

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΣΧΕΔΙΟ ΠΑΡΟΥΣΙΑΣΗΣ
ΑΤΟΜΙΚΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Μάιος 2011

Ατομική Διπλωματική Εργασία

**ΑΝΑΠΤΥΞΗ ΣΥΣΤΗΜΑΤΟΣ ΔΙΑΧΕΙΡΗΣΗΣ ΕΡΕΥΝΗΤΙΚΩΝ
ΠΡΟΓΡΑΜΜΑΤΩΝ**

Ρουμπίνα Κάττου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2011

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ανάπτυξη Συστήματος Διαχείρισης Ερευνητικών Προγραμμάτων

Ρουμπίνα Κάττου

Επιβλέπων Καθηγητής

Σκέυος Ευριπίδου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης μεταπτυχιακού Πληροφορικής στο Τμήμα Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2011

Ευχαριστίες

Ευχαριστώ θερμά τον καθηγητή μου Παρασκευά Ευριπίδου που με ενθουσιασμό και ζήλο με καθοδηγούσε και συμβούλευε. Ακόμη τον ευχαριστώ για τις πολύτιμες διορθώσεις και εισηγήσεις του. Χρήσιμη ήταν και η βοήθεια που μου έδωσαν οι υπάλληλοι του γραφείου διαχείρισης ερευνητικών προγραμμάτων του τμήματος Πληροφορικής, με μεγάλη τους ευχαρίστηση μου έδιναν όλες τις πληροφορίες που χρειαζόμουν.

Τέλος, ευχαριστώ πάρα πολύ όλους τους φίλους και την οικογένεια μου για την υποστήριξη και την βοήθεια που μου παρείχαν κατά την εκπόνηση της διπλωματικής μου εργασίας.

Περίληψη

Σκοπός της διπλωματικής μου εργασίας είναι η ανάπτυξη συστήματος διαχείρισης προσωπικών δαπανών των ερευνητικών προγραμμάτων του τμήματος Πληροφορικής. Στο συγκεκριμένο σύστημα θα έχουν πρόσβαση εξουσιοδοτημένοι χρήστες, καθηγητές και φοιτητές. Κάθε χρήστης θα έχει καθορισμένες προσβάσεις. Το σύστημα με την ολοκλήρωση του προορίζεται να αποτελέσει μια εφαρμογή διαδικτύου και θα λειτουργεί σαν συμπληρωματικό σύστημα στο γραφείο διαχείρισης ερευνητικών προγραμμάτων του τμήματος πληροφορικής καθώς μέχρι σήμερα υπάρχει σύστημα μόνο για καταγραφή εισόδων και εξόδων. Οι προδιαγραφές του συστήματος έγιναν βάση των προδιαγραφών που δίνει το Πανεπιστήμιο Κύπρου για τα ερευνητικά προγράμματα. Αποτελεί κοινή ιδιοκτησία του Πανεπιστημίου Κύπρου και του ατόμου που το υλοποίησε.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή	1
	1.1 Σκοπός της διπλωματικής	1
	1.2 Τεχνολογία Λογισμικού	2
	1.3 Έρευνα	2
Κεφάλαιο 2	Απαιτήσεις	
	2.1 Εισαγωγή	5
	2.2 Γενική Περιγραφή	7
	2.3 Συγκεκριμένες Απαιτήσεις	11
Κεφάλαιο 3	Προδιαγραφές	
	3.1 Εισαγωγή	18
	3.2 Τρία Βήματα Αντικειμενοστρεφούς Ανάλυσης	21
	3.3 Software Project Management Plan	33
Κεφάλαιο 4	Σχεδιασμός	
	4.1 Αντικειμενοστρεφής Σχεδίαση	35
	4.2 Κατασκευή αναλυτικών Class Diagram	46
	4.3 Σχεδίαση προϊόντος μέσω “Clients Of Objects”	46
	4.4 Πίνακες Βάσης Δεδομένων	47
	4.5 Διάγραμματος Σχέσεων Οντοτήτων	50
Κεφάλαιο 5	Υλοποίηση	
	5.1 Περιγραφή εργαλείων που χρησιμοποιήθηκαν	52

5.2 Περιγραφή γλωσσών προγραμματισμού και βάσης δεδομένων	53
5.3 Αναφορά στα δύσκολα σημεία του κώδικα	54
5.4 Αποσπάσματα κώδικα	55
5.5 Περιπτώσεις δοκιμής	65
Κεφάλαιο 6 Συμπεράσματα	
6.1 Συμπεράσματα	66
6.2 Μελλοντική εργασία	67
6.3 Πλεονεκτήματα συστήματος	67
6.4 Μειονεκτήματα συστήματος	67
Βιβλιογραφία	68
Παράρτημα Α	A-1

Κεφάλαιο 1

Εισαγωγή

1.1 Σκοπός της διπλωματικής	1
1.2 Τεχνολογία Λογισμικού	2
1.3 Περιγραφή συστήματος	2

1.1 Σκοπός της διπλωματικής

Σκοπός της διπλωματικής μου εργασίας είναι η ανάπτυξη συστήματος διαχείρισης προσωπικών δαπανών των ερευνητικών προγραμμάτων του τμήματος Πληροφορικής. Στο συγκεκριμένο σύστημα θα έχουν πρόσβαση εξουσιοδοτημένοι χρήστες, καθηγητές και φοιτητές. Κάθε χρήστης θα έχει καθορισμένες προσβάσεις. Το σύστημα με την ολοκλήρωση του προορίζεται να αποτελέσει μια εφαρμογή διαδικτύου και θα λειτουργεί σαν συμπληρωματικό σύστημα στο γραφείο διαχείρισης ερευνητικών προγραμμάτων του τμήματος πληροφορικής καθώς μέχρι σήμερα υπάρχει σύστημα μόνο για καταγραφή εισόδων και εξόδων. Οι προδιαγραφές του συστήματος έγιναν βάση των προδιαγραφών που δίνει το Πανεπιστήμιο Κύπρου για τα ερευνητικά προγράμματα. Αποτελεί κοινή ιδιοκτησία του Πανεπιστημίου Κύπρου και του ατόμου που το υλοποίησε.

Η διπλωματική μου εργασία στοχεύει στην δημιουργία:

- μια βάσης δεδομένων για αποθήκευση όλων των προγραμμάτων, των ερευνητών, υπεύθυνων ερευνητών, προγραμμάτων, συμβολαίων και timesheet.
- μιας εφαρμογής που θα βρίσκεται στο διαδίκτυο από όπου θα έχουν πρόσβαση οι εξουσιοδοτημένοι χρήστες, οι καθηγητές και φοιτητές με καθορισμένες προσβάσεις.

1.2 Τεχνολογία Λογισμικού

Για την ανάπτυξη του συστήματος χρησιμοποιήσα διαδικασίες Τεχνολογίας Λογισμικού. Συγκεκριμένα το σύστημα πέρασε από τα πρώτα τέσσερα βήματα κύκλου ζωής ενός συστήματος:

- Απαιτήσεις Συστήματος
- Προδιαγραφές συστήματος
- Σχεδιασμός συστήματος
- Υλοποίηση συστήματος

Η πρώτη φάση του κύκλου ζωής ενός συστήματος στοχεύει στην εξαγωγή των απαιτήσεων του συστήματος, την ανεύρεση και την καταγραφή όλων των λειτουργικοτήτων που πρέπει να ικανοποιεί. Όπως επίσης και το αναγκαίο λογισμικό και υλικό για να μπορεί να τεθεί η εφαρμογή μας σε λειτουργία.

Η δεύτερη φάση, φάση προδιαγραφών αφορά την λεπτομερή ανάλυση των λειτουργιών που καταγράφηκαν στην φάση των απαιτήσεων με διάφορες τεχνικές ανάλυσης.

Η τρίτη φάση του σχεδιασμού είναι η λεπτομερής ανάλυση του πως θα λειτουργεί το σύστημα, καλύπτοντας όλες τις ανάγκες που εντοπίστηκαν και αναλύθηκαν.

Τέλος, η φάση της υλοποίησης αποτελεί την περιγραφή εργαλείων που χρησιμοποιήθηκαν, ενδεικτικός κώδικας που καταγράφηκε και περιγραφή διαπροσωπειών τελικού συστήματος.

1.3 Περιγραφή συστήματος

Στόχος του κεφαλαίου αυτού είναι μια σύντομη αλλά περιεκτική παρουσίαση του συστήματος. Το τμήμα πληροφορικής του Πανεπιστημίου Κύπρου αναλαμβάνει όλο και περισσότερα ερευνητικά προγράμματα με αποτέλεσμα να γίνει δύσκολη η διαχείριση τους. Πιο συγκεκριμένα με την αύξηση των ερευνητικών προγραμμάτων αυξήθηκε και ο

αριθμός των ερευνητών και γι αυτό πρέπει να δημιουργηθεί ένα σύστημα που θα κρατά κατάσταση για τις μηνιαίες απολαβές, πληρωμές για υπηρεσίες που προσέφεραν, δημιουργία timesheet και ετοιμασία αναγκαίων φορμών.

Για κάθε ερευνητικό πρόγραμμα κάθε μήνα αποκόπτονται μισθοί που αφορούν τους ερευνητές τους και πληρωμές για υπηρεσίες που προσφέρθηκαν. Για την παρακολούθηση των εξόδων και εσόδων κάθε ερευνητικού προγράμματος γίνεται διαχείριση της οικονομικής κατάστασης από το λογιστήριο και δίδεται σαν πίνακας excel στα άτομα του γραφείου. Το μόνο σύστημα που υπάρχει για το γραφείο διαχείρισης ερευνητικών προγραμμάτων είναι ένα για την καταγραφή των εξόδων κάθε προγράμματος. Δεν υπάρχει οποιοδήποτε άλλο σύστημα για να παρακολουθούν τις πληρωμές κάθε προγράμματος και την οικονομική κατάσταση της κατηγορίας αυτής.

Το σύστημα το οποίο θα αναπτυχθεί θα παρέχει πληροφορίες για την κατάσταση των προσωπικών δαπανών τόσο για πραγματικές πληρωμές όσο και για προσωπικές συνεισφορές (own contribution) για κάθε ερευνητικό πρόγραμμα. Από την έρευνα που έχει διεξαχθεί μέσα στα πλαίσια της διπλωματικής μου εργασίας διαφαίνεται ότι δεν υπάρχει σύστημα που να ικανοποιεί πλήρως τις ανάγκες διαχείρισης του personnel cost των ερευνητικών προγραμμάτων.

Το γραφείο διαχείρισης των ερευνητικών προγραμμάτων έχει ανάγκη από ένα σύστημα που να μπορεί να βλέπει τις τρέχον πληρωμές, τα timesheet που δεν έχουν συμπληρωθεί και το διαθέσιμο υπόλοιπο της κατηγορίας personnel κόστος. Το λογιστήριο παρέχει κατάσταση με όλες τις κατηγορίες και για να βρουν τα υπόλοιπα και όλες τις πληρωμές που έγιναν, το γραφείο κάθε φορά επεξεργάζεται το αρχείο για να εξάγει τις πληροφορίες που χρειάζεται. Το γραφείο λοιπόν χρειάζεται ένα πρόγραμμα που θα παράγει αυτόματα τα στοιχεία της οικονομικής κατάστασης κάθε προγράμματος.

1.3.2 Personnel Cost

Η κατηγορία αυτή είναι χωρισμένη σε δυο ομάδες τις πραγματικές πληρωμές και τις συνεισφορές. Όταν λέμε πληρωμές εννοούμε τις μηνιαίες απολαβές, πληρωμές για

υπηρεσίες και συμβόλαια. Συνεισφορές λέγονται οι υπηρεσίες που παρέχονται στο ερευνητικό πρόγραμμα χωρίς να χρεώνετε το πρόγραμμα.

Μηνιαίες απολαβές αφορούν αμοιβές σε ερευνητές τέλος κάθε μήνα ανάλογα με τις ώρες που εργάστηκαν. Πληρωμές για υπηρεσίες είναι οι αμοιβές σε άτομα για κάποια υπηρεσία που παρείχαν στο ερευνητικό πρόγραμμα και ήταν για ένα σύντομο χρονικό διάστημα. Πληρωμές βάσει συμβολαίων είναι οι αμοιβές στα άτομα τα οποία έχουν συμβόλαιο με το ερευνητικό πρόγραμμα και πληρώνονται για τις εργασίες που πραγματοποιούν. Όσο αφορά την δεύτερη κατηγορία, own contribution, έχουμε παροχή υπηρεσιών χωρίς να χρεώνετε το πρόγραμμα. Συνήθως οι καθηγητές δηλώνουν τις ώρες που εργάστηκαν και πληρώνονται από το Πανεπιστήμιο και όχι από το ερευνητικό πρόγραμμα.

Για την πραγματοποίηση οποιασδήποτε αμοιβής πρέπει να παρέχονται timesheet για να διαφαίνονται οι ώρες εργασίας κάθε ερευνητή και να μην ξεπερνιούνται οι ενδεικνυόμενες ώρες από τους νόμους που διέπουν το Πανεπιστήμιο. Με κάθε συμβόλαιο ή αμοιβή για παροχή υπηρεσιών υπάρχουν συγκεκριμένες φόρμες που πρέπει να συμπληρώνονται και να στέλνονται στην υπηρεσία ανθρώπινου δυναμικού και λογιστήριο αντίστοιχα.

1.3.3 Έρευνα για δεδομένα

Για εύρεση δεδομένων και καταγραφή τους ακολούθως στο σύστημα είχα συνεχή συνεργασία με τα δυο άτομα που εργάζονται στον τομέα διαχείρισης ερευνητικών προγραμμάτων του τμήματος πληροφορικής. Επίσης, είχα εργαστεί για ενάμισι χρόνο στο γραφείο αυτό σαν φοιτήτρια, οπότεν είχα μια καλή βάση για εξαγωγή δεδομένων και δημιουργία εικόνας για το πώς δουλεύει το συγκεκριμένο γραφείο και ποιες ήταν οι πραγματικές τους ανάγκες. Είναι πολύ σημαντικό να εξάγεις τα σωστά δεδομένα γιατί είναι αυτά που θα αποθηκεύεις στη βάση σου και μετά από επεξεργασία τους από το σύστημα θα παίρνεις τις απαιτούμενες πληροφορίες. Όταν κτίσεις μια σωστή βάση με τα κατάλληλα δεδομένα τότε είναι εύκολη η διεξαγωγή πληροφοριών.

Κεφάλαιο 2

Απαιτήσεις Συστήματος

2.1 Εισαγωγή	5
2.2 Γενική Περιγραφή	7
2.3 Συγκεκριμένες Απαιτήσεις	11

2.1 Εισαγωγή

2.1.1 Σκοπός και αποδέκτες

Σκοπός είναι η δημιουργία ενός συστήματος διαχείρισης ερευνητικών προγραμμάτων από όπου υπάλληλοι του γραφείου διαχείρισης ερευνητικών προγραμμάτων. Πρόσβαση για συμπλήρωση των timesheet θα έχουν οι υπεύθυνου ερευνητές και οι ερευνητές. Θα δημιουργηθεί μια βάση δεδομένων που θα κρατά όλες τις σχετικές πληροφορίες. Στην βάση αυτή θα κρατούνται όλα τα σχετικά στοιχεία για τους ερευνητές, τα ερευνητικά προγράμματα, τους υπεύθυνους ερευνητές και τους χρήστες του συστήματος. Τέλος, θα κρατά και την οικονομική κατάσταση κάθε προγράμματος για την κατηγορία των προσωπικών δαπανών.

Αποδέκτες του συγκεκριμένου συστήματος είναι το τμήμα Πληροφορικής του Πανεπιστημίου Κύπρου. Αυτό το έγγραφο απευθύνεται σε όλα τα άτομα που θα εμπλακούν και θα συμβάλουν στην ανάπτυξη και την υλοποίηση του Συστήματος. Απευθύνεται σε όσους ενδιαφέρονται να μάθουν τις λειτουργικότητες και τις δυνατότητες του μελλοντικού Συστήματος. Επίσης, απευθύνεται σε όσους θα χρησιμοποιήσουν άμεσα το Σύστημα έτσι ώστε να έχουν την ευκαιρία να εκφράσουν τις απόψεις τους.

2.1.2 Εμβέλεια

Το σύστημα ονομάζεται «Διαχείριση Ερευνητικών Προγραμμάτων Πληροφορικής». Η εφαρμογή θα έχει ως στόχους τη διαχείριση τις οικονομικής κατάστασης των προσωπικών

κοστών κάθε ερευνητικού προγράμματος του τμήματος. Επιπλέον, θα μπορεί να διαχειρίζεται τα timesheet και τις φόρμες που απαιτούνται για να πραγματοποιηθεί μια πληρωμή ή ένα συμβόλαιο. Για την επίτευξη των πιο πάνω στόχων θα δημιουργηθεί βάση δεδομένων η οποία θα κρατά τα στοιχεία των ερευνητών, ερευνητικών προγραμμάτων, χρηστών, πληρωμές σε άτομα και λογιστικών λεπτομερειών.

2.1.3 Ορισμοί, Αρκτικόλεξα και Συντομογραφίες

USERNAME: όνομα χρήστη για πρόσβαση στο σύστημα.

PASSWORD: μυστικός κωδικός πρόσβασης του κάθε χρήστη στο σύστημα.

Administrator: Διαχειριστής του συστήματος

Interface: Διαπροσωπεία

User Interfaces: Διασύνδεση Χρηστών

Hardware Interfaces: Διασύνδεση Υλικού

Memory Constraints: Περιορισμοί μνήμης

Backup = Εφεδρικά αντίγραφα αρχείων

Reports = Αναφορές

Timesheet=έντυπο όπου συμπληρώνονται οι ώρες εργασίας για μια περίοδο

2.1.4 Αναφορές

- Πρότυπο IEEE 830-1998, αναθεώρηση του IEE 830-1993
- “Software Project Survival Guide Steve McConnell, Microsoft Press (1998)”

2.1.5 Σύνοψη

Πιο κάτω ακολουθεί μια σύντομη περιγραφή του τι θα ακολουθήσει στο έγγραφο αυτό. Στο κεφαλαίο 2 που ακολουθεί γίνεται μια γενική περιγραφή του συστήματος σε σχέση με τις διάφορες διαπροσωπείες αλλά καθορίζεται και η προοπτική του συστήματος.

Προχωρώντας ακολούθως στο κεφαλαίο 3 έχουμε μια πιο λεπτομερή περιγραφή του συστήματος. Γίνεται μια λεπτομερής περιγραφή του συστήματος σε σχέση με τις διάφορες διαπροσωπείες. Ακόμα, αναφέρονται αναλυτικότερα οι λειτουργίες του συστήματος. Σαν επέκταση του προηγούμενου κεφαλαίου αναφέρονται απαιτήσεις απόδοσης αν και εφόσον υπάρχουν. Επίσης στα χαρακτηριστικά του συστήματος θίγονται αναλυτικά τα θέματα αξιοπιστίας, διαθεσιμότητας και ασφάλειας του συστήματος. Ακολουθούν στο ίδιο

κεφαλαίο θέματα που αφορούν την συντήρηση και συμβατότητα του συστήματος με άλλα συστήματα που πιθανόν να συνεργάζεται. Επίσης αναφέρονται οι λογικές απαιτήσεις σε βάσεις δεδομένων αλλά και τυχόν άλλες απαιτήσεις. Τέλος έχουμε ένα ευρετήριο και τα παραρτήματα του εγγράφου αυτού.

2.2 Γενική Περιγραφή

2.2.1 Προοπτική Προϊόντος

Το σύστημα που θα υλοποιήσω θα βρίσκεται 24ώρες το εικοσιτετράωρο στο διαδίκτυο. Έτσι οι χρήστες του συστήματος θα μπορούν οποιαδήποτε στιγμή να εισέλθουν στο σύστημα και να το χρησιμοποιήσουν. Συγκεκριμένα το σύστημα μου θα διαχειρίζεται τα timesheet οπότεν θα μπορούν ερευνητές να το συμπληρώσουν. Επίσης θα είναι διαθέσιμο και για το γραφείο διαχείρισης ερευνητικών προγραμμάτων του τμήματος σαν βοηθητικό για την καταγραφή της οικονομικής κατάσταση στην κατηγορία προσωπικών δαπανών. Ως σύστημα δεν θα είναι πλήρως ανεξάρτητο, αφού όλες οι πληροφορίες, τα στοιχεία και τα δεδομένα πρέπει να υπάρχουν σε κάποια βάση δεδομένων έτσι ώστε οποιαδήποτε στιγμή χρειαστεί να μπορούν να ανακληθούν ή να ενημερωθούν. Ως εκ τούτου, η εφαρμογή που θα αναπτύξω θα πρέπει να είναι σε θέση να επικοινωνεί με μία βάση δεδομένων με δυνατότητα να χειρίζεται των όγκο δεδομένων που απαιτείται για την ικανοποίηση των αναγκών του νοσοκομείου που θα το χρησιμοποιεί.

2.2.1.1 Διαπροσωπείες Συστήματος

Το σύστημα που θα αναπτύξω δεν θα διασυνδεθεί με οποιοδήποτε υπάρχον σύστημα. Η βασική διασύνδεση που θα χρειαστεί θα είναι αυτή μεταξύ του λογισμικού μας με τη βάση δεδομένων.

2.2.1.2 Διαπροσωπείες Χρηστών

Η διαπροσωπεία του συστήματος θα είναι κοινή και θα ακολουθεί τον ίδιο τύπο για όλους τους τύπους χρηστών. Δεν θα υπάρχει κάποια διαφοροποίηση όσο αφορά τις λειτουργίες που θα εκτελεί ο κάθε ένας, αν και αναμένεται ο τρόπος με τον οποίο αλληλεπιδρά ο κάθε χρήστης με το σύστημα να διαφέρει ελαφρά, χωρίς βέβαια να σημαίνει πως αυτό είναι απόλυτο.

Τόσο οι σελίδες του συστήματος που θα χρησιμοποιεί ο χρήστης σαν περιβάλλον του συστήματος όσο και οι αναφορές θα έχουν παρόμοια μορφή μεταξύ τους, ώστε να επιτυγχάνεται η ευχρηστία και η καλύτερη και γρηγορότερη κατανόηση του λογισμικού. Η μορφή αυτή θα είναι η μορφή που έχουν οι σελίδες διαδικτύου που κατασκευάζουμε στο εργαλείο Microsoft Visual Studio 2010, όπως αυτή εμφανίζεται στις σελίδες του εγγράφου αυτού και συγκεκριμένα στην ενότητα 2. Δηλαδή θα είναι κατά πολύ όμοια με τις φόρμες που τρέχουν οι εφαρμογές του λειτουργικού συστήματος Microsoft Windows.

Τα κουμπιά που χρησιμοποιούνται έχουν όλα το ίδιο μέγεθος, και επιπρόσθετα ο χώρος στον οποίο βρίσκονται στις ανάλογες φόρμες είναι ο ίδιος και εξαρτάται από τη λειτουργία του καθενός. Επιπρόσθετα, το μενού θα βρίσκεται στην ίδια θέση σε όλες τις σελίδες του συστήματος και όλες θα έχουν τις ίδιες επιλογές.

Για ενίσχυση της συνοχής του λογισμικού τα πεδία και οι επικεφαλίδες τους που εμφανίζονται στις σελίδες θα εμφανίζονται και στις αντίστοιχες αναφορές που θα εξάγονται από το σύστημα ώστε να αποφευχθούν παρεξηγήσεις και προβλήματα κατανόησης και εξοικείωσης με το λογισμικό.

2.2.1.3 Διαπροσωπείες Υλικού

Για να μπορεί να λειτουργήσει το σύστημα θα πρέπει να υπάρχει ένας server όπου και αν βρίσκεται η βάση δεδομένων του συστήματος καθώς και ηλεκτρονικός υπολογιστής από όπου ο χρήστης θα μπορεί να εισέλθει στο σύστημα.

2.2.1.4 Διαπροσωπείες Λογισμικού

Το σύστημα θα λειτουργεί στο διαδίκτυο και οι χρήστες θα μπορούν να έχουν πρόσβαση σε αυτό από οποιοδήποτε χώρο φτάνει το υπολογιστής που θα χρησιμοποιήσουν να είναι συνδεδεμένος με το διαδίκτυο.

2.2.1.5 Διαπροσωπείες Επικοινωνίας

Το σύστημα που θα αναπτύξω απαιτεί επικοινωνία μέσω διαδικτύου. Λόγω της χρήσης διαδικτύου και του περιορισμού πρόσβασης σε δεδομένα ανάλογα με τον τύπο του χρήστη που θα εισέλθει στο σύστημα θα πρέπει να εισαγάγει κωδικούς.

2.2.1.6 Λειτουργίες

Το σύστημα θα έχει καταχωρημένα στη βάση του τα στοιχεία των προγραμμάτων και την κατάσταση της κατηγορίας προσωπικών δαπανών, τους ερευνητές, τους χρήστες, τις πληρωμές και συμβόλαια. Ο χρήστης τύπου administrator μπορεί να έχει πρόσβαση σε όλα τα στοιχεία που αφορούν το σύστημα ενώ οι ερευνητές θα έχουν μόνο στα timesheet.

Οι σημαντικότερες λειτουργίες που θα εκτελεί το σύστημα περιγράφονται πιο κάτω:

- Προσθήκη, ενημέρωση, διαγραφή χρηστών, αφού πρώτα επιλεγεί η ειδικότητα του χρήστη καθώς και το προφίλ του.
- Κατηγοροποίηση των χρηστών σε ομάδες και καθορισμός δικαιωμάτων εισαγωγής χρηστών.
- Εισαγωγή, ενημέρωση, διαγραφή προγράμματος καθώς και τα οικονομικά του.
- Εισαγωγή, ενημέρωση, διαγραφή ερευνητή
- Εισαγωγή, ενημέρωση, διαγραφή υπεύθυνου ερευνητή
- Εισαγωγή, ενημέρωση, διαγραφή συμβολαίου
- Εισαγωγή, ενημέρωση, διαγραφή τύπου προγράμματος
- Εισαγωγή timesheet αφού πρώτα γίνει εύρεση για το ερευνητικό πρόγραμμα και τον ερευνητή που αφορά.
- Εκτύπωση αναφορών για τα στοιχεία της βάσης δεδομένων για το κάθε πρόγραμμα
- Εύκολη πρόσβαση στα στοιχεία του συστήματος.

2.2.2 Λειτουργίες Προϊόντος

Οι πιο σημαντικές από τις λειτουργίες που θα εκτελεί το πρόγραμμα είναι οι ακόλουθες:

- Εισαγωγή και ενημέρωση στοιχείων:
 - Περιλαμβάνει εισαγωγή και ενημέρωση στοιχείων που αφορούν ερευνητές, ερευνητικά προγράμματα και συμβόλαια
 - Περιλαμβάνει τη διαχείριση στοιχείων που αφορούν ερευνητές, ερευνητικά προγράμματα και συμβόλαια

- Καταχώρηση timesheet.
- Εύρεση στοιχείων και καταχώρηση, ενημέρωση, διαγραφή συμβολαίων για πληρωμές.
- Διαχείριση της κατηγορίας των personnel δαπανών για το κάθε ερευνητικό πρόγραμμα.
- Έλεγχος και παρακολούθηση χρηστών που έχουν καταχωρηθεί, ενημερωθεί από το λειτουργό του συστήματος.
- Εξαγωγή αναφορών βασισμένες σε διάφορα κριτήρια που θέτει ο χρήστης και αφορούν οποιαδήποτε στοιχεία είναι αποθηκευμένα στη βάση.

2.2.3 Χαρακτηριστικά Χρηστών

Οι κύριοι χρήστες που θα χρησιμοποιούν το συγκεκριμένο σύστημα αναμένεται να είναι μέσοι χρήστες με μέτρια εμπειρία στην χρήση ηλεκτρονικών υπολογιστών. Έτσι κρίνεται απαραίτητη η δημιουργία ενός φιλικού περιβάλλοντος προς το χρήστη καθώς και πλήρης καθοδήγηση και βοήθεια. Οι ερευνητές θα είναι πολύ καλοί χρήστες των ηλεκτρονικών υπολογιστών (κυρίως μεταπτυχιακοί πληροφορικής και καθηγητές πανεπιστημίου τμήματος πληροφορικής) αλλά θα έχουν πρόσβαση μόνο σε πολύ μικρό κομμάτι της διπλωματικής.

2.2.4 Περιορισμοί

Λόγω χρήσης της Microsoft SQL σαν εργαλείο υλοποίησης της βάσης η λειτουργία του περιορίζεται σε λειτουργικά συστήματα της Microsoft (windows). Η πιθανότητα φύλαξης λανθασμένων πληροφοριών πρέπει να είναι μηδαμινή λόγω της σημαντικότητας τους, Επιπλέον τεχνικές ελέγχου θα πρέπει να υλοποιηθούν γιατί το σκοπό. Ακόμη, λόγω του μεγάλου βαθμού εμπιστευτικότητας των πληροφοριών, επιπλέον έλεγχος και στον τομέα της ασφάλειας θα είναι απαραίτητος.

Οι κωδικοί πρόσβασης οι οποίοι θα προέρχονται από τον διαχειριστή (λειτουργό) του συστήματος θα είναι μοναδικοί και θα δίνονται ερευνητές και χρήστες όταν το προφίλ τους υπάρχει ήδη στο σύστημα. Κάθε χρήστης θα γνωρίζει μόνο το δικό του κωδικό πρόσβασης.

2.3 Συγκεκριμένες Απαιτήσεις

2.3.1 Εξωτερικές Απαιτήσεις Διαπροσωπείας

Στο σύστημα υπάρχει μόνο μια μεγάλη κατηγορία φορμών και αυτό οφείλετε στο ότι δεν υπάρχουν φόρμες με περιορισμένη πρόσβαση.

2.3.1.1 Διαπροσωπείες Χρηστών

Καθώς το επίπεδο εξοικείωσης και γνώσης ηλεκτρονικών υπολογιστών και από τους δύο χρήστες του συστήματος μας περιορίζεται στις βασικές γενικές γνώσεις απαιτείται μια ευκατανόητη σχεδίαση των φορμών και των αναφορών που να έχει συνοχή σε ολόκληρο το σύστημα ώστε να μην προκαλεί σύγχυση. Ως βασικό χαρακτηριστικό που θα πρέπει να πληρούν οι σελίδες του συστήματος είναι η συναφής μορφή.

Εξάλλου, το μέγεθος και ο χώρος στον οποίο τοποθετούνται τα κουμπιά σε κάθε φόρμα αποφάσισα να εξαρτάται από τη λειτουργία για την οποία προορίζονται και θα είναι κοινά σε όλες τις παρόμοιες σελίδες. Πιο συγκεκριμένα τα κουμπιά του μενού σε όλες τις σελίδες που χρησιμοποιούνται για πλοήγηση μεταξύ των διαφόρων σελίδων τοποθετήθηκαν στο ίδιο σημείο κάθε φόρμας. Τα κουμπιά που χρησιμοποιούνται για καταχώρηση, ενημέρωση και διαγραφή στοιχείων βρίσκονται στο κάτω δεξιά μέρος κάθε φόρμας.

Ακόμη, πεδία τα οποία μπορούν να λάβουν συγκεκριμένες τιμές αποφασίσαμε όπως τα ορίσουμε σαν λίστες περιορισμένης επιλογής ώστε να επιταχυνθεί η εισαγωγή δεδομένων και να αποφευχθούν σφάλματα κατά την εισαγωγή και αναζήτηση.

Τέλος, οι αναφορές θα περιέχουν τα ίδια πεδία και επικεφαλίδες που έχουν και οι φόρμες του προγράμματος για να επιτευχθεί μεγαλύτερος βαθμός συνοχής και συσχέτισης μεταξύ των δύο, και να είναι πιο καλά κατανοητή η χρήση τους από τον χρήστη.

2.3.1.2 Διαπροσωπείες Υλικού

Για να μπορεί να λειτουργήσει το σύστημα θα πρέπει να υπάρχει ένας server όπου θα βρίσκεται εγκατεστημένη η βάση δεδομένων του συστήματος καθώς και ηλεκτρονικός υπολογιστής από όπου ο χρήστης θα μπορεί να εισέλθει στο σύστημα. Λόγω του ότι είναι πολλά από τα δεδομένα που θα αποθηκεύονται στη βάση θα πρέπει να υπάρχει μεγάλος χώρος για αποθήκευση των δεδομένων.

2.3.1.3 Διαπροσωπεία Λογισμικού

Το σύστημα θα λειτουργεί στο διαδίκτυο και οι χρήστες θα μπορούν να έχουν πρόσβαση σε αυτό από οποιοδήποτε χώρο φτάνει το υπολογιστής που θα χρησιμοποιήσουν να είναι συνδεδεμένος με το διαδίκτυο.

2.3.1.4 Διαπροσωπεία Επικοινωνίας

Το σύστημα που θα αναπτύξω απαιτεί επικοινωνία μέσω διαδικτύου. Λόγω της χρήσης διαδικτύου οι χρήστες θα μπορούν να εισέλθουν στο σύστημα μονό με εισαγωγή κωδικών.

2.3.2 Χαρακτηριστικά Γνωρίσματα Προϊόντος Λογισμικού

Η διαπροσωπεία χρήστη, τα δικαιώματα πρόσβασης, καθώς και οι δυνατότητες αλλαγής στη βάση θα είναι τα ίδια και για όλους του χρήστες του συστήματος. Συνεπώς, όποτε κάνει login στο σύστημα κάποιος χρήστης θα του παρουσιάζεται το ίδιο κεντρικό μενού και από εκεί θα έχει πρόσβαση στις διάφορες λειτουργίες του προγράμματος.

Πιο κάτω αναφέρονται οι χαρακτηριστικές λειτουργίες του συστήματος που πρόκειται να αναπτύξω.

2.3.2.1 Σελίδα Εισαγωγής Κωδικών

- **Περιγραφή**

Ο χρήστης εισάγει των κωδικό και το όνομα χρήστη ώστε να μπορεί να εισέλθει στο σύστημα.

- **Έισοδος**

Ο χρήστης εισάγει τους κωδικού πρόσβασης του που του έχουν δοθεί από το λειτουργό του συστήματος αφού πρώτα έχει καταχωρηθεί το προφίλ του στο σύστημα

- **Έξοδος**

Εισαγωγή στο σύστημα στην αρχική σελίδα.

- **Πρόσβαση**

Πρόσβαση σε όλους τους χρήστες με κωδικούς. Γίνεται πρόσβαση εφόσον καταχωρήσει σωστά τα πεδία και το ρόλο χρήσης του συστήματος.

2.3.2.2 Αρχική Σελίδα

Παρόμοια γίνεται η εισαγωγή υπεύθυνου ερευνητή και προγράμματος γι' αυτό δεν αναλύεται εδώ.

- **Περιγραφή**

Ο χρήστης έχει εισαχθεί στο σύστημα και επιλέγει από το μενού την λειτουργία που θέλει να πραγματοποιήσει.

- **Έισοδος**

Ο χρήστης επιλέγει από το μενού τη λειτουργία που θέλει να μεταβεί.

- **Έξοδος**

Εισαγωγή στη σελίδα που επέλεξε.

- **Πρόσβαση**

Πρόσβαση σε μόνο στους δυο χρήστες με κωδικούς.

2.3.2.3 Σελίδα εισαγωγής ερευνητή

Παρόμοια γίνεται η εισαγωγή γιατρού και νοσοκόμου και δεν αναλύονται εδώ.

- **Περιγραφή**

Ο χρήστης συμπληρώνει τα πεδία με τα στοιχεία του ερευνητη για να καταχωρηθεί στο σύστημα.

- **Έισοδος**

Ο χρήστης συμπληρώνει τα στοιχεία του ερευνητή στα πεδία. Ακολούθως επιλέγει το κουμπί καταχώρηση ή ακύρωση.

- **Έξοδος**

Εμφάνιση μηνύματος επιτυχής καταχώρησης ή σφάλματος που δεν κατέστησε εφικτή την καταχώρησης. Ενώ σε περίπτωση επιλογής ακύρωσης διαγραφή όλων των στοιχείων από τα πεδία.

- **Πρόσβαση**

Πρόσβαση μόνο στους χρήστες λειτουργούς με κωδικούς.

2.3.2.4 Σελίδα εισαγωγής συμβολαίου

Είναι για οποιαδήποτε είδος πληρωμής ερευνητή ή υπεύθυνου ερευνητή.

- **Περιγραφή**

Αρχικά ο χρήστης μέσα από αναζήτηση επιλέγει το πρόγραμμα και ακολούθως επιλέγει τον ερευνητή ή τον υπεύθυνο ερευνητή για να γίνει πληρωμή από το πρόγραμμα προς τον ερευνητή. Τέλος, ο χρήστης συμπληρώνει τα πεδία με τα στοιχεία του συμβολαίου για να καταχωρηθεί στο σύστημα.

- **Έισοδος**

Αφού γίνει επιλογή προγράμματος και ερευνητή τότε ο χρήστης συμπληρώνει τα στοιχεία του συμβολαίου στα πεδία. Ακολούθως επιλέγει το κουμπί καταχώρηση ή ακύρωση.

- **Έξοδος**

Εμφάνιση μηνύματος επιτυχής καταχώρησης ή σφάλματος που δεν κατέστησε εφικτή την καταχώρησης. Ενώ σε περίπτωση επιλογής ακύρωσης διαγραφή όλων των στοιχείων από τα πεδία.

- **Πρόσβαση**

Πρόσβαση μόνο στους χρήστες λειτουργούς με κωδικούς.

2.3.2.5 Σελίδα εισαγωγής τύπου ερευνητικού προγράμματος

- **Περιγραφή**

Ο χρήστης εισάγει τον νέο τύπο ερευνητικού προγράμματος για να καταχωρηθεί στο σύστημα.

- **Έισοδος**

Ο χρήστης συμπληρώνει τα πεδία του τύπου ερευνητικού προγράμματος για να καταχωρηθεί στο σύστημα. Ακολούθως επιλέγει το κουμπί καταχώρηση ή ακύρωση.

- **Έξοδος**

Εμφάνιση μηνύματος επιτυχής καταχώρησης ή σφάλματος που δεν κατέστησε εφικτή την καταχώρησης. Ενώ σε περίπτωση επιλογής ακύρωσης διαγραφή όλων των στοιχείων από τα πεδία.

- **Πρόσβαση**

Πρόσβαση μόνο στους χρήστες λειτουργούς με κωδικούς.

2.2.3 Απαιτήσεις Απόδοσης

Οι απαιτήσεις απόδοσης που υπάρχουν είναι οι εξής:

Το σύστημα θα πρέπει να μπορεί να χρησιμοποιείται ταυτόχρονα από 2 ή και περισσότερους χρήστες. Ο αριθμός των δεδομένων που θα αποθηκεύονται στη βάση είναι σχετικά μεγάλος. Οι χρήστες το μόνο που θα κάνουν είναι να αναζητούν δεδομένα στη βάση, να καταχωρούν νέα δεδομένα, να ενημερώνουν ή να διαγράφουν δεδομένα που υπάρχουν στη βάση και δεν τα χρειάζονται πλέον.

Η εκτέλεση του κάθε Query στην βάση δεδομένων δεν θα πρέπει να ξεπερνάει τα 12 δευτερόλεπτα σε καμία εκτέλεση. Ο χρόνος αυτός πρέπει να τηρείται και στην περίπτωση όπου θα έχουμε reports, στην οποία έχουμε καθυστέρηση λόγω της καθυστέρησης που προκαλεί το πρόγραμμα που χρησιμοποιείται για τη δημιουργία τους από τη βάση μας και της ταυτόχρονης πρόσβασης πέραν του ενός χρήστη σε αυτή.

2.2.4 Περιορισμοί Σχεδιασμού

Η υλοποίηση της βάσης δεδομένων θα γίνει σε γλώσσα Microsoft SQL και η υλοποίηση των σελίδων θα γίνει σε asp με Visual Basic Script, γιατί διαθέτει όλα τα εργαλεία που υποστηρίζουν τη δημιουργία διπροσωπίας βάσεων δεδομένων ώστε να αποφευχθούν συγκεκριμένοι και ιδιαίτεροι περιορισμοί στον σχεδιασμό, είναι εύκολη σαν γλώσσα και τέλος λόγω του ότι διατίθεται δωρεάν. Κατά τη φάση του σχεδιασμού θα ακολουθηθεί αντικειμενοστρεφής προσέγγιση. Η μέθοδος αυτή θα βοηθήσει σε μελλοντικό στάδιο την ενσωμάτωση επιπρόσθετων λειτουργιών – δυνατοτήτων στο σύστημα αλλά ταυτόχρονα θα κάνει πιο εύκολο το έργο της συντήρησης του. Ακόμη αυτό θα βοηθήσει σημαντικά σε μετέπειτα σύνδεση του συστήματος με άλλα.

2.2.5 Χαρακτηριστικά Συστήματος Λογισμικού

2.2.5.1 Αξιοπιστία

Πολύ σημαντικός παράγοντας για την ομαλή λειτουργία του συστήματος. Για το σκοπό αυτό εφαρμόζουμε μια σειρά από ελέγχους για να δούμε πόσο αξιόπιστο είναι το σύστημα:

- Συχνότητα εμφάνισης λαθών
- Μέσο χρόνο μεταξύ λαθών
- Χρόνος που χρειάζεται το σύστημα να επανέλθει σε λειτουργία μετά από κάποιο σφάλμα.

2.2.5.2 Διαθεσιμότητα

Το σύστημα μας θα είναι διαθέσιμο σε όλους τους χρήστες όλες τις μέρες 24 ώρες το 24ωρο αρκεί ο υπολογιστής που θα έχει το ρόλο του server να είναι ανοικτός και συνεπώς να μπορούν να ενώνονται στο διαδίκτυο όλοι οι χρήστες με κωδικούς πρόσβασης..

2.2.5.3 Ασφάλεια

Για να μπορέσει κάποιος να λειτουργήσει το σύστημα θα πρέπει να εισάγει κωδικό χρήστη (username) και συνθηματικό (password). Ο λειτουργός του συστήματος θα μπορεί να χειριστεί το σύστημα και θα έχει τη δυνατότητα να δίνει κωδικούς στους χρήστες. Οι κωδικοί στο σύστημα μπαίνουν καθαρά για θέματα ασφάλειας λόγω του δικαιώματος του ιατρικού απορρήτου των ασθενών.

2.2.5.4 Συντηρησιμότητα

Η χρήση αντικειμεντοστρεφούς προσέγγισης μπορεί να βελτιώσει σημαντικά τη συντηρησιμότητα του συστήματος. Μπορεί εύκολα να γίνει ο εντοπισμός και η διόρθωση σφαλμάτων αλλά και η πιο ομαλή επέκταση του συστήματος μελλοντικά.

2.2.5.5 Μεταφερσιμότητα

Το σύστημα θα υλοποιηθεί σε συγκεκριμένη πλατφόρμα και δεν θα μπορεί να μεταφερθεί αλλού.

2.2.6 Λογικές Απαιτήσεις Βάσης Δεδομένων

Ορισμός Οντοτήτων:

Παρακάτω γίνεται ορισμός των οντοτήτων της βάσης με σκοπό να καθοριστούν και να οργανωθούν τα δεδομένα που θα αποθηκευτούν.

Οι οντότητες του συστήματος είναι :

- **Οι ερευνητές:** Ως ερευνητές θεωρούνται όλοι τα άτομα τα οποία εργάζονται σε κάποιο από τα ερευνητικά προγράμματα και παίρνουν αμοιβή για τις υπηρεσίες που παρέχουν. Για να μπορούν να δοθούν κωδικοί σε ένα ερευνητή θα πρέπει να υπάρχει το προφίλ του καταχωρημένο στη βάση.
- **Οι υπεύθυνοι ερευνητές:** Είναι όλοι οι υπεύθυνοι ερευνητικών προγραμμάτων μαζί με τους λειτουργούς που θα χρησιμοποιούν το σύστημα.
- **Προγράμματα:** Είναι τα ερευνητικά προγράμματα τα οποία διαχειρίζεται μέσω του συστήματος οι λειτουργοί. Για κάθε πρόγραμμα κρατείται κατάσταση για τα οικονομικά του και πιο συγκεκριμένα για τις δαπάνες σε προσωπικό.
- **Συμβόλαια:** Είναι οι διάφορες πληρωμές που γίνονται σε άτομα. Μπορεί να είναι μηνιαίος μισθός, αμοιβή για υπηρεσίες και own contribution. Μέσω της διαχείρισης τους παράγονται excel αρχεία με τις πληροφορίες τους.
- **Timesheet:** Οι ερευνητές και οι υπεύθυνοι ερευνητές μπορούν να επιλέξουν όσα προγράμματα επιθυμούν και να δημιουργήσουν timesheet βλέποντας τις ώρες που δούλεψαν μια μέρα σε κάθε πρόγραμμα και να τηρούν τους νόμους που διέπουν το Πανεπιστήμιο Κάρου για τα ωράρια εργασίας.
- **Αναφορές:** Με βάση τα δεδομένα που υπάρχουν στη βάση δεδομένων παράγονται αναφορές με ομαδοποιημένες πληροφορίες ανά ερευνητή, ανά υπεύθυνο ερευνητή, ανά πρόγραμμα, συμβόλαιο κτλ.

Κεφάλαιο 3

Προδιαγραφές

3.1 Εισαγωγή	18
3.2 Τρία Βήματα Αντικειμενοστρεφούς Ανάλυσης	21
3.3 Software Project Management Plan	33

3.1 Εισαγωγή

3.1.1 Διάγραμμα Ροής Δεδομένων

3.1.1.1 Σημειογραφία που χρησιμοποιείται



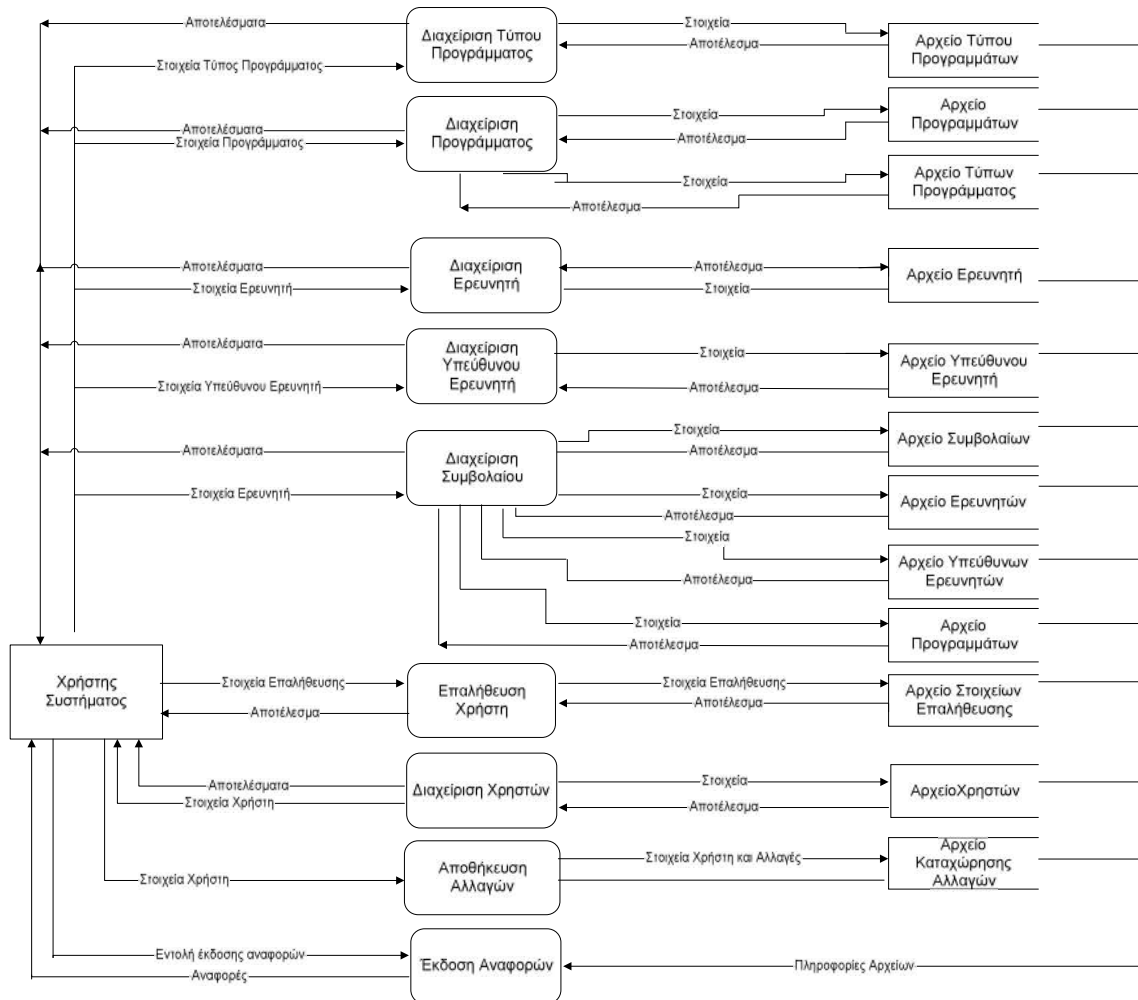
Εξωτερικός πράκτορας: Οι εξωτερικοί πράκτορες αντιπροσωπεύουν τα σημεία αλληλεπίδρασης του συστήματος με το εξωτερικό περιβάλλον. Επιπλέον καθορίζουν την εμβέλεια του συστήματος.

Διαδικασία: Οι διαδικασίες αντιπροσωπεύουν τις διάφορες διαδικασίες που εκτελούνται στο σύστημα ή που προκαλούνται από την είσοδο δεδομένων στο σύστημα.

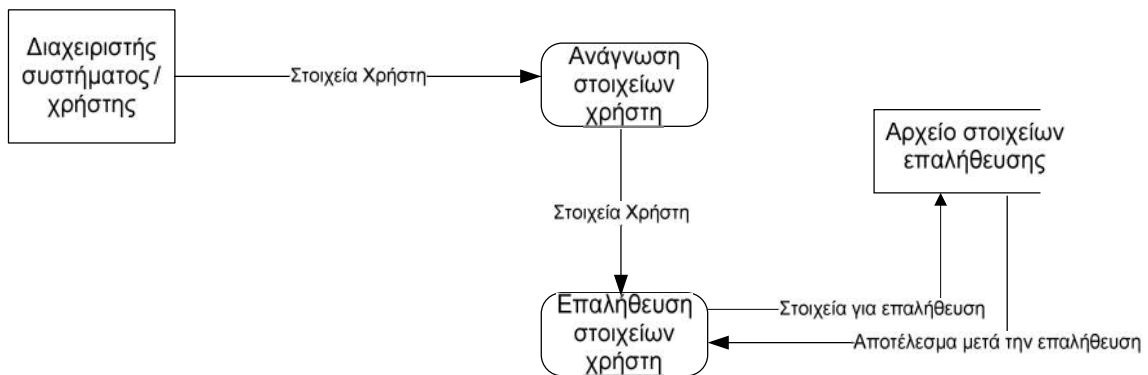
Αποθηκευτική μονάδα: Είναι οι «χώροι» στους οποίους αποθηκεύονται δεδομένα τα οποία χρησιμοποιεί το σύστημα για κάποια λειτουργία του, ή αποθηκεύει δεδομένα μετά από την εκτέλεση κάποιας διεργασίας.

Ροή των Δεδομένων: Ροή είναι μια μεταβίβαση κομματιών (ή «πακέτων») πληροφορίας από ένα τμήμα του συστήματος σε ένα άλλο.

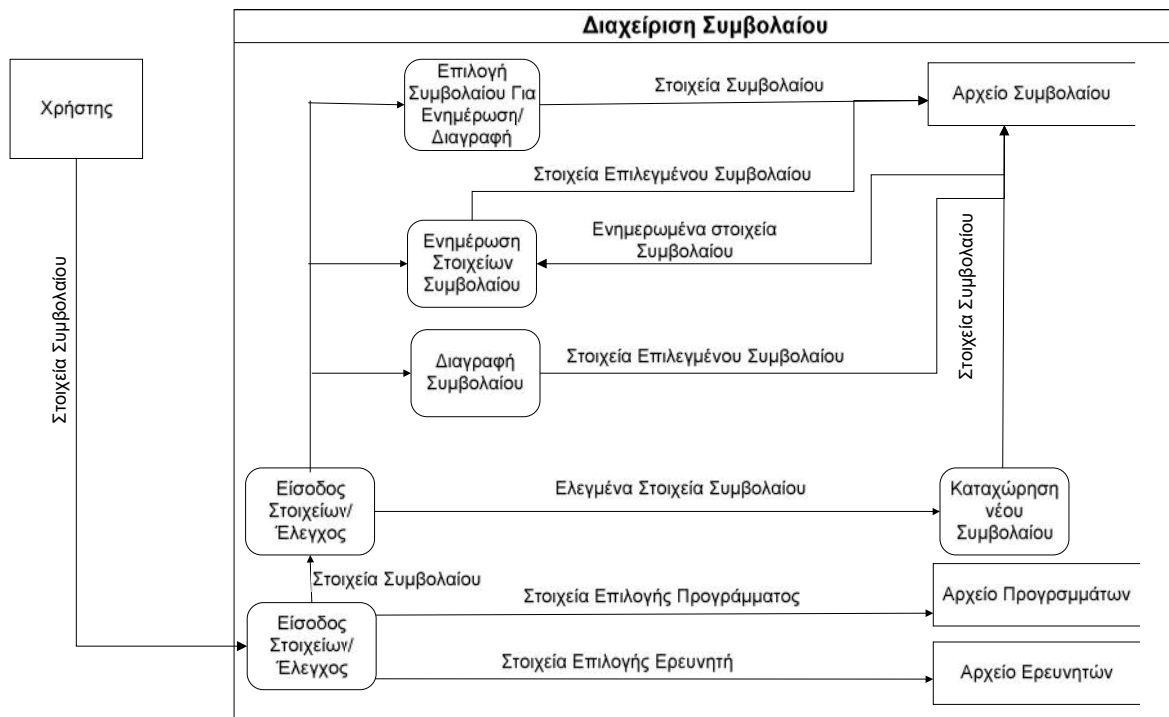
3.1.1.1 ΔΡΑ Διαχείρισης Συστήματος



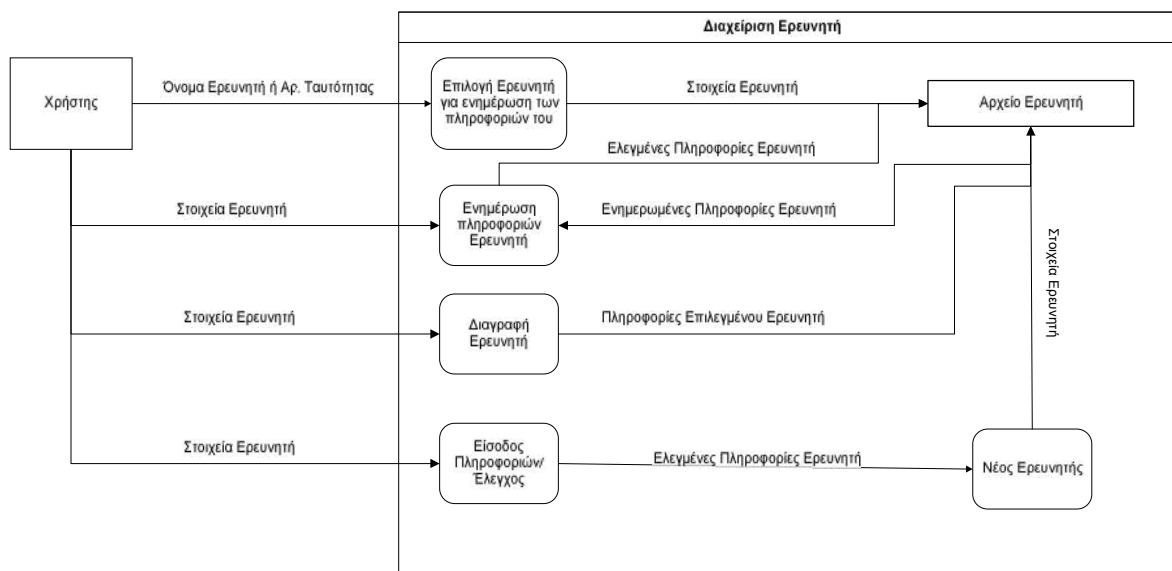
3.1.1.1.2 ΔΡΑ Εισόδου στο σύστημα



3.1.1.1.3 ΔΡΑ Συμβολαίων



3.1.1.1.4 ΔΡΑ Ερευνητή



3.2 Τρία Βήματα Αντικειμενοστρεφούς Ανάλυσης

3.2.1 Use Case Diagrams

Ένα διάγραμμα Use Case είναι μια γραφική αναπαράσταση υψηλού επιπέδου της λειτουργίας του συστήματος. Περιλαμβάνει use cases τα οποία αποτελούν τα κομμάτια της λειτουργικότητας όπου θα παρέχει το σύστημα και τους actors όπου είναι ένα άτομο, σύστημα, κομμάτι του hardware ή οτιδήποτε άλλο αλληλεπιδρά με το σύστημα.

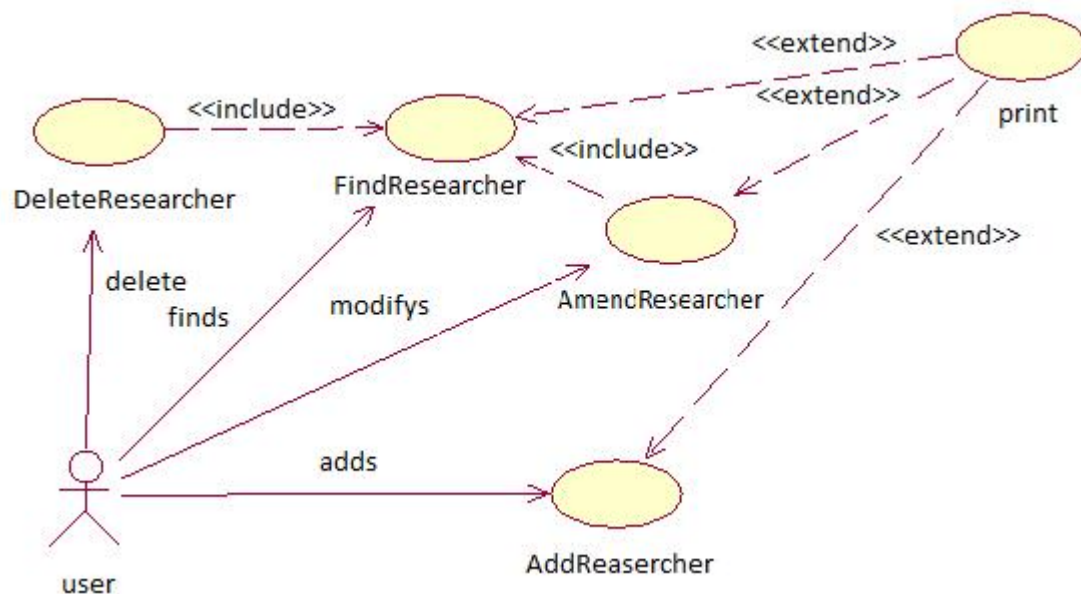
Οι επικοινωνιακές σχέσεις (communicates relationship) μεταξύ ενός actor και ενός use case δείχνει ότι ο actor αρχικοποιεί το use case.

Η includes σχέση προτείνει ότι ένα use case πρέπει να περιλαμβάνει ένα άλλο. Με άλλα λόγια όταν τρέχει ένα use case πρέπει και το άλλο να τρέχει ταυτόχρονα. Ένα Use case μπορεί να περιλαμβάνεται από ένα ή περισσότερα use cases.

Η extends σχέση χρησιμοποιείται όταν ένα use case προαιρετικά επεκτείνει τη λειτουργικότητα που παρέχεται από άλλο use case. Με άλλα λόγια αν τρέχει ένα use case, ένα extending use case μπορεί να τρέχει αλλά μπορεί και να μην τρέχει.

Πιο κάτω παραθέτονται τα διαγράμματα Use Case μαζί με τη περιγραφή του κάθε διαγράμματος

3.2.1.1 Διαχείριση Ερευνητή



Λειτουργίες:

Add Researcher: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να αποθηκεύσει στην βάση τα στοιχεία ενός καινούργιου ερευνητή.

Amend Researcher: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να επεξεργαστεί οποιοδήποτε χαρακτηριστικό ενός ερευνητή ο οποίος βρίσκεται στην βάση. Για να επεξεργαστεί τα στοιχεία κάποιου ερευνητή θα πρέπει να προηγηθεί η αναζήτησή του με βάση το όνομα ή την ταυτότητα του.

Delete Researcher: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να διαγράψει από την βάση ένα ερευνητή. Για να γίνει η διαγραφή του ερευνητή θα πρέπει προηγηθεί η αναζήτηση του με βάση το όνομα ή την ταυτότητα του.

Print: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να εκτυπώσει τα στοιχεία ενός ερευνητή

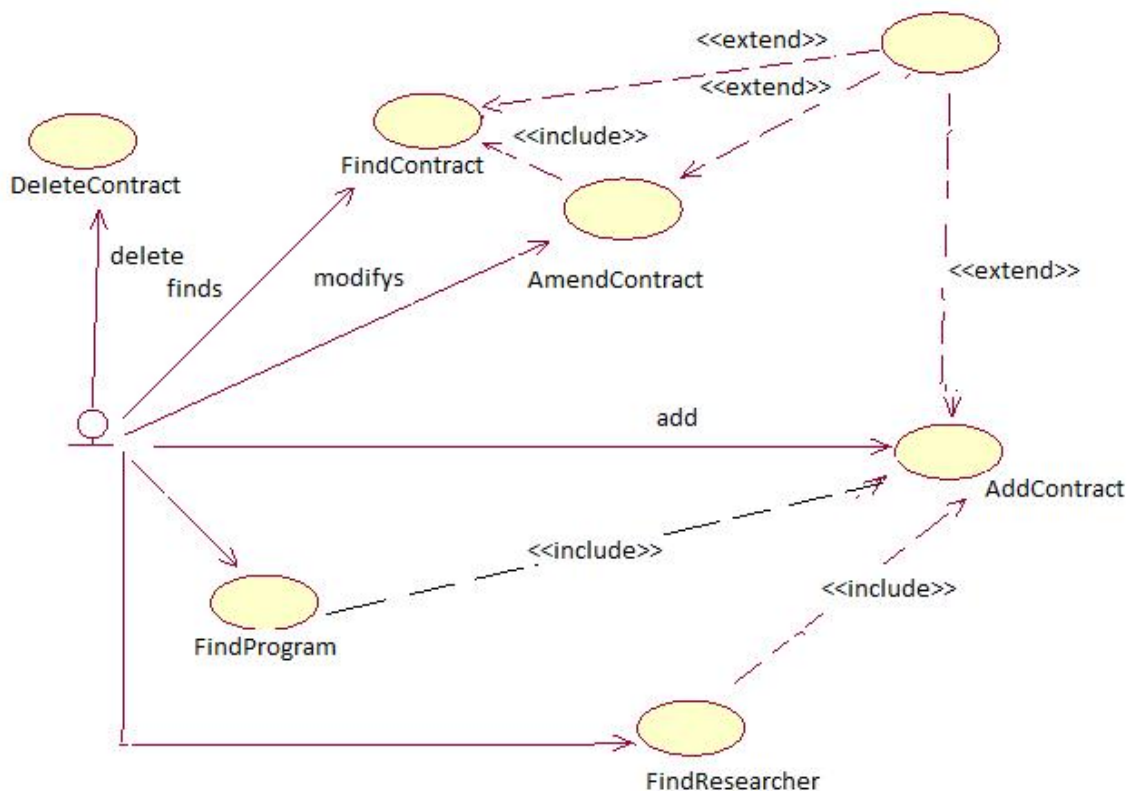
Find Researcher: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να αναζητήσει έναν ερευνητή με βάση κάποιο ή κάποια χαρακτηριστικά.

Εξαρτήσεις μεταξύ λειτουργιών:

Εξάρτηση <<extend>> μεταξύ των λειτουργιών Add new Researcher – Print, Amend Researcher - Print και Find Researcher - Print : Όταν ο χρήστης εκτελέσει τη λειτουργία Add new Researcher ή Amend Researcher ή Find Researcher θα μπορεί αν θέλει και να εκτελέσει τη λειτουργία Print.

Εξάρτηση <<include>> μεταξύ των λειτουργιών Amend Researcher - Find Researcher και Delete Researcher - Find Researcher: Ο χρήστης για να εκτελέσει τη λειτουργία Amend Researcher ή Delete Researcher θα πρέπει πρώτα να εκτελέσει τη λειτουργία Find Researcher

3.2.1.2 Διαχείριση Συμβολαίου



Λειτουργίες:

Add Contract: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να αποθηκεύσει στην βάση τα στοιχεία ενός νέου συμβολαίου.

Amend Contract: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να επεξεργαστεί οποιοδήποτε χαρακτηριστικό ενός συμβολαίου το οποίο βρίσκεται στην βάση. Για να επεξεργαστεί τα στοιχεία κάποιου συμβολαίου θα πρέπει να προηγηθεί η αναζήτησή του.

Delete Contract: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να διαγράψει από την βάση ένα συμβόλαιο. Για να γίνει η διαγραφή συμβολαίου θα πρέπει προηγηθεί η αναζήτηση του με βάση το πρόγραμμα.

Find Contract: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να αναζητήσει ένα συμβόλαιο με βάση το πρόγραμμα..

Print: Μέσω αυτής της λειτουργίας ο χρήστης θα μπορεί να εκτυπώσει τα στοιχεία ενός συμβολαίου.

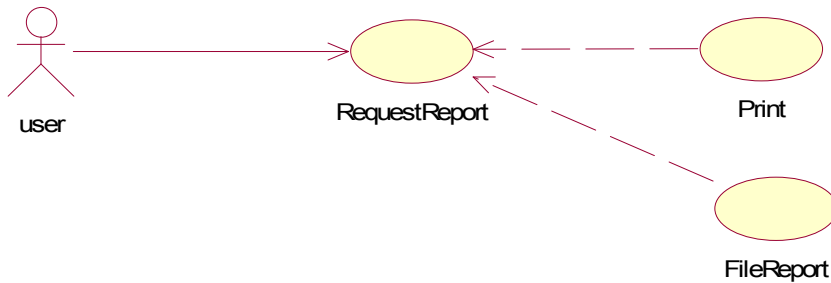
Εξαρτήσεις μεταξύ λειτουργιών:

Εξάρτηση <<extend>> μεταξύ των λειτουργιών Add Contract – Print, Amend Contract - Print και Find Contract - Print : Όταν ο χρήστης εκτελέσει τη λειτουργία Add Contract ή Amend Contract ή Find Contract θα μπορεί αν θέλει και να εκτελέσει τη λειτουργία Print.

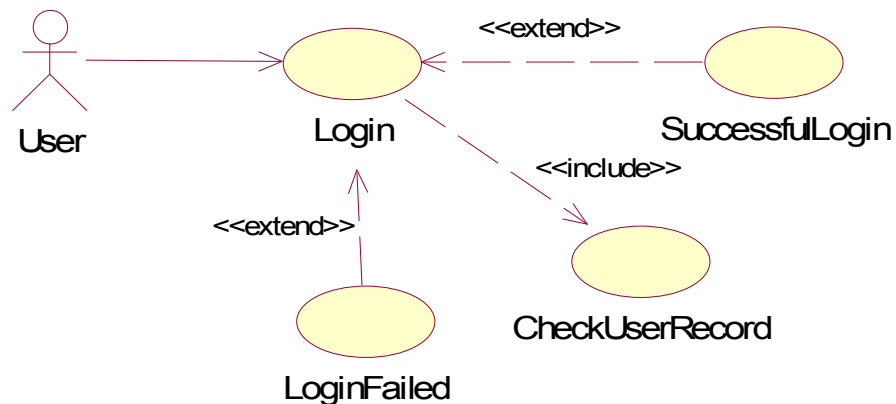
Εξάρτηση <<include>> μεταξύ των λειτουργιών Amend Contract - Find Contract και Delete Contract - Find Contract: Ο χρήστης για να εκτελέσει τη λειτουργία Amend Contract ή Delete Contract θα πρέπει πρώτα να εκτελέσει τη λειτουργία Find Contract.

Εξάρτηση <<include>> μεταξύ των λειτουργιών Find Researcher - Add Contract και Find Program - Add Contract: Ο χρήστης για να εκτελέσει τη λειτουργία Add Contract θα πρέπει πρώτα να εκτελέσει τη λειτουργία Find Researcher και Find Program για να εισάγει νέο συμβόλαιο.

3.2.1.3 Διαχείριση Ερευνητή



3.2.1.4 Διαχείριση Ερευνητή



3.2.2 Class Diagram

Στο class diagram που έχω σχεδιάσει φαίνονται όλες οι συσχετίσεις μεταξύ των κλάσεων που θα λαμβάνουν μέρος στο σύστημα που θα έχω υλοποιήσει. Όλες οι κλάσεις αποτελούν τα αντικείμενα που περικλείουν τις μεθόδους που θα έχω να υλοποιήσω για την λειτουργία του συστήματός μου. Οι συσχετίσεις μεταξύ των κλάσεων περιγράφονται από τις ροές μεταξύ του διαγράμματος που αναπαριστούν τις λειτουργίες και τις ενέργειες που διεκπεραιώνονται μεταξύ των αντικειμένων.

Μέσα από την κάθε κλάση βλέπουμε επίσης και τα στοιχεία που την αποτελούν. Το class Diagram αποτελεί μια εικόνα της βάσης δεδομένων όσο αφορά τις πληροφορίες που θα αποθηκεύονται σε αυτή καθώς και τις μεθόδους που θα υπόκεινται τα δεδομένα.

Κλάση Υπεύθυνου Ερευνητή:

Μεταβλητές:

Pcode κωδικός στο σύστημα
name
surname
fathername
birthdate
sex
maritalstatus
idnum
socialinsnum
telnum
postcode
town
country
email
int1
char1
address

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός υπεύθυνου ερευνητή και το εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(int pcode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του υπεύθυνου ερευνητή και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός υπεύθυνου ερευνητή και για τον αντίστοιχο υπεύθυνο ερευνητή ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(int pcode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του υπεύθυνου ερευνητή και επιστρέφει τα χαρακτηριστικά του αντίστοιχου υπεύθυνου ερευνητή που βρίσκονται στην βάση δεδομένων.

void maxCode()

: συνάρτηση που επιστρέφει το μέγιστο αριθμό κωδικού υπεύθυνου ερευνητή που βρίσκεται στη βάση δεδομένων

Κλάση Προγράμματος:

Μεταβλητές:

Prcode : κωδικός συστήματος
name
shortname
contactnum
manager
type
typedetail
duration
tcost
dcost
subcontracting
personel
icost
startdate
enddate
int1
char1
openclose

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός προγράμματος και το εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(int prcode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του προγράμματος και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός προγράμματος και για το αντίστοιχο πρόγραμμα ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(int prcode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του προγράμματος και επιστρέφει τα χαρακτηριστικά του αντίστοιχου προγράμματος που βρίσκονται στην βάση δεδομένων.

void maxCode()

: συνάρτηση που επιστρέφει το μέγιστο αριθμό κωδικού προγράμματος που βρίσκεται στη βάση δεδομένων

Κλάση Υπεύθυνος Ερευνητής:

Μεταβλητές:

Pcode κωδικός στο σύστημα
name
surname
fathername
birthdate
sex
maritalstatus
idnum
socialinsnum
telnum
address
postcode
town
country
email
int1
char1

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός υπεύθυνου ερευνητή και τον εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(int pcode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του υπεύθυνου ερευνητή και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός υπεύθυνου ερευνητή και για τον αντίστοιχο υπεύθυνο ερευνητή ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(int pcode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του υπεύθυνου ερευνητή και επιστρέφει τα χαρακτηριστικά του αντίστοιχου υπεύθυνου ερευνητή που βρίσκονται στην βάση δεδομένων.

void maxCode()

: συνάρτηση που επιστρέφει το μέγιστο αριθμό κωδικού υπεύθυνου ερευνητή που βρίσκεται στη βάση δεδομένων

Κλάση Συμβολαίου:

Μεταβλητές:

Ccode κωδικός στο σύστημα

scode

prcode

startdate

enddate

salary

rate

int3

int2

char1

researcher

program

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός συμβολαίου και το εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(int ccode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του συμβολαίου και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός συμβολαίου και για το αντίστοιχο σύμβολο ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(int ccode)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του συμβολαίου και επιστρέφει τα χαρακτηριστικά του αντίστοιχου συμβολαίου που βρίσκονται στην βάση δεδομένων.

void maxCode()

: συνάρτηση που επιστρέφει το μέγιστο αριθμό κωδικού συμβολαίου που βρίσκεται στη βάση δεδομένων

Κλάση Χρήστη:

Μεταβλητές:

Username

Password

Type

type

code

Usertype

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός χρήστη και τον εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(int username)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του χρήστη και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός χρήστη και για τον αντίστοιχο χρήστη ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(int username)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του χρήστη και επιστρέφει τα χαρακτηριστικά του αντίστοιχου χρήστη που βρίσκονται στην βάση δεδομένων.

Κλάση Τύπος Προγράμματος:

Μεταβλητές:

typeid
name

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός τύπου προγράμματος και τον εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(username, password)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του τύπου προγράμματος και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός τύπου προγράμματος και για τον αντίστοιχο τύπου χρήστη ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(username, password)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του τύπου προγράμματος και επιστρέφει τα χαρακτηριστικά του αντίστοιχου τύπου χρήστη που βρίσκονται στην βάση δεδομένων.

void maxCode()

: συνάρτηση που επιστρέφει το μέγιστο αριθμό κωδικού τύπου προγράμματος που βρίσκεται στη βάση δεδομένων

Κλάση Timesheet:

Μεταβλητές:

tcode
ccode
year
month
total
Day1
Day2
Day3
Day4
Day5
Day6
Day7
Day8
Day9
Day10
Day11
Day13
Day14
Day15
Day16
.....
Day30
Day31

Μέθοδοι:

void add(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός timesheet και τον εισάγει στον αντίστοιχο πίνακα της βάσης

void delete(int username)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του timesheet και τον διαγράφει από τον αντίστοιχο πίνακα της βάσης

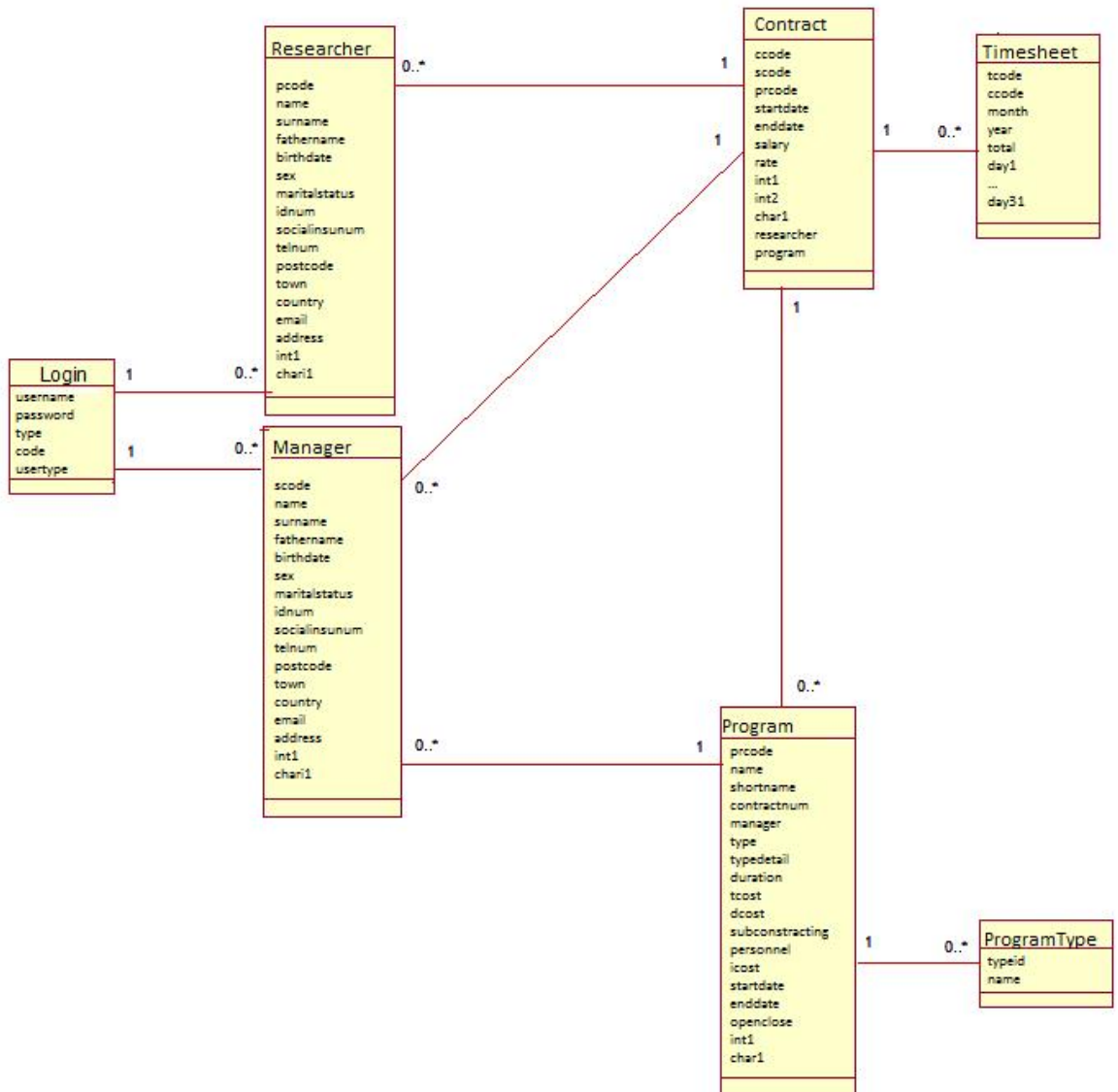
void update(...)

: συνάρτηση που παίρνει σαν παραμέτρους τα χαρακτηριστικά ενός timesheet και για τον αντίστοιχο timesheet ενημερώνεται η βάση δεδομένων με τα χαρακτηριστικά τα οποία δεν είναι κενά

void find(int username)

: συνάρτηση που παίρνει σαν παράμετρο τον κωδικό αριθμό του timesheet και επιστρέφει τα χαρακτηριστικά του αντίστοιχου timesheet που βρίσκονται στην βάση δεδομένων.

Class Diagram:



Περιγραφή των λειτουργιών του συστήματος βάση του class diagram

Στο διάγραμμα που κατασκευάσα περιγράφονται όλες οι συσχετίσεις μεταξύ των κλάσεων. Βλέπουμε τον υπεύθυνο ερευνητή και τον ερευνητή να επικοινωνεί με τον χρήστη του συστήματος γιατί κάθε ένας από αυτούς θα διαθέτει κωδικούς εισόδου στο σύστημα μας. Κάθε ερευνητής και υπεύθυνος θα μπορεί να συμπληρώνει το timesheet του για κάθε πρόγραμμα που εργάζεται. Για να γίνει αυτό θα πρέπει ο λειτουργός του συστήματος να δημιουργήσει συμβόλαιο μεταξύ του προγράμματος και του ερευνητή. Ο λόγος που δεν υπάρχει απευθείας σύνδεση μεταξύ ερευνητών και timesheet ή υπεύθυνων ερευνητών και timesheet είναι γιατί το timesheet παίρνει τα δεδομένα από την κλάση συμβόλαιο. Για ένα συμβόλαιο μπορεί να δημιουργηθούν πολλά timesheet, κάθε timesheet γίνεται για κάθε μήνα κάθε συμβολαίου. Η κλάση συμβόλαιο είναι ενωμένη με τις κλάσεις ερευνητής και υπεύθυνος ερευνητής γιατί κάθε συμβόλαιο πρέπει να έχει ένα ερευνητή ή υπεύθυνο ερευνητή που να γίνεται γι' αυτό το συμβόλαιο. Επίσης ένα συμβόλαιο γίνεται για κάποιο πρόγραμμά γι' αυτό και υπάρχει και η ένωση μεταξύ των δύο κλάσεων. Όσο αφορά την κλάση πρόγραμμα είναι ενωμένη με την κλάση τύπος προγράμματος γιατί κάθε πρόγραμμα ανήκει σε ένα τύπο προγράμματος.

3.3 Software Project Management Plan

3.3.1 Εισαγωγή

Το μοντέλο ανάπτυξης και κύκλου ζωής που χρησιμοποίησα για την υλοποίηση του συστήματος είναι το μοντέλου του καταρράκτη (waterfall model). Το μοντέλο του καταρράκτη χρησιμοποιήθηκε λόγω του ότι είναι μια πειθαρχημένη προσέγγιση με συνεχή έλεγχο και τεκμηρίωση και παρέχει ευκολότερη συντήρηση. Επιπλέον το γεγονός ότι κάθε μία από τις αρχικές φάσεις ανάπτυξης πρέπει να επαληθευτεί πριν την έναρξη της επόμενης, προσφέρει πιο σίγουρες και σωστές βάσεις για τα μετέπειτα στάδια ανάπτυξης του προϊόντος. Ακολουθώντας, ο συνεχής έλεγχος δίνει τη δυνατότητα τόσο για σταθερή σύγκριση συνέπειας μεταξύ της παρούσης και των προηγούμενων φάσεων, όσο και σιγουριά ότι ξεκινά κάθε νέα φάση χωρίς προβλήματα να προωθούνται σε αυτή από προηγούμενες φάσεις.

Με την **επικύρωση** (*validation*) επαληθεύεται ότι έχουν υλοποιηθεί όλες οι προδιαγραφές και το λογισμικό ανταποκρίνεται στις απαιτήσεις από το σύστημα δηλαδή, «το λογισμικό κάνει αυτό που ήθελε ο πελάτης να κάνει»

Με την **επαλήθευση** (*verification*) επαληθεύεται ότι κάθε λειτουργία/μονάδα του συστήματος εκτελείται σωστά από προγραμματιστικής άποψης, «το λογισμικό κάνει σωστά αυτό που ήθελε ο πελάτης να κάνει»

Ανάλυση απαιτήσεων: Σε αυτή τη φάση πρέπει να κατανοηθεί τι χρειάζεται να κάνω.

Καθορισμός Προδιαγραφών: Σε αυτή τη φάση γίνεται ο καθορισμός των εργασιών που θα επιτελεί το λογισμικό, καθώς και των περιορισμών που ισχύουν. Οι προδιαγραφές δεν πρέπει να είναι διφορούμενες, ημιτελείς ή αντιφατικές. Οτιδήποτε αναγράφεται στο έγγραφο αυτό πρέπει να είναι σε πλήρη συνέπεια με τα όσα αναφέρονται στο έγγραφο απαιτήσεων.

Σχεδιασμός: Σε αυτή τη φάση γίνεται ο καθορισμός των δομικών ενοτήτων του συστήματος και της μεταξύ τους επικοινωνίας. Επίσης γίνεται η επιλογή των αλγορίθμων, των δομών δεδομένων και των ροών δεδομένων. Είναι ουσιαστικά το στάδιο το οποίο μεταφράζει το τι πρέπει να γίνει από την προηγούμενη φάση στο πώς αυτό θα επιτευχθεί.

Υλοποίηση: Σε αυτή τη φάση γίνεται η υλοποίηση του αναλυτικού σχεδιασμού δηλαδή η γραφή του κώδικα. Βασική προϋπόθεση για επιτυχία στη φάση αυτή είναι η επιλογή της καταλληλότερης γλώσσας σε σχέση με το προϊόν και το άτομο που πρόκειται να το αναπτύξει. Στη συνέχεια είναι αναγκαίο να ακολουθηθούν πρακτικές καλού προγραμματισμού, όπως αυτοπεπεξηγηματικά ονόματα σε μεταβλητές και σταθερές, πράγμα που καθιστά πιο εύκολη τη μελλοντική συντήρηση. Εξάλλου, σημαντικό είναι να διατηρηθεί η υψηλή συνεκτικότητα εντός των modules και χαμηλή η σύζευξη μεταξύ τους, δηλαδή ο βαθμός επικοινωνίας μεταξύ αυτών. Επίσης γίνεται η τεκμηρίωση δηλαδή η προσθήκη σχολίων στον κώδικα, η ανάλυση περιπτώσεων, τα αποτελέσματα καθώς και ο έλεγχος της φάσης αυτής. Ο έλεγχος του κώδικα πρέπει να είναι εξαντλητικός όσο αφορά όλα τα μονοπάτια του κώδικα.

Συνένωση: Σε αυτή τη φάση αυτή γίνεται η συνένωση των modules που υλοποιούνται στην προηγούμενη φάση και γίνεται έλεγχος αν το προϊόν λειτουργεί σωστά ως μια ενιαία οντότητα. Η φάση αυτή μπορεί να γίνει παράλληλα με την υλοποίηση έτσι ώστε να απομονωθούν τα λάθη και να βρίσκονται πιο νωρίς με σκοπό τη γρήγορη διόρθωσή τους. Από την άλλη όμως μπορεί να επιλεγεί η τακτική της ολοκλήρωσης της υλοποίησης όλων των modules και μετά να ξεκινήσει η συνένωσή τους σαν μια ενιαία οντότητα.

Κεφάλαιο 4

Σχεδιασμός

4.1 Αντικειμενοστρεφής Σχεδίαση	35
4.2 Κατασκευή αναλυτικών Class Diagram	46
4.3 Σχεδίαση προϊόντος μέσω “Clients Of Objects”	46
4.4 Πίνακες Βάσης Δεδομένων	47
4.5 Διαγράμματος Σχέσεων Οντοτήτων	50

4.1 Αντικειμενοστρεφής Σχεδίαση

4.1.1 Κατασκευή Interaction Diagrams για κάθε σενάριο

4.1.1.1 Εισαγωγή στο σύστημα

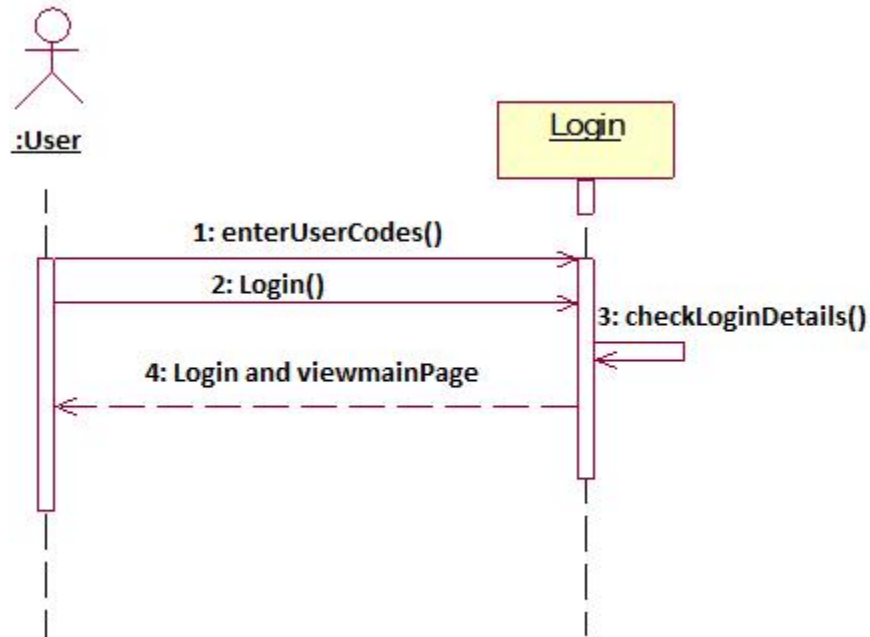
4.1.1.1.1 Σενάριο normal

- Εμφάνιση οθόνης στον χρήστη για είσοδό του στο σύστημα.
- Εισαγωγή του user name
- Εισαγωγή του password.
- Έλεγχος από το σύστημα εάν είναι σωστό το user name και το password που έχει εισαχθεί από τον χρήστη.
- Εφόσον είναι σωστά και τα δύο, τότε γίνεται επιτρεπτή η είσοδος του στο σύστημα

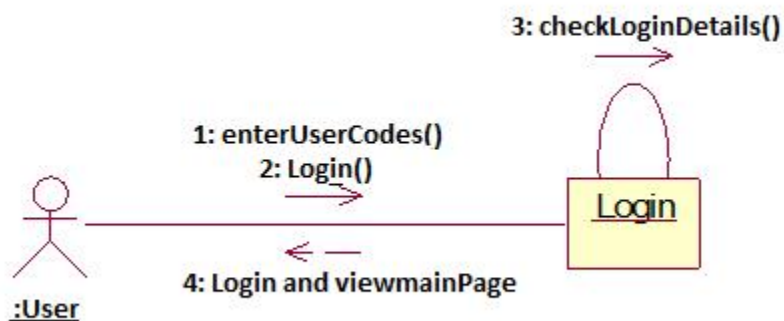
4.1.1.1.2 Σενάριο abnormal

- Εμφάνιση οθόνης στον χρήστη για είσοδό του στο σύστημα.
- Εισαγωγή του user name
- Εισαγωγή του password.
- Έλεγχος από το σύστημα εάν είναι σωστό το user name και το password που έχει εισαχθεί από τον χρήστη.

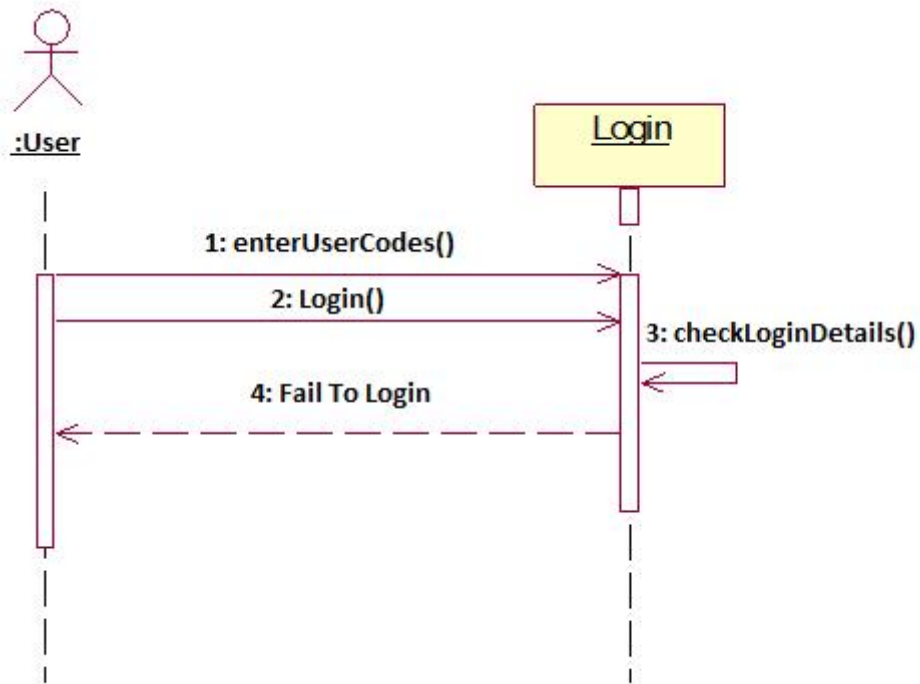
- Εμφάνιση μηνύματος στον χρήστη για λάθος εισαγωγή user name και password και έτσι δεν γίνεται επιτρεπτή η είσοδος του χρήστη στο σύστημα.



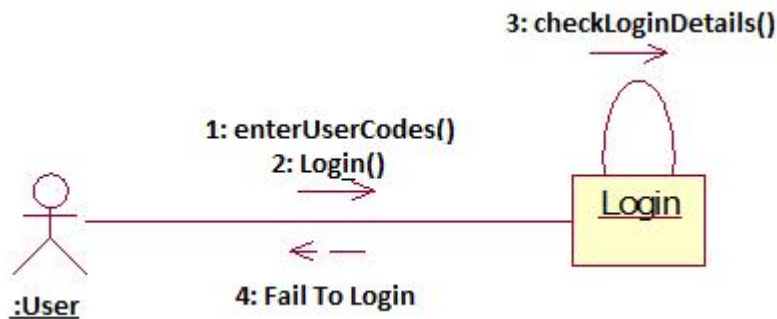
Εικόνα Sequence Diagram εισαγωγής χρήστη (normal σενάριο)



Εικόνα Collaboration Diagram εισαγωγής χρήστη (normal σενάριο)



Εικόνα Sequence Diagram εισαγωγής χρήστη (abnormal σενάριο)



Εικόνα Collaboration Diagram εισαγωγής χρήστη (abnormal σενάριο)

4.1.1.2 Προσθήκη νέου Ασθενή

4.1.1.2.1 Σενάριο normal

Εφόσον επιτραπεί η είσοδος του χρήστη στο σύστημα μπορεί να εκτελέσει διάφορες λειτουργίες. Μια από αυτές είναι η διαχείριση ερευνητή.

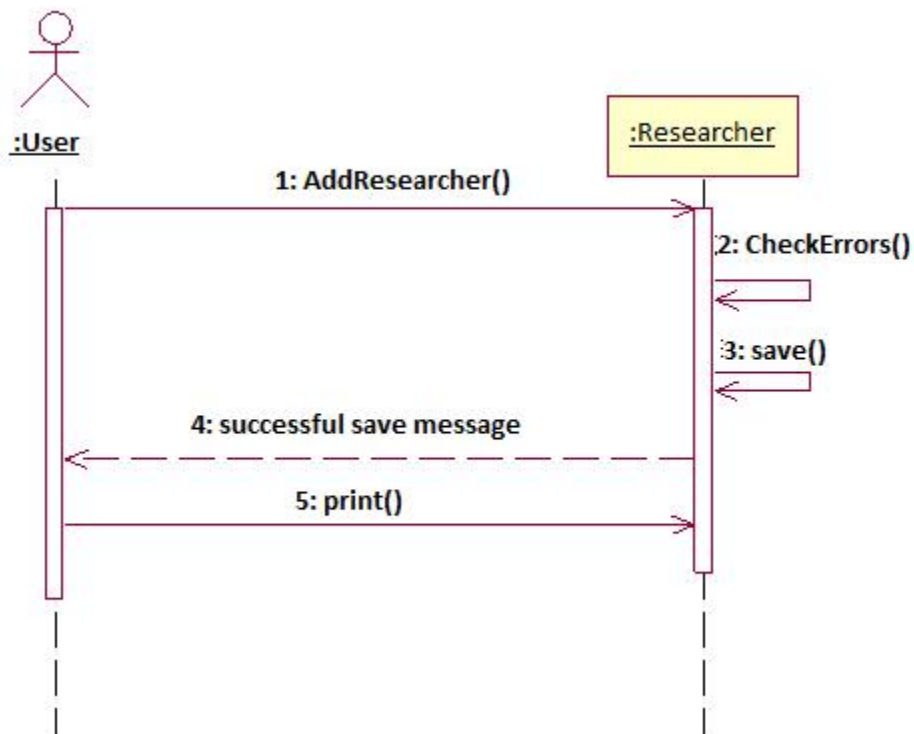
- Επιλογή προσθήκης ερευνητή
- Καταχώρηση στοιχείων ερευνητή
- Πάτημα κουμπιού για αποθήκευση

- Έλεγχος για ορθότητα των στοιχείων εισαγωγής
- Αποθήκευση του καινούργιου ερευνητή στη βάση.
- Μήνυμα επιτυχούς καταχώρησης

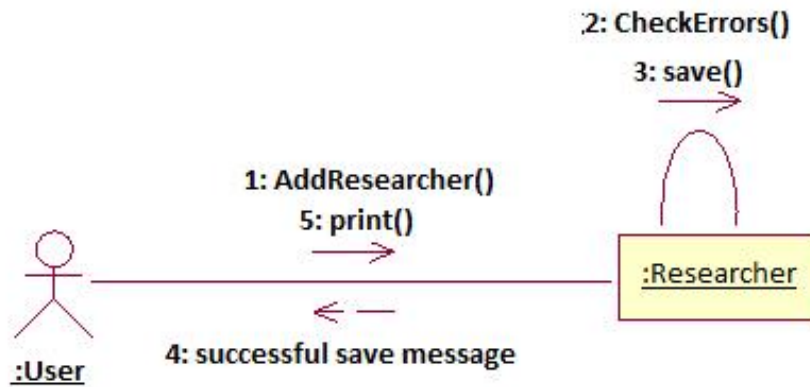
4.1.1.2.2 Σενάριο **abnormal**:

Εφόσον επιτραπεί η είσοδος του χρήστη στο σύστημα μπορεί να εκτελέσει διάφορες λειτουργίες. Μια από αυτές είναι η διαχείριση ερευνητή.

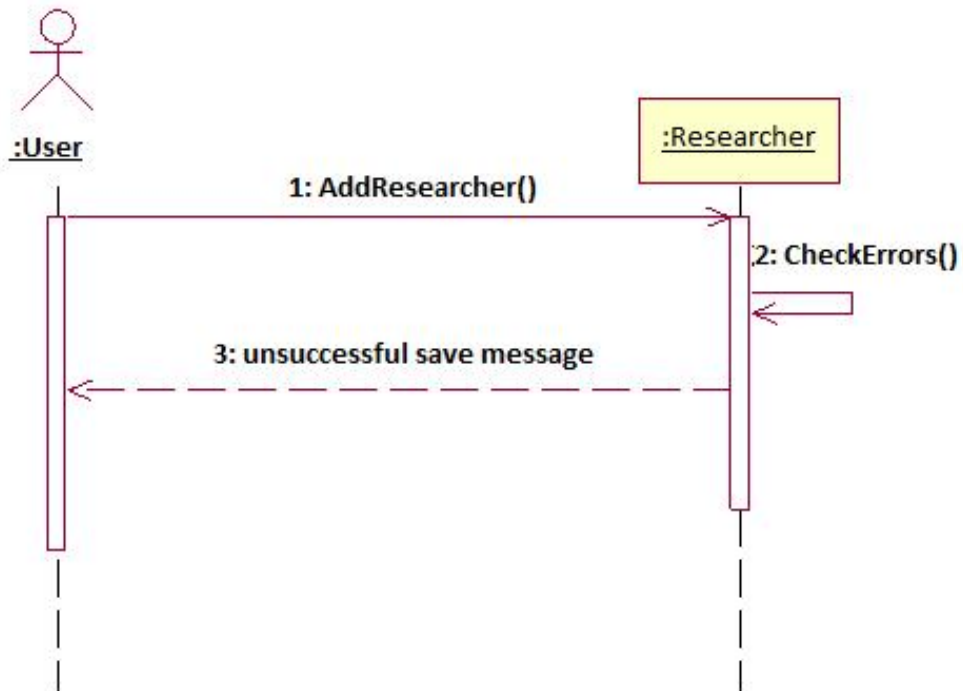
- Επιλογή προσθήκης ερευνητή
- Καταχώρηση στοιχείων ερευνητή
- Πάτημα κουμπιού για αποθήκευση
- Έλεγχος για ορθότητα των στοιχείων εισαγωγής
- Η αποθήκευση δεν γίνεται δεκτή είτε γιατί σε κάποιο πεδίο της φόρμας έχει εισαχθεί λάθος στοιχείο ερευνητή (για παράδειγμα στο πεδίο «τηλέφωνο ερευνητή» έχουν εισαχθεί και γράμματα αντί μόνο αριθμοί) είτε γιατί κάποιο πεδίο της φόρμας έχει παραμείνει κενό κάτι το οποίο μπορεί να μην επιτρεπόταν να γίνει
- Μήνυμα ανεπιτυχούς καταχώρησης



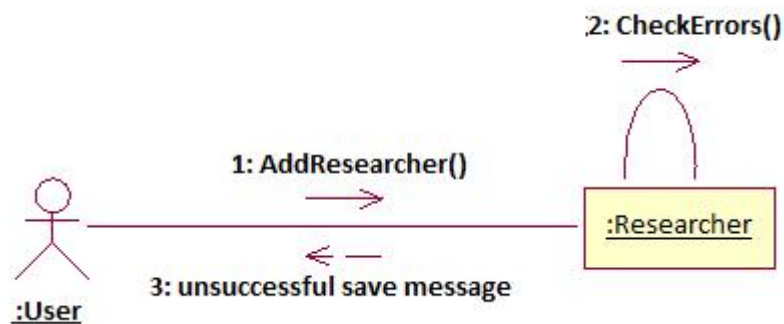
Εικόνα Sequence Diagram προσθήκης ερευνητή (normal σενάριο)



Εικόνα Collaboration Diagram προσθήκης ερευνητή (normal σενάριο)



Εικόνα Sequence Diagram προσθήκης ερευνητή (abnormal σενάριο)



Εικόνα Collaboration Diagram προσθήκης ερευνητή (abnormal σενάριο)

4.1.1.3 Επεξεργασία στοιχείων Ερευνητή

4.1.1.3.1 Σενάριο normal:

Εφόσον επιτραπεί η είσοδος του χρήστη στο σύστημα μπορεί να εκτελέσει διάφορες λειτουργίες. Μια από αυτές είναι η διαχείριση ασθενή.

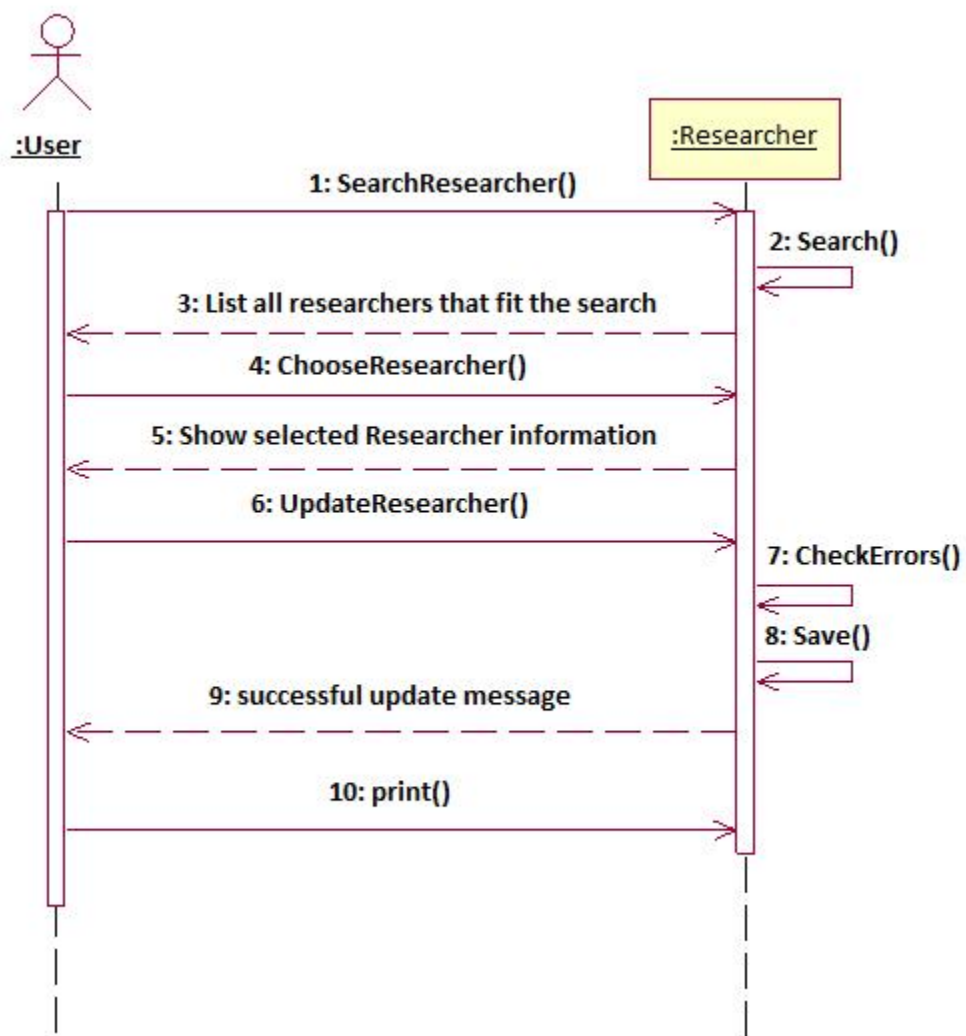
- Επιλογή επεξεργασίας ερευνητή
- Εύρεση πελάτη με βάση κάποιας λίστας χαρακτηριστικών (π.χ. όνομα ερευνητή)
- Εμφάνιση του συνόλου των ερευνητή που είναι καταχωρημένοι στη βάση και πληρούν τα κριτήρια αναζήτησης
- Επιλογή του ερευνητή για τον οποίο εκτελείται η λειτουργία της επεξεργασίας πεδία της φόρμας
- Επεξεργασία οποιουδήποτε χαρακτηριστικού του ερευνητή
- Αποθήκευση των καινούργιων χαρακτηριστικών του ερευνητή στη βάση.
- Μήνυμα επιτυχούς καταχώρησης

4.1.1.2.2 Σενάριο abnormal:

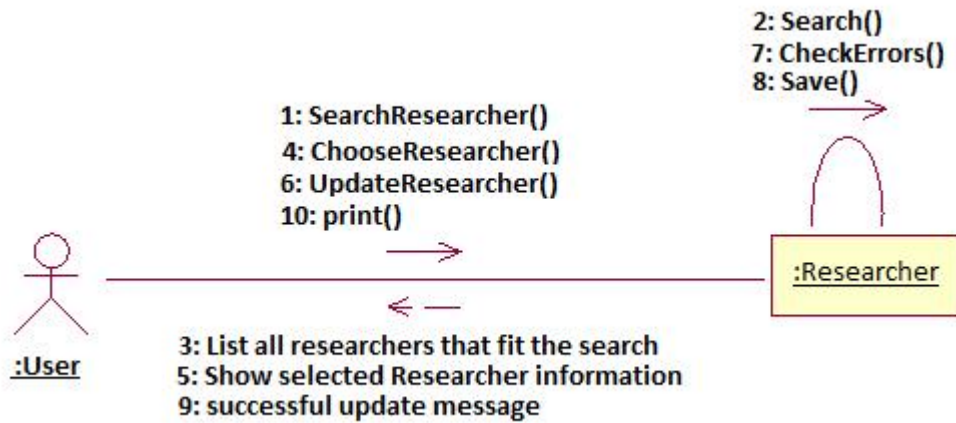
Εφόσον επιτραπεί η είσοδος του χρήστη στο σύστημα μπορεί να εκτελέσει διάφορες λειτουργίες. Μια από αυτές είναι η διαχείριση ερευνητή.

- Επιλογή επεξεργασίας ερευνητή
- Εύρεση ερευνητή με βάση κάποιας λίστας χαρακτηριστικών (π.χ. όνομα ερευνητή)
- Εμφάνιση του συνόλου των ερευνητών που είναι καταχωρημένοι στη βάση και πληρούν τα κριτήρια αναζήτησης

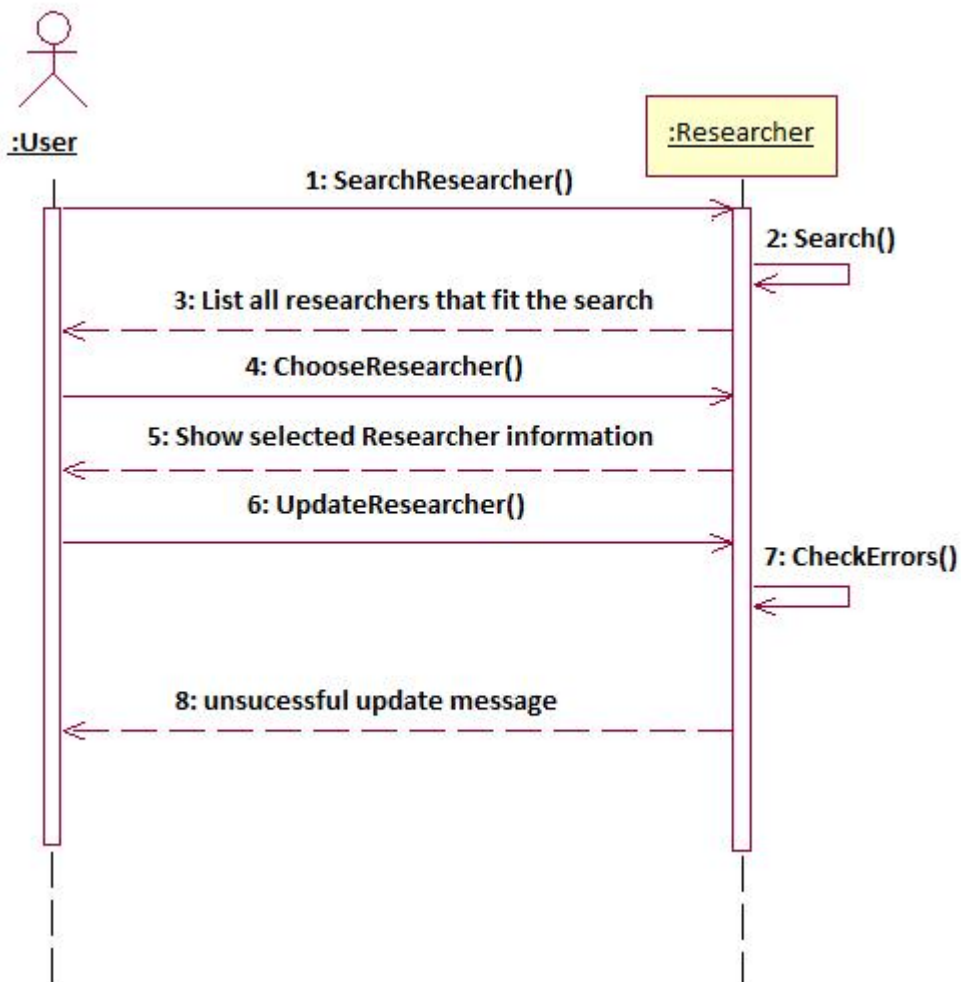
- Εμφάνιση των χαρακτηριστικών του συγκεκριμένου ερευνητή στα αντίστοιχα
Επιλογή του ερευνητή για τον οποίο εκτελείται η λειτουργία της επεξεργασίας
- Εμφάνιση των χαρακτηριστικών του συγκεκριμένου ερευνητή στα αντίστοιχα
πεδία της φόρμας
- Επεξεργασία οποιουδήποτε χαρακτηριστικού του ερευνητή
- Η αποθήκευση δεν γίνεται δεκτή είτε γιατί σε κάποιο πεδίο της φόρμας έχει
εισαχθεί λάθος στοιχείο ερευνητή είτε γιατί κάποιο πεδίο της φόρμας έχει
παραμείνει κενό κάτι το οποίο μπορεί να μην επιτρεπόταν να γίνει
- Μήνυμα ανεπιτυχούς καταχώρησης



Εικόνα Sequence Diagram επεξεργασίας ερευνητή (normal σενάριο)



Εικόνα Collaboration Diagram επεξεργασίας ερευνητή (normal σενάριο)



Εικόνα Sequence Diagram επεξεργασίας ερευνητή (abnormal σενάριο)



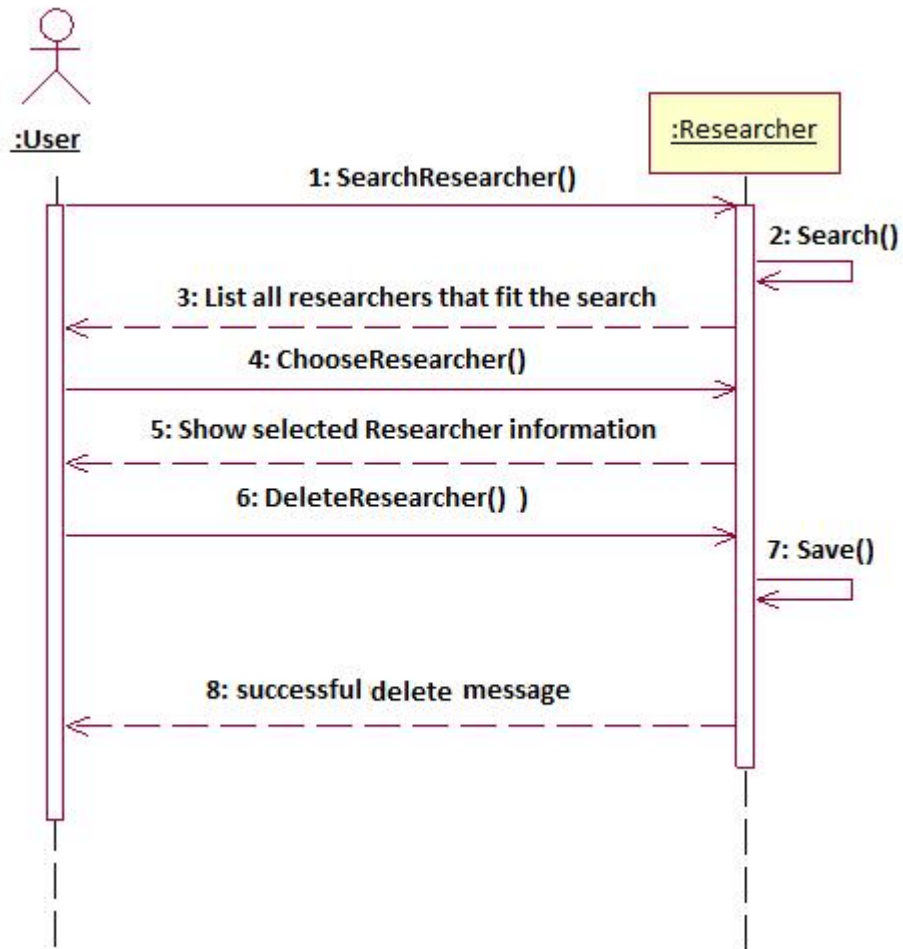
Εικόνα Collaboration Diagram επεξεργασίας ερευνητή (abnormal σενάριο)

4.1.1.4 Διαγραφή ερευνητή

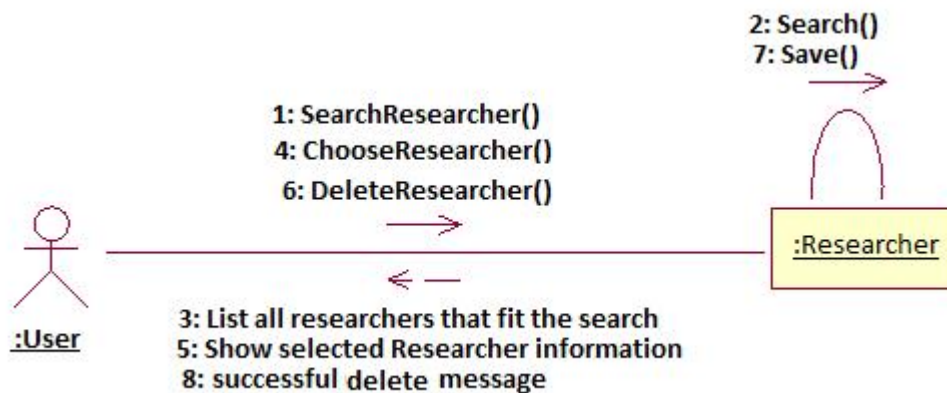
4.1.1.4.1 Σενάριο normal

Εφόσον επιτραπεί η είσοδος του χρήστη στο σύστημα μπορεί να εκτελέσει διάφορες λειτουργίες. Μια από αυτές είναι η διαχείριση ερευνητή.

- Επιλογή διαγραφής ερευνητή
- Εύρεση ερευνητή με βάση κάποιας λίστας χαρακτηριστικών (π.χ. τηλέφωνο ασθενή)
- Εμφάνιση του συνόλου των ερευνητών που είναι καταχωρημένοι στη βάση και πληρούν τα κριτήρια αναζήτησης
- Επιλογή του ερευνητή για τον οποίο εκτελείται η λειτουργία της διαγραφής
- Εμφάνιση των χαρακτηριστικών του συγκεκριμένου ερευνητή στα αντίστοιχα πεδία της φόρμας
- Διαγραφή ερευνητή
- Εμφάνιση μηνύματος επιτυχούς διαγραφής



Εικόνα Sequence Diagram διαγραφής ερευνητή (normal σενάριο)



Εικόνα Sequence Diagram διαγραφής ερευνητή (normal σενάριο)

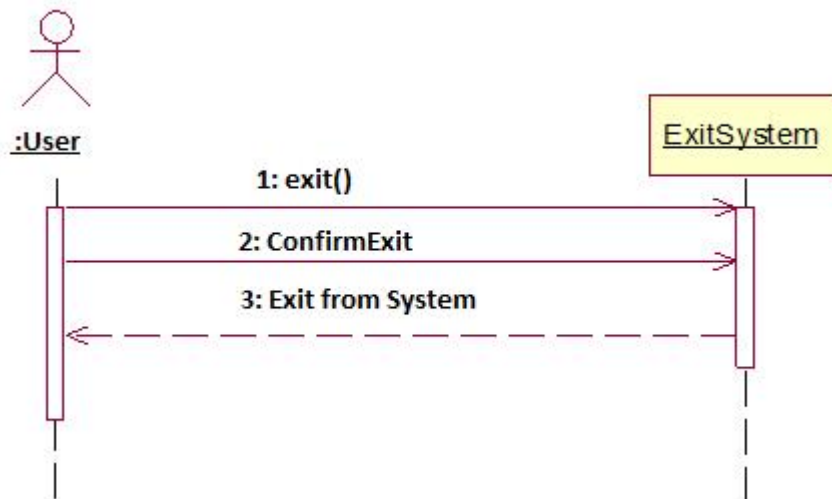
4.1.1.5 Έξοδος από το σύστημα

4.1.1.5.1 Σενάριο normal:

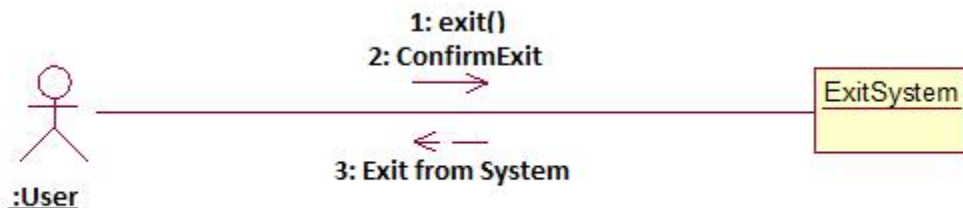
- Επιλογή του χρήστη για έξοδό του από το σύστημα
- Εμφάνιση μηνύματος για επιβεβαίωση της εξόδου του από το σύστημα
- Επιβεβαίωση της εξόδου του από το σύστημα
- Έξοδος από το σύστημα

4.1.1.5.2 Σενάριο abnormal:

- Επιλογή του χρήστη για έξοδό του από το σύστημα
- Εμφάνιση μηνύματος για επιβεβαίωση της εξόδου του από το σύστημα
- Ακύρωση της εξόδου από το σύστημα
- Επιστροφή στο σύστημα



Εικόνα Sequence Diagram εξόδου από το σύστημα (normal σενάριο)



Εικόνα Sequence Diagram εξόδου από το σύστημα (normal σενάριο)

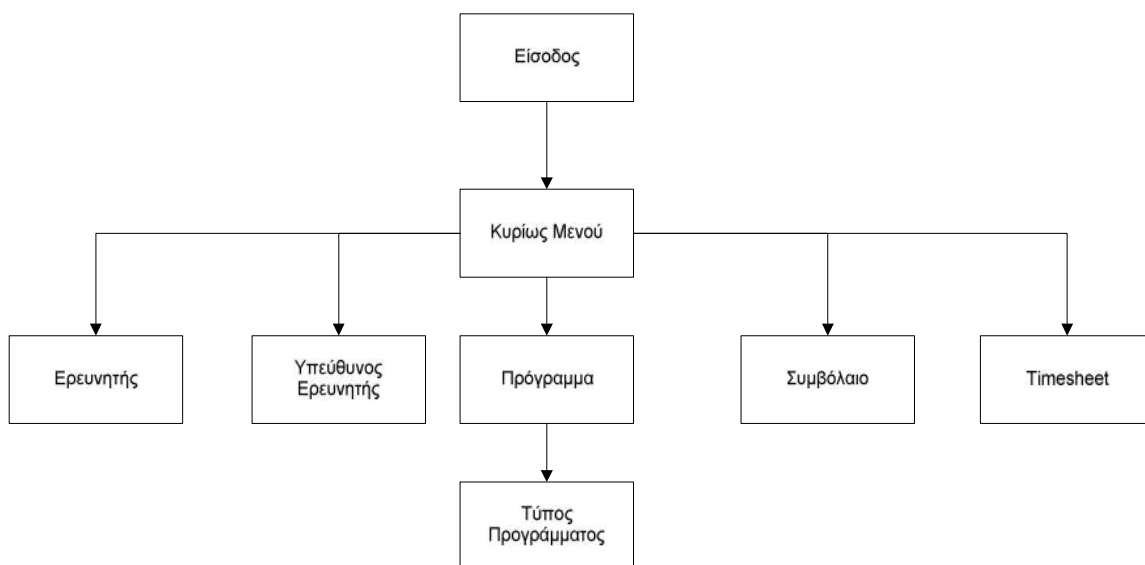
4.2 Κατασκευή αναλυτικών Class Diagram

Στο class diagram που έχω σχεδιάσει φαίνονται όλες οι συσχετίσεις μεταξύ των κλάσεων που θα λαμβάνουν μέρος στο σύστημα που έχω να υλοποιήσω. Όλες οι κλάσεις αποτελούν τα αντικείμενα που περικλείουν τις μεθόδους που θα έχω να υλοποιήσω για την λειτουργία του συστήματος μου. Οι συσχετίσεις μεταξύ των κλάσεων περιγράφονται από τις ροές μεταξύ του διαγράμματος που αναπαριστούν τις λειτουργίες και τις ενέργειες που διεκπεραιώνονται μεταξύ των αντικειμένων.

4.3 Σχεδίαση προϊόντος μέσω “Clients Of Objects”

Το πιο κάτω διάγραμμα Clients of Objects απεικονίζει τις σχέσεις μεταξύ των κλάσεων του συστήματος. Για να υπάρξει μια συσχέτιση μεταξύ δύο αντικειμένων απαραίτητη προϋπόθεση είναι να υπάρξει κλήση από μία φόρμα που αντιστοιχεί σε κάποιο αντικείμενο, προς κάποια άλλη φόρμα άλλου αντικειμένου.

Με την εκκίνηση του προγράμματος, ο χρήστης έχει άμεση πρόσβαση στη φόρμα του Κεντρικού Μενού αφού πρώτα καταχωρήσει τους κωδικούς πρόσβασης του και επαληθευτούν. Από τη σελίδα του μενού μπορεί να καλεσει οποιαδήποτε από τις σελίδες του συστήματος οι οποίες φαίνονται και από το πιο κάτω σχήμα. Συγκεκριμένα μπορεί να διαχειριστεί τους ερευνητές, τους υπεύθυνους ερευνητές, τα προγράμματα, τα συμβόλαια και τα timesheet.



4.4 Πίνακες Βάσης Δεδομένων

4.4.1 Πίνακας Υπεύθυνου Ερευνητή

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
pcode	int	✓		Κωδικός υπεύθυνου ερευνητή στο σύστημα
name	nvarchar(50)			Όνομα υπεύθυνου ερευνητή
surname	nvarchar(50)			Επίθετο υπεύθυνου ερευνητή
fathurname	nvarchar(50)			Όνομα πατρός υπεύθυνου ερευνητή
birthdate	date			Ημερομηνία γέννησης
sex	nvarchar(50)			Φύλο
maritalstatus	nvarchar(50)			Οικογενειακή Κατάσταση
idnum	nvarchar(50)			Αρ. Ταυτότητας
socialinsnum	int			Αρ. Κοινωνικών Ασφαλίσεων
address	nvarchar(50)			Διεύθυνση
telnum	int			Τηλέφωνο
postcode	int			Ταχυδρομικός κώδικας
town	nvarchar(50)			Πόλη
country	nvarchar(50)			Χώρα
email	nvarchar(50)			Ηλεκτρονική διεύθυνση
int1	int			
char1	nvarchar(50)			

4.4.2 Πίνακας Ερευνητή

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
scode	int	✓		Κωδικός ερευνητή στο σύστημα
name	nvarchar(50)			Όνομα ερευνητή
surname	nvarchar(50)			Επίθετο ερευνητή
fathurname	nvarchar(50)			Όνομα πατρός ερευνητή
birthdate	date			Ημερομηνία γέννησης
sex	nvarchar(50)			Φύλο
maritalstatus	nvarchar(50)			Οικογενειακή Κατάσταση
idnum	nvarchar(50)			Αρ. Ταυτότητας
socialinsnum	int			Αρ. Κοινωνικών Ασφαλίσεων
telnum	int			Τηλέφωνο
postcode	int			Ταχυδρομικός κώδικας
address	nvarchar(50)			Διεύθυνση
town	nvarchar(50)			Πόλη
country	nvarchar(50)			Χώρα

email	nvarchar(50)			Ηλεκτρονική διεύθυνση
int1	int			
char1	nvarchar(50)			

4.4.3 Πίνακας Προγράμματος

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
prcode	int	✓		Κωδικός προγράμματος στο σύστημα
name	nvarchar(200)			Όνομα προγράμματος
shortname	nvarchar(50)			Συντομογραφία προγράμματος
contactnum	nvarchar(50)			Αρ. Συμβολαίου
manager	int			Αρ. Υπεύθυνου Ερευνητή
type	nvarchar(50)			Τύπος
typedetail	nvarchar(100)			Είδος Τύπου
duration	nvarchar(50)			Διάρκεια
tcost	float			Προϋπολογισμός
dcost	float			Κατηγορία direct Cost
subcontracting	float			Κατηγορία subcontracting Cost
personel	float			Κατηγορία personnel Cost
icost	float			Κατηγορία indirect Cost
startdate	date			Ημερομηνία έναρξης
enddate	date			Ημερομηνία Λήξης
openclose	nvarchar(50)			Ανοικτό / Κλειστό
int1	int			
char1	nvarchar(50)			

4.4.4 Πίνακας Συμβολαίου

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
ccode	int	✓		Κωδικός συμβολαίου στο σύστημα
scode	int		✓	Κωδικός ερευνητή στο σύστημα
prcode	int		✓	Κωδικός προγράμματος στο σύστημα
startdate	date			Ημερομηνία έναρξης
enddate	date			Ημερομηνία Λήξης
salary	float			Μισθός
rate	float			Ωριαίος μισθός
int3	int			Υπεύθυνος Ερευνητής ή όχι
int2	int			
char1	nvarchar(50)			Τύπος συμβολαίου

researcher	nvarchar(50)			Όνοματεπώνυμο Ερευνητή
program	nvarchar(50)			Συντομογραφία Προγράμματος

4.4.5 Πίνακας Εισόδου στο Σύστημα

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
username	nvarchar(50)	✓		Όνομα χρήστη
password	nvarchar(50)			Κωδικός χρήστη
type	nvarchar(50)			Τύπος χρήστη για είσοδο στο σύστημα
code	int		✓	Κωδικός χρήστη στο σύστημα
usertype	int			Τύπος χρήστη στο σύστημα

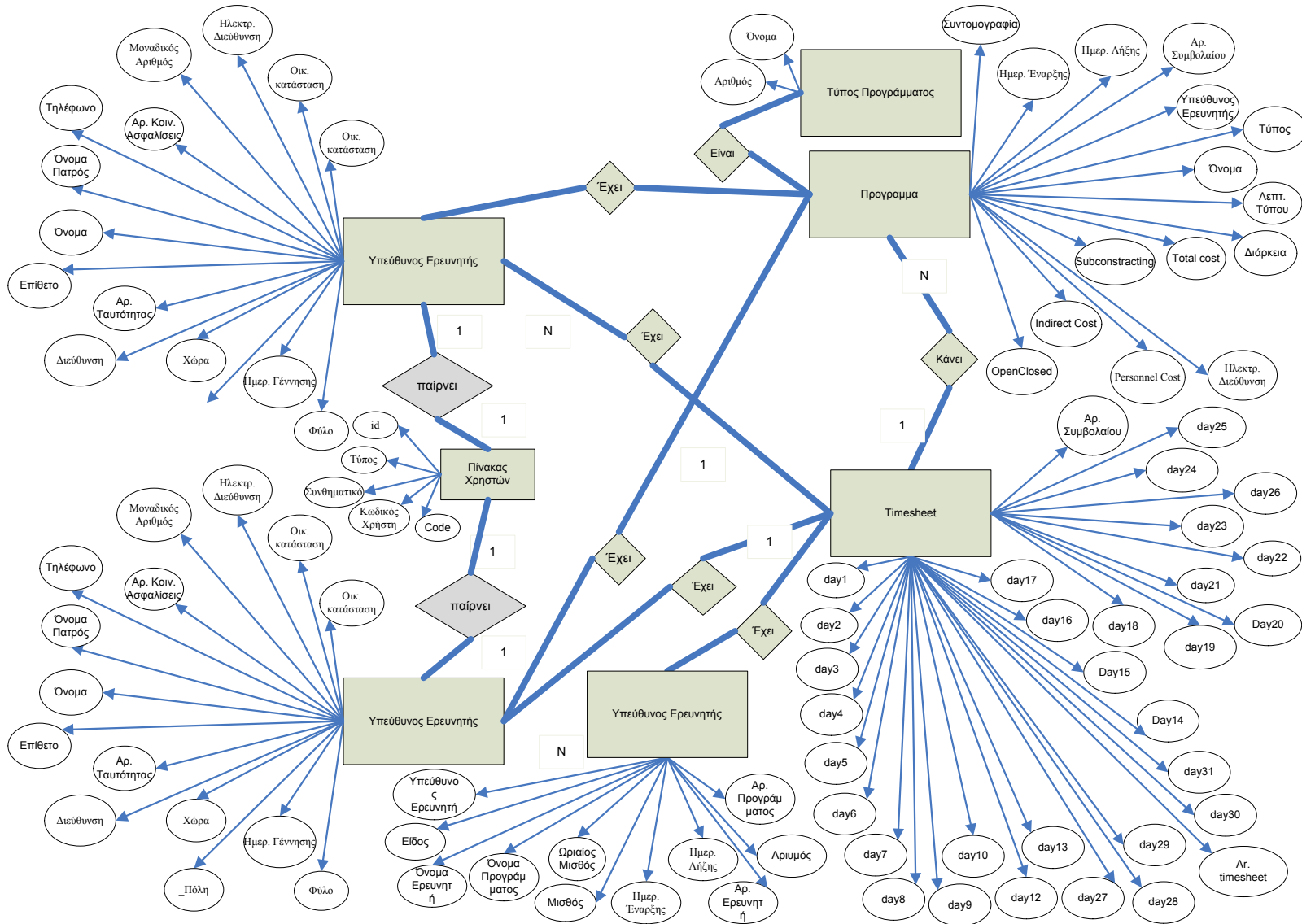
4.4.6 Πίνακας Τύπος Προγράμματος

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
typeid	int	✓		Κωδικός τύπου προγράμματος στο σύστημα
name	nvarchar(50)			Όνομα τύπου προγράμματος

4.4.7 Πίνακας Timesheet

Όνομα	Τύπος	Πρωτεύον κλειδί	Ξένο κλειδί	Περιγραφή
tcode	int	✓		
ccode	int		✓	
year	int			
month	int			
total	float			
Day1	int			
Day2	int			
Day3	int			
Day4	int			
Day14	int			
Day15	int			
Day16	int			
.....	int			
Day30	int			
Day31	int			

4.5 Διάγραμματος Σχέσεων Οντοτήτων



Κεφάλαιο 5

Υλοποίηση

5.1 Περιγραφή εργαλείων που χρησιμοποιήθηκαν στην ανάπτυξη του συστήματος	52
5.2 Περιγραφή γλωσσών προγραμματισμού και βάσης δεδομένων	53
5.3 Αναφορά στα δύσκολα σημεία του κώδικα	54
5.4 Αποσπάσματα κώδικα	55
5.5 Περιπτώσεις δοκιμής	65

5.1 Περιγραφή εργαλείων που χρησιμοποιήθηκαν στην ανάπτυξη του συστήματος

5.1.1 Microsoft Visual Studio

Πρόκειται για ένα εργαλείο το οποίο υποστηρίζει μια μεγάλη ποικιλία γλωσσών προγραμματισμού, όπως επίσης και τη δημιουργία γραφικής διαπροσωπείας. Είναι προϊόν της εταιρίας Microsoft. Για το συγκεκριμένο σύστημα χρησιμοποίησα asp με Visual Basic scriptx, της οποίας θα γίνει περιγραφή στη συνέχεια. Οι δυνατότητες του Visual Studio είναι πολύ πολύτιμες στους προγραμματιστές καθώς υποστηρίζει online ανίχνευση λαθών στον editor του καθώς και αυτόματη στοίχιση και συμπλήρωση κώδικα. Επίσης προσφέρει πάρα πολλές δυνατότητες σε σχέση με την δημιουργία Graphical User Interface καθώς μπορεί κάποιος να σχεδιάσει φόρμες και να τις συνδέσει μεταξύ τους. Υποστηρίζει επίσης τη δημιουργία Web Applications, class designer και επιπλέον παρέχει έναν Database Schema Designer. Επιτρέπει ακόμη την προσθήκη κάποιων Plug Ins τα οποία μπορούν να αυξήσουν την αποδοτικότητα σε κάθε επίπεδο, ακόμη και στην υποστήριξη κάποιων Source control systems. Υπάρχει μια πλήρης εργαλειοθήκη στην οποία μπορούμε να βρούμε ποικίλα εργαλεία για να χρησιμοποιήσουμε στη διαπροσωπεία του συστήματος, όπως combo boxes, radio

buttons, buttons και text boxes. Το συγκεκριμένο εργαλείο επιτρέπει επίσης τη χρήση πολλών λειτουργιών οι οποίες είναι όμοιες με εκείνες του Λειτουργικού Συστήματος Windows, που είναι κατά πολύ οικίες στους μελλοντικούς χρήστες του συστήματος.

5.1.2 Microsoft SQL Server

Είναι το εργαλείο το οποίο χρησιμοποιήσα για τη δημιουργία και διαχείριση της βάσης δεδομένων του συστήματος. Τόσο για τη δημιουργία των πινάκων όσο και για τη δημιουργία των ερωτημάτων (queries) χρησιμοποιήσα τη γλώσσα SQL που είναι η γλώσσα που υποστηρίζει ο Microsoft SQL Server. Το εργαλείο αυτό είναι relational database management system (RDBMS) το οποίο είναι δημιουργία της εταιρίας Microsoft. Ο Microsoft SQL Server είναι ένα εργαλείο το οποίο παρέχει ασφάλεια και προστασία σε διάφορα κρίσιμα δεδομένα. Έχει αρκετά γρήγορη απόδοση σε σχέση με άλλα εργαλεία του ίδιου τύπου. Παρέχει ακόμη αυτόματα τη λειτουργία της εφεδρικής αποθήκευσης βάσης δεδομένων, φτάνει να ενεργοποιηθεί μέσω του προγράμματος στο οποίο χρησιμοποιούμε τον SQL Server για τη διαχείριση της βάσης δεδομένων.

5.2 Περιγραφή γλωσσών προγραμματισμού και βάσης δεδομένων

5.2.1 Γλώσσες Προγραμματισμού

5.2.1.1 Visual Basic

Είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού η οποία υποστηρίζει την υλοποίηση γραφικής διαπροσωπείας. Η γλώσσα αυτή αναπτύχθηκε από την Microsoft και υποστηρίζει κυρίως εφαρμογές που είναι όμοιες με εκείνες του Λειτουργικού Συστήματος Microsoft Windows. Περιέχει μια πλούσια εργαλειοθήκη η οποία μας επιτρέπει να δημιουργήσουμε εξαιρετικής ποιότητας γραφικά για τις εφαρμογές μας και επίσης έχει πολλά χαρακτηριστικά που την κάνουν μια γλώσσα κατάλληλη για τις πλείστες Windows like εφαρμογές.

5.2.1.2 HTML

Είναι μια γλώσσα η οποία χρησιμοποιείται για τη δημιουργία ιστοσελίδων. Τη συγκεκριμένη γλώσσα τη χρησιμοποιήσα για τη δημιουργία του ηλεκτρονικού οδηγού χρήσης και βοήθειας του συστήματος. Συγκεκριμένα δημιούργησα τα κείμενα του

ηλεκτρονικού οδηγού βοήθειας σε ένα επεξεργαστή κειμένου και έπειτα τους δώσαμε μια πιο φιλική προς το χρήστη μορφή μετατρέποντας τα σε μορφή ιστοσελίδας. Μέσω αυτής της μετατροπής σε ιστοσελίδα γίνεται ευκολότερα η πλοήγηση από σελίδα σε σελίδα στον ηλεκτρονικό οδηγό βοήθειας.

5.2.2 Περιγραφή της βάσης δεδομένων

5.2.2.1 SQL Server Express 2005

Ως βάση δεδομένων χρησιμοποίησα την δωρεάν έκδοση του Microsoft SQL Server η οποία ονομάζεται SQL Server Express 2005. Ο SQL Server Express 2005 χρησιμοποιεί την SQL (Structured Query Language) μέσω της οποίας μας παρέχεται η δυνατότητα δημιουργίας συναλλαγών προς αυτή. Επίσης μας παρέχει τη διασφάλιση των δεδομένων και τη μη ανεπιθύμητη αλλαγή τους, μέσω της αυτόματης δημιουργίας εφεδρικής αποθήκευσης της βάσης την οποί και παρέχει.

5.3 Αναφορά στα δύσκολα σημεία του κώδικα

5.3.1 Προβλήματα και λύσεις

Για τη δημιουργία της βάσης και της διαπροσωπείας του συστήματος δεν αντιμετώπισα ιδιαίτερα προβλήματα. δυσκολίες αντιμετώπισα στη συνένωση της βάσης με τη διαπροσωπεία του συστήματος για δημιουργία excel αρχεία καθώς και για να τα ανοίγω και γιατί δεν γνώριζα καθόλου την όλη διαδικασία και εκτός από κώδικα έπρεπε να προστεθούν κάποια reference και βιβλιοθήκες. Επιπρόσθετα, προσπάθησα να βάλω ο χρήστης να κάνει έρευνα για επιλογή φακέλου για να αποθηκεύσει τα timesheet αλλά αυτό το μονοπάτι δεν μπορούσε να αποθηκευτεί γιατί δημιουργείται με thread.

Για να αντιμετωπίσω τα όποια προβλήματα χρησιμοποίησα διάφορες πηγές από το διαδίκτυο, οι οποίες είχαν δείγματα κώδικα της Visual Basic και τα οποία χρησιμοποίησα τροποποιώντας τα ώστε να εκτελούν τη λειτουργία που με αφορούσε.

Επιπρόσθετα αντιμετώπισα αρκετές δυσκολίες στον τρόπο με τον οποίο θα ανέπτυσσα το όλο σύστημα γιατί ξεκίνησα από μηδενική βάση αφού δεν υπήρχε κάποιο άλλο

σύστημα προηγουμένως. Για αντιμετώπιση αυτής της δυσκολίας έλαβα σοβαρά υπόψη μου διάφορους παράγοντες, όπως για παράδειγμα να αναπτύξω το σύστημα με τέτοιο τρόπο ώστε να μπορεί να επεκταθεί εύκολα αλλά και να είναι ευέλικτο. Σε κάθε πίνακα της βάσης δεδομένων έβαζα ένα ακέραιο πεδίο και ένα πεδίο χαρακτήρων, αυτό φάνηκε πολύ χρήσιμο γιατί μετά κάποια από αυτά τα χρησιμοποιήσα χωρίς να χρειαστεί να ξανακτίσω τη βάση δεδομένων μου. Επίσης ανέπτυξα το σύστημα με τρόπο που να μπορεί εύκολα να τύχει συντήρησης τόσο διορθωτικής όσο και ενισχυτικής.

Τέλος αντιμετώπισα δυσκολία στη δημιουργία αναφορών γιατί χρησιμοποιώντας τα έτοιμα δείγματα δεν έτρεχαν online, μονό τοπικά. Κατά την ενσωμάτωσή τους κώδικας γραφόταν στο κώδικα για να τρέχει το σύστημα webconfig με αποτέλεσμα να χρειαστεί να μην τρέχει το σύστημα online. Για να ανταπεξέλθω από αυτό το πρόβλημα διέγραψα τις φόρμες αναφορών και τον κώδικα από το webconfig και έκανα τις δικές με αναφορές με πίνακες οι οποίοι μετατρέπονται και σε excel αρχεία.

5.4 Αποσπάσματα κώδικα

5.4.1 Κώδικας Εισαγωγής νέου Προγράμματος

```
Public Class AddProgram
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
        Handles Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs)
        Handles Button1.Click
        'dilosi metavliton pou tha xrisimopoihthoun
        Dim mtb As New DataSet1TableAdapters.managerTableAdapter
        Dim tb As New DataSet1TableAdapters.programTableAdapter
        Dim name As String
        Dim shortname As String
        Dim prcode As Integer
        Dim ContractNumber As String
        Dim managercode As Integer
        Dim manager As String
        Dim openclosed As String
        Dim PrType As String
        Dim typedetail As String
        Dim duration As String
        Dim tcost As Double
        Dim dcost As Double
        Dim scost As Double
        Dim incost As Double
        Dim pcost As Double
        Dim startdate As Date
```

```

Dim enddate As Date
Dim dateandtime As DateTime
Dim month As String
Dim day As String
Dim yeary As String
Dim abstract As Integer
Dim errorfound As Integer
'arxikopoihsi metavlitis gia evresi lathous
errorfound = 0
prcode = tb.MaxPr() + 1
'elegxos an dothi h oxi to abstract tou programmatos
If Me.Abstract.Checked = True Then
    abstract = 1
Else
    abstract = 0
End If

'gia ola ta pedia tis vasis ginetai elegxos na ine sto sosto format kai
afairountai ta kena

'diavasma ton pedio typou character apo ti forma
name = Trim(emptyText(Me.name.Text))
shortname = Trim(emptyText(Me.shortname.Text))
ContractNumber = Trim(emptyText(Me.ContractNumber.Text))
PrType = Trim(emptyText(Me.PrType.Text))
typedetail = Trim(emptyText(Me.Typedetail.Text))
'duration = Trim(emptyText(Me.duration.Text))
name = Trim(emptyText(Me.name.Text))

'diavasma ton pedio typou akeraiou apo ti vasi
If Not Trim(Me.pcost.Text).Length = 0 Then
    pcost = Val(Me.pcost.Text)
End If

If Not Trim(Me.scost.Text).Length = 0 Then
    scost = Val(Me.scost.Text)
End If

If Not Trim(Me.tcost.Text).Length = 0 Then
    tcost = Val(Me.tcost.Text)
End If

If Not Trim(Me.dcost.Text).Length = 0 Then
    dcost = Val(Me.dcost.Text)
End If

If Not Trim(Me.incost.Text).Length = 0 Then
    incost = Val(Me.incost.Text)
End If

'assign(values) for dropdown menus
If Me.Manager.SelectedIndex = -1 Then
    manager = 0
Else
    manager = Me.Manager.SelectedValue
    ' sex = 1
End If

```

```

managercode = mtb.ManagerId(manager)

If Me.openClosed.SelectedIndex = -1 Then
    openclosed = ""
Else
    openclosed = Me.openClosed.SelectedValue
    ' sex = 1
End If

If Me.smmTextBox.SelectedIndex = -1 Then
    day = 0
Else
    day = Me.smmTextBox.SelectedValue
    'day = 1
End If
If Me.smmTextBox.SelectedIndex = -1 Then
    month = 0
Else
    month = Me.smmTextBox.SelectedValue
    If month = "JANUARY" Then
        month = 1
    ElseIf month = "FEBRUARY" Then
        month = 2
    ElseIf month = "MARCH" Then
        month = 3
    ElseIf month = "APRIL" Then
        month = 4
    ElseIf month = "MAY" Then
        month = 5
    ElseIf month = "JUNE" Then
        month = 6
    ElseIf month = "JULY" Then
        month = 7
    ElseIf month = "AUGUST" Then
        month = 8
    ElseIf month = "SEPTEMBER" Then
        month = 9
    ElseIf month = "OCTOBER" Then
        month = 10
    ElseIf month = "NOVEMBER" Then
        month = 11
    Else
        month = 12
    End If
End If

If Me.syyyyTextBox.SelectedIndex = -1 Then
    year = 0
Else
    year = Me.syyyyTextBox.SelectedValue
    ' year = 1
End If
dateandtime = String.Concat(day, "/", month, "/", year)
startdate = Convert.ToDateTime(dateandtime)

If Me.emmTextBox.SelectedIndex = -1 Then
    day = 0

```



```

Else
    day = Me.emmTextBox.SelectedValue
    'day = 1
End If
If Me.emmTextBox.SelectedIndex = -1 Then
    month = 0
Else
    month = Me.emmTextBox.SelectedValue
    If month = "JANUARY" Then
        month = 1
    ElseIf month = "FEBRUARY" Then
        month = 2
    ElseIf month = "MARCH" Then
        month = 3
    ElseIf month = "APRIL" Then
        month = 4
    ElseIf month = "MAY" Then
        month = 5
    ElseIf month = "JUNE" Then
        month = 6
    ElseIf month = "JULY" Then
        month = 7
    ElseIf month = "AUGUST" Then
        month = 8
    ElseIf month = "SEPTEMBER" Then
        month = 9
    ElseIf month = "OCTOBER" Then
        month = 10
    ElseIf month = "NOVEMBER" Then
        month = 11
    Else
        month = 12
    End If
End If

If Me.eyyyyTextBox.SelectedIndex = -1 Then
    year = 0
Else
    year = Me.eyyyyTextBox.SelectedValue
    ' year = 1
End If
'metatropi se imerominia gia apothikevsi sti vasi dedomenon
dateandtime = String.Concat(day, "/", month, "/", year)
startdate = Convert.ToDateTime(dateandtime)

If Me.duration.SelectedIndex = -1 Then
    duration = 0
Else
    duration = Me.duration.SelectedValue
End If

If Not (errorfound) Then
    'insert program
    tb.InsertPr(prcode, name, shortname, ContractNumber, managercode,
PrType, typedetail, duration, tcost, dcost, scost, pcost, incost, startdate,
enddate, abstract, manager, openclosed)
    Me.error1.Visible = True
    Me.error1.Text = "RECORD SAVED SUCCESSFULLY!"

```

```

        End If

    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs)
Handles Button2.Click
        'ola ta pedia arxikopoiountai me keno h 0 analoga me ton tipo tus
        Me.name.Text = ""
        Me.shortname.Text = ""
        Me.ContractNumber.Text = ""
        Me.Manager.SelectedIndex = 0
        Me.openClosed.SelectedIndex = 0
        Me.PrType.Text = ""
        Me.Typedetail.Text = ""
        Me.duration.Text = ""
        Me.tcost.Text = ""
        Me.dcost.Text = ""
        Me.scost.Text = ""
        Me.pcost.Text = ""
        Me.incost.Text = ""
        Me.smmTextBox.SelectedIndex = 0
        Me.syyyyTextBox.SelectedIndex = 20
        Me.sddTextBox.SelectedIndex = 0
        Me.emmTextBox.SelectedIndex = 0
        Me.eyyyyTextBox.SelectedIndex = 20
        Me.eddTextBox.SelectedIndex = 0
    End Sub
    Private Function emptyText(ByVal txt As String) As String
        'function that checks if a field is empty
        If String.IsNullOrEmpty(Trim(txt)) Then
            txt = ""
        End If
        Return txt
    End Function
End Class

```

5.4.2 Κώδικας Ενημέρωσης Επιλεγμένου Υπεύθυνου Ερευνητή

```

Public Class EditProfessor
    Inherits System.Web.UI.Page
    Dim updateflag As Integer

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load

    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs)
Handles Button1.Click
        'dilosi metavliton pou tha xrisimopoihthoun
        Dim tb As New DataSet1TableAdapters.managerTableAdapter
        Dim pcode As Integer
        Dim name As String
        Dim surname As String
        Dim fatherName As String
        Dim telnum As Integer
        Dim idnum As String

```

```

Dim socialidnum As Integer
Dim address As String
Dim town As String
Dim country As String
Dim postcode As Integer
Dim email1 As String
Dim birthday As Date
Dim sex As String
Dim maritalstatus As String
Dim birthdate As DateTime
Dim errorFound As Boolean

updateflag = 1
errorFound = 0
pcode = Val(Me.pcode.Text)
'elegxos an iparxoun kenoi xaraktires stin arxi h sto telos
'kai aferesi tous mesw tis sinartisi emptytext
'diavazoume tis times pou isaxthikan sti forma
name = Trim(emptyText(Me.name.Text))
surname = Trim(emptyText(Me.surname.Text))
fatherName = Trim(emptyText(Me.fathername.Text))
address = Trim(emptyText(Me.address.Text))
town = Trim(emptyText(Me.town.Text))
country = Trim(emptyText(Me.country.Text))
email1 = Trim(emptyText(Me.email1.Text))
idnum = Trim(emptyText(Me.idnum.Text))

'assign values
If Not Trim(Me.postcode.Text).Length = 0 Then
    postcode = Val(Me.postcode.Text)
End If
If (Not (IsNumeric(Me.postcode.Text))) Then
    If Not Trim(Me.postcode.Text).Length = 0 Then
        Me.error1.Visible = True
        Me.error1.Text = "Post Code field accepts only numeric"
        errorFound = 1
    End If
End If
If Not Trim(Me.telnum.Text).Length = 0 Then
    telnum = Val(Me.telnum.Text)
End If
If (Not (IsNumeric(Me.telnum.Text))) Then
    If Not Trim(Me.telnum.Text).Length = 0 Then
        Me.error1.Visible = True
        Me.error1.Text = "Telephone Number field accepts only numeric
value"
        errorFound = 1
    End If
End If

If Not Trim(Me.socialidnum.Text).Length = 0 Then
    socialidnum = Val(Me.socialidnum.Text)
End If
If (Not (IsNumeric(Me.socialidnum.Text))) Then
    If Not Trim(Me.socialidnum.Text).Length = 0 Then
        Me.error1.Visible = True
        Me.error1.Text = "Social Insuarance Number field accepts only
numeric value"

```

```

        errorFound = 1
    End If
End If

'assign(values)
If Me.sex.SelectedIndex = -1 Then
    sex = 0
Else
    sex = Me.sex.SelectedValue
    ' sex = 1
End If

'assign(values)
If Me.maritalstatus.SelectedIndex = -1 Then
    maritalstatus = 0
Else
    maritalstatus = Me.maritalstatus.SelectedValue
    ' sex = 1
End If

birthdate = Me.birthdate.Text
birthday = Convert.ToDateTime(birthdate)

If Not (errorFound) Then
    'modify student
    tb.UpdateMan(pcode, name, surname, fatherName, birthdate, sex,
maritalstatus, idnum, socialidnum, telnum, postcode, town, country, email1, 0,
"0", address, pcode)
    Me.error1.Visible = True
    Me.error1.Text = "RECORD UPDATED SUCCESSFULLY!"
End If
End Sub
Private Function emptyText(ByVal txt As String) As String
    'function that checks if a field is empty
    If String.IsNullOrEmpty(Trim(txt)) Then
        txt = ""
    End If
    Return txt
End Function

Private Sub EditProfessor_PreRender(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Me.PreRender
    'dilisin metavliton pou tha xrisimopoihthoun
    Dim tb As New DataSet1TableAdapters.managerTableAdapter
    Dim student As DataSet1.managerDataTable
    Dim studentrow As DataSet1.managerRow

    Dim pcode As Integer
    Dim birthday As String
    Dim sex As String
    Dim maritalstatus As String
    Dim result As New DataSet1.managerDataTable

    If updateflag = 0 Then

        Me.pcode.Text = Request.QueryString("pcode").ToString

    End If

```

```

        pcode = Val(Me.pcode.Text)
        student = tb.SelectMan(pcode)
        'diavasma tou pediou tou ereyniti pou epilexthike kai emfanisi tous sti
forma
        studentrow = student.Rows(0)
        Me.name.Text = studentrow.name
        Me.surname.Text = studentrow.surname
        Me.fathername.Text = studentrow.fathername
        Me.address.Text = studentrow.address
        Me.town.Text = studentrow.town
        Me.country.Text = studentrow.country
        Me.email1.Text = studentrow.email
        Me.idnum.Text = Val(studentrow.idnum)
        If (Val(studentrow.socialinsnum) = 0) Then
            Me.socialidnum.Text = ""
        Else
            Me.socialidnum.Text = Val(studentrow.socialinsnum)
        End If
        If (Val(studentrow.telnum) = 0) Then
            Me.telnum.Text = ""
        Else
            Me.telnum.Text = Val(studentrow.telnum)
        End If
        If (Val(studentrow.telnum) = 0) Then
            Me.postcode.Text = ""
        Else
            Me.postcode.Text = Val(studentrow.postcode)
        End If

        sex = studentrow.sex
        maritalstatus = studentrow.maritalstatus
        birthday = studentrow.birthdate.ToString
        Me.birthdate.Text = birthday.Substring(0, 8)

        Me.sex.SelectedValue = sex
        Me.maritalstatus.SelectedValue = maritalstatus
    End Sub

    Protected Sub Button2_Click(ByVal sender As Object, ByVal e As EventArgs)
Handles Button2.Click
        'ekkatharisi olon ton pedio sti forma
        Me.name.Text = ""
        Me.surname.Text = ""
        Me.fathername.Text = ""
        Me.address.Text = ""
        Me.town.Text = ""
        Me.country.Text = ""
        Me.email1.Text = ""
        Me.idnum.Text = ""
        Me.telnum.Text = ""
        Me.postcode.Text = ""
        Me.socialidnum.Text = ""
        Me.sex.SelectedIndex = 0
        Me.maritalstatus.SelectedIndex = 0
        Me.birthdate.Text = ""
    End Sub
End Class

```

5.4.3 Κώδικας Επιλογής Ερευνητή

```
Public Class ViewAllStudent
    Inherits System.Web.UI.Page

    Protected Sub GridView1_SelectedIndexChanged(ByVal sender As Object, ByVal e
As EventArgs) Handles GridView1.SelectedIndexChanged
        Dim scode As Integer
        Dim targetURL As String
        targetURL = "EditStudent.aspx?"
        scode = Me.GridView1.SelectedRow.Cells(1).Text
        targetURL &= "scode=" & Server.UrlEncode(scode)
        Response.Redirect(targetURL)
    End Sub
End Class
```

5.4.3 Κώδικας Εισαγωγής νέου Συμβολαίου

```
Public Class AddContract1
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load

        End Sub

    Protected Sub Calendar1_SelectionChanged(ByVal sender As Object, ByVal e As
EventArgs) Handles Calendar1.SelectionChanged
        Me.startdate.Text = Me.Calendar1.SelectedDate

        'Date.Text = Calendar1.SelectedDate.ToShortDateString()
        Dim div As System.Web.UI.Control = Page.FindControl("divCalendar")

        If TypeOf div Is HtmlGenericControl Then
            CType(div, HtmlGenericControl).Style.Add("display", "none")
        End If
        Calendar1.Visible = False
    End Sub

    Protected Sub ImageButton1_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton1.Click
        Calendar1.Visible = True
    End Sub

    Protected Sub Calendar2_SelectionChanged(ByVal sender As Object, ByVal e As
EventArgs) Handles Calendar2.SelectionChanged
        Me.enddate.Text = Me.Calendar2.SelectedDate

        'Date.Text = Calendar1.SelectedDate.ToShortDateString()
        Dim div As System.Web.UI.Control = Page.FindControl("divCalendar")

        If TypeOf div Is HtmlGenericControl Then
            CType(div, HtmlGenericControl).Style.Add("display", "none")
        End If
        Calendar2.Visible = False
    End Sub
```

```

    Protected Sub ImageButton2_Click(ByVal sender As Object, ByVal e As
System.Web.UI.ImageClickEventArgs) Handles ImageButton2.Click
        Calendar2.Visible = True
    End Sub

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As EventArgs)
Handles Button1.Click
        'dilosi metavliton
        Dim scode As Integer
        Dim prcode As Integer
        Dim ccode As Integer
        Dim startdate As Date
        Dim enddate As Date
        Dim salary As Double
        Dim rate As Double
        Dim errorfound As Integer
        Dim tbp As New DataSet1TableAdapters.programTableAdapter
        Dim tb As New DataSet1TableAdapters.contractTableAdapter
        Dim tbs As New DataSet1TableAdapters.researcherTableAdapter

        errorfound = 0
        ccode = tb.MaxCodeCo() + 1
        startdate = Me.startdate.Text
        enddate = Me.enddate.Text
        If Not (Trim(emptyText(Me.rate.Text))).Length = 0 Then
            rate = 0
        Else
            rate = Me.rate.Text
        End If
        If Not (Trim(emptyText(Me.salary.Text))).Length = 0 Then
            salary = 0
        Else
            salary = Me.salary.Text
        End If

        prcode = tbp.ReturnPrCode(Me.proname.SelectedValue)
        scode = tbs.ReturnRcode(Me.rename.SelectedValue)

        If Not (errorfound) Then
            'insert contract
            tb.InsertCo(ccode, scode, prcode, startdate, enddate, salary, rate, 0,
0, "", Me.rename.SelectedValue, Me.proname.SelectedValue)
            Me.error1.Visible = True
            Me.error1.Text = "RECORD SAVED SUCCESSFULLY!"
        End If
    End Sub
    Private Function emptyText(ByVal txt As String) As String
        'function that checks if a field is empty
        If String.IsNullOrEmpty(Trim(txt)) Then
            txt = ""
        End If
        Return txt
    End Function
End Class

```

5.5 Περιπτώσεις δοκιμής

Για έλεγχο του συστήματος χρησιμοποίησα πραγματικά δεδομένα. Έψαξα στο διαδίκτυο και βρήκα στοιχεία για καθηγητές, φοιτητές και ερευνητικά προγράμματα. Επίσης, κατά τακτά διαστήματα τρέχαμε το πρόγραμμα μου με άτομα του γραφείου διαχείρισης ερευνητικών προγραμμάτων για καλύτερη λήψη απαιτήσεων και μικρό αλλαγές στη λειτουργία του.

Κεφάλαιο 6

Συμπεράσματα Τελικού συστήματος

6.1 Συμπεράσματα	66
6.2 Μελλοντική εργασία	67
6.3 Πλεονεκτήματα συστήματος	67
6.4 Μειονεκτήματα συστήματος	67

6.1 Συμπεράσματα

Το σύστημα που ανέπτυξα ήταν ένα πραγματικό πρόβλημα για το γραφείο διαχείρισης ερευνητικών προγραμμάτων. Για την ανάπτυξη του χρειάστηκε η εφαρμογή του κύκλου ζωής ανάπτυξης συστήματος για την επίλυση του προβλήματος της «Διαχείρισης Ερευνητικών Προγραμμάτων Πληροφορικής». Με την ολοκλήρωση του συστήματος εξετάστηκε με πραγματικά δεδομένα. Συμπερασματικά το σύστημα που υλοποίησα επιλύει σε μεγάλο βαθμό το πρόβλημα μη ύπαρξης ενός συστήματος διαχείρισης των προσωπικών δαπανών των ερευνητικών προγραμμάτων. Το πρόγραμμα που υλοποίησα μπορεί να χρησιμοποιηθεί και από άλλα τμήματα του Πανεπιστημίου Κύπρου.

Επιπλέον, προσπάθησα το σύστημα που ανέπτυξα να είναι όσο πιο ευέλικτο και φιλικό προς το χρήστη γίνεται. Πιστεύω ότι μπορούν να το χειρίζονται με ευκολία οι λειτουργοί του γραφείου, οι οποίοι θα είναι οι κύριοι χρήστες του συστήματος.

Ήταν μια πραγματική εμπειρία που βοήθησε στην πρακτική εφαρμογή του θεωρητικού μέρους των ακαδημαϊκών μαθημάτων που έχω διδαχθεί μέσα στα πλαίσια του ακαδημαϊκού προγράμματος των μεταπτυχιακών σπουδών μου.

6.2 Μελλοντική εργασία

Το σύστημα μου το υλοποίησα με τέτοιο τρόπο ώστε να είναι ευέλικτο εύκολα επεκτάσιμο. Μπορεί στο σύστημα μου να προστεθεί σελίδα που να κρατά διάφορες πληροφορίες για τους χρήστες που είχαν πρόσβαση στο σύστημα και τι έκανα.

Υπάρχει μεγάλη ευκολία επίσης ενημέρωσης των είδη καταχωρημένων στοιχείων στη βάση με νέα στοιχεία που θα προκύπτουν στο μέλλον. Όπως για παράδειγμα η δημιουργία νέου τύπου ερευνητικού προγράμματος, ή αλλαγές των σελίδων πρόσβασης στο σύστημα ανάλογα με τον τύπο του χρήστη.

6.3 Πλεονεκτήματα συστήματος

Το σύστημα μου βοηθάει στην πιο συγκεντρωμένη διαχείριση της οικονομικής κατάστασης της κατηγορίας personnel των ερευνητικών προγραμμάτων. Παρέχεται εύκολη και γρήγορη διαχείριση των πληρωμών για παροχή υπηρεσιών των ερευνητικών προγραμμάτων. Είναι σημαντική η υλοποίηση του συστήματος μου αφού τέτοιο σύστημα δεν υπάρχει και ήταν αναγκαίο. Από το σύστημα μου μπορούν να κρατηθούν στατιστικά στοιχεία που αφορούν τους ερευνητές, υπευθύνους ερευνητές και για τα ερευνητικά προγράμματα του τμήματος.

Η παρεμπόδιση δημιουργίας νέου χρήστη ή ενημέρωση νέου χρήστη από οποιοδήποτε εκτός του λειτουργού αποτελεί σημαντικό στοιχείο ελέγχου ασφάλειας του συστήματος μου.

6.4 Μειονεκτήματα συστήματος

Ένα σημαντικό μειονέκτημα της εφαρμογής που ανέπτυξα είναι ότι δεν μπορεί να υπολογίσει την κατάσταση πίεσης με βάση τα δεδομένα που χρησιμοποίησα για ερευνητές.

Επίσης χρειάζεται προσοχή η διαγραφή μιας εγγραφής διότι μπορούν να διαγραφούν σημαντικές πληροφορίες. Υπάρχει φυσικά κάποιος περιορισμός και προειδοποίηση αλλά εντούτοις χρειάζεται προσοχή και από το χρήστη.

Βιβλιογραφία

- [1] www.asp.net
- [2] <http://forums.asp.net>
- [3] Σημειώσεις μαθήματος «Τεχνολογία Λογισμικού» του τμήματος Πληροφορικής Πανεπιστημίου Κύπρου
- [4] (EBook - PDF - Programming) MSDN Training - Microsoft Visual Studio .NET (with source) ASP, Visual Basic, C#
- [5] Έντυπα που χρειάζεται το Πανεπιστήμιο να συμπληρώνονται