μπορεί να δει το κουμπί Καταχώρηση προτίμησης για να επιλέξει το μάθημα, ή Κατάργηση Επιλογής στην περίπτωση που θέλει να αφαιρέσει το μάθημα από την λίστα επιλογών του.

Σύστημα Προεγγραφής Φοιτητών σε Μαθήματα Περιορισμένης Επιλογής

Αρχική Μαθήματα Εξειδικεύσεις Logout Ο λογαριασμός μου

Πίσω

Κατάργηση Προτίμησης

ΕΠΛ344 - Internet Technologies

ΕΠΛ344 - Internet Technologies

Πιστωτικές Μονάδες: 7,5

Εξάμηνο: Χειμερινό, Εαρινό

Στόχοι:

Introduction to Internet and the World-Wide Web Technologies. Emphasis on WWW protocols and programming of Internet Systems and Services. Content: Internet Fundamentals, Principles of Hypertext, HTML and Mark-up Languages, Cascading Style Sheets, Page Layouts, Web Design & Development, Dynamic content and HTML Forms, Client-Side Programming, Introduction to Web2.0 principles and AJAX.

Εικόνα 9. Σελίδα Μαθήματος

5.1.5 Εξειδικεύσεις

Με την επιλογή του **Εξειδικεύσεις** από το μενού πλοήγησης, ο χρήστης μεταβαίνει στη στατική σελίδα, όπου περιέχονται οι ειδικεύσεις που παρέχονται από το τμήμα, καθώς και τα μαθήματα που 3 από εκείνα πρέπει ο φοιτητής να παρακολουθήσει για να λάβει την εξειδίκευση.



Σύστημα Προεγγραφής Φοιτητών σε Μαθήματα Περιορισμένης Επιλογής
Αρχική Μαθήματα ΕξειδικεύσειςLogoutΟ λογαριασμός μου

Εξειδικεύσεις

<Εξειδίκευση Δικτύων Υπολογιστών>
Μαθήματα :
ΕΠΛ344
ΕΠΛ421
ΕΠΛ422
ΕΠΛ423
ΕΠΛ427
ΕΠΛ428
ΕΠΛ432
ΕΠΛ450

<Εξειδίκευση Θεμελιώσεων Πληροφορικής>
Μαθήματα :
ΕΠΛ412
ΕΠΛ414
ΕΠΛ431
ΕΠΛ432
ΕΠΛ433

<Εξειδίκευση Μεγάλα Δεδομένα και Διαδικτυακός Υπολογισμός>
Μαθήματα :

Εικόνα 10. Σελίδα Προβολής Εξειδικεύσεων

5.1.6 Ρυθμίσεις Λογαριασμού

Αν ο φοιτητής πατήσει την επιλογή **Ο λογαριασμός μου** τότε θα δει την σελίδα με τα στοιχεία του και πληροφορίες της φοίτησης του (Εξάμηνο φοίτησης, Προτιμώμενη Εξειδίκευση, Καταχωρημένες Προτιμήσεις μαθημάτων). Επίσης του δίνεται η ευκαιρία να μπορεί να επιλέξει από το μενού άλλη εξειδίκευση και έτσι θα μπορεί να δει άλλες προτάσεις στην αρχική σελίδα του.



Ο λογαριασμός μου

Κωνσταντίνος Αναστασίου

Εξάμηνο: 12

Προτίμηση Εξειδίκευσης: Εξειδίκευση Τεχνητή Νοημοσύνη

Προτιμήσεις Μαθημάτων

ΕΠΛ344 - Internet Technologies

ΕΠΛ421 - Προγραμματισμός Συστημάτων

Επιλέξτε προτίμηση εξειδίκευσης:

Εξειδίκευση Δικτύων Υπολογιστών

ΕΠΙΛΟΓΗ

Εικόνα 11. Σελίδα Λογαριασμού Χρήστη

5.2 Διαχειριστής – Admin

Οι διαχειριστές του συστήματος έχουν διαφορετικές λειτουργίες, καθώς ο ρόλος τους είναι να ενεργοποιήσουν τα μαθήματα που θα προσφέρονται το επόμενο εξάμηνο και να δουν τα αποτελέσματα προτιμήσεων των φοιτητών. Όμως, όπως και οι φοιτητές, για να προβούν σε αυτές τις λειτουργίες, πρέπει πρώτα να συνδεθούν με τον λογαριασμό τους.

5.2.1 Αρχική Σελίδα Διαχειριστή

Η αρχική σελίδα του διαχειριστή έχει διαφορετικές λειτουργίες, αφού οι διαχειριστές δεν χρειάζονται προτάσεις επιλογής μαθημάτων. Στην δική τους αρχική σελίδα, έχουν την επιλογή να αναζητήσουν μαθήματα με βάση το εξάμηνο που διατίθενται (με βάση τον προγραμματισμό και τον οδηγό σπουδών του τμήματος).

Καλώς Ορίσατε -Admin

Αναζήτηση ανά εξάμηνο

Επιλέξτε Εξάμηνο: Χειμερινό ▾

ΕΠΙΛΟΓΗ

Εικόνα 12 Αρχική Σελίδα Διαχειριστή

5.2.2 Μαθήματα ανά Εξάμηνο

Αυτή η σελίδα είναι διαθέσιμη στους διαχειριστές μόνο όταν χρησιμοποιήσουν την αναζήτηση ανά εξάμηνο στην αρχική τους σελίδα. Μέσα από αυτή, μπορούν να δουν όλα τα μαθήματα, ανεξάρτητα από την ιδιότητα τους (activated / deactivated). Αυτή η σελίδα τους δίνει την ευχέρεια να αναζητήσουν ευκολότερα τα μαθήματα που θα ενεργοποιήσουν / απενεργοποιήσουν κάποιο εξάμηνο.

Σύστημα Προεγγραφής Φοιτητών σε Μαθήματα Περιορισμένης Επιλογής -Admin Αρχική Μαθήματα - Admin Όλα τα Μαθήματα Logout
Μαθήματα Εξαμήνου Επιλογής
ΕΠΛ344 - Internet Technologies
ΕΠΛ412 - Τεστ 412
ΕΠΛ434 - Τεστ 434
ΕΠΛ442 - Τεστ 442
ΕΠΛ444 - Τεστ 444
ΕΠΛ445 - Τεστ 445
ΕΠΛ447 - Τεστ 447
ΕΠΛ448 - Τεστ 448

Εικόνα 13. Μαθήματα ανά εξάμηνο Αναζήτησης

5.2.3 Μαθήματα – Admin

Με την επιλογή του **Μαθήματα – Admin** ο administrator έχει την δυνατότητα να δουν τις πληροφορίες από τις επιλογές των μαθημάτων από τους φοιτητές. Εδώ, μπορούν να δουν τα μαθήματα και το σύνολο των προτιμήσεων ανά μάθημα.

Σύνολο Προτιμήσεων - Μαθήματα Περιορισμένης Επιλογής

Μάθημα	Σύνολο Προτιμήσεων
ΕΠΛ344 - Internet Technologies	2
ΕΠΛ421 - Προγραμματισμός Συστημάτων	1

Εικόνα 14. Σελίδα Μαθήματα - Admin

5.2.4 Όλα τα Μαθήματα

Στη σελίδα αυτή δίνεται η ευχέρεια στον χρήστη να δει όλα τα μαθήματα που προσφέρονται στο τμήμα.

Μαθήματα

Μάθημα
ΕΠΛ344 - Internet Technologies
ΕΠΛ421 - Προγραμματισμός Συστημάτων
ΕΠΛ412 - Τεστ 412
ΕΠΛ433 - Τεστ 433

Εικόνα 15. Σελίδα Όλα τα μαθήματα

5.2.5. Σελίδα Μαθήματος – Admin

Με την σελίδα αυτή, ο διαχειριστής έχει την δυνατότητα να δει τις πληροφορίες του συγκεκριμένου μαθήματος, καθώς επίσης και τις πληροφορίες των φοιτητών που επιθυμούν να εγγραφούν στο μάθημα. Επιπρόσθετα από τις πληροφορίες μαθήματος που βλέπει ο φοιτητής, ένας διαχειριστής βλέπει τις θέσεις του μαθήματος, και τις διαθέσιμες θέσεις, ανάλογα με τις προτιμήσεις των φοιτητών, και κατά πόσο το μάθημα είναι ενεργό ή όχι. Στις προτιμήσεις προβάλλεται το όνομα του φοιτητή, το username του και το τρέχον εξάμηνο φοιτητή. Αυτό μπορεί να βοηθήσει τους διαχειριστές να βάλουν κάποιες

εξαιρέσεις ούτως ώστε φοιτητές που είναι κοντά στην περάτωση των σπουδών τους ή κοντά στο όριο των 12 εξαμήνων φοίτησης, να μπορούν να εγγραφούν σημαντικά για εκείνους μαθήματα. Ακόμη, ο administrator μπορεί να ενεργοποιήσει / απενεργοποιήσει μαθήματα.



Σύστημα Προεγγραφής Φοιτητών σε Μαθήματα Περιορισμένης Επίλογής - Admin

Αρχική Μαθήματα - Admin Όλα τα Μαθήματα Logout

Πίσω

Απενεργοποίηση

ΕΠΛ421 - Προγραμματισμός Συστημάτων

ΕΠΛ421 - Προγραμματισμός Συστημάτων

Πιστωτικές Μονάδες: 7,5

Εξάμηνο: Εαρινό

Στόχοι:

Διδασκαλία βασικών και προχωρημένων εννοιών προγραμματισμού και διαχείρισης λειτουργικών Συστημάτων (ΛΣ). Κατανόηση των μηχανισμών με τους οποίους επιτυγχάνεται η πρόσβαση στις ρουτίνες (βιβλιοθήκες - μονάδες) ενός ΛΣ με τη χρήση γλωσσών προγραμματισμού και δημοφιλών κελυφών. Για την πρακτική εξάσκηση και την υλοποίηση προγραμμάτων, θα χρησιμοποιηθεί η γλώσσα προγραμματισμού C και το Bash στο ΛΣ Linux. Οι διδασκόμενες έννοιες έχουν εφαρμογή σε όλη την οικογένεια των ΛΣ UNIX (π.χ., Mac OSX, HP-UX, AIX, Solaris, Android, iOS, Raspbian) αλλά και των Windows (π.χ., Powershell και αντίστοιχες κλήσεις συστήματος). Περιεχόμενο: i) Προχωρημένες εντολές του ΛΣ UNIX για διαχειριστές: φίλτρα με κανονικές εκφράσεις και εργαλεία ωφελιμότητας (sed και awk), ανάλυση επίδοσης συστήματος, κατάστιχα ασφαλείας (syslog), θέματα εκκίνησης, διεργασίες και υποσύστημα αρχείων. ii) Προγραμματισμός στο κέλυφος Bash (περιβάλλον, δομές προγραμματισμού και αποσφαλμάτωσης). iii) Υποσυστήματα αρχείων και καταλόγων με κλήσεις συστήματος. iii) Διαχείριση διεργασιών (περιβάλλον, σήματα) και διαδιεργασιακή επικοινωνία (σωλήνες, FIFO, ουρές μηνυμάτων, κοινόχρηστη μνήμη, σηματοφόροι, υποδοχές δικτύου). iv) Πολυνηματικός προγραμματισμός και θέματα ταυτόχρονης. Προγραμματισμός δικτυακών πρωτοκόλλων RFC με υποδοχές, ταυτόχρονα και θέματα υψηλής επίδοσης. v) Θέματα ασφάλειας συστημάτων, θέματα διαχείρισης υποδομών και υπολογιστικών νεφών (εικονικοποίηση, κέντρα δεδομένων και Green IT).

Θέσεις: 25

Διαθέσιμες θέσεις: 24

Ενεργοποιημένο: true

Φοιτητές που έθεσαν προτίμηση	Username	Τρέχον εξάμηνο φοιτητή
Κωνσταντίνος Αναστασίου	canast07	12

Εικόνα 16. Σελίδα Μαθήματος - Admin

32

Κεφάλαιο 6

Συμπεράσματα

6.1 Συμπεράσματα	33
6.2 Μελλοντική Εργασία	34

6.1 Συμπεράσματα

Κατά την άποψη μου, η παρούσα εργασία θα βοηθήσει τόσο τους φοιτητές, όσο και το Τμήμα Πληροφορικής του Πανεπιστημίου Κύπρου. Κατ' αρχάς, με την χρήση του συστήματος, θα εξομαλυνθεί η διαδικασία εγγραφής φοιτητών στα μαθήματα περιορισμένης επιλογής, καθώς το τμήμα θα μπορεί να λαμβάνει τα «δημογραφικά» σχετικά με τις προτιμήσεις και τα πλάνα εγγραφής των φοιτητών. Ένα σημαντικό κομμάτι που θα ωφελήσει η εφαρμογή αυτή είναι στο κομμάτι του χρόνου. Ειδικά για τις εγγραφές του εαρινού εξαμήνου, το χρονικό περιθώριο για να αποφασίσουν οι φοιτητές τα μαθήματα που θα επιλέξουν είναι πολύ περιορισμένο. Αυτό το σύστημα θα επιτρέψει στους φοιτητές να ενημερωθούν έγκαιρα για τα μαθήματα που πρόκειται να προσφερθούν το επόμενο εξάμηνο και να αποφασίσουν τις προτιμήσεις τους.

Το αποτέλεσμα των πιο πάνω είναι ότι και το ίδιο το Τμήμα Πληροφορικής θα μπορεί να «κερδίσει» χρόνο, λόγω του ότι θα γνωρίζει σε μεγάλο βαθμό τις προθέσεις εγγραφής των φοιτητών και θα μπορεί να προγραμματιστεί κατάλληλα. Ακόμη, τα αποτελέσματα αυτά μπορούν στη συνέχεια να αναλυθούν και να χρησιμοποιηθούν έτσι ώστε να επιλεγθούν ορθότερα τα μαθήματα, το μέγεθος των ακροατηρίων, καθώς επίσης και το εξάμηνο διεξαγωγής του μαθήματος. Με το τέλος της διαδικασίας των εγγραφών μπορεί να γίνεται μια σύγκριση του ποσοστού ένδειξης ενδιαφέροντος για κάθε μάθημα, με τον πραγματικό αριθμό ατόμων που έχουν εγγραφεί επιτυχώς στο μάθημα και να εξαχθούν

συμπεράσματα. Για παράδειγμα, αν ένα μάθημα για συνεχόμενα έτη έχει χαμηλή ζήτηση, τότε το τμήμα θα το παρατηρήσει και θα πρέπει να λάβει μέτρα: είτε την αναβάθμιση του και την προσθήκη περισσότερων θεμάτων που θα αφορούν τους φοιτητές, είτε σαν έσχατη επιλογή θα πρέπει να σκεφτεί την αντικατάστασή του. Αντίστοιχα, αν ένα μάθημα έχει συνέχεια υπερβολική ζήτηση, τότε ίσως να πρέπει να εξεταστεί η διάθεση του μαθήματος και στα 2 εξάμηνα ή η αύξηση του αριθμού του ακροατηρίου, ή ακόμα και άλλα μέτρα που θα κρίνει το τμήμα απαραίτητα.

Επιπρόσθετα, θα μπορεί να δημιουργηθεί ένα πιο ευαισθητοποιημένο σύστημα εγγραφών, καθώς το τμήμα θα μπορεί να αναγνωρίσει τους φοιτητές που είναι πιο σημαντικό για αυτούς να επιλέξουν συγκεκριμένα μαθήματα (π.χ. φοιτητές που βρίσκονται στο 4^ο+ έτος και πλησιάζουν στην ολοκλήρωση του προγράμματος σπουδών τους). Έτσι, θα μπορεί να δοθεί προτεραιότητα στους φοιτητές αυτούς, με σκοπό να καταφέρουν να εγγραφούν επιτυχώς στο μάθημα που επιθυμούν.

Από τα πιο πάνω συμπεράσματα μπορούμε να αντιληφθούμε ότι το σύστημα αυτό θα έχει μια θετική αντιμετώπιση καθώς θα επιλύσει σε μεγάλο βαθμό το πρόβλημα της ζήτησης των μαθημάτων κατά τις εγγραφές και θα έχει ως αποτέλεσμα την βελτίωση της ποιότητας φοίτησης των φοιτητών.

6.2 Μελλοντική Εργασία

Σαν μελλοντικό έργο για το σύστημα αυτό υπάρχουν αρκετές λειτουργίες που μπορούν να προστεθούν και να υλοποιηθούν. Αρχικά θα πρέπει να πραγματοποιηθούν όλες οι λειτουργίες που αναφέρονται σε αυτή την εργασία και δεν έχουν υλοποιηθεί, αλλά και η υλοποίηση νέων λειτουργιών που θα είναι χρήσιμες στους διαχειριστές, όπως η εισαγωγή ιεραρχικά των επιλογών τους, με πρώτη να είναι το μάθημα που είναι πολύ σημαντική η εγγραφή τους σε αυτό. Επίσης, μπορεί να δημιουργηθεί λειτουργία εγγραφής στην σελίδα, την οποία θα χρησιμοποιεί ο κάθε φοιτητής για να αποκτήσει πρόσβαση στο σύστημα.

Ακόμη μια λειτουργία που μπορεί να προστεθεί είναι η πρόταση μιας αυτόματης κατανομής των φοιτητών σε μαθήματα με βάση τις προτιμήσεις τους. Αυτό μπορεί να

βοηθήσει το Τμήμα και σε συνεργασία με το Πανεπιστήμιο, οι εγγραφές στα μαθήματα αυτά να μην γίνονται χειροκίνητα από τους φοιτητές, και να γίνονται αυτόματα από το Τμήμα. Σε περίπτωση που αυτό δεν είναι εφικτό, τότε μπορεί να ετοιμάζεται αυτόματα το πλάνο του κάθε φοιτητή στο σύστημα Banner Web και κατά τη διάρκεια των εγγραφών, ο κάθε φοιτητής με το πάτημα ενός κουμπιού θα μπορεί να εγγραφεί στα μαθήματα. Τέλος, μια λειτουργία που μπορεί να υλοποιηθεί είναι η μετάφραση της σελίδας σε άλλες γλώσσες, όπως τα αγγλικά.

Βιβλιογραφία

- [1] Τμήμα Πληροφορικής Πανεπιστημίου Κύπρου, «Οδηγός Σπουδών», <https://www.cs.ucy.ac.cy/docs/prospectus.pdf>
- [2] IBM Cloud Education, IBM , “What is Three Tier Architecture”, 28 October 2020, <https://www.ibm.com/cloud/learn/three-tier-architecture>
- [3] Wikipedia, “CSS”, <https://el.wikipedia.org/wiki/CSS>
- [4] Wikipedia, “JavaScript”, <https://el.wikipedia.org/wiki/JavaScript>
- [5] Wikipedia, “Visual Studio Code”
https://en.wikipedia.org/wiki/Visual_Studio_Code
- [6] Wikipedia, “MongoDB”, <https://en.wikipedia.org/wiki/MongoDB>
- [7] Wikipedia, “NodeJS”, <https://en.wikipedia.org/wiki/Node.js>
- [8] Wikipedia, “Express JS”, <https://en.wikipedia.org/wiki/Express.js>
- [9] Wikipedia, “JADE (programming language)”,
[https://en.wikipedia.org/wiki/JADE_\(programming_language\)](https://en.wikipedia.org/wiki/JADE_(programming_language))
- [10] Wikipedia, “npm (software)”, [https://en.wikipedia.org/wiki/Npm_\(software\)](https://en.wikipedia.org/wiki/Npm_(software))