

Redmine - Feature #13147

workflow enhancement: overleap status if allowed

2013-02-13 00:25 - Terence Mill

Status: New	Start date:
Priority: High	Due date:
Assignee:	% Done: 0%
Category: Issues workflow	Estimated time: 0.00 hour
Target version:	
Resolution:	
Description	
<p>We really would like to see roles in workflows with status ranstation a followed by b followed by a>b>c being able to set status from a tp c directly without being necessary to set b explicit.</p> <p>B can be overleaped and c set directly from a if role is allowed to set c in status transitio chain.</p> <p>This is very useful if you make workflows being able to handle small to big process and in small case some steps can de left out if user is allowed to.</p> <p>At the moment the user has to do some clicking mania ti set all status between a and c (or even z)</p>	

History

#1 - 2013-02-13 00:32 - Jean-Philippe Lang

Why don't you just allow a->c in the workflow configuration for this role?

#2 - 2013-02-13 07:15 - Terence Mill

The reason is that this combinations (a to c) results from more than one role you need to have

role 1 (manager): allows a->b

role 2 (tester): allows b>c

I can't configure allow a>c, that would need to create a "workaround" role and habe to give this persons this role instead of both roles to be able to set the status directly. Thats what i am doing at the moment (e.g: ManagerAndTester), but acceptance is bad and it has many problems . That is really annoying as it also results in technical roles (many combinations, e. manager and devloper, manageranddeveloper andtester) and blows the roles list for rights management then too, also in project overview etc..

You right if you think wtf they have manageandtesteranddevelopers in one person and you f. right. ;/ I try to change this all the time, but they won't hear me .

However you kann the roles hopefully you umndertand the problems results from "one user has many roles" and therefore has combinations of status flows which shall be overleapable.

#3 - 2013-02-16 16:42 - Jean-Philippe Lang

I understand your problem. But we would end up in a situation where someone with role 1 and role 2 could do things that neither role 1 nor role 2 can do. It could be misleading and not desirable.

#4 - 2013-02-16 21:43 - Terence Mill

No, you are wrong. Its already possible, but cumbersome.

You have to do more clicks and results in more (useless) form change events (aka more history). Of course moving from a->c must respect all required fields, what mean if from a-> b is a required field then user must set this field if overleaping b to c directly. That needs a ajax re-rendering of the issue form as soon as status field is changed, to make such conditions checked before save request.

#5 - 2013-02-16 22:11 - Jean-Philippe Lang

Terence Mill wrote:

No, you are wrong. Its already possible, but cumbersome.

No, I'm not wrong. Doing a->b->c is not just doing a->c with more clicks. If no role is allowed to skip the status b in the workflow, I don't think that a combination of roles should let you do that.

#6 - 2013-02-16 22:37 - Terence Mill

I don't mean to skip the status but instead of sending the form twice is possible to do several steps in one. The conditions of all steps merge together and you have the sum of all steps packed into one change of the form. It is already possible but need more than one step.

Where at the end is the difference if you do

1. status a>b
 2. set required field 1 and status b>c
- or

1. set required fields 1 and set status a>c

Please give an example where the difference at the end result is. The restriction and conditions holds in both case from my understanding.

#7 - 2013-02-17 00:05 - Jean-Philippe Lang

Say role 1 is allowed to do a->b1 and a->b2, role 2 is allowed to do b1->c and b2->c, status b1 has a required field f1 and b2 has a required field f2. A user with role 1 + 2 would be allowed to do a->c. What happens with f1 and f2?

#8 - 2013-02-17 09:14 - Terence Mill

Either b1 or b2 or both has to be set. Sure that only possible with JS and ajax validation.