BUILDING A READING PLATFORM BASED ON BOOKMATE

UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Members:

Andrés Felipe Salazar Malagón — 20202020043

Ichel Delgado Morales — 20202020029

Professor:

Carlos Andrés Sierra Virguez.

Universidad Distrital Francisco José de Caldas

Faculty of Engineering

Databases II

Bogotá, 2025

Abstract

This paper presents the design of the relational database for a digital reading platform inspired by Bookmate. The project focuses on structuring the system's data to support core functionalities such as book and audiobook browsing, reviews, user interactions, and subscription management. Functional requirements were defined based on detailed user stories and organized by feature groups. A normalized relational model was developed to support scalable, consistent, and efficient data handling. A preliminary architecture outlines the role of the database in receiving structured input from administrative and user interfaces and providing the necessary outputs to external layers. This work establishes the foundation for the platform's backend and frontend integration in future development stages.

Keywords: Digital reading platforms, relational database design, data modeling, functional requirements, Bookmate, user interaction, subscriptions, content management, database architecture.

Contents

Chapter 1

Introduction

In recent years, digital platforms have transformed the way users access and consume literary content. Among these, Bookmate stands out for offering a personalized reading experience through subscriptions that include books, audiobooks, and social interaction features. This project presents the design and implementation of the database architecture for a platform inspired by Bookmate, emphasizing personalized content delivery, efficient data handling, and scalability.

The main goal is to model and create a robust, normalized relational database that supports essential functionalities such as user management, content cataloging, recommendation mechanisms, and user interaction. This effort is part of a broader system development project that includes interface design and backend architecture. The database acts as the core foundation to ensure consistency, integrity, and performance of the application.

Chapter 2

Background

The global shift toward digital content consumption has accelerated the evolution of reading habits. Platforms like Bookmate exemplify this transition by combining unlimited access to e-books and audiobooks with community-driven features such as user reviews, reading lists, and social follows. Bookmate's success lies not only in offering a large catalog, but in delivering a personalized, engaging user experience across devices. This hybrid model requires an architecture capable of handling diverse media formats, user-generated content, and real-time interactions.

From a technical perspective, supporting such a platform involves designing a backend that can ensure data consistency, quick retrieval times, and adaptability to growth. This includes managing core entities such as users, books, subscriptions, and reviews, each of which has relational dependencies and requires integrity constraints. Relational database theory, particularly normalization up to third normal form (3NF), plays a crucial role in reducing redundancy, enforcing referential integrity, and enabling efficient querying. By adhering to these principles, a platform can avoid issues like data duplication, update anomalies, and performance bottlenecks.

Moreover, the platform must accommodate complex interactions between components: for example, a user following another user, reacting to a review, or receiving book recommendations based on collaborative filtering. These features go beyond simple CRUD operations and necessitate a schema that supports indexing, foreign key enforcement, and scalable joins. The report builds on these foundational concepts to show how a well-structured relational database supports not just storage, but the overall business logic and user satisfaction of a digital reading ecosystem.

Chapter 3

Objectives

The main objective of this report is to design and justify a relational database model that serves as the structural core for a Bookmate-inspired digital reading platform. This database must be able to efficiently manage users, books, audiobooks, subscriptions, social interactions, and recommendations, all under the principles of integrity, scalability, and normalization.

To achieve this, the following specific objectives are proposed:

- Translate the functional requirements of the business (reading, community, monetization, and personalization) into entities, relationships, and constraints typical of a normalized relational model.
- Design an entity-relationship (ER) schema that complies with Third Normal Form (3NF), avoiding redundancies, ensuring data consistency, and facilitating future system extensions.

This report seeks to answer, among others, the following key questions: What data structure efficiently supports reader, subscription, and community flows on a digital platform? How can we ensure that the relationships between users, content, and payments remain intact? How does a well-designed database influence the end-user experience?

Chapter 4

Scope

This report focuses exclusively on the design and justification of the relational database that supports the core functionalities of a digital reading platform modeled after Bookmate. The study includes the analysis of business needs, translation into functional requirements, development of an entity-relationship model, normalization to third normal form (3NF), and the integration of the data model within a conceptual system architecture. The work also covers the documentation and presentation of the model using collaborative tools for modeling and reporting.

The scope includes the modeling of critical entities such as users, books, subscriptions, reviews, reading lists, and social interactions (e.g., follows). It also considers constraints related to data integrity, relational dependencies, and extensibility for future modules like recommendation systems or analytics. Additionally, the report includes assumptions regarding the intended use of the platform (streaming and downloading of books and audiobooks), the monetization strategy (subscription model), and the multi-device user experience.

However, the report excludes the physical implementation of the database (i.e., SQL scripts, indexing strategies, or the selection of a specific database management system). This delimitation ensures a clear focus on the conceptual and logical design of the database that supports the functionality of the reading platform.

Chapter 5

Limitations

As this report constitutes the first theoretical phase of the project, its main limitation lies in the absence of physical implementation. The database model has been developed at a conceptual and logical level only, without accompanying SQL scripts, performance testing, or deployment in a specific database management system (DBMS). Consequently, aspects such as query optimization, index design, and transaction handling remain outside the scope of this stage and may reveal additional challenges once implementation begins.

Another limitation is the lack of real user data. While the entity-relationship model has been designed based on functional requirements and reference platforms like Bookmate, it has not yet been tested with real-world datasets. This could lead to the need for adjustments when validating assumptions about usage patterns, data volume, and edge cases in future iterations.

Finally, the project prioritizes the implementation of the database and its queries, intentionally excluding concepts such as the front-end or back-end of the same to maintain a total focus on the subject of the Database II course, and thus be able to strengthen the concepts seen.

Chapter 6

Results

The Business Model Canvas designed for the platform based on Bookmate reveals a coherent and user-centric approach. It emphasizes delivering unlimited access to a vast catalog of digital and audio books under a subscription model, while also enabling community features like reviews, forums, and collaborative reading lists. This dual emphasis on content and social interaction positions the platform as both a library and a literary social network, aiming to differentiate itself in a saturated market.

One key insight is the alignment between technical decisions and business goals. The use of relational databases for transactional operations (users, subscriptions, payments) ensures data integrity and accountability, while NoSQL components are introduced for fast metadata access and personalization, supporting features such as personalized recommendations and community engagement. This hybrid structure reflects a mature understanding of the technical demands behind Bookmate's scalable infrastructure.

Another relevant outcome is the identification of diverse customer segments, ranging from digital-native readers and audiobook fans to corporate clients and independent authors. Each of these segments demands specific functionality—e.g., bulk licenses for companies, or publishing tools for authors—which in turn require dedicated modules in the database (e.g., UserRole, AuthorProfile, LicenseAgreement). By mapping each component of the canvas to a system feature, the report confirms that the business model is fully supported by the data model, making the Canvas not just a strategic tool but a blueprint for database design.

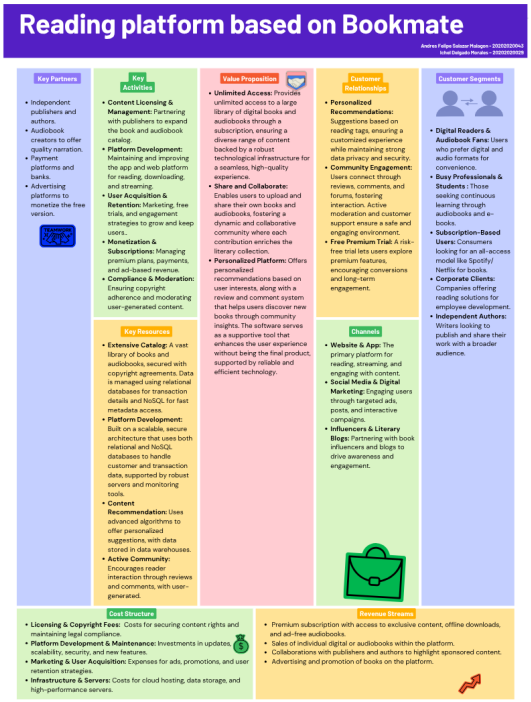

Figure 6.1: Business model for reading platform

A critical outcome of this project was the translation of functional requirements derived from the Business Model Canvas into concrete database entities and relationships. This process ensured that every core functionality expected by users, partners, and the business itself is directly supported by the logical structure of the data model.

This mapping not only ensures functional coverage but also strengthens the model's ability to scale, audit, and extend. In future stages of the project, the traceability table will allow you to verify whether the implemented modules cover all aspects of the business. It also facilitates communication with stakeholders, who can validate that what is in the Canvas is directly and technically reflected in the database.

For example, one of the most important building blocks of the Canvas model is the Value Proposition, which includes unlimited access to books and audiobooks, personalized recommendations, and social features like comments and shared lists. These features became functional requirements such as:

Entity	Description	Key relationship
user	Represents a registered user (reader, creator & administrator).	Has many reviews, favorites, follows, reading_lists, subscriptions, and notifications.
book	Represents a book with its title, author, cover image, etc.	Has many reviews, is in many favorites, and appears in many reading_list_items.
subscription	User subscription information such as plan, dates and payment status.	Belongs to one user Refers to one plan type Has one payment status type
review	Comments and scores made by any user to a book.	Belongs to User. Belongs to Book.
favorite	Relationship between users and books marked as favorites.	Belongs to User. Belongs to Book.
follow	Allows one user to follow another.	One user follows another user.
notification	Represents a notification resulting from any interaction one user has with another user.	Belongs to one user. Refers to a notification_type.
reading_list	Represents a reading list made and personalized by any user.	Created by one user Has many reading_list_item entries

Figure 6.2: Traceability table requirements-entities

- "The user can browse books and audiobooks by title, author, or category."
- "The user can leave a review or rating for a book."
- "Users can follow each other and see their activity."

Each of these requirements involved creating specific entities:

- Book and Audiobook, with attributes such as title, author_id, genre_id, format, etc.
- Review, which connects User and Reserve through a relationship with fields such as rating, comment, and created_at.

The main outcome of this research phase is the design of a normalized relational database model capable of supporting a digital reading platform similar to Bookmate. The model was developed based on 25 functional requirements that cover core functionalities such as user authentication, content management (books and audiobooks), subscriptions, user-generated reviews, and social interactions.

The resulting entity-relationship diagram (ERD) includes key entities such as User, Book, Subscription, Author, Review, Genre, ReadingList, and Follow. Each entity is linked through properly defined relationships and foreign keys to ensure referential integrity. The schema adheres to third normal form (3NF), minimizing data redundancy and ensuring atomicity of attributes.

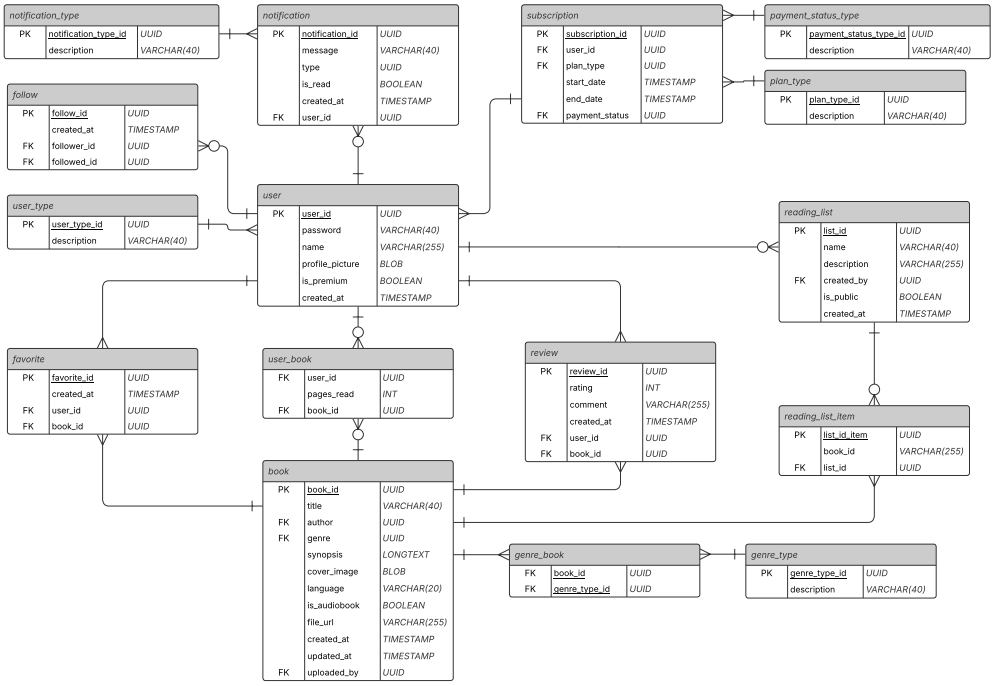

Figure 6.3: Relational model for the reading platform

In addition, a simplified system architecture diagram was created to show how the database interacts with other components of the platform, such as the frontend application, authentication service, and potential analytics or recommendation engines. This conceptual architecture helps illustrate how the data model will support scalability, modularity, and multi-device access.

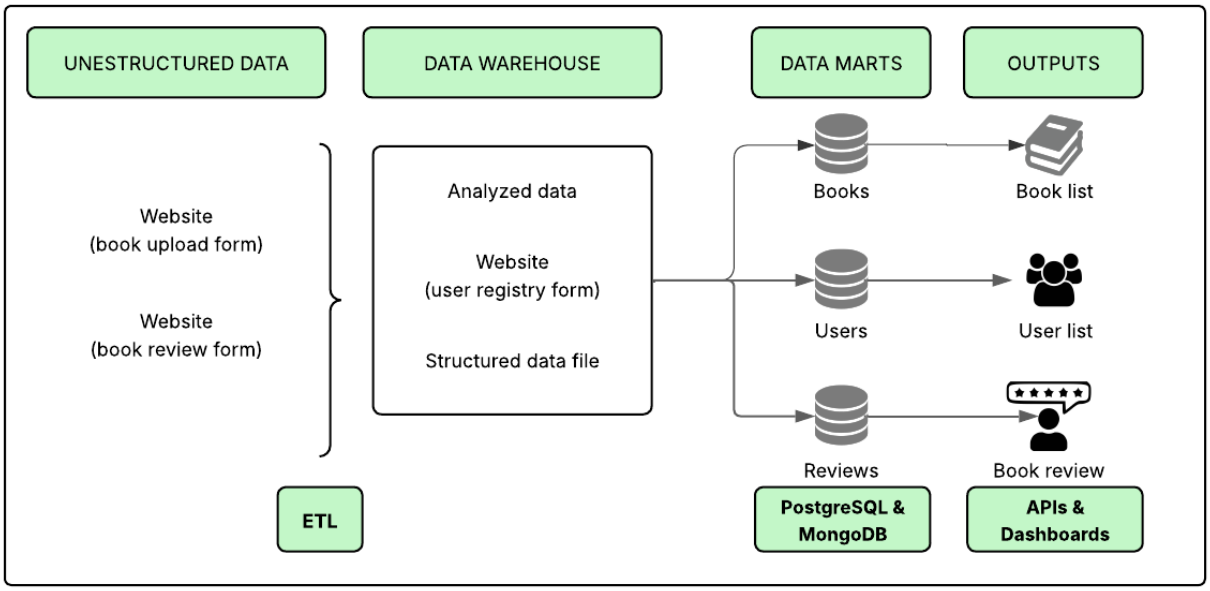

Figure 6.4: Database-Centered Architecture Diagram

Chapter 7

Methodology

A requirements-driven methodology was followed, beginning with the definition of the business model using the Business Model Canvas to identify stakeholders, value propositions, and key functionalities. From there, the team specified functional requirements and user stories to define user interactions and system behavior.

Code	Requirement
FR-	
FR-	
FR-	

Figure 7.1: Requirement table template

Title:	Priority:
User Story:	
Acceptance Criteria:	

Figure 7.2: Use story table template

The project adopts a relational database model to ensure consistency, integrity, and scalability. We used the relational modeling technique, and the diagram was created using crow's foot notation. The central entity is `user`, which supports various roles such as reader, creator, or administrator. Other key entities include `book`, `review`, `subscription`, `favorite`, `follow`, and `reading_list`.

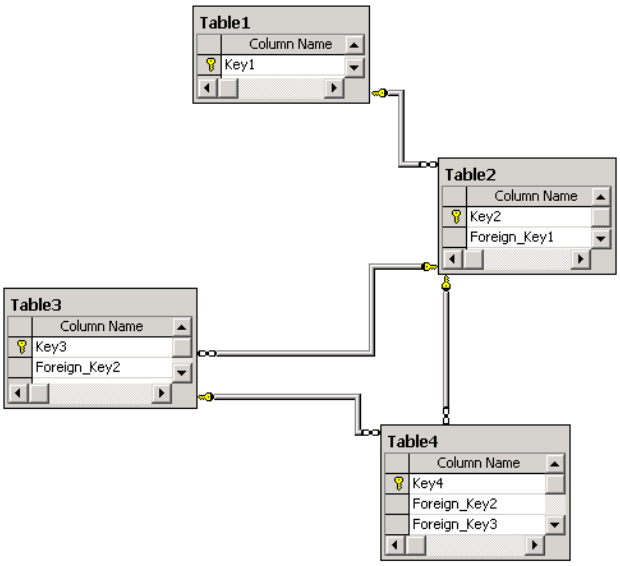

Figure 7.3: Example of a relational model

Design Choices:

- Users can create personalized reading lists composed of multiple books.
- A `review` is a comment and score linked to both a book and a user.
- The `subscription` entity captures plan details, status, and dates.
- `Follow` and `notification` entities enable social interaction.

Chapter 8

Conclusions

The project successfully defined and structured the database layer for a social reading platform inspired by Bookmate. The system requirements were clearly established through user stories, and the resulting relational model was designed to support key features such as subscriptions, user-generated content, and personalized reading lists.

From a database perspective, the architecture ensures proper data flow from administrative and user input to external systems, maintaining integrity and scalability. The relational model provides a solid foundation for querying and managing books, users, interactions, and recommendations.

Future stages of the project will focus on database population, indexing, and performance testing, followed by full integration with backend services and user interfaces.

Chapter 9

References

[1] E. SolutionsHub, *Exploring Business Model Canvas examples*. Oct. 20, 2023. Available: https://solutionshub.epam.com/blog/post/business-model-canvas-examples

[2] IBM, *Relational Databases*. Oct. 20, 2021. Available: https://www.ibm.com/think/topics/relational-databases

[3] Lucidchart, "Database Design Structure - Schema Tutorial". 2025. Available: https://www.lucidchart.com/pages/tutorial/database-design-and-structure

[4] Atlassian, *Acceptance Criteria Explained [+ Examples and Tips]*. 2024. Available: https://www.atlassian.com/work-management/project-management/acceptance-criteria

[5] Ramez E., "FUNDAMENTALS OF FourthEdition DATABASE SYSTEMS.", 2003. [E-book] Available: https://www.uoitc.edu.iq/images/documents/informatics-institute/Competitive_exam/Database_Systems.pdf

Chapter 10

Glossary

Term / Acronym	Definition
ERD (Entity-Relationship Diagram)	A visual representation of the data model showing entities (tables), their attributes (fields), and the relationships between them. Used in the conceptual design phase of databases.
3NF (Third Normal Form)	A stage of database normalization in which all attributes are functionally dependent only on the primary key, eliminating transitive dependencies and redundancy.
Relational Database	A structured collection of data organized into tables with predefined relationships. Ensures data integrity and supports complex queries using SQL.
Subscription Model	A monetization approach where users pay a recurring fee (e.g., monthly) for continuous access to a service or product, commonly used in digital content platforms.
Bookmate	A real-world digital reading platform that offers unlimited access to e-books and audiobooks, along with social features and personalized recommendations. Serves as the inspiration for the platform proposed in this report.
User-Generated Content (UGC)	Content such as reviews, comments, or uploaded books created by users rather than platform administrators. Requires moderation and is key to fostering community.
NoSQL Database	A non-relational database optimized for unstructured or semi-structured data, offering flexibility and high performance in scenarios like metadata storage or recommendations.

Business Model Canvas (BMC)	A strategic tool used to visually map out the core components of a business, including its value proposition, customer segments, key resources, revenue streams, and more.
Foreign Key (FK)	A field in one table that links to the primary key of another table, used to maintain referential integrity in relational databases.
CRUD Operations	The four basic operations of persistent storage: Create, Read, Update, and Delete. Essential for interacting with a database.
Traceability Matrix	A tool that maps requirements to their corresponding system components (e.g., entities, fields), ensuring each requirement is addressed by the design.
Platform Architecture	The high-level design that defines how different components (database, frontend, backend, external services) interact within a software system.