DETAILED TRAINING GUIDE

Procurement Center UI and Logic

Smart Vendor & Procurement System
Part 1 | Page layout and request overview

Build a Procurement Center for viewing request summaries and entering purchase requests. Then configure item-selection logic to retrieve asset information and populate a line-total field from the selected asset purchase price.

1 Prepare the main content area

Open the Procurement Center page in UI Builder. Keep the existing application navigation on the left and establish the main page container in the blank content area on the right. Arrange the content vertically so the header, summary cards, request details, and request form can be reviewed from top to bottom.

2 Add the page header

Place a container at the top of the content area. Add the heading **Procurement Center** and supporting text **Create and manage purchase requests.** Adjust spacing and alignment to group these elements into a clear introduction.

3 Build the summary card row

Add a container below the description and divide it into columns. Configure the first card's dimensions, spacing, typography, and alignment in the properties panel. Reuse this structure for the other cards to maintain a consistent row.

Property	Setting
Total Requests	Example display value: 145; add a supporting icon
Pending	Example display value: 24; add a warning-style icon
Approved	Example display value: 98; add the corresponding status icon

Treat these counts as design examples. For any remaining cards in the design, update the label, value, icon, and status treatment while keeping the shared dimensions and spacing.

4 Review the overview layout

Check that the title and description align with the summary row, that card contents have sufficient space, and that icons and values remain consistently positioned across the row.

Request Details and Entry Fields

5 Build the request details area

Add a container underneath the summary cards. Arrange the request identifier, title, status, and supporting information into structured columns. Keep each detail readable rather than combining the whole request into one text block.

Use the example identifier **PR-2024-0145** and title **Office Furniture for Floor 3** in the corresponding fields. Adjust field widths and spacing so the example record fits the intended layout.

6 Add request actions

Place the required action buttons alongside the request information. Group them visually with the relevant request and refine their dimensions, spacing, and appearance through component properties.

7 Create the purchase request form

Add a section titled **Create Purchase Request** with the description **Fill in the details to create a new purchase request.** Inside it, add a **Request Information** section and organize the input components into columns.

Property	Setting
Request Title	Input for the request title
Vendor	Vendor field with the optional indicator
Department	Department field
Cost Center	Cost center field
Priority	Priority field
Requested By	Requester field

Use the properties panel to control field width, height, spacing, and alignment. Maintain consistent spacing between rows and keep each label associated with its input.

8 Add Justification

Below the request fields, add a larger text input for the request justification. Allow enough height for a longer explanation and separate it clearly from the fields above.

Line Items Summary and Approval

9 Create the Line Items section

Below Justification, add a container with the heading **Line Items**. Insert a table structure for individual items included in the purchase request. Adjust the column widths so the item information remains organized.

10 Configure the item controls and row

Add the required line-item input controls and place an **Add Item** button on the right side of the entry row. Adjust its size and spacing to fit within the table layout.

Create the first line-item row with input cells aligned beneath the corresponding table headers. Review the dimensions and spacing of each input through the properties panel.

11 Add the Summary section

Place a **Summary** section on the right side of the purchase request form. Add its required fields beneath the heading. Use the column layout to keep Summary separate from Request Information while keeping both sections visible together.

12 Build the Approval area

Create a section titled **Approval**, including the **Executive Approval** portion of the workflow. Add the approval information fields and status controls shown in the design. Keep their styling consistent with the rest of the form.

13 Add the primary action

Place the **Submit for Approval** button below the approval information. Adjust its width, spacing, and appearance so it is identifiable as the form's main action.

14 Save and review the interface

Review the complete page in order: header, summary cards, request details and actions, Request Information, Justification, Line Items, Summary, and Approval. Check alignment and spacing, then save the page.

The interface now contains the request-entry and approval controls. The following logic section configures selected-item retrieval and the line-total display.

Capture Items and Retrieve Asset Data

Part 2 | redSling Logic Builder

15 Open the item selection event

Open Logic Builder for the completed Procurement Center. In the component tree, select the procurement item input-select component and open its selection-change event. This event starts the flow when the user changes the selected items.

16 Store the selected options

Add **Store Response Data in Variable**. Create and save a variable for the selected-option data. Store the options returned by the component in this variable so subsequent blocks can access them.

17 Process each selected item

Add a **For Each** block and use the selected-options variable as its input array. Inside the loop, store the current item. Create variables for the asset information needed by the flow, including the asset ID and asset name.

18 Prepare the backend request

Retrieve the existing backend base URL and required authentication token. Add **Make API Call** and configure the URL using the backend base URL and the asset-related route. Store the response in a variable for further processing.

Property	Setting
Content-Type header	Use the required JSON content type: application/json
Authorization header	Connect the authentication token variable
Asset filter	Configure the required condition against SmartVendorWeb_asset data
Request payload	Connect the required key to the current asset information
Response	Store the returned asset data for the next operation

19 Review the asset request

Check that the current selected item supplies the asset information used in the request and filter. Review the URL, headers, payload, and stored response together. Use the configured asset route and field names from the project when completing these settings.

Populate the Line Total and Verify

20 Review the related data models

Open the data model area and review the asset and purchase-order models and fields. Use their field structure to confirm the mapping between the backend response and the procurement form, then return to Logic Builder.

21 Extract the purchase price

Use **Get Property of** to retrieve **purchase_price** from the returned asset record. Store the result in the purchase-price variable.

22 Convert and store the numeric value

Create a variable named **linetotal**. Pass the purchase-price variable through **To Number**, then assign the converted result to linetotal. This makes the value numeric before it is displayed.

In this configured flow, linetotal receives the converted purchase price. Quantity multiplication, tax calculations, and aggregate totals require additional logic if they are needed.

23 Update the form field

Add **Set Property of**, select the corresponding procurement form field, and set its **Value** property using **linetotal**. This connects the retrieved price to the line-total display.

24 Review the complete flow

1. The item selection changes.
2. The selected options are stored in a variable.
3. For Each processes the selected options.
4. The current item supplies the relevant asset information.
5. The backend request retrieves the matching asset record.
6. purchase_price is extracted and converted to a number.
7. The numeric result is assigned to linetotal.
8. The target form field receives linetotal as its Value.

Final verification

Preview the page and check item selection, the matching asset response, and the value displayed in the target field. When multiple items can be selected, verify the row or field targeted by each loop iteration. Review the UI controls separately from the selected-item logic before extending the purchase-request workflow.