



ΑΡΙΣΤΟΤΕΛΕΙΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΟΝΙΚΗΣ

ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΕΦΗΣ ΑΝΑΛΥΣΗ

Διαγράμματα Κλάσεων

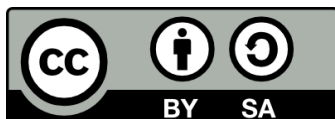
Ιωάννης Σταμέλος

Βάιος Κολοφωτιάς

Πληροφορική

Άδειες Χρήσης

Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons. Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα. Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.



Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Περιεχόμενα

Άδειες Χρήσης.....	2
Χρηματοδότηση.....	2
1. Περιεχόμενο Μαθήματος	4
2. Περιεχόμενα ενότητας.....	4
3. Διαγράμματα κλάσεων(Class Diagrams).....	5
3.1 Λειτουργίες των κλάσεων.....	6
3.1.1 Ορατότητα (Visibility)	6
3.1.2 Σχέσεις (Relationships)	7
3.1.3 Συσχετίσεις (Associations).....	7
3.1.4 Εξαρτήσεις(dependencies)	8
3.1.5 Γενικεύσεις(generalizations)	8
3.1.6 Συναθροίσεις(aggregations).....	8
3.1.7 Συνθέσεις (Compositions)	8
3.2 Είδη Κλάσεων.....	9
3.2.1 Πως επιλέγουμε υποψήφιες κλάσεις;	9
3.2.2 Πως επιλέγουμε οριστικά κλάσεις;	11
4. ΑΣΚΗΣΕΙΣ	13
4.1 ATM Πρώτη ανάλυση κλάσεων	13
4.1.1 Κάρτες Κλάσης – Υποχρέωσης - Συνεργασίας (CRC).....	14
4.1.2 Διάγραμμα Κλάσεων Αναλυτικά	19

1. Περιεχόμενο Μαθήματος

Εβδομάδα	Περιεχόμενο
1 ^η	Εισαγωγή στην Αντικειμενοστρεφή Ανάλυση/UML
2 ^η	Rational Unified Process
3 ^η	Περιπτώσεις Χρήσης
4 ^η	Διαγράμματα Κλάσεων
5 ^η	Διαγράμματα Συνεργασίας
6 ^η	Διαγράμματα Ακολουθίας
7 ^η	Πρότυπα Σχεδίασης
8 ^η	Διεργασία ICONIX
9 ^η	Επιχειρηματική Μοντελοποίηση
10 ^η	Υλοποίηση Σχεδίασης με Java
11 ^η	Μετρικές Αντικειμενοστραφούς Σχεδίασης
12 ^η	Επισκόπηση

2. Περιεχόμενα ενότητας

Σε αυτή την ενότητα θα δούμε τις κλάσεις του αντικειμενοστραφούς προγραμματισμού και τα διαγράμματα τους, τα οποία απεικονίζουν μια στατική δομή τους. Θα αναλύσουμε τις ιδιότητες αλλά και τις λειτουργίες των κλάσεων, όπως η ορατότητα, οι σχέσεις, οι συσχετίσεις, οι συναθροίσεις και οι συνθέσεις (με τα αντίστοιχα γραφικά τους στα διαγράμματα). Θα επιλέξουμε τις κατάλληλες κλάσεις για το διάγραμμα μας αποκαλώντας τις συνοριακές, ελέγχου και οντοτήτων. Παραθέτονται και δύο βοηθητικές ασκήσεις.

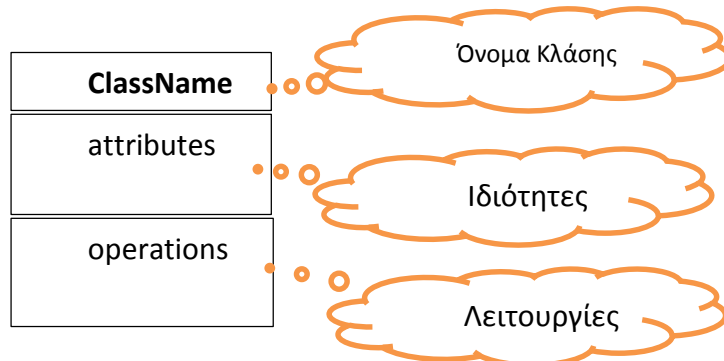
3. Διαγράμματα κλάσεων(Class Diagrams)

Μια κλάση είναι ένα πρότυπο για ένα σύνολο αντικειμένων με παρόμοιες:

- ιδιότητες (attributes),
 - λειτουργίες (operations),
 - σχέσεις (relationships) και
 - σημασιολογία - όρους συμβολισμού(semantics).
- Τα αντικείμενα μιας κλάσης έχουν κατάσταση (state) και συμπεριφορά (behavior).
 - Τα διαγράμματα κλάσεων απεικονίζουν την στατική δομή του συστήματος. Στην δομή περιλαμβάνονται: Οι κλάσεις (classes) και οι σχέσεις (relationships)

Οι κλάσεις είναι το σημαντικότερο δομικό στοιχείο του Αντικειμενοστραφούς Προγραμματισμού.

- Οι κλάσεις χρησιμοποιούνται :
 - σαν λεξικό του συστήματος που αναπτύσσουμε
 - για να αναπαράσταση στοιχείων του λογισμικού, του υλικού
- Στην UML οι κλάσεις απεικονίζονται με ένα τετράγωνο τριών τμημάτων.



Οι κλάσεις περιλαμβάνουν:

- Τις **ιδιότητες** (*attributes*) που είναι τα δεδομένα μιας τάξης
- Τις **λειτουργίες** (*operations*), δηλαδή τις λειτουργίες κλίσης των άλλων αντικειμένων
- Τις **υπευθυνότητες** (*responsibilities*) που είναι συμβόλαια ή υποχρεώσεις της κλάσης

Μια κλάση μπορεί να έχει ιδιότητες ή να μην έχει:

- Οι ιδιότητες είναι κοινές για όλα τα αντικείμενα της κλάσης
- Σε οποιαδήποτε χρονική στιγμή ένα αντικείμενο της κλάσης θα έχει συγκεκριμένες τιμές για κάθε μια ιδιότητα της κλάσης
- Μπορεί να φαίνεται το όνομα της ιδιότητας, ο τύπος των δεδομένων, αλλά και αρχική τιμή σε όλες ή κάποιες από αυτές.

3.1 Λειτουργίες των κλάσεων

- Λειτουργία είναι η υλοποίηση μιας υπηρεσίας που απαιτείται από ένα αντικείμενο της κλάσης
- Η λειτουργία επιδρά στην συμπεριφορά του αντικειμένου και συχνά οδηγεί στην αλλαγή των δεδομένων ή της κατάστασης του
- Μια κλάση μπορεί να έχει ή να μην έχει καθόλου λειτουργίες

Στις λειτουργίες μπορεί να αναγράφεται:

- μόνο το όνομα της λειτουργίας
- το όνομα και η υπογραφή (signature), δηλαδή ο τύπος και οι αρχικές τιμές όλων των παραμέτρων και στις περιπτώσεις των συναρτήσεων ο τύπος του επιστρεφόμενου αποτελέσματος.

3.1.1 Ορατότητα (Visibility)

Η ορατότητα ορίζει αν μια ιδιότητα ή λειτουργία είναι ορατή, δηλαδή μπορεί να προσπελασθεί από άλλες κλάσεις

Η UML παρέχει τέσσερα σύμβολα για την ορατότητα :

+ δημόσια ορατότητα (public). Προσπέλαση και από άλλες κλάσεις.

- ιδιωτική ορατότητα (private). Προσπέλαση μόνο μέσα στην ίδια κλάση.

προστατευμένη ορατότητα (protected). Προσπέλαση έξω από το πακέτο αλλά από υποκλάσεις της τρέχουσας (κληρονομικότητα)

~ ορατότητα πακέτου (package). Προσπέλαση μόνο από το γκρουπ των κλάσεων του πακέτου

3.1.2 Σχέσεις (Relationships)

Μία σχέση είναι μία σύνδεση μεταξύ πραγμάτων

Πέντε είναι οι πιο σημαντικές σχέσεις στην αντικειμενοστρεφή τεχνολογία λογισμικού οι :

- Συσχετίσεις (associations)
- Εξαρτήσεις (dependencies)
- Γενικεύσεις (generalizations)
- Συναθροίσεις (aggregations)
- Συνθέσεις (Compositions)

Διαφορετικές γραμμές και βέλη χαρακτηρίζουν τις σχέσεις

3.1.3 Συσχετίσεις (Associations)

- Μία δομική σχέση που καθορίζει ότι αντικείμενα μιας κλάσης συνδέονται με αντικείμενα της ίδιας ή άλλης κλάσης
- Γραφικά απεικονίζονται με συμπαγείς γραμμές
- Μία συσχέτιση έχει όνομα που γράφεται σαν ετικέτα στο μέσο της γραμμής σύνδεσης (ρήμα). Το όνομα φανερώνει το είδος της συσχέτισης.
- Μία κλάση που μετέχει σε μία συσχέτιση παίζει ένα ρόλο (*role*) σε αυτήν. Γράφουμε το όνομα του ρόλου (ουσιαστικό) στο τέλος της σχέσης, δηλαδή στο σημείο σύνδεσης με την κλάση.


Πολλαπλότητα (multiplicity)

- Καθορίζει πόσα αντικείμενα μετέχουν σε μια συσχέτιση (από 0 έως οποιοδήποτε πλήθος).
- Ακριβώς ένα **1**
- Μηδέν ή ένα **0..1**
- Μηδέν ή περισσότερα **0..***
- Ένα ή περισσότερα **1..***
- Από..Έως **m..n**
- Πολλαπλά διαφορετικά όρια **2, 4..6,**


3.1.4 Εξαρτήσεις(dependencies)

- Εξάρτηση είναι σχέση “χρήσης” δύο κλάσεων, δηλαδή μια αλλαγή στη χρήση της μιας από αυτές μπορεί να επιφέρει αλλαγές και στην άλλη
- Γραφικά απεικονίζεται με διακεκομμένη γραμμή
- Ένα στερεότυπο χρησιμοποιείται για να δείξει το είδος της εξάρτησης


3.1.5 Γενικεύσεις(generalizations)

- Γενίκευση είναι η σχέση όπου μία κλάση (συνήθως υποκλάση ή “κλάση παιδί”) κληρονομεί τις λειτουργίες και τα δεδομένα από μία γενικότερη κλάση (υπερκλάση ή “κλάση γονέα”)
- Η σχέση αυτή είναι τύπου : “είδος της...” ή “είναι...”
- Γραφικά απεικονίζεται με βέλος στερεάς γραμμής και μεγάλη αιχμή που δείχνει τον γονέα 

3.1.6 Συναθροίσεις(aggregations)

- Συναθροίση είναι μία σχέση τύπου “όλου / μερών...” ή “έχει...” όπου μία μεγάλη κλάση αποτελείται από μικρότερες κλάσεις
- Γραφικά απεικονίζεται με βέλος αιχμής σχήματος διαμαντιού 

3.1.7 Συνθέσεις (Compositions)

- Είναι μια ισχυρότερη σχέση τύπου συσσωμάτωσης (όλου / μερών...) μόνο που τα μέρη θεωρούνται αχώριστα από το όλο.
- Γραφικά απεικονίζεται με βέλος αιχμής σχήματος διαμαντιού (γεμάτου) 

3.2 Είδη Κλάσεων

Κλάσεις οντοτήτων: αναπαριστούν οντότητες (απτές ή αφηρημένες) του πραγματικού κόσμου

Συνοριακές κλάσεις (boundary classes): αναλαμβάνουν την επικοινωνία μίας κλάσης οντοτήτων με τον εξωτερικό κόσμο (χρήστη, άλλες κλάσεις κλπ) – αρχικά, συνήθως έχουμε μία Σ.Κ. για κάθε Κ.Ο.

Κλάσεις ελέγχου (control classes): αναλαμβάνουν τον έλεγχο της ροής εργασιών – αρχικά συνήθως έχουμε μία για κάθε περίπτωση χρήσης

3.2.1 Πως επιλέγουμε υποψήφιες κλάσεις;

Για το διάγραμμα κλάσεων ενδιαφερόμαστε κυρίως για κλάσεις οντοτήτων. Εξετάζουμε:

- Γεγονότα ή πράγματα τα οποία το σύστημα πρέπει να “θυμάται”.
- Ένα σύστημα καταγραφής οχημάτων πρέπει να θυμάται (δηλ. να κρατάει πληροφορίες) για την απόκτηση ενός οχήματος από ένα φυσικό πρόσωπο (π.χ. ημερομηνία, τύπος αποδεικτικού εγγράφου)
- Ένα σύστημα που παρακολουθεί ένα πυρηνικό αντιδραστήρα θα πρέπει να καταγράφει σχολαστικά κάθε στοιχείο ενός ατυχήματος.
- Θέση που κατέχουν ή ρόλος που παίζουν οι άνθρωποι που εμπλέκονται στο σύστημα. Κλάσεις που παριστάνουν ανθρώπους είναι συνήθως δυο ειδών:
- οι χρήστες του συστήματος καταγραφής οχημάτων (π.χ. ο υπάλληλος που καταγράφει τα οχήματα)
- οι μη χρήστες (π.χ. οι ιδιοκτήτες των οχημάτων) για τους οποίους όμως το σύστημα επίσης θα πρέπει να κρατάει πληροφορίες.
- Οντότητες σχετικά με οργανωτικά θέματα.

Κάποιοι άνθρωποι που ενδιαφέρονται για το σύστημα μπορεί να ανήκουν σε συγκεκριμένες υπηρεσίες (Δήμος, Υπουργείο, Εταιρία) για τις οποίες χρειάζεται να αποθηκευθεί πληροφορία (π.χ. το όνομα του διευθυντή, ταχυδρομική διεύθυνση, ποσό που εισέπραξε ο υπάλληλος για λογαριασμό της υπηρεσίας).

- Ύπαρξη σχέσεων μεταξύ κλάσεων.
Συχνά εντοπίζουμε ιδιαίτερα χαρακτηριστικά και λειτουργίες σε ειδικές περιπτώσεις κλάσεων (κληρονομικότητα).

- **Ύπαρξη άλλων συστημάτων.**
Τα εξωτερικά συστήματα/περιφερειακές συσκευές (Χειριστές ή μη) με τα οποία επικοινωνεί το σύστημα που εξετάζεται οδηγούν επίσης στον ορισμό Κλάσεων
- **Συσκευές.**
Οι εξειδικευμένες συσκευές (αισθητήρες, sensors) με τις οποίες συνομιλεί το σύστημα απαιτούν Κλάσεις για την αναπαράστασή τους. Ο αναλυτής θα πρέπει να αποφύγει την πρόωρη προδιαγραφή λεπτομερειών σχετικά με συσκευές (όπως μονάδες δίσκων, οθόνες, κλπ) που θα πρέπει να μετατεθεί στη φάση της σχεδίασης.
- **Βήματα Διεργασιών.** Κατά την λειτουργία του συστήματος μπορεί να απαιτούνται κάποια βήματα εργασίας, στα οποία το σύστημα πρέπει να οδηγεί τον χρήστη. Χαρακτηριστικά σημεία τέτοιων Κλάσεων είναι το όνομα της διεργασίας, το επίπεδο εξουσιοδότησης και μια περιγραφή των βημάτων.
- **Τοπολογικές πληροφορίες.** Σε ένα σύστημα που θα εγκατασταθεί σε διάφορα σημεία μπορεί να απαιτείται μία Κλάση Τοποθεσία/Εγκατάστασης με Ιδιότητες γεωγραφικό πλάτος και μήκος, υψόμετρο κλπ.
- **Κλάσεις που αντιπροσωπεύουν αποδέκτες ή γεννήτριες δεδομένων.** Πρόκειται για Κλάσεις που είτε δημιουργούν δεδομένα (π.χ. μια γεννήτρια τυχαίων αριθμών), είτε δέχονται και επεξεργάζονται δεδομένα (π.χ. μια Κλάση που γράφει δεδομένα στον δίσκο ή σε αρχεία).
- **Κλάσεις που παρουσιάζουν ή παρατηρούν δεδομένα.**
 - Συνήθως, ένα αρκετά σημαντικό τμήμα ενός προγράμματος ασχολείται με την παρουσίαση πληροφορίας σε κάποια συσκευή εξόδου (π.χ. μια οθόνη).
 - Ο κώδικας αυτός είναι ιδιαίτερα πολύπλοκος, υπόκειται σε συχνές αλλαγές και είναι σχετικά ανεξάρτητος από τα δεδομένα που παρουσιάζονται.
 - Είναι καλή πρακτική να χρησιμοποιούνται δυο Κλάσεις, μια που κρατά τα κάθε αυτό δεδομένα και μια που αναλαμβάνει την παρουσίασή τους.
 - Συνήθως η πρώτη λέγεται model και η δεύτερη view. Αυτή η τακτική διευκολύνει την επαναχρησιμοποίηση του κώδικα της κλάσης model.

- **Υποβοηθητικές Κλάσεις.** Πρόκειται για Κλάσεις που βοηθούν την εκτέλεση πολύπλοκων λειτουργιών. Για παράδειγμα, στη σχεδίαση μιας καρτέλας, μπορούμε να χρησιμοποιήσουμε Κλάσεις που σχεδιάζουν μια γραμμή ή γράφουν στην οθόνη μια γραμμή κειμένου. Μια άλλη Κλάση μπορεί να χειρίζεται συνδεδεμένες λίστες από καρτέλες.
- Όσον αφορά την διαπροσωπεία ανθρώπου - μηχανής μπορεί είτε να διαχυθεί σε όλο το διάγραμμα Κλάσεων είτε να αναπαρασταθεί με μια συγκεκριμένη Κλάση.

3.2.2 Πως επιλέγουμε οριστικά κλάσεις;

Για να οριστικοποιηθεί η επιλογή μιας Κλάσης είναι απαραίτητο να πληρούνται ορισμένα (αρκετά "χαλαρά") κριτήρια. Τα κριτήρια αυτά πρέπει να εξετάζονται τόσο κατά την αρχική κατασκευή όσο και κατά την αναθεώρηση του διαγράμματος κλάσεων:

- Ανάγκη για απομνημόνευση πληροφοριών
 - Εάν το σύστημα πρέπει να θυμάται κάποιες τιμές για μια Κλάση τότε αυτή θα πρέπει να μπει στο μοντέλο (π.χ. όνομα, η λέξη-κλειδί για πρόσβαση σε ένα σύστημα, το επίπεδο εξουσιοδότησης για πρόσβαση, κλπ.)
 - Ο αναλυτής πρέπει να θυμάται ότι μερικές Κλάσεις-&Αντικείμενα παρουσιάζονται κατά τη συζήτηση με τον εμπειρογνώμονα, δείχνουν ενδιαφέρουσες, αλλά μπορεί να είναι άσχετες με το σύστημα υπό εξέταση.
- Ανάγκη για καταγραφή της συμπεριφοράς
 - Αν το σύστημα απαιτεί κάποιον υπολογισμό/λειτουργία (δηλ. μεθόδους) από κάποιο Αντικείμενο τότε αυτό θα πρέπει να μπει στο μοντέλο.
 - Οι πρωταρχικές μέθοδοι είναι αυτές που δημιουργούν (εγκαθιστούν), συνδέουν, παρέχουν πρόσβαση και διαγράφουν ένα Αντικείμενο και συνήθως δεν αναπαρίστανται στο διάγραμμα Κλάσεων.
- Θεωρητικά ένα Αντικείμενο θα μπορούσε να έχει Μεθόδους χωρίς να έχει Χαρακτηριστικά. Στην πράξη όμως αυτό συμβαίνει συνήθως όταν ο αναλυτής δεν σκέφτεται "αντικειμενοστρεφώς".
- (Συνήθως) Η Κλάση πρέπει να έχει περισσότερες από μία Ιδιότητες
- (Συνήθως) Περισσότερα από ένα Αντικείμενα μέσα σε μια Κλάση
- Ιδιότητες που ισχύουν πάντα
- Μέθοδοι που ισχύουν πάντα

- Απαιτήσεις σχετικές με το πεδίο προβλήματος
 - Ο αναλυτής μπορεί να θέτει σχεδιαστικούς περιορισμούς εκεί όπου επιβάλλεται. Για παράδειγμα, μπορεί να ζητήσει την χρήση ενός αλγόριθμου αν υπάρχει κάποιος κανονισμός που το επιβάλλει (π.χ. κανονισμοί στατικών υπολογισμών σε κτίρια, κρατικά λογιστικά σχέδια, κλπ.)
- Αν δεν απαιτείται κάποιος συγκεκριμένος αλγόριθμος τότε μπορεί να προδιαγραφεί η ακρίβεια των υπολογισμών και να αφηθεί στον σχεδιαστή η απόφαση για την υλοποίηση.
- Ακόμη και όταν ο πελάτης ζητήσει μια συγκεκριμένη αρχιτεκτονική ή σχεδιαστική λύση η οποία υποτίθεται ότι δεν θα αλλάξει ποτέ, καλό είναι αυτή να κρατηθεί έξω από το μοντέλο. Αν χρειασθεί να αλλάξει (!) το μοντέλο-ΑΑ θα υποστεί σχετικά μικρές τροποποιήσεις.
- Οι σχεδιαστές θα προσθέσουν Κλάσεις σχετικά με τις λειτουργίες που καθορίζονται μετά από σχεδιαστικές αποφάσεις (διαπροσωπεία ανθρώπου-μηχανής, διαχείριση δεδομένων, πρωτόκολλα, κλπ).
- Όχι απλά αποτελέσματα υπολογισμών

Θα πρέπει να αποφεύγεται η εισαγωγή στο μοντέλο αποτελεσμάτων απλών υπολογισμών. Για παράδειγμα, αν ένα σύστημα γνωρίζει ήδη την ημερομηνία γέννησης ενός πελάτη δεν χρειάζεται η ιδιότητα “Ηλικία”.

4. ΑΣΚΗΣΕΙΣ

4.1 ATM Πρώτη ανάλυση κλάσεων

Class ATM

- **Boundary / entity objects – μέρη του ATM**

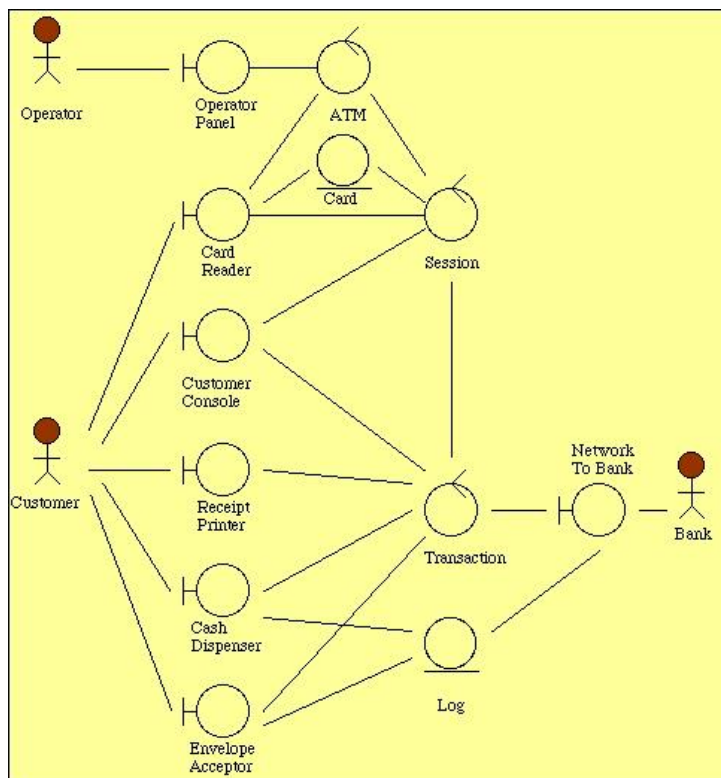
- Class CardReader
- Class CashDispenser
- Class CustomerConsole
- Class Log
- Class NetworktoBank
- Class OperatorPanel
- Class ReceiptPrinter

- **Boundary / entity objects – μέρη του ATM**

- Class Session
- Class Transaction
- Class Withdrawal
- Class Transfer
- Class Inquiry

- **Entity objects – για την ανάθεση υποχρεώσεων σε άλλα αντικείμενα**

- Class Balance
- Class Card
- Class Message
- Class Receipt
- Class Status



Εικόνα 1

Η κλάση ATM χειρίζεται το Startup και Shutdown για αυτό δεν χρειάζονται ξεχωριστά αντικείμενα



Εικόνα 2 Κλάση οντοτήτων, ελέγχου, συνοριακή

4.1.1 Κάρτες Κλάσης – Υποχρέωσης - Συνεργασίας (CRC)

Class ATM	
Υποχρεώσεις (Responsibilities)	Συνεργάτες (Collaborators)
<ul style="list-style-type: none"> • Ξεκινά όταν πατά τον διακόπτη – ON • Τερματίζει όταν πατά τον διακόπτη- OFF • Ξεκινά ένα session όταν εισέρχεται η κάρτα του πελάτη • Επιτρέπει πρόσβαση στις δοσοληψίες 	<ul style="list-style-type: none"> • <u>OperatorPanel</u> • <u>CashDispenser</u> • <u>NetworkToBank</u> • <u>CustomerConsole</u> • <u>Session</u>

Εικόνα 3

Class CardReader	
Υποχρεώσεις (Responsibilities)	Συνεργάτες (Collaborators)
<ul style="list-style-type: none"> • Πληροφορεί το ATM ότι εισήλθε η κάρτα • Διαβάζει πληροφορίες από την κάρτα • Επιστρέφει την κάρτα • Κατακρατεί την κάρτα 	<ul style="list-style-type: none"> • <u>ATM</u> • <u>Card</u>

Εικόνα 4

Class CashDispenser	
Υποχρεώσεις (Responsibilities)	Συνεργάτες (Collaborators)
<ul style="list-style-type: none">• Χειρίζεται τα χρήματα, με αρχικό ποσό εκκίνησης• Αναφέρει αν υπάρχουν διαθέσιμα χρήματα• Δίνει τα χρήματα	<ul style="list-style-type: none">• <u>Log</u>

Εικόνα 5

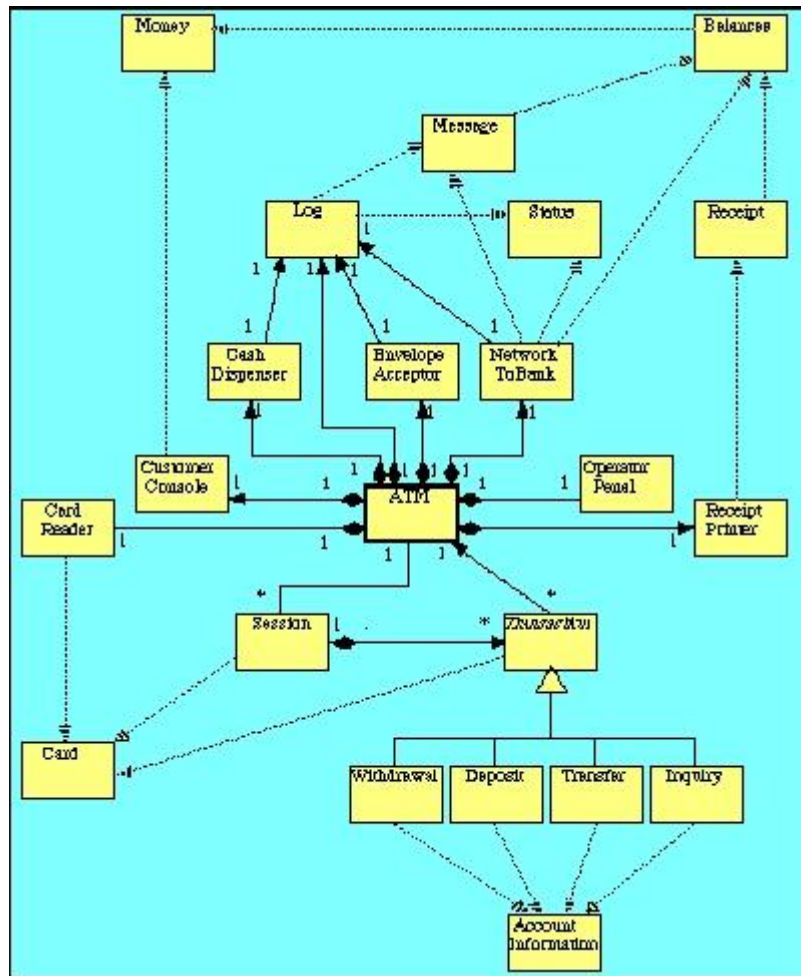
Υποχρεώσεις (Responsibilities)	Συνεργάτες (Collaborators)
Class CustomerConsole	
<ul style="list-style-type: none"> • Εμφάνιση μηνύματος • Δέχεται τον κωδικό – PIN • Εμφανίζει το μενού επιλογών και δέχεται επιλογή από το πληκτρολόγιο • Δέχεται ποσό από το πληκτρολόγιο • Ακυρώνει με το πάτημα του CANCEL 	
Class EnvelopeAcceptor	
<ul style="list-style-type: none"> • Δέχεται τον φάκελο του πελάτη και αναφέρει ακύρωση ή λήξη χρόνου 	<ul style="list-style-type: none"> • <u>Log</u>
Class Log	
<ul style="list-style-type: none"> • Καταγράφει μηνύματα προς την τράπεζα • Καταγράφει μηνύματα από την τράπεζα • Καταγράφει τις αναλήψεις • Καταγράφει τις καταθέσεις 	
Class NetworkToBank	
<ul style="list-style-type: none"> • Ξεκινά την σύνδεση με την τράπεζα • Στέλνει μηνύματα στην τράπεζα και περιμένει απάντηση • Τερματίζει τη σύνδεση με την τράπεζα 	<ul style="list-style-type: none"> • <u>Message</u> • <u>Log</u> • <u>Balances</u> • <u>Status</u>
Class OperatorPanel	
<ul style="list-style-type: none"> • Πληροφορεί το ATM για την κατάσταση του διακόπτη (ON-OFF) • Επιτρέπει τον χειριστή να ορίσει το αρχικό ποσό 	<ul style="list-style-type: none"> • <u>ATM</u>
Class ReceiptPrinter	
<ul style="list-style-type: none"> • Εκτύπωση απόδειξης 	<ul style="list-style-type: none"> • <u>Receipt</u>
Class Session	
<ul style="list-style-type: none"> • Εκτελεί την Περίπτωση Χρήσης-Session 	<ul style="list-style-type: none"> • <u>ATM</u> • <u>CardReader</u> • <u>Card</u> • <u>CustomerConsole</u> • <u>Transaction</u>

Εικόνα 6

Abstract Class Transaction	
<ul style="list-style-type: none"> • Επιτρέπει τον πελάτη να διαλέξει μια κίνηση • Εκτελεί την Περίπτωση Χρήσης – κίνηση • Εκτελεί τον έλεγχο του PIN 	<ul style="list-style-type: none"> • <u>ATM</u> • <u>CustomerConsole</u> • <u>Withdrawal</u> • <u>Transfer</u> • <u>Inquiry</u> • <u>Message</u> • <u>NetworkToBank</u> • <u>Receipt</u> • <u>ReceiptPrinter</u> • <u>CustomerConsole</u> • <u>Session</u> • <u>CardReader</u>
Class Withdrawal	
<ul style="list-style-type: none"> • Εκτελεί λειτουργίες της Περίπτωσης Χρήσης - ανάληψης 	<ul style="list-style-type: none"> • <u>CustomerConsole</u> • <u>CashDispenser</u> • <u>Message</u> • <u>Receipt</u>
Class Deposit	
<ul style="list-style-type: none"> • Εκτελεί λειτουργίες της Περίπτωσης Χρήσης – κατάθεσης 	<ul style="list-style-type: none"> • <u>CustomerConsole</u> • <u>Message</u> • <u>EnvelopeAcceptor</u> • <u>Receipt</u>
Class Transfer	
<ul style="list-style-type: none"> • Εκτελεί λειτουργίες της Περίπτωσης Χρήσης – μεταφορά 	<ul style="list-style-type: none"> • <u>CustomerConsole</u> • <u>Message</u> • <u>Receipt</u>
Class Inquiry	
<ul style="list-style-type: none"> • Εκτελεί λειτουργίες της Περίπτωσης Χρήσης – ενημέρωση 	<ul style="list-style-type: none"> • <u>CustomerConsole</u> • <u>Message</u> • <u>Receipt</u>
Class Balances	
<ul style="list-style-type: none"> • Αντιπροσωπεύει την κατάσταση του λογαριασμού 	
Class Card	
<ul style="list-style-type: none"> • Αντιπροσωπεύει τις κωδικοποιημένες πληροφορίες της κάρτας 	
Class Message	
<ul style="list-style-type: none"> • Αντιπροσωπεύει πληροφορίες που στέλνονται μέσω δικτύου στην τράπεζα 	
Class Receipt	
<ul style="list-style-type: none"> • Αντιπροσωπεύει πληροφορίες που θα εκτυπωθούν στη απόδειξη 	
Class Status	
<ul style="list-style-type: none"> • Αντιπροσωπεύει την πληροφορία κατάστασης - κίνησης της τράπεζας 	

Εικόνα 7

Διάγραμμα Κλάσεων - ATM



Εικόνα 8 Διάγραμμα Κλάσεων – ATM

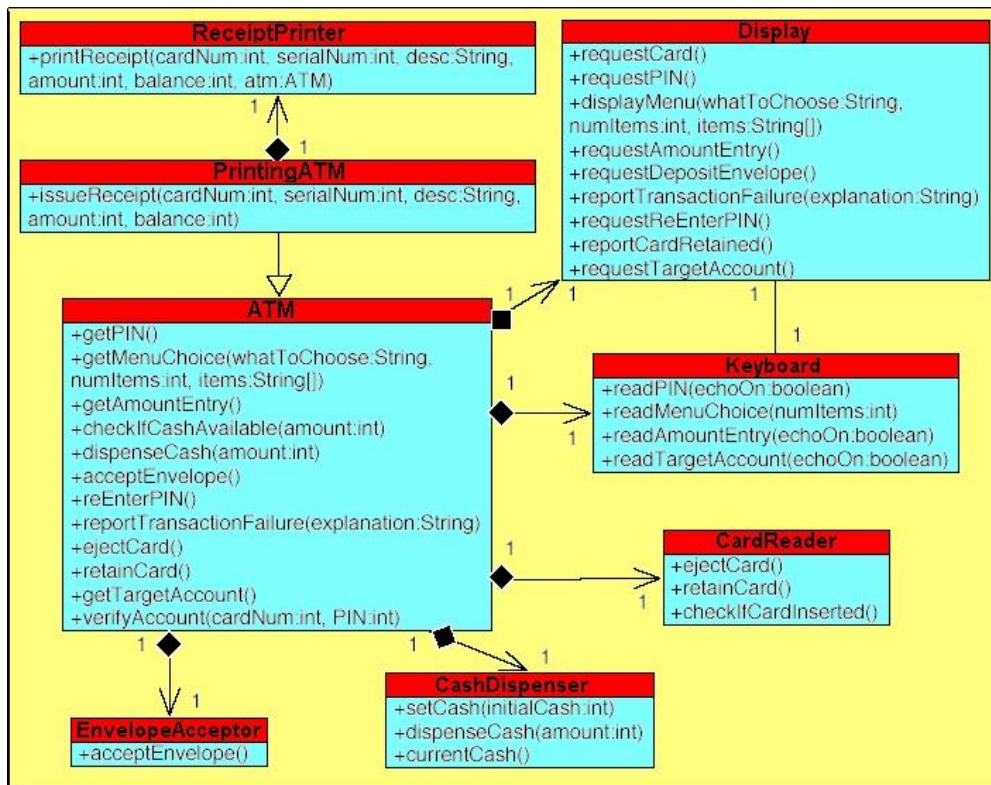
Μερικές κλάσεις προέκυψαν από λεπτομερέστερη ανάλυση, όπως:

- Money, αντιπροσωπεύει χρηματικά ποσά σε αριθμ. θέσεις
- AccountInformation, ονόμα-τα λογαριασμών που χρησιμοποι-ούν οι πελάτες

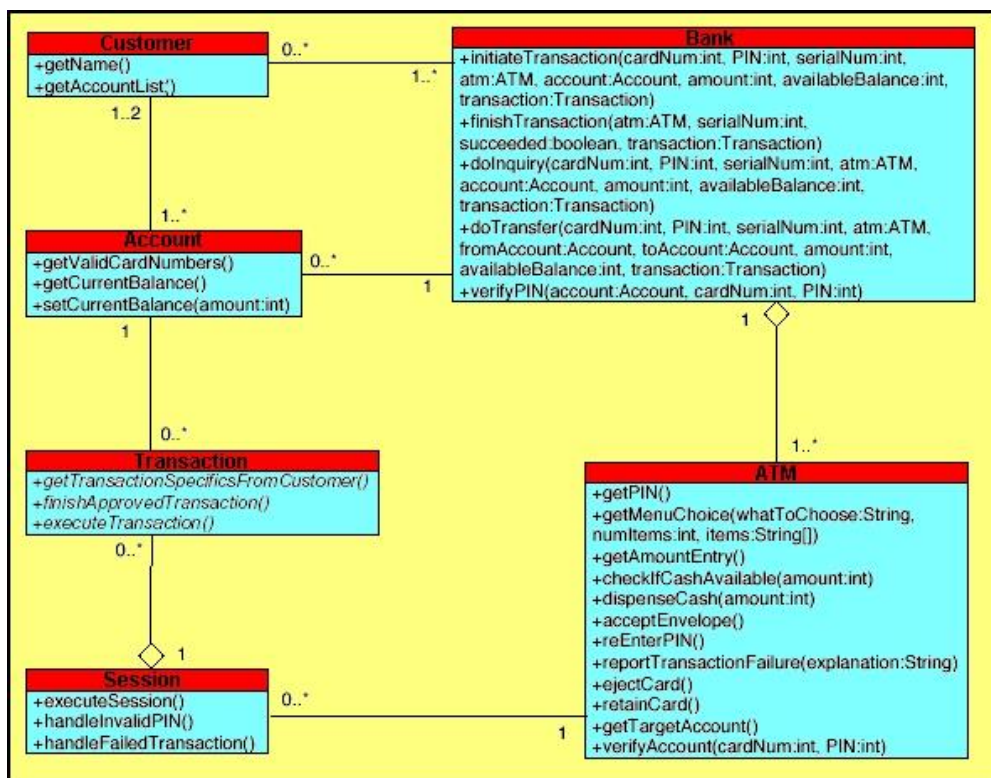
Μερικές κλάσεις προέκυψαν από τις CRC –κάρτες, όπως:

- Message
- Receipt
- Status
- Balances

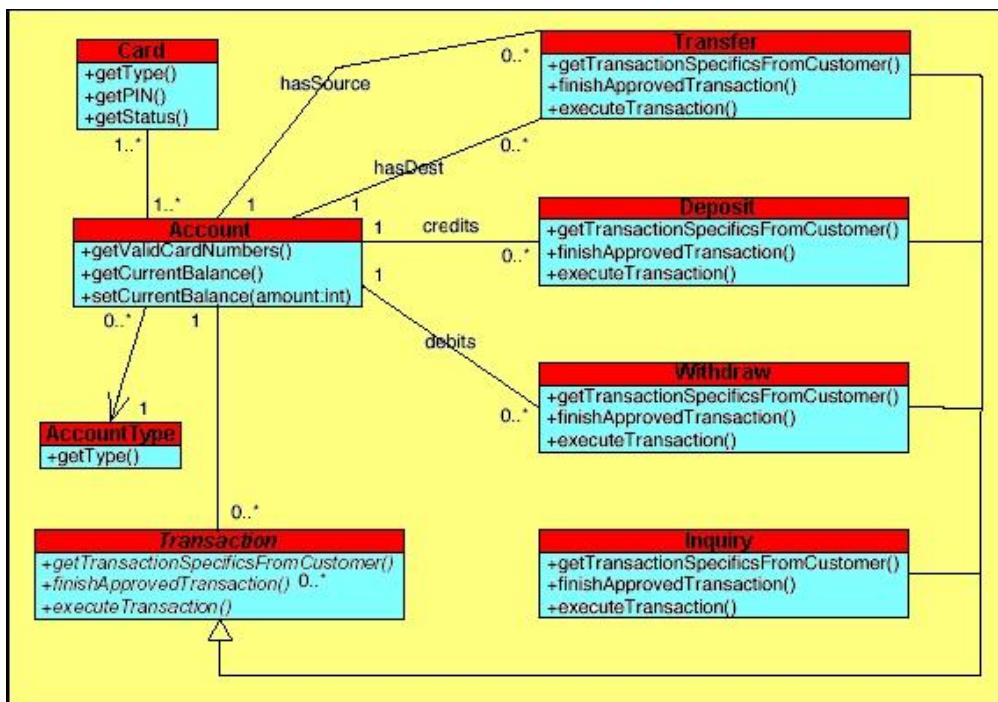
4.1.2 Διάγραμμα Κλάσεων Αναλυτικά



Εικόνα 9



Εικόνα 10



Εικόνα 11