PROJECT REQUIREMENTS

Project Name	ToolShare

Functional Requirements

- Allow students and staff/faculty to find, book, check out, return, and share university-owned equipment through a secure online platform.
- Create an inventory database where authorized staff or lab managers can add, update, or remove equipment details such as names, descriptions, categories, faculty/location, serial numbers, images, and availability status.
- Provide a search and filtering feature that allows users to filter equipment by category, faculty/location, availability date range, and keywords.
- Enable a booking system with calendar-based scheduling, including automated conflict detection to prevent overlapping reservations.
- Support a complete booking lifecycle including request submission, approval or rejection, equipment checkout, return, and overdue tracking.
- Allow faculty or lab managers to approve or reject booking requests and manage equipment checkout and return actions.
- Support equipment condition verification by allowing authorized staff to upload “before” and “after” images during checkout and return, with optional damage notes.
- Offer a personal dashboard for students to view their booking history, active requests, and booking statuses.
- Offer an administrative dashboard for faculty and lab managers to manage incoming requests, monitor checked out and overdue equipment and generate usage reports.
- Provide automated email notifications for booking confirmations, approvals, rejections, upcoming reminders, returns, and overdue alerts.
- Provide in-app notifications to inform users of important booking and system events.
- Support future integration with the university’s single sign-on (SSO) system to ensure secure user authentication (planned enhancement).

Technical/Performance Requirements

- Use a modern web technology stack including React for the front end, Node.js with Express for the back end, and a relational database such as MySQL, hosted on a cloud platform such as AWS or similar services.
- Implement secure authentication using JWT and enforce role-based access control (RBAC) to restrict system actions based on user roles (Student, Faculty, Lab Manager).
- Achieve a target system availability of 99% uptime with regular automated database backups.
- Ensure the system is scalable to support a growing number of users, equipment items, and booking requests without significant performance degradation.
- Optimize database queries and indexing to support reporting and analytics features.
- Maintain hosting and operational costs within an estimated budget of $20–50 per month