

Ατομική Διπλωματική Εργασία

**ΑΝΑΠΤΥΞΗ ΕΝΟΣ ΛΟΓΙΣΜΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ
ΓΙΑ ΔΙΑΧΕΙΡΙΣΗ ΠΡΟΪΟΝΤΩΝ ΣΕ
ΗΛΕΚΤΡΟΝΙΚΟ ΚΑΤΑΣΤΗΜΑ**

Νίκος Οδυσσέως

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2011

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**Ανάπτυξη ενός λογισμικού συστήματος
για διαχείριση προϊόντων σε
ηλεκτρονικό κατάστημα**

Νίκος Οδυσσέως

Επιβλέπων Καθηγητής
Γεώργιος Πάλλης

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2011

Ευχαριστίες

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ. Γιώργο Πάλλη για την υποστήριξη, την καθοδήγηση και τις συμβουλές που μου προσέφερε καθ' όλη τη διάρκεια αυτής της Διπλωματικής Εργασίας.

Ο δρόμος για την δημιουργική ενασχόληση με την επιστήμη είναι δύσκολος και μερικές φορές φαντάζει ιδιαίτερα δύσβατος. Σε τέτοια μονοπάτια είναι σημαντικό να έχεις συμπαραστάτες καλούς ανθρώπους.

Για αυτό το λόγο θα ήθελα να ευχαριστήσω τη μητέρα μου, τους συμφοιτητές μου, τους κολλητούς και φίλους μου, που ήταν πάντα κοντά μου, πραγματικοί σύντροφοι, που μου συμπαραστάθηκαν κάθε στιγμή της ημέρας για πάρα πολύ καιρό. Τους ευχαριστώ βαθύτατα, γιατί υπήρξαν στήριγμα σε κάθε ακαδημαϊκή μου προσπάθεια και δίχως αυτούς δεν θα είχα προχωρήσει όσο προχώρησα.

Τους ευχαριστώ επίσης που έθεσαν στην υπηρεσία μου τους πόρους τους, αλλά και τον χρόνο τους, είτε αφιερώνοντας τις γνώσεις που είχαν σε άλλους τομείς, είτε και στο ρόλο του αξιολογητή καθώς και όσους προσφέρθηκαν να με βοηθήσουν αλλά τελικά δεν χρειάστηκε.

Να αναφέρω τους:

Κυριάκος Δαμιανού, Κωνσταντίνα Κωνσταντίνου, Αντρέας Μιχαήλ, Δάφνη Σταυροπούλου, Γεωργία Χριστοδουλίδου, Μαρία Φιλίππου, Μαρία Στυλιανού.

Λευκωσία, 30 Μαΐου 2011

Νίκος Οδυσσέως

Περίληψη

Στο κεφάλαιο 1 γίνεται αναφορά στα κίνητρα που ώθησαν στην υλοποίηση του εν λόγω συστήματος καθώς και στη συνεισφορά που θα έχει σε εταιρείες λιανικών και χονδρικών πωλήσεων.

Στο κεφάλαιο 2 παρουσιάζεται η σχετική βιβλιογραφία, αναφέροντας τρία συστήματα αγοραπωλησίας προϊόντων μέσω διαδικτύου - Amazon, eBay και ASOS - ώστε να γίνουν σαφείς ο τρόπος και τα κριτήρια με τα οποία επιλέγονται τα προτεινόμενα προς αγορά προϊόντα από το σύστημα για τους χρήστες.

Στο κεφάλαιο 3 γίνεται καταγραφή των προδιαγραφών, των περιορισμών στη σχεδίαση, των εισόδων και εξόδων του συστήματος. Επίσης γίνεται καταγραφή οποιουδήποτε άλλου συστήματος ή συσκευής που πρόκειται να αλληλεπιδρά με το σύστημα αυτό. Γίνεται εύρεση, ανάλυση και τεκμηρίωση όλων των παραγόντων που επηρεάζουν και αλληλεπιδρούν με το σύστημα που πρόκειται να αναπτυχθεί για την υποστήριξη των βασικών λειτουργιών που χρειάζεται κάθε μικροεπιχείρηση στο χώρο των πωλήσεων.

Στο κεφάλαιο 4 παρουσιάζεται η φάση της σχεδίασης, δηλαδή το πώς στο σύστημα θα κάνει αυτά που καταγράφηκαν κατά τη φάση καθορισμού των προδιαγραφών. Στη φάση αυτή προσδιορίζονται τα μέρη από τα οποία θα αποτελείται το σύστημα, ο τρόπος λειτουργίας τους και ο τρόπος σύνδεσης μεταξύ τους. Επίσης προσδιορίζεται η μορφή των απαιτούμενων βάσεων δεδομένων και των οθονών εισαγωγής και εξαγωγής δεδομένων.

Στο κεφάλαιο 5 παρουσιάζεται η φάση της υλοποίησης όπου το σύστημα παίρνει μορφή με βάση τις προδιαγραφές και τη σχεδίασή του. Το σύστημα δημιουργείται με τις κατάλληλες γλώσσες προγραμματισμού και τα κατάλληλα εργαλεία, ελέγχεται και τέλος τίθεται σε λειτουργία. Για την υλοποίηση του συστήματος χρησιμοποιήθηκαν η Microsoft Visual Basic .NET και Microsoft SQL Server® 2008.

Στο κεφάλαιο 6 παρουσιάζεται το εγχειρίδιο χρήσης, όπου επεξηγούνται επακριβώς οι βασικές λειτουργίες του συστήματος και ο τρόπος χρήσης τους.

Τέλος, στο κεφάλαιο 7 γίνεται αξιολόγηση του συστήματος με σκοπό να εντοπιστούν λάθη και αδυναμίες στην υλοποίηση. Αναλύονται οι απαντήσεις των ερωτηματολογίων που δόθηκαν σε άτομα που δοκίμασαν το σύστημα. Αναφέρονται εισηγήσεις και βελτιώσεις που προτάθηκαν από τα ίδια άτομα.

Περιεχόμενα

Κεφάλαιο 1. Εισαγωγή	1
1.1. Κίνητρα	
1.2. Συνεισφορά	
1.3. Μεθοδολογία	
1.4. Εφαρμογή Μεθοδολογίας	
Κεφάλαιο 2. Σχετική βιβλιογραφία	6
2.1. ΑΛΓΟΡΙΘΜΟΙ ΣΥΣΤΑΣΗΣ.....	7
2.1.1. Συνεργατικό φιλτράρισμα (<i>Collaborative filtering</i>)	
2.1.2. Φιλτράρισμα με βάση το περιεχόμενο (<i>Content based filtering</i>)	
2.1.3. Υβριδικές προσεγγίσεις	
2.2. AMAZON	9
2.2.1. Αρχιτεκτονική συστήματος	
2.2.2. Επεξήγηση λειτουργικότητας	
2.3. eBay	13
2.3.1. Αρχιτεκτονική συστήματος	
2.3.2. Επεξήγηση λειτουργικότητας	
2.4. ASOS	17
2.4.1. Επεξήγηση λειτουργικότητας	
Κεφάλαιο 3. Ανάλυση προδιαγραφών.....	20
3.1. ΕΙΣΑΓΩΓΗ.....	21
3.1.1. Σκοπός	
3.1.2. Το σύστημα και οι αποδέκτες του	
3.1.3. Σύνοψη	
3.2. ΜΕΘΟΔΟΛΟΓΙΑ GANE & SARSEN.....	22
3.2.1. Διαγράμματα Ροής Δεδομένων	
3.2.2. Μηχανοργάνωση	
3.2.3. Εκλέπτυνση ροών δεδομένων	
3.2.4. Εκλέπτυνση λογικής διαδικασιών	
3.2.5. Εκλέπτυνση αποθήκευσης δεδομένων	
3.2.6. Ορισμός φυσικών πόρων	
3.2.7. Ορισμός προδιαγραφών για είσοδο/έξοδο δεδομένων	
3.2.8. Εκτίμηση μεγέθους συστήματος	
3.2.9. Απαιτήσεις σε <i>hardware</i>	
3.3. ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΑΝΑΛΥΣΗ	58
3.3.1. Διαγράμματα Χρήσης	
3.3.2. Διάγραμμα Κλάσεων	
3.4. Διάγραμμα Οντοτήτων - Συσχετίσεων	63
Κεφάλαιο 4. Σχεδίαση	65
4.1. ΕΙΣΑΓΩΓΗ.....	66
4.2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΧΕΔΙΑΣΗ	67
4.2.1. Ανάλυση Ροών Δεδομένων	
4.2.2. Διαγράμματα δομής	
4.3. ΑΝΑΛΥΤΙΚΗ ΣΧΕΔΙΑΣΗ	84
4.3.1. Οντότητες με όμοιες λειτουργίες	
4.3.2. Οντότητες με επιπλέον ή διαφορετικές λειτουργίες	

4.4. ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΧΡΗΣΤΗ	90
Κεφάλαιο 5. Υλοποίηση.....	93
5.1. ΕΙΣΑΓΩΓΗ.....	94
5.2. ΣΥΝΤΟΜΗ ΠΕΡΙΓΡΑΦΗ ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑΣ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ	94
5.3. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ	96
5.3.1. <i>Microsoft Visual Basic .NET</i>	
5.3.2. <i>Microsoft SQL Server® 2008</i>	
5.3.3. <i>Περιγραφή βάσης δεδομένων</i>	
5.4. ΔΥΣΚΟΛΑ ΣΗΜΕΙΑ ΣΤΟΝ ΚΩΔΙΚΑ	101
5.4.1. <i>Ενημέρωση ονομαστικών εορτών</i>	
5.4.2. <i>Αποστολή ευχετήριας κάρτας με ηλεκτρονικό ταχυδρομείο</i>	
5.4.3. <i>Ενημέρωση αποθήκης πρώτων υλών και αυτόματη προειδοποίηση εξάντλησης πρώτης ύλης</i>	
5.4.4. <i>Αποθήκευση εικόνων στη βάση δεδομένων</i>	
5.4.5. <i>Εκτύπωση αποδείξεων και τιμολογίων</i>	
5.4.6. <i>Εκτύπωση ποσού ολογράφως</i>	
5.4.7. <i>Εισηγήσεις ειδών ένδυσης</i>	
5.4.8. <i>Εισαγωγή/εξαγωγή δεδομένων</i>	
5.5. ΠΗΓΑΙΟΣ ΚΩΔΙΚΑΣ ΜΕ ΣΧΟΛΙΑ.....	105
5.5.1. <i>Διαχείριση πελατών</i>	
5.5.2. <i>Διαχείριση συνεργαζόμενων εταιρειών</i>	
5.5.3. <i>Διαχείριση πρώτων υλών</i>	
5.5.4. <i>Διαχείριση προϊόντων</i>	
5.5.5. <i>Διαχείριση ειδών ένδυσης</i>	
5.5.6. <i>Διαχείριση λιανικών πωλήσεων</i>	
5.5.7. <i>Διαχείριση χονδρικών πωλήσεων</i>	
5.5.8. <i>Βοηθητικές/Υποστηρικτικές υλοποιήσεις</i>	
Κεφάλαιο 6. Εγχειρίδιο χρήσης	123
6.1. Μενού έναρξης	124
6.1.1. <i>Ημερολόγιο, Εκκρεμούσες παραγγελίες, Εορτάζοντες</i>	
6.1.2. <i>Προειδοποίηση εξαντλημένων πρώτων υλών</i>	
6.2. Ρυθμίσεις παραμέτρων	126
6.3. Διαχείριση πελατών	127
6.3.1. <i>Εισαγωγή νέου πελάτη</i>	
6.3.2. <i>Ενημέρωση στοιχείων πελάτη</i>	
6.3.3. <i>Διαγραφή πελάτη</i>	
6.3.4. <i>Αναζήτηση πελάτη</i>	
6.3.5. <i>Αποστολή ηλεκτρονικού μηνύματος σε πελάτη</i>	
6.4. Διαχείριση συνεργαζόμενων εταιρειών.....	130
6.4.1. <i>Εισαγωγή νέας συνεργαζόμενης εταιρείας</i>	
6.4.2. <i>Ενημέρωση στοιχείων συνεργαζόμενης εταιρείας</i>	
6.4.3. <i>Διαγραφή συνεργαζόμενης εταιρείας</i>	
6.4.4. <i>Αναζήτηση συνεργαζόμενης εταιρείας</i>	
6.4.5. <i>Αποστολή ηλεκτρονικού μηνύματος σε συνεργαζόμενη εταιρεία</i>	
6.4.6. <i>Άνοιγμα ιστοσελίδας συνεργαζόμενης εταιρείας</i>	
6.5. Διαχείριση προμηθευτών.....	133
6.5.1. <i>Εισαγωγή νέου προμηθευτή</i>	
6.5.2. <i>Ενημέρωση στοιχείων προμηθευτή</i>	
6.5.3. <i>Διαγραφή προμηθευτή</i>	

6.5.4.	Αναζήτηση προμηθευτή	
6.5.5.	Αποστολή ηλεκτρονικού μηνύματος σε προμηθευτή	
6.6.	Διαχείριση πρώτων υλών	135
6.6.1.	Εισαγωγή νέας πρώτης ύλης	
6.6.2.	Ενημέρωση στοιχείων πρώτης ύλης	
6.6.3.	Διαγραφή πρώτης ύλης	
6.6.4.	Αναζήτηση προμηθευτή	
6.7.	Διαχείριση προϊόντων	138
6.7.1.	Εισαγωγή νέου προϊόντος	
6.7.2.	Ενημέρωση στοιχείων προϊόντος	
6.7.3.	Διαγραφή προϊόντος	
6.7.4.	Αναζήτηση προϊόντος	
6.8.	Διαχείριση συλλογών προϊόντων	141
6.8.1.	Δημιουργία νέας συλλογής	
6.8.2.	Εισαγωγή νέου προϊόντος σε συλλογή	
6.8.3.	Ενημέρωση στοιχείων προϊόντος	
6.8.4.	Διαγραφή προϊόντος	
6.8.5.	Αναζήτηση προϊόντων από τις συλλογές	
6.9.	Διαχείριση λιανικών πωλήσεων	144
6.9.1.	Εισαγωγή νέας παραγγελίας	
6.9.2.	Ενημέρωση στοιχείων παραγγελίας	
6.9.3.	Ακύρωση παραγγελίας	
6.9.4.	Έκδοση απόδειξης	
6.9.5.	Αναζήτηση παραγγελιών	
6.9.6.	Εκτύπωση τιμολογίου	
6.9.7.	Εκτύπωση απόδειξης	
6.10.	Διαχείριση χονδρικών πωλήσεων.....	146
6.10.1.	Εισαγωγή νέας παραγγελίας	
6.10.2.	Ενημέρωση στοιχείων παραγγελίας	
6.10.3.	Ακύρωση παραγγελίας	
6.10.4.	Έκδοση απόδειξης	
6.10.5.	Αναζήτηση παραγγελιών	
6.10.6.	Εκτύπωση τιμολογίου	
6.10.7.	Εκτύπωση απόδειξης	
6.11.	Εξαγωγή επαφών συστήματος σε αρχείο	148
6.12.	Εισαγωγή επαφών στο σύστημα από αρχείο.....	149
6.13.	Έξυπνες πωλήσεις	149
Κεφάλαιο 7.	Αξιολόγηση	150
7.1.	ΕΙΣΑΓΩΓΗ.....	151
7.2.	ΕΥΧΡΗΣΤΙΑ.....	151
7.3.	ΑΝΑΛΥΣΗ ΕΡΩΤΗΜΑΤΟΛΟΓΙΩΝ.....	151
7.3.1.	Χαμηλή βαθμολόγηση	
7.3.2.	Μελλοντική δουλειά	

Βιβλιογραφία

ΠΑΡΑΡΤΗΜΑ Α

A.1.	Κώδικας ενημέρωσης ονομαστικών νεορτών	A - 1
A.1.1.	Συνάρτηση <i>udf_modulo</i>	
A.1.2.	Συνάρτηση <i>udf_CalculateEaster</i>	

	A.1.3. Αποθηκευμένη διαδικασία <i>sp_CalculateHolidays</i>	
A.2.	Κώδικας αποστολής ευχετήριας κάρτας με ηλεκτρονικό ταχυδρομείο.....	A - 5
	A.2.1. Διαδικασία <i>sendmail()</i>	
	A.2.2. Διαδικασία <i>Today_Birthdays()</i>	
	A.2.3. Διαδικασία <i>Today_Namedays()</i>	
A.3.	Ενημέρωση αποθήκης πρώτων υλών και αυτόματη προειδοποίηση εξάντλησης πρώτης ύλης	A - 7
	A.3.1. Σκανδάλες πίνακα Φορεμάτων	
	A.3.2. Σκανδάλες πίνακα Ειδών Ένδυσης	
	A.3.3. Διαδικασία <i>Running_Out()</i>	
A.4.	Αποθήκευση εικόνων	A - 10
	A.4.1. Διαδικασία <i>ConvertImage()</i>	
A.5.	Εκτύπωση αποδείξεων και τιμολογίων.....	A - 10
	A.5.1. Παράδειγμα σύνδεσης κώδικα <i>Visual Basic</i> με <i>Microsoft Word</i>	
A.6.	Εκτύπωση ποσού ολογράφως	A - 11
	A.6.1. Συνάρτηση <i>NumberToString()</i>	
	A.6.2. Συνάρτηση <i>GroupToWords()</i>	
A.7.	Εισηγήσεις ειδών ένδυσης	A - 14
	A.7.1. Αποθηκευμένη διαδικασία <i>sp_SmartSeller</i>	
A.8.	Εισαγωγή/εξαγωγή δεδομένων.....	A - 17
	A.8.1. Αποθηκευμένη διαδικασία <i>sp_export2csv</i>	
	A.8.2. Συνάρτηση <i>createQuery()</i>	

ΠΑΡΑΡΤΗΜΑ Β

Κεφάλαιο 1. Εισαγωγή

1.1. Κίνητρα	2
1.2. Συνεισφορά.....	2
1.3. Μεθοδολογία.....	2
1.4. Εφαρμογή Μεθοδολογίας.....	4

1.1. Κίνητρα

Η υλοποίηση του εν λόγω συστήματος απαιτούσε την εφαρμογή διαφόρων τεχνικών και αρχών που διδάχτηκαν στο προπτυχιακό επίπεδο. Κάτι τέτοιο αποτέλεσε επιπλέον κίνητρο για την ανάπτυξη του εν λόγω συστήματος, αφού έδωσε την ευκαιρία εφαρμογής των όσων διδάχτηκαν στα πέντε χρόνια προπτυχιακών σπουδών.

Επιπλέον η συγκεκριμένη εφαρμογή απαιτούσε την εμπλοκή βάσεως δεδομένων για την σωστή λειτουργία της. Το πεδίο των Βάσεων Δεδομένων αποτελεί πολύ σημαντικό κομμάτι στις περισσότερες εφαρμογές συστημάτων και παρουσιάζει αρκετές προκλήσεις στη δημιουργία και διαχείρισή του. Ο συνδυασμός επομένως της ανάπτυξης ενός πληροφοριακού συστήματος με μια βάση δεδομένων για την διαχείριση των στοιχείων του συστήματος είναι ένα πεδίο μεγάλου ενδιαφέρον.

Η ανάγκη για αυτοματοποίηση των λειτουργιών οποιασδήποτε μη μηχανογραφημένης εταιρίας ώστε να διευκολυνθεί η διεκπεραίωση των λειτουργιών της, ήταν ένα ακόμα κίνητρο για την υλοποίηση ενός τέτοιου συστήματος.

1.2. Συνεισφορά

Το σύστημα παρέχει διευκολύνσεις σε άτομα του χώρου των πωλήσεων, υλοποιώντας όλες τις χρήσιμες και απαραίτητες λειτουργίες που χρειάζεται κάθε μικροεπιχείρηση στο χώρο των πωλήσεων. Είναι εύκολο στην λειτουργία και προσιτό για κάθε είδους χρήστη που θέλει να το χρησιμοποιήσει. Παρέχει πολλές χρήσιμες πληροφορίες, όπως την ποσότητα των αποθεμάτων που υπάρχουν στις αποθήκες ώστε να γνωρίζει ανά πάσα στιγμή την διαθέσιμη ποσότητα των προϊόντων που έχει καθώς και πληροφορίες σχετικές με τους πελάτες, όπως την ημερομηνία των γενεθλίων τους και την ονομαστική τους εορτή για αποστολή ευχετήριου μηνύματος. Με αυτό τον τρόπο ενδυναμώνονται οι σχέσεις μεταξύ πελάτη και πωλητή.

1.3. Μεθοδολογία

Ο κύκλος ζωής ενός λογισμικού αποτελεί την παλαιότερη μέθοδο ανάπτυξης συστημάτων η οποία χρησιμοποιείται μέχρι και σήμερα σε μικρά και μεγάλα έργα ανάπτυξης λογισμικού. Βάση αυτής της μεθοδολογίας, ένα λογισμικό μπορεί να είναι λειτουργήσιμο και να ανταποκρίνεται στις προσδοκίες του χρήστη για μια δεδομένη χρονική περίοδο έχοντας αρχή, μέση και τέλος. Ο κύκλος ζωής αυτός αποτελείται από τα πιο κάτω έξι στάδια:

1. Απαιτήσεις και Προδιαγραφές
2. Ανάλυση
3. Σχεδιασμός
4. Υλοποίηση
5. Εγκατάσταση
6. Συντήρηση

Για να ολοκληρωθεί ένα στάδιο και να ξεκινήσει το επόμενο, είναι απαραίτητο να διεκπεραιωθούν οι βασικές δραστηριότητες του σταδίου.

Η όλη εργασία που εκπονείται κατά τον κύκλο ζωής πραγματοποιείται ισάξια από τους τελικούς χρήστες και ειδικούς των συστημάτων πληροφοριών. Το μεγαλύτερο μέρος των σταδίων ανάλυσης συστήματος, σχεδιασμού και υλοποίησης εκπονείται από τους αναλυτές και τους προγραμματιστές του συστήματος. Οι τελικοί χρήστες είναι υπεύθυνοι να αναφέρουν και να περιγράψουν τις ανάγκες τους και να αξιολογούν τη δουλειά του τεχνικού προσωπικού. Αν οι τελικοί χρήστες συμφωνούν, το τρέχον στάδιο ολοκληρώνεται με μια επίσημη συμφωνία μεταξύ των χρηστών και των ειδικών του συστήματος.

Το στάδιο *Απαιτήσεις και Προδιαγραφές* απαντά σε ερωτήματα τύπου "Γιατί υπάρχει ανάγκη για την υλοποίηση ενός νέου συστήματος;" και "Τι πρέπει να επιτευχθεί;" Σε αυτό το στάδιο προσδιορίζεται κατά πόσο η εταιρεία αντιμετωπίζει κάποιο πρόβλημα κι αν αυτό μπορεί να επιλυθεί με την υλοποίηση νέου συστήματος ή με την τροποποίηση μέρους του υπάρχοντος συστήματος. Επιπλέον, στο στάδιο αυτό γίνεται προσδιορισμός των γενικών στόχων, καθορισμός του σκοπού του έργου και σχεδίαση του χρονοδιαγράμματος έτσι ώστε να παρουσιαστεί στη διοίκηση του οργανισμού ένα πλάνο πορείας.

Το στάδιο της *Ανάλυσης* αναλύει λεπτομερώς τα προβλήματα που παρατηρούνται στο υπάρχον σύστημα, περιγράφει πώς οι στόχοι που καθορίστηκαν στο στάδιο των Απαιτήσεων και Προδιαγραφών μπορούν να επιτευχθούν μέσω εναλλακτικών λύσεων. Για κάθε εναλλακτική λύση εξετάζεται η σκοπιμότητα της ώστε να γίνει η σωστή επιλογή από το διοικητικό προσωπικό της εταιρείας. Η ανάλυση του συστήματος θέλει τους αναλυτές και τους προγραμματιστές του συστήματος να συλλέγουν πληροφορίες, να ερευνούν και να μελετούν έγγραφα και αναφορές που δημιουργούνται από το υπάρχον σύστημα και να παρατηρούν τον τρόπο λειτουργίας του. Οι δραστηριότητες αυτές πραγματοποιούνται μέσω ερωτηματολογίων προς τους χρήστες και προσωπικών συνεντεύξεων. Τα πορίσματα σε πληροφορίες που εξάγονται από αυτές τις δραστηριότητες χρησιμοποιούνται για να καθοριστούν οι ανάγκες του συστήματος. Επιπλέον, κατά την ανάλυση περιγράφονται εκτενώς όλες οι υπόλοιπες δραστηριότητες του κύκλου ζωής και των εργασιών κάθε φάσης.

Στο στάδιο του *Σχεδιασμού* σχεδιάζονται σε λογικό και φυσικό επίπεδο οι προδιαγραφές του υπό υλοποίηση συστήματος και παρουσιάζονται μέσω διαγραμμάτων ροής δεδομένων, δομής προγραμμάτων, καθώς και ροής του συστήματος. Αυτό θα βοηθήσει έτσι ώστε οι προγραμματιστές να δημιουργήσουν ορθά την κάθε διεργασία και να διευκολυνθούν στην όλη υλοποίηση.

Το στάδιο της *Υλοποίησης* μετατρέπει τα διαγράμματα του σταδίου της σχεδίασης σε κώδικα προγραμμάτων λογισμικού. Για τη σωστή και προσδοκώμενη λειτουργία του συστήματος, απαιτείται η συνεργασία μεταξύ αναλυτών και προγραμματιστών ώστε να γραφτεί προσαρμοσμένος για την περίπτωση κώδικας προγραμμάτων.

Στο στάδιο της *Εγκατάστασης* πραγματοποιούνται τα τελικά βήματα ώστε το νέο ή τροποποιημένο σύστημα να τεθεί σε λειτουργία. Μέσω δοκιμών και ελέγχων, διαπιστώνεται κατά πόσο το λογισμικό ανταποκρίνεται ικανοποιητικά στις αρχικές προσδοκίες του οργανισμού, τόσο από τεχνική όσο και από επιχειρηματική και λειτουργική άποψη. Οι μελλοντικοί χρήστες του συστήματος εκπαιδεύονται κατάλληλα για να λειτουργήσουν σωστά το νέο σύστημα. Το εγχειρίδιο χρήσης, το οποίο παραδίδεται στον οργανισμό, περιγράφει λεπτομερώς την εγκατάσταση και όλες τις λειτουργίες του νέου συστήματος.

Κατά το στάδιο της *Συντήρησης* το σύστημα αξιολογείται με βάση τη χρήση του. Σκοπός της αξιολόγησης είναι να διαπιστωθεί σε ποιο βαθμό το νέο σύστημα ανταποκρίνεται στις αρχικούς στόχους. Αρνητικά σχόλια συνεπάγονται αναθεωρήσεις ή τροποποιήσεις στο υλοποιημένο σύστημα. Η

απόδοση του συστήματος, καθώς και η υποκειμενική ικανοποίηση των χρηστών του, εξαρτώνται από τη συνεχή συντήρησή του.

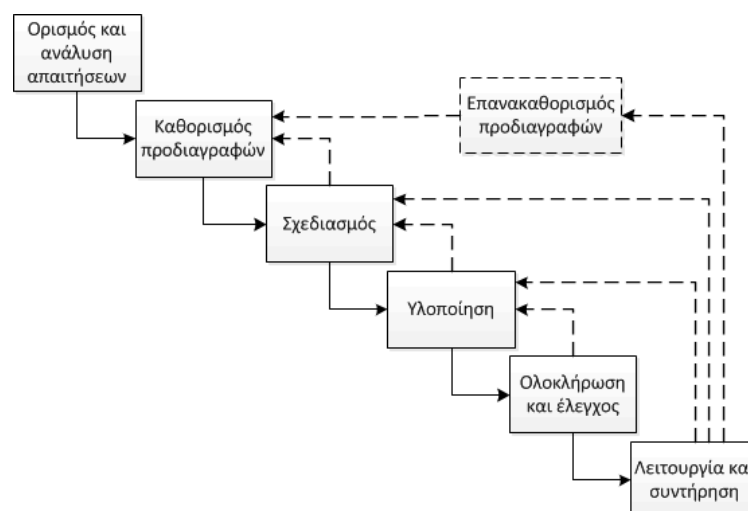
1.4. Εφαρμογή Μεθοδολογίας

Η πιο πάνω μεθοδολογία εφαρμόστηκε κάνοντας χρήση του μοντέλου καταρράκτη.

Το μοντέλο καταρράκτη είναι ένα κλασσικό μοντέλο κύκλου ζωής ανάπτυξης λογισμικού που χρησιμοποιεί μια γραμμική και ακολουθιακή μέθοδο. Η μέθοδος αυτή παρέχει στόχους σε κάθε φάση της ανάπτυξης του συστήματος. Το μοντέλο αυτό συνηθίζεται να χρησιμοποιείται ως μέθοδο ανάπτυξης κομματιών μεγάλου λογισμικού. Το όνομα του μοντέλου πηγάζει από τη μορφολογική του απεικόνιση, καθώς είναι εμφανής η κάθετη μετάβαση που γίνεται από φάση σε φάση. Λόγω της έλλειψης επανάληψης ή επικάλυψης στα βήματα της μεθόδου, στο μοντέλο καταρράκτη παρατηρείται μια απλοποίηση του χρονοπρογραμματισμού των εργασιών του. Σε αυτό συμβάλλει και το γεγονός ότι πραγματοποιείται συνεχής έλεγχος και τεκμηρίωση του κάθε βήματος και μάλιστα η φάση δεν θεωρείται τελειωμένη πριν γραφεί και εγκριθεί η τεκμηρίωσή της.

Το μοντέλο καταρράκτη αποτελείται από τις ακόλουθες φάσεις:

- **Ορισμός και ανάλυση απαιτήσεων:** Καθορίζονται οι απαιτήσεις των χρηστών του συστήματος, καθώς επίσης οι στόχοι και τυχόν περιορισμοί του συστήματος.
- **Καθορισμός προδιαγραφών:** Καταγράφονται οι απαιτήσεις και οι λειτουργικές προδιαγραφές του συστήματος.
- **Σχεδιασμός:** Οι λειτουργίες του λογισμικού παρουσιάζονται μέσω αρχιτεκτονικών και αναλυτικών διαγραμμάτων για να μπορέσουν να μεταφραστούν σε εκτελέσιμο κώδικα.
- **Υλοποίηση:** Αναπτύσσεται ο κώδικας της εφαρμογής.
- **Ολοκλήρωση και έλεγχος:** Οι κώδικες των επιμέρους ενότητων συνενώνονται και ελέγχεται το ολοκληρωμένο πλέον σύστημα. Μετά το πέρας του ελέγχου παραδίδεται στο χρήστη.
- **Λειτουργία και συντήρηση:** Το σύστημα εγκαθίσταται και τίθεται σε λειτουργία. Κατά τη συντήρηση διορθώνονται λάθη που δεν είχαν εντοπιστεί προηγουμένως, βελτιώνεται ο κώδικας και γενικά οι λειτουργίες του συστήματος.



Διάγραμμα 1.1 - Μοντέλο καταρράκτη

Το μοντέλο καταρράκτη, παρόλο που χρησιμοποιείται ευρέως για την ανάπτυξη μεγάλων συστημάτων, παρουσιάζει κάποια μειονεκτήματα. Η αναθεώρηση του υλοποιημένου συστήματος δεν είναι εφικτή. Αυτό αποτελεί ένα από τα βασικά μειονεκτήματα αυτού του μοντέλου. Αυτό γιατί η προσαρμογή των αλλαγών μετά από την ολοκλήρωσή του συστήματος είναι δύσκολη και ακριβή. Επομένως, το μοντέλο καταρράκτη χρησιμοποιείται μόνο όταν οι απαιτήσεις του υπό υλοποίηση λογισμικού έχουν γίνει πλήρως κατανοητές.

Κεφάλαιο 2. Σχετική βιβλιογραφία

2.1.	ΑΛΓΟΡΙΘΜΟΙ ΣΥΣΤΑΣΗΣ.....	7
2.1.1.	Συνεργατικό φιλτράρισμα (<i>Collaborative filtering</i>)	
2.1.2.	Φιλτράρισμα με βάση το περιεχόμενο (<i>Content based filtering</i>)	
2.1.3.	Υβριδικές προσεγγίσεις	
2.2.	AMAZON	9
2.2.1.	Αρχιτεκτονική συστήματος	
2.2.2.	Επεξήγηση λειτουργικότητας	
2.3.	eBay	13
2.3.1.	Αρχιτεκτονική συστήματος	
2.3.2.	Επεξήγηση λειτουργικότητας	
2.4.	ASOS	17
2.4.1.	Επεξήγηση λειτουργικότητας	

2.1. ΑΛΓΟΡΙΘΜΟΙ ΣΥΣΤΑΣΗΣ

Οι περισσότερες διαδικτυακές σελίδες που ασχολούνται με το εμπόριο προϊόντων οφείλουν να ενσωματώνουν στα συστήματά τους κάποιες τεχνολογίες προκειμένου να βοηθήσουν τους πελάτες στις αγορές τους. Αυτό, αποτελεί το λεγόμενο *πρόβλημα σύστασης* το οποίο περιορίζεται στο πρόβλημα της αξιολόγησης των στοιχείων που δεν έχει δει ένας χρήστης. Για να απαληφθεί αυτό το πρόβλημα, χρησιμοποιείται ο λεγόμενος *αλγόριθμος σύστασης* με σκοπό να προτείνει προϊόντα στους πελάτες.

Ο *αλγόριθμος σύστασης* (recommendationsystem ή RS) είναι ένα σύστημα με το οποίο τα διαδικτυακά καταστήματα μπορούν να βελτιώσουν την απόδοσή τους καλύπτοντας τις ανάγκες και τις προτιμήσεις κάθε πελάτη ξεχωριστά. Μπορούν επιπλέον να αυξήσουν το ποσοστό ικανοποίησης των πελατών τους και να επικυρώσουν την εμπιστοσύνη τους. Ένας αλγόριθμος σύστασης εκμεταλλεύεται τις απόψεις των χρηστών του και μπορεί να υποστηρίξει τον πελάτη ώστε να προσδιορίσει τις πληροφορίες ή τα προϊόντα για τα οποία ενδιαφέρεται και είναι κοντά στις ανάγκες και προτιμήσεις του. Οι συστάσεις μπορούν να παρουσιαστούν με πολλές μορφές, όπως εξατομικευμένες προσφορές, τιμές, προϊόντα ή υπηρεσίες, ταξινόμηση, εμφάνιση, προαιρετικές εξηγήσεις ή αναλυτικές πληροφορίες. Ένα παράδειγμα συστάσεων είναι το ιστορικό των αγορών κάποιου χρήστη με το οποίο το σύστημα μπορεί να προβλέψει τις προτιμήσεις των χρηστών.

Υπάρχουν διαφορετικές προσεγγίσεις αυτού του αλγόριθμου οι οποίες διαφέρουν ως προς τον τρόπο με τον οποίο συλλέγουν τα απαραίτητα στοιχεία με τα οποία θα γίνει η επιλογή των προτεινόμενων προϊόντων σε κάθε πελάτη. Συγκεκριμένα, ταξινομούνται σε τρεις βασικές κατηγορίες: το *συνεργατικό φιλτράρισμα*, το *φιλτράρισμα με βάση το περιεχόμενο* και τις *υβριδικές προσεγγίσεις*.

Και στις τρεις περιπτώσεις, ο χρήστης αντιπροσωπεύεται από ένα n -διάστατο διάνυσμα στοιχείων, όπου το n είναι ο αριθμός ευδιάκριτου καταλόγου αντικειμένων. Τα συστατικά του διανύσματος είναι θετικά για τα αγορασμένα ή θετικά εκτιμημένα στοιχεία και αρνητικά για τα αρνητικά εκτιμημένα στοιχεία.

2.1.1. Συνεργατικό φιλτράρισμα (Collaborative filtering)

Το *συνεργατικό φιλτράρισμα* (collaborative filtering), λαμβάνει υπόψη την «ομοιότητα» των χρηστών ώστε να παραχθούν συστάσεις. Με άλλα λόγια, γίνονται προτάσεις στο χρήστη με βάση τις προτιμήσεις χρηστών με παρόμοιες προτιμήσεις. Όταν ο χρήστης επιλέξει οποιοδήποτε προϊόν, αμέσως του εμφανίζονται προς εισήγηση άλλα προϊόντα τα οποία αγόρασαν μαζί με το εν λόγω προϊόν και άλλοι χρήστες και τα οποία πιθανόν να έχουν άμεση σχέση με το προϊόν. Τα συνεργατικά εισηγητικά συστήματα προσπαθούν να προβλέψουν τη χρησιμότητα των στοιχείων για έναν ιδιαίτερο χρήστη βασισμένα στα στοιχεία που έχουν αξιολογηθεί από άλλους χρήστες.

Η χρήση του συνεργατικού φιλτραρίσματος για να παραχθούν οι συστάσεις είναι υπολογιστικά ακριβή. Στη χειρίστη περίπτωση η πολυπλοκότητα είναι της τάξης του $O(m \cdot n)$, όπου το m είναι ο αριθμός των πελατών και το n είναι ο αριθμός των προϊόντων του καταλόγου, δεδομένου ότι εξετάζει m πελάτες και μέχρι n στοιχεία για κάθε πελάτη. Λόγω του ότι το μέσο διάνυσμα πελατών είναι εξαιρετικά αραιό, η απόδοση του αλγόριθμου τείνει της τάξεως του $O(m+n)$. Η εύρεση κάθε πελάτη είναι περίπου $O(m)$, και όχι $O(m \cdot n)$, επειδή σχεδόν όλα τα διανύσματα των πελατών περιέχουν έναν μικρό αριθμό στοιχείων, ανεξάρτητα από το πλήθος του καταλόγου των προϊόντων. Υπάρχουν όμως μερικοί πελάτες οι οποίοι έχουν αγοράσει ή έχουν εκτιμήσει ένα σημαντικό ποσοστό του καταλόγου αυτού, απαιτώντας

χρόνο επεξεργασίας $O(n)$. Κατά συνέπεια, η τελική απόδοση του αλγόριθμο είναι περίπου $O(m+n)$. Ακόμα και έτσι, για πολύ μεγάλα σύνολα στοιχείων - όπως 10 εκατομμύρια ή περισσότεροι πελάτες και 1 εκατομμύριο ή περισσότερα προϊόντα καταλόγου, ο αλγόριθμος παρουσιάζει σοβαρά προβλήματα απόδοσης και μέτρησης.

2.1.2. Φιλτράρισμα με βάση το περιεχόμενο (Contentbasedfiltering)

Το φιλτράρισμα με βάση το περιεχόμενο (contentbasedfiltering), λαμβάνει υπόψη τις παραμέτρους του χρήστη σε σχέση με τα χαρακτηριστικά των προϊόντων. Σε αυτή την προσέγγιση, θα γίνουν προτάσεις στον χρήστη με βάση τα στοιχεία στα οποία έχει δείξει προτίμηση. Ο χρήστης, είτε θα εισάγει κατευθείαν ακριβώς το περιεχόμενο του προϊόντος που επιθυμεί, είτε θα εισάγει απλά μια λέξη-κλειδί και το σύστημα είναι υπεύθυνο να του εμφανίσει όλες τις πιθανές επιλογές που μπορεί να υπάρχουν.

Είναι δυνατό να υπολογιστεί η ομοιότητα μεταξύ δύο στοιχείων με διάφορους τρόπους, αλλά η πιο κοινή μέθοδος είναι η χρήση του *μέτρου συνημίτονου*. Κατά τη μέθοδο αυτή, η ομοιότητα μεταξύ δύο χρηστών A και B υπολογίζεται από το συνημίτονο της γωνίας μεταξύ των δύο διανυσμάτων:

$$\text{Ομοιότητα } (\vec{A}, \vec{B}) = \text{Συνημίτονο } (\vec{A}, \vec{B})$$

Κάθε διάνυσμα αντιστοιχεί σε ένα προϊόν παρά σε έναν πελάτη και οι διαστάσεις του διανύσματος m αντιστοιχούν στους πελάτες που έχουν αγοράσει εκείνο το αντικείμενο. Αυτός ο υπολογισμός του πίνακα των όμοιων στοιχείων είναι εξαιρετικά χρονοβόρος, μιας και η πολυπλοκότητα της χείριστης περίπτωσης του είναι $O(n^2 \cdot m)$. Παρόλα αυτά, κατά την εφαρμογή του είναι πιο κοντά στο $O(n \cdot m)$, καθώς περισσότεροι πελάτες πραγματοποιούν πολύ λίγες αγορές.

2.1.3. Υβριδικές προσεγγίσεις

Τέλος, οι υβριδικές προσεγγίσεις είναι ουσιαστικά ένας συνδυασμός των δύο πιο πάνω μεθόδων, που βοηθά στη αποφυγή τυχόν περιορισμών που μπορεί να παρουσιαστούν με την χρησιμοποίηση των προηγούμενων μεθόδων. Οι διαφορετικοί τρόποι συνδυασμού των μεθόδων αυτών είναι εφαρμόζοντας τις μεθόδους ξεχωριστά και συνδυάζοντας τις προβλέψεις τους, κατασκευάζοντας ένα γενικό πρότυπο που ενσωματώνει χαρακτηριστικά και των δύο μεθόδων και ενσωματώνοντας κάποια χαρακτηριστικά της μεθόδου με βάση το περιεχόμενο σε μια προσέγγιση συνεργατικού φιλτραρίσματος και το αντίθετο.

Η πολυπλοκότητα της προσέγγισης αυτής κυμαίνεται μεταξύ της πολυπλοκότητας των δυο προηγούμενων προσεγγίσεων, αφού η υβριδική προσέγγιση αποτελεί συνδυασμό αυτών. Επομένως, η χείριστη περίπτωση του αλγορίθμου αυτού είναι $O(n^2 \cdot m)$ από το φιλτράρισμα με βάση το περιεχόμενο, ενώ η βέλτιστη περίπτωση είναι της τάξης του $O(n + m)$ από το συνεργατικό φιλτράρισμα.

Στη συνέχεια αναλύονται τρία από τα μεγαλύτερα διαδικτυακά πολυκαταστήματα – Amazon, eBay και Asos – περιγράφοντας τόσο τη λειτουργία τους, όσο και τον αλγόριθμο σύστασης που χρησιμοποιούν.

2.2. AMAZON



Εικόνα 2.1

Η Amazon ιδρύθηκε από τον Jeff Bezos το 1995 και πήρε την ονομασία του από τον ποταμό του Αμαζονίου, αφενός για να υποδείξει το μεγάλο του μέγεθος, καθώς ο Αμαζόνιος αποτελεί έναν από τους μεγαλύτερους ποταμούς του κόσμου και αφετέρου για να εμφανίζεται στην αρχή κάθε αλφαβητικής λίστας, καθώς αρχίζει με Α.

Είναι μια πολυεθνική εταιρεία ηλεκτρονικού εμπορίου η οποία αρχικά ξεκίνησε ως διαδικτυακό βιβλιοπωλείο, στην πορεία όμως εξελίχθηκε και είναι σήμερα ο μεγαλύτερος διαδικτυακός χώρος πωλήσεων. Γι' αυτό άλλωστε από το 2000, το λογότυπο της εταιρείας συνοδεύεται από ένα βέλος από το Α στο Ζ για να δείξει την ικανοποίηση των πελατών από τη μια και την πληθώρα των προϊόντων που παρέχει από την άλλη (εικόνα 2.1).

Αξιοσημείωτο είναι το γεγονός ότι, ενώ αρχικά δεν αναμενόταν κάποιο όφελος από την επιχείρηση, τουλάχιστον στα αρχικά στάδια της λειτουργίας της, εντούτοις όχι μόνο αύξησε τις πωλήσεις της αλλά το έκανε και σε μια περίοδο όπου όλες οι άλλες διαδικτυακές εταιρείες είχαν κλείσει. Αυτό οφείλεται κυρίως στην εξυπνάδα και την εφευρετικότητα του δημιουργού. Καταρχήν, δημιούργησε μια τεράστια βάση δεδομένων με εκατομμύρια τίτλους βιβλίων από τους οποίους ο κάθε ενδιαφερόμενος μπορεί να επιλέξει οτιδήποτε αναζητεί. Η αναζήτηση δεν περιορίζεται στον ακριβή τίτλο του βιβλίου, αλλά ο χρήστης μπορεί να αναζητήσει οποιοδήποτε βιβλίο πληκτρολογώντας μόνο μια λέξη.

Οι υπηρεσίες που προσφέρει η Amazon είναι πάρα πολλές και δεν περιορίζονται μόνο στην αγορά ή πώληση βιβλίων, αλλά επεκτείνονται και σε άλλους τομείς όπως η πώληση ταινιών, παιχνιδιών, ηλεκτρονικών ειδών και από είδη ένδυσης μέχρι είδη υγείας και ομορφιάς. Αυτός είναι και ο κύριος λόγος για τον οποίο ο αριθμός των χρηστών που χρησιμοποιεί η Amazon αυξήθηκε με θεαματικούς ρυθμούς και προσέλκυσε γρήγορα ένα μεγάλο και μορφωμένο κοινό.

Ένα άλλο χαρακτηριστικό της Amazon είναι το γεγονός ότι δεν παραμένει στάσιμη, αλλά αναβαθμίζεται με την έννοια ότι προωθεί συνεχώς νέα προϊόντα στους πελάτες του, κρατώντας αμείωτο το ενδιαφέρον τους για καινούργιες αγορές. Μάλιστα, ανά περιόδους, παρέχει διάφορες προσφορές με σκοπό την προώθηση προϊόντων αλλά και διαφήμισης της εταιρείας. Για καλύτερη εξυπηρέτηση των πελατών της, κυρίως του ευρωπαϊκού κοινού, επεκτάθηκε και δημιούργησε παραρτήματα και σε άλλες χώρες ώστε να εξασφαλίσει φθηνότερα μεταφορικά, όπως για παράδειγμα την amazon.co.uk οποία βρίσκεται στο Ηνωμένο Βασίλειο.

Η εξυπηρέτηση και η χρησιμότητα της Amazon είναι τέτοια ώστε καμία άλλη διαδικτυακή επιχείρηση δεν κατάφερε να ξεπεράσει. Ουσιαστικά η εταιρεία χρησιμοποιεί έναν ιδιαίτερο τρόπο χειρισμού των πελατών της έτσι ώστε να τους ικανοποιεί και να τους αποτρέπει από τη δημιουργία δικού τους διαδικτυακού βιβλιοπωλείου. Συγκεκριμένα, η στρατηγική που ακολουθεί είναι η ενθάρρυνση των χρηστών στη δημιουργία ενός ειδικού περιβάλλοντος, μέσα στα όρια της Amazon, στο οποίο ο χρήστης μπορεί να πωλεί δικά του προϊόντα μέσα από τον δικό του διαδικτυακό χώρο. Στη συνέχεια, για κάθε πώληση η Amazon προσφέρει κάποιο ποσοστό επί της τιμής του κάθε προϊόντος

που πωλείται μέσα από την ιστοσελίδα του εν λόγω χρήστη. Έτσι, εξασφαλίζει ότι οι χρήστες του θα ικανοποιηθούν στην περίπτωση που θέλουν να προωθήσουν δικά τους προϊόντα, θα έχουν οικονομικό συμφέρον από αυτή τους τη δραστηριότητα και θα κάνουν πιο ελκυστική την ιστοσελίδα τους για τους χρήστες του διαδικτύου γενικά.

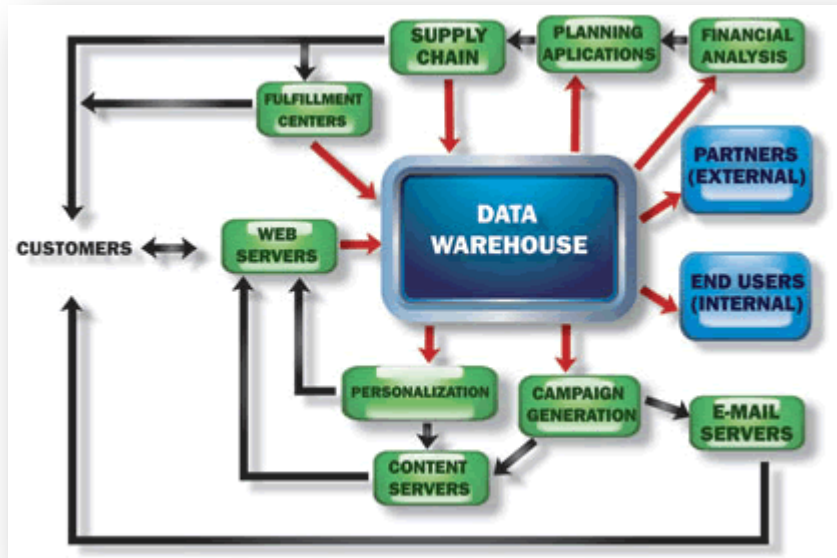
Η Amazon χρησιμοποιεί την προσέγγιση *φιλτραρίσματος με βάση το περιεχόμενο* στη περίπτωση που ο χρήστης είναι εγγεγραμμένος στο σύστημα. Με λίγα λόγια, σε κάθε πελάτη προτείνονται διαφορετικά προϊόντα, φτάνει ο πελάτης να είναι ήδη εγγεγραμμένος στο κατάστημα. Αν ο χρήστης έχει ήδη εγγραφεί, τότε η εταιρεία προτείνει προϊόντα με βάση τα ενδιαφέροντά του, την βαθμολογία που έχει βάλει στα προϊόντα, αλλά και με βάση το είδος του προϊόντος που ψάχνει. Τα στοιχεία αυτά θα πρέπει να τα εισάγει ο ίδιος ο χρήστης καθώς του ζητούνται από το κατάστημα μόλις κάνει την εγγραφή του. Ένας άλλος τρόπος για να προτείνει προϊόντα με βάση την ίδια προσέγγιση είναι μέσω της εφαρμογής *WishList* που προσφέρει η Amazon. Συγκεκριμένα, με το πάτημα αυτού του εικονιδίου, ο χρήστης μπορεί να εισάγει μέσα σε αυτή τη λίστα τα προϊόντα που θα ήθελε να αγοράσει στο μέλλον, οπότε το σύστημα προτείνει προϊόντα με βάση τα προϊόντα που βρίσκονται σε αυτή τη λίστα.

Στην περίπτωση που ο χρήστης δεν έχει εγγραφεί στο σύστημα, τότε χρησιμοποιείται το *συνεργατικό φιλτράρισμα* για την προβολή προϊόντων που έχουν τις μεγαλύτερες πωλήσεις ή τις πιο πρόσφατες πωλήσεις. Σε αυτή την περίπτωση, ο αλγόριθμος της Amazon είναι ένας πίνακας συσχέτισης για τα προϊόντα που έχουν πωληθεί την τελευταία ώρα, τα προϊόντα που κοιτάζουν άλλοι χρήστες τη δεδομένη στιγμή ή τα προϊόντα που συλλέγουν την υψηλότερη βαθμολογία, σύμφωνα με την βαθμολογία που παίρνει το κάθε προϊόν κάθε φορά που αγοράζεται.

2.2.1. Αρχιτεκτονική συστήματος

Η τεχνολογία στην οποία είναι υλοποιημένο η Amazon είναι εξ ολοκλήρου λογισμικό Linux. Από το 2005, η Amazon έχει τις τρεις μεγαλύτερες βάσεις δεδομένων σε Linux στον κόσμο, με συνολική χωρητικότητα 7,8 TB, 18,5 TB και 24,7 TB αντίστοιχα. Η κεντρική αποθήκη δεδομένων της Amazon αποτελείται από 28 διακομιστές, με τέσσερις κεντρικές μονάδες επεξεργασίας ανά κόμβο, οι οποίοι τρέχουν με λογισμικού βάσεων δεδομένων Oracle 9i.

Η αποθήκη δεδομένων χαρακτηρίζεται από τρεις λειτουργίες: επερώτηση, δεδομένα ιστορικών και την πρωταρχική λειτουργία βάσης δεδομένων που εξαγει δεδομένα από μία πηγή και τα τροφοδοτεί σε μια άλλη, ETL (extract, transform and load - απόσπασε, μετασχημάτισε και φόρτωσε). Οι διακομιστές επερωτήσεων (χωρητικότητας 24,7 TB) περιέχουν 15 TB δομικών στοιχείων. Οι διακομιστές δεδομένων ιστορικών (χωρητικότητας 18,5 TB) περιέχουν 14 TB δομικών στοιχείων. Τέλος, το σύμπλεγμα διακομιστών ETL (χωρητικότητας 7,8 TB) περιέχει 5 TB δομικών στοιχείων. Η αρχιτεκτονική της Amazon χειρίζεται κάθε μέρα εκατομμύρια συναλλαγές, καθώς και επερωτήσεις από περισσότερους από μισό εκατομμύριο μεταπωλητές. Σύμφωνα με μια έκθεση που δημοσιεύθηκε από την Oracle, μετά τη μετακίνηση της αποθήκης δεδομένων της Amazon σε λογισμικό Linux το 2003 και το 2004, η νέα αρχιτεκτονική του συστήματος φαίνεται στην εικόνα 2.2.



Εικόνα 2. 2

Ο όγκος πωλήσεων της Amazon ανέρχεται σε εκατοντάδες χιλιάδες χρήστες, οι οποίοι στέλνουν καθημερινά τους αριθμούς των πιστωτικών καρτών τους στους διακομιστές της Amazon, για αυτό και η ασφάλεια είναι μια σημαντική υπόθεση. Εκτός από την αυτόματη κρυπτογράφηση των αριθμών των πιστωτικών καρτών των χρηστών κατά τη διάρκεια της πληρωμής, η Amazon επιτρέπει στους χρήστες να επιλέγουν την κρυπτογράφηση κάθε κομματιού πληροφοριών που εισάγουν, όπως το όνομα, τη διεύθυνση και το φύλο τους.

Η

Amazon χρησιμοποιεί ένα Netscape Secure Commerce Server με πρωτόκολλο κρυπτογράφησης SSL (Secure Socket Layer) όπου αποθηκεύει όλους τους αριθμούς πιστωτικών καρτών σε ξεχωριστή βάση δεδομένων η οποία δεν είναι προσβάσιμη από το Διαδίκτυο, κλείνοντας το πιθανό σημείο εισόδου για τους χάκερ. Εκτός από τα συνήθη προβλήματα ασφάλειας που αφορούν τις ηλεκτρονικές αγορές με πιστωτική κάρτα, η Amazon πάσχει από το ίδιο πρόβλημα "ψαρέματος" (phishing problem) που μαστίζει το eBay και την PayPal, κατά το οποίο ένα πλαστό email ζητά τα στοιχεία του λογαριασμού του χρήστη Amazon.com.

2.2.2. Επεξήγηση λειτουργικότητας



Εικόνα 2.3

Το λογότυπο της εταιρείας με το χαρακτηριστικό βέλος που υποδηλώνει ότι το πολυκατάστημα περιέχει οτιδήποτε θελήσει κάποιος, από το Α μέχρι το Ζ.

Στα αριστερά παρουσιάζονται όλες τις κατηγορίες προϊόντων. Η Amazon προσφέρει μια πληθώρα προϊόντων τα οποία ταξινομούνται σε κατηγορίες που χωρίζονται σε αρκετές υποκατηγορίες η κάθε μια. Ο χρήστης μπορεί να επιλέξει κατευθείαν την κατηγορία ή ακόμα να εισάγει μια λέξη κλειδί στον απαιτούμενο χώρο ώστε το σύστημα να του βρει όλα τα πιθανά προϊόντα που υπάρχουν με βάση αυτή τη λέξη.

Στο πάνω μέρος δεξιά υπάρχουν οι λειτουργίες για το καλάθι αγορών όπου ο χρήστης τοποθετεί τα προϊόντα που προτίθεται να αγοράσει τη δεδομένη στιγμή, αλλά και η λειτουργία του *WishList* στην οποία εισάγονται τα προϊόντα που θα επιθυμούσε ο χρήστης να αποκτήσει κάποια στιγμή στο μέλλον.

Στην περίπτωση που ο χρήστης δεν έχει λογαριασμό, το σύστημα προτείνει προϊόντα με βάση την δημοτικότητα που έχουν από τους χρήστες του συστήματος η οποία επιλέγεται με βάση τις πωλήσεις που έχει κάνει το προϊόν. Αν ο χρήστης είναι ήδη εγγεγραμμένος στο σύστημα, θα πρέπει να εισάγει τους κωδικούς του ώστε να ενεργοποιηθεί η λειτουργία του φιλτραρίσματος με βάση το περιεχόμενο η οποία προτείνει προϊόντα στους χρήστες με βάση τα δικά τους χαρακτηριστικά. Τα προϊόντα επιλέγονται με βάση την λίστα επιθυμίας του πελάτη, όπου προτείνονται προϊόντα που έχουν σχέση με αυτά που βρίσκονται στην λίστα ή με βάση τα ενδιαφέροντά του τα οποία καταχωρεί ο χρήστης με τη δημιουργία του λογαριασμού του.

2.3. eBay



Εικόνα 2.4

Το eBay ιδρύθηκε από τον PierreOmidvar, προγραμματιστή στο επάγγελμα και αρχικά ξεκίνησε ως μέρος μιας μεγάλης προσωπικής ιστοσελίδας το 1995. Είναι ο μεγαλύτερος, ο πιο γνωστός και ο πιο πετυχημένος ιστότοπος ηλεκτρονικών δημοπρασιών, αλλά αποτελεί και διαδικτυακό πολυκατάστημα στο οποίο φιλοξενούνται πωλητές που θέλουν να διαθέσουν τα προϊόντα τους σε συγκεκριμένες τιμές.

Αρχικά, ξεκίνησε με την ονομασία *AuctionWeb*, οι πωλήσεις του όμως, μόλις στα τρία χρόνια λειτουργίας του, ήταν θεαματικές και έτσι η ονομασία άλλαξε και έγινε eBay. Η εταιρεία μεγάλωσε γρήγορα καθώς οι κατηγορίες των προϊόντων επεκτάθηκαν, πέρα από τα αποκλειστικά συλλεκτικά προϊόντα, σε οποιοδήποτε προϊόν μπορεί να πωληθεί. Παρέχει ένα μεγάλο αριθμό από κατηγορίες προϊόντων: από συλλογές και είδη διακόσμησης μέχρι ηλεκτρονικά προϊόντα, έπιπλα και οχήματα τα οποία πωλούνται στο eBay επί καθημερινής βάσεως.

Η καινοτομία που παρουσίασε από την αρχή της λειτουργίας του με τις διαδικτυακές δημοπρασίες υπήρξε καθοριστικός παράγοντας στην ανάπτυξή του. Μάλιστα ξέρει να τις χειρίζεται με έξυπνο τρόπο ώστε να κρατά τους χρήστες του ικανοποιημένους. Μπορεί να δημοπρατηθεί οτιδήποτε στον ιστότοπο, φτάνει όμως να μην είναι παράνομο και να μην παραβιάζει την πολιτική *Απαγόρευσης και Περιορισμού των προϊόντων* που παρέχει το eBay. Η εταιρεία έχει το δικαίωμα να αφαιρέσει τις δημοπρασίες που καταπατούν τους όρους της σύμβασης παροχής υπηρεσιών, αφού στο παρελθόν υπήρξε ένα πολύ χαρακτηριστικό περιστατικό όπου κάποιος χρήστης είχε φτάσει στο σημείο να δημοπρατήσει τον ένα του νεφρό.

Τεχνικές τέτοιου είδους ασφάλειας επιτρέπουν την ολοένα και μεγαλύτερη αύξηση αυτής της διαδικτυακής εφαρμογής. Μέσα από την δημοπρασία προϊόντων, πολλές μεγάλες πολυεθνικές εταιρείες βρήκαν τον τρόπο να προωθήσουν τα προϊόντα τους και παράλληλα να διαφημιστούν. Ένα παράδειγμα, είναι η IBM η οποία πωλεί τα καινούργια της προϊόντα και προτείνει υπηρεσίες στους χρήστες του eBay, χρησιμοποιώντας ανταγωνιστικές δημοπρασίες.

Μέσα από την διεπαφή του διαδικτυακού πολυκαταστήματος, γίνεται ξεκάθαρο ποια προϊόντα διατίθενται για δημοπρασία και ποια διατίθενται απευθείας για πώληση. Τα προϊόντα που διατίθενται για πώληση συνοδεύονται από την επιγραφή *BuyItNow*. Η επιγραφή αυτή υποδηλώνει ότι οι αγοραστές έχουν κερδίσει αμέσως την δημοπρασία στην τιμή που αναγράφεται χωρίς να χρειάζεται να ποντάρουν σε οποιοδήποτε ποσό, ουσιαστικά δεν υπάρχει καν δημοπρασία. Σε όσα προϊόντα δεν υπάρχει η επιγραφή αυτή, αναγράφεται ο αριθμός των χρηστών που έχουν ποντάρει μέχρι στιγμής (π.χ 15 bids).

Εκτός από την επιγραφή *BuyItNow* ή *# bids*, δίπλα από κάθε προϊόν προς πώληση, αναγράφεται το όνομα του πωλητή, η διεύθυνση ηλεκτρονικού ταχυδρομείου του και ένας αριθμός που αναφέρεται στο άθροισμα της φήμης του. Το σύστημα φήμης του eBay παρέχει πολύ σημαντικό βαθμό εμπιστοσύνης, ασφάλειας αλλά και σταθερότητας, στοιχεία πολύ σημαντικά για όλους τους χρήστες, καθώς οι συνδιαλλαγές μέσω διαδικτύου καθιστούν ένα αβέβαιο περιβάλλον ρίσκου. Το σύστημα αυτό βοηθά τους χρήστες να εκτιμούν με υποκειμενικά κριτήρια την ποιότητα των προϊόντων και να κρίνουν

για την αξιοπιστία τους μέσω των αξιολογήσεων που γίνονται από άλλους χρήστες. Κάθε εγγεγραμμένο μέλος του eBay μπορεί να εισάγει οποιασδήποτε μορφής σχόλια επιθυμεί – αρνητικά, θετικά ή ουδέτερα – ενώ δεν έχει τη δυνατότητα να αλλάξει ή να σβήσει οποιοδήποτε σχόλιο.

Με αυτόν τον τρόπο ο κάθε πωλητής, ο οποίος θέτει ένα δικό του προϊόν προς πώληση στο eBay συγκεντρώνει ένα άθροισμα φήμης. Το άθροισμα αυτό διαμορφώνεται αναλόγως σε σχέση με τα σχόλια του κάθε χρήστη. Για θετικό σχόλιο προστίθεται ένας βαθμός στο άθροισμα φήμης, για αρνητικό σχόλιο αφαιρείται ένας και για ουδέτερο σχόλιο το άθροισμα παραμένει ως έχει. Επομένως, οι χρήστες χρησιμοποιούν σε μεγάλο βαθμό αυτό το κριτήριο ώστε να επιλέξουν τον προμηθευτή από τον οποίο θα αγοράσουν το προϊόν που επιθυμούν.

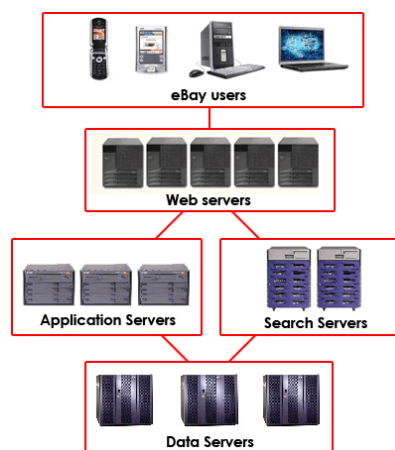
Το eBay ακολουθεί τις ίδιες προσεγγίσεις όπως και η Amazon. Δηλαδή, ακολουθεί την προσέγγιση *φιλτραρίσματος με βάση το περιεχόμενο* στην περίπτωση που ο χρήστης έχει λογαριασμό στο σύστημα, ενώ ακολουθεί την *προσέγγιση συνεργατικού φιλτραρίσματος* όταν ο χρήστης δεν έχει κάνει εγγραφή. Στην πρώτη περίπτωση, το σύστημα προτείνει προϊόντα με βάση τα προϊόντα που είχε αναζητήσει ο χρήστης κατά την τελευταία του είσοδο στο σύστημα. Παρέχει επιπλέον ένα ιστορικό με τις επισκέψεις του χρήστη, ώστε να τον βοηθήσει όσο περισσότερο μπορεί στην αναζήτησή του.

Στην δεύτερη περίπτωση, παρατηρείται ότι αυτή η εφαρμογή απουσιάζει από την διεπιφάνεια του χρήστη και τη θέση της παίρνουν εξ ολοκλήρου άλλα προϊόντα τα οποία διαφέρουν από μέρα σε μέρα. Τα προϊόντα αυτά μπορεί να είναι προϊόντα με τις περισσότερες πωλήσεις ή με τις πιο δημοφιλείς δημοπρασίες ή μπορεί να είναι προϊόντα τα οποία θέλουν να προωθήσουν οι πωλητές για σκοπούς διαφήμισης.

2.3.1. Αρχιτεκτονική συστήματος

Μια σειρά διαταραχών στις υπηρεσίες του το 1999 προκάλεσαν σοβαρά προβλήματα για την επιχείρηση eBay. Η περιοδική υπερφόρτωση των εξυπηρετητών του συστήματος εμπόδιζε τους χρήστες να ελέγξουν δημοπρασίες, να τοποθετήσουν προσφορές ή να ολοκληρώσουν συναλλαγές. Οι αγοραστές, οι πωλητές και το eBay ήταν πολύ δυσαρεστημένοι και μια πλήρη αναδιάρθρωση της τεχνολογικής αρχιτεκτονικής eBay ακολούθησε.

Το 1999, το eBay αποτελείτο από ένα τεράστιο διακομιστή βάση δεδομένων και μερικά ξεχωριστά συστήματα που εκτελούσαν τη λειτουργία αναζήτησης. Το 2005, το eBay αποτελείτο από περίπου 200 διακομιστές βάσεων δεδομένων και 20 διακομιστές αναζήτησης.



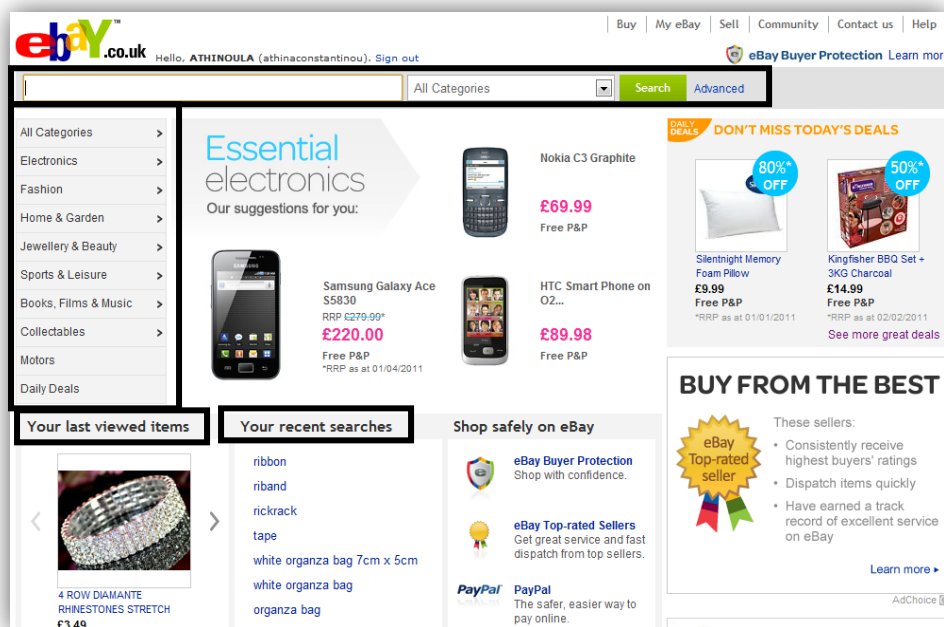
Εικόνα 2.5

Η αρχιτεκτονική του είναι ένα είδος υπολογιστικού πλέγματος που επιτρέπει τόσο τη διόρθωση σφαλμάτων και την ανάπτυξη. Με εξαίρεση τη λειτουργία αναζήτησης, τα πάντα για το eBay μπορούν να διακινούνται αποτελεσματικά σε περίπου 50 διακομιστές - διακομιστές ιστού, διακομιστές εφαρμογών και συστήματα αποθήκευσης δεδομένων. Κάθε διακομιστής έχει μεταξύ έξι και 12 μικροεπεξεργαστών. Αυτοί 50 περίπου διακομιστές λειτουργούν ξεχωριστά, αλλά μπορούν να επικοινωνούν μεταξύ τους και έτσι όλοι γνωρίζουν εάν υπάρχει ένα πρόβλημα κάπου. Το eBay μπορεί με ευκολία να προσθέσει διακομιστές στο δίκτυο του ανάλογα με τις ανάγκες του.

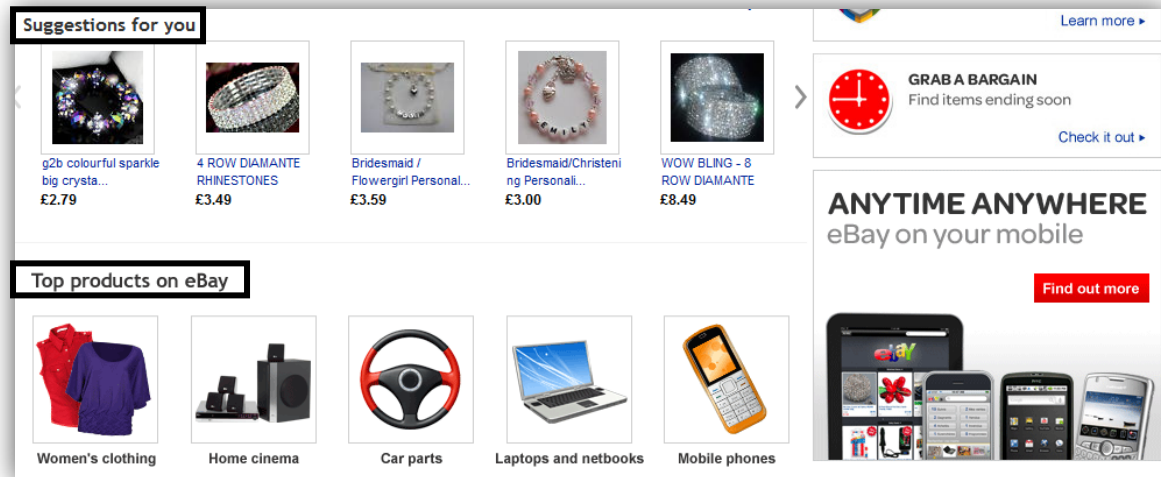
Ενώ η πλειοψηφία του ιστοχώρου μπορεί να τρέξει σε 50 διακομιστές, το eBay διαθέτει τέσσερις φορές περισσότερους. Οι 200 διακομιστές στεγάζονται ανά 50 σε τέσσερις τοποθεσίες σε ολόκληρη την Αμερική. Αν κάποια περιοχή παρουσιάσει πρόβλημα, οι τρεις άλλες αναλαμβάνουν να διεκπεραιώσουν τις λειτουργίες του συστήματος.

Οι τρεις βάσεις δεδομένων που διατηρεί είναι χωρισμένες σύμφωνα με τις λειτουργικές περιοχές του διαδικτυακού καταστήματος. Η πρώτη βάση δεδομένων αφορά τα προϊόντα που περιέχει το κατάστημα. Περιλαμβάνει όλα τα χαρακτηριστικά τους, καταλόγους με τα προϊόντα ταξινομημένα σε κατηγορίες καθώς και κανόνες από τους οποίους διέπονται τα προϊόντα όπως για παράδειγμα ο τρόπος πληρωμής τους. Η δεύτερη βάση δεδομένων αφορά τους χρήστες και περιλαμβάνει στοιχεία αναφορικά με την ταυτότητα του κάθε χρήστη. Η τρίτη βάση δεδομένων είναι υπεύθυνη να καταγράφει τις συναλλαγές που πραγματοποιούνται. Έτσι παρέχεται ανά πάσα στιγμή μια ξεκάθαρη εικόνα για τις αγοραπωλησίες που διεξάγονται.

2.3.2. Επεξήγηση λειτουργικότητας



Εικόνα 2.6



Εικόνα 2.7

Οι χρήστες μπορούν να κάνουν πιο συγκεκριμένη και περιορισμένη την αναζήτησή τους. Μπορούν να επιλέξουν είτε μια συγκεκριμένη κατηγορία είτε να πληκτρολογήσουν κατευθείαν κάποια λέξη κλειδί για αυτό που επιθυμούν να αναζητήσουν. Το σύστημα παρέχει τη δυνατότητα ομαδοποίησης των προϊόντων βάση χαρακτηριστικών, όπως το χρώμα ή το υλικό κατασκευής. Με την επιλογή ενός προϊόντος μπορεί να διαπιστωθούν τα επιπλέον χαρακτηριστικά που διαθέτει, την κατάστασή του – αν είναι μεταχειρισμένο ή όχι – αλλά και τις χώρες στις οποίες μπορεί να αποσταλεί.

Το σύστημα ενημερώνει τον χρήστη για την διαθέσιμη ποσότητα κάθε προϊόντος και κατά πόσο το προϊόν βρίσκεται σε δημοπρασία ή όχι. Θέτει επίσης, ένα χρονικό όριο κατά το οποίο το προϊόν είναι διαθέσιμο – σε περίπτωση που δεν βρίσκεται σε δημοπρασία – ή ένα χρονικό όριο κατά το οποίο φτάνει στο τέλος της η πώλησή του.

2.4. ASOS



Εικόνα 2.8

Το ASOS ιδρύθηκε το 2000 από τον NickRobertson έναν Βρετανό επιχειρηματία. Αποτελεί το μεγαλύτερο διαδικτυακό κατάστημα του Ηνωμένου Βασιλείου αποκλειστικά για προϊόντα μόδας και ομορφιάς. Απευθύνεται τόσο σε άνδρες όσο και σε γυναίκες και διαθέτει μια πληθώρα από όλα τα είδη μόδας.

Η μέχρι τώρα ανάπτυξη του διαδικτυακού καταστήματος είναι αξιόλογη, καθώς ενώ αρχικά άρχισε την λειτουργία του με μόλις τρεις υπαλλήλους, 11 χρόνια μετά όχι μόνο αύξησε τους υπαλλήλους του στους 1000, ανεβάζοντας τα κέρδη του στα ύψη, αλλά έγινε και το μεγαλύτερο ανεξάρτητο διαδικτυακό κατάστημα μόδας και ομορφιάς στη χώρα του. Οι υπηρεσίες που παρέχει και ο τρόπος που αντιμετωπίζει τους πελάτες του έπαιξαν καταλυτικό ρόλο σε αυτή του την ανάπτυξη.

Τα χαρακτηριστικά του καταστήματος είναι οι μοναδικές λειτουργίες που παρέχει όσον αφορά τον κόσμο της μόδας καθώς περιλαμβάνει εκτός από τις γνωστές συλλογές ενδυμάτων, υποδημάτων και αξεσουάρ, βίντεο με μοντέλα στην πασαρέλα ώστε οι πελάτες να έχουν μια πιο ολοκληρωμένη εικόνα του προϊόντος. Επιπλέον, διαθέτει ένα μπλοκ μόδας στο οποίο βρίσκονται, συνεχώς ανανεωμένα, άρθρα συσχετιζόμενα με τον χώρο της μόδας από τα οποία μπορούν να πάρουν χρήσιμες πληροφορίες.

Το κατάστημα απευθύνεται σε όλες τις τάξεις ανθρώπων: από τους πιο ευκατάστατους στους λιγότερο ευκατάστατους, καθώς διαθέτει και πολύ γνωστές φίρμες ρούχων όπως Diesel και TommyHilfiger ενώ με το τέλος μιας σεζόν παρέχει έκπτωση μέχρι και 75% στα προϊόντα της προηγούμενης σεζόν. Χωρίζει όλα τα προϊόντα σε κατηγορίες και τα διαχωρίζει ανάμεσα σε άντρες και γυναίκες. Διαθέτει μια πολύ μεγάλη βάση δεδομένων όπου τα προϊόντα χωρίζονται με βάση τις κατηγορίες που ανήκουν, το χρώμα τους, το υλικό κατασκευής τους και την τιμή τους.

Καθώς η ανάπτυξη του ASOS είναι μεγάλη, το κατάστημα επεκτάθηκε και εκτός των συνόρων του Ηνωμένου Βασιλείου και συγκεκριμένα στην Γερμανία, στην Γαλλία και στις Ηνωμένες Πολιτείες για καλύτερη εξυπηρέτηση των πελατών του. Μάλιστα για μεγαλύτερη ευκολία παρέχει εφαρμογή αυτόματης τροποποίησης του συναλλάγματος ανάλογα με την χώρα την οποία βρίσκεται ο κάθε πελάτης.

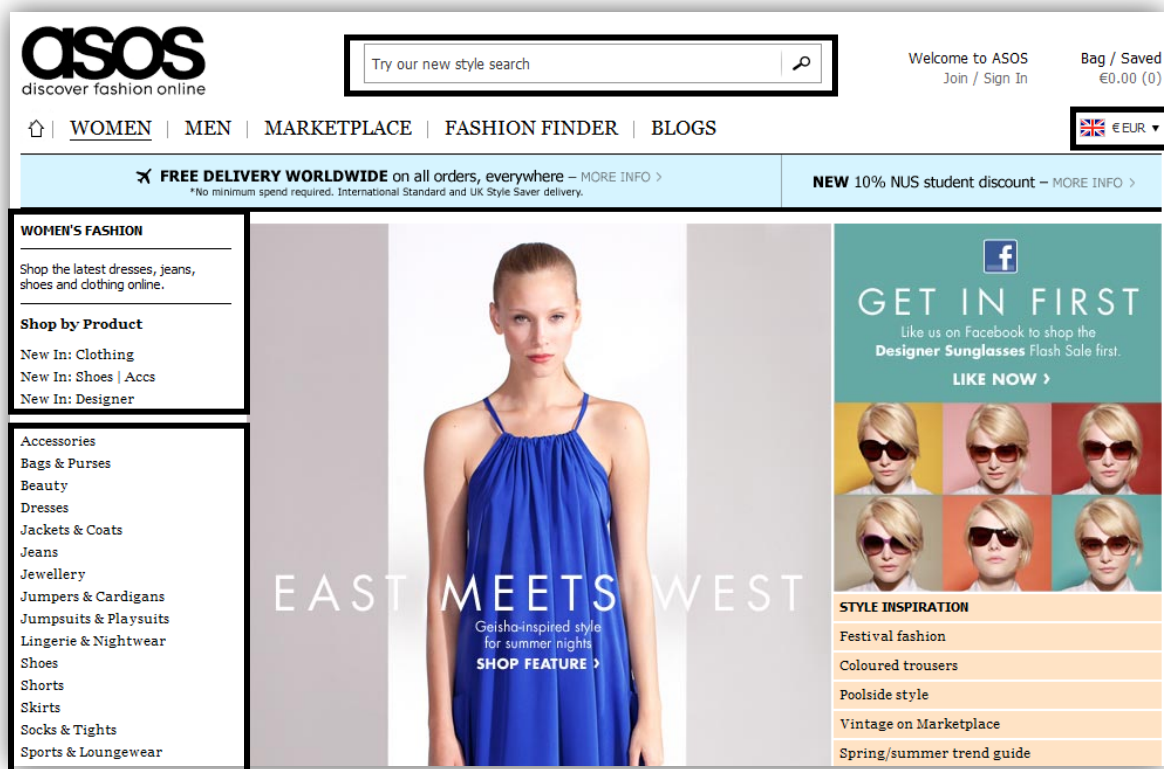
Το ASOS δεν διαχωρίζει την διεπιφάνειά του ανάλογα με το αν ο χρήστης είναι εγγεγραμμένος στο σύστημα ή όχι. Ο μόνος περιορισμός για τον οποίο πρέπει κάποιος να έχει λογαριασμό είναι στην περίπτωση που θέλει να αγοράσει οποιοδήποτε προϊόν. Τα προϊόντα που προτείνονται είναι γενικά και όχι ειδικά. Ουσιαστικά, προβάλλονται όλες οι εφαρμογές και οι κατηγορίες που διαθέτει το σύστημα, χωρίς να επηρεάζονται από την κατηγορία με τις περισσότερες πωλήσεις ή τα πιο δημοφιλή προϊόντα. Συγκεκριμένα, προβάλλονται εφαρμογές για τις δύο κατηγορίες αντρών και γυναικών, λεπτομέρειες σχετικά με τα διαδικτυακά περιοδικά που εκδίδει το ASOS, αλλά και πρόσβαση στα ιστολόγια του συστήματος για επισημάνσεις, παράπονα κλπ.

Για το διαδικτυακό κατάστημα του ASOS, χρησιμοποιείται η προσέγγιση *φιλτραρίσματος με βάση το περιεχόμενο*. Ο χρήστης μπορεί να αναζητήσει κάποιο προϊόν είτε κατευθείαν επιλέγοντας την κατηγορία στην οποία ανήκει το προϊόν είτε πληκτρολογώντας κάποια λέξη κλειδί. Επιπλέον, με την επιλογή οποιουδήποτε προϊόντος εμφανίζονται είτε προτεινόμενα προϊόντα που ολοκληρώνουν την εμφάνιση σε σχέση με το επιλεγόμενο προϊόν, είτε προτεινόμενα προϊόντα που ταιριάζουν με το επιλεγμένο προϊόν, είτε συνδυασμός και των δύο.

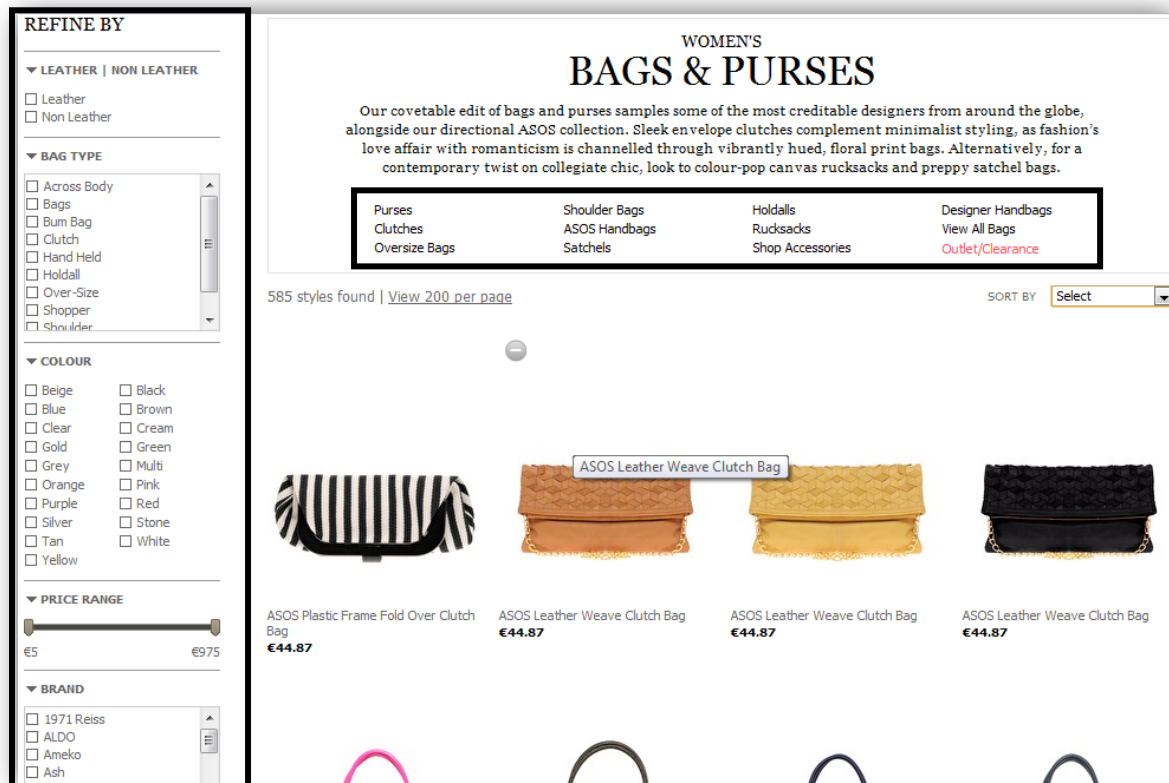
Η λογική που χρησιμοποιεί το σύστημα για να εμφανίσει προτεινόμενα προϊόντα που ολοκληρώνουν την εμφάνιση είναι ότι επιλέγει προϊόντα διαφορετικού είδους αλλά ίδιας μάρκας, στις περιπτώσεις όπου ο χρήστης έχει επιλέξει μια συγκεκριμένη μάρκα προϊόντος. Σε αντίθετη περίπτωση, αν δηλαδή ο χρήστης ψάχνει γενικά, τότε τα προϊόντα αυτά είναι και πάλι διαφορετικού είδους και διαφορετικής μάρκας από το επιλεγμένο προϊόν και η επιλογή γίνεται σύμφωνα με το χρώμα, το στυλ και την τιμή του επιλεγμένου προϊόντος που υπάρχουν ήδη κατηγοριοποιημένα μέσα στη βάση δεδομένων του συστήματος.

Τα προτεινόμενα προϊόντα που ταιριάζουν με το επιλεγμένο προϊόν είναι του ίδιου τύπου, αλλά διαφορετικής μάρκας και αυτό ισχύει τόσο στην περίπτωση που επιλέγεται κάποιος συγκεκριμένος τύπος μάρκας όσο και στην περίπτωση που αναζητείται οποιαδήποτε μάρκα.

2.4.1. Επεξήγηση λειτουργικότητας



Εικόνα 2.9



Εικόνα 2.10

Στις πιο πάνω εικόνες, φαίνεται ότι υπάρχει ένας μεγάλος αριθμός από επιλογές στην κάθε κατηγορία ώστε οι κατηγορίες να γίνουν πιο συγκεκριμένες και να είναι πιο εύκολο για τον χρήστη να βρει αυτό που επιθυμεί. Έχει επιπλέον και την επιλογή της εισαγωγής μιας λέξης-κλειδί για το προϊόν που ψάχνει. Η κάθε κατηγορία έχει τις δικές τις υποκατηγορίες οι οποίες διευκολύνουν τον χρήστη στην αναζήτησή του.

Οι υποκατηγορίες αυτές χωρίζονται με βάση το υλικό κατασκευής του προϊόντος (π.χ. στην περίπτωση τσαντών το υλικό μπορεί να είναι από δέρμα ή όχι), τον τύπο του προϊόντος, το χρώμα του, την τιμή του την οποία μπορεί να ρυθμίσει ο πελάτης ανάλογα με την οικονομική του δυνατότητα και τέλος τη μάρκα του. Ανάλογα με αυτές τις επιλογές γίνεται η αναζήτηση των προϊόντων αλλά και η προβολή των προτεινόμενων προϊόντων σύμφωνα με τα κριτήρια που αναφέρθηκαν πιο πάνω και με τα οποία έχει ρυθμιστεί η βάση δεδομένων να προβάλλει.

Για διευκόλυνση των πελατών του, το ASOS παρέχει αυτόματη μετατροπή του συναλλάγματος ανάλογα με τη χώρα που βρίσκεται ο χρήστης και το συνάλλαγμα που επιθυμεί να κάνει την αγορά του. Με την επιλογή οποιουδήποτε συναλλάγματος, οι τιμές αλλάζουν αυτόματα σε όλες τις σελίδες του συστήματος. Τέλος το κατάστημα θέτει σε προσφορές, ανά τακτά χρονικά διαστήματα, διάφορα προϊόντα τα οποία για να γίνουν εμφανή από τους χρήστες αναγράφονται με κόκκινα γράμματα.

Κεφάλαιο 3. Ανάλυση προδιαγραφών

4.1.	ΕΙΣΑΓΩΓΗ.....	21
4.1.1.	Σκοπός	
4.1.2.	Το σύστημα και οι αποδέκτες του	
4.1.3.	Σύνοψη	
4.2.	ΜΕΘΟΔΟΛΟΓΙΑ GANE & SARSEN.....	22
4.2.1.	Διαγράμματα Ροής Δεδομένων	
4.2.2.	Μηχανοργάνωση	
4.2.3.	Εκλέπτυνση ροών δεδομένων	
4.2.4.	Εκλέπτυνση λογικής διαδικασιών	
4.2.5.	Εκλέπτυνση αποθήκευσης δεδομένων	
4.2.6.	Ορισμός φυσικών πόρων	
4.2.7.	Ορισμός προδιαγραφών για είσοδο/έξοδο δεδομένων	
4.2.8.	Εκτίμηση μεγέθους συστήματος	
4.2.9.	Απαιτήσεις σε hardware	
4.3.	ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΑΝΑΛΥΣΗ	58
4.3.1.	Διαγράμματα Χρήσης	
4.3.2.	Διάγραμμα Κλάσεων	
4.4.	Διάγραμμα Οντοτήτων - Συσχετίσεων	63

3.1. ΕΙΣΑΓΩΓΗ

3.1.1. Σκοπός

Ο σκοπός του κεφαλαίου αυτού είναι η εύρεση, η ανάλυση και η τεκμηρίωση όλων των παραγόντων που επηρεάζουν και αλληλεπιδρούν με το σύστημα που πρόκειται να αναπτυχθεί για την υποστήριξη των βασικών λειτουργιών που χρειάζεται κάθε μικροεπιχείρηση στο χώρο των πωλήσεων.

Στο κεφάλαιο αυτό καταγράφονται οι προδιαγραφές, οι περιορισμοί στην σχεδίαση, οι εισοδοί και έξοδοι του συστήματος. Επίσης γίνεται καταγραφή οποιουδήποτε άλλου συστήματος ή συσκευής που πρόκειται να αλληλεπιδρά με το σύστημα αυτό.

Οι πληροφορίες του κεφαλαίου αυτού θα χρησιμοποιηθούν στην φάση της υλοποίησης του συστήματος, αλλά και στις επόμενες φάσεις, για την σωστή σχεδίαση του συστήματος.

3.1.2. Το σύστημα και οι αποδέκτες του

Το σύστημα που αναμένεται να υλοποιηθεί θα είναι το σύστημα αποθήκευσης και διαχείρισης των δεδομένων των πελατών της επιχείρησης, όπως και το σύστημα διαχείρισης των πωλήσεων. Επίσης, θα διαχειρίζεται τα αποθέματα της επιχείρησης. Οι πιο πάνω λειτουργίες θα πρέπει να υποστηρίζονται από μια βάση δεδομένων και μια διεπιφάνεια χρήστη.

Το σύστημα αποθήκευσης και διαχείρισης δεδομένων των πελατών θα είναι υπεύθυνο για την εισαγωγή των στοιχείων των πελατών, την ενημέρωση τους σε τυχόν αλλαγές και την παρουσίαση τους στο χρήστη του συστήματος. Το σύστημα διαχείρισης των πωλήσεων θα είναι υπεύθυνο για την εισαγωγή παραγγελίας, την ενημέρωση και εμφάνισή της καθώς επίσης και την ετοιμασία τιμολογίου. Το σύστημα διαχείρισης αποθεμάτων είναι υπεύθυνο για την ενημέρωση των ποσοτήτων των πρώτων υλών που χρησιμοποιεί η επιχείρηση.

Η βάση που θα υλοποιηθεί θα περιλαμβάνει τα δεδομένα των πελατών, προϊόντων, πρώτων υλών, καθώς και τα δεδομένα των αποδείξεων, τιμολογίων και καταστάσεων πληρωμών του κάθε πελάτη και της κάθε παραγγελίας. Θα είναι υπεύθυνη για την διεκπεραίωση των διεργασιών που αφορούν τα πιο πάνω συστήματα διαχείρισης.

Τα πλεονεκτήματα που αναμένονται από τη δημιουργία της βάσης είναι:

- Ευκολία στη διαχείριση δεδομένων
- Ταχύτητα στη πρόσβαση των δεδομένων
- Έλεγχο στις παραγγελίες και στις οφειλές των πελατών
- Έλεγχο στα αποθέματα της επιχείρησης
- Οργάνωση του όλου συστήματος της επιχείρησης

Η διεπιφάνεια χρήστη θα περιλαμβάνει φόρμες εισαγωγής, ενημέρωσης και παρουσίασης δεδομένων, καθώς επίσης θα έχει τη δυνατότητα δημιουργίας επιστολών (παραγγελίες, τιμολόγια, αποδείξεις, καταλόγους δεδομένων της βάσης).

Τα πλεονεκτήματα που αναμένονται από τη διεπιφάνεια χρήστη στο σύστημα είναι:

- Ευκολία, ταχύτητα στη χρήση
- Αύξηση παραγωγικότητας του χρήστη

Τα πλεονεκτήματα προκύπτουν από το γεγονός ότι ο χρήστης δεν θα είναι αναγκασμένος να θυμάται κωδικούς για τους πελάτες ή τα προϊόντα του καθώς και για τα άλλα δεδομένα της βάσης, μιας και αυτά θα του παρουσιάζονται στο γραφικό περιβάλλον και θα πρέπει απλά να επιλέγει.

3.1.3. Σύνοψη

Στη συνέχεια του κεφαλαίου θα παρουσιαστούν διάφορα διαγράμματα, τα οποία απεικονίζουν τυποποιημένα τις απαιτήσεις του συστήματος και θα βοηθήσουν στην πλήρη κατανόηση των λειτουργιών.

Ακολουθεί δομημένη ανάλυση των λειτουργιών του συστήματος με την χρήση των Διαγραμμάτων Ροής Δεδομένων (ΔΡΔ), ακολουθώντας την μεθοδολογία Gane&Sarsen. Η μεθοδολογία αυτή αποτελείται από 9 βήματα με σταδιακή τελειοποίηση σε κάθε βήμα. Ακολούθως, θα δημιουργηθεί το Διάγραμμα Σχέσεων – Οντοτήτων (Entity – Relationship Diagram).

3.2. ΜΕΘΟΔΟΛΟΓΙΑ GANE & SARSEN

Η μεθοδολογία Gane&Sarsen είναι μια από τις τεχνικές που χρησιμοποιείται ευρέως στη δομημένη ανάλυση συστημάτων και κυρίως στον αντικειμενοστραφή σχεδιασμό, με στόχο τον καθορισμό των προδιαγραφών με την χρήση γραφικών (διαγραμμάτων). Η μεθοδολογία αυτή αποτελείται από 9 βήματα:

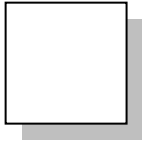
1. Δημιουργία των διαγραμμάτων ροής δεδομένων που θα αντιπροσωπεύουν την λογική ροή των δεδομένων μέσα στο σύστημα.
2. Καταγραφή των αποφάσεων σχετικά με το ποια τμήματα θα μηχανογραφηθούν και ποια όχι, την αιτιολόγηση αυτών των αποφάσεων καθώς και την περιγραφή του πώς θα μηχανογραφηθεί κάθε τμήμα.
3. Εκλέπτυνση των ροών δεδομένων, καταγράφοντας αναλυτικά τα στοιχεία κάθε ροής δεδομένων και αναλύοντας μέχρι να παρουσιάζονται όλα τα στοιχεία με πλήρη λεπτομέρεια.
4. Ανάλυση των διαδικασιών που υπάρχουν στα διαγράμματα ροής δεδομένων και συγκεκριμένα του προσδιορισμού λογικής της κάθε διαδικασίας.
5. Προσδιορισμός της αποθήκευσης δεδομένων όπου ορίζονται τα ακριβή περιεχόμενα κάθε αποθήκευσης και η μορφή τους καθώς επίσης και το πού απαιτείται άμεση πρόσβαση.
6. Καθορίζονται οι φυσικοί πόροι που θα χρησιμοποιηθούν για την εφαρμογή και αναλύονται θέματα όπως το όνομα του κάθε αρχείου, η οργάνωσή του καθώς και τα μέσα αποθήκευσης.
7. Ορισμός των προδιαγραφών για εισόδους / εξόδους.
8. Εκτίμηση μεγέθους.
9. Καθορισμός των απαιτήσεων σε υλικό (hardware).

3.2.1. Διαγράμματα Ροής Δεδομένων

Τα διαγράμματα ροής δεδομένων εκφράζουν τη λογική ροή των δεδομένων του συστήματος, δηλαδή την αλληλεπίδραση μεταξύ εξωτερικών πρακτόρων, διαδικασιών και αποθηκευτικών μονάδων. Για την αναπαράσταση ενός μελλοντικού συστήματος απαιτείται η διάσπαση των λειτουργιών του σε διάφορα επίπεδα αναλόγως της πολυπλοκότητάς του.

Δομικά στοιχεία ενός ΔΡΔ

Εξωτερικός πράκτορας



Αντιπροσωπεύει σημεία αλληλεπίδρασης συστήματος με το εξωτερικό περιβάλλον. Επίσης, καθορίζουν την εμβέλειά του. Οι εξωτερικοί πράκτορες αναπαριστούν πρόσωπα, μονάδες του οργανισμού, κάποιο άλλο σύστημα, οργανισμούς που αλληλεπιδρούν με το σύστημα για την ανταλλαγή δεδομένων.

Διαδικασία



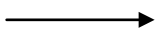
Αντιπροσωπεύει διαδικασία που εκτελείται ή που προκαλείται από την εισροή δεδομένων στο σύστημα.

Αποθηκευτική μονάδα



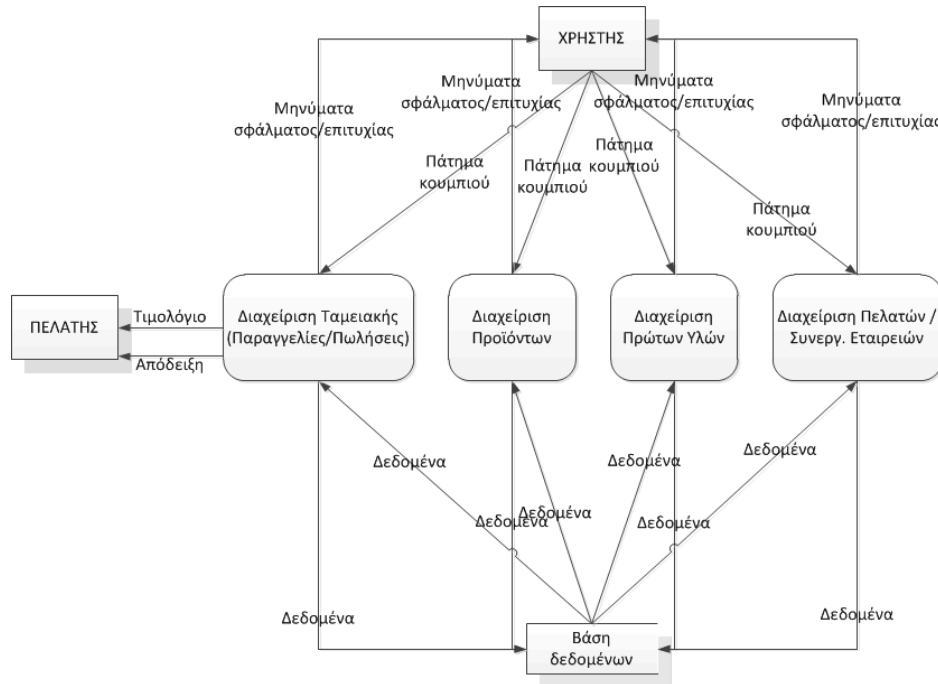
Αντιπροσωπεύει αποθηκευτική μονάδα όπου τα δεδομένα του συστήματος. Είναι ουσιαστικά οι χώροι όπου φυλάγονται τα δεδομένα. Συνήθως υλοποιούνται ως αρχεία ή βάσεις δεδομένων.

Ροές δεδομένων



Αναπαριστούν εισόδους και εξόδους προς και από μια διαδικασία. Προέρχονται από ή καταλήγουν σε εξωτερικούς πράκτορες, αποθηκευτικές μονάδες ή άλλες διαδικασίες.

3.2.1.1. Επίπεδο 0

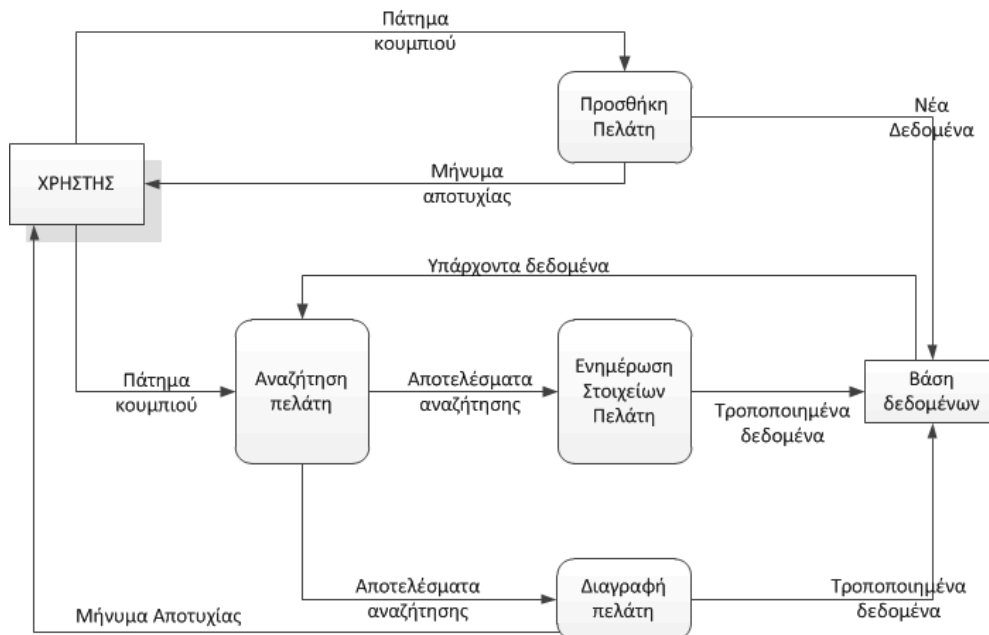


Διάγραμμα 3.1

Στο επίπεδο 0, ο χρήστης αλληλεπιδρά με το σύστημα, κάνοντας χρήση ενός συνόλου λειτουργιών που προσφέρονται μέσω του λογισμικού που πρόκειται να αναπτυχθεί και επίσης με τη βάση δεδομένων όπου φυλάσσονται τα όλα δεδομένα της επιχείρησης.

3.2.1.2. Διαχείριση πελατών / συνεργαζόμενων εταιρειών / προμηθευτών

Λόγω του ότι οι λειτουργίες των πελατών, των συνεργαζόμενων εταιρειών και των προμηθευτών είναι οι ίδιες, παρατίθενται μόνο του πελάτη.

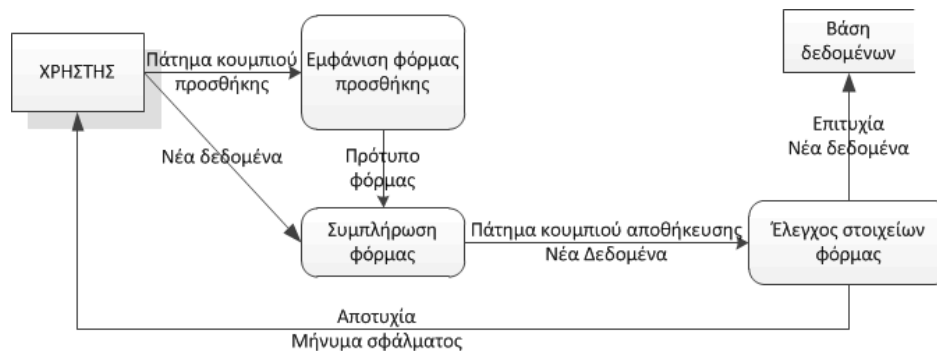


Διάγραμμα 3.2

Κατά τη λειτουργία διαχείρισης πελατών, προσφέρονται στο χρήστη οι εξής υπολειτουργίες:

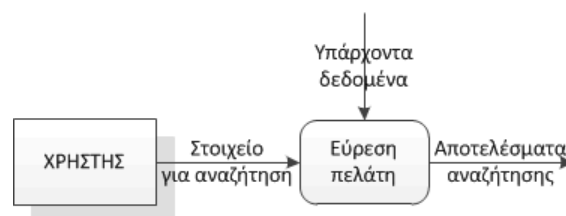
- **Προσθήκη πελάτη:** Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου πελάτη αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- **Αναζήτηση πελάτη:** Ο χρήστης μπορεί να αναζητήσει στοιχεία κάποιου πελάτη. Του επιστρέφεται μια λίστα με τα αποτελέσματα αναζήτησης.
- **Ενημέρωση στοιχείων πελάτη:** Ο χρήστης μπορεί να τροποποιήσει τα στοιχεία ενός πελάτη, αφού πρώτα τον αναζητήσει. Η βάση δεδομένων ενημερώνεται με τα τροποποιημένα στοιχεία του πελάτη.
- **Διαγραφή πελάτη:** Ο χρήστης μπορεί να διαγράψει ένα πελάτη, αφού πρώτα τον αναζητήσει. Η βάση δεδομένων ενημερώνεται για τη διαγραφή του πελάτη, αν υπάρχει, διαφορετικά επιστρέφει μήνυμα αποτυχίας.

3.2.1.2.1. Προσθήκη πελάτη



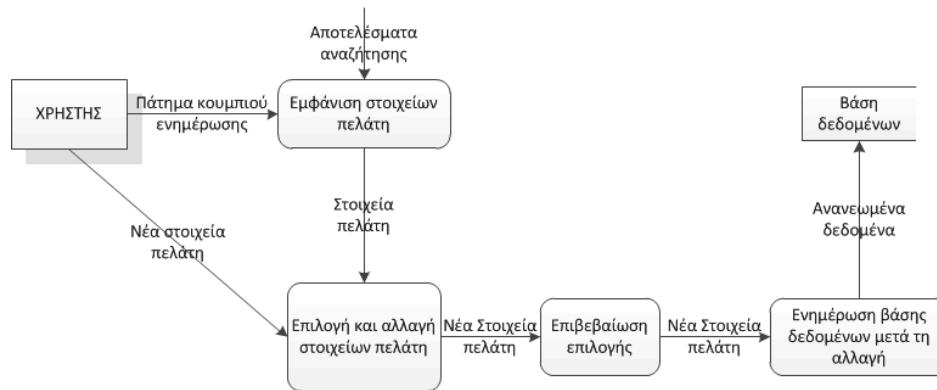
Ο χρήστης επιλέγει με το πάτημα ενός κουμπιού προσθήκης να εισάγει νέα δεδομένα, ένα νέο πελάτη, στη βάση δεδομένων. Αφού εμφανιστεί η φόρμα προσθήκης, τα στοιχεία συμπληρώνονται από το χρήστη. Όταν ο χρήστης επιλέξει "Αποθήκευση δεδομένων", τα νέα δεδομένα ελέγχονται για την ορθότητά τους. Αν υπάρχει κάποιο λανθασμένο δεδομένο, ο χρήστης ενημερώνεται για το σφάλμα. Διαφορετικά, αν όλα τα στοιχεία είναι σωστά, τότε αποθηκεύονται στη βάση δεδομένων και ο χρήστης ενημερώνεται για τη νέα καταχώρηση.

3.2.1.2.2. Αναζήτηση πελάτη



Ο χρήστης εισάγει ένα στοιχείο αναζήτησης και το σύστημα του επιστρέφει μία λίστα με τους υπάρχοντες πελάτες που αναφέρονται με το στοιχείο αυτό.

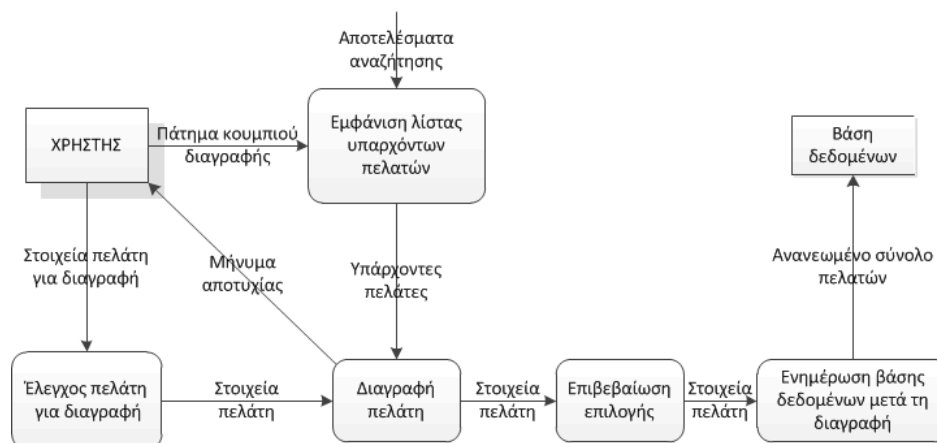
3.2.1.2.3. Ενημέρωση στοιχείων πελάτη



Διάγραμμα 3.5

Ο χρήστης επιλέγει να ενημερώσει τα στοιχεία ενός πελάτη πατώντας το αντίστοιχο κουμπί ενημέρωσης. Εμφανίζεται τότε μια λίστα με τους υπάρχοντες πελάτες, από την οποία ο χρήστης επιλέγει τον πελάτη του οποίου θέλει να ενημερώσει τα στοιχεία. Ο χρήστης εισάγει τα νέα δεδομένα και επιλέγει να αποθηκευτούν οι αλλαγές. Αν ο χρήστης επιβεβαιώσει την επιλογή του, τότε η βάση δεδομένων ενημερώνεται.

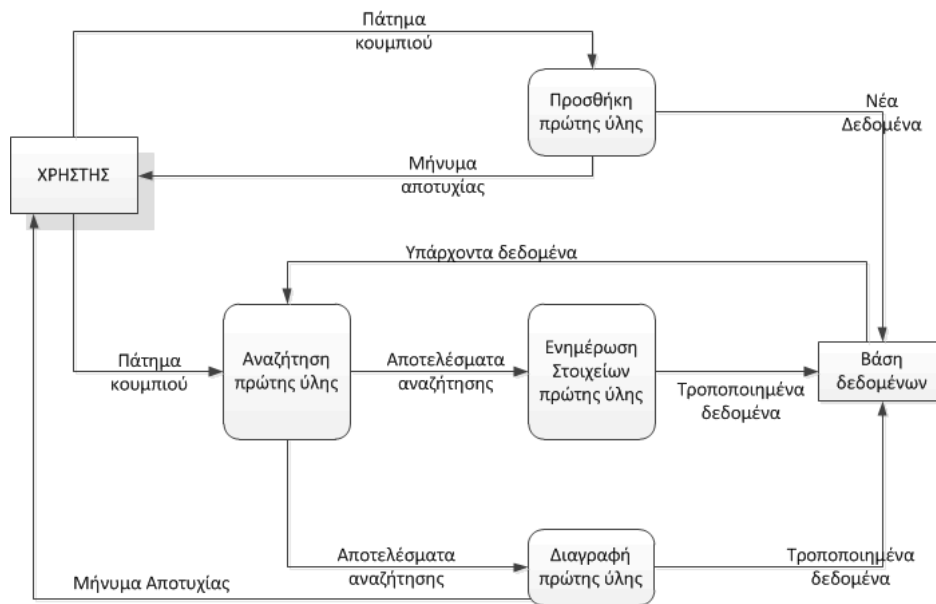
3.2.1.2.4. Διαγραφή πελάτη



Διάγραμμα 3.6

Ο χρήστης επιλέγει να διαγράψει ένα πελάτη πατώντας το αντίστοιχο κουμπί. Αφού εμφανιστεί μία λίστα με τους υπάρχοντες πελάτες, ο χρήστης δίνει ένα στοιχείο του πελάτη που θέλει να διαγράψει. Πριν γίνει διαγραφή, γίνεται έλεγχος κατά πόσον το στοιχείο που δόθηκε αντιστοιχεί σε κάποιο υπάρχοντα πελάτη. Αν δεν αντιστοιχεί, τότε στέλλεται στο χρήστη μήνυμα αποτυχίας, αλλιώς γίνεται η διαγραφή όλων των στοιχείων που αφορούν το συγκεκριμένο πελάτη και η βάση ενημερώνεται με το ανανεωμένο σύνολο πελατών.

3.2.1.3. Διαχείριση πρώτων υλών

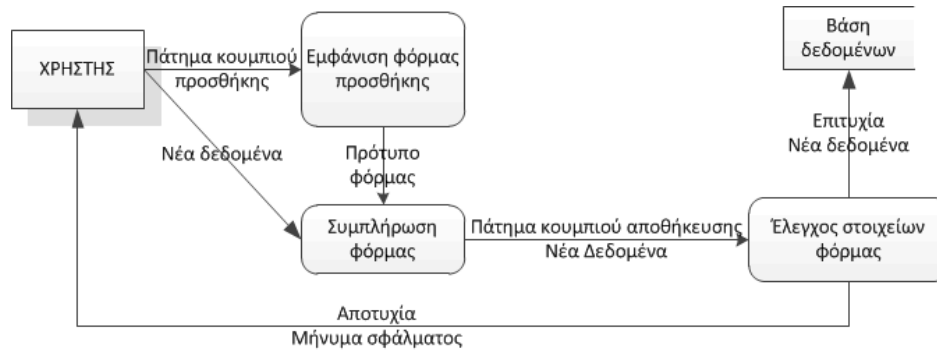


Διάγραμμα 3.7

Κατά τη λειτουργία διαχείρισης πρώτων υλών, προσφέρονται στο χρήστη οι εξής υπολειτουργίες:

- **Προσθήκη πρώτης ύλης:** Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της νέας πρώτης ύλης αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- **Αναζήτηση πρώτης ύλης:** Ο χρήστης μπορεί να αναζητήσει στοιχεία κάποιας πρώτης ύλης. Του επιστρέφεται μια λίστα με τα αποτελέσματα αναζήτησης.
- **Ενημέρωση στοιχείων πρώτης ύλης:** Ο χρήστης μπορεί να τροποποιήσει τα στοιχεία μιας πρώτης ύλης, αφού πρώτα την αναζητήσει. Η βάση δεδομένων ενημερώνεται με τα τροποποιημένα στοιχεία της πρώτης ύλης.
- **Διαγραφή πρώτης ύλης:** Ο χρήστης μπορεί να διαγράψει μια πρώτη ύλη, αφού πρώτα την αναζητήσει. Η βάση δεδομένων ενημερώνεται για τη διαγραφή της πρώτης ύλης, αν υπάρχει, διαφορετικά επιστρέφει μήνυμα αποτυχίας.
- **Προσθήκη προμηθευτή:** Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προμηθευτή αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- **Αναζήτηση προμηθευτή:** Ο χρήστης μπορεί να αναζητήσει στοιχεία κάποιου προμηθευτή. Του επιστρέφεται μια λίστα με τα αποτελέσματα αναζήτησης.
- **Ενημέρωση στοιχείων προμηθευτή:** Ο χρήστης μπορεί να τροποποιήσει τα στοιχεία ενός προμηθευτή, αφού πρώτα τον αναζητήσει. Η βάση δεδομένων ενημερώνεται με τα τροποποιημένα στοιχεία του προμηθευτή.
- **Διαγραφή προμηθευτή:** Ο χρήστης μπορεί να διαγράψει ένα προμηθευτή, αφού πρώτα τον αναζητήσει. Η βάση δεδομένων ενημερώνεται για τη διαγραφή του προμηθευτή, αν υπάρχει, διαφορετικά επιστρέφει μήνυμα αποτυχίας.

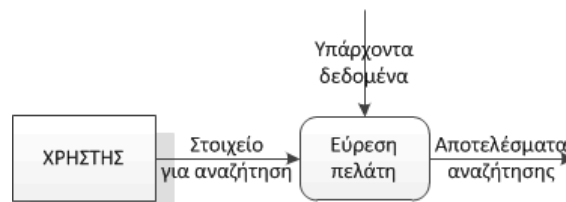
3.2.1.3.1. Προσθήκη πρώτης ύλης



Διάγραμμα 3.8

Ο χρήστης επιλέγει με το πάτημα ενός κουμπιού προσθήκης να εισάγει νέα δεδομένα, μια νέα πρώτη ύλη, στη βάση δεδομένων. Αφού εμφανιστεί η φόρμα προσθήκης, τα στοιχεία συμπληρώνονται από το χρήστη. Όταν ο χρήστης επιλέξει "Αποθήκευση δεδομένων", τα νέα δεδομένα ελέγχονται για την ορθότητά τους. Αν υπάρξει κάποιο λανθασμένο δεδομένο, ο χρήστης ενημερώνεται για το σφάλμα. Διαφορετικά, αν όλα τα στοιχεία είναι σωστά, τότε αποθηκεύονται στη βάση δεδομένων και ο χρήστης ενημερώνεται για τη νέα καταχώρηση.

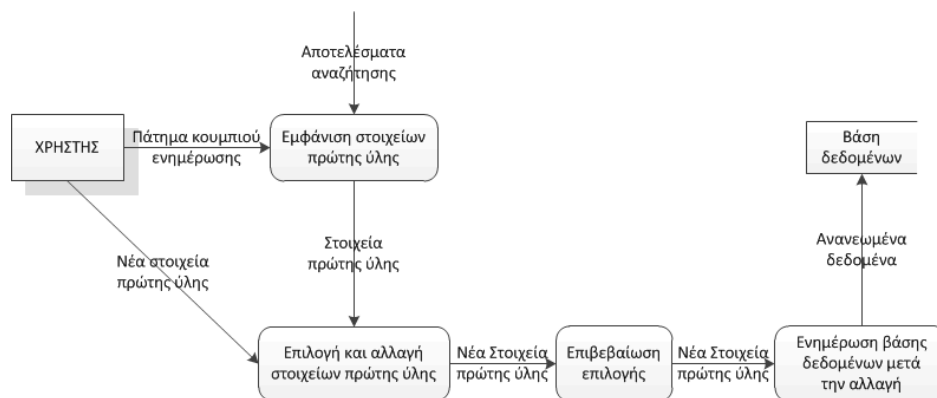
3.2.1.3.2. Αναζήτηση πρώτης ύλης



Διάγραμμα 3.9

Ο χρήστης εισάγει ένα στοιχείο αναζήτησης και το σύστημα του επιστρέφει μία λίστα με τις υπάρχουσες πρώτες ύλες που αναφέρονται με το στοιχείο αυτό.

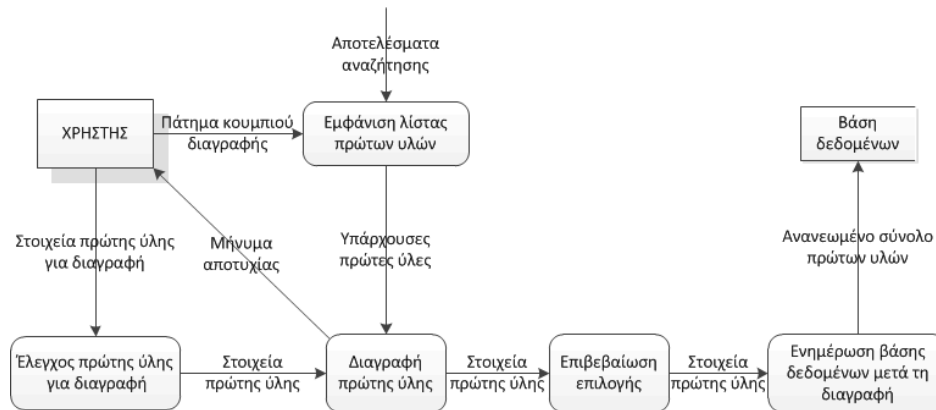
3.2.1.3.3. Ενημέρωση στοιχείων πρώτης ύλης



Διάγραμμα 3.10

Ο χρήστης επιλέγει να ενημερώσει τα στοιχεία μιας πρώτης ύλης πατώντας το αντίστοιχο κουμπί ενημέρωσης. Εμφανίζεται τότε μια λίστα με τις υπάρχουσες πρώτες ύλες, από την οποία ο χρήστης επιλέγει την πρώτη ύλη της οποίας θέλει να ενημερώσει τα στοιχεία. Ο χρήστης εισάγει τα νέα δεδομένα και επιλέγει να αποθηκευτούν οι αλλαγές. Αν ο χρήστης επιβεβαιώσει την επιλογή του, τότε η βάση δεδομένων ενημερώνεται.

3.2.1.3.4. Διαγραφή πρώτης ύλης



Διάγραμμα 3.11

Ο χρήστης επιλέγει να διαγράψει μια πρώτη ύλη πατώντας το αντίστοιχο κουμπί. Αφού εμφανιστεί μία λίστα με τις υπάρχουσες πρώτες ύλες, ο χρήστης δίνει ένα στοιχείο της πρώτης ύλης που θέλει να διαγράψει. Πριν γίνει η διαγραφή, γίνεται έλεγχος κατά πόσον το στοιχείο που δόθηκε αντιστοιχεί σε κάποια υπάρχουσα πρώτη ύλη. Αν δεν αντιστοιχεί, τότε στέλλεται στο χρήστη μήνυμα αποτυχίας, αλλιώς γίνεται η διαγραφή όλων των στοιχείων που αφορούν τη συγκεκριμένη πρώτη ύλη και η βάση ενημερώνεται με το ανανεωμένο σύνολο πρώτων υλών.

3.2.1.3.5. Προσθήκη προμηθευτή

Βλ. παρ. 3.2.1.2.1

3.2.1.3.6. Αναζήτηση προμηθευτή

Βλ. παρ. 3.2.1.2.2

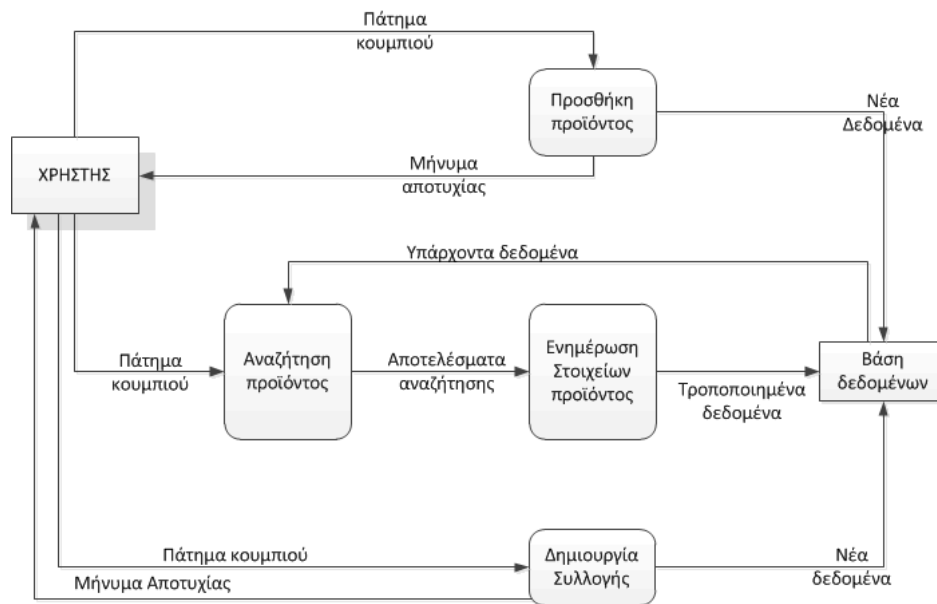
3.2.1.3.7. Ενημέρωση στοιχείων προμηθευτή

Βλ. παρ. 3.2.1.2.3

3.2.1.3.8. Διαγραφή προμηθευτή

Βλ. παρ. 3.2.1.2.4

3.2.1.4. Διαχείριση προϊόντων

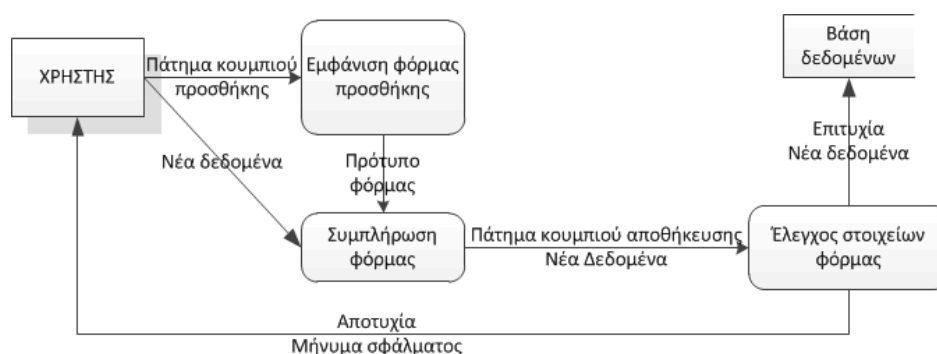


Διάγραμμα 3.12

Κατά τη λειτουργία διαχείρισης προϊόντων, προσφέρονται στο χρήστη οι εξής υπολειτουργίες:

- **Προσθήκη προϊόντος:** Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προϊόντος αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- **Αναζήτηση προϊόντος:** Ο χρήστης μπορεί να αναζητήσει στοιχεία κάποιου προϊόντος. Του επιστρέφεται μια λίστα με τα αποτελέσματα αναζήτησης.
- **Ενημέρωση στοιχείων προϊόντος:** Ο χρήστης μπορεί να τροποποιήσει τα στοιχεία ενός προϊόντος, αφού πρώτα το αναζητήσει. Η βάση δεδομένων ενημερώνεται με τα τροποποιημένα στοιχεία του προϊόντος.
- **Δημιουργία συλλογής:** Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της συλλογής αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.1.4.1. Προσθήκη προϊόντος

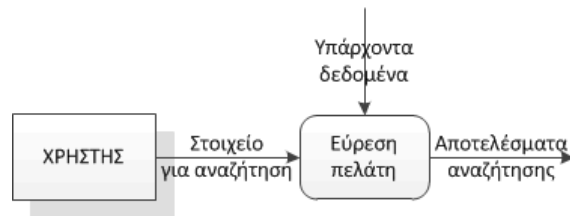


Διάγραμμα 3.13

Ο χρήστης επιλέγει με το πάτημα ενός κουμπιού προσθήκης να εισάγει νέα δεδομένα, ένα νέο προϊόν, στη βάση δεδομένων. Αφού εμφανιστεί η φόρμα προσθήκης, τα στοιχεία συμπληρώνονται

από το χρήστη. Όταν ο χρήστης επιλέξει "Αποθήκευση δεδομένων", τα νέα δεδομένα ελέγχονται για την ορθότητά τους. Αν υπάρξει κάποιο λανθασμένο δεδομένο, ο χρήστης ενημερώνεται για το σφάλμα. Διαφορετικά, αν όλα τα στοιχεία είναι σωστά, τότε αποθηκεύονται στη βάση δεδομένων και ο χρήστης ενημερώνεται για τη νέα καταχώρηση.

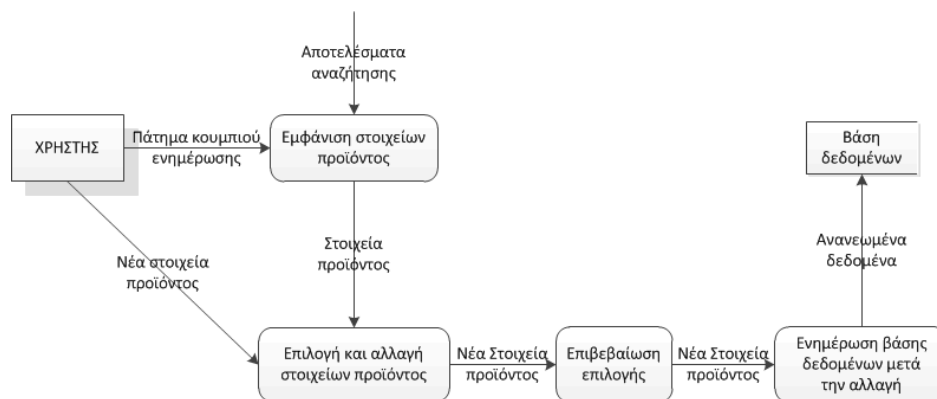
3.2.1.4.2. Αναζήτηση προϊόντος



Διάγραμμα 3.14

Ο χρήστης εισάγει ένα στοιχείο αναζήτησης και το σύστημα του επιστρέφει μία λίστα με τα υπάρχοντα προϊόντα που αναφέρονται με το στοιχείο αυτό.

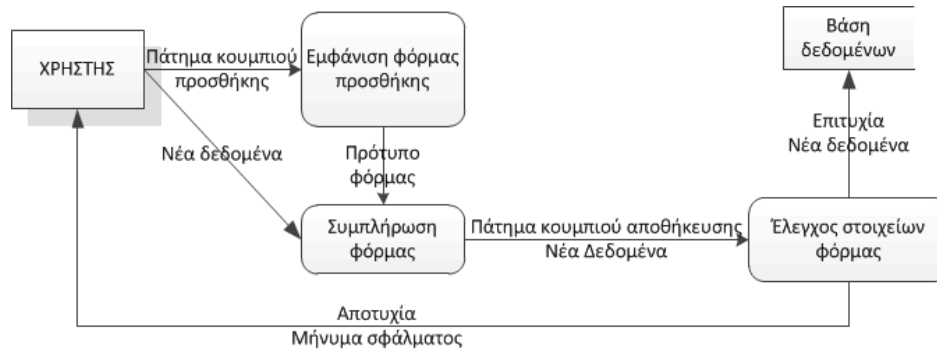
3.2.1.4.3. Ενημέρωση στοιχείων προϊόντος



Διάγραμμα 3.15

Ο χρήστης επιλέγει να ενημερώσει τα στοιχεία ενός προϊόντος πατώντας το αντίστοιχο κουμπί ενημέρωσης. Εμφανίζεται τότε μια λίστα με τα υπάρχοντα προϊόντα, από την οποία ο χρήστης επιλέγει το προϊόν του οποίου θέλει να ενημερώσει τα στοιχεία. Ο χρήστης εισάγει τα νέα δεδομένα και επιλέγει να αποθηκευτούν οι αλλαγές. Αν ο χρήστης επιβεβαιώσει την επιλογή του, τότε η βάση δεδομένων ενημερώνεται.

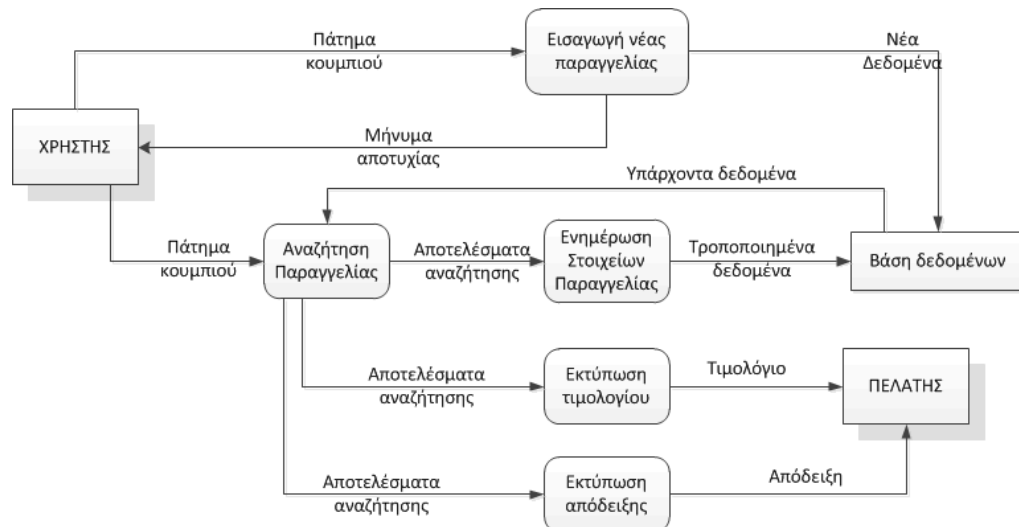
3.2.1.4.4. Δημιουργία συλλογής



Διάγραμμα 3.16

Ο χρήστης επιλέγει με το πάτημα ενός κουμπιού προσθήκης να εισάγει νέα δεδομένα, μία νέα συλλογή, στη βάση δεδομένων. Αφού εμφανιστεί η φόρμα προσθήκης, τα στοιχεία συμπληρώνονται από το χρήστη. Όταν ο χρήστης επιλέξει "Αποθήκευση δεδομένων", τα νέα δεδομένα ελέγχονται για την ορθότητά τους. Αν υπάρξει κάποιο λανθασμένο δεδομένο, ο χρήστης ενημερώνεται για το σφάλμα. Διαφορετικά, αν όλα τα στοιχεία είναι σωστά, τότε αποθηκεύονται στη βάση δεδομένων και ο χρήστης ενημερώνεται για τη νέα καταχώρηση.

3.2.1.5. Διαχείριση ταμειακής



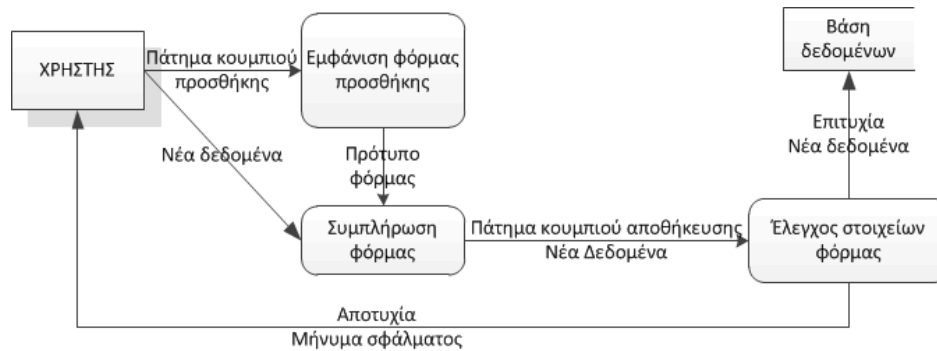
Διάγραμμα 3.17

Κατά τη λειτουργία διαχείρισης ταμειακής, προσφέρονται στο χρήστη οι εξής υπολειτουργίες:

- **Εισαγωγή νέας παραγγελίας:** Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της νέας παραγγελίας αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- **Αναζήτηση παραγγελίας:** Ο χρήστης μπορεί να αναζητήσει στοιχεία κάποιας παραγγελίας. Του επιστρέφεται μια λίστα με τα αποτελέσματα αναζήτησης.
- **Ενημέρωση στοιχείων παραγγελίας:** Ο χρήστης μπορεί να τροποποιήσει τα στοιχεία μιας παραγγελίας, αφού πρώτα την αναζητήσει. Η βάση δεδομένων ενημερώνεται με τα τροποποιημένα στοιχεία της παραγγελίας.

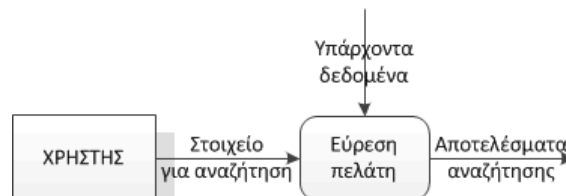
- **Εκτύπωση τιμολογίου:** Ο χρήστης μπορεί να εκτυπώσει το τιμολόγιο μιας παραγγελίας, αφού πρώτα την αναζητήσει. Το εκτυπωμένο τιμολόγιο παραδίδεται στον πελάτη.
- **Εκτύπωση απόδειξης:** Ο χρήστης μπορεί να εκτυπώσει την απόδειξη μιας εξοφλημένης παραγγελίας, αφού πρώτα την αναζητήσει. Η εκτυπωμένη απόδειξη παραδίδεται στον πελάτη.

3.2.1.5.1. Εισαγωγή νέας παραγγελίας



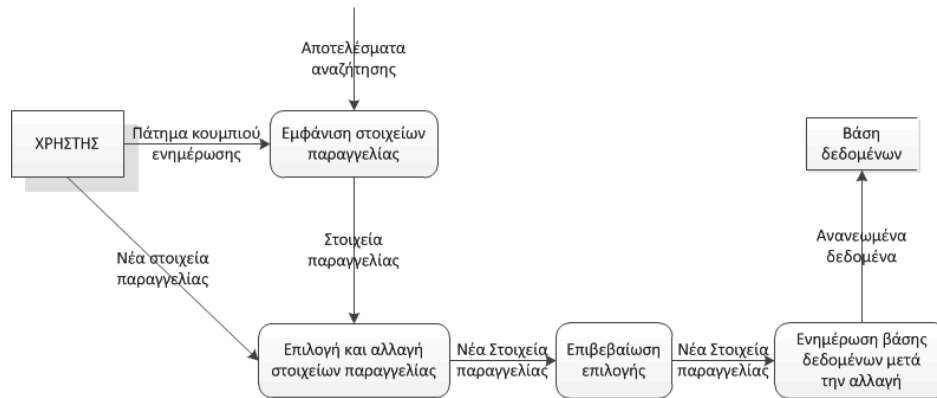
Ο χρήστης επιλέγει με το πάτημα ενός κουμπιού προσθήκης να εισάγει νέα δεδομένα, μια νέα παραγγελία, στη βάση δεδομένων. Αφού εμφανιστεί η φόρμα προσθήκης, τα στοιχεία συμπληρώνονται από το χρήστη. Όταν ο χρήστης επιλέξει "Αποθήκευση δεδομένων", τα νέα δεδομένα ελέγχονται για την ορθότητά τους. Αν υπάρξει κάποιο λανθασμένο δεδομένο, ο χρήστης ενημερώνεται για το σφάλμα. Διαφορετικά, αν όλα τα στοιχεία είναι σωστά, τότε αποθηκεύονται στη βάση δεδομένων και ο χρήστης ενημερώνεται για τη νέα καταχώρηση.

3.2.1.5.2. Αναζήτηση παραγγελίας



Ο χρήστης εισάγει ένα στοιχείο αναζήτησης και το σύστημα του επιστρέφει μία λίστα με τις υπάρχουσες παραγγελίες που αναφέρονται με το στοιχείο αυτό.

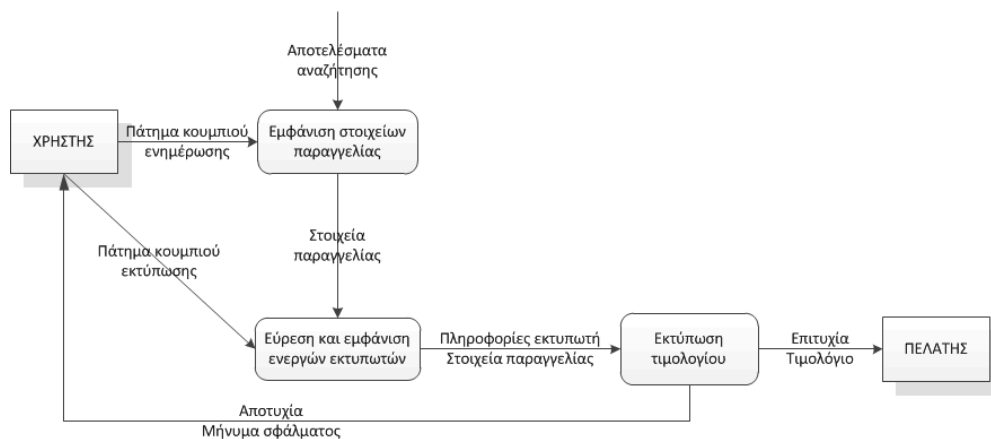
3.2.1.5.3. Ενημέρωση στοιχείων παραγγελίας



Διάγραμμα 3.20

Ο χρήστης επιλέγει να ενημερώσει τα στοιχεία μιας παραγγελίας πατώντας το αντίστοιχο κουμπί ενημέρωσης. Εμφανίζεται τότε μια λίστα με τις υπάρχουσες παραγγελίες, από την οποία ο χρήστης επιλέγει τη παραγγελία της οποίας θέλει να ενημερώσει τα στοιχεία. Ο χρήστης εισάγει τα νέα δεδομένα και επιλέγει να αποθηκευτούν οι αλλαγές. Αν ο χρήστης επιβεβαιώσει την επιλογή του, τότε η βάση δεδομένων ενημερώνεται.

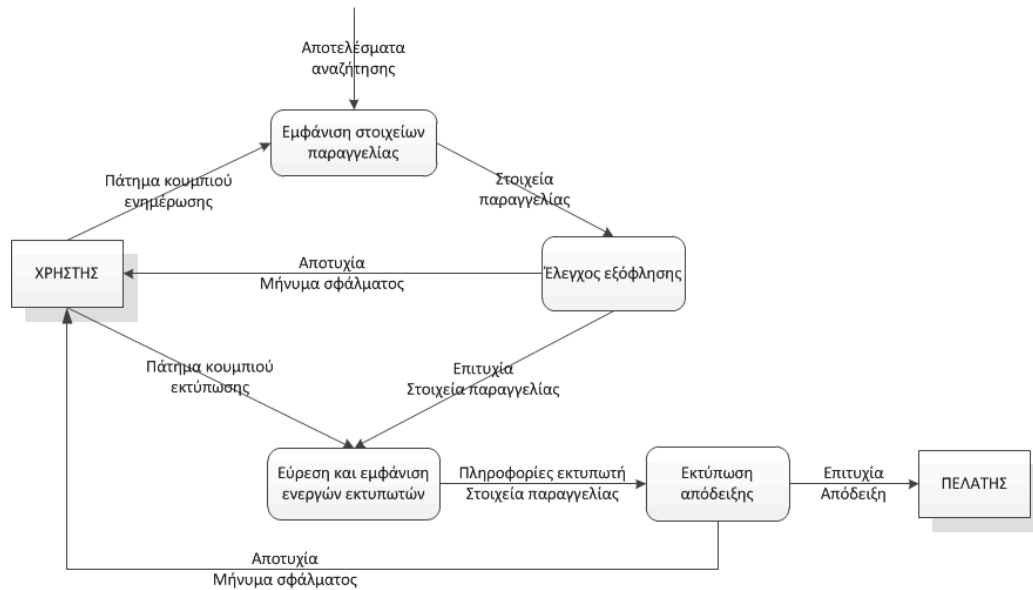
3.2.1.5.4. Εκτύπωση τιμολογίου



Διάγραμμα 3.21

Ο χρήστης επιλέγει να εκτυπώσει το τιμολόγιο για μια παραγγελία με το πάτημα του κουμπιού εκτύπωσης. Γίνεται από το σύστημα μία αναζήτηση στο δίκτυο για να βρεθούν οι ενεργοί εκτυπωτές. Εμφανίζεται η λίστα με τους ενεργούς εκτυπωτές στον χρήστη και επιλέγει. Αν η εκτύπωση αποτύχει, του εμφανίζεται μήνυμα αποτυχίας στην εκτύπωση. Διαφορετικά το τιμολόγιο εκτυπώνεται και παραδίδεται στον πελάτη.

3.2.1.5.5. Εκτύπωση απόδειξης



Διάγραμμα 3.22

Ο χρήστης επιλέγει να εκτυπώσει την απόδειξη μιας παραγγελία με το πάτημα του κουμπιού εκτύπωσης. Η παραγγελία ελέγχεται κατά πόσον έχει εξοφληθεί. Αν δεν έχει εξοφληθεί, εμφανίζεται στον χρήστη μήνυμα σφάλματος. Διαφορετικά, γίνεται από το σύστημα μία αναζήτηση στο δίκτυο για να βρεθούν οι ενεργοί εκτυπωτές. Εμφανίζεται η λίστα με τους ενεργούς εκτυπωτές στον χρήστη και επιλέγει. Αν η εκτύπωση αποτύχει, του εμφανίζεται μήνυμα αποτυχίας στην εκτύπωση. Αλλιώς, η απόδειξη εκτυπώνεται και παραδίδεται στον πελάτη.

3.2.2. Μηχανοργάνωση

Το σύστημα που περιγράφηκε στο προηγούμενο βήμα μέσω των διαγραμμάτων ροής δεδομένων, αναμένεται να υλοποιηθεί όλο.

3.2.3. Εκλέπτυνση ροών δεδομένων

Πιο κάτω, παρουσιάζεται η εκλέπτυνση των ροών δεδομένων που εμφανίζονται στα διαγράμματα ροής δεδομένων. Η εκλέπτυνση ροών δεδομένων αφορά τον ακριβή καθορισμό των πληροφοριών, οι οποίες διακινούνται κατά την εκτέλεση των διαφόρων διεργασιών.

Έννοιες που εμφανίζονται στα διαγράμματα ροής και έχουν την ίδια σημασία:

Αποτελέσματα αναζήτησης: Εμφάνιση αποτελεσμάτων αναζήτησης σε μορφή λίστας στην οθόνη

Μήνυμα σφάλματος: «Σφάλμα! Παρακαλώ ελέγξτε τα στοιχεία...»

Μήνυμα ορθότητας: «Η καταχώρηση/ενημέρωση/διαγραφή έγινε σωστά!»

Μήνυμα επιτυχίας: «Επιτυχής αποθήκευση δεδομένων»

Μήνυμα αποτυχίας: «Αποτυχία κατά την αποθήκευση! Παρακαλώ ελέγξτε...»

3.2.3.1. Επίπεδο 0

- Οι κωδικοί πρόσβασης αναφέρονται στα username και password.
- Τα δεδομένα στις ροές δεδομένων αναφέρονται είτε σε υπάρχοντα δεδομένα της βάσης, είτε σε (νέα) τροποποιημένα δεδομένα.

3.2.3.2. Διαχείριση πελατών

Τα διάφορα δεδομένα πελάτη αναφέρονται στα πιο κάτω:

Αν ο πελάτης είναι ένα άτομο

<u>Βασικά δεδομένα</u>	<u>Διεύθυνση</u>	<u>Στοιχεία Επικοινωνίας</u>	<u>Αναλογίες σώματος</u>
▪ Κωδικός πελάτη ▪ Ονοματεπώνυμο ▪ Ημερ. Γενεθλίων ▪ Ημερ. Γιορτής	▪ Οδός ▪ Αριθμός ▪ Περιοχή ▪ Ταχ. Κώδικας ▪ Πόλη ▪ Χώρα	▪ Τηλέφωνο ▪ Κινητό ▪ Email	▪ Στήθος ▪ Μέση ▪ Γλουτοί

Αν ο πελάτης είναι μια εταιρεία

<u>Βασικά δεδομένα</u>	<u>Διεύθυνση</u>	<u>Στοιχεία Επικοινωνίας</u>
▪ Κωδικός πελάτη ▪ Όνομα εταιρείας ▪ Όνομα μπουτίκ ▪ Άτομο επικοινωνίας	▪ Οδός ▪ Αριθμός ▪ Περιοχή ▪ Ταχ. Κώδικας ▪ Πόλη ▪ Χώρα	▪ Τηλέφωνο ▪ Κινητό ▪ Email ▪ Φαξ ▪ Website

3.2.3.2.1. Προσθήκη πελάτη

Ο χρήστης πρέπει να εισάγει όλα τα πιο πάνω δεδομένα κατά την προσθήκη ενός νέου πελάτη στη βάση δεδομένων. Σε περίπτωση κάποιου σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου πελάτη αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.3.2.2. Αναζήτηση πελάτη

Κατά την αναζήτηση, δίνονται μερικά στοιχεία για ένα πελάτη. Η λειτουργία επιστρέφει μία λίστα με τα πιο πάνω δεδομένα.

3.2.3.2.3. Ενημέρωση στοιχείων πελάτη

Κατά την ενημέρωση στοιχείων, δίνονται όλα τα στοιχεία ενός πελάτη και ενημερώνεται η βάση δεδομένων.

3.2.3.2.4. Διαγραφή πελάτη

Μέσω της αναζήτησης, ανακτώνται όλα τα στοιχεία του πελάτη και ενημερώνεται η βάση δεδομένων ότι η συγκεκριμένη εγγραφή θα διαγραφεί. Αν δεν υπάρχει ο ζητούμενος πελάτης, επιστρέφεται μήνυμα αποτυχίας.

3.2.3.3. Διαχείριση πρώτων υλών

Τα διάφορα δεδομένα πρώτων υλών αναφέρονται στα πιο κάτω:

<u>Βασικά δεδομένα</u>	<u>Προμηθευτής</u>	<u>Αποθήκη</u>
▪ Κωδικός ύλης ▪ Περιγραφή ▪ Σύσταση ▪ Χρώμα	▪ Όνομα ▪ Κόστος ύλης	▪ Αρχική ποσότητα ▪ Ελάχιστη ποσότητα ▪ Απόθεμα ▪ Μονάδα μέτρησης

3.2.3.3.1. Προσθήκη πρώτης ύλης

Ο χρήστης πρέπει να εισάγει όλα τα πιο πάνω δεδομένα κατά την προσθήκη μιας νέας πρώτης ύλης στη βάση δεδομένων. Σε περίπτωση κάποιου σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της νέας πρώτης ύλης αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.3.3.2. Αναζήτηση πρώτης ύλης

Κατά την αναζήτηση, δίνονται μερικά στοιχεία για μια πρώτη ύλη. Η λειτουργία επιστρέφει μία λίστα με τα πιο πάνω δεδομένα.

3.2.3.3.3. Ενημέρωση στοιχείων πρώτης ύλης

Κατά την ενημέρωση στοιχείων, δίνονται όλα τα στοιχεία μιας πρώτης ύλης και ενημερώνεται η βάση δεδομένων.

3.2.3.3.4. Διαγραφή πρώτης ύλης

Μέσω της αναζήτησης, ανακτώνται όλα τα στοιχεία της πρώτης ύλης και ενημερώνεται η βάση δεδομένων ότι η συγκεκριμένη εγγραφή θα διαγραφεί. Αν δεν υπάρχει η ζητούμενη πρώτη ύλη, επιστρέφεται μήνυμα αποτυχίας.

3.2.3.3.5. Προσθήκη προμηθευτή

Ο χρήστης πρέπει να εισάγει τα πιο κάτω δεδομένα κατά την προσθήκη ενός νέου προμηθευτή στη βάση δεδομένων:

<u>Βασικά δεδομένα</u>	<u>Διεύθυνση</u>	<u>Στοιχεία Επικοινωνίας</u>
▪ Κωδικός προμηθευτή ▪ Όνομα προμηθευτή ▪ Άτομο επικοινωνίας	▪ Οδός ▪ Αριθμός ▪ Περιοχή ▪ Ταχ. Κώδικας ▪ Πόλη ▪ Χώρα	▪ Τηλέφωνο ▪ Κινητό ▪ Email ▪ Φαξ

Σε περίπτωση κάποιου σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προμηθευτή αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.3.3.6. Αναζήτηση προμηθευτή

Κατά την αναζήτηση, δίνονται μερικά στοιχεία για κάποιο προμηθευτή. Η λειτουργία επιστρέφει μία λίστα με τα πιο πάνω δεδομένα.

3.2.3.3.7. Ενημέρωση στοιχείων προμηθευτή

Κατά την ενημέρωση στοιχείων, δίνονται όλα τα στοιχεία ενός προμηθευτή και ενημερώνεται η βάση δεδομένων.

3.2.3.3.8. Διαγραφή προμηθευτή

Μέσω της αναζήτησης, ανακτώνται όλα τα στοιχεία του προμηθευτή και ενημερώνεται η βάση δεδομένων ότι η συγκεκριμένη εγγραφή θα διαγραφεί. Αν δεν υπάρχει ο ζητούμενος προμηθευτής, επιστρέφεται μήνυμα αποτυχίας.

3.2.3.4. Διαχείριση προϊόντων

Τα διάφορα δεδομένα προϊόντων αναφέρονται στα πιο κάτω:

<u>Βασικά δεδομένα</u>	<u>Υλικά κατασκευής</u>	<u>Τιμές</u>
▪ Κωδικός προϊόντος ▪ Περιγραφή ▪ Μέγεθος ▪ Διαθέσιμα χρώματα ▪ Εικόνα ▪ Κατηγορία προϊόντος ▪ Συλλογή προϊόντος	▪ Πρώτες ύλες ▪ Ποσότητα	▪ Χονδρική Τιμή ▪ Λιανική Τιμή ▪ Κόστος κατασκευής

3.2.3.4.1. Προσθήκη προϊόντος

Ο χρήστης πρέπει να εισάγει όλα τα πιο πάνω δεδομένα κατά την προσθήκη ενός νέου προϊόντος στη βάση δεδομένων. Σε περίπτωση κάποιου σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προϊόντος αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.3.4.2. Αναζήτηση προϊόντος

Κατά την αναζήτηση, δίνονται μερικά στοιχεία για ένα προϊόν. Η λειτουργία επιστρέφει μία λίστα με τα πιο πάνω δεδομένα.

3.2.3.4.3. Ενημέρωση στοιχείων προϊόντος

Κατά την ενημέρωση στοιχείων, δίνονται όλα τα στοιχεία ενός προϊόντος και ενημερώνεται η βάση δεδομένων.

3.2.3.4.4. Δημιουργία συλλογής

Τα διάφορα δεδομένα συλλογής προϊόντων αναφέρονται στα πιο κάτω:

- Κωδικός συλλογής
- Όνομα συλλογής
- Χρονική Περίοδος

Ο χρήστης πρέπει να εισάγει όλα τα πιο πάνω δεδομένα κατά την δημιουργία μιας νέας συλλογής στη βάση δεδομένων. Σε περίπτωση κάποιου σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προϊόντος αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.3.5. Διαχείριση ταμειακής (Παραγγελίες, Πωλήσεις)

3.2.3.5.1. Εισαγωγή νέας παραγγελίας

Τα διάφορα δεδομένα παραγγελίας αναφέρονται στα πιο κάτω:

<u>Βασικά δεδομένα</u>	<u>Αντικείμενα</u>
▪ Αριθμός παραγγελίας ▪ Κωδικός πελάτη/εταιρείας ▪ Ημερομηνία παραγγελίας ▪ Προθεσμία ▪ Μερικό Σύνολο ▪ ΦΠΑ ▪ Ολικό Σύνολο ▪ Έκπτωση (%) ▪ Έκπτωση ▪ Νέο ολικό ▪ Προκαταβολή (%) ▪ Προκαταβολή ▪ Σημειώσεις ▪ Κατάσταση παραγγελίας	▪ Προϊόν ▪ Ποσότητα ▪ Τιμή ανά τεμάχιο ▪ Μερικό Σύνολο

Ο χρήστης πρέπει να εισάγει όλα τα πιο πάνω δεδομένα κατά την εισαγωγή μιας νέας παραγγελίας στη βάση δεδομένων. Σε περίπτωση κάποιου σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της νέας παραγγελίας αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.2.3.5.2. Αναζήτηση παραγγελίας

Κατά την αναζήτηση, δίνονται μερικά στοιχεία για μια παραγγελία. Η λειτουργία επιστρέφει μία λίστα με τα πιο πάνω δεδομένα.

3.2.3.5.3. Ενημέρωση στοιχείων παραγγελίας

Κατά την ενημέρωση στοιχείων, δίνονται όλα τα στοιχεία μιας παραγγελίας και ενημερώνεται η βάση δεδομένων.

3.2.3.5.4. Εκτύπωση τιμολογίου

Τα διάφορα δεδομένα τιμολογίου αναφέρονται στα πιο κάτω:

- | | |
|------------------------|-----------------|
| • Αριθμός Παραγγελίας | • Μερικό Σύνολο |
| • Ονοματεπώνυμο πελάτη | • ΦΠΑ |
| • Ημερομηνία έκδοσης | • Έκπτωση (%) |
| • Προθεσμία | • Έκπτωση |
| • Κωδικός προϊόντος | • Ολικό Σύνολο |
| • Ποσότητα | • Σημειώσεις |
| • Τιμή ανά τεμάχιο | • |

Όλα τα πιο πάνω δεδομένα εμφανίζονται στην οθόνη και εκτυπώνονται.

3.2.3.5.5. Εκτύπωση απόδειξης

Τα διάφορα δεδομένα απόδειξης αναφέρονται στα πιο κάτω:

- Κωδικός απόδειξης
- Ημερομηνία πληρωμής
- Τρόπος πληρωμής
- Πληρωθέν ποσό
- Λόγος πληρωμής

Όλα τα πιο πάνω δεδομένα εμφανίζονται στην οθόνη και εκτυπώνονται.

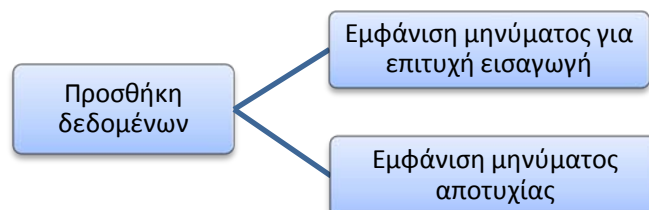
3.2.4. Εκλέπτυνση λογικής διαδικασιών

Το σύστημα θα εκτελεί κάποιες διαδικασίες για τις οποίες χρειάζεται η επικοινωνία με την βάση δεδομένων. Οι διαδικασίες αυτές χρειάζεται να επιστρέφουν μηνύματα επιτυχίας ή αποτυχίας. Για αυτό τον λόγο υπάρχει ανάγκη λογικής. Πιο κάτω παρουσιάζεται ο προσδιορισμός της λογικής για αυτές τις διαδικασίες.

3.2.4.1. Προσθήκη δεδομένων

Η πιο κάτω λογική μπορεί να εκτελεστεί στις περιπτώσεις των:

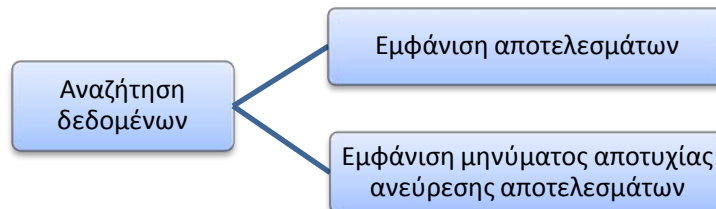
- Εισαγωγή πελάτη/συνεργαζόμενης εταιρείας
- Εισαγωγή πρώτης ύλης
- Εισαγωγή προμηθευτή
- Εισαγωγή προϊόντος
- Δημιουργία συλλογής
- Εισαγωγή νέας παραγγελίας



3.2.4.2. Αναζήτηση δεδομένων

Η πιο κάτω λογική μπορεί να εκτελεστεί στις περιπτώσεις των:

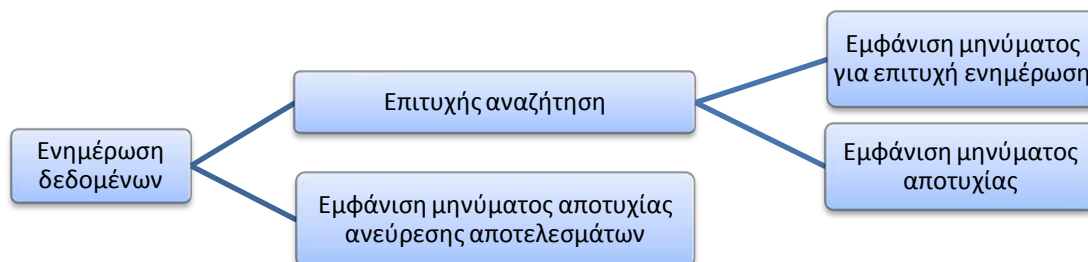
- Αναζήτηση πελάτη/συνεργαζόμενης εταιρείας
- Αναζήτηση πρώτης ύλης
- Αναζήτηση προϊόντος
- Αναζήτηση προμηθευτή
- Αναζήτηση παραγγελίας



3.2.4.3. Ενημέρωση δεδομένων

Η πιο κάτω λογική μπορεί να εκτελεστεί στις περιπτώσεις των:

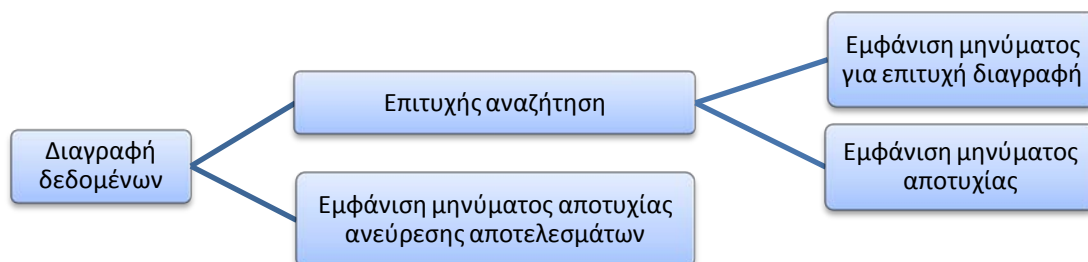
- Ενημέρωση στοιχείων πελάτη/συνεργαζόμενης εταιρείας
- Ενημέρωση στοιχείων πρώτης ύλης
- Ενημέρωση στοιχείων προμηθευτή
- Ενημέρωση στοιχείων προϊόντος
- Ενημέρωση στοιχείων παραγγελίας



3.2.4.4. Διαγραφή δεδομένων

Η πιο κάτω λογική μπορεί να εκτελεστεί στις περιπτώσεις των:

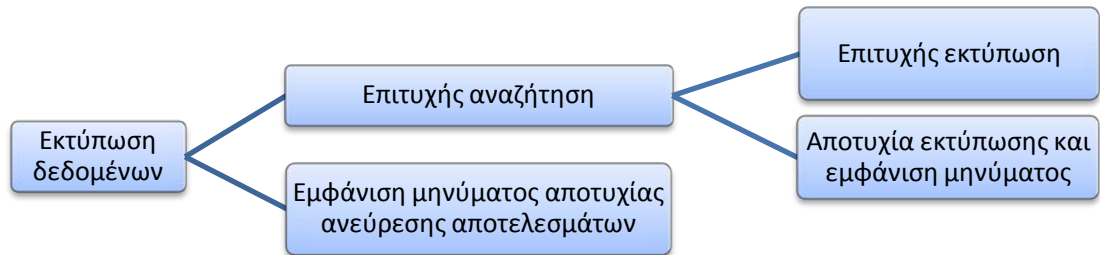
- Διαγραφή πελάτη/συνεργαζόμενης εταιρείας
- Διαγραφή πρώτης ύλης
- Διαγραφή προμηθευτή



3.2.4.5. Εκτύπωση δεδομένων

Η πιο κάτω λογική μπορεί να εκτελεστεί στις περιπτώσεις των:

- Εκτύπωση τιμολογίου
- Εκτύπωση απόδειξης



3.2.5. Εκλέπτυνση αποθήκευσης δεδομένων

Στην παράγραφο αυτή, παρουσιάζονται τα ακριβή περιεχόμενα κάθε αποθηκευτικής μονάδας που βρίσκεται στη βάση δεδομένων του συστήματος που θα αναπτυχθεί σε μορφή πινάκων. Καθορίζονται επίσης και το πού απαιτείται άμεση πρόσβαση.

Ρυθμίσεις Παραμέτρων
Τρέχον χρονική περίοδος
ΦΠΑ
Διεύθυνση email
Κωδικός email
SMTP Διακομιστής
SMTP Θύρα
Αποστολή email?
Ευχή γενεθλίων
Ευχή ονομαστικής εορτής

Ονομαστικές εορτές
 A/A
Ημερομηνία
Όνομα
Σημείωση

Κατάσταση παραγγελίας
 A/A
Κατάσταση

Υπόμνημα

 = Πρωτεύον κλειδί
 = Ξένο κλειδί

Συνεργαζόμενη εταιρεία
 Κωδικός
Όνομα Εταιρείας
Όνομα μπουτίκ
Άτομο επικοινωνίας
Οδός
Αριθμός
Περιοχή
Ταχ. Κώδικας
Πόλη
Χώρα
Τηλέφωνο
Κινητό
Φαξ
Email
Website

Πελάτης
 Κωδικός
Ονοματεπώνυμο
Οδός
Αριθμός
Περιοχή
Ταχ. Κώδικας
Πόλη
Χώρα
Τηλέφωνο
Κινητό
Email
Ημερ. Γενεθλίων
 Ημερ. Εορτής
Στήθος
Μέση
Γλουτοί

Προμηθευτής
 Κωδικός
Όνομα προμηθευτή
Άτομο επικοινωνίας
Οδός
Αριθμός
Περιοχή
Ταχ. Κώδικας
Πόλη
Χώρα
Τηλέφωνο
Κινητό
Φαξ
Email

Προϊόν
 Κωδικός
Περιγραφή
Μέγεθος
 Χρώμα
Κόστος κατασκευής
Λιανική τιμή
Εικόνα

Στοιχεία Κατασκευής Προϊόντος
 Κωδικός Προϊόντος
 Κωδικός πρώτης ύλης
Ποσότητα

Πρώτες Ύλες
 Κωδικός
Περιγραφή
Σύσταση
 Χρώμα
Αρχική Ποσότητα
Κόστος αγοράς
 Προμηθευτής
Απόθεμα
Ελάχιστη Ποσότητα
Μονάδα μέτρησης

Προϊόντα σε συλλογές
 Κωδικός
Περιγραφή
Μέγεθος
 Χρώμα
Κόστος κατασκευής
Χονδρική τιμή
Εικόνα
 Συλλογή
 Κατηγορία

Κατηγορίες ειδών ένδυσης
 Κωδικός κατηγορίας
Όνομα κατηγορίας

Στοιχεία κατασκευής είδους ένδυσης
 Κωδικός είδους ένδυσης
 Κωδικός πρώτης ύλης
Ποσότητα

Συλλογή προϊόντων
 Κωδικός
Όνομα
Χρονική περίοδος
Περιγραφή

Χρωματικός συνδυασμός
 Κωδικός
Χρώμα

Λιανικές Παραγγελίες
 Αριθμός Παραγγελίας
 Κωδικός πελάτη
Ημερομηνία παραγγελίας
Προθεσμία
 Κωδικός προϊόντος
Ποσότητα
Τιμή ανά τεμάχιο
Μερικό Σύνολο
ΦΠΑ
Έκπτωση (%)
Έκπτωση
Σύνολο
Σημειώσεις
 Κατάσταση Παραγγελίας

Χονδρικές Παραγγελίες
 Αριθμός παραγγελίας
 Κωδικός εταιρείας
Ημερομηνία παραγγελίας
Προθεσμία
Μερικό Σύνολο
ΦΠΑ
Ολικό Σύνολο
Έκπτωση (%)
Έκπτωση
Νέο ολικό
Προκαταβολή (%)
Προκαταβολή
Σημειώσεις
 Κατάσταση παραγγελίας

Λεπτομέρειες χονδρικής παραγγελίας
 A/A
 Αριθμός παραγγελίας
 Κωδικός είδους ένδυσης
Ποσότητα
Τιμή ανά τεμάχιο
Μερικό Σύνολο

Λιανικές Πωλήσεις
 Αριθμός Τιμολογίου
 Αριθμός Παραγγελίας
Τρόπος πληρωμής
Ημερομηνία πληρωμής
ΦΠΑ
Μερικό Σύνολο
Έκπτωση (%)
Έκπτωση
Ολικό σύνολο

Χονδρικές πωλήσεις
 Αριθμός Τιμολογίου
 Αριθμός Παραγγελίας
Τρόπος πληρωμής
Ημερομηνία πληρωμής
Ολικό σύνολο
Πληρωθέν ποσό
Υπόλοιπο
Λόγος πληρωμής

3.2.6. Ορισμός φυσικών πόρων

Όλα τα δεδομένα του νέου συστήματος θα αποθηκεύονται στη βάση δεδομένων. Συγκεκριμένα, το μέσο αποθήκευσης είναι ο σκληρός δίσκος του ηλεκτρονικού υπολογιστή που θα χρησιμοποιείται, αφού δεν υπάρχει δίκτυο για να κριθεί αναγκαία η χρήση εξυπηρετητή δικτύου (server) για τη βάση δεδομένων. Τα δεδομένα αυτά, θα αποθηκεύονται σε κατάλληλους πίνακες που θα υπάρχουν στη βάση, ανάλογα με το τι αντιπροσωπεύουν.

Η βάση δεδομένων θα αναπτυχθεί με το λογισμικό *Microsoft SQL Server® 2008 ManagementStudio*. Πιο κάτω φαίνονται τα μεγέθη των τύπων που θα χρησιμοποιηθούν¹.

Τύπος Δεδομένων	Απαιτούμενη Μνήμη (bytes)
bit	1
date	3
int	4
decimal(p, s)	5
datetime	8
money	8
nchar(n)	2n
nvarchar(n)	2n + 2
image	Μέχρι 16 KB

Η οργάνωση των πινάκων είναι indexed. Τα κλειδιά με τα οποία γίνονται indexed οι πίνακες αναφέρονται. Οι πίνακες που θα αποτελούν τη βάση δεδομένων περιγράφονται πιο κάτω μαζί με τον τύπο δεδομένων του κάθε πεδίου και τα πρωτεύοντα, αλλά και ξένα, κλειδιά με τα οποία γίνεται αναφορά στους πίνακες αυτούς.

Πίνακας1: Ρυθμίσεις παραμέτρων

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Τρέχον χρονική περίοδος ΦΠΑ	nvarchar(50)	102	-	Not Null
	decimal(5, 2)	5	-	Not Null
Διεύθυνση email	nvarchar(50)	102	-	Not Null
Κωδικός email	nvarchar(50)	102	-	Not Null
SMTP Διακομιστής	nvarchar(50)	102	-	Not Null
SMTP Θύρα	int	4	-	Not Null
Αποστολή email?	bit	1	-	Not Null
Ευχή γενεθλίων	nvarchar(250)	502	-	Not Null
Ευχή ονομαστικής εορτής	nvarchar(250)	502	-	Not Null
Συνολικό μέγεθος		1422		

Πίνακας2: Ονομαστικές εορτές

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
A/A	int	4	Primary	Not Null

¹ Οι πληροφορίες πάρθηκαν από το Help του SQLServer® 2008 στην ιστοσελίδα:
<http://msdn.microsoft.com/en-us/library/ms187752.aspx>

Ημερομηνία	date	3	-	Not Null
Όνομα	nvarchar(250)	502	-	Not Null
Σημείωση	nchar(20)	40	-	Null
Συνολικό μέγεθος		549		

Πίνακας3: Κατάσταση παραγγελίας

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
A/A	int	4	Primary	Not Null
Κατάσταση	nvarchar(50)	102	-	Not Null
Συνολικό μέγεθος		106		

Πίνακας4: Πελάτης

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κώδικας	int	4	Primary	Not Null
Ονοματεπώνυμο	nvarchar(50)	102	-	Not Null
Οδός	nvarchar(50)	102	-	Null
Αριθμός	int	4	-	Null
Περιοχή	nvarchar(50)	102	-	Null
Ταχ. Κώδικας	int	4	-	Null
Πόλη	nvarchar(50)	102	-	Null
Χώρα	nvarchar(50)	102	-	Null
Τηλέφωνο	int	4	-	Null
Κινητό	int	4	-	Null
Email	nvarchar(50)	102	-	Null
Ημερ. Γενεθλίων	date	3	-	Null
Ημερ. Εορτής	int	4	-	Null
Στήθος	int	4	-	Null
Μέση	int	4	-	Null
Γλουτοί	int	4	-	Null
Συνολικό μέγεθος		652		

Πίνακας5: Συνεργαζόμενη εταιρεία

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός	int	4	Primary	Not Null
Όνομα Εταιρείας	nvarchar(50)	102	-	Not Null
Όνομα μπουτίκ	nvarchar(50)	102	-	Not Null
Άτομο επικοινωνίας	nvarchar(50)	102	-	Not Null
Οδός	nvarchar(50)	102	-	Null
Αριθμός	int	4	-	Null
Περιοχή	nvarchar(50)	102	-	Null
Ταχ. Κώδικας	int	4	-	Null
Πόλη	nvarchar(50)	102	-	Null
Χώρα	nvarchar(50)	102	-	Null
Τηλέφωνο	int	4	-	Null

Κινητό	int	4	-	Null
Φαξ	int	4	-	Null
Email	nvarchar(50)	102	-	Null
Website	nvarchar(50)	102	-	Null
Συνολικό μέγεθος		942		

Πίνακας6: Προμηθευτής

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός	int	4	Primary	Not Null
Όνομα προμηθευτή	nvarchar(50)	102	-	Not Null
Άτομο επικοινωνίας	nvarchar(50)	102	-	Null
Οδός	nvarchar(50)	102	-	Null
Αριθμός	int	4	-	Null
Περιοχή	nvarchar(50)	102	-	Null
Ταχ. Κώδικας	int	4	-	Null
Πόλη	nvarchar(50)	102	-	Null
Χώρα	nvarchar(50)	102	-	Null
Τηλέφωνο	int	4	-	Null
Κινητό	int	4	-	Null
Φαξ	int	4	-	Null
Email	nvarchar(50)	102	-	Null
Συνολικό μέγεθος		738		

Πίνακας7: Πρώτες ύλες

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός	nvarchar(10)	22	Primary	Not Null
Περιγραφή	nvarchar(50)	102	-	Not Null
Σύσταση	nvarchar(50)	102	-	Null
Χρώμα	nvarchar(10)	22	-	Null
Αρχική Ποσότητα	int	4	-	Null
Κόστος αγοράς	money	8	-	Null
Προμηθευτής	int	4	-	Null
Απόθεμα	int	4	-	Null
Ελάχιστη Ποσότητα	int	4	-	Null
Μονάδα μέτρησης	nvarchar(7)	16	-	Null
Συνολικό μέγεθος		288		

Πίνακας8: Προϊόν

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός	nvarchar(10)	22	Primary	Not Null
Περιγραφή	nvarchar(50)	102	-	Not Null
Μέγεθος	nvarchar(50)	102	-	Null
Χρώμα	nvarchar(10)	22	-	Null
Κόστος κατασκευής	money	8	-	Null

Λιανική τιμή	money	8	-	Null
Εικόνα	image		-	Null
Συνολικό μέγεθος		264		

Πίνακας9: Στοιχεία Κατασκευής Προϊόντος

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός Προϊόντος	nvarchar(10)	22	Primary	Not Null
Κωδικός πρώτης ύλης	nvarchar(10)	22	Primary	Not Null
Ποσότητα	int	4	-	Not Null
Συνολικό μέγεθος		48		

Πίνακας10: Προϊόντα σε συλλογή

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός	nvarchar(10)	22	Primary	Not Null
Περιγραφή	nvarchar(50)	102	-	Not Null
Μέγεθος	nvarchar(20)	102	-	Null
Χρώμα	nvarchar(10)	22	-	Null
Κόστος κατασκευής	money	8	-	Null
Χονδρική τιμή	money	8	-	Null
Εικόνα	image		-	Null
Συλλογή	int	4	-	Null
Κατηγορία	int	4	-	Null
Συνολικό μέγεθος		272		

Πίνακας11: Κατηγορίες ειδών ένδυσης

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός κατηγορίας	int	4	Primary	Not null
Όνομα κατηγορίας	nvarchar(50)	102	-	Not null
Συνολικό μέγεθος		106		

Πίνακας12: Στοιχεία Κατασκευής Είδους ένδυσης

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός είδους ένδυσης	nvarchar(10)	22	Primary	Not Null
Κωδικός πρώτης ύλης	nvarchar(10)	22	Primary	Not Null
Ποσότητα	int	4	-	Not Null
Συνολικό μέγεθος		48		

Πίνακας13: Συλλογή προϊόντων

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός	int	4	Primary	Not Null
Όνομα συλλογής	nvarchar(20)	42	-	Not Null
Χρονική περίοδος	nvarchar(10)	22	-	Not Null

Περιγραφή	nvarchar(50)	102	-	Null
Συνολικό μέγεθος		170		

Πίνακας14: Χρωματικός Συνδυασμός

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Κωδικός χρώματος	nvarchar(10)	22	Primary	Not Null
Χρώμα	nvarchar(50)	102	-	Not Null
Συνολικό μέγεθος		124		

Πίνακας15: Λιανικές Παραγγελίες

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Αριθμός Παραγγελίας	nvarchar(20)	42	Primary	Not null
Κωδικός πελάτη	int	4	-	Not null
Ημερ. Παραγγελίας	datetime	8	-	Not Null
Προθεσμία	date	3	-	Null
Κωδικός Προϊόντος	nvarchar(10)	22	-	Not Null
Ποσότητα	int	4	-	Not Null
Τιμή ανά τεμάχιο	money	8	-	Not Null
Μερικό Σύνολο	money	8	-	Not Null
ΦΠΑ	money	8	-	Not Null
Έκπτωση (%)	decimal(5, 2)	5		Null
Έκπτωση	money	8		Null
Σύνολο	money	8		Not Null
Σημειώσεις	nvarchar(100)	202		Null
Κατάσταση Παραγγελίας	int	4		Not Null
Συνολικό μέγεθος		334		

Πίνακας16: Λιανικές Πωλήσεις

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Αριθμός Τιμολογίου	nvarchar(9)	20	Primary	Not null
Αριθμός Παραγγελίας	nvarchar(20)	42	-	Not null
Τρόπος πληρωμής	nvarchar(20)	42	-	Not Null
Ημερομηνία πληρωμής	datetime	8	-	Not Null
ΦΠΑ	money	8	-	Not Null
Μερικό Σύνολο	money	8	-	Not Null
Έκπτωση (%)	decimal(5, 2)	5	-	Null
Έκπτωση	money	8	-	Null
Ολικό σύνολο	money	8	-	Not Null
Συνολικό μέγεθος		149		

Πίνακας17: Χονδρικές Παραγγελίες

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Αριθμός παραγγελίας	nvarchar(20)	42	Primary	Not null

Κωδικός εταιρείας	int	4	-	Not null
Ημερ. παραγγελίας	datetime	8	-	NotNull
Προθεσμία	date	3	-	Null
Μερικό Σύνολο	money	8	-	NotNull
ΦΠΑ	money	8	-	NotNull
Ολικό Σύνολο	money	8	-	Not Null
Έκπτωση (%)	decimal(5, 2)	5	-	Null
Έκπτωση	money	8	-	Null
Νέο ολικό	money	8	-	NotNull
Προκαταβολή (%)	decimal(5, 2)	5	-	Null
Προκαταβολή	money	8	-	Null
Σημειώσεις	nvarchar(100)	202	-	Null
Κατάσταση παραγγελίας	int	4	-	Not Null
Συνολικό μέγεθος		321		

Πίνακας18: Λεπτομέρειες Χονδρικής Παραγγελίας

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
A/A	int	4	Primary	Not null
Αριθμός παραγγελίας	nvarchar(20)	42	-	Not null
Κωδικός είδους ένδυσης	nvarchar(10)	22	-	NotNull
Ποσότητα	int	4	-	NotNull
Τιμή ανά τεμάχιο	money	8	-	NotNull
Μερικό Σύνολο	money	8	-	NotNull
Συνολικό μέγεθος		88		

Πίνακας19: Χονδρικές Πωλήσεις

Όνομα πεδίου	Τύπος	Μέγεθος	Κλειδί	Null / Not Null
Αριθμός Τιμολογίου	nvarchar(9)	20	Primary	Not null
Αριθμός Παραγγελίας	nvarchar(20)	42	-	Not null
Ημερομηνία πληρωμής	date	8	-	Not Null
Τρόπος πληρωμής	nvarchar(20)	42	-	Null
Ολικό σύνολο	money	8	-	Not Null
Πληρωθέν ποσό	money	8	-	Not Null
Υπόλοιπο	money	8	-	Not Null
Λόγος πληρωμής	nvarchar(20)	42	-	Null
Συνολικό μέγεθος		209		

3.2.7. Ορισμός προδιαγραφών για είσοδο/έξοδο δεδομένων

Η δημιουργία ενός απλού, εύχρηστου και φιλικού περιβάλλοντος είναι ένας από τους πρωταρχικούς στόχους για την ανάπτυξη ενός λογισμικού συστήματος. Έχοντας αυτό υπ' όψη, γίνεται ανάπτυξη της διεπιφάνειας χρηστών βασισμένη στις προδιαγραφές και λειτουργίες του συστήματός μας. Παρακάτω παρουσιάζονται κάποιες από τις οθόνες διεπιφάνειας χρηστών για είσοδο και έξοδο δεδομένων από το λογισμικό σύστημα.

3.2.7.1. Φόρμα Διαχείρισης Πελατών

3.2.7.1.1. Προσθήκη πελάτη

Περιγραφή	Ο χρήστης μπορεί να εισάγει νέο πελάτη στο σύστημα δίνοντας τα στοιχεία του.
Είσοδος	Εισάγονται όλα τα πιο κάτω: <u>Περίπτωση πελάτη-ατόμου</u> Κωδικός πελάτη, Ονοματεπώνυμο, Ημερ. Γενεθλίων, Ημερ. Γιορτής, Οδός, Αριθμός, Περιοχή, Ταχ. Κώδικας, Πόλη, Χώρα, Τηλέφωνο, Κινητό, Email, Στήθος, Μέση, Γλουτοί <u>Περίπτωση πελάτη-εταιρείας</u> Κωδικός πελάτη, Όνομα εταιρείας, Όνομα μπουτίκ, Άτομο επικοινωνίας, Οδός, Αριθμός, Περιοχή, Ταχ. Κώδικας, Πόλη, Χώρα, Τηλέφωνο, Κινητό, Email, Φαξ, Website
Έξοδος	Αν τα στοιχεία εισάχθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.1.2. Αναζήτηση πελάτη

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να αναζητήσει ένα υπάρχοντα πελάτη στο σύστημα.
Είσοδος	Μπορεί να εισάγει ένα οποιοδήποτε από τα πιο πάνω στοιχεία.
Έξοδος	Αποτελέσματα αναζήτησης

3.2.7.1.3. Ενημέρωση στοιχείων πελάτη

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να τροποποιήσει τα στοιχεία ενός υπάρχοντα πελάτη.
Είσοδος	Μπορεί να εισάγει μερικά από (ή όλα) τα πιο πάνω στοιχεία.
Έξοδος	Αν τα στοιχεία τροποποιήθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή τροποποίηση και αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.1.4. Διαγραφή πελάτη

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να διαγράψει όλα τα στοιχεία ενός υπάρχοντα πελάτη.
Είσοδος	Τα στοιχεία του πελάτη βρίσκονται ήδη στην φόρμα.
Έξοδος	Αν τα στοιχεία διαγράφηκαν με επιτυχία, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή διαγραφή των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.2. Φόρμα Διαχείρισης πρώτων υλών

3.2.7.2.1. Προσθήκη πρώτης ύλης

Περιγραφή	Ο χρήστης μπορεί να εισάγει νέα πρώτη ύλη στο σύστημα, δίνοντας τα στοιχεία της.
Είσοδος	Εισάγονται όλα τα πιο κάτω: Κωδικός ύλης, Περιγραφή, Σύσταση, Χρώμα, Όνομα προμηθευτή, Κόστος ύλης, Αρχική ποσότητα, Ελάχιστη ποσότητα, Απόθεμα, Μονάδα μέτρησης
Έξοδος	Αν τα στοιχεία εισάχθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.2.2. Αναζήτηση πρώτης ύλης

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να αναζητήσει μια υπάρχουσα πρώτη ύλη στο σύστημα.
Είσοδος	Μπορεί να εισάγει ένα οποιοδήποτε από τα πιο πάνω στοιχεία.
Έξοδος	Αποτελέσματα αναζήτησης

3.2.7.2.3. Ενημέρωση στοιχείων πρώτης ύλης

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να τροποποιήσει τα στοιχεία μιας υπάρχουσας πρώτης ύλης.
Είσοδος	Μπορεί να εισάγει μερικά από (ή όλα) τα πιο πάνω στοιχεία.
Έξοδος	Αν τα στοιχεία τροποποιήθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή τροποποίηση και αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.2.4. Διαγραφή πρώτης ύλης

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να διαγράψει όλα τα στοιχεία μιας υπάρχουσας πρώτης ύλης.
Είσοδος	Τα στοιχεία της πρώτης ύλης βρίσκονται ήδη στην φόρμα.
Έξοδος	Αν τα στοιχεία διαγράφηκαν με επιτυχία, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή διαγραφή των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.2.5. Προσθήκη προμηθευτή

Περιγραφή	Ο χρήστης μπορεί να εισάγει νέο προμηθευτή στο σύστημα, δίνοντας τα στοιχεία του.
Είσοδος	Εισάγονται όλα τα πιο κάτω: Κωδικός προμηθευτή, Όνομα προμηθευτή, Άτομο επικοινωνίας, Οδός, Αριθμός, Περιοχή, Ταχ. Κώδικας, Πόλη, Χώρα, Τηλέφωνο, Κινητό, Email, Φαξ
Έξοδος	Αν τα στοιχεία εισάχθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για

την επιτυχή αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.2.6. Αναζήτηση προμηθευτή

Περιγραφή | Ο χρήστης έχει τη δυνατότητα να αναζητήσει ένα υπάρχοντα προμηθευτή στο σύστημα.

Είσοδος | Μπορεί να εισάγει ένα οποιοδήποτε από τα πιο πάνω στοιχεία.

Έξοδος | Αποτελέσματα αναζήτησης

3.2.7.2.7. Ενημέρωση στοιχείων προμηθευτή

Περιγραφή | Ο χρήστης έχει τη δυνατότητα να τροποποιήσει τα στοιχεία ενός υπάρχοντα προμηθευτή.

Είσοδος | Μπορεί να εισάγει μερικά από (ή όλα) τα πιο πάνω στοιχεία.

Έξοδος | Αν τα στοιχεία τροποποιήθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή τροποποίηση και αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.2.8. Διαγραφή προμηθευτή

Περιγραφή | Ο χρήστης έχει τη δυνατότητα να διαγράψει όλα τα στοιχεία ενός υπάρχοντα προμηθευτή.

Είσοδος | Τα στοιχεία του προμηθευτή βρίσκονται ήδη στην φόρμα.

Έξοδος | Αν τα στοιχεία διαγράφηκαν με επιτυχία, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή διαγραφή των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.3. Φόρμα Διαχείρισης προϊόντων

3.2.7.3.1. Προσθήκη προϊόντος

Περιγραφή | Ο χρήστης μπορεί να εισάγει νέο προϊόν στο σύστημα, δίνοντας τα στοιχεία της.

Είσοδος | Εισάγονται όλα τα πιο κάτω:
Κωδικός προϊόντος, Περιγραφή, Μέγεθος, Διαθέσιμα χρώματα, Εικόνα, Κατηγορία προϊόντος, Συλλογή προϊόντος, Πρώτες ύλες, Ποσότητα, Χονδρική Τιμή, Λιανική Τιμή, Κόστος κατασκευής

Έξοδος | Αν τα στοιχεία εισάχθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.3.2. Αναζήτηση προϊόντος

Περιγραφή | Ο χρήστης έχει τη δυνατότητα να αναζητήσει ένα υπάρχον προϊόν στο σύστημα.

Είσοδος | Μπορεί να εισάγει ένα οποιοδήποτε από τα πιο πάνω στοιχεία.

Έξοδος | Αποτελέσματα αναζήτησης

3.2.7.3.3. Ενημέρωση στοιχείων προϊόντος

Περιγραφή | Ο χρήστης έχει τη δυνατότητα να τροποποιήσει τα στοιχεία ενός υπάρχοντος προϊόντος.

Είσοδος | Μπορεί να εισάγει μερικά από (ή όλα) τα πιο πάνω στοιχεία.

Έξοδος | Αν τα στοιχεία τροποποιήθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή τροποποίηση και αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.3.4. Δημιουργία συλλογής

Περιγραφή | Ο χρήστης μπορεί να δημιουργήσει μια νέα συλλογή προϊόντων, δίνοντας τα στοιχεία της.

Είσοδος | Εισάγονται όλα τα πιο κάτω:
Κωδικός συλλογής, Όνομα συλλογής, Χρονική Περίοδος

Έξοδος | Αν τα στοιχεία εισάχθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.4. Φόρμα Διαχείρισης ταμειακής

3.2.7.4.1. Εισαγωγή νέας παραγγελίας

Περιγραφή | Ο χρήστης μπορεί να εισάγει μια νέα παραγγελία στο σύστημα, δίνοντας τα στοιχεία της.

Είσοδος | Εισάγονται όλα τα πιο κάτω:
Αριθμός παραγγελίας, Κωδικός πελάτη/εταιρείας, Ημερομηνία παραγγελίας, Προθεσμία, Μερικό Σύνολο, ΦΠΑ, Ολικό Σύνολο, Έκπτωση (%), Έκπτωση, Νέο ολικό, Προκαταβολή (%), Προκαταβολή, Σημειώσεις, Κατάσταση παραγγελίας, Προϊόν, Ποσότητα, Τιμή ανά τεμάχιο, Μερικό Σύνολο

Έξοδος | Αν τα στοιχεία εισάχθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.4.2. Αναζήτηση παραγγελίας

Περιγραφή | Ο χρήστης έχει τη δυνατότητα να αναζητήσει μια υπάρχουσα παραγγελία στο σύστημα.

Είσοδος | Μπορεί να εισάγει ένα οποιοδήποτε από τα πιο πάνω στοιχεία.

Έξοδος | Αποτελέσματα αναζήτησης

3.2.7.4.3. Ενημέρωση στοιχείων παραγγελίας

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να τροποποιήσει τα στοιχεία μιας υπάρχουσας παραγγελίας.
Είσοδος	Μπορεί να εισάγει μερικά από (ή όλα) τα πιο πάνω στοιχεία.
Έξοδος	Αν τα στοιχεία τροποποιήθηκαν σωστά, τότε εμφανίζεται ένα μήνυμα που ενημερώνει για την επιτυχή τροποποίηση και αποθήκευση των στοιχείων. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.4.4. Εκτύπωση τιμολογίου

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να εκτυπώσει το τιμολόγιο για μία παραγγελία.
Είσοδος	Τα παρακάτω στοιχεία του τιμολογίου βρίσκονται ήδη στην φόρμα: Αριθμός Παραγγελίας, Ονοματεπώνυμο πελάτη, Ημερομηνία έκδοσης, Προθεσμία, Κωδικός προϊόντος, Ποσότητα, Τιμή ανά τεμάχιο, Μερικό Σύνολο, ΦΠΑ, Έκπτωση (%), Έκπτωση, Ολικό Σύνολο, Σημειώσεις
Έξοδος	Το τιμολόγιο εκτυπώνεται και εμφανίζεται ανάλογο μήνυμα στην οθόνη. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.7.4.5. Εκτύπωση απόδειξης

Περιγραφή	Ο χρήστης έχει τη δυνατότητα να εκτυπώσει την απόδειξη για μία εξοφλημένη παραγγελία.
Είσοδος	Τα παρακάτω στοιχεία της απόδειξης βρίσκονται ήδη στην φόρμα: Κωδικός απόδειξης, Ημερομηνία πληρωμής, Τρόπος πληρωμής, Πληρωθέν ποσό, Λόγος πληρωμής
Έξοδος	Η απόδειξη εκτυπώνεται και εμφανίζεται ανάλογο μήνυμα στην οθόνη. Διαφορετικά, εμφανίζεται ένα μήνυμα αποτυχίας.

3.2.8. Εκτίμηση μεγέθους συστήματος

Στο στάδιο αυτό υπολογίζεται ο όγκος δεδομένων του κάθε πίνακα καθώς και ο όγκος δεδομένων εισόδου την πρώτη φορά που θα λειτουργήσει το σύστημα.

Πιο κάτω παρουσιάζονται τα βασικά μεγέθη που αφορούν τα δεδομένα της βάσης, όπως αυτά υπολογίστηκαν στο στάδιο 6 (παράγραφος 3.2.6), ούτως ώστε να υπάρχει μια ένδειξη ποσότητας μνήμης που θα χρησιμοποιεί η βάση δεδομένων.

A/A	Όνομα Πίνακα	Μνήμη ανά εγγραφή (bytes)
1	Ρυθμίσεις παραμέτρων	1422
2	Ονομαστικές εορτές	549

3	Κατάσταση παραγγελίας	106
4	Πελάτης	652
5	Συνεργαζόμενη εταιρεία	942
6	Προμηθευτής	738
7	Πρώτες ύλες	288
8	Προϊόν	264
9	Στοιχεία Κατασκευής Προϊόντος	48
10	Προϊόντα σε συλλογές	272
11	Κατηγορίες ειδών ένδυσης	106
12	Στοιχεία Κατασκευής Είδους ένδυσης	48
13	Συλλογή προϊόντων	170
14	Χρωματικός Συνδυασμός	124
15	Λιανικές παραγγελίες	334
16	Λιανικές πωλήσεις	149
17	Χονδρικές παραγγελίες	321
18	Λεπτομέρειες Χονδρικής Παραγγελίας	88
19	Χονδρικές πωλήσεις	209
Συνολικό μέγεθος		6830

Από τον πιο πάνω πίνακα φαίνεται ότι το συνολικό μέγεθος των δεδομένων για μια εγγραφή σε κάθε αρχείο είναι περίπου 6,83KB.

3.2.9. Απαιτήσεις σε hardware

3.2.9.1. Αποθηκευτικές μονάδες

Το σύστημα που θα αναπτυχθεί υλοποιεί μια desktop εφαρμογή που διαχειρίζεται μια βάση δεδομένων. Είναι απαραίτητη η χρήση σκληρού δίσκου για την αποθήκευση και ανάκτηση αυτών των δεδομένων. Το σύστημα θα διατηρεί και backup της βάσης δεδομένων. Για το σύστημα αυτό είναι απαραίτητο ένα desktop PC με δίσκο ελάχιστης χωρητικότητας 3 GB για την εισαγωγή των αρχικών δεδομένων, την εγκατάσταση του προγράμματος που αναπτύσσουμε και τον Microsoft SQL Server 2005.

3.2.9.2. Συσκευές εισόδου δεδομένων

Οι συσκευές εισόδου του συστήματος θα είναι το πληκτρολόγιο και το ποντίκι, τα οποία θα χρησιμοποιούνται για την εισαγωγή δεδομένων που αφορούν τους πελάτες, καθώς επίσης και για πλοήγηση στην επιφάνεια εργασίας του προγράμματος.

3.2.9.3. Συσκευές εξόδου δεδομένων

Οι συσκευές εξόδου του συστήματος θα είναι η οθόνη για την παρουσίαση των δεδομένων, ηχεία και ο εκτυπωτής για την εκτύπωση τιμολογίων, αποδείξεων και άλλων στοιχείων που αφορούν την επιχείρηση.

3.2.9.4. Επιπλέον εξοπλισμός

Είναι πιθανόν να παρουσιαστεί η ανάγκη εγκατάστασης ενός επιπλέον σκληρού δίσκου (εσωτερικού ή εξωτερικού) για την φύλαξη των backup αρχείων της βάσης. Με τον τρόπο αυτό θα

εξασφαλιστεί μεγαλύτερη ασφάλεια στο σύστημα, αφού θα είναι σε θέση να ανακτήσει τα δεδομένα του ακόμα και μετά την καταστροφή του δίσκου που φυλάσσει το κύριο αρχείο της βάσης δεδομένων.

3.3. ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΑΝΑΛΥΣΗ

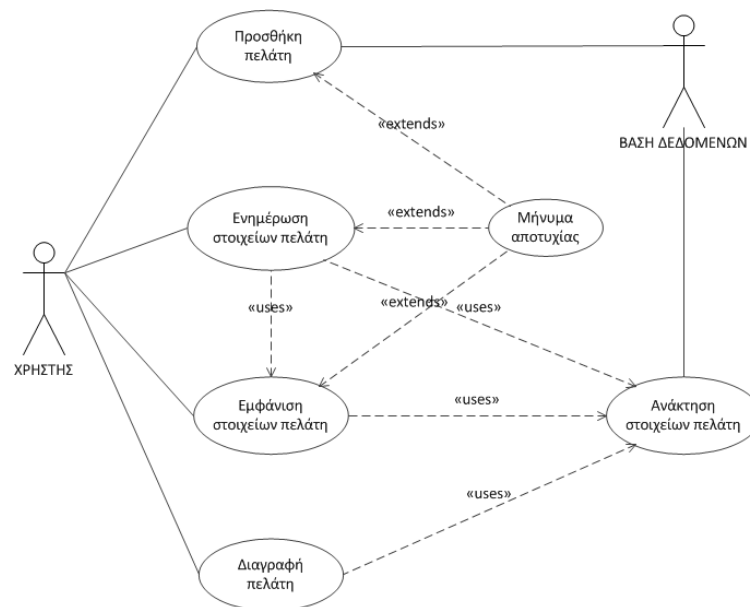
Για την κατανόηση, τον προσδιορισμό και την έκφραση των αναγκών του λογισμικού είναι απαραίτητο ένα ιδεατό μοντέλο των διεργασιών του συστήματος. Τα μοντέλα για την αντικειμενοστραφή ανάλυση χρησιμοποιούνται οι πιο κάτω τεχνικές παραστάσεις:

- Διαγράμματα Χρήσης (Usecase Diagrams)
- Διαγράμματα Κλάσεων (Analysis Class Diagrams)

3.3.1. Διαγράμματα Χρήσης

Ένα σύνολο σεναρίων που συνδέονται με ένα κοινό σκοπό (για τον χρήστη). Συνήθως περιλαμβάνει ένα πρωταρχικό σενάριο (σαν μια ακολουθία από βήματα) και τις εναλλακτικές περιπτώσεις (σαν παραλλαγές της κύριας ακολουθίας βημάτων).

3.3.1.1. Διαχείριση πελατών/συνεργαζόμενων εταιρειών

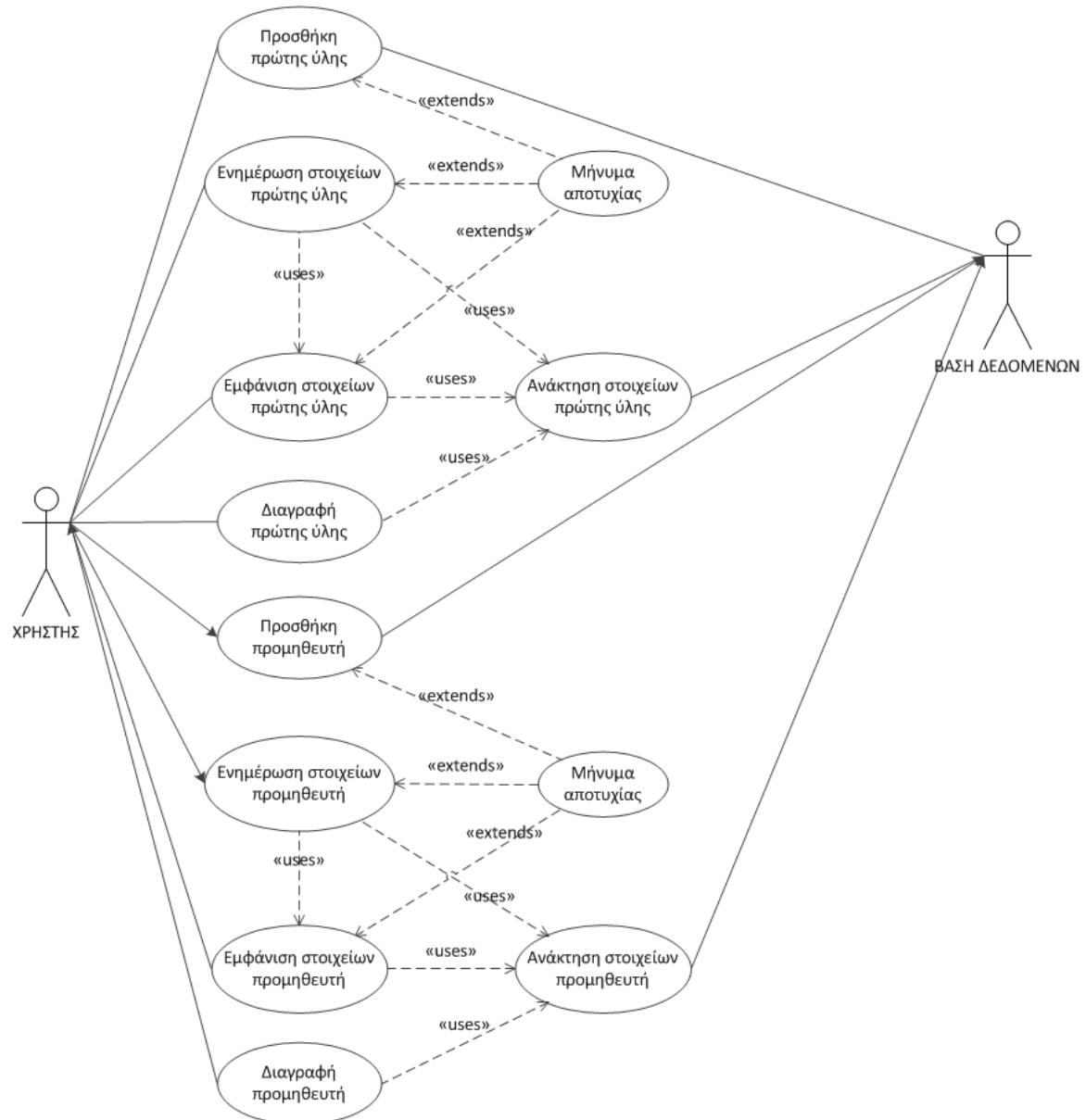


Διάγραμμα 3.23

Κατά τη διαχείριση πελατών, ο χρήστης μπορεί να:

- προσθέσει ένα νέο πελάτη: Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου πελάτη αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- ζητήσει τα στοιχεία κάποιου πελάτη: Αν δε βρεθούν τα στοιχεία του πελάτη, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται σε μορφή λίστας.
- ενημερώσει τα στοιχεία κάποιου πελάτη: Αν δε βρεθούν τα στοιχεία του πελάτη, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται και ενημερώνονται από το χρήστη.
- διαγράψει κάποιο πελάτη: Όταν τα στοιχεία ανακτηθούν από τη βάση δεδομένων, ο πελάτης διαγράφεται.

3.3.1.2. Διαχείριση πρώτων υλών



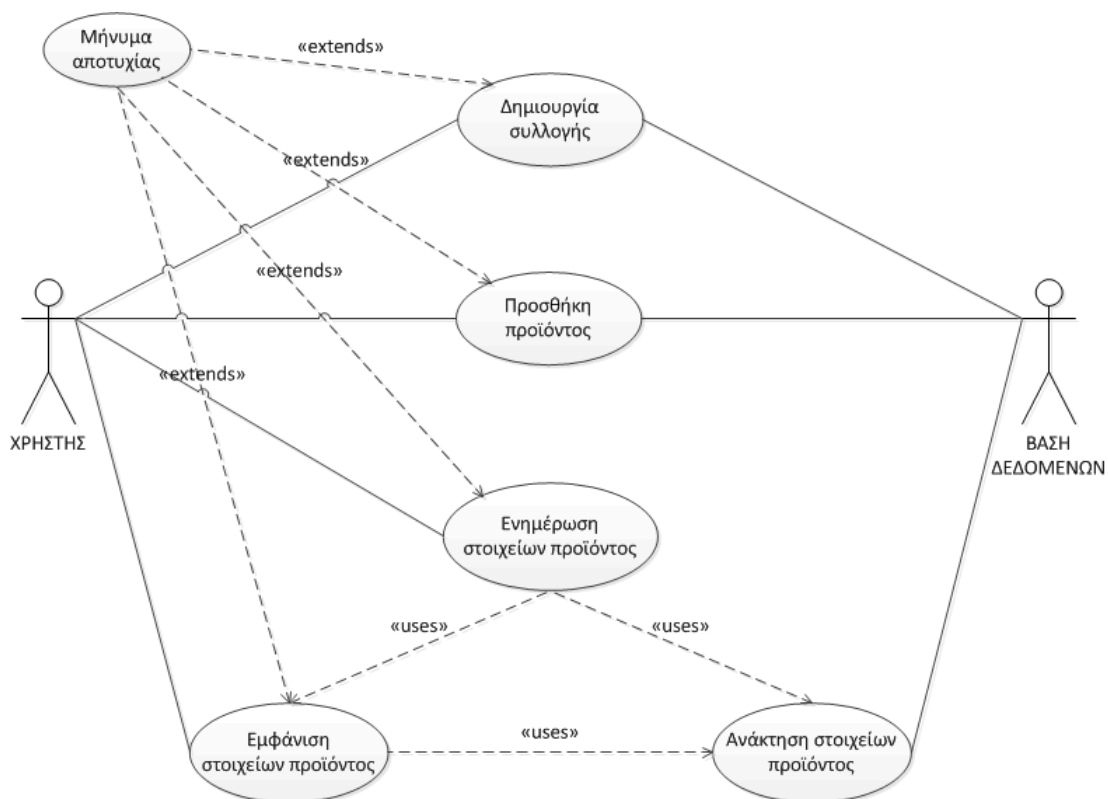
Διάγραμμα 3.24

Κατά τη διαχείριση πρώτων υλών, ο χρήστης μπορεί να:

- προσθέσει μια νέα πρώτη ύλη: Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της νέας πρώτης ύλης αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- ζητήσει τα στοιχεία κάποιας πρώτης ύλης: Αν δε βρεθούν τα στοιχεία της πρώτης ύλης, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται σε μορφή λίστας.
- ενημερώσει τα στοιχεία κάποιας πρώτης ύλης: Αν δε βρεθούν τα στοιχεία της πρώτης ύλης, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται και ενημερώνονται από το χρήστη.
- διαγράψει κάποιας πρώτης ύλης: Όταν τα στοιχεία ανακτηθούν από τη βάση δεδομένων, η πρώτη ύλη διαγράφεται.

- ε) προσθέσει ένα νέο προμηθευτή: Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προμηθευτή αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- στ) ζητήσει τα στοιχεία κάποιου προμηθευτή: Αν δε βρεθούν τα στοιχεία του προμηθευτή, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται σε μορφή λίστας.
- ζ) ενημερώσει τα στοιχεία κάποιου προμηθευτή: Αν δε βρεθούν τα στοιχεία του προμηθευτή, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται και ενημερώνονται από το χρήστη.
- η) διαγράψει κάποιο προμηθευτή: Όταν τα στοιχεία ανακτηθούν από τη βάση δεδομένων, ο προμηθευτής διαγράφεται.

3.3.1.3. Διαχείριση προϊόντων

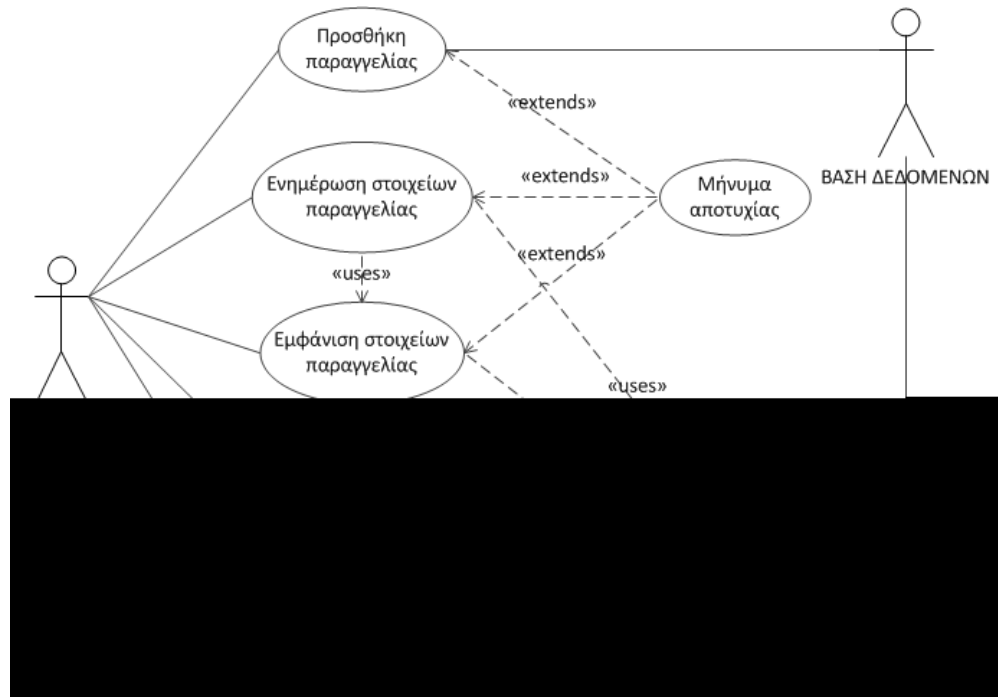


Διάγραμμα 3.25

Κατά τη διαχείριση προϊόντων, ο χρήστης μπορεί να:

- α) προσθέσει ένα νέο προϊόν: Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία του νέου προϊόντος αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- β) ζητήσει τα στοιχεία κάποιου προϊόντος: Αν δε βρεθούν τα στοιχεία του προϊόντος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται σε μορφή λίστας.
- γ) ενημερώσει τα στοιχεία κάποιου προϊόντος: Αν δε βρεθούν τα στοιχεία του προϊόντος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται και ενημερώνονται από το χρήστη.
- δ) δημιουργήσει μια συλλογή προϊόντων: Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της συλλογής αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.

3.3.1.4. Διαχείριση ταμειακής

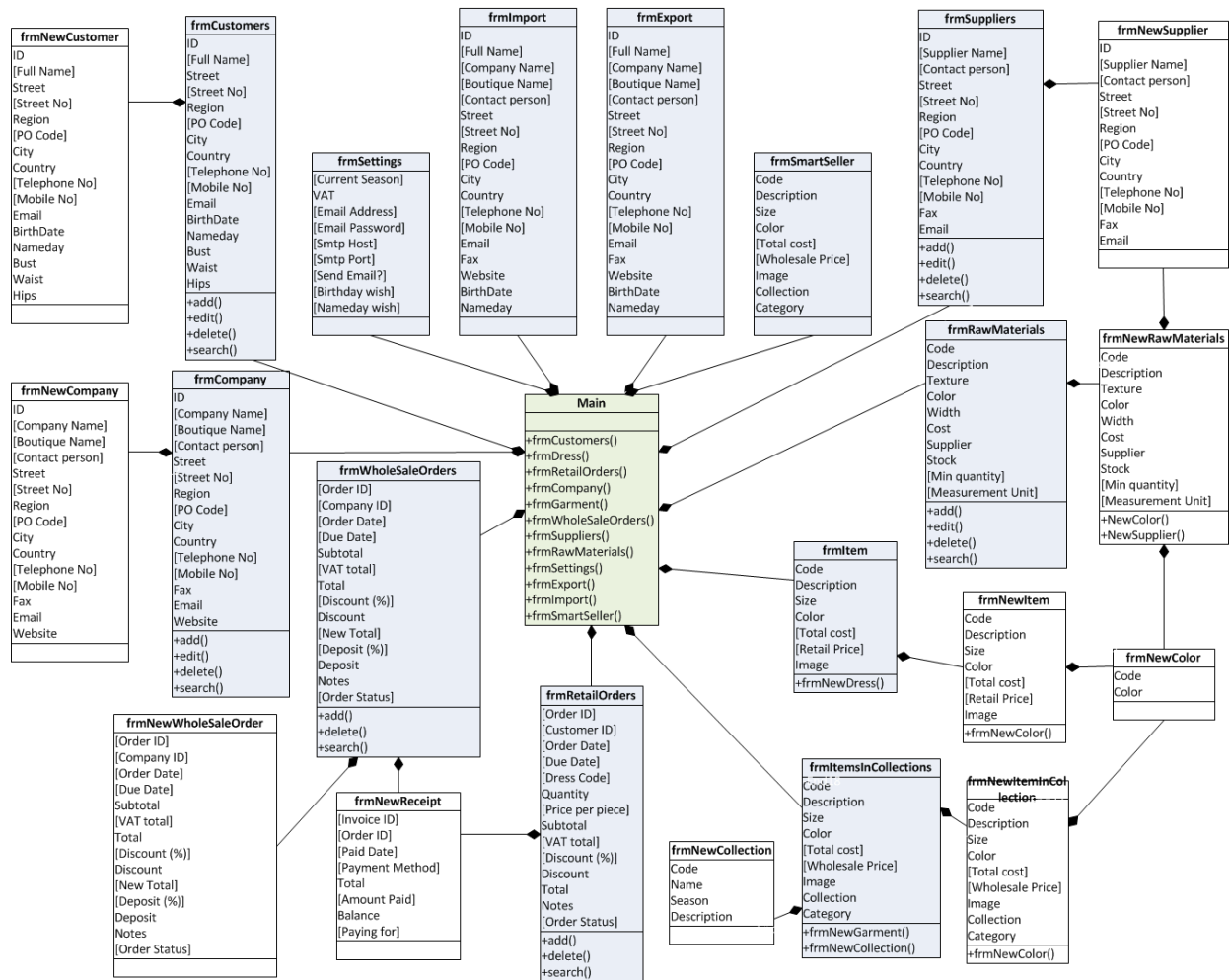


Διάγραμμα 3.26

Κατά τη διαχείριση ταμειακής, ο χρήστης μπορεί να:

- α) εισάγει μια νέα παραγγελία: Σε περίπτωση σφάλματος, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά τα στοιχεία της νέας παραγγελίας αποθηκεύονται στη βάση δεδομένων και ο χρήστης λαμβάνει μήνυμα επιτυχίας.
- β) ζητήσει τα στοιχεία μιας παραγγελίας: Αν δε βρεθεί παραγγελία, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται από τη βάση δεδομένων.
- γ) ενημερώσει τα στοιχεία μιας παραγγελίας: Αν δε βρεθεί παραγγελία, ο χρήστης λαμβάνει μήνυμα αποτυχίας. Διαφορετικά, τα στοιχεία ανακτώνται και ενημερώνονται από το χρήστη.
- δ) εκτυπώσει τιμολόγιο: Όταν τα στοιχεία της παραγγελίας ανακτηθούν από τη βάση δεδομένων, ο χρήστης εκτυπώνεται το τιμολόγιο.
- ε) εκτυπώσει απόδειξη: Όταν τα στοιχεία της παραγγελίας ανακτηθούν από τη βάση δεδομένων, ο χρήστης εκτυπώνεται την απόδειξη.

3.3.2. Διάγραμμα Κλάσεων



Η Main (κλάση με πράσινο χρώμα) είναι η κλάση που ελέγχει όλες τις υπόλοιπες κλάσεις του συστήματος. Από εδώ μπορεί να γίνει κλήση στις βασικές κλάσεις του συστήματος (κλάσεις με γαλάζιο χρώμα).

Μέσω της κλάσης `frmCustomers` μπορεί να εισαχθεί, ενημερωθεί, διαγραφεί ή αναζητηθεί ένας πελάτης. Ομοίως, μέσω των κλάσεων `frmCompany`, `frmSuppliers`, `frmItems`, `frmItemsInCollections`, `frmRawMaterials` μπορούν να εισαχθούν, ενημερωθούν, διαγραφούν ή αναζητηθούν συνεργαζόμενες εταιρείες, προμηθευτές, προϊόντα, προϊόντα σε συλλογές και πρώτες ύλες αντίστοιχα.

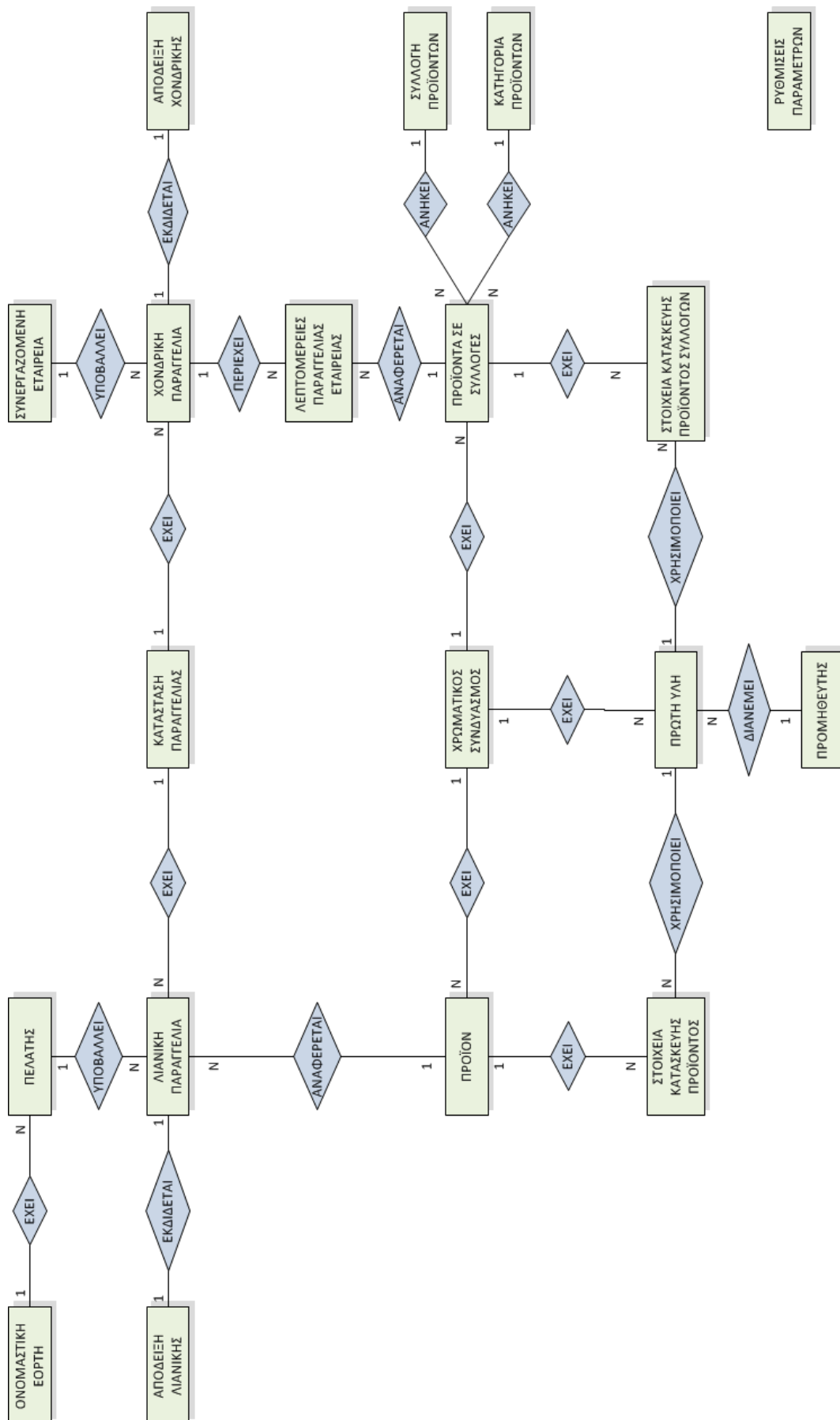
Ο χρήστης μπορεί να δημιουργήσει, διαγράψει ή αναζητήσει μια λιανική ή χονδρική παραγγελία, χρησιμοποιώντας τις κλάσεις `frmRetailOrders` και `frmWholeSaleOrders`.

Μέσω της κλάσης `frmImport` και `frmExport` γίνεται εισαγωγή από αρχείο και εξαγωγή σε αρχείο των δεδομένων των πελατών και των συνεργαζόμενων εταιρειών.

Το σύστημα προσφέρει την ευχέρεια για σύνθετη αναζήτηση ειδών ένδυσης μέσω της κλάσης `frmSmartSeller` όπου ο χρήστης μπορεί να καθορίσει εξειδικευμένα κριτήρια για να αναζητήσει προϊόντα.

Ο χρήστης μπορεί να καθορίσει τις διάφορες παραμέτρους του συστήματος μέσω της κλάσης `frmSettings`. Μπορεί να καθορίσει παραμέτρους που αφορούν το δείκτη του ΦΠΑ, το ηλεκτρονικό ταχυδρομείο και την τρέχουσα χρονική περίοδο.

3.4. Διάγραμμα Οντοτήτων - Συσχετίσεων



Στο πιο πάνω διάγραμμα οντοτήτων - συσχετίσεων φαίνονται όλες οι σχέσεις που διέπουν τους πίνακες που απαρτίζουν την βάση δεδομένων του συστήματος.

Στη βάση δεδομένων γίνεται ένας διαχωρισμός μεταξύ των πελατών της επιχείρησης, σε πελάτες-άτομα και πελάτες-εταιρείες. Επίσης γίνεται διαχωρισμός μεταξύ των παραγγελιών που υποβάλλουν στην επιχείρηση. Οι παραγγελίες των πελατών-ατόμων αναφέρονται ως λιανικές παραγγελίες με τις αντίστοιχες αποδείξεις λιανικής, ενώ οι παραγγελίες των πελατών-εταιρειών αναφέρονται ως χονδρικές παραγγελίες με τις αντίστοιχες αποδείξεις χονδρικής. Η κάθε παραγγελία (είτε χονδρική είτε λιανική) ανά πάσα στιγμή μπορεί να κατατάσσεται σε μόνο μία κατάσταση: απλήρωτη, εκδόθηκε τιμολόγιο, εξοφλημένη, ακυρωμένη.

Οποιοδήποτε προϊόν κατασκευάζεται στην επιχείρηση αποθηκεύεται μαζί με τα χαρακτηριστικά του, π.χ. χρώμα και υλικό κατασκευής, και τα κατασκευαστικά του υλικά στη βάση δεδομένων. Τα υλικά και η ποσότητα που θα χρησιμοποιηθούν, αφαιρούνται από τα αποθέματα των πρώτων υλών της αποθήκης. Όταν το απόθεμα μιας πρώτης ύλης φτάσει σε ένα προκαθορισμένο κατώτατο όριο, ο υπεύθυνος ειδοποιεί τον αντίστοιχο προμηθευτή για την εξαντληθείσα πρώτη ύλη.

Μια επιχείρηση συνεργάζεται και με διάφορες εταιρείες, τόσο εξωτερικές όσο και τοπικές. Οι συνεργαζόμενες εταιρείες υποβάλλουν κατά τακτά χρονικά διαστήματα χονδρικές παραγγελίες, οι οποίες αφορούν διάφορα προϊόντα από τις συλλογές προϊόντων που δημιουργούνται από την επιχείρηση. Οι συλλογές προϊόντων περιέχουν προϊόντα από διάφορες κατηγορίες προϊόντων.

Προς βελτίωση των σχέσεων μεταξύ επιχείρησης και πελατών, στη βάση δεδομένων φυλάσσονται πληροφορίες για τις (σταθερές και κινητές) ονομαστικές εορτές του τρέχοντος έτους. Οι πληροφορίες αυτές, ενημερώνονται κάθε πρώτη του νέου έτους.

Επίσης, οποιαδήποτε δεδομένα τα οποία είναι βοηθητικά για λειτουργίες και υπολογισμούς της επιχείρησης, φυλάσσονται στη βάση δεδομένων, έτσι ώστε να αποφεύγεται η επανάληψη και η συνεχής πληκτρολόγησή τους από τον χρήστη του συστήματος.

Τα χαρακτηριστικά των οντοτήτων φαίνονται στη παράγραφο 3.2.5 - *Εκλέπτυνση αποθήκευσης δεδομένων* (σελ. 43).

Κεφάλαιο 4. Σχεδίαση

5.1.	ΕΙΣΑΓΩΓΗ.....	66
5.2.	ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΧΕΔΙΑΣΗ	67
	5.2.1. Ανάλυση Ροών Δεδομένων	
	5.2.2. Διαγράμματα δομής	
5.3.	ΑΝΑΛΥΤΙΚΗ ΣΧΕΔΙΑΣΗ	84
	5.3.1. Οντότητες με όμοιες λειτουργίες	
	5.3.2. Οντότητες με επιπλέον ή διαφορετικές λειτουργίες	
5.4.	ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΧΡΗΣΤΗ	90

4.1. ΕΙΣΑΓΩΓΗ

Το έγγραφο της σχεδίασης έχει σκοπό να μειώνει τη συνολική πολυπλοκότητα του συστήματος και κατά συνέπεια το κόστος υλοποίησης και το ενδεχόμενο λαθών μέσα από την τμηματικότητα του σχεδίου του συστήματος. Μια καλή μέθοδος σχεδιασμού πρέπει να επιτρέπει:

1. την αποσύνθεση σε τμήματα (modulardecomposability)
2. τη σύνθεση τμημάτων (modular composability)
3. τη χωριστή κατανόηση τμημάτων (modularunderstandability)
4. τη συνέχεια των τμημάτων (modularcontinuity) σε αλλαγές: θα πρέπει να επηρεάζονται λίγα τμήματα
5. την προστασία των τμημάτων (modularprotection): λάθος σε ένα τμήμα δε θα πρέπει να δημιουργεί προβλήματα σε άλλα

Ο καλός σχεδιασμός πρέπει να φέρνει κοντά τμήματα που εμφανίζουν υψηλή συνεκτικότητα. Διακρίνονται τα παρακάτω αυξανόμενα επίπεδα συνεκτικότητας:

1. συμπτωματική συνεκτικότητα (coincidental cohesion)
2. λογική συνεκτικότητα (logical cohesion)
3. χρονική συνεκτικότητα (temporal cohesion)
4. διαδικαστική συνεκτικότητα (procedural cohesion)
5. επικοινωνιακή συνεκτικότητα (communicational cohesion)
6. ακολουθιακή συνεκτικότητα (sequential cohesion)
7. λειτουργική συνεκτικότητα (functional cohesion)
8. Σύζευξη (coupling)

Ο καλός σχεδιασμός πρέπει κατά το δυνατόν να αποφεύγει τη σύζευξη μεταξύ τμημάτων. Διακρίνονται τα παρακάτω αυξανόμενα επίπεδα σύζευξης:

1. σύζευξη δεδομένων (data coupling)
2. σύζευξη δομής δεδομένων (structcoupling) (ή αντιγράφου)
3. σύζευξη ελέγχου (control coupling)
4. σύζευξη κοινών δεδομένων (commoncoupling) (ή από κοινού σύνδεση)
5. σύζευξη εξωτερικών δεδομένων (externalcoupling)
6. σύζευξη περιεχομένων (content coupling)

Επίσης, το έγγραφο της σχεδίασης έχει σκοπό να παρουσιάσει την αρχιτεκτονική και αναλυτική σχεδίαση της κάθε διεργασίας του συστήματος.

Αποδέκτες του εγγράφου σχεδίασης είναι οι προγραμματιστές του συστήματος. Για αυτό στο έγγραφο αυτό, πρέπει να παρουσιάζεται η αναλυτική και αρχιτεκτονική σχεδίαση της κάθε διεργασίας και τμήματος του συστήματος με πλήρη λεπτομέρεια και σαφήνεια, έτσι ώστε οι προγραμματιστές να δημιουργήσουν ορθά την κάθε διεργασία και να διευκολυνθούν στην όλη υλοποίηση.

Στην αρχιτεκτονική σχεδίαση, με βάση τις προδιαγραφές που καθορίστηκαν από την φάση των προδιαγραφών, καθορίζονται τα τμήματα και οι διεργασίες, από τα οποία θα απαρτίζεται το σύστημα.

Στην αναλυτική σχεδίαση, γίνεται η σχεδίαση του κάθε τμήματος και της κάθε διεργασίας με λεπτομέρεια και σαφήνεια, έτσι ώστε να αποφευχθούν λάθη που μελλοντικά είναι βέβαιο πως θα καθυστερήσουν την υλοποίηση.

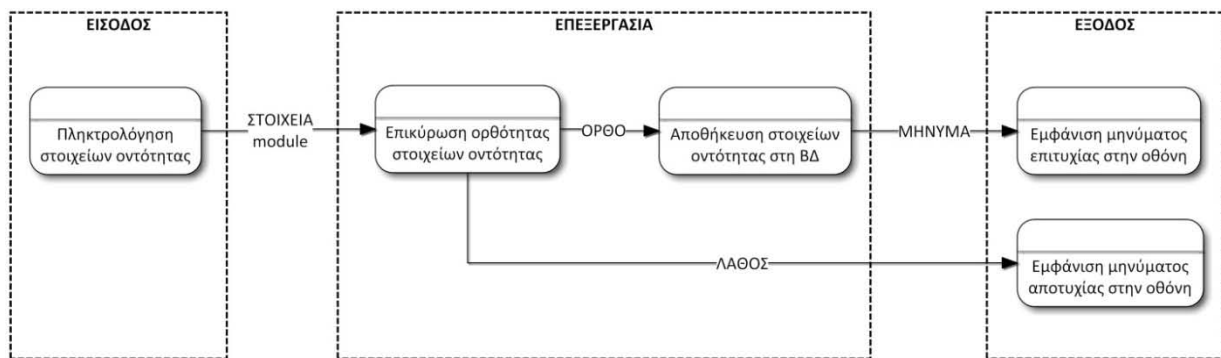
4.2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΧΕΔΙΑΣΗ

Κατά την αρχιτεκτονική σχεδίαση, γίνεται προσδιορισμός των μονάδων που αποτελούν το λογισμικό και πώς σχετίζονται μεταξύ τους. Στη αρχιτεκτονική σχεδίαση, η εργασία αυτή γίνεται λαμβάνοντας ως είσοδο τα διαγράμματα ροής δεδομένων. Σε αυτά εφαρμόζεται ένα σύνολο κανόνων και καθορίζονται οι μονάδες λογισμικού που είναι αναγκαίες για την υλοποίηση των μετασχηματισμών και των ροών δεδομένων.

4.2.1. Ανάλυση Ροών Δεδομένων

4.2.1.1. Οντότητες με όμοιες λειτουργίες²

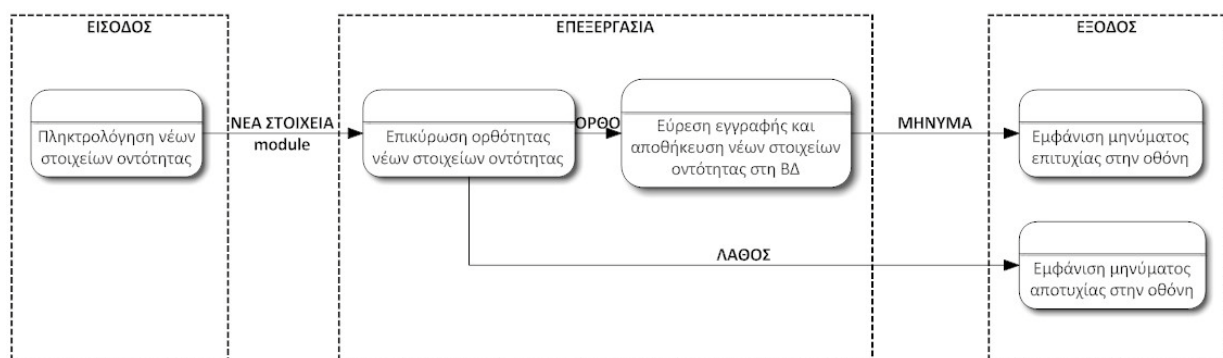
4.2.1.1.1. Προσθήκη στην οντότητα



Διάγραμμα 4.1

Το module *ΠΡΟΣΘΗΚΗ* είναι υπεύθυνο για την αποθήκευση των πληροφοριών που ο χρήστης συμπληρώνει σε μια φόρμα. Αφού ο χρήστης εισάγει τα στοιχεία, το module είναι υπεύθυνο να ελέγξει για την ορθότητά τους. Εάν τα επικυρώσει, τότε τα στοιχεία αποθηκεύονται στην βάση δεδομένων, ειδικά ελέγχει ένα μήνυμα λάθους.

4.2.1.1.2. Ενημέρωση στην οντότητα

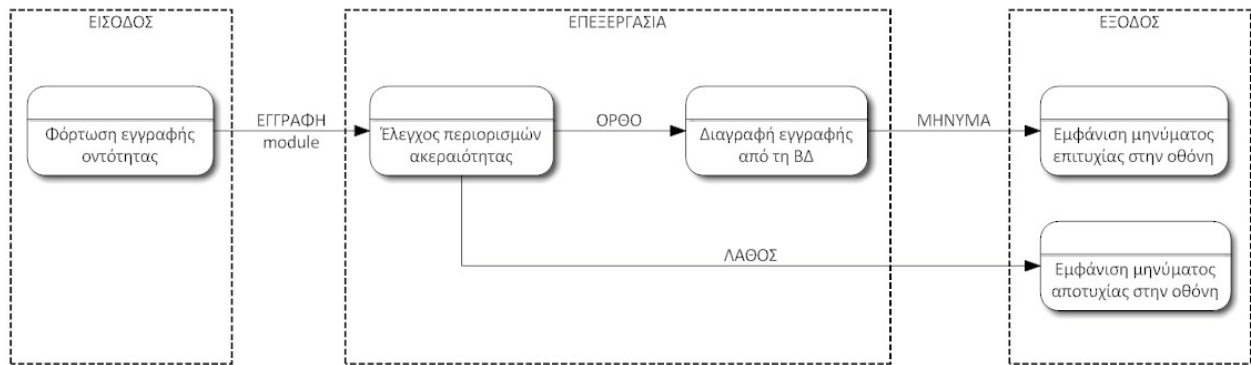


Διάγραμμα 4.2

Το module *ΕΝΗΜΕΡΩΣΗ* είναι υπεύθυνο για την ενημέρωση των πεδίων υπαρχόντων εγγραφών στη βάση δεδομένων. Αφού ο χρήστης εισάγει τα νέα στοιχεία, το module ελέγχει την ορθότητά τους. Εάν τα νέα στοιχεία είναι ορθά, αποθηκεύονται. Διαφορετικά εμφανίζεται μήνυμα λάθους.

² Λόγω επαναλήψεων στην αρχιτεκτονική ανάλυση των λειτουργιών, οι όμοιες λειτουργίες παρατίθενται μια φορά. Όπου αναφέρεται η λέξη *οντότητα* στην παράγραφο 4.2.1.1, εννοείται μία από τις οντότητες Πελάτης, Εταιρεία, Προμηθευτής, Προϊόν, Προϊόντα Συλλογών, Πρώτη Ύλη, Λιανική Πώληση, Χονδρική Πώληση, Χρώμα, Συλλογή.

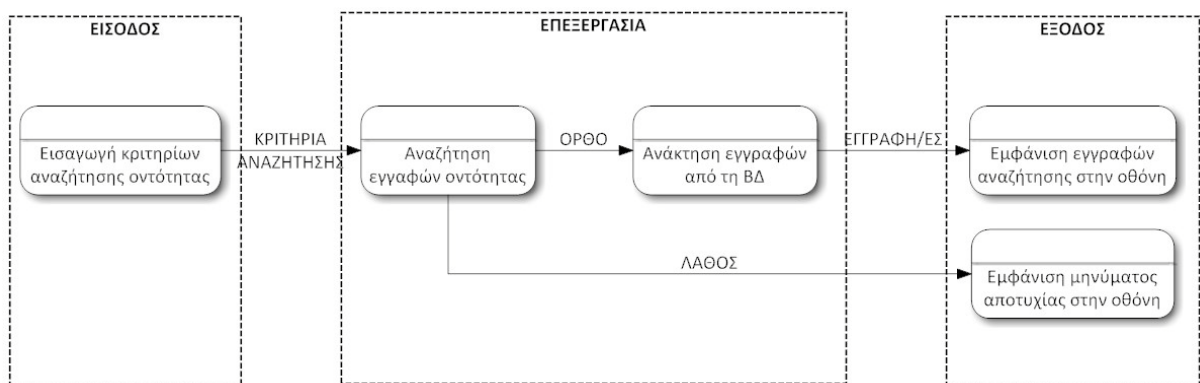
4.2.1.1.3. Διαγραφή από την οντότητα



Διάγραμμα 4.3

Το module **ΔΙΑΓΡΑΦΗ** είναι υπεύθυνο για την ενημέρωση των πεδίων υπαρχόντων εγγραφών στη βάση δεδομένων. Αφού ο χρήστης εισάγει τα νέα στοιχεία, το module ελέγχει την ορθότητά τους. Εάν τα νέα στοιχεία είναι ορθά, αποθηκεύονται. Διαφορετικά εμφανίζεται μήνυμα λάθους.

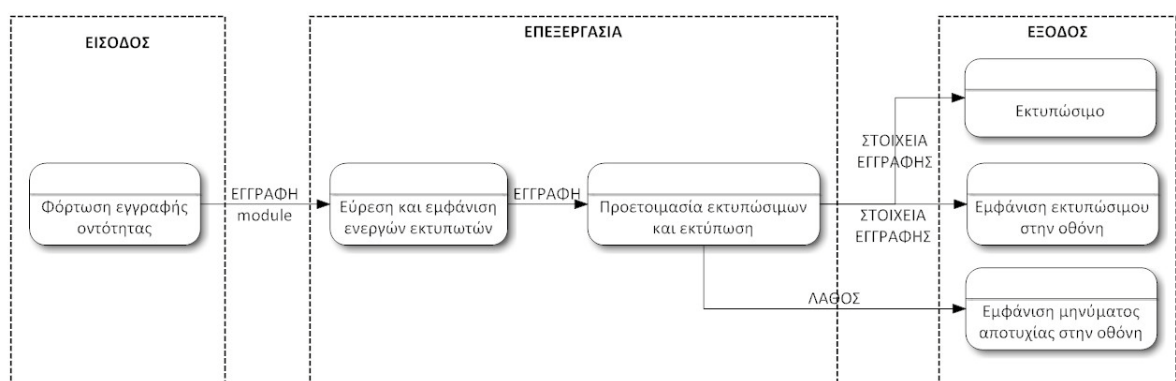
4.2.1.1.4. Αναζήτηση από την οντότητα



Διάγραμμα 4.4

Το module **ΑΝΑΖΗΤΗΣΗ** είναι υπεύθυνο να αναζητήσει εγγραφές από τη βάση δεδομένων οι οποίες ικανοποιούνται κριτήρια που έθεσε ο χρήστης. Σε περίπτωση που υπάρχει λάθος στα κριτήρια που τέθηκαν κατά την αναζήτηση ή δεν έχουν βρεθεί εγγραφές που ικανοποιούν τα κριτήρια, τότε εμφανίζεται μήνυμα λάθους στην οθόνη.

4.2.1.1.5. Εκτύπωση εγγραφής από την οντότητα



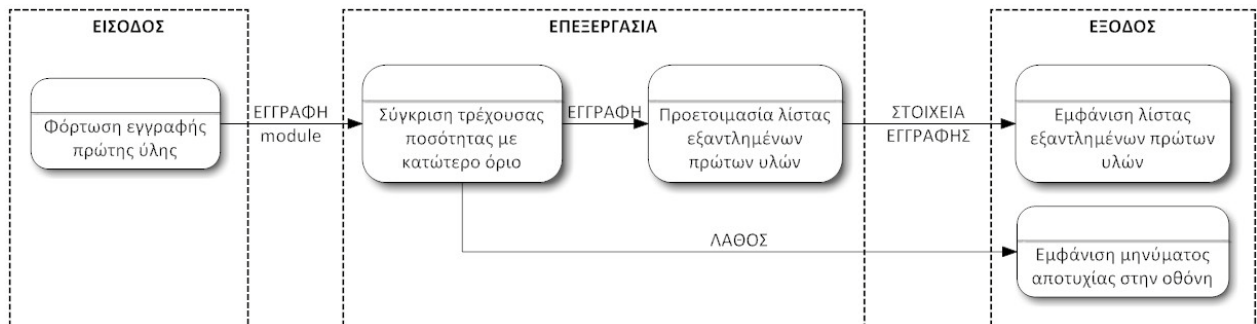
Διάγραμμα 4.5

Το module *ΕΚΤΥΠΩΣΗ* ανακτά μια εγγραφή από τη βάση δεδομένων και αφούβρει τους ενεργούς εκτυπωτές και ομαδοποιεί κατάλληλα τα πεδία της εγγραφής με ένα προκαθορισμένο τρόπο, τα εμφανίζει στην οθόνη σε τυποποιημένη μορφή η οποία είναι δυνατό να τυπωθεί. Σε περίπτωση αποτυχίας της διαδικασίας αυτής, εμφανίζεται μήνυμα λάθους.

Με τον όρο *εκτυπώσιμο* υπονοείται οτιδήποτε μπορεί να εκτυπωθεί από το λογισμικό στον εκτυπωτή. Περιλαμβάνονται τα τιμολόγια, οι αποδείξεις, οι αναφορές και οι εκθέσεις των εγγραφών της βάσης δεδομένων.

4.2.1.2. Οντότητες με επιπλέον ή διαφορετικές λειτουργίες

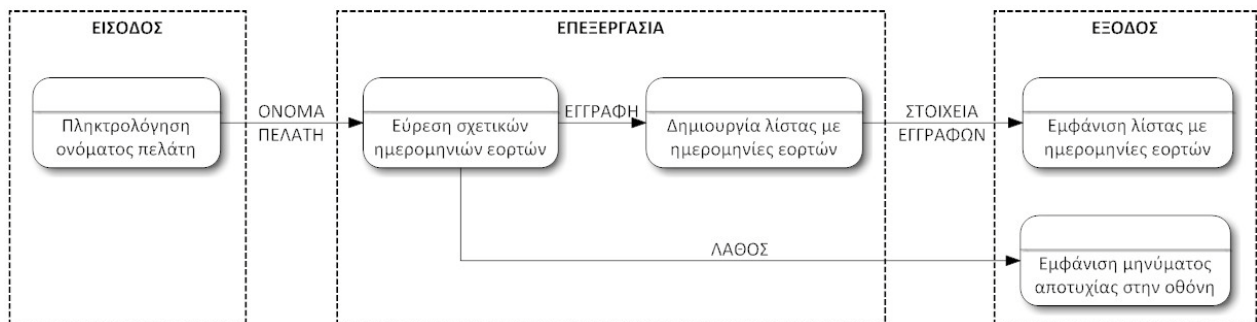
4.2.1.2.1. Αρχιτεκτονική Σχεδίαση Πρώτων Υλών: Αυτόματη Προειδοποίηση εξάντλησης πρώτης ύλης



Διάγραμμα 4.6

Το module *ΑΥΤΟΜΑΤΗ ΠΡΟΕΙΔΟΠΟΙΗΣΗ ΕΞΑΝΤΛΗΣΗΣ ΠΡΩΤΗΣ ΥΛΗΣ* φορτώνει από τη βάση δεδομένων τις εγγραφές των πρώτων υλών, συγκρίνει την τρέχουσα ποσότητα της πρώτης ύλης με το κατώτερο όριο παραγγελίας και αν αυτή είναι ίση ή μικρότερη του ορίου, δημιουργείται μια λίστα η οποία περιλαμβάνει τις πρώτες ύλες οι οποίες έχουν σχεδόν εξαντληθεί. Εάν για κάποιο λόγο αποτύχει αυτή η διαδικασία, εμφανίζεται μήνυμα λάθους.

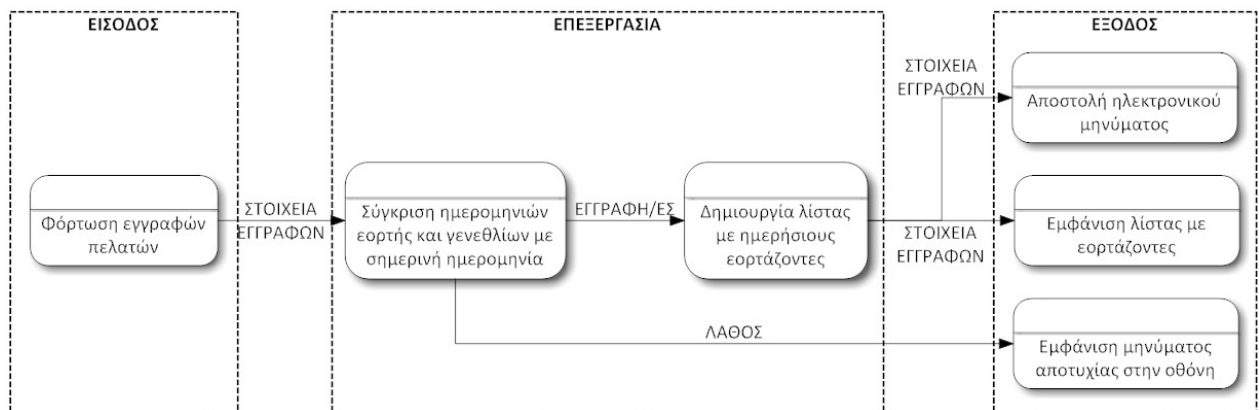
4.2.1.2.2. Αρχιτεκτονική Σχεδίαση Πελάτη: Αυτοματοποιημένη Εύρεση εορτής πελάτη



Διάγραμμα 4.7

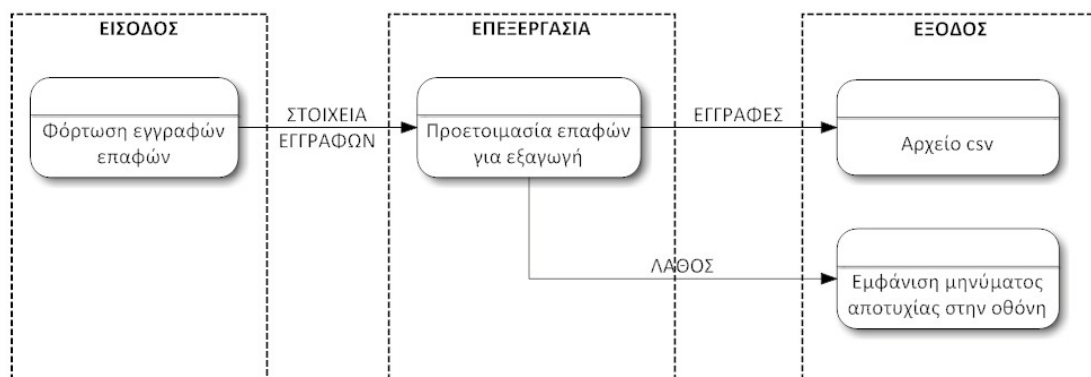
Το module *ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΕΥΡΕΣΗ ΕΟΡΤΗΣ ΠΕΛΑΤΗ* βρίσκει από τη βάση δεδομένων τις εγγραφές των εορτών που σχετίζονται με το όνομα που έχει πληκτρολογηθεί και δημιουργεί μια λίστα η οποία περιλαμβάνει όλες τις ευρεθείσες εορτές. Εάν για κάποιο λόγο αποτύχει αυτή η διαδικασία, εμφανίζεται μήνυμα λάθους.

4.2.1.2.3. Αρχιτεκτονική Σχεδίαση Πελάτη: Αυτοματοποιημένη Αποστολή Ευχετήριας κάρτας



Διάγραμμα 4.8

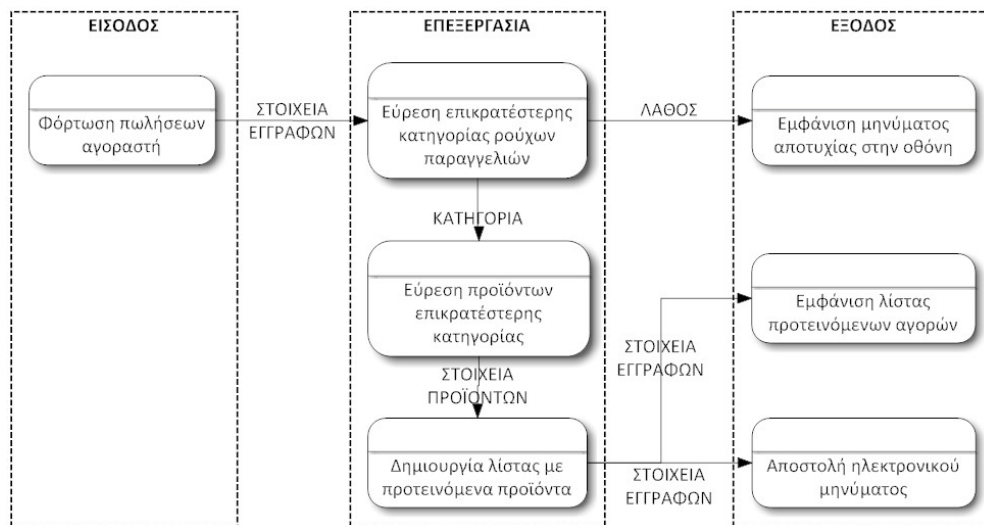
Το module *ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΑΠΟΣΤΟΛΗ ΕΥΧΕΤΗΡΙΑΣ ΚΑΡΤΑΣ* βρίσκει από τη βάση δεδομένων τις εγγραφές των πελατών που έχουν γιορτή ή γενέθλια την τρέχουσα ημέρα και δημιουργεί μια λίστα η οποία περιλαμβάνει όλους τους εορτάζοντες. Ακολουθώντας τους αποστέλλει με ηλεκτρονικό μήνυμα ευχετήρια κάρτα. Εάν για κάποιο λόγο αποτύχει η διαδικασία εύρεσης των εορτάζόντων, εμφανίζεται μήνυμα λάθους.

4.2.1.2.4. Αρχιτεκτονική Σχεδίαση Επαφών³: Φόρτωση επαφών σε κινητό τηλέφωνο

Διάγραμμα 4.9

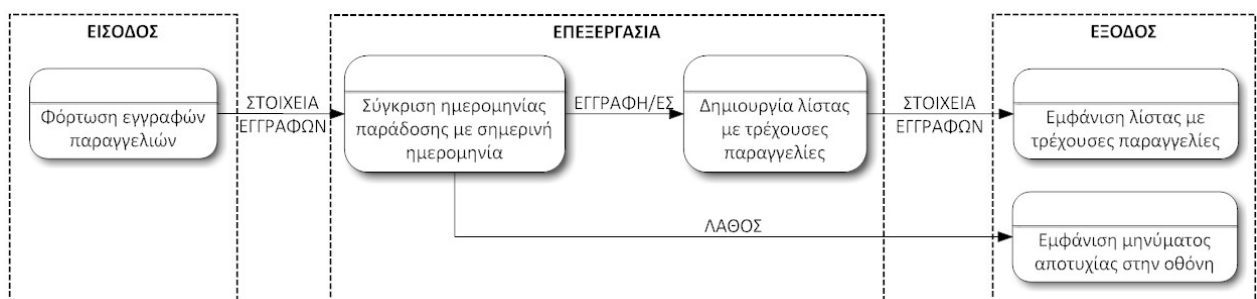
Το module *ΦΟΡΤΩΣΗ ΕΠΑΦΩΝ ΣΕ ΚΙΝΗΤΟ ΤΗΛΕΦΩΝΟ* βρίσκει από τη βάση δεδομένων τις εγγραφές των επαφών και τις εξάγει σε ένα αρχείο τιμών διαχωρισμένες με κόμμα (csvfile=comma-separatedvaluefile). Εάν για κάποιο λόγο αποτύχει η διαδικασία εξαγωγής των επαφών, εμφανίζεται μήνυμα λάθους.

³ Όπου αναφέρεται η λέξη *επαφή* στην παράγραφο 4.2.1.2, εννοείται μία από τις οντότητες Πελάτης και Εταιρεία.

4.2.1.2.5. Αρχιτεκτονική Σχεδίαση Αγοραστή⁴: Έξυπνες πωλήσεις

Διάγραμμα 4.10

Το module *ΕΞΥΠΝΕΣ ΠΩΛΗΣΕΙΣ* βρίσκει από τη βάση δεδομένων τις αγορές ενός αγοραστή και βάση της επικρατέστερης κατηγορίας των ενδυμάτων που παραγγέλθηκαν, δημιουργείται μια λίστα η οποία περιλαμβάνει ενδύματα που ανήκουν στην κατηγορία αυτή και ίσως θα ενδιέφεραν τον αγοραστή. Ακολούθως εμφανίζεται μια λίστα με τα προϊόντα η οποία αποστέλλεται με ηλεκτρονικό μήνυμα στον αγοραστή. Εάν για κάποιο λόγο αποτύχει η διαδικασία εύρεσης της επικρατέστερης κατηγορίας, εμφανίζεται μήνυμα λάθους.

4.2.1.2.6. Αρχιτεκτονική Σχεδίαση Παραγγελίες⁵: Τρέχουσες Παραγγελίες

Διάγραμμα 4.11

Το module *ΤΡΕΧΟΥΣΕΣ ΠΑΡΑΓΓΕΛΙΕΣ* βρίσκει από τη βάση δεδομένων όλες τις παραγγελίες, συγκρίνει την ημερομηνία αποπεράτωσης με την σημερινή ημερομηνία και αν η παραγγελία δεν έχει διεκπεραιωθεί ακόμη και εκκρεμεί, τοποθετείται σε μια λίστα η οποία περιλαμβάνει τις εκκρεμείς και τρέχουσες παραγγελίες. Ακολούθως εμφανίζεται η λίστα με τις παραγγελίες αυτές. Εάν για κάποιο λόγο αποτύχει η διαδικασία εύρεσης των παραγγελιών, εμφανίζεται μήνυμα λάθους.

⁴ Όπου αναφέρεται η λέξη *αγοραστής* στην παράγραφο 4.2.1.2, εννοείται μία από τις οντότητες Πελάτης και Εταιρεία.

⁵ Όπου αναφέρεται η λέξη *παραγγελίες* στην παράγραφο 4.2.1.2, εννοείται μία από τις οντότητες Λιανική Παραγγελία και Χονδρική Παραγγελία.

4.2.2. Διαγράμματα δομής

Ένα διάγραμμα δομής αποτελεί ένα ιεραρχικό διάγραμμα που απεικονίζει την αρχιτεκτονική δομή ενός προγράμματος, παρουσιάζοντας τα τμήματα και τις μεταξύ τους σχέσεις. Βασικό του συστατικό είναι το τμήμα (module). Τα τμήματα συνδέονται μεταξύ τους με βέλη που συμβολίζουν τη μεταφορά ελέγχου. Πάνω στο βέλος μπορεί να τοποθετηθεί ένα μικρότερο βέλος (δεσμός) με όνομα που περιγράφει ένα στοιχείο δεδομένων που κινείται από τη μία ενότητα προς την άλλη.

4.2.2.1. Οντότητες με όμοιες λειτουργίες⁶

4.2.2.1.1. Προσθήκη στην οντότητα



Διάγραμμα 4.12

Περιγραφή

Η ΠΡΟΣΘΗΚΗ ΣΤΗΝ ΟΝΤΟΤΗΤΑ αποτελείται από οκτώ modules. Το γενικό module ονομάζεται ΠΡΟΣΘΗΚΗ ΣΤΗΝ ΟΝΤΟΤΗΤΑ. Το συγκεκριμένο module καλεί τα υπόλοιπα modules έτσι ώστε να γίνουν οι απαραίτητες ενέργειες και να διεκπεραιωθεί η λειτουργία της εκτύπωσης των στοιχείων της οντότητας.

Το module ΕΙΣΑΓΩΓΗ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ είναι υπεύθυνο να διαβάσει από τον χρήστη τα στοιχεία του module.

Το module ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΣΑΓΩΓΗΣ είναι το module το οποίο είναι υπεύθυνο για την επεξεργασία των στοιχείων της εγγραφής που έχει εισαγάγει ο χρήστης. Το module της επεξεργασίας διεκπεραιώνει την λειτουργία του, καλώντας άλλα δύο modules: το module ΕΠΙΚΥΡΩΣΗΣ ΟΡΘΟΤΗΤΑΣ ΤΩΝ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ και το module ΑΠΟΘΗΚΕΥΣΗ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ ΣΤΗ ΒΔ.

Το module ΕΠΙΚΥΡΩΣΗΣ ΟΡΘΟΤΗΤΑΣ ΤΩΝ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ παίρνει ως παράμετρο το στοιχείο οντότητας και επιστρέφει τον έλεγχο της κατάστασης.

⁶ Λόγω επαναλήψεων στην αρχιτεκτονική ανάλυση των λειτουργιών, οι όμοιες λειτουργίες παρατίθενται μια φορά. Όπου αναφέρεται η λέξη οντότητα στην παράγραφο 4.2.2.1, εννοείται μία από τις οντότητες Πελάτης, Εταιρεία, Προμηθευτής, Προϊόν, Προϊόντα Συλλογών, Πρώτη Ύλη, Λιανική Πώληση, Χονδρική Πώληση, Χρώμα, Συλλογή.

Τέλος, το module *ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ* είναι υπεύθυνο για την ενημέρωση του χρήστη μετακατάλληλα μηνύματα. Καλεί δύο modules: το module *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΕΠΙΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* και το module *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* για να ενημερώσει το χρήστη για την επιτυχία ή την αποτυχία (αντίστοιχα) της ολοκλήρωσης της λειτουργίας.

Συνεκτικότητα / Σύζευξη

Με τον πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού υπάρχει σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οι οποίοι είναι αναπόφευκτοι.

4.2.2.1.2. Ενημέρωση οντότητας

▼



Διάγραμμα 4.13

Περιγραφή

Η *ΕΝΗΜΕΡΩΣΗ ΟΝΤΟΤΗΤΑΣ* αποτελείται από οκτώ modules. Το γενικό module *ΕΝΗΜΕΡΩΣΗ ΟΝΤΟΤΗΤΑΣ* είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για να ενημερωθεί μια εγγραφή στη βάση δεδομένων.

Το module *ΕΙΣΑΓΩΓΗ ΝΕΩΝ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ* είναι υπεύθυνο να διαβάσει από τον χρήστη τα νέα στοιχεία τα οποία θα χρησιμοποιηθούν για την ενημέρωση μιας οντότητας μέσω μιας φόρμας.

Το module *ΕΠΕΞΕΡΓΑΣΙΑ ΕΝΗΜΕΡΩΣΗΣ* είναι το module το οποίο είναι υπεύθυνο για την επεξεργασία των νέων στοιχείων που έχει εισαγάγει ο χρήστης. Το module της επεξεργασίας διεκπεραιώνει την λειτουργία του, επικυρώνοντας τα νέα στοιχεία της οντότητας με το module *ΕΠΙΚΥΡΩΣΗ ΟΡΘΟΤΗΤΑΣ ΝΕΩΝ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ* και ακολούθως χρησιμοποιώντας το module *ΑΠΟΘΗΚΕΥΣΗ ΝΕΩΝ ΣΤΟΙΧΕΙΩΝ ΟΝΤΟΤΗΤΑΣ ΣΤΗ ΒΔ* αποθηκεύει τα νέα στοιχεία της οντότητας στη βάση δεδομένων.

Τέλος, το module *ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ* καλεί ένα από τα δύο modules *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΕΠΙΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* και *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* τα οποία εκτυπώνουν το κατάλληλο μήνυμα επιτυχίας ή αποτυχίας στην οθόνη ανάλογα με το αν η ενημέρωση ήταν επιτυχής ή όχι.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οιοποίοι είναι αναπόφευκτοι.

4.2.2.1.3. Διαγραφή από την οντότητα



Διάγραμμα 4.14

Περιγραφή

Η *ΔΙΑΓΡΑΦΗ ΑΠΟ ΤΗΝ ΟΝΤΟΤΗΤΑ* αποτελείται από οκτώ modules. Το γενικό module *ΔΙΑΓΡΑΦΗ ΑΠΟ ΤΗΝ ΟΝΤΟΤΗΤΑ* είναι υπεύθυνο να καλεί ταυτόλοια modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για να διαγραφεί μια εγγραφή στη βάση δεδομένων.

Το module *ΦΟΡΤΩΣΗ ΕΓΓΡΑΦΗΣ ΟΝΤΟΤΗΤΑΣ* είναι υπεύθυνο για τη φόρτωση της εγγραφής η οποία θα διαγραφεί, παίρνοντας την από το module της αναζήτησης.

Το module *ΕΠΕΞΕΡΓΑΣΙΑ ΔΙΑΓΡΑΦΗΣ* είναι το module το οποίο είναι υπεύθυνο να ελέγξει τους περιορισμούς ακεραιότητας που πιθανόν να παραβιάζονται με τη διαγραφή της εγγραφής κάνοντας χρήση του module *ΕΛΕΓΧΟΣ ΠΕΡΙΟΡΙΣΜΩΝ ΑΚΕΡΑΙΟΤΗΤΑΣ* και ακολούθως χρησιμοποιώντας το module *ΔΙΑΓΡΑΦΗ ΕΓΓΡΑΦΗΣ ΟΝΤΟΤΗΤΑΣ ΑΠΟ ΤΗ ΒΔ* να διαγράψει την εγγραφή από τη βάση δεδομένων.

Τέλος, το module *ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ* καλεί ένα από τα δύο modules *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΕΠΙΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* και *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* τα οποία

εκτυπώνουν το κατάλληλο μήνυμα επιτυχίας ή αποτυχίας στην οθόνη ανάλογα με το αν η διαγραφή ήταν επιτυχής ή όχι.

Συνεκτικότητα /Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οι οποίοι είναι αναπόφευκτοι.

4.2.2.1.4. Αναζήτηση στην οντότητα



Διάγραμμα 4.15

Περιγραφή

Η *ΑΝΑΖΗΤΗΣΗ ΣΤΗΝ ΟΝΤΟΤΗΤΑ* αποτελείται από οκτώ modules. Το γενικό module *ΑΝΑΖΗΤΗΣΗ ΣΤΗΝ ΟΝΤΟΤΗΤΑ* είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για την αναζήτηση εγγραφών από τη βάση δεδομένων.

Το module *ΕΙΣΑΓΩΓΗ ΚΡΙΤΗΡΙΩΝ ΑΝΑΖΗΤΗΣΗΣ ΣΤΗΝ ΟΝΤΟΤΗΤΑ* είναι υπεύθυνο να διαβάσει από τον χρήστη τα κριτήρια που θα χρησιμοποιηθούν για την αναζήτηση, μέσω μιας φόρμας.

Το module *ΕΠΕΞΕΡΓΑΣΙΑ ΑΝΑΖΗΤΗΣΗΣ* είναι το module το οποίο είναι υπεύθυνο να αναζητήσει τις εγγραφές που πληρούν τα κριτήρια αναζήτησης χρησιμοποιώντας το module *ΦΙΛΤΡΑΡΙΣΜΑ ΕΓΓΡΑΦΩΝ ΜΕ ΚΡΙΤΗΡΙΑ*. Ακολούθως, χρησιμοποιώντας το module *ΑΝΑΚΤΗΣΗ ΕΓΓΡΑΦΩΝ ΑΠΟ ΤΗ ΒΛ* γίνεται ανάκτηση αυτών των εγγραφών από τη βάση δεδομένων.

Τέλος, το module *ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ* καλεί ένα από τα δύο modules *ΕΜΦΑΝΙΣΗ ΕΓΓΡΑΦΩΝ ΑΝΑΖΗΤΗΣΗΣ ΣΤΗΝ ΟΘΟΝΗ* και *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* τα οποία είτε εκτυπώνουν τις εγγραφές που πληρούν τα κριτήρια αναζήτησης στην οθόνη, είτε εκτυπώνουν το κατάλληλο μήνυμα αποτυχίας στην οθόνη - ανάλογα με το αν η αναζήτηση ήταν επιτυχής ή όχι.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οι οποίοι είναι αναπόφευκτοι.

4.2.2.1.5. Εκτύπωση εγγραφής από την οντότητα



Διάγραμμα 4.16

Περιγραφή

Η ΕΚΤΥΠΩΣΗ ΕΓΓΡΑΦΗΣ ΑΠΟ ΤΗΝ ΟΝΤΟΤΗΤΑ αποτελείται από εννιά modules. Το γενικό module ΕΚΤΥΠΩΣΗ ΕΓΓΡΑΦΗΣ ΑΠΟ ΤΗΝ ΟΝΤΟΤΗΤΑ είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για να εκτυπωθεί μια εγγραφή από τη βάση δεδομένων.

Το module ΦΟΡΤΩΣΗ ΕΓΓΡΑΦΗΣ ΟΝΤΟΤΗΤΑΣ είναι υπεύθυνο να μεταφέρει τα στοιχεία της εγγραφής από τη βάση δεδομένων στο module.

Το module ΕΠΕΞΕΡΓΑΣΙΑ ΕΚΤΥΠΩΣΗΣ είναι υπεύθυνο να βρει τους ενεργούς εκτυπωτές του δικτύου μέσω του module ΕΥΡΕΣΗ ΚΑΙ ΕΜΦΑΝΙΣΗ ΕΝΕΡΓΩΝ ΕΚΤΥΠΩΤΩΝ, να επεξεργαστεί τα στοιχεία της εγγραφής και να τα εκτυπώσει, χρησιμοποιώντας το module ΠΡΟΕΤΟΙΜΑΣΙΑ ΕΚΤΥΠΩΣΙΜΩΝ ΚΑΙ ΕΚΤΥΠΩΣΗ.

Τέλος, το module ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ είναι υπεύθυνο να εμφανίσει στην οθόνη το εκτυπώσιμο όπως ακριβώς θα εκτυπωθεί και να το εκτυπώσει, ή σε περίπτωση που εμφανίστηκε κάποιο πρόβλημα κατά την προετοιμασία ή την εκτύπωση, τότε να εμφανίσει μήνυμα αποτυχίας.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες

πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οιοποίοι είναι αναπόφευκτοι.

4.2.2.2. Οντότητες με επιπλέον ή διαφορετικές λειτουργίες**4.2.2.2.1. Αρχιτεκτονική Σχεδίαση Πρώτων Υλών: Αυτόματη Προειδοποίηση εξάντλησης πρώτης ύλης**

Διάγραμμα 4.17

Περιγραφή

Η ΑΥΤΟΜΑΤΗ ΠΡΟΕΙΔΟΠΟΙΗΣΗ ΕΞΑΝΤΛΗΣΗΣ ΠΡΩΤΗΣ ΥΛΗΣ αποτελείται από οκτώ modules. Το γενικό module ονομάζεται ΑΥΤΟΜΑΤΗ ΠΡΟΕΙΔΟΠΟΙΗΣΗ ΕΞΑΝΤΛΗΣΗΣ ΠΡΩΤΗΣ ΥΛΗΣ. Είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για την εύρεση των πρώτων υλών που έχουν εξαντληθεί.

Το module ΦΟΡΤΩΣΗ ΕΓΓΡΑΦΗΣ ΠΡΩΤΗΣ ΥΛΗΣ είναι υπεύθυνο να μεταφέρει τα στοιχεία της εγγραφής από τη βάση δεδομένων στο module.

Το module ΕΠΕΞΕΡΓΑΣΙΑ ΕΓΓΡΑΦΩΝ είναι το module το οποίο είναι υπεύθυνο να φιλτράρει τις εγγραφές για να βρει τις εξαντλημένες πρώτες ύλες, χρησιμοποιώντας το module ΣΥΓΚΡΙΣΗ ΤΡΕΧΟΥΣΑΣ ΠΟΣΟΤΗΤΑΣ ΜΕ ΚΑΤΩΤΕΡΟ ΟΡΙΟ. Ακολούθως, χρησιμοποιώντας το module ΠΡΟΕΤΟΙΜΑΣΙΑ ΛΙΣΤΑΣ ΕΞΑΝΤΛΗΜΕΝΩΝ ΠΡΩΤΩΝ ΥΛΩΝ δημιουργείται η λίστα με τις ζητούμενες εγγραφές.

Τέλος, το module ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ καλεί ένα από τα δύο modules ΕΜΦΑΝΙΣΗ ΛΙΣΤΑΣ ΕΞΑΝΤΛΗΜΕΝΩΝ ΠΡΩΤΩΝ ΥΛΩΝ ΣΤΗΝ ΟΘΟΝΗ και ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ τα οποία είτε παρουσιάζουν τις εγγραφές που έχουν εξαντληθεί στην οθόνη, είτε εκτυπώνουν το κατάλληλο μήνυμα αποτυχίας στην οθόνη - ανάλογα με το αν η επεξεργασία των εγγραφών ήταν επιτυχής ή όχι.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οιοποίοι είναι αναπόφευκτοι.

4.2.2.2. Αρχιτεκτονική Σχεδίαση Πελάτη: Αυτοματοποιημένη Εύρεση εορτής πελάτη



Διάγραμμα 4.18

Περιγραφή

Η ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΕΥΡΕΣΗ ΕΟΡΤΗΣ ΠΕΛΑΤΗ αποτελείται από οκτώ modules. Το γενικό module ΕΥΡΕΣΗ ΕΟΡΤΗΣ ΠΕΛΑΤΗ είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για την εύρεση των ημερομηνιών εορτών που σχετίζονται με το όνομα του πελάτη από τη βάση δεδομένων.

Το module ΠΛΗΚΤΡΟΛΟΓΗΣΗ ΟΝΟΜΑΤΟΣ ΠΕΛΑΤΗ είναι υπεύθυνο να διαβάσει από τον χρήστη το όνομα του πελάτη που θα χρησιμοποιηθεί για την αναζήτηση.

Το module ΕΠΕΞΕΡΓΑΣΙΑ ΑΝΑΖΗΤΗΣΗΣ είναι το module το οποίο είναι υπεύθυνο να ανακτήσει τις εγγραφές των εορτών από τη βάση δεδομένων, χρησιμοποιώντας το module ΑΝΑΚΤΗΣΗ ΕΓΓΡΑΦΩΝ ΑΠΟ ΤΗ ΒΔ. Ακολούθως, αναζητούνται οι εγγραφές που σχετίζονται με το όνομα που δόθηκε, κάνοντας χρήση του module ΕΥΡΕΣΗ ΣΧΕΤΙΚΩΝ ΗΜΕΡΟΜΗΝΙΩΝ ΕΟΡΤΩΝ.

Τέλος, το module ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ καλεί ένα από τα δύο modules ΕΜΦΑΝΙΣΗ ΗΜΕΡΟΜΗΝΙΩΝ ΕΓΓΡΑΦΩΝ ΣΤΗΝ ΟΘΟΝΗ και ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ τα οποία είτε εκτυπώνουν τις ημερομηνίες των εορτών που σχετίζονται με το όνομα του πελάτη στην οθόνη, είτε εκτυπώνουν το κατάλληλο μήνυμα αποτυχίας στην οθόνη - ανάλογα με το αν η αναζήτηση ήταν επιτυχής ή όχι.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οι οποίοι είναι αναπόφευκτοι.

4.2.2.2.3. Αρχιτεκτονική Σχεδίαση Πελάτη: Αυτοματοποιημένη Αποστολή Ευχετήριας κάρτας



Διάγραμμα 4.19

Περιγραφή

Η ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΑΠΟΣΤΟΛΗ ΕΥΧΕΤΗΡΙΑΣ ΚΑΡΤΑΣ αποτελείται από εννιά modules. Το γενικό module ονομάζεται ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΑΠΟΣΤΟΛΗ ΕΥΧΕΤΗΡΙΑΣ ΚΑΡΤΑΣ. Είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για την εύρεση των ημερησίων εορταζόντων και την αποστολή σε αυτού ευχετήριου ηλεκτρονικού μηνύματος.

Το module ΦΟΡΤΩΣΗ ΕΓΓΡΑΦΩΝ ΠΕΛΑΤΩΝ είναι υπεύθυνο να μεταφέρει τα στοιχεία των εγγραφώντων πελατών από τη βάση δεδομένων στο module.

Το module ΕΠΕΞΕΡΓΑΣΙΑ ΕΓΓΡΑΦΩΝ είναι το module το οποίο είναι υπεύθυνο να φιλτράρει τις εγγραφές των πελατών για να βρει τους εορτάζοντες, χρησιμοποιώντας το module ΣΥΓΚΡΙΣΗ ΗΜΕΡΟΜΗΝΙΩΝ ΕΟΡΤΗΣ ΚΑΙ ΓΕΝΕΘΛΙΩΝ ΜΕ ΣΗΜΕΡΙΝΗ ΗΜΕΡΟΜΗΝΙΑ. Ακολουθώντας, χρησιμοποιώντας το module ΔΗΜΙΟΥΡΓΙΑ ΛΙΣΤΑΣ ΜΕ ΗΜΕΡΗΣΙΟΥΣ ΕΟΡΤΑΖΟΝΤΕΣ δημιουργείται η λίστα με τους ημερήσιους εορτάζοντες.

Τέλος, το module ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ είναι υπεύθυνο να εμφανίσει στην οθόνη τους εορτάζοντες και να αποστείλει στον καθένα τους ευχετήριο ηλεκτρονικό μήνυμα, ή σε περίπτωση που εμφανίστηκε κάποιο πρόβλημα κατά την όλη διαδικασία, να εμφανίσει μήνυμα αποτυχίας.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οιοποίοι είναι αναπόφευκτοι.

4.2.2.2.4. Αρχιτεκτονική Σχεδίαση Επαφών⁷: Φόρτωση επαφών σε κινητό τηλέφωνο



Διάγραμμα 4.20

Περιγραφή

Η ΦΟΡΤΩΣΗ ΕΠΑΦΩΝ ΣΕ ΚΙΝΗΤΟ ΤΗΛΕΦΩΝΟ αποτελείται από έξι modules. Το γενικό module ονομάζεται ΦΟΡΤΩΣΗ ΕΠΑΦΩΝ ΣΕ ΚΙΝΗΤΟ ΤΗΛΕΦΩΝΟ. Είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για να φορτωθούν οι εγγραφές των πελατών ή/και των εταιρειών στο κινητό τηλέφωνο.

Το module ΦΟΡΤΩΣΗ ΕΓΓΡΑΦΩΝ ΕΠΑΦΩΝ είναι υπεύθυνο να μεταφέρει τις εγγραφές των επαφών από τη βάση δεδομένων στο module.

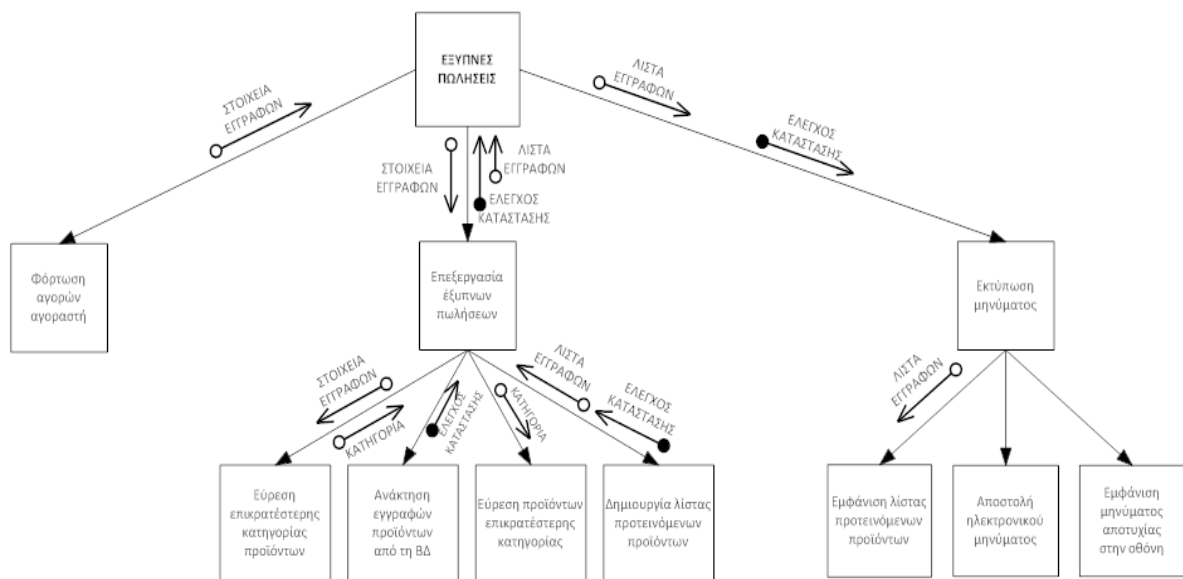
Το module ΠΡΟΕΤΟΙΜΑΣΙΑ ΕΓΓΡΑΦΩΝ ΓΙΑ ΕΞΑΓΩΓΗ είναι το module το οποίο είναι υπεύθυνο να ετοιμάσει τις επαφές για να εγγραφούν σε αρχείο.

Τέλος, το module ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ είναι υπεύθυνο να γράψει σε ένα αρχείο τιμών διαχωρισμένες με κόμμα (csvfile=comma-separatedvaluefile) τις εγγραφές, ή σε περίπτωση που εμφανίστηκε κάποιο πρόβλημα κατά την όλη διαδικασία, να εμφανίσει μήνυμα αποτυχίας.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οι οποίοι είναι αναπόφευκτοι.

⁷ Όπου αναφέρεται η λέξη *επαφή* στην παράγραφο 4.2.2.2, εννοείται μία από τις οντότητες Πελάτης και Εταιρεία.

4.2.2.2.5. Αρχιτεκτονική Σχεδίαση Αγοραστή⁸: Έξυπνες πωλήσεις

Διάγραμμα 4.21

Περιγραφή

Οι *ΕΞΥΠΝΕΣ ΠΩΛΗΣΕΙΣ* αποτελούνται από έντεκα modules. Το γενικό module ονομάζεται *ΕΞΥΠΝΕΣ ΠΩΛΗΣΕΙΣ*. Είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για να βρεθούν προϊόντα που ανταποκρίνονται στα γούστα των πελατών/εταιρειών και να γίνει εισήγηση για αυτά μέσω ηλεκτρονικού ταχυδρομείου.

Το module *ΦΟΡΤΩΣΗ ΑΓΟΡΩΝ ΑΓΟΡΑΣΤΗ* είναι υπεύθυνο να μεταφέρει τις εγγραφές των παραγγελιών του πελάτη/της εταιρείας από τη βάση δεδομένων στο module.

Το module *ΕΠΕΞΕΡΓΑΣΙΑ ΕΞΥΠΝΩΝ ΠΩΛΗΣΕΩΝ* είναι το module το οποίο είναι υπεύθυνο να ετοιμάσει τη λίστα με τις προτεινόμενα προϊόντα η οποία θα αποσταλεί στον αγοραστή. Χρησιμοποιώντας το module *ΕΥΡΕΣΗ ΕΠΙΚΡΑΤΕΣΤΕΡΗΣ ΚΑΤΗΓΟΡΙΑΣ ΠΡΟΪΟΝΤΩΝ ΑΓΟΡΩΝ*, γίνεται ομαδοποίηση των προϊόντων αγορών σε κατηγορίες και καταμέτρηση στην κάθε κατηγορία. Η κατηγορία με τα περισσότερα προϊόντα επιστρέφεται. Με τα modules *ΑΝΑΚΤΗΣΗ ΕΓΓΡΑΦΩΝ ΠΡΟΪΟΝΤΩΝ ΑΠΟ ΤΗ ΒΔ* και *ΕΥΡΕΣΗ ΠΡΟΪΟΝΤΩΝ ΕΠΙΚΡΑΤΕΣΤΕΡΗΣ ΚΑΤΗΓΟΡΙΑΣ ΠΡΟΪΟΝΤΩΝ*, φιλτράρονται τα προϊόντα της βάσης δεδομένων για να παρθούν μόνο αυτά της επικρατέστερης κατηγορίας. Το module *ΔΗΜΙΟΥΡΓΙΑ ΛΙΣΤΑΣ ΠΡΟΤΕΙΝΟΜΕΝΩΝ ΠΡΟΪΟΝΤΩΝ* επιστρέφει τη λίστα η οποία θα αποσταλεί στον αγοραστή.

Τέλος, το module *ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ* είναι υπεύθυνο να εμφανίσει στην οθόνη την εν λόγω λίστα προϊόντων και να την αποστείλει με ηλεκτρονικό μήνυμα στον αγοραστή, ή σε περίπτωση που εμφανίστηκε κάποιο πρόβλημα κατά την όλη διαδικασία, να εμφανίσει μήνυμα αποτυχίας.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οιοποίοι είναι αναπόφευκτοι.

⁸ Όπου αναφέρεται η λέξη *αγοραστής* στην παράγραφο 4.2.2.2, εννοείται μία από τις οντότητες Πελάτης και Εταιρεία.

4.2.2.2.6. Αρχιτεκτονική Σχεδίαση Παραγγελίες⁹: Τρέχουσες Παραγγελίες



Διάγραμμα 4.22

Περιγραφή

Οι *ΤΡΕΧΟΥΣΕΣ ΠΑΡΑΓΓΕΛΕΙΕΣ* αποτελούνται από οκτώ modules. Το γενικό module ονομάζεται *ΤΡΕΧΟΥΣΕΣ ΠΑΡΑΓΓΕΛΕΙΕΣ*. Είναι υπεύθυνο να καλεί τα υπόλοιπα modules, έτσι ώστε να γίνουν οι απαραίτητες ενέργειες για την εύρεση των παραγγελιών που δεν έχουν ακόμη ολοκληρωθεί.

Το module *ΦΟΡΤΩΣΗ ΕΓΓΡΑΦΩΝ ΠΑΡΑΓΓΕΛΙΩΝ* είναι υπεύθυνο να μεταφέρει τα στοιχεία των εγγραφών από τη βάση δεδομένων στο module.

Το module *ΕΠΕΞΕΡΓΑΣΙΑ ΕΓΓΡΑΦΩΝ* είναι το module το οποίο είναι υπεύθυνο να φιλτράρει τις εγγραφές για να βρει τις τρέχουσες παραγγελίες, χρησιμοποιώντας το module *ΣΥΓΚΡΙΣΗ ΗΜΕΡΟΜΗΝΙΑΣ ΠΑΡΑΔΟΣΗΣ ΜΕ ΣΗΜΕΡΙΝΗ ΗΜΕΡΟΜΗΝΙΑ*. Ακολουθώντας, χρησιμοποιώντας το module *ΠΡΟΕΤΟΙΜΑΣΙΑ ΛΙΣΤΑΣ ΜΕ ΤΡΕΧΟΥΣΕΣ ΠΑΡΑΓΓΕΛΙΕΣ* δημιουργείται η λίστα με τις ζητούμενες εγγραφές.

Τέλος, το module *ΕΚΤΥΠΩΣΗ ΜΗΝΥΜΑΤΟΣ* καλεί ένα από τα δύο modules *ΕΜΦΑΝΙΣΗ ΛΙΣΤΑΣ ΜΕ ΤΡΕΧΟΥΣΕΣ ΠΑΡΑΓΓΕΛΙΕΣ* και *ΕΜΦΑΝΙΣΗ ΜΗΝΥΜΑΤΟΣ ΑΠΟΤΥΧΙΑΣ ΣΤΗΝ ΟΘΟΝΗ* τα οποία είτε παρουσιάζουν τις εγγραφές που έχουν διεκπεραιωθεί ακόμη στην οθόνη, είτε εκτυπώνουν το κατάλληλο μήνυμα αποτυχίας στην οθόνη - ανάλογα με το αν η επεξεργασία των εγγραφών ήταν επιτυχής ή όχι.

Συνεκτικότητα / Σύζευξη

Με το πιο πάνω σχεδιασμό, επιτυγχάνεται υψηλή συνεκτικότητα μιας και κάθε module κάνει μόνο μια συγκεκριμένη διαδικασία και περνά στα modules που καλεί μόνο τις απαραίτητες πληροφορίες. Το επίπεδο σύζευξης όμως δεν είναι ιδιαίτερα χαμηλό, αφού έχουμε σύζευξη ελέγχου λόγω των ελέγχων καταστάσεων που χρειάζονται να γίνονται, οιοποίοι είναι αναπόφευκτοι.

⁹ Όπου αναφέρεται η λέξη *παραγγελίες* στην παράγραφο 4.2.2.2, εννοείται μία από τις οντότητες Λιανική Παραγγελία και Χονδρική Παραγγελία.

4.3. ΑΝΑΛΥΤΙΚΗ ΣΧΕΔΙΑΣΗ

Μετά την προκαταρκτική μελέτη ενός νέου κατασκευαστικού μέρους ή προϊόντος, πρέπει να γίνει εξέταση και αναλυτική σχεδίαση κάθε στοιχείου του, έτσι ώστε να εξασφαλιστεί η εγκυρότητα των λειτουργιών που θα εκτελούνται. Παράλειψη ή αδυναμία σε οποιοδήποτε μέρος της σχεδίασης μπορεί να θέσει σε κίνδυνο την επιτυχία του προϊόντος στην αγορά, καθώς τροποποιήσεις στον σχεδιασμό κατά τα τελευταία στάδια της διαδικασίας σχεδιασμού είναι ιδιαίτερα υψηλά σε κόστος.

4.3.1. Οντότητες με όμοιες λειτουργίες¹⁰

4.3.1.1. Module: Προσθήκη στην οντότητα

Αυτό το module είναι υπεύθυνο για τη σωστή εισαγωγή των νέων στοιχείων στην οντότητα. Κατά την εκτέλεσή του, γίνεται επαλήθευση της ορθότητας των νέων στοιχείων που δίνει ο χρήστης, με βάση κάποιους περιορισμούς και επιστρέφεται μια παράμετρο κατάστασης, η οποία δείχνει κατά πόσο τα νέα στοιχεία είναι έγκυρα ή όχι. Αν ο έλεγχος είναι έγκυρος, γίνεται η προσθήκη των στοιχείων στο αρχείο, διαφορετικά εμφανίζεται μήνυμα λάθους αναφέροντας πού έγινε το λάθος.

Όνομα module	New_Record
Τύπος	Συνάρτηση
Τύπος επιστροφής	Boolean
Παράμετροι εισόδου	Στοιχεία οντότητας: string, integer, boolean
Παράμετροι εξόδου	ValidFlag
Μηνύματα λάθους	Λανθασμένα στοιχεία εισαγωγής
Προσπελάσιμα αρχεία	Αρχείο οντότητας
Τροποποίηση αρχείων	Αρχείο οντότητας
Καλούμενα Modules	-----

Πίνακας 4.1

4.3.1.2. Module: Ενημέρωση οντότητας

Αυτό το module είναι υπεύθυνο για την τροποποίηση των στοιχείων στην οντότητα. Κατά την εκτέλεσή του, γίνεται επαλήθευση της ορθότητας των νέων στοιχείων που δίνει ο χρήστης, με βάση κάποιους περιορισμούς και επιστρέφεται μια παράμετρο κατάστασης, η οποία δείχνει κατά πόσο τα νέα στοιχεία είναι έγκυρα ή όχι. Αν ο έλεγχος είναι έγκυρος, γίνεται η ενημέρωση των στοιχείων στο αρχείο, διαφορετικά εμφανίζεται μήνυμα λάθους αναφέροντας πού έγινε το λάθος.

Όνομα module	Edit_Record
Τύπος	Συνάρτηση
Τύπος επιστροφής	Boolean
Παράμετροι εισόδου	Νέα στοιχεία οντότητας: string, integer, boolean
Παράμετροι εξόδου	ValidFlag
Μηνύματα λάθους	Λανθασμένα στοιχεία εισαγωγής
Προσπελάσιμα αρχεία	Αρχείο οντότητας

¹⁰ Λόγω επαναλήψεων στην αρχιτεκτονική ανάλυση των λειτουργιών, οι όμοιες λειτουργίες παρατίθενται μια φορά. Όπου αναφέρεται η λέξη οντότητα στην παράγραφο 4.3.1, εννοείται μία από τις οντότητες Πελάτης, Εταιρεία, Προμηθευτής, Προϊόν, Προϊόντα Συλλογών, Πρώτη Ύλη, Λιανική Πώληση, Χονδρική Πώληση, Χρώμα, Συλλογή.

Τροποποίηση αρχείων	Αρχείο οντότητας
Καλούμενα Modules	-----

Πίνακας 4.2

4.3.1.3. Module: Διαγραφή από την οντότητα

Το module αυτό είναι υπεύθυνο για την διαγραφή μιας εγγραφής από την οντότητα. Κατά τηνεκτέλεσήτου, ελέγχεται πόσο είναι δυνατή η διαγραφή της εγγραφής. Εφόσον ο έλεγχος είναι θετικός, η διαδικασία ολοκληρώνεται με επιτυχία και αφαιρείται η εν λόγω εγγραφή από το αρχείο της οντότητας.

Όνομα module	Delete_Record
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	-----
Παράμετροι εξόδου	-----
Μηνύματα λάθους	Αδύνατη Διαγραφή
Προσπελάσιμα αρχεία	Αρχείο οντότητας
Τροποποίηση αρχείων	Αρχείο οντότητας
Καλούμενα Modules	-----

Πίνακας 4.3

4.3.1.4. Module: Αναζήτηση στην οντότητα

Το module αυτό είναι υπεύθυνο για την αναζήτηση στοιχείων σε μια οντότητα. Κατά την εκτέλεσή του, γίνεται αναζήτηση με τα στοιχεία που έχει δώσει ο χρήστης και επιστρέφεται μια λίστα με τις φιλτραρισμένες εγγραφές.

Όνομα module	Search_record
Τύπος	Συνάρτηση
Τύπος επιστροφής	String
Παράμετροι εισόδου	Στοιχεία οντότητας: string, integer
Παράμετροι εξόδου	Στοιχεία οντότητας
Μηνύματα λάθους	-----
Προσπελάσιμα αρχεία	Αρχείο οντότητας
Τροποποίηση αρχείων	-----
Καλούμενα Modules	-----

Πίνακας 4.4

4.3.1.5. Module: Εκτύπωση εγγραφής από την οντότητα

Το module αυτό είναι υπεύθυνο για την εκτύπωση των στοιχείων μια εγγραφής της οντότητας. Κατά την εκτέλεσή του, γίνεται η αποστολή των δεδομένων της οντότητας που έχει επιλέξει ο χρήστης στον εκτυπωτή.

Όνομα module	Print_record
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	Στοιχεία οντότητας: string, integer, boolean
Παράμετροι εξόδου	Στοιχεία οντότητας
Μηνύματα λάθους	-----
Προσπελάσιμα αρχεία	Αρχείο οντότητας
Τροποποίηση αρχείων	-----
Καλούμενα Modules	Search_record

Πίνακας 4.5

4.3.2. Οντότητες με επιπλέον ή διαφορετικές λειτουργίες

4.3.2.1. Module: Αυτόματη Προειδοποίηση εξάντλησης πρώτης ύλης

Αυτό το module είναι υπεύθυνο για την εύρεση των εγγραφών των πρώτων υλών οι οποίες έχουν εξαντληθεί. Κατά την εκτέλεση του συγκρίνεται την τρέχουσα ποσότητα της πρώτης ύλης με το κατώτερο όριο της και αν αυτή είναι ίση ή μικρότερη του ορίου, προστίθεται μια λίστα η οποία περιλαμβάνει τις πρώτες ύλες οι οποίες έχουν εξαντληθεί.

Όνομα module	Find_run_out
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	Στοιχεία πρώτων υλών
Παράμετροι εξόδου	Στοιχεία πρώτων υλών
Μηνύματα λάθους	-----
Προσπελάσιμα αρχεία	Αρχείο πρώτων υλών
Τροποποίηση αρχείων	-----
Καλούμενα Modules	-----

Πίνακας 4.6

4.3.2.2. Module: Αυτοματοποιημένη Εύρεση εορτής πελάτη

Αυτό το module είναι υπεύθυνο για τη εύρεση όλων των πιθανών εορτών ενός πελάτη. Κατά την εκτέλεσή του, βρίσκει από τη βάση δεδομένων τις εγγραφές των εορτών που σχετίζονται με το όνομα που έχει πληκτρολογηθεί και δημιουργεί μια λίστα η οποία περιλαμβάνει όλες τις ευρεθείσες εορτές. Εάν δεν βρεθεί κάποια ημερομηνία που να σχετίζεται με το δοσμένο όνομα, εμφανίζεται μήνυμα λάθους.

Όνομα module	Find_nameday
Τύπος	Συνάρτηση

Τύπος επιστροφής	Date
Παράμετροι εισόδου	Στοιχεία πελατών, ονομαστικών εορτών
Παράμετροι εξόδου	-----
Μηνύματα λάθους	Μη καταχωρημένο όνομα
Προσπελάσιμα αρχεία	Αρχείο πελατών, ονομαστικών εορτών
Τροποποίηση αρχείων	-----
Καλούμενα Modules	-----

Πίνακας 4.7

4.3.2.3. Module: Αυτοματοποιημένη Αποστολή Ευχετήριας κάρτας

Αυτό το module είναι υπεύθυνο για την εύρεση όλων των εορταζόντων της ημέρας και την αποστολή ευχετήριου μηνύματος. Κατά την εκτέλεσή του, βρίσκει από τη βάση δεδομένων τις εγγραφές των πελατών που έχουν γιορτή ή γενέθλια την τρέχουσα ημέρα και δημιουργεί μια λίστα η οποία περιλαμβάνει όλους τους εορτάζοντες. Ακολουθώντας τους αποστέλλει με ηλεκτρονικό μήνυμα ευχετήρια κάρτα. Εάν για κάποιο λόγο αποτύχει η διαδικασία αποστολής ηλεκτρονικού μηνύματος, εμφανίζεται μήνυμα λάθους.

Όνομα module	Send_email
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	Στοιχεία πελατών, συνεργαζόμενων εταιρειών
Παράμετροι εξόδου	-----
Μηνύματα λάθους	Αποτυχία αποστολής email
Προσπελάσιμα αρχεία	Αρχείο πελατών, συνεργαζόμενων εταιρειών
Τροποποίηση αρχείων	-----
Καλούμενα Modules	-----

Πίνακας 4.8

4.3.2.4. Module: Φόρτωση επαφών¹¹ σε κινητό τηλέφωνο

Αυτό το module είναι υπεύθυνο για την εξαγωγή των εγγράφων σε ένα αρχείο τιμών διαχωρισμένες με κόμμα (csvfile=comma-separatedvaluefile). Κατά την εκτέλεσή του, βρίσκει από τη βάση δεδομένων τις εγγραφές των επαφών και τις εξάγει σε ένα αρχείο. Εάν για κάποιο λόγο αποτύχει η διαδικασία εξαγωγής των επαφών, εμφανίζεται μήνυμα λάθους.

Όνομα module	Export_contacts
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	Στοιχεία πελατών, συνεργαζόμενων εταιρειών
Παράμετροι εξόδου	-----
Μηνύματα λάθους	Αποτυχία εγγραφής σε αρχείο
Προσπελάσιμα αρχεία	Αρχείο πελατών, συνεργαζόμενων εταιρειών
Τροποποίηση αρχείων	Δημιουργία νέου αρχείου επαφών (csv)

¹¹ Όπου αναφέρεται η λέξη *επαφή* στην παράγραφο 4.3.2, εννοείται μία από τις οντότητες Πελάτης και Εταιρεία.

Καλούμενα Modules

Πίνακας 4.9

4.3.2.5. Module: Έξυπνες πωλήσεις

Αυτό το module είναι υπεύθυνο για τη δημιουργία μιας λίστας με προτεινόμενα προϊόντα η οποία αποστέλλεται με ηλεκτρονικό μήνυμα στον αγοραστή¹². Κατά την εκτέλεσή του, βρίσκει από τη βάση δεδομένων τις αγορές¹³ ενός αγοραστή και βάση της επικρατέστερης κατηγορίας των ενδυμάτων που παραγγέλθηκαν, δημιουργεί μια λίστα η οποία περιλαμβάνει προϊόντα που ανήκουν στην κατηγορία αυτή και ίσως θα ενδιέφεραν τον αγοραστή. Εάν για κάποιο λόγο αποτύχει η διαδικασία, εμφανίζεται μήνυμα λάθους.

Όνομα module	Smart_seller
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	Στοιχεία πελάτη, συνεργαζόμενων εταιρειών, λιανικών παραγγελιών, χονδρικών παραγγελιών, ειδών ένδυσης
Παράμετροι εξόδου	-----
Μηνύματα λάθους	Αποτυχία διαδικασίας
Προσπελάσιμα αρχεία	Αρχεία πελάτη, συνεργαζόμενων εταιρειών, λιανικών παραγγελιών, χονδρικών παραγγελιών, ειδών ένδυσης
Τροποποίηση αρχείων	-----
Καλούμενα Modules	-----

Πίνακας 4.10

4.3.2.6. Module: Τρέχουσες Παραγγελίες¹⁴

Αυτό το module είναι υπεύθυνο να βρίσκει τις τρέχουσες παραγγελίες. Κατά την εκτέλεσή του, βρίσκει από τη βάση δεδομένων όλες τις παραγγελίες, συγκρίνει την ημερομηνία αποπεράτωσης με την σημερινή ημερομηνία και αν η παραγγελία δεν έχει διεκπεραιωθεί ακόμη και εκκρεμεί, την τοποθετεί σε μια λίστα η οποία περιλαμβάνει τις εκκρεμείς και τρέχουσες παραγγελίες. Τέλος, εμφανίζει τη λίστα με τις παραγγελίες αυτές.

Όνομα module	Current_orders
Τύπος	Συνάρτηση
Τύπος επιστροφής	Void
Παράμετροι εισόδου	Στοιχεία λιανικών παραγγελιών, χονδρικών παραγγελιών
Παράμετροι εξόδου	-----

¹² Όπου αναφέρεται η λέξη αγοραστής στην παράγραφο 4.3.2, εννοείται μία από τις οντότητες Πελάτης και Εταιρεία.

¹³ Όπου αναφέρεται η λέξη αγορές στην παράγραφο 4.3.2, εννοείται μία από τις οντότητες Λιανικές Παραγγελίες και Χονδρικές Παραγγελίες.

¹⁴ Όπου αναφέρεται η λέξη παραγγελίες στην παράγραφο 4.3.2, εννοείται μία από τις οντότητες Λιανική Παραγγελία και Χονδρική Παραγγελία.

Μηνύματα λάθους	-----
Προσπελάσιμα αρχεία	Αρχείαλιανικών παραγγελιών, χονδρικών παραγγελιών
Τροποποίηση αρχείων	-----
Καλούμενα Modules	-----

Πίνακας 4.11

4.4. ΓΡΑΦΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΧΡΗΣΤΗ

Το φιλικό προς τον χρήστη γραφικό περιβάλλον εργασίας είναι ένας από τους σημαντικότερους παράγοντες στην επιτυχία ενός λογισμικού συστήματος. Το γραφικό περιβάλλον εργασίας πρέπει να είναι φιλικό, εύχρηστο αλλά και έξυπνο, έτσι ώστε ο χρήστης να μπορέσει:

- να το μάθει εύκολα,
- να αποδώσει στην εκτέλεση έργου,
- να κάνει όσο το δυνατόν λιγότερα σφαλμάτων κατά τη λειτουργία του συστήματος
- να θυμάται τη χρήση του, έστω κι αν δεν έχει χρησιμοποιήσει το σύστημα αρκετό καιρό

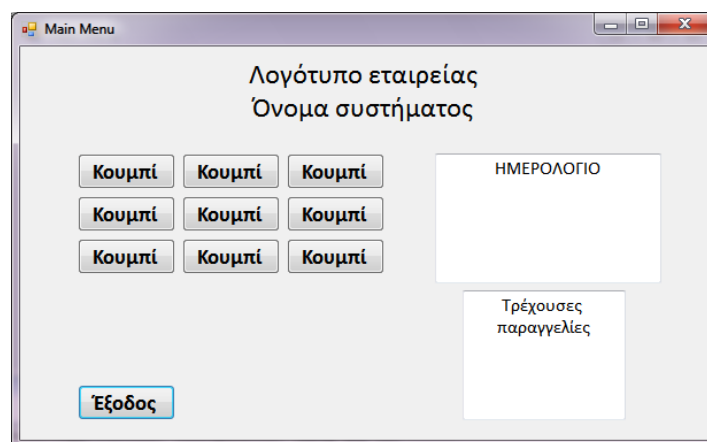
και γενικά να αφήνει τον χρήστη ικανοποιημένο.

Τα κουμπιά σε ένα γραφικό περιβάλλον εργασίας πρέπει να είναι τοποθετημένα με τέτοιο τρόπο, έτσι ώστε ο χρήστης να τα αναγνωρίζει εύκολα και τα ονόματά τους να υπονοούν αμέσως την λειτουργία τους. Επίσης δεν πρέπει να είναι πολλά σε αριθμό, για να αποφευχθεί σύγχυση του χρήστη.

Σημαντικό ρόλο παίζουν τα χρώματα τα οποία θα χρησιμοποιηθούν στο γραφικό περιβάλλον εργασίας, έτσι ώστε να υποβοηθούν στην κατανόηση εννοιών και στην ικανοποίηση του χρήστη.

Όλα τα πιο πάνω οδήγησαν στα πιο κάτω παραδείγματα γραφικού περιβάλλοντος εργασίας:

- **Κυρίως μενού**



Εικόνα 4.1

Στην εικόνα 4.1 παρουσιάζεται ένα πρότυπο του κυρίως μενού του συστήματος. Στο πάνω μέρος θα αναγράφεται το λογότυπο της εταιρείας και το όνομα του συστήματος. Στη φόρμα υπάρχει μια συλλογή από κουμπιά τα οποία αναφέρονται στις βασικές λειτουργίες του συστήματος. Στα δεξιά παρουσιάζονται υπολειτουργίες, όπως το Ημερολόγιο και κάτω από αυτό ένας πίνακας ο οποίος θα παρουσιάζει τις τρέχουσες παραγγελίες. Επίσης υπάρχει και ένα κουμπί εξόδου από το σύστημα.

- **Φόρμα εισαγωγής**

Εικόνα 4.2

Πιο πάνω φαίνεται μια φόρμα εισαγωγής. Ο χρήστης θα μπορεί να συμπληρώνει τα πεδία είτε γράφοντας μέσα σε αυτά, είτε επιλέγοντας από μία λίστα δεδομένων. Στο κάτω μέρος της φόρμας θα υπάρχουν τα κουμπιά:

Αποθήκευση: Αποθηκεύει την εγγραφή και κλείνει τη φόρμα

Ακύρωση: Κλείνει την φόρμα χωρίς να αποθηκεύσει τα στοιχεία

- **Φόρμα αναζήτησης - διαγραφής**

Εικόνα 4.3

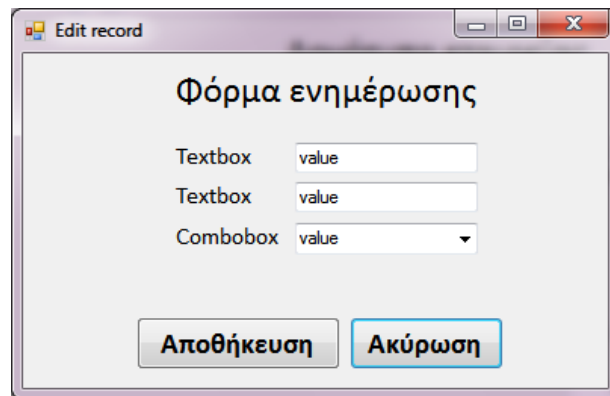
Πιο πάνω φαίνεται μια φόρμα εγγραφών. Ο χρήστης δεν θα μπορεί να συμπληρώνει τα πεδία, αλλά τα πεδία θα εμφανίζονται συμπληρωμένα. Ο χρήστης μπορεί να αναζητήσει μια εγγραφή πληκτρολογώντας κάποια λέξη ή επιλέγοντας από τον πίνακα εγγραφών. Στο κάτω μέρος της φόρμας θα υπάρχουν τα κουμπιά:

Αναζήτηση: Γίνεται αναζήτηση βάση της λέξης που δόθηκε

Διαγραφή: Διαγράφει την εγγραφή

Κλείσιμο φόρμας: Κλείνει τη φόρμα

- **Φόρμα ενημέρωσης**



Εικόνα 4.4

Πιο πάνω φαίνεται μια φόρμα ενημέρωσης. Ο χρήστης θα μπορεί να αναζητά μια εγγραφή και θα του εμφανίζονται τα πεδία συμπληρωμένα. Έτσι θα μπορεί να διαγράψει τα υπάρχοντα πεδία από μια εγγραφή και να συμπληρώσει νέα. Τα τρία κουμπιά στο κάτω μέρος της φόρμας εκτελούν τις παρακάτω λειτουργίες :

Αποθήκευση: Αποθηκεύει τα νέα δεδομένα της εγγραφής και κλείνει τη φόρμα

Ακύρωση: Κλείνει τη φόρμα χωρίς να αποθηκεύσει της αλλαγές

Κεφάλαιο 5. Υλοποίηση

5.1. ΕΙΣΑΓΩΓΗ.....	94
5.2. ΣΥΝΤΟΜΗ ΠΕΡΙΓΡΑΦΗ ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑΣ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ	94
5.3. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ	96
5.3.1. <i>Microsoft Visual Basic .NET</i>	
5.3.2. <i>Microsoft SQL Server® 2008</i>	
5.3.3. <i>Περιγραφή βάσης δεδομένων</i>	
5.4. ΔΥΣΚΟΛΑ ΣΗΜΕΙΑ ΣΤΟΝ ΚΩΔΙΚΑ	101
5.4.1. <i>Ενημέρωση ονομαστικών εορτών</i>	
5.4.2. <i>Αποστολή ευχετήριας κάρτας με ηλεκτρονικό ταχυδρομείο</i>	
5.4.3. <i>Ενημέρωση αποθήκης πρώτων υλών και αυτόματη προειδοποίηση εξάντλησης πρώτης ύλης</i>	
5.4.4. <i>Αποθήκευση εικόνων στη βάση δεδομένων</i>	
5.4.5. <i>Εκτύπωση αποδείξεων και τιμολογίων</i>	
5.4.6. <i>Εκτύπωση ποσού ολογράφως</i>	
5.4.7. <i>Εισηγήσεις ειδών ένδυσης</i>	
5.4.8. <i>Εισαγωγή/εξαγωγή δεδομένων</i>	
5.5. ΠΗΓΑΙΟΣ ΚΩΔΙΚΑΣ ΜΕ ΣΧΟΛΙΑ.....	105
5.5.1. <i>Διαχείριση πελατών</i>	
5.5.2. <i>Διαχείριση συνεργαζόμενων εταιρειών</i>	
5.5.3. <i>Διαχείριση πρώτων υλών</i>	
5.5.4. <i>Διαχείριση προϊόντων</i>	
5.5.5. <i>Διαχείριση ειδών ένδυσης</i>	
5.5.6. <i>Διαχείριση λιανικών πωλήσεων</i>	
5.5.7. <i>Διαχείριση χονδρικών πωλήσεων</i>	
5.5.8. <i>Βοηθητικές/Υποστηρικτικές υλοποιήσεις</i>	

5.1. ΕΙΣΑΓΩΓΗ

Σκοπός του εγγράφου αυτού είναι η καταγραφή της γενικής σύνοψης της υφιστάμενης κατάστασης του συστήματος που αναπτύχθηκε και των γενικών στοιχείων που πιστοποιούν το σύστημα. Με άλλα λόγια, ο σκοπός του εγγράφου είναι η τεκμηρίωση της φάσης της υλοποίησης.

Επιπλέον, σκοπός του εγγράφου αυτού είναι η επεξήγηση των εργαλείων και των γλωσσών προγραμματισμού που χρησιμοποιήθηκαν και της βάσης δεδομένων. Γίνεται μια περιφραστική αναφορά σε συγκεκριμένα σημεία του κώδικα με ιδιαίτερη επισήμανση στα επίμαχα σημεία δυσκολίας και δίνεται μια μικρή περιγραφή του τρόπου επίλυσης των προβλημάτων.

Το πιο σημαντικό και ουσιαστικό κομμάτι του εγγράφου αυτού είναι ο ενδελεχής έλεγχος των περιπτώσεων και υποπεριπτώσεων του προγράμματος που διεξάγεται μέσω testcases για να διαβεβαιωθεί η ορθή λειτουργία του συστήματος που υλοποιήθηκε και ότι η φάση της υλοποίησης έχει ακολουθήσει τα βήματα που προηγήθηκαν και η όλη λειτουργικότητα του προγράμματος έχει γίνει με τον τρόπο που έχει από προηγούμενως ορισθεί.

Αφού ολοκληρωθεί η φάση της υλοποίησης και του εγγράφου, θα πρέπει τα προηγούμενα παραδοτέα έγγραφα που έχουν δημιουργηθεί, να είναι όσο το δυνατόν πιο ολοκληρωμένα και κατανοητά από τον πελάτη και όχι μόνο από την ομάδα υλοποίησης του λογισμικού.

5.2. ΣΥΝΤΟΜΗ ΠΕΡΙΓΡΑΦΗ ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑΣ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ

Το υλοποιημένο σύστημα είναι το σύστημα αποθήκευσης και διαχείρισης των δεδομένων των πελατών της επιχείρησης, όπως και το σύστημα διαχείρισης των πωλήσεων. Επίσης, διαχειρίζεται τις πρώτες ύλες της επιχείρησης και τα αποθέματά τους. Οι πιο πάνω λειτουργίες υποστηρίζονται από μια βάση δεδομένων και μια διεπιφάνεια χρήστη.

Η βάση του συστήματος περιλαμβάνει τα δεδομένα των πελατών, συνεργαζόμενων εταιρειών, προϊόντων, συλλογών προϊόντων, προμηθευτών, πρώτων υλών, καθώς και τα δεδομένα των παραγγελιών, τιμολογίων κι αποδείξεων της κάθε παραγγελίας και καταστάσεων πληρωμών του κάθε πελάτη. Γενικά είναι υπεύθυνη για την διεκπεραίωση των διεργασιών που αφορούν τα πιο πάνω συστήματα διαχείρισης.

Το σύστημα αποθήκευσης και διαχείρισης δεδομένων των πελατών και των συνεργαζόμενων εταιρειών της επιχείρησης είναι υπεύθυνο για την εισαγωγή στοιχείων πελατών και συνεργαζόμενων εταιρειών, την ενημέρωσή τους σε τυχόν αλλαγές, την διαγραφή τους και την παρουσίασή τους στο χρήστη του συστήματος. Η λίστα με τις εορτές που φυλάσσεται στη βάση δεδομένων του συστήματος δίνει την ευχέρεια εύκολου και γρήγορου εντοπισμού της εορτής ενός πελάτη για την μετέπειτα αποστολή ηλεκτρονικής ευχετήριας κάρτας.

Το σύστημα διαχείρισης των πρώτων υλών της επιχείρησης είναι υπεύθυνο για την εισαγωγή στοιχείων πρώτων υλών και προμηθευτών, την ενημέρωσή τους σε τυχόν αλλαγές, την διαγραφή τους και την παρουσίασή τους στο χρήστη του συστήματος. Η αποθήκευση ενός κατώτερου ορίου αποθέματος για κάθε πρώτη ύλη βοηθά στην ανά πάσα στιγμή ενημέρωση του χρήστη για την κατάσταση των πρώτων υλών της επιχείρησης.

Το σύστημα διαχείρισης των προϊόντων της επιχείρησης είναι υπεύθυνο για την εισαγωγή στοιχείων προϊόντων και συλλογών προϊόντων της επιχείρησης, την ενημέρωσή τους σε τυχόν αλλαγές, τη διαγραφή τους και την παρουσίασή τους στο χρήστη του συστήματος. Υπάρχει η δυνατότητα προσθήκης (και τροποποίησης) των κατασκευαστικών λεπτομερειών σε κάθε προϊόν.

Το σύστημα διαχείρισης των παραγγελιών και των πωλήσεων είναι υπεύθυνο για την εισαγωγή λιανικής και χονδρικής παραγγελίας, την ενημέρωση και εμφάνισή της, καθώς επίσης και την ετοιμασία τιμολογίου και απόδειξης. Η κάθε παραγγελία (είτε χονδρική, είτε λιανική) ανά πάσα στιγμή μπορεί να κατατάσσεται σε μόνο μία κατάσταση: απλήρωτη, εκδόθηκε τιμολόγιο, εξοφλημένη, ακυρωμένη. Με αυτό τον τρόπο, διευκολύνεται ο εντοπισμός των εκκρεμούντων παραγγελιών τόσο από τον χρήστη όσο και από το σύστημα μιας και στην κύρια διεπιφάνεια χρήστη παρουσιάζεται η λίστα με τις τρέχουσες, ανολοκλήρωτες παραγγελίες.

Το όλο σύστημα υποστηρίζει εξειδικευμένες λειτουργίες, όπως την εισαγωγή και την εξαγωγή των δεδομένων των πελατών και των συνεργαζόμενων εταιρειών σε αρχείο τιμών διαχωρισμένες με κόμμα (csvfile–comma-separatedvaluefile) για ανέβασμά τους σε κινητό τηλέφωνο. Επίσης υποστηρίζει την αυτοματοποιημένη αποστολή ευχετήριας κάρτας στους πελάτες που έχουν γιορτή ή γενέθλια την τρέχουσα ημέρα μέσω ηλεκτρονικού ταχυδρομείου. Επιπλέον, ο χρήστης μπορεί να διεξάγει μια σύνθετη αναζήτηση στις συλλογές των προϊόντων της επιχείρησης για να βρει προϊόντα που πολύ πιθανόν να ενδιαφέρουν μια συνεργαζόμενη εταιρεία (λόγω προηγούμενων παραγγελιών).

Τέλος, οποιαδήποτε δεδομένα τα οποία είναι βοηθητικά για λειτουργίες και υπολογισμούς της βάσης δεδομένων φυλάσσονται στη βάση δεδομένων, έτσι ώστε να αποφεύγεται η επανάληψη και η συνεχής πληκτρολόγησή τους από τον χρήστη του συστήματος.

5.3. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΟΛΟΓΙΕΣ

Κατά την ανάπτυξη ενός λογισμικού συστήματος, σημαντικό ρόλο παίζουν τα εργαλεία με τα οποία οι απαιτήσεις και οι προδιαγραφές που θέτει ο πελάτης μπορούν να υλοποιηθούν. Όσο πιο τεχνολογικά ανεπτυγμένα είναι αυτά τα εργαλεία, τόσο μεγαλύτερη ευχρηστία και λειτουργικότητα παρέχουν στους κατασκευαστές του συστήματος. Για αυτό, η επιλογή των κατάλληλων εργαλείων για ανάπτυξη ενός λογισμικού συστήματος, παίζει καθοριστικό ρόλο στο βαθμό που θα επιτευχθούν οι στόχοι που θέτονται, στην μελλοντική ανάπτυξη του συστήματος, αλλά και στην ευκολία υλοποίησης του όλου συστήματος.

Πριν την υλοποίηση του συστήματος, μελετήθηκαν προσεκτικά τα διάφορα εργαλεία ανάπτυξης εφαρμογών λογισμικών για την επιλογή των πιο λειτουργικών, εύχρηστων αλλά και κατάλληλων εργαλείων για την υλοποίηση του συστήματος.

Η έρευνα υπέδειξε την χρησιμοποίηση της γλώσσας προγραμματισμού ως εργαλείου δημιουργίας διεπιφανειών χρήστη *Microsoft Visual Basic .NET*, μιας και παρέχει γραφικό περιβάλλον χρήστη. Το περιβάλλον αυτό διευκολύνει σε μεγάλο βαθμό το χρήστη, παρέχοντας του φόρμες οθόνης, χρήση ποντικιού, εμφάνιση βοήθειας και εικόνων. Για προγραμματισμό της βάσης του συστήματος επιλέχθηκε χρήση του *Microsoft SQL Server® 2008*.

Με την χρησιμοποίηση αυτό των εργαλείων, οι στόχοι του πελάτη έχουν επιτευχθεί και το παρόν λογισμικό θα μπορεί να συντηρηθεί και μετατραπεί εύκολα, αν το απαιτήσουν μελλοντικές ανάγκες έτσι ώστε να συμβαδίζει πάντα με τις ανάγκες του πελάτη.

Πιο κάτω δίνεται η περιγραφή των εργαλείων ανάπτυξης του συστήματος, γίνονται αναφορές στις δυσκολίες ανάπτυξης του πηγαίου κώδικα για συγκεκριμένες διεργασίες και οι τρόποι με τους οποίους αντιμετωπίστηκαν οι δυσκολίες αυτές, όπως και κομμάτια πηγαίου κώδικα τα οποία θεωρούνται άξια αναφοράς μαζί με εκτενή περιγραφή. Επίσης, παρουσιάζονται και δεδομένα ελέγχου για κάθε λειτουργία του συστήματος τα οποία χρησιμοποιήθηκαν για να ελεγχθεί η ορθότητα των λειτουργιών του συστήματος.

5.3.1. *Microsoft Visual Basic .NET*

Η όλη ανάπτυξη του συστήματος βασίστηκε σε αυτή τη γλώσσα προγραμματισμού. Η *Visual Basic .NET*, γνωστή και ως *Visual Basic*, είναι γλώσσα προγραμματισμού τρίτης γενιάς, οδηγούμενη από συμβάντα (eventdriven) και έχει ολοκληρωμένο περιβάλλον ανάπτυξης από τη *Microsoft* για το μοντέλο προγραμματισμού COM (Component Object Model). Η *Visual Basic* θεωρείται επίσης μία σχετικά εύκολη γλώσσα προγραμματισμού στην εκμάθηση και στη χρησιμοποίησή της, λόγω των χαρακτηριστικών της, καθώς έχει γραφικό περιβάλλον χρήστη. Η *Visual Basic* επιτρέπει:

- την ταχεία ανάπτυξη εφαρμογών (RAD - Rapid Application Development) με γραφικό περιβάλλον χρήστη (GUI - Graphical User Interface)
- πρόσβαση σε βάσεις δεδομένων χρησιμοποιώντας αντικείμενα (Data Access Objects, Remote Data Objects ή ActiveX Data Objects)
- και τη δημιουργία στοιχείων ελέγχου ActiveX και αντικειμένων.

Προγράμματα γραμμένα σε *Visual Basic* μπορούν, επίσης, να χρησιμοποιήσουν το Windows API, αλλά κάτι τέτοιο απαιτεί δηλώσεις εξωτερικών συναρτήσεων.

Με τη *Visual Basic* είναι δυνατή η δημιουργία εκτελέσιμων (.exe) αρχείων, στοιχείων ελέγχου ActiveX ή αρχείων DLL, αλλά χρησιμοποιείται κυρίως για την ανάπτυξη εφαρμογών για τα Windows και τη διασύνδεση συστημάτων βάσεων δεδομένων. Πλαίσια διαλόγου με λιγότερες λειτουργίες μπορούν

να χρησιμοποιηθούν για pop-up δυνατότητες. Τα στοιχεία ελέγχου παρέχουν τις βασικές λειτουργίες της εφαρμογής, ενώ οι προγραμματιστές μπορούν να εισαγάγουν επιπλέον λογική μέσα στο κατάλληλο χειριστή γεγονότων. Για παράδειγμα, ένα πτυσσόμενο πλαίσιο θα εμφανίζει αυτόματα μια λίστα που θα επιτρέπει στο χρήστη να επιλέξει οποιοδήποτε στοιχείο. Ένας χειριστής γεγονότων καλείται όταν ένα αντικείμενο είναι επιλεγμένο και στη συνέχεια μπορεί να εκτελεστεί πρόσθετος κώδικας που δημιουργείται από τον προγραμματιστή για να εκτελεστεί κάποια ενέργεια που βασίζεται στο στοιχείο που έχει επιλεγεί.

5.3.2. Microsoft SQL Server® 2008

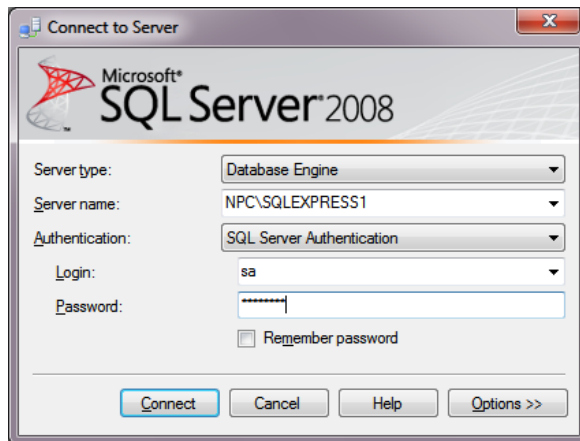
Ο *Microsoft SQL Server* είναι μια σχεσιακή βάση δεδομένων, η οποία αναπτύσσεται από τη Microsoft. Η κύρια μονάδα αποθήκευσης στοιχείων είναι μια βάση δεδομένων, η οποία αποτελείται από μια συλλογή πινάκων και κώδικα. Η κεντρική βάση δεδομένων του *SQL* υποστηρίζει διάφορους τύπους, συμπεριλαμβανομένων των ακεραίων αριθμών, αριθμών κινητής υποδιαστολής, δεκαδικών, αλφαριθμητικών, *varchar* (σειρές χαρακτήρων μεταβλητού μήκους), δυαδικών αριθμών (για τα μη δομημένα δεδομένα), κειμένων (για κείμενα). Η επερώτηση (*query*) είναι ο κύριος τρόπος για την ανάκτηση στοιχείων από μια βάση δεδομένων. Η επερώτηση εκφράζεται χρησιμοποιώντας μια παραλλαγή της *T-SQL*, μιας διαλέκτου *SQL* που αναπτύχθηκε από την *Microsoft* και *Sybase*. Η επερώτηση διευκρινίζει επεξηγηματικά αυτό που είναι να ανακτηθεί από την βάση δεδομένων.

Ο *SQL Server* επιτρέπει επίσης αποθηκευμένες επερωτήσεις σε μορφή διαδικασιών (*stored procedures*). Οι αποθηκευμένες διαδικασίες είναι επερωτήσεις με παραμέτρους *T-SQL*, οι οποίες αποθηκεύονται στον *SQL Server*. Συνήθως αποτελούνται από 100 ως 1000 γραμμές κώδικα *SQL*. Μπορούν να δεχτούν τιμές που στέλνονται από τον πελάτη (*client*) ως παράμετροι εισαγωγής/εισόδου και να στείλουν τα αποτελέσματα ως παραμέτρους παραγωγής/εξόδου. Μπορούν να καλέσουν συναρτήσεις (*functions*) και άλλες αποθηκευμένες διαδικασίες (*stored procedures*).

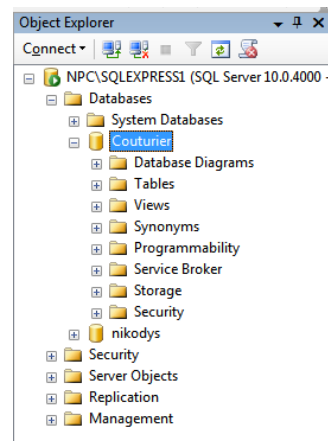
Ένα από τα δυνατά στοιχεία του *SQL Server* είναι η ύπαρξη ενός ξεχωριστού είδους αποθηκευμένων διαδικασιών σε μορφή διαδικασιών, της σκανδάλης (*trigger*). Η σκανδάλη είναι αντικείμενο της βάσης το οποίο είναι προσαρτημένο σε ένα πίνακα. Η κύρια διαφορά του με τις αποθηκευμένες διαδικασίες είναι ότι η σκανδάλη αναφέρεται μόνο σε ένα πίνακα της βάσης δεδομένων και ενεργοποιείται μόνο αν γίνει εισαγωγή, ενημέρωση ή διαγραφή στον εν λόγω πίνακα.

5.3.3. Περιγραφή βάσης δεδομένων

Η βάση δεδομένων του συστήματος δημιουργείται τοπικά στον *Microsoft SQL Server®*. Πρόσβαση στη βάση μπορεί να έχει μόνο ο διαχειριστής της βάσης. Ανοίγοντας τον *Server*, εμφανίζεται μια οθόνη που ζητάει κωδικό πρόσβασης και όνομα χρήστη (Εικόνα 5.1). Αφού δοθούν σωστά, εμφανίζεται στα δεξιά το δενδροειδές διάγραμμα όπου επιλέγεται η βάση του συστήματος (Εικόνα 5.2).



Εικόνα 5.1



Εικόνα 5.2

Οι πίνακες από τους οποίους απαρτίζεται η βάση δεδομένων περιγράφονται στην παράγραφο 3.2.6.

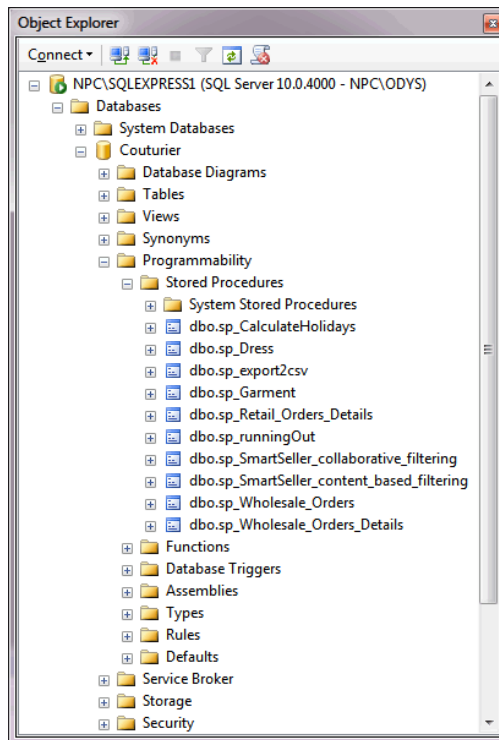
Στη βάση δεδομένων έχουν δημιουργηθεί διάφορες αποθηκευμένες επερωτήσεις σε μορφή διαδικασιών (Εικόνα 5.3) και σκανδάλες (Εικόνες 5.4 - 5.9) για καλύτερη και ευκολότερη λειτουργία του συστήματος.

Οι αποθηκευμένες διαδικασίες `sp_Dress`, `sp_Garment`, `sp_Retail_Orders_Details`, `sp_Wholesale_Orders`, `sp_Wholesale_Orders_Details` και `sp_runningOut` βοηθούν στην εμφάνιση κατανοητών και ευανάγνωστων πληροφοριών σχετικά με προϊόντα, συλλογές προϊόντων, λεπτομέρειες λιανικών παραγγελιών, χονδρικές παραγγελίες, λεπτομέρειες χονδρικών παραγγελιών και εξαντλημένων πρώτων υλών αντίστοιχα.

Η αποθηκευμένη διαδικασία `sp_CalculateHolidays` χρησιμοποιείται για την ενημέρωση του πίνακα `Namedays`, ο οποίος περιλαμβάνει όλες τις γνωστές (σταθερές και κινητές) ονομαστικές εορτές κάθε έτους. Η ενημέρωση γίνεται κάθε πρώτη του νέου έτους. Ο υπολογισμός γίνεται με τη βοήθεια των συναρτήσεων `udf_CalculateEaster` και `udf_modulo`, όπου η πρώτη υπολογίζει την ημερομηνία του Πάσχα για δεδομένο έτος και η δεύτερη το ακέραιο υπόλοιπο σε διαίρεση. (Βλέπε παράγραφο 5.4.1)

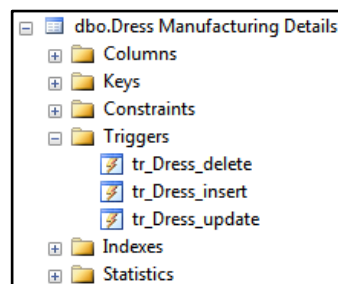
Κάνοντας χρήση της `sp_export2csv`, τα δεδομένα των πελατών εξαγονται σε αρχείο τιμών διαχωρισμένες με κόμμα (csvfile=comma-separatedvaluefile).

Οι αποθηκευμένες διαδικασίες `sp_SmartSeller_content_based_filtering` και `sp_SmartSeller_collaborative_filtering` χρησιμοποιούνται για τη δυναμική δημιουργία επερωτήσεων για σύνθετη αναζήτηση προϊόντων και την ομοιόμορφη παρουσίαση των αποτελεσμάτων τους.

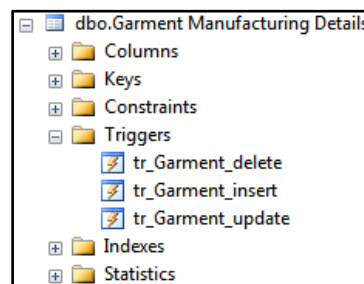


Εικόνα 5.3

Οι σκανδάλες που έχουν δημιουργηθεί για τους πίνακες *Στοιχεία κατασκευής φορέματος* (Εικόνα 5.4) και *Στοιχεία κατασκευής ειδών ένδυσης* (Εικόνα 5.5) βοηθούν στην αυτόματη ενημέρωση του αποθέματος των πρώτων υλών όταν χρησιμοποιηθούν σε κάποιο προϊόν. Έτσι η αποθήκη των πρώτων υλών είναι πάντοτε ενημερωμένη.

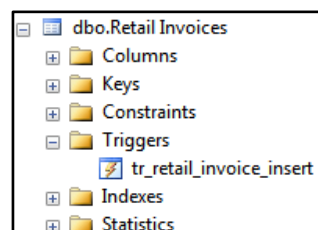


Εικόνα 5.4

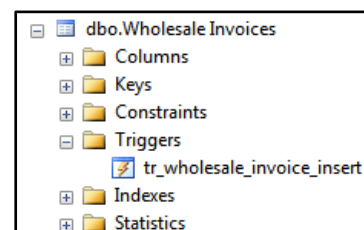


Εικόνα 5.5

Οι σκανδάλες `tr_retail_invoice_insert` (Εικόνα 5.6) και `tr_wholesale_invoice_insert` (Εικόνα 5.7) αντιγράφουν αυτόματα στους πίνακες *Αποδείξεων* πληροφορίες για τα διάφορα χρηματικά ποσά που υπολογίστηκαν στους πίνακες *Παραγγελιών*.

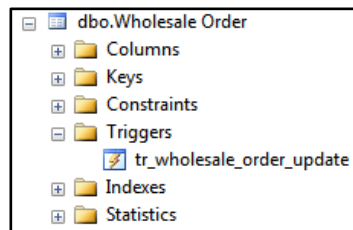


Εικόνα 5.6

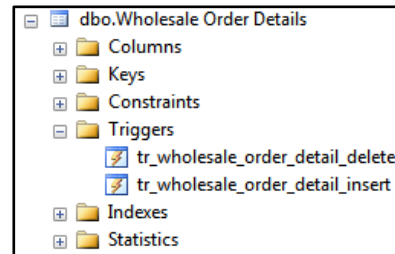


Εικόνα 5.7

Τέλος, οισκανδάλες



Εικόνα 5.8



Εικόνα 5.9

5.4. ΔΥΣΚΟΛΑ ΣΗΜΕΙΑ ΣΤΟΝ ΚΩΔΙΚΑ

5.4.1. Ενημέρωση ονομαστικών εορτών

Προς βελτίωση των σχέσεων μεταξύ της επιχείρησης και των πελατών της, θεωρήθηκε καλό να αποθηκεύονται - μαζί με τα στοιχεία του πελάτη - η ημερομηνία γενεθλίων του καθώς και η ημερομηνία της ονομαστικής του εορτής. Για εξοικονόμηση χρόνου κατά την καταχώρηση των στοιχείων, δημιουργήθηκε στη βάση δεδομένων ο πίνακας `Namedays` όπου φυλάσσονται πληροφορίες σχετικά με τις ονομαστικές εορτές του έτους. Έτσι, κατά την πληκτρολόγηση του ονόματος του πελάτη εμφανίζεται μια λίστα με πιθανές εορτές από όπου ο χρήστης μπορεί να επιλέξει.

Παρατηρήθηκε όμως το πρόβλημα ότι κάποιων πελατών η ονομαστική εορτή είναι κινητή, δηλαδή δεν εορτάζεται κάθε χρόνο την ίδια ημερομηνία. Για παράδειγμα, η γιορτή του Αγίου Γεωργίου γιορτάζεται 23 Απριλίου αν έχει προηγηθεί το Πάσχα, διαφορετικά μεταφέρεται μια μέρα μετά το Πάσχα. Επίσης ο Λάζαρος γιορτάζει τη γιορτή του το Σάββατο του Λαζάρου, δηλαδή 8 ημέρες πριν το Πάσχα, καθιστώντας την γιορτή του κινητή μιας και εξαρτάται από το Πάσχα.

Για να επιλυθεί το πιο πάνω πρόβλημα δημιουργήθηκε στη βάση δεδομένων η αποθηκευμένη διαδικασία `sp_CalculateHolidays` για την ενημέρωση του πίνακα `Namedays`. Για τον υπολογισμό του Πάσχα δημιουργήθηκαν οι συναρτήσεις `udf_CalculateEaster` και `udf_modulo`, όπου η πρώτη υπολογίζει την ημερομηνία του Πάσχα για δεδομένο έτος και η δεύτερη το ακέραιο υπόλοιπο σε διαίρεση. (Βλέπε Παράρτημα Α, Παράγραφος Α.1)

5.4.2. Αποστολή ευχετήριας κάρτας με ηλεκτρονικό ταχυδρομείο

Την ημέρα που κάποιος πελάτης έχει γενέθλια ή ονομαστική εορτή, η επιχείρηση θα ήταν σωστό όπως αποστέλλει ένα ηλεκτρονικό μήνυμα για να εύχεται στον εορτάζοντα ανάλογα με την περίπτωση. Το να πληκτρολογείται κάθε φορά το ίδιο κείμενο σε ηλεκτρονικό μήνυμα ή να προωθείται ένα παλιό μήνυμα και να επιλέγονται ένας-ένας οι παραλήπτες κάθε φορά είναι ταλαιπωρία.

Για αυτό το λόγο δημιουργήθηκε στη βάση δεδομένων ο βοηθητικός πίνακας `Settings` όπου αποθηκεύονται οποιαδήποτε δεδομένα τα οποία είναι βοηθητικά για λειτουργίες και υπολογισμούς της επιχείρησης, με απώτερο σκοπό η αποφυγή επανάληψης και συνεχούς πληκτρολόγησής τους από τον χρήστη του συστήματος. Για την αποστολή ηλεκτρονικού ταχυδρομείου, βοηθητικά δεδομένα αποτελούν τα εξής: διεύθυνση ηλεκτρονικού ταχυδρομείου, κωδικός ηλεκτρονικού ταχυδρομείου, διακομιστής SMTP, θύρα SMTP, κείμενο ευχής γενεθλίων και κείμενο ευχής ονομαστικής εορτής.

Επιπλέον, για την αυτοματοποίηση της λειτουργίας αποστολής ηλεκτρονικού μηνύματος, δημιουργήθηκε στην `VisualBasic .NET` η διαδικασία `sendmail()` η οποία υποβοηθάται από τις διαδικασίες `Today_Birthdays()` και `Today_Namedays()`. (Βλέπε Παράρτημα Α, Παράγραφος Α.2)

5.4.3. Ενημέρωση αποθήκης πρώτων υλών και αυτόματη προειδοποίηση εξάντλησης πρώτης ύλης

Κατά τη δημιουργία ενός νέου προϊόντος, χρησιμοποιείται κάποια ποσότητα από τις πρώτες ύλες της αποθήκης. Η αποθήκη της επιχείρησης πρέπει να ενημερώνεται αυτόματα για το απόθεμα των πρώτων υλών κατά τη χρήση τους. Επιπλέον, αν το σύστημα μπορούσε να προειδοποιεί τον χρήστη ότι κάποια πρώτη ύλη έχει σχεδόν εξαντληθεί θα ήταν πολύ βοηθητικό, μιας και δεν θα χρειαζόταν η καθημερινή καταγραφή των αποθεμάτων της αποθήκης.

Στην επίλυση του προβλήματος αυτού, βοήθησαν πάρα πολύ οι σκανδάλες που προσφέρει ο `SQL Server`. Έτσι δημιουργήθηκαν οι σκανδάλες `tr_Dress_insert`, `tr_Dress_update`, `tr_Dress_delete`, `tr_Garment_insert`, `tr_Garment_update` και `tr_Garment_delete` (Εικόνες

6.23 - 6.24), δημιουργήθηκε στην *VisualBasic .NET* η διαδικασία `Running_Out()` η οποία βρίσκει τις πρώτες ύλες που εξαντλήθηκαν και τις εμφανίζει στη φόρμα αποθήκης στο επιπλέον πλαίσιο που προστέθηκε. Κατά την ενεργοποίησή τους οι σκανδάλες αυξομειώνουν την ποσότητα των πρώτων υλών στον πίνακα Πρώτων Υλών, κρατώντας με αυτό τον τρόπο πάντα ενήμερη την αποθήκη. (Βλέπε Παράρτημα Α, Παράγραφος Α.3)

5.4.4. Αποθήκευση εικόνων στη βάση δεδομένων

Στην αρχική έκδοση της βάσης δεδομένων, για την αποθήκευση των εικόνων είχε οριστεί η αποθήκευση στη βάση δεδομένων της διεύθυνσης της εικόνας στο σκληρό δίσκο ως `nvarchar(250)`. Παρουσιάστηκε όμως πρόβλημα όταν μετακινήθηκαν τα αρχεία των εικόνων σε νέα διεύθυνση στο σκληρό δίσκο. Έπρεπε όμως οι συνδέσεις στη βάση δεδομένων να ενημερωθούν μία προς μία με τη νέα διεύθυνσή των εικόνων. Επίσης, οι συνδέσεις τους με τη βάση δεδομένων έπαυαν να ισχύουν εάν τα αρχεία είχαν αλλοιωθεί ή καταστραφεί. Παρατηρήθηκε επίσης, ότι αυτή η μέθοδος υποστηρίζει περιορισμένο αριθμό τύπων αρχείων.

Το πρόβλημα επιλύθηκε όταν άλλαξε ο τρόπος αποθήκευσης των εικόνων στη βάση δεδομένων. Συγκεκριμένα, πλέον δεν αποθηκεύεται η διεύθυνση της εικόνας στη βάση δεδομένων, αλλά η ίδια η εικόνα σε μορφή `bitarray`. Δηλαδή ο τύπος αποθήκευσης της εικόνας άλλαξε από `nvarchar(250)` σε `image`. Αυτή η μέθοδος είναι απλούστερη στην υλοποίηση. Επίσης, οι εικόνες γίνονται τμήμα της βάσης δεδομένων και μεταφέρονται μαζί με τη βάση. Δεν χρειάζεται να γίνεται ενημέρωση των συνδέσεων των αρχείων εικόνων με τη βάση δεδομένων. Για την αποθήκευση στη βάση δεδομένων δημιουργήθηκε η συνάρτηση `ConvertImage()` η οποία μετατρέπει την εικόνα σε `bitarray`. (Βλέπε Παράρτημα Α, Παράγραφος Α.4)

5.4.5. Εκτύπωση αποδείξεων και τιμολογίων

Μία από τις απαιτήσεις του συστήματος ήταν η εκτύπωση τιμολογίων και αποδείξεων με τυποποιημένη μορφή. Ζητήθηκε να υπάρχει η δυνατότητα επέμβασης στο εκτυπώσιμο έγγραφο για τυχόν τροποποιήσεις και αποθήκευσή του. Τα εργαλεία που προσφέρει η *VisualBasic* δεν ικανοποιούν τις ανάγκες αυτές, μιας και εμφανίζει μόνο μια προεπισκόπηση του εκτυπώσιμου η οποία δεν επιτρέπεται να τροποποιηθεί ή να αποθηκευτεί.

Λύση στο πρόβλημα έδωσε η *Microsoft Word* του *Microsoft Office* και ο ονοματοχώρος (namespace) `Microsoft.Office.Interop.VisualBasic`. Τα έγγραφα τυποποιημένης μορφής (templates) και οι σελιδοδείκτες (bookmarks) που προσφέρει η *Word*, βοήθησαν στη δημιουργία ενός ικανοποιητικού και εμφανίσιμου αποτελέσματος. Αφού δημιουργήθηκε η όλη εμφάνιση του εγγράφου, έπρεπε να βρεθεί τρόπος να τοποθετούνται όλες οι πληροφορίες στο έγγραφο. Έτσι προστέθηκαν σελιδοδείκτες στο έγγραφο και μέσω της *VisualBasic* τοποθετούνται οι πληροφορίες στο σωστό σημείο του εγγράφου. Παράδειγμα του κώδικα που χρησιμοποιήθηκε υπάρχει στο Παράρτημα Α, Παράγραφος Α.5.

5.4.6. Εκτύπωση ποσού ολογράφως

Για λόγους ασφαλείας, αποφασίστηκε η εκτύπωση του εισπραττόμενου ποσού ολογράφως. Η *VisualBasic* δεν έχει συναρτήσεις ή μεθόδους που να υλοποιούν κάτι τέτοιο. Έτσι υλοποιήθηκε η συνάρτηση `NumberToString()` η οποία παίρνει ένα αριθμό και με τη βοήθεια της `GroupToWords()` μετατρέπει τον αριθμό σε λέξεις. Η πρώτη συνάρτηση χωρίζει τον αριθμό σε ομάδες των τριών ψηφίων και η δεύτερη εκτυπώνει την κατάλληλη λέξη για τον τριψήφιο αριθμό. Ο κώδικας των συναρτήσεων φαίνεται στο Παράρτημα Α, παράγραφος Α.6.

5.4.7. Εισηγήσεις προϊόντων

Για καλύτερη και αποτελεσματικότερη εξυπηρέτηση της επιχείρησης, ρωτήθηκε αν ήταν δυνατή η προσθήκη ενός εργαλείου σύνθετης αναζήτησης προϊόντων της οποίας τα αποτελέσματα θα πήγάζαν από φιλτράρισμα των εγγραφών της βάσης δεδομένων. Τα κριτήρια που θα καθορίζονται στην αναζήτηση να μην είναι προκαθορισμένα, αλλά να υπάρχει μια λίστα από όπου μπορεί ο χρήστης να επιλέξει. Επίσης ρωτήθηκε αν ήταν δυνατή η εισήγηση προϊόντων κατά τη διεξαγωγή παραγγελίας, έτσι ώστε να διαφανεί τι προϊόντα παρήγγειλαν άλλοι χρήστες μαζί με ένα συγκριμένο προϊόν.

Για το σκοπό αυτό, γράφτηκαν δύο διαδικασίες στον SQLServer, οι οποίες υλοποιούν τους αλγόριθμους σύστασης, φιλτράρισμα με βάση το περιεχόμενο και συνεργατικό φιλτράρισμα.

5.4.7.1. Φιλτράρισμα με βάση το περιεχόμενο

Για τη δημιουργία του εργαλείου σύνθετης αναζήτησης προϊόντων και την υλοποίηση του φιλτραρίσματος με βάση το περιεχόμενο, δημιουργήθηκε η διαδικασία στον SQLServer `sp_SmartSeller_content_based_filtering`.

Η διαδικασία `sp_SmartSeller_content_based_filtering` παίρνει σαν παραμέτρους τον κωδικό του πελάτη, τις συλλογές προϊόντων από τις οποίες θα αναζητήσει προϊόντα, τις κατηγορίες των προϊόντων που θα αναζητήσει και τα επιθυμητά χρώματα. Χρησιμοποιώντας τις παραμέτρους αυτές, δημιουργεί δυναμικά μία επερώτηση για την εξαγωγή των ζητούμενων δεδομένων. Τα αποτελέσματα της επερώτησης αποθηκεύονται σε ένα πίνακα όψης (viewtable) για την περαιτέρω επεξεργασία τους στην *VisualBasic*.

Η γενική ερώτηση που εκφράζει η δυναμική επερώτηση που δημιουργείται είναι η εξής:

Βρες τα προϊόντα τα οποία ανήκουν σε δεδομένες συλλογές προϊόντων, ανήκουν σε δεδομένες κατηγορίες προϊόντων και έχουν δεδομένο χρώμα.

Ο ψευδοκώδικας που χρησιμοποιήθηκε για την επίλυση του προβλήματος αυτού είναι:

```

1. Σ[] ← Συλλογές προϊόντων
Κ[] ← κατηγορίες προϊόντων
Χ[] ← Χρώματα
Π[] ← Προϊόντα συστήματος
2. Για κάθε π στο Π[] επανέλαβε
    Για κάθε σ στο Σ[] επανέλαβε
        Αν π.Συλλογή = σ Τότε
            Πρόσθεσε (Π1[], π)
3. Για κάθε π στο Π1[] επανέλαβε
    Για κάθε κ στο Κ[] επανέλαβε
        Αν π.Κατηγορία = κ Τότε
            Πρόσθεσε (Π2[], π)
4. Για κάθε π στο Π2[] επανέλαβε
    Για κάθε χ στο Χ[] επανέλαβε
        Αν π.Χρώμα = χ Τότε
            Πρόσθεσε (Π3[], π)

```

Ο κώδικας της διαδικασίας φαίνεται στο Παράρτημα Α, παράγραφος Α.7.1.

5.4.7.2. Συνεργατικό φιλτράρισμα

Για τη δημιουργία του εργαλείου εισηγήσεων προϊόντων κατά τη διεξαγωγή παραγγελίας, έτσι ώστε να διαφανεί τι προϊόντα παρήγγειλαν άλλοι χρήστες μαζί με ένα συγκριμένο προϊόν και την υλοποίηση του συνεργατικού φιλτραρίσματος, δημιουργήθηκε η διαδικασία στον `SQLServersp_SmartSeller_collaborative_filtering`.

Η διαδικασία `sp_SmartSeller_collaborative_filtering` παίρνει ως παράμετρο τον κωδικό ενός προϊόντος. Χρησιμοποιώντας την παράμετρο αυτή, ψάχνει να βρει τις παραγγελίες που περιέχουν το προϊόν και να συγκεντρώσει τα υπόλοιπα προϊόντα που αγοράστηκαν μαζί με το αρχικό προϊόν. Τα αποτελέσματα της επερώτησης αποθηκεύονται σε ένα πίνακα όψης (viewtable) για την περαιτέρω επεξεργασία τους στην *VisualBasic*.

Η γενική ερώτηση που εκφράζει η δυναμική επερώτηση που δημιουργείται είναι η εξής:

Βρες τα προϊόντα τα οποία παραγγέλθηκαν μαζί με το δεδομένο προϊόν.

Ο ψευδοκώδικας που χρησιμοποιήθηκε για την επίλυση του προβλήματος αυτού είναι:

```

1. π = Προϊόν
2. O[] ← Παραγγελίες συστήματος
3. Για κάθε ο στο O[] επανέλαβε
    Για κάθε οπ στο ο.Προϊοντα[] επανέλαβε
        Αν οπ = π τότε
            Πρόσθεσε (O1[], ο)
4. Για κάθε ο στο O1[] επανέλαβε
    Για κάθε οπ στο ο.Προϊοντα[] επανέλαβε
        Πρόσθεσε (Π[], οπ)

```

Ο κώδικας της διαδικασίας φαίνεται στο Παράρτημα Α, παράγραφος Α.7.2.

5.4.8. Εισαγωγή/εξαγωγή δεδομένων

Έγινε εισήγηση όπως προστεθεί στο σύστημα η δυνατότητα εξαγωγής των δεδομένων των πελατών σε αρχείο για την φόρτωση τους σε κινητό τηλέφωνο και η εισαγωγή δεδομένων από αρχείο του ίδιου του προγράμματος ή κάποιου άλλου. Τα δεδομένα μπορούν να είναι συνδυασμό των εξής: ονοματεπώνυμο πελάτη, ταχυδρομική διεύθυνση, τηλέφωνα επικοινωνίας, ηλεκτρονική διεύθυνση και ημερομηνία γεννήσεως.

Η απαίτηση για την εξαγωγή των δεδομένων υλοποιήθηκε μέσω της αποθηκευμένης διαδικασίας `sp_export2csv` η οποία κάνει χρήση της εντολής `bcsp`. Η εντολή `bcsp` παίρνει ως ορίσματα την επερώτηση που θα εκτελεστεί και το αρχείο όπου θα αποθηκευτούν τα αποτελέσματα. Η επερώτηση δημιουργείται δυναμικά στη *VisualBasic* μέσω της συνάρτησης `createQuery()`.

Η απαίτηση για την εισαγωγή των δεδομένων έγινε αρχικά προσπάθεια να υλοποιήθηκε μέσω της εντολής `BULKINSERT` που προσφέρει ο *SQLServer*. Παρουσιάστηκε όμως πρόβλημα όταν η εγγραφή υπάρχει ήδη στη βάση δεδομένων. Αυτό συμβαίνει επειδή η `BULKINSERT` δεν ελέγχει τους περιορισμούς της βάσης δεδομένων κατά την εκτέλεσή του. Έτσι δημιουργήθηκε η συνάρτηση `ImportTextFile()` η οποία παίρνει ως παραμέτρους τη σύνδεση στη βάση δεδομένων, το αρχείο από όπου γίνεται η εισαγωγή, τον πίνακα στον οποίο γίνεται η εισαγωγή, το χαρακτήρα διαχωρισμού των εγγραφών και το χαρακτήρα διαχωρισμού των πεδίων. Για κάθε γραμμή του αρχείου, γίνεται έλεγχος αν η εγγραφή αυτή υπάρχει ήδη στον πίνακα κι αν δεν υπάρχει, γίνεται η εισαγωγή της.

Ο κώδικας και των δύο απαιτήσεων φαίνεται στο Παράρτημα Α, παράγραφος Α.8.

5.5. ΠΗΓΑΙΟΣ ΚΩΔΙΚΑΣ ΜΕ ΣΧΟΛΙΑ

5.5.1. Διαχείριση πελατών

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση, διαγραφή και αναζήτηση πελάτη. Επίσης, έχει υλοποιηθεί λειτουργία εύρεσης ονομαστικής εορτής. Κατά την προσθήκη ή ενημέρωση κάποιου πελάτη, ενώ πληκτρολογείται το όνομά του εμφανίζεται μια λίστα με τις πιθανές ονομαστικές εορτές του πελάτη. Η λίστα προκύπτει από το φιλτράρισμα των ονομαστικών εορτών που βρίσκονται αποθηκευμένες στη βάση.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO Customers(ID, [Full Name], BirthDate, Nameday, Street,
[Street No], Region, [PO Code], City, Country, [Telephone No], [Mobile
No], Email, Bust, Waist, Hips)"& _
" SELECT @id, @FullName, @BirthDate, @Nameday, @Street, @StreetNo,
@Region, @POCode, @City, @Country, @TelephoneNo, @MobileNo, @Email,
@Bust, @Waist, @Hips "& _
" WHERE @id NOT IN (SELECT ID FROM Customers)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας ενημέρωσης:

```
'update the selected record with the new given data
Dim updateSql =
"UPDATE Customers"& _
" SET [Full Name]=@FullName, BirthDate=@BirthDate, Nameday=@Nameday,
Street=@Street, [Street No]=@StreetNo, Region=@Region, [PO Code]=@POCode,
City=@City, Country=@Country, [Telephone No]=@TelephoneNo, [Mobile No]
=@MobileNo, Email=@Email, Bust=@Bust, Waist=@Waist, Hips=@Hips"& _
" WHERE ID = @id"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας διαγραφής:

```
'delete the selected record
Dim deleteSql AsString = "DELETE FROM Customers WHERE ID=@id"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας αναζήτησης:

```
'Search for records which satisfy the criteria
'Due to the numeric and string values, set two cases
If IsNumeric(searchtxtbox.Text) Then
    Dim keyword = Integer.Parse(searchtxtbox.Text)
Me.CustomersTableAdapter.FillBySearchInt(Me.CouturierDataSet.Customers,
keyword)
Else
Dim keyword = "%" + searchtxtbox.Text + "%"

Me.CustomersTableAdapter.FillBySearchString(Me.CouturierDataSet.Customers,
keyword)
EndIf

CustomersTableAdapter.FillBySearchInt(@keyword)
SELECT *
FROM Customers
WHERE ([Telephone No]=@keyword) OR (ID=@keyword) OR ([Mobile No]=@keyword)

CustomersTableAdapter.FillBySearchString(@keyword)
SELECT *
FROM Customers
WHERE ([Full Name] LIKE @keyword) OR (Email LIKE @keyword)
```

Κώδικας αναζήτησης ονομαστικής εορτής:

```
'Find the nameday for the given name
If Full_NameTextBox.Text.Equals("") Then
Me.NamedaysTableAdapter.FillByName(Me.CouturierDataSet.Namedays, "0")
Else
Dim name() AsString = Split(Full_NameTextBox.Text)
Me.NamedaysTableAdapter.FillByName(Me.CouturierDataSet.Namedays, name(0))
EndIf

NamedaysTableAdapter.FillByName(@name)
SELECT *
FROM dbo.Namedays
WHERE names LIKE '%' + @name + '%'
```

Κώδικας εύρεσης ημερήσιων εορταζόντων:

```
'Find the customers who have birthday today
Dim Sql AsString =
"SELECT [Full Name]"& _
" FROM Customers"& _
" WHERE (datepart(day, [BirthDate]) = datepart(day, @date))"& _
" AND (datepart(month, [BirthDate]) = datepart(month, @date))"

Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
sqlcomm.Parameters.Add(New SqlParameter("@date", SqlDbType.Date)).Value =
..

SqlConnection.Open()
Dim r As SqlDataReader = sqlcomm.ExecuteReader()
If r.HasRows() Then
    While r.Read()
        [...]
    End While
End If
```

```

EndWhile
    r.Close()
EndIf
SqlConn.Close()

'Find the customers who have nameday today
Dim Sql AsString =
"SELECT [Full Name]"& _
" FROM Customers C JOIN Namedays N ON C.Nameday = N.id"& _
" WHERE (datepart(day, N.[date]) = datepart(day, @date))"& _
" AND (datepart(month, N.[date]) = datepart(month, @date))"

Dim sqlcomm AsNew SqlCommand(Sql, SqlConn)
sqlcomm.Parameters.Add(New SqlParameter("@date", SqlDbType.Date)).Value =
..

SqlConn.Open()
Dim r As SqlDataReader = sqlcomm.ExecuteReader()
If r.HasRows() Then
    While r.Read()
        [...]
    EndWhile
    r.Close()
EndIf
SqlConn.Close()

```

5.5.2. Διαχείριση συνεργαζόμενων εταιρειών

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση, διαγραφή και αναζήτηση εταιρειών.

Κώδικας προσθήκης:

```

'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO [Cooperating Company](ID, [Company Name], [Boutique Name],
[Contact Person], Street, [Street No], Region, [PO Code], City,
Country, [Telephone No], [Mobile No], Fax, Email, Website)"& _
" SELECT @id, @company, @boutique, @contactPerson, @Street, @StreetNo,
@Region, @POCode, @City, @Country, @TelephoneNo, @MobileNo, @fax,
@Email, @website"& _
" WHERE @id NOT IN (SELECT ID FROM [Cooperating Company])"
Dim sqlcomm = New SqlCommand(insertSql, SqlConn)

SqlConn.Open()
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
[...]
sqlcomm.ExecuteNonQuery()
SqlConn.Close()

```

Κώδικας ενημέρωσης:

```

'update the selected record with the new given data
Dim updateSql =
"UPDATE [Cooperating Company]" & _
" SET [Company Name]=@company, [Boutique Name]=@boutique, [Contact Person]=
@contactPerson, Street=@Street, [Street No]=@StreetNo, Region=@Region, [PO

```

```

Code]= @POCode, City=@City, Country=@Country, [Telephone No]=@TelephoneNo,
[Mobile No]= @MobileNo, Fax=@fax, Email=@Email, Website=@website" & _
" WHERE ID = @id"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()

```

Κώδικας διαγραφής:

```

'delete the selected record
Dim deleteSql AsString = "DELETE FROM [Cooperating Company] WHERE ID=@id"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()

```

Κώδικας αναζήτησης:

```

'Search for records which satisfy the criteria
'Due to the numeric and string values, set two cases
If IsNumeric(searchtxtbox.Text) Then
Dim keyword = Integer.Parse(searchtxtbox.Text)
    Me.Cooperating_CompanyTableAdapter.FillBySearchInt( _
Me.CouturierDataSet.Cooperating_Company, keyword)
Else
Dim keyword = "%" + searchtxtbox.Text + "%"
Me.Cooperating_CompanyTableAdapter.FillBySearchString( _
Me.CouturierDataSet.Cooperating_Company, keyword)
EndIf

Cooperating_CompanyTableAdapter.FillBySearchInt(@keyword)
SELECT *
FROM [Cooperating Company]
WHERE (ID = @keyword) OR ([Telephone No] = @keyword) OR
      ([Mobile No] = @keyword) OR (Fax = @keyword)

Cooperating_CompanyTableAdapter.FillBySearchString(@keyword)
SELECT *
FROM [Cooperating Company]
WHERE ([Company Name] LIKE @keyword) OR
      ([Boutique Name] LIKE @keyword) OR
      ([Contact person] LIKE @keyword) OR
      (Email LIKE @keyword) OR (Website LIKE @keyword)

```

5.5.3. Διαχείριση πρώτων υλών

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση, διαγραφή και αναζήτηση πρώτης ύλης. Επίσης, κατά την προσθήκη ή ενημέρωση μιας πρώτης ύλης μπορεί να γίνει προσθήκη χρωματικής επιλογής, καθώς και νέου προμηθευτή.

Υπάρχει λειτουργία προειδοποίησης εξαντλημένων πρώτων υλών. Κατά την είσοδο στο σύστημα, εμφανίζεται προειδοποίηση αν κάποια πρώτη ύλη έχει εξαντληθεί. Με το άνοιγμα της φόρμας των πρώτων υλών εμφανίζεται η λίστα με αυτές τις πρώτες ύλες.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO [Raw Material](Code, [Description], Texture, Color, Notes,
Cost, Supplier, Stock, [Min quantity], [Measurement Unit])"& _
" SELECT @Code, @description, @texture, @color, @notes, @cost,
@supplier, @stock, @min, @unit"& _
" WHERE @Code NOT IN (SELECT Code FROM [Raw Material])"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας ενημέρωσης:

```
'update the selected record with the new given data
Dim updateSql =
"UPDATE [Raw Material]" & _
" SET Description=@description, Texture=@texture, Color=@color,
Width=@width, Cost=@cost, Supplier=@supplier, Stock=@stock, [Min
quantity]=@min, [Measurement Unit]=@unit" & _
" WHERE Code = @Code"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας διαγραφής:

```
'delete the selected record
Dim deleteSql AsString = "DELETE FROM [Raw Material] WHERE Code=@code"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας αναζήτησης:

```
'Search for records which satisfy the criteria
Dim keyword = "%" + searchtxtbox.Text + "%"
Me.Raw_MaterialTableAdapter.FillBySearch(Me.CouturierDataSet.Raw_Material,
keyword)

Raw_MaterialTableAdapter.FillBySearch(@keyword)
SELECT *
FROM [Raw Material]
```

```
WHERE (Code LIKE @keyword) OR (Description LIKE @keyword) OR
(Texture LIKE @keyword)
```

Κώδικας προσθήκης χρωματικών επιλογών:

```
'insert into database a new row with the given data
Dim insertSql = "INSERT INTO Color(Code, Color)"& _
" SELECT @code, @color"& _
" WHERE @code NOT IN (SELECT Code FROM Color)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
sqlcomm.Parameters.AddWithValue("@Color", CodeTextBox.Text)
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας προειδοποίησης εξαντλημένων πρώτων υλών:

```
'Find out the raw materials whose stock is less than the minimum quantity
SqlConnection.Open()
Dim Sql = "SELECT RM.Code, RM.Description, RM.Stock, RM.[Measurement Unit]"& _
" FROM [Raw Material] RM JOIN Supplier S ON RM.Supplier = S.ID"& _
" WHERE Stock <= [Min quantity]"
Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
Dim r As SqlDataReader = sqlcomm.ExecuteReader()

If r.HasRows() Then
    ErrorProvider1.SetError("Raw materials are running out")

While r.Read()
    [...]
EndWhile
    r.Close()
EndIf
SqlConnection.Close()
```

Υλοποιήθηκαν επίσης η προσθήκη, ενημέρωση, διαγραφή και αναζήτηση προμηθευτών.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO Supplier(ID,[Supplier Name],[Contact
person],Street,[Street No],Region,[PO Code],City,Country,[Telephone
No],[Mobile No],Fax,Email)"& _
" SELECT @id, @supplier, @contactPerson, @Street, @StreetNo, @Region,
@POCode, @City, @Country, @TelephoneNo, @MobileNo, @fax, @Email"& _
" WHERE @id NOT IN (SELECT ID FROM Supplier)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας ενημέρωσης:

```

'update the selected record with the new given data
Dim updateSql =
"UPDATE Supplier" & _
" SET [Supplier Name]=@supplier, [Contact Person]=@contactPerson,
Street=@Street, [Street No]=@StreetNo, Region=@Region, [PO Code]=@POCode,
City=@City, Country= @Country, [Telephone No]=@TelephoneNo, [Mobile
No]=@MobileNo, Fax=@fax, Email= @Email" & _
" WHERE ID = @id"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()

```

Κώδικας διαγραφής:

```

'delete the selected record
Dim deleteSql AsString = "DELETE FROM Supplier WHERE ID=@id"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.Add(New SqlParameter("@id", SqlDbType.Int)).Value = ..
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()

```

Κώδικας αναζήτησης:

```

'Search for records which satisfy the criteria
'Due to the numeric and string values, set two cases
If IsNumeric(searchtxtbox.Text) Then
Dim keyword = Integer.Parse(searchtxtbox.Text)
Me.SupplierTableAdapter.FillBySearchInt(Me.CouturierDataSet.Supplier,
keyword)
Else
Dim keyword = "%" + searchtxtbox.Text + "%"
Me.SupplierTableAdapter.FillBySearchString(Me.CouturierDataSet.Supplier,
keyword)
EndIf

```

SupplierTableAdapter.FillBySearchInt(@keyword)

```

SELECT *
FROM Supplier
WHERE (ID = @keyword) OR ([Telephone No] = @keyword) OR
      ([Mobile No] = @keyword) OR (Fax = @keyword)

```

SupplierTableAdapter.FillBySearchString(@keyword)

```

SELECT *
FROM Supplier
WHERE ([Supplier Name] LIKE @keyword) OR
      ([Contact person] LIKE @keyword) OR
      (Email LIKE @keyword)

```

5.5.4. Διαχείριση προϊόντων

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση, διαγραφή και αναζήτηση προϊόντων. Επίσης, κατά την προσθήκη ή ενημέρωση ενός προϊόντος μπορεί να γίνει προσθήκη χρωματικής επιλογής, καθώς και προσθήκη ή διαγραφή υλικών κατασκευής του προϊόντος.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO Dress(Code)"& _
" SELECT @code"& _
" WHERE @code NOT IN (SELECT Code FROM Dress)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας ενημέρωσης:

```
'update the selected record with the new given data
Dim updateSql =
"UPDATE Dress" & _
" SET Description=@description, Size=@size, Color=@color, [Total
Cost]=@totalcost, [Retail Price]=@retailprice, Image=@image" & _
" WHERE Code LIKE @Code"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας διαγραφής:

```
'delete the selected record
Dim deleteSql AsString = "DELETE FROM Dress WHERE Code=@code"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας αναζήτησης:

```
'Search for records which satisfy the criteria
Dim keyword = "%" + searchtxtbox.Text + "%"
Me.DressTableAdapter.FillBySearch(Me.CouturierDataSet.Dress, keyword)

DressTableAdapter.FillBySearch(@keyword)
SELECT *
FROM Dress
WHERE (Code LIKE @keyword) OR (Description LIKE @keyword)
```

Κώδικας προσθήκης χρωματικών επιλογών:

```
'insert into database a new row with the given data
Dim insertSql = "INSERT INTO Color(Code, Color)"& _
" SELECT @code, @color"& _
" WHERE @code NOT IN (SELECT Code FROM Color)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
sqlcomm.Parameters.AddWithValue("@Color", CodeTextBox.Text)
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας προσθήκης υλικών κατασκευής:

```
'insert into database a new row with the given data
Dim insertSql = "INSERT INTO [Dress Manufacturing Details]"& _
" VALUES(@DressCode, @RawMaterialCode, @quantity)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@DressCode", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας διαγραφής:

```
'delete the selected record
Dim deleteSql =
"DELETE FROM [Dress Manufacturing Details]" & _
" WHERE [Dress Code]=@DressCode AND [Raw Material Code]=@RawMaterialCode"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.AddWithValue("@DressCode", CodeTextBox.Text)
[...]
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

5.5.5. Διαχείριση προϊόντων συλλογών

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση, διαγραφή και αναζήτηση προϊόντων συλλογών. Επίσης, κατά την προσθήκη ή ενημέρωση ενός προϊόντος μπορεί να γίνει προσθήκη χρωματικής επιλογής, καθώς και προσθήκη ή διαγραφή υλικών κατασκευής του προϊόντος. Δίνεται η δυνατότητα δημιουργίας συλλογής προϊόντων.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO Garment(Code, Collection)"& _
" SELECT @code, @collection"& _
" WHERE @code NOT IN (SELECT Code FROM Garment)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
```

```
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας ενημέρωσης:

```
'update the selected record with the new given data
Dim updateSql =
"UPDATE Garment" & _
" SET Description=@description, Size=@size, Color=@color, [Total
Cost]=@totalcost, [Wholesale Price]=@wholesaleprice, Image=@image" & _
" WHERE Code LIKE @Code"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας διαγραφής:

```
'delete the selected record
Dim deleteSql AsString = "DELETE FROM Garment WHERE Code=@code"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας αναζήτησης:

```
'Search for records which satisfy the criteria
Dim keyword = "%" + searchtxtbox.Text + "%"
Me.GarmentTableAdapter.FillBySearch(Me.CouturierDataSet.Garment, keyword)

GarmentTableAdapter.FillBySearch(@keyword)
SELECT *
FROM Garment
WHERE (Code LIKE @keyword) OR (Description LIKE @keyword)

GarmentTableAdapter.FillByCollection(@collection)
SELECT *
FROM Garment
WHERE (Collection = @collection)
```

Κώδικας προσθήκης χρωματικών επιλογών:

```
'insert into database a new row with the given data
Dim insertSql = "INSERT INTO Color(Code, Color)"& _
" SELECT @code, @color"& _
" WHERE @code NOT IN (SELECT Code FROM Color)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@Code", CodeTextBox.Text)
sqlcomm.Parameters.AddWithValue("@Color", CodeTextBox.Text)
sqlcomm.ExecuteNonQuery()
```

```
SqlConnection.Close()
```

Κώδικας προσθήκης υλικών κατασκευής:

```
'insert into database a new row with the given data
Dim insertSql = "INSERT INTO [Garment Manufacturing Details]" & _
" VALUES(@GarmentCode, @RawMaterialCode, @quantity)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@GarmentCode", CodeTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας διαγραφής υλικών κατασκευής:

```
'delete the selected record
Dim deleteSql =
"DELETE FROM [Garment Manufacturing Details]" & _
" WHERE [Garment Code]=@GarmentCode AND [Raw Material Code]=@RawMaterialCode"
Dim sqlcomm AsNew SqlCommand(deleteSql, SqlConnection)
sqlcomm.Parameters.AddWithValue("@DressCode", CodeTextBox.Text)
[...]
SqlConnection.Open()
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας δημιουργίας συλλογής ειδών ένδυσης:

```
'Create a new garment collection
Dim insertSql = "INSERT INTO Collections" & _
" VALUES(@name, @season, @description)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@name", NameTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

5.5.6. Διαχείριση λιανικών πωλήσεων

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση και αναζήτηση λιανικών πωλήσεων. Η αναζήτηση μπορεί να γίνει κατά χρονική περίοδο, κατάσταση παραγγελίας, ονοματεπώνυμο πελάτη ή στοιχεία προϊόντος. Επίσης, δίνεται η δυνατότητα δημιουργίας και εκτύπωσης τιμολογίου και απόδειξης είσπραξης.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO [Retail Order]" & _
" VALUES(@id, @customer, @date, @duedate, @dress, @quantity, @price, @subtotal, @vat, @discountper, @discount, @total, @notes, @status)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)
```

```

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@id", OrderIDTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()

```

Κώδικας ενημέρωσης:

```

'update the selected record with the new given data
Dim updateSql =
"UPDATE [Retail Order]" & _
" SET [Customer ID]=@customer, [Order Date]=@date, [Due Date]=@duedate,
[Dress Code]=@dress, Quantity=@quantity, [Price per piece]=@price,
Subtotal=@subtotal, [VAT total]=@vat, [Discount (%)]=@discountper,
Discount=@discount, Total=@total, Notes=@notes, [Order Status]=@status" &
_
" WHERE [Order ID] LIKE @id"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@id", OrderIDTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()

```

Κώδικας αναζήτησης:

```

'Search for records which satisfy the criteria
Me.SpRetailOrdersDetailsBindingSource.Filter =
String.Format("Status LIKE '" & StatusCBox.Text) & "' AND '" & _
String.Format("[Order ID] LIKE '" & SeasonCBox.Text) & "%'"

Me.SpRetailOrdersDetailsBindingSource.Filter =
String.Format("Description LIKE '%" & searchtxtbox.Text & "%' OR") & _
String.Format("[Full Name] LIKE '%" & searchtxtbox.Text & "%' OR") & _
String.Format("[Dress Code] LIKE '%" & searchtxtbox.Text & "%'")

```

Κώδικας εκτύπωσης τιμολογίου/απόδειξης:

```

'create a word object
Dim objWord As Word.Application
objWord = New Word.Application
objWord.Visible = True

'Load in a document from a predesigned template.
Dim mydocs As String =
System.Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments)
objWord.Documents.Add(Template:=mydocs + "\Print\INVOICE.dotx",
NewTemplate:=False)

'adding text to a document from a specified point (bookmark)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark,
Name:="InvoiceID")
objWord.Selection.TypeText(orderIDtxtbox.Text)
[...]

```

```
objWord.PrintPreview = True
```

Κώδικας εύρεσης εκκρεμούντων παραγγελιών:

```
Dim Sql AsString =
"SELECT [Due Date], [Order ID], [Full Name], Status"& _
" FROM [Retail Order] RO LEFT JOIN Customers C ON RO.[Customer ID] =
C.ID"& _
" LEFT JOIN [Order Status] OS ON RO.[Order Status]=OS.[Status Num]"& _
" WHERE DATEDIFF(day, @date, [Due Date]) >= 0 AND (Status != 'Delivered'
AND Status != 'Canceled')"& _
" ORDER BY [Due Date]"
Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
sqlcomm.Parameters.Add(New SqlParameter("@date", SqlDbType.Date)).Value =
..

SqlConnection.Open()
Dim r As SqlDataReader = sqlcomm.ExecuteReader()
If r.HasRows() Then
While r.Read()
[...]
EndWhile
r.Close()
EndIf
SqlConnection.Close()
```

5.5.7. Διαχείριση χονδρικών πωλήσεων

Στο τμήμα αυτό έχουν υλοποιηθεί η προσθήκη, ενημέρωση και αναζήτηση χονδρικών πωλήσεων. Η αναζήτηση μπορεί να γίνει κατά χρονική περίοδο, κατάσταση παραγγελίας, στοιχεία εταιρείας ή στοιχεία ειδών ένδυσης. Επίσης, δίνεται η δυνατότητα δημιουργίας και εκτύπωσης τιμολογίου και απόδειξης είσπραξης.

Κώδικας προσθήκης:

```
'insert into database a new row with the given data
Dim insertSql =
"INSERT INTO [Wholesale Order]([Order ID], [Order Date], [Discount (%)],
Discount, [Deposit (%)], Deposit, [Order Status])" & _
" VALUES(@orderid, @orderdate, 0, 0, 0, 0, 0)"
Dim sqlcomm = New SqlCommand(insertSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@orderid", OrderIDTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας ενημέρωσης:

```
'update the selected record with the new given data
Dim updateSql =
"UPDATE [Wholesale Order]" & _
" SET [Company ID]=@companyid, [Order Date]=@orderdate, [Due
Date]=@duedate, [Discount (%)]=@discountper, Discount=@discount, [Deposit
(%)]=@depositper, Deposit=@deposit, Notes=@notes, [Order Status]=@status"
```

```
& _
" WHERE [Order ID] LIKE @orderid"
Dim sqlcomm = New SqlCommand(updateSql, SqlConnection)

SqlConnection.Open()
sqlcomm.Parameters.AddWithValue("@orderid", OrderIDTextBox.Text)
[...]
sqlcomm.ExecuteNonQuery()
SqlConnection.Close()
```

Κώδικας αναζήτησης:

```
'Search for records which satisfy the criteria
Me.SpWholesaleOrdersBindingSource.Filter =
String.Format("Status LIKE '" & StatusCBox.Text) & "' AND " & _
String.Format("[Order ID] LIKE '" & SeasonCBox.Text) & "%'"

Me.SpWholesaleOrdersBindingSource.Filter =
String.Format("[Company Name] LIKE '%" & searchtxtbox.Text & "%' OR") & _
String.Format("[Boutique Name] LIKE '%" & searchtxtbox.Text & "%'")
```

Κώδικας εκτύπωσης τιμολογίου/απόδειξης:

```
'create a word object
Dim objWord As Word.Application
objWord = New Word.Application
objWord.Visible = True

'Load in a document from a predesigned template.
Dim mydocs AsString =
System.Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments)
objWord.Documents.Add(Template:=mydocs + "\Print\INVOICE_wholesale.dotx",
NewTemplate:=False)

'adding text to a document from a specified point (bookmark)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark,
Name:="InvoiceID")
objWord.Selection.TypeText(orderIDtxtbox.Text)
[...]

objWord.PrintPreview = True
```

Κώδικας εύρεσης εκκρεμούντων παραγγελιών:

```
Dim Sql AsString =
"SELECT [Due Date], [Order ID], [Company Name], [Boutique Name], Status"& _
" FROM [Wholesale Order] WO LEFT JOIN [Cooperating Company] CC ON WO.[Company ID] = CC.ID"& _
" LEFT JOIN [Order Status] OS ON WO.[Order Status]=OS.[Status Num]"& _
" WHERE DATEDIFF(day, @date, [Due Date]) >= 0 AND (Status != 'Delivered' AND Status != 'Canceled')"& _
" ORDER BY [Due Date]"

Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
sqlcomm.Parameters.Add(New SqlParameter("@date", SqlDbType.Date)).Value = ..

SqlConnection.Open()
```

```

Dim r As SqlDataReader = sqlcomm.ExecuteReader()
If r.HasRows() Then
While r.Read()
[...]
EndWhile
r.Close()
EndIf
SqlConn.Close()

```

5.5.8. Βοηθητικές/Υποστηρικτικές υλοποιήσεις

Περισσότερες πληροφορίες για βοηθητικές και υποστηρικτικές υλοποιήσεις υπάρχουν στη παράγραφο 5.4. Πιο κάτω παρατίθενται μερικά παραδείγματα.

Έχει δημιουργηθεί στη βάση δεδομένων πίνακας που περιέχει όλες τις γιορτές (ημερομηνία με εορτάζοντα ονόματα) για υποστήριξη της λειτουργίας εύρεσης της ονομαστικής εορτής κάποιου πελάτη.

Για τον πίνακα αυτό, έχει υλοποιηθεί συνάρτηση για τον υπολογισμό του Πάσχα κάθε έτους, έτσι ώστε να υποστηριχτεί η διαδικασία ανανέωσης των ημερομηνιών των εορτών, τόσο των σταθερών όσο και των κινητών.

Παράδειγμα κώδικα ανανέωσης ημερομηνίας (κινητής) εορτής:

```

--Saint George
SET @date = cast(('04/23/' + cast(@year as nvarchar)) as datetime)

IF DATEDIFF(day, @date, @easter) < 0
    UPDATE dbo.Namedays SET [date] = @date WHERE notes = 'george'
ELSE
BEGIN
    SET @date = cast(cast(cast(@easter as datetime) as integer) + 1 as datetime)
    UPDATE dbo.Namedays SET [date] = @date WHERE notes = 'george'
END

```

Έχουν δημιουργηθεί triggers (σκανδάλες) στη βάση για την υποστήριξη συνεχούς ενημέρωσης των ποσοτήτων των πρώτων υλών κατά την εισαγωγή, ενημέρωση ή διαγραφή ενός κατασκευαστικού υλικού για κάποιο είδος ένδυσης (συνολικά 6 triggers), καθώς και για υποστήριξη αυτόματων υπολογισμών ποσών χρημάτων (συνολικά 5 triggers).

Trigger ενημέρωσης ποσοτήτων πρώτων υλών κατά την ενημέρωση κατασκευαστικού υλικού:

```

CREATE TRIGGER [dbo].[tr_Dress_update]
ON [dbo].[Dress Manufacturing Details]
AFTER UPDATE
AS
DECLARE @oldquantity int, @newquantity int, @code nvarchar(10)

BEGIN
    SET NOCOUNT ON;

    SELECT @oldquantity = Quantity, @code = [Raw Material Code]
    FROM deleted

    SELECT @newquantity = Quantity, @code = [Raw Material Code]

```

```

FROM inserted

UPDATE dbo.[Raw Material]
SET Stock = Stock + @oldquantity - @newquantity
WHERE Code = @code
END

```

Triggers αυτόματου υπολογισμού ολικού ποσού χονδρικής παραγγελίας:

```

-- Update Wholesale Order on Insert new item in Wholesale Order Details
CREATE TRIGGER [dbo].[tr_wholesale_order_detail_insert]
ON [dbo].[Wholesale Order Details]
AFTER INSERT
AS
DECLARE
@id int, @orderid nvarchar(20), @garment nvarchar(10), @subtotal money,
@price money, @quantity int, @linetotal money, @vat decimal(5,2), @discount money,
@total money, @newtotal money, @deposit decimal(5,2), @vattotal money
BEGIN
SET NOCOUNT ON;

SELECT @id=ID, @orderid=[Order ID], @garment=[Garment ID],
@quantity=Quantity
FROM inserted

-- Find the price of the garment and calculate line total
SELECT @price = ISNULL([Wholesale Price], 0)
FROM Garment
WHERE Code=@garment

UPDATE dbo.[Wholesale Order Details]
SET [Price per piece]=@price, [Line Total] = (@quantity * @price)
WHERE ID=@id

-- Use the new line total to calculate subtotal, vat total,
-- total, new total and deposit of the order
SELECT @linetotal = [Line Total]
FROM dbo.[Wholesale Order Details]
WHERE ID = @id

SELECT @subtotal = Subtotal
FROM dbo.[Wholesale Order]
WHERE [Order ID]=@orderid

UPDATE dbo.[Wholesale Order]
SET Subtotal = @subtotal + @linetotal
WHERE [Order ID]=@orderid

SELECT @vat=VAT
FROM dbo.Settings

SELECT @subtotal = Subtotal
FROM dbo.[Wholesale Order]
WHERE [Order ID]=@orderid

UPDATE dbo.[Wholesale Order]

```

```

SET [VAT total] = @subtotal * @vat / 100
WHERE [Order ID]=@orderid

SELECT @vattotal=[VAT Total]
FROM dbo.[Wholesale Order]
WHERE [Order ID]=@orderid

UPDATE dbo.[Wholesale Order]
SET Total = @subtotal + @vattotal
WHERE [Order ID]=@orderid

SELECT @total=Total, @deposit=[Deposit (%)], @discount=Discount
FROM dbo.[Wholesale Order]
WHERE [Order ID]=@orderid

UPDATE dbo.[Wholesale Order]
SET [New Total] =( @total - @discount)
WHERE [Order ID]=@orderid

SELECT @newtotal=[New Total]
FROM dbo.[Wholesale Order]
WHERE [Order ID]=@orderid

UPDATE dbo.[Wholesale Order]
SET Deposit = @deposit * @newtotal / 100
WHERE [Order ID]=@orderid
END

-- Update Wholesale Order totals after updating other amounts
CREATETRIGGER [dbo].[tr_wholesale_order_update]
ON [dbo].[Wholesale Order]
AFTERUPDATE
AS
DECLARE @orderid nvarchar(20), @total money, @newtotal money,
        @olddiscount money, @newdiscount money,
        @olddeposit decimal(5,2), @newdeposit decimal(5,2)
BEGIN
SETNOCOUNTON;

    SELECT @orderid=[Order ID], @olddiscount=Discount, @olddeposit=[Deposit (%)]
    FROM deleted

    SELECT @orderid=[Order ID], @newdiscount=Discount, @newdeposit=[Deposit (%)],
    @total=Total, @newtotal=[New Total]
    FROM inserted

-- Update Total if discount amount has changed
IF @olddiscount != @newdiscount
    BEGIN
        UPDATE dbo.[Wholesale Order]
        SET [New Total] =( @total - @newdiscount)
        WHERE [Order ID]=@orderid
    END

-- Update Deposit if deposit percentage has changed

```

```
IF @olddeposit != @newdeposit
    BEGIN
    UPDATE dbo.[Wholesale Order]
        SET Deposit = (@newdeposit * @newtotal / 100)
        WHERE [Order ID]=@orderid
    END
END
```

Κεφάλαιο 6. Εγχειρίδιο χρήσης

6.1. Μενού έναρξης	124
6.2. Ρυθμίσεις παραμέτρων	126
6.3. Διαχείριση πελατών	127
6.4. Διαχείριση συνεργαζόμενων εταιρειών.....	130
6.5. Διαχείριση προμηθευτών.....	133
6.6. Διαχείριση πρώτων υλών	135
6.7. Διαχείριση προϊόντων	138
6.8. Διαχείριση συλλογών προϊόντων	141
6.9. Διαχείριση λιανικών πωλήσεων	144
6.10. Διαχείριση χονδρικών πωλήσεων.....	146
6.11. Εξαγωγή επαφών συστήματος σε αρχείο	148
6.12. Εισαγωγή επαφών στο σύστημα από αρχείο.....	149
6.13. Έξυπνες πωλήσεις	149

6.1. Μενού έναρξης

Κατά την εκκίνηση του λογισμικού, εμφανίζεται το κεντρικό μενού επιλογών (Εικόνα 6.1). Στο μενού εμφανίζονται όλες οι βασικές λειτουργίες του συστήματος. Ο χρήστης μπορεί να χρησιμοποιήσει κάποια λειτουργία, πατώντας στην αντίστοιχη εικόνα.



Εικόνα 6.1

Επεξήγησηεικονιδίων

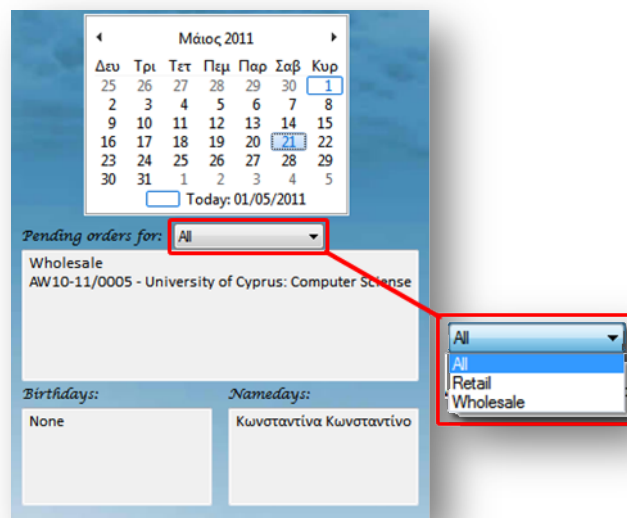
1. Κεντρική φόρμα επιλογών
2. Διαχείριση πελατών
3. Διαχείριση προϊόντων
4. Διαχείριση λιανικών παραγγελιών
5. Διαχείριση συνεργαζόμενων εταιρειών
6. Διαχείριση ειδών ένδυσης και συλλογών προϊόντων
7. Διαχείριση χονδρικών παραγγελιών
8. Διαχείριση προμηθευτών
9. Διαχείριση πρώτων υλών
10. Προειδοποίηση ύπαρξης εξαντλημένων πρώτων υλών
11. Εξαγωγή επαφών συστήματος σε αρχείο
12. Εισαγωγή επαφών από αρχείο
13. Έξυπνος πωλητής
14. Ημερολόγιο
15. Επιλογές εμφάνισης παραγγελιών
16. Τρέχουσες, εκκρεμούσες παραγγελίες

17. Πελάτες με γενέθλια την τρέχουσα ημέρα
18. Πελάτες με ονομαστική εορτή την τρέχουσα ημέρα
19. Ρύθμιση παραμέτρων
20. Έξοδος από το σύστημα

6.1.1. Ημερολόγιο, Εκκρεμούσες παραγγελίες, Εορτάζοντες

Στο δεξιό μέρος της φόρμας, υπάρχει το ημερολόγιο. Επιλέγοντας μια ημερομηνία, εμφανίζονται οι παραγγελίες που εκκρεμούν μέχρι την επιλεγμένη ημέρα, καθώς και οι εορτάζοντες της ημέρας.

Οι παραγγελίες μπορούν να φιλτραριστούν, χρησιμοποιώντας τη πτυσσόμενη λίστα. Για την εμφάνιση όλων των τρέχουσων και εκκρεμούντων παραγγελιών, επιλέξτε *All*. Για το τμήμα των λιανικών πωλήσεων, επιλέξτε *Retail* και για το τμήμα χονδρικών πωλήσεων, επιλέξτε *Wholesale*.



Εικόνα 6. 2

6.1.2. Προειδοποίηση εξαντλημένων πρώτων υλών

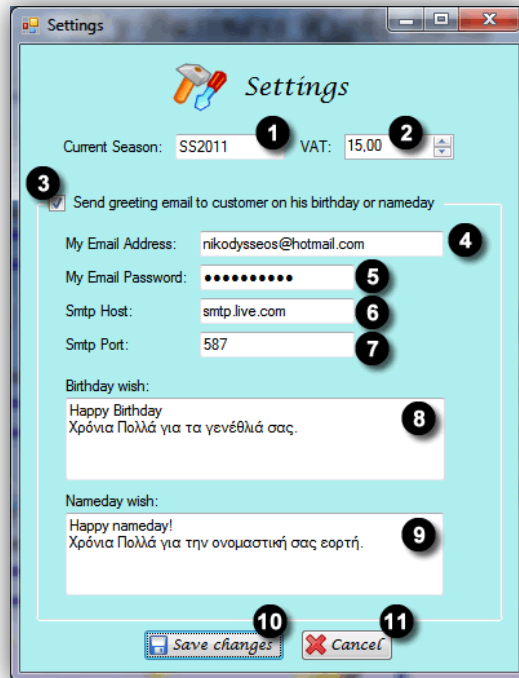
Εάν κατά τη διάρκεια χρήσης του λογισμικού, εμφανιστεί το εικονίδιο του θαυμαστικού δίπλα από τη διαχείριση πρώτων υλών, ελέγξτε τις πρώτες ύλες. Κάποια ή κάποιες εξαντλήθηκαν.



Εικόνα 6.3

6.2. Ρυθμίσεις παραμέτρων

Πατώντας στο εικονίδιο *Settings* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Ρύθμιση παραμέτρων* (Εικόνα 6.4).



The screenshot shows a 'Settings' window with the following elements and numbered callouts:

- 1: 'Current Season' dropdown menu showing 'SS2011'.
- 2: 'VAT' dropdown menu showing '15.00'.
- 3: Check box labeled 'Send greeting email to customer on his birthday or nameday'.
- 4: 'My Email Address' text field containing 'nikodysseos@hotmail.com'.
- 5: 'My Email Password' text field with masked characters.
- 6: 'Smtp Host' text field containing 'smtp.live.com'.
- 7: 'Smtp Port' text field containing '587'.
- 8: 'Birthday wish' text area containing 'Happy Birthday' and 'Χρόνια Πολλά για τα γενέθλιά σας.'.
- 9: 'Nameday wish' text area containing 'Happy nameday!' and 'Χρόνια Πολλά για την ονομαστική σας εορτή.'.
- 10: 'Save changes' button.
- 11: 'Cancel' button.

Εικόνα 6.4

Στη φόρμα *Ρύθμιση παραμέτρων*, μπορείτε να ρυθμίσετε την τρέχουσα χρονική περίοδο (1) και το δείκτη φόρου προστιθέμενης αξίας (2).

Ακόμη, αν επιθυμείτε την αποστολή ευχετήριου ηλεκτρονικού μηνύματος την ημέρα της εορτής ή των γενεθλίων κάποιου πελάτη, μαρκάρετε την ένδειξη *Sendgreetingemailtocustomeronhisbirthdayornameday* (3). Θα πρέπει να ρυθμίσετε τη διεύθυνση του ηλεκτρονικού σας ταχυδρομείου (4), τον κωδικό πρόσβασης (5), το διακομιστή SMTP της υπηρεσίας ηλεκτρονικού ταχυδρομείου (6), καθώς και τη θύρα του διακομιστή (7). Στα πεδία 8 και 9, πληκτρολογήστε το ευχετήριο μήνυμα για τα γενέθλια και την ονομαστική εορτή αντίστοιχα. Αυτό είναι το κείμενο το οποίο θα αναγράφεται στο ηλεκτρονικό μήνυμα που θα αποστέλλεται αυτόματα από το σύστημα.

Για επιβεβαίωση και αποθήκευση των επιλογών σας, πιάστε το πλήκτρο *Savechanges* (10). Για ακύρωση, πατήστε το πλήκτρο *Cancel* (11).



6.3. Διαχείριση πελατών

Πατώντας στο εικονίδιο *Customers* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση πελατών* (Εικόνα 6.5). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

1. τον πίνακα με τους καταχωρημένους πελάτες
2. το πεδίο εμφάνισης των πληροφοριών του επιλεγμένου πελάτη
3. την εργαλειοθήκη
4. το πεδίο αναζήτησης

ID	Full Name
4567	Αντώνης Αντωνίου
123456	Αντρέας Σικράτους
886546	Νίκος Οδυσσεύς
893892	Μαρία Φίλιππου
997469	Κωνσταντίνος Κωνσταντίνου
1234567890	Στέλλα Αντρέου

Εικόνα 6.5

6.3.1. Εισαγωγή νέου πελάτη



1. Από την εργαλειοθήκη, πατήστε το κουμπί *Add*.
2. Εμφανίζεται η φόρμα *NewCustomer*. (Εικόνα 6.6)
3. Συμπληρώστε τη φόρμα με τα στοιχεία του πελάτη.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη δίπλα από το πεδίο με τα λανθασμένα στοιχεία (Εικόνα 6.8). Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Κατά την πληκτρολόγηση του ονοματεπώνυμου του πελάτη, η λίστα του πεδίου *Nameday* γεννημερώνεται με βάση το όνομα που πληκτρολογείται. (Εικόνα 6.7)

Τα πεδία τηλεφώνων μπορούν να περιλαμβάνουν αριθμητικά ψηφία, κενά και τα σύμβολα +, -, (,) μόνο.

Full Name: Μαρία

Birth Date: / / Nameday: 25/07/2011

Address: Street Street No Region

PO Code City Country

Telephone No: Mobile No: Email:

Bust: Waist: Hips:

Save & Close Cancel

Εικόνα 6.6

Adding... New Customer

Basic Info

ID:

Full Name:

Birth Date: / / Nameday:

Address

Street Street No Region

PO Code City Country

Contact info

Telephone No:

Mobile No:

Email:

Vital statistics (cm)

Bust: Waist: Hips:

Save & Close Cancel

Εικόνα 6.7

6.3.2. Ενημέρωση στοιχείων πελάτη

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Edit*.
2. Εμφανίζεται η φόρμα *EditCustomer*. (Εικόνα 6.8)
3. Συμπληρώστε τη φόρμα με τα νέα στοιχεία του πελάτη.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Editing... Edit Customer

Basic Info

ID: 886546

Full Name: Νίκος Οδυσσεύς

Birth Date: 18/07/1987 Nameday: 06/12/2011

Address

Street Αγίας Παρασκευής Street No 3 Region Γερμασιόγεια

PO Code 4044 City Λεμεσός Country Κύπρος

Contact info

Telephone No: (+357) 25 325715

Mobile No: (+357) 99 121876

Email: nikodysseos@hotmail.com

Vital statistics (cm)

Bust: 90 Waist: 90 Hips: 90

Save & Close Cancel

Εικόνα 6.8

6.3.3. Διαγραφή πελάτη

1. Από τον πίνακα πελατών, επιλέξτε την εγγραφή που επιθυμείτε να διαγράψετε.



2. Από την εργαλειοθήκη, πατήστε το κουμπί *Remove*.

3. Επιβεβαιώστε την επιλογή σας.

6.3.4. Αναζήτηση πελάτη

1. Πληκτρολογήστε στο πεδίο αναζήτησης οποιοδήποτε στοιχείο γνωρίζετε και πιέστε *Enter* ή πατήστε το κουμπί .

2. Καθαρίστε το πεδίο πατώντας το κουμπί .

6.3.5. Αποστολή ηλεκτρονικού μηνύματος σε πελάτη

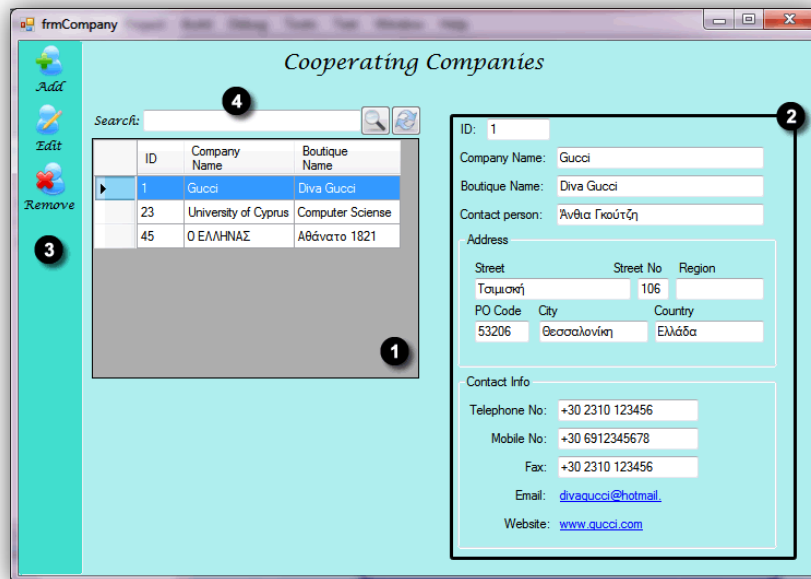
1. Αν υπάρχει καταχωρημένη διεύθυνση ηλεκτρονικού ταχυδρομείου για την επιλεγμένη εγγραφή, μπορείτε να στείλετε ηλεκτρονικό μήνυμα στην εγγραφή, πατώντας στο σύνδεσμο της διεύθυνσης.

2. Στο νέο πλαίσιο διαλόγου, συμπληρώστε τα πεδία που θέλετε και πατήστε *Send*.

6.4. Διαχείριση συνεργαζόμενων εταιρειών

Πατώντας στο εικονίδιο *Companies* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση συνεργαζόμενων εταιρειών* (Εικόνα 6.9). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

1. τον πίνακα με τις καταχωρημένες εταιρείες
2. το πεδίο εμφάνισης των πληροφοριών της επιλεγμένης εταιρείας
3. τη νεργαλειοθήκη
4. το πεδίο αναζήτησης



Εικόνα 6.9

6.4.1. Εισαγωγή νέας συνεργαζόμενης εταιρείας



1. Από την εργαλειοθήκη, πατήστε το κουμπί *Add*.
2. Εμφανίζεται η φόρμα *New Cooperating Company*. (Εικόνα 6.10)
3. Συμπληρώστε τη φόρμα με τα στοιχεία της εταιρείας.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας τον κέρσορα πάνω στην ένδειξη εμφανίζεται επεξήγηση του λάθους.

Τα πεδία τηλεφώνων μπορούν να περιλαμβάνουν αριθμητικά ψηφία, κενά και τα σύμβολα +, -, (,) μόνο.

Εικόνα 6.10

6.4.2. Ενημέρωση στοιχείων συνεργαζόμενης εταιρείας



1. Από την εργαλειοθήκη, πατήστε το κουμπί *Edit*.
2. Εμφανίζεται η φόρμα *Edit Cooperating Company*. (Εικόνα 6.11)
3. Συμπληρώστε τη φόρμα με τα νέα στοιχεία της εταιρείας.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save & Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Εικόνα 6.11

6.4.3. Διαγραφή συνεργαζόμενης εταιρείας

1. Από τον πίνακα εταιρειών, επιλέξτε την εγγραφή που επιθυμείτε να διαγράψετε.



2. Από την εργαλειοθήκη, πατήστε το κουμπί *Remove*.

3. Επιβεβαιώστε την επιλογή σας.

6.4.4. Αναζήτηση συνεργαζόμενης εταιρείας

1. Πληκτρολογήστε στο πεδίο αναζήτησης οποιοδήποτε στοιχείο γνωρίζετε και πιέστε *Enter* ή πατήστε το κουμπί .
2. Καθαρίστε το πεδίο πατώντας το κουμπί .

6.4.5. Αποστολή ηλεκτρονικού μηνύματος σε συνεργαζόμενη εταιρεία

1. Αν υπάρχει καταχωρημένη διεύθυνση ηλεκτρονικού ταχυδρομείου για την επιλεγμένη εγγραφή, μπορείτε να στείλετε ηλεκτρονικό μήνυμα στην εγγραφή, πατώντας στο σύνδεσμο της διεύθυνσης.
2. Στο νέο πλαίσιο διαλόγου, συμπληρώστε τα πεδία που θέλετε και πατήστε *Send*.

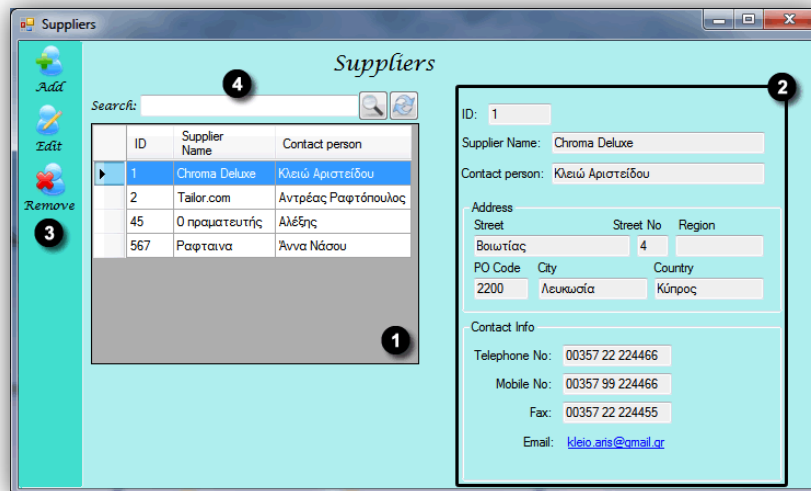
6.4.6. Άνοιγμα ιστοσελίδας συνεργαζόμενης εταιρείας

Αν υπάρχει καταχωρημένη ιστοσελίδα για την επιλεγμένη εγγραφή, μπορείτε να την ανοίξετε στο φυλλομετρητή σας, πατώντας στο σύνδεσμο της ιστοσελίδας.

6.5. Διαχείριση προμηθευτών

Πατώντας στο εικονίδιο *Suppliers* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση προμηθευτών* (Εικόνα 6.12). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

1. τον πίνακα με τους καταχωρημένους προμηθευτές
2. το πεδίο εμφάνισης των πληροφοριών του επιλεγμένου προμηθευτή
3. την εργαλειοθήκη
4. το πεδίο αναζήτησης



Εικόνα 6.12

6.5.1. Εισαγωγή νέου προμηθευτή

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Add*.
2. Εμφανίζεται η φόρμα *New Supplier*. (Εικόνα 6.13)
3. Συμπληρώστε τη φόρμα με τα στοιχεία του προμηθευτή.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save & Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Τα πεδία τηλεφώνων μπορούν να περιλαμβάνουν αριθμητικά ψηφία, κενά και τα σύμβολα +, -, (,) μόνο.



Εικόνα 6.13

6.5.2. Ενημέρωση στοιχείων προμηθευτή

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Edit*.
2. Εμφανίζεται η φόρμα *Edit Supplier*. (Εικόνα 6.14)

3. Συμπληρώστε τη φόρμα με τα νέα στοιχεία του προμηθευτή.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.



Εικόνα 6.14

6.5.3. Διαγραφή προμηθευτή

1. Από τον πίνακα προμηθευτών, επιλέξτε την εγγραφή που επιθυμείτε να διαγράψετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί  *Remove*.
3. Επιβεβαιώστε την επιλογή σας (Εικόνα 6.35).

6.5.4. Αναζήτηση προμηθευτή

1. Πληκτρολογήστε στο πεδίο αναζήτησης οποιοδήποτε στοιχείο γνωρίζετε και πιέστε *Enter* ή πατήστε το κουμπί .
2. Καθαρίστε το πεδίο πατώντας το κουμπί .

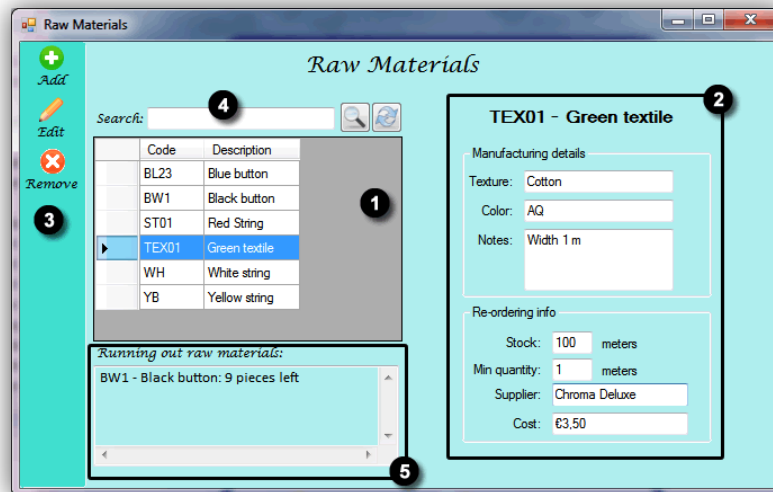
6.5.5. Αποστολή ηλεκτρονικού μηνύματος σε προμηθευτή

1. Αν υπάρχει καταχωρημένη διεύθυνση ηλεκτρονικού ταχυδρομείου για την επιλεγμένη εγγραφή, μπορείτε να στείλετε ηλεκτρονικό μήνυμα στην εγγραφή, πατώντας στο σύνδεσμο της διεύθυνσης.
2. Στο νέο πλαίσιο διαλόγου, συμπληρώστε τα πεδία που θέλετε και πατήστε *Send*.

6.6. Διαχείριση πρώτων υλών

Πατώντας στο εικονίδιο *RawMaterials* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση πρώτων υλών* (Εικόνα 6.15). Η φόρμα αποτελείται από πέντε βασικά μέρη:

1. τον πίνακα με τις καταχωρημένες πρώτες ύλες
2. το πεδίο εμφάνισης των πληροφοριών της επιλεγμένης πρώτης ύλης
3. την εργαλειοθήκη
4. το πεδίο αναζήτησης
5. το πεδίο εμφάνισης των εξαντλημένων πρώτων υλών



Εικόνα 6.15

Αν το απόθεμα κάποιας πρώτης ύλης γίνει ίσο ή μικρότερο του κατώτερου ορίου που ορίστηκε, εμφανίζεται στο πεδίο εμφάνισης των εξαντλημένων πρώτων υλών και η ένδειξη  εμφανίζεται στο κεντρικό μενού επιλογών.

6.6.1. Εισαγωγή νέας πρώτης ύλης

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Add*.
2. Εμφανίζεται η φόρμα *NewRawMaterial*. (Εικόνα 6.16)
3. Συμπληρώστε τη φόρμα με τα στοιχεία της πρώτης ύλης.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Τα κουμπιά  και  βοηθούν στην προσθήκη ενός νέου χρωματικού συνδυασμού ή ενός νέου προμηθευτή αντίστοιχα. Για την προσθήκη ενός νέου χρωματικού συνδυασμού, μπορείτε να βρείτε πληροφορίες στην παράγραφο 6.6.1.1. Για την προσθήκη ενός νέου προμηθευτή, μπορείτε να βρείτε πληροφορίες στην παράγραφο 6.5.1.

6.6.1.1. Προσθήκη χρωματικού συνδυασμού

1. Στη φόρμα *NewColorCode*, συμπληρώστε τον κωδικό και το όνομα του χρωματικού συνδυασμού. (Εικόνα 6.17)
2. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save*. Ακυρώστε, πατώντας το κουμπί *Cancel*.

Εικόνα 6.16

Εικόνα 6.17

6.6.2. Ενημέρωση στοιχείων πρώτης ύλης

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Edit*.
2. Εμφανίζεται η φόρμα *EditRawMaterial*. (Εικόνα 6.18)
3. Συμπληρώστε τη φόρμα με τα νέα στοιχεία της πρώτης ύλης.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Εικόνα 6.18

6.6.3. Διαγραφή πρώτης ύλης

1. Από τον πίνακα πρώτων υλών, επιλέξτε την εγγραφή που επιθυμείτε να διαγράψετε.

2. Από την εργαλειοθήκη, πατήστε το κουμπί  **Remove**.
3. Επιβεβαιώστε την επιλογή σας.

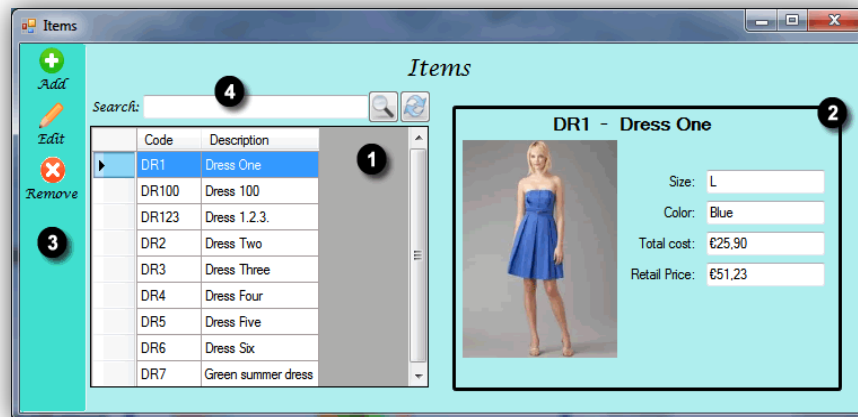
6.6.4. Αναζήτησηπρομηθευτή

1. Πληκτρολογήστε στο πεδίο αναζήτησης οποιοδήποτε στοιχείο γνωρίζετε και πιέστε *Enter* ή πατήστε το κουμπί .
2. Καθαρίστε το πεδίο πατώντας το κουμπί .

6.7. Διαχείριση προϊόντων

Πατώντας στο εικονίδιο *Items* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση προϊόντων* (Εικόνα 6.19). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

1. τον πίνακα με τα καταχωρημένα προϊόντα
2. το πεδίο εμφάνισης των πληροφοριών του επιλεγμένου προϊόντος
3. την εργαλειοθήκη
4. το πεδίο αναζήτησης

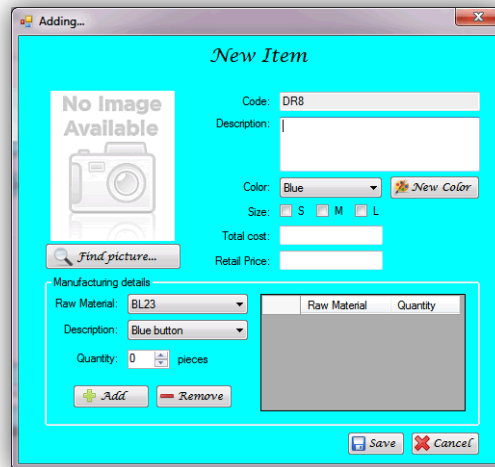


Εικόνα 6.19

6.7.1. Εισαγωγή νέου προϊόντος

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Add*.
2. Πληκτρολογήστε το κωδικό του νέου προϊόντος
3. Εμφανίζεται η φόρμα *NewItem*. (Εικόνα 6.20)
4. Συμπληρώστε τη φόρμα με τα στοιχεία του προϊόντος.
5. Για να προσθέσετε λεπτομέρειες κατασκευής, επιλέξτε την πρώτη ύλη από τη λίστα, καθορίστε την ποσότητα και πατήστε *Add*. Μπορείτε να αφαιρέσετε μια λεπτομέρεια, επιλέγοντας την από τον πίνακα κατασκευαστικών λεπτομερειών και πατώντας *Remove*.
6. Για να προσθέσετε εικόνα του προϊόντος, πατήστε στο πλήκτρο *Find picture*. Στο νέο πλαίσιο διαλόγου, επιλέξτε την εικόνα και πατήστε *Open*.
7. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
8. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

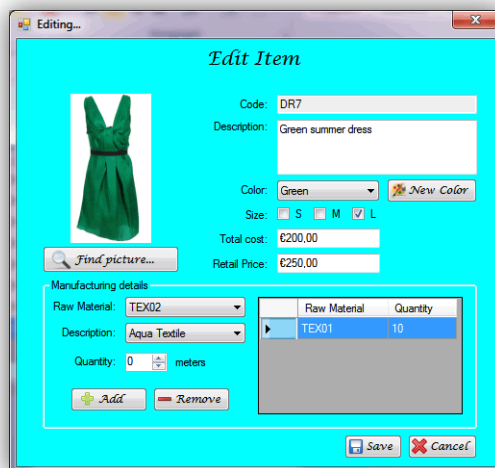
Με το κουμπί  μπορείτε να προσθέσετε ένα νέο χρωματικό συνδυασμό. Για την προσθήκη ενός νέου χρωματικού συνδυασμού, μπορείτε να βρείτε πληροφορίες στην παράγραφο 6.6.1.1.



Εικόνα 6.20

6.7.2. Ενημέρωση στοιχείων προϊόντος

1. Από την εργαλειοθήκη, πατήστε το κουμπί  **Edit**.
2. Εμφανίζεται η φόρμα *EditItem*. (Εικόνα 6.21)
3. Συμπληρώστε τη φόρμα με τα νέα στοιχεία του προϊόντος.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.
5. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.



Εικόνα 6.21

6.7.3. Διαγραφή προϊόντος

1. Από τον πίνακα προϊόντων, επιλέξτε την εγγραφή που επιθυμείτε να διαγράψετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί  **Remove**.
3. Επιβεβαιώστε την επιλογή σας.

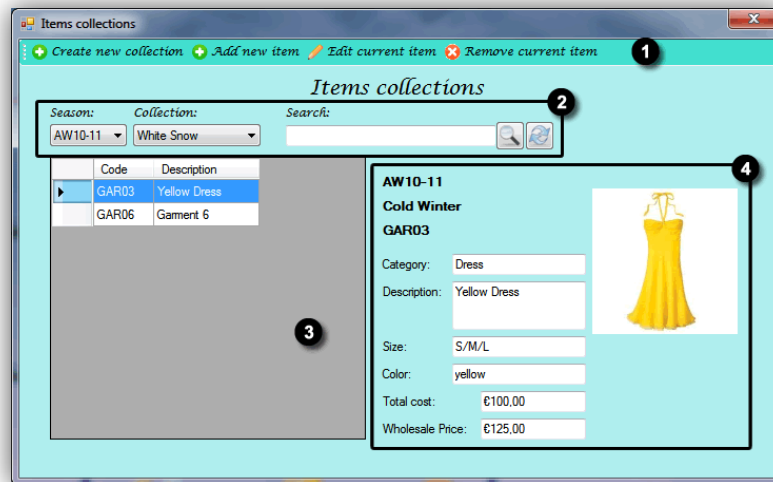
6.7.4. Αναζήτηση προϊόντος

1. Πληκτρολογήστε στο πεδίο αναζήτησης οποιοδήποτε στοιχείο γνωρίζετε και πιέστε *Enter* ή πατήστε το κουμπί .
2. Καθαρίστε το πεδίο πατώντας το κουμπί .

6.8. Διαχείριση συλλογών προϊόντων

Πατώντας στο εικονίδιο *Collections* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση συλλογών προϊόντων* (Εικόνα 6.22). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

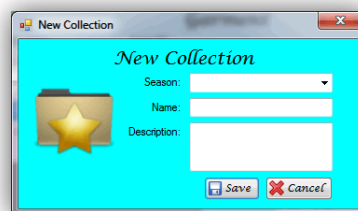
1. τη νεργαλειοθήκη
2. τα πεδία αναζήτησης
3. τον πίνακα με τα καταχωρημένα προϊόντα για την επιλεγμένη συλλογή
4. το πεδίο εμφάνισης των πληροφοριών του επιλεγμένου προϊόντος



Εικόνα 6.22

6.8.1. Δημιουργία νέας συλλογής

1. Από την εργαλειοθήκη, πατήστε το κουμπί  *Create new collection*.
2. Εμφανίζεται η φόρμα *NewCollection*. (Εικόνα 6.23)
3. Συμπληρώστε τη φόρμα με τη χρονική περίοδο, το όνομα και την περιγραφή της συλλογής.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save*. Ακυρώστε, πατώντας το κουμπί *Cancel*.



Εικόνα 6.23

6.8.2. Εισαγωγή νέου προϊόντος σε συλλογή

1. Επιλέξτε τη συλλογή στην οποία επιθυμείτε να προσθέσετε νέο προϊόν, χρησιμοποιώντας τις λίστες *Season* και *Collection*.
2. Από την εργαλειοθήκη, πατήστε το κουμπί  *Add new item*.
3. Πληκτρολογήστε το κωδικό του νέου προϊόντος
4. Εμφανίζεται η φόρμα *NewItem*. (Εικόνα 6.24)
5. Συμπληρώστε τη φόρμα με τα στοιχεία του προϊόντος.

6. Για να προσθέσετε λεπτομέρειες κατασκευής, επιλέξτε την πρώτη ύλη από τη λίστα, καθορίστε την ποσότητα και πατήστε *Add*. Μπορείτε να αφαιρέσετε μια λεπτομέρεια, επιλέγοντας την από τον πίνακα κατασκευαστικών λεπτομερειών και πατώντας *Remove*.
7. Για να προσθέσετε εικόνα του προϊόντος, πατήστε στο πλήκτρο *Findpicture*. Στο νέο πλαίσιο διαλόγου, επιλέξτε την εικόνα και πατήστε *Open*.
8. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωση. Ακυρώστε, πατώντας το κουμπί *Cancel*.
9. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.

Με το κουμπί  μπορείτε να προσθέσετε ένα νέο χρωματικό συνδυασμό. Για την προσθήκη ενός νέου χρωματικού συνδυασμού, μπορείτε να βρείτε πληροφορίες στην παράγραφο 6.6.1.1.



Εικόνα 6.24

6.8.3. Ενημέρωση στοιχείων προϊόντος

1. Επιλέξτε από τον πίνακα προϊόντων το προϊόν που επιθυμείτε να ενημερώσετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί  *Edit current item*.
3. Εμφανίζεται η φόρμα *EditItem*. (Εικόνα 6.25)
4. Συμπληρώστε τη φόρμα με τα νέα στοιχεία του προϊόντος.
5. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα. Ακυρώστε, πατώντας το κουμπί *Cancel*.
6. Σε περίπτωση λάθους, εμφανίζεται η ένδειξη  δίπλα από το πεδίο με τα λανθασμένα στοιχεία. Μετακινώντας το δείκτη πάνω στην ένδειξη, εμφανίζεται επεξήγηση του λάθους.



Εικόνα 6.25

6.8.4. Διαγραφή προϊόντος

1. Από τον πίνακα προϊόντων, επιλέξτε την εγγραφή που επιθυμείτε να διαγράψετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί  *Remove current item*.
3. Επιβεβαιώστε την επιλογή σας.

6.8.5. Αναζήτηση προϊόντων από τις συλλογές

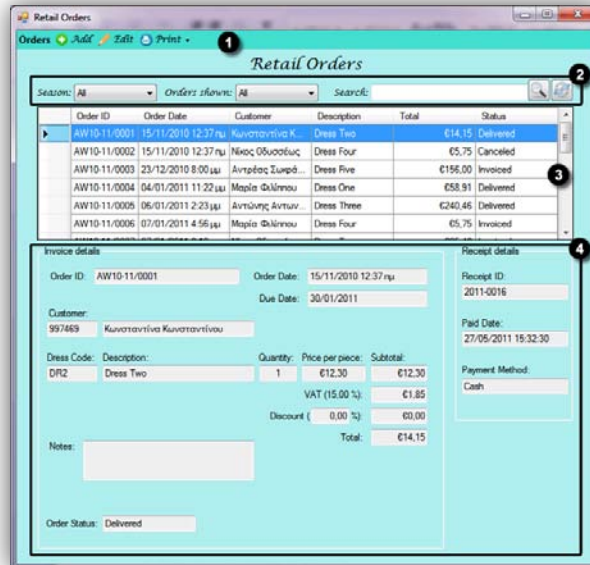
Πληκτρολογήστε στο πεδίο αναζήτησης οποιοδήποτε στοιχείο γνωρίζετε και πιάστε *Enter* ή πατήστε το κουμπί . Καθαρίστε το πεδίο πατώντας το κουμπί .



6.9. Διαχείριση λιανικών πωλήσεων

Πατώντας στο εικονίδιο *Orders* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση λιανικών παραγγελιών* (Εικόνα 6.26). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

1. τη νεργαλειοθήκη
2. τα πεδία αναζήτησης
3. τον πίνακα με τις καταχωρημένες παραγγελίες για την επιλεγμένη χρονική περίοδο
4. το πεδίο εμφάνισης των πληροφοριών της επιλεγμένης παραγγελίας



Εικόνα 6.26

6.9.1. Εισαγωγή νέας παραγγελίας

1. Από την εργαλειοθήκη, πατήστε το κουμπί *Add*.
2. Συμπληρώστε τη φόρμα με τα στοιχεία της παραγγελίας. (Εικόνα 27)
3. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.

Όλα τα χρηματικά ποσά υπολογίζονται αυτόματα από το σύστημα. Παρέχεται όμως η δυνατότητα τροποποίησής τους.

Εικόνα 6.27

6.9.2. Ενημέρωση στοιχείων παραγγελίας

1. Επιλέξτε από τον πίνακα παραγγελιών την παραγγελία που επιθυμείτε να ενημερώσετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *Edit*.
3. Συμπληρώστε τη φόρμα με τα νέα στοιχεία της παραγγελίας.
4. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.

6.9.3. Ακύρωση παραγγελίας

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία που επιθυμείτε να ακυρώσετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *Edit*.
3. Αλλάξτε την ένδειξη στο *OrderStatus* σε *Canceled*.
4. Αποθηκεύστε την παραγγελία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης.

6.9.4. Έκδοση απόδειξης

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία για την οποία επιθυμείτε να εκδώσετε απόδειξη.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *Edit*.
3. Αλλάξτε την ένδειξη στο *OrderStatus* σε *Delivered*.
4. Επιλέξτε τον τρόπο πληρωμής.
5. Αποθηκεύστε την παραγγελία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης.

6.9.5. Αναζήτηση παραγγελιών

Μπορείτε να αναζητήσετε παραγγελίες, με τους πιο κάτω τρόπους:

- α) Επιλογή χρονικής περιόδου από την αντίστοιχη λίστα
- β) Επιλογή κατάστασης παραγγελίας από τη λίστα καταστάσεων
- γ) Πληκτρολόγηση στο πεδίο αναζήτησης οποιουδήποτε στοιχείου. Πιέστε *Enter* μετά την πληκτρολόγηση ή πατήστε το κουμπί . Καθαρίστε το πεδίο πατώντας το κουμπί .

6.9.6. Εκτύπωση τιμολογίου

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία για την οποία επιθυμείτε να εκτυπώσετε τιμολόγιο.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *PrintInvoice*.
3. Το τιμολόγιο ανοίγει στον προεπιλεγμένο επεξεργαστή κειμένου του υπολογιστή σας, έτοιμο για εκτύπωση.

6.9.7. Εκτύπωση απόδειξης

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία για την οποία επιθυμείτε να εκτυπώσετε απόδειξη.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *PrintReceipt*.
3. Η απόδειξη ανοίγει στον προεπιλεγμένο επεξεργαστή κειμένου του υπολογιστή σας, έτοιμη για εκτύπωση.



6.10. Διαχείριση χονδρικών πωλήσεων

Πατώντας στο εικονίδιο *Orders* στην κεντρική φόρμα επιλογών, εμφανίζεται η φόρμα *Διαχείριση χονδρικών παραγγελιών* (Εικόνα 6.28). Η φόρμα αποτελείται από τέσσερα βασικά μέρη:

1. τηνεργαλειοθήκη
2. τα πεδία αναζήτησης
3. τον πίνακα με τις καταχωρημένες παραγγελίες για την επιλεγμένη χρονική περίοδο
4. το πεδίο εμφάνισης των πληροφοριών της επιλεγμένης παραγγελίας

Order ID	Order Date	Company Name	Boutique Name	No of items	New Total	Status
AW10-11/0001	10/02/2011 12:00 ru	University of Cyp...	Computer Science	3	€520.00	Delivered
AW10-11/0002	14/02/2011 1:49 ru	University of Cyp...	Computer Science	6	€2,012.50	Delivered
AW10-11/0003	14/02/2011 2:16 ru	O.E.M.H.N.A.Z.	Abbasia 1821	3	€618.75	Delivered
AW10-11/0004	14/02/2011 11:02 ru	O.E.M.H.N.A.Z.	Abbasia 1821	7	€2,012.50	Delivered
AW10-11/0005	15/05/2011 2:21 ru	University of Cyp...	Computer Science	2	€258.75	Invoiced

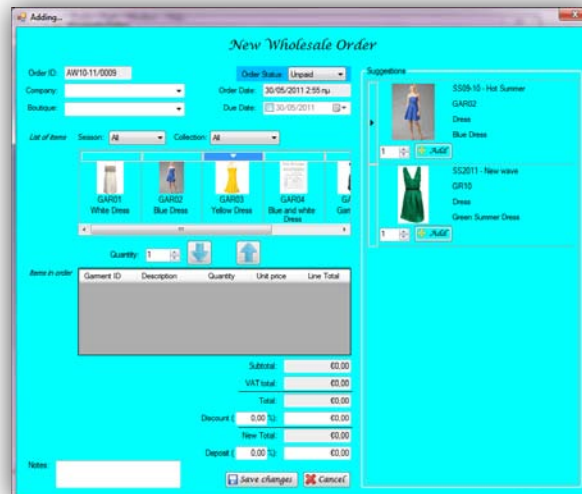
Εικόνα 6.28

6.10.1. Εισαγωγή νέας παραγγελίας

1. Από την εργαλειοθήκη, επιλέξτε *AddNewOrder*.
2. Εμφανίζεται η φόρμα *NewWholesaleOrder*. (Εικόνα 6.29)
3. Συμπληρώστε τη φόρμα με τα στοιχεία της παραγγελίας.
4. Για να προσθέσετε αντικείμενα στην παραγγελία, επιλέξτε το αντικείμενο από τη συλλογή που ανήκει, καθορίστε την ποσότητα και πιέστε το κουμπί *Add*. Μπορείτε να αφαιρέσετε κάποιο αντικείμενο από την παραγγελία, επιλέγοντας το από τον πίνακα *Itemsinorder* και πιέζοντας το κουμπί *Remove*.
5. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.

Όλα τα χρηματικά ποσά υπολογίζονται αυτόματα από το σύστημα. Παρέχεται όμως η δυνατότητα τροποποίησής τους.

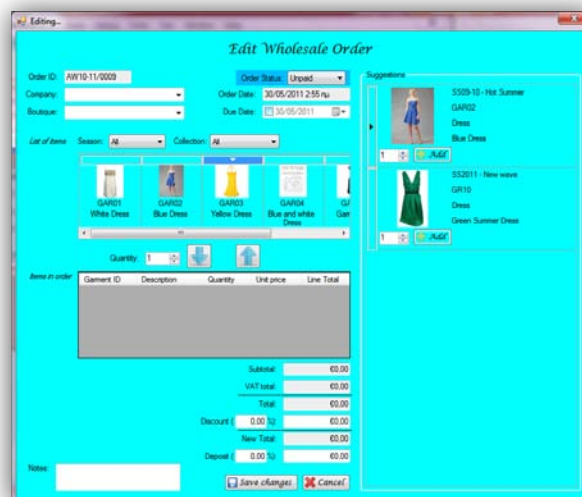
Κάθε φορά που επιλέγεται ένα προϊόν, εμφανίζονται εισηγήσεις για προϊόντα που αγοράστηκαν μαζί με το επιλεγμένο. Μπορείτε να τα προσθέσετε στην παραγγελία, πατώντας το κουμπί *Add* κάτω από το προϊόν.



Εικόνα 6.29

6.10.2. Ενημέρωση στοιχείων παραγγελίας

1. Επιλέξτε από τον πίνακα παραγγελιών την παραγγελία που επιθυμείτε να ενημερώσετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *Edit*.
3. Εμφανίζεται η φόρμα *EditWholesaleOrder*. (Εικόνα 6.30)
4. Συμπληρώστε τη φόρμα με τα νέα στοιχεία της παραγγελίας.
5. Αποθηκεύστε τα νέα στοιχεία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης. Ακυρώστε, πατώντας το κουμπί *Cancel*.



Εικόνα 6.30

6.10.3. Ακύρωση παραγγελίας

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία που επιθυμείτε να ακυρώσετε.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *Edit*.
3. Εμφανίζεται η φόρμα *EditWholesaleOrder*.
4. Αλλάξτε την ένδειξη στο *OrderStatus* σε *Canceled*.
5. Αποθηκεύστε την παραγγελία, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης.

6.10.4. Έκδοση απόδειξης

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία για την οποία επιθυμείτε να εκδώσετε απόδειξη.
2. Από την εργαλειοθήκη, επιλέξτε *AddNewReceipt*.
3. Επιλέξτε τολόγο και τρόπο πληρωμής.
4. Αποθηκεύστε την απόδειξη, πατώντας το κουμπί *Save&Close*. Εμφανίζεται μήνυμα επιβεβαίωσης.

6.10.5. Αναζήτηση παραγγελιών

Μπορείτε να αναζητήσετε παραγγελίες, με τους πιο κάτω τρόπους:

- α) Επιλογή χρονικής περιόδου από την αντίστοιχη λίστα
- β) Επιλογή κατάστασης παραγγελίας από τη λίστα καταστάσεων
- γ) Πληκτρολόγηση στο πεδίο αναζήτησης οποιουδήποτε στοιχείου. Πιέστε *Enter* μετά την πληκτρολόγηση ή πατήστε το κουμπί . Καθαρίστε το πεδίο πατώντας το κουμπί .

6.10.6. Εκτύπωση τιμολογίου

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία για την οποία επιθυμείτε να εκτυπώσετε τιμολόγιο.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *PrintInvoice*.
3. Το τιμολόγιο ανοίγει στον προεπιλεγμένο επεξεργαστή κειμένου του υπολογιστή σας, έτοιμο για εκτύπωση.

6.10.7. Εκτύπωση απόδειξης

1. Από τον πίνακα παραγγελιών, επιλέξτε την παραγγελία για την οποία επιθυμείτε να εκτυπώσετε απόδειξη.
2. Από την εργαλειοθήκη, πατήστε το κουμπί *PrintReceipt*.
3. Η απόδειξη ανοίγει στον προεπιλεγμένο επεξεργαστή κειμένου του υπολογιστή σας, έτοιμη για εκτύπωση.

6.11. Εξαγωγή επαφών συστήματος σε αρχείο



1. Πατήστε στο εικονίδιο *Exportcontacts* στην κεντρική φόρμα επιλογών για να εμφανιστεί η φόρμα *Exportcontacts*.
2. Πιέστε το κουμπί *Browse...* για να επιλέξετε το φάκελο στον οποίο θα αποθηκευτούν οι εξαχθείσες επαφές.
3. Επιλέξτε τους πίνακες τους οποίους θέλετε να εξάγετε και πατήστε *Next*.
4. Επιλέξτε από τους πίνακες τα στοιχεία που θέλετε να εξαχθούν και πατήστε *Finish*. Μπορείτε να επιλέξετε όλα τα στοιχεία, πατώντας *SelectAll*.
5. Εμφανίζεται μήνυμα επιβεβαίωσης.

Το αρχείο βρίσκεται στο φάκελο που επιλέξατε στο βήμα 2. Μπορείτε να μεταβείτε στο προηγούμενο βήμα, πατώντας το κουμπί *Back*.



6.12. Εισαγωγή επαφών στο σύστημα από αρχείο

1. Πατήστε στο εικονίδιο *Importcontacts* στην κεντρική φόρμα επιλογών για να εμφανιστεί η φόρμα *Importcontacts*.
2. Πιέστε το κουμπί *Browse...* για να επιλέξετε το αρχείο από το οποίο θα εισαχθούν οι νέες εγγραφές στο σύστημα.
3. Επιλέξτε τον πίνακα στον οποίο θέλετε να εισάγετε τις νέες εγγραφές και πατήστε *Next*.
4. Αντιστοιχίστε τα πεδία του αρχείου με τις στήλες του επιλεγμένου πίνακα, σύροντας το πεδίο πάνω στη στήλη που θέλετε να το αντιστοιχίσετε. Πατήστε *Finish*. Μπορείτε να αντιστοιχίσετε εκ νέου τα πεδία, πατώντας *ClearAll*.
5. Εμφανίζεται μήνυμα επιβεβαίωσης.

Αν κάποια εγγραφή του αρχείου υπάρχει ήδη στον επιλεγμένο πίνακα, δεν καταχωρείται. Μπορείτε να μεταβείτε στο προηγούμενο βήμα, πατώντας το κουμπί *Back*.



6.13. Έξυπνες πωλήσεις

1. Πατήστε στο εικονίδιο *SmartSeller* στην κεντρική φόρμα επιλογών για να εμφανιστεί η φόρμα *SmartSeller*.
2. Καθορίστε τον πελάτη για τον οποίο θέλετε να εισηγηθείτε προϊόντα.
3. Καθορίστε τα κριτήρια στα πεδία των συλλογών προϊόντων, κατηγοριών και χρώματος.
4. Πατήστε το κουμπί *Suggest*.

Κεφάλαιο 7. Αξιολόγηση

7.1. ΕΙΣΑΓΩΓΗ.....	151
7.2. ΕΥΧΡΗΣΤΙΑ.....	151
7.3. ΑΝΑΛΥΣΗ ΕΡΩΤΗΜΑΤΟΛΟΓΙΩΝ.....	151
7.3.1. Χαμηλή βαθμολόγηση	
7.3.2. Μελλοντική δουλειά	

7.1. ΕΙΣΑΓΩΓΗ

Όταν ένα σύστημα ή πρόγραμμα παραδοθεί, πρέπει να γίνει αξιολόγησή του. Η αξιολόγηση ενός λογισμικού είναι η καταμέτρηση της επίδοσης του και της αποδοτικότητας του. Μια εταιρεία ή μια επιχείρηση, όταν αναπτύσσει ένα λογισμικό, προτού το προωθήσει στην αγορά, πρέπει να είναι σε θέση να γνωρίζει αν το λογισμικό του, μπορεί να πετύχει τους στόχους ενός οργανισμού. Πρέπει να γνωρίζει ποια είναι τα μειονεκτήματά του, τι υπολείπεται και ποια είναι τα πλεονεκτήματά του. Μια εταιρεία πρέπει να είναι σε θέση να γνωρίζει αν το λογισμικό του ικανοποιεί τις ανάγκες του πελάτη της. Σε όλα αυτά, δίνει απάντηση η αξιολόγηση. Γι' αυτό το λόγο είναι σημαντικό να γίνεται, έτσι ώστε να υπάρχουν περισσότερες πιθανότητες επιτυχίας μιας ιστοσελίδας.

Σημαντική παράμετρος στην αξιολόγηση λογισμικού είναι η ευχρηστία. Ένα υλοποιημένο σύστημα που ανταποκρίνεται στις προσδοκίες του χρήστη και ικανοποιεί τις απαιτήσεις του, κάνει τους χρήστες πιο αποδοτικούς όσο αφορά την αλληλεπίδραση τους με την εφαρμογή. Συνεπώς είναι πιθανότερο ο χρήστης να ολοκληρώνει την εργασία του συντομότερο χρονικό διάστημα πό προηγουμένως. Οπότε η καλή ευχρηστία ενός συστήματος είναι σημαντικότερη όσο αφορά την όλη επιτυχία του λογισμικού.

7.2. ΕΥΧΡΗΣΤΙΑ

Ευχρηστία είναι η δυνατότητα ενός προϊόντος/συστήματος ή υπηρεσίας που χρησιμοποιείται από καθορισμένους χρήστες με καθορισμένους στόχους, υπό καθορισμένες συνθήκες χρήσης να είναι αποτελεσματικό (effectiveness), αποδοτικό (efficiency) και να παρέχει υποκειμενική ικανοποίηση (satisfaction) στους χρήστες του. Η ευχρηστία ενός συστήματος εξετάζεται με βάση αρκετές παραμέτρους.

Στην αξιολόγηση ενός συστήματος, εξετάζεται η ευχρηστία του συνήθως μέσω ερωτηματολογίων, που αφορούν το λογισμικό και προσωπικών συνεντεύξεων. Η αξιολόγηση πρέπει να γίνεται για όλα τα μέρη του λογισμικού για να προκύψει μια πλήρης εικόνα για την κατάσταση του λογισμικού. Μέσω των ερωτηματολογίων και των συνεντεύξεων εξετάζονται πτυχές όπως η πλοήγηση στο σύστημα, η αποδοτικότητα του συστήματος, η σχεδίαση του συστήματος και η λειτουργικότητα του όσον αφορά τις λειτουργίες που παρέχει.

Με βάση τη βαθμολόγηση που τυγχάνουν οι ερωτηθέντες στα διάφορα τμήματα του ερωτηματολογίου, γίνεται εξαγωγή συμπερασμάτων. Αν για παράδειγμα ερωτήσεις που αναφέρονται σε σημεία της εφαρμογής συγκέντρωσαν χαμηλή βαθμολογία από πολλούς χρήστες, συμπεραίνεται πρόβλημα και αδυναμία του συστήματος και πρέπει να τύχει διόρθωσης.

7.3. ΑΝΑΛΥΣΗ ΕΡΩΤΗΜΑΤΟΛΟΓΙΩΝ

7.3.1. Χαμηλή βαθμολόγηση

Σε κάποιες ερωτήσεις που αξιολογούν σημαντικά σημεία του συστήματος, παρουσιάστηκε χαμηλή βαθμολόγηση. Οι ερωτήσεις απαντιόνταν με βαθμολόγηση από το 1 έως το 5, όπου 1 = καθόλου, 2 = λίγο, 3 = μέτρια, 4 = πολύ και 5 = πάρα πολύ.

Χαρακτηριστικά Συστήματος	1	2	3	4	5
1. Η ταχύτητα απόκρισης και εκτέλεσης εργασιών είναι ικανοποιητική για τον χρήστη;	0	0	0	0,56	0,44
2. Σε ποιο βαθμό παρέχεται η δυνατότητα ανάνηψης από εσφαλμένους χειρισμούς χρήστη (undo);	0,67	0,33	0	0	0
3. Πόσο κατανοητά είναι τα μηνύματα σφάλματος από τον χρήστη;	0	0	0	0,11	0,89
4. Πόσο εύκολη είναι η εκτέλεση εργασιών από άπειρους χρήστες;	0	0	0,11	0,44	0,44
Λειτουργία Συστήματος	1	2	3	4	5
1. Σε ποιο βαθμό παρέχεται πρόσβαση στα δεδομένα μόνο σε εξουσιοδοτημένα άτομα;	0,56	0,44	0	0	0
2. Πόσο εύκολη είναι η εισαγωγή, επεξεργασία ή διαγραφή δεδομένων;	0	0	0	0,67	0,33
3. Σε ποιο βαθμό παρέχεται δυνατότητα ελέγχου της δραστηριότητας στο σύστημα;	0	0	0	0,67	0,33
4. Σε ποιο βαθμό υπάρχει η δυνατότητα ακύρωσης μιας πράξης; (cancel)	0	0	0	0,56	0,44
Εγχειρίδιο χρήσης	1	2	3	4	5
1. Πόσο σαφές και υποβοηθητικό είναι το εγχειρίδιο χρήσης;	0	0	0	0	1,00
2. Κατά πόσο η επίλυση ενός προβλήματος με χρήση του εγχειριδίου είναι εύκολη;	0	0	0,33	0,44	0,22
3. Πόσο εύκολη είναι η χρήση του συστήματος χωρίς τη βοήθεια του εγχειριδίου χρήσης;	0	0	0	0,56	0,44
4. Πόσο βοηθητική είναι η θέση των βοηθητικών επεξηγήσεων στην οθόνη;	0	0	0,56	0,22	0,22
Πολυμέσα	1	2	3	4	5
1. Πόσο ικανοποιητική είναι η ποιότητα και η ευκρίνεια των εικόνων που χρησιμοποιούνται στο σύστημα;	0	0	0	0,22	0,78
2. Πόσο ευανάγνωστες είναι οι συμβολοσειρές και το μέγεθος των γραμμάτων που χρησιμοποιούνται;	0	0	0	0,33	0,67
3. Πόσο ξεκούραστα για το χρήστη είναι τα χρώματα που χρησιμοποιούνται;	0	0	0	0,67	0,33
4. Πόσο κατανοητή είναι η γλώσσα που χρησιμοποιείται;	0	0	0	0,44	0,56
Παρουσίαση Πληροφορίας	1	2	3	4	5
1. Σε ποιο βαθμό το περιεχόμενο της σελίδας είναι καλά διατεταγμένο;	0	0	0	0,33	0,67
2. Η ακολουθία των οθονών δεν προκαλεί σύγχυση στον χρήστη;	0	0	0,56	0,33	0,11
3. Σε ποιο βαθμό η επόμενη οθόνη στην ακολουθία είναι προβλέψιμη;	0	0	0	0,33	0,67
4. Πόσο εύκολη είναι η επιστροφή στην προηγούμενη οθόνη;	0	0	0	0,22	0,78

Ορολογία - Πληροφορίες	1	2	3	4	5
1. Πόσο σχετική είναι η ορολογία που χρησιμοποιείται με το αντικείμενο εφαρμογής;	0	0	0	0,56	0,44
2. Πόσο συνεπείς είναι τα μηνύματα και οι άλλες πληροφορίες που εμφανίζονται στην οθόνη;	0	0	0	0,44	0,56
3. Πόσο σαφές είναι το περιεχόμενο των μηνυμάτων στην οθόνη;	0	0	0	0,33	0,67
Κατάλογος Προϊόντων	1	2	3	4	5
1. Πόσο εύκολη είναι η πρόσβαση στον κατάλογο των προϊόντων;	0	0	0,11	0,33	0,56
2. Πόσο λογική είναι η δομή και η οργάνωση του καταλόγου;	0	0	0	0,89	0,11
3. Πόσο εύκολη είναι η πλοήγηση στον κατάλογο προϊόντων;	0	0	0	0,33	0,67
4. Σε ποιο βαθμό οι πληροφορίες των προϊόντων ενθαρρύνουν τον επισκέπτη να επιλέξει εύκολα;	0	0	0,11	0,78	0,11
Συναλλαγές	1	2	3	4	5
1. Πόσο κατανοητή είναι η φόρμα παραγγελίας;	0	0	0	0,67	0,33
2. Πόσο γρήγορα ολοκληρώνεται η συναλλαγή;	0	0	0	0,89	0,11
3. Εμφανίζεται κάποια επιβεβαίωση παραγγελίας/αγοράς;	0	0	0	0,67	0,33
4. Πόσο εύκολη είναι η πρόσθεση ή αφαίρεση προϊόντος σε παραγγελία;	0	0	0	0,33	0,67

Πίνακας 7. 1

Όπως φαίνεται από τον πίνακα 7.1, παρουσιάστηκε χαμηλή βαθμολόγηση, στην αξιολόγηση των πιο κάτω:

- Χαρακτηριστικά συστήματος: Ανάνηψη (Undo)
- Λειτουργίες συστήματος: Εξουσιοδοτημένη πρόσβαση
- Λειτουργίες συστήματος: Προβολή Προϊόντων

7.3.1.1. Χαρακτηριστικά συστήματος: Ανάνηψη (Undo)

Όσον αφορά την αξιολόγηση της αποδοτικότητας των χαρακτηριστικών του συστήματος, η χαμηλή βαθμολόγηση προέκυψε στη δεύτερη ερώτηση, η οποία αφορά τη δυνατότητα ανάνηψης από εσφαλμένους χειρισμούς του χρήστη (undo). Το πρόβλημα αυτό προέκυψε κατά τη διαγραφή κάποιου δεδομένου από το σύστημα. Οι περισσότεροι χρήστες ρώτησαν αν μπορούσαν να την ανακτήσουν πίσω. Η αρνητική απάντηση προκάλεσε και αρνητικά σχόλια.

7.3.1.2. Λειτουργίες συστήματος: Εξουσιοδοτημένη πρόσβαση

Οι χρήστες που απάντησαν στο ερωτηματολόγιο ανέφεραν ότι θα ήταν πιο σωστό να υπάρχει οθόνη εισόδου για εισαγωγή ψευδώνυμου και κωδικού πρόσβαση για καλύτερη ασφάλεια στο σύστημα. Έτσι θα μπορεί να καταγράφεται από ποιον έγιναν οι τροποποιήσεις στο σύστημα, παρέχοντας καλύτερο έλεγχο στα δεδομένα του συστήματος.

7.3.1.3. Λειτουργίες συστήματος: Προβολή Προϊόντων

Επιπρόσθετα με τα προβλήματα που παρουσιάστηκαν πιο πάνω, οι χρήστες που απάντησαν στο ερωτηματολόγιο, κατά τη διάρκεια της αξιολόγησης ανέφεραν την ανάγκη παροχής του καταλόγου προϊόντων του συστήματος σε έντυπη μορφή. Δηλαδή τη δυνατότητα εκτύπωσης του καταλόγου με τα προϊόντα.

7.3.2. Μελλοντική δουλειά

Γιατο σύστημα που υλοποιήθηκε, έγιναν κάποιες εισηγήσεις για βελτίωση της λειτουργικότητας της εφαρμογής. Οι εισηγήσεις που προτάθηκαν είναι οι εξής:

- Υλοποίηση λειτουργίας Βοήθειας Χρήστη
- Μετατροπή σε εφαρμογή ιστού
- Αγορά υπηρεσιών φιλοξενίας στο διαδίκτυο

7.3.2.1. Υλοποίηση λειτουργίας Βοήθειας Χρήστη

Η υλοποίηση της λειτουργίας αυτής ενισχύει την ευχρηστία του συστήματος. Η παροχή βοήθειας στο χρήστη αποτελεί σημαντικό κομμάτι για την ικανοποίηση του χρήστη. Προσφέρει στο χρήστη τη δυνατότητα καλύτερης κατανόησης των δυνατοτήτων που του προσφέρει το σύστημα, μέσω των λειτουργιών της εφαρμογής. Αυτό έχει ως αποτέλεσμα την αποδοτικότερη χρήση του συστήματος από τον χρήστη και ενίσχυση της ικανοποίησης των απαιτήσεών του.

Η λειτουργία αυτή θα δίνει περισσότερες πληροφορίες σχετικά με το πώς ο χρήστης θα μπορεί να πλοηγηθεί στο σύστημα και πώς θα μπορεί να χρησιμοποιεί τις λειτουργίες που του παρέχονται για την παραγγελία προϊόντων.

7.3.2.2. Μετατροπή σε εφαρμογή ιστού

Το ηλεκτρονικό εμπόριο εισέβαλε τα τελευταία χρόνια στον χώρο των επιχειρήσεων διευρύνοντας τα οφέλη των εταιριών από τις online πωλήσεις και παρέχοντας νέους, περισσότερο λειτουργικούς τρόπους διαχείρισης των πωλήσεων. Οι πλατφόρμες ηλεκτρονικού εμπορίου εξαλείφουν τους χρονικούς και γεωγραφικούς περιορισμούς, καθώς οι συναλλαγές πραγματοποιούνται 24 ώρες το 24ωρο, 7 ημέρες την εβδομάδα και από οποιοδήποτε σημείο του πλανήτη. Κανένας από τους συναλλασσόμενους δεν χρειάζεται να ταξιδέψει, αφού η διαπραγμάτευση γίνεται από το ίδιο τους το γραφείο, κάτι που μεταφράζεται σε σημαντική εξοικονόμηση χρόνου και χρήματος.

Τα οφέλη μιας επιχείρησης από την είσοδο της στον κόσμο του ηλεκτρονικού εμπορίου είναι πολλά μεταξύ των οποίων:

- Αύξηση των πωλήσεων
- Βέλτιστη διαχείριση των προϊόντων και παραγγελιών
- Άμεση ικανοποίηση των αναγκών των πελατών
- Μεγαλύτερη ασφάλεια στις συναλλαγές
- Ακρίβεια στην ενημέρωση των πελατών για καινούργια προϊόντα και άμεση προώθηση τους στην αγορά
- Βελτίωση της επικοινωνίας με τους πελάτες

7.3.2.3. Αγορά υπηρεσιών φιλοξενίας στο διαδίκτυο

Για τη δημοσίευση της εφαρμογής στον Παγκόσμιο Ιστό, είναι απαραίτητη η αγορά υπηρεσιών φιλοξενίας στο διαδίκτυο από κάποιο παροχέα. Με βάση τα όσα περιγράφηκαν σε προηγούμενο κεφάλαιο, προκύπτει ότι η Amazon είναι η πιο συμφέρουσα επιλογή. Σε σύγκριση με τους άλλους παροχείς, είναι μια από τις ασφαλέστερες υπηρεσίες για εφαρμογές διαδικτύου. Ο πιο φθηνός από άποψη κόστους παροχέας υπηρεσιών φιλοξενίας στο διαδίκτυο είναι η AWS (Amazon's Web Services - Υπηρεσίες Διαδικτύου της Amazon), αφού χρεώνει τους πελάτες της για τις υπηρεσίες που τους προσφέρει, ανάλογα με τους πόρους που αυτοί καταναλώνουν στο σύστημά της.

Βιβλιογραφία

1. «Το ψηφιακό εμπόριο η επιλογή του μέλλοντος». Ανακτήθηκε 17:42, 12 Μαΐου 2011, από www.gba.gr/ilektroniko-emporio/to-psifiako-emporio-i-epilogi-toy-mellontos.html
2. «Παραδοσιακή και δικτυακή διαφήμιση. Ομοιότητες - Διαφορές – Προτάσεις». Ανακτήθηκε 17:50, 12 Μαΐου 2011, από www.eeei.gr
3. «Εισαγωγή στην επιχειρηματικότητα». Ανακτήθηκε 18:07, 12 Μαΐου 2011, από www.career.tuc.gr/05_01.html
4. «Amazon.com». Ανακτήθηκε 09:56, 13 Μαΐου 2011, από <http://en.wikipedia.org/wiki/Amazon.com>
5. «eBay». Ανακτήθηκε 10:12, 13 Μαΐου 2011, από <http://en.wikipedia.org/wiki/EBay>
6. «ASOS». Ανακτήθηκε 10:20, 13 Μαΐου 2011, από <http://en.wikipedia.org/wiki/ASOS.com>
7. E-shop. Ανακτήθηκε 10:24, 13 Μαΐου 2011, από www.e-shop.gr
8. Παπασωτηρίου. Ανακτήθηκε 10:30, 13 Μαΐου 2011, από www.papasotiriou.gr
9. Amazon. Ανακτήθηκε 10:37, 13 Μαΐου 2011, από www.amazon.com
10. Asos. Ανακτήθηκε 10:48, 13 Μαΐου 2011, από www.asos.com
11. eBay. Ανακτήθηκε 11:05, 13 Μαΐου 2011, από www.ebay.com
12. «Data Types (Transact-SQL)». Ανακτήθηκε 16:06, 13 Μαΐου 2011, από <http://msdn.microsoft.com/en-us/library/ms187752.aspx>
13. «HoweBayworks». Ανακτήθηκε 13:38, 13 Μαΐου 2011, από <http://www.bizhelp24.com/basics/how-ebay-works.html>
14. «HowAmazonworks». Ανακτήθηκε 13:50, 13 Μαΐου 2011, από <http://www.bizhelp24.com/basics/how-ebay-works.html>
15. «Amazon Elastic Compute Cloud, Amazon». Ανακτήθηκε 14:03, 13 Μαΐου 2011, από <http://aws.amazon.com/ec2/>
16. «Amazon Relational Database Service, Amazon». Ανακτήθηκε 14:28, 13 Μαΐου 2011, από <http://aws.amazon.com/rds/#pricing>
17. «Storing Images In MySQL». Ανακτήθηκε 11:39, 22 Ιανουαρίου 2011, από <http://www.phpriot.com/articles/images-in-mysql>
18. «Talk on eBay architecture». Ανακτήθηκε 11:27, 26 Μαΐου 2011, από <http://glinden.blogspot.com/2006/12/talk-on-ebay-architecture.html>
19. «eBay Creates Technology Architecture for the Future». Ανακτήθηκε 11:41, 26 Μαΐου 2011, από http://www.sun.com/service/about/success/recent/Sun_eBay6-2_forWeb.pdf
20. «HoweBay Works». Ανακτήθηκε 11:54, 26 Μαΐου 2011, από <http://www.addsimplicity.com/downloads/eBaySDForum2006-11-29.pdf>
21. «HoweBay Works». Ανακτήθηκε 12:13, 26 Μαΐου 2011, από <http://computer.howstuffworks.com/ebay2.htm>

22. «HowAmazon Works». Ανακτήθηκε 12:36, 26 Μαΐου 2011, από <http://money.howstuffworks.com/amazon1.htm>
23. «Επιχειρησιακή Δικτύωση και Ηλεκτρονικό Εμπόριο». Ανακτήθηκε 20:18, 25 Μαΐου 2011, από www.e-businessforum.gr
24. «Νέες τεχνολογίες και Διαφήμιση». Ανακτήθηκε 20:39, 25 Μαΐου 2011, από www.e-businessforum.gr
25. «Collaborative Filtering». Ανακτήθηκε 19:24, 27 Μαΐου 2011, από www.cs.umd.edu/~samir/498/CMSC498K_Hyoungtae_Cho.ppt

ΠΑΡΑΡΤΗΜΑ Α

A.1. Κώδικας ενημέρωσης ονομαστικών εορτών

A.1.1. Συνάρτηση udf_modulo

```
CREATEFUNCTION [dbo].[udf_modulo](@xINT, @yINT)
RETURNSINT
AS
BEGIN
DECLARE @mod INT
SET @mod = @x - @y *FLOOR(CAST(@x ASFLOAT)/ @y)
RETURN @mod
END
```

A.1.2. Συνάρτηση udf_CalculateEaster

```
CREATEFUNCTION [dbo].[udf_CalculateEaster](@Year INT)
RETURNSDATETIME
AS
BEGIN
DECLARE @Date DATETIME
DECLARE @GoldenNumber INT
DECLARE @EpactCalc INT
DECLARE @PaschalDaysCalc INT
DECLARE @PaschalWeekday INT
DECLARE @NumOfDaysToSunday INT
DECLARE @EasterMonth INT
DECLARE @EasterDay INT
IF @Year >= 1900 AND @year <= 2099
BEGIN
-- Golden Number of Year. The Phases of the moon repeat
-- once every 19 years, so each year is associated with
-- a Golden Number between 1 and 19 representing this
-- relationship
SET @GoldenNumber = dbo.udf_Modulo(@Year + 1, 19)

-- Calculation based on Epact. Epact is the measure of
-- the "age" of the moon (i.e., the number of days since the
-- "official" new moon at the start of the year)
SET @EpactCalc = dbo.udf_Modulo(5 + 19 *(@GoldenNumber), 30)

-- Number of days from March 21 to Paschal Full Moon. The
-- Paschal Full Moon is the full moon that is immediately
-- before Easter.
SET @PaschalDaysCalc = @EpactCalc -(@EpactCalc / 28)

-- Calculate Weekday of Paschal Full Moon (0 = Sunday,
-- 1 = Monday)
SET @PaschalWeekday = dbo.udf_Modulo(@Year + @Year / 4 + @PaschalDaysCalc -
13, 7)

-- Calculate Number of Days from March 21 to Sunday
-- on or before Paschal Full Moon
SET @NumOfDaysToSunday = @PaschalDaysCalc - @PaschalWeekday
```

```

-- Calculate Easter Month
SET @EasterMonth = 3 + (@NumOfDaysToSunday + 40) / 44

-- Calculate Easter Day
SET @EasterDay = @NumOfDaysToSunday + 28 - 31 * (@EasterMonth / 4)

-- Return Date
SET @Date = CAST(@Year AS VARCHAR) + '-' + CAST(@EasterMonth AS VARCHAR) + '-'
            + CAST(@EasterDay AS VARCHAR)
END
RETURN @Date
END

```

A.1.3. Αποθηκευμένη διαδικασία sp_CalculateHolidays

```

CREATEPROCEDURE [dbo].[sp_CalculateHolidays]
AS
BEGIN
    UPDATE dbo.Namedays
    SET [date] = DATEADD(year, 1, [date])
    WHERE datepart(year, [date]) != datepart(year, getdate())

    DECLARE @date AS DATE
    DECLARE @year AS INT
    SET @year = datepart(year, getdate())
    DECLARE @easter AS DATE
    SET @easter = dbo.udf_CalculateEaster(@year)

    -- Saint Chloe
    SET @date = cast(('02/13/' + cast(@year as nvarchar)) as datetime)

    IF DATENAME(dw, @date) = 'Sunday'
        UPDATE dbo.Namedays SET [date] = @date WHERE notes = 'chloe'
    ELSE
    BEGIN
        WHILE (DATENAME(dw, @date) != 'Sunday')
        BEGIN
            SET @date = cast(cast(cast(@date as datetime) as integer) + 1
as datetime)
        END
        UPDATE dbo.Namedays SET [date] = @date WHERE notes = 'chloe'
    END

    -- Sunday of Ancestors
    SET @date = cast(('12/11/' + cast(@year as nvarchar)) as datetime)

    IF DATENAME(dw, @date) = 'Sunday'
        UPDATE dbo.Namedays SET [date] = @date WHERE notes = 'ancestors'
    ELSE
    BEGIN
        WHILE (DATENAME(dw, @date) != 'Sunday')
        BEGIN
            SET @date = cast(cast(cast(@date as datetime) as integer) + 1
as datetime)
        END
    END

```

```

        END
        UPDATE dbo.Namedays SET [date]=@date WHERE notes='ancestors'
    END

    --Saint George
    SET @date =cast(('04/23/'+cast(@year asnvarchar))asdatetime)

    IFDATEDIFF(day, @date, @easter)< 0
        UPDATE dbo.Namedays SET [date]=@date WHERE notes='george'
    ELSE
    BEGIN
        SET @date =cast(cast(cast(@easter asdatetime)asinteger)+1
asdatetime)
        UPDATE dbo.Namedays SET [date]=@date WHERE notes='george'
    END

    --Apostole Marcus
    SET @date =cast(('04/23/'+cast(@year asnvarchar))asdatetime)

    IFDATEDIFF(day, @date, @easter)< 0
        UPDATE dbo.Namedays SET
[date]=CAST('04/25/2011'asdatetime)WHERE notes='marcus'
    ELSE
    BEGIN
        SET @date =cast((cast(cast(@easter
asdatetime)asinteger)+2)asdatetime)
        UPDATE dbo.Namedays SET [date]=@date WHERE notes='marcus'
    END

    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-43)asdatetime)WHERE notes='theodore'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-42)asdatetime)WHERE notes='orthodoxy'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-35)asdatetime)WHERE notes='gregory'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-8)asdatetime)WHERE notes='lazarus'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-7)asdatetime)WHERE notes='palm'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-6)asdatetime)WHERE notes='holymon'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)-0)asdatetime)WHERE notes='easter'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)+2)asdatetime)WHERE notes='lamprotriti'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)+3)asdatetime)WHERE notes='theocharis'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)+5)asdatetime)WHERE notes='pigi'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)+7)asdatetime)WHERE notes='thomas'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter
asdatetime)asinteger)+14)asdatetime)WHERE notes='fragrant'
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter

```

```
asdatetime)asinteger)+21)asdatetime)WHERE notes='paralyzed'  
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter  
asdatetime)asinteger)+39)asdatetime)WHERE notes='ascension'  
    UPDATE dbo.Namedays SET [date]=cast((cast(cast(@easter  
asdatetime)asinteger)+50)asdatetime)WHERE notes='holyspirit'  
END
```

A.2. Κώδικας αποστολής ευχετήριας κάρτας με ηλεκτρονικό ταχυδρομείο

A.2.1. Διαδικασία sendmail()

```
PrivateSub sendmail(ByVal receiverList AsString, ByVal wish AsString)
Dim mailobj As MailMessage
Dim smtp AsNew SmtplibClient
Dim Sql AsString
Dim sqlcomm As SqlCommand
Dim r As SqlDataReader

    SqlConnection.Open()
    Sql = "SELECT * FROM Settings"
    sqlcomm = New SqlCommand(Sql, SqlConnection)
    r = sqlcomm.ExecuteReader()
    r.Read()

    IfCBool(r("Send Email?")) Then
        mailobj = New MailMessage()
        mailobj.From = New MailAddress(r("Email Address"))
        mailobj.To.Add(receiverList)
        mailobj.Subject = "Wishes / Ευχές"

    If wish.Equals("Birthday") Then
        mailobj.Body = r("Birthday Wish")
    Else
        mailobj.Body = r("Nameday Wish")
    EndIf

        smtp.Host = r("Smtplib Host")
        smtp.Port = CInt(r("Smtplib Port"))
        smtp.EnableSsl = True
        smtp.Credentials = New Net.NetworkCredential(mailobj.From.ToString,
r("Email Password"))
        smtp.Send(mailobj)

    EndIf

    r.Close()
    SqlConnection.Close()
EndSub
```

A.2.2. Διαδικασία Today_Birthdays()

```
PrivateSub Today_Birthdays()
Dim receiverList AsString = String.Empty

    SqlConnection.Open()
Dim Sql AsString =
"SELECT [Full Name], Email"& _
" FROM Customers"& _
" WHERE (datepart(day, [BirthDate]) = datepart(day, @date))"& _
" AND (datepart(month, [BirthDate]) = datepart(month, @date))"
Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
    sqlcomm.Parameters.Add(New SqlParameter("@date",
```

```

SqlDbType.Date)).Value = MonthCalendar1.SelectionStart

Dim r As SqlDataReader = sqlcomm.ExecuteReader()

If r.HasRows() Then
While r.Read()
    receiverList = receiverList + r("Email") + ", "
EndWhile
    receiverList = receiverList.Remove(receiverList.Length - 2, 2)
EndIf
    r.Close()
    SqlConnection.Close()

If (Not receiverList.Equals(String.Empty)) And sent = False Then
    sent = True
    sendmail(receiverList, "Birthday")
EndIf
EndSub

```

A.2.3. Διαδικασία Today_Namedays()

```

PrivateSub Today_Namedays()
Dim receiverList AsString = String.Empty

    SqlConnection.Open()
Dim Sql AsString =
"SELECT [Full Name], Email"& _
" FROM Customers C JOIN Namedays N ON C.Nameday = N.id"& _
" WHERE (datepart(day, N.[date]) = datepart(day, @date))"& _
" AND (datepart(month, N.[date]) = datepart(month, @date))"
Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
    sqlcomm.Parameters.Add(New SqlParameter("@date",
SqlDbType.Date)).Value = MonthCalendar1.SelectionStart

Dim r As SqlDataReader = sqlcomm.ExecuteReader()
If r.HasRows() Then
While r.Read()
    receiverList = receiverList + r("Email") + ", "
EndWhile
    receiverList = receiverList.Remove(receiverList.Length - 2, 2)
EndIf
    r.Close()
    SqlConnection.Close()

If (Not receiverList.Equals(String.Empty)) And sent1 = False Then
    sent1 = True
    sendmail(receiverList, "Nameday")
EndIf
EndSub

```

A.3. Ενημέρωση αποθήκης πρώτων υλών και αυτόματη προειδοποίηση εξάντλησης πρώτης ύλης

A.3.1. Σκανδάλες πίνακα Φορεμάτων

i) *tr_Dress_insert*

```
-- Description: Update Raw Materials
--               on Insert new Dress Manufacturing Details
CREATETRIGGER [dbo].[tr_Dress_insert]
ON [dbo].[Dress Manufacturing Details]
AFTERINSERT
AS
DECLARE @quantity int, @code nvarchar(10)

BEGIN
SELECT @quantity=Quantity, @code=[Raw Material Code]
FROM inserted

UPDATE dbo.[Raw Material]
SET Stock = Stock - @quantity
WHERE Code = @code
END
```

ii) *tr_Dress_update*

```
-- Description: Update Raw Materials
--               on Update Dress Manufacturing Details
CREATETRIGGER [dbo].[tr_Dress_update]
ON [dbo].[Dress Manufacturing Details]
AFTERUPDATE
AS
DECLARE @oldquantity int, @newquantity int, @code nvarchar(10)

BEGIN
SELECT @oldquantity=Quantity, @code=[Raw Material Code]
FROM deleted

SELECT @newquantity=Quantity, @code=[Raw Material Code]
FROM inserted

UPDATE dbo.[Raw Material]
SET Stock = Stock + @oldquantity - @newquantity
WHERE Code = @code
END
```

iii) *tr_Dress_delete*

```
-- Description: Update Raw Materials
--               on Delete Dress Manufacturing Details
CREATETRIGGER [dbo].[tr_Dress_delete]
ON [dbo].[Dress Manufacturing Details]
AFTERDELETE
AS
DECLARE @quantity int, @code nvarchar(10)

BEGIN
```

```

        SELECT @quantity=Quantity, @code=[Raw Material Code]
FROM deleted

UPDATE dbo.[Raw Material]
SET Stock = Stock + @quantity
WHERE Code = @code
END

```

A.3.2. Σκανδάλες πίνακα Ειδών Ένδυσης

i) *tr_Garment_insert*

```

-- Description: Update Raw Materials
--               on Insert new Garment Manufacturing Details
CREATETRIGGER [dbo].[tr_Garment_insert]
ON [dbo].[Garment Manufacturing Details]
AFTERINSERT
AS
DECLARE @quantity int, @code nvarchar(10)

BEGIN
    SELECT @quantity=Quantity, @code=[Raw Material Code]
    FROM inserted

    UPDATE dbo.[Raw Material]
    SET Stock = Stock - @quantity
    WHERE Code = @code
END

```

ii) *tr_Garment_update*

```

-- Description: Update Raw Materials
--               on Update Garment Manufacturing Details
CREATETRIGGER [dbo].[tr_Garment_update]
ON [dbo].[Garment Manufacturing Details]
AFTERUPDATE
AS
DECLARE @oldquantity int, @newquantity int, @code nvarchar(10)

BEGIN
    SELECT @oldquantity=Quantity, @code=[Raw Material Code]
    FROM deleted

    SELECT @newquantity=Quantity, @code=[Raw Material Code]
    FROM inserted

    UPDATE dbo.[Raw Material]
    SET Stock = Stock + @oldquantity - @newquantity
    WHERE Code = @code
END

```

iii) *tr_Garment_delete*

```

-- Description: Update Raw Materials
--               on Delete Garment Manufacturing Details
CREATETRIGGER [dbo].[tr_Garment_delete]
ON [dbo].[Garment Manufacturing Details]

```

```

AFTERDELETE
AS
DECLARE @quantity int, @code nvarchar(10)

BEGIN
    SELECT @quantity=Quantity, @code=[Raw Material Code]
FROM deleted

UPDATE dbo.[Raw Material]
SET Stock = Stock + @quantity
WHERE Code = @code
END

```

A.3.3. Διαδικασία Running_Out()

```

PublicSub Running_Out()
Me.ErrorProvider1.Clear()
    frmRawMaterials.txt_runout.Text = ""

    SqlConnection.Open()
Dim Sql AsString =
"SELECT RM.Code, RM.Description, RM.Stock, RM.[Measurement Unit]& _
" FROM [Raw Material] RM JOIN Supplier S ON RM.Supplier = S.ID"& _
" WHERE Stock <= [Min quantity]"
Dim sqlcomm AsNew SqlCommand(Sql, SqlConnection)
Dim r As SqlDataReader = sqlcomm.ExecuteReader()

If r.HasRows() Then
Me.ErrorProvider1.SetError(PictureBox9, "Raw materials are running out")

While r.Read()
    frmRawMaterials.txt_runout.Text =
frmRawMaterials.txt_runout.Text + r("Code") + " - " +
    r("Description") + ": " + CInt(r("Stock")).ToString + " " +
    r("Measurement Unit") + " left" + vbNewLine
EndWhile

EndIf
    r.Close()
    SqlConnection.Close()
EndSub

```

A.4. Αποθήκευση εικόνων

A.4.1. Διαδικασία ConvertImage()

```
'Convert an image to an array of bytes
PublicFunction ConvertImage(ByVal myImage As Image) AsByte()
Dim mstream AsNew MemoryStream()
    myImage.Save(mstream, System.Drawing.Imaging.ImageFormat.Jpeg)

Dim myBytes(mstream.Length - 1) AsByte
    mstream.Position = 0

    mstream.Read(myBytes, 0, mstream.Length)

Return myBytes
EndFunction
```

A.5. Εκτύπωση αποδείξεων και τιμολογίων

A.5.1. Παράδειγμα σύνδεσης κώδικα VisualBasic με MicrosoftWord

```
Imports Microsoft.Office.Interop

'create a word object
Dim objWord As Word.Application
objWord = New Word.Application
objWord.Visible = True

'Load in a document from a predesigned template.
Dim mydocs AsString =
System.Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments)
objWord.Documents.Add(Template:=mydocs + "\Print\INVOICE.dotx", NewTemplate:=False)

'adding text to a document from a specified point (bookmark)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="InvoiceID")
objWord.Selection.TypeText(orderIDtxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="OrderDate")
objWord.Selection.TypeText(orderdatetxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="DueDate")
objWord.Selection.TypeText(Due_DateTextBox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="FullName")
objWord.Selection.TypeText(Full_NameTextBox.Text)

SqlConnection.Open()
Dim sqlcomm = New SqlCommand("SELECT * FROM [Customers] WHERE ID=@id", SqlConnection)
sqlcomm.Parameters.AddWithValue("@id", CInt(Customer_IDTextBox.Text))
Dim r As SqlDataReader = sqlcomm.ExecuteReader()
r.Read()
If r.HasRows Then
Dim address = r("Street No").ToString + " " + r("Street") + ", " + _
    r("Region") + " " + vbNewLine + r("PO Code").ToString + _
    " " + r("City") + ", " + r("Country")
Dim telephone AsString
```

```

If r("Telephone No").Equals(DBNull.Value) Or
    r("Mobile No").Equals(DBNull.Value) Then
    telephone = r("Telephone No").ToString + r("Mobile No").ToString
Else
    telephone = r("Telephone No").ToString + " / " + r("Mobile No").ToString
EndIf

objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Address")
objWord.Selection.TypeText(address)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Telephone")
objWord.Selection.TypeText(telephone)
EndIf
SqlConn.Close()

objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="DressCode")
objWord.Selection.TypeText(Dress_CodeTextBox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Description")
objWord.Selection.TypeText(DescriptionTextBox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Quantity")
objWord.Selection.TypeText(QuantityTextBox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="UnitPrice")
objWord.Selection.TypeText(unitpricetextbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="LineTotal")
objWord.Selection.TypeText(subtotaltxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="SubTotal")
objWord.Selection.TypeText(subtotaltxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="VATlabel")
objWord.Selection.TypeText(Label25.Text.ToUpper)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="VatTotal")
objWord.Selection.TypeText(vattotaltxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Discount")
objWord.Selection.TypeText(discounttxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Total")
objWord.Selection.TypeText(totaltxtbox.Text)
objWord.Selection.GoTo(What:=Word.WdGoToItem.wdGoToBookmark, Name:="Notes")
objWord.Selection.TypeText(notestxtbox.Text)

'code to protect document from alteration
objWord.ActiveDocument.Protect(Password:="pass", NoReset:=False,
Type:=Word.WdProtectionType.wdAllowOnlyFormFields)
objWord.PrintPreview = True

```

A.6. Εκτύπωση ποσού ολογράφως

A.6.1. ΣυνάρτησηNumberToString()

```

' Return a word representation of the whole number value.
PublicFunction NumberToString(ByVal num_str AsString, OptionalByVal
use_us_group_names AsBoolean = True) AsString
' Get the appropriate group names.
Dim groups() AsString
If use_us_group_names Then

```

```

        groups = NewString() {"", "thousand", "million", "billion", "trillion",
"quadrillion", "quintillion", "sextillion", "septillion", "octillion",
"nonillion", "decillion", "undecillion", "duodecillion", "tredecillion",
"quattuordecillion", "quintdecillion", "sexdecillion", "septendecillion",
"octodecillion", "novemdecillion", "vigintillion"}
Else
        groups = NewString() {"", "thousand", "million", "milliard",
"billion", "1000 billion", "trillion", "1000 trillion", "quadrillion", "1000
quadrillion", "quintillion", "1000 quintillion", "sextillion", "1000
sextillion", "septillion", "1000 septillion", "octillion", "1000 octillion",
"nonillion", "1000 nonillion", "decillion", "1000 decillion"}
EndIf

' Clean the string a bit.
' Remove "€", ",", leading zeros, and
' anything after a decimal point.
Const CURRENCY AsString = ChrW(8364)
Const SEPARATOR AsString = "."
Const DECIMAL_POINT AsString = ","
    num_str = num_str.Replace(CURRENCY, "").Replace(SEPARATOR, "")
    num_str = num_str.TrimStart(NewChar() {"0"c})

Dim pos AsInteger = num_str.IndexOf(DECIMAL_POINT)
Dim decimals AsString = ""

If pos = 0 Then
Return"zero"
ElseIf pos > 0 Then
    decimals = num_str.Substring(pos + 1)
    num_str = num_str.Substring(0, pos)
EndIf

' See how many groups there will be.
Dim num_groups AsInteger = (num_str.Length + 2) \ 3

' Pad so length is a multiple of 3.
    num_str = num_str.PadLeft(num_groups * 3, " "c)

' Process the groups, largest first.
Dim result AsString = ""
Dim group_num AsInteger
For group_num = num_groups - 1 To 0 Step -1
' Get the next three digits.
Dim group_str AsString = num_str.Substring(0, 3)
    num_str = num_str.Substring(3)
Dim group_value AsInteger = CInt(group_str)

' Convert the group into words.
If group_value > 0 Then
If group_num >= groups.Length Then
    result &= GroupToWords(group_value) & _
" ?, "
Else
    result &= GroupToWords(group_value) & _

```

```

" "& groups(group_num) & ", "
EndIf
EndIf
Next group_num

' Remove the trailing ", ".
If result.EndsWith(", ") Then
    result = result.Substring(0, result.Length - 2)
EndIf

Dim res AsString = result.Trim()
IfNot decimals.Equals("") AndNot decimals.Equals("00") Then
If decimals.Length = 1 Then
    decimals = decimals + "0"
ElseIf decimals.Length > 2 Then
    decimals = Math.Ceiling(CDbl(decimals)).ToString
EndIf
    res = res + " euros and " + GroupToWords(CInt(decimals)).Trim + "
cents"
Else
    res = res + " euros"
EndIf

Return res
EndFunction

```

A.6.2. Σύνταξη GroupToWords()

```

' Convert a number between 0 and 999 into words.
PublicFunction GroupToWords(ByVal num AsInteger) AsString
Static one_to_nineteen() AsString = {"zero", "one", "two", "three", "four",
"five", "six", "seven", "eight", "nine", "ten", "eleven", "twelve",
"thirteen", "fourteen", "fifteen", "sixteen", "seventeen", "eightteen",
"nineteen"}
Static multiples_of_ten() AsString = {"twenty", "thirty", "forty", "fifty",
"sixty", "seventy", "eighty", "ninety"}

' If the number is 0, return an empty string.
If num = 0 ThenReturn""

' Handle the hundreds digit.
Dim digit AsInteger
Dim result AsString = ""
If num > 99 Then
    digit = num \ 100
    num = num Mod 100
    result = one_to_nineteen(digit) & " hundred"
EndIf

' If num = 0, we have hundreds only.
If num = 0 ThenReturn result.Trim()

' See if the rest is less than 20.
If num < 20 Then

```

```

' Look up the correct name.
    result &= " "& one_to_nineteen(num)
Else
' Handle the tens digit.
    digit = num \ 10
    num = num Mod 10
    result &= " "& multiples_of_ten(digit - 2)

' Handle the final digit.
If num > 0 Then
    result &= " "& one_to_nineteen(num)
EndIf
EndIf

Return result.Trim()
EndFunction

```

A.7. Εισηγήσεις προϊόντων

A.7.1. Αποθηκευμένη διαδικασία sp_SmartSeller_content_based_filtering

```

CREATEPROCEDURE [dbo].[sp_SmartSeller_content_based_filtering]
    (@companyID INT,
    @collection_choice INT,
    @collection_list nvarchar(1000),
    @category_choice INT,
    @category_list nvarchar(1000),
    @color_choice INT,
    @color_list nvarchar(1000))
AS
BEGIN
    DECLARE @sql ASnvarchar(4000)

    -- If the view exists, alter it; else create it
    IFNOTEXISTS(SELECT*FROM dbo.sysobjectsWHERE ID
=OBJECT_ID(N'[dbo].[SmartSeller]')ANDOBJECTPROPERTY(id,N'IsView')= 1)
    BEGIN
        SET @sql ='CREATE VIEW SmartSeller AS '
    END
    ELSE
        SET @sql ='ALTER VIEW SmartSeller AS '

    -- Select which columns to display
    SET @sql = @sql +
'SELECT G.Code, G.Description, G.Size, C1.Color, G.[Total cost], G.[Wholesale
Price], G.Image, C2.Season + '' - '' + C2.Name AS Collection, GC.Name AS Category
        FROM Garment G LEFT JOIN Color C1 ON G.Color = C1.Code
        LEFT JOIN Collections C2 ON G.Collection = C2.Code
        LEFT JOIN [Garment Categories] GC ON GC.Code = G.Category'

    -- Set the criteria according to the parameters
    IF @collection_choice = 1
        SET @sql = @sql +
' WHERE C2.Season + '' - '' + C2.Name IN (

```

```

        SELECT DISTINCT C.Season + ' ' - ' ' + C.Name AS Collection
        FROM [Wholesale Order] WO
            LEFT JOIN [Wholesale Order Details] WOD ON WO.[Order ID]
= WOD.[Order ID]
            LEFT JOIN Garment G ON WOD.[Garment ID]=G.Code
            LEFT JOIN Collections C ON G.Collection = C.Code
        WHERE WO.[Company ID]='+CAST(@companyID ASnvarchar)+' AND WO.[Order
Date]=(SELECT MAX(WO1.[Order Date])
        FROM [Wholesale Order] WO1
        WHERE WO1.[Company ID]='+CAST(@companyID ASnvarchar)+'))'
    ELSEIF @collection_choice = 2
        SET @sql = @sql + ' WHERE C2.Season + ' ' - ' ' + C2.Name IN ('+
@collection_list +')'

    IF @collection_choice = 0
    BEGIN
        IF @category_choice <> 0
            SET @sql = @sql + ' WHERE '

    END
    ELSE
    BEGIN
        IF @category_choice <> 0
            SET @sql = @sql + ' AND '

    END

    IF @category_choice = 1
        SET @sql = @sql + ' GC.Name NOT IN (
            SELECT DISTINCT GC.Name
            FROM [Wholesale Order] WO
            LEFT JOIN [Wholesale Order Details] WOD ON WO.[Order ID] =
WOD.[Order ID]
            LEFT JOIN Garment G ON WOD.[Garment ID]=G.Code
            LEFT JOIN [Garment Categories] GC ON GC.Code = G.Category
            WHERE WO.[Company ID]='+CAST(@companyID ASnvarchar)+' AND
WO.[Order Date]=(SELECT MAX(WO1.[Order Date])
            FROM [Wholesale Order] WO1
            WHERE WO1.[Company ID]='+CAST(@companyID ASnvarchar)+'))'
    ELSEIF @category_choice = 2
        SET @sql = @sql + ' GC.Name IN ('+ @category_list +')'

    IF @collection_choice = 0
    BEGIN
        IF @category_choice = 0
        BEGIN
            IF @color_choice <> 0
                SET @sql = @sql + ' WHERE '

        END
        ELSE
            IF @color_choice <> 0
                SET @sql = @sql + ' AND '

        END
    ELSE
    BEGIN

```

```

        IF @color_choice <> 0
            SET @sql = @sql + ' AND'
    END

    IF @color_choice = 1
        SET @sql = @sql + ' C1.Color IN (
            SELECT DISTINCT C1.Color
            FROM [Wholesale Order] WO
            LEFT JOIN [Wholesale Order Details] WOD ON WO.[Order ID] =
WOD.[Order ID]
            LEFT JOIN Garment G ON WOD.[Garment ID] = G.Code
            LEFT JOIN Color C1 ON G.Color = C1.Code
            WHERE WO.[Company ID]=' + CAST(@companyID AS nvarchar) + ' AND
WO.[Order Date]=(SELECT MAX(WO1.[Order Date])

            FROM [Wholesale Order] WO1
            WHERE WO1.[Company ID]=' + CAST(@companyID AS nvarchar) + '))'
    ELSEIF @color_choice = 2
        SET @sql = @sql + ' C1.Color IN (' + @color_list + '))'

    EXEC(@sql)
    SELECT * FROM SmartSeller
END

```

A.7.2. Αποθηκευμένη διαδικασία sp_SmartSeller_collaborative_filtering

```

CREATE PROCEDURE [dbo].[sp_SmartSeller_collaborative_filtering]
    (@garmentID nvarchar(1000))
AS
BEGIN
    DECLARE @sql AS nvarchar(4000)

    -- If the view exists, alter it; else create it
    IF NOT EXISTS (SELECT * FROM dbo.sysobjects WHERE ID
= OBJECT_ID(N'[dbo].[SmartSeller]') AND OBJECTPROPERTY(id, N'IsView') = 1)
    BEGIN
        SET @sql = 'CREATE VIEW SmartSeller AS '
    END
    ELSE
        SET @sql = 'ALTER VIEW SmartSeller AS '

    -- Select which columns to display
    SET @sql = @sql + 'SELECT G.Code, G.Description, G.Size, C1.Color,
G.[Total cost], G.[Wholesale Price], G.Image,
                                C2.Season + '' - '' + C2.Name AS
Collection, GC.Name AS Category
                                FROM Garment G LEFT JOIN Color C1 ON G.Color
= C1.Code
                                LEFT JOIN Collections C2 ON G.Collection
= C2.Code
                                LEFT JOIN [Garment Categories] GC ON
GC.Code = G.Category'

    -- Set the criteria according to the parameters
    SET @sql = @sql + ' WHERE G.Code IN (SELECT DISTINCT WOD.[Garment ID]
                                FROM [Wholesale
Order Details] WOD

```

```

WHERE WOD.[Order
ID] IN (SELECT WOD.[Order ID]

FROM [Wholesale Order Details] WOD

WHERE WOD.[Garment ID] = ''' + @garmentID + '''
AND WOD.[Garment ID] != ''' +
@garmentID + ''')'
EXEC(@sql)
SELECT*FROM SmartSeller
END

```

A.8. Εισαγωγή/εξαγωγή δεδομένων

A.8.1. Αποθηκευμένη διαδικασία sp_export2csv

```

CREATEPROCEDURE [dbo].[sp_export2csv]
    (@query varchar(7500),
    @folder varchar(100))
AS
BEGIN
    DECLARE @sql varchar(8000)
    SET @sql = 'bcp ''' + @query + ''' queryout ' + @folder + '\contacts.csv -w -
t\t -T -S'+@@servername
EXECmaster..xp_cmdshell@sql
END

```

Παράδειγμα κλήσης από Visual Basic

```

Dim sql AsString = createQuery()
Dim sqlcomm AsNew SqlCommand("EXEC sp_export2csv '" + sql + "', '" +
txtPath.Text + "'", sqlConn)

sqlcomm.Connection.Open()
Dim rowsAffected = sqlcomm.ExecuteNonQuery()
sqlcomm.Connection.Close()

```

A.8.2. Συνάρτηση createQuery()

```

PrivateFunction createQuery() AsString
Dim selcols1(0 To 7), selcols2(0 To 7), colnames(0 To 7) AsString
Dim sql AsString

For i AsInteger = 0 To 7
    selcols1(i) = String.Empty
    selcols2(i) = String.Empty
    colnames(i) = String.Empty
Next

If ChckLBoxCustomers.Enabled = TrueThen
ForEach item In ChckLBoxCustomers.CheckedItems
SelectCase ChckLBoxCustomers.GetItemText(item)
Case"Full Name"
            selcols1(0) = "[Full Name] AS Name"
            colnames(0) = "'Name'"
Case"Address"

```

```

                selcols1(1) = "CAST([Street No] AS varchar) +' '+Street+',
''+Region+' ' '+CAST([PO code] AS varchar)+' ' '+City+', ''+Country AS Address"
                colnames(1) = "'Address'"
Case"Telephone Number"
                selcols1(2) = "CAST([Telephone No] AS varchar)"
                colnames(2) = "'Telephone No'"
Case"Mobile Number"
                selcols1(3) = "CAST([Mobile No] AS varchar)"
                colnames(3) = "'Mobile No'"
Case"Email"
                selcols1(5) = "Email"
                colnames(5) = "'Email'"
Case"BirthDay"
                selcols1(6) = "CAST([BirthDate] AS varchar)"
                colnames(6) = "'BirthDate'"
EndSelect
Next
EndIf

If ChckLBoxCompanies.Enabled = TrueThen
ForEach item In ChckLBoxCompanies.CheckedItems
SelectCase ChckLBoxCompanies.GetItemText(item)
Case"Company/Boutique"
                selcols2(0) = "[Boutique Name]+' ', ''+[Company Name] AS Name"
                colnames(0) = "'Name'"
Case"Contact Person"
                selcols2(0) = "[Contact person]+' ', ''+[Boutique Name]+' ',
''+[Company Name] AS Name"
                colnames(0) = "'Name'"
Case"Address"
                selcols2(1) = "CAST([Street No] AS varchar) +' '+Street+',
''+Region+' ' '+CAST([PO code] AS varchar)+' ' '+City+', ''+Country AS Address"
                colnames(1) = "'Address'"
Case"Telephone Number"
                selcols2(2) = "CAST([Telephone No] AS varchar)"
                colnames(2) = "'Telephone No'"
Case"Mobile Number"
                selcols2(3) = "CAST([Mobile No] AS varchar)"
                colnames(3) = "'Mobile No'"
Case"Fax Number"
                selcols2(4) = "CAST([Fax] AS varchar)"
                colnames(4) = "'Fax'"
Case"Email"
                selcols2(5) = "Email"
                colnames(5) = "'Email'"
Case"Website"
                selcols2(7) = "Website"
                colnames(7) = "'Website'"
EndSelect
Next
EndIf

If ChckLBoxCustomers.Enabled=TrueAnd ChckLBoxCompanies.Enabled=TrueThen
For i AsInteger = 0 To 7

```

```

If selcols1(i).Equals(String.Empty) And selcols2(i).Equals(String.Empty) Then
ContinueFor
ElseIf (Not selcols1(i).Equals(String.Empty)) And selcols2(i).Equals(String.Empty)
Then
        selcols2(i) = "NULL"
ElseIf selcols1(i).Equals(String.Empty) And (Not selcols2(i).Equals(String.Empty))
Then
SelectCase i
Case 0
        selcols1(i) = "NULL AS Name"
Case 1
        selcols1(i) = "NULL AS Address"
CaseElse
        selcols1(i) = "NULL AS " + colnames(i).Replace("'", "")
EndSelect
EndIf
Next

Dim joint0, joint1, joint2 AsString
        joint0 = Join(colnames, ", ")
        joint1 = Join(selcols1, ", ")
        joint2 = Join(selcols2, ", ")
While joint0.Contains(", , ") Or joint1.Contains(", , ") Or joint2.Contains(", ,
")
        joint0 = Replace(joint0, ", , ", ", ")
        joint1 = Replace(joint1, ", , ", ", ")
        joint2 = Replace(joint2, ", , ", ", ")
EndWhile
If joint0(joint0.Length - 2).Equals(",c") = TrueThen
joint0 = joint0.Remove(joint0.Length - 2, 2)
EndIf
If joint1(joint1.Length - 2).Equals(",c") = TrueThen
        joint1 = joint1.Remove(joint1.Length - 2, 2)
EndIf
If joint2(joint2.Length - 2).Equals(",c") = TrueThen
        joint2 = joint2.Remove(joint2.Length - 2, 2)
EndIf
        sql = "SELECT '000000', " + joint0 + " UNION ALL "& _
"SELECT CAST(ID AS varchar), " + joint1 + " FROM Couturier.dbo.Customers "& _
"UNION "& _
"SELECT CAST(ID AS varchar), " + joint2 + " FROM Couturier.dbo.[Cooperating
Company]"
ElseIf ChckLBoxCustomers.Enabled = TrueThen
Dim joint0, joint1 AsString
        joint0 = Join(colnames, ", ")
        joint1 = Join(selcols1, ", ")

While joint0.Contains(", , ") Or joint1.Contains(", , ")
        joint0 = Replace(joint0, ", , ", ", ")
        joint1 = Replace(joint1, ", , ", ", ")
EndWhile
If joint0(joint0.Length - 2).Equals(",c") = TrueThen
        joint0 = joint0.Remove(joint0.Length - 2, 2)
EndIf

```

```

If joint1(joint1.Length - 2).Equals(", "c) = TrueThen
    joint1 = joint1.Remove(joint1.Length - 2, 2)
EndIf

    sql = "SELECT '000000', " + joint0 + " UNION ALL "& _
"SELECT CAST(ID AS varchar), " + joint1 + " FROM Couturier.dbo.Customers"
ElseIf ChckLBoxCompanies.Enabled = TrueThen
Dim joint0, joint2 AsString
    joint0 = Join(colnames, ", ")
    joint2 = Join(selcols2, ", ")
While joint0.Contains(", , ") Or joint2.Contains(", , ")
    joint0 = Replace(joint0, ", , ", ", ")
    joint2 = Replace(joint2, ", , ", ", ")
EndWhile
If joint0(joint0.Length - 2).Equals(", "c) = TrueThen
    joint0 = joint0.Remove(joint0.Length - 2, 2)
EndIf
If joint2(joint2.Length - 2).Equals(", "c) Then
    joint2 = joint2.Remove(joint2.Length - 2, 2)
EndIf

    sql = "SELECT '000000', " + joint0 + " UNION ALL "& _
"SELECT CAST(ID AS varchar), " + joint2 + " FROM Couturier.dbo.[Cooperating
Company]"
EndIf

Return sql
EndFunction

```

ΠΑΡΑΡΤΗΜΑ Β

Στο παράρτημα αυτό ακολουθεί το ερωτηματολόγιο αξιολόγησης του συστήματος και τα αποτελέσματά του.

ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ

Απαντήστε τις ερωτήσεις, σημειώνοντας ανάλογα από το 1 έως το 5.

(1 = καθόλου, 2 = λίγο, 3 = μέτρια, 4 = πολύ και 5 = πάρα πολύ)

Χαρακτηριστικά Συστήματος

	1	2	3	4	5
1. Η ταχύτητα απόκρισης και εκτέλεσης εργασιών είναι ικανοποιητική για τον χρήστη;					
2. Σε ποιο βαθμό παρέχεται η δυνατότητα ανάνηψης από εσφαλμένους χειρισμούς χρήστη (undo);					
3. Πόσο κατανοητά είναι τα μηνύματα σφάλματος από τον χρήστη;					
4. Πόσο εύκολη είναι η εκτέλεση εργασιών από άπειρους χρήστες;					

Λειτουργία Συστήματος

	1	2	3	4	5
1. Σε ποιο βαθμό παρέχεται πρόσβαση στα δεδομένα μόνο σε εξουσιοδοτημένα άτομα;					
2. Πόσο εύκολη είναι η εισαγωγή, επεξεργασία ή διαγραφή δεδομένων;					
3. Σε ποιο βαθμό παρέχεται δυνατότητα ελέγχου της δραστηριότητας στο σύστημα;					
4. Σε ποιο βαθμό υπάρχει η δυνατότητα ακύρωσης μιας πράξης; (cancel)					

Εγχειρίδιο Χρήσης

	1	2	3	4	5
1. Πόσο σαφές και υποβοηθητικό είναι το εγχειρίδιο χρήσης;					
2. Κατά πόσο η επίλυση ενός προβλήματος με χρήση του εγχειριδίου είναι εύκολη;					
3. Πόσο εύκολη είναι η χρήση του συστήματος χωρίς τη βοήθεια του εγχειριδίου χρήσης;					
4. Πόσο βοηθητική είναι η θέση των βοηθητικών επεξηγήσεων στην οθόνη;					

Πολυμέσα

	1	2	3	4	5
1. Πόσο ικανοποιητική είναι η ποιότητα και η ευκρίνεια των εικόνων που χρησιμοποιούνται στο σύστημα;					
2. Πόσο ευανάγνωστες είναι οι συμβολοσειρές και το μέγεθος των γραμμάτων που χρησιμοποιούνται;					
3. Πόσο ξεκούραστα για το χρήστη είναι τα χρώματα που χρησιμοποιούνται;					
4. Πόσο κατανοητή είναι η γλώσσα που χρησιμοποιείται;					

Παρουσίαση Πληροφορίας

	1	2	3	4	5
1. Σε ποιο βαθμό το περιεχόμενο της σελίδας είναι καλά διατεταγμένο;					
2. Η ακολουθία των οθονών δεν προκαλεί σύγχυση στον χρήστη;					
3. Σε ποιο βαθμό η επόμενη οθόνη στην ακολουθία είναι προβλέψιμη;					
4. Πόσο εύκολη είναι η επιστροφή στην προηγούμενη οθόνη;					

Ορολογία – Πληροφορίες

	1	2	3	4	5
1. Πόσο σχετική είναι η ορολογία που χρησιμοποιείται με το αντικείμενο εφαρμογής;					
2. Πόσο συνεπείς είναι τα μηνύματα και οι άλλες πληροφορίες που εμφανίζονται στην οθόνη;					
3. Πόσο σαφές είναι το περιεχόμενο των μηνυμάτων στην οθόνη;					

Κατάλογος Προϊόντων

	1	2	3	4	5
1. Πόσο εύκολη είναι η πρόσβαση στον κατάλογο των προϊόντων;					
2. Πόσο λογική είναι η δομή και η οργάνωση του καταλόγου;					
3. Πόσο εύκολη είναι η πλοήγηση στον κατάλογο προϊόντων;					
4. Σε ποιο βαθμό οι πληροφορίες των προϊόντων ενθαρρύνουν τον επισκέπτη να επιλέξει εύκολα;					

Συναλλαγές

	1	2	3	4	5
1. Πόσο κατανοητή είναι η φόρμα παραγγελίας;					
2. Πόσο γρήγορα ολοκληρώνεται η συναλλαγή;					
3. Εμφανίζεται κάποια επιβεβαίωση παραγγελίας/αγοράς;					
4. Πόσο εύκολη είναι η πρόσθεση ή αφαίρεση προϊόντος σε παραγγελία;					

Εισηγήσεις για βελτιώσεις

Αποτελέσματα ερωτηματολογίων

Χαρακτηριστικά Συστήματος

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	4	1	5	4
Χρήστης 2	5	1	4	4
Χρήστης 3	4	1	5	4
Χρήστης 4	4	2	5	5
Χρήστης 5	4	1	5	3
Χρήστης 6	5	1	5	5
Χρήστης 7	5	2	5	4
Χρήστης 8	5	2	5	5
Χρήστης 9	4	1	5	5
Σύνολο	40	12	44	39

Λειτουργία Συστήματος

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	1	5	4	5
Χρήστης 2	1	5	5	5
Χρήστης 3	2	4	4	4
Χρήστης 4	2	5	5	5
Χρήστης 5	2	4	4	4
Χρήστης 6	2	4	4	4
Χρήστης 7	1	4	4	5
Χρήστης 8	1	4	4	4
Χρήστης 9	1	4	5	4
Σύνολο	13	39	39	40

Εγχειρίδιο Χρήσης

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	5	4	4	3
Χρήστης 2	5	3	4	3
Χρήστης 3	5	4	5	4
Χρήστης 4	5	4	4	3
Χρήστης 5	5	3	5	5
Χρήστης 6	5	5	5	4
Χρήστης 7	5	5	5	3
Χρήστης 8	5	3	4	3
Χρήστης 9	5	4	4	5
Σύνολο	45	35	40	33

Πολυμέσα

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	5	4	4	4
Χρήστης 2	5	5	5	5
Χρήστης 3	5	5	4	4
Χρήστης 4	5	4	4	4
Χρήστης 5	4	5	4	5
Χρήστης 6	5	4	5	5
Χρήστης 7	5	5	4	4
Χρήστης 8	4	5	5	5
Χρήστης 9	5	5	4	5
Σύνολο	43	42	39	41

Παρουσίαση Πληροφορίας

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	4	4	4	5
Χρήστης 2	4	3	5	5
Χρήστης 3	5	4	4	5
Χρήστης 4	5	3	5	5
Χρήστης 5	5	3	5	5
Χρήστης 6	5	4	5	5
Χρήστης 7	4	3	5	4
Χρήστης 8	5	5	5	5
Χρήστης 9	5	3	4	4
Σύνολο	42	32	42	43

Ορολογία – Πληροφορίες

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3
Χρήστης 1	4	4	5
Χρήστης 2	4	5	5
Χρήστης 3	5	5	5
Χρήστης 4	5	4	4
Χρήστης 5	4	4	4
Χρήστης 6	5	5	4
Χρήστης 7	4	5	5
Χρήστης 8	4	4	5
Χρήστης 9	5	5	5
Σύνολο	40	41	42

Κατάλογος Προϊόντων

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	4	4	4	4
Χρήστης 2	4	4	5	4
Χρήστης 3	5	4	5	4
Χρήστης 4	5	5	4	4
Χρήστης 5	5	4	5	4
Χρήστης 6	5	4	5	3
Χρήστης 7	5	4	4	4
Χρήστης 8	4	4	5	4
Χρήστης 9	3	4	5	5
Σύνολο	40	37	42	36

Συναλλαγές

	Ερώτηση 1	Ερώτηση 2	Ερώτηση 3	Ερώτηση 4
Χρήστης 1	4	4	4	5
Χρήστης 2	4	4	4	4
Χρήστης 3	5	4	5	5
Χρήστης 4	4	4	4	5
Χρήστης 5	4	4	5	5
Χρήστης 6	5	4	4	5
Χρήστης 7	4	4	4	4
Χρήστης 8	5	5	4	5
Χρήστης 9	4	4	5	4
Σύνολο	39	37	39	42