

ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ

ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ / ΜΕΣΟΛΟΓΓΙ

Πτυχιακή εργασία

ΜΕΛΕΤΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ CLOUD ΕΦΑΡΜΟΓΗΣ
ΓΙΑ ΤΗΝ ΔΙΑΧΕΙΡΙΣΗ ΠΤΥΧΙΑΚΩΝ ΕΡΓΑΣΙΩΝ

Αχιλλέας Δρακάτος Α.Μ:14280

Επιβλέπων καθηγητής
Βασίλειος Στεφανής

Μεσολόγγι 2016-2017

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Διοίκησης Επιχειρήσεων/Μεσολογγίου του ΤΕΙ Δυτικής Ελλάδας δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

ΠΕΡΙΛΗΨΗ

Στο πλαίσιο της πτυχιακής εργασίας αναπτύχθηκε μια διαδικτυακή εφαρμογή με σκοπό οι χρήστες να διαχειρίζονται τις πτυχιακές εργασίες . Συγκεκριμένα, η εφαρμογή ονομάζεται "Διαχείριση Πτυχιακών Εργασιών" και έχει σαν στόχο να απλοποιήσει την διαδικασία δήλωσης πτυχιακής εργασίας, σε σχέση με τον τρόπο που ξέρουμε μέχρι σήμερα, καθώς γίνεται προσπάθεια να καλυφθεί η διαδικασία αυτή εξολοκλήρου ηλεκτρονικά. Προσπάθεια γίνεται να καλυφθούν όλα τα στάδια κατά τη διαδικασία αποπεράτωσης μια πτυχιακής εργασίας, από το στάδιο της ανάθεσης θεμάτων πτυχιακών από τους καθηγητές και την δήλωση αυτών από τους φοιτητές, μέχρι το τελικό στάδιο της βαθμολόγησης της πτυχιακής .

Η διαδικτυακή αυτή εφαρμογή, αναπτύχθηκε χρησιμοποιώντας ως γλώσσα προγραμματισμού την Visual Basic, ενώ για την προβολή του περιεχομένου χρησιμοποιήθηκε η γλώσσα σήμανσης υπερκειμένου HTML. Απαραίτητη προϋπόθεση για την λειτουργία μιας τέτοιας εφαρμογής , είναι η ύπαρξη μιας βάσης δεδομένων για την διαχείριση των δεδομένων μας. Συγκεκριμένα, χρησιμοποιήθηκε η Microsoft SQL.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ.....	iii
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	iv
ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ	vii
ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ	xi
ΕΙΣΑΓΩΓΗ.....	xii
1 Cloud Computing	1
1.1 Εισαγωγή στο Cloud Computing.....	1
1.2 Βασικά Χαρακτηριστικά	2
1.3 Πλεονεκτήματα	3
1.4 Μειονεκτήματα.....	4
2 Συστήματα Διαχείρισης Έργων.....	6
2.1 Σύστημα Διαχείρισης Έργου Bitrix24.....	6
2.2 Σύστημα Διαχείρισης Έργου OrangeScrum.....	8
2.3 Σύστημα Διαχείρισης Έργου Asana	9
3 Γλώσσες Προγραμματισμού	11
3.1 Visual Basic.....	11
3.2 Javascript	12
3.3 HTML.....	12
3.4 CSS	13
4 Εγκατάσταση Εργαλείων Ανάπτυξης Εφαρμογής.....	14
4.1 Εγκατάσταση Microsoft Visual Studio	14
4.2 Εγκατάσταση Microsoft SQL SERVER 2012	18
5 Βάση Δεδομένων.....	26
5.1 Περιγραφή της Βάσης Δεδομένων	26

5.2	Βάση Δεδομένων και Πίνακες.....	26
5.3	Σχεσιακό Διάγραμμα ER Βάσης Δεδομένων	33
6	Βασική Δομή Εφαρμογής	35
6.1	Δημιουργία Project.....	35
6.2	Master Page	36
6.2.1	Δημιουργία Master Page	37
6.3	Content Page.....	39
6.3.1	Δημιουργία Content Page.....	39
6.4	Web Form.....	43
6.4.1	Δημιουργία Web Form	43
6.5	User Control	45
6.5.1	Δημιουργία User Control.....	46
6.6	ASP.NET Controls	48
6.6.1	Βασικά Web Server Controls	52
6.6.2	Validation Server Controls	54
6.6.3	Data Source Controls.....	55
7	Υλοποίηση Οθονών Εφαρμογής	56
7.1	Οθόνη - Εγγραφή Φοιτητών	56
7.2	Οθόνη - Εγγραφή Καθηγητών.....	64
7.3	Οθόνη - Είσοδος Χρηστών.....	66
7.4	Οθόνη - Τα Μηνύματα Μου.....	72
7.5	Οθόνη - Ολοκληρωμένες Πτυχιακές Εργασίες	74
7.6	Οθόνη - Διαχείριση πτυχιακών εργασιών	77
7.7	Οθόνη - Εκδήλωση Ενδιαφέροντος Πτυχιακής εργασίας.....	84
7.8	Οθόνη - Πορεία Πτυχιακής	93
	ΣΥΜΠΕΡΑΣΜΑΤΑ/ ΑΠΟΤΕΛΕΣΜΑΤΑ/ ΕΠΙΛΟΓΟΣ.....	95

ΒΙΒΛΙΟΓΡΑΦΙΑ / ΔΙΑΔΙΚΤΥΟ..... 96

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1.1.1 : Υπολογιστικό Νέφος.....	1
Εικόνα 2.1.1 : Σύστημα Διαχείρισης Έργου Bitrix 24.....	7
Εικόνα 2.2.1 : Σύστημα Διαχείρισης Έργου OrangeScrum	8
Εικόνα 2.3.1 : Σύστημα Διαχείρισης Έργου Asana	9
Εικόνα 4.1.1 : Εγκατάσταση Visual Studio.....	14
Εικόνα 4.1.2 : Εγκατάσταση Visual Studio.....	15
Εικόνα 4.1.3 : Εγκατάσταση Visual Studio.....	15
Εικόνα 4.1.4 : Εγκατάσταση Visual Studio.....	16
Εικόνα 4.1.5 : Εγκατάσταση Visual Studio.....	16
Εικόνα 4.1.6 : Εγκατάσταση Visual Studio.....	17
Εικόνα 4.1.7 : Εγκατάσταση Visual Studio.....	17
Εικόνα 4.2.1 : Εγκατάσταση SQL SERVER.....	18
Εικόνα 4.2.2 : Εγκατάσταση SQL SERVER.....	19
Εικόνα 4.2.3 : Εγκατάσταση SQL SERVER.....	19
Εικόνα 4.2.4 : Εγκατάσταση SQL SERVER.....	20
Εικόνα 4.2.5 : Εγκατάσταση SQL SERVER.....	20
Εικόνα 4.2.6 : Εγκατάσταση SQL SERVER.....	20
Εικόνα 4.2.7 : Εγκατάσταση SQL SERVER.....	21
Εικόνα 4.2.7 : Εγκατάσταση SQL SERVER.....	21
Εικόνα 4.2.8 : Εγκατάσταση SQL SERVER.....	21
Εικόνα 4.2.9 : Εγκατάσταση SQL SERVER.....	22
Εικόνα 4.2.10 : Εγκατάσταση SQL SERVER.....	22
Εικόνα 4.2.11 : Εγκατάσταση SQL SERVER.....	23
Εικόνα 4.2.12 : Εγκατάσταση SQL SERVER.....	23

Εικόνα 4.2.13 : Εγκατάσταση SQL SERVER.....	24
Εικόνα 4.2.14 : Εγκατάσταση SQL SERVER.....	24
Εικόνα 4.2.15 : Εγκατάσταση SQL SERVER.....	25
Εικόνα 4.2.16 : Εγκατάσταση SQL SERVER.....	25
Εικόνα 5.2.1 : Πίνακας register	27
Εικόνα 5.2.2 : Πίνακας graduate_themes	28
Εικόνα 5.2.3 : Πίνακας graduate_requests	29
Εικόνα 5.2.4 : Πίνακας graduate_completed.....	30
Εικόνα 5.2.5 : Πίνακας messages	32
Εικόνα 5.3.1 : Διάγραμμα ER	34
Εικόνα 6.1.1 : Δημιουργία Project	35
Εικόνα 6.2.1 : Content Page	37
Εικόνα 6.2.1.1 : Δημιουργία Master Page	38
Εικόνα 6.2.1.2 : Δημιουργία Master Page	39
Εικόνα 6.3.1.1 : Δημιουργία Content Page	40
Εικόνα 6.3.1.2 : Δημιουργία Content Page	40
Εικόνα 6.3.1.3 : Δημιουργία Content Page	41
Εικόνα 6.3.1.4 : Content Page Σελίδα Παράδειγμα	42
Εικόνα 6.3.1.5 : Συνδυασμός Master Page - Content Page.....	43
Εικόνα 6.4.1.1 : Δημιουργία Web Form.....	44
Εικόνα 6.4.1.2 : Δημιουργία Web Form.....	45
Εικόνα 6.5.1 : Παράδειγμα User Control	46
Εικόνα 6.5.1.1 : Δημιουργία User Control	47
Εικόνα 6.5.1.2 : Δημιουργία User Control	48
Εικόνα 6.6.1 : Εργαλειοθήκη Visual Studio.....	49

Εικόνα 6.6.2 : Δημιουργία event OnClick.....	51
Εικόνα 6.6.3 : Δημιουργία event OnClick.....	51
Εικόνα 6.6.4 : Κλάση κουμπιού	51
Εικόνα 6.6.5 : Οθόνη WebForm1.aspx	52
Εικόνα 7.1.1 : Οθόνη Εγγραφής Φοιτητών register.aspx.....	56
Εικόνα 7.1.2 : Μήνυμα Επιτυχημένης Εγγραφής.....	57
Εικόνα 7.2.1 : Μήνυμα Επιτυχημένης Εγγραφής.....	65
Εικόνα 7.2.2 : Μήνυμα Αποτυχημένης Εγγραφής	65
Εικόνα 7.3.1 : Οθόνη Εισόδου Χρηστών default.aspx.....	66
Εικόνα 7.3.1 : Επιτυχημένη Είσοδος Χρήστη	67
Εικόνα 7.3.2 : Μήνυμα Αποτυχημένης Εισόδου.....	67
Εικόνα 7.4.1 : Οθόνη Μηνυμάτων messages.aspx.....	73
Εικόνα 7.4.2 : Λειτουργία Καταμέτρησης Μηνυμάτων.....	73
Εικόνα 7.4.2 : User Control Μηνυμάτων Messages.ascx	74
Εικόνα 7.5.1 : Οθόνη Ολοκληρωμένες Πτυχιακές Εργασίες intergrated.aspx	75
Εικόνα 7.5.2 : Λειτουργία Στατιστικά Στοιχεία	76
Εικόνα 7.5.3 : Λειτουργία Αναζήτησης	77
Εικόνα 7.6.1 : Οθόνη Διαχείρισης Πτυχιακών Εργασιών.....	78
Εικόνα 7.6.2 : Λειτουργία-Πτυχιακές Εργασίες.....	78
Εικόνα 7.6.3 : User Control - Νέα καταχώριση	79
Εικόνα 7.6.4 : User Control - Μήνυμα Αποτυχίας.....	80
Εικόνα 7.6.5 : User Control - Προβολή Στοιχείων.....	81
Εικόνα 7.6.6 : User Control - Επεξεργασίας Στοιχείων	82
Εικόνα 7.6.7 : Λειτουργία Διαχείρισης Αιτημάτων Πτυχιακών Εργασιών	82
Εικόνα 7.6.8 : Λειτουργία Απορριφθέντων Αιτημάτων Πτυχιακών Εργασιών.....	83
Εικόνα 7.6.9 : Λειτουργία Ανατεθειμένων Πτυχιακών Εργασιών.....	83

Εικόνα 7.6.10 : Λειτουργία Ολοκληρωμένων Πτυχιακών Εργασιών	84
Εικόνα 7.7.1 : Οθόνη Εκδήλωσης Ενδιαφέροντος Πτυχιακής Εργασίας	85
Εικόνα 7.7.2 : Λειτουργία Πτυχιακές Εργασίες Διαθέσιμες	85
Εικόνα 7.7.3 : Μήνυμα Αποτυχίας Δήλωσης Πτυχιακής Εργασίας	86
Εικόνα 7.7.4 : Λειτουργία Εκδήλωσης Ενδιαφέροντος	86
Εικόνα 7.8.1 : Οθόνη Πορεία Πτυχιακής	93

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 6.6.1 : Ιδιότητες Control	50
Πίνακας 6.6.6.1 : Server Controls	54
Πίνακας 6.6.2.1 : Validation Controls	55
Πίνακας 6.6.3.1 : Data Source Controls	55

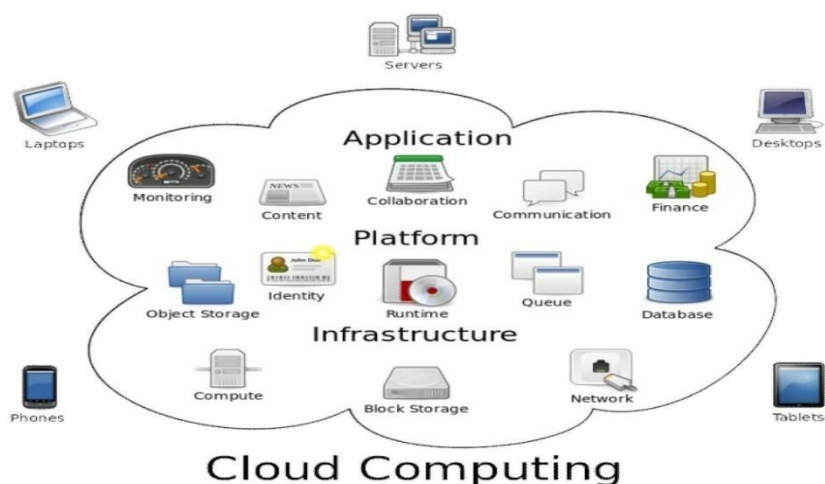
ΕΙΣΑΓΩΓΗ

Κύριο αντικείμενο της πτυχιακής εργασίας είναι η ανάπτυξη μια διαδικτυακής εφαρμογής σχετικά με την διαχείριση πτυχιακών εργασιών. Θα γίνει εκτενής αναφορά σε παρόμοια συστήματα διαχείρισης έργου, καθώς και στις τεχνολογίες που χρησιμοποιήθηκαν για την αποπεράτωση της εφαρμογής. Σκοπός είναι να κατασκευαστεί η διαδικτυακή αυτή εφαρμογή με τέτοιο τρόπο, ώστε η χρήση της να είναι φιλική και εύκολη στους χρήστες.

1 Cloud Computing

1.1 Εισαγωγή στο Cloud Computing

Το cloud computing (Υπολογιστικό Νέφος) αποτελεί ένα νέο επιχειρηματικό μοντέλο παροχής υπηρεσιών το οποίο μελετάται από επιστήμονες που ασχολούνται με την Τεχνολογία Πληροφοριών (Information Technology) και πρόκειται να απασχολήσει τα επόμενα χρόνια σε μεγάλο βαθμό. Είναι δυνατόν να έχουμε πρόσβαση σε πληροφορίες και δεδομένα από οπουδήποτε με την μοναδική προϋπόθεση να υπάρχει πρόσβαση στο διαδίκτυο και όλα αυτά χωρίς να χρειάζεται να αγορασθεί και να εγκατασταθεί ο απαραίτητος εξοπλισμός. Η λειτουργία του μοιάζει αρκετά με την λειτουργία ενός δικτύου ηλεκτρικής ενέργειας, δηλαδή χρησιμοποιούμε ηλεκτρικό ρεύμα χωρίς να γνωρίζουμε την υποδομή και τον εξοπλισμό που απαιτείται για την παροχή της υπηρεσίας αυτής. Πρόκειται για ένα μοντέλο εύκολης πρόσβασης σε απομακρυσμένα sites η οποία γίνεται μέσω του διαδικτύου. Με απλά λόγια, αντί να διατηρεί ο χρήστης το δικό του περιβάλλον υλικού και λογισμικού, το νέφος παρέχει τους υπολογιστικούς πόρους ανάλογα με τις απαιτήσεις του χρήστη. Με αυτόν τον τρόπο, ο χρήστης πληρώνει ανάλογα με τις υπηρεσίες που χρησιμοποιεί χωρίς να έχει ιδιαίτερες γνώσεις σχετικά με τις τεχνολογίες που χρησιμοποιούνται στο σύστημα, αφού η διαχείριση γίνεται από τον πάροχο των υπηρεσιών αυτών. Η ανάγκη επέκτασης τέτοιων συστημάτων αποτελεί μονόδρομο αν λάβουμε υπόψη τις δυνατότητες που προσφέρονται, καθότι μπορούμε να μειώσουμε σημαντικά τα έξοδα που σχετίζονται με την διαχείριση πόρων υλικού και λογισμικού.



Εικόνα 1.1.1 : Υπολογιστικό Νέφος

1.2 Βασικά Χαρακτηριστικά

Τα πέντε βασικά λειτουργικά χαρακτηριστικά του cloud computing είναι τα εξής:

On-demand self-service: Παρέχονται στον καταναλωτή μονομερώς υπολογιστικοί πόροι, όπως ο χρόνος χρήσης του server και το μέγεθος του αποθηκευτικού χώρου, οι οποίοι δεσμεύονται μέσω του δικτύου αυτόματα ανάλογα με τις ανάγκες του χωρίς να απαιτείται ανθρώπινη αλληλεπίδραση με τον φορέα παροχής κάθε υπηρεσίας.

Broad network access: Οι υπολογιστικοί πόροι είναι διαθέσιμοι μέσω του δικτύου και προσβάσιμοι μέσω των τυποποιημένων μηχανισμών που προωθούν τη χρήση ετερογενών τερματικών συσκευών στην πλευρά του τελικού χρήστη, όπως κινητά τηλέφωνα, φορητούς υπολογιστές, PDAs.

Resource pooling: Οι υπολογιστικοί πόροι του παρόχου χρησιμοποιούνται για να εξυπηρετήσουν παράλληλα πολλούς καταναλωτές, ακολουθώντας το μοντέλο πολλαπλών εκμισθωτών (multi-tenant), με τους διάφορους φυσικούς και εικονικούς πόρους να ανατίθενται δυναμικά και εκ νέου ανάλογα με τη ζήτηση των καταναλωτών. Δίνεται μια αίσθηση ανεξαρτησίας όσον αφορά τη φυσική τοποθεσία, καθώς ο καταναλωτής δεν έχει κανέναν έλεγχο ή γνώση σχετικά με την ακριβή τοποθεσία των παρεχόμενων πόρων, αλλά μπορεί να είναι σε θέση να προσδιορίσει σε ένα πιο αφηρημένο επίπεδο την τοποθεσία, όπως χώρα, περιοχή, πόλη ή data center. Παραδείγματα παρεχόμενων πόρων αποτελούν η επεξεργαστική ισχύς, η μνήμη, το εύρος ζώνης (bandwidth) του δικτύου και οι εικονικές μηχανές (virtual machines).

Rapid elasticity: Οι πόροι μπορούν να δεσμευτούν προς χρήση γρήγορα και με ευέλικτο τρόπο, σχεδόν άμεσα και αυτόματα, έτσι ώστε να εμφανίζονται στον καταναλωτή ως απεριόριστοι και διαθέσιμοι προς αγορά ανά πάσα στιγμή.

Measured service: Τα συστήματα cloud ελέγχουν και βελτιστοποιούν αυτόματα τη χρήση των υπολογιστικών πόρων αξιοποιώντας μηχανισμούς μέτρησης ανάλογα με τον τύπο της προσφερόμενης υπηρεσίας. Η χρήση των πόρων μπορεί να παρακολουθείται, ακόμα μπορεί να ελέγχεται αλλά και να καταγράφεται παρέχοντας διαφάνεια και στις δύο πλευρές, τόσο του καταναλωτή-χρήστη όσο και του παρόχου της υπηρεσίας.

1.3 Πλεονεκτήματα

Εξοικονόμηση κόστους : Ίσως το σημαντικότερο όφελος του cloud computing να είναι η εξοικονόμηση κόστους. Δεν έχει σημασία το μέγεθος της επιχείρησης όλες οι επιχειρήσεις έχουν σαν στόχο να κερδίσουν χρήματα διατηρώντας παράλληλα τα λειτουργικά έξοδα στο ελάχιστο. Χρησιμοποιώντας cloud υπηρεσίες εξοικονομείται σημαντικό κόστος από το κεφάλαιό μας, καθώς δεν χρειάζεται να αγοράσουμε και να στεγάσουμε κάποιον server στην επιχείρησή μας. Με το να μην στεγάζουμε κάποιον server στην επιχείρησή μας εξοικονομούνται έξτρα χρήματα από κόστη ρεύματος και συντήρησης. Ακόμα πληρώνουμε για της υπηρεσίες που χρησιμοποιούνται χωρίς να χρειάζεται να ανησυχούμε για κεφάλαια που έχουμε επενδύσει καθώς μπορούμε πολύ εύκολα να αποδεσμευτούμε από τις υπηρεσίες αυτές όποτε θέλουμε. Είναι μια κοινή παρερμηνεία ότι μόνο οι μεγάλες επιχειρήσεις μπορούν να αντέξουν οικονομικά το κόστος αυτών των υπηρεσιών , οι υπηρεσίες αυτές στην πραγματικότητα είναι εξαιρετικά προσιτές για τις μικρότερες επιχειρήσεις.

Αξιοπιστία : Η χρήση υποδομών που χρησιμοποιούνται από τους πάροχους των cloud υπηρεσιών είναι πολύ πιο αξιόπιστες και συνεπείς από αυτές που θα χρειαζόταν να αποκτήσουν οι χρήστες. Οι πάροχοι αυτών των υπηρεσιών εγγυούνται εικοσιτέσσερις ώρες το εικοσιτετράωρο επτά μέρες την εβδομάδα και τριακόσιες εξήντα πέντε μέρες των χρόνων διαθεσιμότητα των υπηρεσιών αυτών και σε ποσοστό που αγγίζει το 99,99 %. Οι πελάτες των υπηρεσιών αυτών μπορούν να εξοικονομήσουν πόρους καθώς οι πάροχοι διαθέτουν γρήγορους μηχανισμούς ανακατεύθυνσης. Εάν ένας μηχανισμός διακοσμητή αποτύχει μπορούν να μεταφερθούν τα δεδομένα μας εύκολα σε άλλον διαθέσιμο server.

Ευκολία στην χρήση : Οι πάροχοι των cloud υπηρεσιών παρέχουν βελτιωμένα και απλοποιημένα διαχειρίσιμα των πόρων. Οι ενημερώσεις καθώς και οι διαδικασίες συντήρησης έχουν εξαλειφθεί. Μπορεί πλέον ο πελάτης να έχει πρόσβαση στις υπηρεσίες από ένα απλοποιημένο περιβάλλον. Οι πάροχοι με ένα συμφωνητικό παροχής υπηρεσιών (Service Level Agreement) εξασφαλίζουν εγγύηση διαθεσιμότητας των δικτύων στους πελάτες τους καθώς και την συντήρηση αυτών .

Στρατηγικό πλεονέκτημα : Χρησιμοποιώντας cloud υπηρεσίες από παρόχους δεν χρειάζεται να ασχοληθούμε με την αγορά και την προμήθεια υπολογιστικών πόρων. Έτσι συγκριτικά με τους ανταγωνιστές μας αποκτάμε στρατηγικό πλεονέκτημα. Η εταιρεία μπορεί

να επικεντρωθεί στις επιχειρηματικές της δραστηριότητες μειώνοντας τον χρόνο και το κόστος που θα χρειαζόταν για την απόκτηση εξοπλισμού.

1.4 Μειονεκτήματα

Προσωρινή Αναστολή : Οι πάροχοι των cloud υπηρεσιών ασχολούνται κάθε μέρα με διάφορους πελάτες και μπορεί να έρθουν αντιμέτωποι με τεχνικά θέματα. Αυτό μπορεί να έχει σαν αποτέλεσμα την προσωρινή αναστολή και διακοπή αυτών των υπηρεσιών. Επιπλέον αν οι πελάτες αν έχουν κάποιο πρόβλημα με την σύνδεση στο δίκτυο, δεν θα είναι σε θέση να διαχειριστούν καμία από τις υπηρεσίες των παρόχων cloud καθώς δεν θα έχουν πρόσβαση και στα δεδομένα που θέλουν να διαχειριστούν.

Ασφάλεια : Παρόλο που οι πάροχοι των cloud υπηρεσιών εφαρμόζουν τα καλύτερα πρότυπα ασφαλείας και είναι πιστοποιημένοι , το να έχεις σημαντικά δεδομένα σε εξωτερικό πάροχο πάντα υπάρχει ρίσκο για την ασφάλεια των δεδομένων αυτών. Το να στεγάζονται τα δεδομένα αυτά σε server σημαίνει πως έχουν πολλοί την δυνατότητα να εντοπίζουν τις αδυναμίες και τα κενά ασφαλείας σε ένα σύστημα. Για παράδειγμα, σε μια αρχιτεκτονική cloud που φιλοξενούνται στον ίδιο server πολλοί χρήστες ένας hacker μπορεί να προσπαθήσει να σπάσει τα δεδομένα των χρηστών που φιλοξενούνται και αποθηκεύονται στον ίδιο διακοσμητή.

Ευελιξία χρήσης : Παρά το γεγονός ότι οι πάροχοι των cloud υπηρεσιών υπόσχονται ότι η χρήση αυτών θα είναι ευέλικτη στην χρήση και στην ενσωμάτωση καθώς και στην εναλλαγή των υπηρεσιών, είναι κάτι που δεν έχει εξελιχθεί ακόμα πλήρως. Μπορεί οι πελάτες να αντιμετωπίσουν δυσκολίες κατά την διαδικασία μεταφοράς από τον έναν προμηθευτή των υπηρεσιών αυτών σε έναν άλλον. Οι χρήστες αυτών των υπηρεσιών μπορεί ακόμα να αντιμετωπίσουν προβλήματα κατά την φιλοξενία των εφαρμογών τους και στην υποστήριξη αυτών από τους παρόχους των υπηρεσιών. Για παράδειγμα, εφαρμογές που έχουν δημιουργηθεί σε περιβάλλον ανάπτυξης της Microsoft (.Net) δεν θα μπορούσαν να λειτουργήσουν σε πλατφόρμα Linux.

Περιορισμένος έλεγχος : Δεδομένου ότι οι cloud υποδομές ανήκουν ολοκληρωτικά και παρακολουθούνται από τους παρόχους των cloud υπηρεσιών, ο έλεγχος από τους πελάτες είναι ελάχιστος. Οι πελάτες μπορούν μόνο να ελέγχουν και να διαχειρίζονται τα δεδομένα και τις υπηρεσίες που λειτουργούν στις cloud υποδομές σε πρώτο επίπεδο και όχι στις υποδομές που αφορά το backend. Βασικές διεργασίες όπως η πρόσβαση στο κέλυφος του

server, στην ενημέρωση και διαχείριση του firmware, δεν ελέγχονται από τον πελάτη και τον τελικό χρήστη.

2 Συστήματα Διαχείρισης Έργων

Στο κεφάλαιο αυτό θα αναλυθούν κάποια από τα πιο διαδεδομένα συστήματα διαχείρισης έργων καθώς και οι βασικές τους λειτουργίες.

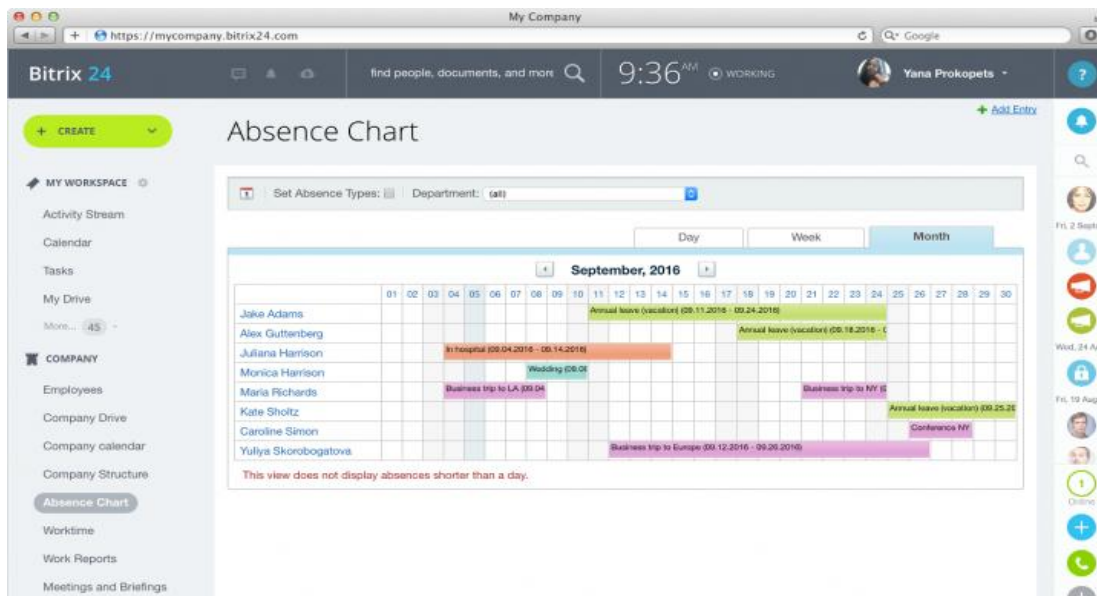
Τι είναι όμως η διαχείριση έργου ; Ο Peter Morris περιέγραψε τη διαχείριση έργου ως: «...διαδικασία ενσωμάτωσης όλων όσα πρέπει να γίνουν (και για τα οποία εφαρμόζονται, συνήθως, ειδικές τεχνικές διαχείρισης έργου), καθώς το έργο διανύει τον κύκλο ζωής του (από τη σύλληψή του μέχρι την παράδοσή του) ώστε να ικανοποιηθούν οι στόχοι του έργου».

Επί της ουσίας, για την ολοκλήρωση ενός έργου, χρησιμοποιείται ένα λογισμικό με διάφορες λειτουργίες και δυνατότητες που μας βοηθάει να ολοκληρώσουμε το έργο μας, καθώς υπάρχει η δυνατότητα να ελεγχθεί με μεγαλύτερη ευκολία χάρη στις δυνατότητες και τις λειτουργίες τέτοιων λογισμικών.

Η ανάπτυξη της εφαρμογής μας σχετικά με την διαχείριση πτυχιακών εργασιών, αποτελεί ένα σύστημα διαχείρισης έργου, αφού παρέχει στους χρήστες δυνατότητες τέτοιες ώστε να ολοκληρωθούν όλα τα στάδια σχετικά με την αποπεράτωση της πτυχιακής εργασίας, από την δήλωση μιας πτυχιακής, μέχρι το τελικό στάδιο της βαθμολόγησής της. Στα επόμενα κεφάλαια θα αναλυθεί κάθε οθόνη της εφαρμογής μας καθώς και οι λειτουργίες της.

2.1 Σύστημα Διαχείρισης Έργου Bitrix24

Το Bitrix24 είναι ένα σύστημα διαχείρισης έργων εντελώς δωρεάν για απεριόριστους χρήστες. Περιορίζει την αποθήκευση σε 5GB το μήνα, με δυνατότητα αναβάθμισης σε περισσότερα για 39 δολάρια το μήνα.



Εικόνα 2.1.1 : Σύστημα Διαχείρισης Έργου Bitrix 24

Θετικά

Οι χρήστες μπορούν να επιλέξουν εάν θα χρησιμοποιήσουν το Bitrix24 στο cloud ή τον στο δικό τους διακομιστή της εταιρείας.

Τα χαρακτηριστικά είναι εξαιρετικά: το Bitrix24 προσφέρει διαγράμματα Gantt, επιλογές διαστρωματωμένων εργασιών, παρακολούθηση χρόνου και διαχείριση, ακόμα και σχεδιασμό φόρτου εργασίας των εργαζομένων.

Το Bitrix24 καθιστά επίσης εφικτή την επικοινωνία σε πραγματικό χρόνο. Υποστηρίζει λειτουργία ομαδικής συζήτησης, τηλεδιάσκεψη καθώς και στιγμιαίο chat. Λειτουργεί επίσης ως εναλλακτική λύση Dropbox. Η δωρεάν έκδοση προσφέρει 5GB αποθήκευσης cloud για εύκολη κοινή χρήση εγγράφων.

Επιπλέον, οι πρόσφατες ενημερώσεις περιλαμβάνουν:

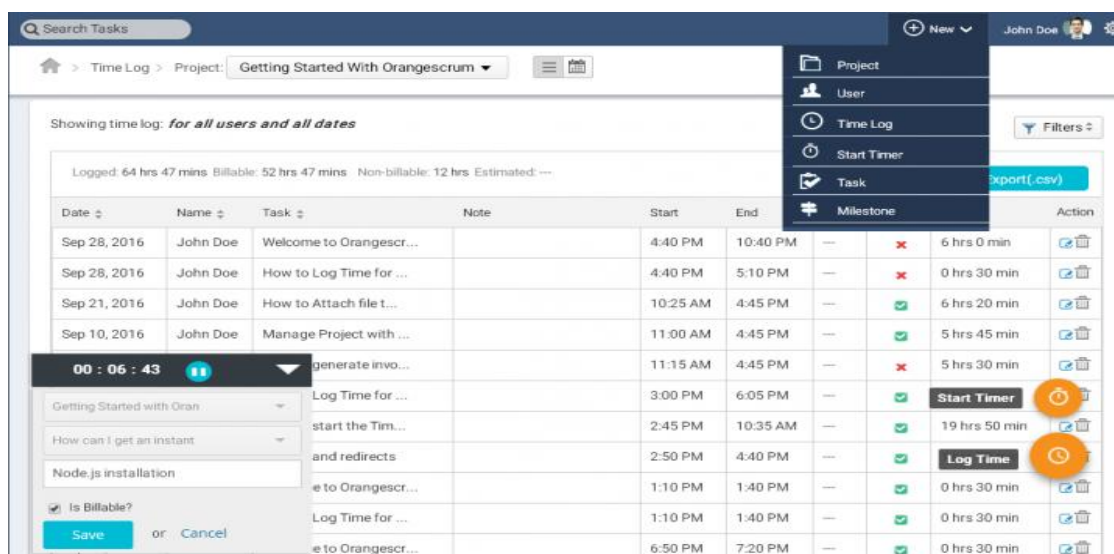
- Ένα εργαλείο σχεδίασης εργατικού φόρτου εργασίας, που επιτρέπει στους διαχειριστές να προγραμματίσουν συγκεκριμένο αριθμό ωρών για μια εργασία και στη συνέχεια να το συγκρίνουν με τον αριθμό των πραγματικών ωρών που δαπανώνται από εκείνους στους οποίους έχει ανατεθεί η εργασία.
- Η δυνατότητα δημιουργίας προτύπων εργασιών που περιέχουν υποστυλώματα και λίστες ελέγχου.

Αρνητικά

Το Bitrix24 στοχεύει στο να αντιμετωπίσει μια σειρά επιχειρηματικών προβλημάτων και οι χρήστες μπορούν εύκολα να κατακλύζονται μεταξύ της λειτουργικότητας διαχείρισης έργων και των εργαλείων CRM. Με αυτό, παράπονα από χρήστες τείνουν να προέρχονται από το δεύτερο μέρος του λογισμικού και όχι από το πρώτο.

2.2 Σύστημα Διαχείρισης Έργου OrangeScrum

Το Orangescrum προσφέρει το καλύτερο τοπικά φιλοξενούμενο δωρεάν λογισμικό διαχείρισης έργων για ομάδες τεχνολογίας πληροφορικής. Παρέχει τις βάσεις για ευέλικτη ανάπτυξη λογισμικού και διαχείριση πόρων. Η έκδοση on-premise κοστίζει \$ 0, ενώ η έκδοση cloud (η οποία είναι βεβαίως πολύ πιο υποστηριζόμενη) ξεκινά από \$ 9 το μήνα



Εικόνα 2.2.1 : Σύστημα Διαχείρισης Έργου OrangeScrum

Θετικά

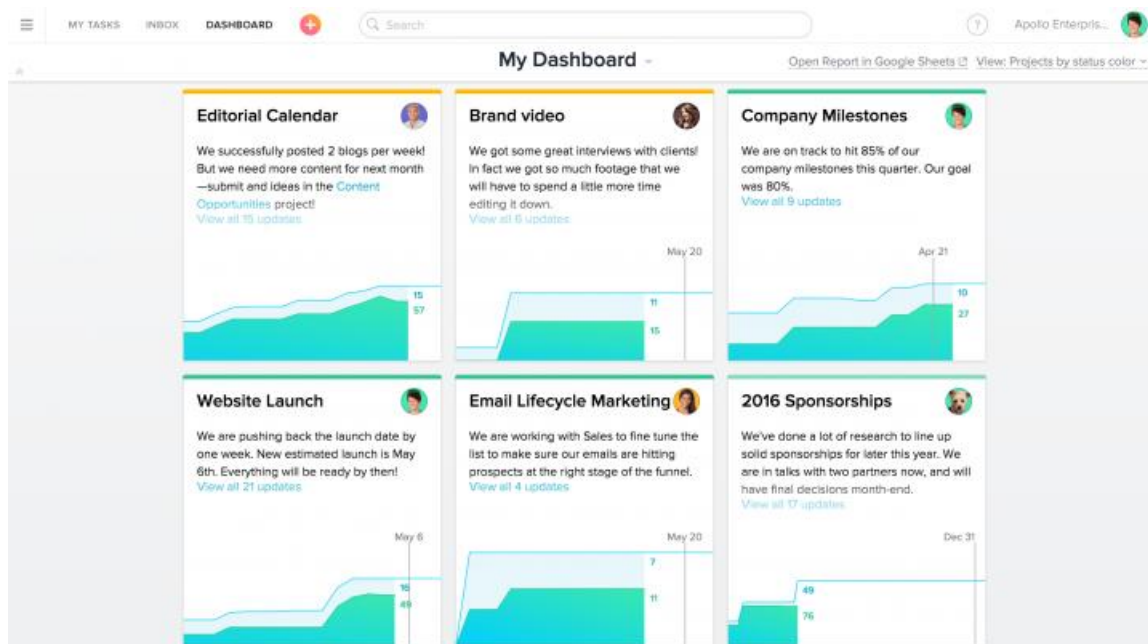
Το Orangescrum είναι πλήρως προσαρμόσιμο. Εάν οι χρήστες διαθέτουν τεχνογνωσία μπορούν να κάνουν αυτό το δωρεάν λογισμικό ό,τι θέλουν να είναι. Οι χρήστες μπορούν να δουν το έργο τους με προβολή Kanban, διαγράμματα Gantt και ομάδες κατάστασης εργασιών. Επιπλέον, οι χρήστες του Orangescrum μπορούν να ενσωματώσουν το Google Drive και το Dropbox στην ελεύθερη (on-premise) έκδοση.

Αρνητικά

Επειδή το Orangesrum είναι τόσο ευπροσάρμοστο, απαιτούνται σοβαρές γνώσεις κωδικοποίησης για να αξιοποιήσουν στο έπακρο αυτό το εργαλείο. Δεν είναι ένα πρόγραμμα για αρχάριους και η περισσότερη υποστήριξη βασίζεται σε φόρουμ.

2.3 Σύστημα Διαχείρισης Έργου Asana

Με βάση τους 140.000 πελάτες και 1.4 εκατομμύρια χρήστες, η Asana είναι μία από τις πιο δημοφιλείς εφαρμογές διαχείρισης έργων που είναι διαθέσιμες. Και το καλύτερο από όλα, για έως και 15 χρήστες, είναι δωρεάν. Και οι κριτικοί το αγαπούν επειδή διατηρεί συνολική συνολική βαθμολογία 4,5 / 5 με 1,300+ κριτικές για το Capterra.



Εικόνα 2.3.1 : Σύστημα Διαχείρισης Έργου Asana

Θετικά

Ο Dustin Moskovitz, ο συνιδρυτής του Facebook, σχεδίασε επίσης την Asana. Με την αισθητική και την απλότητα του πιο δημοφιλούς κοινωνικού δικτύου, το Asana είναι ένα διαισθητικό σύστημα διαχείρισης εργασιών που λειτουργεί καλύτερα για ομάδες που αναζητούν αλληλεπίδραση σε πραγματικό χρόνο.

Το Asana επιτρέπει στους χρήστες του να απεικονίζουν τους στόχους τους, να παρακολουθούν τον χρόνο τους, να δίνουν προτεραιότητα στα καθήκοντά τους και να λαμβάνουν ενημερώσεις για το έργο ακριβώς στο πρόγραμμα. Έχει επίσης μια λειτουργία ημερολογίου, για να γράψουν τις εργασίες της ομάδας απευθείας στον πίνακα οργάνων.

Επιπλέον, κατά το παρελθόν έτος, έχει προστεθεί μια εφαρμογή Android που δίνει την δυνατότητα μετατροπής μιας εργασίας σε ένα έργο, συνομιλίες και πίνακες ελέγχου.

Αρνητικά

Η Asana δεν επιτρέπει τη χρήση εκτός σύνδεσης. Επιπλέον, οι αξιολογητές θεωρούν ότι "μερικές φορές δεν είναι αρκετά διαισθητικό να βρούμε κάτι" και απαριθμούμε μια σειρά απογοητευτικών συνομιλιών με την φαινομενικά ελλιπή εξυπηρέτηση πελατών της Asana.

3 Γλώσσες Προγραμματισμού

Για την ανάπτυξη τις εφαρμογής χρησιμοποιήθηκαν οι εξής γλώσσες προγραμματισμού :

- Visual Basic
- JavaScript

Για να δοθούν στον περιηγητή οι οδηγίες για το τι θα εμφανίζεται στο χρήστη που θα προσπελάσει μία από τις σελίδες της εφαρμογής χρησιμοποιήθηκε η γλώσσα σήμανσης υπερκειμένου HTML.

- HTML

Για τον έλεγχο της εμφάνισης της HTML χρησιμοποιήθηκε η γλώσσα υπολογιστή CSS.

- CSS

3.1 Visual Basic

Η Visual Basic (VB) είναι γλώσσα προγραμματισμού τρίτης γενιάς, οδηγούμενη από συμβάντα (event driven) και έχει ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft για το μοντέλο προγραμματισμού COM. Η VB θεωρείται επίσης μία σχετικά εύκολη γλώσσα προγραμματισμού στην εκμάθηση και τη χρησιμοποίηση, λόγω των χαρακτηριστικών της, καθώς έχει Γραφικό Περιβάλλον Χρήστη και συγγένεια με την γλώσσα προγραμματισμού BASIC. Η Visual Basic προέρχεται από τη BASIC και επιτρέπει την ταχεία ανάπτυξη εφαρμογών (RAD) με Γραφικό Περιβάλλον Χρήστη (GUI), πρόσβαση σε βάσεις δεδομένων χρησιμοποιώντας αντικείμενα (Data Access Objects, Remote Data Objects, ή ActiveX Data Objects), και τη δημιουργία στοιχείων ελέγχου ActiveX και αντικειμένων. Οι γλώσσες προγραμματισμού τύπου "scripting", όπως η VBA και VBScript συντακτικά είναι παρόμοιες με τη Visual Basic, αλλά έχουν διαφορετικές επιδόσεις. Ένας προγραμματιστής μπορεί να ολοκληρώσει μια εφαρμογή χρησιμοποιώντας τα στοιχεία που παρέχονται με την Visual Basic. Προγράμματα γραμμένα σε Visual Basic μπορούν, επίσης, να χρησιμοποιήσουν το Windows API, αλλά κάτι τέτοιο απαιτεί δηλώσεις εξωτερικών συναρτήσεων.

Η τελική έκδοση 6 βγήκε το 1998. Η εκτεταμένη υποστήριξη της Microsoft έληξε το Μάρτιο του 2008 και ορίστηκε διάδοχος της η Visual Basic.NET (γνωστή απλά ως Visual Basic).

3.2 Javascript

Η JavaScript (JS) είναι διερμηνευμένη γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται.

Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme.

Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστρεφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού.

Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων, τέτοια παραδείγματα είναι τα έγγραφα PDF, οι εξειδικευμένοι φυλλομετρητές (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets). Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js) έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side).

3.3 HTML

Η HTML (HyperText Markup Language) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων.

Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από *ετικέτες* (tags), οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα `<html>`), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα `<h1>` και `</h1>`), με την πρώτη να ονομάζεται *ετικέτα έναρξης* και τη δεύτερη *ετικέτα λήξης* (ή σε άλλες περιπτώσεις *ετικέτα*

ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.

Ο σκοπός ενός web browser είναι να διαβάζει τα έγγραφα HTML και τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας.

Τα στοιχεία της HTML χρησιμοποιούνται για να κτίσουν όλους του ιστότοπους. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου) καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, παραθέσεις και άλλα. Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

Οι Web browsers μπορούν επίσης να αναφέρονται σε στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου.

3.4 CSS

Η CSS (*Cascading Style Sheets-Διαδοχικά Φύλλα Στυλ*) είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστοτόπου. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στυλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.

4 Εγκατάσταση Εργαλείων Ανάπτυξης Εφαρμογής

Πριν πραγματοποιηθεί η ανάπτυξη της εφαρμογής, θα πραγματοποιηθεί η βήμα-βήμα παρουσίαση της εγκατάστασης των εργαλείων ανάπτυξης τα οποία είναι απαραίτητα για την ανάπτυξη αυτής στον υπολογιστή μας.

Θα γίνει εγκατάσταση των εξής εργαλείων :

- Microsoft Visual Studio 2012
- Microsoft SQL SERVER 2012

4.1 Εγκατάσταση Microsoft Visual Studio

- 1) Αφού γίνει download και "τρέξει" το αρχείο **Setup.exe** γίνεται αποδοχή των όρων χρήσης και επιλέγεται **Next**.



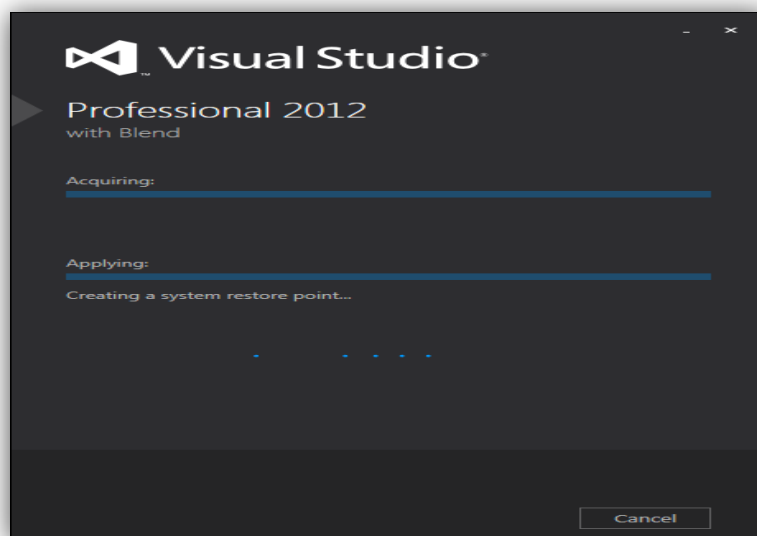
Εικόνα 4.1.1 : Εγκατάσταση Visual Studio

- 2) Επιλέγονται τα απαραίτητα εργαλεία και έπειτα επιλέγεται **Install**. Αν επιλεγούν όλα τα εργαλεία εγκατάστασης θα καταληφθούν γύρω στα 7.90 GB χώρο στο σύστημα μας.



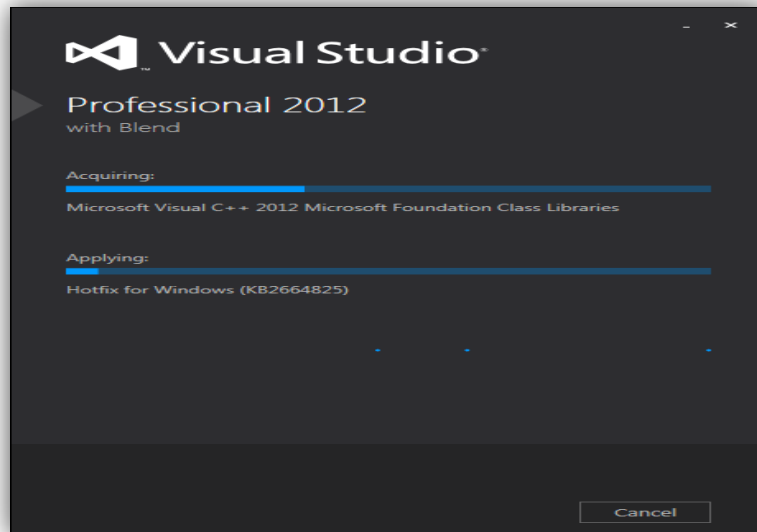
Εικόνα 4.1.2 : Εγκατάσταση Visual Studio

- 3) Πριν αρχίσει η διαδικασία εγκατάστασης θα δημιουργηθεί ένα σημείο επαναφοράς **System Restore Point**.



Εικόνα 4.1.3 : Εγκατάσταση Visual Studio

4) Μόλις αυτό γίνει θα αρχίσει η διαδικασία εγκατάστασης.



Εικόνα 4.1.4 : Εγκατάσταση Visual Studio

5) Κατά την διάρκεια της εγκατάστασης θα ζητηθεί να γίνει επανεκκίνηση. Επιλέγεται το κουμπί **Restart Now** για να πραγματοποιηθεί.



Εικόνα 4.1.5 : Εγκατάσταση Visual Studio

6) Η εγκατάσταση θα συνεχιστεί , όταν το σύστημα εκτελέσει την επανεκκίνηση.



Εικόνα 4.1.6 : Εγκατάσταση Visual Studio

7) Η διαδικασία εγκατάστασης θα ολοκληρωθεί γύρω σε 20-30 λεπτά.



Εικόνα 4.1.7 : Εγκατάσταση Visual Studio

- 8) Η εγκατάσταση ολοκληρώθηκε με επιτυχία. Μπορούμε να αρχίσουμε να δουλεύουμε με το Visual Studio πατώντας το κουμπί **LAUNCH**.



Εικόνα 4.1.8 : Εγκατάσταση Visual Studio

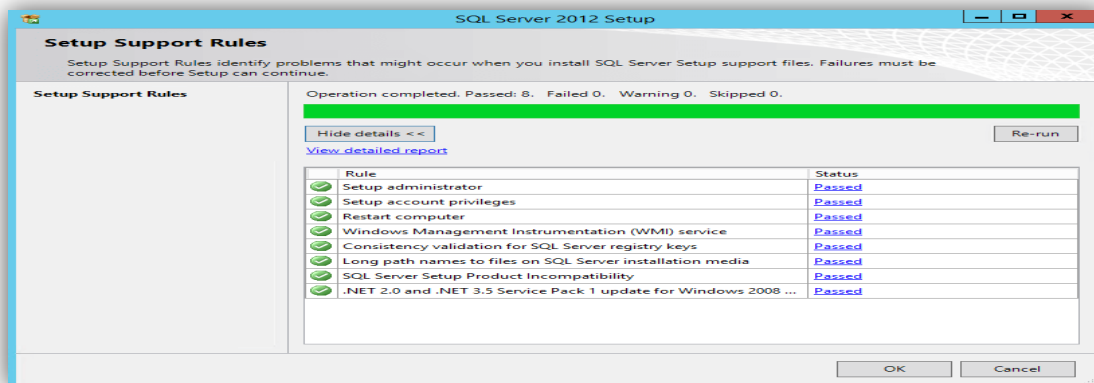
4.2 Εγκατάσταση Microsoft SQL SERVER 2012

- 1) Για να αρχίσει η εγκατάσταση του Microsoft SQL SERVER 2012 στην καρτέλα **Installation** επιλέγεται η πρώτη επιλογή στο παράθυρο.



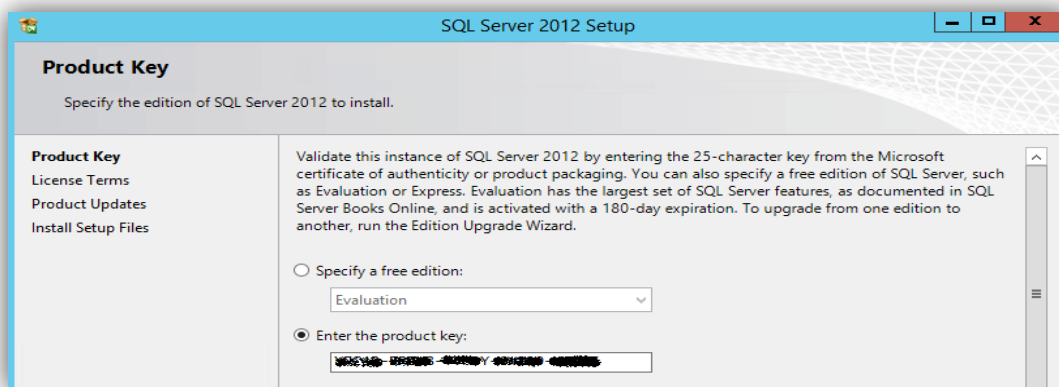
Εικόνα 4.2.1 : Εγκατάσταση SQL SERVER

- 2) Θα γίνει έλεγχος για να διαπιστωθεί αν υπάρχουν όλες οι προϋποθέσεις για την εγκατάσταση του SQL SERVER. Αν όλα είναι εντάξει τότε θα εμφανιστούν κάποια πράσινα check. Επιλέγεται **OK** για να συνεχιστεί η διαδικασία.



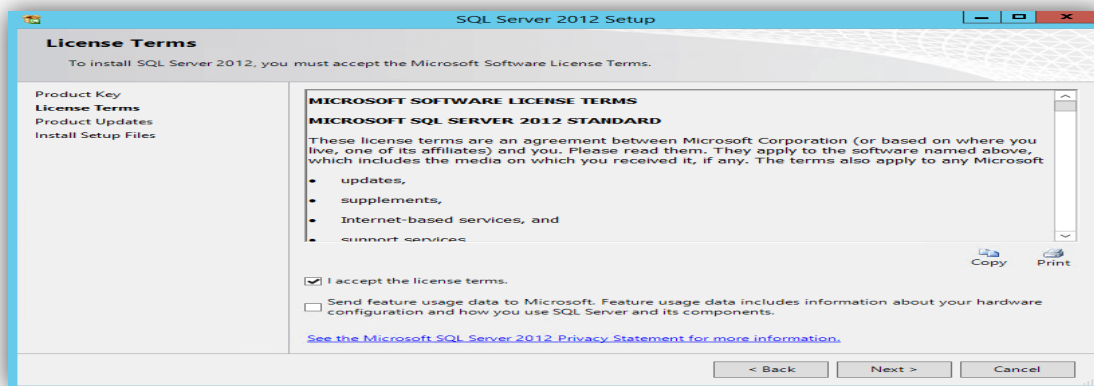
Εικόνα 4.2.2 : Εγκατάσταση SQL SERVER

- 3) Εισάγεται το κλειδί του προϊόντος ή επιλέγεται η evaluation edition και επιλέγεται **Next**.



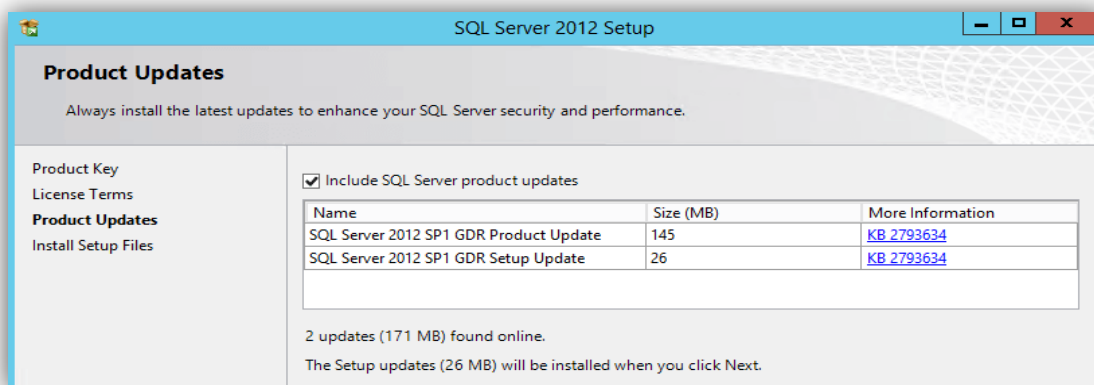
Εικόνα 4.2.3 : Εγκατάσταση SQL SERVER

- 4) Γίνεται αποδοχή των όρων χρήσης του προϊόντος και επιλέγεται **Next**.



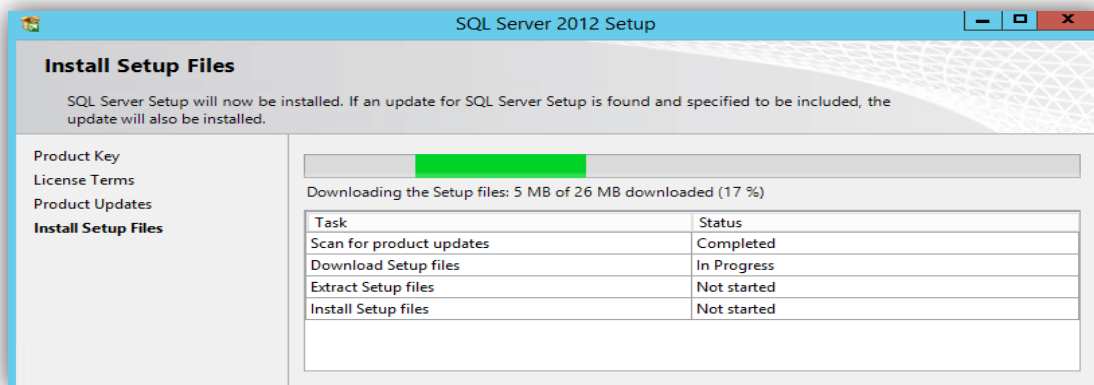
Εικόνα 4.2.4 : Εγκατάσταση SQL SERVER

5) Επιλέγεται να γίνουν κάποιες ενημερώσεις και έπειτα επιλέγεται Next .



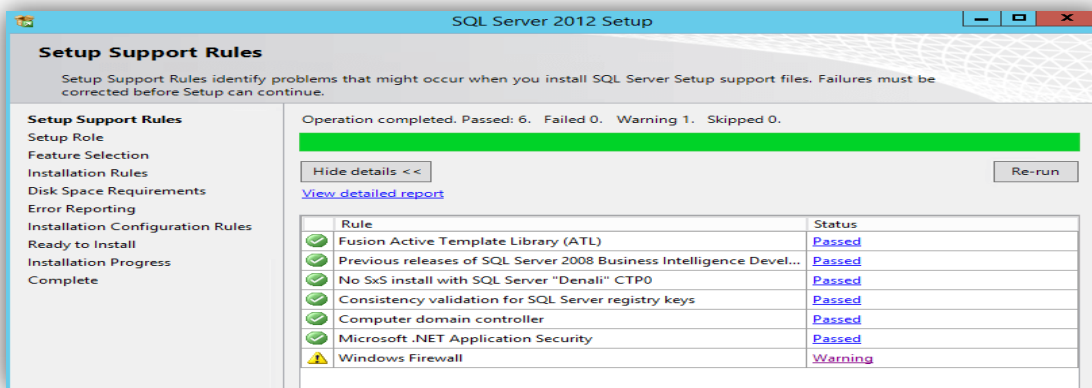
Εικόνα 4.2.5 : Εγκατάσταση SQL SERVER

6) Οι ενημερώσεις αρχίζουν να κατεβαίνουν όπως φαίνεται.



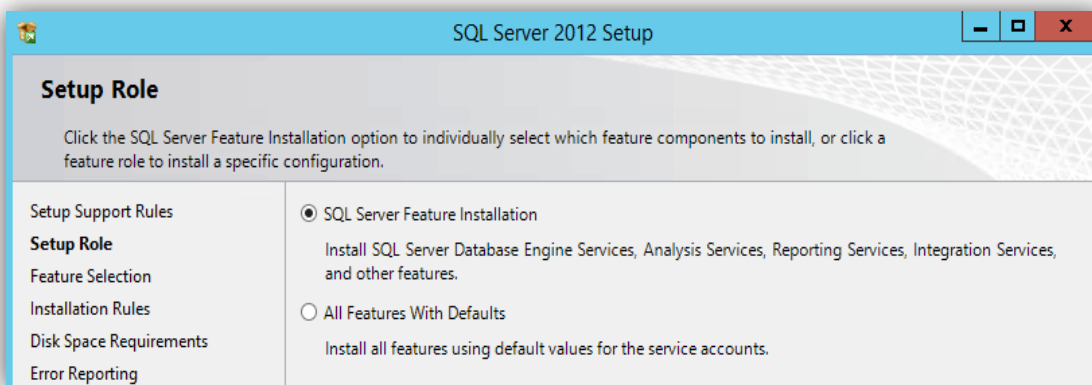
Εικόνα 4.2.6 : Εγκατάσταση SQL SERVER

7) Γίνεται έλεγχος για να διαπιστωθεί να υπάρχουν όλες οι προϋποθέσεις. Αν όλα είναι καλά τότε θα εμφανιστούν κάποια πράσινα check



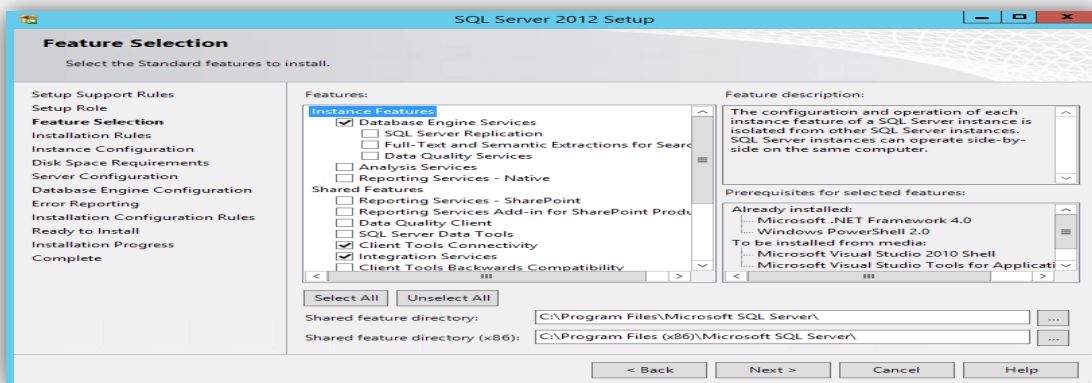
Εικόνα 4.2.7 : Εγκατάσταση SQL SERVER

- 8) Στην καρτέλα **Setup Role** επιλέγεται **SQL server feature Installation** και επιλέγεται **Next**.



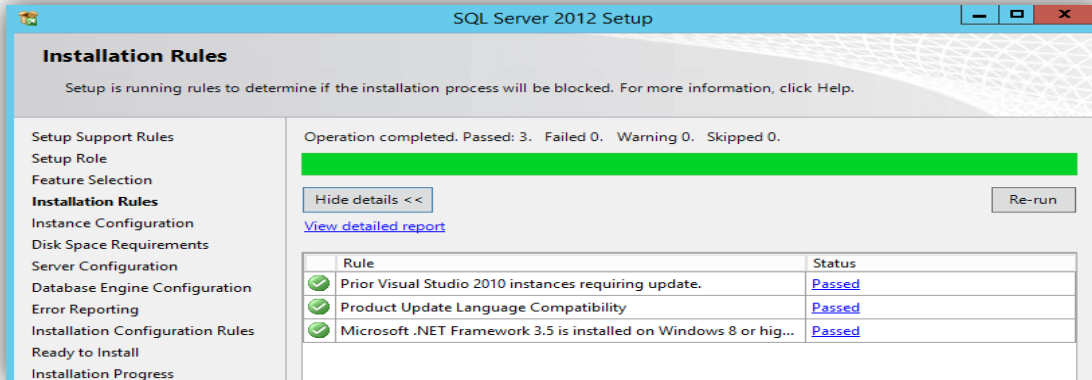
Εικόνα 4.2.7 : Εγκατάσταση SQL SERVER

- 9) Επιλέγονται τα χαρακτηριστικά που είναι επιθυμητά και στη συνέχεια επιλέγεται **Next**.



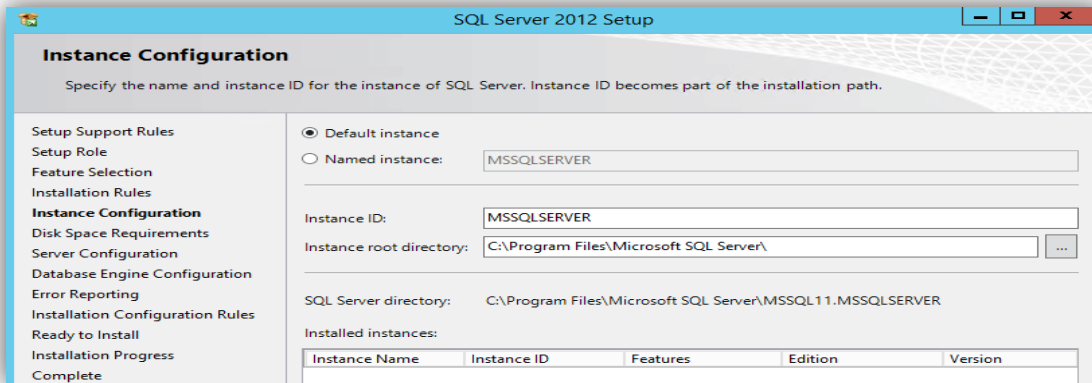
Εικόνα 4.2.8 : Εγκατάσταση SQL SERVER

10) Αφού ελεγχθούν οι όροι εγκατάστασης επιλέγεται **Next** .



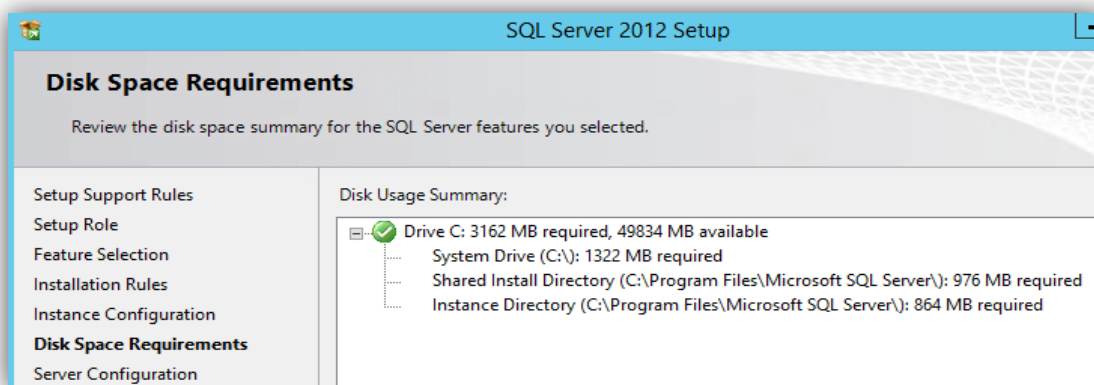
Εικόνα 4.2.9 : Εγκατάσταση SQL SERVER

11) Επιλέγεται **Default Instance** και το μονοπάτι όπου θα γίνει η εγκατάσταση, επιλέγεται **Next** .



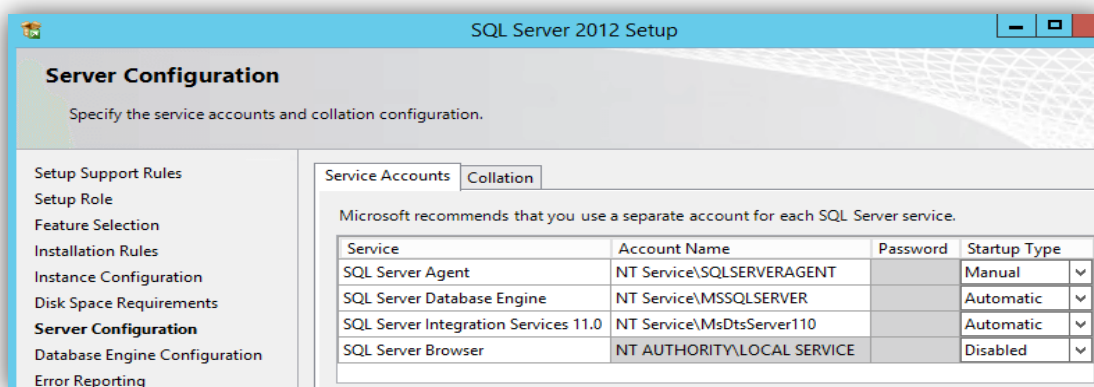
Εικόνα 4.2.10 : Εγκατάσταση SQL SERVER

12) Στην καρτέλα **Disc Space Requirements** είναι ορατό το μονοπάτι που θα γίνει η εγκατάσταση.



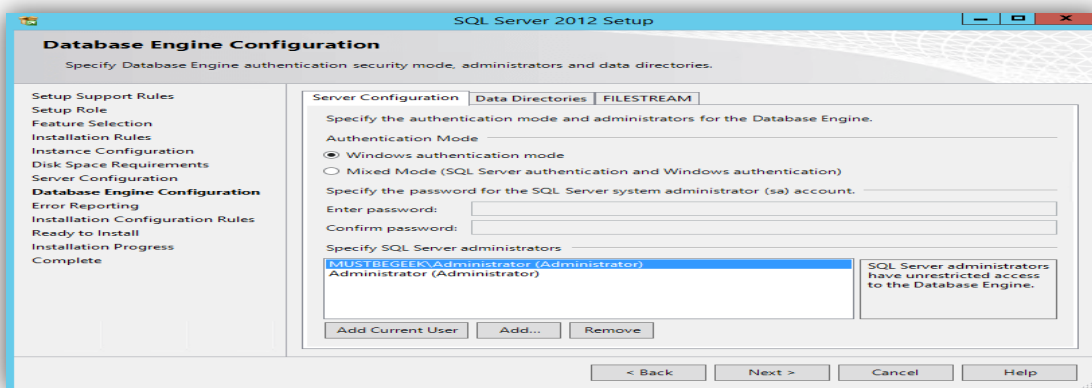
Εικόνα 4.2.11 : Εγκατάσταση SQL SERVER

13) Στην καρτέλα **Server Accounts** αφήνονται οι ήδη υπάρχουσες ρυθμίσεις.



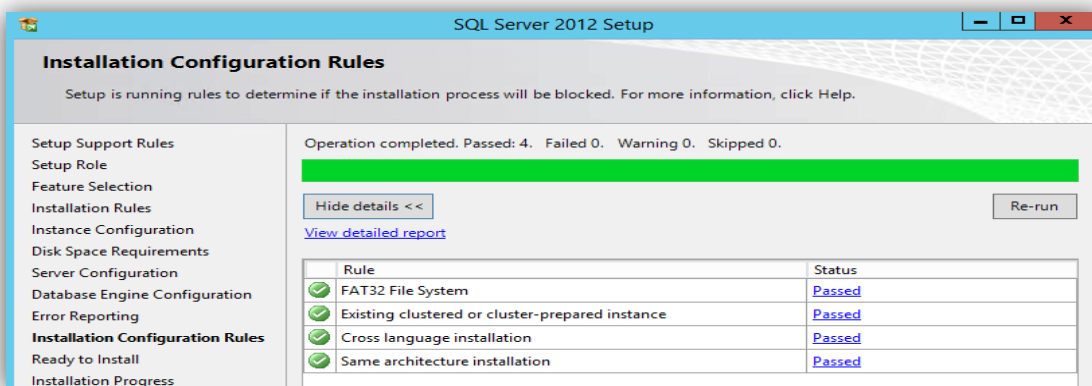
Εικόνα 4.2.12 : Εγκατάσταση SQL SERVER

14) Στην καρτέλα **Database Engine Configuration** επιλέγεται με ποιόν τρόπο θα γίνεται η ταυτοποίηση των στοιχείων ώστε να γίνεται είσοδος στον SQL SERVER. Επιλέγεται Windows authentication mode και ορίζεται ο διαχειριστής και επιλέγεται **Next**.



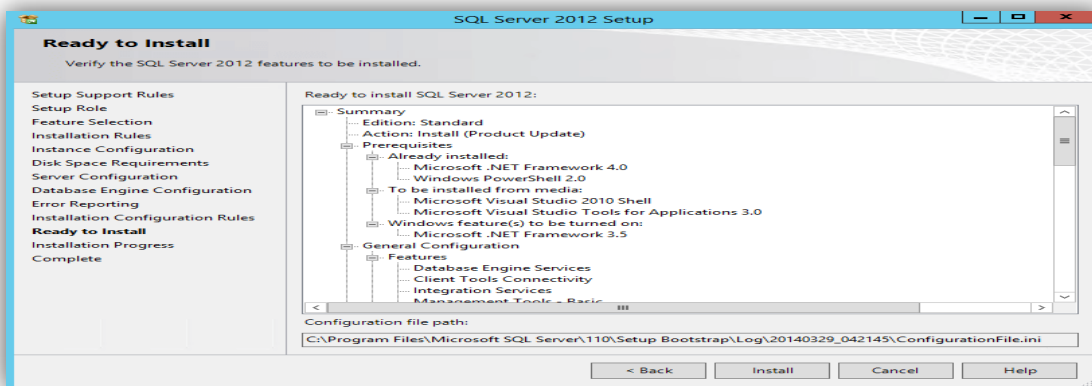
Εικόνα 4.2.13 : Εγκατάσταση SQL SERVER

15) Αν όλα είναι εντάξει τότε θα εμφανιστούν πράσινα check και επιλέγεται **Next**.



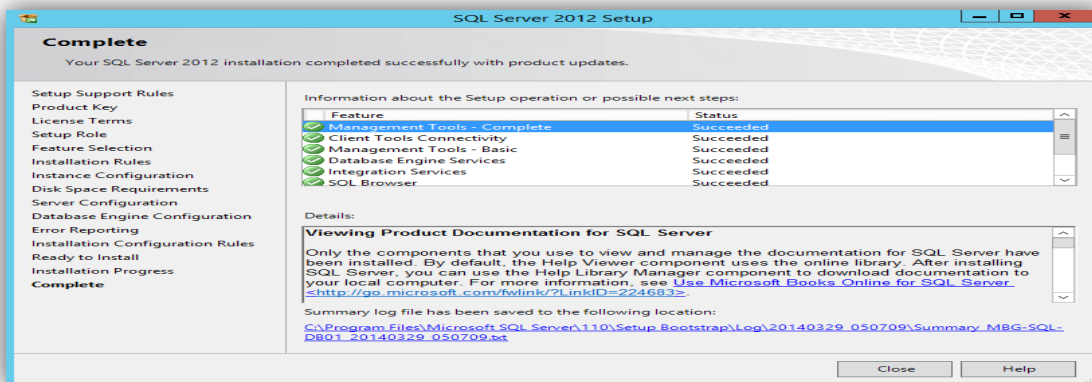
Εικόνα 4.2.14 : Εγκατάσταση SQL SERVER

16) Είναι ορατές στην καρτέλα **Ready to Install** οι πληροφορίες εγκατάστασης και επιλέγεται **Install**.



Εικόνα 4.2.15 : Εγκατάσταση SQL SERVER

17) Αφού ολοκληρωθεί η εγκατάσταση, επιλέγεται **Close**. Είμαστε έτοιμοι λοιπόν να αρχίσουμε να δουλεύουμε με τον Microsoft SQL SERVER 2012.



Εικόνα 4.2.16 : Εγκατάσταση SQL SERVER

5 Βάση Δεδομένων

5.1 Περιγραφή της Βάσης Δεδομένων

Για την κάλυψη της λειτουργίας της διαδικτυακής αυτής εφαρμογής χρειάζεται να διαχειριστούμε δυναμικά δεδομένα. Τα δεδομένα αυτά αποθηκεύονται στην βάση δεδομένων που επικοινωνεί με την εφαρμογή μας. Συγκεκριμένα, χρησιμοποιήθηκε ο Microsoft SQL SERVER 2012 σαν βάση. Στο κεφάλαιο αυτό θα γίνει εκτενής αναφορά στους πίνακες της βάσης μας και στα πεδία που περιλαμβάνουν.

5.2 Βάση Δεδομένων και Πίνακες

Τα δεδομένα που καταχωρούν οι χρήστες αποθηκεύονται σε κάποιους πίνακες οι οποίοι είναι αποθηκευμένοι με την σειρά τους στην βάση δεδομένων.

Το κάθε πεδίο σε έναν πίνακα έχει τα εξής χαρακτηριστικά :

- **Column Name** : Δίνουμε ένα όνομα στο κάθε πεδίο.
- **Data Type** : Δίνουμε τον τύπο που είναι τα δεδομένα.
- **Allow Nulls** : Επιλέγουμε αν θα επιτρέπονται κενές εγγραφές.

Οι πίνακες είναι οι εξής :

- i. Πίνακας **register**

Περιγραφή : Σε αυτόν τον πίνακα καταχωρούνται όλα τα στοιχεία των χρηστών τα οποία δόθηκαν κατά την εγγραφή στον σύστημα .

Δομή :

	Column Name	Data Type	Allow Nulls
🔑	CnID	int	<input type="checkbox"/>
	username	nchar(50)	<input type="checkbox"/>
	password	nvarchar(50)	<input type="checkbox"/>
	FName	nvarchar(50)	<input checked="" type="checkbox"/>
	LName	nvarchar(50)	<input checked="" type="checkbox"/>
	Speciality	nvarchar(50)	<input checked="" type="checkbox"/>
	Am	int	<input checked="" type="checkbox"/>
	Telephone	varchar(50)	<input checked="" type="checkbox"/>
	Skypeid	nvarchar(50)	<input checked="" type="checkbox"/>
	IsProf	bit	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


Εικόνα 5.2.1 : Πίνακας register

- **CnID (Primary Key)** : Είναι ο μοναδικός αριθμός κατά την εγγραφή του χρήστη. Κάθε φορά που κάποιος χρήστης εγγράφεται ο αριθμός αυτός αυξάνεται αυτόματα κατά ένα.
- **username** : Είναι το email που δίνει ο κάθε χρήστης κατά την εγγραφή ώστε να κάνει είσοδο στο σύστημα.
- **password** : Είναι ο κωδικός που δίνει ο χρήστης κατά την εγγραφή ώστε να κάνει είσοδο στο σύστημα.
- **FName** : Είναι το όνομα του χρήστη που δίνει ο χρήστης κατά την εγγραφή.
- **LName** : Είναι το επώνυμο του χρήστη που δίνει ο χρήστης κατά την εγγραφή.
- **Speciality** : Είναι το τμήμα που δίνει ο χρήστης φοιτητής κατά την εγγραφή στο σύστημα.
- **Am** : Είναι ο αριθμός μητρώου που δίνει ο χρήστης φοιτητής κατά την εγγραφή στο σύστημα.
- **Telephone** : Είναι το τηλέφωνο που δίνει ο χρήστης κατά την εγγραφή στο σύστημα.
- **Skypeid** : Είναι το Skype ID που δίνει ο χρήστης κατά την εγγραφή στο σύστημα.
- **IsProf** : Με αυτό το πεδίο θα δώσουμε αργότερα κάποια από τα δικαιώματα στις οθόνες όπου θα έχουν πρόσβαση οι χρήστες . Για τους καθηγητές IsProf = 1 για τους φοιτητές IsProf = 0.

ii. Πίνακας `graduate_themes`

Περιγραφή : Σε αυτόν τον πίνακα καταχωρούνται οι πτυχιακές εργασίες από τους καθηγητές.

Δομή :

	Column Name	Data Type	Allow Nulls
	ID	int	<input type="checkbox"/>
	CnID	int	<input checked="" type="checkbox"/>
	Title	nvarchar(50)	<input checked="" type="checkbox"/>
	ContentType	nvarchar(200)	<input checked="" type="checkbox"/>
	Data	varbinary(MAX)	<input checked="" type="checkbox"/>
	Name	varchar(50)	<input checked="" type="checkbox"/>
	Descr	nvarchar(300)	<input checked="" type="checkbox"/>
	IsDeleted	bit	<input type="checkbox"/>
	IsAvailable	bit	<input type="checkbox"/>
	IsCompleted	bit	<input checked="" type="checkbox"/>

Εικόνα 5.2.2 : Πίνακας `graduate_themes`

- **ID (Primary Key)** : Είναι ο μοναδικός αριθμός όπου εξασφαλίζει ότι κάθε πτυχιακή όπου εισάγει ο καθηγητής είναι μοναδική. Κάθε φορά που γίνεται εισαγωγή μιας πτυχιακής ο αριθμός αυτός αυξάνεται αυτόματα κατά ένα.
- **CnID** : Είναι ο αριθμός που έχει κάθε χρήστης όταν κάνει την εγγραφή στο σύστημα.
- **Title** : Είναι ο τίτλος της πτυχιακής.
- **ContentType** : Είναι ο τύπος του αρχείου που ανεβάζει ο καθηγητής.
- **Data** : Είναι το αρχείο που ανεβάζει ο καθηγητής.
- **Name** : Είναι το όνομα του αρχείου που ανεβάζει ο καθηγητής.
- **Descr** : Είναι η περιγραφή της πτυχιακής.
- **IsDeleted** : Κατά την εισαγωγή μιας πτυχιακής το `IsDeleted = 0` όταν γίνεται διαγραφή μιας πτυχιακής το `IsDeleted = 1`.
- **IsAvailable** : Όταν μια πτυχιακή είναι διαθέσιμη το `IsAvailable = 0` όταν δεν είναι διαθέσιμη `IsAvailable = 1`.

- **IsCompleted** : Όταν μια πτυχιακή δεν είναι ολοκληρωμένη IsCompleted =0 όταν είναι IsCompleted=1.

iii. Πίνακας **graduate_request**

Περιγραφή : Σε αυτόν τον πίνακα καταχωρούνται τα αιτήματα των φοιτητών για ανάθεση πτυχιακής εργασίας.

Δομή :

	Column Name	Data Type	Allow Nulls
PK	IDIncr	int	<input type="checkbox"/>
	ID	int	<input type="checkbox"/>
	CnID	int	<input checked="" type="checkbox"/>
	Title	nvarchar(50)	<input checked="" type="checkbox"/>
	Descr	nvarchar(300)	<input checked="" type="checkbox"/>
	CnIDStudent	int	<input checked="" type="checkbox"/>
	IsAvailable	bit	<input checked="" type="checkbox"/>
	IsCompleted	bit	<input checked="" type="checkbox"/>
	IsProven	bit	<input checked="" type="checkbox"/>
	IsRejected	bit	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Εικόνα 5.2.3 : Πίνακας **graduate_requests**

- **IDIncr (Primary Key)** : Είναι ο μοναδικός αριθμός όπου εξασφαλίζει ότι κάθε πτυχιακή όπου εισάγει ο καθηγητής είναι μοναδική. Κάθε φορά που γίνεται εισαγωγή μιας πτυχιακής ο αριθμός αυτός αυξάνεται αυτόματα κατά ένα.
- **ID** : Κρατάμε το ID από τον πίνακα **graduate_themes**.
- **CnID** : Είναι ο μοναδικός αριθμός που έχει κάθε χρήστης όταν κάνει την εγγραφή στο σύστημα το πεδίο αυτό αφορά τους καθηγητές.
- **Title** : Είναι ο τίτλος της πτυχιακής.
- **Descr** : Είναι η περιγραφή της πτυχιακής.
- **CnIDStudent** : Είναι ο μοναδικός αριθμός που έχει ο κάθε χρήστης όταν κάνει εγγραφή στο σύστημα το πεδίο αυτό αφορά τους φοιτητές.

- **IsAvailable** : Όταν μια πτυχιακή είναι διαθέσιμη το IsAvailable =0 όταν δεν είναι διαθέσιμη IsAvailable =1.
- **IsCompleted** : Όταν μια πτυχιακή δεν είναι ολοκληρωμένη IsCompleted=0 όταν είναι IsCompleted=.
- **IsProven** : Όταν μια πτυχιακή εγκριθεί IsProven=1 αλλιώς IsProven=0.
- **IsRejected** : Όταν μια πτυχιακή απορριφθεί IsRejected=1 αλλιώς IsRejected=0.

iv. Πίνακας **graduate_completed**

Περιγραφή : Σε αυτόν τον πίνακα καταχωρούνται οι ολοκληρωμένες πτυχιακές εργασίες.

Δομή :

	Column Name	Data Type	Allow Nulls
▶ 🔑	IDIncr1	int	<input type="checkbox"/>
	IDIncr	int	<input type="checkbox"/>
	ID	int	<input type="checkbox"/>
	CnID	int	<input checked="" type="checkbox"/>
	Name	varchar(50)	<input checked="" type="checkbox"/>
	ContentType	nvarchar(200)	<input checked="" type="checkbox"/>
	Data	varbinary(MAX)	<input checked="" type="checkbox"/>
	Title	nvarchar(50)	<input checked="" type="checkbox"/>
	Descr	nvarchar(300)	<input checked="" type="checkbox"/>
	CnIDStudent	int	<input checked="" type="checkbox"/>
	IsAvailable	bit	<input checked="" type="checkbox"/>
	IsCompleted	bit	<input checked="" type="checkbox"/>
	Date	datetime	<input checked="" type="checkbox"/>
	Grade	decimal(18, 0)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Εικόνα 5.2.4 : Πίνακας **graduate_completed**

- **IDIncr1 (Primary Key)** : Είναι ο μοναδικός αριθμός όπου εξασφαλίζει ότι κάθε πτυχιακή είναι μοναδική. Κάθε φορά που γίνεται εισαγωγή μιας πτυχιακής ο αριθμός αυτός αυξάνεται αυτόματα κατά ένα.
- **IDIncr** : Κρατάμε το IDIncr από τον πίνακα **graduate_requests**
- **ID** : Κρατάμε το ID από τον πίνακα **graduate_requests**
- **CnID** : Είναι ο μοναδικός αριθμός που έχει κάθε χρήστης όταν κάνει την εγγραφή στο σύστημα. Το πεδίο αυτό αφορά τους καθηγητές.
- **Name** : Είναι το όνομα του αρχείου που ανεβάζει ο καθηγητής.
- **ContentType** : Είναι ο τύπος του αρχείου που ανεβάζει ο καθηγητής.
- **Data** : Είναι το αρχείο που ανεβάζει ο καθηγητής.
- **Title** : Είναι ο τίτλος της πτυχιακής .
- **Descr** : Είναι η περιγραφή της πτυχιακής.
- **CnIDStudent** : Είναι ο μοναδικός αριθμός που έχει ο κάθε χρήστης όταν κάνει εγγραφή στο σύστημα. Το πεδίο αυτό αφορά τους φοιτητές.
- **IsAvailable** : Όταν μια πτυχιακή είναι διαθέσιμη το IsAvailable = 0 όταν δεν είναι διαθέσιμη IsAvailable = 1.
- **IsCompleted** : Όταν μια πτυχιακή δεν είναι ολοκληρωμένη IsCompleted=0 , όταν είναι IsCompleted=1.
- **Date** : Είναι η ημερομηνία ολοκλήρωσης πτυχιακής εργασίας.
- **Grade** : Είναι ο βαθμός πτυχιακής εργασίας.

v. Πίνακας **messages**

Περιγραφή: Σε αυτόν τον πίνακα καταχωρούνται τα μηνύματα των χρηστών.

Δομή:

	Column Name	Data Type	Allow Nulls
🔑	ID	int	<input type="checkbox"/>
	SenderCnID	int	<input type="checkbox"/>
	Title	nvarchar(50)	<input type="checkbox"/>
	Theme	nvarchar(MAX)	<input type="checkbox"/>
	RecieverCnID	int	<input type="checkbox"/>
	IsReaden	bit	<input type="checkbox"/>
	Name	varchar(50)	<input checked="" type="checkbox"/>
	ContentType	nvarchar(200)	<input checked="" type="checkbox"/>
	Data	varbinary(MAX)	<input checked="" type="checkbox"/>
	IsDeletedSender	bit	<input checked="" type="checkbox"/>
	IsDeletedReciever	bit	<input checked="" type="checkbox"/>
	Date	datetime	<input type="checkbox"/>
			<input type="checkbox"/>

Εικόνα 5.2.5 : Πίνακας messages

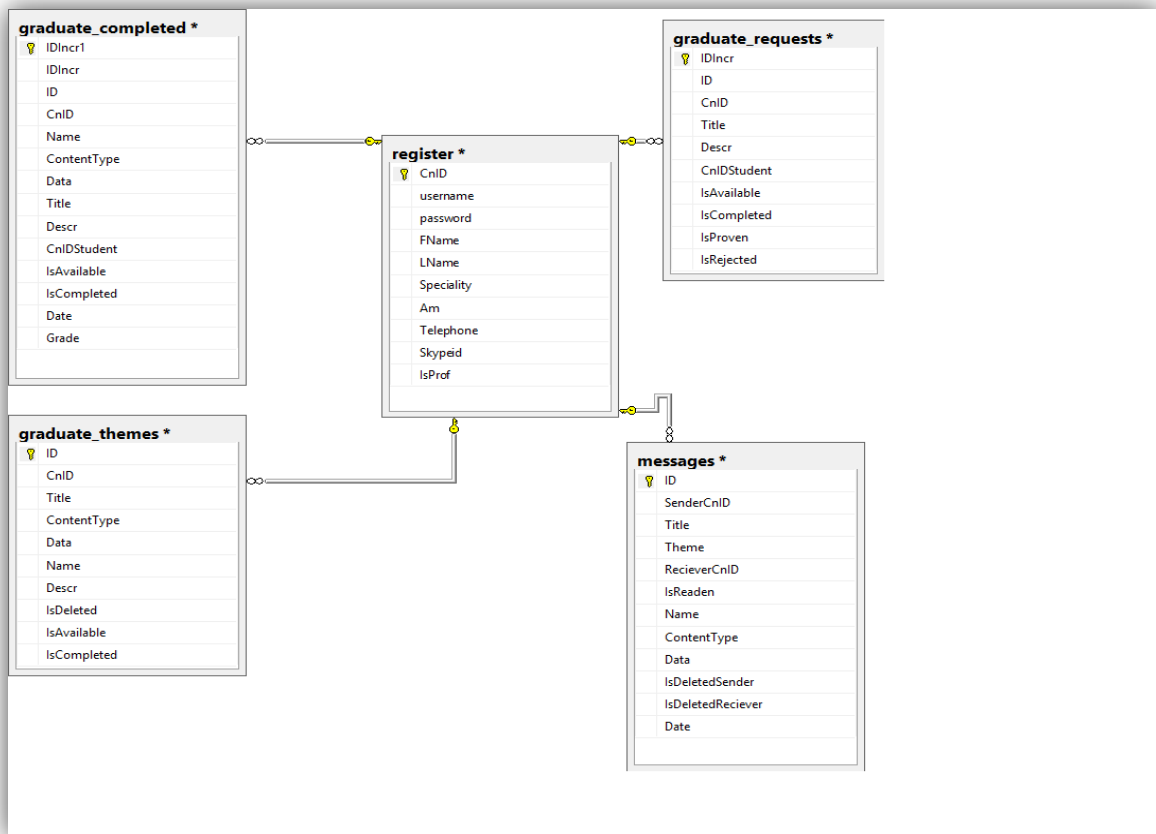
- **ID(Primary Key)** : Είναι ο μοναδικός αριθμός όπου εξασφαλίζει ότι κάθε μήνυμα όπου στέλνουμε είναι μοναδικό. Κάθε φορά που γίνεται αποστολή ενός μηνύματος ο αριθμός αυτός αυξάνεται αυτόματα κατά ένα.
- **SenderCnID** : Είναι ο μοναδικός αριθμός που έχει κάθε χρήστης όταν κάνει την εγγραφή στο σύστημα. Ο αριθμός αυτός είναι του αποστολέα.
- **Title** : Είναι ο τίτλος μηνύματος.
- **Theme** : Θέμα μηνύματος.
- **RecieverCnID** : Είναι ο μοναδικός αριθμός που έχει κάθε χρήστης όταν κάνει την εγγραφή στο σύστημα. Ο αριθμός αυτός είναι του παραλήπτη.
- **IsReaden** : Αν το μήνυμα είναι αδιάβαστο τότε IsReaden =0 , αν είναι διαβασμένο τότε IsReaden =1.
- **Name** : Είναι το όνομα του συνημμένου αρχείου.
- **ContentType** : Είναι ο τύπος του συνημμένου αρχείου.
- **Data** : Είναι το συνημμένο αρχείο.
- **IsDeletedSender** : Αν είναι διαγραμμένο το μήνυμα από τον αποστολέα τότε IsDeletedSender=1 αλλιώς IsDeletedSender =0
- **IsDeletedReciever** : Αν είναι διαγραμμένο το μήνυμα από τον παραλήπτη τότε IsDeletedReciever=1 αλλιώς IsDeletedReciever =0.

- **Date** : Ημερομηνία/Ωρα Αποστολής.

5.3 Σχεσιακό Διάγραμμα ER Βάσης Δεδομένων

Το μοντέλο οντοτήτων συσχετίσεων (Entity Relationship model, ER model) είναι μια διαγραμματική αναπαράσταση της δομής της βάσης δεδομένων, και ως εκ τούτου χρησιμοποιείται κατά το στάδιο του λογικού ή εννοιολογικού σχεδιασμού (conceptual design) της βάσης. Αυτό σημαίνει πως η κεντρική λειτουργία του δεν συνίσταται στον καθορισμό του τρόπου με τον οποίο τα δεδομένα της βάσης θα αποθηκεύουν στο σύστημα, αλλά μάλλον στην ταυτοποίηση των δεδομένων της εφαρμογής και του τρόπου με τον οποίο συσχετίζονται μεταξύ τους. Στο διάγραμμα ER , η κάθε οντότητα απεικονίζεται με ένα ορθογώνιο παραλληλόγραμμο και η κάθε δυαδική συσχέτιση με ένα ευθύγραμμο τμήμα που ενώνει τις δύο οντότητες.

Στην εικόνα παρακάτω εικόνα παρατίθεται το σχεσιακό ER διάγραμμα όπου απεικονίζονται οι πίνακες συνολικά της βάσης δεδομένων μας , καθώς και οι συσχετίσεις που υπάρχουν μεταξύ τους.



Εικόνα 5.3.1 : Διάγραμμα ER

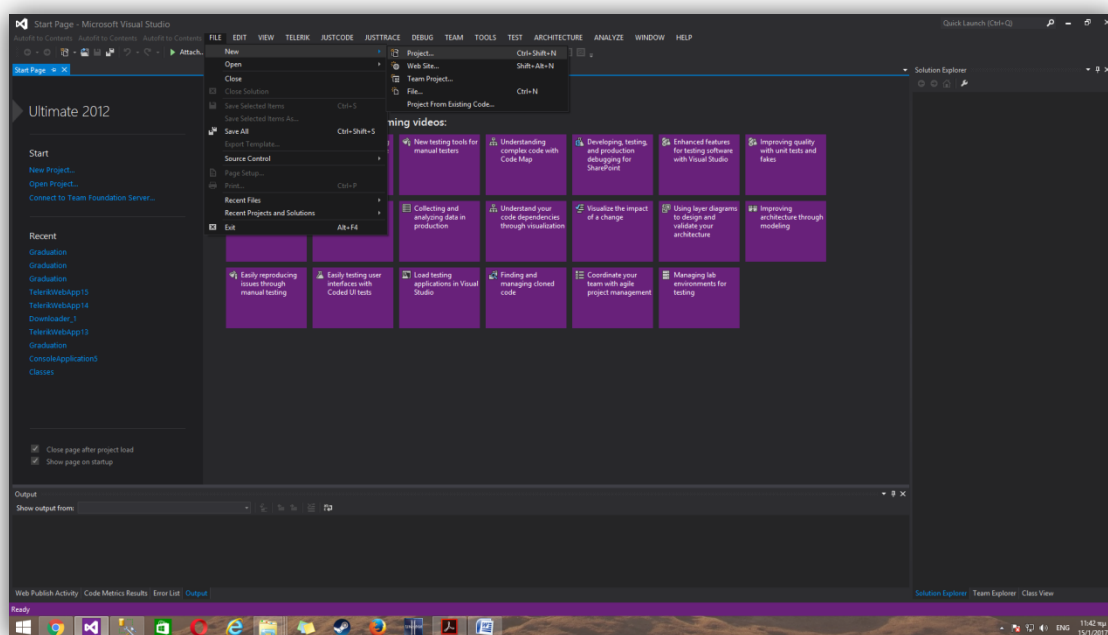
6 Βασική Δομή Εφαρμογής

Σε αυτό το κεφάλαιο πριν αναλυθεί η λειτουργία της κάθε οθόνης ξεχωριστά θα αναλυθεί η δομή της εφαρμογής καθώς και ο τρόπος με τον οποίο πραγματοποιήθηκε η εργασία. Πιο συγκεκριμένα, θα αναλυθεί το πώς δημιουργήθηκε ένα νέο project με το visual studio, πώς είναι δυνατόν να δημιουργηθεί μία master page, μία content page, μία web form αλλά και ένα user control και παράλληλα θα αναφερθούν τα κύρια χαρακτηριστικά τους. Ακόμα, θα παρουσιαστεί ο τρόπος λειτουργίας των διάφορων controls που χρησιμοποιήθηκαν και ο τρόπος με τον οποίο αυτά χρησιμοποιούνται.

6.1 Δημιουργία Project

Για την δημιουργία ενός Νέου Project απαιτούνται τα εξής βήματα :

1. Ανοίγεται το Visual Studio.
2. Επιλέγεται **File** από το κεντρικό menu που βρίσκεται στο πάνω μέρος του Visual Studio > **New** > **Project**.



Εικόνα 6.1.1 : Δημιουργία Project

Αριστερά εμφανίζεται ένα Menu, όπως φαίνεται στην εικόνα παρακάτω. Σε αυτό το Menu υπάρχει μια πληθώρα από επιλογές. Επιλέγεται σαν γλώσσα η **Visual Basic** και έπειτα επιλέγεται **Web** αφού η εφαρμογή που χρησιμοποιείται είναι Web. Αν χρησιμοποιηθεί σαν γλώσσα η C# θα επιλεγόταν αντίστοιχα Visual C# και έπειτα Web.

3. Επιλέγεται **Visual Basic > Web**.

Από την λίστα επιλέγεται :

4. **Telerik Web Form Application**. Αυτό συμβαίνει γιατί έχουν χρησιμοποιηθεί κάποια Control της Telerik αν δεν είχαν χρησιμοποιηθεί αυτά τα εργαλεία θα επιλεγόταν **ASP.NET Web Forms Application**.
5. Δίνεται ένα όνομα στην Εφαρμογή στο textbox **Name** και το μονοπάτι όπου και θα αποθηκευτεί το Project στο textbox **Location**.

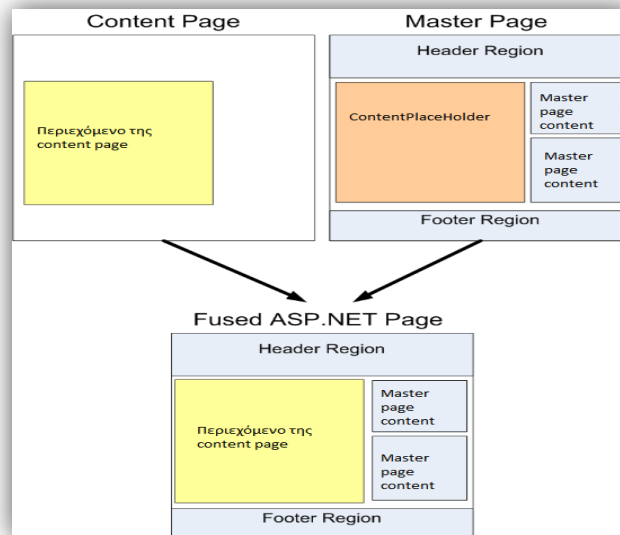
Στο σημείο αυτό αρχίζει να "χτίζεται" η εφαρμογή πάνω στο project που δημιουργήθηκε.

6.2 Master Page

Κατά την δημιουργία της εφαρμογής σε πολλές από τις σελίδες που θα δημιουργηθούν παρατηρείται ότι υπάρχει κοινό περιεχόμενο.

Για να δημιουργηθεί αυτό το κοινό περιεχόμενο υπάρχει η δυνατότητα κάθε φορά που είναι επιθυμητό να αναπαραχθεί, να αντιγραφεί απλά ο markup κώδικας σε όλες τις σελίδες . Όμως αυτό θα δημιουργούσε μια σειρά από προβλήματα, αφού κάθε φορά που θα ήταν απαραίτητο να γίνει αυτή η διαδικασία της αντιγραφής θα έπρεπε να μην παραλείφθει κάποιο κομμάτι αυτού του markup κώδικα. Αντί να επαναλαμβάνεται η διαδικασία αυτή, είναι προτιμότερο να δημιουργηθεί και να χρησιμοποιηθεί μια master page.

Η master page είναι ένας τύπος σελίδας κατά τον οποίο δημιουργείται το κοινό περιεχόμενο μέσα στην σελίδα και έπειτα συνδέεται με μια content page. Με αυτό τον τρόπο οι δύο αυτές σελίδες μοιράζονται κοινό περιεχόμενο. Όπως φαίνεται στην φωτογραφία παρακάτω, δημιουργείται ένα ContentPlaceHolder στην master page όπου ουσιαστικά δηλώνεται απλά η θέση στην οποία στην συνέχεια θα τοποθετηθεί περιεχόμενο σε μια content page σελίδα.

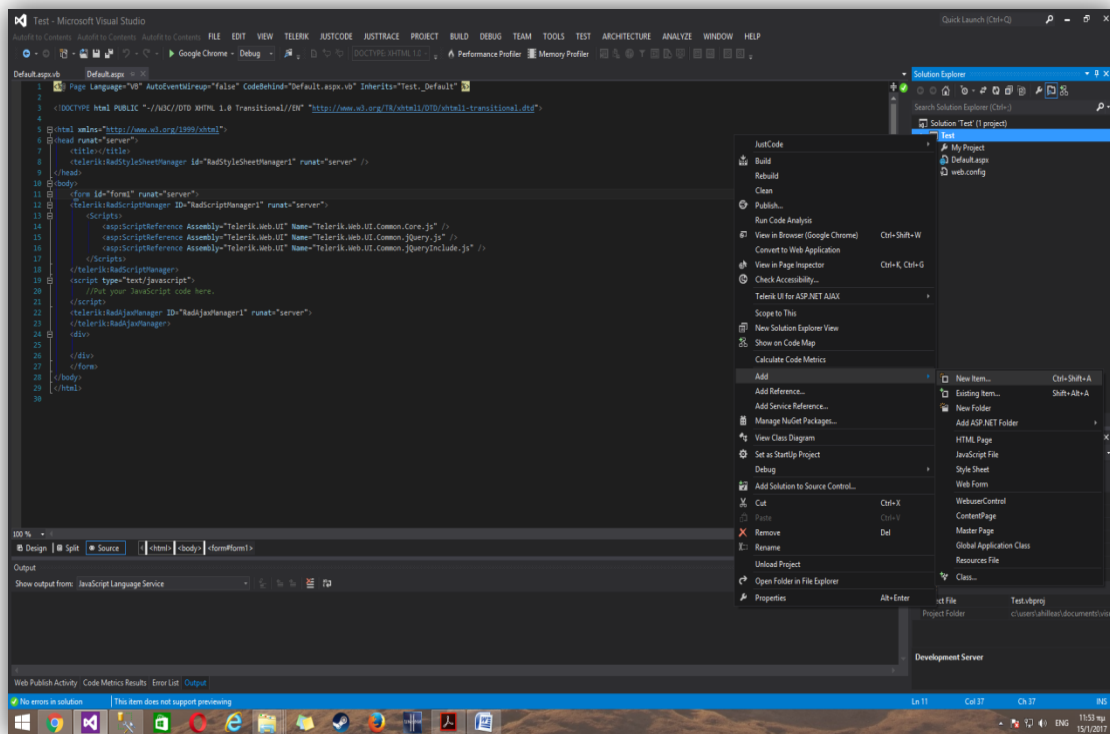


Εικόνα 6.2.1 : Content Page

6.2.1 Δημιουργία Master Page

Για την δημιουργία μιας Master Page απαιτούνται τα εξής βήματα :

1. Δεξί κλικ στο όνομα του Project > Επιλέγεται **Add**.
2. Από το μενού που εμφανίζεται στα δεξιά επιλέγεται **New Item**.

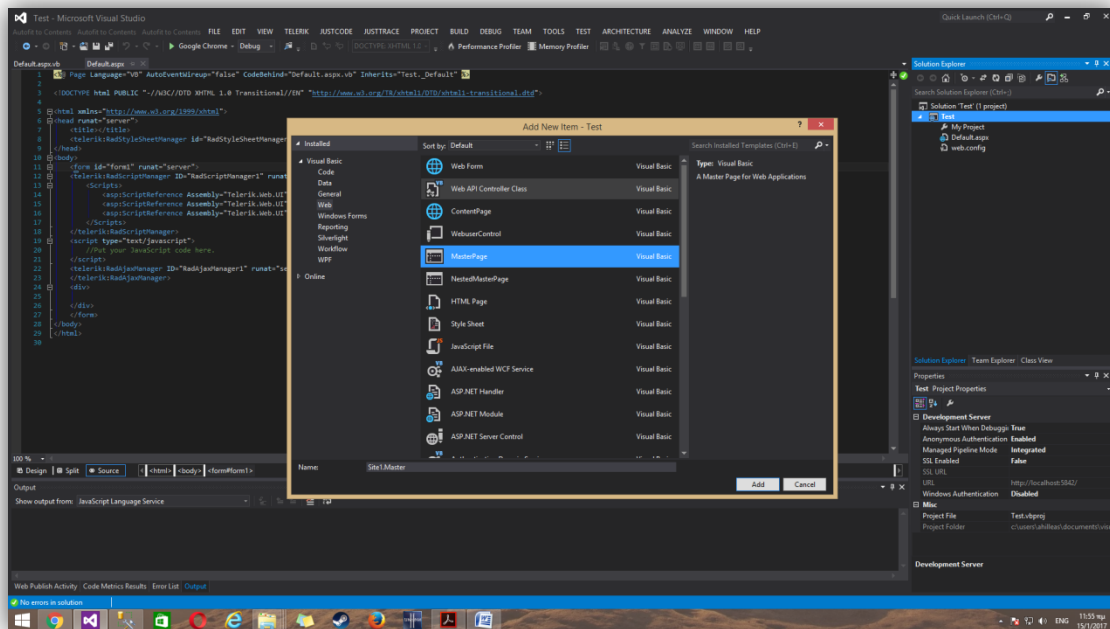


Εικόνα 6.2.1.1 : Δημιουργία Master Page

Από το παράθυρο που θα εμφανιστεί, όπως φαίνεται στην παρακάτω εικόνα επιλέγεται :

3. MasterPage .

4. Δίνεται το όνομα στην MasterPage όπου είναι επιθυμητό στο πεδίο **Name** και επιλέγεται **Add**.



Εικόνα 6.2.1.2 : Δημιουργία Master Page

Στο σημείο αυτό, λοιπόν, είναι δυνατόν να επεξεργαστεί η master page που μόλις δημιουργήθηκε.

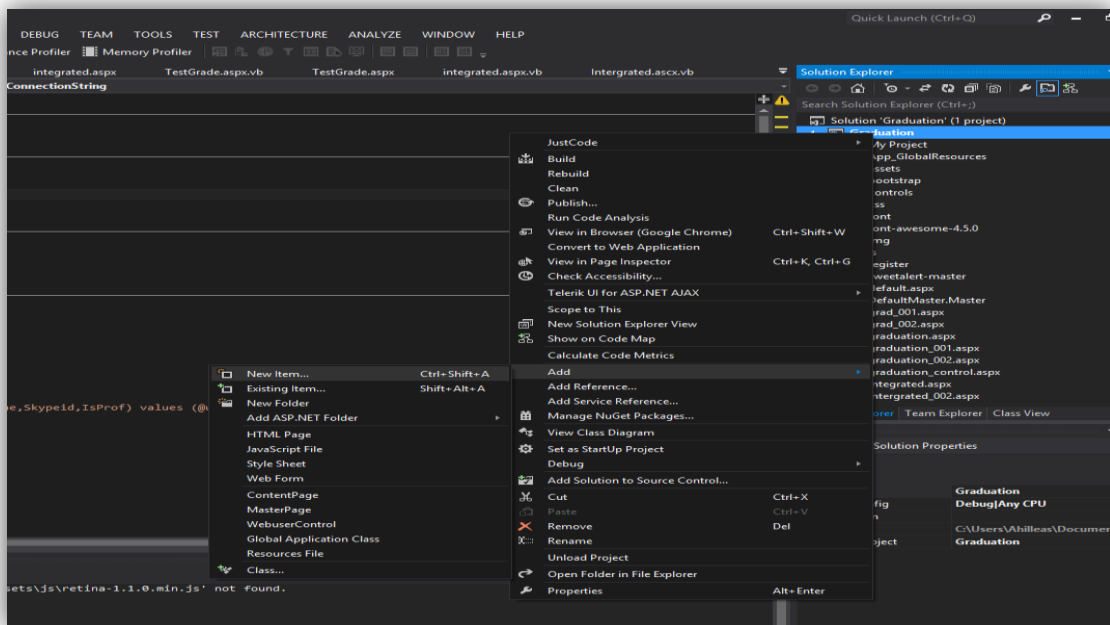
6.3 Content Page

Απαραίτητη προϋπόθεση για την δημιουργία μιας content page είναι η ύπαρξη μιας master page. Ουσιαστικά η content page συνδέεται με την master page και μοιράζονται κοινό περιεχόμενο. Κατά την δημιουργία της master page είχε δημιουργηθεί και ContentPlaceHolder. Αυτό σημαίνει ότι στην content page που θα δημιουργήθηκε θα υπάρχουν και τα αντίστοιχα ContentPlaceHolder που είχαν δημιουργηθεί στην master page.

6.3.1 Δημιουργία Content Page

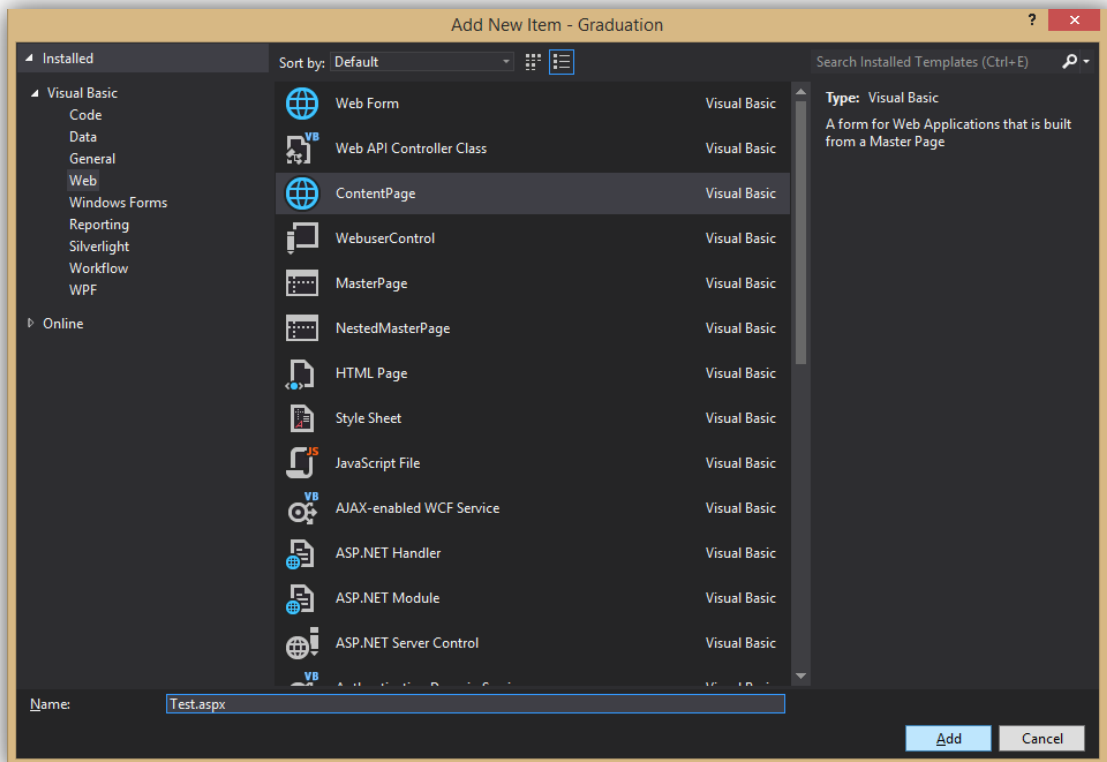
Για να εισαχθεί μια content page στο project :

1. Στο όνομα του project επιλέγεται δεξί κλικ και μετά επιλέγεται από το μενού **Add > New Item**.



Εικόνα 6.3.1.1 : Δημιουργία Content Page

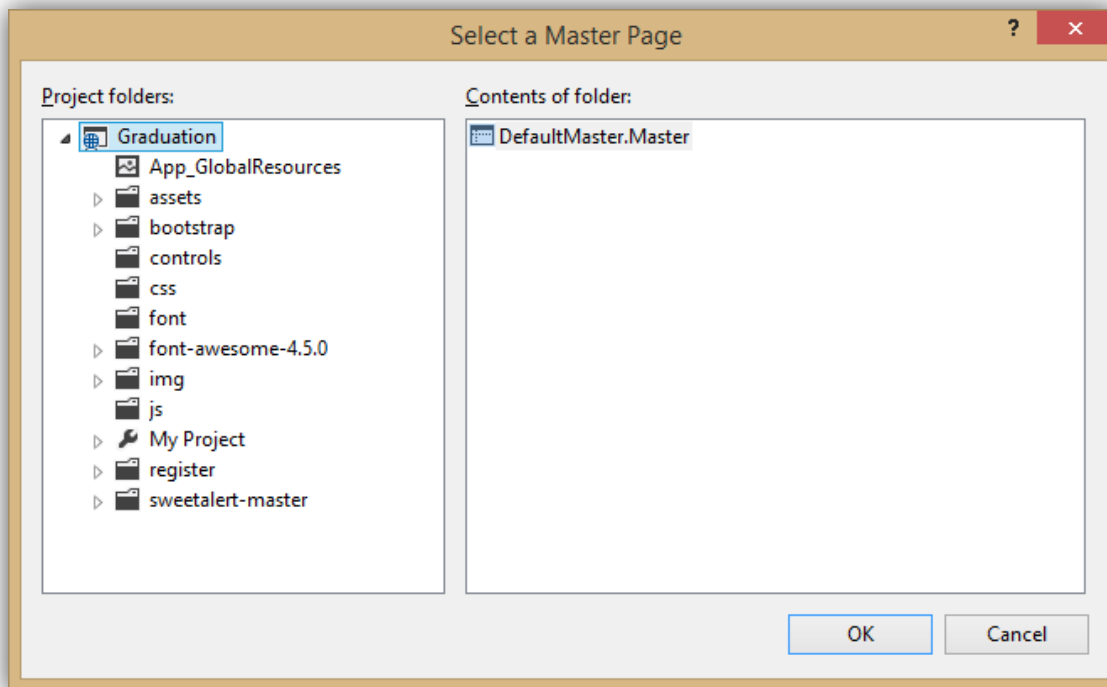
2. Στο παράθυρο που εμφανίζεται, επιλέγεται από την λίστα **ContentPage**.



Εικόνα 6.3.1.2 : Δημιουργία Content Page

3. Δίνεται ένα όνομα στο πεδίο **Name** και επιλέγεται **Add**.

4. Στο επόμενο παράθυρο όπως φαίνεται στην παρακάτω εικόνα, θα εμφανιστεί μια λίστα με τις **master pages** που έχουν δημιουργηθεί. Κατόπιν, επιλέγεται με ποια **master page** θα συνδεθεί η **content page**.



Εικόνα 6.3.1.3 : Δημιουργία Content Page

5. Αφού επιλέγει μια από τις master pages που έχουν δημιουργηθεί, επιλέγεται το **OK** και είναι δυνατόν να επεξεργαστεί η Content Page που δημιουργήθηκε.

Μετά την επεξεργασία της master page θα αναλυθεί πώς δημιουργείται μια κενή content page.

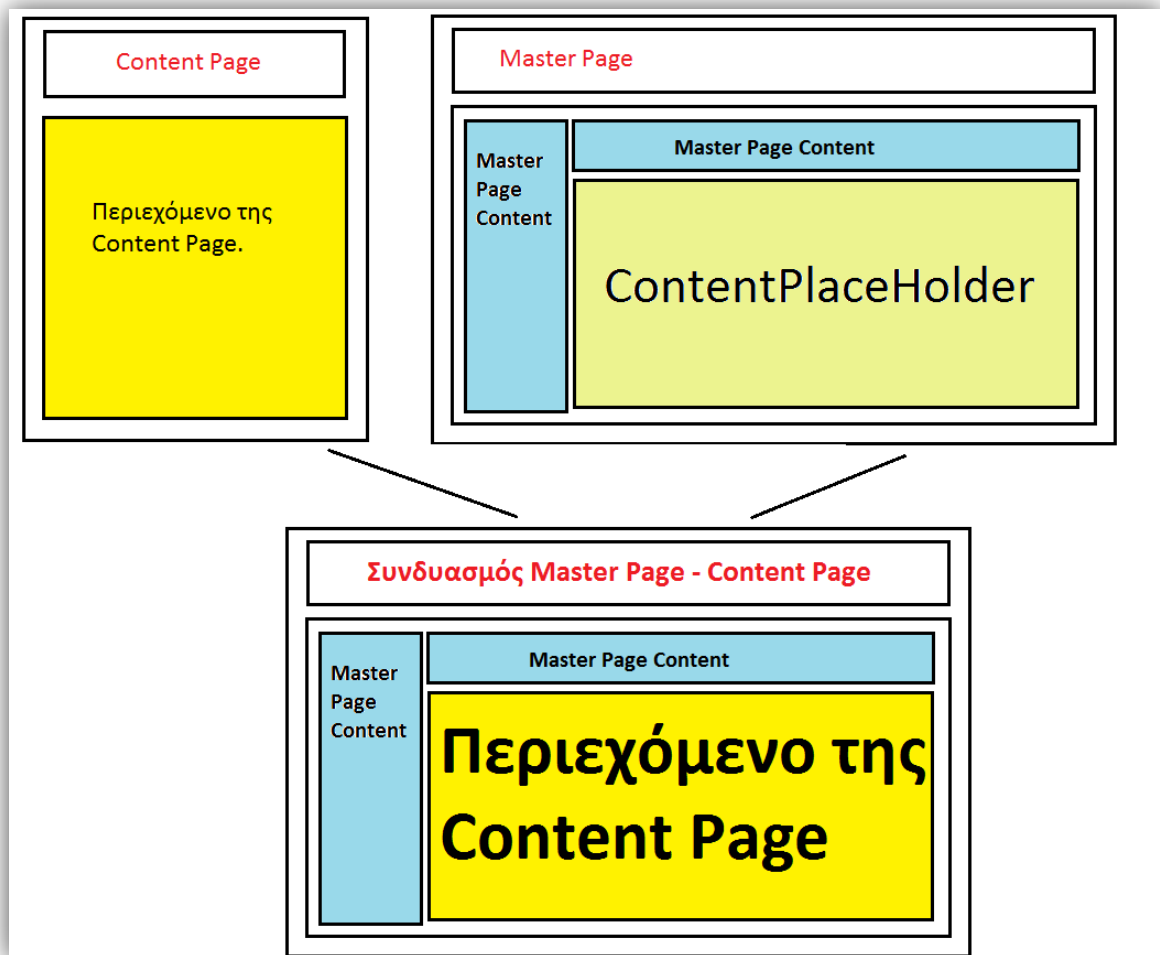
Στην παρακάτω εικόνα φαίνεται αριστερά το κεντρικό μενού πλοήγησης της εφαρμογής και από πάνω μια μπάρα όπου εμφανίζεται το όνομα του χρήστη, που έχει πραγματοποιήσει είσοδο στο σύστημα καθώς και πληροφορίες σχετικά με τα μηνύματά του. Στην μέση υπάρχει ένα λευκό φόντο. Ουσιαστικά, το μενού αριστερά και η μπάρα πάνω είναι στοιχεία της master page δηλαδή το περιεχόμενο που μοιράζεται με την master page η content page σελίδας, ενώ η content page είναι το λευκό φόντο. Μέσα σε αυτό το λευκό φόντο γίνονται οι εργασίες σε μια content page και επεξεργάζεται το περιεχόμενο που είναι επιθυμητό σύμφωνα με τις ανάγκες μας, μέσα πάντα στο ContentPageHolder.



Εικόνα 6.3.1.4 : Content Page Σελίδα Παράδειγμα

Είναι αναγκαίο, λοιπόν, να χρησιμοποιηθεί master page και content page αφού στην εφαρμογή οι περισσότερες σελίδες που θα δημιουργηθούν είναι επιθυμητό να μοιράζονται το περιεχόμενο της master page.

Στην παρακάτω εικόνα παρατίθεται και σε διάγραμμα η ενσωμάτωση μιας content page σελίδας με την master page σελίδα, ώστε να γίνει πιο κατανοητό.



Εικόνα 6.3.1.5 : Συνδυασμός Master Page - Content Page

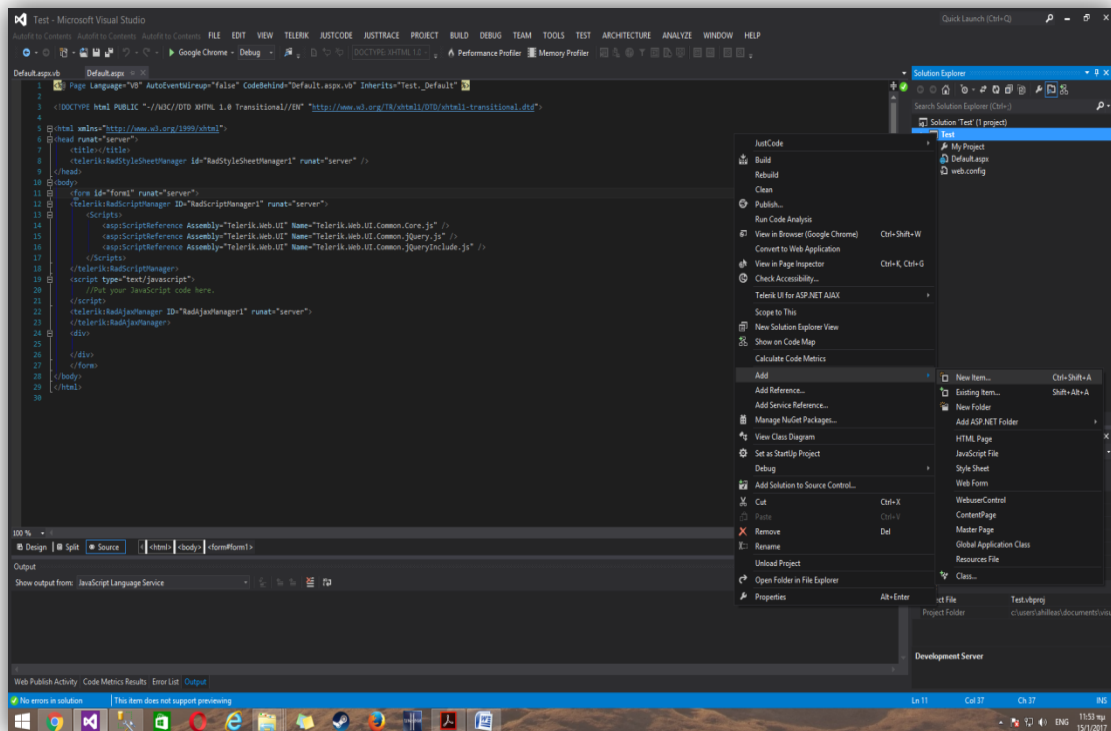
6.4 Web Form

Η διαφορά μιας web form με μια content page, είναι ότι μια web form δεν μοιράζεται κοινό περιεχόμενο με την master page. Λειτουργεί δηλαδή σαν μια ανεξάρτητη σελίδα.

6.4.1 Δημιουργία Web Form

Για την δημιουργία μιας web form απαιτούνται τα εξής βήματα :

1. Δεξί κλικ στο όνομα του Project > Επιλέγεται **Add**.
2. Από το μενού που εμφανίζεται στα δεξιά επιλέγεται **New Item**.

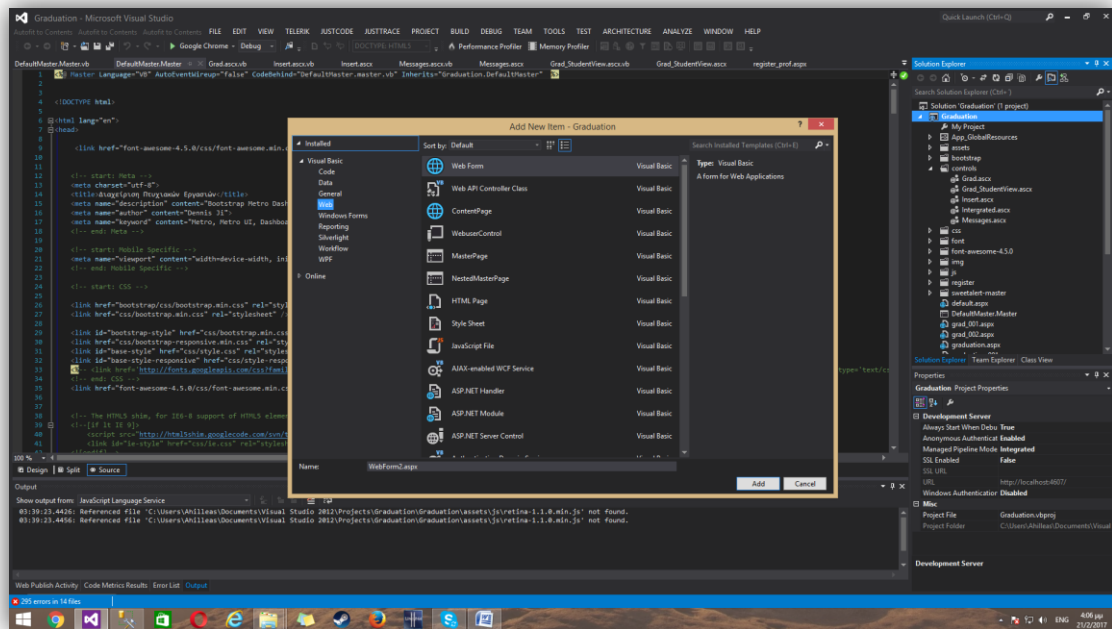


Εικόνα 6.4.1.1 : Δημιουργία Web Form

Από το παράθυρο που θα εμφανιστεί όπως φαίνεται στην παρακάτω εικόνα, θα επιλεγεί :

3. WebForm .

4. Δίνεται το όνομα στην WebForm που είναι επιθυμητό στο textbox **Name** και επιλέγεται **Add**.

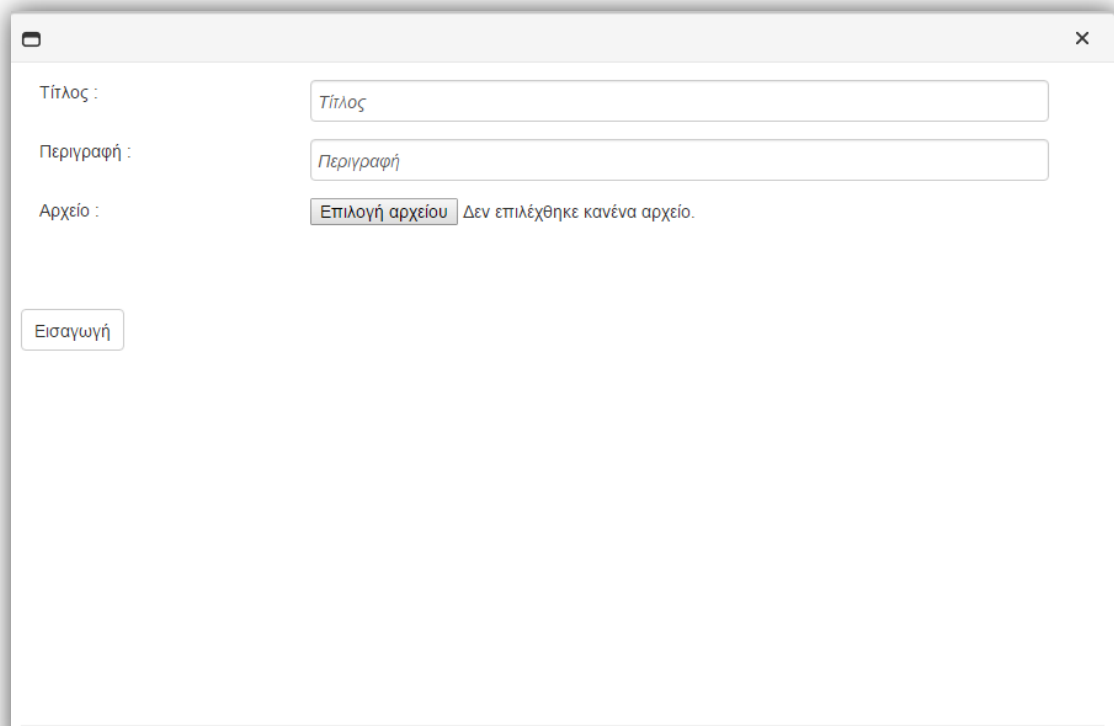


Εικόνα 6.4.1.2 : Δημιουργία Web Form

6.5 User Control

Ένα UserControl είναι μια ανεξάρτητη σελίδα η οποία είναι δυνατόν να δημιουργηθεί και να ανακαλείται ξανά και ξανά από μια web page ή και από άλλες σελίδες. Ένα user control διαμορφώνεται όπως μια κανονική σελίδα, δηλαδή, έχει και αυτό markup κώδικα ο οποίος είναι δυνατόν να διαχειριστεί από κώδικα γραμμένο σε visual basic. Η διαφορά ενός user control από μια κανονική σελίδα είναι ότι το user control μπορεί να συμπεριληφθεί σε πολλές σελίδες ενώ μια σελίδα δεν μπορεί.

Στην εφαρμογή μας χρησιμοποιείται αρκετά συχνά user control. User control χρησιμοποιείται κατά κόρον για να εισαχθούν δεδομένα, για να γίνει ανανέωση σε δεδομένα και για να προβληθούν δεδομένα.

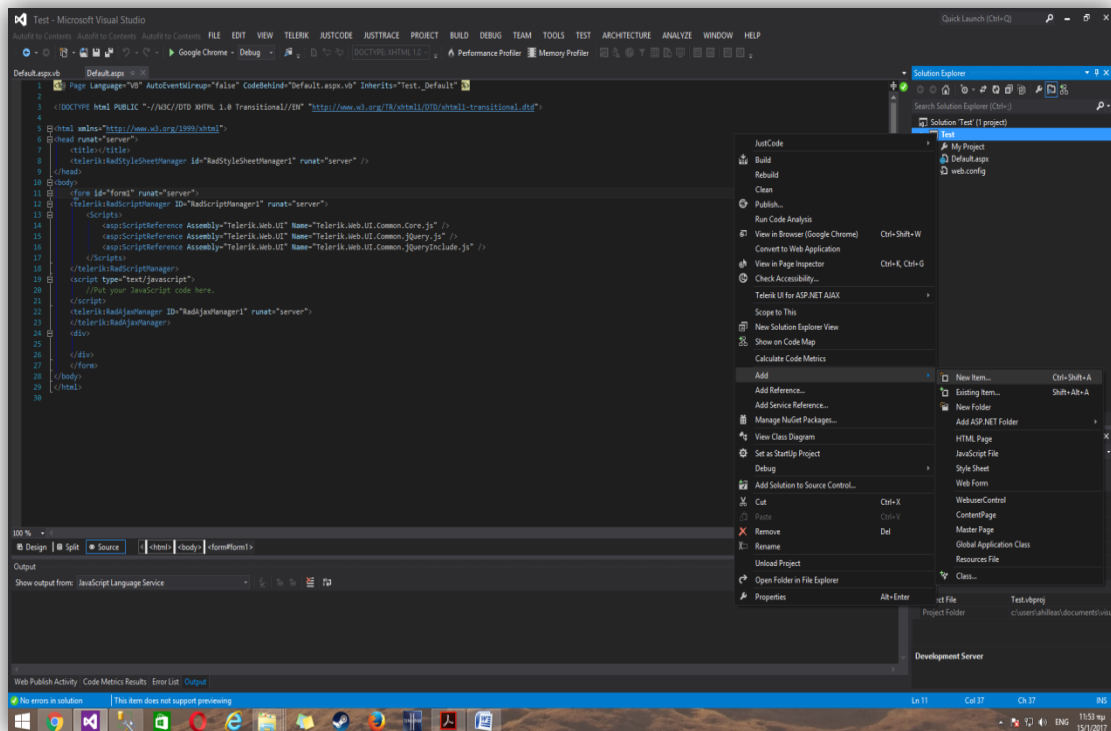


Εικόνα 6.5.1 : Παράδειγμα User Control

6.5.1 Δημιουργία User Control

Για την δημιουργία ενός User Control απαιτούνται τα εξής βήματα :

- 1. Δεξί κλικ στο όνομα του Project > Επιλέγεται Add.**
- 2. Από το μενού που εμφανίζεται στα δεξιά επιλέγεται New Item.**

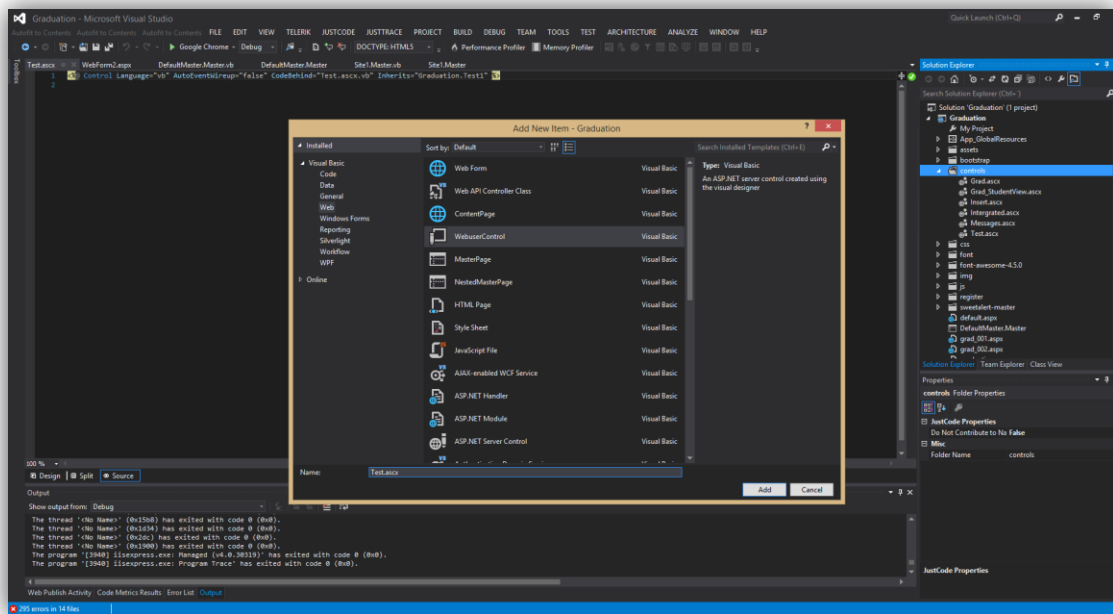


Εικόνα 6.5.1.1 : Δημιουργία User Control

Από το παράθυρο που θα εμφανιστεί όπως φαίνεται στην παρακάτω εικόνα επιλέγεται:

3. WebUserControl .

4. Δίνεται το όνομα στο UserControl που είναι επιθυμητό στο textbox Name και επιλέγεται Add.



Εικόνα 6.5.1.2 : Δημιουργία User Control

Στο σημείο αυτό, θα πραγματοποιηθεί η επεξεργασία του user control που μόλις δημιουργήθηκε.

6.6 ASP.NET Controls

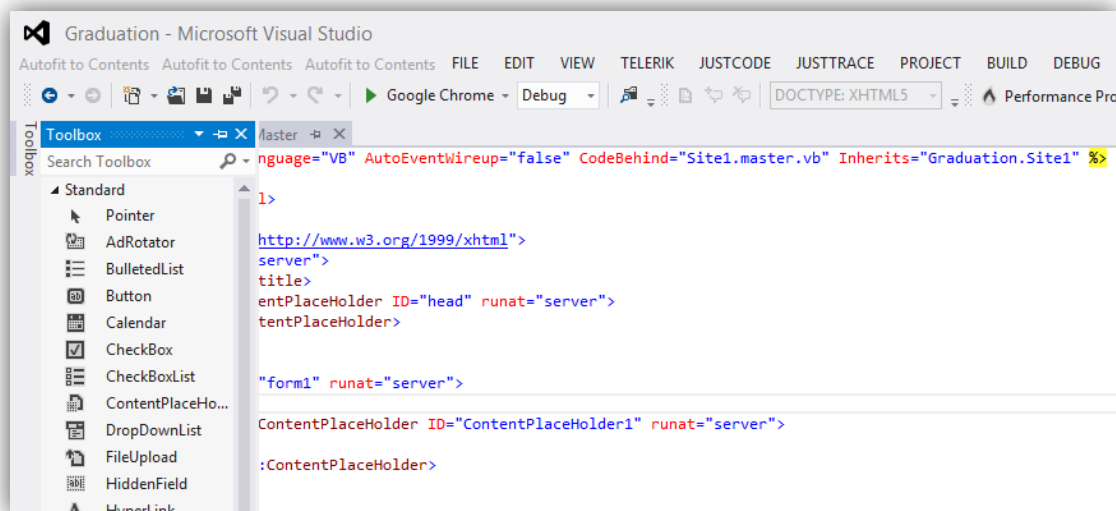
Αριστερά συνήθως από το παράθυρο όπου γράφεται ο κώδικας υπάρχει το Toolbox.

Το Toolbox είναι η εργαλειοθήκη όπου στοιβάζονται όλα τα controls που χρησιμοποιούνται. Τα controls αυτά είναι χωρισμένα σε διάφορες κατηγορίες, όπως **Standard** όπου περιλαμβάνονται τα βασικά controls και τα πιο συνηθισμένα.

Ωστόσο, υπάρχει μια πληθώρα από κατηγορίες όπως η κατηγορία **Data** όπου στοιβάζονται τα controls που θα χρησιμοποιηθούν για να αποκτηθεί πρόσβαση και να υπάρξει αλληλεπίδραση με την βάση δεδομένων και άλλες πολλές κατηγορίες.

Σε περίπτωση που δεν εμφανίζεται η εργαλειοθήκη, για να την εμφανίσουμε από την κεντρική μπάρα που βρίσκεται στο πάνω μέρος του visual studio, επιλέγεται :

View > Toolbox.



Εικόνα 6.6.1 : Εργαλειοθήκη Visual Studio

Ανάλογα με τις ανάγκες της σελίδας χρησιμοποιούνται και τα ανάλογα controls . Επιλέγονται από την λίστα αριστερά τα controls τα οποία θα χρησιμοποιηθούν και μεταφέροντάς τα και αποθέτοντάς τα στην σελίδα μας, αρχίζει η επεξεργασία τους, αλλάζοντας κάποια από τα χαρακτηριστικά τους όπως όνομα και άλλα που θα αναλυθούν παρακάτω.

Τα controls δημιουργούνται μέσα σε ειδικά tags και περιλαμβάνουν κάποια χαρακτηριστικά όπως φαίνεται παρακάτω.

```
<asp:ControlName id="control_id" runat="server"></asp:ControlName>
```

Τα βασικά χαρακτηριστικά είναι τα εξής :

- **ControlName** : Αντιπροσωπεύει το είδος του control. Θα μπορούσε να είναι για παράδειγμα ένα button ή ένα textbox . Θα παρουσιαστούν παρακάτω αρκετά είδη control.
- **id** : Είναι το αναγνωριστικό για κάθε control, πρέπει να είναι μοναδικό για κάθε ένα. Συνήθως δίνεται ένα όνομα σχετικό με την λειτουργία του για να ξεχωρίζεται.
- **runat="server"** : Δεν μπορεί να πάρει άλλη τιμή εκτός από "server". Είναι αναγκαίο για την επικοινωνία με τον server.

Παρακάτω παρατίθεται μια λίστα με ιδιότητες των control πέρα από τις βασικές που αναλύθηκαν.

Ιδιότητα	Περιγραφή
BackColor	Το χρώμα στο φόντο
BorderColor	Το χρώμα του περιγράμματος
BorderStyle	Το style του περιγράμματος
BorderWidth	Το πάχος του περιγράμματος
CssClass	Το όνομα μιας κλάσης CSS που εφαρμόζεται
Enabled	Δηλώνει εάν το control είναι ενεργοποιημένο. Σε περίπτωση που είναι ενεργοποιημένο, ο χρήστης μπορεί να αλληλεπιδρά με το control εφόσον αυτό επιτρέπεται (π.χ. σε Button, TextBox κλπ)
Font	Η γραμματοσειρά με την οποία θα εμφανίζεται το κείμενο του control
ForeColor	Το χρώμα προσκηνίου
Height	Το ύψος που έχει το control όταν σχεδιάζεται
Width	Το πλάτος που έχει το control όταν σχεδιάζεται

Πίνακας 6.6.1 : Βασικές Ιδιότητες Web Sever Controls

Τα web control της asp.net έχουν και κάποιες έξτρα ιδιότητες και λειτουργίες από τα control αυτά της html . Θα δοθεί ένα χαρακτηριστικό παράδειγμα για να γίνουν αντιληπτές αυτές οι ιδιότητες.

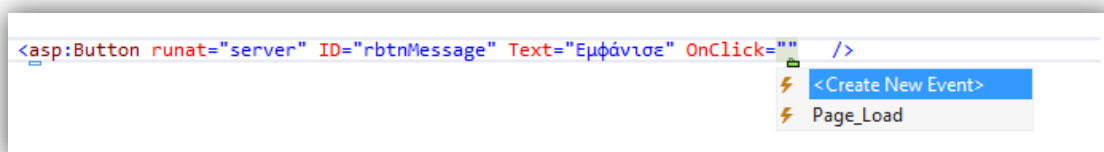
Για παράδειγμα, ένα κουμπί της asp.net έχει κάποια event, δηλαδή γεγονότα. Τα γεγονότα κάθε control είναι δυνατόν να τα βρούμε εύκολα κάνοντας μια αναζήτηση στο διαδίκτυο και ποικίλουν ανάλογα με τον τύπο του control που χρησιμοποιείται. Αυτό σημαίνει ότι τα γεγονότα δεν είναι τα ίδια για όλα τα control.

Χαρακτηριστικό παράδειγμα είναι το γεγονός του πατήματος ενός κουμπιού.

Γενικά οι web εφαρμογές βασίζονται σε τέτοια γεγονότα. Πιο συγκεκριμένα, θα αναλυθεί το παράδειγμα πατήματος ενός κουμπιού και εμφάνισης ενός μηνύματος στον χρήστη.

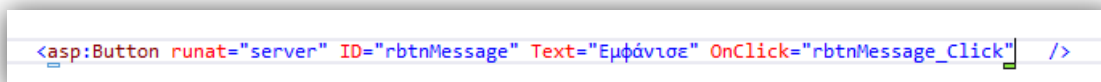
Στο συγκεκριμένο παράδειγμα θα δημιουργηθεί μια web form σελίδα με το όνομα WebForm1. Έπειτα θα προστεθεί το κουμπί στην σελίδα όπου χρησιμοποιούνται τα tags της html και έπειτα στην σελίδα όπου υπάρχει ο κώδικας γραμμένος σε visual basic θα δοθούν οι εντολές για το ποια λειτουργία θα επιτελεί αυτό το κουμπί.

Δημιουργείται, λοιπόν, το κουμπί όπως φαίνεται στην παρακάτω εικόνα και προστίθεται το **OnClick** event. Αυτόματα, δίνεται η επιλογή να δημιουργηθεί ένα νέο event για το συγκεκριμένο κουμπί **Create New Event** και επιλέγεται.



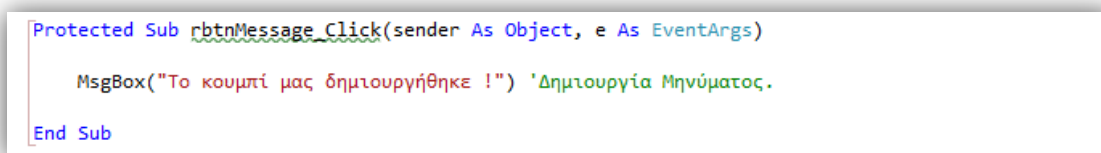
Εικόνα 6.6.2 : Δημιουργία event OnClick

Πλέον αυτό το γεγονός όπως φαίνεται στην παρακάτω εικόνα έχει δημιουργηθεί. Το γεγονός όπως φαίνεται στην παρακάτω εικόνα έχει το όνομα του κουμπιού με την προέκταση Click.



Εικόνα 6.6.3 : Δημιουργία event OnClick

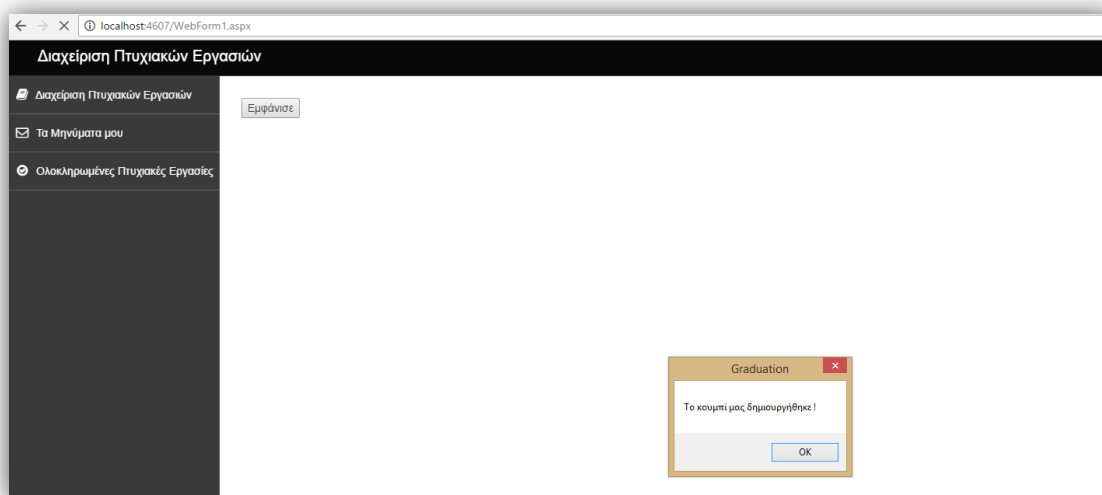
Στην σελίδα όπου υπάρχει ο κώδικας δηλαδή στην **WebForm1.aspx.vb**, κατά την δημιουργία αυτού του event θα δημιουργηθεί κλάση όπου θα γραφτεί κώδικας για να εμφανιστεί ένα μήνυμα .



Εικόνα 6.6.4 : Κλάση κουμπιού

Όταν τεθεί σε λειτουργία η εφαρμογή, όπως φαίνεται στην παρακάτω εικόνα, εμφανίζεται το κουμπί στο αριστερό μέρος της σελίδας.

Πατώντας το κουμπί το αποτέλεσμα θα είναι να εμφανισθεί ένα μήνυμα.



Εικόνα 6.6.5 : Οθόνη WebForm1.aspx

6.6.1 Βασικά Web Server Controls

Παρακάτω παρατίθεται πίνακας με τα server controls που χρησιμοποιούνται συχνότερα στις web εφαρμογές. Τα controls αυτά, όπως προαναφέραμε, χρησιμοποιούνται ανάλογα με τις ανάγκες τις σελίδας μας και με το αποτέλεσμα που είναι επιθυμητό να αποδοθεί.

Web Server Control	Περιγραφή
AdRotator	Δείχνει μία αλληλουχία εικόνων
Button	Δημιουργεί ένα κουμπί
Calendar	Δείχνει ένα ημερολόγιο
CalendarDay	Δείχνει μια μέρα ημερολογίου
CheckBox	Δείχνει ένα check-box

CheckBoxList	Δείχνει μία λίστα ομαδοποιημένων check-box
DataGrid	Δείχνει πεδία δεδομένων από μία πηγή δεδομένων σε ένα πίνακα
DataList	Δείχνει πεδία δεδομένων από μία πηγή δεδομένων χρησιμοποιώντας πρότυπα
DropDownList	Δημιουργεί μια αναδύομενη λίστα
HyperLink	Δημιουργεί ένα σύνδεσμο
Image	Δείχνει μια εικόνα
ImageButton	Δείχνει μία εικόνα στην οποία μπορείς να κάνεις κλικ
Label	Δείχνει στατικό κείμενο (του οποίου το στυλ αλλάζει)
LinkButton	Δημιουργεί ένα κουμπί το οποίο φαίνεται με τη μορφή
ListBox	Δημιουργεί μία λίστα μονής ή πολλαπλής επιλογής
ListItem	Δημιουργεί μια λίστα αντικειμένων
Literal	Δείχνει στατικό κείμενο (του οποίου το στυλ ΔΕΝ αλλάζει)
Panel	Είναι container για τα υπόλοιπα control
PlaceHolder	Κρατάει κενό χώρο για να προστεθούν αντικείμενα προγραμματιστικά
RadioButton	Δημιουργεί radio button
RadioButtonList	Δημιουργεί μία ομάδα radio buttons
BulletedList	Δημιουργεί μία λίστα με κουκίδες
Repeater	Δείχνει μία λίστα αντικειμένων που έχουν

	οριστεί στο control
Style	Δίνει συγκεκριμένη μορφοποίηση στα controls
Table	Δημιουργεί έναν πίνακα
TableCell	Δημιουργεί ένα κελί πίνακα
TableRow	Δημιουργεί γραμμή πίνακα
TextBox	Δημιουργεί ένα κουτάκι εισαγωγής κειμένου
Xml	Δείχνει τα περιεχόμενα ενός XML αρχείου

Πίνακας 6.6.6.1 : Web Server Controls

6.6.2 Validation Server Controls

Οι validation controls χρησιμοποιούνται για έλεγχο. Με την χρήση validation controls εξασφαλίζεται ότι τα δεδομένα που εισάγονται είναι σωστά και δεν θα δημιουργηθεί πρόβλημα ασφαλείας στην εφαρμογή μας.

Εάν τα δεδομένα τα οποία εισάγει ο χρήστης δεν τηρούν τις προϋποθέσεις ασφαλείας, τότε εμφανίζεται μήνυμα λάθους.

Στον παρακάτω πίνακα παρατίθενται όλα τα είδη των validation server controls.

Validation Server Control	Περιγραφή
CompareValidator	Συγκρίνει την τιμή που εισάγει ο χρήστης με την τιμή που έχει εισαχθεί σε κάποιο άλλο control, ή σε μία σταθερή τιμή
CustomValidator	Αφήνει να γράψει ο καθένας τη σύγκριση που θέλει.
RangeValidator	Ελέγχει αν η τιμή που εισήγαγε ο χρήστης είναι μεταξύ δύο τιμών
RegularExpressionValidator	Επιβεβαιώνει ότι η εισαχθείσα τιμή υπάγεται σε κάποια συγκεκριμένη μορφή (πχ Ημερομηνία)

RequiredFieldValidator	Κάνει ένα control υποχρεωτικό να συμπληρωθεί
ValidationSummary	Δείχνει τα μηνύματα σφάλματος (εάν υπάρχουν) των υπολοίπων control συγκεντρωμένα σε ένα σημείο

Πίνακας 6.6.2.1 : Validation Server Controls

6.6.3 Data Source Controls

Τα data source controls επιτρέπουν την σύνδεση της εφαρμογής με πηγές δεδομένων όπως με την βάση δεδομένων. Στον παρακάτω πίνακα παρατίθενται τα κύρια είδη αυτών.

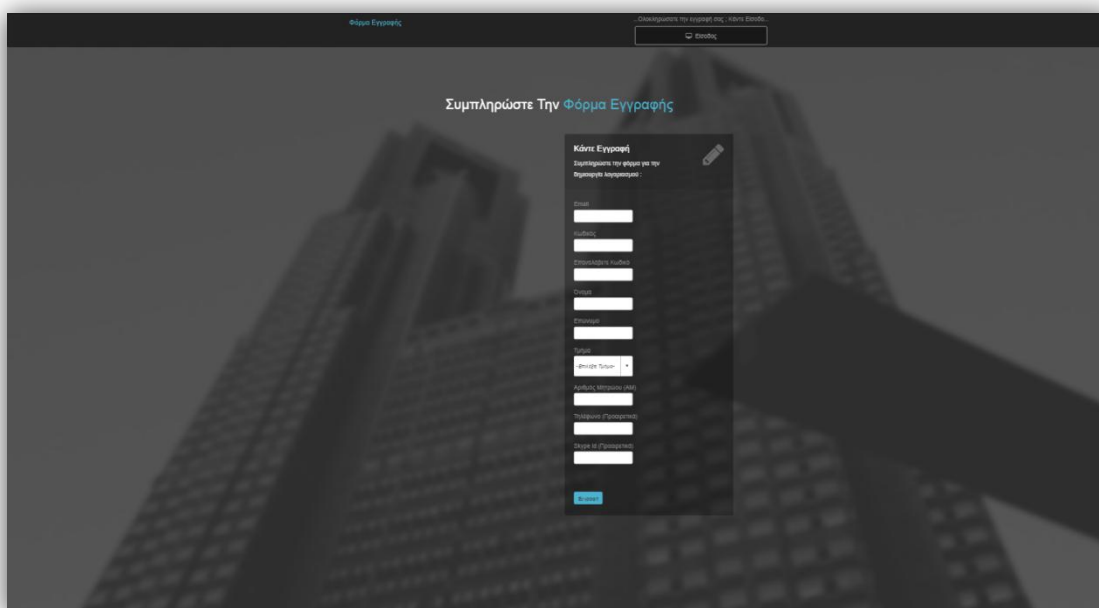
Data Source Control	Περιγραφή
AccessDataSource	Είναι ένα control το οποίο συνδέεται με βάσεις δεδομένων Microsoft Access
SiteMapDataSource	Είναι ένα control το οποίο παίρνει δεδομένα από ένα provider site-map και τα εμφανίζει
SqlDataSource	Είναι ένα control το οποίο συνδέεται με σχεσιακές βάσεις δεδομένων (MS SQL Server, Oracle...)
XmlDataSource	Είναι ένα control το οποίο παίρνει δεδομένα από αρχεία XML

Πίνακας 6.6.3.1 : Είδη Data Source Controls

7 Υλοποίηση Οθονών Εφαρμογής

7.1 Οθόνη - Εγγραφή Φοιτητών

Στοιχεία Σελίδας	
Όνομα σελίδας :	register.aspx
Τύπος σελίδας :	Web Form
User Control :	-



Εικόνα 7.1.1 : Οθόνη Εγγραφής Φοιτητών register.aspx

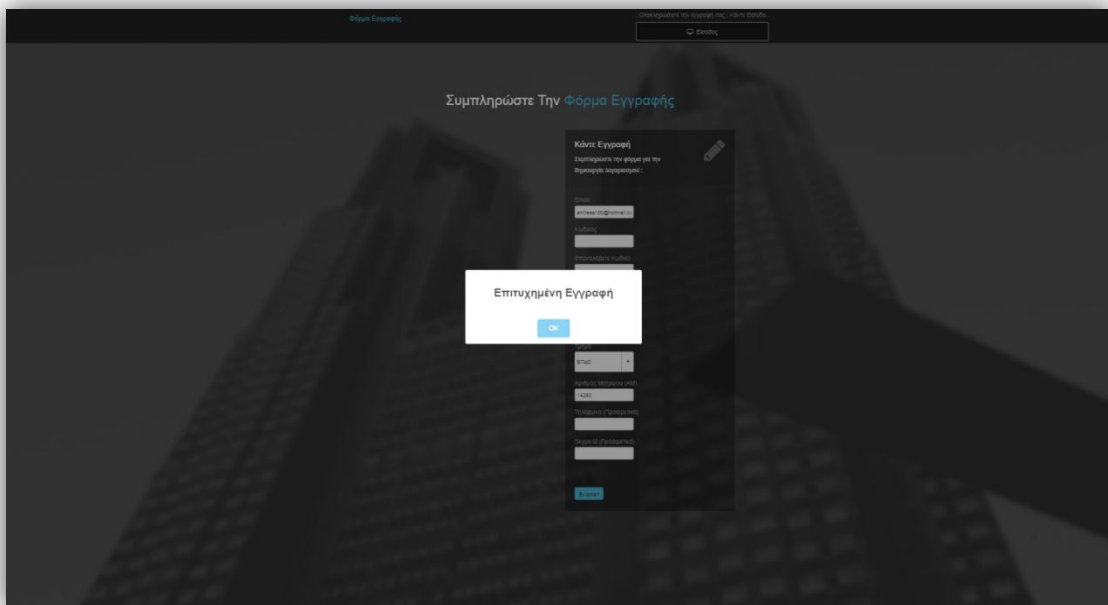
Η register.aspx είναι η σελίδα όπου γίνεται η εγγραφή των φοιτητών. Ο χρήστης φοιτητής κατά την εγγραφή στο σύστημα χρειάζεται να δώσει τα εξής στοιχεία :

- Email (Υποχρεωτικά)
- Κωδικός (Υποχρεωτικά)
- Όνομα (Υποχρεωτικά)
- Επώνυμο (Υποχρεωτικά)

- Τμήμα (Υποχρεωτικά)
- Αριθμό Μητρώου (Υποχρεωτικά)
- Τηλέφωνο (Προαιρετικά)
- Skype id (Προαιρετικά)

Κάποια από αυτά είναι υποχρεωτικά και κάποια είναι προαιρετικά, όπως φαίνεται και παραπάνω.

Ο χρήστης αφού εισάγει στα textbox τα στοιχεία, πατάει το κουμπί **Εγγραφή**. Πατώντας το κουμπί **Εγγραφή**, αν η εγγραφή του στο σύστημα ολοκληρώθηκε με επιτυχία, τότε το σχετικό μήνυμα "**Επιτυχημένη Εγγραφή**" εμφανίζεται σε ένα πλαίσιο διαλόγου.



Εικόνα 7.1.2 : Μήνυμα Επιτυχημένης Εγγραφής

Σε διαφορετική περίπτωση, θα εμφανιστεί το σχετικό μήνυμα "**Αποτυχημένη Εγγραφή**" και φυσικά η αιτία για την οποία απέτυχε η εγγραφή του στο σύστημα.

register.aspx (markup κώδικας)

Στο μπροστά μέρος της εφαρμογής, υπάρχουν τα textbox όπου δίνουν τα στοιχεία τους οι χρήστες κατά την εγγραφή τους στο σύστημα. Ωστόσο, υπάρχουν και κάποιοι validation controls που ελέγχουν τις τιμές κατά την εισαγωγή τους στο σύστημα.

Τέλος, υπάρχει και το κουμπί "Εγγραφή" που με το πάτημά του γίνεται η εγγραφή των χρηστών στο σύστημα. Στο κομμάτι του κώδικα που ακολουθεί, φαίνονται τα controls αυτά καθώς και τα χαρακτηριστικά τους.

```
div class="form-group">
<div>
<asp:Label ID="Label5" runat="server">Email</asp:Label>
</div>
<telerik:RadTextBox runat="server" ID="rtxtEmail"></telerik:RadTextBox>
<asp:RequiredFieldValidator runat="server" ID="rqrdEmail"
ControlToValidate="rtxtEmail" ErrorMessage="* Υποχρεωτικό Πεδίο" Visible="true"
ForeColor="Red"></asp:RequiredFieldValidator>
</div>
<div class="form-group">
<div>
<asp:Label ID="Label1" runat="server">Κωδικός</asp:Label>
</div>
<telerik:RadTextBox runat="server" ID="rtxtPassword" TextMode="Password">
</telerik:RadTextBox>
<asp:RequiredFieldValidator runat="server" ID="rqrdPassword"
ControlToValidate="rtxtPassword" ForeColor="Red" ErrorMessage="* Υποχρεωτικό
Πεδίο">
</asp:RequiredFieldValidator>
</div>
<div class="form-group">
<div>
<asp:Label ID="Label2" runat="server">Επαναλάβετε Κωδικό</asp:Label>
</div>
<telerik:RadTextBox runat="server" ID="rtxtRepeat">
```

```

TextMode="Password"></telerik:RadTextBox>

<asp:RequiredFieldValidator      runat="server"      ID="rqrdRepeatPass"
ControlToValidate="rtxtRepeat"  ErrorMessage="*      Υποχρεωτικό      Πεδίο"
ForeColor="Red"></asp:RequiredFieldValidator>

</div>

<div class="form-group">

<div>

<asp:Label ID="Label3" runat="server">Όνομα</asp:Label>

</div>

<telerik:RadTextBox runat="server" ID="rtxtFName"></telerik:RadTextBox>

<asp:RequiredFieldValidator      runat="server"      ID="rqrdFName"
ControlToValidate="rtxtFName"  ErrorMessage="*      Υποχρεωτικό      Πεδίο"
ForeColor="Red">

</asp:RequiredFieldValidator>

</div>

<div class="form-group">

<div>

<asp:Label ID="Label4" runat="server">Επώνυμο</asp:Label>

</div>

<telerik:RadTextBox runat="server" ID="rtxtLName"></telerik:RadTextBox>

<asp:RequiredFieldValidator      runat="server"      ID="rqrdLName"
ControlToValidate="rtxtLName"  ErrorMessage="*      Υποχρεωτικό      Πεδίο"
ForeColor="Red">

</asp:RequiredFieldValidator>

</div>

<div class="form-group">

<div>

<asp:Label ID="Label9" runat="server">Τμήμα </asp:Label>

</div>

<telerik:RadComboBox runat="server" ID="rcbxSpeciality" EmptyMessage="- Επιλέξτε
Τμήμα- ">

<Items>

```

```

<telerik:RadComboBoxItem runat="server" Text="ΕΠΔΟ" />
<telerik:RadComboBoxItem runat="server" Text="Λογιστική" />
</Items>
</telerik:RadComboBox>
<asp:RequiredFieldValidator runat="server" ID="RequiredFieldValidator2"
ControlToValidate="rcbxSpeciality" ErrorMessage="* Υποχρεωτικό Πεδίο"
ForeColor="Red">
</asp:RequiredFieldValidator>
</div>
<div class="form-group">
<div>
<asp:Label ID="Label8" runat="server">Αριθμός Μητρώου (ΑΜ)</asp:Label>
</div>
<telerik:RadTextBox runat="server" ID="rtxtAm"></telerik:RadTextBox>
<asp:RequiredFieldValidator runat="server" ID="RequiredFieldValidator1"
ControlToValidate="rtxtAM" ErrorMessage="* Υποχρεωτικό Πεδίο" ForeColor="Red">
</asp:RequiredFieldValidator>
</div>
<div class="form-group ">
<div>
<asp:Label ID="Label6" runat="server">Τηλέφωνο (Προαιρετικά)</asp:Label>
</div>
<telerik:RadNumericTextBox runat="server" ID="rtxtTelephone" MaxLength="12" >
<NumberFormat GroupSeparator="" DecimalDigits="0" />
</telerik:RadNumericTextBox>
</div>
<div class="form-group ">
<div>
<asp:Label ID="Label7" runat="server">Skype id (Προαιρετικά)</asp:Label>
</div>

```



```

<telerik:RadTextBox runat="server" ID="rtxtSkype">
</telerik:RadTextBox>
</div>
<asp:Panel runat="server" CssClass="panel-danger">
<asp:RegularExpressionValidator ID="emailValidator" runat="server"
Display="Dynamic"
ErrorMessage="* Πληκτρολογήστε Έγκυρο τύπο email πχ: Example@Gmail.com"
ForeColor="Red" ValidationExpression="^[\\w\\.\\-]+@[a-zA-Z0-9\\-]+(\\. [a-zA-Z0-9\\-
]{1,})*(\\.[a-zA-Z]{2,3}){1,2}$"
ControlToValidate="rtxtEmail">
</asp:RegularExpressionValidator>
<div>
<asp:CompareValidator runat="server" ControlToValidate="rtxtRepeat"
ControlToCompare="rtxtPassword" ErrorMessage="* Δεν ταιριάζουν οι κωδικοί
πρόσβασης" ForeColor="Red" ID="rqrdCompare"></asp:CompareValidator>
</div>
</asp:Panel>
<telerik:RadButton runat="server" ID="rbtnRegister"
OnClick="rbtnRegister_Click" Text="Εγγραφή" BackColor="#4DB2D0"
BorderColor="#4DB2D0"></telerik:RadButton>

```

Στο τέλος της σελίδας υπάρχουν και τα javascript για τα μηνύματα για επιτυχημένη ή αποτυχημένη εγγραφή.

```

<script type="text/javascript">
    function Success() {
        swal("Επιτυχημένη Εγγραφή")
    }
    function Danger() {

```

```

        swal("Αποτυχημένη Εγγραφή")
    }
</script>

```

register.aspx.vb (visual basic κώδικας)

Στον κώδικα που ακολουθεί, η εντολή που δίνεται με το πάτημα του κουμπιού "Εγγραφή" είναι τα στοιχεία που έδωσε ο χρήστης κατά την εγγραφή στο σύστημα, προκειμένου να εισαχθούν στην βάση δεδομένων και πιο συγκεκριμένα στον πίνακα register. Αυτό γίνεται διαρκώς στο συγκεκριμένο σύστημα. Είναι δυνατό, αντίστοιχα, να γίνει κάποια ενημέρωση στον πίνακα ή κάποια διαγραφή ή ακόμα και να επιλεγθούν κάποια στοιχεία από τους πίνακες, ανάλογα με το τι προτίθεται να γίνει σε κάθε περίπτωση. Παρατηρείται ότι δίνεται η τιμή 0 στο πεδίο IsProf. Αυτό συμβαίνει, επειδή αυτός είναι ο τρόπος που διαχωρίζονται οι χρήστες φοιτητές από τους καθηγητές.

```

Protected Sub rbtnRegister_Click(sender As Object, e As EventArgs)
    If Page.IsValid Then ' Av οι validators είναι έγκυρη τότε θα εκτελεστούν οι
        παρακάτω γραμμές κώδικα
        dsRegister.ConnectionString = ConnectionString
        Dim con As New SqlConnection(ConnectionString) ' Δημιουργία connection
        Dim cmd As New SqlCommand ' Δημιουργία SqlCommand
        cmd = New SqlCommand("Insert into register
        (username,password,FName,LName,Am,Speciality,Telephone,Skypeid,IsProf) values
        (@username,@password,@FName,@LName,@Am,@Speciality,@Telephone,@Skypeid,0)",
        con)
        'Τύπος παραμέτρων
        cmd.Parameters.Add("@username", SqlDbType.NChar)
        cmd.Parameters.Add("@password", SqlDbType.NVarChar)
        cmd.Parameters.Add("@FName", SqlDbType.NVarChar)
        cmd.Parameters.Add("@LName", SqlDbType.NVarChar)
        cmd.Parameters.Add("@Am", SqlDbType.Int)
        cmd.Parameters.Add("@Speciality", SqlDbType.NVarChar)

```

```

cmd.Parameters.Add("@Telephone", SqlDbType.VarChar)
cmd.Parameters.Add("@Skypeid", SqlDbType.NVarChar)
'Τιμές παραμέτρων
cmd.Parameters("@username").Value = (rtxtEmail.Text.Trim)
cmd.Parameters("@password").Value = (rtxtPassword.Text)
cmd.Parameters("@FName").Value = (rtxtFName.Text)
cmd.Parameters("@LName").Value = (rtxtLName.Text)
cmd.Parameters("@Am").Value = rtxtAm.Text
cmd.Parameters("@Speciality").Value = rcbxSpeciality.SelectedItem.ToString

If rtxtTelephone.Text = "" Then ' Έλεγχος αν το πεδίο τηλέφωνο είναι κενό. Αν
είναι τότε θα γίνει εισαγωγή τιμής σαν NULL
cmd.Parameters("@Telephone").Value = DBNull.Value
Else ' Αλλιώς θα γίνει Insert η τιμή που έχουμε δώσει
cmd.Parameters("@Telephone").Value = rtxtTelephone.Text.Trim
End If

cmd.Parameters("@Skypeid").Value = rtxtSkype.Text
con.Open() ' Ανοίγω το Connection
cmd.ExecuteNonQuery() ' Εκτελώ το Insert command
ScriptManager.RegisterStartupScript(Me, Page.GetType, "Script", "Success();",
True) ' Θα εμφανιστεί το μήνυμα επιτυχημένης εγγραφής στο σύστημα
con.Close() ' Κλείνω το Connection
Else
ScriptManager.RegisterStartupScript(Me, Page.GetType, "Script", "Danger();",
True) ' Θα εμφανιστεί το μήνυμα αποτυχημένης εγγραφής στο σύστημα
End If
End Sub

```

7.2 Οθόνη - Εγγραφή Καθηγητών

Στοιχεία Σελίδας	
Όνομα σελίδας :	register_prof.aspx
Τύπος σελίδας :	Web Form
User Control :	-

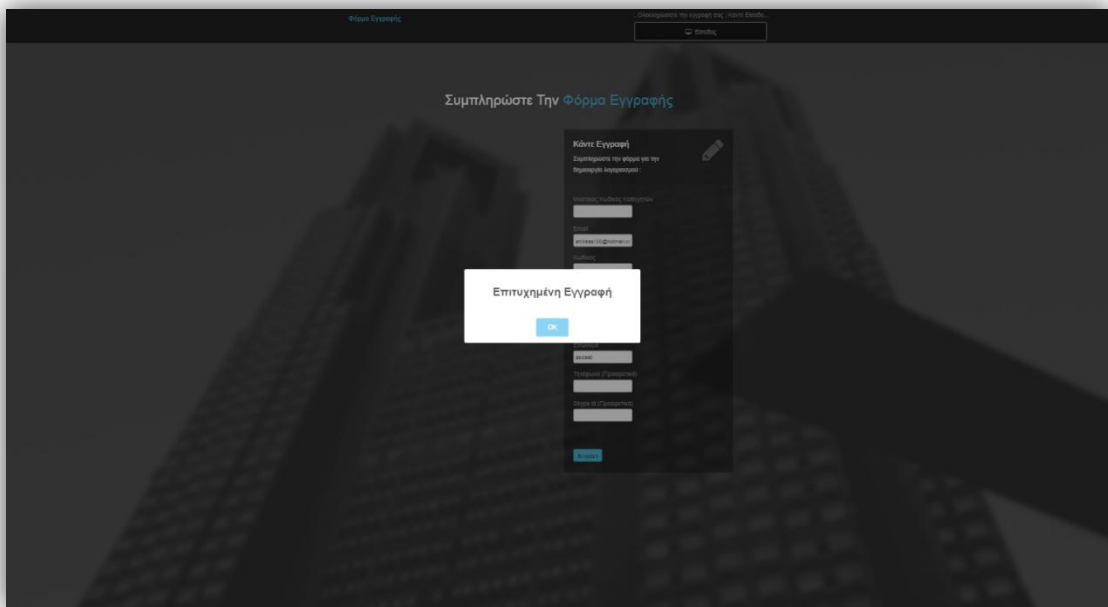
Η register_prof.aspx είναι η σελίδα όπου γίνεται η εγγραφή των καθηγητών και λειτουργεί με παρόμοιο τρόπο με αυτόν στην σελίδα εγγραφής των φοιτητών. Η μόνη διαφορά είναι ότι αλλάζουν τα στοιχεία που θα δώσει ένας καθηγητής κατά την εγγραφή του στο σύστημα. Δεν χρειάζεται να δώσει ο καθηγητής, για παράδειγμα, Αριθμό Μητρώου κατά την εγγραφή του.

Υπάρχει και το πεδίο "Μυστικός Κωδικός Καθηγητών", όπου για να γίνει η εγγραφή στο σύστημα πρέπει ο χρήστης που θα προσπαθήσει να κάνει εγγραφή, να ξέρει αυτόν τον κωδικό. Τον συγκεκριμένο κωδικό, τον γνωρίζουν μόνο οι καθηγητές και είναι απαραίτητος για να κάνουν εγγραφή στο σύστημα.

Ο χρήστης καθηγητής κατά την εγγραφή στο σύστημα χρειάζεται να δώσει τα εξής στοιχεία-κάποια από αυτά είναι υποχρεωτικά και κάποια προαιρετικά :

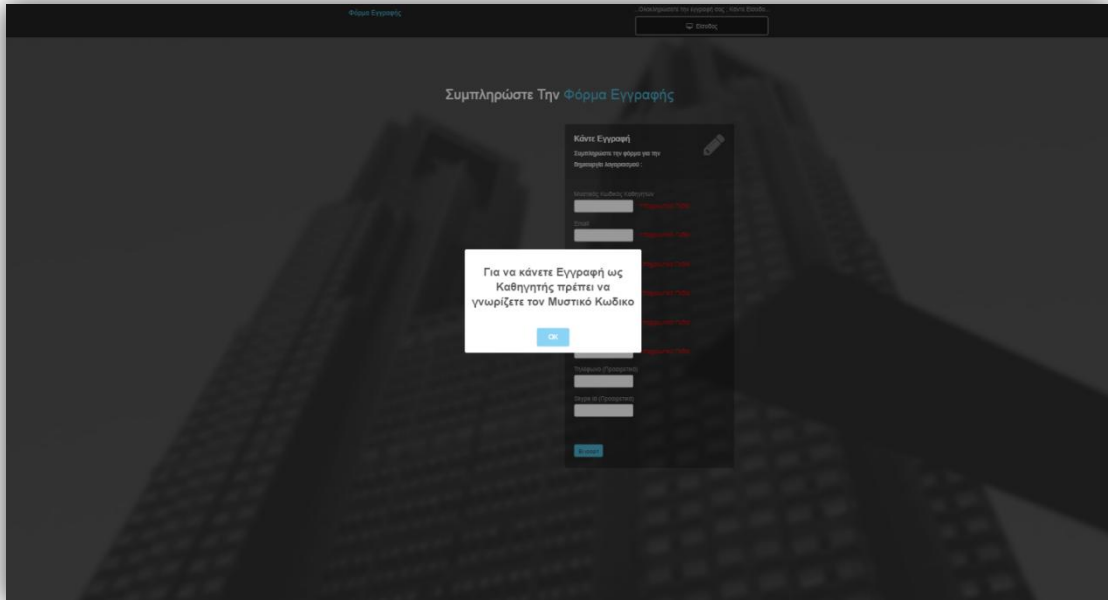
- Μυστικός Κωδικός Καθηγητών(Υποχρεωτικά)
- Email (Υποχρεωτικά)
- Κωδικός (Υποχρεωτικά)
- Όνομα (Υποχρεωτικά)
- Επώνυμο (Υποχρεωτικά)
- Τηλέφωνο (Προαιρετικά)
- Skype id (Προαιρετικά)

Πατώντας το κουμπί "Εγγραφή", αν η εγγραφή του στο σύστημα ολοκληρώθηκε με επιτυχία, τότε το σχετικό μήνυμα "**Επιτυχημένη Εγγραφή**" εμφανίζεται σε ένα πλαίσιο διαλόγου, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 7.2.1 : Μήνυμα Επιτυχημένης Εγγραφής

Σε διαφορετική περίπτωση, θα εμφανιστεί το σχετικό μήνυμα "**Αποτυχημένη Εγγραφή**" και φυσικά η αιτία για την οποία και απέτυχε η εγγραφή του στο σύστημα.



Εικόνα 7.2.2 : Μήνυμα Αποτυχημένης Εγγραφής

Η λογική του κώδικα είναι ίδια με την σελίδα εγγραφής των φοιτητών, το μόνο που αλλάζει είναι τα στοιχεία που θα δώσει ο χρήστης κατά την εγγραφή του στο σύστημα, ωστόσο πρέπει να γνωρίζει και τον μυστικό κωδικό για να ολοκληρώσει την εγγραφή του.

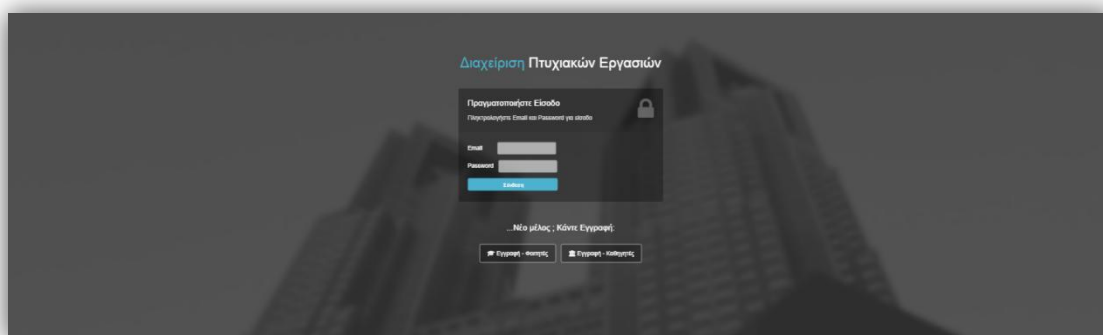
7.3 Οθόνη - Είσοδος Χρηστών

Στοιχεία Σελίδας	
Όνομα σελίδας :	default.aspx
Τύπος σελίδας :	Web Form
User Control :	-

Η σελίδα default.aspx είναι η σελίδα όπου οι χρήστες δίνοντας τα στοιχεία που έχουν ήδη δώσει κατά την εγγραφή στο σύστημα, είτε ως καθηγητές είτε ως φοιτητές, πραγματοποιούν είσοδο στο σύστημα. Αυτή είναι και η κεντρική σελίδα που έχει οριστεί σαν **Start Page**, δηλαδή είναι η σελίδα η οποία όταν "τρέξει" το Project θα εμφανιστεί πρώτη.

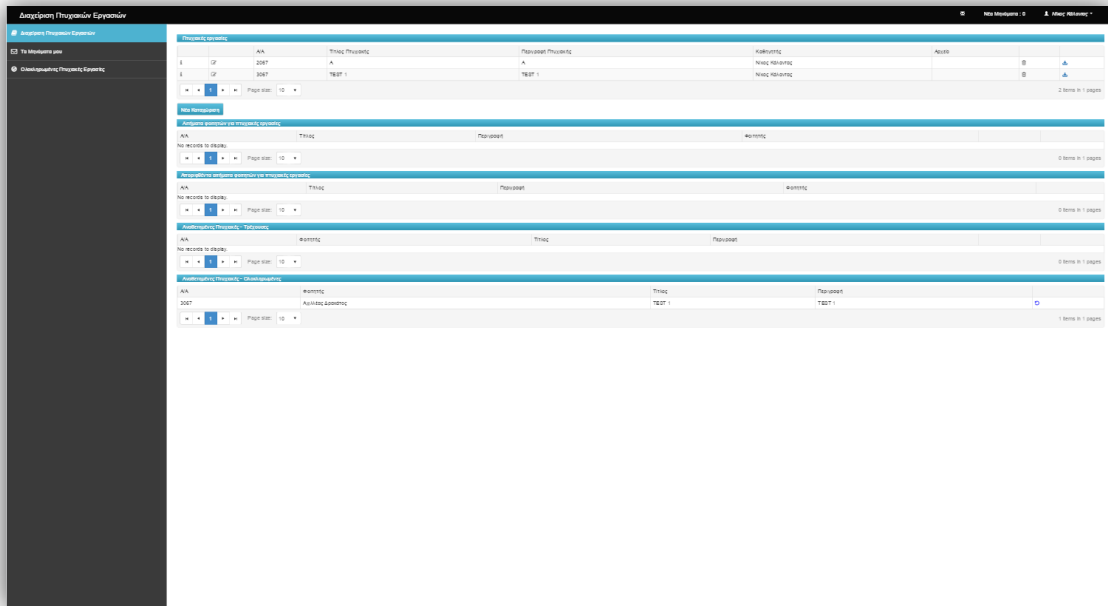
Για να πραγματοποιηθεί η είσοδος των χρηστών και να μπορούν να περιηγηθούν στο σύστημα, είτε ως φοιτητές είτε ως καθηγητές, χρειάζεται να δοθούν τα εξής στοιχεία, με τα οποία πραγματοποιήθηκε και η εγγραφή :

- **E-mail**
- **Password**



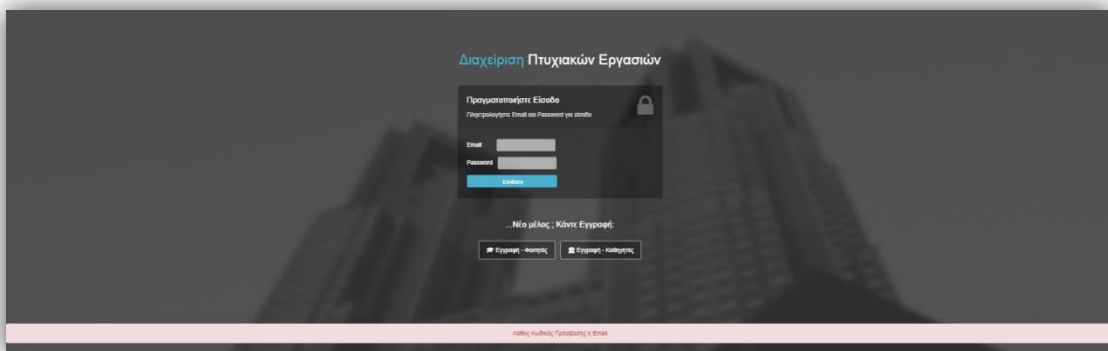
Εικόνα 7.3.1 : Οθόνη Εισόδου Χρηστών default.aspx

Αφού εισαχθούν τα στοιχεία αυτά, ο χρήστης πατάει το κουμπί "Σύνδεση". Αν τα στοιχεία σύνδεσης είναι σωστά, τότε ο χρήστης μεταφέρεται στις οθόνες του συστήματος και μπορεί να περιηγηθεί σε αυτές από το μενού αριστερά, όπως αποτυπώνεται στην παρακάτω εικόνα.



Εικόνα 7.3.1 : Επιτυχημένη Είσοδος Χρήστη

Αν κάποια από τα στοιχεία δεν είναι σωστά, τότε θα αποτραπεί η σύνδεση στο σύστημα και θα εμφανιστεί στο κάτω μέρος της σελίδας το μήνυμα "Λάθος Κωδικός Πρόσβασης ή Email" όπως φαίνεται και στην παρακάτω εικόνα.



Εικόνα 7.3.2 : Μήνυμα Αποτυχημένης Εισόδου

default.aspx (markup κώδικας)

Στο κομμάτι του κώδικα που παρατίθεται παρακάτω, υπάρχουν δυο textbox με τα αντίστοιχα label, όπου ο χρήστης εισάγει το email και τον κωδικό ώστε να αποκτήσει πρόσβαση στο σύστημα.

Υπάρχει ωστόσο και το κουμπί "Σύνδεση" που πατώντας το πραγματοποιείται η είσοδος στην εφαρμογή.

```
<label class="sr-only" for="form-username">Username</label>
<asp:Label ID="Label1" runat="server" Text="Email " ForeColor="White">
</asp:Label>
<telerik:RadTextBox runat="server" ID="rtxtUser" CssClass="opacity"
ForeColor="Black" FocusedStyle-CssClass="unfocused" HoveredStyle-
CssClass="unfocused">
</telerik:RadTextBox>
<label class="sr-only" for="form-password">Password</label>
<asp:Label ID="Label2" runat="server" Text="Password " ForeColor="White">
</asp:Label>
<telerik:RadTextBox runat="server" ID="rtxtPass" TextMode="Password"
CssClass="opacity" ForeColor="Black" HoveredStyle-CssClass="unfocused"
FocusedStyle-CssClass="unfocused">
</telerik:RadTextBox>
<telerik:RadButton runat="server" ID="rbtnLogin" OnClick="rbtnLogin_Click"
Text="Σύνδεση" Width="44%" BackColor="#4DB2D0" BorderColor="#4DB2D0"
ForeColor="White"></telerik:RadButton>
```

default.aspx.vb (visual basic κώδικας)

Πατώντας το κουμπί "Σύνδεση" όπως φαίνεται στον κώδικα παρακάτω, γίνεται μια σειρά από ελέγχους.

Πρώτα από όλα, ελέγχεται η αντιστοιχία των στοιχείων που έδωσε ο χρήστης κατά την είσοδό του. Αν το αποτέλεσμα του ελέγχου είναι επιτυχές και υπάρχει εγγεγραμμένος χρήστης με τα στοιχεία που δόθηκαν, πραγματοποιείται ένας δεύτερος έλεγχος σχετικά με

την ιδιότητα του χρήστη -φοιτητής ή καθηγητής- και αποκτάται πρόσβαση στις αντίστοιχες σελίδες.

```
Protected Sub rbtnLogin_Click(sender As Object, e As EventArgs) ' Πατώντας το
κουμπί για σύνδεση γίνονται τα παρακάτω

dsGrad.ConnectionString = ConnectionString ' Ορίζω την σύνδεση

Dim con As New SqlConnection
Dim con1 As New SqlConnection(ConnectionString)
Dim con2 As New SqlConnection(ConnectionString)
Dim con4 As New SqlConnection(ConnectionString)

Dim cmd As New SqlCommand
Dim cmd1 As New SqlCommand
Dim cmd2 As New SqlCommand
Dim cmd3 As New SqlCommand
Dim cmd4 As New SqlCommand

Dim rd As SqlDataReader ' Δημιουργία SqlDataReader
Dim CnID As Integer ' Ορίζω έναν ακέραιο
Dim FName As String ' Ορίζω ένα String
Dim LName As String ' Ορίζω ένα String
Dim IsProf As Integer ' Ορίζω έναν ακέραιο
Dim dt As New DataTable ' Δημιουργία Datatable

con.ConnectionString = ConnectionString
cmd.Connection = con

con.Open() ' Ανοίγω το Connection

cmd.CommandText = "Select username , password from auth where username='" &
rtxtUser.Text & "' and password='" & rtxtPass.Text & "' " ' Επιλέγω απο τα
textbox τον κωδικό και το email

rd = cmd.ExecuteReader

If rd.HasRows Then ' Αν υπάρχει εγγεγραμμένος χρήστης με τα στοιχεία που δώσαμε
στα textbox τότε γίνονται τα παρακάτω :
```

```

cmd1 = New SqlCommand("Select CnID from register where password=@password and
username=@username ", con1) ' Επιλέγω τον μοναδικό κωδικό CnID
cmd1.Parameters.Add("@username", SqlDbType.NVarChar)
cmd1.Parameters.Add("@password", SqlDbType.NVarChar)
cmd1.Parameters("@username").Value = (rtxtUser.Text)
cmd1.Parameters("@password").Value = (rtxtPass.Text)
con1.Open()
CnID = cmd1.ExecuteScalar
con1.Close()
Session.Add("CnID", CnID) ' Προσθέτω το CnID σε Session
cmd2 = New SqlCommand("Select FName from register where CnID=@CnID", con2)
cmd3 = New SqlCommand("Select LName from register where CnID=@CnID", con2)
cmd2.Parameters.Add("@CnID", SqlDbType.Int)
cmd2.Parameters("@CnID").Value = Integer.Parse(Me.Session("CnID"))
cmd3.Parameters.Add("@CnID", SqlDbType.Int)
cmd3.Parameters("@CnID").Value = Integer.Parse(Me.Session("CnID"))
con2.Open() ' Ανοίγω σύνδεση
Fname = cmd2.ExecuteScalar ' Αποθηκεύω το όνομα στην μεταβλητή FName
LName = cmd3.ExecuteScalar ' Αποθηκεύω το επώνυμο στην μεταβλητή LName
con2.Close() ' Κλείνω σύνδεση
Session.Add("FName", FName) 'Περνάω σε Session το όνομα
Session.Add("LName", LName) 'Περνάω σε Session το επώνυμο
'Περνάω σε Session το IsProf Αν είναι 1 τότε είναι καθηγητής αν είναι 0 τότε
είναι φοιτητής
cmd4 = New SqlCommand("Select IsProf from register where CnID=@CnID", con4)
cmd4.Parameters.Add("@CnID", SqlDbType.Int)
cmd4.Parameters("@CnID").Value = Integer.Parse(Me.Session("CnID"))
con4.Open()
IsProf = cmd4.ExecuteScalar
con4.Close()

```

```

Session.Add("IsProf", IsProf)

Else ' Αν δεν υπάρχει εγγεγραμμένος χρήστης με τα στοιχεία που δώσαμε τότε
εμφανίζεται σε panel μήνυμα αποτυχίας εισόδου

panelCode.Visible = True

Exit Sub

End If

'Έλεγχος για StartPage'

If Me.Session("IsProf") = 0 Then

' Έλεγχος που θα γίνεται για τους χρήστες φοιτητές : θα ελέγχεται αν έχουν
δηλώσει πτυχιακή / Αν έχουν δηλώσει πτυχιακή θα τους κάνει Redirect στην
myprofile.aspx / αλλιώς graduation.aspx

Dim con6 As New SqlConnection(ConnectionString)

Dim cmd6 As SqlCommand

Dim check As Integer

cmd6 = New SqlCommand("Select CnIDStudent from graduate_completed where
CnIDStudent=@CnID ", con6)

cmd6.Parameters.Add("@CnID", SqlDbType.Int)

cmd6.Parameters("@CnID").Value = Me.Session("CnID")

con6.Open()

check = cmd6.ExecuteScalar()

con6.Close()

If check = 0 Then ' Αν το check=0 τότε δεν έχει δηλώσει πτυχιακή ο φοιτητής άρα
graduation.visible = True else graduation.visible = false

Me.Context.Response.Redirect("~/graduation.aspx")

'StartPage.HRef = "graduation.aspx"

Else

Me.Context.Response.Redirect("~/myprofile.aspx")

'StartPage.HRef = "myprofile.aspx"

End If

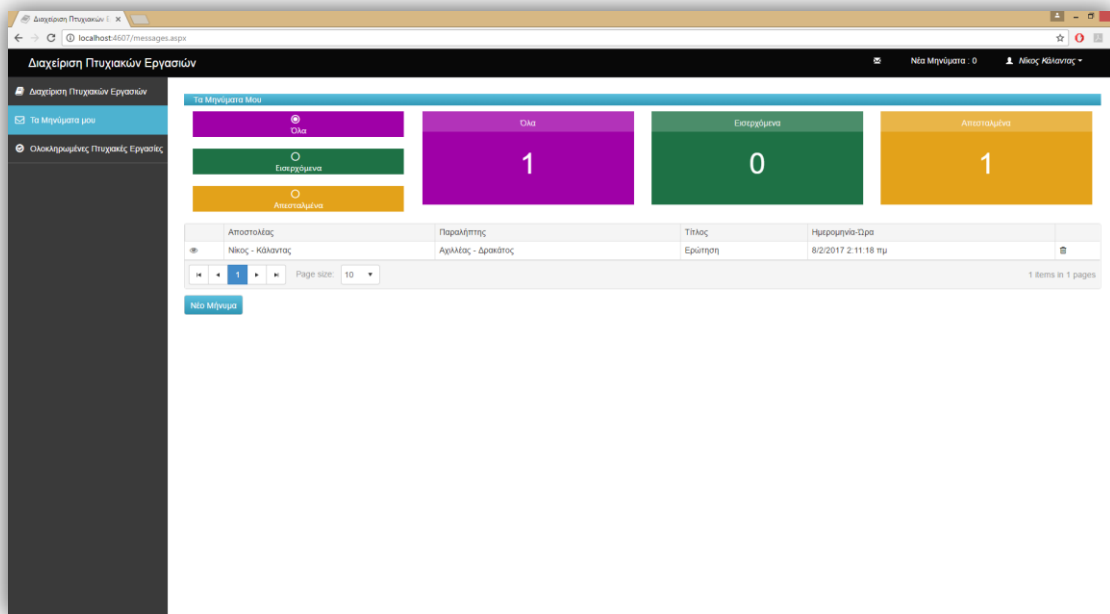
End If

```

```
'Τέλος Ελέγχου φοιτητών'  
'Έλεγχος Καθηγητών'  
If Me.Session("IsProf") = -1 Then  
Me.Context.Response.Redirect("~/graduation_001.aspx")  
End If  
'Τέλος ελέγχου καθηγητών'  
End Sub
```

7.4 Οθόνη - Τα Μηνύματα Μου

Στοιχεία Σελίδας	
Όνομα σελίδας :	messages.aspx
Τύπος σελίδας :	Content Page
User Control :	Messages.ascx



Εικόνα 7.4.1 : Οθόνη Μηνυμάτων messages.aspx

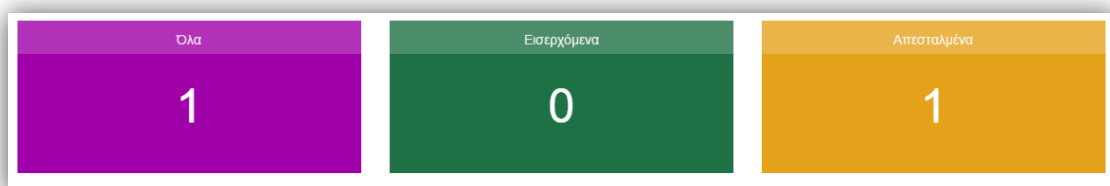
Η σελίδα messages.aspx είναι η σελίδα όπου οι χρήστες διαχειρίζονται τα μηνυμάτά τους . Η συγκεκριμένη σελίδα είναι κοινή για τους χρήστες καθηγητές αλλά και για τους χρήστες φοιτητές.

Σε αυτήν την σελίδα ο χρήστης έχει τη δυνατότητα είτε να στείλει ένα νέο μήνυμα, είτε να διαβάσει κάποιο από τα μηνύματα που του έχουν αποσταλεί. Ακόμα μπορούν οι χρήστες να ανταλλάξουν αρχεία μεταξύ τους μέσω των μηνυμάτων.

Τα μηνύματα χωρίζονται σε τρεις κατηγορίες :

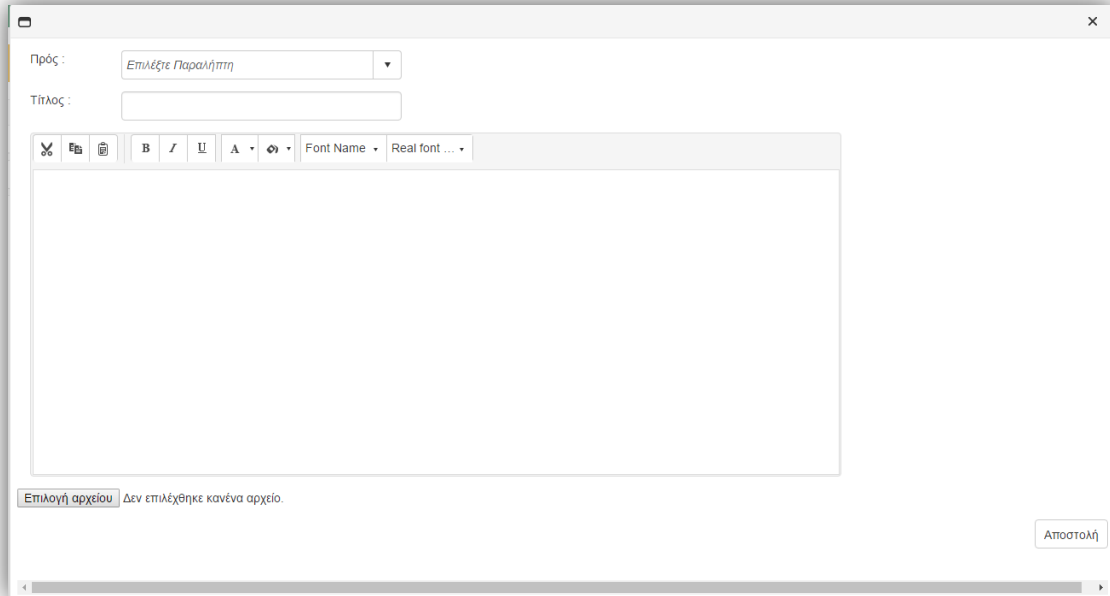
- Όλα
- Εισερχόμενα
- Απεσταλμένα

Μία επιπλέον λειτουργία στην οθόνη των μηνυμάτων, είναι η καταμέτρηση των μηνυμάτων. Ο αριθμός των μηνυμάτων που έχει στείλει , είτε έχει λάβει ο χρήστης εμφανίζεται σε κελιά.



Εικόνα 7.4.2 : Λειτουργία Καταμέτρησης Μηνυμάτων

Ο χρήστης πατώντας στο κουμπί "Νέο Μήνυμα" ανοίγει το user control **Messages.ascx**



Εικόνα 7.4.2 : User Control Μηνυμάτων Messages.ascx

Σε αυτό το control ο χρήστης επιλέγει τον παραλήπτη, δίνει έναν σύντομο τίτλο και εισάγει το θέμα του μηνύματος. Ακόμα, υπάρχει η δυνατότητα επισύναψης κάποιου αρχείου.

Τέλος, πατώντας το κουμπί "**Αποστολή**", αν το μήνυμα αποσταλεί επιτυχώς, εμφανίζεται και το αντίστοιχο μήνυμα σε ένα νέο πλαίσιο διαλόγου με την ένδειξη "**Επιτυχημένη Αποστολή**", είτε το αντίστοιχο μήνυμα "**Αποτυχημένη Αποστολή**" σε περίπτωση αποτυχίας αποστολής του μηνύματος.

7.5 Οθόνη - Ολοκληρωμένες Πτυχιακές Εργασίες

Στοιχεία Σελίδας	
Όνομα σελίδας :	intergrated.aspx
Τύπος σελίδας :	Content Page
User Control :	-

Εικόνα 7.5.1 : Οθόνη Ολοκληρωμένες Πτυχιακές Εργασίες `intergrated.aspx`

Η συγκεκριμένη σελίδα, είναι η σελίδα όπου υπάρχουν όλες οι ολοκληρωμένες πτυχιακές εργασίες των φοιτητών.

Πιο συγκεκριμένα, είναι κάτι σαν portfolio όπου υπάρχει η δυνατότητα να προβληθούν πληροφορίες σχετικά με τις ολοκληρωμένες πτυχιακές εργασίες, καθώς και να γίνει download των ολοκληρωμένων πτυχιακών εργασιών, στις οποίες έχει γίνει upload μετά το πέρας τους από τους καθηγητές.

Σε αυτήν την σελίδα υπάρχουν τρεις λειτουργίες :

- Στατιστικά Στοιχεία
- Αναζήτηση
- Λίστα Ολοκληρωμένων Πτυχιακών

Λίστα Ολοκληρωμένων Πτυχιακών

Όσον αφορά στην λειτουργία "Λίστα Ολοκληρωμένων Πτυχιακών" υπάρχει η δυνατότητα να προβληθούν στοιχεία σχετικά με όλες τις πτυχιακές εργασίες, που έχουν σημειωθεί ως "ολοκληρωμένες" από τους καθηγητές.

Πληροφορίες μπορούν να αποκτηθούν σχετικά με τις ολοκληρωμένες πτυχιακές εργασίες, εμφανίζονται σε μια λίστα και είναι οι εξής :

- Τίτλος Πτυχιακής
- Περιγραφή Πτυχιακής

- Ημερομηνία Ολοκλήρωσης
- Φοιτητής (Όνοματεπώνυμο)
- Καθηγητής (Όνοματεπώνυμο)
- Βαθμός
- Αρχείο (Όνομα αρχείου)

Στην συγκεκριμένη σελίδα, έχουν πρόσβαση είτε χρήστες συνδεδεμένοι ως καθηγητές είτε ως φοιτητές, μόνο που δεν έχουν τα ίδια δικαιώματα σε αυτή τη σελίδα.

Όσον αφορά στους καθηγητές, μπορούν να εισάγουν βαθμολογία αλλά και να επεξεργαστούν τις βαθμολογίες των πτυχιακών εργασιών που έχουν αναθέσει σε έναν φοιτητή και είναι ολοκληρωμένες, καθώς επίσης και να ανεβάσουν ως αρχείο μία ολοκληρωμένη πτυχιακή.

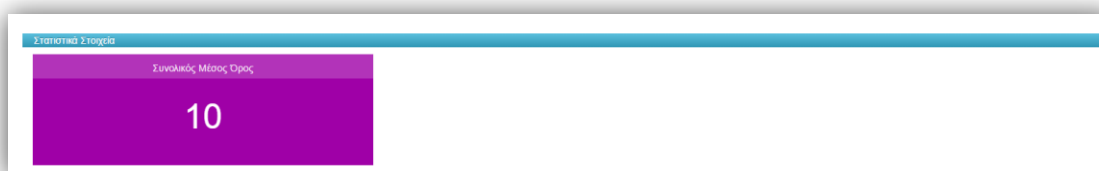
Οι παραπάνω λειτουργίες μπορεί να γίνουν μόνο από τον εκάστοτε καθηγητή που έχει σημάνει μια πτυχιακή ως "ολοκληρωμένη" και το θέμα της πτυχιακής το έχει δημιουργήσει ο ίδιος, αλλά όχι από τους φοιτητές, όπως άλλωστε είναι λογικό.

Τέλος, αν ο καθηγητής έχει ανεβάσει σαν αρχείο μία ολοκληρωμένη πτυχιακή εργασία, μπορεί ο οποιοσδήποτε χρήστης να κατεβάσει το αρχείο αυτό, είτε έχει εισέλθει στο σύστημα ως καθηγητής είτε ως φοιτητής.

Στατιστικά Στοιχεία

Όσον αφορά στην λειτουργία "Στατιστικά Στοιχεία" είναι δυνατό να προβληθεί ο συνολικός μέσος όρος όλων των πτυχιακών εργασιών.

Ο μέσος όρος υπολογίζεται αυτόματα βάσει των ολοκληρωμένων και βαθμολογημένων πτυχιακών εργασιών.



Εικόνα 7.5.2 : Λειτουργία Στατιστικά Στοιχεία

Αναζήτηση

Εικόνα 7.5.3 : Λειτουργία Αναζήτησης

Στην λειτουργία της αναζήτησης είναι δυνατόν να γίνει αναζήτηση των ολοκληρωμένων πτυχιακών εργασιών που ο χρήστης επιθυμεί και υπάρχουν φυσικά στην λίστα με τις "ολοκληρωμένες εργασίες".

Παράλληλα, υπάρχει η δυνατότητα να τις κατεβάσει είτε ως αρχείο PDF, είτε ως αρχείο Word, είτε ως αρχείο Excel, σύμφωνα με τα κριτήρια που ο χρήστης επιθυμεί.

Τα κριτήρια αυτά είναι τα εξής :

- Καθηγητής
- Φοιτητής
- Τίτλος Πτυχιακής
- Περιγραφή

Εισάγοντας, για παράδειγμα, το όνομα "Νίκος" στο textbox "φοιτητής" θα εμφανισθεί στη λίστα με όλες τις ολοκληρωμένες πτυχιακές, όπου το όνομα του φοιτητή είναι "Νίκος".

Έπειτα, θα υπήρχε η δυνατότητα να γίνει download της λίστας όλων των φοιτητών που έχουν ολοκληρωμένες εργασίες με το όνομα "Νίκος".

Αντίστοιχα, θα ήταν δυνατόν να γίνει αναζήτηση με τα κριτήρια που αναφέρθηκαν παραπάνω.

7.6 Οθόνη - Διαχείριση πτυχιακών εργασιών

Στοιχεία Σελίδας	
Όνομα σελίδας :	graduation_001.aspx
Τύπος σελίδας :	Content Page
User Control :	Grad.ascx

Πτυχιακές εργασίες							
		A/A	Τίτλος Πτυχιακής	Περιγραφή Πτυχιακής	Καθηγητής	Αρχείο	
1	<input type="checkbox"/>	2067	A	A	Νίκος Κάλαντας		<input type="checkbox"/> <input type="checkbox"/>
2	<input type="checkbox"/>	3067	TEST 1	TEST 1	Νίκος Κάλαντας		<input type="checkbox"/> <input type="checkbox"/>

Page size: 10 2 items in 1 pages

Νέα Καταχώρηση

Αιτήματα φοιτητών για πτυχιακές εργασίες

A/A	Τίτλος	Περιγραφή	Φοιτητής
No records to display.			

Page size: 10 0 items in 1 pages

Απορριφθέντα αιτήματα φοιτητών για πτυχιακές εργασίες

A/A	Τίτλος	Περιγραφή	Φοιτητής
No records to display.			

Page size: 10 0 items in 1 pages

Ανατεθειμένες Πτυχιακές - Τρέχουσες

A/A	Φοιτητής	Τίτλος	Περιγραφή
No records to display.			

Page size: 10 0 items in 1 pages

Ανατεθειμένες Πτυχιακές - Ολοκληρωμένες

A/A	Φοιτητής	Τίτλος	Περιγραφή
3067	Αχιλλέας Δρακάτος	TEST 1	TEST 1

Page size: 10 1 items in 1 pages

Εικόνα 7.6.1 : Οθόνη Διαχείρισης Πτυχιακών Εργασιών

Η παραπάνω είναι η κεντρική σελίδα διαχείρισης πτυχιακών εργασιών.

Στην συγκεκριμένη σελίδα, πρόσβαση έχουν μόνο οι χρήστες που είναι καθηγητές. Ουσιαστικά σε αυτήν την σελίδα, γίνεται η όλη διαχείριση των πτυχιακών εργασιών όσον αφορά στους καθηγητές.

Σε αυτήν την σελίδα υπάρχουν πέντε λειτουργίες και είναι οι εξής :

- Πτυχιακές εργασίες
- Αιτήματα φοιτητών για πτυχιακές εργασίες
- Απορριφθέντα αιτήματα φοιτητών για πτυχιακές εργασίες
- Ανατεθειμένες Πτυχιακές - Τρέχουσες
- Ανατεθειμένες Πτυχιακές - Ολοκληρωμένες

Πτυχιακές εργασίες

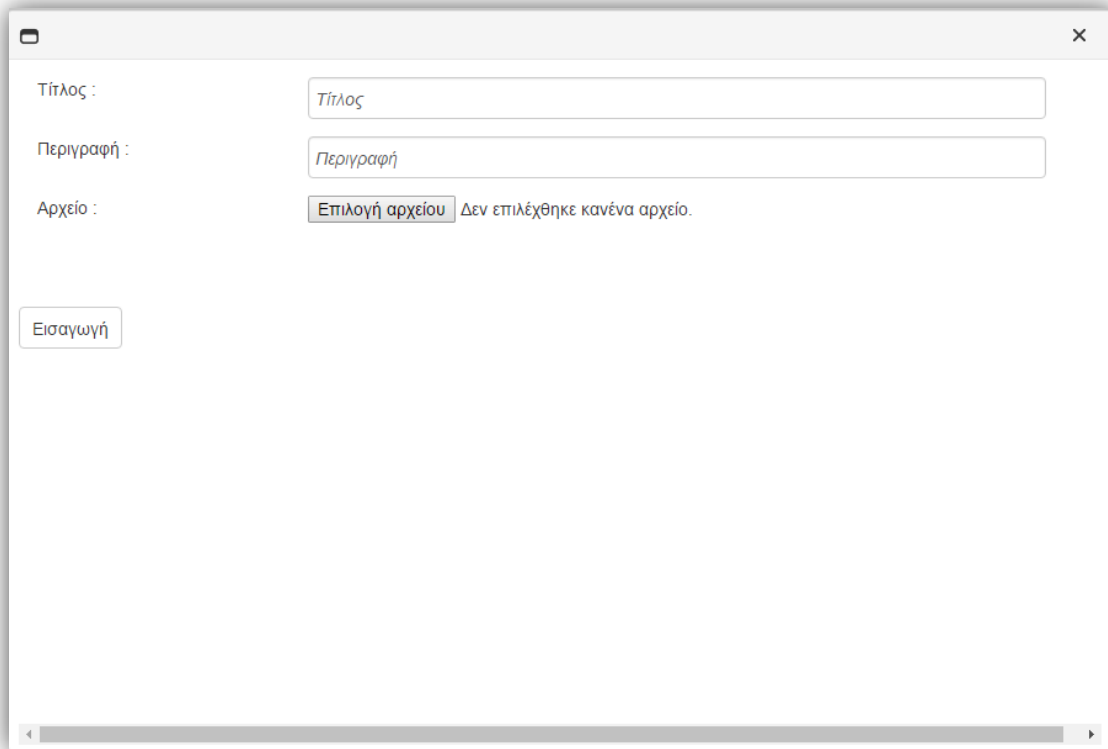
Πτυχιακές εργασίες							
		A/A	Τίτλος Πτυχιακής	Περιγραφή Πτυχιακής	Καθηγητής	Αρχείο	
1	<input type="checkbox"/>	2067	A	A	Νίκος Κάλαντας		<input type="checkbox"/> <input type="checkbox"/>
2	<input type="checkbox"/>	3067	TEST 1	TEST 1	Νίκος Κάλαντας		<input type="checkbox"/> <input type="checkbox"/>
3	<input type="checkbox"/>	4067	Web Trends	Δημιουργία Web Site	Νίκος Κάλαντας		<input type="checkbox"/> <input type="checkbox"/>

Page size: 10 3 items in 1 pages

Νέα Καταχώρηση

Εικόνα 7.6.2 : Λειτουργία-Πτυχιακές Εργασίες

Στην τρέχουσα λειτουργία υπάρχει μία λίστα με τις πτυχιακές εργασίες που έχει εισάγει στο σύστημα ο καθηγητής. Κάτω από την λίστα αυτή, υπάρχει το κουμπί **Νέα Καταχώριση**. Πατώντας το κουμπί **Νέα Καταχώριση** θα ανοίξει το user control, όπως φαίνεται παρακάτω.



Εικόνα 7.6.3 : User Control - Νέα καταχώριση

Σε αυτό το user control εισάγονται τα στοιχεία της πτυχιακής ώστε να γίνει μια νέα καταχώριση στο σύστημα.

Υπάρχουν τρία textbox στα οποία θα δοθούν τα στοιχεία της εκάστοτε πτυχιακής.

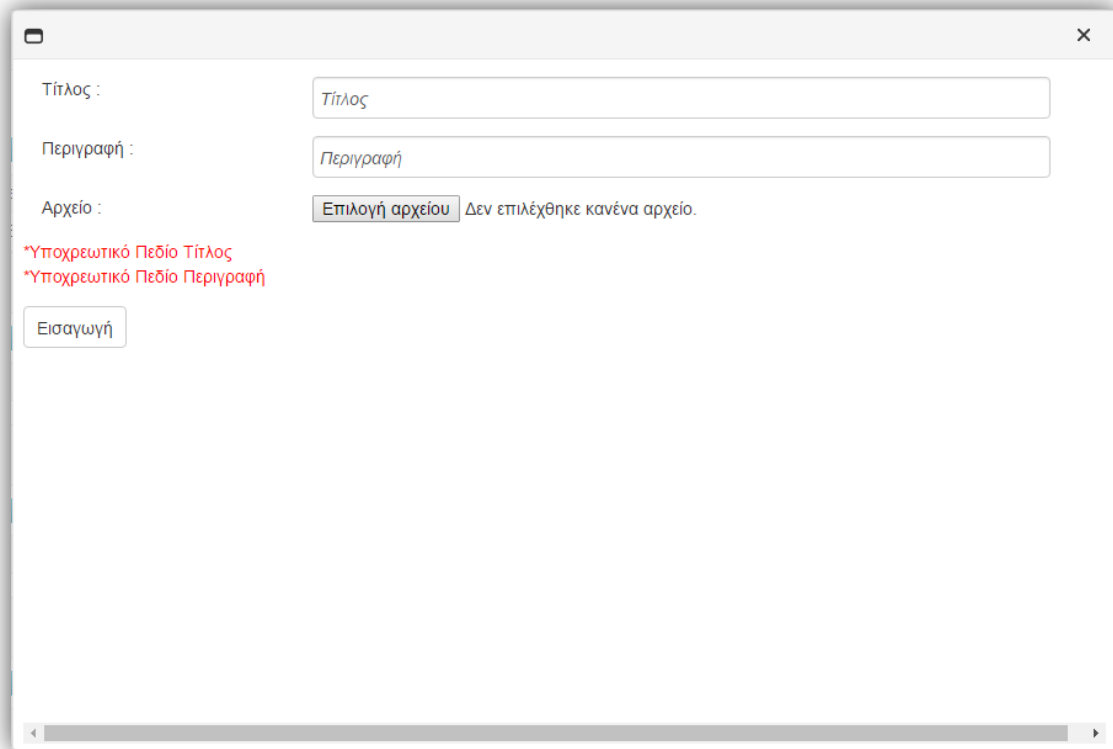
Πιο συγκεκριμένα:

- **Τίτλος** : Εισάγεται ο τίτλος της νέας πτυχιακής εργασίας.
- **Περιγραφή** : Δίνεται μια σύντομη περιγραφή σχετικά με την πτυχιακή εργασία.
- **Αρχείο** : Εισάγεται ένα αρχείο σχετικά με την πτυχιακή εργασία.

Τέλος, πατώντας το κουμπί "Εισαγωγή" αν καταχωρηθούν επιτυχώς, πραγματοποιείται η εισαγωγή και κλείνει το user control.

Σε διαφορετική περίπτωση, αν για παράδειγμα πατηθεί το κουμπί "Εισαγωγή" και δεν έχει δοθεί "Τίτλος" ή "Περιγραφή" που τα πεδία αυτά είναι υποχρεωτικά, θα εμφανιστεί στο user control το μήνυμα "**Υποχρεωτικό Πεδίο Τίτλος**" ή αντίστοιχα "**Υποχρεωτικό Πεδίο**

Περιγραφή" με έντονα κόκκινα γράμματα και θα αποτραπεί η εισαγωγή μιας νέας πτυχιακής εργασίας όπως φαίνεται στην παρακάτω εικόνα.



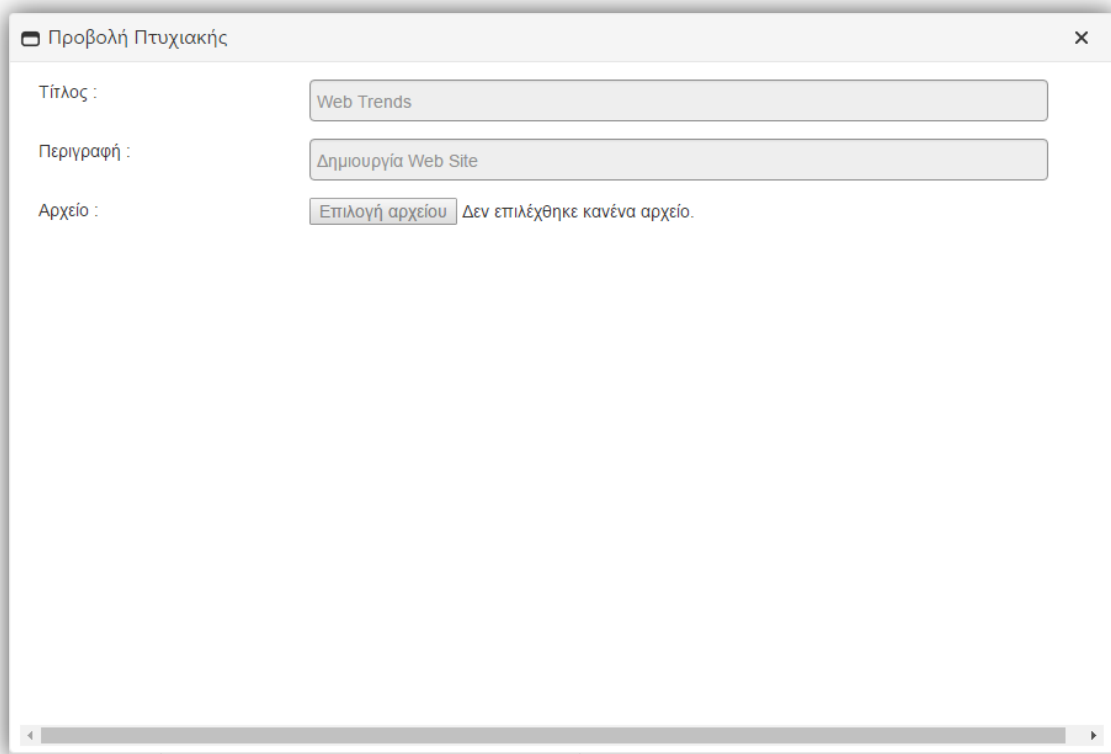
Εικόνα 7.6.4 : User Control - Μήνυμα Αποτυχίας

Αφού ο χρήστης καθηγητής εισάγει τις πτυχιακές που επιθυμεί, μπορεί μέσω των κουμπιών που υπάρχουν στην λίστα να εκτελέσει κάποιες λειτουργίες.

Στη λίστα υπάρχουν ακόμα τέσσερα κουμπιά δεξιά και αριστερά.

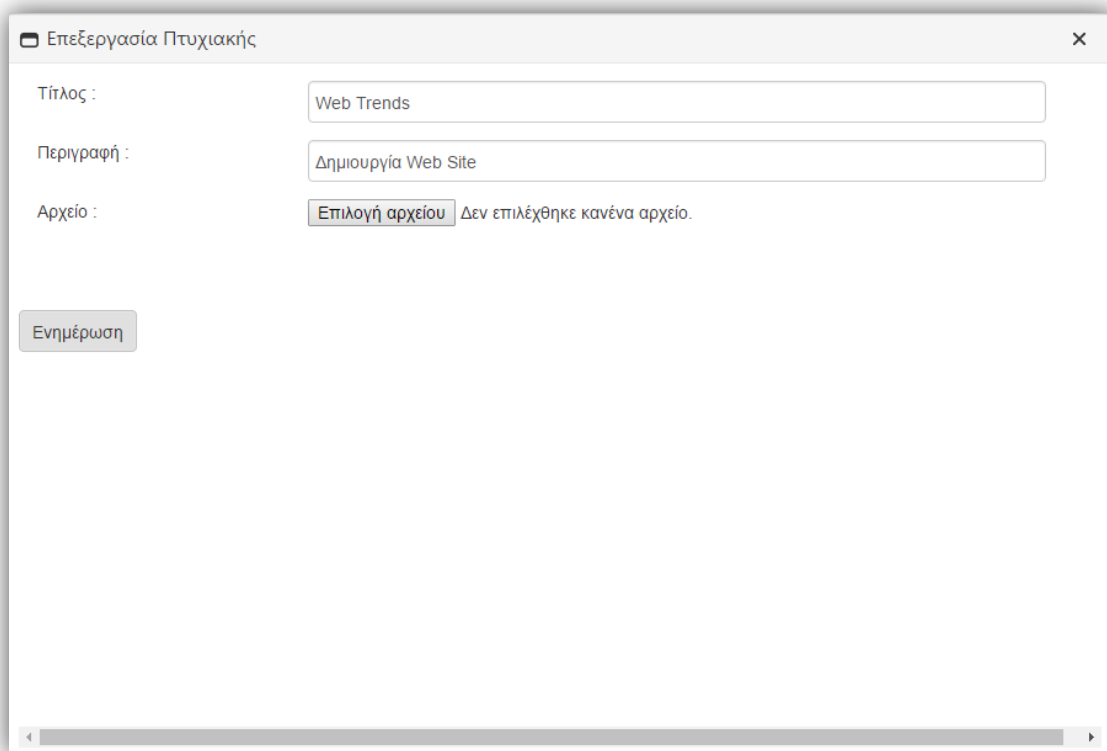
Υπάρχει το κουμπί "**Προβολή**", "**Επεξεργασία**", "**Διαγραφή**", "**Download**".

Πατώντας το κουμπί "**Προβολή**" προβάλλονται στο user control στοιχεία σχετικά με την πτυχιακή όπως αποτυπώνεται παρακάτω.



Εικόνα 7.6.5 : User Control - Προβολή Στοιχείων

Πατώντας το κουμπί "**Επεξεργασία**" είναι δυνατόν να επεξεργαστεί ο χρήστης τα στοιχεία της πτυχιακής εργασίας, όπως φαίνεται παρακάτω. Με τον τρόπο αυτό, μπορεί να αλλάξει τα πεδία που επιθυμεί και πατώντας το κουμπί "**Ενημέρωση**" και γίνεται επεξεργασία των στοιχείων αυτών.



Εικόνα 7.6.6 : User Control - Επεξεργασίας Στοιχείων

Πατώντας το κουμπί "Διαγραφή" διαγράφεται η πτυχιακή εργασία.

Τέλος, πατώντας το κουμπί "Download" ο καθηγητής μπορεί να κατεβάσει το έγγραφο που υπάρχει ήδη στο σύστημα.

Αιτήματα φοιτητών για πτυχιακές εργασίες

Αιτήματα φοιτητών για πτυχιακές εργασίες					
A/A	Τίτλος	Περιγραφή	Φοιτητής		
3067	TEST 1	TEST 1	Αχιλλέας Δρακάτος	✓	✗
4067	Web Trends	Δημιουργία Web Site	Κώστας Δρόσος	✓	✗

Page size: 10 | 2 items in 1 pages

Εικόνα 7.6.7 : Λειτουργία Διαχείρισης Αιτημάτων Πτυχιακών Εργασιών

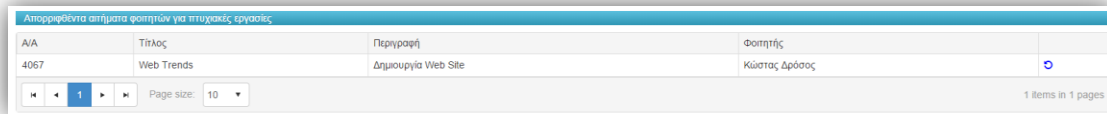
Στην λειτουργία αυτή υπάρχει μια λίστα όπου ο καθηγητής μπορεί να δει τα αιτήματα των φοιτητών σχετικά με τις πτυχιακές εργασίες που έχει εισάγει.

Δεξιά υπάρχουν δύο επιλογές.

Πατώντας την επιλογή "Ανάθεση Πτυχιακής" η πτυχιακή μεταφέρεται στην λίστα "Ανατεθειμένες Πτυχιακές-Τρέχουσες", που θα αναλυθεί παρακάτω, διαφορετικά, με την

επιλογή "Απόρριψη" η πτυχιακή μεταφέρεται στην λίστα "Απορριφθέντα αιτήματα φοιτητών για πτυχιακές εργασίες".

Απορριφθέντα αιτήματα φοιτητών για πτυχιακές εργασίες



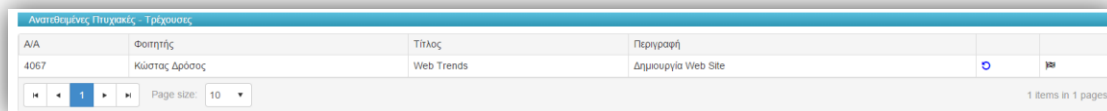
A/A	Τίτλος	Περιγραφή	Φοιτητής	
4067	Web Trends	Δημιουργία Web Site	Κώστας Δρόσος	<input type="radio"/>

Εικόνα 7.6.8 : Λειτουργία Απορριφθέντων Αιτημάτων Πτυχιακών Εργασιών

Σε αυτήν την λειτουργία υπάρχει μια λίστα σχετικά με τις πτυχιακές εργασίες που απορρίφθηκαν από τον καθηγητή.

Δεξιά υπάρχει η επιλογή "Αναίρεση". Πατώντας ο χρήστης την επιλογή της αναίρεσης για οποιονδήποτε λόγο, για παράδειγμα επειδή επέλεξε λάθος πτυχιακή, αναιρεί την προηγούμενη επιλογή του. Έτσι, η πτυχιακή εργασία εμφανίζεται και πάλι στην λίστα της λειτουργίας "Αιτήματα φοιτητών για πτυχιακές εργασίες".

Ανατεθειμένες Πτυχιακές-Τρέχουσες



A/A	Φοιτητής	Τίτλος	Περιγραφή		
4067	Κώστας Δρόσος	Web Trends	Δημιουργία Web Site	<input type="radio"/>	<input type="radio"/>

Εικόνα 7.6.9 : Λειτουργία Ανατεθειμένων Πτυχιακών Εργασιών

Σε αυτήν την λειτουργία υπάρχει μια λίστα όπου εμφανίζονται όλες οι πτυχιακές εργασίες οι οποίες έχουν ανατεθεί, όπως είδαμε προηγουμένως, από την λειτουργία "Αιτήματα φοιτητών για πτυχιακές εργασίες".

Δεξιά της λίστας υπάρχουν δυο επιλογές, η επιλογή "Αναίρεση" και η επιλογή "Σήμανση - Ολοκληρωμένη Πτυχιακή".

Ο χρήστης επιλέγοντας την "Αναίρεση" μεταφέρεται η πτυχιακή στην λίστα της λειτουργίας "Αιτήματα φοιτητών για πτυχιακές εργασίες" και δεν εμφανίζεται πλέον στην λίστα της συγκεκριμένης λειτουργίας.

Πατώντας, τέλος, την επιλογή "Σήμανση- Ολοκληρωμένη Πτυχιακή" η πτυχιακή μεταφέρεται στην λίστα της λειτουργίας "Ανατεθειμένες Πτυχιακές - Ολοκληρωμένες", όπως φαίνεται παρακάτω.

Ανατεθειμένες Πτυχιακές - Ολοκληρωμένες

A/A	Φοιτητής	Κώστας Δρόσος	Τίτλος	Web Trends	Περιγραφή	Δημιουργία Web Site
4067						

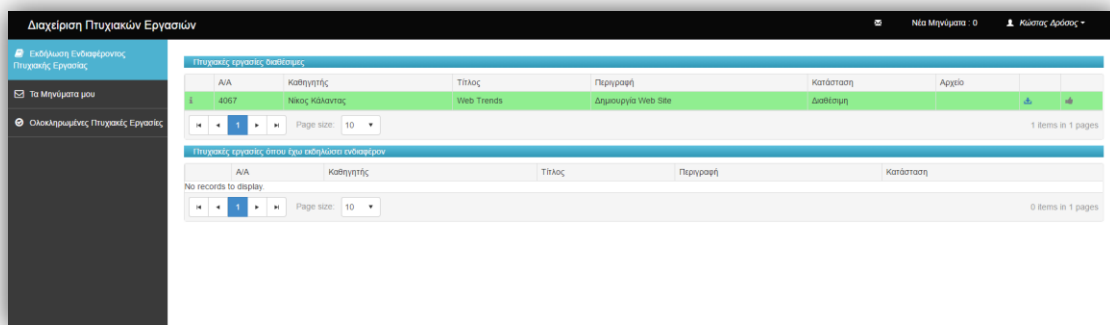
Εικόνα 7.6.10 : Λειτουργία Ολοκληρωμένων Πτυχιακών Εργασιών

Η λειτουργία αυτή είναι και η τελευταία λειτουργία της συγκεκριμένης οθόνης που αναλύεται.

Σε αυτή τη λειτουργία υπάρχει, όπως και σε όλες τις υπόλοιπες λειτουργίες της οθόνης αυτής, μία λίστα. Η λίστα αυτή περιέχει όλες τις πτυχιακές εργασίες που είναι ολοκληρωμένες. Δεξιά της λίστας υπάρχει η επιλογή "Αναίρεση". Με την επιλογή αυτή, η εργασία μεταφέρεται στην λίστα της λειτουργίας "Ανατεθειμένες Πτυχιακές-Τρέχουσες".

7.7 Οθόνη - Εκδήλωση Ενδιαφέροντος Πτυχιακής εργασίας

Στοιχεία Σελίδας	
Όνομα σελίδας :	graduation.aspx
Τύπος σελίδας :	Content Page
User Control :	-



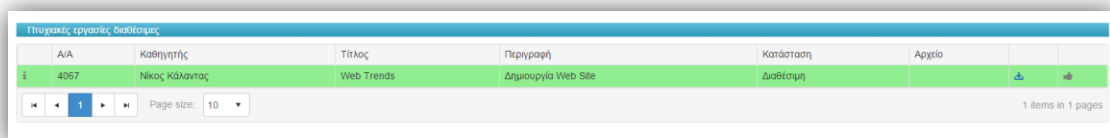
Εικόνα 7.7.1 : Οθόνη Εκδήλωσης Ενδιαφέροντος Πτυχιακής Εργασίας

Σε αυτήν την σελίδα πρόσβαση έχουν οι χρήστες φοιτητές που δεν τους έχει γίνει ανάθεση πτυχιακής εργασίας. Η σελίδα αυτή, είναι η κύρια οθόνη για τους χρήστες φοιτητές, που συναντούν κατά την είσοδό τους.

Σε αυτήν την σελίδα υπάρχουν δύο λειτουργίες και είναι οι εξής :

- Πτυχιακές εργασίες διαθέσιμες.
- Πτυχιακές εργασίες που έχω εκδηλώσει ενδιαφέρον.

Πτυχιακές Εργασίες Διαθέσιμες



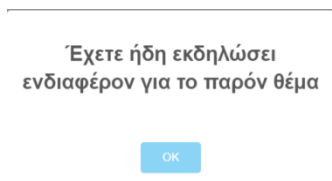
Εικόνα 7.7.2 : Λειτουργία Πτυχιακές Εργασίες Διαθέσιμες

Σε αυτήν την λειτουργία οι χρήστες φοιτητές έχουν πρόσβαση σε μία λίστα όπου υπάρχουν οι διαθέσιμες πτυχιακές εργασίες, που έχουν αναρτήσει οι καθηγητές.

Στην λίστα, ο χρήστης μπορεί να δει στοιχεία που αφορούν στην πτυχιακή, όπως το ονοματεπώνυμο του εκάστοτε καθηγητή που ανήρτησε την πτυχιακή, τον τίτλο, μια σύντομη περιγραφή, καθώς επίσης υπάρχει η δυνατότητα να κατεβάσει ένα αρχείο που έχει ανεβάσει ο καθηγητής σχετικά με αυτήν.

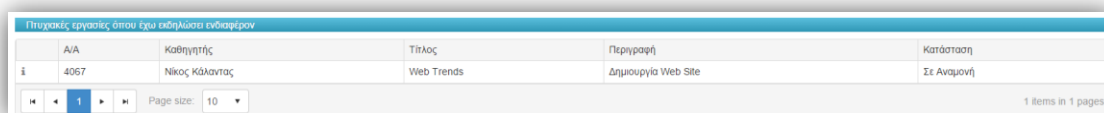
Κύρια όμως λειτουργία είναι η εκδήλωση ενδιαφέροντος για την πτυχιακή. Αυτό γίνεται μέσω μιας επιλογής δεξιά από κάθε διαθέσιμη πτυχιακή. Επιλέγοντας ο χρήστης την επιλογή αυτή, δηλώνει την πτυχιακή.

Σε περίπτωση που έχει ήδη εκδηλώσει ενδιαφέρον για μια πτυχιακή και κάνει τη συγκεκριμένη επιλογή, το σύστημα θα τον αποτρέψει και θα εμφανιστεί το σχετικό μήνυμα "Έχετε ήδη εκδηλώσει ενδιαφέρον για το παρόν θέμα" σε ένα παράθυρο διαλόγου.



Εικόνα 7.7.3 : Μήνυμα Αποτυχίας Δήλωσης Πτυχιακής Εργασίας

Πτυχιακές εργασίες που έχω εκδηλώσει ενδιαφέρον



A/A	Καθηγητής	Τίτλος	Περιγραφή	Κατάσταση
4067	Νίκος Κόλλαντας	Web Trends	Δημιουργία Web Site	Σε Αναμονή

Εικόνα 7.7.4 : Λειτουργία Εκδήλωσης Ενδιαφέροντος

Μέσω της λειτουργία αυτής, ο χρήστης φοιτητής έχει πρόσβαση σε μία λίστα σχετικά με τις πτυχιακές εργασίες που έχει εκδηλώσει ενδιαφέρον. Μπορεί να δει τα στοιχεία των πτυχιακών εργασιών που έχει εκδηλώσει ενδιαφέρον. Ακόμα, μπορεί να ελέγξει και την κατάσταση της πτυχιακής εργασίας.

Υπάρχουν δύο καταστάσεις :

- **Σε αναμονή** : Σε αναμονή βρίσκονται οι πτυχιακές εργασίες που ο καθηγητής δεν τις έχει εγκρίνει, αλλά ούτε τις έχει απορρίψει .
- **Απορρίφτηκε** : Σε αυτήν την κατάσταση βρίσκονται οι πτυχιακές που ο καθηγητής έχει απορρίψει.

graduation.aspx (markup κώδικας)

Στο σημείο αυτό, θα αναλυθεί η λειτουργία των λιστών που συναντώνται σχεδόν σε όλες τις οθόνες της εφαρμογής καθώς είναι ένας εύκολος τρόπος να παρουσιαστούν πολλά δεδομένα στον χρήστη της εφαρμογής.

Όλες οι λίστες που συναντώνται σε πολλές από τις οθόνες μας, δουλεύουν κατά τον ίδιο τρόπο .

Θα αναλυθεί, λοιπόν, η λειτουργία των λιστών που βρίσκονται στην λειτουργία **Πτυχιακές εργασίες διαθέσιμες**.

Είναι απαραίτητοι δύο τύποι control, ένα **Grid** και ένα **data source control**, με το οποίο θα "γεμίσει" το **Grid** με δεδομένα από την βάση δεδομένων.

Προστίθεται, λοιπόν, όπως φαίνεται παρακάτω ένα **data source control** και πιο συγκεκριμένα ένα **SqlDataSource** με το οποίο θα επιλεγούν όλες τις εγγραφές από τον πίνακα **graduate_themes** που βρίσκεται στην βάση δεδομένων. Ο πίνακας **graduate_themes** περιέχει όλες τις πτυχιακές εργασίες που εισάγουν οι καθηγητές.

Επιλέγονται, λοιπόν, όλες τις πτυχιακές εργασίες οι οποίες δεν είναι διαγραμμένες και είναι διαθέσιμες. Το αν οι πτυχιακές δεν είναι διαγραμμένες αναπαριστάται από το πεδίο **IsDeleted=0**. Αντίστοιχα, οι πτυχιακές εργασίες που είναι διαθέσιμες αναπαρίστανται από το πεδίο **IsAvailable=0**.

```
<asp:SqlDataSource ID="dsGrad" runat="server" SelectCommand="Select * from
graduate_themes where IsDeleted = 0 and IsAvailable = 0 "
CancelSelectOnNullParameter="false"></asp:SqlDataSource>
```

Προστίθεται και το Grid που θα εμφανίζει τις εγγραφές αυτές, όπως φαίνεται παρακάτω.

```
<telerik:RadGrid ID="rgGrad" runat="server" DataSourceID="dsGrad"
AllowPaging="true" AutoGenerateColumns="false"
OnItemCommand="rgGrad_ItemCommand" OnItemDataBound="rgGrad_ItemDataBound"
AllowAutomaticDeletes="true">
<PagerStyle Mode="NumericPages" AlwaysVisible="true" />
<MasterTableView DataKeyNames="ID,IsAvailable" DataSourceID="dsGrad">
<Columns>
<telerik:GridTemplateColumn>
<ItemTemplate>
<asp:LinkButton runat="server" ID="ibtnView" Text="<i class=' fa fa-info'></i>"
CommandName="ViewGrad" ForeColor="Gray" ToolTip="Προβολή"></asp:LinkButton>
</ItemTemplate>
```

```

</telerik:GridTemplateColumn>
<telerik:GridBoundColumn                                DataField="ID"
HeaderText="Α/Α"></telerik:GridBoundColumn>
<telerik:GridBoundColumn                                DataField="Title"
HeaderText="Τίτλος"></telerik:GridBoundColumn>
<telerik:GridBoundColumn                                DataField="Descr"
HeaderText="Περιγραφή"></telerik:GridBoundColumn>
<telerik:GridBoundColumn                                DataField="IsAvailable"   HeaderText="Κατάσταση"
UniqueName="situation"></telerik:GridBoundColumn>
<telerik:GridBoundColumn                                DataField="Name"
HeaderText="Αρχείο"></telerik:GridBoundColumn>
<telerik:GridTemplateColumn                              UniqueName="EditCommandColumn"
AllowFiltering="false" FilterControlAltText="Filter Edit column">
<ItemTemplate>
<asp:LinkButton                                        Tooltip="Download"                                ID="ibtnDownload"
OnClick="ibtnDownload_Click" runat="server" Text="<i class='fa fa-download'>
</i>"></asp:LinkButton>
</ItemTemplate>
</telerik:GridTemplateColumn>
<telerik:GridTemplateColumn>
<ItemTemplate>
<asp:LinkButton runat="server" ID="ibtnIntrested" CommandName="mark" Text="<i
class='fa fa-thumbs-up'> </i>" ForeColor="Gray"></asp:LinkButton>
</ItemTemplate>
</telerik:GridTemplateColumn>
</Columns>
<PagerStyle AlwaysVisible="true" />
</MasterTableView>
</telerik:RadGrid>

```

Για να γεμίσει το Grid αυτό με δεδομένα συνδέεται ουσιαστικά το **SqlDataSource** που δημιουργήθηκε παραπάνω με το Grid.

Αυτό γίνεται από τις ιδιότητες του Grid, ως εξής **DatasourceID="dsGrad"**, δίνοντας δηλαδή το όνομα του Datasource που έχει δημιουργηθεί στην ιδιότητα **DatasourceID**.

Δημιουργώντας, λοιπόν, τις στήλες μέσα στο Grid δηλώνονται ποια πεδία θα εμφανιστούν σε αυτές από την βάση δεδομένων με τον εξής τρόπο :

```
<telerik:GridBoundColumn DataField="Title"
HeaderText="Τίτλος"></telerik:GridBoundColumn>
```

Αυτό γίνεται για όλα τα πεδία που επιθυμεί κανείς να εμφανίζονται σε στήλες από την βάση δεδομένων.

Στις ιδιότητες του Grid είναι απαραίτητο να δημιουργηθούν δύο γεγονότα (event) που είναι σημαντικά για την λειτουργία του Grid.

Αυτά είναι τα εξής :

- **OnItemCommand** : Event κατά το οποίο δίνεται η επιλογή σε κάποια κουμπιά μέσα στο Grid να εκτελούν κάποιες λειτουργίες.
- **OnItemDataBound** : Event κατά το οποίο γίνεται προσπέλαση όλων των στοιχείων στο Grid.

Στο σημείο αυτό, να τονισθεί ότι κατά την δημιουργία αυτών των event, δημιουργούνται στην σελίδα με τον κώδικα, κλάσεις. Θα ολοκληρωθεί, λοιπόν, η λειτουργία αυτών των event από την σελίδα **graduation.aspx.vb**, μέσα στις κλάσεις που δημιουργήθηκαν όπως θα παρουσιαστεί παρακάτω.

graduation.aspx.vb (κώδικας visual basic)

Αφού έχουν δημιουργηθεί τα event **OnItemCommand** και **OnItemDataBound** μέσα στο Grid, θα εξηγηθεί η λειτουργία τους μέσα από την σελίδα **graduation.aspx.vb**. Μέσα στις κλάσεις που έχουν δημιουργηθεί θα γραφτεί κώδικας στην γλώσσα προγραμματισμού visual basic.

Παρακάτω φαίνεται το event **OnItemDataBound** κατά το οποίο γίνεται η προσπέλαση των στοιχείων μέσα στο Grid. Οι λειτουργίες που είναι επιθυμητό να εκτελούνται κατά την προσπέλαση αυτών των στοιχείων είναι οι εξής :

Αλλαγή κατάστασης σε "Διαθέσιμη" ή "Μη διαθέσιμη".

Αλλαγή χρώματος στις γραμμές του Grid, ανάλογα με την κατάσταση της πτυχιακής. Αν η κατάσταση της πτυχιακής είναι "Διαθέσιμη", τότε η γραμμή γίνεται πράσινη, αν η κατάσταση είναι "Μη Διαθέσιμη", τότε η γραμμή μέσα στο Grid γίνεται κόκκινη.

```
Protected Sub rgGrad_ItemDataBound(sender As Object, e As
Telerik.Web.UI.GridItemEventArgs) ' Event OnItemDataBound

If TypeOf e.Item Is GridDataItem Then

Dim item As GridDataItem = DirectCast(e.Item, GridDataItem)

Dim Status As String = item.GetDataKeyValue("IsAvailable")

Dim ID As String = item.GetDataKeyValue("ID")

Dim ibtntView As LinkButton = TryCast(item.FindControl("ibtnView"), LinkButton)

If Not ibtntView Is Nothing Then

ibtntView.CommandArgument = ID

End If

If item("situation").Text = False Then ' Με αυτόν τον τρόπο αλλάζω το όνομα
μέσα στην κατάσταση / 'Όταν θέλω να πάρω το κελί λέω item("μεσα εδώ βάζω το
UniqueName απο το GridBoundColumn")

item("situation").Text = "Διαθέσιμη"

Else

item("situation").Text = "Μη Διαθέσιμη"

End If

If Status = False Then ' Είναι Διαθέσιμη

'Αν είναι διαθέσιμο το χρώμα δεν το αλλάζω / θα μπορούσα να το αλλάξω σε
πράσινο

rgGrad.MasterTableView.BackColor = Drawing.Color.LightGreen

Else

rgGrad.MasterTableView.Columns.Item("6").Display = False ' Θα κρύψει το 6
δηλαδή το κουμπί όπου δηλώνω πτυχιακή

rgGrad.MasterTableView.BackColor = Drawing.Color.LightCoral

End If

End If

End Sub
```

Παρακάτω φαίνεται το event OnItemCommand.

Όταν ο χρήστης κάνει την επιλογή "εκδήλωση ενδιαφέροντος πτυχιακής εργασίας" οι λειτουργίες που εκτελούνται είναι δύο :

- Γίνεται έλεγχος για το αν η πτυχιακή εργασία έχει ήδη δηλωθεί από τον χρήστη. Εάν αυτό έχει ήδη γίνει, εμφανίζεται μήνυμα αποτυχίας και αποτρέπεται η δήλωση αυτής.
- Αν κατά τον έλεγχο ο χρήστης φοιτητής δεν έχει ήδη εκδηλώσει ενδιαφέρον για το συγκεκριμένο θέμα, ολοκληρώνεται η διαδικασία εκδήλωσης ενδιαφέροντος.

```
Protected Sub rgGrad_ItemCommand(sender As Object, e As Telerik.Web.UI.GridCommandEventArgs)
    If e.CommandName = "mark" Then ' Όταν ο χρήστης πατήσει το κουμπί με CommandName="mark" εκτελούνται τα παρακάτω.
        .....'Ελεγχος'.....
        Dim item As GridDataItem = DirectCast(e.Item, GridDataItem)
        Dim ID As String = item.GetDataKeyValue("ID")
        Dim CnIDStudent As Integer
        Dim a As Integer
        Dim con1 As New SqlConnection(ConnectionString) ' Άνοιγμα σύνδεσης
        Dim con2 As New SqlConnection(ConnectionString) ' Άνοιγμα σύνδεσης
        Dim cmd1 As SqlCommand
        Dim cmd2 As SqlCommand
        cmd1 = New SqlCommand("Select ID from graduate_requests where ID=@ID and CnIDStudent=@CnIDStudent ", con1) ' Select
        cmd1.Parameters.Add("@ID", SqlDbType.Int)
        cmd1.Parameters.Add("@CnIDStudent", SqlDbType.Int)
        cmd1.Parameters("@ID").Value = ID
        cmd1.Parameters("@CnIDStudent").Value = Me.Session("CnID")
        cmd2 = New SqlCommand("Select CnIDStudent from graduate_requests where ID=@ID", con2) ' Select
```

```

cmd2.Parameters.Add("@ID", SqlDbType.Int)

cmd2.Parameters("@ID").Value = ID

con1.Open() ' Άνοιγμα σύνδεσης

a = cmd1.ExecuteScalar

con1.Close() ' Κλείσιμο σύνδεσης

con2.Open()

CnIDStudent = cmd2.ExecuteScalar

con2.Close()

'Αν ο Φοιτητής έχει ήδη εκδηλώσει ενδιαφέρον για την συγκεκριμένη πτυχιακή τότε
θα εμφανιστεί μήνυμα αποτυχίας.

If a = ID And CnIDStudent = Me.Session("CnID") Then

ScriptManager.RegisterStartupScript(Me, Page.GetType, "Script",
"AlreadyTitle();", True)

Exit Sub

End If

.....Τέλος Ελέγχου.....

'Αφού έγινε ο έλεγχος ότι ο Χρήστης φοιτητής δεν έχει δηλώσει την συγκεκριμένη
πτυχιακή τότε θα μπορέσει να δηλώσει την πτυχιακή

Dim con As New SqlConnection(ConnectionString) 'Δημιουργία Connection

Dim cmd As SqlCommand

cmd = New SqlCommand("INSERT INTO graduate_requests (ID, CnID, , Title ,
Descr,IsAvailable,IsCompleted,CnIDStudent,IsRejected,IsProven) SELECT ID, CnID,
Title,Descr,IsAvailable,IsCompleted,@CnIDStudent,0,0 FROM graduate_themes WHERE
ID=@ID", con)

cmd.Parameters.Add("@ID", SqlDbType.Int)

cmd.Parameters("@ID").Value = ID

cmd.Parameters.Add("@CnIDStudent", SqlDbType.Int)

cmd.Parameters("@CnIDStudent").Value = Me.Session("CnID")

con.Open() 'Άνοιγμα σύνδεσης

cmd.ExecuteScalar() ' Εισαγωγή δεδομένων στην βάση

con.Close() ' Κλείσιμο σύνδεσης

```



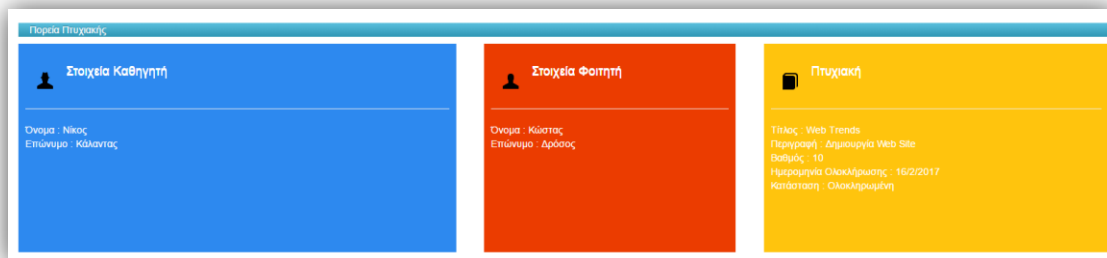
```
rgGrad_001.Rebind()
```

```
End If
```

```
End Sub
```

7.8 Οθόνη - Πορεία Πτυχιακής

Στοιχεία Σελίδας	
Όνομα σελίδας :	myprofile.aspx
Τύπος σελίδας :	Content Page
User Control :	-



Εικόνα 7.8.1 : Οθόνη Πορεία Πτυχιακής

Στην σελίδα αυτήν πρόσβαση έχουν οι χρήστες φοιτητές που τους έχει ανατεθεί πτυχιακή εργασία. Αν δεν τους είχε ανατεθεί εργασία στην σελίδα αυτή δεν θα είχαν πρόσβαση οι χρήστες. Σε αυτήν την περίπτωση οι χρήστες φοιτητές θα είχαν πρόσβαση στην σελίδα **Εκδήλωση ενδιαφέροντος πτυχιακής εργασίας** της οποίας η λειτουργία έχει αναλυθεί παραπάνω.

Η σελίδα αποτελείται από τρία κελιά όπου εμφανίζονται στοιχεία σχετικά με την πτυχιακή.

Πιο συγκεκριμένα, εμφανίζονται τα εξής στοιχεία:

- Στοιχεία Καθηγητή: Το όνομα και το επώνυμο του καθηγητή.
- Στοιχεία Φοιτητή: Το όνομα και το επώνυμο του φοιτητή.

- **Πτυχιακή:** Ο τίτλος της πτυχιακής , σύντομη περιγραφή αυτής , ο βαθμός της πτυχιακής -αν υπάρχει. Σε άλλη περίπτωση αντί για τον βαθμό, θα υπήρχε η ένδειξη "-". Αν η πτυχιακή ήταν ολοκληρωμένη θα εμφανιζόταν η ημερομηνία ολοκλήρωσης, αλλιώς η ένδειξη "-" και η κατάσταση της πτυχιακής. Αν ήταν ολοκληρωμένη θα εμφανιζόταν η ένδειξη "Ολοκληρωμένη" αλλιώς η ένδειξη "Μη Ολοκληρωμένη".

ΣΥΜΠΕΡΑΣΜΑΤΑ/ ΑΠΟΤΕΛΕΣΜΑΤΑ/ ΕΠΙΛΟΓΟΣ

Στην παρούσα πτυχιακή εργασία, το κύριο αντικείμενο είναι η δημιουργία μιας διαδικτυακής εφαρμογής σχετικά με την διαχείριση πτυχιακών εργασιών.

Στις ενότητες που καλύφθηκαν στην εργασία, έγινε προσπάθεια να καλυφθούν τόσο το θεωρητικό κομμάτι που αφορά στις υπηρεσίες του υπολογιστικού νέφους (cloud computing) -μιας και η εφαρμογή είναι διαδικτυακή και η πρόσβαση σε αυτήν γίνεται αποκλειστικά από το διαδίκτυο- όσο και το κομμάτι σχετικά με την υλοποίησή της και τις τεχνολογίες που χρησιμοποιήθηκαν για την ολοκλήρωση αυτής. Επιπρόσθετα, έγινε αναφορά σε συστήματα διαχείρισης έργων μιας και η εφαρμογή μας είναι ουσιαστικά ένα σύστημα διαχείρισης έργου, που αφορά στην διαχείριση πτυχιακών εργασιών.

Κατά τη γνώμη μου, η συγκεκριμένη εφαρμογή θα αποτελούσε ένα σημαντικό εργαλείο τόσο στα χέρια των φοιτητών όσο και των καθηγητών, κατά την επιλογή και αποπεράτωση των πτυχιακών εργασιών.

Σε ένα δεύτερο επίπεδο, θα αποτελούσε εργαλείο για την προώθηση της έρευνας και έναν διαδικτυακό τόπο που θα προωθούσε τη συνεργασία πολλών Τεχνολογικών Εκπαιδευτικών Ιδρυμάτων και Πανεπιστημιακών Ιδρυμάτων, ανά την Ελλάδα.

Κατά τη διαδικασία δημιουργίας μιας τέτοιας διαδικτυακής εφαρμογής, έγινε αντιληπτό πως δεν πρόκειται για κάτι τόσο εύκολο όσο φαίνεται. Για την ανάπτυξη μιας τέτοιας διαδικτυακής εφαρμογής, χρειάζονται γνώσεις προγραμματισμού, καθώς και γνώσεις σχετικές με βάσεις δεδομένων, ώστε να γίνει σωστά η διαχείριση των δεδομένων.

Η υλοποίηση της εφαρμογής αυτής έστω και στο πειραματικό στάδιο που προτείνεται μέσω της πτυχιακής μου εργασίας, σίγουρα θα αποτελούσε πέρα από ένα σημαντικό εργαλείο και μια καινοτομία στον τομέα της Τριτοβάθμιας Εκπαίδευσης. Οι επεκτάσεις που μπορεί να λάβει είναι απεριόριστες, απαιτούν όμως τη συνεργασία διαφόρων τομέων , μια εκτενέστερη μελέτη των αναγκών και απαιτήσεων , τόσο των φοιτητών όσο και των καθηγητών και φυσικά την ανάλογη υλικοτεχνική υποστήριξη από τους αρμόδιους φορείς.

ΒΙΒΛΙΟΓΡΑΦΙΑ / ΔΙΑΔΙΚΤΥΟ

- i. https://en.wikipedia.org/wiki/Cloud_computing#History
- ii. <http://www.inforisktoday.com/5-essential-characteristics-cloud-computing-a-4189>
- iii. <http://www.levelcloud.net/why-levelcloud/cloud-education-center/advantages-and-disadvantages-of-cloud-computing/>
- iv. <http://blog.capterra.com/free-open-source-project-management-software/>
- v. https://el.wikipedia.org/wiki/Visual_Basic
- vi. <https://el.wikipedia.org/wiki/JavaScript>
- vii. <https://el.wikipedia.org/wiki/HTML>
- viii. <https://el.wikipedia.org/wiki/CSS>

Πνευματικά δικαιώματα

Copyright © ΤΕΙ Δυτικής Ελλάδας. Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1988 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα εργασία αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον.

Αχιλλέας Δρακάτος, 2017