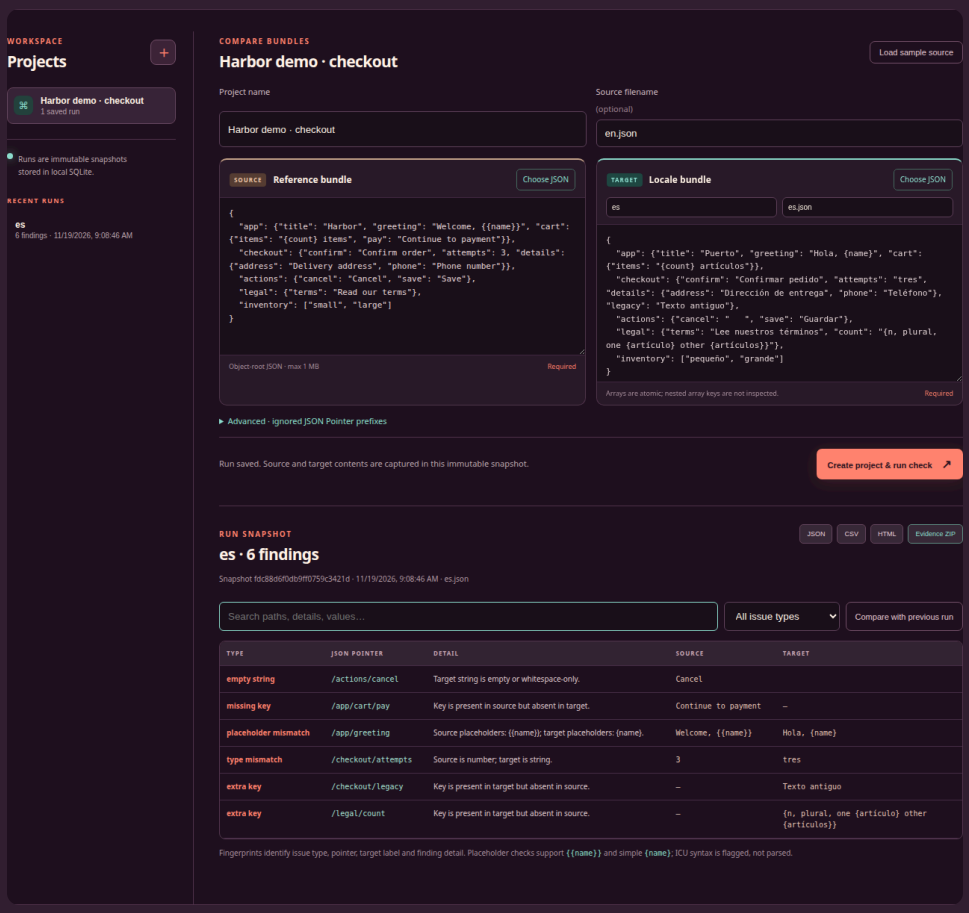
STRINGHARBOR / v1.0.0

LOCAL JSON LOCALIZATION QA STUDIO

Find the gaps before release.

A focused, single-user workspace for comparing one source bundle with one target locale at a time—on your own computer.



Actual v1.0.0 studio screenshot · sample project and run

Local-first | Python standard library | Immutable run snapshots

BUYER PREVIEW

HOW IT WORKS

One project.
One target per run.

Keep a source bundle as the reference, then check each locale separately. Findings become saved evidence you can revisit or share as output.

1

CREATE A PROJECT

Set the reference

Name a project and load an object-root JSON source bundle. Source and ignored-prefix settings stay fixed for that project.

2

CHOOSE A LOCALE

Check one target

Enter a target label and paste or select one target bundle. Run the comparison in the browser studio or with the command line.

3

REVIEW & EXPORT

Keep the snapshot

Browse findings, search or filter, compare fingerprints with the immediately older run, then export portable evidence.

SAVED LOCALLY

Runs live under Projects

Each run stores immutable snapshots of both JSON texts, settings, timestamp and findings in a local SQLite database. Select prior runs from the project sidebar.

USEFUL DIAGNOSTICS

Findings with paths

Use JSON Pointer locations to pinpoint missing or extra nested keys, type mismatches, blank target strings, supported placeholder differences and recognizable ICU-style constructs.

Real sample, not a benchmark: the included Spanish example report shows six findings, including a blank target string, a missing key and a placeholder-style mismatch.

Search spans type, pointer, detail and values. Issue-type filters and previous-run fingerprint comparison help keep review focused.

WHAT IT CHECKS—AND WHERE IT STOPS

QA evidence, with a narrow remit.

StringHarbor reports structural and selected string-level signals. It does not decide whether a translation is natural, complete in context, or correct.

Structure & validity

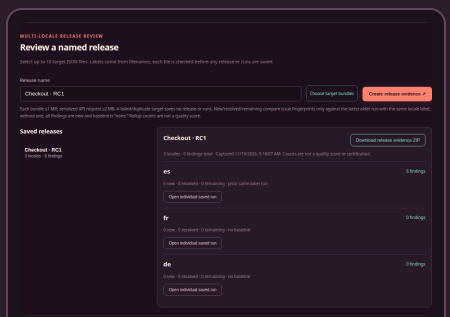
Strict JSON parsing with duplicate-key detection; both roots must be objects. Detects missing/extra nested object keys and JSON type mismatches.

String signals

Flags empty or whitespace-only target strings and checks parity for documented `{{name}}` and `{name}` placeholders.

Named multi-locale releases

Select 1–10 target JSON files for one named release. Each filename (without its extension) supplies the locale label; labels must be unique ignoring case. Every object-root bundle is validated before any save: one invalid or duplicate target aborts the whole release, with no partial release or runs. Each locale compares with the most recent older run using the exact same locale label in that project, whether standalone or released; new, resolved and remaining counts are frozen with the release. One ZIP contains release-rollup JSON/CSV/HTML, per-locale JSON/CSV/HTML evidence, and a SHA-256 manifest. Limits: 1 MB per bundle and 2 MB per serialized HTTP request.



Actual v1.0.0 release-review screen · sample run, 3 locales

Ignore & compare

Ignore JSON Pointer prefixes. Review prior saved snapshots and compare issue fingerprints with the immediately older run.

Local & read-only in scope

Designed for one user on one trusted computer. Loopback-only server; no outbound requests, accounts, cloud sync or team workspace. Release totals are observations, not a quality score or certification.

Important limits to plan around

ICU is a warning, not a parser: recognizable ICU-style patterns are flagged as unsupported; syntax, nesting and plural branches are not validated. **Arrays are atomic:** contents, indexes and nested keys are not inspected; array-to-array changes alone produce no finding. **Placeholders:** only documented double-brace `{{name}}` and simple `{name}` forms are covered. Root arrays are rejected.

Fictional sample release: source en → es/fr/de shows 6 / 0 / 0 findings (6 total); a later clean es run resolves 6 against its exact-locale baseline. No AI translation, locale grammar or plural-rule checks, CI service, team workspace or correctness guarantee.

GET STARTED & KEEP CONTROL

Runs on your own machine.

Python 3.10+ is the only requirement. StringHarbor uses the Python standard library—no third-party packages, build step, account or network service dependency.

```
$ cd StringHarbor-v1.0.0
$ python3 app.py serve

Open http://127.0.0.1:8891/
Server binds to loopback only.
Stop with Ctrl+C
```

LOCAL WORKFLOW

In the studio, name a project, load the source JSON, then add a target label and bundle. Run the check and find the saved snapshot in Projects.

Default database: `data/stringharbor.sqlite3` beside the app. `STRINGHARBOR_DB` can point to another local path. Back up the database file with the app stopped; no backup/restore UI is included.

JSON	Machine-readable evidence
CSV	Issue rows; formula-leading values are made safer for spreadsheets
HTML	Self-contained report with embedded styling and no remote assets
ZIP	Evidence package containing `evidence.json`, `issues.csv`, `report.html` and a SHA-256 manifest. The manifest hashes those three evidence files—not itself.

SOURCE-INCLUDED · SINGLE-PURCHASER LICENSE

The purchaser may use the software for personal work or within one business, on computers under the purchaser's control, and may modify it for those uses. The purchaser may share localization outputs made with the software. Redistribution, sublicensing, resale, publication or hosting of the software itself, modified or unmodified, and providing it as a service are not permitted without additional written permission. The software is provided “as is”; no support, maintenance or update promise is made. See the included LICENSE for the full terms.

The purchaser-controlled-computers wording above summarizes the license. Operationally, StringHarbor is designed for one user on one trusted computer; do not expose it to a LAN or public network. The local database contains original bundle text—protect it like source files. A diagnostic run is not certification of locale correctness. The included LICENSE controls.

Small, inspectable checks. Portable evidence. No translation claims.