

P05: Creating Business Rules in Power Apps

Tuesday, March 11, 2025 10:48 AM

Context

Item	Detail
Tag	Vertical Patterns Power Apps
Contributors	Vincent Kimiti
Version Control	V1.0
Application Use Case	Enforce business logic by setting default values and displaying error messages based on conditions
Reference Usage	Automating form behavior with business rules in Power Apps

Story Behind The Pattern

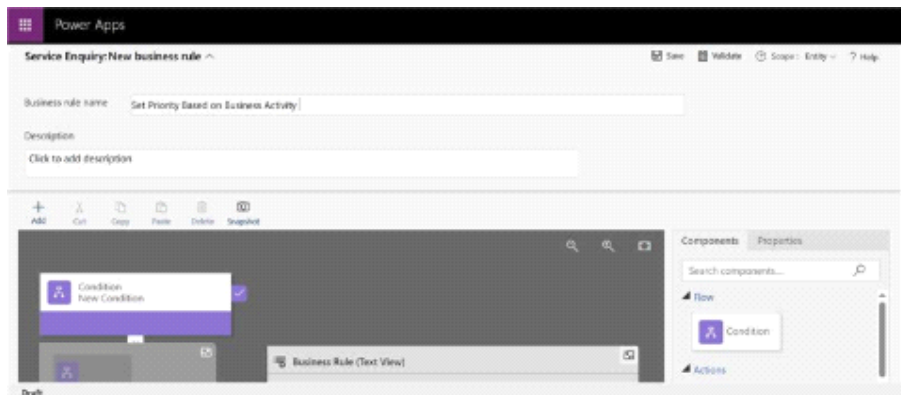
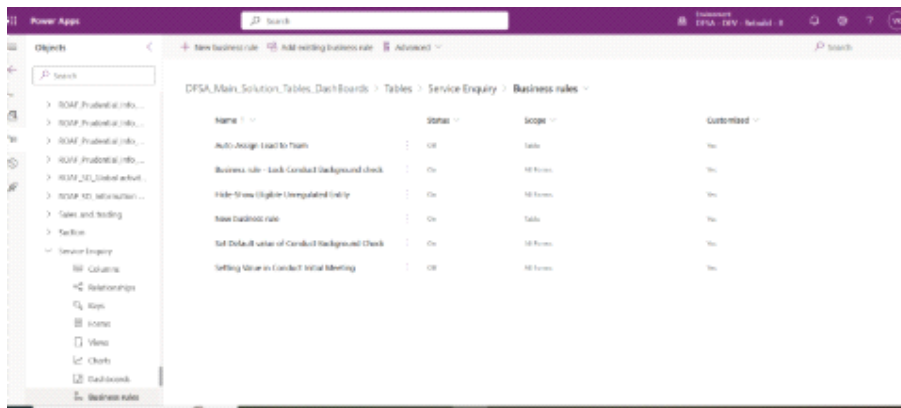
Item	Detail
The Problem	Manual enforcement of business logic (e.g., setting default values or validating inputs) on forms is error-prone and inconsistent.
The Solution	Use business rules in Power Apps to automate form behavior, such as setting default values and displaying error messages when conditions are met or not met.
Dependencies	Power Apps environment, access to a table (entity) with a form, permissions to create and edit business rules.

The Pattern

Method : Creating Business Rules in Power Apps

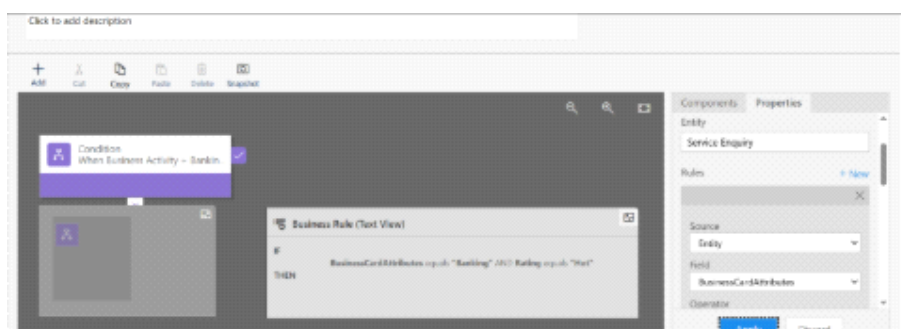
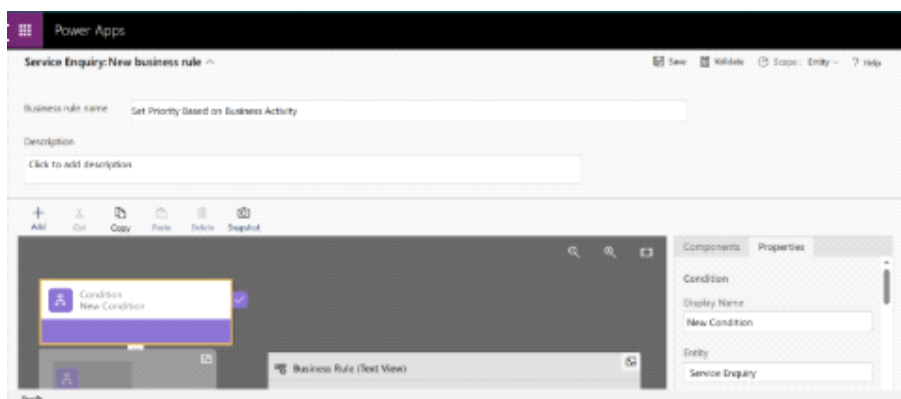
Here's a step-by-step guide for creating a business rule in Power Apps to enforce logic on a form, such as setting a default value and displaying an error message based on a condition:

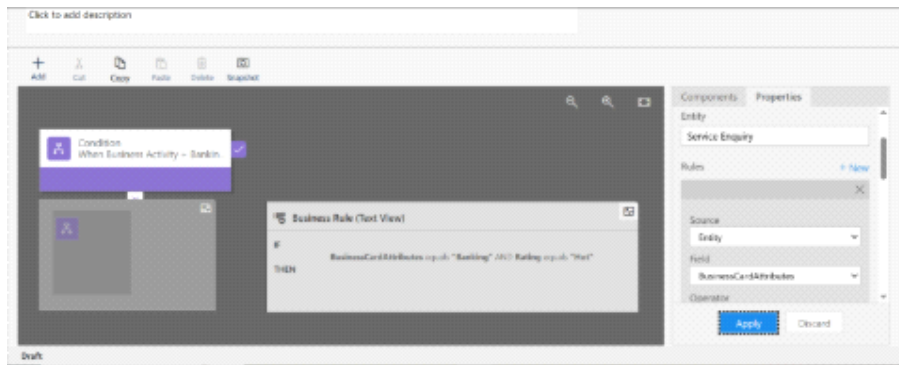
1. Setup Business Rule Name (Naming)
 - o Log in to your Power Apps environment.
 - o Navigate to the Solutions area in the Power Apps Maker Portal.
 - o Select or create a solution where you want to add the business rule.
 - o In the solution, select the table (entity) you want to create the business rule for (e.g., "Accounts" or "Contacts").
 - o Under the table, go to the Business Rules tab and click New Business Rule.
 - o In the business rule editor, provide a meaningful name, such as "Set Default Priority and Validate Status."



2. Setting Up Condition (Set Condition and Name of Condition)

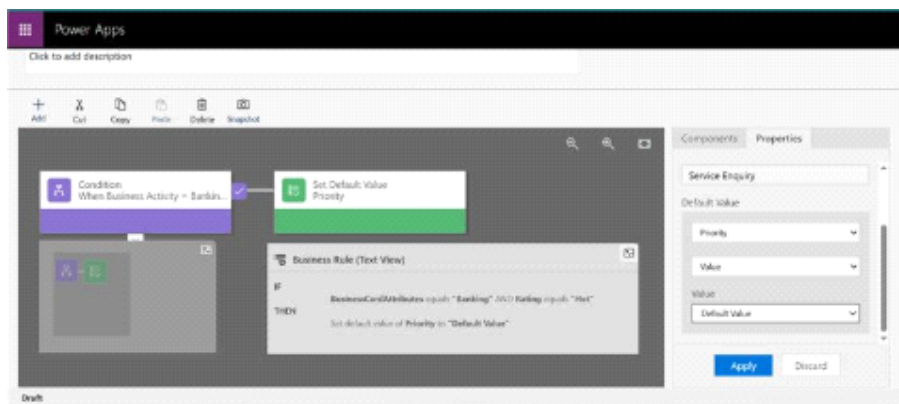
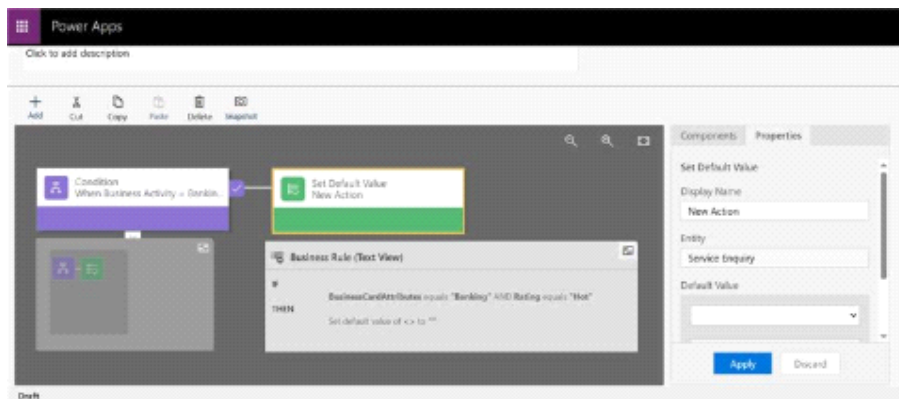
- In the business rule editor, click + Add Condition to define a condition that will trigger the rule.
- Name the condition for clarity, e.g., "Check Status Field."
- Set the condition logic, such as:
 - Field: "Status" (or another field on the form).
 - Operator: Equals (or another operator like Contains, Greater Than, etc.).
 - Value: "Active" (or another value relevant to your use case).
- This condition will determine when the business rule actions are applied (e.g., if the "Status" field is "Active").





3. Setup Action on Condition Fulfillment (Set Default Value Action)

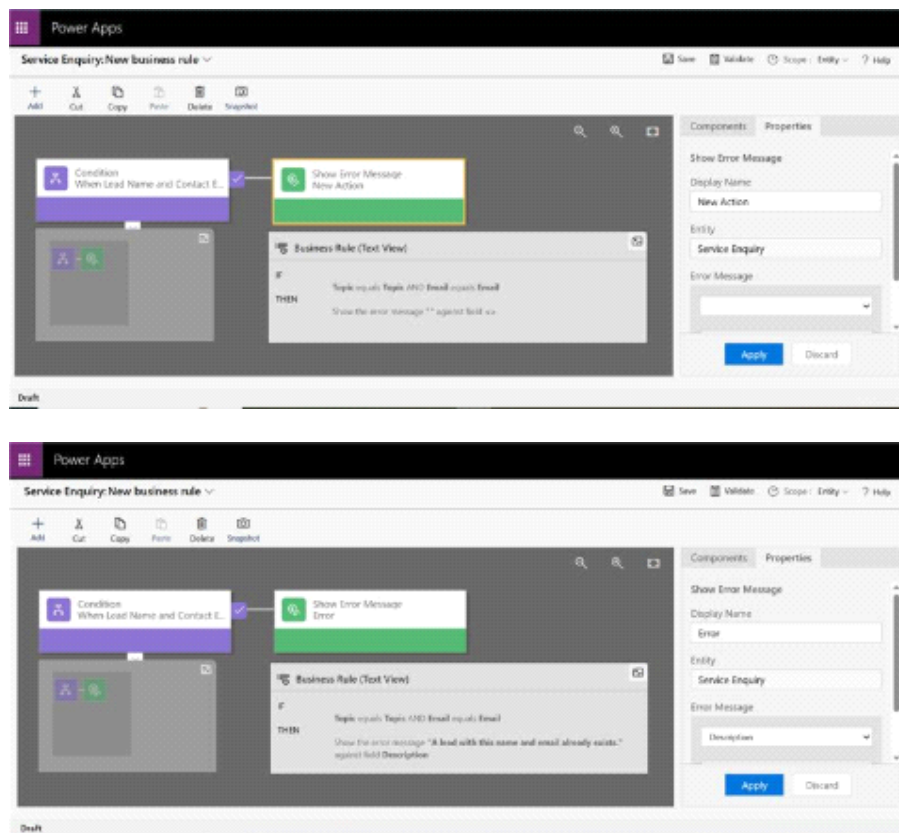
- In the "If Yes" branch of the condition, click + Add Action to define what happens when the condition is met.
- Select Set Field Value as the action type.
- Configure the action to set a default value:
 - Field: "Priority" (or another field you want to set a default value for).
 - Value: "High" (or another default value, such as a number, text, or choice).
- This action ensures that when the condition is fulfilled (e.g., Status = Active), the "Priority" field is automatically set to "High."



4. Set Error Message Action (Set Error Message)

- In the "If No" branch of the condition (or add another condition if needed), click + Add Action to define what happens when the condition is not met.
- Select Show Error Message as the action type.
- Configure the error message:
 - Field: Select the field to associate the error message with (e.g., "Status").
 - Message: Enter a clear error message, such as "Status must be set to Active to proceed."
- This action will display the error message on the form if the condition fails (e.g., if the

"Status" field is not "Active").



5. Save Business Rule

- Once the condition and actions are configured, click Validate in the business rule editor to check for errors.
- If no errors are found, click Save to save the business rule.
- Click Activate to enable the business rule, making it active on the form.
- Test the business rule by opening a record in the associated table's form, changing the "Status" field, and verifying that the "Priority" field updates and the error message appears as expected.

