

Tools für dein (S)EP GitLab/Bugtracker

IEEE Student Branch Passau
Ben Borgards

23. Juli 2020

Plattform für gesamten Entwicklungslifecycle

- ▶ Hostet git Projekt
- ▶ Benutzer/zugriffsverwaltung
- ▶ Issue-Tracker
- ▶ CI/CD integration
- ▶ Wiki
- ▶ Statistiken
- ▶ und vieles mehr


- ▶ **Für vieles!**
- ▶ Echtes Bugtracking
- ▶ Verknüpfung SCM/Bugtracker
- ▶ Aufgabenverwaltung
- ▶ Timetracking
- ▶ Diskussion
- ▶ Wiki
- ▶ Code-Review
- ▶ etc.

Einwand

Aber ich kann doch einfach TODOs und FIXMEs in den Code schreiben!

- ▶ Was ist mit nicht Code-bezogenen Bugs?
 - ▶ Die Datenbank schmiert ständig ab.
 - ▶ Fig 2.1 im Entwurf ist verpixelt.
- ▶ Welche TODOs gehören in welchen Milestone?
- ▶ Wer kümmert sich um welches TODO?
- ▶ Feedback und Details passen nicht in TODOs.

Ben Borgards > SEP-SS20-Team2 > Issues > #103

Closed Opened 4 minutes ago by  Ben Borgards

Reopen issue

New issue

Selenium Tests sind flaky



Die Tests mit Selenium sind flaky! Im Debugmodus funktioniert es einwandfrei, jedoch im Normalen nicht zuverlässig.

Linked issues  0 +



0




0



Oldest first ▾

Show all activity ▾

 New branch unavailable

Discussion 1 Designs



Ben Borgards @borgards changed due date to July 10, 2020 4 minutes ago



Ben Borgards @borgards changed milestone to %Validierung 4 minutes ago



Ben Borgards @borgards added **Bug** **To Do** labels 4 minutes ago



Ben Borgards @borgards assigned to @borgards 4 minutes ago



Ben Borgards @borgards · 1 minute ago

Maintainer



Das liegt daran das die Seite noch nicht ganz geladen ist! Selenium versucht mit Elementen zu interagieren, die noch nicht vollständig geladen sind. Lösung: Selenium muss an bestimmten stellen warten bis die Seite/ das Element geladen ist!



Ben Borgards @borgards added **Doing** label and removed **To Do** label 54 seconds ago



Ben Borgards @borgards removed **Doing** label just now

- ▶ Bugzilla / Mantis
- ▶ Trac / Apache Bloodhound
- ▶ Redmine
- ▶ JIRA
- ▶ Trello
- ▶ Github / Gitlab / Bitbucket

- ▶ Sehr bekannt und verbreitet
- ▶ Solide Bugtracker auch für Großprojekte
- ▶ Self-hosted (Perl/PHP)
- ▶ Etwas angestaubt
- ▶ Features
 - ▶ Bug Tracking & Diskussion
 - ▶ Time Tracking

- ▶ Apache Bloodhound ist ein Trac-Fork
- ▶ SVN, git, etc. integrierbar
- ▶ Self-hosted (Python)
- ▶ Features:
 - ▶ Task Tracking & Diskussion
 - ▶ Sourcecode Browsing
 - ▶ Roadmap
 - ▶ Wiki
 - ▶ Statistiken

- ▶ Konzept und Features ähnlich wie Trac
- ▶ SVN, git, etc. integrierbar
- ▶ Self-hosted (Ruby on Rails)
- ▶ Features:
 - ▶ Gantt chart
 - ▶ Task Tracking & Diskussion
 - ▶ Roadmap
 - ▶ Wiki
 - ▶ Time Tracking

- ▶ Mächtiges Fehlerverwaltungs- und Projektmanagement-Tool
- ▶ Self-hosted (Java) oder gehostet bei `atlassian.com`
- ▶ Als Student kostenlos (nur self-hosted Variante)
- ▶ SVN, git, etc. integrierbar
- ▶ Features:
 - ▶ Issue Tracking & Diskussion
 - ▶ Roadmap
 - ▶ Time Tracking

- ▶ Aufgabenverwaltungstools im Kanban-Stil
- ▶ Kostenlose Webversion
- ▶ SVN, git, etc. integrierbar
- ▶ Mit vielen weiteren Programmen verknüpfbar

- ▶ Sehr rundes, integriertes Git-Ökosystem
- ▶ Von der Uni gehostet
 - ▶ `https://git.fim.uni-passau.de`
- ▶ Alternativen: GitHub / Bitbucket
 - ▶ Private Repos für Studenten kostenlos
- ▶ Features:
 - ▶ Issue-Tracker
 - ▶ Inline-Code-Editing und Kommentare
 - ▶ Milestones
 - ▶ Wiki
 - ▶ Umfangreiche Statistiken

- ▶ Redmine: `http://demo.redmine.org`
- ▶ JIRA: `https://de.atlassian.com/software/jira/demo`
- ▶ Github: `https://github.com/explore`
- ▶ Gitlab: `https://about.gitlab.com/handbook/marketing/product-marketing/demo`

- ▶ **Empfehlung:** Github / GitLab / Bitbucket
 - ▶ Optimal, wenn man ohnehin Git verwenden will
 - ▶ Aber auch mit SVN möglich!
Einfach „leeres“ Github Repo anlegen und Bucktracker verwenden
- ▶ Eigenes Bugzilla/Redmine/JIRA, wenn man das Know-How und die Lust hat
- ▶ **Auf keinen Fall nichts!**