

---

## Joget DX

# Designing your first Process

 <http://facebook.com/jogetworkflow>

 <http://twitter.com/jogetworkflow>

# Prerequisites

---

- Installed Joget.
- Understand various components of Joget.
- General understanding of Joget as an end user.

# Content

---

1. Business Process Design (BPE)
2. Designing your First Process with Process Builder
3. Running your First Automated Process
4. Participant Mapping
5. Process Monitoring

---

# Chapter 1

## Business Process Design (BPE)

# Business Process Management

---

- BPM is a management approach focused on aligning all aspects of an organization with the wants and needs of clients.
- Wants & needs = business goals.
- BPM attempts to improve processes continuously.

# Business Process Engineering

---

1. Identify **a process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# Identify a Process

---

- E.g:
  - Purchase Request Process
  - Purchase Order Process

Note: Let's start simple by focusing **only** on purchase request process first.

# Business Process Engineering

---

1. Identify a **process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# Start & End of Purchase Request Process

---

- Start: 
  - Applicant submits **purchase request form**
- End: 
  - Applicant informed with **purchase request result**

# Business Process Engineering

---

1. Identify a **process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# Process Participants

---

-  Request applicant.
-  Request approver.
-  Purchasing department (PD).

Note: The approver can also be the same participant that handles the purchase orders.

# Business Process Engineering

---

1. Identify a **process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# As-IS Process Activities

---

-  Applicant **submits** purchase request form to PD.
-  PD staff **checks** available stock.
-  PD staff **approves / rejects** purchase request.
-  PD staff **informs applicant** on the request status.
-  PD staff **escalates** approved requests to the purchaser in the department to perform purchase ordering.

# Business Process Engineering

---

1. Identify a **process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# Business Rules

---

1. If requested item in stock is low:
  - Approver processes purchase request

Else if many more stock is available:

  - Reject purchase request
2. If purchase request is approved:
  - Escalate approved request to purchaser to handle ordering.
  - Inform applicant on the approved request

Else if purchase request is rejected:

  - Inform applicant on the rejected request

# Business Process Engineering

---

1. Identify a **process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# To-Be Process Activities

## As-Is

1. Applicant **submits purchase request** form to PD
2. PD **checks available stock**
3. PD **informs** applicant on **existing abundance of stock**
4. PD **approves / rejects** purchase request
5. PD **informs** applicant on **request status**
6. PD **escalates** approved requests to purchaser to handle purchase ordering

## To-Be

1. Applicant **submits purchase request** form to PD
2. System **checks available stock**
3. System **notifies** applicant on **existing abundance of stock**
4. PD **approves / rejects** purchase request
5. System **notifies applicant on request status & purchaser on new approved request via email**

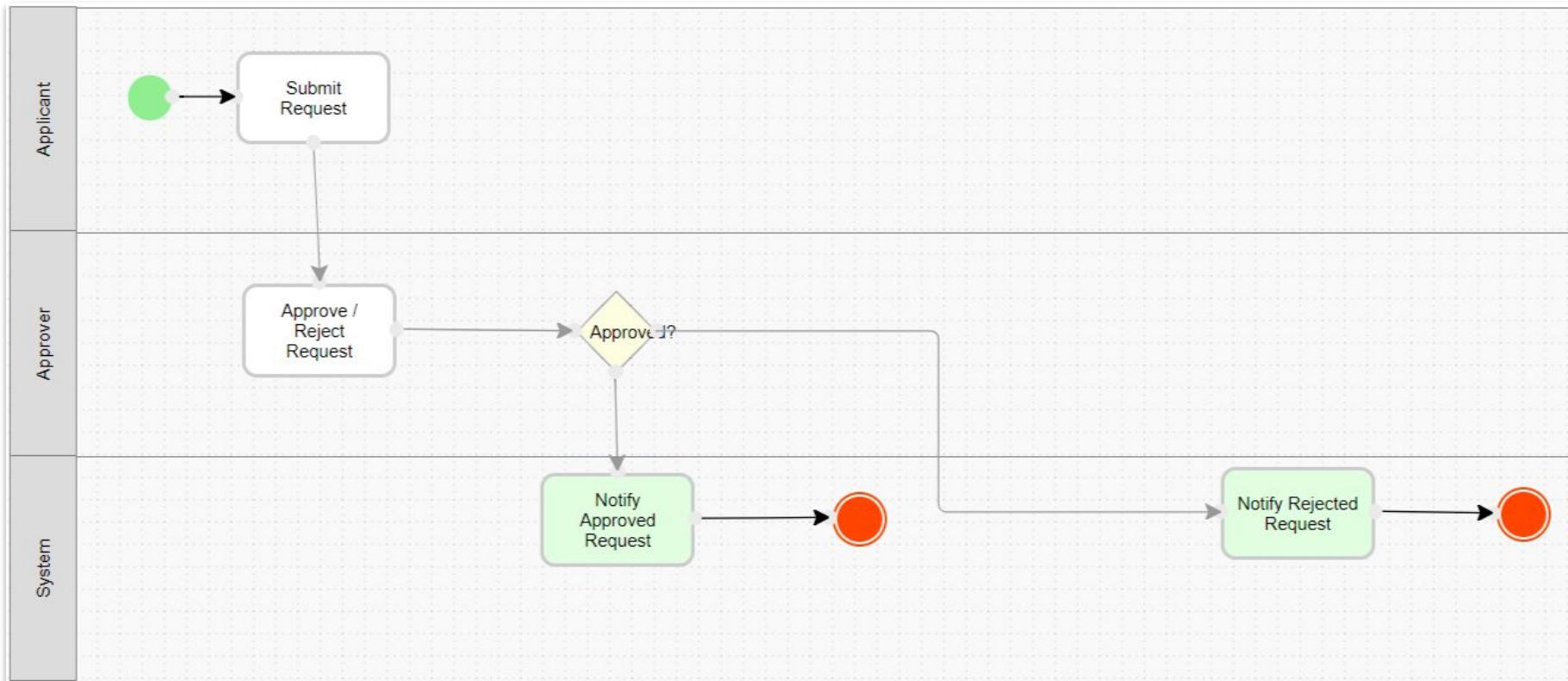


# Business Process Engineering

---

1. Identify a **process**
2. Identify **start** and **end** of the process
3. Identify process **participants**
4. Identify **as-is** process **activities**
5. Identify **business rules**
6. Identify **areas for optimization**: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / **to-be** process as flowchart

# To-Be Purchase Request Process

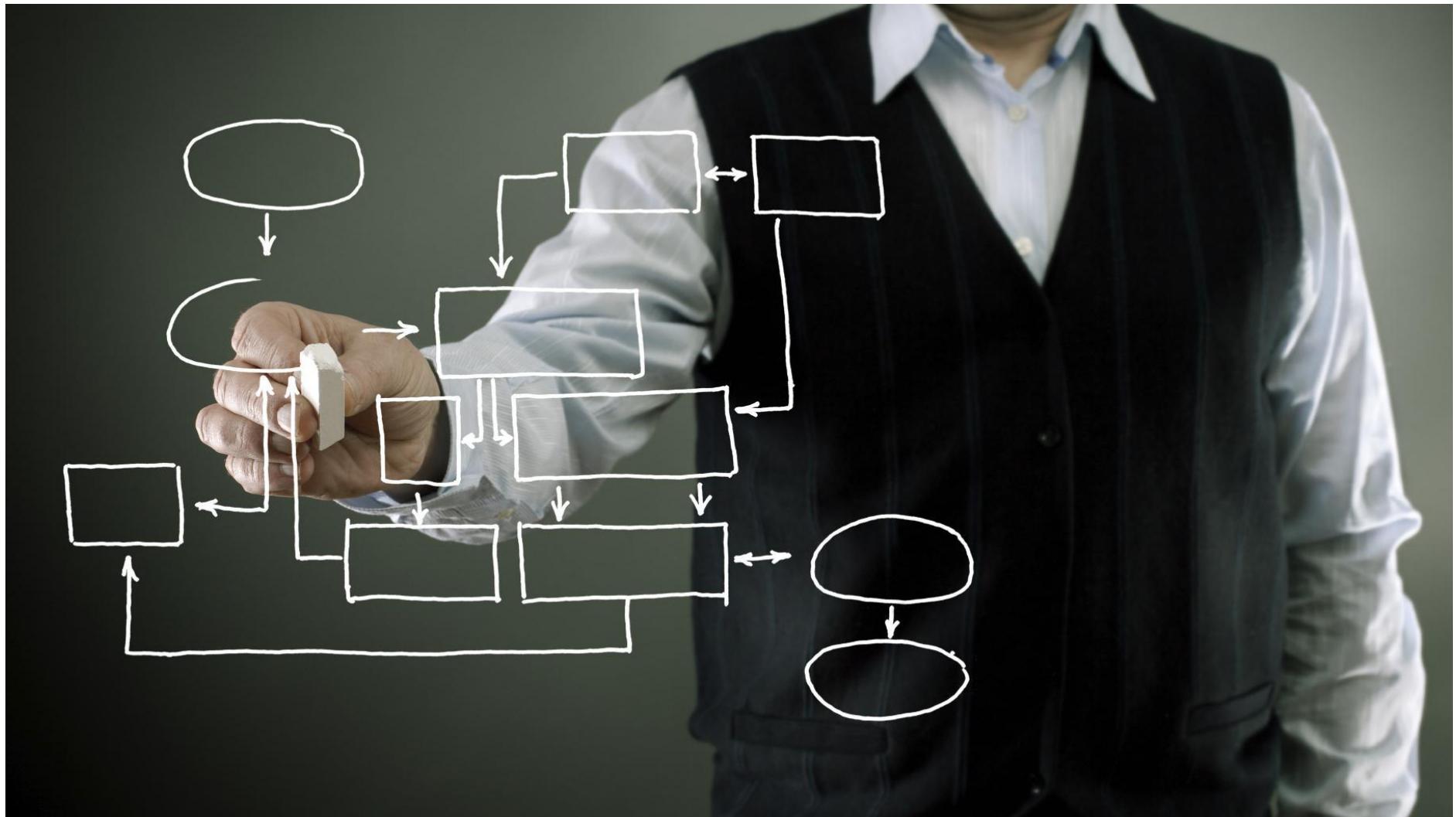


# Business Process Engineering

---

1. Identify a process
2. Identify start and end of the process
3. Identify process participants
4. Identify as-is process activities
5. Identify business rules
6. Identify areas for optimization: process activities and business rules that can be:
  - Improved
  - Eliminated
7. Draw the improved / to-be process as flowchart

# Now, Identify a Process, and Engineer It!



# Sample Process Statement

---

- Jane needs to get some stationeries.  
The request that she submits must be first approved by PD staff before being sent to another respective staff in the department to handle the order.  
The department will then notify Jane accordingly.

# Hands-On: Business Process Engineering

---

1. Process Name

2. Start and End

3. Participants

4. As-Is Activities

5. Business Rules

7. To-Be Process

6. Optimize

# Chapter 1 Review

---

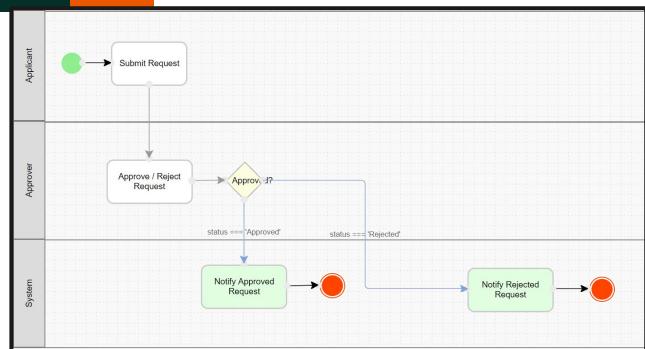
- General understanding on how to best design a Process.
- With the finished process design, one can now bring the finished process design/flow and implement it in Joget.

---

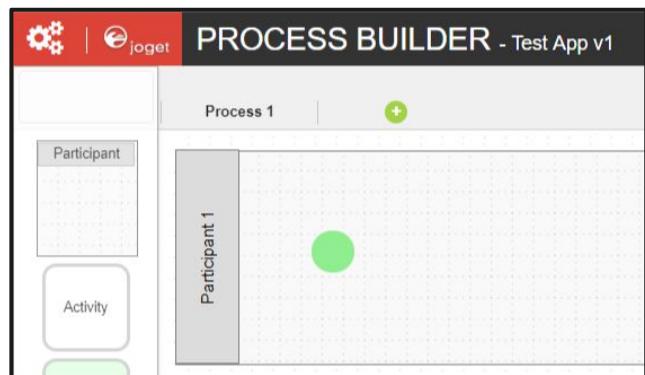
# Chapter 2

Designing your first Process  
with Process Builder

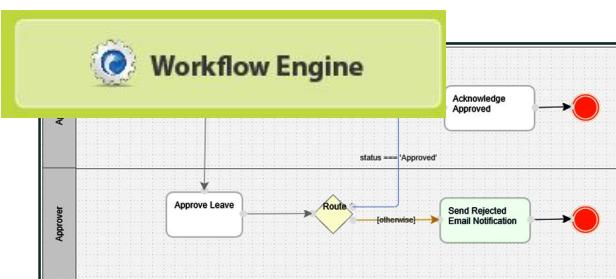
# Business Process Automation



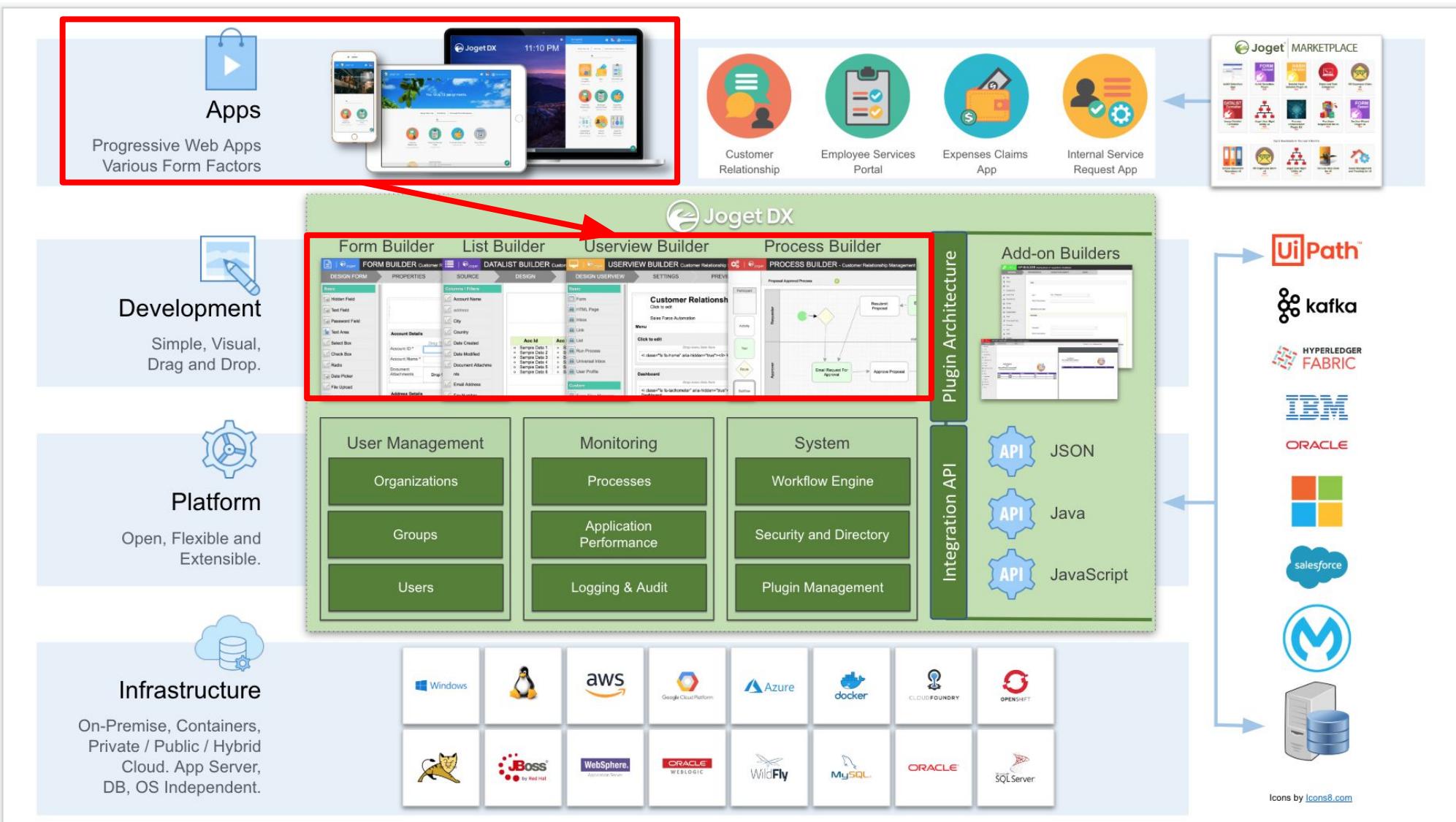
**Business Process Design**

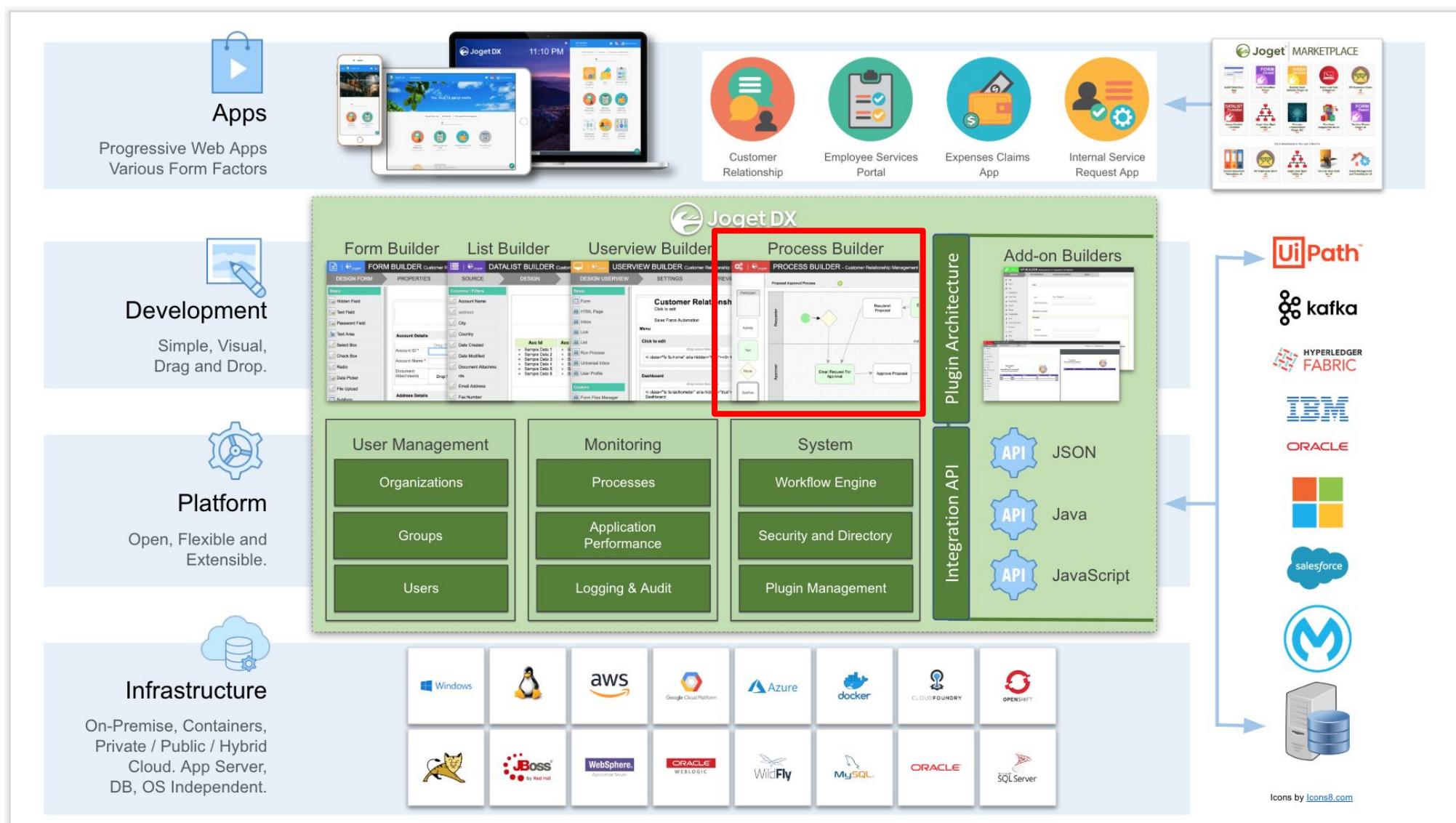


**Process Design**



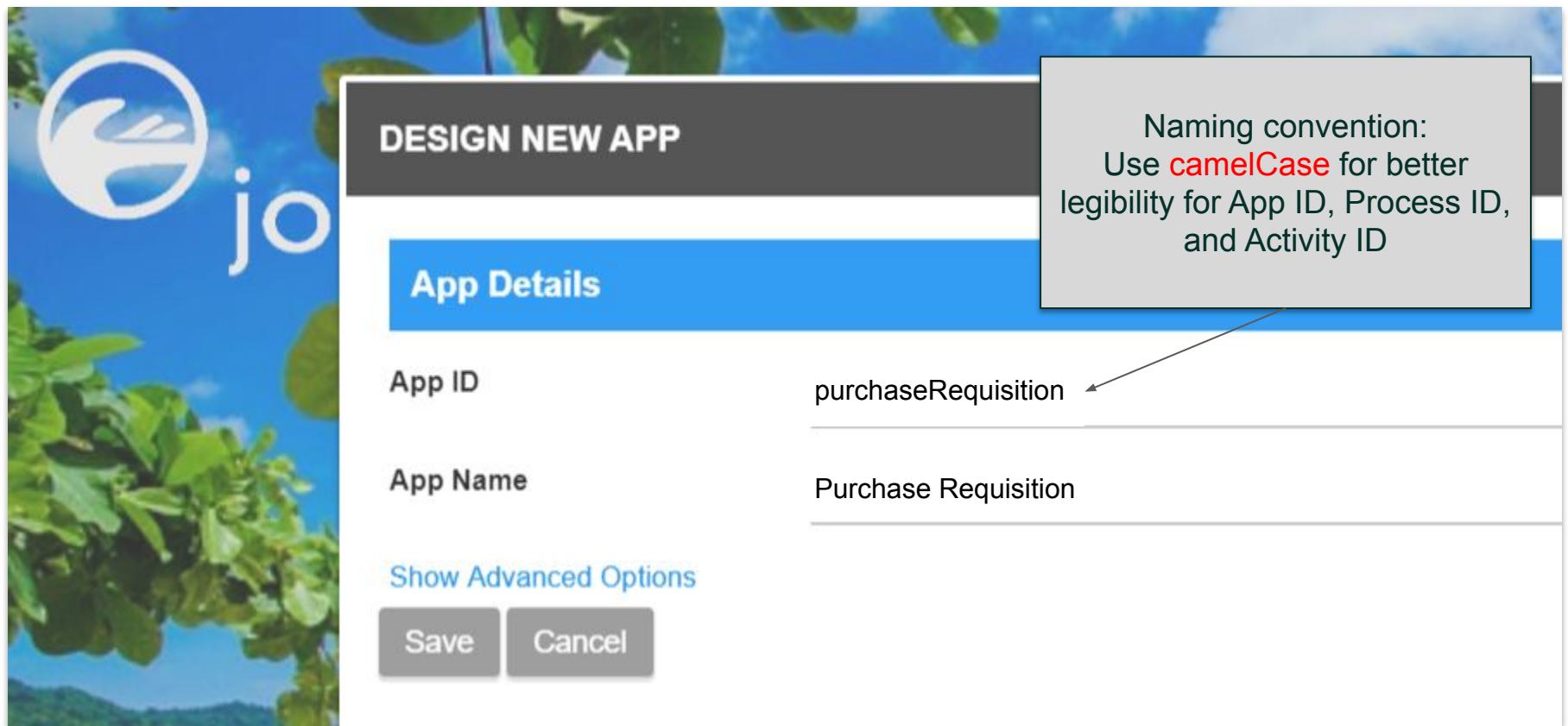
**Process**





# Create Your First App

- Design Apps -> Design New App



# Naming Convention

---

- Use **camel case** for App ID, Process ID, Activity ID.  
For example:-
  - **purchaseRequisition**
  - **purchaseRequestProcess**
  - **submitRequest**
- Do not use **snake case** as it conflicts with process instance ID naming convention:-  
`1_purchaseRequisition_purchaseRequestProcess_submitRequest`
- Maintain consistent naming convention that you have decided across the board.

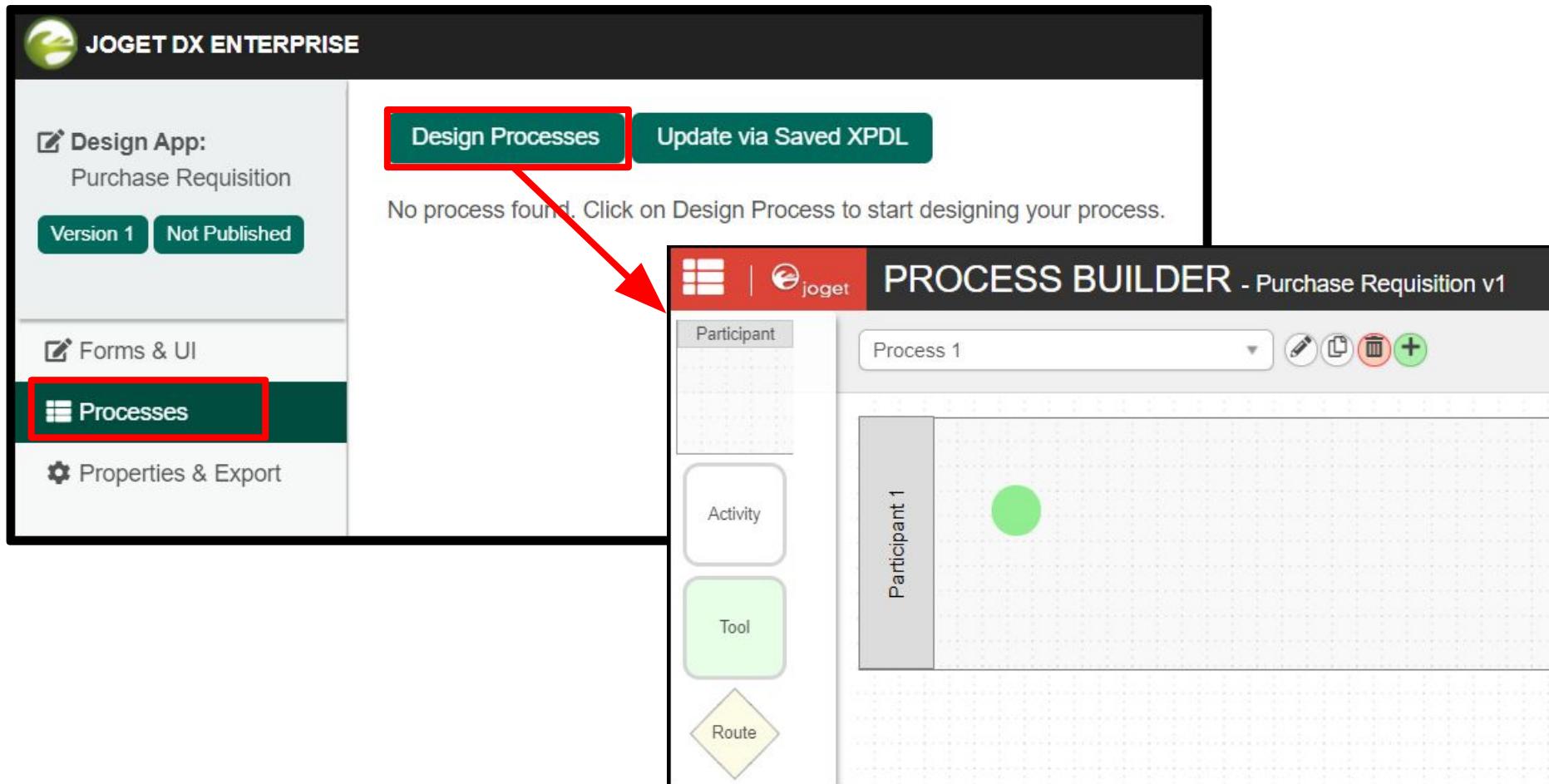
# Create Your First App

---

- Purchase Requisition Application
  - Purchase Request Process
    - Applicant **submits**
    - Approver **approves**
    - Applicant is **notified** of the request result

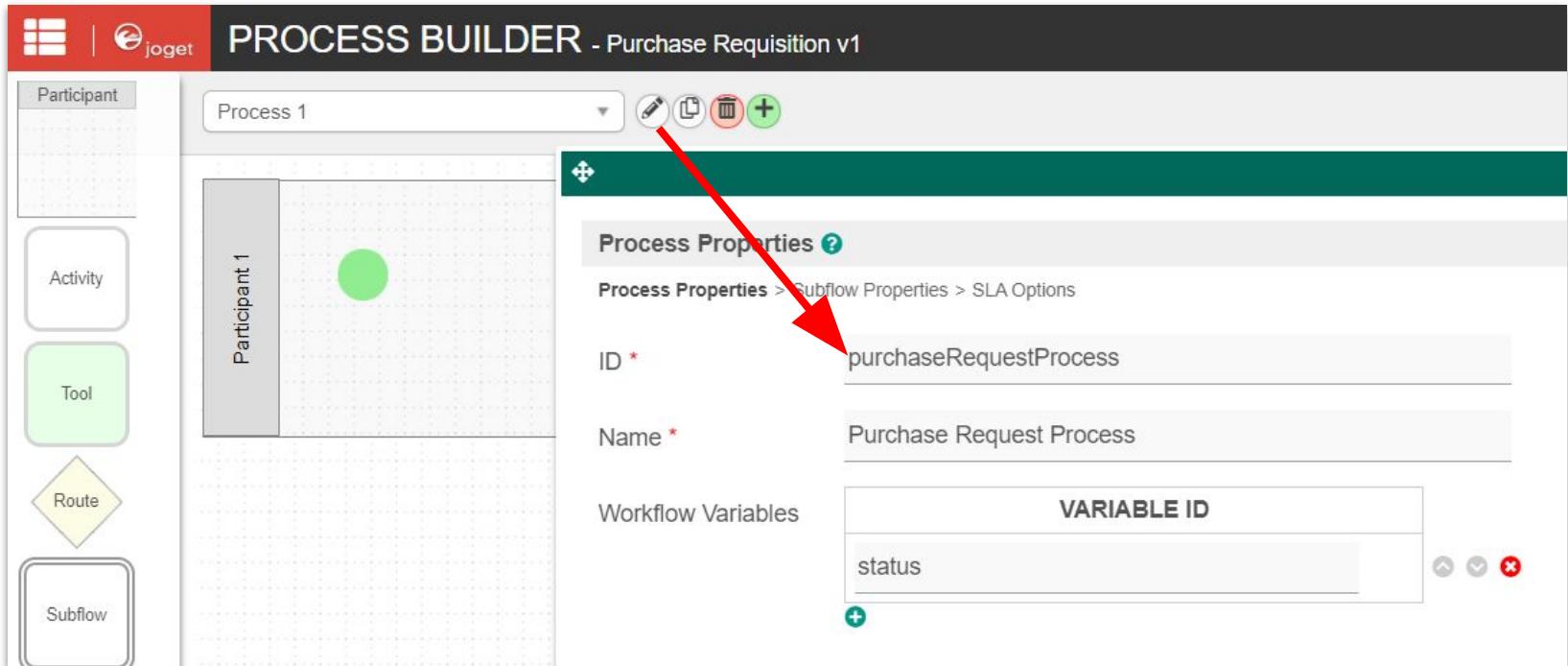
# Design Your First Process

- Launch Process Builder – Processes > Design Processes



# Defining Your Process

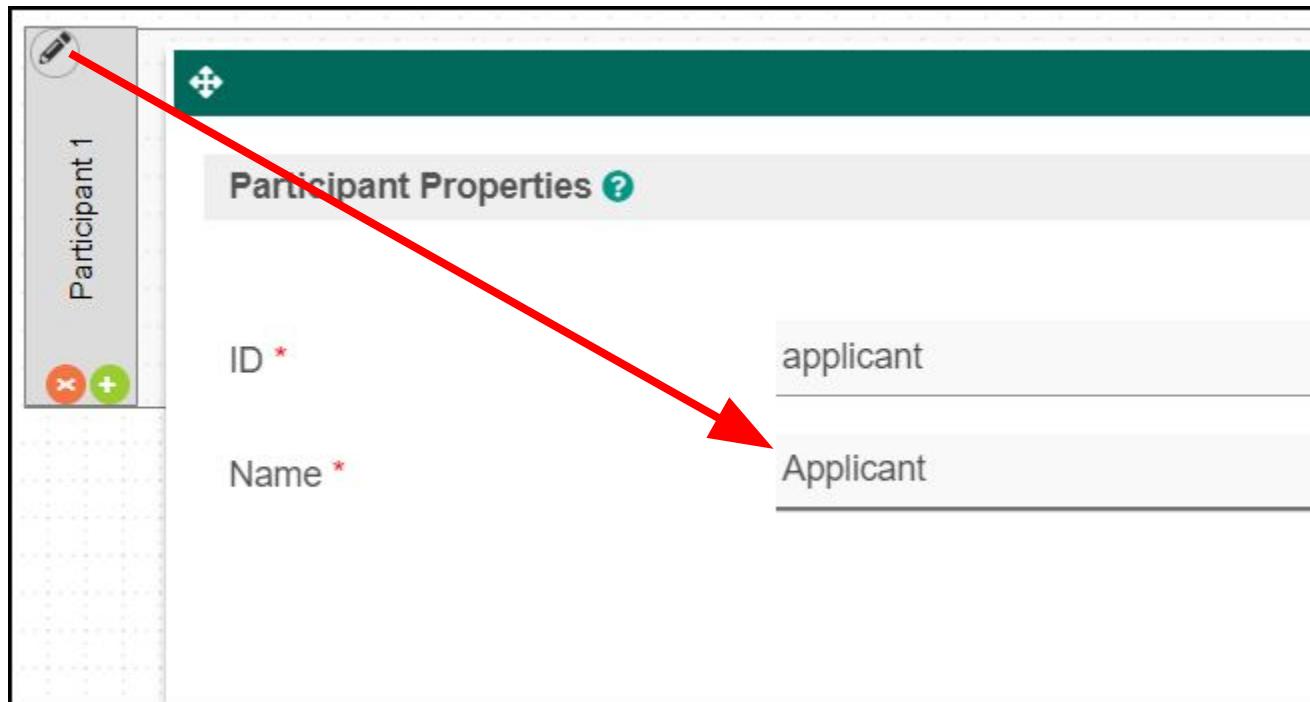
- Rename Process ID and Name
  - Use **camel case** for ID - **purchaseRequestProcess**
  - Do not use **snake case** as it conflicts with process instance ID naming convention



# Defining Your Participant

---

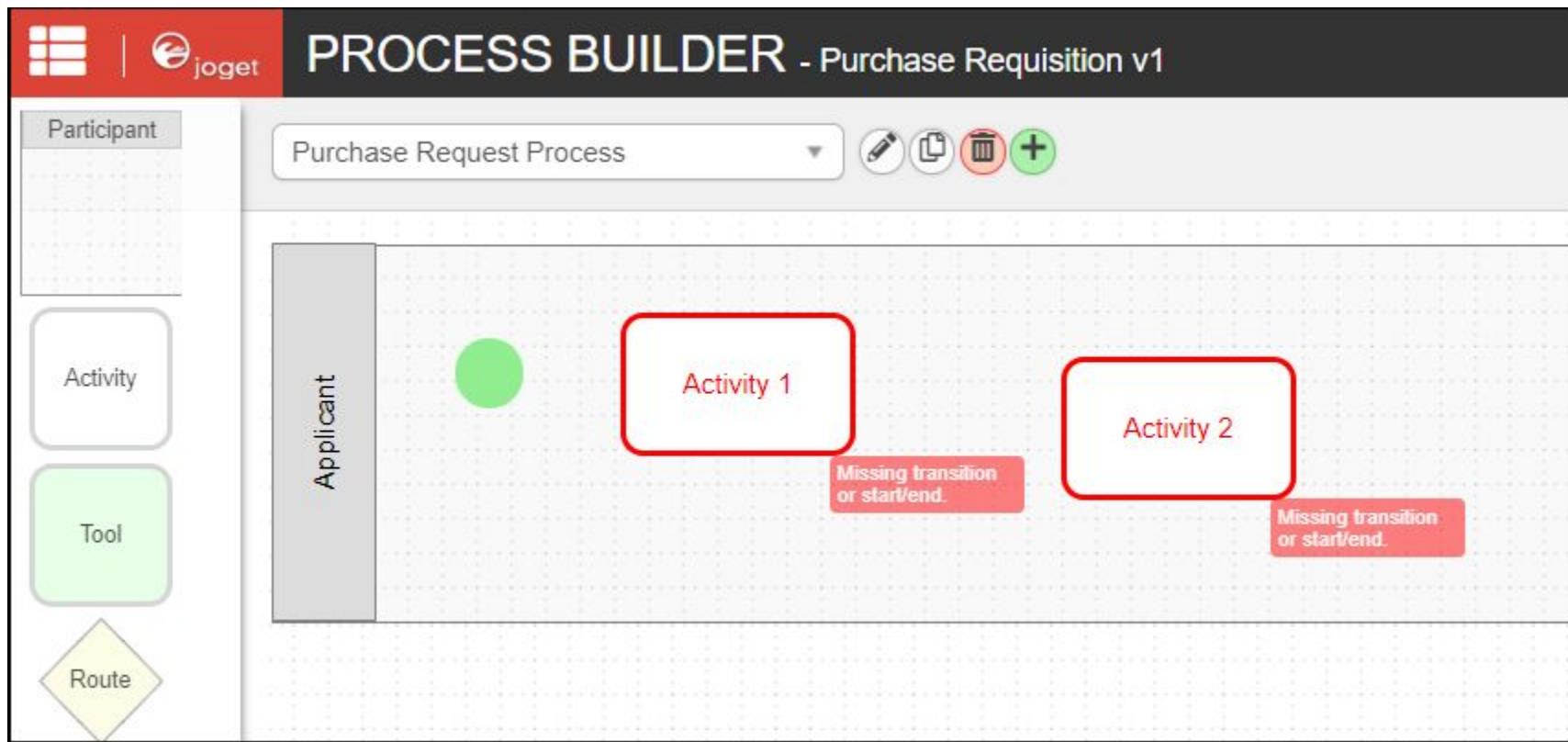
- Rename the existing participant to **Applicant**.



# Smart Guide

---

- The Process Builder will highlights logical errors as you design.



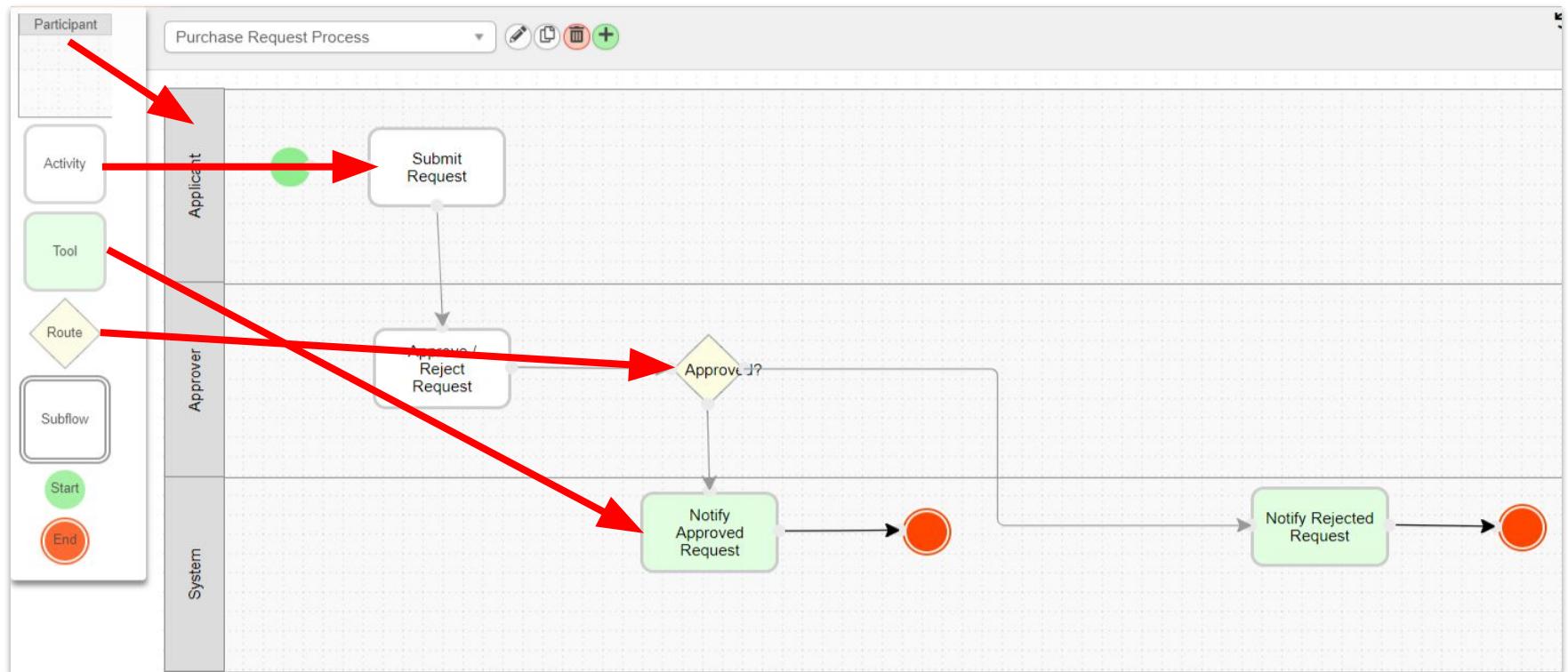
# About Smart Guide

---

- Advises only on possible design errors.
- Will **NOT** advise on flawed business logic, design inefficiency.

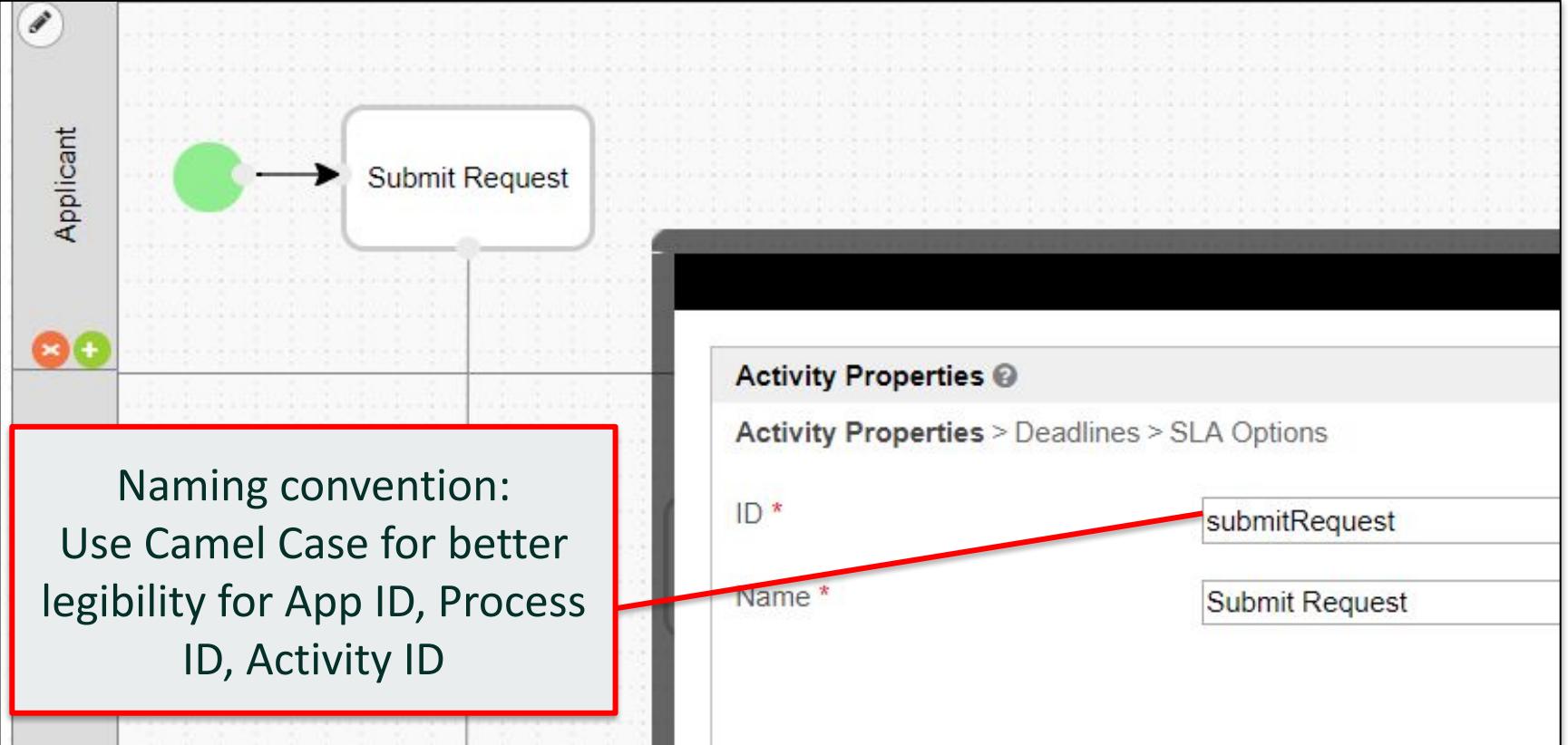
# Insert Activities and Transitions

- Drag and drop, select the appropriate nodes to build the Process.



# Rename Activity ID and Name

- Rename the activity node to make it meaningful.
- Use **camelCase** naming convention for ID (submitRequest).



The screenshot shows a BPMN process editor interface. On the left, there is a process diagram titled "Applicant" with a green start event and a "Submit Request" activity node. A red box highlights a tooltip message: "Naming convention: Use Camel Case for better legibility for App ID, Process ID, Activity ID". On the right, a modal window titled "Activity Properties" is open, showing the current values for "ID" (submitRequest) and "Name" (Submit Request). A red arrow points from the "Name" field in the modal back to the "Name" field in the tooltip message.

Naming convention:  
Use Camel Case for better  
legibility for App ID, Process  
ID, Activity ID

Activity Properties

Activity Properties > Deadlines > SLA Options

ID \*

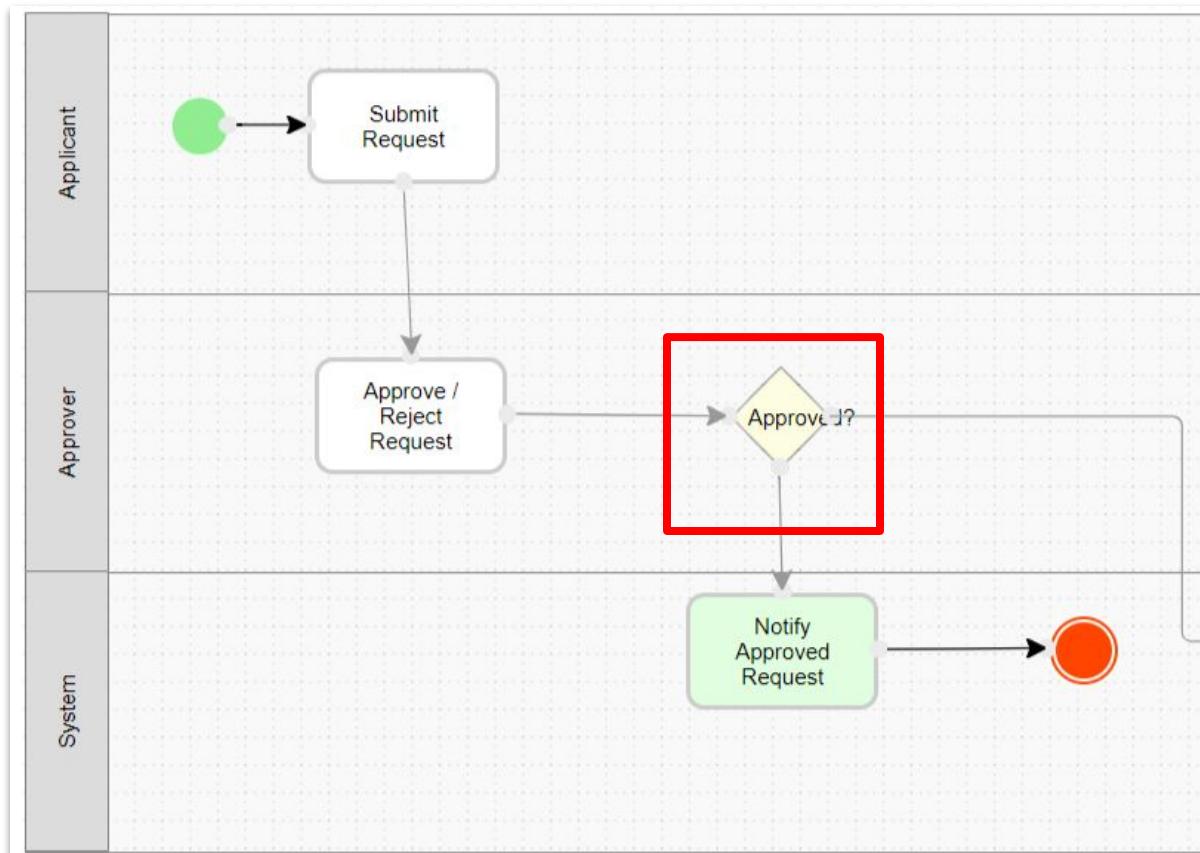
submitRequest

Name \*

Submit Request

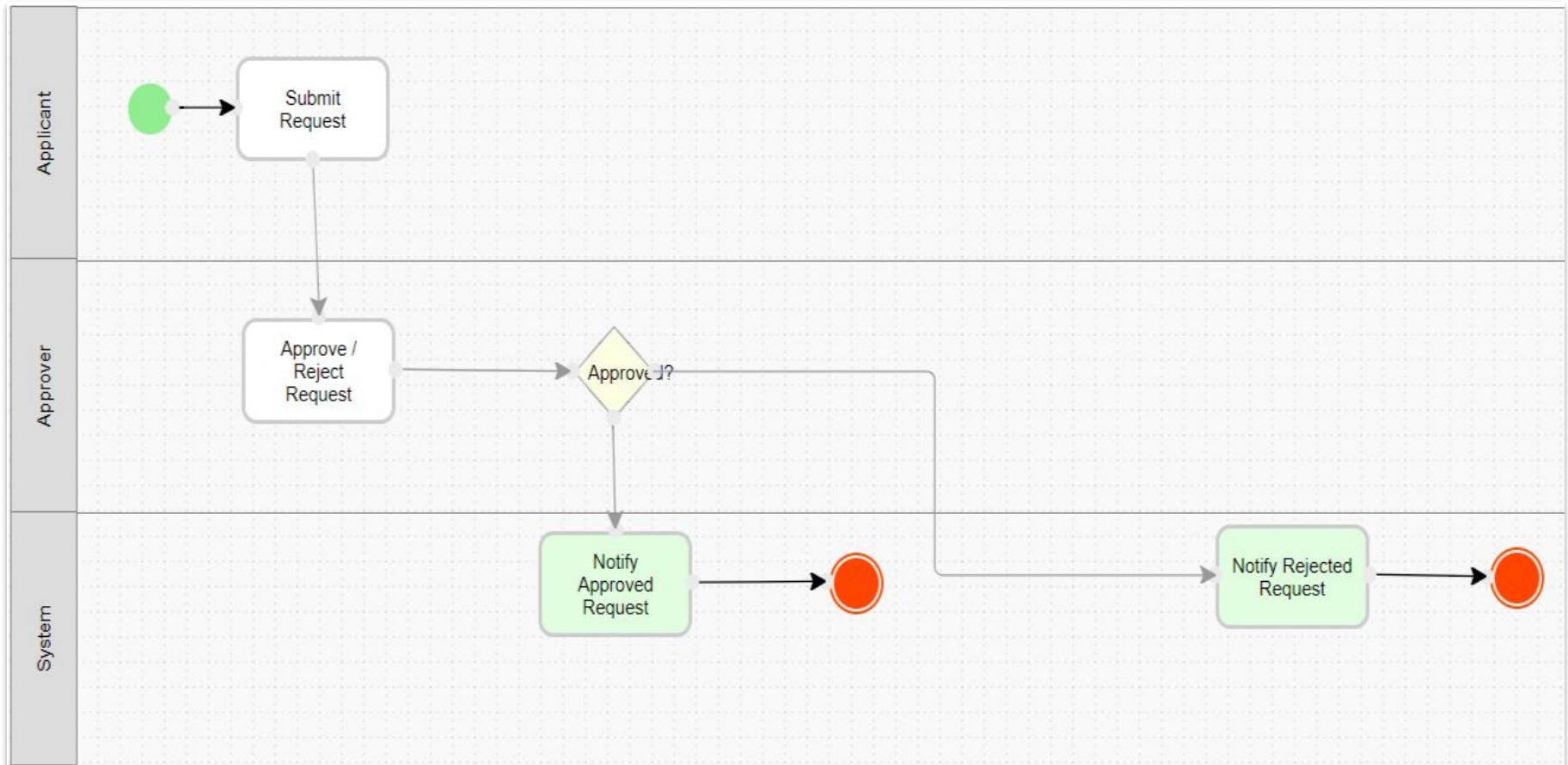
# Implementing the Business Rule

- Route: Diamond-shaped node – Used to implement business rules.



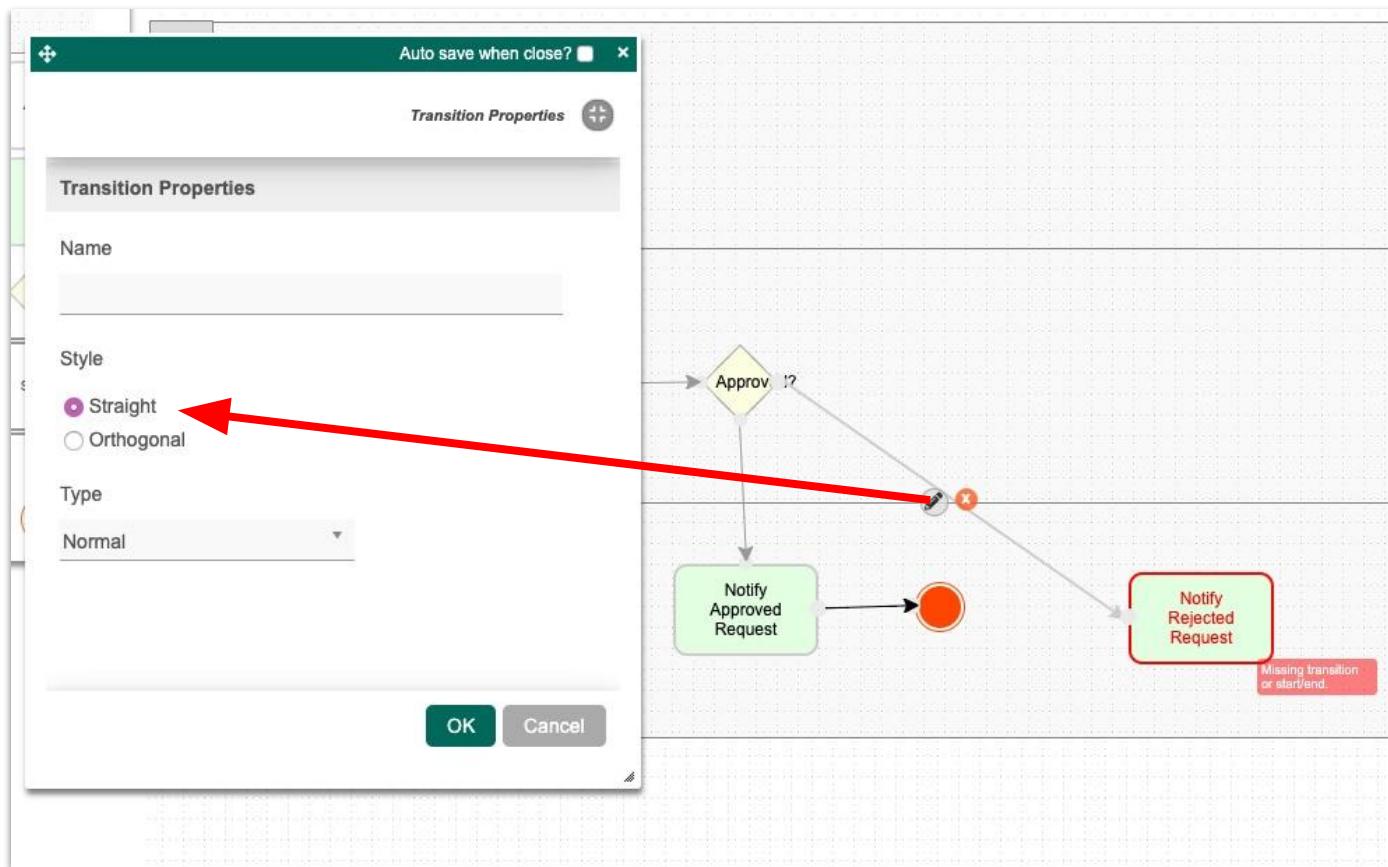
# Completing the Design

- Insert the rest of the activities.



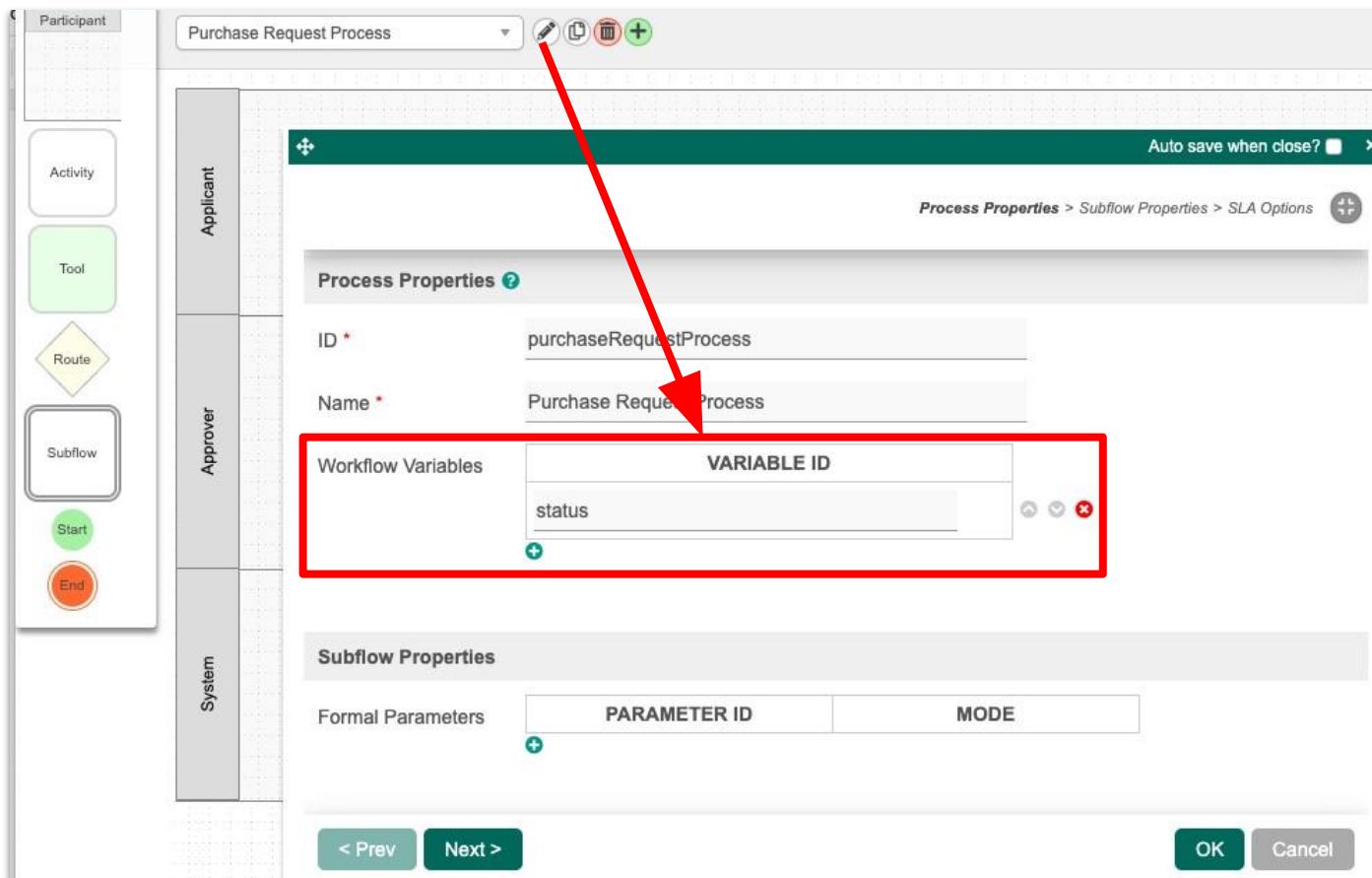
# Managing the Transition Style

- Set shape to Straight / Orthogonal to shape your transition.



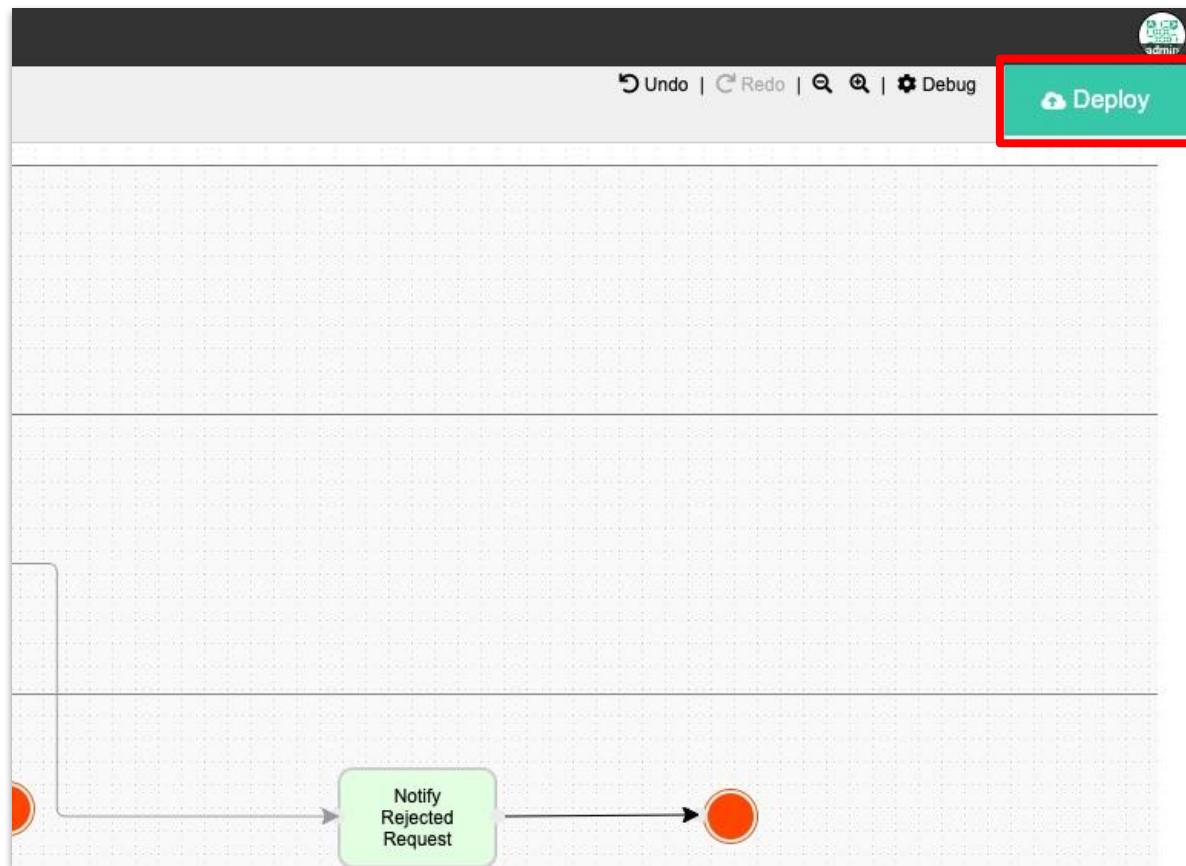
# Declaring the Workflow Variable

- Variables used in the Process must be declared in the Process's properties.



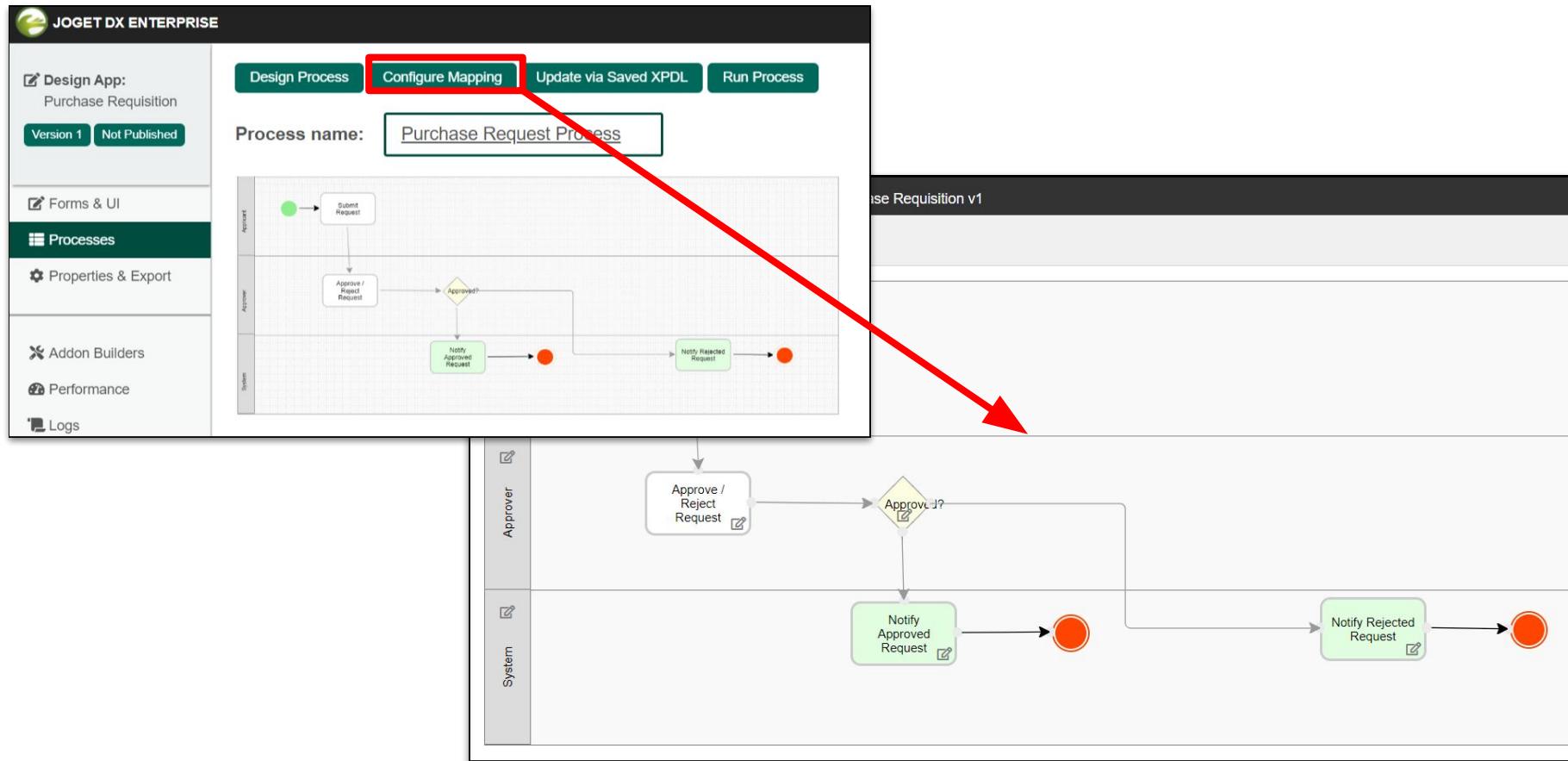
# Deploying Your Process Design

- On the top right corner of the designer, click on “Deploy” to push your finished design back into your Joget App.



# Implementing the Business Rules

- Launch Process Mapper.  
App Design > Processes > Configure Mapping



# Implementing the Business Rules

- Route: Diamond-shaped node – Used to implement business rules.

PROCESS MAPPER - Purchase Requisition v1

Purchase Request Process

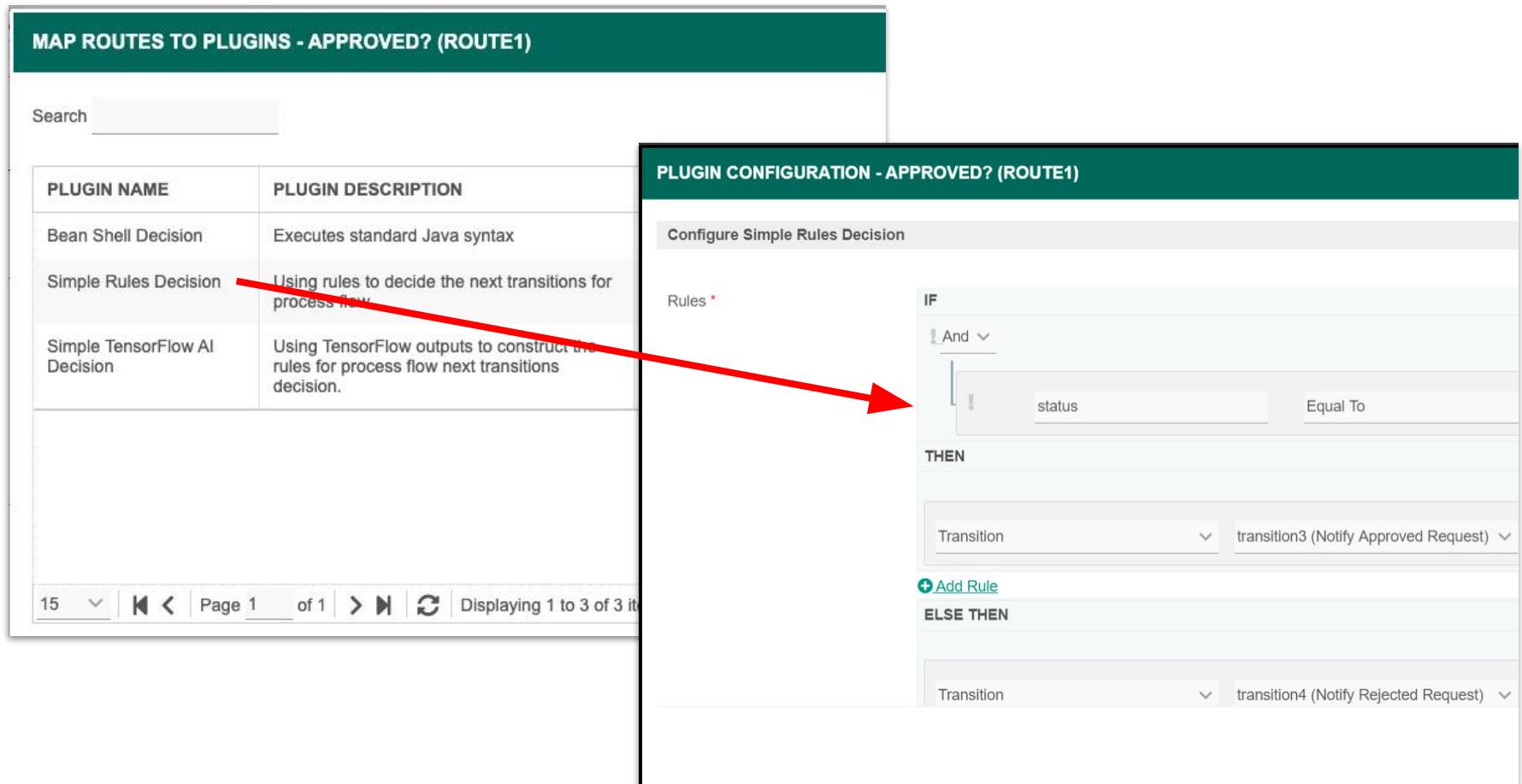
MAP ROUTES TO PLUGINS - APPROVED? (ROUTE1)

PLUGIN NAME	PLUGIN DESCRIPTION	PLUGIN VE
Bean Shell Decision	Executes standard Java syntax	7.0.0
Simple Rules Decision	Using rules to decide the next transitions for process flow.	7.0.0
Simple TensorFlow AI Decision	Using TensorFlow outputs to construct the rules for process flow next transitions decision.	7.0.0

15 | Page 1 of 1 | 45 | Displaying 1 to 3 of 3 items

# Implementing the Business Rules

- Choose Simple Rules Decision.



The screenshot illustrates the process of implementing business rules in the Joget platform. It consists of two main panels:

- Left Panel: MAP ROUTES TO PLUGINS - APPROVED? (ROUTE1)**
  - A search bar labeled "Search".
  - A table with columns "PLUGIN NAME" and "PLUGIN DESCRIPTION".
  - Items listed:
    - Bean Shell Decision: Executes standard Java syntax.
    - Simple Rules Decision**: Using rules to decide the next transitions for process flow. This item is highlighted with a red arrow pointing to the configuration screen.
    - Simple TensorFlow AI Decision: Using TensorFlow outputs to construct the rules for process flow next transitions decision.
  - Pagination controls at the bottom: "15" dropdown, navigation icons (back, forward), "Page 1 of 1", "Displaying 1 to 3 of 3 items".
- Right Panel: PLUGIN CONFIGURATION - APPROVED? (ROUTE1)**
  - A title "Configure Simple Rules Decision".
  - A section for "Rules \*".
  - An "IF" condition: "And" (selected) with "status" set to "Equal To".
  - An "THEN" transition: "Transition" set to "transition3 (Notify Approved Request)".
  - A link "+ Add Rule".
  - An "ELSE THEN" section: "Transition" set to "transition4 (Notify Rejected Request)".

# Setting the Rules for Transition Condition

- When the **status** variable is met with certain condition(s), route accordingly.

PLUGIN CONFIGURATION - APPROVED? (ROUTE1)

Configure Simple Rules Decision

Rules \*

IF

And

status Equal To Approved

THEN

Transition transition3 (Notify Approved Request)

+ Add Rule

ELSE THEN

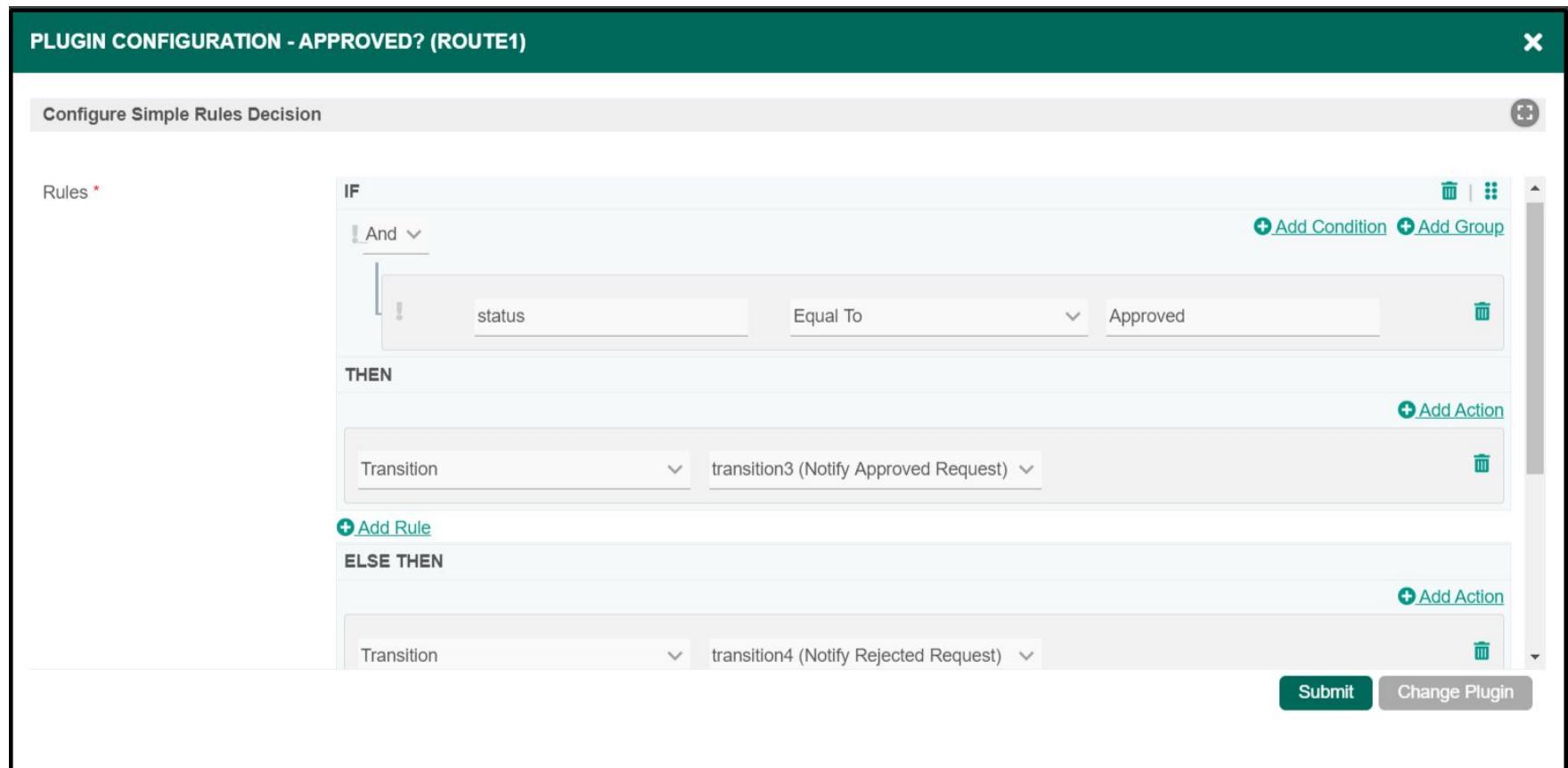
Transition transition4 (Notify Rejected Request)

+ Add Action

+ Add Condition + Add Group

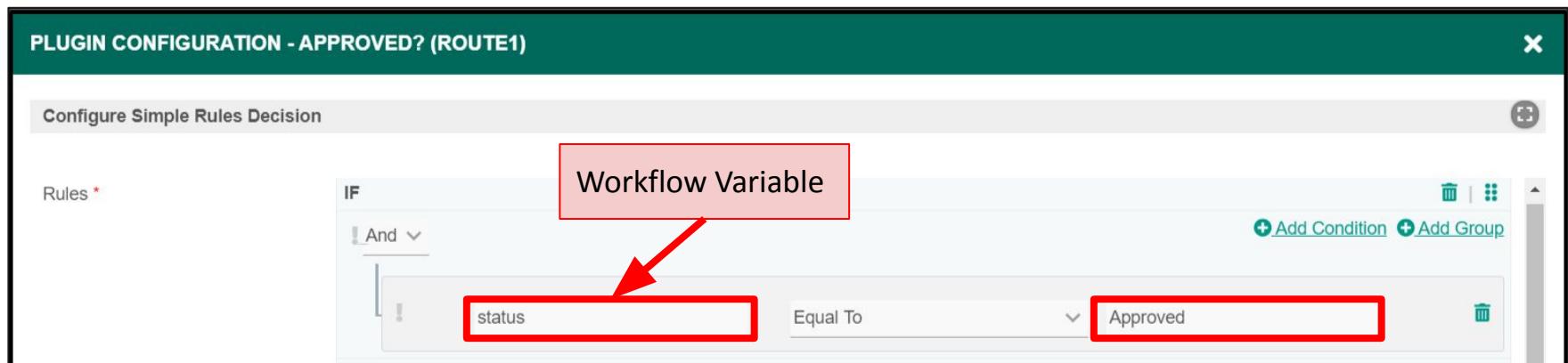
+ Add Action

Submit Change Plugin



# Simple Rules Decision Plugin

- Hash **variable** is supported in input fields as well for complex use cases.

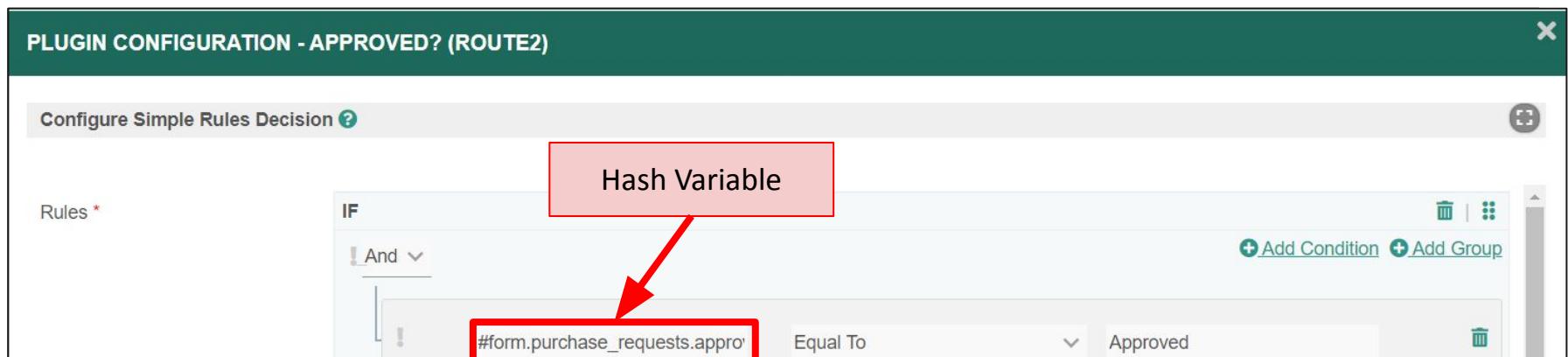


**Note:** This means you don't even need workflow variables to control the route!

e.g.: using Form Hash variable to read form field value to route

# Simple Rules Decision Plugin

- Hash **variable** is supported in input fields as well for complex use cases.



e.g.: using *Form Hash Variable* to read form field value to route  
**#form.purchase\_requests.approval#**

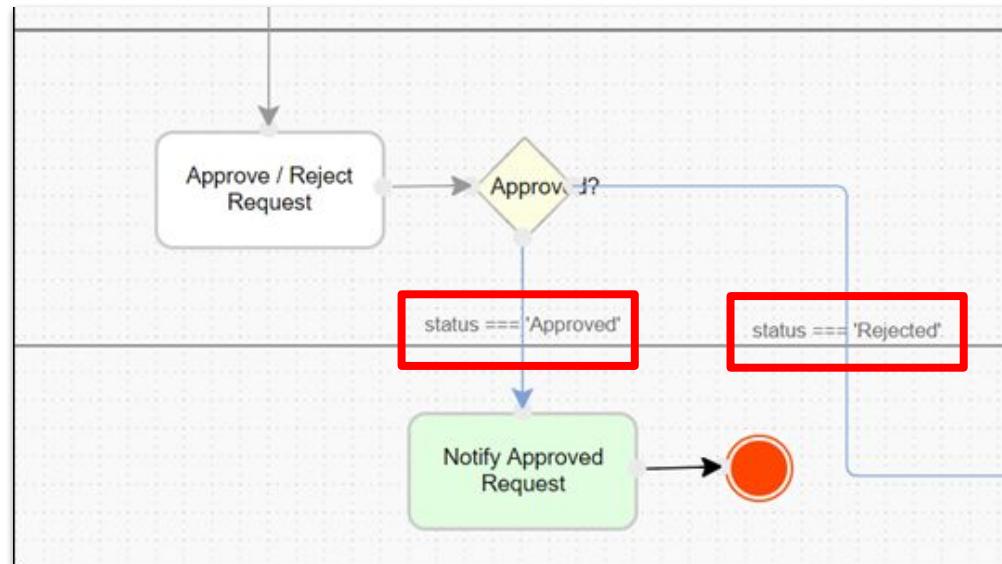
# If and otherwise...

---

- Does this resemble the typical programming if-else condition block?
- Is this the best design?
- Why not if-else-if-else everything? (In other words, explicitly set all possible conditions)

# Alternative Business Rule Implementation

- Prior to DX, business rules implementation must be done in the Process Builder.
- This approach is **no longer recommended** as it is difficult to change business logic without redeploying the whole process design of the entire app.



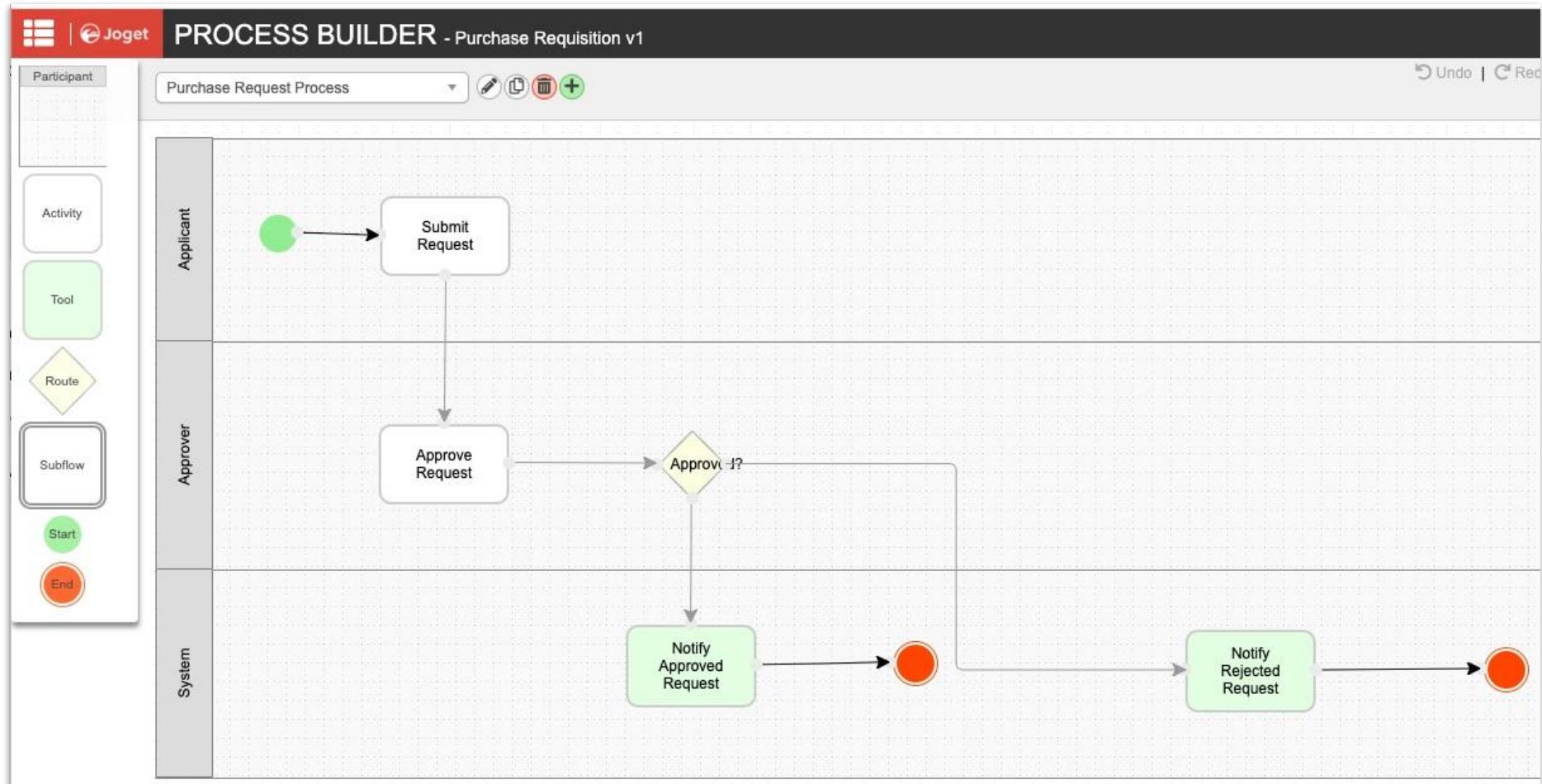
# Advantages of using Process Mapper

---

- No need to constantly deploy the Process Builder when changing the business logic.
- Decision Tool Plugin takes precedence over Process Builder conditional setup.

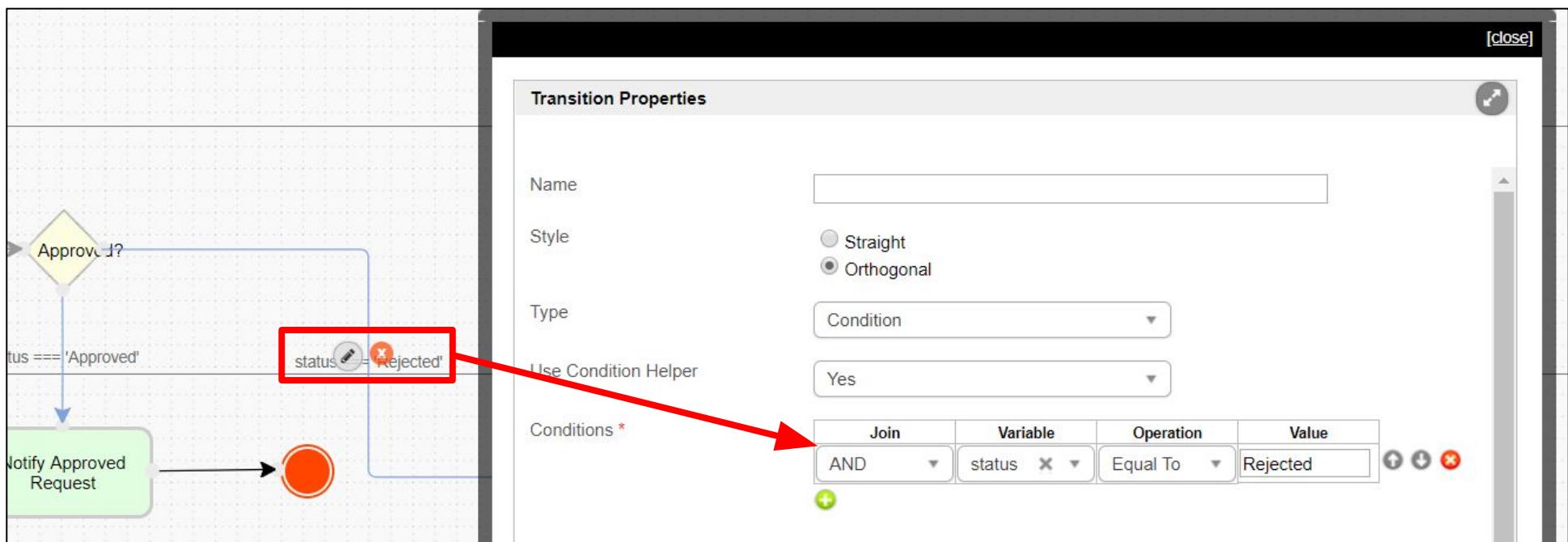
# Alternative Business Rule Implementation

- App Design > Processes > Design Process.



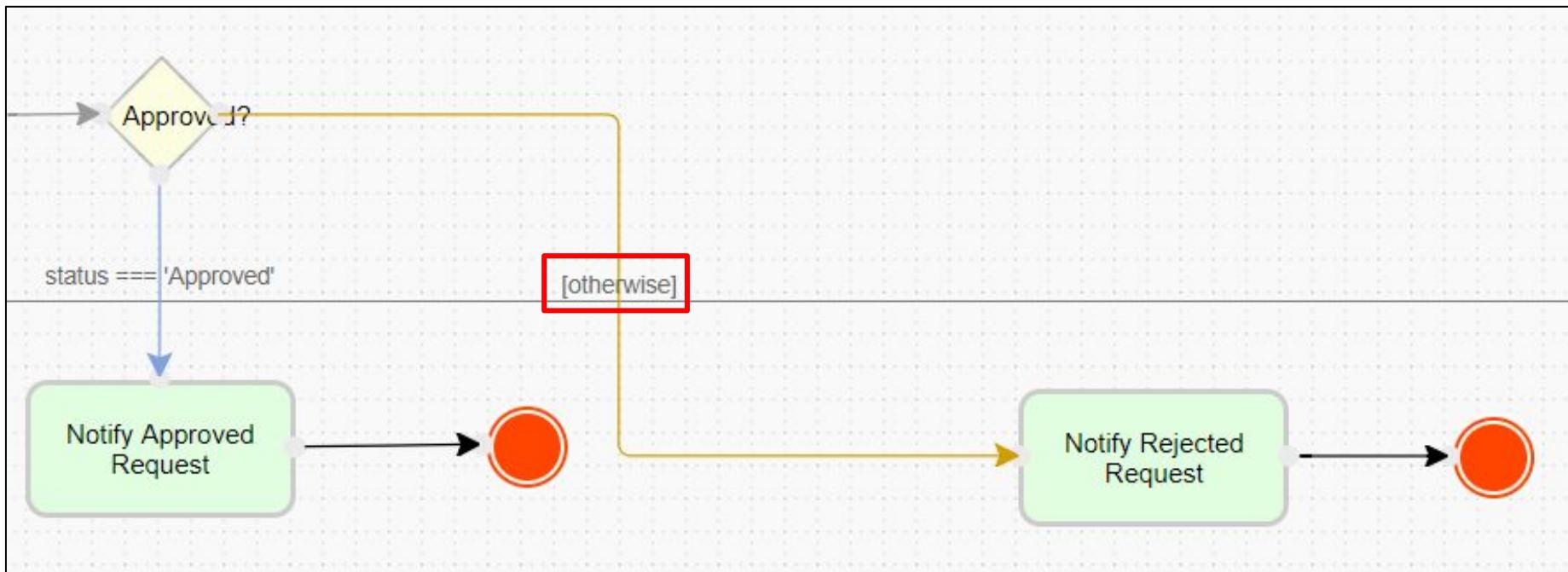
# Alternative - Defining the Transition Condition

- When the **status** variable is met with certain condition(s), route accordingly.



# Alternative - Anything else...

- Handling the “otherwise”, if any.



Note: In this case, “otherwise” is also correct. All conditions are set explicitly for educational purposes.

# Good to Know

---

- Expressions in “Condition” are evaluated in JavaScript.
- Workflow variable are stored as data type **string**.

# Good to Know

- Native Javascript functions are accepted. For example...
  - `status == "Approved" && parseInt(balance) > 0`
- Make sure of debugger tool such as **Firebug / Developer Tools** to test the expression before deploying.



The screenshot shows the Firebug JavaScript Console interface. The toolbar at the top includes icons for file operations, a search bar, and tabs for Console, HTML, CSS, Script, DOM, Net, and Cookies. The main area displays a script block with numbered lines of code. The code checks if a balance is 2 and the status is Approved, then logs OK or KO to the console based on the condition.

```
balance = "2"; status = "Approved"; if( status == "Approved" && parseInt(balance) > 0){    console.log("OK");}else{    console.log("KO");}
```

# Chapter 2 Review

---

- Create a new App.
- Design and deploy Process design into the App.
- Understand the naming convention.
- Define various kind of process design elements.
- Define workflow variable.
- Deploying process design to Joget.

---

# Chapter 3

## Running your First Automated Process

# How to Test?

---

- There are 2 ways in general.
  - Make use of the “**Run Process**” button in App Design > Processes (only for admin)
  - Make use of **Userview** (for all users)

# How to Test?

---

- There are 2 ways in general.
  - Make use of the “**Run Process**” button in App Design > Processes (only for admin)
  - Make use of **Userview** (for all users)

# Verify the Process Using Run Process

- Run through the Process to verify.

JOGET DX ENTERPRISE

Design App:  
Purchase Requisition  
Version 1 Not Published

Forms & UI

**Processes**

Properties & Export

Addon Builders

Performance

Logs

Design Process   Configure Mapping   Update via Saved XPDL   **Run Process**

Process name: Purchase Request Process

The process diagram illustrates a workflow for handling purchase requests. It starts with an 'Applicant' submitting a request. This leads to an 'Approver' who can either 'Approve / Reject Request'. If approved, the system sends a 'Notify Approved Request' message. If rejected, the system sends a 'Notify Rejected Request' message. Both paths lead to a final red circular end state.

```
graph TD; Start(( )) --> SubmitRequest[Submit Request]; SubmitRequest --> ApproveRejectRequest[Approve / Reject Request]; ApproveRejectRequest --> Decision{Approved?}; Decision -- Approved --> NotifyApproved[Notify Approved Request]; Decision -- Rejected --> NotifyRejected[Notify Rejected Request]; NotifyApproved --> End(( )); NotifyRejected --> End;
```

# Verify the Process

- Complete the assignment to proceed.

JOGET DX ENTERPRISE

Design App:  
Purchase Requisition

Version 1 Not Published

Forms & UI

Processes

Properties & Export

PURCHASE REQUEST PROCESS - SUBMIT REQUEST

status

Complete

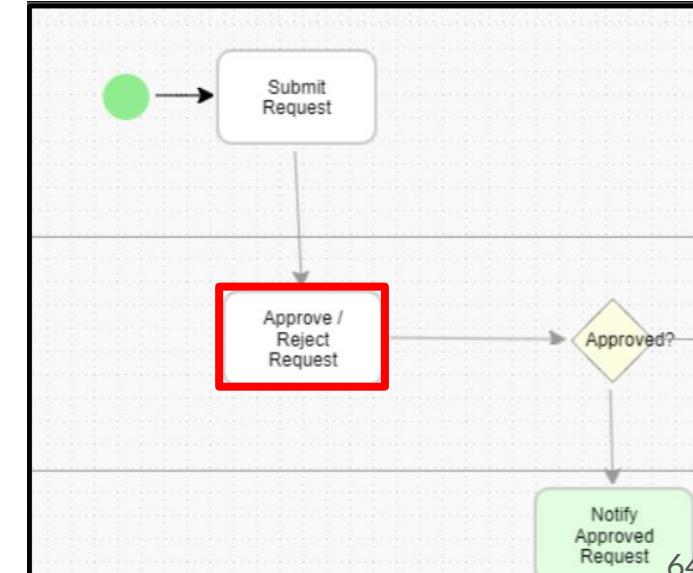
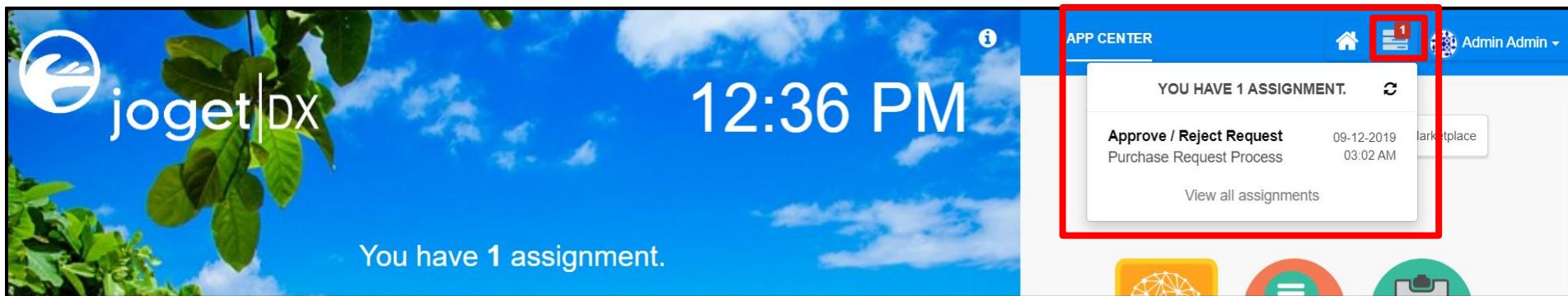
HINT: Activity name is configured in Process Builder

```
graph TD; Start(( )) --> Submit[Submit Request]; Submit --> Approval{Approve / Reject Request}; Approval --> Decision{Approved?}; Decision -- Approved --> Notify[Notify Approved Request]; Decision -- Rejected --> End(( ));
```

The process starts with a green circle representing the start event, which leads to a rectangular activity labeled "Submit Request". An arrow points from "Submit Request" to a rounded rectangle labeled "Approve / Reject Request". From "Approve / Reject Request", the flow continues to a diamond labeled "Approved?". If the answer is "Approved", the process moves to a green rounded rectangle labeled "Notify Approved Request". If the answer is "Rejected", the process ends at a green circle representing the end event.

# Accessing the Assignments

- App Center -> Universal Inbox



# Determining the Flow Using Route

- Manipulate the Process using Workflow Variable and complete the process.



# Exercise

---

- Run through the process from the start again to test out the “Rejected” case scenario.
- The last chapter will inform you how to be *exactly* sure if your process flows properly.

# Good to know

---

- At this point of time, the App is NOT published yet, therefore end user would not be able to run it.
- “Run Process” button at the App Designer’s Processes tab is a convenient way to verify your process flow.

# How to Test?

---

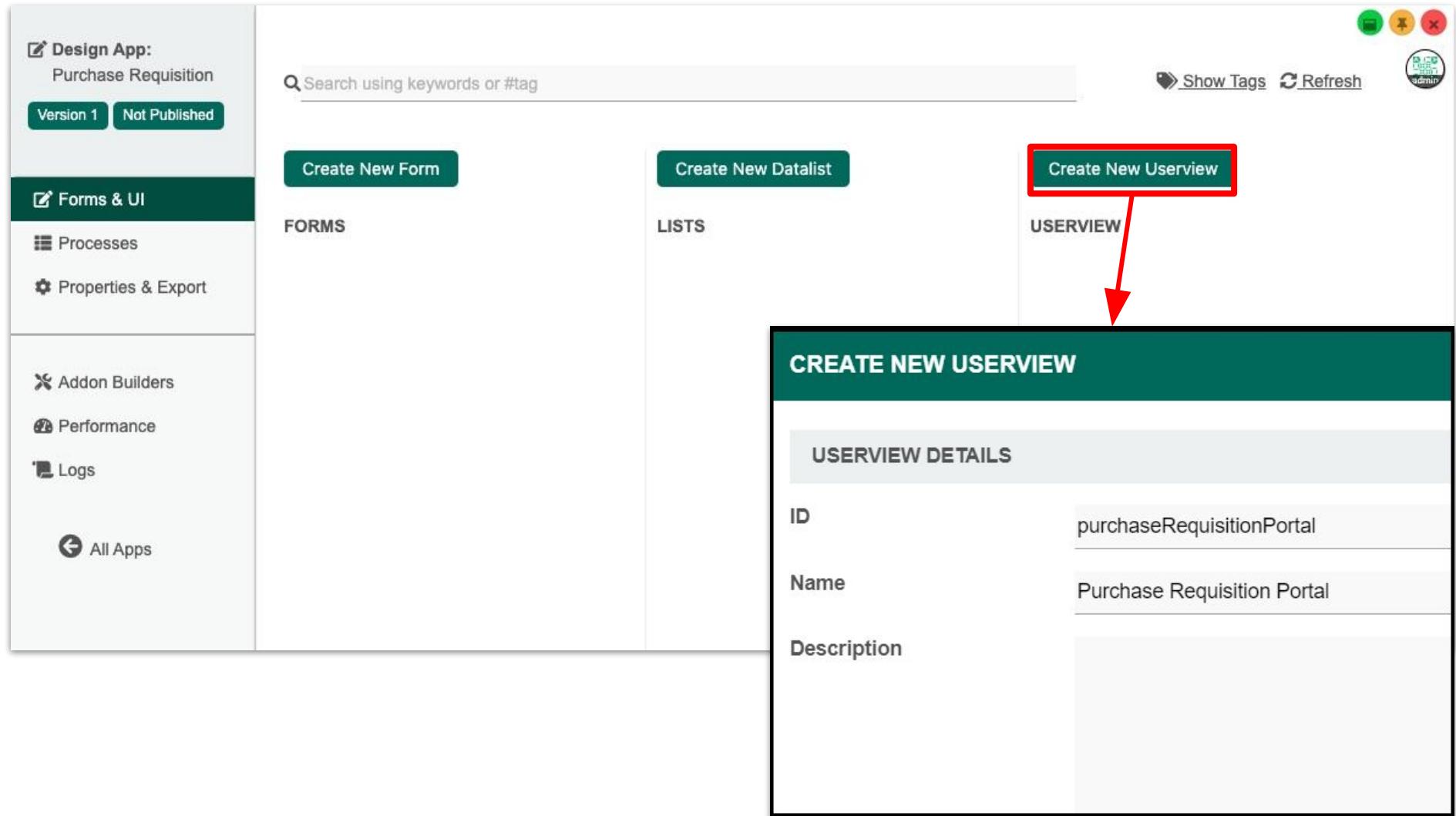
- There are 2 ways in general.
  - Make use of the “**Run Process**” button in App Design > Processes (only for admin)
  - **Make use of Userview (for all users)**

# Verify the Process Using Userview

---

- Create a new userview.
- Add **Run Process** and **Inbox** element.
- **Publish** the App.
- Try running the whole process again by accessing the userview from the app center.

# Create New Userview



The screenshot shows the Joget platform interface for creating a new Userview. On the left, there is a sidebar with the following menu items:

- Design App:
  - Purchase Requisition
  - Version 1 Not Published
- Forms & UI
  - Processes
  - Properties & Export
- Addon Builders
- Performance
- Logs
- All Apps

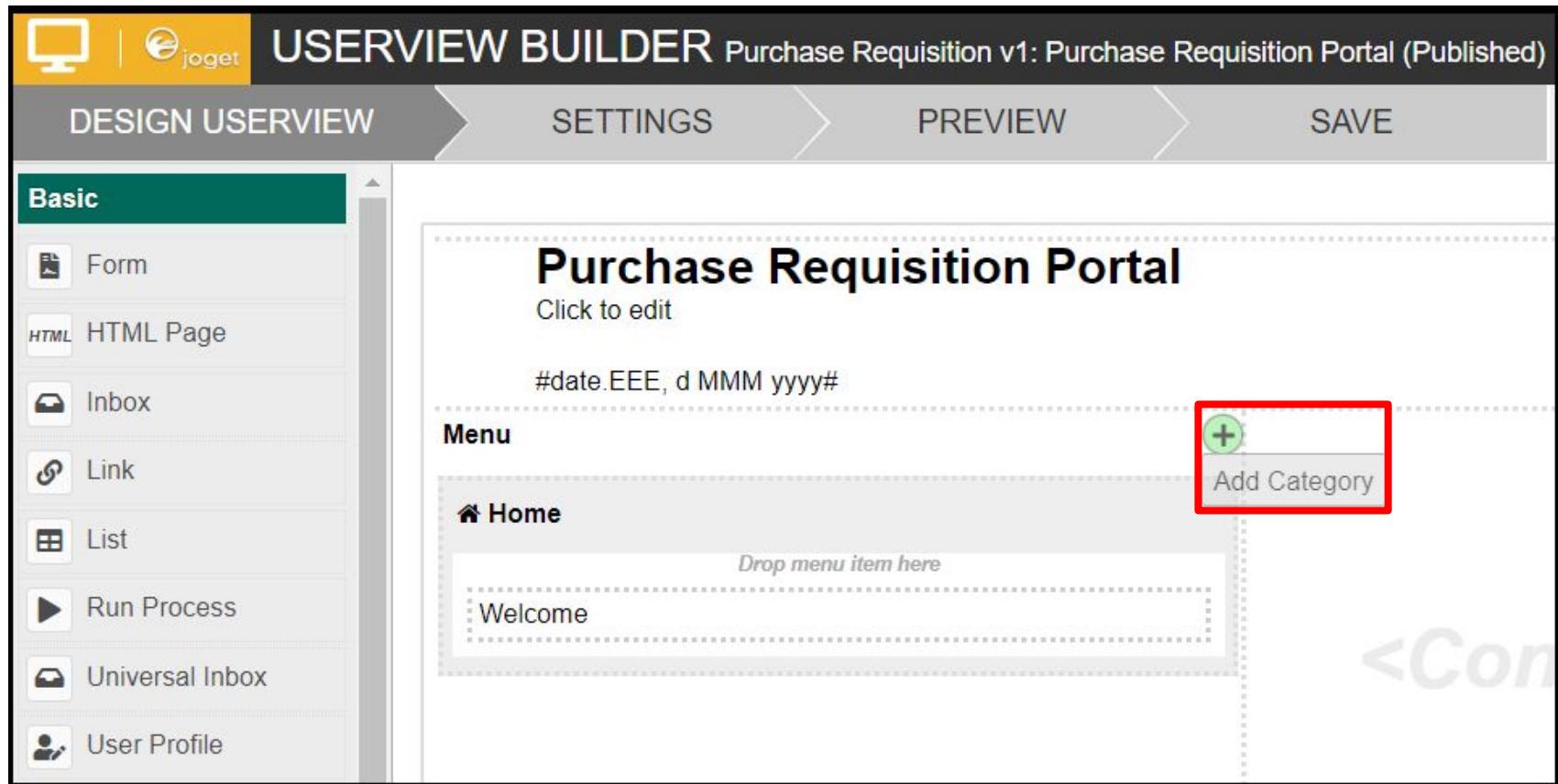
The main area is divided into three sections: FORMS, LISTS, and USERVIEW. Each section has a "Create New" button: "Create New Form" for FORMS, "Create New Datalist" for LISTS, and "Create New Userview" for USERVIEW. The "Create New Userview" button is highlighted with a red box and a red arrow points to it from the bottom right.

The "CREATE NEW USerview" dialog box is open, showing the following details:

USERVIEW DETAILS	
ID	purchaseRequisitionPortal
Name	Purchase Requisition Portal
Description	(empty)

# Userview Category

- Items are organized by Category.
- Add 1 new category – Purchase Requisition.

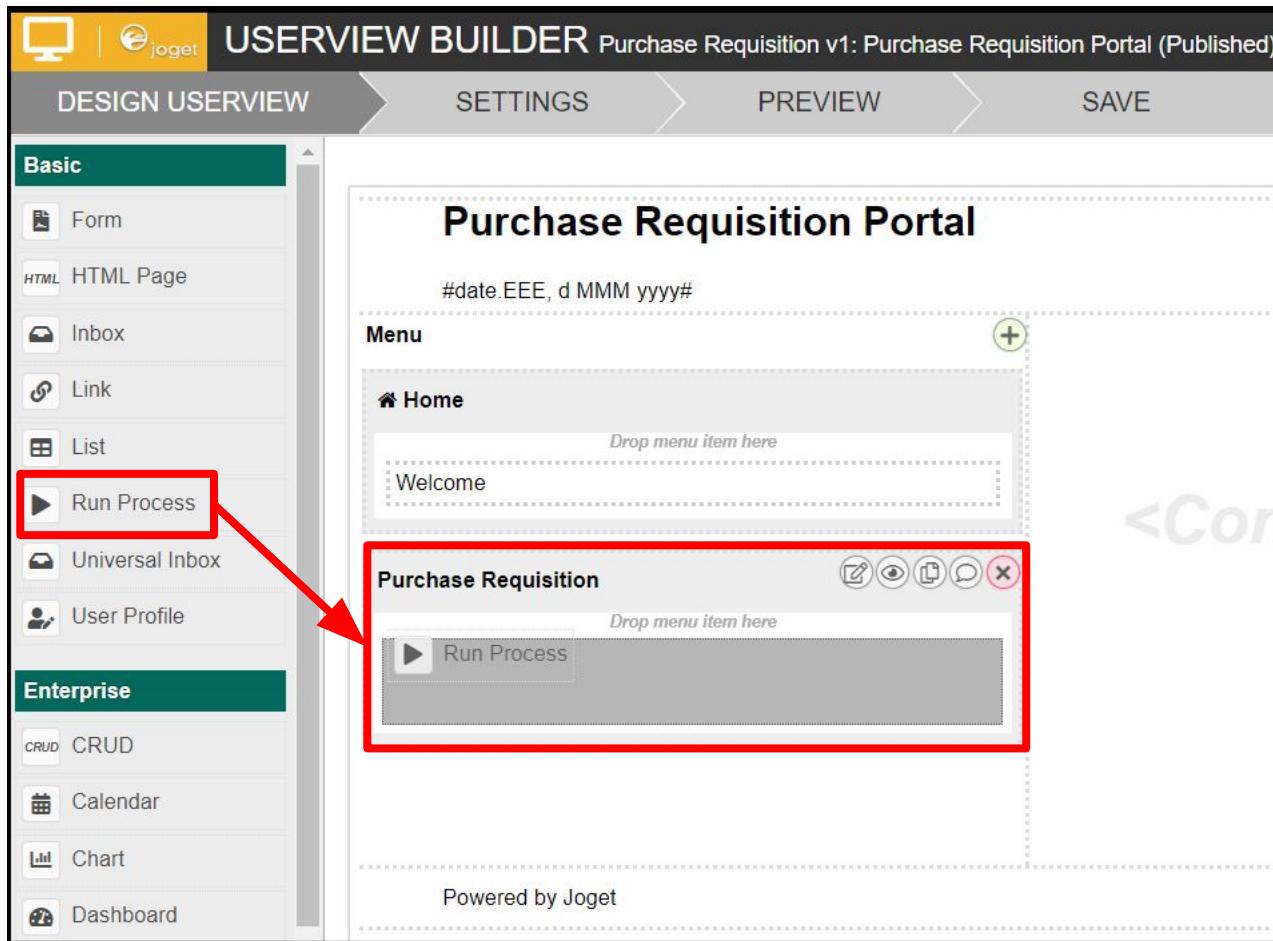


The screenshot shows the Userview Builder interface with the following details:

- Header:** USERVIEW BUILDER Purchase Requisition v1: Purchase Requisition Portal (Published)
- Navigation:** DESIGN USERVIEW → SETTINGS → PREVIEW → SAVE
- Left Sidebar (Basic category):** Form, HTML Page, Inbox, Link, List, Run Process, Universal Inbox, User Profile.
- Main Content:** Purchase Requisition Portal. It includes a timestamp (#date.EEE, d MMM yyyy#), a menu section with "Home" and a placeholder "Welcome", and a "Drop menu item here" area.
- Right Side (highlighted):** An "Add Category" button with a green plus sign icon, which is highlighted with a red box.

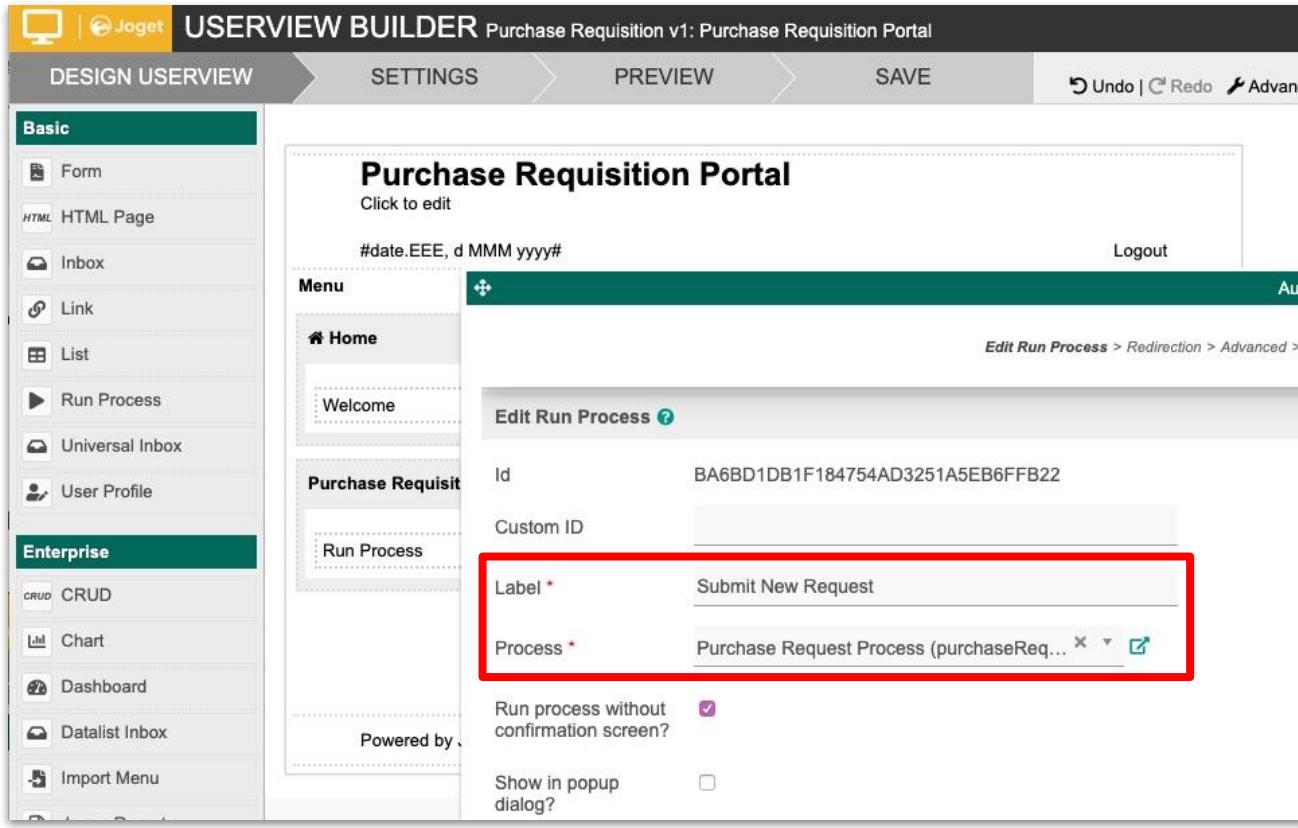
# Add Run Process Into Userview

- Drag-n-drop Run Process into Purchase Requisition category.



# Integrate Run Process Into Userview

- Set the appropriate label.
- Select Purchase Request Process in Process.



The screenshot shows the Joget Userview Builder interface. The top navigation bar includes tabs for DESIGN USerview, SETTINGS, PREVIEW, and SAVE, along with Undo, Redo, and Advanced buttons. The main workspace displays a userview titled "Purchase Requisition Portal". On the left, a sidebar lists various userview components under "Basic" and "Enterprise" categories. A red box highlights the configuration panel for a "Run Process" element. The panel shows the "Label" field set to "Submit New Request" and the "Process" field set to "Purchase Request Process (purchaseReq...)" with a dropdown arrow. Other visible settings include "Run process without confirmation screen?" (checked) and "Show in popup dialog?" (unchecked).

# Add Inbox Into Userview

---

- Add **Inbox** into **Purchase Requisition** category.
- Configure it to show **All Assignments**.

# Publish the App

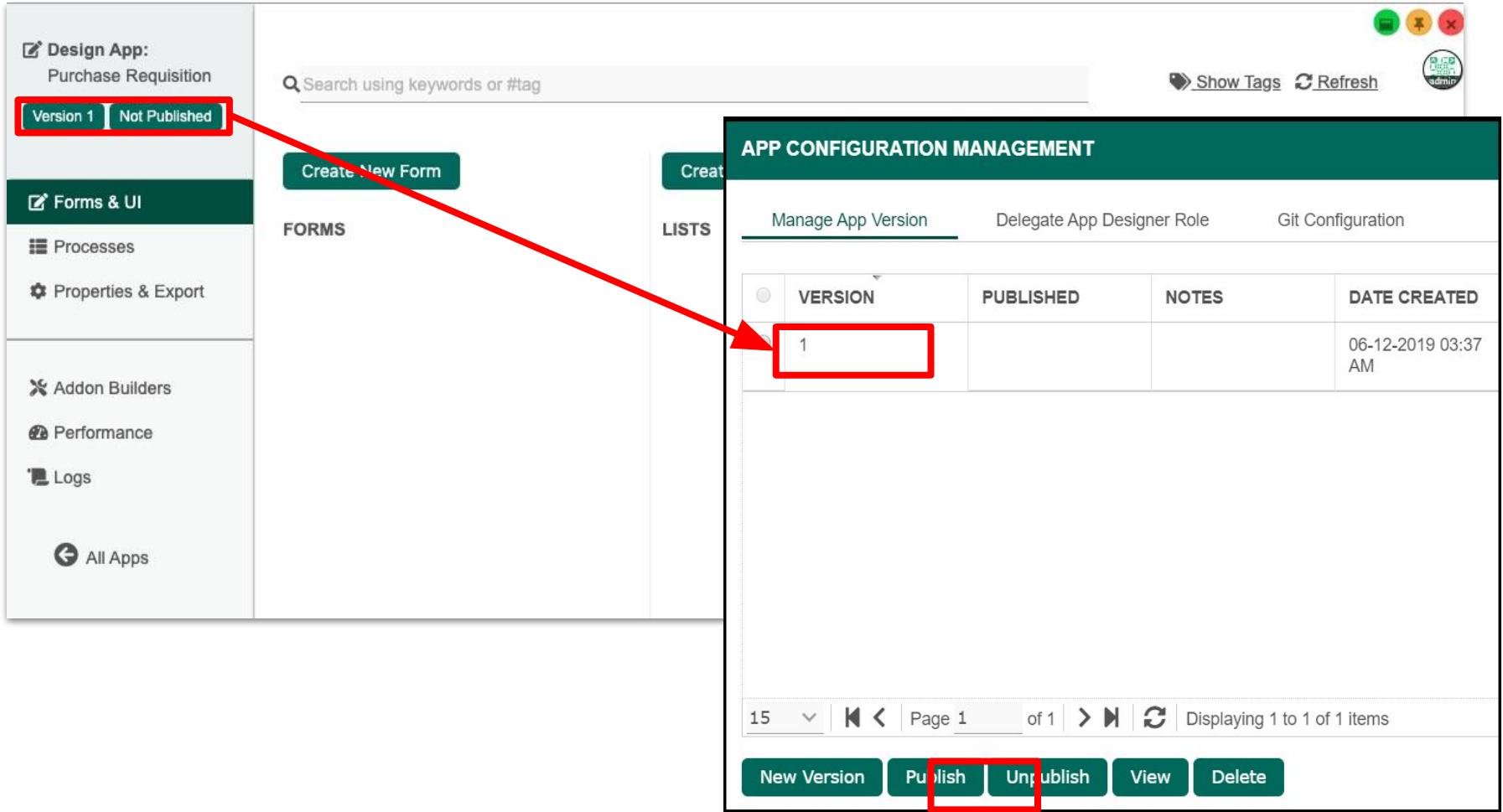
---

- With the Userview created, it is now ready for prime time.
- End users can now access them through this interface when the App is made published.

Note: Only **ONE** App version can be made published at any point of time.

# Publish the App

- Click on Version/Publish button, select the version to publish.



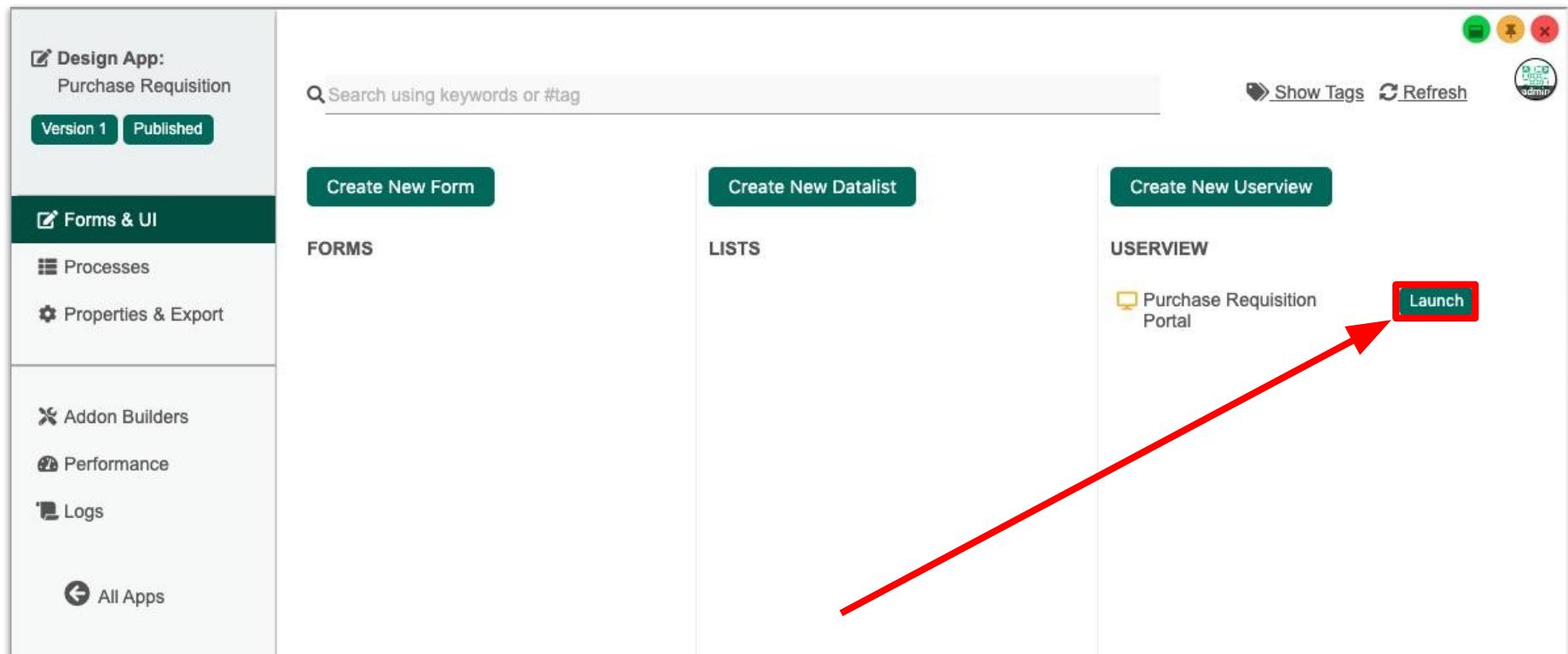
The screenshot illustrates the process of publishing an app version. On the left, the main application dashboard shows a sidebar with various options like Design App, Forms & UI, Processes, and Logs. A red box highlights the 'Version 1 | Not Published' status under the 'Design App' section. In the center, there's a search bar and a 'Create New Form' button. On the right, there are sections for 'FORMS' and 'LISTS'. A modal window titled 'APP CONFIGURATION MANAGEMENT' is open, showing a table for managing app versions. The table has columns for VERSION, PUBLISHED, NOTES, and DATE CREATED. A single row is present with the value '1' in the VERSION column. A red box surrounds this row. At the bottom of the modal, there are buttons for 'New Version', 'Publish', 'Unpublish', 'View', and 'Delete'. The 'Publish' button is also highlighted with a red box. The top right corner of the modal shows user information: 'DIDU DASH admin'.

VERSION	PUBLISHED	NOTES	DATE CREATED
1			06-12-2019 03:37 AM

New Version Publish Unpublish View Delete

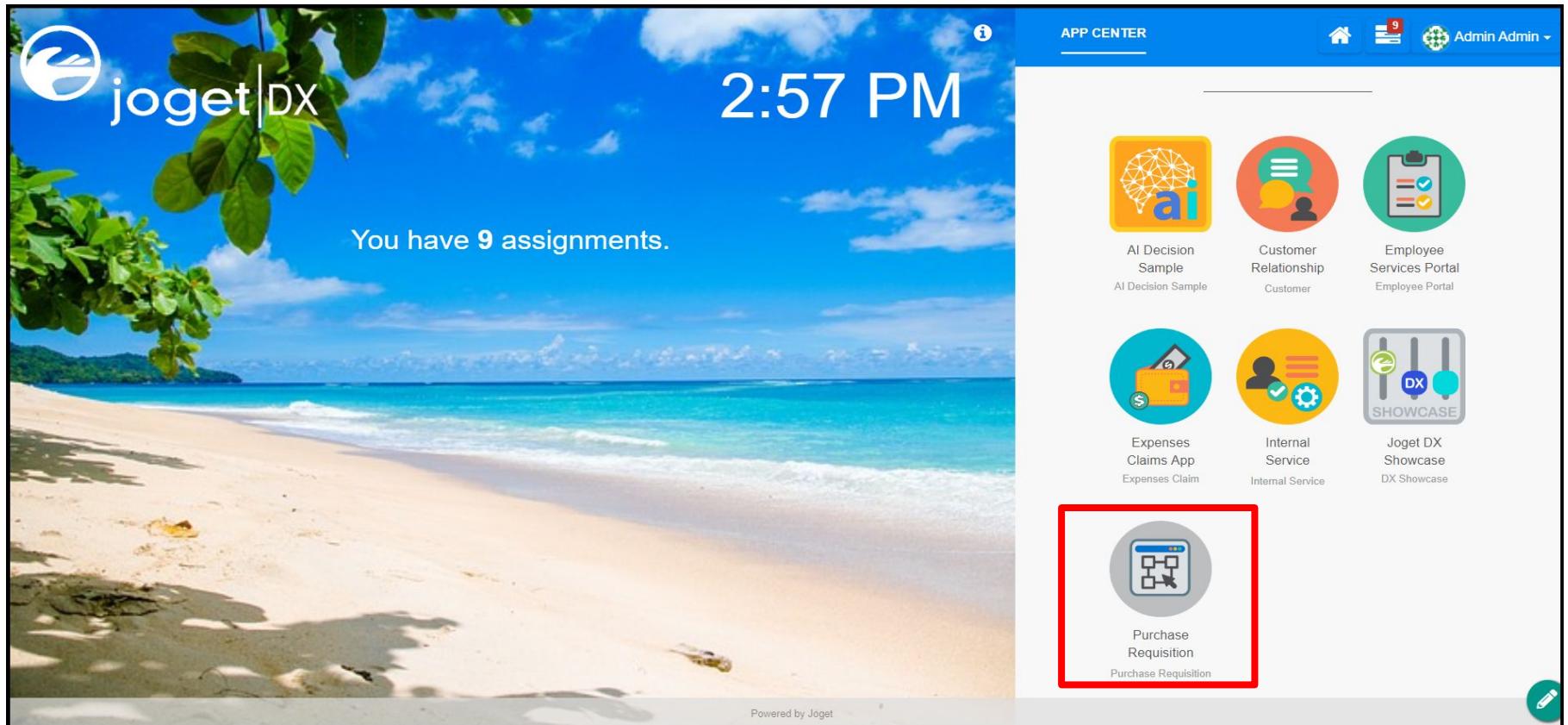
# Launching the Published App

- Click on **Launch** under Userview pane.



# Launching the Published App

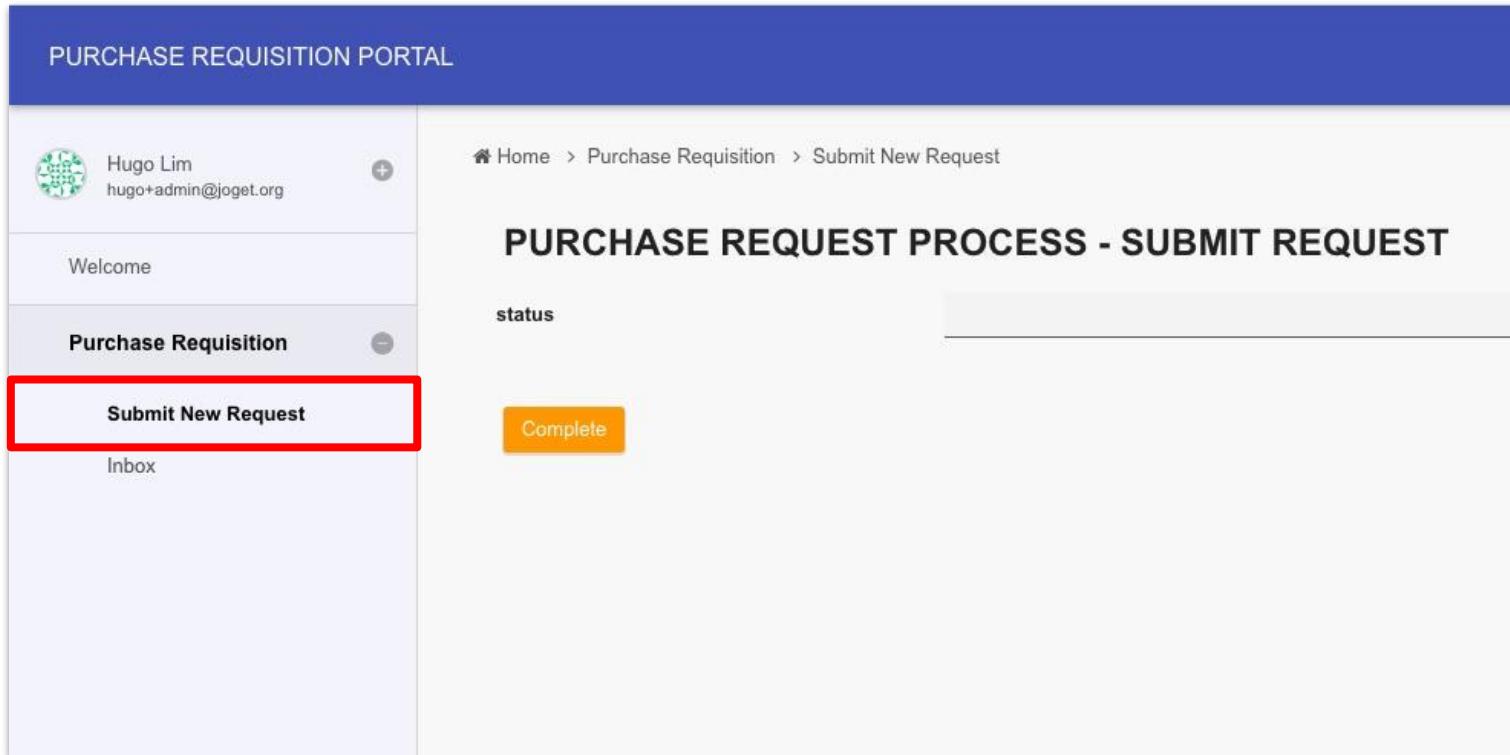
- One can also access it through the **App Center** itself.



# Run Through the Process

---

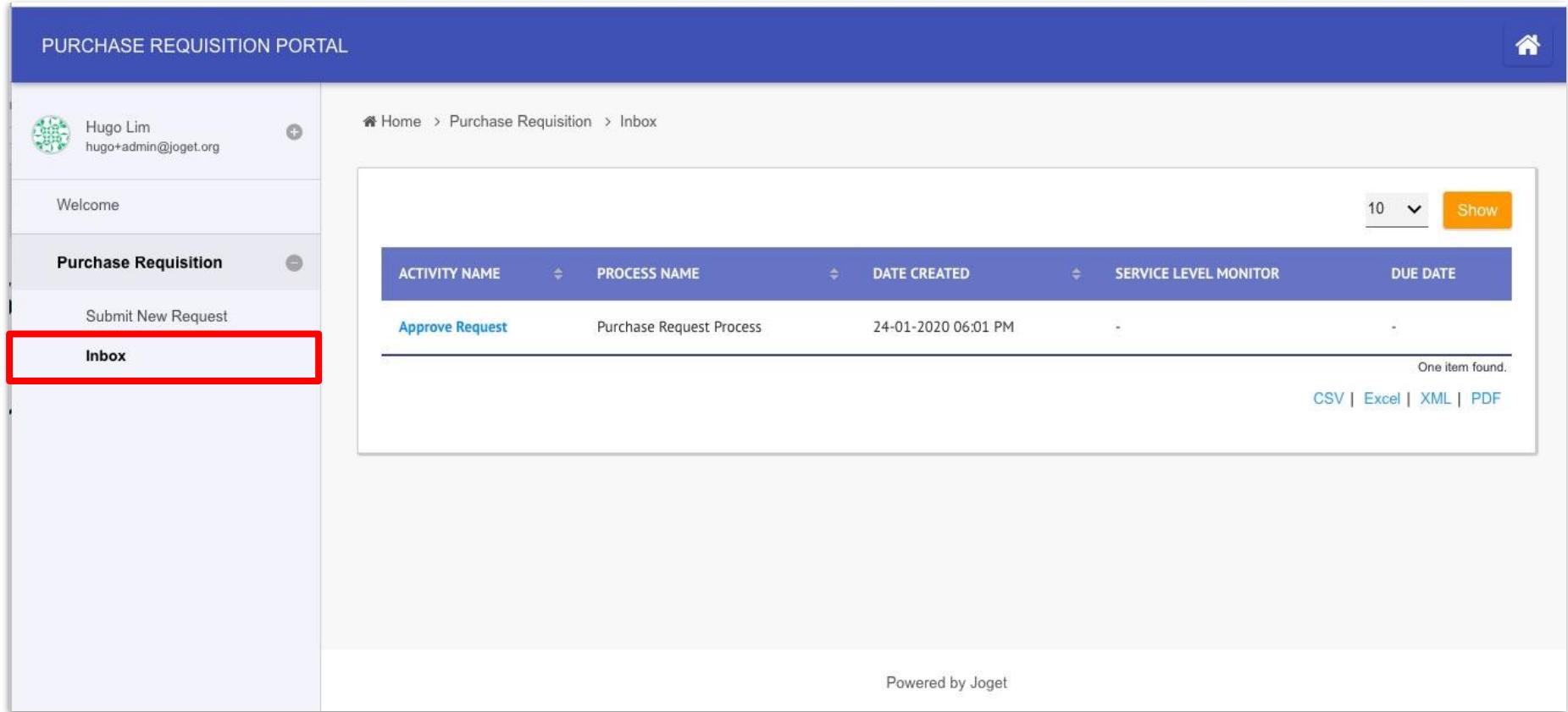
- With the Userview created, run through the entire **Purchase Request Process** again by using the userview.



The screenshot shows the "PURCHASE REQUISITION PORTAL" interface. On the left, there is a sidebar with a user profile for "Hugo Lim" (hugo+admin@joget.org) and a "Purchase Requisition" section containing a red box around the "Submit New Request" button. The main content area shows the "PURCHASE REQUEST PROCESS - SUBMIT REQUEST" page with a breadcrumb navigation: Home > Purchase Requisition > Submit New Request. Below the breadcrumb is a "status" field with a horizontal line. At the bottom of the page is an orange "Complete" button.

# Run through the Process

- Access subsequent assignment through the Inbox menu.



The screenshot shows the Purchase Requisition Portal interface. On the left, there is a sidebar with a user profile for Hugo Lim (hugo.admin@joget.org) and a navigation menu with options like Welcome, Purchase Requisition (which is expanded), Submit New Request, and Inbox. The 'Inbox' option is highlighted with a red rectangle. The main content area shows a breadcrumb path: Home > Purchase Requisition > Inbox. Below this is a table with one item:

ACTIVITY NAME	PROCESS NAME	DATE CREATED	SERVICE LEVEL MONITOR	DUUE DATE
Approve Request	Purchase Request Process	24-01-2020 06:01 PM	-	-

At the bottom right of the main content area, there are links for CSV, Excel, XML, and PDF. The footer of the page says 'Powered by Joget'.

# Good To Know

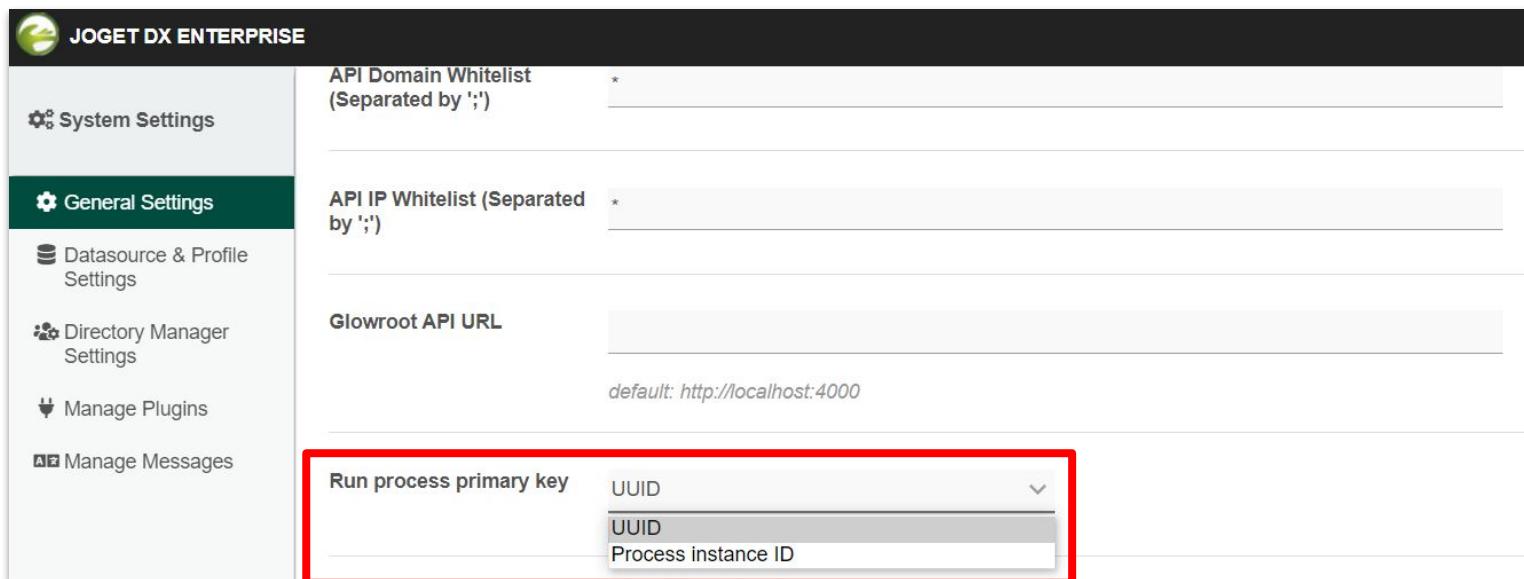
---

- At this point of time, the App is published, therefore end user would be able to access it.

# Important Note!

---

- In DX, by **default**, Run Process will generate **UUID** as the Primary Key instead of **Process Instance ID**.
- You can **change** it back at Admin Bar > Settings > General Settings > System Administration Settings > Run Process primary key.



# Discussion

---

- Who is Applicant? Who is Approver?  
Have we actually defined who they really are?

# Chapter 3 Review

---

- Evaluate and verify the process by using:-
  - Run Process button available to only the App Designer/Admin.
  - Userview which is accessible by end users.



# Chapter 4

## Participant Mapping

# Participant Mapping

---

- User Mapping Categories.
  - Map to User or Group
  - Map to Org Chart
  - Map to Workflow Variable
  - Map to Plugin

# Plugin Types



## App Builder Components & Supported Plugin Types

### Userview Builder

- Userview Menu
- Userview Permission
- Userview Theme

### Datalist Builder

- Datalist Binder
- Datalist Action
- Datalist Column Formatter
- Datalist Filter Type

### Process Builder

- Deadline
- **Process Participant**
- Process Tool
- Process Form Modifier
- Decision Tool

### Form Builder

- Form Element
- Form Load Binder
- Form Store Binder
- Form Options Binder
- Form Validator
- Form Permission
- Process Form Modifier
- Post Form Submission Processing
- Generator

### Addon Builder

### App Platform

- Audit Trail
- Hash Variable
- Web Service

### Platform

- Directory Manager
- Web Service

# Where to map participants?

JOGET DX ENTERPRISE

Design App: Purchase Requisition  
Version 1 Not Published

Forms & UI

**Processes**

Properties & Export

Addon Builders

Performance

Logs

All Apps

Design Process Configure Mapping Update via Saved XPDL Run Process

Process name: Purchase Request Process

Show Additional Info

Map Participants to Users Map Activities to Forms Map Tools to Plugins Map Routes to Plugins Variable List

Map Participants to Users

Participant	ID	Action
Applicant	applicant	Add/Edit Mapping
Approver	approver	Add/Edit Mapping
System	system	Add/Edit Mapping
Process Start White List		Add/Edit Mapping
Type	Role	
Value	Admin	

Remove Mapping

ALL participants must be mapped, except “System”

# Participant Mapping

Let's assume that, applicant's HOD is responsible to approve any purchase requests made by the applicant.

Map Participants to Users    Map Activities to Forms    Map Tools to Plugins    Map Routes to Plugins    Variable List

This is the list of participants defined in the Workflow design.

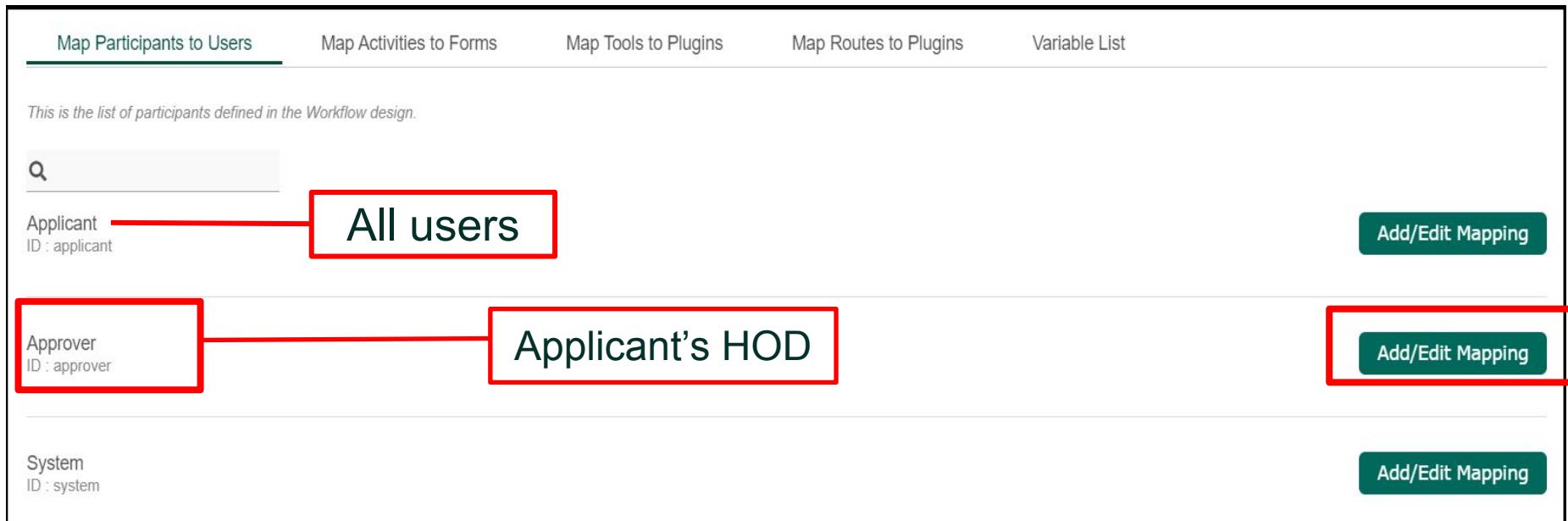
All users Add/Edit Mapping

Add/Edit Mapping Add/Edit Mapping

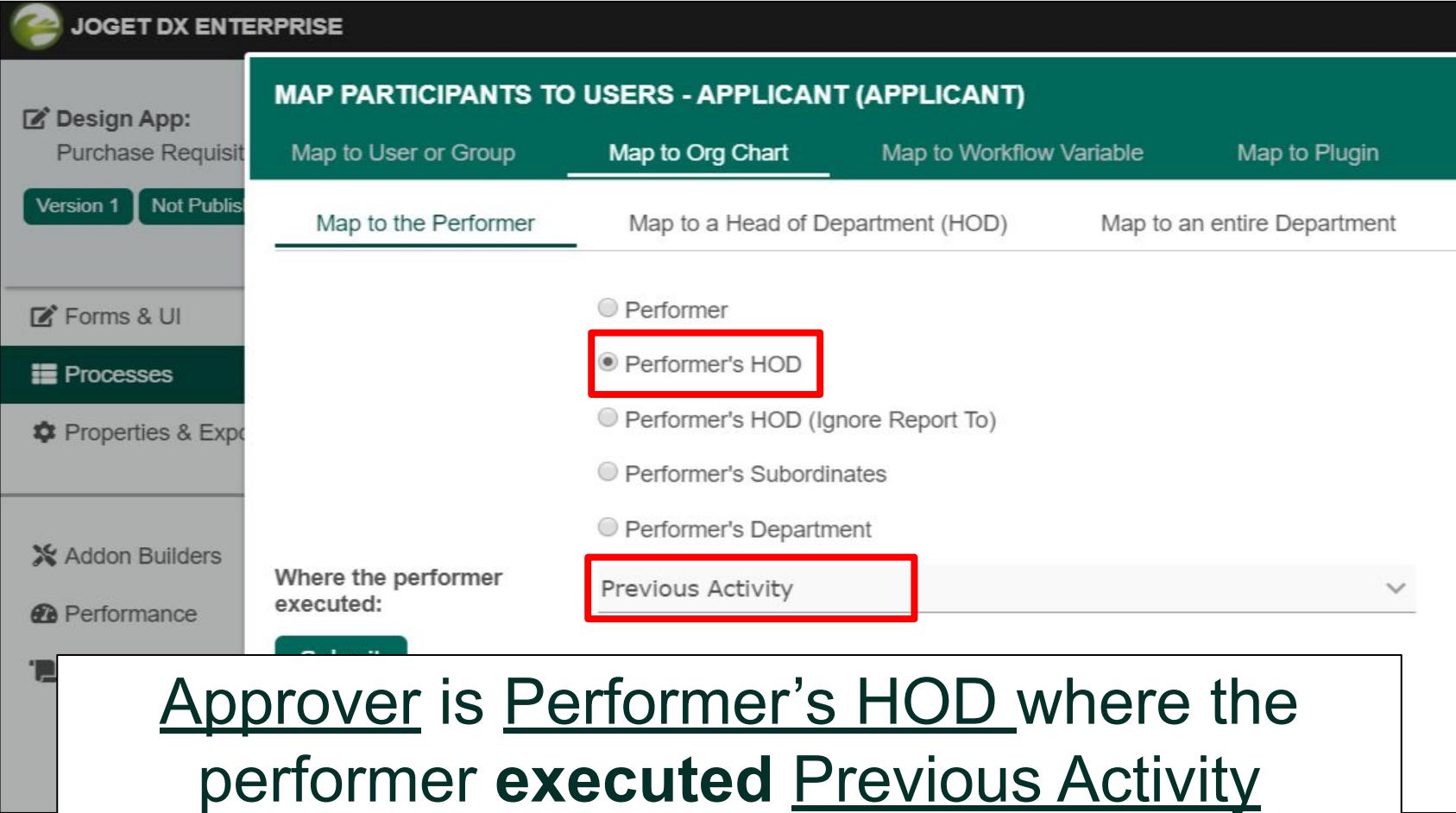
Applicant  
ID : applicant

Approver  
ID : approver

System  
ID : system



# Approver: Performer's HOD



JOGET DX ENTERPRISE

Design App: Purchase Requisit  
Version 1 Not Public

Forms & UI Processes Properties & Export Addon Builders Performance

MAP PARTICIPANTS TO USERS - APPLICANT (APPLICANT)

Map to User or Group **Map to Org Chart** Map to Workflow Variable Map to Plugin

Map to the Performer **Map to a Head of Department (HOD)** Map to an entire Department

Performer  Performer's HOD  Performer's HOD (Ignore Report To)  Performer's Subordinates  Performer's Department

Where the performer executed: **Previous Activity**

Approver is Performer's HOD where the performer executed Previous Activity

# Process Participant Mapping

Map Participants to Users    Map Activities to Forms    Map Tools to Plugins    Map Routes to Plugins    Variable List

*This is the list of participants defined in the Workflow design.*

Participant Details		Action
Applicant ID : applicant	Type : Performer Value : Activity Definition ID : <a href="#">runProcess</a>	<a href="#">Add/Edit Mapping</a>
Approver ID : approver	Type : Performer's HOD Value : Previous Activity	<a href="#">Add/Edit Mapping</a>

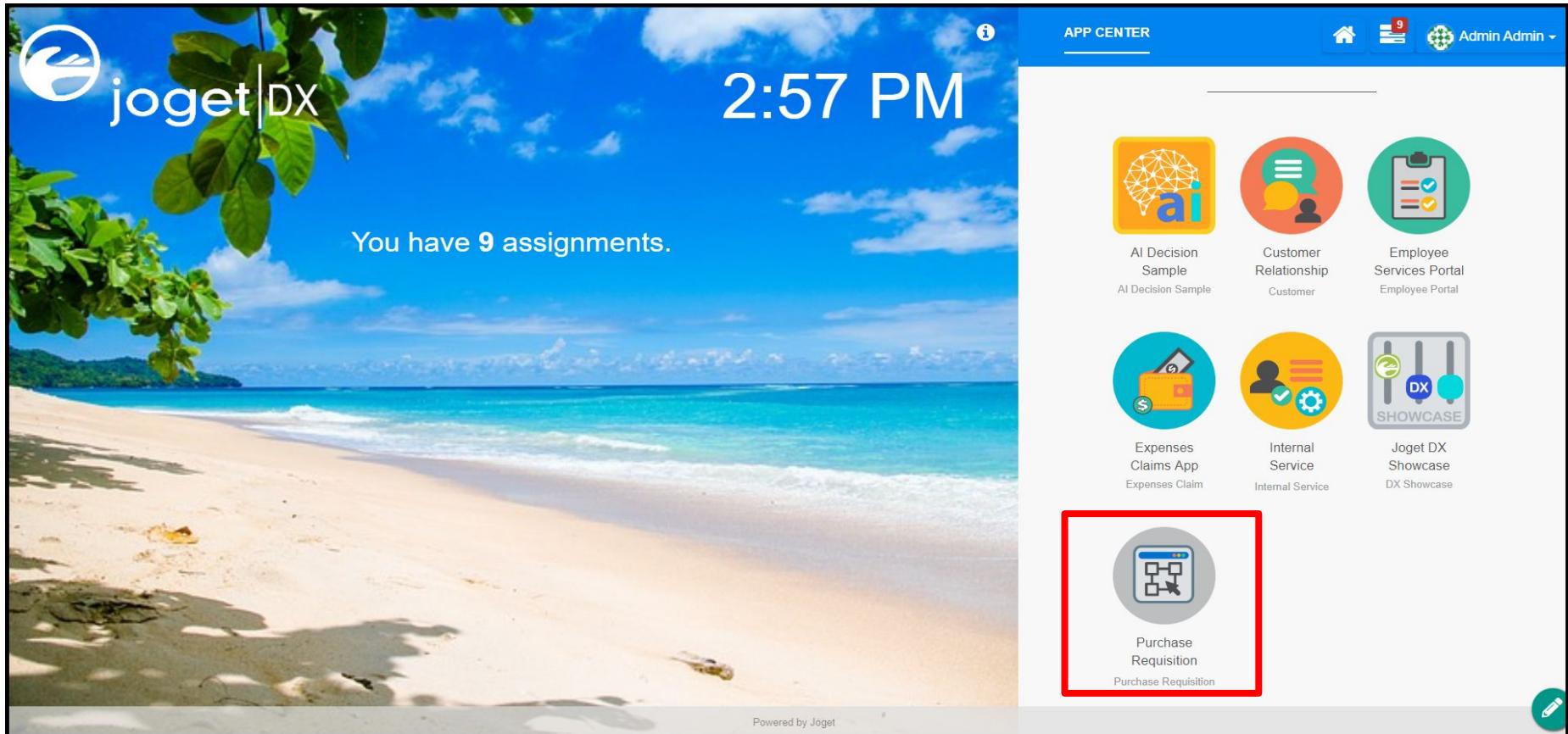
# Test Run the Process

---

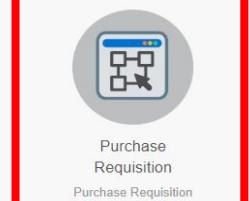
- With the participant mapping set, run the process again as the user “cat”, through the **Purchase Requisition Portal**.

# Login As cat/password

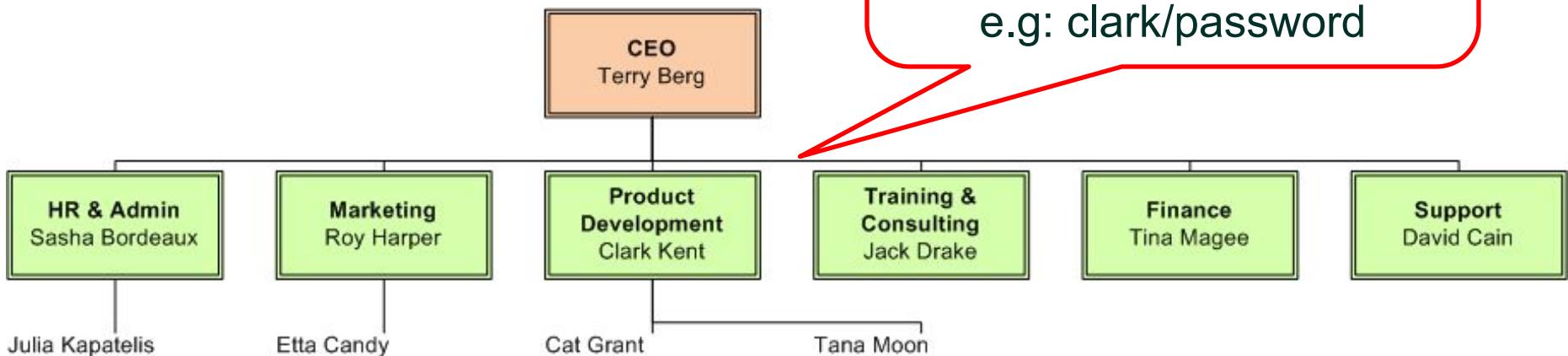
- Open up App Center to access Purchase Requisition portal.



The image displays two screenshots of a digital dashboard or application center. The left screenshot shows a scenic beach background with the Joget DX logo and a message indicating 9 assignments. The right screenshot shows the App Center interface with various app icons and names. A red box highlights the 'Purchase Requisition' app icon.

App Icon	App Name	Description
	AI Decision Sample	AI Decision Sample
	Customer Relationship	Customer
	Employee Services Portal	Employee Portal
	Expenses Claims App	Expenses Claim
	Internal Service	Internal Service
	Joget DX Showcase	DX Showcase
	Purchase Requisition	Purchase Requisition

# Who Is Cat Grant's HOD?



Note: This is the sample organization chart provided upon initial installation, for your testing convenience. The 3 users: *cat*, *clark*, and *admin*, is free by default.

# Important Notes

---

- All participants MUST be mapped.
- Do NOT choose activity from the same swimlane as the participant that is being mapped unless it is intended so.
- Choose only activity that has been completed.
- If NO user is found, it will be defaulted to the last performer. (Possibly, the current logged in user who had just completed the prior activity OR the process requester)
- If mapped to an inactive user, do delegate, re-assign, or change mapping and re-evaluate the assignment.

# Chapter 4 Review

---

We have learnt to:

1. Understand various Participant Mapping options available.
2. Understand the important rules in Participant Mapping.
3. Able to relate and manage participant mapping with the administrative process monitoring module.

---

# Chapter 5

## Process Monitoring

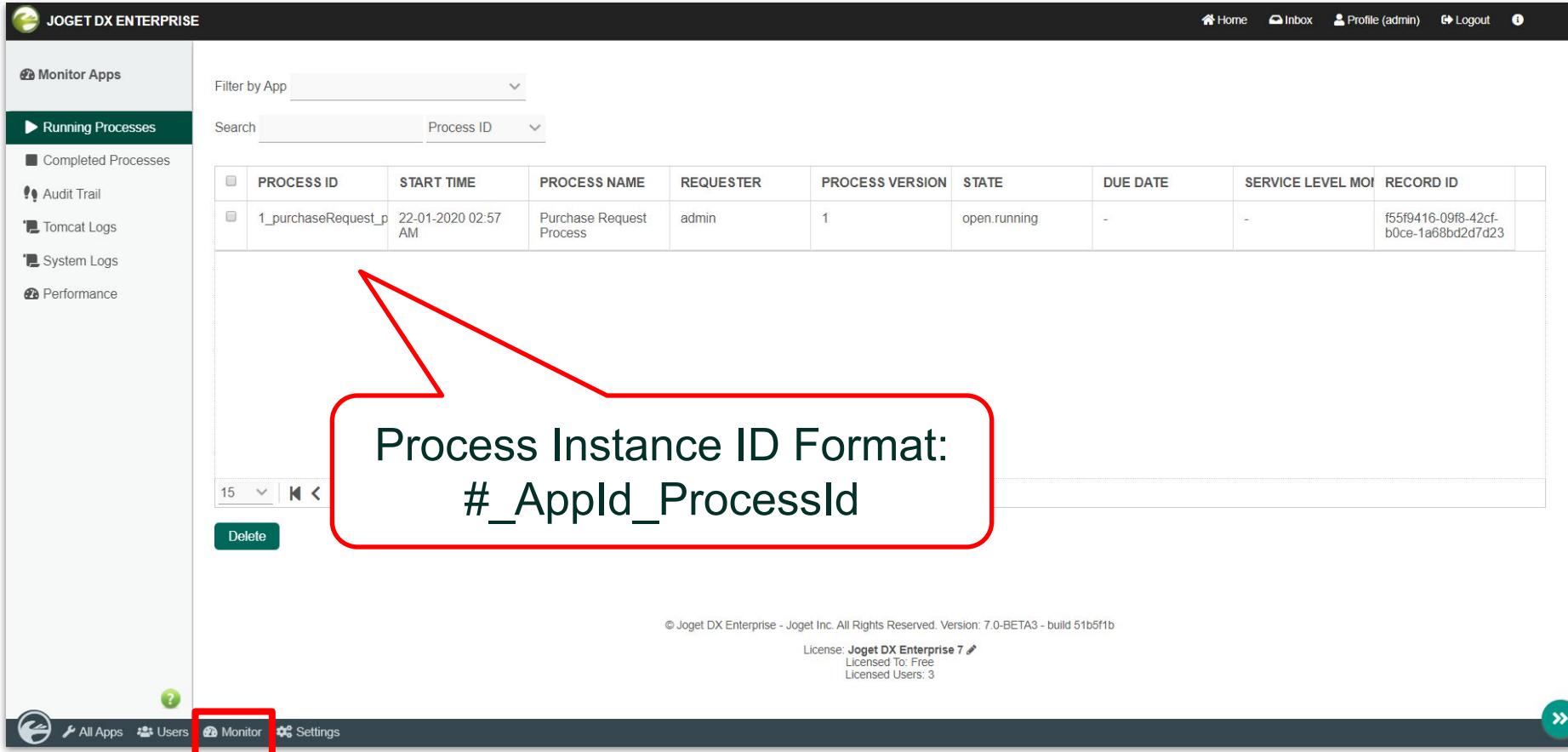
# Monitoring Processes

---

- Allows you to monitor the status of processes, both **running** and **completed**.
- Allows you to view an **audit trail** of process actions.
- Captures process data to allow for the generation of reports.

# Monitor Running Processes

- Click on Monitor from the Admin Bar to access.



The screenshot shows the Joget DX Enterprise monitor interface. On the left, a sidebar menu includes options like Monitor Apps, Running Processes (which is selected and highlighted in green), Completed Processes, Audit Trail, Tomcat Logs, System Logs, and Performance. The main content area displays a table of running processes. A red callout box highlights the process instance ID '1\_purchaseRequest\_p' in the table, which is annotated with the text 'Process Instance ID Format: #\_AppId\_ProcessId'. The table columns are: PROCESS ID, START TIME, PROCESS NAME, REQUESTER, PROCESS VERSION, STATE, DUE DATE, SERVICE LEVEL MOI, and RECORD ID. The process listed has a start time of 22-01-2020 02:57 AM, is named 'Purchase Request Process', requested by 'admin', version 1, and is in an 'open.running' state. The record ID is f55f9416-09f8-42cf-b0ce-1a68bd2d7d23. At the bottom of the page, there is copyright information and a license section. The footer features a navigation bar with icons for All Apps, Users, Monitor (which is highlighted with a red box), and Settings.

JOGET DX ENTERPRISE

Monitor Apps

Running Processes

Completed Processes

Audit Trail

Tomcat Logs

System Logs

Performance

Filter by App

Search Process ID

PROCESS ID	START TIME	PROCESS NAME	REQUESTER	PROCESS VERSION	STATE	DUE DATE	SERVICE LEVEL MOI	RECORD ID
1_purchaseRequest_p	22-01-2020 02:57 AM	Purchase Request Process	admin	1	open.running	-	-	f55f9416-09f8-42cf-b0ce-1a68bd2d7d23

Process Instance ID Format:  
#\_AppId\_ProcessId

15 ⏪ ⏩

Delete

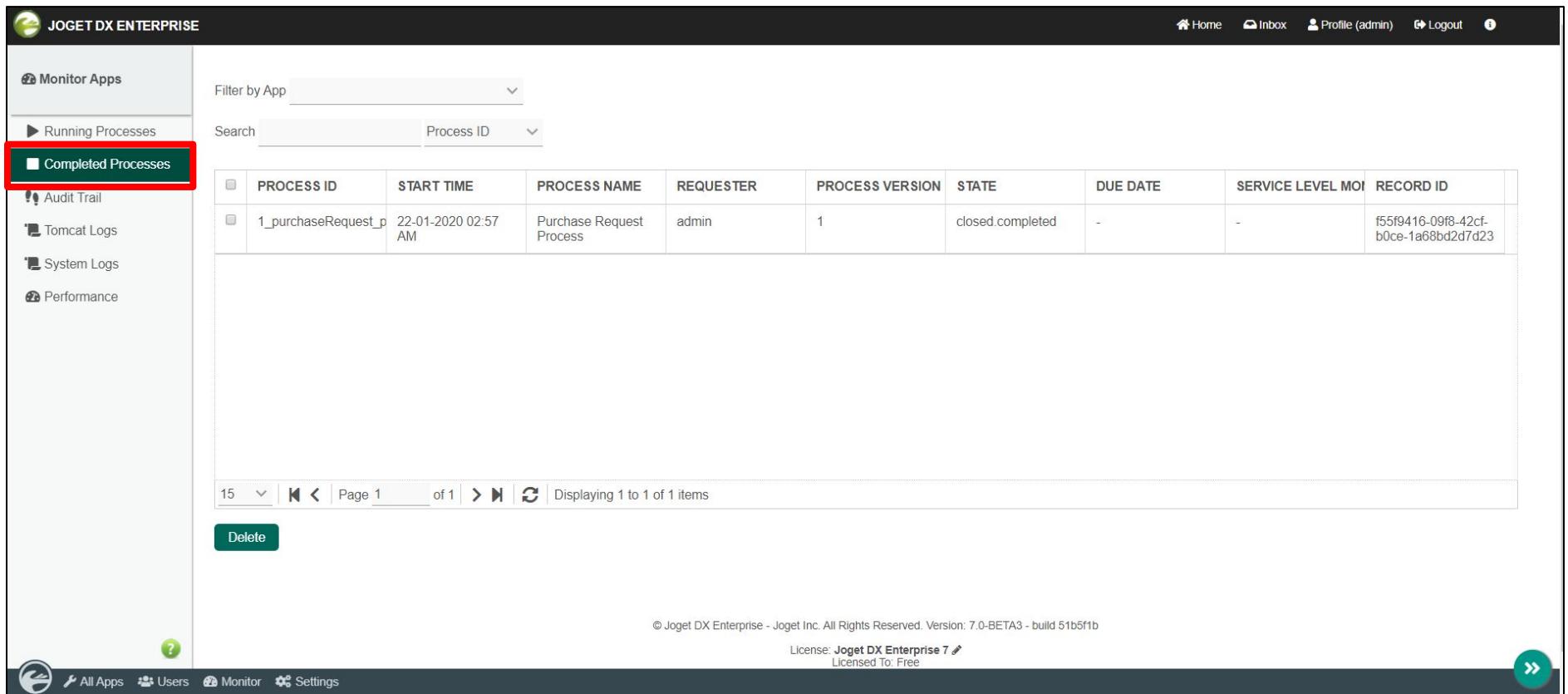
© Joget DX Enterprise - Joget Inc. All Rights Reserved. Version: 7.0-BETA3 - build 51b5f1b

License: Joget DX Enterprise 7.0  
Licensed To: Free  
Licensed Users: 3

All Apps Users Monitor Settings

# Monitor Completed Processes

- Click on “Completed Processes” to see completed process instances.



The screenshot shows the Joget DX Enterprise web application. On the left, there is a sidebar with the following menu items:

- JOGET DX ENTERPRISE
- Monitor Apps
  - Running Processes
  - Completed Processes** (highlighted with a red box)
  - Audit Trail
  - Tomcat Logs
  - System Logs
  - Performance

On the right, the main content area displays a table of completed processes. The table has the following columns:

PROCESS ID	START TIME	PROCESS NAME	REQUESTER	PROCESS VERSION	STATE	DUUE DATE	SERVICE LEVEL MOI	RECORD ID
1_purchaseRequest_p	22-01-2020 02:57 AM	Purchase Request Process	admin	1	closed.completed	-	-	f55f9416-09f8-42cf-b0ce-1a68bd2d7d23

At the bottom of the page, there is a footer with the following information:

© Joget DX Enterprise - Joget Inc. All Rights Reserved. Version: 7.0-BETA3 - build 51b5f1b  
License: Joget DX Enterprise 7  
Licensed To: Free

Navigation icons include a magnifying glass, a question mark, and arrows for navigation.

# View the Details of a Running Process Instance

---

- From the list of running processes, **click into any instance**.
- **Details on the current state** of the selected process instance will be shown.

# View the Details of a Running Process Instance

**JOGET DX ENTERPRISE**

**Monitor Apps**    **View Graph**    **Abort Instance**    **Remove Instance**    **Re-evaluate**

**Running Processes**

- Completed Processes
- Audit Trail
- Tomcat Logs
- System Logs
- Performance

App	Purchase Requisition v1
Process name	Purchase Request Process
Process ID	172_purchaseRequisition_purchaseRequestProcess
Record ID	e0b8a0cf-c5fd-4904-a479-4552feb0fd30
Process Definition ID	purchaseRequisition#2#purchaseRequestProcess
Process Version	2
State	open.running
Service Level Monitor	-
Requester	admin
Start Time	10-12-2019 01:23 AM

Activity Instance ID format:  
#\_ProcessInstanceId\_ActivityId

Which activities are completed?  
Which activity is active now?

ACTIVITY				
ACTIVITY ID	ACTIVITY NAME	STATE	CREATE TIME	SERVICE LEVEL MOI
913_172_purchaseReq	Approve / Reject Request	open.not_running.not_	10-12-2019 01:23 AM	-
912_172_purchaseReq	Submit Request	closed.completed	10-12-2019 01:23 AM	-

# View the Details of a Running Process Instance

- These are actions that can be taken on a **process instance**.
  - **View Graph**
    - Displays the process diagram, where the current activities are highlighted in yellow.
  - **Abort Instance**
    - Terminates the process instance, leaving the process data intact
  - **Remove Instance**
    - Permanently delete the process instance, process data will be removed as well
  - **Re-evaluate**
    - In cases where participant mappings are changed before the process is completed, this action will allow pending assignments to be reassigned to the newly mapped participants.



# View the Details of an Activity

JOGET DX ENTERPRISE

Monitor Apps

Running Processes

Completed Processes

Audit Trail

Tomcat Logs

System Logs

Performance

Re-evaluate

Re-evaluate Assignment(s) For User

Reassign User

Complete

Process Instance: 172\_purchaseRequisition\_purchaseRequestProcess

Activity ID: 913\_172\_purchaseRequisition\_purchaseRequestProcess\_approveRejectRequest

Activity Definition ID: approveRejectRequest

Activity Name: Approve / Reject Request

State: open.not\_running.not\_started

Service Level Monitor: -

List of Pending: admin

Create time:

Date limit:

Due date: 913\_172\_purchaseRequisition\_purchaseRequestProcess\_approveRejectRequest

Delay:

Finish time: From Date Created

ACTIVITY LIST

ACTIVITY ID	ACTIVITY NAME	STATE	CREATE TIME	SERVICE LEVEL MOI
913_172_purchaseRequisition_purchaseRequestProcess_approveRejectRequest	Approve / Reject Request	open.not_running.not_started	10-12-2019 01:23 AM	-
912_172_purchaseRequisition_purchaseRequestProcess_submitRequest	Submit Request	closed.completed	10-12-2019 01:23 AM	-

Workflow variable value

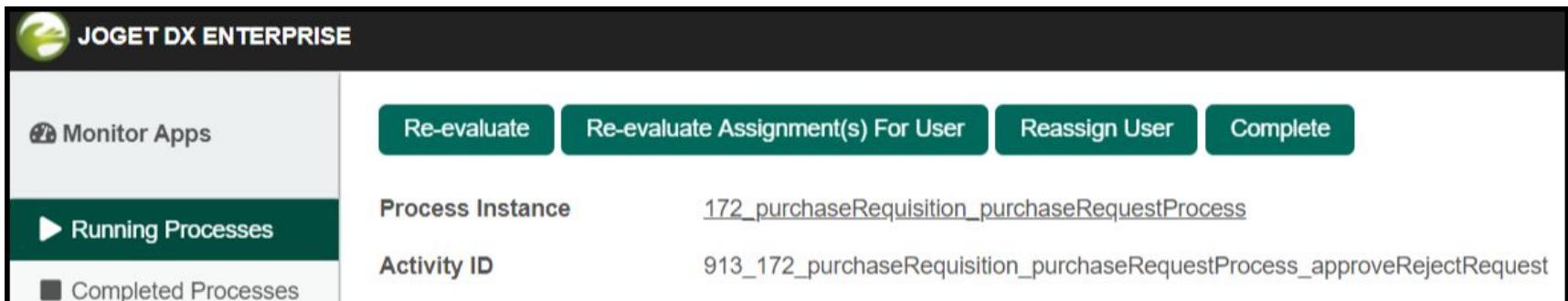
VARIABLE LIST

status: Approved

Set

# View the Details of a Running Activity Instance

- These are the actions that can be taken on an **activity instance**.
  - **Re-evaluate**
    - Re-evaluate the participant mapping for the current activity instance.
  - **Re-evaluate assignment(s) for user**
    - Re-evaluate all assignments of a particular user.
  - **Reassign user**
    - Reassign one of the assignee(s) of this activity instance to another user.
  - **Complete**
    - Complete the activity instance as the current logged in user.



The screenshot shows the Joget DX Enterprise web interface. At the top, there's a navigation bar with the Joget logo and links for 'JOGET DX ENTERPRISE', 'Monitor Apps', 'Running Processes' (which is highlighted in green), and 'Completed Processes'. Below the navigation, there's a toolbar with four buttons: 'Re-evaluate', 'Re-evaluate Assignment(s) For User' (which is also highlighted in green), 'Reassign User', and 'Complete'. To the right of the toolbar, there are two sections: 'Process Instance' with the value '172\_purchaseRequisition\_purchaseRequestProcess' and 'Activity ID' with the value '913\_172\_purchaseRequisition\_purchaseRequestProcess\_approveRejectRequest'.

# Discussion

---

- What if a person who is assigned to an assignment has left the organization? What can we do?

# Chapter 4 Review

---

We have learned to:

1. Monitor running and completed processes.
2. Drill down into activity details of each process that is running or completed.
3. Debug processes by examining workflow variable value.
4. Appreciate the “Re-evaluate” feature in process monitoring.

# Module Review

---

1. Business Process Design (BPE)
2. Designing your first Process with Process Builder
3. Running your first Automated Process
4. Participant Mapping
5. Process Monitoring

# Recommended Further Learning

---

- Designing Forms for the activities created in your Process Flow.
- Configure Process Tool plugin for the tool created in your Process Flow.

# Stay Connected with Joget

---

- [www.joget.org](http://www.joget.org)
- [community.joget.org](http://community.joget.org)
- [twitter.com/jogetworkflow](http://twitter.com/jogetworkflow)
- [facebook.com/jogetworkflow](http://facebook.com/jogetworkflow)
- [youtube.com/jogetworkflow](http://youtube.com/jogetworkflow)