

Scrumtool : Gestion de projet de type SCRUM

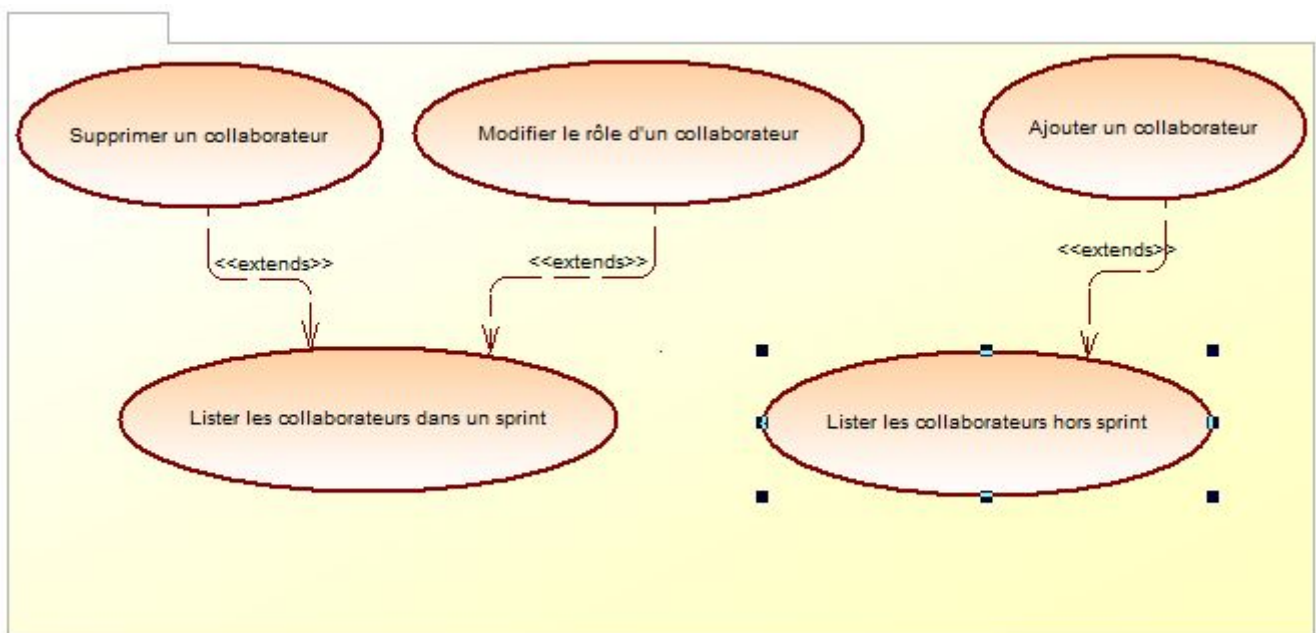
Gestion des collaborateurs par sprint

Description :

Cette fonctionnalité permet d'ajouter, de modifier le rôle et de supprimer des collaborateurs dans un sprint.

1 - Fonctionnalités

Acteur : Administrateur



2 - Vues

SCRUMTOOL

SCRUMTOOL

by TEAMSIO

Collaborator(s) in sprint

Other collaborator(s)

Name	Role
user user	Developer

Edit Role

Collaborator :

user user

Role :

Save

Edit role

Remove

[illegible]

3 - Contrôleurs

3.1 CeditCollaboratorsSprint - Add collaborator

```
view.getBtnEditCollaborators().addSelectionListener(new SelectionListener() {

    @Override
    public void widgetSelected(SelectionEvent arg0) {
        if (view.getBtnEditCollaborators().getText() == "Remove") {
            // Code remove collaborator
        } else {
            ParticipateId id = new ParticipateId(sprint.getId(),
collaboratorActif.getId(), roleActif.getId());
            Participate pr = new Participate(id, sprint, roleActif,
collaboratorActif);
```

```
        DAO.addParticipate(pr);

view.getTableViewerAddCollaborators().remove(collaboratorActif);
        view.getTableViewerRemoveCollaborators().add(pr);

        view.getBtnEditCollaborators().setEnabled(false);
        view.getTableRole().setEnabled(false);

        Utils.MsgBox(view.getShell(), "Collaborator add successfully",
"information");
    }

}

});
```

3.2 CeditCollaboratorsSprint - Remove collaborator

```
view.getBtnEditCollaborators().addSelectionListener(new SelectionListener() {

    @Override
    public void widgetSelected(SelectionEvent arg0) {
        if (view.getBtnEditCollaborators().getText() == "Remove") {
            Collaborator col = new
Collaborator(participateActif.getCollaborator().getId(),
participateActif.getCollaborator().getLogin(),
participateActif.getCollaborator().getPassword(),
participateActif.getCollaborator().getFirstname(),
participateActif.getCollaborator().getLastname(),
participateActif.getCollaborator().getEmail(),
participateActif.getCollaborator().getAdministrator(),
participateActif.getCollaborator().getUserstories(),
participateActif.getCollaborator().getParticipates(),
participateActif.getCollaborator().getPlayroles(),
participateActif.getCollaborator().getComments());

            DAO.delParticipate(participateActif);

view.getTableViewerRemoveCollaborators().remove(participateActif);
            view.getTableViewerAddCollaborators().add(col);

            view.getSashFormRemoveCollaborators().setWeights(new int[] { 1,
0 });

            view.getBtnEditCollaborators().setEnabled(false);
            view.getBtnEditRole().setEnabled(false);

            Utils.MsgBox(view.getShell(), "Collaborator delete from
Sprint", "information");
        } else {
            // Code add collaborator
        }

    }

});
```

```
});
```

3.3 CeditCollaboratorsSprint - Edit rôle

```
view.getBtnSaveEditRole().addSelectionListener(new SelectionListener() {  
  
    @Override  
    public void widgetSelected(SelectionEvent arg0) {  
  
        ParticipateId id = new ParticipateId(sprint.getId(),  
participateActif.getCollaborator().getId(), roleEditActif.getId());  
        Participate pr = new Participate(id, sprint, roleEditActif,  
participateActif.getCollaborator());  
  
        DAO.delParticipate(participateActif);  
        DAO.addParticipate(pr);  
  
        view.getTableViewerRemoveCollaborators().remove(participateActif);  
        view.getTableViewerRemoveCollaborators().add(pr);  
  
        view.getSashFormRemoveCollaborators().setWeights(new int[] { 1, 0  
});  
  
        view.getBtnEditRole().setEnabled(false);  
        view.getBtnSaveEditRole().setEnabled(false);  
        view.getBtnEditCollaborators().setEnabled(false);  
  
        Utils.MsgBox(view.getShell(), "Role edit successfully",  
"information");  
    }  
});
```

From:

<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:

<http://slamwiki2.kobject.net/etudiants/2014/bertrand/scrumtool7?rev=1401750020>

Last update: **2019/08/31 14:32**

