UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Web Applications A.Y. 2024-2025

Homework 1 – Server-side Design and Development

Master Degree in Computer Engineering
Master Degree in Cybersecurity
Master Degree in ICT for Internet and Multimedia

Deadline: 17 April, 2025

Group Acronym	Bookly	
Last Name	**First Name**	**Badge Number**
Jolani Mameghani	Amirreza	2106526
Saeedidana	Parisa	2142717
Doe	Jane	123456
Doe	John	123456
Doe	Jane	123456
Doe	John	123456

1 Objectives

The bookstore web application aims to provide an intuitive, feature-rich, and user-friendly platform for browsing, purchasing, and managing books online. The primary objective of this project is to enhance the online book-shopping experience by offering users a well-organized and efficient system where they can explore books, add them to their shopping cart or wishlist, leave reviews, apply discounts, and securely complete their purchases. This application is designed for book enthusiasts who want a convenient way to discover and buy books without visiting a physical store. Users can create an account, browse books by category, author, or publisher, and use additional features like wishlists to save books for future purchases. The wishlist functionality allows users to keep track of books they are interested in without immediately adding them to their shopping cart, enhancing flexibility in decision-making. A key feature of the system is the shopping cart, where users can add multiple books before proceeding to checkout. The ordering system ensures a structured purchasing process, allowing users to place orders and complete payments securely. Payment transactions include details such as the amount, method, and date, ensuring transparency and security. The discount feature provides users with promotional offers through discount codes, enhancing the affordability of books and improving customer satisfaction. To further engage users, the application includes a review system, allowing customers to leave feedback on books they have read. This feature helps other users make informed decisions based on ratings and reviews. Additionally, book categories, authors, and publishers are well-structured, making book discovery easier and more efficient. By integrating essential e-commerce functionalities with interactive features such as wishlists and reviews, this project aims to create a modern, engaging, and accessible online bookstore. The system is designed to improve book accessibility, offer a smooth shopping experience, and provide users with a reliable platform for discovering, saving, reviewing, and purchasing books in an efficient and enjoyable way.

2 Main Functionalities

Bookly is an interactive online bookstore designed to provide a seamless experience for both customers and administrators. The platform allows users to explore and purchase books while ensuring efficient management of payments. At the core of the system, customers can search for books by various criteria, including title, author, and use a filter of category, and publisher, to find a book easier. The book details page is contain of book information, such as ISBN, price, stock availability, and publication year. Additionally, customers can engage with the platform by bookmarking books for future reference, leaving reviews, and receiving personalized recommendations based on their browsing and purchasing history.

To enhance the purchasing experience, customers can add books to their shopping cart and proceed to checkout using a simple and secure payment method. The system ensures that stock availability is updated in real-time, preventing purchasing unavailable books. Customers can also track their orders. Alongside managing their account information.

Bookly is built on a structured system that integrates RESTful APIs to handle book data, user interactions, and order processing. The system includes error-handling mechanisms that provide meaningful feedback for issues such as failed transactions or invalid inputs. By maintaining a clear and organized approach to managing book sales and customer interactions, Bookly ensures a streamlined and user-friendly experience for both customers and administrators, making book discovery, purchase, and management effortless.

3 Data Logic Layer

3.1 Entity-Relationship Schema

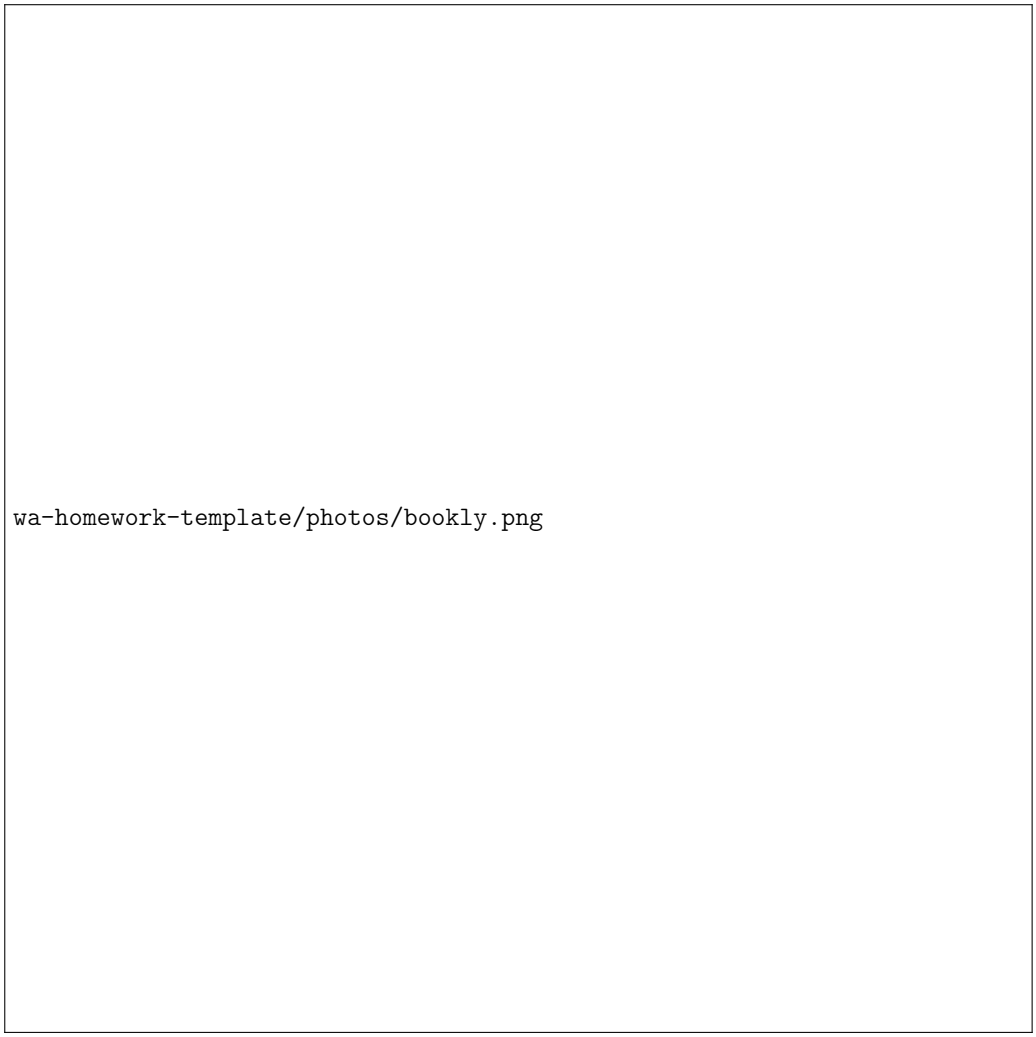


Figure 1: ER-Diagram

The ER schema contains the following entities:

User represents individuals who interact with the Bookly platform, such as browsing books, placing orders, writing reviews, and creating wishlists. Each user is uniquely identified by `user_id` (type: SERIAL, primary key). The `username` (type: booklySchema.username_domain, i.e., `VARCHAR(50)` with format constraints) must be unique, and the `password` (type: booklySchema.password_domain, i.e., `VARCHAR(255)`) must have at least 8 characters. Users also have `first_name` and `last_name` (both of type: `VARCHAR(50)`), and a unique `email` (type: `VARCHAR(100)`). The optional `phone` field uses a custom domain (type: phone_domain, i.e., `VARCHAR(20)`) that enforces a valid format, and `address` (type: `TEXT`) stores shipping or billing information. The `role` attribute is defined by an ENUM type `user_role` (values: 'user', 'admin') and defaults to 'user'. Each user is associated with a unique shopping cart via the `shopcart` foreign key (type: INTEGER), linking to the `shopcart` entity (1–1 relationship).

Author captures metadata about individuals who have written books. Each author is identified by a unique `author_id` (type: SERIAL, primary key). Other attributes include `first_name` and `last_name` (both of type: `VARCHAR(100)`), a `biography` field (type: TEXT) containing an optional description of the author's life or achieve-

ments, and nationality (type: VARCHAR(100)) describing the author's country of origin. Authors can be linked to one or more books through the writes relationship, which is modeled as a many-to-many association.

Publisher represents companies or organizations that publish books. Each publisher has a unique publisher_id (type: SERIAL, primary key), a publisher_name (type: VARCHAR(100)), an optional phone number (type: phone_domain), and an address (type: TEXT). Each book can be published by exactly one publisher, and this is modeled by the published_by relationship (many-to-one).

Category is used to classify books by genre, theme, or other organizing criteria. It includes a category_id (type: SERIAL, primary key), a category_name (type: VARCHAR(50)), and an optional description (type: TEXT). A book can belong to one or more categories, managed by the category_belongs relationship (many-to-many).

Book is the core item of the platform, representing all available reading material. Each book is uniquely identified by book_id (type: SERIAL, primary key), and includes a title (type: VARCHAR(150)), optional language (type: VARCHAR(50)), and a unique isbn (type: VARCHAR(20)). The price (type: REAL) is a required field representing the cost. Additional attributes include edition (type: VARCHAR(50)), publication_year (type: INTEGER) restricted between 1000 and the current year, number_of_pages (type: INTEGER, must be positive), and stock_quantity (type: INTEGER, default 0, must be non-negative). The average_rate (type: REAL) stores the average of user-submitted ratings, and summary (type: TEXT) provides an optional overview of the book. Books are connected to categories, publishers, authors, shopping carts, and reviews via appropriate relationships.

ShopCart represents a temporary list of books a user intends to purchase. Each cart has a cart_id (type: SERIAL, primary key), a shippment_method (type: ENUM shippment_method, values: 'in_person', 'credit_card'), and a user_id (type: INTEGER) as a unique foreign key referencing users, enforcing a 1–1 user-cart relationship. It also includes created_date (type: TIMESTAMP, default: current time), quantity (type: INTEGER, default: 0), discount (type: INTEGER) as a foreign key referencing the discounts table, and order (type: INTEGER) as a foreign key referencing the orders table.

Order represents completed purchases. Each order has a unique order_id (type: SERIAL, primary key), a total_amount (type: NUMERIC(10,2)) representing the final price, a payment_method (type: ENUM payment_method), and a payment_date (type: TIMESTAMP, default: current timestamp). Orders are linked to shopping carts (1–1), and may include discounts.

Discount defines promotional codes that reduce order totals. It includes a discount_id (type: SERIAL, primary key), a unique code (type: VARCHAR(50)), a discount_percentage (type: NUMERIC(5,2)) restricted to 0–100, and an optional expired_date (type: DATE). Each discount can be associated with a shopping cart or order to adjust pricing.

Reviews captures feedback from users about books. Each review has a unique review_id (type: SERIAL, primary key), and includes user_id and book_id (both of type: INTEGER) as foreign keys. It also includes review_text (type: TEXT), a rating (type: INTEGER between 1 and 5), review_date (type: TIMESTAMP, default: current timestamp), and optional counters for number_of_likes and number_of_dislikes (both of type: INTEGER). This entity supports the social and feedback features of the platform.

Wishlist allows users to save books they are interested in for future reference without adding them to the cart. Each wishlist entry has a unique wishlist_id (type: SERIAL, primary key), and includes user_id and book_id (both of type: INTEGER) as foreign keys to link the wishlist to a specific user and book. Additionally, it contains wishlist_date (type: TIMESTAMP, default: current timestamp), to track when a book was added to the wishlist. This entity enhances user experience by enabling users to keep track of books they may want to purchase later.

3.2 Other Information

(Amirreza - parisa)

4 Presentation Logic Layer

When a user accesses the website, they will be directed to the home page without needing to sign in. From there, they can navigate to various other sections, including:

- Sign In/Sign Up Page : For user authentication.
- User Profile Page : Displays user details.
- Book Page : Provides details about a specific book.
- Basket Page : Shows the user's selected books for purchase.
- Search Results Page : Displays books based on the user's search query.

4.1 Home Page

The Home Page serves as the central hub where all users (whether signed in or not) can browse through the book catalog. It will feature several sections, such as:

- **Most Popular** : Showcasing books with the highest reviews from our users.
- **New Arrivals** : Displaying the latest books added to our stock.
- **Best Sellers** : Highlighting books with the highest sales.

Additionally, the homepage will include:

- A **search bar** allowing users to find books by title, author, or genre within the store's catalog.
- A **basket icon** that provides quick access to the user's shopping cart. Users can click on it to navigate directly to the Basket Page.
- An **account icon** that, when clicked, redirects the user to their User Profile Page if they are signed in. If not, they will be prompted to the Sign In/Sign Up Page.

4.2 Sign In/Sign Up Page

This page allows users to register or log in to their accounts. It includes fields for username, password, an option for password recovery, and a button leading to the Sign-up Page.

4.3 User Profile Page

Users can also access their profile page, which displays their personal details and order history. The page will include the following information:

- **First Name and Last Name**: The full name of the user.
- **Role**: Indicates whether the user is a *Client* or an *Admin*.
- **Username**: A unique identifier for the user within the platform.
- **Email Address**: The registered email associated with the account.
- **Address**: The user's delivery address.
- **Phone Number**: The user's contact number.

The page will also provide:

- **Edit Buttons**: Users will have the option to update their personal information through dedicated buttons.
- **Change Password Link**: A dynamic text link labeled *"Change my password"* that, when clicked, redirects users to a dedicated password update page.

4.4 Book Page

Each book will have its own dedicated page containing the following details: **Title**, **Author**, **Price**, **Edition**, **Genre**, **Summary** and a **cover image** of the book.

In top of that, the page will also have:

- A **Review Section**, where users can:
  - Like or dislike existing reviews.
  - Add their own review by writing a comment and rating the book on a scale from **0 to 5 stars**.
- A **"Add to basket"** button, allowing users to purchase the book.
- A **"Add to wishlist"** button, allowing users to save the book on their wishlist.

4.5 Basket Page

This page will display the books added on the basket by the user, along with options to update quantities. It will also include the following features:

- A **recap of all selected books**, displaying their titles, prices, and quantities.
- The **total price**, dynamically updated based on the selected books, their quantities and the shipping price.
- Several input fields for the user to enter their **billing address**.
- A **payment method selection**, offering the choice between:
  - **Credit/Debit Card Payment**.
  - **Cash on Delivery**.
- An optional input field for entering a **discount code** if the user has one.
- A **"Confirm Order"** button that finalizes the payment and processes the order once all required fields are completed.

4.6 Search Results Page

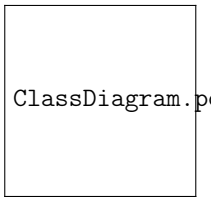
When a user enters a query in the search bar, they will be redirected to this page, where relevant books will be displayed based on title, author, or genre.

For each book displayed in the search results, the following details will be shown:

- The **book title** and **cover image**, both acting as clickable links that redirect to the corresponding Book Page.
- The **average rating**, displayed as a star rating out of 5, based on user reviews.
- A **"Add to Basket"** button along with the corresponding price of the book, allowing users to quickly add the book to their shopping cart.

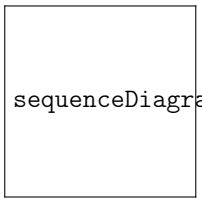
5 Business Logic Layer

5.1 Class Diagram



(Amirreza and parisa)

5.2 Sequence Diagram



(Adam)

5.3 REST API Summary

URI	Method	Description	Filter
(Gul) What is the URI?	What HTTP method uses?	Describe briefly the result of calling the URI	is it behind a filter?

Table 2: Describe in this table your REST API

5.4 REST Error Codes

Error Code	HTTP Status Code	Description
(Amirreza and gul) Error Code Identifier	Corresponding HTTP Status Code	Description of the error

Table 3: Describe in this table your REST API

5.5 REST API Details

A resource

(Gul)

- URL: `the URL to retrieve it`
- Method: `Method to retrieve it`

- URL Parameters:
- Data Parameters:
- Success Response:
- Error Response:

6 Group Members Contribution

Parisa Saeedidana Wrote the objective of the project, Drew the ER diagram for the bookstore.

Amirreza Jolani mameghani Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Jane Doe Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

John Doe Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Jane Doe Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

John Doe Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.