

# Automatische Aufgabengenerierung [abapGit]

Dank des devtoberfestes habe ich diese Woche einiges über github gelernt. Ein cooles Feature möchte ich gerne mit euch teilen:

## Github actions

Github actions können verwendet werden, um auf github Ereignisse reagieren zu können. Ein wichtiges Ereignis ist der Commit einer Datei (push).

## TODO to Issue Action

Mit einer YAML-Definition kann man mit einer *Action* auf die Ereignisse reagieren. Ich habe im github Marketplace die Action TODO to issue action gefunden und einmal ausprobiert.

Mit dieser Action ist es möglich auf einen bestimmten *Marker* zu reagieren, der im Quelltext vorkommt. Sinnvoll ist im ABAP-Umfeld das Doppelte Anführungszeichen, dass einen Kommentar einleitet, gefolgt von dem Wort „TODO“. Findet der Workflow beim Einchecken (push) eines Sourcecodes diesen Marker, dann erstellt die Action automatisch ein Issue im github Repository.



## Test Repository

Um das Ganze auszuprobieren, habe ich ein Testrepository im github angelegt: [TODO to Issue](#).

## Vorbereitungen

Um TODO to Action Issues in deinem Repository zu aktivieren, musst du zwei Dinge tun:

- Erstellen der Workflowdatei
- Anlegen des TODO Labeles

## YAML-Workflowdatei

Um den Workflow einzurichten, brauchst du nur eine YML-Datei in dem Ordner `.github/workflows` erstellen. Wahrscheinlich existiert der Ordner noch nicht. Dann lege ihn einfach an.

Die Datei kann zum Beispiel wie folgt aussehen:

```
name: todo2issue
on: push
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@master
      - name: TODO to Issue
        uses: alstr/todo-to-issue-action@v2.0
        with:
          REPO: ${{ github.repository }}
          BEFORE: ${{ github.event.before }}
          SHA: ${{ github.sha }}
          TOKEN: ${{ secrets.GITHUB_TOKEN }}
          LABEL: "\"TODO"
          COMMENT_MARKER: "\""
          CLOSE_ISSUES: true
        id: "todo"
```

Die wichtige Anpassung für ABAP-Programme ist die Definition des `COMMENT_MARKERs` mit `„`. Da das doppelte Anführungszeichen ein Sonderzeichen ist, muss es mit dem Escape-Zeichen `‚` erstellt werden.

Die Definition des `*` als Kommentarmarker hat bei mir zu einem Syntaxfehler im YAML-File geführt. Das doppelte Anführungszeichen finde ich aber eh sinnvoller.

## Label TODO

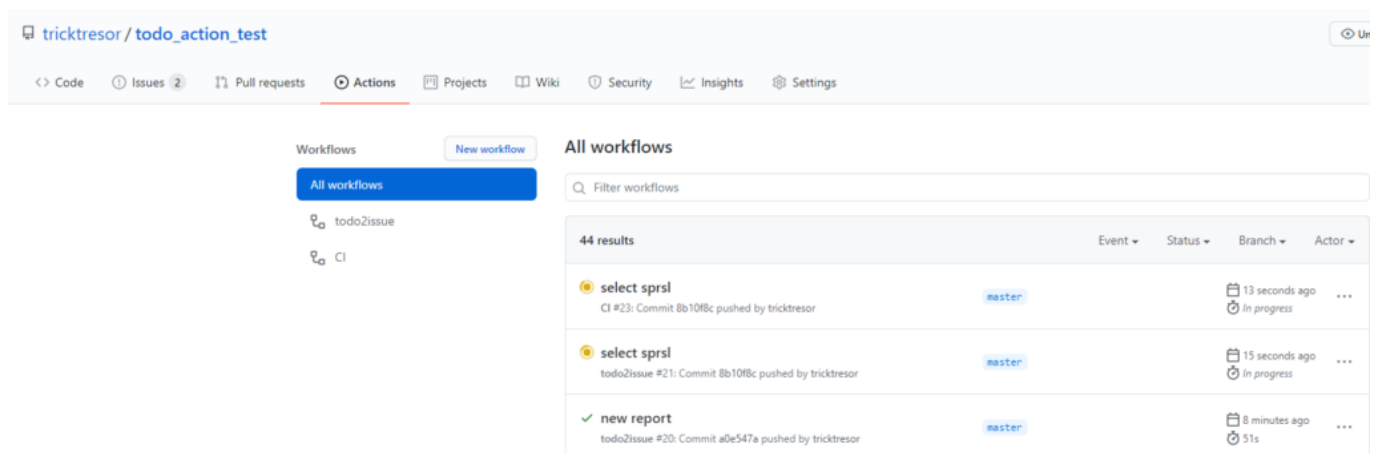
Wechsele in deinem Repository zu der Registerkarte *Issues* und lege ein Issue an. Auf der rechten Seite kannst du aus den verfügbaren Labels auswählen. `TODO` existiert noch nicht und du musst es anlegen.

# Anwendung

Um die Anlage eines Issues zu prüfen, habe ich ein einfaches Programm erstellt und einen TODO-Kommentar eingefügt:

```
LOOP AT langS INTO DATA(lang). "TODO use ALV grid for display
WRITE: / lang-sptxt.
ENDLOOP.
```

Beim Einchecken mit abapGit wird auf das Ereignis *push* reagiert und der Workflow startet:

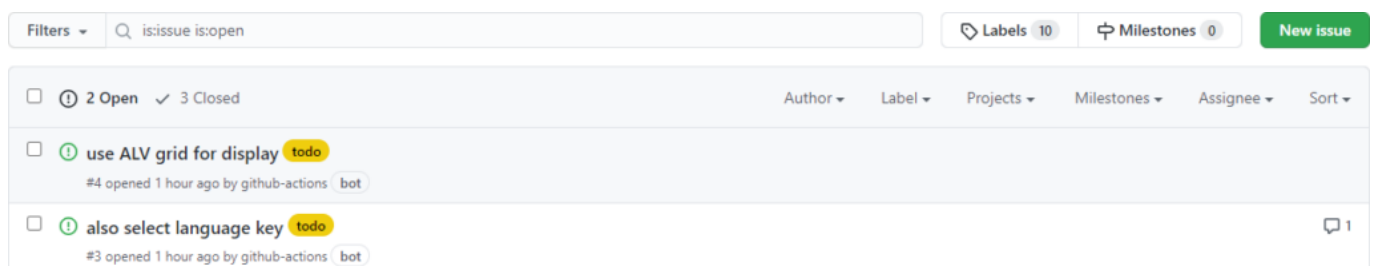


The screenshot shows the GitHub Actions interface for the repository `trickresor/todo_action_test`. The 'All workflows' section is active, displaying a list of workflow runs. The table below summarizes the visible runs:

Workflow Name	Event	Status	Branch	Actor	Time
select sprsl	CI #23: Commit 8b10f8c pushed by trickresor	In progress	master	trickresor	13 seconds ago
select sprsl	todo2issue #21: Commit 8b10f8c pushed by trickresor	In progress	master	trickresor	15 seconds ago
new report	todo2issue #20: Commit a0e547a pushed by trickresor	Completed	master	trickresor	8 minutes ago

Registerkarte „Actions“ direkt nach dem push

Die Verarbeitung dauert etwa eine Minute. Danach ist das Issue in der Registerkarte *Issues* vorhanden:



The screenshot shows the GitHub Issues interface for the repository `trickresor/todo_action_test`. The search filter is set to `is:issue is:open`. The list of issues is as follows:

Issue Title	Label	Author	Time
use ALV grid for display	todo	github-actions bot	#4 opened 1 hour ago
also select language key	todo	github-actions bot	#3 opened 1 hour ago

Die automatisch generierten Aufgaben in der Registerkarte „Issues“

# Features

In der Action *TODO to Issue Action* gibt es noch ein paar zusätzliche Features, die sehr interessant sind:

## Closing issues

Wenn das Label `CLOSE_ISSUES` auf `TRUE` gesetzt wird, dann erkennt die Action, wenn ein `TODO` aus dem Quelltext entfernt wurde. Was will man mehr?

## Multiline Todos

Mit mehrzeiligen Kommentaren kann ein Issue mit einer Beschreibung angelegt werden:

```
"TODO Come up with a more imaginative greeting
"Everyone uses hello world and it's boring.
```

## Dynamic Labels

Fügt man dem `TODO`-Kommentar das Schlüsselwort *labels:* voran, dann kann man zusätzliche Tags (Labels) mitgeben:

```
labels: enhancement, help wanted
```

## Fazit

Ich finde, dass das Feature *TODO to Action Issue* extrem hilfreich ist. Die Verwaltung funktioniert nach ersten Tests sehr gut. Im Issue selbst ist gut erkenntlich, in welchem Zusammenhang das `todo` steht. Ein direkter Link zum Sourcecode ist ebenfalls vorhanden. Auch das Löschen von Todos funktioniert gut. Änderungen im Quelltext erzeugen keine doppelten Issues. Wenn ein Issue geschlossen wurde, wird es erneut angelegt, sobald ein Issue mit dem gleichen Text wieder auftaucht.



Viel Spaß beim Verwaltungen deiner Projekte mit Issues!