Scrumtool : Gestion de projet de type SCRUM
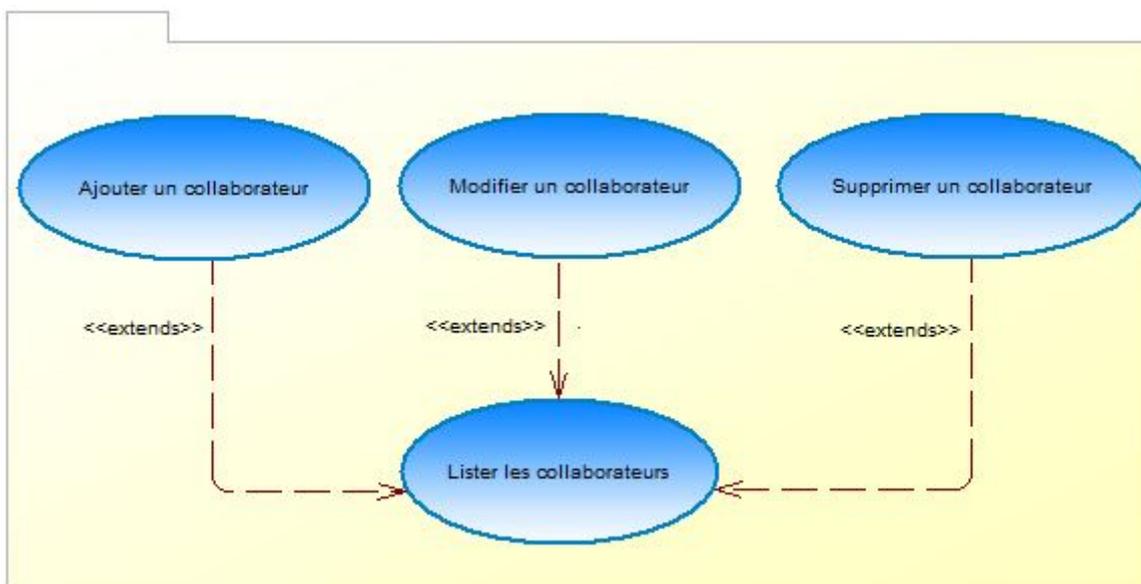
Gestion des collaborateurs

Description :
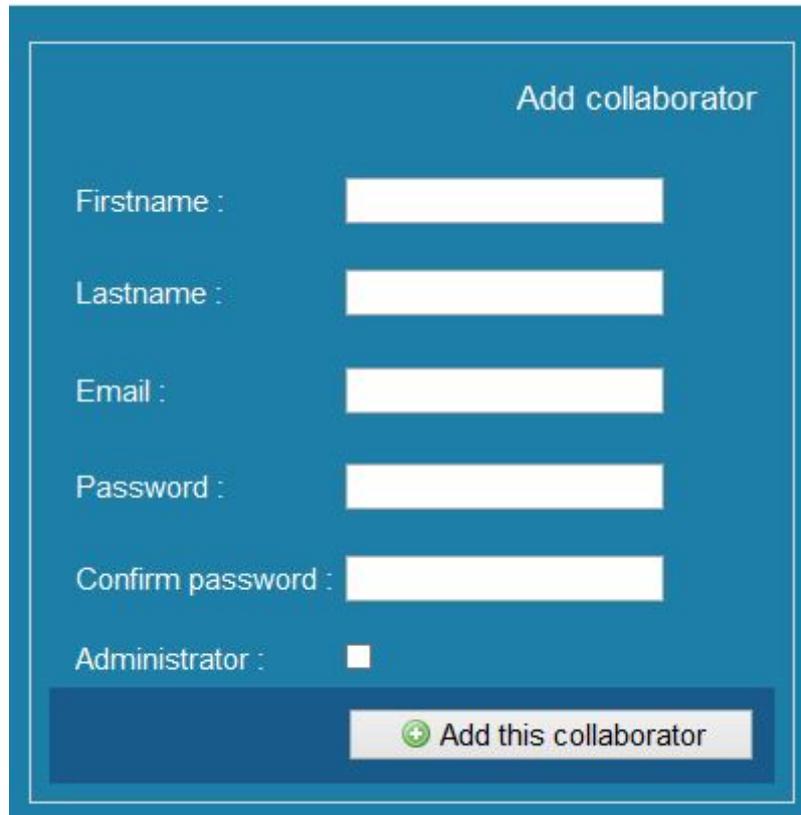
Cette fonctionnalité permet d'ajouter, de modifier et de supprimer des collaborateurs.

1 - Fonctionnalités

Acteur : Administrateur

2 - Vues

3 - Contrôleurs

3.1 Ccollaborators - Add Collaborator

```java
view.getBtnCollaborators().addSelectionListener(new SelectionListener() {

            @Override
            public void widgetSelected(SelectionEvent arg0) {
                if (view.getBtnCollaborators().getText() == "Add this
collaborator") {
                    if (view.getTxtFirstnameCollaborators().getText() == "" ||
view.getTxtLastNameCollaborators().getText() == "" ||
view.getTxtEmailCollaborators().getText() == "" ||
view.getTxtPasswordCollaborators().getText() == "" ||
view.getTxtConfirmPasswordCollaborators().getText() == "") {
                        Utils.MsgBox(view.getShell(), "Please, complete all fiels",
"information");
                    } else {
                        boolean nomUtilise = false;
                        for (Collaborator collaborator : DAO.getCollaborators()) {
                            if
(collaborator.getEmail().equals(view.getTxtEmailCollaborators().getText())) {
                                nomUtilise = true;
                            }
                        }
                        if (nomUtilise == true) {
                            Utils.MsgBox(view.getShell(), "A collaborator already
has this email", "information");
                        } else {
                            if
(!view.getTxtPasswordCollaborators().getText().equals(view.getTxtConfirmPasswordCol
laborators().getText())) {
                                Utils.MsgBox(view.getShell(), "Passwords are not
the same", "information");
```

```
                            } else {
if(view.getCheckButtonCollaborator().getSelection()){
                                    admin=true;
                                }else{
                                    admin=false;
                                }
                                Collaborator collaborator = new
Collaborator(view.getTxtFirstnameCollaborators().getText(),
view.getTxtPasswordCollaborators().getText(),
view.getTxtFirstnameCollaborators().getText(),
view.getTxtLastNameCollaborators().getText(),
view.getTxtEmailCollaborators().getText(), admin, null, null, null, null);
                                DAO.SQLCollaborator(collaborator, "insert");
view.getTableViewerCollaborators().add(collaborator);
                                Utils.MsgBox(view.getShell(), "Collaborator is
create", "information");
                                init();

                            }
                        }
                    }
                } else {
                    // Code update collaborator
                }
    });
```

3.2 Ccollaborators - Edit Collaborator

```java
view.getBtnCollaborators().addSelectionListener(new SelectionListener() {

            @Override
            public void widgetSelected(SelectionEvent arg0) {
                if (view.getBtnCollaborators().getText() == "Add this
collaborator") {
                    // Code add collaborator
                } else {
                    if (view.getTxtFirstnameCollaborators().getText() == "" ||
view.getTxtLastNameCollaborators().getText() == "" ||
view.getTxtEmailCollaborators().getText() == "" ||
view.getTxtPasswordCollaborators().getText() == "") {
                        Utils.MsgBox(view.getShell(), "Please, complete all fiels",
"information");
                    } else {
                        Boolean email = false;
                        java.util.List<Collaborator> verifEmail =
DAO.getCollaborators();
                        for (Collaborator emailV : verifEmail) {
                            if
(view.getTxtEmailCollaborators().getText().equals(emailV.getEmail())) {
                                if
(emailV.getEmail().equals(collaboratorActif.getEmail())) {
                                    email = false;
                                } else {
                                    email = true;
                                }
                            }
                        }

                        if (email == true) {
                            Utils.MsgBox(view.getShell(), "Email already exist",
"information");
                        } else {
                            if
(!view.getTxtPasswordCollaborators().getText().equals(view.getTxtConfirmPasswordCol
laborators().getText())) {
                                if
(view.getTxtConfirmPasswordCollaborators().getText() == "") {
                                    if
(view.getTxtPasswordCollaborators().getText().equals(collaboratorActif.getPassword(
))) {
if(view.getCheckButtonCollaborator().getSelection()){
                                            admin=true;
                                        }else{
                                            admin=false;
                                        }
                                        collaboratorActif.setAdministrator(admin);
collaboratorActif.setFirstname(view.getTxtFirstnameCollaborators().getText());
collaboratorActif.setLastname(view.getTxtLastNameCollaborators().getText());
collaboratorActif.setEmail(view.getTxtEmailCollaborators().getText());
collaboratorActif.setPassword(view.getTxtPasswordCollaborators().getText());
                                        DAO.SQLCollaborator(collaboratorActif,
"update");
view.getTableViewerCollaborators().refresh();
                                        Utils.MsgBox(view.getShell(), "Collaborator
```

```
is update", "information");
                                     } else {
                                         Utils.MsgBox(view.getShell(), "This is not
your current password. \nPlease confirm your password.", "information");
                                     }
                               } else {
                                   Utils.MsgBox(view.getShell(), "Passwords are
not the same", "information");
                               }
                           } else {
if(view.getCheckButtonCollaborator().getSelection()){
                                   admin=true;
                               }else{
                                   admin=false;
                               }
                               collaboratorActif.setAdministrator(admin);
collaboratorActif.setFirstname(view.getTxtFirstnameCollaborators().getText());
collaboratorActif.setLastname(view.getTxtLastNameCollaborators().getText());
collaboratorActif.setEmail(view.getTxtEmailCollaborators().getText());
collaboratorActif.setPassword(view.getTxtPasswordCollaborators().getText());
                               DAO.SQLCollaborator(collaboratorActif, "update");
                               view.getTableViewerCollaborators().refresh();
                               Utils.MsgBox(view.getShell(), "Collaborator is
update", "information");
                           }
                       }
                   }
               }
           }
});
```

3.3 Ccollaborator - Delete Collaborator

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

         @Override
         public void widgetSelected(SelectionEvent arg0) {

               MessageBox messageBoxCollaborator = new MessageBox(view.getShell(),
SWT.ICON_QUESTION | SWT.YES | SWT.NO);
               messageBoxCollaborator.setText("SCRUMTOOL");
               messageBoxCollaborator.setMessage("Do you really want to delete
this collaborator ?");
               int responseCollaborator = messageBoxCollaborator.open();
               if (responseCollaborator == SWT.YES) {
                   DAO.SQLCollaborator(collaboratorActif, "delete");
                   view.getTableViewerCollaborators().remove(collaboratorActif);
                   view.getGrpCollaborators().setVisible(false);
               }

         }
});
```

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

Permanent link:
http://slamwiki2.kobject.net/etudiants/2014/bertrand/scrumtool3

Last update: **2019/08/31 14:21**