



Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	Μηχανοργάνωση διαδικασιών ISO με χρήση υπηρεσιών διαδικτύου
Όνοματεπώνυμο Φοιτητή	Μαρία Σερεμετάκη
Πατρώνυμο	Γιώργος
Αριθμός Μητρώου	ΜΠΠΛ/ 06066
Επιβλέπων	Τσιχριτζής, Καθηγητής

Ημερομηνία Παράδοσης **Μάρτιος 2011**

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΑΙΑ

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)

Γεώργιος Τσιχριτζής
Καθηγητής

(υπογραφή)

Αποστόλου Δημήτριος
Επίκουρος Καθηγητής

(υπογραφή)

Χαράλαμπος
Κωνσταντόπουλος
Λέκτορας

Πίνακας Περιεχομένων

ΜΗΧΑΝΟΡΓΑΝΩΣΗ ΔΙΑΔΙΚΑΣΙΩΝ ISO ΜΕ ΧΡΗΣΗ ΥΠΗΡΕΣΙΩΝ ΔΙΑΔΙΚΤΥΟΥ	1
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	3
ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ	5
ΠΕΡΙΕΧΟΜΕΝΑ ΠΙΝΑΚΩΝ	6
1. ΕΙΣΑΓΩΓΗ	7
1.1. Γενικά	7
1.2. Σκοπός της Εργασίας	9
1.3. Δομή της Εργασίας	10
2. ΠΙΣΤΟΠΟΙΗΣΗ ΠΡΟΪΟΝΤΩΝ ΚΑΙ ΔΙΑΔΙΚΑΣΙΩΝ	11
2.1. Σημασία Προτύπων	11
2.2. Ο Ρόλος των Προτύπων	12
2.3. Οφέλη από την Χρήση Προτύπων	13
2.4. Διαδικασία Πιστοποίησης από Αρμόδιο Φορέα	15
2.4.1. Γενικά / Διαδικασία	15
2.4.2. Σκοπός της Πιστοποίησης	16
2.5. Πιστοποιήσεις ISO	19
2.5.1. Γενική Περιγραφή	19
2.5.2. Ορισμός Προδιαγραφών κατά ISO	20
2.5.3. Ανάπτυξη Προτύπων ISO	21
2.5.4. Στάδια Ανάπτυξης Διαδικασιών	27
2.5.4.1. Στάδιο 1: Στάδιο πρότασης	28
2.5.4.2. Στάδιο 2: Φάση προετοιμασίας	28
2.5.4.3. Στάδιο 3: Στάδιο επιτροπής	28
2.5.4.4. Στάδιο 4: Στάδιο έρευνας	29
2.5.4.5. Στάδιο 5: Στάδιο έγκρισης	29
2.5.4.6. Στάδιο 6: Στάδιο δημοσίευσης	30
3. ΥΠΗΡΕΣΙΕΣ ΔΙΑΔΙΚΤΥΟΥ	31
3.1. Γενικά	31
3.2. Αρχιτεκτονικές Εφαρμογών Διαδικτύου	32
3.2.1. Peer to Peer	32
3.2.1.1. Γενική Περιγραφή	32

3.2.1.2.	Κατηγορίες Δικτύων Peer-to-Peer	33
3.2.1.3.	Χρήση των Peer-to-Peer δικτύων	35
3.2.2.	Repository Model	36
3.2.2.1.	Εισαγωγή	36
3.2.2.2.	Σκοποί του Αποθετήρια	37
3.2.3.	Client - Server	39
3.2.3.1.	Γενική Περιγραφή	39
3.2.3.2.	Λειτουργία	40
4.	ΕΦΑΡΜΟΓΗ ΜΗΧΑΝΟΡΓΑΝΩΣΗΣ ΔΙΑΔΙΚΑΣΙΩΝ ISO	43
4.1.	Αρχιτεκτονική	43
4.2.	Σχεδιασμός Υλοποίησης	45
4.3.	Πρότυπη Υλοποίηση	47
4.3.1.	Βάση Δεδομένων	47
4.3.2.	Ανάπτυξη Εφαρμογής	49
5.	ΠΕΙΡΑΜΑΤΙΚΟ ΜΕΡΟΣ	51
5.1.	Χρήση της Πρότυπης Εφαρμογής	51
5.2.	Λειτουργικό Σενάριο	52
5.2.1.	Λειτουργία Διαχειριστή Εφαρμογής	52
5.2.2.	Λειτουργία Απλού Χρήστη Εφαρμογής	54
6.	ΣΥΝΟΨΗ – ΣΥΜΠΕΡΑΣΜΑΤΑ	56
6.1.	Σύνοψη	56
6.2.	Μελλοντικές Επεκτάσεις – Συμπεράσματα	57
	ΠΑΡΑΡΤΗΜΑ Α ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ	58
	ΠΑΡΑΡΤΗΜΑ Β	75
	ΑΚΡΩΝΥΜΙΑ	76
	ΒΙΒΛΙΟΓΡΑΦΙΑ	77

Πίνακας Εικόνων

Εικόνα 3.1 Αρχιτεκτονική Peer to Peer.....	33
Εικόνα 3.2 Αρχιτεκτονική Client - Server.....	41
Εικόνα 4.1 Προτεινόμενη Αρχιτεκτονική.....	43
Εικόνα 4.2 Διάταξη Υλοποίησης.....	46
Εικόνα 4.3. Σχηματική Διάταξη Πινάκων Βάσης Δεδομένων.....	48
Εικόνα 5.1. Οθόνη Πιστοποίησης Χρηστών.....	51
Εικόνα 5.2. Οθόνη Εγγραφής Χρηστών.....	51
Εικόνα 5.3. Κεντρική Σελίδα Εφαρμογής.....	52
Εικόνα 5.4. Κεντρική Σελίδα Ενεργειών.....	53
Εικόνα 5.5. Κεντρική Σελίδα Διαδικασιών (Processes).....	53
Εικόνα 5.6. Τροποποίηση Διαδικασιών (Processes).....	54
Εικόνα 5.7. Συσχέτιση Διαδικασιών (Processes).....	54
Εικόνα 5.8. Κεντρική Σελίδα Διαδικασιών για Χρήστη με Περιορισμένη Πρόσβαση.....	55

Περιεχόμενα Πινάκων

Πίνακας 1. Κώδικας Διαχείρισης Σύνδεσης στη Βάση Δεδομένων.....	58
Πίνακας 2. Βοηθητική Κλάση Παρουσίασης	59
Πίνακας 3. Διαχείριση Χρηστών και Δικαιωμάτων	60
Πίνακας 4. Σελίδα Πιστοποίησης Χρήστη.....	64
Πίνακας 5. Σελίδα Παρουσίασης Διαδικασιών	65
Πίνακας 6. Σελίδα Διαχείρισης Διαδικασίας	67

1. Εισαγωγή

1.1. Γενικά

Παγκοσμίως καταγράφονται περισσότερα από 150.000.000 ατυχήματα σε ώρα και χώρο εργασίας τα οποία συμβαίνουν σε ετήσια βάση και έχουν άμεσο αποτέλεσμα το θάνατο 100.000 εργαζομένων ή την πρόκληση προσωρινής ή μόνιμης ανικανότητας για εργασία, είτε ολική είτε μερική. Εκτός αυτού το κόστος αυτών των ατυχημάτων είναι πολύ μεγάλο.

Η αύξηση του κινδύνου και του επαγόμενου ρίσκου σε επαγγελματικούς χώρους και κατ' επέκταση η ανάπτυξη ασθενειών και εργατικών ατυχημάτων που έχουν τα αίτιά τους στους εργασιακούς χώρους συνδέεται με κάποια αρνητικά αποτελέσματα, όπως η πληγείσα ανθρώπινη σωματική ακεραιότητα, πολλές χαμένες εργατοώρες, υψηλές ασφαλιστικές αποζημιώσεις, καθώς και πολλές άλλες επιπτώσεις. Η πρόληψη και η καταπολέμηση των εργατικών ατυχημάτων είναι, επομένως, προφανής ανάγκη και επείγον κοινωνικό καθήκον.

Στα γενικότερα πλαίσια που περιγράφηκαν στις προηγούμενες γραμμές, η εφαρμογή ενός συστήματος διαχείρισης ασφάλειας και ποιότητας σε μια επιχείρηση παρέχει τη δυνατότητα στην επιχείρηση τόσο να παρέχει αναβαθμισμένα προϊόντα ή/και υπηρεσίες που ανταποκρίνονται με συνέπεια στις απαιτήσεις της αγοράς γενικά και των πελατών της, όσο και να παράγεται αυτό το προϊόν με ασφάλεια για τους εργαζομένους.

Παράλληλα μπορεί να διασφαλιστεί η συμμόρφωση με τις διάφορες νομικές και κανονιστικές απαιτήσεις, όπου αυτό είναι απαραίτητο, αλλά και γίνεται ευκολότερη και αποδοτικότερη διαχείριση όλων των σταδίων μιας εμπορικής διαδικασίας.

Αναγνωρίζοντας τις προαναφερθείσες ανάγκες, πολλοί διεθνείς οργανισμοί έχουν αναλάβει την θέσπιση κανόνων και διαδικασιών διασφάλισης και παράλληλα οι περισσότερες εταιρείες ακολουθούν την

πιστοποίηση των σταδίων της παραγωγικής τους διαδικασίας σύμφωνα με τα διεθνώς αναγνωρισμένα πρότυπα. Έτσι μπορεί και διασφαλίζεται η ποιότητα των παρεχόμενων υπηρεσιών και προϊόντων, οι κανόνες και διαδικασίες ασφαλείας τόσο κατά την παραγωγή όσο και κατά την μεταφορά των προϊόντων, αλλά και γενικότερα δημιουργούνται οι υποδομές για την αέναη ανάπτυξη και βελτίωση των εταιρειών.

1.2. Σκοπός της Εργασίας

Παρατηρώντας τον διαρκώς αυξανόμενο αριθμό εταιρειών και οργανισμών που έχουν πιστοποιηθεί και ακολουθούν προτυποποιημένες διαδικασίες σε όλα τα στάδια της παραγωγικής τους διαδικασίας, αντιλαμβανόμαστε τον μεγάλο όγκο και πολυπλοκότητα του συστήματος εφαρμογής και παρακολούθησης των διαδικασιών για κάθε εταιρεία ή / και οργανισμό.

Πράγματι, και αναλόγως του τομέα στον οποίο δραστηριοποιείται η κάθε εταιρεία, η επιτήρηση, η ανάπτυξη νέων διαδικασιών, η τροποποίηση υπαρχόντων και γενικά όλη η δραστηριότητα που σχετίζεται με τη διαχείριση και ανάπτυξη των προτυποποιημένων διαδικασιών είναι αρκετά πολύπλοκη και επίπονη διαδικασία. Στην παρούσα εργασία γίνεται μια προσπάθεια για εφαρμογή τεχνολογιών του διαδικτύου για την μηχανοργάνωση, με σκοπό να διευκολυνθεί η αποκεντρωμένη προσβασιμότητα στα έγγραφα και στοιχεία διαχείρισης.

Για το σκοπό αυτό μελετώνται βιβλιογραφικά τόσο ο τομέας της πιστοποίησης προϊόντων και διαδικασιών, όσο και της αρχιτεκτονικής του διαδικτύου, ώστε να προκύψει συμπερασματικά η δυνατότητα εφαρμογής ενός μοντέλου μηχανοργάνωσης των προαναφερθεισών διαδικασιών που να στηρίζεται στις υπηρεσίες διαδικτύου.

1.3. Δομή της Εργασίας

Έχοντας ολοκληρώσει μια συνοπτική περιγραφή της υπάρχουσας κατάστασης και των προκυπτουσών αναγκών στον τομέα της βιομηχανίας όσον αφορά την πιστοποίηση διαδικασιών και προϊόντων, στη συνέχεια η εργασία δομείται ως εξής.

Στο δεύτερο κεφάλαιο γίνεται μια βιβλιογραφική ανασκόπηση της πιστοποίησης προϊόντων και διαδικασιών, στην οποία περιλαμβάνονται όλες οι σχετικές συνιστώσες. Έτσι περιγράφεται τόσο αυτή καθαυτή η διαδικασία ανάπτυξης προτύπων πιστοποίησης, όσο και η διαδικασία εφαρμογή τους και η σημασία που έχει στην αποτελεσματικότητα και λειτουργία των επιχειρήσεων.

Στο τρίτο κεφάλαιο γίνεται μια παρουσίαση της τεχνολογικής υποδομής στο διαδίκτυο. Περιγράφονται οι αρχιτεκτονικές των υπάρχουσών εφαρμογών διαδικτύου, με απώτερο στόχο να επιλεγεί η αποδοτικότερη / καταλληλότερη για την εφαρμογή στους σκοπούς της παρούσας εργασίας.

Στο τέταρτο κεφάλαιο παρουσιάζεται το κεντρικό αντικείμενο της παρούσας εργασίας, ήτοι η αρχιτεκτονική, ο σχεδιασμός και η πρότυπη υλοποίηση της εφαρμογής που αναπτύχθηκε στα πρότυπα των υπηρεσιών διαδικτύου για την μηχανοργάνωση των διαδικασιών ISO.

Στο πέμπτο κεφάλαιο περιγράφεται η διαδικασία εγκατάστασης και εκτέλεσης της εφαρμογής και στο τελευταίο (έκτο) κεφάλαιο γίνεται μια επισκόπηση των αποτελεσμάτων της παρούσας εργασίας.

2. Πιστοποίηση Προϊόντων και Διαδικασιών

2.1. Σημασία Προτύπων

Τα πρότυπα έχουν μια πολύ μεγάλη θετική συμβολή στις περισσότερες πτυχές της ζωής μας. Καταρχήν εξασφαλίζουν επιθυμητά χαρακτηριστικά των προϊόντων και των υπηρεσιών, όπως η ποιότητα, η φιλικότητα προς το περιβάλλον, η ασφάλεια, η αξιοπιστία, η αποτελεσματικότητα και εναλλαξιμότητα - και όλα αυτά με έναν οικονομικά αποδοτικό τρόπο.

Όταν τα προϊόντα και οι υπηρεσίες ανταποκρίνονται στις προσδοκίες μας, έχουμε την τάση να το θεωρούμε αυτό ως δεδομένο και να αγνοούμε το ρόλο των προτύπων. Ωστόσο, όταν τα πρότυπα απουσιάζουν, σύντομα παρατηρούμε τη διαφορά. Σύντομα βρίσκουμε τα προϊόντα να αποδεικνύονται χαμηλής ποιότητας, να μην ταιριάζουν, να είναι ασύμβατα με τον εξοπλισμό που έχουμε ήδη, να είναι αναξιόπιστα ή και επικίνδυνα.

Όταν τα προϊόντα, τα συστήματα, τα μηχανήματα και οι συσκευές που λειτουργούν καλά και με ασφάλεια, είναι συχνά επειδή συμμορφώνονται με κάποια πρότυπα. Και η οργάνωση που είναι υπεύθυνη για πολλές χιλιάδες των προτύπων στον κόσμο είναι το ISO.

2.2. Ο Ρόλος των Προτύπων

Τα πρότυπα εξασφαλίζουν τις παρακάτω διαδικασίες:

- Κάνουν την ανάπτυξη, την κατασκευή και την προμήθεια προϊόντων και υπηρεσιών πιο αποτελεσματική, ασφαλέστερη και καθαρότερη.
- Διευκολύνουν τις εμπορικές συναλλαγές μεταξύ των χωρών και τις καταστούν δικαιότερες.
- Παρέχουν στις κυβερνήσεις μια τεχνική βάση για την υγεία, την ασφάλεια και την περιβαλλοντική νομοθεσία, καθώς και την αξιολόγηση συμμόρφωσης με πρότυπα και νομοθεσίες.
- Μοιράζουν την τεχνολογική πρόοδο και τις καλές πρακτικές διαχείρισης.
- Διαδίδουν την καινοτομία.
- Προστατεύουν τον καταναλωτή, και γενικά τους χρήστες των προϊόντων και υπηρεσιών.
- Διευκολύνουν την ζωή με την παροχή λύσεων σε συνηθισμένα προβλήματα.

2.3. Οφέλη από την Χρήση Προτύπων

Το πρότυπα παρέχουν τεχνολογικά, οικονομικά και κοινωνικά οφέλη, σε πολλούς διαφορετικούς τομείς τους οποίους αναφέρουμε στη συνέχεια.

Για τις επιχειρήσεις, η ευρεία υιοθέτηση των Διεθνών Προτύπων σημαίνει ότι οι προμηθευτές μπορούν να αναπτύξουν και να προσφέρουν προϊόντα και υπηρεσίες σύμφωνα με κάποιες προδιαγραφές που έχουν ευρεία διεθνή αποδοχή στον τομέα τους. Ως εκ τούτου, οι επιχειρήσεις που χρησιμοποιούν τα Διεθνή Πρότυπα μπορούν να είναι ανταγωνιστικές σε πολλές περισσότερες αγορές σε όλο τον κόσμο.

Για τους πρωτοπόρους των νέων τεχνολογιών, τα διεθνή πρότυπα σε θέματα όπως η ορολογία, η συμβατότητα, η ταχύτητα και η ασφάλεια μέχρι και τη διάδοση των καινοτομιών και την ανάπτυξή τους σε κατασκευάσιμα και εμπορεύσιμα προϊόντα.

Για τους πελάτες, η παγκόσμια συμβατότητα της τεχνολογίας, η οποία επιτυγχάνεται όταν τα προϊόντα και οι υπηρεσίες βασίζονται σε διεθνή πρότυπα, τους δίνει μια ευρεία επιλογή προσφορών. Μπορούν επίσης να επωφεληθούν από τα αποτελέσματα του ανταγωνισμού μεταξύ των προμηθευτών.

Για τις κυβερνήσεις, τα διεθνή πρότυπα παρέχουν τεχνολογικές και επιστημονικές βάσεις που στηρίζουν την υγεία, την ασφάλεια και την περιβαλλοντική νομοθεσία.

Για τους υπαλλήλους του εμπορίου, τα διεθνή πρότυπα δημιουργούν "ίσους όρους" για όλους τους ανταγωνιστές στις αγορές αυτές. Η ύπαρξη διαφορετικών εθνικών ή περιφερειακών προτύπων μπορούν να δημιουργήσουν τεχνικά εμπόδια στο εμπόριο. Τα Διεθνή Πρότυπα είναι τα τεχνικά μέσα με τα οποία πολιτικά εμπορικές συμφωνίες μπορούν να εφαρμοστούν στην πράξη.

Για τις αναπτυσσόμενες χώρες, τα διεθνή Πρότυπα, που αντιπροσωπεύουν μια διεθνή συναίνεση σχετικά με την εξέλιξη της

τεχνολογίας αποτελούν σημαντική πηγή τεχνογνωσίας. Με τον καθορισμό των χαρακτηριστικών ότι τα προϊόντα και οι υπηρεσίες θα πρέπει να αναμένονται να ανταποκριθεί στις εξαγωγικές αγορές, τα διεθνή πρότυπα παρέχουν στις αναπτυσσόμενες χώρες τη βάση για την λήψη των σωστών αποφάσεων όταν επενδύουν περιορισμένων πόρων τους και έτσι να αποφύγουν διασπάθιση τους.

Για τους καταναλωτές, η συμμόρφωση των προϊόντων και των υπηρεσιών με τα Διεθνή Πρότυπα παρέχει βεβαιότητα σχετικά με την ποιότητα, την ασφάλεια και την αξιοπιστία τους.

Για όλους εμάς, τα διεθνή πρότυπα συμβάλλουν στην ποιότητα της ζωής γενικότερα, διασφαλίζοντας ότι η μεταφορά, μηχανήματα και εργαλεία που χρησιμοποιούμε είναι ασφαλή.

Για τον πλανήτη που μας φιλοξενεί, τα διεθνή πρότυπα για τον αέρα, το νερό και την ποιότητα του εδάφους, σχετικά με τις εκπομπές των αερίων και ακτινοβολίας και περιβαλλοντικές πτυχές των προϊόντων μπορεί να συμβάλει στις προσπάθειες για την προστασία του περιβάλλοντος.

2.4. Διαδικασία Πιστοποίησης από Αρμόδιο Φορέα

2.4.1. Γενικά / Διαδικασία

Με την εγκατάσταση ενός Συστήματος Διαχείρισης Ποιότητας (ΣΔΠ), το οποίο είναι προσαρμοσμένο στις ανάγκες και στις ιδιαιτερότητες της κάθε επιχείρησης υπάρχει ως στόχος η αφομοίωση της φιλοσοφίας της Ποιότητας από όλο το ανθρώπινο δυναμικό της η διατήρηση καθώς και η συνεχής βελτίωση ενός υψηλού επιπέδου παραγωγής προϊόντων/υπηρεσιών.

Πρώτο στάδιο: Ανάλυση Υφιστάμενης Κατάστασης

1. Προσέγγιση Κύριων & Υποστηρικτικών Λειτουργιών / Διεργασιών
2. Προσέγγιση αναγκαίων διαδικασιών με βάση το Πρότυπο

Δεύτερο Στάδιο: Σχεδιασμός & Ανάπτυξη Συστήματος

1. Ανάπτυξη Κύριων & Υποστηρικτικών διεργασιών / διαδικασιών
2. Ανάπτυξη Συστήματος (Εγχειρίδιο Ποιότητας, Διαδικασίες, Έντυπα & Οδηγίες)
3. Εγκατάσταση Λογισμικού

Τρίτο Στάδιο: Εφαρμογή Συστήματος

1. Εκπαίδευση
- Καθορισμός Αρμοδιοτήτων ευθυνών Διοίκησης

- Καθορισμός ευθυνών Προσωπικού
- Καθορισμός ευθυνών εσωτερικών επιθεωρητών

2. Ανασκόπηση Συστήματος

Τέταρτο Στάδιο: Πιστοποίηση Συστήματος

1. Επιθεώρηση από αρμόδιο Φορέα Πιστοποίησης
2. Διορθωτικές Ενέργειες

2.4.2. Σκοπός της Πιστοποίησης

Σκοπός της υπηρεσίας είναι η εγκατάσταση Συστήματος Διαχείρισης Ποιότητας (ΣΔΠ), σαφώς εναρμονισμένο με τις ανάγκες και την κουλτούρα της επιχείρησης και με στόχους την αφομοίωση της φιλοσοφίας της Ποιότητας από όλο το ανθρώπινο δυναμικό της τη διατήρηση και συνεχή βελτίωση ενός υψηλού επιπέδου παραγωγής προϊόντων/υπηρεσιών

Μεθοδολογία - Οφέλη

1Η ΦΑΣΗ: Ανάλυση Υφιστάμενης Κατάστασης

Στην πρώτη φάση συμπεριλαμβάνεται η ανάλυση της υπάρχουσας κατάστασης μέσω της ανάλυσης των κύριων και υποστηρικτικών λειτουργιών, αλλά και των αναγκαίων/απαραίτητων διαδικασιών βάσει των προδιαγραφών που θέτει το πρότυπο.

2Η ΦΑΣΗ: Σχεδιασμός & Ανάπτυξη Συστήματος

Στην δεύτερη φάση συμπεριλαμβάνεται η ανάπτυξη των προαναφερθέντων κύριων και υποστηρικτικών λειτουργιών, αλλά και των αναγκαιών/απαραίτητων διαδικασιών βάσει των προδιαγραφών που θέτει το πρότυπο. Παράλληλα προδιαγράφεται η εγκατάσταση σχετικού λογισμικού.

3Η ΦΑΣΗ: Εφαρμογή Συστήματος

Στην Τρίτη φάση λαμβάνει χώρα η εφαρμογή του συστήματος. Πιο συγκεκριμένα γίνεται ο κύκλος της εκπαίδευσης που περιλαμβάνει τα παρακάτω:

- Καθορισμός Αρμοδιοτήτων ευθυνών Διοίκησης
- Καθορισμός ευθυνών Προσωπικού
- Καθορισμός ευθυνών εσωτερικών επιθεωρητών

Τέλος, στη φάση αυτή περιλαμβάνεται και η ανασκόπηση του όλου συστήματος.

4Η ΦΑΣΗ: Πιστοποίηση Συστήματος

Στην τέταρτη φάση γίνεται η επιθεώρηση του συστήματος από τον αντίστοιχο αρμόδιο φορέα πιστοποίησης και συστήνονται και εφαρμόζονται τυχόν απαιτούμενες διορθωτικές ενέργειες.

Αποτέλεσμα της πιστοποίησης είναι όλα όσα περιγράφηκαν στην προηγούμενη ενότητα προσθέτοντας ότι η αποτελεσματικότητα που επιτυγχάνεται είναι πλέον αυστηρά ορισμένη μέσω μετρήσιμων μεγεθών που έχουν οριστεί, θεσπιστεί και ενσωματωθεί στην εκάστοτε παραγωγική διαδικασία.

Επίσης μέσα από τη χρήση των προτύπων προδιαγράφονται τα επόμενα οφέλη:

- Εξασφαλίζεται η ικανοποίηση αναγκών των πελατών
- Βελτιώνεται η θέση του φορέα ως προς τον ανταγωνισμό
- Βελτιώνονται και αυξάνονται οι δυνατότητες του ανθρώπινου δυναμικού
- Παράγονται προϊόντα ή παρέχονται υπηρεσίες σταθερής ποιότητας
- Παρακολουθείται η αποτελεσματικότητα θεσπίζοντας μετρήσιμους στόχους ποιότητας.

2.5. Πιστοποιήσεις ISO

2.5.1. Γενική Περιγραφή

Επειδή ο "Διεθνής Οργανισμός Τυποποίησης» θα είχε διαφορετικά ακρωνύμια σε διαφορετικές γλώσσες ("IOS" στα αγγλικά, «OIN" στα γαλλικά - Organisation internationale de normalisation), οι ιδρυτές του, αποφάσισαν να του δώσουν ένα δεύτερο, σύντομο όνομα, για όλες τις χρήσεις. Επέλεξαν το "ISO", που προέρχεται από την ελληνική λέξη «Ίσος». Όποια και αν είναι η χώρα, ανεξάρτητα από τη γλώσσα, η σύντομη μορφή του ονόματος της οργάνωσης είναι πάντα ISO. [6]

Ο διεθνής οργανισμός ISO (International Organization for Standardization) είναι ο μεγαλύτερος οργανισμός ανάπτυξης και δημοσίευσης Διεθνών Προτύπων στον κόσμο. Το ISO είναι ένα δίκτυο εθνικών ιδρυμάτων προτύπων 163 χωρών, ένα μέλος ανά χώρα, με μια Κεντρική Γραμματεία στη Γενεύη της Ελβετίας, που συντονίζει το σύστημα. Παράλληλα ο οργανισμός ISO είναι μια μη κυβερνητική οργάνωση που διαμορφώνει μια γέφυρα μεταξύ του δημόσιου και του ιδιωτικού τομέα. Από τη μία πλευρά, πολλά από τα κράτη μέλη του αποτελούν μέρος της κυβερνητικής δομής των χωρών τους, είτε έχουν εντολή από την κυβέρνησή τους. Από την άλλη πλευρά, τα υπόλοιπα μέλη έχουν τις ρίζες τους μεμονωμένα στον ιδιωτικό τομέα, που έχει συσταθεί από τις εθνικές συμπράξεις και ενώσεις του κλάδου.

Συνεπώς, το ISO δίνει τη δυνατότητα μιας συναίνεσης που θα επιτευχθεί σε λύσεις που ικανοποιούν τόσο τις απαιτήσεις των επιχειρήσεων όσο και τις ευρύτερες ανάγκες της κοινωνίας.

Η εφαρμογή του προτύπου ISO 9001:2008 στην καθημερινή πρακτική λειτουργία μιας επιχείρησης έχει τους παρακάτω οριοθετημένους στόχους:

- Παροχή εμπιστοσύνης προς τον πελάτη ότι το προϊόν/υπηρεσία που του παρασχέθηκε είναι σύμφωνη με τις προδιαγραφές που τέθηκαν.

- Παράδοση προϊόντων στον πελάτη τα οποία συμμορφώνονται με τις προδιαγραφές που έχουν συμφωνηθεί μεταξύ πελάτη και επιχείρησης.
- Εφαρμογή μέτρων συνεχούς βελτίωσης με σκοπό την ικανοποίηση όχι μόνο των τωρινών αλλά και μελλοντικών απαιτήσεων των πελατών της.

Τα κύρια οφέλη της εφαρμογής του διεθνούς αυτού προτύπου είναι:

- Στρατηγικό πλεονέκτημα έναντι των ανταγωνιστών
- Καλύτερος διαχειριστικός έλεγχος
- Διαφάνεια στην εφαρμογή των διαδικασιών της επιχείρησης
- Αξιοπιστία έναντι των πελατών και των προμηθευτών
- Συνεχής βελτίωση

2.5.2. Ορισμός Προδιαγραφών κατά ISO

Ένα κανονιστικό έγγραφο, που αναπτύχθηκε σύμφωνα με τις ορισμένες διαδικασίες συναίνεσης, εγκρίνεται από τα μέλη του ISO και τα μέλη της αρμόδιας επιτροπής, σύμφωνα με το Μέρος 1 της οδηγίας ISO ως ένα σχέδιο διεθνούς προτύπου ή / και ως τελικό σχέδιο ενός διεθνούς προτύπου και η οποία έχει δημοσιευθεί από την ISO Κεντρική Γραμματεία.

Ένα κείμενο που αντιστοιχεί σε ένα εγκεκριμένο στοιχείο εργασίας αναπτύσσεται ανάλογα με τις ανάγκες, μέσω της προετοιμασίας και / ή των σταδίων της επιτροπής έως ότου επιτευχθεί συναίνεση στην επιτροπή. (Σε περίπτωση αμφιβολίας, η έγκριση από τα 2 / 3 των μελών με δικαίωμα ψήφου μπορεί να θεωρηθεί ότι συνιστά συναίνεση. Το κείμενο που υποβάλλεται σε όλους τους φορείς-μέλη του ISO για πέντε μήνες για ψηφοφορία ως σχέδιο του Διεθνούς Προτύπου (DIS) και είναι εγκριθεί, αν δηλαδή τα δύο τρίτα των ψήφων των μελών είναι

καταφατικά και όχι περισσότερο από το ένα τέταρτο του συνόλου των ψήφων είναι αρνητικά.

Ένα τελικό κείμενο είναι έτοιμο, λαμβάνοντας υπόψη τις παρατηρήσεις των μελών σχετικά με την DIS και το κείμενο αυτό έχει εκδοθεί για την επίσημη ψηφοφορία ως τελικό σχέδιο Διεθνούς Προτύπου (FDIS). Εάν το κείμενο εγκριθεί και πάλι από τα δύο τρίτα των με δικαίωμα ψήφου μελών και, εάν δεν υπερβαίνουν το ένα τέταρτο των ψήφων οι αρνητικές, τότε το κείμενο έχει εγκριθεί και η Κεντρική Γραμματεία το δημοσιεύει το Διεθνές Πρότυπο.

2.5.3. Ανάπτυξη Προτύπων ISO

Πώς ο οργανισμός ISO αποφασίζει να αναπτύξει ένα πρότυπο ISO και δρομολογεί την ανάπτυξη νέων προτύπων για την αντιμετώπιση σαφώς αναγνωρισμένες ανάγκες για διάφορους τομείς της βιομηχανίας;

Ένας τομέας της βιομηχανίας ή οποιαδήποτε άλλη ενδιαφερόμενη ομάδα επικοινωνεί συνήθως την αίτησή της για ένα πρότυπο σε ένα από τα εθνικά μέλη του ISO. Η τελευταία προτείνει στη συνέχεια το νέο αντικείμενο εργασιών στη σχετική επιτροπή ανάπτυξης προτύπων στον οργανισμό ISO. Νέα στοιχεία εργασίας μπορούν επίσης να προτείνονται από τις οργανώσεις σε συνεργασία με αυτές τις επιτροπές. Όταν θέματα εργασίας δεν σχετίζονται με τις υφιστάμενες επιτροπές, οι προτάσεις μπορούν επίσης να πραγματοποιούνται από το μέλη ISO για τη δημιουργία νέων τεχνικών επιτροπών για την κάλυψη νέων τομέων δραστηριότητας.

Πώς ο οργανισμός ISO αποφασίζει να αναπτύξει ένα πρότυπο

Το ISO δρομολογεί την ανάπτυξη νέων προτύπων για την αντιμετώπιση τομέων και φορέων που εκφράζουν μια σαφώς αναγνωρισμένη ανάγκη για αυτούς. Ένας επιχειρηματικός κλάδος ή μια άλλη ομάδα ενδιαφερομένων επικοινωνεί συνήθως την αίτησή της για ένα πρότυπο σε ένα από τα εθνικά μέλη του ΔΟΖ. Η τελευταία προτείνει στη συνέχεια το νέο αντικείμενο εργασιών στη σχετική τεχνική επιτροπή ανάπτυξης προτύπων ISO στον τομέα αυτό. Νέα στοιχεία εργασίας μπορούν επίσης να προτείνονται από τις οργανώσεις σε συνεργασία με τις σχετικές επιτροπές. Όταν θέματα εργασίας δεν σχετίζονται με τις υφιστάμενες επιτροπές, οι προτάσεις μπορούν επίσης να πραγματοποιούνται από τα μέλη του οργανισμού ISO για τη δημιουργία νέων τεχνικών επιτροπών για την κάλυψη νέων τομέων δραστηριότητας.

Για να γίνει αποδεκτό για ανάπτυξη το προτεινόμενο αντικείμενο εργασίας πρέπει να έχει την υποστήριξη της πλειοψηφίας των συμμετεχόντων μελών του ISO τεχνικής επιτροπής η οποία, μεταξύ άλλων κριτηρίων, επαληθεύει την «παγκόσμια σημασία» του προτεινόμενου σημείου - αυτό σημαίνει ότι ανταποκρίνεται πράγματι σε μια διεθνή ανάγκη και τελικά θα είναι κατάλληλα για την εφαρμογή στην ευρύτερη δυνατή βάση σε όλο τον κόσμο.

Εκτός από τις τεχνικές επιτροπές που είναι υπεύθυνες για την τυποποίηση σε έναν συγκεκριμένο τομέα, ο οργανισμός ISO έχει επίσης επιτροπές ανάπτυξης πολιτικής για τη διευθέτηση της τυποποίησης σε ανάγκες των αναπτυσσόμενων χωρών (DEVCO), των καταναλωτών (COPOLCO) και την αξιολόγηση της συμμόρφωσης (CASCO). Αυτά μπορεί να συστήσουν την ανάπτυξη νέων προτύπων για τις ομάδες ενδιαφερομένων τους, οι οποίες στη συνέχεια υποβάλλονται στην διαδικασία έγκρισης που περιγράφεται ανωτέρω, ή, στην περίπτωση της CASCO, στην ανάπτυξη νέων προτύπων.

Ποιος αναπτύσσει πρότυπα ISO

Τα πρότυπα του ISO που αναπτύχθηκαν από τεχνικές επιτροπές, (υποεπιτροπών και τεχνικών επιτροπών) που αποτελείται από εμπειρογνώμονες από τους βιομηχανικούς, τεχνικούς και επιχειρηματικούς τομείς που έχουν ζητήσει τα πρότυπα, και τα οποία στη συνέχεια να τους φανούν χρήσιμα. Οι εμπειρογνώμονες αυτοί μπορούν να ενωθούν με τους εκπροσώπους των κυβερνητικών υπηρεσιών, των εργαστηριακών δοκιμών, ενώσεων καταναλωτών, τις μη κυβερνητικές οργανώσεις και τους ακαδημαϊκούς κύκλους.

Οι προτάσεις για τη δημιουργία των νέων τεχνικών επιτροπών υποβάλλονται σε όλους τους ISO εθνικούς φορείς μέλη, οι οποίοι μπορούν να επιλέξουν να συμμετέχουν (Participants), παρατηρητή (Observers) ή μη μέλη της επιτροπής. Η γραμματεία (δηλαδή το όργανο που θα παρέχει τη διοικητική υποστήριξη στο έργο της επιτροπής) χορηγείται από το Τεχνικό διοικητικό συμβούλιο (το οποίο εκθέσει προς το Συμβούλιο ISO), συνήθως στον οργανισμό μέλος του ISO που έκανε την πρόταση. Η γραμματεία είναι υπεύθυνη για τον ορισμό ενός ατόμου να ενεργεί ως πρόεδρος της τεχνικής επιτροπής. Η καρέκλα είναι διορίζεται τυπικά από την τεχνική διοικητικό συμβούλιο.

Οι εμπειρογνώμονες συμμετέχουν ως εθνικές αντιπροσωπείες, που επιλέγονται από τον ISO εθνικού οργανισμό μέλος για την εκάστοτε χώρα. Οι εθνικές αντιπροσωπείες που απαιτούνται για να μην εκπροσωπούν μόνο τις απόψεις των οργανώσεων στις οποίες συμμετέχουν εμπειρογνώμονες για το έργο τους. Οι εθνικές αντιπροσωπείες συνήθως βασίζονται και υποστηρίζονται από τις αντίστοιχες εθνικές επιτροπές.

Σύμφωνα με το πρότυπο ISO τα μέλη του εθνικού οργανισμού μέλους αναμένεται να λάβουν υπόψη τις απόψεις όλων των μερών που

ενδιαφέρονται για το πρότυπο υπό ανάπτυξη. Αυτό τους επιτρέπει να παρουσιάσουν μια ενοποιημένη, εθνική θέση σε συναίνεση με την τεχνική επιτροπή.

Διεθνείς και περιφερειακοί οργανισμοί και από τις επιχειρήσεις και από τον δημόσιο τομέα μπορεί να υποβάλουν αίτηση για καθεστώς σύνδεσης ώστε να συμμετέχουν στην ανάπτυξη ενός προτύπου, ή να ενημερωθούν για αυτή τη διαδικασία. Αυτές οι "οργανώσεις σύνδεσμοι" γίνονται αποδεκτές μέσω ψηφοφορίας από την αρμόδια επιτροπή ISO. Μπορούν να σχολιάσουν τα διαδοχικά σχέδια, να προτείνουν νέα στοιχεία εργασίας ή ακόμα και να προτείνει έγγραφα για γρήγορη μελέτη, αλλά δεν έχουν δικαίωμα ψήφου.

Πώς αναπτύσσονται τα πρότυπα ISO

Οι εθνικές αντιπροσωπείες εμπειρογνομόνων συναντιούνται για να συζητήσουν μέχρι να καταλήξουν σε συμφωνία σχετικά με ένα προσχέδιο προτύπου. Οι "συνεργαζόμενες οργανώσεις" συμμετέχουν επίσης σε αυτό το έργο. Σε ορισμένες περιπτώσεις, η προηγμένη εργασία εντός αυτών των οργανώσεων που σημαίνει ότι μια σημαντική τεχνική εξέλιξη και η αντίστοιχη συζήτηση έχει ήδη συμβεί, οδηγεί σε κάποια διεθνή αναγνώριση και σε αυτή την περίπτωση, ένα έγγραφο μπορεί να υποβληθεί για "ταχεία" επεξεργασία. Σε αμφότερες τις περιπτώσεις, το έγγραφο που προέκυψε κυκλοφορεί ως Σχέδιο Διεθνούς Προτύπου (DIS) στους φορείς μέλη ISO για ψηφοφορία και σχόλια.

Αν η ψηφοφορία είναι υπέρ, το έγγραφο, με τις ενδεχόμενες τροποποιήσεις, διανέμεται στα μέλη ISO, ως τελικό Σχέδιο Διεθνούς Προτύπου (FDIS). Αν αυτή η ψήφος είναι θετική, το έγγραφο αυτό, στη συνέχεια, δημοσιεύεται ως διεθνές πρότυπο.

Κάθε εργάσιμη ημέρα του έτους, κατά μέσο όρο επτά ISO τεχνικές συνεδριάσεις λαμβάνουν χώρα σε όλο τον κόσμο. Μεταξύ των

συνεδριάσεων, οι εμπειρογνώμονες συνεχίζουν τις εργασίες ανάπτυξης προτύπων δια αλληλογραφίας. Όλο και περισσότερο, το έργο τους πραγματοποιείται με ηλεκτρονικά μέσα, γεγονός που επιταχύνει την ανάπτυξη των προτύπων και μειώνει τα κόστη ταξιδιών.

Η πρόσβαση του κοινού

Τα περισσότερα μέλη του ISO έχουν κάποια μορφή δημόσιας διαδικασίας προσφυγής για τη γνωστοποίηση θεμάτων εργασίας και προτύπων σε κάθε ενδιαφερόμενο. Τα μέλη του ISO λαμβάνουν στη συνέχεια υπόψη τυχόν σχόλια που λαμβάνουν κατά τη διαμόρφωση της θέσης τους σχετικά με το προτεινόμενο θέμα εργασίας ή στο πρότυπο σχέδιο.

Τα σχέδια προτύπων είναι επίσης διαθέσιμα προς πώληση στα ενδιαφερόμενα μέλη του κοινού που μπορεί να τροφοδοτήσει μέσω του μέλους του ISO στη χώρα τους. Το κοινό μπορεί να αγοράσει τα Διεθνή Πρότυπα ISO μέσω του Web Store ή μέσω των εθνικών μελών του ΔΟΖ.

Συναίνεση

Επειδή τα πρότυπα ISO είναι εθελοντικές συμφωνίες, θα πρέπει να βασίζονται σε μια ισχυρή συναίνεση της διεθνούς γνώμης εμπειρογνομόνων. Συναίνεση, η οποία απαιτεί την επίλυση των σημαντικών ενστάσεων, αποτελεί ουσιώδη αρχή. Παρόλο που είναι αναγκαίο για το τεχνικό έργο να υπάρχει πρόοδος γρήγορα, επαρκής χρόνος απαιτείται πριν από το στάδιο έγκρισης για τη συζήτηση, τη διαπραγμάτευση και την επίλυση των σημαντικών τεχνικών διαφωνιών.

Η «Συναίνεση» είναι επισήμως καθορισμένη (στο πρότυπο ISO / IEC Guide 2) ως «γενική συμφωνία, η οποία χαρακτηρίζεται από την απουσία διαρκών αντιθέσεων σε θέματα ουσίας από οποιοδήποτε σημαντικό μέρος του οικείου φορέα και από μια διαδικασία που συνδέεται με την αναζήτηση για να λάβει υπόψη τις απόψεις όλων των

ενδιαφερομένων μερών και να συμβιβάσουν τις αντικρουόμενες απόψεις». Επίσης επισημαίνεται ότι «Συναίνεση δεν χρειάζεται να σημαίνει ομοφωνία».

Ψηφοφορία

Για να γίνει ένα έγγραφο δεκτό ως διεθνές πρότυπο ISO, θα πρέπει να εγκριθεί από τουλάχιστον δύο τρίτα των εθνικών μελών ISO που συμμετείχαν στην ανάπτυξή της και να μην απορριφθεί από περισσότερο από το ένα τέταρτο όλων των μελών του ISO που συμμετείχαν στην ψηφοφορία.

Προσφυγές

Οι εθνικοί φορείς-μέλη ISO έχουν το δικαίωμα προσφυγής σε μια μητρική τεχνική επιτροπή για την απόφαση της υποεπιτροπής, στο τεχνικό διοικητικό συμβούλιο μετά από απόφαση της τεχνικής επιτροπής και με το πρότυπο Συμβούλιο ISO σχετικά με την απόφαση του τεχνικού διοικητικού συμβουλίου. Οι προσφυγές μπορούν να αφορούν σε διαδικαστικά, τεχνικά ή διοικητικά θέματα.

Παραδοτέα ISO

Εκτός από τα διεθνή πρότυπα, ο οργανισμός ISO μπορεί επίσης να προσφέρει άλλες μορφές κανονιστικών συμφωνιών για καταστάσεις στις οποίες η ταχεία δημοσίευση είναι σημαντική, και έχει αναπτύξει μια σχηματική αναπαράσταση των διαφόρων τύπων του παραδοτέου που είναι διαθέσιμοι.

Λεπτομερής περιγραφή των σταδίων ανάπτυξης των Διεθνών Προτύπων ISO γίνεται στη συνέχεια.

2.5.4. Στάδια Ανάπτυξης Διαδικασιών

Ένα διεθνές πρότυπο είναι το αποτέλεσμα μιας συμφωνίας μεταξύ του μελών φορέων του ISO. Μπορεί να χρησιμοποιούνται ως έχουν ή μπορούν να υλοποιηθούν μέσω ενσωμάτωσης στα εθνικά πρότυπα των διαφόρων χωρών.

Τα Διεθνή Πρότυπα αναπτύσσονται από τον οργανισμό ISO τεχνικές επιτροπές (Technical Committees - TC) και υποεπιτροπών (SubCommittees - SC) με μια διαδικασία που αποτελείται από έξι στάδια:

- **Στάδιο 1: Στάδιο Πρότασης**
- **Στάδιο 2: Φάση Προετοιμασίας**
- **Στάδιο 3: Στάδιο Επιτροπής**
- **Στάδιο 4: Στάδιο Έρευνας**
- **Στάδιο 5: Στάδιο Έγκρισης**
- **Στάδιο 6: Στάδιο Δημοσίευσης**

Εάν ένα έγγραφο με έναν ορισμένο βαθμό ωριμότητας είναι διαθέσιμο κατά την έναρξη ενός έργου τυποποίησης, για παράδειγμα, ένα πρότυπο που αναπτύχθηκε από άλλη οργάνωση, είναι δυνατό να παραλείπονται ορισμένα στάδια. Στη λεγόμενη «ταχεία διαδικασία», ένα έγγραφο υποβάλλεται άμεσα για έγκριση ως σχέδιο του Διεθνούς Προτύπου (Draft International Standard - DIS), στους οργανισμούς μέλη ISO (στάδιο 4) ή, εάν το έγγραφο έχει αναπτυχθεί από έναν αναγνωρισμένο διεθνή οργανισμό τυποποίησης από το Συμβούλιο ISO, ως ένα τελικό σχέδιο Διεθνούς Προτύπου (Final Draft International Standard - FDIS, στάδιο 5), χωρίς να διέρχονται από τα προηγούμενα στάδια.

Στη συνέχεια ακολουθεί μια περίληψη του καθενός από τα έξι στάδια:

2.5.4.1. Στάδιο 1: Στάδιο πρότασης

Το πρώτο βήμα για την ανάπτυξη ενός Διεθνούς Προτύπου είναι να επιβεβαιώσει ότι το συγκεκριμένο προτεινόμενο Διεθνές Πρότυπο είναι απαραίτητη. Μια νέα πρόταση στοιχείου εργασίας έχει υποβληθεί σε ψηφοφορία από τα μέλη της σχετικής τεχνικής επιτροπής (TC) ή υποεπιτροπής (SC) για να καθοριστεί η ένταξη του αντικειμένου εργασίας σε πρόγραμμα εργασίας.

Η πρόταση γίνεται δεκτή εάν η πλειοψηφία των μελών του TC / SC ψηφίσουν υπέρ και εάν τουλάχιστον πέντε μέλη δηλώνουν τη δέσμευσή τους να συμμετάσχουν ενεργά στο σχέδιο. Στη φάση αυτή ένας επικεφαλής του έργου διορίζεται και είναι υπεύθυνος για την ομάδα εργασίας.

2.5.4.2. Στάδιο 2: Φάση προετοιμασίας

Συνήθως, μια ομάδα εργασίας από εμπειρογνώμονες, ο πρόεδρος (σύνεδρος), εκ των οποίων είναι ο επικεφαλής του έργου, έχει συσταθεί από το TC / SC για την προετοιμασία ενός σχεδίου εργασίας. Διαδοχικά σχέδια εργασίας μπορεί να αναθεωρηθούν μέχρι η ομάδα εργασίας να είναι πεπεισμένη ότι έχει αναπτύξει την καλύτερη τεχνική λύση στο πρόβλημα που αντιμετωπίζεται. Σε αυτό το στάδιο, το σχέδιο διαβιβάζεται στην αρμόδια επιτροπή της μητρικής της ομάδας εργασίας για τη φάση της οικοδόμησης συναίνεσης.

2.5.4.3. Στάδιο 3: Στάδιο επιτροπής

Μόλις ένα πρώτο σχέδιο της επιτροπής είναι διαθέσιμο, έχει καταχωρηθεί από την Κεντρική Γραμματεία προτύπων ISO. Διανέμεται για σχολιασμό και, εφόσον απαιτείται, η ψηφοφορία, από τα μέλη του

TC / SC. Διαδοχικά σχέδια της επιτροπής μπορεί να αναθεωρηθούν έως ότου επιτευχθεί συναίνεση σχετικά με το τεχνικό περιεχόμενο. Μετά την επίτευξη της συναίνεσης, διατυπώνεται οριστικά το κείμενο για την υποβολή ως σχέδιο του Διεθνούς Προτύπου (DIS).

2.5.4.4. Στάδιο 4: Στάδιο έρευνας

Το σχέδιο του Διεθνούς Προτύπου (DIS) διανέμεται σε όλους τους φορείς - μέλη ISO από την Κεντρική Γραμματεία προτύπων ISO για ψηφοφορία και σχολιασμό εντός προθεσμίας πέντε μηνών. Έχει εγκριθεί για την υποβολή ως τελικό σχέδιο Διεθνούς Προτύπου (FDIS) εάν μια πλειοψηφία των δύο τρίτων των μελών του TC / SC είναι υπέρ και όχι περισσότερο από το ένα τέταρτο του συνολικού αριθμού των ψήφων είναι αρνητικά. Εάν δεν πληρούνται τα κριτήρια έγκρισης, το κείμενο επιστρέφεται στην αρχική TC / SC για περαιτέρω μελέτη και ένα αναθεωρημένο έγγραφο θα είναι και πάλι να δοθεί για ψηφοφορία ώστε να διατυπωθεί ένα σχέδιο διεθνούς προτύπου.

2.5.4.5 Στάδιο 5: Στάδιο έγκρισης

Το τελικό σχέδιο Διεθνούς Προτύπου (FDIS) κοινοποιείται σε όλα τα μέλη ISO από την Κεντρική Γραμματεία προτύπων ISO για την τελική ψηφοφορία με ναι ή όχι, εντός προθεσμίας δύο μηνών. Αν λαμβάνονται τεχνικές παρατηρήσεις κατά τη διάρκεια αυτής της περιόδου, αυτές δεν συνεκτιμώνται πλέον σε αυτό το στάδιο, αλλά είναι εγγεγραμμένοι για να εξεταστεί κατά τη μελλοντική αναθεώρηση του Διεθνούς Προτύπου. Το κείμενο έχει εγκριθεί ως διεθνές πρότυπο, και απαιτείται πλειοψηφία των δύο τρίτων των μελών του TC / SC είναι υπέρ και όχι περισσότερο από το ένα τέταρτο του συνολικού αριθμού των ψήφων είναι αρνητικά. Αν δεν πληρούνται αυτά τα κριτήρια έγκρισης, το πρότυπο παραπέμπεται στην αρχική TC / SC για επανεξέταση συνοδευόμενη με

τα σχόλια που προσκομίζονται προς υποστήριξη των αρνητικών ψήφων που έλαβε.

2.5.4.6 Στάδιο 6: Στάδιο δημοσίευσης

Μόλις ένα τελικό σχέδιο διεθνούς προτύπου έχει εγκριθεί μόνο ήσσονος σημασίας συντακτικές τροποποιήσεις, εάν και όπου απαιτείται, προστίθενται στο τελικό κείμενο. Το τελικό κείμενο στέλνεται από την Κεντρική Γραμματεία προτύπων ISO, η οποία δημοσιεύει το Διεθνές Πρότυπο.

Αναθεώρηση των Διεθνών Προτύπων (Επιβεβαίωση, Αναθεώρηση, Απόσυρση)

Όλα τα Διεθνή Πρότυπα επανεξετάζονται τουλάχιστον τρία έτη μετά τη δημοσίευση και κάθε πέντε χρόνια μετά την πρώτη εξέταση από όλους τους φορείς μέλη του ISO. Η πλειοψηφία των μελών του TC / SC αποφασίζει αν ένα Διεθνές Πρότυπο πρέπει να επιβεβαιωθεί, αναθεωρηθεί ή να αποσυρθεί.

3. Υπηρεσίες Διαδικτύου

3.1. Γενικά

Σήμερα είναι διαθέσιμες πάρα πολλές υπηρεσίες που μπορούν και παρέχονται μέσω διαδικτύου από χιλιάδες εταιρίες, αλλά και υπηρεσίες που αφορούν το ίδιο διαδίκτυο, όπως οι υπηρεσίες σύνδεσης με το διαδίκτυο ή υπηρεσίες κατασκευής και φιλοξενίας διαδικτυακών τόπων. Το διαδίκτυο αποτελεί για τις επιχειρήσεις και στην Ελλάδα πλέον ένα μέσο προβολής και δημοσιοποίησης των δραστηριοτήτων τους. Τα τελευταία χρόνια όμως πολλές επιχειρήσεις παγκοσμίως άρχισαν να χρησιμοποιούν το Internet και ως μέσο παροχής υπηρεσιών προς τους πελάτες τους αλλά και προς άλλες επιχειρήσεις.

Οι υπηρεσίες στο διαδίκτυο είναι μια αναπτυσσόμενη και δυναμική βιομηχανία σε όλους τους κλάδους της που γνωρίζουν τεράστια ανάπτυξη. Είναι ένας τομέας της οικονομίας που δεν χρειάζεται απαραίτητα μεγάλες επενδύσεις, αλλά οραματισμό, γνώση και πολύ δουλειά. Είναι μια μοναδική ευκαιρία που δίνεται και σε εταιρίες από την Ελλάδα να αντιληφθούν τα μηνύματα των καιρών και να πάρουν μια καλή θέση στον διεθνή ανταγωνισμό.

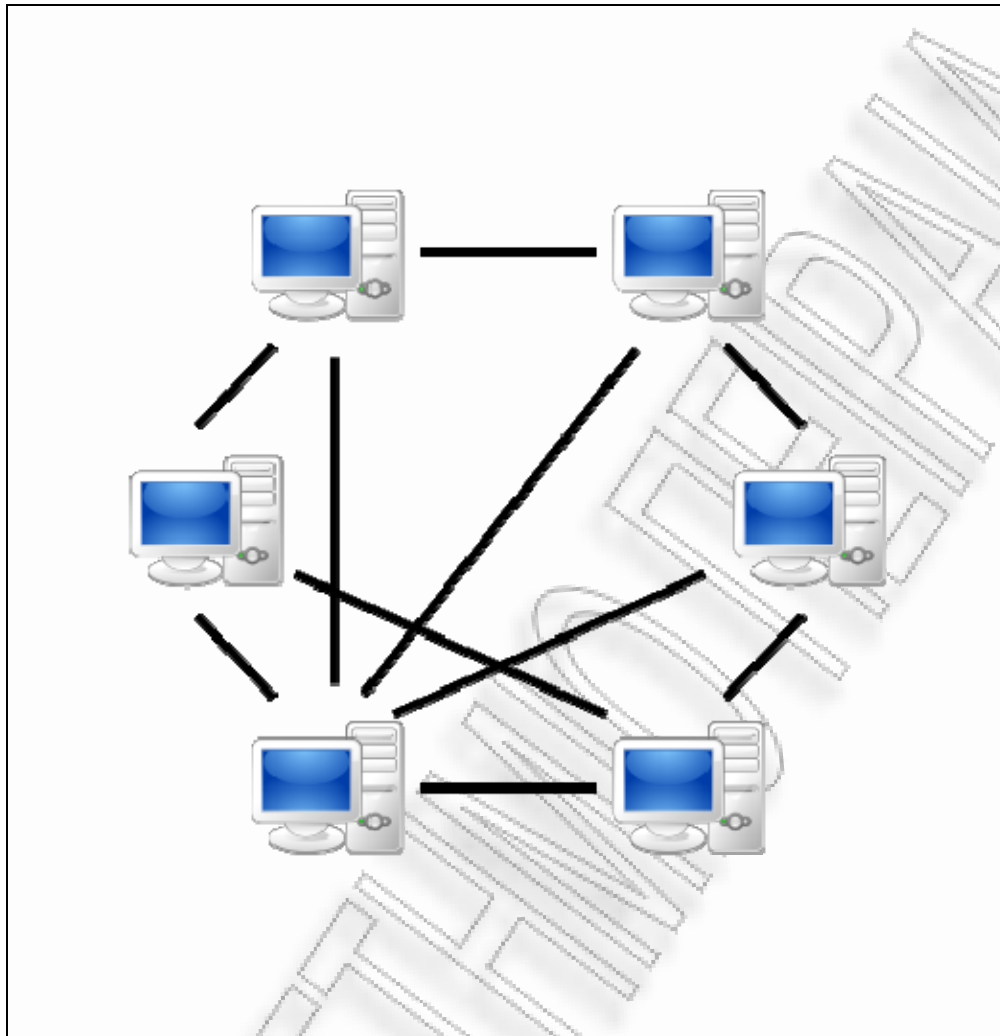
Η εξέλιξη αυτή είχε σαν επακόλουθο την δημιουργία ενός συνόλου από θέματα που είχαν να κάνουν με την ευχρηστία, τη λειτουργικότητα και την απλότητα των παρεχόμενων υπηρεσιών. Και ενώ οι προσπάθειες που έγιναν για να δοθούν λύσεις ήταν πολλές, το αποτέλεσμα ήταν το ίδιο: υπήρχε ένα μεγάλο σύνολο από έτοιμες υπηρεσίες στο Internet αλλά η χρησιμοποίησή τους ήταν πάρα πολύ μικρή.

3.2. Αρχιτεκτονικές Εφαρμογών Διαδικτύου

3.2.1. Peer to Peer

3.2.1.1. Γενική Περιγραφή

Ένα δίκτυο υπολογιστών peer-to-peer (ή P2P) είναι ένα δίκτυο που δίνει τη δυνατότητα σε δύο ή περισσότερους υπολογιστές να μοιράζονται τους διαθέσιμους υπολογιστικούς πόρους τους ισοδύναμα. Ένα τέτοιο δίκτυο χρησιμοποιεί την διαθέσιμη επεξεργαστική ισχύ, τον αποθηκευτικό χώρο και το εύρος ζώνης (bandwidth) όλων των κόμβων. Οι κόμβοι του δικτύου έχουν ίσα δικαιώματα και υποχρεώσεις. Οι πληροφορίες που βρίσκονται σε έναν από τους κόμβους, ανάλογα με τα δικαιώματα που καθορίζονται, μπορούν να διαβαστούν από τους υπόλοιπους ανάλογα με την πολιτική ασφάλειας και αντίστροφα. [16] – [28].



Εικόνα 3.1 Αρχιτεκτονική Peer to Peer

3.2.1.2. Κατηγορίες Δικτύων Peer-to-Peer

Τα Peer-to-Peer δίκτυα κατανέμονται ανάλογα με τα χαρακτηριστικά τους στις εξής τρεις κατηγορίες:

- **Συγκεντρωτικά P2P δίκτυα**

Πολλοί, όταν αναφέρονται σε αυτά, χρησιμοποιούν τη φράση «πρώτης γενιάς P2P δίκτυα». Εδώ, υπάρχει ένας κεντρικός Index Server στον οποίο αποθηκεύονται οι πληροφορίες για τα περιεχόμενα των καταλόγων που οι συμμετέχοντες επιθυμούν να μοιράζονται. Οι χρήστες μπορούν να αναζητήσουν στους Index Servers αυτούς τα αρχεία που ψάχνουν, χρησιμοποιώντας ένα κατάλληλο πρόγραμμα-πελάτη. Όταν το

αρχείο βρεθεί, ανοίγει μια σύνδεση μεταξύ των δύο χρηστών για τη μεταφορά του. Σε αυτή τη κατηγορία ανήκουν το Napster το DC++ και το WinMX.

- **Αποκεντρωτικά P2P δίκτυα**

Η φιλοσοφία εδώ είναι εντελώς διαφορετική. Κάθε σύστημα που συμμετέχει αποτελεί ταυτόχρονα client και server (ή αλλιώς servent). Μόλις κάποιος συνδεθεί μέσω ενός ανάλογου προγράμματος-πελάτη P2P, κάνει γνωστή την παρουσία του σε ένα μικρό αριθμό υπολογιστών ήδη συνδεδεμένων οι οποίοι με τη σειρά τους προωθούν τη δήλωση παρουσίας του σε ένα μεγαλύτερο δίκτυο υπολογιστών κ.λ.π. Πλέον ο χρήστης έχει τη δυνατότητα να αναζητήσει οποιαδήποτε πληροφορία μεταξύ των διαμοιραζόμενων αρχείων. Τα δίκτυα αυτά λέγονται και δεύτερης γενιάς. Η μεταφορά των αρχείων είναι όμοια με αυτή των αποκεντρωτικών P2P δικτύων. Σε αυτή τη κατηγορία ανήκουν το Kazaa, το Gnutella και το BearShare.

- **P2P δίκτυα τρίτης γενιάς**

Είναι αυτά τα οποία διαθέτουν χαρακτηριστικά ανωνυμίας όπως το Freenet, το I2P και το Entropy. Είναι αποκεντρωτικού τύπου και η φιλοσοφία του βασίζεται εκτός από την ανωνυμία, στην υψηλή βιωσιμότητα του, στο συνεχή διαμοιρασμό των αρχείων και στην κωδικοποίησή τους έτσι ώστε κανείς να μην μπορέσει ποτέ να αποκτήσει κανένα είδος ελέγχου πάνω σε αυτό. Τα δίκτυα αυτού του τύπου είναι υπό ανάπτυξη και έχουν χαρακτηριστεί ως μικρά παγκόσμια δίκτυα.

3.2.1.3. Χρήση των Peer-to-Peer δικτύων

Η χρήση δικτύων Peer to Peer εξασφαλίζει την επικοινωνία χρηστών από όλο τον κόσμο λειτουργώντας χωρίς λογοκρισία, ελέγχους ή φραγμούς και έτσι προάγεται και υποστηρίζεται η βασική ιδέα της δημιουργίας και λειτουργίας του παγκοσμίου ιστού: η ελεύθερη διακίνηση ιδεών και τη δωρεάν παροχή υπηρεσιών και πληροφοριών.

Η απλουστευμένη δομή, το πολύ χαμηλό κόστος και η ακανόνιστη ροή πληροφορίας είναι τα στοιχεία που χαρακτηρίζουν τη λειτουργία των P2P δικτύων. Η φιλοσοφία τους δίνει τη δυνατότητα στους χρήστες / συμμετέχοντες να δημιουργούν δυναμικά αναπτυσσόμενες περιοχές, το περιεχόμενο των οποίων καθορίζεται από τους ίδιους τους χρήστες.

Παράλληλα, η προαναφερθείσα δομή των P2P δικτύων καθιστά δυνατή την αντιγραφή και διανομή αρχείων μεταξύ χρηστών, τα οποία μπορεί να προστατεύονται από πνευματικά δικαιώματα, όπως ταινίες, λογισμικό ή/και τραγούδια, χωρίς τη συναίνεση του κατόχου των πνευματικών αυτών δικαιωμάτων. Η ευρεία χρήση των δικτύων p2p για αυτόν τον σκοπό συντέλεσε ώστε τα δίκτυα να ταυτιστούν με έννοιες όπως «πειρατεία» και «παρανομία» και να διώκονται τόσο τα ίδια όσο και οι δημιουργοί / διαχειριστές τους, όσο και οι χρήστες τους. Ειδικά στα συγκεντρωτικά p2p δίκτυα όπως το Napster η κατηγορία που προβαλλόταν ήταν ότι η μεσολάβηση μεταξύ των χρηστών και η αποθήκευση στον κεντρικό server της εταιρίας των στοιχείων που ήταν απαραίτητα για την ανταλλαγή αρχείων ήταν αρκετή, για να στοιχειοθετηθεί συμμετοχή της εταιρίας που διαχειριζόταν το δίκτυο στην παραβίαση της πνευματικής ιδιοκτησίας που τελούσαν οι χρήστες. Στην υπόθεση Napster στην Αμερική η ομώνυμη εταιρία καταδικάστηκε να καταβάλει υψηλές αποζημιώσεις σε πνευματικούς δημιουργούς και κατόχους πνευματικών δικαιωμάτων ως συνεργός στην παραβίαση των δικαιωμάτων τους που τελούσαν οι χρήστες, ανταλλάσσοντας παράνομα

αντίγραφα μεταξύ τους. Σαν αντίδραση σε αυτήν τη νομολογία αναπτύχθηκαν τα αποκεντρωτικά συστήματα p2p, στα οποία δεν υπάρχει κεντρικός server που να αποθηκεύει την οποιαδήποτε πληροφορία σχετικά με τα ανταλλασσόμενα αρχεία, παρά μόνο ένα λογισμικό, το οποίο επιτρέπει τη διασύνδεση των υπολογιστών των τελικών χρηστών πρόσφατη προσπάθεια πανεπιστημίων και ερευνητικών κέντρων για τη δημιουργία μιας εφαρμογής για P2P δίκτυα. Το εγχείρημα αυτό γνωστό και ως LionShare βασίζεται στα δεύτερης γενιάς P2P δίκτυα και φτιάχνεται για το διαμοιρασμό στους χρήστες τους ακαδημαϊκού υλικού.

3.2.2. Repository Model

3.2.2.1. Εισαγωγή

Πολλά πολιτιστικά και εκπαιδευτικά ιδρύματα βρίσκονται στη διαδικασία της επιλογής ή ανάπτυξης αποθετηρίων για να υποστηρίξουν ένα ευρύ φάσμα ψηφιακών δραστηριοτήτων, συμπεριλαμβανομένης της διαχείρισης περιεχομένου, την υποβολή, αναζήτηση, αρχειοθέτηση, δημοσίευση, την πρόσβαση και τη συντήρηση. Επιπλέον, υπάρχει μια αυξανόμενη έμφαση στην ανάπτυξη των συστημάτων που υποστηρίζουν περιεχόμενο μεταβλητού προορισμού και παροχή ενός ευρέος φάσματος υπηρεσιών διαδικτύου. Στην ενότητα αυτή θα περιγραφούν οι στρατηγικές που ακολουθούνται για την κάλυψη των υπαρχόντων απαιτήσεων με τη χρήση συστημάτων και λειτουργιών που ακολουθούν το μοντέλο αποθετηρίου. Η διαδικασία επιλογής του χώρος απόθεσης περιλαμβάνει διάφορα βασικά στάδια συμπεριλαμβανομένης της ανάλυσης των ενδιαφερόμενων μερών, την αξιολόγηση των αναγκών, τον ορισμό των υπηρεσιών, και τον προσδιορισμό των περιπτώσεων χρήσης και θέματα διαχείρισης. Επίσης, είναι απαραίτητο να λάβει χώρα λεπτομερής ανάλυση των πόρων καθώς είναι προϋπόθεση για να εξασφαλίσει τη βιωσιμότητα της ανάπτυξης, της διαχείρισης καθώς και δραστηριοτήτων αξιολόγησης. Εξίσου σημαντική είναι και η κουλτούρα

εργασίας και οι πρακτικές του αποθηκευτικού χώρου από τα ενδιαφερόμενα μέρη. Η τελική επιτυχία ενός αποθηκευτικού χώρου εφαρμογών καθορίζεται συχνά από το πόσο καλά θα υποστηρίζει τις οργανωτικές διαδικασίες, τις διαχειριστικές πολιτικές και τις συνεργασίες με άλλα συστήματα. [7] – [15].

3.2.2.2. Σκοποί του Αποθετήρια

Ο βασικοί λόγοι ύπαρξης και αποδοτικής λειτουργίας των δικτύων που βασίζονται στο μοντέλο αποθετηρίου είναι ο εξής:

- Ενεργοποίηση της ψηφιακής διαχείρισης υπολογιστικών και όχι μόνο πόρων
- Υπηρεσίες εξασφάλισης προσφοράς
- Παροχή εξασφάλισης βιωσιμότητας μέσω της πρόσβασης σε συλλογική πνευματική εργασία
- Υποστήριξη της μάθησης, της διδασκαλίας και της έρευνας
- Παροχή υπηρεσιών εύρεσης περιεχομένου
- Υποστήριξη επαναχρησιμοποιησιμότητας και αλλαγής χρήσης του περιεχομένου.
- Υποστήριξη αρχείων επιχειρηματικών απαιτήσεων.
- Παροχή εναλλακτικών δικτύων για την υποστήριξη της επικοινωνίας μεταξύ της επιστημονικής κοινότητας.
 - Οργάνωση της πληροφορίας ώστε να είναι δυνατή η αποτελεσματική διαχείριση περιεχομένου και πρόσβασης σε αυτό.
- Να παρέχουν πρόσβαση σε αποτελέσματα των δημόσια χρηματοδοτούμενων ερευνητικών πρωτοβουλιών

- Ενίσχυση της συνεργασίας μεταξύ των δημιουργών περιεχομένου και παρόχων με τους διαχειριστές του περιεχομένου.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΡΑΙΑ

3.2.3. Client - Server

3.2.3.1. Γενική Περιγραφή

Η αρχιτεκτονική client-server αναφέρεται σε μια βασική αλλαγή στην αντιμετώπιση των υπολογιστών: την αλλαγή από συστήματα που βασίζονται στα μηχανήματα σε συστήματα που βασίζονται και επικεντρώνονται στον χρήστη.

Πιο συγκεκριμένα, ένα σύστημα client-server είναι ένα σύστημα στο οποίο το δίκτυο ενώνει διάφορους υπολογιστικούς πόρους (resources), ώστε οι πελάτες (clients ή αλλιώς front end) να μπορούν να ζητούν και να λαμβάνουν υπηρεσίες από έναν εξυπηρετητή (server ή αλλιώς back end), ο οποίος παρέχει τις ζητούμενες υπηρεσίες.

Αναλυτικότερα, στο client-server μοντέλο, ο client στέλνει μια αίτηση και ο server επιστρέφει μια απάντηση ή κάνει μια σειρά από ενέργειες. Ο server μπορεί να ενεργοποιείται άμεσα για την αίτηση αυτή ή να τοποθετεί την αίτηση σε μια ουρά. Η άμεση ενεργοποίηση για την αίτηση μπορεί, για παράδειγμα, να σημαίνει ότι ο server υπολογίζει έναν αριθμό και τον επιστρέφει αμέσως στον client. Η τοποθέτηση της αίτησης σε μια ουρά μπορεί να σημαίνει ότι η αίτηση πρέπει να τεθεί σε αναμονή για να εξυπηρετηθεί. Ένα καλό παράδειγμα για αυτό είναι όταν εκτυπώνουμε ένα κείμενο σε ένα εκτυπωτή δικτύου. Ο server τοποθετεί την αίτηση σε μια ουρά μαζί με αιτήσεις εκτυπώσεων και από άλλους clients. Μετά επεξεργάζεται την αίτηση με βάση την σειρά προτεραιότητας, η οποία, σε αυτή την περίπτωση, καθορίζεται από τη σειρά με την οποία ο server παρέλαβε την απαίτηση.

Η αρχιτεκτονική client-server παρέχει τα εξής πολύ σημαντικά πλεονεκτήματα:

- Αποτελεσματική χρήση της υπολογιστικής ισχύος του εξυπηρετητή.

- Μείωση του κόστους συντήρησης, δημιουργώντας συστήματα client-server που απαιτούν λιγότερη συντήρηση και κοστίζουν λιγότερο στην αναβάθμιση.
- Αύξηση της παραγωγικότητας, προσφέροντας στους χρήστες ξεκάθαρη πρόσβαση στις αναγκαίες πληροφορίες μέσω σταθερών και εύκολων στην χρήση διασυνδέσεων.
- Αύξηση της ευελιξίας και της δυνατότητας δημιουργίας συστημάτων που υποστηρίζουν πολλά περιβάλλοντα.

Αναγνωρίζοντας τα προαναφερθέντα προτερήματα, οι διάφοροι οργανισμοί κινούνται προς την κατεύθυνση των client-server αρχιτεκτονικών και τεχνολογιών, αυξάνοντας κατά πολύ την ανταγωνιστικότητά τους. Στην Εικόνα 3.1 φαίνεται μια γραφική αναπαράσταση της αρχιτεκτονικής Client – Server.

3.2.3.2. Λειτουργία

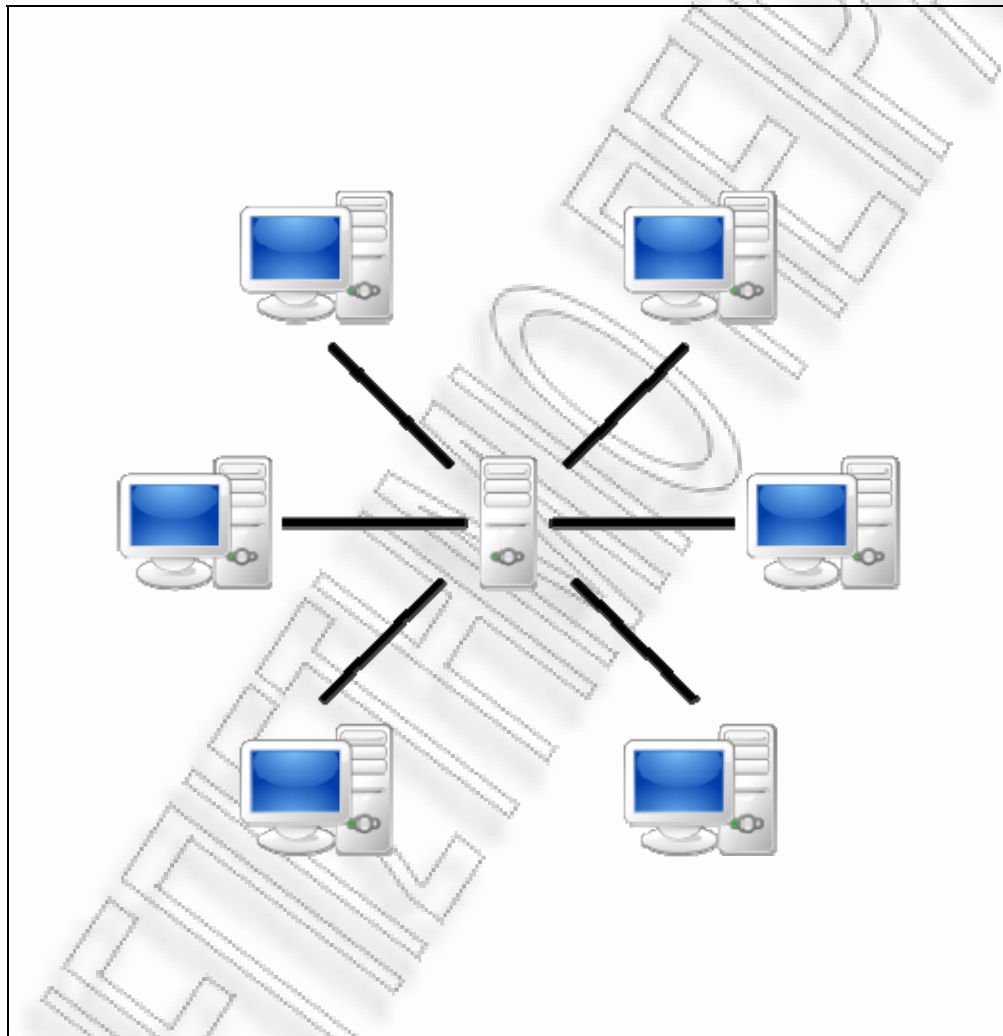
Η πλευρά του client πρώτα στέλνει ένα μήνυμα για να καλέσει σε ετοιμότητα τον server. Από τη στιγμή που ο client και ο server έχουν επικοινωνία μεταξύ τους, ο client μπορεί να υποβάλλει την αιτήσή του.

Ο client είναι ο αιτών των υπηρεσιών. Ο client δεν μπορεί παρά να είναι ένας υπολογιστής. Οι υπηρεσίες που ζητούνται από τον client μπορεί να υπάρχουν στους ίδιους σταθμούς εργασίας ή σε απομακρυσμένους σταθμούς εργασίας που συνδέονται μεταξύ τους μέσω ενός δικτύου. Ο client ξεκινάει πάντα την επικοινωνία.

- Τα συστατικά του client είναι πολύ απλά. Μια client μηχανή πρέπει να μπορεί να κάνει τα ακόλουθα:
- Να τρέχει το λογισμικό των γραφικών διεπαφών χρηστών (GUIs).
- Να δημιουργεί τις αιτήσεις για πληροφορίες και να τις στέλνει στον server.

- Να αποθηκεύει τις επιστρεφόμενες πληροφορίες.

Αυτές οι αιτήσεις καθορίζουν πόση μνήμη χρειάζεται, ποια ταχύτητα επεξεργασίας θα μπορούσε να βελτιώσει τον χρόνο ανταπόκρισης, και πόση χωρητικότητα αποθήκευσης απαιτείται.



Εικόνα 3.2 Αρχιτεκτονική Client - Server

Ο server απαντάει στις αιτήσεις που γίνονται από τους clients. Ένας client μπορεί να ενεργεί ως server εάν λαμβάνει και επεξεργάζεται αιτήσεις όπως ακριβώς και τις στέλνει (για παράδειγμα, ένας σταθμός εργασίας που χρησιμοποιείται και ως server εκτυπώσεων από άλλους). Οι server δεν ξεκινάνε τις επικοινωνίες -περιμένουν τις αιτήσεις των clients.

Φέρνοντας σαν παράδειγμα τον server εκτυπώσεων ενός δικτύου, ο client ζητάει από τον server να εκτυπώσει ένα κείμενο σε έναν συγκεκριμένο εκτυπωτή και ο server προσθέτει την εκτύπωση σε μια ουρά και ενημερώνει τον client όταν το κείμενο εκτυπωθεί επιτυχημένα. Η διαδικασία του client μπορεί να ανήκει φυσικά στον ίδιο σταθμό εργασίας με την διαδικασία του server. Στο παράδειγμα εδώ, μια εντολή εκτύπωσης μπορεί να εκδίδεται στον server του σταθμού εργασίας του δικτύου, χρησιμοποιώντας την διαδικασία του server εκτυπώσεων σε αυτόν τον σταθμό εργασίας.

Τα συστατικά του server είναι επίσης πολύ απλά. Μια server μηχανή πρέπει να μπορεί να κάνει τα ακόλουθα :

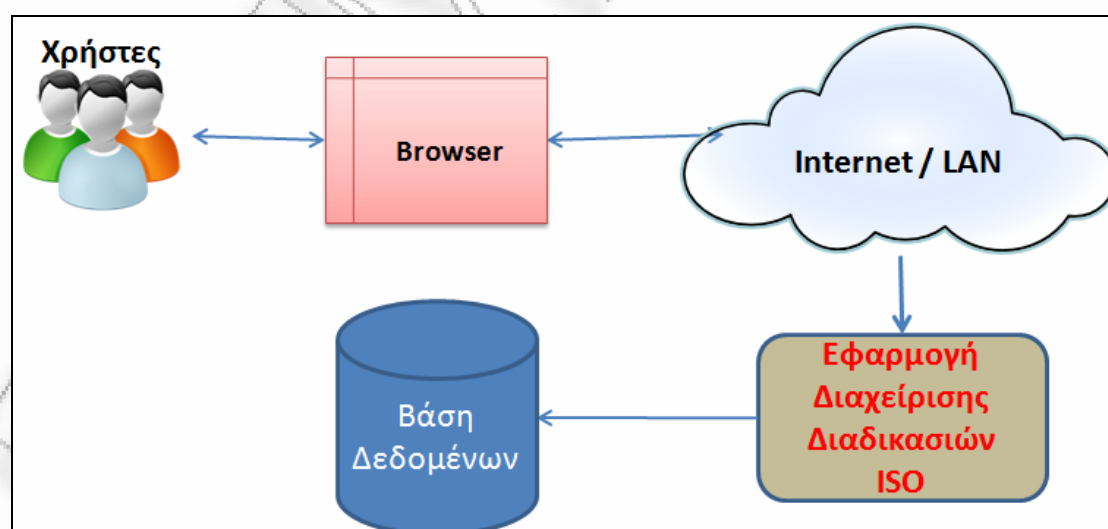
- Να αποθηκεύει, να ανακτά και να προστατεύει πληροφορίες.
- Να επιθεωρεί τις αιτήσεις των clients.
- Να δημιουργεί εφαρμογές διαχείρισης πληροφοριών, όπως δημιουργία αντιγράφων, ασφάλεια κτλ.
- Να διαχειρίζεται πληροφορίες.

4. Εφαρμογή Μηχανοργάνωσης Διαδικασιών ISO

4.1. Αρχιτεκτονική

Η αρχιτεκτονική του υπό μελέτη συστήματος βασίζεται στο μοντέλο πελάτη διακομιστή (client – server model) το οποίο περιγράφεται αναλυτικότερα στην παράγραφο 3.2.3.

Πιο συγκεκριμένα προδιαγράφεται η ύπαρξη ενός – απομακρυσμένου στη γενική περίπτωση – διακομιστή (server) ο οποίος φιλοξενεί την εφαρμογή καθώς και χωρίς βλάβη της γενικότητας την βάση δεδομένων. Φυσικά η βάση δεδομένων θα μπορούσε να βρίσκεται εγκατεστημένη οπουδήποτε υπάρχει δικτυακή πρόσβαση από τον διακομιστή, αλλά για λόγους απλότητας θεωρούμε ότι στον ίδιο server φιλοξενείται και η βάση δεδομένων. Ο διακομιστής αυτός παρέχει τη δυνατότητα στους απομακρυσμένους χρήστες του συστήματος να συνδέονται και να κάνουν χρήση των διαθέσιμων υπηρεσιών.



Εικόνα 4.1 Προτεινόμενη Αρχιτεκτονική

Η Εικόνα 4.1 απεικονίζει την εν συντομία προαναφερθείσα διάταξη, και περιλαμβάνει αναλυτικότερα τα εξής στοιχεία:

- **Εφαρμογή Διαχείρισης Διαδικασιών ISO:** Η εφαρμογή διαχείρισης διαδικασιών ISO θα αναλαμβάνει την καταχώρηση, παρουσίαση, τροποποίηση και συσχέτιση εγγράφων και διαδικασιών ISO.
- **Βάση Δεδομένων:** Στη σχεσιακή βάση δεδομένων θα καταχωρούνται τα σχετικά με την εφαρμογή αντικείμενα (έγγραφα, στοιχεία τροποποίησης, κλπ).
- **Δικτυακή Υποδομή (Internet - LAN):** Η δικτυακή πρόσβαση στην εφαρμογή θα μπορεί να γίνεται μέσω τοπικού δικτύου και μέσω διαδικτύου, ανάλογα με τις πολιτικές ασφάλειας της εκάστοτε εταιρείας που αφορά την εφαρμογή.
- **Περιηγητής (Browser) :** Το βασικό μέσο πρόσβασης στην εφαρμογή θα είναι οποιοδήποτε πρόγραμμα περιηγητή στο διαδίκτυο.
- **Χρήστες Εφαρμογής:** Προδιαγράφεται αυτόνομο σύστημα πιστοποίησης χρηστών και καθορισμός ξεχωριστών επιπέδων δικαιωμάτων πρόσβασης στις διάφορες ενότητες της εφαρμογής.

4.2. Σχεδιασμός Υλοποίησης

Στα πλαίσια του σχεδιασμού μιας web εφαρμογής, επελέγη να ακολουθηθεί το αρχιτεκτονικό μοντέλο MVC (Model – View Controller). Στα πλαίσια αυτής της αρχιτεκτονικής γίνεται διαχωρισμός μεταξύ ρόλων στους παρακάτω:

- **Model:** Το μοντέλο αντικειμένων (για αντικειμενοστραφείς γλώσσες προγραμματισμού) που αντιπροσωπεύει τα δεδομένα της βάσης. Κατά τη λειτουργία της εφαρμογής τα αντικείμενα αυτά (objects) μεταφέρουν τα δεδομένα από και προς τη βάση.
- **View:** Η παρουσίαση των αποτελεσμάτων διεργασιών που διεξάγονται από την εφαρμογή προς το χρήστη μέσω της κατάλληλης διεπαφής. Μέσω της ίδιας διεπαφής ο χρήστης εισάγει όποια δεδομένα απαιτείται από την εφαρμογή.
- **Controller:** Ο «ελεγκτής» των λειτουργιών της εφαρμογής. Στο κομμάτι αυτό υλοποιείται η «λογική» της λειτουργικότητας που παρέχει η εφαρμογή.

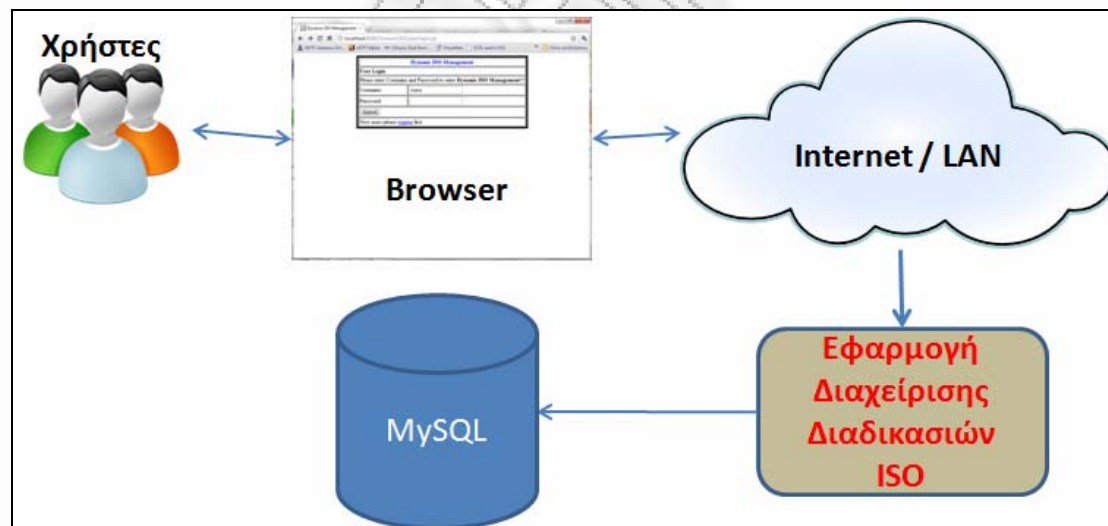
Οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση αυτού του συστήματος διακρίνονται σε τρεις βασικές ενότητες:

- Το σύστημα σχεσιακής βάσης δεδομένων (RDBMS)
- Την μοντελοποίηση του RDBMS συστήματος
- Την εφαρμογή με τη λογική λειτουργικότητας (business logic)
- Το επίπεδο παρουσίασης της εφαρμογής – διεπαφής με τον χρήστη (user interface)

Η βάση δεδομένων που χρησιμοποιήθηκε ήταν η MySQL της εταιρείας Oracle Corporation, ενώ για την μοντελοποίηση και αντιστοίχιση των δεδομένων της βάσης δεδομένων με την εφαρμογή έγινε χρήση του πακέτου ανοικτού λογισμικού Hibernate. Το Hibernate χρησιμοποιήθηκε και για τη διαχείριση των συνδέσεων στη βάση δεδομένων.

Στο επίπεδο της υλοποίησης επελέγη η χρήση της αντικειμενοστραφούς γλώσσας προγραμματισμού Java της εταιρείας Oracle-Sun Microsystems, σε συνδυασμό με μια βάση δεδομένων MySQL της εταιρείας Oracle. Για την παρουσίαση και την υλοποίηση της λειτουργικότητας επελέγη η χρήση της j2ee τεχνολογίας της Java JSP (Java Server Pages).

Στην Εικόνα 4.2 φαίνεται η αντιστοίχιση των επιλεγόμενων τεχνολογιών με την προδιαγεγραμμένη αρχιτεκτονική (εικόνα 4.1)



Εικόνα 4.2 Διάταξη Υλοποίησης

4.3. Πρότυπη Υλοποίηση

4.3.1. Βάση Δεδομένων

Για τη σύνδεση της εφαρμογής με τη βάση δεδομένων γίνεται χρήση του πακέτου λογισμικού Hibernate [2] και μέσω αυτού γίνεται το διάβασμα, η ανανέωση και η εισαγωγή νέων δεδομένων στη βάση.

Για τις διαχειριστικές λειτουργίες στη βάση δεδομένων έγινε χρήση της εφαρμογής `phrmyadmin` [3], μιας διαδικτυακής πλατφόρμας βασισμένης στην γλώσσα `php`. Η εφαρμογή αυτή τρέχει μέσα σε ένα `apache web server` [4].

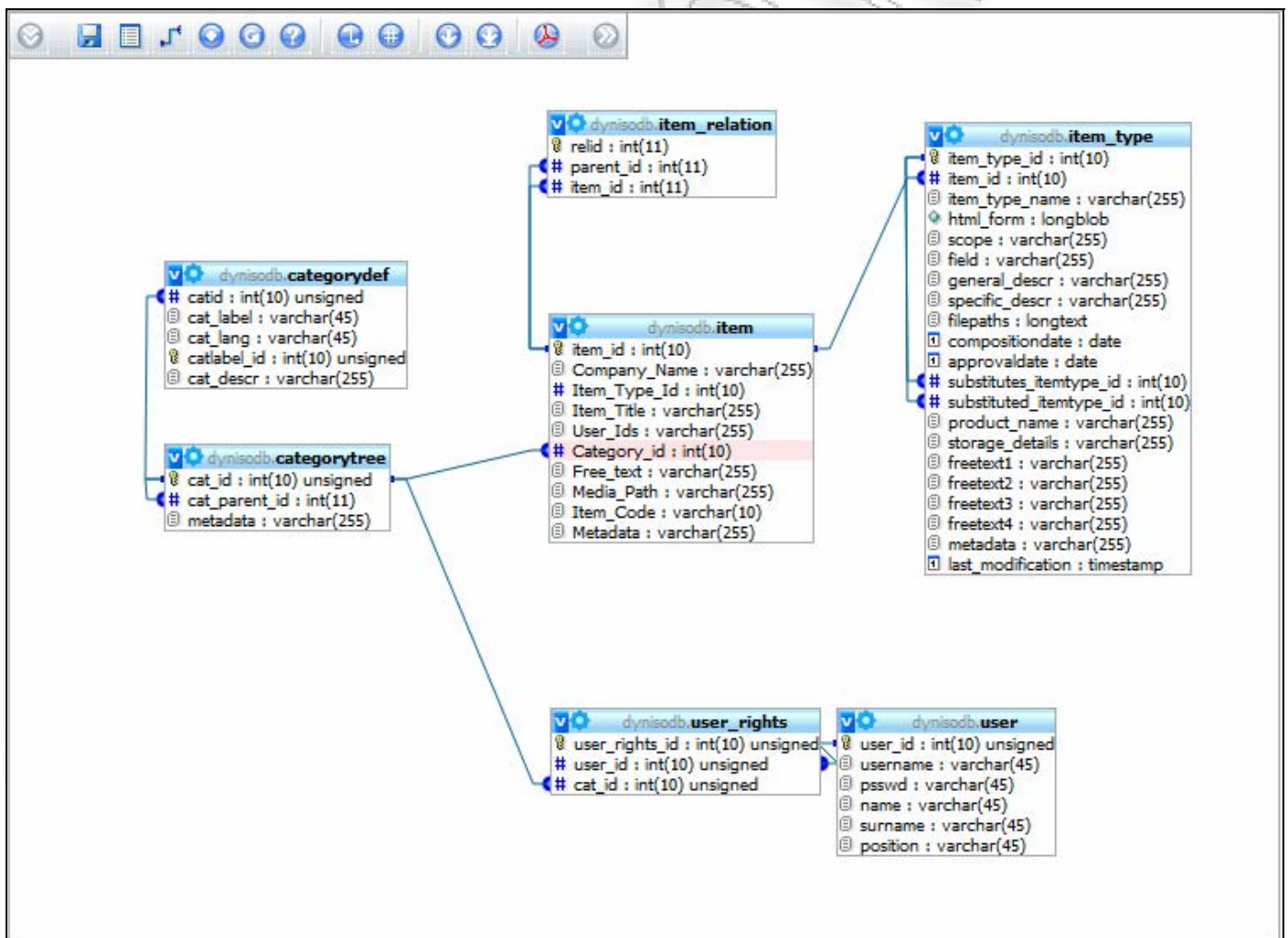
Στη σχεσιακή βάση δεδομένων υπάρχουν οι παρακάτω πίνακες, για τους οποίους γίνεται μια σύντομη περιγραφή της λειτουργικότητας παρακάτω

- **Πίνακας `Item`:** Στον πίνακα αυτό αποθηκεύονται όλα τα σχετικά με τις διαδικασίες ISO αντικείμενα, μαζί με τα αντίστοιχα συσχετιζόμενα έγγραφα. Εδώ αποθηκεύονται και όλες οι εκδόσεις των εγγράφων σε ξεχωριστές εγγραφές οι οποίες συσχετίζονται.
- **Πίνακας `Item_relation`:** Στον πίνακα αυτόν καταχωρούνται οι συσχετίσεις μεταξύ των αντικειμένων. Έγγραφο από όλες τις δυνατές κατηγορίες διαδικασιών ISO μπορεί να συσχετιστούν μέσω της αντίστοιχης λειτουργικότητας της εφαρμογής και καταχωρούνται σε αυτόν τον πίνακα.
- **Πίνακας `Item_type`:** Στον πίνακα αυτόν καταχωρούνται οι τύποι των αντικειμένων.
- **Πίνακας `Categorydef`:** Στον πίνακα αυτόν καταχωρούνται οι διαθέσιμες κατηγορίες εγγράφων και διαδικασιών ISO.
- **Πίνακας `Categorytree`:** Στον πίνακα αυτόν καταχωρείται η δενδροειδής δομή των κατηγοριών.

- **Πίνακας User:** Στον πίνακα αυτόν καταχωρούνται οι διαφορετικοί χρήστες της εφαρμογής, οι οποίοι στη συνέχεια μπορούν να πιστοποιηθούν και να εισέλθουν για διεξαγωγή διάφορων λειτουργιών.
- **Πίνακας User_rights:** Στον πίνακα αυτόν καταχωρούνται τα δικαιώματα των χρηστών για την διεξαγωγή των διαφόρων –ανά κατηγορία- ενεργειών.

Σχηματικά οι πίνακες και οι σχέσεις τους φαίνονται στην Εικόνα

4.3



Εικόνα 4.3. Σχηματική Διάταξη Πινάκων Βάσης Δεδομένων

4.3.2. Ανάπτυξη Εφαρμογής

Για την ανάπτυξη της εφαρμογής έγινε χρήση του Eclipse IDE [5]. Ακολουθώντας όλες τις προδιαγραφές που ορίστηκαν κατά τον σχεδιασμό της αρχιτεκτονικής αναπτύχθηκαν οι παρακάτω εννοιολογικές ενότητες:

- **Σύστημα διαχείρισης και πιστοποίησης χρηστών.** Το σύστημα αυτό καταχωρεί τους χρήστες που εγγράφονται από στο σύστημα από τη σχετική οθόνη της διεπαφής. Στη συνέχεια κάποιος διαχειριστής της εφαρμογής αναλαμβάνει να δώσει δικαιώματα δράσης του συγκεκριμένου χρήστη στις διαφορετικές ενότητες της εφαρμογής. Το σύστημα επίσης αναλαμβάνει και την πιστοποίηση του χρήστη κατά την αρχική εισαγωγή του στην εφαρμογή μέσω του σχετικού ονόματος και συνθηματικού χρήστη (username – password).
- **Σύστημα σύνδεσης και ανάκτησης δεδομένων από τη βάση δεδομένων.** Το σύστημα αυτό βασίζεται στο πακέτο Hibernate [2] το οποίο κάνοντας χρήση τον ορισμό της μοντελοποίησης της βάσης σε μορφή αντικειμένων Java γεμίζει τα αντικείμενα αυτά με δεδομένα από τη βάση προς χρήση από την εφαρμογή. Αντίστροφα, όταν πρέπει να καταχωρηθεί ένα σύνολο δεδομένων στη βάση, το Hibernate παίρνει τα δεδομένα σε μορφή αντικειμένων της αντικειμενοστραφούς γλώσσας προγραμματισμού Java και μετατρέποντάς τα στη μορφή που προδιαγράφεται από τη βάση τα καταχωρεί σε αυτή.
- **Τμήμα λειτουργικών εφαρμογών.** Στο τμήμα αυτό υλοποιήθηκαν όλες οι λειτουργικές απαιτήσεις από την εφαρμογή, όπως η συσχέτιση διαφορετικών αντικειμένων, η δυνατότητα τροποποίησης, η δυνατότητα αποθήκευσης

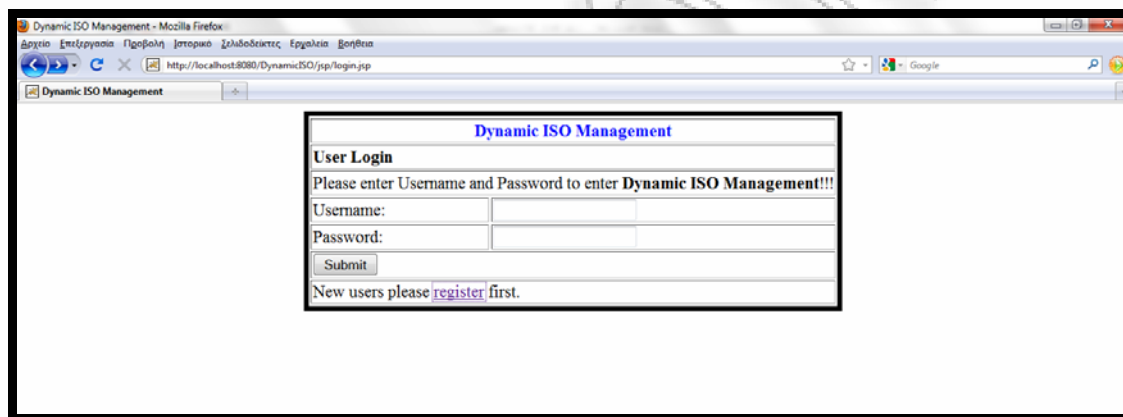
διαφορετικών εκδόσεων κάποιου αντικειμένου/διαδικασίας σε διαφορετικούς χρόνους και από διαφορετικούς χρήστες, κ.α.

- **Γραφική διεπαφή εφαρμογής.** Η γραφική διεπαφή είναι το σύστημα στο οποίο εργάζεται ο χρήστης της εφαρμογής. Αποτελεί το υποσύστημα που αναλαμβάνει την παρουσίαση τόσο των δεδομένων, όσο και των αντίστοιχων λειτουργιών που διαθέτει η εφαρμογή. Το εν λόγω υποσύστημα βασίζεται στην τεχνολογία της Java Enterprise Edition – JSP η οποία υποστηρίζει συνδυαστική λειτουργία της γλώσσας προγραμματισμού Java με στοιχεία παρουσίασης (HTML).

5. Πειραματικό Μέρος

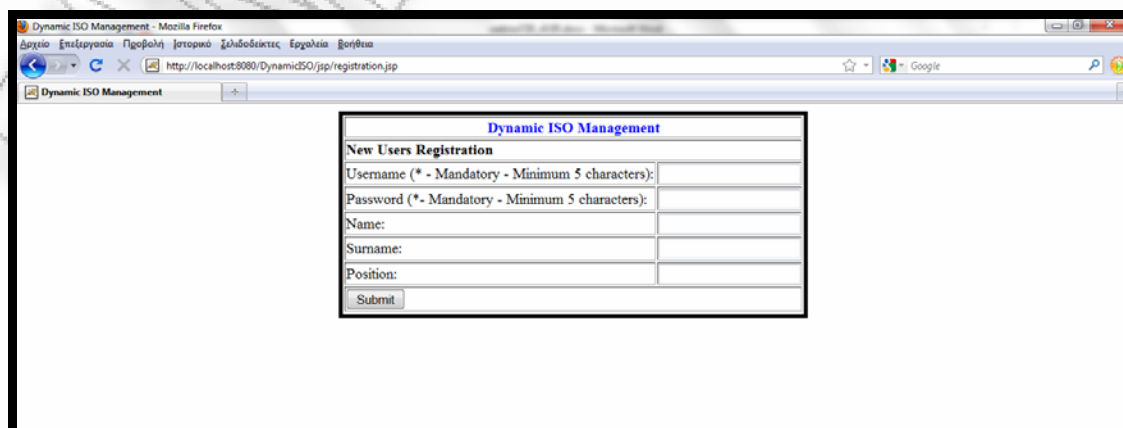
5.1. Χρήση της Πρότυπης Εφαρμογής

Για τη χρήση της εφαρμογής απαιτείται εξ αρχής η πιστοποίηση του χρήστη ο οποίος επιχειρεί να εισέλθει στην εφαρμογή. Η πιστοποίηση γίνεται βάσει των αποθηκευμένων στη βάση στοιχείων αναφορικά με τους χρήστες (πίνακες User και User_rights).



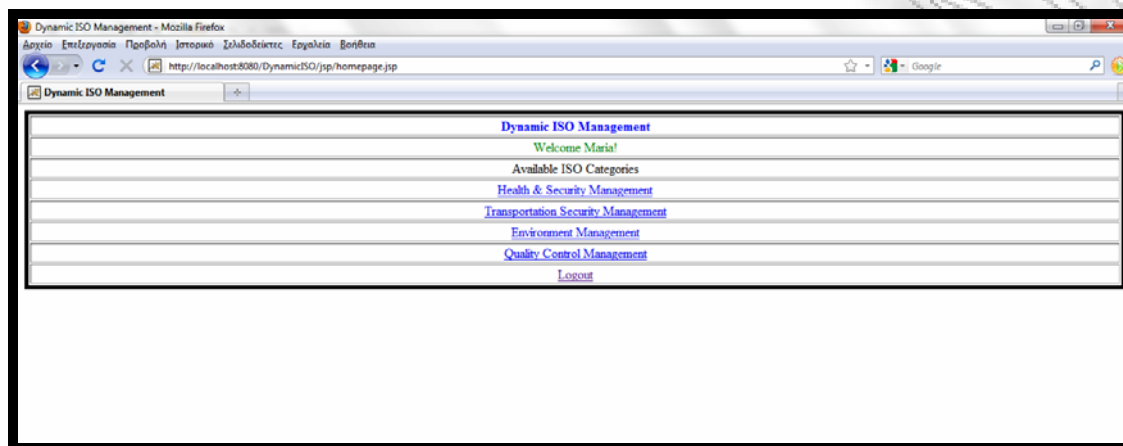
Εικόνα 5.1. Οθόνη Πιστοποίησης Χρηστών

Στην Εικόνα 5.1 φαίνεται η οθόνη πιστοποίησης χρηστών, ενώ στην Εικόνα 5.2 φαίνεται η αντίστοιχη εικόνα εγγραφής χρηστών.



Εικόνα 5.2. Οθόνη Εγγραφής Χρηστών

Στη συνέχεια, και ανάλογα με το επίπεδο των δικαιωμάτων του χρήστη που πιστοποιήθηκε για χρήση της εφαρμογής, ο χρήστης περιηγείται μέσω της κεντρικής σελίδας της εφαρμογής (Εικόνα 5.3).



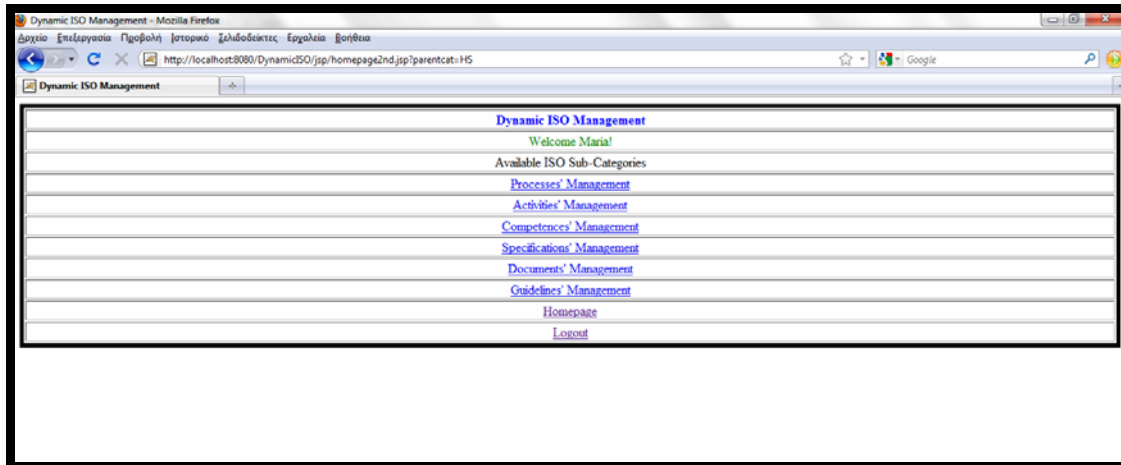
Εικόνα 5.3. Κεντρική Σελίδα Εφαρμογής

5.2. Λειτουργικό Σενάριο

Το λειτουργικό σενάριο που προδιαγράφεται στα πλαίσια της δοκιμής της πρότυπης εφαρμογής που αναπτύχθηκε αποτελείται από δύο βασικά μέρη. Την είσοδο και τη διεκπεραίωση δραστηριοτήτων από έναν χρήστη επίπεδο ασφάλειας «διαχειριστή εφαρμογής», και μια είσοδο και τροποποίηση μιας διαδικασίας ISO από έναν απλό χρήστη της εφαρμογής.

5.2.1. Λειτουργία Διαχειριστή Εφαρμογής

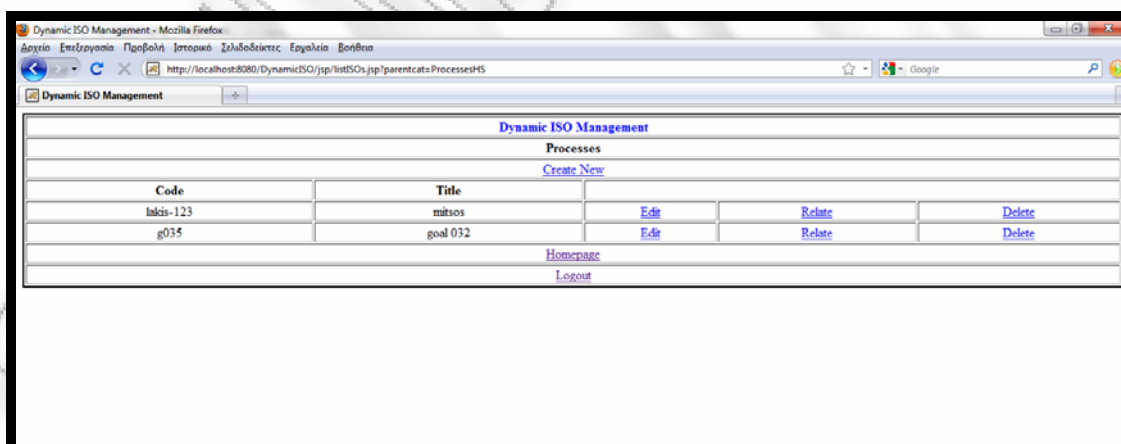
Ως διαχειριστής ορίζεται ο χρήστης που έχει πλήρη δικαιώματα στην εφαρμογή. Έτσι είναι δυνατόν να περιηγηθεί σε όλες τις ενότητες της εφαρμογής (Processes, Activities, Competences, Specifications, Documents και Guidelines) – (Εικόνα 5.4).



Εικόνα 5.4. Κεντρική Σελίδα Ενεργειών

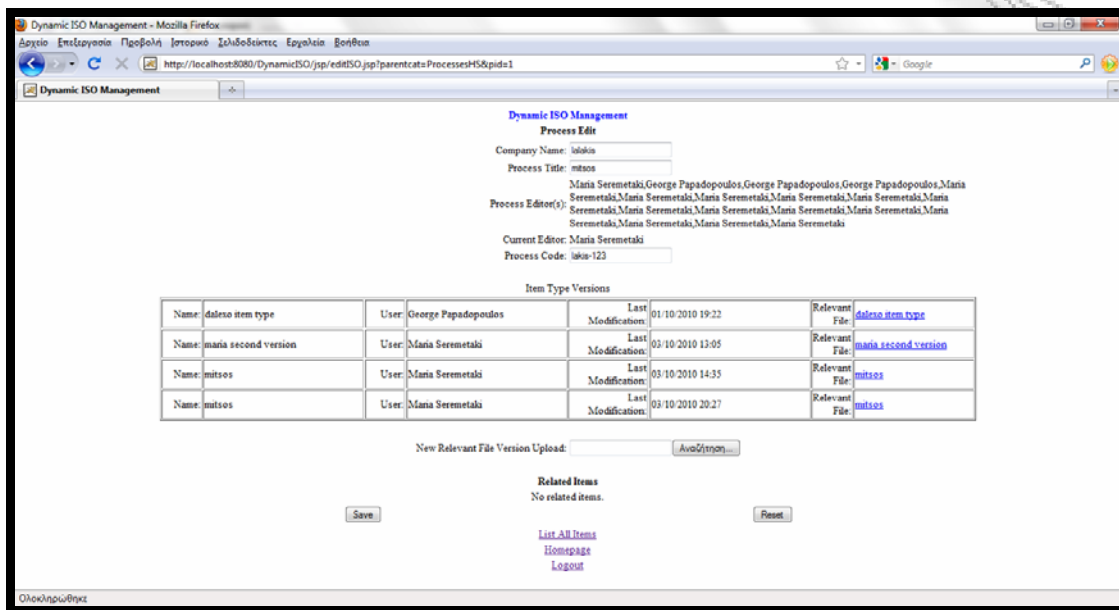
Σε κάθε ενότητα παρουσιάζονται τα αντίστοιχα αντικείμενα, καθώς και η δυνατότητα για:

- Δημιουργία νέου αντικειμένου
- Τροποποίηση κάποιου από τα υπάρχοντα (δημιουργία νέας έκδοσης)
- Συσχέτιση ενός αντικειμένου με κάποιο άλλο
- Διαγραφή κάποιου από τα υπάρχοντα.

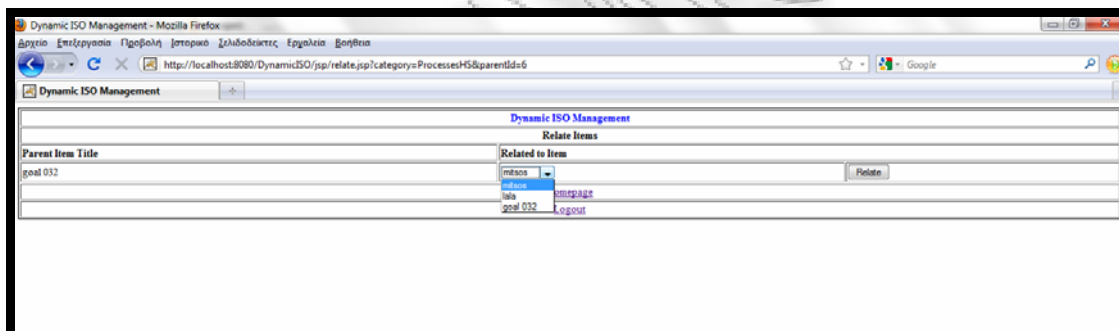


Εικόνα 5.5. Κεντρική Σελίδα Διαδικασιών (Processes)

Στο σενάριο ο χρήστης τροποποιεί μια διαδικασία (Process) (Εικόνα 5.5 και Εικόνα 5.6) καθώς και συσχετίζει μια διαδικασία με μία άλλη (Εικόνα 5.7).



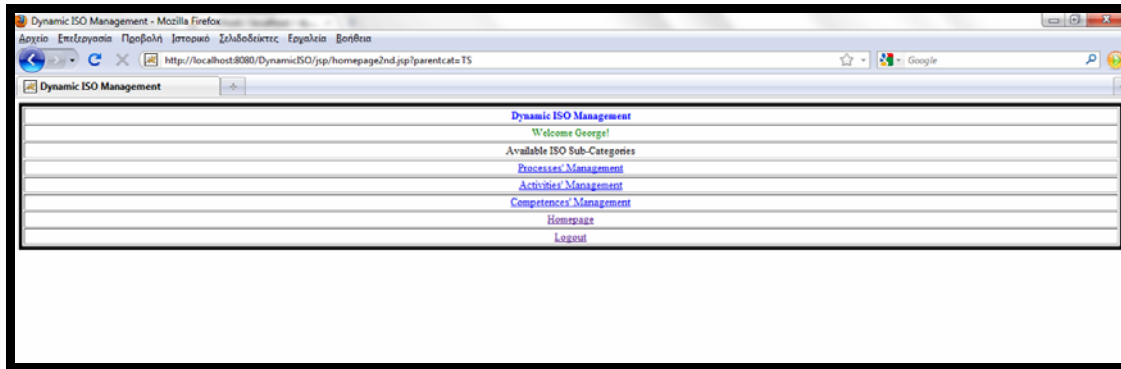
Εικόνα 5.6. Τροποποίηση Διαδικασιών (Processes)



Εικόνα 5.7. Συσχέτιση Διαδικασιών (Processes)

5.2.2. Λειτουργία Απλού Χρήστη Εφαρμογής

Ο απλός χρήστης ορίζεται ως ο χρήστης που έχει περιορισμένη πρόσβαση σε μερικές μόνο από τις ενότητες της εφαρμογής διαδικασιών / εγγράφων (Εικόνα 5.8). Έτσι μπορεί να «δει», να τροποποιήσει και να συσχετίσει μόνο αντικείμενα από τις ενότητες που έχει ο εκάστοτε χρήστης πρόσβαση.



Εικόνα 5.8. Κεντρική Σελίδα Διαδικασιών για Χρήστη με Περιορισμένη Πρόσβαση

6. Σύνοψη – Συμπεράσματα

6.1. Σύνοψη

Στα πλαίσια της παρούσας εργασίας έγινε μελέτη και ανάλυση των διαθέσιμων αρχιτεκτονικών στις υπηρεσίες διαδικτύου με σκοπό τον ορισμό ενός πλαισίου υποστήριξης διαδικασιών προτυποποίησης κατά ISO.

Έτσι, αφού περιγράφηκε η διαδικασία ανάπτυξης και τροποποίησης προτύπων ISO για να κατανοηθούν οι ανάγκες που προκύπτουν κατά τις εφαρμογή τους, αναλύθηκε και προδιαγράφηκε μια αρχιτεκτονική για την προαναφερθείσα υποστήριξη. Το πλαίσιο (framework) αυτό παρέχει τη δυνατότητα για αποθήκευση, ανάκτηση, τροποποίηση και συσχέτιση αντικειμένων που συμμετέχουν στα πρότυπα ISO, μέσω της προσβασιμότητας μέσω διαδικτύου. Παράλληλα διαχωρίζει τους χρήστες σε διαφορετικά επίπεδα δικαιωμάτων, και ανάλογης πρόσβασης σε ενότητες της εφαρμογής.

Αφού προδιαγράφηκε η αρχιτεκτονική προχωρήσαμε σε μια πρότυπη υλοποίηση κάνοντας χρήση τεχνολογιών ανοικτού κώδικα και τη γλώσσα προγραμματισμού Java. Έτσι προέκυψε μια διαδικτυακή εφαρμογή η οποία εγκαταστάθηκε και δοκιμάστηκε σε διαφορετικά λειτουργικά συστήματα, και έγινε πρόσβαση σε αυτήν από διαφορετικές τοποθεσίες (αποκεντρωμένη πρόσβαση).

Για την δοκιμή της εφαρμογής εκτελέστηκαν δύο διαφορετικά σενάρια. Το πρώτο αποτελούσε πρόσβαση για τροποποίηση διαδικασιών από ένα χρήστη με πλήρη δικαιώματα προσβασιμότητας στην εφαρμογή, ενώ το δεύτερο ανάλογες λειτουργίες από ένα χρήστη με περιορισμένα δικαιώματα. Τα αποτελέσματα του πειραματικού αυτού μέρους έδειξαν ότι η πρότυπη υλοποίηση είναι καθόλα λειτουργική, και δύναται να διατελέσει υποστηρικτικό ρόλο σε μια εταιρεία για την

αποτελεσματικότερη παρακολούθηση και ανάπτυξη των απαιτούμενων κατά ISO διαδικασιών.

6.2. Μελλοντικές Επεκτάσεις - Συμπεράσματα

Αυτονόητο είναι για κάθε ερευνητική προσπάθεια ότι προκύπτουν νέες ανάγκες για περαιτέρω έρευνα και ανάπτυξη στο ίδιο ή σε παραπλήσια ερευνητικά πεδία. Έτσι με την ανάλυση και υλοποίηση όλων όσων πραγματεύτηκε η παρούσα εργασία, προκύπτει η ανάγκη για περαιτέρω ανάπτυξη λειτουργικότητας για την εφαρμογή, κάνοντάς την φιλικότερη προς τον τελικό χρήστη, καθώς σε τελική ανάλυση η επιτυχία κάθε συστήματος επαφίεται στην δυνατότητά της να «κερδίσει» τον τελικό χρήστη παραμένοντας παράλληλα λειτουργική και αποτελεσματική. Έτσι μια βελτιωμένη σχεδίαση της διεπαφής με τον χρήστη και η προσθήκη αυτοματισμών και νέων λειτουργιών είναι κάποιες κατευθυντήριες προς τις οποίες θα μπορούσε ενδεχομένως να είναι ο στόχος των επόμενων προσπαθειών στον συγκεκριμένο τομέα.

Συμπερασματικά βλέπουμε πως στον τομέα της προτυποποίησης κατά ISO υπάρχει χώρος για υποστήριξη από ένα αποκεντρωμένο διαδικτυακό σύστημα, το οποίο φάνηκε άμεσα από την σχεδίαση και πρότυπη υλοποίηση που ακολουθήθηκε στα πλαίσια αυτής της εργασίας. Παρόλα αυτά, καμία εφαρμογή δεν μπορεί να αποτελέσει πανάκεια κάνοντας την ανάγκη για διαρκή βελτίωση της παρούσας προτεινόμενης ιδέας επιτακτική.

ΠΑΡΑΡΤΗΜΑ Α Κώδικας Εφαρμογής

Πίνακας 1. Κώδικας Διαχείρισης Σύνδεσης στη Βάση Δεδομένων

```
package com.sh.dyniso.util;

import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class HibernateUtil {

    private static final SessionFactory sessionFactory = buildSessionFactory();

    private static SessionFactory buildSessionFactory() {
        try {
            // Create the SessionFactory from hibernate.cfg.xml
            return new Configuration().configure().buildSessionFactory();
        }
        catch (Throwable ex) {
            // Make sure you log the exception, as it might be swallowed
            System.err.println("Initial SessionFactory creation failed." + ex);
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }
}
```

Πίνακας 2. Βοηθητική Κλάση Παρουσίασης

```
package com.sh.dyniso.util;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.Iterator;
import java.util.List;

import com.sh.dyniso.model.Item;
import com.sh.dyniso.model.ItemHome;
import com.sh.dyniso.model.ItemRelation;
import com.sh.dyniso.model.ItemRelationHome;
import com.sh.dyniso.model.User;
import com.sh.dyniso.model.UserHome;

public class JspUtil {

    private static ItemHome itemHome = new ItemHome();
    private static UserHome userHome = new UserHome();

    public static String getEditorNames(Integer itemId) {
        StringBuffer editors = new StringBuffer();
        try {
            Item item = itemHome.findById(itemId);
            String userIds = item.getUserIds();
            String[] uids = userIds.split(",");
            for (int i=0; i<uids.length; i++) {
                try {
                    User user = userHome.findById(new
Integer(uids[i]));
                    if (i>0 && user !=null) editors.append(",");
                    if (null!=user) editors.append(user.getName() +
" " + user.getSurname());
                } catch (Exception e) {
                    // do nothing - skip possible erroneous userid
                    entry
                }
            }
        } catch (Exception e) {
            //do nothing, return empty string
        }
        return editors.toString();
    }

    public static String formatDate(Date date, String format) {
        String formatted = "";
    }
}
```

```

SimpleDateFormat sdf = new SimpleDateFormat(format);
if (date!=null) {
    formatted = sdf.format(date);
} else {
    formatted = sdf.format(new Date());
}

return formatted;
}

public static Date parseDate(String source, String format) throws Exception {
    SimpleDateFormat sdf = new SimpleDateFormat(format);
    if (source!=null) {
        return sdf.parse(source);
    } else {
        return new Date();
    }
}

public static ArrayList<Item> getRelatedItems(Item item) throws Exception {
    ArrayList<Item> list = new ArrayList<Item>();

    ItemHome iHome = new ItemHome();
    ItemRelationHome irHome = new ItemRelationHome();
    List<ItemRelation> mlist = irHome.findByHQL("from ItemRelation
where parentId = " + item.getItemId());

    if (mlist!=null) {
        Iterator<ItemRelation> mit = mlist.iterator();
        ItemRelation tmpIR = null;
        while (mit.hasNext()) {
            tmpIR = mit.next();
            list.add(iHome.findById(tmpIR.getItemId()));
        }
    }

    return list;
}
}
}

```

Πίνακας 3. Διαχείριση Χρηστών και Δικαιωμάτων

```
package com.sh.dyniso.control;
```

```

import java.util.HashSet;
import java.util.Iterator;
import java.util.List;
import java.util.Set;

import com.sh.dyniso.model.Categorydef;
import com.sh.dyniso.model.Categorytree;
import com.sh.dyniso.model.CategorytreeHome;
import com.sh.dyniso.model.User;
import com.sh.dyniso.model.UserHome;
import com.sh.dyniso.model.UserRights;

public class UserControl {

    private static UserHome uh = new UserHome();
    private static CategorytreeHome cth = new CategorytreeHome();

    public static boolean canAccessSection(int categoryId, String userName) {

        boolean canAccess = false;

        User exampleUser = new User();
        exampleUser.setUsername(userName);
        List<User> matchingUsers = uh.findByExample(exampleUser);
        exampleUser = (matchingUsers.size()>0) ? matchingUsers.get(0) : null;

        if (null != exampleUser ) {
            Set<UserRights> rights = exampleUser.getUserRightses();
            Iterator<UserRights> rightsIt = rights.iterator();
            UserRights testur;
            while (rightsIt.hasNext()) {

```

```

        testur = rightsIt.next();
        if (testur.getCategorytree().getCatId() == categoryId)
            canAccess = true;
    }
}

return canAccess;

}

public static boolean canAccessSection(String categoryName, String
userName) {

    boolean canAccess = false;

    User exampleUser = new User();
    exampleUser.setUsername(userName);
    List<User> matchingUsers = uh.findByExample(exampleUser);
    exampleUser = (matchingUsers.size()>0) ? matchingUsers.get(0) : null;

    CategorytreeHome ch = new CategorytreeHome();

    if (null != exampleUser ) {
        Set<UserRights> rights = exampleUser.getUserRightses();
        Iterator<UserRights> rightsIt = (rights!=null) ? rights.iterator() :
new HashSet().iterator();
        UserRights testur;
        Categorytree ct;
        Set<Categorydef> scd;
        while (rightsIt.hasNext()) {
            testur = rightsIt.next();
            ct = ch.findById(testur.getCategorytree().getCatId());

```

```

        scd = (ct!=null) ? ct.getCategorydefs() : new
HashSet<Categorydef>();
        Iterator<Categorydef> scdit = scd.iterator();
        while (scdit.hasNext()) {
            Categorydef cd = scdit.next();
            if ( cd!=null &&
cd.getCatLabel().trim().equalsIgnoreCase(categoryName.trim()) ) {
                canAccess = true;
                break;
            }
        }
    }
}

return canAccess;
}
}

```

Πίνακας 4. Σελίδα Πιστοποίησης Χρήστη

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<%@ page import="java.util.*" %>
<%@ page import="com.sh.dyniso.model.*" %>
<jsp:useBean id="userHome" class="com.sh.dyniso.model.UserHome"
scope="session"/>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Dynamic ISO Management</title>
</head>
<body>

<%
session.setAttribute("user", null);
%>

<form action="homepage.jsp" method="post">
    <table border="5" bordercolor="black" align="center"><tr><td
colspan="2" colour="blue" align="center"><b><span
style="color:blue">Dynamic ISO Management</span></b></td></tr>

        <tr>
            <td colspan="2"><b>User Login</b></td>
        </tr>
        <tr>
            <td colspan="2">Please enter Username and Password
to enter <b>Dynamic ISO Management</b>!!!</td>
        </tr>
        <tr>
            <td><u>Username</u>: </td> <td><input type="text"
name="username"/></td>
        </tr>
        <tr>
            <td>Password: </td> <td><input type="password"
name="passwd"/></td>
        </tr>
        <tr>
            <td colspan="2"><input type="submit"
value="Submit"/></td>
        </tr>
        <tr>
            <td colspan="2">New users please <a
href="registration.jsp">register</a> first.</td>
        </tr>
    </table>
</form>

</body>
</html>
```


Πίνακας 5. Σελίδα Παρουσίασης Διαδικασιών

```
<%@ page language="java" contentType="text/html; charset=iso-8859-7"
    pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<%@ page import="java.util.*" %>
<%@ page import="com.sh.dyniso.model.*" %>

<jsp:useBean id="userHome" class="com.sh.dyniso.model.UserHome" scope="session"/>
<jsp:useBean id="itemHome" class="com.sh.dyniso.model.ItemHome" scope="session"/>
<jsp:useBean id="itemTypeHome" class="com.sh.dyniso.model.ItemTypeHome" scope="session"/>
<jsp:useBean id="catDefHome" class="com.sh.dyniso.model.CategorydefHome" scope="session"/>

<%@page import="com.sh.dyniso.model.User"%>
<%@page import="com.sh.dyniso.control.UserControl"%>
<%@page import="com.sh.dyniso.util.HibernateUtil"%>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Dynamic ISO Management</title>
</head>
<body>

<%
String msg = "";

User sessionUser = (User)session.getAttribute("user");
boolean errorOccured = false;

try {
    HibernateUtil.getSessionFactory().getCurrentSession().beginTransaction();

    String parentcat = request.getParameter("parentcat");
    if (sessionUser != null && UserControl.canAccessSection(parentcat,
sessionUser.getUsername())) {
        List<Categorydef> catDefList =catDefHome.findByHQL("from Categorydef where
catLabel='"+parentcat+"'");
        Categorydef cd = (catDefList!=null && catDefList.size()>0) ? catDefList.get(0)
String catLabel = (null!=cd) ? cd.getCatDescr() : "Processes (default)";
String catId = (null!=cd) ? cd.getCategorytree().getCatId().toString() : "-1";

        List<Item> processes = itemHome.findByHQL("from Item where categoryId="+catId)

        Iterator<Item> proci = processes.iterator();
    }

    <table border="2" bordercolor="black" align="center" width="100%">
    <tr><td colspan="5" colour="blue" align="center"><b><span
style="color:blue">Dynamic ISO Management</span></b></td></tr>

    <tr><td colspan="5" colour="blue" align="center"><b><span
style="color:black"><%=catLabel%></span></b></td></tr>

    <tr><td colspan="5" colour="blue" align="center"><a
href="editISO.jsp?parentcat=<%=parentcat%>&pid=new">Create New</a></td></tr>

    <tr>
        <td colour="blue" align="center">
            <b>Code</b>
        </td>
    </tr>
    </table>
    </body>
    </html>
    </pre>
```

```

        <td colour="blue" align="center">
            <b>Title</b>
        </td>
        <td colspan="3" colour="blue" align="center">
            &nbsp;
        </td>
    </tr>
    <%
    Item tmpit = null;
    while (procit.hasNext()) {
        tmpit = procit.next();
        %>
    <tr>
        <td colour="blue" align="center">
            <%=tmpit.getItemCode()%>
        </td>
        <td colour="blue" align="center">
            <%=tmpit.getItemTitle()%>
        </td>
        <td colour="blue" align="center">
            <a
href="editISO.jsp?parentcat=<%=parentcat%>&pid=<%=tmpit.getItemId()%>">Edit</a>
        </td>
        <td colour="blue" align="center">
            <a
href="relate.jsp?category=<%=parentcat%>&parentId=<%=tmpit.getItemId()%>">Relate</a>
        </td>
        <td colour="blue" align="center">
            <a
href="editISO.jsp?parentcat=<%=parentcat%>&delete=true&pid=<%=tmpit.getItemId()%>">D
        </td>
    </tr>
    <%
    }

    %>
    <tr><td colspan="5" colour="black" align="center"><a
href="homepage.jsp">Homepage</a><br/></td></tr>
    <tr><td colspan="5" colour="black" align="center"><a
href="login.jsp">Logout</a><br/></td></tr>
    </table>
    <%
    } else {
        %>User not logged in or not authorized to access this section!!!<%
    }

    HibernateUtil.getSessionFactory().getCurrentSession().getTransaction().commit();
} catch (Exception exe) {
    HibernateUtil.getSessionFactory().getCurrentSession().getTransaction().rollback();
    exe.printStackTrace();
}
%>
</body>
</html>

```

Πίνακας 6. Σελίδα Διαχείρισης Διαδικασίας

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR
<%@ page import="java.util.*" %>
<%@ page import="java.io.*" %>
<%@ page import="com.sh.dyniso.model.*" %>
<%@ page import="org.apache.commons.fileupload.*" %>
<%@ page import="org.apache.commons.fileupload.servlet.*" %>
<%@ page import="org.apache.commons.fileupload.disk.*" %>

<jsp:useBean id="userHome" class="com.sh.dyniso.model.UserHome" scope="session"/>
<jsp:useBean id="itemHome" class="com.sh.dyniso.model.ItemHome" scope="session"/>
<jsp:useBean id="itemTypeHome" class="com.sh.dyniso.model.ItemTypeHome" scope="sessi
<jsp:useBean id="itemRelHome" class="com.sh.dyniso.model.ItemRelationHome" scope="se
<jsp:useBean id="catDefHome" class="com.sh.dyniso.model.CategorydefHome" scope="sess

<%@page import="com.sh.dyniso.model.User"%>
<%@page import="com.sh.dyniso.control.UserControl"%>
<%@page import="com.sh.dyniso.util.HibernateUtil"%>
<%@page import="com.sh.dyniso.util.JspUtil"%>

<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Dynamic ISO Management</title>
</head>
<body>

<%
String msg = "";

User sessionUser = (User)session.getAttribute("user");
boolean errorOccured = false;
String parentcat = request.getParameter("parentcat");

try {
    HibernateUtil.getSessionFactory().getCurrentSession().beginTransaction()

if (sessionUser != null && UserControl.canAccessSection(parentcat, sessionUser.getUs
    %>
    <table bordercolor="black" align="center" width="100%">
        <tr><td colour="blue" align="center"><b><span style="color:blue">Dynamic
        <tr><td colour="blue" align="center"><b><span style="color:black">Proces

    <%

List<Categorydef> catDefList =catDefHome.findByHQL("from Categorydef where cat
Categorydef cd = (catDefList!=null && catDefList.size()>0) ? catDefList.get(0)
String catLabel = (null!=cd) ? cd.getCatDescr() : "Processes (default)";
String catId = (null!=cd) ? cd.getCategorytree().getCatId().toString() : "-1";

String pidStr = null;
String itemTitle = null;
String itemCode = null;
String currEditorId = null;
String companyName = null;

String itemTypeLastMod = null;
String itemTypeScope = null;
String itemTypeField = null;
```

```

String itemTypeGeneralDescr = null;
String itemTypeSpecificDescr = null;
String pidOldMultipart = null;
String deleteMultipart = null;

byte[] itemTypeFile = null;
String itemTypeFileName = null;

boolean isMultipart = ServletFileUpload.isMultipartContent(request);
if (isMultipart) {
    // Create a factory for disk-based file items
    FileItemFactory factory = new DiskFileItemFactory();
    // Create a new file upload handler
    ServletFileUpload upload = new ServletFileUpload(factory);
    // Parse the request
    List items = upload.parseRequest(request);

    // Process the uploaded items
    Iterator iter = items.iterator();
    while (iter.hasNext()) {
        FileItem item = (FileItem) iter.next();
        if (item.isFormField()) {
            String name = item.getFieldName();
            String value = item.getString();
            if (name.trim().equalsIgnoreCase("pid")) {
                pidStr = (value==null) ? "-1" : value;
            } else if (name.trim().equalsIgnoreCase("itemTitle")) {
                itemTypeTitle = value;
            } else if (name.trim().equalsIgnoreCase("itemCode")) {
                itemTypeCode = value;
            } else if (name.trim().equalsIgnoreCase("editorId")) {
                currEditorId = value;
            } else if (name.trim().equalsIgnoreCase("companyName")) {
                companyName = value;
            } else if (name.trim().equalsIgnoreCase("itemTypeLastMod")) {
                itemTypeLastMod = value;
            } else if (name.trim().equalsIgnoreCase("itemTypeScope")) {
                itemTypeScope = value;
            } else if (name.trim().equalsIgnoreCase("itemTypeField")) {
                itemTypeField = value;
            } else if (name.trim().equalsIgnoreCase("itemTypeGeneralDescr")) {
                itemTypeGeneralDescr = value;
            } else if (name.trim().equalsIgnoreCase("itemTypeSpecificDescr")) {
                itemTypeSpecificDescr = value;
            } else if (name.trim().equalsIgnoreCase("pidOld")) {
                pidOldMultipart = value;
            } else if (name.trim().equalsIgnoreCase("delete")) {
                deleteMultipart = value;
            }
        } else { //item is a file
            String fieldName = item.getFieldName();
            itemTypeFileName = item.getName();
            itemTypeFile = item.get();
        }
    }
} else {
    pidStr = request.getParameter("pid");
    pidStr = (pidStr==null) ? "-1" : pidStr;

    itemTypeTitle = request.getParameter("itemTitle");
    itemTypeCode = request.getParameter("itemCode");
    currEditorId = request.getParameter("editorId");
}

```

```

        companyName = request.getParameter("companyName");
        itemTypeLastMod = request.getParameter("itemTypeLastMod");
        itemTypeScope = request.getParameter("itemTypeScope");
        itemTypeField = request.getParameter("itemTypeField");
        itemTypeGeneralDescr = request.getParameter("itemTypeGeneralDescr");
        itemTypeSpecificDescr = request.getParameter("itemTypeSpecificDescr");
    }

    if (pidStr.trim().equalsIgnoreCase("newsubmission")) {
        // new item submitted, catch all data and insert into db

        //create item: note itemtypeid = -1 as long as itemtype has not yet been
        Item newItem = new Item(companyName, -1, itemTitle, currEditorId, Integer.parseInt(itemCode));
        itemHome.persist(newItem);

        String meta = (itemTypeFile!=null && itemTypeFileName!=null) ?
"filetype="+itemTypeFileName.substring(itemTypeFileName.lastIndexOf("."))+"," : "";
        //create itemtype: note that we get the just assigned itemid AFTER execution
        ItemType newItemType = new ItemType(newItem.getItemId().intValue(), newItemTypeScope, itemTypeField, itemTypeGeneralDescr, itemTypeSpecificDescr,
            null, new Date(),
            null, -1,
            -1, null,
            null, sessionUser.getName()+" "+sessionUser.getSurname(), null,
            null, null, meta,
            new Date());

        itemTypeHome.persist(newItemType);

        //Finally update item with newly created itemtype id
        newItem.setItemTypeId(newItemType.getItemId().intValue());
        itemHome.merge(newItem);

        %>
        <tr><td colour="green" align="center"><b><span style="color:green">Process
successfully!!!</span></b></td></tr>
        <%

    } else if (pidStr.trim().equalsIgnoreCase("oldsubmission")) {
        // new item submitted, catch all data and insert into db
        //create item: note itemtypeid = -1 as long as itemtype has not yet been
        String pidOldStr = (!isMultipart)? request.getParameter("pidOld") : pidOldStr;
        Integer pidOld = new Integer(pidOldStr);

        Item oldItem = itemHome.findById(pidOld);
        oldItem.setItemTitle(itemTitle);
        oldItem.setItemCode(itemCode);
        oldItem.setCompanyName(companyName);
        if (oldItem.getUserIds()==null || oldItem.getUserIds().length()==0) {
            oldItem.setUserIds(currEditorId);
        } else {
            oldItem.setUserIds(oldItem.getUserIds()+","+currEditorId);
        }
        itemHome.merge(oldItem);

        if (itemTypeFile!=null) {
            String meta = (itemTypeFile!=null && itemTypeFileName!=null) ?
"filetype="+itemTypeFileName.substring(itemTypeFileName.lastIndexOf("."))+"," : "";
            //create itemtype: note that we get the just assigned itemid AFTER execution
            ItemType newItemType = new ItemType(oldItem.getItemId().intValue(), newItemTypeScope, itemTypeField, itemTypeGeneralDescr, itemTypeSpecificDescr,
                null, new Date(),
                null, -1,
                null, -1,
                null, null, meta,
                new Date());
        }
    }
}

```

```

        -1, null,
        null, sessionUser.getName()+" "+sessionUser.getSurname(),
        null, null, meta,
        new Date());
        itemTypeHome.persist(newItemType);
    }

    /* Old application: update one itemType, NOW multiple itemTypes for each
    ItemType oldItemType = itemTypeHome.findById(new Integer(oldItem.getItemType()));
    oldItemType.setLastModification(new Date());
    oldItemType.setScope(itemTypeScope);
    oldItemType.setField(itemTypeField);
    oldItemType.setGeneralDescr(itemTypeGeneralDescr);
    oldItemType.setSpecificDescr(itemTypeSpecificDescr);
    if (itemTypeFile!=null) {
        oldItemType.setHtmlForm(itemTypeFile);
        if (null!=oldItemType.getMetadata() && oldItemType.getMetadata().i

    oldItemType.setMetadata(oldItemType.getMetadata()+", filetype="+itemTypeFileName
    }
    itemTypeHome.merge(oldItemType);
    */

    %>
    <tr><td colour="green" align="center"><b><span style="color:green">Process
successfully!!!</span></b></td></tr>
    <%

    } else {

        Item process = null;
        List<ItemType> itemTypeList = new ArrayList<ItemType>();
        //ItemType processType = null;
        if (pidStr.trim().equalsIgnoreCase("new")) {
            //create new item/itemtype objects to avoid nullpointers
            process = new Item();
            //processType = new ItemType();
        } else { //numeric value, existing process to edit
            Integer pid = new Integer(pidStr);
            process = itemTypeHome.findById(pid);
            itemTypeList = itemTypeHome.findByHQL("from ItemType where itemId="
            //processType = itemTypeHome.findById(new Integer(process.getItemType

        }

        String delete = (isMultipart) ? deleteMultipart : request.getParameter("
        if (delete!=null && delete.length()>0) {

            List<ItemRelation> delRelList = itemRelHome.findByHQL("from ItemRe
or itemId = " + process.getItemId());
            if (delRelList!=null && delRelList.size()>0) {
                Iterator<ItemRelation> delreelit = delRelList.iterator();
                ItemRelation deltmp = null;
                while (delreelit.hasNext()) {
                    deltmp = delreelit.next();
                    itemRelHome.delete(deltmp);
                }
            }
            itemTypeHome.delete(process);
            if (itemTypeList.size()>0) {
                Iterator<ItemType> typeIt = itemTypeList.iterator();
                ItemType tmp = null;
                while (typeIt.hasNext()) {
                    tmp = typeIt.next();

```

```

        itemTypeHome.delete(tmp);
    }
}
//itemTypeHome.delete(processType);
%>
<tr><td colour="red" align="center"><b><span style="color:red">Pro
successfully!!!</span></b></td></tr>
<%
} else {

%>

<tr>
<td>
<form name="editform" id="editform" action="editISO.jsp?pare
enctype="multipart/form-data">
<table bordercolor="black" align="center" width="75%">
<!-- Start of Part ONE, Item attributes -->
<tr>
<td width="50%" align="right">Company Name
value="<%=process.getCompanyName()%>" /></td>

</tr>
<tr>
<td width="50%" align="right">Process Title
<td>
<%
if (pidStr.trim().equalsIgnoreCase("
%>
<input type="hidden" name="pid" valu
<%
} else {
%>
<input type="hidden" name="pid
<input type="hidden" name="pid
<%
}
%>
<input type="text" name="itemTitle"
</td>
</tr>
<tr>
<td width="50%" align="right">Process Edit
<td><%=JspUtil.getEditorNames(process.getItemId())%></td>
</tr>
<tr>
<td width="50%" align="right">Current Edit
value="<%=sessionUser.getUserId()%>" /><%=sessionUser.getName()%>&nbsp;<%=sessionUser
</tr>
<tr>
<td width="50%" align="right">Process Code
value="<%=process.getItemCode()%>" /></td>
</tr>
<!-- End of Part ONE, Item attributes -->

<!-- Start of Part TWO, ItemType attributes -->
<tr>

```



```

                <b>Code</b>
            </td>
            <td colour="blue" align="center">
                <b>Title</b>
            </td>
            <td colspan="2" colour="blue" align="center">
                &nbsp;
            </td>
        </tr>
    <%
        Item tmpit = null;
        while (procit.hasNext()) {
            tmpit = procit.next();
        >
        <tr>
            <td colour="blue" align="center">
                <%=tmpit.getItemCode() %>
            </td>
            <td colour="blue" align="center">
                <%=tmpit.getItemTitle() %>
            </td>
            <td colour="blue" align="center">
                <a
                    href="editISO.jsp?parentcat=<%=parentcat%>&pid=<%=tmpit.getItemId() %>">Edit</a>
            </td>
            <td colour="blue" align="center">
                <a
                    href="editISO.jsp?parentcat=<%=parentcat%>&delete=true&pid=<%=tmpit.getItemId() %>">Delete</a>
            </td>
        </tr>
    <%
        }
    >
    </table>
</td>
</tr>
<%
} else {
    >
    <tr>
        <td colspan="2" align="center">No related items found.
    </tr>
    >
}
>
<!-- End of Part TWO, ItemType attributes -->
<tr>
    <td width="50%" align="center"><input type="button" value="Add" />
    <td width="50%" align="center">
        <input type="reset" value="Reset" width="100px" />
    </td>
</tr>
</table>
</form>
</td>
</tr>
<%

```

```

        } // not delete function
    }

    %>
    <tr><td colour="black" align="center"><a href="listISOs.jsp?parentcat=<%
    <tr><td colour="black" align="center"><a href="homepage.jsp">Homepage</a>
    <tr><td colour="black" align="center"><a href="login.jsp">Logout</a><br/>
</table>
<%
} else {
    %>User not logged in or not authorized to access this section!!!<%
}

HibernateUtil.getSessionFactory().getCurrentSession().getTransaction().commit(
} catch (Exception exe) {
    HibernateUtil.getSessionFactory().getCurrentSession().getTransaction().rollback();
    exe.printStackTrace();
}

%>
</body>
</html>

```

ΠΑΡΑΡΤΗΜΑ Β

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΡΠΑ

Ακρωνύμια

- GUI – Graphical User Interface
- TC - Technical Committee
- SC – Sub Committee
- DIS - Draft International Standard
- FDIS - Final Draft International Standard
- LAN – Local Access Network
- RDBMS - Relational Database Management System
- HTML – HyperText Markup Language
- JSP – Java Server Pages

Βιβλιογραφία

- [1] Πακέτο Λογισμικού hibernate - <http://www.hibernate.org>
- [2] Εφαρμογή phpmyadmin - <http://www.phpmyadmin.net>
- [3] Γλώσσα προγραμματισμού Java - <http://www.oracle.com/technetwork/java/index.html>
- [4] Apache Web Server - <http://httpd.apache.org/>
- [5] Περιβάλλον Ανάπτυξης Eclipse IDE - <http://www.eclipse.org/>
- [6] Διεθνής Οργανισμός ISO - <http://www.iso.org/iso/home.html>
- [7] Olivier Crameri, Ricardo Bianchini, Nikola Knezevic, Willy Zwaenepoel, Dejan Kostic. Staged Deployment in Mirage, an Integrated Software Upgrade Testing and Distribution System. SOSP'07, October 14-17, 2007, Stevenson, Washington, USA.
- [8] Roberto Di Cosmo, Stefano Zacchiroli, Paulo Trezentos. Package upgrades in FOSS distributions: details and challenges. In ACM Conference on Object-Oriented Programming: Systems, Languages, and Applications (OOPSLA), 2008.
- [9] E. Dolstra and A. Löh. NixOS: A purely functional Linux distribution. In ICFP'08.
- [10] Archana Ganapathi, Yi-Min Wang, Ni Lao, Ji-Rong Wen. Why PCs Are Fragile and What We Can Do About It: A Study of Windows Registry Problems. Proc. IEEE International Conference on Dependable Systems and Networks (DSN), June 2004.
- [11] Christos Gkantsidis, Thomas Karagiannis, Pablo Rodriguez, and Milan Vojnovic. Planet Scale Software Updates. In Proceedings of SIGCOMM, September 2006.
- [12] Slinger Jansen, Gerco Ballintijn, and Sjaak Brinkkemper. A process model and typology for software product updaters. In 9th European Conference on Software Maintenance and Reengineering (CSMR), 2005.

- [13] Havard Johansen and Dag Johansen. Resilient Software Mirroring With Untrusted Third Parties. In ACM Conference on Object-Oriented Programming: Systems, Languages, and Applications (OOPSLA), 2008.
- [14] Arjen Lenstra and Benne de Weger. On the possibility of constructing meaningful hash collisions for public keys. ACISP 2005, Brisbane, Australia, July 4-6, 2005.
- [15] Robert A. Martin. Managing Vulnerabilities in Networked Systems. Computer, vol. 34, no. 11, pp. 32-38, Nov. 2001, doi:10.1109/2.963441
- [16] A. Adya, W.J. Bolosky, M. Castro, G. Cermak, R. Chaiken, J.R. Douceur, J. Howell, J.R. Lorch, M. Theimer, and R.P. Wattenhofer, "FARSITE: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment," Proc. Symp. Operating Systems Design and Implementation (OSDI '02), Dec. 2002.
- [17] A. Awan, R. Ferreira, A. Grama, and S. Jagannathan, "Distributed Uniform Sampling in Large Real-World Networks," technical report, Dept. of Computer Sciences, Purdue Univ., Oct. 2004.
- [18] M. Bawa, H. Garcia-Molina, A. Gionis, and R. Motwani, "Estimating Aggregates on a Peer-to-Peer Network," technical report, Computer Science Dept., Stanford Univ., 2003.
- [19] W.J. Bolosky, S. Corbin, D. Goebel, and J.R. Douceur, "Single Instance Storage in Windows 2000," Proc. Fourth USENIX Windows Systems Symp., Aug. 2000.
- [20] S. Boyd, P. Diaconis, and L. Xiao, "Fastest Mixing Markov Chain on a Graph," SIAM Rev., Problems and Techniques Section, vol. 46, no. 4, pp. 667-689, Dec. 2004.
- [21] L. Cox, C. Murray, and B. Noble, "Pastiche: Making Backup Cheap and Easy," Proc. Symp. Operating Systems Design and Implementation (OSDI '02), Dec. 2002.
- [22] L. Cox and B. Noble, "Samsara: Honor among Thieves in Peer-to-Peer Storage," Proc. ACM Symp. Operating Systems Principles (SOSP '03), Oct. 2003.
- [23] J. Douceur, A. Adya, W. Bolosky, D. Simon, and M. Theimer, "Reclaiming Space from Duplicate Files in a Serverless Distributed File

- System,” Proc. IEEE Int’l Conf. Distributed Computing Systems (ICDCS ’02), July 2002.
- [24] J.R. Douceur and W.J. Bolosky, “A Large-Scale Study of FileSystem Contents,” Proc. SIGMETRICS ’99, pp. 59-70, 1999.
- [25] R. Ferreira, M.K. Ramanathan, A. Awan, A. Grama, and S. Jagannathan, “Search with Probabilistic Guarantees in Unstructured Peer-to-Peer Networks,” Proc. IEEE Int’l Conf. Peer-to-Peer Computing (P2P ’05), Aug. 2005.
- [26] Freenet, <http://freenet.sourceforge.net/>, 2007.
- [27] C. Gkantsidis, M. Mihail, and A. Saberi, “Random Walks in Peer-to-Peer Networks,” Proc. IEEE INFOCOM ’04, Mar. 2004.
- [28] Gnutella, <http://gnutella.wego.com/>, 2007.
- [29] Wong, J. and Hong, J. I., Making Mashups with Marmite: Towards End-User Programming for the Web, In Proc of CHI 2007, pp. 1435 - 1444.
- [30] Khan, B. H. (ed.) (1997). Web-Based Instruction. Englewood Cliffs, NJ: Educational Technology Publications.
- [31] Laudon, K. C., and Laudon, J. P. (2000). Management Information Systems, Organization and Technology in Networked Enterprise (6th Edition). Prentice Hall Inc.
- [32] Liaw, S. S., and Huang, H. M. (2000). Enhancing interactivity in Web-based instruction: A review of the literature. Educational Technology, 40(3), 41-45.