

InterfaceJira.js - Bibliothèque TestComplete de création de ticket dans Jira

Version : 1.01

Auteur : Biache Benoit - bbiache@rationalconsulting.fr

Rôle	2
Prérequis	2
Propriétés et méthodes publiées	2
Propriétés	2
Méthodes	3
Usage.....	4
Intégration.....	4
Création simple.....	4
Création riche	4
Ajout direct.....	4
Ajout sur ticket spécifique.....	5
Ajout et contrôle.....	5
Exemple complet avec integration Squash	5
Messages.....	6

Rôle

Cette bibliothèque permet, depuis un script de test automatisé TestComplete, la création d'anomalie dans Jira ainsi que de l'ajout a posteriori de pièces jointes à une anomalie existante.

Prérequis

Cette bibliothèque fonctionne à partir de TestComplete 14.

Elle utilise les APIs de Jira version 7.XX.

Le paramétrage des propriétés suivantes est obligatoire :

- `jira.settings.url`
- `jira.settings.login`
- `jira.settings.pwd`
- `jira.settings.base`

Ces propriétés sont décrites dans la section Propriétés et méthodes publiées.

Les éléments suivants sont nécessaires pour la gestion des pièces jointes :

- droit d'exécution de la ligne de commande **CMD**,
- **Cur1** disponible (par défaut dans Windows 10),
- pour l'envoi de la capture écran, droit d'écriture dans le dossier **%temp%**,
- pour l'envoi des pièces jointes, droit de lecture des pièces jointes

Propriétés et méthodes publiées

Propriétés

L'objet `interfaceJira.jira` porte les propriétés de paramétrage et de résultat.

```
// Bloc de paramétrage
jira.settings = { // Url d'accès aux APIs
    url           : "https://jira.seb.com/rest/api/latest/issue/",
    // Login du compte utilisé
    login        : "TestAuto.jira.admin",
    // Mot de passe du compte utilisé
    pwd          : "GROUPESEB",
    // Encodage authentification, valorisation automatique
    credentials  : "",
    // Clé du projet concerné (pas L'ID mais bien la clé)
    project      : "IILAT",
    // Url de base pour l'accès direct aux tickets créés
    base         : "https://jira.seb.com/projects/",
    // La valeur littérale du type de ticket créé
    issuetype    : "Bug",
```

```

// La valeur littérale du niveau de priorité par défaut des tickets
// créés
issuepriority : "High",
// La valeur littérale du rapporteur par défaut
reporter      : "TestAuto.jira.admin",
// Optionnel - Cookeat, Sandbox, Aspirobot, Hybris
components   : "",
// True alors des messages supplémentaires de debug sont consignés
debug        : true,
// True alors n'effectue pas les requêtes réellement
emulate      : true,
// True alors les messages de réussite sont loggés
verbose      : true
};

// Bloc de résultat
jira.result = { result : -1, // -1 = non exécuté, 0 = échec, 1 = succès
                error  : "", // Chaîne d'erreur si elle s'est produite, chaîne vide si
                             // aucune erreur
                key    : "", // Clé du ticket créé
                id     : "", // Identifiant du ticket créé
                url    : ""  // Url d'accès direct du ticket créé
};

// Types d'attributs de Log pour une mise en valeur sans déclencher warnings ou erreurs
jira.logSuccess      = Log.CreateNewAttributes();
jira.logSuccess.FontColor = 0x00A000;
jira.logSuccess.Bold   = true;
jira.logWarning      = Log.CreateNewAttributes();
jira.logWarning.FontColor = 0x00A0C7;
jira.logWarning.Bold   = true;
jira.logError        = Log.CreateNewAttributes();
jira.logError.FontColor = 0x0010D0;
jira.logError.Bold    = true;

```

Méthodes

```
function SendFileJira(fileUrl = "", key = "")
```

La méthode SendFileJira permet d'ajouter une pièce jointe à un ticket existant dans Jira.

Paramètres :

fileUrl	String	Obligatoire	Nom et chemin complet du fichier à attacher, exemple "C:\\tools\\test.txt"
key	String	Optionnel	Clé du ticket à attacher, si elle n'est pas fournie alors utilise la clé du dernier ticket par la méthode CreateJira

Retour :

Renvoie une chaîne vide si tout s'est bien exécuté, ou la description de l'erreur si une erreur s'est produite
 Une entrée de log est également réalisée détaillant soit les éléments de la pièce jointe attachée ou soit l'erreur rencontrée

```
function CreateJira(summary = "", description = "", components = "", attachments = "", screen =
"", priority = "", type = "", project = "", reporter = "")
```

La méthode CreateJira permet de créer un ticket dans Jira.

Paramètres :

summary	String	Obligatoire	Titre du ticket
description	String	Obligatoire	Description du ticket
components	String	Optionnel	Composant concerné par le ticket, si il n'est pas fourni alors utilise la valeur par défaut jira.settings.components
attachments	Array	Optionnel	Tableau de chaînes contenant chacune le nom et chemin complet des fichiers à attacher
screen	Picture	Optionnel	Objet image à attacher en pièce jointe au ticket, par défaut chaîne vide pour ne pas attacher de capture écran
priority	String	Optionnel	Valeur littérale de la priorité du ticket, si elle n'est pas fournie alors utilise la valeur par défaut jira.settings.issuepriority , exemple "Majeur"
type	String	Optionnel	Valeur littérale du type du ticket, si elle n'est pas fournie alors utilise la valeur par défaut jira.settings.issuetype , exemple "Correctif"
project	String	Optionnel	Clé du projet à rajouter le ticket, si elle n'est pas fournie alors utilise la valeur par défaut jira.settings.project
reporter	String	Optionnel	Rapporteur à associer au ticket, si il n'est pas fourni alors utilise la valeur par défaut jira.settings.reporter

Retour :

Renvoie True si tout s'est bien exécuté, ou False si une erreur s'est produite

Une entrée de log est également réalisée détaillant soit les éléments du ticket créé ou soit l'erreur rencontrée

Usage

Intégration

Ajouter le fichier **interfaceJira.js** à votre projet de test en tant qu'item existant dans la section **Script**.

Pour utiliser les fonctionnalités, la bibliothèque doit être intégrée via une déclaration d'inclusion dans le script de test automatisé (première ligne du script).

```
//USEUNIT interfaceJira
```

Création simple

Créer un ticket simple concernant le composant **Aspirobot** avec les paramètres par défaut du type de ticket, de priorité et du rapporteur :

```
CreateJira("Test connectivité", "A ignorer, test, Acceptance_fonctionnelle", "Aspirobot");
```

Création riche

Créer un ticket associé à un cas de test (nécessite intégration **interfaceSquash**), concernant le composant **Hybris** en y joignant une capture d'écran et deux pièces jointes :

```
CreateJira("Test connectivité", "A ignorer, test, Acceptance_fonctionnelle\nLien vers le cas de test: " + interfaceSquash.jira.settings.tcurl, "Hybris", ["C:\\tools\\gfx\\nik\\dramatic-sep2.np", "C:\\OEMLOGO\\LOGO.BMP"], Sys.Desktop.Picture());
```

Ajout direct

Ajouter un fichier au dernier ticket qui a été créé dans la session de test courante :

```
SendFileJira("C:\\tools\\Fonts\\18cents.ttf");
```

Ajout sur ticket spécifique

Ajouter un fichier à un ticket spécifique :

```
SendFileJira("C:\\tools\\Fonts\\18cents.ttf", "ILAT-29");
```

Ajout et contrôle

Vérifier que l'ajout s'est bien passé et récupérer l'url du ticket qui a été créé :

```
if (SendFileJira("C:\\tools\\Fonts\\18cents.ttf", "ILAT-29") == "")
    Log.Message("Ajout bien effectué, l'url est : " + interfaceJira.globals.result.url)
else
    Log.Message("Echec de l'ajout, l'erreur est : " + interfaceJira.globals.result.error);
```

Exemple complet avec intégration Squash

```
//USEUNIT interfaceJira
//USEUNIT interfaceSquash

function Acceptance_fonctionnelle() {
    // Récupérer l'itération id d'après Les noms
    GetIterationIdSquash("PREPROD", "EPHEMERE");

    // Démarrer Le cas de test 17853
    InitializeSquash("17853");
    SetTestCaseResultSquash("", "RUNNING");
    // Ici Le code de test du 1er pas
    // ...
    // ...
    // Indiquer que La première étape est réussie
    SetStepResultSquash("1", "SUCCESS", "Ca répond vite en plus");
    // Ici Le code de test du 2eme pase
    // ...
    // ...
    // Indiquer que La deuxième étape est réussie
    SetStepResultSquash("2", "SUCCESS", "Rien a dire");

    // Démarrer Le cas de test 17854
    InitializeSquash("17854");
    SetTestCaseResultSquash("", "RUNNING");
    // Ici Le code de test du 1er pas
    // ...
    // ...
    // Indiquer que La première étape est réussie
    SetStepResultSquash("", "SUCCESS", "Ca répond vite en plus");

    // Démarrer Le cas de test 17855
    InitializeSquash("17855");
    // L'accès a La plateforme concernée par Le test n'est pas possible, on L'indique
    SetTestCaseResultSquash("", "UNTESTABLE", "La plateforme de recette n'est pas disponible");

    // Démarrer Le cas de test 17856
    InitializeSquash("17856");
    SetTestCaseResultSquash("", "RUNNING");
    // Ici Le code de test du 1er pas
    // ...
    // Une erreur se produit, on créé Le ticket associé en y joignant une capture d'écran et 2
    pièces jointes
    CreateJira("Test connectivité", "A ignorer, test, Acceptance_fonctionnelle\\nLien vers le cas
de test: " + interfaceSquash.jira.settings.tcurl, "Aspirobot", ["C:\\tools\\gfx\\nik\\dramatic-
sep2.np", "C:\\OEMLOGO\\LOGO.BMP"], Sys.Desktop.Picture());
```

```

// On rajoute une pièce jointe supplémentaire au ticket venant d'être créé
SendFileJira("C:\\tools\\Fonts\\18cents.ttf");
// Indiquer L'échec de la première étape
SetStepResultSquash("", "BLOCKED", "Ca répond pas vite du tout en plus\nCréation JIRA:\n " +
interfaceJira.globals.result.url);
// On indique au niveau cas de test également afin de prendre en compte L'url du ticket
associé dans Le commentaire (API v1.1.0), inutile en API v1.1.4
SetTestCaseResultSquash("", "BLOCKED","Ah ben là tout est cassé !\nCréation JIRA:\n " +
interfaceJira.globals.result.url);

// Démarrer Le cas de test 17852
InitializeSquash("17852");
SetTestCaseResultSquash("", "RUNNING");
// Ici Le code de test du 1er pas
// ...
// Une erreur se produit, on créé Le ticket associé en y joignant une pièce jointe
CreateJira("Test connectivité", "A ignorer, test, Acceptance_fonctionnelle\\nLien vers le cas
de test: " + interfaceSquash.jira.settings.tcurl, "Hybris", ["C:\\tools\\gfx\\nik\\dramatic-
sep2.np"]);
// Indiquer L'échec de la première étape
SetStepResultSquash("", "FAILURE", "Cassé la machine ... \nCréation JIRA:\n " +
interfaceJira.globals.result.url);
// On indique au niveau cas de test également afin de prendre en compte L'url du ticket
associé dans Le commentaire (API v1.1.0), inutile en API v1.1.4
SetTestCaseResultSquash("", "FAILURE","Ca a pas marché ben dis donc !\nQu'est ce qu'on fait
?\nBon on peut continuer pour voir, c'est pas bloqué .. \nCréation JIRA:\n " +
interfaceJira.globals.result.url);
}

```

Messages

Méthode	Action	Message
SendFileJira	Debug	DEBUG -- Ajout pièce jointe JIRA demandé Fichier local : XXX Ticket à modifier : XXX
SendFileJira	Succès	JIRA - L'ajout de la pièce jointe au ticket XXX a réussi Ticket modifié : XXX Pièce jointe ajoutée : XXX
SendFileJira	Echec	JIRA - L'ajout de la pièce jointe au ticket XXX a échoué Le paramètre 'fileUrl' n'est pas fourni !
SendFileJira	Echec	JIRA - L'ajout de la pièce jointe au ticket XXX a échoué Le paramètre 'key' n'est pas fourni !
SendFileJira	Echec	JIRA - L'ajout de la pièce jointe au ticket XXX a échoué Le fichier indiqué XXX n'est pas trouvé !
SendFileJira	Echec	JIRA - L'ajout de la pièce jointe au ticket XXX a échoué Echec CURL JIRA Url appelée : XXX Code : XXX
CreateJira	Debug	DEBUG -- Création ticket JIRA demandé Résumé : XXX Description : XXX Priorité : XXX Type : XXX Projet : XXX Composant : XXX Rapporteur : XXX Pièce(s) jointe(s) : XXX

CreateJira	Debug	DEBUG -- Paramètres JIRA Url : XXX Login : XXX Pwd : XXX Url base : XXX Projet : XXX Type : XXX Priorité : XXX Rapporteur : XXX Composant : XXX Debug : XXX
CreateJira	Debug	DEBUG - Corps de la requête JIRA XXX
CreateJira	Debug	DEBUG -- Résultat JIRA Résultat : XXX Erreur : XXX Id ticket : XXX Clé ticket : XXX Url ticket : XXX
CreateJira	Succès	JIRA - La création du ticket XXX a réussi L'url est : XXX
CreateJira	Warning	JIRA - La création du ticket 'XXX' est incomplète L'url est : XXX L'erreur est : XXX
CreateJira	Echec	JIRA - La création du ticket 'XXX' a échoué Le paramètre 'summary' n'est pas fourni !
CreateJira	Echec	JIRA - La création du ticket 'XXX' a échoué Le paramètre 'description' n'est pas fourni !
CreateJira	Echec	JIRA - La création du ticket 'XXX' a échoué Echec API JIRA Url appelée : XXX Code : XXX XXX
CreateJira	Echec	JIRA - La création du ticket 'XXX' a échoué Echec sauvegarde capture écran JIRA XXX
CreateJira	Echec	JIRA - La création du ticket 'XXX' a échoué <<Message d'erreur de SendFileJira>>