

Library Management System

Vision Document 1.0

Práctica Profesional

Victor Grycuk

Contents

[Contents 1](#_Toc16282815)

[Revision 2](#_Toc16282816)

[Introduction 3](#_Toc16282817)

[Purpose 3](#_Toc16282818)

[Scope 3](#_Toc16282819)

[Definitions, Acronyms, and Abbreviations 3](#_Toc16282820)

[Stakeholders and Users description 4](#_Toc16282821)

[Stakeholders summary 4](#_Toc16282822)

[Key Users 4](#_Toc16282823)

[User Environment 4](#_Toc16282824)

[Stakeholders Profile Librarian 4](#_Toc16282825)

[Member 4](#_Toc16282826)

[Global Product Description 5](#_Toc16282827)

[Product Perspective 5](#_Toc16282828)

[Characteristics Summary 5](#_Toc16282829)

[Assumptions and Dependencies 5](#_Toc16282830)

[Detailed Product Description 6](#_Toc16282831)

[Context Diagram 6](#_Toc16282832)

[Deployment Diagram 6](#_Toc16282833)

[Domain Model 7](#_Toc16282834)

[Class Diagram 8](#_Toc16282835)

[Primary Use Cases 9](#_Toc16282836)

[Activity Diagrams 10](#_Toc16282837)

[Registering New Book 10](#_Toc16282838)

[Registering New Author 11](#_Toc16282839)

[Registering New Member 12](#_Toc16282840)

[Checking out Book 13](#_Toc16282841)

[Returning a Book 14](#_Toc16282842)

[Alerting expiring due date 15](#_Toc16282843)

[Sequence Diagram 16](#_Toc16282844)

[Checking out Book 16](#_Toc16282845)

[Adding New Book 17](#_Toc16282846)

[Returning Book 18](#_Toc16282847)

[Alerting expiring due date 19](#_Toc16282848)

[Restrictions 20](#_Toc16282849)

[Functional Requirements 20](#_Toc16282850)

[Register new book manually 20](#_Toc16282851)

[Register new membership 20](#_Toc16282852)

[Update Book 20](#_Toc16282853)

[Searching for a book in the data base 21](#_Toc16282854)

[Checking out a book 21](#_Toc16282855)

[Other Product Requirements 21](#_Toc16282856)

[Standards 21](#_Toc16282857)

[System Requirements 21](#_Toc16282858)

[Use Cases Specifications 22](#_Toc16282859)

[Use Cases 22](#_Toc16282860)

[Log In / Log Out Management 26](#_Toc16282861)

[Multi Language Management 26](#_Toc16282862)

[Log Management 26](#_Toc16282863)

[Back Up Management 26](#_Toc16282864)

[Encryption Management 26](#_Toc16282865)

[Navigation Map 27](#_Toc16282866)

[User Interface Prototypes 28](#_Toc16282867)

[Log in window 28](#_Toc16282868)

[Main Menu 28](#_Toc16282869)

[Add new book 29](#_Toc16282870)

[Book Reservations 29](#_Toc16282871)

# Revision

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
| 09/08/2019 | 1.0 | Victor Grycuk | First version of Vision Document |

# Introduction

## Purpose

This document will gather, analyze, and define the high-level necessities and characteristics of the management system. This document also focuses in the required functionality by the participants in the project and the end users. The details regarding how the system covers the requirements can be found in the specification of the use cases and other additional documents.

## Scope

This Vision Document applies to the Library Management System (LMS), which will be developed by Victor Grycuk. Victor Grycuk will develop the LMS to work on the Windows based computers.

The purpose of this system is to allow librarians to manage their library using a digital library. It will allow the librarian to keep track of multiple editions of the same book, download a book information from internet when adding a new book, keep track of checked out books, and send reminders to members when the due date for returning a book is about to expire.

It will also allow them to manage the database of members, which will assign a Member ID when creating a new member.

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| Acronym | Definition |
| API | Application Programming Interface |
| DB | Data Base |
| DoB | Date of Birth |
| ID | Identification |
| ISBN | International Standard Book Number |
| LMS | Library Management System |
| SMTP | Simple Mail Transfer Protocol |
| TCP/IP | Transmission Control Protocol/Internet Protocol |

# Stakeholders and Users description

## Stakeholders summary

### Key Users

|  |  |  |
| --- | --- | --- |
| Name | Description | Responsibilities |
| Librarians | Main user | * Adding, updating, and removing book entries in the system. * Creates account for new members * Keep track of checked out books |
| Members | Passive user | * Checks out and returns books |

## User Environment

* The Library Management System (*LMS*) will be accessible in appointed computers in the library that meet the following minimal requirements:
  + Windows 7 or higher
  + At last 4gb of RAM
  + Microsoft .NET Framework 4.3 or higher
  + Internet access (optional)
* The *LMS* will be used by authorized librarians. For this, an account for each authorized librarian will be created, along with its user and password.
* In order to access the *LMS* each librarian will have to login using their respective personal credentials. After logging in, the main menu will be displayed.
* Once logged, the users will be able to change their preferred language.

## Stakeholders ProfileLibrarian

|  |  |
| --- | --- |
| Representant | None |
| Description | Person in charge of attending the general public and managing the library’s book stock. |
| Type | Key User |
| Responsibility | Person in charge of managing the catalogue of books, lending books to members, receiving returned books, and keeping tracked of checked-out books. |
| Exit Criteria | None |
| Participation Level | High |

## Member

|  |  |
| --- | --- |
| Representant | None |
| Description | End customer of the library |
| Type | Passive User |
| Responsibility | Almost none. No direct contact with the system, except for receiving emails from it, and checking out/returning books. |
| Exit Criteria | None |
| Participation Level | Low |

# Global Product Description

## Product Perspective

The *LMS* was developed to help librarians manage their library catalogue in a digital platform.

*LMS* can help them to quickly add new books and update existing ones in their digital catalogue. Additionally, they have the option to download the book information from an online source instead of typing them manually.

*LMS* will also have a digital database of Members of the library, that will help to keep track of lent books, alerting both the librarian and members if the return date of a book is about to expire, apply a fine for returning a book after the specified return date of the book, and notifying a member of new available books according to their preferences.

## Characteristics Summary

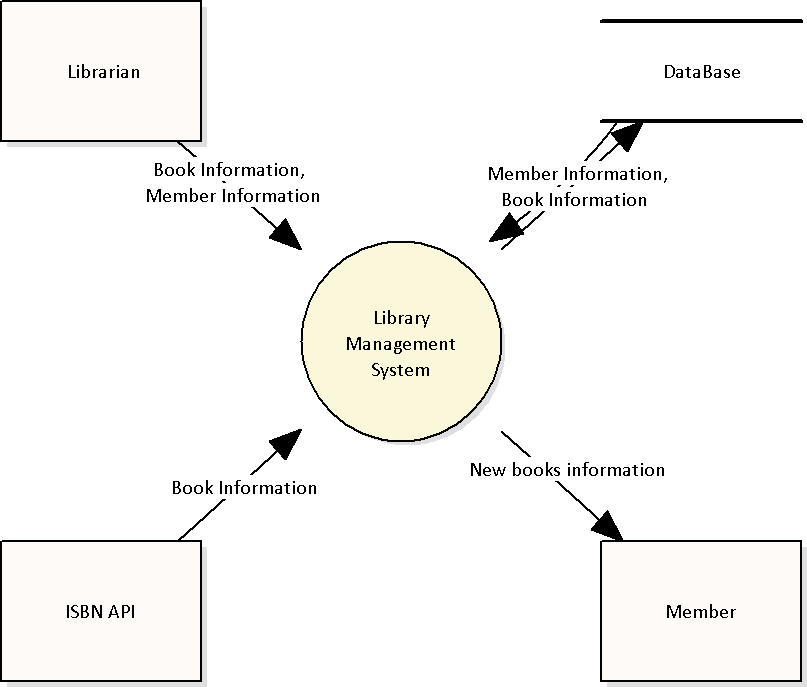
|  |  |
| --- | --- |
| Characteristic | Benefit for the Client |
| General | * Multi-language * Notifications of updates |
| Catalogue Management | * Add, remove, and update entries * Download book information from internet * Search of books in catalogue by their attributes (ISBN, Author, Release Date, etc.) |
| Member Management | * Create, update, and deactivate members account * Search for member by Name, Member ID, or any of their attribute * Notification of new books |
| Book check-out management | * Keep track of let books * Notifications of approaching returning date * Listing and search of current checked out books and the associated member * Application of fines for books returned after their specified date |
| Database | * Back up * Encryption |

## Assumptions and Dependencies

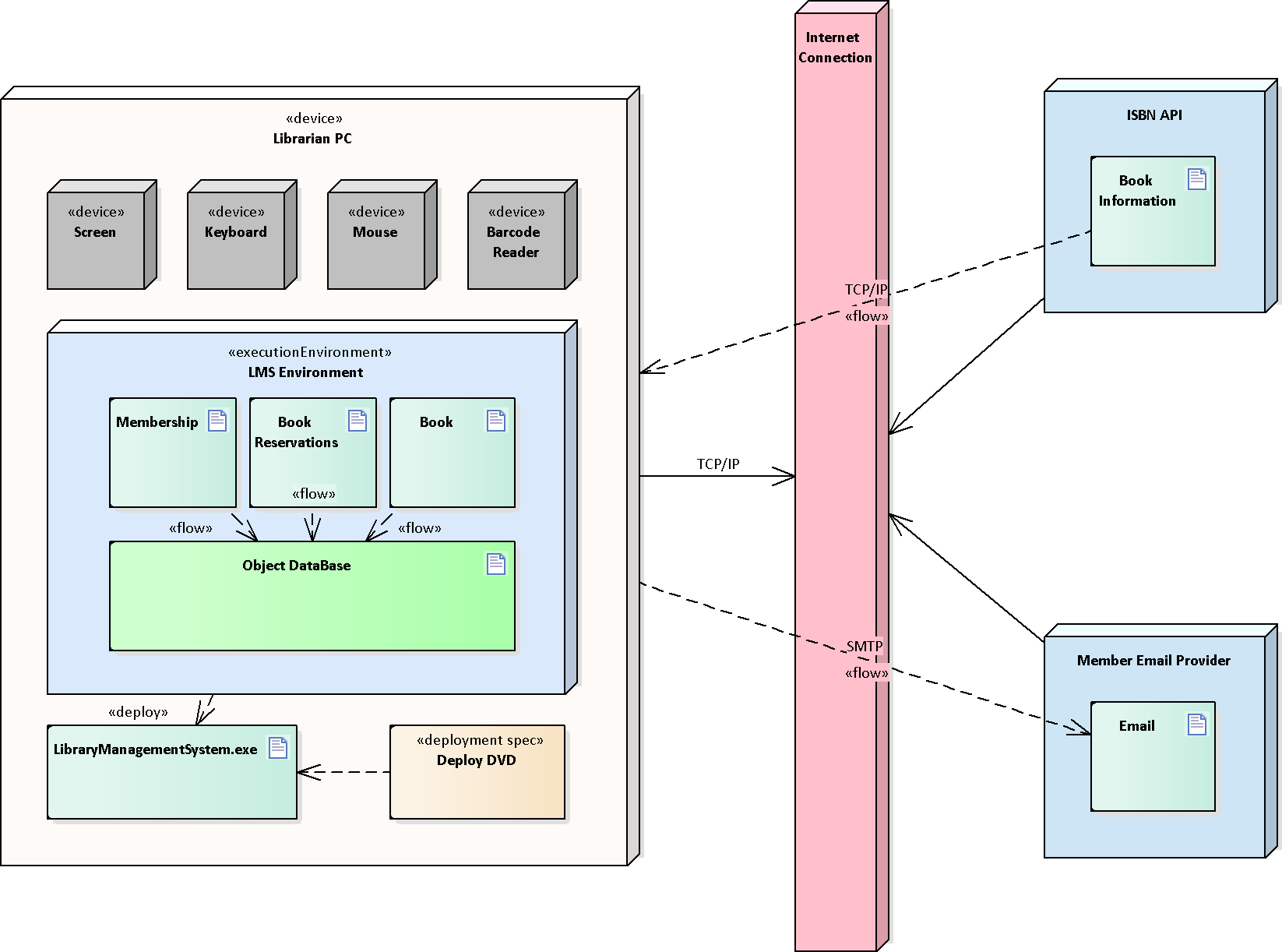
* It is assumed that the user has basic understanding of the Windows environment.
* The *LMS* default language is English. It is assumed the user knows enough English to be able to change the default language to the language of their preference.
* When trying to download a book information from an API, it is assumed that the computer will have access to the Internet.
* After the initial configuration, the user will oversee the creation of new accounts for librarians and users

# Detailed Product Description

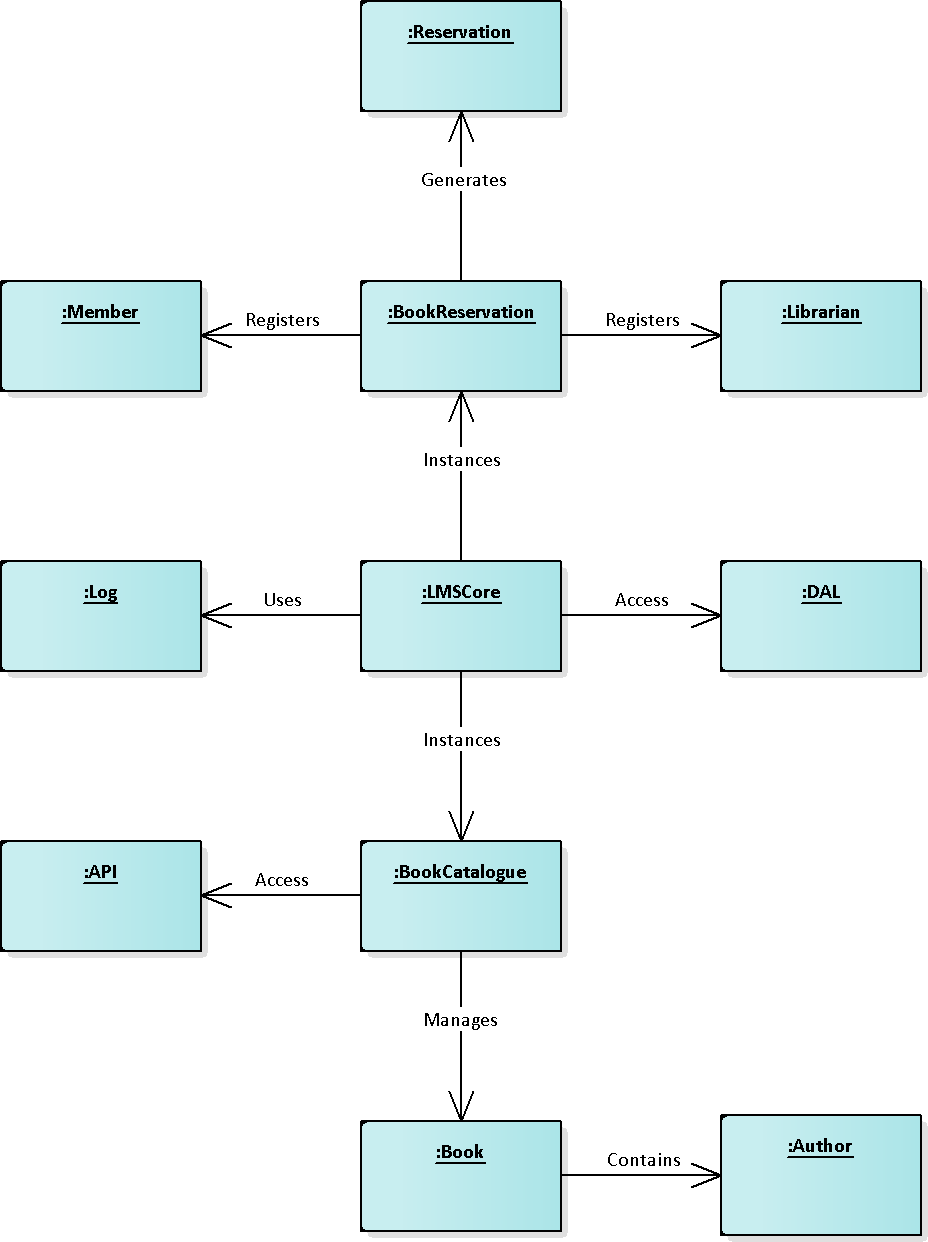
## Context Diagram



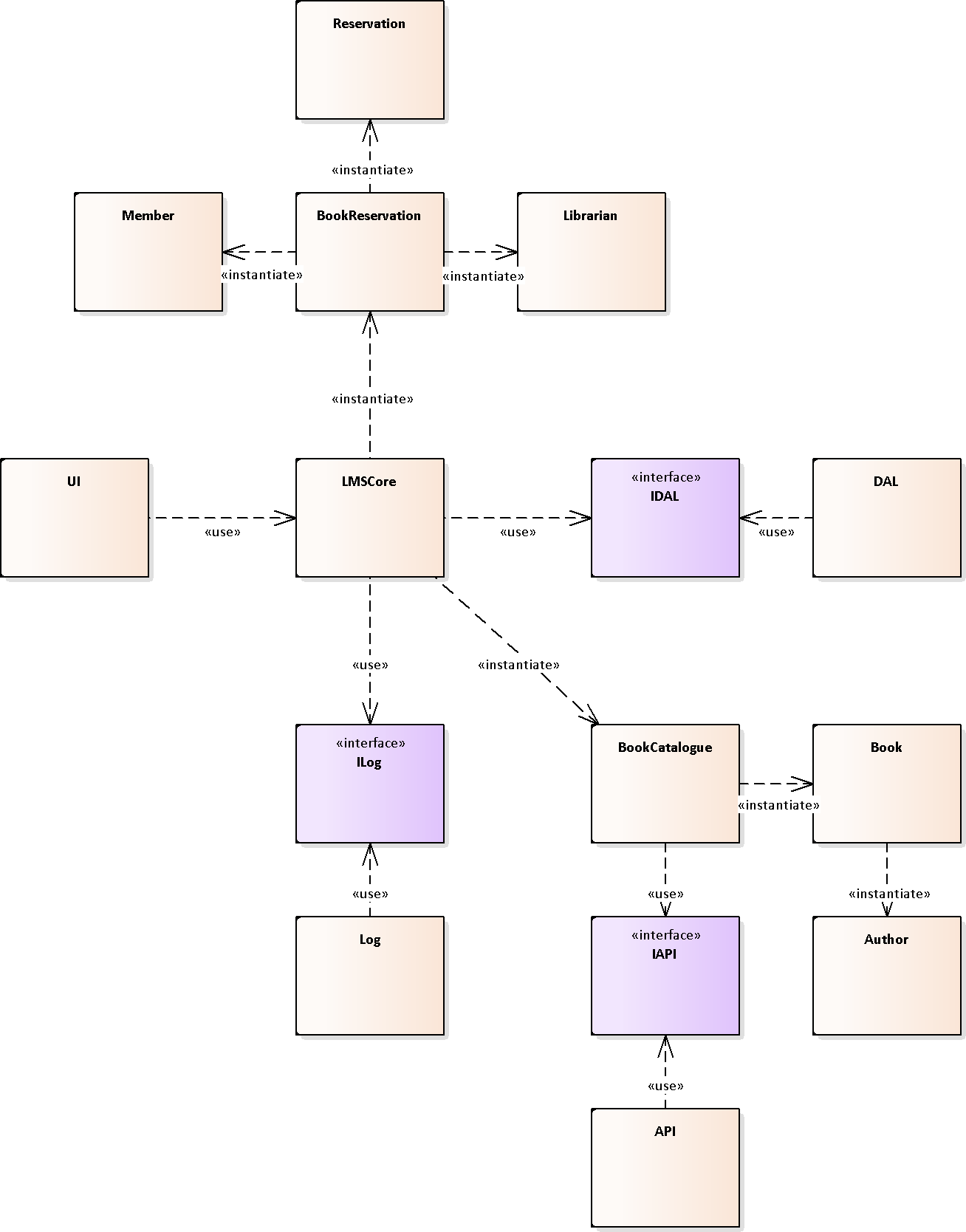
## Deployment Diagram



## Domain Model

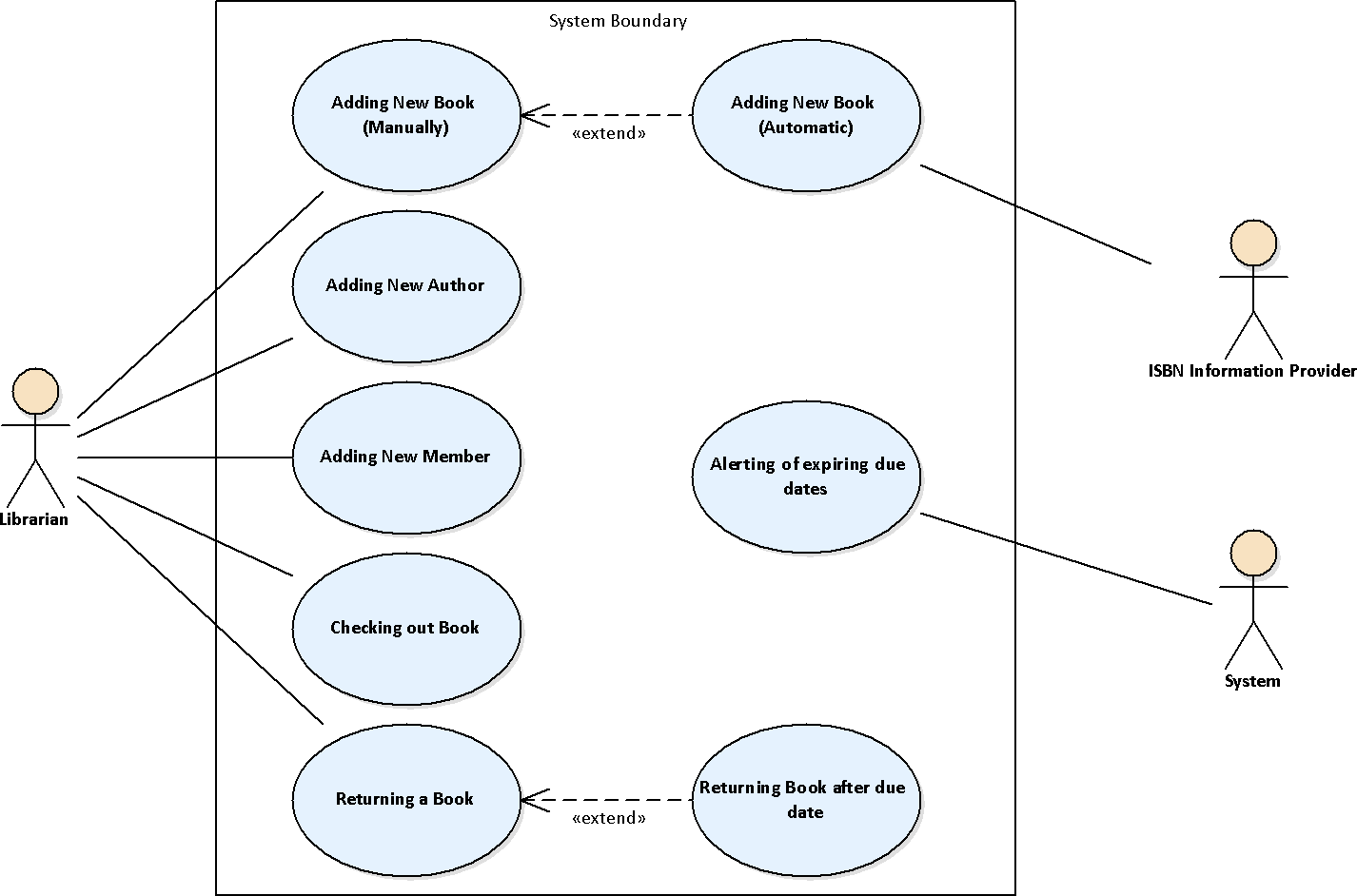


## Class Diagram



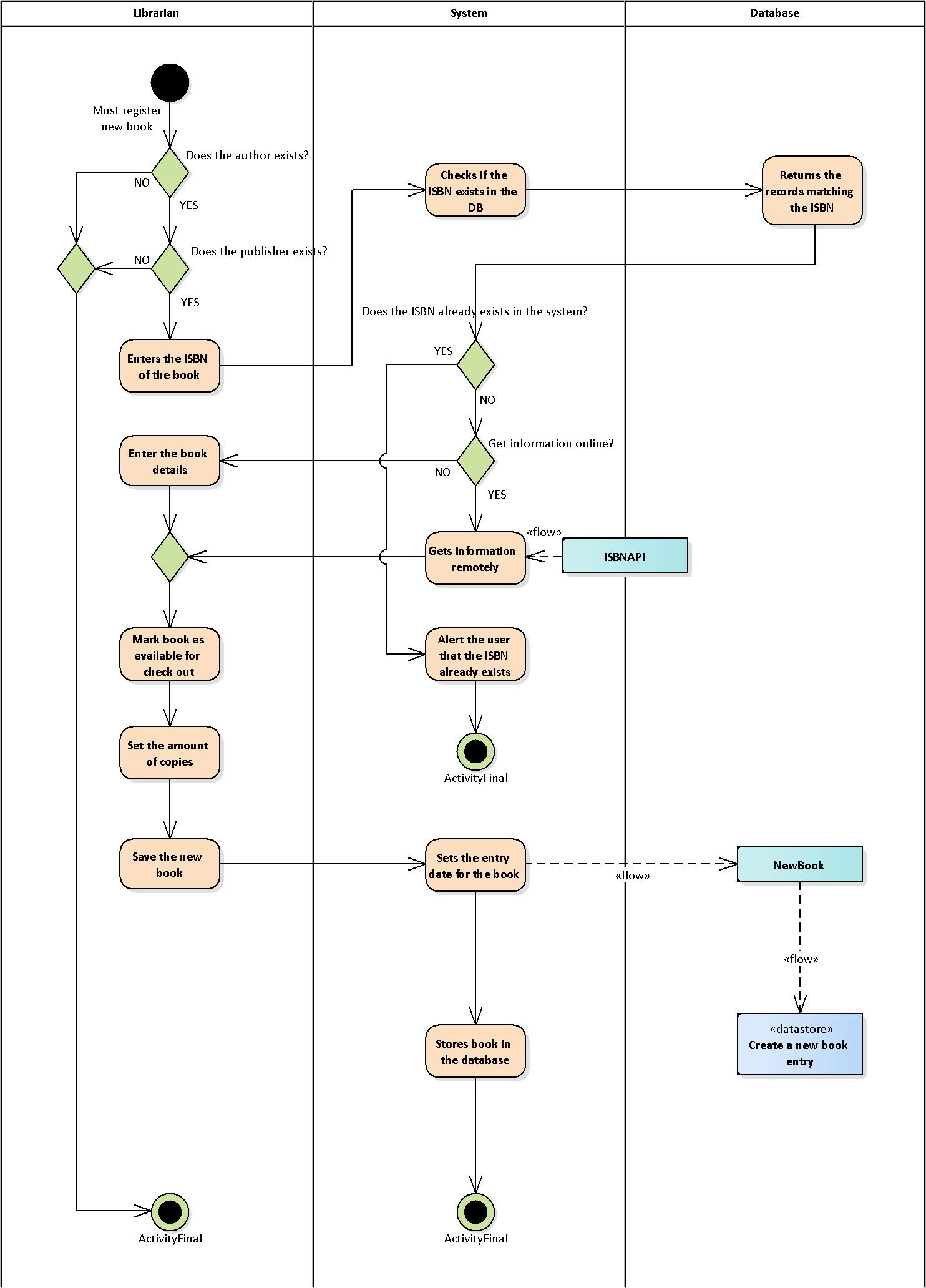
***Note:*** *This diagram offers a high-level view of the system, it might change during development.*

## Primary Use Cases

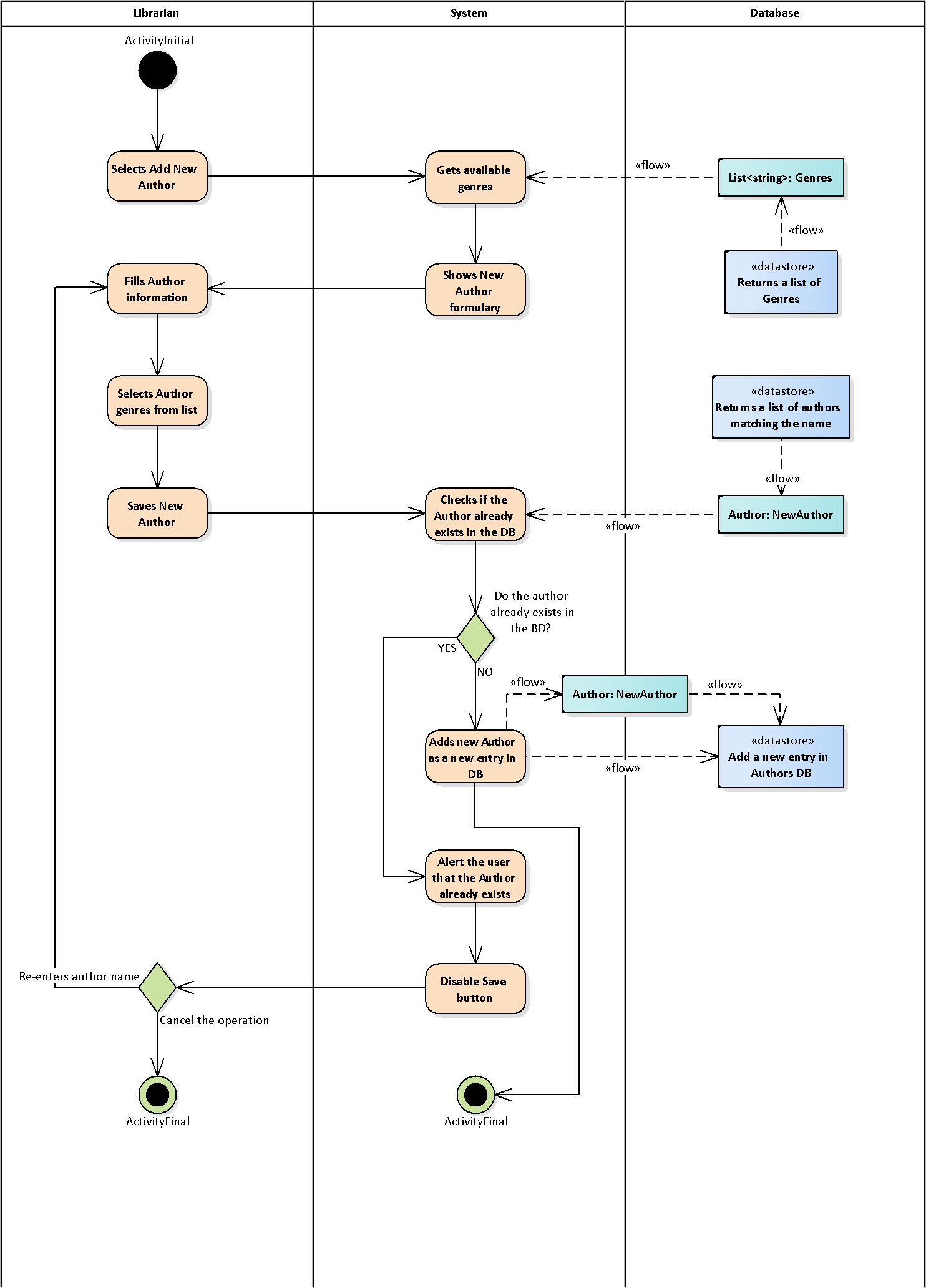


## Activity Diagrams

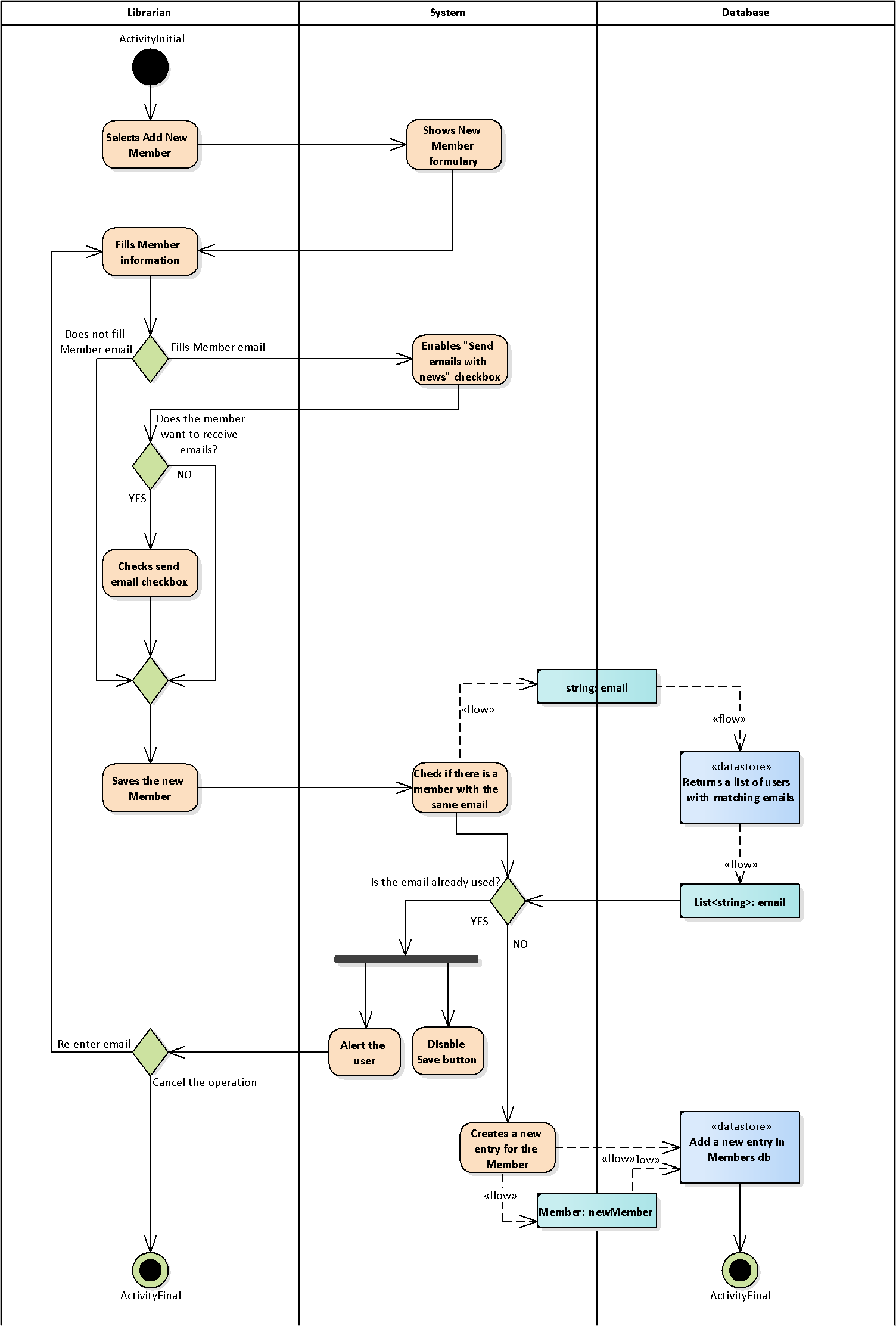
### Registering New Book



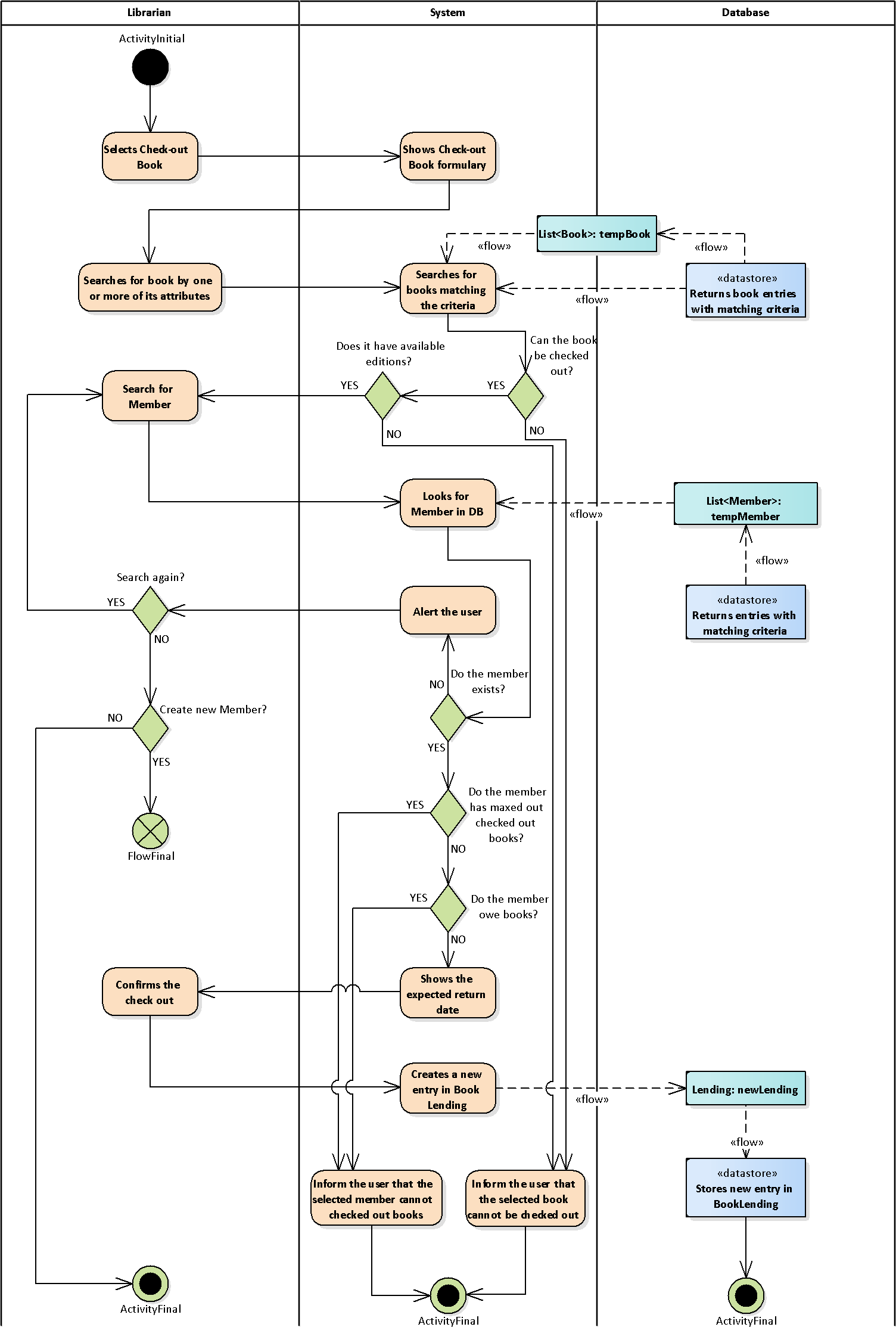
### Registering New Author



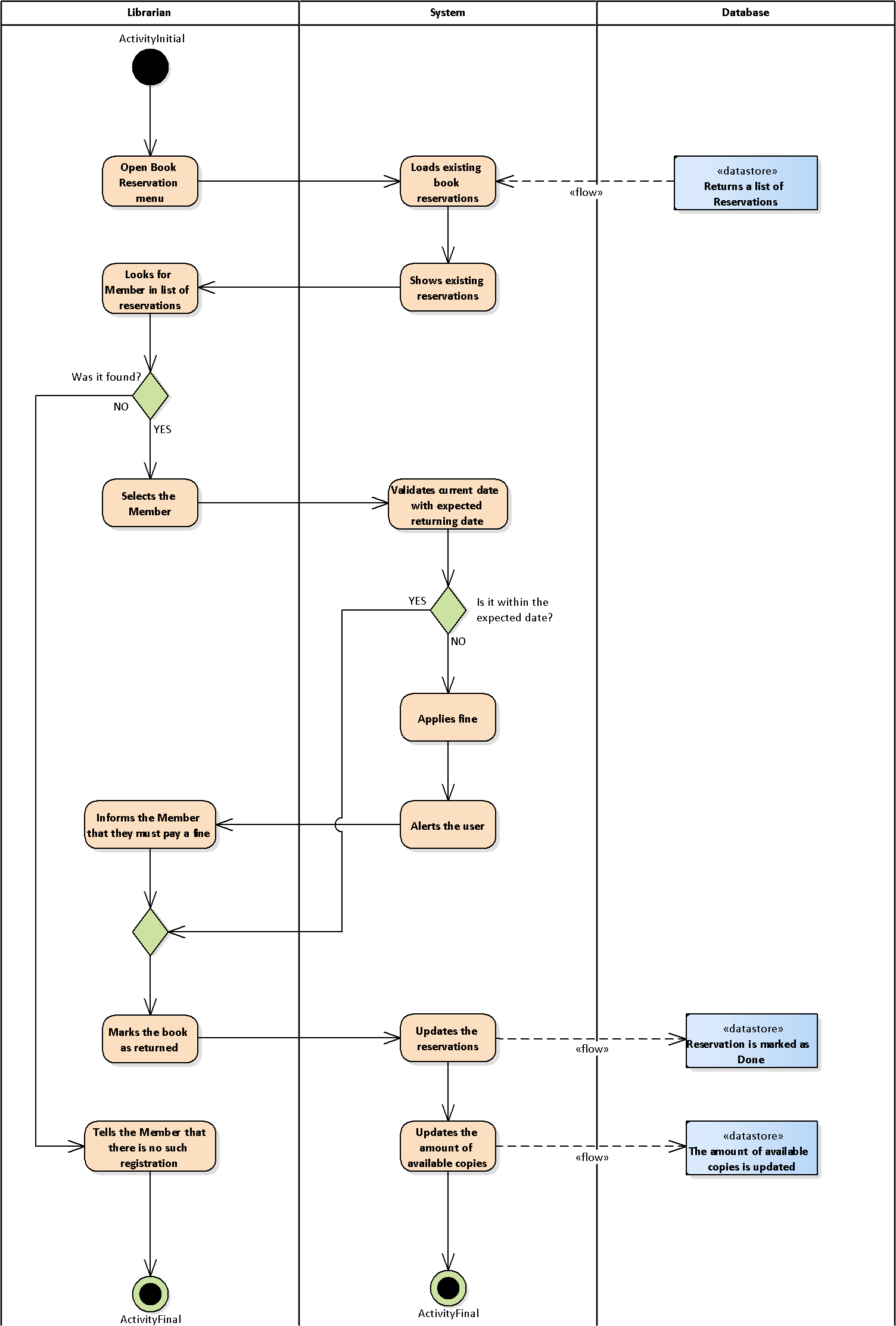
### Registering New Member



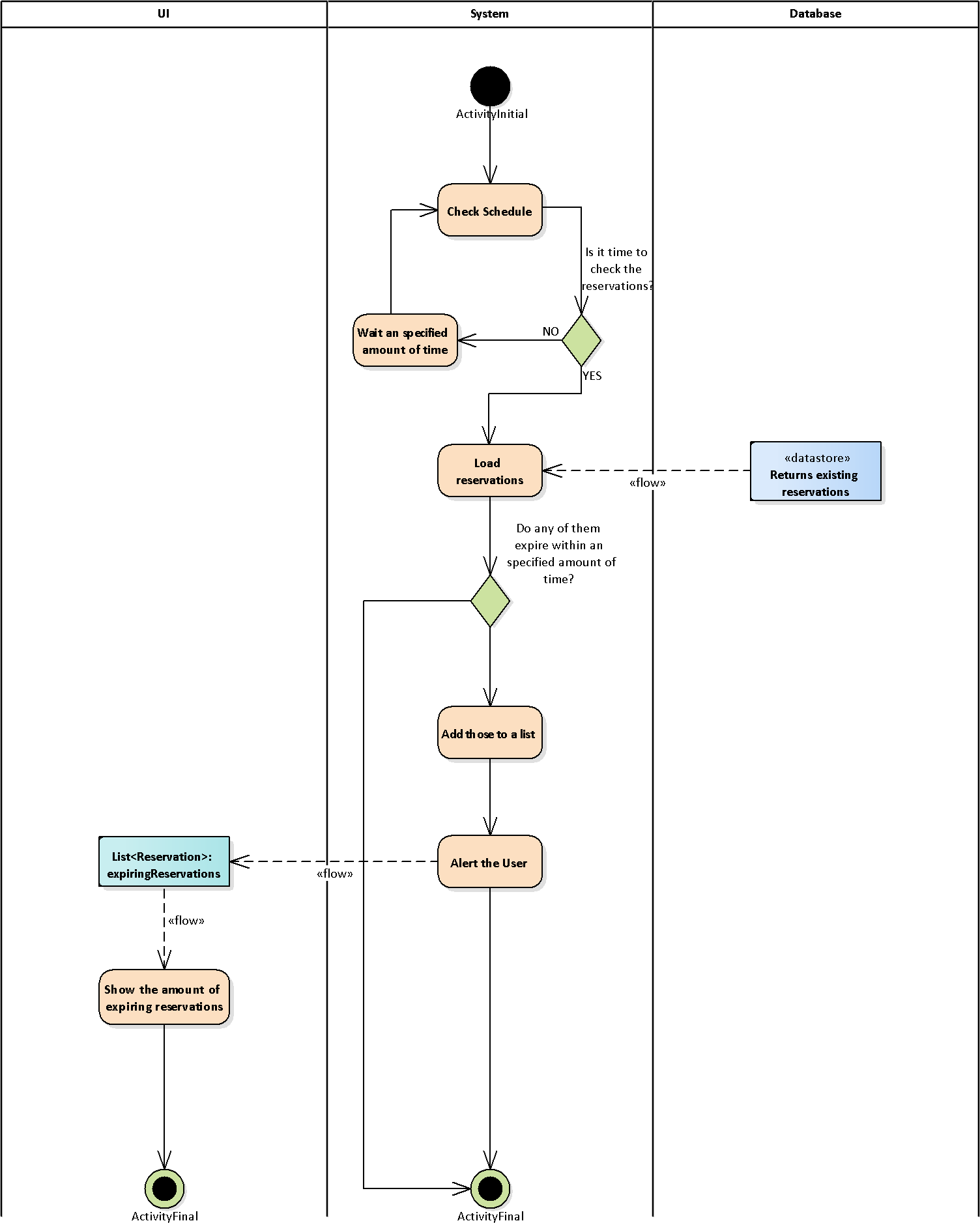
### Checking out Book



### Returning a Book

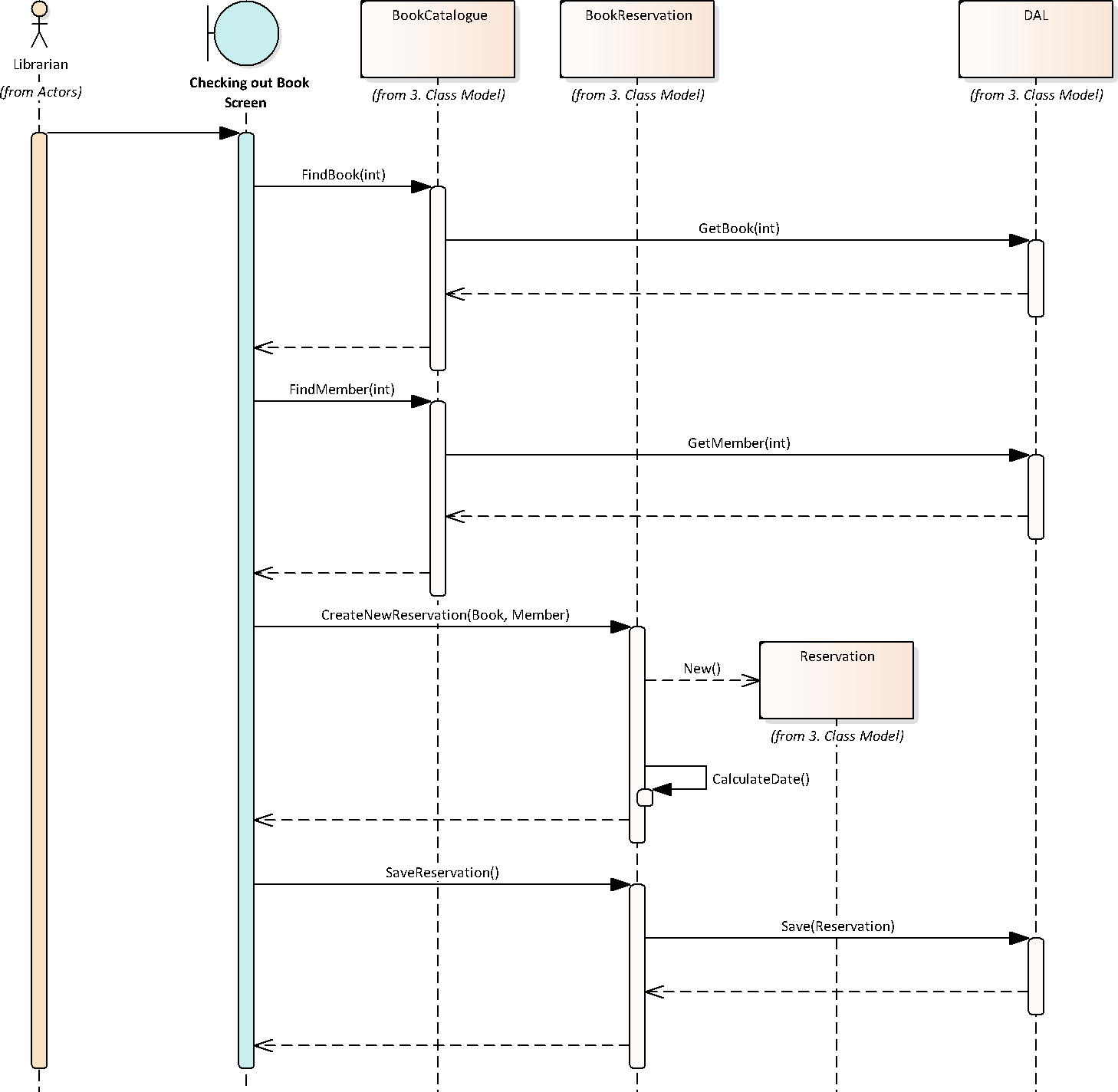


### Alerting expiring due date

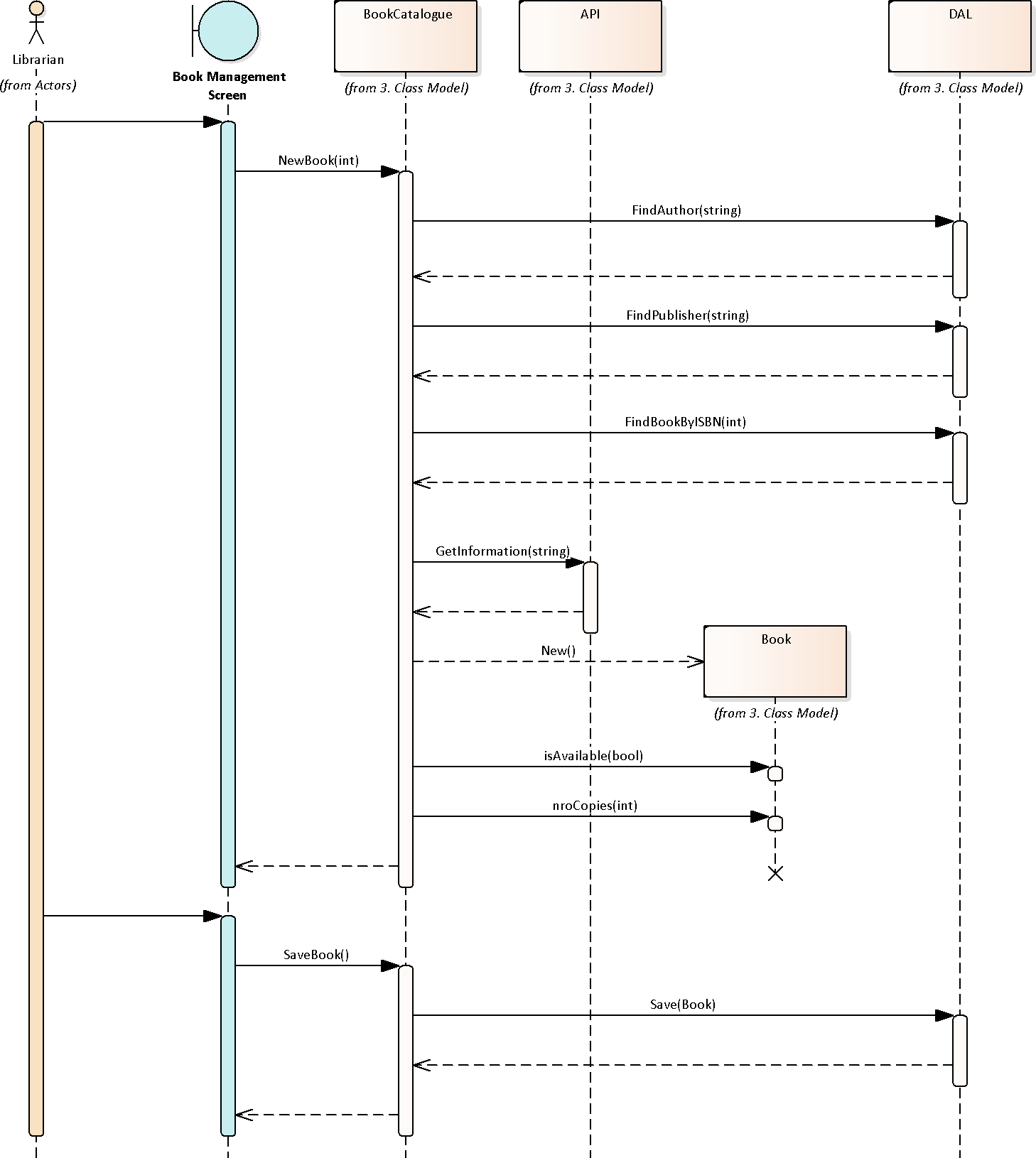


## Sequence Diagram

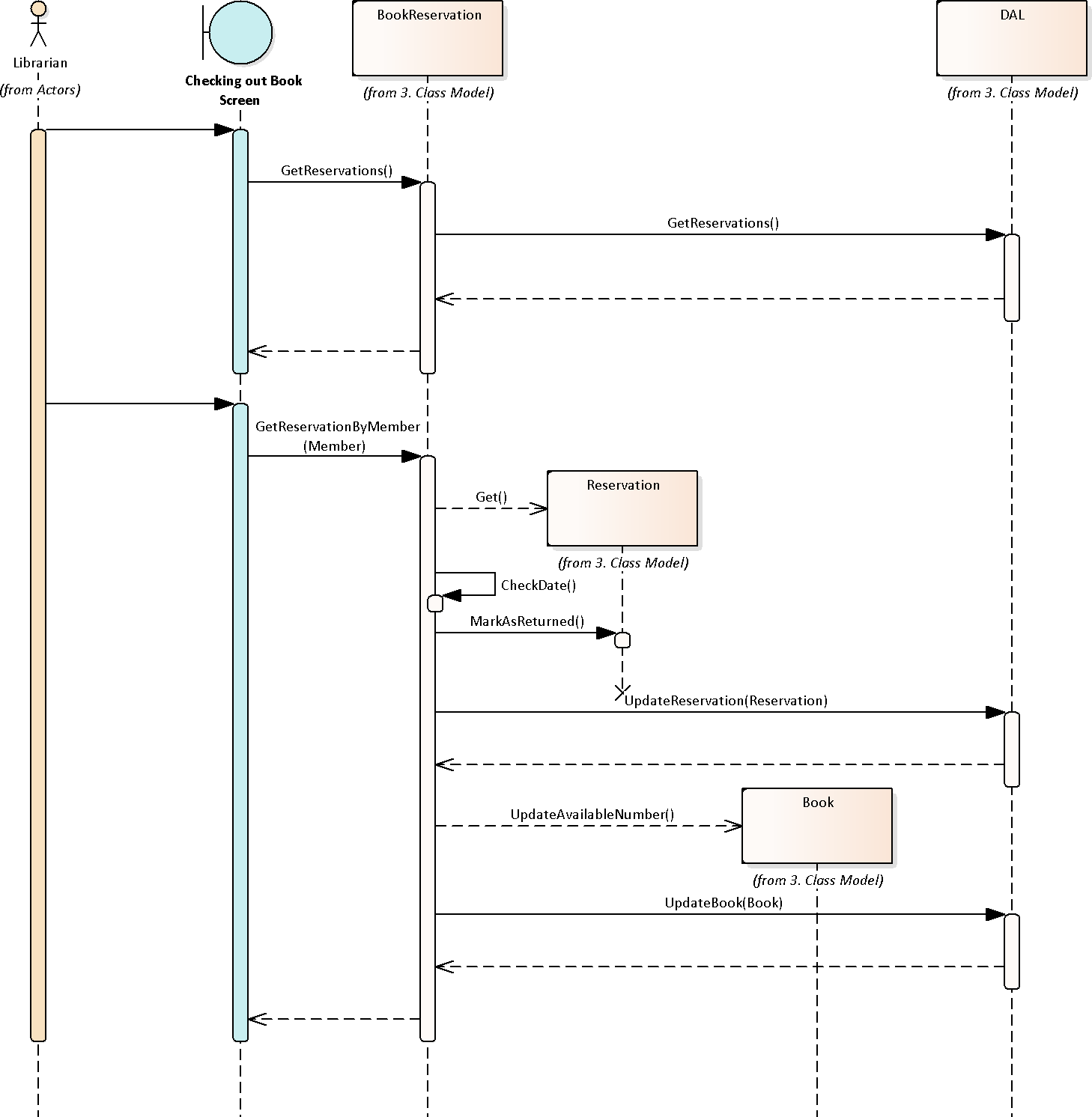
### Checking out Book



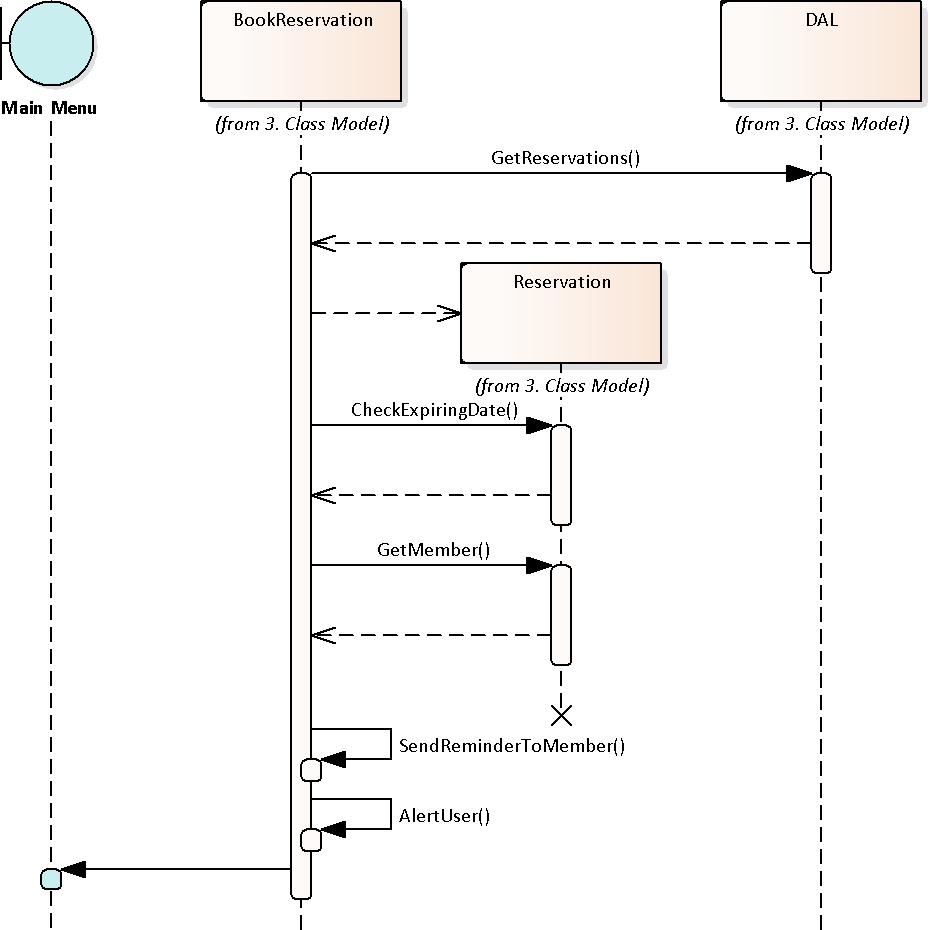
### Adding New Book



### Returning Book



### Alerting expiring due date



# Restrictions

## Functional Requirements

### Register new book manually

|  |  |
| --- | --- |
| Introduction | The System must allow to register new books manually or atumatically. |
| Required Information | * Details of the book * Date of registration |
| Process | * The author of the book must already exist in the system * The publisher of the book must already exist in the system * Fill the ISBN of the book * Verify that the ISBN does not already exists in the system * Choose to either enter information manually or download it from internet   + Fill the name of the book   + Fill the author or authors of the book   + Fill the date of acquisition   + Fill the optional fields * Mark if the book will be available for check out * Fill the amount of copies |
| Goal | The book must exist in the system database with its required information |

### Register new membership

|  |  |
| --- | --- |
| Introduction | The System must allow to register new memberships |
| Required Information | * Member Information |
| Process | * The member must not exist in the DB * The Librarian fills the members information * The System assign a *Member ID* to the Member * The System creates a new Member in the DB |
| Goal | The new member must exist in the system database with its required information |

### Update Book

|  |  |
| --- | --- |
| Introduction | The System must allow to update a book details with online information. |
| Required Information | * ISBN of the book |
| Process | * Search for the required book to update * Update the book details by getting online information * Accept the changes |
| Goal | The book details must be updated with online information |

### Searching for a book in the data base

|  |  |
| --- | --- |
| Introduction | The System must allow to search and list the books registered by using different types of criteria. |
| Required Information | * Any information regarding the book * Any information regarding an author * Any information regarding a publisher |
| Process | * Fill any information regarding a book, author, or publisher * A list of books with matching details should be displayed |
| Goal | A list of books with the desired criteria should be shown |

### Checking out a book

|  |  |
| --- | --- |
| Introduction | The system must allow to check out books to customers and keep track of the time. |
| Required Information | * Customer information * ISBN of the book |
| Process | * Verify that the book exists in the system * Verify that the book is available for checking out * Search for the book in the system * Mark the copy of the book as checked out * Update the remaining available copies of the book * Keep track of remaining days until getting the book back |
| Goal | * The available copes for checking out for the book must be updated * The book and customer information must appear in the list of books to be returned |

## Other Product Requirements

None specified.

## Standards

None specified.

## System Requirements

|  |  |
| --- | --- |
| Operating System: (32- and 64-bit) | * Windows Server 2008 R2 with Service Pack 1 * Windows 7 SP 1 * Windows 8 * Windows 8.1 Microsoft .NET framework 3.5 with Service Pack 1 installed * Windows 10 |
| Memory | * 1 GB RAM (1.5 GB Recommended) for Windows Windows 7 Professional, Windows 8, Windows 10 Windows Server 2008 R2, and 2012 |
| Processor | * 1.7 GHz processor or better |
| Hard Disk | * Minimum free space is approximately 100 mb, with additional 4.5 GB for the Microsoft .NET 4.6.1 framework software, if not previously installed. |
| Connectivity | * Port 80 must be open for ISBN API connection and software updates |
| Screen Resolution | * The minimum supported resolution is 1024x768 |

# Use Cases Specifications

## Use Cases

|  |  |
| --- | --- |
| ID | UC-01-01 |
| Name | **Adding New Book (Manually)** |
| Description | Adding a new book edition entry to the DB |
| Actors | Librarian |
| Triggers | A new book edition is acquired by the library |
| Pre-Condition | The book edition must not exist in the DB already |
| Post-Condition | The new book edition must be accessible from the System search |
| Main Course | 1. The librarian selects the option *Add new Book* 2. The screen *1.1.1.1 Add new Book* will be shown 3. The librarian must enter the ISBN of the book    1. For downloading book information, see **UC-01-02** 4. The System checks if the ISBN exists in the DB    1. If the ISBN already exists, see **EX-01-01** 5. The librarian must select the authors of the book    1. If the author/s does not exist, see **UC-02-01** 6. The librarian must enter the date of acquisition of the book 7. The librarian must complete the rest of the required fields 8. The librarian enters the amount of available editions 9. The librarian selects *Save* 10. The System will create a new entry in the DB with the book information |
| Exceptions | * **EX-01-01**: The System will alert the librarian that a book with the same ISBN already exists in the DB, and won’t allow to continue. |

|  |  |
| --- | --- |
| ID | UC-01-02 |
| Name | **Adding New Book (Automatic)** |
| Description | Add a new book edition entry to the DB, and getting the book information automatically from Internet |
| Actors | Librarian |
| Triggers | * A new book edition is acquired by the library * UC-01-01 |
| Pre-Condition | The book edition must not exist in the DB already  The System must have access to Internet |
| Post-Condition | The new book edition must be accessible from the System search |
| Main Course | 1. After filling the ISBN of the book in screen *1.1.1.1 Add new Book*, the librarian selects *Get information from online data base* 2. The System checks if the ISBN exists in the DB    1. If the ISBN already exists, see **EX-01-01** 3. The system will search for the ISBN online and bring back the information of the book    1. If no matching ISBN is found, see **EX-01-02** 4. The System will autocomplete the book information fields 5. The librarian enters the amount of available editions 6. The librarian selects *Save* 7. The System will create a new entry in the DB with the book information |
| Exceptions | * **EX-01-01**: The System will alert the librarian that a book with the same ISBN already exists in the DB and won’t allow to continue. * **EX-01-02**: The user will be alerted that the ISBN is not correct and wont continue until a valid one is provided or exit. |

|  |  |
| --- | --- |
| ID | UC-02-01 |
| Name | **Adding New Author** |
| Description | Adding a new author entry to the DB |
| Actors | Librarian |
| Triggers | A new book is acquired whose author does not yet exist in the DB |
| Pre-Condition | The author must not exist in the DB |
| Post-Condition | The new author must be accessible from the System search |
| Main Course | 1. The librarian selects the option *Add new Author* 2. A formulary with the author details fields will be shown 3. The librarian must enter the Author names 4. The librarian must select the associated genres 5. The librarian selects *Save* 6. The System will create a new entry in the DB with the author information |

|  |  |
| --- | --- |
| ID | UC-03-01 |
| Name | **Adding New Member** |
| Description | Adding a new library member to the DB |
| Actors | Librarian |
| Triggers | A person applying for a library membership |
| Pre-Condition | The person must not be an existing member |
| Post-Condition | The new member must exist in the DB |
| Main Course | 1. The librarian selects the option *Create new Membership* in screen *1.1.2 Members Management* 2. The screen *1.1.2.1 Add new Member* with the member information fields will be shown 3. The librarian must complete the personal information of the new Member 4. The librarian must check if the new Member wishes to receive emails with offers and news about books 5. The librarian selects *Save* 6. The System will inform the new *Member ID* |

|  |  |
| --- | --- |
| ID | UC-04-01 |
| Name | **Checking out Book** |
| Description | A Member wish to check out a book |
| Actors | Librarian |
| Triggers | A Member checking out a book |
| Pre-Condition | * The book must exist in the library catalogue * The person must be an existing Member |
| Post-Condition | * The amount of available numbers if the checked-out book must be decreased by 1 * A new registry must be created in the *Book Reservation* detailing the member, the book, and when the checked-out book must be returned |
| Main Course | 1. The librarian selects the option *Check-out Book* in screen *1.1.3 Reservations* 2. The screen *1.1.3.1 Create Reservation* will be shown for the librarian to choose the book to check out and the member checking out the book 3. The librarian searches for the book to check out 4. The System checks if the book can be checked out    1. If the book cannot be check out, see **EX-04-01** 5. The System checks if there are available editions    1. If there no amount available to check out, see **EX-04-02** 6. The librarian must select the member checking out the book    1. If the member exceeds the amount of checked out books, see **EX-04-03**    2. If the member has checked out books that exceed certain threshold, see **EX-04-04**    3. If the member does not exist, see **UC-03-01** 7. The System will inform the return date 8. The librarian confirms the information with the Member 9. The librarian selects *Check out* 10. The System creates a new entry in *Book Reservation* |
| Exceptions | * **EX-04-01**: Some books are not allowed to be checked out of the library, these books will be marked as un check-out-ables. The user won’t be able to continue with the process. * **EX-04-02**: If there is no amount available left of book to check out, the System will alert the librarian. The user won’t be able to continue with the process. * **EX-04-03**: If the Member exceeded the amount of allowed checked out books, the System will alert the librarian and won’t be able to continue with the process. * **EX-04-04**: If the Member has unreturned books that exceed certain amount of days beyond the return date, the System will alert the librarian and won’t be able to continue with the process. |

|  |  |
| --- | --- |
| ID | UC-05-01 |
| Name | **Returning a Book** |
| Description | A Member returning a book |
| Actors | Librarian |
| Triggers | A Member returning a book |
| Pre-Condition | * The book must exist in the library catalogue * The person must be an existing Member |
| Post-Condition | * The amount of available numbers if the checked-out book must be increased by 1 * The registry of the checked-out book must be marked as *Done* |
| Main Course | 1. The librarian searches for the Member in the screen *1.1.3 Reservations*    1. If the Member is not found, see **EX-05-01** 2. The librarian selects the Member returning the book, and mark the returning book as *Returned*    1. If the returning date is beyond the due date, see **UC-05-02** 3. The librarian selects *Done* |
| Exceptions | * **EX-05-01**: The System will alert the Librarian that the Member returning a book is not found and won’t let them continue. |

|  |  |
| --- | --- |
| ID | UC-05-02 |
| Name | **Returning a Book after due date** |
| Description | A Member returns a book after its due date |
| Actors | Librarian |
| Triggers | UC-05-01 |
| Pre-Condition | * The book must exist in the library catalogue * The person must be an existing Member |
| Post-Condition | * The amount of available numbers if the checked-out book must be increased by 1 * The registry of the checked-out book must be marked as *Done* |
| Main Course | 1. The System informs the librarian that the book being returned is past its due date 2. The System shows the fine that the Member must pay 3. The librarian informs the Member the fine they must pay 4. The librarian selects *Done* |

|  |  |
| --- | --- |
| ID | UC-06-01 |
| Name | **Alerting of expiring due dates** |
| Description | The System must alert daily the librarian of books with expiring due dates |
| Actors | System |
| Triggers | Daily automatic trigger |
| Pre-Condition | At least one book in *Book Reservation* |
| Post-Condition | The System must show an alert showing the number of books to be returned that day |
| Main Course | 1. The System analyze the entries in *Book Reservation* 2. If an entry contains a book that should be returned the same day, the entry is added to *Expiring Today* 3. In the System main menu, a counter with the number of books to be returned will be shown |

## Log In / Log Out Management

When launching the application, and user will be prompted to input their username and password, assigning their corresponding profile according the system.

Exiting the application with automatically log out the user.

## Multi Language Management

The system will allow the use of a flexible multi language service. This means that the user will be able to change the default display language or add a new one.

## Log Management

The system will log the operations of the logged user, registering the username, activity and additional information associated with the activity.

The system will also provide means to show the log and filter through these fields.

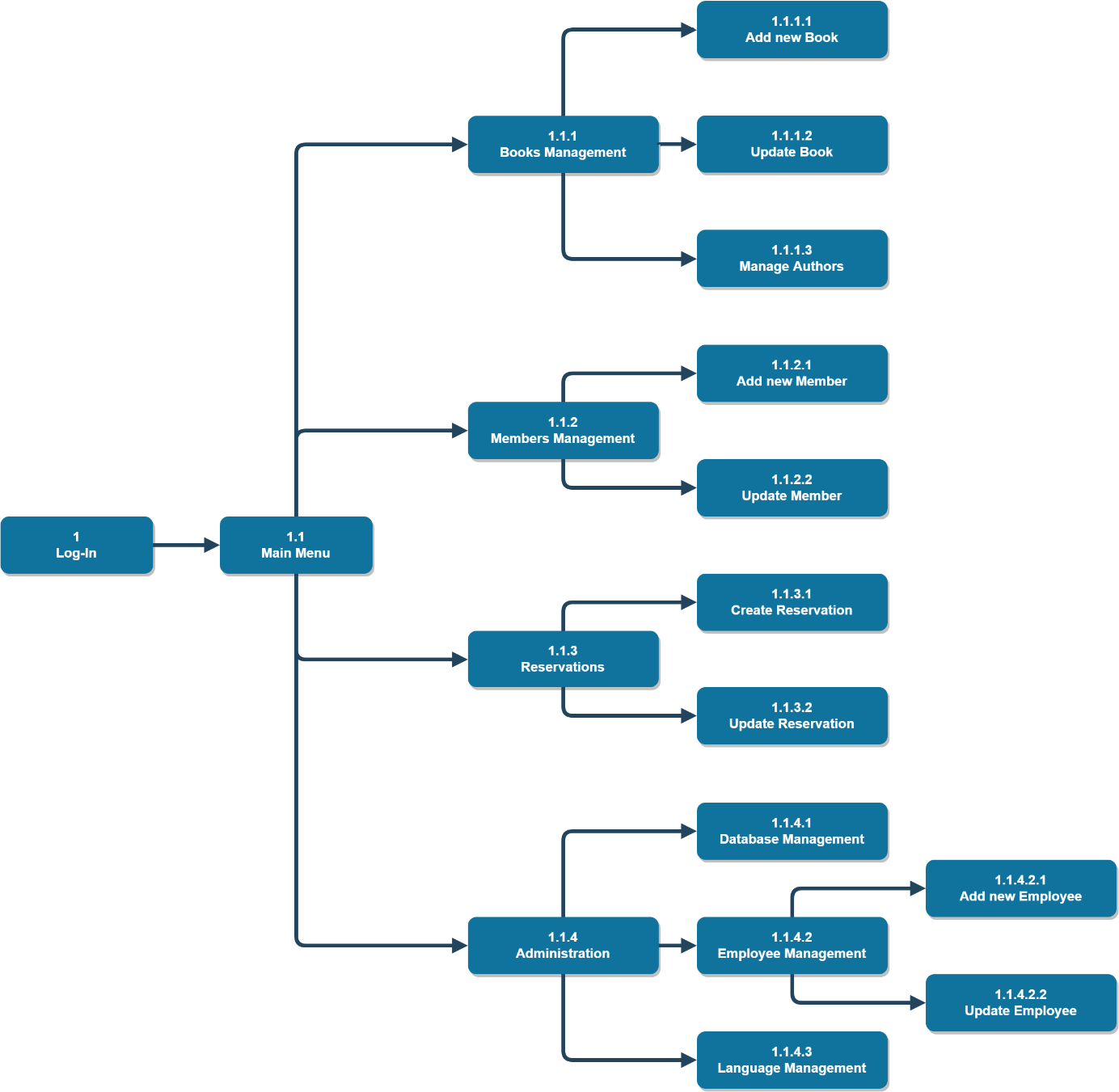
## Back Up Management

The system will allow to create and manage back up of its Database.

## Encryption Management

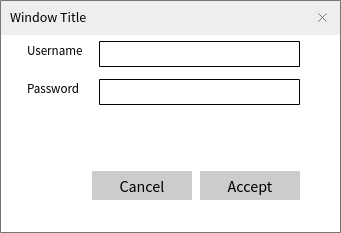
The system will enforce the encryption of any sensitive information in the Database.

# Navigation Map

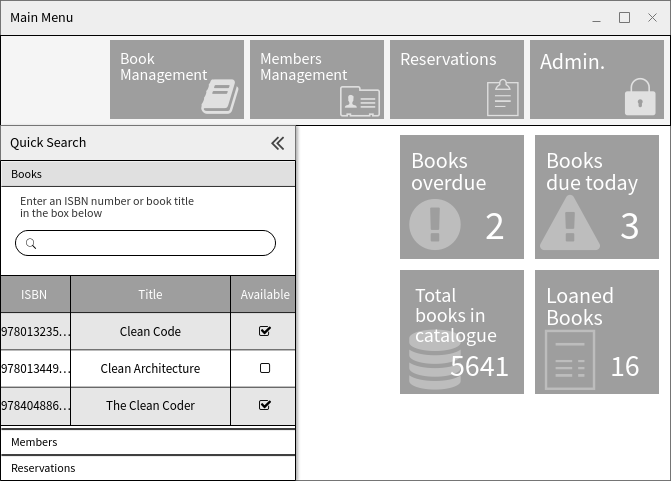


# User Interface Prototypes

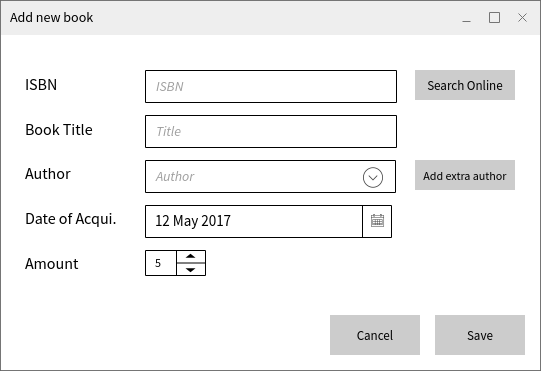
### Log in window



### Main Menu



### Add new book



### Book Reservations

