

GIT- Einführung

Was ist Git?

Git ist ein Versionskontrollsystem. Es ist ein verteiltes System und findet in der Softwareentwicklung seine Anwendung. Jeder Entwickler verfügt bei Git über einen lokalen Klon des Repositorys und dessen gesamten Verlauf. Somit kann jeder lokal an dem Projekt arbeiten und Änderungen dann in das zentrale Repository hochladen. Git verwaltet und speichert alle vorigen Versionen, so kann jederzeit zu einer Vorgängerversion gewechselt werden.

Wie funktioniert Git?

1. Du erstellst ein "Repository" (Projekt) mit einem Git-Hosting-Tool (z. B. Bitbucket).
2. Du kopierst (oder klonst) das Repository auf dein lokales System.
3. Du fügst eine Datei deinem lokalen Repository hinzu und führst einen "Commit" für die Änderungen durch (d. h., du speicherst sie).
4. Du verschiebst deine Änderungen per "Push" in deinen Main-Branch.
5. Du nimmst mit einem Git-Hosting-Tool eine Änderung an der Datei vor und führst einen Commit dafür durch.
6. Du führst einen "Pull" durch, um die Änderungen auf dein lokales System zu übernehmen.
7. Du erstellst einen "Branch" (eine Version), nimmst eine Änderung vor und führst einen Commit dafür durch.
8. Du öffnest einen "Pull-Request" (Änderungsvorschläge für den Main-Branch).
9. Du führst einen "Merge" deiner Änderungen in den Main-Branch durch.

Git Begriffe

Pull:

Damit werden die Inhalte aus dem Remote-Repository heruntergeladen und aktualisieren das lokalen Repository, damit die Inhalte vom Klon und dem Main übereinstimmen

Push:

Damit werden alle Änderungen vom lokalen Repository in das Main Repository hochgeladen und für alle veröffentlicht.

Commit:

Ein Commit ist ein Kommentar zu einer vorgenommenen Änderung, welcher beim Push in das Main Repository mit gesendet wird. Commits sind Grundbausteine eines Projektes, sie bilden auch Snapshots und Meilensteine auf der Projektzeitleiste ab.

Branches:

Verweise auf einen Snapshot der bisherigen Änderungen. Git speichert keine Dateien vom Verzeichnis, sondern einen Branch als Verweis auf einen Commit. Ein Branch ist also die Spitze aufeinander folgender Commits.

Merge:

Es werden mehrere Commit Abfolgen in einen einheitlichen Verlauf zusammengeführt. Es dient dazu zwei Branches zu vereinen.