

Online Digital Library – Final Submission

Link to index.html

<http://flip3.engr.oregonstate.edu:9416/>

Link to Operational Website

<http://flip2.engr.oregonstate.edu:60250/>

Contents

Link to index.html	1
Link to Operational Website	1
Executive Summary.....	1
Database Outline	2
a) Overview	2
b) Database Outline, in Words.....	2
Entity-Relationship Diagram	5
Schema.....	6
User Interface with Informative Titles	7

Executive Summary

Our initial proposal was to build an online library app to track and connect patrons, online libraries, and e-book lending. Overall, the project scope stayed the same. However, three main areas of change included: **1.)** Changing the ERD and Schema to better represent intersection tables. (Our most significant shift was to make “Books” an intersection table for “Authors” and “Genres” to fix this M:M relationship.) **2.)** Fixes for missed requirements, misunderstanding the prompt, or typos. **3.)** Scaling back size of the HTML design to make the project more manageable.

Feedback from our TA was most valuable as Andrew Kamand helped us find missing attributes in the UI (“rental duration”, “rental date”), understand the difference between ERD/Schema, notice inconsistencies, and guide future steps with tips such as create a NULL-able relationship (Books). We took all his suggestions and implemented them.

Feedback from peers was also highly valuable. It helped us understand how to correctly format DDL files for direct use on phpMyAdmin and catch typos in the outline. Some classmates had excellent design suggestions, but we were not able to implement everything due to time limitations.

Finally, we had to scale back our HTML website design. It provided for many complex select, search, update, and delete queries. As we began working on the final web app, we realized that extra time was needed for implementation and testing. Thus, we simplified our final design to meet the requirements and deadline.

Database Outline

a) Overview

With an ever-increasing move to a digital world, libraries are forced to adapt and provide options to library patrons to access written works. By creating an Online Digital Database, libraries can access information regarding patrons, books, and book trends. This helps the library decide what books to buy in the future, based on demand, and informs the number of e-subscriptions.

This is a backend database for an online digital library and its patrons. This database allows patrons to find available e-books by searching multiple online libraries. This library app is currently small but is hoping to grow. It currently serves 300 patrons and 6 online libraries that represent a collection of over 40,000 books. A well-organized database would allow the app builders to track 3 main categories: patron (whom), checked out items (what), and e-library (where).

b) Database Outline, in Words

The database stores information about the following entities (bolded attributes are required in the database):

- **Patrons**: This is an object table that records patrons who want to use the library app. A patron can exist without checking out any items.
 - **IDPatron**: INT, auto_increment, unique, notNULL, PK
 - **FirstName**: VARCHAR(500), Null, UNIQUE
 - **LastName**: VARCHAR(500), Null, UNIQUE
 - **Email**: VARCHAR(500), UNIQUE
 - Primary Key: *IDPatron*
 - Foreign Keys: N/A
 - Relationship:
 - 1:M relationship between *CheckedOutItems* and *Patrons*
 - Constraints:
 - *FullName*: Checks that first and last name combined is unique.
- **CheckedOutItems**: This represents each transaction where a patron checks out items. It serves as the transaction table and records the patron id.
 - **IDCheckedOutItem**: INT, auto_increment, not NULL, PK
 - **IDPatronFK**: INT, notNull
 - Primary Key: *IDCheckedOutItem*
 - Foreign Keys:
 - **IDPatronFK**: fk to *Patrons* – ON DELETE CASCADE & ON UPDATE CASCADE implemented
 - Relationship:
 - 1:M relationship between *CheckedOutItems* and *Patrons*
 - 1:M relationship between *CheckedOutItems* and *CheckedOutDetails*
- **CheckOutDetails**: This is an object table that stores check out information about a book. To exist it must be connected to a book. It contains *CheckedOut*, *RentalDate*, *RentalDuration*, and foreign keys to connect it to *Books*, *Patrons*, and *CheckedOutItems*.

- **IDCheckOutDetail**: INT, auto_increment, unique, notNULL, PK
- **CheckedOut**: INT, notNull, DEFAULT 1 (to indicate book in not checked out)
- **RentalDuration**: INT, notNull, DEFAULT 21 (to indicate 21 days)
 - *Constraints: RentalDuration type CHECK (RentalDuration IN (7, 14, 21))*
- **RentalDate**: DATE, notNull
- **IDCheckedOutItemFK**: INT, notNull
- **IDCollectionFK**: INT, notNull
- Primary Key: **IDCheckOutDetail**
- Foreign Keys:
 - **IDCheckedOutItemFK**: fk to *CheckedOutItems* – ON DELETE CASCADE & ON UPDATE CASCADE implemented
 - **IDCollectionFK**: fk to *Collections* – ON DELETE CASCADE & ON UPDATE CASCADE implemented
- Relationship:
 - This table is an intersection table that facilitates a M:M relationship between *Books* and *CheckedOutItems*.
 - 1:M relationship between *CheckOutDetails* and *Books*
 - 1:M relationship between *CheckOutDetails* and *CheckedOutItems*
- **Authors**: This is an object table that stores author information.
 - **IDAuthor**: INT, auto_increment, unique, notNULL, PK
 - **AuthorFirstName**: VARCHAR(500), notNull
 - **AuthorLastName**: VARCHAR(500), notNull
 - Primary Key: **IDAuthor**
 - Foreign Keys: N/A
 - Relationship:
 - 1:M relationship between *Author* and *Books*
 - Constraints:
 - **FullName**: Checks that first and last name combined is unique.
- **Genres**: This is an object table that stores genre categories.
 - **IDGenre**: INT, auto_increment, unique, notNULL, PK
 - **GenreCategory**: VARCHAR(500), NOT NULL unique
 - Primary Key: **IDGenre**
 - Foreign Keys: N/A
 - Relationship:
 - 1:M relationship between *Genre* and *Books*
- **Books**: This is an intersection table that stores book information including title and connects 'Author' and 'Genre'.
 - **IDBook**: INT, auto_increment, unique, notNULL, PK
 - **Title**: VARCHAR(500), notNull
 - **IDAuthorFK**: INT
 - **IDGenreFK**: INT
 - Primary Key: **IDBook**

- Foreign Keys:
 - **IDAuthorFK**: fk to Author – ON DELETE CASCADE & ON UPDATE CASCADE implemented, can be NULL
 - **IDGenreFK**: fk to Genre – ON DELETE CASCADE & ON UPDATE CASCADE implemented, can be NULL
- Relationship:
 - This is an intersection table between *Authors* and *Online Genres*. It solves the M:M relationship between *Authors* and *Genres*. It also stores the Title of the book which could be used as a unique identifier. However, we thought a separate integer for “IDBook” was a simpler primary key. (An author can write in many genres; a genre contains many authors.)
 - 1:M relationship between *Books* and *ItemLocations*
 - 1:M relationship between *Genre* and *Books*
 - 1:M relationship between *Author* and *Books*
- Constraints:
 - *Title_Unique*: Checks that the title is unique.
- *ItemLocations*: This is an intersection table that connects online libraries to books.
 - **IDItemLocation**: INT, auto_increment, notNULL, PK
 - **IDBookFK**: INT, fk to *Books*
 - **IDOnlineLibraryFK**: INT, fk to *OnlineLibraries*
 - Primary Key: **IDItemLocation**
 - Foreign Keys:
 - **IDBookFK**: INT, fk to *Books* – ON DELETE CASCADE & ON UPDATE CASCADE implemented
 - **IDOnlineLibraryFK**: INT, fk to *OnlineLibraries* – ON DELETE CASCADE & ON UPDATE CASCADE implemented
 - Relationship:
 - This is an intersection table between *Books* and *Online Libraries*. It solves the M:M relationship between *OnlineLibraries* and *Books*. (A library can have many books; a book copy can be held by many libraries.)
 - 1:M relationship between *OnlineLibraries* and *ItemLocations*
 - 1:M relationship between *Books* and *ItemLocations*
- *OnlineLibraries*: This is an object table that stores the name of the library.
 - **IDOnlineLibrary**: INT, auto_increment, unique, notNULL, PK
 - **LibraryName**: VARCHAR(500), Null, Unique
 - Primary Key: **IDOnlineLibrary**
 - Foreign Keys: N/A
 - Relationship:
 - 1:M relationship between *OnlineLibraries* and *ItemLocations*
 - This relationship is used as a pipeline to facilitate the M:M relationship between *OnlineLibraries* and *Books*

Entity-Relationship Diagram

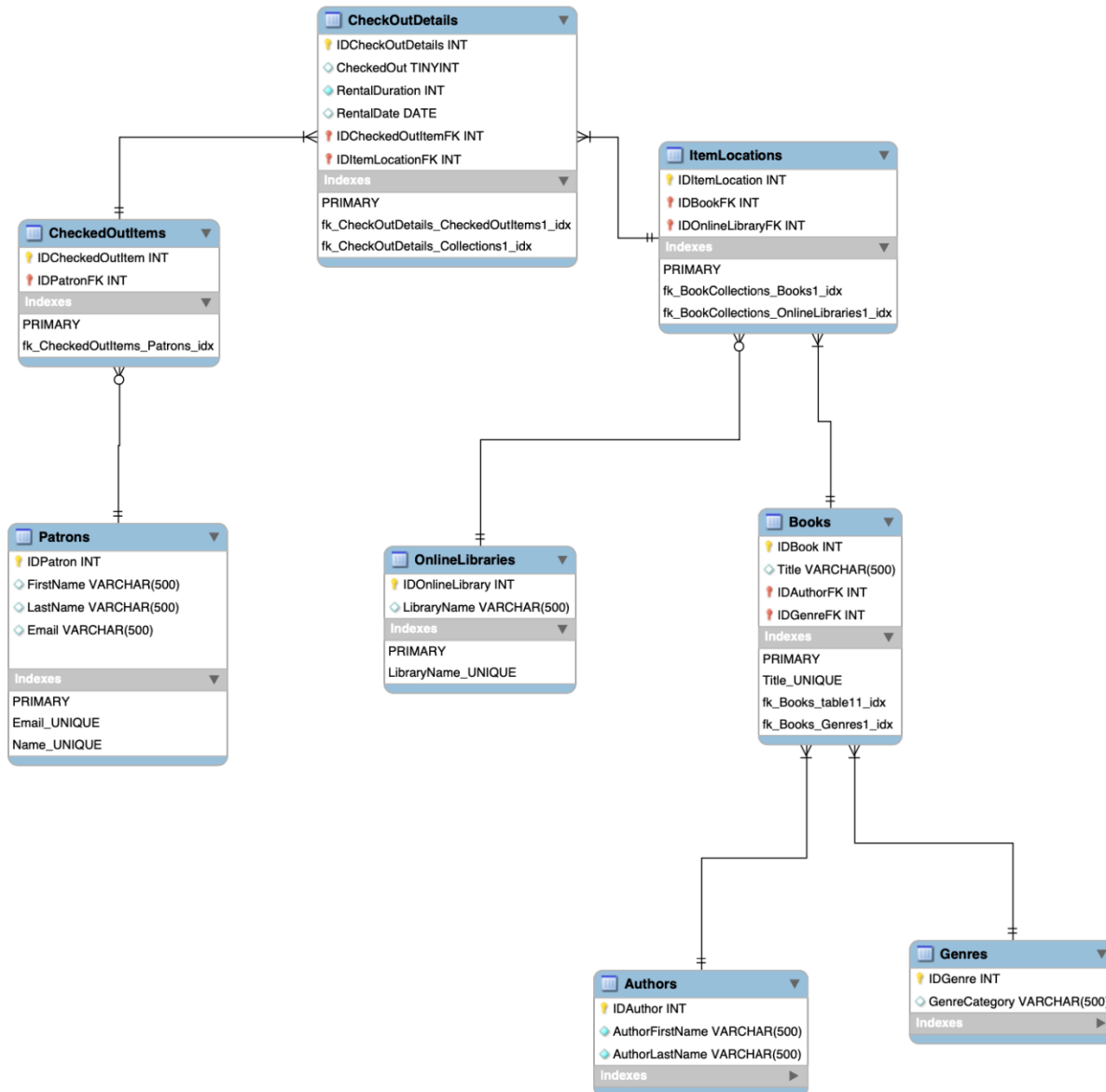


Figure 1. Revised Online Digital Library ERD created with MySQL

Schema

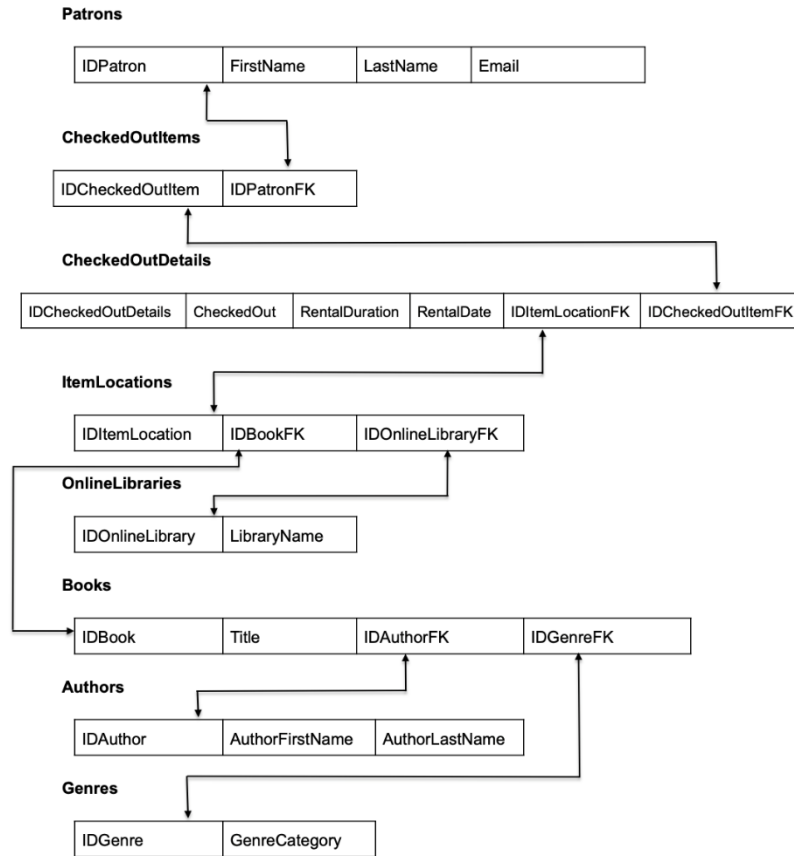


Figure 2. Online Library Schema (revised per feedback and Professor Safonte ED Discussion post on 7/19/22)

User Interface with Informative Titles

Read Home Page

HOME | PATRONS | CHECKED OUT ITEMS | ITEM LOCATIONS | ONLINE LIBRARIES | BOOKS | AUTHORS | GENRES

Welcome!

Use the above navigation bar to view the Online Libraries and more connected to our network!

Read/Search/Add/Update/Delete Patrons Page

HOME | PATRONS | CHECKED OUT ITEMS | ITEM LOCATIONS | ONLINE LIBRARIES | BOOKS | AUTHORS | GENRES

Patrons

View, add, update or delete patrons from your database.

Overview Patrons

> > >

IDPatron	FirstName	LastName	Email	Delete
1	Jonas	Tilde	jtilde@gmail.com	Delete
2	Amanda	Danai	danniii@gmail.com	Delete
3	Shinya	Ragna	sra@gmail.com	Delete

Add Patron

To add a new patron to the database, please enter their information below. Email is not required.

First Name Last Name Email

Submit Query

Update Patron Email

To update patron email, select patron and enter email.

Select Patron to Update

E-mail

Submit Query

Read/Add new Check Out Item using Patron FK, Item Location FK/Delete Checked Out Items Page

[HOME](#) | [PATRONS](#) | [CHECKED OUT ITEMS](#) | [ITEM LOCATIONS](#) | [ONLINE LIBRARIES](#) | [BOOKS](#) | [AUTHORS](#) | [GENRES](#)

Checked Out Items

View, add, update, or delete checked out items. (Notes: This is the page that best represents CheckedOutItemDetails; however Checked Out Items made for a better page title. Both tables represent this list of checked out items. In addition, this page pulls from the tables: Patrons, ItemLocations, Books, OnlineLibraries, Authors, and Genres. It makes it easier to see how items would be updated.)

Overview Checked Out Items

Checked Out Item ID	Patron First Name	Patron Last Name	Title	Library Name	Length Checked Out	Date Checked Out	Delete
1	Jonas	Tilde	Where the Crawdads Sing	Universal Digital Library	21	Tue May 10 2022 00:00:00 GMT-0700 (Pacific Daylight Time)	<button>Delete</button>
1	Jonas	Tilde	Where the Crawdads Sing	Los Angeles Public Library	7	Sat Nov 12 2022 00:00:00 GMT-0800 (Pacific Standard Time)	<button>Delete</button>
2	Amanda	Danai	Where the Crawdads Sing	Universal Digital Library	14	Sun Jul 10 2022 00:00:00 GMT-0700 (Pacific Daylight Time)	<button>Delete</button>

Check Out Item

Patron Amanda Danai ▼
Item Location Where the Crawdads Sing at Los Angeles Public Library ▼
Checked Out Date mm / dd / yyyy
Check Out Length (days) ▼
Check Out
Cancel

Read/Search using Online Library FK/Add using Book Title FK, Online Library FK/Delete Item Locations Page

[HOME](#) | [PATRONS](#) | [CHECKED OUT ITEMS](#) | [ITEM LOCATIONS](#) | [ONLINE LIBRARIES](#) | [BOOKS](#) | [AUTHORS](#) | [GENRES](#)

Item Locations

View, add or delete book locations. There is no need for an edit function as an item can only *be* at or *not be* at a location.

Overview Item Locations

> >

Item Location ID	Library Name	Book Title	Delete
1	Los Angeles Public Library	Where the Crawdads Sing	<button>Delete</button>
2	Universal Digital Library	Where the Crawdads Sing	<button>Delete</button>

Search for Books at a Certain Online Library

Displays all books and their item location ID at a library.

Library Name:

Add Book/Library Location

Book:

Library Location:

Read/Search/Add/Update/Delete Online Libraries Page

[HOME](#) | [PATRONS](#) | [CHECKED OUT ITEMS](#) | [ITEM LOCATIONS](#) | [ONLINE LIBRARIES](#) | [BOOKS](#) | [AUTHORS](#) | [GENRES](#)

Online Libraries

View, add, update or delete an online library in your database.

Please use the Item Locations button to add books to each library.

Overview Online Libraries

> > >

IDOnlineLibrary	LibraryName	Delete
1	Los Angeles Public Library	<button>Delete</button>
2	New York Public Library	<button>Delete</button>
4	Universal Digital Library	<button>Delete</button>
3	World Digital Library	<button>Delete</button>

Search for an Online Library

Online Library Name

Search Library Reset

Add Online Library

Online Library Name

Add Library

Update a Library Name

Select Online Library to Update:

New Library Name:

Submit Query

Read/Add Book, Author FK, and Genre FK/Update using Author FK, and Genre FK with NULLable option/ Delete Books Page

[HOME](#) | [PATRONS](#) | [CHECKED OUT ITEMS](#) | [ITEM LOCATIONS](#) | [ONLINE LIBRARIES](#) | [BOOKS](#) | [AUTHORS](#) | [GENRES](#)

Books

View, add, update or delete books in your database.

Overview Books

Book ID	Title	Author First Name	Author Last Name	Genre	Delete
1	Where the Crawdads Sing	Colleen	Hoover	Fiction	<button>Delete</button>
2	It Ends with Us	Delia	Owens	Fiction	<button>Delete</button>
3	Verity	Colleen	Hoover		<button>Delete</button>
5	Sparring Partners	John	Grisham	Fiction	<button>Delete</button>

Add a New Book

To add a new book to the database, please enter their information below. Select Author and Genre from the drop downs.

If desired genre or author does not exist, please first create these fields using the following links: [Authors](#) | [Genres](#)

Title

Author Genre

Submit Query

Update Book

To update a book's author or genre, select the book and then select author and genre.

Select the Book to Update

Author Genre

Submit Query

Read/Search/Add/Delete Authors Page

[HOME](#) | [PATRONS](#) | [CHECKED OUT ITEMS](#) | [ITEM LOCATIONS](#) | [ONLINE LIBRARIES](#) | [BOOKS](#) | [AUTHORS](#) | [GENRES](#)

Authors

View, add, update or delete authors in your database.

CAUTION any updates or deletes to an author will affect all associated books. Deleting an author from your collection will also delete all their books from the collection.

Overview Authors

> > > >

IDAuthor	AuthorFirstName	AuthorLastName	Delete
2	Colleen	Hoover	<button>Delete</button>
1	Delia	Owens	<button>Delete</button>
3	Elin	Hilderbrand	<button>Delete</button>
5	Harry	Potter	<button>Delete</button>
4	John	Grisham	<button>Delete</button>

Search for Author by Last Name

Displays all authors matching this last name.

Last Name

Search

Reset

Add Author

To add a new author to the database, please enter their information below.

First Name Last Name

Submit Query

Read/Add/Update/Delete Genres Page

HOME | PATRONS | CHECKED OUT ITEMS | ITEM LOCATIONS | ONLINE LIBRARIES | BOOKS | AUTHORS | GENRES

Genres

View, add, update or delete genres in your database.

Overview Genres

> > > >

IDGenre	GenreCategory	Delete
3	Biography	<button>Delete</button>
1	Fiction	<button>Delete</button>
4	Horror	<button>Delete</button>
2	Nonfiction	<button>Delete</button>

Add Genre

Genre Category

Add Genre

Update Genre Category

CAUTION this will update the genre name for all associated books.

Select Genre Category to Change:

Change Genre Category:

Submit Query