Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Eine Einführung in Git

Paul Nykiel

3. Mai 2020

Eine Einführung in Git Paul Nykiel	1 Einleitung
Einleitung	2 Konzept
Nutzung GitLab Git bei uns	3 Nutzung
Abschluss Praxis	4 GitLab
	5 Git bei uns
	6 Abschluss
	7 Praxis

Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Einleitung

Einleitung

Konzept

Nutzung

Nutzun

Git bei uns

.

Abschluss

Praxis

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

Nutzung

Git bei uns

Abschluss

Praxis

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

Nutzung

Git bei uns

Abschluss

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

Probleme:

Datei Austausch zwischen Nutzer

Nutzung

Git bei uns

Abschlus:

Praxis

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

- Datei Austausch zwischen Nutzer
- Aber: nicht sofort, erst wenn fertig

Git bei uns

Abschlus

Dravia

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

- Datei Austausch zwischen Nutzer
- Aber: nicht sofort, erst wenn fertig
- Alte Codestände sollten getestet werden können

Git bei uns

Abschlus

Dravia

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

- Datei Austausch zwischen Nutzer
- Aber: nicht sofort, erst wenn fertig
- Alte Codestände sollten getestet werden können

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Abscilluss

Praxis

• Authentisierung: von wem ist der Code

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Was noch?

- Authentisierung: von wem ist der Code
- Kein permanenter Internetzugriff

Einleitung

Konzept

Nutzung

Git bei uns

Abschluss

• Authentisierung: von wem ist der Code

- Kein permanenter Internetzugriff
- Einfache Nutzung

Einleitung

Konzept

Nutzung

Git bei uns

Abschluss

.

- Authentisierung: von wem ist der Code
- Kein permanenter Internetzugriff
- Einfache Nutzung
- Schnell

Einleitung

Konzept

Nutzung

Git bei uns

Abschluss

- Authentisierung: von wem ist der Code
- Kein permanenter Internetzugriff
- Einfache Nutzung
- Schnell
- Sicher

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

• Freie Software zur Versionsverwaltung



Konzept

Nutzung

 GitLab

Git bei uns

Abschluss

Praxis

- Freie Software zur Versionsverwaltung
- Dezentral



Konzept

Nutzung

GitLal

Git bei uns

Abschluss

Praxis

- Freie Software zur Versionsverwaltung
- Dezentral
- Wurde 2005 von Linus Torvals für Linux initiiert



Konzept

Nutzung

GitLal

Git bei uns

Abschluss

Pravis

- Freie Software zur Versionsverwaltung
- Dezentral
- Wurde 2005 von Linus Torvals für Linux initiiert
- Defakto Standard



Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung GitLab

Git bei uns

GIL Del UIIS

Abschluss

Praxis

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Aufbau Git

• Repository: Eine zusammenhänge Codebase/Projekt

Nutzung

Gitl ah

Git bei uns

Abschluss

. .

Aufbau Git

- Repository: Eine zusammenhänge Codebase/Projekt
- Ein Repository ist ein (azyklischer) gerichterer Graph aus Commits

Nutzung

Gitl ah

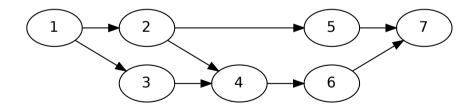
Git bei uns

Abschluss

Praxis

Aufbau Git

- Repository: Eine zusammenhänge Codebase/Projekt
- Ein Repository ist ein (azyklischer) gerichterer Graph aus Commits



Konzept Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Commit

Code-Stand

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

- Code-Stand
- Eindeutige Bezeichnung

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

- Code-Stand
- Eindeutige Bezeichnung
- Autor

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Commit

- Code-Stand
- Eindeutige Bezeichnung
- Autor
- Datum

Git bei uns

Commit

Code-Stand

Eindeutige

Bezeichnung

Autor

Datum

Nachricht

commit_feecb67fe1fa0490a2b836d5ba35da5812a3d27

Author: Paul Nykiel

Mon Apr 20 22:55:41 2020 +0200 Date:

Added ADTF type header

Nutzung

GitLab

Git bei uns

Abschluss

, (05011105

Praxis

Einen Commit anlegen

Working-Dir



File 2

File 3

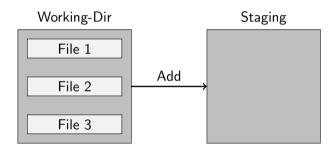
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



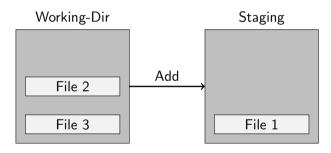
GitLab

Git bei uns

Abschluss

Abscillus

Praxis



Einleitung

Konzept

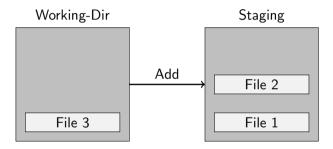
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



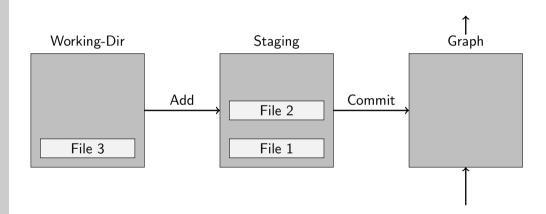
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



Einleitung

Konzept

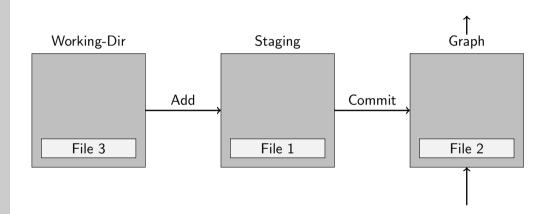
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



GitLab

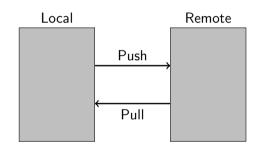
Git bei uns

Abschluss

Praxis

Remotes

• Bis jetzt alles lokal



Nutzung

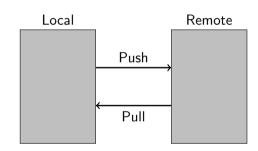
Git bei uns

Abschluss

Praxis

Remotes

- Bis jetzt alles lokal
- Codebase auf anderem Host: "Remote"



Nutzung

....

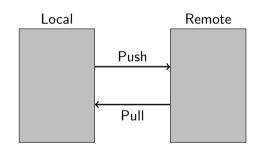
Git bei uns

Abschluss

Praxis

Remotes

- Bis jetzt alles lokal
- Codebase auf anderem Host: "Remote"
- Operationen: Commits von Remote kopieren, Commits zu Remote kopieren



C'...

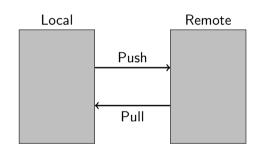
Git bei uns

Abschluss

Praxis

Remotes

- Bis jetzt alles lokal
- Codebase auf anderem Host: "Remote"
- Operationen: Commits von Remote kopieren, Commits zu Remote kopieren
- Oftmals ein zentraler Server



Konzept

Nutzung

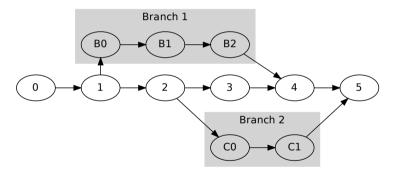
GitLab

Git bei uns

Abschluss

Praxis

Branches



Branches

Paul Nykiel

Einleitung

Konzept

Nutzung

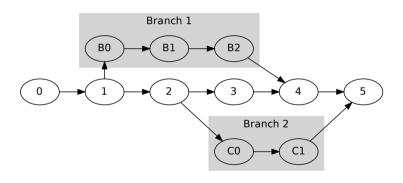
GitLak

Git bei uns

Abschluss

Abscilluss

Praxis



Ein Branch ist eine Liste von Commits



Eine Einführung in Git

Paul Nykiel

Einleitung Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Überblick

• Manipulation des Zustandes über Git (https://git-scm.com/)

C'...

Git bei uns

Abschluss

Abschluss

Pravis

Überblick

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments

GitLab

Git bei uns

Abschluss

, (100011110101

Praxis

Überblick

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:

Git bei uns

Abschluss

ADSCIIIUS:

Praxis

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:
 - add

Cirl

Git bei uns

Abschluss

Praxis

Überblick

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:
 - add
 - commit

Konzept

Nutzung

Abschluss

Git bei uns

Manipulation des Zustandes über Git (https://git-scm.com/)

• Nutzung: git command arguments

Häufige Befehle:

- add
- commit
- merge

Git bei uns

Abschluss

Abscillus.

Praxis

Überblick

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:
 - add
 - commit
 - merge
 - checkout

Konzept

 $\mathsf{Nutzung}$

Git bei uns

Abschluss

Praxis

• Manipulation des Zustandes über Git (https://git-scm.com/)

• Nutzung: git command arguments

Häufige Befehle:

- add
- commit
- merge
- checkout
- push

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:
 - add
 - commit
 - merge
 - checkout
 - push
 - pull

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Pravis

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:
 - add
 - commit
 - merge
 - checkout
 - push
 - pull
 - status

Konzept

Nutzung

Git bei uns

Git bei un

Abschluss

Praxis

- Manipulation des Zustandes über Git (https://git-scm.com/)
- Nutzung: git command arguments
- Häufige Befehle:
 - add
 - commit
 - merge
 - checkout
 - push
 - pull
 - status
 - log

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

git add FILES...

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

git add FILES...

git add -A

Einleitung Konzept

Nutzung

GitLab Git bei uns

Abschluss

Praxis

git add FILES...

git add -A

git add -u

Konzept

NutzungGitLab

Git bei uns

Abschluss

Praxis

git commit -m "message"

Konzept

NutzungGitLab

Git bei uns

Abschluss

Praxis

git merge BRANCH

Paul Nykiel
Einleitung

Konzept

Nutzung

Git bei uns

GIL DEI UIIS

Abschluss

Dravia

git checkout IDENTIFIER mit Identifier: Branch Name, Commit-Hash, ...

inloitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

git checkout IDENTIFIER mit Identifier: Branch Name, Commit-Hash, ...

git checkout -b NEW_BRANCH

Konzept

NutzungGitLab

Git bei uns

Abschluss

Praxis

git push REMOTE BRANCH

Konzept

NutzungGitLab

Git bei uns

Abschluss

Praxis

git pull REMOTE BRANCH

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

git status

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

git status

git log

Paul Nykiel
Einleitung

Konzept

Nutzung

Git bei uns

Git bei uns

Abschluss

Praxis

Demo: Einfache Nutzung

Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

 GitLab

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

GitLab ist ein Git-Server ("Remote") mit Web-Frontend

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Features: Praxis

GitLab ist ein Git-Server ("Remote") mit Web-Frontend

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Abscilluss

Praxis

GitLab ist ein Git-Server ("Remote") mit Web-Frontend

Features:

• Zugriffskontrolle

Paul Nykiel

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Features:

Zugriffskontrolle

GitLab ist ein Git-Server ("Remote") mit Web-Frontend

Merge Requests

Einleitung Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Adschiuss

Praxis

GitLab ist ein Git-Server ("Remote") mit Web-Frontend

Features:

- Zugriffskontrolle
- Merge Requests
- Issue Tracker

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

GitLab ist ein Git-Server ("Remote") mit Web-Frontend

Features:

- Zugriffskontrolle
- Merge Requests
- Issue Tracker
- Continuous Integration

Paul Nykiel

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Demo: GitLab

Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Git bei uns

Konzept

Nutzung

Git bei uns

Praxis

Git-Flow

- Definition von Abläufen, die für stabilen, aber auch aktuellen Code sorgen sollen
- Zwei langläufige Branches: master und develop



Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Abschluss

Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Git bei uns

Abschluss

Abscilluss

Praxis

Repository-Einrichten

 \bullet Legt ein Repository mit eurem Namen in der Gruppe GitWorkshop an

Git bei uns

Abschluss

Abscilluss

Praxis

- Legt ein Repository mit eurem Namen in der Gruppe GitWorkshop an
- Erzeugt eine lokale Kopie des Repositories auf eurem Rechner (clone)

Konzept

Nutzung

Git bei uns

Absobluse

Abscilluss

Praxis

- Legt ein Repository mit eurem Namen in der Gruppe GitWorkshop an
- Erzeugt eine lokale Kopie des Repositories auf eurem Rechner (clone)
- Legt eine Datei .gitignore an

Git bei uns

Abechluse

Abscilluss

Praxis

- Legt ein Repository mit eurem Namen in der Gruppe GitWorkshop an
- Erzeugt eine lokale Kopie des Repositories auf eurem Rechner (clone)
- Legt eine Datei .gitignore an
- Legt die Branchstruktur für Git-Flow an

GitLab

Git bei uns

Abschluss

Praxis

- Legt ein Repository mit eurem Namen in der Gruppe GitWorkshop an
- Erzeugt eine lokale Kopie des Repositories auf eurem Rechner (clone)
- Legt eine Datei .gitignore an
- Legt die Branchstruktur für Git-Flow an
- Legt ein Issue an und setzt euch als Verantwortlichen

Paul Nykiel
Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

• Legt einen Feature Branch an

Git bei uns

Abschluss

Praxis

Ein erster MR

- Legt einen Feature Branch an
- Committed den Code vom C++-Workshop (Teil 2) auf den Feature-Branch

Einleitung

Konzept

Nutzung

Chlab

Git bei uns

Absolutes

Abscilluss

- Legt einen Feature Branch an
- Committed den Code vom C++-Workshop (Teil 2) auf den Feature-Branch
- Legt einen WIP-MR dafür an, setzt mich als Verantwortlichen

GitLab

Git bei uns

Abschluss

Praxis

Ein erster MR

- Legt einen Feature Branch an
- Committed den Code vom C++-Workshop (Teil 2) auf den Feature-Branch
- Legt einen WIP-MR dafür an, setzt mich als Verantwortlichen
- Ich werde euch Feedback in Form von Anmerkungen geben

Konzept

Nutzung

Git bei uns

Praxis

Fin erster MR

- Legt einen Feature Branch an
- Committed den Code vom C++-Workshop (Teil 2) auf den Feature-Branch
- Legt einen WIP-MR dafür an, setzt mich als Verantwortlichen
- Ich werde euch Feedback in Form von Anmerkungen geben
- Arbeitet die Anmerkungen ein

Paul Nykiel Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

• Fügt zu eurem WIP-MR den Code vom C++ Workshop Teil 3 hinzu

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

- Fügt zu eurem WIP-MR den Code vom C++ Workshop Teil 3 hinzu
- Ich werde wieder Feedback geben, das ihr einarbeiten sollt

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Abscilluss

- Fügt zu eurem WIP-MR den Code vom C++ Workshop Teil 3 hinzu
- Ich werde wieder Feedback geben, das ihr einarbeiten sollt
- Wenn alles OK ist bekommt ihr einen 🖒

Konzept

Nutzung

Git bei uns

Abschluss

Praxis

• Fügt zu eurem WIP-MR den Code vom C++ Workshop Teil 3 hinzu

- Ich werde wieder Feedback geben, das ihr einarbeiten sollt
- Wenn alles OK ist bekommt ihr einen 🖒
- Dann könnt ihr den MR mergen

Konzept

Nutzung

Git bei uns

Praxis

• Fügt zu eurem WIP-MR den Code vom C++ Workshop Teil 3 hinzu

• Ich werde wieder Feedback geben, das ihr einarbeiten sollt

• Wenn alles OK ist bekommt ihr einen 🖒

- Dann könnt ihr den MR mergen
- Aktualisiert euere lokalen Branches