Engineering_Code_Standards_Policy

Purpose:
Ensure clean, maintainable, and consistent code across all engineering teams.

Key Rules:

- Follow approved coding standards and style guides

- Write readable and well-documented code

- Mandatory code reviews before merging

- Avoid hard-coded secrets or credentials

Ownership: Engineering Team Leads

📄 Version_Control_and_Branching_Policy

Purpose:
Define rules for using version control systems.

Key Rules:

- Use company-approved repositories only

- Follow defined branching strategy (e.g., main, dev, feature)

- Commit messages must be clear and meaningful

- No direct commits to production branches

Ownership: Engineering & DevOps Team

📄 Software_Development_Lifecycle_Policy

Purpose:
Standardize the software development lifecycle (SDLC).

Key Rules:

- Follow defined stages: requirement, design, development, testing, deployment

- Maintain documentation at each stage

- Testing is mandatory before release

- Changes must follow approval process

Ownership: Engineering Management

📄 Code_Review_and_Quality_Assurance_Policy

Purpose:
Maintain high code quality and reduce defects.

Key Rules:

- All code must be peer-reviewed

- Automated tests should pass before merge

- Follow static analysis and linting rules

- Fix critical issues before release

Ownership: Engineering & QA Teams

📄 Security_and_Secure_Coding_Policy

Purpose:
Protect systems, applications, and data from security risks.

Key Rules:

- Follow secure coding best practices

- Do not expose secrets, tokens, or API keys

- Regularly update dependencies

- Report security vulnerabilities immediately

Compliance: Company Security Standards

📄 Deployment_and_Release_Policy

Purpose:
Ensure safe and controlled software releases.

Key Rules:

- Deploy only tested and approved builds

- Maintain rollback plans

- Monitor systems post-deployment

- Emergency releases require approval

Ownership: DevOps & Engineering Teams

📄 Engineering_Documentation_Policy

Purpose:
Ensure proper technical documentation for long-term maintainability.

Key Rules:

- Maintain README and API documentation

- Update docs with every major change

- Store documentation in approved repositories

Ownership: Engineering Teams

📄 Incident_and_Outage_Management_Policy

Purpose:
Handle production incidents effectively.

Key Rules:

- Report incidents immediately

- Follow defined escalation process

- Perform root cause analysis (RCA)

- Document lessons learned

Ownership: Engineering & SRE Teams