ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΣΧΟΛΗ ΨΗΦΙΑΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

ΠΜΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

ΚΑΤΕΥΘΥΝΣΗ: ΥΠΟΛΟΓΙΣΤΙΚΕΣ & ΔΙΑΔΙΚΤΥΑΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ & ΕΦΑΡΜΟΓΕΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

«ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΠΤΥΧΙΑΚΩΝ ΚΑΙ ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ»

ΚΕΧΑΓΙΑΣ ΗΛΙΑΣ

ΕΠΙΒΛΕΠΩΝ: Δρ. **ΜΙΧΑΛΑΚΕΛΗΣ ΧΡΗΣΤΟΣ**, *ΛΕΚΤΟΡΑΣ*

ΑΘΗΝΑ 2016

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Αθήνα, Ταύρος / 02 / 2016

Επιτροπή Αξιολόγησης

1. Δρ. **ΜΙΧΑΛΑΚΕΛΗΣ ΧΡΗΣΤΟΣ**, *ΛΕΚΤΟΡΑΣ*

2. Δρ. **ΝΙΚΟΛΑΪΔΟΥ ΜΑΡΑ**, *ΚΑΘΗΓΗΤΡΙΑ*

3. Δρ. **ΑΝΑΓΝΩΣΤΟΠΟΥΛΟΣ ΔΗΜΟΣΘΕΝΗΣ**, *ΚΑΘΗΓΗΤΗΣ*

Ευχαριστίες

Ευχαριστώ όλους τους καθηγητές μου για την πολύτιμη συμβολή τους όλα αυτά τα χρόνια στην απόκτηση νέων γνώσεων. Ιδιαίτερα, θα ήθελα να ευχαριστήσω όλο το εκπαιδευτικό προσωπικό του τμήματος Μηχανικών Πληροφορικής του Α.Τ.Ε.Ι. Θεσσαλίας για τη διαμόρφωση μου ως Τεχνολόγο Μηχανικό. Ομοίως, θα ήθελα να ευχαριστήσω τον κύριο Μιχαλακέλη Χρήστο, Λέκτορα στο γνωστικό αντικείμενο Ευρυζωνικά Τηλεπικοινωνιακά Δίκτυα και Τεχνοοικονομική Ανάλυσή τους, για την άψογη συνεργασία στην υλοποίηση αυτής της εργασίας, για τις συμβουλές, την εμπιστοσύνη και για τις προτάσεις προς το πρόσωπό μου. Επιπλέον, ευχαριστώ τον κύριο Πιπερίδη Κώστα από τη Διαχείριση δικτύου, ανάπτυξη υπηρεσιών & υποστήριξη χρηστών του Χαροκοπείου Πανεπιστημίου για την παροχή του Server ώστε να τρέξει η εφαρμογή. Τέλος, ευχαριστώ τους φίλους μου και την οικογένειά μου για την αμέριστη συμπαράσταση.

Κεχαγιάς Ηλίας

Μηχανικός Πληροφορικής & Τηλεπικοινωνιών (Τ.Ε.)

Πίνακας Περιεχομένων

Πίνακας Περιεχομένων i
Περίληψη v
Λέξεις Κλειδιά v
Abstract vi
Key Words vi
ΚΕΦΑΛΑΙΟ 1 - ΕΙΣΑΓΩΓΗ 1
1.1 Αντικείμενο της Διπλωματικής Εργασίας 1
1.2 Οργάνωση της Διπλωματικής Εργασίας 2
ΚΕΦΑΛΑΙΟ 2 – ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ 3
2.1 Εισαγωγή 3
2.2 Περιγραφή της τεχνολογίας MS SQL – Server 2014 3
2.3 Ανάλυση του σχεδιασμού των πινάκων της Βάσης Δεδομένων 4
2.3.1 Πίνακας UserProfile 5
2.3.2 Πίνακας webpages_Membership 6
2.3.3 Πίνακας webpages_Roles 6
2.3.4 Πίνακας webpages_UsersInRoles 7
2.3.5 Πίνακας webpages_OAuthMembership 7
2.3.6 Πίνακας AkEtosSet 7
2.3.7 Πίνακας ComitteeSet 8
2.3.8 Πίνακας ProfessorSet 8
2.3.9 Πίνακας ProjectSet 9
2.3.10 Πίνακας RequestSet 9
2.3.11 Πίνακας SecretarySet 10
2.3.12 Πίνακας StudentSet 10
2.4 Διάγραμμα Οντοτήτων – Συσχετίσεων (ER - DIAGRAM) 11
2.5 Αναφορές & Βιβλιογραφία 12
ΚΕΦΑΛΑΙΟ 3 – ΠΛΑΤΦΟΡΜΕΣ & ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ 13
3.1 .NET Framework 13

3.2 Microsoft Visual Studio 14
3.3 Αναφορές & Βιβλιογραφία 15
ΚΕΦΑΛΑΙΟ 4 – ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ 16
4.1 Αρχική σελίδα (Index.aspx) 16
4.2 Πτυχιακή Εργασία (Ptyxiaki.aspx) 16
4.3 Διπλωματική Εργασία (Diplomatiki.aspx) 17
4.4 About - Δημιουργία (About.aspx) 18
micro – site 19
4.5 Log in – Σύνδεση (Login.aspx) 19
4.5.1 Log in – Σύνδεση (Login.aspx) με κάποιο από τα πεδία κενό 20
4.5.2 Log in – Σύνδεση (Login.aspx) με λάθος username / password 20
4.6 Manage – Διαχείριση Λογαριασμού (Manage.aspx) 21
4.7 Αρχική Σελίδα – Συνδεδεμένος/η Φοιτητής/ρια (Index.aspx) 22
4.8 Αρχική Φοιτητές (Student.aspx) 22
4.9 Επικοινωνία (Contact.aspx) 23
4.9.1 Επικοινωνία με επιτυχής αποστολή e-mail (Contact.aspx) 24
4.10 Βιογραφικό Σημείωμα CV (Cv.aspx) 25
4.10.1 Βιογραφικό Σημείωμα CV με επιτυχής ενημέρωση (Cv.aspx) 25
4.11 Πτυχιακές Εργασίες (Proptyxiakes.aspx) 25
4.12 Διπλωματικές Εργασίες (Metaptyxiakes.aspx) 26
4.13 Λεπτομέρειες (DetailsProject.aspx) 27
Προεπισκόπηση εκτύπωσης σελίδας 29
4.14 Υποβολή Αίτησης (NewRequest.aspx) 30
4.15 Αρχική Σελίδα – Συνδεδεμένος/η Διδάσκων (Index.aspx) 30
4.16 Αρχική σελίδα Διδάσκων (Professor.aspx) 31
Λεπτομέρειες Εργασίας 33
Επεξεργασία Εργασίας 34
4.17 Επικοινωνία (Contact.aspx) 34
4.18 Βαθμολογία Εργασίας (Grades.aspx) 35
4.18.1 Βαθμολογία Εργασίας με κάποια παράληψη από τον/την καθηγητή/ρια 35
4.18.2 Βαθμολογία Εργασίας χωρίς δικαιοδοσία. 36

4.19 Αιτήσεις (Requests.aspx) 36
Πληροφορίες Αίτησης 37
Αποδοχή Αίτησης 38
4.20 Δημιουργία Επιτροπής (Comittee/Create.aspx) 38
4.21 Νέα Εργασία (Project/Create.aspx) 39
4.22 Αρχική Σελίδα – Συνδεδεμένη Γραμματεία (Index.aspx) 40
4.23 Γραμματεία - Διαχείριση (Secretary/Manage.aspx) 40
4.24 Εκτύπωση για Γενική Συνέλευση (Print.aspx) 41
Φίλτρα Αναζήτησης 42
Αποτελέσματα Αναζήτησης 43
4.24.1 Εκτύπωση για Γενική Συνέλευση (Print.aspx) 43
4.24.2 Αποτέλεσμα Εκτύπωσης για Γενική Συνέλευση 44
4.25 Ακ. Έτη (AkEtos/Index.aspx) 45
4.26 Δημιουργία Ακ. Έτους (AkEtos/Create.aspx) 46
4.27 Επεξεργασία Ακ. Έτους (AkEtos/Edit.aspx) 47
4.28 Διαγραφή Ακ. Έτους 47
4.29 Διαχείριση Επιτροπών (Comittee/Index.aspx) 47
Δημιουργία / Επεξεργασία Επιτροπής 49
4.30 Επεξεργασία Επιτροπής (Comittee/Edit.aspx) 49
4.31 Λεπτομέρειες Επιτροπής (Comittee/Delete.aspx) 50
4.32 Διαγραφή Επιτροπής (Comittee/Delete.aspx) 50
4.33 Διαχείριση Αιτήσεων (Request/Index.aspx) 51
4.34 Υποβολή Αίτησης (Request/Create.aspx) 52
4.35 Επεξεργασία Αίτησης (Request/Edit.aspx) 53
4.36 Λεπτομέρειες Αίτησης (Request/Details.aspx) 54
4.37 Διαγραφή Αίτησης (Request/Delete.aspx) 54
4.38 Διαχείριση Καθηγητών (Professor/Index.aspx) 55
4.39 Καταχώρηση Καθηγητή (Professor/Create.aspx) 56
4.40 Επεξεργασία Καθηγητή/ριας (Professor/Edit.aspx) 57
4.41 Διαγραφή Καθηγητή/ριας (Professor/Delete.aspx) 58
4.42 Λεπτομέρειες Καθηγητή/ριας (Professor/Details.aspx) 58

4.43 Διαχείριση Εργασιών (Project/Index.aspx) 59
4.44 Νέα Εργασία (Project/Create.aspx) 60
4.45 Επεξεργασία Εργασίας (Project/Edit.aspx) 61
4.46 Διαγραφή Εργασίας (Project/Delete.aspx) 62
4.47 Λεπτομέρειες Εργασίας (Project/Details.aspx) 63
4.48 Διαχείριση Φοιτητών (Student/Index.aspx) 64
4.49 Καταχώρηση Φοιτητή/ριας (Student/Create.aspx) 65
4.50 Επεξεργασία Φοιτητή/ριας (Student/Edit.aspx) 66
4.51 Διαγραφή Φοιτητή/ριας (Student/Delete.aspx) 67
4.52 Λεπτομέρειες Φοιτητή/ριας (Student/Details.aspx) 68
4.53 Default Σελίδα σφαλμάτων(Error.aspx) 69
ΚΕΦΑΛΑΙΟ 5 – ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΚΩΔΙΚΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ 70
5.1 Εισαγωγή 70
5.1.2 Η αρχιτεκτονική Model – View – Controller 71
5.1.3 Η γλώσσα Προγραμματισμού C# 71
5.2 Το Class Diagram της εφαρμογής 72
5.3 Assembly Dependencies graph 73
5.4 Αποτελέσματα μετρήσεων 74
5.4.1 CPU Sampling 75
5.4.2 .NET memory allocation 75
5.5 Βιβλιογραφία & Αναφορές 76
ΚΕΦΑΛΑΙΟ 6 – ΣΥΜΠΕΡΑΣΜΑΤΑ & ΕΠΕΚΤΑΣΕΙΣ 77
6.1 Αποτίμηση της Διπλωματικής Εργασίας 77
6.2 Μελλοντικές επεκτάσεις της εφαρμογής 77
Βιβλιογραφία 78
Sites που βοήθησαν στη παραγωγή κώδικα 81
ΠΑΡΑΡΤΗΜΑ Α A1

Περίληψη

Προγραμματισμός σημαίνει κατανόηση.
- Kristen Nygaard

Στο τέλος των σπουδών τους, τόσο οι προπτυχιακοί όσο και οι μεταπτυχιακοί φοιτητές πρέπει να εκπονήσουν την πτυχιακή ή τη διπλωματική τους εργασία, αντίστοιχα. Για το λόγο αυτό, η Γραμματεία απευθύνει πρόσκληση στους διδάσκοντες των προγραμμάτων σπουδών (ΠΠΣ / ΠΜΣ) να αποστείλουν κατάλογο των προτεινόμενων θεμάτων προκειμένου οι φοιτητές να επιλέξουν την εργασία που θα εκπονήσουν. Δεν υπάρχει όμως κάποιο επίπεδο αυτοματοποίησης, μηχανοργάνωσης και παρακολούθησης της διαδικασίας αυτής.

Η παρούσα εργασία αφορά στην ανάλυση απαιτήσεων, το σχεδιασμό και την ανάπτυξη ενός διαδικτυακού πληροφοριακού συστήματος για την ανάρτηση θεμάτων πτυχιακών και διπλωματικών εργασιών, τη διαχείριση αιτήσεων ανάληψης εργασιών εκ μέρους των φοιτητών, την παραγωγή στατιστικών και αναφορών και άλλες σχετικές εργασίες.

Το πληροφοριακό σύστημα έχει αναπτυχθεί και θα τεθεί σε κατάσταση κανονικής λειτουργίας για τις εκπαιδευτικές ανάγκες των προγραμμάτων σπουδών του Τμήματος.

Λέξεις Κλειδιά

Πληροφοριακά Συστήματα, Εκπαίδευση, Βάσεις Δεδομένων, Προγραμματιστικά Εργαλεία, Γλώσσες Προγραμματισμού, Web, Frameworks, Testing, Software Performance.

Abstract

Programming is understanding
- Kristen Nygaard

At the end of their studies, both undergraduate and postgraduate students should prepare a Diploma Thesis or their postgraduate Thesis, respectively. For this reason, the Secretariat invites teachers of study programs (Undergraduate / Postgraduate) to send the list of proposed topics to the students to choose the work that will develop. But there is some level of automation, computerization and monitoring of this process.

This work relates to requirements analysis, design and development of an online information system for posting issues and graduate dissertations, management operations taking requests from students, the production of statistics and reports and other relevant tasks.

The information system developed will be in normal mode for the educational needs of the Department's study programs.

Key Words

Information Systems, Education, Databases, Software Development Tools, Programming Languages, Web, Frameworks, Testing, Software Performance.

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Αντικείμενο της Διπλωματικής Εργασίας

Το αντικείμενο της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη μιας ολοκληρωμένης διαδικτυακής εφαρμογής που ως σκοπό έχει την παροχή κάποιου επιπέδου αυτοματοποίησης, μηχανοργάνωσης και παρακολούθησης της διαδικασίας όπου η γραμματεία του τμήματος απευθύνει πρόσκληση στους διδάσκοντες των προγραμμάτων σπουδών (ΠΠΣ / ΠΜΣ) να αποστείλουν κατάλογο των προτεινόμενων θεμάτων προκειμένου οι φοιτητές να επιλέξουν την εργασία που θα εκπονήσουν.

Το πληροφοριακό σύστημα χωρίζεται σε δύο βασικά μέρη. Το πρώτο μέρος αφορά μη εξουσιοδοτημένους χρήστες, τους επισκέπτες, όπου τους παρέχονται κάποιες βασικές πληροφορίες που αφορούν την εφαρμογή και την διαδικασία αίτησης, ανάθεσης πτυχιακής και διπλωματικής εργασίας. Το δεύτερο κομμάτι της εφαρμογής, που είναι και το ουσιώδες, περιλαμβάνει όλες τα στάδια και τις διαδικασίες που λαμβάνουν χώρα κατά τη διαδικασία αυτή. Ειδικότερα, υπάρχουν τρείς ρόλοι η γραμματεία, οι διδάσκοντες και οι φοιτητές, σε αυτούς τους ρόλους εντάσσονται διάφοροι χρήστες με σκοπό να εξυπηρετηθούν από την εφαρμογή. Κάθε χρήστης ανήκει σε κάποιο ρόλο και άρα έχει πρόσβαση σε διαφορετικές λειτουργίες ανάλογα με την ιδιότητά του, έτσι υπάρχουν τρία διαφορετικά Dashboards. Συγκεκριμένα, μερικές από τις διαδικασίες που παρέχονται είναι η πρόταση ενδεικτικών θεμάτων πτυχιακών και διπλωματικών εργασιών από τους διδάσκοντες για το τρέχον Ακαδημαϊκό Έτος, η επιλογή θεμάτων προς αίτηση από τους φοιτητές, η αξιολόγηση και αποδοχή των αιτήσεων από τους καθηγητές, η επικοινωνία μεταξύ διδασκόντων, φοιτητών και γραμματείας. Επιπροσθέτως, η διαχείριση των φοιτητών, των διδασκόντων, των εργασιών, των αιτήσεων και των εξεταστικών επιτροπών από τη γραμματεία του τμήματος παίζοντας τον ρόλο του διαχειριστή του συστήματος. Παράλληλα, υπάρχει διευκόλυνση ως προς την εκτύπωση των εργασιών για έγκριση από τη Γενική Συνέλευση του τμήματος. Τέλος, η διαδικτυακή εφαρμογή περιλαμβάνει βοήθεια, διαφορετική για κάθε ρόλο. Αναλυτικότερες πληροφορίες σχετικά με την εφαρμογή υπάρχουν στο Κεφάλαιο 4 – Εγχειρίδιο χρήσης.

Στα πλαίσια της δημιουργίας της εφαρμογής, σχεδιάστηκε και υλοποιήθηκε με την πλατφόρμα Microsoft Visual Studio 2013 Ultimate. Η βάση δεδομένων σχεδιάστηκε και υλοποιήθηκε με τον Microsoft SQL Server 2014 ο οποίος παρέχεται μέσα στη προγραμματιστική πλατφόρμα. Πιο συγκεκριμένα, χρησιμοποιήθηκε η γλώσσα προγραμματισμού visual C# και από τα προεπιλεγμένα projects το ASP.NET MVC 4 Web Application. Επιπλέον, χρησιμοποιήθηκε το ADO.NET Entity Data Model από το Entity Framework 6.1.1 για τη για την πρόσβαση στα δεδομένα. Άλλες τεχνολογίες που χρησιμοποιήθηκαν ως προς το client side της εφαρμογής είναι η markup γλώσσα HTML 5, η style sheet γλώσσα CSS 3, η δυναμική γλώσσα προγραμματισμού JavaScript με τη βιβλιοθήκη JQUERY.

1.2 Οργάνωση της Διπλωματικής Εργασίας

Αναλύεται τόσο η μεθοδολογία σχεδιασμού της εφαρμογής όσο και η υλοποίησή της. Αποτελείται από συνολικά έξι κεφάλαια.

Στο *1ο κεφάλαιο* γίνεται μια εισαγωγή όπου αναφέρονται το αντικείμενο της διπλωματικής εργασίας και η οργάνωση της διπλωματικής εργασίας και συγκεκριμένα είναι το τρέχον κεφάλαιο.

Στο *2ο κεφάλαιο* γίνεται μια μικρή παρουσίαση της τεχνολογίας που χρησιμοποιήθηκε για τη βάση δεδομένων συνοδευόμενη από την ανάλυσή της. Παρουσιάζονται αναλυτικά και μέσα από PrtScn οι πίνακες με τα αντίστοιχα πεδία τους καθώς επίσης και το UML διάγραμμα Οντοτήτων Συσχετίσεων.

Στο *3ο κεφάλαιο* παρουσιάζονται συνοπτικά τα προγραμματιστικά εργαλεία και οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν και αναφέρθηκαν παραπάνω και μια σύντομη ανάλυση του ακρογωνιαίου λίθου τους που δεν είναι άλλο από το .NET Framework.

Στο *4ο κεφάλαιο* δίνεται ένα κατά το δυνατόν κατανοητό εγχειρίδιο χρήσης που τμηματικά χρησιμοποιείται και στην εφαρμογή ως *Βοήθεια*. Εδώ προβάλλονται screenshots της εφαρμογής συνοδευόμενα από σύντομη περιγραφή της λειτουργικότητας που προσφέρουν αλλά και από διάφορες επισημάνσεις προς αποφυγή σφαλμάτων και επεξήγηση αυτών.

Στο *5ο κεφάλαιο* παρατίθεται μια επεξήγηση μέρους του κώδικα της εφαρμογής για το πώς υλοποιούνται κάποιες βασικές λειτουργίες, χωρίς τα κρίσιμα τμήματα διότι η εφαρμογή θα παραμείνει ενεργή στο τμήμα Πληροφορικής και Τηλεματικής του Πανεπιστημίου. Στο κεφάλαιο αυτό, τέλος, παρουσιάζεται το UML διάγραμμα κλάσεων, ο γράφος με τις Assembly Dependencies της εφαρμογής και αποτελέσματα μετρικών κατά την εκτέλεση της εφαρμογής σε πραγματικό χρόνο, όπως CPU sampling και .NET memory allocation που παρέχονται μέσα από το MS visual studio.

Στο *6ο κεφάλαιο* γίνεται μια ανασκόπηση της διπλωματικής εργασίας και παρουσιάζονται τα συμπεράσματα και τυχόν μελλοντικές επεκτάσεις που θα μπορούσαν γίνουν με σκοπό τη βελτίωση της παρούσας εφαρμογής.

Αναφορές σε βιβλία, σε διαδικτυακές σελίδες και σε επιστημονικά άρθρα που κατέστησαν δυνατή τη πραγματοποίηση της εργασίας αυτής τόσο όσον αφορά στη σύνταξη του βιβλίου όσο και στην υλοποίηση της εφαρμογής υπάρχουν στο τέλος κάθε κεφαλαίου.

ΚΕΦΑΛΑΙΟ 2

ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

2.1 Εισαγωγή

Ο Microsoft SQL Server είναι ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων που αναπτύχθηκε από τη Microsoft. Ως βάση δεδομένων, είναι ένα προϊόν λογισμικού του οποίου η κύρια λειτουργία είναι να αποθηκεύει και να ανακτά τα δεδομένα που απαιτούνται από άλλες εφαρμογές λογισμικού, είτε πρόκειται για εκείνες που αφορούν τον ίδιο υπολογιστή ή εκείνες που εκτελούνται σε έναν άλλο υπολογιστή σε ένα δίκτυο (συμπεριλαμβανομένου του Διαδικτύου). Υπάρχουν τουλάχιστον μια ντουζίνα διαφορετικές εκδόσεις του Microsoft SQL Server (SQL Server 2005, SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, SQL Server 2014) που στοχεύουν σε διαφορετικά ακροατήρια (Azure, Compact (SQL CE), Developer, Embedded (SSEE), Evaluation, Fast Track, LocalDB, Parallel Data Warehouse (PDW), Datawarehouse Appliance Edition) και για εργασίες που κυμαίνονται από μικρές εφαρμογές μίας μηχανής έως μεγάλες διαδικτυακές εφαρμογές που αντιμετωπίζουν πολλούς ταυτόχρονους χρήστες. Πρωτογενείς γλώσσες επερωτήσεων είναι η T-SQL και ANSI SQL. (http://en.wikipedia.org/wiki/Microsoft_SQL_Server)

Η αρχιτεκτονική βασίζεται στο στρώμα πρωτοκόλλου που υλοποιεί την εξωτερική διεπαφή με τον SQL Server. Όλες οι λειτουργίες που μπορούν να αντιταχθούν στον SQL Server ανακοινώνονται μέσω της Microsoft σε καθορισμένη μορφή, που ονομάζεται Πινακοποιημένη ροή δεδομένων (TDS). Το TDS είναι ένα πρωτόκολλο επιπέδου εφαρμογής, που χρησιμοποιείται για τη μεταφορά δεδομένων ανάμεσα σε έναν εξυπηρετητή βάσης δεδομένων και ενός πελάτη. Αρχικά σχεδιάστηκε και αναπτύχθηκε από την Sybase Inc. για σχεσιακή βάση δεδομένων του κινητήρα τους Sybase SQL Server 1984, και αργότερα από τη Microsoft στον Microsoft SQL Server, τα πακέτα TDS μπορούν να περικλείονται σε άλλα εξαρτώμενα πρωτόκολλα φυσικής μεταφοράς, συμπεριλαμβανομένου του πρωτοκόλλου TCP / IP, σε Named pipes, και στην κοινή μνήμη. Κατά συνέπεια, η πρόσβαση σε SQL Server είναι διαθέσιμη πάνω από αυτά τα πρωτόκολλα. Επιπλέον, το SQL Server API εκτίθεται επίσης πάνω από διαδικτυακές υπηρεσίες. [1]

2.2 Περιγραφή της τεχνολογίας MS SQL – Server 2014

Ο SQL Server 2014 παρέχει τη δυνατότητα στους πελάτες να δημιουργήσουν κρίσιμες εφαρμογές και λύσεις Big Data χρησιμοποιώντας υψηλής απόδοσης, τεχνολογία in-memory σε OLTP, αποθήκευση δεδομένων, επιχειρηματική ευφυΐα και analytics φόρτου εργασίας χωρίς να χρειάζεται να αγοράσουν ακριβά add-ons ή high-end συσκευές. Ο SQL Server 2014 χρησιμοποιεί ένα κοινό σύνολο εργαλείων για την ανάπτυξη και διαχείριση βάσεων δεδομένων, τόσο σε εγκαταστάσεις όσο και στο σύννεφο (Cloud), το οποίο καθιστά ευκολότερο για τους πελάτες να επωφεληθούν το Cloud με τις υπάρχουσες δεξιότητες. (http://www.microsoft.com/en-us/server-cloud/products/sql-server/)

Ειδικότερα, ο SQL Server 2014 παρέχει κρίσιμης σημασίας απόδοση σε όλο το φόρτο εργασίας με ενσωματωμένη μνήμη, γρηγορότερα γνώσεις από όλα τα δεδομένα με γνωστά εργαλεία, καθώς και μια πλατφόρμα για υβριδικά cloud επιτρέποντας στους οργανισμούς την εύκολη κατασκευή, ανάπτυξη και διαχείριση λύσεων που εκτείνονται στις εγκαταστάσεις και στο cloud.

(http://download.microsoft.com/download/D/7/D/D7D64E12-C8E5-4A8C-A104-C945C188FA99/SQL_Server_2014_Datasheet.pdf)

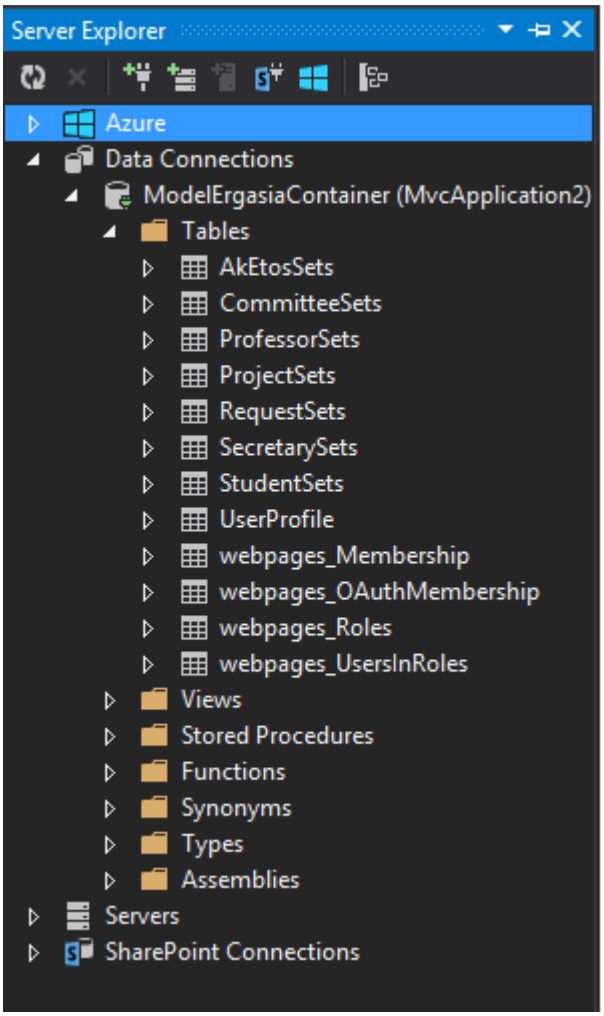
2.3 Ανάλυση του σχεδιασμού των πινάκων της Βάσης Δεδομένων

Σε αυτή την υποενότητα παρουσιάζεται αναλυτικά η βάση δεδομένων, δηλαδή οι πίνακες, τα πεδία τους, καθώς και οι μεταξύ τους συσχετίσεις. Η βάση δεδομένων της εφαρμογής χωρίζεται σε δύο μέρη, το ένα μέρος αντιπροσωπεύει τους πίνακες που είναι «υπεύθυνοι» για την αυθεντικοποίηση του χρήστη, περιλαμβάνοντας τα στοιχεία του, διάφορα στοιχεία ασφάλειας και κατάταξη των χρηστών σε ρόλους. Ενώ το άλλο μέρος, αποθηκεύει δεδομένα / πληροφορίες σχετικά με την εφαρμογή και καθιστά την ορθή λειτουργία της. Η βάση δεδομένων περιλαμβάνει τους παρακάτω βασικούς πίνακες:

- Χρήστες
- Στοιχεία των Χρηστών
- Ρόλοι
- Χρήστες σε Ρόλους

- Φοιτητές
- Διδάσκοντες
- Γραμματεία
- Εργασίες
- Αιτήσεις
- Επιτροπές
- Ακαδημαϊκά Έτη

Ακολουθεί ανάλυση των παραπάνω πινάκων και των πεδίων τους.



2.3.1 Πίνακας UserProfile

Αυτός ο πίνακας περιέχει τη «μισή» πληροφορία, μόνο το όνομα χρήστη, των χρηστών της εφαρμογής που σε συνδυασμό με τον κωδικό πρόσβασης, που βρίσκεται για διάφορους λόγους στον επόμενο πίνακα της βάσης, μπορούν αυθεντικοποιηθούν κατά την διαδικασία Log in (Σύνδεσης) στην εφαρμογή και να αποκτήσουν πρόσβαση στο σύστημα διαχείρισης πτυχιακών και διπλωματικών εργασιών. Ο κωδικός του χρήστη καθιστά την ύπαρξη του επόμενου πίνακα με περαιτέρω στοιχεία αλλά και στην κατάταξη των χρηστών σε ρόλους σε επόμενο πίνακα.

Update Script File: dbo.UserProfile.sql

Name	Data Type	Allow Nulls	Default	Length	Identity Increment
UserId	int	☐			1
UserName	nvarchar(56)	☐		56	
		☐			

2.3.2 Πίνακας webpages_Membership

Αυτός ο πίνακας περιέχει την άλλα «μισή» πληροφορία, τον κωδικό πρόσβασης, συνοδευόμενη από πολλά στοιχεία ασφάλειας του λογαριασμού κάθε χρήστη, όπως ειπώθηκε και παραπάνω, σε συνδυασμό με το όνομα χρήστη εδραιώνεται μία ευρεία μέθοδος αυθεντικοποίησης χρηστών. Ο κωδικός πρόσβασης του χρήστη αποθηκεύεται κρυπτογραφημένος με μια ψευδοτυχαία ακολουθία κατά τη δημιουργία του έτσι ώστε να παρέχεται, όσο το δυνατόν, μεγαλύτερη ασφάλεια.

Update Script File: dbo.webpages_Membership.sql

Name	Data Type	Allow Nulls	Default
UserId	int	☐	
CreateDate	datetime	☑	
ConfirmationToken	nvarchar(128)	☑	
IsConfirmed	bit	☑	((0))
LastPasswordFailureDate	datetime	☑	
PasswordFailuresSinceLastS	int	☐	((0))
Password	nvarchar(128)	☐	
PasswordChangedDate	datetime	☑	
PasswordSalt	nvarchar(128)	☐	
PasswordVerificationToken	nvarchar(128)	☑	
PasswordVerificationToken	datetime	☑	
		☐	

2.3.3 Πίνακας webpages_Roles

Ο πίνακας αυτός περιέχει τους ρόλους της εφαρμογής, που είναι συγκεκριμένα τρείς (Γραμματεία, Φοιτητές, Διδάσκοντες). Ο κωδικός του κάθε ρόλου θέτει νόημα ύπαρξης στον επόμενο πίνακα.

Update Script File: dbo.webpages_Roles.sql

	Name	Data Type	Allow Nulls	Default
🔑	RoleId	int	☐	
	RoleName	nvarchar(256)	☐	
			☐	

2.3.4 Πίνακας webpages_UsersInRoles

Ο πίνακας αυτός αντιστοιχίζει τους υπάρχοντες ρόλους με τους χρήστες της εφαρμογής, υπενθυμίζοντας ότι υποχρεωτικά κάθε χρήστης ανήκει σε κάποιον ρόλο.

Update Script File: dbo.webpages_UsersInRoles.sql

	Name	Data Type	Allow Nulls	Default
🔑	UserId	int	☐	
🔑	RoleId	int	☐	
			☐	

2.3.5 Πίνακας webpages_OAuthMembership

Ο πίνακας αυτός δεν ανήκει στους βασικούς πίνακες της εφαρμογής, γι' αυτό και δεν αναφέρθηκε παραπάνω, δίνει τη δυνατότητα σε εξωτερική αυθεντικοποίηση χρηστών μέσω κάποιου αξιόπιστου παρόχου, όπως για παράδειγμα σύνδεση με το λογαριασμό του Facebook, Twitter, Hotmail. Εδώ δεν υπάρχει αυτή η δυνατότητα, έχει αποκλεισθεί για ευνόητος λόγους.

Update Script File: dbo.webpages_OAuthMembershi

	Name	Data Type	Allow Nulls	Default
🔑	Provider	nvarchar(30)	☐	
🔑	ProviderUserId	nvarchar(100)	☐	
	UserId	int	☐	
			☐	

2.3.6 Πίνακας AkEtosSet

Ο πίνακας αυτός περιλαμβάνει τα Ακαδημαϊκά έτη της εφαρμογής που σε αυτά δηλώνονται εργασίες από τους διδάσκοντες, κανονίζονται γενικές συνελεύσεις και γενικότερα έχουν το ρόλο τους στην εφαρμογή.

Update Script File: dbo.AkEtosSets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐		1	
Date	nvarchar(MAX)	☐			MAX
		☐			

2.3.7 Πίνακας ComitteeSet

Ο πίνακας αυτός περιλαμβάνει τις εξεταστικές επιτροπές της εφαρμογής και προαιρετικά παρέχει και μηχανισμό βαθμολόγησης των εργασιών.

Update Script File: dbo.CommitteeSets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐		1	
Vathmos1	decimal(18,0)	☑			
Melos1	nvarchar(MAX)	☐			MAX
Melos2	nvarchar(MAX)	☐			MAX
Vatmos2	nvarchar(MAX)	☑			MAX
Epivlepwn	nvarchar(MAX)	☐			MAX
Vathmos	nvarchar(MAX)	☑			MAX
ProfessorSets_Id	int	☑			
		☐			

2.3.8 Πίνακας ProfessorSet

Ο πίνακας αυτός περιλαμβάνει τους διδάσκοντες εξωτερικούς ή μη της εφαρμογής συνοδευόμενους από τα απαραίτητα στοιχεία τους, διευκρινίζοντας ότι το πεδίο Email αντιστοιχεί στο Όνομα χρήστη της εφαρμογής, υψίστης σημασίας.

Update Script File: dbo.ProfessorSets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐		1	
Onomatepwnumo	nvarchar(MAX)	☐			MAX
Email	nvarchar(MAX)	☐			MAX
Secretary_Id	int	☑			
		☐			

2.3.9 Πίνακας ProjectSet

Ο πίνακας αυτός περιλαμβάνει τις πτυχιακές και διπλωματικές εργασίες της εφαρμογής με πλήθος στοιχείων προαιρετικών και μη για την καταχώρησή τους από τους διδάσκοντες.

Update Script File: dbo.ProjectSets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐		1	
Thema	nvarchar(MAX)	☐			MAX
Perigrafi	nvarchar(MAX)	☐			MAX
Vivliografia	nvarchar(MAX)	☐			MAX
Gnwseis	nvarchar(MAX)	☑			MAX
Energo	bit	☑			
CreateDate	datetime	☐			
ArFoititwn	smallint	☐			
Eksamino	nvarchar(MAX)	☐			MAX
Diathesimo	bit	☐			
EndDate	datetime	☐			
Eidos	nvarchar(MAX)	☐			MAX
AkEtos_Id	int	☐			
CommitteeSet_Id	int	☐			
ProfessorSet_Id	int	☑			
Paratiriseis	nvarchar(MAX)	☑			MAX
		☐			

2.3.10 Πίνακας RequestSet

Ο πίνακας αυτός περιλαμβάνει τις αιτήσεις που υποβάλουν οι προπτυχιακοί και μεταπτυχιακοί φοιτητές συνοδευόμενες από τα απαραίτητα στοιχεία.

Update | Script File: dbo.RequestSets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐		1	
RequestDate	datetime	☐			
Egkrisi	nvarchar(MAX)	☑			MAX
Student_Id	int	☑			
Project_Id	int	☐			
		☐			

2.3.11 Πίνακας SecretarySet

Ο πίνακας αυτός περιλαμβάνει τις γραμματείες της εφαρμογής με τον αντίστοιχο κωδικό τους όπου χρησιμοποιείται για τον έλεγχο ως προς το κομμάτι της διαχείρισης.

Update | Script File: dbo.SecretarySets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐			
		☐			

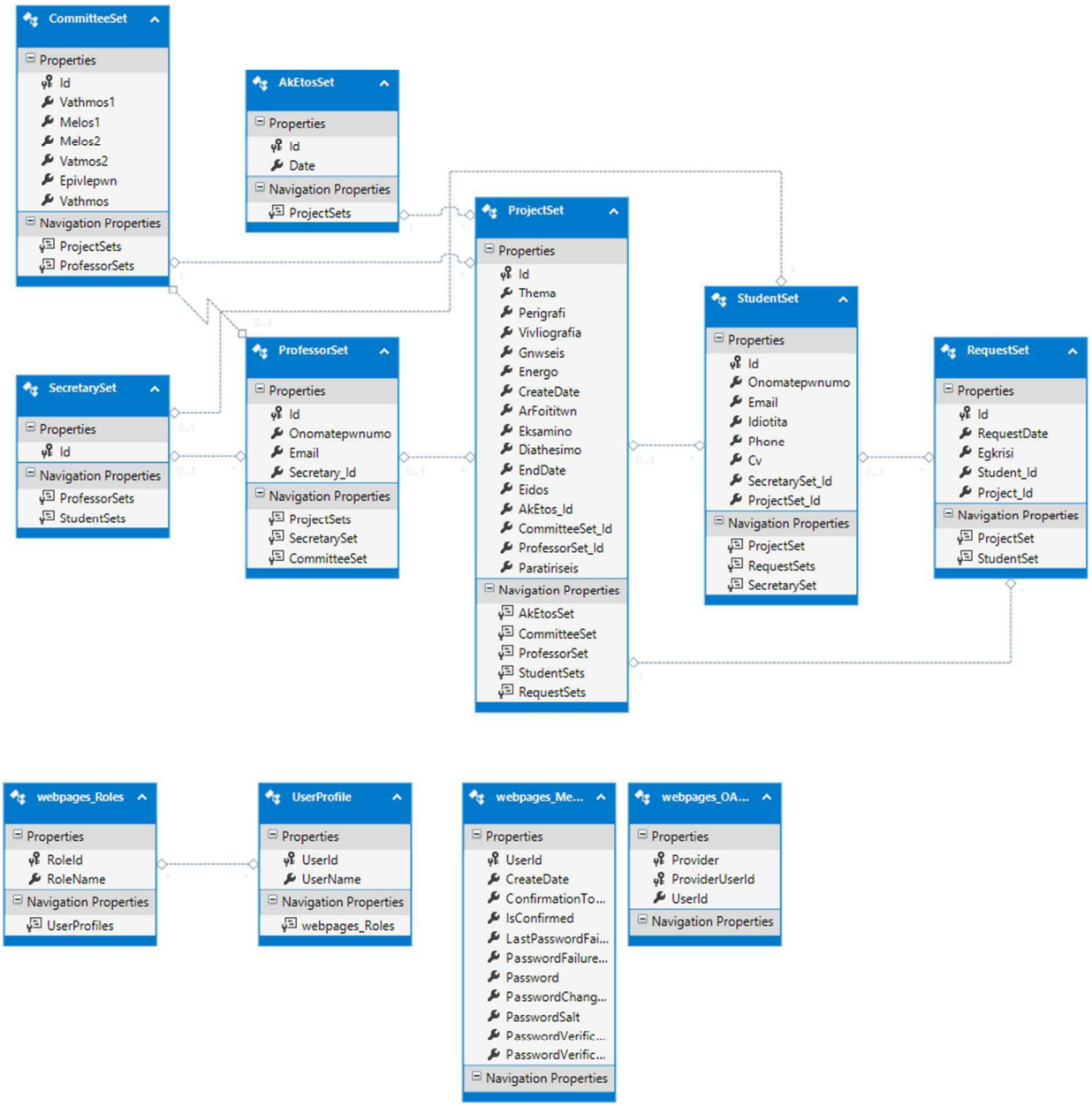
2.3.12 Πίνακας StudentSet

Ο πίνακας αυτός περιλαμβάνει τους φοιτητές προπτυχιακούς και μεταπτυχιακούς της εφαρμογής συνοδευόμενους από τα απαραίτητα στοιχεία τους, διευκρινίζοντας ότι το πεδίο Email αντιστοιχεί στο Όνομα χρήστη της εφαρμογής, υψίστης σημασίας.

Update Script File: dbo.StudentSets.sql

Name	Data Type	Allow Nulls	Default	Identity Increment	Length
Id	int	☐			
Onomatepwnumo	nvarchar(MAX)	☐			MAX
Email	nvarchar(MAX)	☐			MAX
Idiotita	nvarchar(MAX)	☐			MAX
Phone	nvarchar(MAX)	☐			MAX
Cv	nvarchar(MAX)	☑			MAX
SecretarySet_Id	int	☑			
ProjectSet_Id	int	☑			
		☐			

2.4 Διάγραμμα Οντοτήτων – Συσχετίσεων (ER - DIAGRAM)



2.5 Αναφορές & Βιβλιογραφία

[1] Kalen Delaney. Inside Microsoft SQL Server 2005: The Storage Engine. Microsoft Press. ISBN 0-7356-2105-5.

Συλλογικό έργο. Συστήματα βάσεων δεδομένων : Η πλήρης θεωρία των βάσεων δεδομένων / Συλλογικό έργο, Abraham Silberschatz, Henry F. Korth, S. Sudarshan · μετάφραση Μαρία Γκλαβά. - 4η έκδ. - Αθήνα : Γκιούρδας Μ., 2004.

Free Microsoft ebook: Introducing Microsoft SQL Server 2014, Ross Mistry and Stacia Misner, April 2014. (Microsoft Virtual Academy, http://www.microsoftvirtualacademy.com/ebooks)

ΚΕΦΑΛΑΙΟ 3

ΠΛΑΤΦΟΡΜΕΣ & ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ

3.1 .NET Framework

Το .NET Framework είναι μια πλατφόρμα ανάπτυξης για τη δημιουργία εφαρμογών για Windows, Windows Phone, Windows Server και Microsoft Azure. Αποτελείται από το common language runtime (CLR) και τη .NET Framework class library, η οποία περιλαμβάνει κλάσεις, interfaces και value types που υποστηρίζουν ένα ευρύ φάσμα τεχνολογιών. Το πλαίσιο .NET παρέχει ένα διαχειριζόμενο περιβάλλον εκτέλεσης, απλοποιημένης ανάπτυξης και εγκατάστασης, ενσωμάτωσης μιας ποικιλίας γλωσσών προγραμματισμού, συμπεριλαμβανομένης της Visual Basic και Visual C#.

Η τεκμηρίωση για το πλαίσιο .NET (Εικόνα 1) περιλαμβάνει μια εκτενή αναφορά της class library, εννοιολογικές επισκοπήσεις, βήμα-βήμα διαδικασίες, καθώς και πληροφορίες σχετικά με τα δείγματα (samples), τους compilers και τα εργαλεία της γραμμής εντολών (Εικόνα 3). (https://msdn.microsoft.com/en-us/library/w0x726c2%28v=vs.110%29.aspx)

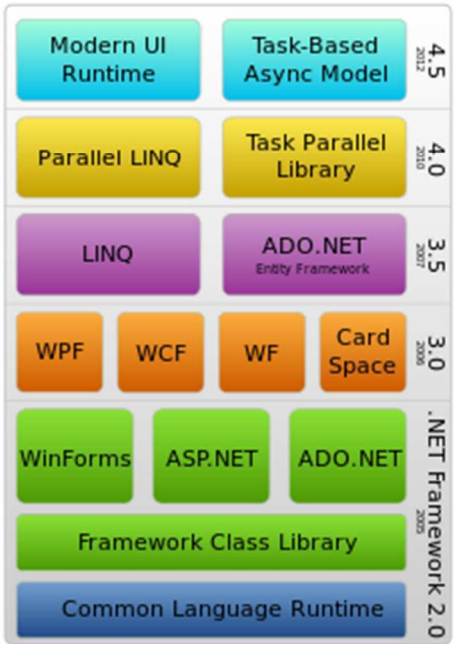
Το .NET προσφέρει:

- Interoperability
- Common Language Runtime engine
- Language independence
- Framework Class Library
- Simplified deployment
- Security
- Portability

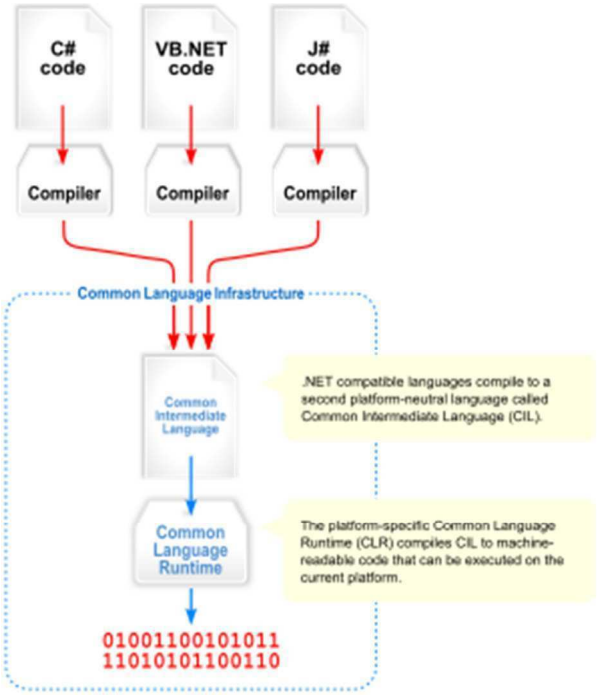
Η Αρχιτεκτονική του .NET Framework:

- Common Language Infrastructure (CLI) (Εικόνα 2)
- Security
- Class library
- Memory management

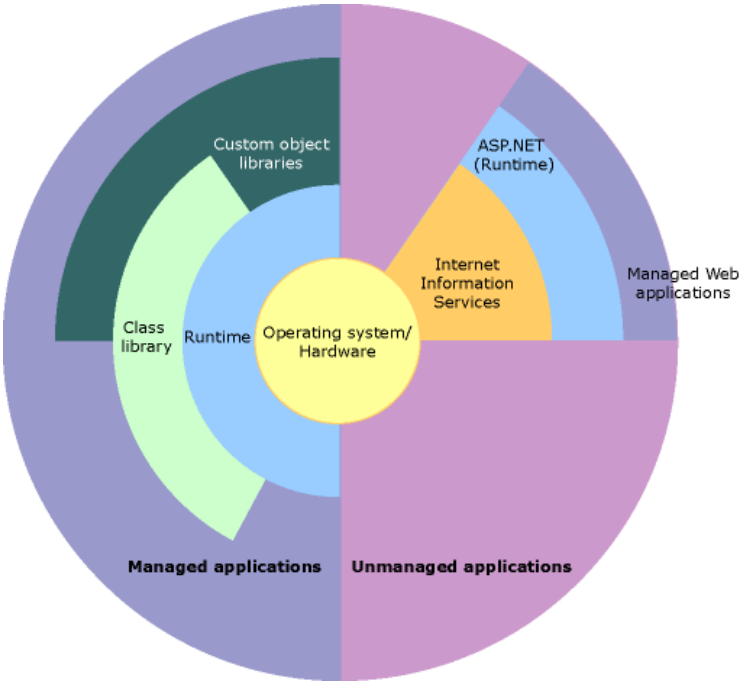
(http://en.wikipedia.org/wiki/.NET_Framework)



Εικόνα 1: The .NET Framework Stack (http://www.microsoft.com/net)



Εικόνα 2: Visual overview of the Common Language Infrastructure (CLI) (http://en.wikipedia.org/wiki/Common_Language_Infrastructure)



Εικόνα 3: .NET Framework in context (https://msdn.microsoft.com/en-us/library/zw4w595w(v=vs.110).aspx)

3.2 Microsoft Visual Studio

Το Microsoft Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft. Χρησιμοποιείται για την ανάπτυξη προγραμμάτων ηλεκτρονικών υπολογιστών για τα Microsoft Windows, καθώς και ιστοσελίδες, διαδικτυακές εφαρμογές και υπηρεσίες web. Το Visual Studio χρησιμοποιεί πλατφόρμες ανάπτυξης λογισμικού της Microsoft, όπως Windows API, Windows Forms, Windows Presentation Foundation, Windows Store και το Microsoft Silverlight. Μπορεί να παράγει εγγενή και διαχειριζόμενο κώδικα.

Το Visual Studio περιλαμβάνει έναν επεξεργαστή κώδικα που υποστηρίζει το IntelliSense (ένα component συμπλήρωσης κώδικα), καθώς επίσης και code refactoring. Το ολοκληρωμένο πρόγραμμα εντοπισμού σφαλμάτων λειτουργεί ως ένα πρόγραμμα εντοπισμού σφαλμάτων πηγαίου κώδικα (source - level debugger) και ως ένα πρόγραμμα εντοπισμού σφαλμάτων «μηχανής» (machine - level debugger). Άλλα ενσωματωμένα εργαλεία είναι η σχεδίαση φόρμας για τη δημιουργία εφαρμογών GUI, η σχεδίαση ιστοσελίδων, η σχεδίαση κλάσεων και η σχεδίαση σχήματος βάσης δεδομένων. Δέχεται plug-ins που ενισχύουν τη λειτουργικότητα σχεδόν σε κάθε επίπεδο, προσθέτοντας υποστήριξη για συστήματα πηγαίου κώδικα (source - control systems), όπως η Subversion, προσθήκη νέων toolsets όπως editors και visual designers για domain - specific γλώσσες προγραμματισμού καθώς και toolsets για άλλες πτυχές κύκλου ανάπτυξης λογισμικού (όπως τον client Team Foundation Server: Team Explorer).

Το Visual Studio υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και παρέχει code editor και debugger (σε διάφορους βαθμούς) σχεδόν σε οποιαδήποτε γλώσσα, εφόσον υπάρχει μια language – specific υπηρεσία. Built - in γλώσσες αποτελούν C, [1], C ++ και C ++ / CLI (μέσω της Visual C ++), VB.NET (μέσω της Visual Basic .NET), C # (μέσω Visual C #), και F # (όπως και στο Visual Studio 2010 [2]). Υποστήριξη για άλλες γλώσσες, όπως η M, η Python και η Ruby, μεταξύ άλλων, είναι διαθέσιμες μέσω των υπηρεσιών εγκατάστασης γλωσσών που παρέχεται, για κάθε μια ξεχωριστά. Υποστηρίζει, επίσης, XML / XSLT, HTML / XHTML, JavaScript και CSS.

Η Microsoft παρέχει "Express" εκδόσεις του Visual Studio του χωρίς κανένα κόστος. Εμπορικές εκδόσεις του Visual Studio μαζί με επιλεγμένες προηγούμενες εκδόσεις διατίθενται δωρεάν στους φοιτητές μέσω του προγράμματος DreamSpark της Microsoft. [3] (http://en.wikipedia.org/wiki/Microsoft_Visual_Studio)

3.4 Αναφορές & Βιβλιογραφία

[1] Brenner, Pat (19 July 2013). "C99 library support in Visual Studio 2013". Visual C++ Team Blog. Microsoft. Retrieved 3 August 2014.

[2] http://research.microsoft.com/en-us/um/cambridge/projects/fsharp/

[3] Microsoft DreamSpark - Software Catalog. Dreamspark.com (2013-05-31). Retrieved on 2013-10-23.

ΚΕΦΑΛΑΙΟ 4

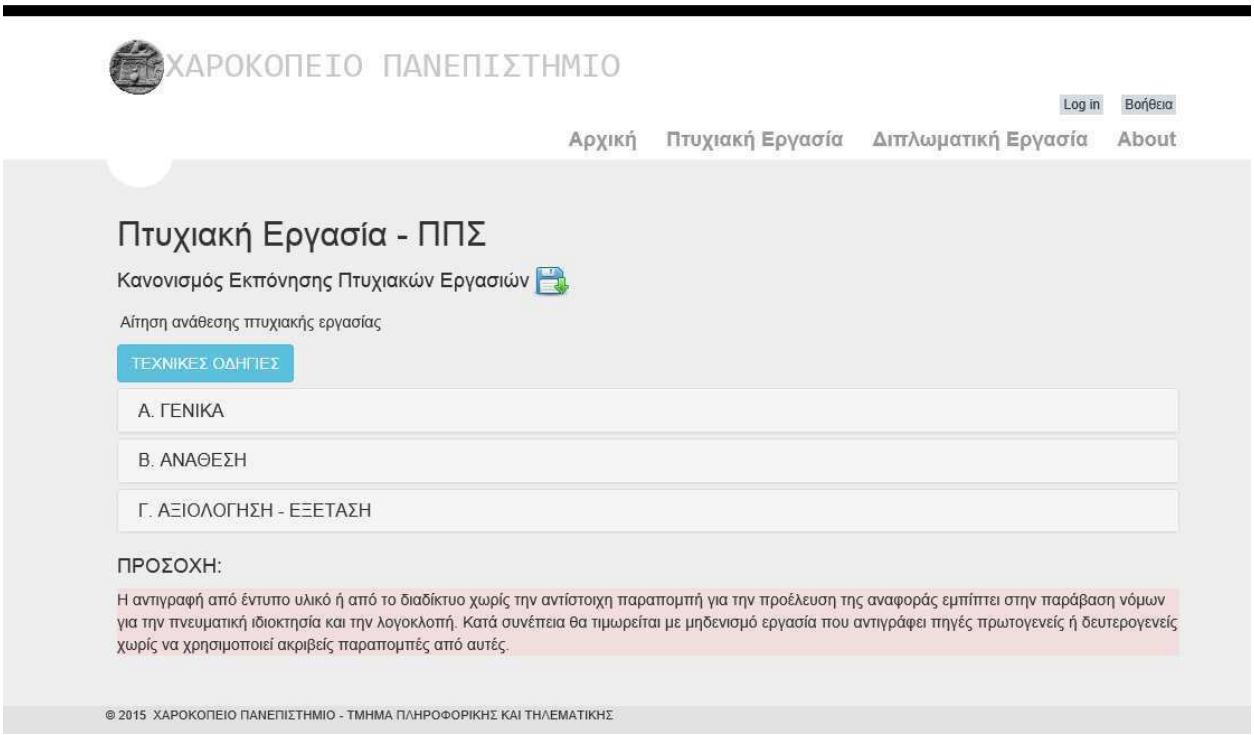
ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ

4.1 Αρχική σελίδα (Index.aspx)

Η Αρχική Σελίδα της εφαρμογής παρέχει μέσα από ένα απλό Interface (Διεπαφή) διάφορες λειτουργίες που αφορούν έναν επισκέπτη της εφαρμογής. Υπάρχει μενού επιλογών με σκοπό την ενημέρωση και τη διευκόλυνση ως προς τις απαιτούμενες διαδικασίες ανάληψης Πτυχιακής/Διπλωματικής Εργασίας. Τέλος, παρέχονται τα στοιχεία της εφαρμογής και των δημιουργών της αλλά και διαδικασία αυθεντικοποίησης των χρηστών, πατώντας *Log in* και *Βοήθεια*. Η βοήθεια που παρέχεται δεν είναι κάτι άλλο παρά το εγχειρίδιο χρήστη τμηματικά, ανάλογα με το είδος του χρήστη.

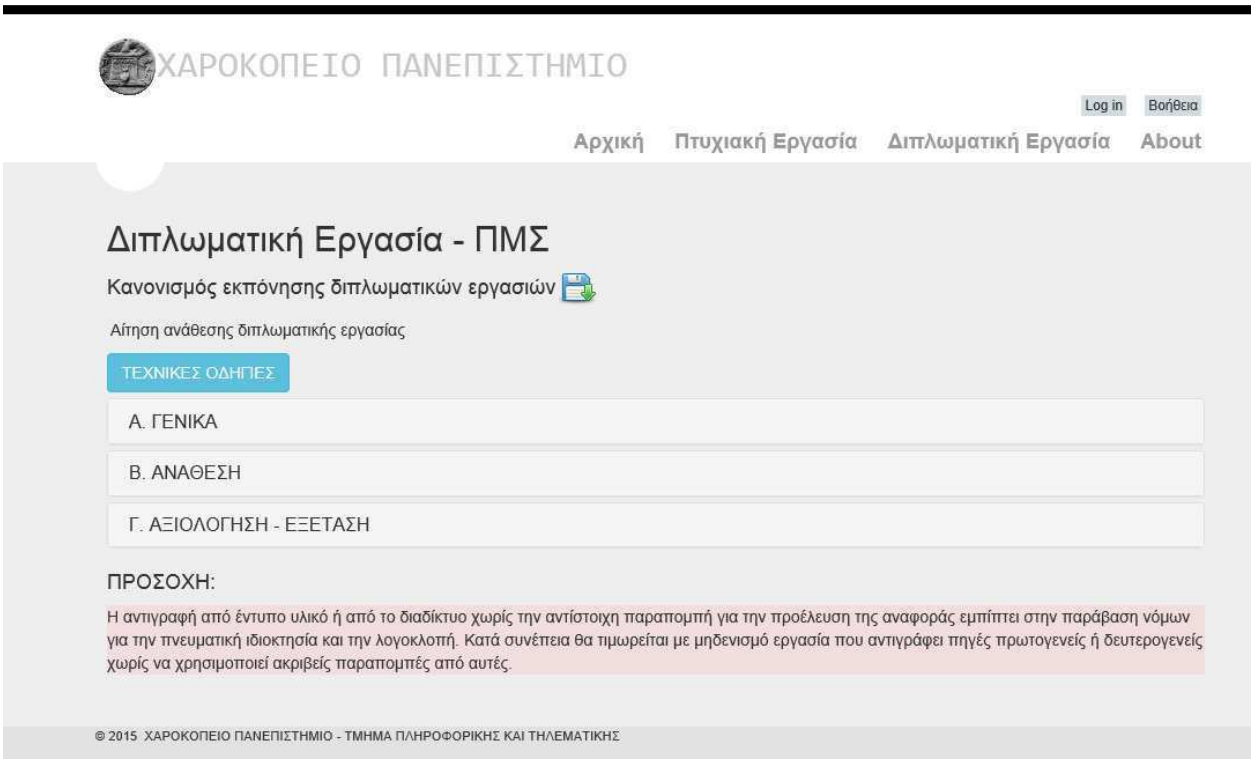
4.2 Πτυχιακή Εργασία (Ptyxiaki.aspx)

Η σελίδα Πτυχιακή Εργασία παρέχει στους προπτυχιακούς φοιτητές όλες τις πληροφορίες που αφορούν τη διαδικασία εκπόνησής της. Τον κανονισμό εκπόνησης πτυχιακών εργασιών υπάρχει δυνατότητα λήψης του σε αρχείο (.pdf). Επιπλέον, περιέχει χρήσιμους συνδέσμους (links) όπως την αίτηση ανάθεσης πτυχιακής εργασίας για κατάθεση στη Γραμματεία του τμήματος, απευθείας απόθεση στο αποθετήριο της Βιβλιοθήκης του Πανεπιστημίου και διάφορες άλλες πληροφορίες που συνοδεύουν τη διαδικασία.



4.3 Διπλωματική Εργασία (Diplomatiki.aspx)

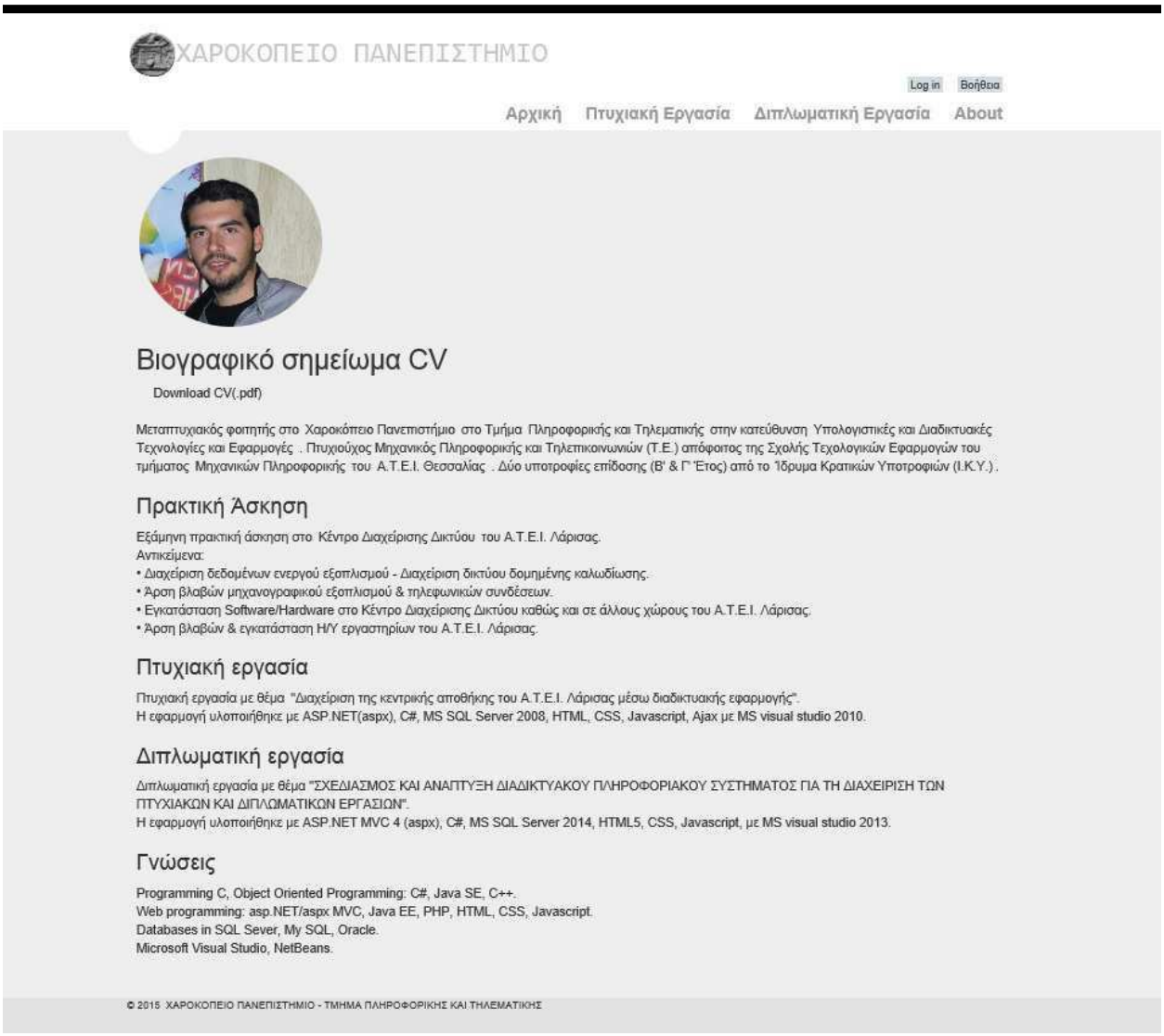
Η σελίδα Διπλωματική Εργασία παρέχει στους μεταπτυχιακούς φοιτητές όλες τις πληροφορίες που αφορούν τη διαδικασία εκπόνησης της διατριβής. Τον κανονισμό εκπόνησης διπλωματικών εργασιών υπάρχει δυνατότητα λήψης του σε αρχείο (.pdf). Επιπλέον, περιέχει χρήσιμους συνδέσμους (links) όπως την αίτηση ανάθεσης διπλωματικής εργασίας για κατάθεση στη Γραμματεία του τμήματος, απευθείας απόθεση στο αποθετήριο της Βιβλιοθήκης του Πανεπιστημίου και διάφορες άλλες πληροφορίες που συνοδεύουν τη διαδικασία.



4.4 About - Δημιουργία (About.aspx)

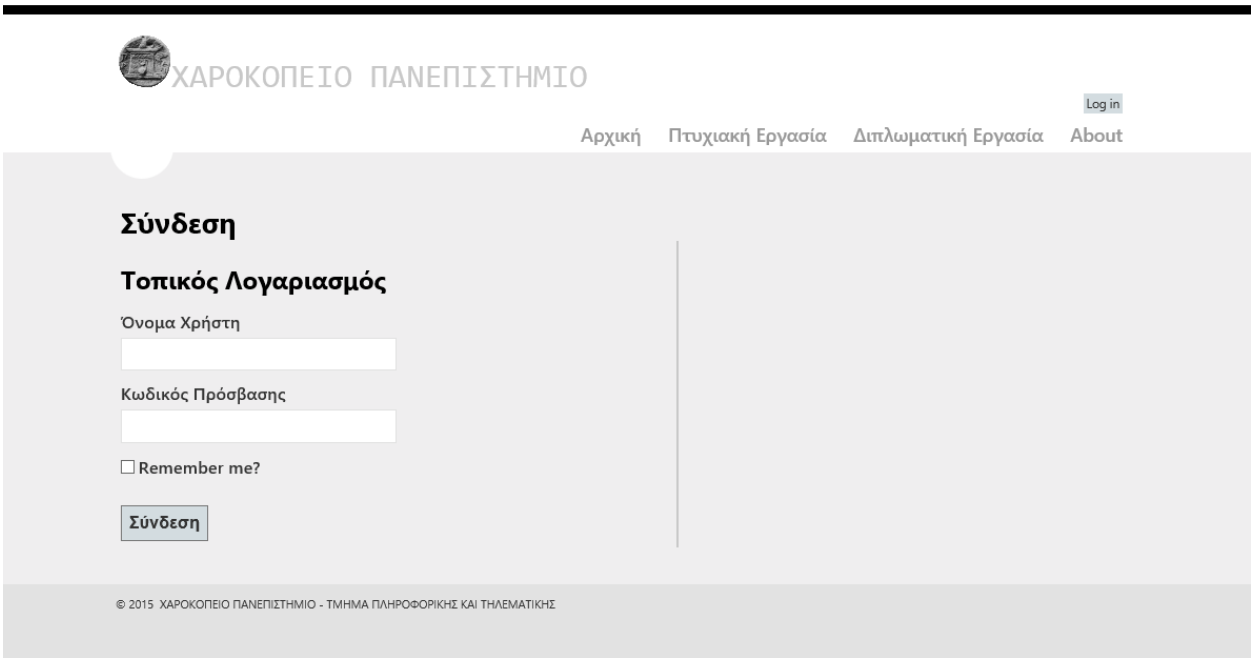
Στη σελίδα About παρέχονται τα στοιχεία της εφαρμογής και των δημιουργών της.

Επιπλέον, περιέχει ένα micro – site με τα στοιχεία του δημιουργού της εφαρμογής.

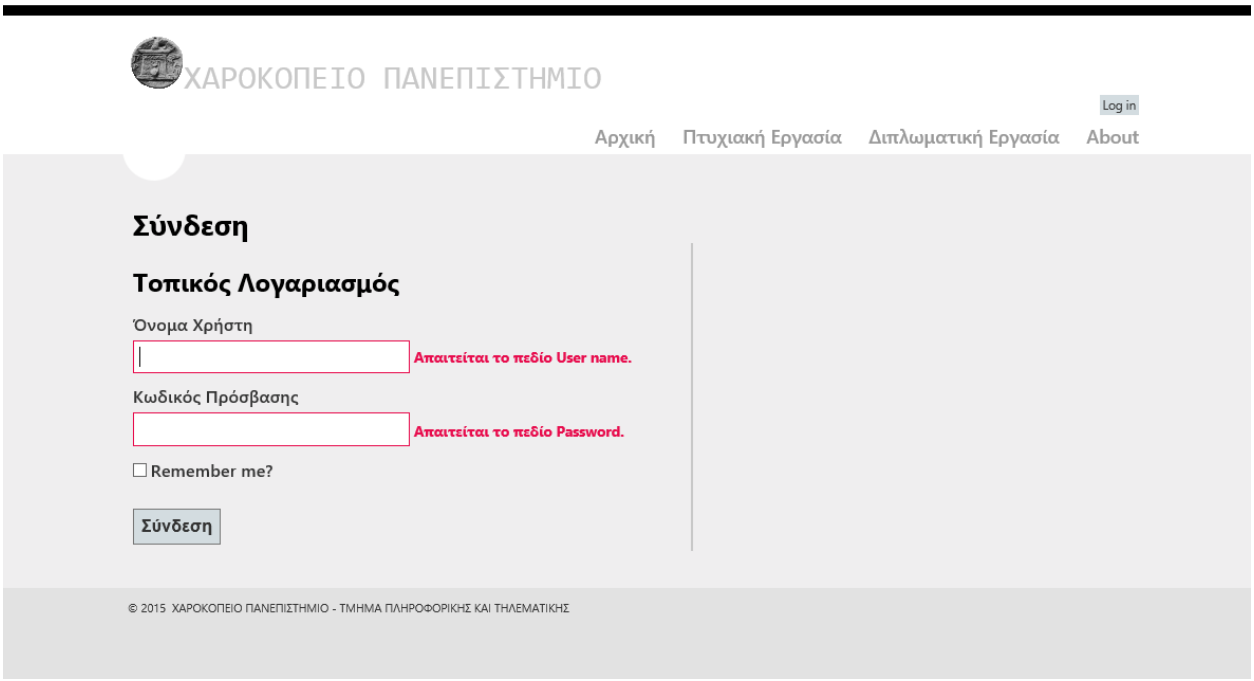


4.5 Log in – Σύνδεση (Login.aspx)

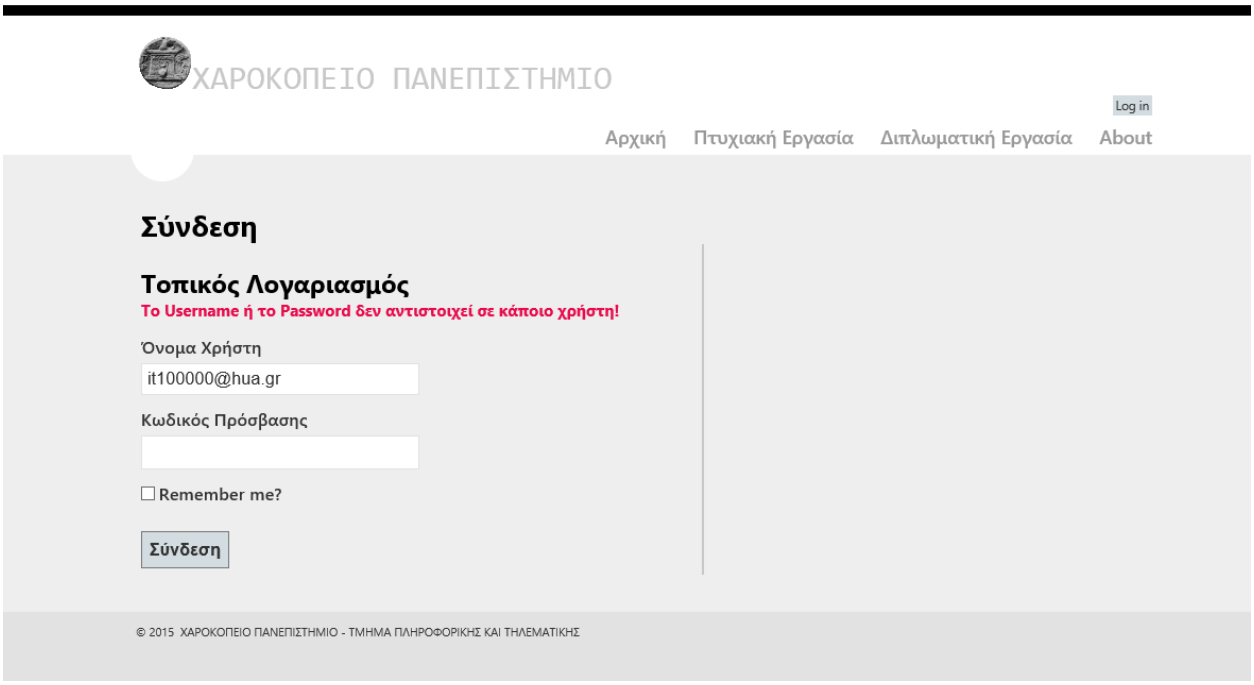
Στη σελίδα Log in ξεκινά η διαδικασία αυθεντικοποίησης των χρηστών, οι χρήστες της εφαρμογής ανήκουν σε τρία είδη ρόλων. Ο χρήστης μπορεί να είναι Φοιτητής/ρια, Καθηγητής/ρια ή Γραμματεία. Στις ενότητες 4.5.1 & 4.5.2 ακολουθούν οι οθόνες με τα μηνύματα λάθους της σελίδας.



4.5.1 Log in – Σύνδεση (Login.aspx) με κάποιο από τα πεδία κενό

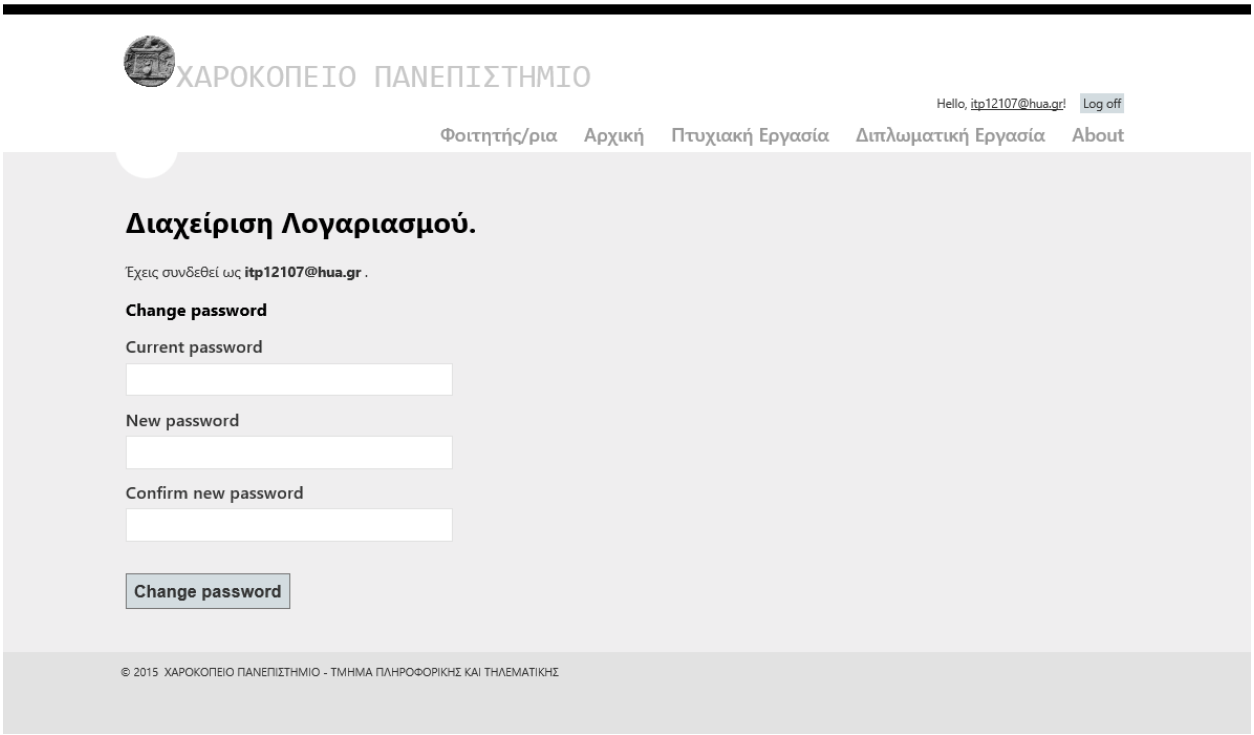


4.5.2 Log in – Σύνδεση (Login.aspx) με λάθος username / password



4.6 Manage – Διαχείριση Λογαριασμού (Manage.aspx)

Η σελίδα Manage παρέχει ένα μηχανισμό αλλαγής κωδικού πρόσβασης (password). Είναι όμοια για κάθε αυθεντικοποιημένο χρήστη του συστήματος, ανοίγει πατώντας στο e-mail δίπλα στο Hello στο επάνω δεξιά κομμάτι της σελίδας.

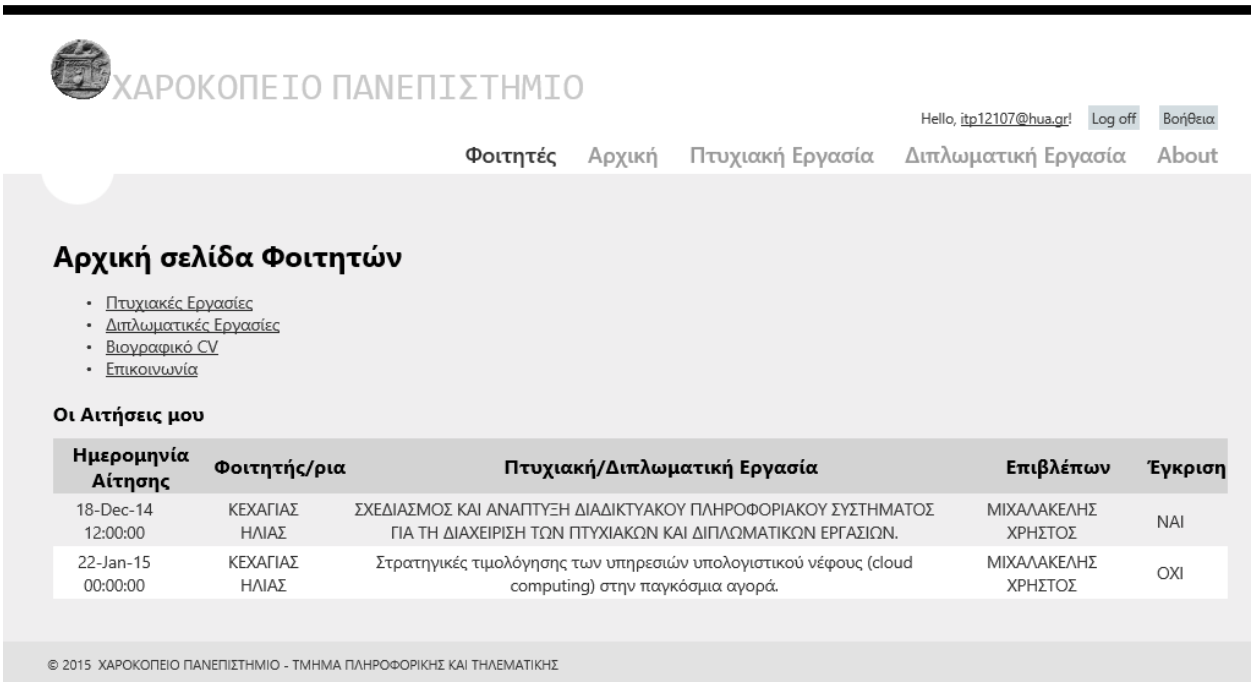


4.7 Αρχική Σελίδα – Συνδεδεμένος/η Φοιτητής/ρια (Index.aspx)

Η Αρχική προσωπική σελίδα των φοιτητών με νέο μενού που δίνει πρόσβαση στις επιμέρους διαδικασίες όπως η προβολή των διαθέσιμων εργασιών, η υποβολή αιτήσεων και η επικοινωνία.

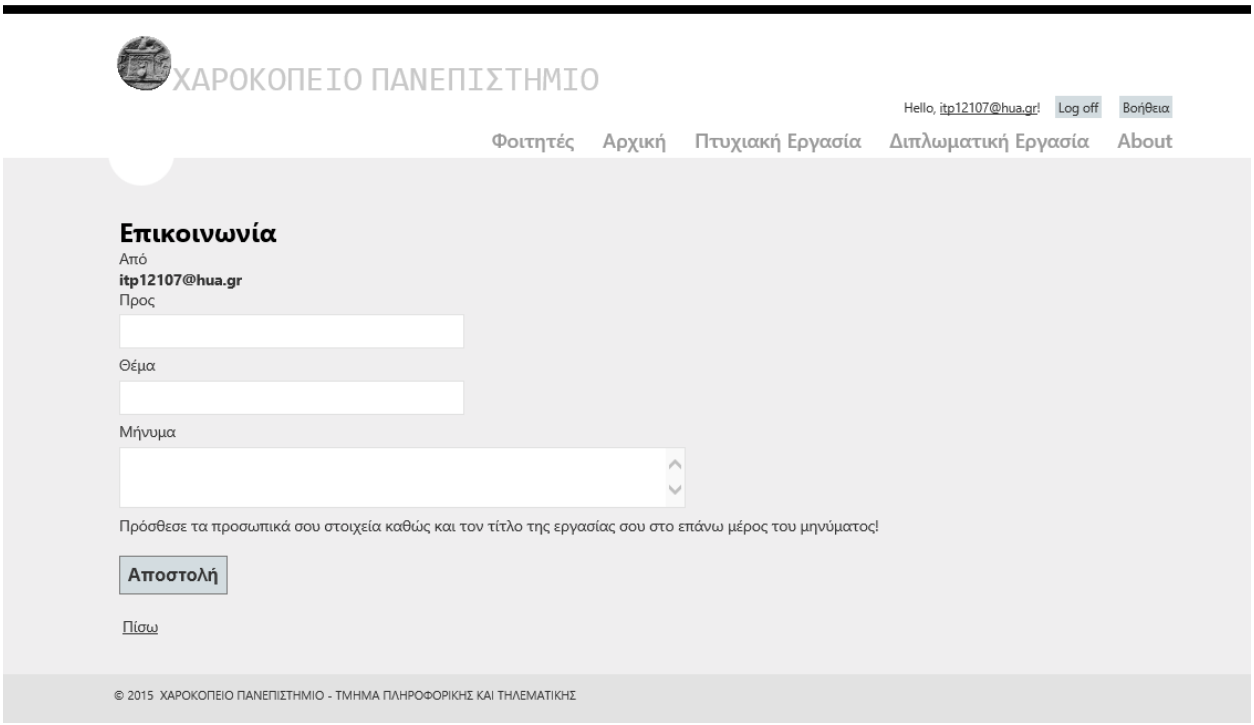
4.8 Αρχική Φοιτητές (Student.aspx)

Η σελίδα αυτή εμφανίζεται πατώντας στο πρώτο κεντρικό μενού της εφαρμογής, εμφανίζει τις αιτήσεις του φοιτητή και την πορεία τους. Παράλληλα, στο υπό-μενού περιέχει διάφορες λειτουργίες.

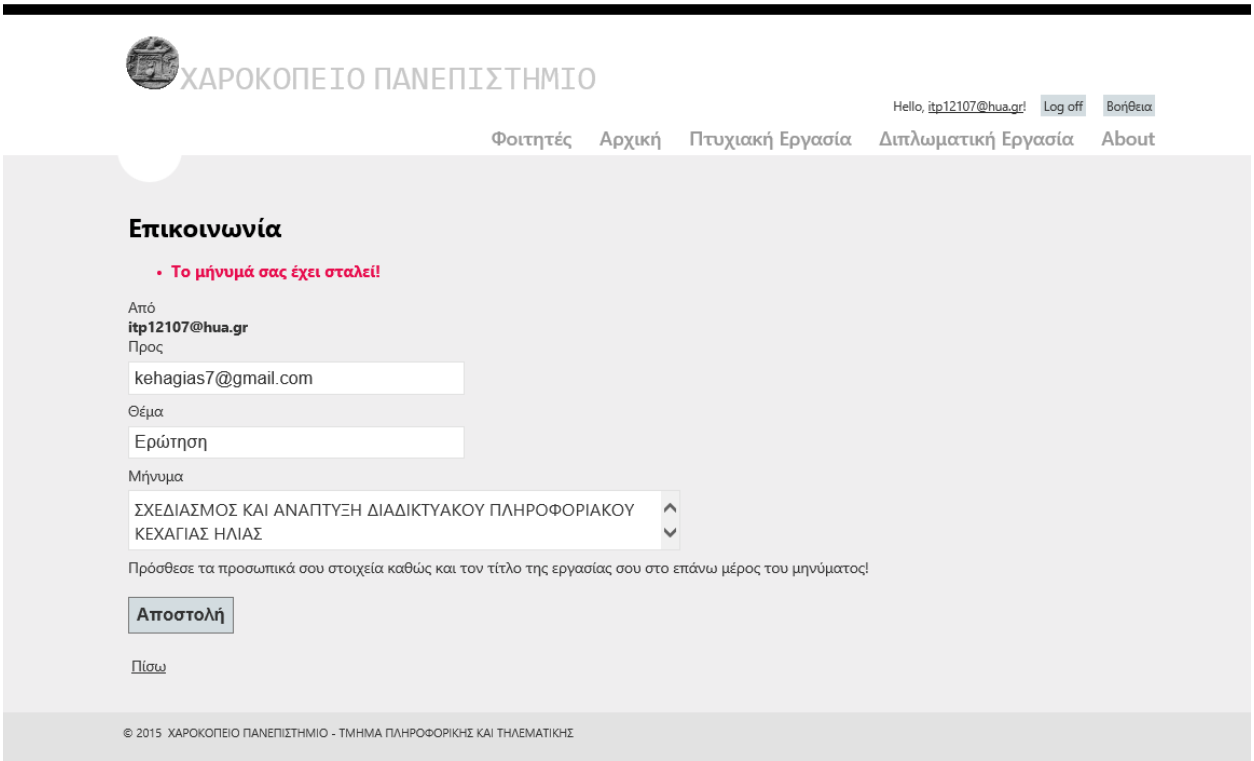


4.9 Επικοινωνία (Contact.aspx)

Παρέχει επικοινωνία με αποστολή e-mail προς κάποιο καθηγητή/φοιτητή που αφορά εργασία είτε για ενημέρωση είτε για συμμετοχή σε ομαδική εργασία, αντίστοιχα. Ο χρήστης μπορεί να στείλει το μήνυμα σε πολλούς παραλήπτες ταυτόχρονα αρκεί στο πεδίο *Προς* να εισάγει τις σωστές διευθύνσεις ηλεκτρονικού ταχυδρομείου χωρισμένες με κόμμα, όπως ακριβώς γίνεται σε οποιοδήποτε πάροχο λόγου χάρη στο Gmail.



4.9.1 Επικοινωνία με επιτυχής αποστολή e-mail (Contact.aspx)



4.10 Βιογραφικό Σημείωμα CV (Cv.aspx)

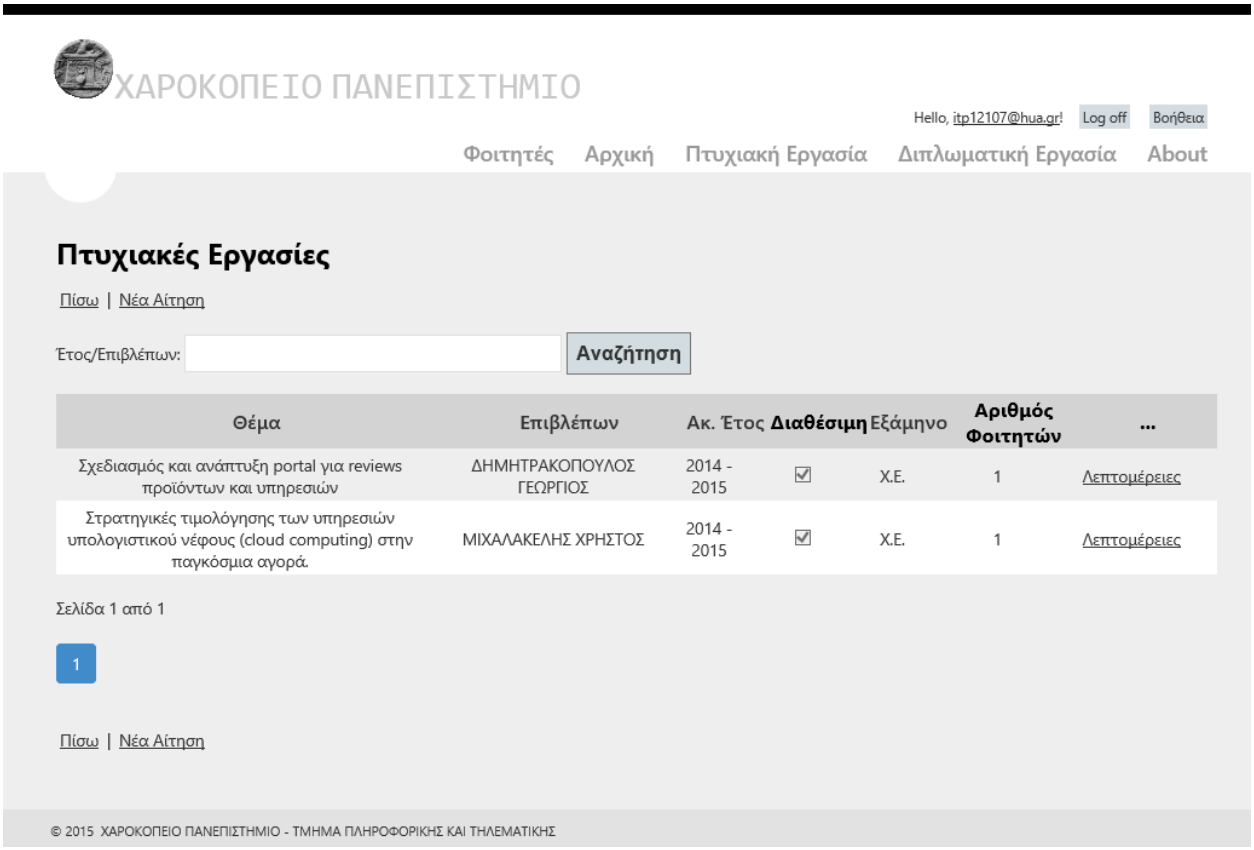
Στη σελίδα αυτή κάθε φοιτητής/ρια μπορεί να παραμετροποιήσει το σύντομο βιογραφικό σημείωμα που παρέχει η εφαρμογή σε μορφή κειμένου με σκοπό την προώθηση της αίτησής του/της για επιλογή από τον επιβλ. Καθηγητή/ρια.

4.10.1 Βιογραφικό Σημείωμα CV με επιτυχής ενημέρωση (Cv.aspx)

4.11 Πτυχιακές Εργασίες (Proptyxiakes.aspx)

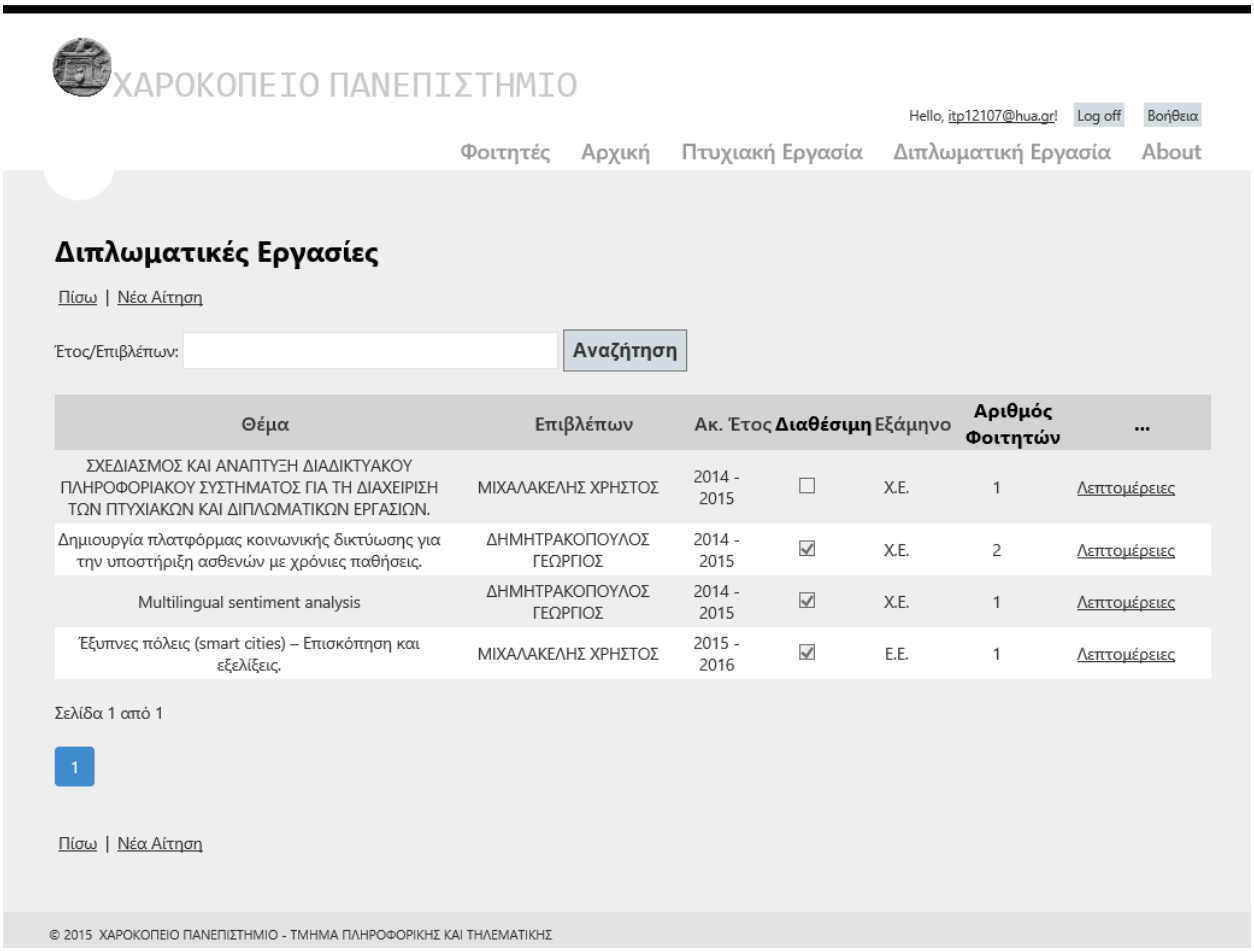
Από τη σελίδα αυτή κάθε φοιτητής/ρια μπορεί να δει τα διαθέσιμα ενδεικτικά θέματα ανά καθηγητή/ρια με περισσότερες πληροφορίες από το link Λεπτομέρειες στην τελευταία στήλη καθώς και να προβεί σε νέα Αίτηση. Η διαδικασία νέας αίτησης πραγματοποιείται με δυο τρόπους είτε από τον link Νέα Αίτηση (που βρίσκεται στο πάνω και στο κάτω μέρος της σελίδας) είτε από το link Λεπτομέρειες στην τελευταία στήλη, ακολουθώντας τα βήματα και στους δύο τρόπους αντίστοιχα (ακολουθεί αναλυτική περιγραφή).

Παράλληλα, παρέχεται ταξινόμηση ορισμένων πεδίων, όπου κρίνεται απαραίτητο, κάνοντας «κλικ» στον τίτλο της στήλης. Επίσης, υπάρχει διακριτός διαχωρισμός των γραμμών και των στηλών του πίνακα για εύκολη περιήγηση στα δεδομένα, πολλαπλή (περισσότερη του ενός πεδίου) διαδικασία αναζήτησης και σελιδοποίηση των αποτελεσμάτων (20 εγγραφές ανά σελίδα) με απλό και κατανοητό τρόπο στο κάτω μέρος της σελίδας με σκοπό την αποφυγή μεγάλου όγκου πληροφορίας στον χρήστη.



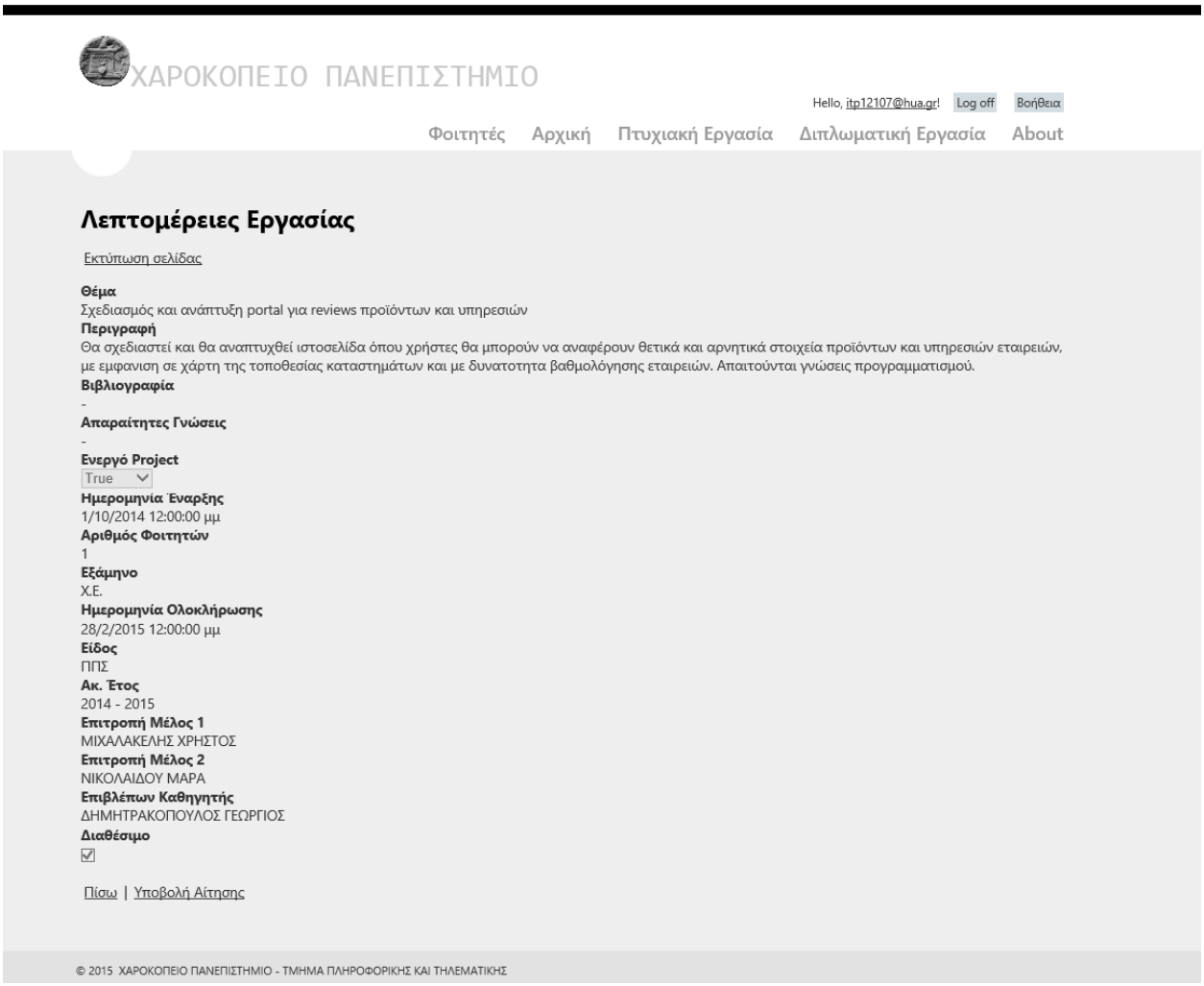
4.12 Διπλωματικές Εργασίες (Metaptyxiakes.aspx)

Ομοίως, αλλά για μεταπτυχιακούς φοιτητές.



4.13 Λεπτομέρειες (DetailsProject.aspx)

Προβάλει περισσότερες πληροφορίες που αφορούν στην επιλεγμένη εργασία από τον/την φοιτητή/ρια και δίνει τη δυνατότητα εκτύπωσης της σελίδας πληροφοριών. Τέλος, παρέχει τη δυνατότητα *Υποβολής Αίτησης* της εργασίας.



Η σελίδα εκτύπωσης επιλεγμένης εργασίας περιέχει περισσότερες λεπτομέρειες με σκοπό την καλύτερη δυνατή ενημέρωση των φοιτητών.

Λεπτομέρειες Εργασίας Page 1 of 2

ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

Hello, itp12107@hua.gr! Log off Βοήθεια

Φοιτητές Αρχική Πτυχιακή Εργασία Διπλωματική Εργασία About

Λεπτομέρειες Εργασίας

Εκτύπωση σελίδας

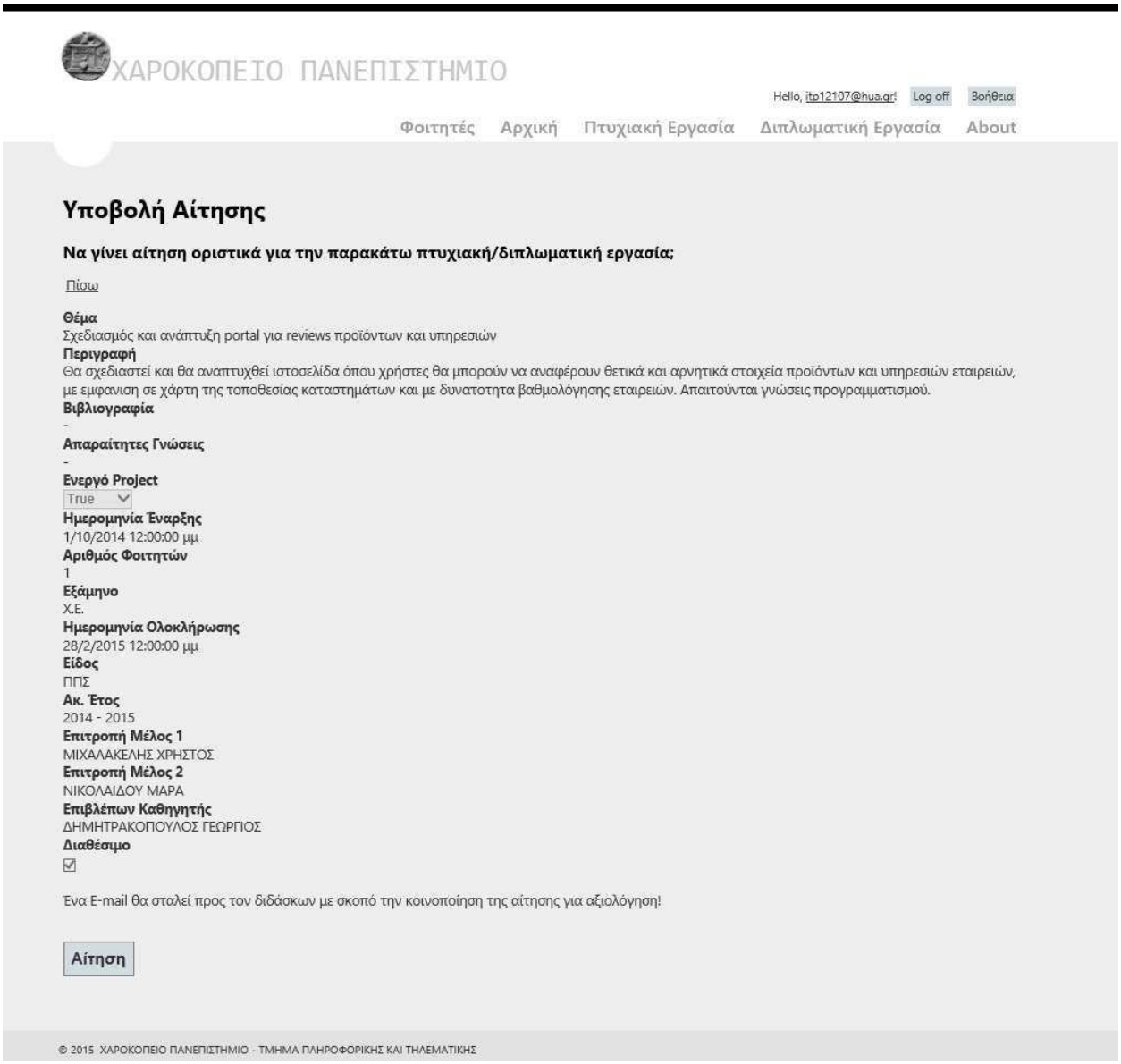
Είδος
ΠΜΣ
Ακ. Έτος
2014 - 2015
Εξάμηνο
Χ.Ε.
Θέμα
ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΠΤΥΧΙΑΚΩΝ ΚΑΙ ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ.
Περιγραφή
Η παρούσα εργασία αφορά στην ανάλυση απαιτήσεων, το σχεδιασμό και την ανάπτυξη ενός διαδικτυακού πληροφοριακού συστήματος για την ανάρτηση θεμάτων πτυχιακών και διπλωματικών εργασιών, τη διαχείριση αιτήσεων ανάληψης εργασιών εκ μέρους των φοιτητών, την παραγωγή στατιστικών και αναφορών και άλλες σχετικές εργασίες. Το πληροφοριακό σύστημα που θα αναπτυχθεί θα τεθεί σε κατάσταση κανονικής λειτουργίας για τις εκπαιδευτικές ανάγκες των προγραμμάτων σπουδών του Τμήματος.
Βιβλιογραφία
www.msdn.com
Απαραίτητες Γνώσεις
Γνώσεις προγραμματισμού σε διαδικτυακό περιβάλλον.
Ημερομηνία Έναρξης
01-Nov-14 12:00:00
Ημερομηνία Ολοκλήρωσης
28-Feb-15 12:00:00
Αριθμός Φοιτητών
1
Διαθέσιμο
☐
Επιβλέπων Καθηγητής
ΜΙΧΑΛΑΚΕΛΗΣ ΧΡΗΣΤΟΣ

Πίσω

http://localhost:1146/Student/DetailsProject/1 11-Feb-15

4.14 Υποβολή Αίτησης (NewRequest.aspx)

Σελίδα Υποβολής Αίτησης για την επιλεγμένη εργασία του συνδεδεμένου φοιτητή και με αποστολή e-mail προς τον αντίστοιχο διδάσκοντα με όλα τα απαραίτητα στοιχεία για αποδοχή/απόρριψη της αίτησης.

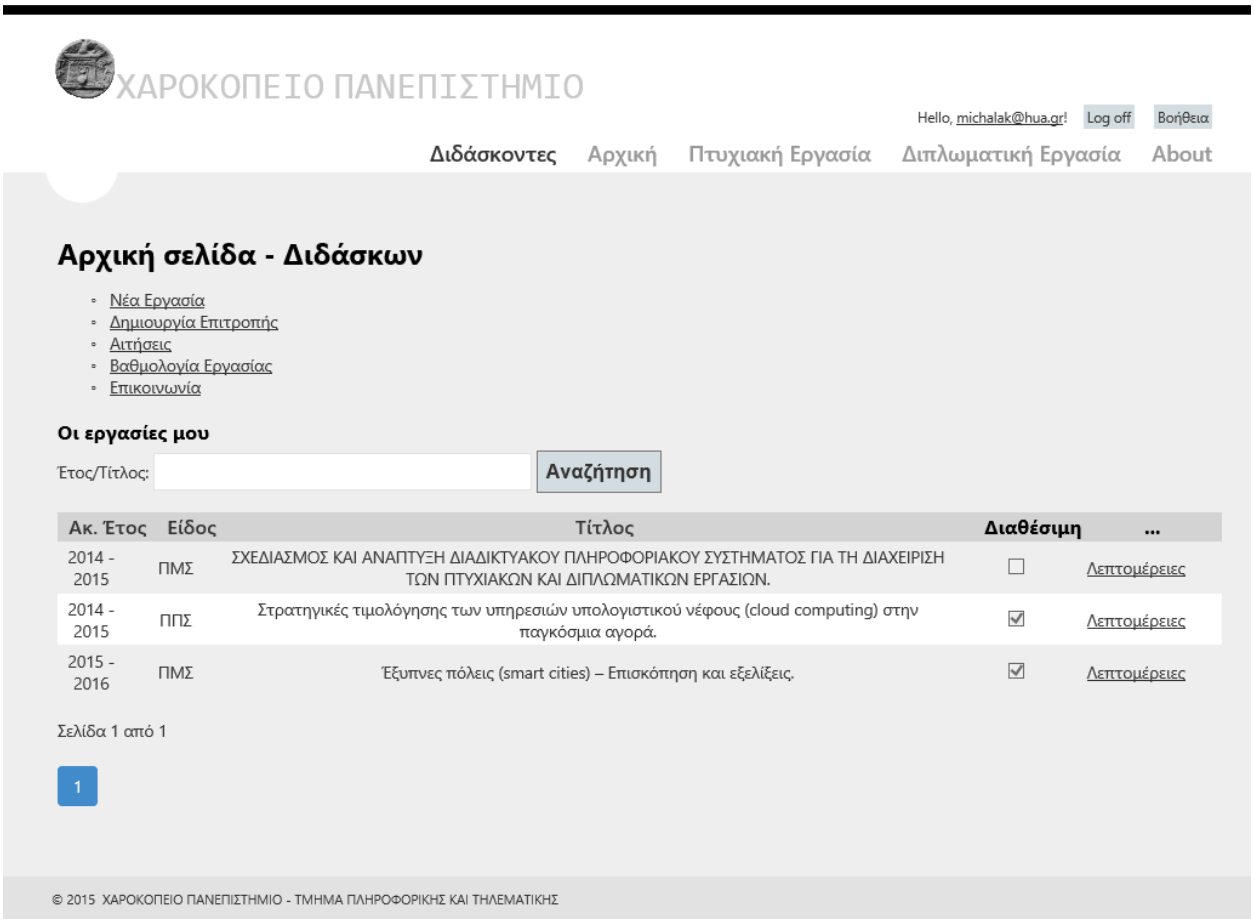


4.15 Αρχική Σελίδα – Συνδεδεμένος/η Διδάσκων (Index.aspx)

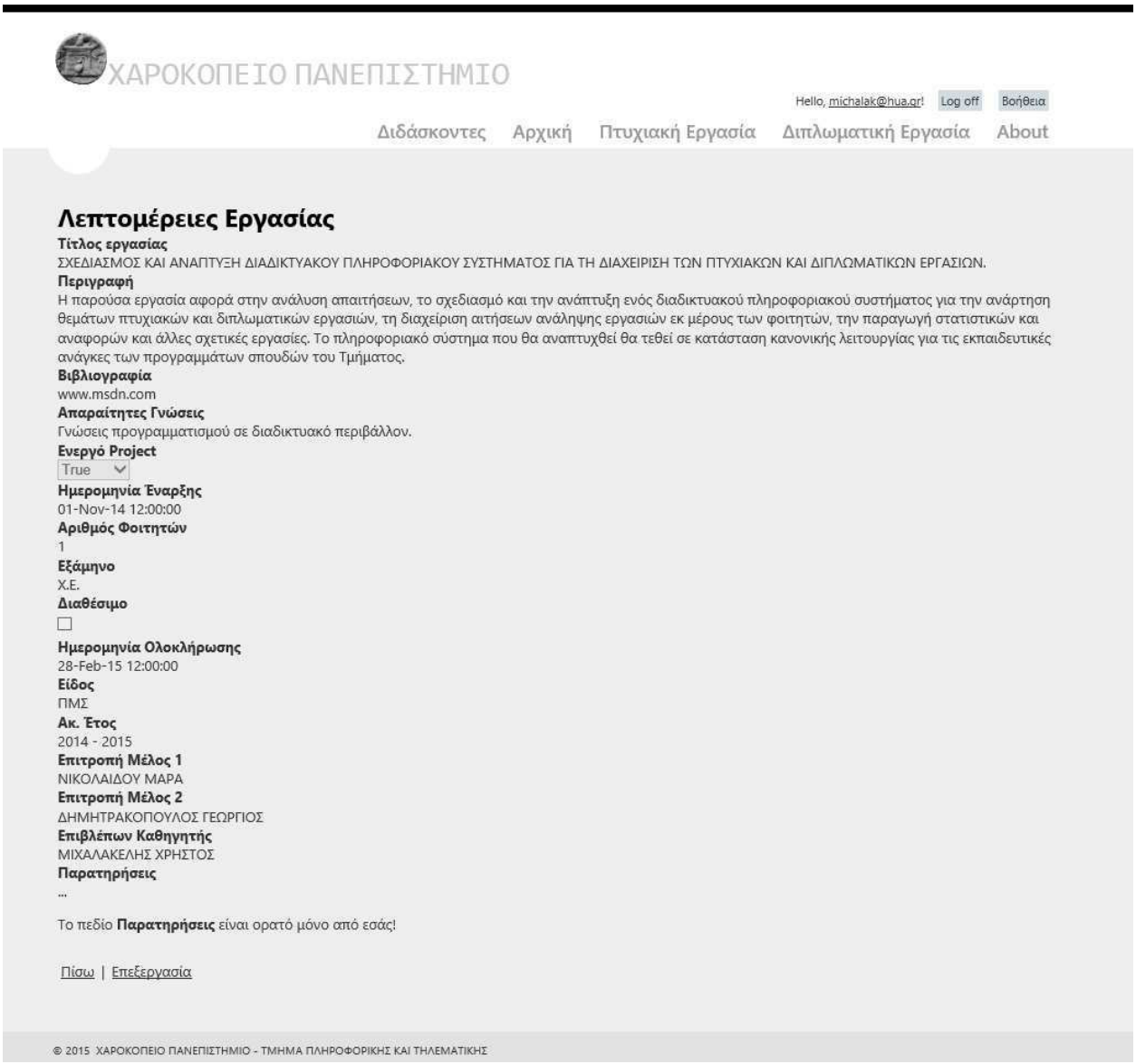
Η Αρχική προσωπική σελίδα των καθηγητών με νέο μενού που δίνει πρόσβαση στις επιμέρους διαδικασίες όπως η προβολή των εργασιών που έχει ανεβάσει, η καταχώρηση νέας εργασίας, η δημιουργία επιτροπής, η βαθμολόγηση εργασίας, η διαχείριση των αιτήσεων και η επικοινωνία.

4.16 Αρχική σελίδα Διδάσκων (Professor.aspx)

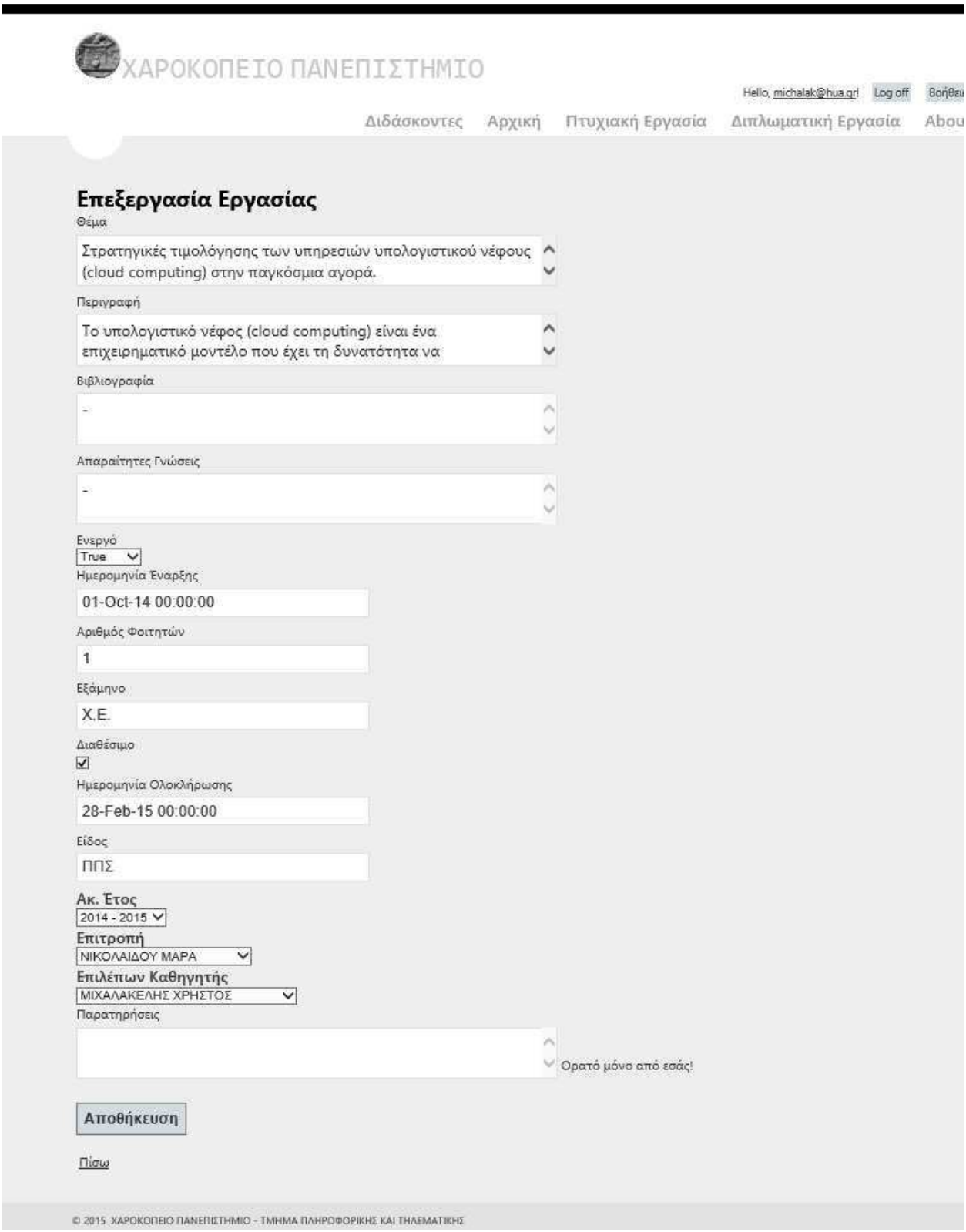
Η σελίδα αυτή εμφανίζεται πατώντας στο πρώτο κεντρικό μενού της εφαρμογής, εμφανίζει τις εργασίες που έχει ανεβάσει ο/η Διδάσκων με χρήσιμες πληροφορίες , τα μέλη της εξεταστικής επιτροπής που έχει προτείνει/εγκεκριμένα από το συμβούλιο και τυχόν προσωπικές παρατηρήσεις που έγιναν από τον διδάσκοντα. Παράλληλα, στο υπό-μενού περιέχει διάφορες λειτουργίες που αναλύθηκαν παραπάνω (4.15).



Παρέχονται, επίσης, περισσότερες λεπτομέρειες για την εκάστοτε επιλεγμένη εργασία και δίνεται η δυνατότητα επεξεργασίας των εργασιών από τους διδάσκοντες.

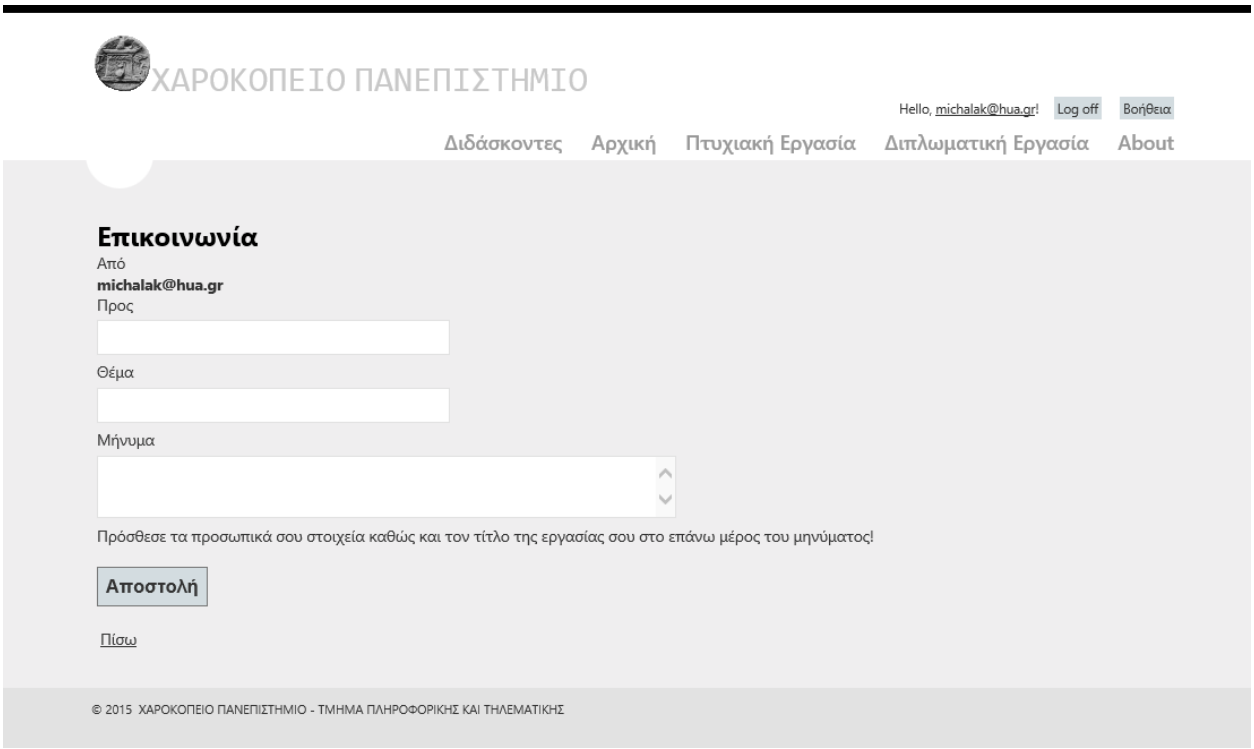


Επεξεργασία της επιλεγμένης εργασίας από τους διδάσκοντες.

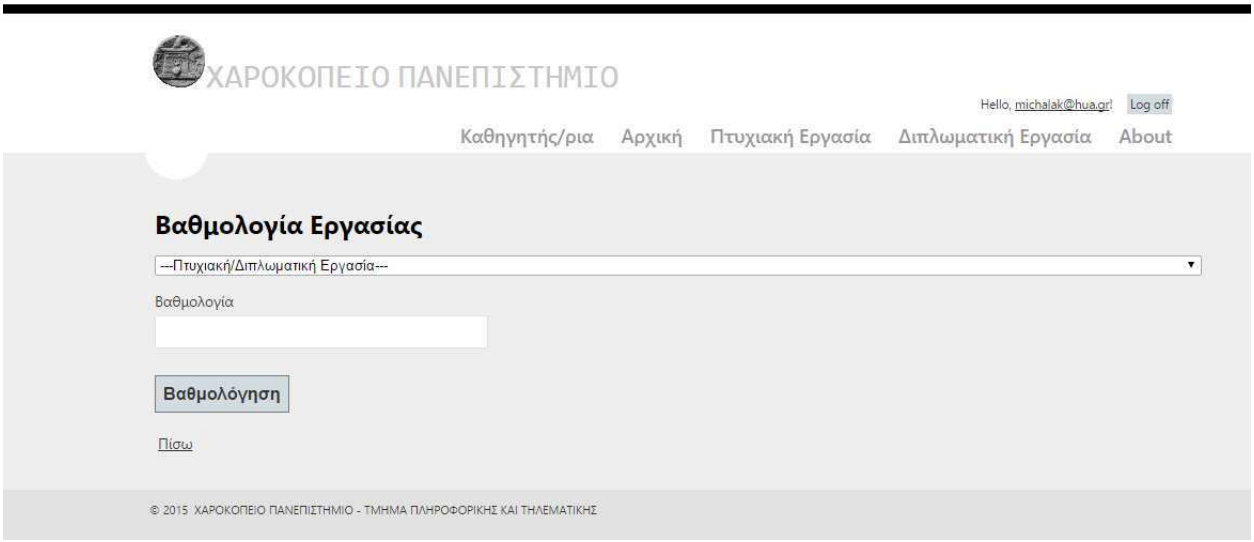


4.17 Επικοινωνία (Contact.aspx)

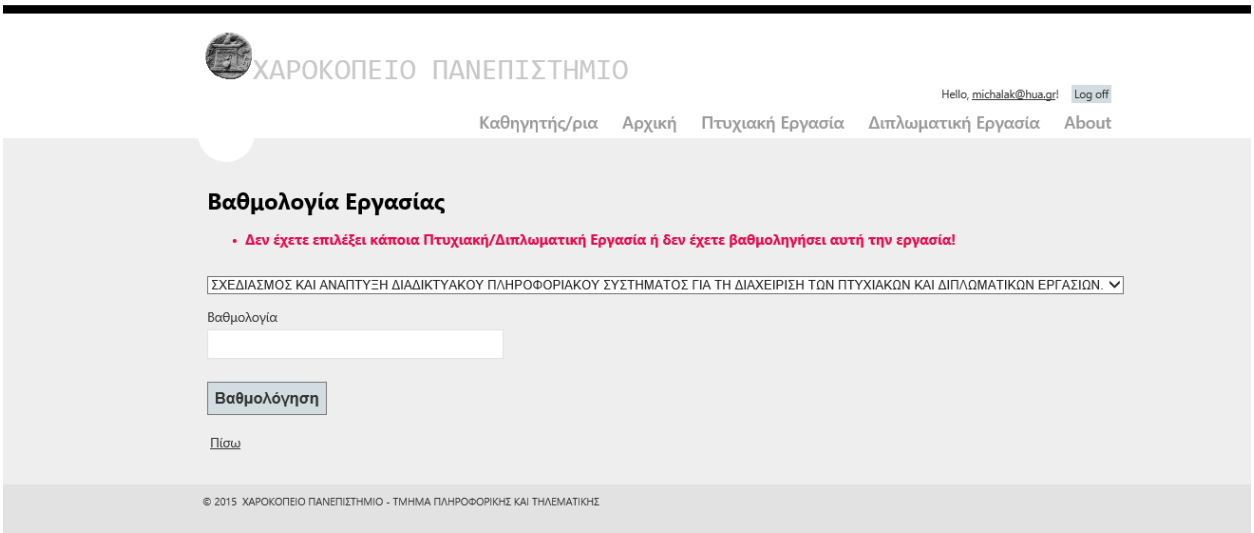
Όμοια με την σελίδα επικοινωνίας για τους φοιτητές.



4.18 Βαθμολογία Εργασίας (Grades.aspx)

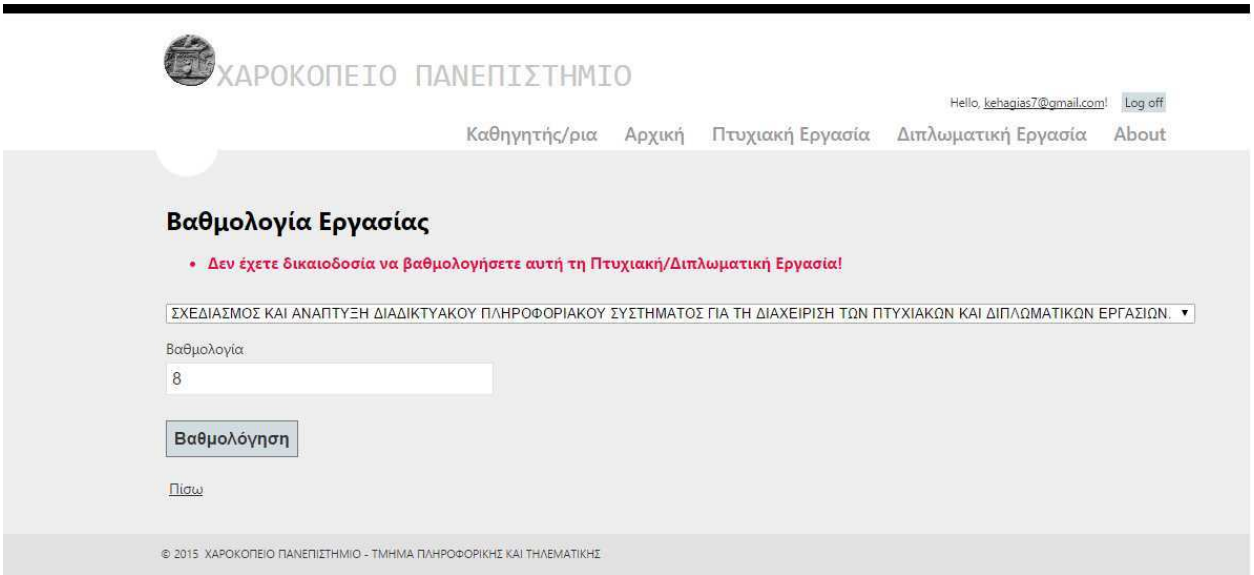


4.18.1 Βαθμολογία Εργασίας με κάποια παράληψη από τον/την καθηγητή/ρια (Grades.aspx)



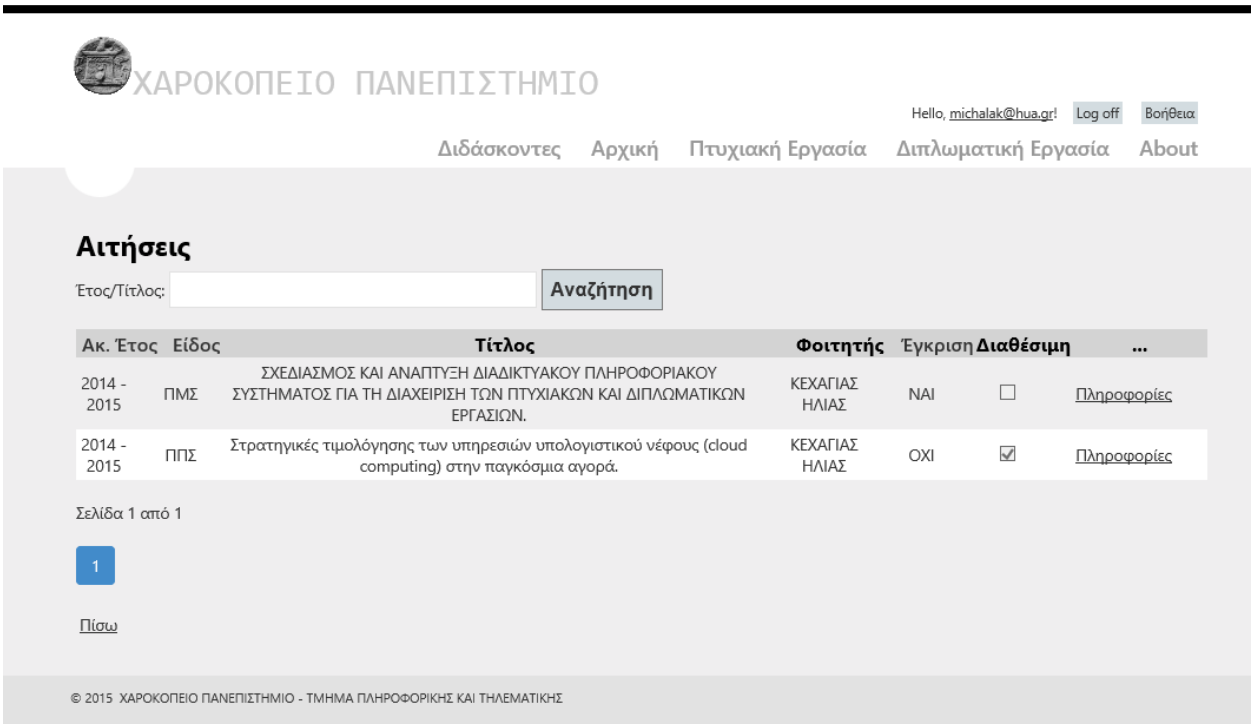
4.18.2 Βαθμολογία Εργασίας με προσπάθεια βαθμολόγησης εργασίας χωρίς δικαιοδοσία (Grades.aspx)

Ο/Η καθηγητής/ρια δεν ανήκει στην 3μελή εξεταστική επιτροπή και κατ' επέκταση δεν είναι ούτε ο/η Επιβλέπων Διδάσκων.

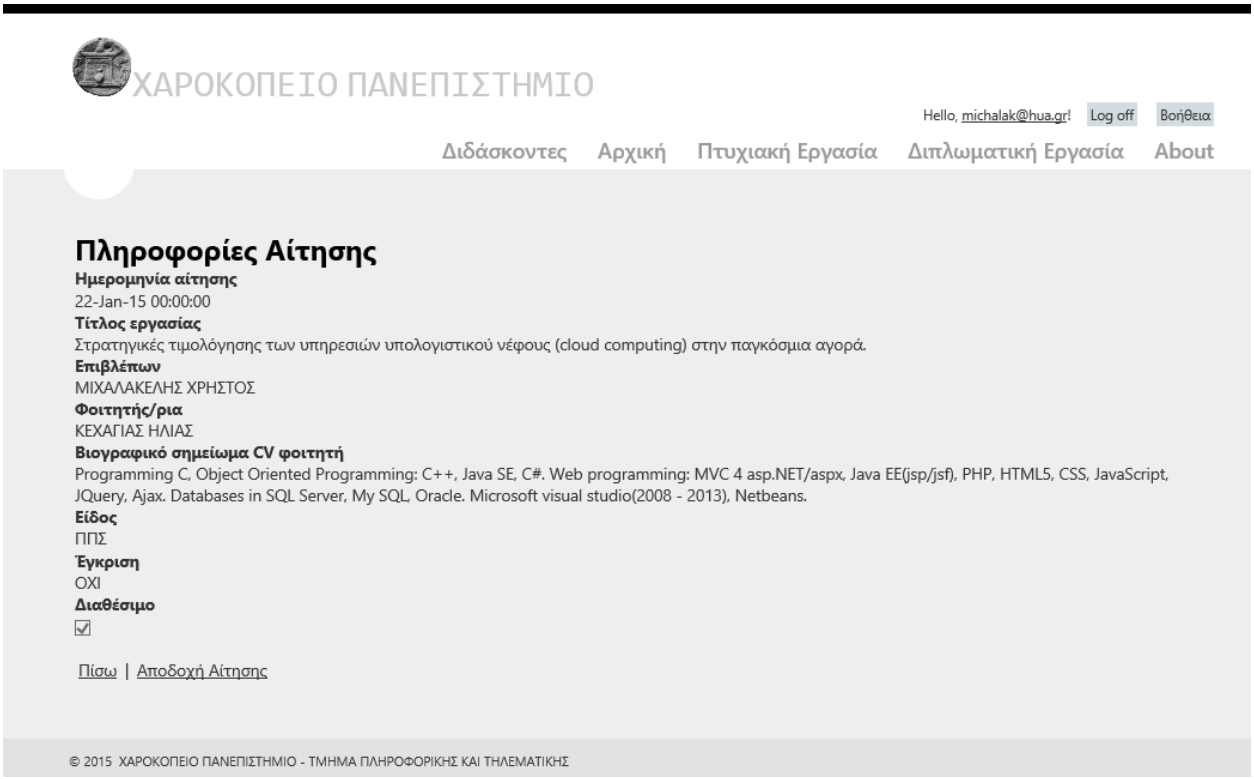


4.19 Αιτήσεις (Requests.aspx)

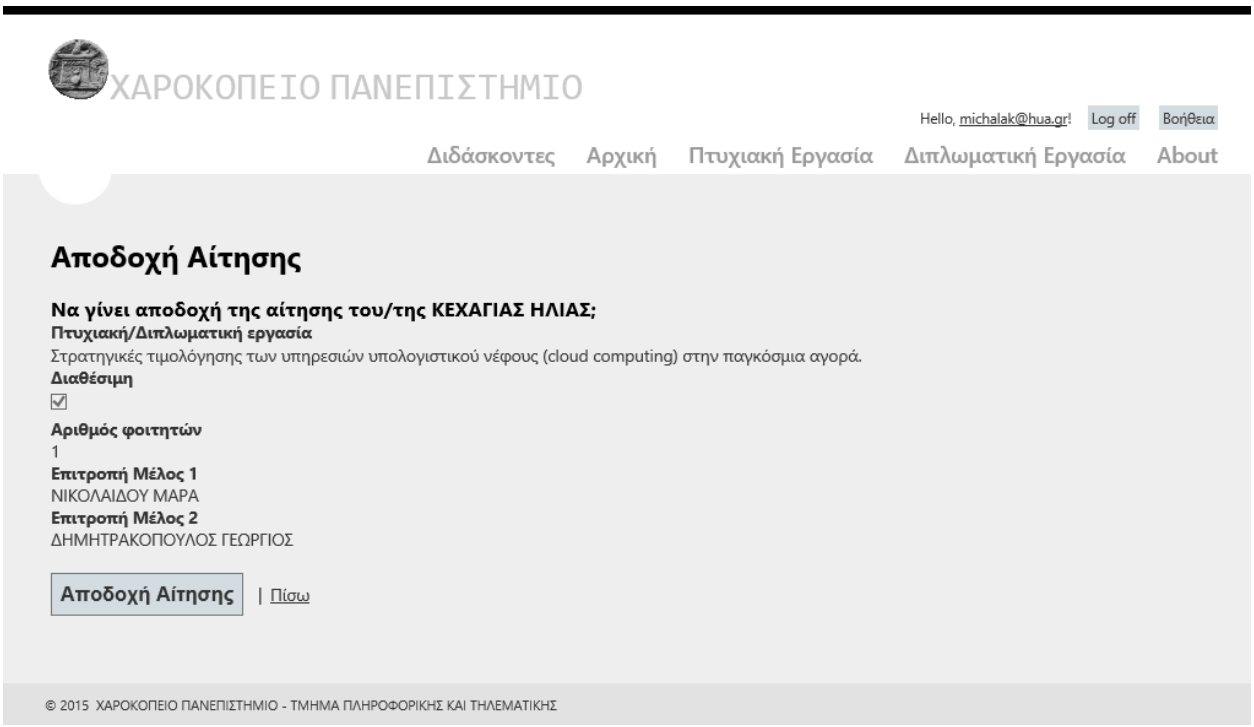
Στη σελίδα αιτήσεις παρουσιάζονται όλες οι αιτήσεις που έχουν υποβληθεί από τους φοιτητές προς τα ενεργά και διαθέσιμα project του διδάσκων με διαδικασία αποδοχής των αιτήσεων αυτών μέσω του link *Πληροφορίες*.



Δίνεται η δυνατότητα περισσότερων πληροφοριών που αφορά μια αίτηση με δυνατότητα αποδοχής της αίτησης αυτής.

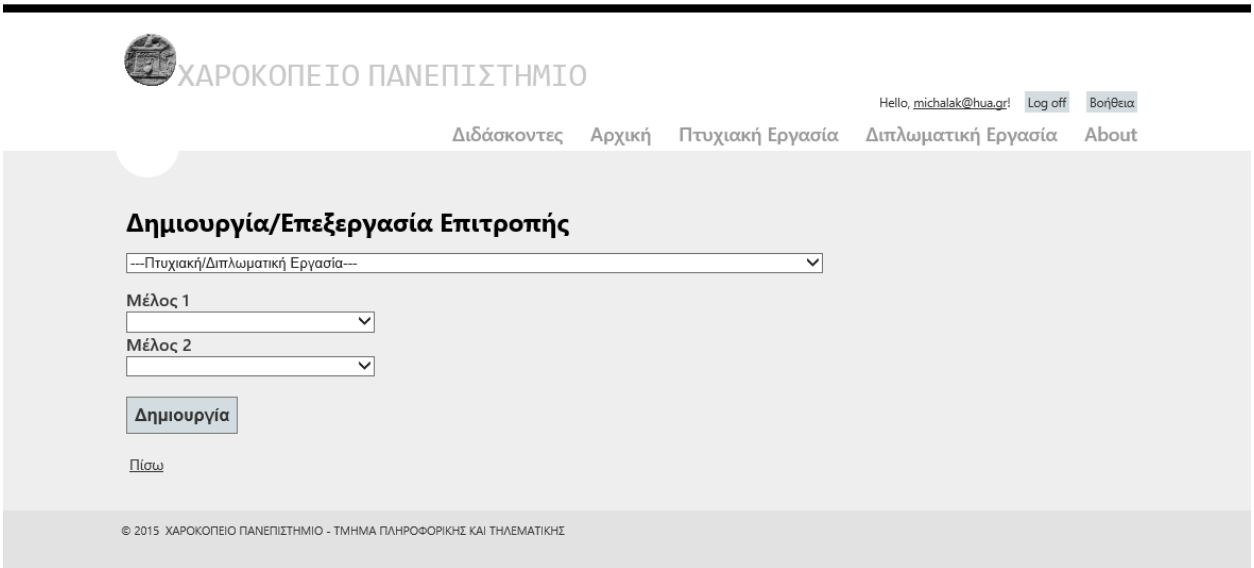


Κατά την αποδοχή της αίτησης στην ουσία αλλάζει η τιμή στο checkbox ώστε να μην είναι ορατή η εργασία σε άλλους φοιτητές προς αίτηση, εκτός αν η εργασία είναι ομαδική.



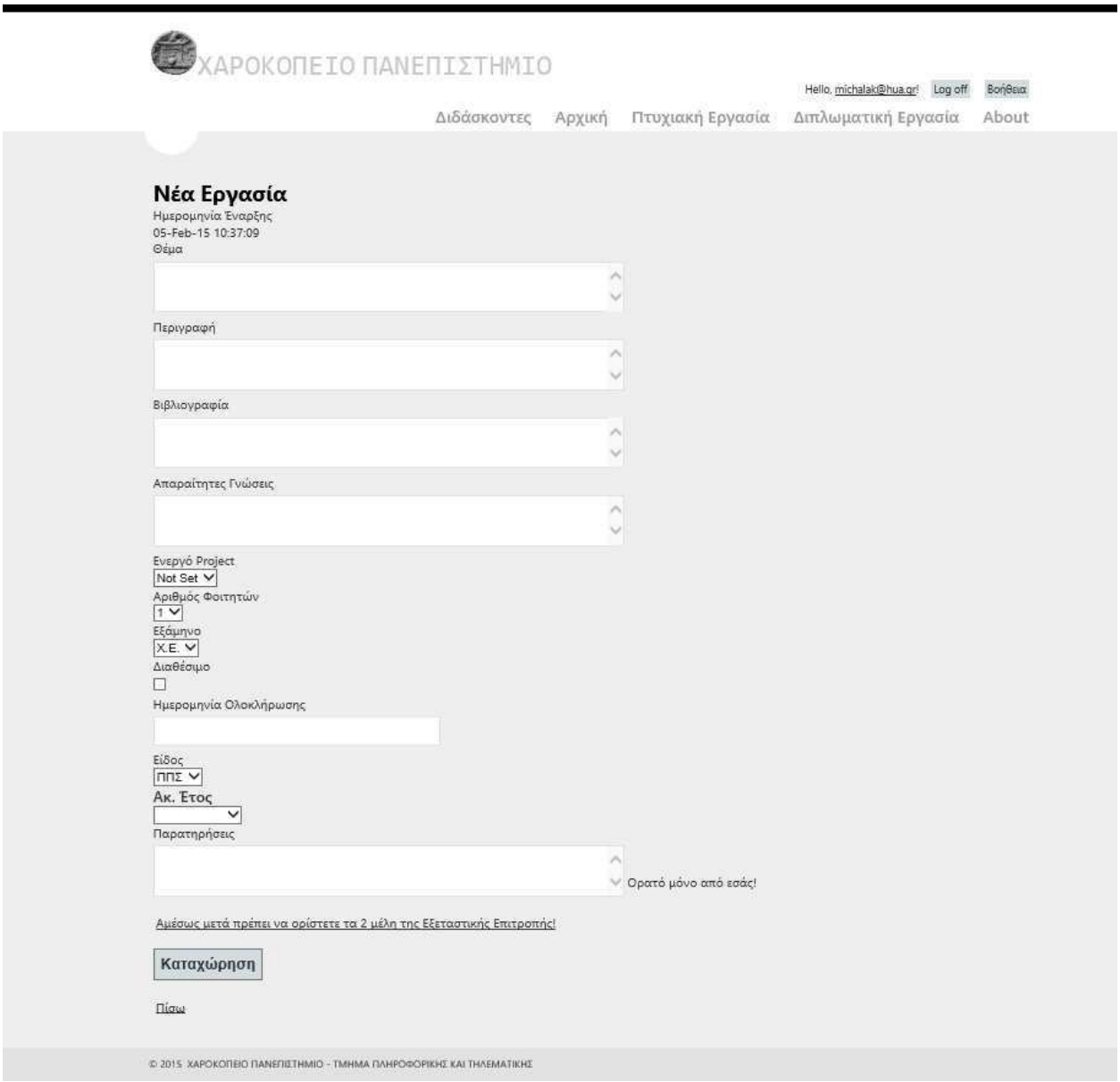
4.20 Δημιουργία Επιτροπής (Comittee/Create.aspx)

Από τη σελίδα αυτή ο/η επιβλέπων καθηγητής/ρια μπορεί να προτείνει 3μελή εξεταστική επιτροπή για κάθε εργασία που έχει ανεβάσει.



4.21 Νέα Εργασία (Project/Create.aspx)

Κάθε Διδάσκων από εδώ καταχωρεί και ταυτόχρονα προτείνει projects για πτυχιακές/διπλωματικές εργασίες. Όλα τα πεδία έχουν κάποιο ρόλο στην εφαρμογή, εκτός από τα πεδία *Απαραίτητες Γνώσεις* και *Παρατηρήσεις*, για παράδειγμα μόνο τα «ενεργά» και «διαθέσιμα» projects είναι διαθέσιμα προς νέα αίτηση ή γίνεται εμφάνιση και εκτύπωση από τη γραμματεία κατά *Είδος*, ανά *Ακ. Έτος* και ανά *Εξάμηνο*. **Προσοχή** με την συμπλήρωση των πεδίων! Το πεδίο *Παρατηρήσεις* είναι προσωπικό και δεν είναι ορατό από κανέναν χρήστη.



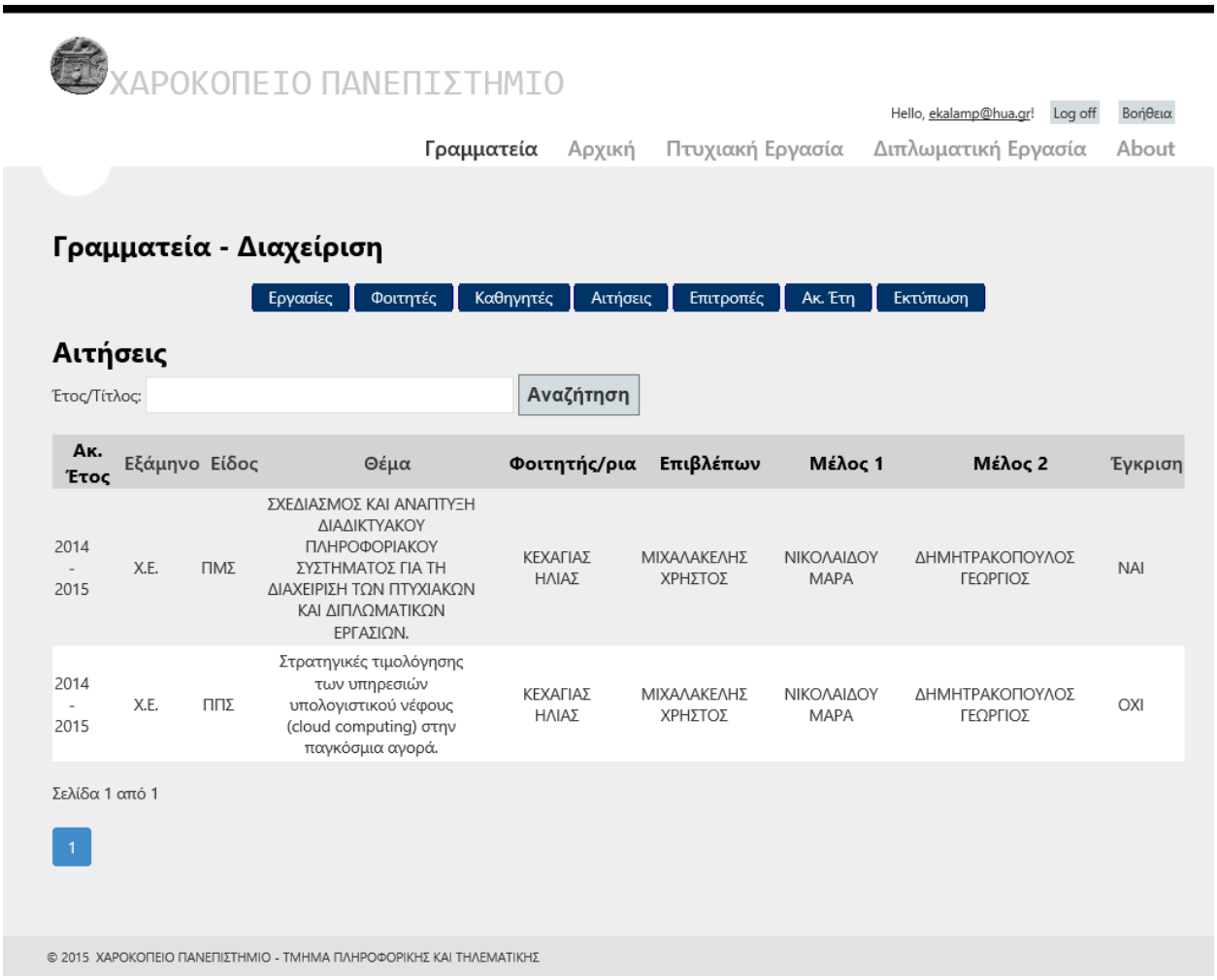
4.22 Αρχική Σελίδα – Συνδεδεμένη Γραμματεία (Index.aspx)

Η Αρχική προσωπική σελίδα της γραμματείας με νέο μενού που δίνει πρόσβαση στις επιμέρους διαδικασίες όπως η προβολή όλων των εργασιών, η διαχείριση των φοιτητών, των καθηγητών, των εργασιών, των εξεταστικών επιτροπών, των αιτήσεων, των ακ. ετών και η εκτύπωση των στοιχείων για το συμβούλιο των καθηγητών.

4.23 Γραμματεία - Διαχείριση (Secretary/Manage.aspx)

Η σελίδα αυτή εμφανίζεται πατώντας στο πρώτο κεντρικό μενού της εφαρμογής, εμφανίζει τις εργασίες που έχουν ανεβάσει οι καθηγητές/ριες με χρήσιμες πληροφορίες , τα μέλη της εξεταστικής επιτροπής που έχουν προτείνει/είναι εγκεκριμένα από το συμβούλιο ανάλογα με το Ακ. Έτος και το εξάμηνο που έχουν δηλωθεί από τους/τις φοιτητές/ριες. Παράλληλα, στο υπό-μενού περιέχει διάφορες λειτουργίες που αναλύθηκαν παραπάνω (4.22).

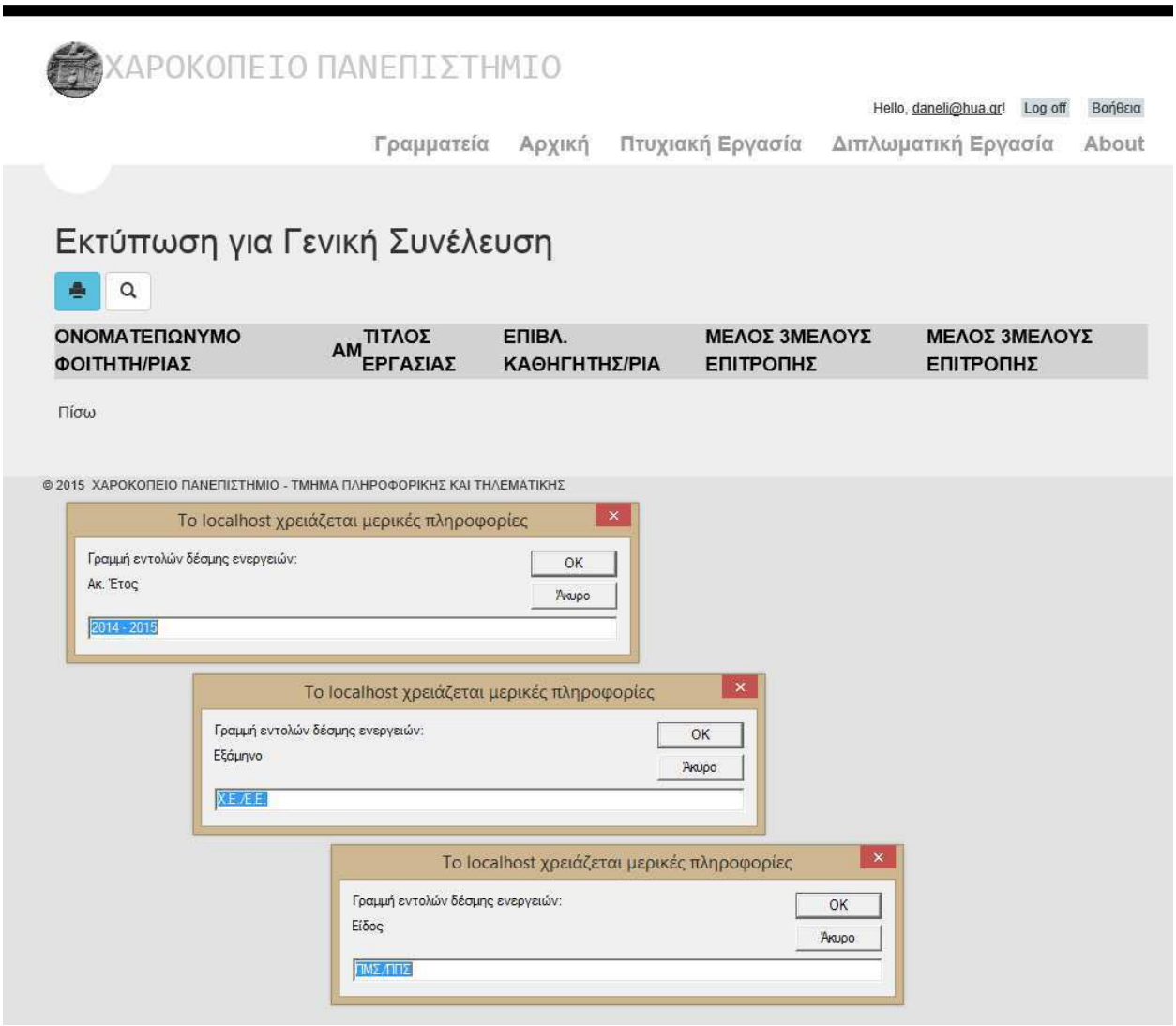
Με γνώμονα την καλύτερη δυνατή εξυπηρέτηση της γραμματείας παρέχεται ταξινόμηση ορισμένων πεδίων, όπου κρίνεται απαραίτητο, κάνοντας «κλικ» στον τίτλο της στήλης. Επίσης, υπάρχει διακριτός διαχωρισμός των γραμμών και των στηλών του πίνακα για εύκολη περιήγηση στα δεδομένα, πολλαπλή (περισσότερη του ενός πεδίου) διαδικασία αναζήτησης και σελιδοποίηση των αποτελεσμάτων (30 εγγραφές ανά σελίδα) με απλό και κατανοητό τρόπο στο κάτω μέρος της σελίδας με σκοπό την αποφυγή μεγάλου όγκου πληροφορίας στον χρήστη.



Ακ. Έτος	Εξάμηνο	Είδος	Θέμα	Φοιτητής/ρια	Επιβλέπων	Μέλος 1	Μέλος 2	Έγκριση
2014 - 2015	Χ.Ε.	ΠΜΣ	ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΠΤΥΧΙΑΚΩΝ ΚΑΙ ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ.	ΚΕΧΑΓΙΑΣ ΗΛΙΑΣ	ΜΙΧΑΛΑΚΕΛΗΣ ΧΡΗΣΤΟΣ	ΝΙΚΟΛΑΙΔΟΥ ΜΑΡΑ	ΔΗΜΗΤΡΑΚΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ	ΝΑΙ
2014 - 2015	Χ.Ε.	ΠΠΣ	Στρατηγικές τιμολόγησης των υπηρεσιών υπολογιστικού νέφους (cloud computing) στην παγκόσμια αγορά.	ΚΕΧΑΓΙΑΣ ΗΛΙΑΣ	ΜΙΧΑΛΑΚΕΛΗΣ ΧΡΗΣΤΟΣ	ΝΙΚΟΛΑΙΔΟΥ ΜΑΡΑ	ΔΗΜΗΤΡΑΚΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ	ΟΧΙ

4.24 Εκτύπωση για Γενική Συνέλευση (Print.aspx)

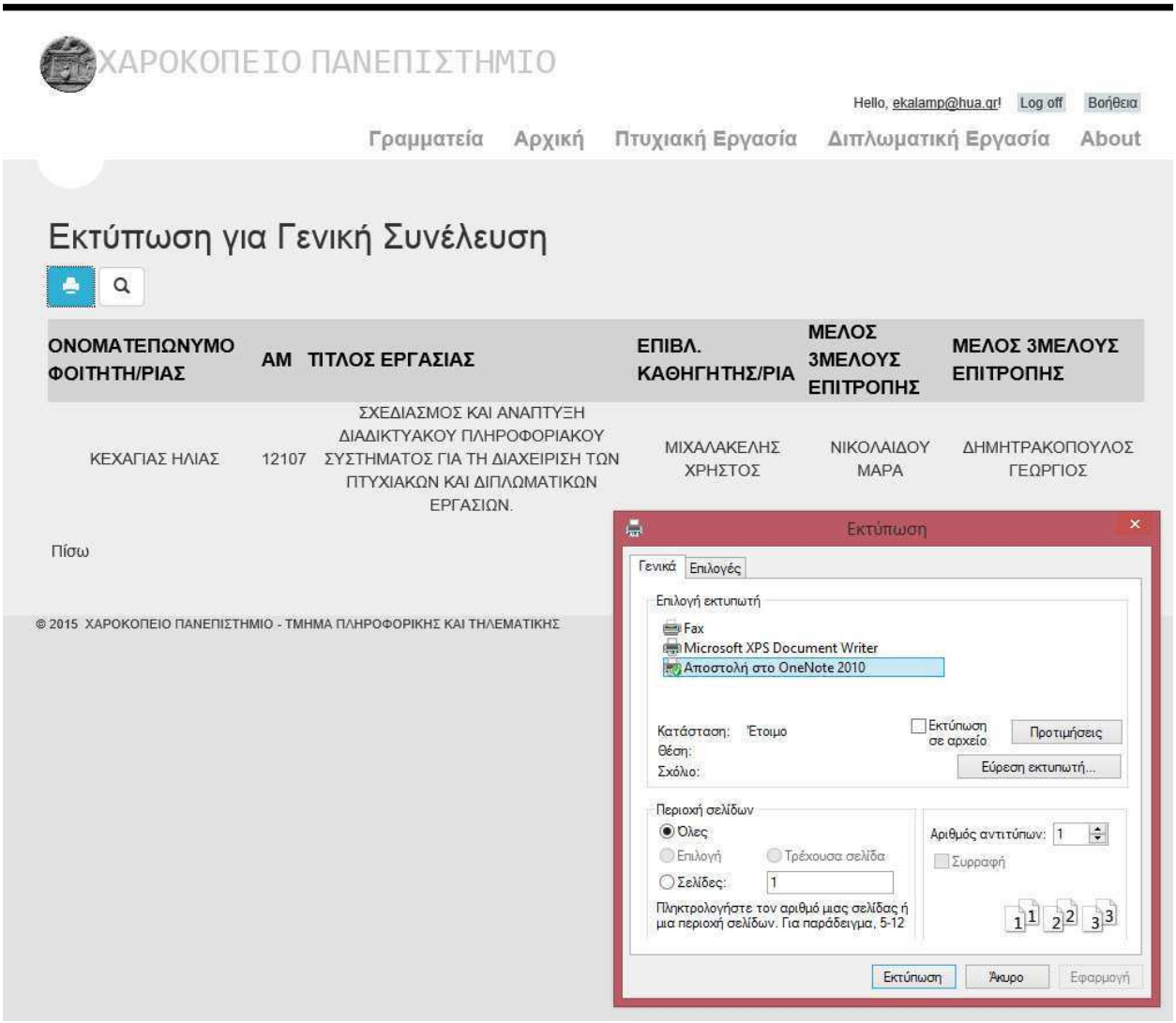
Από τη σελίδα αυτή η γραμματεία εξάγει/εκτυπώνει τα στοιχεία των πτυχιακών/διπλωματικών εργασιών για τη Γενική Συνέλευση του Τμήματος με σκοπό την έγκριση ή μη των αιτήσεων που αποδέχτηκαν οι καθηγητές και των εξεταστικών επιτροπών τους. Από το κουμπί με τον μεγεθυντικό φακό η γραμματεία του τμήματος θέτει φίλτρα/συνθήκες μέσα από «παράθυρα» διαλόγου του Browser για τα εμφανιζόμενα αποτελέσματα προς εκτύπωση, συγκεκριμένα εργασίες που οι καθηγητές έχουν αποδεχθεί τις αντίστοιχες αιτήσεις από τους φοιτητές αλλά δεν έχουν εγκριθεί. Όπως φαίνεται και παρακάτω στο στιγμιότυπο της σελίδας μπορούν να εμφανιστούν εργασίες ανάλογα με το Ακ. Έτος που έχουν καταχωρηθεί, με το Εξάμηνο διεξαγωγής/εξέτασής τους και με το Είδος των εργασιών (πτυχιακών/διπλωματικών) για το αντίστοιχο συμβούλιο των καθηγητών.



Αποτελέσματα προς εκτύπωση/αποθήκευση, από το κουμπί με τον εκτυπωτή πραγματοποιείται εκτύπωση των αποτελεσμάτων για τη Γενική συνέλευση του τμήματος ή αποθήκευση για το αρχείο του τμήματος αφού εισαχθούν πρώτα τα αντίστοιχα κριτήρια/φίλτρα.

4.24.1 Εκτύπωση για Γενική Συνέλευση (Print.aspx)

Εδώ δίνεται η δυνατότητα εκτύπωσης των αποτελεσμάτων πατώντας στο αντίστοιχο κουμπί ή η δυνατότητα αποθήκευσης του αρχείου ως XPS Document Writer για το Αρχείο του τμήματος.



4.24.2 Αποτέλεσμα Εκτύπωσης για Γενική Συνέλευση

Εκτύπωση Για Συμβούλιο Page 1 of 1

ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

Hello, daneli@hua.gr! Log off

Γραμματεία Αρχική Πτυχιακή Εργασία Διπλωματική Εργασία About

Εκτύπωση Για Συμβούλιο

Εκτύπωση σελίδας *

ΟΝΟΜΑΤΕΠΩΝΥΜΟ ΦΟΙΤΗΤΗ/ΡΙΑΣ	ΑΜ	ΤΙΤΛΟΣ ΕΡΓΑΣΙΑΣ	ΕΠΙΒΛ. ΚΑΘΗΓΗΤΗΣ/ΡΙΑ	ΜΕΛΟΣ 3ΜΕΛΟΥΣ ΕΠΙΤΡΟΠΗΣ	ΜΕΛΟΣ 3ΜΕΛΟΥΣ ΕΠΙΤΡΟΠΗΣ
ΚΕΧΑΓΙΑΣ ΗΛΙΑΣ	12107	ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΑΝΑΠΤΥΞΗ ΔΙΑΔΙΚΤΥΑΚΟΥ ΠΛΗΡΟΦΟΡΙΑΚΟΥ ΣΥΣΤΗΜΑΤΟΣ ΓΙΑ ΤΗ ΔΙΑΧΕΙΡΙΣΗ ΤΩΝ ΠΤΥΧΙΑΚΩΝ ΚΑΙ ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ.	ΜΙΧΑΛΑΚΕΛΗΣ ΧΡΗΣΤΟΣ	ΝΙΚΟΛΑΙΔΟΥ ΜΑΡΑ	ΔΗΜΗΤΡΑΚΟΠΟΥΛΟΣ ΓΕΩΡΓΙΟΣ

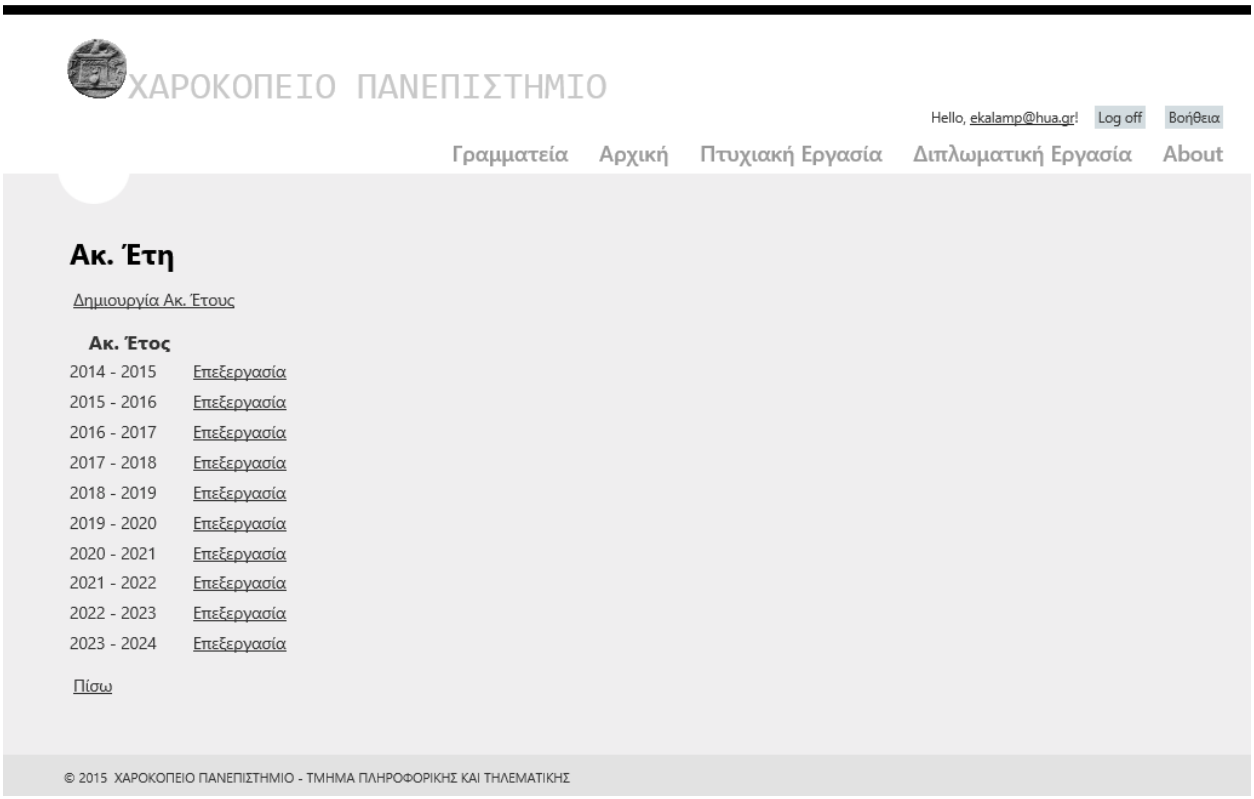
Πίσω

© 2015 ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ - ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

http://localhost:1146/Secretary/Print?aketos=2014 - 2015&examino=Χ.Ε.&eidos=ΠΜΣ 31-Jan-15

4.25 Ακ. Έτη (AkEtos/Index.aspx)

Πρόκειται για την αρχική σελίδα των Ακαδημαϊκών Ετών. Από εδώ προβάλλονται όλα τα ακ. Έτη με δυνατότητα δημιουργίας νέου ακολουθώντας το link στο επάνω μέρος της οθόνης, επεξεργασίας και διαγραφής των υπαρχόντων Ακ. Ετών καθώς και πλοήγησης στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



4.26 Δημιουργία Ακ. Έτους (AkEtos/Create.aspx)

Δεν είναι κάτι άλλο από μια απλή καταχώρηση έτους όταν εξαντληθεί η υπάρχουσα λίστα. Απαιτεί προσοχή η μορφή εισαγωγής διότι επηρεάζει την λειτουργία του συστήματος! Απαιτούμενη μορφή: «*2014 - 2015*».

4.27 Επεξεργασία Ακ. Έτους (AkEtos/Edit.aspx)

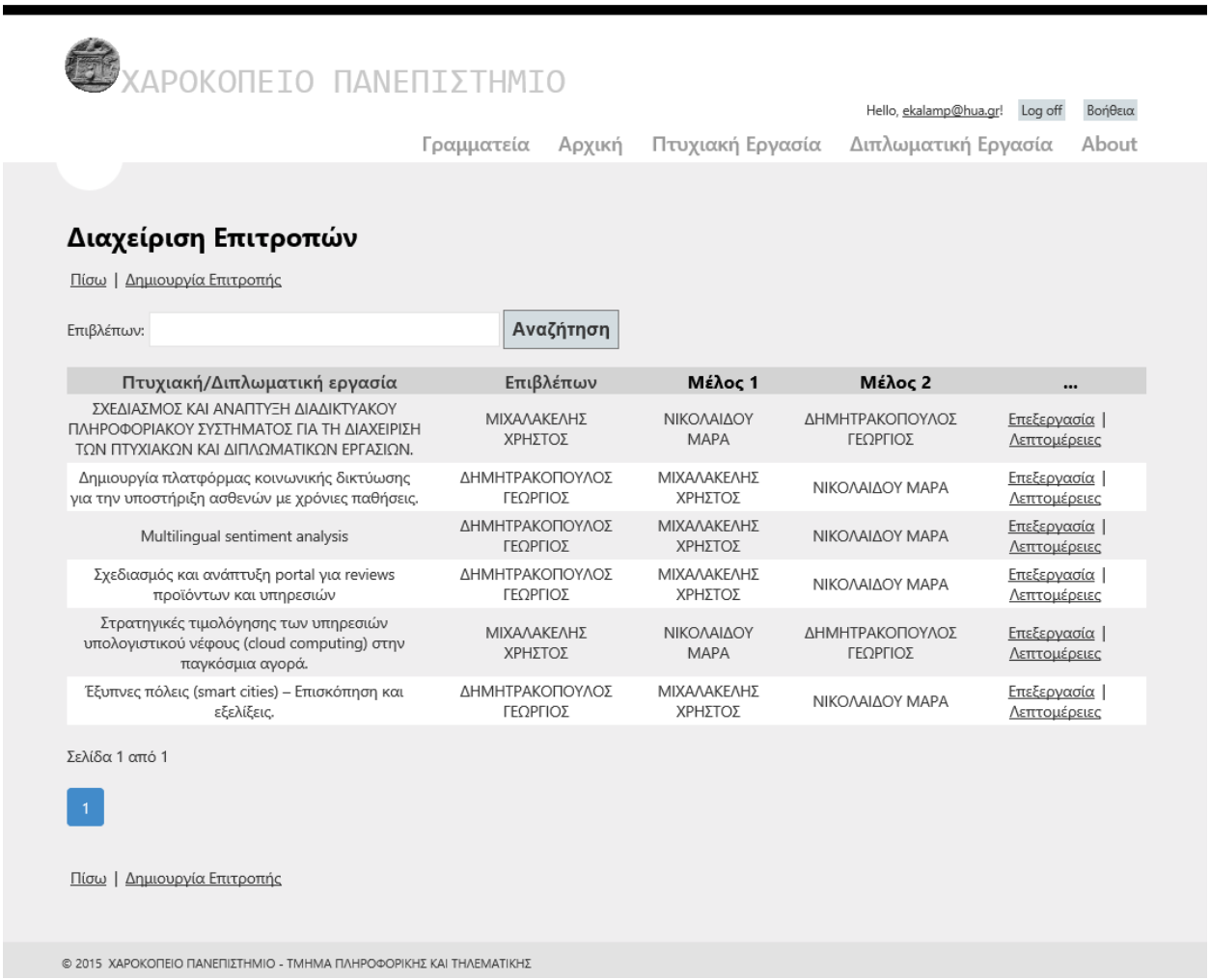
Αν κάτι δεν γίνει σωστά στην εισαγωγή νέου έτους η γραμματεία μπορεί να το επεξεργαστεί και να το διορθώσει.

4.28 Διαγραφή Ακ. Έτους

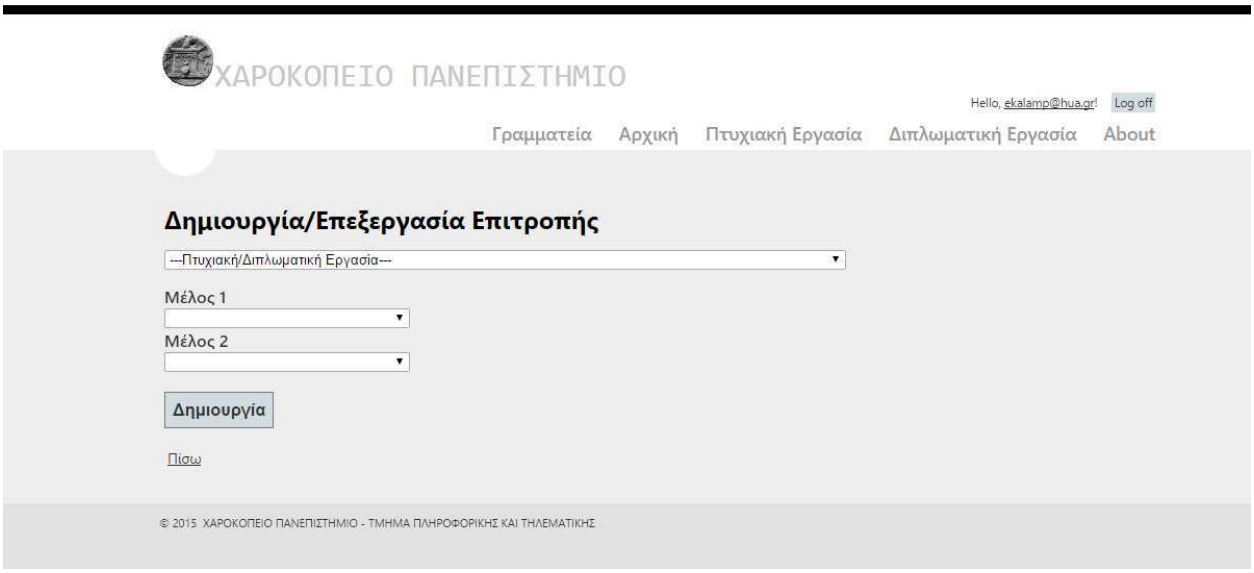
Δεν απαιτείται, διότι η γραμματεία έχει τη δυνατότητα επεξεργασίας τυχόν λάθος καταχώρησης.

4.29 Διαχείριση Επιτροπών (Comittee/Index.aspx)

Πρόκειται για την αρχική σελίδα των Εξεταστικών Επιτροπών. Από εδώ προβάλλονται όλες οι επιτροπές με δυνατότητα δημιουργίας νέας ακολουθώντας το link στο επάνω μέρος της οθόνης, επεξεργασίας, διαγραφής και προβολής λεπτομερειών για την επιλεγμένη επιτροπή καθώς και πλοήγησης στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.

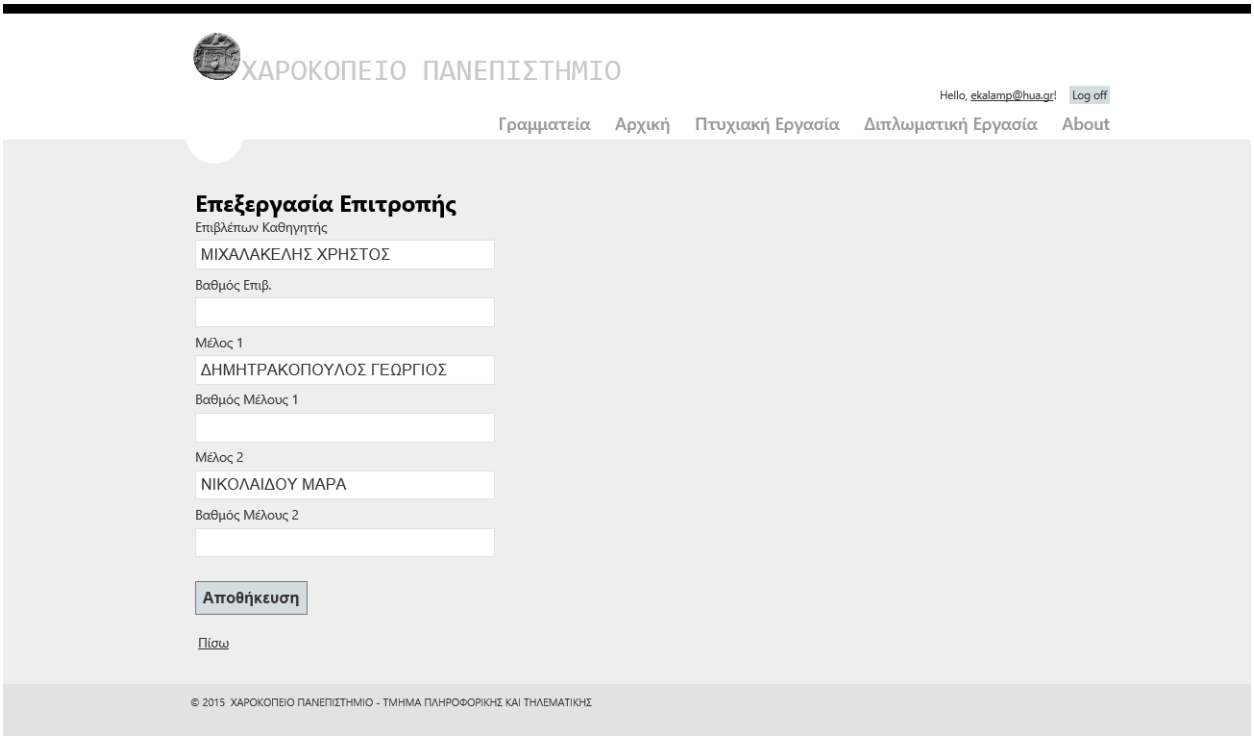


Η Δημιουργία επιτροπής ως σελίδα είναι όμοια με αυτή της δημιουργίας επιτροπής των Καθηγητών.



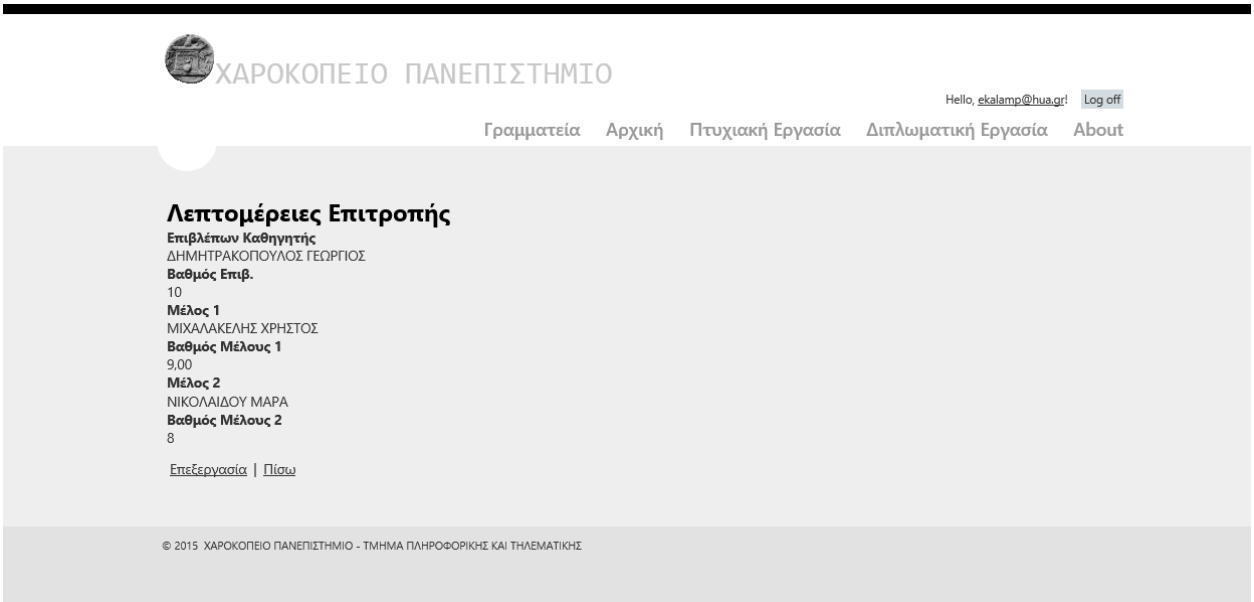
4.30 Επεξεργασία Επιτροπής (Comittee/Edit.aspx)

Αν κάτι δεν γίνει σωστά κατά την δημιουργία εξεταστικής επιτροπής η γραμματεία μπορεί να το επεξεργαστεί και να το διορθώσει.



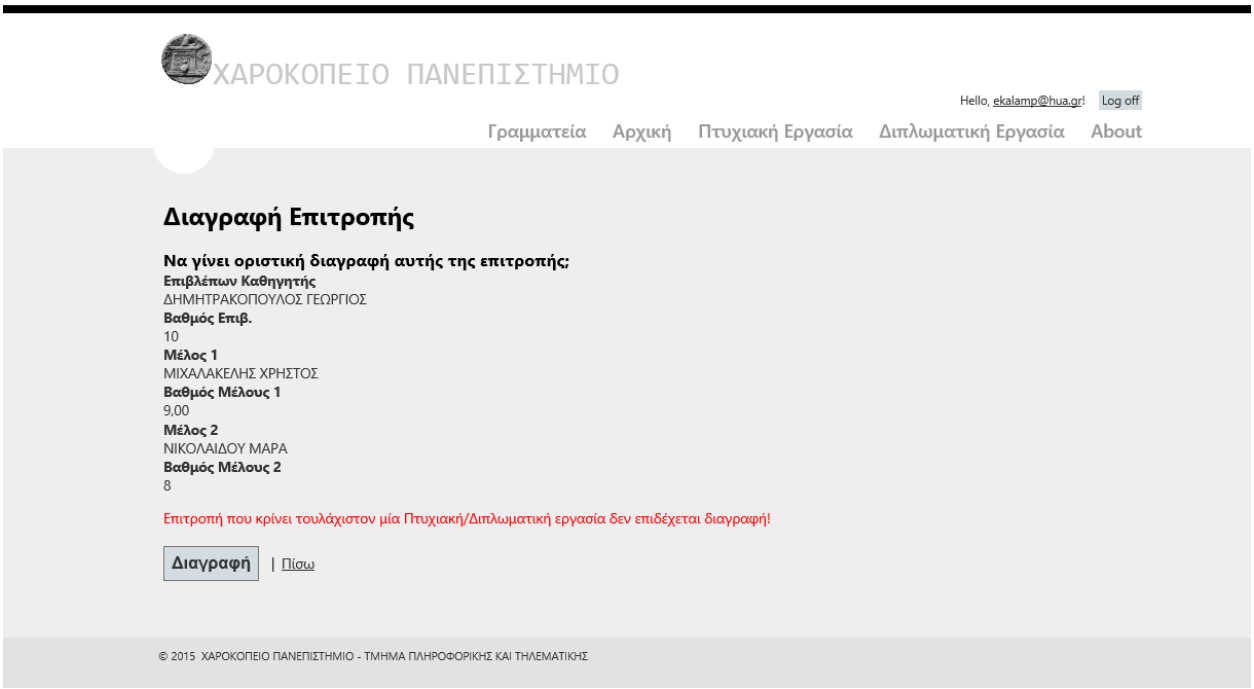
4.31 Λεπτομέρειες Επιτροπής (Comittee/Delete.aspx)

Στη σελίδα αυτή προβάλλονται με λεπτομέρειες όλα τα στοιχεία της επιλεγμένης επιτροπής. Με δυνατότητα επεξεργασίας και επιστροφής στην προηγούμενη σελίδα από τα links στο κάτω μέρος της οθόνης.



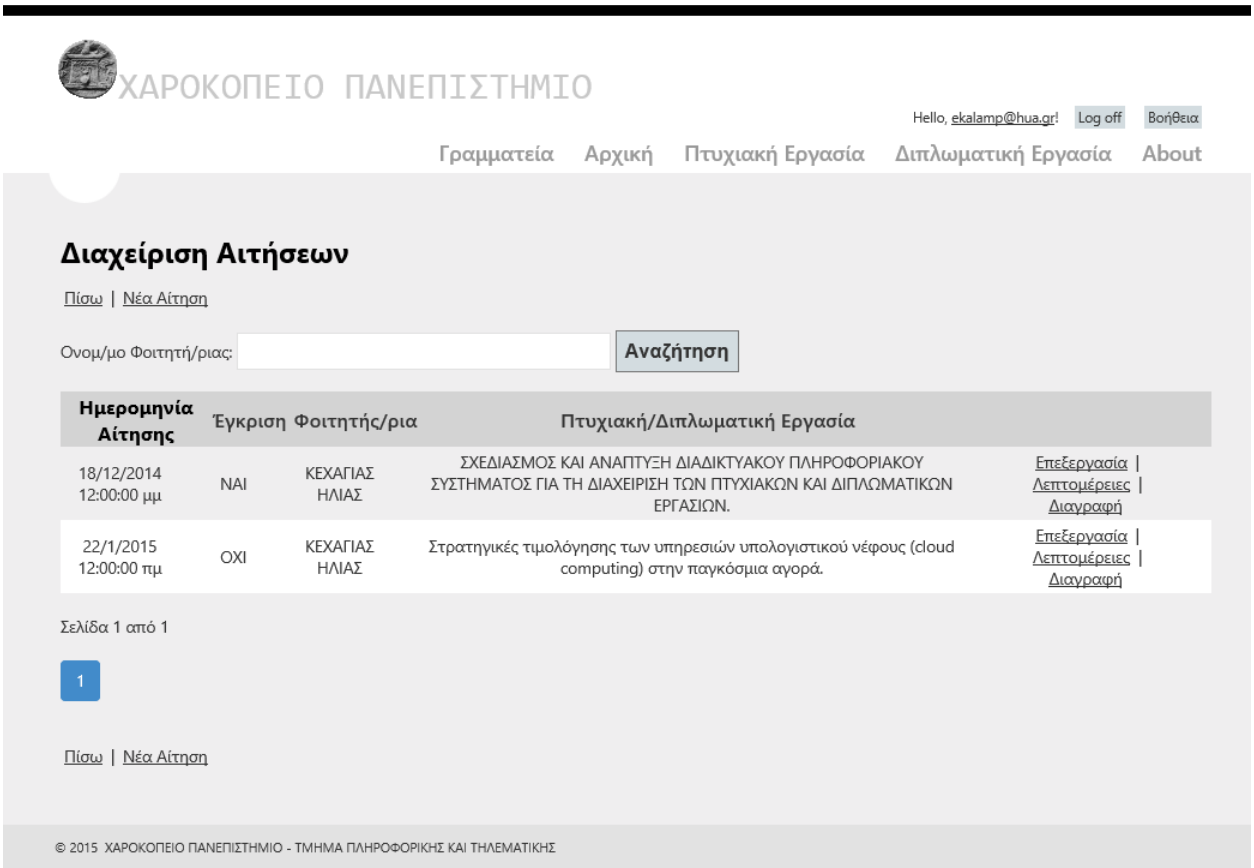
4.32 Διαγραφή Επιτροπής (Comittee/Delete.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να προβεί σε Διαγραφή της επιλεγμένης επιτροπής. Στην ουσία είναι η τελευταία προειδοποίηση από το σύστημα πριν γίνει η οριστική διαγραφή επιτροπής. Επιπλέον, εμφανίζει όλα τα στοιχεία αυτής της επιτροπής και ένα προειδοποιητικό μήνυμα διότι αν η γραμματεία προβεί σε διαγραφή επιτροπής που αφορά φοιτητή/εργασία/καθηγητή το σύστημα δεν θα το επιτρέψει! Η γραμματεία θα δει την σελίδα Error με τις κατάλληλες υποδείξεις. Τέλος, μπορείς να πλοηγηθείς στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



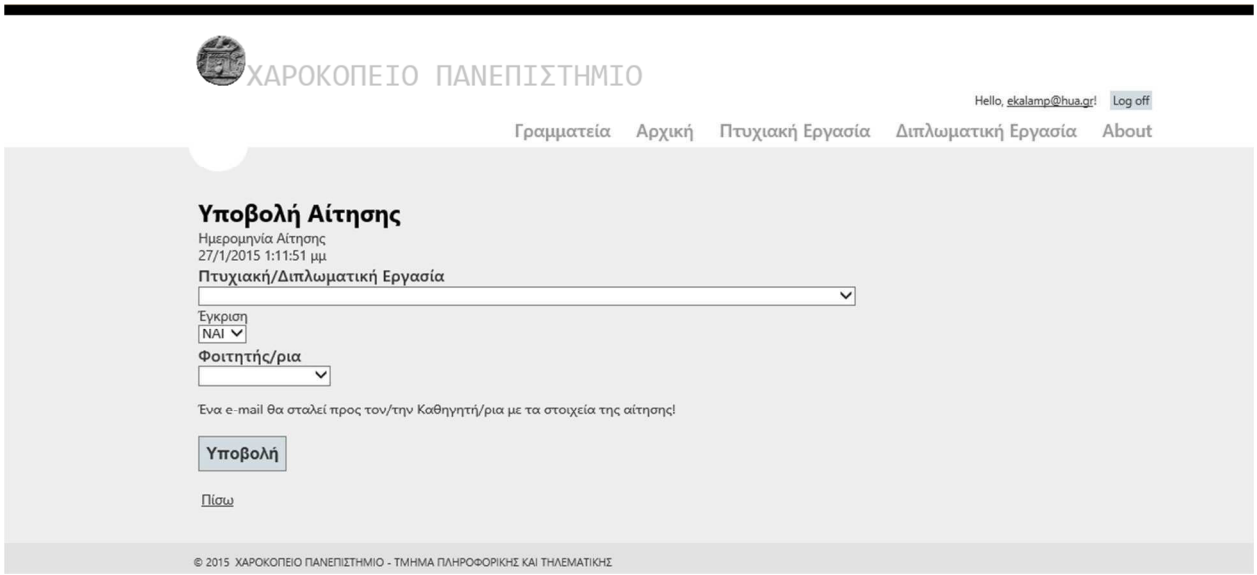
4.33 Διαχείριση Αιτήσεων (Request/Index.aspx)

Πρόκειται για την αρχική σελίδα των Αιτήσεων. Από εδώ προβάλλονται όλες οι αιτήσεις με δυνατότητα δημιουργίας νέας ακολουθώντας το link στο επάνω μέρος της οθόνης, επεξεργασίας, διαγραφής και προβολής λεπτομερειών για την επιλεγμένη αίτηση καθώς και πλοήγησης στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



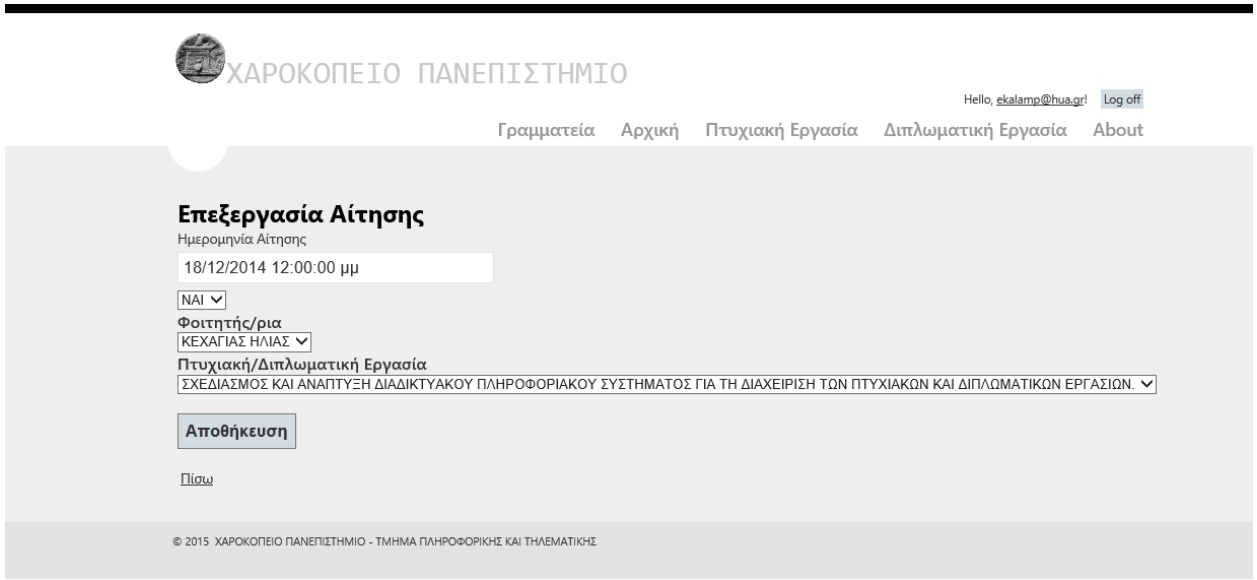
4.34 Υποβολή Αίτησης (Request/Create.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να υποβάλει αιτήσεις για εργασίες στο σύστημα εκ μέρους κάποιου φοιτητή πάντα. Επίσης, εμφανίζει πεδία εισαγωγής για όλα τα στοιχεία αυτής της αίτησης και ένα προειδοποιητικό μήνυμα διότι ένα e-mail θα σταλεί στον καθηγητή που αφορά η αίτηση με τα στοιχεία που τη συνοδεύουν με σκοπό να εισέλθει στην εφαρμογή και να αξιολογήσει την αίτηση! Προσοχή στην ορθή συμπλήρωση των πεδίων!



4.35 Επεξεργασία Αίτησης (Request/Edit.aspx)

Αν κάτι δεν γίνει σωστά κατά την υποβολή μιας αίτησης η γραμματεία μπορεί να την επεξεργαστεί και να τη διορθώσει. Παράλληλα, μπορεί να κάνει αίτηση εκ μέρους κάποιου/ας φοιτητή/ριας που δεν έχει δυνατότητα πρόσβασης στο σύστημα. Από εδώ η γραμματεία αλλάζει το πεδίο *Έγκριση* σε Ναι από Όχι αν η εργασία εγκριθεί από τη Γενική Συνέλευση.

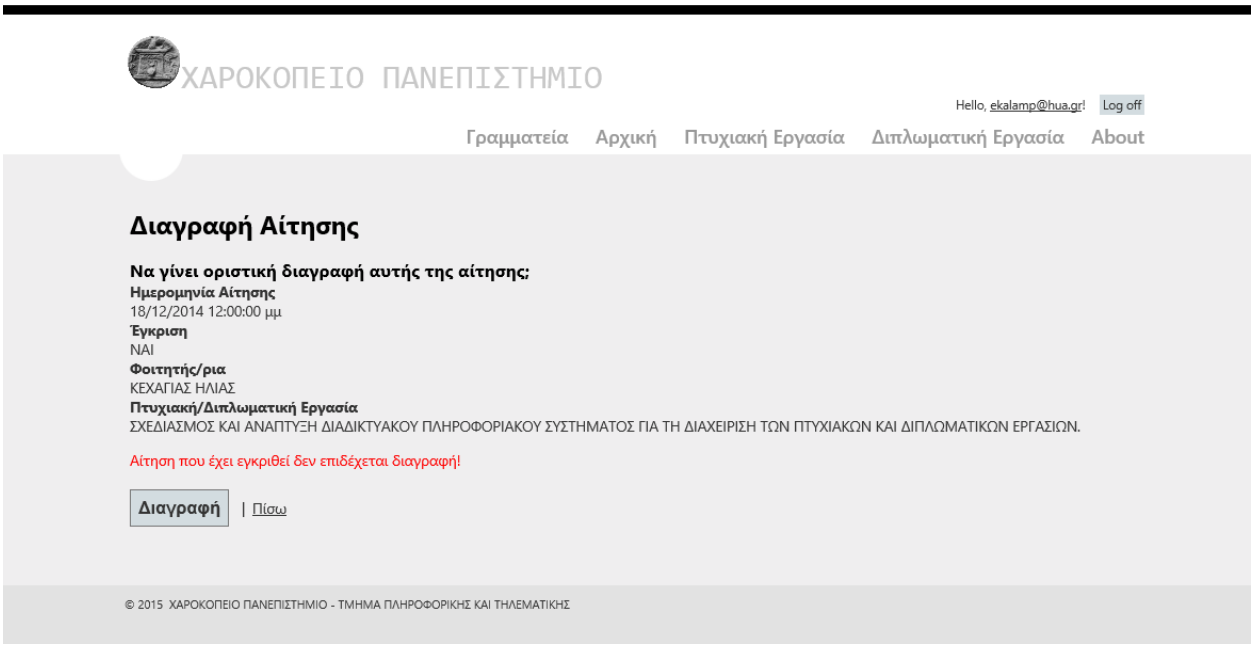


4.36 Λεπτομέρειες Αίτησης (Request/Details.aspx)

Στη σελίδα αυτή προβάλλονται με λεπτομέρειες όλα τα στοιχεία της επιλεγμένης αίτησης. Με δυνατότητα επεξεργασίας και επιστροφής στην προηγούμενη σελίδα από τα links στο κάτω μέρος της οθόνης.

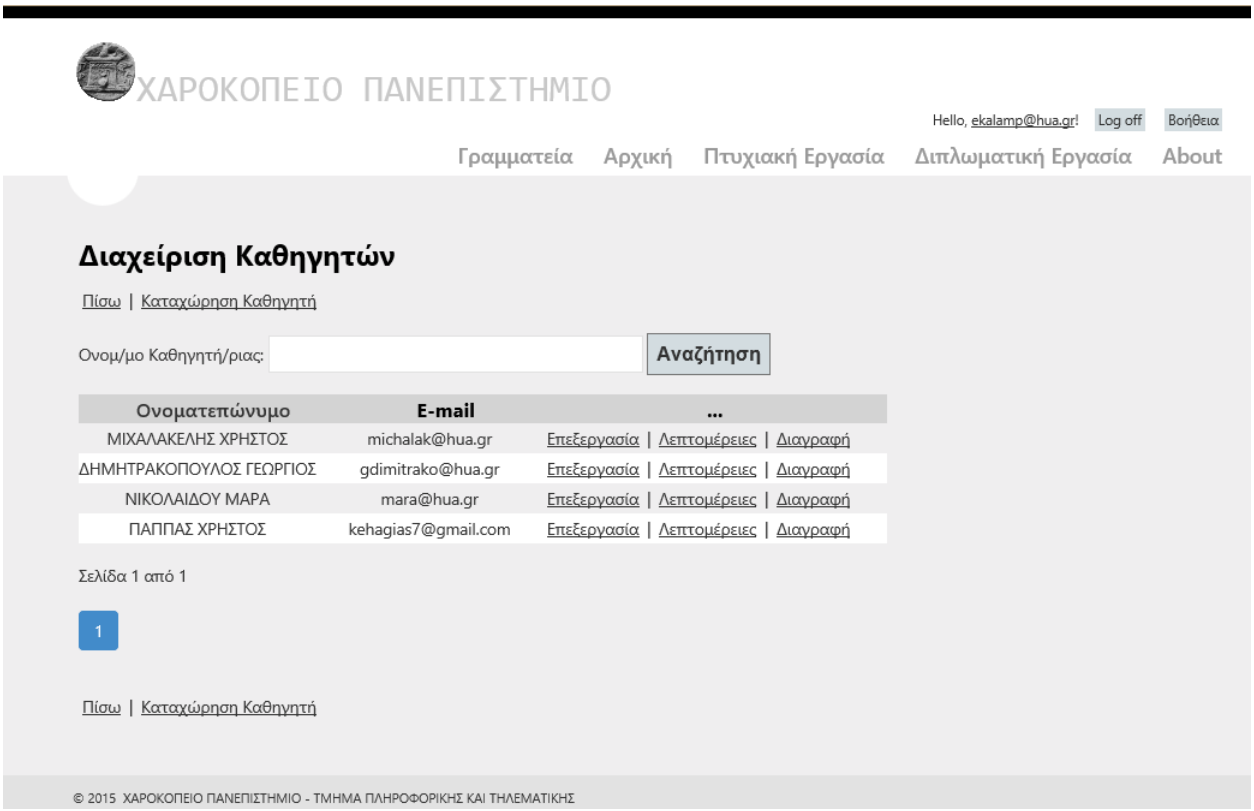
4.37 Διαγραφή Αίτησης (Request/Delete.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να προβεί σε Διαγραφή της επιλεγμένης αίτησης. Στην ουσία είναι η τελευταία προειδοποίηση από το σύστημα πριν γίνει η οριστική διαγραφή της αίτησης. Επιπλέον, εμφανίζει όλα τα στοιχεία αυτής της αίτησης και ένα προειδοποιητικό μήνυμα διότι αν η γραμματεία προβεί σε διαγραφή αίτησης που αφορά φοιτητή/εργασία το σύστημα δεν θα το επιτρέψει! Η γραμματεία θα δει την σελίδα Error με τις κατάλληλες υποδείξεις. Τέλος, μπορείς να πλοηγηθείς στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



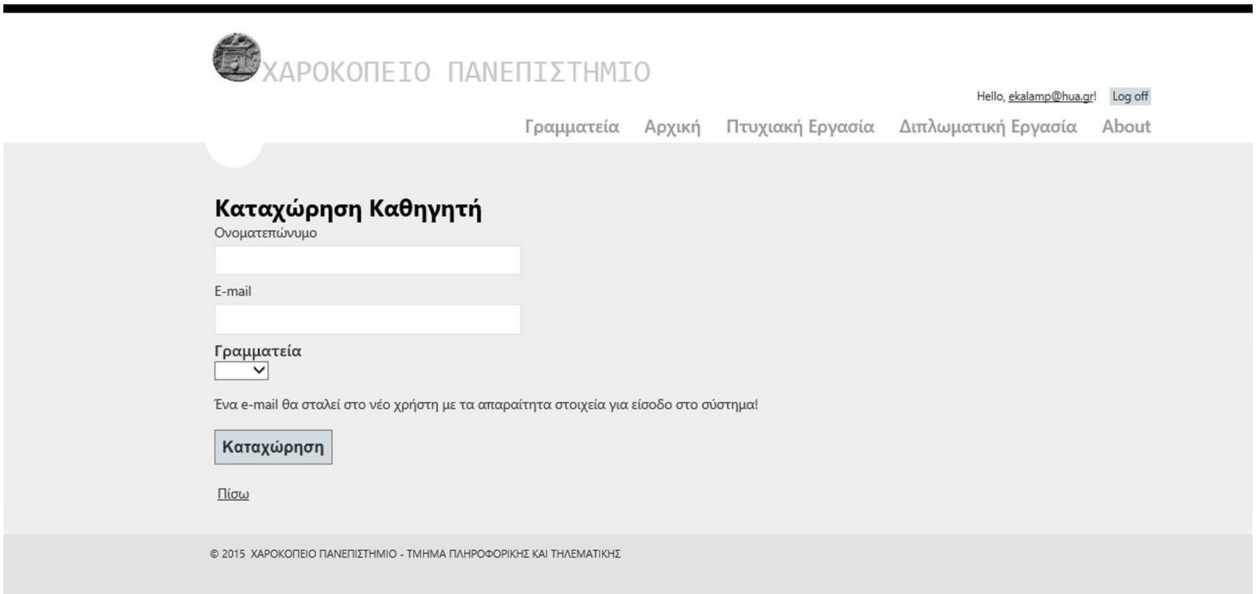
4.38 Διαχείριση Καθηγητών (Professor/Index.aspx)

Πρόκειται για την αρχική σελίδα των Καθηγητών με σκοπό την διαχείρισή τους από την γραμματεία του τμήματος. Κάθε διδάσκων ανήκει στον αντίστοιχο ρόλο και έχει δικό του λογαριασμό. Από εδώ προβάλλονται όλοι οι καθηγητές του τμήματος με δυνατότητα δημιουργίας νέου/ας ακολουθώντας το link στο επάνω μέρος της οθόνης, επεξεργασίας, διαγραφής και προβολής λεπτομερειών για τον/την επιλεγμένο/η καθηγητή/α καθώς και πλοήγησης στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



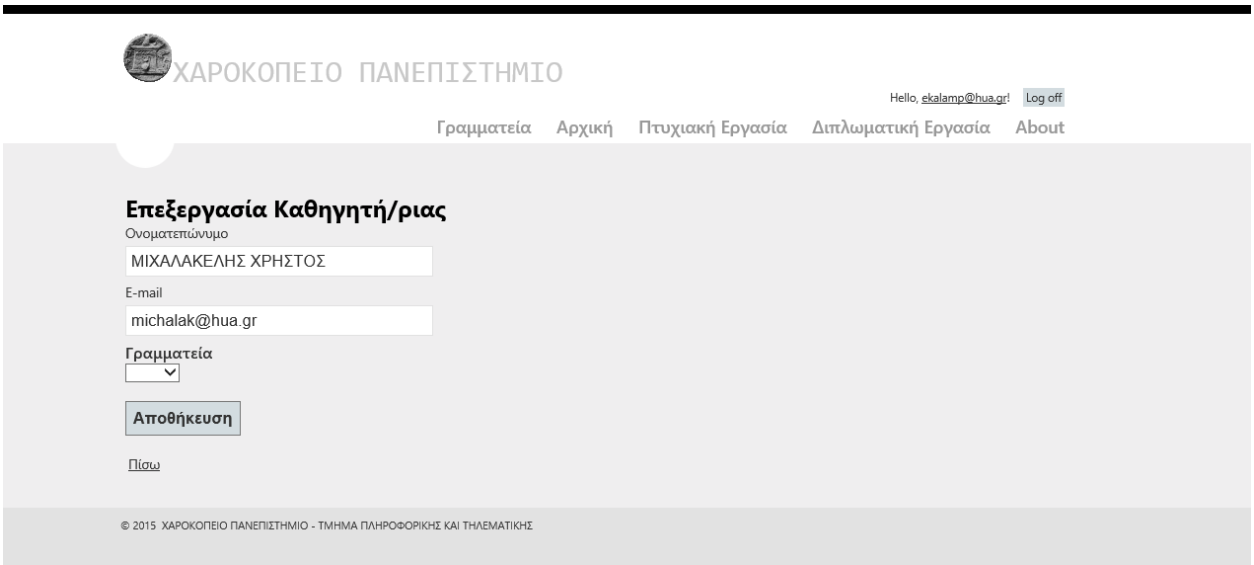
4.39 Καταχώρηση Καθηγητή (Professor/Create.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να καταχωρήσει νέους καθηγητές/ριες στο σύστημα. Στην ουσία σημαίνει και δημιουργία του αντίστοιχου λογαριασμού! Επιπλέον, εμφανίζει πεδία εισαγωγής για όλα τα στοιχεία αυτού/ης του/της καθηγητή/ριας και ένα προειδοποιητικό μήνυμα διότι ένα e-mail θα σταλεί στον καθηγητή που καταχωρήθηκε με τα στοιχεία του για την εισαγωγή του στο σύστημα! Για να φαίνεται στο σύστημα ποιος από τη γραμματεία έκανε την αλλαγή καλό θα ήταν να συμπληρώνεται το αντίστοιχο προαιρετικό πεδίο, χωρίς να έχει κρίσιμη σημασία στην συγκεκριμένη εφαρμογή. **Προσοχή** στην ορθή συμπλήρωση των πεδίων!



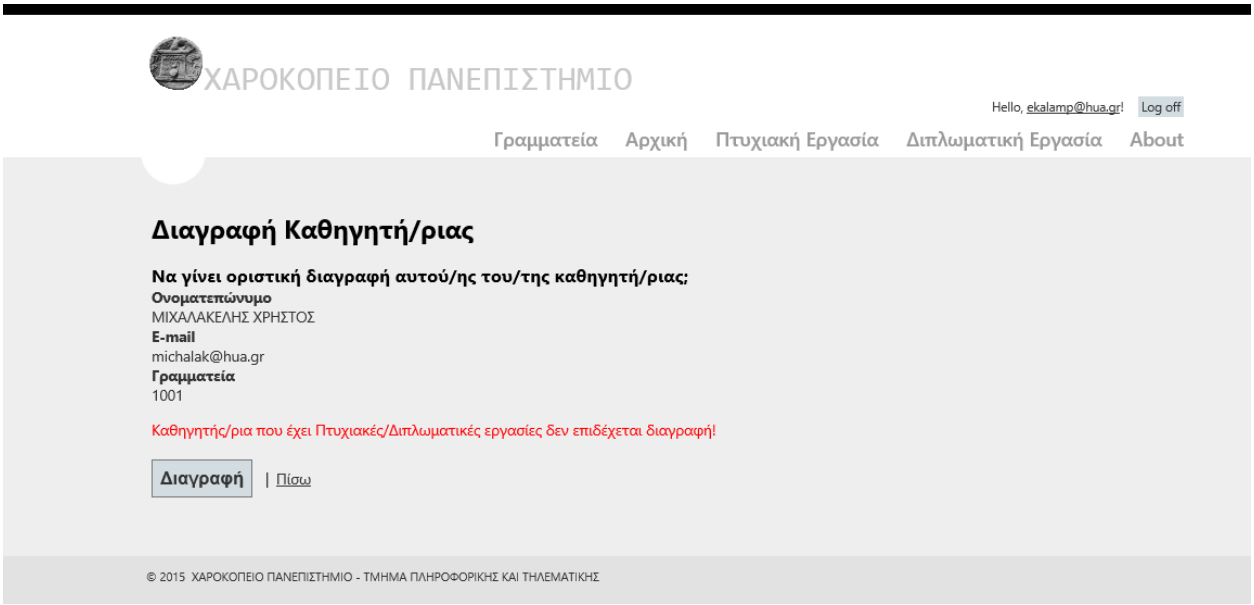
4.40 Επεξεργασία Καθηγητή/ριας (Professor/Edit.aspx)

Αν κάτι δεν γίνει σωστά κατά την καταχώρηση των καθηγητών η γραμματεία έχει τη δυνατότητα να προβεί σε επεξεργασία για διόρθωση. Για να φαίνεται στο σύστημα ποιος από τη γραμματεία έκανε την αλλαγή καλό θα ήταν να συμπληρώνεται το αντίστοιχο προαιρετικό πεδίο, χωρίς να έχει κρίσιμη σημασία στην συγκεκριμένη εφαρμογή.



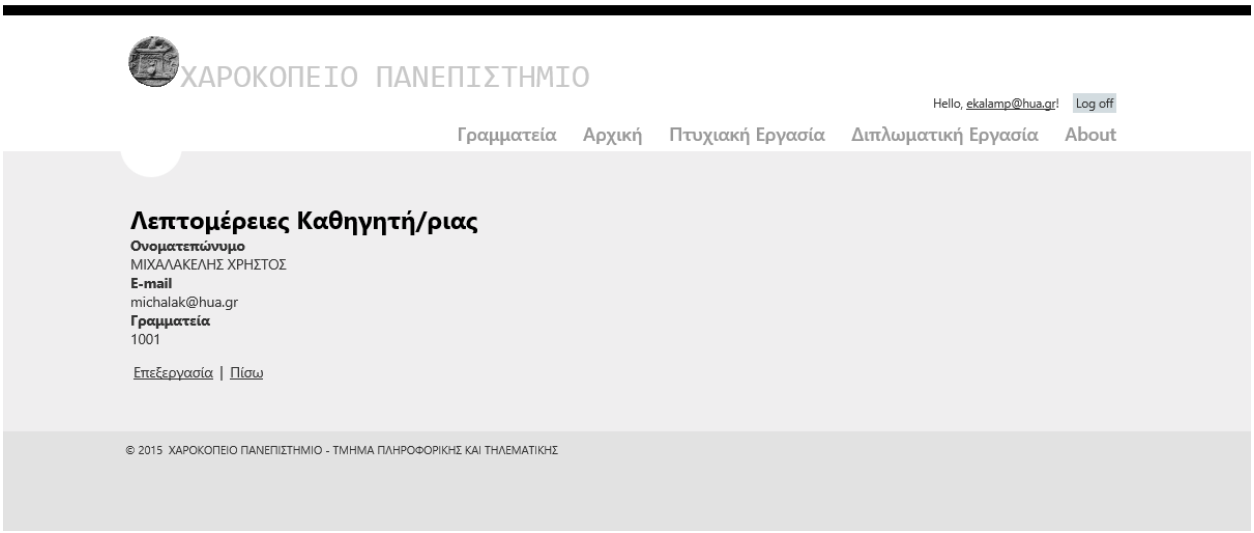
4.41 Διαγραφή Καθηγητή/ριας (Professor/Delete.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να προβεί σε Διαγραφή του/της επιλεγμένου/νης καθηγητή/ριας. Στην ουσία είναι η τελευταία προειδοποίηση από το σύστημα πριν γίνει η οριστική διαγραφή των καθηγητών. Για λόγους ασφαλείας η διαγραφή δεν σημαίνει και διαγραφή του αντίστοιχου λογαριασμού! Επιπλέον, εμφανίζει όλα τα στοιχεία αυτού/ης του/της καθηγητή/ριας και ένα προειδοποιητικό μήνυμα διότι αν η γραμματεία προβεί σε διαγραφή καθηγητή/ριας που έχει αντιστοιχηθεί σε κάποια εργασία το σύστημα δεν θα το επιτρέψει! Η γραμματεία θα δει την σελίδα Error με τις κατάλληλες υποδείξεις. Τέλος, μπορείς να πλοηγηθείς στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



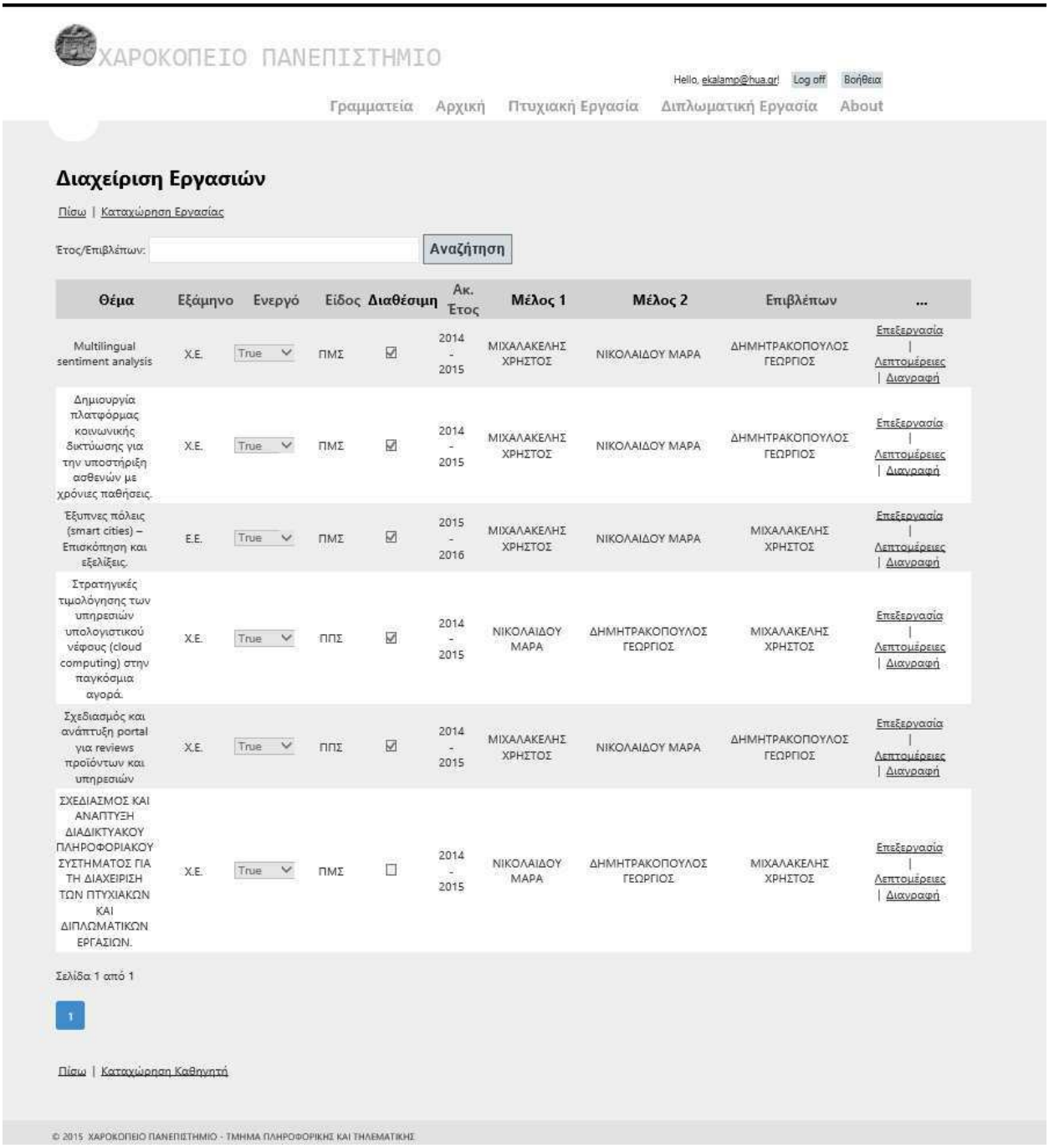
4.42 Λεπτομέρειες Καθηγητή/ριας (Professor/Details.aspx)

Στη σελίδα αυτή προβάλλονται με λεπτομέρειες όλα τα στοιχεία του/της επιλεγμένου/ης καθηγητή/ριας. Με δυνατότητα επεξεργασίας και επιστροφής στην προηγούμενη σελίδα από τα links στο κάτω μέρος της οθόνης.



4.43 Διαχείριση Εργασιών (Project/Index.aspx)

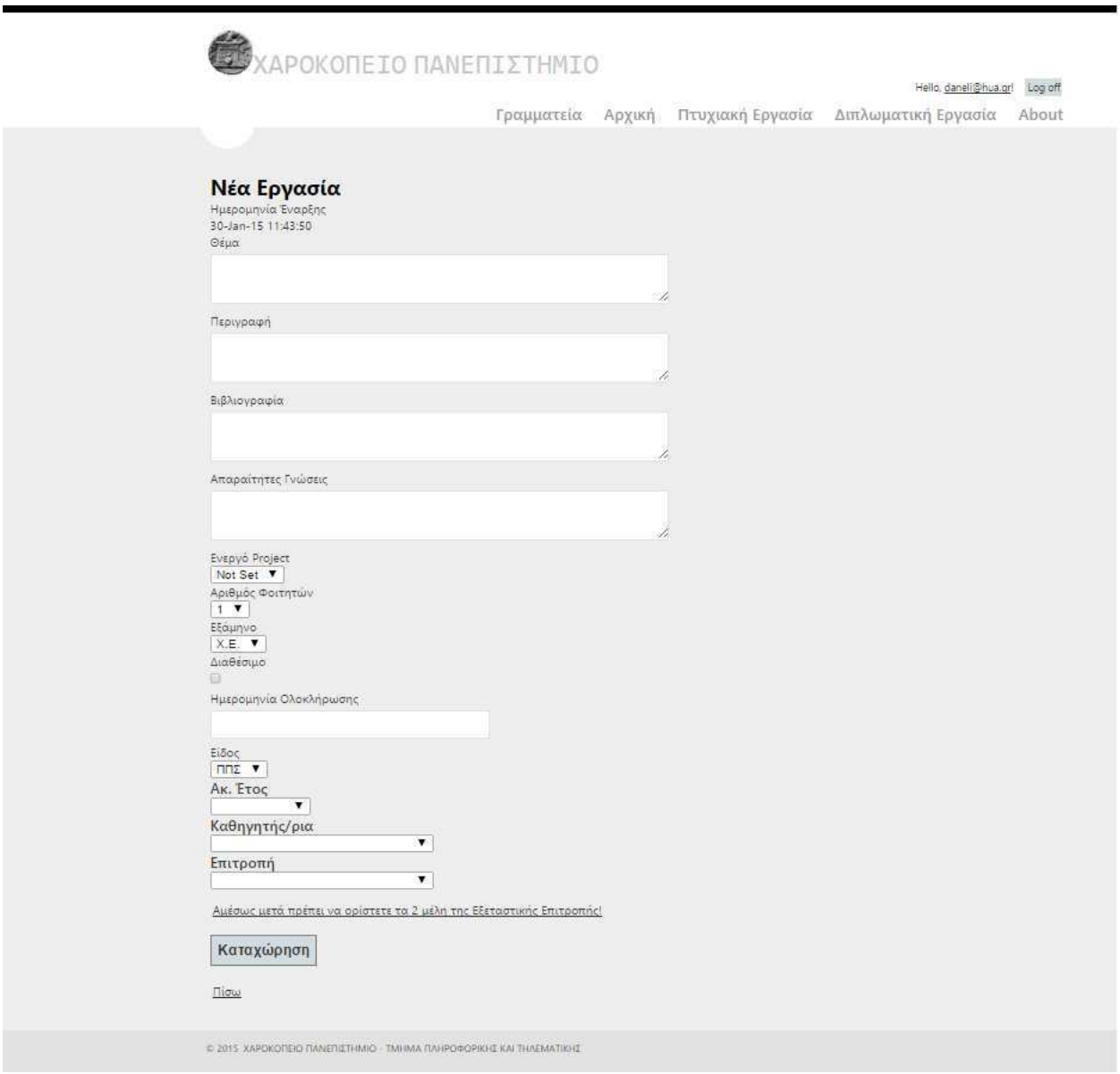
Πρόκειται για την αρχική σελίδα των Εργασιών. Από εδώ προβάλλονται όλες οι εργασίες με δυνατότητα δημιουργίας νέας ακολουθώντας το link στο επάνω μέρος της οθόνης, επεξεργασίας, διαγραφής και προβολής λεπτομερειών για την επιλεγμένη εργασία καθώς και πλοήγησης στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



4.44 Νέα Εργασία (Project/Create.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να καταχωρήσει νέες εργασίες στο σύστημα εκ μέρους κάποιου καθηγητή πάντα. Επίσης, εμφανίζει πεδία εισαγωγής για όλα τα στοιχεία αυτής της εργασίας και ένα προειδοποιητικό μήνυμα σε link διότι το σύστημα απαιτεί τον ορισμό ων δύο

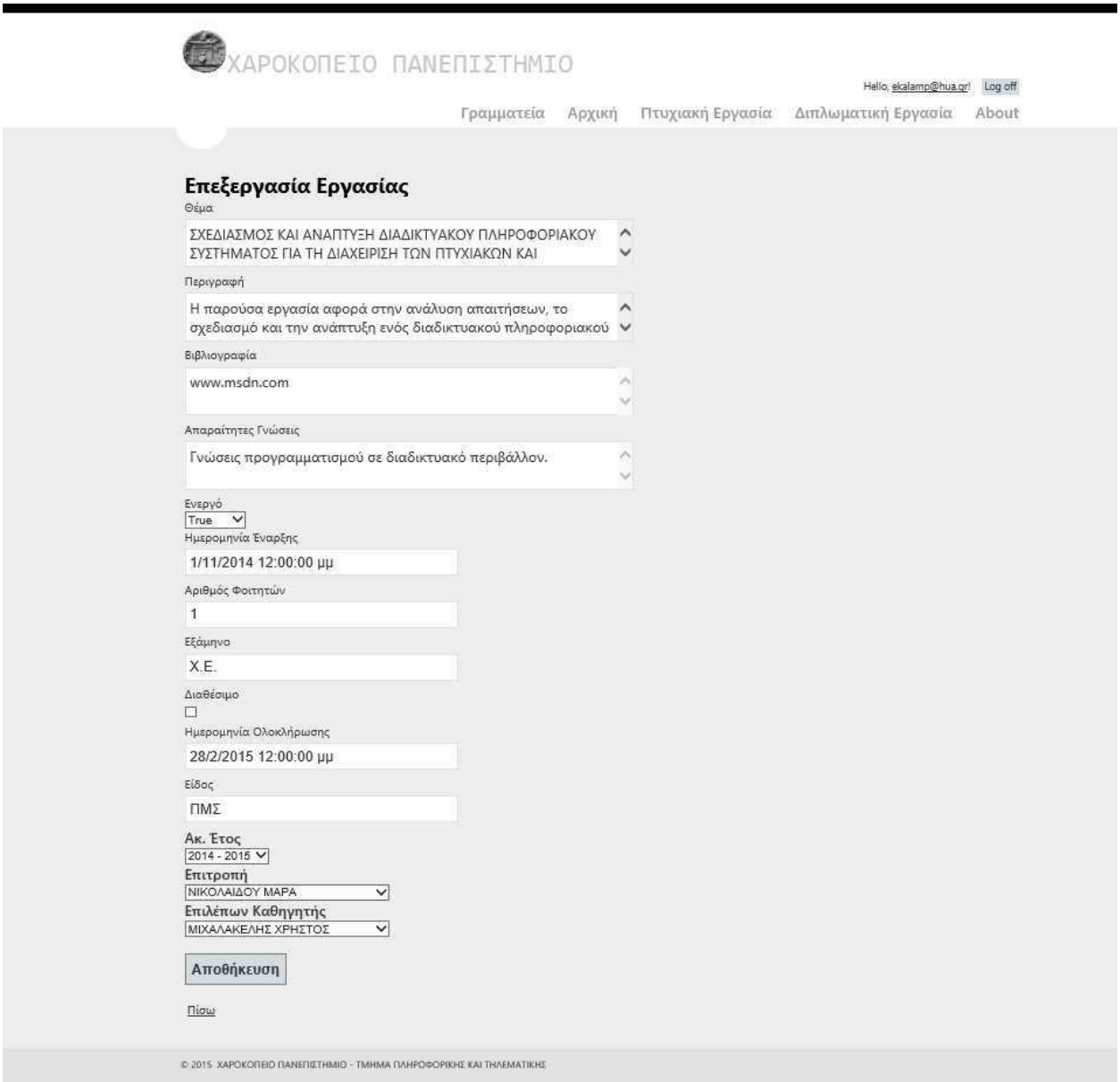
μελών της εξεταστικής επιτροπής. Το πεδίο *απαραίτητες γνώσεις* δεν απαιτείται παρ' όλα βοηθά τους φοιτητές στην επιλογή πτυχιακής/διπλωματικής εργασίας. **Προσοχή** στην ορθή συμπλήρωση των πεδίων!



4.45 Επεξεργασία Εργασίας (Project/Edit.aspx)

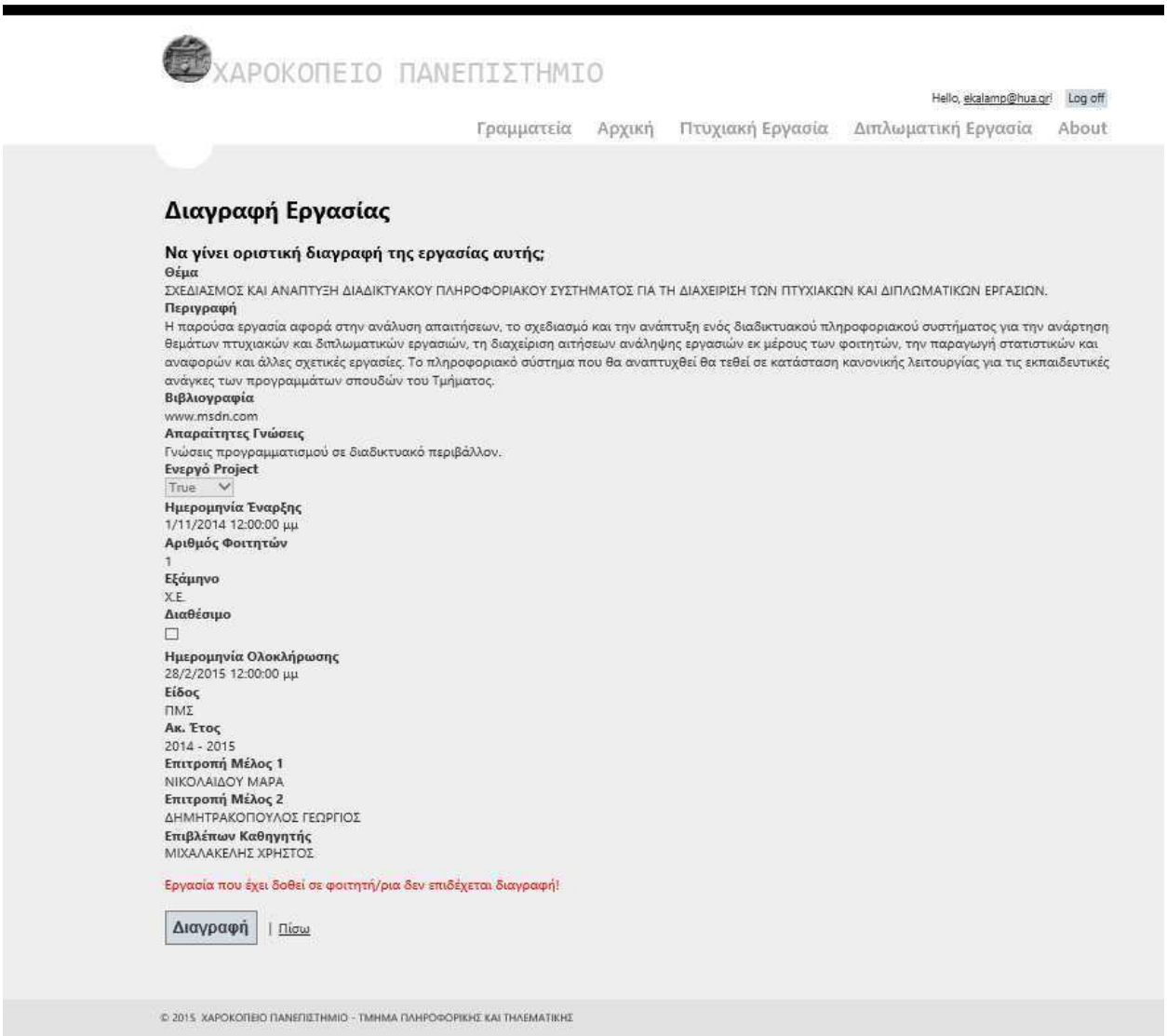
Αν κάτι δεν γίνει σωστά κατά την καταχώρηση μιας εργασίας η γραμματεία μπορεί να την επεξεργαστεί και να τη διορθώσει. Καλό είναι να αποφεύγεται η αλλαγή του πεδίου της *επιτροπής* ώστε να μην δημιουργούνται προβλήματα! Η επιτροπή αλλάζει από την *επεξεργασία επιτροπής* για

κάθε εργασία!



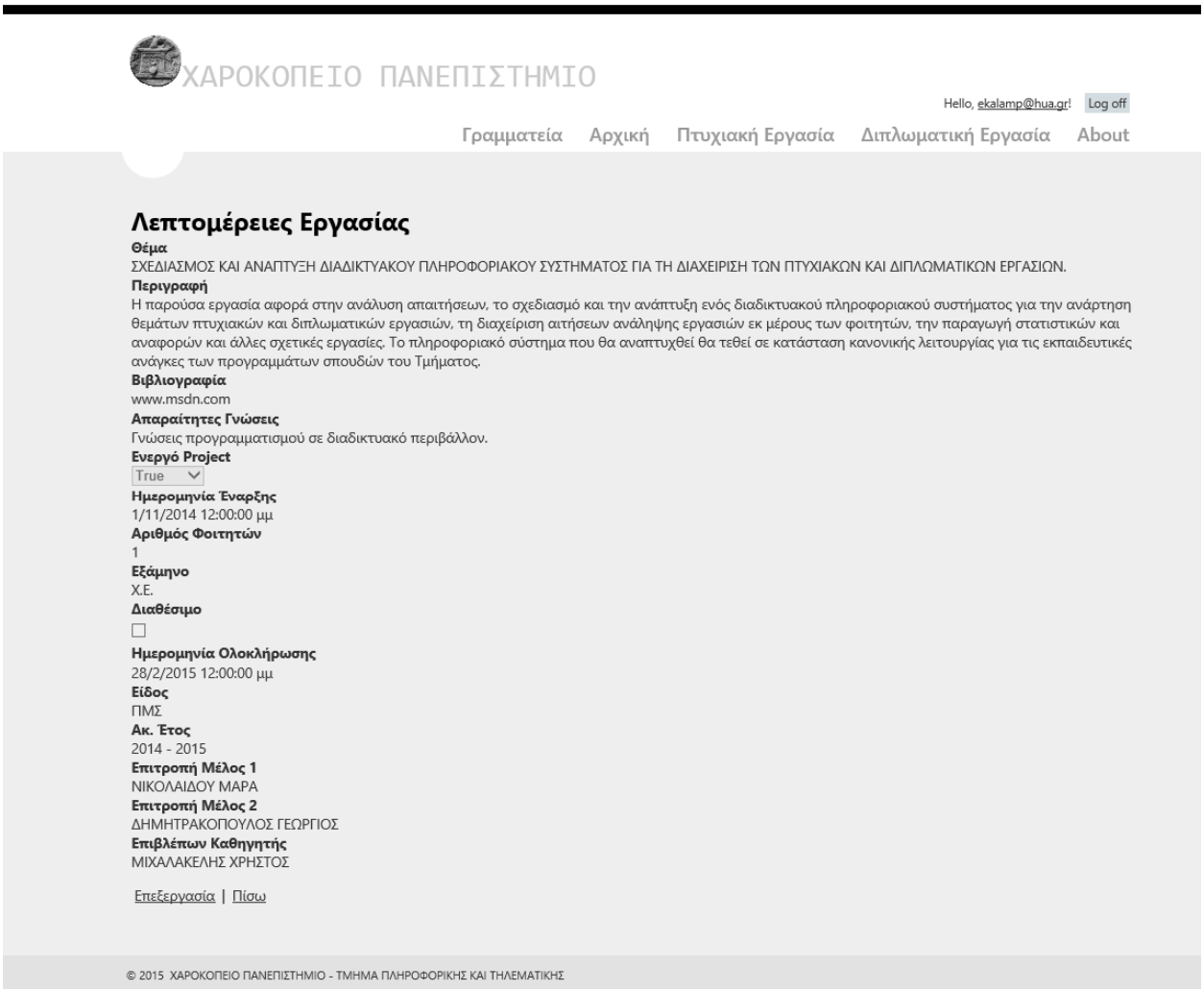
4.46 Διαγραφή Εργασίας (Project/Delete.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να προβεί σε Διαγραφή της επιλεγμένης εργασίας. Στην ουσία είναι η τελευταία προειδοποίηση από το σύστημα πριν γίνει η οριστική διαγραφή της επιλεγμένης εργασίας. Επιπλέον, εμφανίζει όλα τα στοιχεία αυτής της εργασίας και ένα προειδοποιητικό μήνυμα διότι αν η γραμματεία προβεί σε διαγραφή εργασίας που αφορά καθηγητή/φοιτητή το σύστημα δεν θα το επιτρέψει! Η γραμματεία θα δεί την σελίδα Error με τις κατάλληλες υποδείξεις. Τέλος, μπορείς να πλοηγηθείς στην προηγούμενη σελίδα κάνοντας «κλίκ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



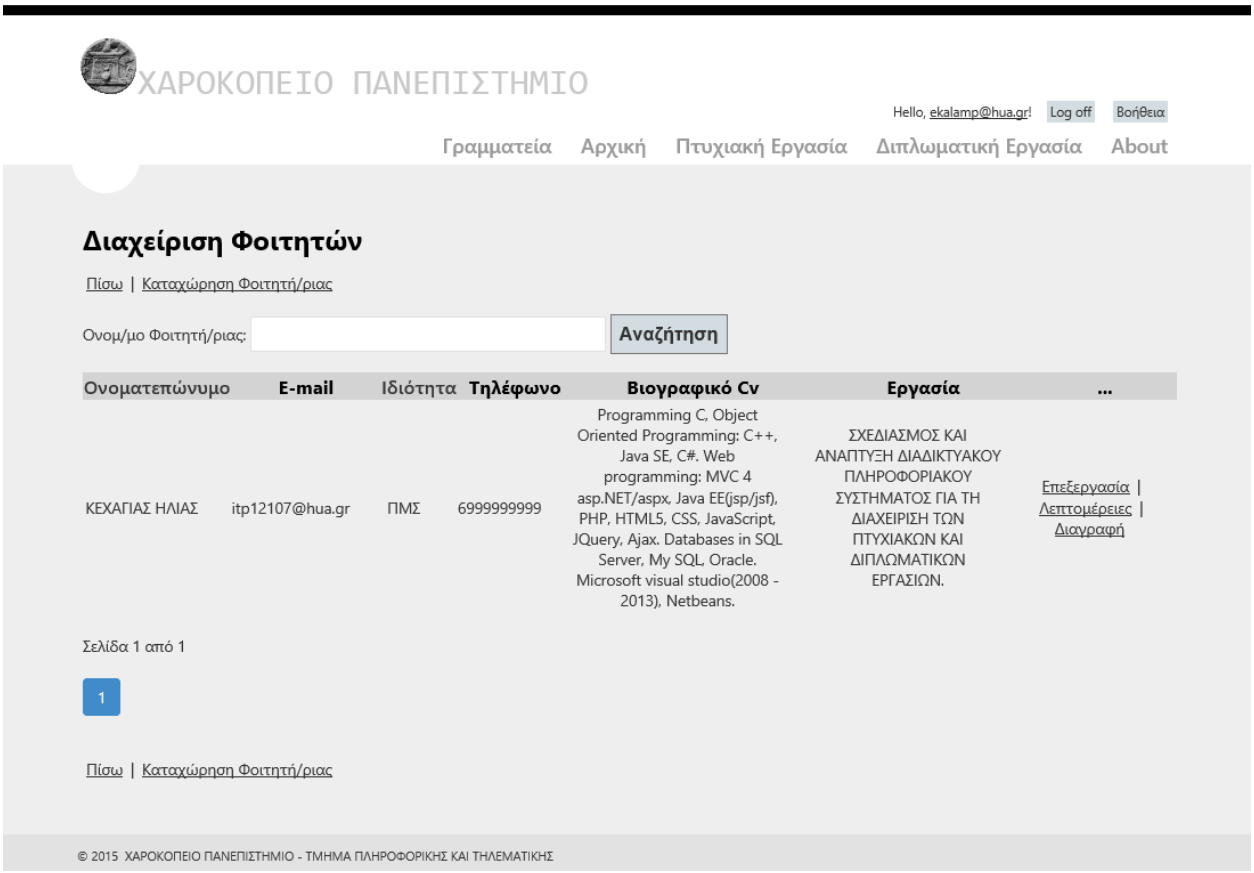
4.47 Λεπτομέρειες Εργασίας (Project/Details.aspx)

Στη σελίδα αυτή προβάλλονται με λεπτομέρειες όλα τα στοιχεία της επιλεγμένης εργασίας. Με δυνατότητα επεξεργασίας και επιστροφής στην προηγούμενη σελίδα από τα links στο κάτω μέρος της οθόνης.



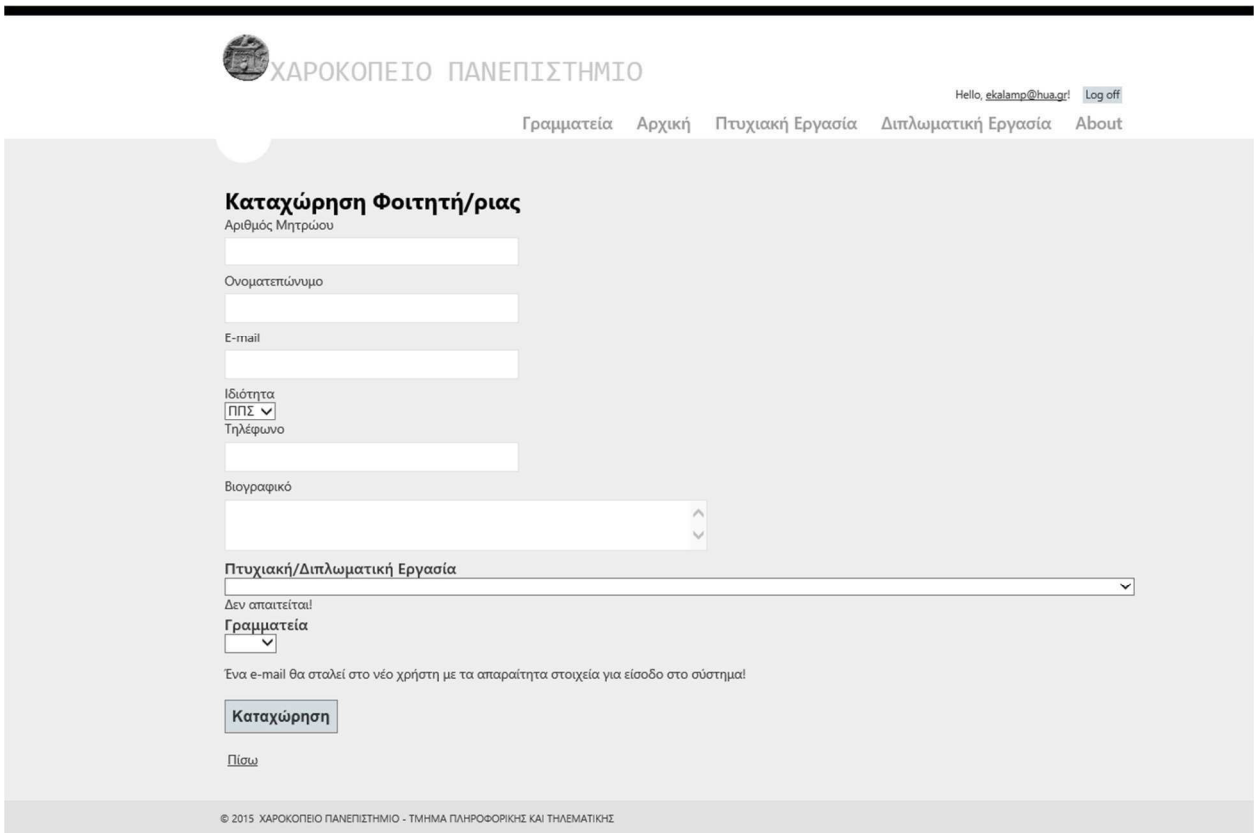
4.48 Διαχείριση Φοιτητών (Student/Index.aspx)

Πρόκειται για την αρχική σελίδα των Φοιτητών με σκοπό την διαχείριση τους από την γραμματεία του τμήματος. Κάθε φοιτητής/ρια ανήκει στον αντίστοιχο ρόλο και έχει δικό του λογαριασμό. Από εδώ προβάλλονται όλοι οι φοιτητές του τμήματος με δυνατότητα δημιουργίας νέου/ας ακολουθώντας το link στο επάνω μέρος της οθόνης, επεξεργασίας, διαγραφής και προβολής λεπτομερειών για τον/την επιλεγμένο/η φοιτητή/ρια καθώς και πλοήγησης στην προηγούμενη σελίδα κάνοντας «κλικ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



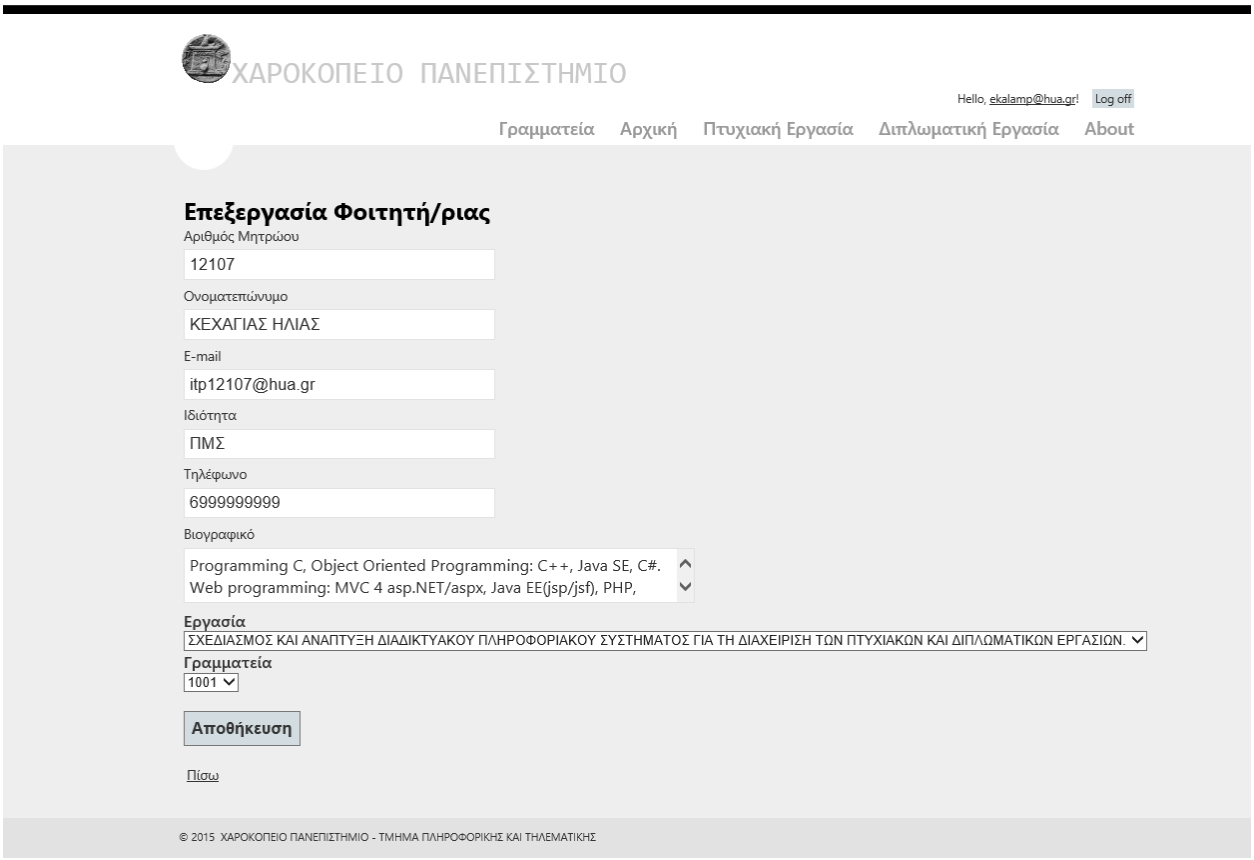
4.49 Καταχώρηση Φοιτητή/ριας (Student/Create.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να καταχωρήσει νέους φοιτητές/ριες στο σύστημα. Στην ουσία σημαίνει και δημιουργία του αντίστοιχου λογαριασμού! Επιπλέον, εμφανίζει πεδία εισαγωγής για όλα τα στοιχεία αυτού/ης του/της φοιτητή/ριας και ένα προειδοποιητικό μήνυμα διότι ένα e-mail θα σταλεί στον φοιτητή που καταχωρήθηκε με τα στοιχεία του για την εισαγωγή του στο σύστημα! Για να φαίνεται στο σύστημα ποιος από τη γραμματεία έκανε την αλλαγή καλό θα ήταν να συμπληρώνεται το αντίστοιχο προαιρετικό πεδίο, χωρίς να έχει κρίσιμη σημασία στην συγκεκριμένη εφαρμογή. Δεν απαιτείται αντιστοίχηση του φοιτητή σε κάποια εργασία οπότε μπορεί να μείνει ως έχει (κενό). **Προσοχή** στην ορθή συμπλήρωση των πεδίων!



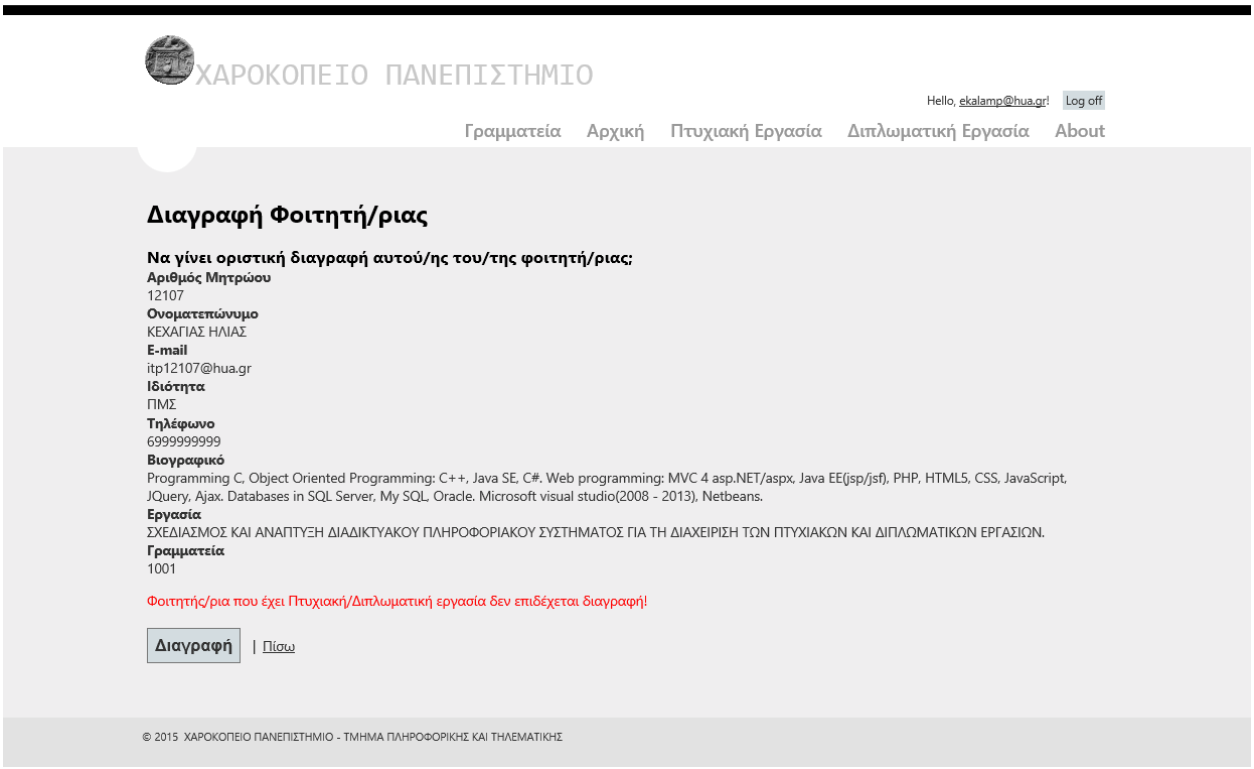
4.50 Επεξεργασία Φοιτητή/ριας (Student/Edit.aspx)

Αν κάτι δεν γίνει σωστά κατά την καταχώρηση των φοιτητών η γραμματεία έχει τη δυνατότητα να προβεί σε επεξεργασία για διόρθωση. Για να φαίνεται στο σύστημα ποιος από τη γραμματεία έκανε την αλλαγή καλό θα ήταν να συμπληρώνεται το αντίστοιχο προαιρετικό πεδίο, χωρίς να έχει κρίσιμη σημασία στην συγκεκριμένη εφαρμογή. **Προσοχή** στην ορθή συμπλήρωση των πεδίων!



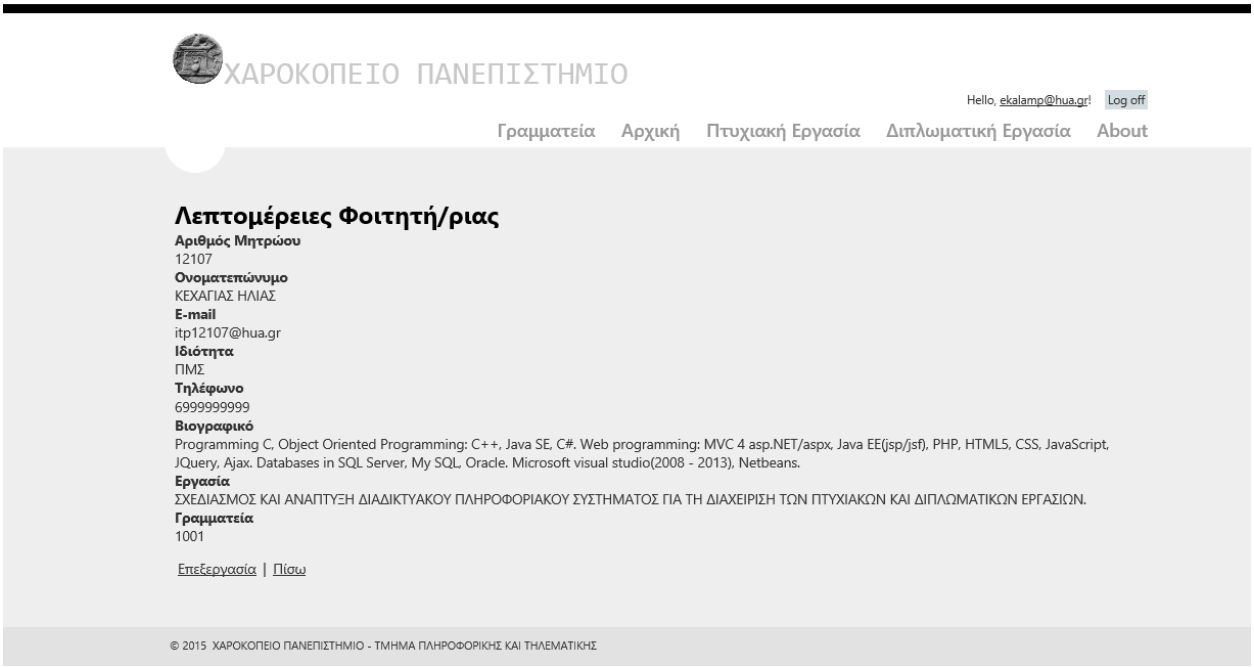
4.51 Διαγραφή Φοιτητή/ριας (Student/Delete.aspx)

Μέσω της σελίδας αυτής η γραμματεία έχει τη δυνατότητα να προβεί σε Διαγραφή του/της επιλεγμένου/νης φοιτητή/ριας. Στην ουσία είναι η τελευταία προειδοποίηση από το σύστημα πριν γίνει η οριστική διαγραφή των φοιτητών. Για λόγους ασφαλείας η διαγραφή δεν σημαίνει και διαγραφή του αντίστοιχου λογαριασμού! Επιπλέον, εμφανίζει όλα τα στοιχεία αυτού/ης του/της φοιτητή/ριας και ένα προειδοποιητικό μήνυμα διότι αν η γραμματεία προβεί σε διαγραφή φοιτητή/ριας που έχει αντιστοιχηθεί σε κάποια εργασία το σύστημα δεν θα το επιτρέψει! Η γραμματεία θα δεί την σελίδα Error με τις κατάλληλες υποδείξεις. Τέλος, μπορείς να πλοηγηθείς στην προηγούμενη σελίδα κάνοντας «κλίκ» στον σύνδεσμο στο κάτω μέρος της οθόνης.



4.52 Λεπτομέρειες Φοιτητή/ριας (Student/Details.aspx)

Στη σελίδα αυτή προβάλλονται με λεπτομέρειες όλα τα στοιχεία του/της επιλεγμένου/ης φοιτητή/ριας. Με δυνατότητα επεξεργασίας και επιστροφής στην προηγούμενη σελίδα από τα links στο κάτω μέρος της οθόνης.



4.53 Default Σελίδα σφαλμάτων(Error.aspx)

Εκτός από τα προειδοποιητικά μηνύματα στις σελίδες που απαιτείται υπάρχει και η σελίδα σφαλμάτων μέσω της οποίας κάθε χρήστης ενημερώνεται για το τι πήγε στραβά στην διάρκεια ορθής χρήσης της εφαρμογής. Υπάρχουν και κάποιες υποδείξεις ώστε να δράσει κατάλληλα. Ευχής έργο θα ήταν να μην εμφανίζεται συχνά αν όχι ποτέ!

ΚΕΦΑΛΑΙΟ 5

ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΚΩΔΙΚΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

5.1 Εισαγωγή

Το ASP.NET είναι ένα open source [2] server-side Web application framework σχεδιασμένο για την ανάπτυξη δυναμικών ιστοσελίδων. Αυτό αναπτύχθηκε από τη Microsoft για να επιτρέπει στους προγραμματιστές την κατασκευή δυναμικών ιστοσελίδων, εφαρμογών και υπηρεσιών web.

Ήταν η πρώτη που κυκλοφόρησε τον Ιανουάριο του 2002 με την έκδοση 1.0 του .NET Framework, και είναι ο διάδοχος της τεχνολογίας της Microsoft Active Server Pages (ASP). Το ASP.NET είναι χτισμένο στην Common Language Runtime (CLR), που επιτρέπει στους προγραμματιστές να γράφουν κώδικα ASP.NET χρησιμοποιώντας οποιαδήποτε υποστηριζόμενη γλώσσα .NET. Το framework επέκτασης SOAP ASP.NET επιτρέπει ASP.NET components να επεξεργάζονται μηνύματα SOAP. [1]

Οι ASP.NET ιστοσελίδες, επίσημα γνωστές ως Web Forms, [3], είναι τα κύρια δομικά στοιχεία για την ανάπτυξη εφαρμογών. [4] Οι διαδικτυακές φόρμες που περιέχονται σε αρχεία με κατάληξη ".aspx" περιέχουν συνήθως στατικές (X) HTML markup, όπως markup που ορίζουν server-side Web Controls και User Controls, όπου οι προγραμματιστές τα ρυθμίζουν κατάλληλα για τη λειτουργικότητα της ιστοσελίδας. Επιπλέον, περιέχουν δυναμικό κώδικα, ο οποίος «τρέχει» στον server και μπορεί να τοποθετηθεί σε μια σελίδα μέσα σε ένα μπλοκ <% - δυναμικός κώδικας - %>. Είναι παρόμοιο με άλλα είδη τεχνολογιών ανάπτυξης Web σελίδων όπως PHP, JSP, ASP. Με το ASP.NET Framework 2.0, η Microsoft εισήγαγε ένα νέο code-behind μοντέλο που επιτρέπει στατικό κείμενο να παραμείνει στη σελίδα .aspx, ενώ ο δυναμικός κώδικας παραμένει στο αρχείο .aspx.vb ή .aspx.cs ή .aspx.fs (ανάλογα σχετικά με τη γλώσσα προγραμματισμού που χρησιμοποιείται). [5]

Παρέχονται τα εξής,

- Code - behind model
- Directives
- User controls
- Custom controls
- Rendering technique
- State management
- Application
- Session state
- In-process mode
- State server mode
- SQL Server mode
- View state
- Server - side caching

- Cookies, Caching, και χρήση Query String

Το ASP.NET είναι ένα σύγχρονο και modular web framework, μαζί με άλλα frameworks, όπως το Entity Framework . Το νέο framework θα κάνει χρήση του νέου Compiler ανοικτού πηγαίου κώδικα της .NET πλατφόρμας (με την κωδική ονομασία "Roslyn») έτσι ώστε να γίνει cross platform (ASP.NET MVC, ASP.NET Web API, και ASP.NET ιστοσελίδες), μια πλατφόρμα, χρησιμοποιώντας μόνο τις σελίδες Razor που θα συγχωνευθούν σε ένα ενιαίο MVC 6. [6] Το έργο ονομάζεται "ASP.NET vNext".

5.1.2 Η αρχιτεκτονική Model – View – Controller

Το Model - View - Controller (MVC) είναι ένα αρχιτεκτονικό πρότυπο λογισμικού για την υλοποίηση των διεπαφών χρήστη. Χωρίζει μια δεδομένη εφαρμογή λογισμικού σε τρία αλληλοσυνδεόμενα μέρη, έτσι ώστε να διαχωρίζονται η εσωτερική αναπαράσταση της πληροφορίας από τις λειτουργίες και τις πληροφορίες που παρουσιάζονται στον χρήστη. [7] [8]

Η αρχιτεκτονική αποτελείται από:

- τους Controllers, που περιέχουν τον πηγαίο κώδικα της εφαρμογής και μπορούν να στείλουν εντολές στο μοντέλο με σκοπό να ενημερώσουν την κατάστασή του (π.χ., η επεξεργασία ενός εγγράφου). Μπορούν επίσης να στείλουν εντολές σε συνδεδεμένες Views και να αλλάξει η παρουσίαση του μοντέλου (π.χ., από την κύλιση μέσω ενός εγγράφου).
- τα Models, που γνωστοποιούν στις σχετικές Views και στους Controllers του, όταν έχει υπάρξει μια αλλαγή στην κατάσταση του. Η γνωστοποίηση αυτή εξασφαλίζει στις Views την παραγωγή ενημερωμένης εξόδου, και στους Controllers την αλλαγή του διαθέσιμου συνόλου των εντολών. Σε ορισμένες περιπτώσεις, μια εφαρμογή MVC μπορεί αντί να είναι «παθητική», να αναζητά στο Model για τυχόν ενημερώσεις.
- τις Views, που αναζητούν πληροφορίες από το μοντέλο που χρησιμοποιείται για να δημιουργήσουν μια αναπαράσταση εξόδου στον χρήστη. [9]

5.1.3 Η γλώσσα Προγραμματισμού C#

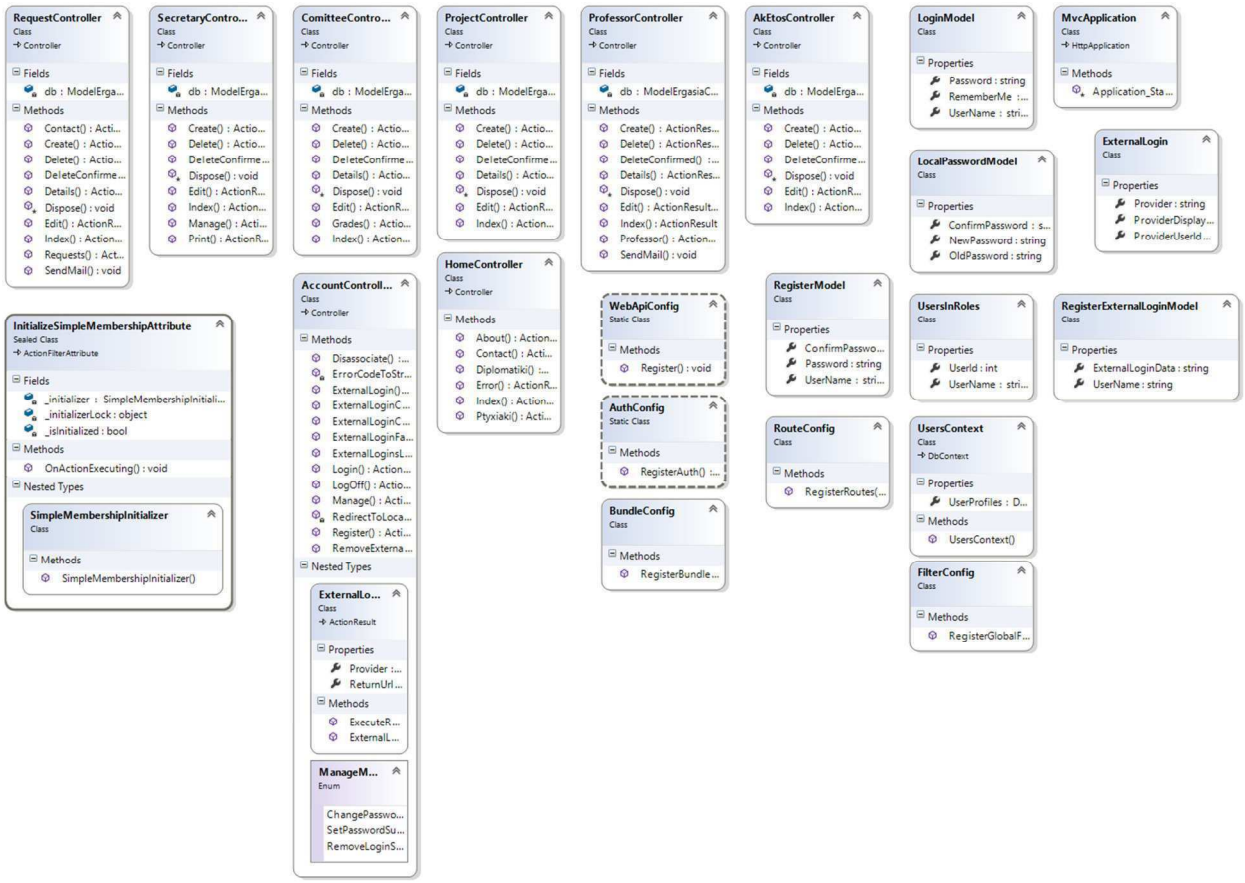
Η C # (προφέρεται "see sharp") είναι μια γλώσσα προγραμματισμού ηλεκτρονικών υπολογιστών. Έχει αναπτυχθεί από τη Microsoft. Δημιουργήθηκε για να χρησιμοποιήσει όλες τις δυνατότητες της πλατφόρμας .NET. Η πρώτη έκδοση κυκλοφόρησε το 2001. Η πιο πρόσφατη έκδοση είναι η C # 5.0, που κυκλοφόρησε τον Αύγουστο του 2012. Η C # είναι μια σύγχρονη γλώσσα. Η C # δημιουργήθηκε από μια ομάδα ανάπτυξης λογισμικού της Microsoft με επικεφαλής τον Anders Hejlsberg, τον δημιουργό της Delphi. (http://simple.wikipedia.org/wiki/C_Sharp_(programming_language))

Παρά το γεγονός ότι η C # προέρχεται από την γλώσσα προγραμματισμού C, εισάγει μερικά μοναδικά και ισχυρά χαρακτηριστικά, όπως delegates (που μπορεί να θεωρηθεί ως ασφαλείς τύπου δείκτες συναρτήσεων) και εκφράσεις λάμδα που εισάγουν τα στοιχεία των λειτουργικών γλωσσών προγραμματισμού, καθώς επίσης και μια απλούστερο ενιαίο ταξικό μοντέλο κληρονομικότητας (από τη C ++) και με μια πολύ γνωστή σύνταξη «C - like» που μπορεί να βοηθήσει τους αρχάριους να

προχωρήσουν γρήγορα. Είναι παρόμοια με την Java, είναι object-oriented, έρχεται με μια εκτεταμένη class library, και υποστηρίζει χειρισμό των exception, πολλαπλούς τύπους πολυμορφισμού και το διαχωρισμό των διεπαφών από την υλοποιήση. Τα χαρακτηριστικά αυτά, σε συνδυασμό με τα ισχυρά εργαλεία ανάπτυξης, την πολυ-πλατφόρμα υποστήριξης, και τα generics, καθιστούν τη C # μια καλή επιλογή για πολλούς τύπους έργων ανάπτυξης λογισμικού. Για παράδειγμα, έργα ταχείας ανάπτυξης εφαρμογών, έργα που υλοποιούνται από μεμονωμένα άτομα ή μικρές ή μεγάλες ομάδες, εφαρμογές Διαδικτύου, και έργα με αυστηρές απαιτήσεις αξιοπιστίας. Testing frameworks, όπως το NUnit κάνουν τη C # κατάλληλη για test - driven development και συνεπώς, μια καλή γλώσσα για Extreme Programming (XP). Είναι strong typing και αυτό βοηθά στην πρόληψη πολλών λαθών προγραμματισμού και πληκτρολόγησης χωρίς να συμβαίνει το ίδιο σε πολλές γλώσσες προγραμματισμού. (http://en.wikibooks.org/wiki/C_Sharp_Programming)

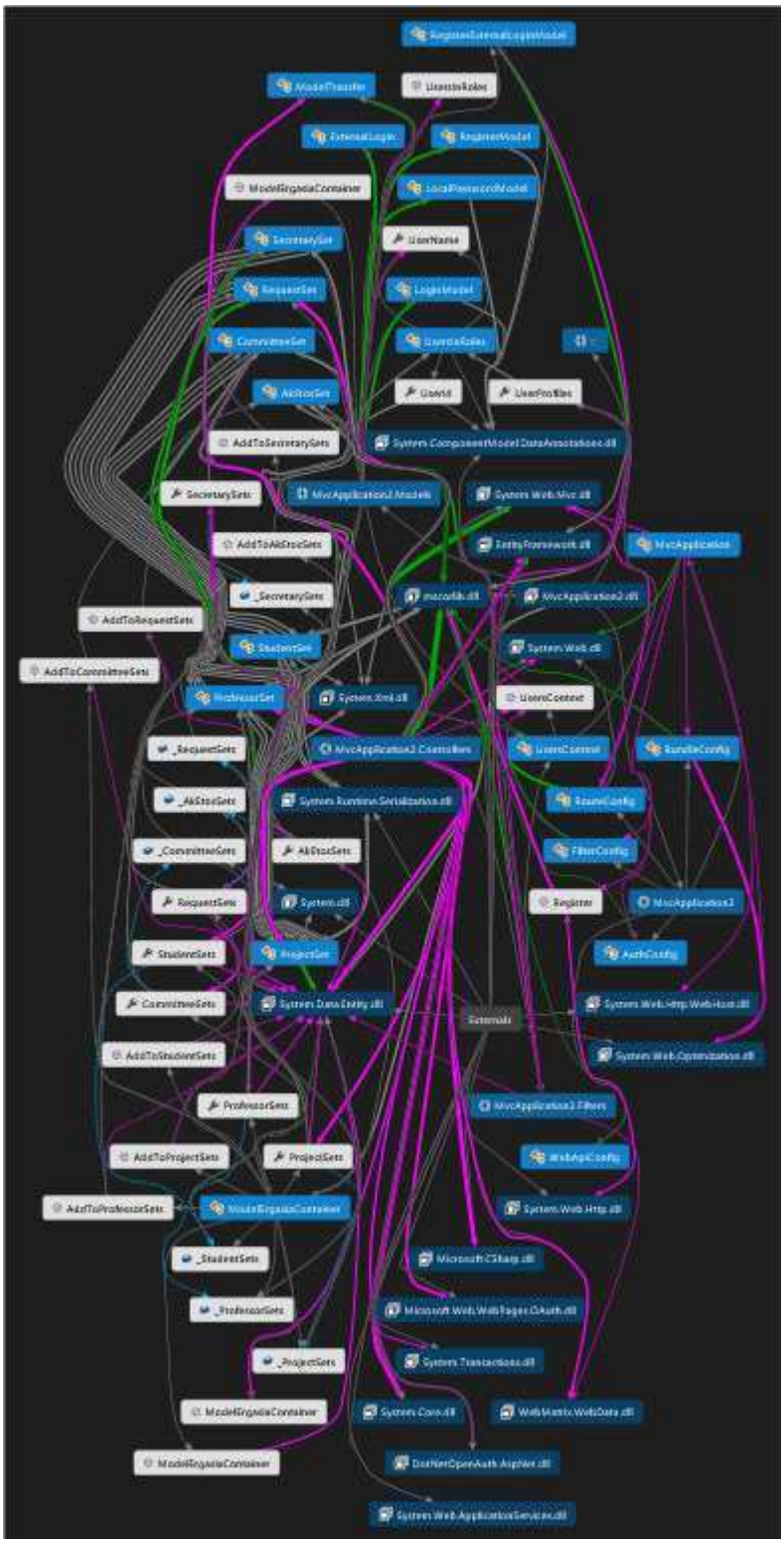
5.2 Το Class Diagram της εφαρμογής

Στο σημείο αυτό παραθέτονται οι κλάσεις τις εφαρμογής με τα αντίστοιχα πεδία και τις μεθόδους τους όπως παράχθηκαν με τη βοήθεια του Entity Framework 6.1.



5.3 Assembly Dependencies graph

Απεικονίζονται οι εξαρτήσεις του κώδικα της εφαρμογής σε γράφο. Χρησιμοποιείται για την κατανόηση αυτών των εξαρτήσεων σε πολύπλοκα συστήματα. Αναλυτικότερα, παρουσιάζει τον τρόπο που συνδέονται συνολικά όλα τα κομμάτια κώδικα μαζί χωρίς την ανάγνωση μέσα από ξεχωριστά αρχεία και γραμμές. Ο «χάρτης» παράγεται με το MS Visual Studio 2013 Ultimate Αυτό σας βοηθά να δούμε

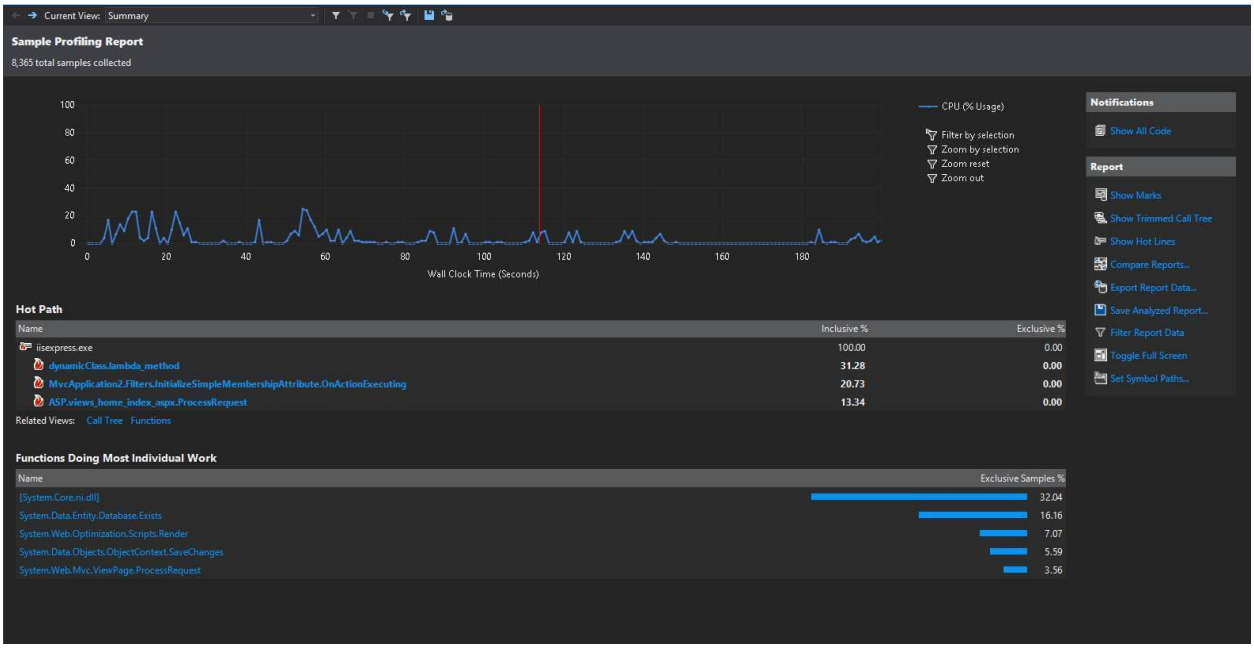


5.4 Αποτελέσματα μετρήσεων

Τα αποτελέσματα αυτά προκύπτουν από την ανάλυση της απόδοσης και των διαγνωστικών της εφαρμογής μέσα από το Microsoft Visual Studio 2013 Ultimate.

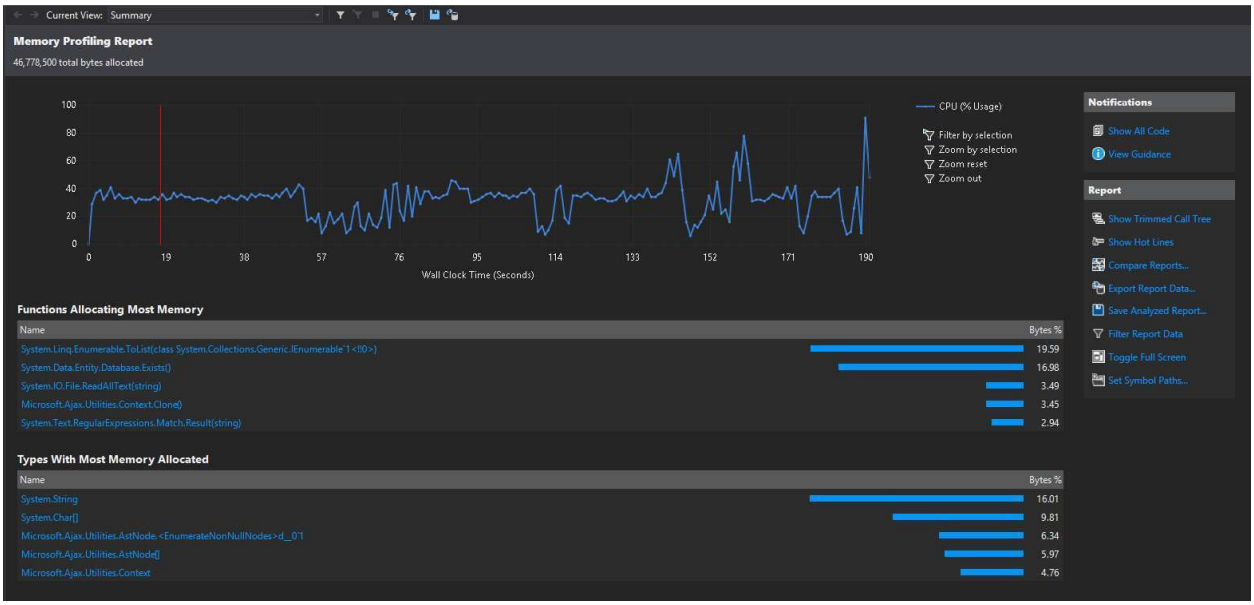
5.4.1 CPU Sampling

Στο στιγμιότυπο οθόνης που ακολουθεί αναλύεται η CPU(% Usage) σε Wall Clock Time (Seconds).



5.4.2 .NET memory allocation

Στο στιγμιότυπο οθόνης που ακολουθεί αναλύεται η CPU(% Usage) σε Wall Clock Time (Seconds).



5.5 Βιβλιογραφία & Αναφορές

[1] MacDonald Matthew, Szpuszta Mario (2005). Pro ASP.NET 2.0 in C# 2005 (1st edition ed.). Apress. ISBN 1-59059-496-7.

[2] "ASP.NET is part of a great open source .NET community". Microsoft. May 14, 2013.

[3] Staff (November 2001). "Overview of ASP.NET and Web Forms". Microsoft. Retrieved 2011-06-05.

[4] (MacDonald & Szpuszta 2005, p. 63)

[5] "Code Behind vs. Code Inline". Microsoft .NET Framework. Microsoft. Archived from the original on 11 November 2010. Retrieved 2010-11-22.

[6] ASP.NET 5 Overview | The ASP.NET Site

[7] "More deeply, the framework exists to separate the representation of information from user interaction." The DCI Architecture: A New Vision of Object-Oriented Programming - Trygve Reenskaug and James Coplien - March 20, 2009.

[8] Burbeck (1992): "... the user input, the modeling of the external world, and the visual feedback to the user are explicitly separated and handled by three types of object."

[9] Buschmann, Frank (1996) Pattern-Oriented Software Architecture.

(C# Developer's Headstart, Michaelis - Spokas, 200. ISBN 0-07219116-3).

ΚΕΦΑΛΑΙΟ 6

ΣΥΜΕΡΑΣΜΑΤΑ & ΕΠΕΚΤΑΣΕΙΣ

Είναι η κατάλληλη στιγμή για να θέσετε τη
δουλειά σας σε γερές θεωρητικές βάσεις.
- Sam Morgan

6.1 Αποτίμηση της Διπλωματικής Εργασίας

Αρχική επιδίωξη της εργασίας αυτής ήταν η ανάπτυξη μιας ολοκληρωμένης διαδικτυακής εφαρμογής που ως σκοπό έχει την παροχή κάποιου επιπέδου αυτοματοποίησης, μηχανοργάνωσης και παρακολούθησης της διαδικασίας όπου η γραμματεία του τμήματος απευθύνει πρόσκληση στους διδάσκοντες των προγραμμάτων σπουδών (ΠΠΣ / ΠΜΣ) να αποστείλουν κατάλογο των προτεινόμενων θεμάτων προκειμένου οι φοιτητές να επιλέξουν την εργασία που θα εκπονήσουν.

Έγινε μια αξιόλογη προσπάθεια επίτευξης αυτού του στόχου και έτσι σχεδιάστηκε και υλοποιήθηκε αυτό το ολοκληρωμένο Πληροφοριακό Σύστημα διαχείρισης της διαδικασίας.

Καμία προσπάθεια, όμως, δεν πετυχαίνεται χωρίς δυσκολίες. Από τη δυσκολία να συγκεντρώσεις το υλικό που θα ενσωματώσεις στην εφαρμογή σου έως τη δυσκολία να δημιουργήσεις μια απλή και κατανοητή, χωρίς αδιέξοδα και χωρίς «μαύρα σημεία» εφαρμογή στον τελικό χρήστη, όσο είναι εφικτό.

Στην παρούσα εφαρμογή χρειάστηκε να γραφούν αρκετές γραμμές κώδικα διότι πρέπει κάθε φορά ο χρήστης να ενημερώνεται για την κατάστασή της και να παίρνει τα κατάλληλα μηνύματα ορθής ή εσφαλμένης απόκρισης. Επίσης, χρειάστηκε αρκετός αριθμός ελέγχων μέσω δομών τύπου if … else, try … catch, switch … case, foreach για τη διασφάλιση της σωστής λειτουργίας της.

Αξίζει να σημειωθεί και η δυσκολία εμπιστοσύνης στο πρόσωπο ενός νέου μηχανικού με διαφορετικές αντιλήψεις και συνήθειες από αυτές του συνόλου του Τμήματος και οι διάφορες καταστάσεις που αυτό απορρέει.

6.2 Μελλοντικές επεκτάσεις της εφαρμογής

Κανένα σύστημα δε είναι τέλειο, αψεγάδιαστο! Τα περισσότερα λογισμικά συστήματα εμπεριέχουν πολλές ατέλειες και δυσλειτουργίες που με τον καιρό λειτουργίας τους, βελτιώνονται από τους προγραμματιστές - δημιουργούς τους λαμβάνοντας υπόψη τη γνώμη και την κρίση των χρηστών.

Να σημειωθεί ότι το σύστημα διαμορφώθηκε κατά τη βούληση του τμήματος οπότε οποιαδήποτε δική μου πρωτοβουλία θα ενσωματωθεί στην επόμενη έκδοση της εφαρμογής με σκοπό την εξυπηρέτηση και άλλων τμημάτων σε άλλα Εκπαιδευτικά Ιδρύματα. (Για παράδειγμα, βαθμολόγηση πτυχιακής / διπλωματικής εργασίας, απευθείας απόθεση στο διαδικτυακό τόπο κάθε Ιδρύματος του Βιβλίου της εργασίας και του κώδικα της εφαρμογής, αν υπάρχει και διαδικασία αναζήτησης όπου απαιτείται).

Είμαι ιδιαίτερα χαρούμενος που δεν απαιτείται να σταθώ σε θέματα ασφάλειας της εφαρμογής, έχω απλά υιοθετήσει τις Default ρυθμίσεις συστημάτων MVC 4 Web Application (Internet Application) για τις σελίδες και του SQL Server 2014 για τη βάση δεδομένων.

Όσον αφορά, τη διαδικασία αυθεντικοποίησης των χρηστών κρίνεται αναγκαίο να δοθεί η δυνατότητα ταυτοποίησής τους μέσα από Lightweight Directory Access Protocol (LDAP) που τα περισσότερα, αν όχι όλα τα Εκπαιδευτικά Ιδρύματα υποστηρίζουν.

Από την πλευρά της διαχείρισης της εφαρμογής καλό θα ήταν μελλοντικά να προστεθεί και ο ρόλος *Διαχειριστής* (Administrator). Έτσι θα παρέχεται μια άλλου επιπέδου διαχείριση που τώρα αναλαμβάνει ο ρόλος *Γραμματεία*.

Επιπροσθέτως, βελτίωση απαιτεί και η διαδικασία αποστολής E-mail που τώρα πετυχαίνεται με ένα Google Account (Gmail / mail.google.com) με όνομα *managementthesishua* και διεύθυνση ηλεκτρονικού ταχυδρομείου *managementthesishua@gmail.com*. Μπορεί να αντικατασταθεί με το open source λογισμικό Postal μέσω του Nuget packager που παρέχεται με το visual studio.

Τέλος, το Graphical user interface (GUI) της εφαρμογής είναι δομημένο στην τεχνολογία ASP.NET (.aspx), η οποία μέσω του MVC 3 και μετά μπορεί να αλλάξει και να γίνει ASP.NET Razor (.cshtml) το οποίο είναι Open Source. Όσον αφορά το GUI, παρόλο που έγινε προσπάθεια για να είναι κατανοητό, ευχάριστο και απλό είμαι σίγουρος ότι επιδέχεται αλλαγές, που όπως ανέφερα και πιο πάνω, περιμένω Feedback από τους χρήστες…

Βιβλιογραφία

[1] Scott Guthrie (3 October 2007). "Releasing the Source Code for the NET Framework". Archived from the original on 7 September 2010. Retrieved 15 September 2010.

[2] ".NET Framework 3.5 Deployment Guide for Administrators".MSDN. Microsoft. Retrieved 26 June 2008.

[3] Chen, Raymond (4 April 2011). "Windows is not a .NET Framework delivery channel either". Retrieved 17 June 2014.

[4] ".NET Framework Versions and Dependencies".MSDN. Microsoft. Retrieved 17 January 2014.

[5] "What's New in Visual Studio .NET 2003". MSDN.Microsoft. Retrieved 1 September 2014. "Visual Studio .NET 2002 shipped with the Microsoft .NET Framework SDK version 1.0. Visual Studio .NET 2003 ships with .NET Framework SDK version 1.1."

[6] ".NET Framework Developer Center – Frequently Asked Questions". Archived from the original on July 24, 2012.

[7] "What's New in Visual Studio 2005". MSDN. Microsoft. Retrieved 1 September 2014.

[8] Chinnathambi, Kirupa (4 December 2006). "Expression Blend -- What Is That?". Expression Blend and Design. Microsoft. Retrieved 1 September 2014.

[9] Guthrie, Scott (19 November 2007). "Visual Studio 2008 and .NET 3.5 Released". Scott Gu's Blog. Microsoft. Retrieved 1 September 2014.

[10] "What's New in Visual Studio 2010". MSDN. Microsoft. Retrieved 1 September 2014.

[11] "What's New in Visual Studio 2012". MSDN. Microsoft. Retrieved 1 September 2014.

[12] "What's New in Visual Studio 2013". MSDN. Microsoft. Retrieved 1 September 2014.

[13] http://myitforum.com/myitforumwp/2014/10/02/windows-10-management/

[14] "Installing the .NET Framework 3.5 on Windows 8 or 8.1".MSDN. Microsoft. Retrieved 13 August 2014.

[15] "Scott Guthrie: Silverlight and the Cross-Platform CLR".Channel 9. 30 April 2007.

[16] "ECMA 335 - Standard ECMA-335 Common Language Infrastructure (CLI)". ECMA. 1 June 2006. Archived from the original on 14 June 2008. Retrieved 1 June 2008.

[17] "ISO/IEC 23271:2006". Standards.iso.org. 2006-09-29. Retrieved 2012-04-17.

[18] "Technical Report TR/84 Common Language Infrastructure (CLI) - Information Derived from Partition IV XML File". ECMA. 1 June 2006.

[19] "ECMA-334 C# Language Specification". ECMA. 1 June 2006.

[20] "Standard ECMA-372 C++/CLI Language Specification". ECMA. 1 December 2005.

[21] "Base Class Library". Retrieved 1 June 2008.

[22] "Garbage Collection: Automatic Memory Management in the Microsoft .NET Framework". Archived from the originalon 3 July 2007. Retrieved 1 June 2008.

[23] "Garbage collection in .NET". Archived from the original on 25 May 2008. Retrieved 1 June 2008.

[24] "Garbage Collection—Part 2: Automatic Memory Management in the Microsoft .NET Framework". Archived from the original on 26 June 2007. Retrieved 1 June 2008.

[25] "Standard ECMA-335: Common Language Infrastructure (CLI)". ecma-international.org (6 ed.). ECMA. June 2012.

[26] "Standard ECMA-334: C# Language Specification". ecma-international.org (4 ed.). ECMA. June 2006.

[27] "ISO/IEC 23271:2012 Information technology - Common Language Infrastructure (CLI)". iso.org (3 ed.). International Organization for Standardization. 13 February 2012.

[28] "ISO/IEC 23270:2006 - Information technology - Programming languages - C#". iso.org (2 ed.). International Organization for Standardization. 26 January 2012.

[29] "Microsoft's Empty Promise". Free Software Foundation. 16 July 2009. Archived from the original on 5 August 2009. Retrieved 3 August 2009. "However, there are several libraries that are included with Mono, and commonly used by applications like Tomboy, that are not required by the standard. And just to be clear, we're not talking about Windows-specific libraries like ASP.NET and Windows Forms. Instead, we're talking about libraries under the System namespace that provide common functionality programmers expect in modern programming languages"

[30] ISO 9001:2008, Foreword

[31] "Using VB.NET with the .NET Micro Framework « /dev/mobile". Christec.co.nz. 2008-04-01. Retrieved 2012-04-17.

[32] "CrossNet". Codeplex.com. Retrieved 2012-04-17.

[33] "Understanding Garbage Collection in .NET".

[34] ''Mono's SIMD Support: Making Mono safe for Gaming'". Tirania.org. 2008-11-03. Retrieved 2012-04-17.

[35] "Reverse Engineering Risk Assessment".

[36] Gartner, Inc. as reported in "Hype Cycle for Cyberthreats, 2006", September 2006, Neil MacDonald; Amrit Williams, et al.

[37] Dotfuscator Community Edition 4.0

(http://msdn.microsoft.com/en-us/library/vstudio/hh420390(v=vs.110).aspx#whatsnew_feature)

(http://www.microsoft.com/en-us/server-cloud/products/sql-server-benchmarks/microsoft-tested.aspx)

(http://www.microsoft.com/en-us/server-cloud/products/sql-server-benchmarks/industry.aspx)

(http://www.microsoft.com/en-us/server-cloud/products/sql-server-benchmarks/partner.aspx)

http://msdn.microsoft.com/en-us/library/vstudio/w0x726c2(v=vs.110).aspx

Sites που βοήθησαν στη παραγωγή κώδικα

- **code project** http://www.codeproject.com/Articles/720471/A-Simple-MVC-Application-using-LINQ-to-SQL

- **Aaron Hoffman's Blog** http://aaron-hoffman.blogspot.gr/2013/02/aspnet-mvc-4-membership-users-passwords.html

- http://www.dotnetcurry.com/showarticle.aspx?ID=619

- **The ASP.NET Site** http://www.asp.net/web-pages/overview/security/16-adding-security-and-membership

- **The ASP.NET Site** http://www.asp.net/mvc/overview/gettingstarted/gettingstartedwithefusingmvc/advancedentityframeworkscenariosforanmvcwebapplication

- **code project** http://www.codeproject.com/Articles/635324/Another-set-of-ASP-NET-MVC-tips

- **C.M. Jackson .Net** http://www.cmjackson.net/2009/10/23/aspnetmvcusingformsauthenticationwithldap/

- **code project** http://www.codeproject.com/Articles/159749/ASP-NET-MVC-Model-Binding-Part

- **msdn** https://msdn.microsoft.com/en-us/data/gg685494.aspx

- **code project** http://www.codeproject.com/Articles/695850/Complete-CRUD-Operations-in-MVC

- **msdn** http://msdn.microsoft.com/enus/data/jj592907.aspx

- **msdn** https://msdn.microsoft.com/enus/library/ff647076.aspx

- **Aaron's Thoughts** http://www.aaroncoleman.name/post/2011/06/16/ForcingEntityFrameworktomarkDateTimefieldsatUTC.aspx

- https://www.digicert.com/ssl-certificate-installation-microsoft-iis-8.htm

- **code project** http://www.codeproject.com/Articles/288631/Secure-ASP-NET-MVC-applications

- **TechBrij** http://techbrij.com/role-based-menu-asp-net-mvc

- **RSSBus** http://www.rssbus.com/ado/ldap/hiw.aspx

- **msdn** http://msdn.microsoft.com/en-us/library/5k850zwb%28v=vs.100%29.aspx

- **The ASP.NET Site** http://www.asp.net/mvc/overview/olderversions/usingoauthproviderswithmvc

- **code project** http://www.codeproject.com/Tips/615776/WebGridinASPNETMVC

- **csharpcorner** http://www.csharpcorner.com/UploadFile/4b0136/performpagingsearchingsortinginAspNetmvc5

- Sorting, Filtering, and Paging with the Entity Framework in an ASP.NET MVC Application | **The ASP.NET Site**

ΠΑΡΑΡΤΗΜΑ Α

Η βάση δεδομένων

```
-- --------------------------------------------------
-- Entity Designer DDL Script for SQL Server 2005, 2008, and Azure
-- --------------------------------------------------
-- Date Created: 12/17/2014 13:36:36
-- Generated from EDMX file: E:\Documents\Visual Studio
2012\Projects\MvcApplication2\MvcApplication2\Models\ModelErgasia.edmx
-- --------------------------------------------------

SET QUOTED_IDENTIFIER OFF;
GO
USE [aspnet-MvcApplication2-20141102141035];
GO
IF SCHEMA_ID(N'dbo') IS NULL EXECUTE(N'CREATE SCHEMA [dbo]');
GO

-- --------------------------------------------------
-- Dropping existing FOREIGN KEY constraints
-- --------------------------------------------------

IF OBJECT_ID(N'[dbo].[FK_AkEtosProjectSet]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[ProjectSets] DROP CONSTRAINT [FK_AkEtosProjectSet];
GO
IF OBJECT_ID(N'[dbo].[FK_CommitteeSetProjectSet]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[ProjectSets] DROP CONSTRAINT [FK_CommitteeSetProjectSet];
GO
IF OBJECT_ID(N'[dbo].[FK_ProfessorSetCommitteeSet]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[CommitteeSets] DROP CONSTRAINT [FK_ProfessorSetCommitteeSet];
GO
IF OBJECT_ID(N'[dbo].[FK_ProfessorSetProjectSet]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[ProjectSets] DROP CONSTRAINT [FK_ProfessorSetProjectSet];
GO
IF OBJECT_ID(N'[dbo].[FK_ProjectSetStudentSet]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[StudentSets] DROP CONSTRAINT [FK_ProjectSetStudentSet];
GO
IF OBJECT_ID(N'[dbo].[FK_RequestProject]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[RequestSets] DROP CONSTRAINT [FK_RequestProject];
GO
IF OBJECT_ID(N'[dbo].[FK_SecretaryProfessor]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[ProfessorSets] DROP CONSTRAINT [FK_SecretaryProfessor];
GO
IF OBJECT_ID(N'[dbo].[FK_StudentRequest]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[RequestSets] DROP CONSTRAINT [FK_StudentRequest];
GO
IF OBJECT_ID(N'[dbo].[FK_StudentSetSecretarySet]', 'F') IS NOT NULL
    ALTER TABLE [dbo].[StudentSets] DROP CONSTRAINT [FK_StudentSetSecretarySet];
GO

-- --------------------------------------------------
-- Dropping existing tables
-- --------------------------------------------------
```

```
IF OBJECT_ID(N'[dbo].[AkEtosSet]', 'U') IS NOT NULL
    DROP TABLE [dbo].[AkEtosSet];
GO
IF OBJECT_ID(N'[dbo].[CommitteeSets]', 'U') IS NOT NULL
    DROP TABLE [dbo].[CommitteeSets];
GO
IF OBJECT_ID(N'[dbo].[ProfessorSets]', 'U') IS NOT NULL
    DROP TABLE [dbo].[ProfessorSets];
GO
IF OBJECT_ID(N'[dbo].[ProjectSets]', 'U') IS NOT NULL
    DROP TABLE [dbo].[ProjectSets];
GO
IF OBJECT_ID(N'[dbo].[RequestSets]', 'U') IS NOT NULL
    DROP TABLE [dbo].[RequestSets];
GO
IF OBJECT_ID(N'[dbo].[SecretarySets]', 'U') IS NOT NULL
    DROP TABLE [dbo].[SecretarySets];
GO
IF OBJECT_ID(N'[dbo].[StudentSets]', 'U') IS NOT NULL
    DROP TABLE [dbo].[StudentSets];
GO

-- --------------------------------------------------
-- Creating all tables
-- --------------------------------------------------

-- Creating table 'AkEtosSets'
CREATE TABLE [dbo].[AkEtosSets] (
    [Id] int IDENTITY(1,1) NOT NULL,
    [Date] nvarchar(max)  NOT NULL
);
GO

-- Creating table 'CommitteeSets'
CREATE TABLE [dbo].[CommitteeSets] (
    [Id] int IDENTITY(1,1) NOT NULL,
    [Vathmos1] decimal(18,0)  NULL,
    [Melos1] nvarchar(max)  NOT NULL,
    [Melos2] nvarchar(max)  NOT NULL,
    [Vatmos2] nvarchar(max)  NOT NULL,
    [Epivlepwn] nvarchar(max)  NOT NULL,
    [Vathmos] nvarchar(max)  NOT NULL,
    [ProfessorSets_Id] int  NULL
);
GO

-- Creating table 'ProfessorSets'
CREATE TABLE [dbo].[ProfessorSets] (
    [Id] int IDENTITY(1,1) NOT NULL,
    [Onomatepwnumo] nvarchar(max)  NOT NULL,
    [Email] nvarchar(max)  NOT NULL,
    [Secretary_Id] int  NULL
);
GO

-- Creating table 'ProjectSets'
```

```
CREATE TABLE [dbo].[ProjectSets] (
    [Id] int IDENTITY(1,1) NOT NULL,
    [Thema] nvarchar(max)  NOT NULL,
    [Perigrafi] nvarchar(max)  NOT NULL,
    [Vivliografia] nvarchar(max)  NOT NULL,
    [Gnwseis] nvarchar(max)  NULL,
    [Energo] bit  NULL,
    [CreateDate] datetime  NOT NULL,
    [ArFoititwn] smallint  NOT NULL,
    [Eksamino] nvarchar(max)  NOT NULL,
    [Diathesimo] bit  NOT NULL,
    [EndDate] datetime  NOT NULL,
    [Eidos] nvarchar(max)  NOT NULL,
    [AkEtos_Id] int  NOT NULL,
    [CommitteeSet_Id] int  NOT NULL,
    [ProfessorSet_Id] int  NULL,
    [Paratiriseis] nvarchar(max)  NULL
);
GO

-- Creating table 'RequestSets'
CREATE TABLE [dbo].[RequestSets] (
    [Id] int IDENTITY(1,1) NOT NULL,
    [RequestDate] datetime  NOT NULL,
    [Egkrisi] nvarchar(max)  NULL,
    [Student_Id] int  NULL,
    [Project_Id] int  NOT NULL
);
GO

-- Creating table 'SecretarySets'
CREATE TABLE [dbo].[SecretarySets] (
    [Id] int  NOT NULL
);
GO

-- Creating table 'StudentSets'
CREATE TABLE [dbo].[StudentSets] (
    [Id] int  NOT NULL,
    [Onomatepwnumo] nvarchar(max)  NOT NULL,
    [Email] nvarchar(max)  NOT NULL,
    [Idiotita] nvarchar(max)  NOT NULL,
    [Phone] nvarchar(max)  NOT NULL,
    [Cv] nvarchar(max)  NULL,
    [SecretarySet_Id] int  NULL,
    [ProjectSet_Id] int  NULL
);
GO

-- --------------------------------------------------
-- Creating all PRIMARY KEY constraints
-- --------------------------------------------------

-- Creating primary key on [Id] in table 'AkEtosSets'
ALTER TABLE [dbo].[AkEtosSets]
ADD CONSTRAINT [PK_AkEtosSets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
```

```
GO

-- Creating primary key on [Id] in table 'CommitteeSets'
ALTER TABLE [dbo].[CommitteeSets]
ADD CONSTRAINT [PK_CommitteeSets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
GO

-- Creating primary key on [Id] in table 'ProfessorSets'
ALTER TABLE [dbo].[ProfessorSets]
ADD CONSTRAINT [PK_ProfessorSets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
GO

-- Creating primary key on [Id] in table 'ProjectSets'
ALTER TABLE [dbo].[ProjectSets]
ADD CONSTRAINT [PK_ProjectSets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
GO

-- Creating primary key on [Id] in table 'RequestSets'
ALTER TABLE [dbo].[RequestSets]
ADD CONSTRAINT [PK_RequestSets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
GO

-- Creating primary key on [Id] in table 'SecretarySets'
ALTER TABLE [dbo].[SecretarySets]
ADD CONSTRAINT [PK_SecretarySets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
GO

-- Creating primary key on [Id] in table 'StudentSets'
ALTER TABLE [dbo].[StudentSets]
ADD CONSTRAINT [PK_StudentSets]
    PRIMARY KEY CLUSTERED ([Id] ASC);
GO

-- --------------------------------------------------
-- Creating all FOREIGN KEY constraints
-- --------------------------------------------------

-- Creating foreign key on [AkEtos_Id] in table 'ProjectSets'
ALTER TABLE [dbo].[ProjectSets]
ADD CONSTRAINT [FK_AkEtosProjectSet]
    FOREIGN KEY ([AkEtos_Id])
    REFERENCES [dbo].[AkEtosSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_AkEtosProjectSet'
CREATE INDEX [IX_FK_AkEtosProjectSet]
ON [dbo].[ProjectSets]
    ([AkEtos_Id]);
GO

-- Creating foreign key on [CommitteeSet_Id] in table 'ProjectSets'
```

```
ALTER TABLE [dbo].[ProjectSets]
ADD CONSTRAINT [FK_CommitteeSetProjectSet]
    FOREIGN KEY ([CommitteeSet_Id])
    REFERENCES [dbo].[CommitteeSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_CommitteeSetProjectSet'
CREATE INDEX [IX_FK_CommitteeSetProjectSet]
ON [dbo].[ProjectSets]
    ([CommitteeSet_Id]);
GO

-- Creating foreign key on [ProfessorSet_Id] in table 'ProjectSets'
ALTER TABLE [dbo].[ProjectSets]
ADD CONSTRAINT [FK_ProfessorSetProjectSet]
    FOREIGN KEY ([ProfessorSet_Id])
    REFERENCES [dbo].[ProfessorSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_ProfessorSetProjectSet'
CREATE INDEX [IX_FK_ProfessorSetProjectSet]
ON [dbo].[ProjectSets]
    ([ProfessorSet_Id]);
GO

-- Creating foreign key on [Secretary_Id] in table 'ProfessorSets'
ALTER TABLE [dbo].[ProfessorSets]
ADD CONSTRAINT [FK_SecretaryProfessor]
    FOREIGN KEY ([Secretary_Id])
    REFERENCES [dbo].[SecretarySets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_SecretaryProfessor'
CREATE INDEX [IX_FK_SecretaryProfessor]
ON [dbo].[ProfessorSets]
    ([Secretary_Id]);
GO

-- Creating foreign key on [ProjectSet_Id] in table 'StudentSets'
ALTER TABLE [dbo].[StudentSets]
ADD CONSTRAINT [FK_ProjectSetStudentSet]
    FOREIGN KEY ([ProjectSet_Id])
    REFERENCES [dbo].[ProjectSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_ProjectSetStudentSet'
CREATE INDEX [IX_FK_ProjectSetStudentSet]
ON [dbo].[StudentSets]
    ([ProjectSet_Id]);
GO

-- Creating foreign key on [Project_Id] in table 'RequestSets'
ALTER TABLE [dbo].[RequestSets]
```

```
ADD CONSTRAINT [FK_RequestProject]
    FOREIGN KEY ([Project_Id])
    REFERENCES [dbo].[ProjectSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_RequestProject'
CREATE INDEX [IX_FK_RequestProject]
ON [dbo].[RequestSets]
    ([Project_Id]);
GO

-- Creating foreign key on [Student_Id] in table 'RequestSets'
ALTER TABLE [dbo].[RequestSets]
ADD CONSTRAINT [FK_StudentRequest]
    FOREIGN KEY ([Student_Id])
    REFERENCES [dbo].[StudentSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_StudentRequest'
CREATE INDEX [IX_FK_StudentRequest]
ON [dbo].[RequestSets]
    ([Student_Id]);
GO

-- Creating foreign key on [SecretarySet_Id] in table 'StudentSets'
ALTER TABLE [dbo].[StudentSets]
ADD CONSTRAINT [FK_StudentSetSecretarySet]
    FOREIGN KEY ([SecretarySet_Id])
    REFERENCES [dbo].[SecretarySets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_StudentSetSecretarySet'
CREATE INDEX [IX_FK_StudentSetSecretarySet]
ON [dbo].[StudentSets]
    ([SecretarySet_Id]);
GO

-- Creating foreign key on [ProfessorSets_Id] in table 'CommitteeSets'
ALTER TABLE [dbo].[CommitteeSets]
ADD CONSTRAINT [FK_CommitteeSetProfessorSet]
    FOREIGN KEY ([ProfessorSets_Id])
    REFERENCES [dbo].[ProfessorSets]
        ([Id])
    ON DELETE NO ACTION ON UPDATE NO ACTION;

-- Creating non-clustered index for FOREIGN KEY 'FK_CommitteeSetProfessorSet'
CREATE INDEX [IX_FK_CommitteeSetProfessorSet]
ON [dbo].[CommitteeSets]
    ([ProfessorSets_Id]);
GO

-- --------------------------------------------------
-- Script has ended
-- --------------------------------------------------
```

Το site.css

```
html {
    background-color: #e2e2e2;
    margin: 0;
    padding: 0;
}

body {
    background-color: #fff;
    border-top: solid 10px #000;
    color: #333;
    font-size: .85em;
    font-family: "Segoe UI", Verdana, Helvetica, Sans-Serif;
    margin: 0;
    padding: 0;
}

a {
    color: #333;
    outline: none;
    padding-left: 3px;
    padding-right: 3px;
    text-decoration: underline;
}

    h2 a {
        text-decoration:none;
    }

    a:link, a:visited,
    a:active, a:hover {
        color: #333;
    }

    a:hover {
        background-color: #c7d1d6;
    }

header, footer, hgroup,
nav, section {
    display: block;
}

mark {
    background-color: #a6dbed;
    padding-left: 5px;
    padding-right: 5px;
}

.float-left {
    float: left;
}
```

```
.float-right {
    float: right;
}

.clear-fix:after {
    content: ".";
    clear: both;
    display: block;
    height: 0;
    visibility: hidden;
}

h1, h2, h3,
h4, h5, h6 {
    color: #000;
    margin-bottom: 0;
    padding-bottom: 0;
}

h1 {
    font-size: 2em;
}

h2 {
    font-size: 1.75em;
}

h3 {
    font-size: 1.2em;
}

h4 {
    font-size: 1.1em;
}

h5, h6 {
    font-size: 1em;
}

    h5 a:link, h5 a:visited, h5 a:active {
        padding: 0;
        text-decoration: none;
    }


/* main layout
-----------------------------------------------------------*/
.content-wrapper {
    margin: 0 auto;
    max-width: 960px;
}

#body {
    background-color: #efeeef;
    clear: both;
    padding-bottom: 35px;
}
```

```
    .main-content {
        background: url("../Images/accent.png") no-repeat;
        padding-left: 10px;
        padding-top: 30px;
    }

    .featured + .main-content {
        background: url("../Images/heroAccent.png") no-repeat;
    }

header .content-wrapper {
    padding-top: 20px;
}

footer {
    clear: both;
    background-color: #e2e2e2;
    font-size: .8em;
    height: 100px;
}


/* site title
-----------------------------------------------------------*/
.site-title {
    color: #c8c8c8;
    font-family: Rockwell, Consolas, "Courier New", Courier, monospace;
    font-size: 2.3em;
    margin: 0;
}

.site-title a, .site-title a:hover, .site-title a:active {
    background: none;
    color: #c8c8c8;
    outline: none;
    text-decoration: none;
}


/* login
-----------------------------------------------------------*/
#login {
    display: block;
    font-size: .85em;
    margin: 0 0 10px;
    text-align: right;
}

    #login a {
        background-color: #d3dce0;
        margin-left: 10px;
        margin-right: 3px;
        padding: 2px 3px;
        text-decoration: none;
    }
```

```
    #login a.username {
        background: none;
        margin: 0;
        padding: 0;
        text-decoration: underline;
    }

    #login ul {
        margin: 0;
    }

    #login li {
        display: inline;
        list-style: none;
    }


/* menu
-----------------------------------------------------------*/
ul#menu {
    font-size: 1.3em;
    font-weight: 600;
    margin: 0 0 5px;
    padding: 0;
    text-align: right;
}

    ul#menu li {
        display: inline;
        list-style: none;
        padding-left: 15px;
    }

        ul#menu li a {
            background: none;
            color: #999;
            text-decoration: none;
        }

        ul#menu li a:hover {
            color: #333;
            text-decoration: none;
        }


/* page elements
-----------------------------------------------------------*/
/* featured */
.featured {
    background-color: #fff;
}

    .featured .content-wrapper {
        background-color: #7ac0da;
        background-image: -ms-linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
        background-image: -o-linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
```

```
        background-image: -webkit-gradient(linear, left top, right top, color-stop(0,
#7ac0da), color-stop(1, #a4d4e6));
        background-image: -webkit-linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
        background-image: linear-gradient(left, #7ac0da 0%, #a4d4e6 100%);
        color: #3e5667;
        padding: 20px 40px 30px 40px;
    }

        .featured hgroup.title h1, .featured hgroup.title h2 {
            color: #fff;
        }

        .featured p {
            font-size: 1.1em;
        }

/* page titles */
hgroup.title {
    margin-bottom: auto;
    margin-top:auto;
}

hgroup.title h1, hgroup.title h2 {
    display:block;
}

hgroup.title h2 {
    font-weight: normal;
    margin-left: 3px;
    margin-top:auto;
}

/* features */
section.feature {
    width: 300px;
    float: left;
    padding: 10px;
}

/* ordered list */
ol.round {
    list-style-type: none;
    padding-left: 0;
}

    ol.round li {
        margin: 25px 0;
        padding-left: 45px;
    }

        ol.round li.zero {
            background: url("../Images/orderedList0.png") no-repeat;
        }

        ol.round li.one {
            background: url("../Images/orderedList1.png") no-repeat;
        }
```

```
        ol.round li.two {
            background: url("../Images/orderedList2.png") no-repeat;
        }

        ol.round li.three {
            background: url("../Images/orderedList3.png") no-repeat;
        }

        ol.round li.four {
            background: url("../Images/orderedList4.png") no-repeat;
        }

        ol.round li.five {
            background: url("../Images/orderedList5.png") no-repeat;
        }

        ol.round li.six {
            background: url("../Images/orderedList6.png") no-repeat;
        }

        ol.round li.seven {
            background: url("../Images/orderedList7.png") no-repeat;
        }

        ol.round li.eight {
            background: url("../Images/orderedList8.png") no-repeat;
        }

        ol.round li.nine {
            background: url("../Images/orderedList9.png") no-repeat;
        }

/* content */
article {
    float: left;
    width: 70%;
}

aside {
    float: right;
    width: 25%;
}

    aside ul {
        list-style: none;
        padding: 0;
    }

        aside ul li {
            background: url("../Images/bullet.png") no-repeat 0 50%;
            padding: 2px 0 2px 20px;
        }

.label {
    font-weight: 700;
}
```

```
/* login page */
#loginForm {
    border-right: solid 2px #c8c8c8;
    float: left;
    width: 55%;
}

    #loginForm .validation-error {
        display: block;
        margin-left: 15px;
    }

    #loginForm .validation-summary-errors ul {
        margin: 0;
        padding: 0;
    }

    #loginForm .validation-summary-errors li {
        display: inline;
        list-style: none;
        margin: 0;
    }

    #loginForm input {
        width: 250px;
    }

    #loginForm input[type="checkbox"],
    #loginForm input[type="submit"],
    #loginForm input[type="button"],
    #loginForm button {
        width: auto;
    }

#socialLoginForm {
    margin-left: 40px;
    float: left;
    width: 40%;
}

    #socialLoginForm h2 {
        margin-bottom:  5px;
    }

#socialLoginList button {
    margin-bottom: 12px;
}

#logoutForm {
    display: inline;
}

/* contact */
.contact h3 {
    font-size: 1.2em;
}
```

```
.contact p {
    margin: 5px 0 0 10px;
}

.contact iframe {
    border: 1px solid #333;
    margin: 5px 0 0 10px;
}

/* forms */
fieldset {
    border: none;
    margin: 0;
    padding: 0;
}

    fieldset legend {
        display: none;
    }

    fieldset ol {
        padding: 0;
        list-style: none;
    }

        fieldset ol li {
            padding-bottom: 5px;
        }

label {
    display: block;
    font-size: 1.2em;
    font-weight: 600;
}

label.checkbox {
    display: inline;
}

input, textarea {
    border: 1px solid #e2e2e2;
    background: #fff;
    color: #333;
    font-size: 1.2em;
    margin: 5px 0 6px 0;
    padding: 5px;
    width: 300px;
}

textarea {
    font-family: inherit;
    width: 500px;
}

    input:focus, textarea:focus {
        border: 1px solid #7ac0da;
```

```
    }

    input[type="checkbox"] {
        background: transparent;
        border: inherit;
        width: auto;
    }

    input[type="submit"],
    input[type="button"],
    button {
        background-color: #d3dce0;
        border: 1px solid #787878;
        cursor: pointer;
        font-size: 1.2em;
        font-weight: 600;
        padding: 7px;
        margin-right: 8px;
        width: auto;
    }

    td input[type="submit"],
    td input[type="button"],
    td button {
        font-size: 1em;
        padding: 4px;
        margin-right: 4px;
    }

/* info and errors */
.message-info {
    border: 1px solid;
    clear: both;
    padding: 10px 20px;
}

.message-error {
    clear: both;
    color: #e80c4d;
    font-size: 1.1em;
    font-weight: bold;
    margin: 20px 0 10px 0;
}

.message-success {
    color: #7ac0da;
    font-size: 1.3em;
    font-weight: bold;
    margin: 20px 0 10px 0;
}

.error {
    color: #e80c4d;
}

/* styles for validation helpers */
.field-validation-error {
```

```
    color: #e80c4d;
    font-weight: bold;
}

.field-validation-valid {
    display: none;
}

input.input-validation-error {
    border: 1px solid #e80c4d;
}

input[type="checkbox"].input-validation-error {
    border: 0 none;
}

.validation-summary-errors {
    color: #e80c4d;
    font-weight: bold;
    font-size: 1.1em;
}

.validation-summary-valid {
    display: none;
}


/* tables
------------------------------------------------------------*/
table {
    border-collapse: collapse;
    border-spacing: 0;
    margin-top: 0.75em;
    border: 0 none;
}

th {
    font-size: 1.2em;
    text-align: left;
    border: none 0px;
    padding-left: 0;
}

    th a {
        display: block;
        position: relative;
    }

    th a:link, th a:visited, th a:active, th a:hover {
        color: #333;
        font-weight: 600;
        text-decoration: none;
        padding: 0;
    }

    th a:hover {
        color: #000;
```

```
    }

    th.asc a, th.desc a {
        margin-right: .75em;
    }

    th.asc a:after, th.desc a:after {
        display: block;
        position: absolute;
        right: 0em;
        top: 0;
        font-size: 0.75em;
    }

    th.asc a:after {
        content: '▲';
    }

    th.desc a:after {
        content: '▼';
    }

td {
    padding: 0.25em 2em 0.25em 0em;
    border: 0 none;
}

tr.pager td {
    padding: 0 0.25em 0 0;
}


/********************
*   Mobile Styles   *
********************/
@media only screen and (max-width: 850px) {

    /* header
    ------------------------------------------------------------*/
    header .float-left,
    header .float-right {
        float: none;
    }

    /* logo */
    header .site-title {
        margin: 10px;
        text-align: center;
    }

    /* login */
    #login {
        font-size: .85em;
        margin: 0 0 12px;
        text-align: center;
    }
```

```
    #login ul {
        margin: 5px 0;
        padding: 0;
    }

    #login li {
        display: inline;
        list-style: none;
        margin: 0;
        padding: 0;
    }

    #login a {
        background: none;
        color: #999;
        font-weight: 600;
        margin: 2px;
        padding: 0;
    }

    #login a:hover {
        color: #333;
    }

/* menu */
nav {
    margin-bottom: 5px;
}

ul#menu {
    margin: 0;
    padding: 0;
    text-align: center;
}

    ul#menu li {
        margin: 0;
        padding: 0;
    }


/* main layout
-----------------------------------------------------------*/
.main-content,
.featured + .main-content {
    background-position: 10px 0;
}

.content-wrapper {
    padding-right: 10px;
    padding-left: 10px;
}

.featured .content-wrapper {
    padding: 10px;
}
```

```
/* page content */
article, aside {
    float: none;
    width: 100%;
}

/* ordered list */
ol.round {
    list-style-type: none;
    padding-left: 0;
}

    ol.round li {
        padding-left: 10px;
        margin: 25px 0;
    }

        ol.round li.zero,
        ol.round li.one,
        ol.round li.two,
        ol.round li.three,
        ol.round li.four,
        ol.round li.five,
        ol.round li.six,
        ol.round li.seven,
        ol.round li.eight,
        ol.round li.nine {
            background: none;
        }

 /* features */
 section.feature {
    float: none;
    padding: 10px;
    width: auto;
 }

    section.feature img {
        color: #999;
        content: attr(alt);
        font-size: 1.5em;
        font-weight: 600;
    }

/* forms */
input {
    width: 90%;
}

/* login page */
#loginForm {
    border-right: none;
    float: none;
    width: auto;
}

    #loginForm .validation-error {
```

```
            display: block;
            margin-left: 15px;
        }

    #socialLoginForm {
        margin-left: 0;
        float: none;
        width: auto;
    }


    /* footer
    ---------------------------------------------------------*/
    footer .float-left,
    footer .float-right {
        float: none;
    }

    footer {
        text-align: center;
        height: auto;
        padding: 10px 0;
    }

        footer p {
            margin: 0;
        }
}

#navcontainer ul
{
margin: 0;
padding: 0;
list-style-type: none;
text-align: center;
}

#navcontainer ul li { display: inline; }

#navcontainer ul li a
{
text-decoration: none;
padding: .2em 1em;
color: #fff;
background-color: #036;
}

#navcontainer ul li a:hover
{
color: #fff;
background-color: #369;
}
```