Database Project: ER Modeling Requirements Document

**Part 1: Introduction**

**Project Overview**

The purpose of this project is to design and implement a database that organizes and manages the key information identified in the requirements phase. The database will store entities, their attributes, and the relationships between them, ensuring that data can be accessed, updated, and maintained efficiently. The primary function of the system is to provide a structured data model that supports accurate storage and retrieval of information for the intended use cases.

**Scope**

The scope of this project is limited to modeling the core entities, attributes, and relationships required to represent the data needs outlined in the requirements document. The database will not include features outside of these identified requirements, such as advanced analytics, user interface design, or external system integration. The focus is on delivering a complete ER model that captures all necessary data elements, enforces integrity through constraints, and provides a clear blueprint for future implementation.

**Glossary**

* Entity – An object or concept that stores data in the database.
* Attribute – A property or characteristic of an entity.
* Relationship – A connection between two or more entities.
* Cardinality – The numerical mapping that defines the relationships between entities
* Primary Key (PK) – A unique identifier for each record in an entity.
* ER Model – A diagrammatic representation of entities, attributes, and relationships.
* Supertype - generic entity that contains the common attributes shared by multiple related entities.
* Subtype - specialized entity that inherits the attributes but also has its own unique attributes.
* Transaction - A single logical unit of work that consists of one or more operations
* Foreign Key (FK) - Database is a field (or a set of fields) in one table that uniquely identifies a row in another table.
* Candidate Key - a database is a minimal set of attributes that can uniquely identify a row in a relation.

**Part 2: Identify ER Modeling Components**

**Identify Entities**

List of major entities to be included in the database:

Membership Type

Client

Item (supertype for all loanable materials)

Book (subtype of Item)

Digital Media (subtype of Item)

Magazine (subtype of Item)

Loan (borrowing transaction)

Reservation (hold/queue for items on loan)

Payment (records fee payments)

Notification (due soon, overdue, reservation available.)

**Define Attributes**

Each entity will have a set of attributes with defined data types or constraints. For instance:

\*\*MembershipType\*\*

- MembershipTypeID (PK, INT)

- Name (VARCHAR(50)) — regular, students, senior citizens, etc.

- MaxConcurrentLoans (INT, ≥ 0)

- LoanDurationDays (INT, ≥ 1)

- DailyLateFee (DECIMAL(6,2), ≥ 0)

- ReservationLimit (INT, ≥ 0)

\*\*Client\*\*

- ClientID (PK, INT)

- FullName (VARCHAR(120))

- Email (VARCHAR(120), UNIQUE)

- Phone (VARCHAR(25))

- Address (VARCHAR(255))

- JoinDate (DATE)

- AccountStatus (ENUM('Active','Suspended','Closed'))

- MembershipTypeID (FK → MembershipType)

\*\*Item (supertype)\*\*

- ItemID (PK, INT)

- Title (VARCHAR(255))

- Genre (VARCHAR(60))

- PublicationYear (INT)

- AvailabilityStatus (ENUM('Available','OnLoan','Reserved','Lost','Damaged'))

- Price (DECIMAL(6,2))

- StockQuantity (INT, ≥ 0)

- ItemType (ENUM('Book','DigitalMedia','Magazine'))

\*\*Book (subtype of Item)\*\*

- ItemID (PK/FK → Item)

- ISBN (VARCHAR(17), UNIQUE)

- Publisher (VARCHAR(120))

\*\*DigitalMedia (subtype)\*\*

- ItemID (PK/FK → Item)

- MediaType (VARCHAR(40))

- Publisher (VARCHAR(120))

\*\*Magazine (subtype)\*\*

- ItemID (PK/FK → Item)

- IssueDate (DATE)

- Publisher (VARCHAR(120))

\*\*Loan\*\*

- LoanID (PK, INT)

- ClientID (FK → Client)

- ItemID (FK → Item)

- CheckedOutAt (DATETIME)

- DueAt (DATETIME)

- ReturnedAt (DATETIME)

- LateFeeCharged (DECIMAL(8,2), ≥ 0, default 0)

- Status (ENUM('Open','Closed','Overdue'))

\*\*Reservation\*\*

- ReservationID (PK, INT)

- ClientID (FK → Client)

- ItemID (FK → Item)

- PlacedAt (DATETIME)

- FulfilledAt (DATETIME, NULLABLE)

- Status (ENUM('Active','Cancelled','Fulfilled','Expired'))

- QueuePosition (INT, ≥ 1)

\*\*Payment\*\*

- PaymentID (PK, INT)

- ClientID (FK → Client)

- Amount (DECIMAL(8,2), > 0)

- PaidAt (DATETIME)

- Method (ENUM('Cash','Card','Online'))

\*\*Notification\*\*

- NotificationID (PK, INT)

- ClientID (FK → Client)

- Type (ENUM('DueSoon','Overdue','ReservationAvailable'))

- Message (TEXT)

**Define Relationships**

The relationships between entities will be defined along with their multiplicities and constraints. For example, ExampleEntity1 may have a one-to-many relationship with ExampleEntity2.

Each Client belongs to one MembershipType, which determines their borrowing and reservation limits.

A Client can borrow multiple items through Loan, while each Item may appear in many loans.

A Client can also place multiple Reservations for items currently checked out.

Each Payment record corresponds to one Client and tracks any fees paid.

Notifications are linked to clients to inform them about due dates, overdue items, or available reservations.

MembershipType (1) ↔ (0..\*) Client

Client (1) ↔ (0..\*) Loan

Item (1) ↔ (0..\*) Loan

Client (1) ↔ (0..\*) Reservation

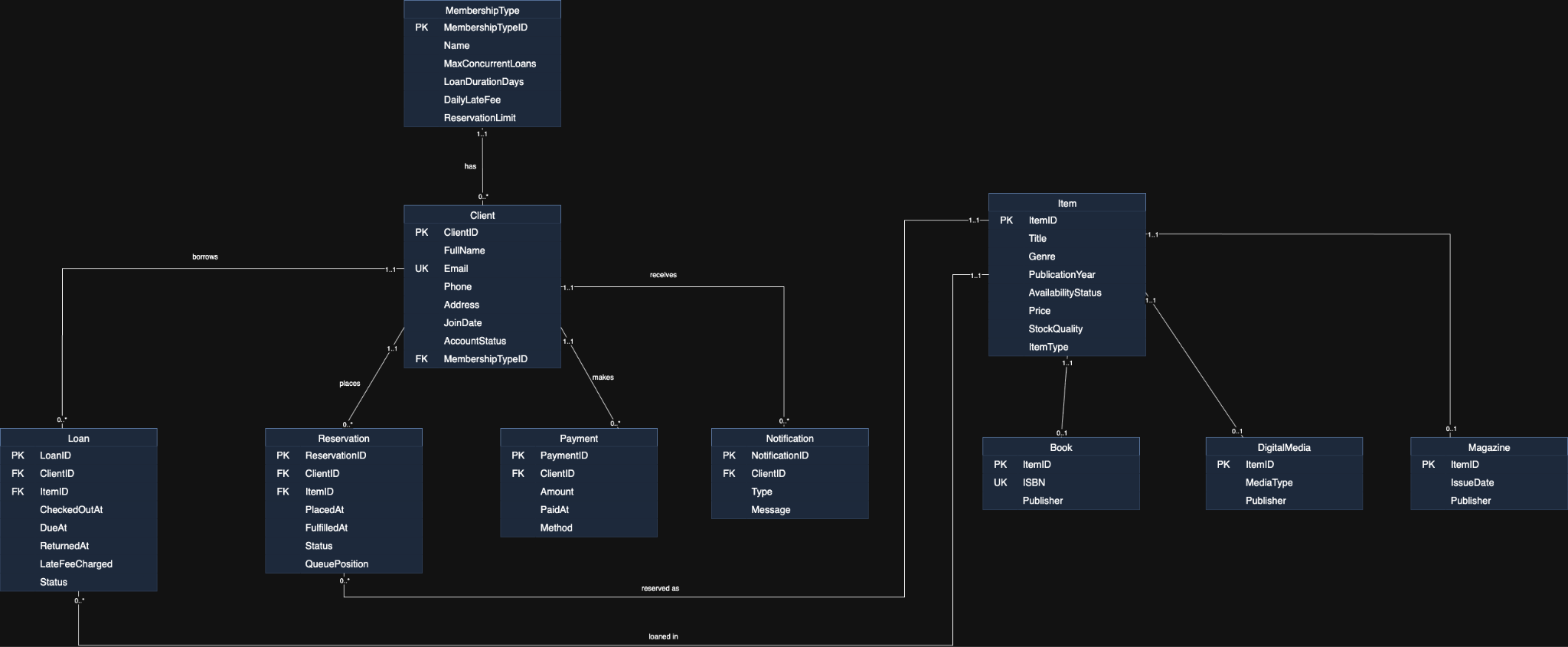
Item (1) ↔ (0..\*) Reservation

Client (1) ↔ (0..\*) Payment

Client (1) ↔ (0..\*) Notification

**Part 3: Create the ER Model**

An ER diagram will be developed using tools such as Draw.io, Lucidchart, or Visual Paradigm. The model will include:  
 • All identified entities and attributes  
 • Primary keys for each entity  
 • Relationships with cardinalities in min..max format  
 • Additional constraints and assumptions



**Part 4: Appendices**

This section provides additional notes and justifications for design choices. It explains why certain entities and attributes were selected, details assumptions made, and discusses how relationships were determined.

**Third Project Meeting Notes**

Date: October 4, 2025

Time: 8:00 PM – 9:00 PM

Location: Zoom Meeting

Objective: Allocate task for Part 3

Team Members Present: Jack, Jamie, K, Aidan, Maxwell

Task Completion Confirmation:

K Li: Completed Part 2 – Identify ER Modeling Components, defining main entities, attributes, and relationships for the ER model.

Aidan Lowry - Constructed the document, and completed part 1 of the forum.

Jack Gerety: Reviewed and revised the document

Jamie King: Completed Part 3 - Create an ER Model

Maxwell Phachanla: Reviewed, revised, and uploaded the document to GitHub

Brainstorming Session:

Team responsibilities:

Each member is responsible for one part of the Requirements Document.

Ensure drafts are uploaded to GitHub before the next meeting.

Review each other’s sections for consistency and clarity.

Tasks Allocated:

- [Aidan Lowry]: Responsible for Part 1 (Overview, Scope, Glossary, and layout of paper)

- [Jamie King]: Responsible for Part 3 (Create an ER model)

- [K Li ]: Responsible for Part 2 (ER Modeling Components)

- [Jack Gerety]: Responsible for reviewing and revising Part 3 of the project

- [Maxwell Phachanla]: Responsible for reviewing and revising Part 3 of the project

Follow-Up Actions:  
 Next meeting scheduled for **10/18/25** at **8:00PM**