



Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Εξάμηνο 7ο

Τελική εργασία στο μάθημα Τεχνολογία Πολυμέσων

## Περιγραφή Εφαρμογής

Στα πλαίσια της εργασίας θα υλοποιηθεί ένα σύστημα ηλεκτρονικής βιβλιοθήκης. Η εφαρμογή θα επιτρέπει στον διαχειριστή την παραμετροποίηση και εποπτεία του υλικού της βιβλιοθήκης. Οι χρήστες του συστήματος θα έχουν την δυνατότητα να αναζητήσουν βιβλία, να κάνουν αίτηση για το δανεισμό τους και να προσθέτουν σχόλια και βαθμολογία για το διαθέσιμο υλικό.

### A.1. Σχεδιασμός και υλοποίηση λογικής (30%)

Στη συνέχεια περιγράφονται οι δυνατότητες που μπορεί να έχει ο διαχειριστής του συστήματος ή οποιοσδήποτε χρήστης. Λειτουργίες Διαχειριστή

- Προσθήκη, διαγραφή και τροποποίηση των πληροφοριών ενός βιβλίου. Οι σχετικές πληροφορίες θα πρέπει να περιλαμβάνουν: τον τίτλο, συγγραφέα, εκδοτικό οίκο, ISBN, έτος έκδοσης, κατηγορία στην οποία ανήκει (π.χ. μυθιστόρημα) και τον αριθμό αντιτύπων. Για κάθε νέο βιβλίο η αρχική βαθμολογία θα είναι 0. Επιπροσθέτως, θα δίνεται η δυνατότητα διαγραφής ενός βιβλίου. Σε αυτή την περίπτωση πρέπει να αφαιρούνται από το σύστημα οι ανοιχτοί δανεισμοί για το βιβλίο που έχει διαγραφεί. Επιπλέον, ο διαχειριστής θα μπορεί να τροποποιήσει τα στοιχεία ενός βιβλίου, όμως από την τροποποίηση εξαιρούνται τα σχόλια και η βαθμολογία των χρηστών.
- Προσθήκη, διαγραφή και τροποποίηση κατηγορίας. Ο διαχειριστής θα μπορεί να ορίζει νέες κατηγορίες, δίνοντας το σχετικό τίτλο. Επιπλέον, θα μπορεί να τροποποιήσει τον τίτλο μιας κατηγορίας. Για λόγους απλότητας θεωρούμε ότι δεν υπάρχουν υποκατηγορίες. Επίσης, θα μπορεί να διαγράψει μια κατηγορία μαζί με τα βιβλία που ανήκουν σε αυτή. Επιπλέον, θα πρέπει να γίνεται η κατάλληλη διενήμερωση των ανοιχτών δανεισμών για τα βιβλία που διαγράφονται.

- Διαγραφή και τροποποίηση στοιχείων χρήστη. Σε περίπτωση διαγραφής χρήστη που έχει ανοιχτούς δανεισμούς θα πρέπει να ανανεώνεται αντίστοιχα και ο αριθμός των διαθέσιμων αντιγράφων για τα βιβλία που υπήρχε δανεισμός.
- Διαχείριση δανεισμών: Ο διαχειριστής θα μπορεί να βλέπει όλους τους ανοιχτούς δανεισμούς βιβλίων. Επίσης θα μπορεί να τερματίζει τον δανεισμό ενός βιβλίου. Σε αυτή την περίπτωση θα πρέπει να ανανεώνεται αντίστοιχα και ο αριθμός των διαθέσιμων αντιγράφων.

### Λειτουργίες Χρήστη

- Εγγραφή στο σύστημα. Ο χρήστης πρέπει να εισάγει κατά την εγγραφή του επιθυμητό όνομα χρήστη (username), κωδικό, όνομα, επώνυμο, ΑΔΤ και email. Η εγγραφή είναι απαραίτητη ώστε να μπορεί ο χρήστης να δανειστεί υλικό και να προσθέσει σχόλια και βαθμολογία.
- Αναζήτηση υλικού. Οι χρήστες έχουν τη δυνατότητα αναζήτησης βιβλίων με βάση οποιουσδήποτε συνδυασμούς των παρακάτω κριτηρίων: τίτλος, συγγραφέας και έτος έκδοσης.
- Δανεισμός βιβλίων. Σε περίπτωση που υπάρχουν διαθέσιμα αντίτυπα για κάποιο βιβλίο δίνεται στον χρήστη η δυνατότητα να κάνει κράτηση του υλικού ώστε να το δανειστεί. Το όριο των βιβλίων που μπορεί να δανειστεί είναι 2 και το χρονικό όριο δανεισμού είναι 5 ημέρες. Για κάθε βιβλίο που δανείζεται ο χρήστης μπορεί να προσθέσει σχόλια και να το βαθμολογήσει στην κλίμακα 1 έως 5.

## **A.2. Αποθήκευση και ανάκτηση πληροφοριών εφαρμογής (20%)**

Μια αντίστοιχη εφαρμογή στην πραγματικότητα θα υποστηρίζονταν από ένα εξειδικευμένο σύστημα διαχείρισης δεδομένων, όπως π.χ. κάποια βάση δεδομένων. Ωστόσο, στα πλαίσια της εργασίας θα υλοποιήσετε έναν πιο απλό μηχανισμό που θα βασίζεται στην χρήση μιας σειράς από αρχεία για την αποθήκευση και ανάκτηση όλων των πληροφοριών της εφαρμογής.

Σε αυτή την διαδικασία θα αξιοποιηθεί η δυνατότητα που παρέχει η Java για την μετατροπή αντικειμένων σε μια σειρά από bytes που μπορούν στη συνέχεια, μεταξύ άλλων, να αποθηκευτούν σε ένα μέσο μόνιμης αποθήκευσης (π.χ. αρχείο στο filesystem). Η διαδικασία μετατροπής των αντικειμένων σε bytes ονομάζεται *object serialization*. Αντίστοιχα, όταν ένα αντικείμενο μετατραπεί σε μια σειρά από bytes μπορούμε να ακολουθήσουμε την αντίστροφη διαδικασία προκειμένου να δημιουργήσουμε ένα αντίγραφο του αντικειμένου στο JVM. Η διαδικασία ανάκτησης ενός αντικειμένου από την “σειριοποίηση” του ονομάζεται *object deserialization*. Περισσότερες πληροφορίες μπορείτε να βρείτε στις διαφάνειες του εργαστηρίου στην ενότητα “Java I/O”.

Αρχικά θα πρέπει να αποφασίσετε το σύνολο των αρχείων που θα χρησιμοποιηθούν για την αποθήκευση των αντικειμένων της εφαρμογής σας. Τα ειδικά αρχεία θεωρούμε ότι αποθηκεύονται σε ένα φάκελο με όνομα **“medialab”**. Στην συνέχεια πρέπει να υλοποιήσετε μέσω των κατάλληλων κλάσεων τις μεθόδους που θα σας επιτρέπουν να γίνεται ανάκτηση των αντικειμένων που έχουν τα αρχεία καθώς και ανανέωση των αρχείων ώστε να μπορεί να διατηρείται η συνολική κατάσταση του συστήματος μεταξύ διαδιαχικών εκτελέσεων της εφαρμογής.

Στη συνέχεια περιγράφουμε την λογική που πρέπει να υλοποιηθεί για την ανάκτηση και ανανέωση των δεδομένων της εφαρμογής.

- Αρχικοποίηση εφαρμογής: Θα πρέπει να γίνεται η ανάκτηση του συνόλου των πληροφοριών που βρίσκονται στα ειδικά αρχεία και ταυτόχρονα η αρχικοποίηση των αντίστοιχων αντικειμένων (desearilization) στην εφαρμογή σας.
- Εκτέλεση εφαρμογής: Η εφαρμογή θα χρησιμοποιεί τις πληροφορίες κατάστασης που έχουν ανακτηθεί στην μνήμη του προγράμματος κατά την αρχικοποίηση. Όλες οι λειτουργίες που σχετίζονται με την ανάγνωση, δημιουργία, τροποποίηση, αναζήτηση και διαγραφή δεδομένων της εφαρμογής (π.χ., βιβλία, χρήστες, δανεισμούς, σχόλια, βαθμολογία) θα εκτελούνται με βάση τις πληροφορίες που υπάρχουν στην μνήμη του προγράμματος.
- Τερματισμός εφαρμογής: Η ανανέωση των ειδικών αρχείων με τις πληροφορίες κατάστασης του συστήματος θα γίνεται αποκλειστικά πριν τον τερματισμό της εφαρμογής. Η υλοποίηση θα πρέπει να αποθηκεύει στα ειδικά αρχεία την συνολική κατάσταση της εφαρμογής κατά την στιγμή του τερματισμού μέσω του serialization των σχετικών αντικειμένων.

Τέλος, για την διευκόλυνση της διαδικασίας βαθμολόγησης θα πρέπει η υλοποίηση της εφαρμογής να υποστηρίζει την σύνδεση ενός default διαχειριστή με στοιχεία:

- **Username**: medialab
- **Password**: medialab\_2024

### A.3. Δημιουργία γραφικής διεπαφής (30%)

Θα πρέπει να σχεδιάσετε και να υλοποιήσετε την κατάλληλη γραφική διεπαφή χρήστη (Graphical User Interface - GUI) χρησιμοποιώντας το πλαίσιο JavaFX [1][2].

**Σημείωση:** Στη συνέχεια παρουσιάζονται οι βασικές προδιαγραφές για την γραφική διεπαφή, για όλες τις λεπτομέρειες της τελικής υλοποίησης μπορείτε να κάνετε όποιες επιλογές θέλετε σχετικά με την εμφάνιση και τη γενικότερη αλληλεπίδραση του χρήστη με την εφαρμογή, χωρίς καμία επίπτωση στην τελική βαθμολογία. Για παράδειγμα, μπορείτε να επιλέξετε μια απλή απεικόνιση για τα διάφορα στοιχεία ή να συνδυάσετε

διάφορα χαρακτηριστικά από το JavaFX ώστε να δημιουργήσετε ένα αποτέλεσμα που να αντιστοιχεί σε μια σύγχρονη εφαρμογή. Σε κάθε περίπτωση, δεν υπάρχει λόγος να κάνετε πολύπλοκο το συγκεκριμένο μέρος της εργασίας.

Η γραφική διεπαφή που θα εμφανίζεται κατά την έναρξη της εφαρμογής θα πρέπει να υποστηρίζει τις παρακάτω λειτουργίες:

- Εμφάνιση των 5 βιβλίων με το μεγαλύτερο μέσο όρο βαθμολογίας: θα πρέπει για κάθε βιβλίο να εμφανίζεται ο τίτλος, ο συγγραφέας, το ISBN, ο μέσος όρος της βαθμολογίας και το πλήθος των χρηστών που το έχουν βαθμολογήσει.
- Εγγραφή νέου χρήστη: θα πρέπει να υπάρχει η αντίστοιχη φόρμα που θα επιτρέπει την εγγραφή νέου χρήστη με βάση τις προδιαγραφές από την ενότητα A.1. Δεν επιτρέπεται να υπάρχουν χρήστες με το ίδιο email ή ADT. Αν υπάρχει κάποιο σφάλμα θα πρέπει να ενημερώνεται ο χρήστης με κατάλληλο μήνυμα.
- Σύνδεση χρήστη: θα πρέπει να υπάρχει η αντίστοιχη φόρμα για την σύνδεση ενός χρήστη με βάση το όνομα χρήστη και κωδικό. Αν υπάρχει κάποιο σφάλμα θα πρέπει να ενημερώνεται ο χρήστης με κατάλληλο μήνυμα.

Όταν ένας χρήστης συνδεθεί επιτυχώς στην εφαρμογή η γραφική διεπαφή θα πρέπει να υποστηρίζει τις παρακάτω λειτουργίες:

- Αναζήτηση βιβλίου: θα πρέπει να υπάρχει η αντίστοιχη φόρμα που θα επιτρέπει την αναζήτηση με βάση τις προδιαγραφές από την ενότητα A.1. Τα αποτελέσματα θα πρέπει να περιλαμβάνουν: τίτλο, συγγραφέα, ISBN, μέσος όρος βαθμολογίας και το πλήθος των χρηστών που το έχουν βαθμολογήσει.
- Παρουσίαση βιβλίου και δανεισμός: για κάθε αποτέλεσμα μιας αναζήτησης ο χρήστης θα μπορεί να επιλέξει είτε να δει όλες τις διαθέσιμες πληροφορίες (τίτλο, συγγραφέα, εκδοτικό οίκο, ISBN, έτος έκδοσης, κατηγορία, διαθέσιμος αριθμός αντιτύπων, μέσος όρος βαθμολογίας, πλήθος χρηστών που βαθμολόγησαν και σχόλια χρηστών) είτε να προχωρήσει στον δανεισμό του με βάση τις προδιαγραφές από την ενότητα A.1.
- Επισκόπηση ανοιχτών δανεισμών χρήστη: θα πρέπει να παρουσιάζεται το υλικό που έχει δανειστεί ο χρήστης και δεν έχει ακόμη κλείσει ο δανεισμός από τον διαχειριστή. Οι πληροφορίες πρέπει να περιλαμβάνουν τον τίτλο, το ISBN, την ημερομηνία δανεισμού και την μέγιστη δυνατή ημερομηνία επιστροφής.
- Βαθμολογία και σχολιασμός: για κάθε ανοιχτό δανεισμό βιβλίου ο χρήστης θα πρέπει να μπορεί να προσθέσει κάποιο σύντομο σχόλιο και να βαθμολογήσει το βιβλίο στην κλίμακα από 1 έως 5.

Όταν ένας διαχειριστής συνδεθεί επιτυχώς στην εφαρμογή η γραφική διεπαφή θα πρέπει να υποστηρίζει τις παρακάτω λειτουργίες:

- Διαχείριση βιβλίων: η εφαρμογή πρέπει να παρουσιάζει την λίστα με τα διαθέσιμα βιβλία ανά κατηγορία. Επίσης θα πρέπει να υπάρχει η δυνατότητα ο διαχειριστής να ορίζει νέα βιβλία, να τροποποιεί τις πληροφορίες ενός βιβλίου και να διαγράφει κάποιο βιβλίο. Η υλοποίηση πρέπει να γίνει σύμφωνα με την αντίστοιχη λογική από την ενότητα A.1.
- Διαχείριση κατηγοριών: η εφαρμογή πρέπει να παρουσιάζει την λίστα με τις κατηγορίες. Ακόμη θα πρέπει ο διαχειριστής να μπορεί να ορίζει νέες κατηγορίες, να αλλάζει το όνομα μιας κατηγορίας και να διαγράφει κατηγορίες. Η υλοποίηση πρέπει να γίνει σύμφωνα με την αντίστοιχη λογική από την ενότητα A.1.
- Διαχείριση χρηστών: η εφαρμογή πρέπει να παρουσιάζει την λίστα με τους διαθέσιμους χρήστες. Ακόμη θα πρέπει ο διαχειριστής να μπορεί να τροποποιεί τα στοιχεία των χρηστών και να διαγράφει χρήστες από το σύστημα. Η υλοποίηση πρέπει να γίνει σύμφωνα με την αντίστοιχη λογική από την ενότητα A.1.
- Διαχείριση δανεισμών: η εφαρμογή πρέπει να παρουσιάζει όλους τους ανοιχτούς δανεισμούς βιβλίων. Επίσης θα πρέπει ο διαχειριστής να μπορεί να ορίζει το τέλος του δανεισμού ενός βιβλίου. Σε αυτή την περίπτωση θεωρούμε ότι το βιβλίο έχει επιστραφεί και πρέπει να ανανεώνεται αντίστοιχα και ο αριθμός των διαθέσιμων αντιγράφων.

Τέλος, θα πρέπει η υλοποίηση της γραφικής σας διεπαφής κατά τον τερματισμό της εφαρμογής να εξασφαλίζει την ανανέωση των πληροφοριών του συστήματος στα ειδικά αρχεία, σύμφωνα με την διαδικασία που περιγράφεται στην ενότητα A.2.

#### A.4. Λοιπές απαιτήσεις (20%)

- Η υλοποίηση θα πρέπει να ακολουθεί τις αρχές σχεδίασης του αντικειμενοστραφούς προγραμματισμού (OOP design principles).
- Σε μια κλάση της επιλογής σας θα πρέπει κάθε public μέθοδος που περιέχει να είναι τεκμηριωμένη σύμφωνα με τις προδιαγραφές του εργαλείου javadoc [3].

**Σημείωση:** Για οτιδήποτε δεν είναι ξεκάθαρο από την εκφώνηση μπορείτε να κάνετε τις δικές σας παραδοχές και υποθέσεις. Η εκφώνηση περιγράφει τις βασικές απαιτήσεις για την εφαρμογή, ωστόσο μπορείτε να κάνετε δικές σας σχεδιαστικές παραδοχές προσπαθώντας να κάνετε πιο ρεαλιστική την εφαρμογή, χωρίς παράλληλα να κάνετε πολύπλοκη την υλοποίηση.

## Παραδοτέα

- Το project (του IDE της επιλογής σας) με την υλοποίηση της εφαρμογής.
- Μια σύντομη (το πολύ 2 σελίδες) αναφορά που θα περιέχει μια γενική περιγραφή του σχεδιασμό της υλοποίησης, θα αναφέρετε όποια λειτουργικότητα δεν έχετε υλοποιήσει καθώς και όποιες επιπρόσθετες παραδοχές έχετε κάνει. Σε καμία περίπτωση δεν θα πρέπει να συμπεριλάβετε τμήματα κώδικα.

## Αναφορές

[1] <https://docs.oracle.com/javase/8/javafx/get-started-tutorial/jfx-overview.htm>

[2] [https://docs.oracle.com/javafx/2/get\\_started/jfxpub-get\\_started.htm](https://docs.oracle.com/javafx/2/get_started/jfxpub-get_started.htm)

[3] <https://www.oracle.com/technical-resources/articles/java/javadoc-tool.html>

Book

Category

Loan

User

Admin