Book Management Application

Student: Casiana Stefana Campean

**Group: 30238**

Table of Contents

1. Requirements Analysis 3

1.1 Assignment Specification 3

1.2 Functional Requirements 3

1.3 Non-functional Requirements 3

2. Use-Case Model 3

3. System Architectural Design 3

4. UML Sequence Diagrams 3

5. Class Design 3

6. Data Model 3

7. System Testing 3

8. Bibliography 3

1. Requirements Analysis

# Assignment Specification

For this assignment we need to build a book management service.

A user should be able to create an account, choose a payment plan and log in to search the book library.

Payments can be done via a cash only policy and need to be validated by library staff.

The library is managed by staff and can be filtered by release date, author, title, genre.

If a book is available, a user can add it to it’s library. If not, the user can join a waiting list. Once a book has been read by a user, it can be returned via the online library return function. This assign the book to the next user in the waiting list after validation by the library staff.

The service also provides with dynamic recommendations based on the latest trends (popular borrowed books) or user defined interests by genre or topic.

# Functional Requirements

The functional requirements are:

* to implement and test the application
* to use layered architecture
* use a factory method for building user recommendations by trends, genres or topics
* Store data in a database
* Validate inputs before submitting he data and saving it in the database

# Non-functional Requirements

The functional requirements are:

* To commit the work to my Git repository and to do it iteratively as I progress
* To use OOP language
* Use layered architecture

2. Use-Case Model

*[Create the use-case diagrams and provide one use-case description (according to the format below).*

*Use-Case description format:*

*Use case: <use case goal>*

*Level: <one of: summary level, user-goal level, sub-function>*

*Primary actor: <a role name for the actor who initiates the use case>*

*Main success scenario: <the steps of the main success scenario from trigger to goal delivery>*

*Extensions: <alternate scenarios of success or failure>*

*]*

3. System Architectural Design

**3.1 Architectural Pattern Description**

*[Describe briefly the used architectural patterns.]*

**3.2 Diagrams**

*[Create the system’s conceptual architecture; use architectural patterns and describe how they are applied. Create package, component and deployment diagrams]*

4. UML Sequence Diagrams

*[Create a sequence diagram for a relevant scenario.]*

5. Class Design

**5.1 Design Patterns Description**

*[Describe briefly the used design patterns.]*

**5.2 UML Class Diagram**

*[Create the UML Class Diagram and highlight and motivate how the design patterns are used.]*

6. Data Model

*[Present the data models used in the system’s implementation.]*

7. System Testing

*[Present the used testing strategies (unit testing, integration testing, validation testing) and testing methods (data-flow, partitioning, boundary analysis, etc.).]*

8. Bibliography

[1] Hibernate Tutorial: <https://www.youtube.com/playlist?list=PLEAQNNR8IlB7fNkRsUgzrR346i-UqE5CG>