

Softwaretechnik I

Übung 3

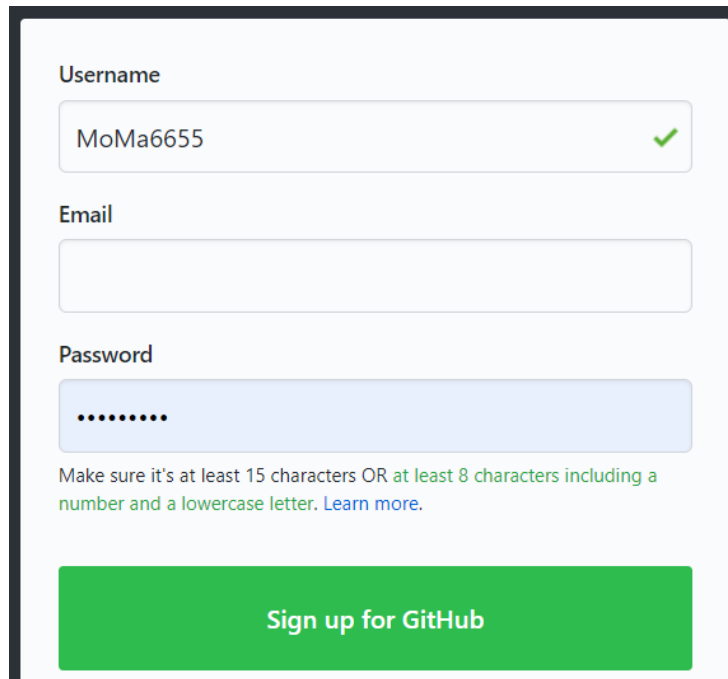
Martius Monika & Sivarasa Litharshi

Für die Aufgabe wurde ein grafischer Git-Client verwendet.

Aufgabe 1)

Eröffnen eines neuen GitHub-Kontos:

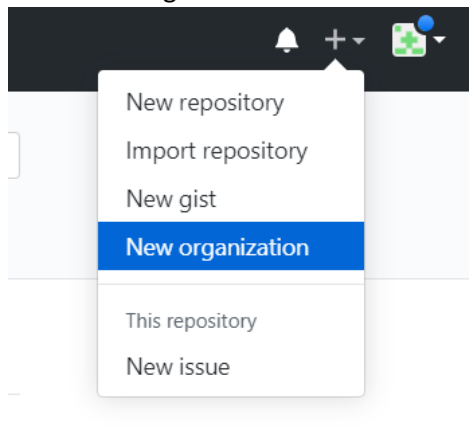
1. Auf <https://github.com/> gehen und ein Konto durch die Eingabe eines Username / Email und Password erstellen.

A screenshot of the GitHub sign-up form. It features three input fields: 'Username' with the value 'MoMa6655' and a green checkmark, 'Email' which is empty, and 'Password' which is masked with dots. Below the password field is a text requirement: 'Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. Learn more.' At the bottom is a large green button labeled 'Sign up for GitHub'.

2. In der angegebenen E-Mail-Adresse die Anmeldung bestätigen.
3. Auf <https://github.com/login> mit Username oder E-Mail-Adresse einloggen.

Erstellen einer Organisation für Teambearbeitung:

Eine neue Organisation erstellen durch hinzufügen „New organization“



1. Kostenlose Version auswählen für die Teambearbeitung, dann Name für das Team vergeben und Account Auswählen und bestätigen.

A screenshot of the GitHub 'Create team' form. On the left, there is a card for 'Team for Open Source' with a '\$0 USD' price tag. Below the card, a list of features is shown with checkmarks: 'Unlimited public repositories', 'Unlimited private repositories', 'GitHub Actions', 'GitHub Packages', 'Team access controls', and 'User management'. On the right, the form fields are: 'Organization account name' (with 'NameTeam' entered and a warning that the name is taken), 'Contact email' (with 'monikamartius@gmail.com' entered), and 'This organization belongs to' (with 'My personal account' selected). At the bottom right, there is a blue 'Next' button.

Team for Open Source
\$0 USD

Collaboration tools for teams who don't need private repositories

Choose Team for Open Source

- ✓ Unlimited public repositories
- ✗ Unlimited private repositories
- ✓ GitHub Actions
See details
- ✓ GitHub Packages
See details
- ✓ Team access controls
- ✓ User management

Organization account name *

NameTeam

The name 'NameTeam' is already taken.

Contact email *

monikamartius@gmail.com

This organization belongs to: *

☒ My personal account
I.e., MoMa666

☐ A business or institution
For example: GitHub, Inc., Example Institute, American Red Cross

Next

Team erstellen.

Paket(free) auswählen

2. Teammitglied hinzufügen: (Eingabe des GitHub Usernames des Mitgliedes oder der E-Mail. Das Teammitglied muss sich auch einen kostenlosen Account bei Github erstellt haben.)

Add organization members

Organization members will be able to view repositories, organize into teams, review code, and tag other members using @mentions.

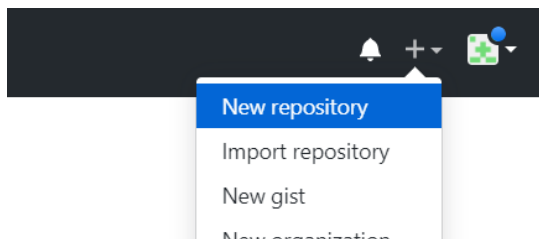
[Learn more about permissions for organizations →](#)

Search by username, full name or email address

Complete setup

Erstellen eines Repository:

1. Rechts oben neben dem Avatar oder Identicon auf + und dann Neues Repository auswählen.



2. Bei Owner die Organisation auswählen.
3. Benennen des Repository in hello-world
4. Eine kurze Beschreibung hinzufügen.
5. Wählen -> Dieses Repository mit einer README-Datei initialisieren.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository? [Import a repository.](#)

Owner

Repository name *

SWT-Aufgabe3Martius-Sivarasa

/ hello-worlds

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

Description (optional)

SWT1 Aufgabe3

☒ Public

☐ Private

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

Your current plan does not support private repositories. [Upgrade to Team](#)

Skip this step if you're importing an existing repository.

☒ Initialize this repository with a README

This will let you immediately clone the repository to your computer.

Rechte fürs Team einstellen:

1. Nachdem das Repository erstellt wurde auf der Übersicht auf People klicken und das Zahnradsymbol des betroffenen Mitgliedes auswählen um die Rechte zu ändern.

SWT-Aufgabe3Martius-Sivarasa

Repositories 1

Packages

People 2

Teams

Projects

Settings

Members
Outside collaborators

Invite member

Team members
2FA
Role

Isivarasa

2FA
Private
Member
0 teams

MoMa666

2FA
Private
Owner
0 teams

People 2
Teams
Projects
Settings

Isivarasa has access to 1 repository

SWT-Aufgabe3Martius-Sivarasa/hello-world

Write on this repository

Manage access

Bei Manage access die Rechte festlegen.

SWT-Aufgabe3Martius-Sivarasa/hello-world

Isivarasa has **write** access to this repository

Read
Inactive

Nothing explicitly gives Isivarasa read access to SWT-Aufgabe3Martius-Sivarasa/hello-world.

Write
Active

Membership privilege
Edit

Remove write access

Von Read auf Write stellen.

Aufgabe 2)

Jedes Team-Mitglied kann entweder in Git-Hub die Datei erstellen [Create new file](#) oder von seinem Pc aus die Dateien anlegen und hochladen über [Upload files](#).
Erstellte Dateien.

Branch: master ▼

Commits on Jan 11, 2020

- Create Sivarasa.pdf ...
Isivarasa committed 11 seconds ago
- CreateMartius.pdf
MoMa666 committed 1 minute ago
- Create Sivarasa.txt ...
Isivarasa committed 1 minute ago
- Create Martius.txt
MoMa666 committed 1 minute ago
- Initial commit
MoMa666 committed 1 hour ago

SWT1 Aufgabe3 [Edit](#)

[Manage topics](#)

10 commits 3 branches 0 packages 0 releases 2 contributors

Branch: master ▼ [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download ▼](#)

MoMa666 Merge pull request #2 from SWT-Aufgabe3Martius-Sivarasa/MoMa666-patch-1 ... Latest commit 051a5be 19 hours ago

Martius.pdf	CreateMartius.pdf	20 hours ago
Martius.txt	Create Martius.txt	20 hours ago
README.md	Initial commit	21 hours ago
Sivarasa.pdf	Create Sivarasa.pdf	20 hours ago
Sivarasa.txt	Create Sivarasa.txt	20 hours ago

Aufgabe 3)

Nachdem die Team.txt Datei in Github erstellt wurde, wurde diese zeitgleich von den Teammitgliedern geöffnet und bearbeitet. Derjenige welcher zuerst committed hat erhält keine Nachricht, jedoch erhalten alle anderen eine Warnung:

MoMa666 has committed since you started editing. [See what changed](#)

SWT-Aufgabe3Martius-Sivarasa / [hello-world](#)

<> Code

! Issues 0

🔗 Pull requests 0

▶ Actions

📁 Projects

[hello-world](#) / team.txt

Cancel

<> Edit file

👁 Preview changes

```
1  Hallo Welt!  
2  Wie geht dir?  
3
```

Nun kann sich derjenige die Änderungen anzeigen lassen.

🔗 1 commit

📄 1 file changed



Commits on Jan 11, 2020



MoMa666

Update team.txt



Showing 1 changed file with 1 addition and 1 deletion.

▼ 2 team.txt

... @@ -1,2 +1,2 @@

1 1 Hallo Welt!

2 - Halloooo

2 + Wie geht dir?

Danach ist es ihm möglich diese mit einem Pull Request zu vergleichen oder einfach mit Merge die Datei mit seinem Stand zu überschreiben. Durch den Request wird dann eine neue branch erstellt und mit der aktuellen verglichen, hier kann man nun die Datei bearbeiten und die Dinge zb. Zeilenabsätze korrigieren, so dass beide Texte wie oben Halloooo und Wie geht es dir übernommen werden könne. Nach der Bestätigung des Request kann dieser im Anschluss gelöscht werden.

The screenshot shows a GitHub pull request interface. At the top, a comment by 'MoMa666' is visible. Below it, a commit update is shown with a green checkmark and the text 'Update'. A merge commit is displayed with the message 'MoMa666 merged commit 051a5be into master now'. A 'Revert' button is present. Below the merge, a purple box indicates 'Pull request successfully merged and closed' with a 'Delete branch' button. At the bottom, a comment editor is visible with a 'Write' button and a 'Preview' tab.

Der Pull Request wäre dann auch hier die angegebene Lösung für das Problem. Der erste Commit wird übernommen alle anderen werden über diesen informiert bei dem Versuch zu commiten.

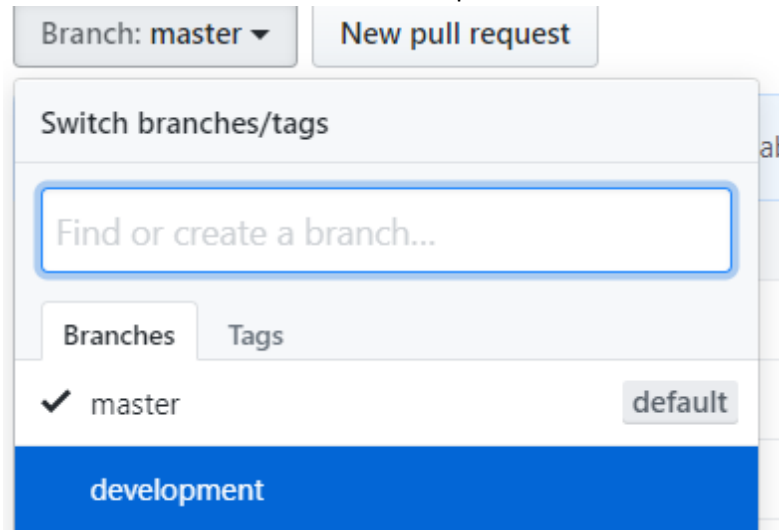
Aufgabe 4)

Erzeugen der Branch development:

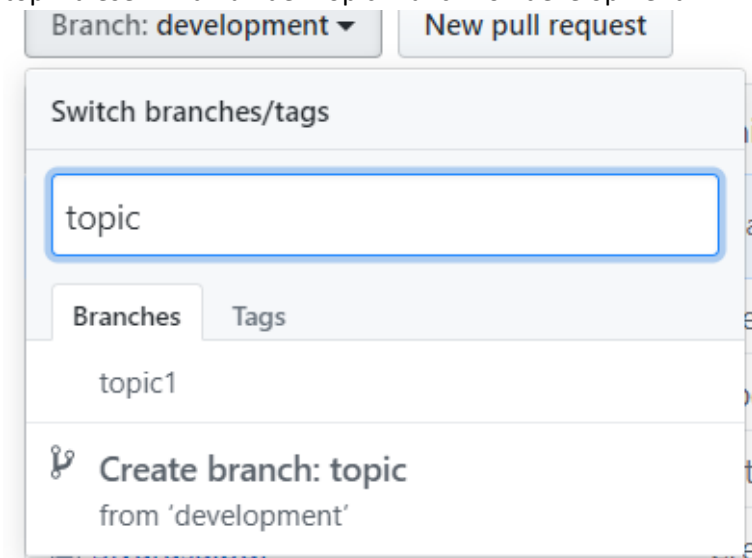
Unter Branch : master auf den Pfeil klicken und einen neuen Branch namens development erstellen.

The screenshot shows the GitHub 'Switch branches/tags' dropdown menu. The 'development' branch is selected and highlighted. Below the dropdown, the text 'Create branch: development from 'master'' is visible. The 'Branch: master' dropdown is also visible at the top of the menu.

Danach wählt man den Branch development aus.



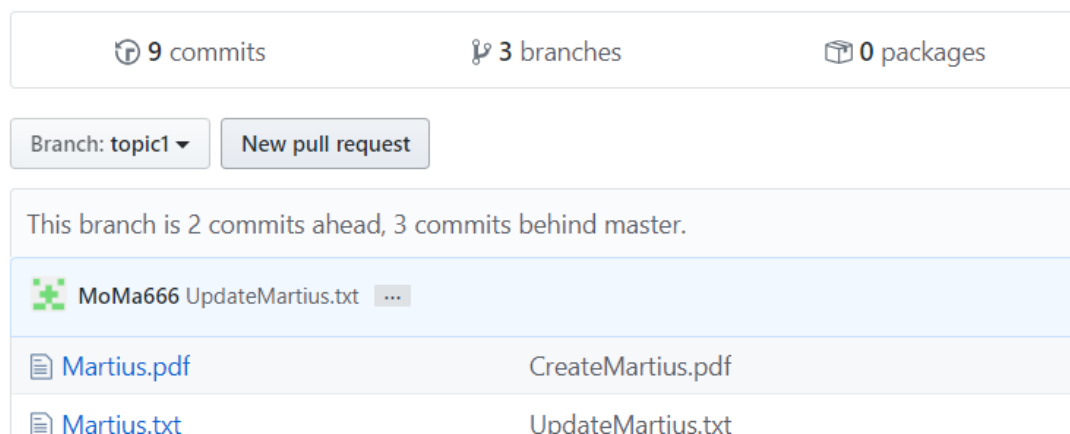
Nun geht man wie oben vor und klickt erneut auf den Pfeil und erstellt einen Branch mit den Namen topic1 dieser wird nun der Topic Branch von development.



Nun wurden die Text Dateien Martius.txt und Sivarasa.txt bearbeitet in Topic1. Hierfür wählt man in Branch Topic1 aus, bearbeitet dann dort die Dateien und wählt dann New pull request aus.

SWT1 Aufgabe3

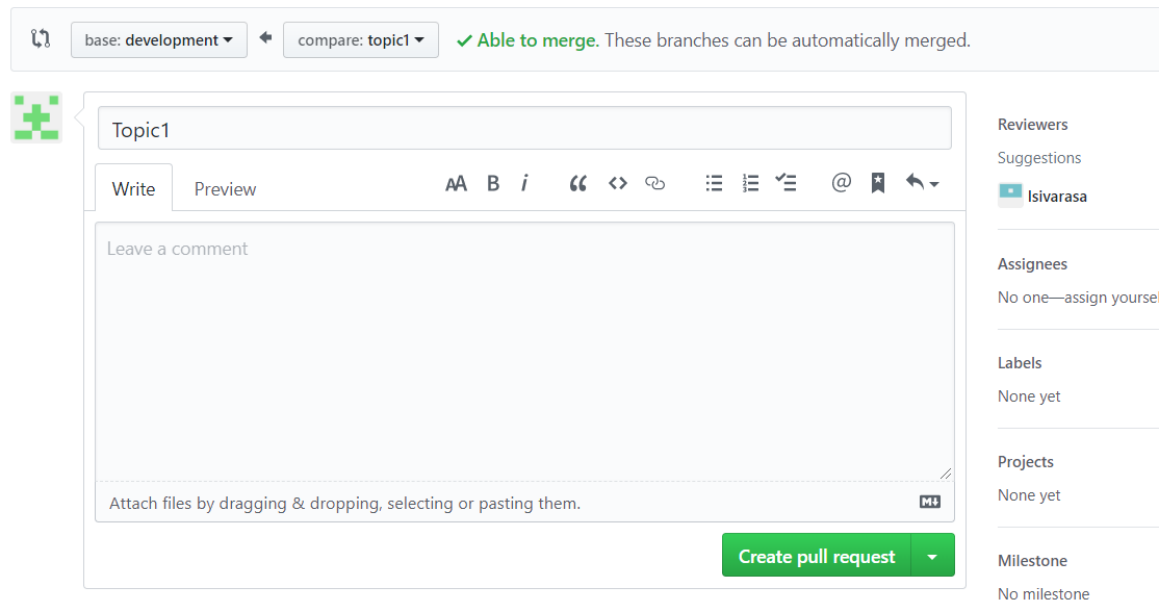
[Manage topics](#)



Im folgenden Fenster, welches sich nun öffnet, wählt man aus welcher Branche miteinander verglichen werden sollen, bzw. welcher Inhalt zusammengefügt wird.

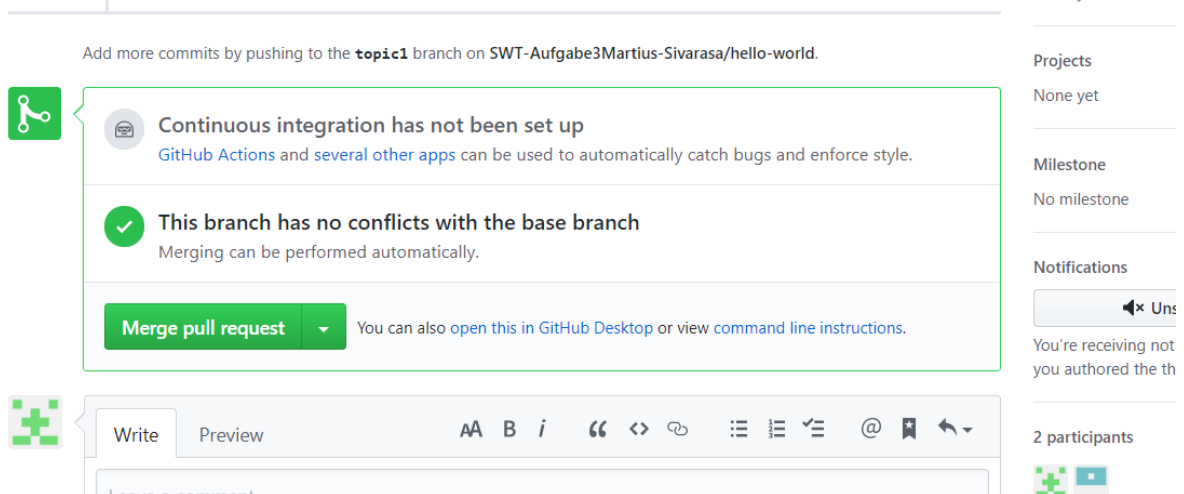
Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

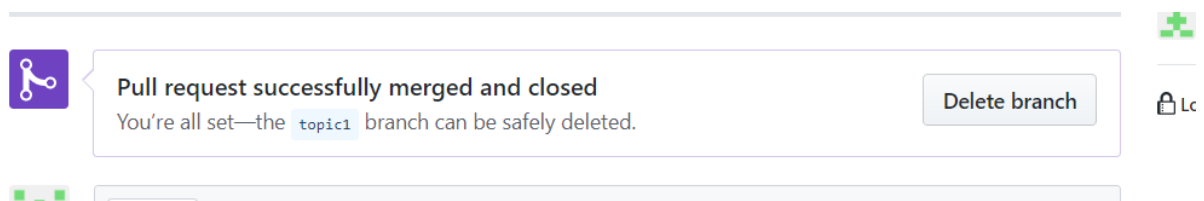


Hier wählt man als Base development aus und als compare topic1. Danach klickt man auf Create pull request.

Wenn nun keine Konflikte bestehen erhält man folgende Meldung.



Dann kann man durch Merge pull request den Merge abschließen.



Sollten Konflikte auftreten erhält man folgende Meldung:

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base: development

compare: topic1

✗ Can't automatically merge. Don't worry, you can still create the pull request.

UpdateAufgaben

Write Preview

AA B i “ <> @

Leave a comment

Reviewers

No reviews

Assignees

No one—assign yourself

Nun erstellt man wieder einen Pull Request und erhält folgende Meldung.

UpdateAufgaben

Verified a5c9ea7

Add more commits by pushing to the **topic1** branch on **SWT-Aufgabe3Martius-Sivarasa/hello-world**.

⚠

This branch has conflicts that must be resolved

Use the [web editor](#) or the [command line](#) to resolve conflicts.

Resolve conflicts

Conflicting files

team.txt

Merge pull request

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

Um die Konflikte zu bearbeiten/beheben klickt man nun auf Resovle confilcts und gelangt zu den entstandenen Konflikten:

Resolving conflicts between **topic1** and **development** and committing changes to **topic1**

1 conflicting file

team.txt

1 conflict Prev Next

Mark as resolved

team.txt

team.txt

1 <<<<<< topic1

2 Mein neue Liste.

3 Schlafen

4 Essen

5 Schlafen

6 =====

7 Wir haben heute Sonntag bitte denke daran die Fenster zu putzen.

8 Milch einkaufen.

9 Mit dem Hund raus gehen.

10 Boden wischen.

11 >>>>>> development

Nachdem man diese nun beseitigt hat, indem man die Bereiche von topic1 und development bearbeitet, erstellt man eine Lösung durch das Betätigen des Button Mark a resolved:

team.txt

1 conflict Prev Next

Mark as resolved

1 Wir haben heute Sonntag bitte denke daran die Fenster zu putzen.

2 Milch einkaufen.

3 Mit dem Hund raus gehen.

4 Boden wischen.

5 Schlafen

6 Essen

7 Schlafen


Danach wird der Commit merge verfügbar:


Commit merge


✓ Resolved

Dies wiederholt sich solange bis alle Konflikte in jeder Datei behoben wurde danach wird der gesamte Pull Request beendet, wie oben beschrieben.


Add more commits by pushing to the **topic1** branch on SWT-Aufgabe3Martius-Sivarasa/hello-world.



**Continuous integration has not been set up**
GitHub Actions and several other apps can be used to automatically catch bugs and enforce style.

**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




Pull request successfully merged and closed
You're all set—the **topic1** branch can be safely deleted.

Delete branch



2 participants



Notifications

🔊 Uns

You're receiving not
you authored the th

Projects


None yet

Milestone

No milestone

Aufgabe 5)

Durch die Funktion Download als ZIP ist es möglich eine Sicherungskopie auf den Desktop herunterzuladen, dies kann jedes Teammitglied tun.


 **Compare & pull request**

new file Upload files Find file

Clone or download ▼

Clone with HTTPS ? [Use SSH](#)







Use Git or checkout with SVN using the web URL.



[Open in Desktop](#) [Download ZIP](#)

Es ist auch möglich den ZIP Ordner direkt auf einen USB-Stick oder Externen Festplatte

abzuspeichern. In dem Ordner sind dann folgende Dateien enthalten:

3.Semester > Softwaretechnik 1 > Übungen > Übung3 > hello-world-master (1) > hello-world-master						"hello-world-master" durch	
<input type="checkbox"/>	Name	Typ	Komprimierte Größe	Kennwortg...	Größe	Verhältnis	
	Martius	PDF-Datei	1 KB	Nein	1 KB	0%	
	Martius	Textdokument	1 KB	Nein	1 KB	0%	
	README.md	MD-Datei	1 KB	Nein	1 KB	0%	
	Sivarasa	PDF-Datei	1 KB	Nein	1 KB	0%	
	Sivarasa	Textdokument	1 KB	Nein	1 KB	0%	
	team	Textdokument	1 KB	Nein	1 KB	7%	

Er enthält so zu sagen alle Dateien welche erstellt wurden von der letzten Änderung aus.