Use Case: Community Library Management System (CLMS)

**Overview:**

The CLMS, developed in the Java programming language, is BookHaven MS's groundbreaking solution for rejuvenating community libraries. It centralizes the management of employees, clients, books, and transactions, offering self-service features.

**Use Case:** Implementing the CLMS

**Scope:** Community Member Interaction

**Level:** user goal

**Primary Actor:** Client/Employees

**Stakeholders and Interests:**

-Employee/Librarian: Requires a system that simplifies the management of books and clients, providing tools for accountability and book tracking.

-Client/Community Member: Desires an organized system for borrowing and returning books via both self-service kiosks and traditional customer service.

-System Administrators: Ensure system stability, manage updates, and uphold an intuitive UI/UX.

-BookHaven MS Management: Wants an automatic and fast update of inventing and accounting(accounting of money if only a client exceeds his/her date limit of return). Interested in a system that ensures user satisfaction, precise book tracking, and transparent transaction records.

– Payment Authorization Service: Wants to receive digital authorization requests in the correct format and protocol. Wants to accurately account for their payables to the store.

**Success Guarantee (Postconditions):** Ability to manage employees, clients, and books, resolve discrepancies with the system, and generate reports on borrowing and returning patterns. Accountability and transparency in the library’s operation with clear records of all book transactions.

**Main Success Scenario (Basic Flow):**

**Admin**

1. Admin logs in/out
2. Admin manages employees
   1. Adds employees
   2. Enables/disables employees
3. Admin manages clients
   1. Adds clients
   2. Enables/disables clients

**Employee (Client using traditional customer service to borrow/return the book)**

1. Employee logs in/out
2. Employee manages clients
   1. Adds clients
3. Employee manages books
   1. Adds books
   2. Removes books
   3. Lends books to clients
   4. Receives returned books from clients
4. Employee browses book availability
5. Employee generates inventory report

**Client using traditional customer service to borrow a book:**

1. **Initiating Borrowing Process:**

* The client approaches the customer service counter, expressing the desire to borrow books.
* The helpful library employee assists the client in the borrowing process.

1. **Selection of Books:**

* With a stack of chosen books, the client places them on the counter for borrowing.

1. **Scanning Books:**

* The library employee takes each book and, using an integrated scanner, scans the barcodes.

1. **Checking Book Availability (optional):**

* The employee uses the BookHaven system to verify the availability of each book in real time. If a book is reserved or currently checked out, the client is notified promptly.

1. **Confirming Borrowing Transaction:**

* Once all books are confirmed available and scanned, the employee finalizes the borrowing transaction.

1. **Updating System Records:**

* The BookHaven system updates the client’s borrowing history, reflecting the newly borrowed books.
* Simultaneously, the library’s inventory is updated, reducing the available stock of the borrowed books.

**Client using traditional customer service to return a book:**

1. **Initiating Return Process:**

* The client approaches the customer service counter with books they intend to return.
* The library employee assists the client in the return process.

1. **Scanning Returned Books:**

* Each returning book is scanned by the employee using the integrated scanner.

1. **Verifying Return Details:**

* The system verifies that the returned books match the client's borrowing history, ensuring accurate returns.

1. **Confirming Return Transaction:**

* The employee confirms the return of each book in the system.

1. **Updating System Records:**

* The BookHaven system updates the client’s borrowing history, moving the returned books from 'Currently Borrowed' to 'Past Borrowed Books'.
* The library’s inventory is also updated, increasing the available stock of the returned books.

1. **Viewing Returning History (optional):**

* If desired, the client can request a view of their return history.
* The library employee navigates the BookHaven system to display the client's list of 'Currently Returned Books' and 'Past Returned Books', allowing the client to keep track of their return habits and any overdue fines.

**Client using self-checkout**

**Login/Logout Process:**

* The client approaches the self-service kiosk.
* The client logs into their account.
* The system provides a clear and user-friendly interface with options to scan, borrow, and return books.

**Borrowing Books:**

* The client selects the option to borrow books
* Using the integrated scanner, they scan the barcodes of the books they wish to borrow
* The kiosk system checks the availability of the books in real time. If a book is unavailable or reserved, it notifies the client immediately.
* Once all desired books are scanned, the client confirms the borrowing transaction
* The system updates the client’s borrowing history and the library’s inventory instantly.

**Returning Books:**

* The client chooses the option to return books
* They scan the barcodes of the books they are returning
* The kiosk check book’ state and confirm each return, ensuring the correct book is being returned by the correct client and noting the return date
* The system immediately updates its inventory and the client’s borrowing history, reflecting the returned books.

**Logout:**

* Once all desired transactions are completed, the client logs out, ensuring their account details are secure.

**Extensions (Alternative Flow):**

**System administrators perform updates or maintenance:**

**1. The system requires downtime for a significant update:**

* All users are notified in advance and provided with a timeline for the system's unavailability.

**Admin:**

1. **Admin logs in/out:**

**a.** If admin credentials are incorrect, show an error message and prompt for re-entry / navigate to the menu.

**b.** If the admin forgets the password, provide a "Forgot Password" option to recover/navigate to the menu.

1. **Admin manages employees:**

**a.i.** If the employee's details are incomplete during addition, prompt to complete all fields.

**a.ii.** If an added employee has a duplicate ID or email, show a warning and prevent addition.

**b.i.** If the admin tries to disable an already disabled employee, show a warning.

**b.ii.** Before disabling, prompt admin for confirmation.

1. **Admin manages clients:**

**3.a.i.** If client details are incomplete, prompt to fill in all necessary fields.

**3.a.ii.** If an added client has a duplicate ID or email, warn and prevent the addition.

**3.b.i.** If the admin tries to disable an already disabled client, display a warning.

**3.b.ii.** Prompt for confirmation before disabling a client.

**Employee/librarian:**

1. **Employee logs in/out:**

**a.** If employee credentials are incorrect, show an error message and prompt for re-entry or navigate the employee to the menu.

**b.** If the employee account is disabled by the admin, display an account-disabled message.

1. **Employee manages clients:**

**a.i**. Incomplete client details prompt a message for full entry.

**a.ii.** If a client with the same ID or email is added, provide an error message.

1. **Employee manages books:**

**a.i**. If the employee tries to disable an already disabled book, warn them.

**a.ii.** Before disabling, ask the employee for confirmation.

1. **Employee lends books to clients:**

**a.** If a client has outstanding late books (by checking suspension), warn the employee before lending or band the client.

**b.** If the book is not available, inform the employee.

1. **The employee receives returned books:**

**a.** If a returned book is damaged, allow the employee to mark it, notify, and charge the client some fee.

**b.** If the book is returned late, calculate and inform the client of late fees.

1. **Employees can browse the availability of the books:**

**a.** If the system is unable to fetch the book list, display an error message.

1. **Employees can generate inventory reports:**

**a.** If the system fails to generate the report, prompt an error message.

**b.** Allow filtering options in the report by date, availability, etc.

**Client:**

**1. The client signs in using their credentials:**

**1.a. Client fails authentication:**

* Incorrect client credentials result in an error message.
* If the client account is disabled by the admin, notify them upon attempted login. Otherwise, the system provides an option for the client to retry or navigate to the main menu.
* The forgotten password option provided security questions to verify identity.

**2. Client browses books, and enters a book ID/book name/author name/genre for borrowing:**

**2.a. Book ID/book name/author name/genre entered by the client is invalid or does not exist:**

* The system prompts the client for re-entry.
* Option provided to browse available books and choose from there.

**3. The client returns the borrowed book:**

**3.a. Returned book was not borrowed by the client:**

* The system alerts the user of the discrepancy.
* Prompts for the correct book ID or offers to scan from the client's borrowing history.
* If the returned book belongs to another client or is not recognized in the system, proper protocols are followed to ensure correct handling.

**3.b. Client returns the damaged book or past the return date by using a self-checkout kiosk or traditional service:**

* The client will be charged an amount of fee as a penalty. They may pay by cash, debit, or credit.
* Client does not return the book after a long time or does not pay the penalty fee.The client may get banned from the system.

**4. Client approaches self-service kiosk:**

**4.a. Kiosk is offline or unresponsive:**

* The system alerts the client of the issue and displays an error message.

**4.b. Client logs into the self-service kiosk using their credentials:**

* If credentials are incorrect, the kiosk prompts for re-entry / navigate to the menu.
* If the client forgets the password, the kiosk offers a "Forgot Password" option.

**Special Requirements:**

* Integration of self-service kiosks with real-time database synchronization to ensure accurate inventory updates.
* Secure authentication mechanism for user sign-in at both self-service kiosks.
* Backup and restore functions to safeguard data.

**Technology and Data Variations List:**

The system should be compatible with the desktop interface via a console app built in Java and integrated self-service kiosks.

**Frequency of Occurrence:**

* Employee/librarian sign-in: Daily
* Book transactions (rent/return): Multiple times daily
* Client sign-in: Daily
* Book and client data updates: Occasionally

**Open Issues:**

* Need clarity on the reporting module specifics: What kind of reports are required and how often?
* The mechanism for feedback collection from community members is yet to be detailed.
* This enriched use case, while combining elements from all three texts, provides a comprehensive understanding of the CLMS's functionality, addressing the various stakeholders' interests and concerns.