Books App — MySQL Mini Report

Course: Object-Oriented Analysis & Design + Advanced Web Programming
Team: Yoganathan Renaud, Dambron Max, Sghaier Souhaib Date: 08/10/2025

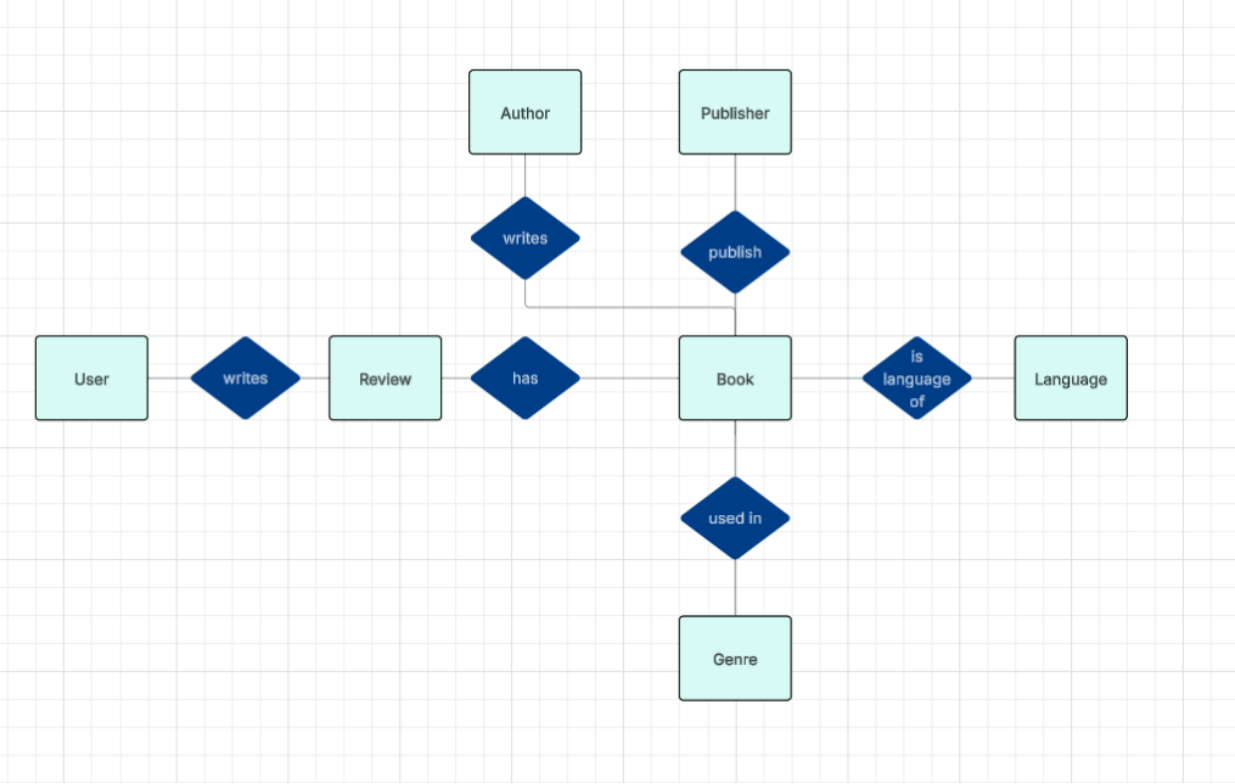
1) Goal

Build a simple, normalized MySQL-backed website for discovering books, browsing by genre, and posting user reviews.
This version adds a more detailed database structure and an entity-relationship model to illustrate our design choices.

2) ER Diagram

The ER diagram shows how the main entities are connected:

- Book is the central entity.
- Each Book is written by one Author, published by one Publisher, and written in one Language.
- A Book can belong to several Genres through the link table Book_Genre.
- Users can post Reviews about different books.
- Relationships are mainly one-to-many, except Book ↔ Genre, which is many-to-many.



3) **Table Structure Description**

The diagram represents the logical structure of the database used for the book application. The goal was to design a simple, normalized model where each entity has a clear and unique purpose.

The **Books** table is the central element of the model.
Each book is written by one **Author** and published by one **Publisher**, which explains the *one-to-many* relationships between these tables — one author or publisher can be linked to several books.
The **language** field was integrated directly inside the *Books* table to simplify the design and avoid an unnecessary extra table.

Users are represented in the **Users** table and can write multiple **Reviews**, each review being related to a single book.
This creates a *one-to-many* relation between *Users* → *Reviews* and *Books* → *Reviews*.

Books can also belong to several **Genres**, and each genre can contain multiple books.
This *many-to-many* relation is managed by the associative table **Book_Genres**, ensuring flexibility when classifying books.