Institute of Primate Research

STANDARD OPERATING PROCEDURE (SOP) DOCUMENT

Reproducible coding practices (Git, R Markdown, Jupyter.)

SOP No.	Issue Number	Issue Date	Revision Status	Revision Date
SOP/KIPRE/RPD/DSAS/3.1.76	Version 01	October 2025	-	-

Approvals

	Name	**Signature**	**Date**
Developed by:	_Patrick Waweru Mwaura_	_____	**_6th October; 2025_**
	_____	_____	_____
	_____	_____	_____
Reviewed by:	_____	_____	_____
Approved by:	_____	_____	_____

Table of Contents

1. PURPOSE 4

2. SCOPE 4

3. PERSONS RESPONSIBLE: 4

4. FREQUENCY 4

5. MATERIALS 4

6. PROCEDURE 5

7. REFERENCES 5

1. PURPOSE

To establish standardized, transparent, and reproducible coding practices across DS&AS projects, ensuring all analyses, pipelines, and reports are version-controlled, auditable, and maintainable. This SOP provides guidance for using Git-based version control, literate programming tools (R Markdown, Jupyter, Quarto), and collaborative workflows to support reproducibility, traceability, and compliance with institutional policies (SOPs 1, 6, 8, 16) and best practices in computational research.

2. SCOPE

Applies to all coding activities within DS&AS-supported projects, including statistical analyses, machine learning, bioinformatics pipelines, data preprocessing, and report generation. This SOP covers the use of version control, literate programming tools, and collaborative coding practices to ensure reproducibility, transparency, and traceability of all project workflows.

3. PERSONS RESPONSIBLE:

- **DS&AS Analysts and Data Scientists:** Follow reproducible coding practices for all scripts, analyses, and documentation.
- **Data Engineer:** Maintains and administers version-control systems (Git, GitHub/GitLab), ensuring access control, backups, and repository organization.
- **Head of DS&AS:** Monitors adherence to reproducibility standards, approves workflows, and ensures institutional compliance with coding SOPs.
- **Project Leads / Principal Investigators:** Ensure that team members implement SOP 17 guidelines throughout the project lifecycle.

4. FREQUENCY

- **Throughout Project Lifecycle:** Reproducible coding practices must be applied continuously during all stages of data analysis, bioinformatics, machine learning, and reporting.
- **Quarterly Audits:** DS&AS conducts repository reviews to ensure compliance with coding standards and version-control practices.

- **Pre-Release Review:** All major project outputs must be verified for reproducibility before dissemination or publication.

5. MATERIALS

- **Version Control Systems:** Git, GitHub Enterprise, GitLab, or institutional Git repositories.
- **Documentation & Analysis Tools:** R Markdown, Jupyter Notebooks, Quarto, or other literate programming frameworks.
- **Institutional Coding Guidelines:** Standard templates, naming conventions, and workflow protocols.
- **Collaboration Tools:** Pull requests, code review frameworks, and issue tracking systems.

6. PROCEDURE

1. **Repository Setup**
    - Create a dedicated project repository using Git (institutional or GitHub Enterprise).
    - Organize folders for raw data, scripts, results, and documentation according to institutional standards.
2. **Version Control**
    - Commit all scripts, notebooks, and configuration files with descriptive messages.
    - Use branching for feature development, bug fixes, or experiments, and merge only after code review.
    - Tag versions corresponding to major milestones or project releases.
3. **Reproducible Documentation**
    - Use R Markdown, Jupyter, or Quarto to combine code, results, and narrative explanations in a single document.
    - Ensure all analyses can be re-run from raw data to final outputs using provided scripts.
4. **Collaboration & Peer Review**

- Employ pull requests for all major code changes.
- Conduct peer reviews of scripts and notebooks before merging to main branches.
- Maintain issue logs to track bugs, updates, and reproducibility concerns.

5. **Archiving & Metadata**
    - Store final project outputs, scripts, and documentation in the DS&AS central repository.
    - Include metadata describing dataset versions, code dependencies, software environments, and computational parameters.
6. **Compliance Audits**
    - Conduct quarterly reviews of repositories to ensure adherence to reproducibility standards.
    - Document audit findings, and implement corrective actions if deviations from standards are identified.

7. REFERENCES

1. **Linked SOPs:**
    - SOP 12: Genome and Proteome Data Management – secure handling of sequence/proteomic data.
    - SOP 13: Bioinformatics Pipelines – reproducible workflows for raw sequence analysis.
    - SOP 14: Development and Validation of Computational Tools – standards for in-house software development.
    - SOP 16: Handling Large Datasets and Trend Detection – reproducible processing of big datasets.
2. **Standards & Guidelines:**
    - ISO/IEC 25010:2011 – Systems and software engineering: Software product quality requirements and evaluation (SQuaRE).
    - FAIR Principles – Wilkinson et al., 2016.
    - Kenya Data Protection Act, 2019.
    - KIPRE Institutional Data Governance and Software Development Guidelines.

- Sandve, G. K., et al., 2013. *Ten simple rules for reproducible computational research*, PLoS Comput Biol.

3. **Tools & Documentation:**
   - Git/GitHub/GitLab documentation – version control best practices.
   - R Markdown, Jupyter, Quarto – guidelines for literate programming and reproducible reporting.
   - Docker/Conda – environment reproducibility standards.