

SAMPLE

Harbourline Supply Co. and the Tally system are fictional. This is a specimen of the report DigiCoyote delivers at the end of a discovery phase, built from patterns we see repeatedly in inherited codebases. No real client information appears here.

DIGICOYOTE SOFTWARE • VANCOUVER, BC

DISCOVERY REPORT

Taking over Tally

An assessment of the order and inventory system Harbourline built in-house, what it would take to make it safe to keep running, and what the first phase of that work costs.

PREPARED FOR

Dana Kirsch, Operations Director
Harbourline Supply Co.

SYSTEM

Tally, v3 (internal)

DISCOVERY WINDOW

2 – 8 September 2026

REFERENCE

DC-2026-0914-HRB

CURRENCY

All amounts in Canadian dollars

01 Summary

Tally works. Twenty-two people use it every day and it carried roughly \$4.1M of orders last year. That is a real achievement and it is why this report recommends repairing the system rather than replacing it.

It is also not safe to keep running as it stands. The application checks who you are in the browser and then trusts the browser, so every order record in the system can be read or altered by anyone who can reach the site and open developer tools. The database has never been backed up. Nobody has a record of how the schema reached its current shape, because it was changed by hand.

None of this is unusual for software built the way Tally was built, by a capable person solving a real problem under time pressure with AI assistance. The tools are very good at producing something that works on the first try and largely silent about the things that only matter on the four hundredth day. Tally is on roughly its eight hundredth.

CRITICAL FINDINGS	HIGH FINDINGS	MEDIUM FINDINGS	RECOMMENDATION
3 Exploitable now, no mitigation in place	3 Causing loss or error today	4 Compounding with growth	Repair Rebuild is not warranted. The domain model is sound.

We propose a four week stabilisation phase at a fixed price of **\$18,000 CAD**, described in section 9. It closes every critical finding, puts backups and tests in place, and leaves Harbourline with a system that can be worked on safely. It does not add features. That comes after.

02 What we examined

Harbourline gave us read access to the Tally repository, a copy of the production database with customer names removed, and four hours with Marcus Oyelaran, who built the system and still maintains it.

- **The codebase.** 41,200 lines across 214 files. Full git history, 1,106 commits, all authored by one person.
- **The running system.** We stood Tally up locally against a copy of the production data, then tested the deployed instance from the outside as an unauthenticated visitor.
- **The database.** Schema, indexes, row counts, query plans for the twelve slowest pages.
- **The operational picture.** How it deploys, who can deploy, what is monitored, what happens when it breaks.

We did not perform a formal penetration test, and this report is not one. Section 11 sets out what that would add and when we would recommend it.

03 The system today

Tally is a Next.js application deployed on Vercel, talking to a Postgres database hosted on Supabase. Order entry, inventory adjustments, purchase orders and the weekly reconciliation all live in it. Two staff also use a spreadsheet that reads from the same database directly.

The important structural fact is the one the diagram below shows in red: the browser holds a credential that grants unrestricted database access, and a meaningful amount of business logic runs in the browser where the user can change it.

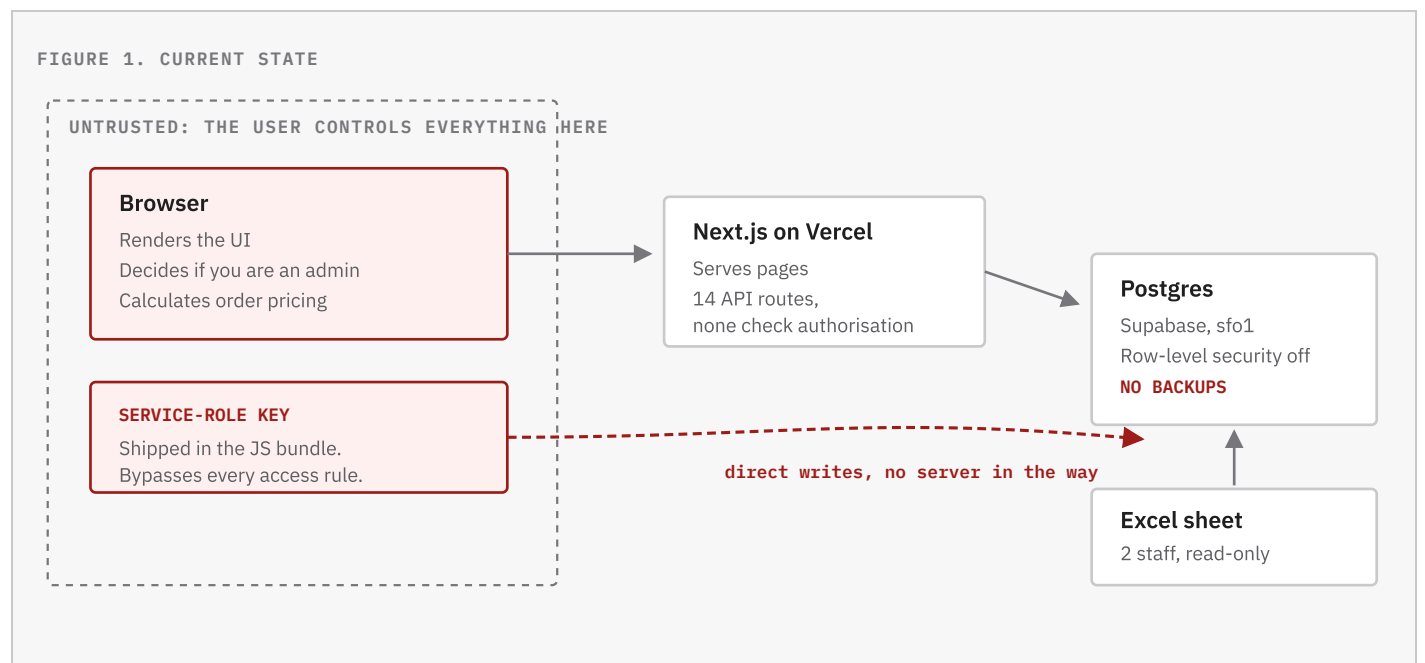


FIG 1 Everything inside the dashed boundary is under the user's control. Two things cross it that should not: a database key that bypasses all access rules, and the decision about whether you are an administrator.

04 What works

A takeover report that lists only faults is not an honest one, and it would lead you to the wrong decision. These are the things Tally gets right, and they are the reason a rebuild would be wasteful.

- ✓ **The domain model is correct.** Orders, order lines, holds, partial shipments and backorders are modelled the way Harbourline actually trades. This is the part that is genuinely hard, and it usually takes a rebuild two attempts to get right. Tally has it.
- ✓ **The interface is better than most commercial equivalents.** We watched three staff enter orders. They are fast, and they are fast because the screens were designed by someone who does the job. That is not something a replacement would inherit.
- ✓ **Postgres is the right database,** and the schema, whatever route it took to get here, is reasonable. Types are sensible. Money is stored in integer cents, which is more than many production systems manage.
- ✓ **It is one codebase, in git, with real history.** Every change is attributable. This makes the repair work far cheaper than it would otherwise be.

05 What is wrong

Ten findings, ranked. Severity reflects what happens if the finding is exploited or triggered, weighed against how likely that is in the next twelve months.

ID	FINDING	SEVERITY	PHASE
F1	API routes perform no server-side authorisation. Any visitor can read or modify any order.	CRITICAL	1
F2	The Supabase service-role key is bundled into client JavaScript, granting unrestricted database access.	CRITICAL	1
F3	No database backups are configured. A deletion or corruption today is unrecoverable.	CRITICAL	1
F4	Schema is changed by hand through the Supabase console. No migrations, no record, no way to reproduce the database.	HIGH	1
F5	Order pricing is implemented in five places. Three of them disagree on volume-break rounding.	HIGH	2
F6	No automated tests and no CI. Every change is verified by hand, or not at all.	HIGH	1
F7	Fourteen dependencies carry published security advisories, four rated high.	MEDIUM	1
F8	No error tracking or uptime monitoring. Failures are discovered when a staff member telephones.	MEDIUM	1
F9	The order list issues one query per row. It takes 9.4 seconds at current volume and is getting slower.	MEDIUM	2
F10	Deployment is manual from a single laptop. No other person can currently ship a fix.	MEDIUM	1

FIGURE 2. FINDINGS BY LIKELIHOOD AND IMPACT

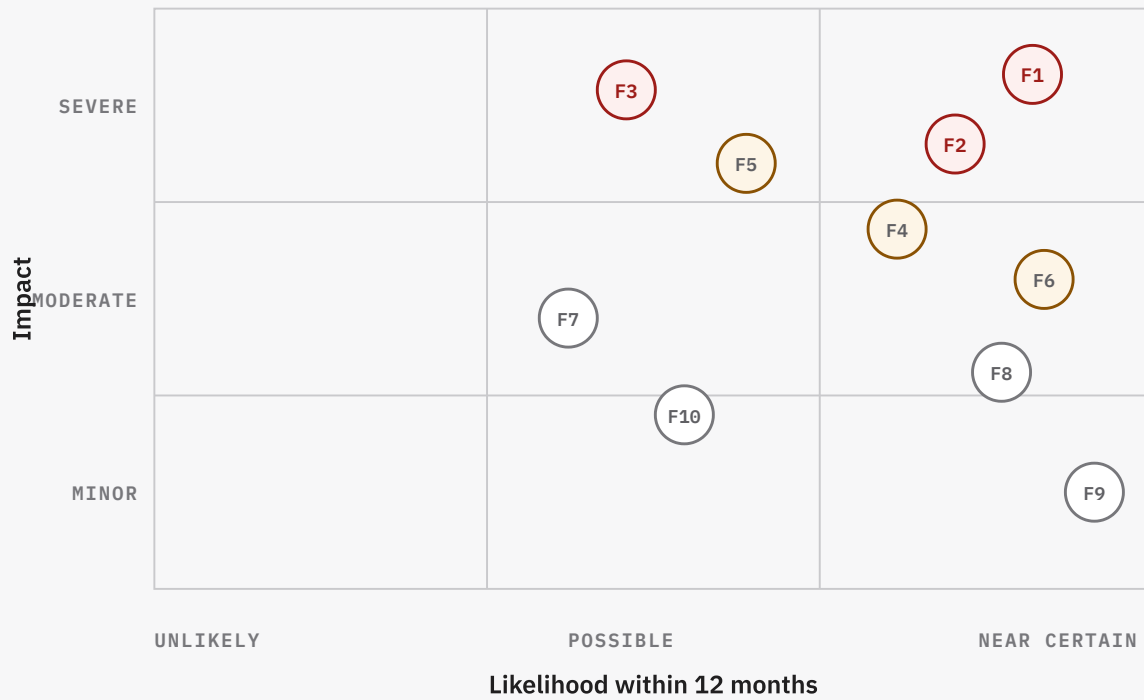


FIG 2 F1, F2 and F3 sit in the upper right because they need no attacker skill and no unusual circumstance. F9 is plotted as near certain because it is not a risk, it is already happening every day.

06 The four that matter most

F1 API routes perform no server-side authorisation

Tally decides what you are allowed to do in the browser. The React code checks a `role` value and hides the buttons you should not see. The API routes behind those buttons check nothing at all.

We confirmed this from an ordinary browser, signed out, in under ten minutes. We were able to read the full order table and change a price on a test record we created for the purpose. We made no other modifications and have supplied Marcus with the request log.

WHAT THIS MEANS

Anyone who can reach the Tally URL can read every customer, every price and every margin in the business, and can alter orders. There is no log that would tell you it happened.

F2 The service-role key ships to the browser

Supabase issues two keys. The anonymous key is meant for browsers and is constrained by access rules. The service-role key is meant to stay on a server and deliberately bypasses those rules. Tally uses the service-role key, and because the code that uses it runs in the browser, the key is in the JavaScript bundle that every visitor downloads.

This is a common outcome. The anonymous key stops working the moment row-level security is switched on without policies written, and swapping in the service-role key makes the error go away immediately.

WHAT THIS MEANS

The key must be treated as public and rotated. Anyone who has visited Tally since deployment may hold a copy that grants full database access, independent of the application.

F3 There are no database backups

Supabase takes backups on paid plans. The Harbourline project is on the free plan, where that feature is absent. No external dump has ever run. The database has one copy of every order Harbourline has taken since March 2024.

WHAT THIS MEANS

A mistaken delete, a bad migration or an account compromise ends the business's order history. Recovery is not slow, it is not possible.

F5 Pricing is implemented five times and the copies disagree

Volume-break pricing appears in the order form, the quote preview, the PDF generator, the reconciliation export and a scheduled job. Three of the five round the break differently. On a 240-unit order at the 200-unit break we reproduced a \$46.80 spread between what the screen showed, what the PDF said and what the export billed.

WHAT THIS MEANS

Some invoices have been wrong. We cannot say how many without a reconciliation against the accounting system, which we would scope separately. This is the finding most likely to have cost money already.

07 How we fix it

The repair is structural rather than cosmetic, and it is smaller than it sounds. Every critical finding has the same root: there is no server between the user and the database. Put one there, and F1, F2 and the direct-write path close together.

Tally already runs on Next.js, which can execute code on the server. The work is to move the queries that currently run in the browser into server routes, put a single authorisation check in front of them, and take the service-role key out of the bundle. The screens do not change. Staff will not notice.

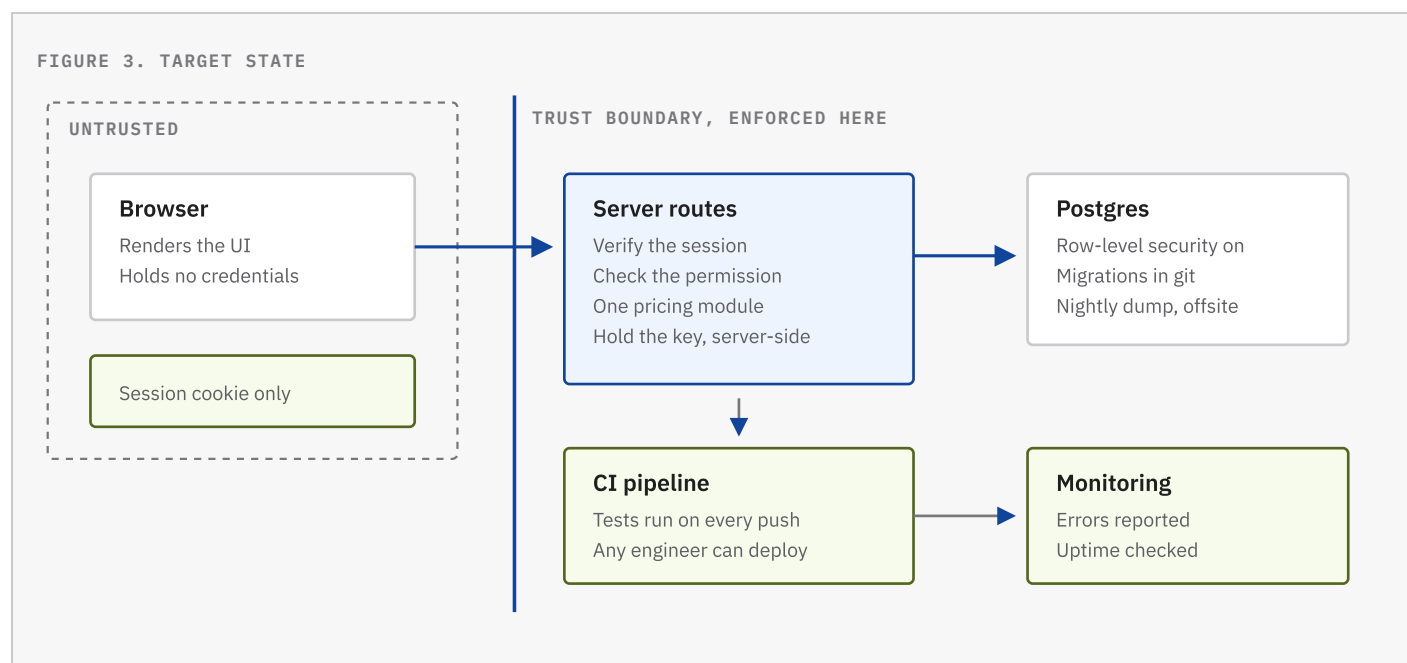


FIG 3 The same application, with the trust boundary moved to where it belongs. The browser keeps a session cookie and nothing else. Every path to the database runs through one authorisation check.

Sequencing, and why it is in this order

Backups come first, on day one, before anything else is touched. We will not modify a system we cannot restore. Credential rotation follows, because the exposed key stays dangerous until it is replaced. Then the authorisation work, then tests and CI, which is what makes every subsequent change cheaper.

08 The plan

Three phases. Only the first is quoted as a fixed price, because only the first is fully understood. We will not price work we have not scoped, and section 9 explains what that means for your budgeting.

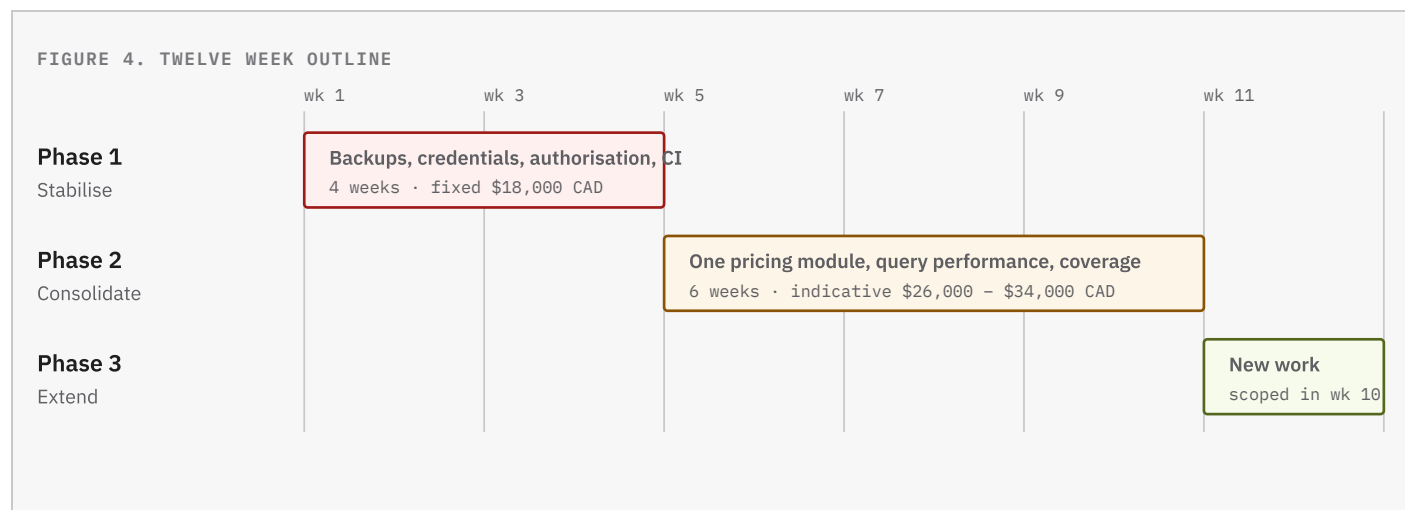


FIG 4 Phase 2 is priced as a range because its shape depends on what Phase 1 uncovers once tests exist. We scope and fix that price in week 10, before you commit to it.

Phase 1, Stabilise: weeks 1 to 4

Closes F1, F2, F3, F4, F6, F7, F8 and F10. At the end of it Tally is safe to run, safe to change, and recoverable. No new features.

Phase 2, Consolidate: weeks 5 to 10

Closes F5 and F9. Collapses the five pricing implementations into one, reconciles the historic discrepancy, and fixes the query patterns that make the order list slow. Test coverage extends from the critical paths to the business rules.

Phase 3, Extend: week 11 onward

Whatever Harbourline actually wants next. This is where the system starts returning value again rather than absorbing repair. We scope it with you in week 10.

09 **What Phase 1 costs**

Fixed price, quoted against deliverables rather than time. If the work takes longer than we expect, that is our problem, not yours. All amounts in this report are Canadian dollars.

DELIVERABLE	CLOSES	AMOUNT
Automated nightly backups, offsite, with a tested restore	F3	\$1,500
Rotate exposed credentials, remove them from the bundle and history	F2	\$1,800
Move queries server-side, add one authorisation layer, enable row-level security	F1	\$3,600
Baseline the schema and introduce migrations	F4	\$3,000
CI pipeline, first test suite over the order and pricing paths	F6, F10	\$3,600
Error tracking and uptime monitoring, alerting to your team	F8	\$1,500
Patch dependencies carrying advisories	F7	\$1,800
Handover documentation and an operations runbook	-	\$1,200
Phase 1 total		\$18,000

QUOTED: PHASE 1, STABILISE

\$18,000 CAD

Fixed price, four weeks from start. Invoiced 50% on commencement and 50% on delivery. Valid for 30 days from the date of this report. Excludes third-party subscription costs, itemised in section 11.

The discovery phase that produced this report is a separate engagement, fixed at \$5,000 for the deliverable and already invoiced. It is not credited against Phase 1. Discovery stands on its own: the findings above are yours to act on, with us or without us, and they hold whether or not you proceed.

10 Next steps

1

Configure a backup tonight

Do this whether or not you engage us. It takes about twenty minutes on the Supabase paid tier and it is the single largest risk in this report. We will walk Marcus through it at no charge.

2

Decide on Phase 1 by 30 September

That is the date this quote expires. A decision either way is fine; we would rather have a clear no than a quiet maybe.

3

Give us a working session with Marcus

Two hours in week one. He knows things about Tally that are not in the code, and the repair goes faster with him in the room than around him.

4

Nominate who receives alerts

Monitoring is only useful if it reaches somebody. We need a name and an address before week four.

5

Consider a reconciliation of historic invoices

Separate from Phase 1 and scoped separately. Finding F5 means some invoices were wrong. You may want to know how many before a customer tells you.

11 Assumptions and exclusions

This report assumes

- Harbourline owns the Tally source outright, with no third-party contribution under separate terms.
- Production data volumes remain within an order of magnitude of what we measured.
- Tally continues to run on Vercel and Supabase. Moving hosts is out of scope and we see no reason to.
- Marcus remains available for questions during Phase 1.

Phase 1 excludes

- New features of any kind.
- Third-party subscription costs. Phase 1 requires the Supabase paid tier at roughly \$34 CAD per month and an error tracking service at roughly \$36 CAD per month. Both are billed to Harbourline directly. Both vendors price in US dollars, so the Canadian amount moves with the exchange rate.
- A formal penetration test. We recommend one after Phase 2, performed by a firm other than the one that wrote the code, which is to say not us. We can help you brief it.
- Reconciliation of historic invoices affected by F5.
- Staff training. The interface does not change, so we do not believe any is needed.

On the finding evidence

Our testing of F1 was limited to a record we created for the purpose and was carried out with written authorisation from Dana Kirsch dated 2 September 2026. No customer data was read, modified or retained. The full request log has been supplied separately and is not reproduced here.