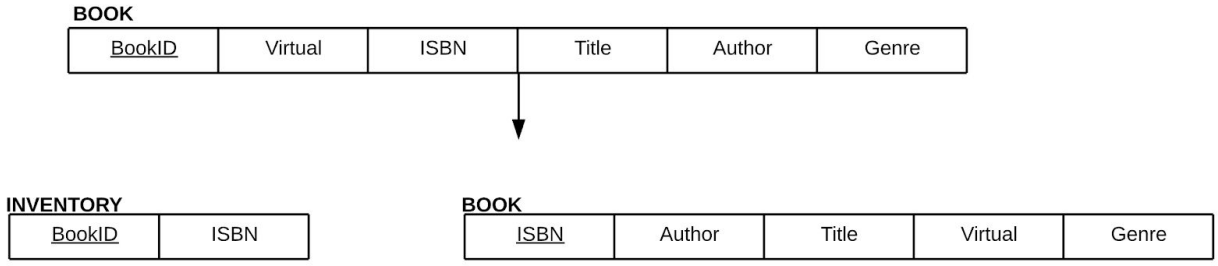
Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

BCNF

CUSTOMER TABLE- This table is in BCNF form. Firstly, there are no multivalued values and/or nested relationships, so the table is in First Normal Form. Secondly, there are no keys that are functionally dependent on just one primary key, i.e. Address and Phone # can rely on Name and CustomerID; therefore the table is in Second Normal Form. Thirdly, this table does not have any transitive dependencies, and there are no non-trivial functional dependencies, so this table is in Third Normal Form and BCNF.

LIBRARIAN TABLE- This table is in First, Second, and Third Normal Form for the same reasons as listed above, (instead of CustomerID, it is LibrarianID). Additionally, this table is also in BCNF form because there are no functional dependencies that involve a primary key.

BOOK TABLE- This table, in order to be in Second Normal Form, has to be split into two different tables. Since the author, title, virtual, and genre attributes were determined by the ISBN, they had to be moved to another table, where ISBN would become the primary key.



BORROWS TABLE- This table is already in BCNF (and the other preceding forms as well) because there are no non primary keys that have a functional dependency.

RESERVES TABLE- This table is already in BCNF (and the other preceding forms as well) because there are no non primary keys that have a functional dependency.

Set of Queries

```
CREATE TABLE customer (
customer_id SERIAL PRIMARY KEY,
name varchar(255) NOT NULL,
address text NOT NULL,
phone char(10) NOT NULL
);
CREATE TABLE librarian (
```

Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

```
employee_id SERIAL PRIMARY KEY,
name varchar(255) NOT NULL,
address text NOT NULL,
phone char(10) NOT NULL
);

CREATE TABLE inventory (
book_id SERIAL PRIMARY KEY,
ISBN varchar(13) NOT NULL,
FOREIGN KEY (ISBN)
REFERENCES book (ISBN)
);

CREATE TABLE book (
ISBN varchar(13) PRIMARY KEY,
author text NOT NULL,
title text NOT NULL,
virtual BOOLEAN NOT NULL,
genre text NOT NULL
);

CREATE TABLE borrows (
customer_id int NOT NULL,
book_id int NOT NULL ,
rent_date DATE DEFAULT CURRENT_DATE,
due_date DATE DEFAULT CURRENT_DATE + INTERVAL '15 days',
FOREIGN KEY (customer_id),
REFERENCES customer (customer_id),
FOREIGN KEY (book_id),
REFERENCES inventory (book_id)
);

CREATE TABLE reserves (
customer_id int NOT NULL,
employee_id int NOT NULL,
book_id int NOT NULL,
reserve_date DATE DEFAULT CURRENT_DATE,
```

Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

```
expiration_date DATE DEFAULT CURRENT_DATE + INTEGER '3',
FOREIGN KEY (customer_id)
REFERENCES customer (customer_id),
FOREIGN KEY (employee_id)
REFERENCES librarian (employee_id),
FOREIGN KEY (book_id)
REFERENCES inventory (book_id)
);

INSERT INTO customer (name, address, phone)
VALUES ('John Jones', '223 Asbury Ave, Ewing, NJ 08560', '8553613599');

INSERT INTO customer (name, address, phone)
VALUES ('Tim Smith', '23 West Ave, Ewing, NJ 08560', '8553313567');

INSERT INTO librarian (name, address, phone)
values ('James Blake', '123 Oak Ave, Ewing, NJ 08560', '7751234567');

INSERT INTO librarian (name, address, phone)
values ('Jon Peterson', '11 Chestnut Street, Ewing, NJ 08560', '7756789012');

INSERT into inventory (isbn) VALUES ('9992223331235');

INSERT into inventory (isbn) VALUES ('9992223331236');

INSERT INTO book (ISBN, author, title, virtual, genre)
VALUES ('9992243339879', 'Gish Jen', 'The Resisters', 'FALSE', 'History');

INSERT INTO book (ISBN, author, title, virtual, genre)
VALUES ('9992243339880', 'Andrew Kriwak', 'The Bear', 'TRUE', 'History');

INSERT INTO book (ISBN, author, title, virtual, genre)
VALUES ('9992243339887', 'John Lacester', 'The Wall', 'FALSE', 'History');

INSERT INTO borrows (customer_id, book_id)
VALUES ('3', '27');
```

Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

```
INSERT INTO borrows (customer_id, book_id)
VALUES ('3', '35');

INSERT INTO reserves (customer_id, employee_id, book_id)
VALUES ('1', '1', '19');

INSERT INTO reserves (customer_id, employee_id, book_id)
VALUES ('4', '2', '32');

UPDATE customer
SET name = ‘Jon B Peterson’, address = ‘11 Chestnut Street, Ewing, NJ 08560’, phone =
‘7756789012’
WHERE customer_id = ‘<customer_id>’;

UPDATE librarian
SET name = ‘James C Blake’, address = ‘123 Oak Ave, Ewing, NJ 08560’, phone =
‘7751234567’
WHERE librarian_id = ‘<librarian_id>’;

UPDATE book SET isbn='9992223339889' WHERE isbn='999222333987';

DELETE FROM reserves WHERE expiration_date < 'TODAY';

DELETE FROM borrows WHERE customer_id='2' and book_id='11';
```

Finding which books are overdue

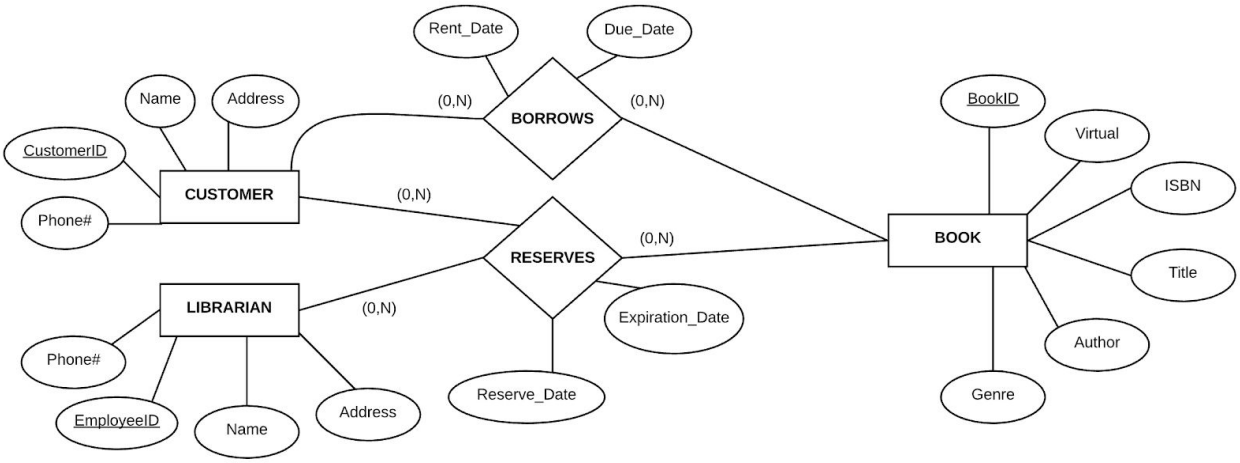
```
SELECT name, title, due_date FROM book NATURAL JOIN inventory
NATURAL JOIN borrows NATURAL JOIN customer
WHERE due_date < 'TODAY';
```

Selecting titles/physical copies of books in the history genre

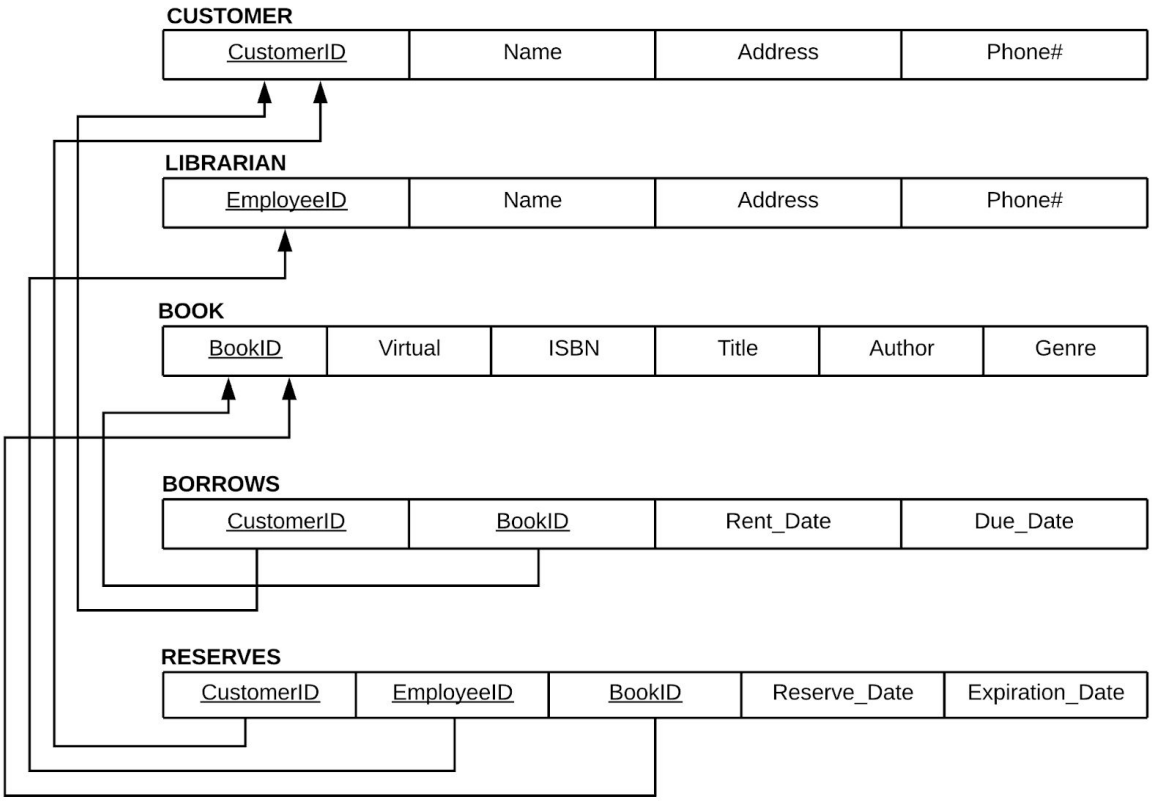
```
SELECT title, book_id FROM book
LEFT JOIN inventory ON book.isbn=inventory.isbn
WHERE book.genre=’History’
);
```

Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

Entity-Relationship Diagram



Relational Schema



Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

Initial database size (approximate number of records)

Currently, the approximate number of records would be roughly around two hundred books, depending on the number of users accessing the system.

Types and average number of searches

Types of searches include: book searches (physical and virtual), reserving books.
Average number of searches: 150 per day.

Project Proposal and Specifications (Updated)

Problem statement

Just recently, a new local library has opened in Ewing Township, NJ. However, this library does not currently have a database in place to manage daily library activities, and only rely on a written system to keep track of book inventory, rentals, etc.

Objective of the module

Our team has been tasked with creating this database in order to create a quick, convenient library experience. Our objective is to create a sustainable library database.

Description of the desired end product, and the part you will develop for this class

This database would be able to automate typical library tasks and activities such as borrowing books, keeping track of rentals (borrowers), and book information. This database would provide/help librarians/borrowers with rentals, book inventory, book information, and the like. There would be two interfaces to this database system: the librarian interface and the user interface. The librarian(s) would be responsible for adding and modifying new book inventories, book information, and borrowers (e.g. adding new members). Regarding books, the librarian would be able to check-in, reserve, and return books for the members. The borrowers would be able to search the database for books, check-in/out books, reserve, and renew books as well. Our database will give the option of renting either a physical or a digital copy of the book. Regarding the database system itself, librarians and borrowers will be able to search for books by using their title, author, or genre through joining the database tables. By reserving a book, there would be a reserve date and an expiration date, when you would have to return the book. Additionally, each book will have its own unique identification number(whether it is physical or virtual), so it will be able to be located when needed. Overall, as stated above, this database system will be able to perform the normal daily activities/tasks of a library.

Team Members: Adam Casto, Nikola Kilibarda, Poean Lu
Library Management System
July 6, 2020

Description of the importance and need for the module, and how it addresses the problem

The importance of this database is that it will create a more convenient, user-friendly experience for the borrower and the librarian. It addresses the problem statement because it will help alleviate the pressure of having to keep up with the daily book inventory with only a pen and paper system. By moving library activities online, there will be less hassle to maintain and keep track of the library inventory.

Plan for how you will research the problem domain and obtain the data needed

We will research library reading lists online and pass them into the database for testing purposes. If the database system would be implemented in an actual library, we would be putting in the actual book inventory in the database.

Other similar systems/approaches that exist, and how your module is different or will add to the existing system

All public libraries have online databases used to keep track of borrowed books. Additionally, there are other businesses that have a similar system as ours, such as movie rental companies or video game rental companies. We will add the feature of having either online or physical copies of books, a feature that not every library has.

Possible other applications of the system (how it could be modified and reused.)

This system can also be modified to keep track of renting video games and movies. Once the borrower checks out the item(s), he or she would be sent a notification saying when that certain movie or video game is due back. Another scenario would be renting skis and snowboards: as soon as they are rented, the borrower would have a certain time limit to use the desired vehicle (borrower's request), before getting a notification about returning the skis or snowboards.

Performance – specify how and to what extent you will address this

We will maximize the performance of our database by optimizing the query performance. This will allow our database to run quicker and smoother, and there will be less delay in retrieving information and checking out books.

Security – specify how and to what extent you will provide security features

Regarding security, the librarian/database administrator will have more freedom to access the full database. In order to keep from reporting and tracking an incorrect book inventory,

borrowers will not be able to have full access to the database system. They will only be able to search for books, view availability/unavailability, etc.

Backup and recovery – specify how and to what extent you will implement this

We will back up a copy of the rental information of the books in the library every day. The information about the books available will be backed up weekly. This data could be backed up to a separate hard drive in a different location.

Technologies and database concepts the team will need to learn, and a plan for learning these

In order to complete this database system, we would need to learn web programming (Python) in order to create a sufficient user interface. Also, we would need to learn how to create the tables and the relationships between entities, including attributes as well. Additionally, learning database languages, such as SQL, would be a necessity, in order to be able to successfully communicate with the database.

A diagrammatic representation of the system boundary that specifies what data you will model and which queries you will implement

The data we would be representing is keeping track of book inventory and the borrowers who borrow them. Each book will have the following properties: title, author, genre, ISBN number, and book/license ID number. If the rental books are overdue, reminders/notifications are sent to the borrower. Another aspect of data we can represent is enabling the borrower to see if a book is available or not.