

Test Automation Using Telerik Test Studio

Script – CrewPlan – Update_Expiarydates_for_Certification

```
// Update_Expiarydates_for_Certification.tstest.cs
```

```
using Telerik.TestingFramework.Controls.KendoUI;
using Telerik.WebAii.Controls.Html;
using Telerik.WebAii.Controls.Xaml;
using System;
using System.Collections.Generic;
using System.Text;
using System.Linq;

using ArtOfTest.Common.UnitTesting;
using ArtOfTest.WebAii.Core;
using ArtOfTest.WebAii.Controls.HtmlControls;
using ArtOfTest.WebAii.Controls.HtmlControls.HtmlAsserts;
using ArtOfTest.WebAii.Design;
using ArtOfTest.WebAii.Design.Execution;
using ArtOfTest.WebAii.ObjectModel;
using ArtOfTest.WebAii.Silverlight;
using ArtOfTest.WebAii.Silverlight.UI;

namespace MerlotAero_AutomationProject
{
    public class Update_Expiarydates_for_Certification : 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
```

```
int columnNumber;  
// Add your test methods here...  
  
[CodedStep(@"Type 03/06/2017 into startDate_ActiveFlagWindow")]  
public void Update_Expirydates__for_Qualification_CodedStep()  
{  
  
    var now = DateTime.Now.AddMonths(1);  
    var first = new DateTime(now.Year, now.Month, 1);  
    var last = first.AddDays(1);  
    // Type "03/06/2017" into startDate_ActiveFlagWindow
```

```
Pages.MerlotAero0.SilverlightApp.startDate_ActiveFlagWindow.SetText(true,  
Utility.getNextMonthDate(1, 10, 100);  
    System.Threading.Thread.Sleep(2000);  
    Manager.Desktop.KeyBoard.SendString("{TAB}");  
}
```

```
[CodedStep(@"Type 03/07/2017 into endDate_ActiveflagWindow")]  
public void Update_Expirydates__for_Qualification_CodedStep2()  
{  
  
    var now = DateTime.Now.AddMonths(1);  
    var first = new DateTime(now.Year, now.Month, 1);  
    var last = first.AddDays(1);  
    // Type "03/07/2017" into endDate_ActiveflagWindow
```

```
Pages.MerlotAero0.SilverlightApp.endDate_ActiveflagWindow.SetText(true,  
Utility.getNextMonthDate(2, 10, 100);  
    System.Threading.Thread.Sleep(2000);  
}
```

```
[CodedStep(@"LeftClick on FilterTextblock")]  
public void Update_Expiarydates_for_Certification_CodedStep1()  
{  
    System.Threading.Thread.Sleep(5000);  
    int extentHeight;  
    int rowHeight;  
    int rowCounter = 1;  
  
    RadGridView grid =  
Pages.MerlotAero0.SilverlightApp.Course_Expiry_EntityDataGridRadgridview;  
    FrameworkElement VirtualizingPanel =  
grid.Find.ByType("GridViewVirtualizingPanel");  
    extentHeight = (int)VirtualizingPanel.GetProperty(new  
AutomationProperty("ExtentHeight", typeof(int)));  
    rowHeight = (int)grid.GetProperty(new  
AutomationProperty("RowHeight", typeof(int)));  
    int totalPageRows = 0;  
    totalPageRows = (extentHeight/rowHeight);
```

```

Log.WriteLine("Total Rows : " + totalPageRows.ToString());
String getCourse=Utility.getCourseCode();
//Log.WriteLine(getCourse+"--have no rows to be update ");
if(totalPageRows!=0)
{
    Log.WriteLine("Course Data found in grid");
}
else
{
    Log.WriteLine("Not Found so updating");
    //
Pages.MerlotAero0.SilverlightApp.Item91Grid.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 33, 56, ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));

Pages.MerlotAero0.SilverlightApp.Item108Grid.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 7, 50, ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));

Manager.Desktop.KeyBoard.KeyPress(System.Windows.Forms.Keys.Clear);

    System.Threading.Thread.Sleep(2000);

Pages.MerlotAero0.SilverlightApp.FilterTextblock0.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 40, 56,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));
    System.Threading.Thread.Sleep(40000);

    //.....search fro expite date range..
    //
Pages.MerlotAero0.SilverlightApp.FilterTextblock0.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 25, 50,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));

Pages.MerlotAero0.SilverlightApp.PARTDropDownButtonToggleButton0.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 58, 33,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));
    System.Threading.Thread.Sleep(10000);

Pages.MerlotAero0.SilverlightApp.PARTFilter1ComboBoxRadcombobox.ToggleDropDown(
);
    System.Threading.Thread.Sleep(5000);

Pages.MerlotAero0.SilverlightApp.IsLessThanTextblock.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 84, 56,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));

```

```

Pages.MerlotAero0.SilverlightApp.Item254Grid.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 36, 70,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));

```

```

        var now = DateTime.Now.AddMonths(0);

```

```

Pages.MerlotAero0.SilverlightApp.PARTDateTimeInputRadwatermarktextbox.SetText(true, now.ToString("MM/dd/yyyy"), 10, 100, false, true);

```

```

Pages.MerlotAero0.SilverlightApp.pairingsFilter_Button.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 61, 50,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys)(0)));
//

```

```

Pages.MerlotAero0.SilverlightApp.flight_FilterCloseButton.User.Click(ArtOfTest.WebAii.Core.MouseClickType.LeftClick, 7, 7,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Pixel);

```

```

    }

```

```

        extentHeight = (int)VirtualizingPanel.GetProperty(new AutomationProperty("ExtentHeight", typeof(int)));
        rowHeight = (int)grid.GetProperty(new AutomationProperty("RowHeight", typeof(int)));
        totalPageRows = 0;
        totalPageRows = (extentHeight/rowHeight);
        Log.WriteLine("Total Rows : " + totalPageRows.ToString());

```

```

int headCount=1;
IList<GridViewHeaderCell> headers = grid.HeaderRow.HeaderCells;
foreach(GridViewHeaderCell header in headers)
{
    if(header.Text.Equals("Expiry"))
    {
        Log.WriteLine("Expiry header found");
        columnNumber=headCount;
        Log.WriteLine("ColumnNumber : " + columnNumber);
        break;
    }
    headCount++;
}
System.Threading.Thread.Sleep(15000);
SilverlightApp app = ActiveBrowser.SilverlightApps()[0]; //Get Silverlight app

```

```

        Telerik.WebAii.Controls.Xaml.RadGridView rgv =
app.Find.ByType<Telerik.WebAii.Controls.Xaml.RadGridView>("uxEntityDataGrid");
//Get RadGrid

```

```

        //GridViewVirtualizingPanel
gvvp=grid.Find.ByType<GridViewVirtualizingPanel>();
// IList<GridViewRow> rows=gvvp.Find.AllByType<GridViewRow>();
Log.WriteLine("Rows count:"+rgv.Rows.Count.ToString());
foreach(Telerik.WebAii.Controls.Xaml.GridViewRow gRow in rgv.Rows){

    Log.WriteLine("<-----Start of row
"+rowCounter.ToString()+"----->");
    if(rowCounter<2)
    {
        Log.WriteLine("rowCounter : " + rowCounter);
        int cellCounter = 1;
        foreach(Telerik.WebAii.Controls.Xaml.GridViewCell gCell
in gRow.Cells){

            if(cellCounter==columnNumber)
            {
                TextBlock tb = gCell.Find.ByType<TextBlock>();
                Log.WriteLine("date in current cell" + tb.Text);

                // DateTime dt= Convert.ToDateTime(tb.Text);
                DateTime dt = DateTime.ParseExact(tb.Text.ToString(),
"MM/dd/yyyy", null);
                Log.WriteLine("Date:"+dt.ToString());

                String
updateddate=dt.AddYears(1).ToString("MM/dd/yyyy");
                Log.WriteLine("UpdatedDate:"+updateddate);

                Log.WriteLine("Reached inside cell");
                gCell.User.Click();
                System.Threading.Thread.Sleep(2000);
                gCell.User.Click();
                Log.WriteLine("Performed First Click");
                //TextBlock tb = gCell.Find.ByType<TextBlock>();

                // Log.WriteLine("date in current cell" + tb.Text);
                // var getUpdateDate= DateTime.Now.AddYears(1);

                Pages.MerlotAero0.SilverlightApp.MMDdYyyyMerlotdatepicker.SetText(true,
updateddate, 10, 100);

                System.Threading.Thread.Sleep(2000);

                Manager.Desktop.KeyBoard.KeyPress(System.Windows.Forms.Keys.Escape);
                System.Threading.Thread.Sleep(2000);
            }
            // else
            //{
            //     Log.WriteLine("cellCounter Not Selected");
            // }
            cellCounter++;
        }
    }
}

```

```

        }
        rowCounter++;
    }

}

[CodedStep(@"Type 'ted' into TextTextbox")]
public void Update_Expiarydates_for_Certification_CodedStep()
{
    String courseCodecertf=Utility.getCourseCodeCertf();
    Pages.MerlotAero0.SilverlightApp.TextTextbox10.SetText(true,
courseCodecertf, 10, 100, false, true);

}

[CodedStep(@"LeftClick on Item165Grid")]
public void Update_Expiarydates_for_Certification_CodedStep2()
{
    // LeftClick on Item108Grid

Pages.MerlotAero0.SilverlightApp.Item108Grid.User.Click(ArtOfTest.WebAii.Core.M
ouseClickType.LeftClick, 15, 72,
ArtOfTest.Common.OffsetReference.TopLeftCorner,
ArtOfTest.Common.ActionPointUnitType.Percentage,
((System.Windows.Forms.Keys) (0)));

}

[CodedStep(@"Desktop command: LeftClick on
PARTApplyFilterButtonButton")]
public void Update_Expiarydates_for_Certification_CodedStep4()
{
    // Desktop command: LeftClick on apply_Filter_Button

Pages.MerlotAero0.SilverlightApp.pairingsFilter_Button.User.Click(ArtOfTest.Web
Aii.Core.MouseClickType.LeftClick, 0, 0,
ArtOfTest.Common.OffsetReference.AbsoluteCenter,
ArtOfTest.Common.ActionPointUnitType.Percentage);

}
}
}

```

