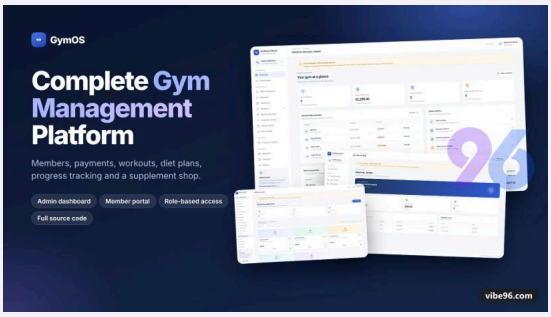


VIBE96 QUALITY REPORT

GymOS – Complete Gym Management & Member Fitness Platform

React/TypeScript gym management app with member portal, for single-site gym owners and front-desk staff.

Product ID	**B0539-76647**
Builder	**Code Dash**
Version reviewed	**1.0.0 · released 6 Nov 2026**
Stack	**React 18.3.1 with Vite 6.4.5 · TypeScript 5.9.3 · PostgreSQL 16 (via Supabase in connected mode)**
Built with	**OpenAI Codex, Claude Code**
Review period	**29 Oct 2026 – 6 Nov 2026 · 2 rounds of changes**

78 / 100

Passed with 5 warnings

Pass mark 75 and no failing check.

Scores

Security	82
Code structure	82
Docs & completeness	80
Dependency health	68
Build & deploy	68

✓ **AI code audit**
6 checks · 29 Oct 2026

✓ **Installed & run**
6 Nov 2026

✓ **Demo vs delivered**
6 Nov 2026

✓ **Final sign-off**
6 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.
- ! 12 of 15 promised items delivered (setup guide, example config and the rest).
- ✓ Security scored 82/100 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-26-00104. 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 three secret-scan candidates (App.tsx:117, authService.ts:56, authService.ts:83) are function parameters and a Supabase Auth call passing user-entered passwords, not hardcoded secrets. .env.example uses only placeholders (YOUR_PROJECT_ID, YOUR_SUPABASE_PUBLISHABLE_KEY), and docs and test READMEs use YOUR_LOCAL_PASSWORD. The only caveat is that not every file was visible in the digest (for example the seed SQL and the supabase.ts client), so the clean result covers the excerpts shown.	97	Pass
Dependency vulnerabilities	All runtime and dev dependencies in a project file are pinned to exact versions, with no wildcards or ranges, which is good. However, the file tree shows no package-lock.json (it may have been omitted as a lock file from the digest), and the Dockerfile copies only package.json and runs `npm install`, so builds are not reproducible. I could not confirm any known critical vulnerability from the listed versions.	68	Warning
Authorisation & access control	GymOS is a static React SPA with no custom API endpoints. All authorization rests on Supabase Auth and PostgreSQL RLS. Passwords are handled only by Supabase Auth, and the sign-in and profile restore flow is sound. However, the schema and RLS policies were not included in the excerpts, and the tests ran only against a Postgres stub. Demo mode has no real authentication, and the client-side role checks are UX only.	72	Warning
Database access rules	The visible client code uses only the Supabase query builder and Supabase Auth, with no string-built SQL and no unscoped deletes or updates. The schema files (a project file, seed.sql, cloudStorage.ts) were not included in the excerpts, so foreign keys, indexes, cascade rules and N+1 patterns could not be verified. A documented 116-check RLS test suite is a positive sign, but it is only evidence and was not run.	78	Warning
Code structure & duplication	The source is split sensibly into pages, features, lib and components, and the auth service is kept separate from the pure auth helpers. The main weakness is that the root and Project/ trees each carry copies of the same docs, schema and compose files, which will drift. Most of this is a packaging issue rather than a code-quality problem.	82	Warning
Build reproducibility	The Vite/React project has pinned dependencies, engines and a coherent Dockerfile, so a clean machine can likely build it. However the archive duplicates the whole project at root and in Project/, lacks a committed lockfile, and docs reference files and a Dockerfile that are not visible, leaving setup ambiguous.	68	Warning

Open warnings

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

Warning 1 **Dependencies exactly pinned, but no lockfile in the submission and installs use npm install**

All runtime and dev dependencies in a project file are pinned to exact versions, with no wildcards or ranges, which is good. However, the file tree shows no package-lock.json (it may have been omitted as a lock file from the digest), and the Dockerfile copies only package.json and runs `npm install`, so builds are not reproducible. I could not confirm any known critical vulnerability from the listed versions.

Warning 2 **No injection seen in visible code; schema and RLS contents not provided, so completeness and indexing are unverified**

The visible client code uses only the Supabase query builder and Supabase Auth, with no string-built SQL and no unscoped deletes or updates. The schema files (a project file, seed.sql, cloudStorage.ts) were not included in the excerpts, so foreign keys, indexes, cascade rules and N+1 patterns could not be verified. A documented 116-check RLS test suite is a positive sign, but it is only evidence and was not run.

Warning 3 **No server routes; authz relies on Supabase RLS, which is untested live and only partly visible in this digest**

GymOS is a static React SPA with no custom API endpoints. All authorization rests on Supabase Auth and PostgreSQL RLS. Passwords are handled only by Supabase Auth, and the sign-in and profile restore flow is sound. However, the schema and RLS policies were not included in the excerpts, and the tests ran only against a Postgres stub. Demo mode has no real authentication, and the client-side role checks are UX only.

Warning 4 **Mostly reproducible, but duplicated trees, missing lockfile and doc/reference mismatches create gaps**

The Vite/React project has pinned dependencies, engines and a coherent Dockerfile, so a clean machine can likely build it. However the archive duplicates the whole project at root and in Project/, lacks a committed lockfile, and docs reference files and a Dockerfile that are not visible, leaving setup ambiguous.

Warning 5 **Modular layout is sound, but docs, schema and compose files are duplicated across root and Project/**

The source is split sensibly into pages, features, lib and components, and the auth service is kept separate from the pure auth helpers. The main weakness is that the root and Project/ trees each carry copies of the same docs, schema and compose files, which will drift. Most of this is a packaging issue rather than a code-quality problem.

Requirement list

What the builder agreed to deliver before uploading — 12 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	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 6 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 6 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 6 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.