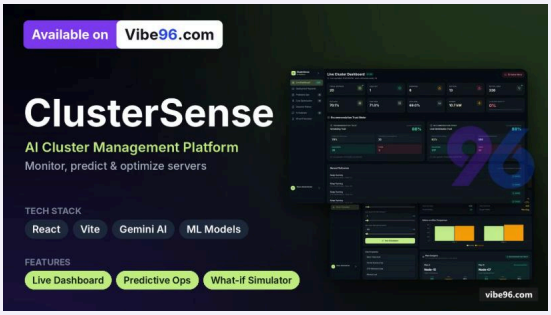


VIBE96 QUALITY REPORT

ClusterSense AI – Intelligent Cloud Infrastructure Management

Admin dashboard for hosting providers: ML node scheduling,failure prediction,cost tips and a Gemini-optional AI copilot.

Product ID	**B1151-99934**
Builder	**Radoanul Arifen**
Version reviewed	**1.0.0 · released 25 Nov 2026**
Stack	**FastAPI 0.111 + React 18 (Vite 5, Tailwind 3) · Python 3.11 (backend) + JavaScript (React 18 frontend) · PostgreSQL 15 (SQLAlchemy 2.0 ORM)**
Built with	**OpenAI Codex, Claude Code**
Review period	**11 Nov 2026 – 25 Nov 2026 · 2 rounds of changes**

76 / 100

Passed with 5 warnings

Pass mark 75 and no failing check.

Scores

Security	80
Code structure	84
Docs & completeness	73
Dependency health	58
Build & deploy	72

✓ **AI code audit**
6 checks · 11 Nov 2026

✓ **Installed & run**
24 Nov 2026

✓ **Demo vs delivered**
24 Nov 2026

✓ **Final sign-off**
25 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.
- ! 11 of 15 promised items delivered (setup guide, example config and the rest).
- ✓ Security scored 80/100 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-26-00153. 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	All flagged candidates are CHANGE_ME placeholders, environment variable reads, hashed-password column names or localStorage token reads. No API keys, private keys or live connection strings appear in the digest. Compose enforces required secrets via ${VAR:?} so no weak defaults ship.	**96**	Pass
Dependency vulnerabilities	The frontend has a package-lock.json, but the backend has only a requirements.txt with direct pins and no lock for transitive packages (the README admits no audit was done). Several pinned versions have known advisories (python-jose, python-multipart, fastapi/starlette, scikit-learn, passlib/bcrypt incompatibility), which is a moderate-risk finding, not a confirmed critical production-path exploit.	**58**	Warning
Authorisation & access control	The visible auth code hashes passwords (passlib/bcrypt), and a test asserts every /api/ route except login and /me depends on require_admin; AGENTS.md states the same rule. However, only auth.py was provided, so the per-route guards in the 12 other routers could not be verified directly. Several smaller gaps remain: the test only checks direct dependencies, there is no login throttling, tokens are stored in localStorage, and the single-role model has no ownership checks.	**72**	Warning
Database access rules	The visible code (auth.py) uses parameterized SQLAlchemy ORM queries, and AGENTS.md forbids SQL concatenation. However, schema.sql, models.py, seed.sql and most API/service files were not in the excerpts, so foreign keys, indexes, cascade rules and any unscoped updates or deletes cannot be confirmed. The access model is a single administrator workspace guarded by require_admin, so per-user scoping is not required.	**72**	Warning
Code structure & duplication	The backend separates API routers, models, services and ML code, and the frontend splits pages, components and the API client, so the layout is navigable. The main costs are a duplicated configuration template and shared authorization logic living in the auth router, where every other router has to import it. File sizes and function lengths could not be fully verified from the excerpts provided.	**84**	Warning
Build reproducibility	Manifests, Dockerfiles and compose reference folders that exist, Node/Python versions are consistent, and a lockfile pins frontend packages. However, HowToSetup.md was not provided for verification of step order, Python transitive dependencies are unlocked, and some Docker details (nginx proxy, schema/seed ordering, healthcheck port) cannot be confirmed from what was shipped.	**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 Routers are split by domain, but config is duplicated and auth dependencies sit inside a router module

The backend separates API routers, models, services and ML code, and the frontend splits pages, components and the API client, so the layout is navigable. The main costs are a duplicated configuration template and shared authorization logic living in the auth router, where every other router has to import it. File sizes and function lengths could not be fully verified from the excerpts provided.

Warning 2 **Admin-only design with tests, but most route files unseen; some public/health endpoints and login hardening gaps**

The visible auth code hashes passwords (passlib/bcrypt), and a test asserts every /api/ route except login and /me depends on require_admin; AGENTS.md states the same rule. However, only auth.py was provided, so the per-route guards in the 12 other routers could not be verified directly. Several smaller gaps remain: the test only checks direct dependencies, there is no login throttling, tokens are stored in localStorage, and the single-role model has no ownership checks.

Warning 3 **Auth queries are safe ORM calls, but most DB code and the schema were not provided, so it is only partly verifiable**

The visible code (auth.py) uses parameterized SQLAlchemy ORM queries, and AGENTS.md forbids SQL concatenation. However, schema.sql, models.py, seed.sql and most API/service files were not in the excerpts, so foreign keys, indexes, cascade rules and any unscoped updates or deletes cannot be confirmed. The access model is a single administrator workspace guarded by require_admin, so per-user scoping is not required.

Warning 4 **Python deps pinned but several have known CVEs and no lockfile; npm lockfile present but ranges are caret-based**

The frontend has a package-lock.json, but the backend has only a requirements.txt with direct pins and no lock for transitive packages (the README admits no audit was done). Several pinned versions have known advisories (python-jose, python-multipart, fastapi/starlette, scikit-learn, passlib/bcrypt incompatibility), which is a moderate-risk finding, not a confirmed critical production-path exploit.

Warning 5 **Mostly reproducible Docker/native build; setup doc unseen, Python deps unlocked, a few coherence gaps**

Manifests, Dockerfiles and compose reference folders that exist, Node/Python versions are consistent, and a lockfile pins frontend packages. However, HowToSetup.md was not provided for verification of step order, Python transitive dependencies are unlocked, and some Docker details (nginx proxy, schema/seed ordering, healthcheck port) cannot be confirmed from what was shipped.

Requirement list

What the builder agreed to deliver before uploading — 11 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)	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 24 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 24 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 25 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.