

Εργασία Java-Διαχείριση Βιβλιοθήκης Πανεπιστημίου

Κωσταντίνος Τάσσης (Ε24170)
Άγγελος Καραχάλης (Ε24059)

Περιεχόμενα

- i. Εισαγωγή
- ii. Υλοποίηση
- iii. Εγχειρίδιο Χρήσης
- iv. Μελλοντικές Βελτιώσεις

Εισαγωγή :

Ο σκοπός του προγράμματος αυτού είναι η δημιουργία ενός λειτουργικού συστήματος βιβλιοθήκης ενός πανεπιστημίου μέσω του οποίου θα μπορούν οι χρήστες να δανειστούν βιβλία και να πληρώσουν τυχόν πρόστιμα που έχουν. Επίσης βοηθάει στην καταγραφή των στοιχείων των φοιτητών και στην επεξεργασία αυτών και στην επιβολή προστίμων σε όσους δεν επιστρέφουν τα βιβλία στην ημερομηνία που έπρεπε. Τέλος, επιτρέπει στους βιβλιοθηκονόμους να προσθέσουν, να επεξεργαστούν και να διαγράψουν βιβλία από το σύστημα.

Στο αρχείο αυτό θα περιγράψω περιληπτικά την λειτουργία του κώδικα θα δώσω επίσης οδηγίες στο πώς μπορεί ο χρήστης να κινηθεί μέσα στο σύστημα με παραδείγματα και τέλος μελλοντικές βελτιώσεις που θα μπορούσαν να γίνουν στον κώδικα.

Υλοποίηση :

Η υλοποίηση του προγράμματος έγινε με την δημιουργία δύο πακέτων από τα οποία το ένα περιείχε τις “οντότητες” που εμφανίζονται μέσα στο πρόγραμμα όπως οι φοιτητές, τα βιβλία κλπ. Το δεύτερο πακέτο περιείχε τις κλάσεις οι οποίες δημιουργούσαν τα παράθυρα όταν τρέχει το πρόγραμμα .

1ο Package (lab.unipi.gui.JavaFXLab):

1. App.java :

Η κλάση App είναι το σημείο εκκίνησης της JavaFX εφαρμογής. Η κλάση αυτή περιέχει τις λίστες στις οποίες αποθηκεύοντουσαν τα στοιχεία των φοιτητών, βιβλίων, δανεισμών και προστίμων. Αρχικοποιεί τις σκηνές πλοήγησης για το κύριο μενού και τις επιμέρους λειτουργίες της βιβλιοθήκης

2. Book.java :

Η κλάση αυτή περιέχει τα στοιχεία που πρέπει να έχει κάθε αντικείμενο Book.

3. FeePolicy.java :

Η κλάση FeePolicy χρησιμοποιήθηκε για τον υπολογισμό του προστίμου.

4. Fine.java :

Η κλάση αυτή αποτελεί τα στοιχεία που πρέπει να έχει ένα πρόστιμο.

5. Librarian.java :

Η κλάση Librarian είναι μια υποκλάση της κλάσης User προσθέτοντας κάποια περαιτέρω πεδία αποκλειστικά για τους βιβλιοθηκονόμους.

6. Loan.java :

Η κλάση Loan είναι τα στοιχεία ενός δανεισμού βιβλίου από έναν φοιτητή

7. Student.java :

Η κλάση Student περιέχει τα στοιχεία που πρέπει να έχει κάποιος φοιτητής και αποτελεί υποκλάση της user

8. User.java :

Η κλάση User είναι μια γενική κλάση για όλους τους χρήστες της βιβλιοθήκης όπου πολλά από τα στοιχεία που περιλαμβάνει επεκτείνονται τόσο στους φοιτητές όσο και στους βιβλιοθηκονόμους.

2o Package (lab.unipi.gui.JavaFXLab.scenes):

- SceneCreator.java : Το SceneCreator είναι μια κλάση που χρησιμοποιείτε σε κάθε κλάση “τύπου” SceneCreator καθώς εκτελεί την βασική λειτουργία του ότι δημιουργεί το παράθυρο πάνω στο οποίο γίνονται όλες οι λειτουργίες του προγράμματος. Αποθηκεύει τις διαστάσεις κάθε σκηνής (width, height) και ορίζει τη μέθοδο createScene() που υλοποιείται από τις υποκλάσεις.
- MainSceneCreator.java : Η κλάση αυτή δημιουργεί το κεντρικό παράθυρο μέσω του οποίου ο χρήστης μπορεί να κινηθεί σε όλες τις λειτουργίες που προσφέρονται από το σύστημα αυτό με την χρήση τεσσάρων κουμπιών. Κάθε κουμπί οδηγεί στη σχετική σκηνή μέσω του App.primaryStage.
- BookSceneCreator.java : Η κλάση BookSceneCreator δημιουργεί τη σκηνή για τη διαχείριση βιβλίων, οι χρήστες μπορούν είτε να καταχωρίσουν ένα νέο βιβλίο, είτε να επεξεργαστούν τα στοιχεία των βιβλίων, είτε να αφαιρέσουν κάποιο βιβλίο από το σύστημα.
- StudentSceneCreator.java : Σε αυτή την σκηνή μπορεί ο βιβλιοθηκονόμος να παρακολουθεί και να επεξεργάζεται τα στοιχεία των φοιτητών. Μέσω της σκηνής αυτής μπορεί να πραγματοποιήσει εγγραφή ενός νέου φοιτητή, να ενημερώσει τα υπάρχοντα στοιχεία φοιτητών και να έχει πρόσβαση στο ιστορικό δανεισμών.
- LoanSceneCreator.java : Η κλάση LoanSceneCreator αποτελεί το περιβάλλον διαχείρισης δανεισμών βιβλίων από τον βιβλιοθηκονόμο. Σε αυτή τη σκηνή μπορεί ο χρήστης να δημιουργεί ένα καινούργιο δανεισμό με αναζήτηση του βιβλίου που θα δανειστεί και την αναζήτηση του φοιτητή που θα το δανειστεί. Ακόμη μέσω αυτής της κλάσης πραγματοποιείτε η επιστροφή βιβλίων, η

προβολή λίστας των δανεισμών (ενεργοί , καθυστερημένοι και ολοκληρωμένοι) και ο υπολογισμός και καταχώρηση προστίμων σε περίπτωση καθυστέρησης επιστροφής βιβλίου.

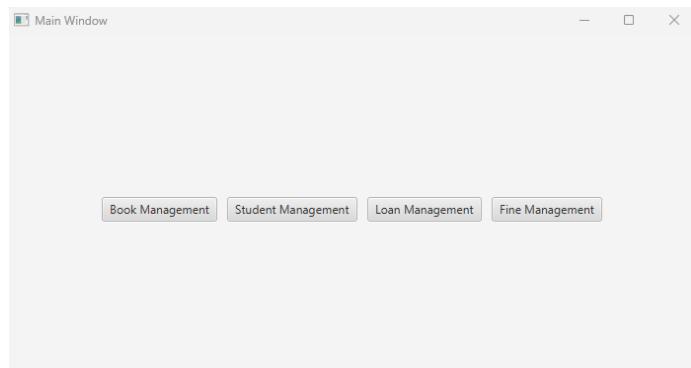
- PaymentSceneCreator.java : Στο παράθυρο αυτό ο βιβλιοθηκονόμος μπορεί να παρακολουθήσει τις πληρωμές προστίμων των φοιτητών και να τις καταχωρίσει . Σε αυτή τη σκηνή, ο χρήστης μπορεί να παρακολουθήσει τυχόν μη πληρωμένα πρόστιμα , να καταχωρίσει κάποιο πρόστιμο ως πληρωμένο και να παρακολουθήσει όλα τα πληρωμένα πρόστιμα για ελεγκτικούς σκοπούς.
- Η οργάνωση όλων των παραθύρων και των διαφόρων “εργαλείων” που περιλαμβάνουν έγινε με χρήση TableView , TextField , Button και Alert , ώστε τα παράθυρα να είναι πιο φιλικά στην χρήση και να διευκολύνουν τον χρήστη. Δόθηκε προσοχή στην τυχόν καταχώρηση λάθος στοιχείων από τον χρήστη και τέλος σε διάφορες κλάσεις χρησιμοποιήθηκαν βοηθητικές μέθοδοι για την ευκολότερη οργάνωση και ανάγνωση του προγράμματος.

Εγχειρίδιο Χρήσης :

1. Αρχικό Παράθυρο (Main Window):

Κατά την εκκίνηση του προγράμματος εμφανίζεται το αρχικό παράθυρο στο οποίο υπάρχουν 4 κουμπιά. Το κάθε κουμπί μας πηγαίνει στην αντίστοιχη λειτουργεία.

Για διαχείριση Βιβλίων πατήστε το κουμπί “Book Management”



Για διαχείριση Φοιτητών πατήστε το κουμπί “Student Management”

Για διαχείριση δανεισμών πατήστε το κουμπί “Loan management”

Για διαχείριση προστίμων πατήστε το κουμπί “Fine Management”

2. Παράθυρο Διαχείρισης Βιβλίων (Book Management Window):

Πατώντας το κουμπί “Book Management” προχωράμε στο παράθυρο διαχείρισης βιβλίων. Στο συγκεκριμένο παράθυρο εμφανίζεται ένας πίνακας με όλα τα βιβλία που υπάρχουν το σύστημα. Ο χρήστης έχει 4 επιλογές:

- New Book: Αφού ο χρήστης έχει συμπληρώσει όλα τα πεδία (title, author, isbn etc.) μπορεί να προσθέσει κάποιο καινούργιο βιβλίο στο σύστημα. **Προσοχή!** Στα πεδία “Year” και “Availability” πρέπει να εισαχθεί μόνο αριθμός integer και τιμή Boolean (true or false) αντίστοιχα.
- Update: Με το κουμπί update ο χρήστης έχει την δυνατότητα αφού εισάγει πρώτα ένα έγκυρο “ISBN” να αλλάξει οποιοδήποτε άλλο στοιχείο του συγκεκριμένου βιβλίου.
- Delete: Το κουμπί Delete δίνει την δυνατότητα διαγραφής ενός βιβλίου συμπληρώνοντας στο κατάλληλο πεδίο το αντίστοιχο “ISBN”.
- Back: Το τελευταίο κουμπί “Back” δίνει την δυνατότητα στον χρήστη να επιστρέψει στην αρχική οθόνη (“Main window”)

Book Management Window			
Title	Author	ISBN	Publisher
To kill a mockingbird	Harper Lee	123-456-789-1	Penguin Random
1984	George Orwell	123-456-789-2	Harper Collins
The Lord of the Rings	J.R.R. Tolkien	123-456-789-3	Simon and Schuster
The Great Gatsby	F. Scott Fitzgerald	123-456-789-4	Macmillan Publishing
Brave New World	Aldous Huxley	123-456-789-5	Hachette Book Group
Pride and Prejudice	Jane Austen	123-456-789-6	John Wiley and Sons
The Catcher in the Rye	J.D. Salinger	123-456-789-7	Merriam-Webster
Jane Eyre	Charlotte Brontë	123-456-789-8	Scholastic
The Alchemist	Paulo Coelho	123-456-789-9	Pearson
Harry Potter and the Sorcerer's Stone	J.K. Rowling	123-456-789-0	Houghton Mifflin

New Book Update Delete Back

3. Παράθυρο Διαχείρισης Φοιτητών (Student Management Window):

Μέσα από το “Student Management” κουμπί, προχωράμε στο Student management window. Στο συγκεκριμένο παράθυρο εμφανίζονται όλοι οι φοιτητές που είναι περασμένοι στο σύστημα. Ο χρήστης έχει την δυνατότητα:

- Να προσθέσει νέο Φοιτητή στο σύστημα με την χρήση του “New Student” κουμπιού, αφού έχει συμπληρώσει σωστά όλα τα απαραίτητα πεδία (Code, Name, Surname, Birth Date etc.) **Προσοχή!** Στο πεδία Code πρέπει να εισαχθεί αποκλειστικά αριθμός Integer.
- Να επεξεργαστεί τα στοιχεία ενός φοιτητή αφού συμπληρώσει ένα έγκυρο “Code” επεξεργαστεί τα απαραίτητα στοιχεία και πατήσει το κουμπί “Update Info”.
- Να επιλέξει έναν Φοιτητή από τον Πίνακα “Student List” και με την χρήση του “Loan History” κουμπιού να εμφανίσει στον αντίστοιχο πίνακα το ιστορικό δανεισμών του συγκεκριμένου φοιτητή.
- Back: Το τελευταίο κουμπί “Back” δίνει την δυνατότητα στον χρήστη να επιστρέψει στην αρχική οθόνη (“Main window”)

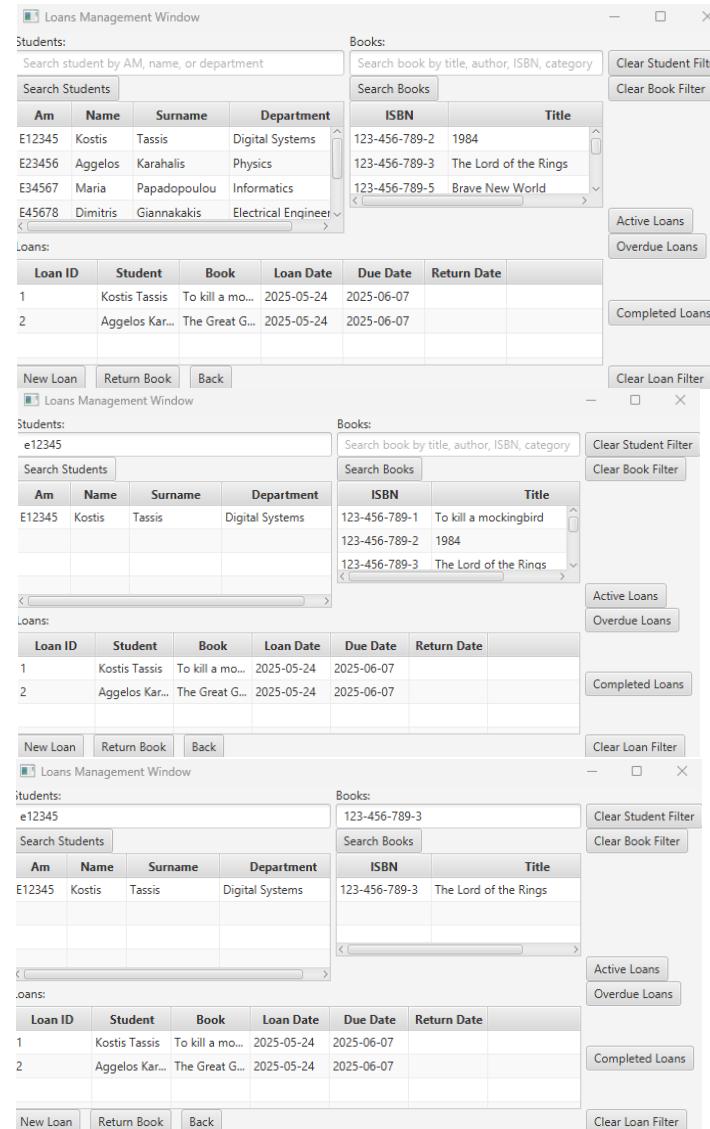
Student Management Window							
Student List							
Code	Name	Surname	Birth Date	Phone	Email	Am	Dep
1	Kostis	Tassis	10-05-2006	6948392203	kostis@unipi.com	E12345	Di...
2	Aggelos	Karahalis	15-09-2006	6920394219	aggelos@unipi.com	E23456	Pr...
3	Maria	Papadopoulou	12-04-2000	6941234567	maria@unipi.com	E34567	In...
4	Dimitris	Giannakakis	05-08-2000	6909876543	dimitris@unipi.com	F45678	Fl...

Loan History					
Loan ID	Student	Book	Loan Date	Due Date	Return Date
No content in table					

New Student Update Info Loan History Back

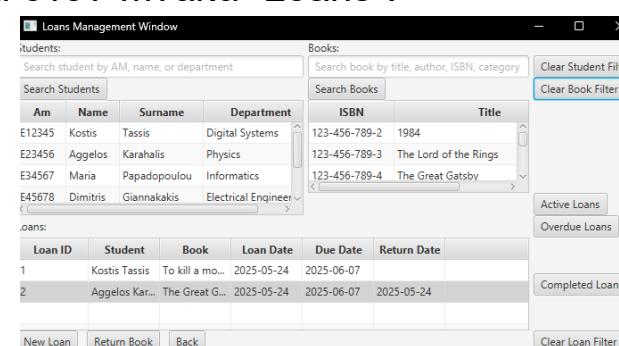
4. Παράθυρο Διαχείρισης Δανεισμών (Loan Management Window):

Πατώντας το κουμπί “Loan Management” βρισκόμαστε στο παράθυρο διαχείρισης δανεισμών. Στο συγκεκριμένο παράθυρο υπάρχουν τρείς πίνακες, ένας για την λίστα των μαθητών, ένας για την λίστα των βιβλίων, και ένας για όλους τους δανεισμούς. Στους πίνακες των φοιτητών και των βιβλίων έχει την δυνατότητα ο χρήστης να αναζητήσει έναν φοιτητή (με βάση το ΑΜ, το όνομα ή το τμήμα φοίτησης) και να αναζητήσει ένα βιβλίο (με βάση τον τίτλο, τον συγγραφέα, τον ειδικό κωδικό ISBN, και την κατηγορία). Δηλαδή αφού ο χρήστης πληκτρολογήσει το επιθυμητό στοιχείο μέσα στο πεδίο “Search Student”, πατάει το αντίστοιχο κουμπί “Search Student” ώστε να εμφανιστεί στον πίνακα μόνο ο επιθυμητός φοιτητής. Το ίδιο ισχύει και για την αναζήτηση βιβλίων. Με τα κουμπιά “Clear Student Filter” και “Clear Book Filter” οι δύο πίνακες δείχνουν πάλι όλους τους μαθητές και όλα τα βιβλία.



Οι βασικές λειτουργίες του παραθύρου είναι οι εξής:

- **New Loan:** Επιλέγοντας έναν φοιτητή και ένα βιβλίο από κάθε πίνακα ο χρήστης μπορεί να δημιουργήσει έναν δανεισμό. Αφού δημιουργηθεί ο δανεισμός εμφανίζεται στον πίνακα “Loans”.
- **Return Book:** Ο χρήστης πρέπει να επιλέξει έναν δανεισμό από τον πίνακα “Loans” και να πατήσει το συγκεκριμένο κουμπί ώστε να ολοκληρωθεί η επιστροφή του βιβλίου.
- **Back:** Το τελευταίο κουμπί “Back” δίνει την δυνατότητα στον χρήστη να επιστρέψει στην αρχική οθόνη (“Main window”)

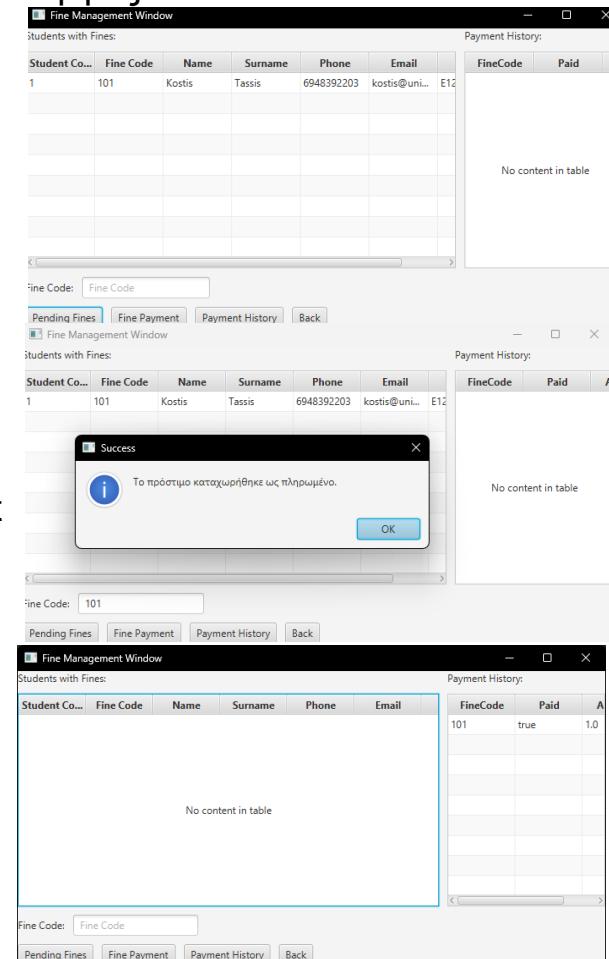


Όταν έχουμε μεγάλο αριθμό δανεισμών είναι σημαντικό με κάποιο τρόπο να «φιλτράρουμε» τον πίνακα ώστε να διαχωρίζουμε τους «ενεργούς» δανεισμούς, από τους «αργοπορημένους» δανεισμού και από τους «ολοκληρωμένους» δανεισμούς. Έτσι με τα αντίστοιχα κουμπιά “Active Loans”, “Overdue Loans” και “Completed Loans” εμφανίζονται μόνο οι αντίστοιχοι δανεισμοί. Τέλος, με το κουμπί “Clear Loan Filter” επιστρέφουμε τον πίνακα “Loans” στην αρχική του μορφή.

5. Παράθυρο Διαχείρισης Προστίμων (Fine Management Window):

Στο παράθυρο αυτό εμφανίζονται δύο πίνακες “Students With Fines”, “Payment History”. Στον πίνακα “Student With Fines” εμφανίζονται όλοι οι φοιτητές που έχουν απλήρωτα πρόστιμα. Στον πίνακα “Payment History” εμφανίζονται όλα τα πληρωμένα πρόστιμα. Λειτουργίες:

- Pending Fines: Με το κουμπί αυτό ανανεώνεται ο πίνακας “Student with Fines” και εμφανίζονται όλοι οι φοιτητές με απλήρωτα πρόστιμα.
- Fine Payment: Εισάγοντας σωστά το “Fine Code” με το κουμπί Fine Payment θέτω το συγκεκριμένο πρόστιμο ως πληρωμένο.
- Payment History: Με το συγκεκριμένο κουμπί ανανεώνεται ο πίνακας “Payment History”, και εμφανίζονται όλα τα πληρωμένα πρόστιμα.
- Back: Το τελευταίο κουμπί “Back” δίνει την δυνατότητα στον χρήστη να επιστρέψει στην αρχική οθόνη (“Main window”)



Μελλοντικές Βελτιώσεις :

1. Εμφάνιση ειδοποιήσεων όταν ένας φοιτητής είτε ξεπερνά την αναμενόμενη ημέρα επιστροφής του βιβλίου είτε όταν φτάνει κοντά σε αυτή την ημερομηνία (πχ. 2 μέρες πριν). Ο δανεισμός του φοιτητή αυτού μπορεί να εμφανίζεται με κάποιο χρώμα όσο περνάνε οι μέρες πηγαίνοντας για παράδειγμα από πράσινο σε κίτρινο πορτοκαλί και τέλος κόκκινο ανάλογα με το πόσο κοντά φτάνει στην ημερομηνία που είναι να επιστρέψει το βιβλίο. Μια άλλη εναλλακτική θα ήταν η αποστολή email σε φοιτητές αντί για υπενθύμιση στην εφαρμογή.
2. Μια σημαντική βελτίωση που θα μπορούσε να έχει το πρόγραμμα θα ήταν η αποθήκευση των στοιχείων που εισάγουν οι χρήστες σε βάσεις δεδομένων ώστε να διατηρούνται και μετά το κλείσιμο της εφαρμογής .
3. Ακόμη θα μπορούσαμε να προσθέσουμε και αλλά φίλτρα αναζήτησης σε όλα τα πεδία της εφαρμογής όπως εμφάνιση των μαθητών ταξινομημένα ανάλογα με τα αρχικά των ονομάτων τους ή ανάλογα με τον κωδικό που έχει ο κάθε φοιτητής.
4. Δημιουργία ενός section στην εφαρμογή που θα προορίζεται για χρήση από τους φοιτητές , όπως για παράδειγμα ένα tableView που θα περιέχει όλα τα βιβλία μέσω του οποίου θα μπορούν οι φοιτητές να κάνουν μόνοι τους την διαδικασία του δανεισμού.
5. Τέλος έξυπνη θα ήταν η προσθήκη ενός section για χρήσιμες πληροφορίες στην εφαρμογή όπως για παράδειγμα τα ποσά προστίμου και τις ημέρες που θα μπορεί ο χρήστης να έχει το βιβλίο στην διάθεση του .