Customization Framework on Mobile Client

Overview

From product version 2023.06, users can start the scanner application of the **Mobile Client in customization mode** and you can customize all the workflows available. From product version 2024.06, users can now create more complex customized workflows in the customization mode by creating user queries and adjusting & filtering the options in the **Customization Manager.**

Users have the possibility to customize the buttons and the screens of the flow while different customization for user groups and users can be defined in a way that is optimal for the warehouse.

Videos on the customization mode are available:

- **Produmex WMS 2021.03 Demo: UI Framework**
- **Produmex WMS UI Framework**

Overview about the customization UIs:

Customization mode

The name of the parameter is /cust. When you start the Mobile Client in customization mode, a customization icon (a cog sign) is displayed on the top-right corner of the screen. If you click the icon, it becomes red and the customization mode is active.

To customize the flow, proceed as follows:

1. Click the customization icon.
2. If you want to customize the screen, you can click anywhere on the screen. If you want to customize a button, click on the given button to be customized.
3. Customize your screen on the displayed Customization form (see customization options below) and click Save.

Note: In order to see the changes the user needs to restart the Mobile Client with disabled/switched off customization mode.

Customization options

1. Customization options on the Customization Tab

User Group: If you select a specific user group, the customization applies to the users in the given user group.

User: By default, no user is selected. Instead of user groups, you can define a specific user to whom the customization applies. Enable the User option and use the drop-down menu to select a user.

Default Button: The Default Button section is active if you customize a button.

- If you select the Default Button option, the user does not need to tap the button when working with the flow because the system automatically proceeds with the button. Only one button can be set as the default button on a screen.
- If you use the Default Button option, you can also set a screen timeout in seconds. In this case the system displays the button for the user for the defined interval. Within this interval the user can tap another button or the flow proceeds with the default button.

Visible: By default, the visible option is enabled.

- If you disable the option, the screen is not displayed for the user during the flow.
- If you disable the option, the button is not displayed for the user even if it is set as a default button.

Note: Hiding buttons overrides customizations. 'User' settings override 'User Group' settings and specific 'User Group' settings override 'All Users' settings.

Example: The customization applies to the Inventory user group. The Order button is set as the default button on the Select a Filter screen. The screen is displayed for the user for two seconds and if the user doesn't tap another button, the system proceeds with the Order button.

Example: The GS1/EAN Barcode button is not visible for any group on the Select a Filter screen.

2. Customization options on the Events Tab

Load Event: In this field you can customize an existing event, write the preferable name in the field than push the save button. What we do here is selecting any of the events that are already in use and modifying it's action.

Load Event Name: In this field you can modify an existing event by giving a unique name for your new customized event, after saving the unique name open the Query Manager in B1 where you can add your query to the name. For example create a *“sales_return”* name in the Load Event field and open it up in the Query Manager for furthermore customization.

Load Event List: Under the Load Event field there are a list of the previous events and actions that you have already been through.

Manage: Pushing the manage button will show the customization manager screen.

Customization Manager

The **Customization Manager** screen is a helpful UI to manage your customizations in a visual way. On this screen you can find all of your user queries in a simple table. In the **Customization Manager** you can not change the database informations, you only have the option to filter/activate/inactivate/delete your added queries.

1. All Components

In the **All Components** part you can find a tree structure, the purpose of this navigation tree is to easily manage the rows where you added a new queries. The “golden arrow” shows the selected row in the tree structure. If you select the **All components** row than in the table on the right side will show every grid, every lines that are found below it in the system.

The main components in the tree structure:

- Mainflows
- Subflows

2. Filter

The **Filter** section contain all the filter options to search for a specific query.

Screen: By default, no screen is selected. In the screen from dropdown menu you can select the preferred screen.

User Group: If you select a specific user group, the customization applies to the users in the given user group.

Options:

- All Users
- Finance
- Inventory
- Purchase
- Sales

User: By default, no user is selected. Instead of user groups, you can define a specific user to whom the customization applies. Enable the User option and use the drop-down menu to select a user.

Show: This options is a dropdown menu where you can choose between several options to search a specific group of queries e.g. if you would like to check on the all of your inactive queries.

Options:

- Everything
- All Active
- All Inactive
- Active Visible
- Active Invisible

Customization: By clicking the options inside of the Customization aggregation you can easily set a quick filter and search for a group of queries.

Options:

- Screen
- Controls
- User Quiers

Full Workflow: Clicking on the Full Workflow will extend the Workflow column with extra information about the path of the flows.

Reset Filters: With this button you can clear the filters that you previously set.

Customization Examples

1. Limitation:
On the **Events tab** you can see your previous steps/actions listed under the **Load Event** field. In that list you can clearly follow all your steps since you opened the mobile client in customization mode. Be aware you will not find those kind of actions listed when you selected a value from a list by a manual click, for example you chose a product by clicking it's name from the list instead of scanning the item barcode.

The selected values will only appear in the action list when you manually entered the

value (customercode, location etc.) into the field or scanned that value.

2. Limitation:
If a user query are no longer used there is a specific procedure to remove the query from the system. First you have to delete the query from the **Customization Manager**, after that action open the **Query Manager** in SBO and remove the unused query from that table.

Example 1. - Default customer for sales return

Insert the <*customercode*> to the input field, and after you added the query to the **Query Manager**, the query automatically will select the predefined customer then waits 1 second and clicks the forward button.

Load Event Name: sales_return

MSSQL:

```
SELECT '<customercode>' as "txtCustomerCode", 'btnForward' as
"DefaultButton", 1 as "DefaultButtonClickTimeout"
```

HANA:

```
SELECT '<customercode>' as "txtCustomerCode", 'btnForward' as
"DefaultButton", 1 as "DefaultButtonClickTimeout" FROM DUMMY
```

Implementation: Add the query in the Query Manager in SBO.

Example 2. - Validations of the default value

Validation of the input quantity value on the reception flow, in this example we are showing as default quantity the minor between still to receive and default quantity logistic unit.

Load Event Name: default_quantity

Implementation: Add the query in the Query Manager in SBO.

MSSQL:

```
SELECT
CASE
  WHEN CAST(LEFT('$[lblQuantity]', CHARINDEX(' ', '$[lblQuantity]')-1) AS
INT) < U_PMX_DQLU THEN LEFT('$[lblQuantity]', CHARINDEX(' ',
'$[lblQuantity]')-1)
  ELSE U_PMX_DQLU
END AS "edtCounter0"
FROM "OITM" WHERE U_PMX_CUDE = '$[lblItem]'
```

HANA:

```
SELECT
CASE
WHEN CAST(LEFT('$[lblQuantity]', LOCATE( '$[lblQuantity]',' ')-1) AS
INTEGER) < "U_PMX_DQLU" THEN LEFT('$[lblQuantity]', LOCATE(
'$[lblQuantity]',' ')-1)
ELSE "U_PMX_DQLU"
END AS "edtCounter0"
FROM "OITM" WHERE "U_PMX_CUDE" = '$[lblItem]'
```

Example 3. - Finding a Pick List connected to a default customer

In this example if the query finds a Pick List that is connected to the <*customercode*> then the query will select the first Pick List then clicking on the forward button, if the query will not find a Pick List then nothing will happen.

Load Event Name: finding_picklist_connected_default_customer

Implementation: Add the query in the Query Manager in SBO.

MSSQL & HANA:

```
SELECT TOP 1 "DocEntry" AS "txtPickList",
CASE WHEN "PMX_PLHE"."DocEntry" IS NOT NULL THEN 'btnForward'
ELSE ''
END AS "DefaultButton",
CASE WHEN "PMX_PLHE"."DocEntry" IS NOT NULL THEN '1'
ELSE ''
END AS "DefaultButtonClickTimeout"
FROM "PMX_PLHE" WHERE "CardCode" = '<customercode>'
ORDER BY "DocEntry" ASC
```

Example 4. - Sorting and/or Truncating in "Select from a List" Screen Types

On **“Select from a List”** screen types, users can apply custom logic to truncate and/or sort the displayed data.

In this example, we use a custom query to filter specific items during the purchase order process. Only those specific items will be displayed on the **“Select a Product”** screen in the customized purchase order.

Next Event Name: PurchaseProductListFilter

Implementation: Add the query in the Query Manager in SBO.

MSSQL & HANA:

```
SELECT
  'ProductCode in (' +  STRING_AGG(CONCAT('''', "ItemCode", ''''), ', ') +
')' as "RowFilter",
  'ProductCode desc' as "Sort"
FROM
  OITM
WHERE
  "ItemCode" like 'I%'
```

Result:
In SBO, open the Query Manager to check the results after executing the query. The values in the RowFilter column will be filtered in the dataset and sorted by the Sort column during the customized process.

We can define filters and sortings with the ListColumns values.

ListColumns from the Dataset that can be used:

```
-- $[ListColumns]=ProductCode, ProductDescription, GTIN, U_PMX_HBBD,
U_PMX_PILR, ManBtchNum, U_PMX_LOUN, NumInBuy, BuyUnitMsr, InvntryUom,
CodeBars, ItemName
```

Example 5. - Validating Inserted Values on "Forward" Click

Use a query within the Customization Framework solution to validate inserted values as you move forward in the flow.

In this example, our warehouse has a predefined location fixed for a specific item. During an AdHoc Move, this item can only be moved to its fixed location and cannot be relocated elsewhere. To ensure this, we must validate both the item and the location, that validation can be customized with a query.

If an attempt is made to move the item to a non-fixed location, the system will warn us that the **“Location is not fixed for this item!”**.

In the OSE, set the location to fixed for the specific item.

Next Event Name: AHMoveValidateEnteredDestinationLocation

Implementation: Add the query in the Query Manager in SBO

MSSQL:

```
SELECT
    CASE
        WHEN '$[Prev1.lblItem]' = ''
        THEN ''
        ELSE
            CASE
                WHEN
                    (
                        select COUNT(*) from PMX_SLIT left join
PMX_OITM_MANAGED_BY_PMX on PMX_SLIT."ItemCode" =
PMX_OITM_MANAGED_BY_PMX."ItemCode"
where PMX_SLIT."ItemCode" + ' - ' + PMX_OITM_MANAGED_BY_PMX."ItemCode" =
'$[Prev1.lblItem]' and "ParentCode" = '$[txtStringValue]'
                    ) = 0
                THEN 'Location is not fixed for this item!'
                ELSE ''
        END
    END AS "ValidationError"
```

HANA:

```
SELECT
    CASE
        WHEN '$[Prev1.lblItem]' = ''
        THEN ''
        ELSE
            CASE
                WHEN
                    (
                        select COUNT(*) from PMX_SLIT left join
PMX_OITM_MANAGED_BY_PMX on PMX_SLIT."ItemCode" =
PMX_OITM_MANAGED_BY_PMX."ItemCode"
where PMX_SLIT."ItemCode" + ' - ' + PMX_OITM_MANAGED_BY_PMX."ItemCode" =
'$[Prev1.lblItem]' and "ParentCode" = '$[txtStringValue]'
```

```
                    ) = 0
                THEN 'Location is not fixed for this item!'
                ELSE ''
        END
    END AS "ValidationError"
FROM DUMMY
```

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

Permanent link:
https://wiki.produmex.name/doku.php?id=implementation:wms:confguide_main:mobile_client:customization_mode

Last update: **2025/07/23 09:00**