项目文档

# Functional Requirement

2.1 Asset Registration  
Input: Asset details (name, type, serial number, location, purchase date, cost, supplier)  
Output: Confirmation message and asset ID  
Description: Users can register new assets into the system. Each asset must be assigned a unique identifier.  
2.2 Asset Categorization  
Input: Category name, description  
Output: Category list update  
Description: Assets are grouped into categories for better management and reporting.  
2.3 Asset Allocation  
Input: Asset ID, user ID, allocation date, return date (if applicable)  
Output: Allocation confirmation and status update  
Description: Assets can be allocated to users. The system tracks who has which asset and when it was issued.  
2.4 Asset Return / Deactivation  
Input: Asset ID, reason for deactivation/return  
Output: Status change and audit log entry  
Description: Assets can be returned or marked as inactive. A record is kept for auditing purposes.  
2.5 Reporting and Analytics  
Input: Report type (e.g., asset utilization, depreciation), time range, filters  
Output: Generated report (PDF or Excel)  
Description: The system provides customizable reports on asset usage, availability, and distribution.  
2.6 Approval Workflow  
Input: Request for action (allocation, deactivation, etc.), approval chain  
Output: Approval status and audit trail  
Description: Certain actions require multi-level approvals before execution. The workflow is configurable per organization policy.  
2.7 Data Import and Export  
Input: File (CSV, Excel), data mapping configuration  
Output: Imported records, export file (CSV, Excel, PDF)  
Description: Supports bulk import of asset data and export of reports and asset lists.  
2.8 Role-Based Access Control  
Input: User roles (Admin, Manager, User), permissions  
Output: Access control enforcement  
Description: Different roles have different levels of access based on their responsibilities.  
2.9 Audit Trail  
Input: User activity (login, modification, deletion)  
Output: Log entries with timestamps and user IDs  
Description: All significant user actions are logged for auditing and accountability.  
2.10 Email Integration  
Input: Notification triggers (approval request, due return, system alerts)  
Output: Email sent to designated recipients  
Description: Automated email notifications for key events such as pending approvals or upcoming returns.  
2.11 Asset Maintenance Tracking  
Input: Asset ID, maintenance type (routine, repair), date, cost, technician name  
Output: Updated asset status and maintenance log  
Description: The system tracks all maintenance activities related to an asset. This helps in planning preventive maintenance and tracking expenses.  
2.12 Asset Depreciation Calculation  
Input: Purchase date, cost, depreciation method (straight-line or reducing balance)  
Output: Current book value and depreciation history  
Description: The system calculates the depreciated value of assets over time based on predefined methods and policies.  
2.13 Loan / Temporary Allocation  
Input: Asset ID, user ID, loan period, return deadline  
Output: Temporary allocation record with reminders  
Description: Supports temporary borrowing of assets with automatic reminders for return. Loan requests may require approval depending on policy.  
2.14 Inventory Reconciliation  
Input: Manual inventory data (CSV or scanned input), reconciliation date  
Output: Discrepancy report and updated asset status  
Description: Allows users to compare physical inventory against system records and update accordingly. Generates a report of discrepancies.  
2.15 Multi-Language Support  
Input: User language preference  
Output: UI displayed in selected language  
Description: The system supports multiple languages to accommodate diverse user bases. Language settings are user-specific and configurable.  
2.16 Mobile Access (Optional)  
Input: Mobile app login credentials  
Output: Mobile-friendly interface for key functions  
Description: A mobile version of the system allows users to scan barcodes, view asset details, and receive notifications on-the-go.  
2.17 Barcode / QR Code Integration  
Input: Asset barcode/QR code  
Output: Direct access to asset details via scanning  
Description: Each asset is assigned a unique barcode or QR code. Users can quickly retrieve asset information by scanning it using a mobile device or scanner.  
2.18 Customizable Dashboards  
Input: User role, preferred widgets (e.g., asset count, utilization rate, upcoming returns)  
Output: Personalized dashboard  
Description: Users can customize their dashboards to display key metrics relevant to their role. Admins can configure default dashboards per role.  
2.19 Role-Based Workflow Configuration  
Input: Role name, workflow steps, approvers  
Output: Configured workflow for specific roles  
Description: Different roles can have different approval workflows configured. For example, a "Manager" might approve asset loans, while a "Director" approves deactivations.  
2.20 Data Backup and Restore  
Input: Backup command or schedule, restore request  
Output: Encrypted backup file, restored database  
Description: The system provides automated daily backups of all data. Administrators can manually trigger or restore backups as needed.  
2.21 Audit Report Export  
Input: Time range, audit type (login, modification, deletion)  
Output: Exported audit report (PDF or Excel)  
Description: Allows administrators to export detailed logs of user activity for compliance or internal auditing purposes.  
2.22 Notification Preferences  
Input: User notification preferences (email, in-app, SMS)  
Output: Personalized notifications  
Description: Users can choose how they receive system notifications. Options include email, in-app messages, or SMS alerts (if available).

# External Description

3.1 Database Interface  
Purpose: Store and retrieve asset and user data.  
Interface Type: SQL or NoSQL database (MySQL, PostgreSQL, MongoDB)  
Data Exchange: Structured queries via ORM or direct SQL commands  
3.2 Email Service Interface  
Purpose: Send automated notifications and alerts.  
Interface Type: SMTP server or third-party email service (e.g., SendGrid, Amazon SES)  
Data Exchange: JSON/XML payloads containing notification content  
3.3 Authentication Interface  
Purpose: Provide secure login and session management.  
Interface Type: OAuth 2.0, LDAP, or custom token-based authentication  
Data Exchange: JWT tokens, username/password credentials  
3.4 File Storage Interface  
Purpose: Support import/export of data files.  
Interface Type: Local storage or cloud storage (AWS S3, Google Cloud Storage)  
Data Exchange: File upload/download operations