LibVault

Relational Modeling

Version 1

4/6/2025

1. Introduction

1.1 Project Overview

LibVault is a relational database system designed to execute library operations efficiently and in a user-friendly manner. It allows for the organization of loanable items, enforces borrowing policies, tracks user memberships, and generates reports in a manner that is useful for both staff and the clientele.

1.2 Scope

LibVault's Library Management System will cover library operations such as Book & Digital Media Management, Membership Management, Borrowing & Returns, and Reservations & Notifications. It will automate tracking of loans, returns, and overdue items, while also generating reports to support data-driven decision-making. The program will include the creation of an ER model, a relational schema, and an SQL-based database with realistic records. However, the project does not include the development of a front-end user interface or integration with external library systems at this stage.

1.3 Glossary

- **Database Management System (DBMS):** Software used to create and manage databases, ensuring structured data storage and retrieval.
- **ER Model (Entity-Relationship Model):** A diagram representation of entities and their relationships in a database.
- **SQL (Structured Query Language):** A programming language used for managing and querying relational databases.
- **Primary Key:** A unique identifier for a record in a database table.
- **ISBN (International Standard Book Number):** A unique identifier assigned to books and other publications.
- **Role-Based Access Control (RBAC):** A security model that restricts access based on user roles.
- **Overdue Tracking:** The process of monitoring items that have not been returned by their due date.
- **Reservations:** A system that allows patrons to place holds on unavailable items.
- **SSL/TLS (Secure Sockets Layer/Transport Layer Security):** Cryptographic protocols that provide secure communication over a network.
- **SSH (Secure Shell):** A network protocol that allows secure access to remote computers.

2. Relational Schema Modelling

2.1 Relation Directory

Relation	Attributes	Primary Key	Foreign Keys	Notes
Client	MemberID, Name, PhoneNumber, MembershipType, AccountStatus	MemberID	N/A	MembershipType ∈ {Regular, Student, Administrator}
Media	MediaID, Title, Author, ISBN, PublicationYear, Genre, AvailabilityStatus, Type	ISBN	N/A	AvailabilityStatus ∈ {Available, On Hold, Reserved}
Reservation	ReservationID, ReservationDate, ExpirationDate, MemberID	ReservationID	MemberID → Client(MemberID)	Up to 10 reservations per client
Report	ReportID, OverdueFee, MemberID	ReportID	MemberID → Client(MemberID)	Weak entity, linked to Client
Purchases	PurchaseID, MemberID, ISBN	PurchaseID	MemberID → Client, ISBN → Media	No explicit limit to amount with regards to Client
Makes (Client→Reservation)	(handled via Reservation.MemberID)	N/A	N/A	Relationship handled in Reservation
Reserves (Reservation→Media)	ReservationID, ISBN	(ReservationID, ISBN)	ReservationID → Reservation, ISBN → Media	Many-to-many
Generates (Client→Report)	(handled via Report.MemberID)	N/A	N/A	Handled in Report
Analyzes (Report→Reservation)	ReportID, ReservationID	(ReportID, ReservationID)	ReportID → Report, ReservationID → Reservation	Many-to-many

2.2 Relations

1. Client:

```
Client(
   MemberID CHAR(10) PRIMARY KEY,
   Name VARCHAR(100),
   PhoneNumber VARCHAR(15),
   MembershipType CHAR(15), -- ENUM: Regular, Student, Administrator
   AccountStatus CHAR(10)   -- ENUM: Active, Suspended
)
```

2. Media:

```
Media(
   ISBN CHAR(13) PRIMARY KEY,
   MediaID CHAR(10),
   Title VARCHAR(100),
   Author VARCHAR(100),
   PublicationYear CHAR(4),
   Genre VARCHAR(50),          -- ENUM: Fiction, Non-fiction, Drama, etc.
   AvailabilityStatus CHAR(15),-- ENUM: Available, On Hold, Reserved
   Type VARCHAR(50)
)
```

3. Reservation:

```
Reservation(
   ReservationID CHAR(10) PRIMARY KEY,
   ReservationDate DATE,
   ExpirationDate DATE,
   MemberID CHAR(10),
   FOREIGN KEY (MemberID) REFERENCES Client(MemberID)
)
```

4. Report:

```
Report(
   ReportID CHAR(10) PRIMARY KEY,
   OverdueFee FLOAT(5,2),
   MemberID CHAR(10),
   FOREIGN KEY (MemberID) REFERENCES Client(MemberID)
)
```

5. Reserves:

```
Reserves(
    ReservationID CHAR(10),
    ISBN CHAR(13),
    PRIMARY KEY (ReservationID, ISBN),
    FOREIGN KEY (ReservationID) REFERENCES Reservation(ReservationID),
    FOREIGN KEY (ISBN) REFERENCES Media(ISBN)
)
```

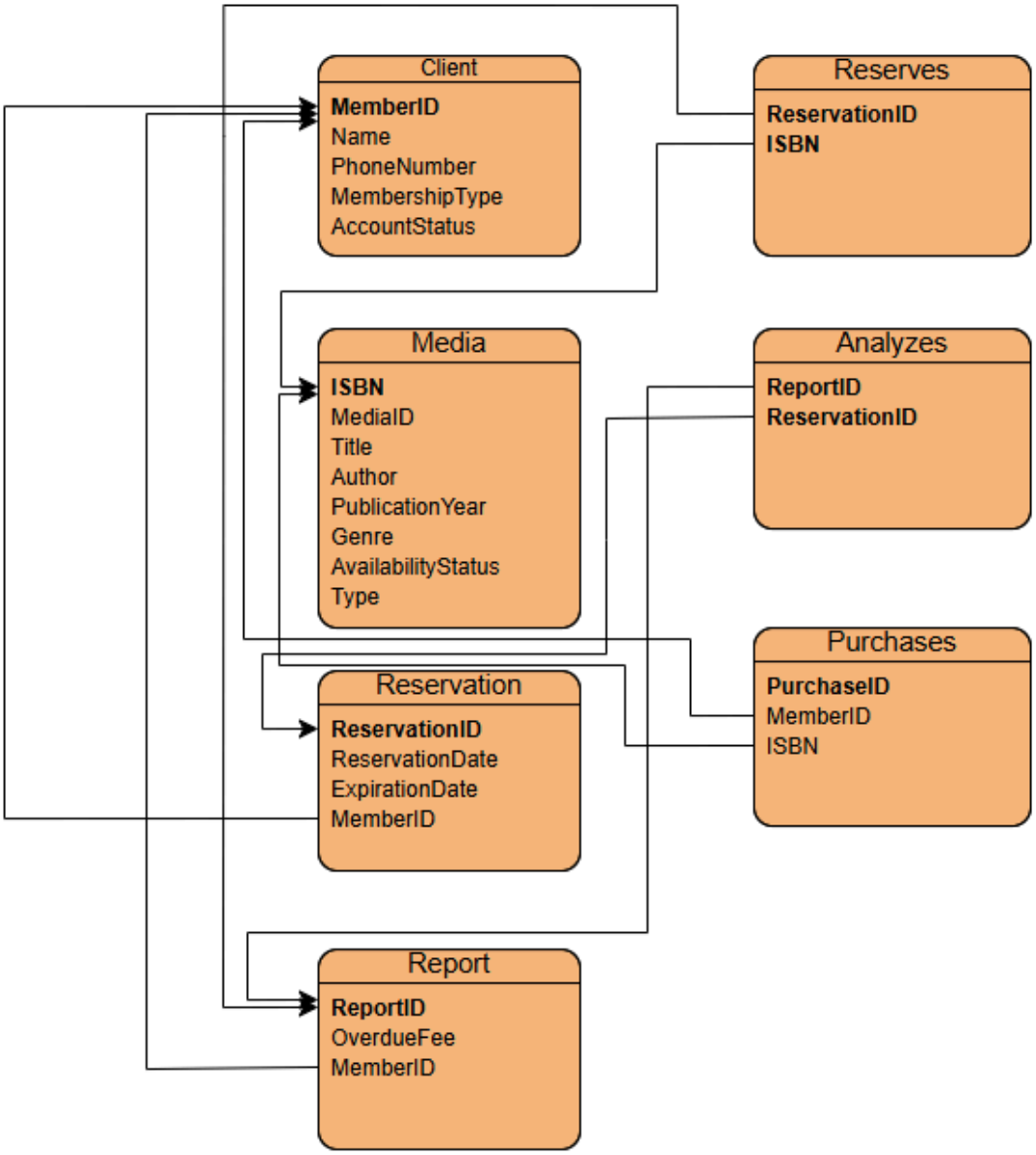
6. Analyzes:

```
Analyzes(
    ReportID CHAR(10),
    ReservationID CHAR(10),
    PRIMARY KEY (ReportID, ReservationID),
    FOREIGN KEY (ReportID) REFERENCES Report(ReportID),
    FOREIGN KEY (ReservationID) REFERENCES Reservation(ReservationID)
)
```

7. Purchases:

```
Purchases(
    PurchaseID CHAR(10) PRIMARY KEY,
    MemberID CHAR(10),
    ISBN CHAR(13),
    FOREIGN KEY (MemberID) REFERENCES Client(MemberID),
    FOREIGN KEY (ISBN) REFERENCES Media(ISBN)
)
```

3. Relational Schema Diagram



Bold Text for Primary Key

4. Data Dictionary

Relation	Attribute	Data Type	Domain / Description
Client	MemberID	CHAR(10)	Unique client identifier
	Name	VARCHAR(100)	Full name
	PhoneNumber	VARCHAR(15)	Valid phone number
	MembershipType	CHAR(15)	{'Regular', 'Student', 'Administrator'}
	AccountStatus	CHAR(10)	{'Active', 'Suspended'}
Media	ISBN	CHAR(13)	Unique book/media identifier
	MediaID	CHAR(10)	Local system ID (optional)
	Title	VARCHAR(100)	Title of the media item
	Author	VARCHAR(100)	Name of the author
	PublicationYear	CHAR(4)	Year of publication (e.g., '2021')
	Genre	VARCHAR(50)	{'Fiction', 'Non-fiction', 'Drama', ...}
	AvailabilityStatus	CHAR(15)	{'Available', 'On Hold', 'Reserved'}
	Type	VARCHAR(50)	Media type (e.g., Book, DVD, eBook)
Reservation	ReservationID	CHAR(10)	Unique ID for reservation
	ReservationDate	DATE	Start date of reservation

	ExpirationDate	DATE	End date of reservation
	MemberID	CHAR(10)	Linked client ID
Report	ReportID	CHAR(10)	Unique report ID
	OverdueFee	FLOAT(5,2)	Total fee for overdue items
	MemberID	CHAR(10)	Client who generated the report
Reserves	ReservationID	CHAR(10)	FK to Reservation
	ISBN	CHAR(13)	FK to Media
Analyzes	ReportID	CHAR(10)	FK to Report
	ReservationID	CHAR(10)	FK to Reservation
Purchases	PurchaseID	CHAR(10)	Unique purchase ID
	MemberID	CHAR(10)	Client who made the purchase
	ISBN	CHAR(13)	Purchased media ID