

Automation Title

Medical Device Maintenance & Monitoring Automation System

Category

Healthcare Operations / Clinical Engineering / Maintenance Management

Image / Thumbnail

See Folder

Detailed Description

Healthcare facilities rely on dozens(sometimes hundreds) of clinical devices that must be serviced on a routine schedule. When maintenance is missed, devices fail, downtime increases, and compliance audits become more difficult. Traditionally, hospitals rely on manual logbooks or spreadsheets to track service dates, resulting in inconsistencies, missed deadlines, and potential compliance risks.

This automation solves that problem by creating a fully integrated maintenance monitoring system that automatically logs maintenance data, updates device records, checks for overdue or upcoming service dates, sends alerts to clinical engineering teams, and generates a monthly summary report. The goal is to reduce manual work, eliminate oversight, and enable proactive management of clinical equipment.

The system is built around Google Sheets as a simple and accessible database, Make.com as the automation engine, optional vendor email ingestion for external maintenance reports, and a scheduled Google Apps Script that generates month-end summaries. Device data comes from three possible input sources: Fillout forms (staff submissions), vendor emails (external reports), and the daily scheduled workflow (which updates overdue status and alerts).

This automation is ideal for small and mid-sized clinics, biomedical departments, or any organization that uses devices requiring routine maintenance. It ensures accuracy, timeliness, and transparency across all stakeholders and supports compliance reporting. The expected outcome is a centralized, reliable, and automated process that keeps device records current and reduces downtime.

How It Works (Functionality)

Workflow Overview

The automation is made up of **three coordinated flows**:

1. Daily Maintenance Operations (Make Scenario 1) – Scheduled

Trigger: Scheduler (runs once or twice per day)

Actions:

1. Fetch all new maintenance log entries from the *Maintenance Logs* sheet.
2. If *processed* = “no”:
 - If Device ID exists: update the corresponding device’s last serviced date.
 - If Device ID is new: create a new entry in the *Devices* sheet.
3. Recalculate status fields (Active, Due Soon, Overdue).
4. Check for devices that are overdue or due soon.
5. Send Slack notifications to the maintenance team.
6. Mark maintenance log entries as processed.

Output:

Updated Google Sheets + Slack alerts.

2. Vendor Email Processing (Make Scenario 2) – Email Triggered

Trigger: Gmail → Watch Emails (from specific vendors)

Actions:

1. Parse incoming vendor reports.
2. Extract maintenance details (from CSV or plain text).
3. Log extracted data into the *Maintenance Logs* sheet.
4. Scenario 1 later updates the *Devices* sheet using these logs.

Output:

Standardized maintenance logs regardless of vendor format.

3. Monthly Summary (Apps Script) – Scheduled

Trigger: Time-driven script (monthly)

Actions:

1. Scan entire **Devices** sheet.
2. Build:
 - Total device count
 - Overdue count
 - Due soon count
 - Active count
 - Devices serviced this month
 - A table of overdue devices
3. Format results into an HTML email.

4. Send summary to leadership or engineering teams.

Output:

Automated monthly maintenance report.

Tools Required

- **Make.com** — Automation builder
 - **Google Sheets** — Device database & logs
 - **Fillout** — Form submissions for new & existing devices
 - **Slack** — Alert notifications
 - **Gmail** — Incoming vendor maintenance reports
 - **Google Apps Script** — Monthly summary email
 - *(Optional)* Google Workspace admin access for advanced triggers
-

Size of Project

This is a **Large** project.

Why?

- 10 distinct tasks, including form ingestion, device matching, row updates, vendor parsing, Slack alerts, and monthly scripting.
-

Setup Requirements

1. Google Sheets

- Devices sheet
- Maintenance Logs sheet

2. Fillout

- New Device Form
- Maintenance Log Form (prefilled with device ID)

3. Make Integrations

- Google Sheets connection
- Gmail connection
- Slack Bot token connection
- Fillout webhook (optional)

4. Apps Script

- Monthly summary script
- Time-driven trigger setup (Monthly)

5. Security

- No credentials stored in workflow files
- OAuth connections only

Deployment Time Estimate

Without customizations: 2–4 hours

Includes setup of sheets, Make scenarios, Slack alerts, and basic testing.

With customizations (department routing, compliance, vendor variations, notifications):

2–5 days depending on the complexity and approval workflows.

Value Proposition

This automation saves:

- **10–20 hours/month** of manual device log updates
- **3–5 hours/month** compiling monthly reports
- **5–10 hours/month** notifying teams and tracking overdue devices

If a team member's hourly rate is **\$35/hour**, savings per month:

Estimated savings:

$$\approx (20 + 5 + 10) \times \$35 = \$1,225/\text{month}$$

Beyond time, it reduces risk of downtime, improves compliance readiness, and ensures maintenance is always up to date.

Known Limitations

- Vendor formats must be somewhat structured (CSV or clean text).
- Google Sheets may become slower if total rows exceed ~10,000.
- Monthly summary depends on Apps Script, not Make.
- This is a demo setup; a fully HIPAA-compliant system requires additional architecture (no PHI included here).

Testimonials / Use Cases

- Clinics logging maintenance manually can eliminate forgotten entries.
 - Biomedical engineering teams gain real-time visibility.
 - Administrators receive compliance-friendly monthly summaries automatically.
 - Vendor reports can be integrated regardless of structure.
-

Size Classification

Large Automation

(>10 tasks across multiple scenarios + Apps Script)

Estimated ROI

ROI Formula:

$(\text{hours saved} \times \text{hourly labor rate}) - \text{automation tool costs}$

Example:

$(35 \text{ hours} \times \$35/\text{hr}) - \$16 \text{ Make.com plan} = \$1,209 \text{ monthly ROI}$

Version & Updates

v1.0 — Initial release

- Daily Scheduled Workflow

- Vendor Email Parsing
- Slack Alerts
- Monthly Summary Script

v1.1 (Upcoming)

- Multi-department routing
- Device grouping by vendor
- Optional KPI dashboard (Looker Studio)