

Pam Website QC

Pam checks every product page against your source of truth, and flags anything missing, wrong, or out of order, before it goes live.

Deterministic. Fast. No guesswork. · Share-ready overview for the team

TL;DR

- Pam compares a finished product page to **what you said should be on it** and returns a clean pass / fail / fix list in seconds.
- The one thing the team has to build is a **source of truth**: a simple sheet, one row per product, columns for the fields that must be there.
- Pam pulls the page data from **Shopify (read-only)**, so the check is structured and reliable, not screen-scraping guesswork.
- It is **deterministic work**: the comparison is rule-based, so it is cheap, consistent, and runs on every launch.
- To start we need a filled template for **5 to 10 pilot products**. Then Pam QCs every new product before it goes live.

The problem it kills

Manual page QC is slow, and things slip live: wrong price, missing size chart, no fabric content, images in the wrong order, a color or size that never got added. Each one is a return, a support ticket, or a lost sale. Pam catches it in seconds against what you already decided the page should have.

How it works

STEP 1

Give Pam the source of truth

STEP 2

Point Pam at the page

What must be on the page, per product.

The live URL or the Shopify product.

STEP 3

Pam checks field by field

Source vs page, line by line.

STEP 4

Pam returns a QC report

All clear, or a specific fix list.

What you provide (this is the part the team is stuck on)

The unlock is **one structured source of truth**. Simplest version: a Google Sheet, one row per product, columns equal the required fields. Pam reads it. You only fill the fields you care about. **Blank means not checked.**

Recommended columns:

- Product name / handle / live URL
- Price (and compare-at price)
- Variants: every color and size that must exist
- Description: required sections or key phrases
- Images: required count and required shots (front, back, detail, on-model)
- Size chart: required? which chart
- Fabric / material content
- Care instructions
- Fit notes (runs small / large, model height and size worn)
- Tags / collections it must belong to
- Meta title and meta description (SEO)
- Badges (new, bestseller, etc.)

What Pam checks

- **Presence:** is each required field on the page at all?
- **Match:** does the value match the source (price, variants, fabric)?

- **Order:** are images and sections in the expected order?
- **Completeness nudges:** "no size chart found", "only 2 images, you usually have 5", "missing fabric content".

Sample QC report

● PASS · present and correct

● FAIL · missing or wrong

● WARN · mismatch to confirm

● SUGGEST · nice to add

Basic Tee: Black

PASS Price \$24 matches source.

PASS 6 of 6 variants present (S through 3XL).

FAIL Size chart missing.

FAIL Image 3 (back shot) not found. Only front and detail uploaded.

WARN Fabric on page reads "95% cotton / 5% spandex" but source says "100% cotton".

WARN Meta description is 184 characters, over the 160 limit.

SUGGEST Add an on-model image. This style usually has one.

How it is built (note for whoever builds it)

- Pam pulls product data straight from **Shopify via a read-only API** (structured and reliable). Optional: also scan the live page for display-only items like the size-chart widget and badges.
- The comparison is **deterministic and rule-based**. Pam only uses judgment on fuzzy text matches. That keeps it cheap, fast, and consistent.
- Pilot on 5 to 10 products, tune the field list, then scale to every launch.

What we need to start

- 1 The source-of-truth template filled for a handful of pilot products. We provide the sheet.
- 2 Shopify read-only access for the brand. Already in place.
- 3 2 to 3 sample product pages to pilot.

Then Pam QCs every new product before launch, and can re-check live pages on demand.

Phases

PHASE 1

Pilot

Manual trigger, a few products, prove it.

PHASE 2

Every launch

On-demand or scheduled before go-live.

PHASE 3

Full catalog

Catalog audit plus ongoing monitoring of live pages.