

Robustness-diagrams-v1.0

LibShare

Ονόματα / ΑΜ / Έτος:

Γρηγόρης Καπαδούκας / 1072484 / 4°

Χρήστος Μπεστητζάνος / 1072615 / 4°

Νικόλαος Αυγέρης / 1067508 / 5°

Περικλής Κοροντζής / 1072563 / 4°

1 Robustness Diagrams

Σημειώσεις και Παραδοχές Έκδοσης v1.0:

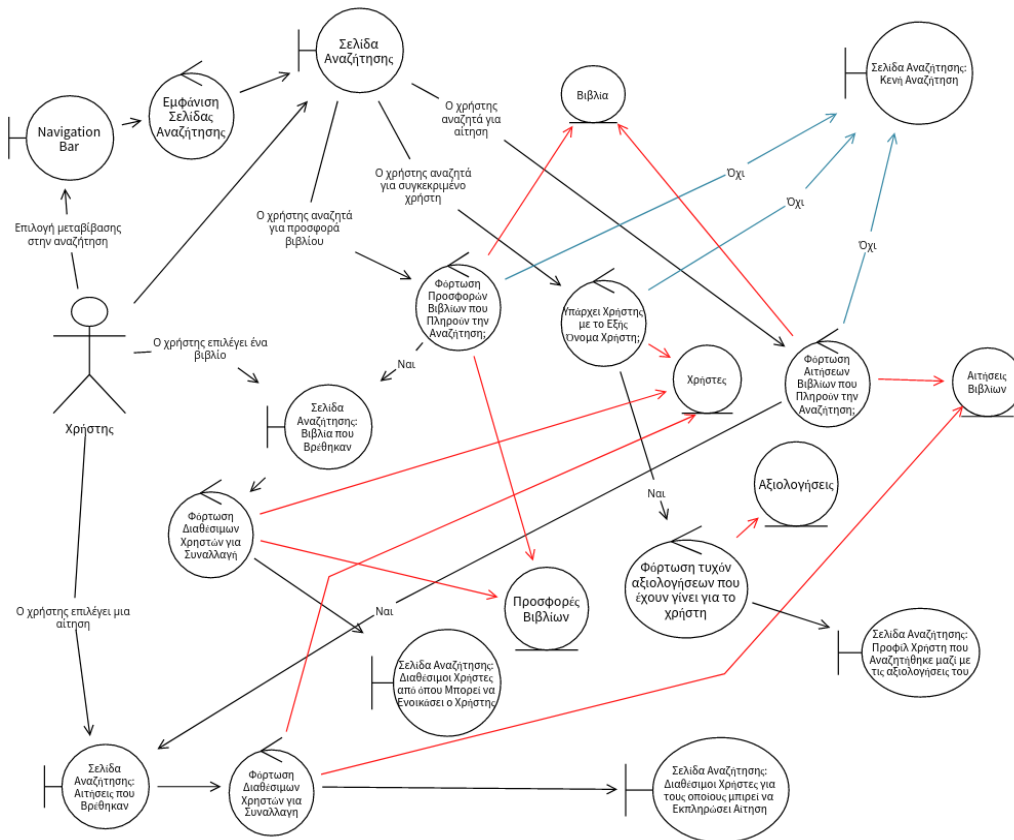
- Σε όλα τα παρακάτω robustness diagrams έχουμε χρησιμοποιήσει μπλε χρώμα για να δείξουμε τις εναλλακτικές ροές και κόκκινο για να δείξουμε τα βελάκια προς αντικείμενα οντοτήτων, ώστε να ξεχωρίζουν καλύτερα σε σχέση με τα άλλα βέλη.
- Επίσης επειδή σε πολλές περιπτώσεις θέλουμε να δηλώσουμε την επιστροφή σε προηγούμενη οθόνη, χωρίς όμως να υπάρχει αβεβαιότητα ως προς την ακολουθία ενεργειών και να ακολουθήσουμε πιστά τις ενέργειες όπως τις ορίσαμε στα use cases, αναγκαστήκαμε να αναφέρουμε τα τμήματα της οθόνης που χρησιμοποιούμε κάθε φορά στα συνοριακά αντικείμενα, με αποτέλεσμα πολλές φορές πολλά συνοριακά να αναφέρονται στην ίδια οθόνη. Αυτό το αναγράφουμε με την σύνταξη "<Όνομα Οθόνης>: <Τμήμα>" στα ονόματα των συνοριακών.

Παρόλα αυτά όμως, για λόγους απλότητας της αρχιτεκτονικής της εργασίας, προχωρώντας στα sequence diagrams και έπειτα στον κώδικα της άσκησης, για όλα αυτά τα συνοριακά αντικείμενα που αποσκοπούν στην ίδια οθόνη θα μεταφερθεί μόνο ένα συνοριακό στα sequence, οπότε θα προκύψει μια κλάση για όλα τα συνοριακά αυτά και στο κώδικα, αντί για τη συνηθισμένη τεχνική της μεταφοράς όλων των συνοριακών και τη δημιουργία μιας κλάσης για κάθε συνοριακό. Σε κάθε περίπτωση που κάνουμε αυτή τη παραδοχή θα το αναφέρουμε ρητά κάτω από το robustness diagram όπου το εφαρμόζουμε και θα εξηγούμε γιατί πήραμε αυτή τη απόφαση και πως μας βοήθησε έναντι της συνηθισμένης τεχνικής μεταφοράς όλων.

- Επίσης αναφέρουμε ότι σε ορισμένα robustness diagrams, έχουμε πάλι πολλαπλά αντικείμενα οντοτήτων πολλές φορές που θέλουμε να αποσκοπούν σε ένα lifeline του sequence διαγράμματος και σε μια κλάση του κώδικα. Αυτό συμβαίνει κυρίως όταν το use cases θέλουμε στο τελικό κώδικα να χρησιμοποιεί πολλαπλά στιγμιότυπα της ίδιας κλάσης (πχ "Προφίλ ενοικιαστή" και "Προφίλ ιδιοκτήτη" που αποσκοπούν στην ίδια κλάση "User", εφόσον όλοι οι χρήστες μπορούν να είναι και ενοικιαστές και ιδιοκτήτες σε διαφορετικές συναλλαγές). Εδώ πάλι ο λόγος που χρησιμοποιούμε πολλαπλά αντικείμενα οντοτήτων είναι για να είναι ξεκάθαρο το στιγμιότυπο που χρησιμοποιείται κάθε φορά, και για να δείξουμε ότι αναφερόμαστε στην ίδια κλάση (και

ότι θα γίνει μεταφορά μόνο ενός entity object στο sequence diagram) χρησιμοποιούμε τη σύνταξη "<Όνομα Οντότητας>: <Πεδίο ή/και περιγραφή στιγμιότυπου>". Πάλι εδώ σε κάθε use case που θέλουμε και μας διευκολύνει να δημιουργήσουμε μόνο μια κλάση από πολλά entity objects θα το αναφέρουμε από κάτω από το robustness διάγραμμα μαζί με αιτιολόγηση για τη κάθε περίπτωση.

1.1 Αναζήτηση βιβλίων / χρήστη / αιτήσεων



Σχήμα 1: Robustness Diagram: Αναζήτηση βιβλίων / χρήστη / αιτήσεων

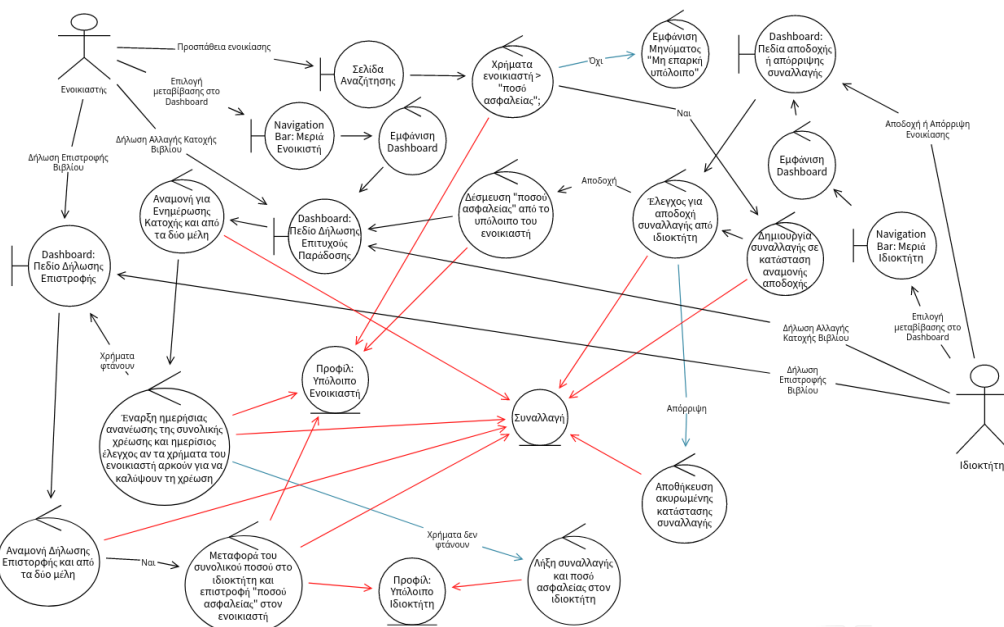
Σχετικά με το robustness diagram για την "Αναζήτηση βιβλίων / χρήστη / αιτήσεων", θέλαμε να αναφέρουμε πως τα συνοριακά αντικείμενα "Σελίδα Αναζήτησης", "Σελίδα Αναζήτησης: Βιβλία που Βρέθηκαν", "Σελίδα Αναζήτησης: Διαθέσιμοι Χρήστες από όπου Μπορεί να Ενοικιάσει ο Χρήστης", "Σελίδα Αναζήτησης: Προφίλ Χρήστη που Αναζητήθηκε μαζί με τις αξιο-

λογήσεις του”, ”Σελίδα Αναζήτησης: Αιτήσεις που Βρέθηκαν”, ”Σελίδα Αναζήτησης: Διαθέσιμοι Χρήστες για τους οποίους μπορεί να Εκπληρώσει Αίτηση” και ”Σελίδα Αναζήτησης: Κενή Αναζήτηση” όλες αναφέρονται στην ίδια οθόνη της ”Σελίδας Αναζήτησης”.

Ο λόγος που χρησιμοποιούμε πολλά συνοριακά για την ίδια οθόνη είναι για να φαίνεται καθαρά η ακολουθία των ενεργειών στο διάγραμμα και να μην σπάσουμε το τεστ των ακμών, μέσω της επιστροφής σε προηγούμενο συνοριακό.

Για λόγους απλότητας όμως, μεταφέροντας τα συνοριακά αντικείμενα ως κλάσεις στα sequence diagrams και στο class diagram σε επόμενο βήμα, και τέλος και στο κώδικα, θα μεταφέρουμε μόνο τη ”Σελίδα Αναζήτησης”. Γνωρίζουμε ότι θα μπορούσαμε να δημιουργήσουμε περισσότερες κλάσεις για τα αποτελέσματα για παράδειγμα, αλλά πιστεύουμε ότι θα οδηγούσε σε πολύ σύνθετη αρχιτεκτονική κώδικα, καθώς και οι επιπλέον κλάσεις που θα δημιουργούσαμε δεν θα μπορούσαν να επαναχρησιμοποιηθούν αλλού. Οπότε προτιμήσαμε να θεωρήσουμε τη ”Σελίδα Αναζήτησης (SearchPage κλάση αργότερα) ως μια κλάση μαζί με όλο της το περιεχόμενο.

1.2 Αποδοχή Προσφοράς Βιβλίου και Ενοικίαση



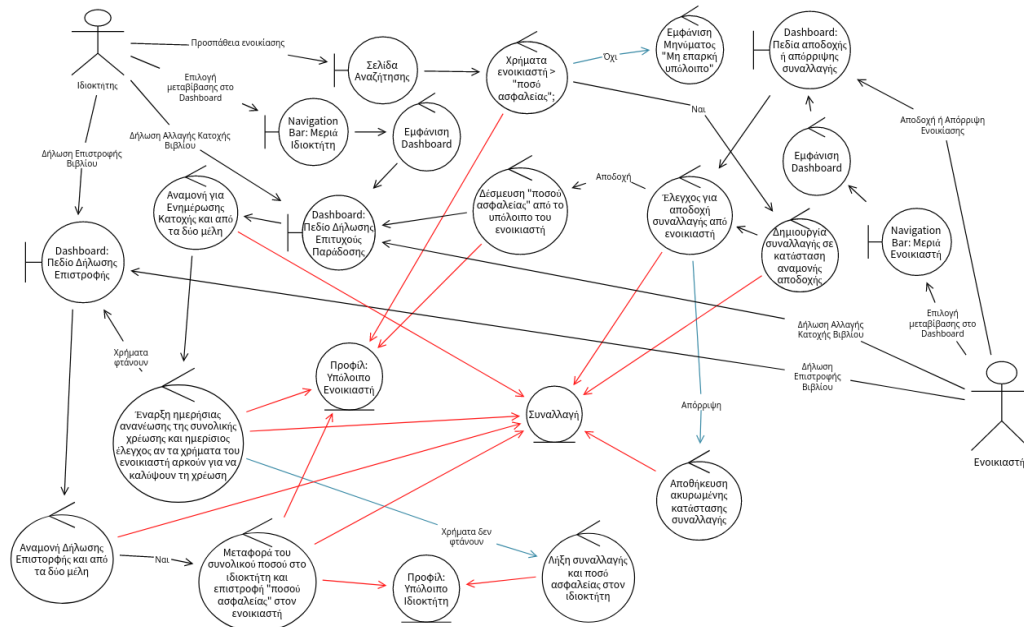
Σχήμα 2: Robustness Diagram: Αποδοχή Προσφοράς Βιβλίου και Ενοικίαση

Στο robustness diagram για τη "Αποδοχή προσφοράς Βιβλίου και Ενοικίαση" έχουμε κάνει πάλι παρόμοια παραδοχή για ορισμένα συνοριακά αντικείμενα όπως κάναμε στο κεφάλαιο 1.1, αυτή τη φορά όμως θεωρώντας τα συνοριακά αντικείμενα "Dashboard Πεδία αποδοχής ή απόρριψης συναλλαγής", "Dashboard: Πεδίο Δήλωσης Επιτυχούς Παράδοσης", "Dashboard: Πεδίο Δήλωσης Επιστροφής" να αναφέρονται σε διαφορετικά τμήματα της ίδιας οθόνης, του "Dashboard", που θα μεταφερθεί ως μια κλάση στο class diagram, sequence diagram και κώδικα.

Επίσης αναφέρουμε ότι τα συνοριακά αντικείμενα "Navigation Bar: Μεριά Ενοικιαστή" και "Navigation Bar: Μεριά Ιδιοκτήτη" αναφέρονται σε δύο διαφορετικά στιγμιότυπα της ίδιας κλάσης, ένα για κάθε χειριστή της εφαρμογής. Οπότε και αυτό μεταφέρεται σαν μια κλάση στο class diagram και robustness diagram και κώδικα (την GUI).

Τέλος τα αντικείμενα οντότητας "Προφίλ: Υπόλοιπο Ενοικιαστή" και "Προφίλ: Υπόλοιπο Ιδιοκτήτη" αναφέρονται πάλι σε δύο διαφορετικά στιγμιότυπα της μετέπειτα κλάσης που δημιουργήσαμε στο sequence diagram "User" (και μεταφέρθηκε στο class diagram και κώδικα), η οποία κλάση περιέχει πεδίο για υπόλοιπο χρήστη. Ο λόγος που και ο ενοικιαστής και ιδιοκτήτης είναι στιγμιότυπα της ίδιας κλάσης "User" είναι επειδή όλοι οι χρήστες μπορούν να είναι και ιδιοκτήτες και ενοικιαστές σε διαφορετικές συναλλαγές.

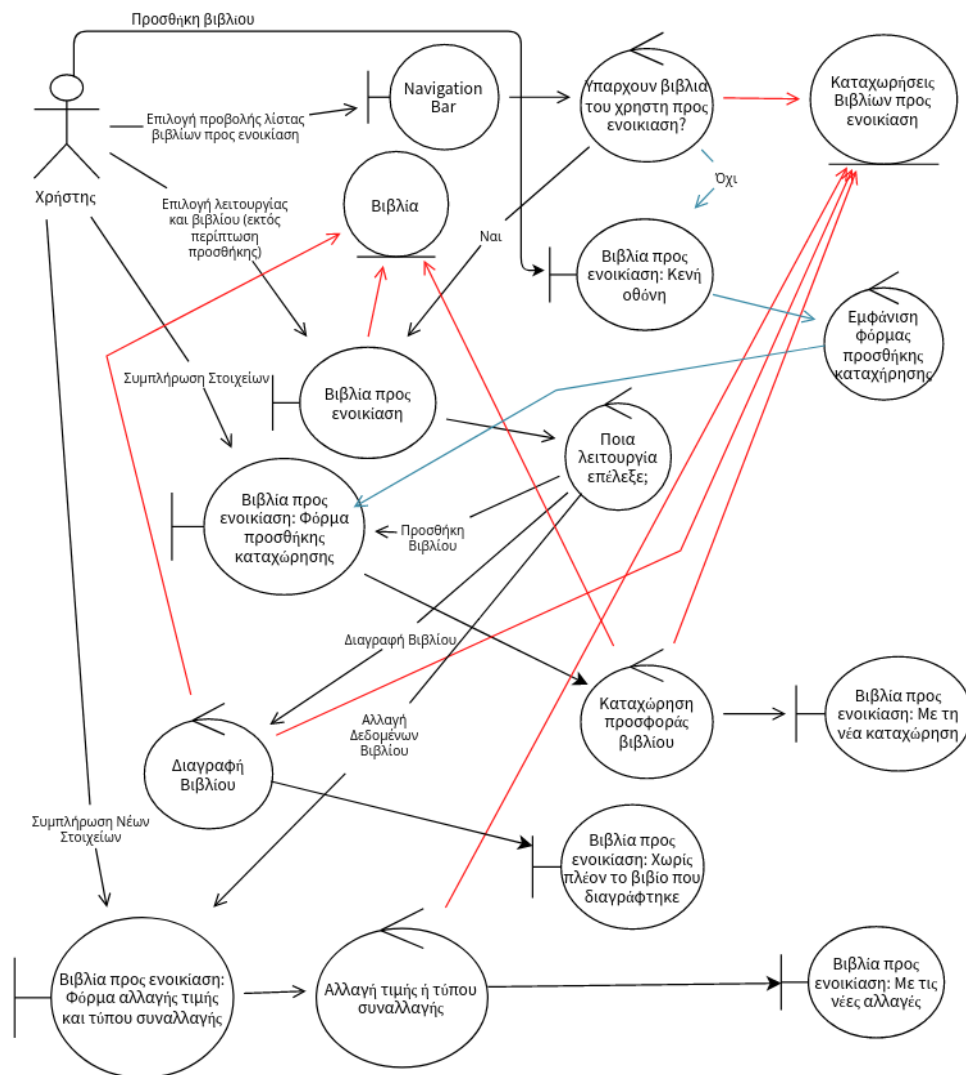
1.3 Αποδοχή Αίτησης Βιβλίου και Ενοικίαση



Σχήμα 3: Robustness Diagram: Αποδοχή Αίτησης Βιβλίου και Ενοικίαση

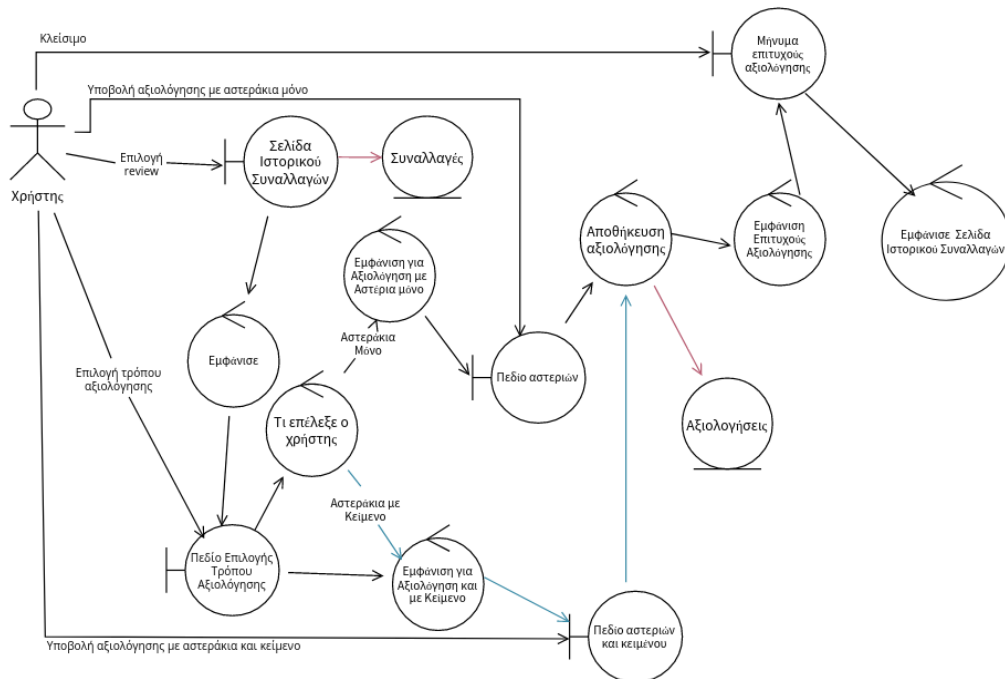
Στο robustness diagram για τη "Αποδοχή Αίτησης Βιβλίου και Ενοικίαση" ισχύουν οι ίδιες σημειώσεις και παραδοχές που ίσχυαν και για το robustness diagram στο κεφάλαιο 1.2, εφόσον τα δύο use cases "Αποδοχή Αίτησης Βιβλίου και Ενοικίαση" και "Αποδοχή Προσφοράς Βιβλίου και Ενοικίαση" αποτελούν δύο πτυχές της ίδιας διαδικασίας, που εξηγήσαμε στο τεχνικό κείμενο "Use-cases-v1.0" γιατί τα αναφέρουμε ξεχωριστά.

1.4 Διαχείριση των βιβλίων που προσφέρει ο χρήστης προς ενοικίαση από άλλους



Σχήμα 4: Robustness Diagram: Διαχείριση των βιβλίων που προσφέρει ο χρήστης προς ενοικίαση από άλλους

1.6 Αξιολόγηση άλλων χρηστών μετά από την ολοκλήρωση συναλλαγής

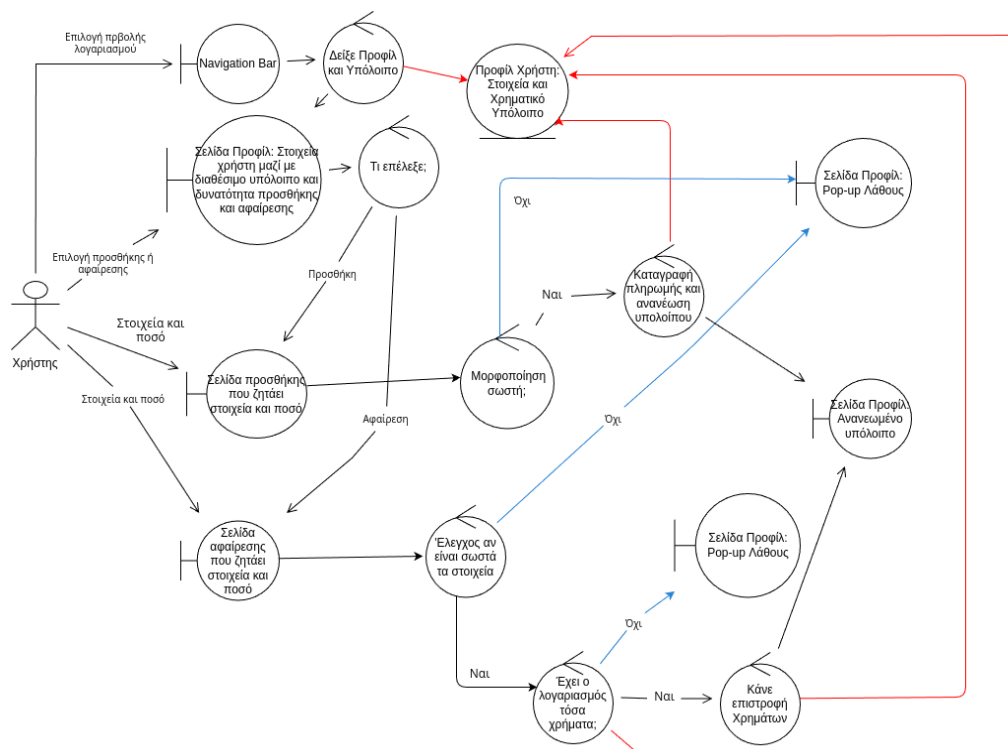


Σχήμα 6: Robustness Diagram: Αξιολόγηση άλλων χρηστών μετά από την ολοκλήρωση συναλλαγής

1.7 Προβολή και Επεξεργασία στοιχείων λογαριασμού χρήστη

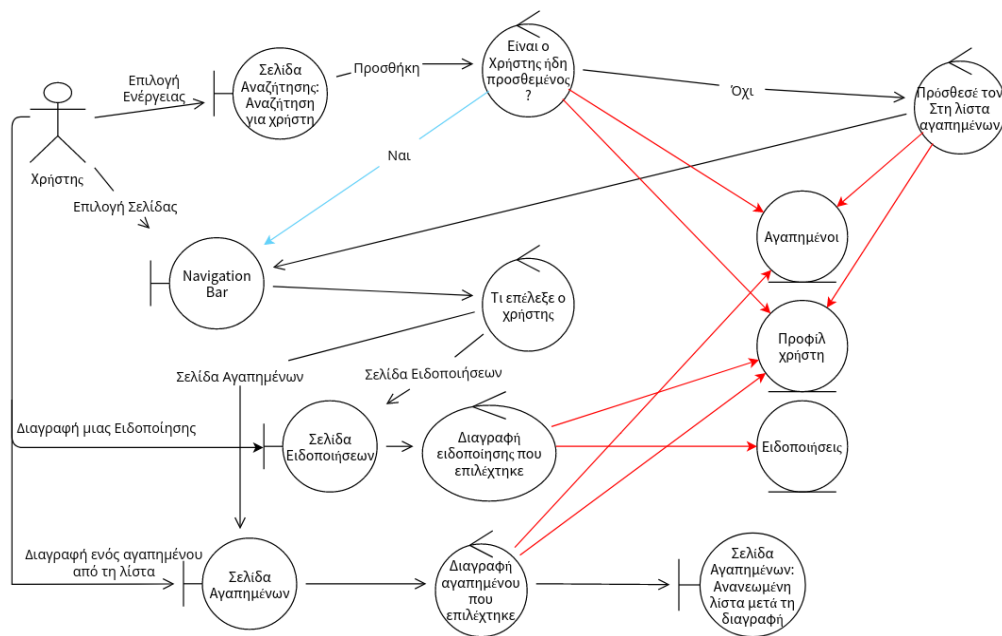
Σημείωση: Στο διάγραμμα αυτό τα δύο boundary objects "Οθόνη Προφίλ με Δυνατότητα Επεξεργασίας" και "Οθόνη προφίλ με ανανεωμένα στοιχεία" αναφέρονται στην ίδια οθόνη, πριν και μετά την αλλαγή αντίστοιχα. Ο λόγος που τα βάλαμε ξεχωριστά είναι για να φαίνεται η αλληλουχία των πράξεων.

1.8 Επεξεργασία χρηματικού υπολοίπου χρήστη



Σχήμα 8: Robustness Diagram: Επεξεργασία χρηματικού υπολοίπου χρήστη

1.9 Επεξεργασία λίστας αγαπημένων και χρήση συστήματος ειδοποιήσεων



Σχήμα 9: Robustness Diagram: Επεξεργασία λίστας αγαπημένων και χρήση συστήματος ειδοποιήσεων

3 Αλλαγές από έκδοση σε έκδοση

3.1 Από έκδοση v0.1 σε έκδοση v0.2

- Αλλαγή στο Robustness Diagram για "Αναζήτηση βιβλίων / χρήστη / αιτήσεων".
- Διαγραφή των Robustness Diagrams για "Αναζήτηση βιβλίων / χρήστη / αιτήσεων" και "Ενοικίαση βιβλίου από άλλο χρήστη" και αντικατάστασή τους με τα αντίστοιχα για "Αποδοχή Προσφοράς Βιβλίου και Ενοικίαση" και "Αποδοχή Αίτησης Βιβλίου και Ενοικίαση".
- Ανανέωση του Robustness Diagram για "Επεξεργασία χρηματικού υπολοίπου χρήστη".
- Ανανέωση του Robustness Diagram για "Επεξεργασία λίστας αγαπημένων και χρήση συστήματος ειδοποιήσεων".
- Ανανέωση του Robustness Diagram για "Προβολή και Επεξεργασία στοιχείων λογαριασμού χρήστη".
- Ανανέωση του Robustness Diagram για "Διαχείριση των βιβλίων που προσφέρει ο χρήστης προς ενοικίαση από άλλους".
- Διόρθωση λάθος αρίθμησης κεφαλαίων στο Κεφάλαιο 2.