

# CCT College Dublin

## Assessment Cover Page

<b>Module Title:</b>	Algorithms and Constructs
<b>Assessment Title:</b>	CCT Library System
<b>Lecturer Name:</b>	Aldana Louzan
<b>Student Full Name:</b>	FELIPE DA SILVA CUNHA/ RAPHAELLA DEBORAH R B GOMES
<b>Student Number:</b>	2022250 / 2022247
<b>Assessment Due Date:</b>	23/04/2023
<b>Date of Submission:</b>	23/04/2023

### Declaration

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

## SUMMARY

Introduction .....	3
GitHub repository and video.....	4
Search for a specific book by title and/or author name/ student name/ ID student.....	4
List all books by title/ by author's name/ list all students by student's name/ by ID student....	4
Register that a student has borrowed a book.....	6
Waiting queue for a book.....	6
Return a book .....	6
Trying to borrow a book for a student that is not the first on the queue and for the first of the queue .....	7
For a specific student, list the books that they have borrowed.....	7
Extra functionalities .....	7
Class diagram.....	8
Challenges .....	8
References .....	9

---

---

---

## INTRODUCTION

---

The CCT College requests for develop a system Library. This CA will cover appropriate constructs and data structures. The system have track stock of book, register the borrow of book, control the availability of book, return of the book to library, queue of waiting when the book is borrowed by another students ,list of queue waiting by book ,searching of student by Name/ID , searching of book by Author's name/Title of book, allow the user see the list of book are/was borrowed, list all the book that specific students has borrowed, store the register of book, students, controller of availability of book, If a book is returned and has a waiting queue, display to the user the next student waiting for that book.

---

## GITHUB REPOSITORY AND VIDEO

---

We had to create two new repositories on GitHub because we had versioning problems with the project. The updated version is LibraryVersion3

<https://github.com/FelipeCunha03/Library>  
<https://github.com/FelipeCunha03/LibraryNewVersion>  
<https://github.com/FelipeCunha03/LibraryVersion3>

Video explaining the main functionalities of the system:

<https://www.youtube.com/watch?v=1YK2FkXOWKM>

---

## SEARCH FOR A SPECIFIC BOOK BY TITLE AND/OR AUTHOR NAME/ STUDENT NAME/ ID STUDENT

---

I(Felipe) have done implemented the searching algorithm. I had choose the linear searching because the list(Student and Book) are not big son I will be farther anyway, and as the list of students and books are not sorted when I am doing my linear searching where is not possible to use complexly algorithm for search farther where the list have to have to be organize in ascending or descending order. Another reason is because I have to find just one element of my list that way the linear searching can be simples and efficient.

```
*****
*****
====> Enter with the option:
1
Inform the book's title:
vamps

Id Book: 273ddc90-fff0-4e29-a75c-39cb09cfff8e
Author's name: whitby mcatamney
Book's title: vamps
Genre: Comedy|Horror|Romance
```

```
*****
```

```
====> Enter with the option:
6
Inform the student's ID:
1
```

```
-----Student's details -----
Id Student: 1
Student's name: amanda lor
Address: 24 Old County Road
City: Dublin
Phone Number: 35383089634
Gender: F
Date of Birth: 3/20/1985
```

```
====> Enter with the option:
2
Inform the name book's author :
Whitby McAtamney

Id Book: 273ddc90-fff0-4e29-a75c-39cb09cfff8e
Author's name: whitby mcatamney
Book's title: vamps
Genre: Comedy|Horror|Romance
```

```
====> Enter with the option:
5
Inform the student's name:
Amanda Lor
```

```
-----Student's details -----
Id Student: 1
Student's name: amanda lor
Address: 24 Old County Road
City: Dublin
Phone Number: 35383089634
Gender: F
Date of Birth: 3/20/1985
```

---

## LIST ALL BOOKS BY TITLE/ BY AUTHOR'S NAME/ LIST ALL STUDENTS BY STUDENT'S NAME/ BY ID STUDENT

---

The method I (Raphaella) chose to carry out sorting in the program was Bubble Sorting. It was used in the student's name list, student ID card list, book author name list, and book title list.

Among the existing sorting methods, I initially chose Bubble sorting because it was the easiest code to understand, and the implementation was initially simple. Furthermore, as the list of students does not have as much data, the method works perfectly. I found it a bit difficult just in the classes to list by student's name and list by the name of the author, as these two attributes are divided into two variables each (first name and last name). Sorting was initially performed by the first name only. Example: if there were two students with the name Jhon, but different last names, when the students' names were listed and sorted, Jhon would be displayed in the correct position in the list, but between Jhon Miller and Jhon Dunnes, the sort order. The solution was simple, just two more conditions were included in the if to perform the last name comparison as well.

```

===> Enter with the option:
3

*****LIST BOOKS BY BOOK'S TITLE ORDER*****

[
Id Book: 72feae8d-e321-4132-a050-a69bfda6467c
Author's name: sonnnie thebes
Book's title: "adventures of mark twain"
Genre: The"

,
Id Book: d58548e6-b307-4c72-ad64-b8860f966207
Author's name: harrison bilton
Book's title: "arise"
Genre: My Love"

,
Id Book: f293b6c6-ce7d-425a-ac60-f001574383b8
Author's name: doe congrave
Book's title: "auntie from chicago"
Genre: The (I theia apo to Chicago)"

===> Enter with the option:
7

*****LIST STUDENTS BY NAME ORDER*****

[
-----Student's details -----
Id Student: 9
Student's name: amanda barbosa
Address: 24 Old County Road
City: Dublin
Phone Number: 35383089634
Gender: F
Date of Birth: 3/20/1985,
-----Student's details -----
Id Student: 30
Student's name: ana almeida
Address: 77 Parnnel Street
City: DOn Laoghaire
Phone Number: 35383089644
Gender: M
Date of Birth: 3/21/1985,
-----Student's details -----
Id Student: 7
Student's name: caio jesus
Address: 24 Old County Road
City: Dublin
Phone Number: 35383089634
Gender: F
Date of Birth: 3/20/1985,

===> Enter with the option:
4

*****LIST BOOKS BY AUTHOR ORDER*****

[
Id Book: 55110b38-28ba-4af2-abfe-20412e754d5b
Author's name: abagail salle
Book's title: secret reunion (ui-hyeong-je)
Genre: Drama|Thriller

,
Id Book: 107542e7-be09-494e-81ed-c91c70839658
Author's name: abba geach
Book's title: kiss the bride
Genre: Comedy|Romance

,
Id Book: 4e49b188-b103-4de7-9961-36alad4ce38d
Author's name: abdel mcgowran
Book's title: paranorman
Genre: Adventure|Animation|Comedy

===> Enter with the option:
8

*****LIST STUDENTS BY ID ORDER*****

[
-----Student's details -----
Id Student: 1
Student's name: jhon souza
Address: 77 O Connell Street
City: DOn Laoghaire
Phone Number: 35383089644
Gender: M
Date of Birth: 3/21/1985,
-----Student's details -----
Id Student: 2
Student's name: joana santos
Address: 77 The Street
City: Dun Laoghaire
Phone Number: 35383089644
Gender: M
Date of Birth: 3/21/1985,
-----Student's details -----
Id Student: 3
Student's name: eloisa serra
Address: 77 Parnnel Street
City: DOn Laoghaire
Phone Number: 35383089644
Gender: M
Date of Birth: 3/21/1985,

```

---

## REGISTER THAT A STUDENT HAS BORROWED A BOOK.

---

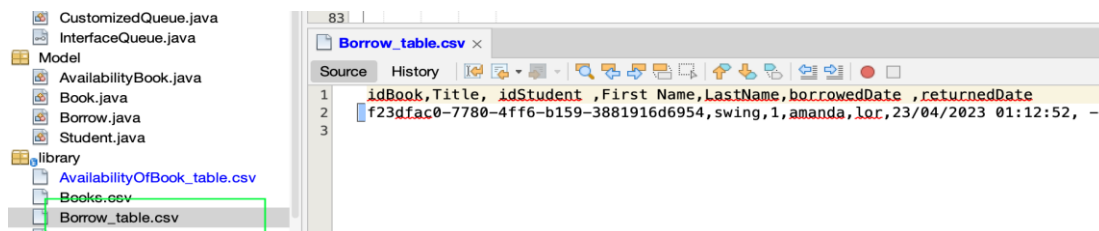
It was created the class borrow where I had put some attributes from Student and Book where I found necessary. After done it I got all those attributes and I created a OBJ and add this OBJ borrow on the list of borrow.

```
====> Enter with the option:
9
Inform the book's title:
Swing
Informe the student's ID:
1

***Confirmed the borrowing of the book to the student***

-----Borrow's Details-----
Id Book: f23dfac0-7780-4ff6-b159-3881916d6954
Book's title: swing
Id Student: 1
Student's Name: amanda lor
Data Borrowing: 23/04/2023 01:12:52
Data Returned: ----
```

After registered the borrow of book the list of borrow store on file CVS.



---

## WAITING QUEUE FOR A BOOK

---

To the queue of students waiting for a book, we created a map with the obj Book as a key, and a queue as a value. We created our own queue with all the methods that we need, as for example, add and remove student on the queue, get the first student among others. To add the student on the queue, the book must be borrowed by someone else, and when I try borrow the same book for another student, if the user confirms, the system will insert my book in my map (if it doesn't exist already in the map) and will add the student on the queue for that book.

```
====> Enter with the option:
9
Inform the book's title:
vamps
Informe the student's ID:
30
Book is borrowed. Would you like to wait on the queue for the book? Y/N
Y

***Confirmed! The student ana almeida is on the queue for the book vamps***
```

---

## RETURN A BOOK

---

After I search for the book that I want to return, I will go through all my list of available book and check if my book is borrowed. If it is, I will set the book to be available again, I will write on my file that the book is available to be borrowed and I will set the returned date of that borrow. After that, I will check in my map if there is someone waiting on the queue for that book. If it is, I will call the method to get the id of the first student on the queue that we developed in our own queue, I compare this id with the id of my list student to get the name

of the student and then I will show to the user the name of the first student waiting on the queue for that book.

```
===> Enter with the option:
10
Inform the book's title:
vamps

*Book returned!*
Returned date: 23/04/2023 02:51:30

The first student waiting on the queue for this book is: ana almeida - id student: 30.
```

---

### TRYING TO BORROW A BOOK FOR A STUDENT THAT IS NOT THE FIRST ON THE QUEUE AND FOR THE FIRST OF THE QUEUE

---

When I start the borrow, the method will check if the student who wants to borrow the book is the first student on the queue waiting for that book. If it is positive, I call the method in our own queue to remove this student of the queue and I let this student borrow the book. If it is not, the system doesn't allow any student to borrow the book and show to the user the name of the first student waiting on the queue.

```
===> Enter with the option:
9
Inform the book's title:
vamps
Informe the student's ID:
7
The book is not available to you yet because the first student of the queue is: ana almeida
```

---

### FOR A SPECIFIC STUDENT, LIST THE BOOKS THAT THEY HAVE BORROWED

---

For this list it was used the method that already created for to do the linear searching just to get the Student. After got the student It was used a FOR to find the student in list of borrow, if the student is in list of borrow get the borrow just output this student. This will happen still for read all the elements of list borrow. That way if the student had borrowed more than one book the method will output all.

---

### EXTRA FUNCTIONALITIES

---

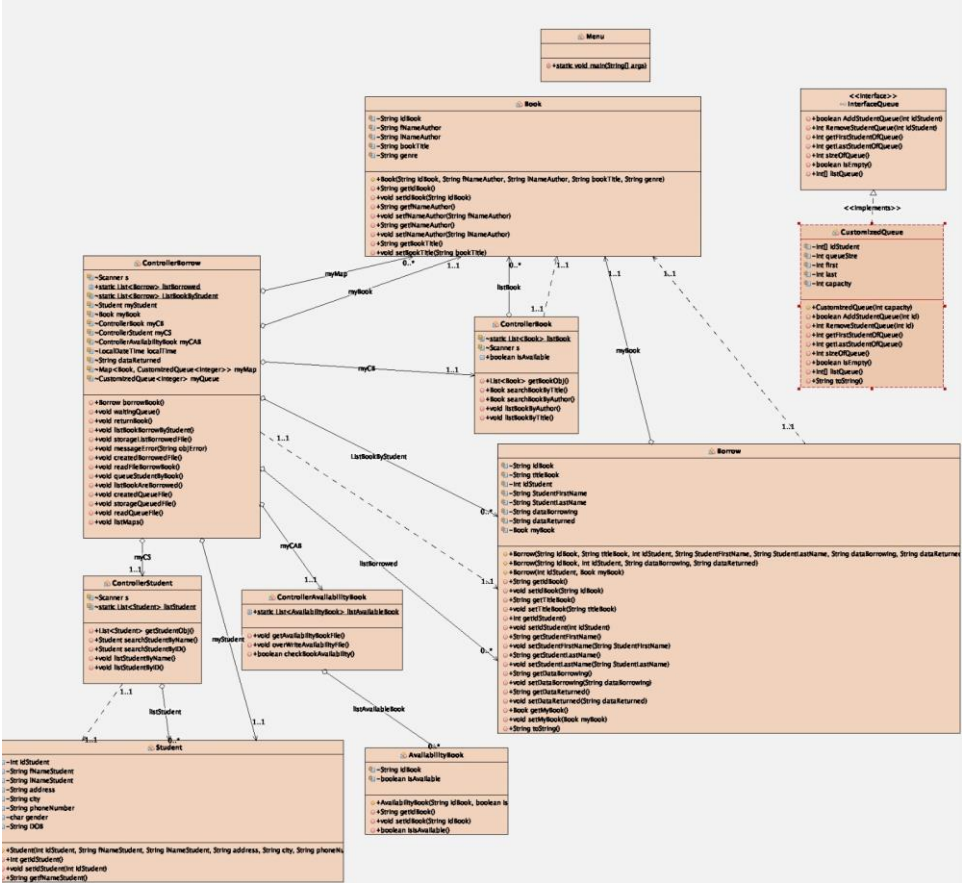
**Check book availability:** It was created a class 'book availability' with attributes idBook and available . It was read a file CSV with all idBook and available, this file we created by ourselves. After read the file we just created list of obj type of 'book availability' and add this obj a list of available .It is useful for check if the book is available for borrow. We used this list of available for do not allow the student borrow the book if the book is already borrowed.

**List all the books that was borrowed:** Just output the list of borrow.

**List all the books that are borrowed:** Output all the books that are borrowed at moment. It was done a FOR to compare if the book that it is in list of book is available in list of available. If it is true just pass this or those books to Array<Book> and output this array after.

**List the queue by book:** Output all the students(Queue) are in the waiting list for borrow specific book. It was used a FOR for to read the map and get the Queue by specific book thar student type on the screen.

## CLASS DIAGRAM



## CHALLENGES

As there were many implicit requirements that generated many doubts between the participants, we had many challenges working together, as each one thought in a different way about what and how the system should be developed, and we did not have much time to discuss the doubts with the teacher to move forward with the project, due to the short deadline established. We had a little difficulty with Java, as we have basic knowledge of the language and, while we were developing the system, we were learning together. We had difficulty listing all the students who are in the queue waiting for a specific book. As we had to access the queue, we created to display this list, and this queue is inside a map, this generated a bit of confusion when retrieving this information. But the solution was simple, we had just to create a new method inside our queue, sending only the idStudent attribute that was created as an array in the queue class. While we were developing the program, we realized how big it was and the time we had for delivery was not enough for us to be able to finish it, and it generated a lot of stress. We identified that the program should persist the data of borrowed books and the queue of students with a few days left for the delivery of the assignment. We finished the borrow's recording and reading, but we were only able to finish the recording of the queue.



---

## REFERENCES

---

DevMedia. (n.d.). Os Melhores Artigos sobre Java, Javascript, .NET, PHP, Python... [online] Available at: <https://www.devmedia.com.br/artigos/>.

Stack Overflow (2022). Stack Overflow - Where Developers Learn, Share, & Build Careers. [online] Stack Overflow. Available at: <https://stackoverflow.com/>.