FAQs

Friday, March 31, 2017   6:49 PM

R&D, Findings, Suggestions

Suggestions in Test Complete Page object files

Friday, March 31, 2017   7:34 PM

Subject	**Suggestions in Test Complete Page object files**
From	Sruthi Cheekoti
To	Core Automation; Platform Automation
Cc	QAAutomation TLs
Sent	Friday, March 31, 2017 7:20 PM

Hi All,

We are creating the page objects in Test complete scripts and I have few suggestions to implement in the page object files to reduce scripting effort.

1. I felt it would be better to declare a common variable on the top of the file and using that variable for getting the object references.
   Like this:

```
1    //USEUNIT UtilsCommon
2    var str_function_name = "";
3    var ObjectRef = false;
4
5    /***********************************************************************/
6    /* Name: ResearchPaneGrid                                           */
7    /* Parameters:RMS Window and StepId                                 */
8    /* Returns Grid reference if it exists else returns false           */
9    /***********************************************************************/
10   function ResearchPaneGrid(RMSwindow,strStep_ID){
11     var str_function_name = "Browse_Tab_Refs::ReseachPaneGrid";
12     Log.AppendFolder(str_function_name);
13     try{
14         ObjectRef = AssertNoError("UtilsCommon.WaitForObject",RMSwindow,["Name","ObjectType"],["WinFormsO
15         if(!ObjectRef.Exists){
16             Log.Error("In Step-Id :"+ strStep_ID+" My Grid ResearchPane doesn't exist in the window");
17             Log.Picture(Sys.Desktop.Picture(), "Desktop capture");
18         }
19         return ObjectRef;
20     }
21     catch(e){
22         HandleError(e, str_function_name );
23         return false;
24     }
25     finally{
26         Log.PopLogFolder();
27     }
28   }
29
30   /***********************************************************************/
31   /* Name: CalenderPaneGrid                                           */
32   /* Parameters: DockPanel and StepId                                 */
33   /* Returns Dock Panel reference if it exists else returns false.    */
34   /***********************************************************************/
35   function CalenderPaneGrid(DockPanel,strStep_ID){
36     var str_function_name = "Browse_tab_Refs::MyCalenderPaneGrid";
37     Log.AppendFolder(str_function_name);
38     try{
39         ObjectRef = AssertNoError("UtilsCommon.WaitForObject",DockPanel,["Name","ObjectType"],["WinFormsO
40         if(!ObjectRef.Exists){
41             Log.Error("In Step-Id :"+ strStep_ID+" Claender Grid doesn't exists in browse tab");
42             Log.Picture(Sys.Desktop.Picture(), "Desktop capture");
```

2. Checking the 'exists' or 'enabled' properties of an object and logging the error in the page object files itself. Which helps in reducing the scripting effort.

```
/*****************************************************DETAILPANES dETAIL FORM RELATED REFERENCES*********************

/***************************************************************************/
/* Name: CaptionTxtField                                                 */
/* Parameters: Detail Panes Detail form and StepId                       */
/* Returns Caption text field reference if it exists else returns false.  */
/***************************** ****************************************/
function CaptionTxtField(WindowRef,strStep_ID){
    var str_function_name = "OpenRMS_DetailPanes::CaptionTxtField";
    Log.AppendFolder(str_function_name);
    try{
        ObjectRef = AssertNoError("UtilsCommon.WaitForObject",WindowRef,["Name","ObjectType"],["WinFormsObject('teCaption'
        if(!ObjectRef.Exists){
            Log.Error("In Step-Id :"+ strStep_ID+" Caption text field doesn't exists in "+WindowRef.Caption+" window");
            Log.Picture(Sys.Desktop.Picture(), "Desktop capture");
            return false;
        }
        if(!ObjectRef.Enabled){
            Log.Error("In Step-Id :"+ strStep_ID+" Caption text field doesn't enabled in "+WindowRef.Caption+" window");
            Log.Picture(Sys.Desktop.Picture(), "Desktop capture");
            return false;
        }
        return ObjectRef;
    }
    catch(e){
        HandleError(e,str_function_name );
        return false;
    }
    finally{
        Log.PopLogFolder();
    }
}


/***************************************************************************/
```

3. We can also create a method for checking the 'Exists' and 'Enabled' properties and can call the method
   after getting the object reference. And I am going to implement in CodeRed Page objects.

   Thanks,
   Sruthi.

Utility to get the data from JSON file

Monday, May 01, 2017   8:27 PM

Subject	**Created TestComplete Utility to get the data from JSON file.**
From	Sruthi Cheekoti
To	AppQA Hyderabad Automation Team
Sent	Tuesday, April 25, 2017 11:47 PM

Hi Team,

I have created a utility for TestComplete projects to get the data from JSON file. It is available in the UtilsCommon.sj so that everyone can use it.
Utility:

```
function GetDataFromJSON(strFilename){
 var str_function_name = "UtilsCommon::GetDataFromJSON";
 try{
       Log.AppendFolder(str_function_name);
       if((aqString.Find(strFilename, ":\\")) == -1)
              strFilename = g_strUtilDocsPath+strFilename;
        if(!AssertNoError("UtilsCommon.DoesFileExist",strFilename)){
              Log.Error(strFilename+"File doesn't exists");
               return false;
       }
       var data = aqFile.ReadWholeTextFile(strFilename, aqFile.ctANSI);
       var obj = eval("(" + data + ")");
       return obj;
 }
 catch(e) {
    SetError(e,str_function_name);
    return false;
 }
 finally {
       Log.PopLogFolder();
 }
}
```

We can use JSON files to document the hardcoded data(test data).

JSON file:
```
{
'NewItemsMenuWndCaption' : 'Manage New Item Menus',
'NewNewItemMenuWndCaption' :'(New New Item Menu)'
}
```

The Function "**GetDataFromJSON**" is in //depot/smqa/helios/TC8/Utils/UtilsCommon.sj, which reads the JSON file and Converts the whole content/data to a JavaScript Object and return that Object.

In scripts, we can call the function by passing the JSON file name or the JSON file path as a

parameter.

```
var data=AssertNoError("UtilsCommon.GetDataFromJSON","Objects.json");
Log.Message(data. NewItemsMenuWndCaption); //Prints a string - Manage New Item Menus
```

Thanks,
Sruthi.

Boiler Plate Code

Wednesday, January 10, 2018   2:24 PM

Please follow the below instructions to use the boiler plate code for test complete scripts.

Code location: H:\Department\QualityAssurance\QAAutomation\Automation\Boiler Plate\ SJ-Boiler

Note: Please install 3rd party local server like Xampp to run the project and copy the code from the shared directory to the server directory.

Steps to download Boilerplate code for test case:
- Navigate to url: http://localhost/SJBoiler/test.php (example url).
- Click on 'Choose File' and select the excel file (i.e. excel file of required test case downloaded from the justifier).
- Select the optional fields.
- Click on 'Get SJ' button.
- Click on 'Click here to download' link.

Automation Issues

Monday, March 12, 2018   4:22 PM