Library Database: Inception

SSE 657 Project 1

Usman Khan

Stephen McMurry

Mikae Nguyen

September 7, 2020

Table of Contents

Table of Figures

Table of Tables

Introduction

The MUS Development team has been assigned to rebuild the library database system for Bibb County Public Library. In this report, the team will be describing the Inception process regarding the creation of the library database system. The Inception process details system development regarding the development and design of software and is typically the first step completed in typical system development scenarios. Inception defines the scope and vision of the project, as well as includes the design of use cases that demonstrates the system's capabilities and functions. The process typically only covers about 10 percent of total use cases and does not entirely define all the system's capabilities. Inception is meant to establish the project vision and worth. Once the Inception phase has been approved by the client, MUS Development will then move on to the Elaboration portion of the project.

Purpose

MUS Development has been tasked to build an online library database. The library database will store a collection of books that will allow users and admins to check out the books within the database inventory. The library database will allow the library's users and admins to fulfill their library needs. The library management system will increase overall productivity in the system by allowing individual users to have direct access to catered functions specifically curated with their needs in mind. This new system will allow users to have access to their public library without physically being there.

Scope

The library database will feature two primary actors – admins and users. A user can either view the books in the database or be issued a book by an admin. An admin, like a user, can also view books in the database and be issued books. In addition to this, an admin can add books, add users, view users, issue books, and return books. Admins can also have the option of creating another library database as well as having the capabilities of manipulating the current database.

Glossary

Table 1. Glossary

Term	Definition	Example
Actor	Something with behavior.	Admins, members, teacher, student
Admin	An actor who can add or view users, add or view books, issue books or view issued books, return books, and create or reset the library database. Also, see Member template for the Admin template.	201, ukhan, password, true
Book	A template that contains the book ID, title, genre and price.	1000001, Harry Potter and the Sorcerer's Stone, Fantasy, 1500
Library	A database that stores Books.	Bibb County Public Library
Member	An actor and a template that contains user ID, username, password, and a Boolean saying whether they are an admin. This template can also apply to Admin.	1001, pmacneil, GoBears, false
Use Case	Scenarios that describe an actor using a system to support a goal.	See pg. 9 and pg. 11
Use Case Model	A set of typical scenarios of using a system.	See pg. 8

Project Overview

The library system has two primary actors: members and admins. Both members and admins can interact with the library system, however members have limited permission and can only interact with the system in a restricted manner. Admins have access and can interact with the library system in an unrestricted manner due to the difference in levels of permission that admins have over users. The admins and members' roles are shown in **Figure 1**.

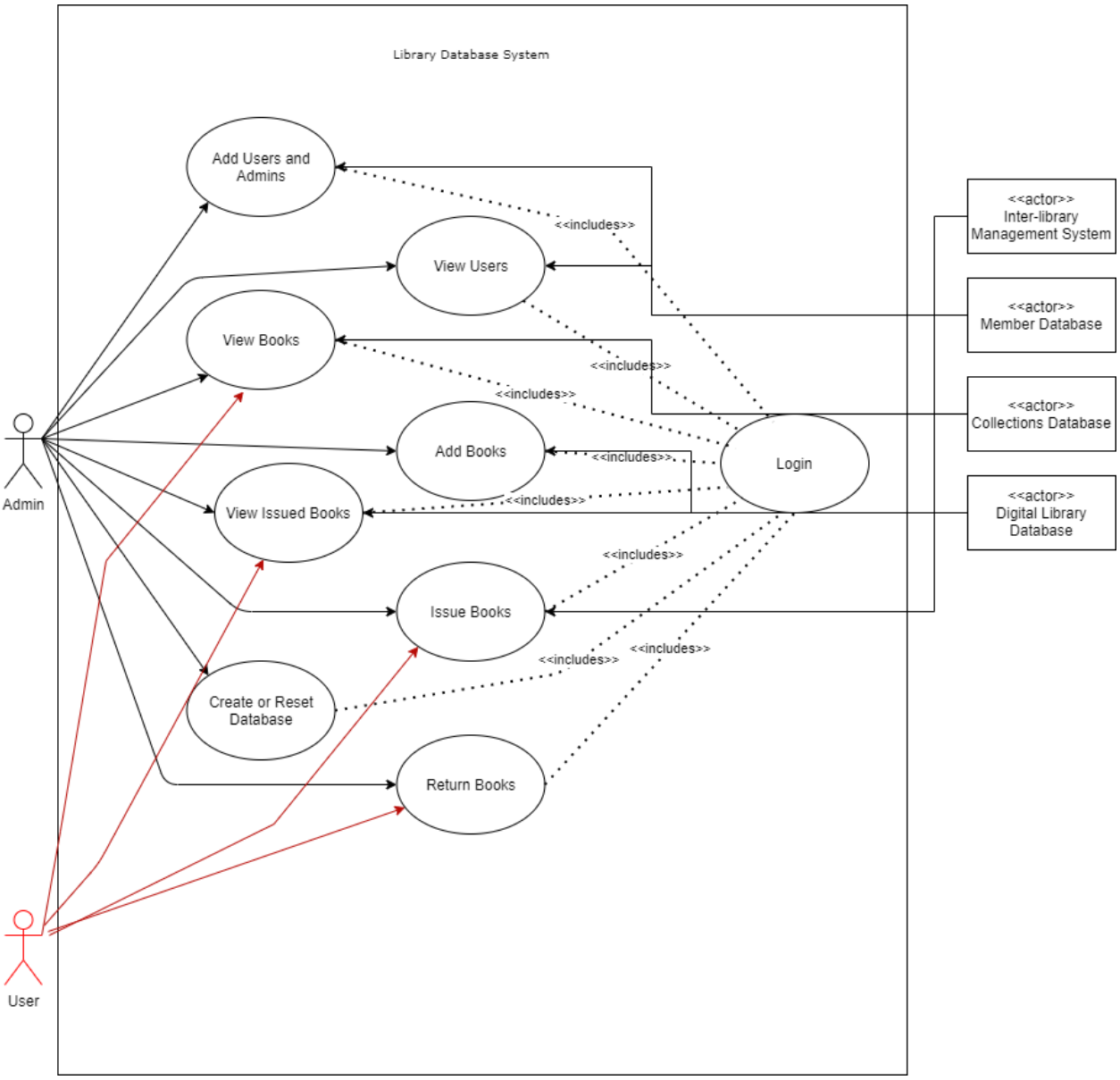

Figure 1. Use Case Diagram

Admins have access to all capabilities that the user can perform. In addition, an admin can create new users and view existing users. This process lets the admin assign different traits to another user's account, such as a user ID, username, password, and have the permission to

determine whether the user will have admin or member privileges. An admin can view total user accounts with the "view users" feature. Another function that an admin has, is the ability to view current books within the database system, as well as add or remove books on an as-needed basis. This process works similarly to the process of adding a new user to the system. However, instead of inputting information regarding the user's account, the admin instead can create a new book within the system that has traits such as book ID, title, genre, and price. Once the book is created, the primary actors can then view the newly created book with the "View Books" feature. Admins can issue books to be lent and can view currently issued books. This process allows for the admin to have the ability to issue a book with a specified return date. This book can then be returned with the "Return Book" feature. Admins also have permission to create a new database entirely or remove a currently existing library database.

Members can view the current books that are housed within the library database system as well as having the ability to check out and return books. Members can also request a book that is not within the library database system by making an inter-library loan request. This allows for users to loan a book from a different library. However, not all inter-library loan requests are logistically possible to be fulfilled due to various reasons. If members fail to return the book by the end of the lease period, then a late fee will be charged to their account.

UC1 depicts the use case of the admin's add books feature and includes a diagram displaying said use case.

Use Case UC1: Add Book

Scope: Library Database System

Level: User Goal

Primary Actor: Admin

Stakeholders and Interests:

- Admin: Wants to be able to add a book as defined in the Glossary and be displayed when view books is selected.
- Member: Wants to view the newly added book using the view books feature.

Preconditions: Admin enters correct username and password to login in successfully.

Success Completion: Book is added to library database.

Main Success Scenario:

1. Admin navigates the user menu to Add Books.
2. Admin enters the book ID, title, genre, and price.
3. Book is added to the database.
4. Admin is notified the book is added.
5. Admin is sent back to the user menu and navigates to View Books.
6. Admin sees the book has been added with corresponding descriptions.

Extensions:

1. Book ID is already in use:
    a. If book ID is already in use, admin is notified to put in a different book ID.

Special Requirements:

- Admin account has already been created at the start of the database conception or added by another admin.

Technology and Data Variations List:

N/A

Frequency of Occurrence:

Once the book is entered, the process is completed.

An example walkthrough of **UC1** is found in **Figure 2**.

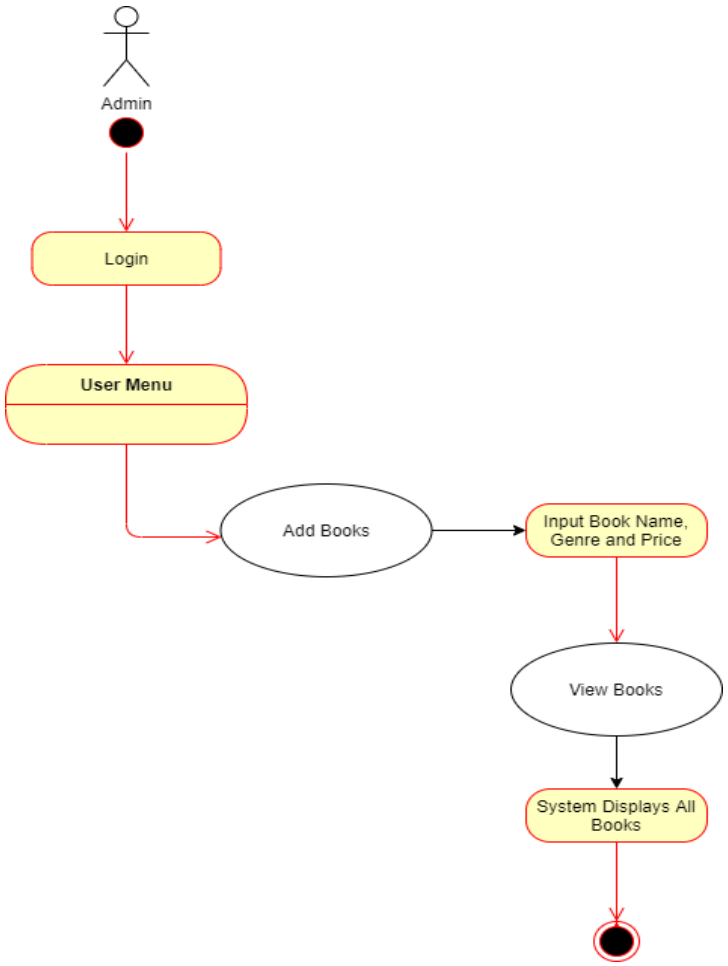

Figure 2. Use Case Model 1

Use Case UC2: Acquire Loan for Book

Scope: Interlibrary Loan System and Library Database System

Level: User Goal

Actors: Admin and Member

Stakeholders and Interest:

- **Member**: Wants to see if a book is available within the inventory of the Library Database. If the book is not available within the library's inventory, the member is given the option to request this book to be lent through the Interlibrary Loan System from another library.
- **Admin**: Wants to view requests and either approve and send interlibrary loan requests or deny requests.

Pre-conditions: Members and admins must successfully log into the system.

Success Completion: Book is either lent to the library from a different partnering library or the user is informed that the book is not available to be loaned.

Main Success Scenario:

1. Member navigates the menu and checks to see if the book requested is within the library inventory.
2. If the book is not within the library inventory, the member will be given the option to put in a request to lend the book using the interlibrary loan system.
3. The member will place a request and the system will notify the member that their request has been put in. An admin will later view this request.
4. Once the admin has viewed the request and approves of the request, the admin will request a partner library for the loan.
5. The system will then notify the member that their request has been approved and their book will arrive in an estimated amount of days.
6. The member will return the book to the library which will then be returned to the library, which will return the book to the partnering library.

Extensions:

1. If the book is already within the library inventory:
    a. The system will notify the member that the library is in the inventory.
    b. The process will resume at step 3.
2. If the interlibrary request is denied:
    a. The system will notify the member that their request has been denied.

Special Requirements: The library must be partnered with other libraries to participate within the Interlibrary Loan system.

Technology and Data Variations List: N/A

Frequency of Occurrence: This process occurs when a user makes a request and ends when the book is returned.

An example walkthrough of **UC2** is found in **Figure 3**.

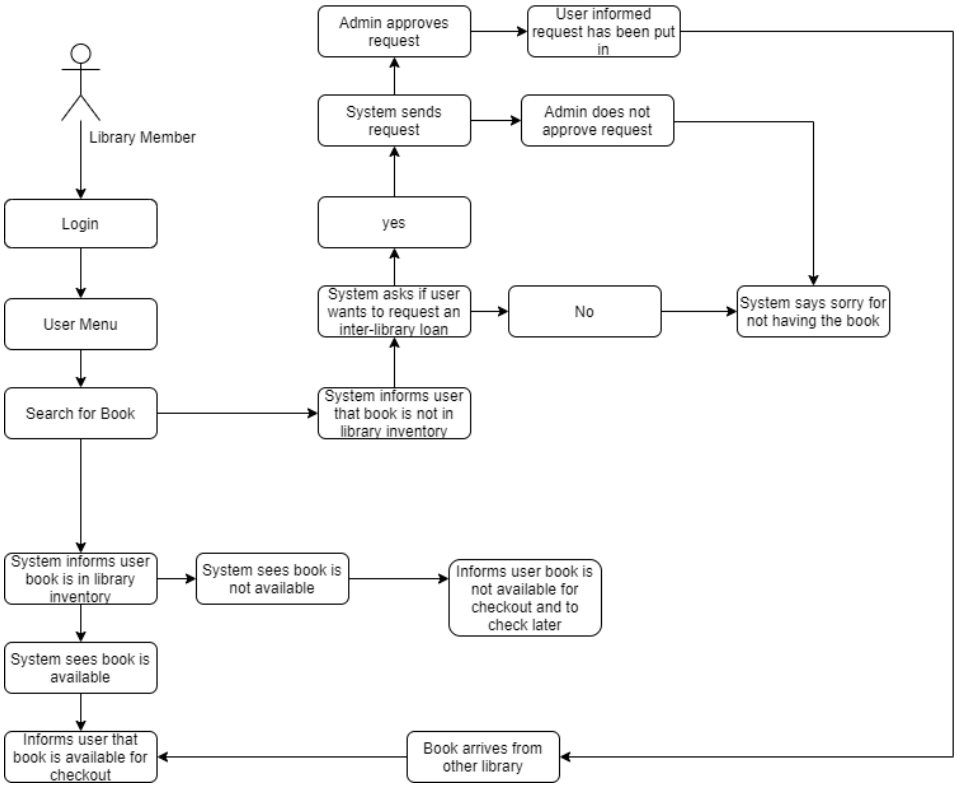

Figure 3. Use Case Model 2

Resources

The IDE that the team will use to develop the system is Eclipse. Eclipse is an IDE that is user friendly and is focused on Java programming. Java also includes several packages that will be used during development, such as Java Swing and SQL integration. Java Swing will be used to develop the GUI that will allow the user to navigate through different use cases within the system. The Java SQL package will allow the team to make an online database that will store user and book information. The team will also use standard Java packages during the creation of the system.

The team will create a variety of classes to suit the library system's needs such as Book, Library, Member, etc. These classes will showcase an extensive use of object-oriented programming in order to increase the efficiency of the overall system and the coding behind the system.

For the Elaboration phase, the team will create and program the entire system as detailed during the Iteration phase. The team hopes to fulfill all the requirements that were given to the team by the client. The team will then prepare the Elaboration phase portion of the report and will submit a completed report as a final deliverable.

References

Larman, C. (2004). *Applying UML and patterns: An introduction to object-oriented analysis and design and the unified process*. Prentice-Hall.