



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Ανάπτυξη δικτυακού πληροφοριακού συστήματος: Πολλαπλό σύστημα Helpdesk Network information system development: A Multiple Helpdesk System
Όνοματεπώνυμο Φοιτητή	Αλεξάνδρα Καραχάλιου
Πατρώνυμο	Θεόδωρος
Αριθμός Μητρώου	ΜΠΣΠ/ 10024
Υπεύθυνος καθηγητής	Χρήστος Δουληγέρης, Καθηγητής
Επιβλέπων ερευνητής	Δρ. Σαράντης Μητρόπουλος

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

(υπογραφή)

(υπογραφή)

Χρήστος Δουληγέρης
Καθηγητής

Δημήτριος Αποστόλου
Επ. Καθηγητής

Παναγιώτης Κοτζανικολάου
Λέκτορας

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Ευχαριστίες

Η εργασία αυτή αναπτύχθηκε στο πλαίσιο της μεταπτυχιακής μου διατριβής του Π.Μ.Σ. Προηγμένα Συστήματα Πληροφορικής. Ξεκίνησε δε ως ιδέα, σε άλλη μορφή και υλοποίηση, στα πλαίσια του μαθήματος «Πληροφορικά Συστήματα στο Διαδίκτυο» σε συνεργασία με τις συμφοιτήτριές μου Αθηνά Γλυκοφρύδη και Μαρία Χαραλαμπίδου με τις οποίες στη διάρκεια του Μ.Π.Σ. αποτελούσαμε «Ένωση» για την ολοκλήρωση του προγράμματος, και τις οποίες ευχαριστώ για την εξαιρετική συνεργασία που μου προσέφεραν σε ολόκληρη την πορεία μας στο Πανεπιστήμιο. Ιδιαίτερα τις ευχαριστώ για τη γενναιόδωρη παραχώρηση της πρώτης προσέγγισης αυτής της εργασίας.

Εξίσου σημαντική για την ολοκλήρωση αυτής της εργασίας ήταν και η υποστήριξη του καθηγητή μας, κυρίου Σαράντη Μητρόπουλου, καθηγητή του μαθήματος «Πληροφορικά Συστήματα στο Διαδίκτυο» και επιβλέποντος της παρούσας εργασίας. Ο κύριος Μητρόπουλος μου έδωσε τις βάσεις για την κατανόηση του προγραμματισμού web εφαρμογών, οδήγησε τις σκέψεις της αρχικής μου ιδέας σε ένα ανώτερο επίπεδο και μου έδωσε σημαντικές κατευθύνσεις σε τεχνολογίες και πεδία που μου ήταν μέχρι την παρούσα εντελώς άγνωστα.

Η ολοκλήρωση της εργασίας αυτής θα ήταν βέβαια αδύνατη χωρίς το συντονισμό, την υπομονή, το ενδιαφέρον και τη συνεργασία του υπεύθυνου καθηγητή μας Χρήστου Δουληγέρη, που παρά το φορτωμένο πρόγραμμά του βρίσκει πάντα χρόνο για τους φοιτητές του και τους βοηθά με προθυμία ακόμα και όταν αυτοί καθυστερούν με τις υποχρεώσεις τους.

Τέλος, θα ήθελα να ευχαριστήσω όλους όσους συνέβαλαν με τον τρόπο τους στην ολοκλήρωση αυτής της προσπάθειας, τους καθηγητές και τους συμφοιτητές μου για τη συνεργασία τους, την οικογένειά μου για την υποστήριξή τους και ιδιαίτερα τη μικρή Δέσποινα για την τεράστια υπομονή της.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Στη Δέσποινα

Περίληψη

Ζητούμενο της παρούσας εργασίας είναι η ανάπτυξη μίας δικτυακής εφαρμογής που έχει ως σκοπό την εξυπηρέτηση των εργασιών που εκτελούνται από τμήματα τεχνικής υποστήριξης - τμήματα αρωγής (helpdesk), και η αξιολόγησή της.

Η εργασία μας αποτελείται από δύο κύρια μέρη: Στο πρώτο μέρος γίνεται μελέτη της εφαρμογής από την πλευρά του τεχνολόγου λογισμικού, όπου παρουσιάζεται η ανάλυση, ο σχεδιασμός και η υλοποίηση της δικτυακής μας εφαρμογής, την οποία αναπτύξαμε κυρίως με βάση το πρότυπο Struts και τις τεχνολογίες JSP/Servlets. Στο δεύτερο μέρος η εφαρμογή μας προσεγγίζεται από την πλευρά του μάντζερ, όπου παρουσιάζεται η αξιολόγησή της εφαρμογής στο περιβάλλον χρήσης της, μέσα από τη μέθοδο Balanced Scorecard.

Στοιχείο καινοτομίας της εφαρμογής που αναπτύχθηκε είναι η δυνατότητα λειτουργίας της σε πολλαπλά επίπεδα. Ο σχεδιασμός της επιτρέπει σε εταιρείες με διαφορετικά μοντέλα – οργανογράμματα να χρησιμοποιούν την εφαρμογή εισάγοντας σε αυτή τη δομή της ιεραρχίας τους. Με τον τρόπο αυτό, η εφαρμογή προάγει τη συνεργασία ανάμεσα σε τμήματα ή και εταιρείες και εξυπηρετεί το σύνολο των υπαλλήλων μιας εταιρείας.

Abstract

The subject of this dissertation is the development of a web application to support the tasks undertaken by a helpdesk department, and ways to evaluate its contribution.

This dissertation consists of two main parts: In part A we present the approach of the application from the view of software developer. Here we present the analysis, the design and the implementation of our web application. We chose to develop our application based on Struts and JSP/Servlets technology. In part B we concentrate on the approach of the application from the view of manager. Here we evaluate the contributions of our application in a real world working environment, using as a tool the Balanced Scorecard.

The application developed has introduced an innovative feature. It has the ability to work in multiple levels. Its design allows the usage from companies with different organizational models, by inserting the company's hierarchy in the application. With this feature, the application promotes the cooperation between departments or companies and serves all of the employees.

Λέξεις - Κλειδιά:

Τμήμα Αρωγής, Helpdesk, δελτία, ανάπτυξη λογισμικού, μελέτη περιβάλλοντος, τεχνολογία λογισμικού, μελέτη/ανάλυση απαιτήσεων, διαγράμματα UML, διαγράμματα BPMN, JSP/Servlets, JSP/Servlets MVC, Apache Struts, JavaScript, jQuery, AJAX, Netbeans, MySql, Αξιολόγηση εφαρμογής, Μάνατζμεντ τμημάτων αρωγής, Στρατηγικοί στόχοι, Balanced Scorecard, BSC, Δείκτες

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Πίνακας Περιεχομένων

Εισαγωγή.....	15
ΜΕΡΟΣ Α: Η εφαρμογή από την πλευρά του τεχνολόγου λογισμικού	17
1. Εφαρμογές Helpdesk και Θεωρία Τεχνολογιών.....	17
1.1 Μελέτη περιβάλλοντος.....	17
1.1.1 Υλοποιήσεις: Χαρακτηριστικά και Παραδείγματα	17
1.1.2 Στη χώρα μας	24
1.1.3 Κριτήρια Επιλογής Λογισμικού Helpdesk	25
1.2 Διαδικασίες Τμημάτων Helpdesk.....	27
1.3 Σύνοψη επιχειρησιακής μελέτης	28
1.4 Χρησιμοποιηθείσες Τεχνολογίες για την Υλοποίηση της Εφαρμογής	28
1.4.1 Εισαγωγή.....	28
1.4.2 Προγραμματισμός στην πλευρά του εξυπηρετητή	29
1.4.3 Τεχνολογίες JSP / Servlets.....	30
1.4.3.2 Java Server Pages - JSPs	33
1.4.4 Αρχιτεκτονική JSP / Servlets Model View Controller (MVC)	35
1.4.5 Apache Struts	37
1.4.6 Συμπληρωματικές Τεχνολογίες.....	38
2. Ανάλυση Εφαρμογής	41
2.1 Ορισμός Προβλήματος	41
2.2 Ανάλυση Απαιτήσεων	43
2.2.1 Απαιτήσεις Εφαρμογής.....	44
2.2.1.1 Ευχρηστία.....	44
2.2.1.2 Ιδιωτικότητα.....	45
2.2.1.3 Διαχείριση δελτίων	45
2.2.1.4 Διαχρονικότητα.....	46
2.2.1.5 Διεθνοποίηση.....	46
2.2.2 Απαιτήσεις Χρήστη	46
2.3 Σχεδιασμός.....	47
2.3.1 Δυνατότητες Πρόσβασης των Χρηστών της Εφαρμογής.....	47
2.3.2 Σχεδιασμός της Διαδικασίας Σύνδεσης (login) του κάθε Χρήστη στην Εφαρμογή καθώς και της Διαδικασίας Εγγραφής του	52

2.3.3 Σχεδιασμός των Διαδικασιών: Δημιουργία Προφίλ Νέας Εταιρείας και Δημιουργία Νέου Τμήματος - Department	54
2.3.4 Σχεδιασμός για την Διαδικασία Αλλαγής Κωδικού Χρήστη	56
2.3.5 Σχεδιασμός για την Διαδικασία Αναζήτησης – Search	57
2.3.6 Σχεδιασμός για την Δημιουργία ενός Νέου Δελτίου Καταγραφής, σε περίπτωση Εμφάνισης Αιτήματος Αποκατάστασης Βλάβης καθώς και για της Διαχείριση του	58
3. Η Εφαρμογή	63
3.1 Αρχική σελίδα- διαδικασία αυθεντικοποίησης	63
3.2 Ανάλυση μενού επιλογών	66
3.2.1 Εισαγωγή	66
3.2.2 Στοιχείο Μενού: «ΕΤΑΙΡΕΙΕΣ»	68
3.2.2 Στοιχείο Μενού: «ΤΜΗΜΑΤΑ»	70
3.2.3 Στοιχείο Μενού: «ΧΡΗΣΤΕΣ»	71
3.2.4 Στοιχείο Μενού: «ΔΕΛΤΙΑ ΒΛΑΒΗΣ»	74
3.2.5 Στοιχείο Μενού: «ΠΕΛΑΤΕΣ»	78
3.2.6 Στοιχείο Μενού: «ΚΑΤΗΓΟΡΙΕΣ»	79
3.2.7 Στοιχείο Μενού: «ΣΤΑΤΙΣΤΙΚΑ»	80
4 Σύνδεση με υποθετικό σενάριο – συμπεράσματα ανάλυσης/υλοποίησης	86
4.1 Χρήση εφαρμογής από ένα τμήμα	86
4.2 Χρήση εφαρμογής στα πλαίσια του intranet μιας εταιρείας	87
4.3 Χρήση εφαρμογής σε intranet και εξωτερικούς συνεργάτες	88
4.4 Χρήση “Third party”	89
ΜΕΡΟΣ Β - Αξιολόγηση . Η εφαρμογή από την πλευρά του μανατζμεντ.	90
5 . Εισαγωγή Μάνατζμεντ	90
5.1 Balanced Scorecard – Θεωρητικό υπόβαθρο	90
5.1.1. Εισαγωγή	90
5.1.2 Balanced Scorecard in action	92
5.2 Balanced Scorecard - Στοιχεία μάνατζμεντ στην εφαρμογή μας	95
5.2.1 Πριν την αγορά	95
5.2.2 Μελέτη τμημάτων αρωγής μέσα από τις οπτικές του Balanced Scorecard	96
5.2.2.1 Στρατηγικός χάρτης	96
5.2.2.2 Δείκτες helpdesk	98
5.2.3 Η εφαρμογή μας. Ένα εργαλείο μάνατζμεντ	102

5.3 Σύνοψη μάνατζμεντ	104
Συμπεράσματα - Περίληψη	105
Παράρτημα Α- Εγχειρίδια Εφαρμογής.....	107
Εγχειρίδιο Προγραμματιστή	107
Εγχειρίδιο Χρήστη	112
Παράρτημα Β - Βοηθητικά εργαλεία.....	113
Γραφική Ανάλυση με BPMN	113
Γραφική Ανάλυση με UML	115
Αναφορές.....	118
Βιβλιογραφία.....	121

Κατάλογος πινάκων, εικόνων, σχημάτων

ΠΙΝΑΚΕΣ

Πίνακας 1 - Οι 10 δημοφιλέστερες δικτυακές εφαρμογές helpdesk. (πηγή υποσημείωση 2)	18
Πίνακας 2 Εφαρμογές Helpdesk Ανοιχτού Κώδικα – Open Source. Πηγή [2]	22
Πίνακας 3 Μενού Στατιστικά	80
Πίνακας 4 Δείκτες helpdesk	100
Πίνακας 5 Κάρτα balanced scorecard – από τη πλευρά του τμήματος αρωγής	101
Πίνακας 6 Στατιστικά και δείκτες	103
Πίνακας 7 Δραστηριότητες (activities) στην BPMN.	113
Πίνακας 8 Γεγονότα (events) στην BPMN.	114
Πίνακας 9 Πύλες (gateways) στην BPMN.	114
Πίνακας 10 Σύνδεσμοι (connectors) στην BPMN	115
Πίνακας 11 Γραφικά Σύμβολα που χρησιμοποιούνται στα Διαγράμματα Περιπτώσεων Χρήσης	115
Πίνακας 12 Γραφικά Σύμβολα που χρησιμοποιούνται στα Διαγράμματα Δραστηριοτήτων	116
Πίνακας 13 Γραφικά Σύμβολα που χρησιμοποιούνται στα Διαγράμματα Ακολουθίας.	117

ΣΧΗΜΑΤΑ

Σχήμα 1 Συνήθης Διαδικασία Εξυπηρέτησης Αιτήματος που Εισέρχεται μέσω Τηλεφωνικής Κλήσης.	27
Σχήμα 2 Συνήθης Διαδικασία Εξυπηρέτησης Αιτήματος που Εισέρχεται μέσω E-mail.	27
Σχήμα 3 Διαδικασία HTTP Αίτησης – Request και HTML Απάντησης - Response.	29
Σχήμα 4 Επικοινωνία Client – Web Server – Web Container.	31
Σχήμα 5 Αρχιτεκτονική JSP / Servlets Model View Controller.	36
Σχήμα 6 Το πλαίσιο Struts με βάση το MVC	37
Σχήμα 7 Πρόταση Ένωσης για Ολοκληρωμένο Δικτυακό Σύστημα Ticketing. Όλοι επικοινωνούν δια μέσου της Εφαρμογής	43
Σχήμα 8 Διαδικασία που Ακολουθείται κατά την Εξυπηρέτηση Κλήσης/Αιτήματος	44
Σχήμα 9 Διάγραμμα Περιπτώσεων Χρήσης για τους Χρήστες της Εφαρμογής.	51
Σχήμα 10 Διάγραμμα Ακολουθίας για τον Σχεδιασμό της Διαδικασίας Εγγραφής Χρήστη (registration) που πραγματοποιείται είτε από τον Administrator είτε από τον Supervisor.	53
Σχήμα 11 Διάγραμμα Δραστηριοτήτων για τον Σχεδιασμό της Διαδικασίας Login.	54
Σχήμα 12 Σχεδιασμός της Διαδικασίας Δημιουργίας Προφίλ Νέας Εταιρείας που πραγματοποιείται από τον Administrator.	55
Σχήμα 13 Σχεδιασμός της Διαδικασίας Δημιουργία Νέου Τμήματος που πραγματοποιείται είτε από τον Administrator είτε από τον Supervisor.	56
Σχήμα 14 Διάγραμμα Δραστηριοτήτων για την Διαδικασία Αλλαγής Κωδικού Χρήστη.	57
Σχήμα 15 Διάγραμμα Ακολουθίας για τον Σχεδιασμό της Διαδικασίας Αναζήτησης.	58
Σχήμα 16 Σχεδιασμός της Διαδικασίας Δημιουργίας Νέου Δελτίου Καταγραφής σε εμφάνιση Αιτήματος Αποκατάστασης Βλάβης	60
Σχήμα 17 Σχεδιασμός της Διαδικασίας Διαχείρισης Καταχωρημένου Ανοιχτού Δελτίου Καταγραφής για την Εξυπηρέτησή του.	62
Σχήμα 18 Η χρήση του Balanced Scorecard ως ένα σύστημα στρατηγικού μανάτζμεντ	92
Σχήμα 19 Ερωτήματα τεσσάρων οπτικών	93

Σχήμα 20 The nine steps success TM	95
---	----

ΕΙΚΟΝΕΣ

Εικόνα 1 Διεπαφή της Εμπορικής Εφαρμογής Zendesk. (πηγή υποσημείωση 3).....	19
Εικόνα 2 Διεπαφή της Εμπορικής Εφαρμογής Fusion. (πηγή υποσημείωση 4).....	20
Εικόνα 3 Διεπαφή Εμπορικής Εφαρμογής Help Desk Pilot (πηγή υποσημείωση 5).....	20
Εικόνα 4 Διεπαφή Εμπορικής Εφαρμογής Hesk (πηγή υποσημείωση 6)	21
Εικόνα 5 Διεπαφή Τεχνικού της Εμπορικής Εφαρμογής Readydesk πηγή:[3]	22
Εικόνα 6 Αρχική Σελίδα Εφαρμογής – Index Page.	63
Εικόνα 7 Διεπαφή για Υπενθύμιση Κωδικού Πρόσβασης.....	64
Εικόνα 8 Αποστολή E-mail για Υπενθύμιση Κωδικού Πρόσβασης.....	64
Εικόνα 9 Διεπαφή Διαχειριστή – Administrator.	65
Εικόνα 10 Διεπαφή Προϊστάμενου – Supervisor.....	65
Εικόνα 11 Διεπαφή Τεχνικού – Technician.	66
Εικόνα 12 Διεπαφή Απλού Χρήστη – Simple User	66
Εικόνα 13 Αρχική σελίδα χρήστη	67
Εικόνα 14 Υπόλοιπα Δελτία Καταγραφής.....	67
Εικόνα 15 Στοιχείο μενού «ΕΤΑΙΡΕΙΕΣ».....	68
Εικόνα 16 Δημιουργία Προφίλ Νέας Εταιρείας.....	69
Εικόνα 17 Διαχείριση ήδη Εγγεγραμμένων Εταιρειών	69
Εικόνα 18 Μήνυμα επιβεβαίωσης διαγραφής εταιρείας (κάλυψη απαίτησης ευχρηστίας)	70
Εικόνα 19 Στοιχείο μενού «ΤΜΗΜΑΤΑ»	70
Εικόνα 20 Καταχώρηση νέου τμήματος - Μήνυμα προτροπής για συμπλήρωση υποχρεωτικού πεδίου (κάλυψη απαίτησης ευχρηστίας)	71
Εικόνα 21 Μήνυμα επιβεβαίωσης καταχώρησης τμήματος (κάλυψη απαίτησης ευχρηστίας) ...	71
Εικόνα 22 Στοιχεία μενού «ΧΡΗΣΤΕΣ».....	72
Εικόνα 23 Δημιουργία νέου χρήστη.....	72
Εικόνα 24 Ανάπτυξη προφίλ χρήστη.....	73
Εικόνα 25 Ενημέρωση προφίλ χρήστη	73
Εικόνα 26 Υπομενού Προφίλ Χρήστη - Αλλαγή Κωδικού Χρήστη.	74
Εικόνα 27 Στοιχείο μενού «ΔΕΛΤΙΑ ΒΛΑΒΗΣ»	74
Εικόνα 28 Φόρμα Συμπλήρωσης για Καταχώρηση Νέου Δελτίου Βλάβης.....	75
Εικόνα 29 Μήνυμα επιτυχούς καταχώρησης (κάλυψη απαίτησης ευχρηστίας)	75
Εικόνα 30 Φόρμα ενημέρωσης δελτίου.....	76
Εικόνα 31 Κλείσιμο δελτίου	77
Εικόνα 32 Χρήση φίλτρου	77
Εικόνα 33 Στοιχείο μενού «ΠΕΛΑΤΕΣ».....	78
Εικόνα 34 Ανάπτυξη εγγραφής πελάτη.....	78
Εικόνα 35 Φόρμα δημιουργία κατηγορίας.....	79
Εικόνα 36 Επιλογή διαχείρισης κατηγοριών- ανάπτυξη εγγραφής	79
Εικόνα 37 Στοιχεία μενού «ΣΤΑΤΙΣΤΙΚΑ».....	80
Εικόνα 38 Φόρμα επιλογής παραμέτρων στατιστικών «ΚΙΝΗΣΗ ΔΕΛΤΙΩΝ»	81
Εικόνα 39 Αποτέλεσμα στατιστικών «ΚΙΝΗΣΗ ΔΕΛΤΙΩΝ».....	81
Εικόνα 40 Φόρμα επιλογής παραμέτρων στατιστικών «ΑΝΟΙΧΤΑ ΔΕΛΤΙΑ»	82
Εικόνα 41 Αποτελέσματα στατιστικών «ΑΝΟΙΧΤΑ ΔΕΛΤΙΑ».....	82
Εικόνα 42 Φόρμα επιλογής παραμέτρων στατιστικών «ΧΡΗΣΤΩΝ»	83

Εικόνα 43 Αποτελέσματα στατιστικών «ΧΡΗΣΤΩΝ»	83
Εικόνα 44 Φόρμα επιλογής παραμέτρων στατιστικών «ΔΕΛΤΙΑ ΑΝΑ ΚΑΤΗΓΟΡΙΑ»	84
Εικόνα 45 Αποτελέσματα στατιστικών «ΔΕΛΤΙΑ ΑΝΑ ΚΑΤΗΓΟΡΙΑ»	84
Εικόνα 46 Φόρμα επιλογής παραμέτρων στατιστικών «ΧΡΟΝΟΣ ΔΙΕΚΠΕΡΑΙΩΣΗΣ»	85
Εικόνα 47 Αποτελέσματα στατιστικών «ΧΡΟΝΟΣ ΔΙΕΚΠΕΡΑΙΩΣΗΣ»	85
Εικόνα 48 Χρήση εφαρμογής από ένα τμήμα.	87
Εικόνα 49 Χρήση εφαρμογής σε intranet	88
Εικόνα 50 Χρήση εφαρμογής σε intranet και από τρίτους	89
Εικόνα 51 Χρήση εφαρμογής σε εταιρεία third party helpdesk	89
Εικόνα 52 Πρωτότυπο στρατηγικού χάρτη.....	93
Εικόνα 53 Στρατηγικός χάρτης από την πλευρά των τμημάτων αρωγής.....	98
Εικόνα 54 Επιλογές στο NetBeans.	107
Εικόνα 55 PhpMyAdmin	108
Εικόνα 56 Σελίδα Διαχείρισης Tomcat – Επιλογή Tomcat Manager.....	109
Εικόνα 57 Tomcat Manager – deploy τον φάκελο HelpDeskAssistant που βρίσκεται στον φάκελο webapps του Tomcat	109
Εικόνα 58 Δομή Web pages	110
Εικόνα 59 Δομή Source packages – αρχείο ApplicationResource.properties	111
Εικόνα 60 Δομή logic.....	111

Εισαγωγή

Η ιδέα της χρήσης λογισμικού για τη διαχείριση αιτημάτων σε διάφορα τμήματα αρωγής και εξυπηρέτησης χρηστών δεν είναι καινούρια. Πάντα υπήρχε η ανάγκη σε τέτοια τμήματα για καταγραφή αιτημάτων και προβλημάτων για εξυπηρέτηση ώστε να μη χάνονται αιτήματα, να υπάρχει συνέχεια σε σχέση με την εργασία των υπαλλήλων που εξυπηρετούν και να κρατείται κάποια σχετική γνώση για επαναλαμβανόμενα προβλήματα.

Όλο και περισσότερες εταιρείες όλων των μεγεθών παγκοσμίως αντιλαμβάνονται την αξία των εφαρμογών helpdesk. Με τις εφαρμογές helpdesk δεν διευκολύνουμε απλά την εργασία των τεχνικών, αυτό που μακροπρόθεσμα επιτυγχάνεται στην πραγματικότητα είναι η διατήρηση καλών σχέσεων ανάμεσα σε εταιρεία και πελάτη, με ό,τι αυτό συνεπάγεται. Έτσι, η βιομηχανία λογισμικού helpdesk θεωρείται μία από τις πλέον αναπτυσσόμενες και κερδοφόρες [1]

Οι εφαρμογές που διαχειρίζονται αιτήματα έχουν ως βασική την έννοια του δελτίου (ticket – κάθε αίτημα/πρόβλημα καταγράφεται σε ένα δελτίο) και βιβλιογραφικά αναφέρονται ως Συστήματα Διαχείρισης Άρσης Βλαβών (Trouble Ticketing) ή απλά εφαρμογές τμημάτων εξυπηρέτησης χρηστών/ πελατών ή τμημάτων αρωγής (helpdesk) με τον αγγλικό όρο να έχει επικρατήσει του ελληνικού.

Τι είναι όμως οι εφαρμογές helpdesk; Είναι το λογισμικό που χρησιμοποιείται για να παρέχεται στους πελάτες τεχνική πληροφορία/υποστήριξη για προϊόντα ή υπηρεσίες, παρέχοντάς τους μία σύνδεση με έναν αντιπρόσωπο υποστήριξης. Στόχος του λογισμικού αυτού είναι η αύξηση της αποδοτικότητας και της παραγωγικότητας της εταιρείας. Στην πραγματικότητα μια εφαρμογή helpdesk είναι μία πολύ μεγάλη και πολύπλοκη βάση δεδομένων με πληροφορίες πελατών, αναφορές κλήσεων, πληροφορίες για προβλήματα υλικού/ λογισμικού, καταγραφές προβλημάτων και λύσεων, κ.α. [1]

Τέτοιου είδους εφαρμογές έχει νόημα να υλοποιούνται ως δικτυακές εφαρμογές αφού συνήθως χρησιμοποιούνται ταυτόχρονα από πλήθος υπαλλήλων. Για το λόγο αυτό έχει επίσης νόημα οι εφαρμογές αυτές να έχουν κάποιο περιβάλλον διαχείρισης χρηστών (της ίδιας της εφαρμογής) για να παρέχεται κάποια σχετική ασφάλεια. Έτσι είναι λοιπόν υλοποιημένες οι περισσότερες εφαρμογές, ενώ όπως θα δούμε και παρακάτω, πολλές από αυτές ενσωματώνουν και άλλα εργαλεία, όπως ημερολόγιο, λίστα με συχνές ερωτήσεις (FAQ)¹ με λύσεις για τους τεχνικούς, εργαλεία αναζητήσεων, υπηρεσίες ηλεκτρονικού ταχυδρομείου, πελατολόγια κ.α.

Οι δημοφιλέστερες εφαρμογές για helpdesk, γράφτηκαν σε PHP,ASP.NET, C#, Perl, και Java με βάσεις δεδομένων SQL και φυσικά οι περισσότερες είναι διαθέσιμες στην αγγλική γλώσσα [2]. Εταιρείες που δεν διαθέτουν λογισμικό helpdesk προσπαθούν να καλύψουν τις ανάγκες τους, καταγράφοντας τα αιτήματα σε λογιστικά φύλλα ή βάσεις δεδομένων.

Στην έρευνά μας διαπιστώσαμε ότι υπάρχει αρκετά μεγάλο πλήθος εφαρμογών ανοιχτού κώδικα και πολλά εμπορικά πακέτα αφιερωμένα για τέτοιες εργασίες (κάποια από τα οποία θα παρουσιάσουμε παρακάτω) και αυτό μας κάνει να σκεφτόμαστε πόσο μεγάλη ανάγκη υπάρχει για τέτοιες εφαρμογές και ότι είναι πολύ σημαντικό η εφαρμογή να ταιριάζει απόλυτα στις ανάγκες κάθε εταιρείας

Γιατί όμως υπάρχουν τόσες εμπορικές εφαρμογές με σημαντικό κόστος και μάλιστα με άδειες ανά χρήστη, όταν είναι δωρεάν τόσες πολλές εφαρμογές;

Θα πρέπει σε αυτό το σημείο να τονίσουμε ότι σχεδόν πάντα οι εφαρμογές helpdesk χρειάζονται παραμετροποίηση για να ταιριάζουν στα δεδομένα της κάθε εταιρείας. Παραμετροποιούμε ώστε να υπάρχει στην εφαρμογή έγκυρη ενημέρωση σε σχέση με τα τμήματα της εταιρείας, τις περιπτώσεις βλαβών, τους συνεργάτες, το πελατολόγιο και τους χρήστες της εφαρμογής. Αυτό βέβαια περιέχει κάποιο κόστος. Ακόμα και για τις δωρεάν εφαρμογές, πρέπει να απασχοληθεί κάποιος προγραμματιστής για να φέρει την εφαρμογή στα «μέτρα» της κάθε εταιρείας. (Κάτι που πολλές φορές είναι δυσκολότερο από το ετοιμάσει μία

¹ FAQ ή FAQs – ακρωνύμιο για το Frequent Asked Questions- μία λίστα με τις συχνότερες ερωτήσεις για ένα θέμα μαζί με τις απαντήσεις τους.

νέα εφαρμογή από την αρχή.) Οι εταιρείες που εκδίδουν εμπορικές εφαρμογές συνήθως τις υποστηρίζουν τόσο για παραμετροποίηση όσο και για νέες εκδόσεις, ώστε να καλύπτουν τυχόν προβλήματα ασφάλειας.

Είναι πραγματικά πολύ σημαντικό κάθε εταιρεία ανάλογα με το μέγεθος και το σκοπό της, να διαλέξει μία σωστή λύση (στα μέτρα της) ώστε να παρέχει σωστές υπηρεσίες και υποστήριξη. Υπάρχει πάντα ο κίνδυνος όταν συντηρείται μια μεγάλη εφαρμογή, μια εταιρεία να χάνει το σκοπό της αποστολής της, που είναι κύρια η εξυπηρέτηση των πελατών και όχι η γραφειοκρατία πολύπλοκων διαδικασιών/μέσων. (Κάτι που δυστυχώς συμβαίνει πολύ συχνά!). Από την άλλη πλευρά, πρέπει η εφαρμογή που θα επιλεγεί να καλύπτει πλήρως τις λειτουργικές ανάγκες και να μη χρειάζονται επιπλέον εργαλεία για τη λειτουργία.

Γι' αυτό και η διαδικασία επιλογής λογισμικού είναι ένα δύσκολο έργο και οι εταιρείες μπορεί να κάνουν μήνες να επιλέξουν, λόγω κυρίως των πάρα πολλών επιλογών που διατίθενται στην αγορά.

Το πλήθος των εφαρμογών, μας κάνει επίσης να σκεφτόμαστε ότι η εφαρμογή μας αντιμετωπίζει μεγάλο ανταγωνισμό. Πώς λοιπόν θα μπορούμε να φέρουμε σε μία τέτοια πολυεφαρμοσμένη ιδέα, στοιχεία καινοτομίας; Η ιδέα μας είναι η εφαρμογή μας να είναι ένα πολλαπλό σύστημα helpdesk, που θα επιτρέπει και θα προάγει τη συνεργασία ανάμεσα σε τμήματα ή και εταιρείες.

Τα κεφάλαια που ακολουθούν αναλύονται ως εξής:

Μέρος Α - Στο κεφάλαιο 1 παρουσιάζεται το θεωρητικό υπόβαθρο για το περιβάλλον και το είδος της εφαρμογής, για τις τεχνολογίες και τις τεχνικές που θα χρησιμοποιήσουμε για την υλοποίησή της. Στο κεφάλαιο 2 βρίσκεται η ανάλυση απαιτήσεων και ο σχεδιασμός της εφαρμογής. Στο κεφάλαιο 3 παρουσιάζεται η ίδια η εφαρμογή. Στο κεφάλαιο 4, δίνονται κάποια συμπεράσματα από το μέρος Α της εργασίας.

Μέρος Β. – Στο Κεφάλαιο 5, παρουσιάζεται το θεωρητικό υπόβαθρο στοιχείων του μανάτζμεντ και του Balanced Scorecard. Στο κεφάλαιο 6, ακολουθεί πρακτικό μέρος όπου παρουσιάζονται στοιχεία μανάτζμεντ στην εφαρμογή μας.

Η εργασία μας κλείνει στον επίλογο όπου δίνονται συμπεράσματα και μελλοντικές βελτιώσεις, ενώ υπάρχουν παραρτήματα με τα εγχειρίδια χρήστη και προγραμματιστή, βοηθητικά εργαλεία κ.α.

ΜΕΡΟΣ Α: Η εφαρμογή από την πλευρά του τεχνολόγου λογισμικού

1. Εφαρμογές Helpdesk και Θεωρία Τεχνολογιών

1.1 Μελέτη περιβάλλοντος

Εισαγωγή

Στις μέρες μας υπάρχει η τάση, όπως και σε πολλά άλλα ζητήματα, οι εφαρμογές helpdesk, να προσφέρουν πλήθος άλλων χαρακτηριστικών και να γίνονται όλο και περισσότερο web based. Επίσης κυρίως στο εξωτερικό, όλο και περισσότερες εφαρμογές προσφέρονται μέσω cloud computing, λόγω της μεγάλης ανάπτυξης και διάδοσης της συγκεκριμένης τεχνολογίας εκεί. Για το λόγο αυτό οι εφαρμογές πλέον διαχωρίζονται σε δύο κατηγορίες: σε self-hosted και σε cloud computing.

Οι helpdesk εφαρμογές που είναι self-hosted χάνουν πλέον σε δημοτικότητα λόγω του cloud computing και αυτό επειδή γενικότερα κοστίζουν περισσότερο, απαιτούν περισσότερο χρόνο ανάπτυξης, έχουν διαχειριστικό κόστος και έχουν μικρότερη δυνατότητα κλιμάκωσης με κίνδυνο να μην μπορούν να καλύψουν τις ανάγκες της εταιρείας καθώς αυτή αναπτύσσεται. Από την άλλη πλευρά, έχουν το πλεονέκτημα της αποκλειστικότητας και του πλήρους ελέγχου, με μικρότερο ρίσκο σε θέματα ασφάλειας (αν βέβαια υπάρχει η σχετική φροντίδα από τους διαχειριστές του πληροφοριακού συστήματος της εφαρμογής).

Οι εφαρμογές σε cloud computing είναι συνήθως πιο ευέλικτες σε σχέση με την ανάπτυξη της εταιρείας. Τείνουν να έχουν περισσότερες ενημερώσεις, επαγγελματική υποστήριξη, πρόσβαση για τους χρήστες από οπουδήποτε και οποτεδήποτε με πλήρη δικαιώματα, κάτι που είναι πολύ σημαντικό για εταιρείες με υπαλλήλους που ταξιδεύουν συχνά ή έχουν πολλά υποκαταστήματα. Τα συστήματα αυτά έχουν χαμηλότερο κόστος σε σχέση με τα self-hosted και απαιτούν μικρότερο χρόνο ανάπτυξης. [1]

1.1.1 Υλοποιήσεις: Χαρακτηριστικά και Παραδείγματα

Χαρακτηριστικά

Τα χαρακτηριστικά [1] που μπορούν πλέον να υπάρχουν σε εφαρμογές help desk είναι πάρα πολλά. Κάποια από τα οποία είναι τα παρακάτω:

- Web-based Interface - Γραφική διεπαφή δια μέσου δικτύου, προσβάσιμη από φυλλομετρητή. Παρέχει στους χρήστες άμεση σύνδεση με τις πληροφορίες και τους αντιπροσώπους της εταιρείας.
- Single Sign On Authentication (SSO) – Δίνει τη δυνατότητα στους χρήστες να πιστοποιούνται στην εφαρμογή με τα credentials που χρησιμοποιούν και στο υπόλοιπο σύστημα.
- Branding – Πολλές εφαρμογές δίνουν τη δυνατότητα μορφοποίησης με τα λογότυπα, τα χρώματα και τις γραμματοσειρές που χαρακτηρίζουν την εταιρεία.
- Πολυγλωσσικότητα – Δυνατότητα διεπαφής σε διάφορες γλώσσες.
- Γνωσιακή Βάση - Βάση με λύσεις παλαιότερων προβλημάτων και με τις λύσεις των συχνότερων προβλημάτων, όπου κάποιος τεχνικός μπορεί να συμβουλευτεί για να εξυπηρετήσει κάποιο αίτημα.
- Forums – Forums στις διεπαφές των πελατών, όπου κάποιος μπορεί να επικοινωνήσει ανοιχτά για τα προϊόντα ή τις υπηρεσίες της εταιρείας. Τα forums βοηθούν την εταιρεία να παρακολουθεί την απήχηση που έχουν τα προϊόντα της στην αγορά και να διακρίνει τις ανάγκες των πελατών της.

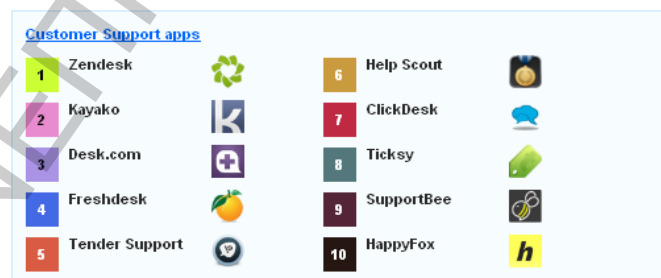
- Προηγμένα Εργαλεία Αναζήτησης - Για αναζήτηση σε ολόκληρη τη βάση της εφαρμογής.
- Tagging – Δημιουργία tags (ετικετών) για να δίνεται κάποιο επιπλέον επίπεδο οργάνωσης στα δελτία.
- Αυτόματες Απαντήσεις - Άμεσες απαντήσεις με την παραλαβή του αιτήματος, οι οποίες έχουν προκαθοριστεί από το προσωπικό, άλλα και περιοδικές αποστολές μηνυμάτων για θέματα που πρέπει να υπενθυμίζονται.
- Tracking Δελτίων - Οι τεχνικοί αλλά και οι πελάτες έχουν τη δυνατότητα να παρακολουθούν την εξέλιξη των αιτημάτων τους.
- Συγχώνευση Δελτίων – Όταν οι πελάτες στέλνουν για το ίδιο πρόβλημα πάνω από ένα αιτήματα, τα δελτία αυτά μπορούν να συγχωνευτούν σε ένα.
- Διαχείριση Spam - Όποιο μήνυμα φαίνεται να είναι spam, δρομολογείται αυτόματα σε καραντίνα.
- Web-based Απομακρυσμένη Διαχείριση - Δυνατότητα οι τεχνικοί να διορθώνουν απομακρυσμένα προβλήματα μέσω Διαδικτύου. Μια τέτοια υπηρεσία μπορεί να μειώσει σημαντικά το χρόνο διεκπεραίωσης ενός αιτήματος.
- Στατιστικά - Αναφορές πραγματικού χρόνου για τις αποδόσεις των τεχνικών, τα είδη των προβλημάτων κ.α.
- Mobile Optimization – Είναι η δυνατότητα πρόσβασης στην εφαρμογή μέσω «έξυπνων τηλεφώνων» - smart phones.

Παραδείγματα Υλοποιήσεων

Ας δούμε σε αυτό το σημείο μερικές παλαιότερες και νέες δημοφιλείς εμπορικές εφαρμογές για να διαπιστώσουμε πώς καλύπτονται οι ανάγκες της αγοράς και πώς έχουν εξελιχθεί τα χαρακτηριστικά του ticketing. Ποιές είναι σήμερα οι δημοφιλέστερες εφαρμογές; Τι χαρακτηριστικά έχουν; Πόσο κοστίζουν;

Στον πίνακα που ακολουθεί (πίνακας 1) εμφανίζονται οι δημοφιλέστερες 10 δικτυακές εφαρμογές helpdesk τον Ιούνιο του 2012 με βάση τα στοιχεία από το Alexa². Φυσικά η λίστα αυτή αναφέρεται σε δικτυακή κίνηση και έτσι, δε περιλαμβάνει τοπικές εφαρμογές. Ακολούθως παρουσιάζονται κάποια ενδεικτικά παραδείγματα εφαρμογών helpdesk εμπορικών και open source εφαρμογές, όχι μόνο με βάση τη δημοτικότητα αλλά και με κριτήριο τη διαφορετικότητα. (επαληθεύοντας όσα αναγράφονται στην § 1.1.1)

Πίνακας 1 - Οι 10 δημοφιλέστερες δικτυακές εφαρμογές helpdesk. (πηγή υποσημείωση 2)



Based on Alexa traffic data, updated on 13 Jul 2012 at 03:40 GMT.

² Ιστότοπος <http://www.alexa.com>. Alexa ranking: Υπολογίζει την κίνηση αναλύοντας τη χρήση του Διαδικτύου εκατομμυρίων χρηστών που διαθέτουν το toolbar Alexa αλλά και από δεδομένα από άλλες πηγές κίνησης. Η πληροφορία ταξινομείται, γίνεται ανώνυμη, και υπολογίζεται μέχρι να δοθεί ως υπηρεσία από το Alexa.

Zendesk ³

Αποτελεί μια cloud computing λύση για μικρές και μεγάλες επιχειρήσεις. Είναι εφαρμογή χωρίς εγκατάσταση και παραμετροποιείται για να παρουσιάζεται σύμφωνα με την επιθυμία του πελάτη. Τα χαρακτηριστικά του Zendesk περιλαμβάνουν διεπαφή web-based, SSL κρυπτογράφηση, SSO, στατιστικά πραγματικού χρόνου, ροή εργασιών, ανάθεση δελτίων με βάση κανόνες. Το Zendesk επίσης περιλαμβάνει φίλτρα spam, συμβατότητα με «έξυπνα τηλέφωνα»- smart phones και πλήρες ιστορικό όλων των δελτίων. Το κόστος του ξεκινάει στα \$9 ανά μήνα ανά πράκτορα για μέγιστο όριο τριών πρακτόρων. Το ακριβότερο πακέτο κοστίζει \$99 ανά μήνα ανά πράκτορα. [1]

Εικόνα 1 Διεπαφή της Εμπορικής Εφαρμογής Zendesk. (πηγή υποσημείωση 3)

Kayako Fusion ⁴

Το πακέτο Fusion διατίθεται και ως self-hosted και ως remote-hosted στους εξυπηρετητές της Kayako. Μπορεί να λειτουργήσει καλά για μικρές και μεσαίες επιχειρήσεις και διαθέτει φιλική διεπαφή χρήστη και χρέωση. Οι μεγαλύτερες εταιρείες μάλλον θα πρέπει να αγοράσουν κάποια πρόσθετα. Τα χαρακτηριστικά του Fusion περιλαμβάνουν live chat, αυτόματες απαντήσεις δελτίων, διαχείριση email, δυνατότητα δημιουργία γνωσιακής βάσης δεδομένων και εργαλεία διαχείρισης project. Το Fusion είναι γραμμένο σε PHP και τρέχει σε: Linux, FreeBSD, Solaris,

³ Zendesk: επίσημος ιστότοπος <http://www.zendesk.com/>

⁴ Επίσημος ιστότοπος <http://www.kayako.com/>

και Microsoft Windows. Ως self-hosted πακέτο κοστίζει \$999 και ως web-hosted \$49 ανά μήνα ανά θέση. [1]

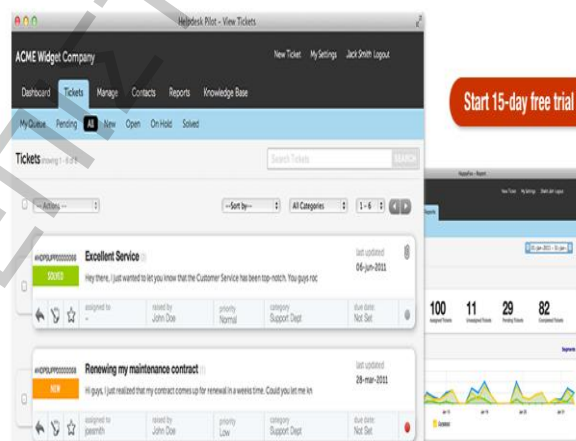


Εικόνα 2 Διεπαφή της Εμπορικής Εφαρμογής Fusion. (πηγή υποσημείωση 4)

Help Desk Pilot ⁵



Το Help Desk Pilot είναι ένα self – hosted PHP λογισμικό που έχει τη δυνατότητα να μετατρέπει emails σε δελτία. Διαθέτει εκδόσεις για μικρές και μεσαίες επιχειρήσεις. Τα χαρακτηριστικά του Help Desk Pilot περιλαμβάνουν ανάθεση δελτίων με βάση κανόνες, αυτόματες απαντήσεις email, υποστήριξη πολυγλωσσικότητας, RSS feeds και εργαλεία αναζήτησης. Το κόστος του ξεκινά στα \$399.95 και φτάνει τα \$1,799.95. [1]



Εικόνα 3 Διεπαφή Εμπορικής Εφαρμογής Help Desk Pilot (πηγή υποσημείωση 5)

⁵ Επίσημος Ιστότοπος <http://www.helpdeskpilot.com>

Hesk ⁶

Το Hesk είναι ένα δημοφιλές λογισμικό γραμμένο σε PHP που διαθέτει και δωρεάν έκδοση. Διατίθεται ως self-hosted και ως remote-hosted και απευθύνεται σε μικρές και μεσαίες επιχειρήσεις με περιορισμένο προϋπολογισμό. Τα χαρακτηριστικά του Hesk περιλαμβάνουν διεπαφή Web-based, απεριόριστη γνωσιακή βάση, αποτροπή spam, αυτόματες απαντήσεις email και προηγμένες αναζητήσεις. [1]



Εικόνα 4 Διεπαφή Εμπορικής Εφαρμογής Hesk (πηγή υποσημείωση 6)

PeriDesk ⁷

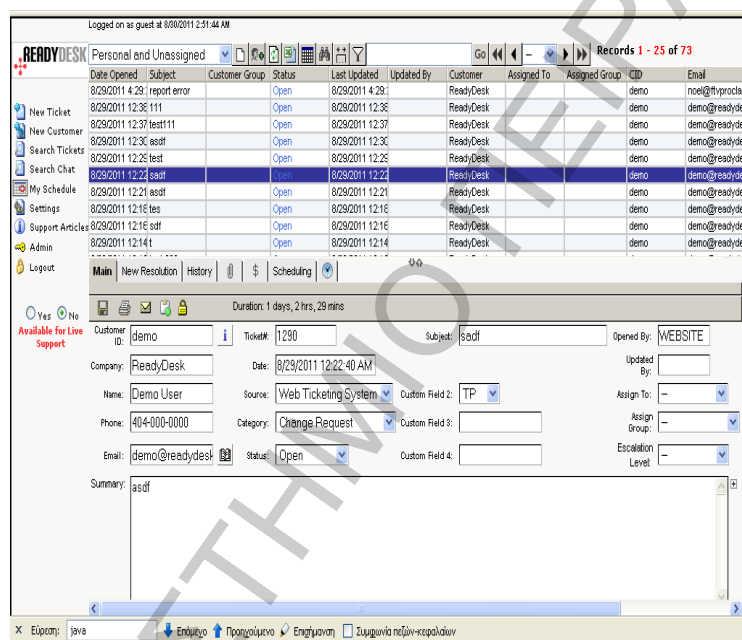
Λογισμικό γραμμένο σε Perl όπως προδίδει το όνομά του, διαθέσιμο ως self-hosted και σε cloud computing. Διαθέτει πακέτα που απευθύνονται σε μικρές και μεγάλες επιχειρήσεις. Για να τρέξει η self-hosted έκδοση του PeriDesk πρέπει να υπάρχει Perl, MySQL, Apache για συστήματα Linux. Για μηχανήματα Windows απαιτείται MySQL. Τα χαρακτηριστικά του PeriDesk περιλαμβάνουν brand user διεπαφή, branded email templates, γνωσιακή βάση, αυτόματη παραγωγή δελτίων και στατιστικά. Μία άδεια για μικρή επιχείρηση ξεκινάει στα \$199.99, ενώ μία μεγαλύτερη έκδοση φτάνει στα \$499.99. [1]

⁶ Επίσημος ιστότοπος <http://www.hesk.com/>

⁷ Επίσημος ιστότοπος <http://www.perldesk.com/>

ReadyDesk⁸

Η ReadyDesk ήταν η πρώτη php help desk εφαρμογή που εκδόθηκε εμπορικά από την εταιρεία ReadyDesk.com (1999). Η εφαρμογή υποστηρίζεται ακόμα και έχει φτάσει στην έκδοση 5.9, ενώ είναι εμπορικό προϊόν με αρχικό κόστος απόκτησης τα \$200. Τα χαρακτηριστικά που ενσωματώνει το ReadyDesk σήμερα είναι πολύ περισσότερα από ένα σύστημα δελτίων και περιλαμβάνουν σύστημα χρέωσης, time tracking και scheduling για τους τεχνικούς, γνωσιακή βάση δεδομένων, διαχείριση e-mail, live chat, διαχείριση αποθεμάτων κ.α. Στις μέρες μας διατίθεται σε εκδόσεις για συστήματα windows και μη, σε γλώσσες asp.net και php και διαθέτει πολλαπλές διεπαφές για τεχνικούς, διαχειριστές αλλά και πελάτες. [3]



Εικόνα 5 Διεπαφή Τεχνικού της Εμπορικής Εφαρμογής Readydesk πηγή:[3]

Εφαρμογές ανοιχτού κώδικα

Εκτός των εμπορικών εφαρμογών δεν πρέπει να ξεχνάμε και τις εφαρμογές ελεύθερου κώδικα που είναι πολλές φορές καλή λύση για μικρής κλίμακας τμήματα IT. Ο επόμενος πίνακας (Πίνακας 2) παρέχει μία λίστα με εφαρμογές help desk open source [2].

Πίνακας 2 Εφαρμογές Helpdesk Ανοιχτού Κώδικα – Open Source. Πηγή [2]

aDefHelpDesk	Προϊόν σε ASP.NET με DotNetNuke module.
Asset Tracker	Δημιουργεί πολλαπλές asset databases περιέχοντας όποια πληροφορία επιθυμείτε να παρακολουθείτε.
Astres	Ένα PHP based help desk εργαλείο στην Αγγλική και Γαλλική γλώσσα.

⁸ Επίσημος Ιστότοπος <http://www.readydesk.com/>

Bartley Helpdesk	Ένα ελαφρύ help desk εργαλείο για μικρά και μεσαία τμήματα IT με διαχείριση αποθέματος.
BATTS	Σχεδιασμένο για Linux/Unix command line. Περιλαμβάνει διεπαφή e-mail για πελάτες.
Big Help	Ένα help desk portal φτιαγμένο με Ruby on Rails.
BugTracker.NET	Web based εφαρμογή για bugs γραμμένη σε ASP.NET, C#, με χρήση MSDE. Περιλαμβάνει εργαλείο screen capture.
Bugzilla	Perl based software bug tracking system που χρησιμοποιείται και για help desk. Με χρήση από το Mozilla Foundation.
DITrack	Με command line interface.
Double Choco Latte	Ένα GNU επιχειρηματικό πακέτο που διαχειρίζεται κλήσεις αλλά και έργα.
Elastik	PHP help desk ticket software με πλήρως προσδιορισμό σχήμα για workflow.
eTicket Support	PHP-based electronic support ticket system.
Eventum	Ένα open source help desk πακέτο από τους δημιουργούς της Βάσης Δεδομένων MySQL.
Help Desk Lite	Ένα απλό πακέτο λογισμικού help desk με δύο κύριες λειτουργίες: service ticket tracking και operator assignment.
Help Desk Reloaded	PHP based help desk λογισμικό. Χαρακτηριστικά του: προτεραιότητες στα δελτία, πολλαπλούς τεχνικούς υποστήριξης πελατών και αναζήτηση.
Helpdesk Issue Manager	Open source λογισμικό σε PHP.
Information Resource Manager	PHP based help desk και asset tracker για τμήματα IT. Υπάρχει και έκδοση GLPI.
IssueTrackerProduct	Κυρίως για διαχείριση software bugs με έμφαση στην επικοινωνία email.
itracker	Java based open source help desk.
jHelpdesk	Java/JSP based open source corporate help desk system.
JTrac	Java based open source issue tracking system.
Jutda Helpdesk	Σύστημα ticket tracking για μικρές επιχειρήσεις.
Liberum Help Desk	Πακέτο λογισμικού Web based σε ASP και VBScript με απαραίτητο Microsoft SQL Server ή Access. Αυτόματη δρομολόγηση περιπτώσεων, ενημέρωση emails και αναφορές.
Mantis	PHP λογισμικό bug tracking και issue tracking σύστημα.
One or Zero	Πακέτο λογισμικού help desk βέλτιστο για task oriented help desks.
OpenPsa	Ένα πλήρες σύστημα project με help desk component.
OS Ticket	Ένα ελαφρύ ticketing system γραμμένο σε PHP και σχεδιασμένο για εύκολη εγκατάσταση και χρήση.
OTRS	Ένα σύστημα trouble ticket για track κλήσεων και emails.

phpBugTracker	Λογισμικό web-based bug tracker. Ο σχεδιασμός του επικεντρώνεται στο διαχωρισμό των επιπέδων παρουσίασης, εφαρμογής και βάσης δεδομένων για ευκολότερη παραμετροποίηση.
Redmine	Project management web application.
Request Tracker	Open source ticket tracking εργαλείο. Διαθέτει web-based help desk portal όπως επίσης υποστήριξη για δημιουργία ticket μέσω email.
Roundup	Ένα απλό σύστημα issue tracking με command line, web και email interfaces. Περιλαμβάνει και επίσης todo lists και sales lead tracking.
Savane	Web-based Libre Software hosting system.
SimpleDesk	Ένα help desk plugin για το Simple Machines Forum system.
Sinergia	Open source framework για δημιουργία customized help desk εφαρμογών C# .NET 3.0. Παρέχει μία βάση όπου μπορείτε να αναπτύξετε τη δική σας web-based help desk λύση.
SiT! Support Incident Tracker	SLA aware support incident tracking system, PHP/MySQL.
Slick Ticket - Featured App	ASP based ticket system με υποστήριξη SQL Server and Active Directory.
Subissue	Χρησιμοποιεί Subversion version control system για αποθήκευση θεμάτων στο software code repository.
SugarCRM Open Source	Κυρίως CRM system αλλά επίσης διαχειρίζεται ελαφρά θέματα help desk.
Trac	Minimalistic web based project management and bug tracking tool. Trac also includes wiki functionality. Written in Python.
Trouble Ticket Express	Web based CGI help desk system. Both MySQL and Microsoft SQL Server supported.
WATS Wordpress Help Desk Plugin	Wordpress plugin για τρέξιμο help desk σε Wordpress.
ZenTrack	Help desk, bug tracking, and project management system.
ZwikiTracker	Κάθε δελτίο σε αυτό το σύστημα είναι μια σελίδα wiki.

1.1.2 Στη χώρα μας

Δυστυχώς ή ευτυχώς για τη χώρα μας δεν μπορέσαμε να εντοπίσουμε κάποια μεγάλη εγχώρια εμπορική εφαρμογή που να καλύπτει τις ανάγκες τμημάτων αρωγής.

Διαπιστώνουμε ότι οι περισσότερες εταιρείες αναζητούν custom made προϊόντα και προσαρμόζουν λογισμικά ανοιχτού κώδικα κυρίως για τρεις λόγους.

Πρώτον, η πλειοψηφία των επιχειρήσεων στη χώρα μας είναι μικρομεσαίες επιχειρήσεις, που δε διαθέτουν μεγάλο προϋπολογισμό για εργαλεία πληροφορικής, αλλά ωστόσο υπάρχει πλέον κάποια χρήση του υπολογιστή και του Διαδικτύου. [4,5,6] Δεύτερον, η χρήση της πληροφορικής είναι ακόμα σε μικρότερα επίπεδα σε σχέση με το εξωτερικό και τρίτον, υπάρχει πάντα το εμπόδιο της γλώσσας [7]. Οι περισσότερες διεπαφές είναι γραμμένες στην αγγλική. Τα περισσότερα πακέτα λογισμικού του εξωτερικού δεν προσφέρονται στην ελληνική, την οποία φυσικά προτιμούμε να χρησιμοποιούμε ως μητρική μας γλώσσα.

Για όλα αυτά βρίσκουμε ακόμα ένα λόγο για τον οποίο επιλέγουμε την ανάπτυξη της εφαρμογής Helpdesk.

Εφαρμογή για Λειτουργίες Τμημάτων Αρωγής (Πολλαπλό σύστημα Helpdesk)

1.1.3 Κριτήρια Επιλογής Λογισμικού Helpdesk

Με τι κριτήρια όμως μία εταιρεία θα πρέπει να επιλέξει να αγοράσει τέτοιο λογισμικό και πότε μία εταιρεία είναι έτοιμη για μια τέτοια επένδυση; Εκτός από το μέγεθος της εταιρείας και τον προϋπολογισμό, υπάρχουν κάποια σημεία που κάποιος θα μπορούσε να λάβει υπ' όψιν του πριν αποφασίσει το λογισμικό που θα εφαρμόσει.[1]

1. Απόφαση για δωρεάν ή εμπορικό λογισμικό. Μπορεί ένα δωρεάν λογισμικό να καλύψει τις ανάγκες μου; Υπάρχει κατάλληλα εκπαιδευμένο προσωπικό για να το υποστηρίξει; Μήπως τελικά μου κοστίζει περισσότερο σε χρήμα, χρόνο αλλά και σε φήμη; Χρειάζεται εμπειρία για να διακρίνουμε τα χαρακτηριστικά του λογισμικού που μας προσφέρεται. Όταν κάτι είναι δωρεάν είναι πάντα σίγουρα δελεαστικό, αν και μπορεί να μην είναι πάντα αυτό που ζητάμε. Χωρίς βέβαια αυτό να σημαίνει ότι δεν υπάρχει ποιοτικό δωρεάν λογισμικό. Βέβαια μπορεί για περιπτώσεις μικρών εταιρειών με κάποια εμπειρία σε προγραμματισμό, αυτό να είναι η καλύτερη λύση.
2. Απόφαση ανάμεσα σε self-hosted και cloud computing λογισμικό. Οι μικρότερες εταιρείες με περιορισμένο budget που θέλουν άμεσα έτοιμο λογισμικό, χωρίς έμπειρο εξειδικευμένο προσωπικό, επιλέγουν συνήθως εφαρμογή cloud computing. Εταιρείες με τμήματα IT, με εμπειρία σε θέματα πληροφορικής, διαθέσιμο προσωπικό για υλοποίηση και διαχείριση του λογισμικού αλλά και του υλικού αυτού, μπορούν πιο εύκολα να αποφασίσουν ανάμεσα στα δύο.
3. Προϋπολογισμός. Το λογισμικό help desk κυμαίνεται από \$0 μέχρι και πάνω από \$1000. Πρέπει λοιπόν κάποιος να συντάξει προϋπολογισμό και να μην τον αγνοεί στην απόφασή για αγορά λογισμικού. Θα πρέπει να υπάρχει μεγάλη προσοχή για υπηρεσίες που τυχόν χρεώνονται με το μήνα ή με βάση τους υπαλλήλους που χρησιμοποιούν την εφαρμογή, καθώς και το κόστος που έχει η αρχική και μετέπειτα παραμετροποίηση και συντήρηση.
4. Επιλογή των τεχνικών χαρακτηριστικών και χαρακτηριστικών διαχείρισης που επιθυμούνται. Είναι σημαντικό η εταιρεία να επιλέξει εκείνα τα χαρακτηριστικά (features) που θα καλύπτουν όλες τις λειτουργίες που είναι αναγκαίες για τη διατήρηση ενός επιτυχημένου τμήματος υποστήριξης πελατών. Μέσα από συζητήσεις με τους υπαλλήλους του τμήματος, μπορεί να φτιαχτεί μία λίστα με τα χαρακτηριστικά που ενδιαφέρουν και από την εμπειρία τους έχουν δει ότι τους εξυπηρετούν. Εκτός των τεχνικών χαρακτηριστικών πρέπει να διακρίνονται σωστά και τα χαρακτηριστικά διαχείρισης που είναι επιθυμητά. Ποιος έχει δικαιώματα να ξαναοίξει π.χ. ένα κλειστό δελτίο. Η επιλογή πρέπει να γίνεται πάντα με βάση την πολιτική που έχει υιοθετηθεί, και όχι με βάση πια εφαρμογή προσφέρει περισσότερα χαρακτηριστικά ή είναι πιο φθηνή.
5. Υπολογισμός του πληροφοριακού συστήματος- υποδομή και αποθηκευτικός χώρος. Πολύ σημαντική παράμετρος για την ανάπτυξη της εφαρμογής στο μέλλον. Μια εταιρεία θα πρέπει να υπολογίζει στα κόστη της την υποδομή πάνω στην οποία θα εφαρμόσει το λογισμικό, καθώς και τις πιθανές μελλοντικές αναβαθμίσεις που μπορεί να επέλθουν κυρίως σε αποθηκευτικό χώρο, ιδίως για εφαρμογές που κρατούν μεγάλα άρθρα και πιθανές λύσεις προβλημάτων.
6. Έλεγχος συμβατότητας. Πρέπει η λύση που πρόκειται να επιλεγεί να ελεγχθεί για συμβατότητα με το υπόλοιπο λογισμικό που χρησιμοποιεί η εταιρεία, δηλαδή τους φυλλομετρητές, τη γλώσσα, την υποδομή email, ιδιαίτερα αν κάποια από αυτά ενσωματώνονται στο νέο λογισμικό.
7. Χρήση brand. Από πλευράς marketing, το να παρουσιάζεται το λογισμικό με το δικό σας brand στους πελάτες, είναι πολύ σημαντικό για τη φήμη της εταιρείας. Μπορεί να βρεθεί μία λύση που να εφαρμόζει το λογότυπο, το χρώμα, τη γραμματοσειρά της εταιρείας.
8. Ασφάλεια. Μια πολύ σημαντική παράμετρος για κάθε πληροφοριακό σύστημα. Συνήθως, οι εφαρμογές helpdesk διαχειρίζονται ευαίσθητα προσωπικά δεδομένα τόσο για την εταιρεία όσο και για τους πελάτες της (π.χ. πελατολόγιο, προσωπικό κ.α.). Όταν ένα σύστημα ανταλλάσσει ευαίσθητη πληροφορία, θα πρέπει να επιλεγεί ένα σύστημα

με στοιχεία ασφάλειας όπως χρήση ασφαλούς σύνδεσης (π.χ. SSL). Οι ειδικοί ασφάλειας μπορούν να βοηθήσουν στην αξιολόγηση του πληροφοριακού συστήματος της εταιρείας και αν αυτό χρειάζεται παραμετροποίηση ή αναβάθμιση για τη διασφάλιση της νέας εφαρμογής. Η παραπάνω εργασία ενέχει σημαντικό κόστος και θα πρέπει να συνυπολογιστεί εξ' αρχής για τη λήψη της απόφασης.

9. Web interface. Κάποια συστήματα επιτρέπουν στους πελάτες τους, δημιουργία δελτίων μέσω Διαδικτύου. Πρέπει να μελετηθεί με σύνεση αν κάτι τέτοιο θα ωφελούσε ή θα έβλαπτε την εταιρεία και τους πελάτες.
10. Ανθρώπινο δυναμικό. Πρέπει να λαμβάνεται υπ' όψιν το επίπεδο που υπάρχει στο ανθρώπινο δυναμικό της εταιρείας, τις γνώσεις και την εμπειρία που διαθέτει. Πολλές εφαρμογές έχουν υψηλές απαιτήσεις ανάπτυξης και διατήρησης και κάποιες άλλες όχι.
11. Δοκιμή. Οι περισσότερες εταιρείες διαθέτουν demo, video και βοηθητικό εκπαιδευτικό λογισμικό πριν από την αγορά. Δοκιμάστε τα.

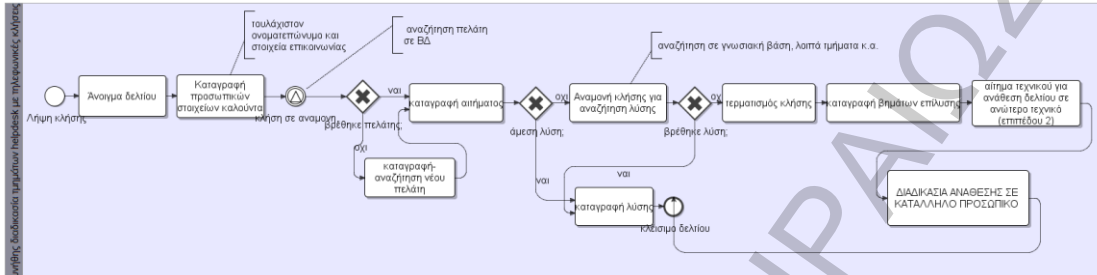
Πότε πρέπει να επενδυθεί από μία εταιρεία ποσό για λογισμικό help desk ;

Δεν υπάρχει γενικός κανόνας για μία εταιρεία ώστε να κάνει μία τέτοια επένδυση. Αν όμως ισχύει για την εταιρεία κάτι από τα παρακάτω [1], τότε θα πρέπει πιθανά να ληφθεί απόφαση για μία επένδυση σε λογισμικό helpdesk.

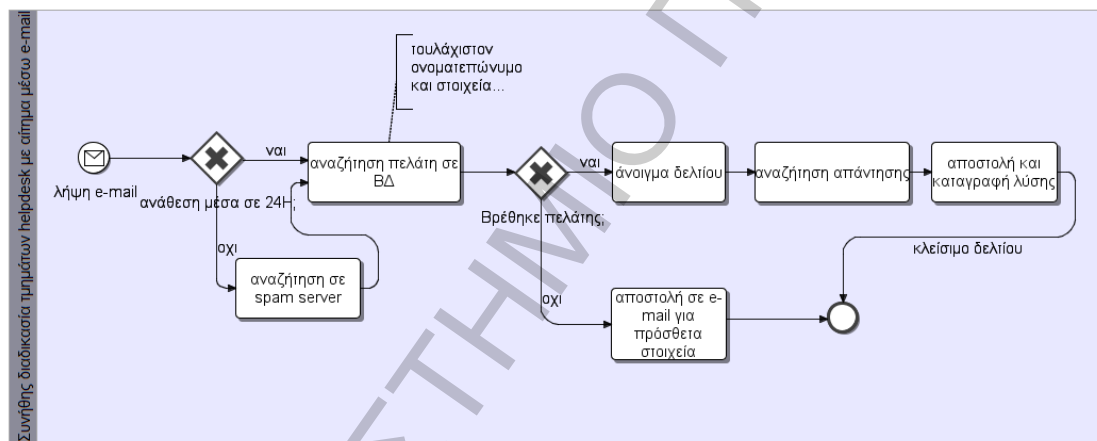
1. Ο αριθμός των πελατών και των διαθέσιμων προϊόντων είναι κάπως μεγάλος και συνεχώς αναπτυσσόμενος.
2. Δεν υπάρχει κεντρική βάση δεδομένων με τα προϊόντα με πρόσβαση από τους πελάτες και τους υπαλλήλους από οπουδήποτε.
3. Οι υπάλληλοι δεν έχουν ένα απλό μέσο με τα στοιχεία των πελατών.
4. Οι υπάλληλοι δυσκολεύονται να βρουν απαντήσεις για τους πελάτες σε εύλογο χρονικό διάστημα. Δεν υπάρχει μία κεντρική βάση δεδομένων με απαντήσεις.
5. Οι λύσεις δίνονται σε μεγάλο χρονικό διάστημα ακόμα και αν το πρόβλημα έχει παρουσιαστεί ξανά.
6. Οι υπάλληλοι πρέπει να ψάχνουν σε πολλές διαφορετικές θέσεις για να προσφέρουν λύσεις (διάφορα αρχεία και βάσεις).
7. Οι πελάτες της εταιρείας μας δεν έχουν τρόπο να λύνουν τα προβλήματά τους, εκτός από το να επικοινωνούν μαζί της.
8. Δεν υπάρχει τρόπος οι υπάλληλοι να ανταλλάσσουν νέα πληροφορία μεταξύ τους.
9. Οι υπάλληλοι δουλεύουν κάποια υπόθεση ταυτόχρονα χωρίς να το γνωρίζουν.
10. Είναι δύσκολο να διαπιστωθεί η ικανοποίηση των πελατών (ή όταν αυτή πέφτει) και η απόδοση των τεχνικών.
11. Δεν υπάρχει μέσο για να διαπιστωθεί πόσα αιτήματα παρελήφθησαν και εξυπηρετήθηκαν, ούτε ποιου τύπου (δηλαδή ποια είναι τα συχνότερα προβλήματα).
12. Όταν η εταιρεία χάνει πελάτες λόγω ανεπάρκειας του τμήματος υποστήριξης.
13. Όταν οι πελάτες της εταιρείας για να βρουν λύση πρέπει να επικοινωνήσουν με πολλά τμήματα.
14. Όταν υπάρχουν παράπονα πελατών για το τεχνικό τμήμα.
15. Όταν χάνονται κλήσεις ή email με αιτήματα.
16. Όταν οι πελάτες της εταιρείας δεν μπορούν να ξέρουν αν το αίτημα τους έχει παραληφθεί και εξυπηρετείται.
17. Όταν οι πελάτες της εταιρείας επικοινωνούν συχνά για ένα μόνο θέμα τους, δυσκολεύοντας επιπλέον τη διαδικασία επίλυσης.
18. Όταν οι τεχνικοί της εταιρείας δυσκολεύονται να εντοπίσουν στο σύστημα τα θέματα πάνω στα οποία δουλεύουν.

1.2 Διαδικασίες Τμημάτων Helpdesk

Για να καταλάβουμε καλύτερα την εργασία των τμημάτων helpdesk ας δούμε σε αυτό το σημείο ένα διάγραμμα μίας συνηθισμένης διαδικασίας υποστήριξης ενός τηλεφωνικού αιτήματος και στη συνέχεια, πώς αυτό διαφοροποιείται όταν λαμβάνεται αίτημα με e-mail (Σχήμα 1, Σχήμα 2). Να δούμε δηλαδή με ποιο τρόπο λειτουργούν οι δύο πιο συχνές διαδικασίες λήψης/ διαχείρισης δελτίων.



Σχήμα 1 Συνήθης Διαδικασία Εξυπηρέτησης Αιτήματος που Εισέρχεται μέσω Τηλεφωνικής Κλήσης.



Σχήμα 2 Συνήθης Διαδικασία Εξυπηρέτησης Αιτήματος που Εισέρχεται μέσω E-mail.

Επιλέξαμε να παρουσιάσουμε κάποιες τυπικές διαδικασίες για τμήματα help desk για να μπορέσουμε να καταλάβουμε σε κάποιο βαθμό πώς λειτουργούν αλλά κυρίως για να δούμε πόσο διαφορετική μπορεί να είναι η αντιμετώπιση κάποιου προβλήματος από ένα τμήμα σε άλλο.

Το μικρό παράδειγμα που είδαμε διακρίνει το αίτημα ανάλογα με τον τρόπο που εισέρχεται στο τμήμα (τηλεφωνικά ή μέσω e-mail). Πολύ διαφορετικές είναι οι διαδικασίες και για διαφορετικούς λόγους. Επηρεάζονται από πολλούς παράγοντες, όπως το μέγεθος της εταιρείας και το πλήθος των υπαλλήλων, το αν η εταιρεία προσφέρει υπηρεσίες ή προϊόντα, αν εξυπηρετεί πιστοποιημένους πελάτες ή το ευρύ κοινό, τα επίπεδα των τεχνικών που διαθέτει η εταιρεία, γενικά τις πολιτικές εξυπηρέτησης χρηστών που έχει υιοθετήσει η εταιρεία, κ.α.

Αφιερώσαμε αυτή την παράγραφο για τη λειτουργία των help desk, επειδή είναι πολύ σημαντικό να καταλάβουμε τη διαφορετικότητα που υπάρχει στα τμήματα help desk μιας και η διαφορετικότητα αυτή σημαίνει παραμετροποίηση για τις εφαρμογές help desk. Αυτή η διαφορετικότητα σε συνδυασμό με τη διευκόλυνση της εργασίας που μπορεί να προσφέρει μια τέτοια εφαρμογή στους χρήστες, είναι οι κύριες αιτίες που επιτρέπουν στους επαγγελματίες τεχνολόγους λογισμικού να βλέπουν αξία στην ανάπτυξη εφαρμογών help desk.

1.3 Σύνοψη επιχειρησιακής μελέτης

Μελετώντας όλα τα παραπάνω, δηλαδή τις υπάρχουσες εφαρμογές, τις ανάγκες και τη λειτουργικότητα αλλά και το εξωτερικό περιβάλλον στο οποίο καλούμαστε να υλοποιήσουμε την εφαρμογή μας, ας δούμε πλέον συνοψίζοντας τους λόγους για τους οποίους επιλέξαμε την ανάπτυξη της συγκεκριμένης εφαρμογής και ας αποφασίσουμε την τεχνολογία που θα χρησιμοποιήσουμε για την υλοποίηση.

Για ποιους λόγους επιλέξαμε λοιπόν την ανάπτυξη μιας εφαρμογής helpdesk; Μπορεί μια νέα εφαρμογή helpdesk να έχει μερίδιο στην αγορά;

Είδαμε ότι το εξωτερικό περιβάλλον που κινούμαστε ευνοεί την ανάπτυξη κατά παραγγελία (custom made) εφαρμογών με ανάγκη διεπαφής στην ελληνική γλώσσα (μικρομεσαίες επιχειρήσεις). Λόγω και της οικονομικής συγκυρίας που περνά η χώρα μας, διατίθενται όλο και λιγότερα χρήματα για επενδύσεις. Τα δύο αυτά στοιχεία δίνουν ευκαιρία στα οικονομικότερα εγχώρια προϊόντα σε σχέση με τα μεγάλα και ακριβότερα πακέτα λογισμικού που είδαμε στα παραδείγματά μας. [4,5,6] Αν λάβουμε επιπλέον υπ' όψιν μας και την τάση που υπάρχει στην ελληνική κοινωνία για την προτίμηση ελληνικών προϊόντων, τότε η εφαρμογή μας έχει εξ' αρχής κάποιο πλεονέκτημα. [8]

Είδαμε ακόμα μέσα από τα παραδείγματά μας (Παρ. 1.1.3, 1.2) ότι η χρήση εφαρμογών Help Desk διευκολύνει πολύ την εργασία των agent (κάτι που φιλοδοξούμε να αποδείξουμε και στην αξιολόγηση της εφαρμογής μας). Μια εφαρμογή helpdesk γίνεται με τη χρήση της απαραίτητο εργαλείο για όποιον εργάζεται με αυτή. Η εξάρτηση αυτή που δημιουργεί, σε συνδυασμό με την αρχική παραμετροποίηση αλλά και την πιθανή συνεχή συντήρησή της, καθιστούν μια τέτοια εφαρμογή κερδοφόρα.

Η πολύχρονη εμπειρία ως εργαζόμενη σε τμήματα τεχνικής υποστήριξης σε μικρούς και μεγάλους οργανισμούς, μου έχει δείξει ότι η χρήση εφαρμογών Help Desk είναι απόλυτα αναγκαία. Η πείρα αυτή μου δίνει την αυτοπεποίθηση ότι γνωρίζοντας εκ των έσω τη λειτουργία τμημάτων τεχνικής υποστήριξης, η ανάλυση θα είναι πλήρης και η υλοποίηση θα έχει γερές βάσεις, που σε συνδυασμό με το στοιχείο της καινοτομίας (multiple system), θα καταστήσουν την εφαρμογή μας ικανή να σταθεί απέναντι σε ανταγωνιστικά προϊόντα.

Ποιες όμως τεχνολογίες πρέπει να επιλέξουμε για την υλοποίησή μας; Όπως είδαμε στην εισαγωγή, οι εφαρμογές Help Desk συνήθως χρησιμοποιούνται από πολλούς ταυτόχρονα, κάτι που μας προσανατολίζει σε ένα δικτυακό πληροφοριακό σύστημα. Η επιλογή που κάναμε ήταν η ανάπτυξη web εφαρμογής με χρήση JSP/Servlet σύμφωνα με το πλαίσιο Struts. Οι λόγοι επιλογής απορρέουν από την ανάλυση και τα πλεονεκτήματα που καταγράφονται αναλυτικά στην επόμενη ενότητα 1.4.

1.4 Χρησιμοποιηθείσες Τεχνολογίες για την Υλοποίηση της Εφαρμογής

1.4.1 Εισαγωγή

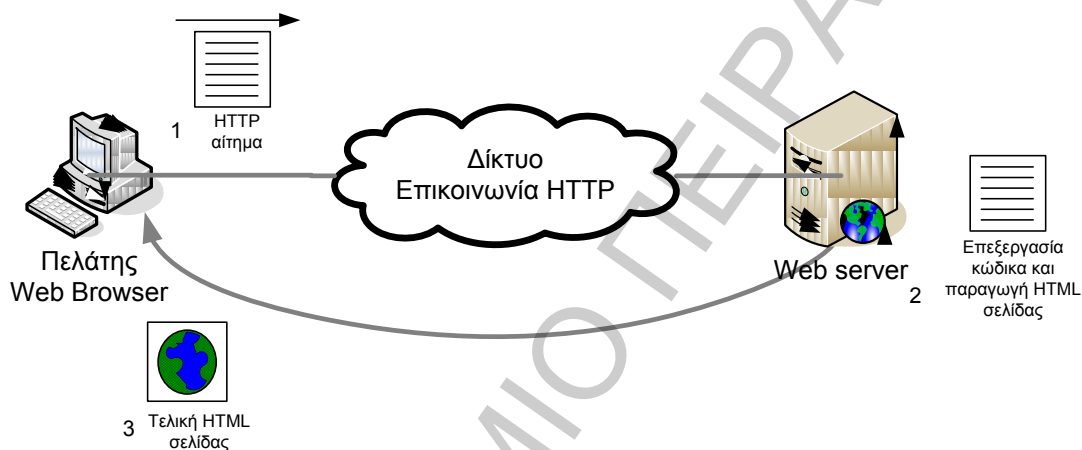
Η ανάπτυξη της προς μελέτη εφαρμογής βασίστηκε κυρίως στις web τεχνολογίες JSP (Java Server Pages) και Servlets σύμφωνα με το μοντέλο Struts. Οι τεχνολογίες JSP/ Servlet εντάσσονται στις γλώσσες παραγωγής δυναμικού περιεχομένου. Οι λόγοι που επιλέγηκαν οι παραπάνω τεχνολογίες για την υλοποίησή μας καλύπτονται αναλυτικά από τα πλεονεκτήματα αυτών στις παραγράφους που ακολουθούν.

Στο σημείο αυτό πριν αναλυθούν περαιτέρω οι βασικές αυτές web τεχνολογίες, κρίθηκε σκόπιμο να δοθεί πρώτα μία σύντομη αναφορά για τον προγραμματισμό στην πλευρά του εξυπηρετητή (server side programming), όπου ανήκουν οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση καθώς επίσης μία αναφορά για την σημαντικότητα της κατασκευής δυναμικών ιστοσελίδων.

1.4.2 Προγραμματισμός στην πλευρά του εξυπηρετητή

Με τον όρο «προγραμματισμός στην πλευρά του εξυπηρετητή» αναφερόμαστε κυρίως στους Εξυπηρετητές Ιστού ή Εξυπηρετητές Παγκόσμιου Ιστού (Web Servers- ο αγγλικός όρος είναι συνηθέστερος). Ένας εξυπηρετητής Ιστού είναι ένα πληροφοριακό σύστημα που δέχεται αιτήσεις HTTP (Hypertext Transfer Protocol requests) και επιστρέφει ως απάντηση αρχεία HTML (response HyperText Markup Language). Οι πιο γνωστοί Εξυπηρετητές Ιστού είναι ο Apache Web Server⁹ της Apache Foundation και ο Internet Information Services –(IIS)¹⁰ της Microsoft.

Οι Εξυπηρετητές Ιστού είναι αυτοί που συνήθως απαντάνε όταν πληκτρολογούμε μια διεύθυνση σε ένα φυλλομετρητή (browser) [9]. Στο σχήμα που ακολουθεί (Σχήμα 3) διακρίνεται η HTTP επικοινωνία μεταξύ του φυλλομετρητή και του Web Server.



Σχήμα 3 Διαδικασία HTTP Αίτησης – Request και HTML Απάντησης - Response.

Οι σελίδες που υπάρχουν στο Διαδίκτυο χωρίζονται σε δύο βασικές κατηγορίες, στις στατικές και στις δυναμικές. Οι στατικές σελίδες δημιουργούνται από στατικά αρχεία των οποίων ο κώδικας βασίζεται συνήθως σε HTML, είναι απλές και μπορούν να φορτωθούν γρήγορα στο μηχάνημα του πελάτη. Οι στατικές ιστοσελίδες έχουν προκαθορισμένο περιεχόμενο, ίδιο για όλους τους πελάτες. Το περιεχόμενο αυτό μπορεί να αλλάξει ρητά μόνο από τον διαχειριστή της σελίδας. [10]

Οι στατικές ιστοσελίδες ήταν επαρκείς και κυρίαρχες στο παρελθόν. Καθώς όμως το Διαδίκτυο επεκτείνεται συνεχώς σε νέες υπηρεσίες πέρα από αυτές που είχαμε συνηθίσει και η νέα γενιά Παγκόσμιου Ιστού (Web 2.0) απαιτεί όλο και περισσότερο την παραγωγή δυναμικού περιεχομένου, δημιουργήθηκε η ανάγκη για την κατασκευή δυναμικών σελίδων.

Οι δυναμικές ιστοσελίδες έχουν μη προκαθορισμένο περιεχόμενο, που το αλλάζουν και το προσαρμόζουν δυναμικά, επεξεργάζονται δεδομένα και λαμβάνουν αποφάσεις με σκοπό την εξυπηρέτηση του χρήστη. Αναλυτικότερα, οι δυναμικές ιστοσελίδες χρησιμοποιούν εφαρμογές που εκτελούνται στην πλευρά του εξυπηρετητή (server side), λαμβάνουν τα δεδομένα που εισάγει ο χρήστης, τα επεξεργάζονται ή ανατρέχουν σε κάποια βάση δεδομένων και στην συνέχεια δημιουργούν μια προσαρμοσμένη ιστοσελίδα, την οποία και βλέπει τελικά ο χρήστης.

Με τις δυναμικές ιστοσελίδες ο χρήστης έχει την δυνατότητα να αλληλεπιδράσει με το περιεχόμενο της ιστοσελίδας. [10] Αυτού του είδους οι ιστοσελίδες δημιουργούνται είτε από δυναμικά αρχεία εξυπηρετητών είτε από στατικά αρχεία τα οποία εκτός από τον κώδικα της HTML, περιλαμβάνουν σενάρια (scripts) βασισμένα σε JavaScript ή VBScript.

⁹ Επίσημος Ιστότοπος: <http://httpd.apache.org/>

¹⁰ Επίσημος Ιστότοπος <http://www.iis.net/>

Πότε όμως είναι επιτακτική η ανάγκη για δυναμικές ιστοσελίδες; Αυτό συμβαίνει: [11]

- Όταν η ιστοσελίδα βασίζεται σε δεδομένα που υποβάλλονται από τον χρήστη. Για παράδειγμα σε ένα ηλεκτρονικό κατάστημα (e-shop), οι σελίδες της διαμόρφωσης παραγγελίας πρέπει να είναι συγκεκριμένες στα ιδιαίτερα αιτήματα του κάθε χρήστη. Δεν είναι γνωστό τι πρέπει να παρουσιαστεί στην ιστοσελίδα μέχρι να διαβαστούν τα δεδομένα που υποβάλλει ο χρήστης και έτσι η ιστοσελίδα δεν μπορεί να έχει κατασκευαστεί από πριν (στατική σελίδα).
- Όταν η ιστοσελίδα παράγεται από δεδομένα που μεταβάλλονται συχνά. Για παράδειγμα οι ιστοσελίδες που παρουσιάζουν δελτία καιρού, τίτλους ειδήσεων ή δεδομένα χρηματιστηρίου πρέπει να κατασκευάζονται δυναμικά και ίσως να επιστρέφουν σε μία προηγούμενη έκδοση σελίδας όταν αυτή η σελίδα εξακολουθεί να ισχύει μέχρι εκείνη την στιγμή του αιτήματος.
- Όταν η ιστοσελίδα χρησιμοποιεί πληροφορίες από βάσεις δεδομένων ή από άλλες πηγές στην πλευρά του εξυπηρετητή (server side). Για παράδειγμα, μία ιστοσελίδα ηλεκτρονικού εμπορίου (e-commerce), χρειάζεται να χρησιμοποιεί τεχνολογίες που εκτελούνται στην πλευρά του εξυπηρετητή (π.χ. Servlet) ώστε να ανατρέχει στην βάση δεδομένων και να κατασκευάζει μία σελίδα, η οποία να καταγράφει - παρουσιάζει την τρέχουσα τιμή και διαθεσιμότητα του κάθε είδους προς πώληση.

1.4.3 Τεχνολογίες JSP / Servlets

1.4.3.1 Java Servlets

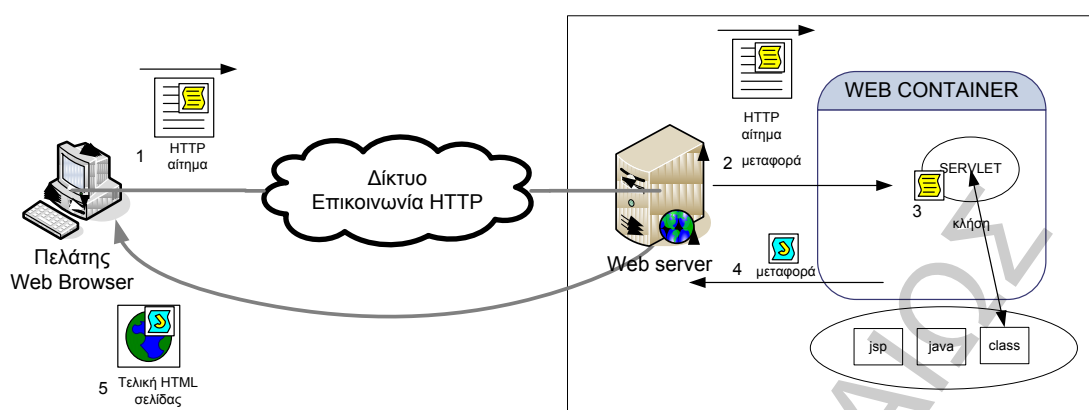
Τα Java Servlet¹¹ (ή απλά Servlet) είναι μια web τεχνολογία που χρησιμοποιείται για την επέκταση και την βελτίωση των δυνατοτήτων των εξυπηρετητών ιστού, δηλαδή των εξυπηρετητών που βασίζονται στο πρωτόκολλο HTTP. Ουσιαστικά είναι κλάσεις της Java, οι οποίες είναι σύμφωνες με το Java Servlet API – ένα πρωτόκολλο με το οποίο η Java ανταποκρίνεται σε HTTP αιτήσεις. Τα Servlet κληρονομούν όλα τα πλεονεκτήματα της γλώσσας προγραμματισμού Java, περιλαμβάνοντας φορητότητα, απόδοση, επαναχρησιμοποίηση και αντοχή. [12]

Ένα Servlet χρησιμοποιείται για την δημιουργία ιστοσελίδων, που το περιεχόμενό τους δεν είναι στατικό, αλλά δυναμικό. Εκτελείται σε έναν εξυπηρετητή ιστού και όχι στα τερματικά των πελατών. Είναι δηλαδή όπως λέμε μια συνιστώσα λογισμικού της πλευράς του εξυπηρετητή (server side component) και χρησιμοποιείται σε εφαρμογές που βασίζονται στο μοντέλο αιτήματος – απόκρισης και σε εφαρμογές που απαιτούν προεπεξεργασία πριν την αποστολή της απόκρισης στον πελάτη.

Τα Servlets επικοινωνούν με έναν εξυπηρετητή ιστού μέσω ενός περιέκτη (Web Container – ο αγγλικός όρος είναι συνηθέστερος). Ο Web Container είναι ουσιαστικά το συστατικό (component) ενός εξυπηρετητή Ιστού που αλληλεπιδρά με τα Servlets και είναι υπεύθυνος για την διαχείριση του κύκλου ζωής (life cycle) των Servlets.

Αυτό που πρακτικά συμβαίνει είναι το εξής: Ο Web Container (π.χ. Tomcat) δέχεται την αίτηση από τον πελάτη και καλεί το αντίστοιχο Servlet (κλάση) περνώντας του και τις παραμέτρους του χρήστη. Το Servlet αυτό (κλάση) εμπεριέχει την λογική που πρέπει να εκτελεστεί με είσοδο τις παραμέτρους του χρήστη, και απαντάει επιστρέφοντας μια σελίδα HTML [9]. Η σελίδα αυτή είναι δυναμική καθώς παρήχθη μόλις ζητήθηκε και με βάση τις εκάστοτε παραμέτρους του χρήστη. Η διαδικασία της επικοινωνίας μεταξύ πελάτη, εξυπηρετητή και Web Container (που καλεί το servlet) διακρίνεται στο παρακάτω σχήμα (Σχήμα 4).

¹¹ Επίσημος Ιστότοπος: <http://www.oracle.com/technetwork/java/javaee/overview/index.html>



Σχήμα 4 Επικοινωνία Client – Web Server – Web Container.

Αναλυτικότερα, οι εργασίες που εκτελεί ένα servlet είναι οι εξής: [11]

- **Ανάγνωση των δεδομένων που υποβάλλονται.** Συνήθως ένας χρήστης εισάγει τα δεδομένα αυτά σε μία HTML φόρμα συμπλήρωσης μιας ιστοσελίδας. Ωστόσο, αυτά τα δεδομένα μπορούν επίσης να προέρχονται από ένα applet¹² ή από ένα HTTP πρόγραμμα πελάτη.
- **Ανάγνωση των δεδομένων της αίτησης HTTP που αποστέλλεται από το φυλλομετρητή.** Τα δεδομένα της αίτησης HTTP μπορούν να περιλαμβάνουν cookies¹³, το όνομα του host¹⁴, πληροφορίες σχετικά με τους τύπους μέσων (media types) και τα compression σχήματα που αντιλαμβάνεται ο φυλλομετρητής κ.α.
- **Δημιουργία αποτελεσμάτων.** Η διαδικασία αυτή συχνά απαιτεί πρόσβαση σε κάποια βάση δεδομένων. Όμως για να είναι εφικτή η επικοινωνία μεταξύ του φυλλομετρητή και της βάσης δεδομένων, εφόσον δεν μπορούν να επικοινωνήσουν απευθείας, χρειάζεται ένα ενδιάμεσο επίπεδο (Web middle layer) ώστε να εξάγαγει τα εισερχόμενα δεδομένα από το HTTP stream, να επικοινωνήσει με την εφαρμογή και να ενσωματώσει τα αποτελέσματα σε ένα αρχείο.
- **Διαμόρφωση των αποτελεσμάτων σε αρχείο για την αποστολή τους στον τελικό χρήστη.** Τα αποτελέσματα μπορούν να αποσταλούν στον τελικό χρήστη μέσω διάφορων μορφών (formats). Μπορεί να είναι κείμενο (HTML, XML), δυαδικά δεδομένα (εικόνες .GIF) ή ακόμα αρχεία συμπιεσμένων τύπων. Ο πιο κοινός τύπος αρχείου, στον οποίο ένα Servlet ενσωματώνει τα αποτελέσματα είναι το HTML αρχείο.
- **Αποστολή των δεδομένων της απάντησης HTTP.** Η αποστολή των δεδομένων της HTTP απάντησης περιλαμβάνει την ενημέρωση του φυλλομετρητή για τον τύπο αρχείου που επιστρέφεται, τον καθορισμό των cookies, τον καθορισμό των απαραίτητων παραμέτρων και άλλα παρόμοια καθήκοντα.

Πιο τεχνικά, όταν ένα Servlet δέχεται ένα αίτημα από τον πελάτη, λαμβάνει δύο αντικείμενα (objects): ένα ServletRequest, που εξασφαλίζει την επικοινωνία από τον πελάτη προς τον εξυπηρετητή και ένα ServletResponse, που εξασφαλίζει την επικοινωνία από το Servlet πίσω στον πελάτη. Και τα δύο παραπάνω αντικείμενα ServletRequest, ServletResponse είναι διεπαφές (interfaces) ορισμένες στο πακέτο javax.servlet.

Υπάρχουν δύο είδη αιτημάτων που μπορούν να υποβληθούν από έναν φυλλομετρητή. Αυτά είναι οι αιτήσεις GET και POST. [9] Στην περίπτωση της αίτησης GET όλα τα προς μεταφορά

¹² Applet: πρόγραμμα γραμμένο σε Java που μπορεί να συμπεριληφθεί σε μία HTML σελίδα, περίπου με τον ίδιο τρόπο που συμπεριλαμβάνεται μία εικόνα. [13]

¹³ Cookie: ένα μικρό κομμάτι δεδομένων που αποστέλλεται από μια ιστοσελίδα και αποθηκεύεται στο φυλλομετρητή του χρήστη ενώ ο χρήστη περιηγείται μέσω του φυλλομετρητή [14]

¹⁴ Οικοδεσπότης υπολογιστής ή απλά οικοδεσπότης. Μπορεί να εννοείται ένα λογισμικό, ή το λειτουργικό σύστημα, ή και ο υπολογιστής ως μηχανήμα, ανάλογα με την περίπτωση

δεδομένα τοποθετούνται στο τέλος του URL, μετά το σύμβολο ? Η μορφή μιας αίτησης GET έχει ως εξής:

`http://<host>:<port><request_path>?<par_1=value_1>&...&<par_n=value_n>`, όπου:

- `host`, είναι η διεύθυνση IP του εξυπηρετητή Ιστού
- `port`, είναι η θύρα στην οποία «ακούει» ο εξυπηρετητής Ιστού
- `request_path`, είναι το μονοπάτι όπου βρίσκεται το servlet μέσα στον εξυπηρετητή
- `par_n`, είναι το όνομα της παραμέτρου `n`
- `value_n`, είναι η τιμή της παραμέτρου `n`
- το σύμβολο `&` διαχωρίζει τις διάφορες παραμέτρους μεταξύ τους

Από την άλλη πλευρά, στην περίπτωση της αίτησης POST τα προς μεταφορά δεδομένα αποστέλλονται σαν επικεφαλίδα ξεχωριστά από το URL, δηλαδή μέσω διαφορετικού stream εξόδου. Ενώ με την αίτηση GET μπορεί να αποσταλεί μόνο ένας συγκεκριμένος αριθμός χαρακτήρων, με την αίτηση POST μπορούν να σταλούν θεωρητικά άπειρα bytes [9].

Τα Servlets χειρίζονται τις δυο αιτήσεις που προαναφέρθηκαν, μέσω μεθόδων που κληρονομούνται από την κλάση `HttpServletRequest`: `doGet(HttpServletRequest, HttpServletResponse)` και `doPost(HttpServletRequest, HttpServletResponse)`.

Όταν μεταδίδονται δεδομένα από τον πελάτη σε ένα Servlet, τότε αυτά αποθηκεύονται σαν ένα string από την κλάση `HttpServletRequest`. Ενώ όταν ένα Servlet θέλει να επικοινωνήσει με αυτόν που το έχει καλέσει, στέλνει πληροφορίες μέσω της κλάσης `HttpServletResponse`. Η πρώτη ενέργεια που πρέπει να γίνει είναι να οριστεί το είδος του περιεχομένου που θα στείλει το Servlet σαν απόκριση με την βοήθεια της μεθόδου `setContentType(String type)`, όπου `type` είναι ο τύπος του περιεχομένου.

Τα πλεονεκτήματά των Servlets έναντι των άλλων τεχνολογιών CGI ¹⁵ (Common Gateway Interface) : [11]

- **Είναι περισσότερο αποδοτικά**, διότι παραμένουν στην μνήμη μεταξύ διαδοχικών κλήσεων. Με τις τεχνολογίες CGI σε κάθε αίτημα που κάνει ο χρήστης ενός φυλλομετρητή για μία ιστοσελίδα, για τον εξυπηρετητή είναι σαν να ξεκινά από την αρχή. Στα Servlets υπάρχει η έννοια της συνόδου (session), με την οποία μπορούμε να κρατήσουμε πληροφορίες που σχετίζονται με την επικοινωνία του χρήστη με το σύστημα που μπορούν να διατηρούνται ανάμεσα σε διαδοχικές κλήσεις (get). Αυτό είναι απαραίτητο σε πολλές εφαρμογές. Ένα απλό παράδειγμα είναι το καλάθι αγορών (shopping cart) σε μία εφαρμογή πώλησης προϊόντων μέσω Internet, στο οποίο ο χρήστης τοποθετεί προϊόντα καθώς πλοηγείται από σελίδα σε σελίδα. Όπως είναι αυτονόητο, το καλάθι αγορών θα πρέπει να συγκρατεί τα περιεχόμενά του ανάμεσα από διαδοχικές αιτήσεις για νέες σελίδες.
- **Είναι εύκολα στην χρήση**. Τα Servlets βασίζονται και αναπτύσσονται στην γλώσσα προγραμματισμού Java και έτσι έχουν μία εκτενή υποδομή που δίνει μεγάλες δυνατότητες. Οι λειτουργίες πολλών διαδικασιών γίνονται αυτόματα, όπως η αποκωδικοποίηση των HTML δεδομένων, η παρακολούθηση συνόδων και ο χειρισμός των cookies. Με τις τεχνολογίες CGI τέτοιες λειτουργίες πρέπει να εκτελούνται χειροκίνητα.
- **Είναι περισσότερο ισχυρά**. Τα Servlets έχουν την δυνατότητα να διατηρούν πληροφορίες από το ένα αίτημα στο άλλο, απλοποιώντας κάποιες τεχνικές, όπως η παρακολούθηση συνόδων και η αποθήκευση προηγούμενων υπολογισμών.
- **Έχουν μεγαλύτερες δυνατότητες φορητότητας**. Όπως έχει ήδη αναφερθεί τα Servlets έχουν αναπτυχθεί στην γλώσσα προγραμματισμού Java και ακολουθούν ένα

¹⁵ CGI (Common Gateway Interface) είναι μια μέθοδος επέκτασης της λειτουργικότητας του Web Server με την δυνατότητα εκτέλεσης προγραμμάτων ή scripts. Μια κοινή χρήση του CGI είναι για παράδειγμα, μια φόρμα παραγγελίας όπου ο χρήστης στέλνει δεδομένα σε ένα CGI script στο server, το script επικοινωνεί με τη βάση δεδομένων και επιστρέφει τα αποτελέσματα πίσω στο χρήστη σε μια σελίδα.

προκαθορισμένο API¹⁶. Έτσι τα Servlets μπορούν να τρέξουν σε σχεδόν όλα τα λειτουργικά συστήματα και εξυπηρετητές.

- **Παρέχουν μεγαλύτερη ασφάλεια.** Τα Servlets παρέχουν μεγαλύτερη ασφάλεια σε σχέση με τα CGI προγράμματα τα οποία συχνά αντιμετωπίζουν διάφορα προβλήματα, όπως η υπερχείλιση μνήμης (buffer overflow).
- **Είναι περισσότερο εφαρμόσιμα.** Τα Servlets σε σύγκριση με τις CGI τεχνολογίες, έχει αποδειχθεί ότι υποστηρίζονται από πολλούς εξυπηρετητές όπως Apache, Oracle, IBM, Sybase, BEA, Macromedia, Caucho, Sun/iPlanet, New Atlanta, ATG, Fujitsu, Lutris, Silverstream, the World Wide Web Consortium (W3C). Επίσης έχουν την δυνατότητα να τρέχουν σχεδόν σε όλα τα λειτουργικά συστήματα (Windows, Unix/Linux, MacOS, VMS, IBM) και αποτελούν την πιο δημοφιλή επιλογή για την ανάπτυξη διαδικτυακών εφαρμογών μιας και χρησιμοποιούνται από την αεροπορική βιομηχανία, το ηλεκτρονικό εμπόριο, το online banking κ.α.

1.4.3.2 Java Server Pages - JSPs

Μία απλουστευμένη άποψη για τα Servlets είναι ότι αυτά είναι Java προγράμματα με ενσωματωμένο HTML κώδικα (μοιάζουν περισσότερο με μία κανονική Java κλάση), ενώ μία επίσης απλουστευμένη άποψη για τις Σελίδες Εξυπηρετητή Java (Java Server Pages - JSP) είναι ότι αυτά είναι HTML ιστοσελίδες με ενσωματωμένο Java κώδικα (μοιάζουν περισσότερο με μία κανονική HTML σελίδα). [11]

Ο κώδικας των Servlets είναι δομημένος σε κλάσεις που γράφονται, μεταγλωττίζονται και εκτελούνται όπως κάθε κλάση Java, ενώ οι σελίδες JSP αποτελούνται από στατικό HTML και δυναμικά παραγόμενο HTML περιεχόμενο, το οποίο είναι διαχωρισμένο με ειδικές ετικέτες (tags) της μορφής `< % java κώδικας % >` [16].

Αν και ακούγονται διαφορετικά, τα Servlets και οι σελίδες JSP είναι κατά μια έννοια το ίδιο πράγμα, επειδή οι σελίδες JSP μεταφράζονται σε Servlets. Οι σελίδες JSP βασίζονται στην τεχνολογία των Java Servlets και μπορούν να θεωρηθούν ως μια υψηλού επιπέδου αφαίρεση (high level abstraction) τους. Οι εργασίες για τις οποίες είναι χρήσιμες το καθένα διαφέρουν: τα Servlets είναι κατάλληλα για εργασίες προσανατολισμένες στην λογική και στην επεξεργασία (oriented toward processing), ενώ οι JSP ιστοσελίδες είναι περισσότερο κατάλληλες για εργασίες προσανατολισμένες στην παρουσίαση (oriented toward presentation). [11]

Το κυριότερο μειονέκτημα με την χρήση των Servlets είναι ότι ο προγραμματιστής θα πρέπει να γράφει αρκετές εντολές `out.print` για να σχηματίσει την απόκριση του Servlet στον τελικό χρήστη (δηλαδή το αποτέλεσμα). Αυτό είναι κουραστικό όταν η απόκριση είναι πολύπλοκη (π.χ. έχει πίνακες ή άλλα σύνθετα HTML στοιχεία). Είναι πιο εύκολο η απόκριση να σχηματίζεται απευθείας γράφοντας HTML και όπου χρειάζεται να χρησιμοποιείται κώδικας Java. Αυτή ακριβώς την ανάγκη έρχονται να καλύψουν οι σελίδες JSP. [16]

Η τεχνολογία JSP παρέχει σημαντικές δυνατότητες διαχωρισμού του προγράμματος από το περιεχόμενο της ιστοσελίδας. Η τεχνολογία JSP διαχωρίζει την διεπαφή χρήστη από την δημιουργία περιεχομένου, επιτρέποντας έτσι στους σχεδιαστές να αλλάζουν την συνολική εμφάνιση της σελίδας χωρίς να μεταβάλλεται το βασικό δυναμικό περιεχόμενο της. [17]

Αυτό που πρακτικά συμβαίνει είναι το εξής: Τα αρχεία JSP είναι αποθηκευμένα στον εξυπηρετητή Ιστού, ο οποίος μπορεί να συνδέεται και με έναν εξυπηρετητή βάσεων δεδομένων. Για να μπορέσει ο εξυπηρετητής Ιστού να εκτελέσει προγράμματα JSP παρέχοντας δυναμικό περιεχόμενο στους πελάτες, πρέπει να μπορεί να υποστηρίζει την τεχνολογία JSP. Αυτό επιτυγχάνεται με ένα περιέκτη JSP (JSP Container) ή αλλιώς εξυπηρετητή εφαρμογών (application server).¹⁷ Ο ρόλος του JSP Container είναι να διερμηνεύει (parsing) τις ετικέτες JSP (tags), να παράγει το δυναμικό περιεχόμενο και να το στέλνει πίσω στον πελάτη ως HTML σελίδα.

¹⁶ Application programming interface (API) : Μία προδιαγραφή που έχει ως σκοπό να χρησιμοποιηθεί ως διεπαφή από τμήματα λογισμικού ώστε αυτά να μπορέσουν να επικοινωνούν μεταξύ τους [15]

¹⁷ Για την υλοποίηση μας χρησιμοποιήθηκε ο Netbeans, για την εκτέλεση του JSP κώδικα χρησιμοποιήθηκε ο ενσωματωμένος web container Tomcat

Η διαδικασία που ακολουθείται όταν ένας πελάτης ζητήσει μία JSP σελίδα είναι η εξής: [18]

1. Ο φυλλομετρητής του πελάτη στέλνει ένα αίτημα μέσω του διαδικτύου για μια σελίδα JSP.
2. Το αίτημα φθάνει στον εξυπηρετητή Ιστού.
3. Ο εξυπηρετητής Ιστού αναγνωρίζει ότι ζητήθηκε ένα ειδικό αρχείο (.jsp) και περνάει το JSP αρχείο στο JSP Servlet Engine (δηλαδή στον εξυπηρετητή εφαρμογών).
4. Αν το JSP αρχείο ζητείται για πρώτη φορά τότε διερμηνεύεται (parsing), διαφορετικά η διαδικασία προχωράει στο βήμα 7.
5. Το επόμενο βήμα είναι να μεταγλωττιστεί το JSP αρχείο σε Servlet (όλα τα HTML στοιχεία μετατρέπονται σε out.println δηλώσεις).
6. Ο πηγαίος κώδικας του Servlet που προκύπτει μετατρέπεται σε αρχείο .class.
7. Το Servlet εκτελείται, όπου καλούνται οι κατάλληλες μέθοδοι (init, service).
8. Ο HTML κώδικας που παράγεται ως έξοδος (output) του Servlet, αποστέλλεται πίσω στον φυλλομετρητή του client μέσω του διαδικτύου.
9. Ο HTML κώδικας φθάνει στον φυλλομετρητή του πελάτη και προβάλλεται.

Κλείνοντας, για τις JSP σελίδες, ας δούμε κάποια βασικά χαρακτηριστικά της τεχνολογίας αυτής και συγκεκριμένα κάποιες ετικέτες (tags) και τις προκαθορισμένες μεταβλητές, οι οποίες αναλαμβάνουν σημαντικό ρόλο κατά την εκτέλεση του προγράμματος. Οι ετικέτες ομαδοποιούν συγκεκριμένες λειτουργίες του JSP κώδικα, επιτρέποντας την ευέλικτη ενσωμάτωσή του στο αρχείο HTML. Οι προκαθορισμένες μεταβλητές δημιουργούνται αυτόματα στην μέθοδο εξυπηρέτησης (Post, Get) μιας JSP σελίδας, είναι διαθέσιμες για τους προγραμματιστές JSP σελίδων και μπορούν να χρησιμοποιηθούν μέσα από τμήματα κώδικα (scriptlets) και εκφράσεις (expressions). [9, 16]

Ετικέτες (tags)

Οι ετικέτες που χρησιμοποιούνται από την τεχνολογία JSP είναι παρόμοιες με τις ετικέτες της HTML και κατηγοριοποιούνται ανάλογα με τη λειτουργία τους.

- **Εκφράσεις (Expressions).** Η ετικέτα αυτή επιτρέπει την ενσωμάτωση Java εκφράσεων, οι οποίες εκτιμώνται και αποστέλλονται στον τελικό χρήστη κάθε φορά που καλείται η σελίδα. Οι Java εκφράσεις είναι της μορφής `<%= Java Value %>` και σημειώνεται ότι στο τέλος του κώδικα που τοποθετείται ανάμεσα σε αυτές τις σημάνσεις δεν προστίθεται το ερωτηματικό (;).
- **Δηλώσεις (Declarations).** Η ετικέτα αυτή χρησιμοποιείται για τον ορισμό μεταβλητών ή μεθόδων αντικειμένων, όταν η σελίδα μετατρέπεται σε Servlet. Οι δηλώσεις είναι της μορφής `<%! Field Definition %>` ή `<%! Method Definition %>` και σημειώνεται ότι ο κώδικας που τοποθετείται ανάμεσα σε αυτές τις σημάνσεις πρέπει να ακολουθείται με ερωτηματικό (;).
- **Οδηγίες (Directives).** Η ετικέτα αυτή παρέχει ειδικές πληροφορίες σχετικά με την JSP σελίδα στο JSP Servlet Engine. Τέτοιες πληροφορίες μπορεί να είναι για παράδειγμα υψηλού επιπέδου πληροφορία σχετικά με την δομή του Servlet κώδικα. Οι οδηγίες είναι της μορφής `<%@ directive att="val" %>`.
- **Τμήματα κώδικα (Scriptlets).** Η ετικέτα αυτή επιτρέπει την ενσωμάτωση τμημάτων Java κώδικα. Τα τμήματα κώδικα είναι της μορφής `<% Java Statement %>` και σημειώνεται ότι ο κώδικας που τοποθετείται ανάμεσα σε αυτές τις σημάνσεις πρέπει να ακολουθείται με ερωτηματικό (;).

Προκαθορισμένες μεταβλητές

- **Request.** Αυτή η μεταβλητή είναι το αντίστοιχο του HttpServletRequest αντικειμένου που είναι παράμετρος στην μέθοδο εξυπηρέτησης της αίτησης του χρήστη (π.χ. doGet ή doPost) για τα Servlets.
- **Response.** Αυτή η μεταβλητή είναι το αντίστοιχο του HttpServletResponse αντικειμένου που είναι παράμετρος στην μέθοδο εξυπηρέτησης της αίτησης του χρήστη (π.χ. doGet ή doPost) για τα Servlets.

- **Out.** Αυτή η μεταβλητή είναι το αντίστοιχο του PrintWriter που στα Servlets ανακατούμε καλώντας την μέθοδο getWriter() στο αντικείμενο HttpServletResponse.
- **Session.** Αυτό είναι ένα προ-δημιουργημένο αντικείμενο τύπου HttpSession. Παρέχει τις μεθόδους getAttribute και setAttribute για την ανάκτηση και αποθήκευση ιδιοτήτων αντίστοιχα στο session του χρήστη με τον δικτυακό τόπο.

1.4.4 Αρχιτεκτονική JSP / Servlets Model View Controller (MVC)

Η αρχιτεκτονική Model View Controller (MVC ή αλλιώς Model 2), αποτελεί το σχεδιαστικό πρότυπο για τη χρήση των δύο τεχνολογιών που είδαμε στις παραπάνω ενότητες (JSP/Servlets), επιτρέπει σε κάθε τεχνολογία να επικεντρωθεί σε αυτό που υπερέχει, και συχνά χρησιμοποιείται από εφαρμογές που χρειάζονται την δυνατότητα να διατηρούν πολλαπλές διαφορετικές παρουσιάσεις των ίδιων δεδομένων. [19, 20]

Το συγκεκριμένο σχεδιαστικό πρότυπο επιτρέπει το διαχωρισμό της λογικής από την παρουσίαση στο υπό ανάπτυξη σύστημα με αποτέλεσμα να προσφέρει μεγαλύτερη ευελιξία στην ανάπτυξη και στη συντήρηση του συστήματος και ευκολότερο έλεγχο των επιμέρους υποσυστημάτων.

Ρόλοι

Οι τρεις ρόλοι που ορίζονται από το MVC περιγράφονται παρακάτω:

- **Μοντέλα (Models).** Τα μοντέλα διατηρούν τα δεδομένα, τα οποία χρησιμοποιεί και επεξεργάζεται η εφαρμογή και προσθέτουν λογική στην πληροφορία των εκάστοτε δεδομένων. Όταν ένα μοντέλο μεταβάλλει την κατάστασή του, τότε ενημερώνει τις όψεις (views) που σχετίζονται με αυτό ώστε να ανανεωθούν με τη νέα κατάσταση.
- **Όψεις (Views).** Οι όψεις αναπαριστούν όλα ή μέρος των δεδομένων σε μορφή κατανοητή από το χρήστη και προσαρμοσμένη στο μέσο απεικόνισης. Συνήθως παρέχονται διαφορετικές όψεις για κάθε μοντέλο, οι οποίες χρησιμοποιούνται ανάλογα με την εκάστοτε ανάγκη παρουσίασης της πληροφορίας του μοντέλου.
- **Ελεγκτές (Controllers).** Οι ελεγκτές δέχονται και διαχειρίζονται τις εντολές του χρήστη, οι οποίες επηρεάζουν τα μοντέλα ή τις όψεις. Αποτελούν τον συνδετικό κρίκο του συστήματος και είναι υπεύθυνοι για τη διεκπεραίωση των εντολών του χρήστη της εφαρμογής. Η συνήθης λειτουργία τους είναι η αναγνώριση των εντολών του χρήστη, η άντληση και επεξεργασία της πληροφορίας από τα μοντέλα βάσει των εντολών του χρήστη και η παρουσίαση της ζητούμενης πληροφορίας μέσω της κατάλληλης όψης.

Σημειώνεται ότι οι ελεγκτές έχουν πρόσβαση τόσο στις όψεις, τις οποίες δίνουν προς απεικόνιση όσο και στα μοντέλα, από τα οποία αντλούν πληροφορίες ενώ και οι όψεις με τη σειρά τους έχουν τη δυνατότητα άντλησης πληροφοριών από τα μοντέλα.

Τρόπος λειτουργίας- Αρχιτεκτονική

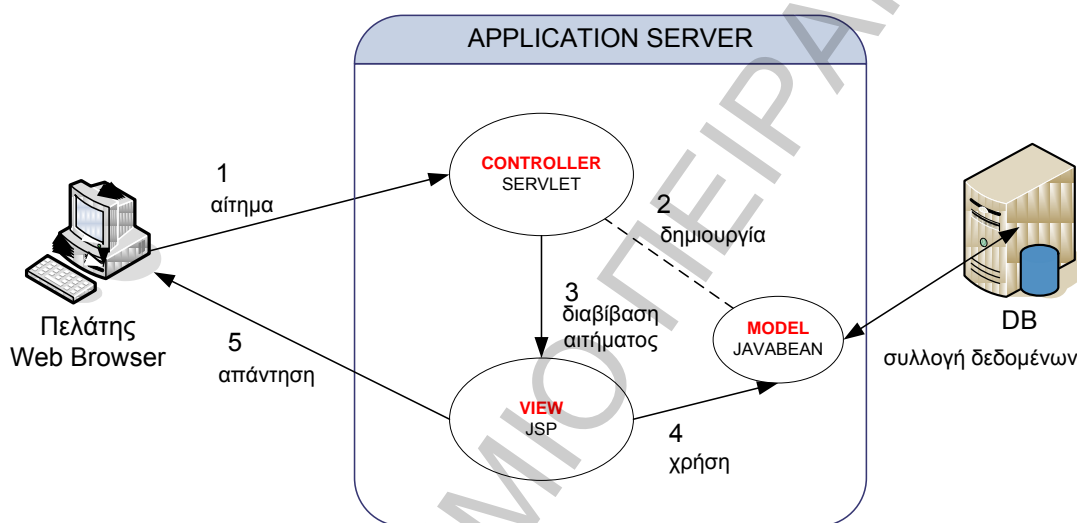
Ένα γεγονός συνήθως εμφανίζεται και προκαλεί τον ελεγκτή να αλλάξει το μοντέλο ή την όψη ή και τα δύο. Όταν ένας ελεγκτής αλλάζει τα δεδομένα ή τις ιδιότητες κάποιου μοντέλου, τότε ανανεώνονται αυτόματα όλες οι όψεις που εξαρτώνται από αυτό το μοντέλο. Με τον ίδιο τρόπο όταν ένας ελεγκτής αλλάζει μια όψη, για παράδειγμα με το να αποκαλύπτει περιοχές που μέχρι πρότινος δεν ήταν ορατές, τότε η όψη αντλεί δεδομένα από το μοντέλο για να ανανεωθεί [20].

Όταν για την υλοποίηση μιας εφαρμογής με δυναμικό περιεχόμενο χρησιμοποιείται ο συνδυασμός των τεχνολογιών JSP και Servlets (όπως συμβαίνει και στην περίπτωση μας) τότε μπορούμε να πούμε ότι το πρότυπο MVC διαμορφώνεται ως εξής:

Το αίτημα διατηρείται από ένα Servlet. Το Servlet λειτουργεί ως ελεγκτής· επικαλείται την επιχειρησιακή λογική (business logic) και την επεξεργασία δεδομένων κώδικα. Είναι υπεύθυνο για την επεξεργασία του αιτήματος και την δημιουργία βοηθητικών κλάσεων και Java Beans, τα οποία εξαρτώνται από τις ενέργειες του χρήστη.

Οι κλάσεις αυτές και τα Java Beans αποτελούν το μοντέλο. Στην συνέχεια το Servlet αποφασίζει ποια σελίδα JSP είναι κατάλληλη για να παρουσιάσει αυτά τα αποτελέσματα και διαβιβάζει το αίτημα σε αυτή. Επομένως, η JSP σελίδα έχει τον ρόλο της όψης. Σημειώνεται ότι δεν υπάρχει καμία επεξεργασία της λογικής από την σελίδα JSP. Αυτή είναι υπεύθυνη μονό α) για την ανάκτηση των κλάσεων ή/και των Java Beans, τα οποία έχουν δημιουργηθεί προηγουμένως από το Servlet και β) για να εξάγει το δυναμικό περιεχόμενο από το Servlet με σκοπό την εισαγωγή του στα στατικά πρότυπα – templates [19, 21].

Η προσέγγιση αυτή έχει ως αποτέλεσμα τον διαχωρισμό της παρουσίασης από το περιεχόμενο, οδηγώντας στην σαφή οριοθέτηση των ρόλων και των υπευθυνότητων για τους προγραμματιστές και τους σχεδιαστές μιας ομάδας υλοποίησης. Η αρχιτεκτονική JSP / Servlets MVC αναπαριστάται στο ακόλουθο σχήμα (Σχήμα 5). [21]



Σχήμα 5 Αρχιτεκτονική JSP / Servlets Model View Controller.

Πλεονεκτήματα

Η υιοθέτηση της αρχιτεκτονικής αυτής, παρέχει κάποια πλεονεκτήματα τα οποία συνοψίζονται ως εξής:

- Σαφήνεια σχεδιασμού. Οι μέθοδοι που ακολουθούνται στο αρχιτεκτονικό αυτό μοντέλο επιτρέπουν τον έλεγχο της συμπεριφοράς του προγράμματος ενώ επίσης καθιστούν όλο το πρόγραμμα ευκολότερο στην υλοποίηση και συντήρηση του.
- Αποδοτικός σχεδιασμός. Ο τρόπος με τον οποίο σχεδιάζεται το πρόγραμμα επιτρέπει τόσο στον χρήστη όσο και στον προγραμματιστή να αντικαταστήσει τα συστατικά της εφαρμογής κατά βούληση. Οι αλλαγές σε κάποιο μέρος του προγράμματος δεν συνδέονται με άλλα μέρη, διευκολύνοντας έτσι την αποσφαλμάτωση.
- Πολλαπλές όψεις. Με την αρχιτεκτονική αυτή, η εφαρμογή μπορεί να εμφανίσει την κατάσταση ενός μοντέλου με πολλούς διαφορετικούς τρόπους. Όλες οι όψεις χρησιμοποιούν τα ίδια ακριβώς δεδομένα αλλά τα παρουσιάζουν με τελείως διαφορετικό τρόπο.
- Ευκολία ανάπτυξης. Τόσο οι ελεγκτές όσο και οι όψεις μπορούν να αναπτύσσονται καθ' όλη την διάρκεια ανάπτυξης του μοντέλου. Όλες οι παλιές όψεις και οι παλιόι ελεγκτές μπορούν να χρησιμοποιούνται όσο διατηρείται η κοινή διεπαφή.

Στα πλαίσια ανάπτυξης της εφαρμογής HelpDeskAssistant υιοθετήσαμε την αρχιτεκτονική MVC για όλους τους λόγους που αναφέρθηκαν παραπάνω, για να εξασφαλίσουμε την

επεκτασιμότητα της εφαρμογής. Πιο συγκεκριμένα, υλοποιήσαμε την αρχιτεκτονική MVC μέσα από την τεχνική Struts που αναλύουμε στην επόμενη παράγραφο.

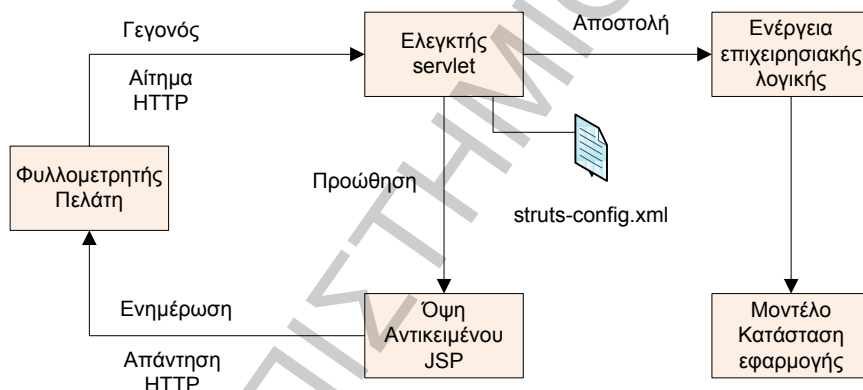
1.4.5 Apache Struts

Με τον όρο Struts¹⁸ αναφερόμαστε σε ένα πλαίσιο (framework)¹⁹ ανάπτυξης δικτυακών εφαρμογών Java που αναπτύσσεται από την Apache Software Foundation²⁰ (βρίσκεται κάτω από την άδεια Apache License), και το οποίο χρησιμοποιεί και επεκτείνει το Java Servlet API. [22]

Σκοπός του Struts είναι να ενθαρρύνει τους προγραμματιστές να υιοθετήσουν την αρχιτεκτονική MVC. Με τη χρήση του Struts στοχεύουμε στο διαχωρισμό του μοντέλου από τις όψεις και τους ελεγκτές (§ 1.4.3). Για να γίνει αυτό το Struts παρέχει ένα ελεγκτή (το Servlet ActionServlet) και διευκολύνει τον προγραμματιστή στη συγγραφή προτύπων για τις όψεις, ενώ με τη χρήση ενός αρχείου xml (struts-config.xml) γίνεται η σύνδεση ανάμεσα σε μοντέλο, όψεις και ελεγκτές.

Δουλειά του προγραμματιστή είναι να δημιουργεί αυτό το κεντρικό xml αρχείο και να γράφει τον κώδικα για το μοντέλο.

Ο τρόπος λειτουργίας των προγραμμάτων που αναπτύσσονται με Struts είναι ο εξής: Ο πελάτης (client), στέλνει αιτήματα στον ελεγκτή με τη μορφή Ενεργειών (όπως ορίζονται στο κεντρικό xml configuration αρχείο). Όταν ο ελεγκτής λάβει ένα αίτημα καλεί την αντίστοιχη Ενέργεια (κλάση) που αλληλεπιδρά με το σχετικό κομμάτι κώδικα του μοντέλου. Στη συνέχεια το μοντέλο επιστρέφει ένα string (ActionForward) το οποίο λέει στον ελεγκτή ποια σελίδα να στείλει στον πελάτη. Η πληροφορία ανάμεσα σε μοντέλο και όψεις γίνεται με JavaBeans, των οποίων το περιεχόμενο διαβάζεται και εγγράφεται από το επίπεδο παρουσίασης με τη βοήθεια μίας βιβλιοθήκης με tags.



Σχήμα 6 Το πλαίσιο Struts με βάση το MVC

Πηγή IBM. Best practices for Struts development [23]

Όπως είναι αναμενόμενο το Struts δεν είναι το μοναδικό πλαίσιο για την ανάπτυξη εφαρμογών Java. Υπάρχουν πολλές επιλογές για πλαίσια που ακολουθούν ή όχι το μοντέλο MVC, όπως τα Spring MVC, Stripes, JSF, Tapestry, Wicket κ.α.[22]. Γιατί λοιπόν επιλέγουμε το Struts για την ανάπτυξη της εφαρμογής μας;

Το Struts είναι πλέον ένα καλά τεκμηριωμένο, ώριμο και δημοφιλές πλαίσιο για την ανάπτυξη εμπρόσθιων τμημάτων εφαρμογών Java. Το εγγενές πλεονέκτημα του Struts είναι ότι

¹⁸ Επίσημος Ιστότοπος: <http://struts.apache.org>

¹⁹ Πλαίσιο (framework): μία συλλογή από κλάσεις και διεπαφές που συνεργάζονται για να λύσουν ένα προγραμματιστικό πρόβλημα.

²⁰ Επίσημος Ιστότοπος: www.apache.org

παρέχει μια έτοιμη υλοποίηση του προτύπου Εμπρόσθιου Ελεγκτή (Front Controller pattern)²¹ [24, 25].

Άλλα πλεονεκτήματα [26, 27] που αναφέρονται σε σχέση με άλλες τεχνικές είναι:

- Κεντρικοποιημένη διαμόρφωση με βάση αρχείο. Πολλές από τις τιμές του Struts αναπαρίστανται σε XML ή σε αρχεία ιδιοτήτων με αποτέλεσμα να μπορούν να γίνονται πολλές αλλαγές χωρίς τροποποίηση και επαναμεταγλώττιση του Java κώδικα.
- Beans φόρμας. Το Apache Struts επεκτείνει τη δυνατότητα αυτόματης δημιουργίας συστατικών JavaBean με βάση παραμέτρους που περιέχονται σε εισερχόμενα αιτήματα στα Servlet, κάτι που δεν είναι διαθέσιμο μέσω του τυπικού API.
- Bean Tags. Το Struts παρέχει ένα σύνολο JSP tags που επιτρέπουν την εύκολη εξαγωγή των ιδιοτήτων των συστατικών JavaBeans .
- HTML Tags. Το Struts παρέχει ένα σύνολο JSP tags για τη δημιουργία φορμών HTML που συσχετίζονται με συστατικά JavaBeans.
- Το Struts έχει ενσωματωμένες δυνατότητες που ελέγχουν τις τιμές των φορμών, ώστε αυτές να έχουν την απαιτούμενη μορφή (format).
- Το Struts ενθαρρύνει τη συνεπή χρήση του MVC σε ολόκληρη την εφαρμογή.

Είναι επίσης σημαντικό, λόγω της γενικότερης τάσης εξωστρέφειας που υπάρχει στις μέρες μας ότι το Struts υποστηρίζει τη διεθνοποίηση (internationalization)²² του λογισμικού με δικτυακές φόρμες και περιλαμβάνει ένα μηχανισμό που ονομάζει πλακίδιο (Tile) που επιτρέπει στο επίπεδο παρουσίασης να αποτελείται από ανεξάρτητα συστατικά μέρη. [22]

Ολοκληρώνοντας · Servlets και Struts είναι δύο συσχετιζόμενες αλλά ξεχωριστές τεχνολογίες που χρησιμοποιούνται για τη δημιουργία δυναμικών ιστοσελίδων με τη χρήση της γλώσσας Java. Μιας και οι δύο αυτές τεχνολογίες αλληλεπικαλύπτονται είναι δύσκολο να ξεχωρίσουμε την αρχή και το τέλος της καθεμιάς. [27]

1.4.6 Συμπληρωματικές Τεχνολογίες

Εκτός από τις βασικές τεχνολογίες JSP/Servlets και Struts, για την ανάπτυξη του συστήματος χρησιμοποιήθηκαν ως συμπληρωματικές οι τεχνολογίες JavaScript, jQuery και AJAX.

Η διαχείριση των δεδομένων και η εμφάνισή τους σε δυναμικούς πίνακες, στηρίχτηκε σε ένα jQuery plug-in που λέγεται DataTables²³. Το συγκεκριμένο plug-in μας φάνηκε ιδιαίτερα χρήσιμο καθώς πέραν της εμφάνισης των δεδομένων σε πίνακες με δυναμικά δεδομένα, συμπεριλάμβανε κάποιες επιπλέον λειτουργικότητες, όπως η σελιδοποίηση και η αναζήτηση στους πίνακες.

Σε όλες τις περιπτώσεις υποβολής μιας φόρμας, χρησιμοποιήθηκαν κατάλληλα scripts φτιαγμένα σε JavaScript, για το έλεγχο των πεδίων της φόρμας και τις αντίστοιχες ειδοποιήσεις στον χρήστη.

Τέλος, κάποιες φορές χρειάστηκε να χρησιμοποιήσουμε την τεχνολογία AJAX, όπως για παράδειγμα στις περιπτώσεις που ο χρήστης επιλέγει ένα στοιχείο σε ένα drop down μενού και βάσει αυτής της επιλογής «συμπληρώνεται» με τα κατάλληλα στοιχεία ένα άλλο drop down μενού.

Κρίνουμε λοιπόν απαραίτητο να γίνει μία σύντομη αναφορά για κάθε μία από αυτές τις τεχνολογίες.

²¹ Προτύπου Εμπρόσθιου Ελεγκτή (Front Controller pattern): πρότυπο σχεδίασης λογισμικού που αναφέρεται σε πολλούς σχετικούς καταλόγους. Το πρότυπο αυτό σχετίζεται με τη σχεδίαση ιστοσελίδων. «Παρέχει ένα κεντρικοποιημένο σημείο εισόδου για το χειρισμό αιτημάτων» [28]

²² Διεθνοποίηση (internationalization) στην τεχνολογία λογισμικού είναι η προσαρμογή ενός προγράμματος σε διαφορετικές γλώσσες, τοπικές διαφοροποιήσεις και τεχνικές απαιτήσεις μιας στοχευόμενης αγοράς. Γενικότερα η τάση να κάνει κάτι διεθνές. [πηγή: en.wikipedia.org/wiki/Internationalization_and_localization]

²³ Περισσότερες πληροφορίες γι' αυτό: <http://www.datatables.net> .

Η JavaScript είναι μια αντικειμενοστρεφής scripting γλώσσα (object-oriented scripting language) που έχει ως σκοπό την παραγωγή δυναμικού περιεχομένου σε HTML ιστοσελίδες. Αποτελεί μια γλώσσα που μπορεί να υποστηρίξει αντικειμενοστρεφή (object – oriented), επιτακτικό (imperative) και λειτουργικό (functional) προγραμματισμό. [29] Χρησιμοποιείται για να δώσει πρόσβαση σε αντικείμενα, μεταξύ του πελάτη και λοιπών εφαρμογών. Η JavaScript δεν είναι ίδια με την γλώσσα προγραμματισμού Java. Ο κώδικας της JavaScript ενσωματώνεται μέσα στον κώδικα της HTML μέσω ειδικών ετικετών (tags) της μορφής: <script> και </script> και μπορεί να εκτελεστεί αμέσως ή όταν συμβαίνει ένα γεγονός (event). Δεν γίνεται μεταγλώττιση του κώδικα της JavaScript (compilation), αλλά απαιτείται ο φυλλομετρητής να υποστηρίζει την JavaScript.

Συνοψίζοντας για την JavaScript, δίνονται επιγραμματικά οι δυνατότητές της: [30]

- Αποτελεί ένα προγραμματιστικό εργαλείο για τους HTML σχεδιαστές. Εφόσον είναι μία scripting γλώσσα έχει απλή σύνταξη και διευκολύνει τους HTML σχεδιαστές που δεν είναι πολύ εξοικειωμένοι με τον προγραμματισμό.
- Μπορεί να αντιδράσει σε διάφορα γεγονότα (events). Ένα JavaScript μπορεί να ρυθμιστεί να εκτελείται όταν συμβαίνει ένα γεγονός, όπως για παράδειγμα όταν ένας χρήστης επιλέγει ένα HTML στοιχείο.
- Μπορεί να διαβάσει και να γράψει HTML στοιχεία. Ένα JavaScript έχει την δυνατότητα να διαβάσει και να αλλάξει το περιεχόμενο ενός HTML στοιχείου.
- Μπορεί να χρησιμοποιηθεί για επαλήθευση δεδομένων. Ένα JavaScript μπορεί να χρησιμοποιηθεί για να επαληθευθεί μια φόρμα δεδομένων, πριν αυτή υποβληθεί στον επεξεργαστή (server). Αυτό έχει ως αποτέλεσμα την προστασία του επεξεργαστή από επιπλέον επεξεργασία.
- Μπορεί να χρησιμοποιηθεί για την ανίχνευση του φυλλομετρητή κάποιου επισκέπτη.
- Μπορεί να χρησιμοποιηθεί για να δημιουργήσει cookies. Ένα JavaScript μπορεί να χρησιμοποιηθεί για να αποθηκεύει αλλά και να ανακτά πληροφορία από τον υπολογιστή του επισκέπτη.

Στην συνέχεια όσον αφορά την τεχνολογία jQuery. Αυτή δεν αναφέρεται σε scripting γλώσσα αλλά σχετίζεται άμεσα με την scripting γλώσσα JavaScript, όπου αποτελεί μία βιβλιοθήκη της που δίνει έμφαση στην αλληλεπίδραση μεταξύ JavaScript και HTML. Είναι δηλαδή μία συλλογή από έτοιμες ρουτίνες γραμμένες σε JavaScript, οι οποίες μπορούν να χρησιμοποιηθούν για να εκτελέσουν συγκεκριμένες λειτουργίες.

Η βιβλιοθήκη jQuery είναι δωρεάν, ανοιχτού κώδικα λογισμικό και η σύνταξή της έχει σχεδιαστεί ώστε να είναι ευκολότερη η περιήγηση σε ένα αρχείο, επιλέγοντας στοιχεία DOM, δημιουργώντας εφέ, διαχειρίζοντας γεγονότα –events και αναπτύσσοντας εφαρμογές AJAX. Η jQuery παρέχει επίσης την δυνατότητα στους προγραμματιστές να δημιουργήσουν τα δικά τους plug-ins στην βιβλιοθήκη, με τέτοιο τρόπο ώστε να την επεκτείνουν. Με την παροχή αυτής της δυνατότητας, οι προγραμματιστές μπορούν να δημιουργήσουν χαμηλού επιπέδου αλληλεπιδράσεις προχωρημένα εφέ και υψηλού επιπέδου widgets.[31]

Επιπλέον είναι σημαντικό να προστεθεί ότι η βιβλιοθήκη jQuery απλοποιεί την εκμάθηση και την χρήση της γλώσσας JavaScript που χρησιμοποιείται στην δημιουργία ιστοσελίδων και web εφαρμογών, ενώ επίσης εξαλείφει και τα προβλήματα που συνεπάγονται την χρήση της JavaScript, όπως το πρόβλημα συμβατότητάς της με τους διάφορους φυλλομετρητές

Ολοκληρώνοντας για την βιβλιοθήκη jQuery της JavaScript, δίνονται περιληπτικά οι δυνατότητες που παρέχει η χρήση της : [32]

- Εύκολη προσθήκη εφέ στα στοιχεία – elements.
- Δημιουργία XML αιτημάτων – requests. Τα XML αυτά αιτήματα – requests, χρησιμοποιούνται από την JavaScript για να ζητούν πρόσθετες πληροφορίες από τους εξυπηρετητές ιστού, χωρίς να χρειάζεται να ανανεωθεί ξανά η σελίδα.
- Χειρισμός των DOM. Εύκολη προσθήκη, αφαίρεση και αναδιάταξη του περιεχομένου της ιστοσελίδας με την χρήση μόνο λίγων γραμμών κώδικα.
- Εύκολη δημιουργία «Drop – Down» μενού.

- Δημιουργία «Drag and Drop» διεπαφών. Παρέχεται η δυνατότητα δημιουργίας σελίδων με στοιχεία τα οποία μπορούν να επανατοποθετηθούν και να αναδιαταχθούν εύκολα (Drag and Drop).
- Προσθήκη πολύπλοκης φόρμας. Παρέχεται η δυνατότητα εύκολης προσθήκης φόρμας όπου τα πεδία της συμπληρώνονται αυτόματα (AJAX), αντλώντας δεδομένα από βάση δεδομένων στην πλευρά του εξυπηρετητή (server side).

Τέλος, η τεχνολογία AJAX (Asynchronous JavaScript and XML) αποτελεί έναν συνδυασμό από συσχετιζόμενες μεταξύ τους μεθόδους ανάπτυξης ιστοσελίδων, που χρησιμοποιούνται στην πλευρά του πελάτη (client-side) για την δημιουργία διαδραστικών διαδικτυακών εφαρμογών.

Με την τεχνολογία AJAX οι διαδικτυακές εφαρμογές μπορούν να στέλνουν και να ανακτούν δεδομένα προς και από τον εξυπηρετητή ασύγχρονα (στο παρασκήνιο) χωρίς να παρεμβαίνουν στην υπόλοιπη υπάρχουσα σελίδα, δίνοντας έτσι την εντύπωση μιας desktop εφαρμογής στον τελικό χρήστη. Τα δεδομένα συνήθως ανακτώνται μέσω του αντικειμένου XMLHttpRequest, όπου παρά το όνομα του, δεν καθίσταται απαραίτητη η χρήση της γλώσσας XML και ούτε οι αιτήσεις χρειάζεται να είναι ασύγχρονες.

Αξίζει να σημειωθεί ότι η τεχνολογία AJAX δεν είναι μια τεχνολογία από μόνη της, αλλά ένας συνδυασμός τεχνολογιών. Η AJAX χρησιμοποιεί τον συνδυασμό HTML και CSS²⁴ για σήμανση της δομής και της εμφάνισης, ενώ τον συνδυασμό DOM (Document Object Model) και JavaScript για δυναμική εμφάνιση και για να επιτρέπει στον χρήστη να αλληλεπιδρά με τις πληροφορίες που παρουσιάζονται.

Ακόμα, χρησιμοποιεί την JavaScript σε συνδυασμό με το αντικείμενο XMLHttpRequest για να καλύψει τον χρόνο που κάνει μια σελίδα να φορτώσει (page loading). Δηλαδή, με την χρήση της τεχνολογίας αυτής, δεν υφίσταται φόρτωση ολόκληρης της σελίδας (page loading), παρά μόνο φόρτωση συγκεκριμένης πληροφορίας (partial loading). [33]

Ουσιαστικά η τεχνολογία AJAX χρησιμοποιεί προγράμματα σεναρίων στην πλευρά του πελάτη για να δώσει στις διαδικτυακές εφαρμογές μικρότερους χρόνους απόκρισης. Η τεχνολογία αυτή διαχωρίζει την αλληλεπίδραση της πλευράς του πελάτη από την επικοινωνία με τον εξυπηρετητή και τις εκτελεί παράλληλα, μειώνοντας τις καθυστερήσεις της επεξεργασίας στην πλευρά του εξυπηρετητή που υπομένει ο χρήστης.

²⁴ Cascading Style Sheets-Διαδοχικά Φύλλα Στυλ

2. Ανάλυση Εφαρμογής

Εισαγωγή

Για να μπορέσουμε να κάνουμε μια πλήρη ανάλυση της προς υλοποίησης εφαρμογής και για να γίνει πλήρως κατανοητή η χρησιμότητα και η λειτουργικότητά της, ας κάνουμε ένα υποθετικό σενάριο εργασίας, για μια εταιρεία που ζητά ένα νέο εξειδικευμένο λογισμικό διαχείρισης δελτίων. Ας υποθέσουμε επίσης ότι και εμείς οι ίδιοι ανήκουμε σε μία εταιρεία ανάπτυξης λογισμικού (με την ονομασία Ένωση²⁵) και καλούμαστε να υλοποιήσουμε το συγκεκριμένο project.

Με τον τρόπο αυτό θα δούμε πώς μια εταιρεία ανάπτυξης λογισμικού θα πρέπει να αντιμετωπίζει ένα έργο (project) από την αρχή ως το τέλος.

Πριν από τη λήψη ενός έργου, εκτός από τη μελέτη του εξωτερικού περιβάλλοντος, στοιχεία του οποίου είδαμε στο πρώτο κεφάλαιο, και τη σύνταξη του επιχειρησιακού σχεδίου που είναι απαραίτητο για την οικονομική πρόοδο μίας εταιρείας, είναι εξίσου σημαντικό για την υλοποίηση του ίδιου του project να ακολουθηθούν τα βήματα που περιγράφει η επιστήμη της τεχνολογίας λογισμικού.

Στη συνέχεια του κεφαλαίου και πριν τη παρουσίαση της εφαρμογής θα δούμε κάποια στοιχεία της τεχνολογίας λογισμικού. Αρχικά θα ορίσουμε το πρόβλημα και τις διαδικασίες και θα συνεχίσουμε με την ανάλυση απαιτήσεων και έπειτα με το σχεδιασμό. Η ίδια η υλοποίηση θα παρουσιαστεί στο επόμενο κεφάλαιο.

Για την αναπαράσταση όλων των παραπάνω στο επίπεδο του σχεδιασμού χρησιμοποιήθηκε η γλώσσα UML²⁶, κάποια σχετική περιγραφή της οποίας δίνεται στο παράρτημα Α της εργασίας μας.

2.1 Ορισμός Προβλήματος

Σενάριο

Σε μία αναπτυσσόμενη εταιρεία παροχής υπηρεσιών, πρόσφατα αποφασίστηκε να δημιουργηθεί τμήμα Τεχνικής Υποστήριξης (Τ.Υ.) το οποίο εκτός των άλλων, έχει ως υποχρέωση να λαμβάνει κλήσεις/αιτήματα για προβλήματα τεχνικής φύσης από τους υπαλλήλους της εταιρείας, από εξωτερικούς πελάτες ή ακόμα και από συνεργαζόμενες εταιρείες με σκοπό την εξυπηρέτησή τους και τη διατήρηση της καλής λειτουργίας του πληροφοριακού συστήματος.

Στο πλαίσιο της οργάνωσης του τμήματος και με σκοπό την άμεση απόκριση στα αιτήματα, χωρίς αυτά να χάνονται, αλλά και επειδή χρειάζεται να καταγράφονται όλες οι ενέργειες των υπαλλήλων της εταιρείας και του συγκεκριμένου τμήματος, ώστε να διαπιστώνεται η ορθή χρήση του πληροφοριακού συστήματος από τους υπαλλήλους, αποφασίζεται να δημιουργηθεί μία διαδικτυακή εφαρμογή με βάση δεδομένων ώστε κυρίως:

- Να καταγράφονται τα αιτήματα των χρηστών με σκοπό να μη χάνονται.
- Να καταγράφονται οι ενέργειες των υπαλλήλων της Τ.Υ. έτσι ώστε να αξιολογείται η εργασία τους αλλά και για να υπάρχει συνέχεια της εργασίας από οποιοδήποτε υπάλληλο του τμήματος.
- Να δημιουργείται μία γνωσιακή βάση δεδομένων με τις λύσεις των προβλημάτων ώστε να είναι σε όλους τους τεχνικούς, άμεσα διαθέσιμες λύσεις σε επαναλαμβανόμενα προβλήματα.

Για όλα τα παραπάνω αποφασίστηκε να δοθεί σε εταιρεία ανάπτυξης λογισμικού (με την ονομασία Ένωση) το παραπάνω project.

²⁵ Ομάδας εργασίας του Μ.Π.Σ. Γλυκοφρύδη Αθηνά, Καραχάλιου Αλεξάνδρα, Χαραλαμπίδου Μαρία

²⁶ Unified Modeling Language. <http://www.uml.org/>

Πρόταση Ένωσης

Γενικά, η λειτουργία της εφαρμογής ορίζεται ως εξής:

Σε περίπτωση κάποιου προβλήματος, οι τεχνικοί της Τ.Υ. λαμβάνουν τηλεφωνικά ή με e-mail αίτημα για αποκατάσταση της λειτουργίας του Π.Σ. Το αίτημα αυτό καταγράφεται σε κάποιο τεχνικό δελτίο το οποίο χαρακτηρίζεται ως ανοιχτό. Σε περίπτωση που το πρόβλημα που αναφέρει ο χρήστης επιλύεται τηλεφωνικά, τότε ο τεχνικός ενημερώνει στις ενέργειες για τις οδηγίες που έδωσε και το δελτίο κλείνει.

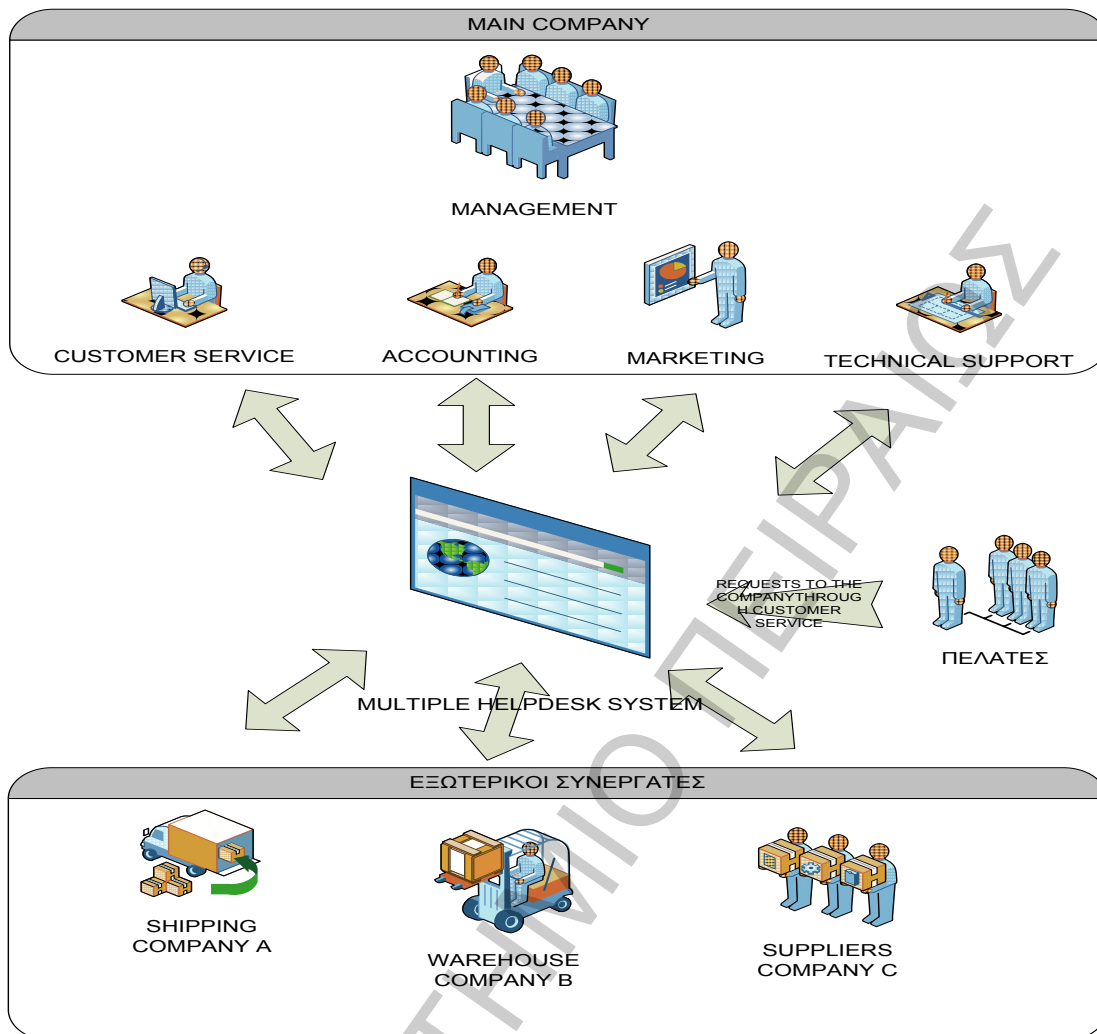
Τα ανοιχτά δελτία χρεώνονται στον τεχνικό που τα άνοιξε εκτός και αν ο προϊστάμενος (supervisor) κρίνει διαφορετικά και αλλάξει την ανάθεσή τους. Η ανάθεση σε διαφορετικό τεχνικό γίνεται επίσης όταν ο προϊστάμενος (supervisor) κρίνει ότι το πρόβλημα μπορεί να το διαχειριστεί άλλος τεχνικός που διαθέτει καταλληλότερη εκπαίδευση ή εμπειρία, ή όταν ο τεχνικός που έλαβε την κλήση ή το e-mail αδυνατεί να εξυπηρετήσει το αίτημα. Στη τελευταία περίπτωση, δηλώνει την αδυναμία στον προϊστάμενο (supervisor) και αυτός κρίνει για τη ανάθεση του δελτίου σε άλλον τεχνικό ή την ανάθεση του δελτίου σε άλλο τμήμα – department, δηλαδή σε άλλον προϊστάμενο (supervisor).

Τα δελτία ενημερώνονται με τις ενέργειες των τεχνικών και κλείνουν με την εκπλήρωση του αιτήματος ή με την γραπτή άρνηση εκπλήρωσης αιτήματος από τη Διεύθυνση του τμήματος. Επίσης τα δελτία κατατάσσονται σε κατηγορίες και μπορούν να αναζητηθούν έτσι ώστε να μπορούν οι τεχνικοί να ανατρέχουν σε παρόμοια προβλήματα και να αναζητούν λύσεις.

Για να διαφέρει η εφαρμογή από τις ήδη υπάρχουσες, και για να φέρει κάποια στοιχεία καινοτομίας, η πρόταση της Ένωσης είναι η εφαρμογή να μην είναι αφιερωμένη μόνο στο τμήμα της Τεχνικής Υποστήριξης, αλλά να μπορεί κάθε τμήμα της εταιρείας να έχει δικά της δελτία, με τα δικά της είδη αιτημάτων. Για παράδειγμα, το τμήμα Προμηθειών να μπορεί να καταγράφει τα αιτήματα για προμήθεια υλικού από τα υπόλοιπα τμήματα. Κάτι τέτοιο εκτιμάται ότι μπορεί να μειώσει τη γραφειοκρατία και να αυξήσει την αποδοτικότητα.

Είναι σημαντικό σε αυτή την περίπτωση να υπάρχουν καθαρά διακριτοί ρόλοι χρηστών, αλλά και να οριστεί με ακρίβεια το ποιος μπορεί να προσθέτει αιτήματα και που. Το πώς θα γίνει αυτό, αλλά και πώς θα γίνει ολόκληρη η μελέτη για την εφαρμογή παρουσιάζεται στα επόμενα κεφάλαια της ανάλυσης απαιτήσεων αλλά και του σχεδιασμού, όπου ουσιαστικά παρουσιάζεται κομμάτι της εργασίας της Ένωσης.

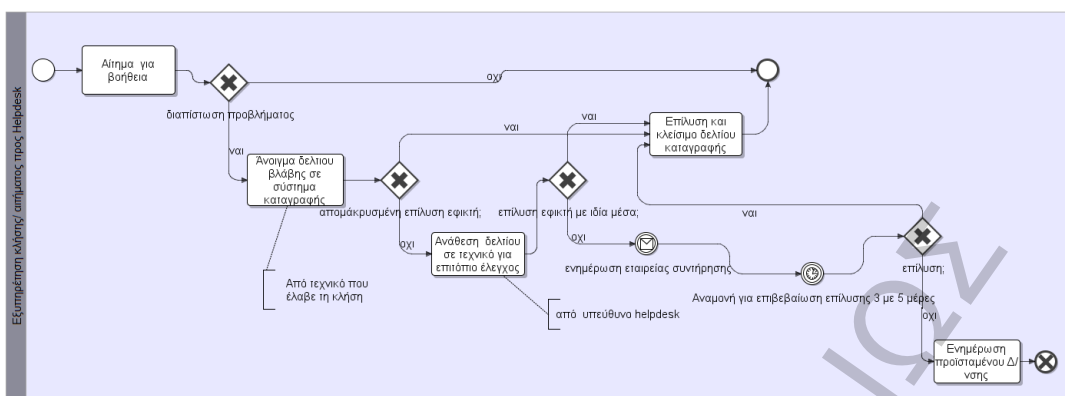
Επίσης, επειδή διαπιστώθηκε ότι η εταιρεία λαμβάνει πολλά τηλεφωνικά αιτήματα από πελάτες της αλλά και ότι έχει συνεχή επικοινωνία απαραίτητη για τη λειτουργία της, με διάφορες συνεργαζόμενες εταιρείες, η πρόταση της Ένωσης είναι να χρησιμοποιηθεί η νέα εφαρμογή όχι μόνο για την καταγραφή των αιτημάτων των υπαλλήλων της εταιρείας αλλά και για την καταγραφή των αιτημάτων των πελατών και των συνεργαζόμενων εταιρειών (Σχήμα 7).



Σχήμα 7 Πρόταση Ένωσης για Ολοκληρωμένο Δικτυακό Σύστημα Ticketing. Όλοι επικοινωνούν δια μέσου της Εφαρμογής

2.2 Ανάλυση Απαιτήσεων

Η πρώτη εργασία της Ένωσης είναι μέσα από συνεντεύξεις και ερωτηματολόγια να διακρίνει τις ανάγκες του προσωπικού και της εταιρείας, και να διακρίνει τις απαιτήσεις των χρηστών και της ίδιας της εφαρμογής, ώστε να τις εντάξει στο σχεδιασμό. Στο σχήμα που ακολουθεί (Σχήμα 8), διαγράφεται σχηματικά η διαδικασία που διαπιστώθηκε από την Ένωση ότι θα πρέπει να ακολουθείται. Εργασία της ένωσης είναι επίσης να καλύψει τα πιθανά προβλήματα ή κενά της παρακάτω διαδικασίας.



Σχήμα 8 Διαδικασία που Ακολουθείται κατά την Εξυπηρέτηση Κλήσης/Αιτήματος.

2.2.1 Απαιτήσεις Εφαρμογής

2.2.1.1 Ευχρηστία

Η προς μελέτη εφαρμογή διαπιστώθηκε μέσω της διαδικασίας των συνεντεύξεων ότι απευθύνεται σε έμπειρους χρήστες, καλούς γνώστες του χειρισμού εφαρμογών. Παρόλο το έμπειρο προσωπικό της εταιρείας, η βασική απαίτηση της εφαρμογής που κρίνεται ότι πρέπει να καλυφθεί είναι η ευχρηστία της.

Με τον όρο ευχρηστία λογισμικού, εννοούμε (σύμφωνα με το πρότυπο ISO 9241-11) «την δυνατότητα ενός συστήματος που χρησιμοποιείται από καθορισμένους χρήστες με καθορισμένους στόχους, υπό καθορισμένες συνθήκες χρήσης, να παρέχει αποτελεσματικότητα, αποδοτικότητα και υποκειμενική ικανοποίηση στους χρήστες του». [34] Έτσι η ευχρηστία του συστήματος/λογισμικού αναφέρεται στον σχεδιασμό της διεπαφής χρήστη (user interface), ώστε αυτή να είναι εύκολη στην χρήση και αποτελεσματική από όλους τους χρήστες.

Για να εξασφαλιστεί η απαίτηση της ευχρηστίας στην προς ανάπτυξη εφαρμογή, λάβαμε υπ' όψιν μας τα ακόλουθα για την σχεδίαση της διεπαφής χρήστη – user interface :

- Αρχικά δόθηκε έμφαση στην διεύθυνση της σελίδας (url) ώστε να είναι ευκολομνημόνευτη.
- Στο σύνολο της εφαρμογής χρησιμοποιείται κοινό φόντο, όμοιος χρωματισμός, κοινή γραμματοσειρά και μέγεθος για τις ίδιες λειτουργίες και χαρακτηρισμούς (επικεφαλίδες, βασικό κείμενο κ.τ.λ.).
- Γενικότερα, σχεδιάστηκε απλό γραφικό περιβάλλον στην εφαρμογή, όπου χρησιμοποιούνται εύκολα (στην χρήση) και κατανοητά menus ώστε να δίνεται η δυνατότητα στον χρήστη να αντιλαμβάνεται άμεσα όλες τις λειτουργίες που παρέχονται από την εφαρμογή.
- Για τις περισσότερες λειτουργίες της εφαρμογής που είναι διαθέσιμες στους χρήστες της, σχεδιάστηκαν φόρμες συμπλήρωσης ενώ τα πεδία συμπλήρωσης σε κάθε φόρμα σχεδιάστηκαν κατά πλειοψηφία με drop down menu, με βασικό στόχο την διευκόλυνση των χρηστών και την μείωση του χρόνου διεκπεραίωσης μιας λειτουργίας / διαδικασίας.
- Για τον εύκολο προσανατολισμό των χρηστών, κατηγοριοποιήθηκαν όλες οι διαθέσιμες λειτουργίες της εφαρμογής, ενώ επιπλέον κάθε υποσελίδα έχει τον δικό της τίτλο, ο οποίος προδίδει και το περιεχόμενό της και παρέχει πληροφορίες για το τι μπορεί να βρει ο χρήστης σε άλλες υποσελίδες.
- Επίσης η διεπαφή σχεδιάστηκε έτσι ώστε να εμφανίζονται μηνύματα λάθους και μηνύματα καθοδήγησης σε πιθανά λάθη των χρηστών. Για παράδειγμα κατά την σύνδεση του χρήστη (login), σε περιπτώσεις πληκτρολόγησης λάθους κωδικού (password) ή/και ονόματος χρήστη (username) εμφανίζονται μηνύματα καθοδήγησης για την ενημέρωση του χρήστη με τα ορθά στοιχεία. Άλλα τέτοια παρόμοια μηνύματα

ενημέρωσης και καθοδήγησης των χρηστών εμφανίζονται σε λειτουργίες όπως: δημιουργία νέας εταιρείας, δημιουργία νέου τμήματος, εγγραφή χρήστη κ.α.

2.1.2.2 Ιδιωτικότητα

Άλλη απαίτηση της εφαρμογής που εντοπίστηκε κατά την ανάλυση και κρίθηκε ως απαραίτητη για να εξασφαλιστεί, είναι αυτή της ιδιωτικότητας. Με τον όρο ιδιωτικότητα αναφερόμαστε στην «αξίωση των ατόμων, ομάδων και οργανισμών να καθορίζουν το χρόνο, τον τρόπο και την έκταση αναφορικά με τη συλλογή και επεξεργασία των προσωπικών τους δεδομένων» [35] Για να καλυφθεί η απαίτηση αυτή στην εφαρμογή, βασιστήκαμε στο μοντέλο Ελέγχου Πρόσβασης βάσει Ρόλων (Role Based Access Control – RBAC). Σύμφωνα με το μοντέλο αυτό, ανατίθενται στους χρήστες ρόλοι και στην συνέχεια εκχωρούνται διαφορετικά δικαιώματα πρόσβασης (access permissions) σε κάθε ρόλο. Συγκεκριμένα για τους υπαλλήλους – χρήστες της εφαρμογής διακρίνονται οι παρακάτω ρόλοι:

- Απλός χρήστης (Simple user): δικαίωμα πρόσβασης στην εφαρμογή μόνο για την απλή παρακολούθηση των τεχνικών δελτίων καταγραφής που αντιστοιχούν στην πορεία εξέλιξης των αιτημάτων αποκατάστασης βλάβης.
- Τεχνικός ή Πράκτορας (Technician or Agent): δικαίωμα πρόσβασης στην εφαρμογή τόσο για την παρακολούθηση των τεχνικών δελτίων καταγραφής όσο και για την επέμβαση (τροποποίηση) σε αυτά με στόχο την επίλυσή τους.
- Προϊστάμενος (Supervisor): δικαίωμα πρόσβασης στην εφαρμογή για τον έλεγχο των τεχνικών δελτίων καταγραφής και για την μερική διαχείριση των χρηστών.
- Διαχειριστής (Administrator): δικαίωμα πρόσβασης στην εφαρμογή για την πλήρη διαχείριση και τον πλήρη έλεγχο τόσο των χρηστών και των τεχνικών δελτίων καταγραφής όσο και της γενικότερης σχεδιαστικής δομής της εφαρμογής.

2.1.2.3 Διαχείριση δελτίων

Σημαντική απαίτηση της συγκεκριμένης εφαρμογής είναι η εύκολη και αποτελεσματική διαχείριση των αιτημάτων αποκατάστασης βλάβης από τους χρήστες, κάτι το οποίο απαιτεί τον προσεκτικό σχεδιασμό των τεχνικών δελτίων καταγραφής που αντιστοιχούν στην εξυπηρέτηση των αιτημάτων αυτών.

Εφόσον η εξυπηρέτηση των αιτημάτων αποκατάστασης βλάβης είναι το κύριο μέλημα της εφαρμογής αυτής, κρίθηκε απαραίτητο να κατασκευαστούν φόρμες συμπλήρωσης για τα τεχνικά δελτία καταγραφής και να δοθεί ιδιαίτερη προσοχή στα πεδία που θα συντελούν τις αντίστοιχες αυτές φόρμες. Έτσι αποφασίστηκε να ενταχθούν στην φόρμα για τα τεχνικά δελτία καταγραφής τα ακόλουθα πεδία συμπλήρωσης :

- Πεδίο που αφορά τον τίτλο του δελτίου καταγραφής, ώστε να καταχωρείται μία ονομασία για κάθε δελτίο καταγραφής για να μπορούν τα δελτία αυτά να αναζητηθούν με ευκολία.
- Πεδίο που αφορά την περιγραφή του δελτίου καταγραφής, ώστε να καταχωρείται μία σύντομη περιγραφή για να είναι αντιληπτό το πρόβλημα της βλάβης κάθε φορά που απαιτεί επεξεργασία το εκάστοτε δελτίο και παράλληλα σύμφωνα με αυτή να μπορεί να αναζητηθεί το δελτίο.
- Πεδία που αναφέρονται στο όνομα της εταιρείας και στο όνομα του τμήματος, στα οποία ανήκει ο τεχνικός που έλαβε το αίτημα αποκατάστασης βλάβης και ανατέθηκε σε αυτόν αυτόματα η επίλυση του, ώστε να καταχωρείται η εταιρεία καθώς και το τμήμα, τα οποία είναι υπεύθυνα για την εξυπηρέτηση του δελτίου καταγραφής.
- Πεδίο που αφορά τον πελάτη που ζήτησε την αποκατάσταση βλάβης ώστε να καταχωρείται ο πελάτης και να γίνεται ενημέρωση του εφόσον εξεταστεί το αίτημα του.
- Πεδίο που αφορά την κατηγοριοποίηση του αιτήματος, όπου μέσω ενός drop down menu κατατάσσεται το αίτημα σε μία από τις διαθέσιμες κατηγορίες, ώστε να είναι ευκολότερη η εύρεση του καθώς επίσης να είναι και ευκολότερη η προβολή παρόμοιων προβλημάτων της ίδιας κατηγορίας και των λύσεων που έχουν δοθεί σε αυτά.

- Πεδίο που αφορά τον βαθμό προτεραιότητας του κάθε αιτήματος (χαμηλή, μεσαία, υψηλή προτεραιότητα) έτσι ώστε να επιλύονται πρώτα οι περισσότερο κρίσιμες βλάβες οι οποίες μπορεί να επηρεάζουν και άλλες δευτερεύουσας προτεραιότητας βλάβες.
- Πεδίο που αφορά τις απαραίτητες ενέργειες που πρέπει να πραγματοποιηθούν με σκοπό την εξυπηρέτηση του εκάστοτε αιτήματος. Με αυτή την καταχώρηση συλλέγονται όλες οι ενέργειες επίλυσης προβλημάτων ώστε να προβάλλονται ευκολότερα οι πιθανές λύσεις και να επιλύονται γρηγορότερα τα προβλήματα παρόμοιας φύσης.
- Πεδίο που αφορά την ανάθεση του αιτήματος αποκατάστασης βλάβης σε κάποιον τεχνικό (technician) ή προϊστάμενο (supervisor) τμήματος, ώστε να καταχωρείται ο υπεύθυνος για την επίλυση του εκάστοτε αιτήματος και να αξιολογείται η εργασία του.
- Πεδίο που αφορά την κατάσταση του κάθε τεχνικού δελτίου καταγραφής (ανοιχτό ή κλειστό) ώστε να διαχωρίζονται τα αιτήματα που έχουν εξεταστεί και έχουν ολοκληρωθεί (κλειστά δελτία) και τα αιτήματα που απαιτούν ακόμα επίλυση (ανοιχτά δελτία).

2.1.2.4 Διαχρονικότητα

Σημειώνεται ότι σημαντικός στόχος μιας νέας εφαρμογής που ολοκληρώνεται και γίνεται η εγκατάστασή της είναι να μπορέσει να είναι διαχρονική, γεγονός το οποίο απαιτεί τον συνεχή έλεγχο και επαναπροσδιορισμό της. Έτσι για την κάλυψη και αυτής της απαίτησης προτείνεται στην ενδιαφερόμενη εταιρεία, αφού της δοθεί η εφαρμογή από την εταιρεία ανάπτυξης λογισμικού «Ένωση» και γίνει η εγκατάστασή της, να πραγματοποιηθούν από την πλευρά της οι λειτουργίες ελέγχου και διόρθωσης.

Ουσιαστικά προτείνεται στην εταιρεία, μετά την αλληλεπίδραση των χρηστών με την εφαρμογή να γίνουν πιθανές αλλαγές και διορθώσεις στην διεπαφή της εφαρμογής σύμφωνα με την ανάδραση (feedback) των χρηστών. Με τον συνεχή επανέλεγχο της εφαρμογής και τις αλλαγές που θα αντιστοιχούν στις προτιμήσεις των χρηστών θα είναι εφικτή η διαχρονικότητα της εφαρμογής.

2.1.2.5 Διεθνοποίηση

Αν και δε φαίνεται να αποτελεί κύρια απαίτηση της εφαρμογής, διαπιστώσαμε κατά τη μελέτη μας για το Struts ότι η διεθνοποίηση αποτελεί ένα από τα κύρια πλεονεκτήματα αυτής της τεχνικής. Το γεγονός αυτό σε συνδυασμό με την τάση που υπάρχει στην Ε.Ε. (μέλος της οποίας είναι και η χώρα μας) για την ενίσχυση της πολυπολιτισμικότητας και της δυσκολίας που μπορεί να φέρει η χρήση της ελληνικής από κάποιον αλλοδαπό, μας οδήγησαν στην πλήρη παραμετροποίηση της διεπαφής της εφαρμογής, τόσο στην Ελληνική όσο και στην Αγγλική. Επιπλέον, με την απλή τροποποίηση των λεκτικών του αρχείου `applicationresourceproperties` που προσφέρει το `struts`, η εφαρμογή μπορεί να μεταφραστεί εύκολα σε οποιαδήποτε άλλη γλώσσα.

2.2.2 Απαιτήσεις Χρήστη

Οι χρήστες μας είναι κυρίως τεχνικοί, καλοί ή άριστοι γνώστες της πληροφορικής και του χειρισμού εφαρμογών με παραθυρικό ή web interface. Ως εκ τούτου, πιστεύουμε ότι η γνώμη τους για τη λειτουργία αλλά και τη διεπαφή της εφαρμογής είναι αρκετά κατατοπιστική για το σχεδιασμό, και τον περαιτέρω έλεγχο της λειτουργίας της εφαρμογής.

Οι απαιτήσεις χρηστών που διακρίναμε (μέσα από τη διαδικασία ερωτηματολογίων και συνέντευξης) παρουσιάζονται παρακάτω. Οι χρήστες επιθυμούν τα εξής:

- Κάθε υπάλληλος να διαχειρίζεται την εφαρμογή με το δικό του username και password.
- Να διαχωρίζονται ρόλοι μεταξύ των υπαλλήλων (απλοί χρήστες – simple users, τεχνικοί - technicians, προϊστάμενοι - supervisors και διαχειριστές - administrators.)
- Κάθε χρήστης - τεχνικός να μπορεί να βλέπει όλα τα αιτήματα που έχουν εξυπηρετηθεί (κλειστά δελτία) και που αντιστοιχούν στο τμήμα που ανήκει, αλλά να μην μπορεί να τα τροποποιεί.

- Κάθε χρήστης - τεχνικός να μπορεί βλέπει όλα τα τρέχοντα αιτήματα (ανοιχτά δελτία) που αντιστοιχούν στο τμήμα που ανήκει και να μπορεί να προσθέτει ενέργειες σε αυτά αλλά και να μπορεί να τα κλείνει.
- Κάθε απλός χρήστης να μπορεί μόνο να βλέπει την πορεία των αιτημάτων (δελτίων καταγραφής).
- Ο προϊστάμενος (supervisor) να μπορεί να διαχειρίζεται το σύνολο των δελτίων στο τμήμα που είναι υπεύθυνος, να προσθέτει ενέργειες, να αναθέτει τα δελτία σε διαφορετικό τεχνικό ή σε διαφορετικό τμήμα, να κλείνει δελτία, να ανοίγει τα κλειστά δελτία του τμήματός του.
- Ο προϊστάμενος (supervisor) να μπορεί να διαχειρίζεται το σύνολο των χρηστών που ανήκουν στο τμήμα, στο οποίο είναι υπεύθυνος.
- Ο προϊστάμενος (supervisor) υπάρχει ανάγκη για να εξάγει στατιστικά στοιχεία για τους υπαλλήλους του τμήματός του.
- Να υπάρχει ένας τουλάχιστον διαχειριστής (administrator), ο οποίος να είναι υπεύθυνος για την διαχείριση της ίδιας της εφαρμογής. Να έχει την πλήρη διαχείριση όλων των χρηστών, τον πλήρη έλεγχο όλων των δελτίων, να φροντίζει για την αδιάλειπτη λειτουργία της εφαρμογής (servers, backup κ.τ.λ), και να διαχειρίζεται το σύνολο των στατιστικών στοιχείων.
- Δελτία να μπορούν να ανοίγουν οι τεχνικοί, οι προϊστάμενοι (supervisors) και οι διαχειριστές (administrators).
- Η διαγραφή δελτίων να είναι δυνατή μόνο από τον διαχειριστή (administrator) και από τον προϊστάμενο (supervisor), με την διαφορά ότι ο προϊστάμενος να μπορεί να διαγράψει μόνο τα υπάρχοντα δελτία που ανήκουν στο τμήμα του.
- Τα πεδία που συμπληρώνονται για την δημιουργία δελτίων να είναι κατά το δυνατό περισσότερο με drop down menu.
- Να μπορούν τα δελτία να λαμβάνουν κάποιον βαθμό προτεραιότητας (επείγον, κανονική, σημαντική κ.α.)

2.3 Σχεδιασμός

Με βάση την ανάλυση απαιτήσεων τόσο της εφαρμογής όσο και των χρηστών που προηγήθηκε, οδηγούμαστε αρχικά στον σχεδιασμό της εφαρμογής χρησιμοποιώντας διαγράμματα UML, όπου στην συνέχεια θα πραγματοποιηθεί η υλοποίηση αυτών των διαγραμματικών αναπαραστάσεων.

2.3.1 Δυνατότητες Πρόσβασης των Χρηστών της Εφαρμογής

Για τους υπαλλήλους της εταιρείας, για τους χρήστες της εφαρμογής δηλαδή, διακρίνονται ρόλοι ώστε να υπάρχει διαχωρισμός στις δυνατότητες πρόσβασης και τροποποίησης στην εφαρμογή. Οι ρόλοι που αντιστοιχούν στους χρήστες της εφαρμογής είναι: απλοί χρήστες (simple users), τεχνικοί (technicians), προϊστάμενοι (supervisors) και διαχειριστές (administrators). Κάθε χρήστης ανεξαρτήτως ρόλου για την σύνδεσή του (login) στην εφαρμογή έχει ένα προσωπικό μοναδικό συνδυασμό παραμέτρων username και password, όπου ανάλογα με αυτόν γίνεται η σύνδεση στην αντίστοιχη διεπαφή στην οποία μπορεί να έχει πρόσβαση ο χρήστης.

Καθένας από τους ρόλους αυτούς έχει διαφορετικές δυνατότητες πρόσβασης στην εφαρμογή, οι οποίες αναλύονται παρακάτω ενώ παρουσιάζονται και μέσω του ακόλουθου διαγράμματος περιπτώσεων χρήσης (Σχήμα 9).

Ο απλός χρήστης (simple user) μπορεί:

- Να συνδεθεί στο σύστημα, περίπτωση χρήσης «Log in»
- Να προβάλει την πορεία των δελτίων καταγραφής του τμήματος στο οποίο ανήκει, περίπτωση χρήσης με τίτλο «Προβολή Πορείας Δελτίων»

- Να προβάλει τους καταχωρημένους πελάτες της εταιρείας στην οποία ανήκει, περίπτωση χρήσης «Προβολή Πελατών»
- Να προβάλει τις καταχωρημένες κατηγορίες που αφορούν τα δελτία καταγραφής που αντιστοιχούν στο τμήμα της εταιρείας που ανήκει, περίπτωση χρήσης «Προβολή Κατηγοριών»
- Να αναζητήσει τεχνικά δελτία καταγραφής μέσω της διαδικασίας αναζήτησης - search, περίπτωση χρήσης με τίτλο «Αναζήτηση Δελτίου Καταγραφής»
- Να αλλάξει τον προσωπικό κωδικό χρήστη (password) του, μέσω του προφίλ εφόσον έχει εισαγάγει τον κωδικό χρήστη που του έχει αποσταλεί από την δημιουργία λογαριασμού του, γεγονός που αντιστοιχεί στην περίπτωση χρήσης με τίτλο «Αλλαγή Password»
- Να αποσυνδεθεί από το σύστημα, περίπτωση χρήσης «Log out»

Στη συνέχεια ο χρήστης – τεχνικός (technician) μπορεί:

- Να συνδεθεί στο σύστημα, περίπτωση χρήσης «Log in»
- Να αναζητήσει τεχνικά δελτία καταγραφής μέσω της διαδικασίας αναζήτησης - search, περίπτωση χρήσης με τίτλο «Αναζήτηση Δελτίου Καταγραφής»
- Να αλλάξει τον προσωπικό κωδικό χρήστη (password) του, μέσω του προφίλ εφόσον έχει εισαγάγει τον κωδικό χρήστη που του έχει αποσταλεί από την δημιουργία λογαριασμού του, γεγονός που αντιστοιχεί στην περίπτωση χρήσης με τίτλο «Αλλαγή Password»
- Να προβάλει το σύνολο των δελτίων που έχουν ανατεθεί σε αυτόν (συγκεκριμένο χρήστη – τεχνικό), γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Προβολή Συνόλου Προσωπικών Δελτίων»
- Να προβάλει το σύνολο των δελτίων που έχουν ανατεθεί στους υπολοίπους χρήστες του τμήματος στο οποίο ανήκει, περίπτωση χρήσης «Προβολή Συνόλου Υπολοίπων Δελτίων»
- Να ανοίξει ένα νέο δελτίο, περίπτωση χρήσης «Άνοιγμα Νέου Δελτίου Καταγραφής»
- Να κλείσει (εφόσον έχει δώσει λύση) ένα ανοιχτό δελτίο, είτε δελτίο που έχει ανατεθεί σε αυτόν είτε που έχει ανατεθεί σε άλλους χρήστες του ίδιου όμως τμήματος στην εταιρεία που ανήκει, περίπτωση χρήσης «Κλείσιμο Ανοιχτού Δελτίου»
- Να προσθέσει ενέργειες σε ένα ανοιχτό δελτίο που έχει ανατεθεί τόσο σε αυτόν όσο και σε άλλους χρήστες του ίδιου όμως τμήματος, περίπτωση χρήσης «Προσθήκη Ενέργειών σε Ανοιχτό Δελτίο»
- Να δημιουργήσει νέες κατηγορίες για τα τεχνικά δελτία καταγραφής για το τμήμα και την εταιρεία που ανήκει, περίπτωση χρήσης με τίτλο «Δημιουργία Νέας Κατηγορίας Βλάβης»
- Να δημιουργήσει νέους πελάτες για την εταιρεία στην οποία ανήκει, περίπτωση χρήσης με τίτλο «Δημιουργία Νέου Πελάτη»
- Να αποσυνδεθεί από το σύστημα, περίπτωση χρήσης «Log out»

Επίσης ο χρήστης – προϊστάμενος (supervisor) έχει τις εξής δυνατότητες:

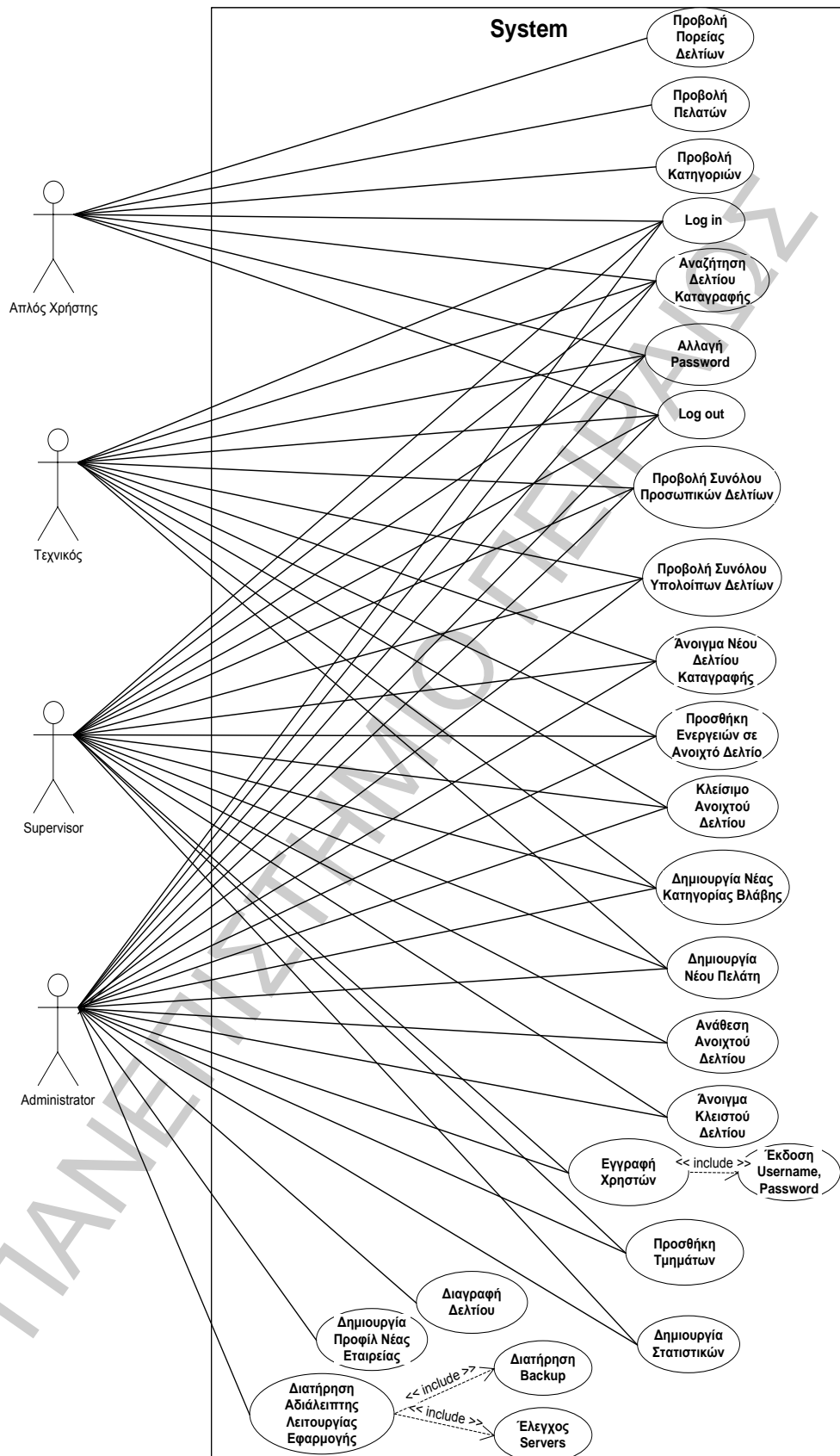
- Να συνδεθεί στο σύστημα, περίπτωση χρήσης «Log in»
- Να αναζητήσει τεχνικά δελτία καταγραφής μέσω της διαδικασίας αναζήτησης - search, περίπτωση χρήσης με τίτλο «Αναζήτηση Δελτίου Καταγραφής»
- Να αλλάξει τον προσωπικό κωδικό χρήστη (password) του, μέσω του προφίλ εφόσον έχει εισαγάγει τον κωδικό χρήστη που του έχει αποσταλεί από την δημιουργία λογαριασμού του, γεγονός που αντιστοιχεί στην περίπτωση χρήσης με τίτλο «Αλλαγή Password»
- Να προβάλει το σύνολο των δελτίων που έχουν ανατεθεί σε αυτόν (συγκεκριμένο χρήστη – προϊστάμενο), γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Προβολή Συνόλου Προσωπικών Δελτίων»

- Να προβάλλει το σύνολο των δελτίων που έχουν ανατεθεί στους υπολοίπους χρήστες του τμήματος στο οποίο ανήκει, περίπτωση χρήσης «Προβολή Συνόλου Υπολοίπων Δελτίων»
- Να ανοίξει ένα νέο δελτίο, περίπτωση χρήσης «Άνοιγμα Νέου Δελτίου Καταγραφής»
- Να προσθέσει ενέργειες σε ένα ανοιχτό δελτίο που έχει ανατεθεί τόσο σε αυτόν όσο και σε άλλους χρήστες του ίδιου όμως τμήματος, περίπτωση χρήσης «Προσθήκη Ενεργειών σε Ανοιχτό Δελτίο»
- Να κλείσει (εφόσον έχει δώσει λύση) ένα ανοιχτό δελτίο, είτε δελτίο που έχει ανατεθεί σε αυτόν είτε που έχει ανατεθεί σε άλλους χρήστες στο τμήμα της εταιρείας που ανήκει περίπτωση χρήσης «Κλείσιμο Ανοιχτού Δελτίου»
- Να αναθέσει ένα ανοιχτό δελτίο σε κάποιον άλλον τεχνικό του τμήματος στο οποίο ανήκει ή ακόμα σε έναν άλλον προϊστάμενο (supervisor) διαφορετικού τμήματος στην εταιρεία που ανήκει, περίπτωση χρήσης «Ανάθεση Ανοιχτού Δελτίου»
- Να ανοίξει ένα κλειστό δελτίο του τμήματος της εταιρείας στην οποία ανήκει, δηλαδή ένα δελτίο που έχει ήδη εξυπηρετηθεί, περίπτωση χρήσης «Άνοιγμα Κλειστού Δελτίου»
- Να δημιουργήσει νέες κατηγορίες για τα τεχνικά δελτία καταγραφής για το τμήμα της εταιρείας στην οποία ανήκει, περίπτωση χρήσης με τίτλο «Δημιουργία Νέας Κατηγορίας Βλάβης»
- Να δημιουργήσει νέους πελάτες για την εταιρεία στην οποία ανήκει, περίπτωση χρήσης με τίτλο «Δημιουργία Νέου Πελάτη»
- Να κάνει εγγραφή χρηστών (registration) αλλά έχει την δυνατότητα αυτή μόνο στο τμήμα (department) της εταιρείας στην οποία ανήκει, γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Εγγραφή Χρηστών» όπου περιλαμβάνει την περίπτωση χρήσης «Έκδοση Username, Password»
- Να προσθέσει νέα τμήματα αλλά μόνο στην εταιρεία στην οποία ανήκει, περίπτωση χρήσης «Προσθήκη Τμήματος»
- Να διαγράψει μια ήδη υπάρχουσα εγγραφή που όμως ανήκει μόνο στο τμήμα της εταιρείας στο οποίο είναι υπεύθυνος, γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Διαγραφή Εγγραφής». Οι εγγραφές αυτές μπορούν να αφορούν δελτία, κατηγορίες, πελάτες, τμήματα ακόμα και χρήστες.
- Να δημιουργεί αναφορές στατιστικών αλλά μόνο για το τμήμα στο οποίο ανήκει, περίπτωση χρήσης με τον τίτλο «Δημιουργία Στατιστικών».
- Να αποσυνδεθεί από το σύστημα, περίπτωση χρήσης «Log out»

Τέλος, ο χρήστης - διαχειριστής (administrator) έχει τις ακόλουθες δυνατότητες:

- Να συνδεθεί στο σύστημα, περίπτωση χρήσης «Log in»
- Να αναζητήσει τεχνικά δελτία καταγραφής μέσω της διαδικασίας αναζήτησης - search, περίπτωση χρήσης με τίτλο «Αναζήτηση Δελτίου Καταγραφής»
- Να αλλάξει τον προσωπικό κωδικό χρήστη (password) του μέσω του προφίλ, γεγονός που αντιστοιχεί στην περίπτωση χρήσης με τίτλο «Αλλαγή Password»
- Να προβάλλει το σύνολο των δελτίων που έχουν ανατεθεί στους υπολοίπους χρήστες που ανήκουν σε όλες τις εταιρείες που μπορεί να διαχειριστεί, περίπτωση χρήσης «Προβολή Συνόλου Υπολοίπων Δελτίων»
- Να ανοίξει ένα νέο δελτίο, περίπτωση χρήσης «Άνοιγμα Νέου Δελτίου Καταγραφής»
- Να προσθέσει ενέργειες σε ένα ανοιχτό δελτίο, σε δελτίο που έχει ανατεθεί είτε σε αυτόν είτε σε άλλους χρήστες των εταιρειών που έχει την δυνατότητα να διαχειριστεί, περίπτωση χρήσης «Προσθήκη Ενεργειών σε Ανοιχτό Δελτίο»
- Να κλείσει (εφόσον έχει δώσει λύση) ένα ανοιχτό δελτίο που έχει ανατεθεί τόσο σε αυτόν όσο και σε άλλους χρήστες που ανήκουν στα διαθέσιμα τμήματα και εταιρείες, περίπτωση χρήσης «Κλείσιμο Ανοιχτού Δελτίου»

- Να αναθέσει ένα ανοιχτό δελτίο σε κάποιον άλλον τεχνικό (που μπορεί να ανήκει σε όλα τα διαθέσιμα τμήματα των εταιρειών) ή ακόμα και σε άλλον προϊστάμενο (supervisor) όλων των διαθέσιμων τμημάτων των εταιρειών, περίπτωση χρήσης «Ανάθεση Ανοιχτού Δελτίου»
- Να ανοίξει ένα κλειστό δελτίο που μπορεί να ανήκει σε όλα τα διαθέσιμα τμήματα και εταιρείες, δηλαδή ένα δελτίο που έχει ήδη εξυπηρετηθεί, περίπτωση χρήσης «Ανοιγμα Κλειστού Δελτίου»
- Να δημιουργήσει νέες κατηγορίες για τα τεχνικά δελτία καταγραφής όλων των διαθέσιμων τμημάτων και εταιρειών, περίπτωση χρήσης με τίτλο «Δημιουργία Νέας Κατηγορίας Βλάβης»
- Να δημιουργήσει νέους πελάτες για όλες τις διαθέσιμες εταιρείες, περίπτωση χρήσης με τίτλο «Δημιουργία Νέου Πελάτη»
- Να κάνει εγγραφή χρηστών (registration) σε όλα τα τμήματα και σε όλες τις διαθέσιμες εταιρείες (έχει την πλήρη διαχείριση χρηστών), γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Εγγραφή Χρηστών» όπου περιλαμβάνει την περίπτωση χρήσης «Έκδοση Username, Password»
- Να προσθέτει νέα τμήματα (departments) σε όλες τις διαθέσιμες εταιρείες τις οποίες μπορεί να διαχειρίζεται, περίπτωση χρήσης «Προσθήκη Τμήματος»
- Να διαγράψει μια ήδη υπάρχουσα εγγραφή, γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Διαγραφή Εγγραφής». Οι εγγραφές αυτές μπορούν να αφορούν δελτία, κατηγορίες, εταιρείες, πελάτες, τμήματα ακόμα και χρήστες.
- Να δημιουργεί προφίλ νέας εταιρείας, γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Δημιουργία Προφίλ Νέας Εταιρείας»
- Να διατηρεί την αδιάλειπτη λειτουργία της εφαρμογής, γεγονός που αντιστοιχεί στην περίπτωση χρήσης «Διατήρηση Αδιάλειπτης Λειτουργίας Εφαρμογής», όπου περιλαμβάνει τις περιπτώσεις χρήσης «Διατήρηση Backup» και «Έλεγχος Servers»
- Να δημιουργεί αναφορές στατιστικών για το σύνολο των στοιχείων περίπτωση χρήσης με τον τίτλο «Δημιουργία Στατιστικών».
- Να αποσυνδεθεί από το σύστημα, περίπτωση χρήσης «Log out»



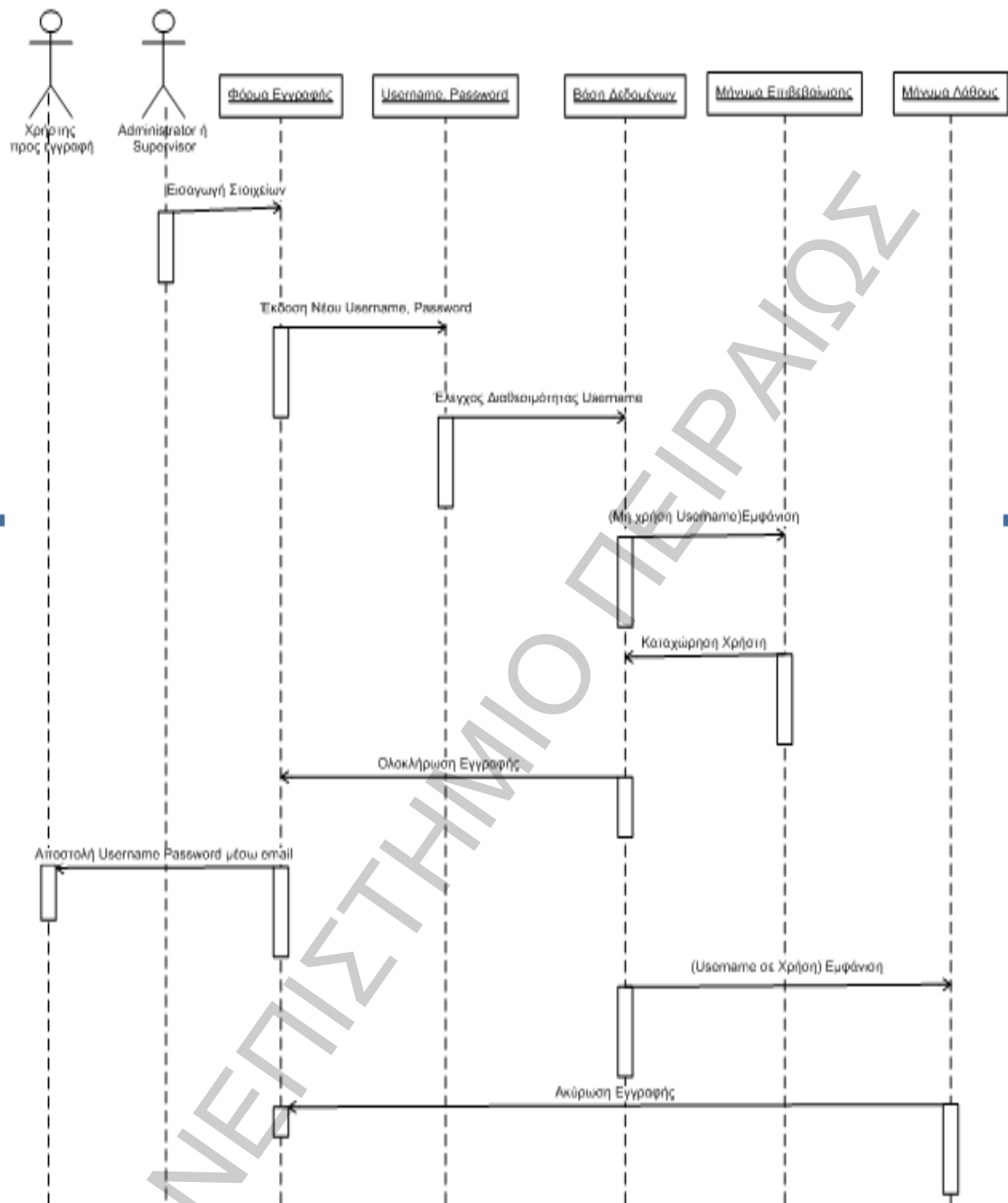
Σχήμα 9 Διάγραμμα Περιπτώσεων Χρήσης για τους Χρήστες της Εφαρμογής.

2.3.2 Σχεδιασμός της Διαδικασίας Σύνδεσης (login) του κάθε Χρήστη στην Εφαρμογή καθώς και της Διαδικασίας Εγγραφής του

Εφόσον αναλύσαμε τους διαφορετικούς ρόλους των χρηστών της εφαρμογής και έγινε κατανοητό ότι για την σύνδεση (login) οποιουδήποτε χρήστη στην εφαρμογή είναι απαραίτητος ο συνδυασμός όνομα χρήστη (username) και κωδικός πρόσβασης (password), είναι σκόπιμο να δοθεί προσοχή στον σχεδιασμό της διαδικασίας που θα ακολουθείται κατά την σύνδεση του χρήστη ώστε να αποφευχθούν λάθη μη εξουσιοδοτημένης πρόσβασης.

Αρχικά σημειώνεται ότι η εγγραφή (registration) κάθε χρήστη πραγματοποιείται είτε από τον διαχειριστή (administrator) είτε από τον προϊστάμενο (supervisor), με την διαφορά ότι ο προϊστάμενος μπορεί να προσθέσει χρήστες μόνο στο τμήμα της εταιρείας στο οποία ανήκει, ενώ ο διαχειριστής σε όλα τα τμήματα των εταιρειών που έχει την δυνατότητα να διαχειρίζεται. Ο administrator ή ο supervisor καταχωρεί τα στοιχεία του χρήστη που απαιτούνται για την εγγραφή: όνομα, επίθετο, όνομα εταιρείας, όνομα τμήματος, τηλέφωνο, email, ρόλος χρήστη, username και password στην φόρμα για την εγγραφή χρηστών.

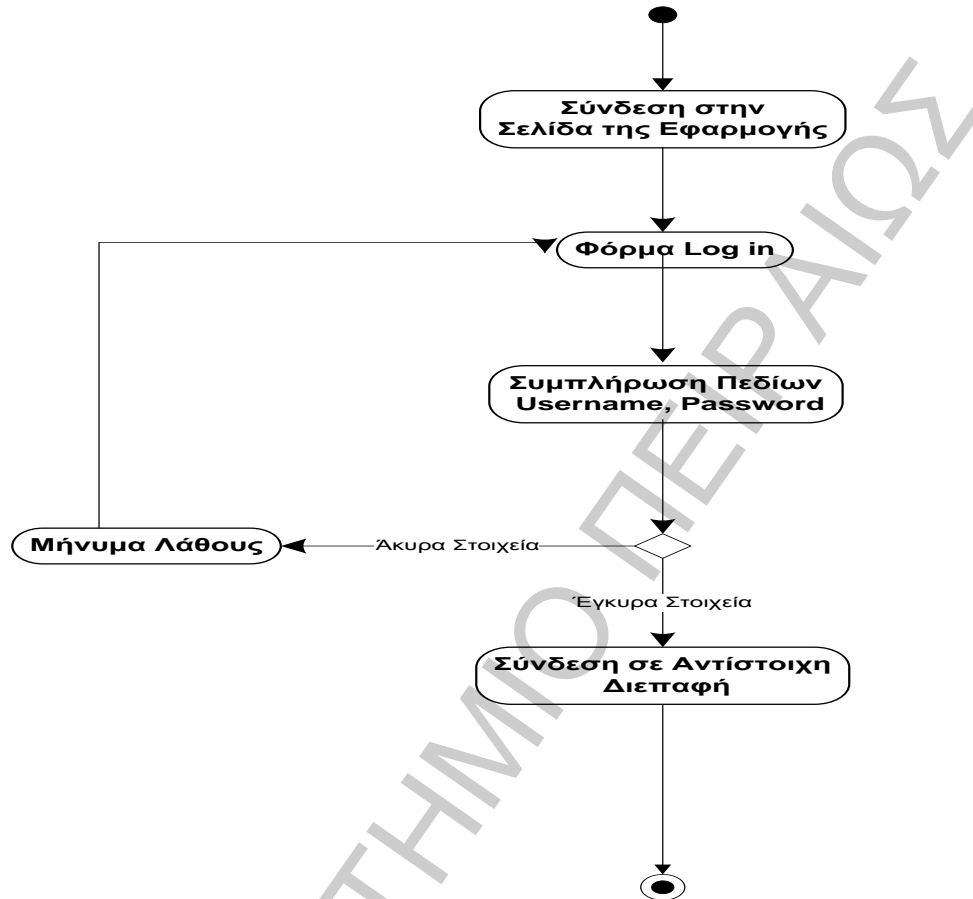
Για την έκδοση νέου ονόματος χρήστη (username) και κωδικού (password) πραγματοποιείται προηγουμένως έλεγχος διαθεσιμότητας στην βάση δεδομένων και αν το συγκεκριμένο όνομα χρήστη (username) που εισάγει κατά την εγγραφή ο administrator ή ο supervisor δεν χρησιμοποιείται, τότε καταχωρείται ο νέος χρήστης στην βάση δεδομένων, ολοκληρώνεται η εγγραφή του και αποστέλλεται ο απαραίτητος συνδυασμός για την σύνδεση (login) μέσω email στον εγγεγραμμένο πλέον χρήστη. Διαφορετικά εμφανίζεται ένα μήνυμα λάθους ώστε να επιλεχτεί ένα διαφορετικό όνομα χρήστη (username) από τον administrator ή από τον supervisor και να ολοκληρωθεί η εγγραφή. Η παραπάνω διαδικασία αναπαριστάται και στο επόμενο διάγραμμα ακολουθίας (Σχήμα 10), όπου περιγράφεται η χρονική αναπαράσταση των αλληλεπιδράσεων.



Σχήμα 10 Διάγραμμα Ακολουθίας για τον Σχεδιασμό της Διαδικασίας Εγγραφής Χρήστη (registration) που πραγματοποιείται είτε από τον Administrator είτε από τον Supervisor.

Στην συνέχεια κάθε εγγεγραμμένος χρήστης συνδέεται στην σελίδα της εφαρμογής μέσω του φυλλομετρητή (browser) της επιλογής του και συμπληρώνει τα πεδία όνομα χρήστη (username) και κωδικός (password) της φόρμας σύνδεσης (login) που υπάρχει στην αρχική σελίδα. Με την συμπλήρωση των πεδίων αυτών, πραγματοποιείται έλεγχος στην βάση δεδομένων για την εγκυρότητα των στοιχείων που καταχώρησε ο χρήστης και αν πράγματι τα στοιχεία είναι έγκυρα, τότε αυθεντικοποιείται ο χρήστης και συνδέεται στην αντίστοιχη διεπαφή που του επιτρέπεται (ανάλογα με τον ρόλο του, ο οποίος πιστοποιείται με την απλή εκχώρηση Εφαρμογή για Λειτουργίες Τμημάτων Αρωγής (Πολλαπλό σύστημα Helpdesk)

του username και του password). Σε διαφορετική περίπτωση εμφανίζεται στον χρήστη ένα μήνυμα λάθους που του ζητάει να ξαναπροσπαθήσει, συμπληρώνοντας πάλι την αντίστοιχη φόρμα. Η διαδικασία αυτή αναπαριστάται και μέσω ενός διαγράμματος δραστηριοτήτων (Σχήμα 11) που παρουσιάζεται παρακάτω.



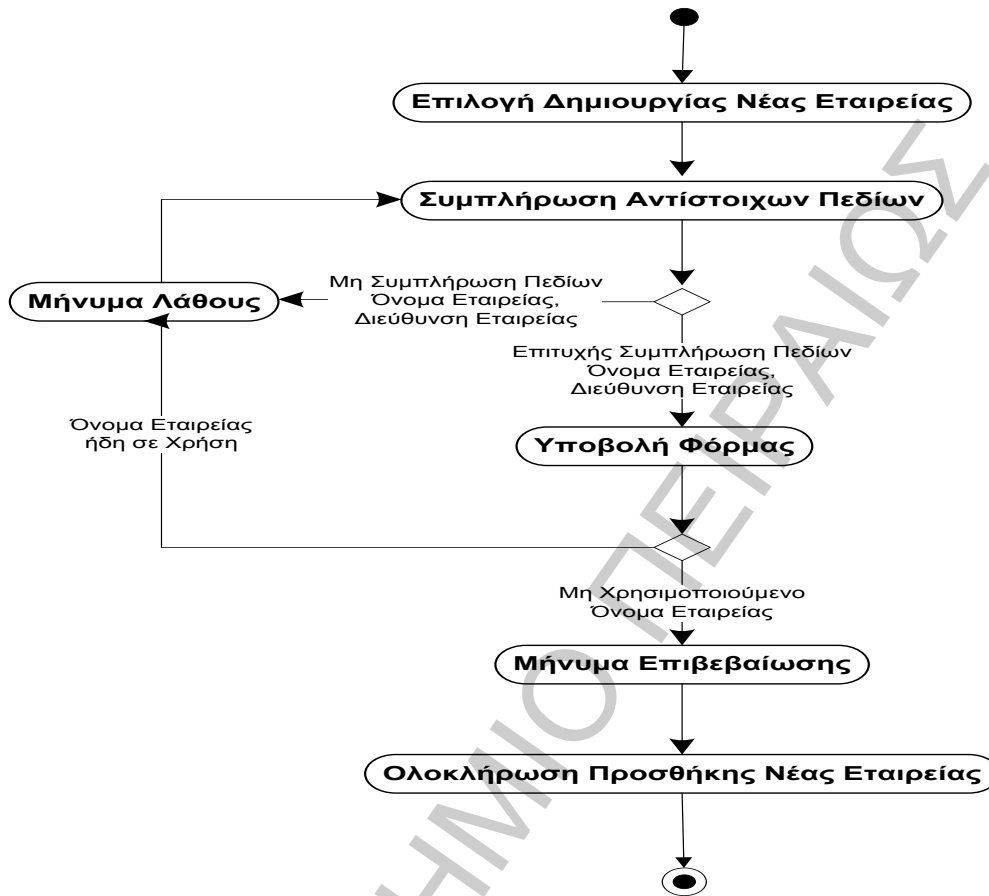
Σχήμα 11 Διάγραμμα Δραστηριοτήτων για τον Σχεδιασμό της Διαδικασίας Login.

2.3.3 Σχεδιασμός των Διαδικασιών: Δημιουργία Προφίλ Νέας Εταιρείας και Δημιουργία Νέου Τμήματος - Department

Επίσης τόσο για την δημιουργία προφίλ νέας εταιρείας από τον διαχειριστή (administrator) όσο και για την προσθήκη νέου τμήματος (department) είτε από τον διαχειριστή (administrator) είτε από τον προϊστάμενο (supervisor) σχεδιάστηκε να πραγματοποιούνται κάθε φορά απαραίτητοι έλεγχοι.

Για την προσθήκη νέας εταιρείας πραγματοποιείται αρχικά έλεγχος ως προς την συμπλήρωση των πεδίων με το όνομα της εταιρείας και την διεύθυνση της εταιρείας, τα οποία πεδία σχεδιάστηκαν ως υποχρεωτικά για συμπλήρωση. Εάν ο διαχειριστής (administrator) δεν συμπληρώσει ένα (ή και τα δύο) από τα υποχρεωτικά πεδία αυτά, τότε εμφανίζεται ένα μήνυμα λάθους που τον καθοδηγεί να συμπληρώσει τα πεδία, ενώ σε διαφορετική περίπτωση η προσθήκη της εταιρείας ολοκληρώνεται χωρίς κανένα πρόβλημα. Στην συνέχεια εφόσον έχει υποβληθεί η φόρμα για την προσθήκη της νέας εταιρείας, πραγματοποιείται έλεγχος στην βάση δεδομένων όπου αν υπάρχει ήδη καταχωρημένη εταιρεία με ίδιο όνομα με αυτό που εισήγαγε ο διαχειριστής στο αντίστοιχο πεδίο, εμφανίζεται ένα μήνυμα λάθους που ενημερώνει τον διαχειριστή για την καταχώρηση αυτή, ενώ αν η εισαγωγή του ονόματος της νέας εταιρείας είναι σωστή τότε η προσθήκη της νέας εταιρείας πραγματοποιείται με επιτυχία και εμφανίζεται ένα

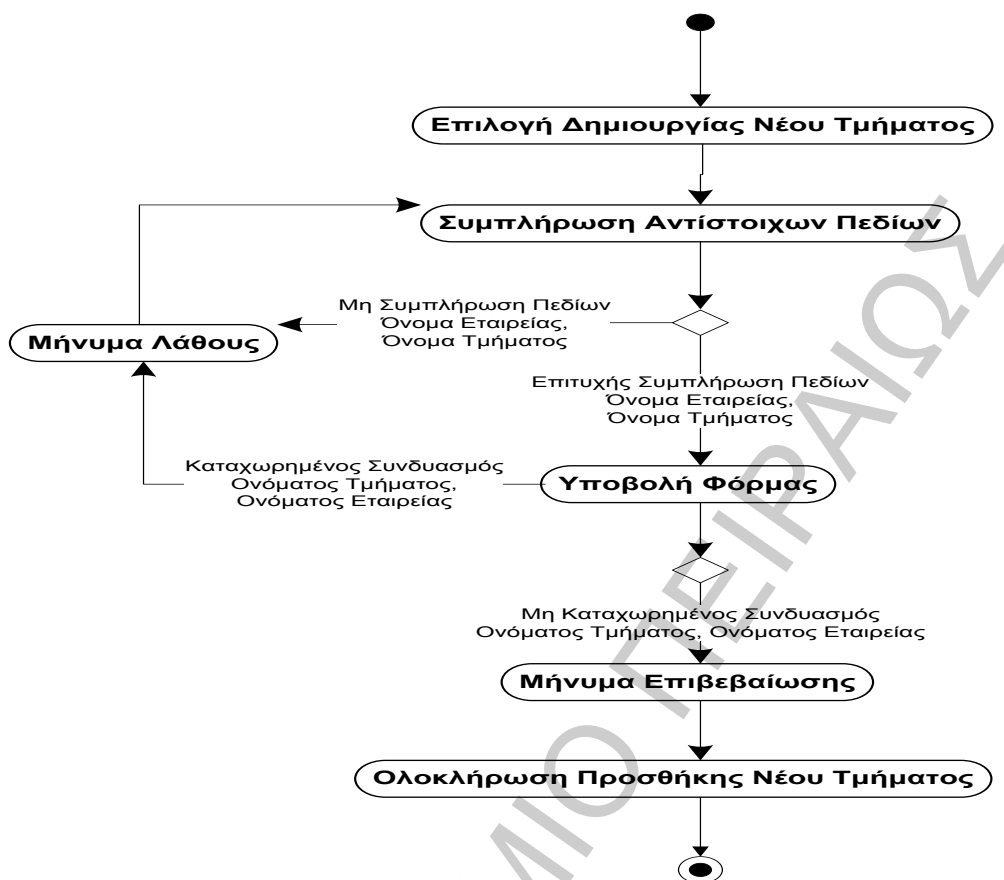
μήνυμα επιβεβαίωσης. Η διαδικασία αυτή αναπαριστάται και διαγραμματικά από το παρακάτω σχήμα (Σχήμα 12).



Σχήμα 12 Σχεδιασμός της Διαδικασίας Δημιουργίας Προφίλ Νέας Εταιρείας που πραγματοποιείται από τον Administrator.

Παρομοίως κατά την διαδικασία της προσθήκης νέου τμήματος, ο έλεγχος πραγματοποιείται αρχικά στα πεδία με το όνομα του τμήματος και το όνομα της εταιρείας στην οποία αντιστοιχεί το εκάστοτε τμήμα, όπου τα πεδία αυτά στην φόρμα αυτή σχεδιάστηκαν ως υποχρεωτικά για συμπλήρωση. Εάν κατά την διαδικασία της προσθήκης νέου τμήματος τα πεδία αυτά (ένα ή και τα δύο) δεν συμπληρωθούν από τον χρήστη, τότε εμφανίζεται ένα μήνυμα λάθους που ενημερώνει τον χρήστη για την υποχρεωτική συμπλήρωση των πεδίων αυτών, ενώ διαφορετικά η διαδικασία ολοκληρώνεται κανονικά.

Ακολούθως μετά την υποβολή της φόρμας για την προσθήκη νέου τμήματος, πραγματοποιείται ακόμα έλεγχος στην βάση δεδομένων, όπου αν ο συνδυασμός του ονόματος τμήματος και του ονόματος της εταιρείας στην οποία ανήκει υπάρχει ήδη καταχωρημένος, τότε εμφανίζεται ένα μήνυμα λάθους ως προς ενημέρωση του χρήστη, αλλιώς εμφανίζεται ένα μήνυμα επιβεβαίωσης για την επιτυχή προσθήκη του νέου τμήματος. Η διαδικασία που μόλις περιγράφηκε περιγράφεται και στο ακόλουθο διάγραμμα δραστηριοτήτων (Σχήμα 13).



Σχήμα 13 Σχεδιασμός της Διαδικασίας Δημιουργία Νέου Τμήματος που πραγματοποιείται είτε από τον Administrator είτε από τον Supervisor.

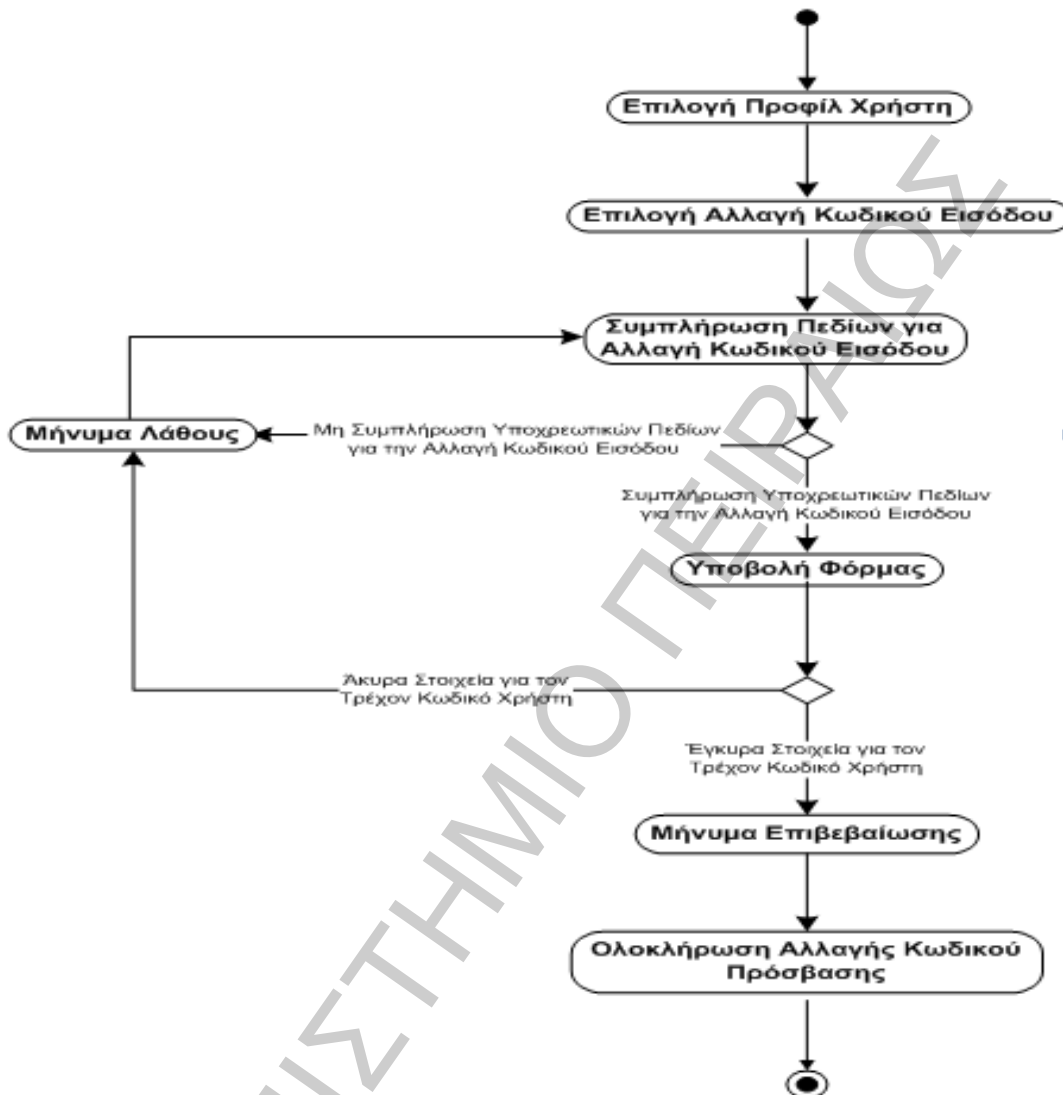
2.3.4 Σχεδιασμός για την Διαδικασία Αλλαγής Κωδικού Χρήστη

Όλοι οι συνδεδεμένοι χρήστες της εφαρμογής, των οποίων ο λογαριασμός έχει δημιουργηθεί είτε από διαχειριστή (administrator) είτε από προϊστάμενο (supervisor) και τους έχει αποσταλεί το όνομα χρήστη (username) και ο κωδικός (password) στον λογαριασμό email τους, έχουν την δυνατότητα διαχειρίζοντας το εκάστοτε προφίλ τους να αλλάξουν τον κωδικό χρήστη (password) που τους έχει αποσταλεί και αντί αυτού να χρησιμοποιούν κάποιον άλλον της προτίμησής τους. Κατά την διαχείριση του προφίλ, οι χρήστες δεν μπορούν να τροποποιήσουν τα προσωπικά στοιχεία του λογαριασμού τους που εμφανίζονται στο προφίλ, παρά μόνο τον κωδικό χρήστη – password.

Στην διαδικασία αλλαγής κωδικού χρήστη, αρχικά πραγματοποιείται έλεγχος για τον αν τα πεδία: τρέχων κωδικός εισόδου, νέος κωδικός χρήστη και επιβεβαίωση νέου κωδικού χρήστη συμπληρώνονται από τον κάθε χρήστη. Αν κάποιο από τα παραπάνω πεδία (ή ακόμα και όλα) δεν συμπληρωθεί από τον χρήστη τότε εμφανίζεται κάποιο μήνυμα λάθους που ενημερώνει τον χρήστη για την υποχρεωτική συμπλήρωση των πεδίων αυτών, διαφορετικά η διαδικασία για την αλλαγή κωδικού πρόσβασης ολοκληρώνεται κανονικά. Μετά την υποβολή της αντίστοιχης φόρμας, πραγματοποιείται ακολούθως έλεγχος στην βάση δεδομένων για το αν το πεδίο που αφορά τον τρέχον κωδικό χρήστη έχει συμπληρωθεί με έγκυρα στοιχεία, δηλαδή με τον σωστό κωδικό πρόσβασης (password) που έχει αποσταλεί στον κάθε χρήστη με την ολοκλήρωση της εγγραφής του.

Εάν ο τρέχων κωδικός πρόσβασης που εισάγει ο χρήστης είναι έγκυρος, τότε εμφανίζεται ένα μήνυμα επιβεβαίωσης και η ενέργεια της αλλαγής κωδικού χρήστη πραγματοποιείται με επιτυχία, αλλιώς αν τα στοιχεία στο συγκεκριμένο πεδίο δεν πιστοποιηθούν ως προς την εγκυρότητα τους, τότε εμφανίζεται ένα μήνυμα λάθους προς ενημέρωση του χρήστη για την

αδυναμία της αλλαγής του κωδικού εισόδου του. Η παραπάνω διαδικασία αναπαριστάται μέσω του επόμενου διαγράμματος δραστηριοτήτων (Σχήμα 14).



Σχήμα 14 Διάγραμμα Δραστηριοτήτων για την Διαδικασία Αλλαγής Κωδικού Χρήστη.

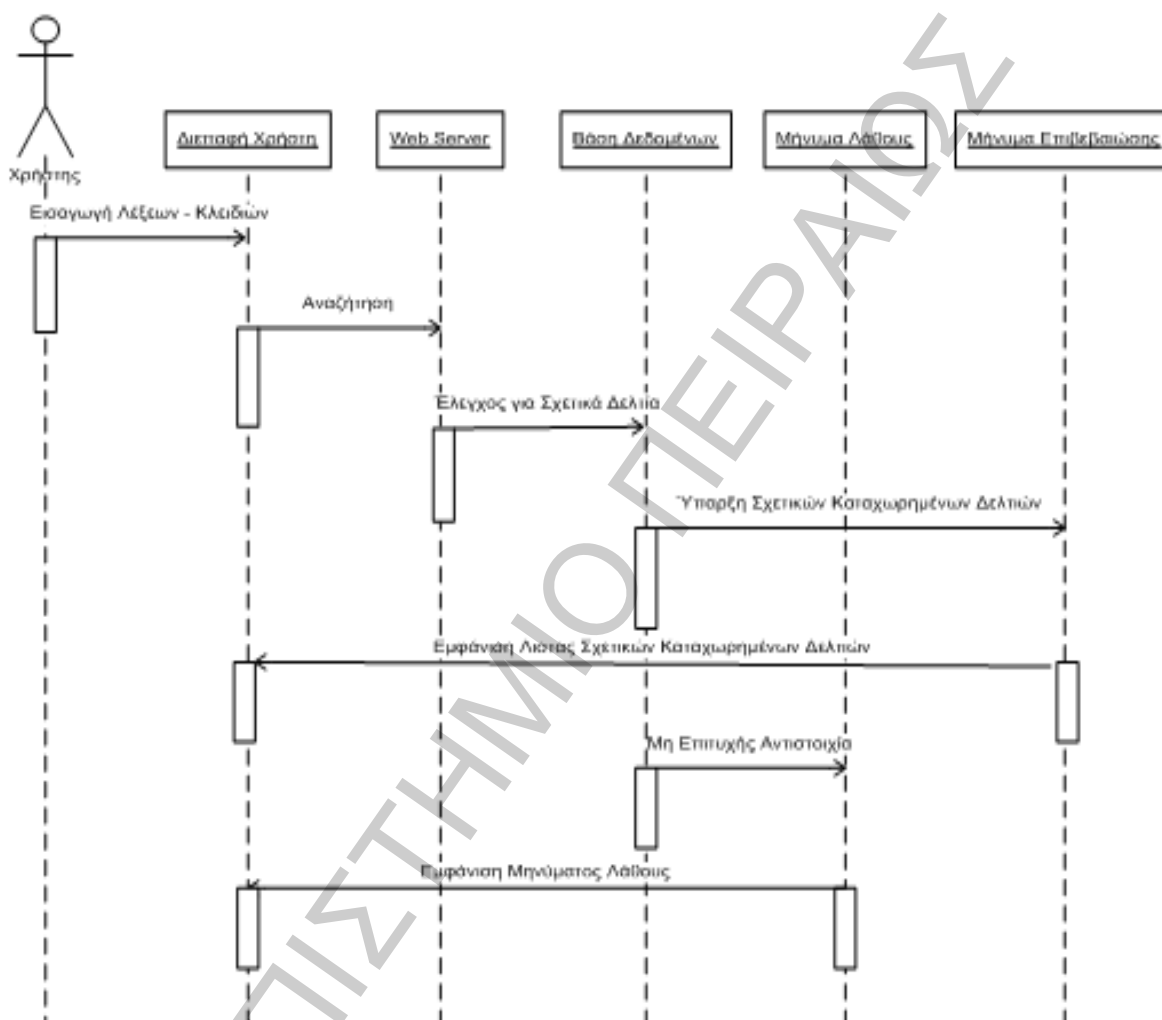
2.3.5 Σχεδιασμός για την Διαδικασία Αναζήτησης – Search

Κάθε εγγεγραμμένος χρήστης που είναι συνδεδεμένος στην εφαρμογή έχει την δυνατότητα να προβάλει όλα τα τεχνικά δελτία καταγραφής είτε αυτά είναι ανοιχτά είτε είναι κλειστά από την αντίστοιχη λίστα που υπάρχει στο βασικό μενού της εκάστοτε διεπαφής. Επίσης η δυνατότητα προβολής κάποιου τεχνικού δελτίου καταγραφής παρέχεται στον χρήστη και από την διαδικασία αναζήτησης – search.

Ο χρήστης εισάγει κάποιες λέξεις – κλειδιά στο πεδίο της αναζήτησης – search που προβλέπουν είτε τον τίτλο – ονομασία του δελτίου καταγραφής που τον ενδιαφέρει είτε την περιγραφή του και γίνεται έλεγχος στην βάση δεδομένων για το αν υπάρχουν καταχωρημένα δελτία καταγραφής σχετικά τις λέξεις - κλειδιά που εισήγαγε ο χρήστης. Εάν εντοπιστούν δελτία καταγραφής που είτε στον τίτλο τους είτε στην περιγραφή τους περιέχουν αυτές τις λέξεις – κλειδιά, τότε εμφανίζεται μία λίστα με τα σχετικά δελτία καταγραφής ενώ διαφορετικά εμφανίζεται ένα μήνυμα λάθους που πληροφορεί τον χρήστη για την αποτυχία της αντιστοίχας των λέξεων – κλειδιών με τα καταχωρημένα δελτία καταγραφής στο σύστημα. Το διάγραμμα

ακολουθίας που παρατίθεται στην συνέχεια αναπαριστά την διαδικασία αναζήτησης για τα δελτία καταγραφής που μόλις περιγράφηκε (Σχήμα 15).

Σημειώνεται ότι η διαδικασία της αναζήτησης πραγματοποιείται αντίστοιχα και για τις υπόλοιπες εγγραφές: εταιρείες, πελάτες, χρήστες, τμήματα εταιρειών, κατηγορίες δελτίων καταγραφής.



Σχήμα 15 Διάγραμμα Ακολουθίας για τον Σχεδιασμό της Διαδικασίας Αναζήτησης.

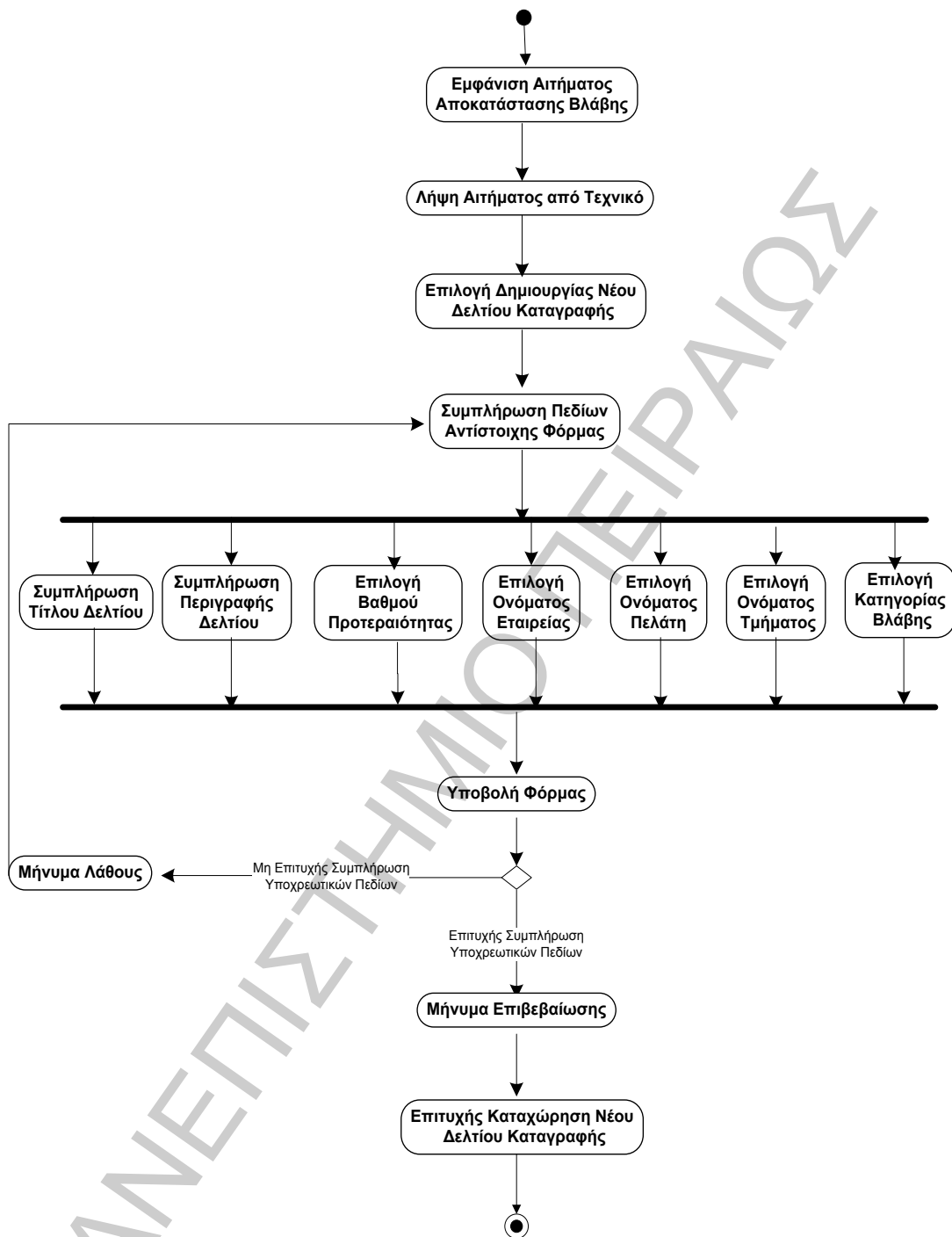
2.3.6 Σχεδιασμός για την Δημιουργία ενός Νέου Δελτίου Καταγραφής, σε περίπτωση Εμφάνισης Αιτήματος Αποκατάστασης Βλάβης καθώς και για της Διαχείριση του

Στο σημείο αυτό είναι σημαντικό να δοθεί έμφαση στον σχεδιασμό των τεχνικών δελτίων καταγραφής, τα οποία αποτελούν και το βασικό κομμάτι της εφαρμογής, μιας και μέσω αυτών θα καταγράφονται και θα εξυπηρετούνται τα αιτήματα αποκατάστασης βλάβης που θα προέρχονται από εσωτερικά τμήματα της εταιρείας ή από εξωτερικούς πελάτες και συνεργαζόμενες εταιρείες.

Αρχικά σχεδιάστηκε η διαδικασία που θα ακολουθείται για την καταχώρηση ενός νέου δελτίου καταγραφής σε πιθανή εμφάνιση κάποιου αιτήματος βλάβης που θα απαιτεί την εξυπηρέτηση του από κάποιο τεχνικό. Ο τεχνικός που λαμβάνει το αίτημα (τηλεφωνικά ή με e-mail) για αποκατάσταση βλάβης είναι υπεύθυνος για την επίλυσή του και δημιουργεί ένα νέο δελτίο καταγραφής για να καταχωρήσει το αίτημα αυτό.

Με την επιλογή της δημιουργίας ενός νέου δελτίου καταγραφής, τα απαραίτητα στοιχεία του δελτίου που πρέπει να συμπληρωθούν από τον τεχνικό είναι τα πεδία : τίτλος δελτίου όπου δίνεται ενδεικτικά μία ονομασία για το νέο δελτίο, περιγραφή δελτίου όπου προστίθεται μια σύντομη περιγραφή που αντιστοιχεί στο περιεχόμενο της βλάβης-δελτίου, προτεραιότητα όπου δίνεται ένα βαθμός προτεραιότητας (χαμηλή, μεσαία, υψηλή) ανάλογα με την κρίση του τεχνικού, όνομα εταιρείας και όνομα τμήματος στα οποία ανήκει ο τεχνικός που έλαβε το αίτημα αποκατάστασης βλάβης, όνομα πελάτη το οποίο απευθύνεται στον πελάτη που δήλωσε το πρόβλημα βλάβης και κατηγορία βλάβης όπου κατατάσσεται η προς εξέταση βλάβη σε μία από τις διαθέσιμες καταχωρημένες κατηγορίες.

Όπως ήδη αναφέρθηκε όλα τα παραπάνω πεδία της φόρμας για την δημιουργία νέου δελτίου καταγραφής είναι υποχρεωτικά για συμπλήρωση. Επομένως σχεδιάστηκε να πραγματοποιείται έλεγχος ως προς την συμπλήρωση των πεδίων μετά την υποβολή της συγκεκριμένης φόρμας. Εάν κάποιο από τα πεδία της φόρμας δεν συμπληρωθεί από τον τεχνικό, τότε εμφανίζεται ένα μήνυμα λάθους που τον ενημερώνει για την υποχρεωτική συμπλήρωση, διαφορετικά το νέο δελτίο καταγραφής καταχωρείται χωρίς κανένα πρόβλημα και εμφανίζεται ένα μήνυμα επιβεβαίωσης. Για την διαδικασία αυτή κατασκευάστηκε ένα διάγραμμα δραστηριοτήτων (Σχήμα 16) ώστε να διακρίνονται παράλληλα και τα απαραίτητα στοιχεία καταχώρησης για την δημιουργία ενός νέου δελτίου.



Σχήμα 16 Σχεδιασμός της Διαδικασίας Δημιουργίας Νέου Δελτίου Καταγραφής σε εμφάνιση Αιτήματος Αποκατάστασης Βλάβης

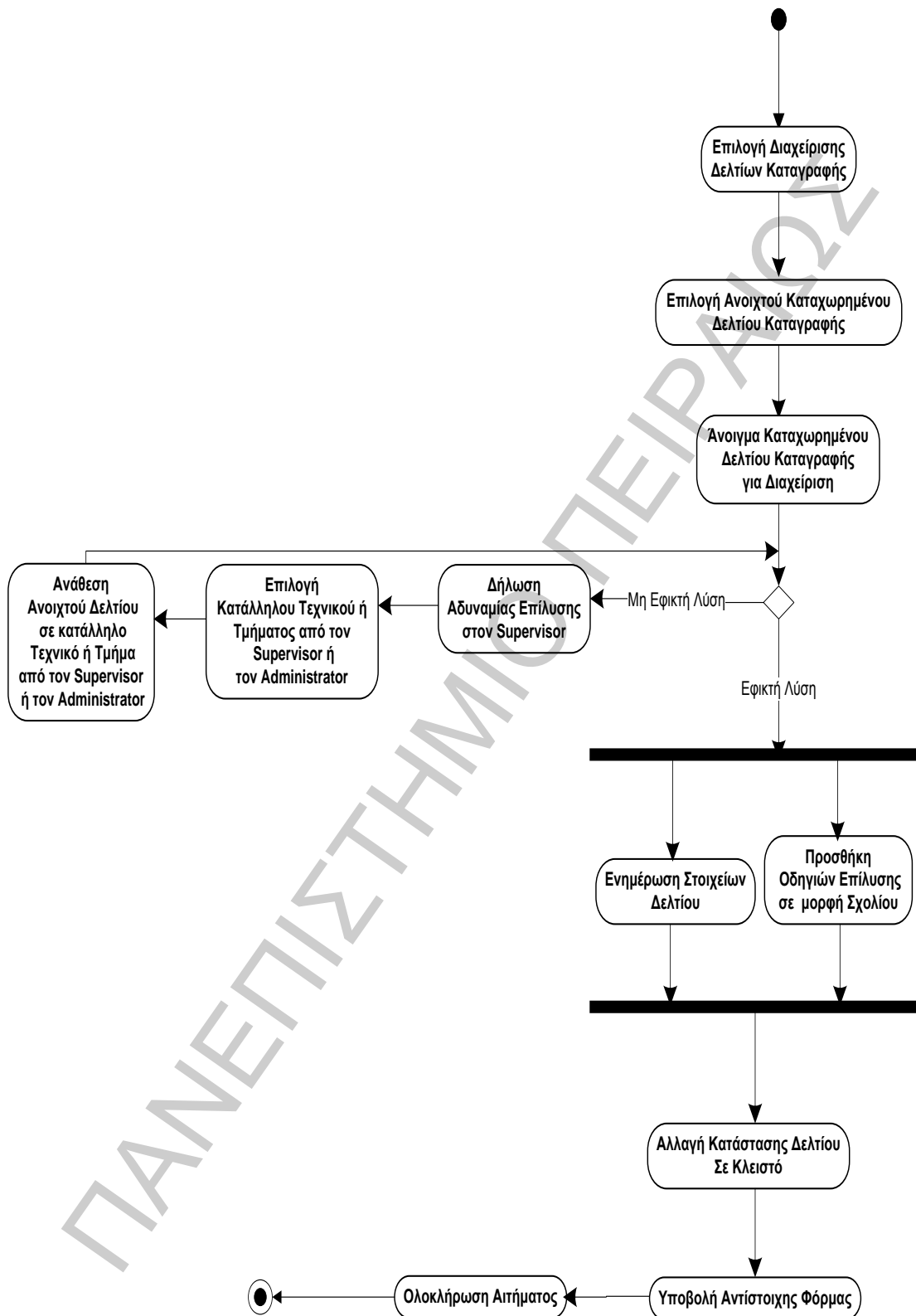
Στην συνέχεια σχεδιάστηκε η διαδικασία της διαχείρισης των ήδη καταχωρημένων δελτίων καταγραφής. Εφόσον έχει καταχωρηθεί με επιτυχία το νέο δελτίο καταγραφής, είτε ο τεχνικός που είναι στην ανάθεση του το δελτίο αυτό (δηλαδή ο τεχνικός που δημιούργησε και καταχώρησε το δελτίο) είτε κάποιος άλλος χρήστης (τεχνικός, προϊστάμενος ή διαχειριστής) που

έχει δικαίωμα να προβάλλει το δελτίο αυτό, μπορεί να το ανοίξει ώστε να το διαχειριστεί και να προσφέρει μια πιθανή λύση.

Ανοίγοντας ένα ήδη καταχωρημένο δελτίο όπου δεν έχει ακόμα εξυπηρετηθεί (δηλαδή η κατάσταση του συγκεκριμένου δελτίου είναι ανοιχτό), ο χρήστης έχει την δυνατότητα να ενημερώσει τα στοιχεία του δελτίου κάνοντας κάποιες τροποποιήσεις και να προσθέσει σε μορφή σχολίου οδηγίες για την επίλυση της βλάβης και την εξυπηρέτηση του αιτήματος ώστε το δελτίο να κλείσει. Επίσης σε περίπτωση που ο χρήστης που ανοίγει το ήδη καταχωρημένο δελτίο έχει ρόλο προϊστάμενου ή διαχειριστή μπορεί επιπλέον να κάνει ανάθεση του δελτίου αυτού σε κάποιον άλλον τεχνικό ή σε άλλον προϊστάμενο διαφορετικού τμήματος.

Συγκεκριμένα η διαδικασία της διαχείρισης ενός καταχωρημένου δελτίου καταγραφής που δεν έχει εξυπηρετηθεί περιγράφεται ως εξής: Ο χρήστης επιλέγει το δελτίο καταγραφής που απαιτεί λύση και το ανοίγει ώστε να το διαχειριστεί. Σε πρώτη φάση πραγματοποιείται έλεγχος, όπου αν ο χρήστης κρίνει ότι μπορεί να προσφέρει λύση στο συγκεκριμένο αίτημα, τότε ενημερώνει τα στοιχεία του δελτίου καταγραφής, προσθέτει οδηγίες επίλυσης της βλάβης σε μορφή σχολίου, αλλάζει την κατάσταση του δελτίου καταγραφής ως κλειστό και υποβάλλει την φόρμα. Με την τελευταία ενέργεια αυτή, το ανοιχτό δελτίο κλείνει και ολοκληρώνεται το αίτημα.

Εναλλακτικά αν ο χρήστης, εφόσον επιλέξει και ανοίξει το προς εξέταση δελτίο καταγραφής κρίνει κατά τον έλεγχο ότι αδυνατεί να προσφέρει λύση στο συγκεκριμένο αίτημα, δηλώνει αδυναμία επίλυσης αιτήματος στον προϊστάμενο (supervisor) σε μορφή σχολίου και υποβάλλει την φόρμα, χωρίς να αλλάξει την κατάσταση του δελτίου. Ο προϊστάμενος ή ο διαχειριστής με την σειρά του, επιλέγει έναν άλλον τεχνικό από τους διαθέσιμους τεχνικούς του τμήματος και αναθέτει σε αυτόν το ανοιχτό δελτίο ή ακόμα μπορεί να αναθέσει το ανοιχτό αυτό δελτίο σε άλλο τμήμα - department δηλαδή σε διαφορετικό προϊστάμενο (supervisor). Ακολούθως πραγματοποιούνται οι ίδιες δραστηριότητες από την πλευρά του τεχνικού ή του προϊστάμενου που ανατέθηκε τώρα το δελτίο καταγραφής, με αυτές που περιγράφηκαν προηγουμένως μέχρι να κλείσει το ανοιχτό δελτίο και να ολοκληρωθεί το αίτημα. Για τον σχεδιασμό όλης της διαδικασίας που αναλύθηκε παραπάνω κατασκευάστηκε το επόμενο διάγραμμα δραστηριοτήτων (Σχήμα 17).



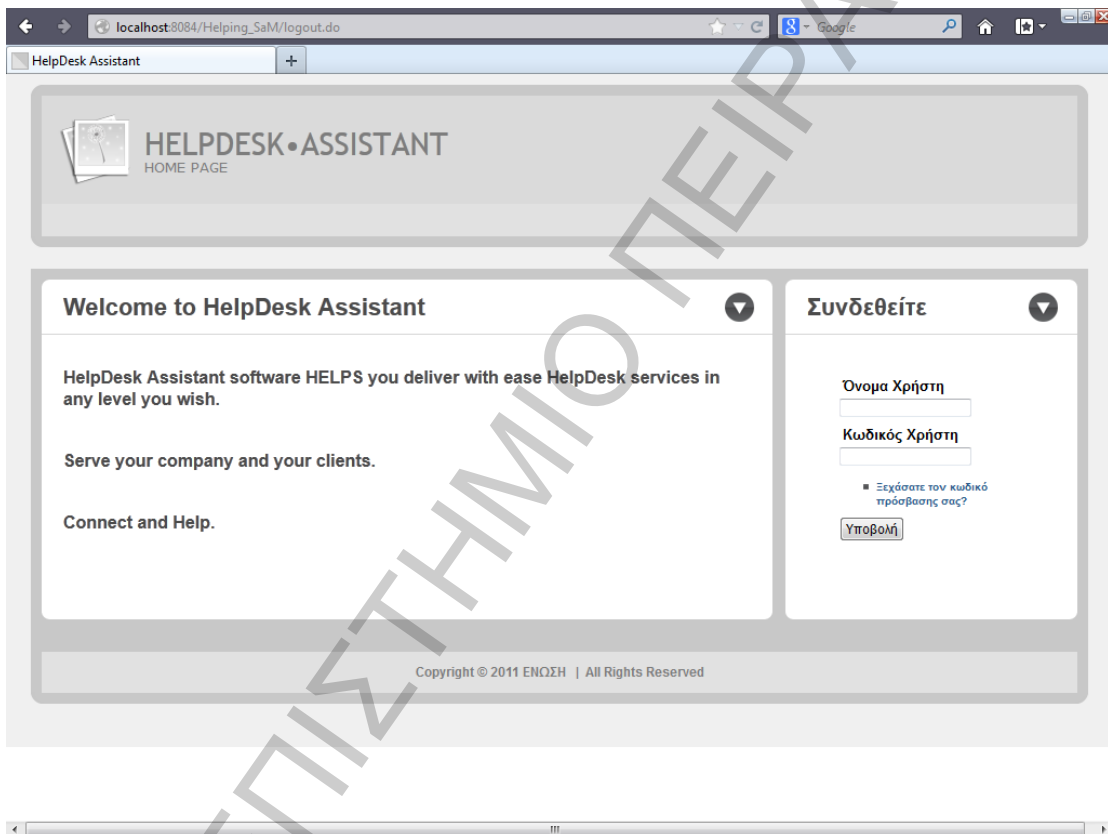
Σχήμα 17 Σχεδιασμός της Διαδικασίας Διαχείρισης Καταχωρημένου Ανοιχτού Δελτίου Καταγραφής για την Εξυπηρέτησή του.

3. Η Εφαρμογή

Στο κεφάλαιο αυτό παρουσιάζεται μέσα από κάποια screenshots η ίδια η εφαρμογή μας. Η παρουσίαση αυτή χρησιμεύει και ως εγχειρίδιο χρήστη (user manual).

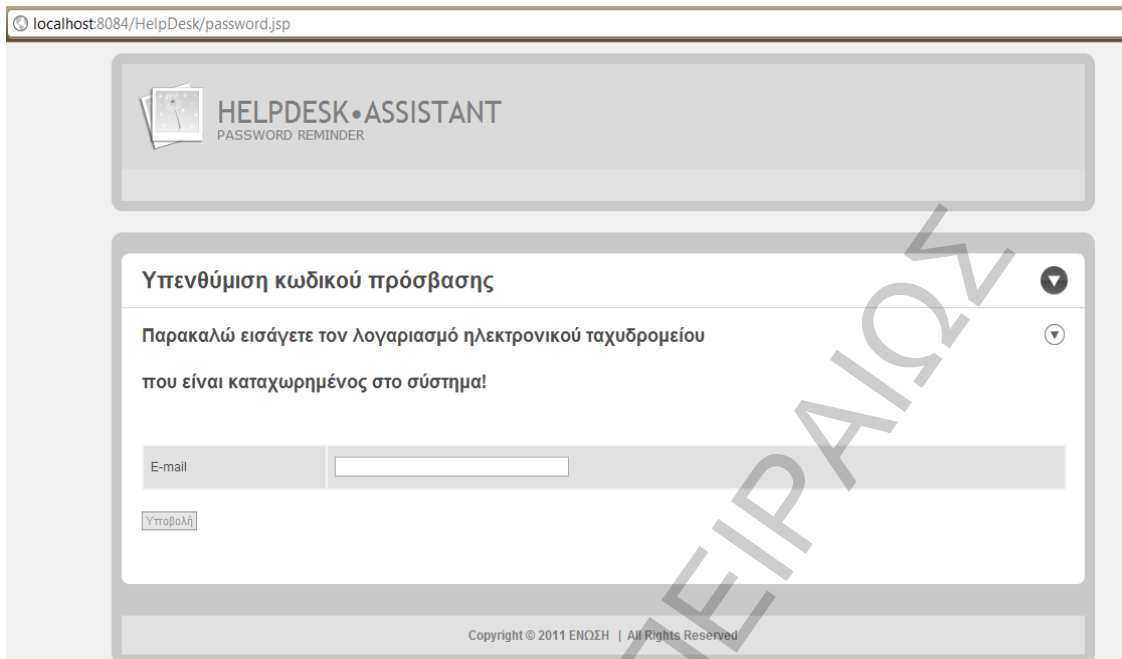
3.1 Αρχική σελίδα- διαδικασία αυθεντικοποίησης

Η αρχική κύρια σελίδα είναι μια απλή διεπαφή που προτρέπει τους χρήστες να συνδεθούν στην εφαρμογή δίνοντας όνομα και κωδικό χρήστη. Στη σελίδα αυτή έχει επίσης προβλεφθεί επιλογή για κάποιον που έχει ξεχάσει τον κωδικό πρόσβασης (Εικόνα 6).

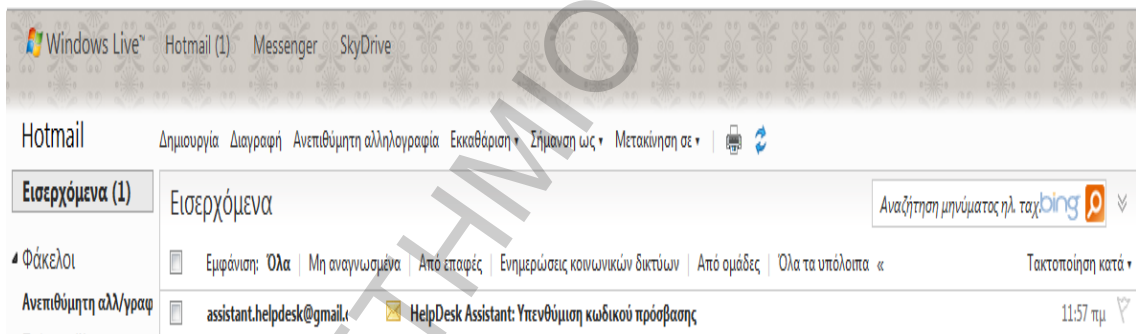


Εικόνα 6 Αρχική Σελίδα Εφαρμογής – Index Page.

Εάν ο χρήστης επιλέξει την επιλογή για την υπενθύμιση κωδικού πρόσβασης τότε οδηγείται στην παρακάτω διεπαφή (Εικόνα 7) όπου συμπληρώνοντας το αντίστοιχο πεδίο με τον έγκυρο λογαριασμό e-mail του, αποστέλλεται σε αυτόν ένα μήνυμα ενημέρωσης (e-mail) με τις παραμέτρους όνομα και κωδικό χρήστη για την σύνδεσή του (Εικόνα 8).

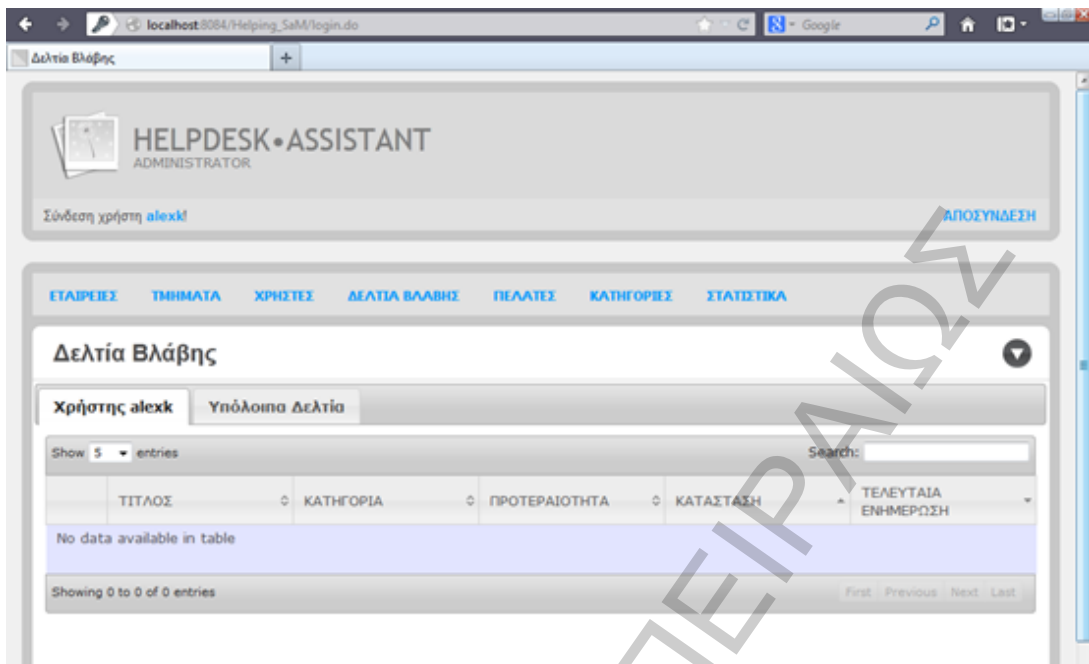


Εικόνα 7 Διεπαφή για Υπενθύμιση Κωδικού Πρόσβασης.

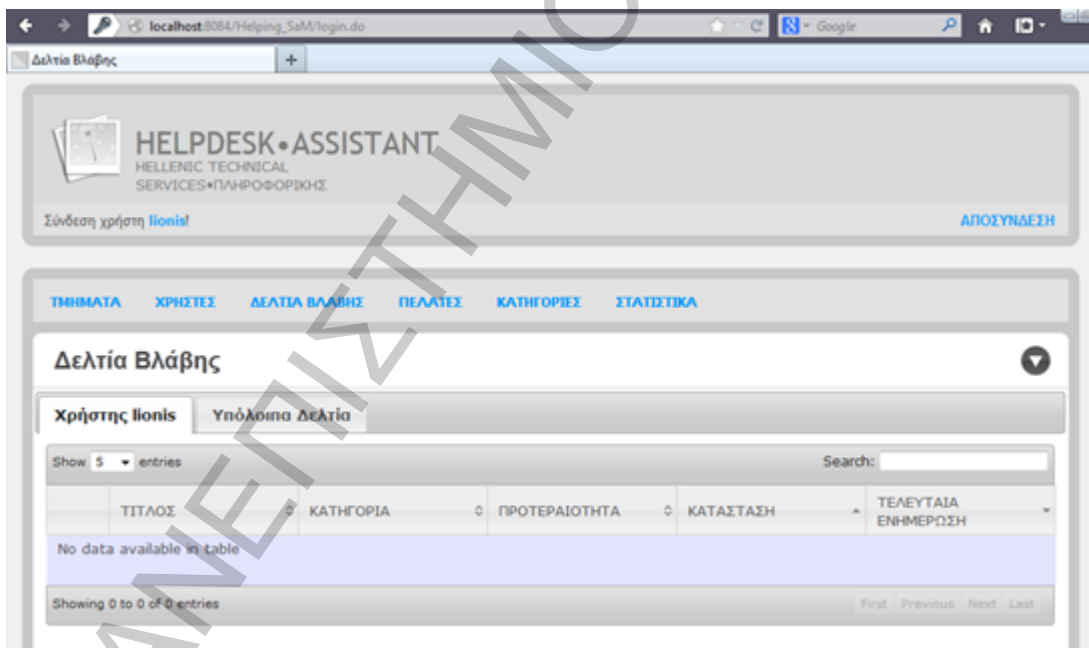


Εικόνα 8 Αποστολή E-mail για Υπενθύμιση Κωδικού Πρόσβασης.

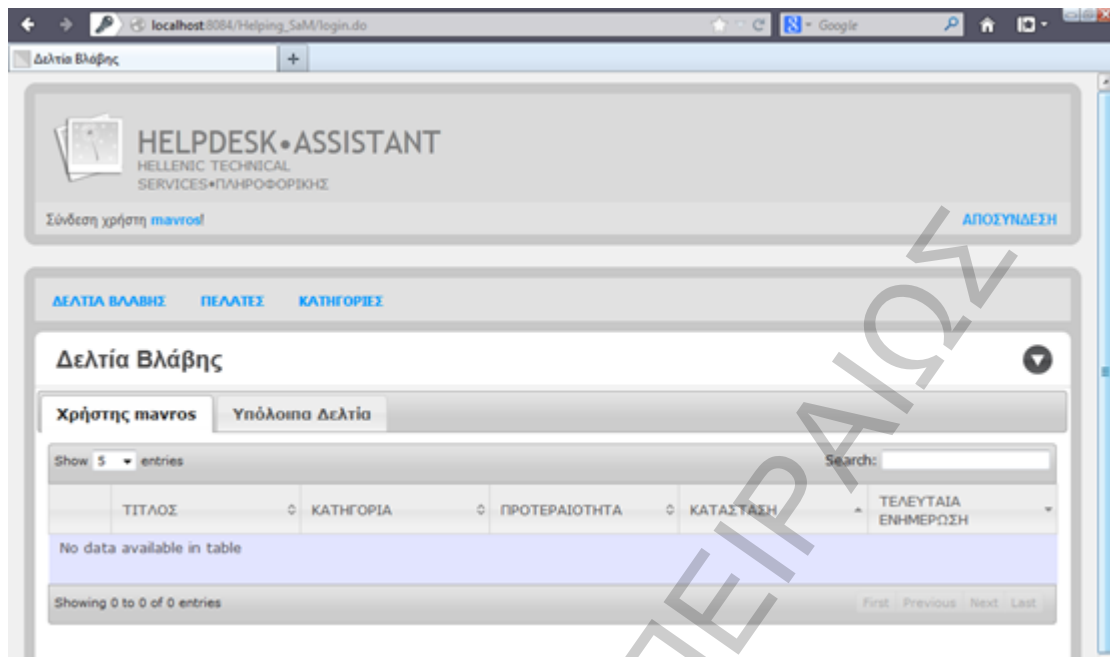
Διαφορετικά, όταν ο χρήστης δώσει τα στοιχεία του και συνδεθεί στην σελίδα που εμφανίζεται, ανοίγει αρχικά ο κατάλογος των τεχνικών δελτίων και ανάλογα με τα δικαιώματα που του έχουν οριστεί (ανάλογα με το ρόλο που έχει administrator, simple user, technician, supervisor), εμφανίζεται και διαφορετικό μενού επιλογών. Στις παρακάτω εικόνες εμφανίζονται οι διεπαφές των διάφορων χρηστών, με τη σειρά ως εξής: διαχειριστής, προϊστάμενος, τεχνικός, απλός χρήστης (αντίστοιχα Εικόνες 9 με 12).



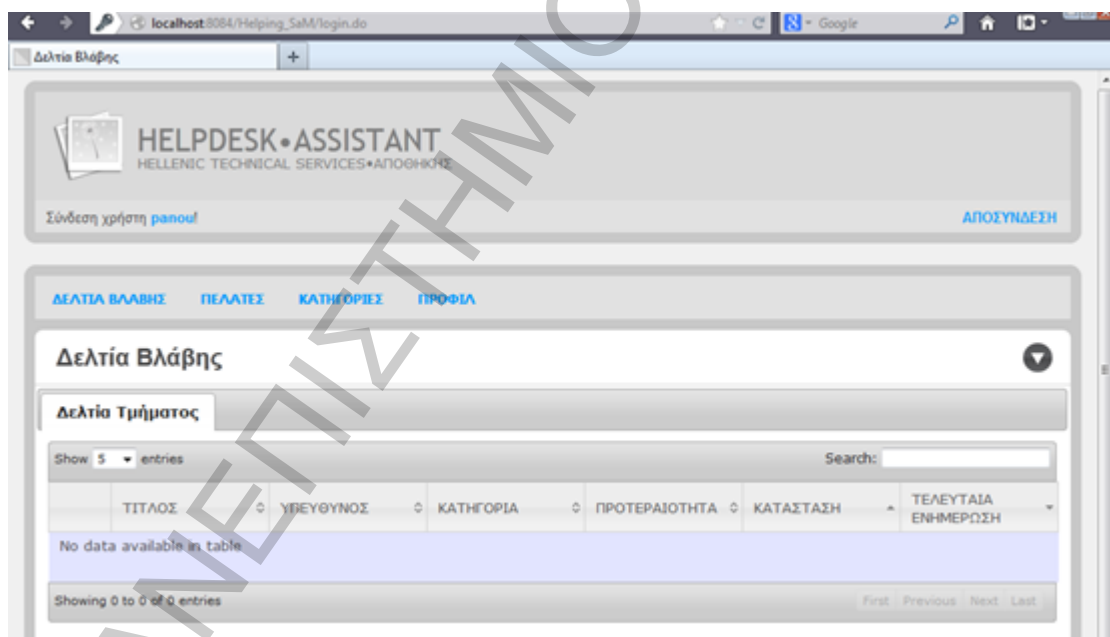
Εικόνα 9 Διεπαφή Διαχειριστή – Administrator.



Εικόνα 10 Διεπαφή Προϊστάμενου – Supervisor.



Εικόνα 11 Διεπαφή Τεχνικού – Technician.



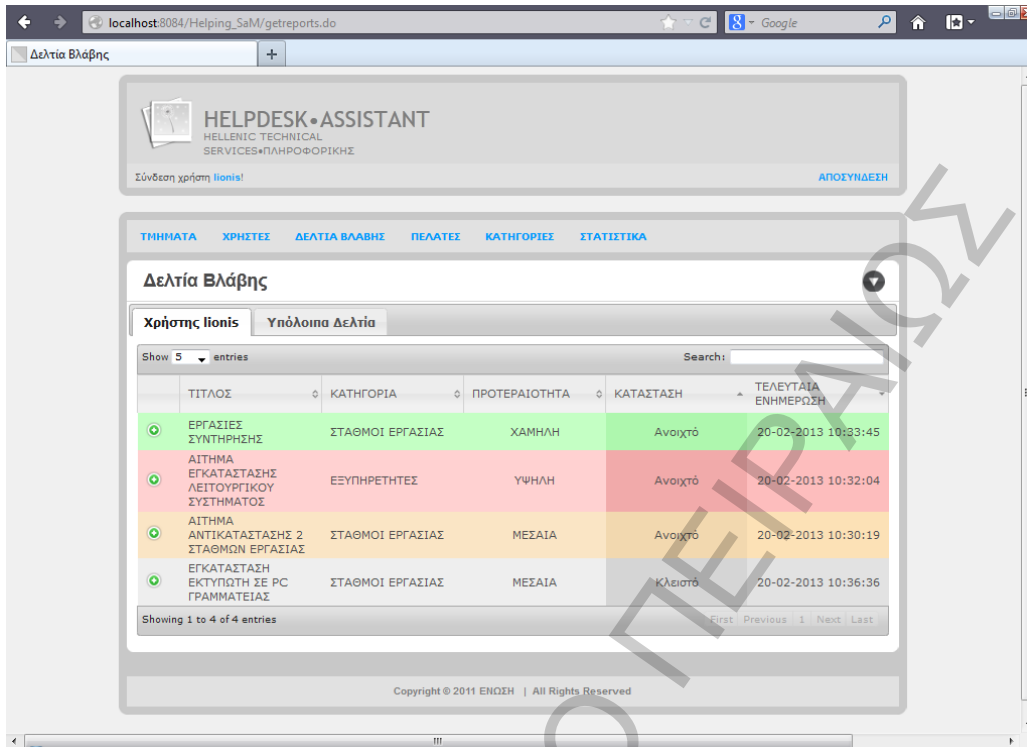
Εικόνα 12 Διεπαφή Απλού Χρήστη – Simple User

3.2 Ανάλυση μενού επιλογών

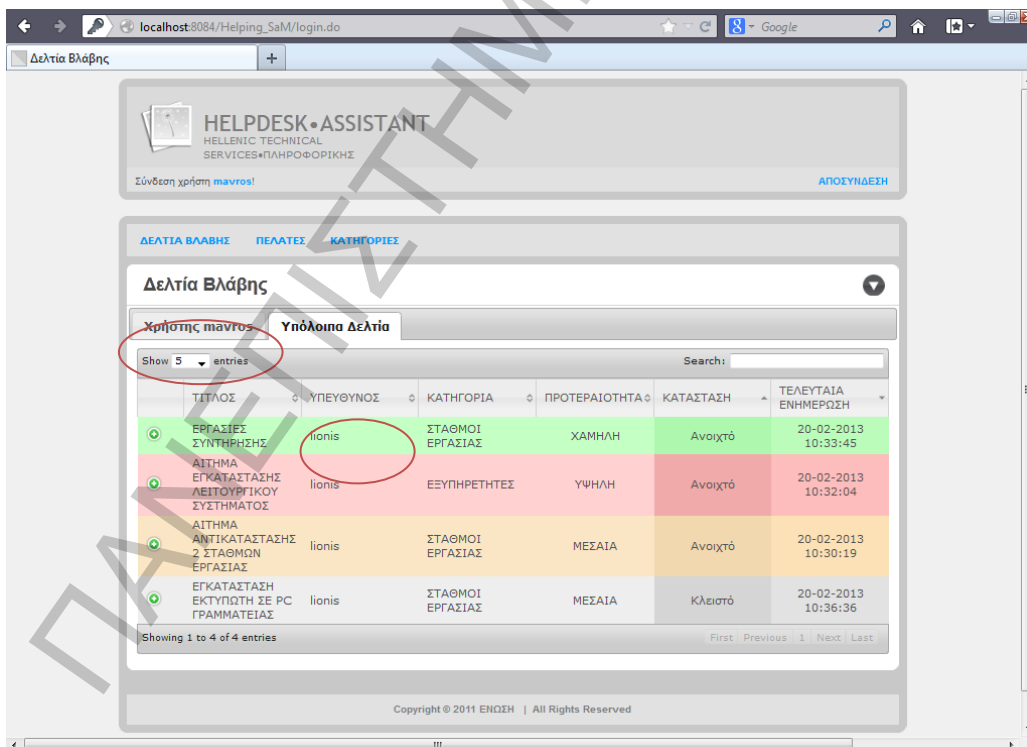
3.2.1 Εισαγωγή

Όπως είδαμε στην αρχική σελίδα εμφανίζονται τα δελτία του συνδεδεμένου χρήστη. (εικόνα 13). Στο tab «ΥΠΟΛΟΙΠΑ ΔΕΛΤΙΑ» εμφανίζονται όλα τα υπόλοιπα δελτία της εταιρείας που δεν ανήκουν στον χρήστη που έχει συνδεθεί, γι' αυτό και στην εγγραφή του δελτίου εμφανίζεται επιπλέον και ο υπεύθυνος τεχνικός ή προϊστάμενος (Εικόνα 14).

Εφαρμογή για Λειτουργίες Τμημάτων Αρωγής (Πολλαπλό σύστημα Helpdesk)



Εικόνα 13 Αρχική σελίδα χρήστη



Εικόνα 14 Υπόλοιπα Δελτία Καταγραφής.

Καλύπτοντας τις απαιτήσεις για ομοιομορφία, τα μενού «ΕΤΑΙΡΕΙΕΣ», «ΤΜΗΜΑΤΑ», «ΔΕΛΤΙΑ ΒΛΑΒΗΣ», «ΠΕΛΑΤΕΣ» και «ΚΑΤΗΓΟΡΙΕΣ» ακολουθούν την ίδια μορφοποίηση και Εφαρμογή για Λειτουργίες Τμημάτων Αρωγής (Πολλαπλό σύστημα Helpdesk)

τη λειτουργία, το κάθε ένα για το δικό του αντικείμενο, ενώ το μενού «ΧΡΗΣΤΕΣ» έχει και αυτό τις ίδιες επιλογές και λειτουργίες ενώ διαθέτει επιπλέον την επιλογή «ΠΡΟΦΙΛ», όπου ο συνδεδεμένος χρήστης μπορεί να δει τα στοιχεία του προφίλ του και να αλλάξει κωδικό πρόσβασης (Εικόνα 23). Το μενού που διαφοροποιείται περισσότερο από τα άλλα είναι αυτό των στατιστικών «ΣΤΑΤΙΣΤΙΚΑ» (εικόνα 37)

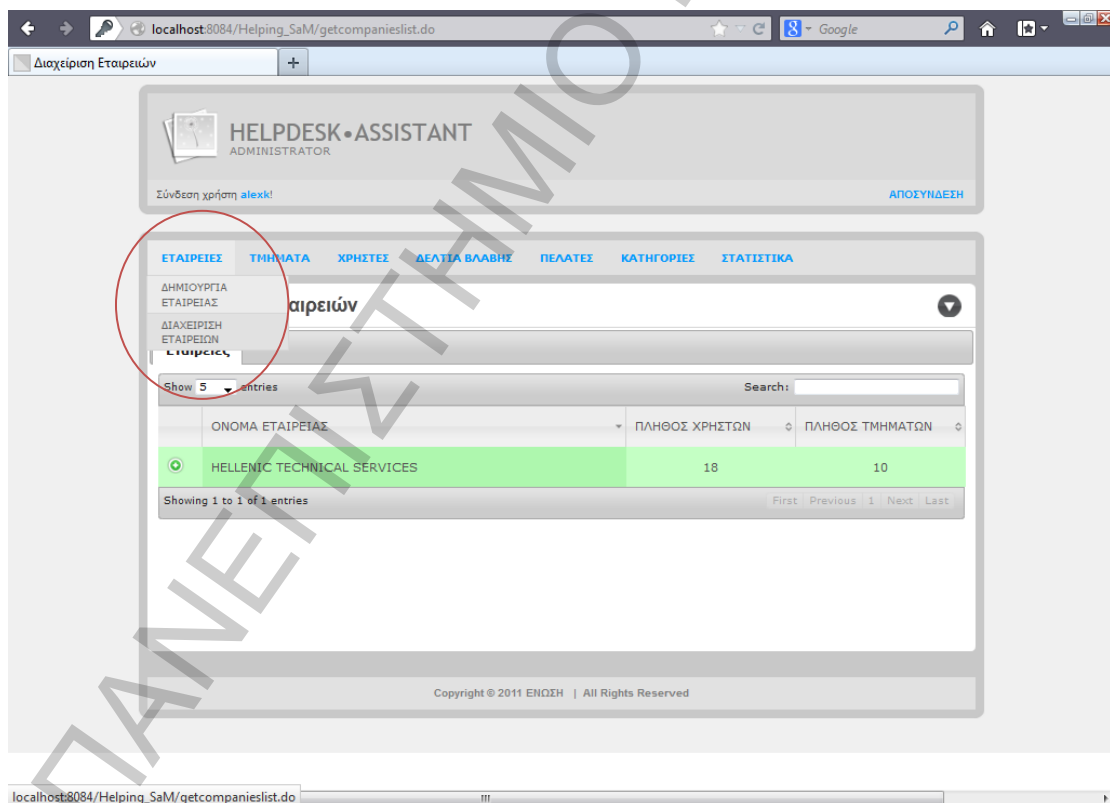
Ας δούμε σε αυτό το σημείο τις επιλογές του μενού της διεπαφής του διαχειριστή, μιας και αυτό καλύπτει το σύνολο των επιλογών της εφαρμογής, μέρος των οποίων είναι διαθέσιμα για τους υπόλοιπους χρήστες.

3.2.2 Στοιχείο Μενού: «ΕΤΑΙΡΕΙΕΣ»

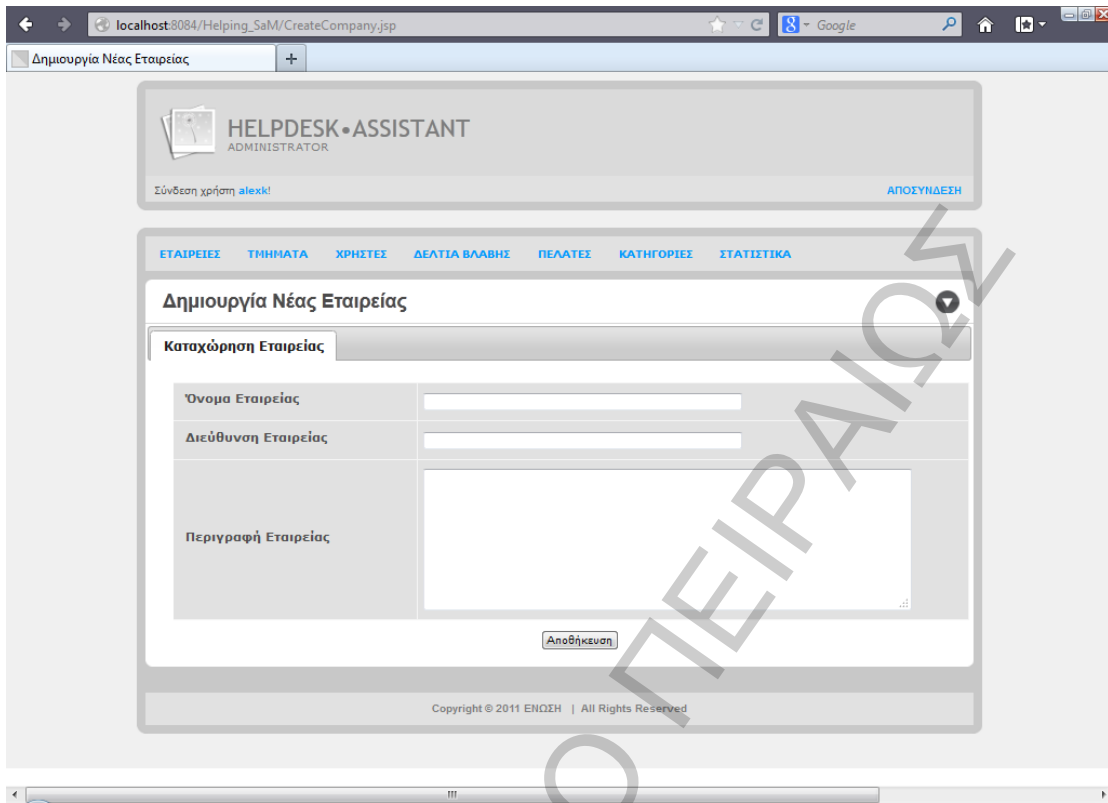
Το στοιχείο μενού «ΕΤΑΙΡΕΙΕΣ» διαθέτει υπομενού με δύο επιλογές «ΔΗΜΙΟΥΡΓΙΑ ΕΤΑΙΡΕΙΑΣ» και «ΔΙΑΧΕΙΡΙΣΗ ΕΤΑΙΡΕΙΩΝ» (Εικόνα 15).

Η πρώτη επιλογή δίνει φόρμα με τα πεδία που πρέπει να συμπληρωθούν ώστε να γίνει προσθήκη νέας εταιρείας (όνομα, δ/ση, περιγραφή). Η δεύτερη διαχειρίζεται τις ήδη υπάρχουσες εταιρείες. Στις εγγραφές που εμφανίζονται στη φόρμα διαχείρισης με διπλό κλικ στην εγγραφή της εταιρείας δίνεται δυνατότητα ενημέρωσης της εγγραφής. Με κλικ στο σχήμα με το συν (+) στην αρχή της εγγραφής εμφανίζονται για ανάγνωση τα στοιχεία της εγγραφής, ενώ δίνεται επιπλέον δυνατότητα διαγραφής, η επιλογή της οποίας απαιτεί επιβεβαίωση. (εικόνες 17, 18)

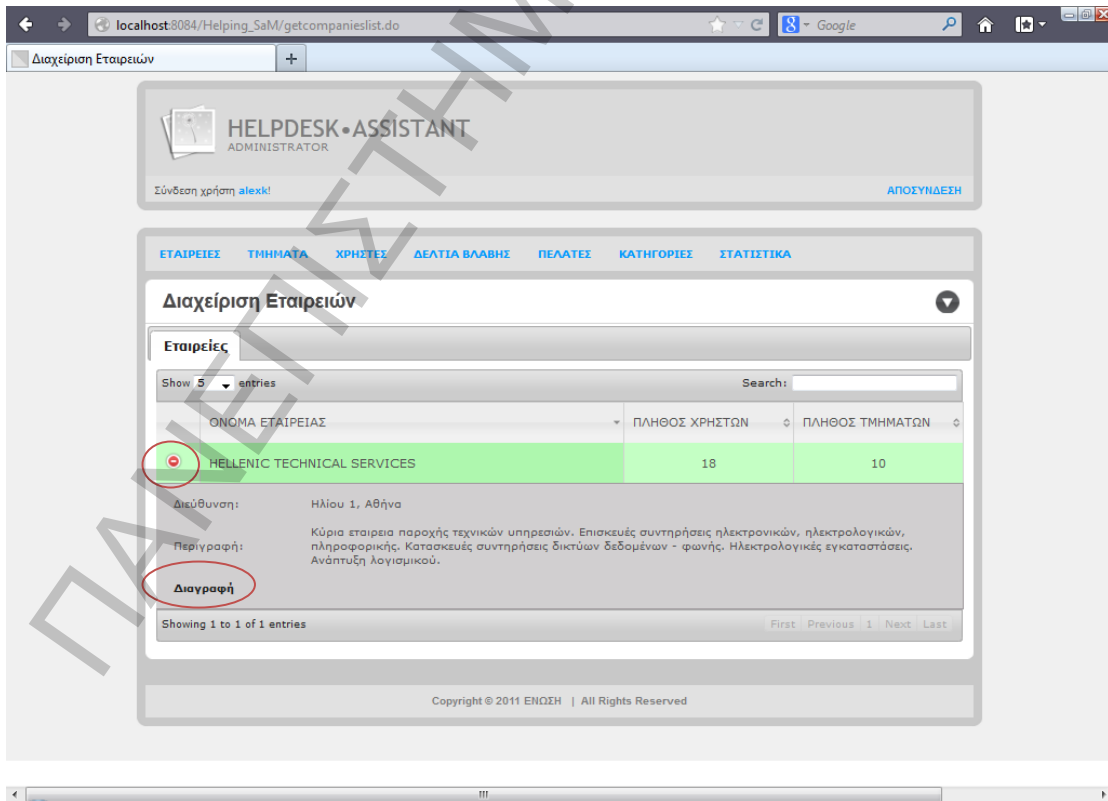
Τα παραπάνω ισχύουν για όλα τα στοιχεία του μενού εκτός των στατιστικών.



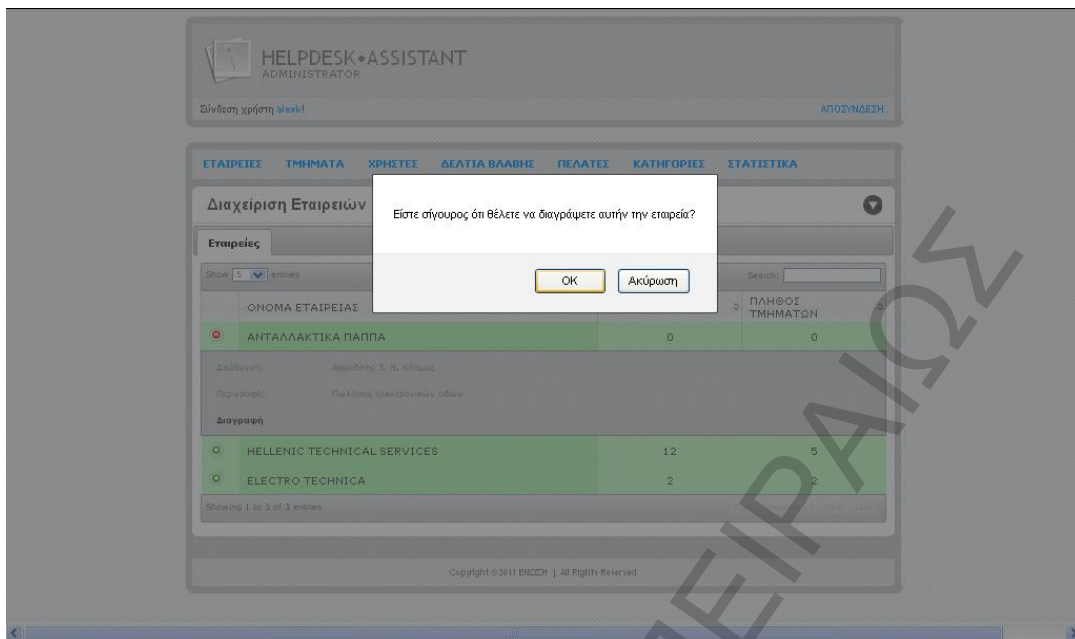
Εικόνα 15 Στοιχείο μενού «ΕΤΑΙΡΕΙΕΣ»



Εικόνα 16 Δημιουργία Προφίλ Νέας Εταιρείας.



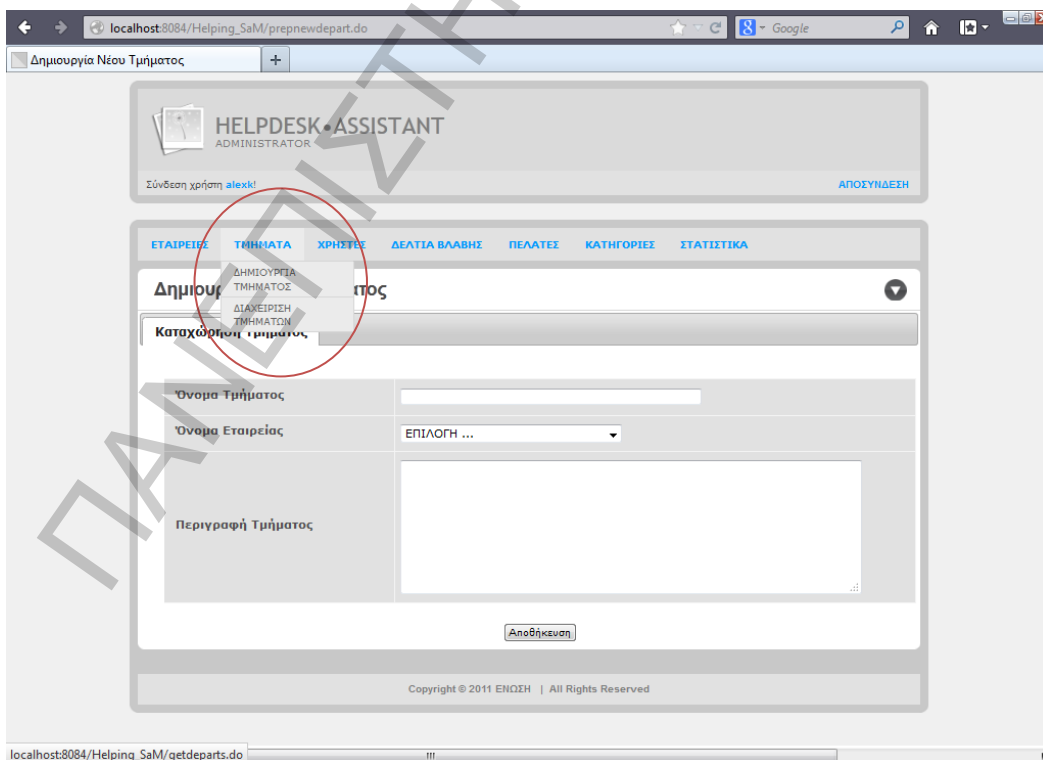
Εικόνα 17 Διαχείριση ήδη Εγγεγραμμένων Εταιρειών



Εικόνα 18 Μήνυμα επιβεβαίωσης διαγραφής εταιρείας (κάλυψη απαίτησης ευχρηστίας)

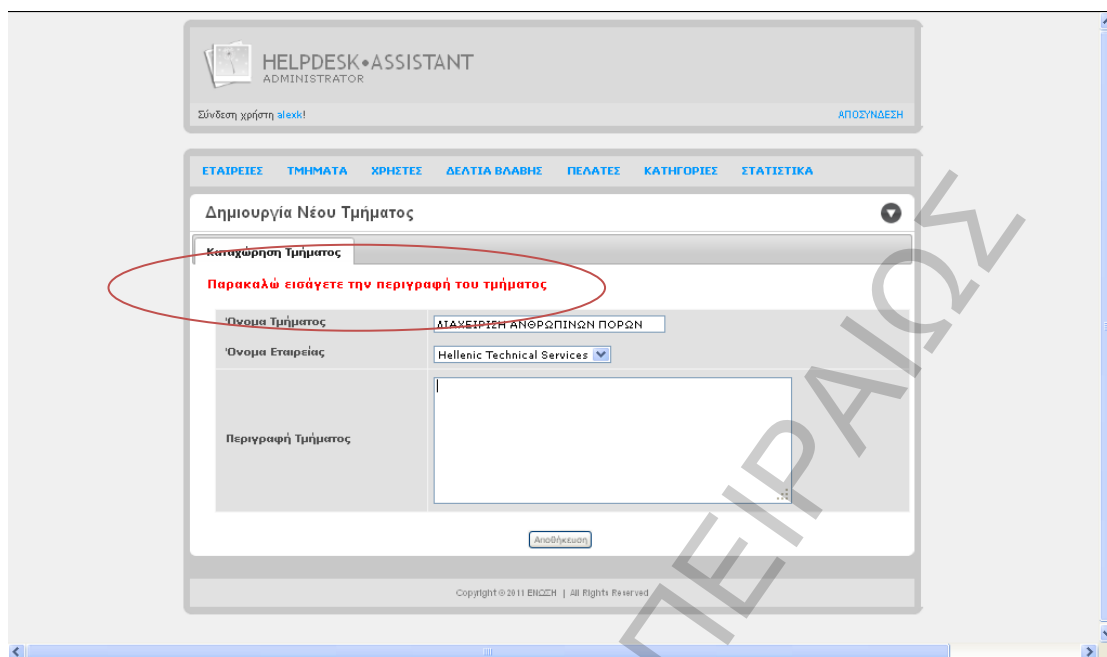
3.2.2 Στοιχείο Μενού: «ΤΜΗΜΑΤΑ»

Στην εικόνα 19 που ακολουθεί εμφανίζεται το μενού «ΤΜΗΜΑΤΑ» και η φόρμα δημιουργίας τμήματος όπου ζητούνται τα πεδία όνομα τμήματος, όνομα εταιρείας και περιγραφή τμήματος. Καλύπτοντας την απαίτηση για Ευχρηστία, στη περίπτωση που δε συμπληρωθεί κάποιο πεδίο εμφανίζεται μήνυμα λάθους (εικόνα 20) ενώ στην εικόνα 21 εμφανίζεται η επιτυχής καταχώρηση. Η τακτική αυτή ακολουθείται σε όλες τις φόρμες της εφαρμογής.

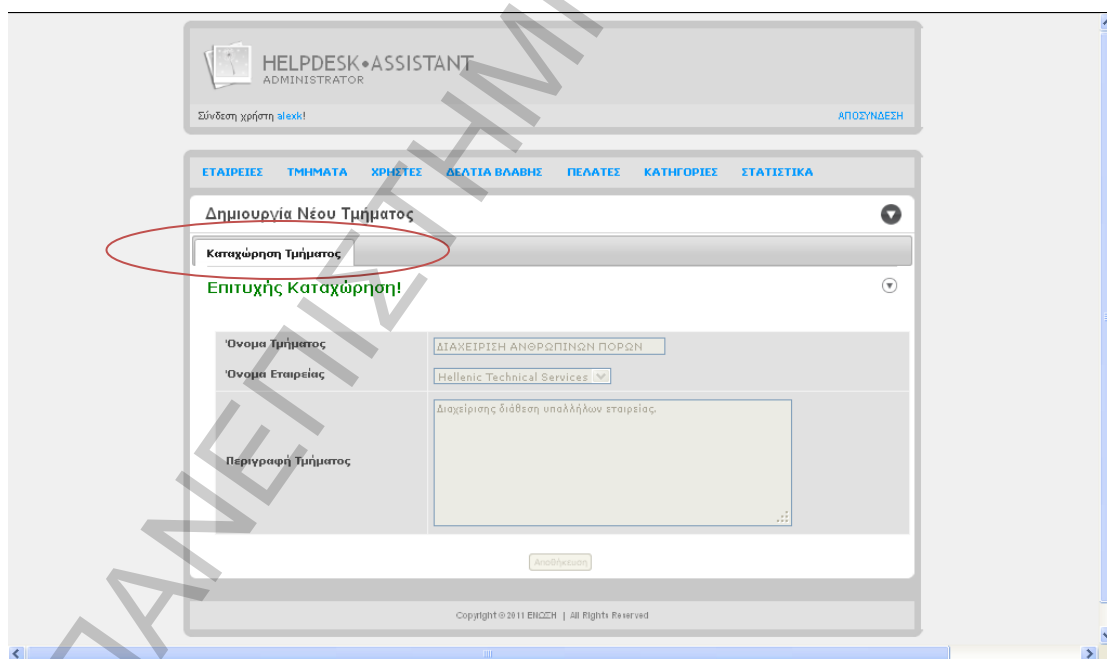


Εικόνα 19 Στοιχείο μενού «ΤΜΗΜΑΤΑ»

Εφαρμογή για Λειτουργίες Τμημάτων Αρωγής (Πολλαπλό σύστημα Helpdesk)



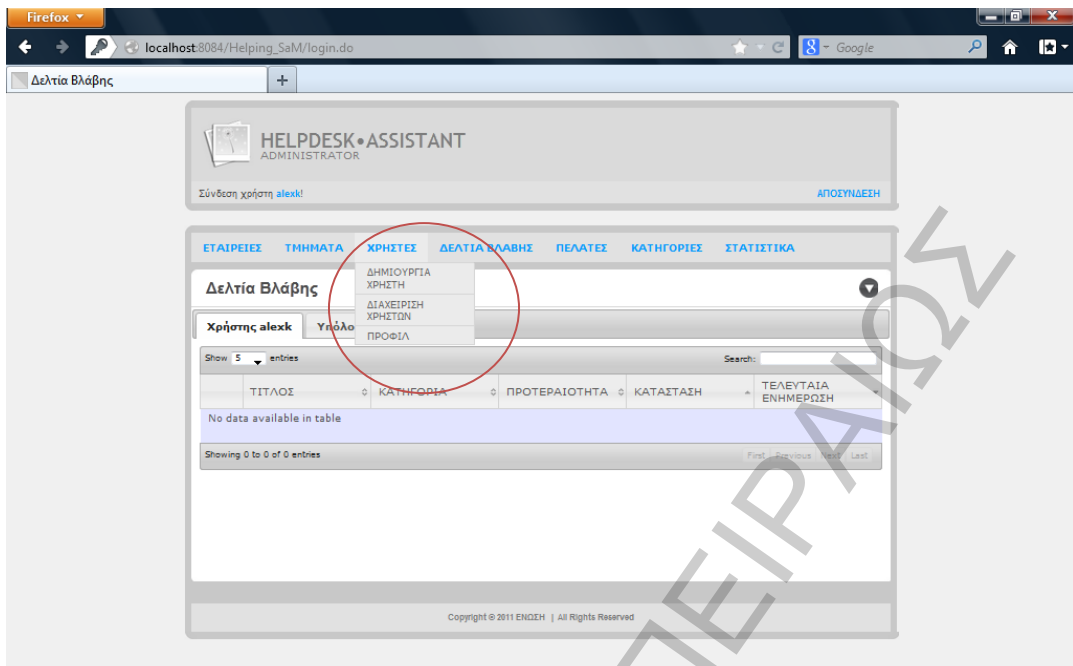
Εικόνα 20 Καταχώρηση νέου τμήματος - Μήνυμα προτροπής για συμπλήρωση υποχρεωτικού πεδίου (κάλυψη απαίτησης ευχρηστίας)



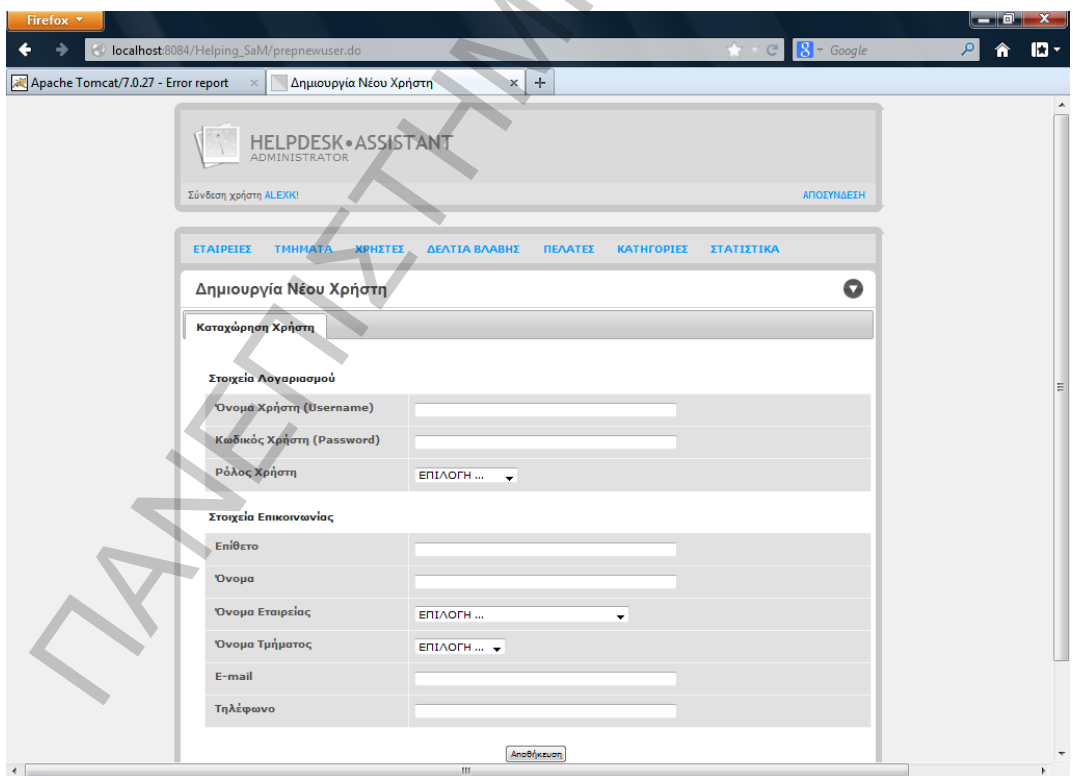
Εικόνα 21 Μήνυμα επιβεβαίωσης καταχώρησης τμήματος (κάλυψη απαίτησης ευχρηστίας)

3.2.3 Στοιχείο Μενού: «ΧΡΗΣΤΕΣ»

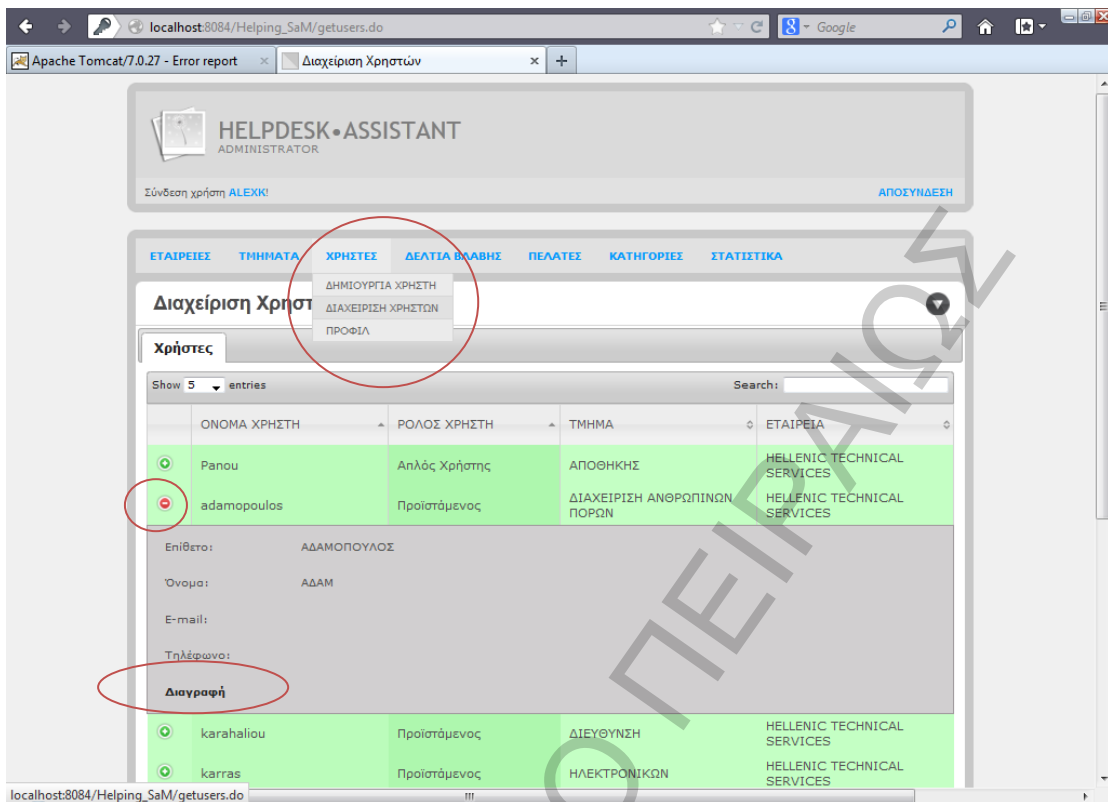
Στην εικόνα που ακολουθεί εμφανίζεται το μενού «ΧΡΗΣΤΕΣ» (εικόνα 22). Ακολουθούν οι εικόνες της φόρμας καταχώρησης νέου χρήστη, της ανάπτυξης εγγραφής χρήστη, της ενημέρωσης λογαριασμού και η εικόνα του υπομενού προφίλ χρήστη και ενδιαφέρουσα την επιλογή για αλλαγή κωδικού εισόδου.



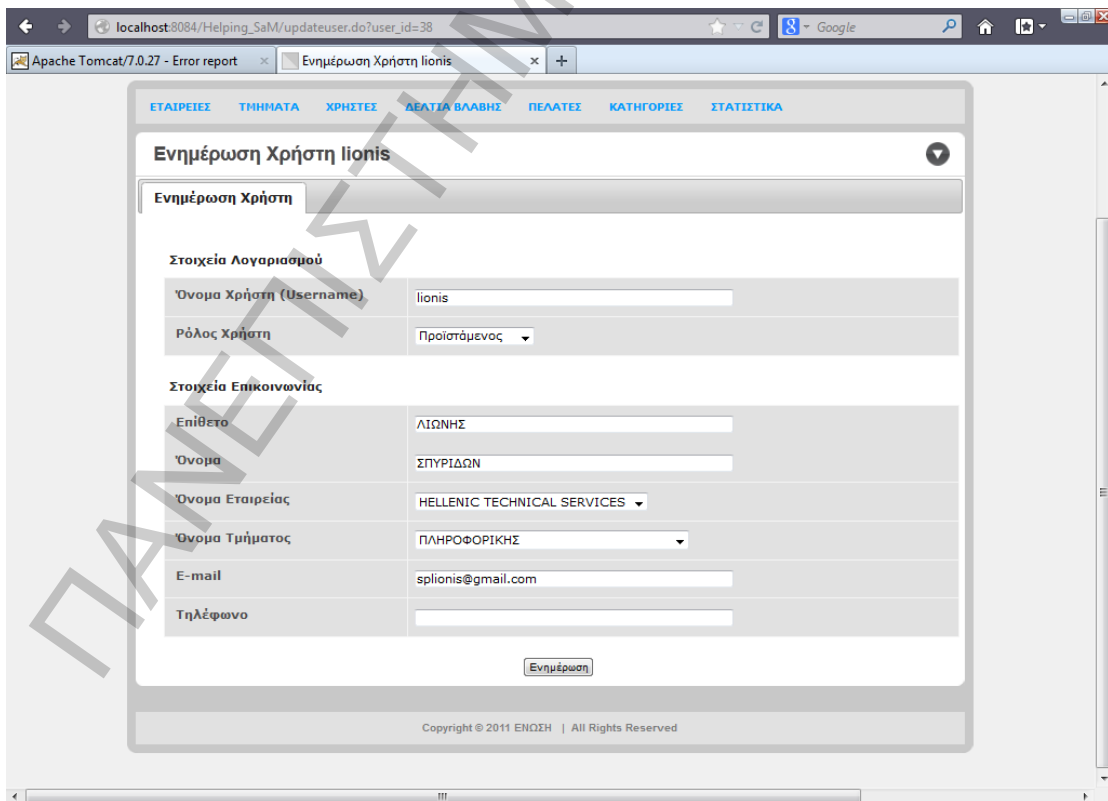
Εικόνα 22 Στοιχεία μενού «ΧΡΗΣΤΕΣ»



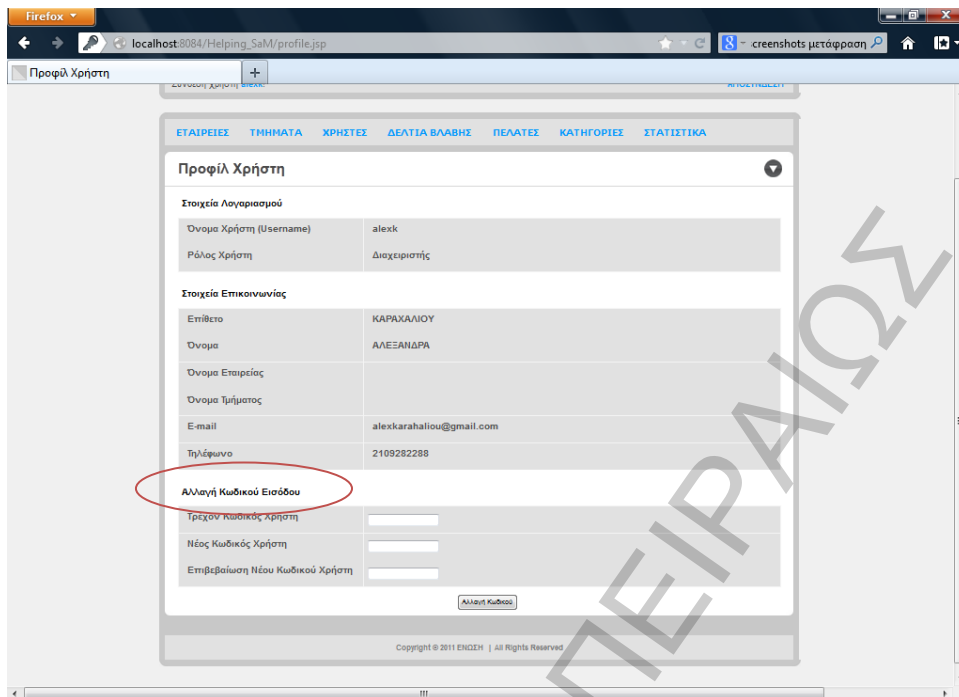
Εικόνα 23 Δημιουργία νέου χρήστη



Εικόνα 24 Ανάπτυξη προφίλ χρήστη



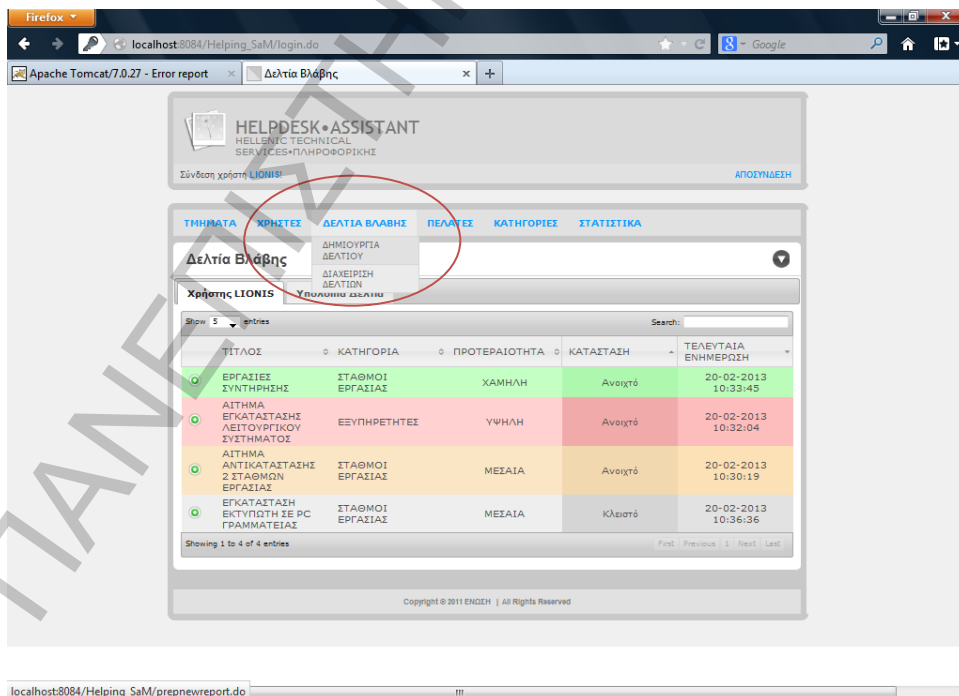
Εικόνα 25 Ενημέρωση προφίλ χρήστη



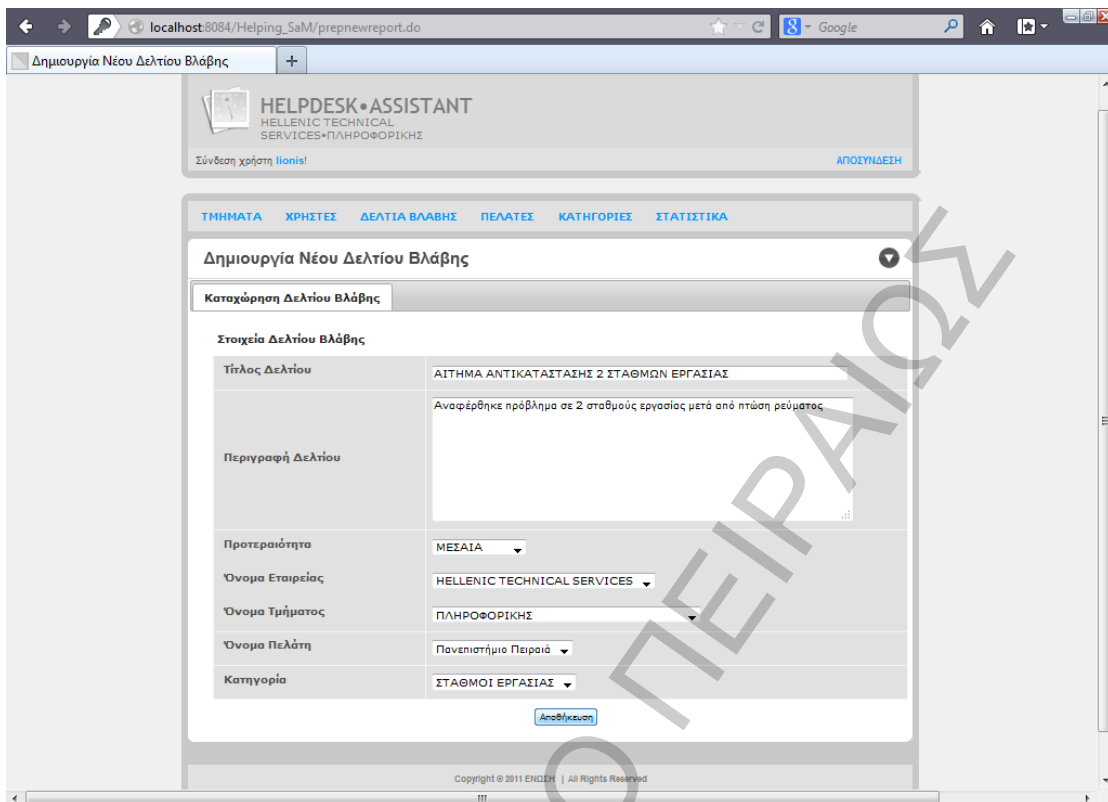
Εικόνα 26 Υπομενού Προφίλ Χρήστη - Αλλαγή Κωδικού Χρήστη.

3.2.4 Στοιχείο Μενού: «ΔΕΛΤΙΑ ΒΛΑΒΗΣ»

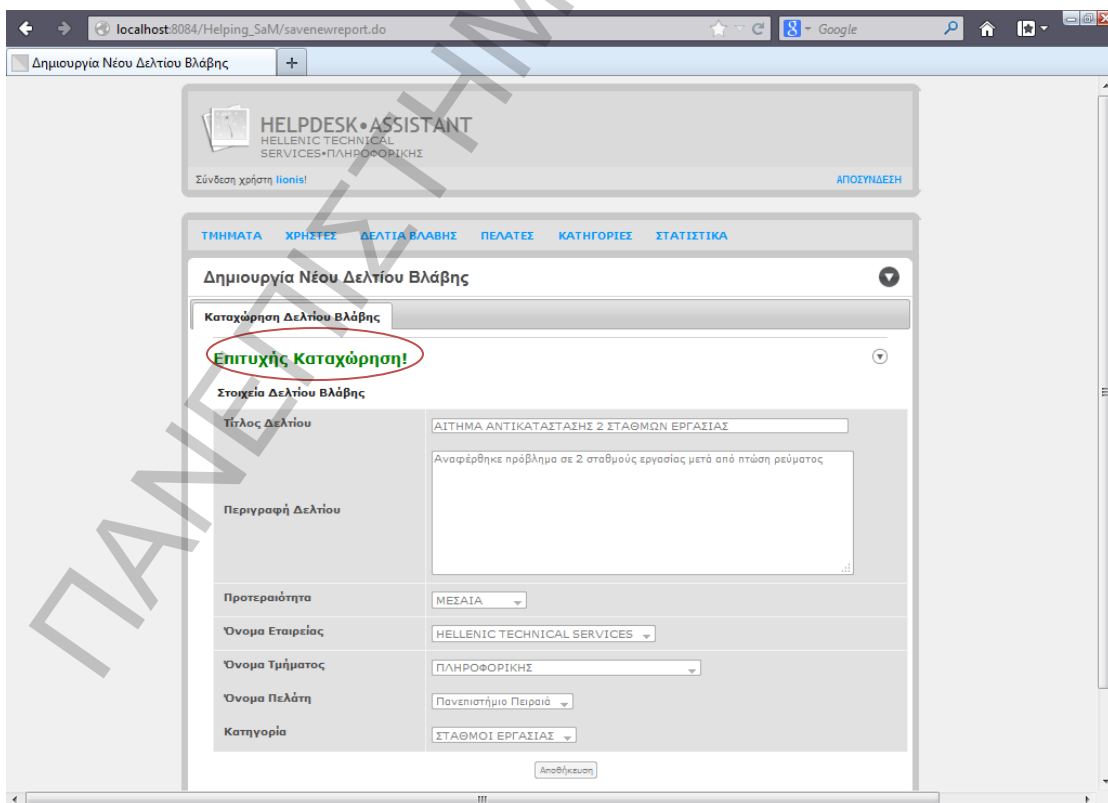
Στο μενού «ΔΕΛΤΙΑ ΒΛΑΒΗΣ» – «ΔΗΜΙΟΥΡΓΙΑ ΔΕΛΤΙΟΥ» εμφανίζεται φόρμα καταγραφής με υποχρεωτικά προς συμπλήρωση όλα τα πεδία (Εικόνα 27, 28).



Εικόνα 27 Στοιχείο μενού «ΔΕΛΤΙΑ ΒΛΑΒΗΣ»



Εικόνα 28 Φόρμα Συμπλήρωσης για Καταχώρηση Νέου Δελτίου Βλάβης.



Εικόνα 29 Μήνυμα επιτυχούς καταχώρησης (κάλυψη απαίτησης ευχρηστίας)

Στην επιλογή «ΔΙΑΧΕΙΡΙΣΗ ΔΕΛΤΙΩΝ ΒΛΑΒΗΣ», κάνοντας διπλό κλικ στην εγγραφή ενός δελτίου εμφανίζονται οι λεπτομέρειες του δελτίου με τρία πεδία επιλογών:

Ενημέρωση στοιχείων δελτίου, όπου μπορεί να γίνει τροποποίηση κάποιων βασικών πεδίων της εγγραφής του δελτίου

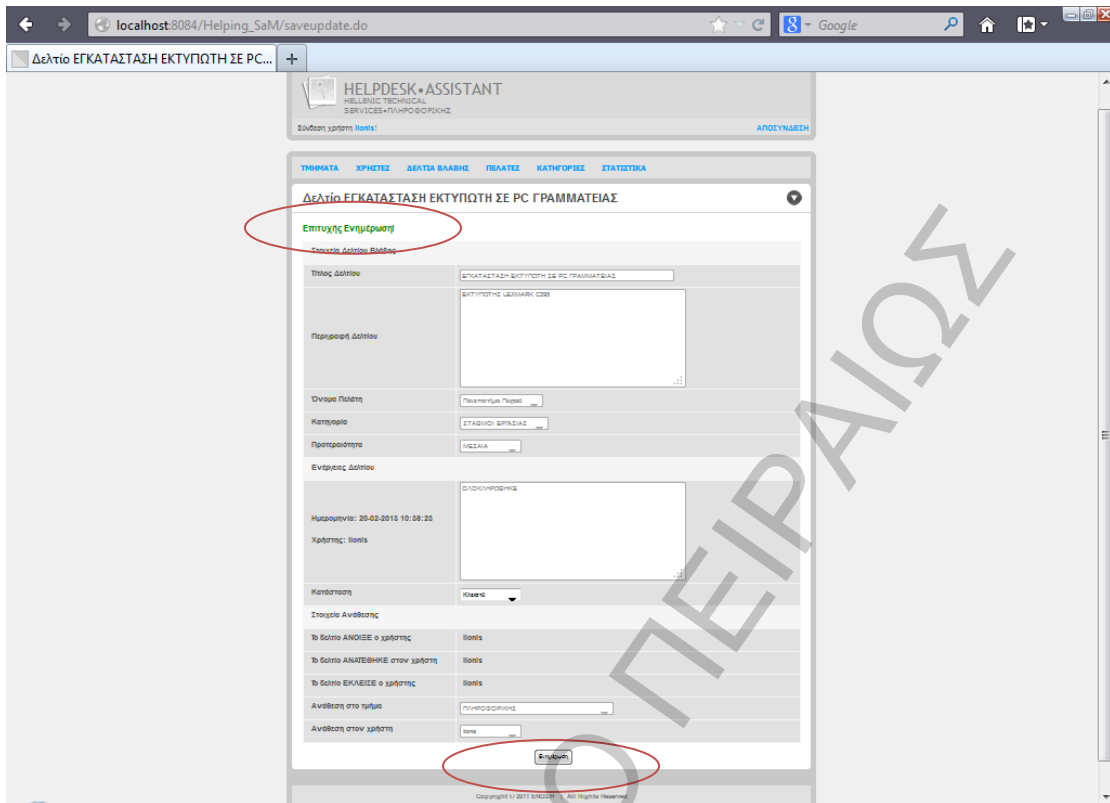
Ενημέρωση κατάστασης δελτίων, όπου γίνεται η ενημέρωση των ενεργειών που έγιναν για τη βλάβη αλλά και από όπου το δελτίο κλείνει.

Ανάθεση δελτίου, όπου μπορεί να αλλάξει ο υπεύθυνος τεχνικός ή το υπεύθυνο τμήμα (διαφορετικός προϊστάμενος) για το εκάστοτε δελτίο. (Εικόνα 30)

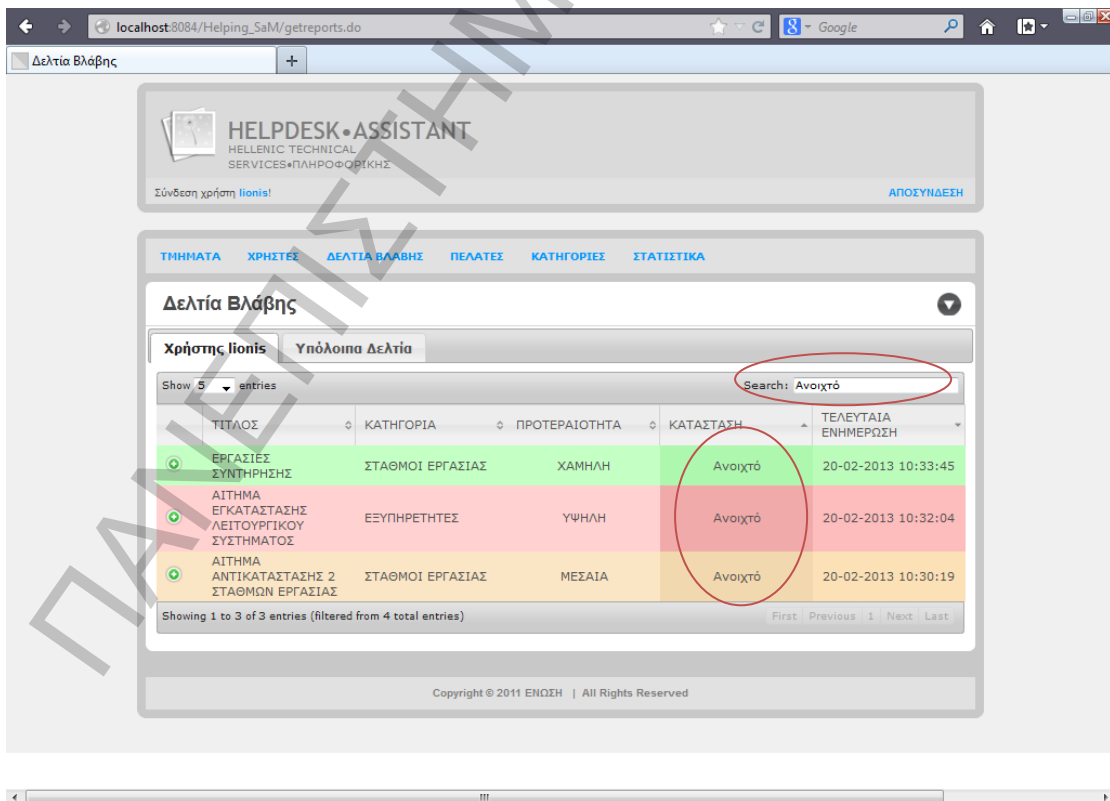
Στην εικόνα 31 εμφανίζεται η επιτυχής ενημέρωση κλεισίματος δελτίου, ενώ στην εικόνα 32 επισημαίνεται η επιλογή της αναζήτησης, που λειτουργεί ως φίλτρο για τα δελτία και είναι ένα σημαντικό χαρακτηριστικό για την εφαρμογή μας, όπως θα διαπιστώσετε στο μέρος Β.

The screenshot shows a web browser window with the URL `localhost:8084/Helping_SaM/updatesreport.do?report_id=4`. The page title is "Δελτίο ΕΓΚΑΤΑΣΤΑΣΗ ΕΚΤΥΠΩΤΗ ΣΕ PC ΓΡΑΜΜΑΤΕΙΑΣ". The interface includes a navigation menu with options like "ΤΜΗΜΑΤΑ", "ΧΡΗΣΤΕΣ", "ΔΕΛΤΙΑ ΒΛΑΒΗΣ", "ΠΕΛΑΤΕΣ", "ΚΑΤΗΓΟΡΙΕΣ", and "ΣΤΑΤΙΣΤΙΚΑ". The main content area is a form for editing a report. The form has several sections: "Στοιχεία Δελτίου Βλάβης" (with a red circle around the "Επιλογή Δελτίου Βλάβης" field), "Όνομα Πελάτη", "Κατηγορία", "Προτεραιότητα", "Ενεργειές Δελτίου" (with a red circle around the dropdown menu), "Εισάγετε Ενέργεια", "Κατάσταση" (with a red circle around the dropdown menu), and "Στοιχεία Ανάθεσης" (with a red circle around the "Ανάθεση στον χρήστη" field). The form also includes a "Συδραση χρήστη" field and a "ΒΑΝΕΛΑ" button. The background of the image contains a large, semi-transparent watermark that reads "ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑΣ".

Εικόνα 30 Φόρμα ενημέρωσης δελτίου



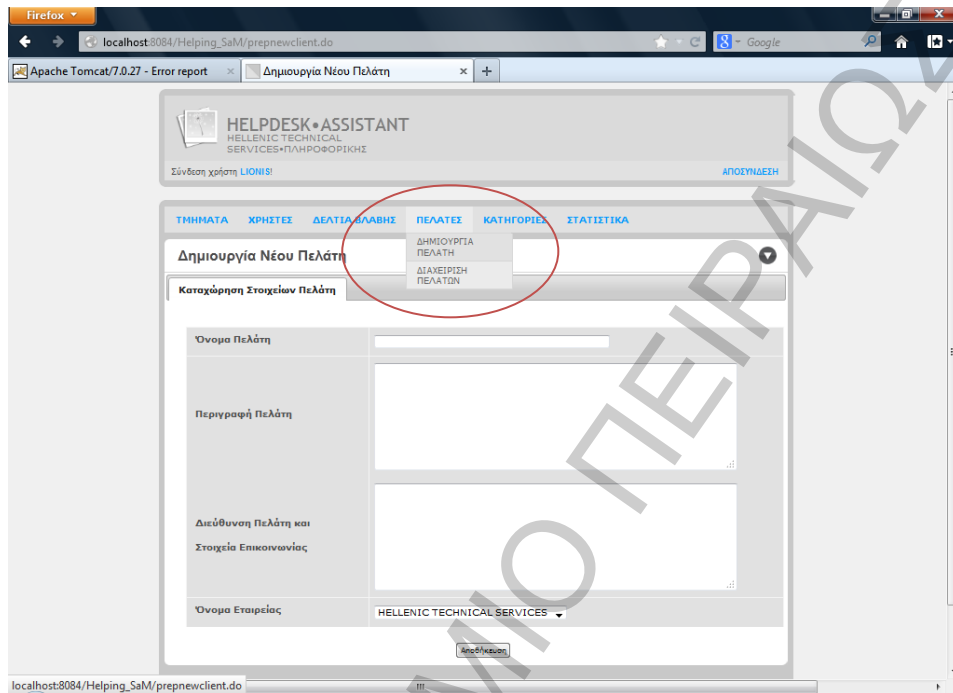
Εικόνα 31 Κλείσιμο δελτίου



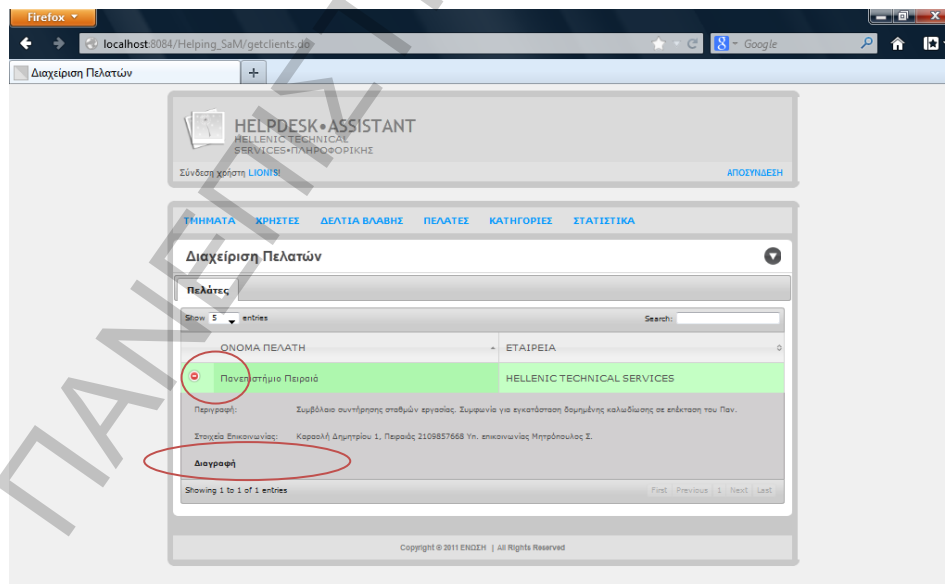
Εικόνα 32 Χρήση φίλτρου

3.2.5 Στοιχείο Μενού: «ΠΕΛΑΤΕΣ»

Οι εικόνες που ακολουθούν παρουσιάζουν το μενού επιλογών «ΠΕΛΑΤΕΣ», τη φόρμα «ΔΗΜΙΟΥΡΓΙΑ ΠΕΛΑΤΗ» και τη «ΔΙΑΧΕΙΡΙΣΗ ΠΕΛΑΤΗ». Μπορείτε και εδώ να διαπιστώσετε την ομοιομορφία που υπάρχει με τα προηγούμενα μενού (π.χ. «ΤΜΗΜΑΤΑ» κ.τ.λ. Έτσι και εδώ ισχύουν τα μηνύματα λάθους, η ανάπτυξη των εγγραφών και η διαδικασία διαγραφής όπως παρουσιάστηκαν παραπάνω.



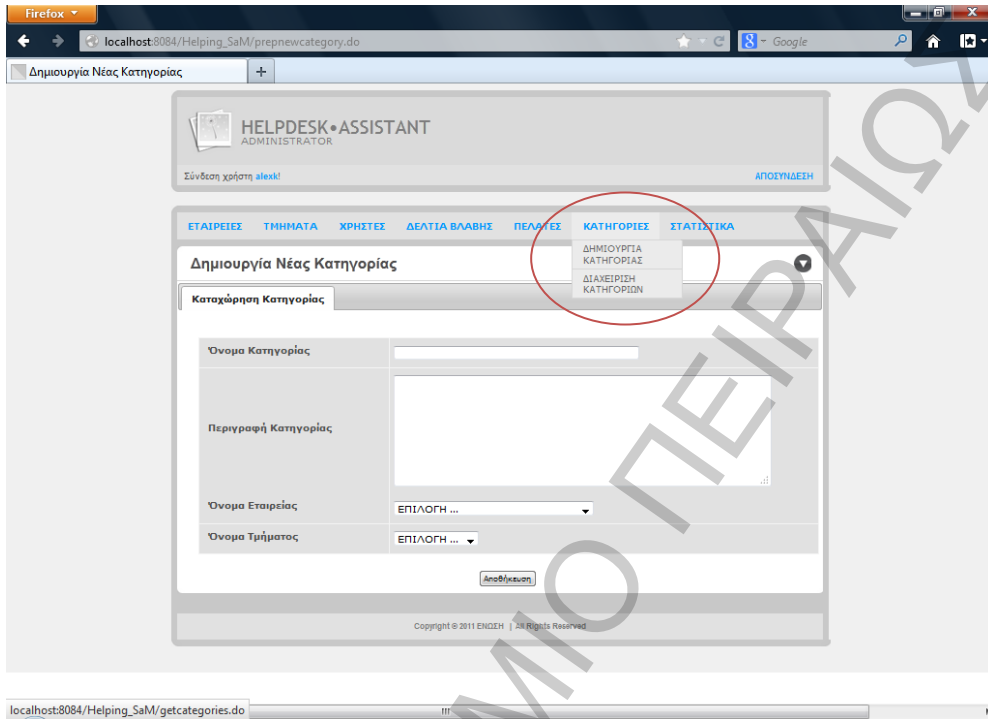
Εικόνα 33 Στοιχείο μενού «ΠΕΛΑΤΕΣ»



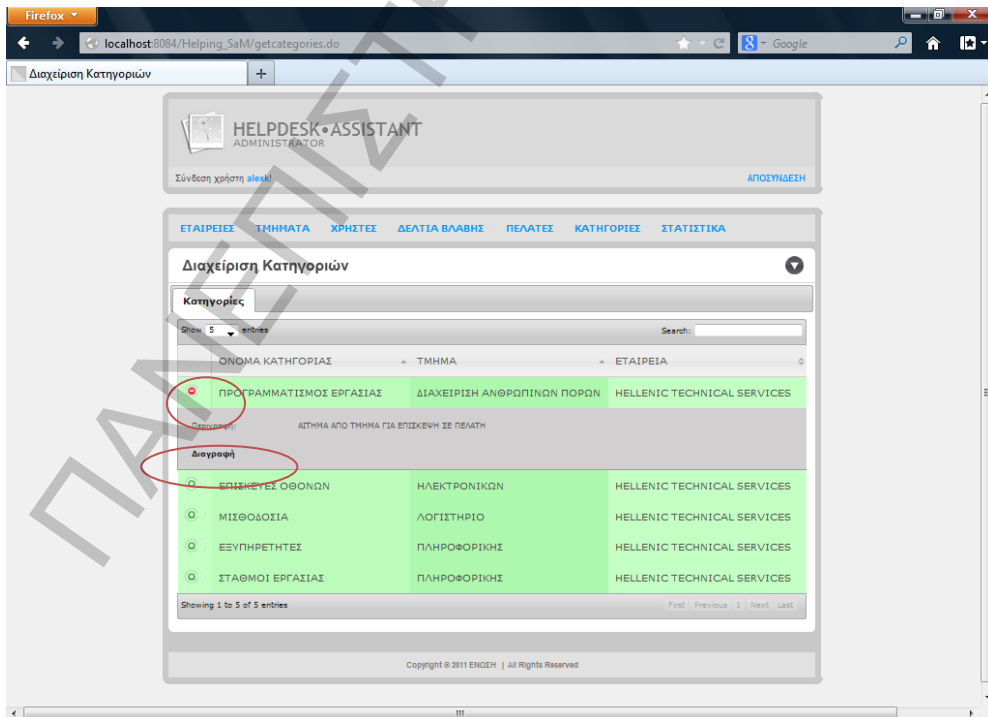
Εικόνα 34 Ανάπτυξη εγγραφής πελάτη

3.2.6 Στοιχείο Μενού: «ΚΑΤΗΓΟΡΙΕΣ»

Παρομοίως με τα παραπάνω, οι εικόνες που ακολουθούν παρουσιάζουν το μενού «ΚΑΤΗΓΟΡΙΕΣ», τη φόρμα δημιουργίας κατηγορίας και τη διαχείριση των κατηγοριών. Και εδώ διαπιστώνεται η ομοιομορφία.



Εικόνα 35 Φόρμα δημιουργία κατηγορίας



Εικόνα 36 Επιλογή διαχείρισης κατηγοριών- ανάπτυξη εγγραφής

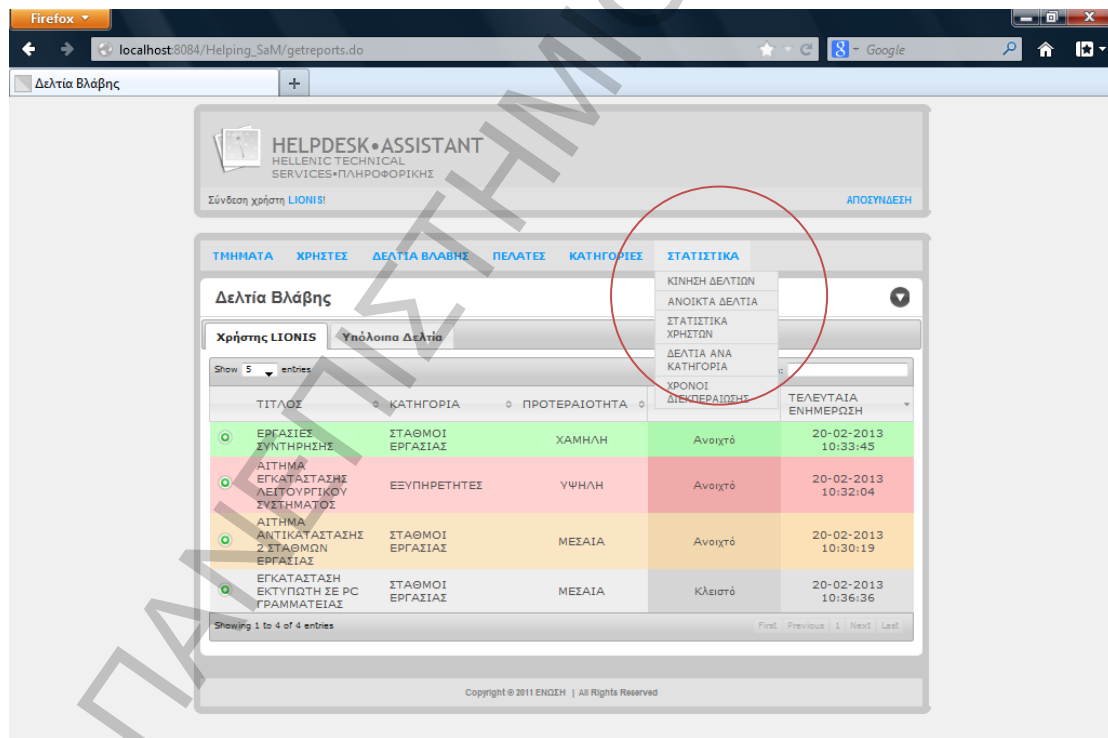
3.2.7 Στοιχείο Μενού: «ΣΤΑΤΙΣΤΙΚΑ»

Το μενού των στατιστικών είναι το μόνο που διαφέρει αρκετά προς τις επιλογές του. Ακολουθεί όμως και αυτό την ομοιομορφία που υπάρχει και στα υπόλοιπα μενού στα θέματα διαχείρισης λάθους κ.α.

Όπως φαίνεται και στην εικόνα 37, το μενού των στατιστικών έχει τις εξής επιλογές:

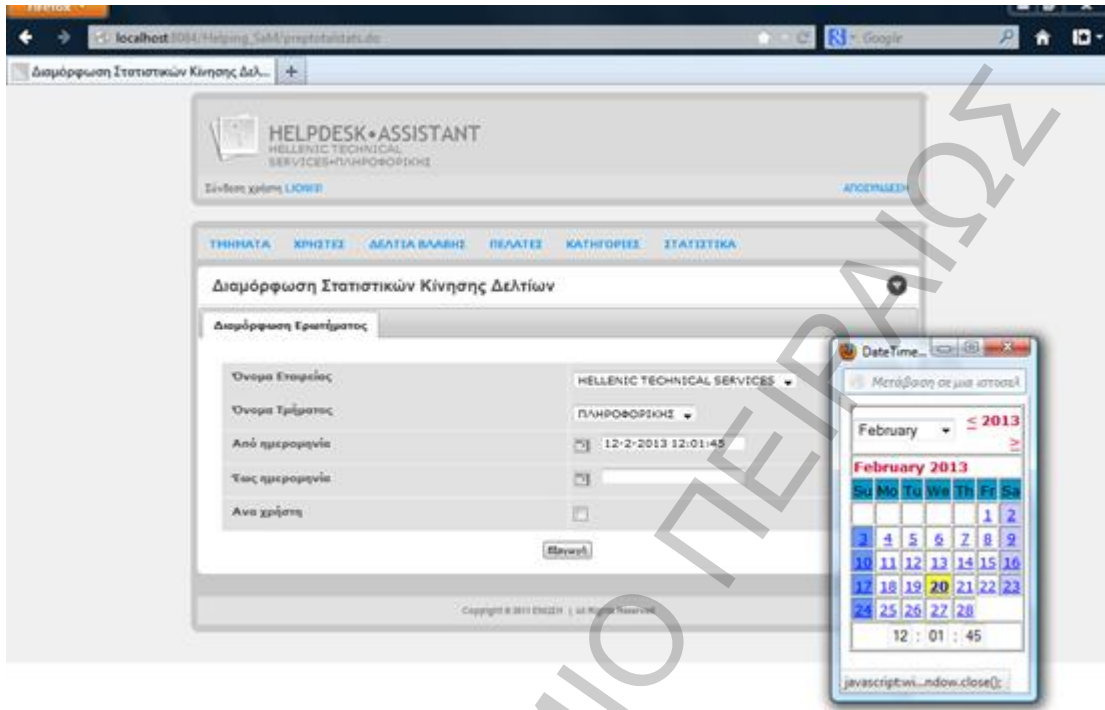
Πίνακας 3 Μενού Στατιστικά

ΣΤΑΤΙΣΤΙΚΑ	Περιγραφή
Κίνηση δελτίων	Δείχνει πόσα δελτία άνοιξαν και έκλεισαν σε ένα συγκεκριμένο χρονικό διάστημα
Ανοιχτά δελτία	Δείχνει πόσα δελτία είναι ανοιχτά για κάθε εταιρεία/ τμήμα αλλά και το μέσο χρόνο που έχουν μείνει τα δελτία ανοιχτά
Στατιστικά χρηστών	Δείχνει για κάθε χρήστη της εφαρμογής το πλήθος των δελτίων που άνοιξε, έκλεισε και ενημέρωσε (ενέργειες) σε ένα συγκεκριμένο χρονικό διάστημα
Δελτία ανά κατηγορία	Δείχνει πόσα δελτία άνοιξαν, έκλεισαν αλλά και ενημερώθηκαν σε μία εταιρεία/ τμήμα σε μία κατηγορία σε ένα συγκεκριμένο χρονικό διάστημα
Χρόνοι διεκπεραίωσης	Δείχνει το μέσο όρο, τον ελάχιστο και το μέγιστο χρόνο των δελτίων μίας εταιρείας/ τμήματος σε ένα συγκεκριμένο χρονικό διάστημα

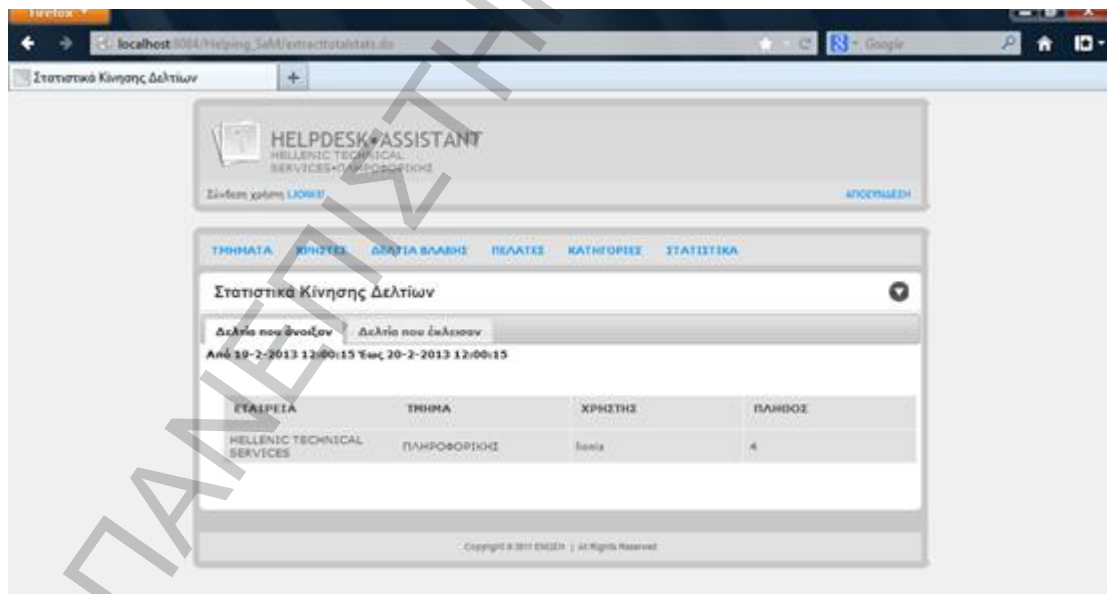


Εικόνα 37 Στοιχεία μενού «ΣΤΑΤΙΣΤΙΚΑ»

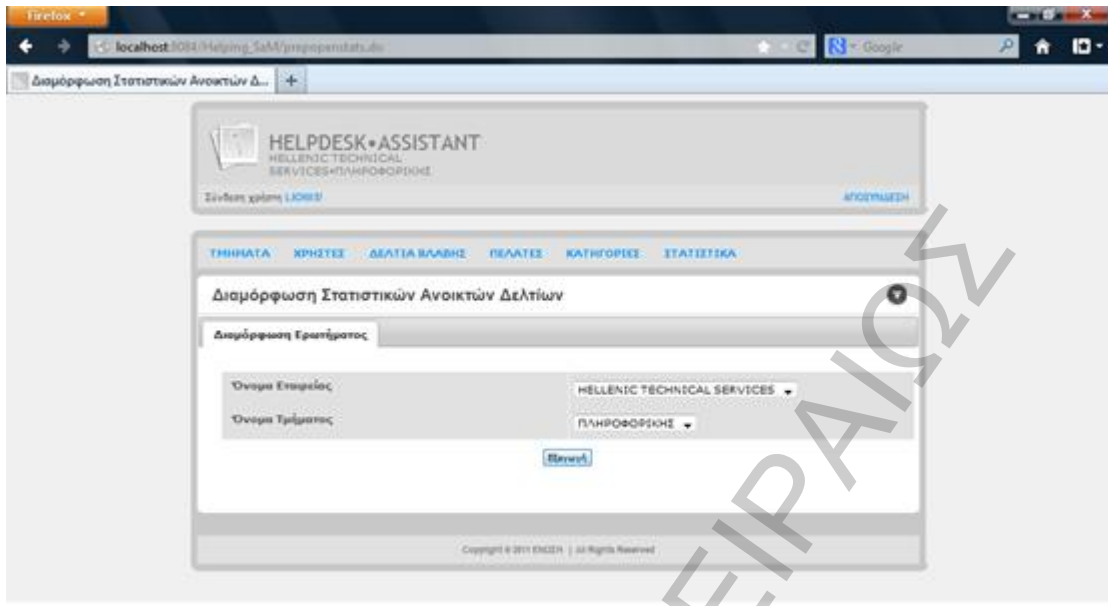
Ακολουθούν στιγμιότυπα όλων των παραπάνω επιλογών. Σε όλα τα παρακάτω θα διαπιστώσετε ομοιομορφία και στοιχεία ευχρησίας όπως για παράδειγμα τη λειτουργία του ημερολογίου για την συμπλήρωση πεδίων ημερομηνίας.



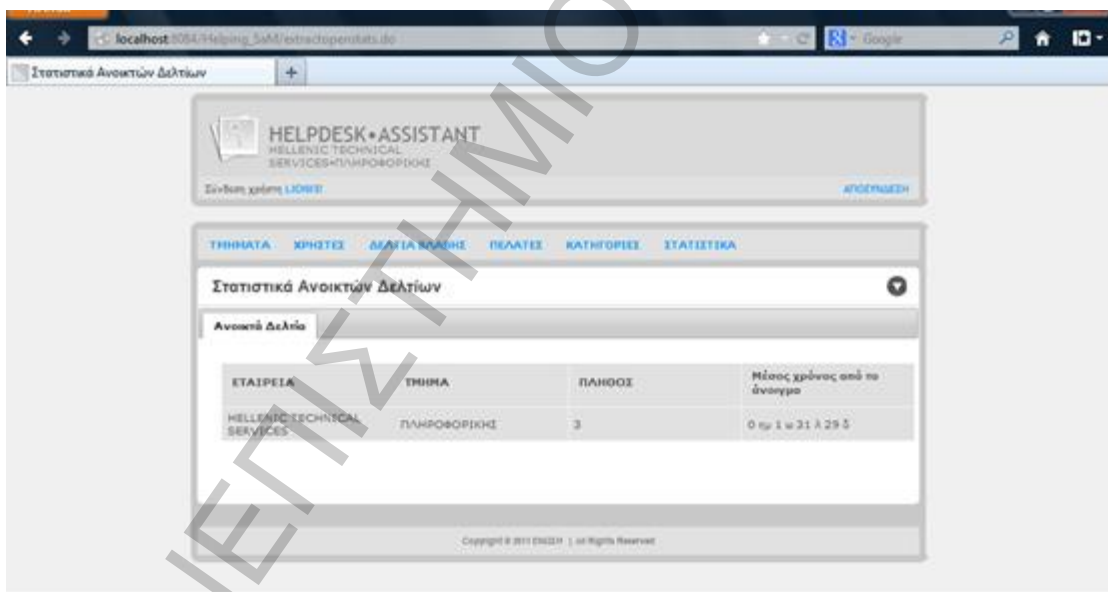
Εικόνα 38 Φόρμα επιλογής παραμέτρων στατιστικών «ΚΙΝΗΣΗ ΔΕΛΤΙΩΝ»



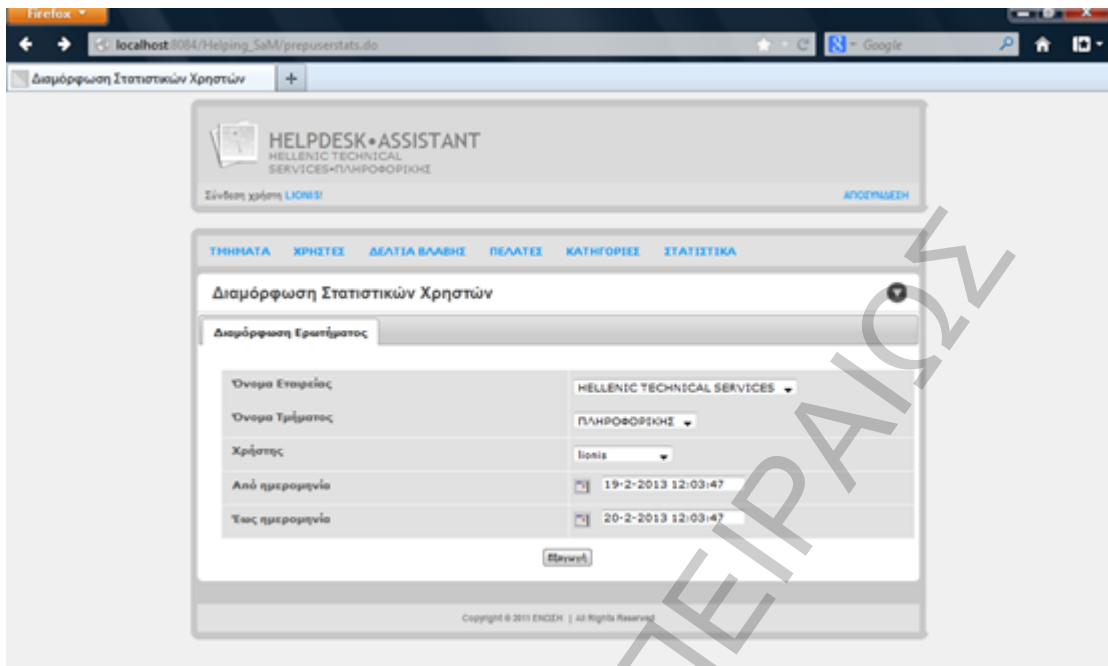
Εικόνα 39 Αποτέλεσμα στατιστικών «ΚΙΝΗΣΗ ΔΕΛΤΙΩΝ»



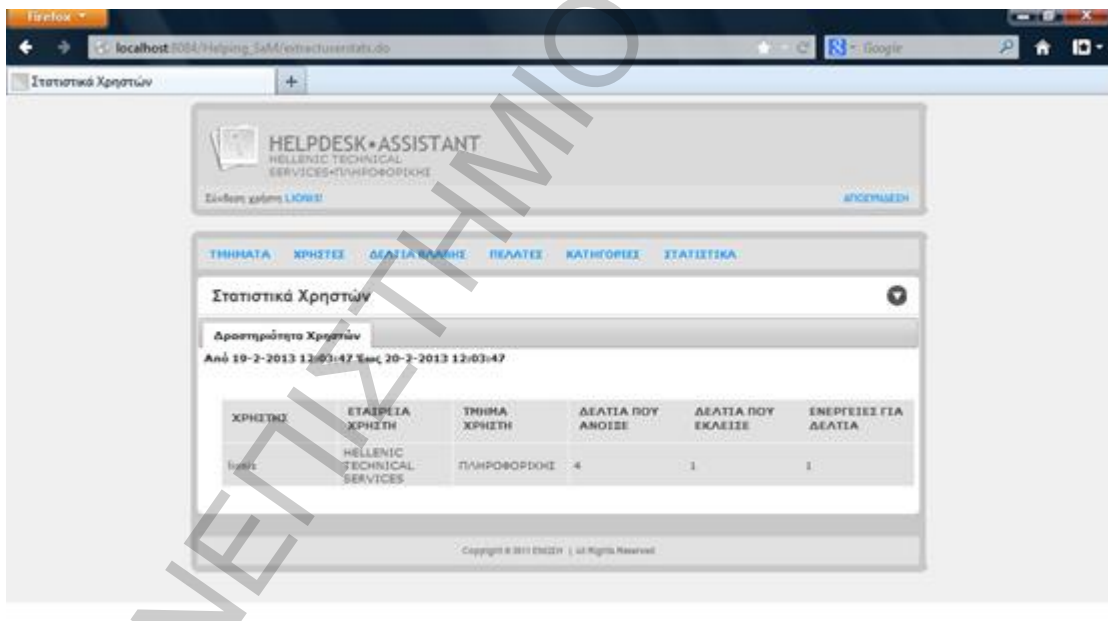
Εικόνα 40 Φόρμα επιλογής παραμέτρων στατιστικών «ΑΝΟΙΧΤΑ ΔΕΛΤΙΑ»



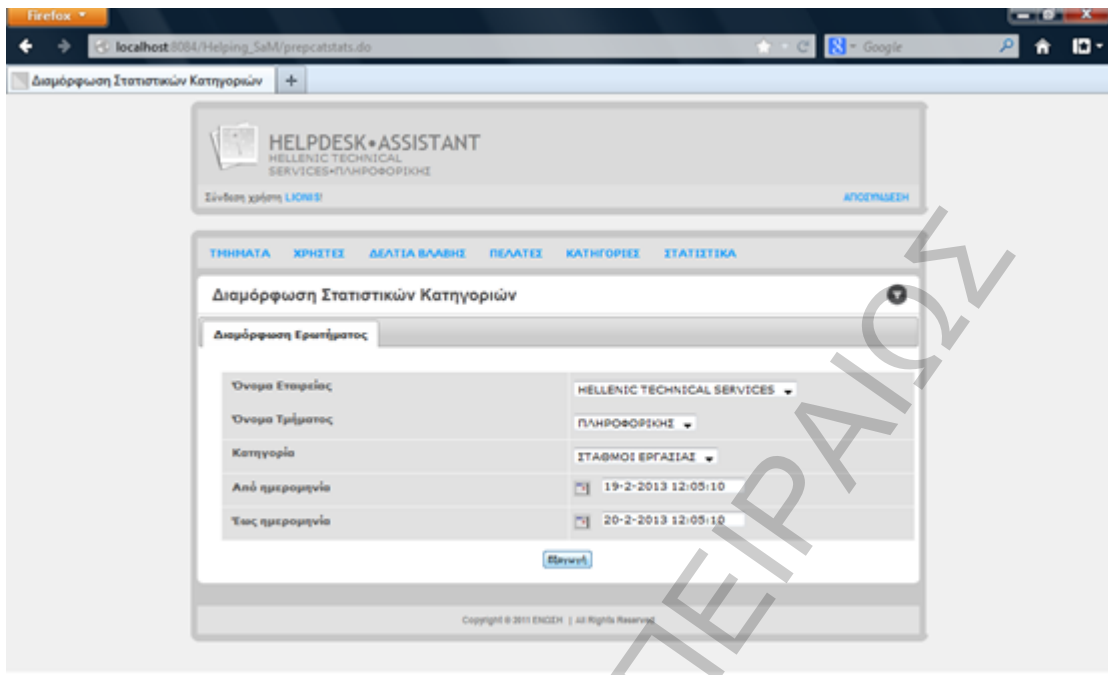
Εικόνα 41 Αποτελέσματα στατιστικών «ΑΝΟΙΧΤΑ ΔΕΛΤΙΑ»



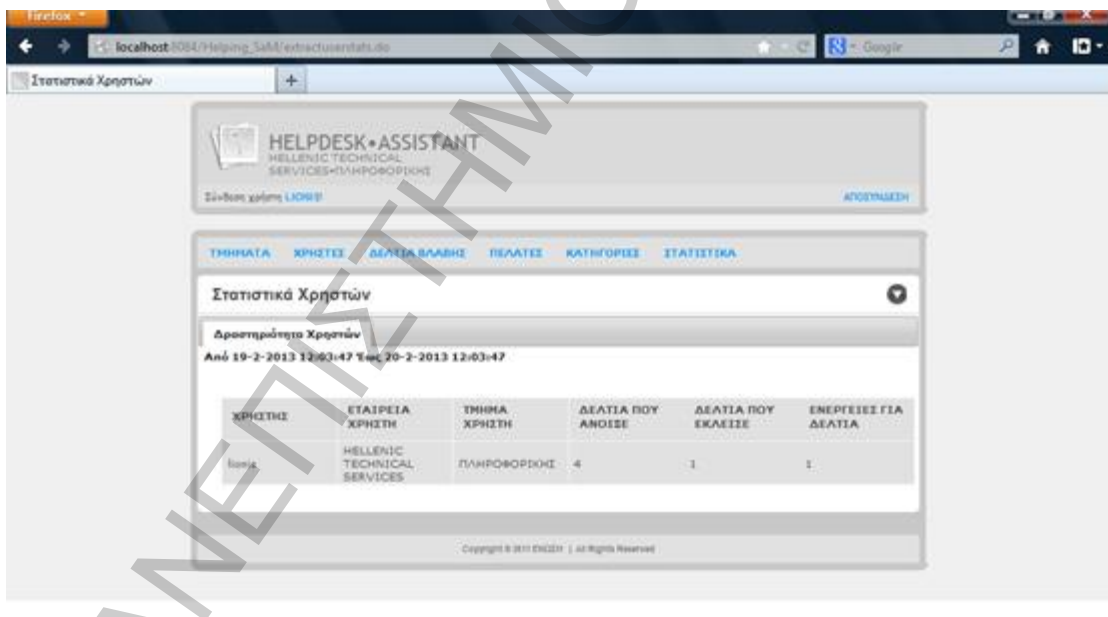
Εικόνα 42 Φόρμα επιλογής παραμέτρων στατιστικών «ΧΡΗΣΤΩΝ»



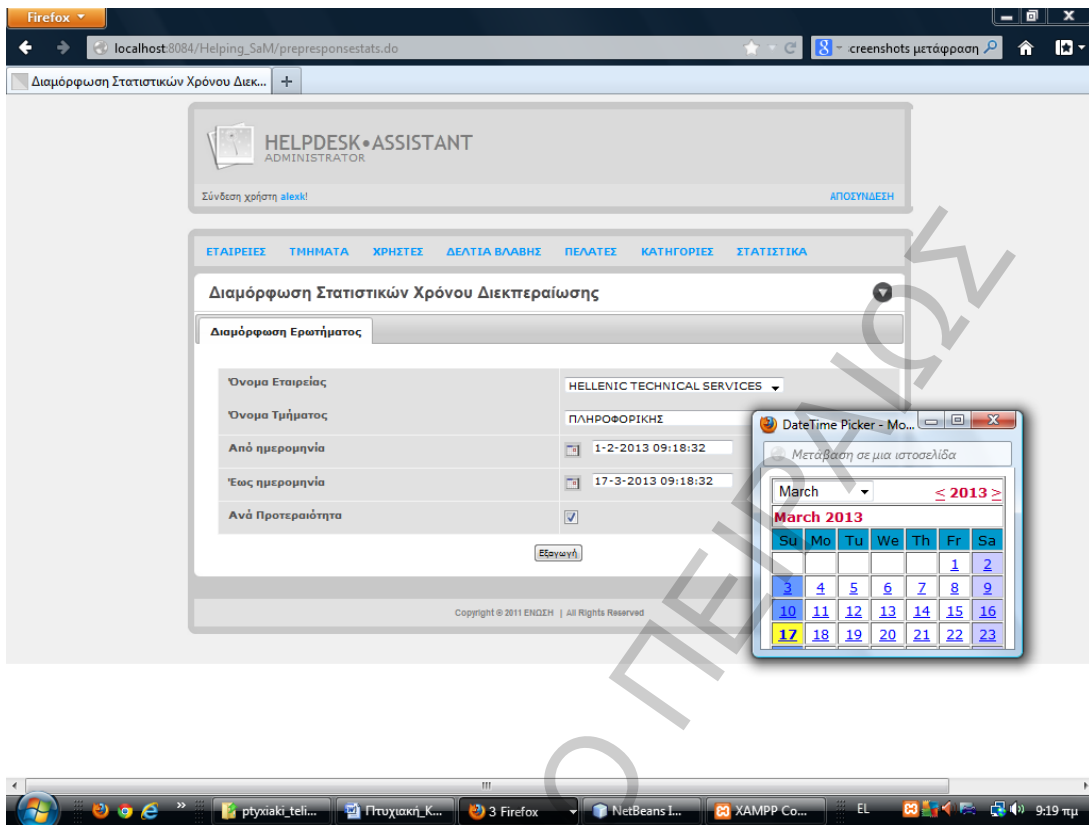
Εικόνα 43 Αποτελέσματα στατιστικών «ΧΡΗΣΤΩΝ»



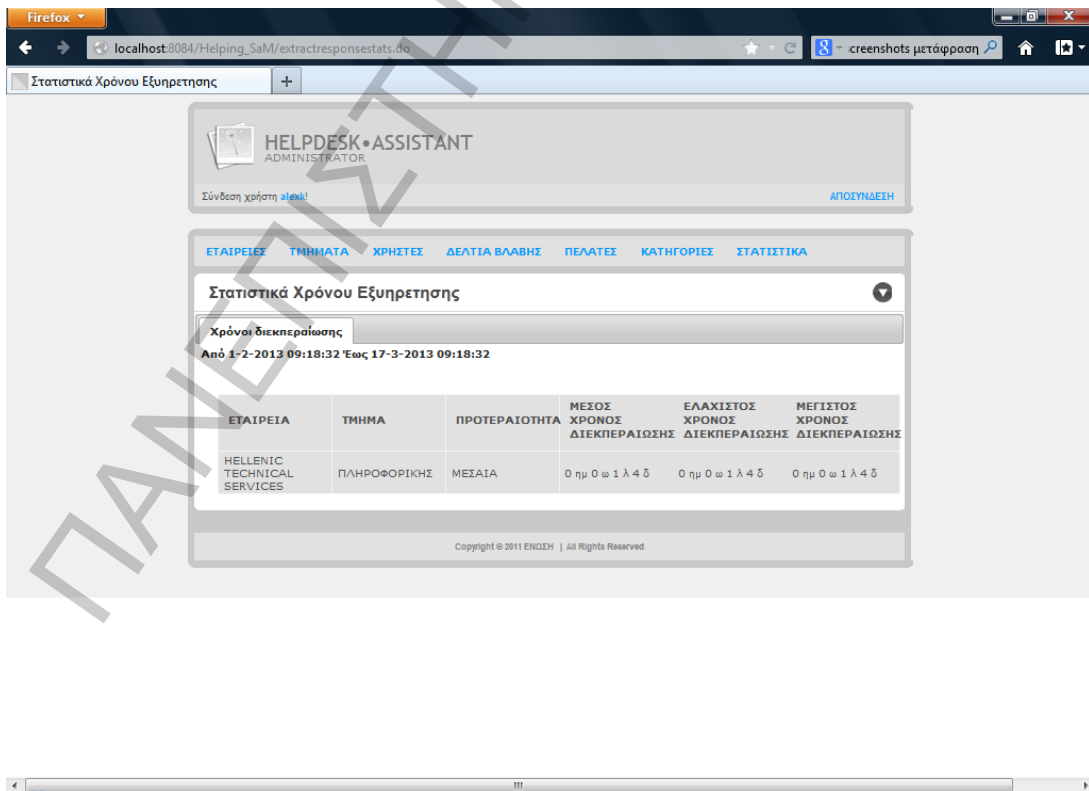
Εικόνα 44 Φόρμα επιλογής παραμέτρων στατιστικών «ΔΕΛΤΙΑ ΑΝΑ ΚΑΤΗΓΟΡΙΑ»



Εικόνα 45 Αποτελέσματα στατιστικών «ΔΕΛΤΙΑ ΑΝΑ ΚΑΤΗΓΟΡΙΑ»



Εικόνα 46 Φόρμα επιλογής παραμέτρων στατιστικών «ΧΡΟΝΟΣ ΔΙΕΚΠΕΡΑΙΩΣΗΣ»



Εικόνα 47 Αποτελέσματα στατιστικών «ΧΡΟΝΟΣ ΔΙΕΚΠΕΡΑΙΩΣΗΣ»

4 Σύνδεση με υποθετικό σενάριο – συμπεράσματα ανάλυσης/υλοποίησης

Εισαγωγή

Αφού είδαμε μέχρι τώρα το σύνολο της ανάλυσης αλλά και την ίδια την εφαρμογή, είναι ώρα να συνδέσουμε την υλοποίηση με το υποθετικό σενάριο της παραγράφου 2.1.

Μελετώντας την εφαρμογή, συμπεραίνουμε ότι καλύπτει τις ανάγκες του υποθετικού σεναρίου αλλά επίσης διακρίνουμε κάτι ακόμα πολύ σημαντικό. Το υποθετικό μοντέλο της εταιρείας της παραγράφου 2.1 είναι μόνο ένα από τα πολλά μοντέλα εταιρειών πάνω στα οποία μπορούμε να βασίσουμε την ανάλυση και το σχεδιασμό της εφαρμογής μας.

Ένα από τα κύρια πλεονεκτήματα που διαθέτει η εφαρμογή μας, το οποίο απορρέει από την αρχική ανάλυση είναι το γεγονός ότι, με την κατάλληλη παραμετροποίηση από το διαχειριστή και όχι από τον προγραμματιστή, μπορεί να χρησιμοποιηθεί από ένα σύνολο εταιρειών με διαφορετικά μοντέλα- οργανογράμματα.

Ο διαχειριστής δημιουργώντας τις κατάλληλες εγγραφές σχετικά με τις εταιρείες, τα τμήματα και τους χρήστες, μπορεί να χρησιμοποιήσει την εφαρμογή σε όποιο επίπεδο επιθυμεί.

Προσθέτοντας μόνο μία εταιρεία και τα τμήματα αυτής, η εφαρμογή λειτουργεί ως εσωτερική εφαρμογή της εταιρείας. Προσθέτοντας μία κύρια εταιρεία και από κάτω από αυτή τις συνεργαζόμενες ή και τις θυγατρικές της, έχουμε μία εφαρμογή που αυξάνει σε πολυπλοκότητα, ανεβαίνει ένα επίπεδο παραπάνω και εξυπηρετεί έναν όμιλο εταιρειών ως μία κοινή δικτυακή εφαρμογή δια μέσω της οποίας επικοινωνούν όλοι.

Επίσης, μπορούμε να δούμε την εφαρμογή ως ένα εργαλείο μίας third party εταιρείας που δίνει υπηρεσίες helpdesk σε πελάτες εταιρειών που την έχουν προσλάβει για αυτή τη δουλειά. (δηλαδή εξυπηρετεί πελάτες πελατών της).

Τα επίπεδα που μπορεί να φτάσει η εφαρμογή είναι 3, όπως φαίνεται και στο τελευταίο παράδειγμα, και έτσι δεν κινδυνεύει να μεγαλώσει υπερβολικά ώστε να υπάρχουν δυσκολίες στη λειτουργία της.

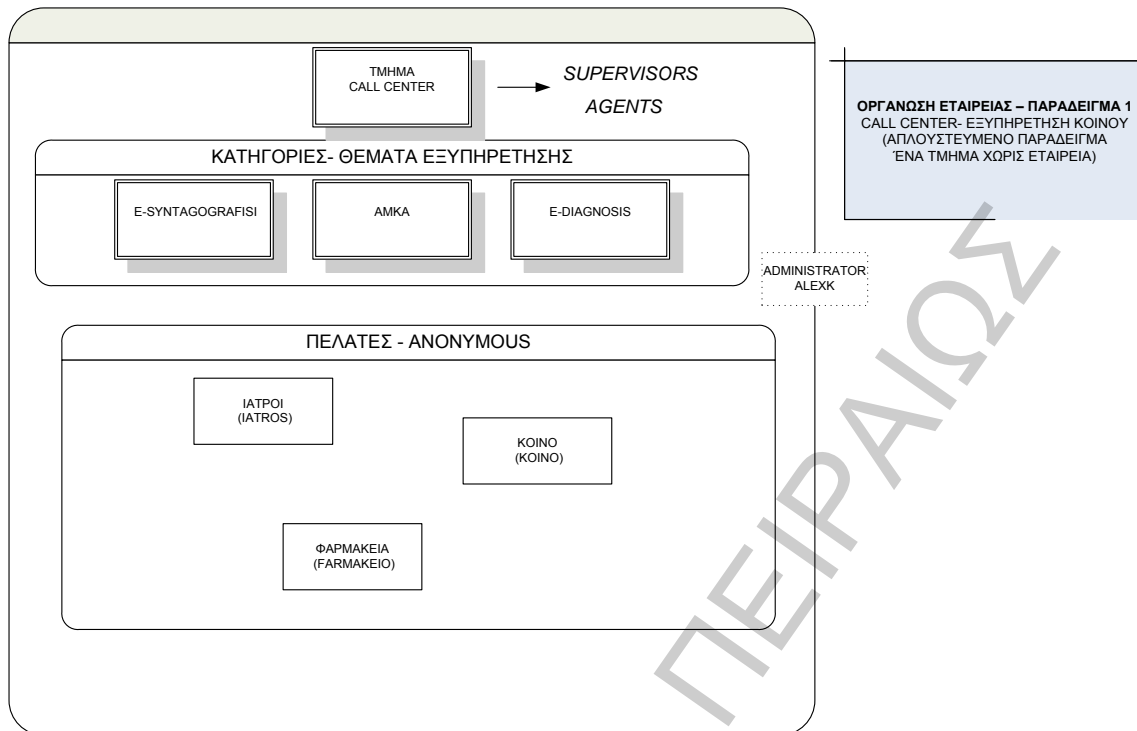
Ανάλογα το σχήμα και την πολιτική της κάθε εταιρείας, η εφαρμογή προσαρμόζεται και χρησιμοποιείται αναλόγως.

Ας δούμε σε αυτό το σημείο τέσσερα οργανογράμματα – μοντέλα εταιριών και πως μπορεί να χρησιμοποιηθεί η εφαρμογή μας σε αυτά:

4.1 Χρήση εφαρμογής από ένα τμήμα

Το απλούστερο μοντέλο εξυπηρέτησης δελτίων. Ένα τμήμα μιας εταιρείας που εξυπηρετεί αιτήματα (συνήθως κλήσεις) για διάφορες κατηγορίες θεμάτων – βλαβών, πολλές φορές χωρίς συγκεκριμένους πελάτες.

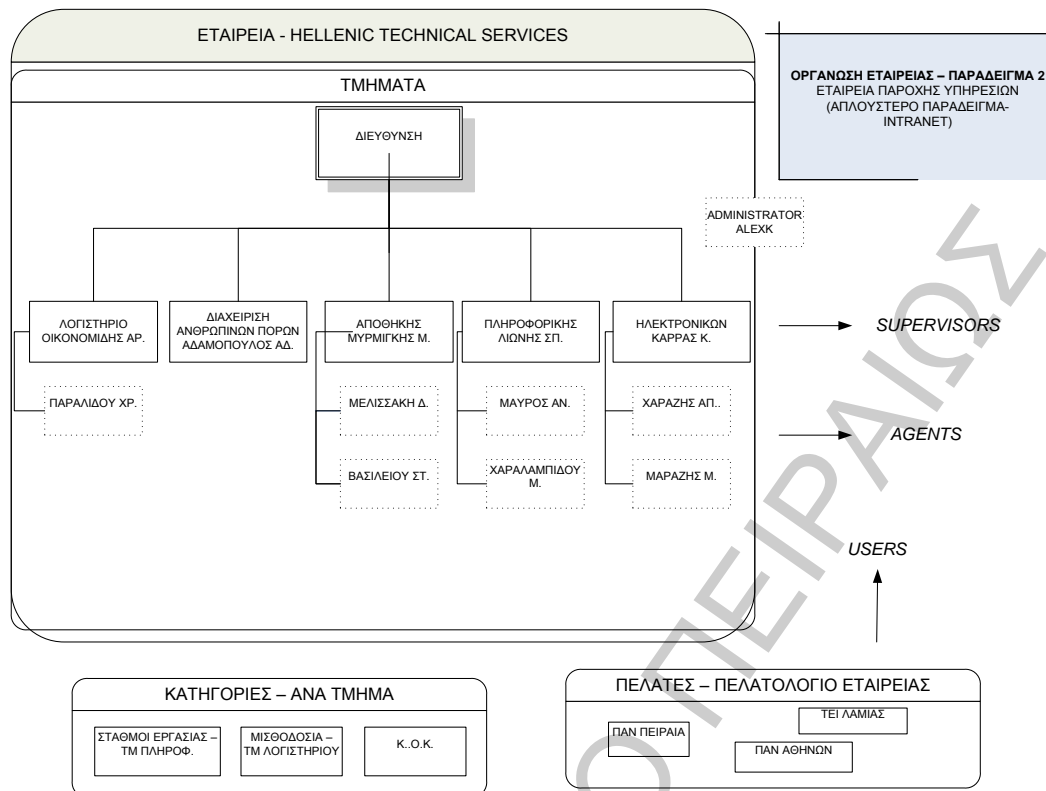
Εδώ τυπικά δηλώνεται στο όνομα της εταιρείας κάτω από την οποία δημιουργούμε το τμήμα, της κατηγορίες του και δημιουργούμε ως πελάτες έναν ανώνυμο.



Εικόνα 48 Χρήση εφαρμογής από ένα τμήμα.

4.2 Χρήση εφαρμογής στα πλαίσια του intranet μιας εταιρείας.

Εδώ η εφαρμογή χρησιμοποιείται στα δεδομένα όρια μιας εταιρείας, για επίλυση προβλημάτων μέσα στα τμήματα, αλλά και την διαχείριση αιτημάτων ανάμεσα στα τμήματα. Διαχειριστής και λειτουργός είναι η ίδια η εταιρεία (ορίζει administrator), supervisors ορίζονται οι προϊστάμενοι των τμημάτων, οι οποίοι είναι υπεύθυνοι για τη διατμηματική επικοινωνία, agents (χρήστες) ορίζονται οι υπάλληλοι της εταιρείας, ενώ μπορεί να δίνεται δικαίωμα απλού χρήστη στους πελάτες.



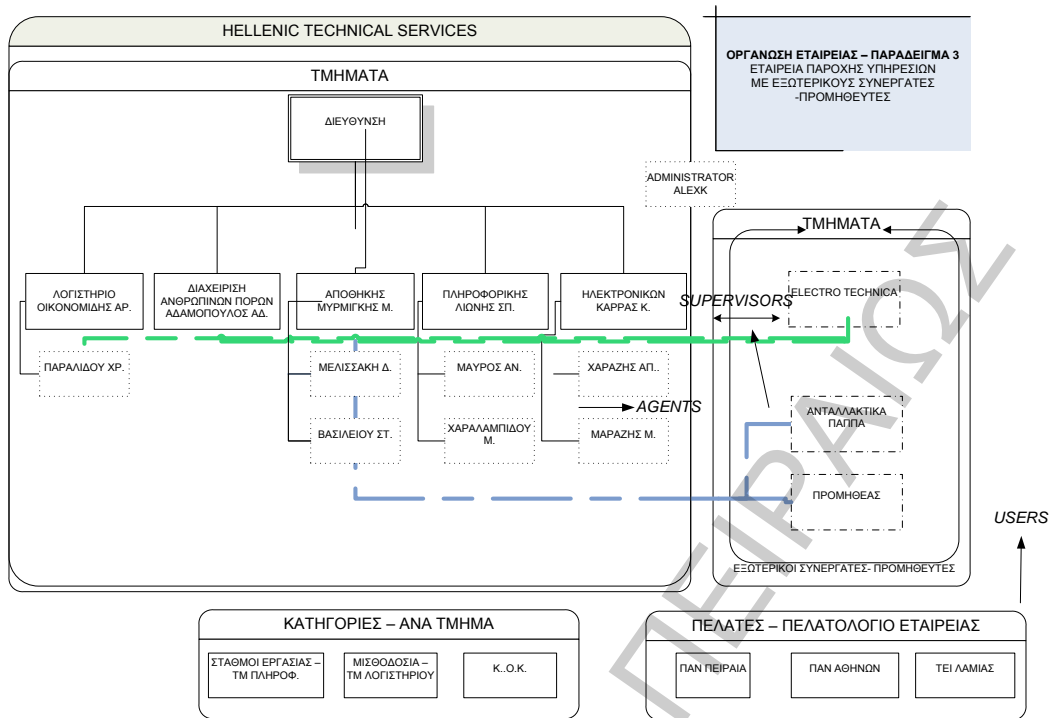
Εικόνα 49 Χρήση εφαρμογής σε intranet

4.3 Χρήση εφαρμογής σε intranet και εξωτερικούς συνεργάτες.

Η εφαρμογή εδώ χρησιμοποιείται τόσο από την ίδια την εταιρεία αλλά και από τρίτους, για όφελος της ίδιας της εταιρείας. Η περίπτωση αυτή μοιάζει με την προηγούμενη, μόνο που εδώ οι εξωτερικοί συνεργάτες, οι προμηθευτές και άλλοι, λογίζονται ως μια προέκταση των τμημάτων της εταιρείας. Δημιουργούνται για αυτούς τμήματα και λαμβάνουν λογαριασμούς διαχείρισης των τμημάτων τους ως supervisors.

Η περίπτωση αυτή είναι που καλύπτει και την περιγραφή του υποθετικού σεναρίου μας.

Η υποχρέωση χρήσης ειδικού λογισμικού σε τρίτους δεν είναι καθόλου ασυνήθιστη σε μεγάλες εταιρείες, όπου οι συνεργάτες αυτών είναι πολύ μικρότερου μεγέθους.

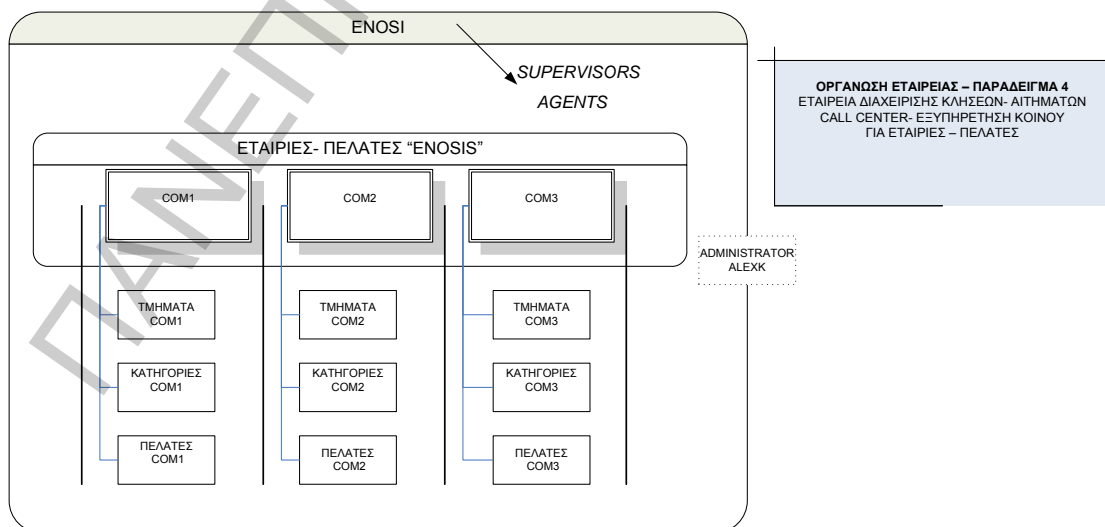


Εικόνα 50 Χρήση εφαρμογής σε intranet και από τρίτους

4.4 Χρήση “Third party”

Μία εταιρεία διαχείρισης κλήσεων- αιτημάτων κ.α. εξυπηρετεί πελάτες, πελατών της. Οι πελάτες της εταιρείας απλά δίνουν το μοντέλο οργάνωσής τους, τα προβλήματα που θέλουν να εξυπηρετούν, τις κατηγορίες, και συνήθως δεν εμπλέκονται με τη διαδικασία εξυπηρέτησης χρηστών σε κανένα επίπεδο.

Η κύρια εταιρεία που φιλοξενεί την εφαρμογή, διαθέτει το προσωπικό για την εξυπηρέτηση των πελατών, την διαχείριση και χρήση της εφαρμογής ενώ συνήθως οι πελάτες της δεν έχουν δικαιώματα ούτε απλών χρηστών.



Εικόνα 51 Χρήση εφαρμογής σε εταιρεία third party helpdesk

Εφαρμογή για Λειτουργίες Τμημάτων Αρρωγής (Πολλαπλό σύστημα Helpdesk)

ΜΕΡΟΣ Β - Αξιολόγηση . Η εφαρμογή από την πλευρά του μανάτζμεντ.

5 . Εισαγωγή Μάνατζμεντ

Στο πλαίσιο μιας μεταπτυχιακής διατριβής, δεν θα ήταν σωστό να παραλείψουμε μια αποτίμηση του έργου μας. Δεν πρέπει να λείπει από την μελέτη μας η αξιολόγηση της ίδιας της εφαρμογής ενταγμένη στα πρακτικά πλαίσια μιας εταιρείας και μία αποτίμηση του αν και κατά πόσο φέρνει πρόοδο στην εταιρεία.

Για το λόγο αυτό κληθήκαμε να εξελίξουμε το υποθετικό μοντέλο της εταιρείας της παραγράφου 2.1 και να πείσουμε για τα οφέλη της εφαρμογής σε σχέση με βασικά ερωτήματα του μανάτζμεντ.

Γιατί κάποιος να επενδύσει χρήμα και χρόνο για να εντάξει μια νέα τεχνολογία στις διαδικασίες του; Πώς θα πείσουμε μετόχους και εργαζόμενους μιας επιχείρησης για την αξία και τη χρησιμότητα της εφαρμογής; Πώς μπορεί η εφαρμογή να βοηθήσει τη διοίκηση στη λήψη αποφάσεων και στη χάραξη στρατηγικής της εταιρείας;

Για να απαντήσουμε αυτά και άλλα τέτοια ερωτήματα, θα μελετήσουμε την εφαρμογή μας μέσα από τη μέθοδο Balanced Scorecard²⁷ (ελληνική απόδοση «πλαίσιο εξισορροπημένης στοχοθεσίας» ή «πίνακας σταθμισμένων δεικτών απόδοσης»), μία μέθοδο που χρησιμοποιείται από τους μανάτζερ ως ένα πολύ χρήσιμο εργαλείο για τη μέτρηση της απόδοσης.

Φυσικά, ο σκοπός μας δεν είναι να εφαρμόσουμε πλήρως τη μέθοδο Balanced Scorecard στην εταιρεία του υποθετικού μας μοντέλου - κάτι τέτοιο θα ταίριαζε περισσότερο στη μελέτη ενός ειδικού της διοίκησης επιχειρήσεων – αλλά, με τη χρήση της μεθόδου, να αξιολογήσουμε και να τεκμηριώσουμε την εργασία μας και να βοηθήσουμε τους ειδικούς της διοίκησης επιχειρήσεων να αναγνωρίσουν τα στοιχεία της επιστήμης τους στη δουλειά μας. Πρακτικά, αυτό που θα δούμε είναι η ανάλυση με βάση το Balanced Scorecard ενός helpdesk περιβάλλοντος.

Στο κεφάλαιο αυτό λοιπόν αρχικά παρουσιάζεται θεωρητικά η μέθοδος Balanced Scorecard. Στη συνέχεια μελετάμε την εφαρμογή από την πλευρά του μανάτζμεντ, και εντοπίζουμε τα χαρακτηριστικά της εφαρμογής που βοηθούν στη μέτρηση απόδοσης, στη λήψη αποφάσεων και τη χάραξη στρατηγικών.

5.1 Balanced Scorecard – Θεωρητικό υπόβαθρο

5.1.1. Εισαγωγή

Η μέθοδος Balanced Scorecard (BSC) αναπτύχθηκε τη δεκαετία του '90 από τους καθηγητές Robert S. Kaplan και David P. Norton, αρχικά με σκοπό να καλύψουν την αδυναμία που είχαν τα παραδοσιακά λογιστικά συστήματα στην αποτίμηση άυλων πόρων, καθώς μετρούσαν την αποδοτικότητα μιας επιχείρησης μόνο μέσω οικονομικών μεγεθών και δεικτών.

Η μέθοδος BSC ήρθε για να αλλάξει αυτή την προσέγγιση και να αντιμετωπίσει την επιχείρηση ως ένα μεγαλύτερο σύνολο με διαφορετικές πτυχές. Χαρακτηριστικά οι Kaplan και Norton στο άρθρο τους «Using Balanced Scorecard as a management system» αναφέρουν «Η μέθοδος Balanced Scorecard συμπληρώνει τους παραδοσιακούς οικονομικούς δείκτες με κριτήρια που μετρούν την απόδοση από τρεις επιπλέον οπτικές – του πελάτη, των εσωτερικών διαδικασιών, της μάθησης και ανάπτυξης... Οι εταιρείες πρέπει να δημιουργήσουν μελλοντική αξία μέσα από την επένδυση σε πελάτες, σε προμηθευτές, υπαλλήλους, διαδικασίες, τεχνολογία και καινοτομία». [36]

²⁷ Balanced Scorecard Institute. Επίσημος ιστότοπος: <http://www.balancedscorecard.org>

Η άποψη αυτή δίνει κύρος στην εργασία μας ως επιστήμονες της πληροφορικής και της ανάπτυξης λογισμικού. Η υιοθέτηση μιας εφαρμογής, η οποία επιφέρει αλλαγή στις διαδικασίες, στη τεχνολογία μιας επιχείρησης και μπορεί επιπλέον να έχει στοιχεία καινοτομίας, δεν αποτελεί ένα ανεξάρτητο παράγοντα αλλά πλέον με μετρήσιμο τρόπο συνδέεται με την απόδοση και επιτυχία της επιχείρησης.

Για το λόγο αυτό επιλέξαμε τη μέθοδο αυτή για την αξιολόγησή μας. Η αξία της μεθόδου Balanced Scorecard τεκμηριώνεται στο άρθρο των Mooraj, Oyon, Hostettler «The balanced scorecard: a necessary good or unnecessary evil?». Οι συγγραφείς καταλήγουν: «Η μέθοδος balanced scorecard είναι ένα 'αναγκαίο καλό' για τους σημερινούς οργανισμούς. Προσθέτει αξία επειδή παρέχει στους μάνατζερ σχετικές και εξισορροπημένες πληροφορίες με περιεκτικό τρόπο». [37]

Το μοντέλο εξελίχθηκε και έχει πλέον διαμορφωθεί ως ένα εργαλείο πρότυπο για μέτρηση απόδοσης από πλευράς μάνατζμεντ, και στις μέρες μας υποστηρίζεται από πολλές αυτόματες μεθόδους και εργαλεία σχεδίασης, τα οποία χρησιμοποιούνται ευρέως από μάνατζερ ώστε να βλέπουν τον αντίκτυπο των ενεργειών των υπαλλήλων, διαδικασιών κ.α. στην απόδοση – επιτυχία της εταιρείας. [38, 39, 40]

Οι τέσσερις οπτικές

Όπως είδαμε παραπάνω, η μέθοδος BSC, προτείνει να δούμε την εταιρεία μέσα από τέσσερις διαφορετικές οπτικές: την οικονομική οπτική (financial perspective), την οπτική του πελάτη (customer perspective), την οπτική των εσωτερικών διαδικασιών (internal business processes perspective), την οπτική της μάθησης και ανάπτυξης (learning and growth perspective). [36],

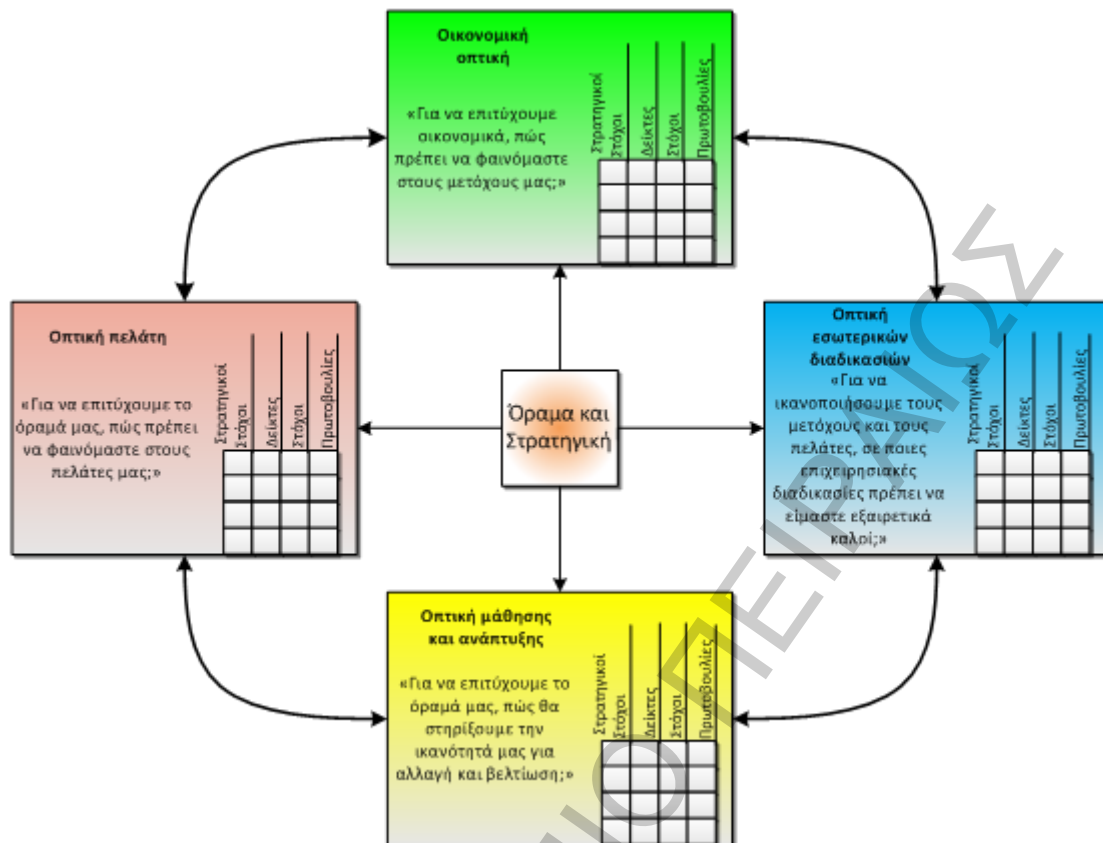
Η οικονομική οπτική ορίζει πώς η εταιρεία επιθυμεί να φαίνεται στους μετόχους. Πρακτικά συνοψίζει «τις μετρήσιμες οικονομικές συνέπειες ενεργειών που έχουν ήδη πραγματοποιηθεί» [38]

Η οπτική του πελάτη ορίζει πώς η εταιρεία επιθυμεί να φαίνεται στους πελάτες. Περιέχει μέτρα που «αναγνωρίζουν τους τομείς πελατών και αγοράς για τους οποίους θα ανταγωνιστεί η επιχείρηση και τους δείκτες της απόδοσης της εταιρείας σε αυτούς τους στοχευόμενους τομείς» [38]

Η οπτική των εσωτερικών διαδικασιών περιγράφει τις επιχειρησιακές διαδικασίες που πρέπει να υιοθετηθούν για να είναι ικανοποιημένοι οι μέτοχοι και οι πελάτες. Μετρά «τις κρίσιμες εσωτερικές διαδικασίες στις οποίες ο οργανισμός πρέπει να είναι εξαιρετικά καλός». [38]

Η οπτική της μάθησης και ανάπτυξης μας δείχνει τις αλλαγές και βελτιώσεις που πρέπει να υποστεί μια εταιρεία για να πραγματοποιήσει το όραμά της. Δείχνει «την υποδομή που πρέπει να χτίσει ο οργανισμός για να δημιουργήσει μακροπρόθεσμη ανάπτυξη και βελτίωση» [38]

Τα παραπάνω αναπαρίστανται από τους δημιουργούς του Balanced Scorecard στο σχήμα που ακολουθεί:



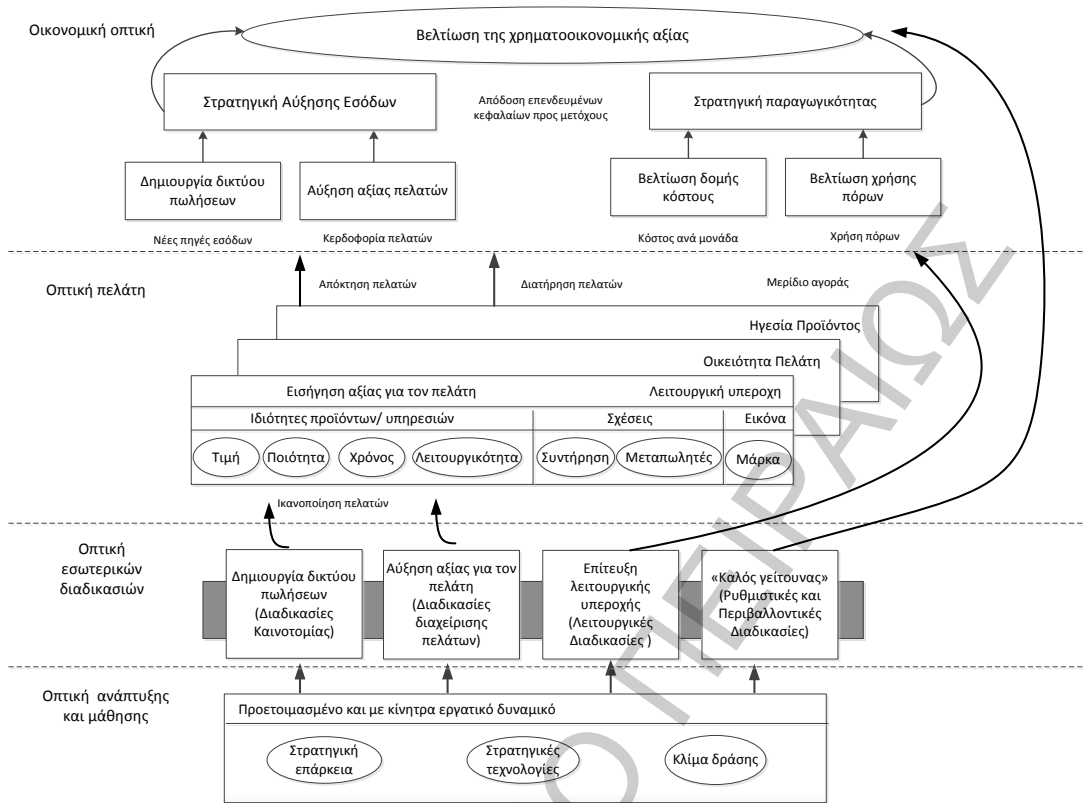
Σχήμα 18 Η χρήση του Balanced Scorecard ως ένα σύστημα στρατηγικού μανάτζμεντ

Πηγή: Kaplan R, Norton D. «Using the Balanced Scorecard as a Strategic Management System» [36, 41]

5.1.2 Balanced Scorecard in action

Μία από τις πιο χρήσιμες προσθήκες στην θεωρία του balanced scorecard είναι ο στρατηγικός χάρτης balanced scorecard (Kaplan και Norton, 2001). Ένας καλά σχεδιασμένος χάρτης στρατηγικής παρουσιάζει ξεκάθαρα τη στρατηγική της εταιρείας και τους δείκτες της επιτυχούς υλοποίησής της. Μας επιτρέπει να έχουμε μια συνοπτική αλλά συγχρόνως και ολική εικόνα των κρίσιμων στρατηγικών ενεργειών για κάθε έναν από τους τέσσερις τομείς. [38, 42]

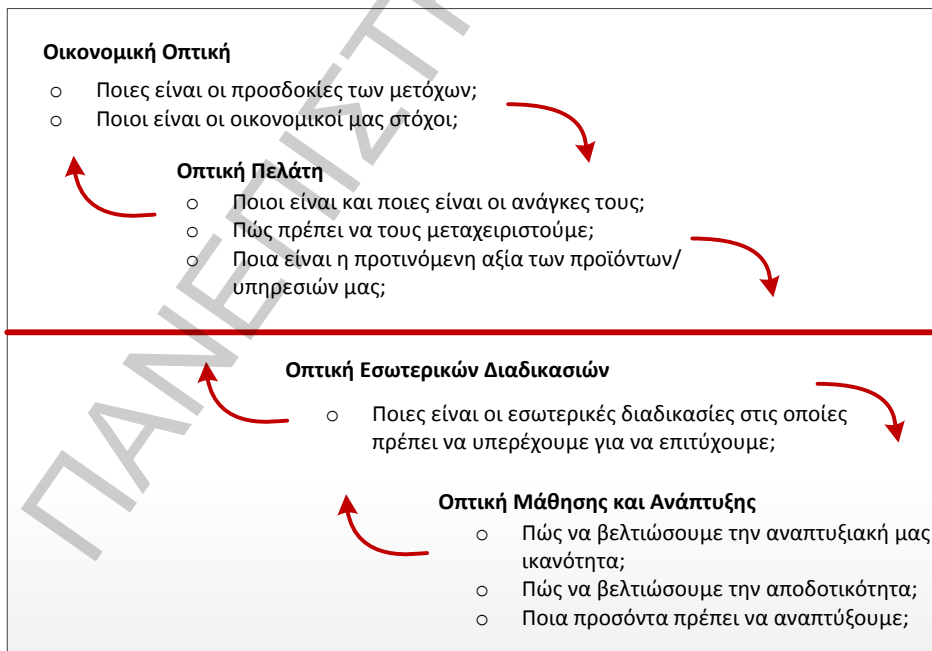
Ένα πρότυπο ενός στρατηγικού χάρτη δίδεται από τους Kaplan και Norton όπως παρουσιάζεται στην επόμενη εικόνα (εικόνα 52).



Εικόνα 52 Πρωτότυπο στρατηγικού χάρτη

Πηγή Kaplan, R, Norton, D «Partnering and the Balanced Scorecard» [42,43]

Με την προσέγγιση αυτή λαμβάνουμε απαντήσεις σε βασικά ερωτήματα των τεσσάρων οπτικών, όπως αυτά που παρουσιάζονται στο σχήμα 19.



Σχήμα 19 Ερωτήματα τεσσάρων οπτικών
Πηγή AQS [44]

Για περισσότερα σχετικά με το Balanced Scorecard μπορείτε να ανατρέξετε στη διεθνή βιβλιογραφία η οποία είναι εκτεταμένη. Μπορείτε εύκολα να βρείτε πολλά θεωρητικά στοιχεία και ολοκληρωμένα παραδείγματα υλοποιήσεων, κάποια από τα οποία προτείνουμε και εμείς στη βιβλιογραφία μας.

Ολοκληρώνοντας μόνο να πούμε για κάποιον που θέλει να υλοποιήσει ολοκληρωμένο το balanced scorecard ότι το συγκεκριμένο έργο έχει αρκετά στάδια που χρησιμοποιούν διάφορα εργαλεία του μάνατζμεντ. [42, 45]

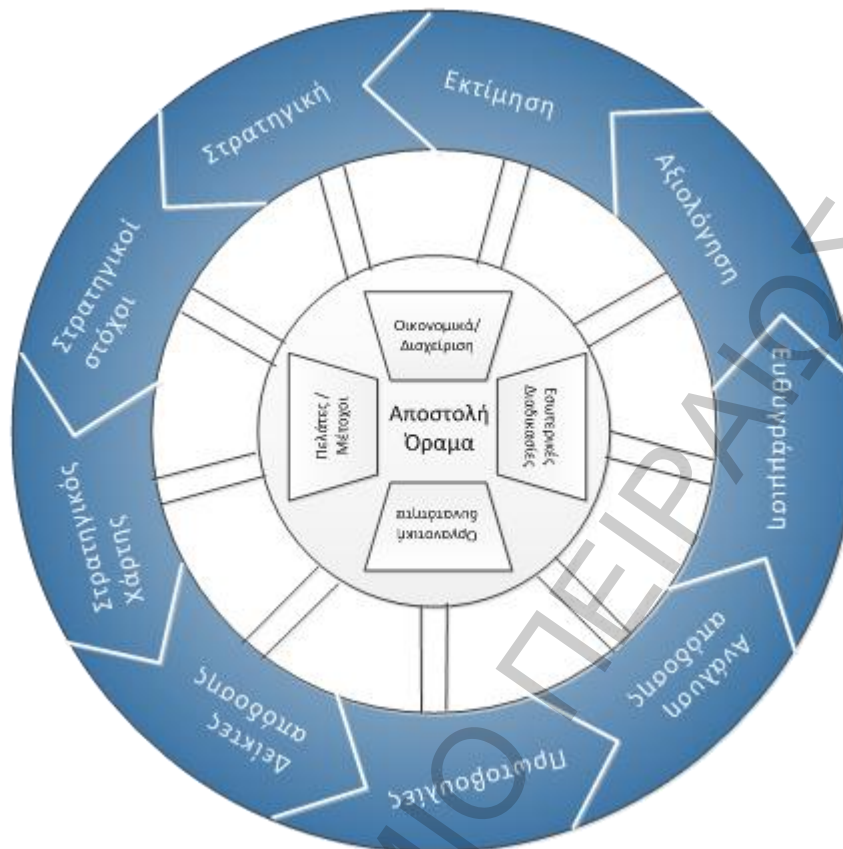
Τυπικά ένα τέτοιο έργο περιλαμβάνει:

- Αναγνώριση εσωτερικού περιβάλλοντος. Μέσα από συνεντεύξεις, συγκέντρωση διαδικασιών κ.α. προσπαθούμε να γνωρίσουμε καλά το περιβάλλον της επιχείρησης.
- Αποτύπωση των γνώσεων για την εταιρεία με μία μέθοδο π.χ. SWOT, ανάλυση και ανασκόπηση της στρατηγικής που ήδη εφαρμόζεται. Εδώ ανακαλύπτουμε τη δυναμική της επιχείρησης και εξετάζουμε αν η στρατηγική που εφαρμόζεται ήδη καλύπτει ακόμα τις ανάγκες της επιχείρησης.
- Ανάπτυξη στρατηγικού χάρτη.
- Ανάπτυξη των δεικτών από τις τέσσερις οπτικές. Ποια είναι εκείνα τα στοιχεία που θα μετρήσουμε για να επαληθεύσουμε τη στρατηγική; Ο χάρτης στρατηγικής συμπληρώνεται με τις μετρήσεις.
- Ανάπτυξη στρατηγικών στόχων. Θέτουμε στόχους σε σχέση με τους δείκτες. Ο χάρτης στρατηγικής συμπληρώνεται με τους στόχους
- Ανάπτυξη στρατηγικών πρωτοβουλιών. Διαμηματικές πρωτοβουλίες που συμβάλουν στην υλοποίηση των στρατηγικών σε βάθος χρόνου. Ο χάρτης στρατηγικής συμπληρώνεται με τα βοηθητικά προγράμματα δράσης.

Το ινστιτούτο Balanced Scorecard ²⁸ έχει αναπτύξει και προτείνει το δικό του πλαίσιο πρακτικής προσέγγισης ανάπτυξης στρατηγικού σχεδιασμού και διαχείρισης συστημάτων με βάση τη μέθοδο balanced scorecard. Το πλαίσιο αυτό ονομάζεται Εννέα βήματα για την επιτυχία (Nine Steps to SuccessTM), παρουσιάζεται γραφικά στο σχήμα 20. Για περισσότερες πληροφορίες επισκεφθείτε την επίσημη σελίδα του ινστιτούτου²⁹.

²⁸ ©1998-2013 Balanced Scorecard Institute

²⁹ Επίσημος ιστότοπος του πλαισίου The nine steps success TM
<http://www.balancedscorecard.org/bscresources/theninestepstosuccess/tabid/58/default.aspx>



Σχήμα 20 The nine steps success TM

Πηγή (υποσημείωση 29)

Στη συνέχεια θα επικεντρωθούμε στο θεωρητικό μας μοντέλο του υποθετικού σεναρίου, θα κάνουμε μια ανασκόπηση των επιχειρησιακών στοιχείων του πρώτου μέρους και θα δούμε πως μπορούμε να αντιμετωπίσουμε την εφαρμογή μας από την πλευρά του μανάτζμεντ.

5.2 Balanced Scorecard - Στοιχεία μανάτζμεντ στην εφαρμογή μας

5.2.1 Πριν την αγορά

Πριν πάμε στα επιμέρους οφέλη που έχει η επιχείρηση από την αξιοποίηση της εφαρμογής, ας δούμε πρώτα από όλα αν μπορούμε να απαντήσουμε στο κύριο ερώτημα ενός επενδυτή. Γιατί κάποιος να δώσει χρήμα και χρόνο για ένα νέο προϊόν, να εντάξει δηλαδή μια νέα τεχνολογία στις διαδικασίες του

Όπως είδαμε ήδη στην ανάλυση των εφαρμογών Helpdesk είναι πολύ εύκολο να πείσουμε για την επιτακτική ανάγκη χρήσης της εφαρμογής, αν λάβουμε υπ' όψιν τις δηλώσεις περίπτωσης ανάγκης λογισμικού της παραγράφου 1.1.2. Οι δηλώσεις αυτές αφορούν περιπτώσεις όπου η λειτουργία της εταιρείας βρίσκεται σε κίνδυνο και η ικανοποίησή τους είναι απαραίτητη.

Να θυμίσουμε για παράδειγμα «Όταν οι πελάτες της εταιρείας επικοινωνούν συχνά για ένα μόνο θέμα τους, δυσκολεύουν επιπλέον τη διαδικασία επίλυσης». Η πολλαπλή αυτή επικοινωνία επιφέρει τα εξής:

Ένας ή περισσότεροι υπάλληλοι της εταιρείας είναι υποχρεωμένοι να απασχολούνται επαναλαμβανόμενα με ένα συγκεκριμένο θέμα, να δίνουν πληροφορίες ακόμα και όταν δεν υπάρχει κάτι νεότερο. Συχνά πολλοί υπάλληλοι απασχολούνται ταυτόχρονα, καθυστερώντας και

όχι επιταχύνοντας τη λύση, αφού πιθανότερα, ο ένας δε γνωρίζει για τις ενέργειες του άλλου. Το τηλεφωνικό κέντρο ο χρόνος του οποίου είναι πάντα κρίσιμος, απασχολείται για ένα θέμα το οποίο είναι ήδη γνωστό. Ο πελάτης μπορεί να λαμβάνει διαφορετικές και συγκεχυμένες πληροφορίες, κάνοντας την εταιρεία να φαίνεται αναξιόπιστη. Στην τελική, όταν ένας πελάτης καλεί επαναλαμβανόμενα φαίνεται ότι είναι ένας πελάτης ανικανοποίητος και πιθανότερα δυσαρεστημένος.

Όλες αυτές οι δηλώσεις «κρύβουν» μέσα τους τέτοιες μικρές ιστορίες, που αν τις κατέχει ένας πωλητής πείθει εύκολα τους πιθανούς αγοραστές. Γιατί τώρα κάποιος να επιλέξει τη δική μας εφαρμογή; Ελπίζουμε ότι με τη μέχρι στιγμής ανάγνωση των προηγούμενων κεφαλαίων του παρόντος, σας έχουμε πείσει αρκετά για τα οφέλη, τα πλεονεκτήματα και την καινοτομία της εφαρμογής μας. Πιστεύουμε ότι στη συνέχεια της ανάλυσης μας από πλευράς μανάτζμεντ θα ανακαλύψουμε επιπλέον οφέλη από τη χρήση της εφαρμογής.

5.2.2 Μελέτη τμημάτων αρωγής μέσα από τις οπτικές του Balanced Scorecard

Ας θυμηθούμε αρχικά με λίγα λόγια το υποθετικό σενάριο της εταιρείας μας. Μία αναπτυσσόμενη εταιρεία παροχής υπηρεσιών, δημιουργεί τμήμα τεχνικής υποστήριξης για να εξυπηρετεί αιτήματα τεχνικής φύσης από τους υπαλλήλους, τους πελάτες της και τους συνεργάτες της (τμήμα αρωγής). Για τη μελέτη από πλευράς μανάτζμεντ θα εστιάσουμε ακόμα περισσότερο στο κομμάτι του help desk που αφορά εξυπηρέτηση πελατών.

Φυσικά το οργανόγραμμα της εταιρείας περιέχει περισσότερα τμήματα και δ/νσεις, όμως όπως έχουμε ήδη πει, σκοπός του παρόντος δεν είναι η επιχειρησιακή μελέτη του συνόλου της εταιρείας, αλλά περισσότερο η μελέτη της λειτουργίας ενός τμήματος αρωγής. Σε αυτό το σημείο λοιπόν, με τη βοήθεια του balanced scorecard θα δούμε πώς πρέπει να εξετάζεται από πλευράς μανάτζμεντ ένα τμήμα αρωγής, και με ποιον τρόπο βοηθά η εφαρμογή μας σε αυτή τη μελέτη.

Ένα τμήμα αρωγής έχει πάντα ως κύριο σκοπό του την ικανοποίηση των πελατών μέσα από την εξυπηρέτηση αιτημάτων. Στόχος του τμήματος αρωγής είναι η εξυπηρέτηση όλων των αιτημάτων όσο το δυνατόν γρηγορότερα, αρτιότερα, οικονομικότερα. Το ιδανικό είναι όταν λαμβάνεται μία κλήση (αίτημα) αυτή να εξυπηρετείται άμεσα, από τον τεχνικό που θα λάβει την κλήση, χωρίς να μένει σε εκκρεμότητα το αίτημα (να κλείνει άμεσα το δελτίο) και μάλιστα σε μικρό χρονικό διάστημα.

Σε ένα τέτοιο περιβάλλον, ποια είναι πραγματικά η δουλειά ενός μανάτζερ; Μα φυσικά η αύξηση της απόδοσης και της αποτελεσματικότητας. Όλα τα αιτήματα πρέπει να φτάνουν όσο το δυνατό περισσότερο την ιδανική περίπτωση εξυπηρέτησης. Ο μανάτζερ πρέπει να βρει τρόπο ώστε τα αιτήματα να εξυπηρετούνται όσο το δυνατό καλύτερα, γρηγορότερα, με το χαμηλότερο δυνατό κόστος. Το Balanced Scorecard έρχεται ως εργαλείο για τους μανάτζερ, ώστε να επιτύχουν την απόδοση και την αποτελεσματικότητα, αλλά και για να πείσουν τη διοίκηση για την καλή λειτουργία του helpdesk.

Ας μπούμε λοιπόν στη θέση ενός μανάτζερ υπεύθυνου του τμήματος αρωγής της εταιρείας μας και ας δούμε με ποιον τρόπο θα καταφέρουμε τα παραπάνω.

5.2.2.1 Στρατηγικός χάρτης

Ποιοι μπορεί να είναι λοιπόν οι στόχοι και τα στοιχεία (δείκτες) ενός τμήματος αρωγής που μπορούν να μετρηθούν ώστε να αξιολογηθεί η εργασία του; Από τη μελέτη που κάναμε και την εμπειρία μας καταλήγουμε στα εξής:

Από την οικονομική οπτική

Είναι σημαντικό το τμήμα αρωγής να μη θεωρείται απλά ένα κέντρο κόστους. Πρέπει να γίνει κατανοητό στη διοίκηση ότι τα χρήματα που δαπανούνται για τη λειτουργία του δεν είναι χαμένα αλλά ότι αυτά «επενδύονται» για την αξία του προϊόντος και για την καλή φήμη της εταιρείας. Οι οικονομικοί στόχοι που τίθενται συνήθως για ένα τμήμα αρωγής είναι η καλή διαχείριση του προϋπολογισμού και η μείωση λειτουργικού κόστους.

Από την πλευρά του πελάτη

Η υποστήριξη των πελατών πριν και μετά την πώληση κρίνει κατά ένα πολύ μεγάλο βαθμό το επιτυχημένο προφίλ μιας εταιρείας. Είναι εξαιρετικά σημαντικό ο πελάτης να βρίσκει ανταπόκριση και λύση στο πρόβλημά του για να είναι ευχαριστημένος. Πόσες φορές δεν έχετε σκεφτεί και εσείς οι ίδιοι να προτιμήσετε ένα προϊόν επειδή σας πρόσφεραν καλές υπηρεσίες πριν την αγορά, ή επειδή πιστεύετε ότι η εταιρεία του συγκεκριμένου προϊόντος προσφέρει καλύτερες υπηρεσίες μετά την πώληση σε σχέση με τους ανταγωνιστές της;

Μπορούμε να πούμε λοιπόν ότι πελατειακοί στόχοι για ένα τμήμα αρωγής είναι να δημιουργεί νέους πελάτες, αλλά κυρίως να διατηρεί (ικανοποιεί) τους υπάρχοντες πελάτες.

Εσωτερικές διαδικασίες

Με ποιον τρόπο λοιπόν πρέπει να λειτουργεί ένα τμήμα αρωγής ώστε να πετύχει τους οικονομικούς και πελατειακούς του στόχους; Θεωρούμε απαραίτητα τη σωστή διαχείριση κλήσεων, τη θέσπιση διαδικασιών και τη χρήση εργαλείων για α) τη διαχείριση των κλήσεων (τηλεφωνικό κέντρο) β) τη διαχείριση των δελτίων (εφαρμογή Helpdesk).

Όπως είδαμε και στην ανάλυση της εφαρμογής (βλέπε Κεφάλαιο 2), οι υπάλληλοι πρέπει να έχουν διακριτούς ρόλους και να γνωρίζουν καλά τη ροή που πρέπει να έχουν τα αιτήματα. Φυσικά και το πρώτο εργαλείο του helpdesk είναι η επικοινωνία. Όμως με τα τεχνολογικά εργαλεία των τηλεφωνικών κέντρων και των εφαρμογών helpdesk ελέγχουμε και εξασφαλίζουμε την ποσότητα και την ποιότητα των υπηρεσιών μας.

Μπορούμε λοιπόν να πούμε για τις εσωτερικές διαδικασίες ότι έχουμε τρεις κύριες κατευθύνσεις α) την επικοινωνία, β) την καινοτομία, γ) τη λειτουργικότητα και δύο κύριους στόχους: α) τη τήρηση των διαδικασιών (για την εξασφάλιση της ποιότητας) β) την τεχνολογική εξέλιξη.

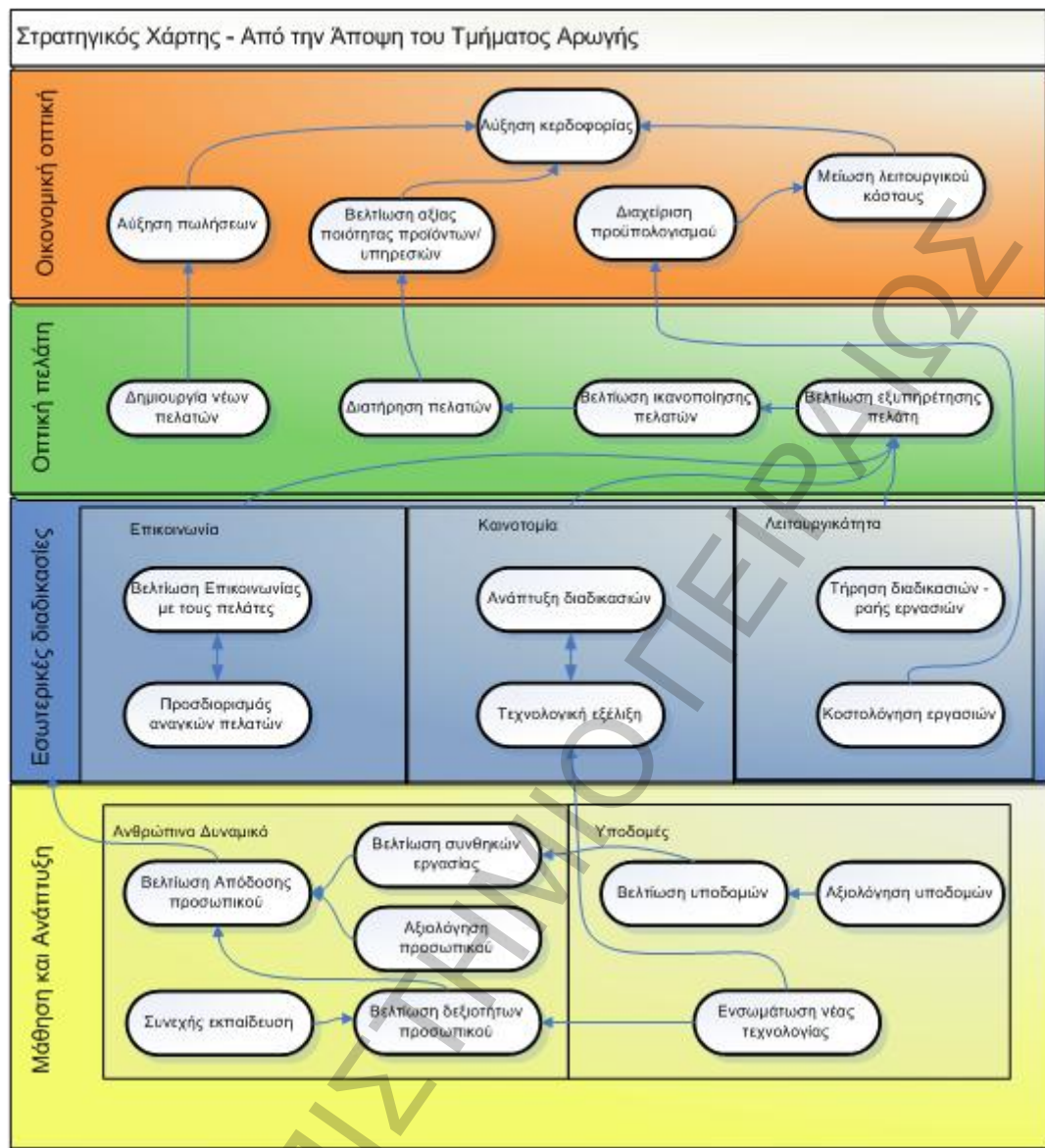
Εκπαίδευση και Ανάπτυξη

Το προσωπικό των τμημάτων αρωγής είναι το «πρόσωπο» αλλά και τα «αυτιά» της εταιρείας. Πολλές φορές ένας υπάλληλος του τμήματος αρωγής είναι ο μοναδικός εκπρόσωπος της εταιρείας που επικοινωνούν οι πελάτες. Οι υπάλληλοι των τμημάτων αρωγής πρέπει με υπομονή και ευγένεια να δίνουν τεκμηριωμένες υπηρεσίες (λύσεις) σε μικρό χρόνο, ώστε να κρατούν τον προϋπολογισμό χαμηλά και τους πελάτες ικανοποιημένους.

Η συμπεριφορά, η κατάρτισή του, οι γνώσεις και ικανότητές του προάγουν ή υποβαθμίζουν την καλή φήμη της εταιρείας. Επίσης, η ικανότητά του να αξιολογεί ότι ακούει για όφελος της εταιρείας είναι ένα πολύ σημαντικό χαρακτηριστικό για τη λήψη αποφάσεων για την πολιτική της εταιρείας.

Το τμήμα αρωγής στηρίζεται στο ανθρώπινο δυναμικό του και γι αυτό έχει ως στόχο και πρέπει να γίνονται: α) συστηματικές εκπαιδεύσεις, β) συνεχείς αξιολογήσεις αλλά και γ) να ελέγχεται η ανάπτυξη του ανθρώπινου δυναμικού, δ) να αξιολογείται και να αναπτύσσονται οι υποδομές.

Από τα παραπάνω μπορούμε να καταλήξουμε σε ένα χάρτη στρατηγικής περίπου σαν αυτόν που εμφανίζεται ακολούθως (εικόνα 53). Στο χάρτη αυτό βλέπουμε ποιοι είναι οι στόχοι αλλά και πώς αυτοί συνδέονται μεταξύ τους ώστε να οδηγηθούμε στην εκπλήρωση του οράματος της εταιρείας.



Εικόνα 53 Στρατηγικός χάρτης από την πλευρά των τμημάτων αρρωγής

Ήδη από τα παραπάνω έχει αρχίσει να φαίνεται καθαρά η αναγκαιότητα και η χρησιμότητα των εφαρμογών helpdesk. Οι οπτικές εσωτερικών διαδικασιών και μάθησης και ανάπτυξης περιέχουν στόχους σαφώς προσανατολισμένους στην τεχνολογική ανάπτυξη ως υπόβαθρο των στόχων των οπτικών οικονομίας και πελατών.

Αργότερα θα δούμε αναλυτικότερα για την εφαρμογή μας ποιους στόχους βοηθά να εκπληρώσουμε αλλά και ποιους δείκτες μας προσφέρει για την εκπλήρωσή τους.

5.2.2.2 Δείκτες helpdesk

Αφού είδαμε τους στόχους που μπορεί να έχει θέσει μια εταιρεία και ειδικότερα ένα τμήμα αρρωγής για την εκπλήρωση του οράματος της εταιρείας, συνεχίζουμε με τη μελέτη των δεικτών. Όλοι οι παραπάνω στόχοι μπορούν να μετρηθούν με τους κατάλληλους δείκτες. Στη συνέχεια θα επικεντρωθούμε ιδιαίτερα σε εκείνους τους δείκτες που μετρούνται συνήθως σε τμήματα αρρωγής.

Από τη μελέτη μας και την εμπειρία μας καταλήξαμε ότι για τα τμήματα τηλεφωνικής εξυπηρέτησης, υποσύνολο των οποίων είναι συνήθως τα τμήματα αρωγής, μετρούνται δείκτες όπως: [46, 47]

- **Έσοδα:** Πόσο τιμολογήσαμε τις υπηρεσίες που προσφέρθηκαν; Πρέπει να γίνεται περιοδικός απολογισμός των εσόδων που απέφεραν οι υπηρεσίες helpdesk.
- **Κόστος ανά δελτίο (συναλλαγή κ.α.):** Ποιο είναι το πραγματικό οικονομικό κόστος για την επίλυση ενός προβλήματος; Τι έχει κοστίσει στην εταιρεία; Εδώ μπορούν να υπολογίζονται τα τηλεπικοινωνιακά έξοδα, οι ανθρωποώρες, ανταλλακτικά που χρειάστηκαν για τη βλάβη κ.α.
- **Ρυθμός εγκατάλειψης (κλήσεων, κ.α. Abandon rate –ABN):** Είναι το ποσοστό των κλήσεων που εγκαταλείφθηκαν προς τις συνολικές κλήσεις που λάβαμε. Φυσικά θέλουμε αυτόν τον αριθμό πολύ μικρό. Μία κλήση που δεν εξυπηρετήθηκε είναι ένας δυνητικά δυσαρεστημένος πελάτης.
- **Μέσος χρόνος απόκρισης σε κλήση (Average Speed of Answer – ASA):** Είναι ο μέσος χρόνος που περιμένει κάποιος για να απαντηθεί η κλήση του. Και εδώ θέλουμε αυτό τον αριθμό μικρό, μιας και δεν θέλουμε να κάνουμε τους πελάτες να περιμένουν.
- **Ποσοστό λύσης στην πρώτη κλήση (First Call Resolution – FCR):** Είναι ο αριθμός των αιτημάτων που εξυπηρετήθηκαν με την πρώτη κλήση. Ο δείκτης αυτός είναι πολύ σημαντικός για να διαπιστώσει κανείς ένα σύνολο πραγμάτων. Μία κλήση που εξυπηρετείται με την πρώτη σημαίνει: α) ικανοποιημένος πελάτης, β) πιθανά μικρότερο κόστος επίλυσης γ) έμπειρο προσωπικό δ) καλή διαδικασία - ροή επίλυσης. Φυσικά θέλουμε αυτό το ποσοστό να είναι μεγάλο.
- **Μέσος χρόνος χειρισμού (Average Handling Time – AHT):** Είναι ο συνολικός χρόνος που ξοδεύτηκε για την επίλυση του προβλήματος. Περιλαμβάνει α) τον χρόνο που ξοδεύτηκε για την κλήση (μαζί με το χρόνο που μπορεί να είχαμε τον πελάτη σε αναμονή), αλλά και β) το χρόνο που χρειάστηκε για τη συμπλήρωση των δελτίων, δηλαδή για την καταγραφή του συμβάντος. Ο χρόνος αυτό μετράται είτε ανά κλήση είτε ανά δελτίο. Φυσικά και θέλουμε αυτό τον αριθμό μικρό.
- **Επίπεδο υπηρεσιών (Service Level):** Είναι ο αριθμός των κλήσεων που απαντήθηκαν ή ο αριθμός των ενεργειών που έγιναν σε ένα συγκεκριμένο χρονικό διάστημα.
- **Αναλογία ανοιχτών/κλειστών δελτίων:** Πόσα δελτία εκκρεμούν και πόσα εξυπηρετήσαμε; Ένας από τους πιο σημαντικούς δείκτες από τους οποίους μπορούν να βγουν αρκετά συμπεράσματα. Τελικά, στο τέλος της μέρας «πόση» δουλειά έγινε; Αν ο αριθμός των δελτίων που ανοίγουν είναι πολύ μικρός, ίσως θα πρέπει να αναθεωρηθεί ο προϋπολογισμός και η στελέχωση του τμήματος αρωγής ή ακόμα και να χρειάζεται εκπαίδευση στους υπαλλήλους ή τους πελάτες μας ώστε να χρησιμοποιούν την εφαρμογή help desk. Αν ο αριθμός των δελτίων που κλείνουν είναι πολύ μικρός, ο αριθμός των εργασιών θα αρχίζει να συσσωρεύεται, και πιθανά να μην μπορούν να εξυπηρετηθούν είτε επειδή υπάρχει έλλειψη προσωπικού, είτε από αδυναμία του προσωπικού που μπορεί να χρειάζεται εκπαίδευση, είτε ακόμα από πρόβλημα στις διαδικασίες αλλά και από εξωτερικούς παράγοντες. Φυσικά χρειάζεται κάποια ισορροπία ανάμεσα στα εισερχόμενα (ανοιχτά) και εξερχόμενα (κλειστά) αιτήματα.
- **Ποιότητα:** Δεν παίζει ρόλο μόνο αν το αίτημά σου εξυπηρετήθηκε ή όχι. Όταν κάποιος σου ζητά κάποια υπηρεσία περιμένει από εσένα να ενδιαφερθείς για το πρόβλημά του, περιμένει ευγένεια και επαγγελματισμό στην εξυπηρέτησή του. Ο τρόπος με το οποίο συναλλασσόμαστε με τους πελάτες καθορίζει σε ένα μεγάλο βαθμό την ποιότητα στις υπηρεσίες που προσφέρουμε.
- **Βαθμός ικανοποίησης πελατών - (Customer satisfaction score – CSAT):** Συνήθως με ερωτηματολόγια και έρευνα προσπαθούμε να μάθουμε πόσο ευχαριστημένοι είναι οι πελάτες μας από τις υπηρεσίες που έλαβαν. Φυσικά θέλουμε τον αριθμό αυτό μεγάλο.

- **Ενεργός χρόνος πρακτόρων:** Πόση ώρα είναι πραγματικά απασχολημένοι οι υπάλληλοί μας; Μπορεί να μη θέλουμε να κάνουμε τους πελάτες μας να περιμένουν και να κρατάμε το ρυθμό εγκατάλειψης και το χρόνο απόκρισης μικρούς, όμως με ποιό κόστος; Φυσικά και δε θέλουμε τους υπαλλήλους μας να κάθονται. Πρέπει να υπάρχει κάποια ισορροπία σε σχέση με τον αριθμό των υπαλλήλων μας και τις εισερχόμενες κλήσεις.
- **Ευσυνειδησία - απουσία υπαλλήλων:** Μια εταιρεία δεν φτάνει να θέλει να κρατά τους πελάτες της ευχαριστημένους, πρέπει πρώτα από όλα να φροντίζει για τους υπαλλήλους της, ώστε να μπορούν και αυτοί με τη σειρά τους να φροντίζουν για την εταιρεία. Ένα μεγάλο πρόβλημα το οποίο φαίνεται να αντιμετωπίζουν - ιδιαίτερα τα μεγάλα - τμήματα υποστήριξης είναι ο μεγάλος χρόνος που οι πράκτορες τείνουν να είναι ανενεργοί (δεν λαμβάνουν νέα αιτήματα) και ο μεγάλος βαθμός μη προγραμματισμένων απουσιών των υπαλλήλων. Αυτό συμβαίνει κυρίως λόγω των μεγάλων απαιτήσεων που έχει η εργασία στα τμήματα αρωγής σε ωράρια, διαθεσιμότητα, επιμόρφωση, ψυχικό σθένος (για την αντιμετώπιση των παραπόνων) κ.α.
Οι μάνατζερ πρέπει να δίνουν λύσεις στα προβλήματα των υπαλλήλων, να διατηρούν ένα καλό εργασιακό περιβάλλον, να δίνουν κίνητρα στην εργασία και να επιβραβεύουν την ευσυνειδησία. Η συμπεριφορά των υπαλλήλων είναι πράγματι πολύ σημαντική για τη απόδοση της εταιρείας και για το λόγο αυτό η ευσυνειδησία των υπαλλήλων είναι ένας από τους μετρήσιμους δείκτες για την αξιολόγησή της.
- **Συνεχής βελτίωση – επιμόρφωση των πρακτόρων:** Η ποιότητα του ανθρώπινου δυναμικού των τμημάτων αρωγής είναι πολύ σημαντική, όπως αναλύσαμε και παραπάνω στην οπτική της εκπαίδευσης και ανάπτυξης. Τόσο για το ίδιο το προσωπικό, όσο και για την εταιρεία είναι απαραίτητη η συνεχής επιμόρφωση του προσωπικού στις νέες διαδικασίες, τις τεχνολογικές εξελίξεις κ.α. κυρίως λόγω της ίδιας της φύσης της εργασίας που απαιτεί από τους εργαζόμενους να γνωρίζουν καλύτερα από άλλους τα θέματα τα οποία εξυπηρετούν.

Είδαμε κάποιους από τους δείκτες που θα μπορούσαν να επιλεγούν από ένα μάνατζερ τμήματος τηλεφωνικής υποστήριξης ως μέτρα βαθμολόγησης. Όλοι οι παραπάνω δείκτες συνδέονται με έναν ή περισσότερους στόχους όλων των οπτικών. Φυσικά υπάρχουν και άλλοι, και κάθε εταιρεία και τμήμα θα πρέπει να επιλέξει αυτούς που θεωρεί ότι ταιριάζουν καλύτερα στους στόχους και στο όραμά τους. Επιλέγουμε και εμείς εννέα από τους παραπάνω δείκτες που πιστεύουμε ότι ταιριάζουν καλύτερα σε ένα τμήμα helpdesk.

Πίνακας 4 Δείκτες helpdesk

Κόστος ανά δελτίο
Ρυθμός εγκατάλειψης (κλήσεων)
Αναλογία κλειστών δελτίων
Ποσοστό λύσης στην πρώτη κλήση (First Call Resolution – FCR)
Μέσος χρόνος χειρισμού δελτίων (average handling time – AHT)
Επίπεδο υπηρεσιών
Ποιότητα κλήσεων
Βαθμός ικανοποίησης πελατών
Συνεχής βελτίωση - εκπαίδευσης πρακτόρων

Μια απλή και καλή πρακτική εφαρμογή των καρτών του balanced scorecard γίνεται με τη συμπλήρωση πίνακα σε ένα πρόγραμμα λογιστικών φύλλων (excel). Με βάση τα παραπάνω, ακολουθεί ένα ενδεικτικό παράδειγμα του πως περίπου θα συμπληρωνόταν μία κάρτα balanced scorecard, σε ένα πραγματικό περιβάλλον help desk. (πίνακας 5).

Πίνακας 5 Κάρτα balanced scorecard – από τη πλευρά του τμήματος αρωγής

Οπτική	Στρατηγικός στόχος	Δείκτες	Βάρος	Στόχος	Πραγματική απόδοση	Score	Balanced Score	Πηγή μέτρησης
Οικονομίας	Κοστολόγηση	Κόστος ανά δελτίο	15%	5 €	6,50 €	76,92%	11,54%	Λογιστήριο, Εφαρμογή helpdesk
Πελάτη	Βελτίωση εξυπηρέτησης πελάτη	Ποσοστό λύσης στην πρώτη κλήση (FCR)	15%	70%	55%	78,57%	11,79%	Εφαρμογή helpdesk
		Επίπεδο υπηρεσιών (ενέργειες / ημέρα)	10%	200	160	80,00%	8,00%	Εφαρμογή helpdesk
	Βελτίωση ικανοποίησης πελάτη	Βαθμός ικανοποίησης πελατών	20%	90%	70%	77,78%	15,56%	Survey
Εσωτερικών Διαδικασιών	Βελτίωση επικοινωνίας με πελάτες	Ρυθμός εγκατάλειψης (κλήσεων)	5%	5%	10%	50,00%	2,50%	Τηλεφωνικό κέντρο
Εκπαίδευσης & ανάπτυξης	Βελτίωση Απόδοσης προσωπικού	Αναλογία κλειστών δελτίων	10%	80%	70%	87,50%	8,75%	Εφαρμογή helpdesk
		Μέσος χρόνος χειρισμού δελτίων (AHT) σε ώρες	15%	4	6	66,67%	10,00%	Εφαρμογή helpdesk
		Ποιότητα κλήσεων	5%	80%	70%	87,50%	4,38%	Survey
	Συνεχής εκπαίδευση	Συνεχής βελτίωση - εκπαίδευσης πρακτόρων	5%	80%	60%	75,00%	3,75%	Survey
		ΣΥΝΟΛΟ	100%				76,25%	

Στον παραπάνω πίνακα στην πρώτη, δεύτερη και τρίτη στήλη αναγράφονται οι δείκτες που επιθυμούμε να μετρήσουμε, καθώς και σε πια οπτική και στόχο ανήκουν αυτοί. Η κατηγοριοποίηση που έχει γίνει είναι ενδεικτική και κάποιος θα μπορούσε να εντάξει ένα δείκτη σε άλλη οπτική και στόχο.

Στην τέταρτη στήλη αναγράφεται το βάρος, δηλαδή η σημαντικότητα που έχει για εμάς ο δείκτης. Η πέμπτη στήλη δηλώνει τις επιθυμητές τιμές στόχους των δεικτών. Η έκτη στήλη περιέχει τις πραγματικές τιμές των δεικτών, όπως αυτές μετρήθηκαν από τις διάφορες πηγές, οι οποίες εμφανίζονται στην τελευταία στήλη. Η έβδομη στήλη δείχνει το απόλυτο ποσοστό προσέγγισης του στόχου (σχέση ανάμεσα στόχου και πραγματικής τιμής). Η όγδοη στήλη προσαρμόζει το ποσοστό (score) του επιτύχαμε για τους δείκτες με βάση το βάρος που έχει κάθε δείκτης.

Τα νούμερα του πίνακα είναι υποθετικά. Στην πραγματικότητα για να υπολογίσουμε τις τιμές των δεικτών λαμβάνουμε υπ' όψιν μας κάποιους άλλους «υπό-δείκτες» και χρησιμοποιούμε εργαλεία όπως στατιστικά στοιχεία, ερωτηματολόγια, έρευνες κ.α.

Για παράδειγμα, ας αναλύσουμε έναν από τους πιο «αφηρημένους» δείκτες του πίνακά μας, «Συνεχής βελτίωση - εκπαίδευσης πρακτόρων». Πως ποσοτικοποιούμε κάτι τόσο θεωρητικό όσο η πρόοδος των υπαλλήλων; Και εδώ ο μάνατζερ καλείται να επιλέξει ποια είναι εκείνα τα στοιχεία (δείκτες) που θα μετρήσει ώστε να μπορεί να δώσει την πρόοδο των υπαλλήλων με μαθηματικό τρόπο. Κάποιος θα μπορούσε να μετρήσει την πρόοδο α) κάνοντας έρευνα με συχνή χρήση τεστ και ερωτηματολογίων να χρησιμοποιήσει το «βαθμό επιτυχίας» β) να αναλύσει τα προβλήματα που ένας υπάλληλος προωθεί σε 2^ο επίπεδο επίλυσης. Συνεχίζει ο υπάλληλος να προωθεί επαναλαμβανόμενα δελτία με το ίδιο κατηγορία βλάβης; Μήπως χρειάζεται βελτίωση σε μία συγκεκριμένη κατηγορία; Με αυτό τον τρόπο ανακαλύπτουμε επίσης τα σημεία που θέλουν βελτίωση.

Αρα μπορούμε να πούμε ότι ο δείκτης «Συνεχής βελτίωση - εκπαίδευσης πρακτόρων» έχει δύο υπό-δείκτες α) Βαθμολογία ερωτηματολογίου και β) Ποσοστό προωθούμενων ανοιχτών δελτίων με ίδια κατηγορία. Αναλυτικότερα δηλαδή ο παραπάνω πίνακας θα έπρεπε να εμπλουτιστεί με τους δύο αυτούς υπό-δείκτες από τις τιμές των οποίων θα έβγαινε η πραγματική τιμή του δείκτη.

Με αυτόν τον τρόπο εργαζόμαστε για το σύνολο των οπτικών. Μέχρι τώρα πρέπει να έχει γίνει προφανής η χρησιμότητα της εφαρμογής helpdesk για τη εργασία των μάνατζερ. Και πράγματι, τα εργαλεία help desk βοηθούν εξαιρετικά σε αυτή την κατεύθυνση. Στην ενότητα που ακολουθεί θα δούμε περιγραφικά τη δική μας εφαρμογή helpdesk από την πλευρά του μάνατζμεντ.

5.2.3 Η εφαρμογή μας. Ένα εργαλείο μάνατζμεντ

Λαμβάνοντας υπ' όψιν μας όλα τα παραπάνω, ας προσπαθήσουμε να δούμε τη συνεισφορά της εφαρμογής μας στην εργασία των μάνατζερ, τον τρόπο που η εφαρμογή μας βοηθά στην επίτευξη των στρατηγικών στόχων και του οράματος της εταιρείας.

Πρώτα από όλα πρέπει να πούμε ότι η ίδια η εφαρμογή μας, ως προϊόν, αποτελεί μία επένδυση στην τεχνολογική υποδομή μίας εταιρείας και εφόσον καλύπτει τις απαιτήσεις της (όπως τις είδαμε στην παράγραφο 2.2), αποτελεί ένα προϊόν το οποίο επιτυγχάνει τους στόχους του αλλά και τους πιθανούς στόχους της εταιρείας για τεχνολογική εξέλιξη, ενσωμάτωση νέας τεχνολογίας, ανάπτυξη υποδομών κ.α.

Το δυνατό σημείο όμως της εφαρμογής για τους μάνατζερ είναι τα «ΣΤΑΤΙΣΤΙΚΑ» που προσφέρουν μετρήσεις για διάφορους δείκτες. Στον πίνακα που ακολουθεί παρουσιάζονται τα «ΣΤΑΤΙΣΤΙΚΑ» της εφαρμογής μας καθώς και οι δείκτες που προσφέρουν αλλά και πιθανούς στόχους που καλύπτουν.

Πίνακας 6 Στατιστικά και δείκτες

ΣΤΑΤΙΣΤΙΚΑ	Περιγραφή	Πιθανοί Δείκτες	Πιθανοί Στόχοι
Κίνηση δελτίων	Δείχνει πόσα δελτία άνοιξαν και έκλεισαν σε ένα συγκεκριμένο χρονικό διάστημα	Αναλογία ανοιχτών/κλειστών δελτίων Ποσοστό εξυπηρέτησης	Βελτίωση απόδοσης προσωπικού Βελτίωση εξυπηρέτησης πελάτη
Ανοιχτά δελτία	Δείχνει πόσα δελτία είναι ανοιχτά για κάθε εταιρεία/ τμήμα αλλά και το μέσο χρόνο που έχουν μείνει τα δελτία ανοιχτά	Κόστος ανά δελτίο Μέσος χρόνος χειρισμού Αναλογία ανοιχτών/κλειστών δελτίων	Βελτίωση απόδοσης προσωπικού Κοστολόγηση Βελτίωση εξυπηρέτησης πελάτη
Στατιστικά χρηστών	Δείχνει για κάθε χρήστη της εφαρμογής το πλήθος των δελτίων που άνοιξε, έκλεισε και ενημέρωσε (ενέργειες) σε ένα συγκεκριμένο χρονικό διάστημα	Κόστος ανά δελτίο Αναλογία ανοιχτών/κλειστών δελτίων Επίπεδο υπηρεσιών Ευσυνειδησία υπαλλήλων	Βελτίωση απόδοσης προσωπικού Κοστολόγηση Βελτίωση εξυπηρέτησης πελάτη Αξιολόγηση προσωπικού
Δελτία ανά κατηγορία	Δείχνει πόσα δελτία άνοιξαν, έκλεισαν αλλά και ενημερώθηκαν σε μία εταιρεία/ τμήμα σε μία κατηγορία σε ένα συγκεκριμένο χρονικό διάστημα	Συχνότερη Κατηγορία βλαβών Μέσος χρόνος χειρισμού Επίπεδο υπηρεσιών	Προσδιορισμός αναγκών πελατών Βελτίωση δεξιοτήτων προσωπικού Βελτίωση αξίας/ ποιότητας προϊόντων/ υπηρεσιών Βελτίωση εξυπηρέτησης πελάτη
Χρόνοι διεκπεραίωσης	Δείχνει το μέσο όρο, τον ελάχιστο και το μέγιστο χρόνο των δελτίων μίας εταιρείας/ τμήματος σε ένα συγκεκριμένο χρονικό διάστημα	Μέσος χρόνος χειρισμού Ενεργός χρόνος πρακτόρων Ευσυνειδησία υπαλλήλων	Βελτίωση απόδοσης προσωπικού Αξιολόγηση προσωπικού Βελτίωση εξυπηρέτησης πελάτη

Ένας μάνατζερ μπορεί να επιλέξει στο δικό του περιβάλλον κάποιον από τους παραπάνω συνδυασμούς ή ακόμα να βρει νέους δείκτες που να ικανοποιούν τους δικούς του στόχους.

Τα στατιστικά στοιχεία όμως δεν είναι τα μόνα από τα οποία βγαίνουν συμπεράσματα – δείκτες. Ένα βασικό χαρακτηριστικό των εφαρμογών help desk είναι ότι λειτουργούν ως γνωσιακές βάσεις δεδομένων. Οι υπάλληλοι μπορούν να ανατρέχουν σε παλαιότερα δελτία για την επίλυση νέων παρόμοιων περιστατικών. Καλύπτουν λοιπόν στόχους όπως αυτούς της «συνεχής εκπαίδευσης» και της «βελτίωσης δεξιοτήτων προσωπικού».

Ένα ακόμα χαρακτηριστικό της εφαρμογής που μπορεί να βοηθήσει στην εργασία του μάνατζερ είναι το εργαλείο της αναζήτησης. Με τη χρήση της αναζήτησης μπορείς εύκολα να ταξινομήσεις και να καταμετρήσεις τα δελτία με κάποιον τρόπο που σε εξυπηρετεί, και να βρεις τιμές για αρκετούς από τους παραπάνω δείκτες.

Για παράδειγμα, αν μετά από συμφωνία των υπαλλήλων, σε κάθε δελτίο που εξυπηρετούν με την πρώτη αναγράφουν ένα λεκτικό «FCR» και κάνουν αναζήτηση στα δελτία της ημέρας για το συγκεκριμένο λεκτικό, μπορούν να βρουν πόσα και ποια δελτία εξυπηρετήθηκαν με την πρώτη (δείκτης Ποσοστό λύσης στην πρώτη κλήση). Ή αν στο τίτλος κάθε δελτίου που ανοίγεται μετά από αίτημα μέσω e-mail αναγράφεται το λεκτικό «Mail», τότε με την αναζήτηση στο τέλος της μέρα μπορεί κάποιος να γνωρίζει πόσα και ποια δελτία ήταν αιτήματα μέσω e-mail. Κάτι τέτοιο θα ήταν εξαιρετικά χρήσιμο σε μια εταιρεία που θα ήθελε να μειώσει τα λειτουργικά της κόστη (π.χ. τηλεπικοινωνιακά).

Ακόμα και κάποια από τα χαρακτηριστικά που προέκυψαν από την κάλυψη των απαιτήσεων θα μπορούσαν να είναι στοιχεία της κάλυψης των στρατηγικών στόχων. Για παράδειγμα, η απαίτηση της Διεθνοποίησης. Καλύπτοντας την ανάγκη αυτή φέρνουμε την εταιρεία πιο κοντά σε στόχους που είδαμε ήδη όπως, της «Βελτίωσης της επικοινωνίας με του πελάτες», της «Βελτίωσης συνθηκών εργασίας» αλλά και πιθανούς άλλους στόχους όπως «Επέκταση της επιχείρησης στο εξωτερικό» ή «Εξυπηρέτηση ξενόγλωσσων εταιρειών/πελατών»

5.3 Σύνοψη μάνατζμεντ

Δώσαμε μία περιγραφή του balanced scorecard και μία κατεύθυνση του τρόπου που μπορεί να εργαστεί ένας μάνατζερ σε ένα τμήμα αρωγής. Ανάλογα με το μοναδικό περιβάλλον στο οποίο κινούμαστε μπορούμε να επιλέξουμε αρκετούς από τους στόχους και τους δείκτες που παρουσιάσαμε, αλλά και να ανακαλύψουμε νέους. Μπορούμε ακόμα να βρούμε νέους τρόπους λειτουργίας της εφαρμογής (π.χ. αναζήτηση μέσω των λεκτικών) ή να χρησιμοποιήσουμε τα ΣΤΑΤΙΣΤΙΚΑ για να δώσουμε τιμές σε διαφορετικούς δείκτες.

Η πίστη μας στην ανθρώπινη σκέψη και εφευρετικότητα δε μας επιτρέπει να δώσουμε τέλος στις δυνατότητες που μπορεί να έχει η εφαρμογή μας (και κάθε εργαλείο) στα χέρια ενός ανθρώπου. Είμαστε βέβαιοι ότι στα χέρια ενός έμπειρου μάνατζερ τα παραπάνω θα μπορούσαν να καλύπτουν περισσότερες σελίδες με ιδέες.

Εμείς σε όλη την έκταση της εργασίας μας προσπαθήσαμε να σας προσφέραμε τον τρόπο με τον οποίο σκεφτήκαμε και τον τρόπο που θα έπρεπε να αντιμετωπίζει κάποιος ένα έργο, από τη δική του πλευρά. Ελπίζουμε ολοκληρώνοντας και το κομμάτι της οπτικής του μάνατζμεντ να σας έχουμε πείσει ότι και από αυτή την πλευρά το έργο μας έχει πολλά να προσφέρει.

Συμπεράσματα - Περίληψη

Τα συμπεράσματά μας για την ανάπτυξη της παρούσας εργασίας είναι θετικά και περιγράφηκαν αναλυτικά στις τελευταίες ενότητες των κεφαλαίων. Στις ενότητες αυτές αναφερθήκαμε στην αξία του είδους της εφαρμογής που επιλέξαμε, στην αξία των τεχνολογιών που χρησιμοποιήσαμε και την τελική αξία που είδαμε στην εφαρμογή μας, τόσο από την πλευρά του ειδικού της πληροφορικής όσο και από την πλευρά του ειδικού της διοίκησης επιχειρήσεων.

Αυτό που θέλουμε να φανεί κυρίως στον επίλογό μας είναι η εμπειρία που αποκομίσθηκε κατά την εκπόνηση της εργασίας αυτής.

Διαπιστώθηκε ότι για την ανάπτυξη μιας δικτυακής εφαρμογής ο βαθμός πολυπλοκότητας είναι μεγάλος. Απαιτήθηκε από εμάς να βρεθούμε σε διαφορετικές θέσεις και να αντιμετωπίσουμε το έργο μας από πολλές οπτικές.

Εκτός από την πολύ καλή μελέτη του ίδιου του αντικείμενου, για το οποίο βοήθησε προηγούμενη εμπειρία, έπρεπε να κάνουμε ανάλυση περιβάλλοντος ως αναλυτές πληροφοριακών συστημάτων, ανάλυση εφαρμογής ως τεχνολόγοι λογισμικού, ανάπτυξη ως προγραμματιστές και αξιολόγηση ως μάντζερ.

Το γεγονός αυτό δεν μας κάνει εντύπωση. Όλα τα έργα χρειάζονται πολλά ταλέντα για να ολοκληρωθούν και η ανάπτυξη λογισμικού θα πρέπει για να έχει συνέχεια να αντιμετωπίζεται ως μεγάλο έργο. Η ενασχόληση μας με όλα τα παραπάνω αντικείμενα μας προσέφερε εμπειρία και αυτοπεποίθηση σε πεδία που μας ήταν μέχρι πρότινος άγνωστα.

Η πορεία της εργασίας μας είχε ως εξής: Σε πρώτο στάδιο έγινε ανάλυση περιβάλλοντος, όπου διακρίναμε τη χρησιμότητα τους και την αξία των εφαρμογών helpdesk. Ας υπενθυμίσουμε: Με τις εφαρμογές Helpdesk διευκολύνεται η εργασία μεταξύ των τμημάτων της εταιρείας ή μεταξύ εταιρειών και μειώνεται σημαντικά ο χρόνος διεκπεραίωσης εργασιών, άλλα σημαντικότερο πλεονέκτημα τους αποτελεί η μακροπρόθεσμη αύξηση της παραγωγικότητας και της αποδοτικότητας της εταιρείας με την χρήση των εφαρμογών αυτών.

Ακολούθησε η ανάλυση της εφαρμογής με την ανάλυση απαιτήσεων και έγινε προσεκτικός σχεδιασμός όλων των συμπεριλαμβανόμενων διαδικασιών της προς υλοποίησης εφαρμογής. Με τον τρόπο αυτό καθορίστηκαν οι ανάγκες των χρηστών στους οποίους απευθύνεται η εφαρμογή – target group και οι απαιτήσεις του συγκεκριμένου είδους της εφαρμογής ώστε να δημιουργηθεί μια εφαρμογή εύχρηστη, λειτουργική και που να εξυπηρετεί πλήρως το target group της. Στο σύνολο της εφαρμογής απαιτείται προσεκτικός σχεδιασμός ώστε όλες οι διαδικασίες και λειτουργίες της εφαρμογής να μπορούν να εκτελεστούν με ευκολία από τους χρήστες και σε πολλές περιπτώσεις να μπορούν να τους καθοδηγούν.

Στη συνέχεια έγινε μελέτη για τις τεχνολογίες ανάπτυξης δικτυακών εφαρμογών και η υλοποίηση της εφαρμογής, η οποία βασίστηκε στις μελέτες της ανάλυσης ώστε να εξασφαλιστεί η βιωσιμότητα της.

Οι τεχνολογίες που ελέγχθησαν ήταν οι JSP/Servlet και Struts, και αποδείχθηκαν σχετικά εύκολες στην υιοθέτηση τους αλλά και πολύ χρήσιμες για την κατασκευή εφαρμογών με δυναμικό περιεχόμενο κάτι που αντιστοιχεί στην δική μας περίπτωση. Επιπλέον η υιοθέτηση της αρχιτεκτονικής JSP / Servlets Model View Controller μέσω Struts για το σύστημα μας παρέχει αρκετά οφέλη σχετικά με την ευελιξία στην ανάπτυξη και στη συντήρηση του συστήματος και τον ευκολότερο έλεγχο των επιμέρους υποσυστημάτων.

Η εργασία μας ολοκληρώθηκε μελετώντας την εφαρμογή με εργαλεία του μάντζερ. Η εφαρμογή μας σε μία εταιρεία που χρησιμοποιεί τη μέθοδο Balanced Scorecard μπορεί να δώσει σημαντικούς δείκτες που θα αξιολογούν την απόδοση και θα καλύπτουν στρατηγικούς στόχους.

Σε μια δεύτερη προσέγγιση η εφαρμογή μας θα μπορούσε να αναπτυχθεί περαιτέρω για να ενσωματώσει επιπλέον εργαλεία όπως ημερολόγιο, σημειωματάριο κ.α. και θα μπορούσε να ενσωματωθεί σε μεγαλύτερα προγράμματα όπως αυτά της διαχείρισης ανταλλακτικών – αποθήκης, διακίνησης εγγράφων κ.α. Πολύ εύκολα ακόμα το παρόν πρόγραμμα μπορεί να μεταφραστεί και σε άλλες γλώσσες, (πέραν της αγγλικής που συμπεριλαμβάνουμε στα παραδοτέα), λόγω των δυνατοτήτων του Struts. Ακόμα, με βάση το δεύτερο μέρος της

εργασίας μας, ένας ειδικός της διοίκησης επιχειρήσεων θα μπορούσε να κάνει μια πλήρη μελέτη για το σύνολο μιας εταιρείας με τμήμα τηλεφωνικής εξυπηρέτησης ή τμήμα αρωγής.

Τέλος, μία μελλοντική μελέτη που πιστεύουμε ότι θα είχε εξαιρετικό ενδιαφέρον, για τους τομείς της μοντελοποίησης χρηστών, της εκπαίδευσης και της ασφάλειας είναι η διερεύνηση της δυνατότητας εξαγωγής συμπερασμάτων από την εφαρμογή για τη συμπεριφορά των χρηστών προς όφελος των παραπάνω τομέων. Με άλλα λόγια, θα πρέπει να μελετηθεί αν κατά πόσο η μελέτη της κίνησης των δελτίων μπορεί να βγάλει συμπεράσματα για πιθανές ανάγκες για την εκπαίδευση των πελατών/τεχνικών ή για πιθανά ρήγματα ασφάλειας, και σε συνέχεια να ομαδοποιηθούν οι κατηγορίες βλαβών που μπορούν να «προδώσουν» αδυναμίες των χρηστών, της ασφάλειας κ.α. Θεωρούμε ότι μία εφαρμογή helpdesk μπορεί να είναι ένα καλό υποστηρικτικό σύστημα για λήψη αποφάσεων (όπως είδαμε και στο μανατζμεντ) και για άλλους τομείς.

Παράρτημα Α- Εγχειρίδια Εφαρμογής

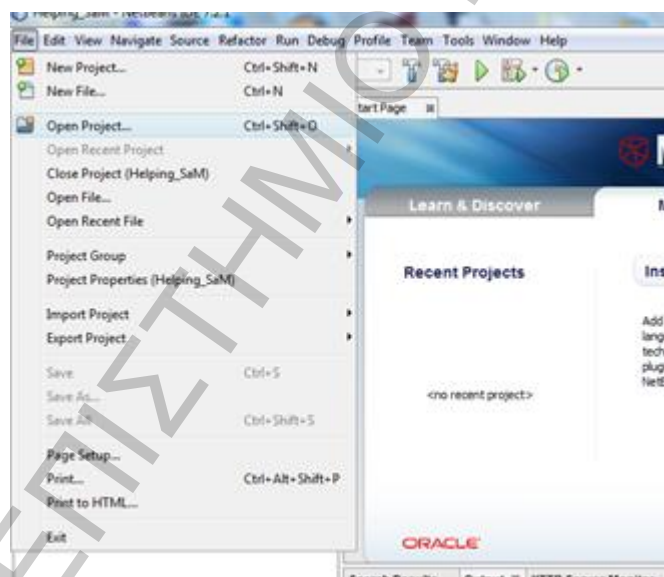
Εγχειρίδιο Προγραμματιστή

Εγκατάσταση και Λειτουργία

Για την δημιουργία της εφαρμογής, χρησιμοποιήθηκε το περιβάλλον ανάπτυξης NetBeans7.2.1³⁰ και για την υποστήριξη της βάσης δεδομένων MySQL6.0³¹. Βοηθητικά χρησιμοποιήθηκε το πρόγραμμα Xampp που λειτουργεί ως Apache και MySQL server για λειτουργικά Windows. Για την εγκατάσταση των προγραμμάτων αυτών υπάρχει επαρκής τεκμηρίωση στις ιστοσελίδες που παρέχονται.

Για να λειτουργήσει η εφαρμογή μας απαιτείται υποστήριξη από Web Container (όπως για παράδειγμα ο Apache Tomcat), MySQL Server, JRE και η χρήση φυλλομετρητή (Browser).

Για να τροποποιηθεί η εφαρμογή από κάποιο προγραμματιστή, πρέπει να ανοίξουν τα αρχεία της εφαρμογής (συμπεριλαμβάνεται στα παραδοτέα - φάκελος HelpdeskAssistant) ως project του NetBeans (Open project) και στις επιλογές του project να δοθούν οι επιλογές που φαίνονται στην παρακάτω εικόνα (Categories → Java Web και Projects → Web Application with Existing Sources) – (Εικόνα 54). Στο επόμενο παράθυρο επιλέγεται ο φάκελος της εφαρμογής και η διαδικασία ολοκληρώνεται δηλώνοντας τον web container (στην περίπτωση μας tomcat) και δίνοντας τις επιθυμητές πληροφορίες για όνομα site κ.α.



Εικόνα 54 Επιλογές στο NetBeans.

Επιπλέον, πρέπει να έχει προστεθεί στον MySQL Server η βάση δεδομένων της εφαρμογής. Στα παραδοτέα της εφαρμογής παραδίδεται dump σε sql με όνομα hddb.sql για τη δημιουργία μίας βάσης δεδομένων με προκαθορισμένο παράδειγμα επίδειξης της εφαρμογής. Η προσθήκη γίνεται απλά από το phpmyadmin (xampp). Εκεί απλά κάνουμε εισαγωγή του αρχείου .sql (Εικόνα 56).

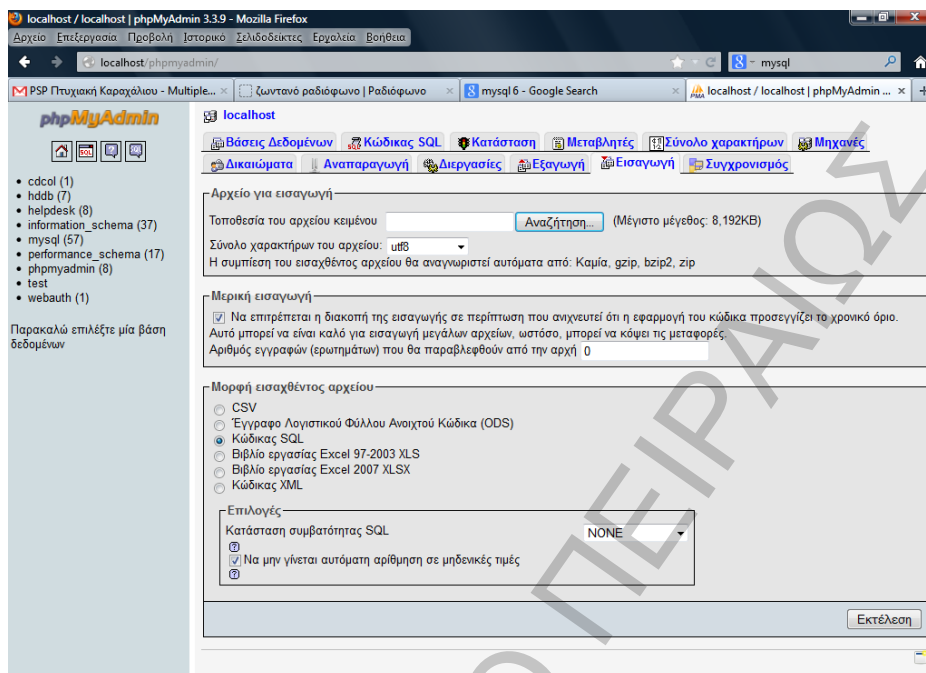
Το script της Βάσης Δεδομένων, συμπεριλαμβάνει και την δημιουργία ενός χρήστη της MySQL με όνομα 'alexk' και κωδικό πρόσβασης 'hddb'. Με τον συγκεκριμένο λογαριασμό και μέσω των servlets συνδέεται η εφαρμογή στην MySQL Βάση Δεδομένων.

Για να υποστηρίξει η Βάση Δεδομένων μας ελληνικούς χαρακτήρες, χρειάστηκε να τροποποιήσουμε το αρχείο 'my.ini', που συμπεριλαμβάνει τις ρυθμίσεις της MySQL, και να

³⁰ Επίσημος Ιστότοπος: www.netbeans.org

³¹ Επίσημος Ιστότοπος: www.mysql.com

εισάγουμε την τιμή 'utf8' στην παράμετρο default-character-set και στην παράμετρο character-set-server. Την κωδικοποίηση utf-8 χρησιμοποιήσαμε και κατά την δημιουργία των πινάκων της Βάσης Δεδομένων.

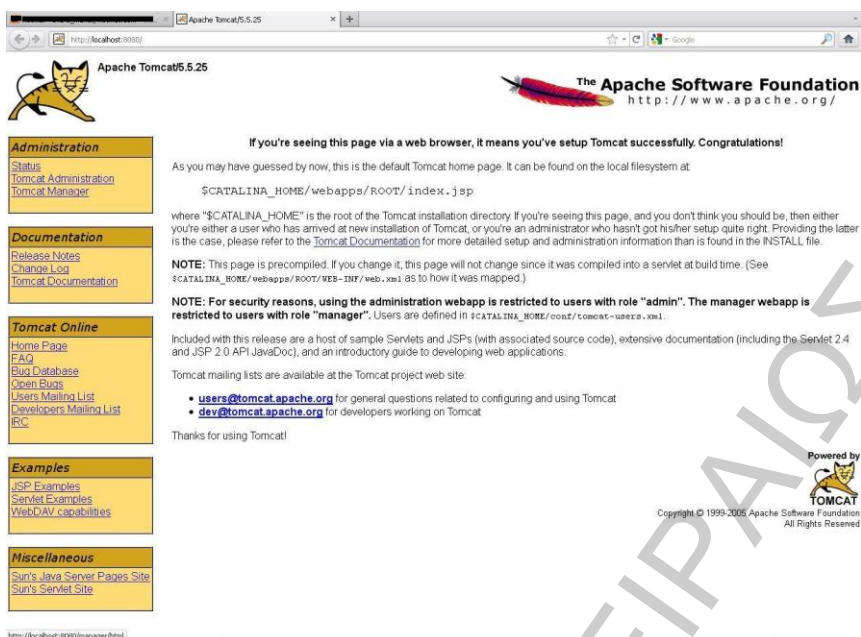


Εικόνα 55 PhpMyAdmin

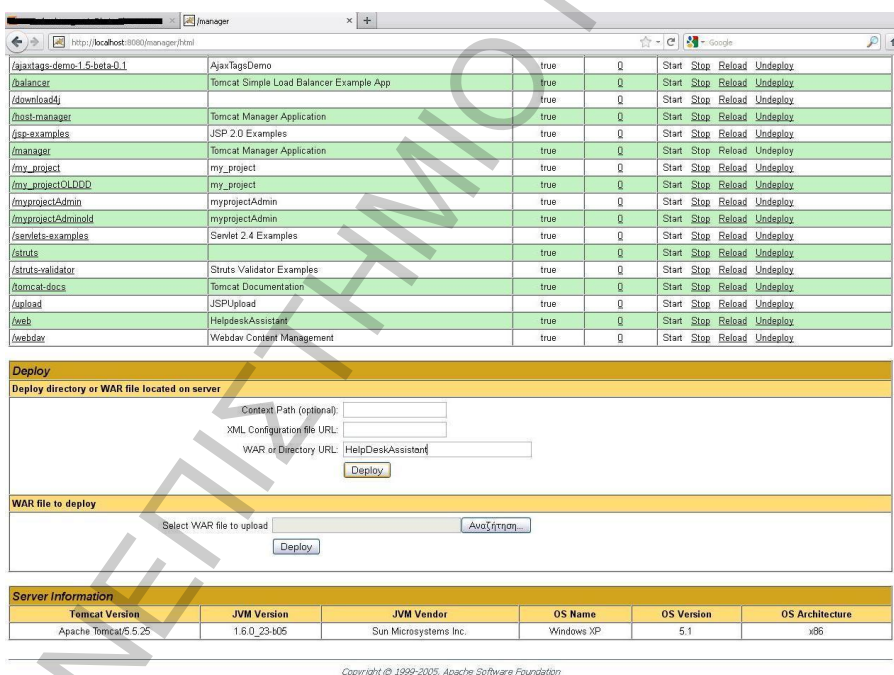
Αφού προστεθεί και η βάση δεδομένων, το project είναι έτοιμο για να τρέξει. Αυτό το κάνουμε από NetBeans με την επιλογή Run Main Project (F6).

Τέλος, πέραν από το προγραμματιστικό περιβάλλον, για να εγκαταστήσουμε την εφαρμογή μας σε ένα live σύστημα, θα πρέπει ο server που θα επιλέξουμε να έχει οπωσδήποτε εγκατεστημένο ένα container (όπως ο Tomcat) αλλά και να υποστηρίζει MySQL Βάση Δεδομένων (MySQL Server). Σε αυτήν την περίπτωση, αρκεί να αντιγράψουμε τον φάκελο 'web' (βρίσκεται μέσα τον φάκελο build και έχει δημιουργηθεί από το project) στον φάκελο 'webapps' του Tomcat, και αφού τον μετονομάσουμε (για παράδειγμα σε HelpDeskAssistant) να το κάνουμε 'deploy' με τον τρόπο που φαίνεται στις παρακάτω εικόνες.

Η εφαρμογή μας ανοίγει με τη χρήση φυλλομετρητή. Πληκτρολογούμε τη διεύθυνση του server και το όνομα του project που έχουμε φορτώσει (π.χ. localhost:8084/Helping_SaM)



Εικόνα 56 Σελίδα Διαχείρισης Tomcat – Επιλογή Tomcat Manager



Εικόνα 57 Tomcat Manager – deploy τον φάκελο HelpDeskAssistant που βρίσκεται στον φάκελο webapps του Tomcat.

Η δομή του προγράμματος

Μετά το άνοιγμα της εφαρμογής στο Netbeans IDE στο αριστερό μέρος του παραθύρου αποκαλύπτεται η δομή.

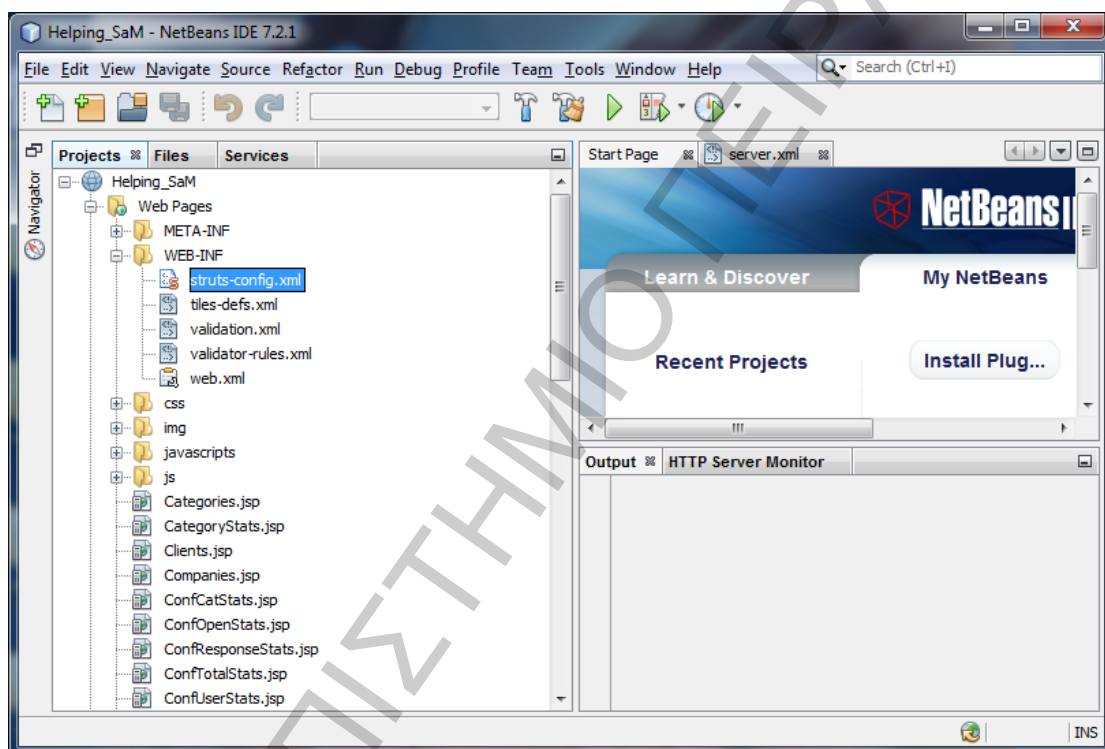
Στο φάκελο web pages περιέχονται οι JSP ιστοσελίδες μας καθώς και οι υποφάκελοι web-inf, javascripts και js.

Μέσα στον υποφάκελο web-inf εντοπίζουμε το αρχείο struts-config.xml το οποίο αποτελεί το βασικό αρχείο παραμετροποίησης μιας εφαρμογής Struts. Μέσα στο αρχείο αυτό αρχικά δηλώνονται τα ActionForm Beans που θα χρησιμοποιηθούν στα πλαίσια ενός session. Στη

συνέχεια υπάρχουν τα `action-mappings` τα οποία συσχετίζουν κάθε `struts action` με το `actionform bean` που θα χρησιμοποιηθεί για να του μεταφέρει τα δεδομένα και με το `web path` που αντιστοιχεί σε αυτό το `action` και στη συνέχεια προσδιορίζουν τα `forwards`. Τα `forwards` προσδιορίζουν τη σελίδα στην οποία θα οδηγηθούμε μετά την εκτέλεση του `action` ανάλογα με το αποτέλεσμα της επεξεργασίας που αυτό περιέχει.

Στον υποφάκελο `javascripts` υπάρχουν κάποια `java scripts` τα οποία περιέχουν λειτουργίες που χρησιμοποιούνται μέσα στις σελίδες `jsp`. Κάποιες από αυτές τις λειτουργίες έχουν να κάνουν με τη μορφοποίηση εμφάνισης κάποιων πινάκων και κάποιες άλλες έχουν να κάνουν με την αλλαγή ενός τμήματος μιας σελίδας ως αποτέλεσμα μίας επιλογής που γίνεται σε κάποιο άλλο μέρος της σελίδας (π.χ. αλλάζει σε μία λίστα επιλογών η επιλεγμένη εταιρεία και ως αποτέλεσμα ενημερώνεται η λίστα επιλογής τμημάτων που βρίσκεται από κάτω).

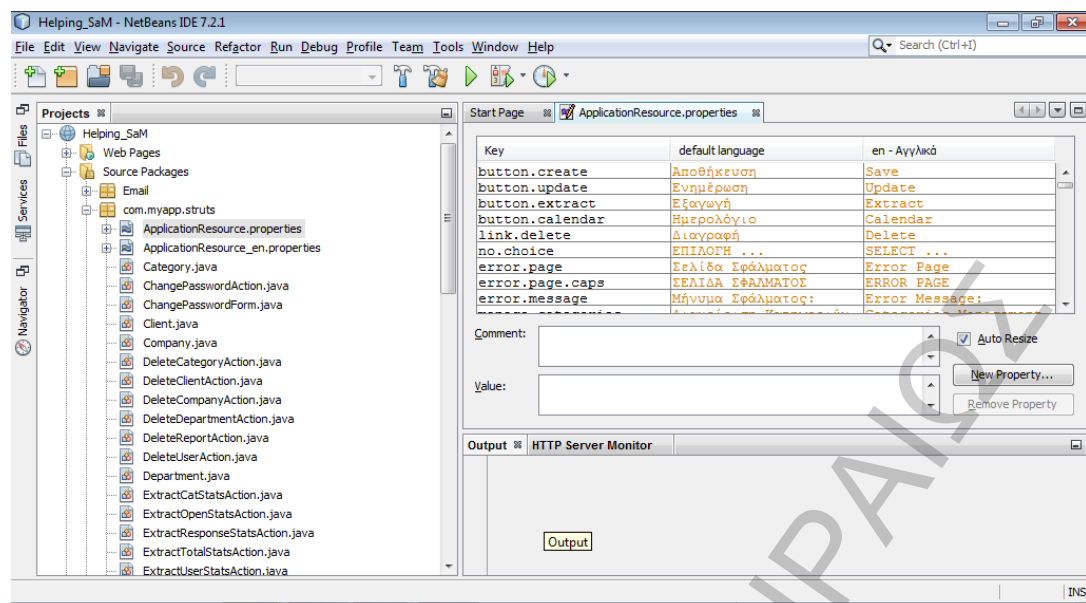
Στον υποφάκελο `js` υπάρχουν επίσης κάποια `java scripts` τα οποία προέρχονται από ένα `package` που χρησιμοποιείται για το χειρισμό πινάκων (ταξινόμηση, αναζήτηση μέσα σε πίνακα κτλ).



Εικόνα 58 Δομή Web pages

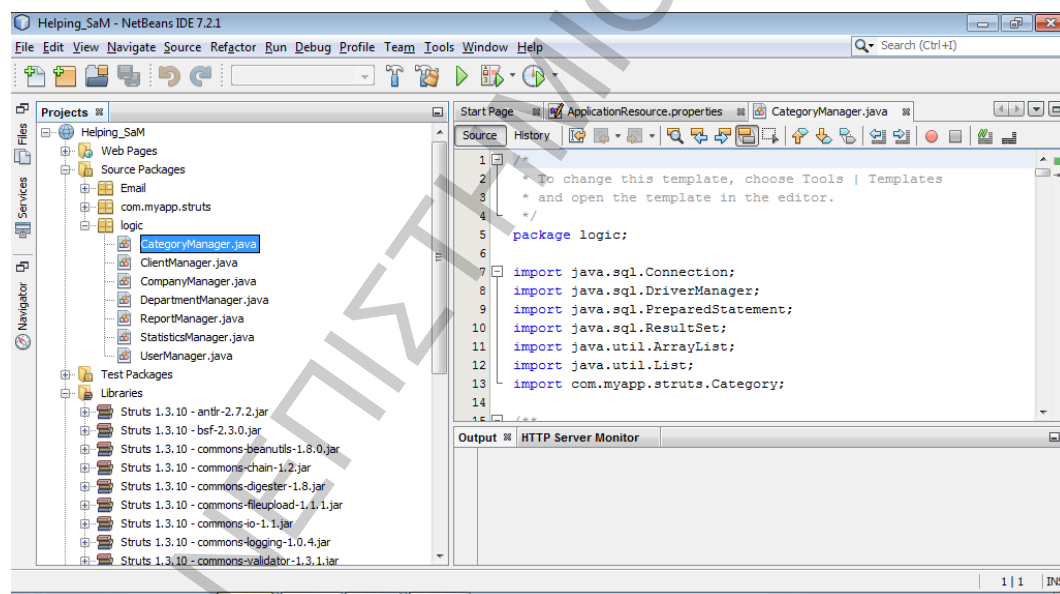
Ο φάκελος `Source Packages` περιέχει όλες τις κλάσεις `java` χωρισμένες σε τρεις υποφακέλους. Στον υποφάκελο `mail` υπάρχει μία κλάση `java` που παρέχει στην εφαρμογή της τη δυνατότητα αποστολής `mail`.

Στον υποφάκελο `com.myapp.struts` βρίσκονται όλες οι `java` κλάσεις του πλαισίου `struts`. Κλάσεις για τα `struts actionform beans` (`category.java`, `client.java`, `company.java`, `Department.java`, `Report.java`, `StatisticsForm.java`, `User.java` και το `TicketUpdate.java` το οποίο ορίζεται ως κλάση και χρησιμοποιείται ως `attribute` της κλάσης `report`) και κλάσεις για τα `struts actions`. Τέλος στον υποφάκελο αυτό υπάρχουν και τα αρχεία `ApplicationResource.properties` `ApplicationResource_en.properties` μέσα στα οποία υπάρχουν οι αντιστοιχισίες κλειδιών-μηνυμάτων στα ελληνικά και στα αγγλικά αντίστοιχα.



Εικόνα 59 Δομή Source packages – αρχείο ApplicationResource.properties

Τέλος στον υποφάκελο logic υπάρχουν κλάσεις οι μέθοδοι των οποίων διεκπεραιώνουν τις λειτουργίες που γίνονται πάνω στη βάση δεδομένων μας (σύνδεση, αποσύνδεση, εκτέλεση ερωτημάτων, εισαγωγή εγγραφών, διαγραφή εγγραφών, ενημέρωση εγγραφών) και καλούνται για το σκοπό αυτό από τα διάφορα struts actions.



Εικόνα 60 Δομή logic

Για περισσότερες πληροφορίες ανατρέξτε στον ίδιο τον κώδικα όπου δίνεται σχετική τεκμηρίωση.

Εγχειρίδιο Χρήστη

Ως εγχειρίδιο χρήστη (user manual) δίνεται η ολοκληρωμένη παρουσίαση της εφαρμογής που παρατίθεται στο Κεφάλαιο 3.

Υπενθυμίζουμε:

Η εφαρμογή μας ανοίγει με τη χρήση φυλλομετρητή. Πληκτρολογούμε τη διεύθυνση του server και το όνομα του φακέλου που έχουμε αποθηκεύσει την εφαρμογή (π.χ. localhost:8084/Helping_SaM)

Ανάλογα με την επιλεγμένη γλώσσα του φυλλομετρητή η εφαρμογή μας προσφέρεται στην Ελληνική ή την Αγγλική γλώσσα. Προεπιλεγμένη γλώσσα είναι η Ελληνική.

Παράρτημα Β - Βοηθητικά εργαλεία

Στην ενότητα αυτή παρουσιάζεται όλο το θεωρητικό υπόβαθρο για τα βοηθητικά εργαλεία που χρησιμοποιήσαμε κατά την ανάλυση και το σχεδιασμό. Η αναφορά αυτή είναι ιδιαίτερα χρήσιμη για να μπορεί οποιοσδήποτε να αναγνώσει με ευκολία τα επόμενα κεφάλαια, χωρίς να βρεθεί αντιμέτωπος με άγνωστους συμβολισμούς ή τεχνικές.

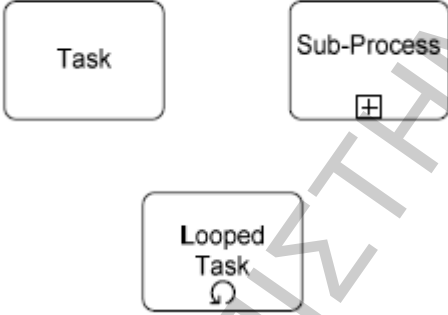
Γραφική Ανάλυση με BPMN

Επιλέξαμε να καταγράψουμε τις επιχειρησιακές διαδικασίες που αφορούν την εργασία μας με τη βοήθεια του δωρεάν λογισμικού Intalio³² και τη γλώσσα BPMN (Business Process Modeling Notation – Σημειογραφία Μοντελοποίησης Επιχειρησιακών Διαδικασιών). Για τη καλύτερη κατανόηση των γραφημάτων κρίνεται σκόπιμο να δώσουμε μία μικρή αναφορά για τη γλώσσα BPMN και το εργαλείο Intalio.

Η γλώσσα προγραμματισμού BPMN είναι ένας συμβολισμός επιχειρησιακών διαδικασιών που βασίζεται σε διαγράμματα ροών. Τα στοιχεία των διαγραμμάτων της BPMN είναι οι δραστηριότητες (activities), τα γεγονότα (events), οι πύλες (gateways) και οι σύνδεσμοι (connectors).

Στους πίνακες που ακολουθούν παρουσιάζονται τα βασικά σχήματα που χρησιμοποιούνται στην BPMN.






















Πίνακας 7 Δραστηριότητες (activities) στην BPMN.

Δραστηριότητες ³³	
	<p>Μία δραστηριότητα είναι μία εργασία που εκτελείται κατά τη επιχειρησιακή διεργασία.</p>






³² Διαθέσιμο online από τη σελίδα <http://www.intalio.com/downloads>.

³³ Τα σχήματα στους πίνακες είναι από το [48].




Πίνακας 8 Γεγονότα (events) στην BPMN.

Γεγονότα			
Γεγονότα αρχής	Ενδιάμεσα γεγονότα	Γεγονότα τέλους	
None  Message  Timer  Rule  Link  Multiple 	None  Message  Timer  Error  Compensation  Rule  Link  Multiple 	None  Message  Error  Compensation  Link  Terminate  Multiple 	<p>Ένα γεγονός είναι κάτι που «συμβαίνει» κατά τη διάρκεια μιας διαδικασίας. Τα γεγονότα επηρεάζουν τη ροή εργασιών και συνήθως έχουν μια αιτία (πυροδοτητή - trigger) ή ένα αποτέλεσμα.</p> <p>Υπάρχουν 3 τύποι γεγονότων, βασισμένοι στο πότε επηρεάζουν τη ροή της εργασίας: Αρχή, Ενδιάμεσο και Τέλος. [48]</p>

Πίνακας 9 Πύλες (gateways) στην BPMN.

Πύλες	
Exclusive Data-Based  Event-Based  Inclusive  Complex  Parallel 	<p>Μια πύλη χρησιμοποιείται για να ελέγξει τη ροή εργασιών. Αποφασίζει για θέματα διακλαδώσεων: διαχωρισμό, συγχώνευση και ένωση μονοπατιών. Το Exclusive Data-Based μπορεί να συμβολίζεται επίσης και με ένα X στο κέντρο.</p>

Πίνακας 10 Σύνδεσμοι (connectors) στην BPMN.

Σύνδεσμοι	
Sequence Flow  Message Flow  Association 	<p>Ένας σύνδεσμος χρησιμοποιείται για να συσχετίζει πληροφορία (π.χ. Σχόλια) με Αντικείμενα Ροής (Flow Objects). Κείμενο και γραφικά αντικείμενα που δεν απεικονίζουν ροή μπορούν επίσης να συσχετιστούν με αντικείμενα ροής μέσω του συνδέσμου. [49]</p>

Για περισσότερες πληροφορίες για τη γλώσσα προγραμματισμού BPMN ανατρέξτε στα [48] και [49].

Γραφική Ανάλυση με UML

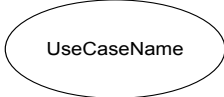
Επίσης για να καταγράψουμε τις διαδικασίες που θα ακολουθούνται στην εφαρμογή, στις οποίες καταλήξαμε κατά την ανάλυση απαιτήσεων, χρησιμοποιήσαμε την γλώσσα μοντελοποίησης λογισμικού UML (Unified Modeling Language) και το εργαλείο Microsoft Office Visio.

Η UML είναι μία οπτική αντικειμενοστραφής γλώσσα μοντελοποίησης που χρησιμοποιείται για τη γραφική απεικόνιση, προσδιορισμό, κατασκευή και τεκμηρίωση των στοιχείων ενός συστήματος λογισμικού. Μπορεί να χρησιμοποιηθεί σε διάφορες φάσεις ανάπτυξης, από την ανάλυση απαιτήσεων ως τον έλεγχο ενός ολοκληρωμένου συστήματος και αποτελείται από ένα σύνολο προσυμφωνημένων όρων, συμβόλων και διαγραμμάτων.[50]

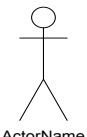



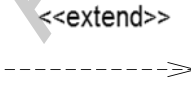
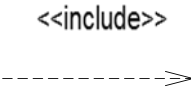
Συγκεκριμένα στην περίπτωση μας, επιλέξαμε να κατασκευάσουμε διαγράμματα περιπτώσεων χρήσης (use case diagrams), διαγράμματα δραστηριοτήτων (activity diagrams) καθώς και διαγράμματα ακολουθίας (sequences diagrams) ώστε να μοντελοποιήσουμε τις απαιτήσεις των χρηστών και της εφαρμογής, τα δυναμικά χαρακτηριστικά της εφαρμογής και τα βήματα εκτέλεσης κάποιων διαδικασιών.

Στους πίνακες³⁴ που ακολουθούν παρουσιάζονται τα γραφικά σύμβολα που αντιστοιχούν στα διαγράμματα που προαναφέρθηκαν καθώς και η επεξήγηση τους, ώστε να είναι κατανοητά και ευανάγνωστα τα διαγράμματα που παρατίθενται σε ακόλουθες ενότητες.






Πίνακας 11 Γραφικά Σύμβολα που χρησιμοποιούνται στα Διαγράμματα Περιπτώσεων Χρήσης


Όνομασία	Επεξήγηση	Γραφικό Σύμβολο
Περίπτωση Χρήσης (Use Case)	Περιγράφει ποια είναι η λειτουργία ενός συστήματος/ κλάσης. Είναι ένα σύνολο από δραστηριότητες που επιτελούνται από το σύστημα ώστε να παραχθεί κάποια συμπεριφορά που είναι αισθητή από κάποιο ενεργοποιό.	

³⁴ Οι πίνακες κατασκευάστηκαν βάση των [51] και [52].

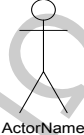
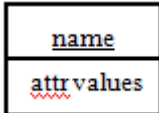




Ενεργοποιός (Actor)	Εξωτερική οντότητα του συστήματος που χρησιμοποιεί τη λειτουργικότητα του. Το σύμβολο αυτό αναφέρεται και σε εξωτερικά συστήματα, όχι μόνο σε φυσικά πρόσωπα. [51]	
Όριο του Συστήματος (System Boundary)	Ορίζει το όριο το οποίο διαχωρίζει το σύστημα από τους ενεργοποιούς που αλληλεπιδρούν με το σύστημα. [51]	
Συσχέτιση (Association)	Η συμμετοχή και ο τρόπος επικοινωνίας ενός ενεργοποιού με μία περίπτωση χρήσης. [51]	
Γενίκευση (Generalization)	Μια σχέση ταξινόμησης ανάμεσα σε μια γενική περίπτωση χρήσης και μια πιο ειδικευμένη περίπτωση χρήσης.	
Επέκταση (Extend)	Μια σχέση ανάμεσα σε περιπτώσεις χρήσης που ορίζει εναλλακτικούς τρόπους διαχείρισης της βασικής περίπτωσης χρήσης.	
Συνυπολογισμός (Include)	Μια σχέση ανάμεσα σε μια βασική και μια ή περισσότερες μερικές περιπτώσεις χρήσης που ορίζει ότι η βασική περίπτωση χρήσης απαιτεί τη λειτουργία των μερικών περιπτώσεων χρήσης. [51]	

Πίνακας 12 Γραφικά Σύμβολα που χρησιμοποιούνται στα Διαγράμματα Δραστηριοτήτων

Ονομασία	Επεξήγηση	Γραφικό Σύμβολο
Δραστηριότητα (Activity)	Ένας κόμβος δραστηριότητας ορίζει ένα βήμα σε μια διαδικασία. Έχει ένα σύνολο εισερχόμενων και εξερχόμενων ροών δραστηριοτήτων και δεν μπορεί να αναλυθεί περαιτέρω. [52]	
Αρχικός Κόμβος	Κόμβος από τον οποίο ξεκινά η ροή. Σημείο έναρξης. [52]	
Τελικός Κόμβος	Σημείο λήξης, είναι ένας κόμβος στον οποίο η ροή σε μια δραστηριότητα σταματά. [52]	
Ακμή	Ροή Εργασιών. Μετάβαση στην επόμενη δραστηριότητα, μετά την ολοκλήρωση της προηγούμενης δραστηριότητας. [52]	
Κόμβος Απόφασης	Ο κόμβος απόφασης περιέχει συνθήκες, όπου πραγματοποιείται έλεγχος αυτών και επιλέγεται ανάλογα με το αποτέλεσμα του ελέγχου μια εξερχόμενη ροή. Ο κόμβος αυτός, έχει μια εισερχόμενη ροή και πολλαπλές εξερχόμενες ροές δραστηριοτήτων. [52]	

Σημείο Συγχρονισμού	Τα σημεία συγχρονισμού ορίζουν σημεία διεπαφής παράλληλων ροών. Παράλληλη εκτέλεση δραστηριοτήτων. [52]	
---------------------	---	---

Πίνακας 13 Γραφικά Σύμβολα που χρησιμοποιούνται στα Διαγράμματα Ακολουθίας.

Όνομασία	Επεξήγηση	Γραφικό Σύμβολο
Ενεργοποιός (Actor)	Εξωτερική οντότητα του συστήματος που χρησιμοποιεί τη λειτουργικότητα του. Το σύμβολο αυτό αναφέρεται και σε εξωτερικά συστήματα, όχι μόνο σε φυσικά πρόσωπα. [51]	
Αντικείμενο (Object)	Μια οντότητα με μοναδικά ορισμένη υπόσταση. Αυτή η οντότητα μπορεί να λαμβάνει μηνύματα και μπορεί να βρίσκεται σε διαφορετικές καταστάσεις (states). [51]	
Μήνυμα (Message)	Ορίζει την επικοινωνία ανάμεσα σε δύο υλοποιήσεις. [52]	
Σύνδεσμος (Association)	Σύνδεση ανάμεσα σε υλοποιήσεις. [52]	
Γραμμή Ετοιμότητας (Lifeline)	Αντιστοιχεί σε κάποιο αντικείμενο ή ενεργοποιό, όπου δηλώνει αναμονή για ενεργοποίηση. [52]	
Μπάρα Ενεργοποίησης (Activation Bar)	Αντιστοιχεί στην διάρκεια ζωής ενός αντικειμένου ή ενεργοποιού. [52]	

Αναφορές

- [1] HelpDeskSoftware.org, «*A Comprehensive Guide to Help Desk Software Solutions*», (2003-2011), Διαθέσιμο σε <http://www.helpdesksoftware.org/> (προσπέλαση: Αύγουστος 2011).
- [2] Open Source Help Desk Software , «*A Complete List of Help Desk Software Products*», Διαθέσιμο σε <http://www.opensourcehelpdesklist.com> (προσπέλαση: Αύγουστος 2011).
- [3] ReadyDesk.com, (2011), Διαθέσιμο σε <http://www.readydesk.com/> (προσπέλαση: Αύγουστος 2011).
- [4] Παρατηρητήριο για τη Διοικητική Μεταρρύθμιση, (2012) «Έρευνα για την ανάπτυξη & διείσδυση υπηρεσιών ηλεκτρονικής εξυπηρέτησης Πολιτών & Επιχειρήσεων», Διαθέσιμο σε <http://www.observatory.gr/page/default.asp?la=1&id=2101&pk=445&return=183> (προσπέλαση Ιανουάριος 2013)
- [5] Παρατηρητήριο για τη ΚΤΠ (2011), «Έρευνα Οικονομικής Συγκυρίας στον τομέα Τεχνολογιών Πληροφορικής και Επικοινωνιών» Διαθέσιμο σε <http://www.observatory.gr/page/default.asp?la=1&id=2101&pk=421&return=183> (προσπέλαση Ιανουάριος 2013)
- [6] Εθνικό κέντρο τεκμηρίωσης, (2013) «Έκθεση του ΙΟΒΕ για την επιχειρηματικότητα στην Ελλάδα την περίοδο 2011-2012» Διαθέσιμο σε <http://www.ekt.gr/content/display?prnbr=86569> (προσπέλαση Μάρτιος 2013)
- [7] Eurostat, (2012), «High-tech statistics», Διαθέσιμο σε http://epp.eurostat.ec.europa.eu/statistics_explained/index.php/High-tech_statistics (προσπέλαση Ιανουάριος 2013)
- [8] Γιώργου Μανέππα (2012), «Αγορά: Στροφή των καταναλωτών στα ελληνικά προϊόντα» Ημερησία, διαθέσιμο σε <http://www.imerisia.gr/article.asp?catid=26519&subid=2&pubid=112860124> (προσπέλαση: Φεβρουάριος 2013)
- [9] Βαρβαρίγου Θ. (2010), «*JSPs – Servlets*», Εθνικό Μετσόβιο Πολυτεχνείο, διαθέσιμο σε https://ecourses.dbnet.ntua.gr/fsr/17202/6a_JSP-Servlets-1.rpt, (προσπέλαση: Αύγουστος 2011).
- [10] ΤΕΙ Αθήνας, «Java Server Pages», διαθέσιμο σε <http://netlab.teiath.gr/JSPWiki/attach/JavaLab/JavaServerPages.pdf>, (προσπέλαση: Αύγουστος 2011).
- [11] Marty Hall, «An Overview of Servlet and JSP Technology», paper, Prentice Hall and Sun Microsystems Press, <http://pdf.coreservlets.com/Overview.pdf>, (προσπέλαση: Αύγουστος 2011).
- [12] Oracle, «Java Servlet Technology Overview», Oracle Technology Network, Διαθέσιμο σε <http://www.oracle.com/technetwork/java/overview-137084.html>, (προσπέλαση: Αύγουστος 2011).
- [13] Wikipedia, Applet, Διαθέσιμο σε <http://el.wikipedia.org/wiki/Applet> (προσπέλαση Μάρτιος 2013)
- [14] Barth A, (2011) «HTTP State Management Mechanism» U.C. Berkley Διαθέσιμο σε <http://tools.ietf.org/html/rfc6265#section-3> (προσπέλαση Ιανουάριος 2013)
- [15] Wikipedia, Application programming interface, Διαθέσιμο σε http://en.wikipedia.org/wiki/Application_programming_interface (προσπέλαση Ιανουάριος 2013)
- [16] Marty Hall, «Overview of JSP Technology», paper, Prentice Hall and Sun Microsystems Press, Διαθέσιμο σε <http://pdf.coreservlets.com/JSP-Overview.pdf>, (προσπέλαση: Αύγουστος 2011).
- [17] Oracle, «Java Server Pages Overview», Oracle Technology Network, Διαθέσιμο σε <http://www.oracle.com/technetwork/java/overview-138580.html>, (προσπέλαση: Αύγουστος 2011).
- [18] Γκάμας Α., «Προγραμματισμός στην Πλευρά του Εξυπηρετητή: Τεχνολογία Java Server Pages (JSP) », Πανεπιστήμιο Ιωαννίνων, Διαθέσιμο σε http://www.cs.uoi.gr/~gkamas/internet/content/Lecture_08_JSP.pdf, (προσπέλαση: Αύγουστος 2011).

- [19] Marty Hall, «Integrating Servlets and JSP: The Model View (MVC) Architecture», paper, Prentice Hall and Sun Microsystems Press, Διαθέσιμο σε <http://pdf.coreservlets.com/MVC.pdf>, (προσπέλαση: Αύγουστος 2011).
- [20] eNode, «Model-View-Controller Pattern», tutorial, (2002), Διαθέσιμο σε <http://www.enode.com/x/markup/tutorial/mvc.html>, (προσπέλαση: Αύγουστος 2011).
- [21] Govind Seshadri (2000), «Understanding Java Server Pages Model 2 Architecture», JavaWorld.com, Διαθέσιμο σε <http://www.javaworld.com/javaworld/jw-12-1999/jw12-ssj-jspmvc.html>, (προσπέλαση: Αύγουστος 2011).
- [22] Wikipedia, Apache Struts, διαθέσιμο σε http://en.wikipedia.org/wiki/Apache_Struts (προσπέλαση Σεπτέμβριος 2012)
- [23] IBM, «Best practices for Struts development», Διαθέσιμο σε <http://www.ibm.com/developerworks/web/library/wa-struts/> (προσπέλαση Μάρτιος 2013)
- [24] The Apache Foundation, «Struts», Διαθέσιμο σε <http://struts.apache.org/1.x/faqs/kickstart.html> (προσπέλαση Σεπτέμβριος 2012)
- [25] TechTracer, (2007) «The Struts framework – Why choose it?» Διαθέσιμο σε <http://techtracer.com/2007/06/03/the-struts-framework-why-choose-it/> (προσπέλαση Σεπτέμβριος 2012)
- [26] Coreservlets, «Advantages of Struts» Διαθέσιμο σε <http://www.coreservlets.com/Apache-Struts-Tutorial/Understanding-Struts.html#Pros> (Προσπέλαση Σεπτέμβριος 2012)
- [27] eHow.com, «Difference Between Servlet & Struts», Διαθέσιμο σε http://www.ehow.com/facts_7843368_difference-between-servlet-struts.html#ixzz20UraaU00 (προσπέλαση Σεπτέμβριος 2012)
- [28] Alur, Deepak; John Crup, Dan Malks (2003), «Core J2EE Patterns, Best Practices and Design Strategies», 2nd Ed.. Sun Microsystems Press. pp. 650p.
- [29] Wikipedia, «JavaScript», (2011), Διαθέσιμο σε <http://en.m.wikipedia.org/wiki/JavaScript>, (προσπέλαση: Αύγουστος 2011).
- [30] W3Schools, (2011) «JavaScript Introduction», Διαθέσιμο σε http://www.w3schools.com/js/js_intro.asp, (προσπέλαση: Αύγουστος 2011).
- [31] Wikipedia, «jQuery», (2011), <http://en.m.wikipedia.org/wiki/JQuery>, (προσπέλαση: Αύγουστος 2011).
- [32] Doyle Matt, «What is jQuery?», tutorial, (February 2010), Elated.com, <http://www.elated.com/articles/what-is-jquery/>, (προσπέλαση: Αύγουστος 2011).
- [33] Wikipedia, «Ajax (programming)», (2011), http://en.m.wikipedia.org/wiki/Ajax_%28programming%29, (προσπέλαση: Αύγουστος 2011).
- [34] Βικιπαίδεια, «Ευχρηστία», <http://el.wikipedia.org/wiki>, (προσπέλαση: Αύγουστος 2011).
- [35] Westin A. F., «Privacy and Freedom», New York: Atheneum, 1967, page 7, <http://www.privacilla.org/fundamentals/privacyandfreedom.html>, (προσπέλαση: Αύγουστος 2011)
- [36] Kaplan, R. & Norton, D. (1996). «The balanced scorecard.» Harvard Business Review, Harvard Business Press, (January-February 1996): 76.
- [37] Mooraj, S., Oryon, D., Hostettler, D. (1999) «The balanced scorecard: a necessary god or an unnecessary evil?», European management journal Elsevier Science, Vol.17 No5 pp 481-491
- [38] Knapp K. (2001), «The balanced Scorecard: Historical Development and Context, As Developed by Robert Kaplan & David Norton», paper, Anderson University DBA
- [39] Wikipedia, «Balanced Scorecard» Διαθέσιμο σε http://en.wikipedia.org/wiki/Balanced_scorecard (προσπέλαση Φεβρουάριος 2013)
- [40] Μπουρντάλη, Ευάγ., Μεντζέλου, Ανδ. (2006) «Μοντέλα μέτρησης απόδοσης», Διοικητική ενημέρωση, Ειδική εκδοτική Α.Ε.Ε.Β.Ε. Τεύχος 38, σελ 95
- [41] Balanced Scorecard Institute, «Balanced Scorecard Basics» Διαθέσιμο σε <http://www.balancedscorecard.org/BSCResources/AbouttheBalancedScorecard/tabid/55/Default.aspx> (προσπέλαση Φεβρουάριος 2013)
- [42] Φωτίου, Φ., (2011), «Ανάπτυξη balanced scorecard, ως εργαλείο διοίκησης της στρατηγικής της εταιρείας», Διατριβή Μ.Π.Σ. Παν. Πειραιά Διαθέσιμο σε <http://digilib.lib.unipi.gr/dspace/handle/unipi/3894> (προσπέλαση Φεβρουάριος 2013)

- [43] Kaplan, R, Norton, D., (2002) «Partnering and the Balanced Scorecard» Harvard Business School Διαθέσιμο σε <http://hbswk.hbs.edu/item/3231.html> (προσπέλαση Ιανουάριος 2013)
- [44] AQS (2007) «Δείκτες αποδοτικότητας στο σύστημα logistics σε σύνδεση με το Balanced Scorecard», Διαθέσιμο σε http://www.aqs.gr/?cat_id=570&article_id=104 (προσπέλαση Ιανουάριος 2013)
- [45] Γορανίτου, Θ., (2008), «Balanced scorecard : το αποτελεσματικότερο σύστημα αξιολόγησης της επίδοσης των επιχειρήσεων», <http://digilib.lib.unipi.gr/dspace/handle/unipi/2782> (προσπέλαση Ιανουάριος 2013)
- [46] Boehm, D (2010) «Call Center Scorecards» White paper, Spectrum corporation-specorp.com Διαθέσιμο σε <http://www.slideshare.net/onstage/call-center-scorecarding> (προσπέλαση Ιανουάριος 2013)
- [47] Boehm, D «Call Center Statistics or Center Performance metrics? » White paper, Spectrum corporation-specorp.com Διαθέσιμο σε <http://www.slideshare.net/guest14c061/call-center-statistics-or-performance-metrics> (προσπέλαση Ιανουάριος 2013)
- [48] Stephen A. White, (2006) «OMG BPMN Tutorial», Διαθέσιμο σε http://www.bpmn.org/Documents/OMG_BPMN_Tutorial.pdf (προσπέλαση: Ιούλιος 2011).
- [49] PLANET – ΕΠΙΣΕΥ – ΑΤC, (2008), «Σημειογραφία των Προτύπων BPMN και UML (Activity Diagrams) », White paper, Διαθέσιμο σε, www.ermis.gov.gr/portal/pls/portal/docs/1/210341.PDF (προσπέλαση: Ιούλιος 2011)
- [50] Βικιπαίδεια, (2010), «Γλώσσες Μοντελοποίησης Λογισμικού», <http://el.wikipedia.org/wiki/UML>, (προσπέλαση: Ιούλιος 2011).
- [51] Κοντογιάννης Κ., «Τεχνολογία Λογισμικού», Διαφάνειες Διαλέξεων, Μετσόβιο Πολυτεχνείο, Τμήμα ΗΜΜΥ - Ε.Μ.Π., <http://courses.softlab.ntua.gr/softeng/>, (προσπέλαση: Ιούλιος 2011).
- [52] Randy Miller, «Practical UML: A Hands-On Introduction for Developers», Tutorial, διαθέσιμο σε, <http://edn.embarcadero.com/article/31863#use-case-diagram>, (προσπέλαση: Ιούλιος 2011).

Βιβλιογραφία

Για Helpdesk

Beisse, Fred, (2012), «A Guide to Computer User Support for Help Desk and Support Specialists, 5th Ed., », Course Technology, Cengage Learning

Knapp, Donna, (2003), «A Guide to Help Desk Concepts, Second Edition», Course Technology, Cengage Learning

Για ανάπτυξη λογισμικού

Sommerville Ian, (2009), «Βασικές αρχές Τεχνολογίας Λογισμικού», Μετάφραση από 8η αγγλική έκδοση, Επιμ. Δ. Τσιλογιάννης, Εκδόσεις Κλειδάριθμος

Perry, Bruce, (2004) «Java Servlet & JSP Cookbook», O'Reilly Media, Inc.

Hall, Marty. (2007), «Servlets και σελίδες διακομιστή Java : Τεχνολογίες πυρήνα» Μετάφραση Δ. Τσιλογιάννης., Εκδόσεις Κλειδάριθμος

Brown, D., Davis, B., Stanlick, S., (2008), «Struts in Action», Manning Publication

Για Balanced Scorecard

Kaplan, R, Norton, D., (2002), «The Balanced Scorecard: Translating Strategy into Action», Harvard

Olive, N., Roy J., Wetter, M. (2009), «Performance Drivers : A Practical Guide to Using the Balanced Scorecard », Wiley