20 Registered User Rents Book

## Description

As a user

I want to allow a registered user to rent a book

So that I can see in the database that that user has a copy of that book

## API

|  |  |
| --- | --- |
| Request | POST |
| URL | http://localhost:8080/api/library/rent?userId={userId}&title={title} |
| Path Variable |  |
| Request Body |  |
| Output: happy case | A User object (JSON) – fields: id, name, age, email, password, profile with fields {id, bio, place}, list of memberships with fields {id, startDate, endDate, type}, list of rented books with fields {id, title, numberInStock, price, inColor, availableCopies} |
| Output: unhappy case | Response 400 BAD REQUEST  One of the following JSON error messages:   * “userId”: “User with id {id} does not exist” * “userId”: “User with id {id} has no valid membership today” * “title”: “This book is not in the library” * “title”: “There are no items of the book {title} available” |
| Acceptance Criteria | * The book field “number in stock” indicates how many copies the library has (bought) of the given book. Each copy can be rented by another user, so a book can be rented by several users. * A user can rent only one copy of a book with given title. * A user can only rent a book when he has a valid membership (i.e. there is a membership with the date of today between start and end date). * When a user rents a book with given title, it is added to the list of books that the user has rented. * The number of available copies of the book is decreased by 1. The number in stock is *not* modified. The number of copies cannot be less than 0. * You cannot delete a book when a copy of this book is rented. * When you update a book, the number in stock cannot be less than the number of copies that are already rented. * When a user is deleted, all rented books are returned |

## Technical Requirements

* Add a field “availableCopies” to the class book to remember the number of copies not yet rented. When a book is created, “availableCopies” equals to “numberInStock”.