



Jonas Michler

GitHub und GitLab

Plattformen für das Teilen von Code

GitHub und GitLab

Wer hat...

- ...davon bereits gehört?
- ...eines davon bereits genutzt?
- ...einen GitHub Account?

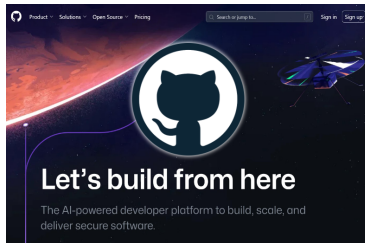
Table of contents

Was sind GitHub und GitLab?

kurz:

- Cloud-Dienste zur Versionsverwaltung für Software-Entwicklungsprojekte

Was ist Git?



Vorteile von GitHub und GitLab

■ Versionsverwaltung von Code

- ▶ Effiziente Verwaltung und Nachverfolgung von Codeänderungen
- ▶ Historie zur Wiederherstellung früherer Versionen

■ Kollaborative Softwareentwicklung

- ▶ Gleichzeitige Zusammenarbeit mehrerer Entwickler
- ▶ Effektive Zusammenarbeit in dezentralen Teams

■ Unterstützung von Open Source und Privaten Projekten

- ▶ Plattformen ermöglichen öffentliche und private Repositories
- ▶ Schutz des geistigen Eigentums in privaten Projekten

Cloud-basierte Versionskontrolle

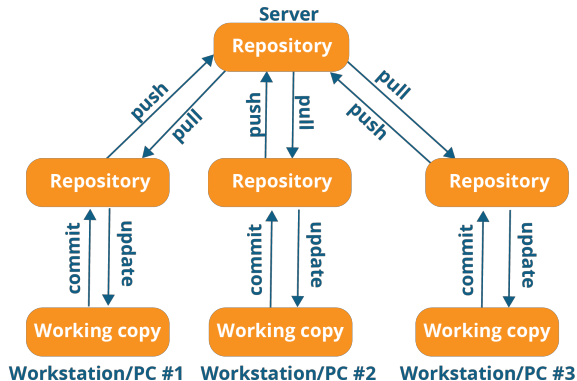
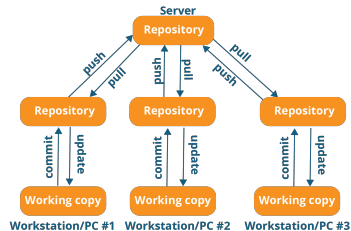


Figure: Distributed Version Control

Cloud-basierte Versionskontrolle

- Lokales Arbeitsrepository
- Änderungen können hochgeladen (gepusht) werden
- Änderungen von Teammitgliedern können aus der Cloud heruntergeladen werden
- Konflikte können auftreten, wenn mehrere Personen gleichzeitig arbeiten



GitHub vs. GitLab

GitHub

- Beliebte Plattform für Versionsverwaltung
- Hauptsächlich für Open Source-Projekte
- Funktionen: Issues, Pull Requests, Actions
- Cloud-basiert

GitLab

- Alternative mit ähnlichen Funktionen
- Geeignet für Open Source und private Projekte
- Umfassende CI/CD-Integration
- Cloud und lokale Installation

GitHub vs. GitLab

Hauptunterschiede:

GitHub

- Fokus auf Open Source
- Separate Actions
- Nur Cloud

GitLab

- Vielseitig verwendbar
- Integrierte CI/CD
- Cloud und lokal

GitHub

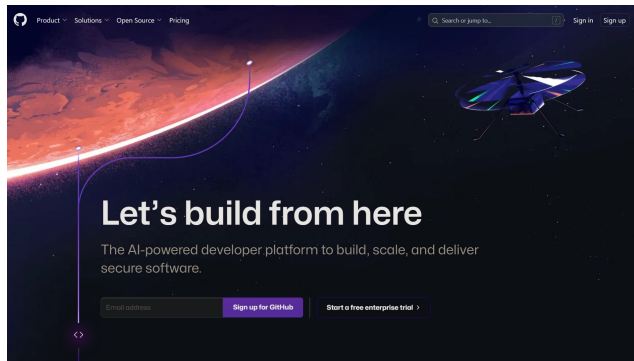


Figure: GitHub Homepage

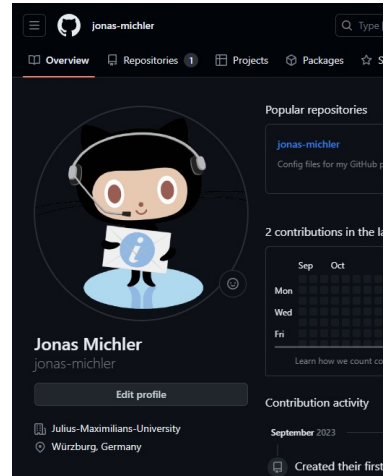
GitHub: Anmeldung

- 1 <https://github.com> besuchen
- 2 Schaltfläche "Sign Up"
- 3 E-Mail-Adresse + sicheres Passwort
- 4 Benutzernamen wählen
- 5 E-Mail verifizierung



GitHub: Profil einrichten

- 1 Profil-Icon (oben rechts), im Dropdown "Your profile"
- 2 Profilbild und eine kurze Beschreibung hinzufügen
- 3 Weitere Informationen hinzufügen
- 4 Soziale Links oder Website verknüpfen
- 5 Speichern



GitHub-Profil: Überblick

Was Ihr GitHub-Profil zeigt:

- Aktuelle Projekte
- Verwendete Programmiersprachen
- Ihre Follower und gefolgteten Entwickler

Warum ist es wichtig?

- Arbeitgeber und Teams prüfen oft GitHub-Profile
- Zeigt Leidenschaft für Softwareentwicklung
- Dient als Portfolio Ihrer Fähigkeiten
- Ermöglicht Networking und Zusammenarbeit

Repository-Oberfläche

- Code und Dateien im Repository
- README-Datei und Projektinformationen
- Contributors und Aktivitäten
- Branches und Pull Requests
- Projektstatistiken und Insights

Repos:

- `https://github.com/signalapp`
- `https://github.com/audacity/audacity`

GitLab: Repository-Oberfläche

The screenshot shows the GitLab repository page for `gitlab-org/gitlab`. The left sidebar contains navigation links for Project, Issues, Merge Requests, and other repository management tools. The main content area displays the repository overview, including statistics and a list of recent commits.

Repository Overview:

- Project ID: 278964
- 359,363 Commits
- 15,830 Branches
- 67,4 TiB Project Size
- 139 Veröffentlichungen
- 4,318 Umgebungen

Themen: `hacktoberfest`, `Ruby`, `Vue.js`, + 1 weitere

GitLab is an open source end-to-end software development platform with built-in version control, issue tracking, code review, CI/CD, and more. Self-host GitLab on your own servers, in a container, or on a cloud provider.

Recent Commits:

Name	Letzter Commit	Letzte Aktualisierung
<code>github</code>	Rename GitLab CE to FOSS in GitHub issue tem...	vor 3 Jahren
<code>gitlab</code>	Merge branch 'codeowners_auto_devops_temp...	vor 2 Tagen
<code>lefthook/pre-push</code>	Add security harness to Lefthook	vor 5 Monaten
<code>rubocop_todo</code>	Remove more redundant freeze	vor 3 Tagen
<code>.vscode</code>	Add GitLab Workflow as recommended VS Cod...	vor 1 Jahr
<code>app</code>	Merge branch 'smriti-422028/salesforce_logo_...	vor 1 Tag
<code>bin</code>	Add Vite for javascript assets	vor 2 Wochen

Figure: GitLab Repository

sample source: <https://gitlab.com/gitlab-org/gitlab>

GitLab: Profil-Oberfläche

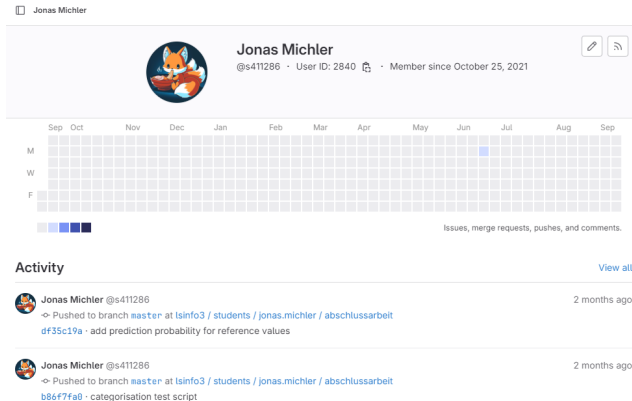


Figure: GitLab Profil

Contribution: Diverse Möglichkeiten zur Mitarbeit

■ **Code-Beitrag:**

- ▶ Schreiben neuer Funktionen
- ▶ Beheben von Fehlern
- ▶ Verbesserung der Dokumentation

■ **Fehlerberichte (Bug Reports):**

- ▶ Melden von Softwarefehlern oder Problemen
- ▶ Hilft bei der Fehlerbehebung und Verbesserung der Stabilität

■ **Support und Diskussion:**

- ▶ Beantworten von Fragen anderer Benutzer
- ▶ Teilnahme an Diskussionen und Entscheidungsfindung

Erstellen detaillierter Fehlerbeschreibungen/Issues

- Issues sind der Schlüssel zur Verfolgung von Fehlern und Verbesserungen
- Beschreiben Sie Fehler so detailliert wie möglich:
 - ▶ Was ist das fehlerhafte Verhalten?
 - ▶ Welche Schritte führen zum Fehler?
 - ▶ Was ist das zu erwartende Verhalten?
 - ▶ Welche Umgebung verwenden Sie?
- Klare Beschreibungen erleichtern das Verständnis und die Replizierbarkeit
- Issues sind **nicht** der Ort, um seine persönliche Meinung zum Produkt zu veröffentlichen

Code-Beiträge an spezifische Probleme/Tickets binden

- Beiträge, Forks und Pull Requests sollten immer zu einem Spezifischen Problem gehören
- Diese Verknüpfung ermöglicht eine klare Zuordnung von Änderungen
- Verbessert die Transparenz und Zusammenarbeit innerhalb des Projekts
- Viele Projekte besitzen eine Dokumentation, wie Beiträge einzureichen sind

Code-Beitrag auf GitHub/GitLab

■ Schritt 1: Fork des Projekts

- ▶ Erstellen einer Kopie des Repositorys auf Ihrem eigenen Konto

■ Schritt 2: Code ändern

- ▶ Bearbeiten des Codes in Ihrem Fork

■ Schritt 3: Pull Request erstellen

- ▶ Reichen Sie Ihre Änderungen als Pull Request (PR) beim Hauptprojekt ein

■ Schritt 4: Review Phase

- ▶ Andere Entwickler überprüfen und diskutieren Ihre Änderungen.

■ Schritt 5: Merge

- ▶ Wenn Ihre Änderungen genehmigt werden, werden sie in das Hauptprojekt integriert

Repository erstellen

- 1 Klicken Sie auf die Schaltfläche “New” oder “+”
- 2 Geben Sie einen Projektnamen und eine Beschreibung ein
- 3 Legen Sie die Sichtbarkeit (öffentlich oder privat) fest
- 4 Fügen Sie eine README-Datei hinzu, um Informationen über Ihr Projekt bereitzustellen

GitHub + GitLab: Ihre Plattformen für kollaborative Entwicklung



Warum GitHub und GitLab nutzen?

- Gemeinschaftliche Arbeit an einem Projekt
- Dokumentation von Entwicklungsschritten
- Öffentliche Bereitstellung des Source-Code
- Möglichkeit spezifisches Feedback zu erhalten