

ClaimBack

I build software that finds the money Amazon owes FBA sellers —
\$2,000 to \$8,000 they never claim.

by AI Soft Tech Solution

*I build apps for online sellers. This one reads your FBA reports,
finds every unit Amazon lost, damaged or miscounted -
and builds the claim that gets your money back.*

1. The Gap - Amazon Owes You Money and Never Tells You

Here is how FBA works on paper: you ship 100 units to an Amazon warehouse. Amazon stores them, sells them, ships them, and pays you. Clean, automatic, hands-off.

Here is how it works in practice. Warehouses are enormous and busy. Units get **lost** between the dock and the shelf. Units get **damaged** by forklifts, water and drops. Shipments get **miscounted** - you sent 100, the system received 94, and nobody flags the missing 6. Customer returns get credited to your account for items that never actually came back.

Amazon's own policy says: when Amazon loses or damages **your** inventory, Amazon owes you a **reimbursement**. The policy is real. What is also real: **Amazon does not come and tell you**. There is no email that says "we lost 14 of your units, here is your money." The debt sits silently in your reports until somebody goes looking for it - and most sellers never look.

How much money are we talking about? Industry audits of FBA accounts keep landing in the same range: across a catalog doing about **\$500K a year**, unclaimed reimbursements typically run **\$2,000 to \$8,000** - money already earned, already owed, never claimed. (SellerView cost audit, 2026; SentryKit Amazon margin guide, 2026.)

And there is a second, quieter leak: the **returns processing fee**. Amazon charges roughly **\$2 to \$5 per returned unit** to process it. On a catalog with hundreds of returns a month, that is thousands of dollars a year in fees most sellers never even track - let alone question.

Why does nobody claim this money? Because doing it by hand is miserable. The evidence is spread across a maze of Seller Central reports - inventory ledgers, shipment receipts, return records - that have to be lined up against each other line by line. A thorough manual audit of one month can eat an entire working day. So sellers do the rational thing: they skip it, and donate the money to Amazon's silence.

That is the gap this guide closes. Not with a day of spreadsheet pain - with a small custom app that does the lining-up for you.

2. The App - ClaimBack, Piece by Piece

ClaimBack is a custom app we build for FBA sellers. It is not a subscription dashboard and it is not a generic "seller tool." It does **one job**: find the money Amazon owes you and turn it into filed, tracked claims. Four pieces, working in sequence:

Piece 1 - The Report Puller

ClaimBack starts where the evidence lives: **Seller Central reports**. You download your FBA inventory report and your shipment (receiving) reports - two clicks in Seller Central - and drop the files into the app. (Version 2 connects to Amazon's **SP-API** and pulls them automatically; the manual upload is the honest, working starting point.) The app reads every line: what you shipped, what Amazon says it received, what is in stock now, what was returned.

Piece 2 - The Mismatch Auditor

This is the heart of the app. The auditor lines the reports up against each other and flags every discrepancy, unit by unit:

- **Shipped vs. received**: you sent 100 units on a shipment, Amazon received 94 - 6 units flagged.
- **Lost in the warehouse**: units that were in stock, never sold, never returned to you - and now gone.
- **Damaged by Amazon**: units written off as warehouse damage - reimbursable under Amazon's own policy.
- **Phantom returns**: refunds issued to customers for items your return records show never came back.

Every flag comes with the numbers attached: SKU, shipment ID, dates, quantities. No vibes - evidence.

Piece 3 - The Case Builder

A flag is not a claim. Amazon pays claims that arrive **with proof, in its format**. The Case Builder takes each confirmed mismatch and assembles the claim case for you: the shipment ID, the expected quantity, the received quantity, the dates, and a clear written summary of what Amazon owes and why. You review it, click copy, and file it in Seller Central - or hand the whole pack to your VA.

Piece 4 - The Payout Tracker

Filed is not paid. Claims get approved, partially approved, or denied - and denied claims can often be re-filed with better evidence. The Payout Tracker lists every claim, its status and its amount, and totals the only number that matters: **money recovered**. It also keeps a running tally of your returns processing fees, so the second leak is visible for the first time.

What ClaimBack is NOT - honest limits: it does not invent claims (every flag traces back to your own reports), it does not guarantee Amazon approves every case (nobody honestly can), and it does not replace checking your shipments when they arrive - it makes sure the checking actually happens.

3. Why This Matters More Than Another Sale

Most seller advice is about making **new** money: better ads, better listings, more traffic. All of that costs money and time before it returns anything. Reimbursement money is different in kind:

- **It is already yours.** No ad spend, no ranking battle, no discount. It is your inventory, lost by Amazon, priced at your value.
- **It is pure margin.** A recovered dollar carries no product cost, no shipping cost, no ad cost. It drops straight to the bottom line - see the math in the next section.
- **The leak grows with you.** The more shipments you send and the more returns you process, the more units slip. A catalog that leaks \$2,000 today leaks \$8,000 at four times the volume. Sellers who build the audit habit early keep the money at every size.
- **Unclaimed does not mean unclaimable - yet.** Amazon lets you claim what it owes, but old discrepancies get harder to prove as records age. Sellers who audit monthly claim while the evidence is fresh; sellers who "get around to it" mostly do not.

Think of it this way: you would not let a supplier short-ship you 6 units in every 100 and stay silent. ClaimBack simply makes Amazon play by the same rule.

4. The Business Impact - Run Your Own Numbers

Use the typical ranges from the audits and your own catalog size. The pattern holds at almost every scale:

Your catalog	Typically owed (unclaimed)	Recovering just half
\$100K / year in FBA sales	\$400 - \$1,600	\$200 - \$800 / year
\$500K / year in FBA sales	\$2,000 - \$8,000	\$1,000 - \$4,000 / year
\$1M / year in FBA sales	\$4,000 - \$16,000	\$2,000 - \$8,000 / year

Now translate "recovering half" on a \$500K catalog - **\$1,000 to \$4,000 a year** - into what it would take to earn the same profit from new sales. At a healthy 15% net margin, \$4,000 of pure profit requires roughly **\$26,000 in additional sales**. ClaimBack finds that money in reports you already have, with zero ad spend.

Add the second leak: 100 returns a month at \$2-\$5 per unit in processing fees is **\$2,400 to \$6,000 a year** flowing out untracked. The tracker alone - just seeing it - changes decisions about which products are actually worth selling.

A worked example - one shipment, audited. You sent 300 phone stands in one shipment. Here is what the auditor does with it:

Check	Your records	Amazon's records	Result
Received at warehouse	300 shipped	291 received	9 units flagged: Shipped-Not-Received
Sold (60 days)	291 available	240 sold	Matches - no flag
In stock now	51 expected	47 in stock	4 units flagged: Lost

Customer returns	12 returns	3 refunds, items never returned	3 units flagged: Phantom Returns
------------------	------------	---------------------------------	----------------------------------

Total from one shipment: **16 units flagged**. At a \$14 unit value, that is **\$224 from a single shipment** - found in minutes, from reports you already had. Multiply across a year of shipments and the \$2,000-\$8,000 range stops sounding surprising.

And the cost side? Building ClaimBack with the roadmap in this guide is a one-time effort on a free or near-free tier. Compare that with audit services that take a percentage of everything they recover, forever. Your app, your reports, your money.

5. The Build Roadmap - Step by Step

You do not need to code. You need a **vibe-coding builder** (Lovable is the easiest start; Bolt and Replit work too), your Seller Central reports, and the prompts below. Work through the steps in order.

Step 1 - Download the two reports

In Seller Central: go to **Reports -> Fulfillment -> Inventory** for your FBA inventory report, and **Reports -> Fulfillment -> Shipments / Receive Inventory** for what Amazon recorded receiving. Export the last 3 months as CSV to start. (Do not share these files publicly - they are your business data. The app you build runs on your own account.)

Step 2 - Scaffold the app

Open **Lovable** (lovable.dev) and paste this prompt:

COPY-PASTE PROMPT - scaffold:

```
Build a web app called "ClaimBack" for an Amazon FBA seller.

Layout:
- A top summary bar with 4 numbers: Total Units Flagged, Estimated Amount Owed, Claims Filed, Money Recovered.
- A main table of "mismatches" with columns: SKU, Product Name, Type (Shipped-Not-Received / Lost / Damaged / Phantom Return), Units, Shipment ID, Date, Estimated Value, Status (New / Case Built / Filed / Paid / Denied).
- An upload area where I can drop CSV report files.
- A clean, professional look: white background, dark text, blue accent color.

Data can be sample data for now. I will connect the real report parsing next.
```

Click through the result. If anything looks off, say it in plain English: "Make the table wider" or "Put the upload area at the top." The builder fixes it.

Step 3 - Teach it to read your reports

COPY-PASTE PROMPT - report parsing:

```
Now make the upload real:

1. When I upload my FBA inventory CSV and my shipment receiving CSV, parse both files in the browser.
2. For every SKU, compare: units I shipped vs. units Amazon received vs. units currently in stock.
3. Create a mismatch row for every gap: shipped more than received = "Shipped-Not-Received"; in stock before, gone now, never sold or returned to me = "Lost"; marked warehouse-damaged = "Damaged".
4. Estimated Value = units x my per-unit cost (add a settings field where I enter my average unit cost).
5. Show me how many rows came from real uploaded data vs. sample data. Never mix them silently.
```

Step 4 - Add the Case Builder

COPY-PASTE PROMPT - case builder:

Add a "Build Case" button on every mismatch row. When clicked, generate a claim summary I can copy:

"Shipment [SHIPMENT ID], SKU [SKU]: [UNITS] units were shipped on [DATE] and [RECEIVED] were received by Amazon. The missing [UNITS] units are eligible for reimbursement under the FBA inventory reimbursement policy. Evidence attached: shipment record and inventory ledger."

Save the generated case text on the row, and change its Status to "Case Built". Add a status dropdown so I can mark rows Filed, Paid or Denied, and update the top summary bar totals automatically.

Step 5 - Add the returns-fee tracker

COPY-PASTE PROMPT - fee tracker:

Add a second tab called "Returns Fees". When I upload my returns report CSV, list every return with its processing fee, total the fees by month and by SKU, and highlight the 3 SKUs with the highest total return fees. Show a monthly trend so I can see whether return fees are rising.

Step 6 - Deploy to a live link

In **Lovable**: click **Share -> Publish** - you get a live link like `claimback.lovable.app`. In **Bolt** or **Replit**: use the **Deploy / Publish** button, or connect the project to **Vercel** (vercel.com) or **Netlify** (netlify.com) for a free custom deployment. Open the link on your phone, upload last month's reports, and run your first real audit.

ADVANCE - when the basics work

- **Connect the SP-API.** Amazon's Selling Partner API (developer.amazon.com/sp-api) lets the app pull reports automatically instead of manual CSV uploads. This is the one genuinely technical step - it needs Amazon developer approval - and it is where a custom build earns its keep.
- **Schedule a monthly audit.** Point the app at fresh reports on the 1st of every month so claims are filed while the evidence is fresh.
- **Add your VA.** Give a team member access to the tracker so case-filing becomes a routine job, not your weekend.

6. Troubleshooting - 4 Common Problems

1. Amazon denies a claim

A denial is not the end. The most common reason is thin evidence: re-open the case from the Payout Tracker, attach the shipment record AND the inventory ledger page for the same dates, and re-file. The Case Builder's summary plus both documents is exactly what the second attempt needs. Track which claims were denied and why - patterns in denials are fixable.

2. The numbers do not line up between reports

Reports pulled on different days disagree, because inventory moves while you download. Pull all reports on the **same day**, covering the **same date range**, and re-run the audit. Nine times out of ten the phantom mismatch disappears - and the real ones stay, which is the point.

3. The estimated value looks wrong

The auditor multiplies flagged units by the unit cost in your settings. If that cost is stale, every estimate is off. Update your average unit cost whenever your supplier pricing changes - and remember the estimate is a working number: Amazon reimburses at its own calculated value, which can be higher or lower.

4. Uploading a report does nothing

Seller Central exports in several formats and only the CSV versions parse cleanly. Re-export as **.csv** (not Excel .xlsx), make sure the file is the inventory or receiving report - not a payments report - and try again. If the builder's chat shows an error, paste the error back to it with: "Fix the upload so this CSV parses correctly."

7. Build Checklist

Keep this open while you build. Tick each box as you go:

- ☐ **Step 1:** Downloaded FBA inventory + shipment receiving reports (last 3 months, CSV)
- ☐ **Step 2:** App shell built - summary bar, mismatch table, upload area
- ☐ **Step 3:** Real reports upload and parse - mismatches flagged from MY data
- ☐ **Step 4:** Case Builder generates copy-ready claim summaries; statuses update totals
- ☐ **Step 5:** Returns-fee tracker live - monthly totals and worst SKUs visible
- ☐ **Step 6:** Published - live link opens on my phone; first real audit run
- ☐ **Habit:** Monthly audit scheduled - new reports in, new claims out
- ☐ **Bonus:** First claim filed in Seller Central from a ClaimBack case

8. What's Next

You now know exactly where FBA money hides and how a small app drags it into the light: pull the reports, flag the mismatches, build the cases, track the payouts. An hour to build, a few minutes a month to run - against \$2,000 to \$8,000 a year that most sellers simply donate.

Want it built for you instead? This is what we do at AI Soft Tech Solution: I build custom apps for online sellers and businesses - reimbursement finders, research tools, dashboards, automation - built around how your business actually runs. Comment CLAIM on the post where you found this guide, or message the page, and let's talk about the tool your business needs.