**Database Course Project**

Online Book Store

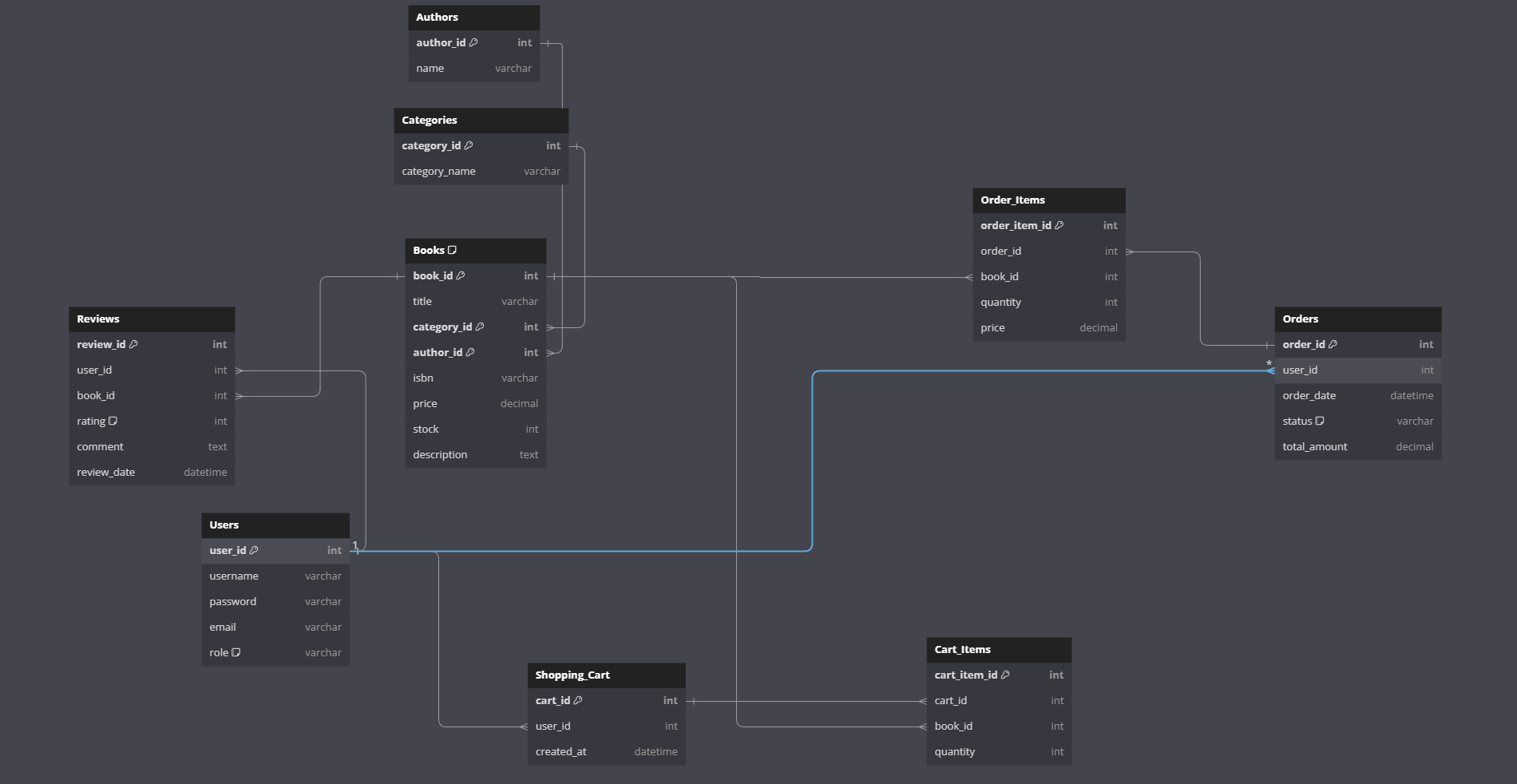
Goal: Fully design and develop DB component for an App.

Author: Andrey Patsino

# Book Online store

1. Introduction:

This report presents the development of a database system for an e-commerce application. It details the process of designing and implementing the necessary database objects to support the application’s functionality. The report describes the creation of an ER diagram, the development of an OLTP solution and the preparation of data for database integration and the construction of an OLAP solution, including a multidimensional data warehouse with fact tables and SCD. It also outlines the development of an ETL process for data migration.

1. ER-diagram:
2. OLTP schema description  
   This component manages various operations such as user management, book inventory, orders, shopping carts, and reviews. Below is a detailed description of the OLTP component and its structure.

### Database Tables:

1. **Users**:
   * **Columns**: user\_id, username, password, email, role
   * **Purpose**: Stores user information, including login credentials and roles (either 'user' or 'admin').
2. **Books**:
   * **Columns**: book\_id, title, isbn, price, stock, description
   * **Purpose**: Maintains the details of each book available in the store, such as title, ISBN, price, stock quantity, and description.
3. **Categories**:
   * **Columns**: category\_id, category\_name
   * **Purpose**: Manages the different categories under which books can be classified.
4. **Authors**:
   * **Columns**: author\_id, name
   * **Purpose**: Stores information about book authors.
5. **Orders**:
   * **Columns**: order\_id, user\_id, order\_date, status, total\_amount
   * **Purpose**: Records all orders placed by users, tracking the order date, status (e.g., pending, shipped, delivered, canceled), and total amount.
6. **Order\_Items**:
   * **Columns**: order\_item\_id, order\_id, book\_id, quantity, price
   * **Purpose**: Details the items within each order, including the book ID, quantity ordered, and price at the time of the order.
7. **Shopping\_Cart**:
   * **Columns**: cart\_id, user\_id, created\_at
   * **Purpose**: Represents the shopping cart for each user, holding the items they intend to purchase.
8. **Cart\_Items**:
   * **Columns**: cart\_item\_id, cart\_id, book\_id, quantity
   * **Purpose**: Contains the items in the shopping cart, detailing the book ID and quantity.
9. **Reviews**:
   * **Columns**: review\_id, user\_id, book\_id, rating, comment, review\_date
   * **Purpose**: Stores user reviews for books, including the rating (1 to 5 stars), comments, and the date of the review.

### Indexes:

* idx\_users\_username: Index on Users(username)
* idx\_books\_title: Index on Books(title)
* idx\_orders\_user\_id: Index on Orders(user\_id)
* idx\_order\_items\_order\_id: Index on Order\_Items(order\_id)
* idx\_cart\_items\_cart\_id: Index on Cart\_Items(cart\_id)
* idx\_reviews\_user\_id: Index on Reviews(user\_id)
* idx\_reviews\_book\_id: Index on Reviews(book\_id)

### Key Functions and Procedures:

1. **AddBookToCart**:
   * **Purpose**: Adds a specified book to a user's shopping cart. If the user doesn't have an existing cart, a new one is created.
   * **Parameters**: p\_user\_id (user ID), p\_book\_id (book ID), p\_quantity (quantity of the book)
2. **PlaceOrder**:
   * **Purpose**: Places an order for all items in the user's shopping cart, transferring the items to the Orders and Order\_Items tables and then clearing the cart.
   * **Parameters**: p\_user\_id (user ID)
3. **DeleteItemFromCart**:
   * **Purpose**: Removes a specified item from the shopping cart.
   * **Parameters**: p\_cart\_item\_id (cart item ID)

### Roles and Permissions:

* **admin**: Full access to all tables and operations.
* **Def\_User**: Limited access to perform select, insert, and delete operations on specific tables, as needed for regular user activities.

1. OLAP Component Description  
   This component involves the creation of dimensional models and fact tables to support analytical queries and reporting. Below is an overview of the OLAP component and its structures.

### Dimensional Models:

1. **Dim\_Customer**:
   * **Columns**: Customer\_ID, Name, Email, StartDate, IsCurrent
   * **Purpose**: Represents customer information, including their name, email, start date (the date when the customer was added to the system), and a flag indicating if the record is current or historical (for SCD Type 2).
2. **Dim\_Product**:
   * **Columns**: Product\_ID, Name, Category, Subcategory, Brand, Price
   * **Purpose**: Stores details about products available in the store, such as name, category, subcategory, brand, and price.

### Fact Tables:

1. **Fact\_Sales**:
   * **Columns**: Sales\_ID, Product\_ID, Customer\_ID, Quantity\_Sold, Total\_Sales, CreatedAt
   * **Purpose**: Records information about sales transactions, including the product sold, customer ID, quantity sold, total sales amount, and the date of the sale.
2. **Fact\_Orders**:
   * **Columns**: Order\_ID, Product\_ID, Customer\_ID, Date\_ID, Order\_Quantity, Order\_Total
   * **Purpose**: Captures data related to customer orders, including the products ordered, customer ID, date of the order (referenced from Dim\_Time), quantity ordered, and total order amount.

### Indexes:

* idx\_fact\_sales\_product\_id: Index on Fact\_Sales(Product\_ID)
* idx\_fact\_sales\_customer\_id: Index on Fact\_Sales(Customer\_ID)
* idx\_fact\_orders\_product\_id: Index on Fact\_Orders(Product\_ID)
* idx\_fact\_orders\_customer\_id: Index on Fact\_Orders(Customer\_ID)

1. Data Loading and Transformation

### A. Data Loading from CSV Files

To load data from CSV files into temporary tables in a PostgreSQL database, the script from file '1.4-Data-Loading.sql' is used. This script is executed twice, once for each dataset (Dataset-1 and Dataset-2).

* **Drop Temporary Tables**: If temporary tables from previous runs exist, they are dropped.
* **Create Temporary Tables**: Temporary tables are created with the same structure as the main tables.
* **Load Data from CSV**: Data is copied from CSV files into the temporary tables using the COPY command.
* **Insert Data into Main Tables**: Data is transferred from temporary tables to the main tables, ensuring no duplicate entries are created.
* **Drop Temporary Tables**: Temporary tables are dropped after the data transfer is complete.

### B. ETL Process

The ETL process in C# is responsible for extracting data from the OLTP system and loading it into the OLAP system. The ETL process uses Entity Framework for database.

To run the ETL script, open the solution from the ETL-Script folder. It contains C# code created to perform ETL on data

* **Context Initialization**: Two database contexts are initialized, one for OLTP and one for OLAP. The contexts weren’t created manually; instead, the Scaffold-DbContext command was used.
* **ETL Process Execution**: The ETL process is executed by calling various methods to extract and load data for different entities such as roles, customers, categories, brands, products, sizes, inventory, and sales.
* **Extract and Load Methods**: Each method extracts data from the OLTP database, checks for existing records in the OLAP database to prevent duplicates, transforms the data as necessary, and then loads it into the OLAP database.