

# Werkzeuge

## Git und (andere) Kooperationswerkzeuge

Jörg Cassens, Jens Rademacher, Bastian Stender

### Medieninformatik Praktikum SoSe 2017



medieninformatik

IMAI – Institut für  
Mathematik und  
Angewandte Informatik

# Outline

## Vorab

Termine

Themenvorstellung

Themenvergabe

Andere Kurse

git: Theorie

git: Praxis

git: Tools

Projekt

Schluß

- 1 Vorab**
  - Termine
  - Themenvorstellung
  - Themenvergabe
  - Andere Kurse
- 2 git: Theorie
- 3 git: Praxis
- 4 git: Tools
- 5 Projektmanagement

## Vorab

Termine

Themenvorstellung

Themenvergabe

Andere Kurse

git: Theorie

git: Praxis

git: Tools

Projekt

Schluß

## Organisatorisches

- Campusfest 31.5./Kurstermine
- Themenvorstellung
- Themenvergabe
- Andere Kurse

## Werkzeuge

- Versionsverwaltung am Beispiel git
- Projektmanagement, Dokumentation, Kommunikation, Kooperation

Vorab

**Termine**

Themenvorstellung

Themenvergabe

Andere Kurse

gjt: Theorie

gjt: Praxis

gjt: Tools

Projekt

Schluß

## Termine

## Dates: Term Meetings

- 05.04., 12-14** ☒ kick-off, topics announced
- 12.04., 12-14** ☒ topics assigned, tools lecture
- 18./19.04.** ☐ group meetings
- 25./26.04.** ☐ group meetings
- 02./03.05.** ☐ group meetings
- 09./10.05.** ☐ group meetings
- 16./17.05.** ☐ group meetings
- 23./24.05.** ☐ group meetings
- 31.05., 12-18** ☒ mid-project presentations **Campusfest**
- 07.06.** ○ *no meetings (project week)*
- 13./14.06.** ☐ group meetings
- 20./21.06.** ☐ group meetings
- 27./28.06.** ○ *no meetings (conference)*
- 05.07., 12-18** ☒ end-project presentations

## Groups deciding to work during autumn break:

- 05.07., 12-18**  mid-project presentations
  - Meetings/Hangouts if needed
- 26.07.**  Status report (email)
- 16.08.**  Status report (email)
- 06.09.**  Status report (email)
- 27.09.**  Status report (email)
- 10.10.**  end-project presentations (time tbd)

## Deliverables for all groups:

- 10.04** ○ Outline of own project idea (email)
- 05.07.** ○ slides mid-project presentation (learnweb)
- 18.10.** ○ slides end-project presentation (learnweb)
- 18.10.** ○ project documentation (learnweb & paper)
- 18.10.** ○ artifact (how depends on artifact type)

Vorab

Termine

**Themenvorstellung**

Themenvergabe

Andere Kurse

gjt: Theorie

gjt: Praxis

gjt: Tools

Projekt

Schluß

## Themenvorstellung



Vorab

Termine

**Themenvorstellung**

Themenvergabe

Andere Kurse



git: Theorie

git: Praxis

git: Tools

Projekt

Schluß

-  Full descriptions
-  Short descriptions

Vorab

Termine

Themenvorstellung

**Themenvergabe**

Andere Kurse

gjt: Theorie

gjt: Praxis

gjt: Tools

Projekt

Schluß

## Themenvergabe

# Attendance List I

Vorab

Termine

Themenvorstellung

Themenvergabe

Andere Kurse

gjt: Theorie

gjt: Praxis

gjt: Tools

Projekt

Schluß

Please fill in the forms that are being handed out.

- MI done = Medieninformatik course passed
- CDIS done = Contextual Design of Interactive Systems passed (Medieninformatik II)
- Seminar done = Seminar Medieninformatik passed
- Vis now = Attending Data & Process Visualization
  
- IMIT = IMIT 😊
- AInf = Angewandte Informatik
- DA = Data Analytics
- WIN = Wirtschaftsinformatik
- Other = Other program (please specify)

## Attendance List II

Vorab

Termine

Themenvorstellung

Themenvergabe

Andere Kurse

gjt: Theorie

gjt: Praxis

gjt: Tools

Projekt

Schluß

Possible time slots for group meetings (any group or individual who cannot attend any of those?)

- Tuesdays or Wednesday, 11-12
- Tuesdays or Wednesday, 12-13
- Tuesdays or Wednesday, 13-14
- Tuesdays or Wednesday, 14-15
- Tuesdays or Wednesday, 15-16
- Tuesdays or Wednesday, 16-17
- Tuesdays or Wednesday, 17-18
- Tuesdays 18-19
  
- Time slot OK/preferred (please max 3 preferred)

Vorab

Termine

Themenvorstellung

Themenvergabe

**Andere Kurse**

gît: Theorie

gît: Praxis

gît: Tools

Projekt

Schluß

## Andere Kurse

Vorab

Termine

Themenvorstellung

Themenvergabe

Andere Kurse

gjt: Theorie

gjt: Praxis

gjt: Tools

Projekt

Schluß

## ■ Winter term

### ■ Media Informatics (Lec + Tut)

#### ■ German only

### ■ Seminar Media Informatics (Se)

## ■ Summer term

### ■ *Lab Course Media Informatics (Pr)*

### ■ Alternating:

#### ■ *Data and Process Visualization (Lec + Tut)*

#### ■ Contextualized Computing and Ambient Intelligent Systems (Lec + Tut)

#### ■ Contextual Design of Interactive Systems (Lec + Tut)

- **Data and Process Visualization**
  - First exam: Monday, 10.07., 14:00-16:00
  - Second exam: Wednesday, 11.10., 14:00-16:00
- **Contextual Design of Interactive Systems (Medieninformatik II)**
  - This year's exam: Wednesday, 14.06., 14:00-16:00
    - Binding registration by 07.06. in POS for this term
    - POS registration is opening 09.05.
  - Next exam: summer term 2018

# Outline

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

1 Vorab

2 git: Theorie

- Architekturen
- git

3 git: Praxis

4 git: Tools

5 Projektmanagement



# Nutzen von und Anforderungen an Versionsverwaltungen

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

- Verwaltung unterschiedlicher Versionen einer Datei

# Nutzen von und Anforderungen an Versionsverwaltungen

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

- Verwaltung unterschiedlicher Versionen einer Datei
- Protokollierung von Änderungen
  - Änderung,
  - Zeitpunkt,
  - Person
- Möglichkeit des Zugriffs auf vorherige Versionen
- Unterstützung mehrerer BenutzerInnen
- Unterstützung von Entwicklungszweigen, Merging, Datensicherheit

Vorab

git: Theorie

**Architekturen**

git

git: Praxis

git: Tools

Projekt

Schluß

## Architekturen

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

- Lokale Versionsverwaltung
  - Versionierung einer einzelnen Datei mit grundlegenden Verwaltungsfunktionen (Protokollierung, Abruf, Wiederherstellung)
  - **Implementierungen:** RCS, proprietäre Software
- Erste Generation, nicht für Gruppenarbeit geeignet

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

- **Zentrale Versionsverwaltung**
  - Einsatz eines zentralen Verwaltungsservers, Entwicklung auf Clients
  - Versionsgeschichte liegt auf dem Server
  - Rechteverwaltung auf dem Server
  - **Implementierungen:** CVS (eingestellt), SVN
- Zweite Generation, für Gruppenarbeit geeignet, auf Server angewiesen

- **Verteilte Versionsverwaltung**
  - Eigenes Repository auf jedem Client
  - Kein zentraler Server notwendig
  - Abgleich der Repositories untereinander
  - Versionsgeschichte kann lokal auf jedem Client liegen
  - Ermöglicht parallele Entwicklung mit anschließendem (möglichst weitgehend automatisiertem) Merge (“Nichtlineare Entwicklung”)
  - **Implementierungen:** Mercurial, Git
- Dritte Generation, für Gruppenarbeit geeignet, offline und nichtlineare Entwicklung

# Outline

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

git

# Eigenschaften

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

- Dezentrale Versionsverwaltung
- Entwickelt von Linus Torvalds für die Versionierung des Linux-Kernels
- Kein zentraler Server vorhanden
- Unterstützung nicht-linearer Programmierung durch Branching und Merging
- Unterstützung vieler Übertragungsprotokolle
- Revisions haben keine aufsteigende ID sondern Hash-Werte
- Authentifizierung bei Repository-Hostern häufig per Private Key



# Vor- und Nachteile

Vorab

git: Theorie

Architekturen

git

git: Praxis

git: Tools

Projekt

Schluß

## ■ Vorteile

- Kein Server notwendig
- keine verstreuten “.svn”-Ordner (lediglich ein .git-Ordner im Hauptverzeichnis)
- Effizienter Arbeit mit dem Repository (BRANCH, DIFF, MERGE)

## ■ Nachteile

- Disziplin notwendig
- Gewöhnungsbedürftiger Einsatz unter Windows
- “Unbequeme” Lernkurve
- Schlechte Performanz bei großen Dateien

# Outline

## 1 Vorab

## 2 git: Theorie

## 3 git: Praxis

- Repository anlegen
- Standard-Aufgaben
- Remote-Repositories
- Branching
- Sonstiges
- Tutorial

## 4 git: Tools

## 5 Projektmanagement

Vorab

git: Theorie

**git: Praxis**

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

Vorab

git: Theorie

git: Praxis

**Repository anlegen**

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

## Repository anlegen

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

Tutorial

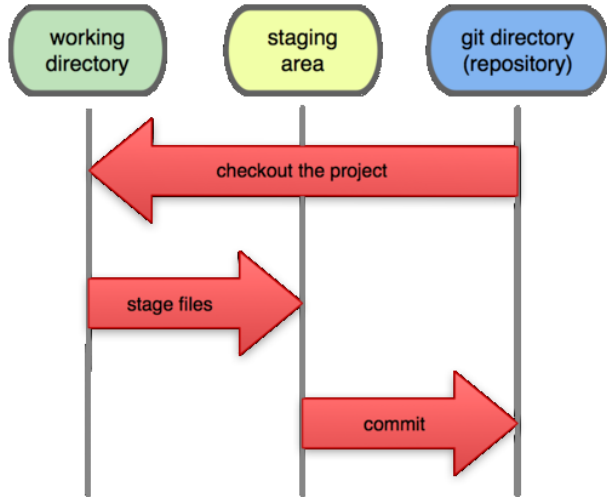
git: Tools

Projekt

Schluß

- Anlegen eines neuen Repositories:
  - **git init** (lokal im Projektordner, .git-Verzeichnis)
  - **git --bare init** (örtlich entferntes Repository)
- Dateien generell von der Versionierung ausschließen:
  - Anlegen einer Textdatei mit dem Namen **.gitignore** im Hauptverzeichnis
  - Eine Regel pro Zeile (etwa: **\*.pyc**)

# Local Operations



Workflow

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

Vorab

git: Theorie

git: Praxis

Repository anlegen

**Standard-Aufgaben**

Remote-Repositories

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

## Standard-Aufgaben

# Grundfunktionen

- File-Status:
  - **Untracked** (unbeobachtet, nicht Teil der Versionsverwaltung)
  - **Modified** (verändert gegenüber letztem Commit)
  - **Staged** (für den Commit vorgemerkt)
  - **Unmodified** (entspricht dem letzten „Committed“)
- Status auslesen:
  - **git status**
- Hinzufügen oder Stagen von Dateien:
  - **git add FILE(S)**
- Commit:
  - **git commit -m “Zusammenfassung“**
- Stage (deleted & modified) & Commit:
  - **git commit -a -m “Zusammenfassung“**
- Unterschiede zu HEAD anzeigen:
  - **git diff HEAD**

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

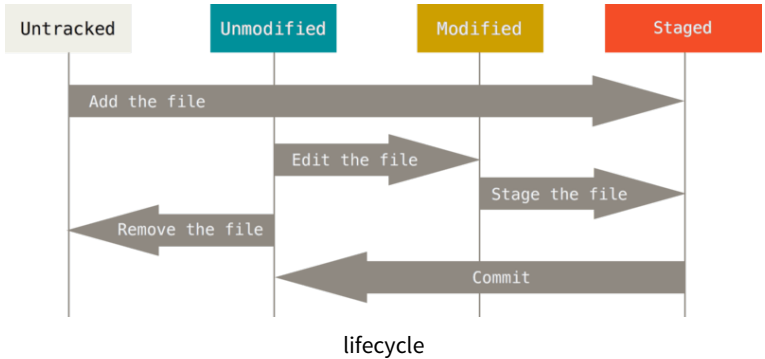
Sonstiges

Tutorial

git: Tools

Projekt

Schluß





# Änderungen Zurücknehmen

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

- Änderungen des letzten Commits rückgängig machen, Commit bleibt erhalten
  - **git revert**
- HEAD-pointer auf genannten Commit setzen
  - **git revert COMMIT**
- Stage/Arbeitsverzeichnis auf den Stand des letzten Commits setzen
  - **git reset**

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

**Remote-Repositories**

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

## Remote-Repositories

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

**Remote-Repositories**

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

- Lokale Kopie eines entfernten Repositories erstellen:
  - **git clone URL**
- Entferntes Repository hinzufügen:
  - **git remote add REMOTENAME URL**
- Lokale Änderungen in entferntes Repository übertragen:
  - **git push REMOTENAME BRANCH**
  - REMOTENAME ist häufig **origin**
  - Standard-Branch ist **master**

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

**Branching**

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

## Branching

# Branching

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

**Branching**

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

- Branch erstellen
  - **git branch NAME**
- Alle Branches anzeigen
  - **git branch -a**
- In Branch wechseln
  - **git checkout NAME**
- Branch mit Master mergen
  - **git checkout master** (in den Master-Zweig wechseln)
  - **git merge NAME** (Mergen)
- Branch löschen
  - **git branch -d NAME**

# Branches

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

**Branching**

Sonstiges

Tutorial

git: Tools

Projekt

Schluß



Arbeiten mit Branches

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

**Sonstiges**

Tutorial

git: Tools

Projekt

Schluß

## Sonstiges

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

Tutorial

git: Tools

Projekt

Schluß

- (Globale) Einstellungen festlegen:
  - **git config [--global] user.name “John Doe“**
  - **git config [--global] user.email john@example.com**
  - **git config [--global] core.autocrlf input** (Linux)
- Hilfe
  - **git COMMAND --help**



Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

Sonstiges

**Tutorial**

git: Tools

Projekt

Schluß

## Tutorial

Vorab

git: Theorie

git: Praxis

Repository anlegen

Standard-Aufgaben

Remote-Repositories

Branching

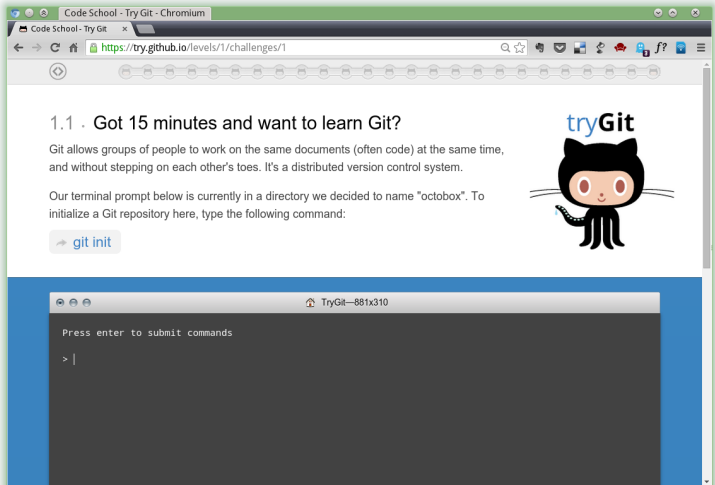
Sonstiges

**Tutorial**

git: Tools

Projekt

Schluß



Code School - Try Git - Chromium

Code School - Try Git

https://try.github.io/levels/1/challenges/1

## 1.1 - Got 15 minutes and want to learn Git?

tryGit

Git allows groups of people to work on the same documents (often code) at the same time, and without stepping on each other's toes. It's a distributed version control system.

Our terminal prompt below is currently in a directory we decided to name "octobox". To initialize a Git repository here, type the following command:

```
git init
```

Press enter to submit commands

```
> |
```

 [try.github.com](https://try.github.com)

# Outline

Vorab

git: Theorie

git: Praxis

**git: Tools**

git: GUI

git: Project Hosting

Projekt

Schluß

1 Vorab

2 git: Theorie

3 git: Praxis

**4 git: Tools**

- git: GUI
- git: Project Hosting

5 Projektmanagement

Vorab

git: Theorie

git: Praxis

git: Tools

**git: GUI**

git: Project Hosting

Projekt

Schluß

## git: GUI

Vorab

git: Theorie

git: Praxis

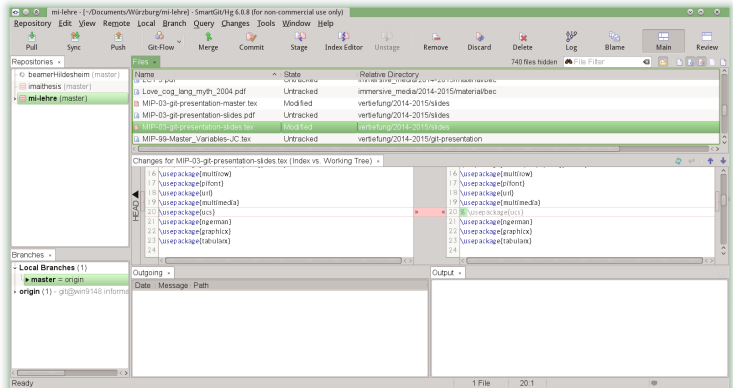
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß



👉 [www.syntevo.com/smartgit](http://www.syntevo.com/smartgit)

Cross-Platform (Linux, Mac, Windows), free for non-commercial use

Vorab

git: Theorie

git: Praxis

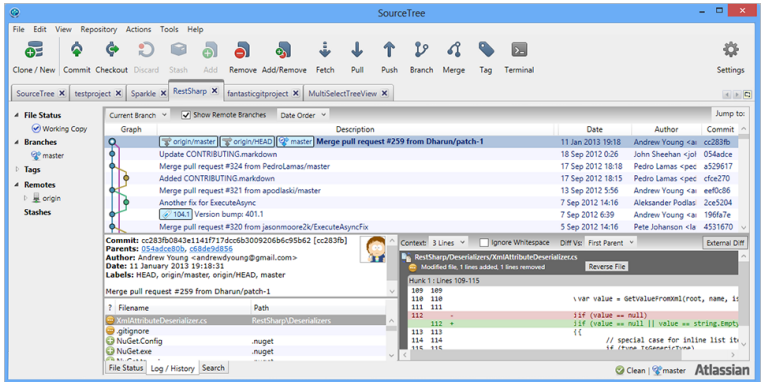
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluss



 [SourceTreeApp.com](http://SourceTreeApp.com)

Mac, Windows; free to use, registration required

# GitKraken

Vorab

git: Theorie

git: Praxis

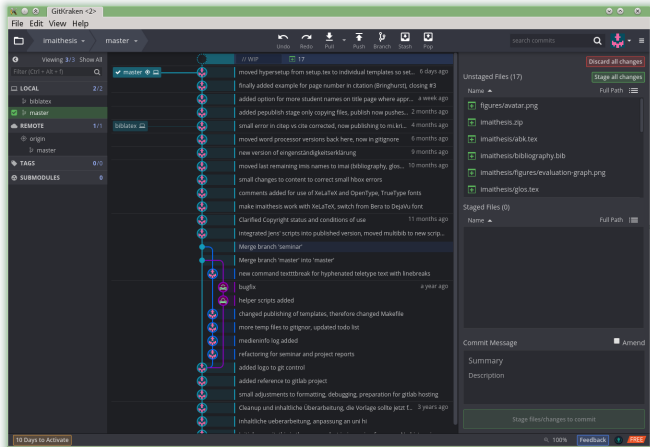
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß



 [gitkraken.com](https://gitkraken.com)

Cross-Platform (Linux, Mac, Windows), free for non-commercial use,  
registration required

Vorab

git: Theorie

git: Praxis

git: Tools

git: GUI

**git: Project Hosting**

Projekt

Schluß

## git: Project Hosting



# Project Hosting

Vorab

git: Theorie

git: Praxis

git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß

- Hilfen für Projekte durch:
  - Issuetracker
  - Wiki (Markdown)
  - Statistiken (Gamification)
  - Download von Projekten
  - Releases
- Gruppenarbeit
- Forks und Pull-Requests
  - Möglichkeit, sich einfach in neue Projekte einzubringen
  - “Standing on the shoulders of giants”
- Mehrere Services mit unterschiedlichen Vor- und Nachteilen
  - gitlab
  - github
  - bitbucket

# gitlab (Uni Hildesheim)

Vorab

git: Theorie

git: Praxis

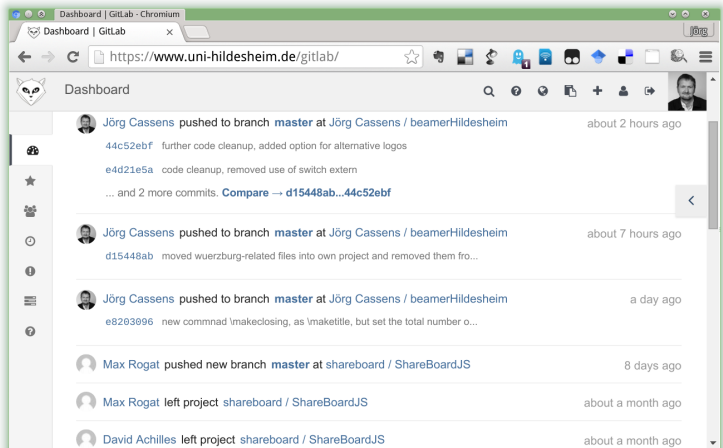
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß



[www.uni-hildesheim.de/gitlab](https://www.uni-hildesheim.de/gitlab)

Hosted OSS-System

# gitlab (Kommerziell)

Vorab

git: Theorie

git: Praxis

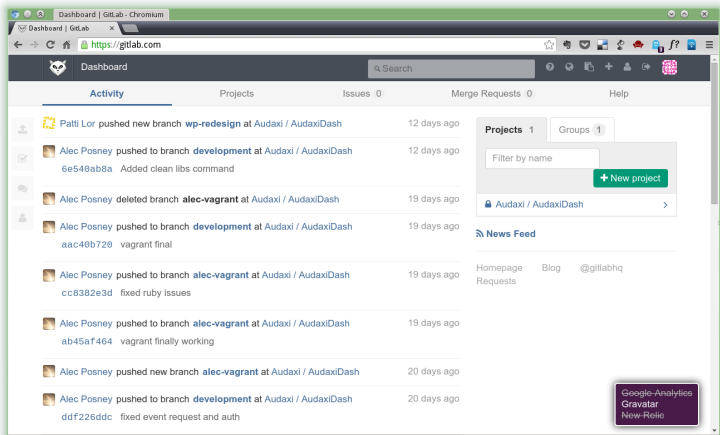
git: Tools

git: GUI

git: Project Hosting


Projekt

Schluß



The screenshot shows the GitLab dashboard for a user named 'Audaxi / AudaxiDash'. The main activity feed lists several recent pushes and deletions of branches. On the right, there is a 'Projects' section with a search filter and a '+ New project' button. Below that is a 'News Feed' section with links to 'Homepage Requests', 'Blog', and '@gitlabhq'. A 'Geogly-Analytics' badge is visible in the bottom right corner.

Activity	Projects	Issues	Merge Requests	Help
Patti Lor pushed new branch <b>wp-redesign</b> at Audaxi / AudaxiDash	12 days ago	0	0	
Alec Posney pushed to branch <b>development</b> at Audaxi / AudaxiDash 6e548ab8a Added clean libs command	12 days ago			
Alec Posney deleted branch <b>alec-vagrant</b> at Audaxi / AudaxiDash	19 days ago			
Alec Posney pushed to branch <b>development</b> at Audaxi / AudaxiDash aac40b720 vagrant final	19 days ago			
Alec Posney pushed to branch <b>alec-vagrant</b> at Audaxi / AudaxiDash cc8382e3d fixed ruby issues	19 days ago			
Alec Posney pushed to branch <b>alec-vagrant</b> at Audaxi / AudaxiDash ab45af464 vagrant finally working	19 days ago			
Alec Posney pushed new branch <b>alec-vagrant</b> at Audaxi / AudaxiDash	20 days ago			
Alec Posney pushed to branch <b>development</b> at Audaxi / AudaxiDash ddf226ddc fixed event request and auth	20 days ago			

 gitlab.com

Kommerzielle Variante, hosted

# gitlab (OSS)

Vorab

git: Theorie

git: Praxis

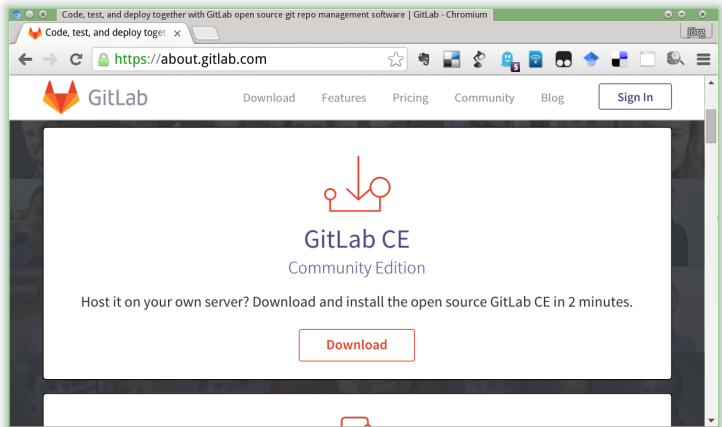
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß



 [about.gitlab.com](https://about.gitlab.com)

Self-Hosted OSS-System

Vorab

git: Theorie

git: Praxis

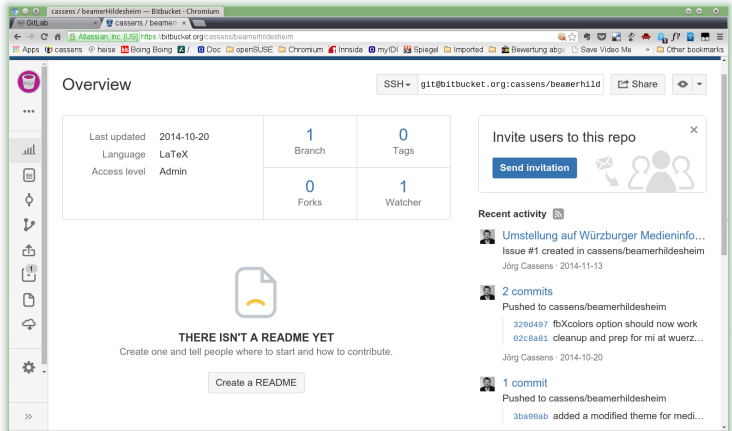
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß



The screenshot shows the Bitbucket 'Overview' page for the repository 'cassens/beamerHildesheim'. The page includes a navigation sidebar on the left, a main content area with repository statistics, and a 'Recent activity' section on the right.


Repository Information		1	0
Last updated	2014-10-20	Branch	Tags
Language	LaTeX	0	1
Access level	Admin	Forks	Watcher

**THERE ISN'T A README YET**  
Create one and tell people where to start and how to contribute.

[Create a README](#)

**Recent activity**

- Umstellung auf Würzburger Medieninfo...**  
Issue #1 created in cassens/beamerhildesheim  
Jörg Cassens · 2014-11-13
- 2 commits**  
Pushed to cassens/beamerhildesheim  
328d497 fbXcolors option should now work  
02c8a81 cleanup and prep for mi at wuerz...  
Jörg Cassens · 2014-10-20
- 1 commit**  
Pushed to cassens/beamerhildesheim  
3ba00ab added a modified theme for medi...

 bitbucket.org

Kommerziell, hosted, Freemium

Vorab

git: Theorie

git: Praxis

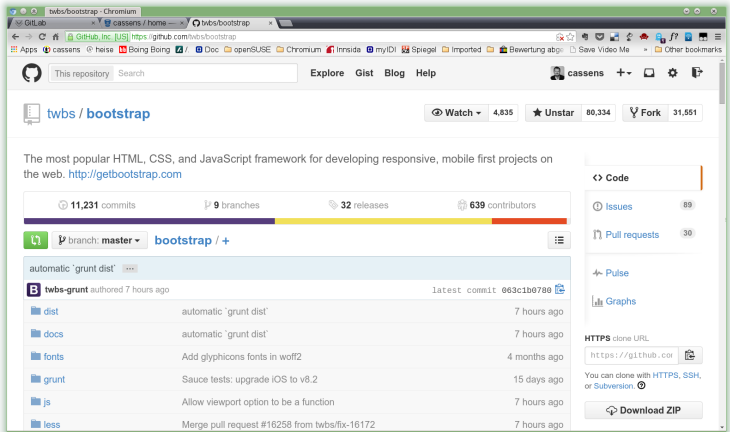
git: Tools

git: GUI

git: Project Hosting

Projekt

Schluss



[www.github.com](http://www.github.com)

Kommerziell, hosted, Freemium

Vorab

git: Theorie

git: Praxis

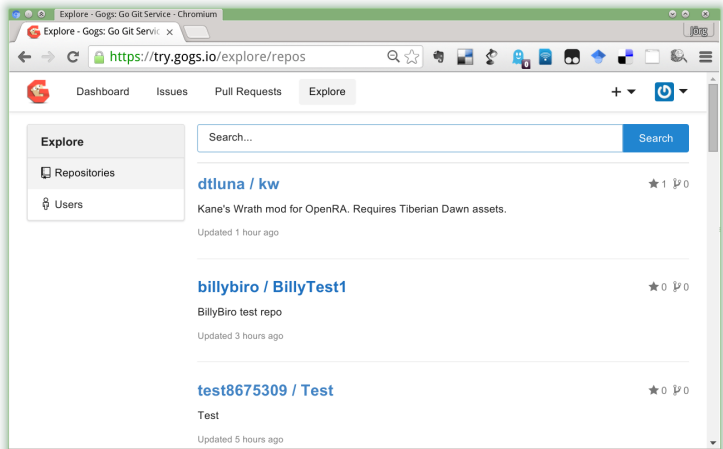
git: Tools


git: GUI

git: Project Hosting

Projekt

Schluß



 gogs.io

Self-Hosted OSS-System

## Verwendung im Praktikum

- Die einzelnen Gruppen sollen jeweils ein Projekt bei einem der genannten Services anlegen
  -  [uni-hildesheim.de/gitlab](https://uni-hildesheim.de/gitlab) – Universität Hildesheim
  -  [www.gitlab.com](https://www.gitlab.com)  
kostenlose öffentliche und private Projekte
  -  [www.bitbucket.com](https://www.bitbucket.com)  
kostenlose öffentliche und private Projekte, kostenlos nur eingeschränkte Teamgrößen
  -  [www.github.com](https://www.github.com)  
kostenlos nur für öffentliche Projekte
- Zu diesem Projekt werde ich eingeladen
  - Zugriff auf Code und Dokumentation
  - Möglichkeit, Tickets zu erstellen
- Ich empfehle das gitlab der Uni
- Wer das nicht benutzen möchte (Zugriff, Uptime) sollte eines der anderen Systeme nutzen

Vorab

git: Theorie

git: Praxis

git: Tools

git: GUI

git: Project Hosting

Projekt

Schluß



# Outline

1 Vorab

2 git: Theorie

3 git: Praxis

4 git: Tools

**5 Projektmanagement**

- Ticketing und Projektplanung
- Dokumentation
- Kommunikation & Koordination
- Automatisierung
- Empfehlung

Vorab

git: Theorie

git: Praxis

git: Tools

**Projekt**

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

# Bedarf

- Versionsverwaltung
  - Wie gesehen
- Ticketing
  - Bei den genannten Hosting-Providern im Grundsatz enthalten
- Projektplanung
  - Zumindest: Meilensteine und Ticketing
- Dokumentation
  - Wiki bei den genannten Providern
  - $\LaTeX$  im git
  - Kollaborative Editoren
- Kommunikation & Koordination
  - Mehr als facebook und Dropbox
- Automatisierung
  - Wenn im git-repo etwas passiert, können auch andere Dinge angestoßen werden

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

**Ticketing und  
Projektplanung**

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

## Ticketing und Projektplanung

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- Die git-hoster stellen in der Regel ein eigenes Ticket-System zur Verfügung
- Zusammen mit Meilensteinen ist rudimentäre Projektplanung möglich
- Vorteile
  - Arbeit mit vorhandenen Werkzeugen
- Nachteile
  - In der Regel nicht sehr flexibel

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

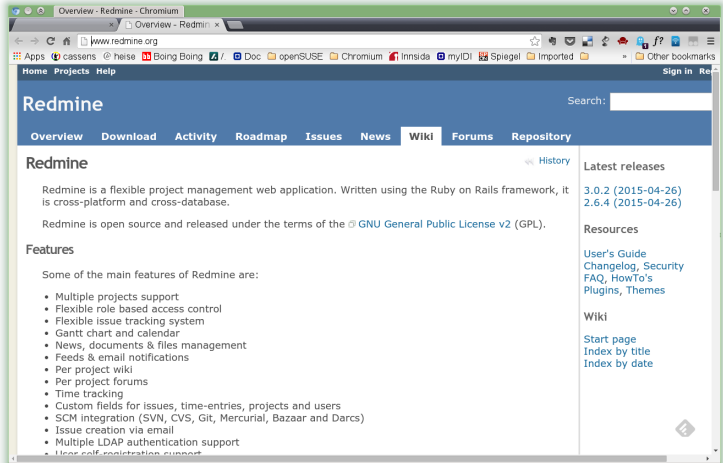
Dokumentation


Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluss



 [www.redmine.org](http://www.redmine.org)  
integriertes System

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung


Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- Umfangreiches integriertes System
  - Projektmanagement
  - Zeiterfassung
  - Dokumentation
  - git-Repositories
- Vorteile
  - Viele Möglichkeiten, erweiterbar
- Nachteile
  - Kein reiner Projektplaner
  - Self-Hosting
- Alternative
  -  trac

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

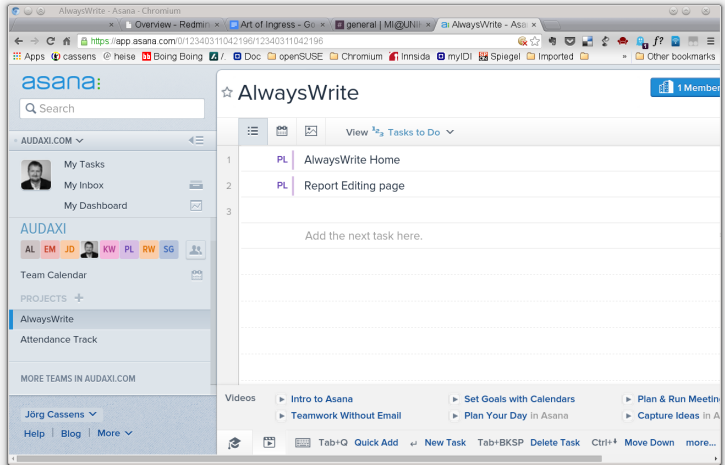
Dokumentation


Kommunikation &  
Koordination

Automatisierung

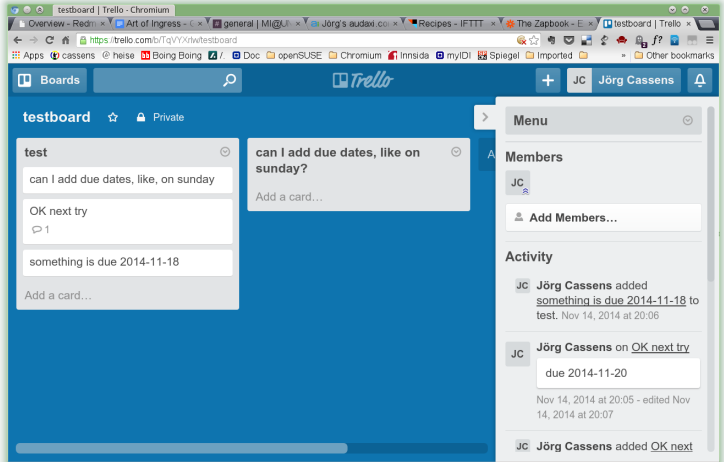
Empfehlung

Schluß



 asana.com

Task Management, Freemium



 trello.com

Kanban-Style Task Management, Freemium



Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

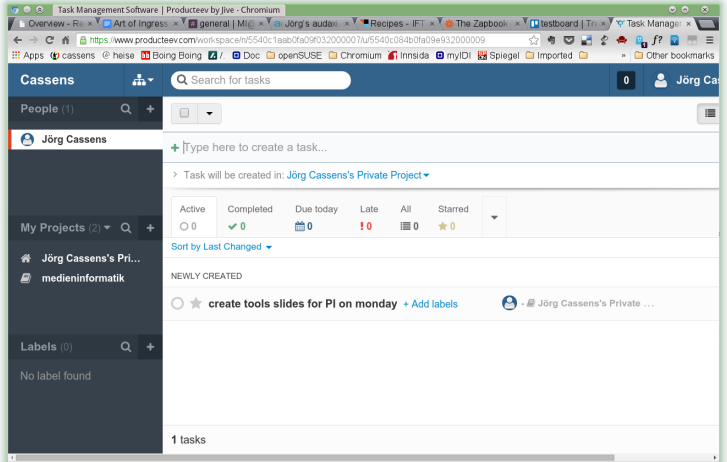
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



[www.producteev.com](http://www.producteev.com)

Task Management, Freemium

# asana, Trello, producteev

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- Fokus auf Projektmanagement
- Unterschiedliche Philosophien (kanban, “traditionell”)
- Vorteile
  - Mächtigkeit
- Nachteile
  - Mächtigkeit

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

**Dokumentation**

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

## Dokumentation

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- git funktioniert sehr gut bei Textdateien
- Für Binary Blobs wie PDF oder Dateien von Textverarbeitungen eher ungeeignet
- L<sup>A</sup>T<sub>E</sub>X ist textbasiert
  - Dokumentation in einem doc-repo
- Nachteile
  - Assets wie Bilder werden semi-optimal behandelt
  - Änderungsverfolgung im Text
    - mit etwas Handarbeit mit latexdiff realisierbar
- Vorteile
  - Professionelle Satzqualität
  - Handhabung mit den gleichen Werkzeugen

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

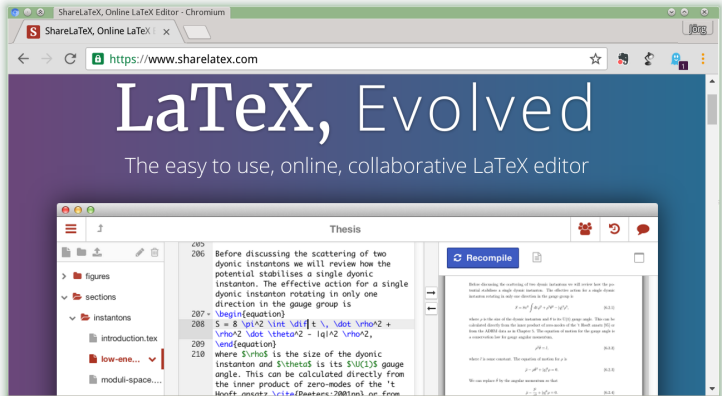
Dokumentation

Kommunikation &  
Koordination

Automatisierung

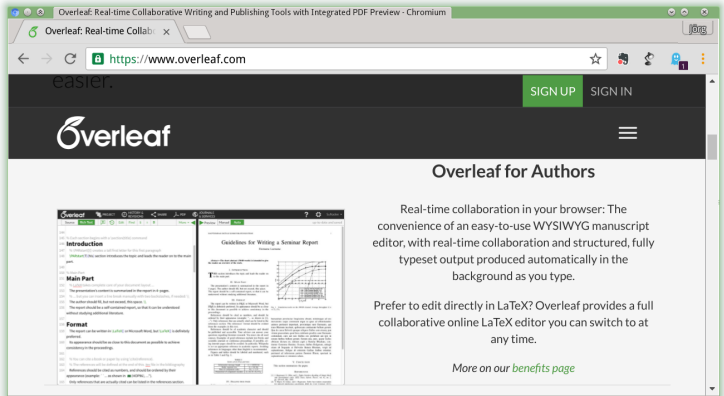
Empfehlung

Schluss



 [sharelatex.com](https://sharelatex.com)

Collaborative (nur in bezahlter Version), online  $\text{\LaTeX}$ -shell, Freemium



 [overleaf.com](https://www.overleaf.com)

Collaborative, online  $\text{\LaTeX}$ -shell, Freemium

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- “Mitgelieferte” Wikis verwenden
- Vorteile
  - Einfache Markdown-Syntax
  - Online
  - Gleiche Werkzeuge
- Nachteile
  - Einfache Markdown-Syntax
  - Online
- Aus Markdown lassen sich mit verschiedenen Werkzeugen andere Formate erzeugen (HTML, PDF,  $\text{\LaTeX}$ , ODF)
  - Beispiel pandoc, [👉 johnmacfarlane.net/pandoc](http://johnmacfarlane.net/pandoc)
  - Beispiel MultiMarkdown, [👉 fletcherpenney.net/multimarkdown](http://fletcherpenney.net/multimarkdown)

# Etherpad

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

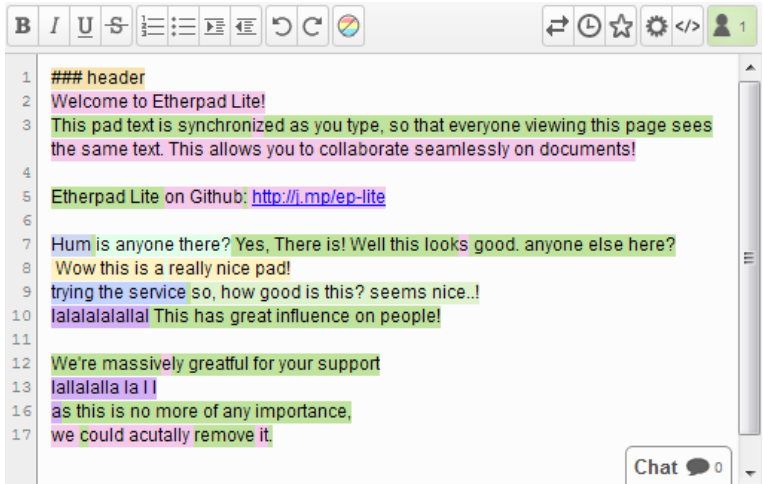
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



The screenshot shows the Etherpad interface. At the top is a toolbar with icons for bold (B), italic (I), underline (U), strikethrough (ABC), bulleted list, numbered list, indent, outdent, undo, redo, and a color picker. To the right of the toolbar are icons for share, clock, star, settings, code, and a user profile icon labeled '1'. The main editing area contains the following text:

```

1  ### header
2  Welcome to Etherpad Lite!
3  This pad text is synchronized as you type, so that everyone viewing this page sees
   the same text. This allows you to collaborate seamlessly on documents!
4
5  Etherpad Lite on Github: http://i.mp/ep-lite
6
7  Hum is anyone there? Yes, There is! Well this looks good. anyone else here?
8  Wow this is a really nice pad!
9  trying the service so, how good is this? seems nice..!
10 lalalalalal This has great influence on people!
11
12 We're massively grateful for your support
13 lalalalla la l l
16 as this is no more of any importance,
17 we could acutally remove it.

```

At the bottom right of the editing area is a chat window with the text "Chat" and a speech bubble icon containing the number "0".

etherpad.org, epad.hosting.uni-hildesheim.de

OSS Collaborative online Text Editor, hosted oder self-hosted



Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

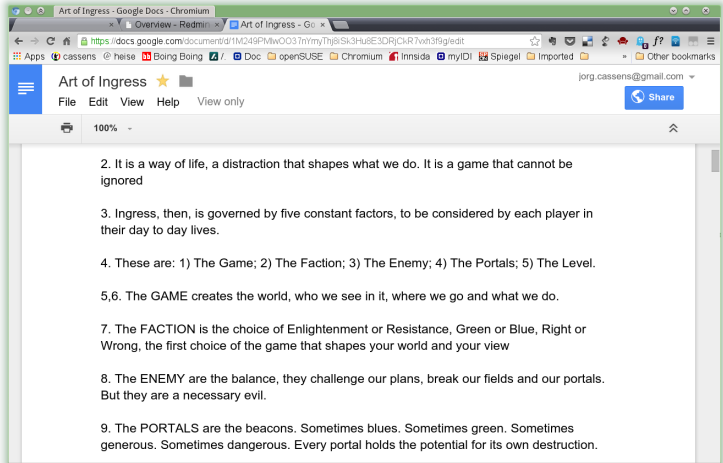
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



Art of Ingress - Google Docs - Chromium

Overview - Redmin x Art of Ingress - Go x

https://docs.google.com/document/d/1M249PMlwO037nYmyThj8Sk3Hu8E3DRjCkR7wh3f9gledit

Apps cassens @ heise Boing Boing Doc openSUSE Chromium Innsida myID! Spiegel Imported Other bookmarks

Art of Ingress ☆

File Edit View Help View only

100%

2. It is a way of life, a distraction that shapes what we do. It is a game that cannot be ignored

3. Ingress, then, is governed by five constant factors, to be considered by each player in their day to day lives.


4. These are: 1) The Game; 2) The Faction; 3) The Enemy; 4) The Portals; 5) The Level.

5,6. The GAME creates the world, who we see in it, where we go and what we do.

7. The FACTION is the choice of Enlightenment or Resistance, Green or Blue, Right or Wrong, the first choice of the game that shapes your world and your view

8. The ENEMY are the balance, they challenge our plans, break our fields and our portals. But they are a necessary evil.

9. The PORTALS are the beacons. Sometimes blues. Sometimes green. Sometimes generous. Sometimes dangerous. Every portal holds the potential for its own destruction.

 docs.google.com

Collaborative online Word Processor

# Microsoft Word Online

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

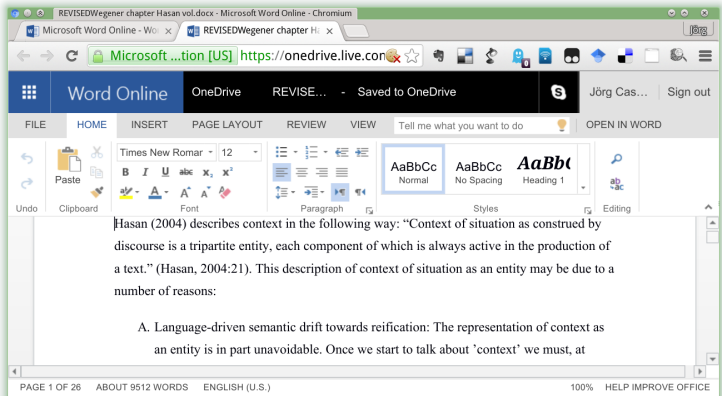
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



 office.live.com

Collaborative online Word Processor

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

**Kommunikation &  
Koordination**

Automatisierung

Empfehlung

Schluß

## Kommunikation & Koordination

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

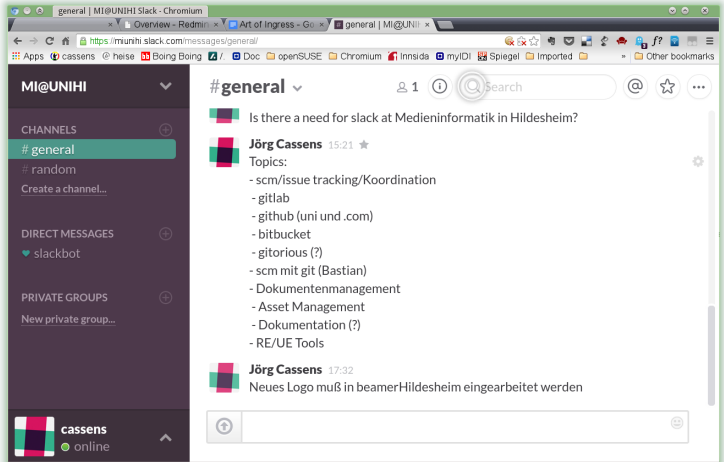
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



general | MI@UNIHI Slack - Chromium

https://miunhi.slack.com/messages/general/

#general

Is there a need for slack at Medieninformatik in Hildesheim?

**Jörg Cassens** 15:21 ★

Topics:

- scm/issue tracking/Koordination
- gitlab
- github (uni und .com)
- bitbucket
- gitorious (?)
- scm mit git (Bastian)
- Dokumentenmanagement
- Asset Management
- Dokumentation (?)
- RE/UE Tools

**Jörg Cassens** 17:32

Neues Logo muß in beamerHildesheim eingearbeitet werden

cassens online

 slack.com

Kommerziell, Freemium

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- Kommunikation in Teams in sogenannten Kanälen
- Strukturierter als Hangouts, übersichtlicher als facebook, einfacher als IRC
- Vorteile
  - Diverse Funktionen und Hooks
  - Gute Verbindung mit anderen Systemen (git commit-messages)
  - Kaum Einschränkungen für kostenlose Accounts
- Nachteile
  - ...

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

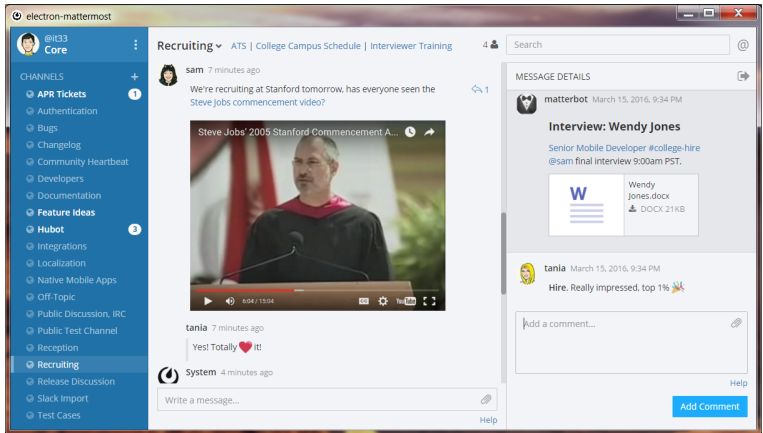
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



The screenshot shows the Mattermost web interface. On the left is a sidebar with a list of channels under the heading 'CHANNELS'. The selected channel is 'Recruiting'. The main content area shows a message from user 'sam' posted 7 minutes ago: 'We're recruiting at Stanford tomorrow, has everyone seen the Steve Jobs commencement video?'. Below the text is a video player showing a still from the 'Steve Jobs' commencement video. Below the video, user 'tania' has commented 'Yes! Totally ❤️!t!'. At the bottom of the message area is a text input field with the placeholder 'Write a message...'. On the right side, the 'MESSAGE DETAILS' panel is open, showing the message content and a file attachment named 'Wendy Jones.docx' (21 KB). Below the details is a comment input field with the placeholder 'Add a comment...' and an 'Add Comment' button.

 mattermost.org

Self-hosted Slack Clone, bei gitlab “dabei”

# Telegram

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

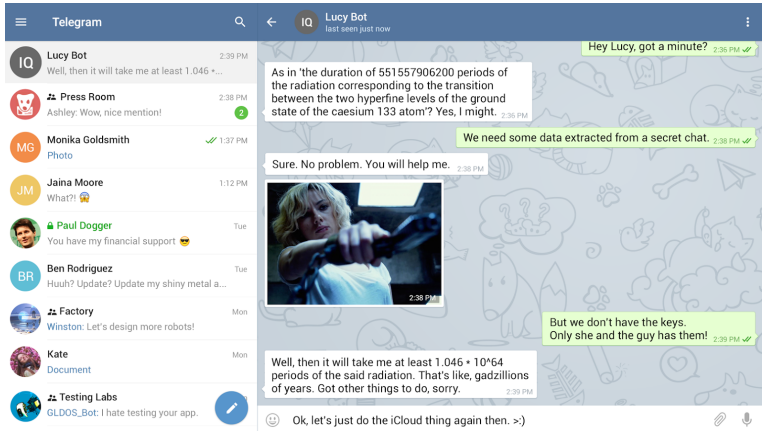
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



The screenshot shows the Telegram app interface. On the left is a list of contacts and groups:

- Lucy Bot** (2:39 PM): Well, then it will take me at least  $1.046 \cdot 10^{64}$ ...
- Press Room** (2:38 PM): Ashley: Wow, nice mention! (2 unread messages)
- Monika Goldsmith** (1:37 PM): Photo
- Jaina Moore** (1:12 PM): What?! 🤔
- Paul Dogger** (Tue): You have my financial support 😊
- Ben Rodriguez** (Tue): Huuh? Update? Update my shiny metal a...
- Factory** (Mon): Winston: Let's design more robots!
- Kate** (Mon): Document
- Testing Labs** (blue icon): GLDOS\_Bot: I hate testing your app. (blue pencil icon)

The main chat area is with **Lucy Bot** (last seen just now). The chat history includes:

- Green bubble: Hey Lucy, got a minute? (2:36 PM ✓)
- Text: As in 'the duration of  $551557906200$  periods of the radiation corresponding to the transition between the two hyperfine levels of the ground state of the caesium 133 atom'? Yes, I might. (2:36 PM)
- Green bubble: We need some data extracted from a secret chat. (2:38 PM ✓)
- Text: Sure. No problem. You will help me. (2:38 PM)
- Image: A photo of a blonde woman (Lucy Liu) pointing a handgun. (2:38 PM)
- Green bubble: But we don't have the keys. Only she and the guy has them! (2:39 PM ✓)
- Text: Well, then it will take me at least  $1.046 \cdot 10^{64}$  periods of the said radiation. That's like, gazillions of years. Got other things to do, sorry. (2:39 PM)
- Text: Ok, let's just do the iCloud thing again then. >:)

 telegram.org

Messenger, End-to-End-Encryption bei privaten chats, Gruppen

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

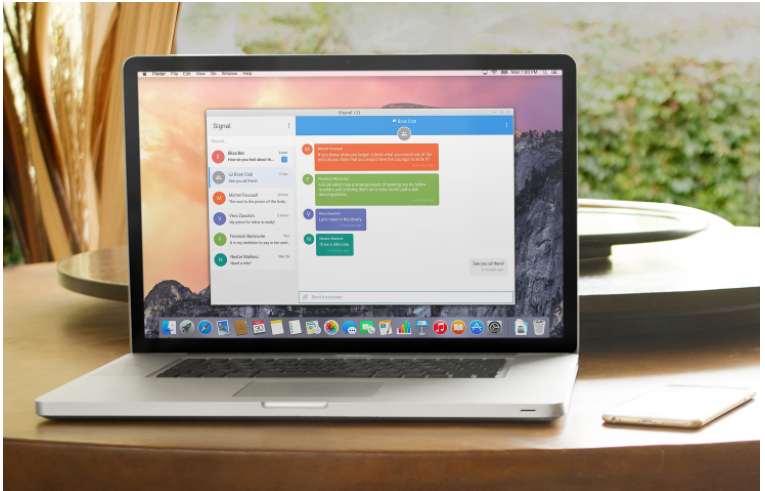
Dokumentation


Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



 [whispersystems.org](https://whispersystems.org)

Messenger, End-to-End-Encryption, Desktop-Version, Gruppen



Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

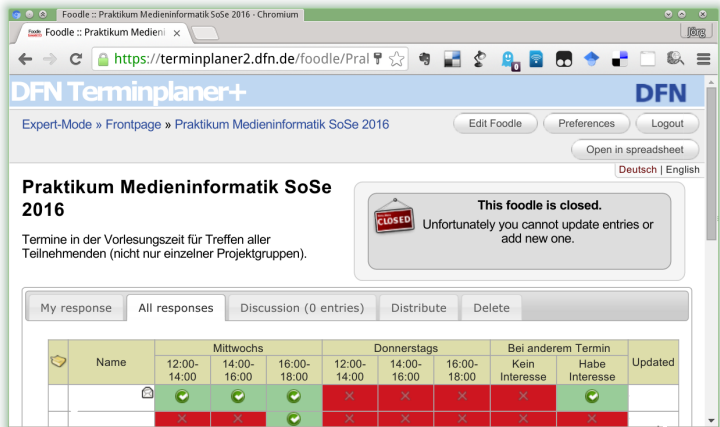
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



DFN Terminplaner+ DFN

Expert-Mode » Frontpage » Praktikum Medieninformatik SoSe 2016 Edit Foodle Preferences Logout

Open in spreadsheet Deutsch | English

## Praktikum Medieninformatik SoSe 2016

Termine in der Vorlesungszeit für Treffen aller Teilnehmenden (nicht nur einzelner Projektgruppen).

**This foodle is closed.**  
Unfortunately you cannot update entries or add new one.

My response | All responses | Discussion (0 entries) | Distribute | Delete

Name	Mittwochs			Donnerstags			Bei anderem Termin		Updated
	12:00-14:00	14:00-16:00	16:00-18:00	12:00-14:00	14:00-16:00	16:00-18:00	Kein Interesse	Habe Interesse	
	✓	✓	✓	✗	✗	✗	✗	✓	
	✗	✗	✓	✗	✗	✗	✗	✗	

🖱️ [terminplaner2.dfn.de](https://terminplaner2.dfn.de)

Terminplaner, DFN

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

**Automatisierung**

Empfehlung

Schluß

## Automatisierung

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- Git stellt sogenannte Hooks zur Verfügung, mit denen Aktionen angestoßen werden können
- Diese sind teilweise bei den genannten Providern vorhanden
  - Mail bei Commit
  - Slack-Messages bei Commit

# if this then that

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

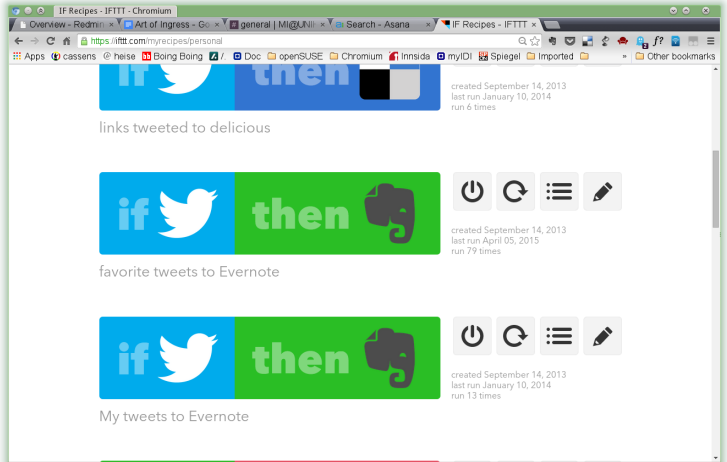
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß





IF Recipes - IFTTT - Chromium

Overview - Redm... x Art of Ingress - Go... x general | Mi@UNI... x Search - Asana x IF Recipes - IFTTT x



https://ifttt.com/myrecipes/personal

Apps cassens @ heise Boing Boing / Doc openSUSE Chromium Innsida myIDI Spiegel Imported Other bookmarks

**if**  **then** 



created September 14, 2013  
last run January 10, 2014  
run 6 times

links tweeted to delicious

**if**  **then** 

created September 14, 2013  
last run April 05, 2015  
run 79 times

favorite tweets to Evernote

**if**  **then** 

created September 14, 2013  
last run January 10, 2014  
run 13 times

My tweets to Evernote

 ifttt.com

If this then that, Kommerziell, Freemium

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

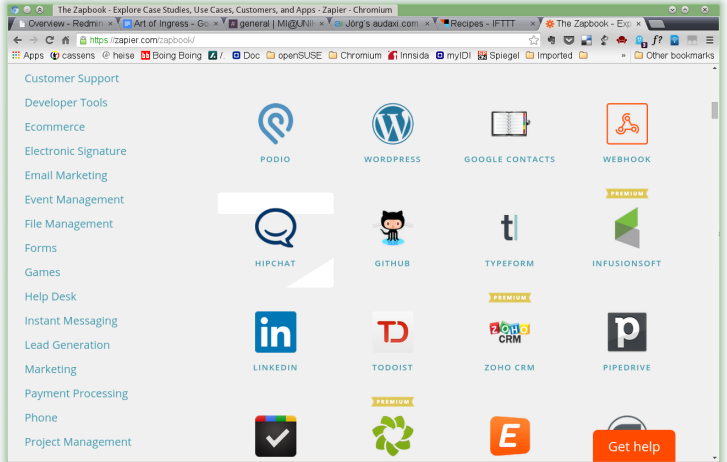
Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß



 zapier.com

Ähnliches Konzept wie ifttt, Kommerziell, Freemium

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

Empfehlung

Schluß

- Ermöglichen es, unterschiedliche Datenquellen und Datensenzen miteinander zu verknüpfen
- ifttt offener in der Nutzung, aber zapier hat manchmal bessere Anbindungen
- Vorteile
  - Automatisierung
- Nachteile
  - Man gibt einem externen Dienstleister Zugriff auf viele unterschiedliche Systeme

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

**Empfehlung**

Schluß

## Empfehlung

# Empfehlung

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Ticketing und  
Projektplanung

Dokumentation

Kommunikation &  
Koordination

Automatisierung

**Empfehlung**

Schluß

- Source-Code Management (git)
- Dokumentation von Anfang an (wiki,  $\LaTeX$ )
- Tickets
- Meilensteine
- Chat (slack, Telegram)



Vorab







git: Theorie

git: Praxis




git: Tools

Projekt



Schluß

- Basis
  -  [git-scm.com](https://git-scm.com) – Git für Windows (Installieren und deutsche Sprachdatei in .old umbenennen)
- Tutorial & Dokumentation
  -  [try.github.com](https://try.github.com) – Github-Tutorial mit der Octocat
  -  [git-scm.com/book/de](https://git-scm.com/book/de) – Deutsches Git-Buch
- GUI-Werkzeuge
  -  [www.syntevo.com/smartgit](http://www.syntevo.com/smartgit)
  -  [SourceTreeApp.com](http://SourceTreeApp.com)
  -  [gitkraken.com](http://gitkraken.com)

## ■ Hosted

-  [uni-hildesheim.de/gitlab](https://uni-hildesheim.de/gitlab) – Universität Hildesheim
-  [www.gitlab.com](https://www.gitlab.com)  
kostenlose öffentliche und private Projekte
-  [www.bitbucket.com](https://www.bitbucket.com)  
kostenlose öffentliche und private Projekte, kostenlos nur eingeschränkte Teamgrößen
-  [www.github.com](https://www.github.com)  
kostenlos nur für öffentliche Projekte

## ■ Self-Hosted

-  [about.gitlab.com](https://about.gitlab.com) – self-hosted
-  [gogs.io](https://gogs.io) – self-hosted

Vorab

git: Theorie

git: Praxis

git: Tools









Projekt

Schluß

## ■ Projektmanagement

-  [www.redmine.org](http://www.redmine.org)
-  [trac.edgewall.org](http://trac.edgewall.org)
-  [asana.com](http://asana.com)
-  [trello.com](http://trello.com)
-  [www.producteev.com](http://www.producteev.com)

## ■ Dokumentation

- $\text{\LaTeX}$  & git
-  [sharelatex.com](http://sharelatex.com)
-  [overleaf.com](http://overleaf.com)
-  [etherpad.org](http://etherpad.org)
-  [epad.hosting.uni-hildesheim.de](http://epad.hosting.uni-hildesheim.de)
-  [docs.google.com](http://docs.google.com)
-  [office.live.com](http://office.live.com)
-  [johnmacfarlane.net/pandoc](http://johnmacfarlane.net/pandoc)
-  [fletcherpenney.net/multimarkdown](http://fletcherpenney.net/multimarkdown)

Vorab

git: Theorie

git: Praxis

git: Tools

Projekt

Schluß

- Kommunikation & Koordination
  -  [slack.com](https://slack.com) – Slack
  -  [mattermost.org](https://mattermost.org) – Slack-Clone
  -  [telegram.org](https://telegram.org) – telegram
  -  [whispersystems.org](https://whispersystems.org) – Signal
  -  [terminplaner2.dfn.de](https://terminplaner2.dfn.de) – Foodle
- Automatisierung
  -  [ifttt.com](https://ifttt.com)
  -  [zapier.com](https://zapier.com)

# Werkzeuge

## Git und (andere) Kooperationswerkzeuge

Jörg Cassens, Jens Rademacher, Bastian Stender

### Medieninformatik Praktikum SoSe 2017



medieninformatik

IMAI – Institut für  
Mathematik und  
Angewandte Informatik