Story 02 - Book Statistics - v2

Changes with respect to v1: see Technical Requirements

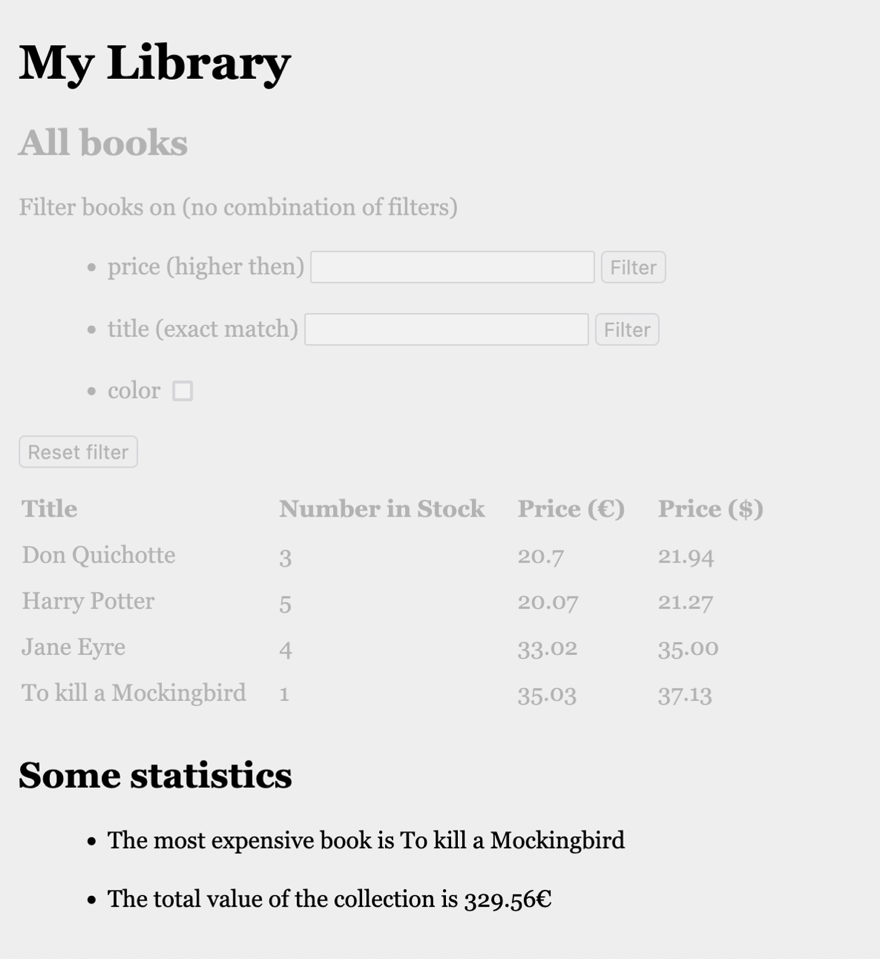
## Description

As a user

I want to see the the most expensive book in the library

And the total value of the collection.

## Wireframe



## Acceptance Criteria

* Below the table overview following information is shown as list
  + The title of the most expensive book
  + The total value of the collection (in €)

## Technical Requirements

* The total value is calculated as (price\*numberInStock)
* API for “View total value”

|  |  |
| --- | --- |
| **View total value of the library** | |
| **Request** | GET |
| **URL** | <http://localhost:8080/api/book/totalValue> |
| **Input** |  |
| **Output: Happy Case** | Number (double) |
| **Output: Unhappy Case** | Returns 0 when there are no books in the library |
| **Acceptance Criteria** | Returns the total value of the books in the library. |

* API for “View most expensive book”

|  |  |
| --- | --- |
| **View most expensive book of the library** | |
| **Request** | GET |
| **URL** | <http://localhost:8080/api/book/mostExpensive> |
| **Input** |  |
| **Output: Happy Case** | Book object (as JSON) with fields “title”, “numberInStock”, “price”, “inColor”, “priceInDollar” |
| **Output: Unhappy Case** | Nothing (null) when there are no books in the library. |
| **Acceptance Criteria** | Returns the most expensive book in the library. |

* The books are stored in a h2 database.
* Search queries are performed in a jpa repository (where possible).