Training TR-102:Day 14 Report

Date: 29th June 2024

Topic: GIT, Agile Methodologies, Trello, Jira, and Practical Implementation

---

Overview:

On the fourteenth day of the TR-102 training, participants deepened their knowledge of **GIT** and its key commands, explored **Agile methodologies**, and familiarized themselves with popular project management tools like **Trello** and **Jira**. The session emphasized practical exercises using GitHub and GitLab, teaching participants how to resolve **merge conflicts** and manage project workflows effectively.

GIT:

Introduction to GIT:

- **Version Control System:**
  GIT is a distributed version control system that tracks changes in source code, making collaboration between developers smoother and maintaining a detailed history of all modifications.
- **Purpose:**
  GIT helps developers collaborate efficiently, ensuring that changes are tracked and managed over time.

Key GIT Commands:

- **Push:** Uploads local repository changes to a remote repository.
  Example: `git push origin main`
- **Pull:** Fetches and integrates changes from a remote repository into the local one.
  Example: `git pull origin main`
- **Merge:** Combines changes from different branches into one.
  Example: `git merge feature-branch`
- **Branch:** Creates a new branch for developing features or fixing bugs.
  Example: `git branch new-feature`

Handling Merge Conflicts:

- **Definition:**
  Merge conflicts occur when changes from different branches clash with each other.

- **Resolution:**
  Conflicts are resolved by manually editing conflicting files and ensuring that all discrepancies are addressed.

GitHub and GitLab: Practical Implementation

Key Operations:

- **Fork:** Creates a personal copy of another user's repository.
  Example: `git fork repository-url`
- **Clone:** Downloads a repository from a remote location to the local machine.
  Example: `git clone repository-url`
- **Push:** Sends local commits to a remote repository.
  Example: `git push origin branch-name`
- **Merge:** Integrates changes from different branches.
  Example: `git merge branch-name`
- **Pull Request:** Proposes changes for review and merging into the main branch.
  Example: `git pull-request`

Practical Exercises:

Participants engaged in hands-on exercises that involved forking, cloning, pushing, merging, and creating pull requests using **GitHub** and **GitLab**. They also practiced resolving **merge conflicts**, ensuring seamless integration of updates across different branches.

Task Management Tools: Trello and Jira

Trello:

- **Description:**
  Trello is a visual project management tool that helps organize tasks using **boards**, **lists**, and **cards**.
- **Usage:**
  Participants learned how to create Trello boards, add lists and cards, assign tasks to team members, and track task progress through a visual workflow.

Jira:

- **Description:**
  Jira is a robust tool for issue tracking and **Agile project management**. It helps teams plan, track, and manage software development projects.
- **Usage:**
  Participants explored how to create projects, write **user stories**, set up **sprints**, and manage task tracking and resolution using Jira.

Conclusion:

Day 14 of the training offered a comprehensive introduction to **GIT**, covering essential commands, managing **merge conflicts**, and hands-on practice with GitHub and GitLab. The session also introduced participants to **Agile methodologies** and task management tools like **Trello** and **Jira**, providing them with the necessary skills to manage and collaborate on projects effectively. Equipped with these tools and methodologies, participants are now prepared to implement them in real-world projects, fostering better collaboration, tracking, and version control.