

GitHub Workshop

Userstory

- Einrichten eines öffentlichen GitHubs

... aber gemeint waren wohl eher ...

Userstories

- I. Einrichten eines 'eigenen' öffentlichen GitHub Accounts
- II. Nutzen des 'eigenen' öffentlichen GitHubs
- III. Einrichten des lokalen GitHub Desktops
- IV. Lokales Arbeiten mit online GitHub Daten

© 2017 Michael Pätzold

Tasks

1. Einrichten eines 'eigenen' öffentlichen GitHub Accounts
2. Nutzen des 'eigenen' öffentlichen GitHubs
3. Öffentliches Repository in 'eigenen' Account holen (wg. Schreibrechten)
4. Einrichten des lokalen GitHub Desktops
5. Starten GitHub Desktop und Grundeinstellungen prüfen
6. Lokales Arbeiten mit online GitHub Daten
7. Änderungen vom 'eigenen' online GitHub in 'anderen' online GitHub uploaden
8. Rollenwechsel: Merge (als User des 'anderen' online GitHub)
9. Rollenwechsel: Check der Übernahme (im 'eigenen' online GitHub)
10. Anderen Usern Mitarbeit im 'eigenen' Repository ermöglichen

Was wird benötigt?

- a) Windows PC mit Internet-Zugang
- b) Mail-Adresse, die auch sofort per Web-Mail genutzt werden kann
- c) ein öffentlicher GitHub (als Bsp. wird "DevGroupGoeKs" on-the-fly erzeugt)
- d) GitHubSetup.exe (wird on-the-fly heruntergeladen)
- e) Account- und Repository-Name des 'anderen' Projekts an dem man mitarbeiten möchte z. Bsp. "DevGroupGoeKs" oder "MyGoeKsTest"
- f) Der User des 'anderen' Accounts – oder sein GitHub-Zugang ;-)
(oder man führt den 1. Task zweimal durch und schafft sich einen zweiten öffentl. GitHub Account)

1. Einrichten eines öffentl. GitHub Accounts

(Motivation: notwendige Voraussetzung zur Nutzung von github.com)

Registrierung:

- <https://github.com/>
- Button 'Sign up for GitHub' (3 Steps: Set up ..., Choose ..., Tailor ...)
- Username: <z.B. DevGroupGoeKs>
- Mail: <z.B. project@xn--devgroupgks-zfb.de>
- Password: <mein_password>
- GitHub-URL : <https://github.com/DevGroupGoeKs>
- GitHub Reaktion: Mail mit Subject
[GitHub] Please verify your email address

2. Nutzen des 'eigenen' öffentlichen GitHubs

(Motivation: Kennenlernen der github.com Grundbegriffe)

Nutzung des Accounts

- Button 'Sign in'
- User-Profile im Pulldown-Menü (ganz rechts)
Bsp.: 'Your profile' : Avatar -> Your profile -> Change your avatar

Erste Schritte nach der Anmeldung:

- GitHub-Button (ganz links) -> Button 'Read the guide'
 - Create a Repository
 - Create a Branch (<Branch name>)
 - Make and commit changes:
 - File anklicken, Pencil-Symbol, ... (Html-Page ganz unten) -> Commit changes
 - Open a Pull Request
 - Compare and Pull Request
 - Merge your Pull Request
 - Delete branch

3. Öffentl. Repository in 'eigenen' Account holen

(Motivation: Nur im eigenen Account hat man Schreibrechte.
Bei Mitarbeit in Repositorys 'anderer' Accounts hat man keine Schreibrechte.
Ausnahme: Man wird vom Owner des Repositorys 'invited'.)

- Auf 'eigenen' öffentlichen GitHub Account einloggen
- GitHub-Button (d. h. zur GitHub-Start-Seite <https://github.com/>)
- Search <GitHub-Account-Name>
- Aus Trefferstatistik Users auswählen -> 'Users' anklicken
- gewünschtes Repository des Users anklicken

WICHTIG:

- Fork-Button 'Fork' (rechts oben) ... und nicht 'Clone or download'
- Checken: Wo bin ich gelandet!
(im 'eigenen' Account, quasi in einem neuen 'Branch'; den ich bearbeiten darf.
Für Änderungen, die ins Ursprungs-Repository einfließen sollen, muss dann 'Pull request' verwendet werden.)

4. Einrichten des lokalen GitHub Desktops

(Motivation: Mit einem Repository in eigener Umgebung arbeiten.)

- Google Suche: „github for windows 10 download“
 - Treffer: <https://desktop.github.com/>
- Button 'Download GitHub Desktop' (GitHubSetup.exe)
- Starten im Download-Verz.: GitHubSetup.exe
- BTW: Wo steht dann die GitHub.exe?
 - Z. Bsp. bei mir in:
C:\Users\<username>\AppData\Local\Apps\2.0\1HXGP1D4.5RM\J1OLTAMN.G
5K\github.exe_317444273a93ac29_0003.0003_none_a7f3b5f9c37dfee8

5. GitHub Desktop Grundeinstellungen prüfen

(Motivation: Um ein online GitHub Repository mit einem lokalen Repository zu 'Sync'en, müssen der lokale GitHub Desktop und der online GitHub über den gleichen GitHub-Account miteinander verbunden sein.)

- Checken des GitHub Logins in GitHub Desktop:
 - Tools and Options -> Options -> Add account
 - Tools and Options -> Options -> Configure git (<Name>, <Mail address>)
- Checken der Version des GitHub Desktop (wg. Software Update):
 - Tools and Options -> 'About GitHub Desktop'

6. Lokales Arbeiten mit online GitHub Daten I

(Motivation: Datenaustausch zwischen lokalem und online Repository.)

- 'eigenen' online GitHub öffnen
- GitHub Desktop öffnen
- Aufgabe: Clonen eines Repositorys aus 'eigenen' Account in GitHub Desktop
- Button 'Clone or download' (Zielpfad angeben)
- das neue Repository erscheint in GitHub Desktop

Hinweis zu 'Clonen' und 'Sync':

- Clonen ist nur notwendig, wenn das Repository lokal nicht existiert.
- Wenn es existiert, wird zum *Synchronisieren* lokal der Button 'Sync' verwendet!

6. Lokales Arbeiten mit online GitHub Daten II

(Motivation: Datenaustausch zwischen lokalem und online Repository.)

- Änderung im Repository (schnelles Bsp. für Testzwecke)
 - Rechte Maustaste auf dem Repository-Name -> 'Open in Explorer'
 - File auswählen, ändern, speichern, close
 - Checken: "Change" wird angezeigt
 - 'Commit to Master'
- Synchronisieren mit Button 'Sync' (rechts oben)
- Lokale Änderungen sind damit in den 'eigenen' öffentlichen GitHub übernommen

7. Vom 'eigenen' in 'anderen' GitHub uploaden

(Motivation: Aktive Mitarbeit an 'anderen' Repositorys, für die man keine Schreibrechte hat.)

- Button 'New pull request' zeigt Comparing changes
- Button 'Create pull request'
 - Kommentar prüfen/ergänzen
 - Button 'Create pull request'
(Damit ist der User des 'anderen' online GitHub informiert)

8. Rollenwechsel: Merge

(Motivation: Als User eines online GitHub die Mitarbeit eines 'anderen' online GitHub-User übernehmen.)

- Karteireiter 'Pull requests (n)' zeigt Änderungsanträge für das eigene Repository an
- Pull requests nach *Kommentar-Texten* auswählen
- Button 'Merge pull request'
- Button 'Confirm Merge'
- Check der Übernahme mit Karteireite '<> Code'

9. Rollenwechsel: Check der Übernahme

(Motivation: Verifizieren, wie mit der Änderungsbereitstellung seitens des Owners verfahren wurde.)

- Notifications beachten

10. Anderen Usern Mitarbeit ermöglichen

(Motivation: Anderen GitHub-Usern Schreibrechte für mein Repository erteilen, damit sie es wie ihr eigenes Repository benutzen können.)

- Das 'eigene' Repository anwählen
- Den Karteireiter 'Settings' aktivieren
- In der linken Auswahlliste die 'Collaborators' anklicken
- Den gewünschten *Collaborator* im Single-Line-Edit unter der Collaborators-Liste eintragen
- Button 'Add collaborator'
- Der Status des eingeladenen *Collaborators* wird in der Collaborators-Liste angezeigt

GitHub – Ende des Workshops

Material zu diesem Workshop unter

<http://www.devgroupgöks.de/>

Vielen Dank für Eure Mitarbeit!