

Verdict: not launch-ready yet. Three blockers, two of them in payments and data security. None require a rebuild.

What I found, worst first

CRITICAL

Your secret keys are sitting in the code

What it means for you: Anyone who sees your code can use your Stripe and database credentials, and bots scan public code for exactly this.

The fix: Move every key into environment variables, and rotate anything that was ever in the code.

CRITICAL

Orders complete without charging the card

What it means for you: A customer can check out, see a success screen, and never be charged, so you deliver for free.

The fix: Complete the charge server-side and confirm it in Stripe before the order is marked paid.

HIGH

The database has no uniqueness rules

What it means for you: The same account can be created twice, which quietly corrupts logins, billing, and reporting.

The fix: Add unique constraints and required fields so duplicate and empty records are rejected.

MEDIUM

Nothing tells you when it breaks

What it means for you: When something fails in production, your users find it first, and you hear about it as a complaint.

The fix: Add error tracking with alerts so a failure pages you, not your customer.

LOW

Test payment details left in checkout

What it means for you: Minor on its own, but it shows the payment flow was never finished for live mode.

The fix: Remove test artifacts and switch to live keys at launch.

Fastest path to launch

Close the two criticals first: get the keys out of the code, then fix the charge so money moves. Add the database rules next, then error alerts. The test-mode cleanup happens at launch. None of this is a rebuild, it is a short, ordered list of fixes.