

2.2. Screens

Customization Articles for WMS scripting:
[How to display product image after selecting the item in Picking and Ad-Hoc picking](#)

Screen can be generated by the *ShowScreen* method of the *session* object. There are different types that we can use.

Necessary classes:

Class	Message
Reference	<pre>using Produmex.Foundation.Messages; using Produmex.Foundation.Wwf.Sbo.LocalServices; using Produmex.Foundation.SlimScreen;</pre>

2.2.1. Message Screen type

Parameters

Name	Type	Description
MessageKey	String	Set your message
ShowButton	Bool	-

Example:

```
Message msg = null;
session.ShowScreen(typeof(Produmex.Foundation.SlimScreen.Interfaces.IShowMessageScreen),
this.DefaultCultureInfo, BuildParamCollection(
    "MessageKey", "YOUR MESSAGE" + DLoc,
    "ShowButton", true
));
msg = WaitForMessage();
```

Result:



2.2.2. Image screen type

Parameters collection

Name	Type	Description
TitleKey	String	Title of the screen
ImagePath	String	Full path of the picture
MessageKey	String	Message under the screen
ShowButton	Bool	-

Example:

```
Message msg = null;
session.ShowScreen(typeof(Produmex.Foundation.SlimScreen.Interfaces.IShowImageScreen),
    this.DefaultCultureInfo, BuildParamCollection(
        "TitleKey", "Picture of the product",
        "MessageKey", "message under the picture",
        "ImagePath", "<PATH OF THE IMAGE>",
        "ShowButton", true
    ));
msg = WaitForMessage();
```

Result:



2.2.3. Enter String Value type

You can capture additional text information on this screen. The captured data can be get from the message object.

The value can be used in the Hookflow for further processing, or if the Hookflow script has an output parameter, then we can put the captured value into the output parameter.

Parameters

Name	Type	Description
InitialErrorKey	String	n.a. in custom usage
TitleKey	String	Title of the screen
Information	String	Additional information on the screen
Parameters	Object of sting	n.a. in custom usage
AllowToGoBack	Bool	-
ForceDataEntry	Bool	-
AllowMultiLine	Bool	-
MinimumNumberOfCharacters	Int	Minimum number of characters that must be typed

Example:

The entered text will be used on a message screen.

```
string initialErrorKey = null;
    string FreeText = "";

    Message msg = null;
session.ShowCustomizedScreen(typeof(Produmex.Foundation.SlimScreen.Interfaces.IEnterStringValueScreen),
    DefaultCultureInfo.Get(), BuildParamCollection(
        "InitialErrorKey", initialErrorKey,
        "TitleKey", "Title of the screen",
        "Information", "Information text",
        "Parameters", new object[] { "" },
        "AllowToGoBack", true,
        "ForceDataEntry", true,
        "AllowMultiLine", true,
        "MinimumNumberOfCharacters", 5
    ),
    WorkflowId,
nameof(PickingScript_Screens.EnterStringValueScreen1));
    msg = WaitForMessage();
    if (msg.Name.EndsWith(".StringEntered"))
    {
        FreeText = ExtractParameter<string>(msg.Parameters,
"stringValue");
    }

    msg = null;
session.ShowScreen(typeof(Produmex.Foundation.SlimScreen.Interfaces.IShowMessageScreen),
    this.DefaultCultureInfo, BuildParamCollection(
        "MessageKey", "Entered text: " + FreeText,
        "ShowButton", true
    ));
    msg = WaitForMessage();
```

Result:

2.2.4. Select Product Screen type

You can create an item list in a DataSet object to select an item from a list. The captured data can be get from the message object.

The value can be used in the Hookflow for further processing, or if the Hookflow script has an output parameter, then we can put the captured value into the output parameter.

Parameters

Name	Type	Description
InitialErrorKey	String	n.a. in custom usage
TitleKey	String	Title of the screen
Information	String	Additional information on the screen
Parameters	Object of sting	n.a. in custom usage
AllowToGoBack	Bool	-
ForceDataEntry	Bool	-
AllowMultiLine	Bool	-
MinimumNumberOfCharacters	Int	Minimum number of characters that must be typed

Example:

The selected item will be used on a message screen.

```
string initialErrorKey = null;
    string FreeText = "";
    Message msg = null;
    DataSet dsItems = null;

    string query = "SELECT DISTINCT PMX_OITMANAGED_BY_PMX.ItemCode
AS ProductCode, PMX_OITMANAGED_BY_PMX.U_PMX_CUDE AS ProductDescription,
PMX_OITMANAGED_BY_PMX.CodeBars AS GTIN,
PMX_OITMANAGED_BY_PMX.U_PMX_HBBD, PMX_OITMANAGED_BY_PMX.U_PMX_PILR,
PMX_OITMANAGED_BY_PMX.ManBtchNum, PMX_OITMANAGED_BY_PMX.U_PMX_LOUN,
PMX_OITMANAGED_BY_PMX.NumInBuy, PMX_OITMANAGED_BY_PMX.BuyUnitMsr,
PMX_OITMANAGED_BY_PMX.InvntryUom, PMX_OITMANAGED_BY_PMX.CodeBars AS
CodeBars, PMX_OITMANAGED_BY_PMX.ItemName FROM PMX_OITMANAGED_BY_PMX WITH
(NOLOCK) WHERE PMX_OITMANAGED_BY_PMX.InvntItem = 'Y' AND
PMX_OITMANAGED_BY_PMX.InvntItem = 'Y' AND NOT (
PMX_OITMANAGED_BY_PMX.frozenFor = 'Y' AND ( (
PMX_OITMANAGED_BY_PMX.frozenFrom IS NULL OR CURRENT_TIMESTAMP >=
PMX_OITMANAGED_BY_PMX.frozenFrom ) AND ( PMX_OITMANAGED_BY_PMX.frozenTo
IS NULL OR CURRENT_TIMESTAMP < DATEADD( day, 1,
PMX_OITMANAGED_BY_PMX.frozenTo ) ) ) ) ORDER BY ProductDescription";

    dsItems = sboProviderService.RunView(false, null, null, query);
session.ShowCustomizedScreen(typeof(Produmex.Foundation.SlimScreen.Interfaces.ISelectProductScreen),
    DefaultCultureInfo.Get(), BuildParamCollection(
    "InitialErrorKey", initialErrorKey,
    "TitleKey", "Title of the screen",
    "ProductDS", dsItems
    ),
WorkflowId,
    nameof(ChecksScript_Screens.SelectProductScreen1));
msg = WaitForMessage();

    if (msg.Name.EndsWith(".ProductSelected"))
    {
```

```

        FreeText = ExtractParameter<string>( msg.Parameters,
"itemCode" );
    }

    msg = null;
    session.ShowScreen(typeof(Produmex.Foundation.SlimScreen.Interfaces.IShowMessageScreen),
        this.DefaultCultureInfo, BuildParamCollection(
            "MessageKey", "Entered text: " + FreeText,
            "ShowButton", true
        ));
    msg = WaitForMessage();

```

Result:

2.2.5. Yes/No question Screen type

Parameters

Name,Type,Description TitleKey,String,Name of the screen MessageKey,String,question string

Example:

```

Message msg = null;
session.ShowCustomizedScreen(typeof(Produmex.Foundation.SlimScreen.Interface
s.IDecisionScreen),
DefaultCultureInfo.Get(), BuildParamCollection(
"TitleKey", "Title of the screen",
"MessageKey", "Do you want to continue?"),
WorkflowId,
nameof(ReceptionScript_Screens.DecisionScreen22));
msg = WaitForMessage();

if (msg.Name.EndsWith(".Yes"))
{
// goto Step_ClearDataBeforeNextItem;
}
if (msg.Name.EndsWith(".No"))
{
BackRequested.Set(true);
}

```

Last update: 2025/04/22 14:24 implementation:wms:wms_scripting_site:screen https://wiki.produmex.name/doku.php?id=implementation:wms:wms_scripting_site:screen

From: <https://wiki.produmex.name/> - **Produmex**

Permanent link: https://wiki.produmex.name/doku.php?id=implementation:wms:wms_scripting_site:screen

Last update: **2025/04/22 14:24**

