regoUniversity
SAN ANTONIO • 2026

Building Custom Solutions and Apps with Studio

Your Guides:
Jair Flores and David Zywiec

Sponsored by
ValueOps™ by Broadcom | Clarity™ by Broadcom | Rally® by Broadcom | ConnectALL by Broadcom | Insights by Broadcom

Agenda

- Introduction
- Basic Configuration, Modern UX Configuration and Navigation
- Attribute and Object Administration

Objects, Attributes, and Views/Fields

Objects

- Objects are the major functional components of Clarity
- Objects define the attributes (fields), subpages (links), page layout, and views that make up your configured instance of Clarity
- In addition to the stock objects delivered with the system, you can create custom objects. Custom objects are essentially tables inside the database that begin with “ODF_CA”
- Use the default objects or create custom objects and sub-objects to manage information for specific business needs
- Once you create an object, add attributes, links, and actions and set up the views

Objects

Each object has four distinct pieces you can configure

- Properties
- Attributes
- Links
- Views

Things to remember

- You can only delete Custom Attributes
- Attributes need an API Attribute ID to show in the New UX
- Adding more than 100 custom attributes to a single custom object may impact performance
- A hierarchy with a maximum of three levels of objects can be created, and allow child objects to inherit properties and access rights from parent objects

Objects: Types

- Stock Objects
  - Primary Standard Objects
    - Project
    - Task
    - Team
    - Resource
    - Company
    - Application
- Custom Objects
  - Master Objects
  - Sub-Objects

Objects: Investment Object

- The Investment object is a special component that provides the ability to define attributes one time and share them across select OOTB objects
- These objects “inherit” attributes from the Investment object:
    - Project, Idea, Other Work, Application, Asset, Product, Service
    - Modern UX Custom Investments
- Streamlines the creation process and ensures consistency across objects
- You may re-label attributes on shared objects if needed (ID remains the same)
- Attributes defined at the investment level are available to the stock objects noted above but are not required
- You must make updates to Investment attributes **at the Investment level**

Attributes

- Attributes are the fields on an object that store information
- The attributes of each object are available on the Attribute screen within the object
- Many attributes are delivered out-of-the-box, but you can create additional attributes using Studio
- Once created, you can organize and place attributes:
    - **Classic UI:** on views and portlets and use for reporting
    - **Modern UX:** Views and Blueprints
    - Example: “Start Date” is an attribute of the project object

Details on the various data types can be found in the Studio Developer Guide

Attribute Data Types

When creating attributes, the following data types are available:

- String (2000-character maximum)
- Large String – Plain Text
- Large String – Rich Text
- Number
- Calculated
- Money (includes currency code)
- Boolean (checkbox)
- Date
- Lookup (related lookup needs to be available - create prior to creating attribute)
- Multi-Valued lookup (same note above applies)
- Attachment
- Time-varying
- URL (Links to actual data)
- **Rich Text (Only available in MUX)**

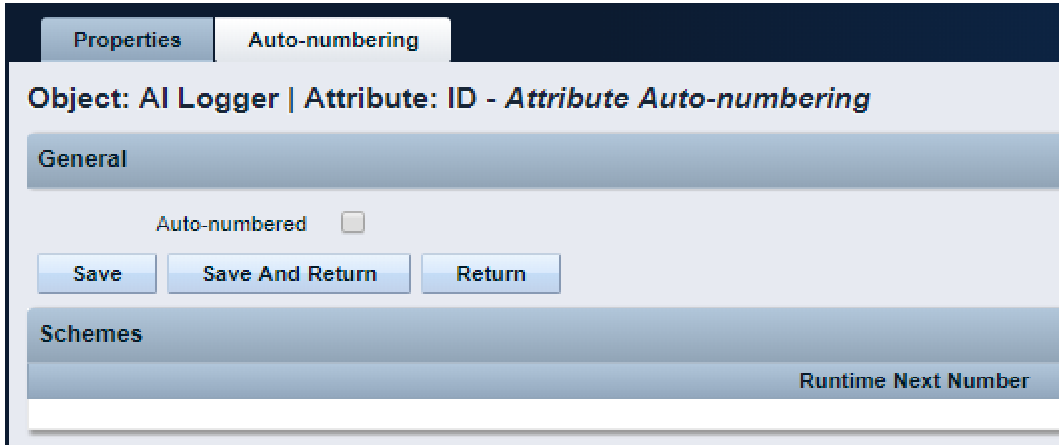
Calculated Attributes

- A Calculated attribute displays a dynamic, read-only value
- Values are calculated from other existing attribute values
- Values are calculated every time the user accesses or refreshes the page
- These values **are not stored** in the database
- Calculated attributes can only read the following data types:
  - **Number**: Use this data type to calculate a number value like a sum or average
  - **String**: Use this data type to concatenate two or more text values
    - Example: concatenate the value of the attribute “created_by” and the constant “2007” to produce a result of “ssmith 2007”
  - **Date**: Use this data type when you need to calculate dates using basic arithmetic or to provide the current date

Note: You cannot delete source attributes used in a calculated attribute!

Attributes: Auto-Numbering

- Clarity provides the ability to create your own numbering/naming scheme for object instances (PRJnnnnn, APPnnnnn, etc)
- The scheme can be numeric or a mixture of characters and numbers
- Two out-of-the-box attributes that are commonly auto-numbered are “Name” and “ID”
- Configuration is done via the auto-numbering tab on the attribute detail

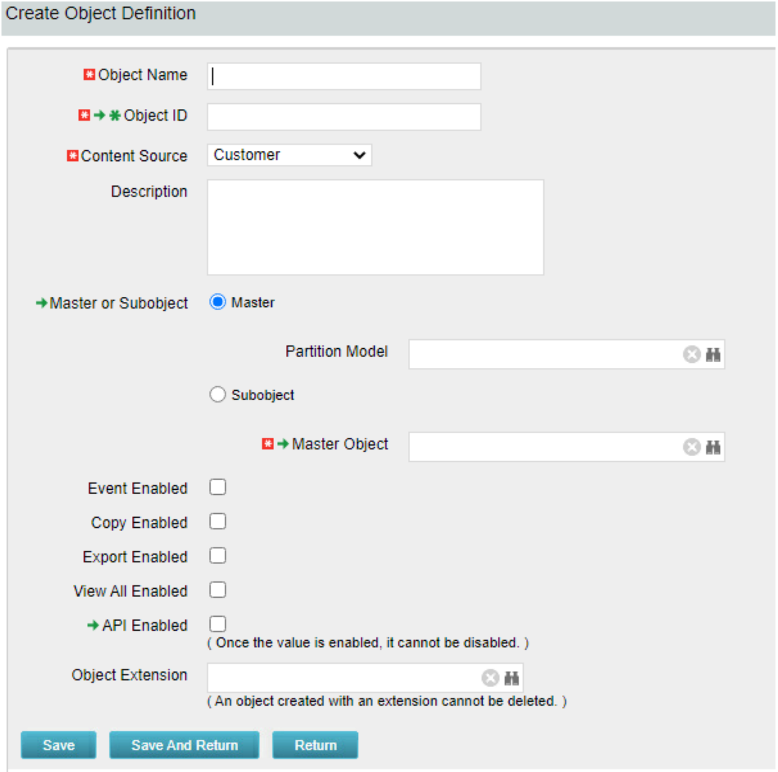


Attributes: Exercise

- Create a new custom sub-object to the project object
  - Administration -> Objects
  - Click New and fill out the required fields

Object Name: Meaningful name summarizing the object function
Object ID: Use a standard naming convention; many start with a customer ID + "_" (special characters and spaces should not be used in IDs)

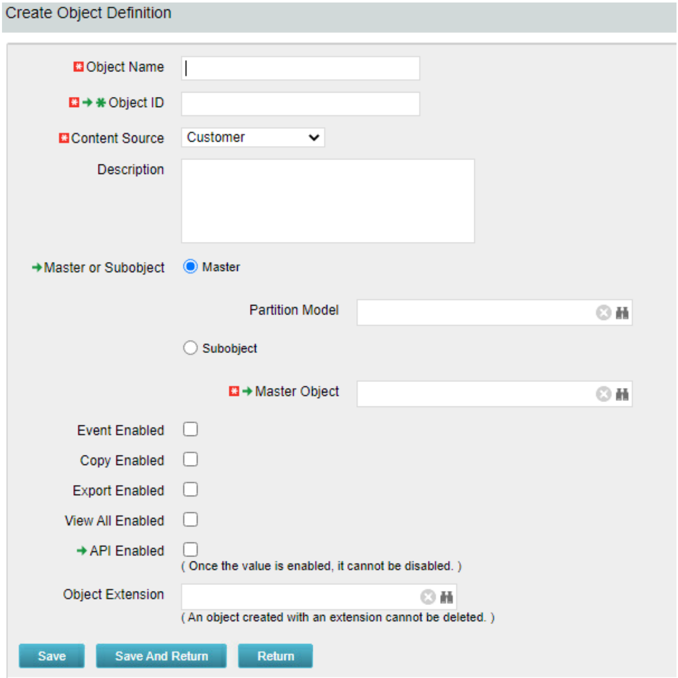
Master or Subobject
- Choose **Master** if object is to be standalone
- Choose **Subobject** if object is to be accessed via another object (1 to many relationship)
  - Choose the Master object by using the browse button



Attributes: Exercise

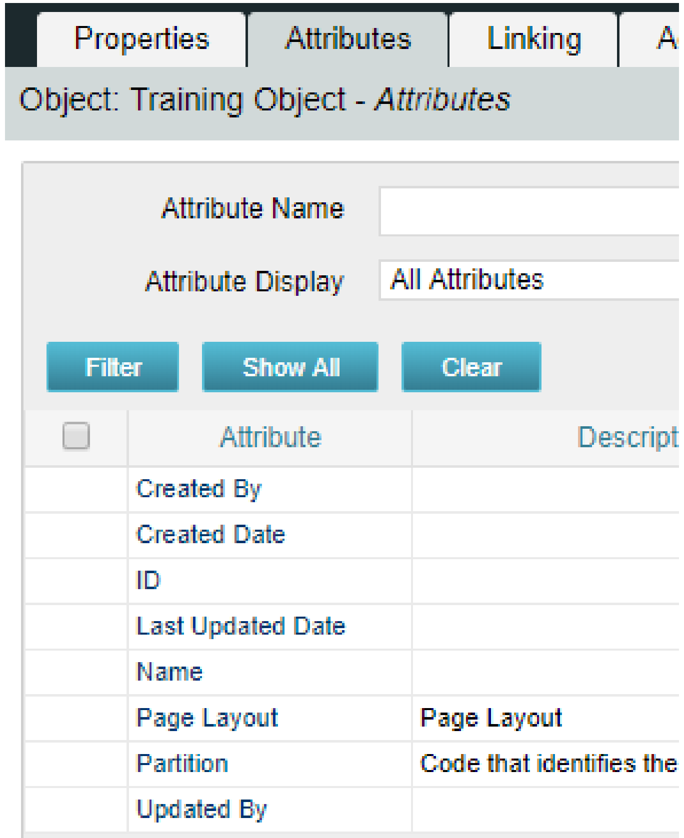
Select the following checkboxes if they apply:

- **Event Enabled:** Specifies that the process engine is notified of object instances that are created or updated. (If a process needs to get driven off the object).
- **Copy Enabled:** Specifies that copies can be made of the object instances.
- **Export Enabled:** Specifies that object instances can be exported to XML.
- **View All Enabled:** Specifies that the object instances can have a view containing all properties, subobject lists, and page portlets that can be personalized on a single page.
- **API Enabled:** Allows custom objects that are a sub object of projects/investments to show in the New UX.
- **Include in Datawarehouse:** Enabled the object to be available in the DWH.
- **Object Extension:** Used to create custom Investments, by associating to the Investment Object.



Attributes: Exercise

Notice the default fields included in the newly created object



- **Created By and Created Date**: Keep track of who and when the record was created.

- **Last Updated By and Updated By**: Keep track of the last person who updated the record and when.
- ** Note: Outside of custom objects there are OOTB jobs / processes that will skew the results of the last updated by and date fields, as the application often makes updates to the record.

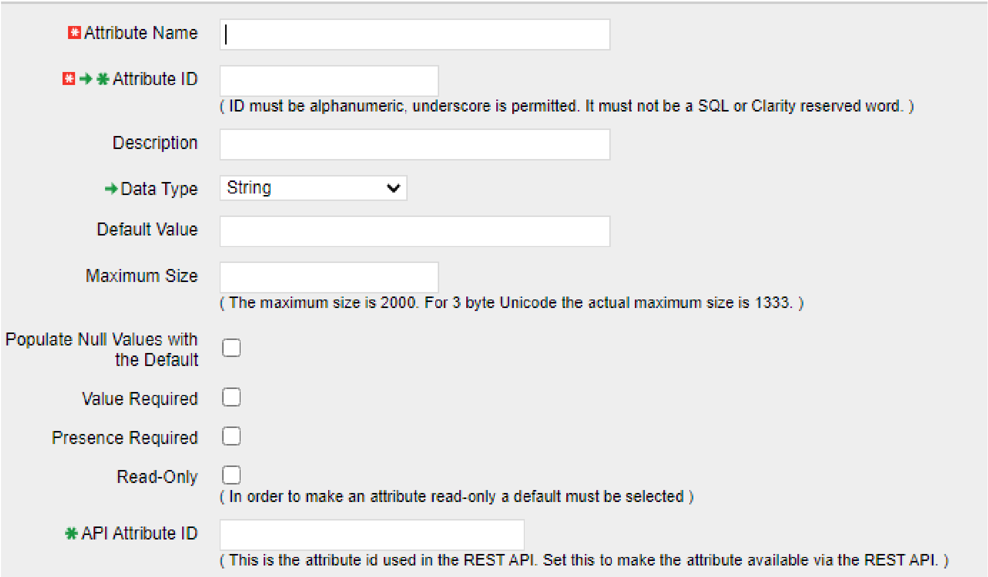
- **Page Layout:** Each object defaults to a standard layout with tabs such as Properties, Processes, and Audit. This can be customized by adding a new custom page layout. (Details later on)

- **Name and ID:** Used to identify the record; Name can be repeated multiple times while the ID has to be unique. Auto-numbering is often used to force that uniqueness and standardization.

Attributes: Exercise

Add 3 or 4 attributes of different varieties to your new custom object

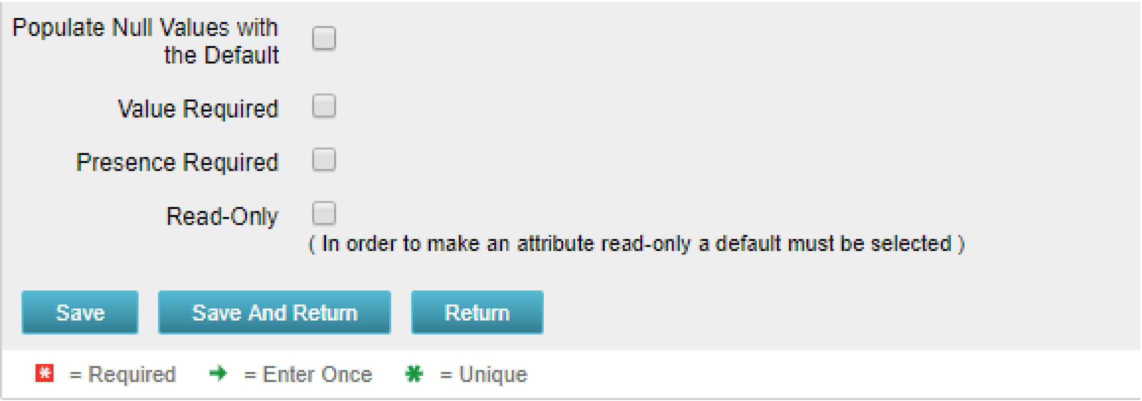
- Click New and fill out required fields as well as Data Type


Attribute Name
Attribute ID
(ID must be alphanumeric, underscore is permitted. It must not be a SQL or Clarity reserved word.)
Description
Data Type String
Default Value
Maximum Size
(The maximum size is 2000. For 3 byte Unicode the actual maximum size is 1333.)
Populate Null Values with the Default
Value Required
Presence Required
Read-Only
(In order to make an attribute read-only a default must be selected)
API Attribute ID
(This is the attribute id used in the REST API. Set this to make the attribute available via the REST API.)

- **Attribute Name:** Name of attribute (display name can be changed later in fields if need be)
- **Attribute ID:** Unique ID for attribute (spaces and special characters not allowed)
- **Description:** Meaningful explanation for attribute usage
- **Data Type:** Type of attribute
- **Lookup:** Only shows up for “Lookup” and “Mutli Valued Lookup” types
- **Default:** Default value to populate attribute with (if applicable)
- **API Attribute ID:** Required for field to show in New UX. Only available if Object is API Enabled

Attributes: Exercise

Select the following checkboxes if they apply:



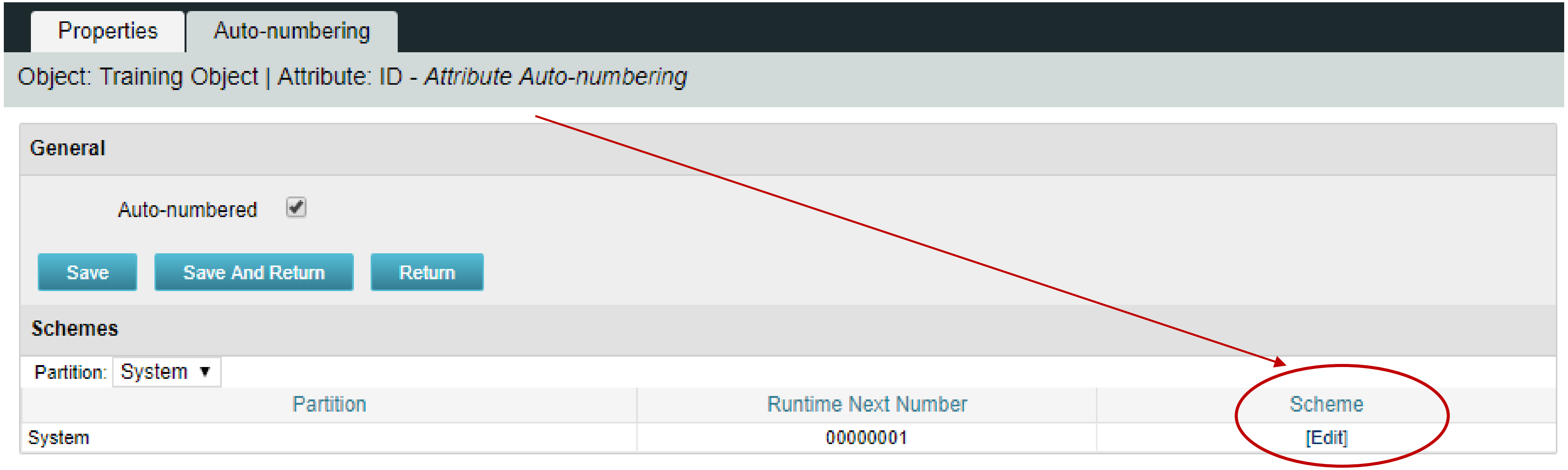
- **Populate Null Values with the Default**:

  If an attribute is added after instances have already been created, this will populate the existing records with the default value.
- **Value Required**: Specifies whether a value is required for the attribute.
- **Presence Required**: Specifies that the attribute always appears in the Edit Properties view.
- **Read-Only**: Specifies that a user cannot make changes to the value in the attribute.
- **Include in the Datawarehouse:** Include the field in the DWH schema.
- **Attribute API Alias:** Alias to enable the attribute through REST API.

Attributes: Exercise

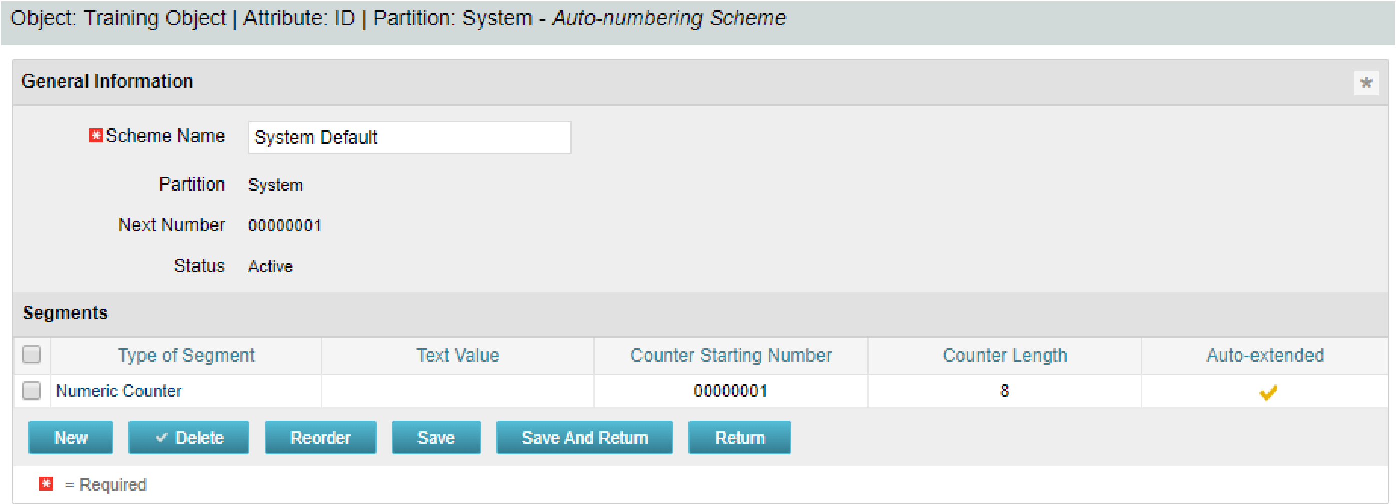
Auto-number the ID attribute

- Select the auto-numbering tab
- Check the "Auto-numbered" box and click "Save"
- Select Scheme -> Edit

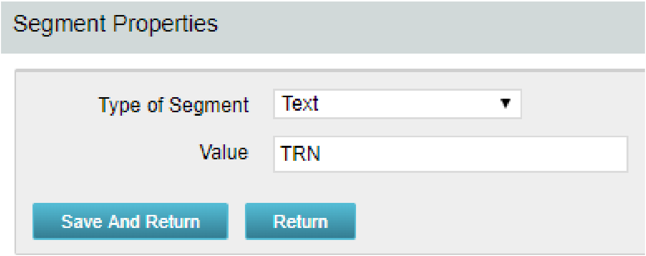


Attributes: Exercise

- The default segment type is numeric, but this can be modified to include text characters as well.

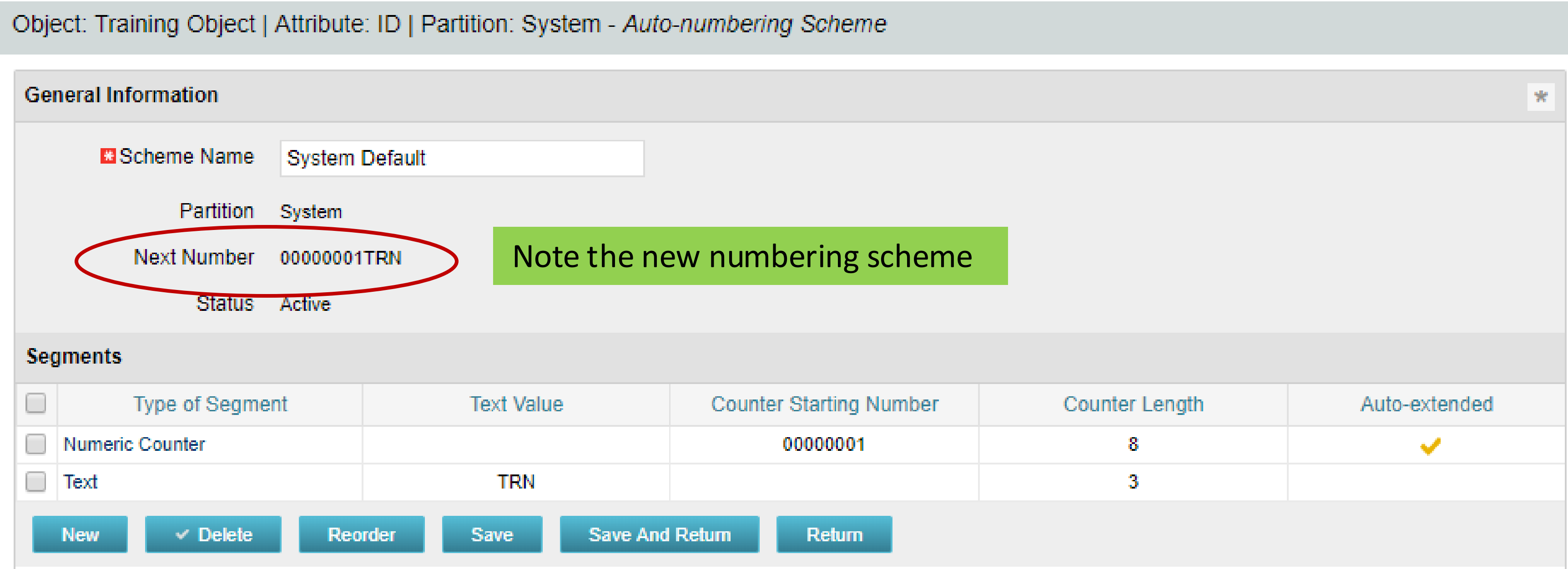


- Select “New” and set Type of Segment = “Text” and type the text value into “Value” field.

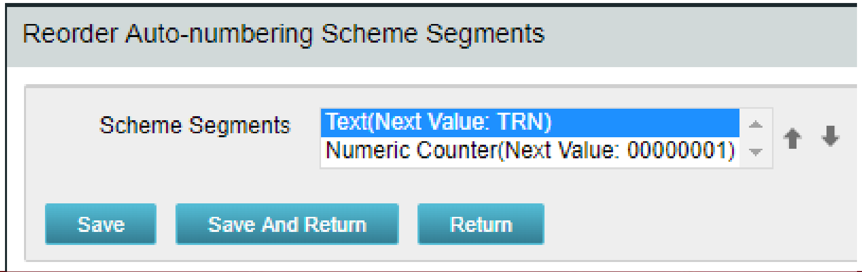


- Click Save and Return.

Attributes: Exercise



- The new scheme defaults into the order it was entered. To reorder to have the text in front select “Reorder” and move the segments accordingly.
- Select “Save and Return.”



Studio and Modern UX

Administration in the Modern UX

The Modern UX Administration is more functional than technical. It can be managed by any user with a good functional understanding of Clarity.

While some administration activities are done in the Classic Clarity UX, others are available only in the Modern UX.

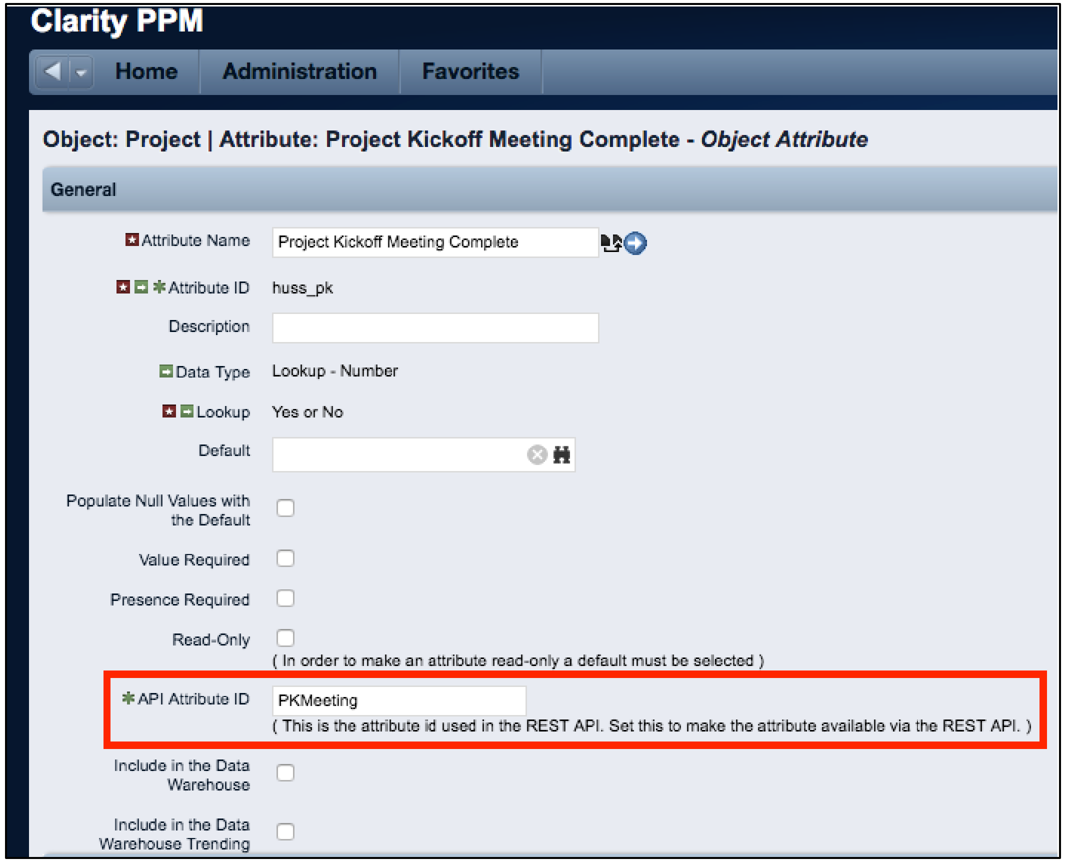
Administrators must have a good understanding of what they want to expose and to whom they wish to make functionality available to make best use of the Classic and New functionality.

Studio Objects and Attributes are available in the Modern UX, they only need an API Field ID to be configured.

Enabling Attributes for the Modern UX

- For custom attributes to be able to be viewable/editable in the Modern UX, their API Attribute ID must be populated.

- Access your custom attributes on the object to which they belong, and populate this value to ensure the attributes may be utilized in the Modern UX.
    - Go to Administration -> Objects -> Select the appropriate object
    - Select Attributes Tab
    - Select Desired Attribute
    - Populate API Attribute ID and **Save**.



Note: API-Enable OBS attributes for them to be utilized in filters, views, etc.

Basic Components Modern UX

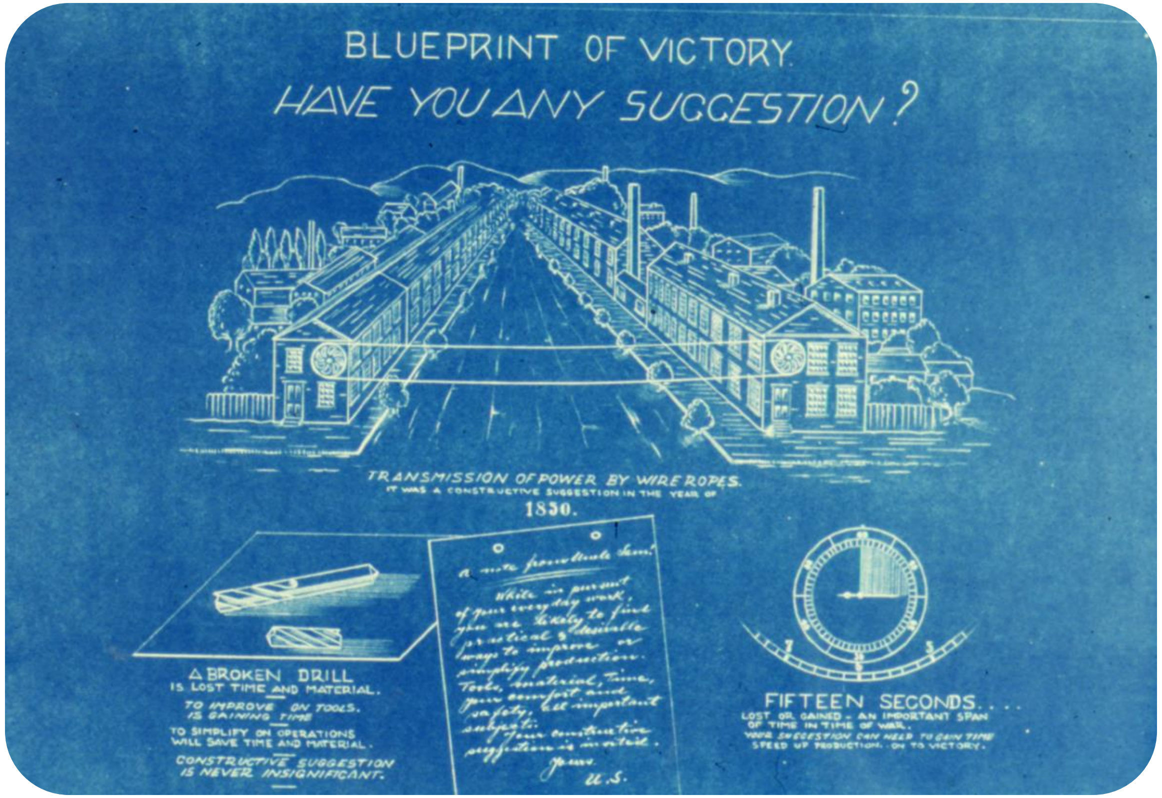
- Blueprints
  - Modules
  - Views
  - Channels
  - Actions
  - Business Rules
- Field Level Security
- Pages (Dashboards)
- User Personalization
  - Views
  - Picklists

Blueprints

- Blueprints
  - Modules
  - Views
  - Channels
  - Actions
  - Business Rules

Overview

- Blueprints are configurable layouts in the New User Experience
- Blueprints can be used (might vary depending on version):
    - Projects
    - Ideas
    - Custom Investments
    - Custom Objects
    - Hierarchy
    - Roadmaps
    - Agreements
    - Pages



Overview

- Depending on the Blueprint type options might vary, but in general Blueprints can:
  - Allow admins to customize the look and feel of Edit Views, Add Modules, Add Channels
  - Impact how certain investments are created from templates
  - Implement Rules to enforce validations like required fields in Objects and Subobjects (16.0.3)
  - Configure Actions to run processes within MUX

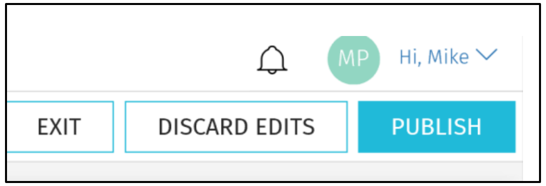
Blueprint Configuration

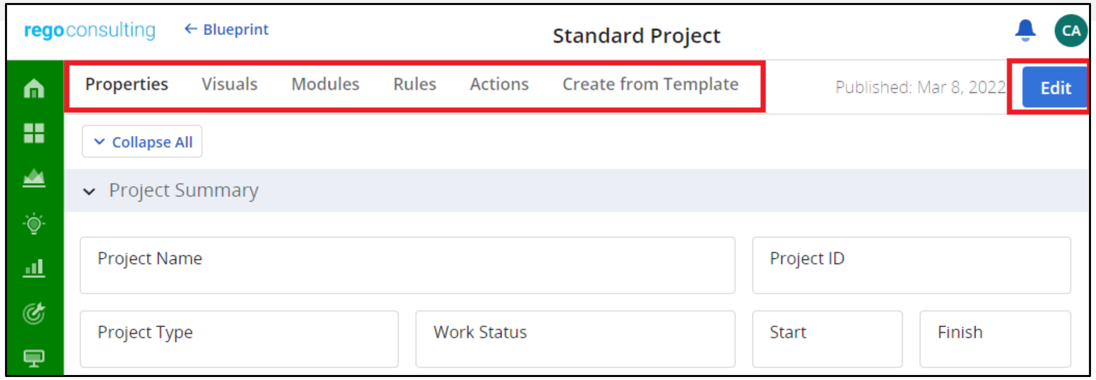
Blueprint Properties

- Using a Project Blueprint as an example, the Properties view will allow you to configure the following. **(We will go more in-depth on these topics later)**

- **Properties** – Defines the properties pane of the object. Admins can configure "Sections" and "Attributes" by dragging, dropping and resizing accordingly.
- **Visuals** – Define the icons on the Project Tiles view. (Only available for Project Blueprints.)
- **Modules** – Supporting "tabs" that can be added or removed from the Blueprint. The modules include functionality like Financials, Teams, Risk, Issues, Changes, Custom Sub-Objects and/or Channels.
- **Rules** – Create business rules to control how end users are engaged, when a rule is met it will perform actions like showing or hiding fields/sections.
- **Actions** – Create actions to allow end users to execute a process, admins can control when this actions are available.
- **Create from Template** – Similar to the properties page, allows admins to add or remove fields for the create from template properties view.

- By default, the Blueprint is in "View Only" mode
- Clicking on "Edit" mode will allow admins to perform changes and publish afterwards

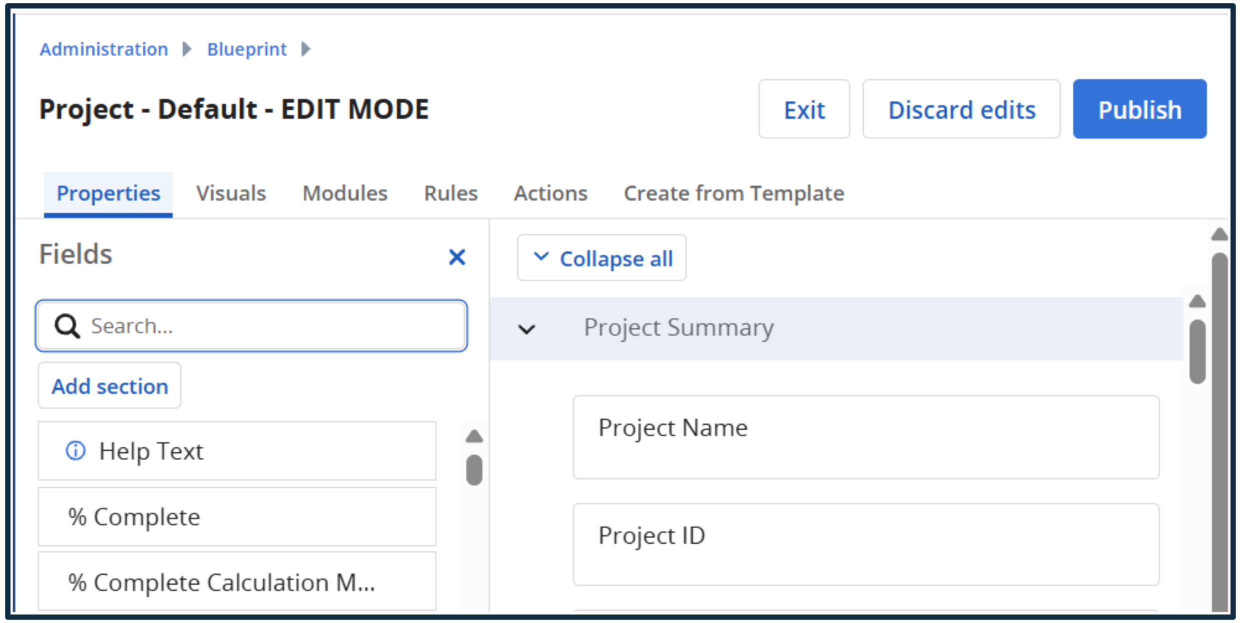




Properties

Fields Pane

- Provides list of fields that can be added:
  - By default, it contains a list of out-of-the-box Project and Investment fields
  - Custom fields can be added (as long as the field has an API Alias)
  - Also includes Help Text that can be used multiple times in the Properties tab to provide different Help Text
- Fields that already exist on your Blueprint are greyed out



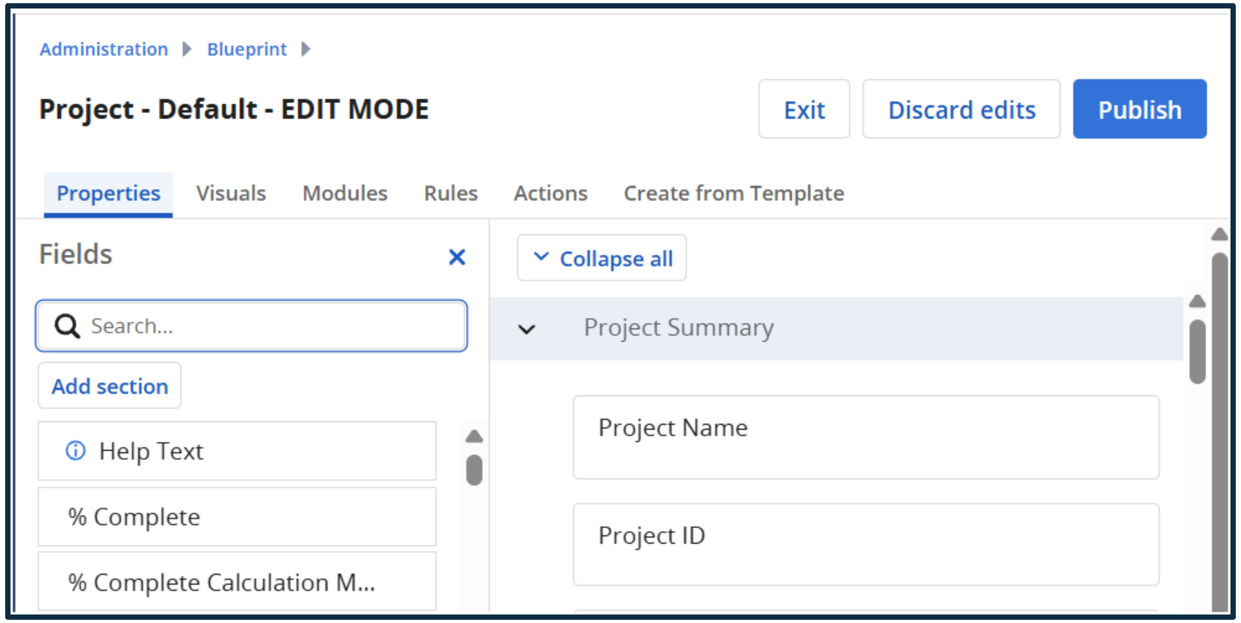
Note: The following attribute type is not compatible with the New User Experience:

- Custom Time-Scaled Value (TSV)

Properties

Layout

- Click the **Add Section** button to add a new section to the Details
- Add or move a field by simply dragging and dropping the field into a section
- Remove a field by clicking the **X** in the top right-hand corner of the field
- Resize the field by dragging the bottom right-hand corner of the field



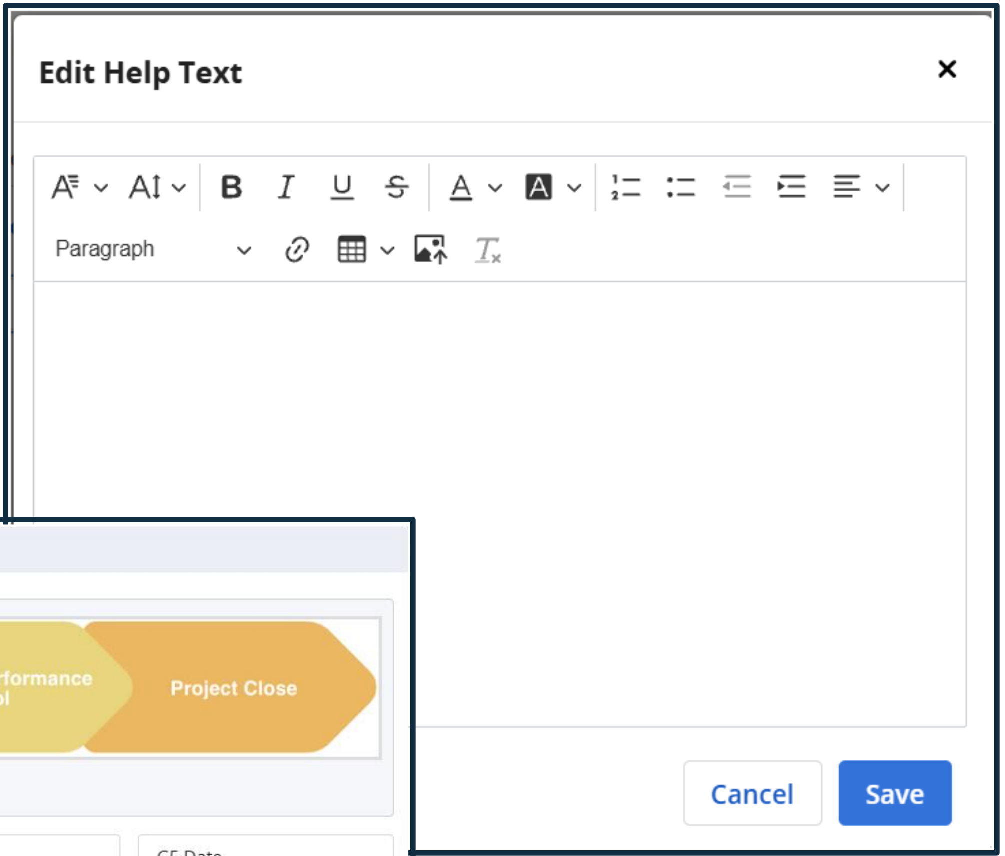
Note: The following attribute type is not compatible with the New User Experience:

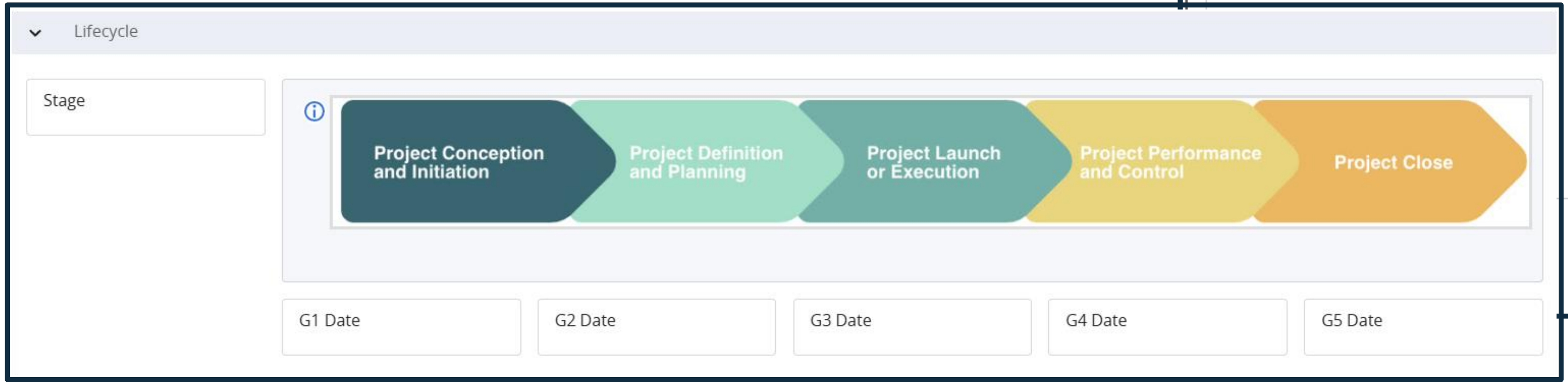
- Custom Time-Scaled Value (TSV)

Help Text Screen

Includes various options such as

- Rich Text Formatted help content
- Links
- Tables
- Pictures
- And more!

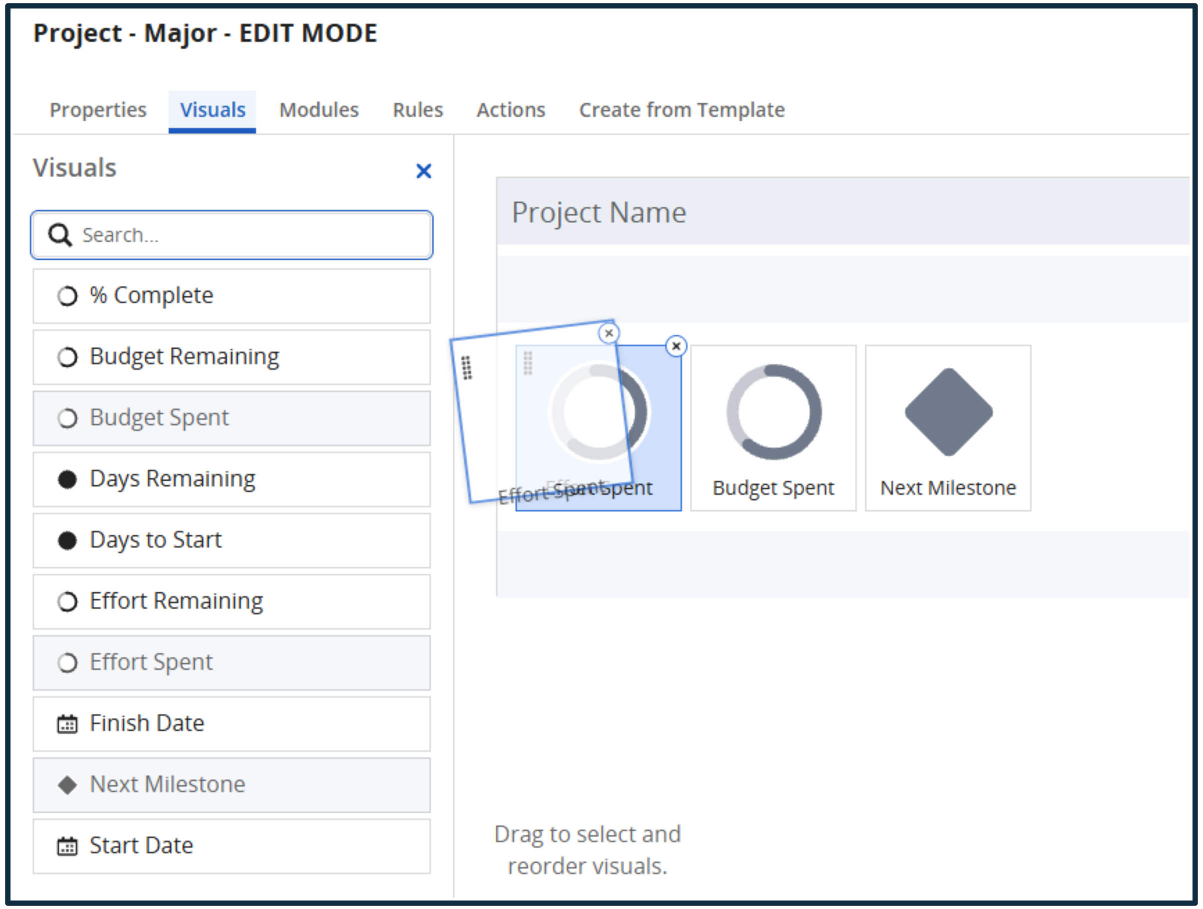




This examples aligns the image of stage gates with the corresponding date fields

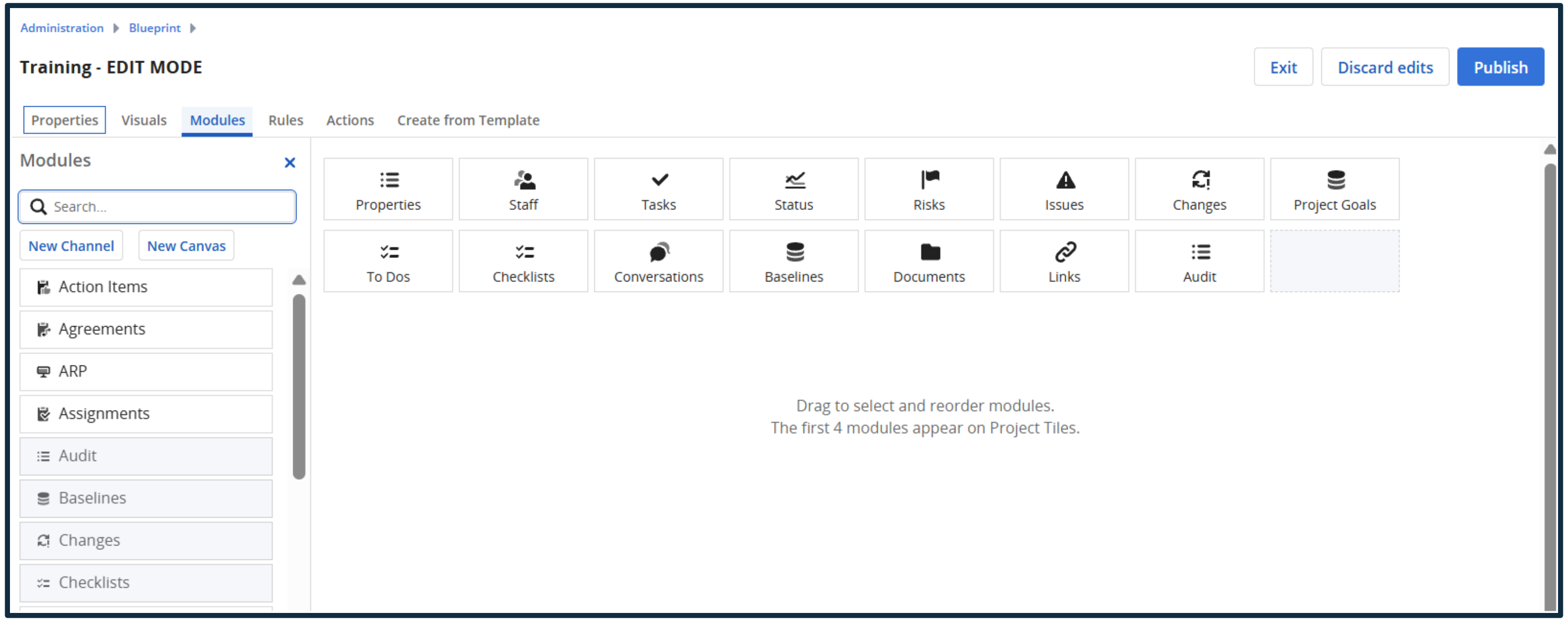
Visuals (Projects Only)

- **Visuals** are displayed on the Project Tiles view
- Currently, there are 10 out-of-the-box Visuals to choose from
- Add or move Visuals by simply dragging and dropping
- Remove a Visual by clicking the **X** in the top right-hand corner of the icon
- Links for the first 4 configured modules will be displayed below the visuals



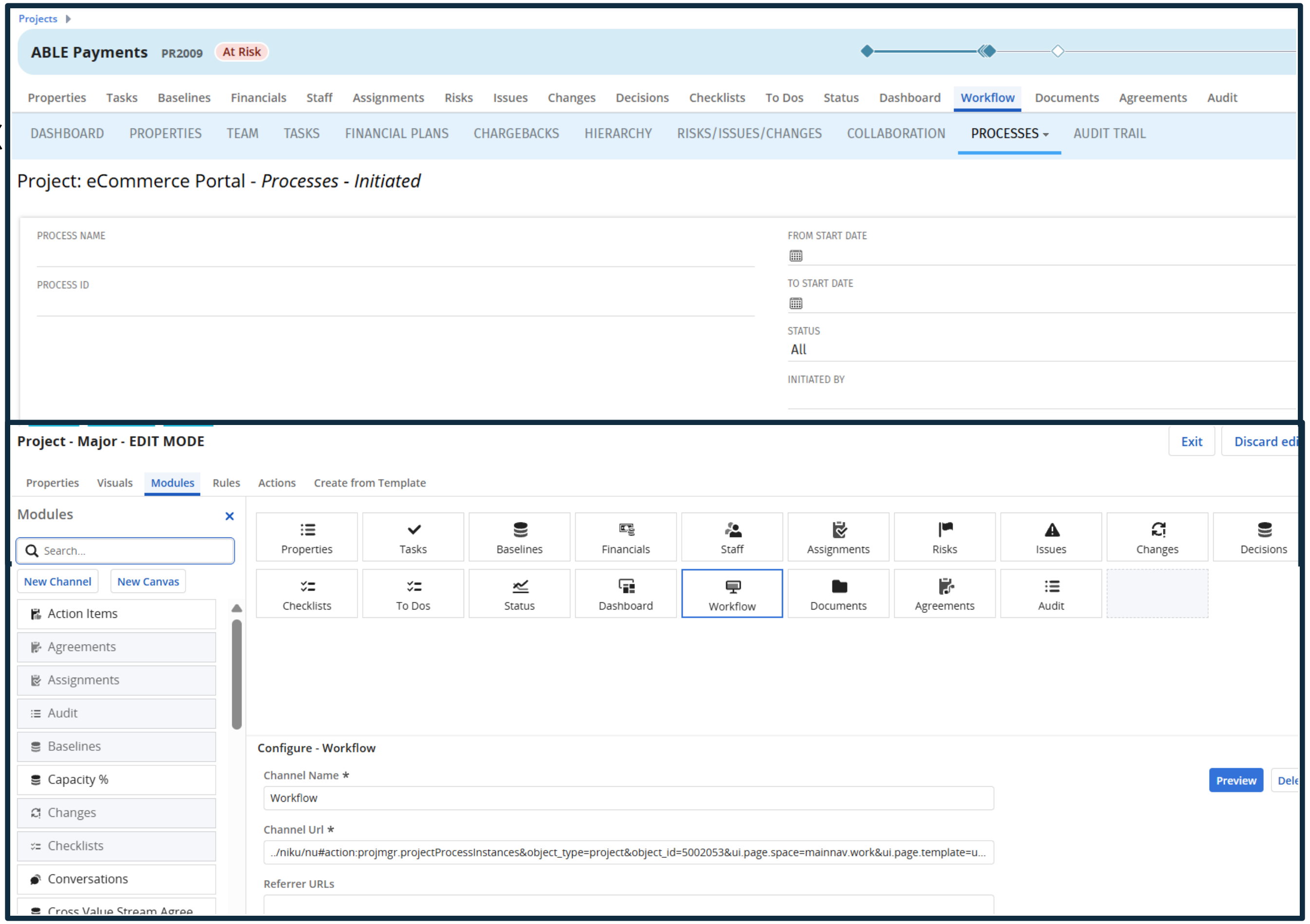
Modules

- Configured **Modules** are displayed across the top tabs for end users as they navigate into Properties view
- There are several core Modules which are not configurable and provide functionality like project financials, To-Dos, etc.
- In addition to the core Modules, there are configurable:
  - Channels
  - Ability to add custom sub-objects (only if the sub-object is API Enabled)



Modules – Channels

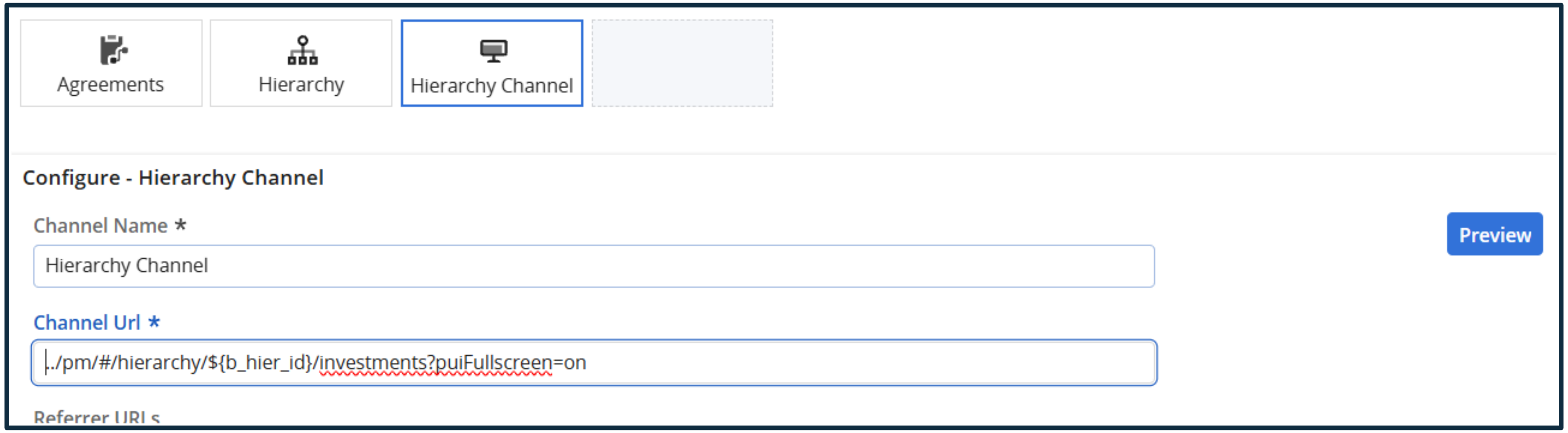
- Channels are configurable Modules which embed: Classic UI URLs (recommend Phoenix Theme and relative URLs), Modern UI URLs and External Applications
- Users can stay directly in Clarity and get the additional pertinent information
- Configuration
  - Channel Name
  - Channel URL
  - Referrer and SSO Configuration
  - Use attributes as parameters for the URL
- Tabs and Menu can be suppressed in channels and parameters can be passed to a channel



Modules – Channels

Channels

- For Clarity application URLs (Classic or Modern), use relative URL –
  - Use “..//pm/.....” Instead of “https://cppmxxxx.ondemand.ca.com/pm/.....”
- Add the following parameter to the Channel URL field to suppress menu: &puiFullscreen=on
- To hide the tabs, append the following parameter: &puiHidePPMTabs=on
- If there are no parameters passed, change the first "&" to "?“



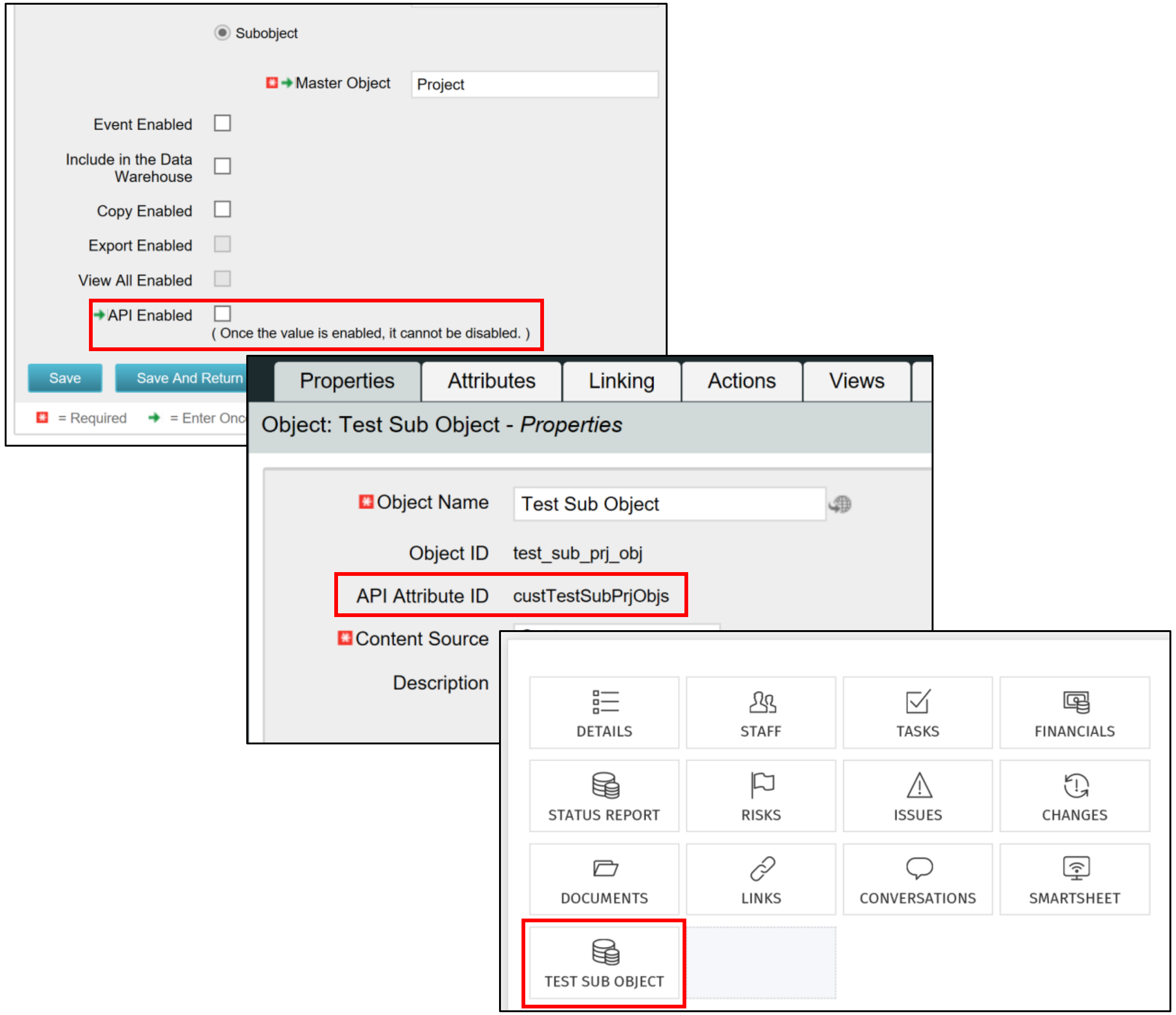
Modules – Channels

Parameters passed

- project id = ${_internalId}
- projectCode = ${code}
- project name =${name}
- project manager =${manager}
- logged in user =${_userId}
- logged in user's internal user ID 5 million number=${_userInternalId}
- logged in user's internal resource ID 5 million number=${_resourceInternalId}
- You can also reference custom attributes with the ${<api alias>}

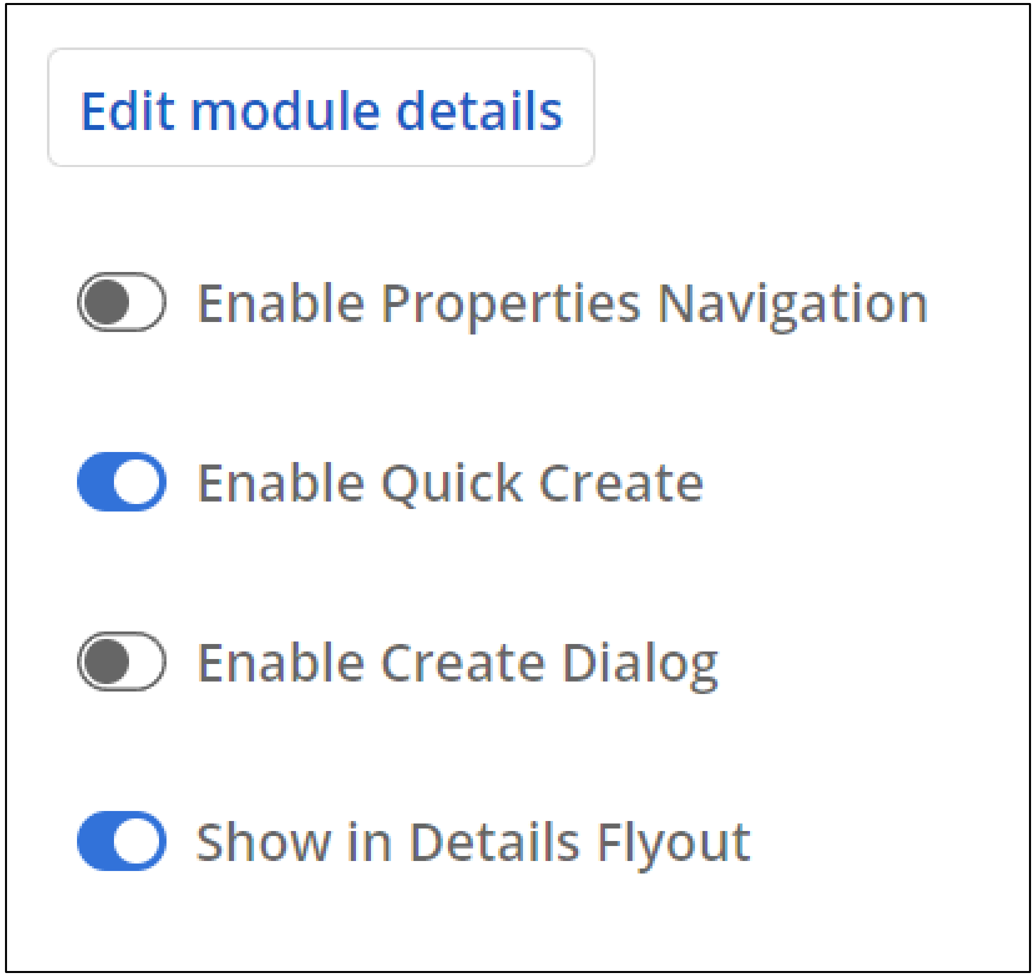
Modules – Custom Sub-Objects

- Custom sub-objects of Projects, Ideas, or Custom Investments as a Module in the New User Experience
- Check the “**API Enabled**” checkbox on the existing custom object or a new custom object
  - **Once this is checked and saved, it can’t be undone**
- After saving, an API Attribute ID will be automatically created for the object and it will be available as a Module
- Depending on the Clarity version, newer functionality is available:
  - Exposing 2nd and 3rd level sub-objects
  - Configuring “Properties pane” for the sub-objects



Modules – Custom Sub-Objects Blueprint

- In the module of the sub object on the blueprint, new settings are available
  - **Edit Module Details**: This is the blueprint for the properties of the sub object
  - **Enable Properties Navigation**: Allows the user to enter a blueprint of the sub object.
  - **Enable Quick Create**: Allow the user to create via the +
  - **Enable Create Dialog**: Allows the user to use comments
  - **Show in Details Flyout**: If the user selects the details flyout of the master, this module becomes a tab if enabled



Rules

- Rules in the Modern UX allow the administrator apply conditional logic that applies certain UI actions or data actions, when met to a specific blueprint
- There are four types of Rules:

View Page

- Allows the administrator apply conditional logic to perform UI actions, when met to a specific blueprint
- Supports conditions based on multiple attributes

Attribute Update

- Allows the administrator apply conditional logic to perform Data Actions as well as UI actions, based on an update to an attribute
- Can apply multiple actions per update of a target attribute

Security Update

- Allows the administrator apply conditional logic to perform Security Rights update actions, based on update to a specific resource lookup attribute

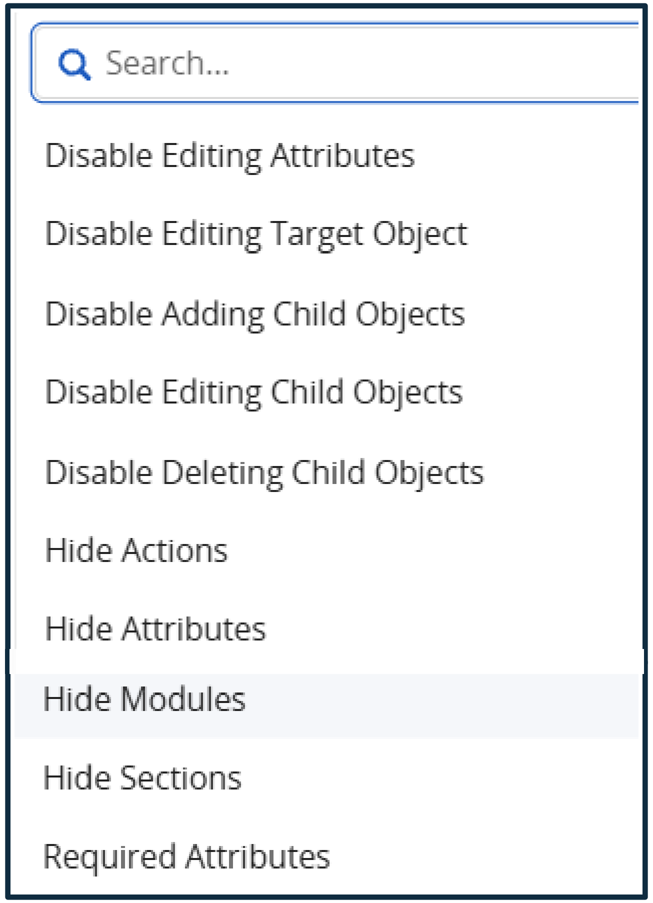
Conditional Required Modal

- Allows the administrator apply conditional logic to forcibly require certain set of fields with a Properties-like painted layout

UI Actions

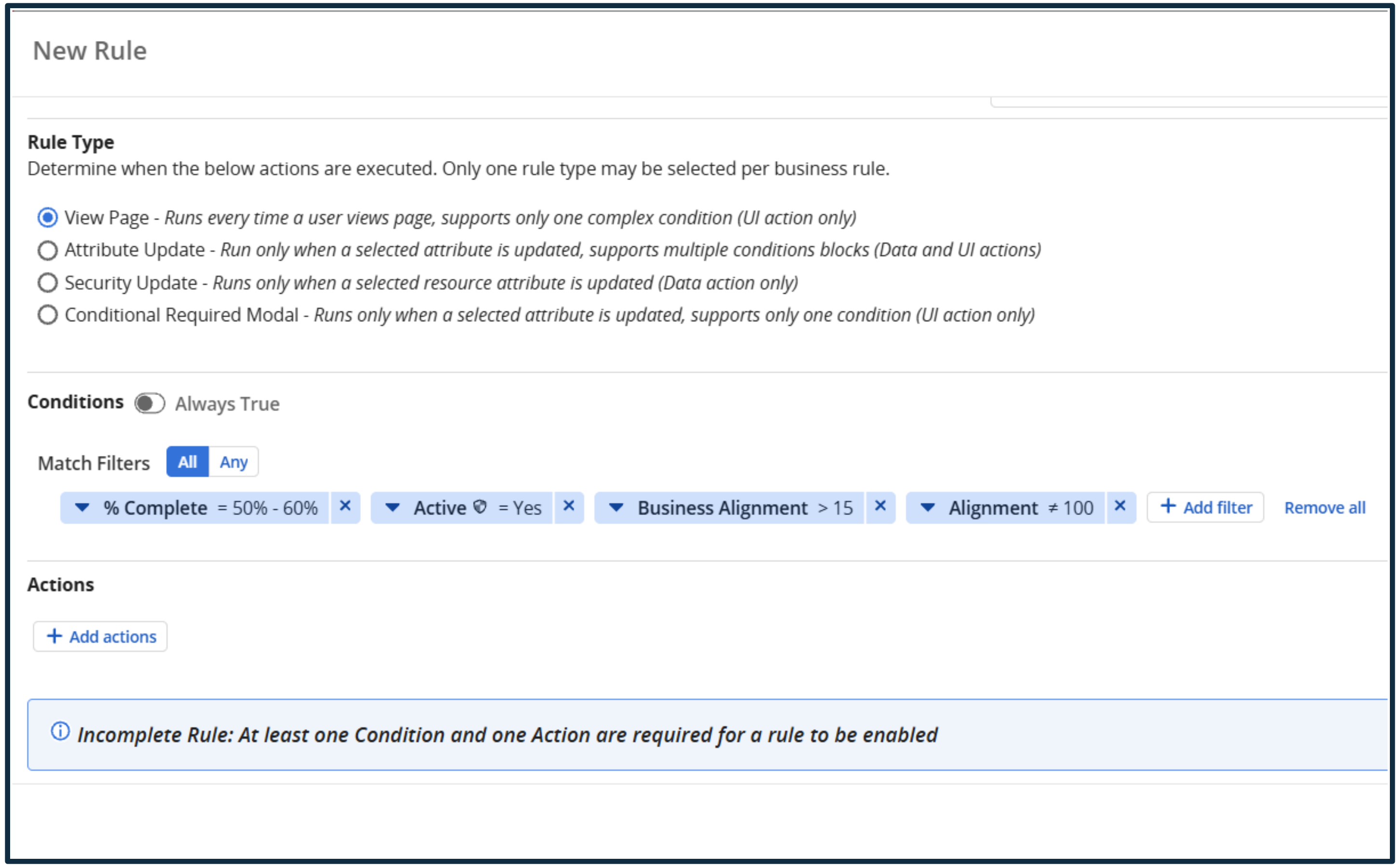
UI Actions that can be performed with a Blueprint rule:

- **Disable Editing Attributes**: Disable specific attributes to be edited
- **Disable Editing Record**: Disable the entire record from being edited
- **Disable Adding Child Objects**: Disable adding a new instance of the specified child object
- **Disable Editing Child Objects**: Disable editing any instance of the specified child object
- **Disable Deleting Child Objects**: Disable deleting any instance of the specified child object
- **Hide Actions**: Disable an action from being used
- **Hide Modules**: Disable specific tabs/channels from appearing to the user
- **Hide Sections**: Disable sections from appearing on the page
- **Required Attributes**: Require specific attributes. *(New to 16.1.3)
    - Note*:* This does not work on Rich Text Fields



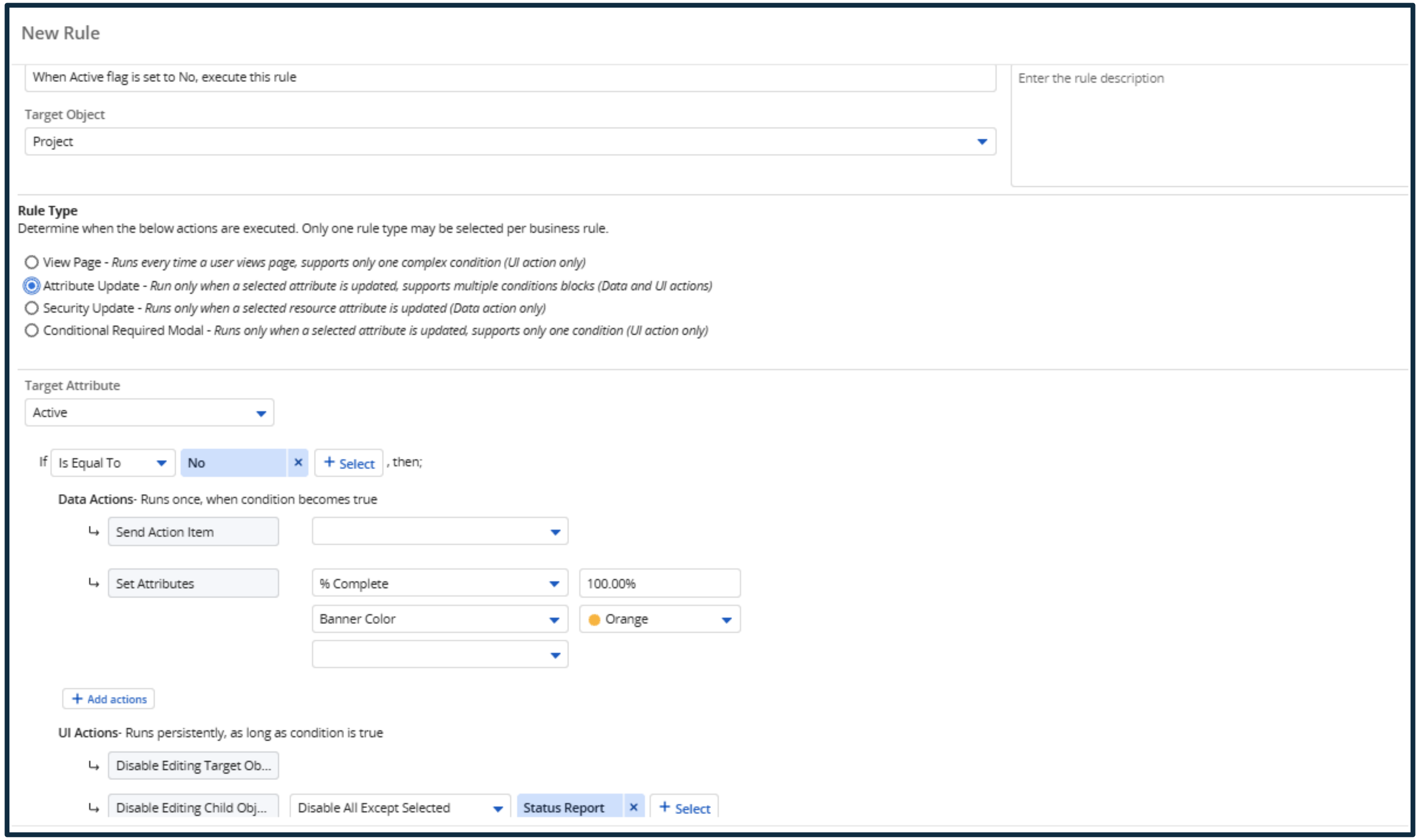
View Page Rule

- Allows the administrator to apply conditional logic to perform UI actions, when met to a specific blueprint
- Supports conditions based on multiple attributes, and supports “All” or “Any” conditions
- Only UI actions are supported with View Page rule



Attribute Update Rule

- Allows the administrator apply conditional logic to perform Data Actions as well as UI actions, based on an update to an attribute
- Can apply multiple actions per update of a target attribute
- Data actions include
    - Updating the value of an attribute
    - Or sending an action item

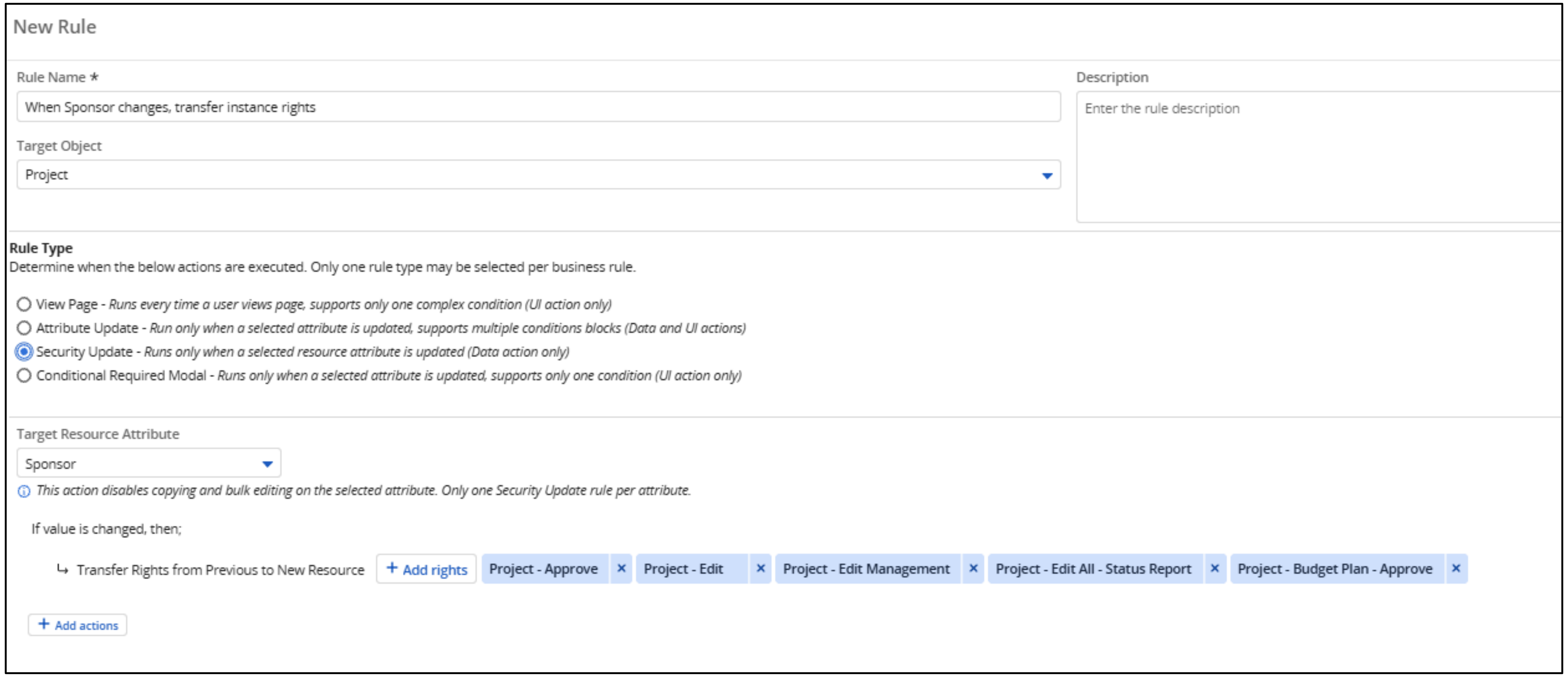


Note: Implementing data actions with **Rules** can be quicker and easier than through a process with a **System Actions**

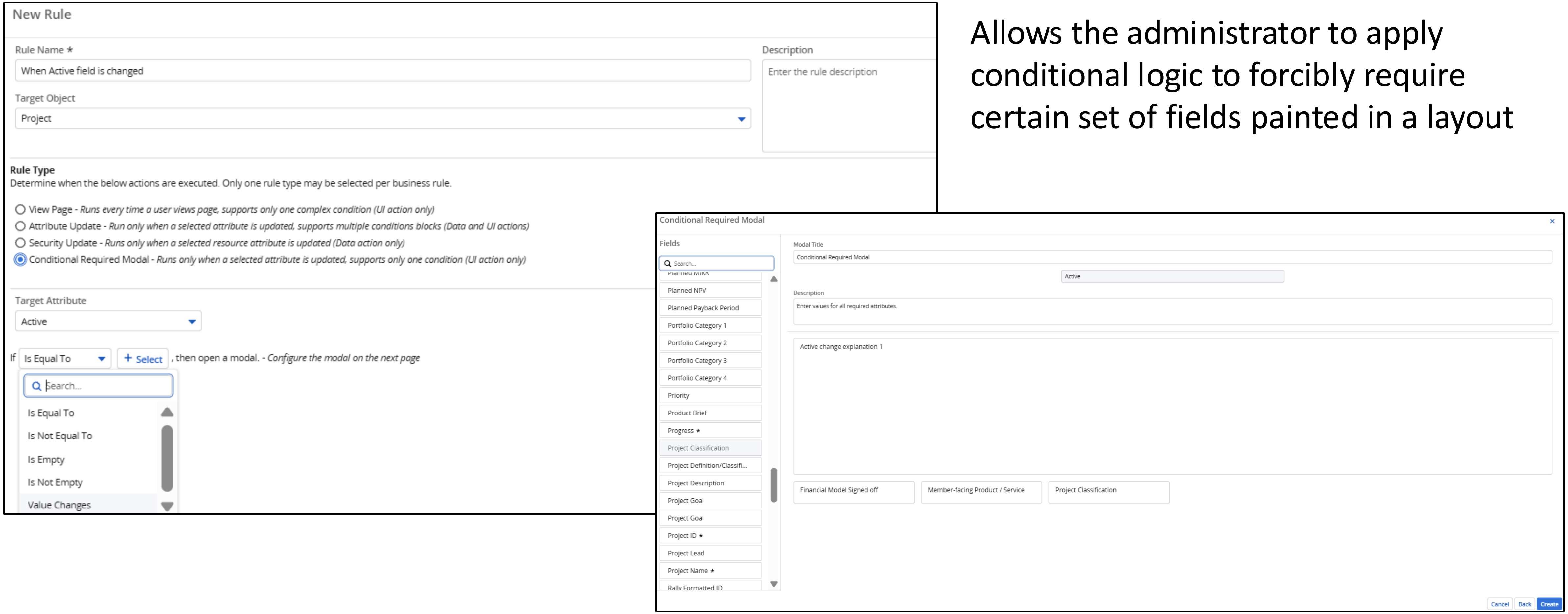
Security Update Rule

- Allows the administrator to apply conditional logic to perform Security Rights update actions, based on update to a specific resource lookup attribute

- Used to transfer (or grant or revoke) specific access rights when a target resource lookup attribute such as Sponsor, Business Owner, or Backup Project Manager is updated in project properties



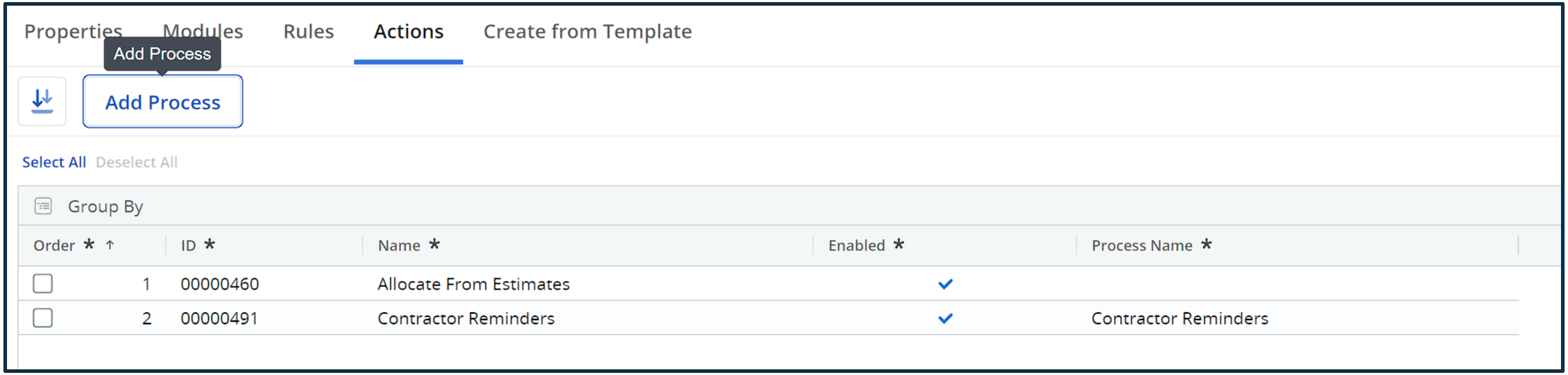
Conditional Required Modal Rule



Allows the administrator to apply conditional logic to forcibly require certain set of fields painted in a layout

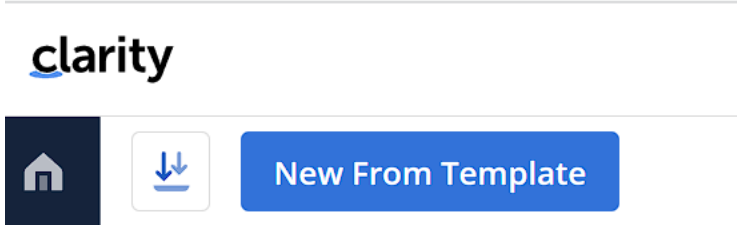
Actions

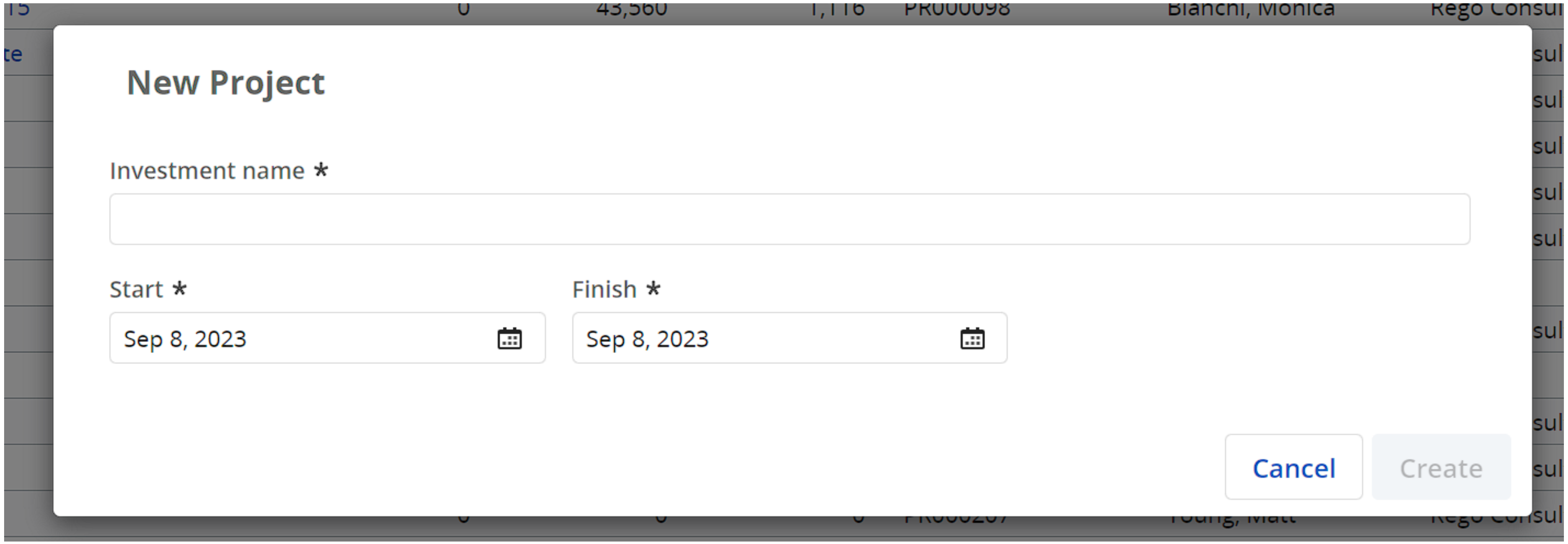
- Actions allow a user to kick off a process on demand
- Security can be applied to blueprint rules to show or hide an action based on user’s access group
- Processes can be specific to the object OR non-object associated processes



Create from Template

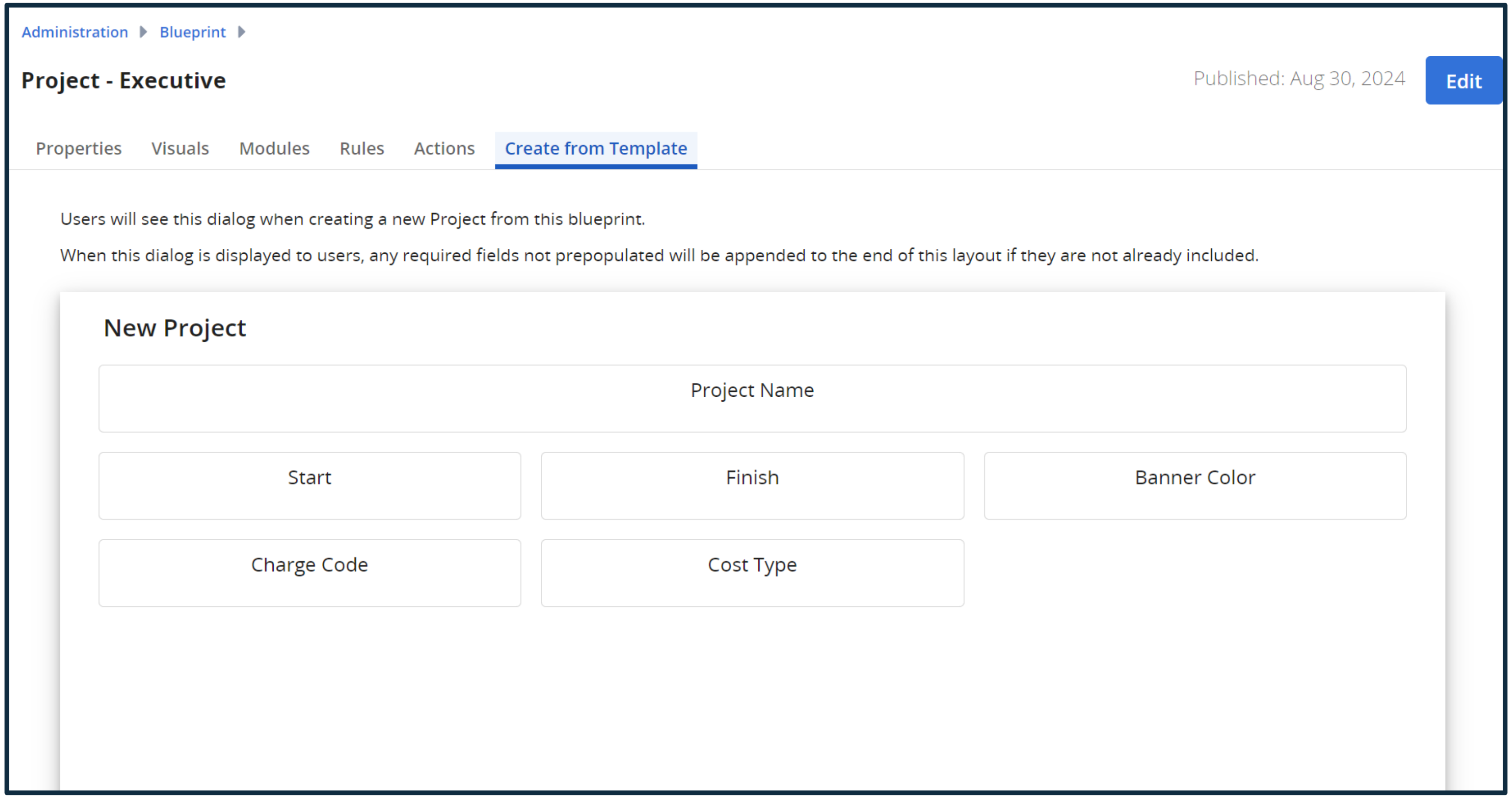
Create from Template defines the Properties View when using Templates. This will allow for required attributes and additional attributes to be prompted to end users.





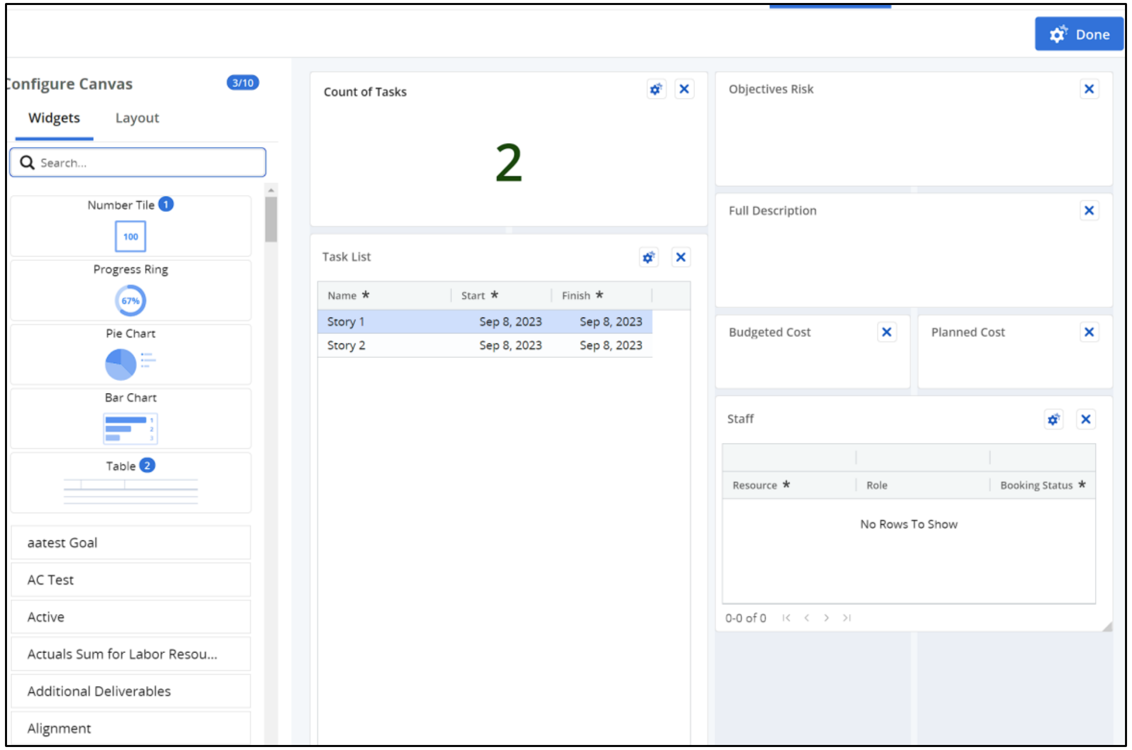
Customize Create from Template Blueprint

- The admin can customize the blueprint on the New From Template module in the blueprint associated to the Template
- Help Text can be added to the Create from Template module
- Sections cannot be added in the layout.



Canvas

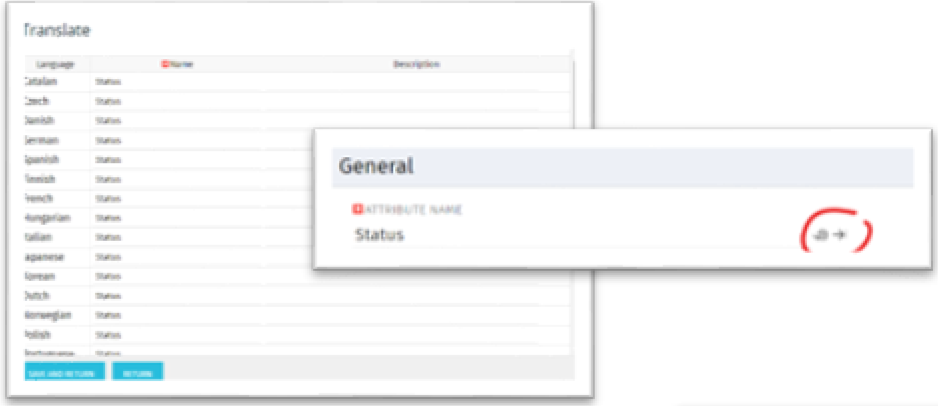
- Introduced in 16.1.0 (previously used for Status Reports only), the canvas. functionality allows you to create a new module that can be configured with fields from the master and sub objects
- Canvas allow grids, fields and widgets to provide Dashboard like experience to end users
- Canvas has 3 different layouts:
  - 4 columns: Good for a small display
  - 6 columns: Good for a medium sized display
  - 8 columns: Good for a dashboard sized display
- You can configure a canvas and save the configuration as a view.
- You can save multiple views in a canvas providing different views of data.

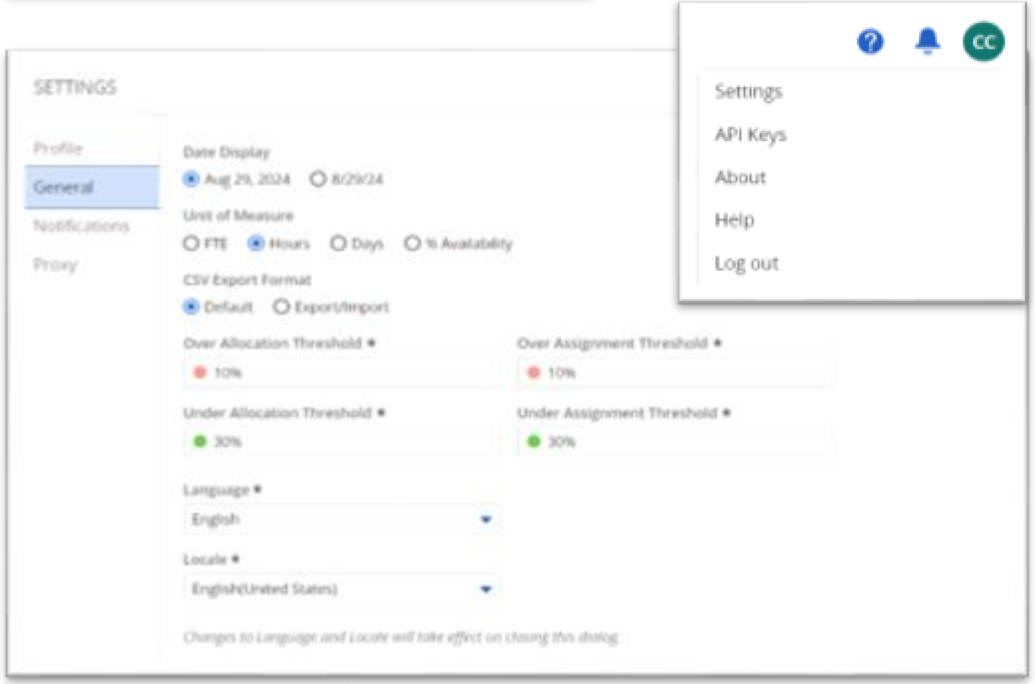


Multilingual Translations

- Attribute translations can be configured in Studio (Classic UI)
  - All languages configurable in one place
  - Applies to attributes in both Classic and MUX

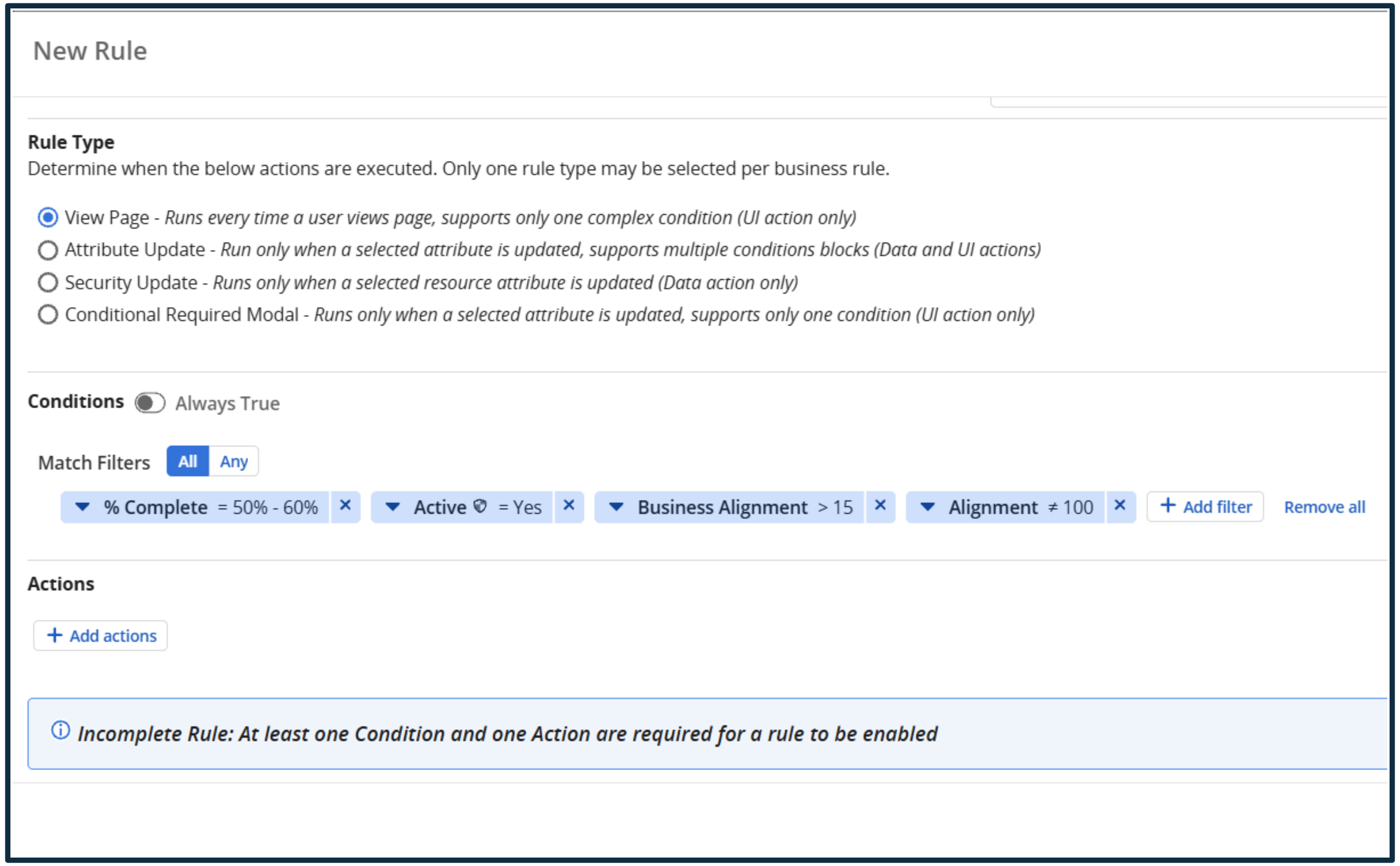
- All MUX translatable elements must be configured by switching languages for the logged-in user. This applies to:
  - Field Labels (Attribute Administration - MUX)
  - Blueprint Sections, Tabs, Action Buttons
  - Help text





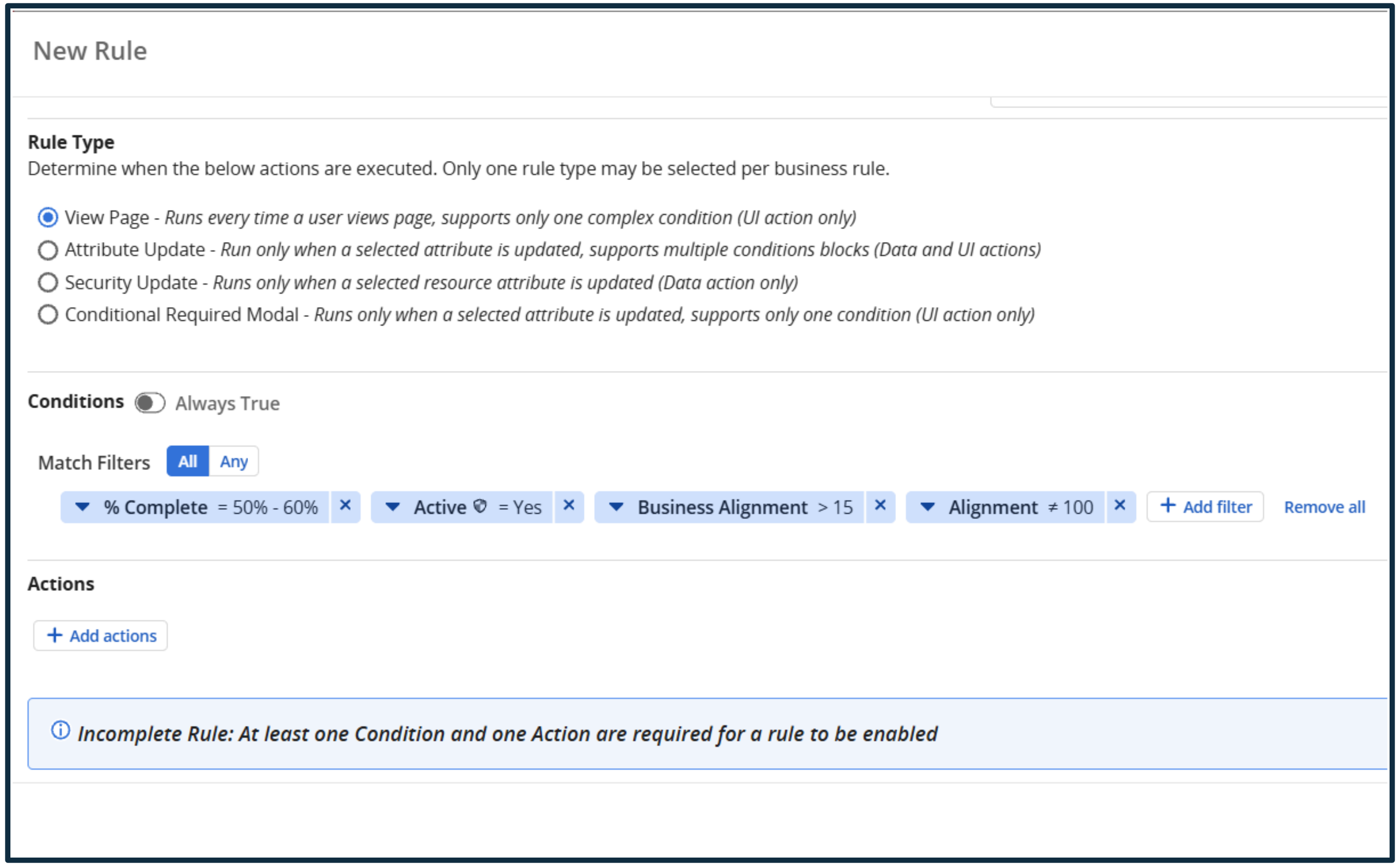
How to Create a Rule

- Select the Rules Tab
- Select New button
- Fill out the following:
  - Name: A meaningful title
  - Description: Add as much detail to understand and manage the rule
  - Target Object: This is where the rule will be applied. This can work on any sub object of the blueprint object.



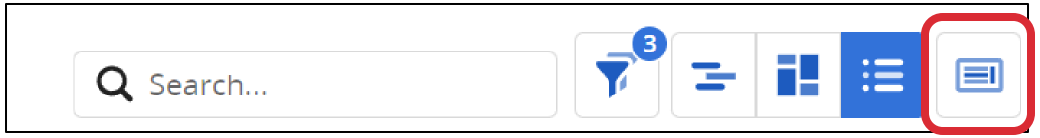
How to Create a Rule

- Fill out the following:
  - Rule Type: This defines when the logic should occur.
  - Conditions: Fill out a condition that needs to occur to apply your action
  - Actions: Setup the various actions you want applied when the conditions are met
- Select Create

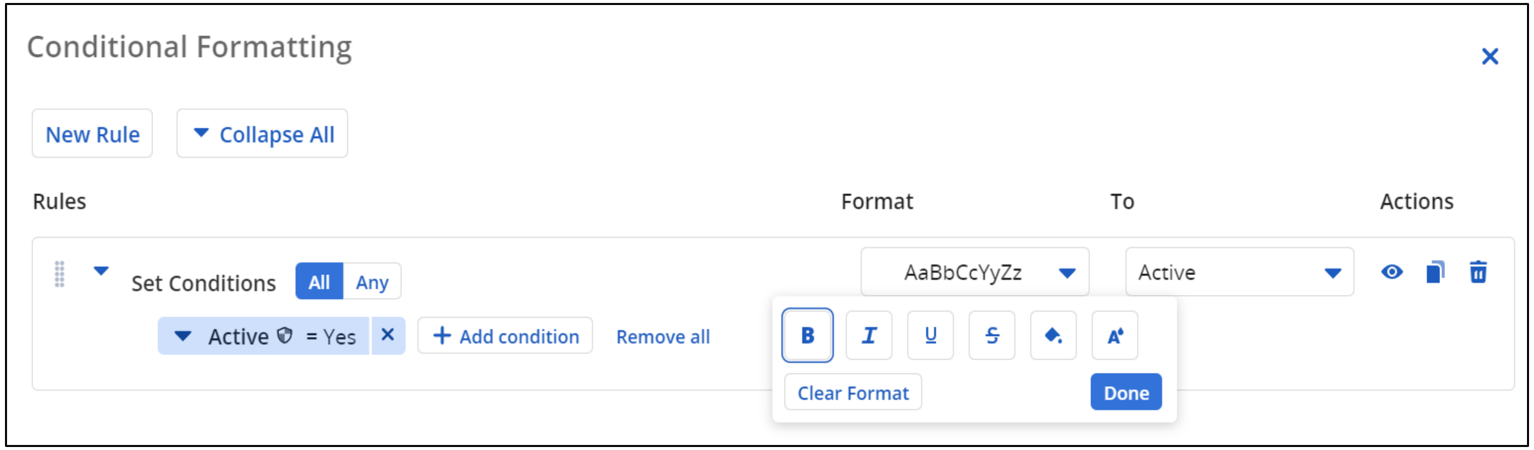


Conditional Formatting

- Conditional Formatting is now available on the list areas in the modern experience as of 16.2



- Rules: Apply a positive test scenario
- Format: Applies the below to the To
  - Bold
  - Italics
  - Underline
  - Strikethrough
  - Background Color
  - Font Color

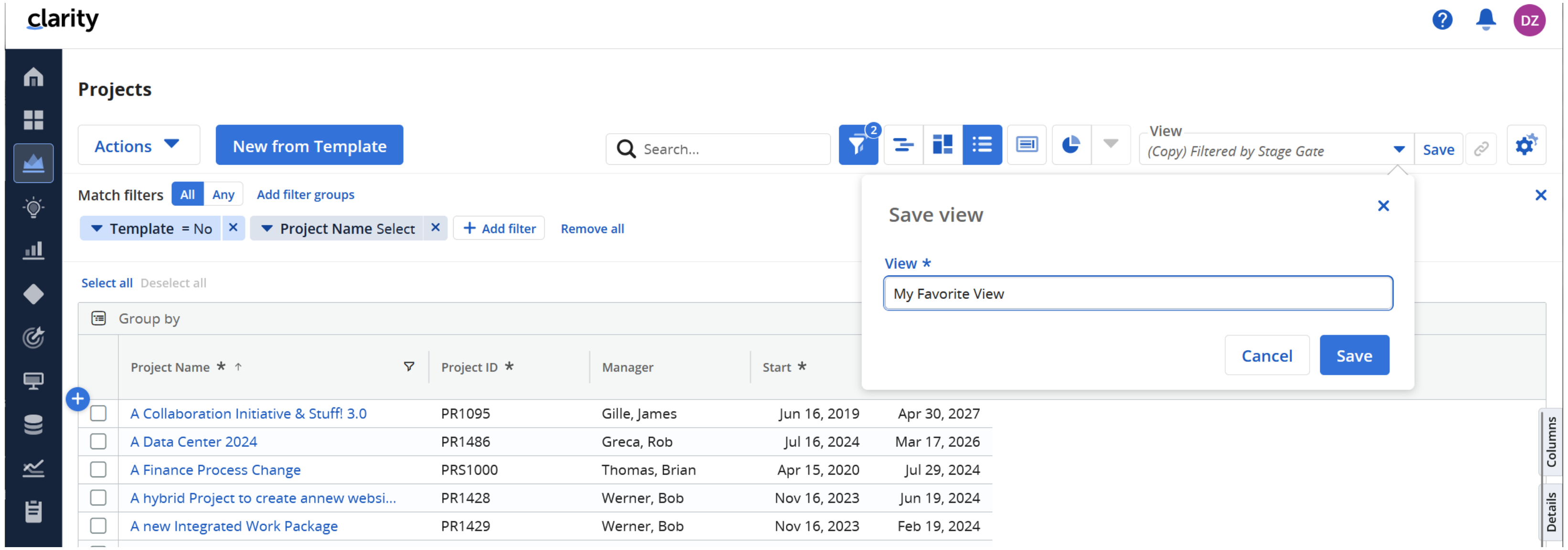


- To: What column should be affected by the formatting

User Personalization Views

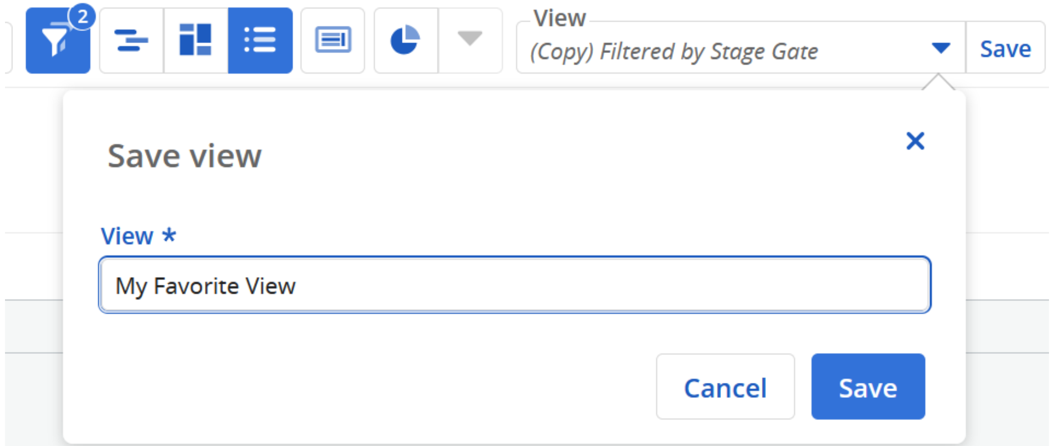
Saving Views and Filters (1)

- In the Classic UX, Administrators may save configurations to list views and filters, to ensure users see the right fields in view.
- That same functionality is not available in the Modern UX, but administrators can save a view, and have the end users choose that view, to allow them to see a predefined set of fields and filters.
- To save a view:
  - Configure the filter and columns as you want the users to see them using the column panel and the filter selection.



Saving Views and Filters (2)

- To save a view (continued):
  - Once the filter and fields are added, use the *View* menu in the upper right corner to select **Save As** to Save the View.

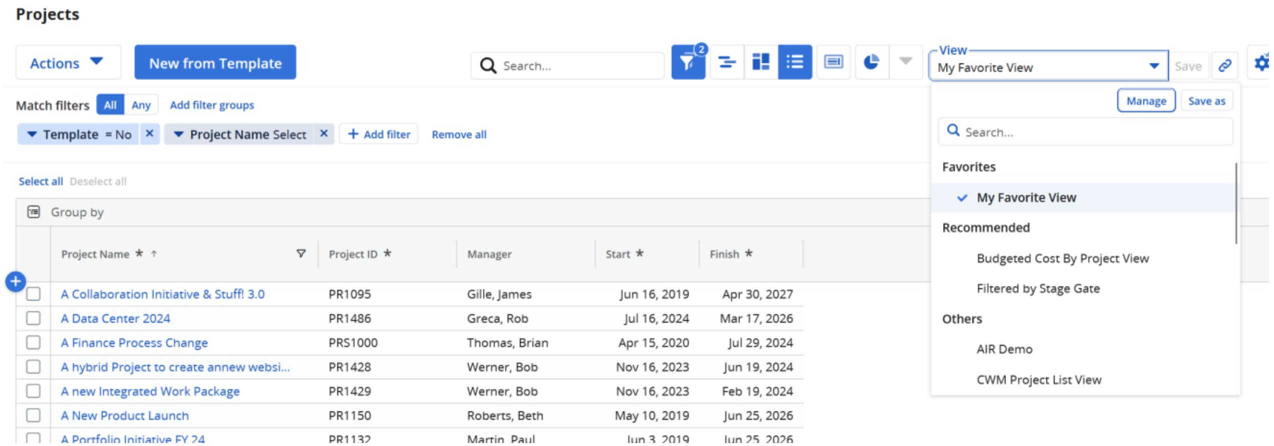


Enter a name for the View, and click **Save**

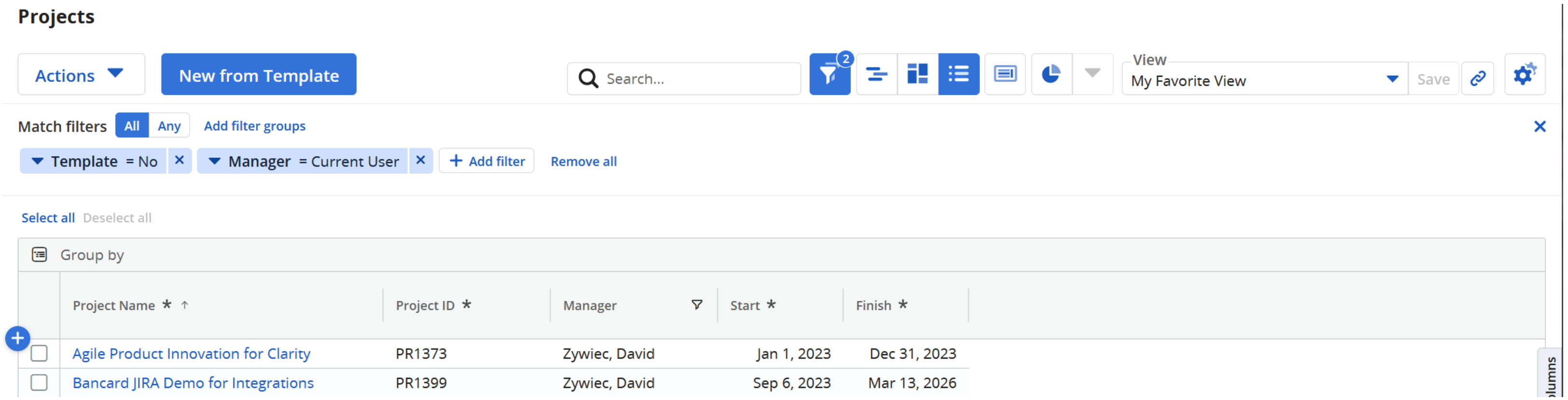
Saving Views and Filters (3)

- Once the View is Saved, End Users may then search for and apply the view applicable to them.
  - Instruct users to search for and apply the view using the View menu.

Original View, without desired fields and filters



New View, with desired fields and filters



User Personalization Picklists

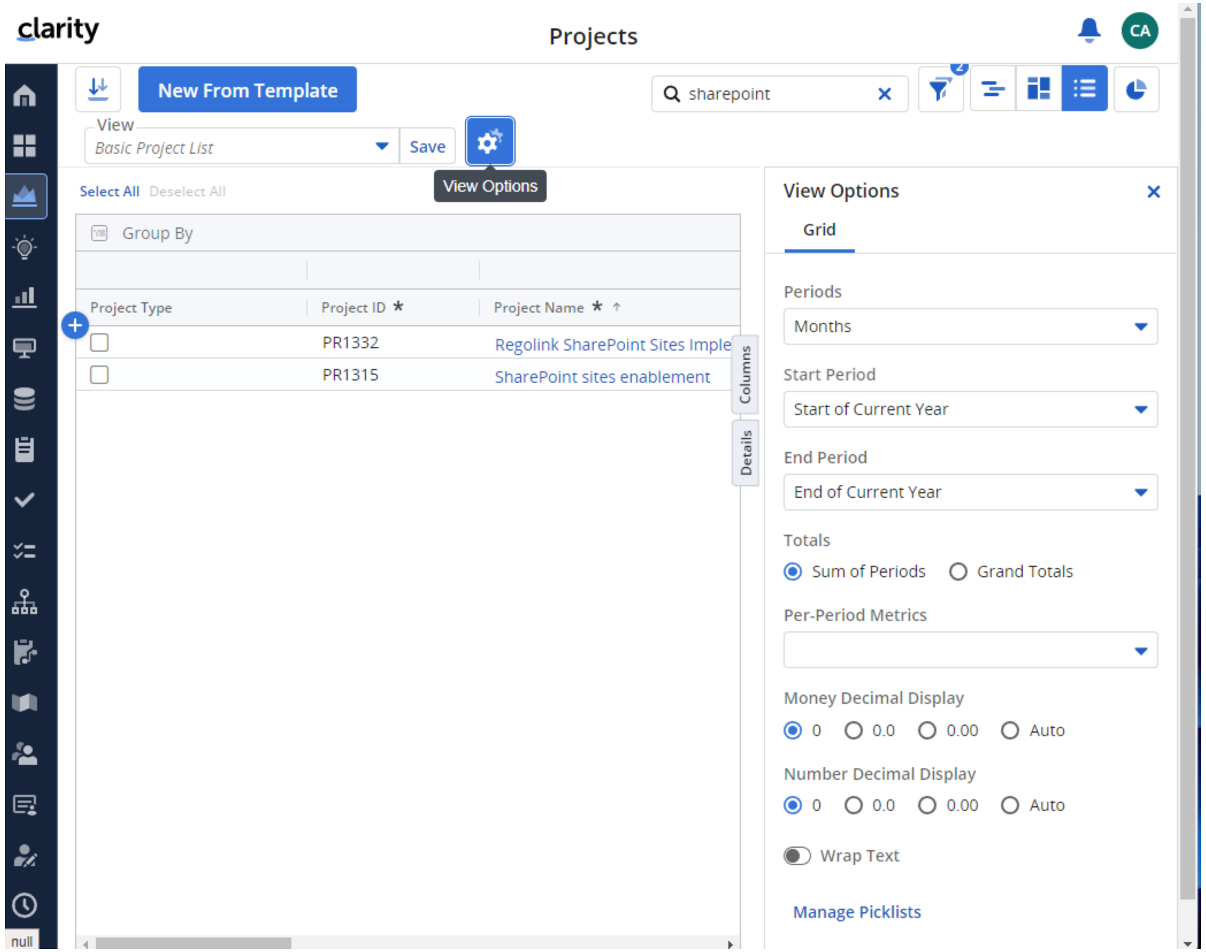
What Are Picklists?

Picklists are similar to Static Lookups, but can be added by users on the Modern UX without depending on Admins/Clarity Studio

They are only visible in Modern UX and under certain cases like Roadmaps or Tasks, the values are specific to that Investment or Roadmap allowing for additional levels of personalization

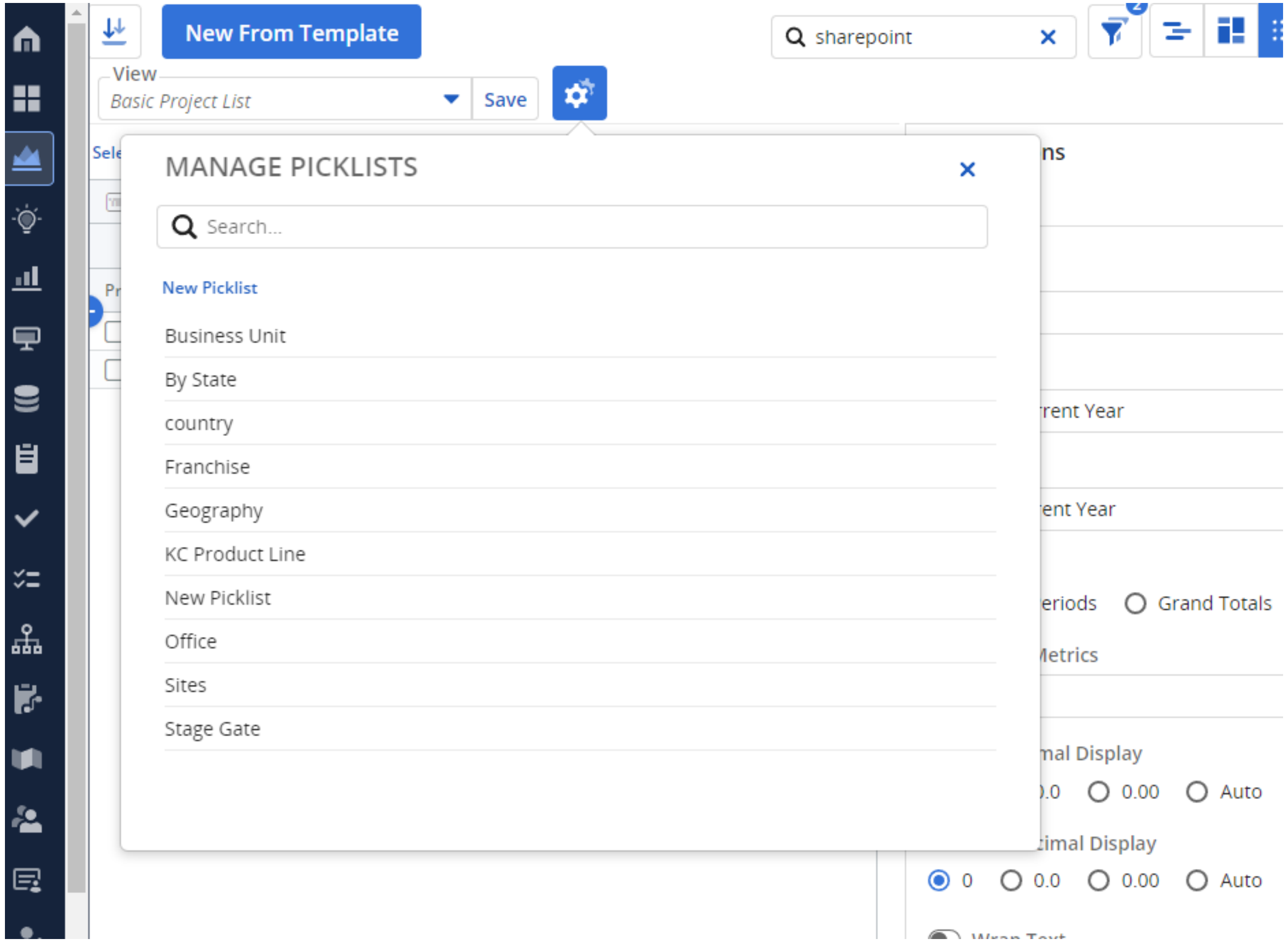
Creating a Picklist

- Navigate to an Object's List View that supports Picklists
- Click on the gear icon and select "Manage Picklists"



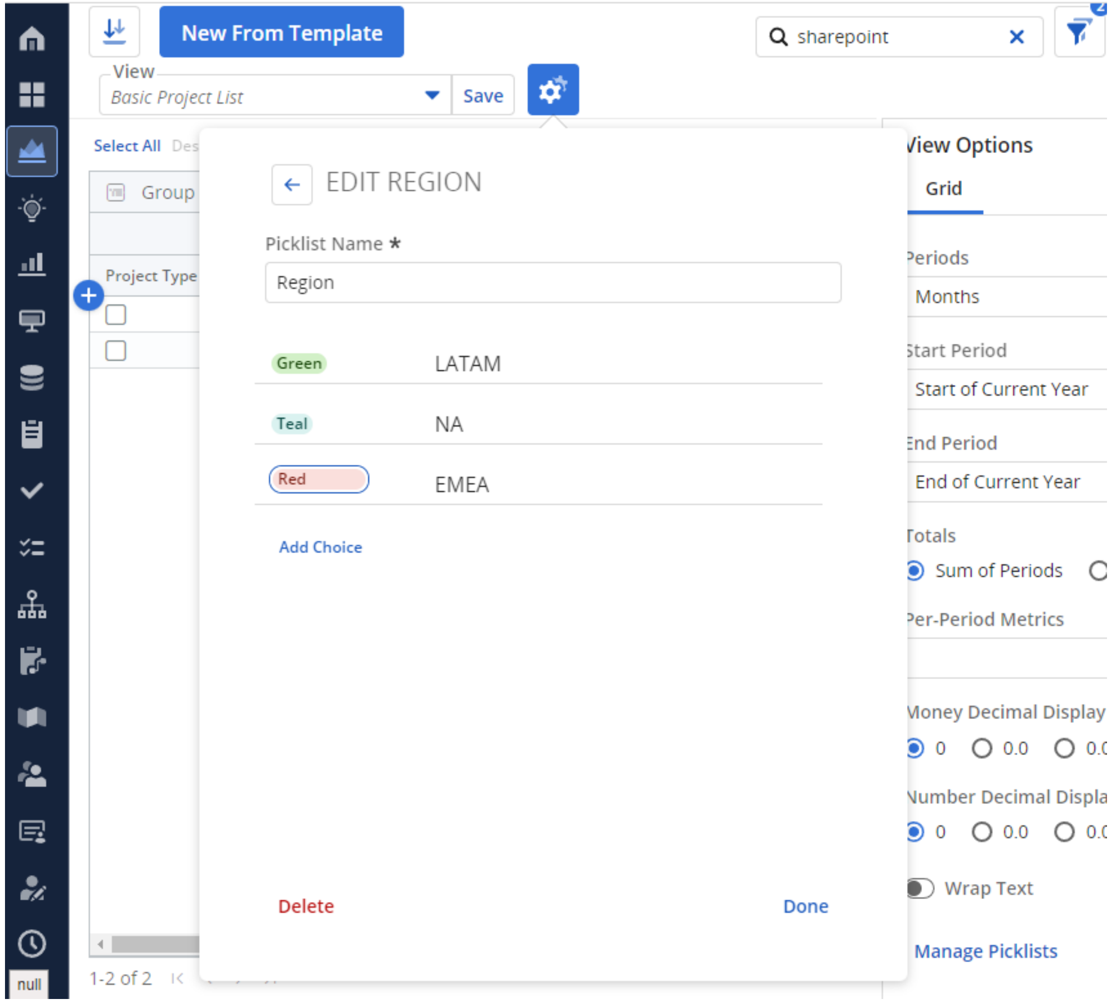
Creating a Picklist

- Once the “Manage Picklists” popup comes up, select “New Picklist”



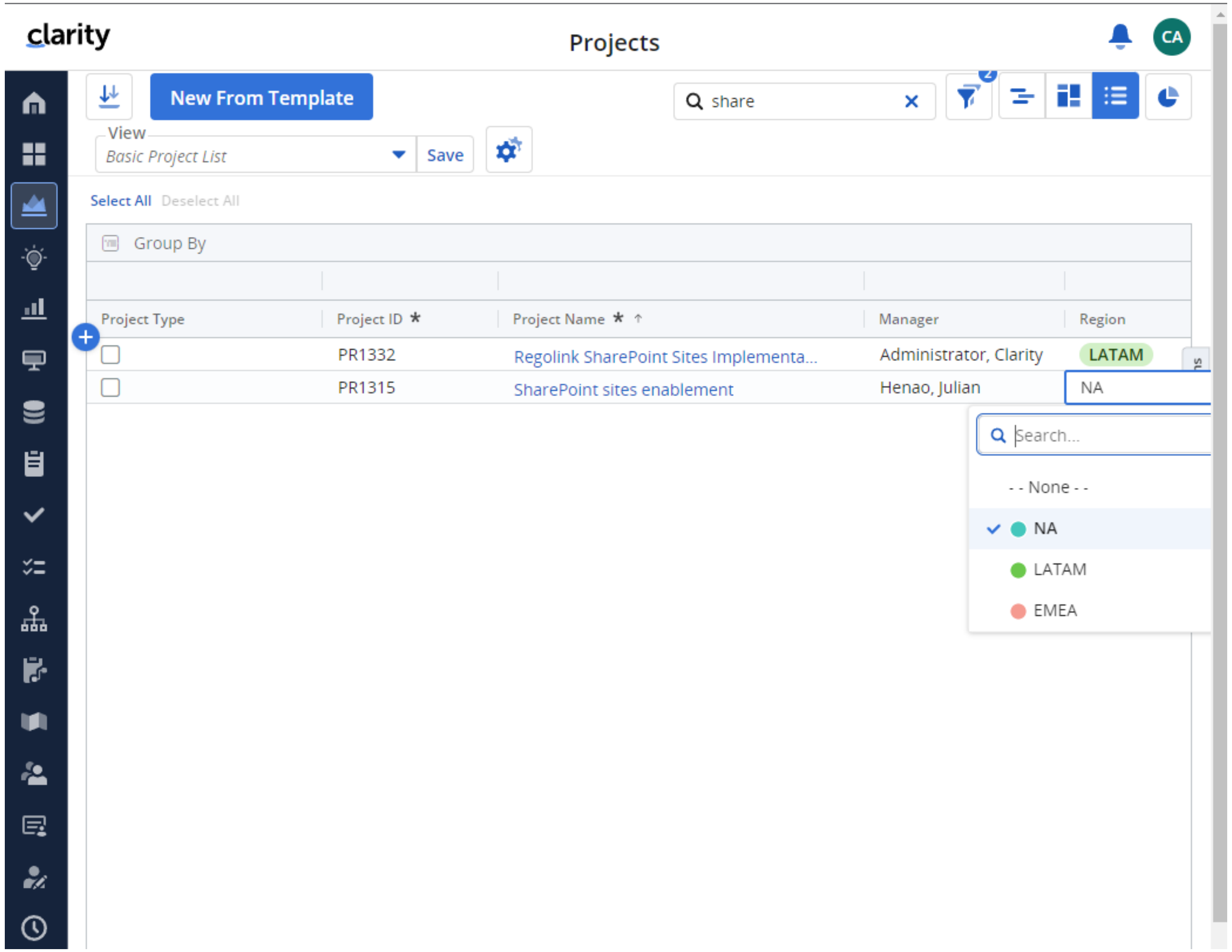
Creating a Picklist

- Provide a “Picklist Name”
- Provide values by selecting “Add Choice”
- Additionally, you can choose a color similar to Classic UI buckets
- After you are finished click on “Done”



Creating a Picklist

- The Picklist and values are now available



Questions?

Survey

Please take a few moments to fill out the class survey.

Your feedback is extremely important for future events.

bitly

Earn Rego University Certifications and Professional Specializations

Your participation in this class may count toward Rego University certification progress.

Certifications

Build recognized Clarity expertise.

Complete 10 eligible learning units within a certification track.

Best Practice
Clarity Administrator

Best Practice
Clarity Lead

Professional Specializations

Deepen your skills in focused topic areas.

Complete 4 eligible learning units within a selected specialization, with the option to earn multiple specializations.

Artificial Intelligence (AI)

Reporting & Analytics

Technical

***Important:* Complete the class survey in the app after each session so your credits can be tracked.**

More info: https://regouniversity.com/certificates

Earn PMI credit

Eligible Rego University sessions qualify for Professional Development Units (PDUs)

- Access your account at pmi.org
- Click on **Certifications**
- Click on **Maintain My Certification**
- Click on **Visit CCR's** button under the **Report PDU's**
- Click on **Report PDU's**
- Click on **Course or Training**
- Class Provider = **Rego Consulting**
- Class Name = **Rego University**
- Course **Description**
- Date Started = **Today's Date**
- Date Completed = **Today's Date**
- Hours Completed = **1 PDU per hour of class time**
- Training classes = **Technical**
- Click on **I agree** and **Submit**

Let us know how we can improve!
Don't forget to fill out the class survey.

Phone
888.813.0444

Email
info@regoconsulting.com

Website
https://regouniversity.com/pmi-credits/

Continue to Get Resources and Stay Connected

1. Use [RegoXchange.com] for instructions and how-to guides.
2. Talk with your account managers and your Rego consultants.
3. Sign up for webinars and join in-person Rego groups near you at RegoConsulting.com
4. Join us for the next Rego University!