Bookworm Database

By: Matt Jones, Ben Fowler, Julina Tusieseina, Paula (Julie)

Paredes-Pozas

IS 4420 - Database Fundamentals

Professor Stan Pugsley

Information Systems Minor Abroad

May 31, 2022

Table of Contents

1. Project Overview

Public libraries are owned by local governmental bodies for the general purpose of ensuring that access to information of all kinds is readily available to the public. Libraries accomplish this goal through two major avenues. Firstly, libraries provide pieces of inventory to the public that take various forms, most predominantly books and movies. Customers can come into public libraries during operating hours, find their desired piece of inventory, and check it out of the location. The customer then has 30 days to return the inventory or they are subjected to a fine of $1 per day. Secondly, libraries host classes with a wide variety of subject matter that customers are welcome to attend.

The libraries only have one primary source of revenue, the aforementioned late fees. They do not charge for class registration or for the initial 30 days of inventory rental because they are not geared for profit and are owned by the public.

Given this business model, the need for a comprehensive, intuitive database is apparent. Classes need registered students, teachers, materials, and assigned rooms that meet the needs of the subject matter. Inventory needs to be classified by its medium, whether it is a book, movie, etc. Its status as in stock or out of stock needs to be tracked. Libraries are also in the somewhat unique position of needing to make sure that pieces of inventory return safely.

These are but a few of the many uses that a public library could have for a well-maintained database. Suppose they did not make use of such a database. In that case, they could face significant costs in the form of lost inventory, uncollected fees, or unsatisfied customers who are either unable to obtain their desired inventory or had some issues arise with their desired classes. In the end, the quality of the Public Library System's database could be the determining factor in whether or not the goal of mass public informational availability is achieved.

2. User Requirements

- What will the system accomplish?
  - The Public Library System will provide library members with easy and efficient access to library items for checkout
- What functions will it perform?
  - Users can search/browse library items (books, movies, periodicals, music, etc)
  - Users can find the specific location of library items
  - Users can add library items to their shopping cart
  - Users can check out library items
  - Users can return library items
  - Users can view fines associated with Customers
  - Users update/add/change their member info
  - Users can search/browse/signup for classes/workshops
  - Admin can limit books checked out
  - Admin can view fines
  - Admin can view/add/update or delete library cardholders
  - Admin can add/update/delete employees
- Who will need access to the data?
  - Admin can access check-in/check-out data
  - Users can access certain information about themselves
- What data will be tracked?
  - What items are checked out
  - How long they have been checked out
  - Check out and return dates
- What questions will the system be able to answer?
  - What books are out right now
  - What are the total fees to be collected
  - What books are overdue at the moment

3. Business Rules

- A user can have zero or more books checked out at one time
- A user can have zero or more overdue books
- A user can belong to one or many subcategories (customer, employee, manager)
- A subcategory can refer to a customer, an employee, or a manager
- Users can make many reservations.
- User information is not collected until the user makes a reservation.
- Each reservation applies to only one user.
- Reservations create one or many logs in the system.
- Each log is only triggered by one reservation.
- Each reservation can only apply to one class.
- Each class can have many reservations.
- A class does not need any reservations to exist in the system.
- Each reservation can only apply to one room.
- Each room can have many reservations.
- A room does not need any reservations to exist in the system.
- Each reservation can only apply to one piece of inventory.
- Each piece of inventory can have many or no reservations.
- Logs update inventory, with each only applying to one piece of inventory, though each piece of inventory can have multiple logs but inventory information isn't collected until at least 1 log is made.
- Inventory consists of books and movies.
- Each class takes place in exactly one room.
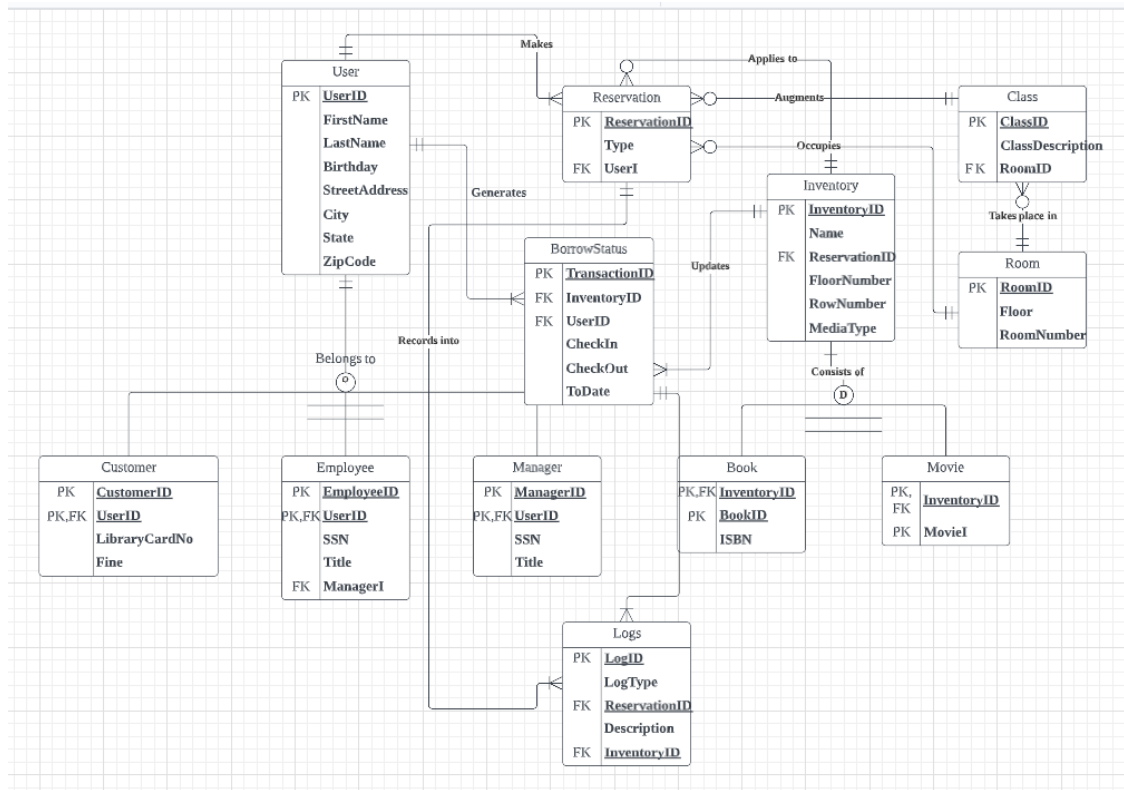- Each room can have zero or many classes.
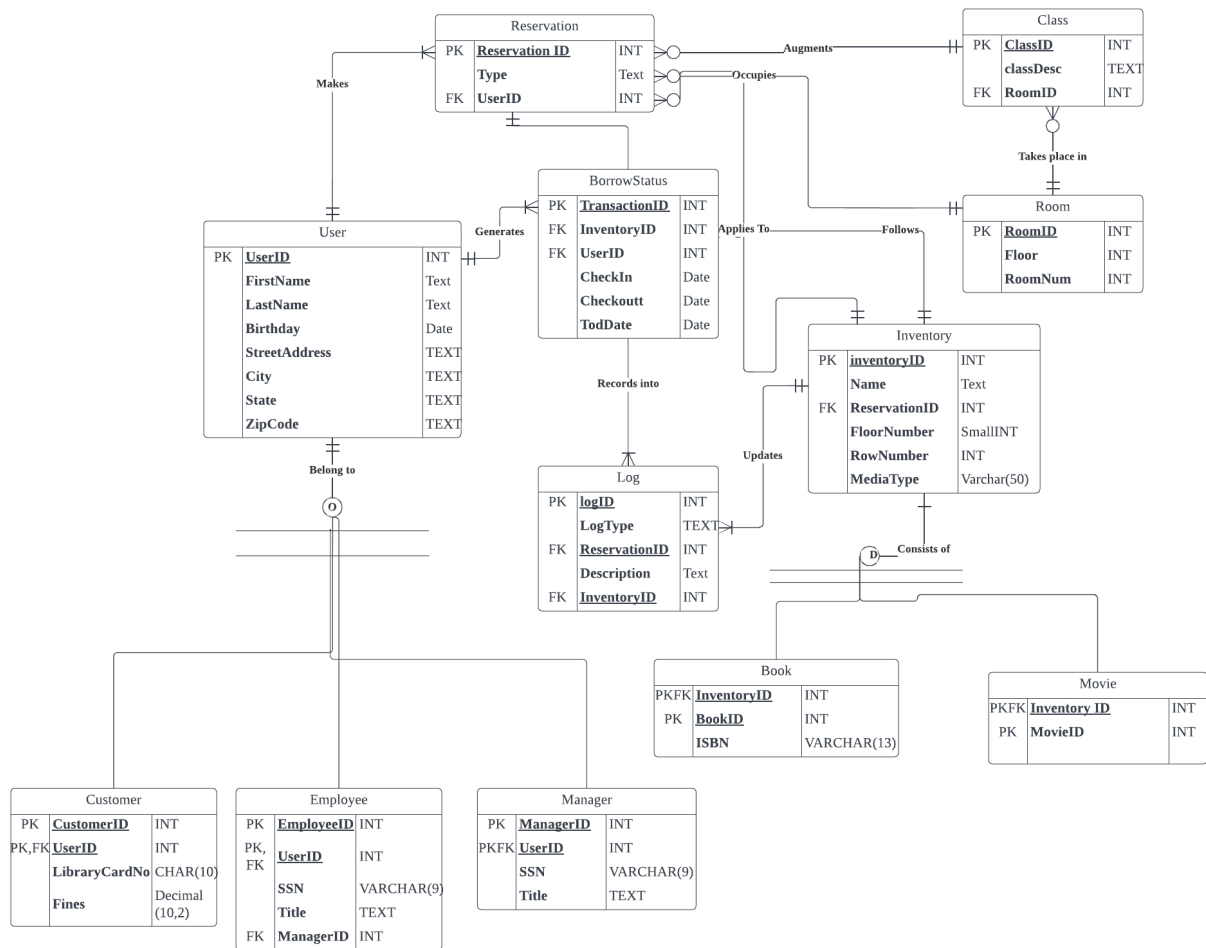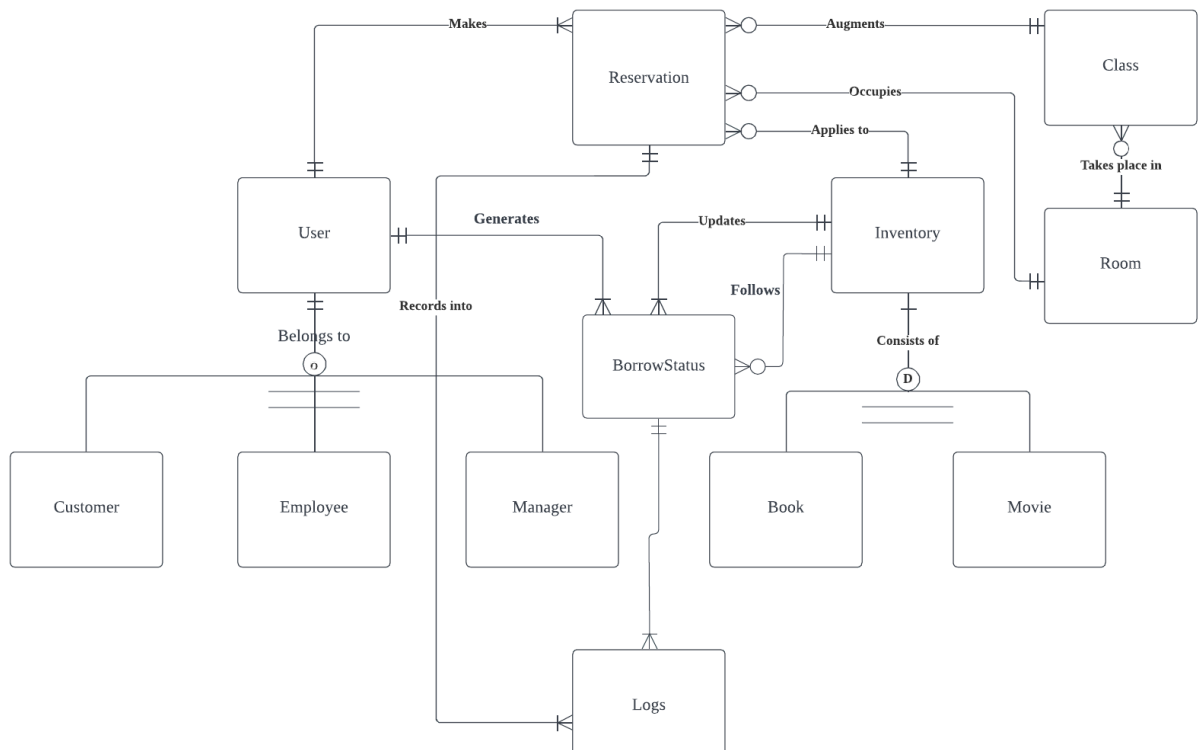
4. Data Outputs/Business Questions

1. What state do our customers mainly come from?
   a. Marketing would benefit from this data output.
   b. This output would be used for strategic planning.
   c. Marketing would learn the geographic locations of our key buyer demographics.
   d. This data would help us to strategically plan our marketing strategy in order to either further lock down states that are already buying or capture states that are not.

2. What item does User #1 have a reservation right now?
   a. Library Manager
   b. This output would be used for operational needs.
   c. The manager would use this data output as a random sample in order to determine which products are popular.
   d. After determining which products are popular, the manager can then make them more readily available, to make the customer's experience easier and encourage them to return.

3. What room and floor will ClassID 29 be held in?
   a. Customer
   b. This output would be used for operational needs
   c. Using the data the customer will be able to find the room they need for the class they want.
   d. It is important that the customer experience be easy and convenient so that they return to check out more materials, so their ability to find their classroom is integral to the functionality of the business model.

4. What is our earliest checkout?
   a. Library Manager
   b. This output would be used for operational needs
   c. This data would be used to verify if our system is functional
   d. By cross-referencing with other data, we will be able to determine whether the system is accurately checking the status of the inventory.

5. What is our latest check-in and what was the name of the customer?

a. Employees

b. This output is operational

c. This will allow employees to see information about our customers

d. This command will allow the employee to see what our most recent item returned is and what their name is. This is super helpful if the item is damaged.

6. How many books do we have overall?

a. Library Manager

b. This output would be for operational needs

c. This data output would be used to ensure that no inventory is missing.

d. If an inventory item is missing and a customer is unable to make a purchase, then we may lose that customer and any late fees they could have incurred.

7. How many checkouts are there overall?

a. Library Staff

b. This is for operational needs

c. This would help make sure that we are actively aware of what is currently out and how much is out

d. This output will allow the staff to see how many books are out with customers at the moment. Managers can use this data to figure out business solutions to issues they might have with this data

8. What's the ISBN of checkInCheckOutID 1?

a. Library Management

b. This is for operation needs

c. This particular question isn't needed but it's the structure of the command that can be useful. This will allow you to pull needed data for a project.

d. Basically the same reason as above. It will help you pull needed data for a project an employee might need.

9. What is the name of our first item added to the database?

a. Employees

b. This is for strategic planning

c. This allows anyone to check and see if our database is working correctly as far as inserting new data

d. This will help you make sure that all is calm with the database.

10. What City and State are 10 of our customers from?

    a. Employees

    b. This is for strategic planning

    c. We can use this data to research more about our customer

    d. We can figure out if we need to get more locally sourced authors, class topics, etc. by seeing where our customers are from

5. Entity Relationship Diagram

Top conceptual diagram entities and relationships:

- Reservation
  - Makes (User — Reservation)
  - Augments (Reservation — Class)
  - Occupies (Reservation — Inventory)
  - Applies to (Reservation — Inventory)
- Class
  - Takes place in (Class — Room)
- Room
- User
  - Generates
  - Records into
  - Belongs to
- Updates (BorrowStatus — Inventory)
- Inventory
  - Follows
  - Consists of (D)
- BorrowStatus
- Customer
- Employee
- Manager
- Book
- Movie
- Logs

Bottom logical diagram:

Reservation

PK	Reservation ID	INT
	Type	Text
FK	UserID	INT

Makes · Augments · Occupies

Class

PK	ClassID	INT
	classDesc	TEXT
FK	RoomID	INT

Takes place in

Room

PK	RoomID	INT
	Floor	INT
	RoomNum	INT

User

PK	UserID	INT
	FirstName	Text
	LastName	Text
	Birthday	Date
	StreetAddress	TEXT
	City	TEXT
	State	TEXT
	ZipCode	TEXT

Generates · Belong to (O)

BorrowStatus

PK	TransactionID	INT
FK	InventoryID	INT
FK	UserID	INT
	CheckIn	Date
	Checkoutt	Date
	TodDate	Date

Records into · Applies To · Updates · Follows

Inventory

PK	inventoryID	INT
	Name	Text
FK	ReservationID	INT
	FloorNumber	SmallINT
	RowNumber	INT
	MediaType	Varchar(50)

Consists of (D)

Log

PK	logID	INT
	LogType	TEXT
FK	ReservationID	INT
	Description	Text
FK	InventoryID	INT

Book

PKFK	InventoryID	INT
PK	BookID	INT
	ISBN	VARCHAR(13)

Movie

PKFK	Inventory ID	INT
PK	MovieID	INT

Customer

PK	CustomerID	INT
PK,FK	UserID	INT
	LibraryCardNo	CHAR(10)
	Fines	Decimal (10,2)

Employee

PK	EmployeeID	INT
PK, FK	UserID	INT
	SSN	VARCHAR(9)
	Title	TEXT
FK	ManagerID	INT

Manager

PK	ManagerID	INT
PKFK	UserID	INT
	SSN	VARCHAR(9)
	Title	TEXT

6. Database Implementation

Implementing the database required the creation of the schema and the addition of significant amounts of dummy data. The first two screenshots below refer to the former, as they demonstrate that we, as a group, used DDL and DML commands in order to create the necessary tables for our data to be properly situated, and then inserted said data. The first two screenshots contain the DDL commands which pertain to the designing and structuring of the database. The third screenshot contains the DML commands, which involve inserting data into the database.

For the data itself, we used mockaroo to generate a minimum of 50 rows of data for each transactional data, and 15 rows of data for every other table. We then used the data that we generated to create a .sql file consisting of an extremely long series of insert commands that would import the dummy data into our database. As mentioned above, the third screenshot features the insert commands for the dummy data concerning the 'movie' entity. We have also attached the .sql file consisting of the rest of the insert commands used to import the remainder of the dummy data.

```
 3 ●    CREATE DATABASE IF NOT EXISTS bookWorm; #Ok, this will be easy
 4 ●    USE bookWorm;
 5      #DROP DATABASE bookWorm;
 6   ⊖  /* This table is the master holder of all user data. This will be used in conjunction with
 7      Customer, Employee, and Manager
 8    ⌐ */
 9 ● ⊕ CREATE TABLE IF NOT EXISTS USERS (
18
19      # This is the first type of user that we can have. Not too much we need to expalin
20 ● ⊖ CREATE TABLE IF NOT EXISTS Customer (
21      customerID INT NOT NULL,
22      userID INT NOT NULL,
23      libraryCardNo CHAR(10) NOT NULL,
24      fines DECIMAL(10,2) NOT NULL,
25      CONSTRAINT PK_Customer_customerID_userID PRIMARY KEY (customerID, userID),
26    ⌐ CONSTRAINT FK_Customer_userID FOREIGN KEY (userID) REFERENCES Users (userID));
```

```
102   /*
103   │  This keeps track of what classes we offer
104   │  We store the room number here so we know which room its in
105   │  We dont store the floor because its in the Room table
106   │  */
107 ● CREATE TABLE IF NOT EXISTS Class (
108   │  classID INT NOT NULL,
109   │  classDesc TEXT NOT NULL,
110   │  roomID INT NOT NULL,
111   │  reservationID INT,
112   │  CONSTRAINT PK_Class_classID PRIMARY KEY (classID),
113   │  CONSTRAINT FK_Class_reservationID FOREIGN KEY (reservationID) REFERENCES Reservation (reservationID),
114   │  CONSTRAINT FK_Class_roomID FOREIGN KEY (roomID) REFERENCES Room (roomID));
115
```

```
406        #LOAD Movie
407 ●   insert into Movie (movieID, inventoryID) values (1, 26);
408 ●   insert into Movie (movieID, inventoryID) values (2, 27);
409 ●   insert into Movie (movieID, inventoryID) values (3, 28);
410 ●   insert into Movie (movieID, inventoryID) values (4, 29);
411 ●   insert into Movie (movieID, inventoryID) values (5, 30);
412 ●   insert into Movie (movieID, inventoryID) values (6, 31);
413 ●   insert into Movie (movieID, inventoryID) values (7, 32);
414 ●   insert into Movie (movieID, inventoryID) values (8, 33);
415 ●   insert into Movie (movieID, inventoryID) values (9, 34);
416 ●   insert into Movie (movieID, inventoryID) values (10, 35);
417 ●   insert into Movie (movieID, inventoryID) values (11, 36);
418 ●   insert into Movie (movieID, inventoryID) values (12, 37);
419 ●   insert into Movie (movieID, inventoryID) values (13, 38);
420 ●   insert into Movie (movieID, inventoryID) values (14, 39);
421 ●   insert into Movie (movieID, inventoryID) values (15, 40);
422 ●   insert into Movie (movieID, inventoryID) values (16, 41);
423 ●   insert into Movie (movieID, inventoryID) values (17, 42);
424 ●   insert into Movie (movieID, inventoryID) values (18, 43);
425 ●   insert into Movie (movieID, inventoryID) values (19, 44);
426 ●   insert into Movie (movieID, inventoryID) values (20, 45);
427 ●   insert into Movie (movieID, inventoryID) values (21, 46);
428 ●   insert into Movie (movieID, inventoryID) values (22, 47);
```

7. Answer Business Questions

In this section, we highlight a few of the questions that we answered with our database. For the rest of the answers to the business questions, refer to the database.

In order to gain a better understanding of our user base, one of our business questions was in regards to the top 10 cities/states that our customers belong to. The question is as follows "What City and State do our customers mainly come from?" Through our database, we were able to find our top 10 Cities and corresponding states; with Detroit, Michigan being the top city/state. This question is relevant and helpful to us, because in order to better target and serve our users, it is important we use data-driven marketing as well as customer experience in order to better serve our users. The way we accomplished this was by selecting the city and state from the customer table which we joined with the user database, because the user database contains data in regards to the location of the customer. We decided to order by random, because we wanted to pull a randomized record of our customers; we limited the data by ten to avoid excessive data.

```
53    /*
54    Question 10
55    What City and State are 10 of our customers from?
56    */
57 •  SELECT city, state FROM Customer LEFT JOIN Users ON Users.userID = Customer.userID ORDER BY rand() LIMIT 10;
```

	city	state
▶	Detroit	MI
	Kansas City	MO
	Houston	TX
	Lakeland	FL
	Pittsburgh	PA
	Pittsburgh	PA
	Tucson	AZ
	Augusta	GA
	Raleigh	NC
	Midland	TX

```
/*
Question 5
What is our latest check in and what was the name of the customer?
*/
SELECT todDate, firstName, lastname FROM BorrowStatus LEFT JOIN Users ON Users.userID = BorrowStatus.userID
WHERE checkIn = 1 ORDER BY todDate DESC LIMIT 1;
```

	todDate	firstName	lastname
▶	2022-04-14	Blakeley	Dorracott

One of the business questions we proposed was "what is our earliest check-out?" and we learned that the earliest check out was on June 4, 2021. This is an important business question because in knowing when our first check-out was, we can gain a better understanding of our beginnings as an institution. This is an important piece of knowledge when considering our first year financials, statistics and projecting the growth

```
/*
Question 4
What is our earliest check out ?
*/
SELECT todDate FROM BorrowStatus WHERE checkOut = 1 ORDER BY todDate ASC LIMIT 1;
/*
```

	todDate
▶	2021-06-04

pattern of checkouts from the time we opened. We selected the date from the Borrow Status table in order to record the day the transaction happened. Then, we ordered the date in ascending order and limited it to one to find our single most recent check-out.

Along with our first check-out, it is equally as important to know our latest check-in. When inquiring about our most recent check in as well as customer information, we found from our database that the latest check-in was on April 14, 2022 by Blakeley Dorracott. Just as our first check-out, our latest check in provides us with a measure of statistics, check-out patterns and other important information that helps us

understand what we can do to improve our business processes. This function also helps retrieve customer data as well as their recent check-outs for library managers and other employees who need access to such data. To find this information, we selected the transaction date and customer name from the Borrow Status table which we left joined with the Users table. We then used the where command to find check-ins and then ordered the transaction dates by descending in order to return our most recent transaction.