

# Fusion development with ABAP Cloud in SAP Build



Pedro Vieira  
João Rocha  
Anderson Silva

# Hands-on Lab: Fusion Development with ABAP Cloud in SAP Build

**asUG**  Tech Connect

In partnership with **SAP TechEd**

# Agenda

## Exercise 1

- Start your development by creating an ABAP project in the SAP Build Lobby

## Exercise 2

- Create a Shopping Cart business object, projection and service with the Graphical Modeler in SAP Build Code

## Exercise 3

- Implement transactional behavior in ADT, create validations and determinations

## Exercise 4

- Create, preview and adapt a SAP Fiori elements application



Github

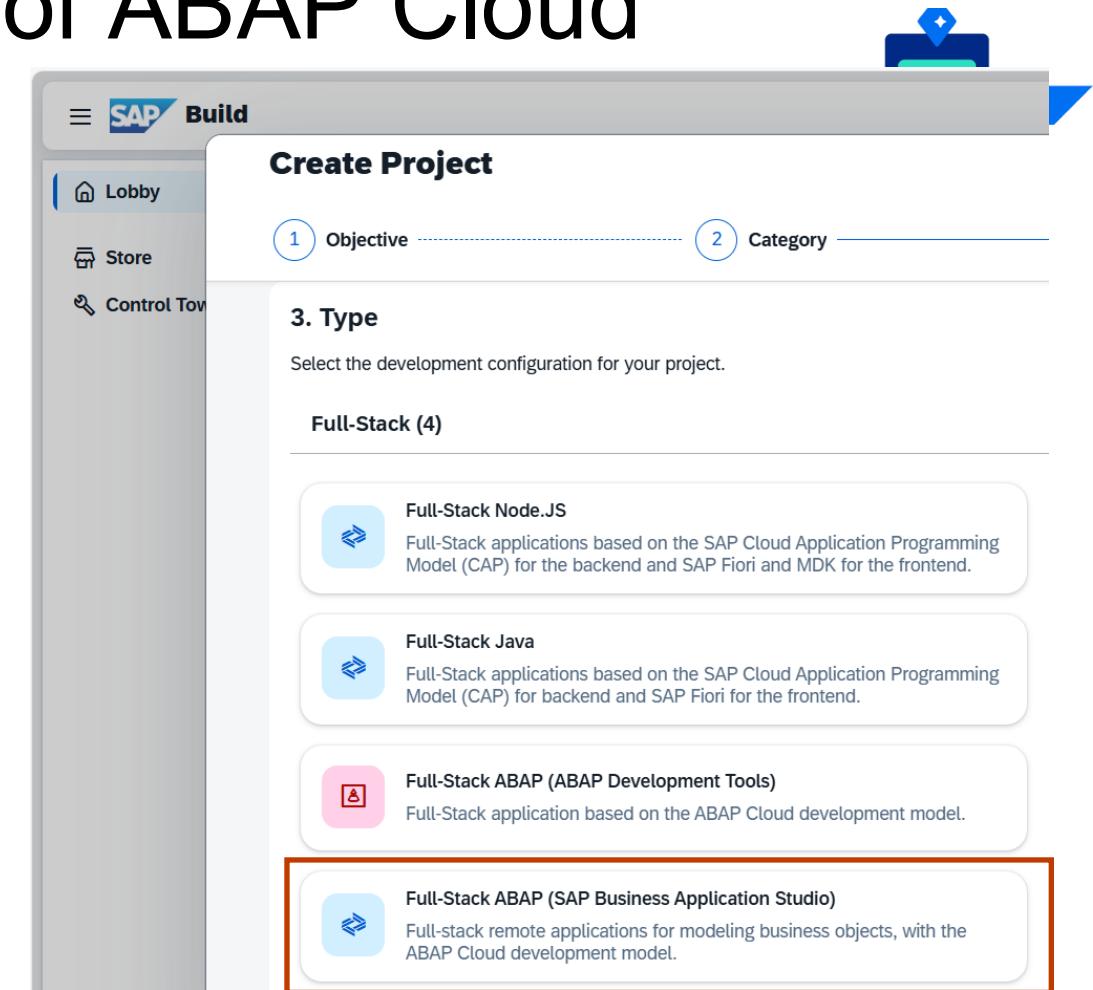


# Introduction

asUG Tech Connect  
In partnership with SAP TechEd

# What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)



# What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)
2. SAP Business Application Studio - Graphical Modeler



The screenshot shows two entity definitions in SAP Business Application Studio:

**Cart** Business Object

	Field Name	Type
Ab	OrderUuid	abap.raw(16)
Ab	Currency	abap.cuky
Ab	Notes	abap.char(100)
Ab	OrderID	abap.numc(8)
Ab	RequestDeliveryDate	abap.datn
12	TotalPrice	abap.curr(11,2)

**Item** Entity

	Field Name	Type
Ab	ItemUuid	abap.raw(16)
10	ParentUuid	abap.raw(16)
Ab	ItemID	abap.numc(8)
Ab	OrderedItem	abap.char(40)
Ab	Quantity	abap.numc(4)
12	ItemPrice	abap.curr(11,2)
Ab	Currency	abap.cuky
12	ItemUnitPrice	abap.curr(11,2)

At the bottom right, there are navigation icons for zooming and searching, along with a status bar that says "Layout: German".

# What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)
2. SAP Business Application Studio – Graphical Modeler
3. ABAP Development Tools - Implementation of Business Logic



```
35
36 validation validateRequestDeliveryDate on save { create; field RequestDeliveryDate; }
37
38 draft action Activate optimized;
39 draft action Discard;
40 draft action Edit;
41 draft action Resume;
42 //draft determine action Prepare;
43 draft determine action Prepare
44 {
45   validation validateRequestDeliveryDate;
46 }
```

A screenshot of an ABAP code editor. The code shown is validation logic for a 'validateRequestDeliveryDate' method. It includes several 'draft' actions and a block of code starting with '//draft determine action Prepare'. The code is highlighted with a red box around the validation block. The editor interface shows line numbers from 35 to 67.

```
64
65 association _Items { create; with draft; }
66
67 }
```

A screenshot of an ABAP code editor showing association logic. The code defines an association named '\_Items' with creation and draft properties. The editor interface shows line numbers from 64 to 67.

```
METHOD validateRequestDeliveryDate.|  
ENDMETHOD.  
Predict RAP Business Logic
```

A screenshot of an ABAP code editor showing a 'METHOD validateRequestDeliveryDate.' declaration. Below it is an 'ENDMETHOD.' declaration. A tooltip labeled 'Predict RAP Business Logic' is visible near the end of the method declaration.

# What you will do in this hands-on workshop – Practice fusion development of ABAP Cloud

1. SAP Build Lobby - Create a Full-Stack ABAP project (SAP Business Application Studio)
2. SAP Business Application Studio – Graphical Modeler
3. ABAP Development Tools - Implementation of Business Logic
4. SAP Business Application Studio - Create, preview and adapt a SAP Fiori elements application



**Exercise 1:**  
**Start your**  
**development by**  
**creating an**  
**ABAP project in**  
**the SAP Build**  
**Lobby**

asUG Tech Connect

In partnership with SAP TechEd

# Exercise 1: Create a Full-Stack ABAP (SAP Business Application Studio) Project

Create a Full-Stack ABAP (SAP Business Application Studio) Project

Connect to backend system

**H63**

Use a pre-created package

**ZAD163-A##**

Create and start Dev Space  
in SAP Business Application Studio

Full-Stack ABAP (SAP Business Application Studio)  
Full-stack remote applications for modeling business objects, with the ABAP Cloud development model.

**Create Project**

4. System and Package

Specify the ABAP system and package to be used

Before proceeding, please make sure you have all the prerequisites

**System**

System: \*  
H63  
System ID: H63  
System Client: 100

**Package**

Existing  
ZAD163\_Z01  
New

**Create Project**

6. Name

Give your project a name.

Name: \*  
ZAD163\_Z01  
Description:  
Project for ZAD163\_Z01  
CSN File Name: \*  
ZAD163\_Z01  
Dev Space: \*  
ZAD163\_Z01

You can run up to two dev spaces at a time. For more options, go to the [Dev Space Manager](#)

Review Cancel

# **Exercise 2 :**

## **Create a Shopping**

### **Cart business**

#### **object,**

#### **projection and**

#### **service with the**

#### **Graphical Modeler**

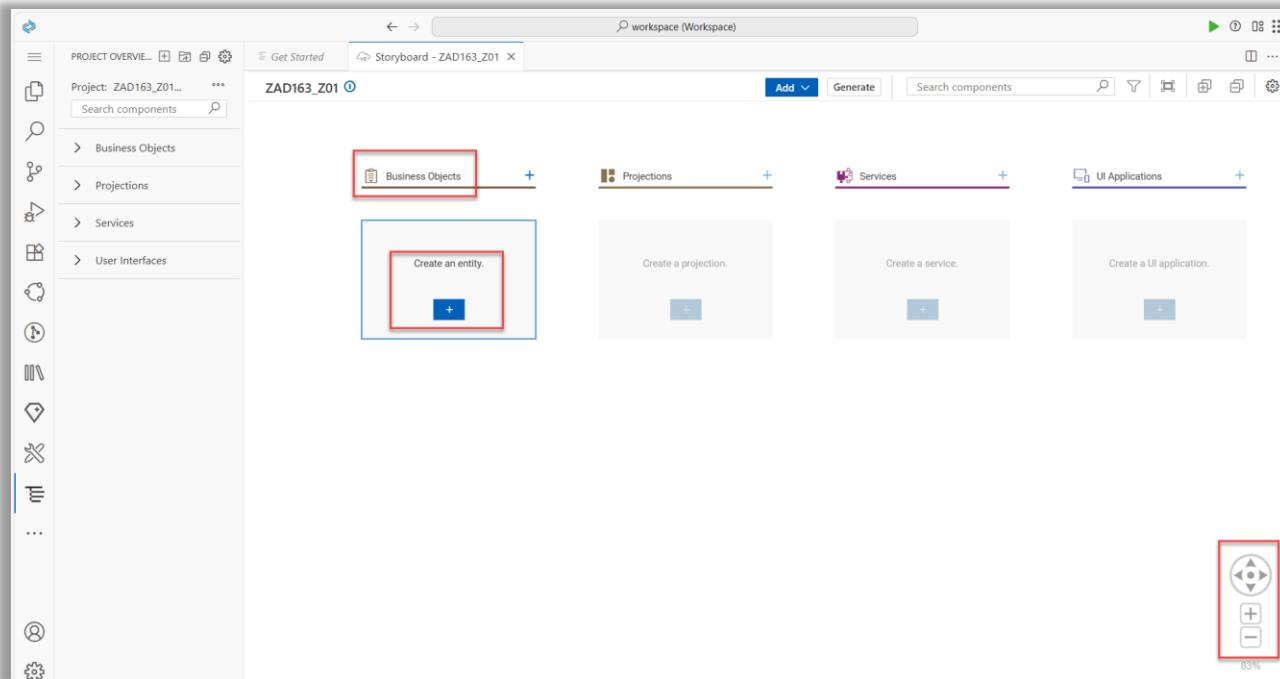
#### **in SAP Build Code**

**asUG Tech Connect**

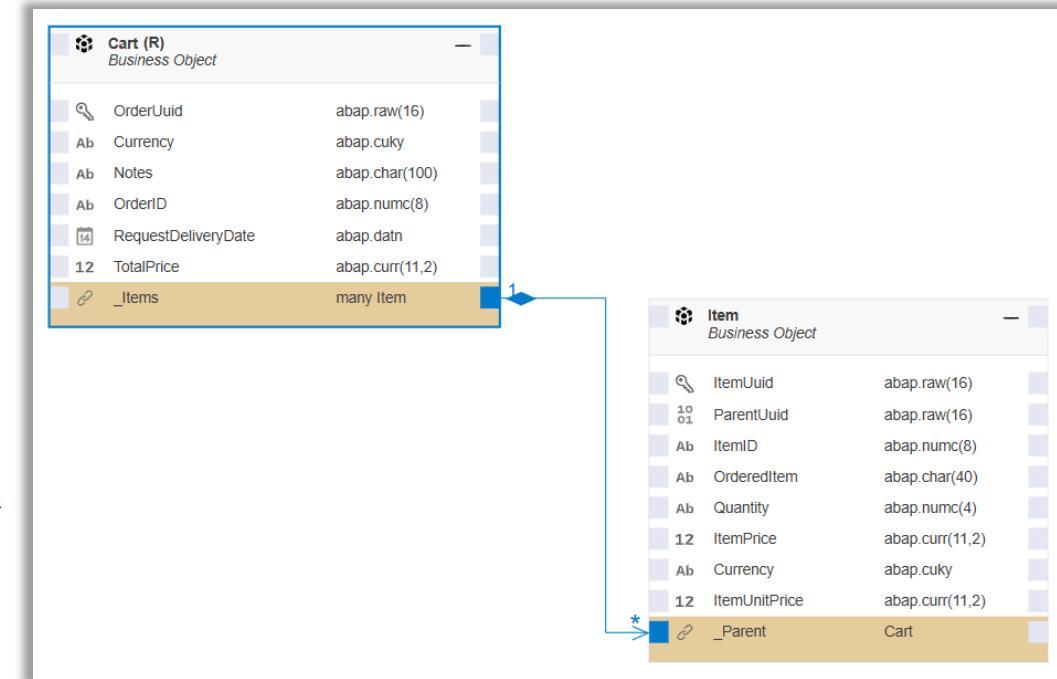
In partnership with **SAP TechEd**

# Exercise 2: Model the ShoppingCart Business Object

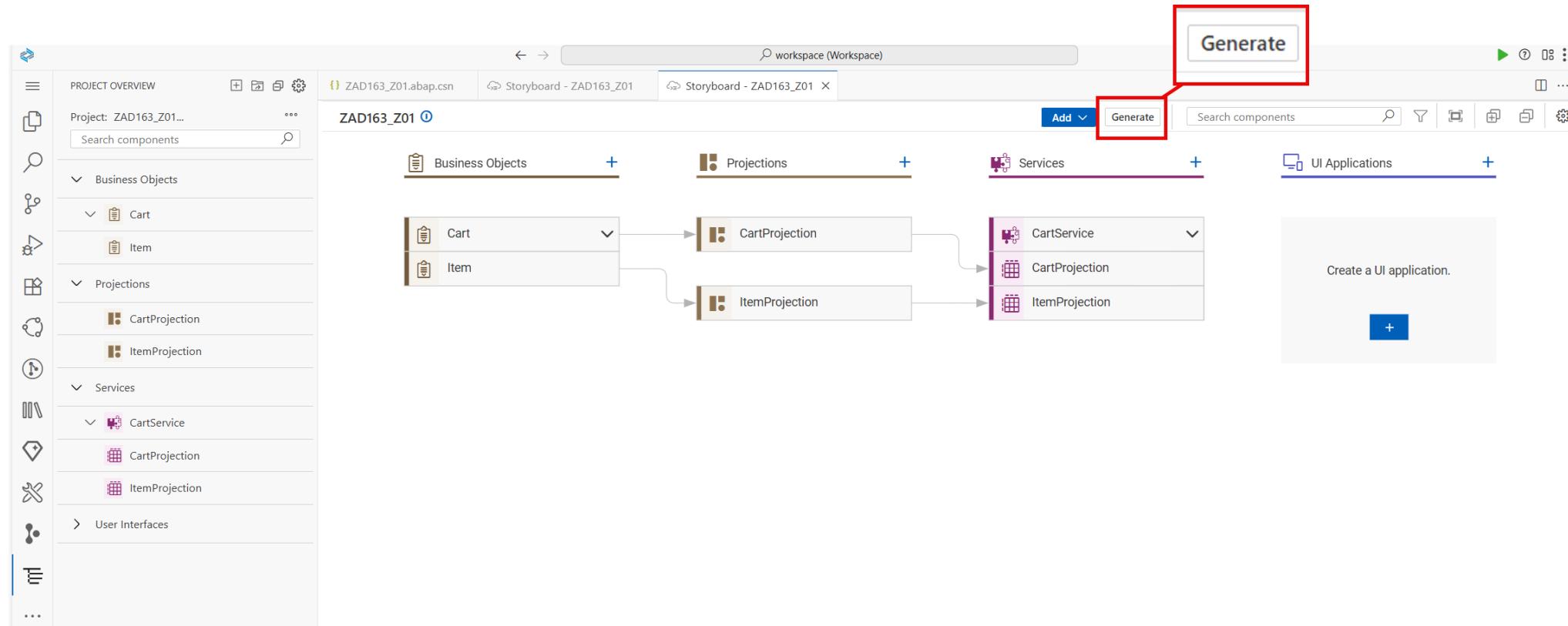
Storyboard



Graphical Modeller



# Exercise 2: Add Projection and Service Layer, Generate ABAP artefacts

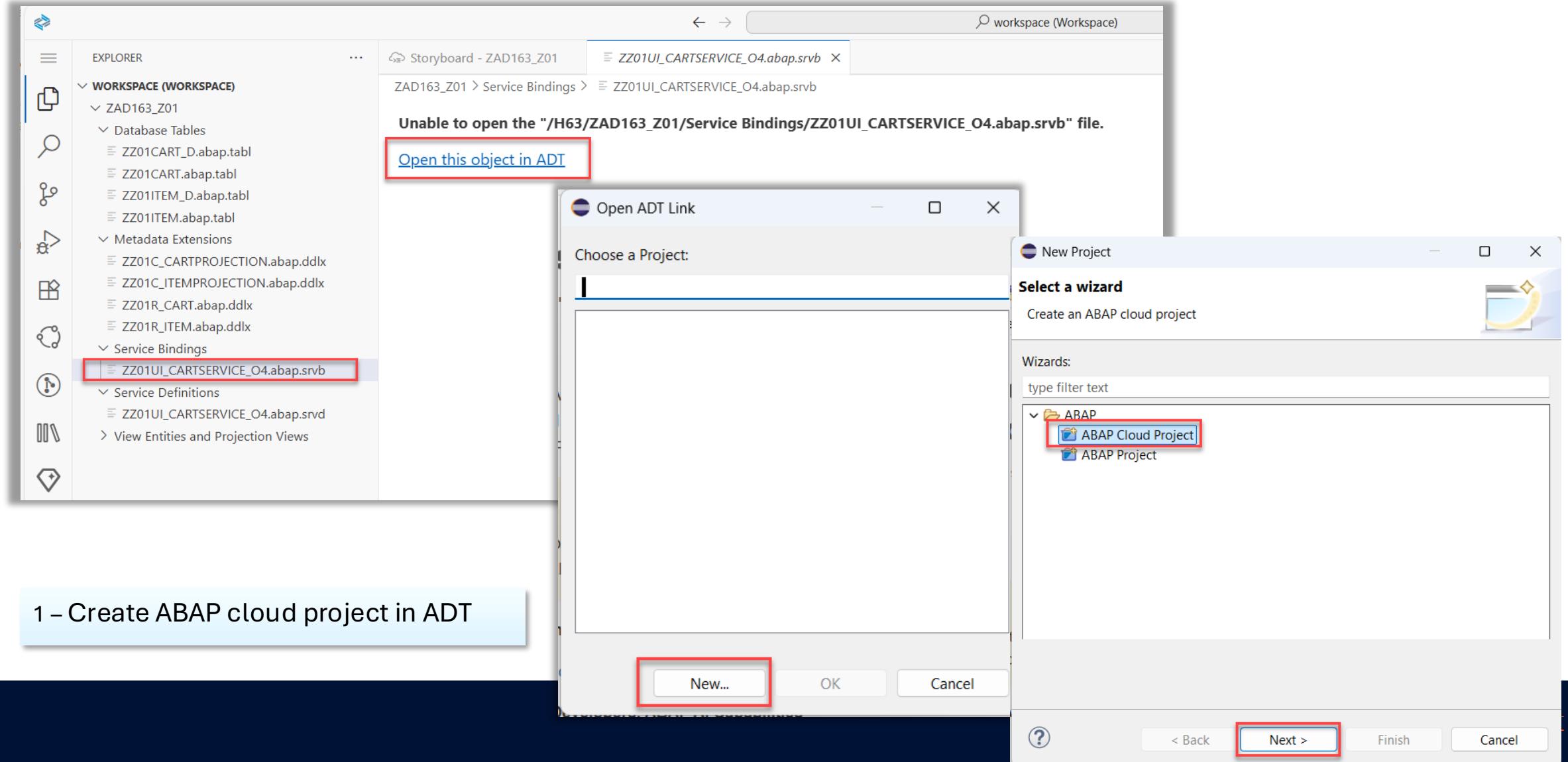


# Exercise 3: Implement transactional behavior in ADT, create validations and determinations

asUG Tech Connect

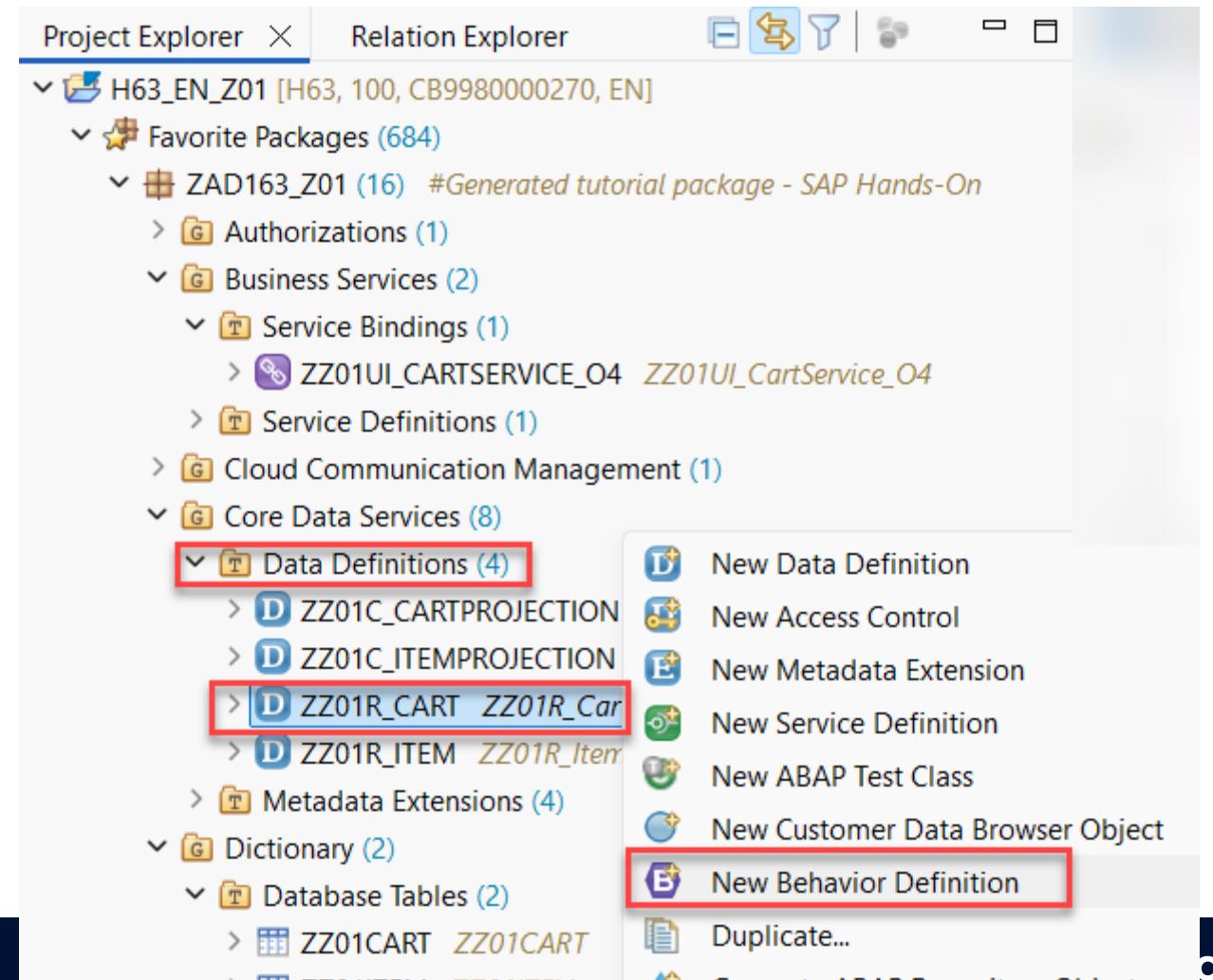
In partnership with SAP TechEd

# Exercise 3: Create ABAP cloud project in ADT



# Exercise 3: Add transactional behavior

```
1 @EndUserText.label : 'ZZ01CART'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zz01cart {
7
8     key client          : abap.clnt not null;
9     key order_uuid      : abap.raw(16) not null;
10    currency           : abap.cuky;
11    notes               : abap.char(100);
12    order_id            : abap.numc(8);
13    request_delivery_date : abap.datn;
14    @Semantics.amount.currencyCode : 'zz01cart.currency'
15    total_price         : abap.curr(11,2);
16    local_created_by    : abp_creation_user;
17    local_created_at    : abp_creation_tstmp;
18    local_last_changed_by : abp_locinst_lastchange_user;
19    local_last_changed_at : abp_locinst_lastchange_tstmp;
20    last_changed_at     : abp_lastchange_tstmp;
21
22 }
```



# Exercise 3: Implement determinations and validations using Joule

The screenshot shows the SAP Studio interface with the RAP Business Logic editor open. The code editor displays several lines of ABAP-like code:

```
35
36 validation validateRequestDeliveryDate on save { create; field RequestDeliveryDate; }
37
38 draft action Activate optimized;
39 draft action Discard;
40 draft action Edit;
41 draft action Resume;
42 //draft determine action Prepare;
43 draft determine action Prepare
{
44
45   validation validateRequestDeliveryDate;
46 }
```

Line 36 and the block from line 42 to 46 are highlighted with a red rectangle.

Below the code editor, a horizontal toolbar has several icons, including a plus sign, minus sign, and various navigation symbols.

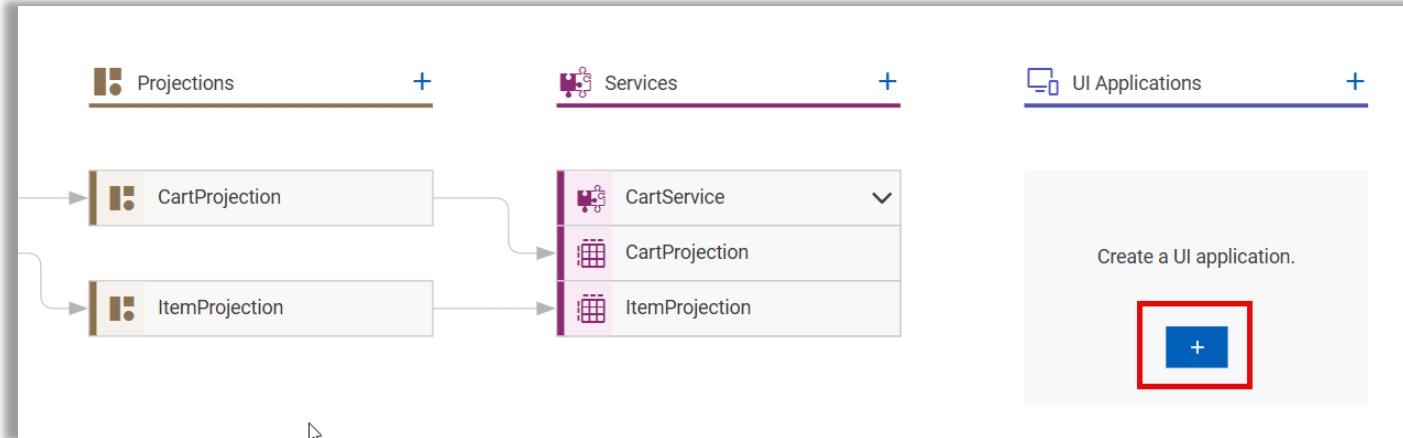
At the bottom of the screen, there is a Predict dialog box with the title "Predict RAP Business Logic". It contains a text input field with the placeholder "METHOD validateRequestDeliveryDate." and an "ENDMETHOD." button below it.

# Exercise 4: Create, preview and adapt a SAP Fiori elements application

asUG Tech Connect

In partnership with SAP TechEd

# Exercise 4: Create, preview and adapt a SAP Fiori elements application



### SAP Fiori generator

**Template Selection**  
Template: List Report Page

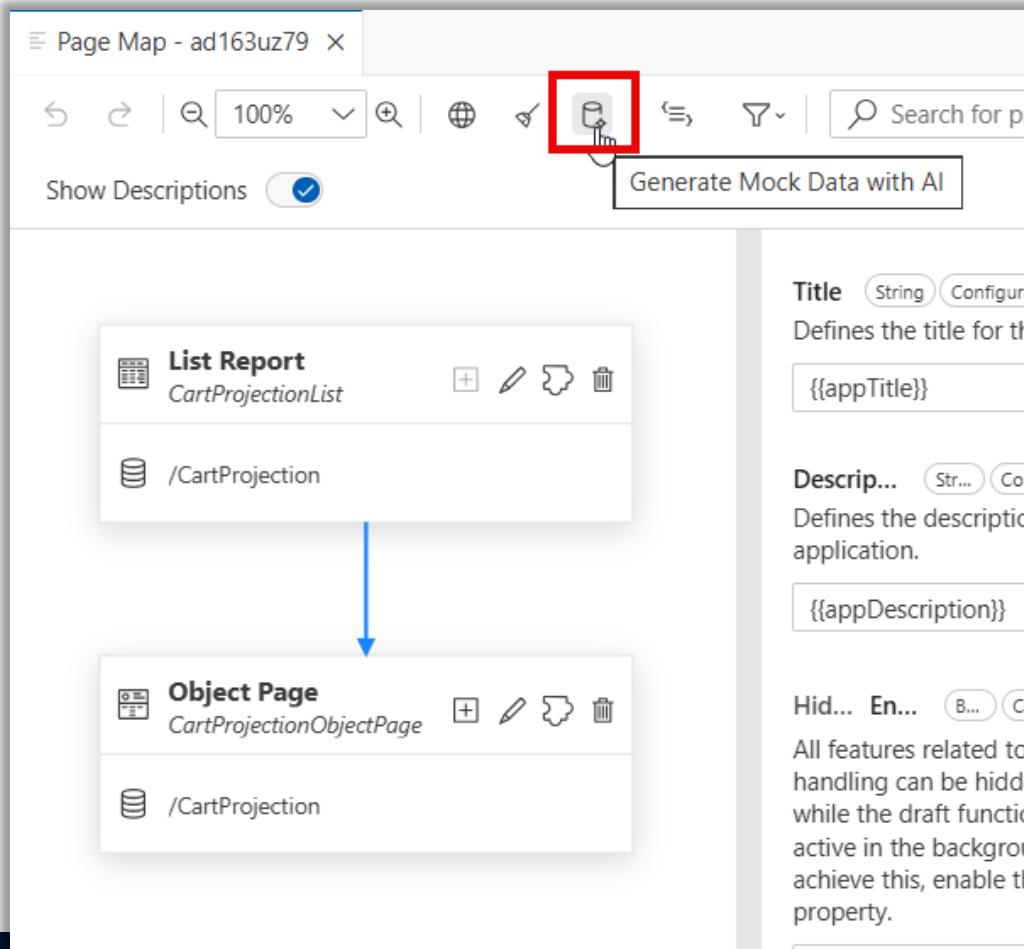
**Data Source and Service Selection**  
Data Source: Connect to a System  
System: H63 (BTP)  
Service: ZZ00UI\_CARTSERVICE\_O4 > ZZ00UI\_CARTSERVICE\_O4 (001) - OData V4

**Entity Selection**  
Main Entity: CartProjection  
Navigation Entity: None  
Generate Annotations: Yes  
Table Type: Responsive

**Project Attributes**  
Module Name: ad163uz00  
Application Title: Shopping Cart App AD163UZ00  
Application Namespace: Shopping Cart App AD163UZ00  
Description: A Fiori application.  
Project Folder Path: /home/user/projects  
Minimum SAPUI5 Version: 1.136.8 (Source system version)  
Enable TypeScript: No  
Add Deployment Configuration: No  
Add SAP Fiori Launchpad Configuration: No  
Use Virtual Endpoints for Local Preview: Yes  
Configure Advanced Options: No

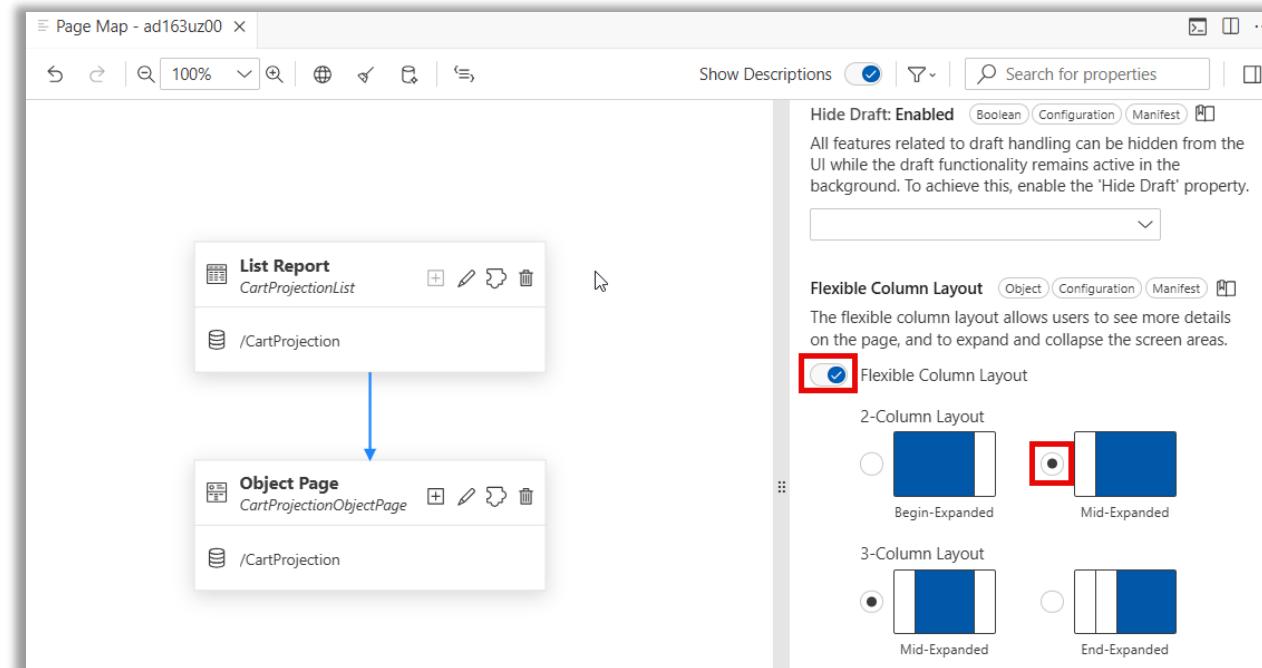
**Buttons**  
< Back **Finish**

# Exercise 4: Create, preview and adapt a SAP Fiori elements application



The screenshot shows the SAP Fiori Elements Page Map interface. At the top right, there is a button labeled "Generate Mock Data with AI" which is highlighted with a red box. Below the header, there is a "Show Descriptions" toggle switch that is turned on. The main area displays two components: a "List Report" and an "Object Page", both associated with the path "/CartProjection". To the right of these components, there are several configuration properties:

- Title**: String Configuration. Defines the title for the application. Value: {{appTitle}}
- Description**: String Configuration. Defines the description of the application. Value: {{appDescription}}
- Hide Draft**: Boolean Configuration. All features related to draft handling can be hidden while the draft functionality remains active in the background. To achieve this, enable the 'Hide Draft' property.



The screenshot shows the SAP Fiori Elements Page Map interface. On the right side, there is a configuration section for "Flexible Column Layout". A checkbox labeled "Flexible Column Layout" is checked and highlighted with a red box. Below this, there are two layout options: "2-Column Layout" and "3-Column Layout". Under "2-Column Layout", the "Mid-Expanded" option is selected and highlighted with a red box. Under "3-Column Layout", the "Mid-Expanded" option is also highlighted with a red box. The main area displays the same "List Report" and "Object Page" components as the first screenshot.



Queremos saber  
a sua opinião.

Acesse o aplicativo do  
SAP BTP Experience  
e responda a pesquisa de  
avaliação desta sessão.