

Test Automation Using Telerik Test Studio

Script – CrewMonitor – Add_An_Event_To_A_Pairing

//Add_An_Event_To_A_Pairing.tstest.cs

```
using Telerik.TestingFramework.Controls.KendoUI;
using System.Linq;

using ArtOfTest.Common.UnitTesting;
using ArtOfTest.WebAii.Silverlight.UI;

namespace MerlotAero_AutomationProject
{
    public class Add_An_Event_To_A_Pairing : BaseWebAiiTest
    {
        #region [ Dynamic Pages Reference ]

        private Pages _pages;

        /// <summary>
        /// Gets the Pages object that has references
        /// to all the elements, frames or regions
        /// in this project.
        /// </summary>
        public Pages Pages
        {
            get
            {
                if (_pages == null)
                {
                    _pages = new Pages(Manager.Current);
                }
                return _pages;
            }
        }

        #endregion

        // Add your test methods here...
        int columnNumber;
        [CodedStep(@"Type scheduled flight (Effective Date From) to date +1")]
        public void Build_Pairing_CodedStep()
        {
            // Type "03/1/2017" into uxFilterFrom_Date

            var now = DateTime.Now.AddMonths(1);
```

```

        var first = new DateTime(now.Year, now.Month, 1);
        var last = first.AddDays(1);

Pages.MerlotAero0.SilverlightApp.uxFilterFrom_Date.SetText(true,first.ToString(
"MM/dd/yyyy"), 10, 100);
        Manager.Desktop.KeyBoard.SendString("{TAB}");

Pages.MerlotAero0.SilverlightApp.uxFilterTo_Date.SetText(true,
last.ToString("MM/dd/yyyy"), 10, 100);

    }

    [CodedStep(@"Type Pairing Name into filterContains_Textbox")]
    public void Add_An_Event_To_A_Pairing_CodedStep()
    {
        // Type '1014' into filterContains_Textbox
        //string pairingName=GetExtractedValue("PairingName").ToString();
        string pairingName=Utility.getPairingName();

Pages.MerlotAero0.SilverlightApp.filterContains_Textbox.SetText(true,
pairingName, 10, 100, false, true);

    }

    [CodedStep(@"Verify pairing opened successfully in pairing details")]
    public void Build_Pairing_CodedStep2()
    {
        SilverlightApp app =
ActiveBrowser.SilverlightApps()[0]; //Get Silverlight app

        Telerik.WebAii.Controls.Xaml.RadGridView rgv =
app.Find.ByType<Telerik.WebAii.Controls.Xaml.RadGridView>(); //Get RadGrid
        int rowCounter = 1;
        int headCount=1;
        //String PairingName=
GetExtractedValue("PairingName").ToString();
        String PairingName= Utility.getPairingName();
        //get the Header Row Cells and iterate through them
        IList<GridViewHeaderCell> headers =
rgv.HeaderRow.HeaderCells;
        foreach(GridViewHeaderCell header in headers)
        {
            //Instead of a sting (here it's "Unit Price"),
you can use a variable
            if(header.Text.Equals("Pair"))
            {
                columnNumber=headCount;
                Log.WriteLine("Column Number
::"+columnNumber);

                break;
            }
            headCount++;
        }
    }

```

```

        foreach (Telerik.WebAii.Controls.Xaml.GridViewRow gRow
in rgv.Rows) { //Loop to go through all the rows

            Log.WriteLine("<-----Start of row
"+rowCounter.ToString()+"----->"); //Write row number to log
            if(rowCounter==1)
            {
                int cellCounter = 1;
                foreach(Telerik.WebAii.Controls.Xaml.GridViewCell
gCell in gRow.Cells) { //Nested loop; Goes through all the cells within each
row

                    if(cellCounter==columnNumber)
                    {

                        Log.WriteLine("Cell "+ cellCounter.ToString()+"
"+gCell.Text); //We output the cell text to the log;
                        Assert.Equals(PairingName,gCell.Text);
                    }
                    cellCounter++;

                }
            }
            rowCounter++;

        }

    }

    [CodedStep(@"New Coded Step")]
    public void Add_An_Event_To_A_Pairing_CodedStep1()
    {
        ActiveBrowser.Window.Minimize();
        ActiveBrowser.Window.Maximize();
    }

    [CodedStep(@"Press ScrollDown_Button for 306 msec")]
    public void Add_An_Event_To_A_Pairing_CodedStep2()
    {
        // Press ScrollDown_Button for 306 msec
        try
        {
            Pages.MerlotAero0.SilverlightApp.ScrollDown_Button.UserPress(306);
        }
        catch(Exception e)
        {
            Log.WriteLine("Scroll Button is not visible");
        }

    }

    [CodedStep(@"Drag MeetingTextblock to application at (235, 175) from
TopLeftCorner")]
    public void Add_An_Event_To_A_Pairing_CodedStep3()

```

```
        {
            // Drag MeetingTextblock to application at (235, 175) from
            TopLeftCorner
            //
            Pages.MerlotAero0.SilverlightApp.MeetingTextblock.Application.ResizeContainer(0
            , 77, 1366, 500);

            Pages.MerlotAero0.SilverlightApp.MeetingTextblock.User.DragTo(Pages.MerlotAero0
            .SilverlightApp.AddNewTextblock);
        }
    }
```