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

Konzept

Nutzung

Git bei uns

Abschluss

Praxis

Wofür ein Versionsverwaltungssystem?

Situation: Mehrere Leute arbeiten über längere Zeit an einer Codebase

Probleme:

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

Probleme:

- 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

Probleme:

- 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

GILLAD

Git bei uns

Abschluss

Pravis

- 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

Paul Nykiel Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

• Freie Software zur Versionsverwaltung



Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

- Freie Software zur Versionsverwaltung
- Dezentral



Paul Nykiel Einleitung

Konzept

Nutzung

GitLal

Git bei uns

Abschluss

Praxis

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



Paul Nykiel Einleitung

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

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

• Repository: Eine zusammenhänge Codebase/Projekt

Nutzung

GitLab

Git bei uns

Abschluss

Dravie

Aufbau Git

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

Git bei uns

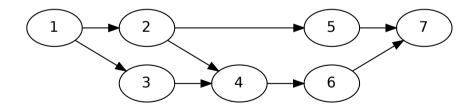
.

Abschluss

Praxis

Aufbau Git

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



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

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Commit

- Code-Stand
- Eindeutige Bezeichnung
- Autor
- Datum

....

Git bei uns

Abschluss

Praxis

• Code-Stand

• Eindeutige

Bezeichnung

Autor

Datum

Nachricht

Commit

 $\verb|commit feeecb| 67 fe1 fa0 490 a2b 836 d5b a35 da5812 a3d27|$

Author: Paul Nykiel

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

Added ADTF type header

GitLab

Git bei uns

Abschluss

Praxis

Einen Commit anlegen

Working-Dir



File 2

File 3

Konzept

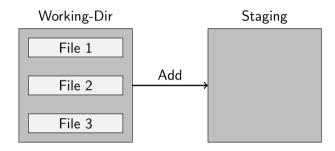
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



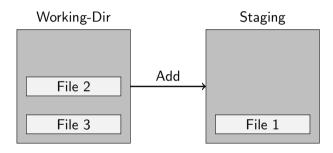
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



Einleitung

Konzept

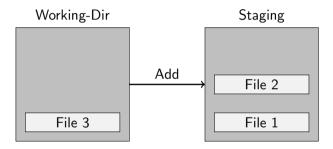
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



Einleitung

Konzept

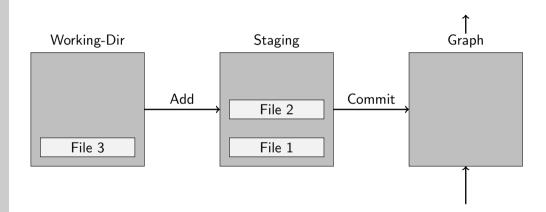
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



Einleitung

Konzept

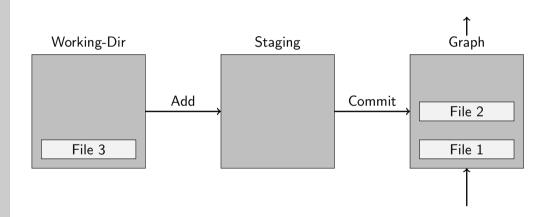
Nutzung

GitLab

Git bei uns

Abschluss

Praxis



GitLab

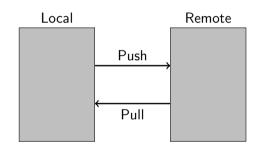
Git bei uns

Abschluss

Praxis

Remotes

• Bis jetzt alles lokal



Konzept

Nutzung

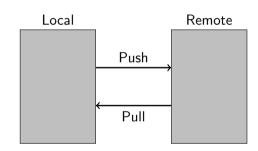
Git bei uns

Abschluss

Praxis

Remotes

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



Konzept

Nutzung

....

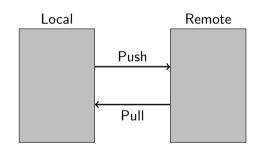
Git bei uns

Abschluss

Praxis

Remotes

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



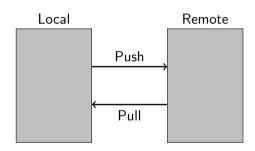
Git bei uns

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

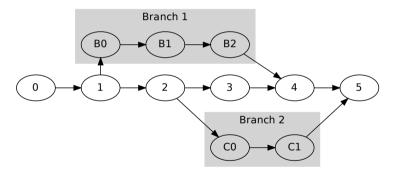
GitLab

Git bei uns

Abschluss

Praxis

Branches



Branches

Paul Nykiel

Einleitung

Konzept

Nutzung

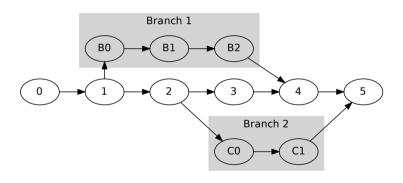
GitLak

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

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

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

OTTLUB

Git bei uns

Abschluss

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

Paul Nykiel

Konzept

 $\mathsf{Nutzung}$

Git bei uns

Abschluss

Abscilluss

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

Praxis

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

Konzept

Nutzung

GitLab

Git bei uns

.

Abschluss

Praxis

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

Features:

• Zugriffskontrolle

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

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

- Zugriffskontrolle
- Merge Requests

Paul Nykiel

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

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

- Zugriffskontrolle
- Merge Requests
- Issue Tracker

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

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

- 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

Paul Nykiel
Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

• Definition von Abläufen, die für stabilen, aber auch aktuellen Code sorgen sollen

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

• Definition von Abläufen, die für stabilen, aber auch aktuellen Code sorgen sollen

• Zwei langläufige Branches: master und develop

Einleitun

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

, (05011105.

- Definition von Abläufen, die für stabilen, aber auch aktuellen Code sorgen sollen
- Zwei langläufige Branches: master und develop
- Für jedes Feature: einen Feature-Branch auf Basis Develop

Konzept Nutzung

Git bei uns

Abschlus

.

- Definition von Abläufen, die für stabilen, aber auch aktuellen Code sorgen sollen
- Zwei langläufige Branches: master und develop
- Für jedes Feature: einen Feature-Branch auf Basis Develop



Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Merge Requests

Template

Paul Nykiel
Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

- Template
- Zwei Reviewer: Thematisch und C++

Paul Nykiel Einleitung

Konzept

Nutzung

Git bei uns

Praxis

Abschluss

- Template
- Zwei Reviewer: Thematisch und C++
- Continuous Integration:

Einleitung

Konzept

Nutzung

Git bei uns

Abschluss

- Template
- Zwei Reviewer: Thematisch und C++
- Continuous Integration:
 - Compilieren

Einleitung

Konzept

Nutzung

GILLAD

Git bei uns

Abschluss

Dravia

- Template
- Zwei Reviewer: Thematisch und C++
- Continuous Integration:
 - Compilieren
 - Unit-Tests

Einleitung

Konzept

Nutzung

Git bei uns

Abschluss

- Template
- Zwei Reviewer: Thematisch und C++
- Continuous Integration:
 - Compilieren
 - Unit-Tests
 - Statische Analyse (Sonarqube)

Git bei uns

Abschluss

Merge Requests

- Template
- Zwei Reviewer: Thematisch und C++
- Continuous Integration:
 - Compilieren
 - Unit-Tests
 - Statische Analyse (Sonargube)
 - Simulator-Tests

Konzept

Nutzung

Git bei uns

Abschluss

- Template
- Zwei Reviewer: Thematisch und C++
- Continuous Integration:
 - Compilieren
 - Unit-Tests
 - Statische Analyse (Sonargube)
 - Simulator-Tests
- Mergen nur von Administratoren

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Abscilluss

Praxis

Demo: Git-Flow

Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Abschluss

Paul Nykiel
Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

Praxis

Mehr Informationen:

• man git

Konzept

Nutzung

Git bei uns

Abschluss

Praxis

Mehr Informationen:

- man git
- https://git.spatz.wtf/spatzenhirn/wiki/-/wikis/Anleitungen/ Git%20Tutorial

Git bei uns

Abschluss

Abschluss

Mehr Informationen:

- man git
- https://git.spatz.wtf/spatzenhirn/wiki/-/wikis/Anleitungen/ Git%20Tutorial
- https://git.spatz.wtf/spatzenhirn/GitIntro

Konzept

Nutzung

Git bei uns

Abschluss

Mehr Informationen:

- man git
- https://git.spatz.wtf/spatzenhirn/wiki/-/wikis/Anleitungen/ Git%20Tutorial
- https://git.spatz.wtf/spatzenhirn/GitIntro

Was fehlt:

Konzept Nutzung

Git bei uns

Abschluss

Mehr Informationen:

- man git
- https://git.spatz.wtf/spatzenhirn/wiki/-/wikis/Anleitungen/ Git%20Tutorial
- https://git.spatz.wtf/spatzenhirn/GitIntro

Was fehlt: fast alles

Eine Einführung in Git

Paul Nykiel

Einleitung

Konzept

Nutzung GitLab

Git bei uns

Abschluss

Praxis

Git bei uns

Abschluss

Praxis

Repository-Einrichten

• Legt ein Repository mit eurem Namen in der Gruppe GitWorkshop an

Gitl ah

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)

. . .

Git bei uns

A I. . . I. I.

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

Konzept

Nutzung

Gitl ah

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

Citlab

Git bei uns

A I. . . I. I.

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
- 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

Einleitung

Konzept

Nutzung

GitLab

Git bei uns

Abschluss

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

Einleitun

Konzept

Nutzung

Chlab

Git bei uns

Abschluss

- 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

. . . .

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