Tool Metadata Report (by MetadataFetcher)

1. General Information

Name	Git (Version Control)
Use Case	Developer Tools
Homepage	https://git-scm.com/
Description	Git is the most widely used distributed version control system in software development, created by Linus Torvalds in 2005. According to 2025 statistics, Git is used by 93.87% of developers worldwide, making it the de facto standard for version control. Git enables developers to track changes in code, collaborate effectively, and maintain project history with branching and merging capabilities.

2. Supported Languages/Technologies:

Git is language-agnostic and supports all programming languages and file types. It integrates seamlessly with:
Development Environments: VS Code, IntelliJ IDEA, Eclipse, Xcode
Hosting Platforms: GitHub, GitLab, Bitbucket, Azure DevOps
CI/CD Tools: Jenkins, GitHub Actions, GitLab CI, Travis CI
Project Management: Jira, Trello, Asana integration
IDEs and Editors: Universal support across all development tools

3. Key Features:

Distributed architecture with local repository capability
Branching and merging with conflict resolution
Staging area for selective commit preparation
Complete project history and rollback capabilities
Collaborative workflows with remote repositories
Lightweight and fast operations
Data integrity through SHA-1 checksums
Flexible workflow support (Git Flow, GitHub Flow)

4. System Requirements:

Operating Systems: Windows, macOS, Linux, Unix variants
Storage: Minimal space requirements (typically <100MB)
RAM: Low memory footprint, works on resource-constrained systems
Network: Optional for local operations, required for remote synchronization
Dependencies: Minimal system dependencies, self-contained installation

5. Installation & Setup:

Git can be installed through platform-specific installers, package managers, or GitHub Desktop. Configuration includes setting user name and email for commit attribution. Popular installation methods include direct download, Homebrew (macOS), and package managers (Linux).

6. Integration Capabilities:

IDEs: Native integration with most development environments
Hosting Services: GitHub, GitLab, Bitbucket, Azure Repos
CI/CD Platforms: Jenkins, GitHub Actions, GitLab CI/CD

Project Management: Issue tracking and project board integration
Code Review: Pull request and merge request workflows
Documentation: Wiki and documentation hosting integration

7. Version Control Support:

Branching: Lightweight branch creation and switching
Merging: Three-way merge with conflict resolution
History: Complete project timeline with commit messages
Tagging: Release versioning and milestone marking
Remote Repositories: Distributed collaboration capabilities
Hooks: Automated scripts for workflow integration

8. Extensions/Plugins:

Git ecosystem includes numerous extensions and tools:
Git LFS: Large File Storage for binary files
Git Flow: Branching model implementation
Git Hooks: Custom automation scripts
GUI Clients: GitKraken, SourceTree, Tower
Command Line Tools: Hub, gh (GitHub CLI), glab (GitLab CLI)

9. Documentation & Tutorials:

Extensive documentation includes official Git manual, tutorials, and community resources. Educational platforms like GitHub Learning Lab, Atlassian Git tutorials, and interactive courses provide comprehensive learning paths for beginners to advanced users.

10. Community & Support:

Git has a massive global community with active development and extensive third-party tool ecosystem. Support is available through official documentation, Stack Overflow, GitHub discussions, and community forums. Regular updates and improvements are maintained by core developers and contributors.

11. Licensing:

GNU General Public License v2 (GPL-2.0) - Free and Open Source

12. Latest Version / Release Date:

Git 2.50+ (2025) with regular feature releases and security updates

13. References:

Official Website: https://git-scm.com/
Documentation: https://git-scm.com/doc
GitHub Repository: https://github.com/git/git

14. Other Links:

https://git-scm.com/downloads - Official Downloads
https://git-scm.com/docs/gittutorial - Official Tutorial
https://learngitbranching.js.org/ - Interactive Git Tutorial
https://github.com/git-guides - GitHub Git Guides
https://www.atlassian.com/git/tutorials - Atlassian Git Tutorials

https://stackoverflow.com/questions/tagged/git - Stack Overflow Git Q&A
https://git-scm.com/docs - Official Documentation
https://gitimmersion.com/ - Git Immersion Tutorial
https://github.com/k88hudson/git-flight-rules - Git Flight Rules
https://ohshitgit.com/ - Git Problem Solutions
https://github.com/arslanbilal/git-cheat-sheet - Git Cheat Sheet
https://www.youtube.com/watch?v=8JJ101D3knE - Git Tutorial Video
https://git-scm.com/book - Pro Git Book (Free)
https://github.com/git-tips/tips - Git Tips Repository
https://gitexplorer.com/ - Git Command Explorer
https://www.git-tower.com/learn - Git Tower Learning
https://github.com/pluralsight/git-internals-pdf - Git Internals PDF
https://training.github.com/ - GitHub Training
https://backlog.com/git-tutorial/ - Backlog Git Tutorial
https://www.codecademy.com/learn/learn-git - Codecademy Git Course