

Σχεδίαση

Τι κάνουμε στη σχεδίαση;

- Στη σχεδίαση βρίσκουμε τη μορφή του συστήματος
- Η σχεδίαση πρέπει να λάβει υπ' όψη της τόσο τις λειτουργικές όσο και τις μη-λειτουργικές απαιτήσεις

Οι βασικοί στόχοι της σχεδίασης

- Οι βασικοί στόχοι της σχεδίασης είναι
 - Η απόκτηση βαθιάς κατανόησης των θεμάτων που αφορούν τις μη-λειτουργικές απαιτήσεις και τους περιορισμούς σχετικά με τη γλώσσα προγραμματισμού, λειτουργικό σύστημα, κτλ
 - Να ορίσει κατάλληλες απαιτήσεις όσον αφορά τα υποσυστήματα και τις διεπαφές τους, προετοιμάζοντας το έδαφος για την υλοποίηση
 - Να βοηθήσει στην υποδιαίρεση των υποσυστημάτων
 - Να αναπαραστήσει γραφικά το σύστημα
 - Να ορίσει μια γενίκευση του συστήματος χωρίς τις λεπτομέρειες της υλοποίησης

Ο ρόλος της σχεδίασης

- Στη σχεδίαση προχωράμε καθώς φεύγουμε από τη φάση επεξεργασίας και προχωρούμε στη φάση υλοποίησης
- Επειδή η σχεδίαση είναι κοντά στην υλοποίηση το μοντέλο σχεδίασης πρέπει να συντηρείται καθ' όλη τον κύκλο ζωής του προϊόντος

Το μοντέλο σχεδίασης

- Το μοντέλο σχεδίασης περιγράφει χρησιμοποιώντας αντικείμενα πως θα πραγματοποιηθούν οι περιπτώσεις χρήσης αν λάβουμε υπ' όψη τους περιορισμούς από τις λειτουργικές και μη-λειτουργικές απαιτήσεις και τους περιορισμούς από το περιβάλλον υλοποίησης
- Το μοντέλο σχεδίασης αντιπροσωπεύεται από ένα σύστημα σχεδίασης.
- Το σύστημα σχεδίασης μπορεί να έχει ένα ή περισσότερα υποσυστήματα
- Τόσο τα υποσυστήματα όσο και οι τάξεις σχεδίασης αποτελούν γενικεύσεις οι οποίες όμως βρίσκονται σε αντιστοιχία 1-1 με τα αντίστοιχα σκευάσματα της υλοποίησης
- Μέσα στο μοντέλο σχεδίασης οι περιπτώσεις χρήσης πραγματώνονται από τάξεις σχεδίασης και τα αντικείμενά τους

Σύγκριση του μοντέλου του Μοντέλου Ανάλυσης και του Μοντέλου Σχεδίασης

- | | |
|--|--|
| <ul style="list-style-type: none"> • Περισσότερο νοητικό μια που αποτελείται από γενικεύσεις και δεν αναφέρεται σε υλοποίηση • Μπορεί να εφαρμοστεί σε πολλά σχέδια • Μόνο 3 είδη τάξεων ανάλυσης • Λιγότερο ακριβό (1:5) από ότι το μοντέλο σχεδίασης • Λίγα επίπεδα • Δυναμικό χωρίς βάρος στην ακολουθία • Σκιαγραφεί τη σχεδίαση του συστήματος και της αρχιτεκτονικής • Δεν είναι ανάγκη να συντηρείται μέχρι τέλους • Ορίζει μια δομή που χρησιμοποιείται ως είσοδο στη σχεδίαση του συστήματος | <ul style="list-style-type: none"> • Φυσικό μοντέλο που αποτελεί το σχέδιο του συστήματος • Δεν είναι γενικό, αφορά μια υλοποίηση • Οποιοσδήποτε αριθμός • Πιο ακριβό (5:1) • Πολλά επίπεδα • Δυναμικό με βάρος στην ακολουθία • Καθορίζει τη σχεδίαση του συστήματος • Καλό είναι να συντηρηθεί μέχρι τέλους • Διαμορφώνει το σύστημα προσπαθώντας να τηρήσει τις επιταγές της ανάλυσης όσο το δυνατόν περισσότερο |
|--|--|

Σχεδιάζοντας μια περίπτωση χρήσης

- Ο σκοπός της σχεδίασης είναι
 - να βρει τις τάξεις σχεδίασης και τα υποσυστήματα
 - να κατανείμει τη συμπεριφορά μιας περίπτωσης χρήσης σε αντικείμενα σχεδίασης που επικοινωνούν μεταξύ τους και στα αντίστοιχα υποσυστήματα
 - να ορίσει τις απαιτήσεις για τις λειτουργίες της περίπτωσης χρήσης και των υποσυστημάτων
 - να αποτυπώσει τις απαιτήσεις υλοποίησης της περίπτωσης χρήσης

Βρίσκοντας τις τάξεις σχεδίασης

- Μελετήστε τις τάξεις ανάλυσης που παίρνουν μέρος στις αναλύσεις των περιπτώσεων χρήσης
- Μελετήστε τις ειδικές απαιτήσεις της αντίστοιχης ανάλυσης της περίπτωσης χρήσης. Ορίστε αντίστοιχες τάξεις σχεδίασης που υλοποιούν αυτές τις απαιτήσεις
- Αποτέλεσμα των παραπάνω είναι να βρεθεί μια τάξη. Ο μηχανικός συστατικών από εδώ και πέρα πρέπει να ορίσει τις λεπτομέρειες της τάξης

Περιγράψτε τις αλληλεπιδράσεις μεταξύ των αντικειμένων

- Αφού βρείτε τα αντικείμενα που παίρνουν μέρος σε μια περίπτωση χρήσης ορίστε και τον τρόπο που αυτά αλληλεπιδρούν μεταξύ τους
- Η περιγραφή μπορεί να γίνει είτε χρησιμοποιώντας ένα διάγραμμα συνεργασίας ή ένα διάγραμμα ακολουθίας
- Χρησιμοποιήστε ένα διάγραμμα για κάθε πιθανή εναλλακτική ροή

Σχεδιάζοντας τις τάξεις

- Κατά τη σχεδίαση των τάξεων ορίζετε
 - τις λειτουργίες της τάξης
 - τα χαρακτηριστικά της
 - τις σχέσεις στις οποίες μετέχει
 - τις μεθόδους που υλοποιούν τις λειτουργίες
 - τις πιθανές καταστάσεις της
 - πιθανές εξαρτήσεις της από τη γενική σχεδίαση
 - Απαιτήσεις από την υλοποίηση
 - Πιθανόν διεπαφές που πρέπει να προσφέρει

Πώς θα μπορούσατε να προχωρήσετε

- Δείτε τις κλάσεις ορίου
- Αν αφορούν επικοινωνία με ανθρώπινο χαρακτήρα, αποφασίστε ποιά ακριβώς πληροφορία θα παρουσιάσουν και πώς
 - Ορίστε για παράδειγμα Buttons, TextAreas κτλ

Πώς θα μπορούσατε να προχωρήσετε

- Δείτε τις κλάσεις ελέγχου
 - Αποφασίστε πως θα αντιδρά κάθε τάξη ελέγχου σε περιπτώσεις λαθών; Τι θα κάνει;

Πώς θα μπορούσατε να προχωρήσετε;

- Δείτε τις τάξεις οντοτήτων;
 - Πώς θα αποθηκεύονται και που;
 - Ανάλογα με την τεχνολογία που θα χρησιμοποιήσετε μπορεί να πρέπει να υλοποιήσετε και ένα μηχανισμό διασύνδεσης
 - Για τις σχεσιακές βάσεις υπάρχουν αυτόματα εργαλεία που μπορούν να κάνουν σε κάποιο βαθμό τη μετατροπή ανάμεσα σε τάξεις και πίνακες. Στο επόμενο μάθημα θα δούμε τις γενικές αρχές που διέπουν αυτή την αλλαγή

Άλλα πράγματα που θα μπορούσατε να κάνετε

- Αν το σύστημά σας εκτελείται σε παραπάνω από ένα υπολογιστές ορίστε τουλάχιστον μια κλάση η οποία έχει και το γενικό έλεγχο και συντονισμό όλων των αντικειμένων που δημιουργούνται στον υπολογιστή. Τέτοιες κλάσεις ονομάζονται «ενεργές» κλάσεις

Άλλα πράγματα που θα μπορούσατε να κάνετε

- Ορισμένα αντικείμενα αλλάζουν πολλές καταστάσεις στη διάρκεια ζωής τους.
- Για παράδειγμα μια παραγγελία μπορεί να είναι «Ανοικτή», «Υπό εκτέλεση», «Υπό παράδοση», «Ολοκληρωμένη», «Πληρωμένη» κτλ.
- Για τέτοια αντικείμενα μπορεί να είναι χρήσιμο να κάνετε ένα διάγραμμα που δείχνει πότε και πως αλλάζουν καταστάσεις

Παράδειγμα Σχεδίασης

Σύστημα καταχώρησης ονομάτων χώρου

- Στο προηγούμενα μαθήματα είδαμε κάποιες περιγραφές για ένα σύστημα ονομάτων χώρου

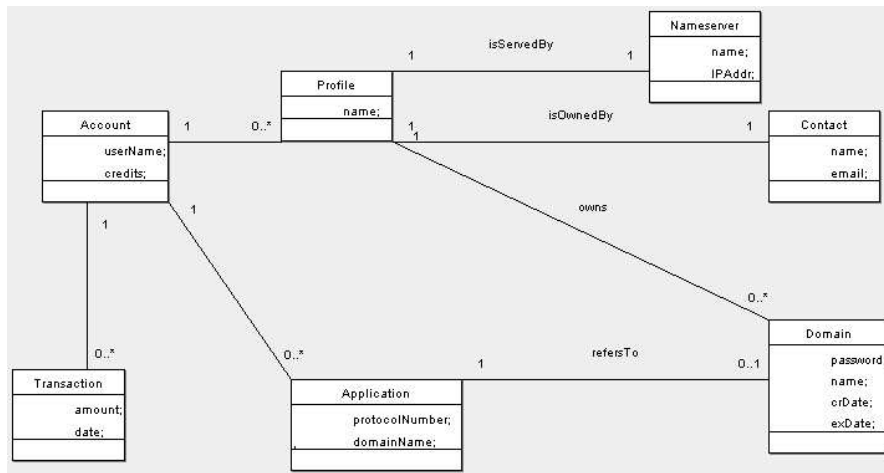
Απαιτήσεις

Περιγραφή συστήματος κατοχύρωσης ονομάτων χώρου

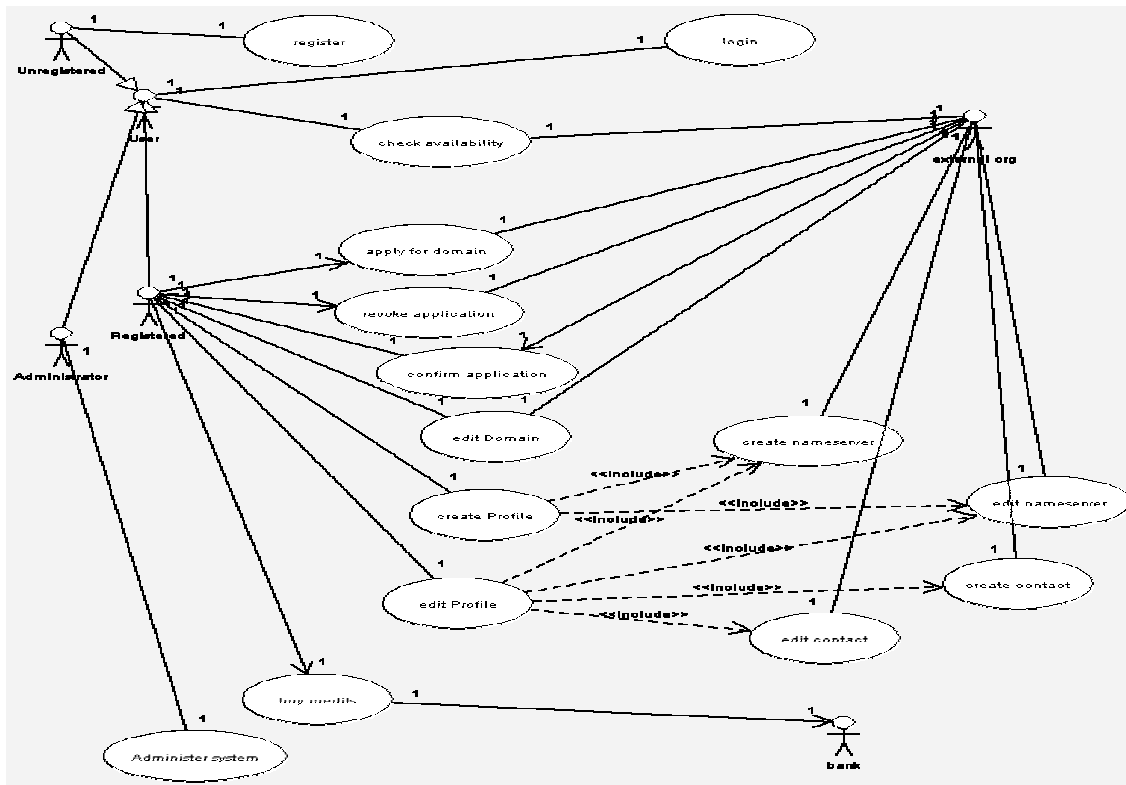
Ένα σύστημα κατοχύρωσης ονομάτων χώρου επιτρέπει στους χρήστες του να κατοχυρώνουν ονόματα χώρου και να δημιουργούν εξυπηρετητές ονοματοδρασίας και υπεύθυνους για το όνομα.

1. Το σύστημα πρέπει να επιτρέπει σε χρήστες να εγγραφούν δωρεάν.
2. Οι μη εγγεγραμμένοι χρήστες έχουν τη δυνατότητα να εγγραφούν και να ελέγξουν τη διαθεσιμότητα ενός ονόματος χώρου.
3. Οι εγγεγραμμένοι χρήστες αφού εισέλθουν έχουν επιπλέον τη δυνατότητα να δημιουργήσουν προφίλ. Σε κάθε προφίλ δίνουν ένα όνομα και τους 4 εξυπηρετητές και υπεύθυνους που ανήκουν σε αυτό το προφίλ. Ένας από τους υπεύθυνους θεωρείται ο ιδιοκτήτης του προφίλ. Κάθε εξυπηρετητής έχει ένα όνομα και μια διεύθυνση IP. Κάθε υπεύθυνος έχει ένα όνομα και ένα email.
4. Οι εγγεγραμμένοι χρήστες αφού εισέλθουν έχουν τη δυνατότητα να επεξεργαστούν τα προφίλ που έχουν δημιουργήσει.
5. Οι εγγεγραμμένοι χρήστες αφού εισέλθουν έχουν τη δυνατότητα να κάνουν μια αίτηση για την κατοχύρωση ενός ονόματος, δίνοντας το όνομα χώρου που επιθυμούν, το προφίλ στο οποίο θα ανήκει και έναν κωδικό. Το σύστημα επικοινωνεί με έναν εξωτερικό οργανισμό για την έγκριση ο οποίος αποδίδει έναν αριθμό πρωτοκόλλου στην αίτηση.
6. Οι εγγεγραμμένοι χρήστες αφού εισέλθουν έχουν τη δυνατότητα για την ανάκληση μιας αίτησης (αν αυτή δεν έχει εγκριθεί ακόμα) δίνοντας τον αριθμό πρωτοκόλλου.
7. Ο εξωτερικός οργανισμός μπορεί να εγκρίνει ή να απορρίψει την αίτηση. Σε κάθε περίπτωση ειδοποιεί το σύστημα δίνοντας τον αριθμό πρωτοκόλλου της αίτησης και τα στοιχεία του ονόματος χώρου που κατοχυρώθηκαν.
8. Ο εγγεγραμμένος χρήστης αφού εισέλθει μπορεί αλλάξει τον κωδικό ενός ονόματος χώρου.
9. Οι εγγεγραμμένοι χρήστες αφού εισέλθουν μπορούν να προσαγορεύσουν μονάδες δίνοντας την πιστωτική τους κάρτα και επιλέγοντας πόσες μονάδες επιθυμούν. Το σύστημα επικοινωνεί με μια τράπεζα η οποία και αναφέρει αν η συναλλαγή ήταν επιτυχημένη.
10. Κάθε επιτυχημένη κατοχύρωση ονόματος χρεώνει το χρήστη με συγκεκριμένο αριθμό μονάδων. Αν ο χρήστης ανακαλέσει την αίτησή του πριν την έγκρισή της τότε δεν υπάρχει χρέωση.
11. Το σύστημα επικοινωνεί με έναν εξωτερικό οργανισμό ο οποίος και είναι υπεύθυνος για την έγκριση ή απόρριψη αιτήσεων, δημιουργίας εξυπηρετητών, υπευθύνων καθώς και οποιονδήποτε αλλαγών σε αυτά. Οι αλλαγές στα στοιχεία ονομάτων χώρου και η δημιουργία εξυπηρετητών και υπευθύνων καθώς και η επεξεργασία των στοιχείων των εξυπηρετητών ή των υπευθύνων εγκρίνεται ή απορρίπτεται άμεσα.
12. Τέλος το σύστημα επιτρέπει σε ένα ειδικό χρήστη, διαχειριστή να δει όλες τις συναλλαγές που έχουν γίνει με την τράπεζα.
13. Το σύστημα πρέπει να είναι πολυγλωσσικό και να δουλεύει μέσω WEB.

Domain model



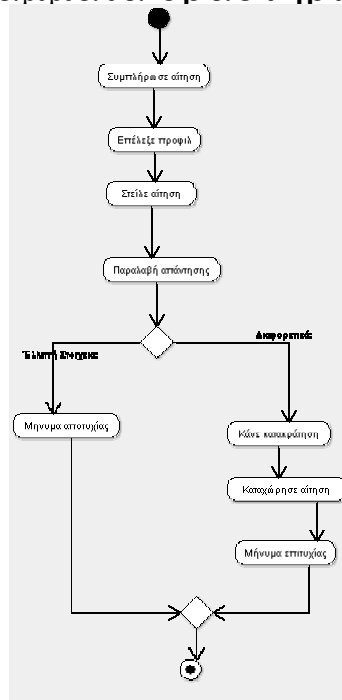
Μοντέλο περιπτώσεων χρήσης



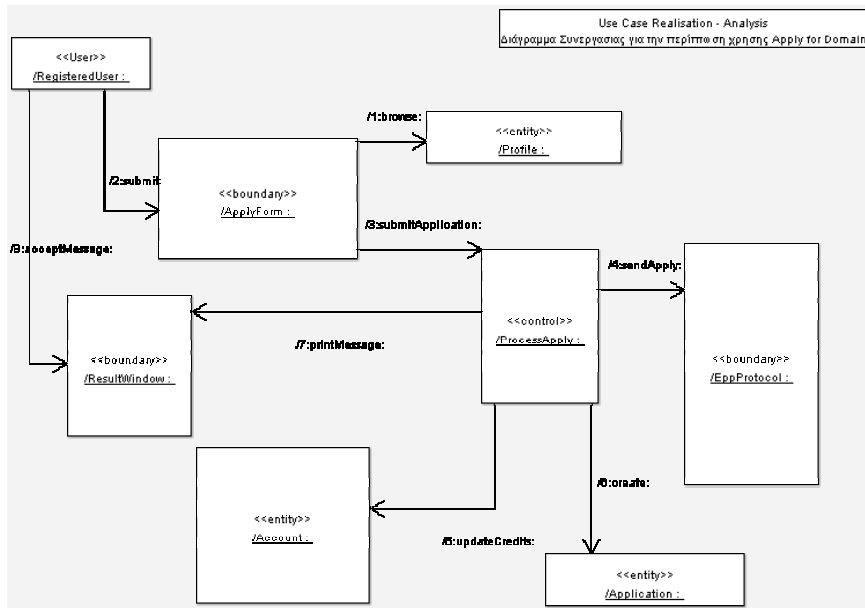
Περιγραφές περιπτώσεων χρήσης

Όνομα Περίπτωσης Χρήσης	Apply for domain
Αριθμός Περίπτωσης Χρήσης	4
Έκδοση	1
Συγγραφείς (με χρονολογική σειρά)	Νίκος Πεταλίδης
Αναφέρεται στις απαιτήσεις υπ αριθμόν	5
Προτεραιότητα (Σημαντική, Μέση, Χαμηλή)	Σημαντική
Χαρακτήρες που αφορά	Εγγεγραμμένος Χρήστης, Εξωτερικός Οργανισμός
Προαπαιτούμενα	Να έχει μονάδες ο χρήστης Να έχει φτιάξει ένα <u>προφίλ</u> Να έχει εισέλθει στο σύστημα
Περιγραφή	1. Αφού εισέλθει ο χρήστης στο σύστημα επιλέγει 'Αίτηση Κατοχύρωσης' 2. Ο χρήστης συμπληρώνει το όνομα χώρου και τον κωδικό που επιθυμεί 3. Ο χρήστης επιλέγει το <u>προφίλ</u> στο οποίο θα ανήκει το όνομα 4. ο χρήστης επιλέγει 'Αποστολή Αίτησης' 5. Το σύστημα επικοινωνεί με τον εξωτερικό οργανισμό ο οποίος αναφέρει ότι η αίτηση ήταν πλήρης και της δίνει έναν αριθμό πρωτοκόλλου. 6. Το σύστημα αποθηκεύει και αναφέρει τον αριθμό πρωτοκόλλου στο χρήστη και παρακρατεί τις μονάδες που αναλογούν στη κατοχύρωση domain
Εναλλακτική ροή A	5. Το σύστημα επικοινωνεί με τον εξωτερικό οργανισμό ο οποίος αναφέρει ότι η αίτηση δεν ήταν δυνατή να επεξεργασθεί λόγω ελλείπων στοιχείων. 6. Το σύστημα αναφέρει αποτυχία αποστολής της αίτησης λόγω ελλείπων στοιχείων

Διαγράμματα δραστηριότητας



Διάγραμμα συνεργασίας για την περίπτωση χρήσης «Καταχώρηση ονόματος»



Για την τάξη ορίου “Apply Form” είχαμε επίσης ορίσει τα ακόλουθα

ApplyForm
-domainName : .
-profileName : .
+submit() : .

Class	ApplyForm	
Superclass(es)		
Subclasses		
	Ευθύνη	Συνεργάτες
	Ο χρήστης μπορεί να δώσει OK, να επιλέξει προφίλ και όνομα χώρου Εμφάνισε όλα τα προφίλ για να επιλέξει ο χρήστης Μόλις ο χρήστης πατήσει OK στέλνονται το όνομα χώρου και το επιλεγμένο προφίλ	User Profile ProcessApply

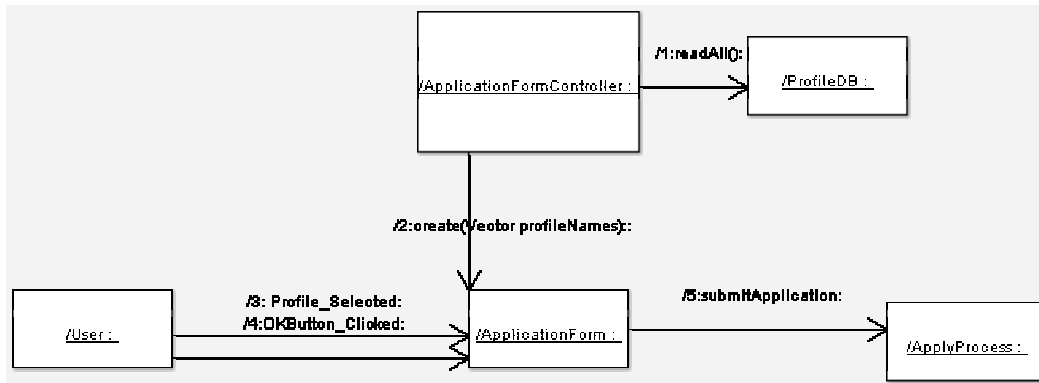
Στη σχεδίαση την τάξη αυτή μπορούμε να την αναλύσουμε περισσότερο

ApplyForm
-domainName : JTextField -profileName : JComboBox -buttonOK : JButton
+OKButton_Clicked() : . +Profil_Selected()

Τι παρατηρούμε;

- Παρατηρούμε ότι κάπως πρέπει να «γεμίζουμε» το JComboBox με τα ονόματα των προφίλς

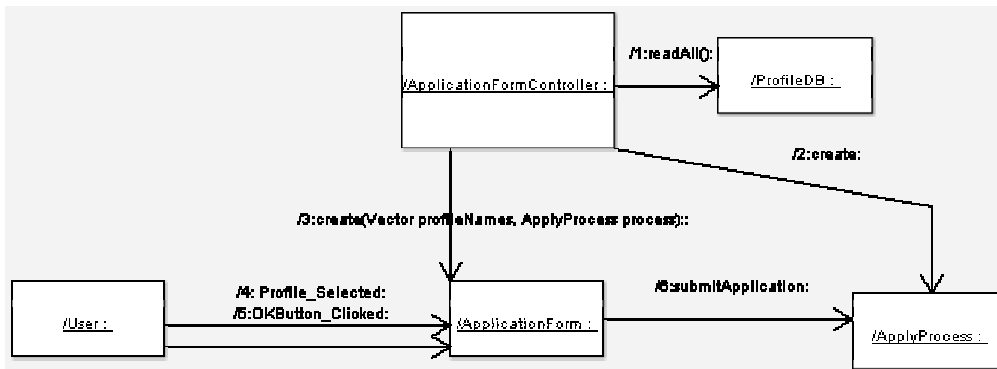
Πρώτη προσπάθεια



Τι άλλο παρατηρούμε

- Η τάξη `ApplyForm` έχει μια μέθοδο `OKButton_Clicked()`.
- Αυτή πρέπει να καλεί την ανάλογη μέθοδο `submit_Application()` της τάξης `ProcessApply`.
- Αλλά δεν ξέρει ποιό αντικείμενο αυτής της τάξης να καλέσει. Οπότε ξανασχεδιάζουμε και αυτή τη φορά έχουμε

Δεύτερη προσπάθεια



Σκιαγραφώντας τις τάξεις ελέγχου

- Η σχεδίαση των τάξεων ελέγχου μπορεί να είναι δύσκολη και λεπτεπίλεπτη δουλειά. Εξαρτάται από
 - την κατανομή του συστήματος
 - τις απαιτήσεις σε απόδοση
 - τις απαιτήσεις σε ασφάλεια

Σχεδιάζοντας την Apply Process

- Τι πρέπει να κάνει αυτή η τάξη; Από τη φάση ανάλυσης βλέπουμε ότι έχει τις ακόλουθες υποχρεώσεις

Class	ProcessApply	
Superclass(es)		
Subclasses		
	Responsibility	Collaborators
	Εξετάζει την ορθότητα της αίτησης που έστειλε ο χρήστης Αποστέλει την αίτηση και ελέγχει την απάντηση Αν η απάντηση ότι η αίτηση παραλήφθηκε χωρίς πρόβλημα δημιουργεί την αίτηση Ενημερώνει τις μονάδες του χρήστη Ενημερώνει το χρήστη	ApplyForm EppProtocol Application Account ResultWindow

Προβλήματα που μπορεί να συναντήσετε

- Ως τάξη ελέγχου η ProcessApply είναι υπεύθυνη για την αντιμετώπιση πιθανών προβλημάτων. Στη συγκεκριμένη περίπτωση τι προβλήματα μπορεί να συναντήσουμε;
 - Να μην έχει συμπληρώσει σωστά ο χρήστης την αίτηση (δηλαδή να μην έχει συμπληρώσει τα στοιχεία του domain name, ή του προφίλ)
 - Να μη μπορεί να στείλει την αίτηση
 - Να τη στείλει αλλά να πάρει απάντηση ότι υπήρχε πρόβλημα
 - Να προσπαθήσει να αποθηκεύσει τα στοιχεία της αίτησης και να μη μπορέσει
 - Να προσπαθήσει να αλλάξει τα στοιχεία των μονάδων του χρήστη και να μη μπορέσει

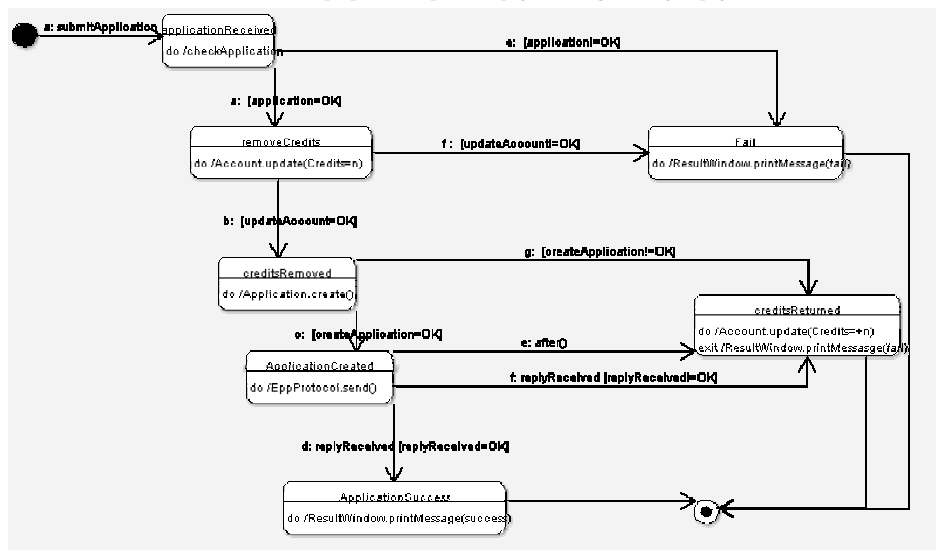
Η σχεδίαση των τάξεων ελέγχου

- Σχεδιάζοντας τις τάξεις ελέγχου σκεφθείτε τι προβλήματα μπορεί να αντιμετωπίσετε

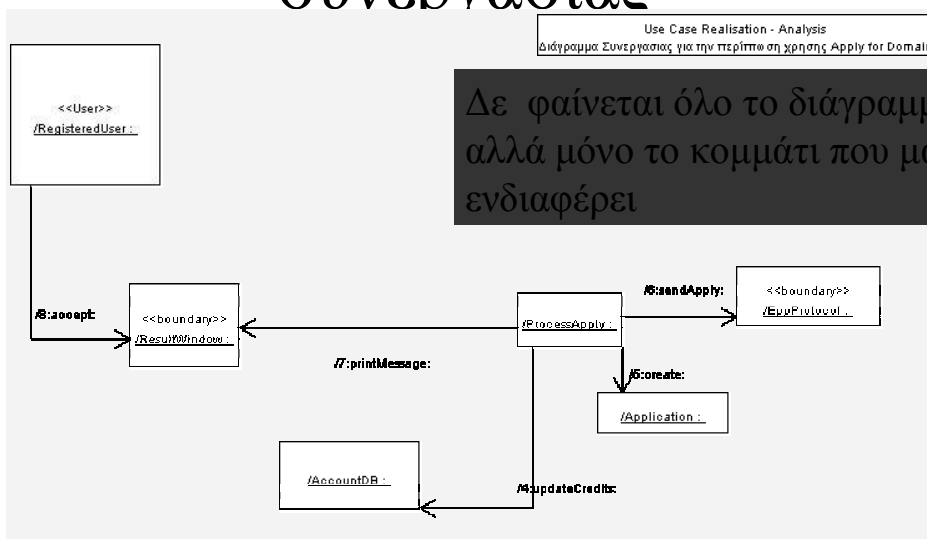
Στη συγκεκριμένη περίπτωση

- Στη συγκεκριμένη περίπτωση αποφασίζουμε ότι σε όλα τα προβλήματα απλά εμφανίζουμε ένα μήνυμα λάθους στο χρήστη
- Ειδικά όμως για την περίπτωση που δε μπορούμε να μειώσουμε τις μονάδες του χρήστη αποφασίζουμε ότι θα αλλάξουμε και τη ροή των γεγονότων. Έτσι πλέον πριν γίνει η αποστολή της αίτησης αφαιρούμε τις μονάδες και μόνο τότε αποστέλλουμε την αίτηση

Μπορούμε επομένως να περιγράψουμε τη ροή της ως εξής



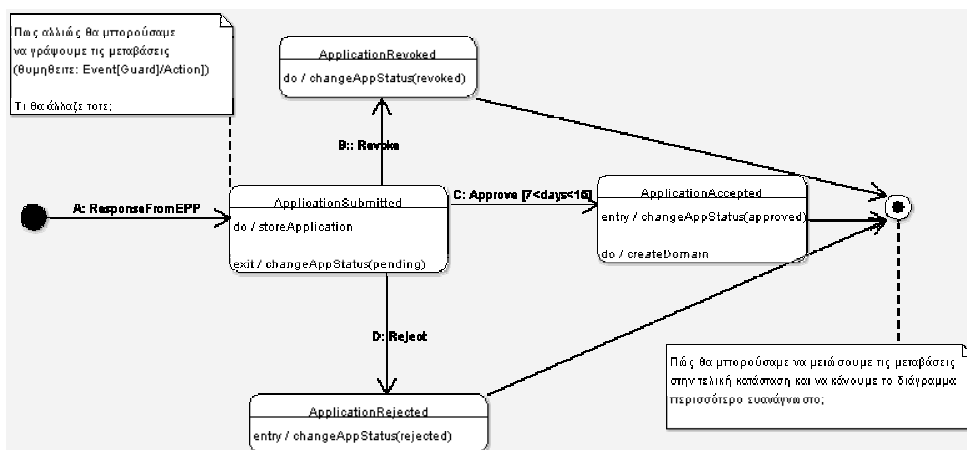
Βελτιώνουμε και το διάγραμμα συνεργασίας



Άλλες ενέργειες που μπορούμε να κάνουμε

- Στη διάρκεια της σχεδίασης πρέπει να κάνουμε ξεκάθαρη όποια διαδικασία θεωρούμε ότι μπορεί να δυσκολέψει την κατανόηση του συστήματος.
- Για παράδειγμα η διαχείριση ονομάτων χώρου κάνει «αιτήσεις» για ονόματα που μπορεί να εγκριθούν ή να απορριφθούν. Το σύστημά μας μάλιστα έχει και μια κλάση με το όνομα «Application» που αντιστοιχεί στις αιτήσεις.
- Μπορούμε να κάνουμε ένα διάγραμμα για να αναπαραστήσουμε πως δημιουργείται και πώς αλλάζει ένα αντικείμενο αυτής της κλάσης.

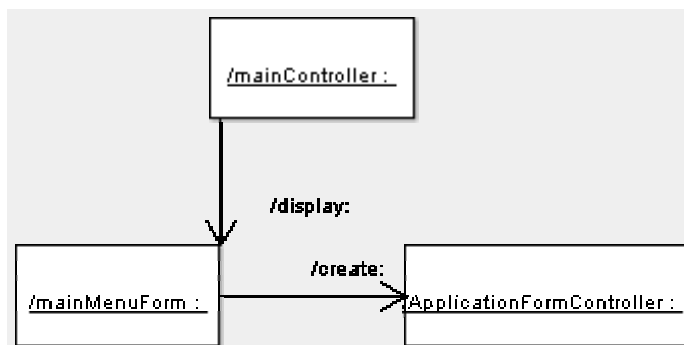
Η ζωή ενός αντικειμένου μιας κλάσης “Application”



Τι άλλο πρέπει να κάνουμε στη σχεδίαση

- Αφού περιγράψουμε το σύστημά μας με τον πιο πάνω τρόπο πρέπει να σκεφτούμε και μία ή περισσότερες τάξεις οι οποίες θα είναι υπεύθυνες για την εκτέλεση του προγράμματος.
- Στην περίπτωση του συστήματος κατοχύρωσης χρειαζόμαστε οπωσδήποτε μια διαδικασία η οποία εκτελείται συνεχώς και εμφανίζει την αρχική οθόνη και στη συνέχεια δημιουργεί τα αντίστοιχα αντικείμενα ανάλογα με τις επιθυμίες του χρήστη.

Παράδειγμα μιας ενεργής τάξης (mainController)



Παράδειγμα

Όνομα Κλάσης	mainController	
Τύπος	Active class	
Υπερκλάσεις		
Υποκλάσεις		
	Ευθύνη	Συνεργασίες με
	Η τάξη αυτή είναι η «καρδιά» του συστήματος	mainMenuForm
	Είναι η τάξη που «ξεκινά» το πρόγραμμα, και εμφανίζει την αρχική οθόνη στο χρήστη	
	Χαρακτηριστικά	

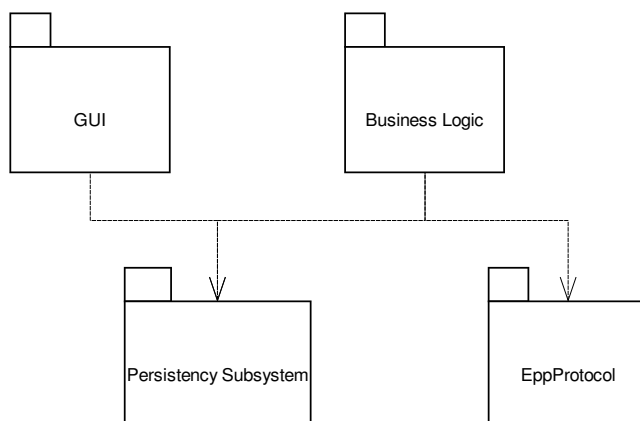
Υποσυστήματα

- Αν και δε θα επιμείνω στη διαίρεση του συστήματός σας σε υποσυστήματα, εντούτοις θα αναφέρω κάποιους τρόπους που θα μπορούσατε να χωρίσετε το σύστημα της κατοχύρωσης ονομάτων

Υποσυστήματα

- Στο σύστημα κατοχύρωσης μπορούμε ορίσουμε τουλάχιστον τρία υποσυστήματα
 - Ένα για τη διαχείριση της διεπαφής χρήστη (GUI)
 - Ένα για την επικοινωνία με τον εξωτερικό οργανισμό (EppProtocol)
 - Ένα για τη διαχείριση και συντονισμό όλων των ενεργειών (δηλαδή περιέχει τις περισσότερες τάξεις ελέγχου) (Business Logic)

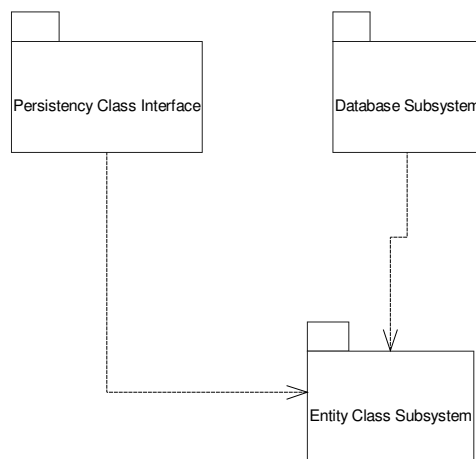
To Domain Registration Subsystem



Persistence Subsystem

- Το υποσύστημα Persistence Subsystem περιέχει όλες τις κλάσεις που αφορούν “entity” classes.
- Συνήθως μπορούμε να το χωρίσουμε και αυτό. Ένα υποσύστημα αφορά τη βάση δεδομένων ένα τις κλάσεις που υλοποιούν τη διασύνδεση και ένα τις ίδιες τις τάξεις

Παράδειγμα



Μια ανασκόπηση

- Στη σχεδίαση ξαναδείτε τις αναλύσεις των περιπτώσεων χρήσης που έχετε κάνει
- Με βάση αυτές και με βάση τις κλάσεις που έχετε ορίσει στην ανάλυση παράγετε πιο λεπτομερή διαγράμματα συνεργασίας/ακολουθίας και ορίστε πιο αναλυτικά τις κλάσεις
- Αν ο ρόλος μιας κλάσης δεν είναι ξεκάθαρος κάντε ένα statechart που να δείχνει κάτω από ποιες προϋποθέσεις δημιουργούνται αντικείμενα αυτής της τάξης και αλλάζουν κατάσταση
- Αν η λειτουργία μιας κλάσης δεν είναι ξεκάθαρη χρησιμοποιήστε ένα statechart ή ένα διάγραμμα δραστηριότητας για να την περιγράψετε καλύτερα

Βιβλιογραφία

- Οι σημερινές σημειώσεις προέρχονται από
- Jacobson, Rumbaugh, Booch, *The Unified Software Development Process*, Addison Wesley