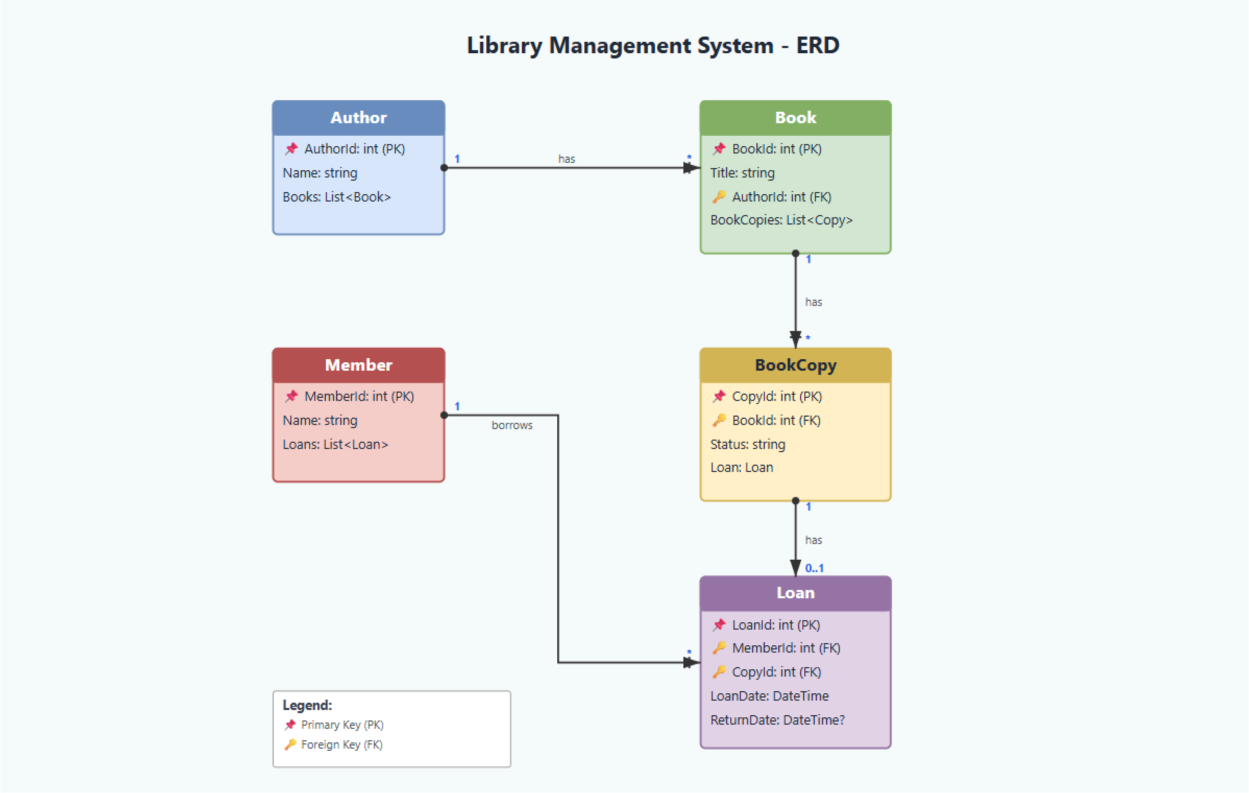
BACKEND - REST API

1. ERD Overview



2. Relationships

Author → Book

Relationship: One-to-Many

Cardinality: 1 Author has many Books

Notes: Author.Books

Book → BookCopy

Relationship: One-to-Many

Cardinality: 1 Book has many BookCopies

BookCopy → Loan

Relationship: One-to-One

Cardinality: 1 BookCopy has 0 or 1 Loan

Member → Loan

Relationship: One-to-Many

Cardinality: 1 Member can have many Loans

Loan → BookCopy

Relationship: One-to-One

Cardinality: Each loan corresponds to exactly one specific BookCopy

3. REST API – Books Resource

Base URL: /api/v1/books

GET /api/v1/books

Retrieve a list of all books.

Response – 200 OK

```
[
  {
    "id": 1,
    "title": "Book Title",
    "description": "Description",
    "authorId": 3
  }
]
```

Response – 404 Not Found

```
{
  "message": "No books found."
}
```

GET /api/v1/books/{id}

Retrieve a specific book by ID.

Response – 200 OK

```
{
  "id": 1,
  "title": "Book Title",
  "description": "Description",
  "authorId": 3
}
```

Response – 404 Not Found

```
{ "message": "Book with id 10 not found." }
```

POST /api/v1/books

Create a new book.

Request Body

```
{
  "title": "New Book",
  "description": "Book description",
  "authorId": 1
}
```

Response – 201 Created

```
{
  "id": 5,
  "title": "New Book",
  "description": "Book description",
  "authorId": 1
}
```

Response – 400 Bad Request

Occurs when validation fails.

PUT /api/v1/books/{id}

Update a book.

Request Body

```
{
  "title": "Updated Title",
  "description": "Updated description",
  "authorId": 1
}
```

Response – 200 OK

```
{
  "id": 1,
  "title": "Updated Title",
  "description": "Updated description",
  "authorId": 1
}
```

Response – 404 Not Found

```
{ "message": "Book with id 10 not found." }
```

DELETE /api/v1/books/{id}

Soft deletes the book by setting all BookCopies to 0.

Response – 200 OK

```
{
  "message": "Book with id 1 has been soft deleted (copies set to 0)."
}
```

Response – 404 Not Found

```
{ "message": "Book with id 1 not found." }
```

4. Book Service

What the Book Service Does

The **BookServiceImpl** handles all business logic for books.
It talks to the database (via EF Core) and is used by the BookController.

It provides 5 main operations:

1. Get all books
2. Get a book by ID
3. Create a new book (or add more copies to an existing one)
4. Update an existing book
5. Soft delete a book (set all copies to 0)

Method Explanations

GetAllAsync()

- Returns a list of all books.
- Includes each book's author.
- Adds up total copies and available copies from BookCopies.

GetByIdAsync(int id)

- Returns a single book with its author and copy details.
- Returns null if no book exists.

CreateAsync(BookCreateDto dto)

This method does **two things**:

If book title already exists:

- Increases the book's TotalCopies
- Increases AvailableCopies
- Does NOT create a new book
- Returns updated book info

If book does NOT exist:

- Checks if the author exists; otherwise creates a new author
- Creates a new book
- Creates a BookCopy record for inventory
- Returns the created book

UpdateAsync(int id, BookUpdateDto dto)

- Updates a book’s title, description, and author.
- Updates copy counts if provided.
- If the book has no copies yet but copy data is included, it creates a new BookCopy.
- Validates input.

SoftDeleteAsync(int id)

- Sets **TotalCopies = 0** and **AvailableCopies = 0** for all the book’s copies.
- Book remains in the database.
- Used to prevent breaking references in Loans.

Private Helper Methods

GetBookByTitleAsync()

Finds a book by title (case-insensitive).

GetBookByIdAsync()

Loads a book with its author and copies.

GetAuthorByNameAsync()

Finds an author by name (case-insensitive).

CreateNewAuthor() / CreateNewBook() / CreateNewBookCopy()

These create new database records.

MapToDto()

Converts database objects into a clean BookResponseDto for API responses.

Validation Rules

When Creating a Book

- Title required, max 200 characters
- Author name required, max 150 characters
- Description max 1000 characters
- TotalCopies must be at least 1

When Updating a Book

- All fields optional
- Title max 200
- Author max 150
- Description max 1000
- Copies cannot be negative