



By Wilmer Toro

superservicebros.com

How to Build a Retail Appliance Model & Serial Tag Parser Using AI and n8n

If you've ever done a physical appliance inventory, you already know how painful this part is:

- Crawling behind units
- Taking pictures of model tags
- Manually typing model numbers
- Manually typing serial numbers
- Double-checking for typos
- Realizing later something was missed

When we had to do a full warehouse inventory, this step alone was costing **hours of manual labor** and introducing errors that later caused mismatches in inventory systems.

This automation eliminated that work almost entirely.

The Real Problem This Solves

Retail and warehouse inventory breaks down because:

- Model tags are inconsistent across brands

- Serial numbers are long and error-prone
- Photos are easy to take, but hard to process
- Manual entry creates typos and duplicates
- Staff get fatigued halfway through inventory

The solution isn't "type faster."

The solution is **don't type at all**.

What This Automation Does (Plain English)

Here's what happens automatically:

1. You take a picture of an appliance model tag
2. You upload it into a specific Google Drive folder
3. AI reads the tag like an experienced technician
4. Model and serial numbers are extracted
5. A confidence score is assigned
6. Results are logged into Google Sheets
7. The image is renamed and organized automatically

No manual typing. No copy-paste. No guessing.

Designed for Real Warehouse Conditions

This workflow assumes:

- Blurry photos
- Glare
- Partial labels
- Multiple numbers on the same tag

That's why it uses **multiple AI layers** instead of basic OCR.

Step-by-Step: How the Automation Actually Works

1. Google Drive Folder Trigger

The workflow watches a **specific Google Drive folder**.

Whenever a new image is added:

- The automation triggers immediately
 - No buttons to click
 - No apps to open
- Retail Model Parser

This makes it perfect for:

- Warehouse teams
 - Phones or tablets
 - Multiple people uploading at once
-

2. Image Download & Preprocessing

The image file is downloaded into the workflow so it can be processed by AI tools.

This ensures:

- Full resolution
 - No compression artifacts
 - Better OCR accuracy
-

3. OCR Layer (Text Extraction)

The workflow sends the image to **Mistral OCR**, a dedicated image-to-text engine.

This step:

- Reads all visible text
 - Preserves layout and formatting
 - Outputs structured OCR data
- Retail Model Parser

OCR alone is not enough — it just gives raw text.

4. AI Agent: “Expert Appliance Technician”

Now the real intelligence kicks in.

An AI agent is prompted to behave like:

An appliance technician with 20+ years of experience reading model and serial tags.

Its job:

- Analyze the OCR output
 - Identify **only** the correct model number
 - Identify **only** the correct serial number
 - Ignore irrelevant numbers
 - Assign a confidence score from 1–100
- Retail Model Parser

If the tag is unclear, the AI is explicitly instructed **not to guess**.

5. JSON-Only Output (No Fluff)

The AI must return **only valid JSON**:

```
{  
  "model_number": "",  
  "serial_number": "",  
  "confidence": 100  
}
```

This is critical because:

- It makes automation reliable
 - It prevents random text
 - It allows downstream logic
-

6. Confidence-Based Decision Logic

A **Switch node** checks the confidence score:

- **≥ 70% confidence**
 - Considered reliable

- Image is marked as processed
 - Data is written to Google Sheets
 - **< 70% confidence**
 - Image is flagged
 - Moved to a “Failed” folder
 - Requires human review
- Retail Model Parser

This prevents bad data from contaminating inventory records.

7. Automatic File Renaming

For high-confidence images:

The original image is renamed to:

MODEL - SERIAL

-
- This makes files searchable and auditable later

This small step saves a surprising amount of time.

8. Google Sheets Inventory Logging

The final step logs results into Google Sheets:

- Model
- Serial
- Confidence score
- Direct link to the image

Rows are **appended or updated**, preventing duplicates when the same serial is encountered again

Retail Model Parser

.

This becomes your **live inventory sheet**.

How This Was Used in Real Life

During a warehouse inventory:

- Staff walked the floor taking photos
- No typing required
- No training required
- Inventory populated in real time
- Low-confidence items were reviewed later

What normally takes **hours or days** was reduced to **a fraction of the time**, with fewer errors.

Who This Automation Is For

- Appliance retailers
- Scratch & dent warehouses
- Open-box appliance sellers
- Service companies with inventory
- Anyone doing physical counts

If you've ever typed a serial number twice because you didn't trust the first time — this is for you.

Why This Works So Well

This workflow succeeds because:

- OCR extracts text
- AI applies human judgment
- Confidence controls risk
- Humans review only exceptions

AI does the boring work.

People handle the edge cases.

That's the right balance.

How to Configure the Retail Model & Serial Tag Parser (Step-by-Step)

This automation is designed so you **import a ready-made workflow** and only connect your own accounts. No logic needs to be rebuilt.

What You'll Need Before You Start

Make sure you have:

- An **n8n account** (cloud or self-hosted)
- A **Google Drive account**
- A **Google Sheets account**
- An **AI provider API key** (used for OCR + interpretation)
- A smartphone or tablet for taking photos
- Clear access to appliance model/serial tags

You do **not** need programming skills.

Step 1 — Import the Workflow into n8n

1. Log into your n8n dashboard
2. Go to **Workflows** → **Import**
3. Upload the file `Retail Model Parser.json`
4. Save the workflow
5. Leave it **OFF** for now

Once imported, you'll see a workflow that starts with Google Drive and ends with Google Sheets.

Step 2 — Configure the Google Drive Trigger (Image Intake)

This automation starts when **a photo is added to a specific folder**.

1. Open the **Google Drive Trigger** node
2. Click **Create New Credential** and authorize your Google account

Select or create a folder, for example:

`/Inventory / Model Tags / Incoming`

- 3.
4. Save the node

This folder becomes the *drop zone* for inventory photos. Anyone with access can upload images from their phone.

Step 3 — Create Supporting Drive Folders (Important)

The workflow expects **two additional folders**:

- **Processed / Success**
- **Needs Review / Failed**

Create these folders manually in Google Drive and note their folder IDs.

You'll map these IDs later so the automation can:

- Move clean images automatically
 - Separate low-confidence results for review
-

Step 4 — Connect the OCR Service (Image → Text)

This workflow uses a dedicated OCR step before AI reasoning.

1. Open the **OCR node**
2. Add your OCR service credentials (already supported in the template)
3. Leave all extraction settings as-is

 Do not replace this with basic OCR.
This step preserves layout and improves accuracy on messy tags.

Step 5 — Configure the AI Agent (Critical Step)

This is where the automation “understands” the tag.

1. Open the **AI Agent** node
2. Create or select your AI credentials
3. Review the prompt (recommended, not required)

The AI is instructed to:

- Act like an experienced appliance technician
- Identify **only** the correct model number
- Identify **only** the correct serial number
- Ignore other numbers
- Assign a confidence score
- **Never guess**

 Do NOT remove:

- JSON-only output requirement

- Confidence scoring
- “Do not guess” rules

These are what make the workflow safe.

Step 6 — Review Confidence Threshold Logic

A **Switch / IF node** checks the AI’s confidence score.

By default:

- **≥ 70% confidence**
 - Treated as reliable
 - Continues to Sheets logging
 - Image is renamed and archived
- **< 70% confidence**
 - Image is flagged
 - Moved to the “Needs Review” folder
 - Still logged for visibility

You can adjust the threshold if needed, but **70% works well in real warehouses**.

Step 7 — Configure Automatic File Renaming

High-confidence images are renamed automatically.

Format:

MODEL_NUMBER - SERIAL_NUMBER

Example:

WRF535SWHZ - KX2134567.jpg

This makes:

- Audits easier
- Duplicate detection possible
- Files searchable later

No setup required here unless you want a different naming format.

Step 8 — Connect Google Sheets (Inventory Log)

This is where parsed data is stored.

1. Open the **Google Sheets node**

Create a new spreadsheet, for example:

Warehouse Inventory – AI Parsed

- 2.
3. Add the following recommended columns:
 - Model
 - Serial
 - Confidence
 - Image Link
 - Status

4. Paste the Spreadsheet ID into the node
5. Connect Google credentials

The workflow:

- Appends new rows
 - Updates existing rows if the same serial appears again
-

Step 9 — Test with Real Photos (Very Important)

Before turning it on:

1. Click **Test Workflow**
2. Upload:
 - One clean model tag photo
 - One blurry or partial photo
3. Confirm:
 - Clean photo → logged + renamed + moved to Success
 - Blurry photo → logged + moved to Needs Review
 - No manual typing required

If something fails:

- It's usually a Drive folder ID mismatch
 - Or a Sheets column mismatch
-

Step 10 — Turn the Workflow ON

Once testing passes:

1. Toggle the workflow **Active**
2. That's it

From now on:

- Photos become inventory records automatically
 - Staff only take pictures
 - Humans only review exceptions
-

How to Use This in the Warehouse (Real-World SOP)

1. Walk the warehouse with a phone
2. Take a clear photo of each model tag
3. Upload photos to the intake folder
4. Continue working
5. Review only the flagged items later

No typing. No double-checking serials.

Common Mistakes to Avoid

- Taking photos too far away
- Cutting off part of the label

- Removing the confidence guardrail
- Mixing non-tag images into the folder

The cleaner the photo, the faster the inventory.

Final Reminder

This automation works because it respects reality:

- Tags are messy
- Photos aren't perfect
- Humans shouldn't type serial numbers all day

AI reads.

Automation organizes.

People review exceptions.

That's how you save hours without risking bad data.

Final Thoughts

Inventory shouldn't be a typing contest.

This automation turns photos into structured inventory data automatically, reduces fatigue, eliminates most manual entry, and gives you confidence in your records.

Once you've used it, you'll never want to inventory appliances the old way again.