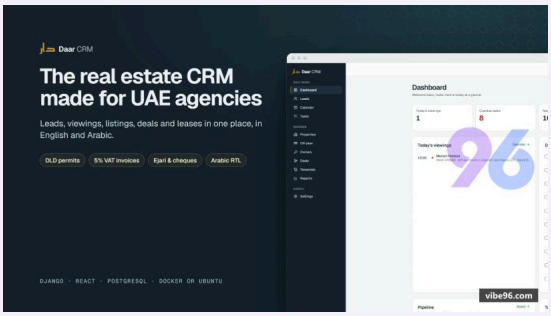


VIBE96 QUALITY REPORT

Daar CRM: Real Estate CRM with Viewing Calendar

Real estate CRM for UAE agencies: leads, viewings, listings, deals, tenancies and reports in English and Arabic.

Product ID	**B0612-83704**
Builder	**Bablu Mia**
Version reviewed	**1.0.0 · released 28 Oct 2026**
Stack	**Django 6.1.2 + React 19.3.1 · Python 3.12 · PostgreSQL 16**
Built with	**Claude Code**
Review period	**24 Oct 2026 – 28 Oct 2026 · 1 round of changes**

90 / 100

Passed with 5 warnings

Pass mark 75 and no failing check.

Scores

Security	93
Code structure	78
Docs & completeness	100
Dependency health	85
Build & deploy	92

✓ **AI code audit**
6 checks · 24 Oct 2026

✓ **Installed & run**
28 Oct 2026

✓ **Demo vs delivered**
28 Oct 2026

✓ **Final sign-off**
28 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 93/100 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-26-00078. 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 credentials were committed. The .env.example files contain weak placeholder passwords ('change-me') that could be deployed unchanged. E2E test code uses hardcoded passwords, which is acceptable for test fixtures. All secret handling follows best practices with environment variables.	95	Warning
Dependency vulnerabilities	Both frontend (package.json) and backend (requirements.txt) dependencies are pinned with lockfiles present. No known critical vulnerabilities detected in the declared versions. Several packages are 6-12 months behind latest stable releases (Django 6.0.9 vs 6.1.x, React 19.3.1 is current, DRF 3.18.3 vs 3.19.x), which is normal maintenance lag. The frontend uses exact versions in package-lock.json. Backend pins exact versions in requirements.txt.	85	Warning
Authorisation & access control	All mutating endpoints require authentication with JWT tokens. Role-based permissions are enforced via Django REST Framework's permission classes. Password storage uses Django's [hidden] hasher. However, several read endpoints (property details, owner details, project details, tenancy details) lack ownership checks, allowing any authenticated user to read any record by I	92	Warning
Database access rules	No SQL injection or unscoped deletes found. Django ORM is used correctly throughout with parameterized queries. Schema migrations are complete and can build from empty. Minor issues: some foreign keys lack indexes (e.g. Deal.property, Viewing.lead), and a few cascade rules could be more explicit (SET_NULL without null=True would fail).	92	Warning
Code structure & duplication	The project has good module boundaries with clear separation between frontend features, backend apps, and API layers. However, there is notable duplication in test setup code, some large test files mixing multiple concerns, and hard-coded values scattered across test helpers that should be centralized configuration.	78	Warning
Build reproducibility	The project is well-structured with consistent dependencies and proper Docker configuration. Both frontend and backend can be built from source. The main issues are minor: HowToSetup.md is missing from the digest (referenced but not shown), the backend .env.example shows DB_PASSWORD as 'change-me' without guidance on generation, and the initial_setup command's password handling could be clearer in documentation.	92	Pass

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 credentials were committed. The .env.example files contain weak placeholder passwords ('change-me') that could be deployed unchanged. E2E test code uses hardcoded passwords, which is acceptable for test fixtures. All secret handling follows best practices with environment variables.

Warning 2 **Dependencies mostly current; a few outdated packages but no known critical vulnerabilities**

Both frontend (package.json) and backend (requirements.txt) dependencies are pinned with lockfiles present. No known critical vulnerabilities detected in the declared versions. Several packages are 6-12 months behind latest stable releases (Django 6.0.9 vs 6.1.x, React 19.3.1 is current, DRF 3.18.3 vs 3.19.x), which is normal maintenance lag. The frontend uses exact versions in package-lock.json. Backend pins exact versions in requirements.txt.

Warning 3 **Strong authentication and role checks; minor ownership gaps on read endpoints**

All mutating endpoints require authentication with JWT tokens. Role-based permissions are enforced via Django REST Framework's permission classes. Password storage uses Django's [hidden] hasher. However, several read endpoints (property details, owner details, project details, tenancy details) lack ownership checks, allowing any authenticated user to read any record by I

Warning 4 **Database access is safe; minor schema gaps in foreign key indexes and cascade rules**

No SQL injection or unscoped deletes found. Django ORM is used correctly throughout with parameterized queries. Schema migrations are complete and can build from empty. Minor issues: some foreign keys lack indexes (e.g. Deal.property, Viewing.lead), and a few cascade rules could be more explicit (SET_NULL without null=True would fail).

Warning 5 **Moderate code duplication and some oversized test files; generally well-structured**

The project has good module boundaries with clear separation between frontend features, backend apps, and API layers. However, there is notable duplication in test setup code, some large test files mixing multiple concerns, and hard-coded values scattered across test helpers that should be centralized configuration.

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 28 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 28 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 28 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.