

Add a User Control to a Form

Last Modified on 07/16/2017 7:43 am EDT

When you add a user control to a form, you should use the Sub-View control and set the **VirtualPath** property to the user control ASCX file.

Procedure

1. Create a user control.
2. Sign the user control project.
3. Set the **Inherits** property of the user control to the full class name.

```
Inherits="WebApplication1.WebUserControl1 ,WebApplication1, Version=1.0.0.0, Culture=neutral, PublicKeyToken=4b06ea5bb276e861"
```

4. Put the user's control assembly in the GAC.
5. Put the ASCX file in the Cora SeQUENCE shared resources folder.
6. On the form, add a Sub-View control, and in the markup, set the **VirtualPath** property.

Get the Cora SeQUENCE Form Control (hosting the user control)

```
private FormControl GetFormControl()
{
    Control control = this.Parent;
    FormControl sequenceForm = null;
    while (control != null)
    {
        sequenceForm = control as FormControl;
        if (sequenceForm != null)
        {
            break;
        }
        control = control.TemplateControl;
        //the TemplateControl property for a user control returns itself or null
        if (control != null)
        {
            control = control.Parent;
        }
    }
    return sequenceForm;
}
```

Add the Save Data Function

```

private void SaveData(List values)
{
    FormControl sequenceForm = GetFormControl();
    if (sequenceForm != null)
    {
        var viewModel = this.GetFormUIViewModel();
        if (viewModel != null)
        {
            DataModel dataModel = ((IDataModelAccessor)sequenceForm).DataModel;
            DataContext ctx = viewModel.DataContextManager.GetDataContext(dataModel);
            IDataTable logTable = ctx.GetDataTable("TableQueryName");
            IDataRow logRow;
            foreach (string s in values)
            {
                logRow = logTable.CreateRow();
                //add the rest of the fields here
                //logRow.SetField("[field name]", [value]);
                logRow.SetField("myString", s);
                logTable.InsertOnSubmit(logRow);
            }
            try
            {
                ctx.SubmitChanges();
            }
            catch (Exception ex)
            {
                DiagnosticUtility.EventLog.LogError("Error submitting the data.\n" + ex.ToS
tring());
                throw;
            }
        }
    }
}

```

Add the Event that Triggers the Save Method

```

protected void Button1_Click(object sender, EventArgs e)
{
    List l = new List();
    l.Add("a");
    l.Add("b");
    SaveData(l);
}

```