

Get started

How the repositories fit together

🕒 3 min read · Updated 27 September 2026

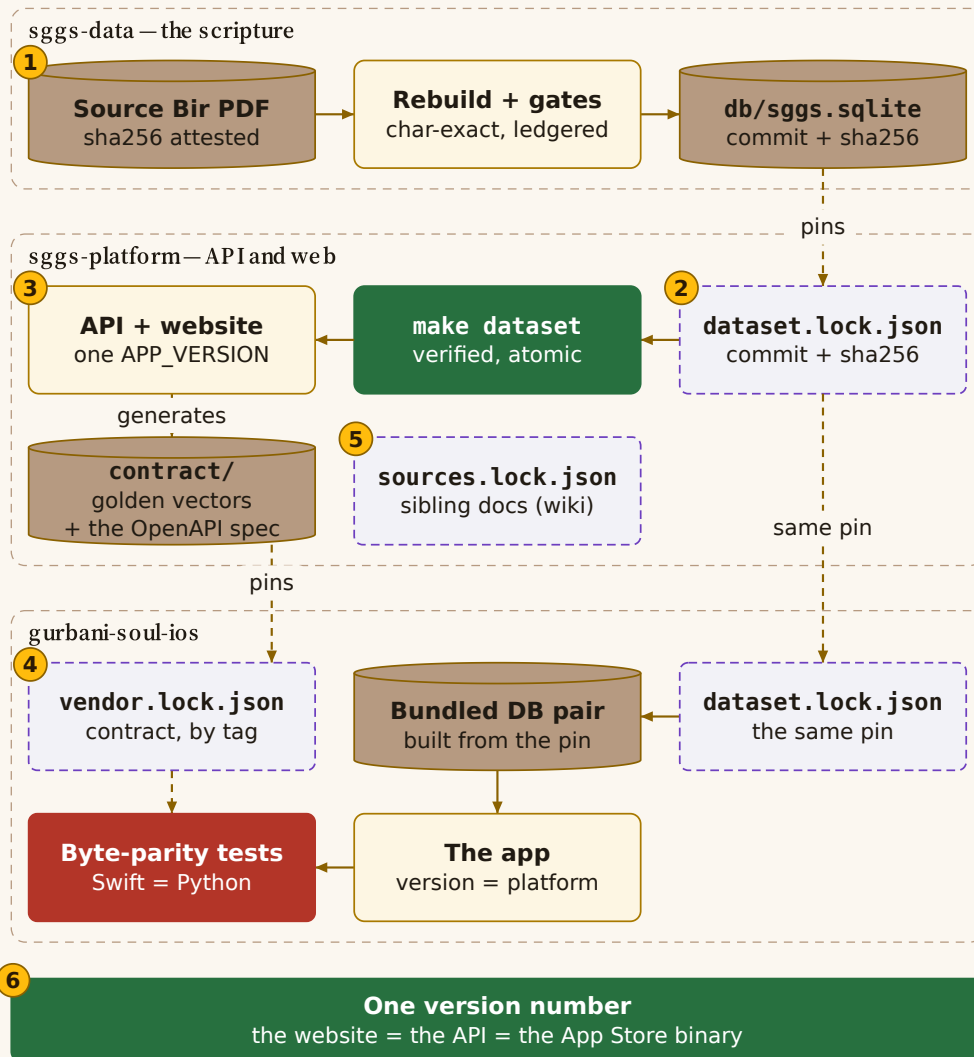
The three repositories — data, platform, app — what lives in each, the pinned artifacts that join them, and where a given kind of change is made.

<https://docs.gurbanisoul.com/onboarding/how-the-repos-fit/>

Since release 1.3.8 the project is three repositories joined by **pins**: small, reviewed files that name an exact commit and the checksum of the artifact taken from it (ADR-0007, ADR-0008). Nothing is copied by hand between them, so nothing can drift.

02 · Three repositories and their pins

Nothing is copied by hand between data, platform and app — every hand-off is a reviewed lock file with a commit and a checksum



Poster 02 — three repositories and their pins: sggs-data builds and publishes the database, sggs-platform pins and serves it, gurbani-soul-ios pins the contract and the dataset, one version number

- 1. The scripture is built and proven in sggs-data.** The source PDF's checksum is attested. The pipeline extracts the corpus, reconciles it character for character against the PDF, runs the golden checks, builds the SQLite database and passes the integrity gate. Any editorial repair is registered in the editorial ledger and approved in a scholar review. The database is published by commit, sha256 and size.

2. **The platform pins it.** `dataset.lock.json` names the `sggs-data` commit and the object's sha256 and size. make dataset downloads exactly that object, verifies it and installs it atomically; make `dataset-check` proves `sggs-data` publishes it at that commit. A new dataset arrives only as a reviewed bump of this file.
3. **The platform serves and publishes.** `webapp/` and `frontend/` serve the API and the website at one version. From the running code, `tools/gen_golden_vectors.py` records the golden contract — thousands of input and output vectors for the Roman fold, search, verification and the reader — and `tools/gen_openapi.py` the OpenAPI spec.
4. **The app vendors and re-proves.** `gurbani-soul-ios` pins the contract at a platform release tag (`vendor.lock.json`, a sha256 per file) and the same dataset pin. It builds its own bundled database pair from that pin and runs byte-parity tests so its Swift port equals the Python source of truth. Its `MARKETING_VERSION` must equal the platform version it vendored.
5. **This wiki pins the sibling docs.** `docs-site/sources.lock.json` records, per sibling repository, a commit and the sha256 of every published documentation file; the build installs them under `/data/` and `/ios/` with a banner naming the canonical source. A pin bump is a small reviewed pull request, opened by the weekly docs-pins job.
6. **One version number.** The web footer, `/api/meta`, `/api/health`, the iOS About screen and the App Store all report the same X.Y.Z, and every platform release is re-archived as iOS X.Y.Z (1) from the release tag. A release is complete only when web, API and the binary report the same number from the same commit.

What lives where

Repository	Owns	You go there to...
sggs-data	the source PDF's corpus, the SQLite/FTS5 database, the rebuild pipeline, the scripture gates (character-exact reconcile, golden checks, the editorial ledger), the answer protocol, the validation reports	change anything about the <i>text</i> or how it is extracted, indexed or proven — always with a scholar's review
sggs-platform (this one)	the stdlib API (<code>webapp/</code>), the website (<code>frontend/</code>), the golden contract (<code>contract/</code>), tooling (<code>tools/</code> , <code>scripts/</code>), CI, and this wiki (<code>docs/</code> , <code>docs-site/</code>)	change search, verification, the API, the website, delivery, documentation
gurbani-soul-ios	the iOS app, its widgets, its bundled database pair, TestFlight and App Store tooling, the Nitnem registry	change the app

A fourth, private repository (`sggs-source`) holds the backup of the source PDF and the private translation inputs; you will not need it.

The pins, in one table

Pin	Lives in	Names	Bump it when
<code>dataset.lock.json</code>	platform, app	an <code>sggs-data</code> commit + the database's sha256 and size	a new dataset was built and reviewed in <code>sggs-data</code>
<code>vendor.lock.json</code>	app	a platform release tag + the sha256 of each vendored contract file	the platform released <code>vX.Y.Z</code>
<code>docs-site/sources.lock.json</code>	platform (this wiki)	a sibling commit + the sha256 of each published doc	the sibling's docs changed and should appear here

A pin bump is an ordinary pull request: small, reviewed, gated. `make dataset` installs the pinned database; `make dataset-check` proves sggs-data publishes exactly that object at that commit.

One version number

The web footer, `/api/meta`, `/api/health`, the iOS About screen and the App Store all report the same `X.Y.Z` ([ADR-0004](#)). Every platform release is re-archived as iOS `X.Y.Z` (1) from the release tag, even a web-only change. The dataset has its own version (`db_version`), released on its own cadence through the pin.

Where does my change go?

- *Search returns the wrong line for a spelling* → platform (`webapp/sggs/search.py`, the fold in `webapp/romannorm.py`) — and if the fold changes, it must change byte-identically in sggs-data and the app (the golden vectors prove it).
- *A verse looks wrong on screen* → **do not edit anything**; open a [scripture fidelity issue](#)¹.
- *A page on the website* → platform (`frontend/`).
- *A screen in the app* → `gurbani-soul-ios`.
- *This wiki* → platform (`docs/`; the site itself in `docs-site/`).
- *A decision about how we work* → an ADR in `docs/adr/`, proposed in a pull request.

Links

1. <https://github.com/Algoythmos-AI/sggs-platform/issues/new?template=scripture-fidelity.yml>

Sri Guru Granth Sahib Ji — Knowledge Base wiki. Generated from commit `c3d9360`. Updated 27 September 2026. This copy may be out of date; the page is at <https://docs.gurbanisoul.com/onboarding/how-the-repos-fit/>