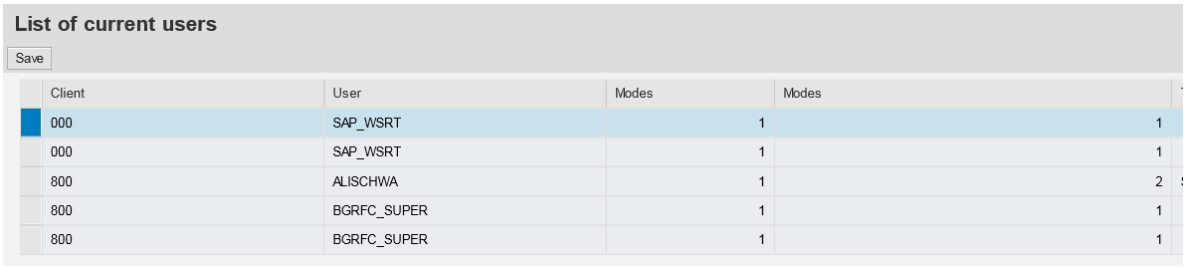
Version 1.0
Date 15.07.2016
Status In Development
Number 6
Initial version by Igor Muntoreanu

Exercise Review:

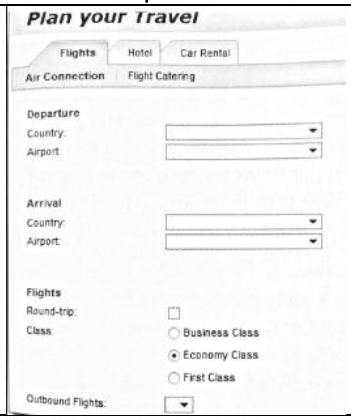
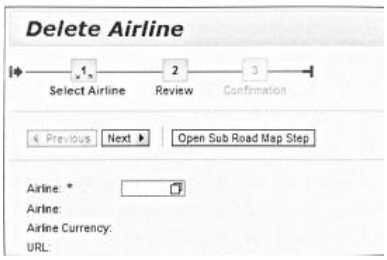
Introduction to FPM, OIF and IDR.

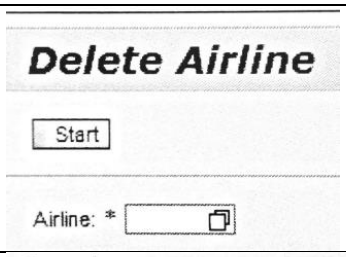
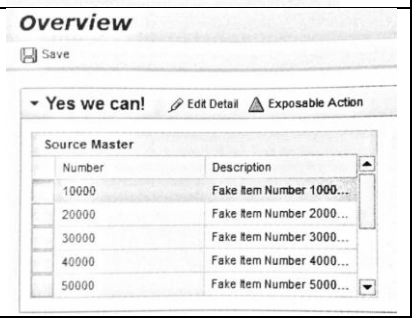


Practical Guide:

What is FPM?

Floorplan Manager (FPM) is a Web Dynpro ABAP application that provides a framework for developing new Web Dynpro ABAP application interfaces consistent with SAP UI guidelines (Standard definition from SAP doc). You can use the Floorplan Manager Configuration editor to combine application-specific views of one or more business applications to a new FloorPlan Manager application. Floorplan manager supports creating and configuring applications with following floorplans:

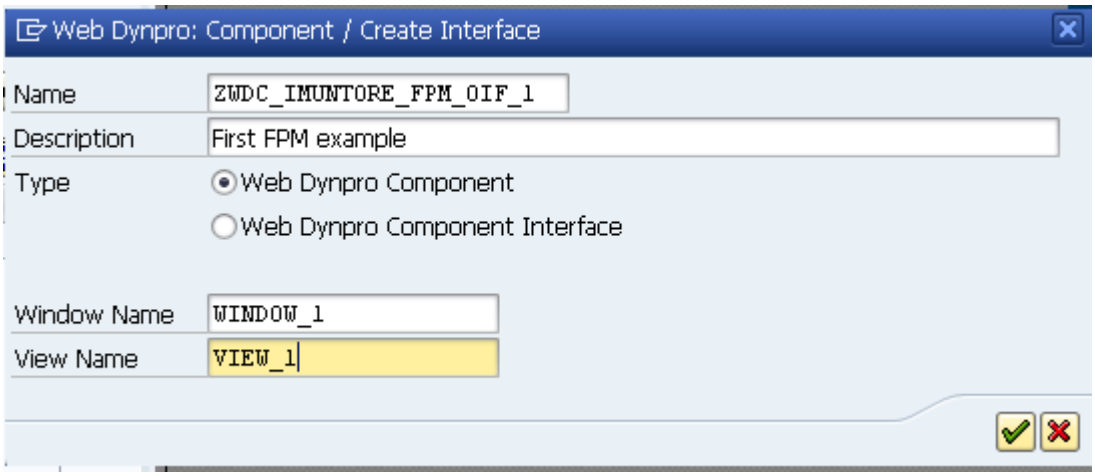
Floorplan Name	Component	Description	Picture Example
Object Instance Floorplan (**OIF**)	FPM_OIF_COMPONENT	Provides a template for displaying and manipulating the data, attributes, and associations of business objects within your system.	
Guided Activity Floorplan(**GAF**)	FPM_GAF_COMPONENT	Provides a template for conducting a user through a step-by-step activity.	

Quick Activity Floorplan(QAF)	FPM_OIF_COMPONENT	Provides a template do display or manipulate specific data in a business object.	
Overview Page Floorplan(OVP)	FPM_OVP_COMPONENT	Provides a template for creating overview pages that display and link to information or tasks about give business objects or scenarios.	

Using the FPM configuration editor following floorplan areas can be configured:

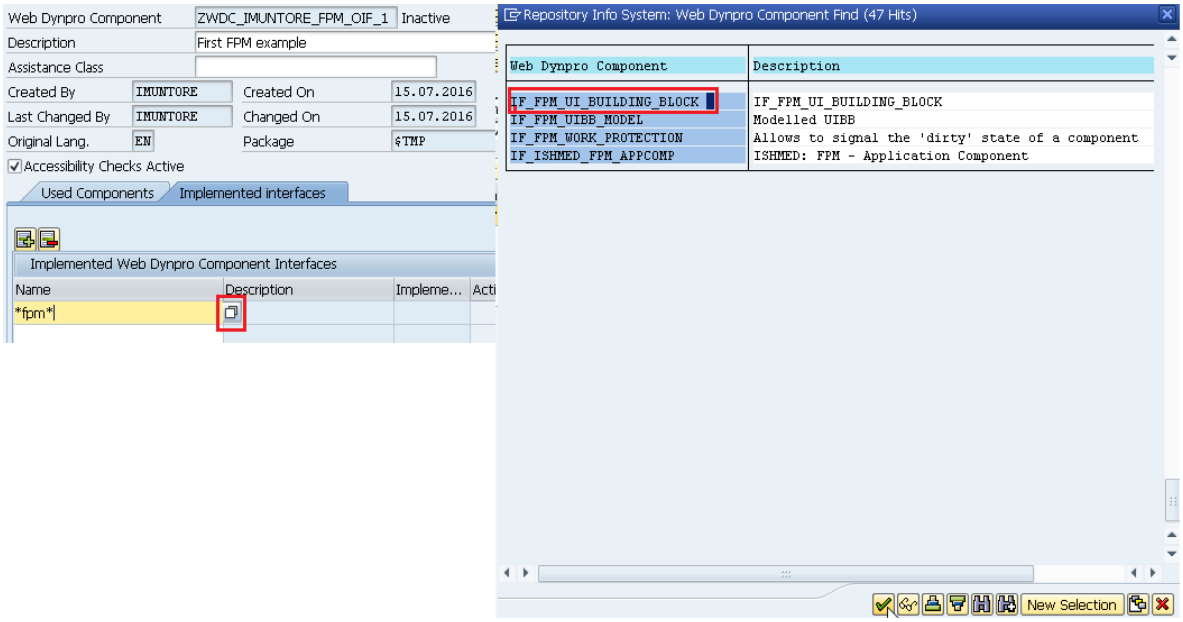
- Identification Region (IDR)
- Message Region (MR)
- Context Navigation Region (CNR)
- Roadmap Element

1) Create a new WD Component:

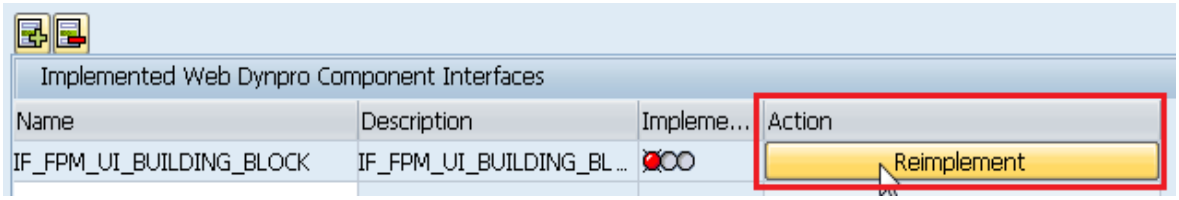


2) Go to the Component in the Implemented Interfaces:

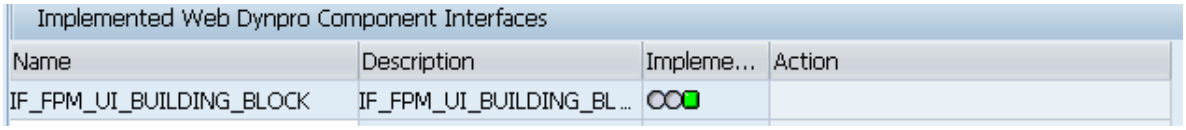
Select the following FPM Component Interface:



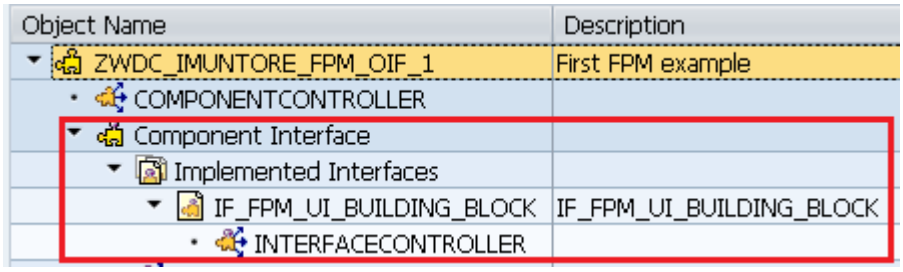
Re-implement it:



Result:

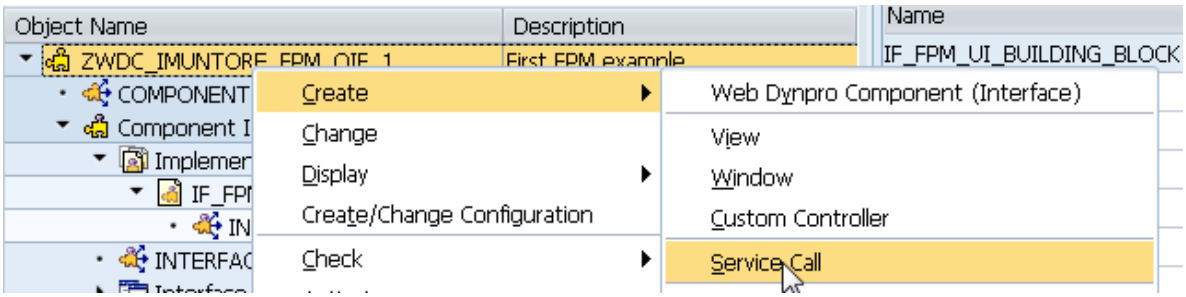


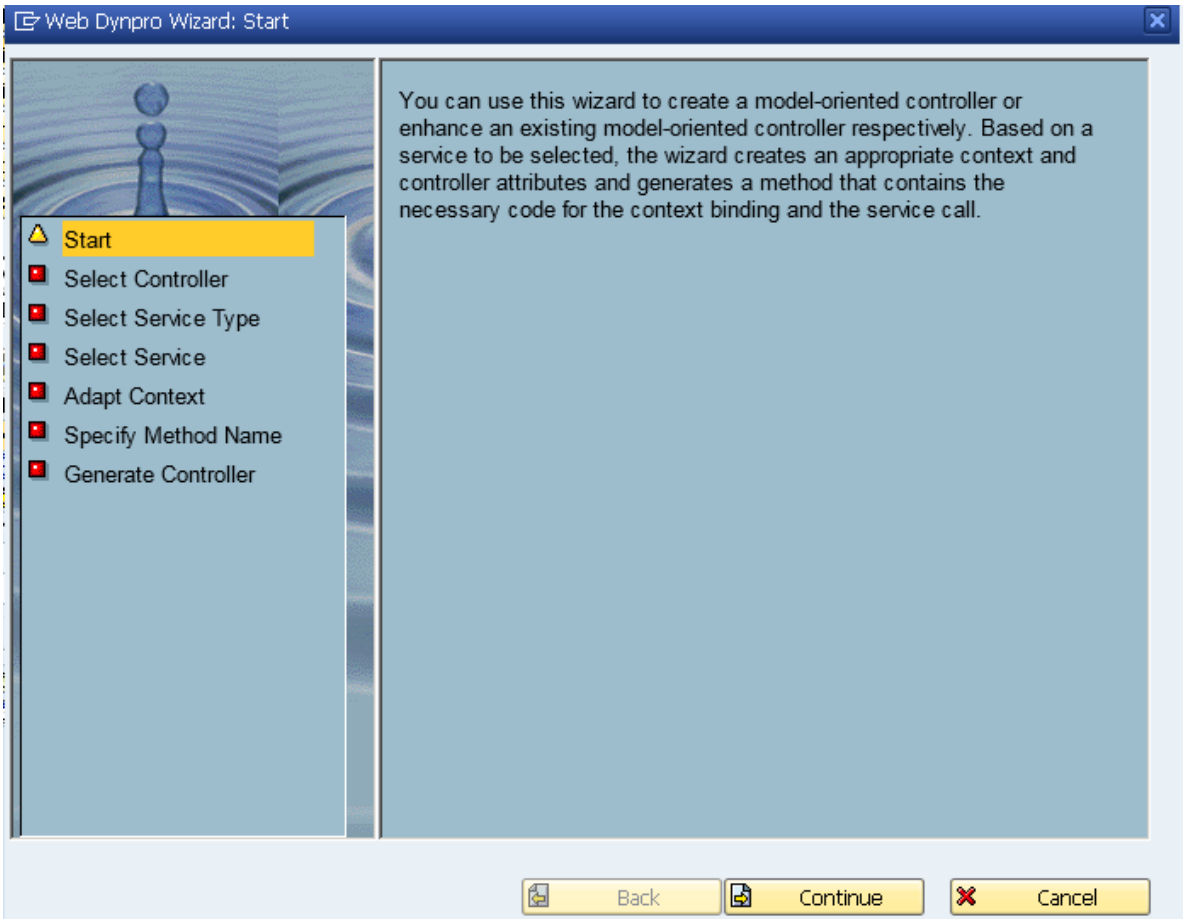
The following entries are now generated in the tree:

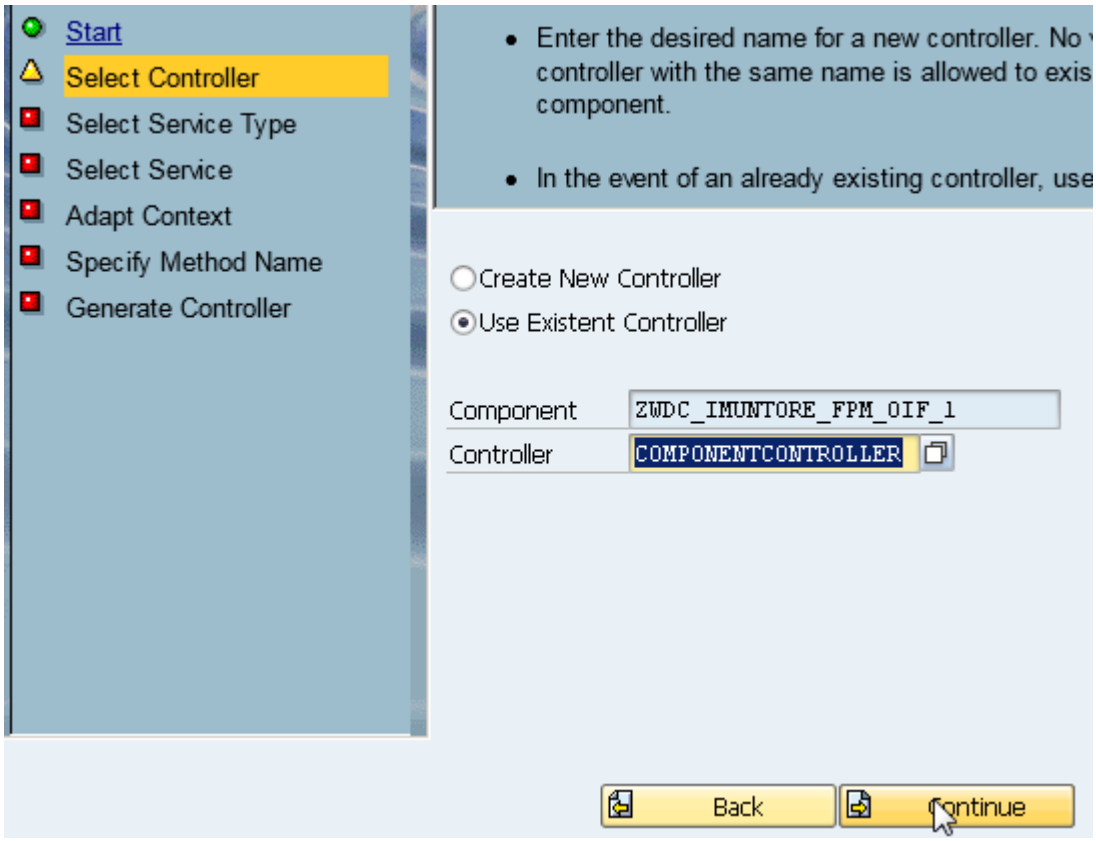


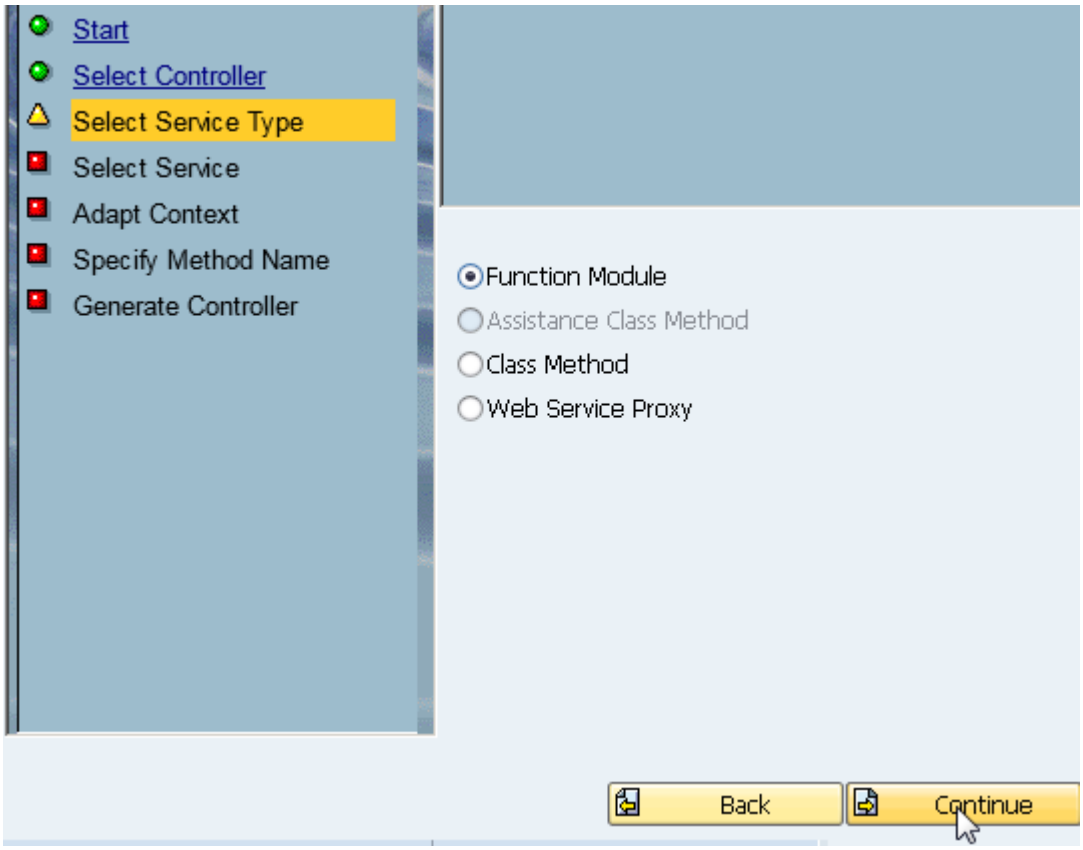
3) Create a Service Call

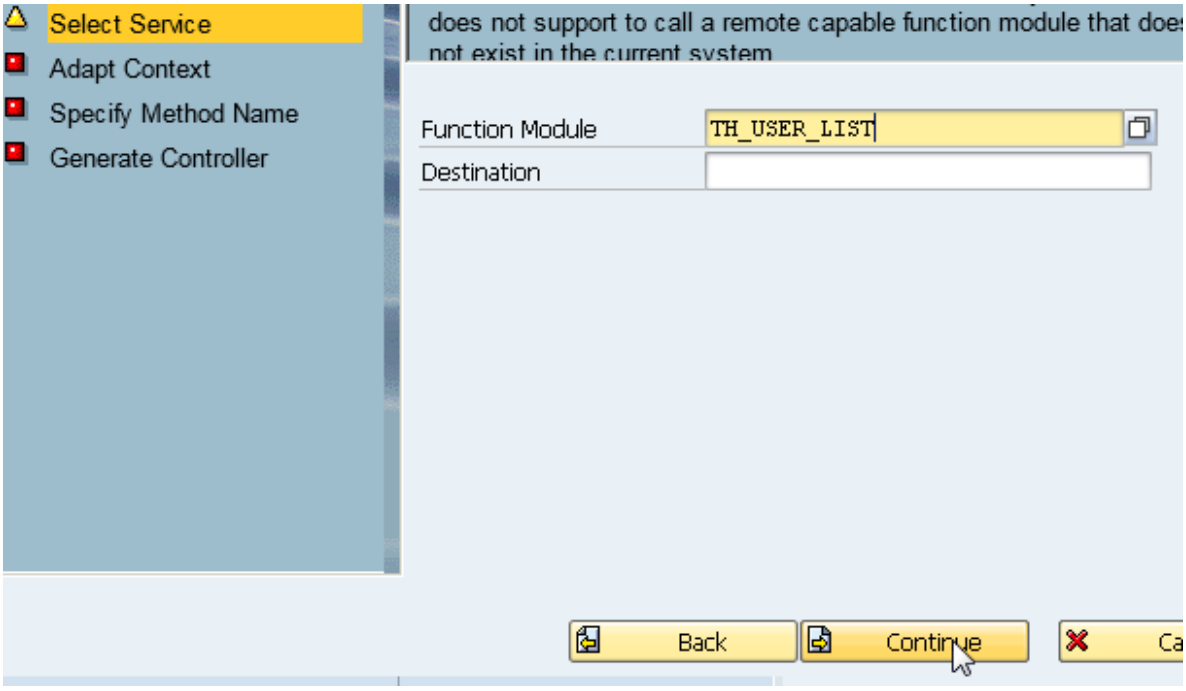
At the moment, we have no logic created. The idea is to bring a list of users that are at the moment in the system connected. We will create here a Service Call

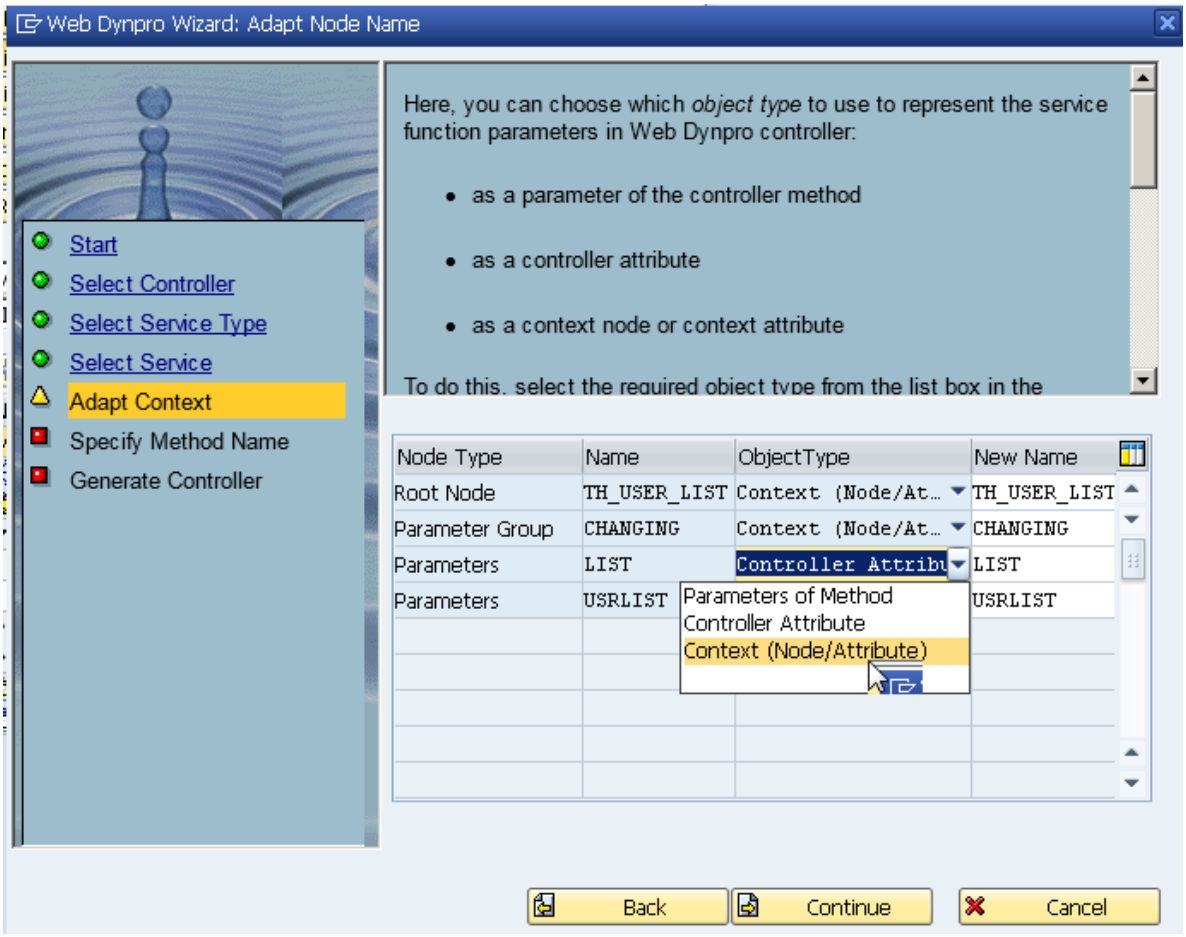

Object Name
Description
Name
IF_FPM_UI_BUILDING_BLOCK
COMPONENT
Component I
Implemer
IF_FPI
IN
INTERFAC
Create
Change
Display
Create/Change Configuration
Check
Web Dynpro Component (Interface)
View
Window
Custom Controller
Service Call

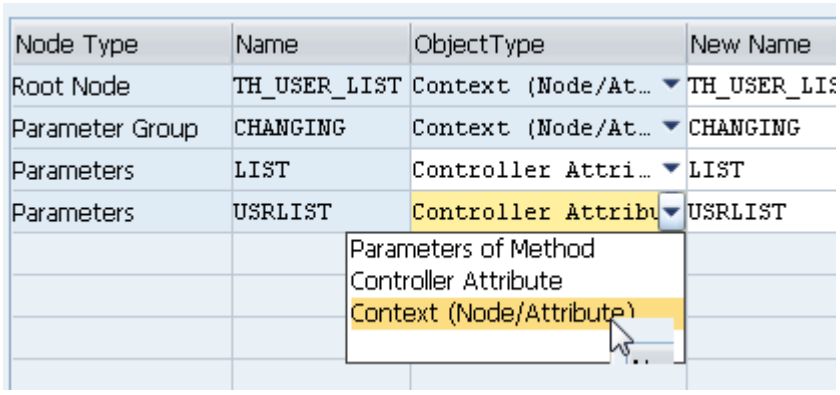

Web Dynpro Wizard: Start
Start
Select Controller
Select Service Type
Select Service
Adapt Context
Specify Method Name
Generate Controller
You can use this wizard to create a model-oriented controller or enhance an existing model-oriented controller respectively. Based on a service to be selected, the wizard creates an appropriate context and controller attributes and generates a method that contains the necessary code for the context binding and the service call.
Back
Continue
Cancel

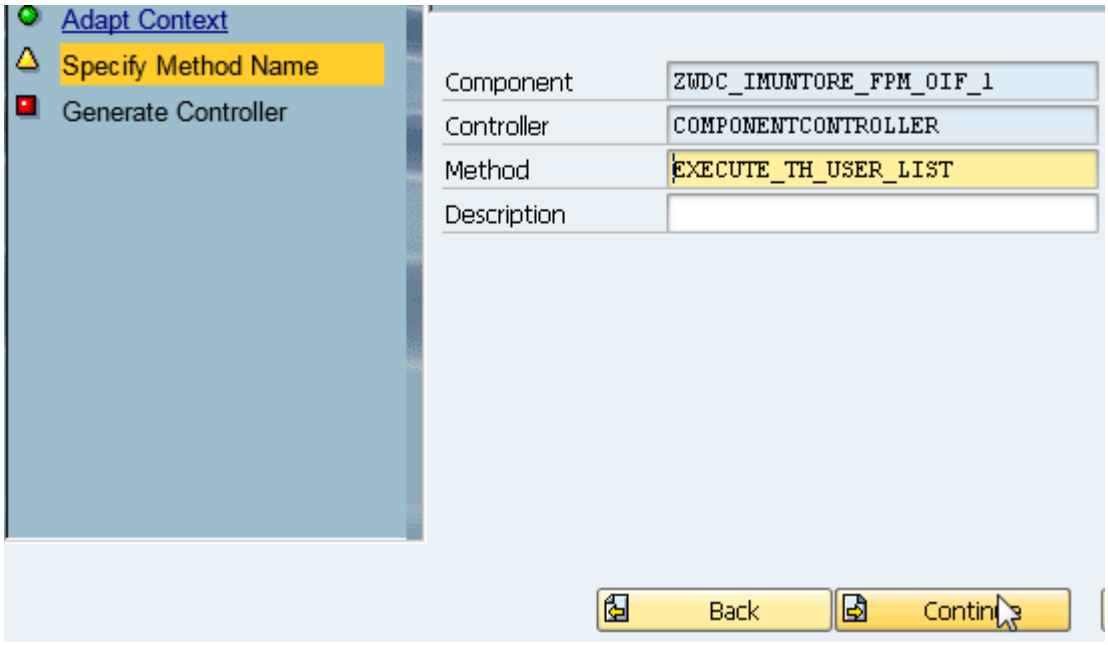

Start
Select Controller
Select Service Type
Select Service
Adapt Context
Specify Method Name
Generate Controller
• Enter the desired name for a new controller. No
controller with the same name is allowed to exis
component.
• In the event of an already existing controller, use
Create New Controller
Use Existent Controller
Component
ZWDC_IMUNTORE_FPM_OIF_1
Controller
COMPONENTCONTROLLER
Back
ntinue

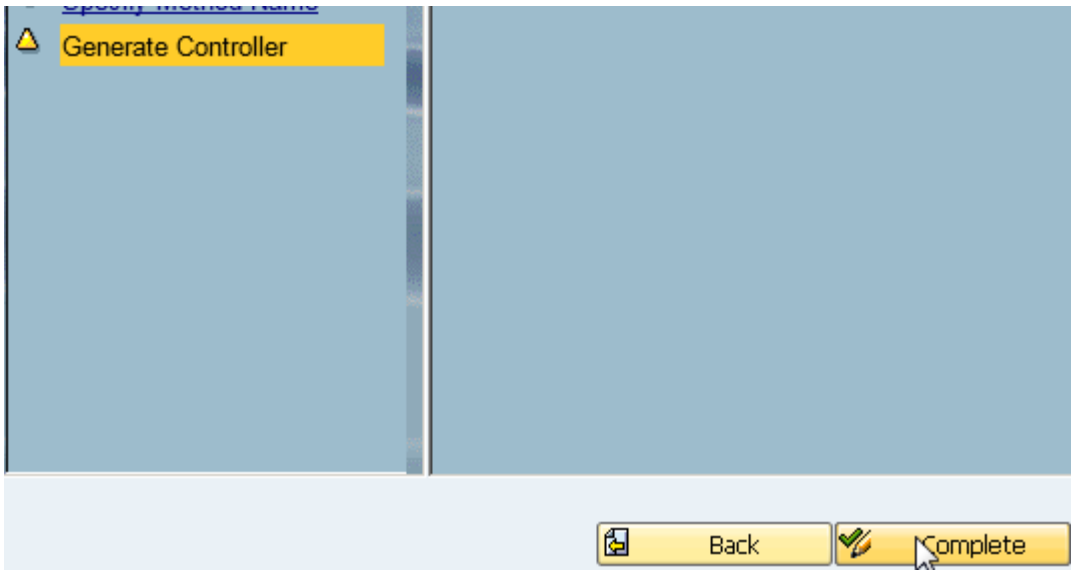

Start
Select Controller
Select Service Type
Select Service
Adapt Context
Specify Method Name
Generate Controller
Function Module
Assistance Class Method
Class Method
Web Service Proxy
Back
Continue


Select Service
Adapt Context
Specify Method Name
Generate Controller
does not support to call a remote capable function module that does
not exist in the current system
Function Module
TH_USER_LIST
Destination
Back
Continue


Web Dynpro Wizard: Adapt Node Name
Start
Select Controller
Select Service Type
Select Service
Adapt Context
Specify Method Name
Generate Controller
Here, you can choose which object type to use to represent the service function parameters in Web Dynpro controller:
as a parameter of the controller method
as a controller attribute
as a context node or context attribute
To do this, select the required object type from the list box in the
Node Type
Name
ObjectType
New Name
Root Node
TH_USER_LIST
Context (Node/At…
TH_USER_LIST
Parameter Group
CHANGING
Context (Node/At…
CHANGING
Parameters
LIST
Controller Attribu
LIST
Parameters
USRLIST
USRLIST
Parameters of Method
Controller Attribute
Context (Node/Attribute)
Back
Continue
Cancel


Node Type
Name
ObjectType
New Name
Root Node
TH_USER_LIST
Context (Node/At…
TH_USER_LIS
Parameter Group
CHANGING
Context (Node/At…
CHANGING
Parameters
LIST
Controller Attri…
LIST
Parameters
USRLIST
Controller Attribu
USRLIST
Parameters of Method
Controller Attribute
Context (Node/Attribute)

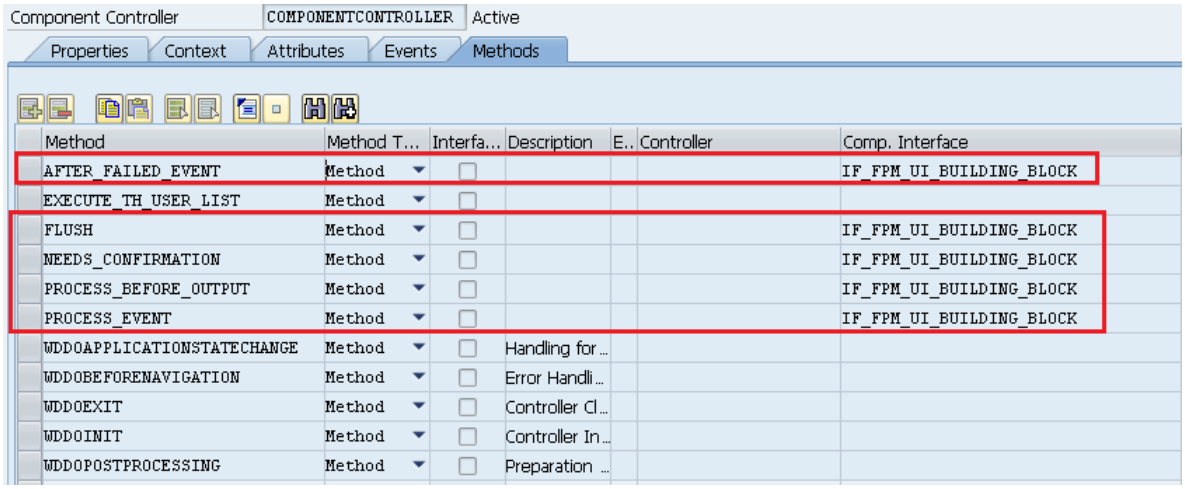




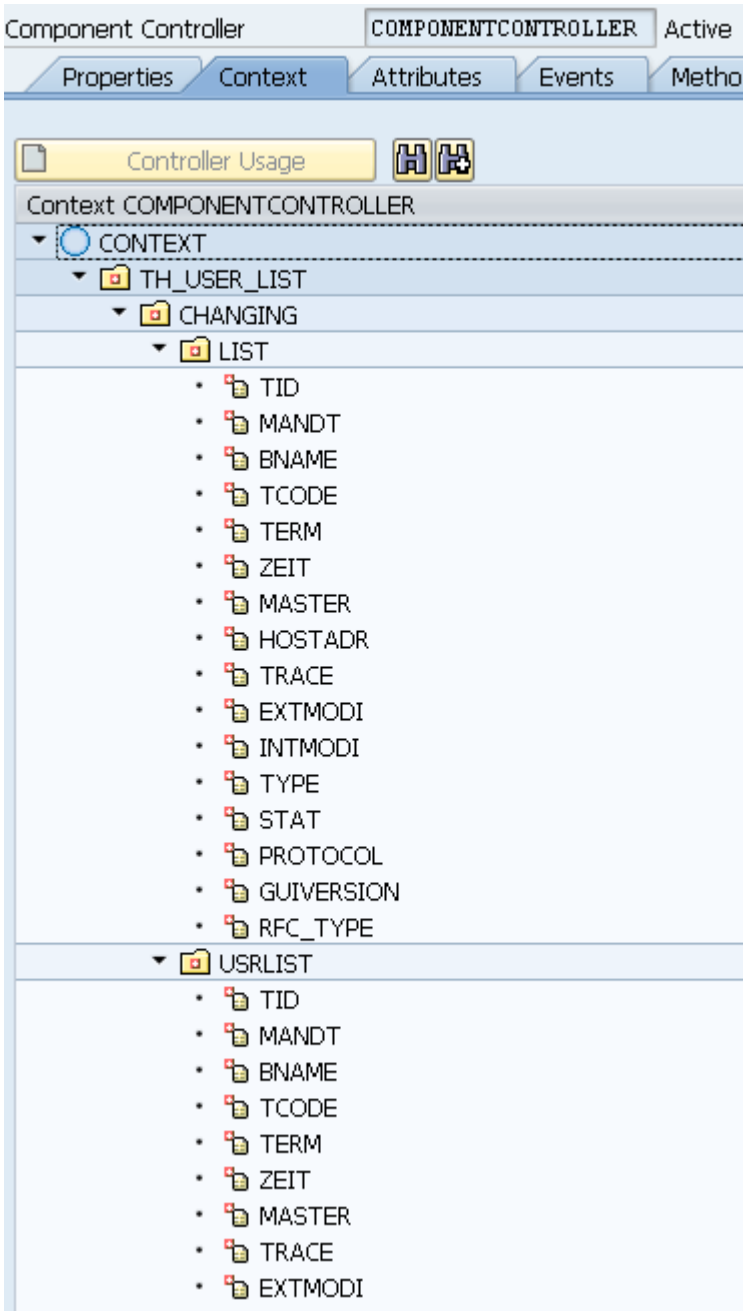
Save and activate everything.

4) Check the Context
Some methods was inherited from the FPM and the Context was brought from the Service

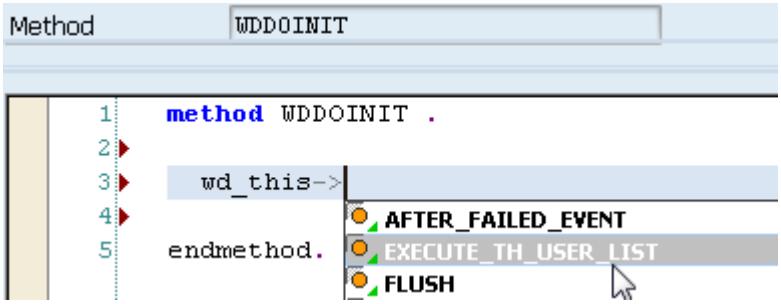
Call Wizard:

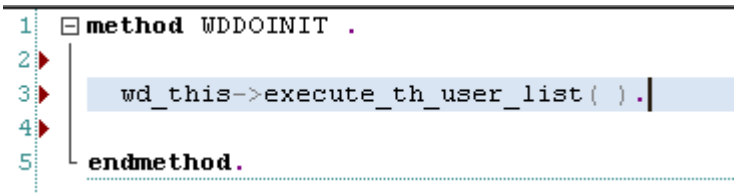


Method	Method T...	Interfa...	Description	E..	Controller	Comp. Interface
AFTER_FAILED_EVENT	Method	☐				IF_FPM_UI_BUILDING_BLOCK
EXECUTE_TH_USER_LIST	Method	☐				
FLUSH	Method	☐				IF_FPM_UI_BUILDING_BLOCK
NEEDS_CONFIRMATION	Method	☐				IF_FPM_UI_BUILDING_BLOCK
PROCESS_BEFORE_OUTPUT	Method	☐				IF_FPM_UI_BUILDING_BLOCK
PROCESS_EVENT	Method	☐				IF_FPM_UI_BUILDING_BLOCK
WDDOAPPLICATIONSTATECHANGE	Method	☐	Handling for ...			
WDDOBEFORENAVIGATION	Method	☐	Error Handli...			
WDDOEXIT	Method	☐	Controller Cl...			
WDDOINIT	Method	☐	Controller In...			
WDDOPOSTPROCESSING	Method	☐	Preparation ...			



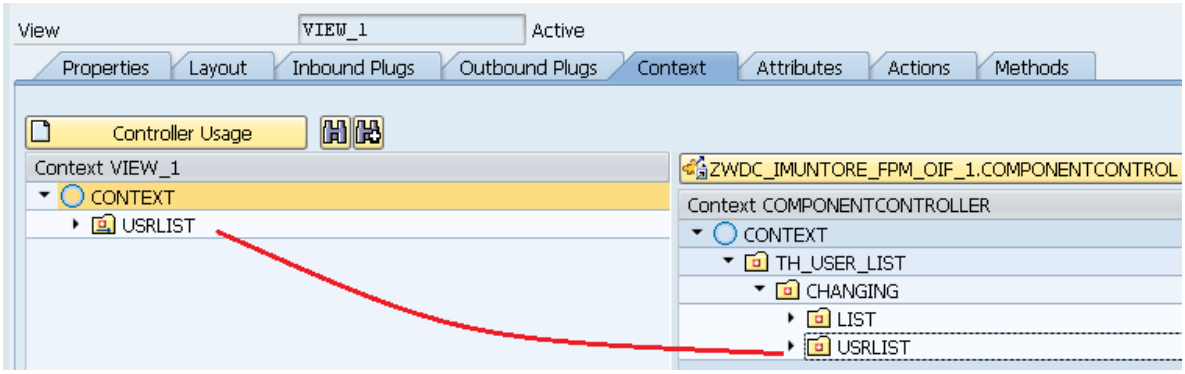
5) Creating and Changing Methods
Navigate to the INIT method and code:



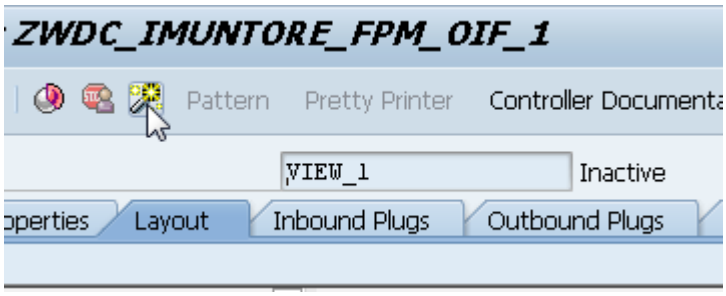


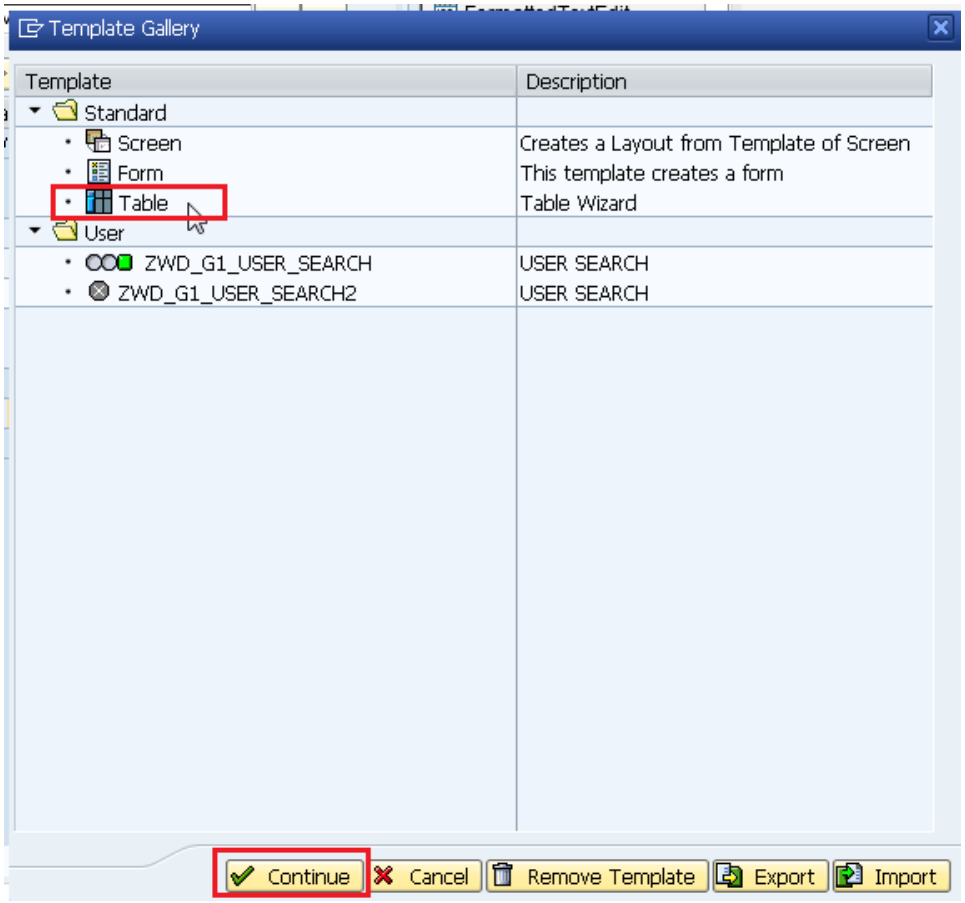
6) View Manipulation

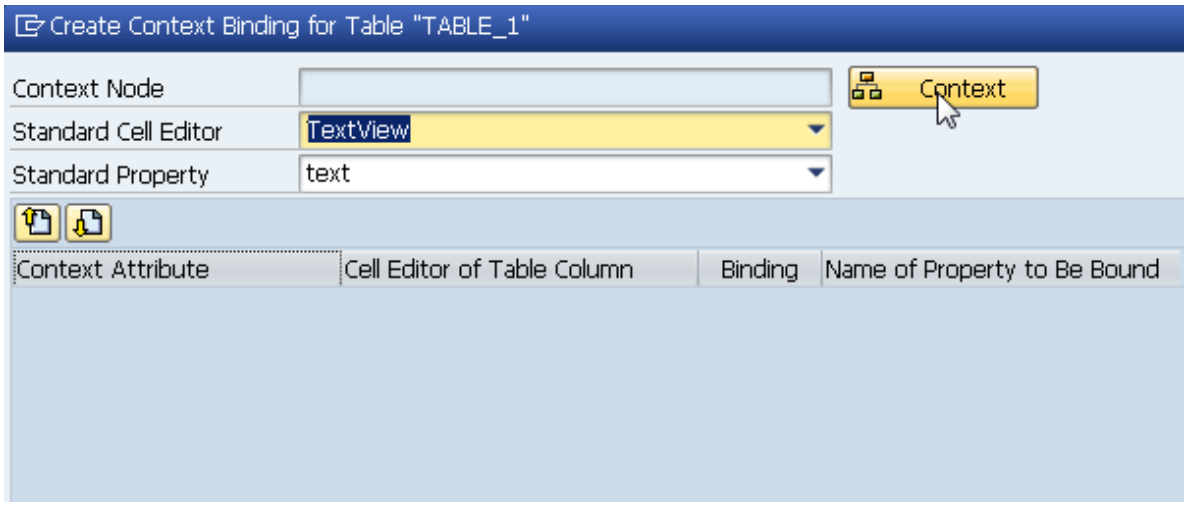
Navigate to the view and bind the following in the Context tab:

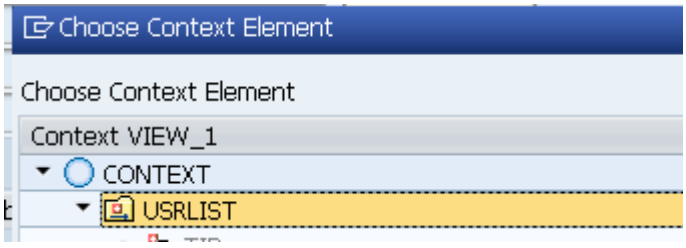


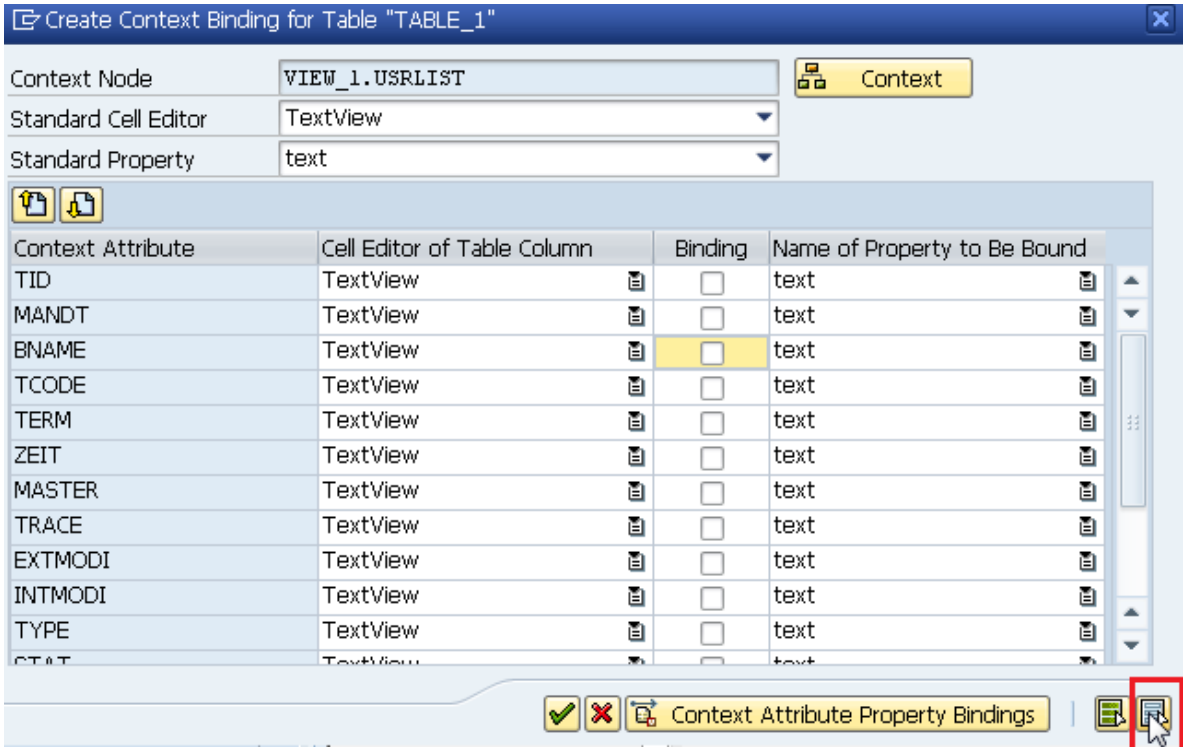
Go to the layout tab and click on the Wizard

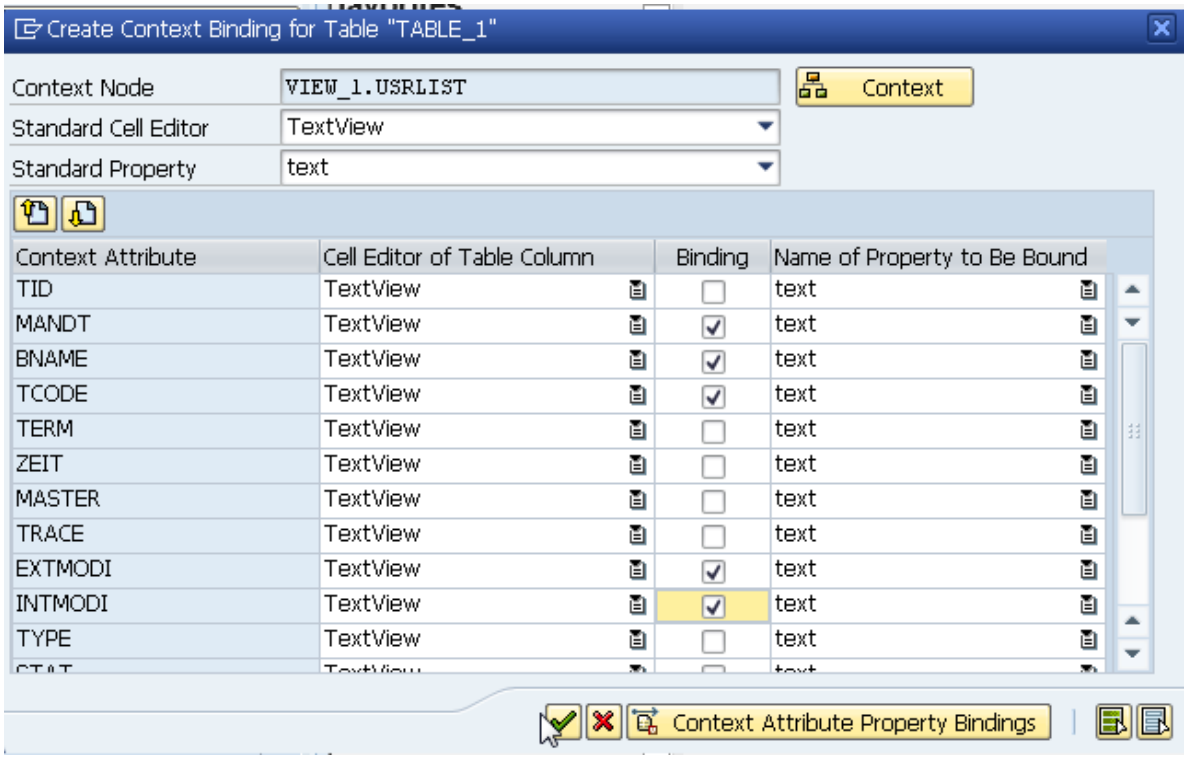

ZWDC_IMUNTORE_FPM_OIF_1
Pattern
Pretty Printer
Controller Documenta
VIEW_1
Inactive
operties
Layout
Inbound Plugs
Outbound Plugs


Template Gallery
Template
Description
Standard
Screen
Creates a Layout from Template of Screen
Form
This template creates a form
Table
Table Wizard
User
ZWD_G1_USER_SEARCH
USER SEARCH
ZWD_G1_USER_SEARCH2
USER SEARCH
Continue
Cancel
Remove Template
Export
Import

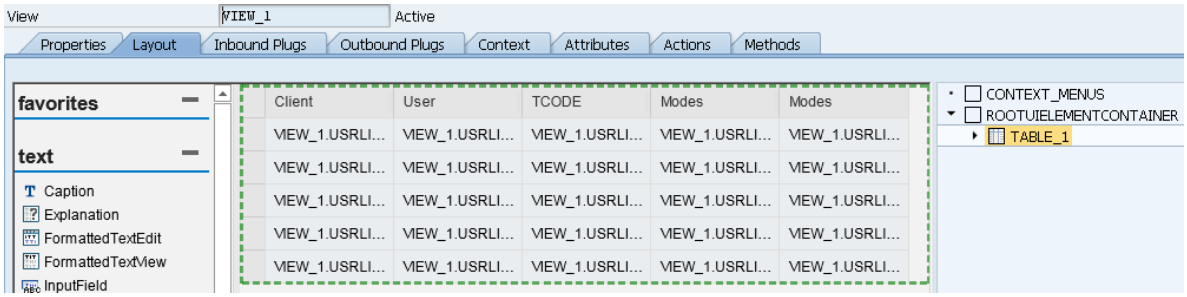

Create Context Binding for Table "TABLE_1"
Context Node
Context
Standard Cell Editor
TextView
Standard Property
text
Context Attribute
Cell Editor of Table Column
Binding
Name of Property to Be Bound


Choose Context Element
Choose Context Element
Context VIEW_1
CONTEXT
USRLIST

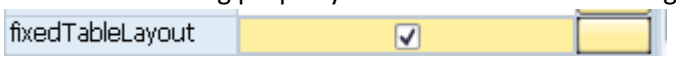

Create Context Binding for Table "TABLE_1"
Context Node
VIEW_1.USRLIST
Context
Standard Cell Editor
TextView
Standard Property
text
Context Attribute
Cell Editor of Table Column
Binding
Name of Property to Be Bound
TID
MANDT
BNAME
TCODE
TERM
ZEIT
MASTER
TRACE
EXTMODI
INTMODI
TYPE
Context Attribute Property Bindings



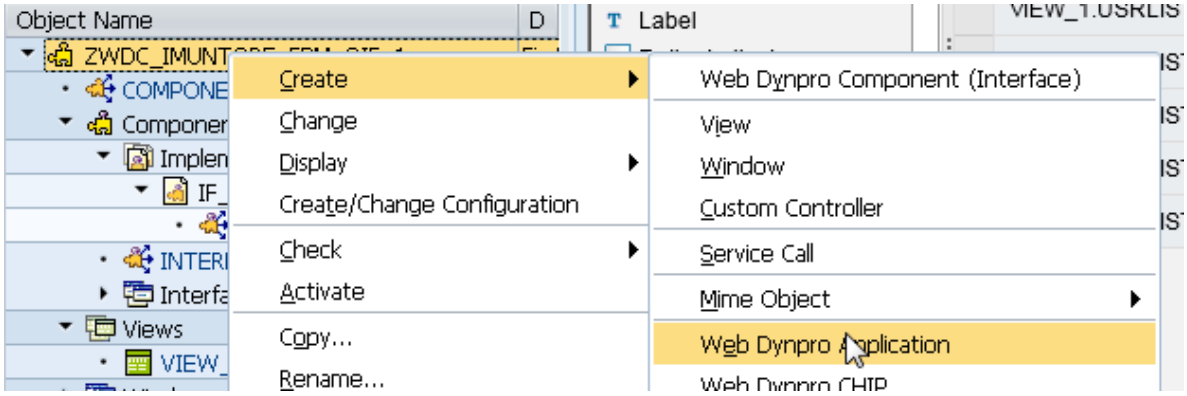
Result:

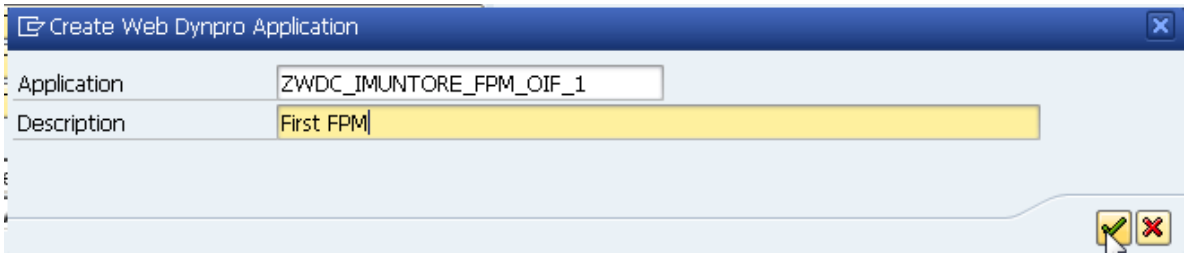


Mark the following property to avoid reallocation during the table scroll down:

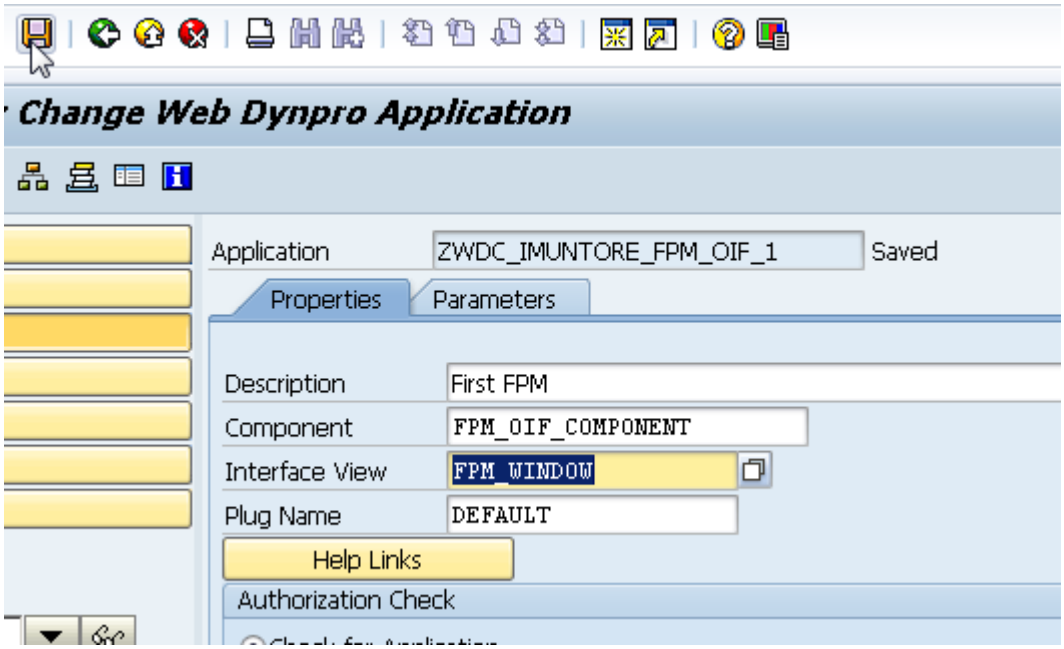


7) Create the WDA:



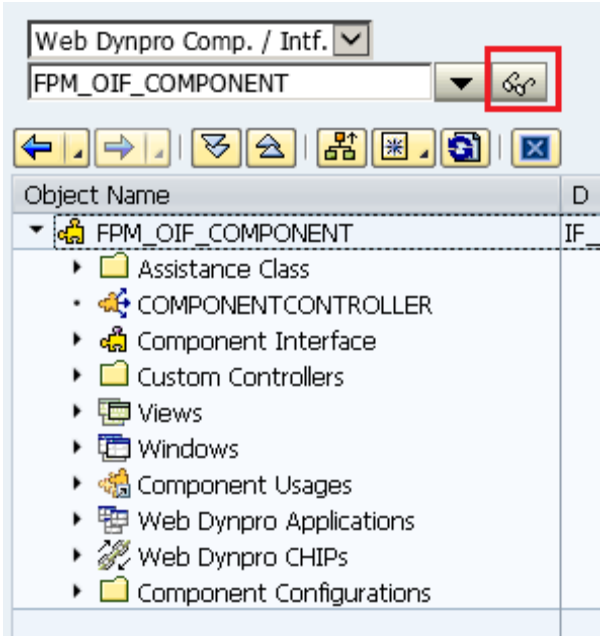


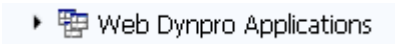
Change the Component as shown below:



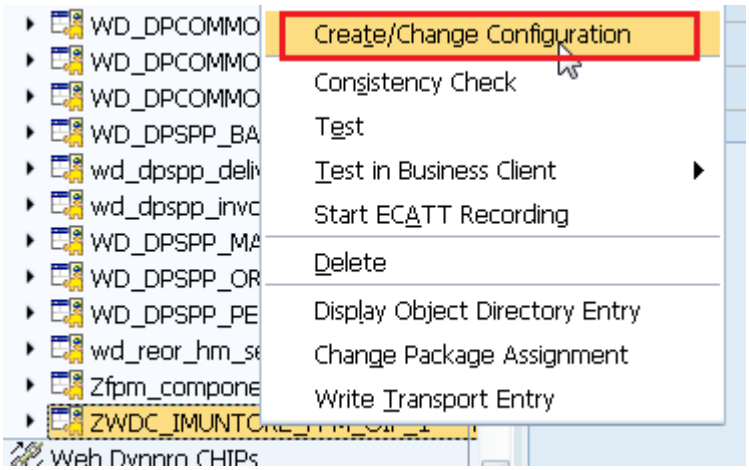
8) Save and Activate everything.

9) Check the FPM in se80 and do the Application and the Components Configuration



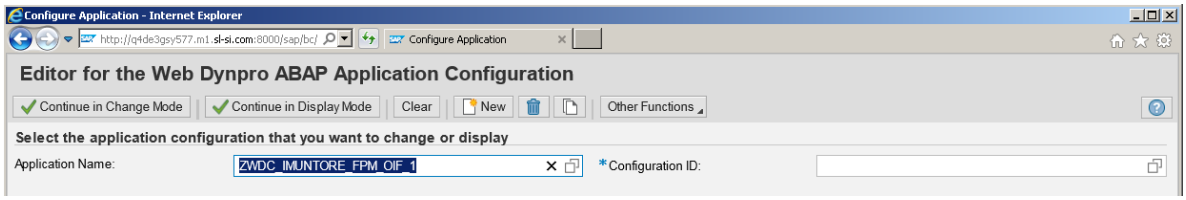
Expand the 

Create Configuration for our Application

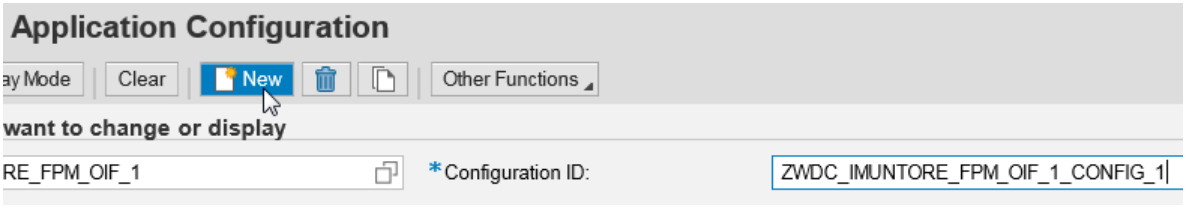


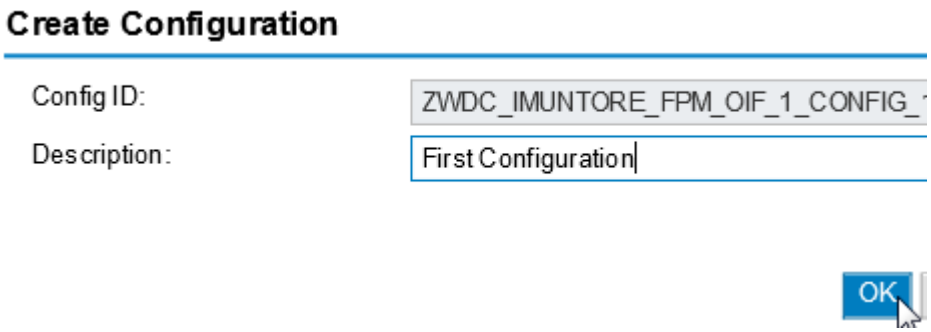
Now here begin the FPM configuration to our Application. It is a little tiresome but when you get used to it, it will become easy to understand:

A webpage will open:

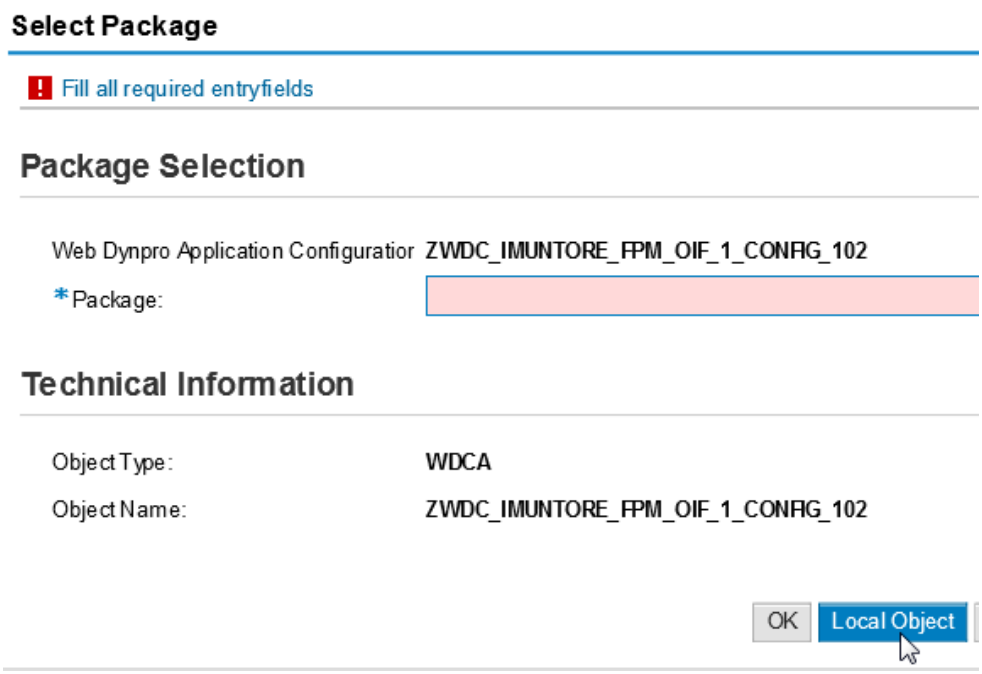


Create, first, the configuration as shown below for the Application:

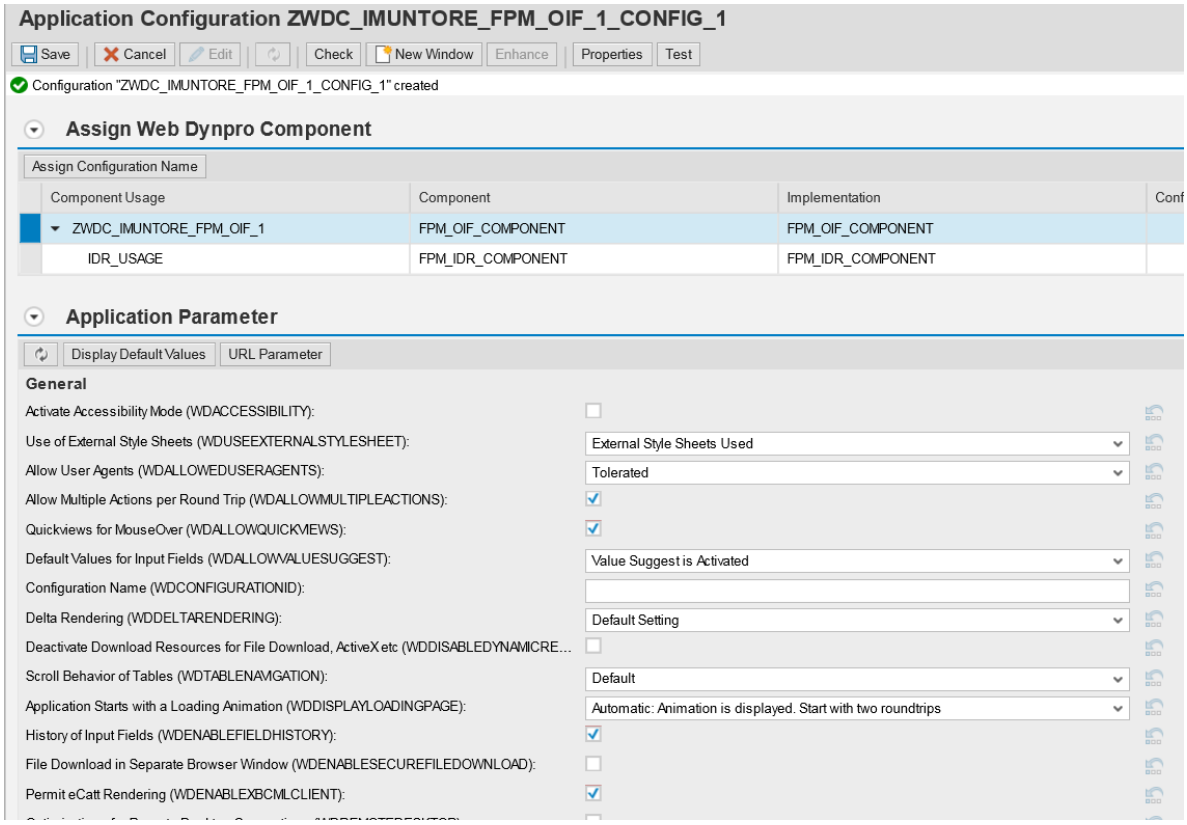




Create as a local object:

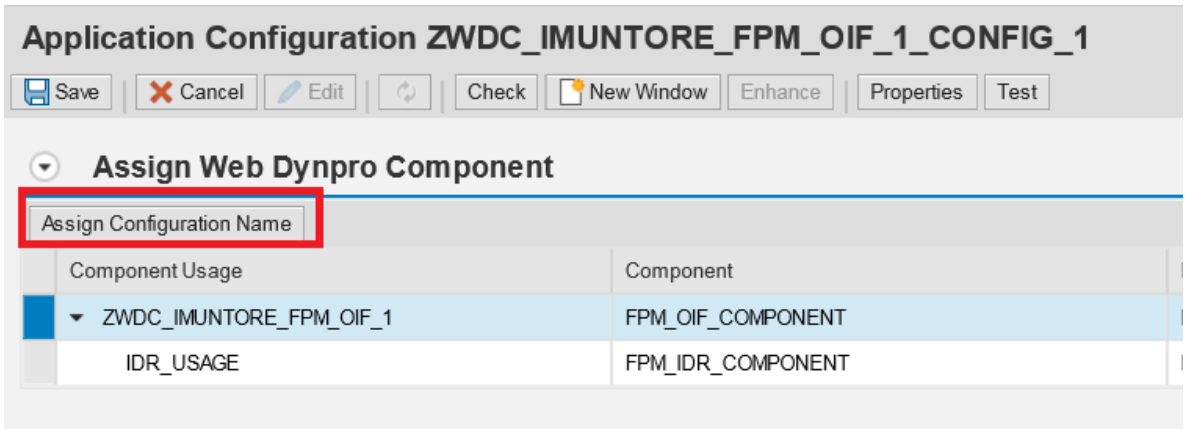


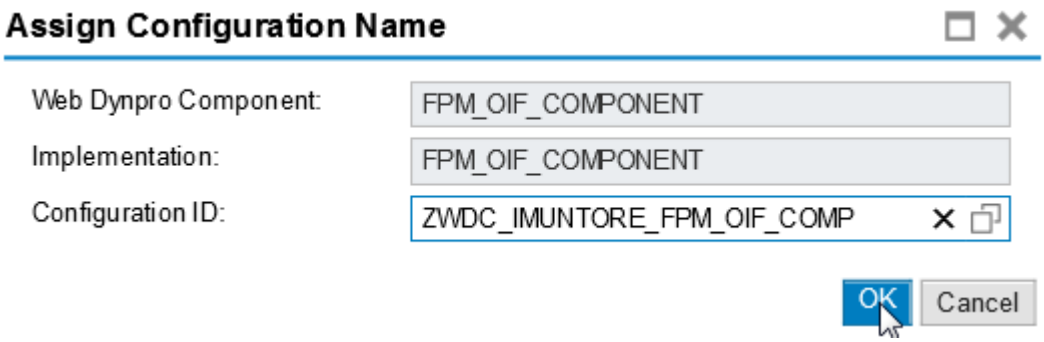
The following screen will be shown:



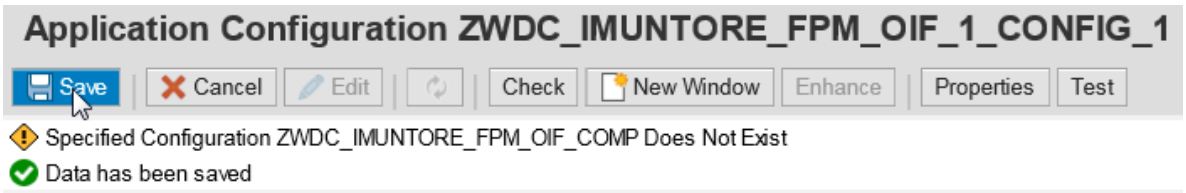
Save it:

Assign the Component to the Configuration ID:

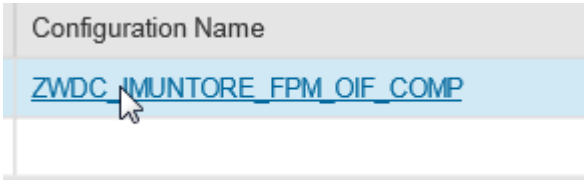


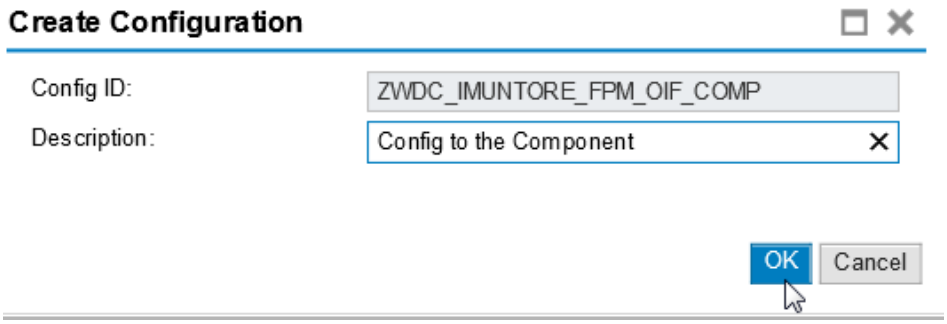


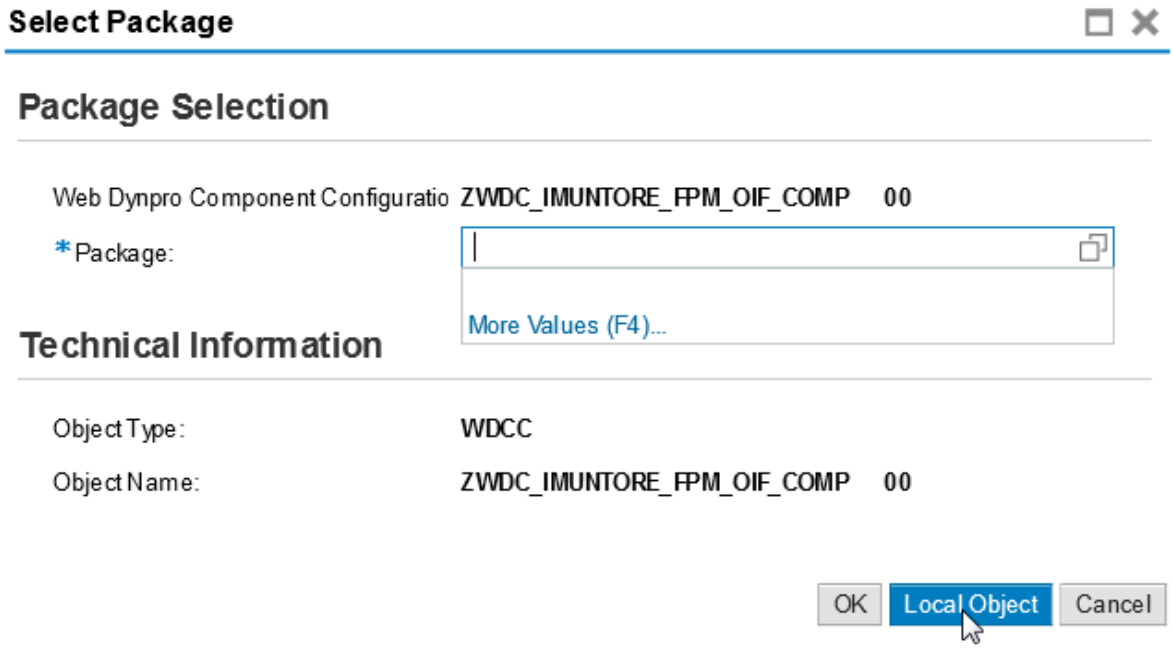
Save it



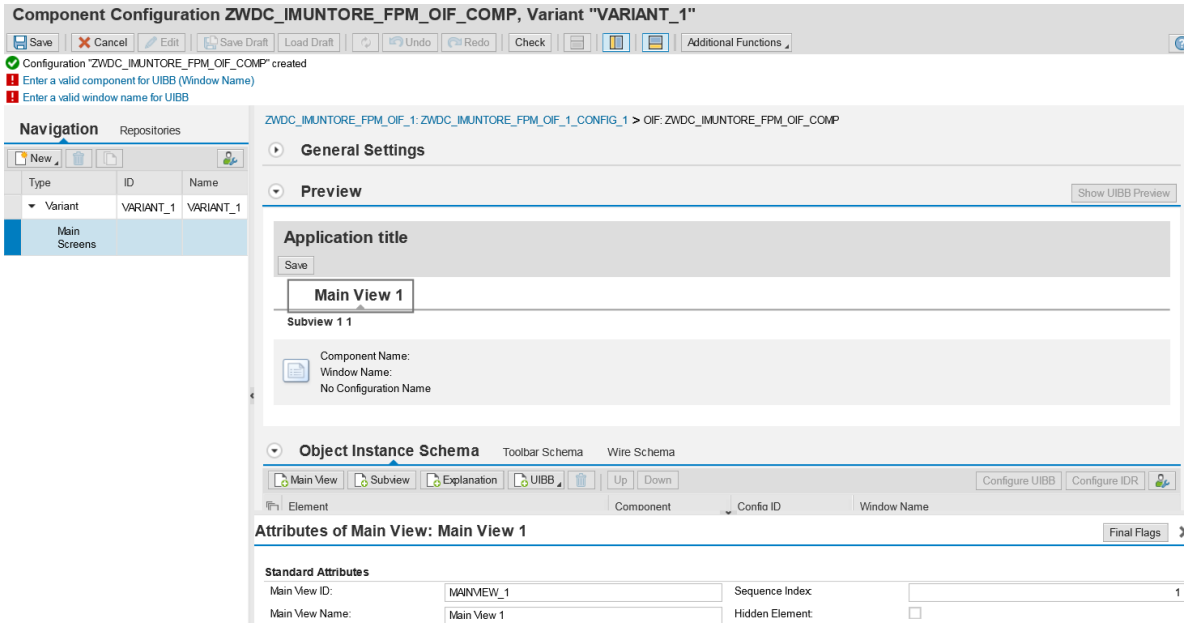
Click on the link to the Configuration as shown below:



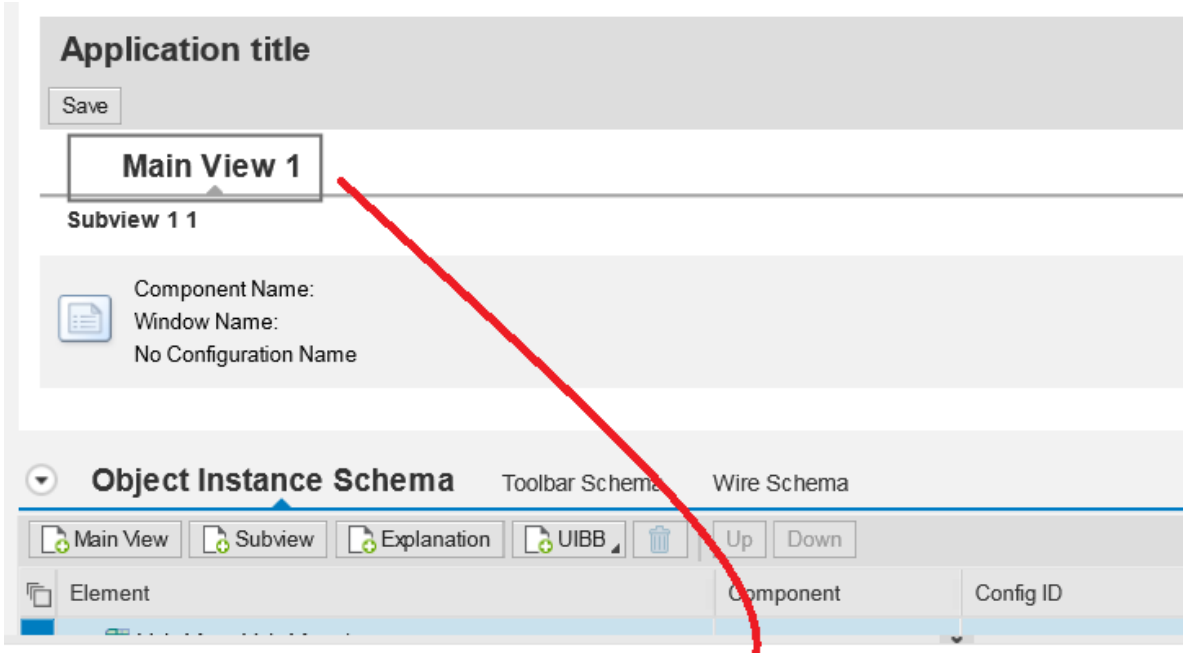


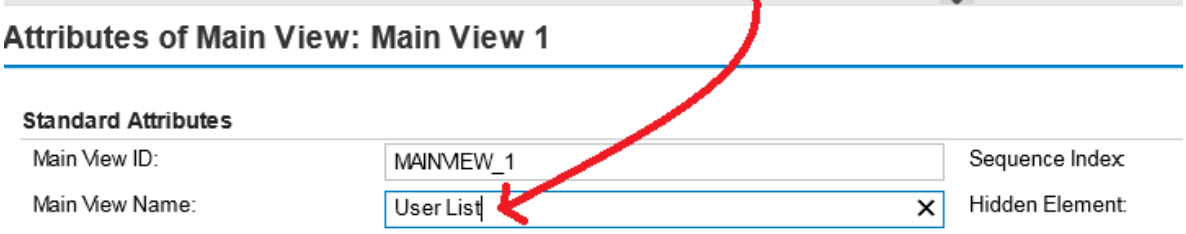


The following screen will open:



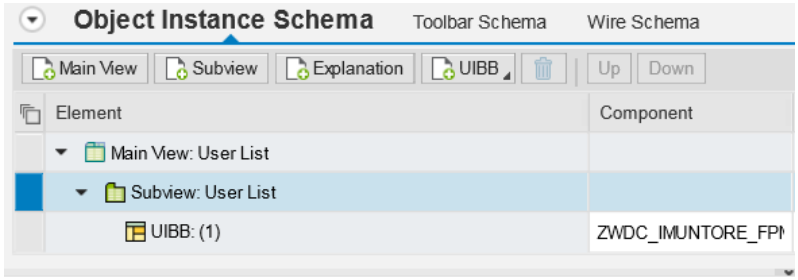
Here we will be able to configure our View as we want to:

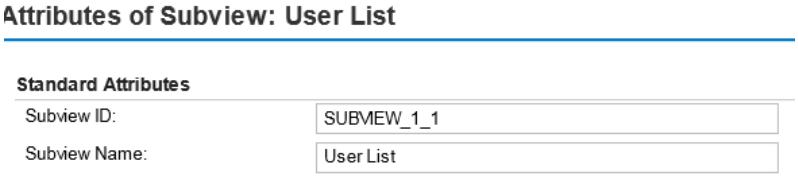




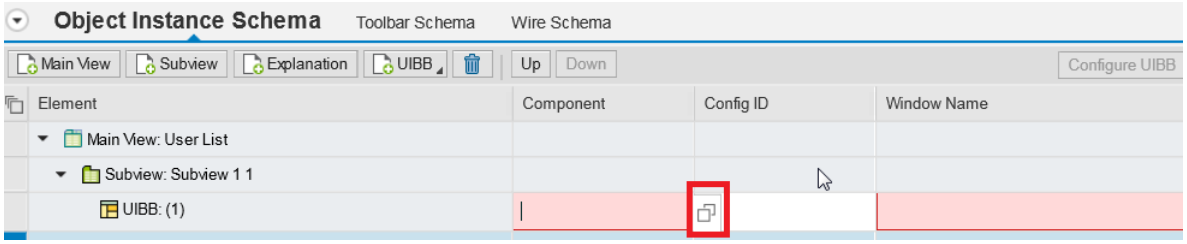
Save it: 

Click on the SubView and set the Subview name:

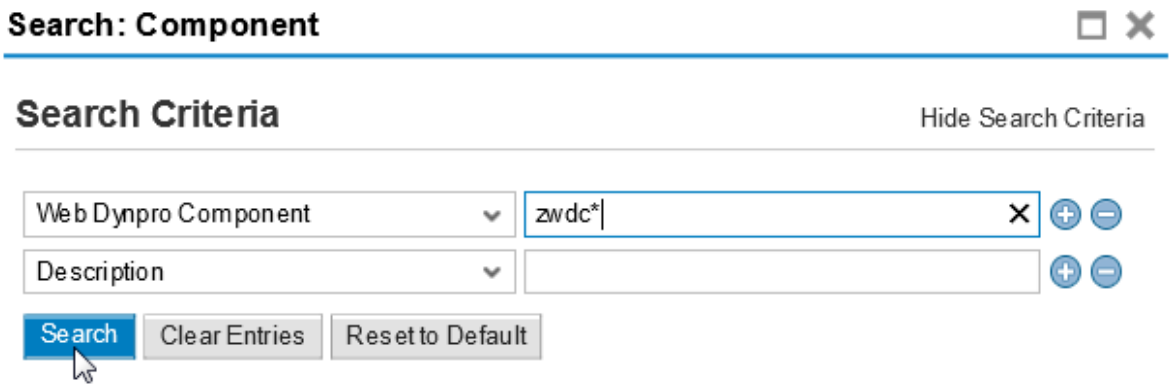




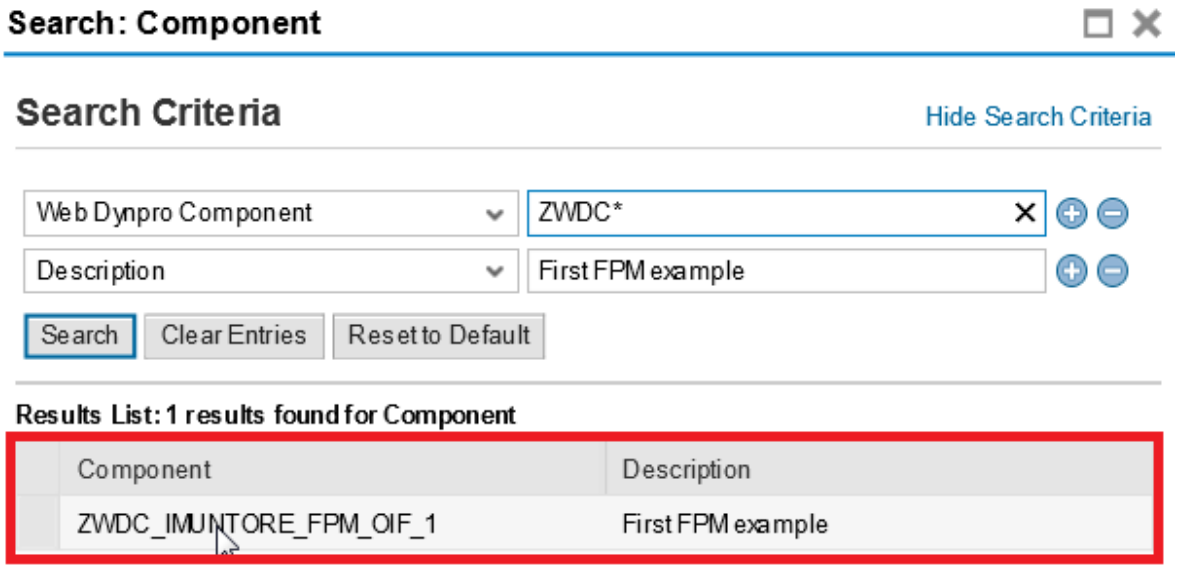
We need to fill some obligatory fields:



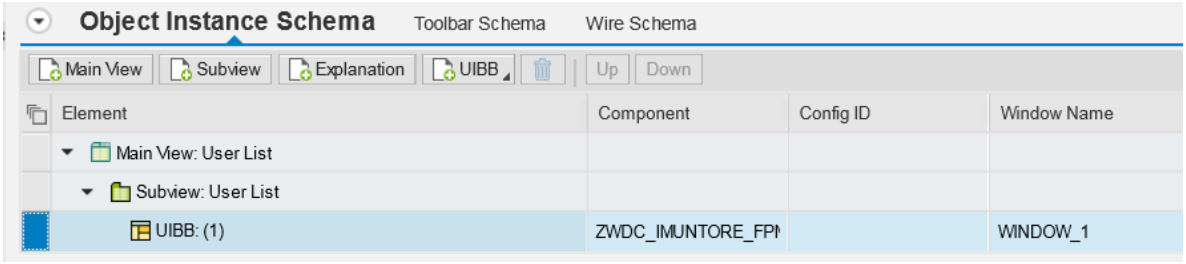
UIBB(User Interface Building Blocks): are components that implement the IF_FPM_UI_BUILDING_BLOCK interface.



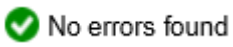
Select our Component by clicking on it



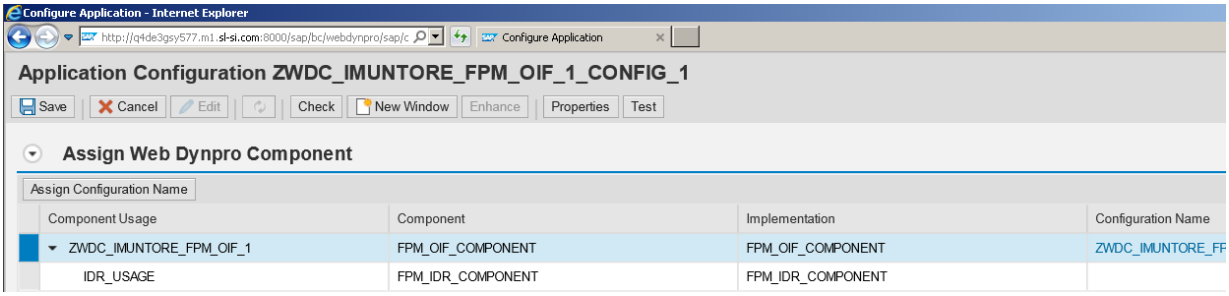
As we can see, the Component and the Window of the UIBB was automatically fed:



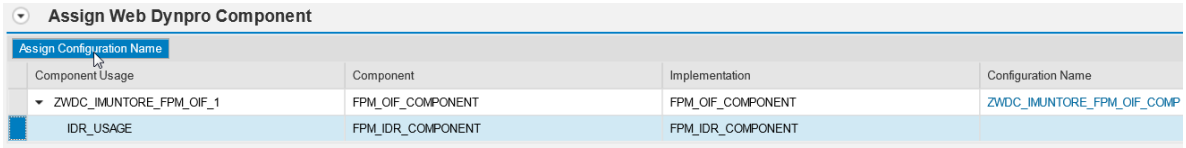
Save it and Check:



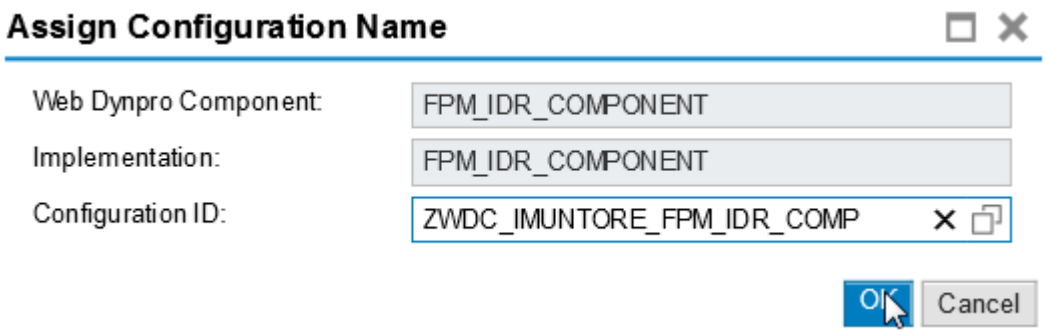
Come back to the Configuration Application:



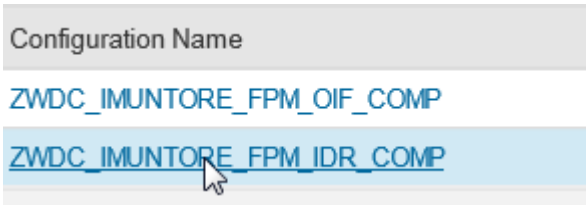
Select the IDR_USAGE and Click in Assign Configuration Name:



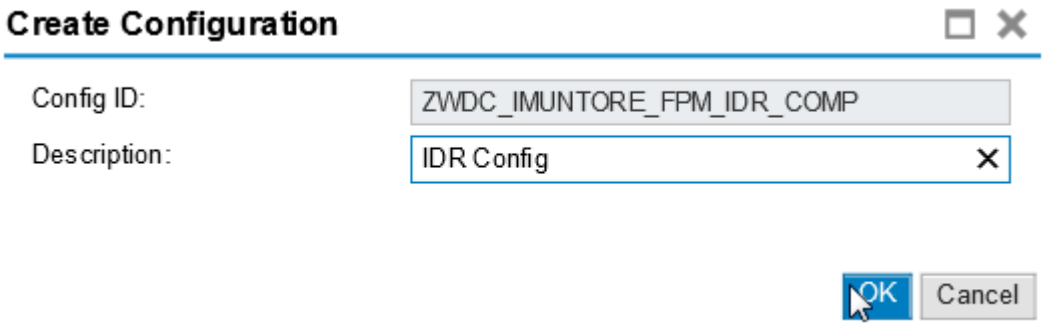
Choose a name and confirm:

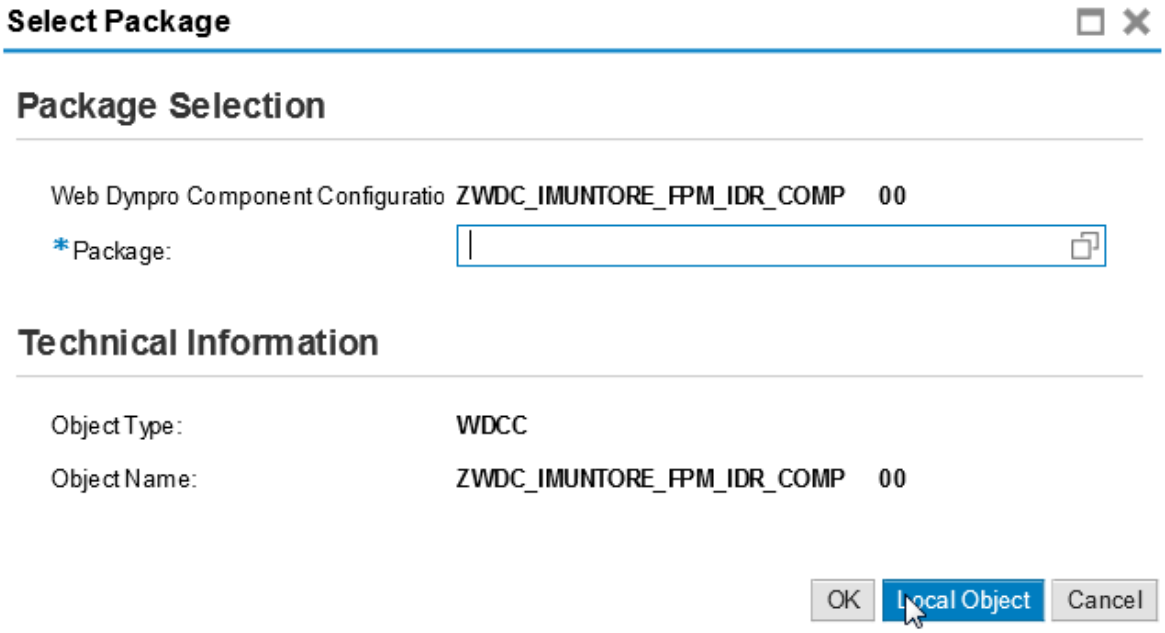


Save it and click in the new IDR Configuration:

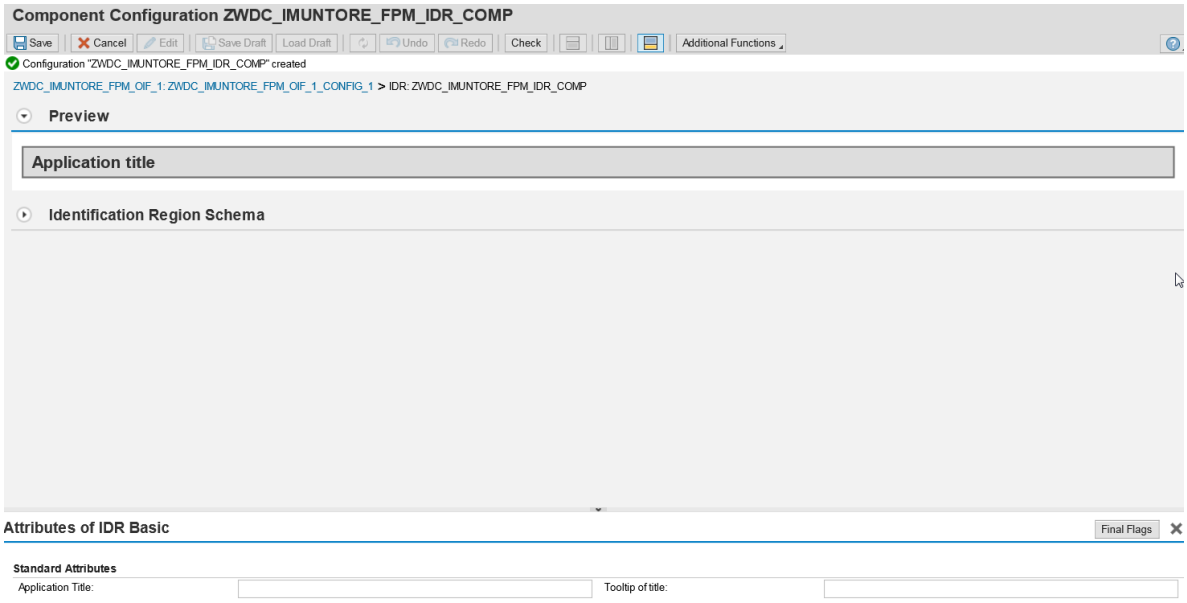


Create a description for it and save as local.

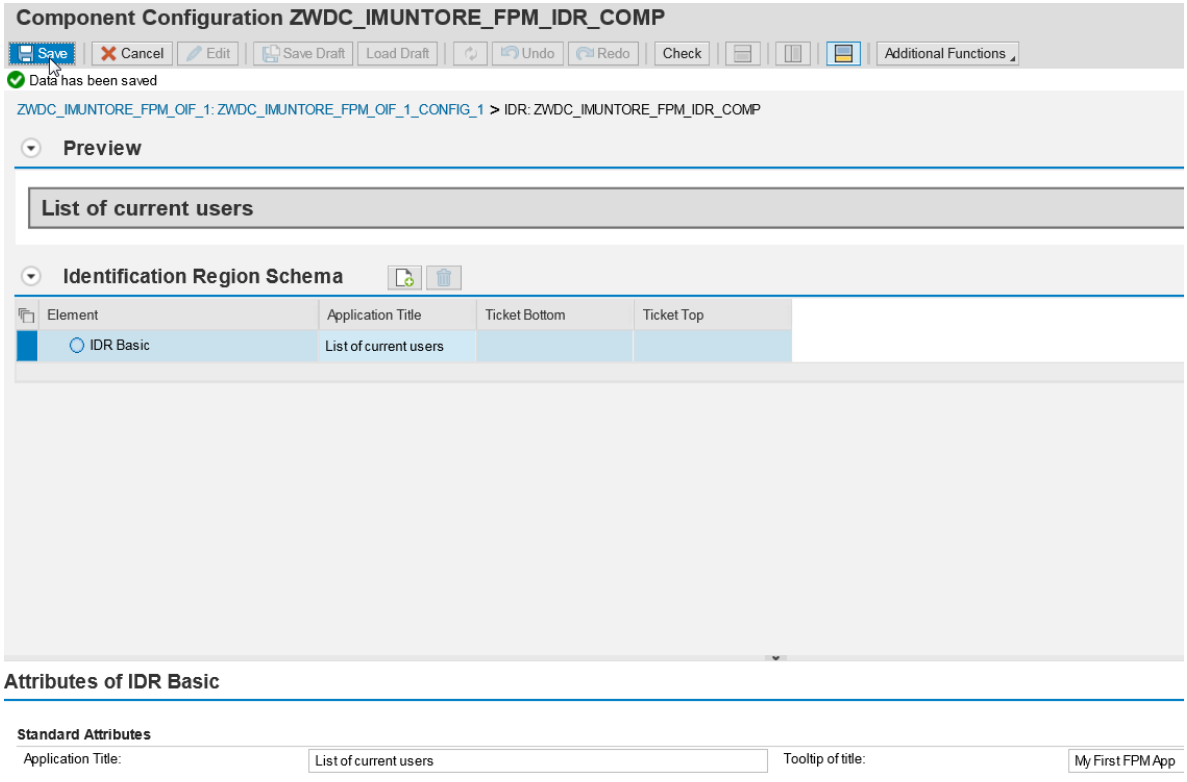




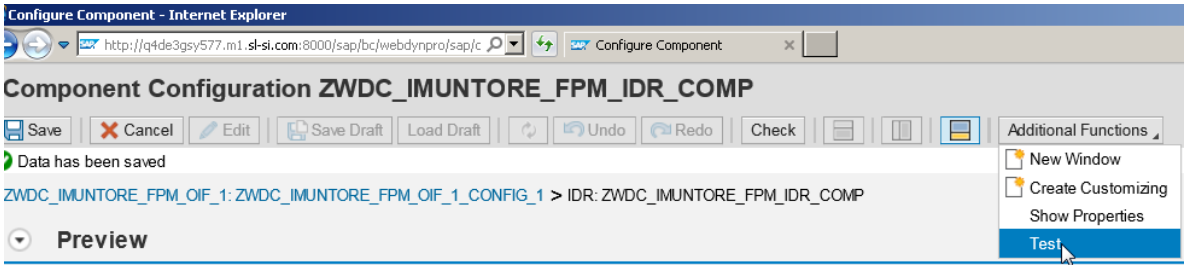
The following window will appear:



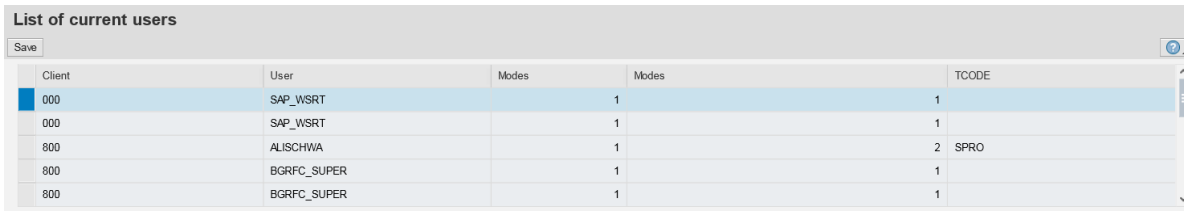
Give a name to the title and save it:



10) Testing



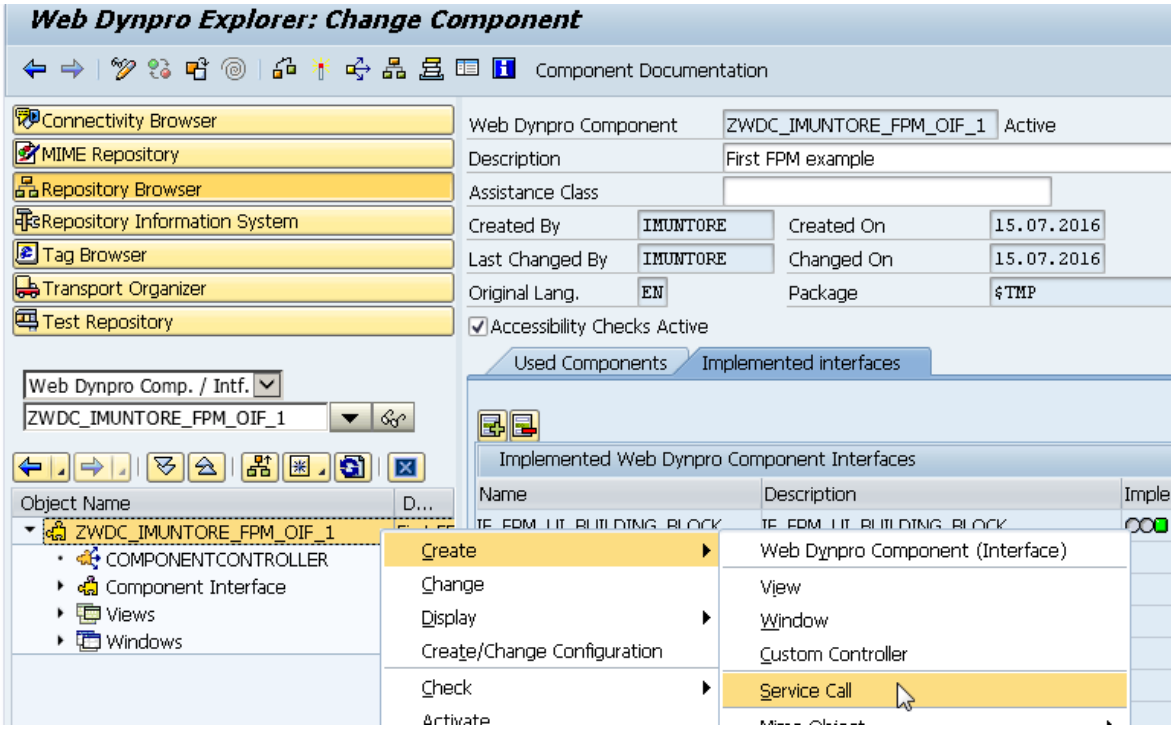
Result:



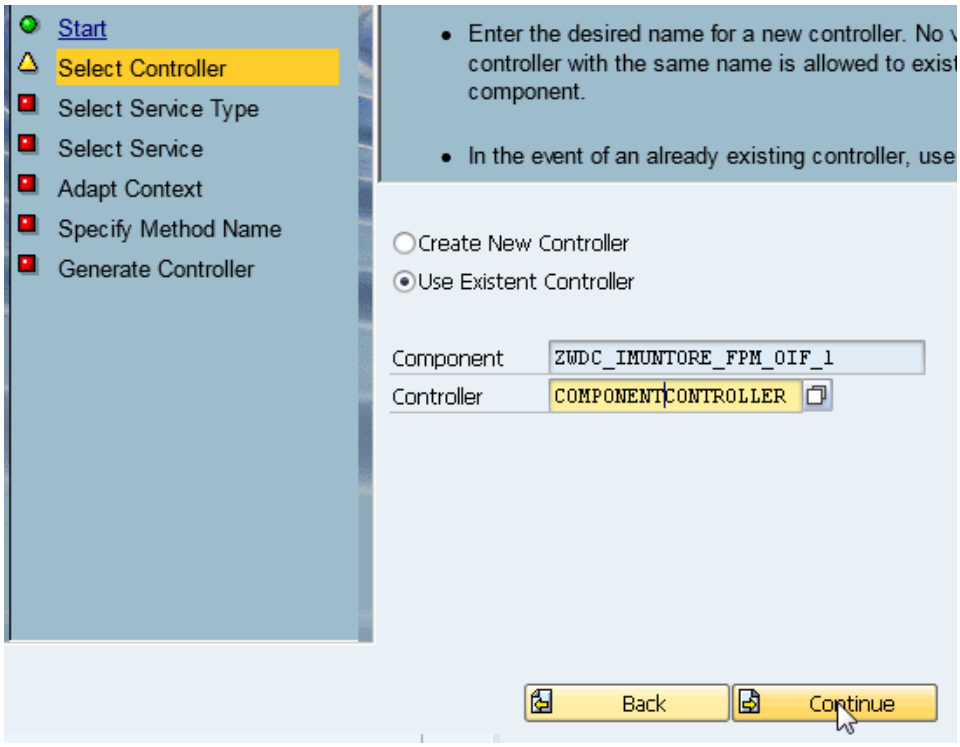
11) Understanding what we have done

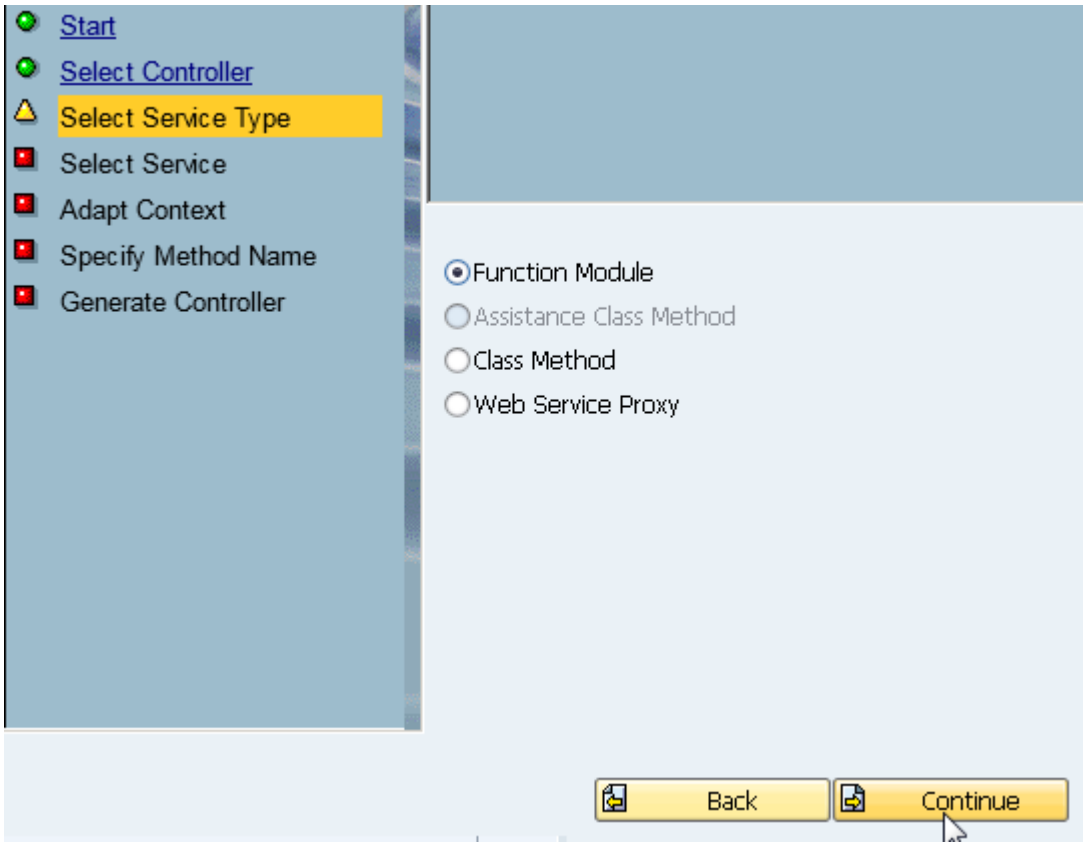
a. Until this moment, we just create our WD Component, Application and the assignment of that to the FPM thought the Implemented Interface IF_FPM_UI_BUILDING_BLOCK.
b. We have also used the OIF FPM configurator to set up our Application.
c. After the next steps you will understand better the utility of the Floorplan OIF.

12) Go to our component and create another Service Call:

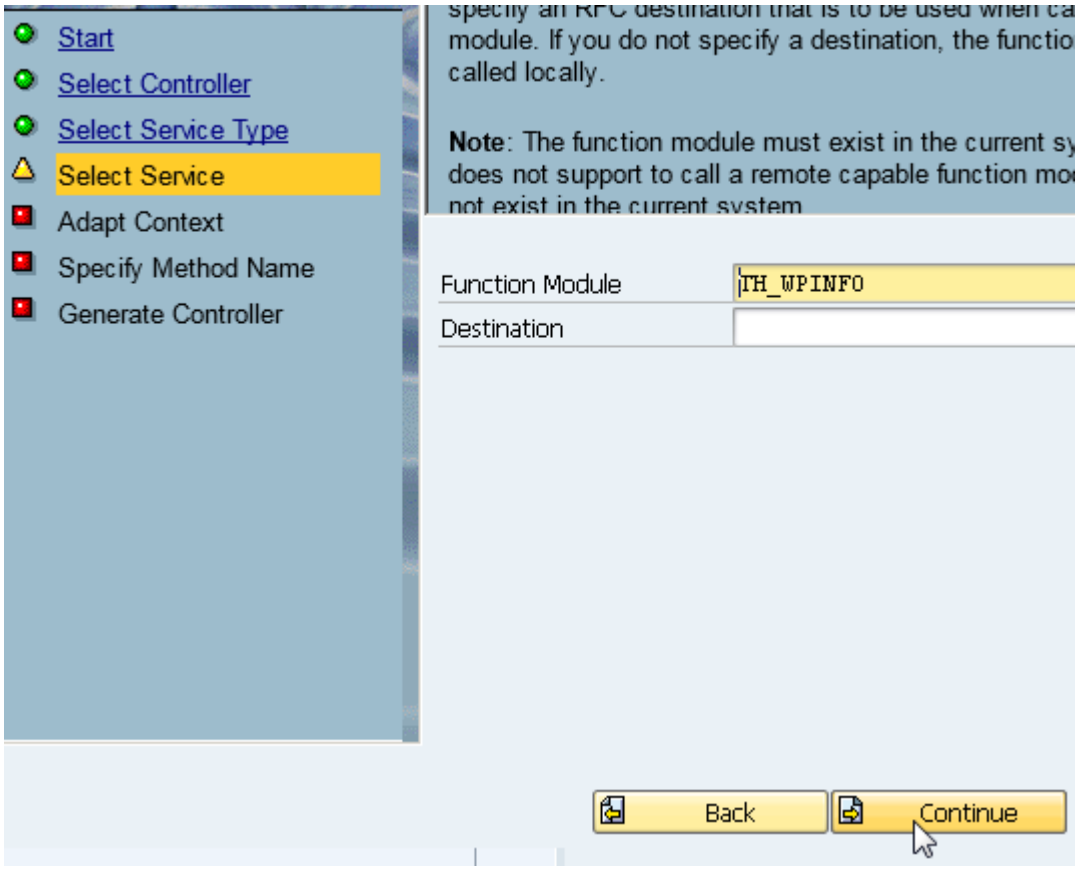


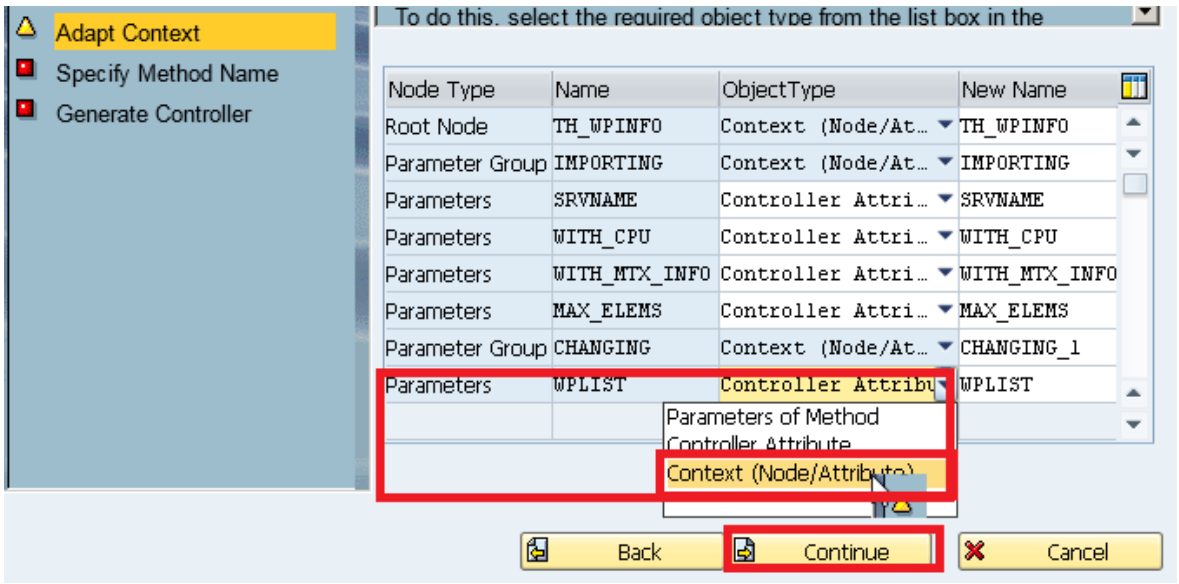
Follow the steps:

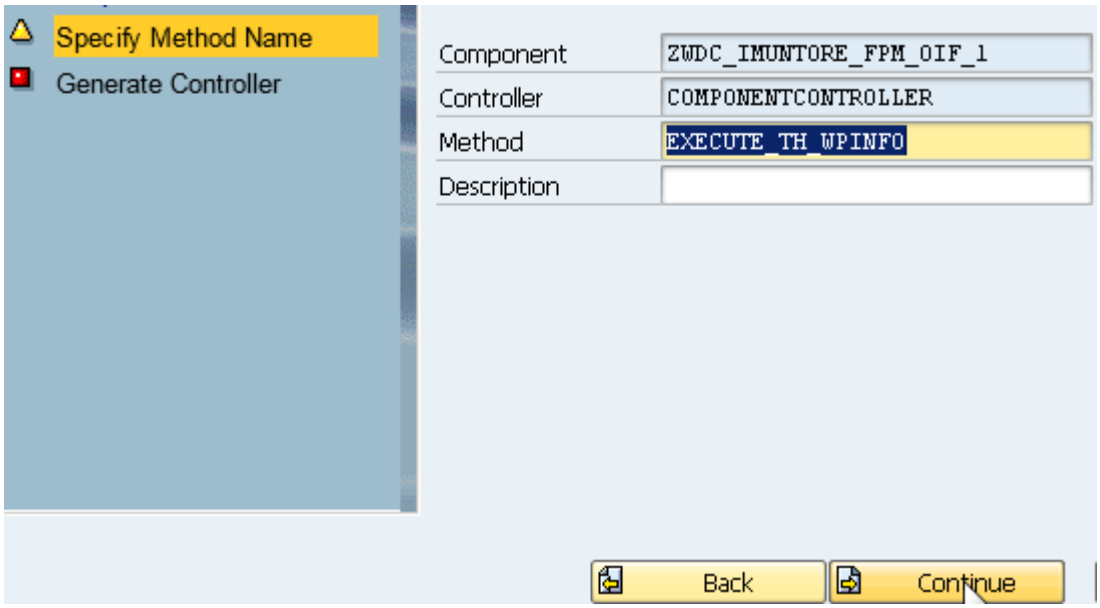


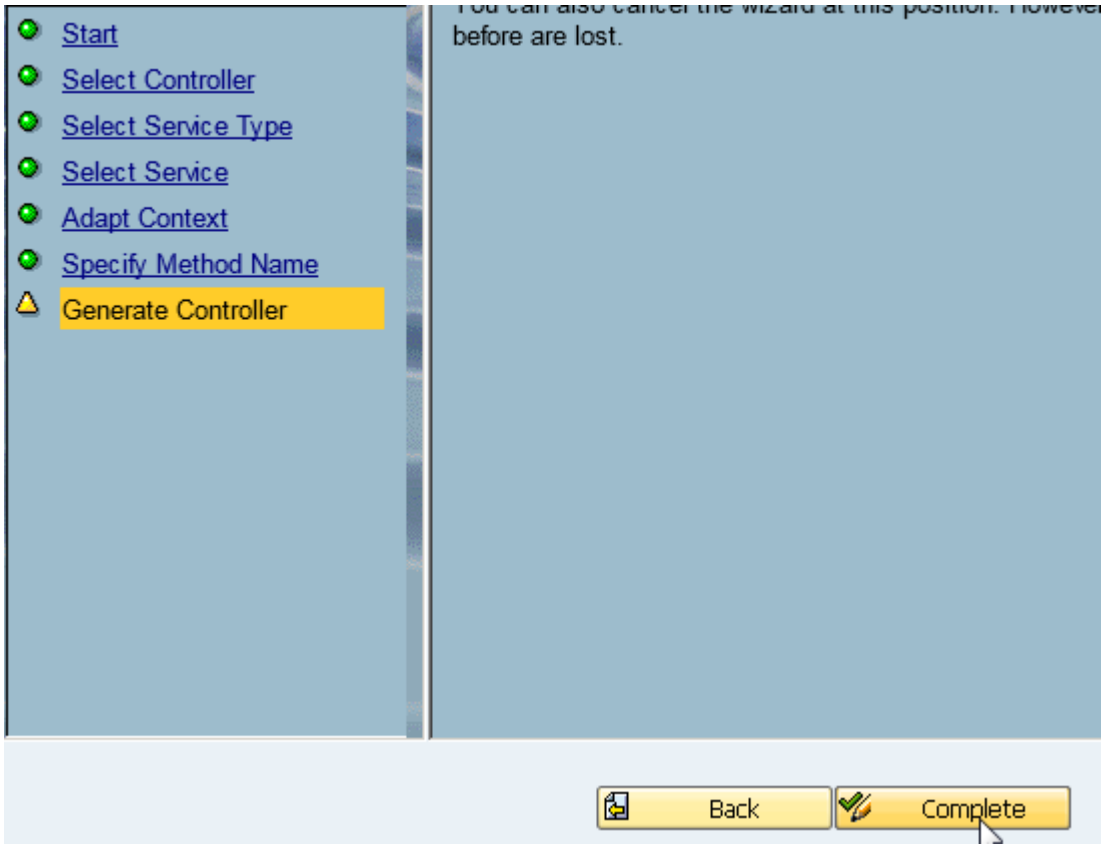


The function TH_WPINFO returns the currents processes that is being running in the system







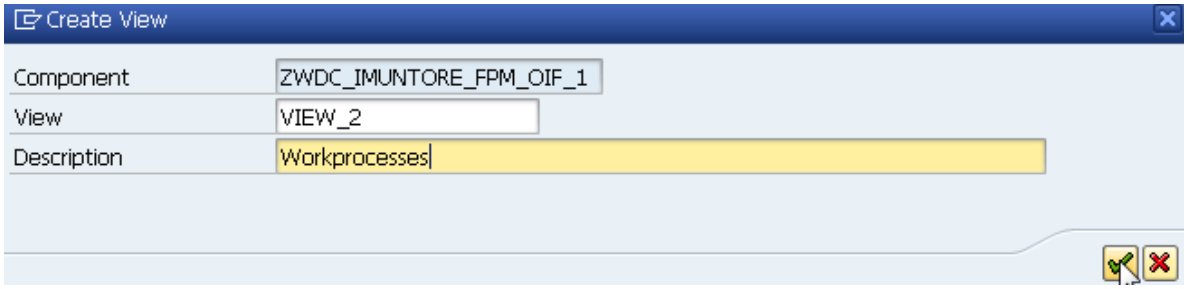


13) Go to the Component Controller methods

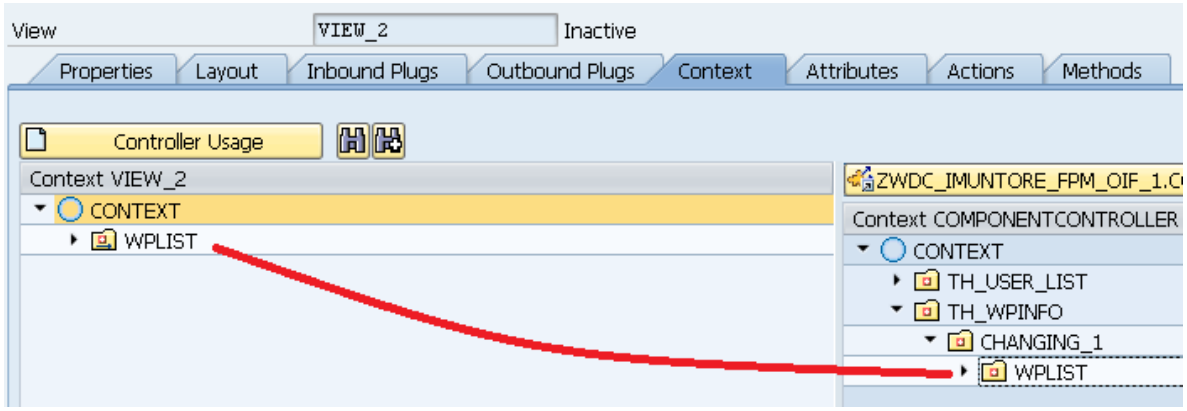
a. Inside the method WDDOINIT, code:

```
method WDDOINIT .

  wd_this->execute_th_user_list( ).
  wd_this->execute_th_wpinfo( ).

endmethod.
```

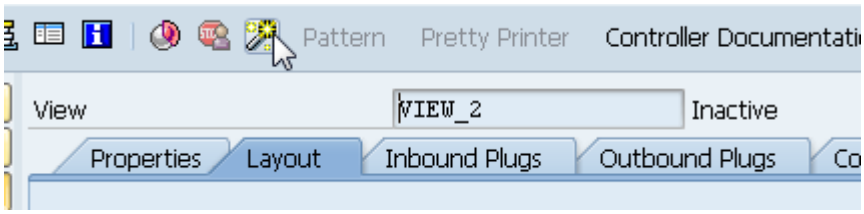
14) Create a New View:

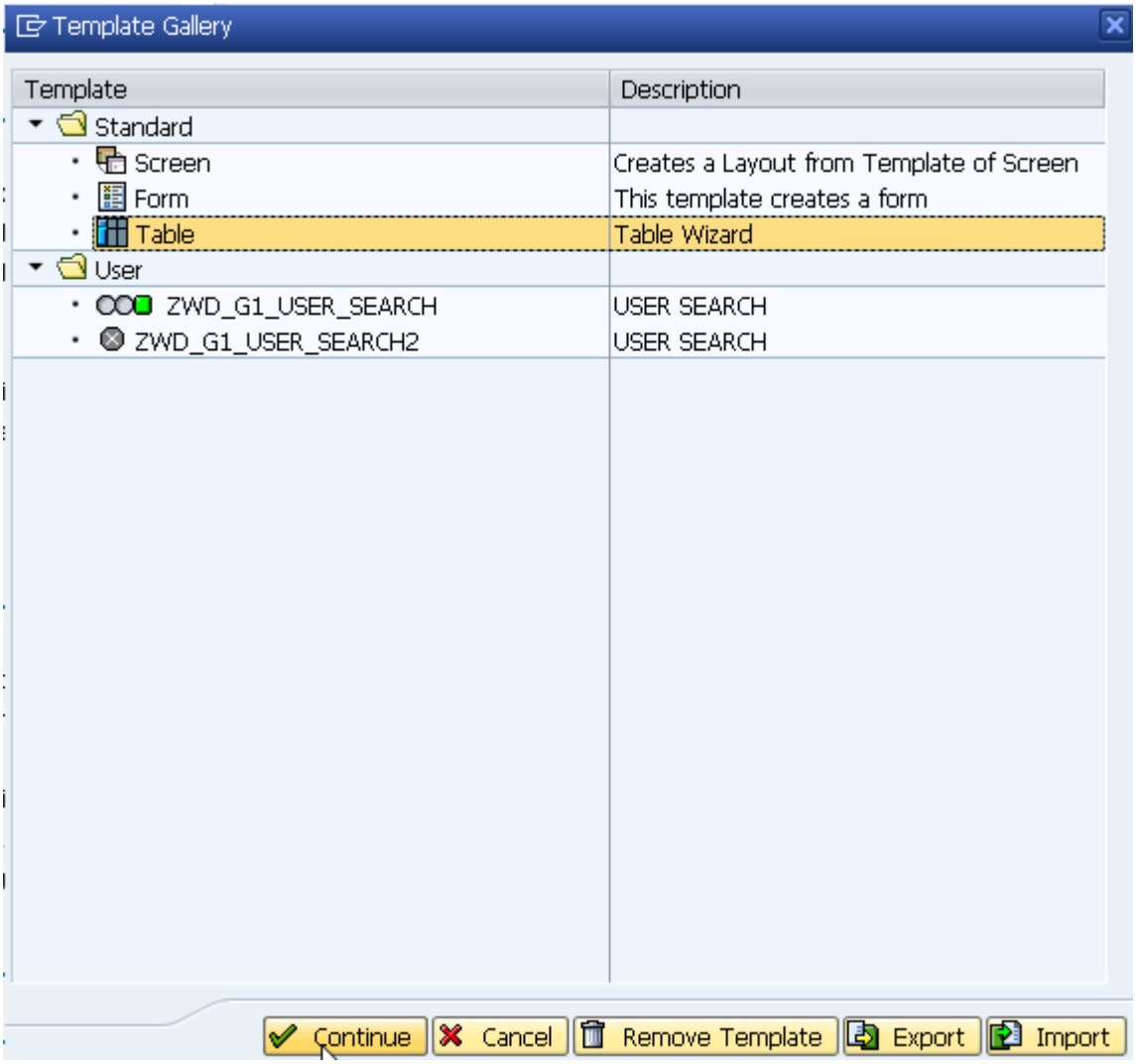


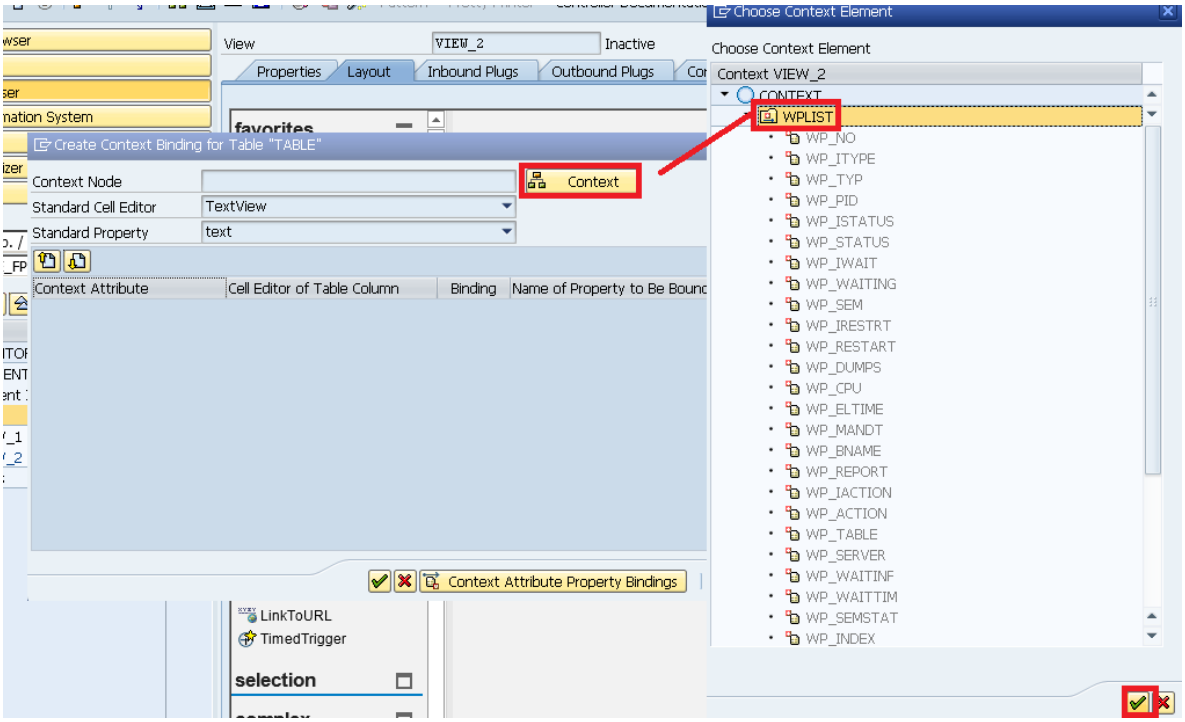
Drag and drop the following Node from the COMPONENTCONTROLLER context:

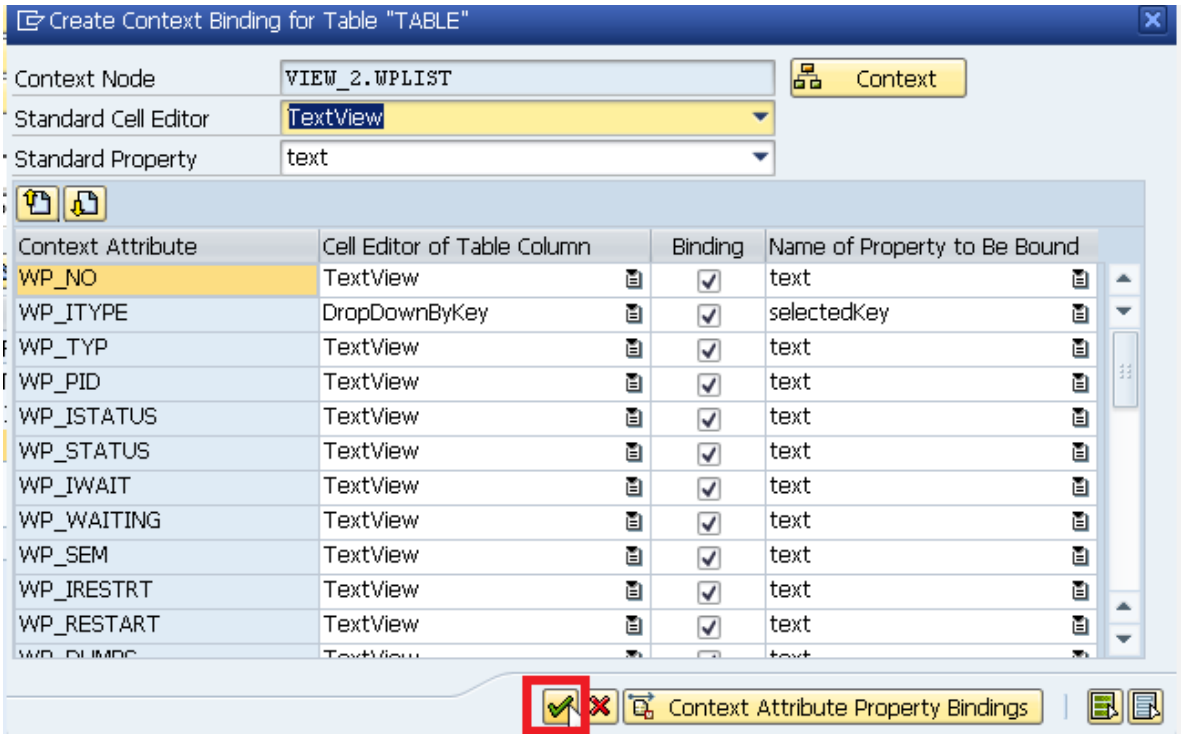


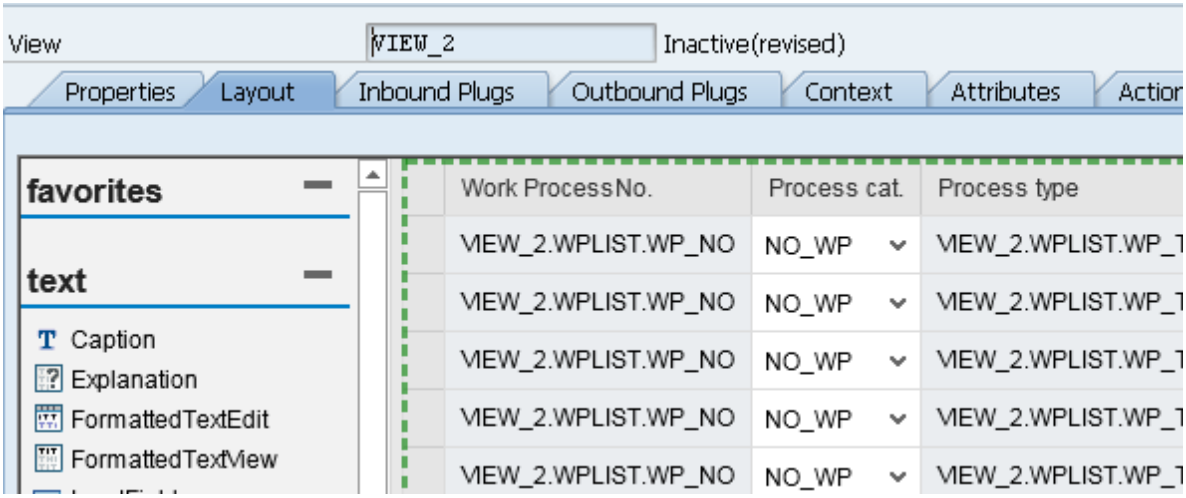
Go to the layout tab and create with the wizard:



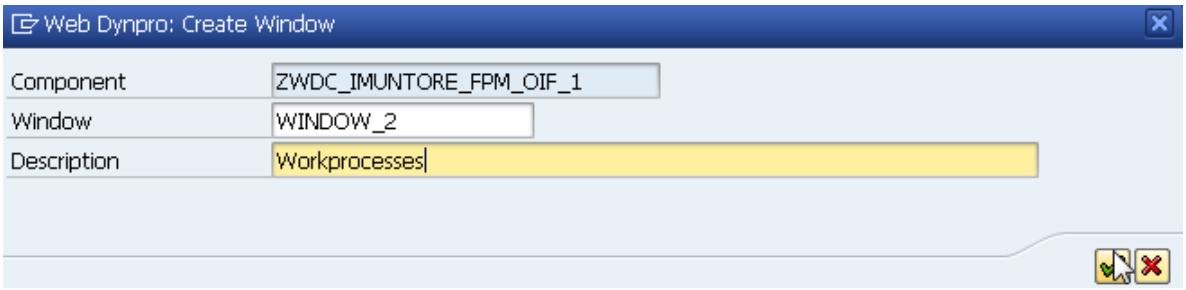

Template Gallery
Template
Description
Standard
Screen
Creates a Layout from Template of Screen
Form
This template creates a form
Table
Table Wizard
User
ZWD_G1_USER_SEARCH
USER SEARCH
ZWD_G1_USER_SEARCH2
USER SEARCH
Continue
Cancel
Remove Template
Export
Import

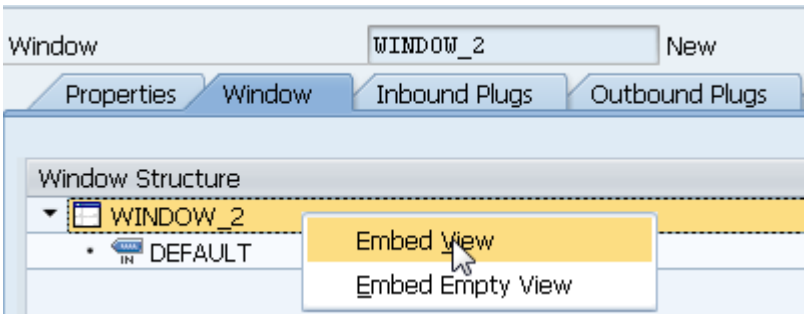

View
VIEW_2
Inactive
Properties
Layout
Inbound Plugs
Outbound Plugs
Create Context Binding for Table "TABLE"
Context Node
Standard Cell Editor
TextView
Standard Property
text
Context
Context Attribute
Cell Editor of Table Column
Binding
Name of Property to Be Bound
Context Attribute Property Bindings
LinkToURL
TimedTrigger
selection
Choose Context Element
Context VIEW_2
CONTEXT
WPLIST
WP_NO
WP_ITYPE
WP_TYP
WP_PID
WP_ISTATUS
WP_STATUS
WP_IWAIT
WP_WAITING
WP_SEM
WP_IRESTRT
WP_RESTART
WP_DUMPS
WP_CPU
WP_ELTIME
WP_MANDT
WP_BNAME
WP_REPORT
WP_IACTION
WP_ACTION
WP_TABLE
WP_SERVER
WP_WAITINF
WP_WAITTIM
WP_SEMSTAT
WP_INDEX

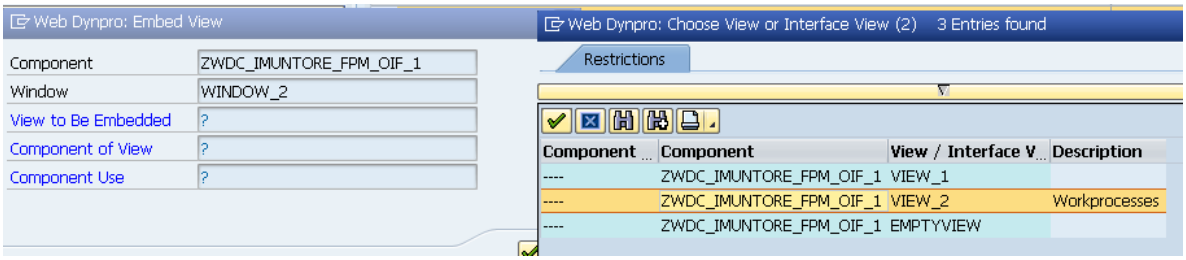

Create Context Binding for Table "TABLE"
Context Node
VIEW_2.WPLIST
Context
Standard Cell Editor
TextView
Standard Property
text
Context Attribute
Cell Editor of Table Column
Binding
Name of Property to Be Bound
WP_NO
TextView
text
WP_ITYPE
DropDownByKey
selectedKey
WP_TYP
TextView
text
WP_PID
TextView
text
WP_ISTATUS
TextView
text
WP_STATUS
TextView
text
WP_IWAIT
TextView
text
WP_WAITING
TextView
text
WP_SEM
TextView
text
WP_IRESTRT
TextView
text
WP_RESTART
TextView
text
Context Attribute Property Bindings

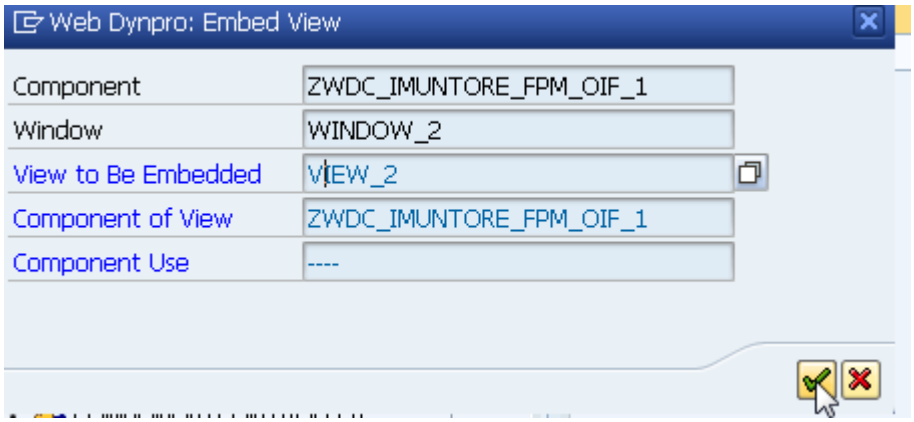


15) Create a new window:



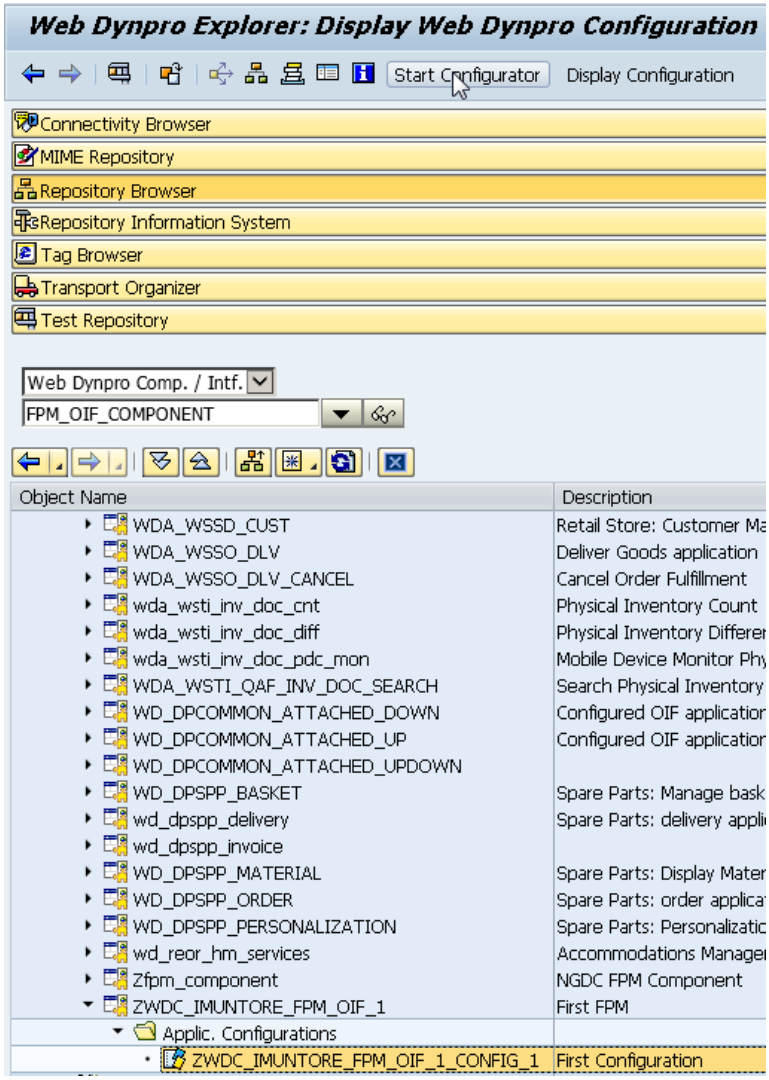


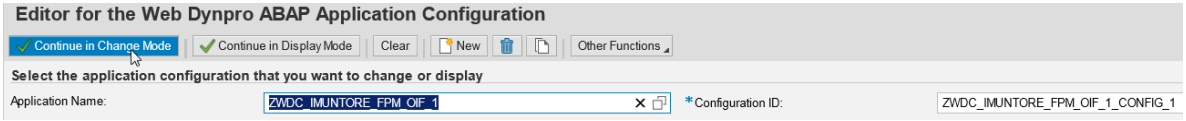




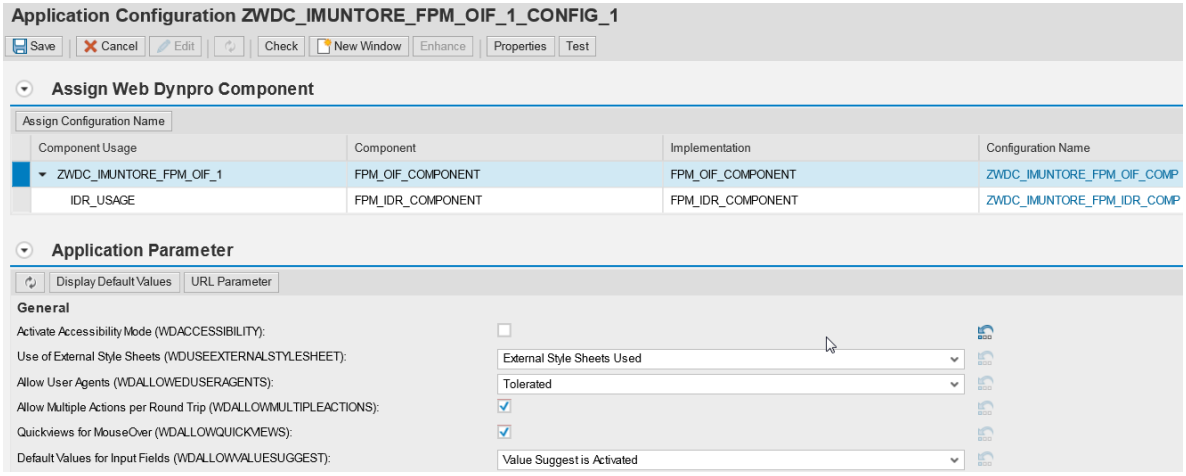
Save and Activate.

16) Starting the FPM OIF Configuration

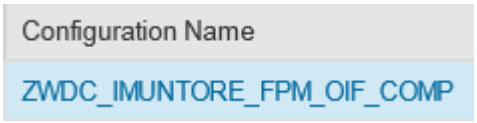




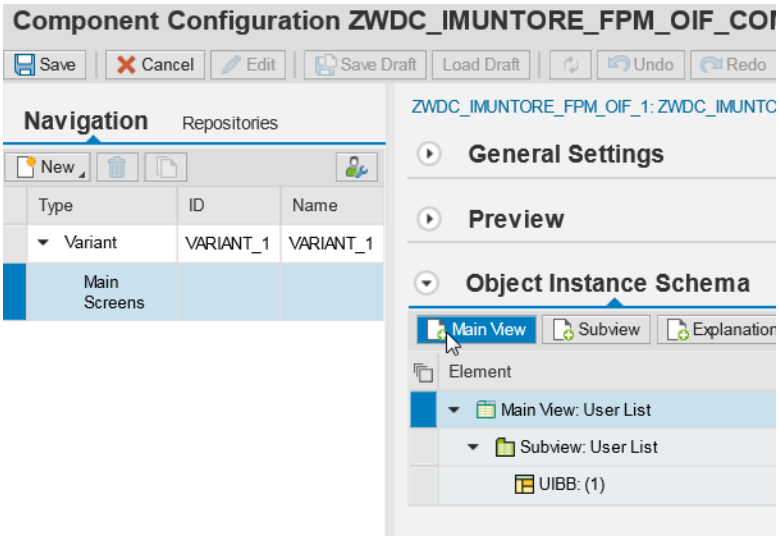
The following window will open:

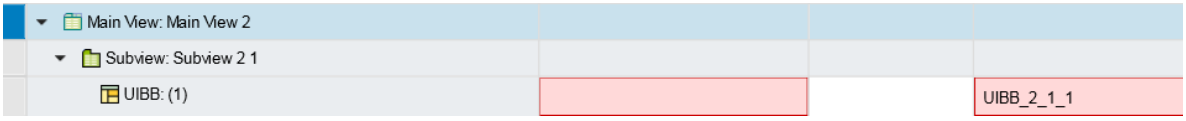


Click here:

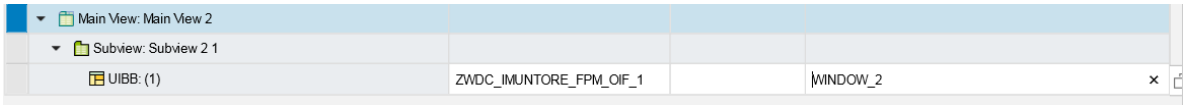


Add a Main View:

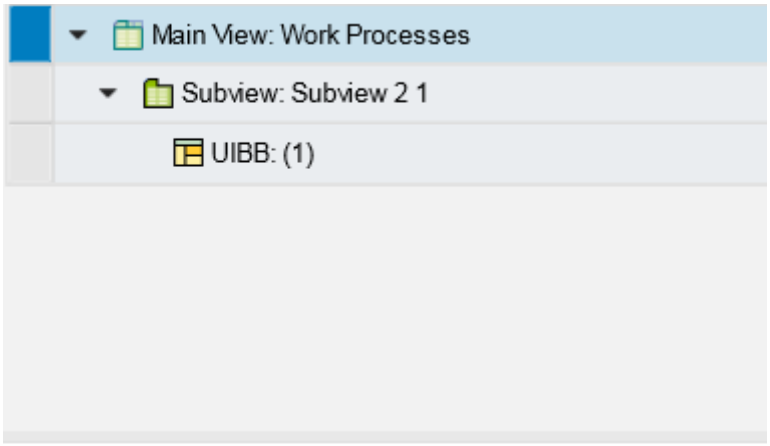


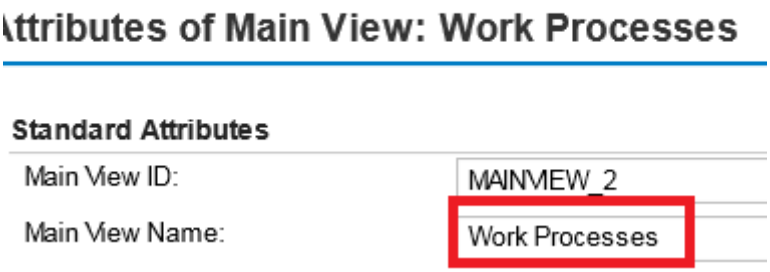


Fill the red fields:

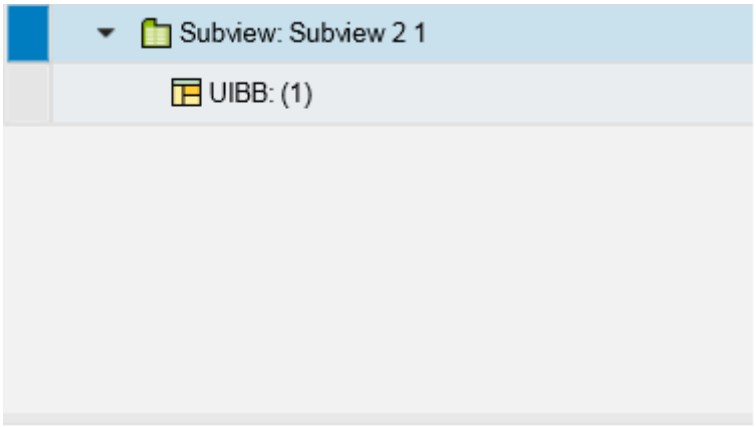


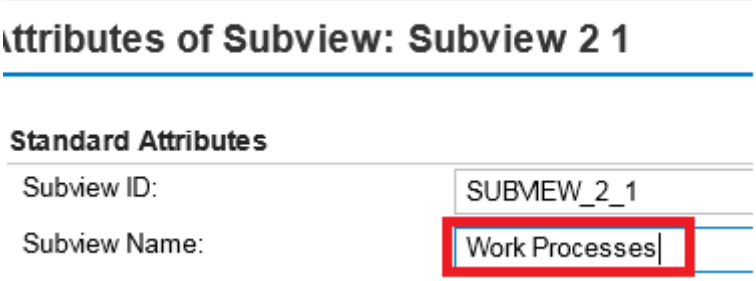
Update the view name:



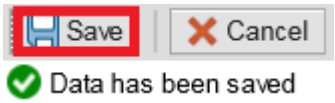


Update the sub view name:





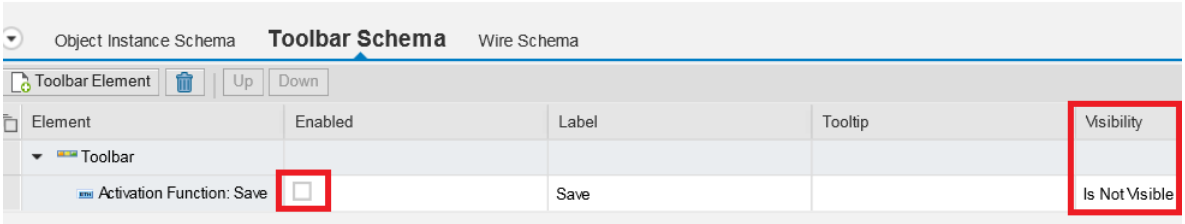
Save it



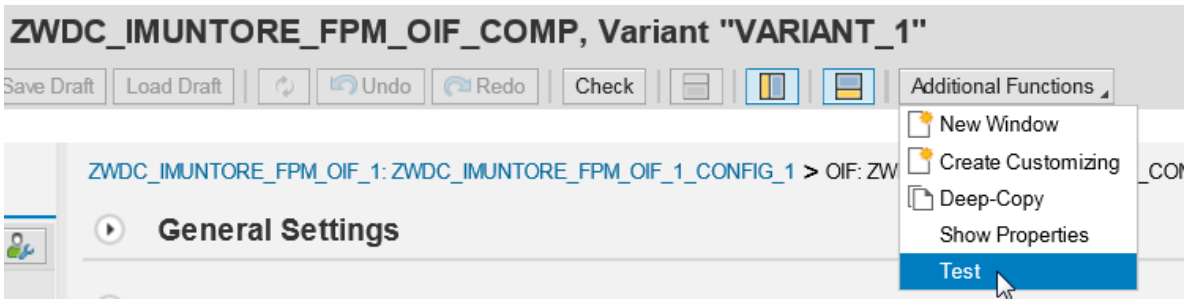
17) Toolbar Schema

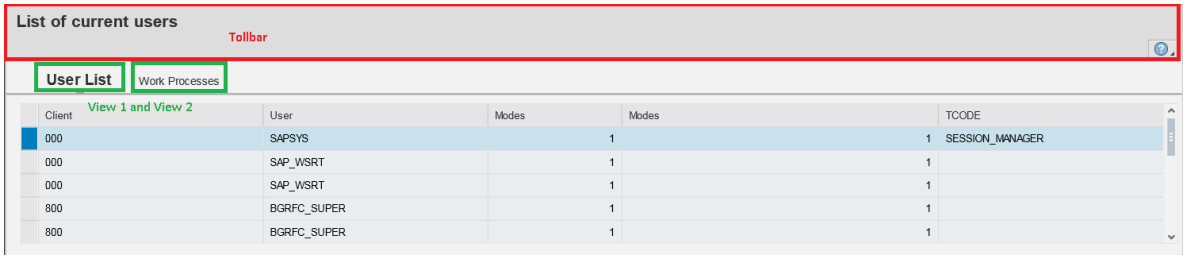
Here you can configure your tollbar:

Disable the Savel Function and its Visibility



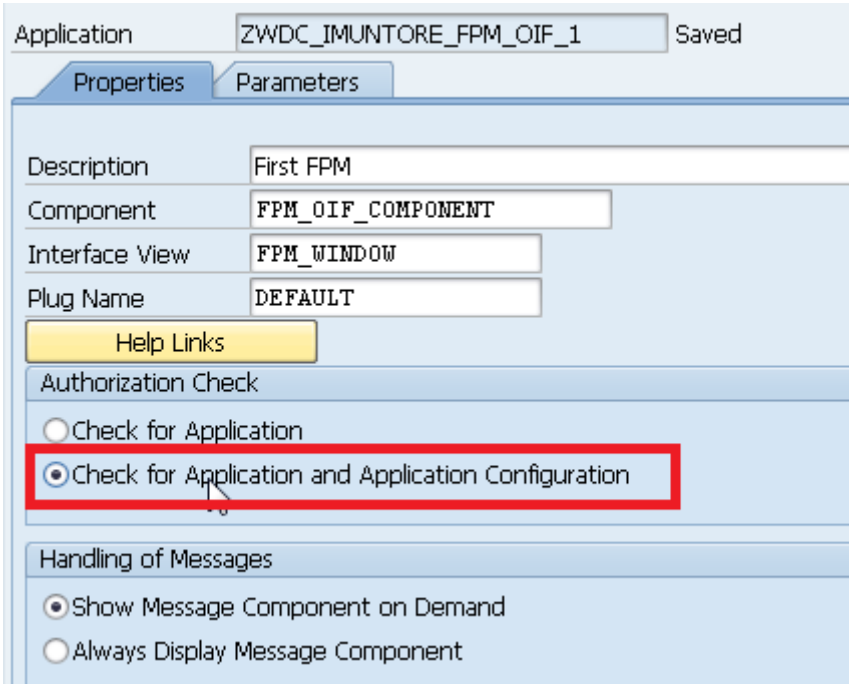
18) Test it!



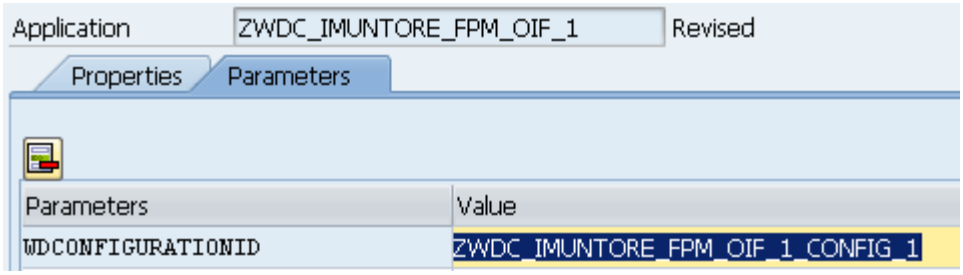


19) Executing the WDA:
Open the WDA ZWDC_IMUNTORE_FPM_OIF_1 and try to execute. You will get the following error:

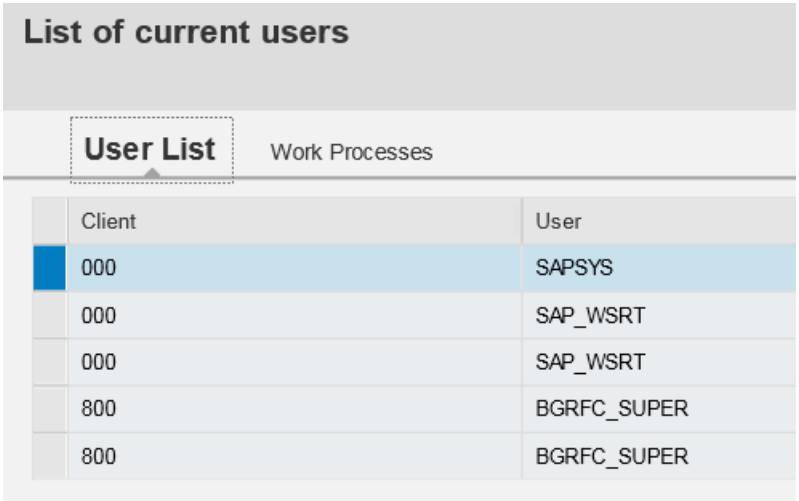
Configure the WDA as shown below:

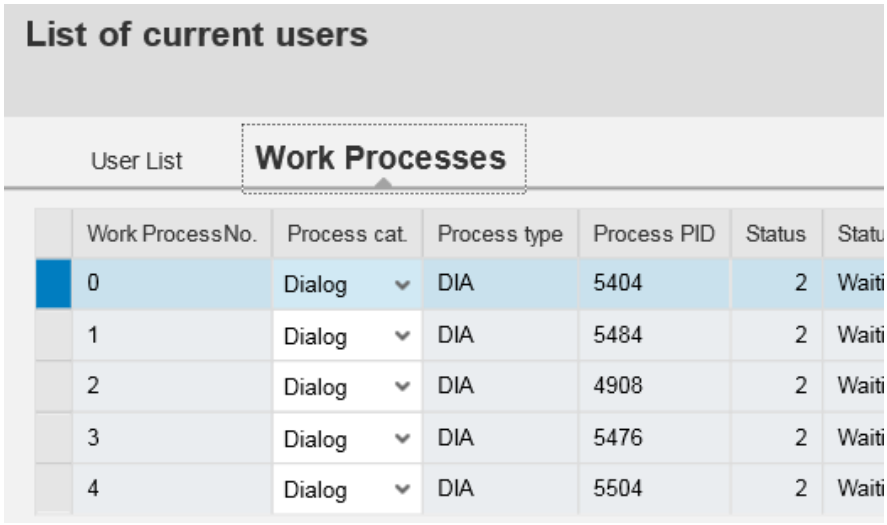


Go to the Parameters tab:



Save and Run:

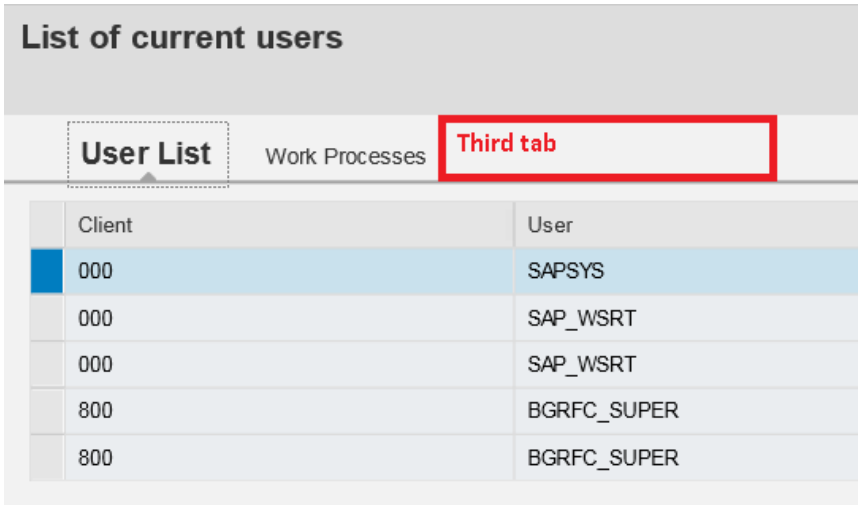




20) Homework:

a. Change the FPM title from “List of current users” to “My Views”

b. Create a third tab with reference to some of your created WD Components.



c. Start the FPM Configurator, create a new “Refresh“ button in the Tollbar by changing the Tollbar Schema inside the OIF Component ZWDC_IMUNTORE_FPM_OIF_COMP.

d. Create an event to the Refresh button inside the Method “PROCESS_EVENT” in the COMPONENTCONTROLLER. The idea is: when the user presses the button, call the methods below again:

i.
```
wd_this->execute_th_user_list( ).
wd_this->execute_th_wpinfo( ).
```