



ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ
ΣΧΟΛΗ
ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
ΤΜΗΜΑ
ΛΟΓΙΣΤΙΚΗΣ

ΕΚΠΟΝΗΣΗ ΟΜΑΔΙΚΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ ΜΕ

ΘΕΜΑ

**“ΤΕΧΝΙΚΕΣ ΑΝΑΛΥΣΗΣ ΚΑΙ ΣΧΕΔΙΑΣΜΟΥ ΣΕ
ΛΟΓΙΣΤΙΚΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ”**

Επιβλέπων Καθηγητής

Φωτεινόπουλος Μιχάλης

Σπουδαστές

Βλαστού Ελένη

Πασσάς Κωσταντίνος

Λαγγή Κυριακή

ΠΑΤΡΑ, 2017

Ευχαριστίες

Για την εργασία αυτή, η οποία εκπονήθηκε για την ολοκλήρωση του κύκλου σπουδών μας από το Τμήμα Λογιστικής και Χρηματοοικονομικής του Α.Τ.Ε.Ι. Πατρών, θέλουμε να ευχαριστήσουμε τον καθηγητή μας κύριο Μιχάλη Φωτεινόπουλο για την υπομονή και την βοήθειά του αλλά και τους γονείς μας, οι οποίοι στήριξαν τις σπουδές μας με διάφορους τρόπους, φροντίζοντας για την καλύτερη δυνατή μόρφωσή μας.

ΠΕΡΙΛΗΨΗ

Η ικανότητα της ενσωμάτωσης των δεδομένων στο λογισμικό των ηλεκτρονικών υπολογιστών καλύπτει πλέον ένα πολύ σημαντικό μέρος των αναγκών του οικονομικού κλάδου. Η αξιοπιστία της πληροφορίας, η οποία έγκειται στον ανθρώπινο παράγοντα και στην διεκπεραίωση τακτικού ελέγχου, αλλά και η ταχύτητα πρόσβασης σε αυτήν παίζουν καθοριστικό ρόλο στη διαχείριση των οικονομικών πληροφοριών και κατ'επέκταση στην αποδοτικότερη διεκπεραίωση των λειτουργιών ενός οικονομικού οργανισμού. Η τεχνολογία των υπολογιστών συνεχώς εξελίσσεται, το λογισμικό αναβαθμίζεται και προσαρμόζεται στις ανάγκες των χρηστών. Στην πτυχιακή αυτή θα επικεντωθούμε στις τεχνικές ανάλυσης και σχεδιασμού που χρησιμοποιούνται κατά τη δημιουργία λογισμικού και συγκεκριμένα θα ασχοληθούμε με την ανάλυση των τεχνικών και μεθόδων που διατίθενται για την δημιουργία των λογιστικών πληροφοριακών συστημάτων.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΥΧΑΡΙΣΤΙΕΣ.....	2
ΠΕΡΙΛΗΨΗ.....	3
ΠΕΡΙΕΧΟΜΕΝΑ.....	4
ΕΙΣΑΓΩΓΗ.....	7

ΜΕΡΟΣ Ι.....

ΤΟ ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ.....	8
---	---

ΚΕΦΑΛΑΙΟ 1ο.....

1.1.....Οικονομικοί οργανισμοί.....	9
1.2.....Η λογιστική και ο σκοπός της.....	11
1.3.....Πληροφοριακά συστήματα και επιχειρήσεις.....	13
1.4.....Λογιστικά πληροφοριακά συστήματα ή ΛΠΣ.....	15

ΚΕΦΑΛΑΙΟ 2ο.....

ΕΙΣΑΓΩΓΗ.....	19
2.1.....Διαδικασίες για την κατασκευή ενός ολοκληρωμένου ΠΣ.....	20
2.2.....Απαιτήσεις.....	21
2.3.....Ανάπτυξη και τεχνικές τεκμηρίωσης συστημάτων ή ΤΤΣ.....	24
2.3.1.....Η δομημένη και η αντικειμενοστραφής προσέγγιση.....	25
2.3.1.1....Δομημένη προσέγγιση ανάπτυξης συστήματος.....	25
2.3.1.2....Αντικειμενοστραφής προσέγγιση ανάπτυξης συστήματος.....	26
2.3.2.....Κατηγορίες τεχνικών τεκμηρίωσης για την ανάπτυξη ΠΣ.....	28
2.4.....Είδη μοντέλων κύκλου ζωής λογισμικού ή ΜΚΖ.....	32
2.4.1.....Το ΜΚΖ λογισμικού του καταρράκτη.....	32
2.4.2.....Το ΜΚΖ λογισμικού της πρωτοτυποποίησης.....	34
2.4.3.....Το ΜΚΖ λογισμικού της λειτουργικής επαύξησης.....	35
2.4.4.....Το σπειροειδές ΜΚΖ λογισμικού.....	36
2.4.5.....Το ΜΚΖ λογισμικού του πίδακα.....	39
2.4.6.....Σύγχρονα ΜΚΖ λογισμικού.....	40
2.4.7.....Κόστος αναθεώρησης κατασκευής λογισμικού.....	41

ΚΕΦΑΛΑΙΟ 3ο.....

3.1.....Μικρή ιστορική αναδρομή των τεχνικών τεκμηρίωσης ΠΣ.....	42
--	----

3.2.....	Διαγράμματα ροής.....	43
3.2.1.....	Είδη συμβόλων.....	43
3.2.2.....	Τύποι διαγραμμάτων ροής.....	45
3.3.....	Διαγράμματα ροής δεδομένων.....	47
3.3.1.....	Βασικές αρχές της κατασκευής ενός ΔΡΔ.....	50
3.3.2.....	Ροές δεδομένων.....	51
3.3.3.....	Διαδικασίες.....	52
3.3.4.....	Εξωτερικές οντότητες, πηγές και προορισμοί δεδομένων.....	54
3.3.5.....	Αποθήκες δεδομένων.....	54
3.4.....	Διαγράμματα οντοτήτων συσχετίσεων.....	56
3.5.....	Διαγράμματα δομής.....	58
3.5.1.....	Οι κύριες έννοιες των βασικών δομικών συστατικών των ΔΔ.....	59
3.6.....	Διαγράμματα HIPO.....	61
3.6.1.....	Δομικά στοιχεία των διαγραμμάτων HIPO.....	61
3.6.2.....	Πλεονεκτήματα των διαγραμμάτων HIPO.....	63
3.7.....	Διαγράμματα HOS.....	63
3.8.....	Διαγράμματα Warnier - Orr.....	64
3.9.....	Διαγράμματα ενεργειών.....	66
3.10.....	Διαγράμματα Michael Jackson.....	68
3.11.....	Δομημένη περιγραφή και ψευδοκώδικας.....	69
3.12.....	Δένδρα αποφάσεων.....	71
3.13.....	Πίνακες αποφάσεων.....	73
3.14.....	Διαγράμματα Nassi - Shneiderman.....	77
.....		
ΚΕΦΑΛΑΙΟ 4ο.....		
ΕΙΣΑΓΩΓΗ.....		
79		
4.1.....	Στόχοι των ERP συστημάτων.....	81
4.2.....	Προβλήματα από την εισαγωγή των συστημάτων ERP.....	81
4.3.....	Υλοποίηση ενός συστήματος ERP.....	82
4.4.....	Στάδια μετά την υλοποίηση.....	85
4.5.....	Παράγοντες επιτυχίας της υλοποίησης συστημάτων ERP.....	86
4.6.....	Προβλήματα στην λειτουργία του ERP, μετά την υλοποίηση.....	87
4.7.....	Προβλήματα από την υλοποίηση του ERP συστήματος.....	88
4.8.....	Τα ανταγωνιστικά πλεονεκτήματα του ERP.....	88
.....		
.....		
ΜΕΡΟΣ ΙΙ.....		
.....		
ΑΣΚΗΣΗ ΠΑΡΑΔΕΙΓΜΑ.....		
90		

Εισαγωγή

Η πτυχιακή μας εργασία πραγματεύεται τις τεχνικές ανάλυσης και σχεδιασμού που χρησιμοποιούνται στην ανάπτυξη των λογιστικών πληροφοριακών συστημάτων και χωρίζεται σε δύο μέρη, στο θεωρητικό και στο πρακτικό.

Μέρος Ι. Θεωρητικό υπόβαθρο πτυχιακής

Κεφάλαιο 1ο

Για να μιλήσουμε για το θέμα αυτό κρίνεται απαραίτητη η εξοικίωση του αναγνώστη με την γενική δομή της επιχείρησης, του περιβάλλοντος και των λειτουργιών της, του ρόλου και των αρμοδιοτήτων του λογιστή, του λόγου ύπαρξης των πληροφοριακών συστημάτων αλλά και του ρόλου που έχουν στην επιχείρηση, όπως περιγράφονται στο πρώτο κεφάλαιο.

Κεφάλαιο 2ο

Το δεύτερο κεφάλαιο αναφέρεται στον τρόπο κατασκευής λογισμικού από την σύλληψη της ιδέας έως την χρήση του. Περιγράφονται οι μέθοδοι δημιουργίας ενός πληροφοριακού συστήματος αλλά και οι τεχνικές τεκμηρίωσης που χρησιμοποιούνται σε κάθε στάδιο κατασκευής.

Κεφάλαιο 3ο

Σε αυτό το κεφάλαιο αναπτύσσεται με μεγαλύτερη λεπτομέρεια ο τρόπος χρήσης των τεχνικών τεκμηρίωσης συστημάτων.

Κεφάλαιο 4ο

Στο τελευταίο κεφάλαιο της θεωρίας παραθέτονται τα χαρακτηριστικά του έτοιμου προς χρήση πληροφοριακού συστήματος προγραμματισμού επιχειρηματικών πόρων ή αλλιώς ERP.

Μέρος ΙΙ. Πρακτικό υπόβαθρο πτυχιακής

Στο δεύτερο μέρος της πτυχιακής αποτυπώνεται μέσα από μια άσκηση παράδειγμα ο τρόπος χρήσης ορισμένων τεχνικών τεκμηρίωσης στην φάση της ανάπτυξης ενός λογιστικού πληροφοριακού συστήματος.

ΜΕΡΟΣ Ι.

ΤΟ ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

ΚΕΦΑΛΑΙΟ 1ο

1.1 Οικονομικοί οργανισμοί

Ένας οικονομικός οργανισμός ή αλλιώς μια επιχείρηση είναι ένας οργανωμένος συνδιασμός συντελεστών παραγωγής που έχει ως αντικειμενικό στόχο την ικανοποίηση των ανθρωπίνων αναγκών. Η αποτελεσματικότερη επίτευξη του στόχου επιτυγχάνεται μέσω του αποδοτικότερου τρόπου χρήσης των συντελεστών παραγωγής της. Δομικά στοιχεία της επιχείρησης είναι οι εγκαταστάσεις της, το ανθρώπινο δυναμικό της και η τεχνολογία της. Μόνο όταν τα μέρη από τα οποία αποτελείται μια επιχείρηση συνεργάζονται κ αλληλεπιδρούν μεταξύ τους με τον καλύτερο δυνατό τρόπο καθίστανται δυνατές οι επιδιώξεις αυτής.

Η επιχείρηση ως σύστημα

Αν λάβουμε υπόψην μας τα βασικά χαρακτηριστικά ενός συστήματος θα διαπιστώσουμε ότι οι επιχειρήσεις είναι συστήματα. Τα βασικά χαρακτηριστικά ενός συστήματος είναι οι εισροές του, δηλαδή οι πόροι τους οποίους προμηθεύεται από το περιβάλλον του, η διαδικασία επεξεργασίας των εισροών για τον μετασχηματισμό τους σε εκροές, το αποτέλεσμα του συστήματος που είναι οι εκροές του και η διαδικασία της ανατροφοδότησης, η οποία εξυπηρετεί στον έλεγχο του αποτελέσματος. Κάθε είδος συστήματος, λοιπόν, απαρτίζεται από μέρη διαφορετικά και αλληλεπιδρώμενα για τη διευκόλυνση της επίτευξης οριζμένου σκοπού, με αποτελεσματικότητα έτσι, ώστε η αξία του αθροίσματος των μερών του συστήματος να είναι μικρότερη από αυτήν της ολότητάς του.

Το περιβάλλον της επιχείρησης

Η αποτελεσματικότητα και η αποδοτικότητα ενός συστήματος επηρεάζεται από το περιβάλλον του. Ως περιβάλλον μιας επιχείρησης θεωρούμαι όλες εκείνες τις δυνάμεις που επιδρούν στην λειτουργία της και στην ανάπτυξή της. Το περιβάλλον μιας επιχείρησης διακρίνεται στο ειδικό ή μικρο-περιβάλλον και το γενικό ή μακρο-περιβάλλον. Στο ειδικό περιβάλλον και πιο συγκεκριμένα στο εσωτερικό της επιχείρησης εντάσσονται οι διάφορες λειτουργίες της, τα τμήματά της, οι διευθύνσεις της και φυσικά το ανθρώπινο δυναμικό της. Ενώ σε άμεση συνεργασία με την επιχείρηση αλλά έξω από αυτήν έχουμε τους προμηθευτές, τους καταναλωτές, τους διαμεσολαβητές αλλά και τις διάφορες ομάδες ενδιαφέροντος όπως πιστωτές, χρηματοδότες, μετόχους, κá. Το γενικό περιβάλλον αποτελείται από οικονομικούς, κοινωνικούς, τεχνολογικούς, πολιτικούς, πολιτισμικούς και νομικούς παράγοντες.

Η ανάγκη για την αποτελεσματικότερη κ αποδοτικότερη λειτουργία της επιχείρησης ωθεί το τμήμα της διοίκησης στην πρόβλεψη των μεταβολών του περιβάλλοντός της. Έχει διαπιστωθεί ότι είναι πιο δυνατή η πρόβλεψη των μεταβολών του ειδικού περιβάλλοντος σε σχέση με του γενικού, αφού οι παράγοντες που αποτελούν το γενικό περιβάλλον της επιχείρησης μεταβάλλονται συνεχώς προκαλώντας αβεβαιότητα, με αποτέλεσμα να στηρίζεται σε δεδομένα του παρελθόντος για πρόβλεψη των μεταβολών αυτών.

Οι λειτουργίες της επιχείρησης

Έχουμε χωρίσει τις λειτουργίες των επιχειρήσεων σε δυο κατηγορίες, στις βασικές και στις δευτερεύουσες. Στις βασικές λειτουργίες έχουμε την παραγωγική, την εμπορική, την οικονομική, την λειτουργία των ανθρωπίνων πόρων και λοιπές λειτουργίες. Ενώ στις δευτερεύουσες περιλαμβάνονται η διοικητική, η λειτουργία της πληροφόρισης και αυτή των δημοσίων σχέσεων. Για να μπορεί μια επιχείρηση να είναι λειτουργική και να έχει ομαλή ανάπτυξη πρέπει όλες οι επιχειρησιακές λειτουργίες που την αποτελούν να είναι σε αρμονία μεταξύ τους.

Η οικονομική λειτουργία

Στην εύρεση οικονομικών πόρων και στην κατάλληλη αξιοποίησή των κεφαλαίων αυτών, με σκοπό την διεκπεραίωση των δραστηριοτήτων της παραγωγικής και της εμπορικής λειτουργίας και γενικότερα της επιχείρησης, είναι το μέλημα της οικονομικής λειτουργίας. Περιέχονται σε αυτήν οι δραστηριότητες της λογιστικής παρακολούθησης της επιχείρησης, της αξιολόγησης των επενδύσεών της, όπως είναι η αγορά νέου μηχανολογικού εξοπλισμού, η επέκταση της επιχείρησης κτλ και της εύρεσης των απαιτούμενων κεφαλαίων για την πραγματοποίηση των στόχων της.

Η λειτουργία των ανθρωπίνων πόρων

από τις πιο σημαντικές επιχειρηματικές λειτουργίες είναι αυτή των ανθρωπίνων πόρων γνωστή και ως λειτουργία προσωπικού, αφού ο ανθρώπινος παράγοντας παίζει καθοριστικό ρόλο στην επιτυχία της επιχείρησης, ανεξαρτήτως των τεχνολογικών εξελίξεων. Σε αυτήν τη λειτουργία περιλαμβάνονται οι δραστηριότητες της επιλογής και της πρόσληψης προσωπικού αντάξιου των απαιτήσεων της επιχείρησης, της σωστής και συνεχούς εκπαίδευσής του, της πολιτικής των αμοιβών, των ωρών και των συνθηκών εργασίας, των προαγωγών, των μεταθέσεων, των αποστρατεύσεων και πολλών άλλων.

Η διοικητική λειτουργία

Η διοικητική λειτουργία αποσκοπεί στη διεκπεραίωση διαδικασιών μεγάλης βαρύτητας, όπως αυτήν της επιλογής των στόχων, του καταμερισμού των ανθρωπίνων, των τεχνολογικών και των φυσικών πόρων και του συντονισμού δράσης για την κατά το μέγιστο απόδοση των κατανεμημένων πόρων. Η κατανομή των ευθυνών μερίζεται σε τρία επίπεδα, στο εκτελεστικό, το διαχειριστικό και σε αυτό της ανώτατης διοίκησης. Στο εκτελεστικό επίπεδο περιλαμβάνονται οι καθημερινές λειτουργίες της επιχείρησης. Το διαχειριστικό επίπεδο περιλαμβάνει την οργάνωση και την υλοποίηση των στρατηγικών αποφάσεων που έχουν ληφθεί από την ανώτατη διοίκηση, η οποία ασχολείται με τον στρατηγικό προγραμματισμό, την μακροπρόθεσμη πολιτική της επιχείρησης κλπ.

Η λειτουργία της πληροφόρησης

Η απόκτηση, η επεξεργασία, ο έλεγχος, η αποθήκευση, η διανομή και η υλοποίηση των πληροφοριών παίζει καθοριστικό ρόλο στο έργο της διοίκησης όσον αφορά στη λήψη αποφάσεων, καθώς επίσης στην συνεργασία και στον συντονισμό για την υλοποίηση των ληφθέντων αποφάσεων. Η ανάγκη αυτή λοιπόν για την εύρεση των δεδομένων, τη δημιουργία και την αξιοποίηση των πληροφοριών έχει δημιουργήσει το παραπάνω σύνολο ενεργειών το οποίο απαρτίζει την λειτουργία της πληροφόρησης.

1.2 Η λογιστική και ο σκοπός της

Η λογιστική έχει ως σκοπό, μέσω της διεξαγωγής και διάθεσης ποσοτικοποιημένων πληροφοριών ζωτικής σημασίας, να υποβοηθήσει τους άμεσα και τους έμμεσα ενδιαφερόμενους να καταλήξουν σε συμπεράσματα μέσω των οποίων θα προβούν στη λήψη ορθών για την επιχείρηση αποφάσεων.

Οι αρμοδιότητες του λογιστή

Την ευθύνη της καταγραφής, της επεξεργασίας και της συνοπτικής παρουσίασης των εμπορικών συναλλαγών των οικονομικών μονάδων φέρει ο λογιστής. Επειδή οι τομείς δραστηριότητας ενός λογιστή είναι αρκετοί, αναλογικά διακρίνεται πληθώρα λογιστικών εργασιών, την ανάγκη για την απλοποίηση των οποίων έρχεται να καλύψει το συνεχώς αναβαθμιζόμενο λογισμικό της τεχνολογίας των υπολογιστών, διευκολύνοντας έτσι τον τρόπο εκτέλεσης των καθημερινών τους καθηκόντων, με την διεκπεραίωσή τους να πραγματοποιείται σε ικανοποιητικά μικρότερα χρονικά διαστήματα.

Σε κάθε κατηγορία λογιστικής δραστηριότητας εκτελούνται και διαφορετικές λογιστικές εργασίες. Συνεπώς, παρακάτω αναφέρονται οι τομείς δραστηριότητας ενός λογιστή και οι αρμοδιότητες που εμπεριέχονται σε κάθε έναν από αυτούς:

- Οικονομική Λογιστική: Στον τομέα δραστηριότητας της οικονομικής λογιστικής, ο λογιστής καλείται να συντάξει οικονομικές αναφορές προς διάθεση των εσωτερικών και εξωτερικών χρηστών της επιχείρησης. Παραδείγματα τέτοιων αναφορών αποτελούν οι καταστάσεις των συγκεντρωτικών υπολοίπων, οι καταστάσεις ταμειακών ροών κ.α.

- Φοροτεχνική Λογιστική: Σε αυτόν τον τομέα, ένας λογιστής ευθύνεται για την προετοιμασία και την υποβολή των φορολογικών δηλώσεων, κάθε είδους, της επιχείρησης.

- Λογιστική Κόστους: Οι αρμοδιότητες ενός λογιστή που δραστηριοποιείται σε αυτόν τον τομέα είναι ο προσδιορισμός του κόστους παραγωγής του προϊόντος ή της υπηρεσίας, μέσω της συγκέντρωσης, της κατάταξης, της καταγραφής και του επιμερισμού των δαπανών, βοηθώντας έτσι τα στελέχη της επιχείρησης να προβούν στην σωστή οικονομική και τιμολογιακή πολιτική καθώς και στη λήψη ορθών αποφάσεων.

- Διοικητική Λογιστική: Οι λογιστές αυτοί κατευθύνουν και ελέγχουν τις δραστηριότητες και τις επιδόσεις των υπόλοιπων λογιστών της επιχείρησης αντίστοιχα. Για τη διεκπεραίωση αυτών των αρμοδιοτήτων είναι απαραίτητη η χρήση συγκεκριμένων λογιστικών πληροφοριακών συστημάτων, τα οποία, μέσω των πληροφοριών που προσφέρουν, βοηθούν στον σχεδιασμό των λογιστικών δραστηριοτήτων της επιχείρησης.

- Ελεγκτική Λογιστική: Στην λογιστική του ελέγχου διακρίνονται δύο τομείς δραστηριοποίησης, ο εσωτερικός και ο εξωτερικός. Σκοπός της διεξαγωγής του εσωτερικού ελέγχου και ειδικότερα του λογιστικού-διαχειριστικού ελέγχου που διενεργεί ο αρμόδιος λογιστής, είναι η επιβεβαίωση της αξιοπιστίας των χρηματοοικονομικών πληροφοριών προς απεικόνιση της πραγματικής εικόνας της κατάστασης της επιχείρησης. Αυτό πραγματοποιείται με τον έλεγχο για αποτροπή εσφαλμένων λογιστικών καταχωρήσεων στα λογιστικά πληροφοριακά συστήματα της επιχείρησης. Σκοπός του εξωτερικού ελέγχου είναι η πιστοποίηση της ορθότητας των χρηματοοικονομικών καταστάσεων της επιχείρησης. Την διενέργεια του εξωτερικού ελέγχου πραγματοποιούν ειδικά εκπαιδευμένοι λογιστές-ελεγκτές.

- Τομέας δραστηριότητας λογιστικής που σχετίζεται με την ανάπτυξη συστημάτων: Οι λογιστές αυτού του τομέα δραστηριότητας συνεργάζονται με τους προγραμματιστές. Ειδικά όταν πρόκειται για τη δημιουργία ενός συστήματος, επικοινωνούν στον ειδικό τις απαιτήσεις του οργανισμού από το πρόγραμμα

ακολουθώντας φυσικά τις κατευθυντήριες γραμμές που έχουν τεθεί από τη διοίκηση. Έχουν τις απαιτούμενες γνώσεις για να διατυπώσουν διαγραμματικά τις ανάγκες της επιχείρησης, γεγονός που διευκολύνει στην κατανόηση μεταξύ τεχνικών και λογιστών αφού υπάρχει κοινό σημείο επαφής. Εκτός όμως της ανάλυσης, του σχεδιασμού και της υλοποίησης, στο εύρος των αρμοδιοτήτων των λογιστών εμπεριέχεται και ο έλεγχος που μπορούν να διεξάγουν στις διαγραμματικές απεικονίσεις που έχουν παραχθεί αλλά και στο προς εφαρμογή πρόγραμμα.

1.3 Πληροφοριακά συστήματα και επιχειρήσεις

Στην εποχή που διανύουμε είναι δύσκολο να φανταστούμε έναν οικονομικό οργανισμό, ο οποίος να βασίζεται μόνο στο ανθρώπινο δυναμικό του για την διεξαγωγή των επιχειρηματικών του δραστηριοτήτων. Το ποσοστό της επιβίωσης ενός οργανισμού που δεν χρησιμοποιεί κανένα από τα τεχνολογικά εργαλεία που υπάρχουν, είναι από ελάχιστο έως μηδαμινό, αφού βρίσκεται στη μειονεκτικότερη θέση έναντι του ανταγωνισμού. Πρέπει λοιπόν τα στελέχη της επιχείρησης να βρουν έναν τρόπο που θα τους επιτρέπει να συλλέγουν, να ασκούν έλεγχο και να μπορούν μέσα από έναν αρκετά μεγάλο όγκο δεδομένων, να αποσπών τις απαραίτητες πληροφορίες για να λειτουργεί και να αναπτύσσεται ομαλά η επιχείρηση. Την προαναφερθήσα ανάγκη έρχεται να καλύψει η τεχνολογία των ηλεκτρονικών υπολογιστών και συγκεκριμένα το λογισμικό, που όχι μόνο προσαρμόζεται στην ανάγκες της κάθε επιχείρησης, αλλά αναβαθμίζεται συνεχώς. Με αυτόν τον τρόπο, ο χρόνος που απαιτείται για την απόκτηση αυτών των πληροφοριών συμπίπτει, προκειμένου τα στελέχη των επιχειρήσεων να προβούν στις κατάλληλες αποφάσεις. Για αυτό το λόγο πολλές επιχειρήσεις επενδύουν αρκετά χρήματα στην τεχνολογία των υπολογιστών και στο λογισμικό των πληροφοριακών συστημάτων.

Ορισμός και χαρακτηριστικά πληροφοριακού συστήματος

Η αλληλεπίδραση μεταξύ του ανθρώπινου με τον τεχνολογικό παράγοντα, και μεταξύ διαδικασιών και λογισμικού, που σκοπεύει στην παροχή χρήσιμων πληροφοριών για την διεκπεραίωση των διάφορων εργασιών της επιχείρησης στα διάφορα επίπεδα οργάνωσής της, και πραγματοποιείται με την επεξεργασία των δεδομένων, λογίζεται ως ένα ολοκληρωμένο σύστημα, το οποίο ορίζεται ως πληροφοριακό. Συνεπώς οι γενικές λειτουργίες ενός πληροφοριακού συστήματος είναι η αναγνώριση και η κάλυψη των πληροφοριακών αναγκών των χρηστών, η συναφής επιλογή δεδομένων, ο μετασχηματισμός των δεδομένων σε πληροφορίες μέσω της χρήσης των κατάλληλων εργαλείων, και τέλος η μεταφορά των πληροφοριών στους χρήστες.

Όπως κάθε σύστημα έτσι και ένα πληροφοριακό σύστημα θα πρέπει να φέρει συγκεκριμένα χαρακτηριστικά. Διακρίνεται, λοιπόν, η είσοδος που αποτελείται από ανεπεξέργαστα δεδομένα προερχόμενα από το εξωτερικό ή το εσωτερικό περιβάλλον του οργανισμού. Η ικανότητα αποθήκευσης των πρωτογενών στοιχείων για τυχόν μεταγενέστερη επεξεργασία τους. Η επεξεργασία για μετατροπή των αποθηκευμένων ή μη δεδομένων, σε μορφή κατανοητή για τους χρήστες. Η έξοδο κατά την οποία πραγματοποιείται μεταξύ των διάφορων τμημάτων του οργανισμού η διάχυση και η διανομή των επεξεργασμένων πληροφοριών στους χρήστες. Και τέλος η ανατροφοδότηση, η οποία ενεργοποιείται σε τυχόν ενέργειες αξιολόγησης ή διόρθωσης της φάσης της εισόδου.

Δομικά συστατικά πληροφοριακού συστήματος

Ένα πληροφοριακό σύστημα αποτελείται από τα παρακάτω συστατικά μέρη:

- Πρώτο συστατικό μέρος αποτελεί ο τεχνολογικός εξοπλισμός, δηλαδή ο πύργος, η οθόνη του υπολογιστή, το πληκτρολόγιο, το ποντίκι, και λοιπά αντικείμενα τα οποία είναι απαραίτητα για τη λειτουργία του και κατ' επέκταση για την δημιουργία πληροφοριών μέσω της επεξεργασίας των δεδομένων.
- Το επόμενο συστατικό μέρος είναι το λογισμικό, δηλαδή το πρόγραμμα με τις οδηγίες λειτουργίας ένας υπολογιστή, που στοχεύει πιο άμεσα στη μετατροπή των δεδομένων σε πληροφορίες.
- Έπειτα έχουμε τα δεδομένα που συνήθως εκφράζονται ως λέξεις, εικόνες, ήχοι, αριθμοί και ως άλλα πρωτογενή στοιχεία, τα οποία αντιπροσωπεύουν γεγονότα που συμβαίνουν σε οργανισμούς ή στο φυσικό περιβάλλον, πριν όμως αυτά οργανωθούν σε μορφή κατανοητή και χρήσιμη στους ανθρώπους.
- Συνεχίζουμε με τις διαδικασίες, οι οποίες αποτελούν αποτυπωμένους σε εγχειρίδια έντυπης ή ηλεκτρονικής μορφής κανόνες και απευθύνονται από τους κατασκευαστές και τους ειδικούς των ηλεκτρονικών υπολογιστών, στους χρήστες του λογισμικού.
- Τέλος το σημαντικότερο των συστατικών μερών ενός πληροφοριακού συστήματος αποτελεί ο ανθρώπινος παράγοντας, αφού ο άνθρωπος μπορεί να το δημιουργήσει, να το χρησιμοποιήσει και να το συντηρήσει.

Βασικοί τύποι πληροφοριακών συστημάτων

Τα πληροφοριακά συστήματα ταξινομούνται στις παρακάτω τέσσερις βασικές κατηγορίες: στα συστήματα επεξεργασίας συναλλαγών, στα συστήματα διαχείρισης πληροφοριών, στα συστήματα υποστήριξης αποφάσεων και στα συστήματα γνώσης.

Κατά τις πρώτες μέρες των υπολογιστών, οι οικονομικοί αλλά και κάθε είδους οργανισμοί, σε συνεργασία φυσικά με τον κατασκευαστή λογισμικού έδιναν

μεμονωμένες λύσεις σε συγκεκριμένα προβλήματα μέσω της δημιουργίας και της χρήσης των πληροφοριακών συστημάτων. Έν καιρό, παρατηρήθηκε πως η συνδιαστική χρήση των συστημάτων της επιχείρησης και του πληροφοριακού συστήματος, έχει τη δυνατότητα να λύσει αρκετά από τα προβλήματά της, λόγω των κοινών τους χαρακτηριστικών, έτσι τα στελέχη προσανατολίστηκαν στην κατασκευή ενός ενιαίου πληροφοριακού συστήματος, για την καλύτερη εξυπηρέτησή τους. Στην πορεία όμως διαπιστώθηκε πως για να επιτευχθεί αυτό απαιτείται ο καθορισμός του τρόπου χρήσης, του λόγου αναγκαιότητας της χρήσης αλλά και το πού θα χρησιμοποιούσαν το συγκεκριμένο πληροφοριακό σύστημα.

Αναγκαιότητα χρήσης πληροφοριακών συστημάτων

Η ύπαρξη των πληροφοριακών συστημάτων κρίνεται αναγκαία και στο εσωτερικό περιβάλλον της επιχείρησης για τη διαχείριση των εσωτερικών παραγωγικών της λειτουργιών, αλλά και στο εξωτερικό περιβάλλον αυτής αφού βελτιώνεται η ανταγωνιστικότητά της και εξασφαλίζεται έτσι η βιωσιμότητά της. Στο εσωτερικό κάθε επιχείρησης διακρίνεται ένα ιεραρχικό μοντέλο κατανομής του έργου και των ευθυνών αυτής, βάση του οποίου κατηγοριοποιούνται τα πληροφοριακά συστήματα. Συνεπώς, αφού η επιλογή του τύπου του πληροφοριακού συστήματος προς χρήση έχει να κάνει με την κατανομή των στελεχών της στα διάφορα τμήματα της επιχείρησης, καταλαβαίνουμε πως θα μπορούσαν να συνυπάρχουν ενεργά διαφορετικοί τύποι πληροφοριακών συστημάτων.

Οι πιο σημαντικοί λόγοι για την αναγκαιότητα χρήσης των πληροφοριακών συστημάτων από μια επιχείρηση παραθέτονται συνοπτικά παρακάτω:

- Η συγκέντρωση των δεδομένων, η αποθήκευσή τους και η μετατροπή τους σε πληροφορίες.
- Η δυνατότητα πληροφόρησης των στελεχών της επιχείρησης χρήσιμης για τη διεξαγωγή συμπερασμάτων που αφορά τον στρατηγικό προγραμματισμό.
- Με τη σύνδεση μεταξύ του πληροφοριακού συστήματος της επιχείρησης και των εξωτερικών πληροφοριακών συστημάτων, όπως είναι αυτά των προμηθευτών και των πελατών, αξιοποιούνται κατά το μέγιστο οι πληροφορίες που προέρχονται από το εξωτερικό περιβάλλον της επιχείρησης.
- Η ικανοποίηση των νέων αναγκών μέσω της συνεχούς δημιουργίας τρόπων.
- Η εκπαίδευση και η εκμάθηση κατ' αντιστοιχία.
- Τέλος, η βοήθεια που παρέχεται στον έλεγχο και στη διοίκηση.

1.4 Λογιστικά πληροφοριακά συστήματα ή ΛΠΣ

Σε μια οργανωμένη οικονομική μονάδα είναι αναγκαία η αποτελεσματική καταγραφή και αξιολόγηση των οικονομικών της συναλλαγών. Την ανάγκη αυτή έρχεται να καλύψει το λογιστικό πληροφοριακό σύστημα ή εν χάρη συντομίας ΛΠΣ, το οποίο υποστηρίζει τη διαδικασία της καταχώρησης πρωτογενών στοιχείων και της επεξεργασίας των δεδομένων αυτών (των συναλλαγών και γενικότερα των οικονομικών πράξεων ενός οργανισμού), προς αποτέλεσμα των κατάλληλων οικονομικών πληροφοριών στους αποδέκτες του.

Ένα ΛΠΣ, εκτός της συλλογής και της επεξεργασίας των δεδομένων με σκοπό τη διεκπεραίωση των καθημερινών οικονομικών πράξεων του οργανισμού, συμβάλλει αποτελεσματικά και στη διαδικασία της λήψης ορθών αποφάσεων, έτσι η ποιότητα των εκροών του το καθιστά απαραίτητο για την αποτελεσματική λειτουργία των οικονομικών μονάδων.

Οι λογιστικές πληροφορίες, παρόλο που δεν καλύπτουν πλήρως την ανάγκη των χρηστών, θεωρούνται από τις πιο σημαντικές πληροφορίες που μπορεί να παράγει και να διανέμει η επιχείρηση στους ενδιαφερόμενούς της. Οι χρήσιμες λογιστικές πληροφορίες, όπως η διαθεσιμότητα των κεφαλαίων της επιχείρησης, το ύψος των διανεμηθέντων κερδών, η αποπληρωμή τυχόν δανείων, η μισθολογική κατάσταση του προσωπικού και γενικότερα η οικονομική κατάσταση και η απόδοσή της, παρέχονται συνήθως από τις οικονομικές καταστάσεις και τις διάφορες εκθέσεις της επιχείρησης.

Τι ονομάζουμε ΛΠΣ και ποιές οι λειτουργίες του

Η κατανόηση του όρου λογιστικό πληροφοριακό σύστημα ή ΛΠΣ, προϋποθέτει την κατανόηση των τριών λέξεων που το αποτελούν. Η λογιστική παρέχει ποσοτικοποιημένες πληροφορίες στα στελέχη ενός οργανισμού αλλά και στους εξωτερικούς ενδιαφερόμενους. Το αποτέλεσμα που προκύπτει από την επεξεργασία πρωτογενών στοιχείων, ονομάζεται πληροφορία. Όταν χρησιμοποιείται η λέξη σύστημα τότε αναφερόμαστε σε μια ολοκληρωμένη οντότητα. Συνεπώς, μπορούμε να πούμε βάση των παραπάνω πως ένα λογιστικό πληροφοριακό σύστημα αποτελεί ένα ολοκληρωμένο πλαίσιο, στο οποίο εισέρχονται δεδομένα από το εσωτερικό και το εξωτερικό περιβάλλον του οργανισμού, τα οποία μετατρέπονται σε οικονομικές πληροφορίες, με σκοπό την παροχή τους στους ενδιαφερόμενους, οι οποίοι χρησιμοποιούν τις πληροφορίες αυτές για τη διενέργεια των δραστηριοτήτων της επιχείρησης, για τη διεξαγωγή διαπιστώσεων, για τη διαδικασία λήψης ορθών αποφάσεων, και για άλλους λόγους.

Το λογιστικό πληροφοριακό σύστημα αποτελεί μέρος του ευρύτερου πληροφοριακού συστήματος που υποστηρίζει έναν οργανισμό, για τον λόγο αυτό θα μπορούσαμε να το χαρακτηρίσουμε και ως υποσύστημα του πληροφοριακού

συστήματος. Το ΛΠΣ είναι ένα αυτοτελές ιδιαίτερο υποσύστημα το οποίο έχει τους δικούς του στόχους, δραστηριότητες και πόρους. Αυτό που έχει τη δυνατότητα να δεχθεί ως εισροή είναι οικονομικά γεγονότα, όπως πώλησεις και αγορές αγαθών και υπηρεσιών, με μετρητά ή με πίστωση και πραγματοποίηση δαπανών. Οι διαδικασίες που λαμβάνουν χώρα σε ένα ΛΠΣ αφορούν στην καταχώρηση των οικονομικών συναλλαγών στο ημερολόγιο, στην μεταφορά των χρηματικών ποσών στο γενικό και στα αναλυτικά καθολικά και τέλος στον συμψηφισμό αυτών προκειμένου να εξαχθούν οι οικονομικές καταστάσεις. Εκροές του συστήματος αποτελούν οι διάφορες οικονομικές αναφορές που εκτυπώνονται σε τακτά χρονικά διαστήματα. Τέλος, ο έλεγχος διασφαλίζει την αποτροπή σφαλμάτων, και όπου αυτά εμφανίζονται την αντιμετώπισή τους.

Όπως κάθε πληροφοριακό σύστημα έτσι και ένα λογιστικό πληροφοριακό σύστημα λογίζεται ως ένα σύστημα στόχου. Η δημιουργία ενός ΛΠΣ προϋποθέτει την παρακολούθηση των οικονομικών δραστηριοτήτων του οργανισμού για τον οποίο φτιάχτηκε, η οποία πραγματοποιείται με την έγκαιρη διάθεση έγκυρης πληροφόρησης προς τα ενδιαφερόμενα μέλη, αφού φυσικά πρώτα έχει διεξαχθεί επεξεργασία στα δεδομένα όλων των οικονομικών συναλλαγών του οργανισμού. Οι πληροφορίες αυτές έχουν ως στόχο να επηρεάσουν τη διαδικασία λήψης αποφάσεων του αποδέκτη αυτών.

Μορφές ΛΠΣ

Πολλοί διαφορετικοί τύποι από λογιστικά πληροφοριακά συστήματα έκαναν την εμφάνισή τους στον επιχειρηματικό κόσμο τα τελευταία πενήντα χρόνια, αφού τα ήδη υπάρχοντα ΛΠΣ χρησιμοποιούνταν κάθε φορά ως παραδείγματα προς βελτίωση. Κάνοντας λοιπόν μια αναδρομή στις περισσότερες χρονικές περιόδους βλέπουμε πως ήταν εν ενεργεία διαφορετικές γενιές συστημάτων από επιχείρηση σε επιχείρηση ταυτόχρονα, ενώ σε ορισμένες επιχειρήσεις διαφορετικά είδη ΛΠΣ συνυπήρχαν παράλληλα.

Παρακάτω παραθέτονται μερικά από τα βασικά είδη ΛΠΣ:

- Τα Manual συστήματα χρησιμοποιούνταν από μικρούς οργανισμούς και αποτελούνται από τα έγγραφα, το γενικό καθολικό, το ημερολόγιο, τα ειδικά ημερολόγια και τα ισοζύγια. Παρόλο που η χρήση τους στις μέρες μας σπανίζει ενέχουν χρησιμότητα στην εκμάθηση χρηστών της διαδικασίας που ακολουθείται με την πραγματοποίηση μιας χρηματοοικονομικής συναλλαγής
- Τα συστήματα Flat-File αλλιώς γνωστά ως συστήματα legacy χρησιμοποιούνται από από μια μειωψηφία μικρών οργανισμών έως και σήμερα. Εδώ ο κάθε χρήστης έχει

δικό του χώρο δεδομένων μέσα στον οποίο καλείται να αποθηκεύσει τα δεδομένα και να τα επεξεργαστεί. Η μη ύπαρξη μιας κοινής “πισίνας” δεδομένων από όπου μπορούν όλοι οι χρήστες να έχουν πρόσβαση σε αυτά μέσω της ανατροφοδότησης, οδήγησε εν τέλει σε αρκετά πρακτικά προβλήματα ως προς τη συλλογή τους, την ενημέρωσή τους και την εγκυρότητα των πληροφοριών μεταξύ άλλων.

- Τα συστήματα βάσεων δεδομένων, τα οποία δημιουργήθηκαν για να καλύψουν τις ελλείψεις των συστημάτων Flat-File. Έτσι εδώ έχουμε έναν ελεγχόμενο κοινόχρηστο χώρο στον οποίο τα δεδομένα εισάγονται μια φορά, ενώ όλοι οι εξουσιοδοτημένοι χρήστες μπορούν να έχουν πρόσβαση σε αυτά. Η προσβασιμότητα ελέγχεται από το σύστημα διαχείρισης βάσεων δεδομένων ή αλλιώς ΣΔΒΔ, ένα ειδικό λογισμικό το οποίο δίνει πρόσβαση στα αιτούμενα από εξουσιοδοτημένους χρήστες δεδομένα ενώ απορρίπτει τις αιτήσεις από μη εξουσιοδοτημένους χρήστες.

- Το μοντέλο Resources Events Agents ή REA το οποίο αφού αναγνωρίζει ως συνθετικά μέρη ενός οικονομικού οργανισμού τα γεγονότα της καθημερινότητας, τους πόρους και τους εμπλεκόμενους παράγοντες, τα προδιορίζει για να καταλήξει στον καθορισμό των μεταξύ τους σχέσεων.

- Τα συστήματα προγραμματισμού επιχειρηματικών πόρων Enterprise Resource Planning ή ευρέως γνωστά ως ERP συστήματα, είναι τα πλέον αποτελεσματικά συστήματα λογισμικού αφού μέσω της χρήσης σχεσιακών βάσεων δεδομένων γίνεται εφικτή η προσβασιμότητα και η επικοινωνία πολλών τμημάτων του οργανισμού, μέσω των διαφόρων πληροφοριακών συστημάτων που εμπεριέχονται σε αυτό, σε πραγματικό χρόνο. Ένσωματώνοντας τις βασικές επιχειρηματικές και διοικητικές διαδικασίες δίνει τη δυνατότητα στη διοίκηση να γνωρίζει ανά πάσα στιγμή την κατάσταση του οργανισμού και να προβεί στην λήψη ορθών αποφάσεων, στον καταλληλότερο στρατηγικό σχεδιασμό και όχι μόνο. Ένα επιπλέον πλεονέκτημα είναι η δυνατότητα επιλογής των υποσυστημάτων που θα εμπεριέχονται στο πακέτο λογισμικού ERP ανάλογα με τις επιχειρησιακές ανάγκες του κάθε οργανισμού.

ΚΕΦΑΛΑΙΟ 2ο

Εισαγωγή

Μεθοδολογία ανάπτυξης πληροφοριακών συστημάτων ή ΠΣ

Ως μεθοδολογία νοείται η φυσική υλοποίηση του λογικού κύκλου ζωής που ενσωματώνει τα στάδια των δραστηριοτήτων της επιχείρησης σε κάθε φάση και υποφάση του συστήματος, τους ατομικούς και ομαδικούς ρόλους που υπάρχουν σε κάθε δραστηριότητα, τα αποτελέσματα και τα πρότυπα ποιότητας για κάθε δραστηριότητα και τέλος τις τεχνικές και τα εργαλεία που θα χρησιμοποιηθούν σε κάθε δραστηριότητα. Η ύπαρξη μεθοδολογίας εξασφαλίζει την συνοχή σε προσεγγίσεις που εφαρμόζονται σε πολλά έργα, περιορίζει τον κίνδυνο των λαθών, ενώ παρέχει παράλληλα τη δυνατότητα της ολοκληρωμένης και με συνοχή τεκμηρίωσης από το ένα έργο στο άλλο.

Εξικοίωση με την έννοια του μοντέλου κύκλου ζωής λογισμικού

Κάθε εφαρμογή λογισμικού, από τη σύλληψη μέχρι την απόσυρσή της, διέρχεται από διάφορες φάσεις, σε καθεμιά εκ των οποίων πρέπει να γίνονται ορισμένες εργασίες ώστε να επιτυγχάνεται το επιθυμητό αποτέλεσμα. Οι πολύ γενικές φάσεις είναι η σύλληψη, η κατασκευή, η χρήση με την αντίστοιχη συντήρηση και η απόσυρση.

Ένα μοντέλο κύκλου ζωής λογισμικού είναι η περιγραφή των δραστηριοτήτων και των επιμέρους φάσεων από τις οποίες αυτό διέρχεται, από τη σύλληψη μέχρι την απόσυρσή της, καθώς και των εργασιών που λαμβάνουν χώρα σε καθεμιά από τις φάσεις αυτές. Σε κάθε διαδικασία ανάπτυξης μπορούμε να διακρίνουμε περισσότερες από μία επιμέρους φάσεις, ενώ σε κάθε επιμέρους φάση μπορούμε να διακρίνουμε περισσότερες από μία εργασίες.

Ένα μοντέλο κύκλου ζωής λογισμικού στοχεύει στην καθοδήγηση του κατασκευαστή προκειμένου αυτός να επιτύχει την πιο παραγωγική υλοποίηση των διαδικασιών ανάπτυξης λογισμικού, με τα λιγότερα δυνατά σφάλματα και το μικρότερο δυνατό ρίσκο στις εκάστοτε συνθήκες. Αυτό διαφοροποιεί ανάλογα με το μέγεθος και το θεματικό πεδίο κάθε εφαρμογής λογισμικού, με την εμπειρία και τα ιδιαίτερα χαρακτηριστικά του κάθε κατασκευαστή και με το εκάστοτε περιβάλλον ανάπτυξής του.

Αυτό που διαφοροποιεί τα διάφορα μοντέλα κύκλου ζωής λογισμικού είναι η εμπέλεια, δηλαδή η έκταση του υπό κατασκευή συστήματος λογισμικού στην οποία αυτές οι διαδικασίες εφαρμόζονται, η επαναληπτικότητα των εργασιών, καθώς και οι

ενδιάμεσες αποτιμήσεις από τον πελάτη ή τον κατασκευαστή. Έτσι, σε κάθε μοντέλο κύκλου ζωής είναι με διαφορετικό τρόπο δυνατός ο εντοπισμός της ανάγκης και η ενσωμάτωση τροποποιήσεων στα χαρακτηριστικά του λογισμικού προτού να ολοκληρωθεί πλήρως η κατασκευή του.

2.1 Διαδικασίες για την κατασκευή ενός ολοκληρωμένου ΠΣ

Οι διαδικασίες ανάπτυξης λογισμικού μπορούν να ταξινομηθούν ως εξής :

1. Αρχικά διακρίνεται η διαδικασία του ορισμού των παραμέτρων βάση των απαιτήσεων του οργανισμού ή αλλιώς οι προδιαγραφές της ανάπτυξης του λογισμικού, δηλαδή ο καθορισμός των εργασιών που θα επιτελεί το λογισμικό, καθώς και των περιορισμών και των παραδοχών που ισχύουν.
2. Έπειτα έχουμε την ανάπτυξη του, δηλαδή την κατασκευή του λογισμικού. Εδώ, σε όλα τα μοντέλα κύκλου ζωής μπορούμε να διακρίνουμε τρεις επιμέρους φάσεις, την ανάλυση, τη σχεδίαση και την κωδικοποίηση του λογισμικού.
3. Στη συνέχεια πραγματοποιείται η διαδικασία της επαλήθευσης, δηλαδή η επιβεβαίωση της ικανοποίησης των προδιαγραφών και της μη ύπαρξης σφαλμάτων ή η διόρθωσή τους όπου αυτά εμφανίζονται όταν αυτό είναι εφικτό.
4. Και τέλος έχουμε τη διαδικασία της εξέλιξης, δηλαδή τη συνεχή βελτίωση και αναβάθμιση των λειτουργικών χαρακτηριστικών του λογισμικού ή την τροποποίηση του ήδη υπάρχοντος, για την ικανοποίηση των μεταβαλλόμενων αναγκών.

Για την κατασκευή του μοντέλου ενός ολοκληρωμένου πληροφοριακού συστήματος, λοιπόν, απαιτείται η πραγματοποίηση των παρακάτω διεργασιών:

- Η ανάλυση και η μοντελοποίηση των επεξεργασιών που γίνονται στο σύστημα, δηλαδή η κατασκευή του μοντέλου επεξεργασιών του συστήματος.
- Η ανάλυση και η μοντελοποίηση των δεδομένων που διαχειρίζεται το σύστημα, δηλαδή η κατασκευή του μοντέλου δεδομένων του συστήματος.
- Και η δημιουργία του μοντέλου επεξεργασίας δεδομένων, η οποία περιλαμβάνει τη συγχώνευση των δύο προηγούμενων μοντέλων.

Η κατασκευή των αντίστοιχων μοντέλων και στις τρεις περιπτώσεις γίνεται κυρίως με τη χρήση διαγραμματικών τεχνικών. Οι τεχνικές αυτές βοηθούν ιδιαίτερα στην κατανόηση, την ανάλυση και την παρουσίαση των απαιτήσεων και των προβλημάτων του συστήματος αφού έχουν την δυνατότητα να περιγράφουν τις λειτουργίες του με τη χρήση εικόνων, όπως είναι τα διαγράμματα, τα δένδροδιαγράμματα, οι πίνακες κ.ά.

2.2 Απαιτήσεις

Η ανάλυση των απαιτήσεων είναι η πρώτη διαδικασία της ανάπτυξης ενός συστήματος. Αποτελεί μια πολύ κρίσιμη διεργασία καθώς εκφράζει την προσπάθεια κατανόησης των αναγκών και των απαιτήσεων της επιχείρησης. Η έλλειψη οργανωμένης και υπεύθυνης προσπάθειας στον καθορισμό των απαιτήσεων είναι σίγουρο ότι θα οδηγήσει σε προβλήματα που θα εμφανιστούν σε μεταγενέστερα στάδια του κύκλου ζωής και θα κοστίσουν σε χρήματα και σε ώρες εργασίας.

Απαίτηση είναι η περιγραφή μιας υπηρεσίας που θα πρέπει να παρέχει το υπό κατασκευή σύστημα, μιας διεργασίας που θα πραγματοποιεί ή μιας συνθήκης που θα πρέπει να ικανοποιεί. Η απαίτηση απαντάει στο ερώτημα του τι κάνει το σύστημα και δεν περιλαμβάνει τον τρόπο που θα γίνει αυτό. Αποτελεί μια προσπάθεια αποσαφήνισης του προβλήματος που στοχεύει να λύσει το πληροφοριακό σύστημα που θα δημιουργηθεί και όχι στην περιγραφή κάποιας λύσης. Η απαίτηση καταγράφεται και τεκμηριώνεται με τέτοιο τρόπο ώστε να είναι κατανοητή από όλα τα ενδιαφερόμενα μέρη στην ανάπτυξη του συστήματος. Επιπλέον η απαίτηση είναι μια σημαντική σταθερά που παραμένει ως ένας παράγοντας αξιολόγησης καθ' όλη τη διάρκεια του κύκλου ζωής του συστήματος, ενώ ελέγχεται ακόμα και μετά το τέλος της ανάπτυξής του.

Οι απαιτήσεις διακρίνονται σε δυο μεγάλες κατηγορίες, στις λειτουργικές και στις μη λειτουργικές. Οι λειτουργικές απαιτήσεις περιγράφουν αναλυτικά την αλληλεπίδραση του συστήματος και του περιβάλλοντος. Πολλές φορές οι λειτουργικές απαιτήσεις αποδίδουν την έννοια του επεξεργαστή της πληροφορίας σε αυτό που περιγράφουν. Έτσι αναφέρονται στην είσοδο ή αλλιώς στα ερεθίσματα που μπορεί να δεχτεί το σύστημα και στον τρόπο που αντιδρά σε αυτά τα ερεθίσματα. Το σύστημα μπορεί να διενεργεί κάποιες επεξεργασίες ή να μεταβάλλει την κατάστασή του. Μετά την επεξεργασία προκύπτει η έξοδος του συστήματος, δηλαδή τα επιθυμητά αποτελέσματα που παράγει σαν απόκριση στο ερέθισμα, στην είσοδο δηλαδή που δέχθηκε.

Οι απαιτήσεις που δεν αναφέρονται σε κάποια λειτουργία του συστήματος ονομάζονται μη λειτουργικές. Οι μη λειτουργικές απαιτήσεις περιγράφουν κάποιες προδιαγραφές που πρέπει να έχει το σύστημα πλαισιώνοντας ουσιαστικά τις λειτουργικές απαιτήσεις. Είναι πολύ σημαντικές καθώς θέτουν κάποιους περιορισμούς στις επιλογές που έχουν οι κατασκευαστές στα στάδια σχεδιασμού και υλοποίησης του συστήματος. Οι μη λειτουργικές απαιτήσεις αναλύονται επιμέρους σε κατηγορίες. Οι πιο σημαντικές κατηγορίες είναι η ασφάλεια, η απόδοση, η

αξιοπιστία, η χρηστικότητα και η υποστήριξη.

Η επιχείρηση είναι αυτή που εντοπίζει ένα πρόβλημα και εκτιμά ότι υπάρχει ανάγκη λύσης του. Το πρόβλημα μπορεί να αφορά τον τρόπο με τον οποίο γίνονται κάποια πράγματα καθημερινά ή ένα παλαιότερο πληροφοριακό σύστημα που θα πρέπει να επεκταθεί ή να αντικατασταθεί. Χρησιμοποιείται η φυσική γλώσσα για να περιγραφούν το πρόβλημα και οι υπηρεσίες που θα παρέχει το νέο πληροφοριακό σύστημα, καθώς και οι συνθήκες που θα πρέπει να ικανοποιεί. Ουσιαστικά η επιχείρηση αποτελεί το πρώτο από τα ενδιαφερόμενα μέλη που περιγράφει τις απαιτήσεις από το σύστημα και το λογισμικό, προτείνει κάποιες λύσεις και κάνει υποδείξεις σε θέματα που αφορούν τον σχεδιασμό του συστήματος. Αυτές βέβαια διαφέρουν πάρα πολύ από τις απαιτήσεις του συστήματος, αφού συχνά πολυάριθμες απαιτήσεις εκφράζονται σαν μία απαίτηση και συγχέονται οι λειτουργικές με τις μη λειτουργικές. Ενώ, έχει παρατηρηθεί πως αρκετές φορές οι απαιτήσεις περιγράφονται σε συντομία, δεν είναι ακριβείς και εμπεριέχουν πολλές ασάφειες.

Η διαδικασία του προσδιορισμού των απαιτήσεων αποτελεί ένα δύσκολο έργο. Η κρισιμότητα αυτής της εργασίας έχει οδηγήσει στη δημιουργία και εφαρμογή μια σειράς διακριτών βημάτων. Ο διαχωρισμός της διαδικασίας του προσδιορισμού των απαιτήσεων σε επιμέρους δραστηριότητες διευκολύνει την εκτέλεσή της. Η πραγματοποίηση κάθε βήματος προϋποθέτει την καταγραφή των πληροφοριών και την τελική παραγωγή του εγγράφου προδιαγραφής των απαιτήσεων. Μέχρι την τελική παραγωγή αυτής της έκθεσης τα βήματα του ορισμού του προβλήματος, της εξαγωγής των απαιτήσεων, της ανάλυσης και της προδιαγραφής τους, επαναλαμβάνονται πολλές φορές. Έτσι επιτυγχάνεται η αναθεώρηση των απαιτήσεων που ορίστηκαν στην αρχή της διαδικασίας.

Στην δραστηριότητα της εξαγωγής των απαιτήσεων οι αναλυτές του συστήματος επικοινωνούν με τα ενδιαφερόμενα μέλη. Η επικοινωνία αυτή έχει στόχο την ανάλυση της υφιστάμενης κατάστασης ώστε να βγάλουν οι αναλυτές τα δικά τους συμπεράσματα και την αποσαφήνιση της εκδοχής των απαιτήσεων των πελατών. Τα ενδιαφερόμενα μέλη είναι ο πελάτης και όλοι οι χρήστες του συστήματος. Συχνά ο πελάτης και οι χρήστες εκφράζουν με ένα δικό τους τρόπο το τι περιμένουν από το νέο σύστημα. Οι αναλυτές εργάζονται με τους ενδιαφερόμενους για να εκμαιεύσουν πληροφορίες ουτως ώστε να κατανοήσουν το πρόβλημα, να καταλάβουν τις ανάγκες που ωθούν στη δημιουργία του νέου συστήματος και τις προσδοκίες των ενδιαφερόμενων από αυτό. Γι' αυτό το σκοπό οργανώνουν συναντήσεις με σαφείς ξεκάθαρους στόχους και συνεντεύξεις με σαφείς και στοχευμένες ερωτήσεις.

Τυχόν λάθη που θα γίνουν σε αυτό το στάδιο θα κοστίσουν σε χρόνο και σε χρήμα, έτσι οι αναλυτές θα πρέπει να είναι ιδιαίτερα προσεκτικοί και να

διασφαλίζουν ότι τα στοιχεία που εκμαιεύουν από τους πελάτες είναι και τα σωστά. Έτσι χρησιμοποιούν τεχνικές για να ελέγξουν τις πληροφορίες που λαμβάνουν από τους ενδιαφερόμενους, όπως παραδείγματος χάριν απευθύνουν το ίδιο ερώτημα με διαφορετικό τρόπο ή διατύπωση στη συνέντευξη και ελέγχουν αν θα πάρουν την ίδια απάντηση. Ομοίως απευθύνουν το ίδιο ερώτημα σε όλες τις ενδιαφερόμενες μεριές και εξετάζουν τις αντιλήψεις που υπάρχουν για ένα συγκεκριμένο θέμα.

Βασική προϋπόθεση για τη σωστή εκτέλεση αυτής της φάσης είναι η άριστη συνεργασία και επικοινωνία ανάμεσα στον αναλυτή και τους ενδιαφερόμενους. Ωστόσο πολλές φορές η καλή επικοινωνία δεν είναι πλήρως εφικτή. Οι αναλυτές είναι αναμενόμενο να μην έχουν ιδιαίτερες γνώσεις πάνω στον τομέα που δραστηριοποιείται ο ενδιαφερόμενος, ούτε να γνωρίζουν την ορολογία που χρησιμοποιείται σε αυτό το πεδίο. Κρίνεται λοιπόν σκόπιμο να προετοιμαστούν κατάλληλα πριν αρχίσει ο κύκλος επαφών με τον πελάτη και τους χρήστες του συστήματος. Θα πρέπει να ενημερωθούν και να αποκτήσουν κάποιες ειδικές γνώσεις πάνω στον συγκεκριμένο τομέα. Έτσι θα πετύχουν βαθύτερη κατανόηση του υπάρχοντος συστήματος, αφού θα δημιουργηθεί καλύτερος κώδικας επικοινωνίας.

Συμπληρωματικά οι αναλυτές μπορούν να χρησιμοποιήσουν την παρατήρηση για να μελετήσουν την υπάρχουσα κατάσταση. Η υπάρχουσα κατάσταση μπορεί να είναι μια χειρωνακτική εργασία που πρέπει να γίνει πιο αυτοματοποιημένη ή ένα παλιό σύστημα που δεν είναι πλέον λειτουργικό ή υπάρχει ανάγκη επέκτασης. Μέσα από την παρατήρηση και την καταγραφή του τρόπου λειτουργίας υπάρχει η δυνατότητα καλύτερης κατανόησης του συστήματος και διάγνωσης ενδεχόμενων δυσλειτουργιών και δυσκολιών.

Στα αρχικά στάδια της εξαγωγής των απαιτήσεων οι αναλυτές επιχειρούν να αποκρυσταλλώσουν τον τρόπο που λειτουργεί η υπάρχουσα κατάσταση για να κατανοήσουν τις απαιτήσεις και τις ανάγκες του πελάτη όπως προαναφέραμε. Έτσι κάνουν τις εξής διεργασίες:

- Δημιουργούν μια λίστα με τους κανόνες που διέπουν την υφιστάμενη κατάσταση.
- Καταγράφουν τις κύριες έννοιες του συστήματος και των χαρακτηριστικών τους.
- Και τέλος καταγράφουν τα σενάρια που περιγράφουν με λεπτομέρειες τις λειτουργίες του συστήματος.

Κατά τη διάρκεια αυτής της φάσης επικαιροποιούν και εμπλουτίζουν διαρκώς αυτές τις πληροφορίες αξιοποιώντας την ανατροφοδότηση που λαμβάνουν στις συναντήσεις με τους ενδιαφερόμενους, τα ευρήματα των παρατηρήσεών τους και της ανάλυσης αυτών. Επίσης διαμορφώνουν μια αρχική άποψη για την δομή του συστήματος σε υψηλό επίπεδο, χωρίς όμως να προχωρούν σε ιδιαίτερες λεπτομέρειες.

Παράλληλα με την συλλογή και καταγραφή των πληροφοριών που αναφέρθηκαν προηγουμένως ο αναλυτής προσδιορίζει τις απαιτήσεις. Δημιουργεί μία λίστα με τον τρόπο που τις περιγράφει ο πελάτης και συνεχίζει την εργασία του με τον έλεγχο, τον εντοπισμό αντιφάσεων και την αναθεώρησή τους. Στο τελικό στάδιο της διαδικασίας οι αναλυτές ομαδοποιούν και ταξινομούν αυτές τις απαιτήσεις.

Συνοπτική περιγραφή πρώτης φάσης

Φάση 1η: Προδιαγραφές συστήματος

Στην πρώτη φάση καθορίζονται ευκρινώς οι προδιαγραφές του συστήματος. Συγκεντρώνονται και καθορίζονται οι απαιτήσεις για τη δημιουργία του μέσω της παρατήρησης των διαδικασιών και των λειτουργιών της επιχείρησης, καθώς και μέσω συνεντεύξεων και συμπλήρωσης ερωτηματολογίων από το προσωπικό της επιχείρησης, όπως προαναφέρθηκε πιο αναλυτικά στο κεφάλαιο ανάλυσης των απαιτήσεων. Σκοπός της φάσης αυτής είναι η συγκέντρωση της περιγραφής των ιδιοτήτων του συστήματος με τη μορφή των απαιτήσεων όπως αυτές εκφράζονται από το εργασιακό περιβάλλον όπου και θα εγκατασταθεί το σύστημα. Η συγκέντρωση αυτών των απαιτήσεων έχει κυρίως τη μορφή κειμένου. Κατά την ολοκλήρωση της φάσης αυτής γνωρίζουμε τις προδιαγραφές του συστήματος και περνάμε στο επόμενο στάδιο, που είναι η φάση της ανάπτυξής του.

2.3 Ανάπτυξη και τεχνικές τεκμηρίωσης συστημάτων ή ΤΤΣ

Για να μπορέσουμε να συνεχίσουμε με την περιγραφή της ανάπτυξης και των επακόλουθων φάσεων κρίνεται απαραίτητη η κατανόηση των τεχνικών τεκμηρίωσης συστημάτων.

Με τον όρο τεχνικές τεκμηρίωσης συστημάτων αναφερόμαστε στις μεθόδους εκείνες που χρησιμοποιούνται για την ανάλυση, τον σχεδιασμό, και την τεκμηρίωση των φάσεων και των υποφάσεων ενός πληροφοριακού συστήματος. Αυτές οι μέθοδοι αποτυπώνονται κυρίως μέσω της χρήσης διαγραμματικών απεικονίσεων και για το λόγο αυτόν αναφερόμαστε σε αυτές με τον όρο γραφικές τεχνικές. Η χρήση των διαγραμμάτων προσφέρει τον απαραίτητο βαθμό εποπτείας για την αναπαράσταση, την ανάλυση και τον έλεγχο ενός πληροφοριακού συστήματος όπως προαναφέρθηκε. Οι γραφικοί μέθοδοι ανάλυσης και τεκμηρίωσης χρησιμοποιούνται για τη δημιουργία ενός πληροφοριακού συστήματος και αποτελούν σημαντικό μέρος αυτού, αφού το συνοδεύουν καθ' όλη τη διάρκεια της ζωής του.

Γνώστες και χρήστες των τεχνικών αυτών είναι οι τεχνικοί προγραμματιστές, οι

λογιστές, οι ελεγκτές και γενικότερα το προσωπικό που ασχολείται με την ανάλυση και τον σχεδιασμό ενός συστήματος, αλλά και οποιοσδήποτε θέλει να κατανοήσει να βελτιώσει ή και να ελέγξει ένα πληροφοριακό σύστημα.

2.3.1 Η δομημένη και η αντικειμενοστραφής προσέγγιση

Ο τρόπος προσέγγισης της σύλληψης της ιδέας, η αποτύπωση της μεθοδολογίας και οι τεχνικές τεκμηρίωσης που διατίθενται προς χρήση για την κατασκευή ενός μοντέλου κύκλου ζωής λογισμικού με το πέρασμα του χρόνου διαφοροποιήθηκαν και εμπλουτίστηκαν σε βαθμό τόσο διακριτό που κρίθηκε αναγκαίος ο διαχωρισμός της νέας από την παλαιά προσέγγιση δημιουργώντας έτσι δυο μεγάλες κατηγορίες, την δομημένη και την αντικειμενοστραφή προσέγγιση ανάπτυξης ενός πληροφοριακού συστήματος.

2.3.1.1 Δομημένη προσέγγιση ανάπτυξης συστήματος

Σύμφωνα με τη δομημένη προσέγγιση το πληροφοριακό σύστημα θεωρείται ότι αποτελεί μια ενιαία και σχετικά αυτοτελή, μεγάλη και σύνθετη λειτουργική μονάδα. Η δομημένη ανάλυση, η δομημένη σχεδίαση και ο δομημένος προγραμματισμός είναι τρεις τεχνικές που αποτελούν την δομημένη ανάπτυξη ή αλλιώς SADT (Structured Analysis and Design Technique).

Σκοπός της δομημένης ανάλυσης είναι η πλήρης κατανόηση του προβλήματος. Μόνο όταν το πρόβλημα καθοριστεί ακριβώς είναι δυνατή η εγγύηση της αποτελεσματικότητας του συστήματος. Αν οι απαιτήσεις του συστήματος δεν είναι κατανοητές τότε πιθανότατα η ανάπτυξη λογισμικού να μη δώσει λύση στο απαιτούμενο πρόβλημα αλλά θα απευθύνεται σε κάτι άλλο.

ΤΤΣ στη δομημένη ανάπτυξη συστήματος

Οι τεχνικές της δομημένης ανάλυσης βοηθούν τον αναλυτή στο να ορίσει το τι ακριβώς πρέπει να κάνει το σύστημα, ποια είναι τα δεδομένα που θα αποθηκευτούν, ποιες είσοδοι και έξοδοι απαιτούνται και ποιές διαδικασίες απαρτίζουν το σύστημα. Για αυτό το λόγο αναπτύχθηκαν διαγραμματικές τεχνικές όπως τα Διαγράμματα Ροής Δεδομένων (ΔΡΔ), οι πίνακες αποφάσεων, τα δένδρα αποφάσεων, τα διαγράμματα οντοτήτων συσχετίσεων και τα λεξικά δεδομένων.

Μικρή ιστορική αναδρομή

Οι τεχνικές του δομημένου προγραμματισμού αναπτύχθηκαν τη δεκαετία του '60 και αποτέλεσαν μια προσπάθεια για την παροχή οδηγιών που θα βελτιώναν την ποιότητα των προγραμμάτων. Οι τεχνικές της δομημένης σχεδίασης αναπτύχθηκαν τη δεκαετία

του '70, για να συνδεθούν διαφορετικά προγράμματα σε ένα πληροφοριακό σύστημα, καθώς παρείχαν οδηγίες για το είδος των προγραμμάτων και την ιεράρχησή τους. Για το σκοπό αυτό αναπτύχθηκαν τα Διαγράμματα Δομής ή ΔΔ. Οι τεχνικές της δομημένης ανάλυσης εξελίχθηκαν στις αρχές της δεκαετίας του '80 για να διευκρινιστούν οι απαιτήσεις ενός πληροφοριακού συστήματος, πριν οι αναλυτές σχεδιάσουν τα προγράμματα. Τα Διαγράμματα Ροής Δεδομένων (ΔΡΔ) αποτελούν τη διαγραμματική αποτύπωση των απαιτήσεων του συστήματος. Από τη φάση της ανάλυσης τα ΔΡΔ μετατρέπονται σε ΔΔ για να χρησιμοποιηθούν στη Σχεδίαση.

Τη δεκαετία του '80, αναπτύχθηκαν τεχνικές σχεδίασης, αρχείων και βάσεων δεδομένων. Τα Συστήματα Διαχείρισης Βάσεων Δεδομένων χρησιμοποιούνται στην δομημένη σχεδίαση με σκοπό την αλληλεπίδραση των προγραμμάτων με τη βάση δεδομένων και για αυτό το λόγο αναπτύχθηκαν τα Διαγράμματα Οντοτήτων Συσχετίσεων (ΔΟΣ). Ενώ παράλληλα αναπτύχθηκαν και τεχνικές σχεδίασης διεπαφής χρήστη όπως ο σχεδιασμός menus.

Τα υπέρ και τα κατά της δομημένης ανάπτυξης

Πλεονέκτημα της δομημένης ανάπτυξης αποτελεί η ανεξαρτησία των δεδομένων από το λογισμικό, η ευκολία στην κατανόηση του κώδικα και η συμβατότητα με τον δομημένο προγραμματισμό. Χαρακτηριστικό της δομημένης ανάπτυξης συστήματος είναι η αυστηρή πειθαρχία στην οργάνωση του κώδικα του προγράμματος και η σειριακή εκτέλεση των φάσεων ανάπτυξης που όμως πολλές φορές οδηγεί σε καθυστερήσεις στην υλοποίηση του έργου. Η δομημένη ανάπτυξη ενός συστήματος με την πάροδο του χρόνου εξελίχθηκε σε αρκετές παραλλαγές. Οι αναλυτές που έμειναν στις αρχικές εκδόσεις δεν χρησιμοποίησαν πολλές βελτιώσεις της, ενώ κάποιοι χρησιμοποίησαν τμήματα των νέων τεχνικών χωρίς να τις κατανοήσουν σε όλη τους την έκταση και να τις μάθουν με όλες τις λεπτομέρειες. Οι τεχνικές που χρησιμοποιούνται δεν καλύπτουν όλες τις ενέργειες της ανάλυσης και σχεδίασης. Επίσης, η μετάβαση από τα ΔΡΔ στα ΔΔ, δηλαδή από τη δομημένη ανάλυση στη δομημένη σχεδίαση, στην πράξη δεν λειτουργεί. Ένα ακόμη μειονέκτημα είναι η πολύ αργή ανάπτυξη, η οποία μπορεί να κρατήσει μερικά χρόνια για μεγάλα ΠΣ, γιατί έχει σειριακό χαρακτήρα αφού πρέπει πρώτα να τελειώσει η ανάλυση για να αρχίσει η σχεδίαση, κοκ. Τέλος οι αλλαγές στις επιχειρηματικές απαιτήσεις που αποτυπώνονται με μεγάλη δυσκολία και δεν επιτρέπουν την επαναχρησιμοποίηση τμημάτων λογισμικού (reuse).

2.3.1.2 Αντικειμενοστραφής προσέγγιση ανάπτυξης συστήματος

Μια εντελώς διαφορετική προσέγγιση στην ανάπτυξη των πληροφοριακών

συστημάτων αποτελεί η αντικειμενοστραφής προσέγγιση. Το σύστημα εξετάζεται ως συλλογή από αλληλεπιδρώντα αντικείμενα, τα οποία συνεργάζονται για την πραγματοποίηση ενός έργου. Το σύστημα αποτελείται από αντικείμενα που μπορούν να απαντούν σε μηνύματα. Αυτό οδήγησε στην ανάπτυξη διαφορετικών προσεγγίσεων στην ανάλυση, στον σχεδιασμό και στον προγραμματισμό των συστημάτων. Επειδή, λοιπόν, η αντικειμενοστραφής προσέγγιση μελετά τα πληροφοριακά συστήματα ως συλλογές από αντικείμενα που αλληλεπιδρούν, η αντικειμενοστραφής ανάλυση ή ΟΟΑ (Object-Oriented Analysis) καθορίζει όλα τα είδη των αντικειμένων που αλληλεπιδρούν με το σύστημα και μελετά τις αλληλεπιδράσεις χρήστη που απαιτούνται για την ολοκλήρωση των εργασιών. Ο αντικειμενοστραφής σχεδιασμός ή ΟΟΔ (Object-Oriented design) καθορίζει όλους τους τύπους των αντικειμένων που είναι απαραίτητοι, δείχνει πώς τα αντικείμενα αλληλεπιδρούν για να ολοκληρωθούν οι εργασίες και τελειοποιεί τον ορισμό του κάθε τύπου αντικειμένου έτσι ώστε να μπορεί να υλοποιηθεί σε μια συγκεκριμένη γλώσσα. Ο αντικειμενοστραφής προγραμματισμός ή ΟΟΡ (Object-Oriented Programming) αποτελείται από γραπτές δηλώσεις σε μια γλώσσα προγραμματισμού για να καθορίσει τι κάνει κάθε τύπος αντικειμένου. Ένα αντικείμενο είναι ένας τύπος πράγματος, ένας πελάτης, ένας υπάλληλος ή ένας μαθητής, καθώς επίσης και ένα κουμπί ή ένα μενού. Κάποια πράγματα, όπως οι πελάτες, υπάρχουν τόσο έξω από το σύστημα (αληθινός πελάτης) όσο και χωριστά στο εσωτερικό του συστήματος (μια αναπαράσταση στον υπολογιστή του πελάτη). Η αντικειμενοστραφής ανάπτυξη πληροφοριακών συστημάτων χρησιμοποιεί ένα διάγραμμα κλάσεων για την εμφάνιση όλων των κλάσεων των αντικειμένων στο σύστημα.

Η προσέγγιση αυτή αναπτύχθηκε όταν στη δεκαετία του '70 άρχισαν να αναπτύσσονται οι αντικειμενοστραφείς γλώσσες προγραμματισμού.

Πλεονεκτήματα αντικειμενοστραφούς προσέγγισης

Η αντικειμενοστραφής προσέγγιση έχει διάφορα βασικά πλεονεκτήματα, μεταξύ των οποίων είναι η φυσικότητα και η επαναχρησιμοποίηση. Η προσέγγιση είναι φυσική για τους ανθρώπους, επειδή έχουν την τάση να σκέφτονται τον κόσμο ως αντικείμενα. Είναι λιγότερο φυσικό για έναν άνθρωπο να σκεφτεί πολύπλοκες διαδικασίες που αναπτύχθηκαν σε διαδικαστικές γλώσσες προγραμματισμού. Επίσης, επειδή η αντικειμενοστραφής προσέγγιση περιλαμβάνει κλάσεις αντικειμένων και πολλά συστήματα στην οργάνωση τους χρησιμοποιούν τα ίδια αντικείμενα, αυτές οι κλάσεις μπορούν να χρησιμοποιηθούν ξανά και ξανά κάθε φορά που χρειάζονται. Για παράδειγμα, σχεδόν όλα τα συστήματα χρησιμοποιούν μενού, πλαίσια διαλόγου, παράθυρα και κουμπιά, αλλά επίσης και πολλά συστήματα εντός της ίδιας εταιρείας

χρησιμοποιούν κλάσεις όπως πελάτης, προϊόν και τιμολόγιο που μπορούν να επαναχρησιμοποιηθούν.

Συμπερασματικά

Η αντικειμενοστραφής προσέγγιση διαφέρει αρκετά από την δομημένη προσέγγιση. Πολλά συστήματα που αναπτύσσονται σήμερα συνδυάζουν παραδοσιακές και αντικειμενοστραφείς προσεγγίσεις. Για το λόγο αυτό, είναι σημαντικό η κάλυψη τόσο των παραδοσιακών-δομημένων προσεγγίσεων όσο και των αντικειμενοστραφών προσεγγίσεων.

2.3.2 Κατηγορίες τεχνικών τεκμηρίωσης για την ανάπτυξη συστημάτων

Οι διαγραμματικές τεχνικές που είναι ευρύτερα διαδεδομένες και χρησιμοποιούνται στην ανάπτυξη του μοντέλου ενός πληροφοριακού συστήματος χωρίζονται κυρίως σε δύο κατηγορίες. Σε αυτές που χρησιμοποιούνται για τη γενική περιγραφή μιας λειτουργίας ή ενός προβλήματος και σε αυτές που χρησιμοποιούνται για την λεπτομερική περιγραφή τους. Υπάρχουν διαγραμματικές τεχνικές που ανήκουν και στις δύο κατηγορίες, δηλαδή μπορούν να χρησιμοποιηθούν τόσο για γενική όσο και για λεπτομερή περιγραφή των λειτουργιών ή και των προβλημάτων ενός πληροφοριακού συστήματος.

Οι πιο γνωστές και χρησιμοποιούμενες τεχνικές που ανήκουν στην κατηγορία των τεχνικών γενικής περιγραφής, είναι οι εξής:

- τα διαγράμματα ροής δεδομένων
- τα διαγράμματα δομής (structure diagrams)
- τα διαγράμματα HIPO (Hierarchical Input – Process – Output)
- τα διαγράμματα HOS (Higher Order Software)
- τα διαγράμματα Warnier -Orr
- τα διαγράμματα ενεργειών
- και τα διαγράμματα Jackson

Οι πιο γνωστές και χρησιμοποιούμενες τεχνικές που ανήκουν στην κατηγορία των τεχνικών λεπτομερούς περιγραφής, είναι οι εξής:

- η δομημένη περιγραφή και ο ψευδοκώδικας
- τα δένδρα αποφάσεων
- οι πίνακες αποφάσεων
- τα διαγράμματα Nassi-Shneiderman
- τα διαγράμματα HIPO (Hierarchical Input – Process – Output)
- τα διαγράμματα HOS (Higher Order Software)

- τα διαγράμματα Warnier Orr
- και τα διαγράμματα ενεργειών

Οι περισσότερες από τις παραπάνω διαγραμματικές τεχνικές χρησιμοποιούνται καθ' όλη τη διάρκεια της ανάλυσης, του σχεδιασμού, της υλοποίησης και της λειτουργίας ενός πληροφοριακού συστήματος.

Η Χρήση των Διαγραμματικών Τεχνικών στις Φάσεις του Κύκλου Ζωής						
		Διερευνητική Μελέτη	Ανάλυση Απαιτήσεων	Αρχιτεκτονική Προγράμματος	Λογική Προγράμματος	Δομή Αρχείων Β.Δ.
1	Δ. Ροής Δεδομένων	x	x			
2	Δ. Warnier - Orr	x	x	x	x	
3	Δ. Michael Jackson	x	x	x		
4	Δ. Ενεργειών	x	x		x	
5	Δ. Δομής			x	x	
6	Δ. HOS	x	x		x	
7	Δ. HIPO	x	x		x	
8	Δ. Οντοτήτων-Συσχετίσεων		x			x
9	Δέντρα Αποφάσεων	x	x		x	
10	Πίνακες Αποφάσεων	x	x			
11	Δ. Nassi - Shneiderman		x		x	
12	Διάγραμμα Δομής Δεδομένων					x

Πίνακας 2.3.2.1. Χρήση των τεχνικών τεκμηρίωσης συστημάτων στις φάσεις του κύκλου ζωής λογισμικού

Συνοπτική περιγραφή δεύτερης φάσης

Φάση 2η: Ανάπτυξη

Η φάση της ανάπτυξης αποτελείται από τρεις επιμέρους φάσεις ή αλλιώς υποφάσεις. Αυτήν της ανάλυσης, αυτήν του σχεδιασμού και τέλος αυτήν της υλοποίησης με την οποία ολοκληρώνεται αυτό το στάδιο. Παρακάτω παρατίθενται πιο αναλυτικά:

Υποφάση 1η: Ανάλυση

Κατά την ανάλυση πραγματοποιείται συστηματική και επιστημονική οργάνωση και αναπαράσταση των προδιαγραφών που συγκεντρώθηκαν στην προηγούμενη φάση. Σκοπός του σταδίου της ανάλυσης είναι να παρουσιαστούν με σαφή και τεκμηριωμένο τρόπο οι λειτουργίες του πληροφοριακού συστήματος και των υποσυστημάτων του, η ροή των πληροφοριών που αυτό θα υποστηρίζει, μια αρχική οργάνωση των δεδομένων-αρχείων και των βάσεων δεδομένων που αυτό θα χρησιμοποιεί και μια συστηματική καταγραφή των εννοιών και των όρων που θα χρησιμοποιούνται γενικά στην ανάπτυξη.

Οι τεχνικές τεκμηρίωσης συστημάτων βοηθούν σημαντικά τους αναλυτές στις παραπάνω εργασίες. Για την αναπαράσταση των λειτουργιών των διαφόρων υποσυστημάτων και της ροής των πληροφοριών μεταξύ των διαδικασιών τους χρησιμοποιούνται τα **διαγράμματα ροής δεδομένων (data flow diagrams ή ΔΡΔ)**. Για την περιγραφή με μεγαλύτερη λεπτομέρεια του τι ακριβώς κάνει κάθε υποσύστημα και κάθε διαδικασία χρησιμοποιούνται τα **διαγράμματα ροής (flow charts)** και τα **αναλυτικά διαγράμματα ροής**. Τέλος για τις βάσεις δεδομένων δημιουργούνται τα **διαγράμματα οντοτήτων συσχετίσεων (ΔΟΣ)**. Με την ολοκλήρωση της υποφάσης της ανάλυσης περνάμε στην επόμενη υποφάση, αυτήν του σχεδιασμού του πληροφοριακού συστήματος.

Υποφάση 2η: Σχεδιασμός

Κατά τη σχεδίαση θα πρέπει το σύστημα να διαμορφωθεί στα τελικά χαρακτηριστικά του, τα οποία πρέπει να περιγραφούν λεπτομερώς. Σκοπός της σχεδίασης είναι ο ακριβής προσδιορισμός των διαφόρων υποσυστημάτων και των διαδικασιών που τα αποτελούν, και ο λεπτομερής προσδιορισμός της λειτουργικότητας κάθε διαδικασίας, δηλαδή ο τρόπος με τον οποίο συνδέονται, συνεργάζονται και ανταλλάσσονται οι διαδικασίες και τα δεδομένα που χρησιμοποιούνται μεταξύ τους. Συνεπώς ορίζεται με ακρίβεια η αρχιτεκτονική του πληροφοριακού συστήματος.

Και σε αυτήν την υποφάση χρησιμοποιούνται γραφικές τεχνικές τεκμηρίωσης συστημάτων. Για τη λεπτομερή περιγραφή των προγραμμάτων έχουμε τα **διαγράμματα ροής προγράμματος (diagram flow charts)**, τα **διαγράμματα IPO** και **HIPO**, τον **ψευδοκώδικα** κ.ά. Στις βάσεις δεδομένων δημιουργείται η τελική μορφή των **διαγραμμάτων οντοτήτων συσχετίσεων ΔΟΣ** και σε περίπτωση χρήσης σχεσιακής βάσης δεδομένων δημιουργούνται οι τελικοί πίνακες ενώ προσδιορίζεται η τελική μορφή και ο τύπος κάθε πεδίου. Σε περίπτωση που το πληροφοριακό σύστημα περιλαμβάνει και διαχείριση εγγράφων, σχεδιάζονται οι ηλεκτρονικές φόρμες διαχείρισης και ο τρόπος δρομολόγησης αυτών. Με την επιτυχή ολοκλήρωση της σχεδίασης περνάμε στο επόμενο βήμα, που είναι η υποφάση της υλοποίησης.

Υποφαση 3η: Υλοποίηση

Σε αυτό το βήμα πραγματοποιείται η κωδικοποίηση, η συνένωση και ο έλεγχος των προγραμμάτων για τους υπολογιστές. Εκπαιδεύεται το προσωπικό, εγκαθίστανται ο νέος εξοπλισμός και διενεργείται η συγγραφή της υπόλοιπης τεκμηρίωσης του συστήματος (documentation).

Μέρος της τεκμηρίωσης αποτελούν διάφορα διαγράμματα που έχουν δημιουργηθεί στις υποφάσεις της ανάλυσης και του σχεδιασμού του συστήματος. Με την ολοκλήρωση της υλοποίησης και αφού έχουν δημιουργηθεί κατά κύριο λόγο τα εγχειρίδια χρήσης και συντήρησής του, ολοκληρώνεται η φάση της ανάπτυξης και περνάμε στην τρίτη κύρια φάση, αυτήν της επαλήθευσης.

Συνοπτική περιγραφή τρίτης φάσης

Φάση 3η: Επαλήθευση

Μπαίνοντας στο επόμενο βήμα διαδικασιών έχουμε το νέο πληροφοριακό σύστημα μαζί με τον νέο εξοπλισμό και το υπόλοιπο υλικό εγκατεστημένο στο εργασιακό περιβάλλον, και έτοιμο για τη δοκιμαστική του λειτουργία. Κατά τη φάση της επαλήθευσης λοιπόν οι εργαζόμενοι λειτουργούν το πληροφοριακό σύστημα δοκιμαστικά, εισάγοντας όμως πραγματικά δεδομένα, για τον έλεγχο της ορθής του λειτουργίας. Αφού διαπιστωθεί ότι πληρεί τις απαιτήσεις που είχαν διατυπωθεί στην αρχή και είναι απαλλαγμένο από λάθη ολοκληρώνεται η φάση της επαλήθευσης και περνάμε στην επόμενη και τελευταία φάση της κανονικής λειτουργίας του πληροφοριακού συστήματος.

Συνοπτική περιγραφή τέταρτης φάσης

Φάση 4η: Λειτουργία

Σε αυτό το στάδιο πραγματοποιούνται εργασίες συντήρησης, βελτίωσης και μετατροπής του συστήματος, όπου χρίζεται αυτό αναγκαίο.

Σημαντικές πληροφορίες

Σημαντικό ρόλο για τη φάση της επαλήθευσης και της λειτουργίας παίζει το υλικό τεκμηρίωσης που δημιουργήθηκε κατά την ανάπτυξη του συστήματος. Επίσης με την ολοκλήρωση κάθε φάσης και υποφάσης διενεργείται έλεγχος και όπου κριθεί αναγκαίο πραγματοποιείται οπιστοδρόμηση στην προηγούμενη φάση ή υποφάση του πληροφοριακού συστήματος, όπου γίνονται οι απαραίτητες μετατροπές. Το σύστημα

εμφανίζει τα τελικά του χαρακτηριστικά μετά την υλοποίησή του, όπου φαίνεται η συνένωση των διάφορων μονάδων από τις οποίες απαρτίζεται.

2.4 Είδη μοντέλων κύκλου ζωής λογισμικού ή MKZ

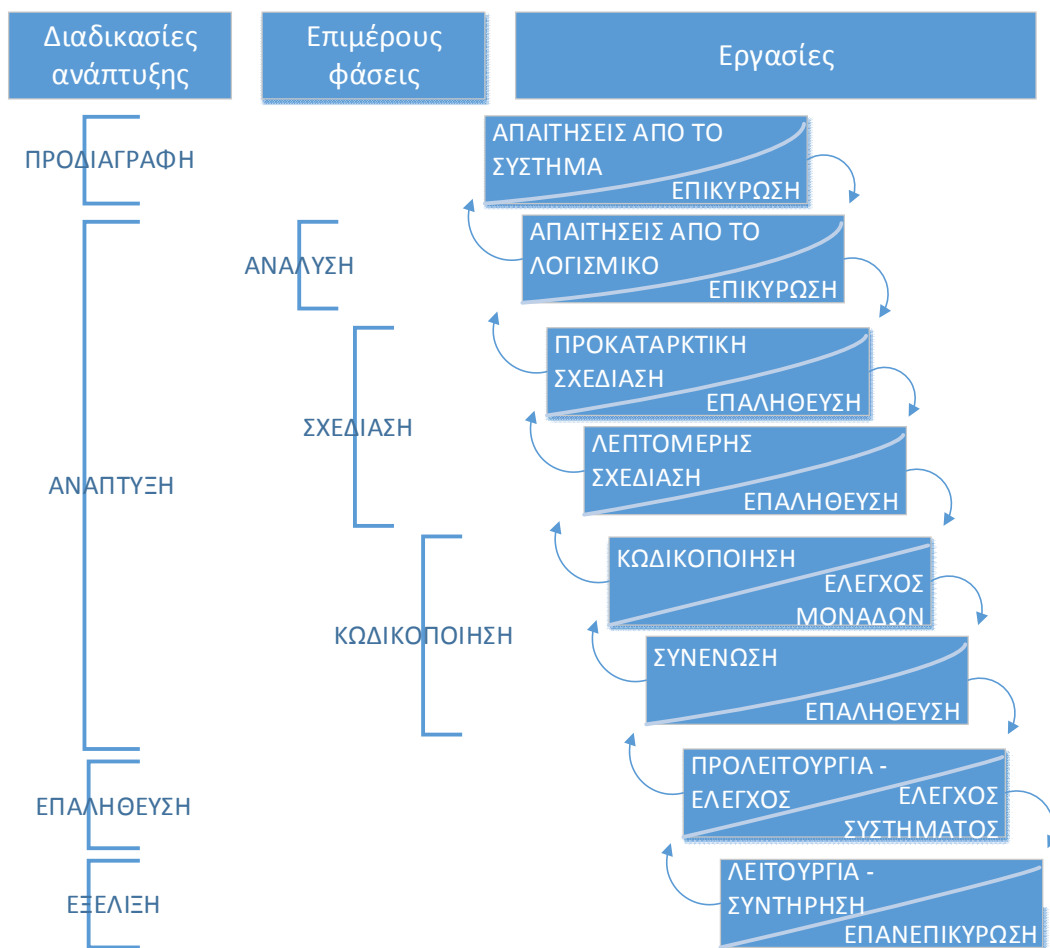
Υπάρχουν αρκετά μοντέλα κύκλου ζωής λογισμικού, τα οποία διαφοροποιούνται ως προς την σύλληψη της ιδέας, τον τρόπο κατασκευής τους, τις επιμέρους φάσεις που προτείνουν, την επαναληπτικότητα και την εμβέλεια των εργασιών τους, τα ενδιάμεσα συστατικά του λογισμικού και την περιγραφή τους, τις οικονομικές και επιχειρηματικές πλευρές της χρήσης τους κ.ά. Καθεμία από τις ενέργειες που περιγράφεται σε ένα μοντέλο κύκλου ζωής λογισμικού είναι μια διαδικασία επίλυσης προβλημάτων.

2.4.1 Το MKZ λογισμικού του καταρράκτη

Ένα από τα πιο διαδεδομένα μοντέλα κύκλου ζωής είναι αυτό του καταρράκτη. Η κεντρική ιδέα του μοντέλου αυτού θέλει το σύστημα λογισμικού να αναπτύσσεται περνώντας ολόκληρο από διαδοχικές επιμέρους φάσεις, καθεμία από τις οποίες θεωρείται περατωμένη με την παραγωγή ορισμένων συστατικών λογισμικού. Κάθε επιμέρους φάση ολοκληρώνεται με μια εργασία επαλήθευσης ή αλλιώς επικύρωσης των προϊόντων της, κατά την οποία αποφασίζεται η μετάβαση ή όχι στην επόμενη. Το λογισμικό εμφανίζεται πλήρες, δηλαδή με όλα τα λειτουργικά του χαρακτηριστικά, από την επιμέρους φάση της συνένωσης και μετά. Χαρακτηριστικό του μοντέλου αυτού είναι το ότι για να ξεκινήσει μια φάση πρέπει να έχει ολοκληρωθεί πλήρως η προηγούμενη. Η ανάπτυξη με τον τρόπο αυτό χαρακτηρίζεται ακολουθιακή, διότι οι επιμέρους φάσεις από τις οποίες διέρχεται είναι διακριτές και ακολουθούν η μία την άλλη.

Αρχικά καθορίζονται οι απαιτήσεις από το σύστημα και το λογισμικό, αντίστοιχα. Το λογισμικό είναι μια μόνο από τις συνιστώσες του συστήματος, το οποίο μπορεί να περιλαμβάνει και άλλες ειδικές συσκευές, άλλες εφαρμογές λογισμικού κ.ά. Έπειτα γίνεται η προκαταρκτική και η λεπτομερής σχεδίασή του. Κατά την προκαταρκτική σχεδίαση καθορίζονται οι μονάδες που θα αποτελούν το λογισμικό, καθώς και οι συσχετίσεις μεταξύ τους. Ο καθορισμός αυτός μπορεί να γίνει σε περισσότερα από ένα επίπεδα λεπτομέρειας, ανάλογα με το μέγεθος και την πολυπλοκότητά του. Το πρώτο επίπεδο, που είναι αυτό με τη μικρότερη λεπτομέρεια, περιέχει τα υποσυστήματα, το δεύτερο περιέχει τις μονάδες μέσα σε κάθε υποσύστημα κοκ. Κατά τη λεπτομερή σχεδίαση καθορίζεται η εσωτερική δομή κάθε μονάδας λογισμικού, η οποία αντιστοιχεί πρακτικά σε μονάδες πηγαίου κώδικα προγράμματος.

Ο καθορισμός αυτός περιλαμβάνει όλα τα απαραίτητα στοιχεία όπως αλγόριθμοι, δομές δεδομένων κτλ., ώστε η συγγραφή του πηγαίου κώδικα, που ακολουθεί, να είναι μια διαδικασία διεκπεραίωσης και μόνο. Ακολουθεί η συνένωση των μονάδων σε σύστημα και ο έλεγχος του συστήματος, η ολοκλήρωση του οποίου επιτρέπει την παράδοση ολόκληρου του προϊόντος στον πελάτη και το πέρασμα στη φάση της λειτουργίας και συντήρησης.



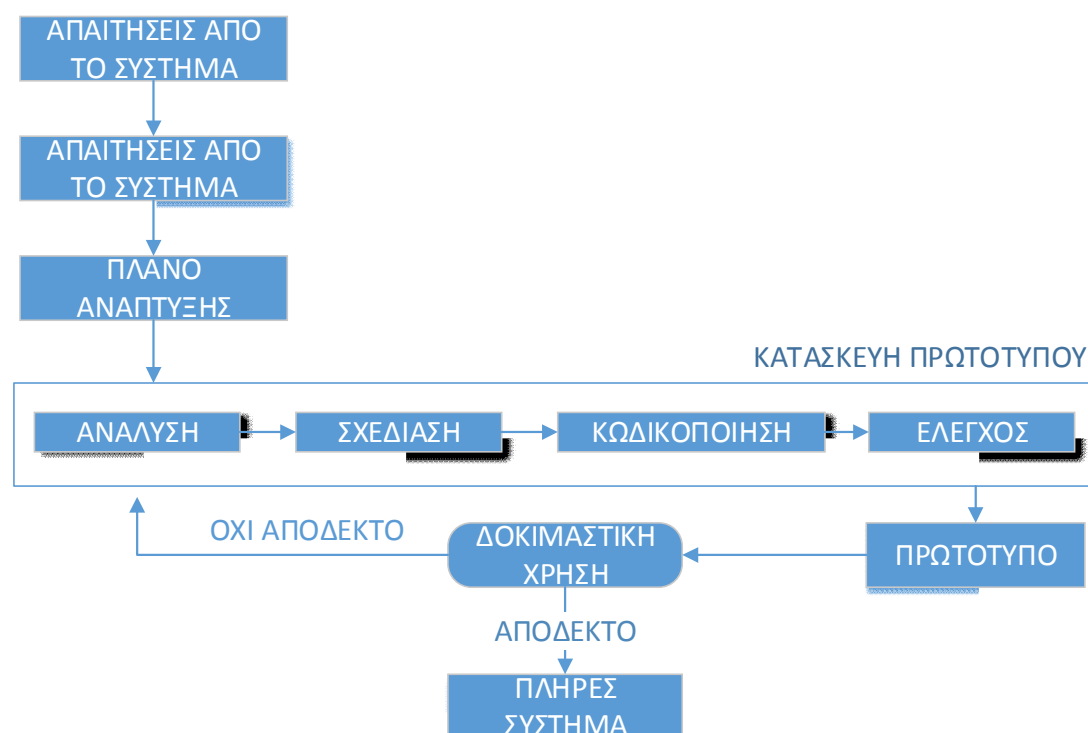
Σχήμα 2.4.1.1 Απεικόνιση του μοντέλου κύκλου ζωής λογισμικού του καταρράκτη

Το μοντέλο του καταρράκτη υπήρξε για μεγάλο διάστημα το πιο διαδεδομένο μοντέλο κύκλου ζωής λογισμικού. Είναι ιδιαίτερα χρήσιμο σε περιπτώσεις όπου οι απαιτήσεις από το λογισμικό είναι από την αρχή γνωστές και δε μεταβάλλονται κατά την ανάπτυξή του. Μπορεί, λοιπόν, να χρησιμοποιηθεί αποτελεσματικά για τη βιομηχανοποίηση της ανάπτυξης εφαρμογών, όπως αυτών της επίλυσης μεγάλων προβλημάτων με τη χρήση μαθηματικών υπολογισμών. Σε πολλές, όμως, περιπτώσεις εφαρμογών οι απαιτήσεις, είτε δεν είναι από την αρχή και με σαφήνεια γνωστές, είτε ενδέχεται να μεταβληθούν κατά τη διάρκεια της ανάπτυξης.

Η συνοπτική περιγραφή ανάπτυξης ενός πληροφοριακού συστήματος που παραθέσαμε παραπάνω, έχει ως βάση το μοντέλο κύκλου ζωής λογισμικού του καταρράκτη.

2.4.2 Το MKZ λογισμικού της πρωτοτυποποίησης

Η κεντρική ιδέα του μοντέλου πρωτοτυποποίησης είναι η ανάπτυξη του λογισμικού όχι εξ' ολοκλήρου, αλλά σε τμήματα, που ονομάζονται πρωτότυπα. Οι διαδικασίες ανάπτυξης επαναλαμβάνονται για ένα τμήμα του συστήματος κάθε φορά και, για το λόγο αυτό, το μοντέλο χαρακτηρίζεται ως επαναληπτικό. Κάθε πρωτότυπο περιλαμβάνει τις βασικές από τις λειτουργίες που προορίζεται να εκτελεί το λογισμικό και τίθεται σε δοκιμασία από τον πελάτη. Από εκεί συλλέγονται παρατηρήσεις και η διαδικασία κατασκευής νέου πρωτοτύπου επαναλαμβάνεται μέχρις ότου ένα πρωτότυπο να ικανοποιεί τις απαιτήσεις, δηλαδή να εκτελεί τις επιθυμητές λειτουργίες με τρόπο ικανοποιητικό και να γίνεται αποδεκτό από τον πελάτη. Από το σημείο αυτό και μετά μπορούν να προστεθούν και οι υπόλοιπες λειτουργίες, ώστε το λογισμικό να ολοκληρωθεί.



Σχήμα 2.4.2.1 Απεικόνιση του μοντέλου κύκλου ζωής λογισμικού πρωτοτυποποίησης

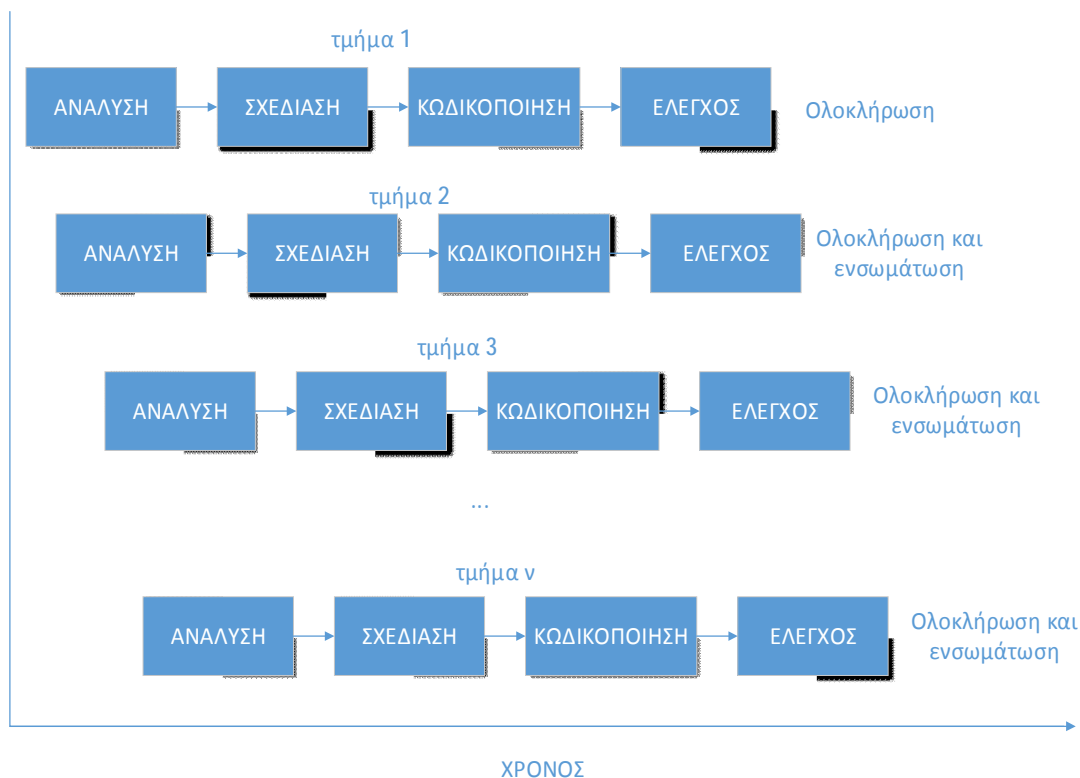
Ένα σημαντικό πλεονέκτημα του μοντέλου αυτού είναι η δυνατότητα απόκτησης

άποψης για την εφαρμογή λογισμικού νωρίτερα απ ό τι στο μοντέλο του καταρράκτη. Αυτό μπορεί να γλιτώσει την ανάπτυξη από καθυστερήσεις και συνεπαγόμενα κόστη ή ακόμη και από ολική αποτυχία, τα οποία θα επέρχονταν, αν ο κατασκευαστής αναγκαζόταν να οπισθοδρομήσει την ανάπτυξη, ενώ αυτή είχε προχωρήσει πολύ. Παράλληλα, ιδιαίτερη σημασία αποκτά η διοίκηση του έργου, η οποία πρέπει να εξασφαλίζει την υλοποιησιμότητα του πρωτοτύπου και την εύκολη τροποποίησή του. Κάθε κατασκευή πρωτοτύπου μπορεί να θεωρηθεί ως ένα μικρό έργο λογισμικού, το οποίο κατασκευάζεται με διαδικασίες που μπορούν να ακολουθούν άλλα μοντέλα κύκλου ζωής, όπως αυτό του καταρράκτη.

Με βάση τις παραπάνω παρατηρήσεις, το μοντέλο πρωτοτυποποίησης χρησιμοποιείται στην ανάπτυξη εφαρμογών λογισμικού για τις απαιτήσεις από τις οποίες δεν υπάρχει βεβαιότητα στην αρχή της ανάπτυξης, οπότε δεν μπορούν να συμφωνηθούν και να παγιωποιηθούν. Τέτοιες είναι εφαρμογές που κατασκευάζονται για πρώτη φορά ή που είναι στενά εξαρτημένες από τον πελάτη, χωρίς να υπάρχει αποδεκτό προηγούμενο παράδειγμα. Ωστόσο, το μέγεθος των εφαρμογών αυτών δεν μπορεί να είναι ιδιαίτερα μεγάλο, διότι ο χρόνος ανάπτυξης κάθε πρωτοτύπου μεγαλώνει και η απαιτούμενη ευελιξία μειώνεται.

2.4.3 Το MKZ λογισμικού της λειτουργικής επαύξεσης

Το μοντέλο της λειτουργικής επαύξεσης συνδυάζει την ακολουθιακή ανάπτυξη του μοντέλου του καταρράκτη με την τμηματική ανάπτυξη του μοντέλου της πρωτοτυποποίησης. Κεντρική ιδέα είναι η κατάτμηση του υπό κατασκευή λογισμικού σε τμήματα που αναπτύσσονται ανεξάρτητα, ακολουθώντας το καθένα ακολουθιακή ανάπτυξη σύμφωνα με το μοντέλο του καταρράκτη. Κατά την αρχική φάση ανάλυσης και σχεδίασης αποφασίζονται τα τμήματα στα οποία θα κατατμηθεί η εφαρμογή, η ανάπτυξη των οποίων γίνεται στη συνέχεια ανεξάρτητα και παράλληλα. Όταν ολοκληρώνεται η ανάπτυξη κάθε τμήματος, αυτό ενσωματώνεται στο σύνολο της εφαρμογής, διαδικασία η οποία δικαιολογεί και την ονομασία αυτού του μοντέλου κύκλου ζωής πληροφοριακού συστήματος.



Σχήμα 2.4.3.1 Απεικόνιση του μοντέλου κύκλου ζωής λογισμικού λειτουργικής επαύξεσης

Πλεονεκτήματα της ιδέας αυτής είναι η δυνατότητα παράλληλης ανάπτυξης, η οποία διαρκεί λιγότερο, και ο διαδοχικός εμπλουτισμός των λειτουργικών χαρακτηριστικών του λογισμικού. Ενώ βασικά μειονεκτήματα του μοντέλου αποτελούν τα τυχόν σφάλματα κατά την αρχική κατάτμηση και γενική σχεδίαση του συστήματος, τα οποία μπορούν να έχουν σημαντικές επιπτώσεις στο λογισμικό που θα κατασκευαστεί. Αλλά και η περίπτωση μεταβολής των λειτουργικών απαιτήσεων κατά τη χρήση του ημιτελούς συστήματος, που μπορεί να μεταβάλει την αρχιτεκτονική του, σε βαθμό που να κλονιστεί η ανάπτυξη των υπόλοιπων τμημάτων του. Για τον λόγο αυτόν το μοντέλο της λειτουργικής επαύξεσης χρησιμοποιείται κατά βάση στην ανάπτυξη μεγάλων εφαρμογών λογισμικού, για τις οποίες ισχύουν οι απαιτήσεις του μοντέλου του καταρράκτη, δηλαδή η σαφής γνώση και η μικρή έως μηδαμινή μεταβλητότητα των απαιτήσεων κατά την ανάπτυξη.

2.4.4 Το σπειροειδές MKZ λογισμικού

Τα μοντέλα που παρουσιάστηκαν μέχρι τώρα αποτελούν παραλλαγές της βασικής ιδέας του μοντέλου του καταρράκτη. Η ανάπτυξη παραμένει, επί της ουσίας, μια

ακολουθιακή διαδικασία, η οποία εφαρμόζεται είτε σε ολόκληρο είτε σε ένα μέρος του συστήματος. Από ότι φαίνεται, δεν είναι η σύλληψη των διαδικασιών ανάπτυξης λογισμικού που διαφοροποιεί τα μοντέλα κύκλου ζωής, αλλά η διάταξή τους. Στα μοντέλα της πρωτοτυποποίησης και της λειτουργικής επαύξησης η κατάτμηση είναι λίγο έως πολύ αυθαίρετη και το ρίσκο δεν αποτιμάται με αποτέλεσμα κάθε οπισθοδρόμηση ή ανατροπή να κοστίζει σε χρόνο, σε οικονομικούς όρους, αλλά πολλές φορές και σε συνολική αποτυχία των έργων. Από την άλλη, η μετά πειθαρχίας αποδοχή των αυστηρών φάσεων που προτείνονται από το μοντέλο του καταρράκτη δεν είναι εφικτό να ακολουθείται σε όλες τις περιπτώσεις και από όλους τους κατασκευαστές, με αποτέλεσμα η ανάπτυξη λογισμικού είτε να γίνεται άναρχα, με βάση τη διαίσθηση των κατασκευαστών, είτε να είναι μια δαπανηρή και στρυφνή διαδικασία, στην οποία πρέπει να ακολουθηθούν κάποια συγκεκριμένα βήματα, ανεξάρτητα από τις εκάστοτε συνθήκες.

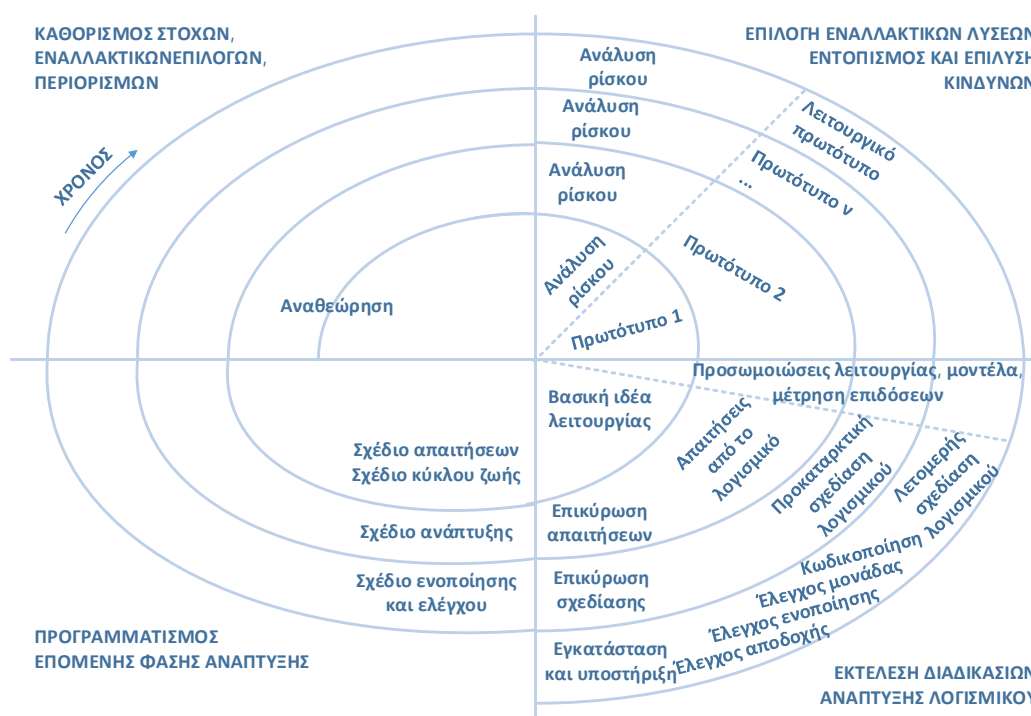
Απάντηση στα παραπάνω έρχεται να δώσει το σπειροειδές μοντέλο, το οποίο πήρε το όνομά του από την διαγραμματική του απεικόνιση. Πρόκειται για μια γενίκευση των μοντέλων της λειτουργικής επαύξησης και της πρωτοτυποποίησης, με σημαντικά νέα στοιχεία, ως εξής:

- Οι φάσεις και οι διαδικασίες ανάπτυξης λογισμικού δεν είναι προκαθορισμένες από το μοντέλο, αλλά εξειδικεύονται στο χώρο εφαρμογής του.
- Η ανάπτυξη ολόκληρου του συστήματος χωρίζεται σε πολλούς κύκλους, σε καθέναν από τους οποίους προστίθενται νέα λειτουργικά χαρακτηριστικά στο σύστημα.
- Ενώ πριν από την έναρξη κάθε κύκλου γίνεται μια μελέτη σκοπιμότητας και ανάλυση των κινδύνων, από την οποία προκύπτουν οι συγκεκριμένες εργασίες που θα εκτελεστούν μέσα στον κύκλο, αλλά και η ίδια η εφικτότητα εκτέλεσης του κύκλου αυτού.

Στο σπειροειδές μοντέλο διακρίνονται τέσσερις κατηγορίες εργασιών, ο προσδιορισμός των στόχων, ο εντοπισμός με την επίλυση των κινδύνων, η εκτέλεση των διαδικασιών ανάπτυξης και επαλήθευσης, και οι εργασίες προγραμματισμού. Κατά τον προσδιορισμό των στόχων καθορίζονται τα αντικείμενα εργασιών κάθε επανάληψης, καταγράφονται οι περιορισμοί επί του προϊόντος, αλλά και επί της διαδικασίας για την οποία κατασκευάζεται ένα αναλυτικό πλάνο διοίκησης. Επίσης, καταγράφονται οι κίνδυνοι που εμπεριέχει η διαδικασία και οι εναλλακτικές λύσεις, όπου υπάρχουν. Κατά τις εργασίες επίλυσης κινδύνων αναλύονται οι κίνδυνοι που έχουν καταγραφεί και αποτιμάται κάθε εναλλακτική λύση. Στο σημείο αυτό λαμβάνονται αποφάσεις για τη συνέχιση ή όχι της ανάπτυξης, για το μοντέλο που θα ακολουθηθεί στη συγκεκριμένη επανάληψη, για την κατασκευή ή όχι πρωτοτύπου κ.ά.

Ακολουθεί η εκτέλεση των βημάτων της διαδικασίας ανάπτυξης λογισμικού που έχει επιλεγεί για το τμήμα εκείνο του συστήματος που αφορά η τρέχουσα επανάληψη. Τέλος, μετά την επαλήθευση των αποτελεσμάτων, δηλαδή των ενδιάμεσων προϊόντων λογισμικού, γίνεται προγραμματισμός της συνέχισης της ανάπτυξης.

Το σπειροειδές μοντέλο δεν καθορίζει εκ των προτέρων τις εργασίες ανάπτυξης λογισμικού που πρέπει να γίνουν, ούτε την έκταση στην οποία αυτές θα εφαρμοστούν. Διαφορετικές διαδικασίες ανάπτυξης μπορεί να επιλεγούν για διαφορετικά τμήματα του λογισμικού. Αυτό που προτείνει είναι ότι ο καθορισμός των λεπτομερειών υλοποίησης πρέπει να γίνεται συνεχώς κατά την ανάπτυξη, και όχι μία φορά, όπως συμβαίνει με τα μοντέλα κύκλου ζωής που αναφέρθηκαν μέχρι τώρα, με ευθύνη και με τεκμηρίωση από πλευράς του ίδιου του κατασκευαστή.



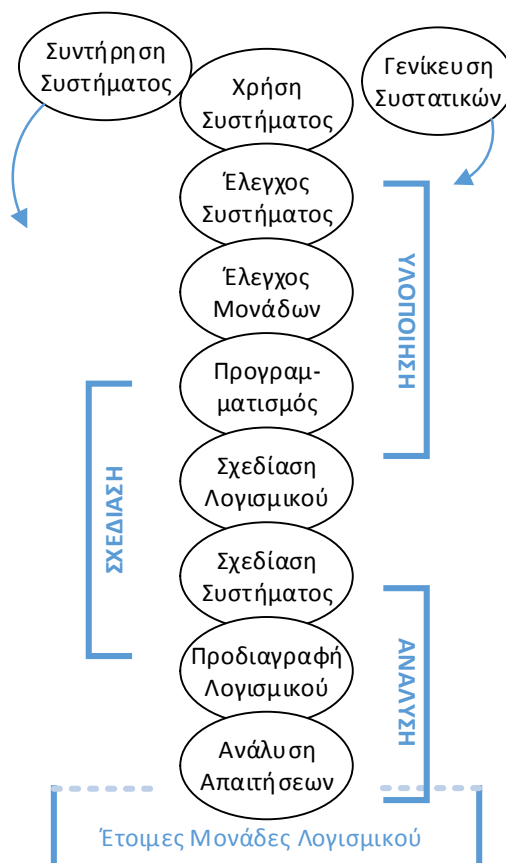
Σχήμα 2.4.4.1 Απεικόνιση του σπειροειδούς μοντέλου κύκλου ζωής λογισμικού

Η εφαρμογή του σπειροειδούς μοντέλου στην πράξη δεν είναι πάντα εύκολη υπόθεση. Εισάγονται νέες εργασίες, που δεν ανήκουν καθαρά στις εργασίες ανάπτυξης λογισμικού, αλλά αφορούν στην τεκμηρίωση της σκοπιμότητας και τον τμηματικό προγραμματισμό της ανάπτυξης. Οι εργασίες αυτές επιφέρουν ασφαλώς

κάποιο κόστος, το οποίο όμως μπορεί να αποσβεστεί από τον έγκαιρο εντοπισμό προβλημάτων και την αποφυγή πιθανής αποτυχίας του πληροφοριακού συστήματος.

2.4.5 Το MKZ λογισμικού του πίδακα

Αρκετά μοντέλα κύκλου ζωής που έχουν προταθεί αποτελούν παραλλαγές αυτών που αναφέρθηκαν, τα χαρακτηριστικά των οποίων υποβάλλονται από τις μεθοδολογίες ανάπτυξης. Οι πρώτες προσεγγίσεις του θέματος με βάση την αντικειμενοστρεφή τεχνολογία διαφοροποίησαν τα επερχόμενα μοντέλα βασιζόμενες σε δύο ιδιαίτερα γνωρίσματά της: πρώτον, ότι οι έννοιες της ανάλυσης, της σχεδίασης, και της κωδικοποίησης έρχονται στο αντικειμενοστρεφές παράδειγμα πολύ πιο κοντά και, δεύτερον, ότι το αποτέλεσμα κάθε διαδικασίας κατασκευής λογισμικού είναι όχι μόνο ένα σύστημα, αλλά και επαναχρησιμοποιήσιμες μονάδες, οι οποίες μπορούν να χρησιμοποιηθούν από τις πρώτες φάσεις της ανάπτυξης μελλοντικών συστημάτων. Με τον τρόπο αυτό προέκυψε το μοντέλο του πίδακα.



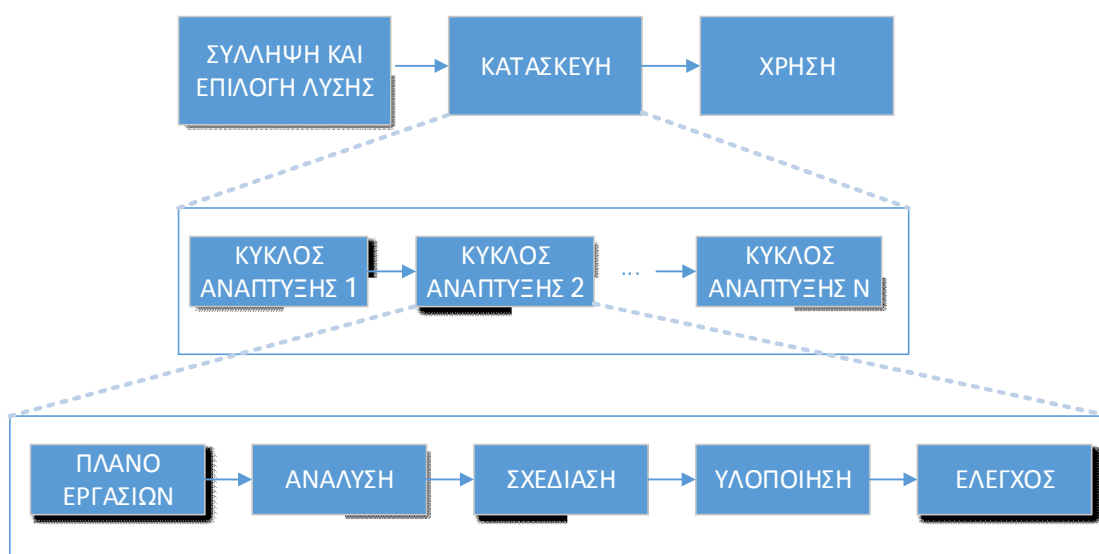
Σχήμα 2.4.5.1 Απεικόνιση του μοντέλου κύκλου ζωής λογισμικού του πίδακα

Κατά την ανάπτυξη παρατηρούνται επικαλύψεις των φάσεων της ανάλυσης, της

σχεδίασης και της κωδικοποίησης. Κατά το τέλος της ανάπτυξης, ορισμένα από τα συστατικά λογισμικού που έχουν παραχθεί ενσωματώνονται σε μια δεξαμενή συστατικών και διατίθενται για να χρησιμοποιηθούν στην ανάπτυξη και νέων συστημάτων. Η ιδέα του μοντέλου κύκλου ζωής του πίδακα τονίζει περισσότερο τα επιθυμητά χαρακτηριστικά της μεθοδολογίας κατασκευής του λογισμικού σύμφωνα με την αντικειμενοστρεφή λογική, ήταν δε αρκετά επίκαιρη κατά την έκρηξη ενδιαφέροντος για την αντικειμενοστρεφή τεχνολογία στα τέλη της δεκαετίας του 80 και στις αρχές της δεκαετίας του 90.

2.4.6 Σύγχρονα MKZ λογισμικού

Μεταγενέστερα μοντέλα κύκλου ζωής λογισμικού προσπαθούν να δώσουν μια γενική κατεύθυνση εφαρμογής των ήδη υπάρχοντων ιδεών, αφήνοντας σημαντικούς βαθμούς ελευθερίας στον κατασκευαστή που τα ακολουθεί. Αυτό είναι ιδιαίτερα επιθυμητό, διότι η αυστηρή πειθαρχία που επιχειρήθηκε να εισαχθεί τα πρώτα χρόνια της έκρηξης της χρήσης του λογισμικού δε συμβάδιζε με την ωριμότητα σκέψης που διέθετε η τεχνική κοινότητα την εποχή εκείνη ούτε και μπορούσε να παρακολουθήσει τους υψηλούς ρυθμούς εξελίξεων στο χώρο της πληροφορικής.



Σχήμα 2.4.6.1 Απεικόνιση του γενικού μοντέλου κύκλου ζωής λογισμικού

Μια περιγραφή ενός σύγχρονου μοντέλου κύκλου ζωής λογισμικού περιέχει μόνο γενικές κατευθύνσεις, οι οποίες εξειδικεύονται στο εκάστοτε περιβάλλον ανάπτυξης, πρόβλημα κτλ. Επίσης, δεν είναι άρρηκτα συνδεδεμένο με κάποια μεθοδολογία ανάπτυξης λογισμικού, αλλά μπορεί να εξειδικευτεί για την πρακτική

του κάθε κατασκευαστή. Ένα τέτοιο μοντέλο μπορεί να χαρακτηριστεί ως απόγονος πολλών από τα μοντέλα που προαναφέρθηκαν. Το γενικό πλαίσιο του μοντέλου αυτού περιλαμβάνει τις φάσεις σύλληψης, κατασκευής και λειτουργίας. Καθεμιά από αυτές αναλύεται σε επιμέρους εργασίες, σύμφωνα με τα χαρακτηριστικά του εκάστοτε περιβάλλοντος. Ιδιαίτερα η γενική φάση της κατασκευής αναλύεται σε κύκλους ανάπτυξης, καθένας εκ των οποίων προσθέτει νέα χαρακτηριστικά και λειτουργίες στο υπό κατασκευή λογισμικό. Τα επιμέρους βήματα μέσα σε κάθε κύκλο ανάπτυξης μοιάζουν με τα βήματα του μοντέλου του καταρράκτη, μόνο που δεν εφαρμόζονται για ολόκληρο το σύστημα, αλλά για το μικρό μέρος του που κατασκευάζεται στον εν λόγω κύκλο, όπως στο μοντέλο της πρωτοτυποποίησης. Για την εκκίνηση κάθε κύκλου ανάπτυξης μπορεί να έχει προηγηθεί ανάλυση ρίσκου και σκοπιμότητας, όπως στο σπειροειδές μοντέλο. Ζητήματα, όπως αλληλουχία των ενεργειών, ακριβής καθορισμός των κύκλων ανάπτυξης κ.ά., αφήνονται στη διακριτική ευχέρεια του κάθε κατασκευαστή, από τον οποίο και καθορίζονται σύμφωνα με τις ιδιαιτερότητες κάθε περίπτωσης. Ένα πραγματικό τέτοιο πλαίσιο ανάπτυξης προτείνεται από τη μεθοδολογία Rational Unified Process, η οποία είναι το προϊόν σύγκλισης των επικρατέστερων αντικειμενοστρεφών μεθοδολογιών ανάπτυξης λογισμικού.

2.4.7 Κόστος αναθεώρησης κατασκευής λογισμικού

Το κόστος της αναθεώρησης των αποφάσεων, της διόρθωσης των σφαλμάτων, ή του συνδιασμού και των δύο, είναι τόσο μεγάλο, όσο και η απαιτούμενη οπισθοδρόμηση της διαδικασίας που αυτή συνεπάγεται. Το κόστος αυτό δεν αφορά μόνο στους οικονομικούς πόρους που αποδίδονται στο έργο, αλλά και στον χρόνο καθυστέρησής του, που δεν είναι πάντα διαθέσιμος σε πραγματικές συνθήκες. Συχνό φαινόμενο δε αποτελούν οι παρενέργειες στο υπόλοιπο σύστημα λογισμικού, οι οποίες μπορούν να μεταβάλλουν προς το χειρότερο τα ποιοτικά του χαρακτηριστικά και δεν είναι εύκολο να εντοπιστούν από την αρχή.

ΚΕΦΑΛΑΙΟ 3ο

3.1 Μικρή ιστορική αναδρομή των τεχνικών τεκμηρίωσης ΠΣ

Το 1921 ο Frank Gilbreth, μέλος των μηχανολόγων μηχανικών της αμερικανικής κοινότητας (γνωστή και ως ASME), παρουσίασε την πρώτη δομημένη μέθοδο ανάλυσης της ροής μιας διαδικασίας στα πλαίσια της έρευνας που διεξάχθηκε για τα πρώτα βήματα στην εύρεση καλύτερης λύσης με τη χρήση των διαγραμμάτων ροής.

Ο μηχανικός Allan H. Mogensen, σε συνέδρια με θεματολογία την απλοποίηση των εργασιών, που διεξάχθηκαν στο Lake Placid της Νέας Υόρκης στις αρχές της δεκαετίας του '30, άρχισε να διδάσκει σε επιχειρηματίες τη χρήση κάποιων εργαλείων της βιομηχανικής μηχανικής.

Ο Art Spinanger, απόφοιτος της τάξης του Mogensen το 1944, υιοθέτησε αυτά τα εργαλεία στην Procter and Gamble, όπου ανέπτυξε το Deliberate Methods Change Program. Ένας άλλος απόφοιτος και πλέον διευθυντής του Formcraft Engineering at Standard Register Corporation ονόματι Ben S. Graham, για να δείξει πολλαπλά δεδομένα και τις μεταξύ τους σχέσεις, προσαρμόσε το διάγραμμα ροής διαδικασιών στην επεξεργασία πληροφοριών με την ανάπτυξη του διαγράμματος πολλαπλών ροών. Στη συνέχεια, η ASME βελτίωσε ένα σύμβολο το οποίο υπήρχε στο αρχικό έργο του Gilbreth, το ASME Standard for Process Charts των Mishad, Ramsan και Raiaan, το 1947.

Ο Daglas Hartree, με τη προχωρημένη για την εποχή του έκθεση η οποία εγκρίθηκε από τους μηχανικούς της IBM, εξήγησε πως το διάγραμμα ροής των Herman Goldstine και John Von Neuman, αρχικά επονομαζόμενο ως "diagram", μπορεί να σχεδιάσει προγράμματα για υπολογιστές. Το αρχικό διάγραμμα ροής των Goldstine και Von Neuman εντοπίζεται στην δημοσίευτη αναφορά τους με τίτλο "Σχεδιάζοντας και κωδικοποιώντας προβλήματα για ένα ηλεκτρονικό υπολογιστικό όργανο", Μέρος II, Τόμος I, του 1947. Πλέον λοιπόν το διάγραμμα ροής αναγνωρίζεται και χρησιμοποιείται διαδομένα ως ένα μέσο για την περιγραφή υπολογιστικών αλγορίθμων. Επεκτάσεις αυτού του διαγράμματος μπορούν να θεωρηθούν, μεταξύ άλλων, τα UML Activity Diagrams που χαρακτηρίζονταν ως μοντέρνες για την εποχή τους τεχνικές.

Στη συνέχεια, κατά τη διάρκεια της δεκαετίας του '70 όπου σύνθηρες φαινόμενο έγινε η χρήση των διαδραστικών computer terminals και των γλωσσών προγραμματισμού τρίτης γενιάς ως εργαλεία για τον προγραμματισμό των υπολογιστών, παρουσιάστηκε μείωση στη δημοτικότητα των διαγραμμάτων ροής. Οι

αλγόριθμοι μπορούν να εκφραστούν πιο συνοπτικά και ευανάγνωστα ως "κώδικας πηγής" σε τέτοια γλώσσα, ενώ συχνά χρησιμοποιούνται ψευδοκώδικες, που χρησιμοποιούν τους κοινούς ιδιωτισμούς τέτοιων γλωσσών χωρίς να εισχωρούν σε λεπτομέρειες.

3.2 Διαγράμματα ροής

Τα **διαγράμματα ροής** ή **flow charts** αποτελούνται από γεωμετρικά σύμβολα μέσω των οποίων αποτυπώνεται η ροή των δεδομένων, της εκτέλεσης των διαδικασιών αλλά και η αλληλουχία των λειτουργιών κάθε υποσυστήματος. Το διάγραμμα ροής προγράμματος είναι ένα κοινού τύπου διάγραμμα το οποίο αναπαριστά έναν αλγόριθμο ή μια διαδικασία απεικονίζοντας τα βήματα ως κουτιά διαφόρων ειδών τα οποία συνδέονται μεταξύ τους με βέλη. Το διάγραμμα ροής είναι ένα σχέδιο για την λύση ενός προβλήματος και αποτελείται από καθορισμένα βήματα. Αποτελεί ένα σχέδιο που υλοποιεί έναν αλγόριθμο επίλυσης ενός προβλήματος.

Σύμφωνα με τον Donald Knuth ένας αλγόριθμος, δηλαδή ο επιλύτης ενός προβλήματος, φέρει τα παρακάτω χαρακτηριστικά:

- Ακρίβεια: Τα βήματα πρέπει να είναι σαφή.
- Μοναδικότητα: Για κάθε είσοδο τα ενδιάμεσα αποτελέσματα είναι μοναδικά.
- Αριθμό βημάτων: Μετά από πεπερασμένο αριθμό βημάτων πρέπει ο αλγόριθμος κάποια στιγμή να τελειώσει.
- Γενικότητα: Ο αλγόριθμος πρέπει να λειτουργεί για τις εισόδους ενός συγκεκριμένου τύπου.
- Είσοδος-Έξοδος: Ο αλγόριθμος πρέπει να παίρνει κάποια είσοδο και να παράγει μια μοναδική έξοδο για κάθε είσοδο.

3.2.1 Είδη συμβόλων

Βάση παλαιότερων εγχειριδίων της επιστήμης των υπολογιστών τα είδη των συμβόλων που περιλαμβάνονται σε ένα τυπικό διάγραμμα ροής ταξινομούνται ανάλογα με τη χρήση τους στις παρακάτω κατηγορίες:

- Στα σύμβολα έναρξης και λήξης, τα οποία αναπαριστώνται ως κύκλοι, οβάλ ή στρογγυλεμένα ορθογώνια παραλληλόγραμμα και περιέχουν είτε μια λέξη που είναι έναρξη ή λήξη, είτε μια φράση που δείχνει την αρχή ή το τέλος της διαδικασίας.
- Στα βέλη, τα οποία δείχνουν την ροή του ελέγχου, αφού ένα βέλος που έρχεται από ένα σύμβολο και καταλήγει σε ένα άλλο δείχνει ότι ο έλεγχος ακολουθεί την ίδια πορεία.
- Στα στάδια της επεξεργασίας, όπου αναπαριστώνται από τα ορθογώνια





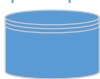













παραλληλόγραμμα.

- Στα σύμβολα εισόδου - εξόδου, που παρουσιάζονται ως παραλληλόγραμμα.
- Και σε αυτά των υποθέσεων ή αλλιώς αποφάσεων, τα οποία αναπαριστώνται από ρόμβους και τυπικά περιέχουν την ερώτηση ναι-όχι ή αληθές-ψευδές. Η ερώτηση αυτή αποτυπώνεται με δύο εξερχόμενα βέλη τα οποία πρέπει να ονοματίζονται. Το ένα βέλος εξέρχεται από την πλαϊνή μύτη του ρόμβου και αντιστοιχεί συνήθως στο όχι-ψευδές, ενώ το άλλο βέλος εξέρχεται από την κάτω μύτη του ρόμβου και αντιστοιχεί στο ναι-αληθές. Περισσότερα από δύο βέλη χρησιμοποιούνται στην περίπτωση που θέλουμε να αποτυπώσουμε μια σύνθετη απόφαση, παρόλο που όταν βρεθούμε σε αυτή τη θέση προβαίνουμε συνήθως είτε σε περαιτέρω ανάλυση, είτε σε αντικατάσταση του συμβόλου από αυτό της προκαθορισμένης διαδικασίας. Η ικανότητα της αποτύπωσης της απόφασης σε ένα διάγραμμα ροής είναι απαραίτητη.

Λιγότερο διεθνώς διαδεδομένα σύμβολα αποτελούν:

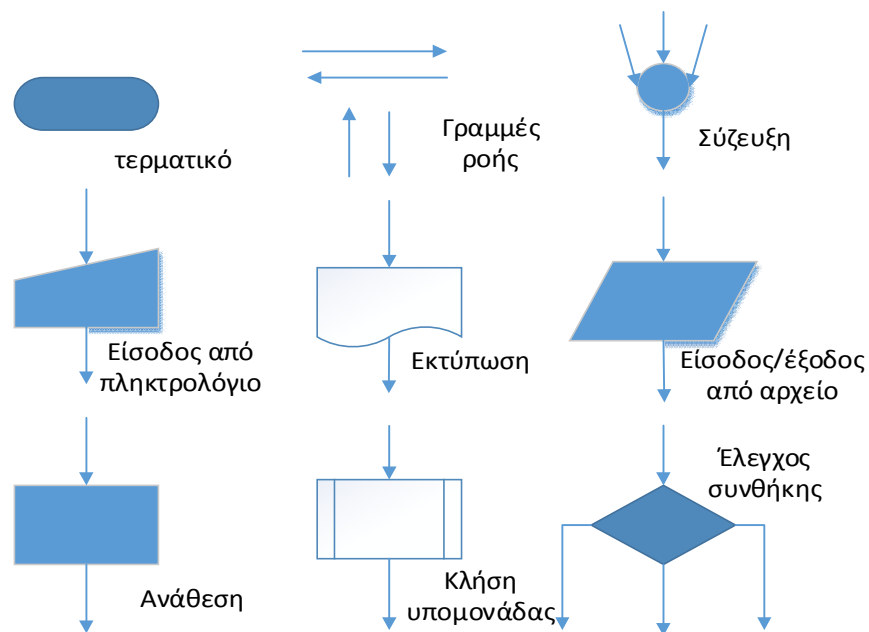
- Το σύμβολο του εγγράφου, το οποίο αναπαριστάται από ένα ορθογώνιο παραλληλεπίπεδο με κυματιστή βάση.
- Το σύμβολο της χειροκίνητης εισόδου, που αναπαριστάται από ένα παραλληλόγραμμα με το πάνω μέρος να γέρνει δεξιά και αριστερά.
- Το σύμβολο της χειροκίνητης λειτουργίας, το οποίο αναπαριστάται από ένα τραπέζιο με τη μεγαλύτερη παράλληλη πλευρά στην κορυφή και δείχνει μια διαδικασία που μπορεί να γίνει μόνο με το χέρι.
- Και το σύμβολο του αρχείου δεδομένων, που αναπαριστάται από τον κύλινδρο.

ΣΥΜΒΟΛΑ ΔΙΑΓΡΑΜΜΑΤΩΝ ΡΟΗΣ

Βασικά	Εξειδικευμένα Εισόδου / Εξόδου	Εξειδικευμένα Επεξεργασίας
Επεξεργασία 	Έγγραφο 	Απόφαση 
Είσοδος / Έξοδος 	Αποθήκευση σε δίσκο 	Χειροκίνητη λειτουργία 
Γραμμή Ροής 	On line Αποθήκευση 	Προκαθορισμένη επεξεργασία 
Σχολιασμός 	Διάτρητη ταινία 	Τερματισμός 
	Αποθήκευση σε μαγνητική ταινία 	Σύνδεση στην ίδια σελίδα 
	Χειροκίνητη είσοδος 	Σύνδεση με άλλη σελίδα 
		Αρχειοθέτηση 
		Ταξινόμηση 

Πίνακας 3.2.1.1 Απεικόνιση συμβόλων στα διαγράμματα ροής

Υπάρχουν και επιπλέον σύμβολα που χρησιμοποιούνται για την κατασκευή των διαγραμμάτων ροής. Για την αποτύπωση των συνδέσμων μεταξύ διαδρομών προς σύγκλιση χρησιμοποιείται το γεωμετρικό σχήμα του κύκλου, ο οποίος μπορεί να δεχτεί παραπάνω από ένα εισερχόμενα βέλη, μα παραμόνο ένα εξερχόμενο. Για την αποτύπωση μιας διαδικασίας η οποία επαναλαμβάνεται (loop) χρησιμοποιείται ένα βέλος το οποίο καταλήγει σε ένα άλλο, παραδείγματος χάριν ξεκινάμε με τον έλεγχο που εισέρχεται σε ένα σύνδεσμο, συνεχίζουμε με τα στάδια της επεξεργασίας, έπειτα ακολουθεί μια απόφαση που αποτυπώνεται με το σύμβολο του ρόμβου και με δύο βέλη που εξέρχονται από αυτόν, το ένα εκ των οποίων επιστρέφει στον αρχικό σύνδεσμο. Για τη διαγραμματική αποτύπωση της σύνδεσης μιας διαδικασίας που συνεχίζεται είτε σε άλλη οθόνη είτε σε άλλη σελίδα, χρησιμοποιούνται οι σύνδεσμοι εκτός σελίδας. Σημαντικότητα ενέχει η διατήρηση μιας λογικής σειράς στις συνδέσεις αυτές αλλά και ο κανόνας που θέλει στο σύνολό τους τις διαδικασίες να αρχίζουν από τα δεξιά προς τα αριστερά και από την κορυφή προς το τέλος.



Πίνακας 3.2.1.2 Απεικόνιση συμβόλων στα διαγράμματα ροής

Για την σχεδίαση ενός διαγράμματος ροής μπορεί να χρησιμοποιηθεί ένα από τα λογισμικά Visio, OmniGraffle, Koffice-kivio, μεταξύ άλλων.

3.2.2 Τύποι διαγραμμάτων ροής

Σύμφωνα με τον Sterneckert τα διαγράμματα ροής μπορούν να μοντελοποιηθούν απο την σκοπιά των διαφορετικών ομάδων χρηστών, έτσι αναλύονται σε τέσσερις γενικούς τύπους, ως εξής:

- Διάγραμμα ροής εγγράφων: Ρύθμιση στην ροή εγγράφων μέσα σε ένα σύστημα
- Διάγραμμα ροής δεδομένων: Ρύθμιση στην ροή δεδομένων μέσα σε ένα σύστημα
- Διάγραμμα ροής συστημάτων: Ρύθμιση στην ροή των πόρων μέσα σε ένα σύστημα
- Διάγραμμα ροής προγραμμάτων: Ρύθμιση στα προγράμματα μέσα σε ένα σύστημα.

Κάθε τύπος διαγράμματος αποβλέπει σε κάποιο είδος ελέγχου, παρά σε μια συγκεκριμένη ροή. Ωστόσο υπάρχουν διάφορες ταξινομήσεις, ενώ σύμφωνα με τον Andrew Veronis το 1975 ονομάστηκαν τρεις βασικοί τύποι διαγραμμάτων, τα διαγράμματα ροής συστημάτων, τα γενικά διαγράμματα και τα αναλυτικά διαγράμματα ροής. Την ίδια χρονιά ο Marilyn Bohl υποστήριξε ότι στην πράξη για τον σχεδιασμό λύσεων απαιτούνται δύο είδη διαγραμμάτων, τα διαγράμματα συστημάτων και τα διαγράμματα προγραμμάτων. Σε όλους τους τύπους διαγραμμάτων ο συμβολισμός παραμένει ίδιος με αυτόν που έχει ήδη αναφερθεί.

Διαγράμματα ροής συστημάτων

Τα διαγράμματα ροής συστημάτων χρησιμοποιούνται για την ανάλυση ενός λογιστικού πληροφοριακού συστήματος. Περιγράφουν ολόκληρο το σύστημα και τις λειτουργίες του, τις εισόδους και τις εξόδους. Για τον λόγο αυτόν δίνεται έμφαση στην παράθεση των διαφόρων λειτουργικών μονάδων παρά στην λεπτομερή περιγραφή της κάθε μιας. Χρήστες των διαγραμμάτων αυτών είναι το προσωπικό ανάπτυξης συστημάτων και οι ελεγκτές τους.

Διαγράμματα ροής προγράμματος

Τα διαγράμματα ροής προγράμματος χρησιμοποιούνται για την πιο λεπτομερή επεξήγηση των λειτουργικών μονάδων ενός διαγράμματος ροής συστήματος, κατά τον σχεδιασμό ενός λογιστικού πληροφοριακού συστήματος. Λόγο του βαθμού λεπτομέρειας που προσφέρουν, το προσωπικό ανάπτυξης συστημάτων μπορεί να δημιουργήσει τον τελικό κώδικα, το πρόγραμμα δηλαδή του συστήματος.

Αναλυτικά διαγράμματα ροής

Τα αναλυτικά διαγράμματα ροής χρησιμοποιούνται για την ανάλυση ενός λογιστικού πληροφοριακού συστήματος, με τρόπο παρόμοιο των διαγραμμάτων ροής συστημάτων ως προς τον βαθμό λεπτομέρειας. Επιπλέον αποτυπώνουν τους συμμετέχοντες και τις δραστηριότητες αυτών, χωρίζοντας τα διαγράμματα σε στήλες

δραστηριοτήτων ανά συμμετέχοντα. Χρήστες των αναλυτικών διαγραμμάτων ροής είναι το προσωπικό ανάπτυξης συστημάτων και οι ελεγκτές τους, όπως και στα διαγράμματα ροής συστημάτων.

Διαγράμματα ροής εγγράφων

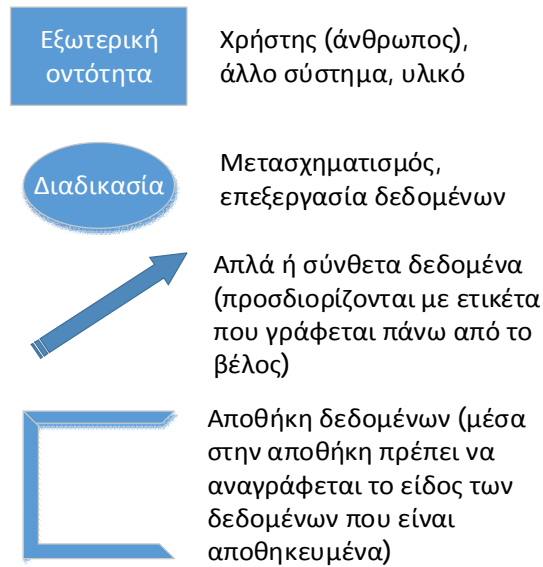
Τα διαγράμματα ροής εγγράφων περιγράφουν τη δημιουργία, τη δρομολόγηση, την επεξεργασία και τον τελικό προορισμό κάθε εγγράφου που χρησιμοποιείται. Παρόλο που έχουν την ίδια μορφή με τα αναλυτικά διαγράμματα ροής, εστιάζουν στη δημιουργία και τη διακίνηση των εγγράφων παρά στην περιγραφή των λειτουργιών του συστήματος, συνεπώς εμφανίζεται κυρίως το σύμβολο του εγγράφου και μόνο όταν είναι απαραίτητο κάποιο άλλο από τα γνωστά σύμβολα. Τα διαγράμματα ροής εγγράφων είναι εξαιρετικά χρήσιμα για την ανάπτυξη συστημάτων διαχείρισης εγγράφων και για τους ελεγκτές.

3.3 Διαγράμματα ροής δεδομένων

Για την αναπαράσταση των λειτουργιών των διαφόρων υποσυστημάτων και της ροής των πληροφοριών μεταξύ των διαδικασιών τους χρησιμοποιούνται τα διαγράμματα ροής δεδομένων ή αλλιώς ΔΡΔ (data flow diagrams or DFD). Τα ΔΡΔ αποτελούν τη βασική διαγραμματική τεχνική της δομημένης ανάλυσης και σχεδίασης (Structured Analysis and Design– SADT). Χρησιμοποιούνται για να απεικονίσουν τη ροή των δεδομένων μέσα σε ένα πληροφοριακό σύστημα και είναι ιδιαίτερα χρήσιμα για την περιγραφή και αναγνώριση των λειτουργιών και των διαδοχικών μετασχηματισμών που υφίστανται τα δεδομένα που κυκλοφορούν σε αυτό. Τα ΔΡΔ ανήκουν στις διαγραμματικές τεχνικές λογικής γενικής περιγραφής.

Τα βασικά στοιχεία ενός ΔΡΔ είναι τα εξής:

- Οι πηγές παροχής των δεδομένων και οι προορισμοί τους, που συμβολίζονται με ορθογώνια παραλληλόγραμμα.
- Οι ροές δεδομένων, που αποτελούν τα κανάλια στα οποία ρέουν τα δεδομένα και συμβολίζονται με βέλη τα οποία δείχνουν την κατεύθυνση της πληροφορίας.
- Η επεξεργασία των δεδομένων που αντιστοιχεί σε μετασχηματισμούς δεδομένων και συμβολίζεται με ελλείψεις.
- Οι αποθήκες δεδομένων που αποτελούν τους αποθηκευτικούς χώρους του ΠΣ, δηλαδή τα αρχεία αποθήκευσης των δεδομένων, που συμβολίζονται με ορθογώνια παραλληλόγραμμα των οποίων η δεξιά κάθετη πλευρά είναι ανοικτή.



Σχήμα 3.3.1 Απεικόνιση των βασικών στοιχείων ενός ΔΡΔ

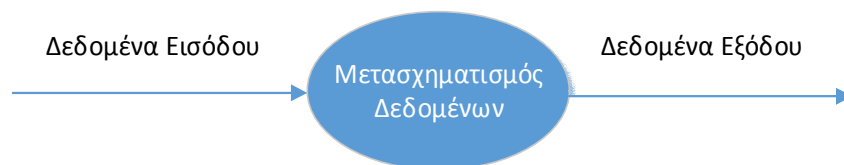
Γενικά, η πληροφορία μετασχηματίζεται καθώς ρέει μέσα από το πληροφοριακό σύστημα. Η δομημένη ανάλυση και σχεδίαση μοντελοποιεί την ροή των δεδομένων χρησιμοποιώντας τα ΔΡΔ. Αρχικά, η συνολική λειτουργία του συστήματος αναπαριστάται με ένα μοναδικό μετασχηματισμό πληροφορίας, μία διαδικασία για όλο το πληροφοριακό σύστημα. Η αναπαράσταση αυτή αναφέρεται ως ανάλυση μηδενικού επιπέδου ή επιπέδου 0, ενώ το ΔΡΔ που χρησιμοποιείται για την αναπαράσταση του συστήματος αναφέρεται ως ΔΡΔ επιπέδου 0 ή πρωταρχικό μοντέλο (Context diagram). Οι εισοδοί αυτής της διαδικασίας προέρχονται από μία ή περισσότερες εξωτερικές οντότητες. Ο μετασχηματισμός παράγει εξόδους που διοχετεύονται σε άλλες εξωτερικές οντότητες.



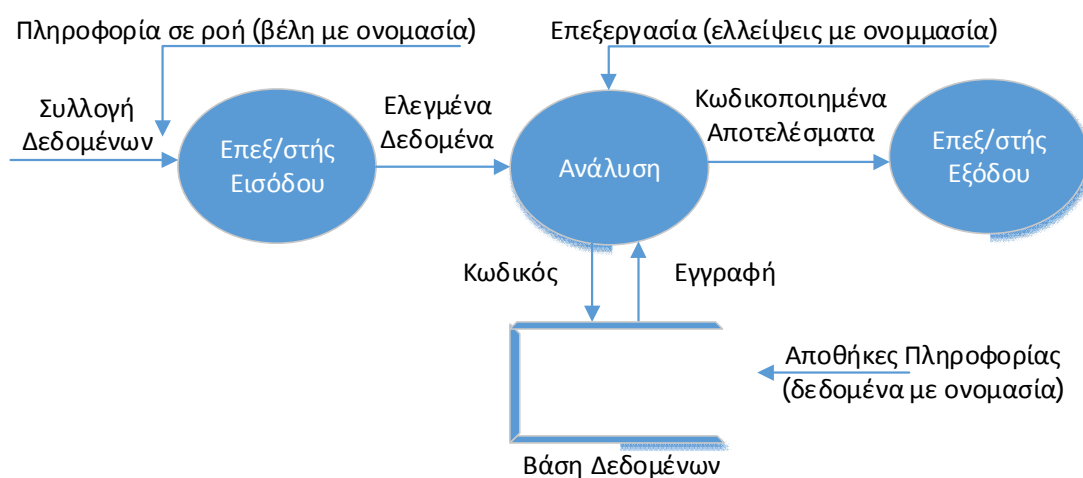
Σχήμα 3.3.2 Γενικό παράδειγμα ενός ΔΡΔ επιπέδου 0

Ένα ΔΡΔ δεν είναι ένα διάγραμμα ροής ελέγχου. Το ΔΡΔ αναφέρεται και περιγράφει τη ροή των δεδομένων μέσα σε ένα σύστημα. Το διάγραμμα ροής ελέγχου ή διάγραμμα ροής αναφέρεται και περιγράφει τη ροή του ελέγχου σε έναν αλγόριθμο επίλυσης ενός προβλήματος. Κατά τη διαδικασία ανάπτυξης του ΔΡΔ ο αναλυτής δεν πρέπει να εμπλέκεται σε λεπτομέρειες που αφορούν την ακολουθία της επεξεργασίας όπως βρόχους και αποφάσεις.

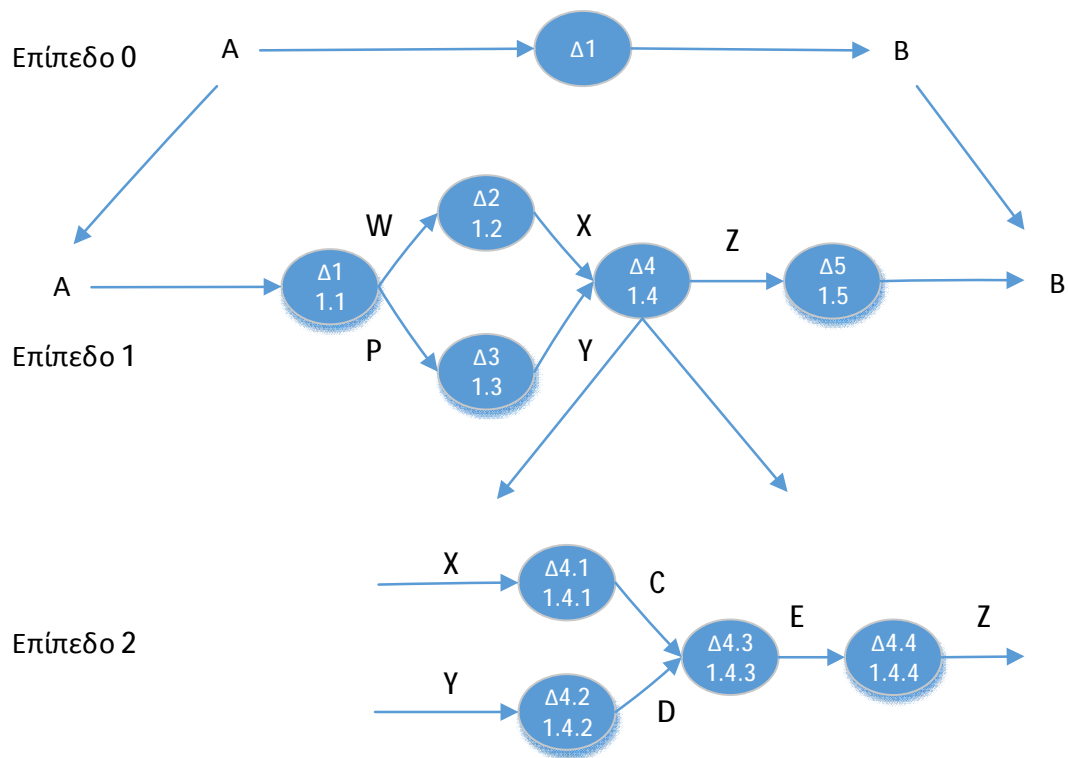
Ένα από τα πλεονεκτήματα των ΔΡΔ είναι η ιεραρχική δομή τους. Ξεκινώντας από την αρχική διαδικασία, δηλαδή τη διαδικασία του επιπέδου 0, ο αναλυτής προσπαθεί να την διασπάσει και να την αναλύσει σε άλλες απλούστερες καθώς κατεβαίνει τα επίπεδα της ανάλυσης. Επομένως, η ανάλυση ενός πληροφοριακού συστήματος με χρήση των ΔΡΔ βασίζεται σε διαδοχική ανάλυση των διαδικασιών που εκτελούνται σε αυτό. Για τη διευκόλυνση της κατανόησης ενός ΔΡΔ οι διαδικασίες αριθμούνται. Η βασική αρχή που πρέπει να ακολουθείται στη σχεδίαση ενός ΔΡΔ, είναι η διατήρηση της συνέχειας στη ροή της πληροφορίας, ενώ η είσοδος και η έξοδος σε κάθε ανάλυση πρέπει να παραμένει η ίδια.



Σχήμα 3.3.3 Γενικό διάγραμμα μιας αρχικής διαδικασίας



Σχήμα 3.3.4 Γενική ανάλυση μιας αρχικής διαδικασίας σε απλούστερες



Σχήμα 3.3.5 Γενικό παράδειγμα αρίθμησης διαδικασιών

3.3.1 Βασικές αρχές της κατασκευής ενός ΔΡΔ

Όταν κατασκευάζεται ένα ΔΡΔ για την ανάλυση και το σχεδιασμό ενός πληροφοριακού συστήματος πρέπει να λαμβάνεται υπόψη ότι αυτό πρέπει να αποτελείται από τα εξής βασικά διαγράμματα:

- Το γενικό διάγραμμα (διάγραμμα επιπέδου 0), το οποίο περιγράφει τα «όρια» του συστήματος, δηλαδή τις πηγές και προορισμούς του συστήματος και τα σχετικά εισερχόμενα και εξερχόμενα δεδομένα.
- Το διάγραμμα 1ου επιπέδου, το οποίο απεικονίζει την πρώτη γενική θεώρηση του συστήματος. Εδώ αναλύεται το όλο σύστημα στις βασικές διαδικασίες του, ενώ παράλληλα περιγράφονται οι βασικές διαδικασίες, οι ροές και αποθηκεύσεις δεδομένων.
- Το διάγραμμα κατωτέρου επιπέδου (2ου , 3ου , κλπ.) το οποίο αναπαριστά με περισσότερη λεπτομέρεια μια διαδικασία και τις σχετικές εισερχόμενες ή εξερχόμενες ροές δεδομένων που εμφανίζονται σε ένα διάγραμμα ανώτερου επιπέδου.

Η κατασκευή ενός διαγράμματος ροής δεδομένων λογίζεται ορθή όταν μέσω αυτού παρέχονται τα παρακάτω στοιχεία :

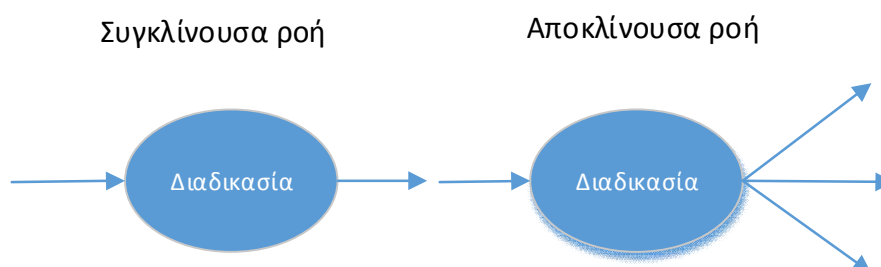
- η ανάλυση του συστήματος σε υποσυστήματα μέσω της διάσπασής του,
- οι ροές δεδομένων που πρέπει να εμπεριέχονται σε αυτό,
- τα δεδομένα όπως αυτά αποτυπώνονται ως εισερχόμενα και εξερχόμενα και οι αποθηκεύσεις αυτών,
- και φυσικά οι πηγές και οι προορισμοί του συστήματος.

Αντιθέτως, στοιχεία που δεν πρέπει να παρέχονται από έναν ΔΡΔ είναι τα εξής:

- οι ανάγκες ανάκτησης των αποθηκευμένων δεδομένων,
- η δομή των ροών δεδομένων στο σύστημα,
- οι επαναλήψεις των διαδικασιών και οι αριθμητικοί υπολογισμοί.
- οι αποφάσεις που λαμβάνονται στο σύστημα,

3.3.2 Ροές δεδομένων

Οι ροές δεδομένων (data flows) αποτελούν δίαυλους κυκλοφορίας δεδομένων γνωστού περιεχομένου. Κάθε ροή δεδομένων (βέλος) πρέπει να ονοματίζεται κατάλληλα. Οι ροές που εισέρχονται σε μία διαδικασία λέγονται εισερχόμενες ροές ενώ αυτές που εξέρχονται από μία διαδικασία αναφέρονται ως εξερχόμενες ροές δεδομένων. Οι ροές δεδομένων μπορούν να είναι είτε συγκλίνουσες είτε αποκλίνουσες.

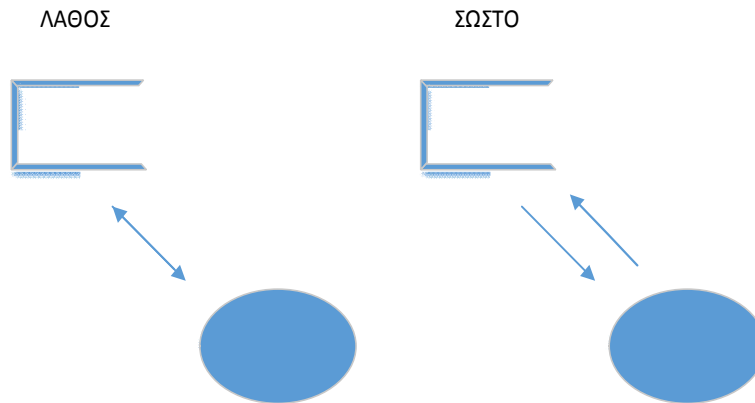


Σχήμα 3.3.2.1 Αποκλίνουσες και συγκλίνουσες ροές δεδομένων.

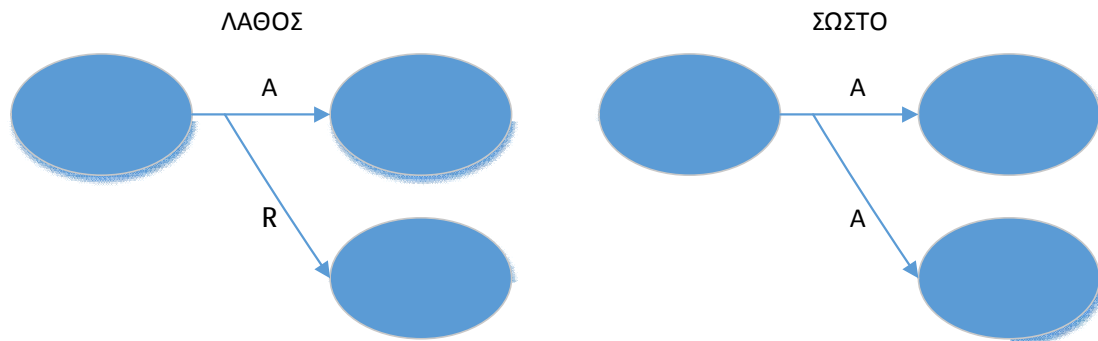
Βασικοί κανόνες σχεδίασης

Οι βασικοί κανόνες σχεδίασης των ΔΡΔ που αναφέρονται στις ροές δεδομένων είναι:

- Η μοναδικότητα κατεύθυνσης της γραμμής ροής, όπως απεικονίζεται στην παρακάτω εικόνα.



- Η ικανότητα της διάσπασης, η οποία αποτυπώνει τη μεταφορά ίδιων δεδομένων από μία πηγή σε δύο ή περισσότερους προορισμούς.



- Η ένωση δύο ή περισσότερων ροών, που δείχνει τα ίδια ακριβώς δεδομένα να μεταφέρονται από δύο ή περισσότερες πηγές σε ένα προορισμό.

- Η ροή με κατεύθυνση προς μια αποθήκη δεδομένων, που σημαίνει αποθήκευση ή ενημέρωση ή διαγραφή δεδομένων.

- Η ροή με κατεύθυνση από μια αποθήκη δεδομένων, που σημαίνει ανάγνωση ή αναζήτηση δεδομένων.

- Στην ονοματοδοσία της ροής δεδομένων πρέπει να μην περιλαμβάνεται ρήμα όταν πρόκειται για μια περιγραφή ή αυτή να ονοματίζεται με κάποιο ουσιαστικό.

Κάποια απλά παραδείγματα ροών δεδομένων είναι τα εξής:

- τα έντυπα και τα παραστατικά
- η είσοδος των στοιχείων στον ηλεκτρονικό υπολογιστή
- το περιεχόμενο της προβολής στην οθόνη του ηλεκτρονικού υπολογιστή
- και τα αρχεία των δεδομένων.

3.3.3 Διαδικασίες

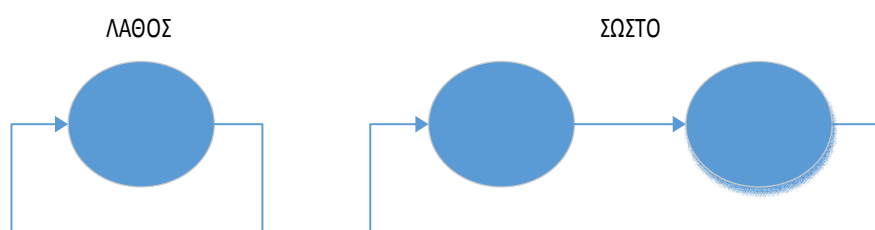
Οι διαδικασίες (processes) αποτελούν διεργασίες που πραγματοποιούνται επί των εισερχόμενων ροών δεδομένων και στοχεύουν στην παραγωγή εξερχόμενων ροών

δεδομένων. Οι διεργασίες αυτές εκτελούνται είτε από ανθρώπους, είτε από τα τεχνολογικά εργαλεία που αυτοί διαθέτουν, όπως ηλεκτρονικοί υπολογιστές και μηχανές. Οι διαδικασίες για να είναι πλήρεις πρέπει να συνοδεύονται από δύο βασικά συστατικά, από έναν αριθμό αναφοράς και ένα περιγραφικό όνομα, που περιγράφει όσο το δυνατόν καλύτερα το μετασχηματισμό που εφαρμόζεται στις εισερχόμενες ροές δεδομένων για να προκύψουν οι εξερχόμενες.

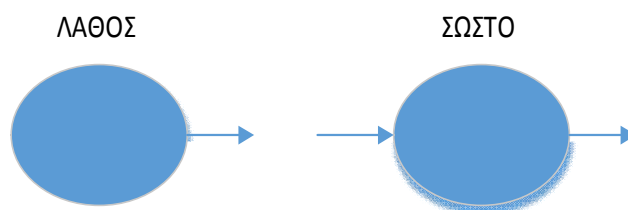
Βασικοί κανόνες σχεδίασης

Οι βασικοί κανόνες σχεδίασης των ΔΡΔ που αναφέρονται στις διαδικασίες είναι:

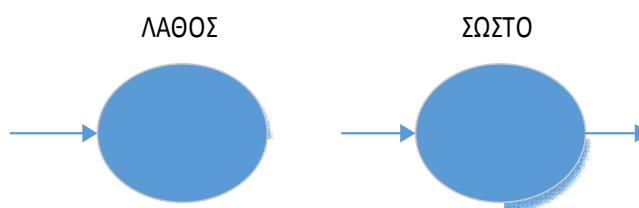
- Οι είσοδοι μιας διαδικασίας πρέπει να αρκούν για την παραγωγή των εξόδων της.
- Οι είσοδοι μιας διαδικασίας πρέπει να είναι διαφορετικές από τις εξόδους της.



- Δεν επιτρέπεται να υπάρχει διαδικασία που να έχει μόνο εξόδους.



- Δεν επιτρέπεται να υπάρχει διαδικασία που να έχει μόνο εισόδους.



- Το περιγραφικό όνομα μιας διαδικασίας πρέπει να περιέχει κάποιο ρήμα που να δηλώνει της ενέργεια ή λειτουργία που επιτελείται.

Κάποια απλά παραδείγματα διαδικασιών είναι τα εξής:

- Η εκτέλεση υπολογισμών
- Ο διαχωρισμός εισερχόμενης ροής δεδομένων σε δύο ή περισσότερες εξερχόμενες, ή ο συνδυασμός δύο ή περισσότερων εισερχομένων ροών δεδομένων σε λιγότερες εξερχόμενες.

- Η αναδιοργάνωση (ταξινόμηση, μορφοποίηση, κλπ) εισερχομένων ροών δεδομένων.
- Η επαλήθευση μιας ροής δεδομένων σε σχέση με μια άλλη (μη τροποποίηση του περιεχομένου).

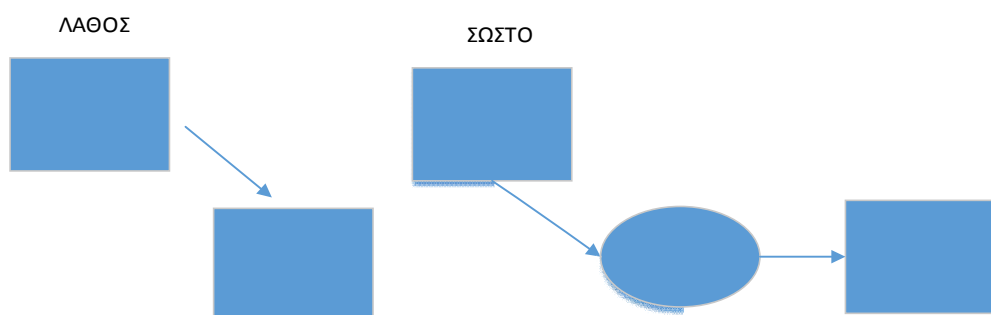
3.3.4 Εξωτερικές οντότητες, πηγές και προορισμοί δεδομένων

Εξωτερικές οντότητες (external entities) ονομάζονται οι πηγές ή οι προορισμοί των δεδομένων, οι οποίες αλληλεπιδρούν με το σύστημα λαμβάνοντας δεδομένα από αυτό ή παρέχοντας δεδομένα σε αυτό, χωρίς όμως να βρίσκονται μέσα στο υπό μελέτη σύστημα.

Βασικοί κανόνες σχεδίασης

Οι βασικοί κανόνες σχεδίασης των ΔΡΔ που αναφέρονται στις εξωτερικές οντότητες είναι οι εξής:

- Δεν επιτρέπεται να μετακινούνται δεδομένα από εξωτερική πηγή σε εξωτερική πηγή, παρά μόνο με παρεμβολή μιας ή περισσότερων διαδικασιών.



- Μια εξωτερικής πηγή πρέπει να ονομάζεται με τη χρήση κάποιου ουσιαστικού ή μιας περιγραφής η οποία δεν περιλαμβάνει ρήμα.

Κάποια απλά παραδείγματα εξωτερικών οντοτήτων είναι τα εξής:

- Απασχολούμενοι της επιχείρησης εκτός εμβέλειας του συστήματος.
- Η πρωτογενής πηγή μιας δοσοληψίας εισόδου.
- Ένα πληροφοριακό σύστημα που αλληλεπιδρά με το υπό μελέτη σύστημα.
- Ένας οργανισμός ή μια υπηρεσία εκτός επιχείρησης.
- Ή και ένα τμήμα της επιχείρησης που είναι εκτός του συστήματος που σχεδιάζεται ή μελετάται.

3.3.5 Αποθήκες δεδομένων

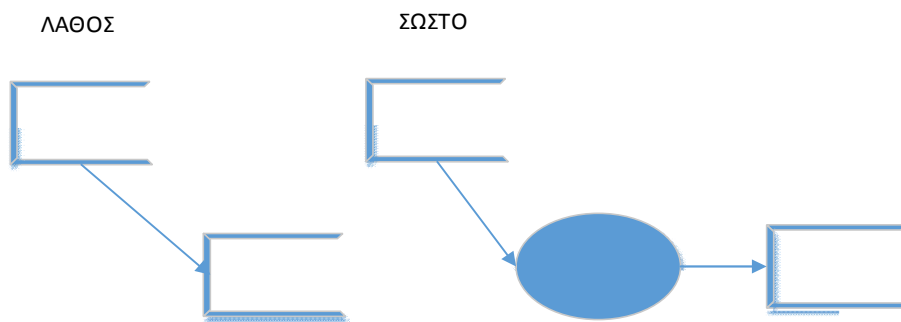
Οι αποθήκες δεδομένων (data stores) αποτελούν τους χώρους αποθήκευσης του πληροφοριακού συστήματος και έχουν δύο ιδιαίτερα χαρακτηριστικά. Αρχικά το ότι

μπορούν να επικοινωνούν μόνο με διεργασίες (παρεμβολή ανθρώπου ή Η/Υ), ενώ η διεύθυνση μιας ροής δεδομένων προς μια αποθήκη δεδομένων έχει ανάλογη ερμηνεία (ανάγνωση και χρήση δεδομένων έναντι προσθήκης, διαγραφής, και τροποποίησής τους.)

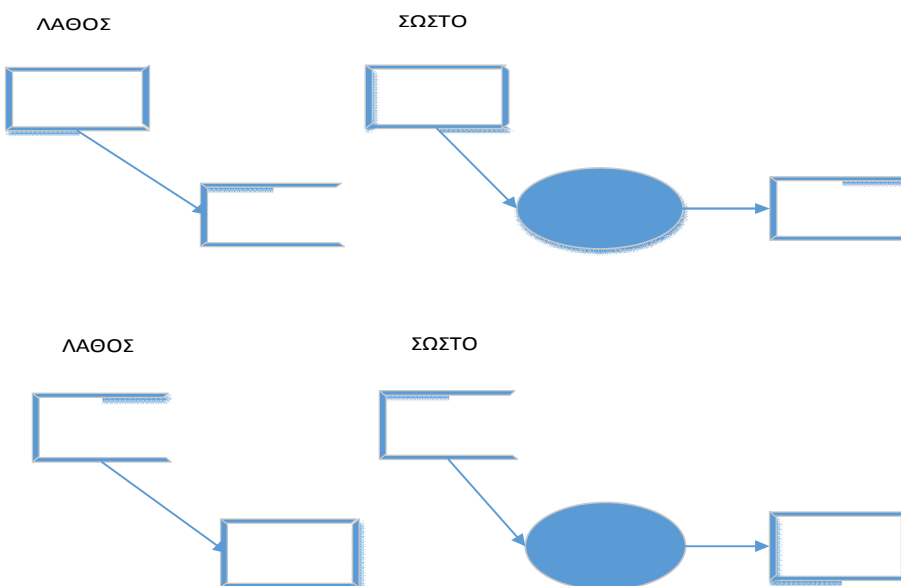
Βασικοί κανόνες σχεδίασης

Οι βασικοί κανόνες σχεδίασης των ΔΡΔ που αναφέρονται στις αποθήκες δεδομένων είναι οι εξής:

- Δεν επιτρέπεται να μετακινούνται δεδομένα από αποθήκη σε αποθήκη, παρά μόνο με την παρεμβολή μίας ή περισσότερων διαδικασιών.



- Δεν επιτρέπεται να μετακινούνται δεδομένα μεταξύ μιας εξωτερικής πηγής και μιας αποθήκης δεδομένων, παρά μόνο με την παρεμβολή μίας ή περισσότερων διαδικασιών.



- Για την ονοματοδοσία της αποθήκης δεδομένων μπορεί να χρησιμοποιηθεί κάποιο

ουσιαστικό ή μια περιγραφή η οποία όμως δεν πρέπει να περιλαμβάνει ρήμα.

- Οι αποθήκες δεν επεξεργάζονται δεδομένα. Από αυτές λαμβάνονται ακριβώς τα ίδια δεδομένα με αυτά που εισήχθησαν σε αυτές.
- Δεν θα πρέπει να υπάρχουν στο σύστημα αποθήκες με μόνο εισερχόμενες ροές δεδομένων. Ομοίως, δεν θα πρέπει να υπάρχουν αποθήκες με μόνο εξερχόμενες ροές δεδομένων.
- Η ίδια αποθήκη μπορεί να υπάρχει σε διάφορα σημεία του ΔΡΔ.
- Μία αποθήκη δεδομένων μπορεί να είναι ένας απλός πίνακας καθώς και σύνθετες συλλογές δεδομένων.

Κάποια απλά παραδείγματα εξωτερικών οντοτήτων είναι τα εξής:

- Βάση δεδομένων ή αρχεία H/Y
- Ευρετήριο με καρτέλες
- Λογιστικά βιβλία

Οι βασικές κατευθυντήριες γραμμές τις οποίες πρέπει να ακολουθεί κανείς όταν θέλει να μοντελοποιήσει ένα πληροφοριακό σύστημα χρησιμοποιώντας ένα ΔΡΔ είναι οι εξής:

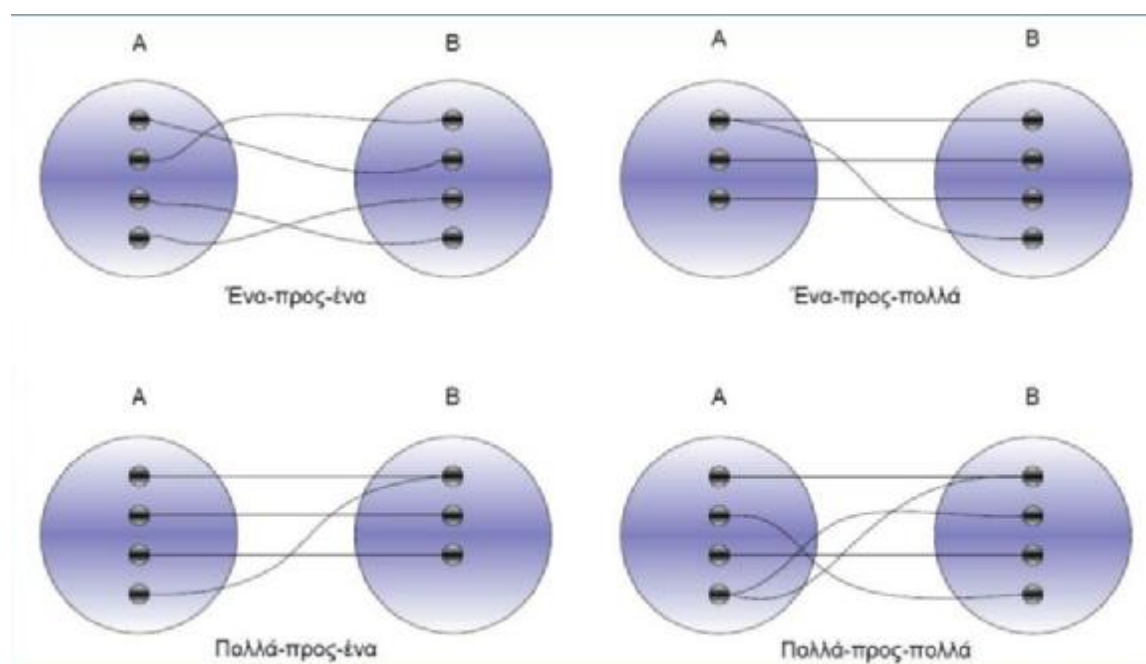
- Το ΔΡΔ μηδενικού επιπέδου απεικονίζει όλο το πληροφοριακό σύστημα σαν μία διεργασία.
- Πρέπει να ονοματιστούν όλες οι ροές δεδομένων και οι διαδικασίες.
- Μία διαδικασία δεν θα πρέπει να έχει όμοιες εισόδους και εξόδους.
- Οι εξωτερικές οντότητες δεν είναι διαδικασίες.
- Πρέπει να διατηρείται η συνέχεια της ροής της πληροφορίας.
- Πρέπει να γίνεται η εκλέπτυνση μίας διαδικασίας τη φορά.
- Σταδιακή εκλέπτυνση (όχι πολύ λεπτομέρεια πολύ νωρίς).
- Αποφυγή περιγραφής ροής ελέγχου.
- Δεν υπάρχει χρονισμός σε ένα ΔΡΔ.
- Οι είσοδοι πρέπει να είναι σαφείς σε κάθε μετασχηματισμό.
- Καθαρή απεικόνιση του ΔΡΔ (καλό σχέδιο).
- Οργάνωση του ΔΡΔ από αριστερά προς τα δεξιά.

3.4 Διαγράμματα οντοτήτων συσχετίσεων

Για τις βάσεις δεδομένων δημιουργούνται τα διαγράμματα οντοτήτων συσχετίσεων ή αλλιώς ΔΟΣ. Το μοντέλο οντοτήτων συσχετίσεων στηρίζεται στην θεωρία ότι ο πραγματικός κόσμος αποτελείται από οντότητες (entities) που έχουν κάποια

γνωρίσματα (attributes) και συνδέονται/συσχετίζονται (relationships) μεταξύ τους. Είναι ένα μοντέλο δεδομένων με συγκεκριμένη δομή μέσω του οποίου δημιουργούνται τα διαγράμματα οντοτήτων συσχετίσεων. Το 1976 έγινε η αρχή από τον Peter Chen και ακολούθησαν άλλες εκδοχές της διαδικασίας.

Τα στοιχεία που περιλαμβάνονται είναι η οντότητα, δηλαδή ένα πρόσωπο ή ένα γεγονός για το οποίο πραγματοποιείται συλλογή δεδομένων. Ακολουθούν τα χαρακτηριστικά γνωρίσματα της οντότητας, τα οποία περιγράφουν την οντότητα και διαφοροποιούν τα στοιχεία της και χωρίζονται σε απλά/σύνθετα, μονότιμα/πλειότιμα και αποθηκευμένα/παραγόμενα. Έπειτα έχουμε τη συσχέτιση των οντοτήτων μεταξύ τους και τον τρόπο που αυτή πραγματοποιείται στον πραγματικό κόσμο, καθώς και ο τρόπος σύνδεσης τους που χωρίζεται σε σχέση ένα προς ένα (1:1), ένα προς πολλά (1 :N) και πολλά προς πολλά (M :N). Τέλος, έχουμε το κύριο κλειδί της οντότητας, το οποίο είναι η ταυτότητά της, αφού την ξεχωρίζει από όλες τις άλλες. Σε περίπτωση που ένα κύριο κλειδί υπάρχει σε μία άλλη οντότητα ως απλό γνώρισμα τότε ονομάζεται ξένο κλειδί.

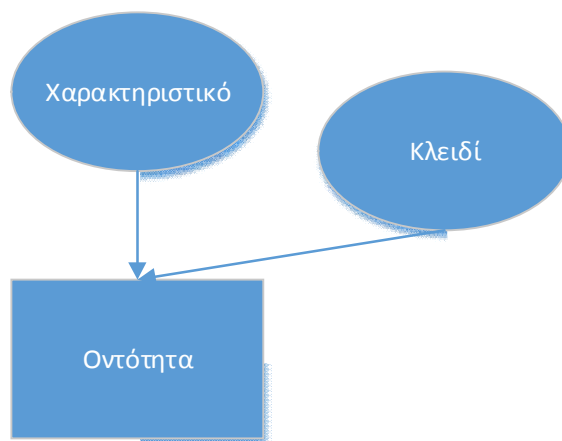


Σχήμα 3.4.1 Απεικόνιση συσχετίσεων οντοτήτων

Για την απεικόνιση των σχέσεων που αναπτύσσονται μεταξύ των οντοτήτων χρησιμοποιούμε τα διαγράμματα οντοτήτων συσχετίσεων καθώς έτσι παρέχετε μια κατανοητή περιγραφή της δομής της βάσης δεδομένων.

Τα σχήματα που κυρίως χρησιμοποιούνται είναι, το ορθογώνιο για την απεικόνιση της οντότητας, ο ρόμβος για την συσχέτιση, η έλλειψη για τα γνωρίσματα

και το ευθύγραμμο τμήμα για την σύνδεση οντοτήτων. Όταν ένα γνώρισμα είναι το κύριο κλειδί τότε υπογραμμίζεται.

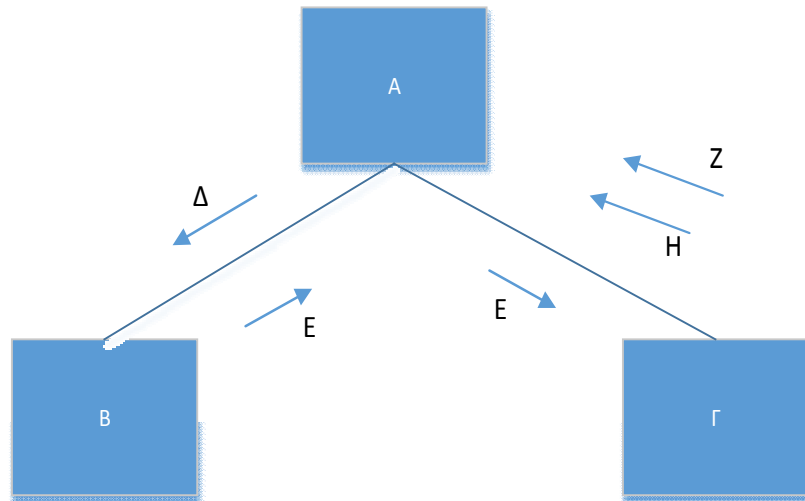


Σχήμα 3.4.1 Απεικόνιση σχημάτων προς χρήση στα διαγράμματα οντοτήτων συσχετίσεων

3.5 Διαγράμματα δομής

Τα διαγράμματα δομής ανήκουν στις διαγραμματικές τεχνικές γενικής περιγραφής και μαζί με τα διαγράμματα ροής δεδομένων αποτελούν τις πιο διαδεδομένες διαγραμματικές τεχνικές της κατηγορίας αυτής. Το βασικό δομικό συστατικό ενός συστήματος είναι η ενότητα (module). Τα δομημένα προγράμματα είναι στην πραγματικότητα οργανωμένα ως μια ιεραρχία ενότητων. Ένα διάγραμμα δομής που χρησιμοποιείται για την περιγραφή ενός συστήματος-προγράμματος, αποτελεί ένα ιεραρχικό διάγραμμα που καθορίζει τη συνολική αρχιτεκτονική δομή του, παρουσιάζοντας τόσο τις ενότητες από τις οποίες αποτελείται το σύστημα-πρόγραμμα όσο και τις μεταξύ τους σχέσεις.

Τα διαγράμματα δομής αποτελούνται από ορθογώνια παραλληλόγραμμα και γραμμές με βέλη τα οποία ενώνουν τα ορθογώνια αυτά μεταξύ τους. Κάθε ορθογώνιο παραλληλόγραμμα παριστάνει μια ενότητα του συστήματος-προγράμματος το όνομα της οποίας αναγράφεται μέσα στο εσωτερικό του ορθογωνίου. Στο παρακάτω σχήμα παρουσιάζεται η γενική μορφή ενός διαγράμματος δομής.



Σχήμα 3.5.1 Η γενική μορφή ενός διαγράμματος ροής.

- Η ενότητα A μπορεί να καλέσει την ενότητα B όπως επίσης και την ενότητα Γ. Αυτό υποδεικνύεται από τις γραμμές που ενώνουν τα ορθογώνια που αντιπροσωπεύουν τις ενότητες. Το διάγραμμα δεν παρουσιάζει τη σειρά με την οποία εκτελούνται οι κλήσεις αυτές ούτε καν εάν εκτελούνται στην πραγματικότητα. Όταν ολοκληρωθεί η εκτέλεση μιας υφιστάμενης διαδικασίας ο έλεγχος επιστρέφει στη διαδικασία που την κάλεσε.

- Όταν η ενότητα A καλεί την ενότητα B, της στέλνει δεδομένα τύπου Δ. Όταν η B ολοκληρώσει την εκτέλεσή της, επιστρέφει δεδομένα τύπου E στην A. Ομοίως, η A επικοινωνεί με την Γ χρησιμοποιώντας δεδομένα τύπου E και H.

- Όταν η ενότητα Γ ολοκληρώσει τη λειτουργία της, στέλνει μια σημαία τύπου Z στην ενότητα A. Μία σημαία χρησιμοποιείται ως μία ροή δεδομένων ελέγχου. Η διαφορά στο συμβολισμό ανάμεσα στα δεδομένα και τις σημαίες είναι το ότι στα δεδομένα ο κύκλος είναι λευκός, ενώ στις σημαίες είναι μαύρος.

3.5.1 Οι κύριες έννοιες των βασικών δομικών συστατικών των ΔΔ

Σχέσεις ελέγχου

Ένα διάγραμμα δομής διατάσσει τις ενότητες του προγράμματος σε επίπεδα. Συνδέοντας τα επίπεδα μεταξύ τους με βέλη απεικονίζει τις σχέσεις ελέγχου ανάμεσα στις ενότητες κάθε επιπέδου. Η φορά του βέλους που ενώνει δύο ενότητες καθορίζει τη φορά κατά την οποία περνάει ο έλεγχος κατά την εκτέλεση του προγράμματος. Μια ενότητα μπορεί να καλεί περισσότερες από μια ενότητες καθώς όμως στο διάγραμμα δομής δε φανερώνεται η ακολουθία δεν είναι γνωστή η σειρά με την οποία

η ενότητα καλεί τις ενότητες που αποτελούν τους απογόνους της. Μια ενότητα χωρίς απογόνους, ονομάζεται φύλλο. Στη συνέχεια παρουσιάζονται οι σχέσεις ελέγχου που εμφανίζονται σε ένα διάγραμμα δομής (Martin and Mc Clure, 1985).

- Στο 1ο επίπεδο ενός διαγράμματος δομής υπάρχει μια και μόνο ενότητα η οποία ονομάζεται ρίζα. Από τη ρίζα ξεκινάει ο έλεγχος που ασκείται στις ενότητες των υπόλοιπων επιπέδων.

- Αρχίζοντας από το 1ο επίπεδο του διαγράμματος δομής, ο έλεγχος περνάει σε όλες τις ενότητες από επίπεδο σε επίπεδο. Ο έλεγχος επιστρέφει πάντοτε στην ενότητα που τον ασκεί, έτσι όταν ολοκληρωθεί η εκτέλεση του προγράμματος, ο έλεγχος επιστρέφει στη ρίζα.

- Ανάμεσα σε δύο οποιεσδήποτε ενότητες, υπάρχει το πολύ μια σχέση ελέγχου. Έτσι, αν μια ενότητα E1 ασκήσει έλεγχο σε μια ενότητα E2, η ενότητα E2 δεν μπορεί να ασκήσει και αυτή με τη σειρά της έλεγχο στην ενότητα E1. Επίσης μια ενότητα δε μπορεί να ασκήσει έλεγχο στον εαυτό της.

Κοινές ενότητες

Επιτρέπεται περισσότερες από μια ενότητες να μεταφέρουν τον έλεγχο στην ίδια ενότητα.

Ενότητες βιβλιοθήκης

Πολλές φορές χρησιμοποιούνται εκ των προτέρων κάποιες υλοποιημένες ενότητες βιβλιοθήκης οι οποίες συνήθως βρίσκονται στο διάγραμμα δομής με τη μορφή φύλλων.

Μεταφορά δεδομένων

Όταν μεταφέρεται ο έλεγχος από μια ενότητα σε μια άλλη μεταφέρονται συνήθως και κάποια δεδομένα τα οποία μπορούν να μεταφερθούν προς οποιαδήποτε κατεύθυνση ανάμεσα στις δύο οντότητες. Η κατεύθυνση καθορίζεται από ένα βέλος ενώ τα ονόματα των δεδομένων που μεταφέρονται σημειώνονται δίπλα στο βέλος.

Κέντρο συναλλαγών

Όταν σε μια ενότητα ενός διαγράμματος δομής γίνεται η επεξεργασία πολλών διαφορετικών τύπων συναλλαγών, μπορεί για κάθε διαφορετικό τύπο συναλλαγής να χρησιμοποιηθεί μια επιμέρους ενότητα.



Σχήμα 3.5.1.1 Διάγραμμα δομής για το υποσύστημα, ενός συστήματος επεξεργασίας αιτήσεων για μεταπτυχιακές σπουδές, το οποίο αφορά την πραγματοποίηση συνεντεύξεων και την υποβολή εξετάσεων σε υποψήφιους μεταπτυχιακούς φοιτητές.

3.6 Διαγράμματα HIPO

Τα διαγράμματα HIPO (Hierarcical Input-Process-Output) δηλαδή ιεραρχικής εισόδου - επεξεργασίας - εξόδου, λόγω της χρήσης τους και των χαρακτηριστικών τους θεωρούνται περισσότερο τεχνικές λεπτομερούς περιγραφής. Παρ' όλ' αυτά, χρησιμοποιούνται και ως τεχνικές γενικής περιγραφής. Αποτελούν μια τεχνική που χρησιμοποιεί ένα σύνολο διαγραμμάτων για να απεικονίσει την είσοδο και την έξοδο ενός προγράμματος καθώς και την επεξεργασία που υφίστανται τα δεδομένα στα πλαίσια του προγράμματος αυτού, δηλαδή τις λειτουργίες που οδηγούν από τα δεδομένα εισόδου στα δεδομένα εξόδου. Στην πραγματικότητα αποτελούν μια διαγραμματική τεχνική που περιγράφει τι κάνει ένα σύστημα και όχι πως το κάνει. Χρησιμοποιούνται συχνά και σε συνδυασμό με τα διαγράμματα ροής δεδομένων, ενώ αποτελούν μια αρκετά καλή μέθοδο τεκμηρίωσης, αφού παρέχουν τη δυνατότητα για περιγραφή των συστημάτων και των προγραμμάτων που αναλύουν.

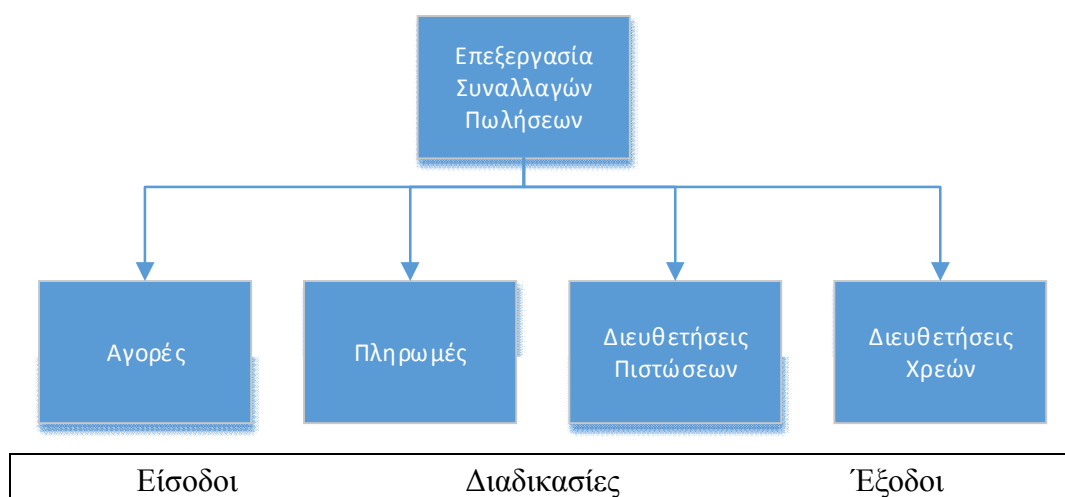
3.6.1 Δομικά στοιχεία των διαγραμμάτων HIPO

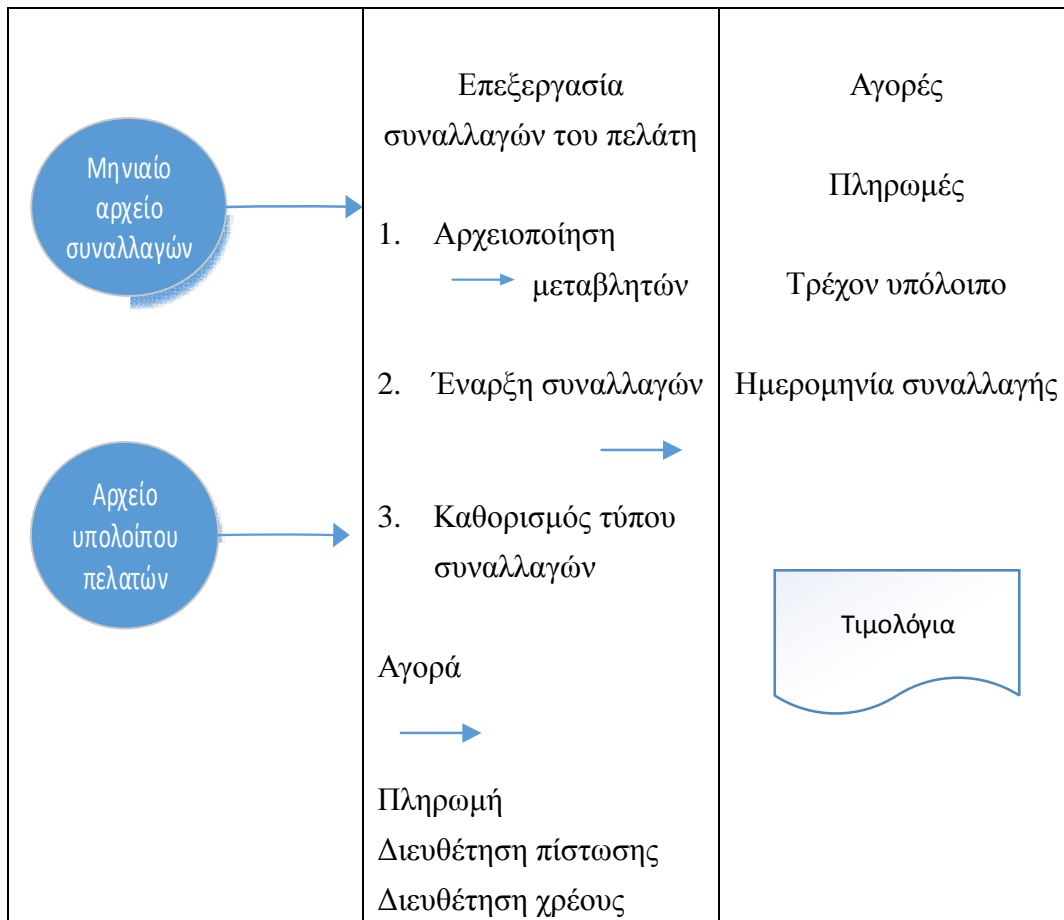
Το γενικότερο πακέτο των διαγραμμάτων HIPO αποτελείται από δύο μέρη: τον παραστατικό πίνακα περιεχομένων (VTOC - Visual Table of Contents) και τα διαγράμματα εισόδου - επεξεργασίας - εξόδου (γενικά και λεπτομερειακά). Ο παραστατικός πίνακας περιεχομένων αποτελείται από:

- το ιεραρχικό διάγραμμα που αποτελεί το διάγραμμα που περιγράφει τις λειτουργίες του συστήματος/προγράμματος. Κάθε λειτουργία αναλύεται σε υπολειτουργίες. Έτσι, με αυτή τη ταξινόμηση διευκολύνεται το έργο του αναλυτή διότι προκύπτουν διάφορα επίπεδα ανάλυσης. Το ιεραρχικό διάγραμμα καθορίζει επίσης τα ονόματα και τους αριθμούς των γενικών και των λεπτομερειακών διαγραμμάτων.
- τον επεξηγηματικό πίνακα που περιέχει όλα τα σύμβολα που χρησιμοποιούνται στο πακέτο και περιγράφει το συμβολισμό τους.
- το τμήμα περιγραφής που δίνει προαιρετικά μια λεπτομερειακή περιγραφή της λειτουργίας του συστήματος/προγράμματος.

Τα διαγράμματα εισόδου-επεξεργασίας-εξόδου διακρίνονται σε:

- γενικά διαγράμματα που περιγράφουν γενικά κάθε λειτουργία του συστήματος. Τα δεδομένα εισόδου αναφέρονται στα αριστερά ενώ τα δεδομένα εξόδου (αποτελέσματα) αναγράφονται στα δεξιά. Στο μεσαίο ορθογώνιο του γενικού διαγράμματος υπάρχει συνήθως μια γενική περιγραφή της λειτουργίας που αναλύεται. Το διάγραμμα δεν περιγράφει πούθενά πως και πού χρησιμοποιούνται τα δεδομένα εισόδου και εξόδου όπως επίσης δεν περιέχει και λεπτομέρειες υλοποίησης της συγκεκριμένης λειτουργίας.
- λεπτομερειακά διαγράμματα όπου τα δεδομένα εισόδου και εξόδου παίρνουν την φυσική τους σημασία, δηλαδή εμφανίζονται με τη μορφή αρχείων ή εγγραφών σε αρχεία. Αυτά τα διαγράμματα περιέχουν και το όνομα της ρουτίνας που υλοποιεί τη συγκεκριμένη λειτουργία.





Σχήμα 3.6.1.1 Παράδειγμα ενός γενικού διαγράμματος HIPO.

3.6.2 Πλεονεκτήματα των διαγραμμάτων HIPO

Τα διαγράμματα HIPO έχουν βρει μεγάλη απήχηση ανάμεσα στους αναλυτές και σχεδιαστές Π.Σ. διότι εμφανίζουν τα εξής πλεονεκτήματα:

- χρησιμοποιούνται για τον ορισμό των συστατικών στοιχείων ενός συστήματος - προγράμματος με γενικό τρόπο. Ο ορισμός που προκύπτει αποτελεί σημείο αφετηρίας για την φάση της σχεδίασης.
- χρησιμοποιούνται για την αναπαράσταση κάθε ενός από τα συστατικά στοιχεία του συστήματος/προγράμματος με λεπτομερή τρόπο.
- βοηθούν στην προοδευτική τεκμηρίωση του νέου συστήματος - προγράμματος.
- υποστηρίζονται από πολλά πακέτα λογισμικού του εμπορίου.

3.7 Διαγράμματα HOS

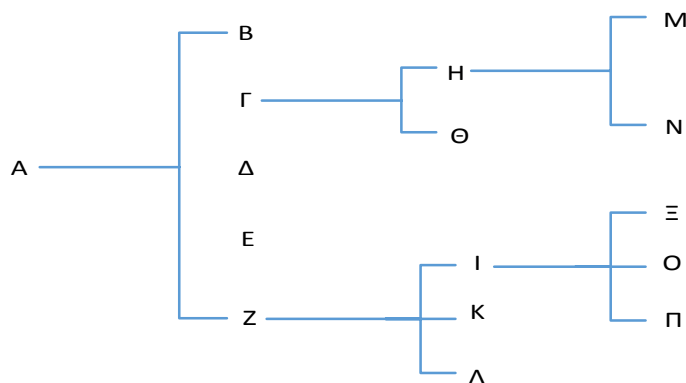
Τα διαγράμματα HOS (Higher Order Software) που δημιουργήθηκαν από τους Hamilton και Zeldin (1976) είναι διαγραμματικές τεχνικές και γενικής και λεπτομερούς περιγραφής. Υποστηρίζουν αποδοτικά τη διαδικασία της λειτουργικής

αποσύνθεσης και μάλιστα δίνουν τη δυνατότητα στον αναλυτή να φτάσει μέχρι τη δημιουργία απλών προγραμμάτων από τα οποία μπορεί εύκολα να παραχθεί ο αντίστοιχος εκτελέσιμος κώδικας.

Τα διαγράμματα HOS επιτρέπουν μόνο αυστηρά καθορισμένες μορφές αποσύνθεσης, σε αντίθεση με άλλες διαγραμματικές τεχνικές οι οποίες αφήνουν τις λεπτομέρειες της αποσύνθεσης των διεργασιών στην κρίση του αναλυτή/σχεδιαστή. Όμως, επειδή οι μορφές αυτές έχουν ορισθεί με μεγάλη ακρίβεια, χρησιμοποιώντας μαθηματικούς κανόνες, η ορθότητα της διαδικασίας αποσύνθεσης είναι εύκολο να ελεγχθεί.

3.8 Διαγράμματα Warnier-Orr

Τα διαγράμματα Warnier-Orr αποτελούν διαγραμματική τεχνική που μπορεί να χρησιμοποιηθεί τόσο για γενική όσο και για λεπτομερή περιγραφή ενός συστήματος. Βασικός τους σκοπός είναι να παραστήσουν διαγραμματικά την ιεραρχική δομή ενός προγράμματος, μιας δομής δεδομένων ή και ολόκληρου του πληροφοριακού συστήματος. Έχουν πάρει το όνομα τους από τους Dominique Warnier και Ken Orr που πρώτοι πρότειναν τη χρησιμοποίησή τους. Στο παρακάτω σχήμα παρουσιάζεται ένα παράδειγμα της δομής ενός τυπικού διαγράμματος Warnier-Orr. Τα βασικά χαρακτηριστικά των διαγραμμάτων αυτών είναι ότι κατά την σχεδίαση τους χρησιμοποιούν άγκιστρα και εκτείνονται από την αριστερή προς τη δεξιά πλευρά της σελίδας. Κάθε άγκιστρο συμβολίζει μια λειτουργική ανάλυση του στοιχείου που βρίσκεται δεξιά από το άγκιστρο στα στοιχεία που βρίσκονται αριστερά του. Τα διαγράμματα Warnier-Orr μπορούν να χρησιμοποιηθούν είτε για την απεικόνιση της δομής ενός προγράμματος είτε για την απεικόνιση της διάσπασης των σύνθετων δεδομένων ενός προγράμματος στα στοιχεία από τα οποία αποτελούνται.



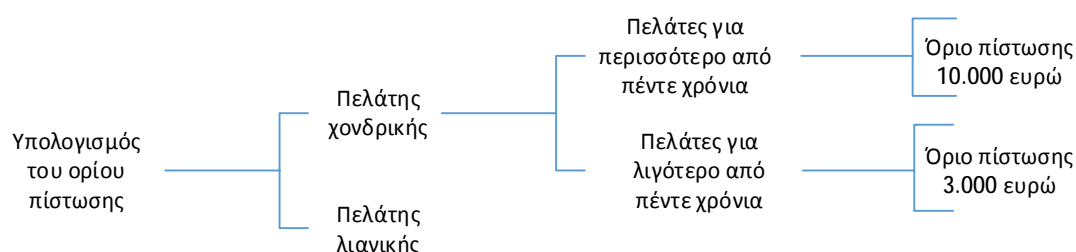
Σχήμα 3.8.1 Η δομή ενός τυπικού διαγράμματος Warnier - Orr.

Τα διαγράμματα Warnier-Orr μπορούν να χρησιμοποιηθούν τόσο κατά τη διαδικασία κατασκευής νέων συστημάτων όσο και κατά την ανάλυση και περιγραφή συστημάτων που ήδη υπάρχουν. Επίσης, πρέπει να τονιστεί στο σημείο αυτό, ότι είναι ιδιαίτερα εύκολη η διαδικασία μετατροπής ενός διαγράμματος Warnier-Orr σε κώδικα δομημένου προγραμματισμού λόγω της χρήσης της δομής begin-end.

Σε σχέση με τα διαγράμματα HIPO παρατηρεί κανείς και οι δύο διαγραμματικές τεχνικές μπορούν με επιτυχία να χρησιμοποιηθούν για την απεικόνιση ενός συστήματος/προγράμματος, σε διάφορα επίπεδα λεπτομέρειας, εφαρμόζοντας μια διεργασία λειτουργικής αποσύνθεσης. Η μόνη διαφορά ανάμεσά τους είναι ότι τα διαγράμματα HIPO χρησιμοποιούνται μόνο για την απεικόνιση διαδικασιών ενώ τα Warnier-Orr χρησιμοποιούνται τόσο για την απεικόνιση διαδικασιών όσο και για την απεικόνιση δεδομένων. Πάντως, τα διαγράμματα Warnier-Orr έχοντας ανώτερη διαγραμματική τεχνική και όντας κατάλληλα να αντικαταστήσουν άλλους συνδυασμούς διαγραμματικών τεχνικών (όπως για παράδειγμα διαγράμματα δομής και ψευδοκώδικα ή διαγράμματα δομής και διαγράμματα Nassi-Shneiderman) είναι προτιμότερα από τα διαγράμματα HIPO.

Στο σημείο αυτό θα πρέπει να τονιστεί το εξής. Τα διαγράμματα Warnier-Orr (όπως και τα διαγράμματα HIPO) αν χρησιμοποιηθούν για την ανάλυση των υψηλών επιπέδων ενός συστήματος - προγράμματος προσφέρουν ξεκάθαρη απεικόνιση της δομής του. Εάν όμως χρησιμοποιηθούν για ανάλυση σε χαμηλό επίπεδο μπορεί να αυξηθεί η έκταση τους και να μειωθεί η αναγνωσιμότητά τους.

Συμπερασματικά, τα διαγράμματα Warnier-Orr αν και μειονεκτούν στο ότι μπορούν να απεικονίσουν μόνο ιεραρχικές δομές δεδομένων είναι κατάλληλα για τη σχεδίαση και την τεκμηρίωση μικρών σχετικά συστημάτων/προγραμμάτων που χρησιμοποιούν απλές δομές αρχείων και χαρακτηρίζονται από υψηλό βαθμό αλληλεπίδρασης με το περιβάλλον τους.



Σχήμα 3.8.2 Η δομή διαγράμματος Warnier-Orr για ένα πρόγραμμα εύρεσης του ορίου πίστωσης για τους πελάτες μιας επιχείρησης.

3.9 Διαγράμματα ενεργειών

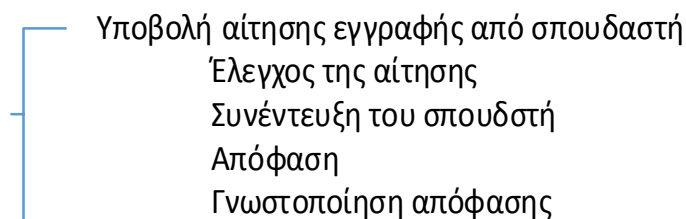
Τα διαγράμματα ενεργειών αποτελούν διαγραμματική τεχνική που μπορεί να χρησιμοποιηθεί τόσο για γενική όσο και για λεπτομερή περιγραφή ενός συστήματος. Αυτό έχει ως αποτέλεσμα να είναι ιδιαίτερα κατάλληλα για την ανάπτυξη πολύπλοκων πληροφορικών συστημάτων, αφού μπορούν να χρησιμοποιηθούν σε όλα τα στάδια της ανάλυσης και του σχεδιασμού προσφέροντας στο τέλος ένα ενιαίο διαγραμματικό περιβάλλον.

Τα βασικά χαρακτηριστικά των διαγραμμάτων ενεργειών είναι τα εξής:

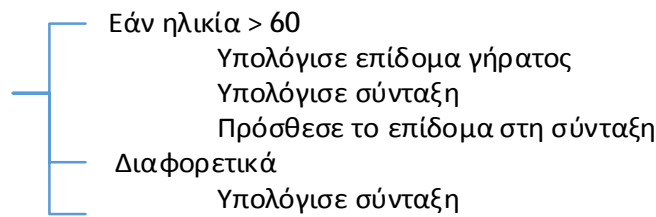
- οι αγκύλες
- οι συνθήκες ελέγχου
- οι υπορουτίνες
- οι ανακυκλώσεις
- τα σημεία τερματισμού
- οι γλώσσες τέταρτης γενιάς
- ο παραλληλισμός

Τα πλεονεκτήματα των διαγραμμάτων ενεργειών είναι τα εξής:

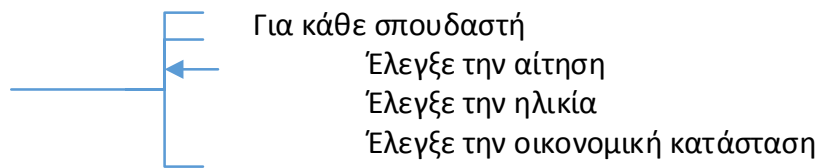
- είναι εύκολη η εκμάθηση του τρόπου χρήσης τους
- προσφέρονται και για χειρογραφική χρήση και για χρήση μέσω Η.Υ.
- παρέχουν τη δυνατότητα για γρήγορη και εύκολη σχεδίαση και εφαρμογή αλλαγών.
- χαρακτηρίζονται από δυνατότητες αποσύνθεσης, δηλαδή επιτρέπουν εύκολα τη μεταβίβαση από τη γενική περιγραφή σε λεπτομερή περιγραφή
- είναι πιο ευανάγνωστα από τον ψευδοκώδικα
- υποστηρίζουν όλες τις δομές του δομημένου προγραμματισμού, που είναι η σειριακή εκτέλεση, η επιλογή, η επανάληψη, η διαδικασία και η ταυτόχρονη εκτέλεση, όπως παραθέτονται στα παρακάτω σχήματα
- μπορούν να χρησιμοποιηθούν σε συνδυασμό με λεξικά δεδομένων
- συνδυάζονται με επιτυχία με τις γλώσσες τέταρτης γενιάς.



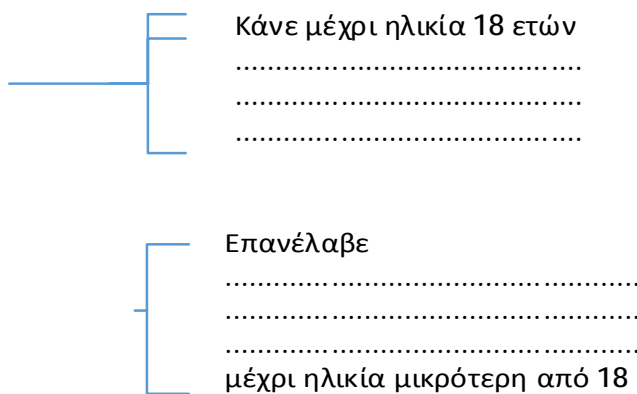
Σχήμα 3.9.1 Διάγραμμα ενεργειών για τη σειριακή εκτέλεση.



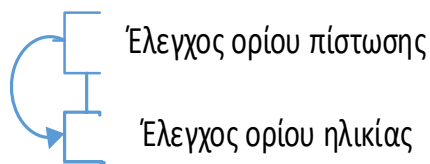
Σχήμα 3.9.2 Διάγραμμα ενεργειών για τη δομή της επιλογής.



Σχήμα 3.9.3 Διάγραμμα ενεργειών για τη δομή της επανάληψης με γνωστό πλήθος επαναλήψεων.



Σχήμα 3.9.4 Διάγραμμα ενεργειών για τη δομή της επανάληψης χωρίς εκ των προτέρων γνωστό πλήθος επαναλήψεων.



Σχήμα 3.9.5 Διάγραμμα ενεργειών για ταυτόχρονη εκτέλεση.

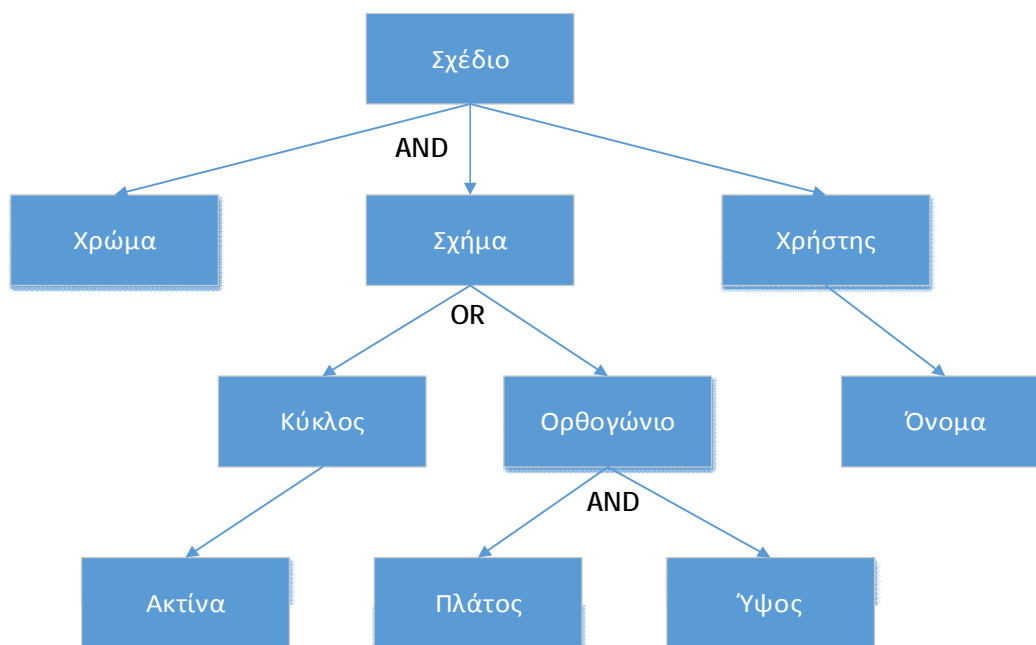
3.10 Διαγράμματα Michael Jackson

Τα διαγράμματα Michael Jackson μπορούν να παραστήσουν γραφικά τόσο δομές προγραμμάτων όσο και δομές δεδομένων, δίνοντας ιδιαίτερη έμφαση στον τρόπο με τον οποίο προκύπτουν οι δομές των προγραμμάτων από τις δομές των δεδομένων. Για το λόγο αυτό, στην ανάλυση ενός προγράμματος χρησιμοποιούνται κυρίως οι δομές των δεδομένων εισόδου και εξόδου. Τα διαγράμματα Michael Jackson χρησιμοποιούν διαγράμματα δενδρικής μορφής και γι' αυτό σε ένα βαθμό μοιάζουν με τα διαγράμματα δομής.

Τα διαγράμματα δενδρικής μορφής που χρησιμοποιούνται αποτελούνται από τα εξής βασικά δομικά στοιχεία:

- στοιχειώδες δομικό στοιχείο
- ακολουθία
- επανάληψη
- επιλογή

Συμπερασματικά, ένα διάγραμμα Michael Jackson, για κάθε συγκεκριμένο πρόγραμμα που εξετάζεται, αντιστοιχεί τα δομικά στοιχεία των δεδομένων εισόδου σε δομικά στοιχεία δεδομένων εξόδου. Έτσι, διευκολύνεται σημαντικά η διαδικασία σχεδίασης των προγραμμάτων.



Σχήμα 3.10.1 Παράδειγμα διαγράμματος Michael Jackson.

3.11 Δομημένη περιγραφή και ψευδοκώδικας

Οι περισσότερες από τις τεχνικές μοντελοποίησης που παρουσιάστηκαν στο προηγούμενο κεφάλαιο χρησιμοποιούν ιεραρχικά διαγράμματα δενδρικής μορφής για την περιγραφή του υπό ανάλυση συστήματος ή προγράμματος. Σε αυτό το κεφάλαιο, οι τεχνικές που παρουσιάζονται (δομημένη περιγραφή και ψευδοκώδικας) δε χρησιμοποιούν διαγράμματα τέτοιας μορφής. Αποτελούν μια διασκευασμένη διάλεκτο της ομιλούμενης γλώσσας που χρησιμοποιείται στη περιγραφή της δομής προγραμμάτων.

Οι διαφορές του ψευδοκώδικα και της δομημένης περιγραφής είναι ελάχιστες. Ο ψευδοκώδικας χρησιμοποιεί περισσότερο τυπικό συμβολισμό και απευθύνεται κυρίως σε ειδικούς ανάπτυξης λογισμικού. Η μορφή του μοιάζει περισσότερο με τη μορφή που θα έχει ο κώδικας που θα υλοποιήσει τη διεργασία που περιγράφεται. Ως αποτέλεσμα πολλές φορές περιλαμβάνει εντολές που αντιστοιχούν σε αντίστοιχες εντολές της γλώσσας προγραμματισμού που θα χρησιμοποιηθεί για την τελική υλοποίηση. Στο παρακάτω σχήμα παρουσιάζεται ένα απλό παράδειγμα ψευδοκώδικα για ένα πρόγραμμα που ζητάει από το χρήστη δύο αριθμούς. Πρώτα ένα μεγάλο και στη συνέχεια ένα μικρότερο.

Ο ψευδοκώδικας παρουσιάζει τα εξής βασικά χαρακτηριστικά :

- έχει ιεραρχική δομή,
- έχει δομή παρόμοια με τη δομή του κώδικα προγράμματος που θα χρησιμοποιηθεί για την υλοποίησή του,
- είναι εύκολη η κατανόησή του,
- υποστηρίζει τη χρήση επεξηγηματικών σχολίων.

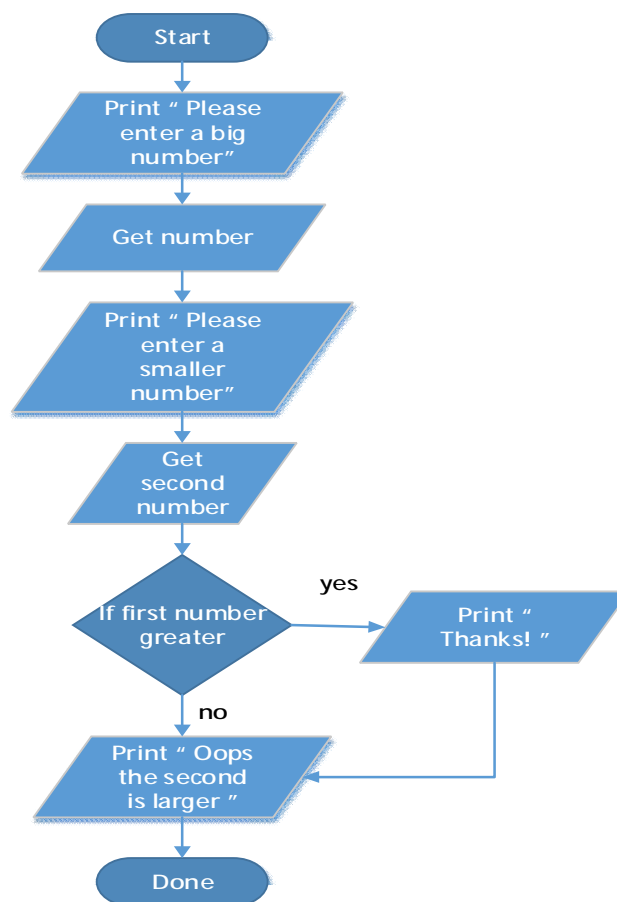
Η δομημένη περιγραφή υποστηρίζει τους εξής βασικούς τύπους δομών:

1. ακολουθία, όπου οι ενέργειες εκτελούνται με τη σειρά που εμφανίζονται (ακολουθιακά) από πάνω προς τα κάτω.
2. επιλογή (if then else), όπου αν μια συνθήκη αληθεύει ,τότε μια συγκεκριμένη ενέργεια εκτελείται. Σε αντίθετη περίπτωση, εκτελείται μια διαφορετική ενέργεια.
3. δομή case, όπου υπάρχουν πολλές αμοιβαία αποκλειόμενες συνθήκες. Όταν μια από τις συνθήκες αυτές αληθεύει, εκτελείται μια συγκεκριμένη ενέργεια που αντιστοιχεί στη συγκεκριμένη συνθήκη.
4. επανάληψη (repeat while, repeat until), όπου η εκτέλεση ενός καθορισμένου συνόλου ενεργειών επαναλαμβάνεται όσο ισχύει μία συγκεκριμένη συνθήκη.

Program Enter a large and smaller number

```
Print "Please enter a big number"
Get the number
Print "Please enter a smaller number"
Get the second number
If the first number is larger
Then
    Print "Thanks!"
Else
    Print "Oops! The second is bigger."
End if
End Program Enter a large and smaller number
```

Παράδειγμα 3.10.1 Ψευδοκώδικας ενός απλού προγράμματος.



Σχήμα 3.10.1 Διαγραμματική αναπαράσταση του ψευδοκώδικα του σχήματος με χρήση Διαγράμματος Ροής.

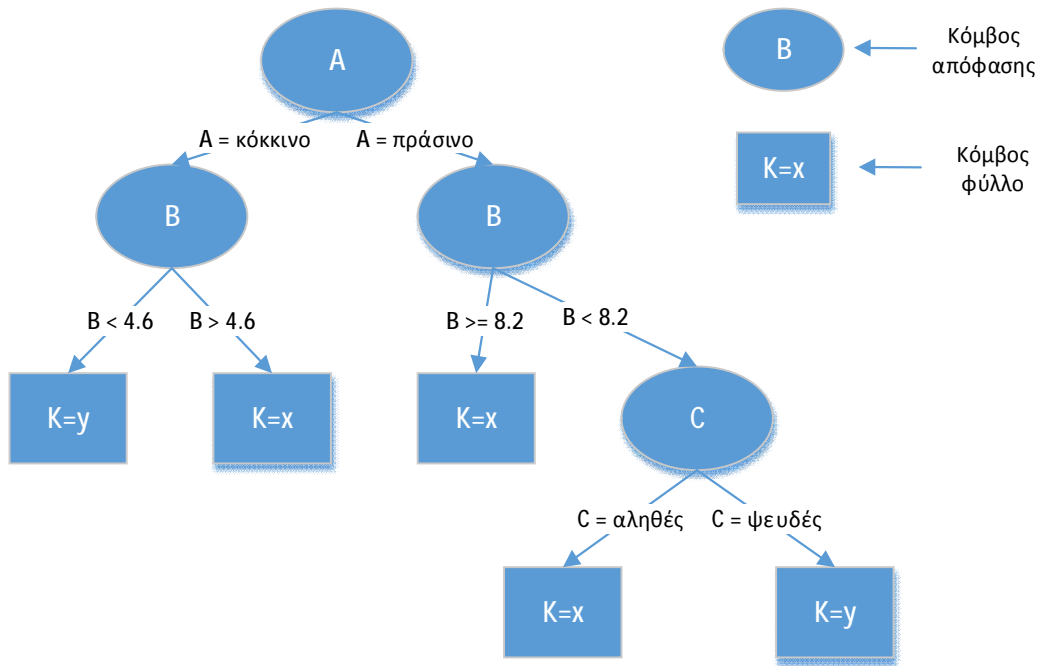
3.12 Δένδρα αποφάσεων

Τα δέντρα αποφάσεων ανήκουν στις διαγραμματικές τεχνικές λεπτομερούς περιγραφής. Εκτός από το χώρο της ανάπτυξης πληροφοριακών συστημάτων, χρησιμοποιούνται και σε διάφορους άλλους επιστημονικούς/τεχνολογικούς χώρους:

- ανάλυση αλγορίθμων.
- αναγνώριση προτύπων.
- σχεδίαση λογικών κυκλωμάτων και έλεγχος αξιοπιστίας τους.

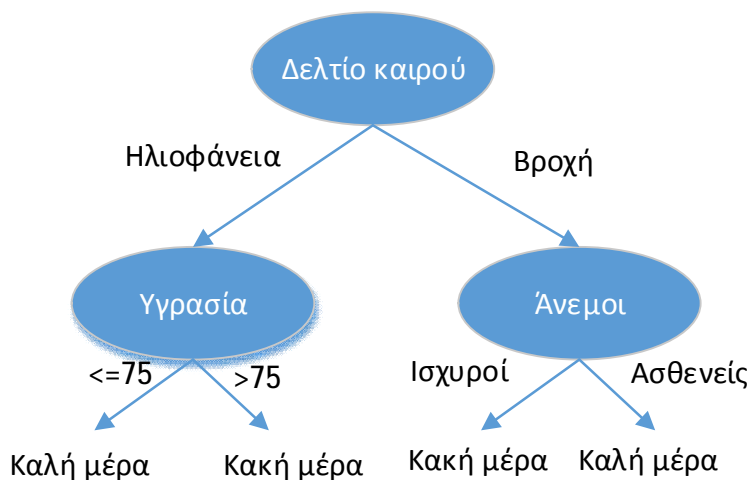
Ένα δέντρο απόφασης στην πραγματικότητα αποτελεί μια διακριτή συνάρτηση η οποία ανάλογα με την τιμή μιας μεταβλητής καθορίζει την εκτέλεση ή μη μιας συγκεκριμένης ενέργειας. Η ενέργεια αυτή μπορεί να είναι είτε η εμφάνιση της τιμής της συνάρτησης είτε η ανάθεσή της τιμής αυτής σε μια άλλη μεταβλητή η οποία με τη σειρά της μπορεί να οδηγήσει στην εκτέλεση μιας άλλης ενέργειας. Επομένως η εκτέλεση μιας ενέργειας εξαρτάται τόσο από την τρέχουσα τιμή μιας μεταβλητής όσο και από όλες τις υπόλοιπες ενέργειες που έχουν ολοκληρωθεί πριν από αυτή.

Τα δέντρα αποφάσεων κατασκευάζονται συνήθως με βάση την περιγραφή ενός προβλήματος. Προσδιορίζουν κυρίως τη διαδικασία λήψης αποφάσεων που απαιτείται για την επίλυση του προβλήματος ενώ παράλληλα καθορίζουν τις μεταβλητές ελέγχου, τις ενέργειες που εκτελούνται και τη σειρά με την οποία πρέπει να εκτελεστούν. Κάθε φορά που εκτελείται ένα δέντρο απόφασης, ακολουθείται ένα μονοπάτι πάνω στο δέντρο, το οποίο εξαρτάται από τις τιμές που παίρνουν οι μεταβλητές ελέγχου. Το μονοπάτι αυτό ξεκινάει πάντα από τη ρίζα και καταλήγει σε ένα φύλλο του δέντρου.



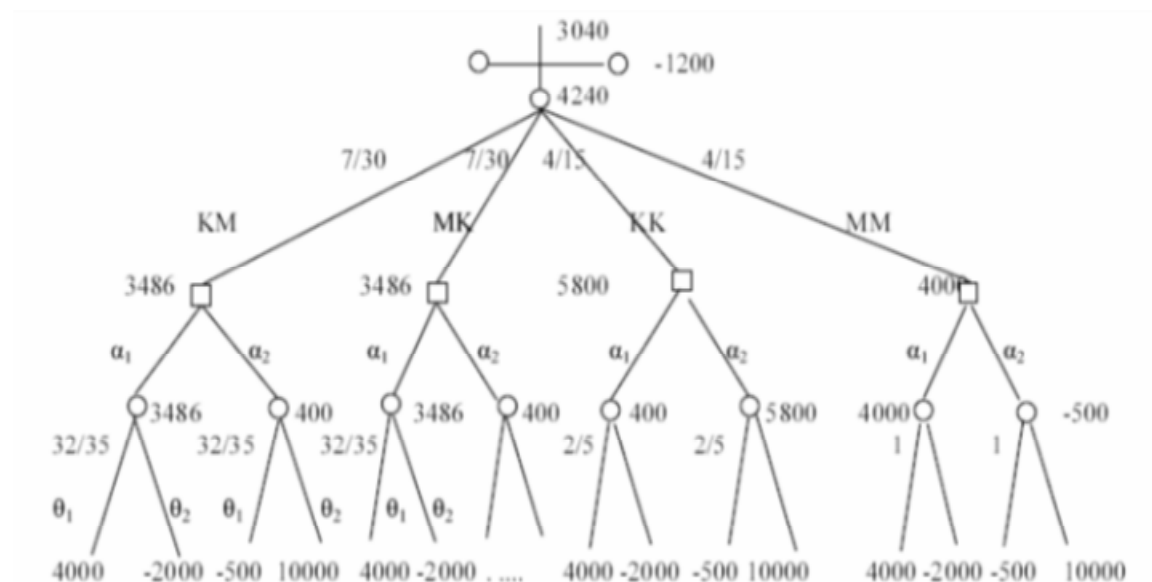
Σχήμα 3.12.1 Ένα απλό παράδειγμα δένδρου απόφασης.

Στο δένδρο απόφασης του παραπάνω σχήματος υπάρχουν δύο είδη κόμβων, οι κόμβοι απόφασης, και οι τερματικοί κόμβοι (κόμβοι φύλλα). Οι κόμβοι απόφασης αντιπροσωπεύουν σημεία στα οποία η επιχείρηση πρέπει να πάρει κάποια απόφαση επιλέγοντας μέσα από ένα σύνολο δυνατών ενεργειών. Η απόφαση συνήθως εξαρτάται είτε από την τιμή μιας μεταβλητής είτε από μια συγκεκριμένη κατάσταση. Οι τερματικοί κόμβοι περιέχουν το τελικό αποτέλεσμα που θα έχει η πορεία μέσα στο δένδρο των αντίστοιχων αποφάσεων που οδηγούν σε αυτούς.



Σχήμα 3.12.2 Παράδειγμα ενός δένδρου απόφασης.

Στο σχήμα 3.12.2 παρουσιάζεται ένα δένδρο απόφασης για το εάν η αυριανή ημέρα είναι «καλή» για να παίξει κάποιος γκολφ.



Σχήμα 3.12.3 Παράδειγμα ενός δένδρου απόφασης με κόμβους τυχαιότητας

Στο σχήμα 3.12.3 παρουσιάζεται ένα δένδρο απόφασης που περιέχει ένα επιπλέον είδος κόμβων, του κόμβους τυχαιότητας (συμβολίζονται με κύκλο). Οι κόμβοι τυχαιότητας αντιπροσωπεύουν σημεία στα οποία η τύχη ή η πιθανότητα παίζει καθοριστικό ρόλο στη λήψη κάποιας απόφασης (ή στην εκτέλεση κάποιας ενέργειας).

3.13 Πίνακες αποφάσεων

Οι πίνακες αποφάσεων ανήκουν στις διαγραμματικές τεχνικές λεπτομερούς περιγραφής. Αποτελούν ένα εναλλακτικό τρόπο παρουσίασης των πληροφοριών που παρέχουν τα δέντρα αποφάσεων. Χρησιμοποιούνται κυρίως για τον προσδιορισμό των ενεργειών που πρέπει να εκτελεστούν όταν ισχύουν ορισμένοι συνδυασμοί συνθηκών. Αποτελούνται από τέσσερα βασικά τμήματα όπως όπως απεικονίζονται στον παρακατω πίνακα.

Τα τμήματα Α (Συνθήκες) και Β (Κανόνες) αποτελούν το μέρος των συνθηκών και των κανόνων, ενώ τα τμήματα Γ (Υλοποίηση ενεργειών) και Δ (Ενέργειες) αποτελούν τα τμήματα των ενεργειών. Για να είναι πλήρης ένας πίνακας, οι κανόνες του πρέπει συνολικά να εξαντλούν όλους τους δυνατούς συνδυασμούς των συνθηκών, διαφορετικά ο πίνακας είναι ατελής.

Στη συνέχεια παρατίθενται κάποιες χρήσιμες συμβουλές σχετικά με το πότε είναι προτιμότερο να χρησιμοποιούνται πίνακες αποφάσεων και πότε δέντρα αποφάσεων:

A) Όταν υπάρχει αμφιβολία ότι το δέντρο αποφάσεων δεν μπορεί να απεικονίσει επαρκώς την πολυπλοκότητα ενός προβλήματος τότε χρησιμοποιείται πίνακας αποφάσεων.

B) Η χρήση των δέντρων αποφάσεων προτιμάται όταν ο αριθμός των ενεργειών είναι μικρός και δεν είναι επιτρεπτοί όλοι οι δυνατοί συνδυασμοί των συνθηκών.

Γ) Η χρήση πίνακα αποφάσεων προτιμάται όταν ο αριθμός των ενεργειών είναι μεγάλος και είναι επιτρεπτοί όλοι ή οι περισσότεροι από τους δυνατούς συνδυασμούς των συνθηκών.

<p>A. Συνθήκες</p> <p>Οι ερωτήσεις ή προϋποθέσεις που πρέπει να πληρούνται</p>	<p>B. Κανόνες</p> <p>Ο συνδιασμός των προϋποθέσεων σε σχέση με τις “τιμές” που παίρνουν οι ερωτήσεις</p>
<p>Δ. Ενέργειες</p> <p>Το σύνολο των δυνατών ενεργειών</p>	<p>Γ. Υλοποίηση Ενεργειών</p> <p>Οι συγκεκριμένες ενέργειες που αντιστοιχούν σε ένα συγκεκριμένο σύνολο τιμών</p>

Πίνακας 3.13.1 Η γενική μορφή ενός πίνακα αποφάσεων.

Υπάρχουν τρεις βασικοί τύποι πινάκων αποφάσεων:

1. Οι πίνακες αποφάσεων περιορισμένων εγγραφών, όπου όλες οι συνθήκες προσδιορίζονται με ακρίβεια στο τμήμα των συνθηκών.

Προϋποθέσεις	Τιμές						
Εξόφληση μέσα σε 15 μέρες	N	N	N	O	O	O	A
Χρηματικό ποσό (E) σε ευρώ	N	O	O	N	O	O	Λ

E>=2000								Λ
Χρηματικό ποσό (E) σε ευρώ 2000<E<1000	Ο	N	Ο	Ο	N	Ο		Ι Ω
Χρηματικό ποσό (E) σε ευρώ 1000<=E	Ο	Ο	N	Ο	Ο	N		Σ
10% Έκπτωση	X							
6% Έκπτωση		X						
0% Έκπτωση			X	X	X	X		
Καμία ενέργεια								X

Πίνακας 3.13.2 Πίνακας απόφασης περιορισμένων εγγράφων που περιγράφει την πολιτική εκπτώσεων μιας επιχείρησης.

2.Οι πίνακες αποφάσεων εκτεταμένων εγγράφων, όπου το τμήμα των συνθηκών απλώς περιγράφει τις συνθήκες του κάθε κανόνα και όχι τις τιμές των συνθηκών.

Χρόνος εξόφλησης	Μέσα σε 15 ημέρες	Μέσα σε 15 ημέρες	Μέσα σε 15 ημέρες	Μετά από 15 ημέρες	Μετά από 15 ημέρες	Μετά από 15 ημέρες
Χρηματικό ποσό (E) σε ευρώ	E>=2000	1000<E<2000	E<=1000	E>=2000	1000<E<2000	E<=1000
Ενέργειες	10% Έκπτωση	6% Έκπτωση	3% Έκπτωση	2% Έκπτωση	1% Έκπτωση	0% Έκπτωση

Πίνακας 3.13.3 Πίνακας απόφασης εκτεταμένων εγγράφων.

3.Οι πίνακες αποφάσεων μικτών εγγράφων, όπου οι συνθήκες μπορεί να είναι είτε περιορισμένες είτε εκτεταμένες.

Αυτός ο πίνακας απόφασης παριστάνει τις σχέσεις μεταξύ των αποθεμάτων μιας αποθήκης, της εβδομαδιαίας κατανάλωσης των αποθεμάτων αυτών, τη διαθεσιμότητα προμηθευτών, τις παραγγελιές που έχουν γίνει και τις αντίστοιχες πράξεις που πρέπει να γίνουν. Οι πράξεις που πρέπει να γίνουν σημειώνονται με το σύμβολο «X» ή με τον προσδιορισμό τους στο μέρος των πράξεων του κάθε κανόνα. Οι πράξεις που δεν χρειάζεται να εκτελεστούν περιέχουν κενά διαστήματα.

Τα στάδια που ακολουθούνται για τη δημιουργία ενός πίνακα αποφάσεων είναι τα εξής:

A) Προσδιορισμός των συνθηκών και των πράξεων που αναφέρονται στο προς επίλυση πρόβλημα.

B) Συμπλήρωση του πίνακα αποφάσεων.

Γ) Απλοποίηση του πίνακα αποφάσεων.

Δ) Τελικός έλεγχος του πίνακα αποφάσεων.

Αποθέματα < 15	N	N	N	N	N	N	O	Α
Κατανάλωση ανά εβδομάδα	>12	>12	10	10	10	<8		Λ
Διαθεσιμότητα προμηθευτή			O	O	N			ΛΙ
Αριθμός παραγγελιών > 25	O	N	O	N	O	O	N	ΩΣ
Παραγγελία που επείγει	X		X					
Κανονική παραγγελία		X		X	X	X		
Ακύρωση παραγγελίας							X	
Καμία παραγγελία								X

Πίνακας 3.13.4 Πίνακας απόφασης μικτών εγγραφών (όπου N = NAI, O = OXI).

Οι πίνακες αποφάσεων χρησιμοποιούνται με κύριο σκοπό τη λογική απεικόνιση σύνθετων περιπτώσεων επεξεργασίας δεδομένων. Για το λόγο αυτό η χρήση τους βρίσκει μεγάλη απήχηση σε εμπορικές ή/και διοικητικές εφαρμογές. Έτσι χρησιμοποιούνται από κατασκευαστικές εταιρίες, ασφαλιστικές εταιρίες, χρηματοδοτικούς οργανισμούς κλπ. Επίσης χρησιμοποιούνται για την υποστήριξη διαφόρων φάσεων έργων πληροφορικής όπως η ανάλυση και ο σχεδιασμός πληροφοριακών συστημάτων, ο προγραμματισμός, ο εντοπισμός σφαλμάτων, η τεκμηρίωση των συστημάτων-προγραμμάτων κλπ.

Οι πίνακες αποφάσεων παρουσιάζουν τα εξής πλεονεκτήματα:

-Μπορούν να χρησιμοποιηθούν τόσο στην ανάλυση συστημάτων όσο και στον προγραμματισμό και την τεκμηρίωση.

-Μπορούν να περιγράψουν σύνθετες μεταβλητές που αλληλεπιδρούν μεταξύ τους.

-Επιτρέπουν τη διάσπαση σύνθετων προβλημάτων σε απλούστερους πίνακες αποφάσεων, οι οποίοι που μπορούν να συνδεθούν μεταξύ τους.

-Παρουσιάζουν όλες τις δυνατές ενέργειες με σαφήνεια και ακρίβεια.

-Τροποποιούνται εύκολα για να συμπεριλάβουν τυχόν αλλαγές στις δυνατές

ενέργειες.

-Αυτοματοποιούνται με εύκολο τρόπο ενώ γίνονται κατανοητοί και από μη ειδικούς.

-Ελέγχονται εύκολα για ασυνέπειες ή παραλείψεις.

Οι πίνακες αποφάσεων παρουσιάζουν όμως και ένα σημαντικό μειονέκτημα. Σε περιπτώσεις όπου οι καταστάσεις είναι πολύ σύνθετες οι πίνακες που κατασκευάζονται για την περιγραφή τους είναι πολύ μεγάλοι, δυσανάγνωστοι και δυσνόητοι.

3.14 Διαγράμματα Nassi - Shneiderman

Τα διαγράμματα Nassi - Shneiderman αποτελούν μια διαγραμματική τεχνική λεπτομερούς περιγραφής, η οποία χρησιμοποιεί διαγράμματα για τη λεπτομερειακή και δομημένη περιγραφή του αλγόριθμου επίλυσης ενός προβλήματος. Χρησιμοποιούνται κυρίως πριν και κατά τη φάση της κυρίως σχεδίασης. Βοηθούν στην κωδικοποίηση δομημένων προγραμμάτων, ενώ χρησιμοποιούνται και ως εργαλεία τεκμηρίωσης.

Τα διαγράμματα N-S συμβολίζονται με ένα ορθογώνιο παραλληλόγραμμο που αναπαριστά μια ενότητα προγράμματος και το οποίο έχει την παρακάτω γενική μορφή.

ΟΝΟΜΑ ΠΡΟΓΡΑΜΜΑΤΟΣ
ΕΙΣΟΔΟΣ ΔΕΔΟΜΕΝΩΝ
ΠΕΡΙΓΡΑΦΗ ΔΙΑΔΙΚΑΣΙΑΣ
ΕΠΙΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ
ΕΞΟΔΟΣ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

3.14.1 Γενική μορφή των διαγραμμάτων N-S

Στην περιγραφή του αλγόριθμου επίλυσης χρησιμοποιούνται οι συνήθεις δομές του δομημένου προγραμματισμού. Επομένως, η περιγραφή ενός αλγόριθμου αποτελεί ένα συνδυασμό από τα εξής στοιχεία:

1. Ακολουθίες από εντολές εκτέλεσης.

2. Μορφές λογικών υποθέσεων:

α) αν η Συνθήκη είναι αληθής, τότε εκτέλεσε την Ενέργεια (IF...THEN)

β) αν η Συνθήκη είναι αληθής, τότε εκτέλεσε την Ενέργεια_1, διαφορετικά εκτέλεσε την Ενέργεια_2 (IF THEN ELSE)

3. Μορφές επαναλήψεων:

α)επανάλαβε την Ενέργεια ή τις Ενέργειες μέχρις ότου η Συνθήκη γίνει αληθής (DO UNTIL)

β)όσο η Συνθήκη είναι αληθής επανάλαβε την Ενέργεια ή τις Ενέργειες τέλος επανάληψης (DO WHILE).

4. Μορφές επιλογής ενεργειών:

επέλεξε Συνθήκη

Επιλογή 1: Ενέργεια_1 ή Ενέργειες_1

Επιλογή 2: Ενέργεια_2 ή Ενέργειες_2

.

.

.

Επιλογή x: Ενέργεια _x ή Ενέργειες_x

τέλος επιλογής (DO CASE)

Τα διαγράμματα N-S δεν είναι σε θέση να παρουσιάσουν την υψηλού επιπέδου ιεραρχική δομή ελέγχου ενός προγράμματος. Όπως αναφέρθηκε και παραπάνω, αποτελούν μια διαγραμματική τεχνική που χρησιμοποιείται κατά κύριο λόγο για τη λεπτομερειακή σχεδίαση προγραμμάτων. Χρησιμοποιούνται πολλές φορές στη θέση των διαγραμμάτων ροής, των λεπτομερών διαγραμμάτων HIPO και του ψευδοκώδικα. Κι αυτό γιατί σε σχέση με τα διαγράμματα αυτά, τα διαγράμματα N-S είναι πιο ευανάγνωστα και μπορούν πιο εύκολα να μετατραπούν σε κώδικα προγράμματος. Επίσης, δεν επιτρέπουν καμία απόκλιση από το σχήμα του δομημένου προγραμματισμού κατά την κωδικοποίηση του προγράμματος. Μειονέκτημα των διαγραμμάτων N-S αποτελεί το γεγονός ότι η σχεδίαση τους είναι δύσκολη και χρονοβόρα ,καθώς επίσης ότι δεν είναι δυνατή η σύνδεση τους με κάποιο λεξικό δεδομένων.

ΚΕΦΑΛΑΙΟ 4ο

Εισαγωγή

Πληροφοριακό συστήματα ERP

Όπως ήδη γνωρίζουμε το επιχειρησιακό περιβάλλον συνεχώς μεταβάλλεται. Ο ανταγωνισμός ανάμεσα στις επιχειρήσεις αυξάνεται αναλογικά με την επέκταση των αγορών και τις προσδοκίες των πελατών. Το καλύτερο δυνατό που μπορεί να κάνει μια επιχείρηση για να ανταπεξέλθει είναι όχι μόνο να μειώσει το κόστος παραγωγής των προϊόντων της, διατηρώντας και βελτιώνοντας όσο το δυνατόν την ποιότητά τους, αλλά και το κόστος διανομής προς τους πελάτες της, τηρώντας το πιο αποτελεσματικό για την επιχείρηση χρονοδιάγραμμα. Απαραίτητη προϋπόθεση για την πραγματοποίηση αυτού είναι η επιχείρηση να μοιράζεται πληροφορίες με τους προμηθευτές, τους πελάτες και τους διανομείς της. Με αυτόν τον τρόπο θα επιτευχθεί καλύτερος συντονισμός μεταξύ της παραγωγής και της προσφοράς των προϊόντων.

Ένα από τα αποτελεσματικότερα συστήματα για την επίτευξη των στόχων της επιχείρησης είναι τα συστήματα προγραμματισμού επιχειρηματικών πόρων, γνωστά ως Enterprise Resources Planning ή αλλιώς ERP συστήματα. Τα ERP συστήματα βοηθούν στον καλό συντονισμό των πληροφοριών που προκύπτουν από τα διάφορα τμήματα της επιχείρησης και έτσι επιτυγχάνεται η καλύτερη δυνατή διαχείριση των επιχειρηματικών της δραστηριοτήτων. Ένα σύστημα προγραμματισμού επιχειρηματικών πόρων απαρτίζεται από ένα πακέτο λογισμικού, το οποίο ενσωματώνει τα διάφορα πληροφοριακά συστήματα της επιχείρησης μέσω της χρήσης σχεσιακής βάσης δεδομένων. Άρα ένα τέτοιο σύστημα παρέχει μια ολοκληρωμένη εικόνα των λειτουργικών περιοχών της επιχείρησης.

Κάποια από τα βασικότερα υποστήματα που εμπεριέχονται σε ένα ERP σύστημα είναι τα παρακάτω:

- Το υποσύστημα της οικονομικής διαχείρισης
- Της διαχείριση των παγίων
- Το υποσύστημα διοίκησης της παραγωγής
- Αυτό της διοίκησης των αποθεμάτων
- Των πωλήσεων και διανομών
- Των ανθρωπίνων πόρων
- Της διοίκησης των πληροφοριών

Ιστορική αναδρομή

Την δεκαετία του '60 οι επιχειρήσεις στην Ελλάδα αλλά και στο εξωτερικό επικεντρώθηκαν στην μηχανογραφημένη υποστήριξη των πολύπλοκων λειτουργιών τους. Αναπτύχθηκαν συγκεκριμένα πληροφοριακά πακέτα τα οποία υποβοηθούσαν κάποιες διαδικασίες οικονομικής διαχείρισης όπως η λογιστική και η μισθοδοσία ή κάποιες τεχνικές εφαρμογές όπως εφαρμογές ελέγχου και αποθεμάτων.

Στη δεκαετία του '60, εξελίχθηκε μία νέα τεχνική σχεδιασμού υλικών απαιτήσεων, γνωστή ως Material Requirements Planning ή αλλιώς MRP. Η τεχνική αυτή εξάγει την ζήτηση του τελικού προϊόντος που αποκτήθηκε από το κύριο σχέδιο παραγωγής Master Production Schedule-MPS για μια συγκεκριμένη δομή προϊόντος ως ένα λεπτομερές σχέδιο παραγγελίας αγορών ή παραγγελίας παραγωγής λαμβάνοντας υπόψη έναν συγκεκριμένο κατάλογο παραγγελιών. Το MRP έχει απλή λογική αλλά το μέγεθος των δεδομένων που εμπεριέχονται σε αυτό το καθιστά υπολογιστικά δυσκίνητο. Αυτή η διαδικασία θα μπορούσε να πραγματοποιηθεί χειροκίνητα γεγονός που την καθιστά χρονοβόρα, συνεπώς η χρήση Η/Υ για τη διακπεραίωσή της είναι απαραίτητη. Το MRP έχει παρουσιάσει αποτελεσματικότητα όχι μόνο στην μείωση των καταλόγων, αλλά και στην μείωση των χρονικών διαστημάτων μεταξύ παραγωγής και παράδοσης βελτιώνοντας έτσι τον συντονισμό των διαδικασιών αυτών, με αποτέλεσμα την αποφυγή των καθυστερήσεων και κατά συνέπεια την αύξηση της αποδοτικότητας, κάνοντας τις υποχρεώσεις πίο ρεαλιστικές. Στα τέλη της δεκαετίας του '70 δημιουργήθηκε η ανάγκη για ολοκλήρωση των εναπομείναντων πόρων ενός κατασκευαστικού οργανισμού. Έτσι αναπτύχθηκε ένα ολοκληρωμένο σύστημα διαχείρισης κατασκευών το οποίο ονομάστηκε διαχείριση κατασκευαστικών πόρων (Manufacturing Resources Planning - MRP II)

Το MRP II δημιουργήθηκε απο διάφορες λειτουργίες που συνδέοντε μεταξύ τους όπως παραδείγματος χάριν ο επιχειρησιακός σχεδιασμός, ο σχεδιασμός παραγωγής, ο κύριος σχεδιασμός παραγωγής, ο σχεδιασμός υλικών απαιτήσεων, ο σχεδιασμός ικανοτήτων απαιτήσεων καθώς και το εκτελεστικό σύστημα ικανοτήτων και προτεραιοτήτων. Τα αποτελέσματα των προαναφερθέντων θα ολοκληρώνονταν με τις οικονομικές αναφορές, την αναφορά αγοράς, τον προϋπολογισμό, την παραγωγή καταλόγων κλπ.

Στις αρχές του 1980 ξεκινά μια έρευνα για την επιχειρηματική ολοκλήρωση (enterprise integration) η οποία χρησιμοποιεί τις βάσεις δεδομένων και προσπαθεί να ενοποιήσει τις επιχειρηματικές διαδικασίες με προτεραιότητα το κύκλωμα οικονομικής διαχείρισης και παραγωγής. Αποτέλεσμα αυτής της έρευνας είναι η εμφάνιση των συστημάτων Προγραμματισμού Επιχειρηματικών Πόρων (ή αλλιώς τα πλέον ευρέως γνωστά Enterprise Resources Planning systems E.R.P. συστήματα) στα τέλη της δεκαετίας του '80 τα οποία ολοκληρώνουν και άλλες επιχειρηματικές

διαδικασίες όπως για παράδειγμα τη διαχείριση των ανθρωπίνων πόρων, το κύκλωμα των πωλήσεων κλπ. Με λίγα λόγια ένα σύστημα ERP ορίζεται ως ένα ολοκληρωμένο σύστημα πληροφορικής το οποίο καλύπτει όλες τις λειτουργικές περιοχές μιας επιχείρησης για την ικανοποίηση των στόχων της. Επίσης προσπαθεί να ενοποιήσει τους πελάτες και τους προμηθευτές με το κατασκευαστικό περιβάλλον του οργανισμού.

4.1 Στόχοι των ERP συστημάτων

Οι βασικοί στόχοι των ERP συστημάτων είναι :

- Η βελτίωση της ποιότητας των παρεχόμενων προϊόντων και υπηρεσιών.
- Η μείωση του χρόνου παραγωγής.
- Η καλύτερη εξυπηρέτηση των πελατών.
- Η βέλτιστη διαχείριση των αποθεμάτων.
- Η μείωση του συνολικού κόστους σε ολόκληρη την εφοδιαστική αλυσίδα.
- Ο αποτελεσματικότερος συντονισμός του κυκλώματος ζήτηση - παραγωγή - προσφορά.

Με λίγα λόγια σκοπός του ERP είναι να αναπαράγει τις επιχειρηματικές διαδικασίες, όπως για παράδειγμα την διαδικασία καταγραφής μιας πώλησης, τον έλεγχο του ωραρίου των εργαζομένων, την μισθοδοσία κ.ά., καθοδηγώντας έτσι τους υπαλλήλους της επιχείρησης, οι οποίοι είναι υπεύθυνοι για τις διαδικασίες αυτές, βήμα προς βήμα δίνοντάς τους τη δυνατότητα αυτοματοποίησης μεγάλου εύρους διαδικασιών.

4.2 Προβλήματα από την εισαγωγή των συστημάτων ERP

Κάποια από τα πιο συχνά προβλήματα που παρουσιάζονται κατά την εγκατάσταση ενός ERP συστήματος είναι τα εξής:

- 1) Το υψηλό κόστος εγκατάστασης και συντήρησής του.
- 2) Κάποια συστήματα ίσως είναι δύσκολο στο να χρησιμοποιηθούν.
- 3) Στο σύστημα υπάρχει μια αλληλένδετη αλληλουχία με αποτέλεσμα ένα πρόβλημα που μπορεί να προκύψει σε ένα τμήμα να έχει επιπτώσεις και σε όλους τους άλλους συμμετέχοντες.
- 4) Το σύστημα έχει τη δυνατότητα πραγματοποίησης πολλών παραπάνω πραγμάτων από αυτά που συνήθως έχει ανάγκη ο χρήστης με αποτέλεσμα να κοστίζει περισσότερο.
- 5) Χωρίς κατάλληλη εκπαίδευση του προσωπικού το πρόγραμμα είναι δυσνόητο.

4.3 Υλοποίηση ενός συστήματος ERP

Η ορθή υλοποίηση ενός συστήματος ERP είναι απαραίτητη προϋπόθεση για την μακροχρόνια επιτυχία του συστήματος. Για να πραγματοποιηθεί γρήγορη και ορθή υλοποίηση απαιτείται μια συγκεκριμένη και αυστηρή διαδικασία.

Πλέον όλα τα σύγχρονα συστήματα ERP ακολουθούντε απο συγκεκριμένη διαδικασία ή μέθοδο υλοποίησης που συνιστά ο δημιουργός τους. Οι διαδικασίες έχουν χαρακτηριστικές ονομασίες και ακολουθούντε από ειδικά χρονοδιαγράμματα. Οι σηματικές φάσεις κάθε μιας απο αυτές τις διαδικασίες είναι παρόμοιες και περιγράφοντε στη συνέχεια.

Φάση 1η: Προετοιμασία και στρατηγικός σχεδιασμός

Σε αυτή τη φάση διακρίνεται η διαμόρφωση στη δομή της ομάδας υλοποίησης, βάση των αναγκών του εκάστοτε έργου. Μια τυπική ιεραρχία περιλαμβάνει τα εξής επίπεδα διάκρισης ρόλων και ανάληψης ευθυνών:

- 1) Αρχικά έχουμε τον χορηγό του έργου, ο οποίος εξασφαλίζει τους απαραίτητους πόρους. Συνήθως ο ρόλος του χορηγού προτείνεται να αναληφθεί από ένα ανώτατο διοικητικό στέλεχος, όπως από τον αναπληρωτή διευθύνοντα σύμβουλο ή από τον γενικό διευθυντή, ούτος ώστε να εξασφαλίζεται η δέσμευση της διοίκησης.
- 2) Ακολουθεί ο υπεύθυνος του έργου, του οποίου οι αρμοδιότητες του περιορίζονται στην διοίκηση του έργου υλοποίησης. Ο υπεύθυνος έργου πρέπει να έχει μια ολόκληρωμένη αντίληψη των επιχειρηματικών διαδικασιών και των διασυνδέσεών τους.
- 3) Στη συνέχεια έχουμε την επιτροπή παρακολούθησης και αξιολόγησης του έργου, η οποία ασκεί εποπτεία επί αυτού και ανά τακτά χρονικά διαστήματα (πχ ένα μήνα) καταγράφει τις απαιτήσεις της επιχείρησης. Συνήθως αυτή την αρμοδιότητα της επιτροπής την έχουν τα διευθυντικά στελέχη της εταιρίας
- 4) Έπειτα ακολουθούν οι ομάδες έργου, στις οποίες εκτελούνται βασικά τμήματα του έργου. Συνηθίζεται ο manager της εταιρίας να είναι και ο υπεύθυνος κάθε ομάδας έργου ο οποίος είναι υποχρεωμένος να αφιερώνει το 40% εως και 60% του διαθέσιμου χρόνου του.
- 5) Υπεύθυνος διασφάλισης ποιότητας του έργου, ο οποίος έχει έναν συμβουλευτικό ρόλο και η θέση αυτή δεν ανήκει σε ιεραρχικό επίπεδο.

Το πρόγραμμα που χρησιμοποιείται για την υλοποίηση είναι το διάγραμμα PERT ή το GANTT και καθορίζεται η κρίσιμη διαδρομή. παίζει σημαντικό ρόλο για την

επιτυχία της υλοποίησης ανάπτυξη διαδικασίας παρακολούθησης και αναθεώρησης του προγράμματος από την επιτροπή παρακολούθησης.

Φάση 2η: Σχεδιασμός και παραμετροποίηση

Αυτή την φάση μπορούμε να την αποκαλέσουμε και ως ο πυρήνας της διαδικασίας υλοποίησης και απαιτεί πολύ μεγάλη προσοχή από όλους τους συμμετέχοντες στην ομάδα. Μερικές σημαντικές δραστηριότητες περιλαμβάνουν:

- 1) την εγκατάσταση του εξοπλισμού και του λογισμικού καθώς και την δοκιμή λειτουργικότητας
- 2) Η εκπαίδευση των ομάδων έργου στα αντίστοιχα εξειδικευμένα υποσυστήματα του πακέτου
- 3) Αποτύπωση των υφιστάμενων επιχειρηματικών διαδικασιών
- 4) Ανάλυση και ύστερα αξιολόγηση των επιχειρηματικών διαδικασιών
- 5) Προσαρμογή των ποιο πάνω διαδικασιών σε συγκεκριμένες διαδικασίες που υποστηρίζει το πακέτο ERP
- 6) Ανάπτυξη των τιμών για τις παραμέτρους των διαδικασιών του συστήματος
- 7) Ο σχεδιασμός και η υλοποίηση οθονών-διεπαφών και αναφορών
- 8) Και τέλος ο σχεδιασμός και η υλοποίηση επιπέδων πρόσβασης και εξατομίκευση περιβάλλοντος χρηστών.

Η αποτύπωση, ανάλυση και προσαρμογή είναι αρκετά σημαντικά βήματα και για την επιτυχή υποστήριξη της επιχείρησης από το πακέτο ERP αλλά και για την ανταγωνιστικότητα.

Προγεννέστερα της επιλογής και της υλοποίησης ενός συστήματος προγραμματισμού επιχειρηματικών πόρων κρίνεται σκόπιμη η διεξαγωγή έργου προς Ανασχεδιασμό των Επιχειρηματικών Διαδικασιών ή BPR (Business Process Reengineering). Είναι σύνηθες φαινόμενο μεταξύ των εταιριών να δίνεται βαρύτητα στη μηχανοργάνωση των υφιστάμενων διαδικασιών και όχι στην αναδιοργάνωση της εταιρία μέσω αυτής. Τα μέλη της ομάδας επικεντρώνουν τις προσπάθειές τους στην διαλογή της ορθότερης παραλλαγής μεταξύ των διάφορων εναλλακτικών διαδικασιών που υποστηρίζει η πλειοψηφία των ισχυρότερων πακέτων, και στην εξειδίκευση των διαδικασιών αυτών μέσω της ανάπτυξης των αντίστοιχων παραμέτρων. Αντιθέτως, όταν η υλοποίηση εμπλέκεται στις ιδιαιτερότητες και τις πολυπλοκότητες της ήδη υπάρχουσας κατάστασης, το έργο στην καλύτερη των περιπτώσεων καθυστερείται. Σημαντικό ρόλο έχει ο σύμβουλος υλοποίησης στην αποτύπωση, στην ανάλυση και την προσαρμογή των διαδικασιών, όπως και στην παραμετροποίηση του συστήματος

Φάση 3η: Προετοιμασία, δοκιμές και πιλοτική εφαρμογή

Σε αυτή την φάση γίνεται η προετοιμασία της παραμετροποίησης για την εφαρμογή και περιλαμβάνει:

- 1) Μετάβαση δεδομένων (data migration)
- 2) Εκπαίδευση χρηστών
- 3) Τεκμηρίωση διαδικασιών καθώς και συστήματος
- 4) Πιλοτική εφαρμογή
- 5) Έλεγχο αποδοχής
- 6) Αρχική εγκατάσταση του συστήματος
- 7) Σχεδιασμό φάσης υποστήριξης

Μερικά από τα στάδια που ακολουθούνται στην εκπαίδευση των χρηστών είναι:

- η εισαγωγή στον τρόπο χρήσης του συστήματος
- η εκπαίδευση των διαδικασιών και των μεθόδων που υποστηρίζονται από αυτό
- η λεπτομερής εκπαίδευση των βημάτων που εκτελούνται από τον κάθε χρήστη στις εκάστοτε οθόνες
- και η εκμάθηση των εργαλείων του συστήματος μεταξύ άλλων.

Προϋποθέσεις επιτυχίας αποτελούν ο κατάλληλος τρόπος διενέργειας της εκπαίδευσης και η προσαρμογή της βάσης των αναγκών των χρηστών.

Το πρόγραμμα εφαρμόζεται δοκιμαστικά σε περιορισμένο εύρος δεδομένων εμβαθύνοντας παράλληλα στις ιδιαιτερότητες κάθε διαδικασίας για να διαπιστωθεί η πληρότητά του και να γίνουν οι απαραίτητες βελτιώσεις. Για την επιτυχή έναρξη της πλήρους λειτουργίας του συστήματος πρέπει να αντιμετωπιστούν πρώτα τα προβλήματα που γίνονται εμφανή, κατά τη διάρκεια της πιλοτικής εφαρμογής, στο σχεδιασμό των διαδικασιών, στην υλοποίησή τους και στην παραμετροποίηση του συστήματος. Για την διεξαγωγή της δοκιμαστικής εφαρμογής του συστήματος καθώς και για την αξιολόγησή του ορίζονται συγκεκριμένες μέθοδοι και η χρήση ειδικών εργαλείων. Με την ολοκλήρωση της αντιμετώπισης των προβλημάτων και βάση γενικότερα των αποτελεσμάτων που προκύπτουν από τη δοκιμαστική εφαρμογή του συστήματος, περνάμε στην πλήρη λειτουργία του όπως αυτή αναλύεται στην τέταρτη φάση υλοποίησής του.

Φάση 4η: Ολοκλήρωση

Σε αυτή την φάση το σύστημα βρίσκεται σε πλήρη λειτουργία και συνήθως περιλαμβάνονται τα εξής:

- 1) Δοκιμασία στην εκτέλεση πλήρους λειτουργίας
- 2) Αποτύπωση και ανάλυση των αποτελεσμάτων της πλήρους λειτουργίας
- 3) Βελτιστοποίηση συστήματος

Στο σημείο της δοκιμαστικής εκτέλεσης το νέο σύστημα ERP και τα ήδη υπάρχοντα συστήματα της εταιρίας “τρέχουν” παράλληλα. Τα συστήματα που είδη υπάρχουν υποστηρίζουν ακόμη τις επιχειρηματικές διαδικασίες της επιχείρησης. Υστερα γίνεται σύγκριση μεταξύ των αποτελεσμάτων της λειτουργίας των δύο συστημάτων και πραγματοποιούνται οι τελευταίες ρυθμίσεις στο σύστημα ERP.

Στην συνέχεια αφού γίνει η έναρξη πλήρους λειτουργίας σημειώνοντε όλα τα λειτουργικά προβλήματα οπου αναλύοντε και διορθώνοντε

4.4 Στάδια μετά την υλοποίηση

Αφου πραγματοποιηθεί η υλοποίηση υπάρχουν τρία στάδια που οδηγούν την επιχείρηση στο τι πρέπει να κάνει απο την στιγμή που ακολουθεί η έναρξη της παραγωγικής χρήσης

Στάδιο 1ο: Σταθεροποίηση

Αρχικά, το πρώτο στάδιο αφου γίνει η υλοποίηση ο οργανισμός έχει αναλάβει την σταθεροποίηση του συστήματος και να συντονίσει μαζί του τους ανθρώπους που έχουν αναλάβει να το δουλεύουν. Οι διαδικασίες ρυθμίζονται καλύτερα, στην αρχή υπάρχει περίπτωση για μείωση παραγωγικότητας μέχρι τον πλήρη επαναπροσδιορισμό των εργασιών. Σε περίπτωση μείωση απόδοσης η επιχείρηση παρέχει εκπαίδευση και πρακτική εξάσκηση στους χρήστες κατα την υλοποίηση για την αντιμετώπιση του. Τα αποτελέσματα είναι να αυξηθεί η ταχύτητα εκτέλεσης και να μειωθεί το κόστος διαχείρισης των επιχειρηματικών συναλλαγών. Η διάρκεια του σταδίου είναι λίγοι μήνες.

Στάδιο 2ο: Σύνδεση

Σε αυτό το στάδιο συνδέεται το ERP με άλλες εφαρμογές για την απόκτηση νέων ικανοτήτων στον οργανισμό καθώς γίνεται προσπάθεια για βελτίωση των διαδικασιών. Κάποιες απο τις βασικές δραστηριότητες του σταδίου είναι η εκπαίδευση των εργαζομένων, η παροχη κινήτρων και ανάπτυξη νέων προσόντων για να είναι ποιος ικανός ο οργανισμός στην ταχύτερη εκμετάλευση. Αυξάνεται η αποτελεσματικότητα του οργανισμού, καθώς υπάρχει βελτίωση στις διαδικασίες απόφασης που υποστηρίζονται απο το ERP. Με λίγα λόγια σε αυτό το στάδιο οι

χρήστες χρησιμοποιούν τις ικανότητες του συστήματος για την βελτιστοποίηση αποθεμάτων. Η διάρκεια του σταδίου είναι περίπου 6 μήνες

Στάδιο 3ο: Δημιουργία αξίας

Αυτό είναι το τρίτο και τελευταίο στάδιο. όπου υπάρχει ομαλή συνεργασία μεταξύ ανθρώπων, τεχνολογίας και διαδικασιών. Πραγματοποιούνται αλλαγές σε επίπεδο ανταγωνιστικότητας του οργανισμού καθώς γίνεται επαναπροσδιορισμός στις επιχειρηματικές του διαδικασίες και δημιουργείται ένας κινητικός οργανισμός που πρέπει να είναι ικανός αντιδρά άμεσα και να εξυπηρετεί νέες αγορές. Η διάρκεια του τρίτου σταδίου είναι 8 έως 18 μήνες

Τα στάδια που αναφέραμε είναι διαδοχικά και οδηγούν στα οφέλη που επιθυμεί ο οργανισμός. Στο κάθε στάδιο πραγματοποιούνται συγκεκριμένες ενέργειες που θα οδηγήσουν στην δημιουργία ικανοτήτων οι οποίες βασίζονται στην επιτυχία των προηγούμενων ενεργειών. Η επιχείρηση είναι υποχρεωμένη να κατανοήσει το κάθε στάδιο για να είναι ικανή στον σχεδιασμό των ενεργειών της ανα στάδιο, όπου θα γνωρίζει που βρίσκεται, τι προσπαθεί να επιτύχει και τον τρόπο που θα το επιτύχει.

Ο πραγματικός στόχος είναι να αποκτήσει ο οργανισμός όλες τις απαραίτητες ικανότητες και τα οφέλη που προσφέρει το σύστημα, ακολουθώντας τα τρία στάδια και με επιτυχία που θα αποφέρει σημαντικά αποτελέσματα.

4.5 Παράγοντες επιτυχίας της υλοποίησης συστημάτων ERP

Βασικός παράγοντας για την επιτυχή υλοποίηση του συστήματος ERP είναι η λήψη μέτρων για περίπτωση προβλημάτων σε κρίσιμα σημεία του έργου. Τρία από τα σημεία που πρέπει να τα προσέξουμε είναι:

- 1) Το αντικείμενο-πεδίο εφαρμογής (project scope)
- 2) Οι ανθρώπινοι πόροι (resources)
- 3) Η διοίκηση του έργου

Το σύστημα ERP πρέπει να οριοθετηθεί με σαφήνεια γιατί πρέπει οπωσδήποτε να υποστηρίζει τις υφιστάμενες επιχειρησιακές διαδικασίες, ειδάλτως το έργο θα μπορούσε να τοποθετηθεί με τέτοιο τρόπο ώστε να συμπεριλαμβάνει ανασχεδιασμό ανασχεδιασμό των διαδικασιών και υποστήριξη των νέων διαδικασιών από το σύστημα. Επιπροσθέτως να αναφέρουμε πως θα ήταν καλό να αποφύγουμε κάθε μεταβολή του αντικειμένου κατά την διάρκεια της υλοποίησης

Τώρα οσον αφορά την επιλογή ανθρώπινων πόρων, απευθύνεται στα στελέχη της επιχείρησης που είναι πρόθυμοι να πάρουν μέρος στην ομάδα υλοποίησης καθώς και

στα στελέχη της ομάδας συμβούλου υλοποίησης.

Αρχικά τα στελέχη της επιχείρησης πρέπει να εκπροσωπούν όλες τις κρίσιμες λειτουργίες της επιχείρησης και είναι σημαντικά η πλήρης απασχόληση ενός εκπαιδευμένου στελέχους. Επίσης, καλό θα ήταν να δίνοντε τα κατάλληλα κίνητρα στα στελέχη για να αποφευχθεί η πιθανότητα να αποχωρήσουν απο την επιχείρηση την διάρκεια της υλοποίησης η μετά την ολοκλήρωση του έργου.

Όσο αφορά τώρα τα στελέχη του συμβούλου πρέπει να εξασφαλίσει η επιχείρηση πως τα στελέχη που θα διαλέξει θα έχουν εμπειρία στην υλοποίηση συστημάτων ERP σε επιχειρήσεις παρομοίου αντικειμένου. Πρέπει το στέλεχος που θα ηγεθεί κατα της συμβουλευτικής ομάδας να επιλεχθεί προσεκτικά να κατανομαστεί και να καθοριστούν κανόνες αντικατάστασης. Τέλος πρέπει να γίνει αξιολόγηση των στελεχών του συμβούλου απο τα στελέχη της επιχείρησης καθώς και απο το σύμβουλο διασφάλισης ποιότητας.

4.6 Προβλήματα στην λειτουργία του ERP, μετά την υλοποίηση

Η εγκατάσταση του συστήματος ERP γίνεται για να βελτιωθούν και να διασυνδεθούν οι διαδικασίες, χωρίς αυτό να σημαίνει πως την στιγμή που η επιχείρηση θα εγκαταστήσει το ERP θα ληθούν αυτόματα όλα τα προβλήματα της. Όμως και τα προβλήματα τα οποία καταφέρνουμε να τα λύσουμε στην αρχή πρέπει να αξιολογούντε κατά την πορεία του χρόνου καθώς υπάρχει περίπτωση η λειτουργία του ERP συστήματος μπορεί απο διάφορες αιτίες να μην είναι αυτές που περιμένουμε. Υπάρχουν κάποια χαρακτηριστικά τα οποία μας δείχνουν οτι δεν γίνεται ορθή η λειτουργία του ERP:

- 1) Εκδοση ακόμα αρκετων χειρόγραφων τον μήνα
- 2) Όταν ο χρόνος μεταξύ της εγγραφης πληρωμής και της πρόσβασης γίνεται σε κάποια δευτερόλεπτα
- 3) Η πιθανότητα ένδειξης αποθεμάτων κατα της διάρκειας του μηνια που στην πραγματικότητα δεν υπάρχουν
- 4) Οι άνθρωποι στην παραγωγή συνεχίζουν να κάνουν χειροκίνητες αλλαγές
- 5) Ενημέρωση για καθυστέρηση στην παραδοση παραγγελίας απο βασικό προμηθευτή
- 6) Διαμαρτυρίες πελατών επειδή καποιες εκπτώσεις η οποίες έχουν συμφωνηθεί δεν καταγράφοντε στα τιμολόγια
- 7) Ανεπαρκές hardware
- 8) Προσωπικό το οποίο έχει έρθει πρόσφατα στην επιχείρηση και δεν έχει την κατάλληλη εκπαίδευση

- 9) Χειρόγραφες παρεμβάσεις την στιγμή της προετοιμασίας της μισθοδοσίας
- 10) Αποστασιοποίηση του προσωπικού απο το σύστημα σταδιακά
- 11) Κάποια προβλήματα δεν επιλύθηκαν έγκαιρα και εκείνοι το παρακάμπτουν
- 12) Υπεβάρυνση υπερβολικά στο σύστημα με συνέπεια να καθυστερεί τους χρήστες

Με λίγα λόγια το ERP δεν πρέπει έχει αντιμετώπιση σαν ένα project με μια καθορισμένη αρχή και τέλος.Απο την στιγμή που η επιχείρηση θέλει πραγματικά να έχει οφέλη απο αυτο πρέπει και μετά την εγκατάσταση του ERP να αποτελεί για την επιχείρηση ένα έργο διάρκειας. Ουσιαστικά η εγκατάσταση ενός συστήματος ERP ολοκληρώνεται με την εγκατάσταση ενός άλλου συστήματος οταν και αν αυτο συμβει

4.7 Προβλήματα από την υλοποίηση του ERP συστήματος

Τα προβλήματα που οι επιχειρήσεις είναι υποχρεωμένες να αντιμετωπίσουν κατα την υλοποίηση αρχικά είναι η φυσική αντίδραση του ανθρώπου σε κάθε νεωτερισμό και η ύπαρξη τομέων στην επιχείρηση με συγκεκριμένα πληροφοριακά συστήματα τα οποία δεν μας επιτρέπουν τη ροή και την ανταλλαγή πληροφοριών σε ολόκληρη την έκταση της επιχείρησης. Επίσης η ευέλικτη αρχιτεκτονική πολλών συστημάτων ERP δεν επιτρέπει την ενσωμάτωση των συστημάτων αυτών. Επιπροσθέτως οι συχνά προτεινόμενες βέλτιστες πρακτικές λύσεις οι διαδικασίες οι οποίες οι ενσωματωμένες σε κάποια απο τα συστήματα ERP απέτυχαν στο να λάβουν υπόψη τους τις νομικές και πολιτισμικές ιδιαιτερότητες και ακόμα τις ιδιαιτερότητες των αγορών, των κλάδων η ακόμα και επιχειρήσεων που αφεύλοντε στον τρόπο δραστηριοποίηση τους. Η δεύτερη γενια η οποία ανήκει στο σήμερα των συστημάτων ERP καλύπτει ακόμα και τις εξωεπιχειρησιακές η οποίες στην νέα οικονομία είναι ιδιαίτερα επιτακτικές και αυξημένες. Για να υπάρξει πραγματοποίηση της εξωεπιχειρησιακης δραστηριότητας είναι αναγκαίο για συνεργασία συστημάτων και ελεύθερη ροή της πληροφορίας, αυτό σημαίνει δυνατότητα σύνδεσης ετερογενών και συστημάτων χωρίς να έχει σημασία απο το που προέρχοντε ή σε τι λειτουργικά συστήματα εκτελούντε, ακόμα και σύνδεση με πελάτες και συνεργάτες οι οποίοι υπάρχει πιθανότητα να χρησιμοποιούν διαφορετικα συστήματα. Σήμερα η επιθυμία της κάθε επιχείρησης είναι η επέκταση της δραστηριότητας της πέρα απο το e-business, και αυτό θα επιτευχθεί μόνο αν δεν υπάρχουν αυτά τα προβλήματα και θα υπάρξει δυνατότητα για την επιχείρηση η επιτυχία στην παγκόσμια αγορά.

4.8 Τα ανταγωνιστικά πλεονεκτήματα του ERP

Στόχος για μια επιχείρηση η οποία θα τοποθετήσει ένα σύστημα ERP είναι να

απατυχθούν οι ικανότητες ώστε να της επιτρέψουν να λειτουργεί ανταγωνιστικά στο περιβάλλον που έχει την τάση να μεταβάλλεται συνεχώς. Οι ικανότητες σύμφωνα με έρευνα της εταιρίας Deloitte&Touche είναι:

- 1) Η βελτίωση της ποιότητας καθώς και της ορατότητας της πληροφορίας αλλά και των επιχειρηματικών διαδικασιών
- 2) Η ομογενοποίηση και η ολοκλήρωση διαδικασιών και συστημάτων σε μια τεχνολογική πλατφόρμα η οποία θα υποστηρίζει τεχνολογικά προηγμένες επιχειρηματικές εφαρμογές
- 3) Να υπάρχει ευέλικτη ανταπόκριση προς τον πελάτη αλλά και τους επιχειρηματικούς πόρους.

Με λίγα λόγια αυτές οι τρεις ικανότητες που ανέφερα έχουν σκοπό στην αύξηση της αποτελεσματικότητας καθώς και στην βελτίωση της αποτελεσματικότητας. Η μεγαλύτερη αποτελεσματικότητα και αποδοτικότητα οδηγούν στην ανάπτυξη της προσαρμοστικότητας σε οποιοσδήποτε νέες συνθήκες.

Συνοπτικά θα μπορούσαμε να πούμε πως κάποια σημαντικά πλεονεκτήματα ενός ERP συστήματος είναι:

- 1) Η διασύνδεση των τμημάτων, των λειτουργιών αλλά και των πληροφοριών μιας επιχείρησης σε ένα ενιαίο περιβάλλον εργασίας
- 2) Η προσφορά κάποιων δυνατοτήτων συντονισμού μεταξύ των διαφόρων υποσυστημάτων της επιχείρησης
- 3) Παρέχει δυνατότητες σύνθεσης γνώσης από τα πολυδιασπασμένα δεδομένα των συναλλαγών
- 4) Εξασφαλίζει μια ολοκληρωμένη και καθαρή εικόνα της επιχειρηματικής δραστηριότητας στην διοίκηση.

ΜΕΡΟΣ ΙΙ.

ΑΣΚΗΣΗ ΠΑΡΑΔΕΙΓΜΑ

5.1 Εκφώνηση άσκησης

Μια επιχείρηση που εμπορεύεται μηχανήματα θέλει να προμηθευτεί ένα λογιστικό πληροφοριακό σύστημα για την κάλυψη των βασικών της λειτουργιών. Η ροή εργασιών έχει ως εξής:

Α.Όσον αφορά τους προμηθευτές

Τμήμα αποθήκης

Με την άφιξη των εμπορευμάτων ο εργαζόμενος της αποθήκης αφού ελέγξει την παραλαβή με το αντίστοιχο παραστατικό προχωρά στην ηλεκτρονική καταχώρηση των στοιχείων του παραστατικού στο λογιστικό πληροφοριακό σύστημα ΑΒΓ και συγκεκριμένα στην διαχείριση αποθήκης, όπου εισάγει στην αντίστοιχη ηλεκτρονική φόρμα τα εμπορεύματα, τον αριθμό τεμαχίων ανά εμπόρευμα και την ημερομηνία παραλαβής. Αφού καταχωρήσει τα στοιχεία αποθηκεύει τη φόρμα και το πρόγραμμα διαχείρισης αποθήκης ενημερώνει αυτόματα το σύστημα προσαυξάνοντας τα εμπορεύματα που καταχωρεί ο εργαζόμενος με τις αντίστοιχες ποσότητες ή εάν πρόκειται για νέο εμπόρευμα δημιουργεί αυτόματα τον αντίστοιχο κωδικό εμπορεύματος και την ποσότητά του. Ενώ με την καταχώρηση αυτή η πληροφορία που εισάγουμε στη διαχείριση αποθήκης του συστήματος είναι προσβάσιμη από όλα τα συνδεδετικά μέρη του λογιστικού πληροφοριακού συστήματος ΑΒΓ.

Τμήμα λογιστηρίου προμηθευτών

Έπειτα το παραστατικό πηγαίνει στο λογιστήριο της επιχείρησης όπου ο υπάλληλος Α του λογιστηρίου που ασχολείται μόνο με τους προμηθευτές προβαίνει στις παρακάτω ενέργειες:

α. Εάν το παραστατικό είναι τιμολόγιο ή τιμολόγιο-δελτίο αποστολής τότε με την παραλαβή του ενημερώνει την εμπορική διαχείριση του λογιστικού πληροφοριακού συστήματος ΑΒΓ καταχωρώντας όλα τα στοιχεία του παραστατικού, τον προμηθευτή, το είδος του παραστατικού, την ημερομηνία παραλαβής, τα εμπορεύματα, τον αριθμό τεμαχίων ανά εμπόρευμα, την καθαρή αξία και τον ΦΠΑ που αναγράφονται στο παραστατικό ανά εμπόρευμα. Έπειτα αρχειοθετεί το παραστατικό στον αντίστοιχο φάκελο προμηθευτή ανά ημερομηνία.

β. Εάν όμως το παραστατικό είναι δελτίο αποστολής τότε ενημερώνει το αρχείο με όνομα “Έκρεμότητες μήνα” με τα στοιχεία του παραστατικού και το βάζει στον φάκελο εκκρεμότητες. Με την άφιξη του τιμολογίου διαγράφει τη καταχώρηση του δελτίου αποστολής από τις “Έκρεμότητες” μήνα και προβαίνει στην καταχώρηση

των στοιχείων στην εμπορική διαχείριση όπως στην περίπτωση α. Έπειτα αρχειοθετεί το επισυναπτόμενο με το δελτίο αποσολής τιμολόγιο στον αντίστοιχο φάκελο προμηθευτή ανά ημερομηνία.

Σχόλιο: το είδος του τιμολογίου δεν επηρεάζει τη διαδικασία που ακολουθεί ο εργαζόμενος της αποθήκης αφού το πρόγραμμα της διαχείρισης αποθήκης πρέπει να ενημερώνεται με κάθε νέα παραλαβή άσχετα με το είδος του παραστατικού.

Τμήμα αγορών

Όταν το τμήμα αποθήκης μένει από κάποιο εμπόρευμα εμφανίζεται μήνυμα στη διαχείριση αποθήκης και στην εμπορική διαχείριση του ΑΒΓ για μηδενικό υπόλοιπο του εμπορεύματος και κωδικό Χ. Έπειτα ο υπάλληλος του τμήματος αγορών διασταυρώνει την πληροφορία με τον υπάλληλο αποθήκης και εάν ισχύει αυτό τότε ο υπάλληλος του τμήματος αγορών προβαίνει σε ηλεκτρονική παραγγελία. Το τμήμα αγορών στέλνει στο τμήμα διοίκησης αναφορά για την αξία των αγορών που έχουμε κάνει από τους προμηθευτές και για τα υπόλοιπα των ωφελών της επιχείρησης προς τους προμηθευτές μέσω της εμπορικής διαχείρισης του προγράμματος ΑΒΓ για κάθε τρίμηνο ως εξής: εισάγοντας την επωνυμία ή τον ΑΦΜ του προμηθευτή, το χρονικό πλαίσιο το οποίο μας ενδιαφέρει και επιλέγοντας αγορές και υπόλοιπο χρεών προς προμηθευτές αντίστοιχα, το ΑΒΓ διαβάσει τις εγγραφές από το αρχείο τιμολογίων ανά πελάτη και εμφανίζει την αναφορά με τα σχετικά φίλτρα που του έχουμε δώσει.

Β. Όσον αφορά τους πελάτες

Τμήμα πωλήσεων και τμήμα αγορών

Ο υπάλληλος του τμήματος πωλήσεων βγάζει προσφορές αγορών για τους πελάτες, συντονίζει τις παραγγελίες και κάνει καταχώρηση των πελατών. Πιο αναλυτικά:

Προσφορά αγορών και συντονισμός παραγγελιών

Αφού ο υπάλληλος πωλήσεων συζητήσει με τον πελάτη και καταλήξουν στα επιθυμητά εμπορεύματα, ο υπάλληλος πωλήσεων προβαίνει σε δημιουργία εντύπου προσφοράς αναλογική με την παλαιότητα και τη συχνότητα αγορών του πελάτη, στοιχεία τα οποία αποκομίζει από τα αντίστοιχα αρχεία στην εμπορική διαχείριση του προγράμματος ΑΒΓ. Αφού ο υπάλληλος πωλήσεων δημιουργήσει το έντυπο προσφοράς, το αποθηκεύει στο σύστημα και δημιουργεί ένα αντίτυπο το οποίο αποστέλει στο τμήμα αγορών.

Το τμήμα αγορών διασταυρώνει την ύπαρξη ή μη των εμπορευμάτων στην

αποθήκη και προβαίνει στις εξής ενέργειες:

α. Εάν τα εμπορεύματα υπάρχουν στην αποθήκη δημιουργεί αναφορά επιβεβαίωσης της προσφοράς, την αποθηκεύει στο σύστημα και δημιουργεί ένα αντίγραφο το οποίο στέλνει στο τμήμα πωλήσεων. Στη συνέχεια ο υπάλληλος του τμήματος πωλήσεων εκτυπώνει το έντυπο προσφοράς και το δίνει στον πελάτη.

β. Ενώ στην περίπτωση που δεν υπάρχουν τα εμπορεύματα στην αποθήκη ο υπάλληλος του τμήματος αγορών προβαίνει σε ηλεκτρονική παραγγελία και έπειτα δημιουργεί αναφορά επιβεβαίωσης προσφοράς με αναμενόμενο χρόνο αναμονής για την άφιξη του εμπορεύματος στην αποθήκη. Έπειτα αποθηκεύει την αναφορά επιβεβαίωσης προσφοράς στο σύστημα και δημιουργεί ένα αντίγραφο το οποίο στέλνει στο τμήμα πωλήσεων. Στη συνέχεια ο υπάλληλος του τμήματος πωλήσεων τροποποιεί το έντυπο προσφοράς αναγράφοντας τον χρόνο αναμονής για άφιξη του εμπορεύματος, το αποθηκεύει, το εκτυπώνει και το δίνει στον πελάτη.

Στην περίπτωση που ο πελάτης δεν θέλει να προβεί σε αγορά τότε μετακινεί το έντυπο προσφοράς στο αρχείο απορριφθέντα. Ενώ στην περίπτωση που ο πελάτης δέχεται τη προσφορά και επιθυμεί να προχωρήσει η παραγγελία διακρίνουμε τις παρακάτω δύο περιπτώσεις:

α. Εάν πρόκειται για νέο πελάτη, τότε ο υπάλληλος του τμήματος πωλήσεων καταχωρεί τον πελάτη στο πρόγραμμα της εμπορικής διαχείρισης του ΑΒΓ εισάγοντας στην αντίστοιχη φόρμα όλα του τα στοιχεία, ΑΦΜ, επωνυμία, διεύθυνση, περιοχή, τ.κ., τηλέφωνο επικοινωνίας και e-mail.

β. Εάν πρόκειται για παλιό πελάτη, τότε ο υπάλληλος του τμήματος πωλήσεων κάνει εύρεση του πελάτη χρησιμοποιώντας τον ΑΦΜ του ή την επωνυμία του στο πρόγραμμα της εμπορικής διαχείρισης του ΑΒΓ, ανοίγει την καρτέλα με τα στοιχεία του πελάτη και έπειτα από προφορική επιβεβαίωση των στοιχείων εάν υπάρχει κάποιο λάθος σε αυτά κάνει μεταβολή των στοιχείων και αποθήκευση των σωστών πληροφοριών, ενώ αν είναι σωστά τότε προβαίνει στο παρακάτω βήμα: Δημιουργεί έντυπο παραγγελίας με τα στοιχεία του πελάτη και με τα εμπορεύματα στις ποσότητες και τις αντίστοιχες αξίες προσφοράς (πληροφορίες που ανακτά από το αποθηκευμένο έντυπο προσφοράς) και το στέλνει στο τμήμα λογιστηρίου πωλήσεων.

Τέλος το τμήμα πωλήσεων στέλνει στο τμήμα διοίκησης αναφορά για την αξία των πωλήσεων προς τους πελάτες και για τα υπόλοιπά τους μέσω της εμπορικής διαχείρισης του προγράμματος ΑΒΓ κάθε τρίμηνο ως εξής: εισάγοντας την επωνυμία ή τον ΑΦΜ του πελάτη, το χρονικό πλαίσιο το οποίο μας ενδιαφέρει και επιλέγοντας πωλήσεις και υπόλοιπο πελάτη αντίστοιχα το ΑΒΓ διαβάσει τις εγγραφές από το

αρχείο τιμολογίων ανά πελάτη και εμφανίζει την αντίστοιχη αναφορά με τα σχετικά φίλτρα που του έχουμε δώσει.

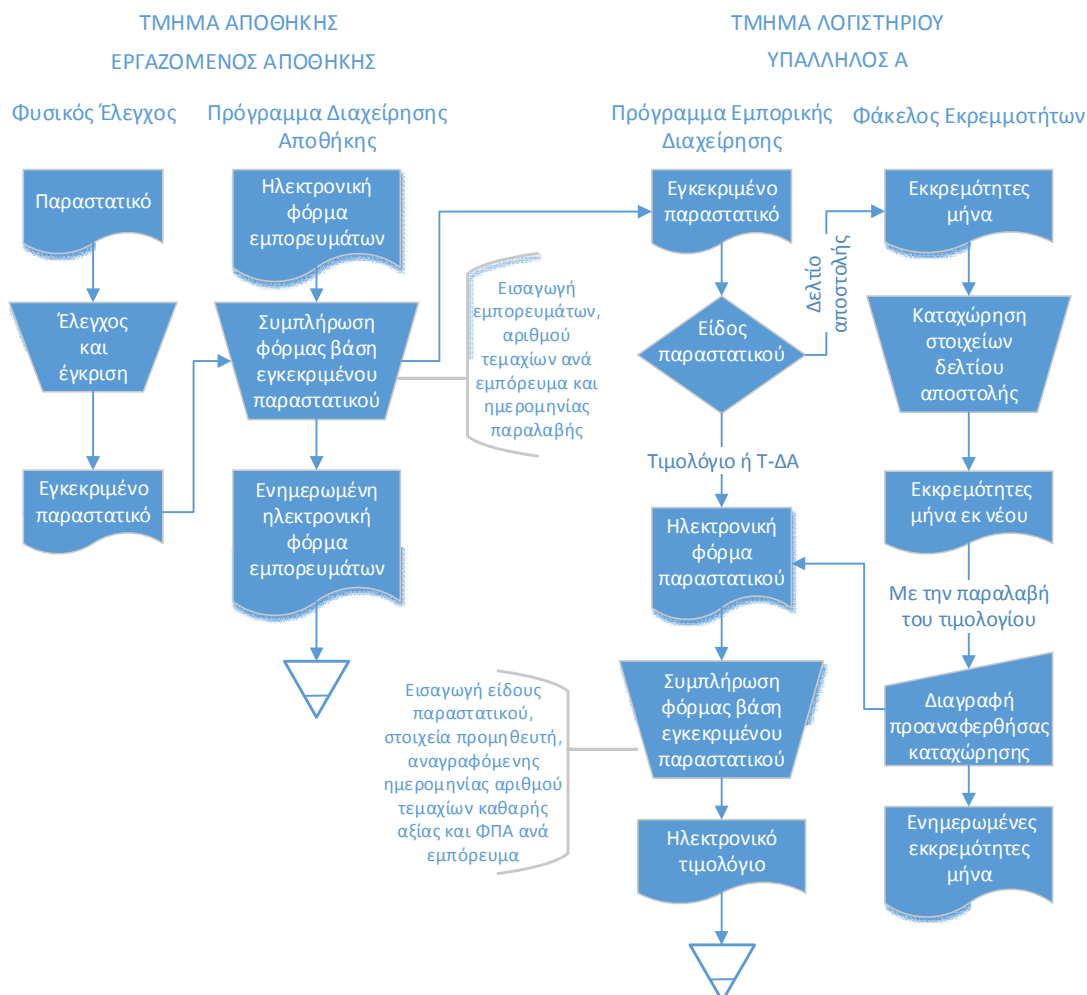
Τμήμα λογιστηρίου πωλήσεων.

Ο υπάλληλος Β του τμήματος λογιστηρίου που ασχολείται μόνο με τις πωλήσεις παραλαμβάνει το έντυπο προσφοράς και εκδίδει το αντίστοιχο παραστατικό το οποίο στέλνει στο τμήμα πωλήσεων. Έπειτα:

α. Εάν πρόκειται για τιμολόγιο ή τιμολόγιο-δελτίο αποστολής καταχωρεί στην εμπορική διαχείριση του προγράμματος ΑΒΓ το παραστατικό που εκδόθηκε με τα στοιχεία του πελάτη, τα εμπορεύματα, στις αντίστοιχες ποσότητες και στις αντίστοιχες αξίες. Έπειτα αρχειοθετεί το παραστατικό στον αντίστοιχο φάκελο εσόδων ανά ημερομηνία.

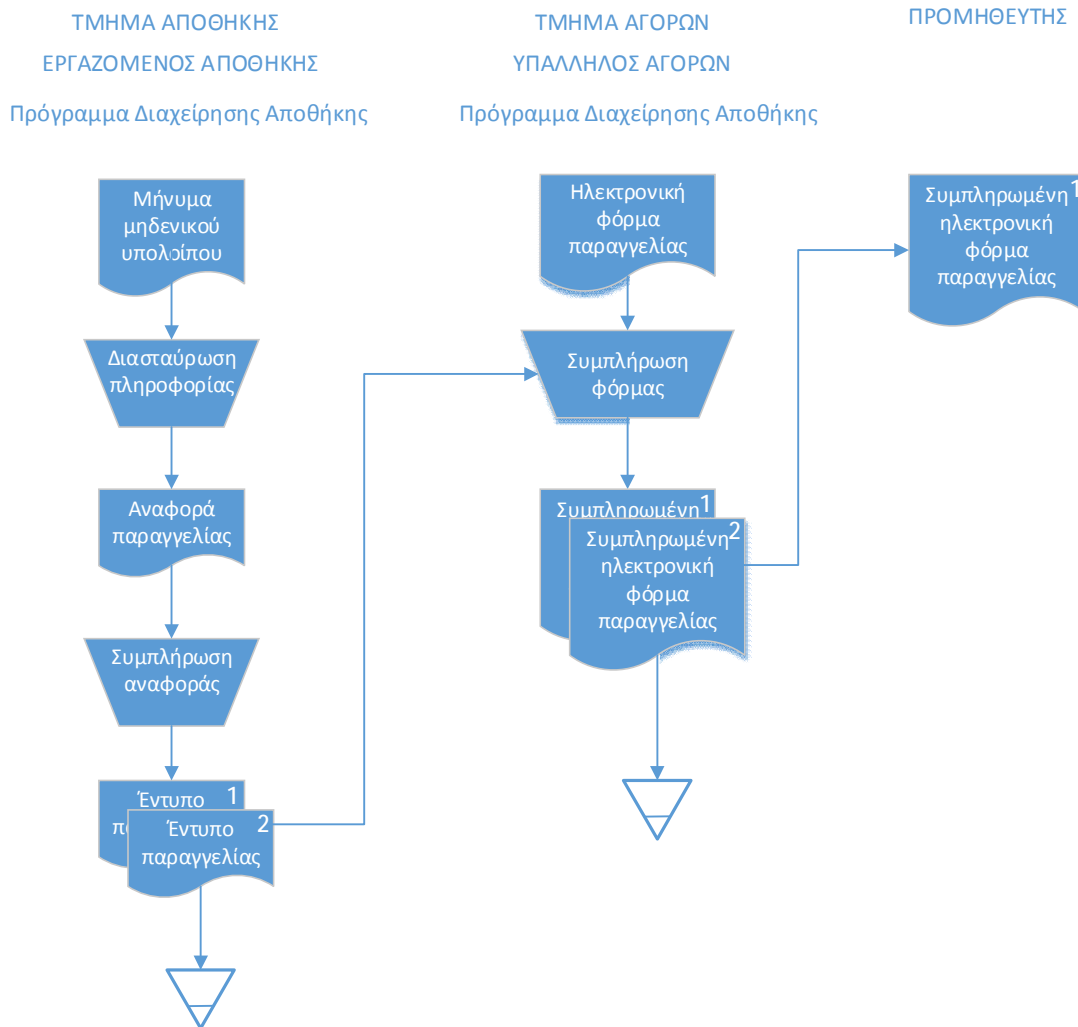
β. Ενώ εάν πρόκειται για δελτίο αποστολής τότε ενημερώνει το αρχείο με όνομα “Εκρεμότητες μήνα” με τα στοιχεία του παραστατικού και το βάζει στον φάκελο εκκρεμότητες. Με την έκδοση του τιμολογίου διαγράφει τη καταχώρηση του δελτίου αποστολής από τις “Εκρεμότητες μήνα” και προβαίνει στην καταχώρηση των στοιχείων στην εμπορική διαχείριση όπως στην περίπτωση α. Έπειτα αρχειοθετεί το επισυναπτόμενο με το δελτίο αποστολής τιμολόγιο στον αντίστοιχο φάκελο εσόδων ανά ημερομηνία.

5.2 Επίλυση Άσκησης



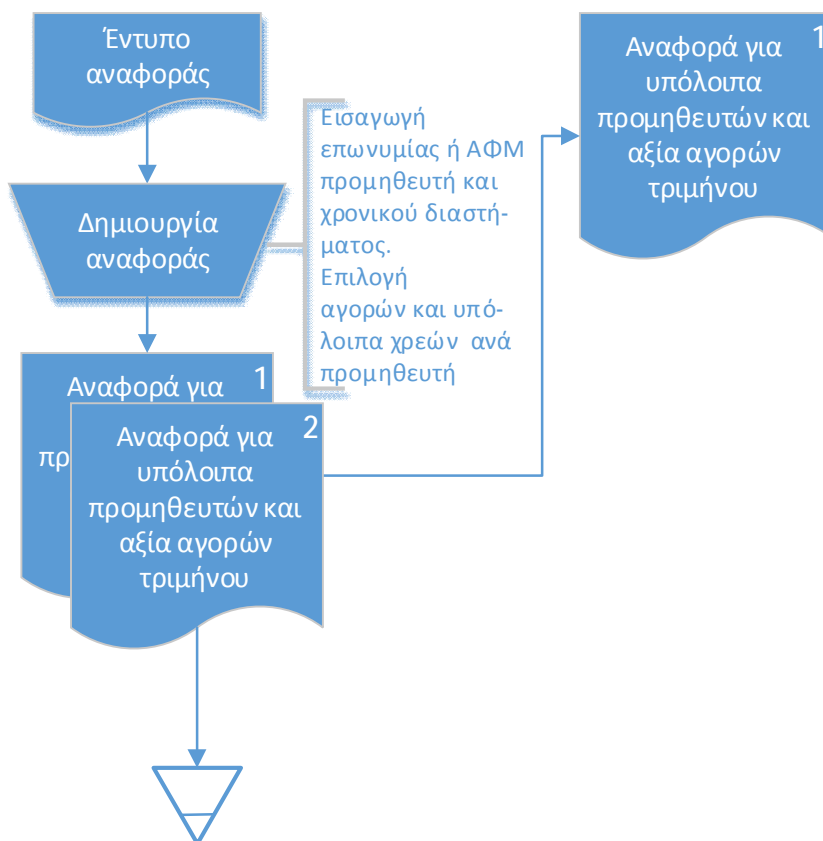
Σχήμα 5.2.1 Αναλυτικό διάγραμμα ροής

Στο αναλυτικό διάγραμμα ροής του σχήματος 5.2.1 απεικονίζεται η αλληλεπίδραση μεταξύ του εργαζόμενου και του υπαλλήλου Α, των τμημάτων αποθήκης και λογιστηρίου αντίστοιχα. Ο εργαζόμενος του τμήματος πραγματοποιεί φυσικό έλεγχο με το παραστατικό στα παραληφθέντα εμπορεύματα, όπως φαίνεται στην πρώτη στήλη. Έπειτα ανοίγει την ηλεκτρονική φόρμα εμπορευμάτων στο πρόγραμμα διαχείρισης αποθήκης, όπου εισάγει τα εμπορεύματα, τον αριθμό των τεμαχίων ανά εμπόρευμα και την ημερομηνία παραλαβής, βάση του εγκεκριμένου παραστατικού, όπως απεικονίζεται στη δεύτερη στήλη. Με την ολοκλήρωση αυτής της διαδικασίας στέλνει το εγκεκριμένο τιμολόγιο στον υπαλληλο Α, ο οποίος ανάμεσα με το είδος του παραστατικού που παρέλαβε καταχωρεί το παραστατικό στο σύστημα όπως απεικονίζεται στο διάγραμμα.



Σχήμα 5.2.2 Αναλυτικό διάγραμμα ροής

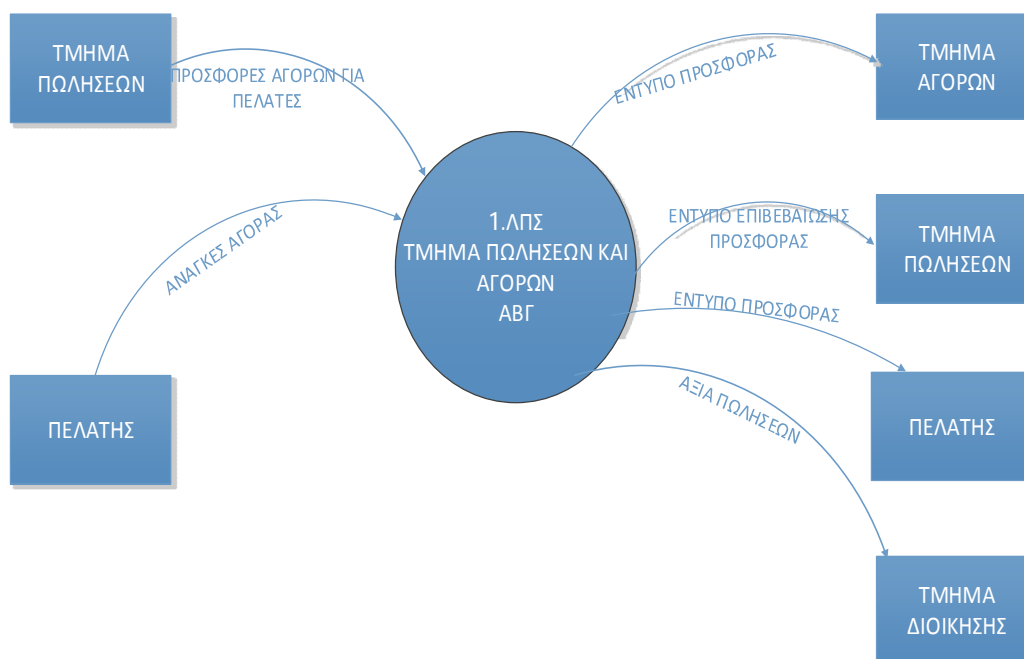
Το αναλυτικό διάγραμμα ροής του σχήματος 5.2.2 έχουμε τρεις συμμετέχοντες, τον εργαζόμενο αποθήκης τον υπάλληλο του τμήματος αγορών και τον προμηθευτή. Με την εμφάνιση του μηνύματος για μηδενικό υπόλοιπο εμπορεύματος ο εργαζόμενος του τμήματος αποθήκης αφού διασταυρώσει την πληροφορία δημιουργεί δύο αντίτυπα αναφοράς παραγγελίας. Το ένα αντίτυπο το αποθηκεύει και το άλλο το στέλνει στον υπάλληλο του τμήματος αγορών, ο οποίος με τη σειρά του το χρησιμοποιεί για να δημιουργήσει την ηλεκτρονική φόρμα παραγγελίας σε δύο αντίτυπα. Το ένα το αποθηκεύει στο σύστημα και το άλλο το στέλνει στον προμηθευτή.



Σχήμα 5.2.3 Αναλυτικό διάγραμμα ροής

Στο αναλυτικό διάγραμμα ροής του σχήματος 5.2.3 δημιουργεί δύο αντίτυπα αναφοράς, με περιεχόμενο τα υπόλοιπα των προμηθευτών και τις αξίες αγορών του τριμήνου, από τα οποία το ένα το αποθηκεύει και το άλλο το στέλνει στο τμήμα της διοίκησης, μέσω του προγράμματος της εμπορικής διαχείρισης.

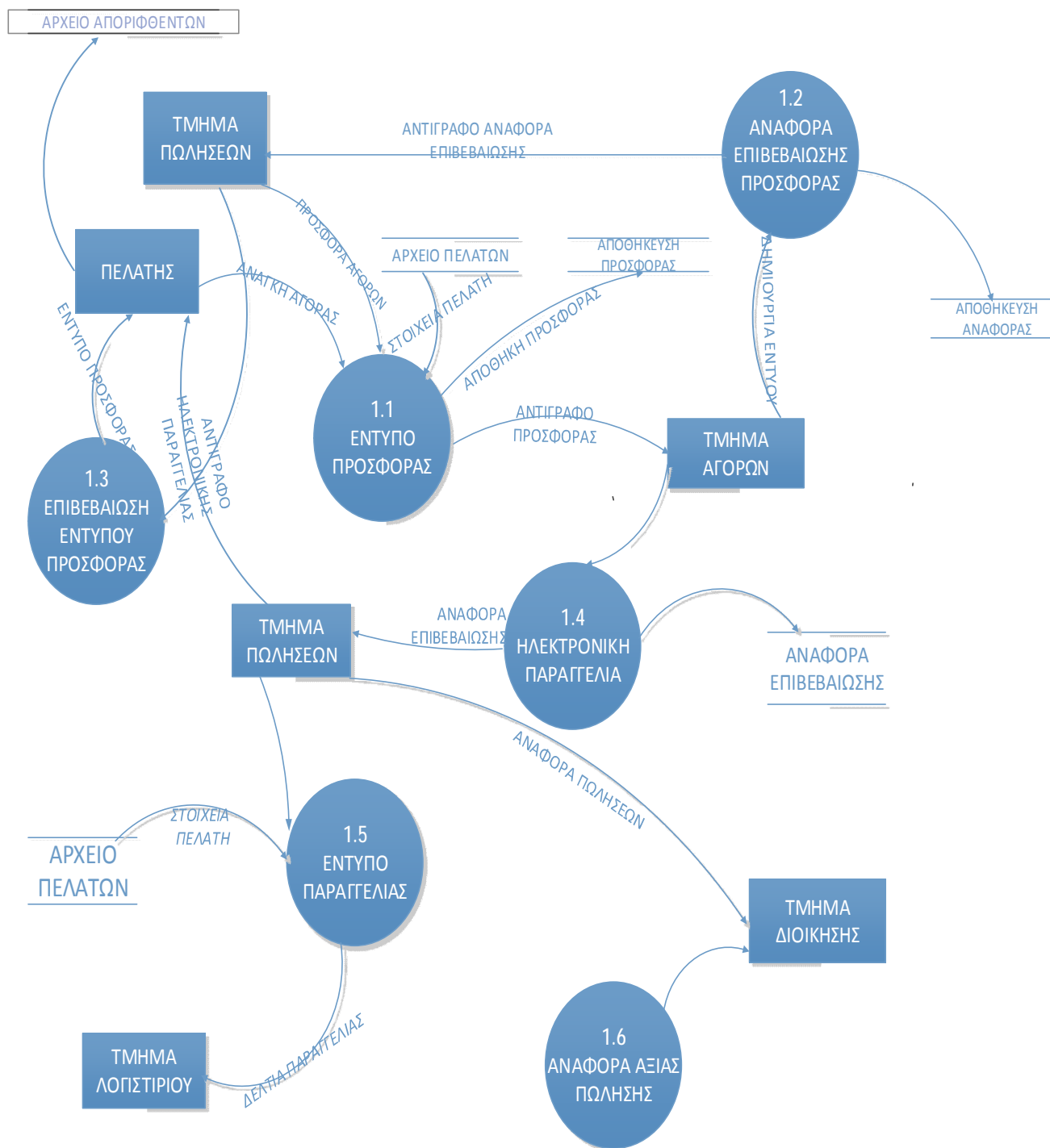
ΜΗΔΕΝΙΚΟ ΕΠΙΠΕΔΟ



Σχήμα 5.2.4 Διάγραμμα ροής δεδομένων μηδενικού επιπέδου

Στο πρωταρχικό μοντέλο, βλέπουμε το τμήμα πωλήσεων να δίνει στη βάση μας ΑΒΓ τις προσφορές αγορών των πελατών. Επίσης, τις ανάγκες για αγορά που έχει ο πελάτης και πώς αυτές τις δύο πληροφορίες τις επεργάζεται το ΑΒΓ. Έπειτα, μέσω της επεξεργασίας τους, προκύπτει το έντυπο προσφοράς που στέλνεται στο τμήμα αγορών, η επιβεβαίωση της προσφοράς που συνδέεται με το τμήμα πωλήσεων, το έντυπο προσφοράς το οποίο στέλνεται στον πελάτη και έχει διαμορφωθεί ανάλογα των αναγκών του και τέλος η αξία πωλήσεων που ενημερώνει το τμήμα διοίκησης.

1ο ΕΠΙΠΕΔΟ

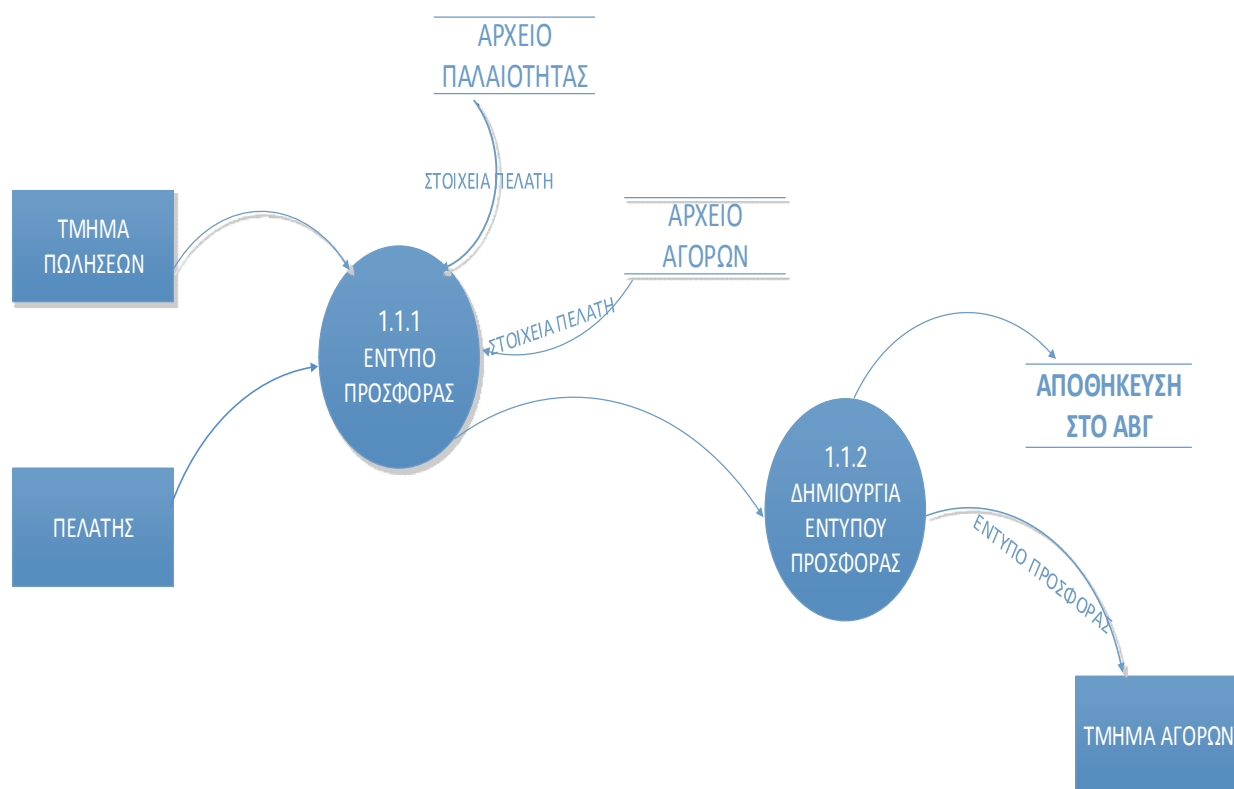


Σχήμα 5.2.5 Διάγραμμα ροής δεδομένων πρώτου επιπέδου ανάλυσης

Στο πρώτο επίπεδο ανάλυσης γίνεται κατανοητός ο τρόπος πραγματοποίησης των συνδέσεων. Πιο αναλυτικά το τμήμα πωλήσεων συνδέεται με το 2.1 που του παραχωρεί τη πληροφορία της προσφοράς αγορών. Το 2.1 ενημερώνεται με τα

στοιχεία του πελάτη και τις ανάγκες του και στέλνει το αντίγραφο προσφοράς στο τμήμα αγορών το οποίο με τη σειρά του επιβεβαιώνει την προσφορά αν υπάρχουν τα εμπορεύματα και στέλνεται πίσω στο τμήμα πωλήσεων έπειτα γίνεται η επιβεβαίωση εντύπου προσφοράς 2.3 το οποίο και στέλνεται στο πελάτη, αν δεν το δεχτεί πηγαίνει στα απορριφθέντα. Σε περίπτωση που οι ανάγκες του πελάτη δεν μπορούν να ικανοποιηθούν λόγω έλλειψης των εμπορευμάτων τότε γίνεται ηλεκτρονική η παραγγελία τους και στέλνεται η αναφορά επιβεβαίωσης στο τμήμα πωλήσεων το οποίο με τη σειρά του στέλνει το έντυπο στον πελάτη. Αν δεν το δεχτεί μεταφέρεται στα απορριφθέντα. Αν το δεχτεί δημιουργείται το έντυπο παραγγελίας και στέλνεται το δελτίο στο λογιστήριο κι έπειτα ενημερώνεται το τμήμα διοίκησης με την αξία της πώλησης.

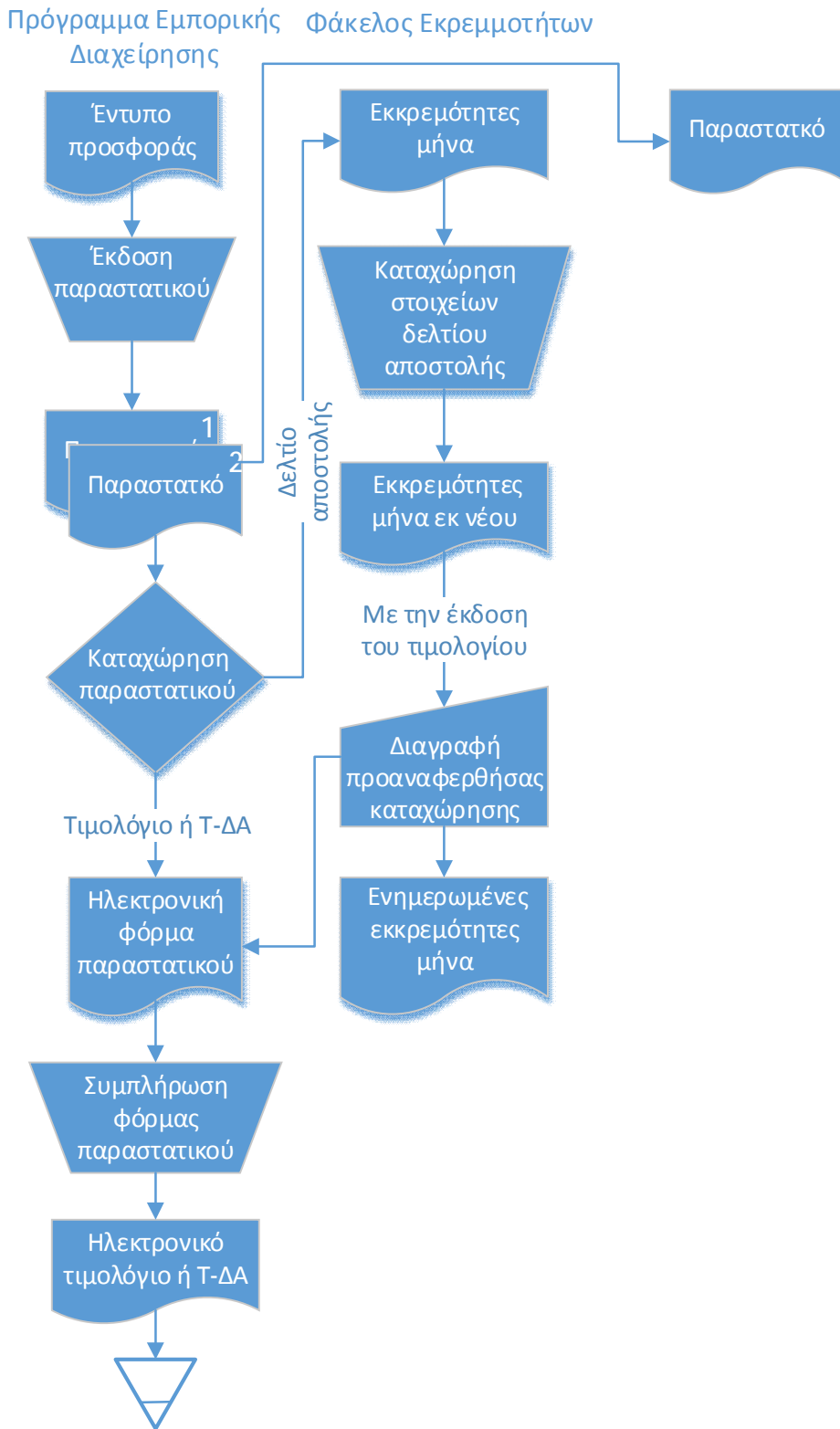
2ο ΕΠΙΠΕΔΟ



Σχήμα 5.2.6 Διάγραμμα ροής δεδομένων δεύτερου επιπέδου ανάλυσης

Στο δεύτερο επίπεδο αναλύουμε τη διαδικασία εκπόνησης προσφοράς από το τμήμα πωλήσεων. Αφού γίνει συμφωνία ανάμεσα στον υπάλληλο και στον πελάτη για τα εμπορεύματα που τον ενδιαφέρουν δημιουργείται το έντυπο προσφοράς με τα στοιχεία του πελάτη ανάλογα με την παλαιότητα και την συχνότητα αγορών του

πελάτη. Αφού γίνει η δημιουργία του εντύπου προσφοράς αποθηκεύεται στην εμπορική διαχείριση του προγράμματος ΑΒΓ. Και τέλος, το έντυπο προσφοράς στέλνεται στο τμήμα αγορών για να το ενημερώσει.



Σχήμα 5.2.7 Αναλυτικό διάγραμμα ροής

Στο αναλυτικό διάγραμμα ροής του σχήματος 5.2.7 ο υπάλληλος Β του τμήματος λογιστηρίου παραλαμβάνει το έντυπο προσφοράς για πώληση εμπορευμάτων, βάση του οποίου εκδίδει το αντίστοιχο παραστατικό σε δύο αντίτυπα. Το ένα το στέλνει στο τμήμα πωλήσεων, και το άλλο το χρησιμοποιεί για να το καταχωρήσει στο σύστημα. Ανάλογα με το είδος του παραστατικού προβαίνει στον ανάλογο τρόπο καταχώρησης όπως φαίνεται στο αναλυτικό διάγραμμα ροής.

Βιβλιογραφία

Κεφάλαιο 1ο

1. Γεωργίου Γεώργιος και Βαίτσης Βασίλειος, με θέμα πτυχικής εργασίας για το ΤΕΙ Ηπείρου: **Ο ρόλος και η σημασία των Λογιστικών Πληροφοριακών Συστημάτων στο Σύγχρονο Χρηματοοικονομικό Περιβάλλον**, Πρέβεζα 2014.
2. Φραγκιαδάκη Γεωργία και Άνδρης Γεώργιος, με θέμα πτυχιακής εργασίας: **Το περιβάλλον στην επιχείρηση, η επίδραση του Μίκρο - Μάκρο περιβάλλον στις σύγχρονες επιχειρήσεις**, Ηράκλειο 2010.
3. Καρυπίδης Μιχαήλ, για το ΤΕΙ Θεσσαλονίκης: **Σημειώσεις θεωρίας ηλεκτρονικών συστημάτων παραγωγής ενδυμάτων**, Κιλκίς 2009.
4. Κονταρίδου Ελένη, για το Πανεπιστήμιο Μακεδονίας: **Κριτήρια επιλογής λογιστικών πληροφοριακών συστημάτων**, Διπλωματική Εργασία, Θεσσαλονίκη 2010.
5. Χριστιάνα Γεωργίου Κυριαζοπούλου, με θέμα διπλωματικής εργασίας για το Πανεπιστήμιο Μακεδονίας: **Κίνδυνοι και Έλεγχοι των Λογιστικών Πληροφοριακών Συστημάτων**, Θεσσαλονίκη 2012.
6. Καρδιασμένος Γεώργιος, με θέμα διπλωματικής εργασίας του τμήματος Βιομηχανική Διοίκησης και Τεχνολογίας: **Συστήματα Διαχείρισης Επιχειρησιακών Πόρων (ERP)–Αποθηκών (WMS) σε εταιρία παροχής υπηρεσιών, προϊόντων τηλεπικοινωνιών και αυτόνομης ενέργειας**, Πειραιάς 2008.
7. Μπουράντας Δημήτριος, Βάθης Άγγελος, Ρεκλείτης Παναγιώτης και Παπακωνσταντίνου Χρήστος, **Αρχές Οργάνωσης και Διοίκησης Επιχειρήσεων και Υπηρεσιών**, Αθήνα 1999.
8. Μαντζάρης Γιάννης, Copy City Ε.Π.Ε, **Οργάνωση και Διοίκηση Επιχειρήσεων**, Σέρρες 2011.
9. Τζωρτζάκη Αλεξία-Μαίρη και Τζωρτζάκης Κώστας, **Οργάνωση και Διοίκηση**, Αθήνα 2007.
10. Κοίλιας Χρήστος, Δημητριάδης Αντώνης, και Κώστας Αθανάσιος, **Λογιστικά Πληροφοριακά Συστήματα**, Αθήνα 2009.

Κεφάλαιο 2ο, 3ο & 4ο

1. Δρ. Β. Τ. Ταμπακάς, **Σημειώσεις για το μάθημα εισαγωγή στα λογιστικά πληροφοριακά συστήματα (θεωρία)**, Πάτρα 2013.
2. Βασίλειος Βεσκούκης, **Τεχνολογία λογισμικού I**, Πάτρα 2000.
3. Γιακουμάκης Ε., **Τεχνολογία Λογισμικού Τόμος Α**, Εκδόσεις Α. Σταμούλη, Αθήνα 1993.
4. Γιακουμάκης Ε., **Τεχνολογία Λογισμικού Τόμος Β**, Εκδόσεις Α. Σταμούλη, Αθήνα 1994.
5. Κιουντούζης Ε., **Μεθοδολογίες Ανάλυσης και Σχεδιασμού Πληροφοριακών συστημάτων**, Εκδόσεις Ε. Μπένου, Αθήνα 1997.

6. Avison D. and Fitzgerald G., **Information Systems Development: Methodologies, Techniques and Tools**, 3rd Edition, McGraw–Hill Education, UK, 2003.
7. Boewen K., “**A conflict approach to the modeling of problems of and in organizations in Operation Research**”, J. Brans (Editor), IFORS 1981, North – Holland, Amsterdam, 1981.
8. Hamilton, M., Zeldin, S., "**Higher order software--A methodology for defining software**", **IEEE Transactions on Software Engineering**, SE-2, 1, pp. 9-32, March 1976.
9. Martin J. and Mc Clure C., **Structured Techniques for Computing**, Prentice Hall, London, 1985.
10. Wilson B., **Systems concepts, methodologies and applications**, John Wiley and Sons, Chichester, England, 1984.
11. Γιάννης Α Πολλάλης, Αθανάσιος Π.Βοζίκης, **Πληροφοριακά συστήματα διαχείρισης επιχειρησιακών πόρων: Στρατηγικές και εφαρμογές.**

Ιστότοποι 3ου κεφαλαίου

<http://www.psounis.gr/%CF%80%CE%BB%CE%B710/%CF%80%CE%BB%CE%B710-%CE%B5%CE%BD%CF%8C%CF%84%CE%B7%CF%84%CE%B1-1-%CE%B5%CE%B9%CF%83%CE%B1%CE%B3%CF%89%CE%B3%CE%AE-%CF%83%CF%84%CE%BF%CF%85%CF%82-%CE%B7%CF%85/%CF%80%CE%BB%CE%B710-%CE%BC%CE%AC%CE%B8%CE%B7%CE%BC%CE%B1-1-4-%CF%84%CE%BF-%CE%B4%CE%B9%CE%AC%CE%B3%CF%81%CE%B1%CE%BC%CE%BC%CE%B1-%CF%81%CE%BF%CE%AE%CF%82-flowchart/>

https://el.wikipedia.org/wiki/%CE%94%CE%B9%CE%AC%CE%B3%CF%81%CE%B1%CE%BC%CE%BC%CE%B1_%CF%81%CE%BF%CE%AE%CF%82

Ιστότοποι 4ου κεφαλαίου

<http://www.plant-management.gr/index.php?id=43#top>

http://repository.library.teimes.gr/xmlui/bitstream/handle/123456789/1895/EPDO_0037.pdf?sequence=1&isAllowed=y

<http://www.plant-management.gr/index.php?id=43#top>