

Implement Client-Side and Server-Side Code

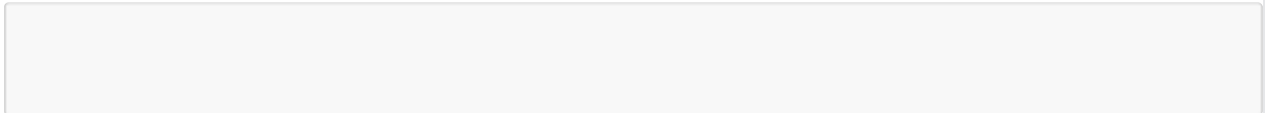
Last Modified on 06/16/2019 3:34 am EDT

Client-Side Code

Implement Client-Side Code

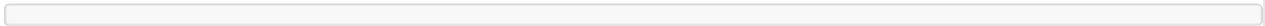
UX Studio enables you to write client-side code, which you can add to a form's markup. The sample below shows code that is executed when a client event occurs, in this case, a user changes a field value.

Client Event



You can register to a wide selection of events for each control you add to the form. The sample below shows how to add JavaScript code to the markup, which is executed when an event occurs, in this case, concatenating the old and new values (see the event above) and placing them into a new div.

Concatenating Old and New Values



Find a Control on the Client Side

Cora SeQuence exposes the following expression parameters in form view expressions.

Expression Parameter	Description
@ncontainer	Reference to a naming container that contains an expression.
@bcontainer	Reference to a binding container that contains an expression.
@template	Reference to the template (form view) that contains an expression.
@page	Reference to a page.

The following code snippet demonstrates how to find a grid control with ID **Grid1** on the client side using the **@ncontainer** parameter:

```
<%@ Control %>
```

Server-Side Code

Implement Server-Side Code

Cora SeQuence enables you to implement server-side code to use in your forms. This functionality enables you to use all available .NET events for form controls.

For example, you can add an event to a button control, which renders a grid automatically when the user

clicks that button.

```
Sq.Grid resultsGridView = (Sq.Grid)this.FindControl("ResultGridView");
    if (ds is IEnumerable)
    {
        primitiveLabel.Visible = false;
        resultsGridView.Visible = true;
        resultsGridView.DataSource = ds;
        resultsGridView.DataBind();
    }
```

Procedure to implement server-side code

1. Create a .NET project with the code, and inherit from Cora SeQuence contols by adding the following lines.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using PNMsoft.Sequence.Forms.Web.UI;
using System.Web.UI.WebControls;
using PNMsoft.Sequence.Linq;
using Sq = PNMsoft.Sequence.Forms.Web.UI.Controls;
using System.Collections;
using PNMsoft.Sequence.Web.UI;
namespace SequenceEx.Forms.Samples
```

2. Compile this project, and place the `.dll` file in the GAC of the Cora SeQuence server.
3. In UX Studio, access the form markup editor, and change the header to inherit from the assembly you created.

```
<%@ Control Inherits="SequenceEx.Forms.Samples.CustomFormControl, SequenceEx.Forms.Samples, Version=1.0.0.0, Culture=neutral, PublicKeyToken=524345e6dd86aeafe"%>
```

Sample Project

This is a sample project server-side code that includes several events.

[ServerSideCodeSample \(1\).rar](#) 