

















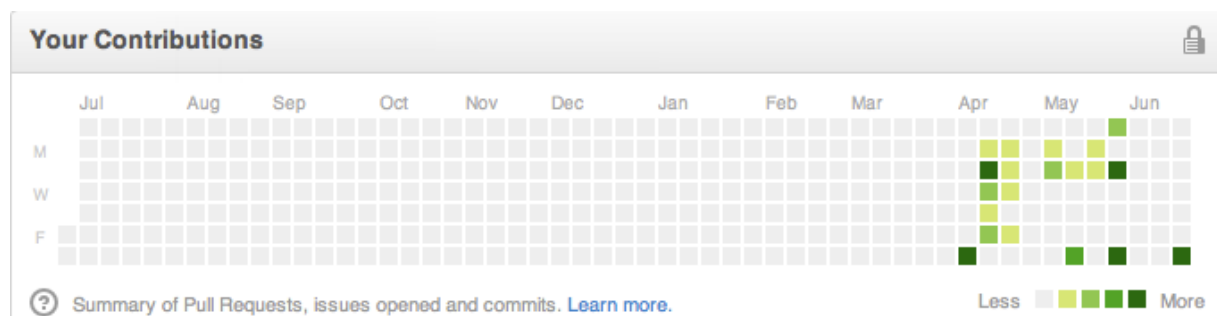
Part 1:

Zur Übung wurden die OPR Aufgaben regelmäßig in ein Repository geladen. Dazu wurde ein GIT Plugin in Eclipse verwendet und das online Repository GitHub.

Aktueller Stand des Repositorys:

moved UE10 once again		
 Neidschel	authored 11 hours ago	latest commit 970eee09db 
 OPR2_UE01_NiglM_MC12028	Complete homework	2 months ago
 OPR2_UE02_NiglM_MC12028	Initial commit	2 months ago
 OPR2_UE03_NiglM_MC12028	Initial commit	2 months ago
 OPR2_UE04_NiglM_MC12028	change formating	2 months ago
 OPR2_UE05_NiglM_MC12028	work finished	2 months ago
 OPR2_UE06_NiglM_MC12028	testing done, exceptions done	2 months ago
 OPR2_UE07_NiglM_MC12028	Project finished	2 months ago
 OPR2_UE08_NiglM_MC12028	initial commit UE10	21 days ago
 OPR2_UE09_NiglM_MC12028	initial commit UE10	21 days ago
 OPR2_UE10_NiglM_MC12028	moved UE10 back	11 hours ago
 dev	moved Ue10	11 hours ago
 doc	Added Folders	13 hours ago
 .gitignore	Complete homework	2 months ago
 README.md	Initial commit	2 months ago

Verwendungsgrad des Repository über die letzten 2 Monate:



Folgende Ignore Liste wurde verwendet:

```
.*
*.class
*/bin
*/lib
*.jar
```

Frage:

Zentrale VCS sind so aufgebaut, dass es ein Repository Online gibt, von dem alle Entwickler Updates beziehen und Änderungen committen. Dabei ist es sehr wahrscheinlich, dass alle Entwickler mit der aktuellen Version arbeiten, sofern regelmäßige Updates durchgeführt werden. Dateninkonsistenzen sollten dadurch relativ schnell bemerkt werden.

Dezentrale VCS haben den Vorteil, dass jeder Nutzer eines solchen VCS ein eigenes Repository am Rechner hat. Das hat den Vorteil, dass der Nutzer nicht an Internet gebunden ist, um Änderungen durchzuführen bzw. Rollbacks durchzuführen. Natürlich gibt es auch ein Online Repository, zu dem mehrere User zugreifen können, aber die Grundidee ist, dass es mehrere Repository gibt, die alle gleichwertig sind und welche einfach miteinander synchronisiert werden können. Einziger leichter Nachteil ist, dass bei diesen VCS mehr darauf geachtet werden muss, dass jeder Entwickler die neueste Version hat, da Änderungen leicht in einem lokalen Repository „liegen“ bleiben können, wenn diese nicht regelmäßig online gespielt werden.

Part 2:


Ein Bug wurde in der OPRAufgabe 04 entdeckt. Um diesen Bug zu beheben, wurde das bugtracking System Lighthouse verwendet.
(<http://neidschel.lighthouseapp.com/dashboard>)

Ein neues Ticket wurde erstellt:

The screenshot shows the Lighthouse bug tracking system interface. At the top, there is a navigation bar with tabs: Welcome, Overview, Tickets (selected), Messages, Milestones, and debugging. Below the navigation bar, there is a search bar with the text "Find tickets: responsible:me state:open" and a "Show me:" dropdown menu. Below the search bar, there is a section titled "Bulk Edit Tickets". Below that, there is a message: "Displaying 1 entry for your open tickets sorted by updated." To the right of this message are two buttons: "Excel" and "CSV". Below the message, there is a table with the following columns: #, State, Title, Milestone, Assigned to, and Age ↓. The table contains one row with the following data: # 1, State new, Title Bug in der contains Methode, Milestone debugging, Assigned to neidschel (at gmx), and Age 1 minute old. Below the table, there is a message: "You can create new tickets for this project by emailing ticket+neidschel.110719-xqvzaxka@lighthouseapp.com. [How does this work?](#)" At the bottom of the page, there is a footer with the text: "activereload/entp Lighthouse Support Our blog FAQ API Privacy Terms of service" and "Copyright © 2007-2013 activereload/entp".


#	State	Title	Milestone	Assigned to	Age ↓
1	new	Bug in der contains Methode	debugging	neidschel (at gmx)	1 minute old

Folgende Meldung gab es für den Ticketempfänger:



neidschel (at gmx)
 You have [1 open ticket](#) for OPRAufgabe 04

Folgende Ticketnachricht wurde angegeben:



Bug in der contains Methode
 Reported by [neidschel \(at gmx\)](#) | June 29th, 2013 @ 11:09 PM | in [debugging](#)
 Wenn die Contains Methode aufgerufen wird, dann wird eine NullPointerException geworfen.


[opr2_ue04_niglm_mc12028.zip](#)
39.1 KB [delete](#)

Nach diesem Ticket wurde das Projekt überarbeitet:


[OPR2_UE04_NiglM_MC12028](#)
debugged contains method
2 minutes ago


Danach wurde das Ticket als gelöst gewertet:



Bug in der contains Methode
 Reported by [neidschel \(at gmx\)](#) | June 29th, 2013 @ 11:09 PM | in [debugging](#)
 Wenn die Contains Methode aufgerufen wird, dann wird eine NullPointerException geworfen.


[opr2_ue04_niglm_mc12028.zip](#)
39.1 KB

Comments and changes to this ticket



neidschel (at gmx)
 June 29th, 2013 @ 11:50 PM
 State changed from "new" to "resolved"
 Die contains Methode wurde überarbeitet. Nach Ausführlichen Tests sollte sie keine NullPointerException mehr werfen.

Folgendes ist in der Ticketübersicht zu sehen:

#	State	Title	Milestone	Assigned to	Age ↓
4	✓ resolved	Bug-in-der-contains-Methode	debugging	neidschel (at gmx)	3 minutes old

Frage 1:

Ein BugReport enthält:

- den Ersteller des Tickets
- den Verantwortlichen für das Ticket
- den Status des Tickets (neu, gelöst, in arbeit, etc...)
- die Ticket ID
- den Zeitpunkt der Erstellung des Tickets
- den Titel des Tickets
- den Inhalt des Tickets (der eigentliche Fehlerreport)
- die Relevanz des Tickets (schwerer Fehler der dringend gelöst werden muss etc)
- eventuell den zugehörigen Meilenstein des Projekts
- eventuell einen Tag für die Kategorie
- eventuell Anhänge

Frage 2:

Der Ersteller des Tickets muss bestätigen, dass der Fehler behoben wurde. Das ist wichtig, da einerseits der Fehler nicht zur Gänze so behoben wird wie es notwendig wäre und andererseits kann es passieren, dass ein Fehler behoben wird der gar nicht gemeint war und nur zufällig in der selben Klasse war.

Die einzige Möglichkeit sicher zu sein, dass der Fehler behoben wurde, ist dass derjenige der den Fehler gefunden hat bestätigt, dass der Fehler nach dem debuggen nicht mehr vorhanden ist.

Frage 3:

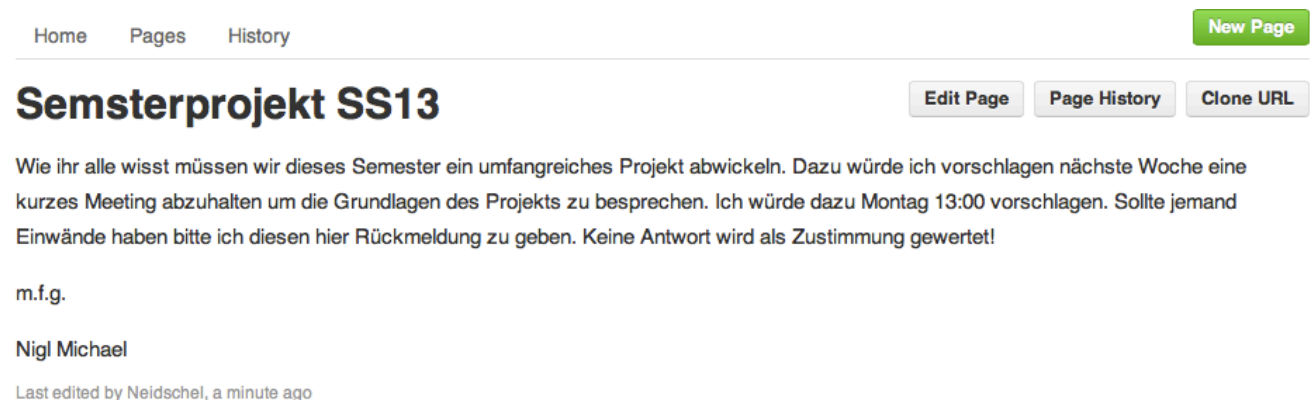
Redmine ist ein webbasiertes Projektmanagement-Tool. Es kann für Benutzer- und Projektverwaltung, Diskussionsforen, Wikis, zur Ticketverwaltung oder Dokumentenablage genutzt werden. Weltweit wird Redmine bei mehreren großen Projekten verwendet, unter anderem bei dem Issue-Tracking-System der Programmiersprache Ruby oder dem Entwicklerportal von TYPO3.

Trac ist ein freies, webbasiertes Projektmanagement-Werkzeug zur Softwareentwicklung. Es enthält unter anderem einen Bugtracker zum Erfassen und Verwalten von Programmfehlern und Erweiterungswünschen. SourceForge bietet Trac als „Hosted App“ an. Es gibt mindestens drei aktiv entwickelte Projekte, Redmine, dessen Fork ChiliProject sowie Apache Bloodhound als jüngsten, direkten Fork, die wesentliche Funktionalitäten von Trac übernommen haben und mit eigenen Zielsetzungen weiterentwickeln.

Bugzilla ist eine freie Bugtracker-Webapplikation zur Verwaltung und Verfolgung von Hinweisen auf Fehler in Software und Wünschen für zusätzliche Funktion. Viele große Open-Source-Projekte – auch Wikipedia – verwenden Bugzilla, um Fehlermeldungen und Wünsche von Benutzern zu sammeln. Neben diesen Grundaufgaben dienen manche Installationen noch für den Aufruf, bestimmte Paketversionen zu testen oder den Fortschritt einer Aufgabe zu dokumentieren.

Part 3:

Der Wikieintrag wurde in dem von GitHub mitgelieferten Wiki erstellt:



Das dazu gehörende Meetingprotokoll liegt im doc Ordner im Repository.

Frage:

Das Wiki eignet sich zur Terminabsprache. Es kann ein Termin eingetragen werden und jeder Projektteilnehmer kann dem Termin zustimmen, absagen und gleichzeitig andere Vorschläge liefern.

Das Wiki kann als Frageplattform verwendet werden. Bei Unklarheiten, Problemen oder anderen Missverständnissen kann über dieses Wiki mit den Kollegen Problemlösungen gesucht werden. Sowohl terminlicher Natur als auch Probleme mit Quellcode können hier besprochen werden.

Das Wiki kann als Fortschrittsprotokoll verwendet werden. Mit neuen Wikieinträgen können abgeschlossene Arbeitspakete bzw. derzeitige Arbeitsvorgänge bekannt gegeben werden um Kollegen zu signalisieren wo sie jetzt weiter Arbeiten können, oder woran sie gerade nicht Arbeiten sollten.

Das Wiki kann vom Auftraggeber verwendet werden um den aktuellen Stand des Projekts einzusehen bzw. zu erfragen.

Dateien können in Posts allen Mitarbeitern zum Download zur Verfügung gestellt werden, die eventuell nicht im Repository sein sollten.