Simple Data Input CRM System

Category: Internal tools and lightweight CRM

Problem: Managing records across multiple tabs required manual searching, copying, and formatting.

What I Built: Built a single-sheet input interface connected to multiple database tabs with automated search, update, and formatting logic.

Tools Used: Google Sheets, Google Apps Script

Output: Lightweight CRM-style data input system

Simple Data Input CRM System

Google Sheets + Apps Script

Quick backstory

The team needed a fast way to manage records without hopping between multiple tabs, copy pasting fields, and reformatting text every time. So I turned a Google Sheet into a lightweight “mini CRM” where staff can search, update, and submit records in one place.

The problem

This becomes necessary when:

- Records live across multiple tabs or stages (example: Active, Special Case, Closed)
- Updates happen often and by different people
- Formatting needs to stay consistent for reporting
- Manual copy paste increases mistakes and slows the team down

Before automation, the process looked like this:

- Search for the right tab
- Find the record manually
- Copy values into the right columns
- Reformat notes into a consistent style
- Combine multi-field inputs (like tags/diagnoses) into one clean summary field
- Risk updating the wrong row or duplicating a record

The system I built

A single “DATA INPUT” sheet that works like a form, connected to multiple database tabs.

Chronologically, here’s how it works:

1. User searches a record by name
    - The system checks across multiple tabs
    - If found, it loads the record into the input area for editing

2. Input formatting is handled automatically
    - Multi-cell entries (example: tags or diagnoses) get combined into one clean field
    - Multi-line notes are stored in a consistent list-style format
3. User submits updates to the correct tab
    - User selects the target tab (example: Active or Closed)
    - If the record exists, it updates the matching row
    - If it doesn't exist, it creates a new row
4. Quick maintenance actions
    - Clear input fields in one click
    - Delete a record by name across tabs (when needed), with on-screen prompts

The tools used

- Google Sheets (input screen + database tabs)
- Google Apps Script (search, load, combine, submit, update, clear, delete)
- Simple in-sheet prompts/alerts for guardrails

The measurable result

Typical impact for recurring updates:

- Before automation: around 3 to 6 minutes per record (searching, copying, formatting, double-checking)
- After automation: around 30 to 60 seconds (search, edit, submit)

Example monthly savings (conservative):

- 10 record updates per day
- ~3 minutes saved per update
- ~30 minutes saved per day
- ~10 hours saved per month (20 workdays)

Also reduced common issues:

- Fewer duplicates
- More consistent formatting
- Lower chance of updating the wrong tab or row

What this same pattern can support

This same setup can be reused for simple CRMs and ops trackers such as:

- Sales pipeline stages (New, Contacted, Qualified, Closed)
- Client onboarding trackers
- Follow-up lists and reminders
- Issue registers and internal request logs
- Reports and dashboards that pull from predictable, clean sheet structures