Library Management System (LMS)

LMS is a web-based system that allows librarians to manage books. The system provides a user-friendly interface for searching, borrowing, and returning books, managing library accounts, generating reports, and so on.

Features:

i. Catalog Management: LMS allows librarians to add, edit, and delete books. It also has a feature to display a subset of all the books, as there is a large collection of books.
ii. Search and Filtering: Users can search books by entering keywords into a search box and can filter books by various attributes such as name, author, genre, and a short description.
iii. Borrowing and Returning: Users can borrow and return materials by scanning a library card or entering their user ID and password. The system tracks the due date and generates automated reminders for overdue books.
iv. User Management: The system allows librarians to create, edit, and delete user accounts, including student and faculty accounts. The system tracks user borrowing history, fines, and payments, and generates reports on user activity. The user management features are implemented with password hashing and token-based authentication to ensure security.
v. Reports and Analytics: The system generates reports and analytics on library activity, including overdue books, popular books, and user demographics.
vi. Barcode Scanning: The system could be enhanced to allow for barcode scanning, which would simplify the process of borrowing and returning books. Users could scan the barcode on the material and the system would automatically update the status of the item.
vii. Notifications: The system could send notifications to users when books become available, when they are due, or when they are overdue. The system could also send notifications to librarians when books are checked out or returned.
viii. Fine-Grained Permissions: The system could be enhanced to allow different types of users (e.g. librarians, faculty, students) to have different levels of access and control over the system. For example, librarians might have full access to the system, while students might only have access to search and borrow books.
ix. Performance Optimization: The system could be optimized for performance, including minimizing the amount of data transferred between the front-end and back-end, using caching to improve response times, and optimizing the DB schema to reduce query times.
x. Recommendation System: The system could incorporate a recommendation system that suggests materials based on a user's borrowing history, search queries, and other user data. This could help users discover new materials that they might be interested in, and improve the overall user experience.

The LMS is designed to be responsive, meaning that it looks and functions well on various devices, including desktops, laptops, tablets, and smartphones. CSS can be used to adjust the layout and styling of the LMS based on the screen size and resolution of the device being used.

Technology: React for the front end, Node.js for the back end, and MySQL for the database.