



TSU en Tecnologías de la Información Área Desarrollo de Software Multiplataforma

Nombre:

García Arreola Howard Isaí

Subject:

Software Development Process Management

Actividad:

Techniques and tools for tracking test

Grupo: 10-B

Profesor:

Ray Brunett Parra Galaviz

Fecha de Realización:

January 13th, 2025

Techniques and tools for tracking evidence

Managing software development projects and tracking issues can be overwhelming without the right tools. These are some of the best tracking tools from the last year:

- Teamwork.com — Best for project management with a focus on collaboration and team harmony
- ProofHub — Best for planning, organizing, and delivering projects with Gantt charts
- Azure DevOps — Best for DevOps teams seeking an end-to-end plan, develop, deliver, and operate solutions
- Linear — Best for teams seeking streamlined issue tracking in a beautiful interface
- YouTrack by JetBrains — Best for agile project management with customizable workflows
- New Relic CodeStream — Best for integrating code discussion directly in your IDE
- Zoho — Best for comprehensive project tracking with time logging
- Jira — Best for software development projects with robust issue and bug tracking
- ClickUp — Best for all-in-one project management with feature-rich functionality
- monday.com — Best for flexible project visualization and customizable workflows

Tracking evidence in software development is essential for ensuring transparency, accountability, and quality throughout the development process. Various techniques and tools are employed to manage and track evidence effectively. The following are some techniques and the tools that are used within those techniques:

1. **Version Control Systems (VCS):** Tools like Git, Subversion, and Mercurial are used to track changes in the source code. They provide a history of modifications, allowing developers to revert to previous versions if needed. VCS

also facilitates collaboration among team members by managing code merges and conflicts.

2. **Issue Tracking Systems:** Tools like Jira, Trello, and Bugzilla help track bugs, feature requests, and tasks. These systems provide a centralized platform for reporting, assigning, and monitoring issues, ensuring that nothing falls through the cracks.
3. **Continuous Integration/Continuous Deployment (CI/CD) Tools:** Tools like Jenkins, Travis CI, and CircleCI automate the process of integrating code changes and deploying applications. They track build and deployment statuses, providing evidence of successful or failed builds and deployments.
4. **Automated Testing Tools:** Tools like Selenium, JUnit, and TestNG are used to automate the testing process. They generate reports on test results, helping developers identify and fix issues early in the development cycle.
5. **Code Review Tools:** Tools like Crucible, GitHub Pull Requests, and GitLab Merge Requests facilitate peer reviews of code changes. They provide a platform for discussing and approving code modifications, ensuring that code quality standards are met.
6. **Documentation Tools:** Tools like Confluence, Notion, and Google Docs are used to create and maintain project documentation. They ensure that all project-related information is documented and easily accessible to team members.

References:

Club, C. (2025, January 13). *Leading the Pack: 25 Best Software Development*

Tracking Tools in 2025 - The CTO Club. The CTO Club.

<https://thectoclub.com/tools/best-software-development-tracking-tools/>

Chappell, E. (2024, December 30). *Best 16 Agile Tools for Project Management in*

2025. ClickUp. <https://clickup.com/blog/agile-tools/>