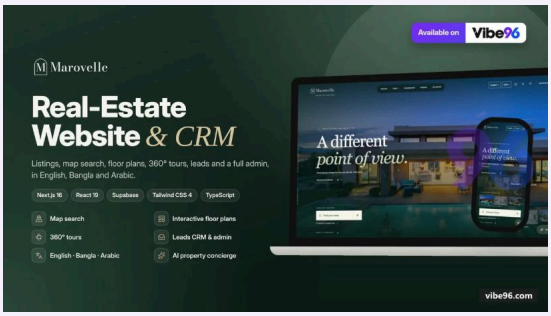


VIBE96 QUALITY REPORT

Marovelle – Real-Estate Website & CRM (Next.js, Supabase)

Real-estate website and CRM: map search, floor plans, 360° tours, payment plans, leads inbox. English, Bangla, Arabic.

Product ID	**B1050-15891**
Builder	**Md. Sazzad Hossain**
Version reviewed	**1.0.0 · released 12 Nov 2026**
Stack	**Next.js 16 (App Router) with React 19 · TypeScript 5.9 on Node.js 22.13+ · PostgreSQL 15+ on Supabase (hosted or self-hosted)**
Built with	**Claude Code**
Review period	**5 Nov 2026 – 12 Nov 2026 · 3 rounds of changes**

82 / 100

Passed with 4 warnings

Pass mark 75 and no failing check.

Scores

Security	80
Code structure	84
Docs & completeness	60
Dependency health	90
Build & deploy	82

✓ **AI code audit**
6 checks · 5 Nov 2026

✓ **Installed & run**
12 Nov 2026

✓ **Demo vs delivered**
12 Nov 2026

✓ **Final sign-off**
12 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.
- ! 9 of 15 promised items delivered (setup guide, example config and the rest).
- ✓ Security scored 80/100 with no failing check.
- ! 4 open warnings — explained in part 1.

Check this report is genuine. Scan the code or open vibe96.com/report/QR-26-00124. 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 regex candidates are env var reads, form fields, translation strings, or test placeholders; no live API keys, tokens or connection strings were found. The only concern is a weak, publicly documented default demo password (hardcoded fallback) that ships to buyers and could be left active on a production deployment.	88	Warning
Dependency vulnerabilities	package.json pins every dependency to an exact version (no ranges, wildcards or 'latest'), and the Dockerfile installs with --frozen-lockfile against pnpm-lock.yaml, so a lockfile exists. No dependency has a known critical vulnerability that I can identify; a few items could not be verified from the digest and are worth tidying.	90	Pass
Authorisation & access control	The project relies on Supabase row-level security, a proxy for sign-in redirects, and documented role helpers, with unit tests for role access and open-redirect protection. However, the API route handlers (reservations, revalidate, concierge, tour, Stripe webhook) and admin layout guards were not in the excerpts, so their authentication and ownership checks could not be confirmed. A few demo-related items need attention before production.	72	Warning
Database access rules	The app uses the Supabase query builder with Zod validation and row-level security on every table, and no string-built SQL appears in the visible excerpts. However, the migration files and most data-access code were not provided, so foreign-key indexes, cascade rules and scoping of deletes/updates could not be verified. Some structural concerns remain around the schema duplicating migrations and an RLS-bypassing secret key used in server paths.	80	Warning
Code structure & duplication	The project separates features, lib, components and app routes clearly, and the docs map the layout. The main structural costs are very large files (a 159 KB Bangla dictionary, a 134 KB Arabic dictionary, a 106 KB global stylesheet) and a few duplicated secret-handling and Supabase-client patterns that could be centralised.	84	Warning
Build reproducibility	The project pins exact dependency versions, declares Node >=22.13 and pnpm via packageManager, and ships a coherent multi-stage Dockerfile and compose file with a setup profile. Gaps are minor: some referenced files (proxy.ts, scripts/, supabase/migrations, pnpm-lock) could not be confirmed in the digest, and the database/setup order depends on docs not shown.	82	Pass

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 shared demo password default ships in code and e2e helpers**

All regex candidates are env var reads, form fields, translation strings, or test placeholders; no live API keys, tokens or connection strings were found. The only concern is a weak, publicly documented default demo password (hardcoded fallback) that ships to buyers and could be left active on a production deployment.

Warning 2 **Supabase client with RLS and migrations; schema excerpts not visible, so key indexes and cascades are unverified**

The app uses the Supabase query builder with Zod validation and row-level security on every table, and no string-built SQL appears in the visible excerpts. However, the migration files and most data-access code were not provided, so foreign-key indexes, cascade rules and scoping of deletes/updates could not be verified. Some structural concerns remain around the schema duplicating migrations and an RLS-bypassing secret key used in server paths.

Warning 3 **Well-organised feature modules; some oversized files and a few hard-coded values to move into config**

The project separates features, lib, components and app routes clearly, and the docs map the layout. The main structural costs are very large files (a 159 KB Bangla dictionary, a 134 KB Arabic dictionary, a 106 KB global stylesheet) and a few duplicated secret-handling and Supabase-client patterns that could be centralised.

Warning 4 **Authz design looks sound (RLS, roles), but most route handlers were not visible so cannot be fully verified**

The project relies on Supabase row-level security, a proxy for sign-in redirects, and documented role helpers, with unit tests for role access and open-redirect protection. However, the API route handlers (reservations, revalidate, concierge, tour, Stripe webhook) and admin layout guards were not in the excerpts, so their authentication and ownership checks could not be confirmed. A few demo-related items need attention before production.

Requirement list

What the builder agreed to deliver before uploading — 9 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	Partial
!	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 12 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 12 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

Round 1: changes requested on 10 Nov 2026; the builder fixed them and uploaded again.

Sign-off: approved for sale on 12 Nov 2026 with 4 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.