ONBOARDING_GUIDE

Waveframe Labs — Contributor Onboarding Guide

Welcome to **Waveframe Labs**, home of the *Aurora Workflow Orchestration (AWO)* and *Waveframe* research series.
This guide explains how to get oriented, set up your environment, and meet the minimum reproducibility standards for participation.

---

1. Core Principles

Every repository under Waveframe Labs adheres to three enforcement pillars:

Principle	Description
Reproducibility	All code, reasoning, and results must be independently regenerable from committed artifacts.
Falsifiability	Every claim or output must be testable against a defined manifest or challenge case.
Neurotransparency	Each inference that affects a claim must have a traceable origin (role, artifact, or hash).

Contributors are expected to internalize these before engaging in any development or documentation work.

---

2. Required Tools

Tool	Purpose
GitHub	Version control and workflow orchestration (repositories, issues, Actions).
Pandoc + TeX	Required to build PDFs automatically from Markdown (used in all AWO projects).
Python 3.10+	Used for local testing, manifest validation, and hashing utilities.

Tool	Purpose
SHA256 utility	Any CLI or script capable of generating deterministic hashes for artifacts.
(Optional) **Streamlit / Jupyter**	For interactive prototypes or visualization of experiment runs.

No specialized proprietary software is required — only open, verifiable tooling.

---

3. Your First Tasks

When joining or reviewing a repository:

1. **Read the Method Spec** (`docs/AWO_Method_Spec_v1.2.1.md`)
   Understand the compliance levels and required artifacts.
2. **Run a local verification**
   - Check for `SHA256SUMS.txt` in the root.
   - Confirm all ADRs are sequential and cited.
   - Review `/runs/` for at least one attested, passing run.
3. **Create your first controlled contribution**
   - Edit or propose a small improvement (README, ADR draft, minor script).
   - Submit as a Pull Request referencing a falsifiability manifest.
   - Tag your role (e.g., `Reviewer`, `Orchestrator`, `Critic`) in the PR body.
4. **Run the applicable GitHub Actions**
   - PDF build → ensures documentation reproducibility.
   - Root SHA256SUMS workflow → updates integrity registry.

---

4. Review Roles (Quick Summary)

Role	Responsibility	Reference
Orchestrator	Coordinates reasoning flow and validates procedure.	§3.1 Method Spec
Auditor	Independently verifies artifacts and checksums.	§3.3 Method Spec
Critic / Red Team	Attempts falsification or contradiction.	§3.5a Method Spec

Role	Responsibility	Reference
Maintainer	Ensures structural compliance and governance logs are current.	ADR-0017

---

5. Checkpoints for Acceptance

A contribution will not be merged unless:

☒ A valid falsifiability manifest is attached.

☒ The corresponding run artifact is frozen and hashed.

☒ Each non-trivial decision has an ADR.

☒ All files are included in `SHA256SUMS.txt`.

☒ Output reproducibility is confirmed via rerun or validator.

☒ No unexplained nondeterminism is present.

---

6. Community Norms

- **Be respectful**, but prioritize precision over politeness.
- **Assume good intent**, but require good artifacts.
- **Document before you debate** — speculation without evidence slows everyone down.
- **Prefer commits over comments** — progress is measurable, opinions are not.

Welcome aboard.
If you can prove it, you belong here.
