Production Readiness Review

Copperline invoice-financing platform

This is a sample. Copperline, its system, the findings and the prices are invented to show what the review delivers. No client data is used.

Reviewed by	Everton Almeida, InflexDev
Review window	`5 business days`
Scope	Web app, API, payout job, onboarding, deploys
Verdict	Not ready for real customers yet. Ready after Milestones 1 and 2.

1. Summary

Copperline works well in a demo and handles its happy path. Four of the six findings would hurt real customers or real money within the first weeks of launch. All of them are fixable in two milestones without a rewrite.

ID	Finding	Severity	Fixed in
F-01	**Any signed-in user can read other companies' invoices.** Data access	Critical	M1
F-02	**A retried payout can pay a supplier twice.** Payouts	High	M1
F-03	**API keys for the payment provider are stored in the repository.** Secrets	High	M1
F-04	**Edited onboarding forms change the meaning of past answers.** Intake	High	M1
F-05	**Balances are never checked against the provider or the bank.** Reconciliation	Medium	M2
F-06	**Deploys run from one laptop and there is no rollback plan.** Operations	Medium	M2

2. Where data and money move

Every path that carries customer data or money, traced end to end. The risk column points to the finding.

Path	Runs through	Risk
Supplier uploads an invoice	Web app → API → database → document storage	F-01: no check that the invoice belongs to the user's company
Invoice is approved and paid out	Admin app → payout job → payment provider → supplier's bank	F-02: retries are not deduplicated
Provider reports the payout result	Payment provider → notification endpoint → database	F-05: no daily match against provider and bank
New company onboards	Onboarding form → API → database	F-04: answers stored without the form version

How this review was done

Meetings	Two working sessions with the founder and the engineer who built the prototype: goals, users, money flows and launch plans.
Code and accounts	Read access to the repository, the cloud account, the payment provider's test environment and the deploy setup. Nothing was changed.
Tracing	Each path above followed from the screen to the database or the bank, with AI-assisted code search and every finding confirmed by hand.
Report	Findings ranked by severity, then turned into a fixed-scope proposal priced per milestone.

3. Findings in detail

F-01 · Any signed-in user can read other companies' invoices.

Severity: Critical · Area: data access · Fixed in M1

Observed	The invoice endpoint loads a record by its ID and checks only that the user is signed in. Changing the ID in the address returns invoices from other companies, including bank details.
Why it matters	One curious customer can see every competitor's receivables. It is a breach to report and the first thing a partner bank's security review would test.
Fix	Scope every query to the user's company at the data layer, not in each screen. Add a test per endpoint that tries another company's ID.

F-02 · A retried payout can pay a supplier twice.

Severity: High · Area: payouts · Fixed in M1

Observed	The payout job retries when the provider is slow to answer. Each retry sends a new request with no unique key, so the provider treats it as a new payout.
Why it matters	Money leaves twice, and getting it back depends on the supplier's goodwill.
Fix	One unique key per payout, sent with every attempt and stored with the payout, so a retry returns the first result. Add a test that forces a timeout.

F-03 to F-06 follow the same format in the full report: what was observed, why it matters, the fix and the milestone.

4. Fixed-scope proposal

Each milestone is demoed and accepted before it is invoiced. Scope changes are agreed in writing as a new milestone.

Milestone	Time	Price
M1 · Lock down data and money paths F-01 to F-04. Tenant checks on every query, idempotent payouts, secrets moved to a vault and rotated, versioned forms with a backfill.	2 weeks	$9,000
M2 · Reconciliation and operations F-05 and F-06. Daily reconciliation with an exceptions queue, automated deploys from the repository, a tested rollback and a runbook.	2 weeks	$7,500
Ship and handover Production cutover, architecture notes, recorded walkthrough and 30 days of support.	1 week	included
Build total	**5 weeks**	**$16,500**
Review fee already paid, credited		−$2,500
Due across the milestones		**$14,000**

Illustrative prices. Real proposals are priced from the findings and the outcome they protect.

5. What you own at the end

- The repository, in your GitHub organization
- Cloud and provider accounts in your company’s name
- Architecture notes, a decision log and a runbook
- A recorded walkthrough of the system
- Rotated credentials; my access removed on request

Everton Almeida

Reviewed by · info@inflexdev.com