**CICD-Automation Team Working Agreement**

**Project:** Internal Developer Support CI/CD-Automation   
**Team Lead:** Srinivasa Varma Penmetsa  
**Developers:** Mehul Antony, Henry Wang  
**Client:** Daniel Shown

**1. Purpose**

This document establishes how the CICD-Automation Team will collaborate to deliver high-quality, reusable CI/CD automation resources. It outlines technical practices, working style, team processes, and client engagement. The agreement is live and will be updated as the team evolves.

**2. Technical Standards**

**2.1 Code Review**

All code changes must be submitted via pull requests and at least one team member must review and approve before merging to the main.

* Pull requests must include a clear description of the change.
* PRs should reference related issues or tasks.
* Reviewers must provide actionable feedback.

**2.2 Branching Strategy**

No direct commits to main. Pull requests must pass code review and tests before merging.

**2.3 Repository Standards**

Our team maintains a central GitHub repository containing CI/CD automation resources.  
**Repo Structure** (main/):

* templates/ → Standardized GitHub Actions workflows (reusable across projects).
* audits/ → Audit scripts, compliance check, timestamped reports.
* helpers/ → Utility scripts, Shared functions, Common utilities.
* README.md → Repo overview, setup, and contribution guidelines.

**2.4 Definition of done**

* Feature or fix is implemented.
* Reviewed and approved via PR.
* Documentation and ADRs updated.
* Verified on at least one project.
* Client reviewed

**3. Team Processes**

**3.1 Meeting Schedule**

* Twice per week via Zoom.
* Emergency/in-person meetings (SSE 210) if needed.
* Client joins sprint planning and review meetings.

**3.2 Communication**

* Primary channel: Slack.
* Response time: within 24 hours on working days.
* Urgent issues can be escalated via Slack tagging.

**3.3 Decision Making**

* All major technical decisions follow ADR format.
* Team consensus is preferred; Tech Lead makes final call if needed.

**3.4 Conflict Resolution**

* Discuss 1:1 first.
* Escalate to Tech Lead if unresolved.
* Final mediation with all parties, including client if required.

**3.5 Working Hours**

* Team members indicate availability at start of sprint.
* Expected overlap hours for collaboration.

**4. Collaboration with Project Teams**

We follow a 5-step collaboration model with project teams:

* Need Identified – Project team flags automation/compliance requirement.
* Collaborative Work – Team works with the project team to build a solution.
* Issue Resolved – Specific implementation created and validated.
* Template/Script Created – Final solution generalized and pushed into repo.
* Documented – ADR written and committed alongside code.

**5. Client Engagement**

The client plays an essential role in:

* Sprint Planning: Help prioritize work and provide input.
* Sprint Review: Review progress and provide timely feedback.
* Community Building: Participate in discussions around broader impact and reuse.

**6. Quality Assurance**

**6.1 Bug Tracking**

* All issues tracked in GitHub Issues with tags.
* Critical bugs flagged to Tech Lead and discussed in next meeting.

**6.2 Performance & Accessibility**

* CI workflows should run in under 15 minutes where possible.
* Tools/scripts should support basic accessibility guidelines.

**6.3 Security**

* Scripts/templates must follow secure coding standards.
* Secrets must be stored in GitHub Secrets, never hardcoded.
* Audits must include checks for dependencies and secrets.

**6.4 Deployment & Releases**

* Releases are tagged in GitHub.
* Release notes include changelog, linked PRs, ADRs.

**7. Document Change Log**

1.0 | 2025-09-05 | Initial draft created | Srinivasa Varma Penmetsa

**8. Signatures**

Srinivasa Varma Penmetsa - Varma

Mehul Antony - Mehul

Henry Wang - Henry

Daniel Shown -