**PROJECT PART 4: THE LOGICAL RELATIONAL MODEL**

**1, Introduction**

Project Overview:

The Library Management System (LMS) is designed to facilitate the effective management of library assets, members, staff, and transactions. It allows for cataloging of physical and digital media, managing membership accounts, tracking item borrowings, handling reservations, and enforcing overdue fines. The system streamlines operations such as book borrowing and return, fine calculation, and report generation, enhancing both user experience and administrative efficiency.

Scope:

This LMS focuses on:

Cataloging books, magazines, and digital media.

Managing members and staff.

Processing borrowing and returning transactions.

Managing item reservations.

Handling fines for overdue returns.

Excluded features include e-book lending, external system integration, and recommendation systems.

Glossary:

ISBN: International Standard Book Number.

Member: A registered user who can borrow items.

Transaction: A borrowing or returning activity.

Reservation: A request for an item currently unavailable.

Availability: Boolean indicator of whether an item can be borrowed.

Fine: A penalty for overdue items.

**1. Relational Schema Mapping**

| **Entity/Relationship Set** | **Corresponding Table** |
| --- | --- |
| auth\_user | auth\_user |
| accounts\_user | accounts\_user |
| catalog\_item | catalog\_item |
| reports\_report | reports\_report |
| loans\_reservation | loans\_reservation |
| loans\_borrowingtransaction | loans\_borrowingtransaction |
| reservations\_reservation | reservations\_reservation |
| notifications\_notification | notifications\_notification |
| core\_auditing | core\_auditing |
| core\_systemconfig | core\_systemconfig |
| auth\_group | auth\_group |
| auth\_permission | auth\_permission |
| auth\_user\_groups | auth\_user\_groups |
| auth\_user\_user\_permissions | auth\_user\_user\_permissions |
| auth\_group\_permissions | auth\_group\_permissions |
| accounts\_user\_user\_permissions | accounts\_user\_user\_permissions |
| accounts\_user\_groups | accounts\_user\_groups |
| django\_content\_type | django\_content\_type |

**2. Attributes and Domains (Data Dictionary)**

**Table: auth\_user**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| password | VARCHAR(128) | Hashed password |
| last\_login | TIMESTAMP | Last login time |
| is\_superuser | BOOLEAN | Is user admin |
| first\_name | VARCHAR(150) | First name |
| last\_name | VARCHAR(150) | Last name |
| email | VARCHAR(254) | Email address |
| is\_staff | BOOLEAN | Staff status |
| is\_active | BOOLEAN | Active user status |
| date\_joined | TIMESTAMP | Account creation timestamp |

**Table: accounts\_user**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| user\_id | INTEGER (FK) | Foreign key to auth\_user.id |
| profile\_pic | VARCHAR | URL/path to profile picture |
| bio | TEXT | User biography |

**Table: catalog\_item**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| item\_type | VARCHAR(50) | Type of item |
| isbn | VARCHAR(20) | ISBN number |
| issue\_number | INTEGER | Issue number |
| publication\_date | DATE | Date of publication |
| title | VARCHAR(200) | Title |
| publication\_year | INTEGER | Year published |
| genre | VARCHAR(50) | Genre |
| creator | VARCHAR(100) | Creator/Author |
| item\_format | VARCHAR(50) | Format (e.g. book, DVD) |
| availability | BOOLEAN | Availability status |

**Table: reports\_report**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| created\_by\_id | INTEGER (FK) | Foreign key to auth\_user.id |
| title | VARCHAR(100) | Report title |
| description | TEXT | Detailed report description |
| created\_at | TIMESTAMP | Timestamp of creation |

**Table: loans\_reservation**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| member\_id | INTEGER (FK) | Foreign key to auth\_user.id |
| item\_id | INTEGER (FK) | Foreign key to catalog\_item.id |
| reservation\_id | INTEGER (FK) | Foreign key to another reservation |
| request\_date | DATE | Date of request |
| due\_date | DATE | Due date |
| status | VARCHAR(20) | Reservation status |

**Table: loans\_borrowingtransaction**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| member\_id | INTEGER (FK) | Foreign key to auth\_user.id |
| reservation\_id | INTEGER (FK) | Foreign key to loans\_reservation.id |
| borrow\_date | DATE | Date borrowed |
| return\_date | DATE | Date returned |

**Table: reservations\_reservation**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| content\_type\_id | INTEGER (FK) | Foreign key to django\_content\_type.id |
| object\_id | INTEGER | ID of referenced object |
| timestamp | TIMESTAMP | Timestamp of reservation |

**Table: notifications\_notification**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| recipient\_id | INTEGER (FK) | Foreign key to auth\_user.id |
| message | TEXT | Notification message |
| created\_at | TIMESTAMP | Time sent |
| is\_read | BOOLEAN | Read status |

**Table: core\_auditing**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| user\_id | INTEGER (FK) | Foreign key to auth\_user.id |
| action | VARCHAR(100) | Performed action |
| timestamp | TIMESTAMP | Time of action |

**Table: core\_systemconfig**

| **Attribute** | **Domain** | **Description** |
| --- | --- | --- |
| id | INTEGER (PK) | Unique identifier |
| config\_key | VARCHAR(100) | Config key |
| config\_value | TEXT | Config value |

RELATIONAL SCHEMA DIAGRAM:

A screenshot of a computer

AI-generated content may be incorrect.

ER DIAGRAM:

**A screenshot of a computer

AI-generated content may be incorrect.**

**3. Primary Keys and Foreign Keys**

**Primary Keys**

| **Table** | **Primary Key** |
| --- | --- |
| auth\_user | id |
| accounts\_user | id |
| catalog\_item | id |
| reports\_report | id |
| loans\_reservation | id |
| loans\_borrowingtransaction | id |
| reservations\_reservation | id |
| notifications\_notification | id |
| core\_auditing | id |
| core\_systemconfig | id |
| auth\_group | id |
| auth\_permission | id |
| auth\_user\_groups | id |
| auth\_user\_user\_permissions | id |
| auth\_group\_permissions | id |
| accounts\_user\_user\_permissions | id |
| accounts\_user\_groups | id |
| django\_content\_type | id |

**4. Foreign Keys**

* catalog\_item.member\_id → auth\_user.id
* loans\_reservation.member\_id → auth\_user.id
* loans\_reservation.item\_id → catalog\_item.id
* loans\_reservation.reservation\_id → reservations\_reservation.id
* loans\_borrowingtransaction.member\_id → auth\_user.id
* loans\_borrowingtransaction.reservation\_id → loans\_reservation.id
* reports\_report.created\_by\_id → auth\_user.id
* notifications\_notification.recipient\_id → auth\_user.id
* core\_auditing.user\_id → auth\_user.id
* auth\_user\_groups.user\_id → auth\_user.id
* auth\_user\_groups.group\_id → auth\_group.id
* auth\_user\_user\_permissions.user\_id → auth\_user.id
* auth\_user\_user\_permissions.permission\_id → auth\_permission.id
* auth\_group\_permissions.group\_id → auth\_group.id
* auth\_group\_permissions.permission\_id → auth\_permission.id
* auth\_permission.content\_type\_id → django\_content\_type.id
* reservations\_reservation.content\_type\_id → django\_content\_type.id
* reservations\_reservation.object\_id → (polymorphic reference)

**5. Functional Dependencies (FDs)**

* In auth\_user:  
  id → password, last\_login, is\_superuser, first\_name, last\_name, email, is\_staff, is\_active, date\_joined
* In catalog\_item:  
  id → item\_type, isbn, issue\_number, publication\_date, title, publication\_year, genre, creator, item\_format, availability
* In loans\_reservation:  
  id → request\_date, due\_date, status, item\_id, member\_id, reservation\_id
* In auth\_user\_groups:  
  {user\_id, group\_id} → ∅ (composite key)
* In auth\_permission:  
  id → content\_type\_id, codename, name
* In auth\_group\_permissions:  
  {group\_id, permission\_id} → ∅
* Here's your **Schema Documentation with a Data Dictionary** formatted for easy copying into a Word document using table structure headings. You can paste each section into Word and use the “Insert Table” feature to make it visually clean.

**3. Schema Documentation with a Data Dictionary**

A screenshot of a data

AI-generated content may be incorrect.

A screenshot of a document

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a document

AI-generated content may be incorrect.

A screenshot of a computer

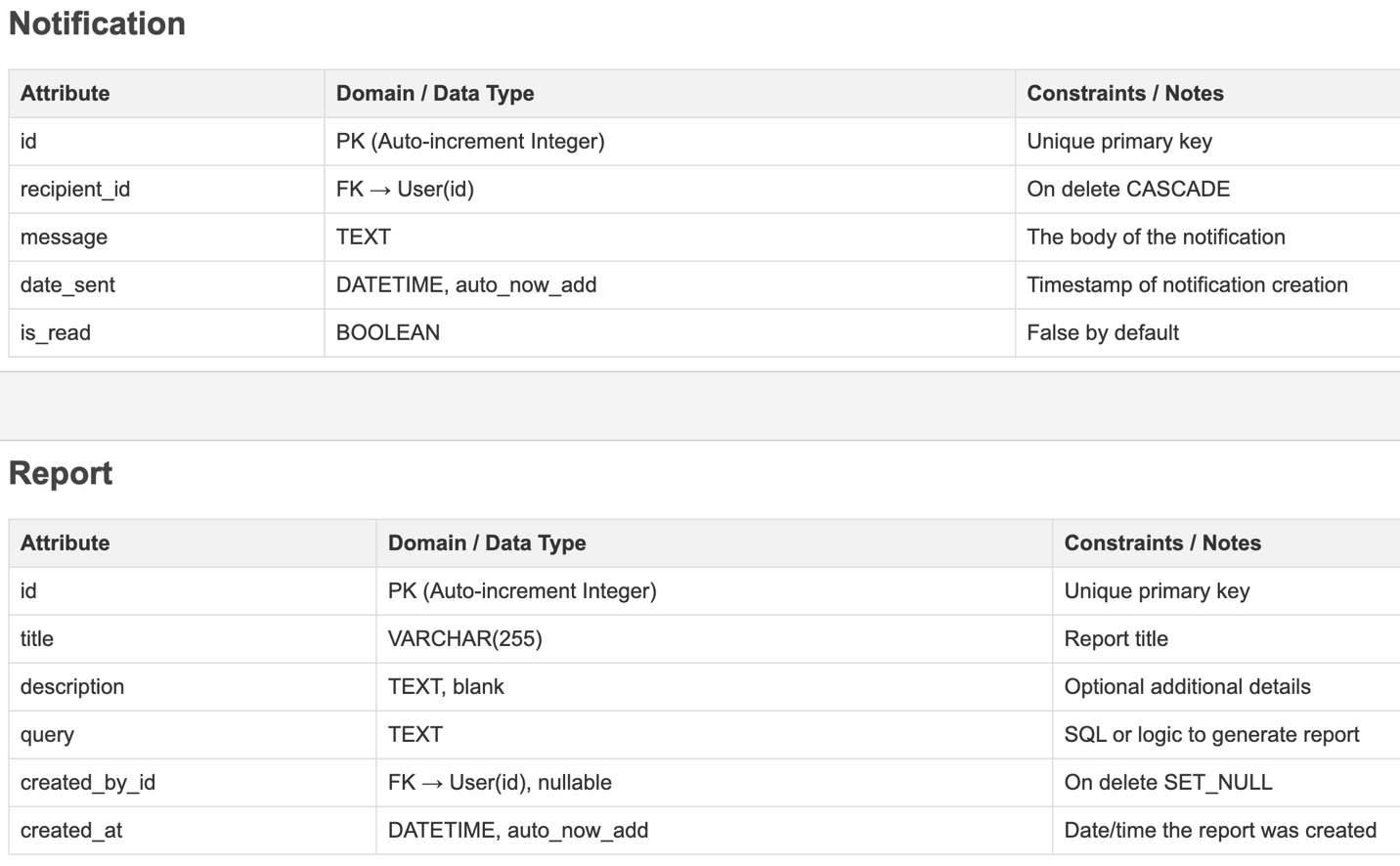
AI-generated content may be incorrect.

SCHEMA:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer code

AI-generated content may be incorrect.

A screenshot of a computer code

AI-generated content may be incorrect.