

Stoffabgrenzung Versionskontrolle

BBZW, Modul 426

Patrick Bucher

08.03.2024

Versionskontrolle

Motivation

1. Sie können die Gründe für die Nutzung einer Versionskontrolle benennen.
2. Sie können die Möglichkeiten und Grenzen einer “handgestrickten” Versionskontrolle, die einzig auf dem Dateisystem basiert, beschreiben.
3. Sie können die Anforderungen, welche eine Versionskontrolle für Dokumente und Quellcode erfordert, unterscheiden.
4. Sie können die Vor- und Nachteile einer zentralen gegenüber einer verteilten Versionskontrolle (und umgekehrt) benennen.

Unterlagen

- Folien “Versionskontrolle: Motivation”
- Aufgaben: “Versionskontrolle: Motivation”
- Skript “Versionskontrolle” (Kapitel 1: “Versionskontrolle: Motivation”)

Git: Grundlagen

1. Sie können die Designziele von Git benennen.
2. Sie können die Unterschiede zwischen Git und traditionellen Versionskontrollsystemen bezüglich deren interner Arbeitsweise nachvollziehen.
3. Sie können den Aufbau, das Konzept und den Zweck eines *Commit Hashes* erklären.
4. Sie können den Begriff *Repository* in Zusammenhang mit Git definieren.
5. Sie können die drei Bereiche eines Repositories (Arbeitsverzeichnis, Staging-Bereich und `.git`-Verzeichnis) voneinander unterscheiden.
6. Sie können die Vorteile des offiziellen Git-Clients für die Kommandozeile gegenüber GUI-Clients für Git benennen.

7. Sie können die drei Ebenen, auf denen Git konfiguriert werden kann, unterscheiden: systemweit, benutzerweit, pro Repository.
8. Sie wissen, warum man bei Git grundlegende Konfigurationseinstellungen (Name, E-Mail-Adresse, Texteditor) vornehmen muss.
9. Sie können die Befehle zur Anzeige der Git-Hilfe auf der Kommandozeile verwenden.

Unterlagen

- Folien “Versionskontrolle: Git (Einstieg)
- Skript “Versionskontrolle” (Kapitel 2: “Git: Grundlagen”)

Git: Verwendung

1. Sie können zwischen den beiden Hauptzuständen *tracked* und *untracked* und den drei *tracked*-Unterzuständen *unmodified*, *modified* und *staged* unterscheiden sowie die Befehle der entsprechenden Zustandsübergänge benennen und verwenden.
2. Sie verstehen das Konzept von `.gitignore` und können zwischen Artefakten unterscheiden, die in einem Repository enthalten oder davon ausgeschlossen sein sollen.
3. Sie können die Git-Befehle `clone`, `status`, `diff`, `add`, `commit`, `push`, `rm`, `mv`, `branch`, `checkout`, `merge`, `reset`, `log`, `tag` richtig verwenden.
4. Sie können erklären, warum man Dateien mit `git rm` löschen sollte.
5. Sie können erklären, warum man Dateien mit `git mv` umbenennen/verschieben sollte.
6. Sie verstehen das Konzept eines Remotes und wissen, wie es im Zusammenhang mit zentralen Repositories verwendet wird.
7. Sie können Branches erstellen, einen anderen Branch aktivieren, Änderungen zwischen Branches vergleichen und Branches zusammenführen.
8. Sie können Anwendungsfälle für Branches benennen und erklären.
9. Sie können den Workflow *Fork/Pull Request* erklären.

Unterlagen

- Folien “Versionskontrolle: Git 1” und “Versionskontrolle: Git 2”
- Aufgaben dazu in den Folien
- Skript “Versionskontrolle” (Kapitel 3: “Git: Verwendung”)