项目文档

# Functional Requirement

2. Functional Requirements (Updated)  
2.1 Asset Registration  
Input: Asset details (name, type, serial number, category, location, purchase date, value)  
Output: Confirmation of successful asset registration and assignment of a unique asset ID.  
Description: Users can register new assets into the system. Administrators may approve or reject registration requests via workflow.  
2.2 Asset Usage Tracking  
Input: Asset ID, user ID, start/end times of usage  
Output: Updated usage log and status of the asset.  
Description: Track when and by whom an asset is used. Supports concurrent usage validation.  
2.3 Asset Transfer  
Input: Source user ID, destination user ID, asset ID  
Output: Updated ownership record and notification to both users.  
Description: Allows transferring ownership of an asset from one user to another. Requires administrative approval.  
2.4 Asset Return  
Input: Asset ID, user ID, reason for return  
Output: Updated status and location of the asset.  
Description: Records the return of an asset to its designated location or storage facility.  
2.5 Report Analysis  
Input: Time range, asset category, user group  
Output: Summary reports in PDF/Excel format.  
Description: Generates reports on asset usage, availability, transfers, and maintenance history.  
2.6 Permission Allocation  
Input: User ID, role (admin/user), permissions list  
Output: Updated user profile and permission settings.  
Description: Assigns and modifies user roles and permissions for different system functions.  
2.7 Approval Workflow  
Input: Request type (registration, transfer, deletion), requester ID  
Output: Status update and email notification upon approval/rejection.  
Description: Implements a configurable multi-step approval process for sensitive operations.  
2.8 Data Import and Export  
Input: CSV/Excel file with asset or user data  
Output: Imported data or exported file in standard format  
Description: Allows bulk import/export of asset and user data for easy migration or backup.  
2.9 Authentication Integration  
Input: Username, password  
Output: Login success/failure message  
Description: Integrates with existing authentication systems (e.g., LDAP, Active Directory) for secure user login.  
2.10 Email Notification  
Input: Event trigger (approval needed, asset returned, etc.)  
Output: Email sent to relevant users  
Description: Sends automated emails for key events like approvals, reminders, and updates.  
2.11 Asset Maintenance Record  
Input: Asset ID, maintenance date, technician name, issue description, solution  
Output: Updated asset status and maintenance history  
Description: Records each maintenance or repair event to track the health and service history of an asset.  
2.12 Asset Disposal / Scrapping  
Input: Asset ID, disposal reason, disposal date  
Output: Updated asset status as "disposed" and audit trail entry  
Description: Provides a formal process for marking an asset as disposed, ensuring it no longer appears in active inventory while maintaining historical records.  
2.13 Inventory Reconciliation / Stocktaking  
Input: Reconciliation period, warehouse location  
Output: Reconciliation report showing discrepancies between system records and physical stock  
Description: Supports periodic or ad-hoc reconciliation of asset inventory to ensure accuracy and detect missing items.  
2.14 Asset Borrowing / Reservation  
Input: User ID, asset ID, borrowing start/end time  
Output: Updated borrowing log and asset status  
Description: Enables temporary borrowing of assets with scheduled check-out and check-in, preventing conflicts in usage.

# External Description

3. External Interfaces  
3.1 Hardware Interfaces  
None specified; the system is web-based and does not require specific hardware beyond standard servers and client devices.  
3.2 Software Interfaces  
Interface Type Description  
Authentication System LDAP or Active Directory for user authentication.  
Email Server SMTP server for sending notifications.  
Database Relational database (e.g., MySQL, PostgreSQL) for storing asset and user data.  
3.3 Communication Interfaces  
RESTful API for integration with third-party systems (e.g., HR systems for user data).  
Web-based UI accessible via HTTPS.