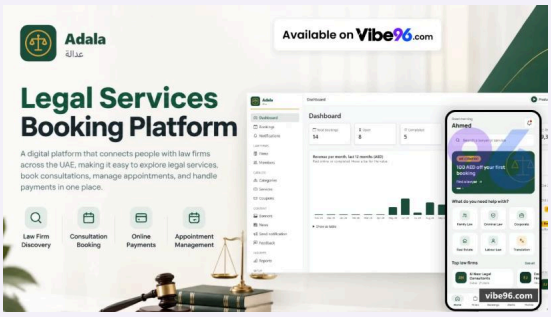


VIBE96 QUALITY REPORT

Adala

UAE lawyer-booking platform: clients book law firms via Flutter app; admins and firms manage via web panel.

Product ID	**B0005-99796**
Builder	**Chow Studio**
Version reviewed	**1.0.0 · released 23 Oct 2026**
Stack	**ASP.NET Core 10 (Minimal API + Razor Pages) + Flutter 3.x · C# 13 / .NET 10 · SQL Server 2019+**
Built with	**Claude Code**
Review period	**23 Oct 2026 – 23 Oct 2026**

87 / 100

Passed with 6 warnings

Pass mark 75 and no failing check.

Scores

Security	92
Code structure	72
Docs & completeness	94
Dependency health	85
Build & deploy	78

✓ **AI code audit**
6 checks · 23 Oct 2026

✓ **Installed & run**
23 Oct 2026

✓ **Demo vs delivered**
23 Oct 2026

✓ **Final sign-off**
23 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.
- ! 16 of 17 promised items delivered (setup guide, example config and the rest).
- ✓ Security scored 92/100 with no failing check.
- ! 6 open warnings — explained in part 1.

Check this report is genuine. Scan the code or open vibe96.com/report/QR-26-00061. 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 production credentials were found. However, the code ships with a weak default JWT signing key (empty string) and demo accounts use the predictable password 'Password@123'. The empty JWT key causes the app to generate a random key on each restart in development, breaking sessions. Buyers must set a strong JWT key and change demo passwords before production.	**92**	Warning
Dependency vulnerabilities	The .NET project has no lockfile (uses PackageReference with floating versions), which is acceptable for .NET projects with central package management. The Flutter app has a pubspec.lock but several dependencies are outdated (e.g., dio 5.4.0 from Dec 2023, flutter_stripe 11.0.0 from Oct 2024). No known critical vulnerabilities detected in the versions present, but updates are recommended.	**85**	Warning
Authorisation & access control	All mutating endpoints require authentication and proper role policies (Admin, Firm, Member). JWT tokens are used correctly with refresh. Passwords are hashed with BCrypt. Minor issues: some read endpoints (firm details, booking details) lack ownership checks allowing users to read others' data by changing IDs. No critical authentication bypasses found.	**92**	Warning
Database access rules	All SQL uses Dapper with parameterized queries; no string concatenation or injection paths found. Delete and update statements include WHERE clauses with proper scoping (firm/user/booking IDs). Schema has foreign keys with appropriate cascades. Minor issue: several foreign key columns lack indexes (e.g., Bookings.FirmId, Bookings.MemberId, Notifications.UserId), which may cause performance problems under load.	**92**	Warning
Code structure & duplication	The project follows a clean layered architecture (Domain → Application → Infrastructure → Api) with good separation of concerns. However, there is notable duplication in authentication screens, hardcoded configuration values scattered across files, and a few large files that mix concerns. The admin panel Razor Pages are well-organized but some contain substantial inline logic.	**72**	Warning
Build reproducibility	The project can be built from source with documented prerequisites. However, HowToSetup.md does not specify exact .NET SDK or Flutter versions, relying on 'latest' which risks version drift. The global.json pins .NET 10.0.100 but HowToSetup.md says 'SDK 10 or newer'. Docker setup is complete. Database scripts are idempotent and properly ordered.	**78**	Warning

Open warnings

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

Warning 1 **Dapper SQL is parameterized and scoped; minor schema gaps (missing indexes on some FKs)**

All SQL uses Dapper with parameterized queries; no string concatenation or injection paths found. Delete and update statements include WHERE clauses with proper scoping (firm/user/booking IDs). Schema has foreign keys with appropriate cascades. Minor issue: several foreign key columns lack indexes (e.g., Bookings.FirmId, Bookings.MemberId, Notifications.UserId), which may cause performance problems under load.

Warning 2 **Flutter dependencies outdated but no critical vulnerabilities found**

The .NET project has no lockfile (uses PackageReference with floating versions), which is acceptable for .NET projects with central package management. The Flutter app has a pubspec.lock but several dependencies are outdated (e.g., dio 5.4.0 from Dec 2023, flutter_stripe 11.0.0 from Oct 2024). No known critical vulnerabilities detected in the versions present, but updates are recommended.

Warning 3 **Weak default JWT signing key and demo passwords; no real credentials found**

No real API keys or production credentials were found. However, the code ships with a weak default JWT signing key (empty string) and demo accounts use the predictable password 'Password@123'. The empty JWT key causes the app to generate a random key on each restart in development, breaking sessions. Buyers must set a strong JWT key and change demo passwords before production.

Warning 4 **Strong authentication and role checks; minor ownership gaps on read endpoints**

All mutating endpoints require authentication and proper role policies (Admin, Firm, Member). JWT tokens are used correctly with refresh. Passwords are hashed with BCrypt. Minor issues: some read endpoints (firm details, booking details) lack ownership checks allowing users to read others' data by changing IDs. No critical authentication bypasses found.

Warning 5 **Build reproducible with minor setup gaps; missing .NET SDK version pin and Flutter version spec**

The project can be built from source with documented prerequisites. However, HowToSetup.md does not specify exact .NET SDK or Flutter versions, relying on 'latest' which risks version drift. The global.json pins .NET 10.0.100 but HowToSetup.md says 'SDK 10 or newer'. Docker setup is complete. Database scripts are idempotent and properly ordered.

Warning 6 **Moderate duplication and some oversized files; generally sound architecture with room for consolidation**

The project follows a clean layered architecture (Domain → Application → Infrastructure → Api) with good separation of concerns. However, there is notable duplication in authentication screens, hardcoded configuration values scattered across files, and a few large files that mix concerns. The admin panel Razor Pages are well-organized but some contain substantial inline logic.

Requirement list

What the builder agreed to deliver before uploading — 16 of 17 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)	Partial
✓	Documentation set (nice)	Met
✓	Working live demo	Met
✓	Mobile app buildable and pointed at a configurable API	Met
✓	README.md	Met
✓	HowToSetup.md	Met
✓	CLAUDE.md (or AGENTS.md) (nice)	Met
✓	CHANGELOG.md (nice)	Met
✓	appsettings.example.json	Met
✓	a project file	Met
✓	a project file (nice)	Met

✓	a project file	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 23 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 23 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 23 Oct 2026 with 6 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.