

Software Projekt II

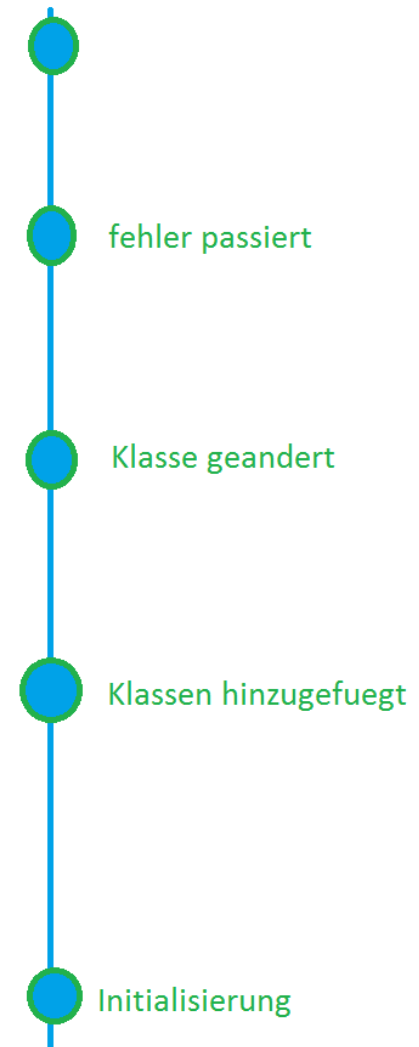
Seminar: Git

Gliederung

- Versionsverwaltung
- Git: Informationen
- Arbeitsvorgehen
- Git Befehle
- „Bedienungsanleitung“
- Praxis

Versionsverwaltung

- Vereinfacht zurueckverfolgen von Aenderungen
- Erleichtert „eliminieren“ von fehlern
- Vereinfacht das Arbeiten in einer Gruppe



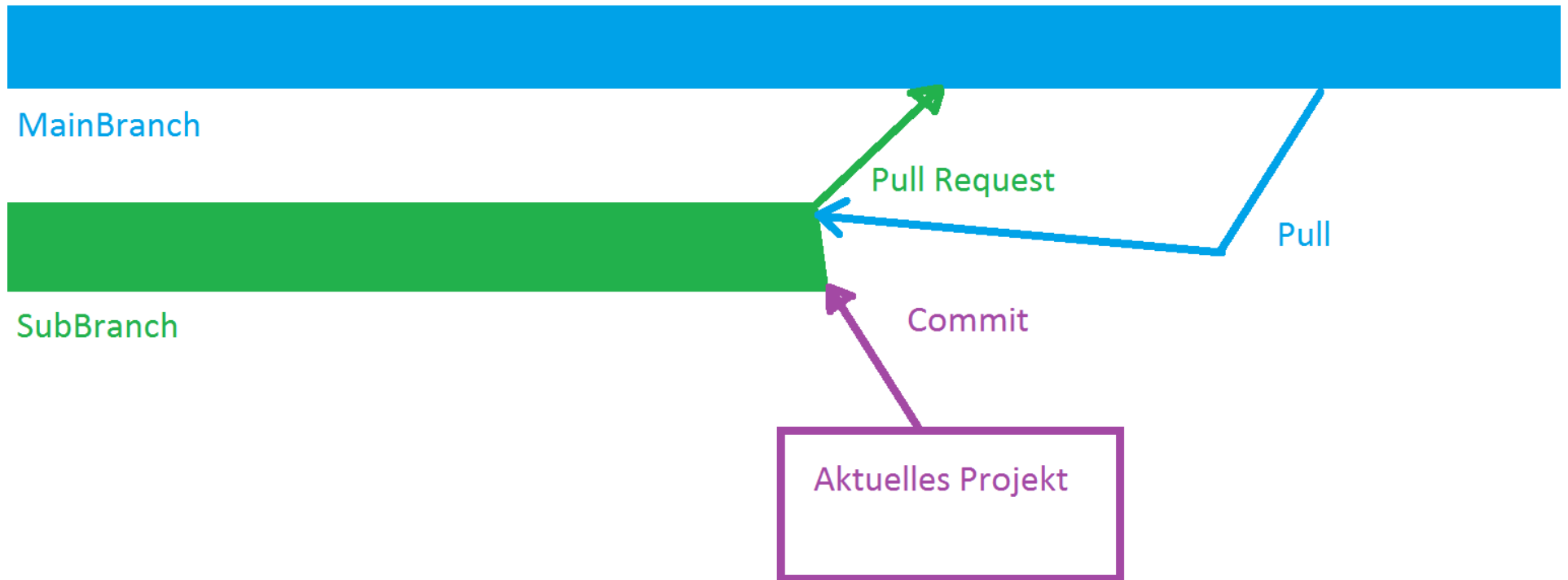
Git: Informationen

- Synergie mit Github (und Unity)
- Jeder Nutzer hat eine Kopie des Repositories
- Keine „Lock“-Funktion
- Effizient / Schnell

Arbeitsvorgehen

- Clone das Repository
- Erzeuge/Nutze einen Subbranch
- Committe Änderungen an den Subbranch
- Wenn Subbranch „bereit“, PullRequest senden

Arbeitsvorgehen



Git Befehle

- `git clone urlOfRepository`
- Zeigt Klont ein Repository auf die verwendete Maschine
- Bsp: `git clone https://github.com/HftStuttgartSP2/monstergame.git`

Git Befehle

- `git checkout -b nameOfABranch`
- Erstellt einen Branch und Checkt ihn aus
- Bsp: `git checkout -b featureA`

- Seperate Befehle:
- `git branch nameOfABranch`
- `git checkout nameOfTheBranch`

Git Befehle

- `Git commit -m „commit summary“`
- Committed aenderung an den aktuellen Branch
- Bsp: `git commit -m „issue 3“`

Git Befehle

Weitere Befehle:

- `git remote -v` → zeigt stream
- `git remote add upstream urlOfOrigin`
- `git fetch upstream` → aktualisieren durch upstream

„Bedienungsanleitung“

- Nutzung von Github (Desktop Applikation)

„Bedienungsanleitung“

- „Optional“ - gitignore datei definieren

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name



EmulatedUser ▾

/

Great repository names are short and memorable. Need inspiration? How about [automatic-waddle](#).

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **Unity** ▾

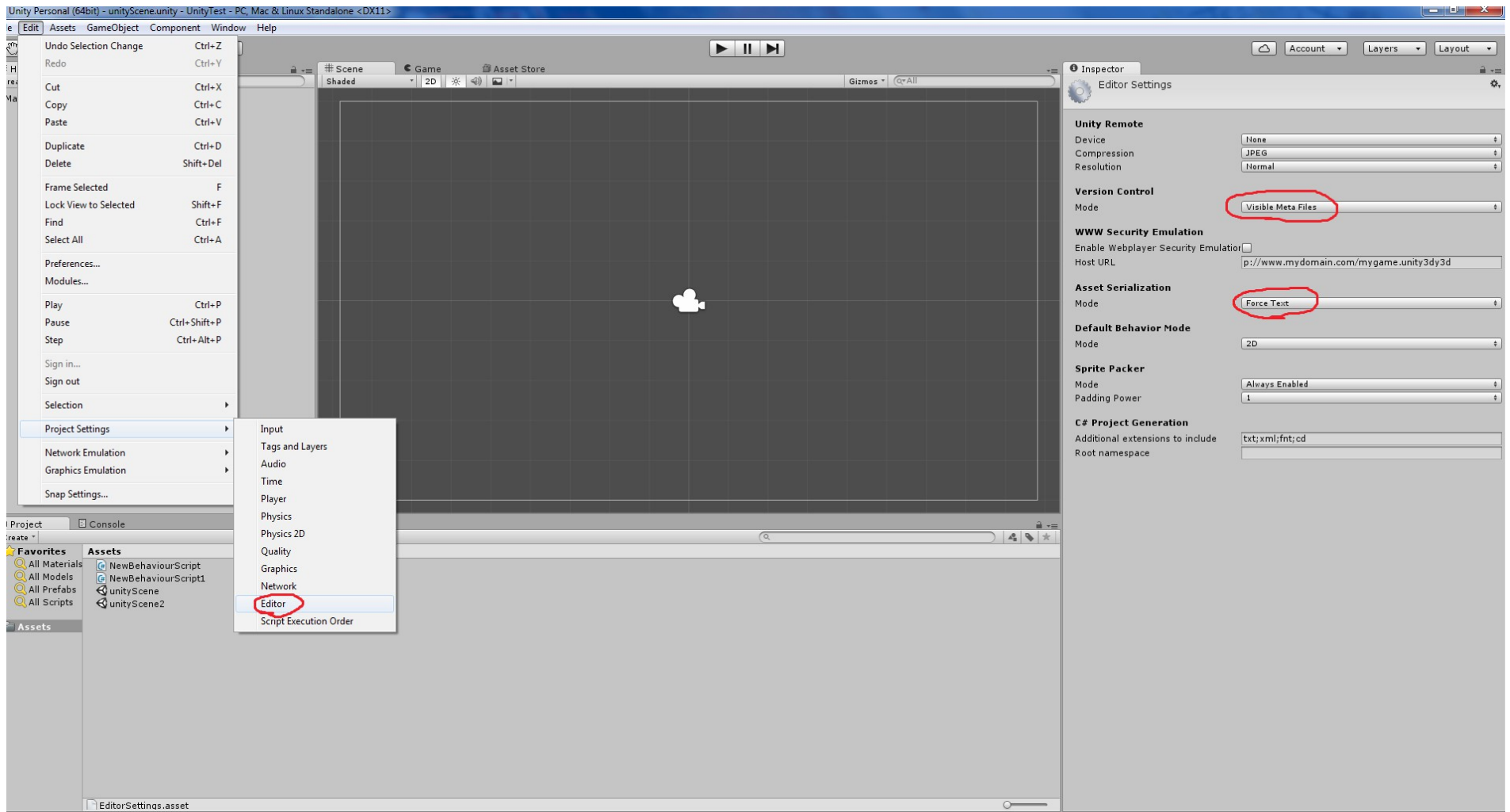
Add a license: **None** ▾



Create repository

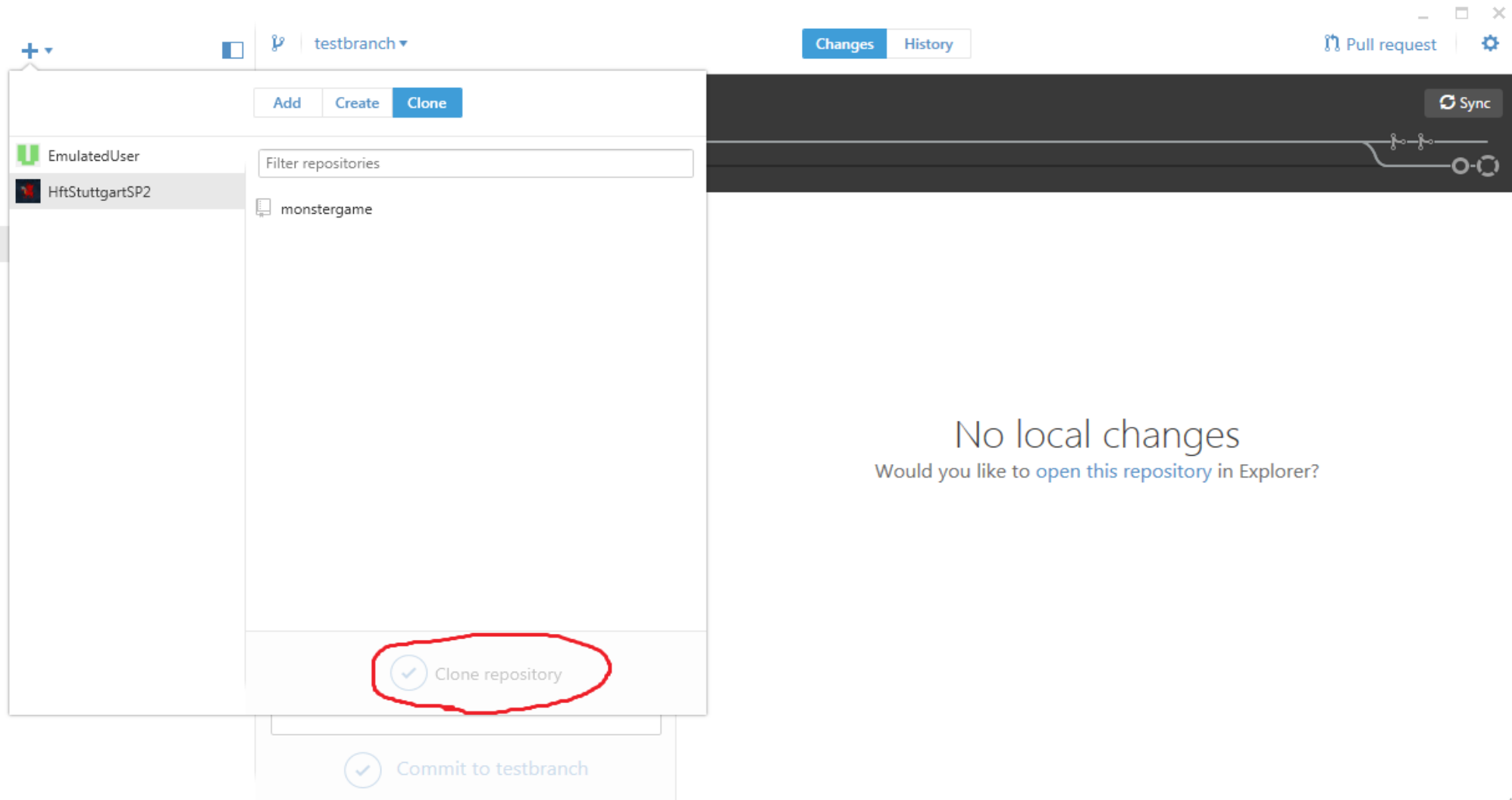
„Bedienungsanleitung“

- Unity Vorbereiten



„Bedienungsanleitung“

- Repository Klonen



„Bedienungsanleitung“

- Branch Selektieren / Erstellen

The screenshot displays the Unity Hub interface. On the left, a sidebar lists repositories under 'GitHub' and 'Other'. The 'UnityTest' repository is selected. The main area shows a comparison between the 'master' and 'testbranch' branches. A red circle highlights the 'testbranch' dropdown menu. Below the comparison, a list of 6 changes is shown, each with a checkbox and a file path. At the bottom, there is a 'Commit to testbranch' button. On the right, a 'Pull request' panel is visible, showing the 'Testbranch' and a 'Send pull request' button.

testbranch

Changes History

Pull request

from testbranch into master

Testbranch

Description...

There aren't any commits to merge.

Send pull request

Your local commits will be synced to GitHub.

Summary

Description

Commit to testbranch

„Bedienungsanleitung“

- Änderungen Commiten

The screenshot displays the GitHub web interface for a repository. On the left sidebar, under the 'GitHub' section, the repository 'UnityTest' is selected. The main area shows the 'testbranch' view with buttons for 'Update from master' and 'View branch'. A 'Changes' tab is active, displaying a list of 12 changes. The changes list includes files like 'UnityTest\Assets\NewBehaviourScript1.cs' and 'UnityTest\Library\ScriptAss...Assembly-CSharp.dll'. At the bottom, there is a 'Commit to testbranch' button, which is circled in red. Above this button are input fields for 'Summary' and 'Description'.

Filter repositories

GitHub

- monstergame
- RP6-Ente
- UnityTest

Other

- Tutorial

testbranch

Changes History

Update from master View branch

master testbranch

Sync

12 changes

- UnityTest\Assets\NewBehaviourScript1.cs
- UnityTest\Assets\NewBehaviourScript1.cs.meta
- UnityTest\Library\assetDatabase3
- UnityTest\Library\CurrentLayout.dwl
- UnityTes...\00000000000000004000000000000000
- UnityTes...\00000000000000006100000000000000
- UnityTest...\253602eae561e2f43a031b9ea7ba5ecd
- Unity...\253602eae561e2f43a031b9ea7ba5ecd.info
- UnityTe...\be462ea25b4585044bb22de8b778cddd
- UnityTest\Library\ScriptAss...\Assembly-CSharp.dll

Summary

Description

Commit to testbranch

„Bedienungsanleitung“

- Pull Request Initiieren

The screenshot displays the Visual Studio interface for initiating a pull request. On the left sidebar, under the 'GitHub' section, the repository 'UnityTest' is selected. The main workspace shows a commit history view with 'master' and 'testbranch' branches. The 'testbranch' is highlighted, and it shows '0 changes'. A message in the center states 'No local changes' and asks 'Would you like to open this repository in Explorer?'. On the right, the 'Pull request' pane is open, showing a 'from testbranch into master' dropdown. Below this, there are fields for 'Summary' and 'Description'. A green checkmark icon indicates 'This pull request can be automatically merged.' and a 'Send pull request' button is visible. The bottom status bar shows 'Created commit: 'Summary''.

Filter repositories

GitHub

- monstergame
- RP6-Ente
- UnityTest

Other

- Tutorial

testbranch

Changes History

master

testbranch

0 changes

No changes

No local changes

Would you like to [open this repository](#) in Explorer?

Summary

Description

[Pull request](#)

from testbranch into master

Summary

Description

This pull request can be automatically merged.

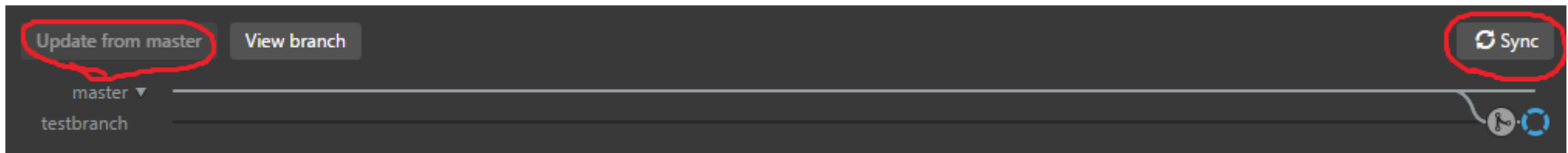
[Send pull request](#)

Your local commits will be synced to GitHub.

Undo Created commit: 'Summary'

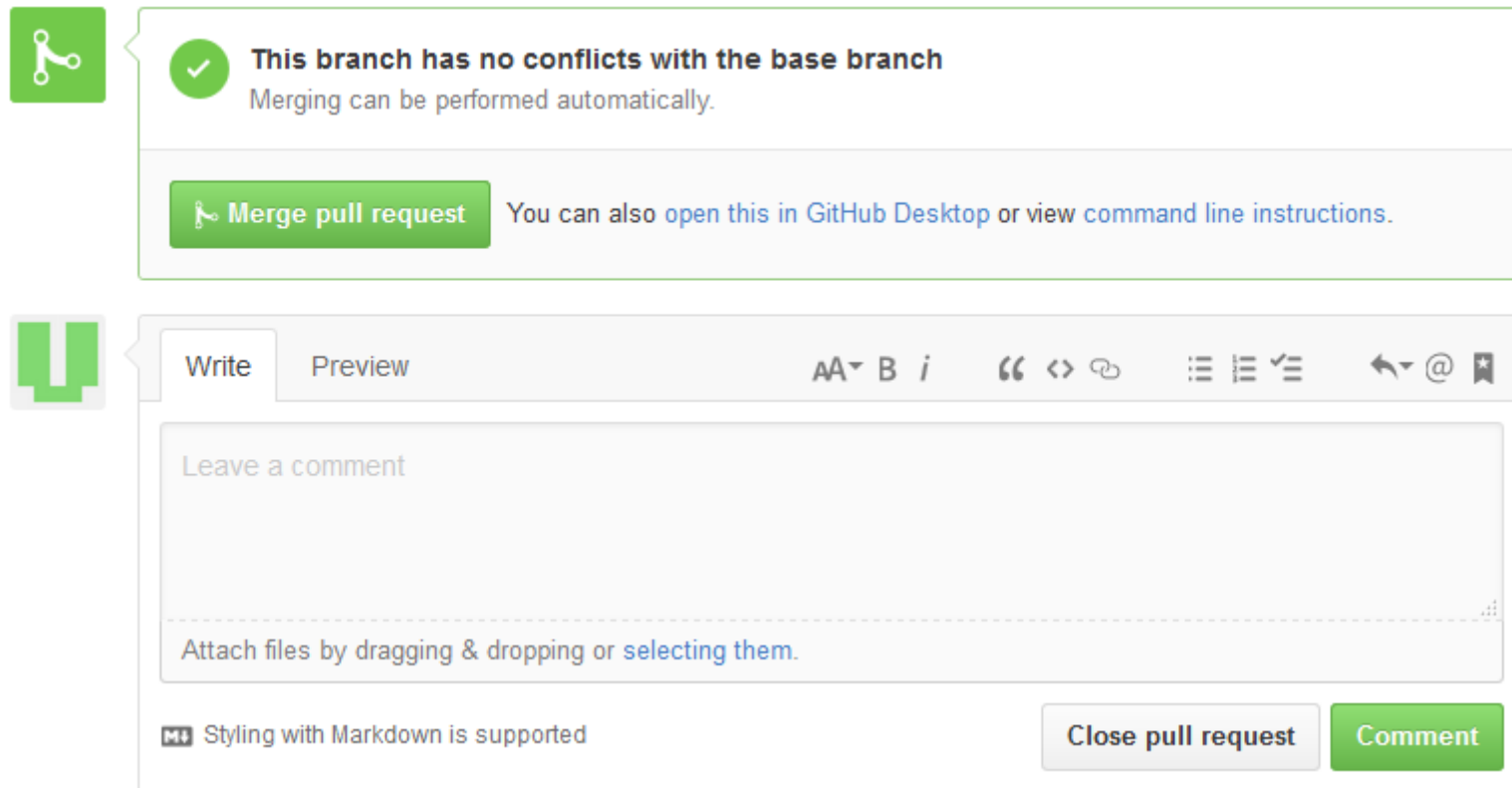
„Bedienungsanleitung“

- Aktualisierung/ Synchronisierung v. Branches






„Bedienungsanleitung“


- Pull Request Akzeptieren



The image shows a GitHub Pull Request interface. At the top, a green box with a checkmark icon indicates that the branch has no conflicts with the base branch and that merging can be performed automatically. Below this, a green button labeled "Merge pull request" is visible, followed by a link to "open this in GitHub Desktop" and a link to "view command line instructions". Below the merge section, there is a comment section with a "Write" tab and a "Preview" tab. The "Write" tab is active, showing a text area with the placeholder "Leave a comment" and a link to "Attach files by dragging & dropping or selecting them." At the bottom of the comment section, there is a note that "Styling with Markdown is supported" and two buttons: "Close pull request" and "Comment".


  **This branch has no conflicts with the base branch**
Merging can be performed automatically.

 **Merge pull request** You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

 **Write** **Preview** **AA** **B** **i** **“** **<>** **↻** **≡** **≡** **✓** **≡** **↶** **@** **★**

Leave a comment

Attach files by dragging & dropping or [selecting them](#).

 Styling with Markdown is supported

Close pull request **Comment**

Praxis