

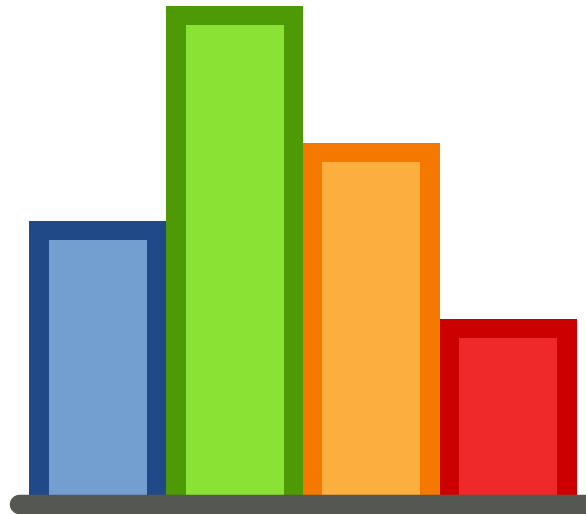


ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Δημιουργία συστήματος ανάκτησης πληροφοριών που αφορούν τους φοιτητές



Του φοιτητή
Κοσματόπουλου Μιχαήλ
Αρ. Μητρώου: 10/3592

Επιβλέπων καθηγητής
Αμανατιάδης Δημήτριος

Θεσσαλονίκη 2015

*Στη μνήμη του πατέρα μου,
λέκτορα του Φυσικού τμήματος ΑΠΘ
Κ. Κοσματόπουλο*

Πρόλογος

Ένα από τα ζητήματα που αφορούν την ομαλή λειτουργία του τμήματος είναι η ανάγκη ύπαρξης στατιστικών δεδομένων στην ιστοσελίδα του. Για το σκοπό αυτό ο επιβλέπων καθηγητής έχει συλλέξει ένα πλήθος δεδομένων των τελευταίων ετών με απώτερο στόχο τη δημιουργία στατιστικών σχετικών με το τμήμα. Τα δεδομένα βρίσκονται όμως σε λογιστικά φύλλα, και αυτό τα καθιστά μη προσβάσιμα στα υπόλοιπα μέλη του τμήματος.

Στόχος της συγκεκριμένης εργασίας είναι να καλύψει το συγκεκριμένο κενό με την αυτοματοποίηση της εξαγωγής στατιστικών, έπειτα από το ανέβασμα των σχετικών αρχείων. Πιο συγκεκριμένα, στα πλαίσια της παρούσας πτυχιακής εργασίας παρουσιάζεται ένα σύστημα υλοποιημένο σε PHP το οποίο κάνει επιπρόσθετα χρήση του Laravel framework.

Περίληψη

Σκοπός της συγκεκριμένης πτυχιακής είναι η δημιουργία ενός συστήματος το οποίο θα παρέχει στατιστικά για τους φοιτητές του Τμήματος Μηχανικών Πληροφορικής του ΑΤΕΙΘ. Η πτυχιακή εργασία αρχικά πραγματοποιεί μια ανάλυση της χειροκίνητης διαδικασίας που ακολουθούσε μέχρι σήμερα ο υπεύθυνος καθηγητής για την εξαγωγή των στατιστικών. Στη συνέχεια, παρουσιάζεται μια ιστορική αναδρομή σχετικά με το διαδίκτυο και τον Παγκόσμιο Ιστό. Έπειτα, παρουσιάζονται οι τεχνολογίες που χρησιμοποιήθηκαν για την δημιουργία του συστήματος. Επιπρόσθετα, η βασικότερη εξ' αυτών αναλύεται διεξοδικά σε επόμενο κεφάλαιο. Τέλος, η εργασία ολοκληρώνεται με την παράθεση τεχνικών λεπτομερειών υλοποίησης καθώς και του οδηγού χρήσης.

Abstract

The goal of this thesis is the development of a system that will provide statistics about the students of the IT Department of ATEITH. At first, the thesis analyzes the manual procedure that the professor in charge had to follow in order to gather these statistics. Then, a historical overview of the internet and the World Wide Web is presented. After that, the technologies that were used in the making of this system are presented. Additionally, the most important of them is analyzed in greater detail at a later chapter. Finally, the thesis concludes with some technical implementation details, and the user's guide.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Δ. Αμανατιάδη, για τη βοήθεια και στήριξή του καθ' όλη τη διάρκεια εκπόνησης της πτυχιακής μου εργασίας. Η καθοδήγηση του ήταν σημαντική, και συνέβαλε απόλυτα στο να ολοκληρωθεί αυτή η εργασία.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου για την υπομονή, εμπιστοσύνη και στήριξη που μου έδωσαν.

Τέλος, ευχαριστώ τους φίλους μου για την ανοχή και ενθάρρυνση που μου πρόσφεραν όλον αυτόν τον καιρό κατά τη συγγραφή της συγκεκριμένης εργασίας.

Περιεχόμενα

Πρόλογος	iii
Περίληψη	iv
Abstract	v
Ευχαριστίες	vi
Περιεχόμενα	vii
Κατάλογος σχημάτων	xi
Κατάλογος πινάκων	xiii
Εισαγωγή	1
1 Υπάρχουσα διαδικασία άντλησης στατιστικών	2
Εισαγωγή	2
1.1 Απόκτηση δεδομένων	2
1.1.1 students.txt	2
1.1.2 classStudents.txt	4
1.1.3 studentCourses.txt	4
1.2 Ενημέρωση Hydra	5
1.3 Ενημέρωση στατιστικών	6
Επίλογος	7
2 Γενικά στοιχεία για το διαδίκτυο	8
Εισαγωγή	8
2.1 Διαδίκτυο	8
2.1.1 Ορισμός διαδικτύου	8
2.1.2 Ιστορία του διαδικτύου	8
2.2 Παγκόσμιος Ιστός	9
2.2.1 Ορισμός παγκόσμιου ιστού	9

2.2.2	Ιστορία του παγκόσμιου ιστού	10
	Επίλογος	11
3	Τεχνολογίες που χρησιμοποιήθηκαν	12
	Εισαγωγή	12
3.1	Apache HTTP Server	12
3.2	HTML	12
3.3	CSS	13
3.4	JavaScript	13
3.5	JSON	13
3.6	jQuery	14
3.7	AJAX	14
3.8	Google Charts	15
3.9	Bootstrap	15
3.10	PHP	15
3.11	MySQL	16
3.12	Composer	16
3.13	Laravel	17
3.14	PhpStorm	17
	Επίλογος	18
4	Το Laravel αναλυτικά	19
	Εισαγωγή	19
4.1	Εγκατάσταση Laravel	19
4.1.1	Απαιτήσεις συστήματος	19
4.1.2	Χρήση εικονικής μηχανής	19
4.1.3	Εγκατάσταση με Composer	20
4.1.4	Βασικές ρυθμίσεις	21
4.1.5	Δοκιμή ιστοσελίδας	22
4.2	Διαδρομές	23
4.3	Ελεγκτές	24
4.4	Πρότυπα Blade	24
4.5	Artisan	26
4.6	Μεταναστεύσεις	28
4.7	Eloquent	30
4.7.1	Συσχετίσεις	31
4.7.1.1	Ένα προς ένα	31
4.7.1.2	Ένα προς πολλά	32

4.7.1.3	Πολλά προς πολλά	34
	Επίλογος	34
5	Υλοποίηση πτυχιακής εργασίας	35
	Εισαγωγή	35
5.1	Σχήμα βάσης δεδομένων	35
5.1.1	year_periods	36
5.1.2	students	36
5.1.3	student_status_histories	36
5.1.4	course_groups	37
5.1.5	courses	37
5.1.6	grades	37
5.1.7	grades_final	38
5.1.8	base_results	38
5.2	Όψεις βάσης δεδομένων	39
5.2.1	stats_students_unprocessed	39
5.2.2	stats_students	40
5.2.3	stats_deletions_unprocessed	40
5.2.4	stats_deletions_semester	41
5.2.5	stats_suspensions	42
6	Συμπεράσματα	44
7	Προτάσεις για μελλοντική έρευνα	45
	Βιβλιογραφία	46
	Παράρτημα Α Δείγμα students.txt	48
	Παράρτημα Β Δείγμα classStudents.txt	49
	Παράρτημα C Δείγμα studentCourses.txt	50
	Οδηγός Χρήσης Λογισμικού	51
	Εισαγωγή	51
	Εγκατάσταση	51
	Λειτουργίες που προσφέρει η εφαρμογή	52
	Αρχική σελίδα	52
	Σύνδεση	53

Φοιτητές	54
Πληροφορίες φοιτητή	55
Στατιστικά	57
Τρέχον εξάμηνο	57
Διαγραφές	57
Αναστολές	59
Μαθήματα	59
Πρόγραμμα σπουδών	61
Απόφοιτοι	61
Βάσεις	62
Ανέβασμα	63
students.txt	63
classStudents.txt	64
studentCourses.txt	64
Επίλογος	65

Κατάλογος σχημάτων

1.1	Είσοδος στην Hydra καθηγητών	6
1.2	Ενημέρωση δεδομένων	6
1.3	Λίστα φοιτητών	6
1.4	Παράδειγμα λογιστικού φύλλου	7
2.1	Sputnik I	8
2.2	Ντουάιτ Αϊζενχάουερ	9
2.3	Τιμ Μπέρνερς Λι	10
3.1	Λογότυπο Apache	12
3.2	Λογότυπο HTML5	13
3.3	Λογότυπο JSON	14
3.4	Λογότυπο jQuery	14
3.5	Λογότυπο Bootstrap	15
3.6	Λογότυπο PHP	16
3.7	Λογότυπο MySQL	16
3.8	Λογότυπο Composer	17
3.9	Λογότυπο Laravel	17
3.10	Λογότυπο PhpStorm	18
3.11	Περιβάλλον PhpStorm	18
4.1	Αρχική σελίδα καινούργιου Laravel project	23
5.1	Σχήμα βάσης δεδομένων	35
1	Αρχική σελίδα	53
2	Σύνδεση	54
3	Φοιτητές	55
4	Πληροφορίες φοιτητή	56
5	Στατιστικά τρέχοντος εξαμήνου	57
6	Στατιστικά διαγραφών	58
7	Φοιτητές σε αναστολή σπουδών	59

8	Στατιστικά μαθημάτων	60
9	Στατιστικά προγράμματος σπουδών	61
10	Στατιστικά αποφοίτων	62
11	Στατιστικά βάσεων	63
12	Ενημέρωση φοιτητών	63
13	Ενημέρωση βαθμολογιών	64
14	Ενημέρωση τελικών βαθμολογιών	64

Κατάλογος πινάκων

1.1	Περιγραφή των αρχείων δεδομένων	2
1.2	Πιθανές τιμές στήλης in_period_ID	3
1.3	Πιθανές τιμές στήλης cond_ID	3
1.4	Πιθανές τιμές στήλης cregtypeID	5
5.1	Στήλες πίνακα year_periods	36
5.2	Στήλες πίνακα students	36
5.3	Στήλες πίνακα student_status_histories	37
5.4	Στήλες πίνακα course_groups	37
5.5	Στήλες πίνακα courses	37
5.6	Στήλες πίνακα grades	38
5.7	Στήλες πίνακα grades_final	38
5.8	Στήλες πίνακα base_results	38
5.9	Παράδειγμα αποτελέσματος όψης stats_students_unprocessed	39
5.10	Παράδειγμα αποτελέσματος όψης stats_students	40
5.11	Παράδειγμα αποτελέσματος όψης stats_deletions_unprocessed	41
5.12	Παράδειγμα αποτελέσματος όψης stats_deletions_semester	42
5.13	Παράδειγμα αποτελέσματος όψης stats_suspensions	43
A.1	Δείγμα students.txt	48
B.1	Δείγμα classStudents.txt	49
C.1	Δείγμα studentCourses.txt	50

Εισαγωγή

Στόχος της πτυχιακής εργασίας είναι η ανάπτυξη ενός συστήματος το οποίο θα παρέχει πληροφορίες σχετικά με τους φοιτητές του τμήματος πληροφορικής. Τα δεδομένα θα προέρχονται από το σύστημα ηλεκτρονικής γραμματείας του ιδρύματος («Πυθία»).

Κάποιες από αυτές τις πληροφορίες είναι:

- Αριθμός ενεργών φοιτητών
- Αριθμός φοιτητών που κινδυνεύουν να διαγραφούν και άλλα στατιστικά διαγραφών
- Λίστα φοιτητών που βρίσκονται σε αναστολή σπουδών
- Στατιστικά μαθημάτων

Ειδικότερα, στο Κεφάλαιο 1 αναφέρεται η μέχρι στιγμής υπάρχουσα διαδικασία άντλησης στατιστικών. Στο Κεφάλαιο 2 δίνονται βασικές έννοιες που αφορούν το διαδίκτυο, και την ιστορία του. Οι τεχνολογίες που χρησιμοποιήθηκαν αναπτύσσονται στα Κεφάλαια 3 και 4. Τέλος, στο Κεφάλαιο 5 δίνονται κάποιες τεχνικές λεπτομέρειες που αφορούν την υλοποίηση της πτυχιακής εργασίας.

Συμπερασματικά λοιπόν, το συγκεκριμένο σύστημα δημιουργήθηκε για να παρέχει χρήσιμες πληροφορίες στο προσωπικό του τμήματος οι οποίες θα οδηγήσουν στην αποτελεσματικότερη και αρτιότερη οργάνωσή του.

1 Υπάρχουσα διαδικασία άντλησης στατιστικών

Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μια επεξήγηση της χειροκίνητης διαδικασίας που έπρεπε να εκτελέσει κάθε φορά ο υπεύθυνος για να ενημερωθούν τα αρχεία με τα στατιστικά των φοιτητών.

1.1 Απόκτηση δεδομένων

Αρχικά, ζητούνται από το Κέντρο Διαχείρισης Δικτύου τα τρέχοντα δεδομένα τα οποία σχετίζονται με τους φοιτητές της ηλεκτρονικής γραμματείας του τμήματος. Αυτά τα δεδομένα έρχονται σε 3 αρχεία, τα ονόματα των οποίων φαίνονται στον Πίνακα 1.1.

Πίνακας 1.1: Περιγραφή των αρχείων δεδομένων

Όνομα αρχείου	Περιγραφή
students.txt	Περιέχει πληροφορίες για όλους τους φοιτητές του τμήματος
classStudents.txt	Περιέχει όλες τις βαθμολογίες όλων των φοιτητών, όλων των εξεταστικών, ανεξαρτήτως αν πέρασαν ή όχι το μάθημα
studentCourses.txt	Περιέχει τις βαθμολογίες των περασμένων μαθημάτων

Όλα τα αρχεία είναι τύπου csv με διαχωριστή τον χαρακτήρα tab. Παρακάτω γίνεται μια περιγραφή των στηλών που περιέχει κάθε αρχείο.

1.1.1 students.txt

Το αρχείο αποτελείται από τις στήλες οι οποίες φαίνονται παρακάτω. Ένα δείγμα του αρχείου αυτού είναι διαθέσιμο στο Παράρτημα Α.

student_ID Μοναδικός αύξων αριθμός φοιτητή (ID). Χρησιμοποιείται εσωτερικά στη βάση.

spec_aem Η στήλη αυτή περιέχει τον αριθμό μητρώου του φοιτητή. Ο αριθμός μητρώου έχει την μορφή *XXYYYY* όπου *XX* είναι οι δύο τελευταίοι αριθμοί της χρονιάς εισαγωγής του φοιτητή, και *YYYY* ένας αύξων αριθμός.

first Η στήλη αυτή περιέχει το όνομα του φοιτητή.

last Η στήλη αυτή περιέχει το επώνυμο του φοιτητή.

fname Η στήλη αυτή περιέχει το πατρώνυμο του φοιτητή.

in_year Η στήλη αυτή περιέχει το έτος εισαγωγής του φοιτητή.

in_period_ID Η στήλη αυτή περιέχει την περίοδο εισαγωγής του φοιτητή. Μπορεί να πάρει τις τιμές οι οποίες φαίνονται στον Πίνακα 1.2.

Πίνακας 1.2: Πιθανές τιμές στήλης in_period_ID

Τιμή	Περιγραφή
1	Χειμερινό εξάμηνο
2	Εαρινό εξάμηνο

cond_ID Η στήλη αυτή περιέχει την κατάσταση του φοιτητή (ενεργός, διαγραμμένος, πτυχιούχος, κτλ). Η τιμή αυτής της στήλης ενδέχεται να αλλάξει από εξάμηνο σε εξάμηνο. Οι πιθανές τιμές που μπορεί να πάρει φαίνονται στον πίνακα 1.3.

Πίνακας 1.3: Πιθανές τιμές στήλης cond_ID

Τιμή	Κατάσταση
1	Ενεργός
2	Διαγραμμένος
3	Απόφοιτος
4	Σε αναστολή σπουδών
6	Σε αναμονή παρουσίασης πτυχιακής

dgr_logos Σε περίπτωση που ο φοιτητής έχει διαγραφεί από τη σχολή, αυτή η στήλη ενδέχεται να περιέχει τον λόγο διαγραφής.

in_exam_ID Εξάμηνο εισαγωγής φοιτητή. Σε περίπτωση που ο φοιτητής έρχεται από μεταγραφή, ενδέχεται να είναι μεγαλύτερο από 1.

exam_ID Τρέχον εξάμηνο φοιτητή. Αυξάνεται μόνο εφόσον η κατάσταση του φοιτητή είναι ενεργή.

1.1.2 classStudents.txt

Το αρχείο αποτελείται από τις στήλες που φαίνονται παρακάτω. Ένα δείγμα του αρχείου αυτού είναι διαθέσιμο στο Παράρτημα Β.

student_ID Το ID του φοιτητή της συγκεκριμένης βαθμολογίας.

spec_aem Ο αριθμός μητρώου του φοιτητή της συγκεκριμένης βαθμολογίας.

courseID Ο κωδικός του μαθήματος της συγκεκριμένης βαθμολογίας.

coursecode Ο κωδικός του μαθήματος της συγκεκριμένης βαθμολογίας, όπως αναγράφεται στο πρόγραμμα σπουδών.

classtitle Ο τίτλος του μαθήματος.

year Η χρονιά κατά την οποία εξετάστηκε το μάθημα.

periodID Η περίοδος που εξετάστηκε το μάθημα. Βλ. Πίνακα 1.2 για τις πιθανές τιμές.

grade Ο βαθμός του συγκεκριμένου φοιτητή τη συγκεκριμένη εξεταστική στο συγκεκριμένο μάθημα.

epshort Περιέχει πληροφορίες για την περίοδο που εξετάστηκε το μάθημα. Οι πιθανές τιμές αυτής της στήλης είναι:

- A XEIM
- B XEIM
- A EAP
- B EAP

1.1.3 studentCourses.txt

Το αρχείο αποτελείται από τις στήλες οι οποίες φαίνονται παρακάτω. Ένα δείγμα του αρχείου αυτού είναι διαθέσιμο στο Παράρτημα C.

student_ID Το ID του φοιτητή της συγκεκριμένης βαθμολογίας.

spec_aem Ο αριθμός μητρώου του φοιτητή της συγκεκριμένης βαθμολογίας.

courseID Ο κωδικός του μαθήματος της συγκεκριμένης βαθμολογίας.

coursecode Ο κωδικός του μαθήματος της συγκεκριμένης βαθμολογίας, όπως αναγράφεται στο πρόγραμμα σπουδών.

title Ο τίτλος του μαθήματος.

cyear Η χρονιά κατά την οποία εξετάστηκε το μάθημα.

cperiod Η περίοδος που εξετάστηκε το μάθημα. Βλ. Πίνακα 1.2 για τις πιθανές τιμές.

cregtypeID Δείχνει εάν ο βαθμός χρειάστηκε κάποια ιδιαίτερη μεταχείριση. Οι πιθανές τιμές που μπορεί να πάρει φαίνονται στον Πίνακα 1.4.

Πίνακας 1.4: Πιθανές τιμές στήλης *cregtypeID*

Τιμή	Περιγραφή
0	Δεν χρειάστηκε ιδιαίτερη μεταχείριση
1	Χρειάστηκε ιδιαίτερη μεταχείριση

epshort Περιέχει πληροφορίες για την περίοδο που εξετάστηκε το μάθημα. Οι πιθανές τιμές αυτής της στήλης είναι:

- A ΧΕΙΜ
- B ΧΕΙΜ
- A ΕΑΡ
- B ΕΑΡ

cgrade Ο τελικός βαθμός του συγκεκριμένου φοιτητή τη συγκεκριμένη εξεταστική στο συγκεκριμένο μάθημα

notes Σε περίπτωση που *cregtypeID* == 1 τότε ενδέχεται να περιέχει περισσότερες πληροφορίες για την συγκεκριμένη βαθμολογία.

1.2 Ενημέρωση Hydra

Αφού ο υπεύθυνος παραλάβει τα επικαιροποιημένα δεδομένα του συστήματος γραμματείας, πρέπει να ενημερώσει την Hydra. Για να γίνει αυτό, κάνει είσοδο με τα στοιχεία του, και μεταβαίνει στην σελίδα ενημέρωσης δεδομένων, και ανεβάζει την καινούργια έκδοση του αρχείου *students.txt*.

Type in the username and the password of your account for server aetos.it.teithe.gr

Username:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="Login"/>	

Σχήμα 1.1: Είσοδος στην Hydra καθηγητών

Import Students	
Select a file to upload and import it in the database. This will synchronize database student entries with uploaded file. Non-existing entries will be deleted! (Note: Database import may take some time) Encoding must be iso8859-7.	
File to upload:	<input type="text" value="Αναζήτηση..."/> Δεν επιλέχθηκε αρχείο.
Update database:	<input type="checkbox"/>
<input type="button" value="Upload"/> <input type="button" value="Cancel"/>	

Σχήμα 1.2: Ενημέρωση δεδομένων

1.3 Ενημέρωση στατιστικών

Στη συνέχεια, ο υπεύθυνος μεταβαίνει στη σελίδα με την λίστα των φοιτητών. Από εκεί, επιλέγει ένα-ένα όλα τα πιθανά εξάμηνα εισαγωγής, και καταγράφει σε ένα υπολογιστικό φύλλο (spreadsheet) τον αριθμό των ενεργών, διαγραμμένων, πτυχιούχων, και σε αναστολή φοιτητών.

Pages: [1-25] [26-50] [51-75] [76-100] [101-125] [126-150] [151-175] [176-200] [201-225] [226-250] [251-275] [276-300] [301-325] [326-350] [351-375] [376-400] [401-425] [426-450] [451-475] [476-500] [501-525] [526-550] [551-575] [576-600] [601-625] [626-650] [651-675] [676-700] [701-725] [726-750] [751-775] [776-800] [801-825] [826-850] [851-875] [876-900] [901-925] [926-950] [951-975] [976-1000] [1001-1025] [1026-1050] [1051-1075] [1076-1100] [1101-1125] [1126-1127]

Status: All Active Deleted Graduated Moved

View: Students Accounts

Semester: All [x8788] [E8788] [x8889] [E8889] [x8990] [E8990] [x9091] [E9091] [x9192] [E9192] [x9293] [E9293] [x9394] [E9394] [x9495] [E9495] [x9596] [E9596] [x9697] [E9697] [x9798] [E9798] [x9899] [E9899] [x9900] [E9900] [x0001] [E0001] [x0102] [E0102] [x0203] [E0203] [x0304] [E0304] [x0405] [E0405] [x0506] [E0506] [x0607] [E0607] [x0708] [E0708] [x0809] [E0809] [x0910] [E0910] [x1011] [E1011] [x1112] [E1112] [x1213] [E1213] [x1314] [E1314] [x1415] [E1415]

A.M.	Eisagogi	Lastname	Firstname	Fathename	Username
042719	X0405	DIMITRIEVSKA	VASILIKIDA	KOCO	vassim
093552	X0910	HAMMOUCH	MOHAMED	IDRISS	hammouch
042722	X0405	ZAKA	KRISTINA	TRIFON	krizak
134057	X1314	ABRAMHΣ	ΚΩΝΣΤΑΝΤΙΝΟΣ	ΣΤΥΛΙΑΝΟΣ	kavramis
103684	X1011	ΑΒΡΑΜΙΔΗΣ	ΙΩΑΝΝΗΣ	ΔΗΜΗΤΡΗΣ	lavram
073225	E0708	ΑΒΡΑΜΙΔΗΣ	ΚΩΝΣΤΑΝΤΙΝΟΣ	ΑΝΤΩΝΙΟΣ	kavramid
052825	X0506	ΑΓΘΟΠΟΥΛΟΣ	ΑΝΤΩΝΙΟΣ	ΓΕΩΡΓΙΟΥ	anagath
083367	X0809	ΑΓΓΕΛΟΥΔΗΣ	ΔΗΜΗΤΡΙΟΣ	ΓΕΩΡΓΙΟΣ	dangel
144275	X1415	ΑΓΓΕΛΟΥΣΗΣ	ΣΩΤΗΡΙΟΣ	ΚΥΡΙΑΚΟΣ	sangelou
113697	X1112	ΑΓΓΟΥΡΙΑΣ	ΘΕΟΦΥΛΑΚΤΟΣ	ΑΘΑΝΑΣΙΟΣ	tangouri
063144	X0607	ΑΓΓΟΥΣΟΓΛΟΥ	ΟΖΓΚΙΟΥΡ	ΜΕΜΕΤ-ΦΙΚΡΗ	ozgang
083347	E0809	ΑΓΗΣΙΛΑΟΥ	ΖΗΝΩΝΑΣ	ΑΓΗΣ	zionas
134076	X1314	ΑΓΚΟ	ΑΙΣΕΦΚΙΟΥΛ	ΑΧΜΕΤ	aagko
042661	X0405	ΑΔΑΜΙΔΗΣ	ΚΩΝΣΤΑΝΤΙΝΟΣ	ΔΙΟΝΥΣΙΟΥ	konadam
093463	X0910	ΑΖΗΖΟΓΛΟΥ	ΧΑΓΙΑΤ	ΧΙΛΜΗ	cazizog
134136	X1314	ΑΖΟΙΔΗΣ	ΑΝΤΥΓΙΑΣ	ΓΕΩΡΓΑΔΗΣ	aazoidis
022070	E0203	ΑΘΑΝΑΣΑΣ	ΙΩΑΝΝΗΣ	ΒΑΣΙΛΕΙΟΥ	athioan
093493	X0910	ΑΘΑΝΑΣΙΑΔΗΣ	ΑΒΡΑΑΜ	ΑΝΔΡΕΑΣ	aathan
022105	E0203	ΑΘΑΝΑΣΙΑΔΗΣ	ΧΡΗΣΤΟΣ	ΛΑΜΠΡΟΣ	chrathan
950691	E9596	ΑΘΑΝΑΣΙΑΔΟΥ	ΧΑΡΑΛΑΜΠΙΑ	ΣΠΥΡΙΑΔΩΝΑ	dock

Σχήμα 1.3: Λίστα φοιτητών

Αφού ολοκληρωθεί αυτή η διαδικασία για όλα τα εξάμηνα εισαγωγής, μπορούν πλέον να αντληθούν πληροφορίες όπως για παράδειγμα το ποσοστό των φοιτητών που έχουν πάρει πτυχίο.

	A	B	C	D	E	F
1	Εξάμηνο	εισήχθησαν	αποφοίτησαν	διαγράφησαν	ενεργοί	διακοπή
2	x1415	234	0	4	228	2
3	e1314	20	0	0	19	1
4	x1314	169	0	16	148	5
5	x1213	151	0	32	116	3
6	x1112	129	0	19	108	2
7	x1011	133	1	42	83	7

Σχήμα 1.4: Παράδειγμα λογιστικού φύλλου

Επίλογος

Σε αυτό το κεφάλαιο έγινε μια σύντομη περιγραφή της διαδικασίας η οποία ακολουθείται σήμερα για την εξαγωγή στατιστικών των φοιτητών. Όπως φάνηκε, είναι μια χρονοβόρα διαδικασία, όπου μπορούν εύκολα να γίνουν λάθη.

2 Γενικά στοιχεία για το διαδίκτυο

Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μια αναφορά σε βασικές έννοιες που αφορούν το διαδίκτυο.

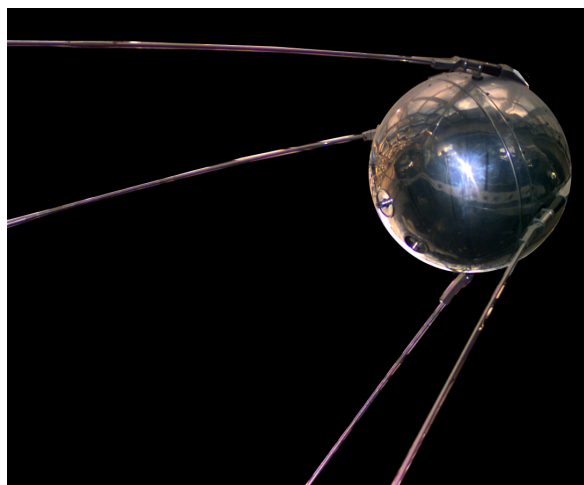
2.1 Διαδίκτυο

2.1.1 Ορισμός διαδικτύου

Το διαδίκτυο (internet) είναι ένα δίκτυο από δίκτυα (Braden, 1989). Αποτελείται από πολλαπλά ιδιωτικά, δημόσια, ακαδημαϊκά, εταιρικά, κυβερνητικά, και άλλα δίκτυα. Χρησιμοποιείται για διάφορες εφαρμογές, όπως για τον παγκόσμιο ιστό, την ηλεκτρονική αλληλογραφία, την τηλεφωνία, και άλλα («Internet», 2015).

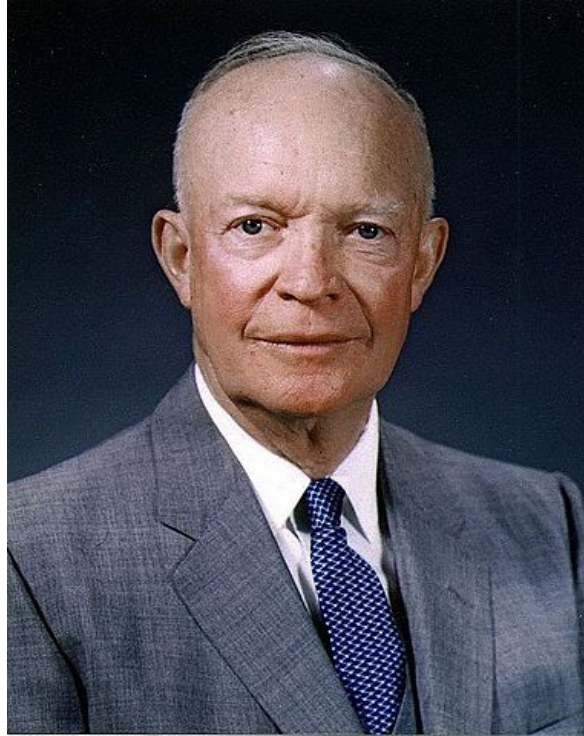
2.1.2 Ιστορία του διαδικτύου

Το 1957 η Ρωσία έθεσε τον πρώτο τεχνητό δορυφόρο (Sputnik I) σε τροχιά γύρω από τη γη.



Σχήμα 2.1: Sputnik I (NSSDC & NASA, 2004)

Το 1958, ο τότε πρόεδρος των Ηνωμένων Πολιτειών της Αμερικής Ντουάιτ Αϊζενχάουερ (Dwight Eisenhower) δημιούργησε τον οργανισμό ARPA (Advanced Research Project Agency — Οργανισμός Έρευνας Προχωρημένων Προγραμμάτων), έτσι ώστε η Αμερική να μπορέσει να ξανακερδίσει την τεχνολογική αιχμή της.



Σχήμα 2.2: Ντουάιτ Αϊζενχάουερ (White House, 1959)

Μετά από λίγα χρόνια, δημιουργήθηκε το δίκτυο ARPANET (ARPA Network). Το δίκτυο ARPANET χρησιμοποιούσε μεταγωγή πακέτων (packet switching – η οποία για την εποχή της αποτελούσε μια νέα έννοια) αντί για μεταγωγή κυκλώματος (circuit switching) για την μεταφορά δεδομένων.

Το πρωτόκολλο δικτύωσης που χρησιμοποιούσε το δίκτυο ARPANET ήταν το NCP (Network Control Program). Το 1983 αντικαταστήθηκε από το TCP/IP που χρησιμοποιείται μέχρι και σήμερα (Stewart, 2000).

2.2 Παγκόσμιος Ιστός

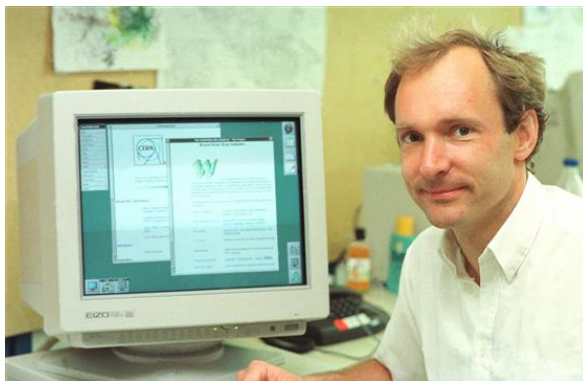
2.2.1 Ορισμός παγκόσμιου ιστού

Ο παγκόσμιος ιστός (World Wide Web — WWW) αποτελείται από διακομιστές (servers) που στέλνουν έγγραφα γραμμένα σε μια ειδική γλώσσα σήμανσης (markup language) που ονομάζεται HTML (βλ. 3.2). Τα έγγραφα αυτά μπορεί να περιέχουν συνδέσμους σε άλλα έγγραφα, εικόνες, βίντεο, ήχους, και άλλα πολλά (Beal, χ.η.).

Για την πρόσβαση στον παγκόσμιο ιστό χρησιμοποιείται ένας φυλλομετρητής (web browser).

2.2.2 Ιστορία του παγκόσμιου ιστού

Ο Τιμ Μπέρνερς Λι (Tim Berners-Lee) — μηχανικός λογισμικού στο CERN — παρατήρησε ότι υπήρχε μεγάλη δυσκολία στην ανταλλαγή πληροφοριών με τους συναδέλφους επιστήμονες του. Ο Τιμ αποφάσισε να εκμεταλλευτεί το διαδίκτυο και την τότε αναδυόμενη τεχνολογία του υπερκειμένου για να αντιμετωπίσει το συγκεκριμένο πρόβλημα. Τον Μάρτιο του 1989 κατέγραψε το όραμά του σε ένα κείμενο με τίτλο «Διαχείριση πληροφοριών: Μια πρόταση», και το κατέθεσε στη διοίκηση του CERN. Αρχικά, ο προϊστάμενος του απέρριψε την πρότασή του Τιμ, αλλά τον Σεπτέμβριο του 1990 του δόθηκε χρόνος για να δουλέψει επάνω σ' αυτό.



Σχήμα 2.3: Τιμ Μπέρνερς Λι (CERN, 1994)

Μέχρι τον Οκτώβριο του 1990 είχε αναπτύξει τις τρεις κυριότερες τεχνολογίες που χρησιμοποιούνται μέχρι και σήμερα στον παγκόσμιο ιστό:

- **HTML:** Γλώσσα σήμανσης για τον παγκόσμιο ιστό
- **URI:** Γνωστό και ως URL. Μια διεύθυνση που χρησιμοποιείται για την μοναδική αντιστοίχιση κάποιου πόρου στον παγκόσμιο ιστό
- **HTTP:** Πρωτόκολλο επικοινωνίας που επιτρέπει την ανάκτηση των HTML εγγράφων μέσω των URI τους

Τον Οκτώβριο του 1994 ο Τιμ έφυγε από το CERN για να ιδρύσει την επιτροπή του παγκόσμιου ιστού (World Wide Web Consortium — W3C), όπου παραμένει διευθυντής μέχρι και σήμερα.

Επίλογος

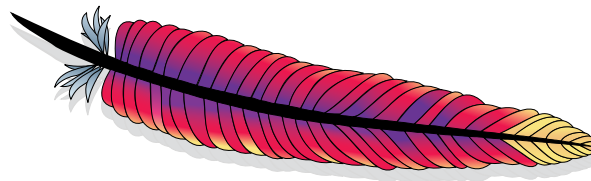
Στο κεφάλαιο αυτό έγινε μια σύντομη περιγραφή του διαδικτύου και του παγκόσμιου ιστού, όπως επίσης και μια ιστορική αναδρομή για τις συγκεκριμένες έννοιες.

3 Τεχνολογίες που χρησιμοποιήθηκαν

Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μια ανάλυση των τεχνολογιών που χρησιμοποιήθηκαν για την πραγματοποίηση της πτυχιακής.

3.1 Apache HTTP Server



Σχήμα 3.1: Λογότυπο Apache (Apache Software Foundation, χ.η.-a)

Επίσημη ιστοσελίδα: <http://httpd.apache.org/>

Ο Apache HTTP Server (ή αλλιώς σκέτο Apache) είναι ένας ασφαλής, αποδοτικός, επεκτάσιμος και ανοιχτού κώδικα HTTP εξυπηρετητής. Κυκλοφόρησε για πρώτη φορά το 1995, και από τον Απρίλιο του 1996 παραμένει μέχρι και σήμερα ο πιο δημοφιλής HTTP διακομιστής (Apache Software Foundation, χ.η.-b). Είναι συμβατός με όλα τα σύγχρονα λειτουργικά συστήματα.

3.2 HTML

Επίσημη ιστοσελίδα: <http://www.w3.org/html/>

Η HTML (HyperText Markup Language — Γλώσσα Σήμανσης Υπερκειμένου), είναι η γλώσσα που χρησιμοποιείται για τη δημιουργία ιστοσελίδων. Δεν αποτελεί γλώσσα προγραμματισμού, αλλά είναι μία γλώσσα που χρησιμοποιεί tags (ετικέτες) για να περιγράψει μία ιστοσελίδα.



Σχήμα 3.2: Λογότυπο HTML5

Οι ετικέτες ξεκινούν με ένα `<` ενώ τελειώνουν με ένα `>`, ενώ ανάμεσα μπαίνει αυτό που περιγράφει την ενέργεια που θα έχει πάνω στην ιστοσελίδα η ετικέτα. Οι ετικέτες τις περισσότερες φορές πρέπει να έχουν μία αρχή, και ένα τέλος.

Για παράδειγμα, η ετικέτα `test` θα είχε ως αποτέλεσμα ο φυλλομετρητής να εμφανίσει το παρακάτω: **test**. Ο φυλλομετρητής μόλις «δει» την ετικέτα `` ξέρει ότι πρέπει να εμφανίσει το κείμενο που είναι μέσα στην ετικέτα με έντονους χαρακτήρες.

3.3 CSS

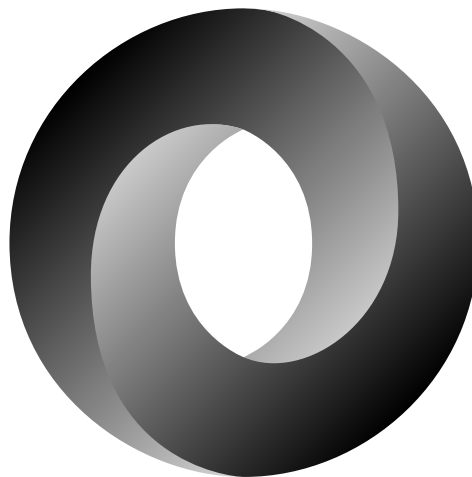
Το CSS (Cascading Style Sheets — Διαδοχικά Φύλλα Στυλ), χρησιμοποιείται για τη μορφοποίηση και γενικότερα την αλλαγή της εμφάνισης μιας ιστοσελίδας γραμμένης σε HTML.

3.4 JavaScript

Η JavaScript είναι μία client-side scripting γλώσσα προγραμματισμού που χρησιμοποιείται κυρίως στις ιστοσελίδες και επιτρέπει τη δημιουργία δυναμικών ιστοσελίδων.

3.5 JSON

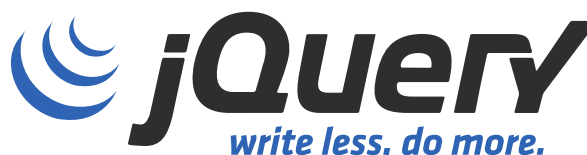
Επίσημη ιστοσελίδα: <http://json.org/>



Σχήμα 3.3: Λογότυπο JSON

Το JSON (JavaScript Object Notation — Συμβολισμός Αντικειμένων JavaScript) είναι ένα ανοιχτό πρότυπο που βασίζεται σε κείμενο που διαβάζεται εύκολα απ' τους ανθρώπους, και χρησιμοποιείται συνήθως για την μεταφορά αντικειμένων δεδομένων.

3.6 jQuery



Σχήμα 3.4: Λογότυπο jQuery

Επίσημη ιστοσελίδα: <http://jquery.com/>

Η jQuery είναι μια ανοιχτού κώδικα JavaScript βιβλιοθήκη, που στόχος της είναι να κάνει τον προγραμματιστή να «γράψει λιγότερα, και να κάνει περισσότερα». Κάνει ευκολότερο τον χειρισμό των HTML εγγράφων, τις κινήσεις (animations), και άλλα πράγματα που θα απαιτούσαν περισσότερες γραμμές κώδικα για να γίνουν με απλή JavaScript.

3.7 AJAX

Η AJAX (Asynchronous JavaScript and XML — Ασύγχρονη JavaScript και XML) είναι μια ομάδα τεχνικών που επιτρέπουν την δημιουργία ιστοσελίδων που φαίνεται να ανταποκρίνονται πιο γρήγορα. Αυτό επιτυγχάνεται με την φόρτωση δεδομέ-

νων στο παρασκήνιο (background), οπότε δεν χρειάζεται να φορτώσει από την αρχή ολόκληρη η ιστοσελίδα, αλλά μόνο το κομμάτι που χρειάζεται να αλλάξει. Η χρήση της XML δεν είναι αναγκαία. Τις περισσότερες φορές χρησιμοποιείται JSON αντί για XML.

3.8 Google Charts

Επίσημη ιστοσελίδα: <https://developers.google.com/chart/>

Η υπηρεσία Google Charts είναι μια δωρεάν υπηρεσία που δίνει στους προγραμματιστές έναν εύκολο και γρήγορο τρόπο να εισάγουν όμορφα και διαδραστικά γραφήματα στις ιστοσελίδες τους.

3.9 Bootstrap



Σχήμα 3.5: Λογότυπο Bootstrap

Επίσημη ιστοσελίδα: <http://getbootstrap.com/>

Το Bootstrap είναι μια συλλογή από εργαλεία ανοιχτού κώδικα που βοηθούν στην εύκολη και γρήγορη δημιουργία ιστοσελίδων που ανταποκρίνονται στα διάφορα μεγέθη συσκευών (responsive design). Αποτελείται κυρίως από CSS, JavaScript και επεκτάσεις της jQuery.

3.10 PHP

Επίσημη ιστοσελίδα: <http://php.net/>

Η αποκλειστική χρήση της HTML μπορεί να δημιουργήσει μόνο στατικές σελίδες, και όχι δυναμικές. Δυναμικές είναι οι ιστοσελίδες όπου το περιεχόμενό τους αλλάζει



Σχήμα 3.6: Λογότυπο PHP

αυτόματα, χωρίς να χρειάζεται να το αλλάξει ο δημιουργός τους. Ένα παράδειγμα μιας δυναμικής ιστοσελίδας είναι ένα forum.

Επειδή λοιπόν η HTML είναι ανίκανη από μόνη της να δημιουργήσει δυναμικές ιστοσελίδες, έχουν δημιουργηθεί διάφορες γλώσσες που επιτρέπουν τη δημιουργία δυναμικών ιστοσελίδων. Μια από αυτές τις γλώσσες είναι η PHP.

Η PHP (PHP: Hypertext Preprocessor — PHP: Προεπεξεργαστής υπερκειμένου) είναι μια server-side scripting γλώσσα προγραμματισμού, που χρησιμοποιείται κυρίως για τη δημιουργία δυναμικών ιστοσελίδων, και όχι μόνο (PHP Group, χ.η.).

Η PHP συνήθως συμπεριλαμβάνεται στον κώδικα HTML με τα tags `<?php ... ?>`. Όταν διερμηνέας αναγνωρίσει αυτά τα tags, θα εκτελέσει ό,τι βρίσκεται εκεί μέσα. Με αυτόν τον τρόπο μπορούν να δημιουργηθούν δυναμικές ιστοσελίδες.

3.11 MySQL



Σχήμα 3.7: Λογότυπο MySQL

Επίσημη ιστοσελίδα: <https://www.mysql.com/>

Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Μια βάση δεδομένων είναι μια δομημένη συλλογή από δεδομένα (Oracle Corporation, χ.η.).

3.12 Composer

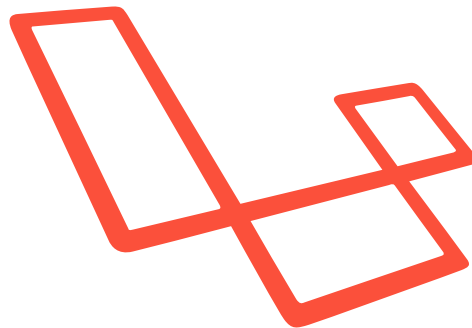
Επίσημη ιστοσελίδα: <https://getcomposer.org/>



Σχήμα 3.8: Λογότυπο Composer

Το Composer είναι ένα σύστημα διαχείρισης εξαρτήσεων για την PHP. Επιτρέπει την εύκολη εγκατάσταση και ενημέρωση όλων των βιβλιοθηκών που ενδέχεται να χρησιμοποιεί μια PHP ιστοσελίδα (Composer, χ.η.).

3.13 Laravel



Σχήμα 3.9: Λογότυπο Laravel

Επίσημη ιστοσελίδα: <http://laravel.com/>

Το Laravel είναι ένα PHP framework ανοιχτού λογισμικού που επιτρέπει την γρήγορη ανάπτυξη ιστοσελίδων. Το Laravel θα αναλυθεί περισσότερο στο κεφάλαιο 4.

3.14 PhpStorm

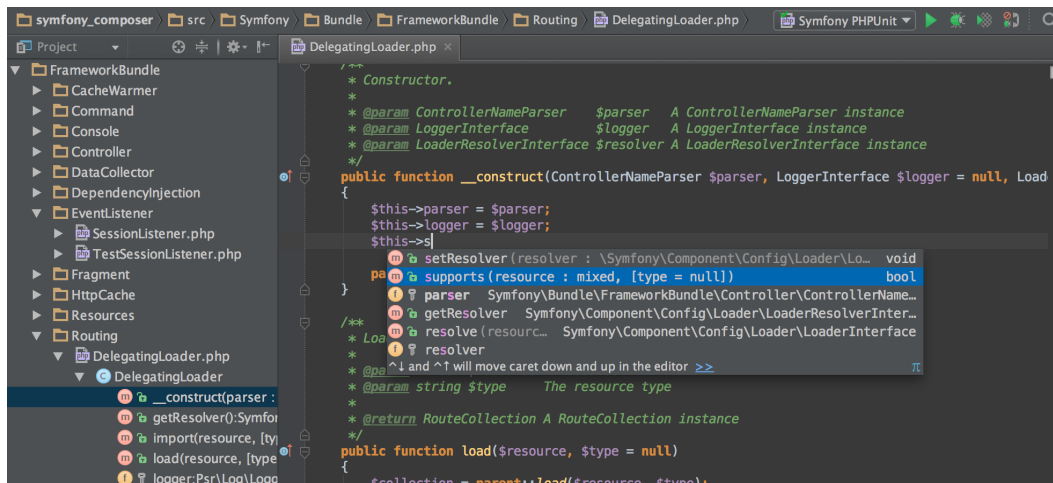
Επίσημη ιστοσελίδα: <https://www.jetbrains.com/phpstorm/>

Το PhpStorm από την εταιρεία JetBrains είναι ένα εμπορικό PHP IDE (integrated development environment — ολοκληρωμένο περιβάλλον ανάπτυξης). Περιέχει λει-



Σχήμα 3.10: Λογότυπο PhpStorm

τουργίες όπως αυτόματη συμπλήρωση κώδικα, επεξεργαστή κειμένου για HTML/CSS/JavaScript αρχεία, υποστήριξη για frameworks (συμπεριλαμβανομένου και του Laravel), και άλλα πολλά.



Σχήμα 3.11: Περιβάλλον PhpStorm

Επίλογος

Σε αυτό το κεφάλαιο έγινε μια περιγραφή των σημαντικότερων τεχνολογιών που χρησιμοποιήθηκαν σε αυτή την πτυχιακή εργασία.

4 Το Laravel αναλυτικά

Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μια εκτενέστερη περιγραφή των λειτουργιών που προσφέρει το Laravel.

4.1 Εγκατάσταση Laravel

Σε αυτή την ενότητα θα γίνει μια σύντομη αναφορά στον τρόπο εγκατάστασης του Laravel (Otwell, χ.η.-f).

4.1.1 Απαιτήσεις συστήματος

Για την ομαλή λειτουργία του Laravel, απαιτούνται τα παρακάτω:

- PHP \geq 5.5.9
- Επέκταση OpenSSL για PHP
- Επέκταση PDO για PHP
- Επέκταση Mbstring για PHP
- Επέκταση Tokenizer για PHP

4.1.2 Χρήση εικονικής μηχανής

Το Vagrant είναι ένα ανοιχτού κώδικα εργαλείο που παρέχει εικονικά περιβάλλοντα ανάπτυξης λογισμικού. Αυτά τα εικονικά περιβάλλοντα περιλαμβάνουν όλα τα τμήματα λογισμικού που μπορεί να χρειαστούν για την ανάπτυξη σε κάποια συγκεκριμένη πλατφόρμα. Είναι δυνατόν να βασιστεί είτε στο VMWare Workstation, είτε στο Oracle VM VirtualBox (Palat, 2012).

Το Laravel παρέχει το Laravel Homestead, το οποίο είναι ένα περιβάλλον για το Vagrant που περιλαμβάνει προεγκατεστημένα τα παρακάτω λογισμικά:

- Ubuntu 14.04
- PHP 5.6
- HHVM
- Nginx
- MySQL
- Postgres
- Node (με PM2, Bower, Grunt, και Gulp)
- Redis
- Memcached
- Beanstalkd
- Laravel Envoy
- Blackfire Profiler

Η εγκατάσταση του Laravel Homestead είναι πολύπλοκη, και έξω από τα πλαίσια της πτυχιακής. Ο ενδιαφερόμενος αναγνώστης μπορεί να μεταβεί στον παρακάτω σύνδεσμο για περισσότερες πληροφορίες:

<http://laravel.com/docs/5.1/homestead>

4.1.3 Εγκατάσταση με Composer

Το Laravel μπορεί επίσης να εγκατασταθεί μέσω του Composer (βλ. 3.12). Για να γίνει αυτό, πρέπει πρώτα να τοποθετηθεί η διεύθυνση `./composer/vendor/bin` στην μεταβλητή περιβάλλοντος PATH. Μετά, πρέπει να εκτελεστεί η παρακάτω εντολή για την εγκατάσταση του προγράμματος εγκατάστασης του Laravel:

```
composer global require "laravel/installer=~1.1"
```

Listing 4.1: Αποτέλεσμα εκτέλεσης εντολής `laravel new`

```
$ composer global require "laravel/installer=~1.1"
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
 - Installing laravel/installer (v1.2.1)
   Downloading: connection...
   Downloading: 0%
   Downloading: 100%

Writing lock file
Generating autoload files
```

Αφού γίνει αυτό, θα πρέπει να εκτελεστεί η παρακάτω εντολή για να δημιουργηθεί καινούργιο project στον τρέχοντα φάκελο:

```
laravel new όνομα_project
```

Listing 4.2: Αποτέλεσμα εκτέλεσης εντολής `laravel new`

```
$ laravel new pithia-stats
Crafting application...
> php -r "copy('.env.example', '.env');"
> php artisan clear-compiled
> php artisan optimize
Generating optimized class loader
> php artisan key:generate
Application key [pyx20XcKMPPhQB5McES7b7FoT09w4U1S3] set successfully.
Application ready! Build something amazing.
```

4.1.4 Βασικές ρυθμίσεις

Στον φάκελο που δημιουργήθηκε το project θα υπάρχει ένα αρχείο με το όνομα `.env`. Το αρχείο αυτό περιέχει μερικές βασικές ρυθμίσεις.

Listing 4.3: Περιεχόμενα αρχείου .env

```
APP_ENV=local
APP_DEBUG=true
APP_KEY=pyx20XcKMPPhQB5McES7b7FoT09w4U1S3

DB_HOST=localhost
DB_DATABASE=homestead
DB_USERNAME=homestead
DB_PASSWORD=secret

CACHE_DRIVER=file
SESSION_DRIVER=file
QUEUE_DRIVER=sync

MAIL_DRIVER=smtp
MAIL_HOST=mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null
```

Ιδιαίτερο ενδιαφέρον παρουσιάζει η ρύθμιση `APP_KEY` που είναι ένα τυχαίο αλφαριθμητικό το οποίο παράγεται τυχαία κατά την δημιουργία του project, και χρησιμοποιείται στην κρυπτογράφηση. Επίσης, οι ρυθμίσεις `DB_HOST`, `DB_DATABASE`, `DB_USERNAME`, και `DB_PASSWORD` αποτελούν τα στοιχεία σύνδεσης στη βάση δεδομένων. Ο προεπιλεγμένος τύπος βάσης δεδομένων είναι η MySQL, αλλά μπορεί εύκολα να αλλάξει μέσα από το αρχείο `./config/database.php`.

4.1.5 Δοκιμή ιστοσελίδας

Για να λειτουργήσει η ιστοσελίδα, ο εξυπηρετητής θα πρέπει να ρυθμιστεί έτσι ώστε να διανέμει τα αρχεία που βρίσκονται στον φάκελο `./public/`. Η διαδικασία αυτή διαφέρει από εξυπηρετητή σε εξυπηρετητή. Το τελικό αποτέλεσμα (για την έκδοση Laravel 5.1) μετά την μετάβαση στην ιστοσελίδα πρέπει να είναι αυτό που απεικονίζεται στο Σχήμα 4.1.

Laravel 5

Σχήμα 4.1: Αρχική σελίδα καινούργιου Laravel project

4.2 Διαδρομές

Οι διαδρομές (routes) είναι αυτές που δείχνουν στο Laravel τι να εκτελέσει όταν πραγματοποιείται μια αίτηση σε ένα συγκεκριμένο URL. Οι διαδρομές ορίζονται στο αρχείο `./app/Http/routes.php`. Οι διαδρομές ορίζονται με τον παρακάτω τρόπο: (Otwell, χ.η.-ε)

Listing 4.4: Παράδειγμα διαδρομών

```
Route::get('/', function () {  
    return 'Hello, World! (από GET)';  
});  
  
Route::post('/', function () {  
    return 'Hello, World! (από POST)';  
});
```

Για παράδειγμα, αν το Laravel έχει εγκατασταθεί σε τοπικό εξυπηρετητή, και γίνει μετάβαση στο `http://127.0.0.1/` (δηλαδή ένα απλό GET ερώτημα), θα εμφανιστεί το μήνυμα «Hello, World! (από GET)». Ομοίως, αν γίνει ένα POST ερώτημα στην ίδια διεύθυνση, θα εμφανιστεί το μήνυμα «Hello, World! (από POST)».

Μπορούν επίσης να περαστούν και παράμετροι μέσα από τη διαδρομή:

Listing 4.5: Παράδειγμα διαδρομών

```
Route::get('name/{name}', function ($name) {  
    return 'Γεια σου ' . $name;  
});
```

Έτσι, αν γίνει μετάβαση στη σελίδα `http://127.0.0.1/name/John`, τότε θα εμφανιστεί το μήνυμα «Γεια σου John».

4.3 Ελεγκτές

Αντί να γράφεται ολόκληρη η λογική της ιστοσελίδας στο αρχείο των διαδρομών, μπορούν να δημιουργηθούν ελεγκτές (controllers) που θα εκτελούν τις ίδιες διαδικασίες. Οι ελεγκτές αποθηκεύονται στον φάκελο `./app/Http/Controllers`. Η απλούστερη μορφή ενός ελεγκτή είναι η παρακάτω: (Otwell, χ.η.-d)

Listing 4.6: Παράδειγμα ελεγκτή

```
<?php

namespace App\Http\Controllers;

use App\Http\Controllers\Controller;

class SayHelloController extends Controller
{
    public function sayHello($name)
    {
        return 'Γεια σου ' . $name;
    }
}
```

Για να χρησιμοποιηθεί ο ελεγκτής, πρέπει να οριστεί στο αρχείο `routes.php` ως εξής:

Listing 4.7: Χρήση ελεγκτή σε διαδρομή

```
Route::get('name/{name}', 'SayHelloController@sayHello');
```

4.4 Πρότυπα Blade

Η μηχανή προτύπων Blade παρέχει έναν εύκολο και εύχρηστο τρόπο διαχωρισμού του HTML κώδικα από τον PHP κώδικα (Otwell, χ.η.-b). Επίσης, επιτρέπει την επαναχρησιμοποίηση κάποιου master layout. Έτσι, το τμήμα κώδικα :

Listing 4.8: HTML μέσα σε PHP

```
Route::get('name/{name}', function ($name) {  
    $output = '  
        <!doctype html>  
        <html lang="el">  
            <head>  
                <meta charset="utf-8">  
                <title>Η ιστοσελίδα μου</title>  
            </head>  
            <body>  
                Γεια σου ';  
    $output .= $name;  
    $output .= '</body>  
        </html>';  
  
    return $output;  
});
```

Μπορούν να γραφτούν τα παρακάτω:

Listing 4.9: ./resources/views/master.blade.php

```
<!doctype html>  
<html lang="el">  
    <head>  
        <meta charset="utf-8">  
        <title>Η ιστοσελίδα μου</title>  
    </head>  
    <body>  
        <div id="content">  
            @yield('content')  
        </div>  
    </body>  
</html>
```

Listing 4.10: ./resources/views/hello.blade.php

```
@extends('master')

@section('content')
    Γεια σου {{$name}}
@endsection
```

Listing 4.11: ./app/Http/routes.php

```
Route::get('name/{name}', function ($name) {
    return view('hello', ['name' => $name]);
});
```

Επίσης, το Blade υποστηρίζει δομές ελέγχου όπως if-then-else, for, while, κτλ.

4.5 Artisan

Το Artisan, ένα εργαλείο γραμμής εντολών (command-line) που συμπεριλαμβάνεται με το Laravel, παρέχει πολλές χρήσιμες εντολές (Otwell, χ.η.-α). Για την εμφάνιση όλων των εντολών, μπορεί να χρησιμοποιηθεί η παρακάτω εντολή:

```
php artisan list
```

Listing 4.12: Αποτέλεσμα εκτέλεσης εντολής php artisan list

```
$ php artisan list
Laravel Framework version 5.1.17 (LTS)

Usage:
  command [options] [arguments]

Options:
  -h, --help            Display this help message
  -q, --quiet           Do not output any message
  -V, --version         Display this application version
  --ansi               Force ANSI output
  --no-ansi            Disable ANSI output
  -n, --no-interaction Do not ask any interactive question
  --env[=ENV]          The environment the command should run under.
  -v|vv|vvv, --verbose Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

Available commands:
  clear-compiled  Remove the compiled class file
  down           Put the application into maintenance mode
```

Πτυχιακή εργασία του φοιτητή Κοσματοπούλου Μιχαήλ

env	Display the current framework environment
help	Displays help for a command
inspire	Display an inspiring quote
list	Lists commands
migrate	Run the database migrations
optimize	Optimize the framework for better performance
serve	Serve the application on the PHP development server
tinker	Interact with your application
up	Bring the application out of maintenance mode
app	
app:name	Set the application namespace
auth	
auth:clear-resets	Flush expired password reset tokens
cache	
cache:clear	Flush the application cache
cache:table	Create a migration for the cache database table
config	
config:cache	Create a cache file for faster configuration loading
config:clear	Remove the configuration cache file
db	
db:seed	Seed the database with records
event	
event:generate	Generate the missing events and listeners based on registration
handler	
handler:command	Create a new command handler class
handler:event	Create a new event handler class
key	
key:generate	Set the application key
make	
make:command	Create a new command class
make:console	Create a new Artisan command
make:controller	Create a new resource controller class
make:event	Create a new event class
make:job	Create a new job class
make:listener	Create a new event listener class
make:middleware	Create a new middleware class
make:migration	Create a new migration file
make:model	Create a new Eloquent model class
make:policy	Create a new policy class
make:provider	Create a new service provider class
make:request	Create a new form request class
make:seeder	Create a new seeder class
make:test	Create a new test class
migrate	
migrate:install	Create the migration repository
migrate:refresh	Reset and re-run all migrations
migrate:reset	Rollback all database migrations
migrate:rollback	Rollback the last database migration
migrate:status	Show the status of each migration
queue	
queue:failed	List all of the failed queue jobs
queue:failed-table	Create a migration for the failed queue jobs database table
queue:flush	Flush all of the failed queue jobs
queue:forget	Delete a failed queue job
queue:listen	Listen to a given queue
queue:restart	Restart queue worker daemons after their current job
queue:retry	Retry a failed queue job
queue:subscribe	Subscribe a URL to an Iron.io push queue
queue:table	Create a migration for the queue jobs database table
queue:work	Process the next job on a queue
route	
route:cache	Create a route cache file for faster route registration
route:clear	Remove the route cache file
route:list	List all registered routes
schedule	
schedule:run	Run the scheduled commands
session	
session:table	Create a migration for the session database table
vendor	
vendor:publish	Publish any publishable assets from vendor packages
view	
view:clear	Clear all compiled view files

Μια χρήσιμη εντολή είναι η `tinker` που επιτρέπει την εκτέλεση PHP κώδικα σε

πραγματικό χρόνο, κάνοντας έτσι πολύ πιο βολικές τις δοκιμές καινούργιου κώδικα.

Listing 4.13: Αποτέλεσμα εκτέλεσης εντολής `php artisan tinker`

```
$ php artisan tinker
Psy Shell v0.5.2 (PHP 5.5.15 - cli) by Justin Hileman
>>> $a = 2;
=> 2
>>> $b = 5;
=> 5
>>> $a + $b;
=> 7
>>>
```

4.6 Μεταναστεύσεις

Οι μεταναστεύσεις (migrations) περιγράφουν τις αλλαγές που έγιναν στο σχήμα της βάσης δεδομένων της εφαρμογής, δίνοντας έτσι την δυνατότητα μια ομάδα προγραμματιστών να έχουν πάντα την ίδια έκδοση του σχήματος, ανεξαρτήτως από ποιον υπολογιστή εργάζονται. Για την δημιουργία μιας μετανάστευσης χρησιμοποιείται η εντολή `make:migration` του `artisan`:

Listing 4.14: Αποτέλεσμα εκτέλεσης εντολής `php artisan tinker`

```
$ php artisan make:migration create_students_table --create=students
Created Migration: 2015_09_20_212312_create_students_table
```

Οι μεταναστεύσεις αποθηκεύονται στον φάκελο `./database/migrations/` και είναι μια PHP κλάση που επεκτείνει την κλάση `Migration`. Υλοποιεί την συνάρτηση `up()`; που εκτελείται κατά την εκτέλεση της μετανάστευσης, και την συνάρτηση `down()`; που εκτελείται κατά την κατάργηση της μετανάστευσης. Ένα ολοκληρωμένο παράδειγμα μιας μετανάστευσης φαίνεται παρακάτω:

Listing 4.15: database/migrations/2015_09_20_212312_create_students_table.php

```

<?php
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreateStudentsTable extends Migration {

    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('students', function(Blueprint $table)
        {
            $table->increments('id');
            $table->string('am')->unique();
            $table->string('first_name');
            $table->string('last_name');
            $table->string('father_name');
            $table->integer('in_year_period_id')->unsigned();
            $table->integer('in_semester')->unsigned();

            $table->foreign('in_year_period_id')->references('id')
                ->on('year_periods');
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::drop('students');
    }
}

```

Για την εκτέλεση των μεταναστεύσεων πρέπει να χρησιμοποιηθεί η εντολή `artisan migrate`.

4.7 Eloquent

Το Eloquent είναι ένα σύστημα ORM (Object-relational mapping — Αντικειμενο-σχεσιακή αντιστοίχιση) που κάνει εύκολη την διαχείριση της βάσης δεδομένων της ιστοσελίδας. Κάθε πίνακας της βάσης έχει ένα αντίστοιχο μοντέλο, που αλληλεπιδρά με τον συγκεκριμένο πίνακα. Το Eloquent κάνει εύκολη την άντληση και εισαγωγή δεδομένων από και προς τη βάση δεδομένων (Otwell, χ.η.-c).

Για την δημιουργία ενός μοντέλου χρησιμοποιείται η εντολή `artisan make:model`:

Listing 4.16: Αποτέλεσμα εκτέλεσης εντολής `php artisan make:model`

```
$ php artisan make:model Student
Model created successfully.
```

Ο κώδικας του μοντέλου που δημιουργείται είναι ο εξής:

Listing 4.17: `./app/Student.php`

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Student extends Model
{
    //
}
```

Η σύμβαση λέει ότι το όνομα του μοντέλου θα πρέπει να είναι ο ενικός αριθμός του ονόματος του πίνακα (π.χ. αν το όνομα του πίνακα είναι `students`, το όνομα του μοντέλου θα πρέπει να είναι `Student`). Αυτό είναι δυνατόν να αλλάξει μέσω της μεταβλητής `$table` στην κλάση του μοντέλου.

Χωρίς να έχει γραφτεί καθόλου κώδικας στην κλάση του μοντέλου, είναι δυνατόν να επεξεργασθούν τα στοιχεία του αντίστοιχου πίνακα στη βάση:

Listing 4.18: Παράδειγμα χρήσης Eloquent

```
$students = Student::all();  
// η μεταβλητή $students περιέχει τώρα όλους τους φοιτητές  
  
$newStudent = new Student;  
  
$newStudent->first_name = "ΚΩΝΣΤΑΝΤΙΝΟΣ";  
$newStudent->last_name = "ΠΑΠΑΔΟΠΟΥΛΟΣ";  
// ...  
  
$newStudent->save();  
// έχει εισαχθεί μια καινούργια γραμμή στον πίνακα students  
// με τα στοιχεία του $newStudent
```

4.7.1 Συσχετίσεις

Το Eloquent υποστηρίζει (μεταξύ άλλων) τις παρακάτω συσχετίσεις μεταξύ των μοντέλων (Otwell, χ.η.-c):

- Ένα προς ένα
- Ένα προς πολλά
- Πολλά προς πολλά

4.7.1.1 Ένα προς ένα

Για τον ορισμό μιας συσχέτισης ένα προς ένα, πρέπει μέσα στο μοντέλο. Για παράδειγμα, έστω ότι κάθε φοιτητής διαθέτει ένα τηλέφωνο. Ο τρόπος συσχέτισης γίνεται όπως φαίνεται παρακάτω:

Listing 4.19: ./app/Student.php - Συσχέτιση με Phone

```
<?php  
  
namespace App;  
  
use Illuminate\Database\Eloquent\Model;  
  
class Student extends Model
```

```
{  
    public function phone()  
    {  
        return $this->hasOne('App\Phone');  
    }  
}
```

Η αντίστροφη συσχέτιση ορίζεται ως εξής:

Listing 4.20: ./app/Phone.php - Συσχέτιση με Student

```
<?php  
  
namespace App;  
  
use Illuminate\Database\Eloquent\Model;  
  
class Phone extends Model  
{  
    public function student()  
    {  
        return $this->belongsTo('App\Student');  
    }  
}
```

4.7.1.2 Ένα προς πολλά

Στις συσχετίσεις ένα προς πολλά, ένα μοντέλο μπορεί να «κατέχει» ένα πλήθος άλλων μοντέλων. Για παράδειγμα, μια ομάδα μαθημάτων αποτελείται από πολλά μαθήματα, αλλά κάθε μάθημα μπορεί να έχει το πολύ μία ομάδα. Αυτό εκφράζεται με την συνάρτηση `hasMany()`:

Listing 4.21: ./app/CourseGroup.php - Συσχέτιση με Course

```
<?php  
  
namespace App;  
  
use Illuminate\Database\Eloquent\Model;
```

```
class CourseGroup extends Model {  
    public function courses() {  
        return $this->hasMany(' \App\Course', 'course_group');  
    }  
}
```

Η αντίστροφη συσχέτιση ορίζεται και πάλι με την belongsTo():

Listing 4.22: ./app/Course.php - Συσχέτιση με CourseGroup

```
<?php  
  
namespace App;  
  
use Illuminate\Database\Eloquent\Model;  
  
class Course extends Model {  
    public function courseGroup() {  
        return $this->belongsTo(' \App\CourseGroup', 'course_group',  
                                'id');  
    }  
}
```

Έχοντας πλέον ορίσει τις παραπάνω συσχετίσεις, καθίσταται εύκολη η μετάβαση από ένα μάθημα προς όλα τα μαθήματα που ανήκουν στην ίδια ομάδα:

Listing 4.23: Παράδειγμα χρήσης συσχετίσεων

```
$php artisan tinker  
Psy Shell v0.4.4 (PHP 5.5.15 - cli) by Justin Hileman  
>>> \App\Course::first()  
=> <App\Course #00000000315909ae0000000043a47b63> {  
    id: 308,  
    course_id: "269-1041",  
    course_code: "1041",  
    course_title: "ΜΑΘΗΜΑΤΙΚΗ ΑΝΑΛΥΣΗ",  
    course_group: 34  
}  
>>> \App\Course::first()->courseGroup  
=> <App\CourseGroup #00000000315909d20000000043a47b63> {  
    id: 34,  
    course_title: "ΜΑΘΗΜΑΤΙΚΗ ΑΝΑΛΥΣΗ"  
}  
>>> \App\Course::first()->courseGroup->courses  
=> <Illuminate\Database\Eloquent\Collection #00000000315909d00000000043a47b63> [  
    <App\Course #00000000315909d70000000043a47b63> {
```

```
        id: 308,  
        course_id: "269-1041",  
        course_code: "1041",  
        course_title: "ΜΑΘΗΜΑΤΙΚΗ ΑΝΑΛΥΣΗ",  
        course_group: 34  
    },  
    <App\Course #00000000315909d60000000043a47b63> {  
        id: 472,  
        course_id: "269-4104",  
        course_code: "4104",  
        course_title: "Μαθηματική Ανάλυση",  
        course_group: 34  
    },  
    <App\Course #00000000315909d50000000043a47b63> {  
        id: 490,  
        course_id: "269-31041",  
        course_code: "31041",  
        course_title: "ΜΑΘΗΜΑΤΙΚΗ ΑΝΑΛΥΣΗ",  
        course_group: 34  
    }  
}
```

4.7.1.3 Πολλά προς πολλά

Η αντιστοίχιση πολλά προς πολλά γίνεται με παρόμοιο τρόπο επιστρέφοντας το αποτέλεσμα της συνάρτησης `belongsToMany()` ; .

Επίλογος

Σε αυτό το κεφάλαιο αναλύθηκαν κάποιες από τις κυριότερες λειτουργίες που προσφέρει το Laravel. Το Laravel παρέχει πολλά περισσότερα πράγματα, και ο ενδιαφερόμενος αναγνώστης μπορεί να μάθει περισσότερα στην επίσημη τεκμηρίωση του Laravel:

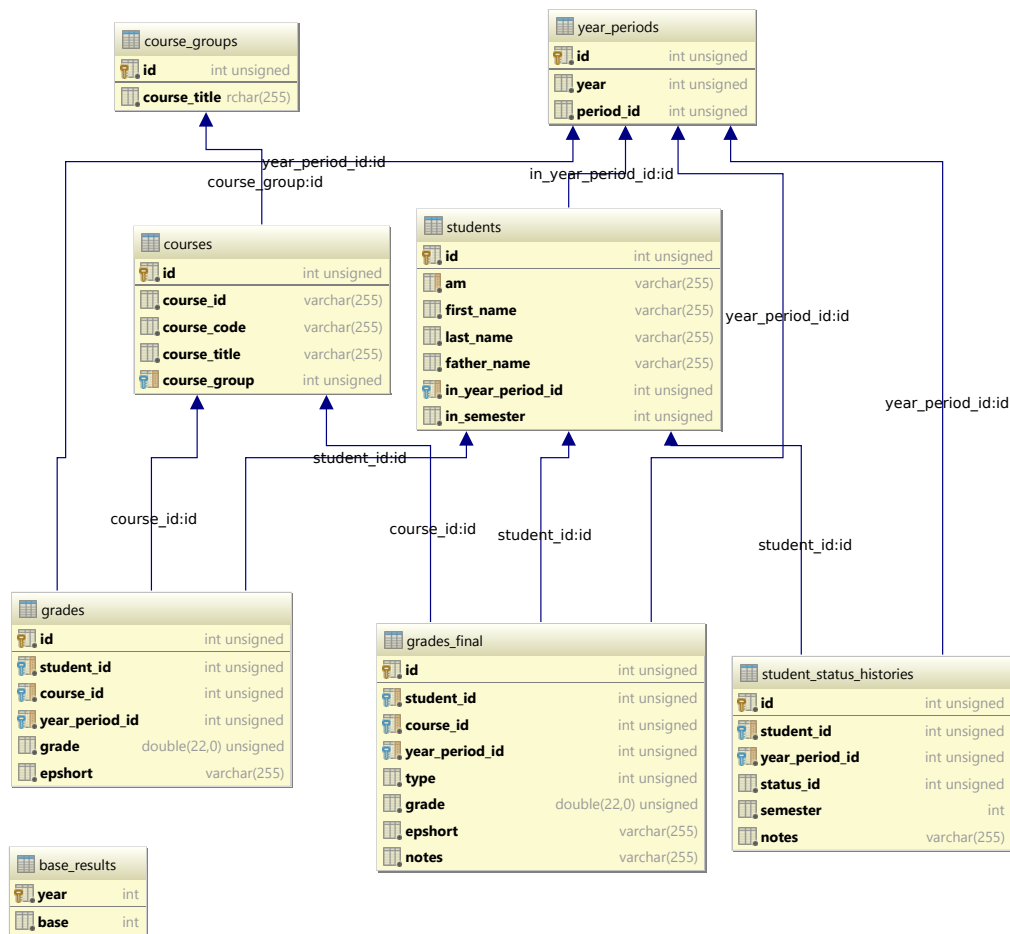
<http://laravel.com/docs/>

5 Υλοποίηση πτυχιακής εργασίας

Εισαγωγή

Σε αυτό το κεφάλαιο θα δοθούν μερικές τεχνικές λεπτομέρειες που αφορούν την υλοποίηση της πτυχιακής εργασίας.

5.1 Σχήμα βάσης δεδομένων



Σχήμα 5.1: Σχήμα βάσης δεδομένων

Στο σχήμα 5.1 φαίνεται το σχήμα της βάσης δεδομένων. Παρακάτω περιγράφεται αναλυτικότερα κάθε πίνακας.

5.1.1 year_periods

Ο πίνακας αυτός περιέχει τα ID των συνδυασμών έτους και χρονολογίας.

Πίνακας 5.1: Στήλες πίνακα year_periods

Στήλη	Περιγραφή
<u>id</u>	Κύριο κλειδί πίνακα
year	Χρονιά
period_id	Τύπος εξαμήνου

5.1.2 students

Ο πίνακας αυτός περιέχει τα στοιχεία των φοιτητών.

Πίνακας 5.2: Στήλες πίνακα students

Στήλη	Περιγραφή
<u>id</u>	Κύριο κλειδί πίνακα
am	A.M. φοιτητή
first_name	Όνομα φοιτητή
last_name	Επώνυμο φοιτητή
father_name	Πατρώνυμο φοιτητή
<i>in_year_period_id</i>	Ξένο κλειδί στον πίνακα year_periods
in_semester	Αριθμός εξαμήνου εισαγωγής του φοιτητή

5.1.3 student_status_histories

Αυτός ο πίνακας περιέχει το ιστορικό των καταστάσεων του συγκεκριμένου φοιτητή, ανά κάθε εξάμηνο που ενημερώνεται η βάση με το αρχείο students.txt.

Πίνακας 5.3: Στήλες πίνακα student_status_histories

Στήλη	Περιγραφή
<u>id</u>	A/A γραμμών
student_id	Ξένο κλειδί στον πίνακα students
year_period_id	Ξένο κλειδί στον πίνακα year_periods
status_id	Η κατάσταση του φοιτητή στο συγκεκριμένο year_period_id
semester	Το εξάμηνο που βρίσκονταν ο φοιτητής
notes	Σημειώσεις σχετικά με τον φοιτητή

5.1.4 course_groups

Σε αυτόν τον πίνακα αποθηκεύονται οι τίτλοι των ομάδων των μαθημάτων.

Πίνακας 5.4: Στήλες πίνακα course_groups

Στήλη	Περιγραφή
<u>id</u>	A/A γραμμής
course_title	Όνομα ομάδας

5.1.5 courses

Σε αυτόν τον πίνακα αποθηκεύονται πληροφορίες σχετικά με τα μαθήματα.

Πίνακας 5.5: Στήλες πίνακα courses

Στήλη	Περιγραφή
<u>id</u>	A/A γραμμών
course_id	Ο κωδικός του μαθήματος
course_code	Ο κωδικός του μαθήματος στο πρόγραμμα σπουδών
course_title	Ο τίτλος του μαθήματος
course_group	Ξένο κλειδί στον πίνακα course_groups

5.1.6 grades

Σε αυτόν τον πίνακα αποθηκεύονται όλες οι βαθμολογίες των φοιτητών.

Πίνακας 5.6: Στήλες πίνακα grades

Στήλη	Περιγραφή
<u>id</u>	A/A γραμμών
<i>student_id</i>	Ξένο κλειδί στον πίνακα students
<i>course_id</i>	Ξένο κλειδί στον πίνακα courses
<i>year_period_id</i>	Ξένο κλειδί στον πίνακα year_periods
grade	Βαθμός φοιτητή
ershort	Περίοδος εξεταστικής

5.1.7 grades_final

Σε αυτόν τον πίνακα αποθηκεύονται όλες οι τελικές βαθμολογίες των φοιτητών.

Πίνακας 5.7: Στήλες πίνακα grades_final

Στήλη	Περιγραφή
<u>id</u>	A/A γραμμών
<i>student_id</i>	Ξένο κλειδί στον πίνακα students
<i>course_id</i>	Ξένο κλειδί στον πίνακα courses
<i>year_period_id</i>	Ξένο κλειδί στον πίνακα year_periods
type	Δείχνει εάν ο βαθμός χρειάστηκε κάποια ιδιαίτερη μεταχείριση
grade	Βαθμός φοιτητή
ershort	Περίοδος εξεταστικής
notes	Σημειώσεις για την συγκεκριμένη βαθμολογία

5.1.8 base_results

Ο πίνακας αυτός περιλαμβάνει τις βάσεις εισαγωγής για κάθε έτος.

Πίνακας 5.8: Στήλες πίνακα base_results

Στήλη	Περιγραφή
<u>year</u>	Έτος
<u>base</u>	Βάση εισαγωγής

5.2 Όψεις βάσης δεδομένων

Εκτός από τους πίνακες, έχουν γραφτεί και μερικές όψεις για διευκόλυνση. Οι σημαντικότερες από αυτές παρουσιάζονται στα επόμενα κεφάλαια.

5.2.1 stats_students_unprocessed

Η όψη αυτή επιστρέφει το άθροισμα των φοιτητών ανά έτος εισαγωγής, ανά κατάσταση.

Listing 5.1: Δημιουργία όψης stats_students_unprocessed

```
CREATE OR REPLACE view stats_students_unprocessed
AS
SELECT `s`.`in_year` AS `in_year`,
       `h`.`status_id` AS `status_id`,
       count(0) AS `count(*)`
FROM   (`students_year_periods` `s`
        JOIN `student_status_histories` `h`
            ON (( `s`.`id` = `h`.`student_id` )))
WHERE  h.year_period_id = (SELECT latest_year_period.id
                          FROM   latest_year_period)
GROUP BY `s`.`in_year`,
         `h`.`status_id`
ORDER BY `s`.`in_year` DESC,
         `h`.`status_id`;
```

Πίνακας 5.9: Παράδειγμα αποτελέσματος όψης stats_students_unprocessed

in_year	status_id	count(*)
2014	1	250
2014	2	4
2014	4	3
2013	1	169
2013	2	18
2013	4	4

5.2.2 stats_students

Η όψη αυτή επιστρέφει μία γραμμή για κάθε χρονιά, και περιλαμβάνει τον αριθμό των εισακτέων, διαγραμμένων, πτυχιούχων, και σε αναστολή φοιτητών της χρονιάς εκείνης. Βασίζεται επάνω στην όψη stats_students_unprocessed.

Listing 5.2: Δημιουργία όψης stats_students

```
CREATE OR REPLACE view stats_students
AS
SELECT `s`.`in_year` AS `in_year`,
       Sum(IF((`s`.`status_id` = 1), `s`.`count(*)`, 0)) AS `inserted`,
       Sum(IF((`s`.`status_id` = 2), `s`.`count(*)`, 0)) AS `deleted`,
       Sum(IF((`s`.`status_id` = 3), `s`.`count(*)`, 0)) AS `graduated`,
       Sum(IF((`s`.`status_id` = 4), `s`.`count(*)`, 0)) AS `stopped`
FROM   `stats_students_unprocessed` `s`
GROUP BY `s`.`in_year`
ORDER BY `s`.`in_year` DESC;
```

Πίνακας 5.10: Παράδειγμα αποτελέσματος όψης stats_students

in_year	inserted	deleted	graduated	stopped
2014	250	4	0	3
2013	169	18	0	4
2012	114	37	0	4
2011	107	22	0	3
2010	82	42	3	6

5.2.3 stats_deletions_unprocessed

Η όψη αυτή επιστρέφει τον αριθμό των φοιτητών που κινδυνεύουν να διαγραφούν σε επόμενα εξάμηνα. Για τους φοιτητές που μπήκαν από το 2010 και μετά, εφαρμόζεται ο κανόνας $n + 2$, ενώ για τους υπόλοιπους εφαρμόζεται ο κανόνας $2n$.

Listing 5.3: Δημιουργία όψης stats_deletions_unprocessed

```
CREATE view stats_deletions_unprocessed
AS
SELECT CASE
```

```

        WHEN in_year < 2010 THEN 2 * ( 2 * 4 ) - semester
        ELSE 2 * ( 4 + 2 ) - semester
    end      AS remaining_semesters,
    Count(*) AS count
FROM      students_year_periods
        INNER JOIN student_status_histories
            ON students_year_periods.id =
                student_status_histories.student_id
WHERE     status_id = 1
        AND student_status_histories.year_period_id = (SELECT id
                                                         FROM
                                                         latest_year_period)
GROUP BY remaining_semesters
HAVING   remaining_semesters <= 2
ORDER BY remaining_semesters;

```

Πίνακας 5.11: Παράδειγμα αποτελέσματος όψης stats_deletions_unprocessed

remaining_semesters	count
0	25
1	20
2	110

5.2.4 stats_deletions_semester

Αυτή η όψη επιστρέφει τον αριθμό των φοιτητών που έχουν διαγραφεί από κάθε εξάμηνο.

Listing 5.4: Δημιουργία όψης stats_deletions_semester

```

CREATE view stats_deletions_semester
AS
    SELECT semester,
           Count(*) AS count
    FROM   student_status_histories
    WHERE  status_id = 2
        AND student_status_histories.year_period_id =
            (SELECT id FROM latest_year_period)
    GROUP BY semester;

```

Πίνακας 5.12: Παράδειγμα αποτελέσματος όψης stats_deletions_semester

semester	count
1	276
2	216
3	123
4	108
5	67

5.2.5 stats_suspensions

Η συγκεκριμένη όψη επιστρέφει τα ονόματα των φοιτητών που βρίσκονται σε αναστολή, το έτος εισαγωγής τους, και των αριθμό των εξαμήνων που βρίσκονται σε αναστολή.

Listing 5.5: Δημιουργία όψης stats_suspensions

```
CREATE OR REPLACE view stats_suspensions
AS
SELECT am,
       first_name,
       last_name,
       p.in_year,
       Cast(lyp.year - p.in_year AS signed) * 2 +
       Cast(lyp.period_id AS signed)
       - Cast(in_period AS signed)
       + in_semester - Cast(semester AS signed) + 1 AS
suspended_semesters
FROM   latest_year_period AS lyp,
       students_year_periods AS p
INNER JOIN student_status_histories h
        ON h.student_id = p.id
WHERE  status_id = 4
AND    h.year_period_id = lyp.id
ORDER BY suspended_semesters DESC,
         am;
```

Πίνακας 5.13: Παράδειγμα αποτελέσματος όψης stats_suspensions

am	first_name	last_name	in_year	suspended_semesters
022076	ΠΕΤΡΟΣ	ΙΚΚΟΥ	2002	8
103653	ΓΑΡΥΦΑΛΛΟΣ	ΚΑΛΥΒΑΣ	2010	8
103666	ΓΕΩΡΓΙΟΣ	ΤΟΥΡΤΟΥΡΑΣ	2010	8
113711	ΣΟΛΩΝ	ΒΟΥΓΑΣ	2011	8
032423	ΦΩΤΕΙΝΗ	ΜΙΧΑΗΛΙΔΟΥ	2003	7

6 Συμπεράσματα

Στη συγκεκριμένη πτυχιακή εργασία αναπτύχθηκε και υλοποιήθηκε ένα σύστημα, το οποίο μπορεί με αυτοματοποίηση να εξάγει συγκεκριμένα στατιστικά στοιχεία. Η χρήση του συστήματος αυτού εξασφαλίζει ταχύτητα και αξιοπιστία στην οργάνωση του Τμήματος Μηχανικών Πληροφορικής του ΑΤΕΙΘ. Ο λόγος της υλοποίησης αυτού του συστήματος ήταν η επίλυση του θέματος της χρονοβόρας διαδικασίας που έπρεπε να ακολουθηθεί από τον υπεύθυνο καθηγητή για την εξαγωγή των ίδιων στατιστικών.

Κατά την ανάπτυξη του συστήματος θεωρήθηκε σκόπιμη η δημιουργία μιας επιπλέον λειτουργίας (καρτέλα φοιτητή), που περιέχει τα στοιχεία του φοιτητή, το ιστορικό κατάστασης του, τις βαθμολογίες του, και τον μέσο όρο εξαμήνων.

7 Προτάσεις για μελλοντική έρευνα

Το σύστημα αυτό θα μπορούσε να επεκταθεί έτσι ώστε να έχει τη δυνατότητα να εμφανίσει στατιστικά φοιτητών και από άλλα τμήματα της σχολής. Επίσης, θα μπορούσαν να δημιουργηθούν διαφορετικές ομάδες χρηστών, οι οποίες θα έχουν διαφορετικά δικαιώματα στα στατιστικά που μπορούν να δουν.

Βιβλιογραφία

- Apache Software Foundation. (χ.η.-a). Apache software foundation graphics. Ανακτήθηκε 14 Σεπτεμβρίου 2015, από <http://www.apache.org/foundation/press/kit/>
- Apache Software Foundation. (χ.η.-b). Welcome! - the apache HTTP server project. Ανακτήθηκε 14 Σεπτεμβρίου 2015, από <http://httpd.apache.org/>
- Beal, V. (χ.η.). What is world wide web? a webopedia definition. Ανακτήθηκε 11 Σεπτεμβρίου 2015, από http://www.webopedia.com/TERM/W/World_Wide_Web.html
- Braden, R. (1989, Οκτωβρίου). Requirements for internet hosts - communication layers - RFC 1112. Ανακτήθηκε 9 Σεπτεμβρίου 2015, από <https://tools.ietf.org/html/rfc1122#page-7>
- CERN. (1994, Ιουλίου 11). Tim berners-lee, world wide web inventor [Bulletin issue: 37/2004, mon 13 sep 2004, le dauphine libere (bellegarde, france) : 7 jul 1998, bulletin issue: 35/2006 & 36/2006, mon 28 aug 2006, CERN bulletin no 26/98 (21 june 1998) (english), the guardian : 2 jul 1998, CERN bulletin 18/99 : 3 may 1999 (french), le pays gessien : 3 jul 1998, CERN courier vol 39 no 9 : november 1999, le temps : 27 jun 1998, CERN bulletin no 26/98 (21 juin 1998) (french), CERN bulletin 18/99 : 3 may 1999 (english), biography]. Ανακτήθηκε 11 Σεπτεμβρίου 2015, από <http://cds.cern.ch/record/39437>
- Composer. (χ.η.). Introduction - composer. Ανακτήθηκε 19 Σεπτεμβρίου 2015, από <https://getcomposer.org/doc/00-intro.md>
- NSSDC & NASA. (2004). *Replica of sputnik 1*. Ανακτήθηκε 11 Σεπτεμβρίου 2015, από https://commons.wikimedia.org/wiki/File:Sputnik_asm.jpg
- Oracle Corporation. (χ.η.). MySQL :: MySQL 5.1 reference manual :: 1.3.1 what is MySQL? Ανακτήθηκε 19 Σεπτεμβρίου 2015, από <https://dev.mysql.com/doc/refman/5.1/en/what-is-mysql.html>
- Internet*. (2015, Σεπτεμβρίου 2), Στο *Wikipedia, the free encyclopedia*. Page Version ID: 679123365. Ανακτήθηκε 11 Σεπτεμβρίου 2015, από <https://en.wikipedia.org/w/index.php?title=Internet&oldid=679123365>

- Otwell, T. (χ.η.-a). Artisan console - laravel - the PHP framework for web artisans. Ανακτήθηκε 20 Σεπτεμβρίου 2015, από <http://laravel.com/docs/5.1/artisan>
- Otwell, T. (χ.η.-b). Blade templates - laravel - the PHP framework for web artisans. Ανακτήθηκε 20 Σεπτεμβρίου 2015, από <http://laravel.com/docs/5.1/blade>
- Otwell, T. (χ.η.-c). Eloquent: getting started - laravel - the PHP framework for web artisans. Ανακτήθηκε 20 Σεπτεμβρίου 2015, από <http://laravel.com/docs/5.1/eloquent>
- Otwell, T. (χ.η.-d). HTTP controllers - laravel - the PHP framework for web artisans. Ανακτήθηκε 20 Σεπτεμβρίου 2015, από <http://laravel.com/docs/5.1/controllers>
- Otwell, T. (χ.η.-e). HTTP routing - laravel - the PHP framework for web artisans. Ανακτήθηκε 20 Σεπτεμβρίου 2015, από <http://laravel.com/docs/5.1/routing>
- Otwell, T. (χ.η.-f). Installation - laravel - the PHP framework for web artisans. Ανακτήθηκε 20 Σεπτεμβρίου 2015, από <http://laravel.com/docs/5.1>
- Palat, J. (2012, Νοεμβρίου 14). Introducing vagrant | linux journal. Ανακτήθηκε 19 Σεπτεμβρίου 2015, από <http://www.linuxjournal.com/content/introducing-vagrant>
- PHP Group. (χ.η.). PHP: what is PHP? - manual. Ανακτήθηκε 14 Σεπτεμβρίου 2015, από <http://php.net/manual/en/intro-what-is.php>
- Stewart, W. (2000, Ιανουαρίου 7). Internet history one-page summary - how invented, created. Ανακτήθηκε 11 Σεπτεμβρίου 2015, από http://www.livinginternet.com/i/ii_summary.htm
- White House. (1959, Μαΐου 29). *Dwight d. eisenhower photo portrait*. Ανακτήθηκε 11 Σεπτεμβρίου 2015, από https://commons.wikimedia.org/wiki/File:Dwight_D._Eisenhower,_official_photo_portrait,_May_29,_1959.jpg

Παράρτημα Α

Δείγμα students.txt

Στον παρακάτω πίνακα φαίνεται ένα μικρό δείγμα (σε μορφή πίνακα) του αρχείου students.txt.

Πίνακας Α.1: Δείγμα students.txt

student_ID	spec_aem	first	last	fname	in_year	in_period_ID	cond_ID	dgr_logos	in_exam_ID	exam_ID
6770	890013	ΑΘΑΝΑΣΙΟΣ	ΔΑΣΚΑΛΟΥΔΗΣ	ΕΥΑΓΓΕΛΟΥ	1989	1	3		1	55
6771	890301	ΖΩΙΤΣΑ	ΣΑΒΒΑ	ΜΑΡΚΟΥ	1989	1	3		1	49
6772	870001	ΣΕΒΑΣΤΗ	ΚΑΝΟΥΡΑ	ΒΑΣΙΛΕΙΟΥ	1987	2	3		1	18
6773	870005	ΛΑΜΠΡΟΣ	ΦΡΟΥΝΤΖΟΣ	ΓΕΩΡΓΙΟΥ	1987	2	3		1	28
6774	870007	ΒΑΣΙΛΕΙΟΣ	ΤΑΟΥΚΤΣΗΣ	ΑΘΑΝΑΣΙΟΥ	1987	1	3		1	35
6775	870008	ΓΡΗΓΟΡΙΟΣ	ΣΤΑΥΡΙΔΗΣ	ΒΑΣΙΛΕΙΟΥ	1987	2	3		1	18
6776	870009	ΧΑΡΑΛΑΜΠΟΣ	ΣΠΥΡΟΠΟΥΛΟΣ	ΔΗΜΗΤΡΙΟΥ	1987	1	3		1	19
6777	870011	ΙΓΝΑΤΙΟΣ	ΣΑΠΤΣΟΓΛΟΥ	ΙΩΑΝΝΗ	1987	2	3		1	18

Παράρτημα Β

Δείγμα classStudents.txt

Στον παρακάτω πίνακα φαίνεται ένα μικρό δείγμα (σε μορφή πίνακα) του αρχείου classStudents.txt.

Πίνακας Β.1: Δείγμα classStudents.txt

student_ID	spec_aem	courseID	coursecode	classtitle	year	periodID	grade	epshort
6793	870032	269-1041	1041	ΜΑΘΗΜΑΤΙΚΗ ΑΝΑΛΥΣΗ	1998	1	5	A ΧΕΙΜ
6793	870032	269-7031	7031	ΤΕΧΝΟΛΟΓΙΑ ΠΟΛΥΜΕΣΩΝ	1997	2	5	A ΕΑΡ
6793	870032	269-A5E	A5E	ΓΛΩΣΣΑ ΠΡΟΓΡ. BASIC	1987	2	5,1	A ΕΑΡ
6793	870032	269-B7Θ	B7Δ	ΑΓΓΛΙΚΑ ΙΙ-Θ	1988	2	5	A ΕΑΡ
6793	870032	269-Γ3Ε	Γ3Ε	ΓΛΩΣΣΑ ΠΡΟΓ.FORTRAN ΙΙ-Ε	1995	2	5	A ΕΑΡ
6793	870032	269-Γ4Δ	Γ4Δ	ΤΕΧΝΟΟΙΚΟΝ.ΑΝΑΛΥΣΗ ΙΙ	1995	2	8,9	A ΕΑΡ
6793	870032	269-Γ7Ε	Γ7Ε	ASSEMBLY	1989	2	8	A ΕΑΡ
6793	870032	269-Γ8Θ	Γ8Δ	ΑΓΓΛΙΚΑ ΙΙΙ-Θ	1989	1	7	A ΧΕΙΜ
6793	870032	269-ΣΤ11Δ	ΣΤ11Δ	ΣΕΜΙΝΑΡΙΟ	1990	1	5	A ΧΕΙΜ

Παράρτημα C

Δείγμα studentCourses.txt

Στον παρακάτω πίνακα φαίνεται ένα μικρό δείγμα (σε μορφή πίνακα) του αρχείου studentCourses.txt.

Πίνακας C.1: Δείγμα studentCourses.txt

student_ID	spec_aem	courseID	coursecode	title	cyear	cperiod	cregtypeID	epshort	cgrade	notes
7645	950720	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1998	1	0	A ΧΕΙΜ	7,5	
7646	950721	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1997	2	0	A ΕΑΡ	5	
7648	950723	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1997	2	0	A ΕΑΡ	6	
7653	950728	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1999	1	0	B ΧΕΙΜ	5	
7661	950736	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1997	1	0	A ΧΕΙΜ	7	
7666	950741	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	2000	2	0	A ΕΑΡ	5	
7669	950744	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1998	1	0	A ΧΕΙΜ	5	
7670	950745	269-302E	3022	ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ-Ε	1998	2	0	B ΕΑΡ	5	

Οδηγός Χρήσης Λογισμικού

Εισαγωγή

Σε αυτό το κεφάλαιο θα περιγραφεί η διαδικασία που πρέπει να ακολουθηθεί για την εγκατάσταση της εφαρμογής. Θα ακολουθήσει μετά ένας σύντομος οδηγός χρήσης.

Εγκατάσταση

Για την εγκατάσταση της εφαρμογής, πρέπει να πληρούνται οι προϋποθέσεις που αναφέρθηκαν στο 4.1.1. Επίσης, θα χρειαστεί κάποιος εξυπηρετητής ιστού (Apache, nginx, κτλ), και ένας διακομιστής MySQL.

Παρακάτω περιγράφονται τα βήματα που θα πρέπει να ακολουθήσει ο χρήστης για την εγκατάσταση της εφαρμογής:

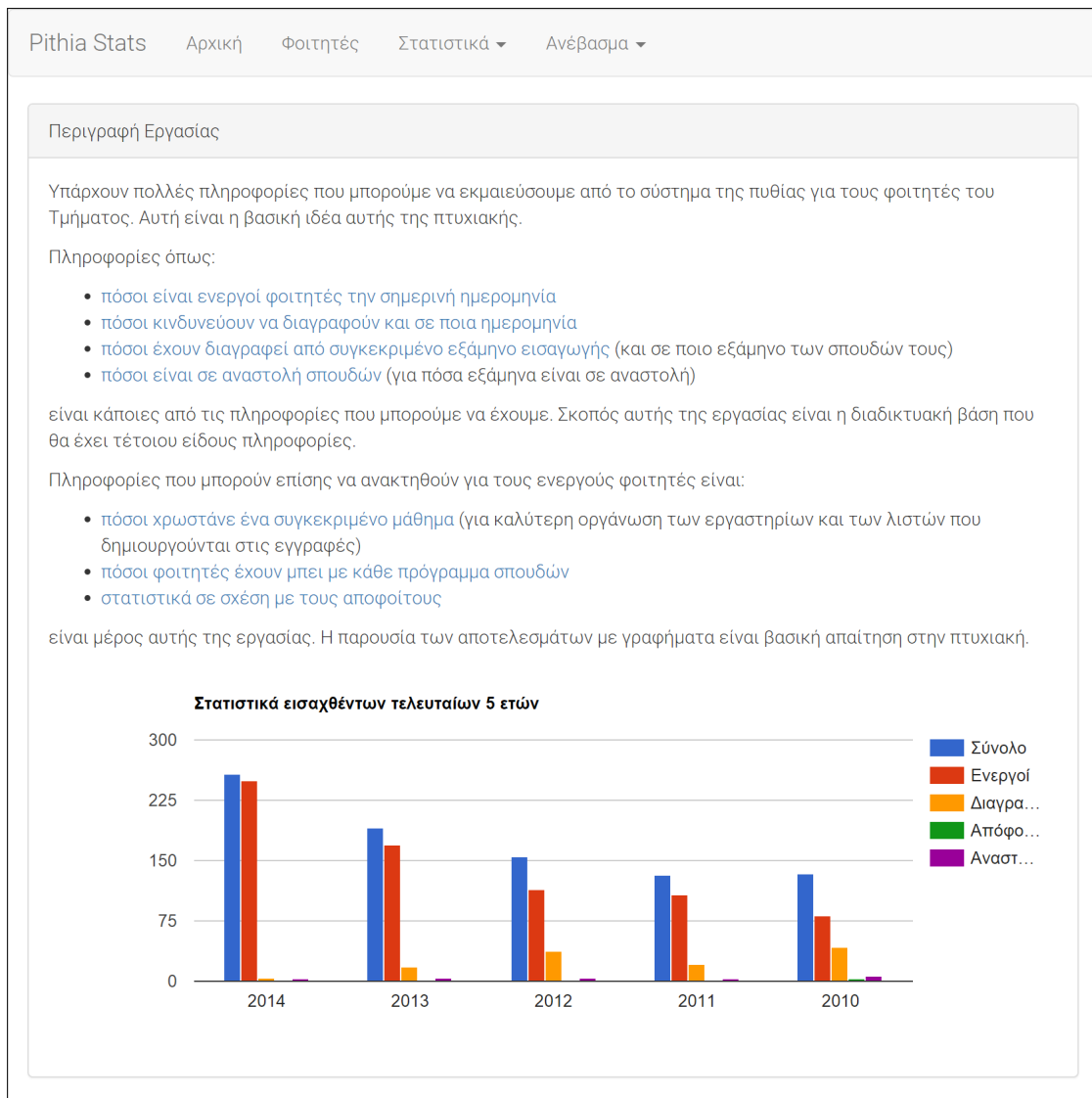
1. Ο χρήστης αντιγράφει τον κώδικα της εφαρμογής από το CD σε έναν φάκελο της αρεσκείας του
2. Μετά εκτελεί την εντολή `composer install` για να γίνει η λήψη όλων των απαραίτητων βιβλιοθηκών
3. Κάνει μετονομασία το αρχείο `.env.example` σε `.env`, και το τροποποιεί βάζοντας τις σωστές τιμές στα `DB_HOST`, `DB_DATABASE`, `DB_USERNAME`, και `DB_PASSWORD`
4. Εκτελεί την εντολή `php artisan key:generate` για να δημιουργηθεί ένα τυχαίο `APP_KEY`
5. Εκτελεί την εντολή `php artisan migrate` ώστε να δημιουργηθούν οι απαραίτητοι πίνακες
6. Εκτελεί την εντολή `php artisan db:seed` για να γεμίσει ο πίνακας των βάσεων με βάσεις παλιότερων ετών

7. Αντιγράφει το αρχείο `Chart.php` από τον φάκελο `./moddedLibs/` στον φάκελο `./vendor/khill/lavacharts/src/Charts`
8. Ρυθμίζει τον εξυπηρετητή του έτσι ώστε να «δείχνει» στον φάκελο `./public`
9. Μεταβαίνει στη διεύθυνση όπου βρίσκεται ο εξυπηρετητής (π.χ. `http://127.0.0.1`)
10. Κάνει αυθεντικοποίηση επιλέγοντας την επιλογή **Σύνδεση**
11. Πηγαίνει στην επιλογή **Ανέβασμα** → `students.txt`, και ανεβάζει ένα-ένα τα αρχεία `students.txt` **από τα παλιότερα στα πιο καινούργια**, επιλέγοντας κάθε φορά την σωστή χρονιά και τη σωστή περίοδο
12. Μετά πηγαίνει στην επιλογή **Ανέβασμα** → `classStudents.txt`, και ανεβάζει το πιο πρόσφατο `classStudents.txt` αρχείο
13. Τέλος, πηγαίνει στην επιλογή **Ανέβασμα** → `studentCourses.txt`, και ανεβάζει το πιο πρόσφατο `studentCourses.txt` αρχείο

Λειτουργίες που προσφέρει η εφαρμογή

Αρχική σελίδα

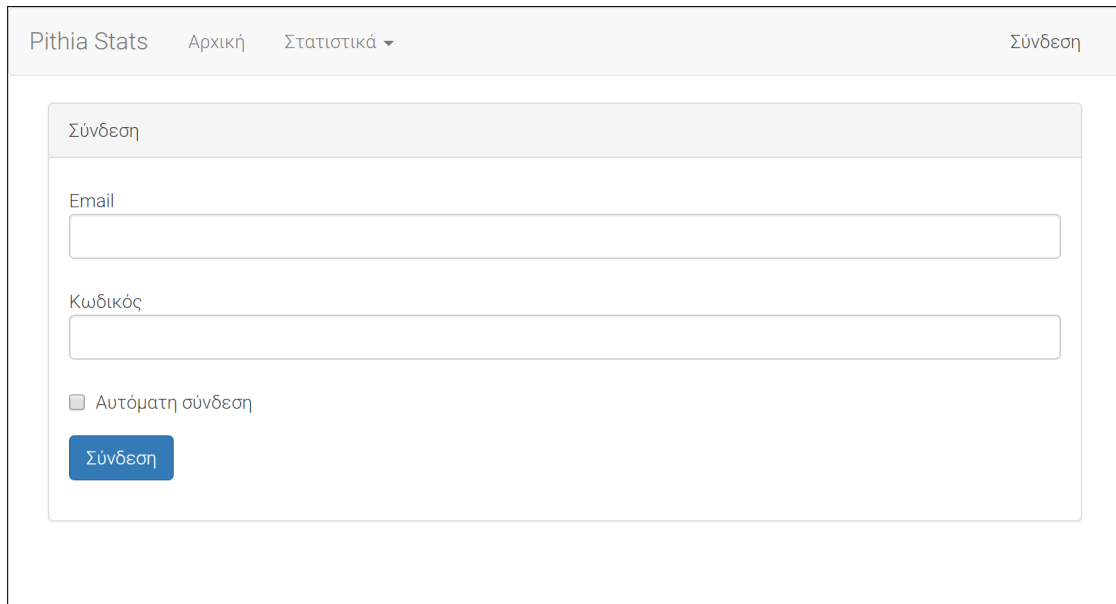
Στην αρχική σελίδα (Σχήμα 1) φαίνεται η περιγραφή της εργασίας με συνδέσμους στην κάθε λειτουργία, καθώς και ένα γράφημα που δείχνει στατιστικά για τους εισακτέους των τελευταίων 5 ετών.



Σχήμα 1: Αρχική σελίδα

Σύνδεση

Ο χρήστης πρέπει πρώτα να κάνει αυθεντικοποίηση για να αποκτήσει πρόσβαση σε κάποιες λειτουργίες που αναφέρονται παρακάτω. Αυτό γίνεται μέσω της λειτουργίας «Σύνδεση» (Σχήμα 2).



The image shows a web interface for logging in. At the top, there is a navigation bar with the text 'Pithia Stats', 'Αρχική', 'Στατιστικά', and 'Σύνδεση'. Below this, there is a central box with a title 'Σύνδεση'. Inside this box, there are two input fields: 'Email' and 'Κωδικός'. Below the input fields, there is a checkbox labeled 'Αυτόματη σύνδεση'. At the bottom of the box, there is a blue button with the text 'Σύνδεση'.

Σχήμα 2: Σύνδεση

Φοιτητές

Σε αυτή τη σελίδα (Σχήμα 3) φαίνονται όλοι οι φοιτητές του τμήματος, με δυνατότητα αναζήτησης. Πατώντας επάνω στο όνομα ενός φοιτητή, εμφανίζονται πληροφορίες για τον συγκεκριμένο φοιτητή.

Πτυχιακή εργασία του φοιτητή Κοσματόπουλου Μιχαήλ

Pithia Stats Αρχική Φοιτητές Στατιστικά ▾ Ανέβασμα ▾

Φοιτητές

Στον παρακάτω πίνακα φαίνονται όλοι οι φοιτητές του τμήματος. Πατήστε σε έναν φοιτητή για περισσότερες πληροφορίες.
Show 10 entries Search:

AM	Όνομα	Επώνυμο	Πατρώνυμο	Έτος εισαγωγής	Περίοδος εισαγωγής	Εξάμηνο εισαγωγής
001423	ΠΑΝΑΓΙΩΤΗΣ	ΠΑΝΤΖΑΡΤΖΙΔΗΣ	ΒΑΣΙΛΕΙΟΥ	2000	Χειμερινό	5
001424	ΕΥΑΓΓΕΛΟΣ	ΝΑΝΝΟΣ	ΙΩΑΝΝΗ	2000	Χειμερινό	1
001425	ΔΗΜΗΤΡΙΟΣ	ΦΛΩΡΟΣ	ΝΙΚΟΛΑΟΥ	2000	Χειμερινό	1
001426	ΠΕΤΡΟΣ	ΚΑΡΑΚΑΝΑΣ	ΝΙΚΟΛΑΟΥ	2000	Χειμερινό	1
001427	ΓΕΩΡΓΙΟΣ	ΓΚΑΓΚΑΝΗΣ	ΙΩΑΝΝΗ	2000	Χειμερινό	1
001428	ΓΕΩΡΓΙΟΣ	ΠΑΠΑΔΟΠΟΥΛΟΣ	ΚΩΝ/ΝΟΥ	2000	Χειμερινό	1
001429	ΙΦΙΓΕΝΕΙΑ	ΠΑΠΑΓΕΩΡΓΙΟΥ	ΝΙΚΟΛΑΟΥ	2000	Χειμερινό	1
001430	ΣΠΥΡΙΔΩΝ	ΜΕΜΕΖΑΣ	ΔΗΜΗΤΡΙΟΥ	2000	Εαρινό	1
001431	ΘΩΜΑΣ	ΠΑΠΑΔΗΜΗΤΡΙΟΥ	ΑΓΓΕΛΟΥ	2000	Χειμερινό	1
001432	ΑΘΑΝΑΣΙΟΣ	ΜΠΑΝΤΣΟΣ	ΧΡΗΣΤΟΥ	2000	Χειμερινό	1

Showing 1 to 10 of 4,578 entries Previous 1 2 3 4 5 ... 458 Next

Σχήμα 3: Φοιτητές

Πληροφορίες φοιτητή

Σε αυτή τη σελίδα (Σχήμα 4) φαίνονται πληροφορίες ενός φοιτητή, όπως το ιστορικό κατάστασής του, οι βαθμολογίες του και μέσο όρο βαθμολογίας ανά εξάμηνο.

Πτυχιακή εργασία του φοιτητή Κοσματόπουλου Μιχαήλ

Pithia Stats
Αρχική
Φοιτητές
Στατιστικά ▾
Ανέβασμα ▾

Στοιχεία φοιτητή

AM 103592

Όνομα ΜΙΧΑΗΛ

Επώνυμο ΚΟΣΜΑΤΟΠΟΥΛΟΣ

Πατρώνυμο ΚΩΝ/ΝΟΣ

Έτος εισαγωγής 2010

Περίοδος εισαγωγής Χειμερινό

Εξάμηνο εισαγωγής 1

Ιστορικό κατάστασης φοιτητή

Έτος	Περίοδος εξαμ.	Κατάσταση	Εξάμηνο
2014	Εαρινό	Ενεργοί	10
2014	Χειμερινό	Ενεργοί	9
2013	Εαρινό	Ενεργοί	8
2013	Χειμερινό	Ενεργοί	7

Βαθμολογίες φοιτητή

Στον παρακάτω πίνακα φαίνονται όλοι οι βαθμοί του συγκεκριμένου φοιτητή.

Show entries Search:

Μάθημα	Βαθμός	Έτος	Περίοδος εξεταστικής
ΑΓΓΛΙΚΗ ΟΡΟΛΟΓΙΑ	8.5	2010	A ΕΑΡ
Αλληλεπίδραση Αθρώπου-Μηχανής & Ανάπτυξη Διεπιφανειών Χρήστη - E	10	2011	A ΧΕΙΜ
Αλληλεπίδραση Αθρώπου-Μηχανής & Ανάπτυξη Διεπιφανειών Χρήστη - Θ	9	2011	B ΧΕΙΜ

Showing 1 to 3 of 64 entries

Previous 1 2 3 4 5 ... 22 Next

Μέσος όρος βαθμολογίας

Στο παρακάτω διάγραμμα φαίνεται ο μέσος όρος της βαθμολογίας του φοιτητή ανά περίοδο εξεταστικής.

Τελικές βαθμολογίες φοιτητή

Στον παρακάτω πίνακα φαίνονται οι τελικοί βαθμοί του συγκεκριμένου φοιτητή.

Show entries Search:

Μάθημα	Βαθμός	Έτος	Περίοδος εξεταστικής	Σημειώσεις
ΑΓΓΛΙΚΗ ΟΡΟΛΟΓΙΑ	8.5	2010	A ΕΑΡ	
Αλληλεπίδραση Αθρώπου-Μηχανής & Ανάπτυξη Διεπιφανειών Χρήστη - E	10	2011	A ΧΕΙΜ	
Αλληλεπίδραση Αθρώπου-Μηχανής & Ανάπτυξη Διεπιφανειών Χρήστη - Θ	9	2011	B ΧΕΙΜ	
Ανάπτυξη Διαδικτυακών Συστημάτων & Εφαρμογών - E	9	2012	A ΧΕΙΜ	
Ανάπτυξη Διαδικτυακών Συστημάτων & Εφαρμογών - Θ	7	2012	A ΧΕΙΜ	
Ανάπτυξη και Διαχείριση Ολοκληρωμένων Πλ. Συστημάτων & Εφαρμογών - E	9.5	2013	B ΧΕΙΜ	
Ανάπτυξη και Διαχείριση Ολοκληρωμένων Πλ. Συστημάτων & Εφαρμογών - Θ	6.3	2013	A ΧΕΙΜ	
Αριθ. Ανάλυση και Προγ/μος Επιστημονικών Εφαρμογών - E	10	2011	A ΧΕΙΜ	
Αριθ. Ανάλυση και Προγ/μος Επιστημονικών Εφαρμογών - Θ	6.6	2011	B ΧΕΙΜ	
Ασφάλεια Πληροφορικών Συστημάτων - E	9.1	2012	A ΕΑΡ	

Showing 1 to 10 of 64 entries

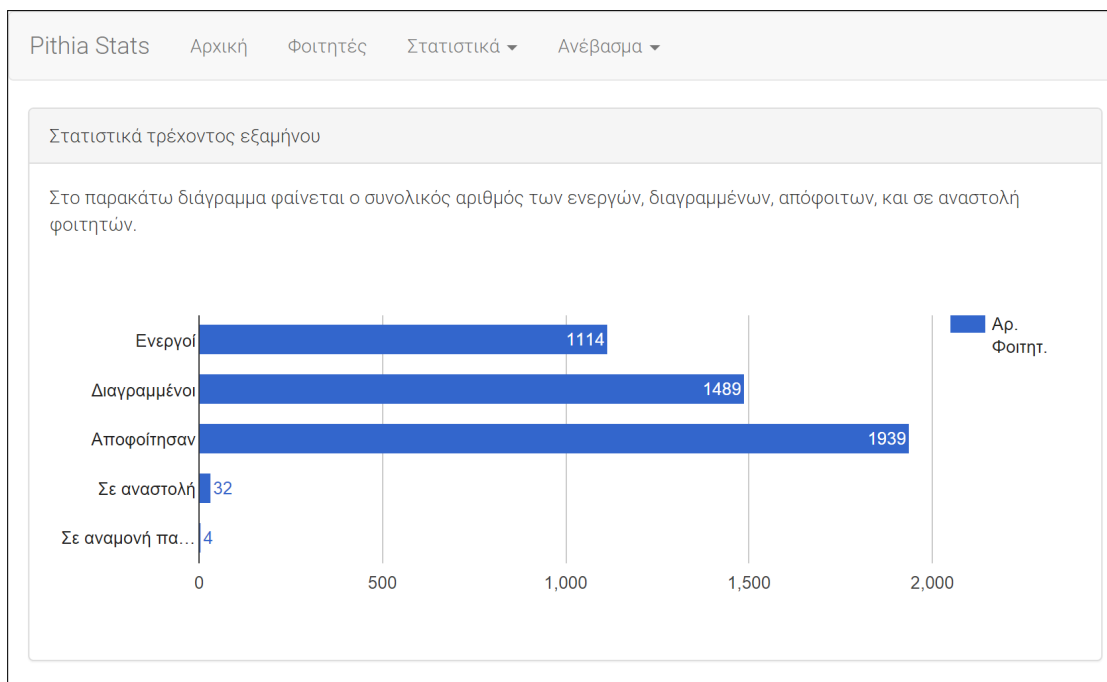
Previous 1 2 3 4 5 6 7 Next

Σχήμα 4: Πληροφορίες φοιτητή

Στατιστικά

Τρέχον εξάμηνο

Σε αυτή τη σελίδα (Σχήμα 5) φαίνεται ο συνολικός αριθμός των ενεργών, διαγραφμένων, πτυχιούχων, κ.α. φοιτητών.

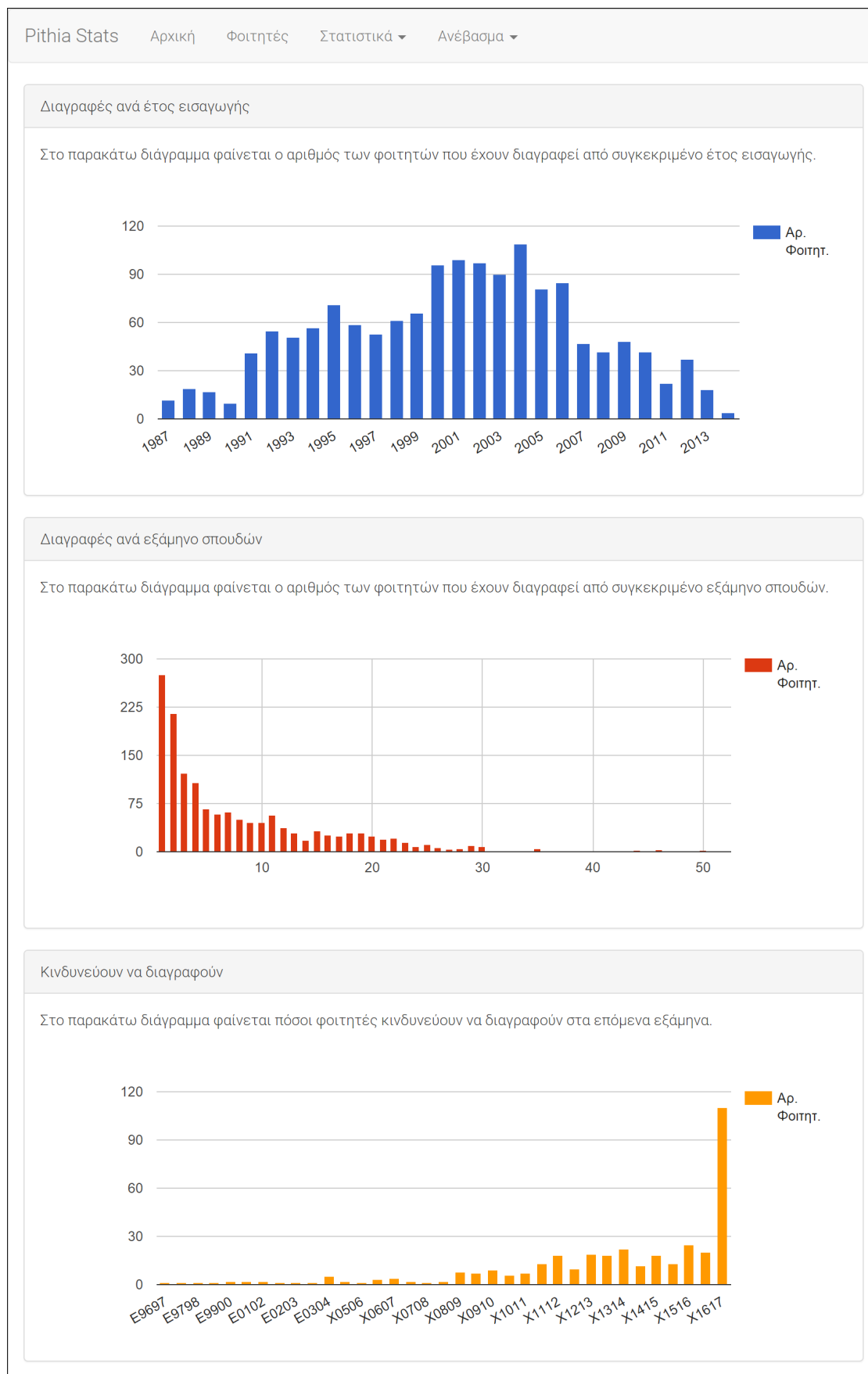


Σχήμα 5: Στατιστικά τρέχοντος εξαμήνου

Διαγραφές

Σε αυτή τη σελίδα (Σχήμα 6) φαίνεται ο αριθμός των φοιτητών που έχουν διαγραφεί από κάθε έτος εισαγωγής, ο αριθμός των φοιτητών που έχουν διαγραφεί από συγκεκριμένο εξάμηνο σπουδών, και ο αριθμός των φοιτητών που κινδυνεύουν να διαγραφούν.

Πτυχιακή εργασία του φοιτητή Κοσματόπουλου Μιχαήλ



Σχήμα 6: Στατιστικά διαγραφών

Αναστολές

Σε αυτή τη σελίδα (Σχήμα 7), φαίνονται οι φοιτητές που βρίσκονται σε αναστολή, καθώς και ο αριθμός των εξαμήνων που βρίσκονται σε αναστολή. Πατώντας επάνω στο όνομα ενός φοιτητή γίνεται μετάβαση στις πληροφορίες του.

Pithia Stats Αρχική Φοιτητές Στατιστικά ▾ Ανέβασμα ▾

Φοιτητές σε αναστολή σπουδών

Στον παρακάτω πίνακα φαίνονται οι φοιτητές που βρίσκονται σε αναστολή, καθώς και ο συνολικός αριθμός των εξαμήνων που βρίσκονται σε αναστολή.

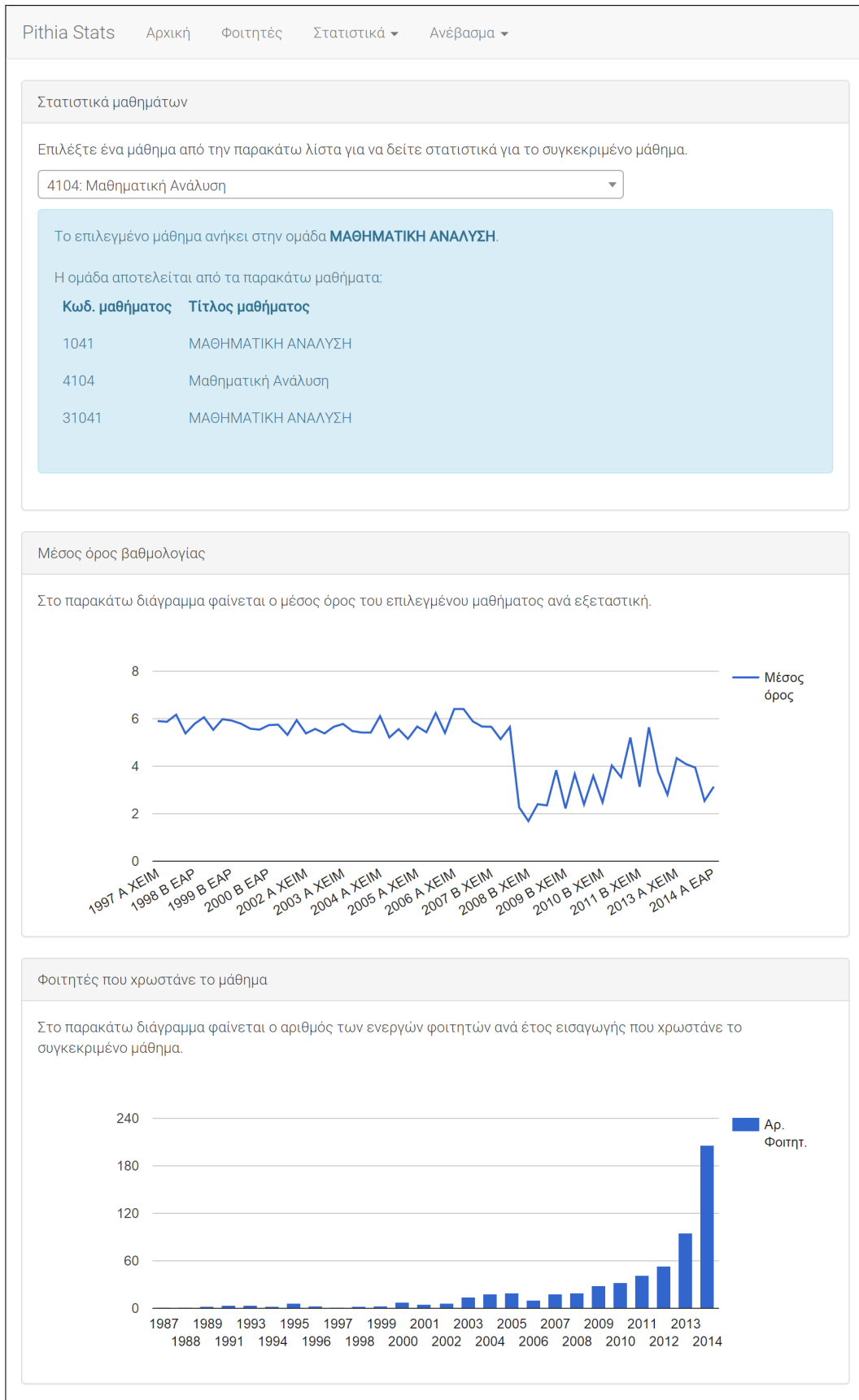
ΑΜ	Όνομα	Επώνυμο	Έτος εισαγωγής	Εξάμηνα σε αναστολή
022076	ΠΕΤΡΟΣ	ΙΚΚΟΥ	2002	8
103653	ΓΑΡΥΦΑΛΛΟΣ	ΚΑΛΥΒΑΣ	2010	8
103666	ΓΕΩΡΓΙΟΣ	ΤΟΥΡΤΟΥΡΑΣ	2010	8
113711	ΣΟΛΩΝ	ΒΟΥΓΑΣ	2011	8
032423	ΦΩΤΕΙΝΗ	ΜΙΧΑΗΛΙΔΟΥ	2003	7
052887	ΒΑΣΙΛΕΙΟΣ	ΧΑΡΑΛΑΜΠΙΔΗΣ	2005	7
073299	ΠΑΣΧΑΛΗΣ	ΜΑΡΚΟΥΔΗΣ	2007	7
052790	ΠΕΤΡΟΣ	ΜΗΤΣΟΠΟΥΛΟΣ	2005	6
103658	ΑΝΤΩΝΙΟΣ	ΜΟΣΧΙΔΗΣ	2010	6
123918	ΑΓΓΕΛΟΣ	ΜΠΕΡΔΟΣ	2012	6
083306	ΕΜΜΑΝΟΥΗΛ	ΦΡΕΓΚΟΣ	2008	5
103570	ΓΕΩΡΓΙΟΣ	ΝΑΛΤΣΑΣ	2010	5
123968	ΙΩΑΝΝΗΣ	ΓΑΝΙΤΗΣ	2012	4

Σχήμα 7: Φοιτητές σε αναστολή σπουδών

Μαθήματα

Σε αυτή τη σελίδα (Σχήμα 8) δίνεται στο χρήστη να επιλέξει ένα μάθημα. Αφού επιλέξει ένα μάθημα, εμφανίζονται οι μέσοι όροι βαθμολογιών για αυτό το μάθημα, όπως επίσης και ο αριθμός των φοιτητών που χρωστάνε αυτό το μάθημα.

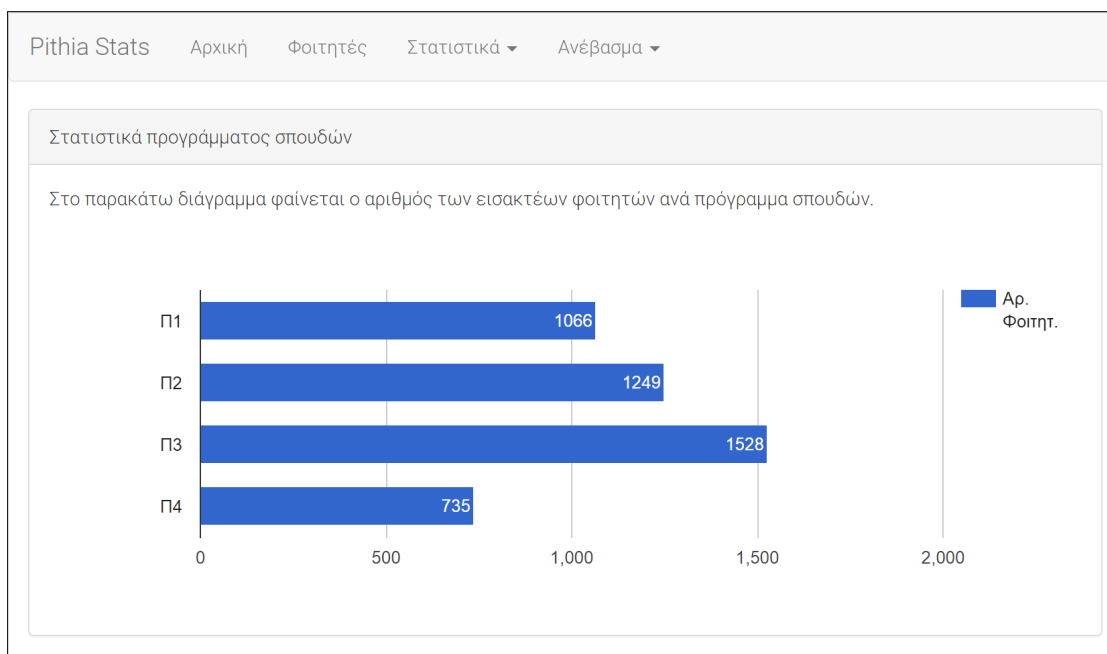
Πτυχιακή εργασία του φοιτητή Κοσματόπουλου Μιχαήλ



Σχήμα 8: Στατιστικά μαθημάτων

Πρόγραμμα σπουδών

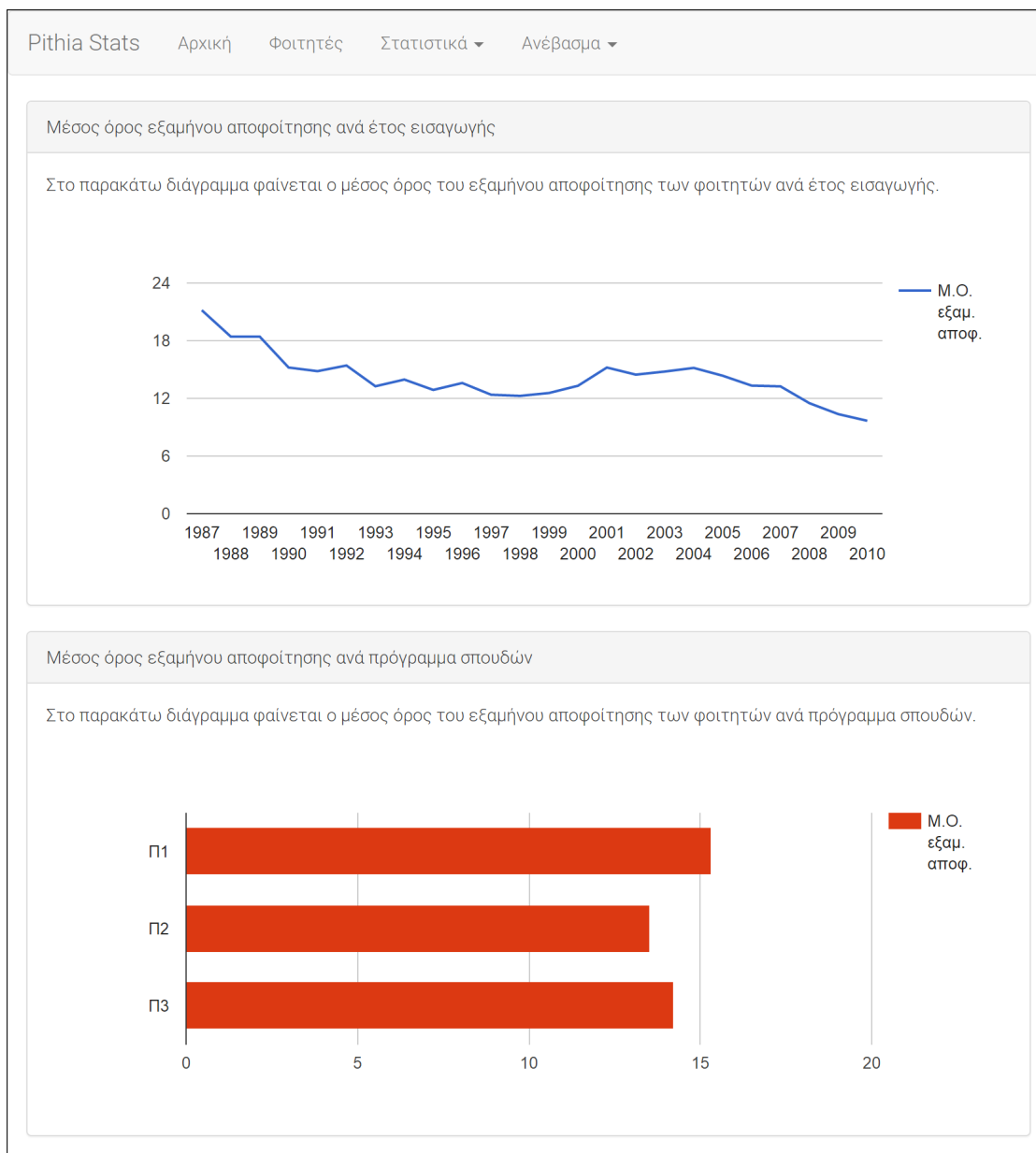
Σε αυτή τη σελίδα (Σχήμα 9) φαίνεται ο αριθμός των εισακτέων φοιτητών ανά πρόγραμμα σπουδών.



Σχήμα 9: Στατιστικά προγράμματος σπουδών

Απόφοιτοι

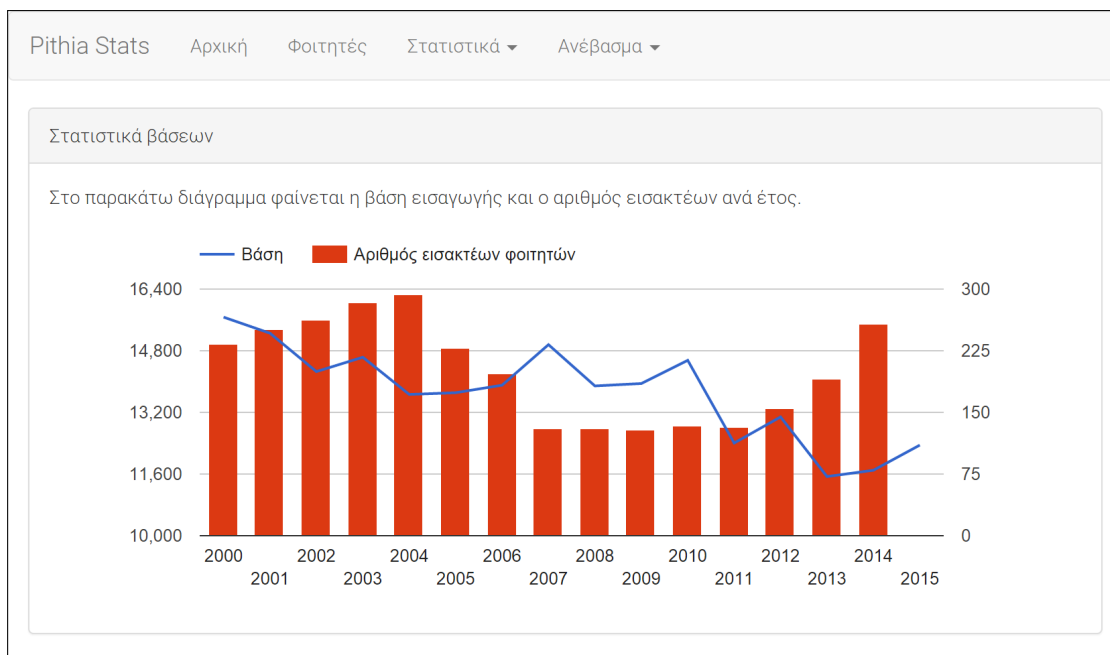
Σε αυτή τη σελίδα (Σχήμα 10) φαίνεται ο μέσος όρος του εξαμήνου αποφοίτησης των φοιτητών ανά έτος εισαγωγής και ανά πρόγραμμα σπουδών.



Σχήμα 10: Στατιστικά αποφοίτων

Βάσεις

Σε αυτή τη σελίδα (Σχήμα 11) φαίνεται σε διάγραμμα η βάση εισαγωγής και ο αριθμός εισακτέων ανά έτος.



Σχήμα 11: Στατιστικά βάσεων

Ανέβασμα

students.txt

Σε αυτή τη σελίδα (Σχήμα 12) δίνεται η δυνατότητα ενημέρωσης των φοιτητών χρησιμοποιώντας το αρχείο students.txt (βλ. 1.1.1).

Ανέβασμα αρχείου φοιτητών

Χρησιμοποιείστε αυτή τη φόρμα για να ενημερώσετε την κατάσταση των φοιτητών.

Αρχείο students.txt

No file chosen

Χρονιά απόκτησης αρχείου

2015

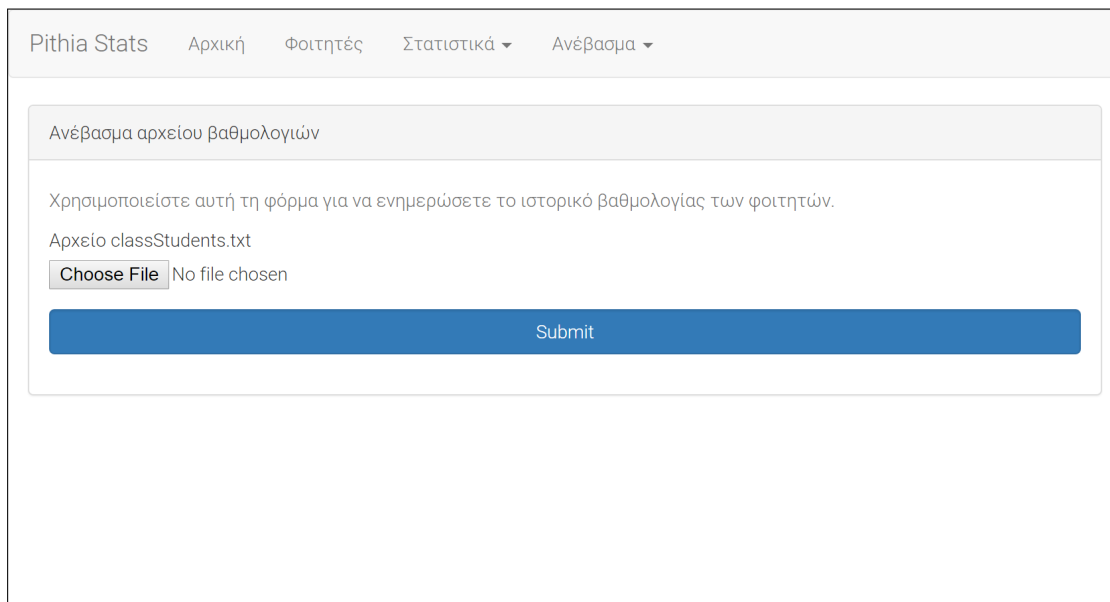
Εξάμηνο απόκτησης αρχείου

Χειμερινό εξάμηνο

Σχήμα 12: Ενημέρωση φοιτητών

classStudents.txt

Σε αυτή τη σελίδα (Σχήμα 13) δίνεται η δυνατότητα ενημέρωσης των βαθμολογιών χρησιμοποιώντας το αρχείο classStudents.txt (βλ. 1.1.2).

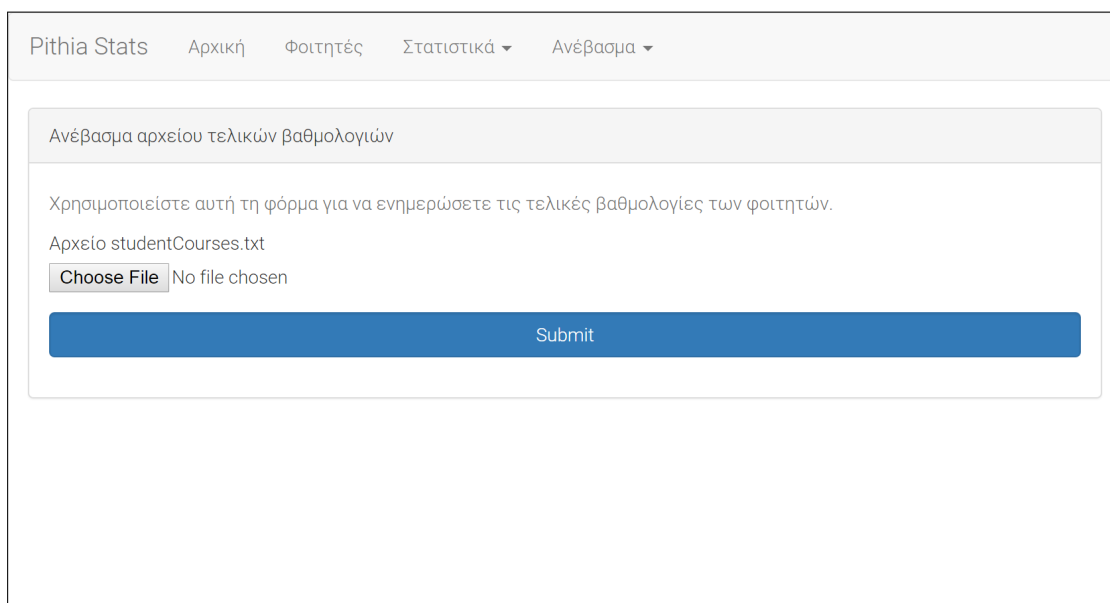


The screenshot shows a web interface for uploading a file. At the top, there is a navigation bar with the text 'Pithia Stats' and several menu items: 'Αρχική', 'Φοιτητές', 'Στατιστικά', and 'Ανέβασμα'. Below the navigation bar, the main content area is titled 'Ανέβασμα αρχείου βαθμολογιών'. It contains the following text: 'Χρησιμοποιείστε αυτή τη φόρμα για να ενημερώσετε το ιστορικό βαθμολογίας των φοιτητών.' followed by 'Αρχείο classStudents.txt'. There is a file selection button labeled 'Choose File' with the text 'No file chosen' next to it. At the bottom of the form is a large blue 'Submit' button.

Σχήμα 13: Ενημέρωση βαθμολογιών

studentCourses.txt

Σε αυτή τη σελίδα (Σχήμα 14) δίνεται η δυνατότητα ενημέρωσης των τελικών βαθμολογιών χρησιμοποιώντας το αρχείο studentCourses.txt (βλ. 1.1.3).



The screenshot shows a web interface for uploading a file. At the top, there is a navigation bar with the text 'Pithia Stats' and several menu items: 'Αρχική', 'Φοιτητές', 'Στατιστικά', and 'Ανέβασμα'. Below the navigation bar, the main content area is titled 'Ανέβασμα αρχείου τελικών βαθμολογιών'. It contains the following text: 'Χρησιμοποιείστε αυτή τη φόρμα για να ενημερώσετε τις τελικές βαθμολογίες των φοιτητών.' followed by 'Αρχείο studentCourses.txt'. There is a file selection button labeled 'Choose File' with the text 'No file chosen' next to it. At the bottom of the form is a large blue 'Submit' button.

Σχήμα 14: Ενημέρωση τελικών βαθμολογιών

Επίλογος

Σε αυτό το κεφάλαιο δόθηκαν οδηγίες για την εγκατάσταση της εφαρμογής, καθώς επίσης και μια περιήγηση των δυνατοτήτων που προσφέρει.