

Git Training Agenda

•Distributed Version Control Systems and Git

- Difference to centralized version control system like cvs, svn
- Advantages
- Usage of Git
- Popular projects using Git
- Git history
- The usage of the staging area
- Git setup and configuration

•Git basics and Git repositories

- Creating Git repositories
- Adding files to a Git repository
- Internal structure of a Git repository
- Remote (bare) repositories

•Working with branches and tags

- Using local and remote branches
- Tagging
- Synchronizing branches

•Using the Git history

- Accessing different versions of the files
- Accessing older versions of the files
- Temporary saving changes with git stash

•Merging changes of different developments

- Merging changes with the merge command
- Merging changes with the rebase command
- Selecting individual changes
- Merge and rebase best practices

•Working and synchronizing with remote repositories

- Configuration of remote repositories
- Receiving changes from remote Git repositories
- Pushing changes to remote repositories

- Creating patches and pull requests
- Change analysis in a Git repository**
 - Differences between two commits
 - See the changed files between two commits
 - Using git bisect for error search
- Git hosting provider (Optional)**
 - Overview of Github and Bitbucket
 - Using a custom Git Server
- Team organization and best practices**
 - Avoiding and solving merge conflicts
 - Best practices for working with Git
- Using the Eclipse IDE for Git operations**
 - Cloning online repositories
 - Git operations with Eclipse
 - Displaying the owner of a change in Eclipse
 - Eclipse advanced merge conflict resolution support
- Gerrit for code review**
 - Advantages of an automated code review system
 - Gerrit installation and setup
 - Gerrit workflow
 - Gerrit code review
 - Outlook: Continuous build integration with Gerrit