1. Introduction

1.1. Project Overview

The logical relational model serves as the foundational blueprint for any well-designed database system, translating conceptual ideas into a structured framework that guides implementation. By defining entities, attributes, and their relationships, this model hopes to ensure data integrity while minimizing redundancy through some normalization principles. For the Library Management System, this logical model represents the critical intermediate step between abstract requirements and physical database creation, mapping out how user profiles, library inventory, transactions, and administrative functions will interact. An optimized database schema helps facilitate efficient data retrieval, enforces business rules through constraints, ideally accommodates future scalability, and serves as a communication tool among stakeholders.

The purpose of this document is to establish the core entities, their attributes, and relationships that will drive the LMS. The model ensures that users, transactions, reservations, library items, fees, notifications, and administrative roles interact seamlessly while maintaining data integrity and performance efficiency.

By defining an optimized database schema with normalization techniques, we can avoid fragmentation, inconsistency, and performance issues which can crop up later as the system is implemented and used in practice.

1.2. Scope

1.2.1. Includes:

- User Management - Structuring user profiles with attributes like membership type, borrowing limits, and account status.
- Library Inventory - Defining books, magazines, and digital media as entities with unique attributes and availability tracking.
- Transactions & Reservations – Establishing how users borrow and return items, with rules for due dates and overdue handlin
- Fees & Fines – Managing late fees, linking transactions to fines, and ensuring overdue charges are properly recorded.
- Notifications System – Implementing a mechanism to inform users about due dates, overdue fines, and reservation statuses
- Administrative Roles – Structuring admin permissions and roles to control system access and manage user activities.

- Reviews & Ratings – Allowing users to provide feedback on library items, supporting engagement and content curation.

1.2.2. Does not Include:

- E-book Integration – The model does not account for digital borrowing through third-party platforms.
- Barcode/RFID Tracking – No physical scanning of books or automated check-in/out using external devices.
- AI-Based Recommendations – Does not include intelligent book suggestions beyond basic user activity tracking.
- Third-Party Payment Processing – The model tracks fees but does not include direct payment integrations (e.g., PayPal, Stripe).

1.3. Glossary

1.3.1. Entity-Relationship Model (ER Model) – A structured representation of database entities and their relationships.

1.3.2. Entity – A fundamental object in the database, such as Users, LibraryItems, or Transactions.

1.3.3. Attribute – A property describing an entity, such as `Title` for books or `Borrow_Date` for transactions.

1.3.4. Primary Key (PK) – A unique identifier for an entity, ensuring each record is distinct.

1.3.5. Foreign Key (FK) – A reference to a Primary Key from another entity, establishing relationships.

1.3.6. One-to-Many (1:M) Relationship – A relationship where one record in an entity relates to multiple records in another (e.g., One User can have multiple Transactions).

1.3.7. One-to-One (1:1) Relationship – A relationship where one record in an entity relates to exactly one record in another.

1.3.8. Weak Entity - An entity that depends on another for identification (e.g., Transactions depend on Users and LibraryItems).

1.3.9. Normalization (3NF/BCNF) - The process of structuring a database to eliminate redundancy and ensure data integrity.

1.3.10. Transactions - Records of borrowing and returning activities in the system.

1.3.11. Reservations - Bookings made by users for unavailable items.

1.3.12. Fees - Penalties imposed for overdue returns.

1.3.13. Notifications - System-generated alerts for users regarding due dates, fees, or reservation updates.

1.3.14. Admin Role (RBAC – Role-Based Access Control) - Defines permissions for administrators (e.g., Librarian, Manager, Clerk).

2. Entities and Attributes

2.1. Users

Strong Entity: Represents individuals using the system, including library patrons and staff.

2.1.1. User_ID: Primary Key, Integer (Auto-increment)

2.1.2. Name: Varchar (NOT Null), Composite Attribute (First Name, Last Name) 2.1.3. Email: Varchar (Unique, NOT Null)

2.1.4. Phone: Varchar (Nullable)

2.1.5. Membership_Type: Enum (“Free”, “Premium”), Determines borrowing privileges

2.1.6. Borrowing_Limit: Integer (NOT Null), Based on Membership_Type

2.1.7. Account_Status: Enum (“Active”, “Suspended”)

2.2. Library Items

Strong Entity: Represents all books, magazines, and digital media in the library

2.2.1. Item_ID: Primary Key, Integer (Auto-increment)

2.2.2. Title: Varchar (NOT Null)

2.2.3. Creator: Varchar (NOT Null), Composite Attribute (First Name, Last Name)

2.2.4. Item_Type: Enum (“Book”, “Magazine”, “Digital Media”)

2.2.5. ISBN: Varchar (Unique, Nullable), Only applicable for books

2.2.6. Genre: Varchar, Multivalued Attribute (item can belong to multiple genres)

2.2.7. Publication_Year: Integer (Nullable)

2.2.8. Availability: Enum (“Available”, “Checked Out”, “Reserved”)

2.3. Transactions

Weak Entity: Tracks borrowing and returning of items, dependent on Users and Library Items

2.3.1. Transaction_ID: Primary Key, Integer (Auto-increment)

2.3.2. User_ID: Foreign Key (Users)

2.3.3. Item_ID: Foreign Key (Library Items)

2.3.4. Borrow_Date: Datetime (NOT Null)

2.3.5. Due_Date: Datetime (NOT Null)

2.3.6. Return_Date: Datetime (Nullable)

2.3.7. Late_Fee: Decimal (5,2), Derived Attribute (Calculated based on due date and return date)

2.4. Reservations

Weak Entity: Tracks items reserved by users, dependent on Users and Library Items

2.4.1. Reservation_ID: Primary Key, Integer (Auto-increment)

2.4.2. User_ID: Foreign Key (Users)

2.4.3. Item_ID: Foreign Key (Library Items)

2.4.4. Reservation_Date: Datetime (NOT Null)

2.4.5. Expiration_Date: Datetime (NOT Null)

2.5. Fees

Weak Entity: Stores fine details, dependent on Users and Transactions

2.5.1. Fee_ID: Primary Key, Integer (Auto-increment)

2.5.2. User_ID: Foreign Key (Users)

2.5.3. Transaction_ID: Foreign Key (Transactions)

2.5.4. Amount: Decimal (5,2) (NOT Null)

2.5.5. Paid_Status: Enum (“Paid”, “Unpaid”)

2.6. Notifications

Weak Entity: Stores system-generated notifications for users

2.6.1. Notification_ID: Primary Key, Integer (Auto-increment)

2.6.2. User_ID: Foreign Key (Users)

2.6.3. Message: Text (NOT Null)

2.6.4. Sent_Date: Datetime (NOT Null)

2.7. Admin Roles

Strong Entity: Represents different staff roles

2.7.1. Role_ID: Primary Key, Integer (Auto-increment)

2.7.2. Role_Name: Enum (“Librarian”, “Manager”, “Clerk”)

2.7.3. Permissions: Text, Multivalued Attribute: A role can have multiple permissions

2.8. Admins

Strong Entity: Represents staff members with assigned roles

2.8.1. Admin_ID: Primary Key, Integer (Auto-increment)

2.8.2. Name: Varchar (NOT Null), Composite Attribute (First Name, Last Name)

2.8.3. Email: Varchar (Unique, NOT Null)

2.8.4. Role_ID: Foreign Key (Admin Roles)

2.9. Reviews

Weak Entity: Represents user feedback on items

2.9.1. Review_ID: Primary Key, Integer (Auto-increment)

2.9.2. User_ID: Foreign Key (Users)

2.9.3. Item_ID: Foreign Key (Library Items)

2.9.4. Rating: Integer (Check - 1 to 5)

2.9.5. Review_Text: Text (Nullable)

2.9.6. Review_Date: Datetime (NOT Null)

3. Relationships

3.1. Users → Transactions

- 1 : M (One to Many)
- Total Participation: Every Transaction must have a user

3.2. Users → Reservations

- 1 : M (One to Many)
- Total Participation: Every reservation must have a user

3.3. Users → Fees

- 1 : M (One to Many)
- Partial Participation: Only users with overdue books will have fees

3.4. Users → Reviews

- 1 : M (One to Many)
- Partial Participation: Only users who leave reviews are involved

3.5. Users → Notifications

- 1 : M (One to Many)
    - Partial Participation: Only users who receive notifications are included

3.6. Users → Admins

- 1 : 1 (One to Many)
    - Partial Participation: Users can be admins if they have an admin role

3.7. Library Items → Transactions

- 1 : M (One to Many)
- Total Participation: Every transaction must be linked to an item

3.8. Library Items → Reservations

- 1 : M (One to Many)
- Partial Participation: Not all items are reserved

3.9. Library Items → Reviews

- 1 : M (One to Many)
- Partial Participation: Only reviewed items are included

3.10. Library Items → Admins

- M : N (Many to Many)
- Partial Participation: Admins manage library inventory by adding/updating/removing items

3.11. Transactions → Fees

- 1 : 1 (One to One)
- Partial Participation: Only transactions with overdue items incur fees

3.12. Transactions → Admins

- M : 1 (Many to One)
- Partial Participation: Admins process checkouts and returns

3.13. Reservations → Admins

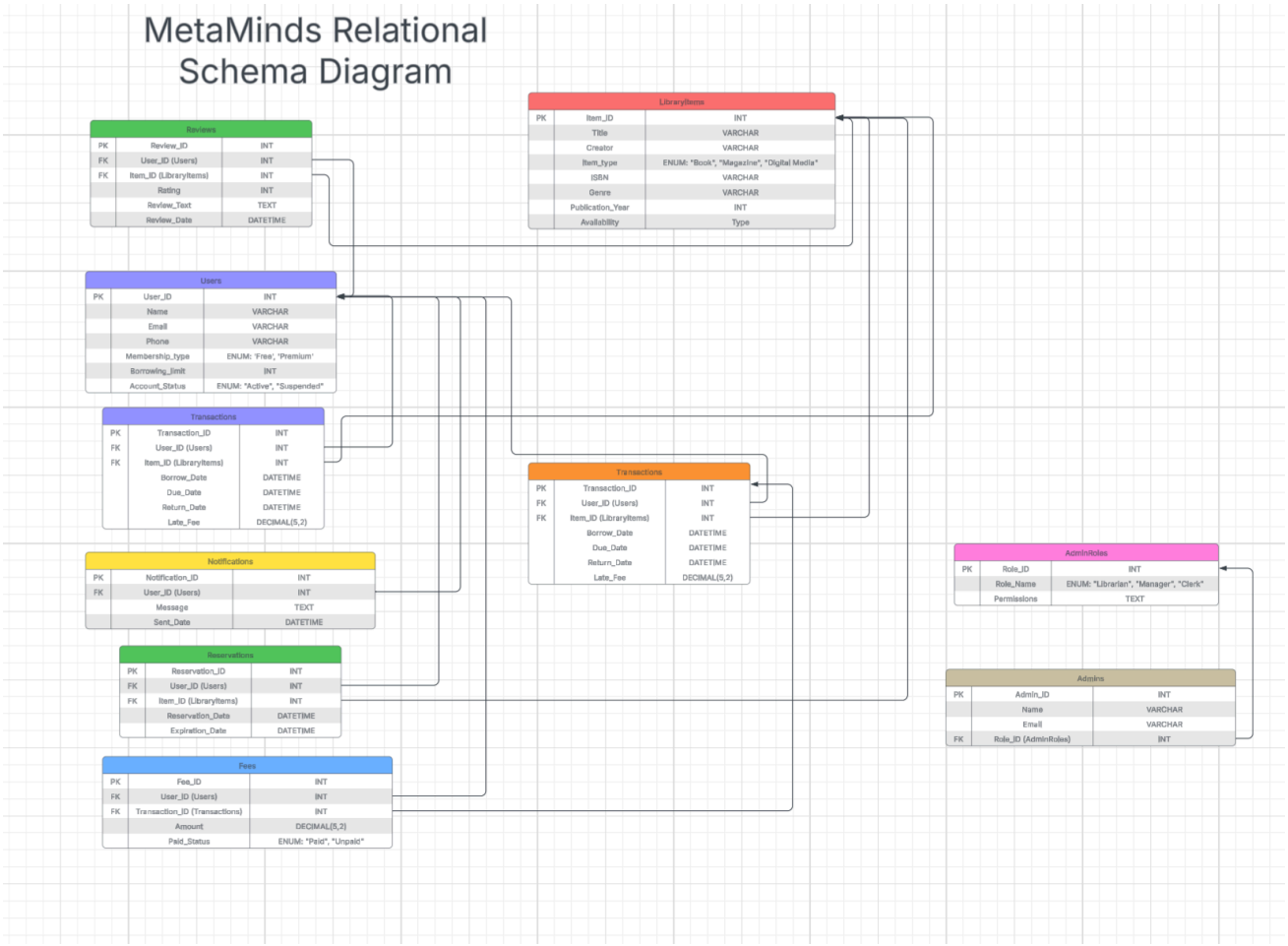
- M : 1 (Many to One)
- Partial Participation: Admins manage reservations, including approvals and cancellations

3.14. Fees → Admins

- M : 1 (Many to One)
    - Partial Participation: Admins oversee fee adjustments and payments

3.15. Admin Roles → Admins

- 1 : M (One to Many)
- Total Participation: Every admin must have a role


MetaMinds Relational Schema Diagram
Reviews
PK Review_ID INT
FK User_ID (Users) INT
FK Item_ID (LibraryItems) INT
Rating INT
Review_Text TEXT
Review_Date DATETIME
LibraryItems
PK Item_ID INT
Title VARCHAR
Creator VARCHAR
Item_type ENUM: "Book", "Magazine", "Digital Media"
ISBN VARCHAR
Genre VARCHAR
Publication_Year INT
Availability Type
Users
PK User_ID INT
Name VARCHAR
Email VARCHAR
Phone VARCHAR
Membership_type ENUM: 'Free', 'Premium'
Borrowing_limit INT
Account_Status ENUM: "Active", "Suspended"
Transactions
PK Transaction_ID INT
FK User_ID (Users) INT
FK Item_ID (LibraryItems) INT
Borrow_Date DATETIME
Due_Date DATETIME
Return_Date DATETIME
Late_Fee DECIMAL(5,2)
Transactions
PK Transaction_ID INT
FK User_ID (Users) INT
FK Item_ID (LibraryItems) INT
Borrow_Date DATETIME
Due_Date DATETIME
Return_Date DATETIME
Late_Fee DECIMAL(5,2)
AdminRoles
PK Role_ID INT
Role_Name ENUM: "Librarian", "Manager", "Clerk"
Permissions TEXT
Notifications
PK Notification_ID INT
FK User_ID (Users) INT
Message TEXT
Sent_Date DATETIME
Reservations
PK Reservation_ID INT
FK User_ID (Users) INT
FK Item_ID (LibraryItems) INT
Reservation_Date DATETIME
Expiration_Date DATETIME
Admins
PK Admin_ID INT
Name VARCHAR
Email VARCHAR
FK Role_ID (AdminRoles) INT
Fees
PK Fee_ID INT
FK User_ID (Users) INT
FK Transaction_ID (Transactions) INT
Amount DECIMAL(5,2)
Paid_Status ENUM: "Paid", "Unpaid"