

ASSO - Library System Architecture

Homework 5

Team 33

Guilherme de Matos Ferreira de Almeida
João Pedro Carvalho Moreira
Jorge Daniel de Almeida Sousa
Lia da Silva Linhares Vieira
Nuno Afonso Anjos Pereira



Masters in Informatics and Computing Engineering

23/02/2024

Contents

1	Introduction	2
2	System Requirements	3
3	System Overview	4
3.1	Architectural Overview	4
3.2	Front-end: An offline-first web-page	5
3.3	Class Diagram	6
3.3.1	User	6
3.3.2	Staff	6
3.3.3	Patron	7
3.3.4	Access Card	7
3.3.5	Library	7
3.3.6	Media	7
3.3.7	Metadata	7
3.3.8	Reservation	7
4	User Interaction	8
4.1	Users	8
4.2	Authentication	8
4.3	Catalog Search	8
4.4	Check Out System	8
4.5	Notification System	8
5	Inventory Management	9
6	Inter-Library Collaboration	9
7	Fines and Penalty System	10
7.1	Fine Calculation	10
7.2	Notification of Overdue Items	10
7.3	Payment and Settlement	10
7.4	Fine Enforcement	10
7.5	Fine Management	10
8	Further Considerations	11
8.1	Scalability and Flexibility Constraints	11
8.1.1	Solution: Monitoring	11
8.2	Cost Considerations	11
8.2.1	Solution: Incremental Expansion	11
9	Conclusion	13

1 Introduction

The design and implementation of an efficient library system architecture are crucial in modernizing library operations, enhancing user experience, and optimizing resource management. This project focuses on conceptualizing and outlining the architecture for an automated library system that caters to the evolving needs of patrons and librarians alike. This system aims to streamline processes such as book checkout, catalog search, inventory management, and inter-library collaboration by leveraging technology. Through this report, we explore various modules, interactions, and considerations essential for the successful deployment of a robust library system. The following sections delve into the intricacies of the system's architecture, providing a comprehensive framework for its development and implementation.

2 System Requirements

1. **Automated Library System:** Design an architecture for a library system with automated functionalities.
2. **User Authentication:** Users (Patrons and Staff) must have accounts for accessing library services. Authentication methods include scanning library cards or entering credentials.
3. **Book Checkout:** Patrons can check out books from the library.
4. **Catalog Search:** Patrons and staff can search the library media catalog (books, DVDs, magazines, etc.) using tags, popularity, stock and many other criteria depending on the media format. Search can be conducted at dedicated stations within the library.
5. **Notification System:** Patrons are notified of late books and upcoming due dates. Notification methods can include direct connect terminals, ChromeBooks, laptops, or mobile devices.
6. **Book Reservation:** Patrons can reserve books that are currently checked out.
7. **Inventory Management:** System should support inventory management such as adding and removing books from the inventory.
8. **Multi-media Support:** Different types of media should be supported including physical books, magazines, DVDs, etc.
9. **Mobile Interaction:** Library interaction should be possible via mobile phones.
10. **Collaboration with Other Libraries:** The system should support collaboration with other libraries, though the specifics are not defined.
11. **Fine Management:** Patrons should be fined for late returns of media items.

3 System Overview

3.1 Architectural Overview

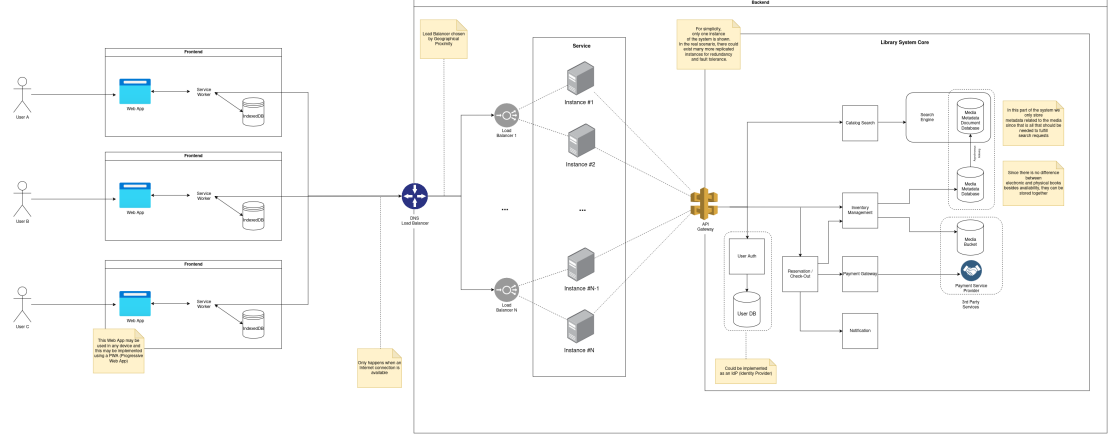


Figure 1: Architectural Overview

The entire system architecture can be described as a **Client-Server** architecture, having Web-Based clients connect to system servers. These clients can also be mobile devices using mobile browsers.

The servers can only be accessed through a load-balancing layer, which itself can only be accessed by means of geo-based load-balancing in order to minimize latency.

The servers themselves follow a **Distributed** service model, with replication occurring at the Media Database level. Other Data Stores can be populated from the main Media Database, meaning that they do not need to be distributed.

Each instance of the server is divided into 6 main modules:

- User Auth
- Reservation / Check Out
- Inventory Management
- Payment Gateway
- Catalog Search
- Notification

with each module's functionality being described below.

The User Auth module can be implemented as an external service by means of an **Identity Provider**: in this situation it is not a direct module of each instance but a centralized service accessible by all instances.

There is a process that periodically indexes the Media Metadata Database into a specialized document data store in order to facilitate usage with Search Engines, which in turn integrate directly with the **Catalog Search** module.

All actual digital media content is stored in a 3rd party object storage service like AWS S3. Fines and reservation payments are mediated by a **Payment Gateway** module that interfaces with an external Payment Service Provider.

3.2 Front-end: An offline-first web-page

The front-end will be developed as an offline-first web page. Typically, web pages are only accessible when the user is online. However, using the Service Workers API [2] on modern browsers, we can cache our page's assets, such as HTML, CSS, and JavaScript, on the users' browsers. This makes it possible to offer an offline experience on our web page.

The Service Worker acts as a proxy for the application front-end. It enables background synchronization and manages offline operations. Additionally, it has its own local IndexedDB, which is an in-browser key-value database [1]. The local IndexedDB ensures that users can make modifications, add or remove items from their shopping lists, and perform other operations even without an active internet connection.

Thus, this setup allows the application to function seamlessly, even in offline scenarios, by persisting data locally. Therefore, we can achieve a local-first design that provides a better experience for our users.

3.3 Class Diagram

The class diagram illustrates the structure of the library system by depicting the classes and their relationships. Each class represents a fundamental entity or concept within the system.

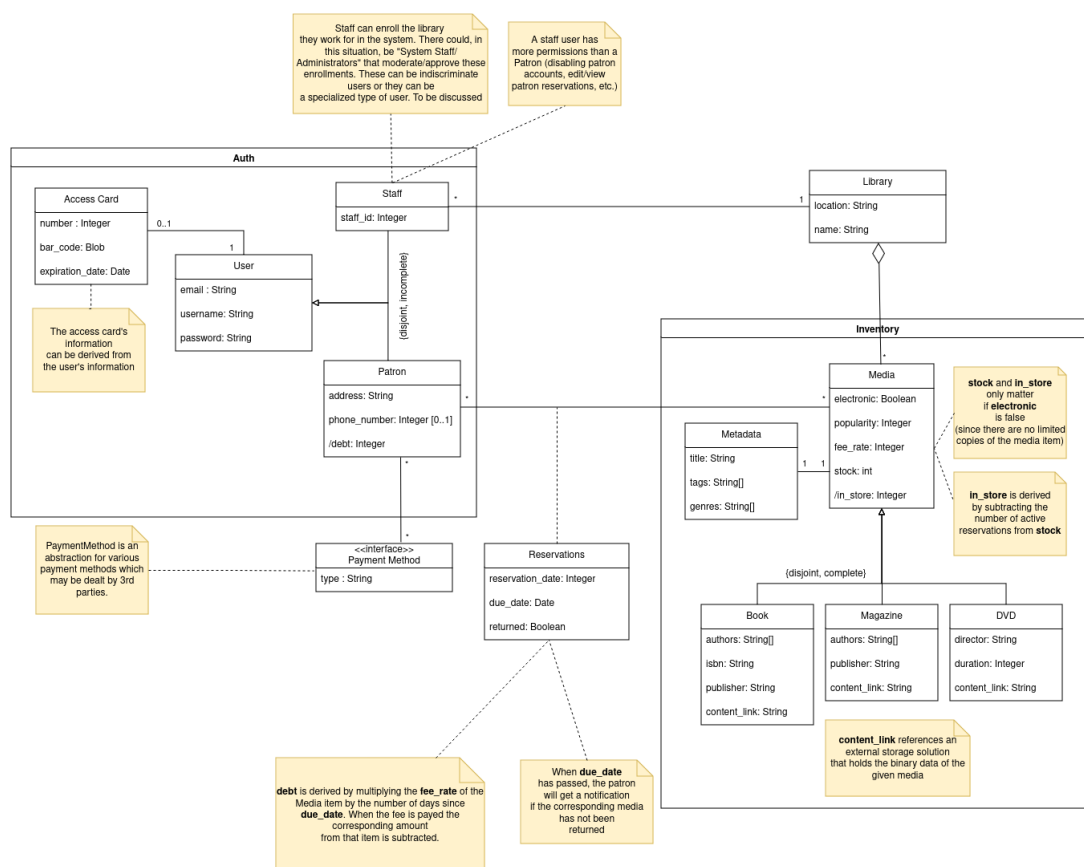


Figure 2: Class Diagram

3.3.1 User

The User class represents individuals who interact with the library system. Users can be further categorized into Patron and Staff subclasses.

3.3.2 Staff

The Staff class represents employees or administrators who have additional privileges within the system, such as managing inventory and user accounts.

3.3.3 Patron

The Patron class represents users who primarily utilize the system for borrowing and returning items, searching the catalog, and managing their accounts.

3.3.4 Access Card

The Access Card class represents the cards used by users for authentication. These cards contain unique identifiers linked to user accounts.

3.3.5 Library

The Library class represents individual library instances within the system. Each library has its collection of media items, users, and administrative staff.

3.3.6 Media

The Media class represents both physical and digital items available in the library's collection, such as books, DVDs, or magazines.

3.3.7 Metadata

The Metadata class contains essential information about media items, including title, tags and genres. It is linked to the Media class to provide comprehensive details for catalog search and management.

3.3.8 Reservation

The Reservation class represents the process of reserving media items that are currently checked out by other patrons. It manages the queue of reservations and notifies patrons when their reserved items become available.

These classes collectively form the backbone of the library system, facilitating user interactions, resource management, and administrative tasks. Each class encapsulates specific functionalities and data attributes essential for the system's operation, as you can observe in Figure 2.

4 User Interaction

4.1 Users

Users can be categorized into **patrons** or **staff** members. Patrons have access to features such as browsing the catalog, checking out media, and managing their accounts. Meanwhile, staff members are affiliated with the library and have additional privileges to perform administrative tasks, such as disabling patron accounts and viewing reservations.

4.2 Authentication

Authentication can be done via the user's access cards or credentials such as usernames and passwords. Each card contains a unique identifier that links to the user's account within the system and makes it easier for users to log in and use library services.

To make user authentication easier and safer, the system could allow integration with third-party identity providers (IdPs). The ability for patrons to access the library system using pre-existing login credentials from sites like Google, Facebook, or institutional login systems simplifies the authentication procedure and improves user comfort.

4.3 Catalog Search

Patrons and staff utilize the catalog search functionality to explore the library's collection based on various criteria such as author, title, genre, or topic. Accessible through a web interface or a mobile application, the search interface provides a user-friendly experience, enabling users to quickly locate desired items.

4.4 Check Out System

The media reservation process is a fundamental interaction point for patrons within the library system. Following authentication, patrons gain access to the catalog, where they can search and select items to check-out. This system also allows patrons to reserve items currently in circulation. Patrons receive notifications when the due date of their reservations approaches and are subject to fines if they return the media past the due date.

4.5 Notification System

The notification system plays a crucial role in keeping patrons informed about important events, such as upcoming due dates, late returns, or the availability of reserved items.

5 Inventory Management

The Inventory Management module deals with the organization and availability of the library's collection. Staff members can insert new items manually into the system and ensure accurate cataloging. This module controls the availability of physical copies, coordinating with the **Check Out System** to manage item requests. Regular maintenance tasks, such as reviewing the catalog for discrepancies and updating item statuses should be conducted to maintain the integrity of the inventory. Overall, the Inventory Management module plays a crucial role in ensuring efficient access to library resources and optimizing resource management.

6 Inter-Library Collaboration

Since the developed system was not made for a single library but any amount of libraries, collaboration between them is implicitly implemented by the system's design.

Using the load-balancing strategy suggested, the need for synchronization between instances arises. Each instance's document store can be indexed from the same instance's database and media storage buckets are implemented as a 3rd party service, accessible by every instance.

This way, we can conclude that only the **Media Metadata Database** needs to be synchronized between system instances when it comes to media consumption and management.

Regarding authentication information, if authentication is implemented as part of the system itself, the data stores that hold that information also need to be replicated (otherwise a user could lose access to the system because the instance they are accessing does not have their user information). If instead user authentication is delegated to an external service (such as an Identity Provider, or IdP for short) this does not become an issue directly related to our system.

7 Fines and Penalty System

The fines and penalty system is an integral part of the library system, ensuring accountability and timely return of borrowed items. When patrons fail to return items by their due dates, fines are incurred according to predefined rules.

7.1 Fine Calculation

Fines are calculated based on the type of media borrowed and the duration of the overdue period. Each media type may have a different fine rate per day, and these rates are configurable within the system. For example, the fine for overdue books might be \$0.10 per day, while DVDs might incur a higher fine of \$0.50 per day.

7.2 Notification of Overdue Items

Patrons are notified of overdue items through the **Notification** module. Automated reminders are sent via email, SMS, or through the library system interface. These notifications inform patrons of their overdue items and the accruing fines.

7.3 Payment and Settlement

Patrons are responsible for settling fines accrued due to late returns. The library system provides various payment methods, including in-person payments at the library, online payments through the library's website, or integration with third-party payment gateways.

7.4 Fine Enforcement

Fines must be enforced consistently to maintain fairness and accountability within the library system. Patrons with outstanding fines may face consequences such as temporary suspension of borrowing privileges, blocking access to certain library services, or referral to collections agencies for severe cases of non-payment.

7.5 Fine Management

Library staff members oversee the management of fines within the system. They have access to tools and reports to track overdue items, monitor fine payments, and enforce penalties when necessary. Staff members can also provide assistance to patrons with questions or concerns regarding fines.

The fines and penalty system operates in conjunction with other modules of the library system, such as the check-out system and user authentication, to ensure smooth operation and adherence to library policies.

8 Further Considerations

8.1 Scalability and Flexibility Constraints

The scalability and flexibility of our system may be affected by the constraints imposed by external service providers. Changes or updates to their APIs, policies, or pricing models could necessitate adjustments to our system, leading to increased maintenance overhead and potential disruptions to our service. Additionally, security issues induced, service outages and downtime experienced by external providers could result in service interruptions for our library system, affecting user experience and operational continuity.

8.1.1 Solution: Monitoring

To mitigate the risks associated with this reliance on external services, we should implement robust system monitoring. By continuously monitoring the performance and availability of external services, we can quickly identify any anomalies or issues that may arise. Automated monitoring tools can provide real-time alerts and notifications, allowing our team to respond much quicker to any problem that is detected within their system. By maintaining visibility and oversight of external service dependencies, we can minimize the impact of scalability and flexibility constraints on our system and ensure an uninterrupted and safer service delivery to our users.

8.2 Cost Considerations

The complexity of our system architecture, with components such as media buckets, load balancers, and distributed services, introduces cost considerations that need to be carefully managed. Implementing and maintaining these components can incur significant expenses, including infrastructure costs, licensing fees, and ongoing operational expenses.

In an early adoption phase of the project, when the user amount is not heavy, some components may not be completely necessary.

8.2.1 Solution: Incremental Expansion

To address the cost considerations associated with our system architecture, we can adopt an incremental expansion approach. Rather than implementing all components and features of the system at once, we can start with a smaller, more streamlined version and gradually expand and enhance it over time. By starting smaller, we can eliminate some of the complex layers, such as the load balancers, or server redundancy initially, reducing initial infrastructure costs and simplifying the design. As the demand for resources and functionality grows, we can incrementally add these components and features, adapting the design to meet evolving requirements while effectively managing costs. This iterative

approach allows us to scale the system in a controlled manner, optimizing resource allocation and ensuring cost-effectiveness throughout the development and deployment process.

9 Conclusion

In conclusion, the Library System architecture demonstrates a comprehensive approach to modernizing library operations while prioritizing efficiency and experience for patrons and staff.

The user interaction aspects, including authentication methods and mobile integration, ensure accessibility and overall usability of the system. Furthermore, the integration of inter-library collaboration features underscores the system's scalability. However, continuous refinement and adaptation can be done to address evolving user needs.

References

- [1] MDN Contributors. IndexedDB API — MDN, Mozilla Documentation Network. https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API, 2023. [Online; accessed 2-November-2023].
- [2] MDN Contributors. Using Service Workers — MDN, Mozilla Documentation Network. https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API/Using_Service_Workers, 2023. [Online; accessed 3-November-2023].