

Digital Library Management System
CS 306 Project

Alper Çamlı 30858
Ege Yardımcı 31024
Emre Berk Hamarat 31188
Korhan Erdoğan 30838



Digital Library Management System Description

The **Digital Library Management System** is designed to manage and streamline the operations of a library in a digital environment. It allows for efficient management of books, members, loans, reservations, fines, librarians, and multiple library branches. This system offers librarians the ability to handle daily tasks with ease, including the borrowing and returning of books, categorizing books into different genres, managing member accounts, and ensuring the timely collection of overdue fines.

The system consists of ten core entities: **Book**, **Author**, **Publisher**, **Category**, **Member**, **Loan**, **Reservation**, **Fine**, **Librarian**, and **Branch**. Each entity plays a critical role in maintaining the library's operations and serves as the foundation for different functionalities of the system.

Books are at the heart of the system and are linked to authors, publishers, and categories. Books are categorized based on genres such as fiction, biography, romance, or historical works, which allows for easier browsing and organization. Every book is written by an **Author** and published by a **Publisher**, both of whom can contribute to multiple works available in the library's catalog.

The **Member** entity represents library users who can borrow or reserve books. Members can hold active library memberships that allow them to take loans of books. The **Loan** entity tracks all borrowing activity, including the date a book was borrowed and when it is due to be returned. Members can reserve books that are currently unavailable, and the **Reservation** entity ensures that they are notified when the book becomes available again.

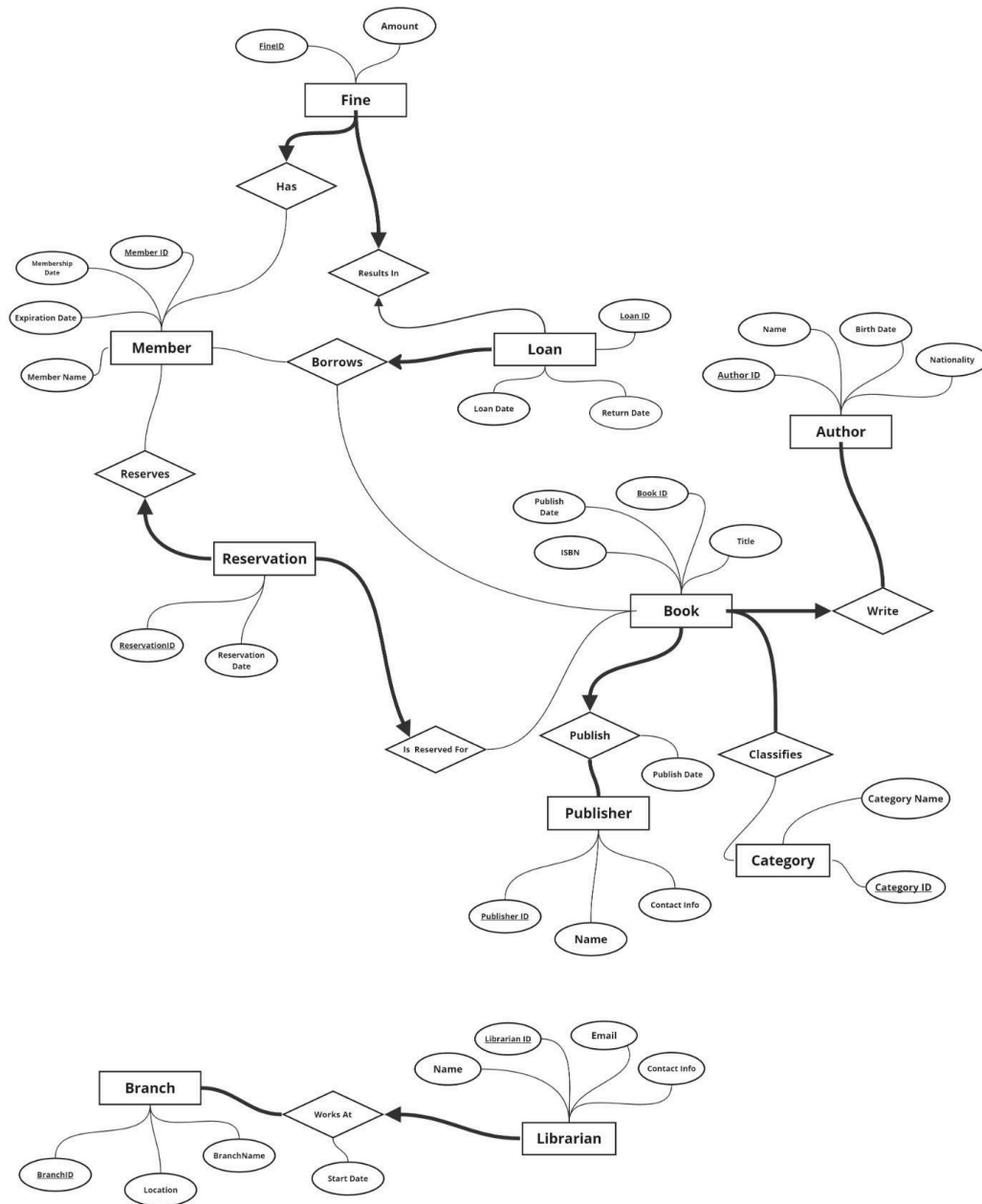
The system also handles the collection of fines for overdue books. Through the **Fine** entity, fines are linked to specific loans, and members are responsible for any penalties they incur. Fines are calculated based on the overdue duration and are tracked until the member settles their dues.

Librarians manage the daily operations of the library, including overseeing loans, reservations, and fines. Librarians are assigned to specific **Branches**, and each branch represents a physical location where books can be stored and borrowed. The system supports multiple branches, allowing members to borrow books from different locations and return them at their convenience.

In this system, relationships between entities are critical for its functionality. For example, members can borrow books (via the **Borrows** relationship), authors can write multiple books (via the **Writes** relationship), and books are classified under multiple categories (via the **Classifies** relationship). The **Digital Library Management System** ensures a smooth flow of information and operations, creating a highly efficient, user-friendly experience for both members and librarians.

The system's ultimate goal is to improve the management and organization of library resources, enhancing the overall experience for members while simplifying the daily tasks of librarians.

ER Diagram



Relational Model

```
CREATE DATABASE Library; -- or your preferred database name
USE Library;
CREATE TABLE Member (
MemberID INT PRIMARY KEY,
MemberName VARCHAR(255),
MembershipDate DATE,
ExpirationDate DATE
);

CREATE TABLE Fine (
FineID INT PRIMARY KEY,
Amount DECIMAL(10, 2)
);

CREATE TABLE Loan (
LoanID INT PRIMARY KEY,
LoanDate DATE,
ReturnDate DATE
);

CREATE TABLE Reservation (
ReservationID INT PRIMARY KEY,
ReservationDate DATE
);

CREATE TABLE Book (
BookID INT PRIMARY KEY,
Title VARCHAR(255),
PublishDate DATE,
ISBN VARCHAR(20)
);

CREATE TABLE Author (
AuthorID INT PRIMARY KEY,
Name VARCHAR(255),
BirthDate DATE,
Nationality VARCHAR(100)
);
```

```
CREATE TABLE Publisher (  
  PublisherID INT PRIMARY KEY,  
  Name VARCHAR(255),  
  ContactInfo VARCHAR(255)  
);  
  
CREATE TABLE Category (  
  CategoryID INT PRIMARY KEY,  
  CategoryName VARCHAR(255)  
);  
  
CREATE TABLE Librarian (  
  LibrarianID INT PRIMARY KEY,  
  Name VARCHAR(255),  
  Email VARCHAR(255),  
  ContactInfo VARCHAR(255)  
);  
  
CREATE TABLE Branch (  
  BranchID INT PRIMARY KEY,  
  BranchName VARCHAR(255),  
  Location VARCHAR(255)  
);  
  
CREATE TABLE WorksAt (  
  LibrarianID INT,  
  BranchID INT,  
  StartDate DATE,  
  PRIMARY KEY (LibrarianID, BranchID),  
  FOREIGN KEY (LibrarianID) REFERENCES Librarian(LibrarianID) ON DELETE CASCADE,  
  FOREIGN KEY (BranchID) REFERENCES Branch(BranchID) ON DELETE RESTRICT  
);  
  
CREATE TABLE Borrows (  
  MemberID INT,  
  BookID INT,  
  LoanID INT,  
  PRIMARY KEY (MemberID, BookID, LoanID),  
  FOREIGN KEY (MemberID) REFERENCES Member(MemberID) ON DELETE RESTRICT,  
  FOREIGN KEY (BookID) REFERENCES Book(BookID) ON DELETE RESTRICT,  
  FOREIGN KEY (LoanID) REFERENCES Loan(LoanID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Publish (  
  PublisherID INT,  
  BookID INT,  
  PublishDate DATE,  
  PRIMARY KEY (PublisherID, BookID),  
  FOREIGN KEY (PublisherID) REFERENCES Publisher(PublisherID) ON DELETE  
  RESTRICT,  
  FOREIGN KEY (BookID) REFERENCES Book(BookID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Has (  
  FineID INT,  
  MemberID INT,  
  PRIMARY KEY (FineID, MemberID),  
  FOREIGN KEY (FineID) REFERENCES Fine(FineID) ON DELETE CASCADE,  
  FOREIGN KEY (MemberID) REFERENCES Member(MemberID) ON DELETE CASCADE  
);
```

```
CREATE TABLE ResultsIn (  
  LoanID INT,  
  FineID INT,  
  PRIMARY KEY (LoanID, FineID),  
  FOREIGN KEY (LoanID) REFERENCES Loan(LoanID) ON DELETE CASCADE,  
  FOREIGN KEY (FineID) REFERENCES Fine(FineID) ON DELETE CASCADE  
);
```

```
CREATE TABLE Reserves (  
  ReservationID INT,  
  MemberID INT,  
  PRIMARY KEY (ReservationID, MemberID),  
  FOREIGN KEY (ReservationID) REFERENCES Reservation(ReservationID) ON DELETE  
  CASCADE,  
  FOREIGN KEY (MemberID) REFERENCES Member(MemberID) ON DELETE CASCADE  
);
```

```
CREATE TABLE IsReservedFor (  
  ReservationID INT,  
  BookID INT,  
  PRIMARY KEY (ReservationID, BookID),  
  FOREIGN KEY (ReservationID) REFERENCES Reservation(ReservationID) ON DELETE  
  CASCADE,  
  FOREIGN KEY (BookID) REFERENCES Book(BookID) ON DELETE CASCADE
```

```
);

CREATE TABLE Writes (
  AuthorID INT,
  BookID INT,
  PRIMARY KEY (AuthorID, BookID),
  FOREIGN KEY (AuthorID) REFERENCES Author(AuthorID) ON DELETE CASCADE,
  FOREIGN KEY (BookID) REFERENCES Book(BookID) ON DELETE CASCADE
);

CREATE TABLE Classifies (
  CategoryID INT,
  BookID INT,
  PRIMARY KEY (CategoryID, BookID),
  FOREIGN KEY (CategoryID) REFERENCES Category(CategoryID) ON DELETE RESTRICT,
  FOREIGN KEY (BookID) REFERENCES Book(BookID) ON DELETE CASCADE
);
```

- ENTITY SETS

```
INSERT INTO Member (MemberID, MemberName, MembershipDate, ExpirationDate)
VALUES
(1, 'Alice Smith', '2023-01-10', '2024-01-10'),
(2, 'Bob Johnson', '2023-02-15', '2024-02-15'),
(3, 'Charlie Brown', '2023-03-20', '2024-03-20'),
(4, 'Daisy Miller', '2023-04-25', '2024-04-25'),
(5, 'Ethan Hunt', '2023-05-30', '2024-05-30'),
(6, 'Fiona Green', '2023-06-05', '2024-06-05'),
(7, 'George White', '2023-07-15', '2024-07-15'),
(8, 'Hannah Blue', '2023-08-10', '2024-08-10'),
(9, 'Ian Black', '2023-09-12', '2024-09-12'),
(10, 'Julia Red', '2023-10-01', '2024-10-01');

INSERT INTO Fine (FineID, Amount) VALUES
(1, 5.00),
(2, 10.00),
(3, 2.50),
(4, 15.00),
(5, 0.00),
```

```
(6, 7.50),  
(7, 12.00),  
(8, 8.75),  
(9, 20.00),  
(10, 1.25);
```

```
INSERT INTO Loan (LoanID, LoanDate, ReturnDate) VALUES
```

```
(1, '2023-01-15', '2023-02-15'),  
(2, '2023-02-20', '2023-03-20'),  
(3, '2023-03-10', '2023-04-10'),  
(4, '2023-04-15', '2023-05-15'),  
(5, '2023-05-01', '2023-05-30'),  
(6, '2023-06-01', '2023-06-30'),  
(7, '2023-07-10', '2023-08-10'),  
(8, '2023-08-20', '2023-09-20'),  
(9, '2023-09-05', '2023-10-05'),  
(10, '2023-10-10', '2023-11-10');
```

```
INSERT INTO Reservation (ReservationID, ReservationDate) VALUES
```

```
(1, '2023-01-01'),  
(2, '2023-01-15'),  
(3, '2023-02-01'),  
(4, '2023-02-20'),  
(5, '2023-03-01'),  
(6, '2023-03-15'),  
(7, '2023-04-01'),  
(8, '2023-04-20'),  
(9, '2023-05-01'),  
(10, '2023-05-15');
```

```
INSERT INTO Book (BookID, Title, PublishDate, ISBN) VALUES
```

```
(1, 'The Great Gatsby', '1925-04-10', '9780743273565'),  
(2, 'To Kill a Mockingbird', '1960-07-11', '9780060935467'),  
(3, '1984', '1949-06-08', '9780451524935'),  
(4, 'Pride and Prejudice', '1813-01-28', '9781503290563'),  
(5, 'Moby-Dick', '1851-10-18', '9781503280786'),  
(6, 'War and Peace', '1869-01-01', '9780199232765'),  
(7, 'The Catcher in the Rye', '1951-07-16', '9780316769488'),  
(8, 'The Hobbit', '1937-09-21', '9780547928227'),  
(9, 'Fahrenheit 451', '1953-10-19', '9781451673319'),  
(10, 'Brave New World', '1932-08-31', '9780060850524');
```

```
INSERT INTO Author (AuthorID, Name, BirthDate, Nationality) VALUES
```



```
(1, 'F. Scott Fitzgerald', '1896-09-24', 'American'),
(2, 'Harper Lee', '1926-04-28', 'American'),
(3, 'George Orwell', '1903-06-25', 'British'),
(4, 'Jane Austen', '1775-12-16', 'British'),
(5, 'Herman Melville', '1819-08-01', 'American'),
(6, 'Leo Tolstoy', '1828-09-09', 'Russian'),
(7, 'J.D. Salinger', '1919-01-01', 'American'),
(8, 'J.R.R. Tolkien', '1892-01-03', 'British'),
(9, 'Ray Bradbury', '1920-08-22', 'American'),
(10, 'Aldous Huxley', '1894-07-26', 'British');
```

```
INSERT INTO Publisher (PublisherID, Name, ContactInfo) VALUES
```

```
(1, 'Scribner', '123 Publishing Rd, NY'),
(2, 'HarperCollins', '456 Book St, NY'),
(3, 'Penguin Random House', '789 Publisher Ave, NY'),
(4, 'Hachette Book Group', '1010 Author Blvd, NY'),
(5, 'Macmillan', '1111 Literature Ln, NY'),
(6, 'Simon & Schuster', '1212 Novel Way, NY'),
(7, 'Oxford University Press', '1313 Academic Blvd, NY'),
(8, 'Cambridge University Press', '1414 Scholarly St, NY'),
(9, 'Bloomsbury', '1515 Fiction Rd, NY'),
(10, 'Scholastic', '1616 Education Dr, NY');
```

```
INSERT INTO Category (CategoryID, CategoryName) VALUES
```

```
(1, 'Fiction'),
(2, 'Non-Fiction'),
(3, 'Science Fiction'),
(4, 'Fantasy'),
(5, 'Mystery'),
(6, 'Romance'),
(7, 'Biography'),
(8, 'History'),
(9, 'Children\'s'),
(10, 'Young Adult');
```

```
INSERT INTO Librarian (LibrarianID, Name, Email, ContactInfo) VALUES
```

```
(1, 'Sarah Connor', 'sarah.connor@example.com', '555-0101'),
(2, 'John Doe', 'john.doe@example.com', '555-0102'),
(3, 'Mary Jane', 'mary.jane@example.com', '555-0103'),
(4, 'Peter Parker', 'peter.parker@example.com', '555-0104'),
(5, 'Bruce Wayne', 'bruce.wayne@example.com', '555-0105'),
(6, 'Clark Kent', 'clark.kent@example.com', '555-0106'),
(7, 'Diana Prince', 'diana.prince@example.com', '555-0107'),
```

```
(8, 'Tony Stark', 'tony.stark@example.com', '555-0108'),  
(9, 'Natasha Romanoff', 'natasha.romanoff@example.com', '555-0109'),  
(10, 'Wade Wilson', 'wade.wilson@example.com', '555-0110');
```

```
INSERT INTO Branch (BranchID, BranchName, Location) VALUES  
(1, 'Main Branch', '123 Main St, City'),  
(2, 'West Branch', '456 West St, City'),  
(3, 'East Branch', '789 East St, City'),  
(4, 'North Branch', '1010 North St, City'),  
(5, 'South Branch', '1111 South St, City'),  
(6, 'Central Branch', '1212 Central Ave, City'),  
(7, 'Downtown Branch', '1313 Downtown Blvd, City'),  
(8, 'Uptown Branch', '1414 Uptown Rd, City'),  
(9, 'Suburban Branch', '1515 Suburban Ln, City'),  
(10, 'Rural Branch', '1616 Rural Rd, City');
```

- RELATIONSHIP SETS

```
INSERT INTO WorksAt (LibrarianID, BranchID, StartDate) VALUES  
(1, 1, '2023-01-01'),  
(2, 1, '2023-01-01'),  
(3, 2, '2023-01-01'),  
(4, 2, '2023-01-01'),  
(5, 3, '2023-01-01'),  
(6, 3, '2023-01-01'),  
(7, 4, '2023-01-01'),  
(8, 5, '2023-01-01'),  
(9, 5, '2023-01-01'),  
(10, 6, '2023-01-01');
```

```
INSERT INTO Borrows (memberid, bookid, loanid) VALUES  
(1, 1, 1),  
(2, 2, 2),  
(3, 3, 3),  
(4, 4, 4),  
(5, 5, 5),  
(1, 6, 6),  
(2, 7, 7),  
(3, 8, 8),  
(4, 9, 9),
```

```
(5, 10, 10);
```

```
INSERT INTO Has (FineID, MemberID) VALUES
```

```
(1, 1),  
(2, 2),  
(3, 3),  
(4, 4),  
(5, 5),  
(6, 1),  
(7, 2),  
(8, 3),  
(9, 4),  
(10, 5);
```

```
INSERT INTO ResultsIn (LoanID, FineID) VALUES
```

```
(1, 1),  
(2, 2),  
(3, 3),  
(4, 4),  
(5, 5),  
(6, 6),  
(7, 7),  
(8, 8),  
(9, 9),  
(10, 10);
```

```
INSERT INTO Reserves (ReservationID, MemberID) VALUES
```

```
(1, 1),  
(2, 2),  
(3, 3),  
(4, 4),  
(5, 5),  
(6, 1),  
(7, 2),  
(8, 3),  
(9, 4),  
(10, 5);
```

```
INSERT INTO IsReservedFor (ReservationID, BookID) VALUES
```

```
(1, 1),  
(2, 2),  
(3, 3),  
(4, 4),
```

```
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10);
```

```
INSERT INTO Writes (AuthorID, BookID) VALUES
```

```
(1, 1),
(2, 2),
(3, 3),
(4, 4),
(5, 5),
(1, 6),
(2, 7),
(3, 8),
(4, 9),
(5, 10);
```

```
INSERT INTO Classifies (CategoryID, BookID) VALUES
```

```
(1, 1), -- The Great Gatsby is Fiction
(1, 2), -- To Kill a Mockingbird is Fiction
(2, 3), -- 1984 is Science Fiction
(3, 4), -- Pride and Prejudice is Fiction
(4, 5), -- Moby-Dick is Fiction
(5, 6), -- War and Peace is Historical Fiction
(6, 7), -- The Catcher in the Rye is Fiction
(7, 8), -- The Hobbit is Fantasy
(8, 9), -- Fahrenheit 451 is Dystopian Fiction
(9, 10); -- Brave New World is Dystopian Fiction
```

```
INSERT INTO Publish (PublisherID, BookID, PublishDate) VALUES
```

```
(1, 1, '1925-04-10'), -- The Great Gatsby by Scribner
(2, 2, '1960-07-11'), -- To Kill a Mockingbird by HarperCollins
(3, 3, '1949-06-08'), -- 1984 by Penguin Random House
(4, 4, '1813-01-28'), -- Pride and Prejudice by Hachette Book Group
(5, 5, '1851-10-18'), -- Moby-Dick by Macmillan
(6, 6, '1869-01-01'), -- War and Peace by Oxford University Press
(7, 7, '1951-07-16'), -- The Catcher in the Rye by Simon & Schuster
(8, 8, '1937-09-21'), -- The Hobbit by Bloomsbury
(9, 9, '1953-10-19'), -- Fahrenheit 451 by Simon & Schuster
(10, 10, '1932-08-31'); -- Brave New World by HarperCollins
```