

Get started

Your first pull request

🕒 3 min read · Updated 25 September 2026

How a change reaches the trunk here: a branch from integration, the local gates, a conventional title, the PR template, the checks, staging review and the owner's merge.

<https://docs.gurbanisoul.com/onboarding/your-first-pr/>

The project's delivery loop is deliberately strict, and every step of it is written down. Follow it once by the book and it becomes muscle memory. Full detail: [Branching](#), [Delivery](#), [CONTRIBUTING](#)¹.

Pick something small

A good first change is one you can describe in one sentence and review in one screen: a documentation fix, a test for an existing behaviour, a small tooling improvement, a diagram. Look for issues labelled *good first issue*, or ask in [Who to ask](#). Do **not** start with search ranking, the fold, the verifier or the dataset pin — those are owner-reviewed paths (`.github/CODEOWNERS`) with scripture consequences.

1. Branch from integration

integration is the trunk; main is production and only ever receives release merges.

```
git fetch origin
git switch -c fix/short-description origin/integration    # or feat/..., docs/...
```

If you keep several changes in flight, use a worktree per branch (`git worktree add ../wt-fix -b fix/short-description origin/integration`) so one never disturbs another.

2. Make the change, then run the gates

```
make ci                # what the required checks run (no PDF needed)
make docs-check        # if you touched docs/
```

Keep the diff under about 400 changed source lines. If you changed behaviour or an invariant, update the handbook page that describes it in the same pull request.

3. Commit as yourself

Commits carry the author's name only: no tool attribution trailers, no generated-by footers, no co-author lines for tools. Write the message for the person reading `git log` in a year.

4. Open the pull request into `integration`

```
git push -u origin fix/short-description
gh pr create --base integration --title "fix(web): short description"
```

- **Title:** conventional-commit style — `fix(web): ...`, `feat(api): ...`, `docs: ...`. The pre-hygiene check enforces it.
- **Body:** the template asks *what and why* (link the issue: Closes #NN), the **scripture-safety checklist** (fill it in for any change to search, verify, display or the dataset pin; delete it otherwise), and the *Definition of Done* (tests, versions, CHANGELOG under *Unreleased*, docs updated, verified on staging).

5. Watch the checks and read what they mean

```
make pr-checks PR=<number>
```

Every check is documented in [CI gates](#). The ones that must be green to merge into `integration` are `python`, `frontend`, `integrity` and `secrets`; the others (`docs`, `playwright`, `static-analysis`, ...) still run and the owner will not merge over a red one. A failing check tells you the file and the line; fix it on the same branch and push again.

6. You cannot merge — and that is the design

Every change is reviewed by the code owner and merged by the owner with the squash strategy. When your checks are green, hand over the exact command:

```
gh pr merge <number> --repo Algorythmos-AI/sggs-platform --squash --admin
```

The merge deploys **staging** (`sggs-staging.vercel.app`, sign in with Vercel) within a few minutes by CI; look at your change there. Production is a separate, later step: a release pull request from `integration` to `main`, merged as a merge commit, which runs the gated production deploy and tags `vX.Y.Z`.

What reviewers look for

- The change does exactly what the title says, and nothing else.
- Scripture is untouched: no edit, normalisation or reorder in storage, search, copy or share; display-only transforms stay display-only; no verse typed into code, tests or docs.

- Tests exist for the behaviour and are green; a bug fix comes with the test that would have caught it.
- Versions stay unified; secrets never appear in the diff or in the conversation.
- The handbook page for the affected area still tells the truth.

Links

1. <https://github.com/Algorithmos-AI/siggs-platform/blob/integration/CONTRIBUTING.md>

Sri Guru Granth Sahib Ji — KnowledgeBase wiki. Generated from commit c3d9360. Updated 25 September 2026. This copy may be out of date; the page is at <https://docs.gurbanisoul.com/onboarding/your-first-pr/>