

# GIT

<p><b>Objectives</b></p> <ul style="list-style-type: none"> <li>✓ Install and configure GIT</li> <li>✓ Master the main commands</li> <li>✓ Using the remote repositories</li> <li>✓ Using GitLab, GitHub, eGit</li> <li>✓ Managing projects with GIT</li> </ul>	<p><b>Target Audience</b></p> <ul style="list-style-type: none"> <li>✓ Developers</li> <li>✓ Project managers</li> <li>✓ Software Architects</li> </ul>	<p><b>Pre-requisites</b></p> <ul style="list-style-type: none"> <li>✓ None</li> </ul>	 <p>Duration 2 days</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------

## Source Code Management

- Introduction to SCM
- The different systems : local, centralized, distributed
- Examples and use cases

### GIT Introduction

- GIT History
- GIT Advantages
- Basic principles : commit, staging, ...

### GIT Installation

- System Install
- Configuration parameters (user, ignore, ...)
- Push and pull configurations

### Basic commands

- Creating a first repository
- Adding or removing files
- Moving and deleting files
- Commit, checkout, reset, log, status
- Displaying commit history and moving inside.

### Branch management

- Introduction to branches and HEAD usage
- Branch creation
- Branch navigation
- Branch merge and rebase

### Remote repositories

- Remote configuration
- Access protocols (ssh, https, git)
- Push, pull and fetch
- Remote branches management

## Advanced commands

- Tag management
- Cherry pick
- Commits diff
- Stash commit
- Reorganizing commits with interactive rebase
- Importing a svn repository with its history
- Sub modules
- Hook management
- GIT Object and optimizations
- Advanced Uses cases

### GIT clients

- Source Tree
- eGit in Eclipse : views (staging, history), commands
- GitKraken
- TortoiseGIT
- Using these tools to browse the exercises

### GIT environments

- GitHub / GitLab : introduction, pull and merge requests
- Pushing the exercises on these environments
- Continuous Integration with GitHub and GitLab

### GIT best practices on a Project

- GitFlow : a branch model
- Best practices to use GIT on your project

### GIT internals

- Using internal objects (blob, trees, files)
- Main plumbery commands

### Exercises

- A lot of different exercises from the simplest to the hardest. All of them are done using command line.