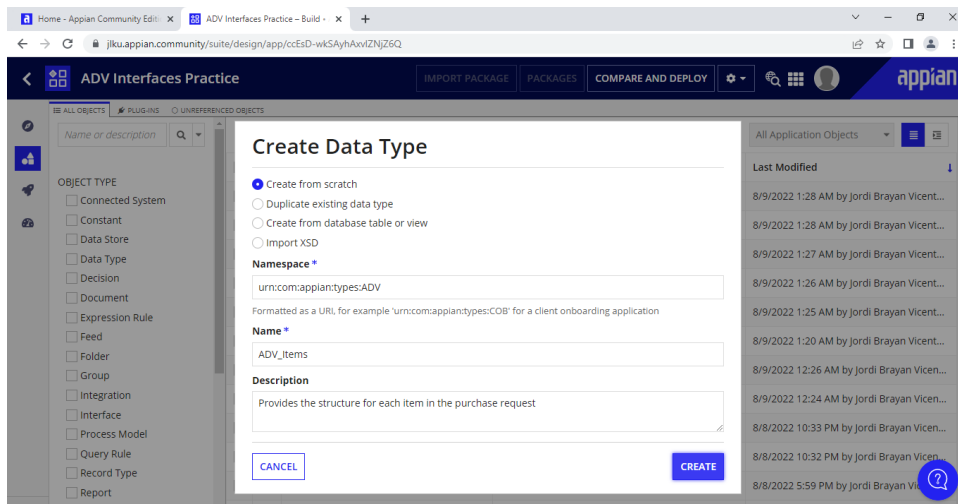
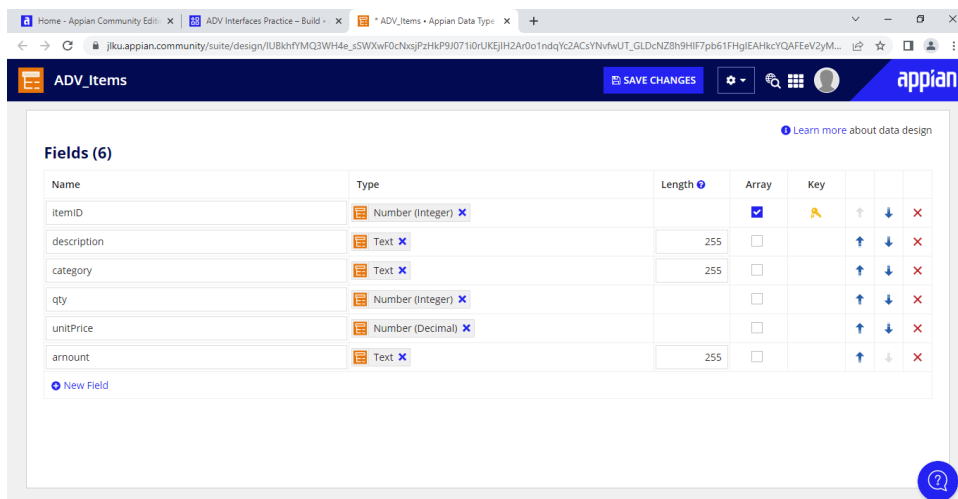


Se crea un nuevo Data Type llamado ADV_items



Se agregan los siguientes campos y sus respectivos tipos de datos



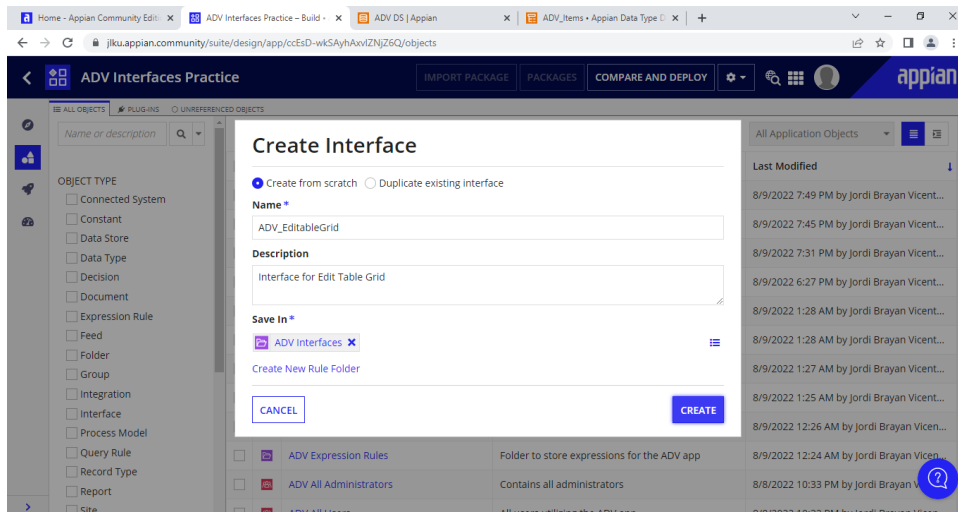
Se añade la entidad al Data Store

The screenshot shows the Appian Data Store configuration page. At the top, there are browser tabs for 'Home - Appian Community Edit...', 'ADV Interfaces Practice - Build -', 'ADV DS | Appian', and 'ADV_Items - Appian Data Type'. The address bar shows the URL: 'jku.appian.community/suite/dataStore.do?id=4949&version=-1&appian_environment=designer'. The main content area displays 'ADV_address' and 'ADV_items' with an 'Add Entity' button. Below this is a 'Schema Management' section with a checkbox for 'Automatically update database schema' and a 'Download DDL Script' link. A blue warning banner states: 'Entity mappings must be verified before publishing data store. The data store cannot be published until the mappings between its entities and the tables in the selected data source have been verified.' Below the banner is a 'VERIFY' button. At the bottom, there are 'SAVE & PUBLISH' and 'SAVE DRAFT' buttons.

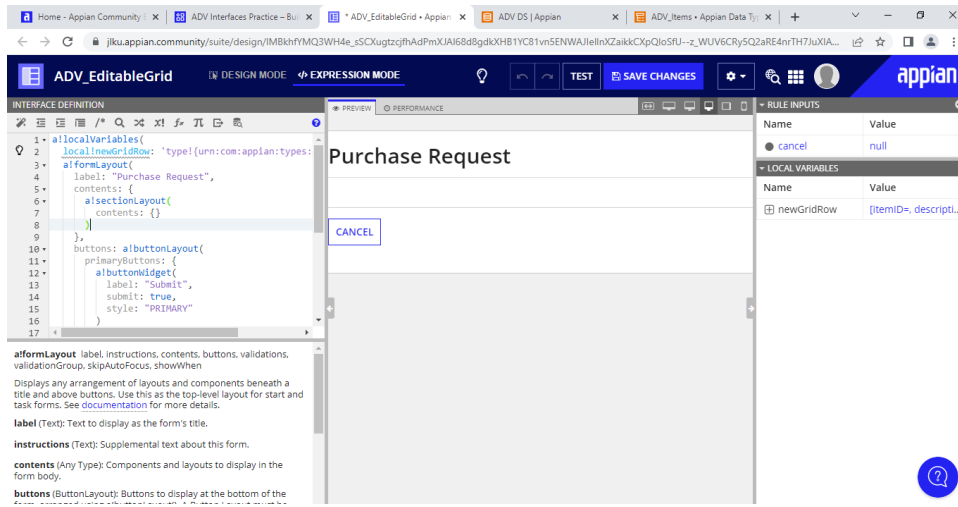
Se crea una nueva constante

The screenshot shows the 'Create Constant' dialog box in Appian. The dialog has two radio buttons: 'Create from scratch' (selected) and 'Duplicate existing constant'. The 'Name' field contains 'ADV_PURCHASE_REQUEST_CATEGORIES'. The 'Description' field contains 'Constant to hold the categories for the Purchase Request grid'. The 'Type' dropdown is set to 'Text', and the 'Array (multiple values)' checkbox is checked. The 'Values (4)' list includes 'Office Supplies', 'Hardware', 'Software', and 'Miscellaneous'. At the bottom, there are 'CANCEL' and 'CREATE' buttons. The background shows the Appian interface with a sidebar on the left containing 'OBJECT TYPE' (Constant, Decision, Expression Rule, Integration, Interface, Query Rule, Rule Folder) and 'LAST MODIFIED BY' and 'LAST MODIFIED ON' sections. The top right corner shows the Appian logo and a user profile icon.

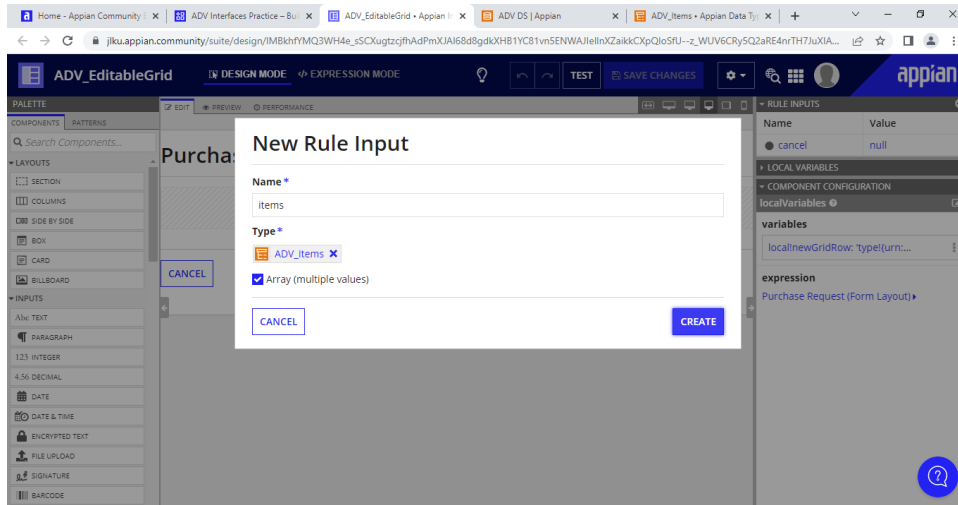
Se crea una nueva interfaz



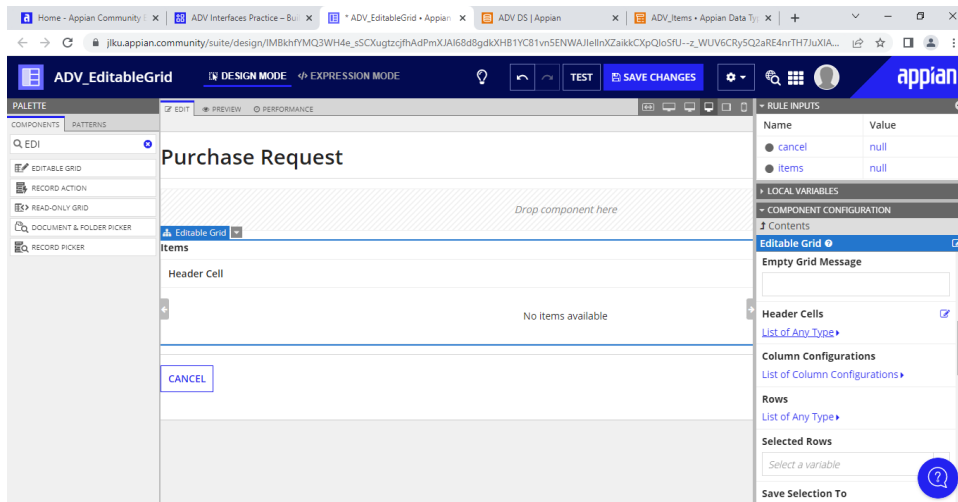
Se agregan componentes y se configura dentro del modo de expresión



Se crea un nuevo Rule Input



Se agrega el componente de editable grid



Se configura el header cell para la lista del item

The screenshot shows the Appian IDE interface in Design Mode. The main workspace displays the configuration for the 'Header Cells (List of Variant)' property of an 'Editable Grid' component. The configuration is an array of 'algridLayoutHeaderCell' objects, each with a 'label' property. The labels are: 'Description', 'Category', 'Qty', 'Unit Price', and 'Amount'. The 'algridLayoutHeaderCell' component is described as a label with properties for help tooltip, alignment, and show when. It is used to define a column header for an editable grid.

```
1 - {
2 -   algridLayoutHeaderCell(
3 -     label: "Description"
4 -   ),
5 -   algridLayoutHeaderCell(
6 -     label: "Category"
7 -   ),
8 -   algridLayoutHeaderCell(
9 -     label: "Qty"
10 -  ),
11 -  algridLayoutHeaderCell(
12 -    label: "Unit Price"
13 -  ),
14 -  algridLayoutHeaderCell(
15 -    label: "Amount"
16 -  )
17 - }
```

algridLayoutHeaderCell label, helpTooltip, align, showWhen
Defines a column header for use in an editable grid (algridLayout).
label (Text): Text to display as the column header.

Se configuran los contenidos dentro de Row para que coincidan los datos

The screenshot shows the Appian IDE interface in Design Mode. The main workspace displays the configuration for the 'Contents' property of a 'Row' component within an 'Editable Grid'. The grid is titled 'Purchase Request' and has columns for 'Description', 'Category', 'Qty', and 'Unit Price'. The 'Contents' property is configured with a list of components: 'Text (Text)', 'Dropdown (Dropdown)', 'Integer (Integer)', 'Decimal (Decimal)', 'Text (Text)', and 'Rich Text'. The 'Category' column is currently displaying a dropdown menu with the text '--- Select a Value ---'.

Purchase Request

Description	Category	Qty	Unit Price
	--- Select a Value ---		

Contents

- Text (Text)
- Dropdown (Dropdown)
- Integer (Integer)
- Decimal (Decimal)
- Text (Text)
- Rich Text

Se configura el item para agregar un foreach

The screenshot shows the Appian interface in 'EXPRESSION MODE'. A dialog box titled 'Item 1' is open, displaying a code editor with the following expression:

```
1+ {
2+   alforEach(
3+     items: rllitems,
4+     expression: agridRowLayout(↔)
5+   )
6+ }
```

Below the code editor, there is a description of the **alforEach** function: "alforEach Items, expression. Evaluates the provided expression once for every item and returns an array of the results. Items (List of Variant): Array or DataSubset containing the items over which to iterate." The dialog has 'CANCEL' and 'OK' buttons.

In the background, the 'ADV_EditableGrid' component is visible, showing a table with columns for 'Items' and 'Description'. The 'Items' column contains a list of items, and the 'Description' column contains a text field.

Se configura el Display Value Text

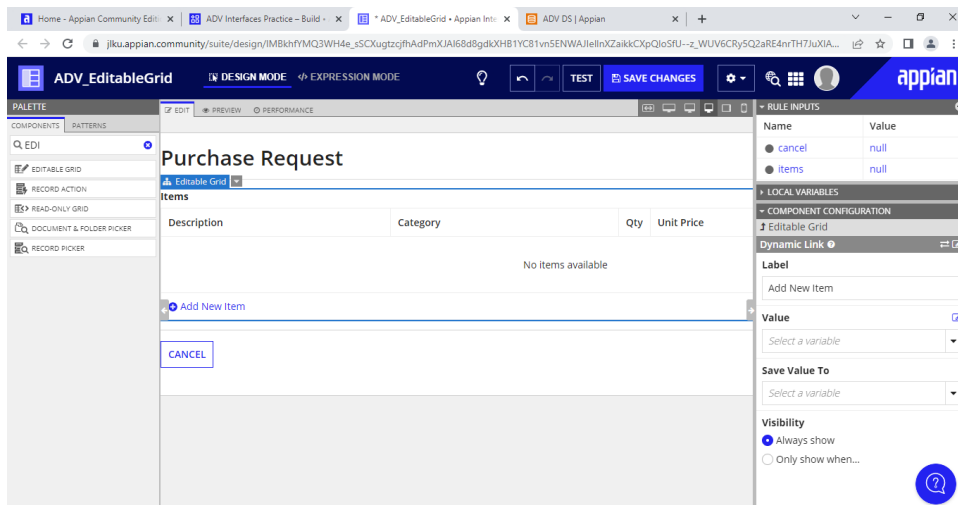
The screenshot shows the Appian interface in 'EXPRESSION MODE'. A dialog box titled 'Display Value (Text)' is open, displaying a code editor with the following expression:

```
1+ if(
2+   or(isnull(fvItem.qty), isnull(fvItem.unitPrice)),
3+   0,
4+   toInteger(fvItem.qty) * toDecimal(fvItem.unitPrice)
5+ )
```

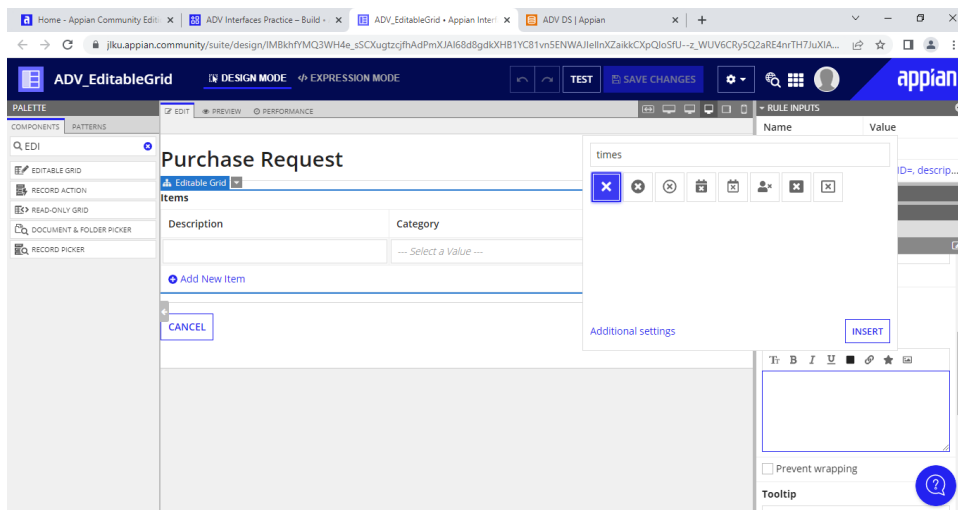
Below the code editor, there is a description of the **isnull** function: "isnull(value). Returns true if value is null, false otherwise. Returns: Boolean". The dialog has 'CANCEL' and 'OK' buttons.

In the background, the 'ADV_EditableGrid' component is visible, showing a table with columns for 'Items' and 'Description'. The 'Items' column contains a list of items, and the 'Description' column contains a text field.

Se agrega un componente para el Add New Item que a su vez aparecerá el componente de agregar uno nuevo



Se agrega un nuevo icono el cual podrá eliminar una nueva selección



Se configura el link dinámico

The screenshot shows the Appian Studio interface with a 'Select Link' dialog box open. The dialog box is titled 'Select Link' and contains two sections: 'NAVIGATION' and 'ACTION'. The 'NAVIGATION' section includes icons for RECORD LINK, USER RECORD LINK, PROCESS TASK LINK, REPORT LINK, NEWS ENTRY LINK, and SAFE LINK. The 'ACTION' section includes icons for DYNAMIC LINK, START PROCESS LINK, SUBMIT LINK, DOCUMENT LINK, and AUTH LINK. The 'DYNAMIC LINK' option is highlighted with a blue box. The background shows the Appian IDE interface with a component palette on the left and a configuration panel on the right.

Se configura el Save Value To de la lista de guardar

The screenshot shows the Appian Studio interface with a 'Save Value To (List of Save)' dialog box open. The dialog box contains a text area with the expression: `1 * {alsave(rllitems, remove(rllitems, fvlindex))}`. The background shows the Appian IDE interface with a component palette on the left and a configuration panel on the right.

Home - Appian Community Edit... ADV Interfaces Practice - Build - ADV_EditableGrid - Appian Inter... ADV_DS | Appian

jkku.appian.community/suite/design/IMBkhfYMQ3WH4e_sSCuqtzcfhAdPmXIAI68d8gdXh81YCB1vn5ENWAJlelnXZaikCXPQloSFU--z_WUV5CRy5Q2aRE4nrTH7uXIA...

ADV_EditableGrid DESIGN MODE EXPRESSION MODE TEST SAVE CHANGES

ADV DS | Appian

Purchase Request

Items

Description	Category	Qty	Unit Price	Amount	
	-- Select a Value --			0	X

+ Add New Item

CANCEL SUBMIT

?