System Requirements Specification Index

For

Library Management Console InMemory

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

TABLE OF CONTENTS

1 Project Abstract

Library Management Console Application is a pure java application with Java collection, where it allows to manage the books inside the library. The Library Management System empowers users to perform CRUD (Create, Read, Update, Delete) operations and search functionalities in different criterias on books. Users can create new book entries, update existing book and other information, delete books and many more relevant operations.

Source Code for the project: https://github.com/NavinIIHT/Library-Management-Core-Java.git

2 Business Requirements:

Screen Name	Console input screen
Problem Statement	1. User needs to enter into the application. 2. The user should be able to do the particular operations 3. The console should display the menu 1) Add Book 2) Search Book by Name 3) Search Books by Author 4) Search Books by Publisher 5) Check Availability 6) Update Book 7) Delete Book 8) Exit

3 Common Constraints

1. Take console input of number of books: (n)
2. Take input of details of each book and store it in a collection.
3. Take input of details of books to be added (only 1 book at a time)

4 Code Structure

4.1 Package: com.elibrary

Resources

Class/Interface	Description
LibraryManagementApp.java(class)	**This represents bootstrap class i.e class with Main method, that shall contain all console interaction with the user.**

4.2 PACKAGE: COM.IIHT.TRAINING.ELIBRARY.MODEL

Resources

Class/Interface	Description
Book (class)	● This class contains all the properties of the Book class.

4.3 PACKAGE: COM.IIHT.TRAINING.ELIBRARY.INVENTORY

Resources

Class/Interface	Description
Inventory (class)	● This class contains all the methods which are used to write the business logic for the application ● You can create any number of private methods in the class ●

4.4 PACKAGE: COM.IIHT.TRAINING.ELIBRARY.EXCEPTION

Resources

Class/Interface	Description
ISBNAlreadyExistsException (Class)	● Custom Exception to be thrown when trying to add a book for which ISBN is already exists
BookNotFoundException (Class)	● Custom Exception to be thrown when trying to update a book or delete a book whose isbn is not found.

5 TEST CASES

You are required to write unit test cases using Junit/Mockito for following scenarios:

1. Test for Adding the new
2. Test for throwing exception if ISBM already exists while adding new book.
3. Test for searching book by name.
4. Test for searching book by author.
5. Test for searching book by publisher.
6. Test for checking the availability of book.
7. Test for updating the book info.
8. Test for throwing the exception of book not found for updating.
9. Test for deletion of book.
10. Test for throwing exception if book not found while deletion.