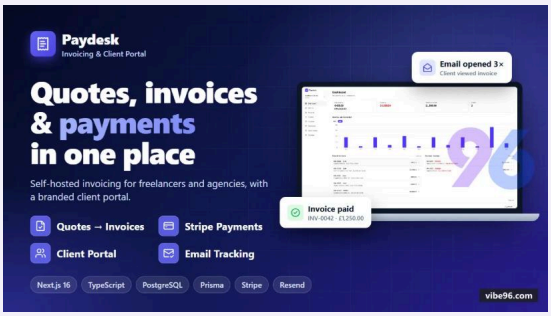


VIBE96 QUALITY REPORT

Paydesk: Invoicing & Client Portal for Freelancers and Agencies

Self-hosted invoicing for freelancers and agencies: quotes, invoices, Stripe payments, reminders and a client portal.

Product ID	**B0367-72103**
Builder	**Rafsan Ahmed Raj**
Version reviewed	**1.0.0 · released 18 Nov 2026**
Stack	**Next.js 16 (App Router) + React 19 · TypeScript 5 on Node.js 20.9+ · PostgreSQL 14+**
Built with	**Claude Code**
Review period	**10 Nov 2026 – 18 Nov 2026 · 1 round of changes**

76 / 100

Passed with 5 warnings

Pass mark 75 and no failing check.

Scores

Security	77
Code structure	84
Docs & completeness	67
Dependency health	72
Build & deploy	72

✓ **AI code audit**
6 checks · 10 Nov 2026

✓ **Installed & run**
18 Nov 2026

✓ **Demo vs delivered**
18 Nov 2026

✓ **Final sign-off**
18 Nov 2026

At a glance

- ✓ Installed and run from the delivered code — 8 of 8 checks answered yes.
- ✓ Live demo compared with the delivered code — 7 of 7 checks answered yes.
- ! 10 of 15 promised items delivered (setup guide, example config and the rest).
- ✓ Security scored 77/100 with no failing check.
- ! 5 open warnings — explained in part 1.

Check this report is genuine. Scan the code or open vibe96.com/report/QR-26-00145. The online copy always shows the latest status.

PART 1

AI code audit

The full source archive was read by our AI auditor against each check. It reads the code — it does not run it; running it is part 2.

CHECK	WHAT WE FOUND	SCORE	RESULT
Secrets & credentials	The regex candidates are mostly Zod schemas, env-var references, and placeholders (whsec_..., empty .env.example values). The only real concern is a hardcoded demo account password in a project file, also published in the README, which would ship to buyers as a weak default if the demo account is seeded in production.	88	Warning
Dependency vulnerabilities	package-lock.json exists, so installs are reproducible, and no known-critical package could be confirmed from the digest. The authentication layer depends on a beta build (next-auth 5.0.0-beta.32), and almost every dependency uses a caret range. Several versions listed (next 16.4.3, react 19.3.0, nodemailer ^10.3.1) are beyond what I can verify, so they should be checked with npm audit.	72	Warning
Authorisation & access control	Passwords are bcrypt-hashed (cost 12), sessions are invalidated after a password reset, and tokens are stored hashed. The digest shows only auth actions, not the ~15 other action files or the API routes (export, pdf, logo, cron, webhooks) where ownership checks matter most. Several unauthenticated-by-design endpoints and some design gaps are flagged below.	72	Warning
Database access rules	The visible code uses Prisma with parameterized queries and hashed tokens and passwords. No injection path or unscoped delete appears in the excerpts. The schema, migrations and most services were omitted, so foreign keys, indexes, cascades and per-workspace scoping are unverified.	70	Warning
Code structure & duplication	The project follows a clear folder structure (app, components, server/actions, server/services, lib/validators) that matches CLAUDE.md, with business logic kept in services rather than views. Weaknesses are limited to a few large files and some hard-coded constants that could move to configuration, based on the excerpts available.	84	Pass
Build reproducibility	The project has a single Next.js app with a lockfile, pinned Prisma 6, Prisma migrations, a Dockerfile, compose file, and setup docs, so a clean build looks plausible. However, the digest shows a few inconsistencies: a source import with no visible file, a tests folder that is not in the tree, and unusual dependency versions that may not resolve.	72	Warning

Open warnings

Problems that did not stop the listing. They stay on the product page until a later version fixes them.

Warning 1 **No real credentials found; a demo password is hardcoded in source and displayed on the login page**

The regex candidates are mostly Zod schemas, env-var references, and placeholders (whsec_..., empty .env.example values). The only real concern is a hardcoded demo account password in a project file, also published in the README, which would ship to buyers as a weak default if the demo account is seeded in production.

Warning 2 **Prisma-only data access looks safe, but schema, tenancy and indexes could not be verified from the digest**

The visible code uses Prisma with parameterized queries and hashed tokens and passwords. No injection path or unscoped delete appears in the excerpts. The schema, migrations and most services were omitted, so foreign keys, indexes, cascades and per-workspace scoping are unverified.

Warning 3 **Lockfile present, but pre-release Auth.js beta and caret ranges; versions could not be checked against advisories**

package-lock.json exists, so installs are reproducible, and no known-critical package could be confirmed from the digest. The authentication layer depends on a beta build (next-auth 5.0.0-beta.32), and almost every dependency uses a caret range. Several versions listed (next 16.4.3, react 19.3.0, nodemailer ^10.3.1) are beyond what I can verify, so they should be checked with npm audit.

Warning 4 **Mostly reproducible; a few manifest and doc gaps need verification before a clean-machine build is certain**

The project has a single Next.js app with a lockfile, pinned Prisma 6, Prisma migrations, a Dockerfile, compose file, and setup docs, so a clean build looks plausible. However, the digest shows a few inconsistencies: a source import with no visible file, a tests folder that is not in the tree, and unusual dependency versions that may not resolve.

Warning 5 **Auth basics are solid, but most routes and actions could not be inspected, so access control is only partly verified**

Passwords are bcrypt-hashed (cost 12), sessions are invalidated after a password reset, and tokens are stored hashed. The digest shows only auth actions, not the ~15 other action files or the API routes (export, pdf, logo, cron, webhooks) where ownership checks matter most. Several unauthenticated-by-design endpoints and some design gaps are flagged below.

Requirement list

What the builder agreed to deliver before uploading — 10 of 15 met.

✓	Complete source for every component	Met
✓	No real credentials in the archive	Met
!	Builds from a clean machine	Partial
✓	Database script or migrations	Met
!	Authentication and roles on every mutating endpoint	Partial
!	Pinned dependencies without critical CVEs (nice)	Partial
✓	Documentation set (nice)	Met
!	Working live demo	Partial
✓	README.md	Met
✓	HowToSetup.md	Met
✓	CLAUDE.md (or AGENTS.md) (nice)	Met
✓	CHANGELOG.md (nice)	Met
✓	.env.example	Met
✓	a project file	Met
!	a project file (nice)	Missing

PART 2

Human review

Reviewers from the Vibe96 team in Dubai worked through each question below; a final reviewer signed it off.

Stage 1 — Installed and run from the delivered code

Completed 18 Nov 2026

A reviewer installed the product on a clean machine, following only the guide that ships with it.

QUESTION	ANSWER	REVIEWER'S NOTE
Packages restored without errors	Yes	
Solution builds clean	Yes	
Database created from the shipped script or migrations	Yes	
First sign-in works with the documented steps	Yes	
Setup guide is accurate and complete	Yes	
Every module in the feature list opens and works	Yes	
No real secrets in the archive	Yes	
Branding and company details can be changed without code	Yes	

Stage 2 — Live demo compared with the delivered code

Completed 18 Nov 2026

A second reviewer checked that what the listing and demo show is what the buyer receives.

QUESTION	ANSWER	REVIEWER'S NOTE
Demo URL is reachable	Yes	
Demo credentials work	Yes	
Demo runs the same version as the archive	Yes	
All five differentiators are visible in the demo	Yes	
Screenshots match the demo	Yes	
Nothing in the demo is missing from the code	Yes	
Datasheet claims are accurate	Yes	

Changes asked for during review

None — approved on the first round.

Sign-off: approved for sale on 18 Nov 2026 with 5 warnings shown publicly.

What this report covers — and what it doesn't

- **It covers** version 1.0.0 exactly as the buyer downloads it: the source archive, the setup guide and the live demo on the dates above.

- **It is not** a penetration test, a performance or load test, or a legal review of third-party package licences.
- **Scores are a point in time.** Libraries age; a later version is audited again and gets its own report.
- **Your setup matters.** Hosting, configuration and changes made after purchase are outside this review.
- **How the score is made:** each AI check is scored 0–100 and weighted (security counts most). A listing needs 75 or more, no failing check, and the human review passed.
- **Always try the live demo** before buying, and ask the builder questions from the product page.