

# Spickzettel: GitHub Issues – Aufgaben, Fehler, Diskussionen verwalten

## Ziel

Mit GitHub Issues strukturiert Aufgaben, Bugs und Ideen im Projekt dokumentieren und bearbeiten.

---

## Was ist ein Issue?

- Eine **diskussionsfähige Einheit** im Repository
  - Geeignet für: Fehler, Feature-Wünsche, ToDos, Dokumentationshinweise, Refactorings, Diskussionen
- 

## Issue erstellen

- Klicke im Repo auf "**Issues**" → "**New Issue**"
- Wähle ggf. ein Template (z. B. Bug, Feature Request)
- Titel: kurz & prägnant
- Beschreibung: mit Reproduktionsschritten, Screenshots, Kontext

## Markdown nutzen

**\*\*Schritte zur Reproduktion:\*\***

1. ...
2. ...

**\*\*Erwartetes Verhalten:\*\***

...

**\*\*Screenshots:\*\***

![Bild](link)

---

## Labels

- Kategorisieren & filtern: bug, enhancement, question, hw, doc, wontfix, ...
  - Farben erleichtern die visuelle Erfassung
  - Labels können automatisiert zugewiesen werden (z. B. über Bot oder Vorlage)
- 

## Meilensteine & Zuordnung

- Meilenstein = Ziel oder Release, dem das Issue zugeordnet ist
  - Issue kann Personen zugewiesen werden ("Assignee")
  - Verlinkung mit PRs: Fixes #42 → schließt das Issue beim Merge automatisch
- 

## Lifecycle & Status

Phase	Beschreibung
Offen	Neu erstellt, noch offen
In Arbeit	Bearbeitung läuft
Review	Wird geprüft (z. B. via PR)
Geschlossen	Erledigt oder verworfen

## Best Practices

- Nur **ein Thema pro Issue** behandeln
  - Klarer, aussagekräftiger Titel
  - Beschreibung strukturiert (was, warum, wie?)
  - Labels & Meilensteine frühzeitig setzen
  - Diskussionen offen dokumentieren (nicht via Chat)
- 

## Tipp für Automatisierung

- Fixes #Nummer in PR-Beschreibung = schließt Issue beim Merge
  - closes, resolves, related to funktionieren ebenso
- 

Issues sind die zentrale Kommunikationsschnittstelle für offene Punkte im Projekt – sie strukturieren Zusammenarbeit, machen Fortschritt sichtbar und unterstützen Reviewprozesse.