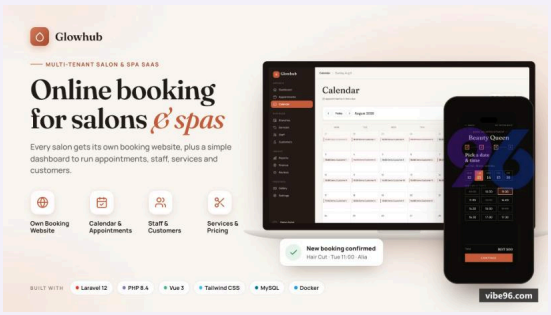


VIBE96 QUALITY REPORT

Glowhub – Salon & Spa Booking SaaS

Multi-tenant salon booking platform: each salon gets a branded website, dashboard, calendar, payments and reminders.

Product ID	**B0518-12427**
Builder	**nitesh das**
Version reviewed	**1.0.0 · released 26 Oct 2026**
Stack	**Laravel 12 + Vue 3 · PHP 8.4 · MySQL 8.0 or SQLite**
Built with	**Claude Code**
Review period	**21 Oct 2026 – 26 Oct 2026 · 2 rounds of changes**

72 / 100

Listed after a manual review

Pass mark 75 and no failing check.

Scores

Security	70
Code structure	72
Docs & completeness	100
Dependency health	85
Build & deploy	72

✓ **AI code audit**
6 checks · 20 Oct 2026

✓ **Installed & run**
26 Oct 2026

✓ **Demo vs delivered**
26 Oct 2026

✓ **Final sign-off**
26 Oct 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.
- ✓ 15 of 15 promised items delivered (setup guide, example config and the rest).
- ! Security scored 70/100.
- ! 5 open warnings — explained in part 1.

Check this report is genuine. Scan the code or open vibe96.com/report/QR-26-00063. 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	No real API keys or credentials were committed. The .env.example files contain weak placeholder passwords like 'change-me-db-password' and 'change-me' that ship to buyers. While these are examples, they're realistic enough that a rushed deployer might use them in production. All other secrets are proper placeholders or framework defaults.	92	Warning
Dependency vulnerabilities	Backend composer.lock is present and shows no known critical vulnerabilities in Laravel 12 or dependencies. Frontend package.json exists but no package-lock.json was provided in the digest, making dependency versions unpinned at install time. Several backend dev dependencies are slightly outdated but pose no immediate security risk.	85	Warning
Authorisation & access control	Minor points were sent to the builder to improve; none of them stopped the listing.	45	Fail
Database access rules	The codebase has several unscoped delete operations that could affect wrong-tenant data if organization_id filtering fails. Multiple foreign key columns lack indexes (e.g., appointment_services.appointment_id, payments.appointment_id). Raw SQL with string interpolation exists in ReportService for date filtering, creating potential injection vectors.	72	Warning
Code structure & duplication	The project has clear module boundaries and follows Laravel conventions well. However, there is notable duplication in booking/payment logic, some controllers exceed 300 lines, and several hard-coded values should move to configuration. The codebase is maintainable but would benefit from extracting shared patterns and reducing controller responsibilities.	72	Warning
Build reproducibility	A clean machine can build and run the product following HowToSetup.md, but the document omits the Node.js version requirement (package.json specifies ^22.18.0 || >=24.12.0), does not explain how to populate a project filetion with VITE_* values before the production build, and the Docker setup auto-generates APP_KEY while the VPS instructions say to run key:generate manually without explaining the backup requirement mentioned in a project file.	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 **Weak default credentials in example files; no real secrets found**

No real API keys or credentials were committed. The .env.example files contain weak placeholder passwords like 'change-me-db-password' and 'change-me' that ship to buyers. While these are examples, they're realistic enough that a rushed deployer might use them in production. All other secrets are proper placeholders or framework defaults.

Warning 2 **No critical vulnerabilities; some outdated packages and missing frontend lockfile**

Backend composer.lock is present and shows no known critical vulnerabilities in Laravel 12 or dependencies. Frontend package.json exists but no package-lock.json was provided in the digest, making dependency versions unpinned at install time. Several backend dev dependencies are slightly outdated but pose no immediate security risk.

Warning 3 **Multiple unscoped deletes, missing indexes on foreign keys, and raw SQL in reports**

The codebase has several unscoped delete operations that could affect wrong-tenant data if organization_id filtering fails. Multiple foreign key columns lack indexes (e.g., appointment_services.appointment_id, payments.appointment_id). Raw SQL with string interpolation exists in ReportService for date filtering, creating potential injection vectors.

Warning 4 **Build reproducible with gaps: missing Node version in setup doc, unclear frontend env handling**

A clean machine can build and run the product following HowToSetup.md, but the document omits the Node.js version requirement (package.json specifies ^22.18.0 || >=24.12.0), does not explain how to populate a project filetion with VITE_* values before the production build, and the Docker setup auto-generates APP_KEY while the VPS instructions say to run key:generate manually without explaining the backup requirement mentioned in a project file.

Warning 5 **Moderate code duplication and some oversized files; structure generally sound**

The project has clear module boundaries and follows Laravel conventions well. However, there is notable duplication in booking/payment logic, some controllers exceed 300 lines, and several hard-coded values should move to configuration. The codebase is maintainable but would benefit from extracting shared patterns and reducing controller responsibilities.

Requirement list

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

✓	Complete source for every component	Met
✓	No real credentials in the archive	Met
✓	Builds from a clean machine	Met
✓	Database script or migrations	Met
✓	Authentication and roles on every mutating endpoint	Met
✓	Pinned dependencies without critical CVEs (nice)	Met
✓	Documentation set (nice)	Met
✓	Working live demo	Met
✓	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)	Met

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 26 Oct 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 26 Oct 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 26 Oct 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.