

# Scrumtool : Gestion de projet de type SCRUM

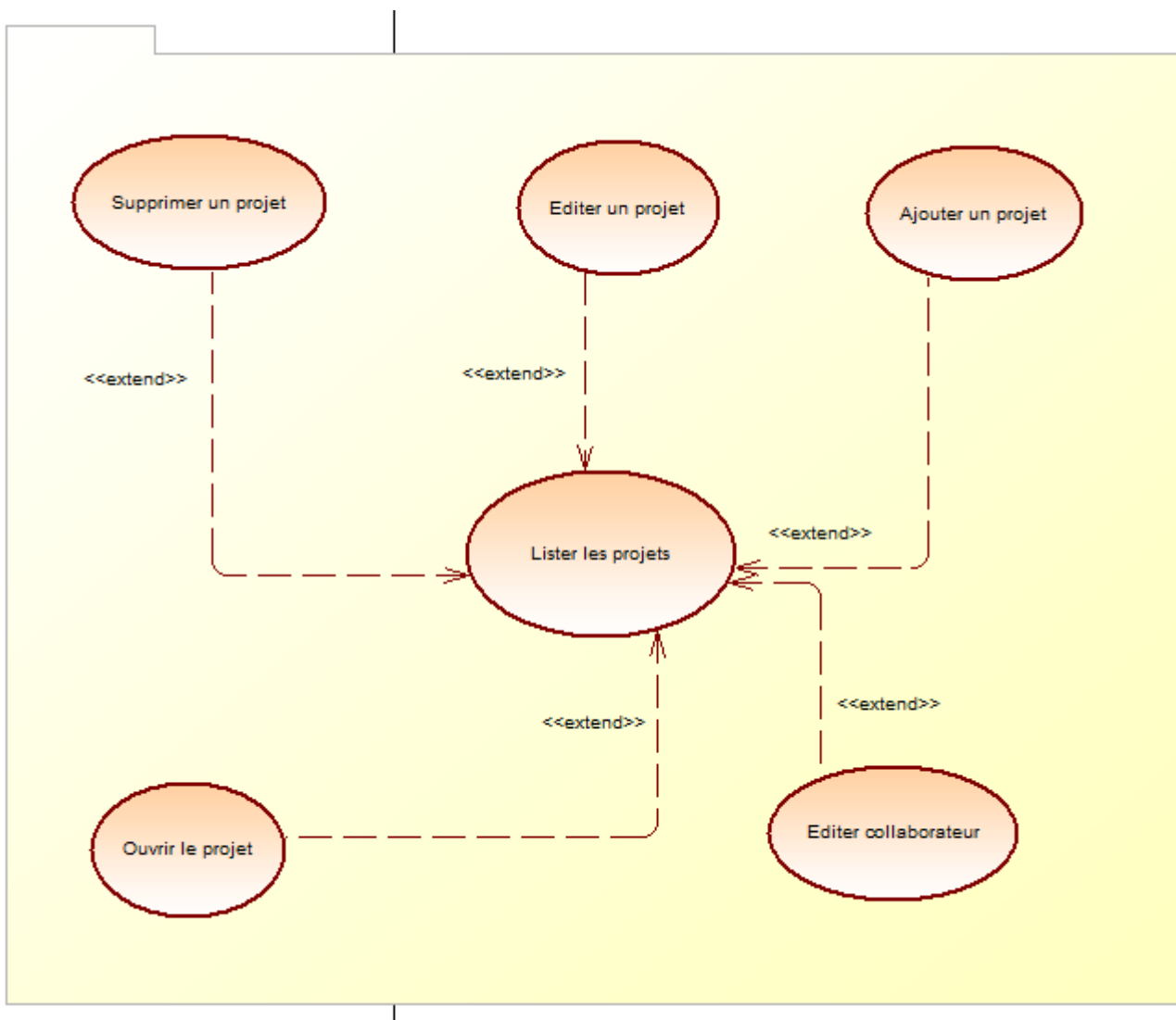
## Gestion des projets

Description :

Cette fonctionnalité est essentielle pour l'application. En effet elle permet l'ajout, la modification ainsi que la suppression d'un projet.

### 1- Fonctionnalités

Acteur : Administrateur



### 2- Vues



The image shows a user interface for adding a new project. It features a blue-themed window titled 'Add product'. Inside, there is a 'Project name' label followed by a text input field containing 'Exemple : My Project'. Below that is a 'Description' label followed by a larger, empty text area. At the bottom right of the window, there is a button with a green plus icon and the text 'Add this project'.

```

view.getBtnProducts().addSelectionListener(new SelectionListener() {

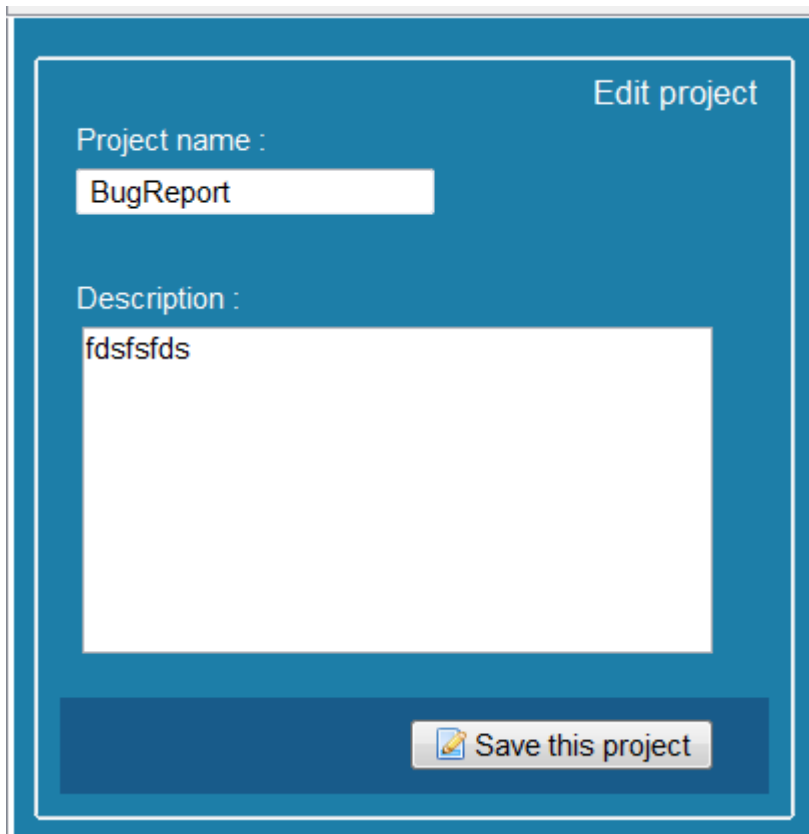
    @Override
    public void widgetSelected(SelectionEvent arg0) {
        if (view.getBtnProducts().getText() == "Add this project") {
            if (view.getTxtNameProject().getText().equals("") ||
view.getTxtDescriptionProduct().getText().equals("")) {
                Utils.MsgBox(view.getShell(), "Please complete all fields",
"information");
            } else {

                boolean nomUtilise = false;
                for (Product product : DAO.getProducts()) {
                    if
(product.getName().equals(view.getTxtNameProject().getText())) {
                        nomUtilise = true;
                    }
                }
                if (nomUtilise == true) {
                    Utils.MsgBox(view.getShell(), "A project already has
this name", "information");
                } else {
                    Product product = new
Product(view.getTxtNameProject().getText(),
view.getTxtDescriptionProduct().getText(), null, null, null);
                    DAO.SQLProduct(product, "insert");
                    view.getTableViewerProducts().add(product);
                    Utils.MsgBox(view.getShell(), "Project is create",
"information");
                }
            }
        }
    }
});

```

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

### 3.2 Cproducts - Edit Product



```
view.getBtnProducts().addSelectionListener(new SelectionListener() {  
  
    @Override  
    public void widgetSelected(SelectionEvent arg0) {  
        if (view.getBtnProducts().getText() == "Add this project") {  
  
            // Code add product  
  
        } else {  
            if (view.getTxtDescriptionProduct().getText().equals("")) {  
                Utils.MsgBox(view.getShell(), "Please complete all fields",  
"information");  
            } else {  
                productActif.setName(view.getTxtNameProject().getText());  
productActif.setDescription(view.getTxtDescriptionProduct().getText());  
            }  
        }  
    }  
});
```

```
        DAO.SQLProduct(productActif, "update");
        view.getTableViewerProducts().refresh();
        Utils.MsgBox(view.getShell(), "Project is update",
"information");
    }
}
});
```

### 3.3 Cproducts - Delete Product

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

@Override
public void widgetSelected(SelectionEvent arg0) {

    MessageBox messageBoxproduct = new MessageBox(view.getShell(),
SWT.ICON_QUESTION | SWT.YES | SWT.NO);
    messageBoxproduct.setText("SCRUMTOOL");
    messageBoxproduct.setMessage("Do you really want to delete this
project ?");

    int responseProduct = messageBoxproduct.open();
    if (responseProduct == SWT.YES) {
        DAO.SQLProduct(productActif, "delete");
        view.getTableViewerProducts().remove(productActif);
        view.getGrpProject().setVisible(false);
    }

}

});
```

### 3.4 Cproducts - Edit collaborator

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

@Override
public void widgetSelected(SelectionEvent arg0) {
    EditCollaboratorsProduct vAddCollaborators = new
EditCollaboratorsProduct();
    CeditCollaboratorsProduct cEditCollaborators = new
CeditCollaboratorsProduct(vAddCollaborators);
    vAddCollaborators.init();
    cEditCollaborators.init(productActif);
    cEditCollaborators.listener(productActif);
    vAddCollaborators.open();
}

});
```

### 3.5 Cproducts - Open product

```
view.getTltnOpenProduct().addSelectionListener(new SelectionListener()  
{  
  
    @Override  
    public void widgetSelected(SelectionEvent arg0) {  
        OverviewProduct overview = new OverviewProduct();  
        overview.init();  
        CoverviewProduct cOverview = new CoverviewProduct(overview);  
        cOverview.listener(userActif,productActif);  
        cOverview.init(userActif, productActif);  
        cOverview.reload(userActif, productActif);  
        view.close();  
        overview.open();  
    }  
  
});
```

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

Permanent link:  
<http://slamwiki2.kobject.net/etudiants/2014/bertrand/scrumtool1?rev=1401199074>

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

