Feasibility Study- Asset Tracker

1. Asset Panda

Introduction

This cloud-based asset management system is built to help businesses of all sizes keep track of their valuable assets with ease. Whether it's equipment, tools, or digital resources, the platform offers a simple and reliable way to manage everything in one place. Designed for use across different industries, it takes the stress out of asset tracking by providing real-time updates, secure access from anywhere, and tools that make day-to-day operations smoother and more efficient.

Functional Requirements

FR-1: Asset Inventory & Categorization
Allows for comprehensive asset registration, categorization by type, location, or status.

FR-2: Real-Time Tracking
Supports GPS tracking, QR/barcode scanning, and real-time asset updates.

FR-3: Check-in/Check-out System
Enables assignment of assets to users/departments with full audit trails.

FR-4: Lifecycle Management
Includes scheduled maintenance, service history logging, and lifecycle cost tracking.

FR-5: Custom Workflows & Fields
Highly customizable to organizational processes via user-defined fields and workflows.

FR-6: Reporting & Analytics
Offers detailed asset utilization, cost, and maintenance insights through dashboards and reports.

Non-Functional Requirements

NFR-1: Performance
Handles up to 50,000+ assets and 5,000+ concurrent users.

NFR-2: Security
Supports SSO/MFA, full encryption, and audit logging.

NFR-3: Scalability
Cloud-native architecture supports enterprise-scale deployments.

NFR-4: Integrations
RESTful API support for ERP, HR, and helpdesk systems.

Limitations

1. The application does not support creating user groups.
2. Premium pricing models can become cost-prohibitive for smaller organizations.
3. The application does not support issue prioritizing

---

2. Snipe-IT

Introduction

Snipe-IT is an open-source, self-hosted IT asset management system tailored for IT departments and educational institutions that prefer to manage their infrastructure internally. It provides an easy-to-use interface for tracking hardware, software licenses, and other technology assets, helping teams stay organized and in control. With the flexibility to host it on their own servers, organizations benefit from full control over data, customizable features, and cost-effective management without relying on third-party cloud services.

Functional Requirements

FR-1: Asset Lifecycle Tracking
Supports asset tagging, assignment, status updates, and audit logs.

FR-2: Role-Based Access Control
Provides granular permission settings for various user roles.

FR-3: License & Consumables Management
Tracks software licenses and consumables like peripherals and accessories.

FR-4: Depreciation and Cost Tracking
Includes purchase tracking, depreciation scheduling, and warranty alerts.

FR-5:Email Alerts & Notifications
Notifies users of expiring assets/licenses and due check-ins.

Non-Functional Requirements

NFR-1: Performance
Optimized for mid-sized teams; handles CRUD in ~2 seconds.

NFR-2: Security
2FA, SSO (OAuth, SAML), and audit logs supported.

NFR-3: Customizability
Source code is open and supports Docker or CLI updates.

NFR-4: Data Management
Backup and recovery tools available.

Limitations

1. Limited scalability; needs infrastructure tuning for large enterprise use.
2. The application does not support issue prioritizing
3. Focus is mainly on IT assets, not general equipment or field assets.

---

3. Kandji

Introduction

Kandji is a commercial Apple Device Management (MDM) platform built specifically for enterprises that rely on Apple ecosystems, especially those with remote or hybrid teams. It offers powerful tools to automate device setup, enforce security policies, and manage updates, all while providing a seamless experience for end users. Designed with IT administrators in mind, Kandji simplifies the complexities of managing Macs, iPads, iPhones, and Apple TVs at scale, making it easier for businesses to support a distributed workforce while maintaining high standards of compliance and security.

Functional Requirements

FR-1: Zero-Touch Deployment
Integrates with Apple Business Manager for automated onboarding.

FR-2: Security & Compliance
Enforces CIS benchmarks, disk encryption, firewall settings, and more.

FR-3: Automated Remediation
Detects and self-corrects non-compliance via prebuilt or custom scripts.

FR-4: Patch Management
Applies OS and third-party app updates automatically.

FR-5: Role-Based Access & SSO
Integrates with Okta, Google Workspace, Azure AD for identity management.

Non-Functional Requirements

NFR-1: Scalability
Built for high concurrency across thousands of Apple devices.

NFR-2:Usability
Intuitive UI optimized for web and mobile use.

NFR-3:Maintainability
Centralized policy and script management with auto-update support.

NFR-4:Security
Full data encryption, SSO, compliance auditing.

Limitations

1. Apple-only support—no cross-platform compatibility.
2. The application only tracks Apple devices.
3. It cannot track digital assets like API keys, etc.

Functional and Non-Functional Requirements: Asset Tracker

Introduction

This document defines the functional and non-functional requirements for the Asset Management System. The system provides asset lifecycle management, issue tracking, and role-based access control to support organizational resource administration.

Functional Requirements

1. Authorization and Authentication

FR-1.1: User Roles

- The first user created in the system shall be assigned the Manager role.
- All subsequent sign-ups shall be assigned the User role by default.

FR-1.2: Access Management

- The system shall authenticate logins based on user roles: Manager, Admin, or User.
- The system shall enforce role-based access control during session-based interactions.

FR-1.3: Firewall

- Integration with firewall

2. Asset Management

FR-2.1: Assets managed by Admin

- Categorizing based on asset category (physical/digital)
- Categorizing based on asset type (Laptops/Keys/ etc)
- Managing asset statuses.
- Adding / Removal of assets, editing metadata etc
- View all assets or details related to an asset.

FR-2.2: Assets managed by users

- View assigned assets

- Filter and search the assets.

3. Asset Assignment

FR-3.1: Only admins have the ability to assign, reassign or un-assign the assets to users or a group of users.

4. Issue Management

FR-4.1: Reporting Issues

- Both users and admins can report an issues related to an asset

FR-4.2: Resolving the issues

- Both admin and user can update the issues created

FR-4.3: Deleting the issues

- Only the admin can delete an issue.
- The user can raise a request to delete an issue raised by that user.
- Resolved issues can be used to keep track of asset issue history.

FR-4.4: Alerting unresolved issues

- Issues unresolved for a long time should be identified by raising alerts to the admin

5. User Management

FR-5.1: Users managed by admin

- Create a new user.
- Promote a user to an admin.
- View all users in the system.
- Delete a user.

FR-5.2: Users managed by Manager

- Same privileges as admin.
- Demote an admin to a user.
- Can transfer title to another admin and demote themselves.

Non-Functional Requirements

1. Performance

NFR-1.1: Response Time

- The system shall respond to user interactions within 2-3 seconds
- The system shall complete backend operations like asset assignment, issue status updates, and data fetches within 1-2 seconds under normal load.

2. Usability

NFR-2.1: User Interface

- The system shall offer a clean, intuitive user experience
- The system shall support responsive design for accessibility on both desktop and mobile devices

3. Security

NFR-3.1: Authentication & Authorization

- The system shall implement Multi-Factor Authentication (MFA) for all users during login

4. Maintainability

NFR-4.1: Code Quality

- Clean code and design

5. Cost Management

NFR-5.1: Resource Cost Tracking

- The system shall provide modules to track and manage costs related to devices and services

- Admins and Managers shall be able to generate reports on asset-related expenditures.

6. Functional Enhancements

NFR-6.1: Issue Prioritization

- The system shall allow Admins and Managers to set priority levels (Low, Medium, High, Critical) for reported issues

- Prioritized issues shall be displayed in dashboards and reports.

NFR-6.2: Cost Tracking

- Track details of service costs incurred by an asset during its usage.

NFR-6.3: Alerts

- Alerts for when the expiry date or service date are about to be reached for an asset depending on the asset type.

NFR-6.4: Dashboard

- Details about issue statistics, cost analysis, asset status can be present in a detailed dashboard.

NFR-6.5: Integration of AI

- **Help Desk Support**: Integrate AI-powered help desk functionality to automate handling of user queries based a knowledge base,
- **Depreciation Analytics**: Automatically calculate and forecast asset depreciation based on usage patterns, lifespan, and financial models, aiding in better financial planning.

Roles

User

1. Can view and filter/search their assigned assets
2. Can report, edit, and request deletion of issues related to their assets
3. Has no access to asset assignment or system-level user management
4. By default, any user created has a user role and can be promoted later.

Admin

1. Can add, remove, and manage all assets and their categories/statuses.
2. Responsible for assigning and reassigning assets to users.
3. Can add,view, update, delete, and prioritize all reported issues.
4. Manages users: can create, delete, and promote users to admin.
5. Cannot demote admins or transfer manager role.

Manager

1. Has all admin privileges.
2. Can demote admins back to user role.
3. Can transfer the manager role to another admin and self-demote.
4. Acts as the root-admin with ultimate control over the system.

Future Scopes

1. Location Tracking

- **GPS-Based Tracking**: Implement GPS tracking for physical assets, especially mobile or remote ones, to ensure visibility, security, and real-time location reporting.

- **RFID/Barcode Integration**: Use RFID tags or barcode scanning to streamline asset check-in/check-out processes.

2. Asset Reservation

- **Pre-Booking Functionality**: Introduce a reservation system that allows users to pre-book shared assets (e.g., laptops, projectors, rooms).

3. Multi-Organizational Architecture

- **Support for Multiple Organizations**: Evolve the platform into a multi-tenant system that enables multiple organizations or departments to operate independently within the same infrastructure.