**Case Study: Library Management System – SDLC Phases Implementation**

**Project Description:**

A **Library Management System (LMS)** is a software application designed to simplify the process of managing a library's operations.

**Actors:**

* **Members**: Customers
* **Librarian**: Head of Library
* **Admin**: Oversees system setup.

**Use Case Scenarios:**

**Searching for Books**

* The member logs into the Library Management System.
* The member searches for books using title or genre.
* The system displays matching results with book availability.

**Borrowing and Returning Books**

* The librarian logs into the system and accesses the member's account.
* The librarian scans the book’s barcode and updates the issue date and due date.
* When returned, the book is marked as available, and fines are calculated.

**Reserving a Book**

* A member logs in and searches for a book.
* If the book is unavailable, they can place a reservation.
* The system notifies the member when the book becomes available.

**Managing Inventory**

* The librarian adds new books, removes old ones, or updates book details.
* The admin generates reports on how much books are being used.

**Sending Notifications**

* The system sends reminders for due dates, overdue books, or reserved book availability.
* Admins and members receive email/SMS/in-app notifications.

**SDLC Phases and Their Contribution:**

**1. Requirements Gathering**

1. Meetings held with librarians, members, and admins to understand daily processes.
2. Functional requirements: search books, return, track due dates.
3. Non-functional requirements: user-friendly interface, fast response time, secure login.

**2. System Design**

1. Created user friendly and detailed designs for user flows, book management, and admin tasks.
2. ER diagrams designed for books, members, transactions, and reservations.

**3. Implementation**

1. Developed front-end interfaces for members and staff.
2. Back-end APIs implemented to handle transactions, book inventory, and user roles.
3. Integrated secure login, tracker, and fine calculations.

**4. Testing**

1. Each module like book search, issue/return tested separately.
2. Full LMS tested for functionality and performance.
3. Checked login security, data protection, and access.
4. Library staff used the system and provided feedback before launch.

**5. Deployment**

1. LMS hosted on cloud servers for 24/7 access.
2. Easy architecture for smooth code updates.
3. HTTPS encryption for safety.

**6. Maintenance**

1. Fixed bugs reported by users.
2. Added features like e-book integration and mobile access.
3. Regular backups and system performance optimization carried out.

**How They Interconnect:**

1. Accurate requirements led to proper design and implementation.
2. Feedback from users influenced improvements before deployment.
3. Ongoing maintenance ensures the system remains useful and error-free.

**Benefits Achieved:**

* Reduced manual work and paperwork for librarians.
* Real-time availability and book tracking improved user satisfaction.
* Faster issue and return process with automated due date and fine tracking.