



**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 requestElse 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 requestElse 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

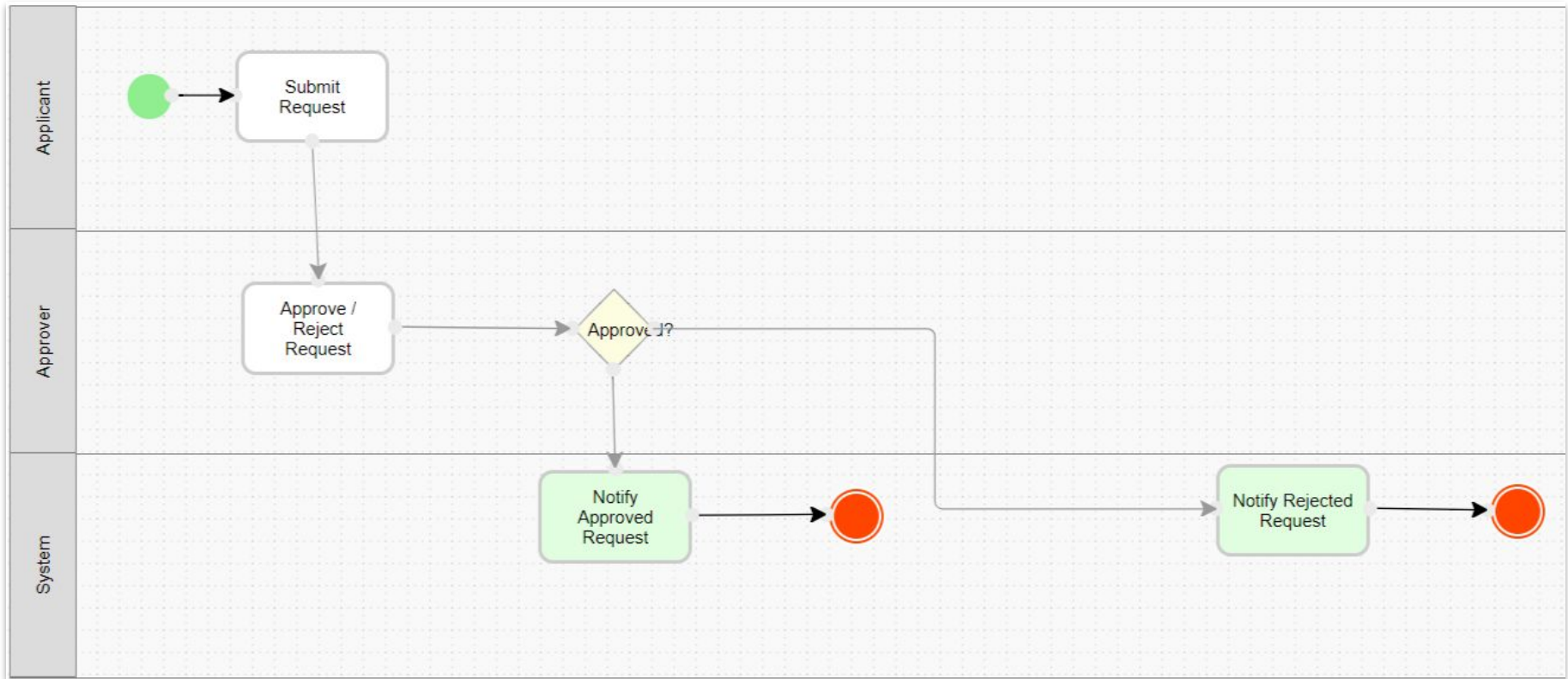
improved

# 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

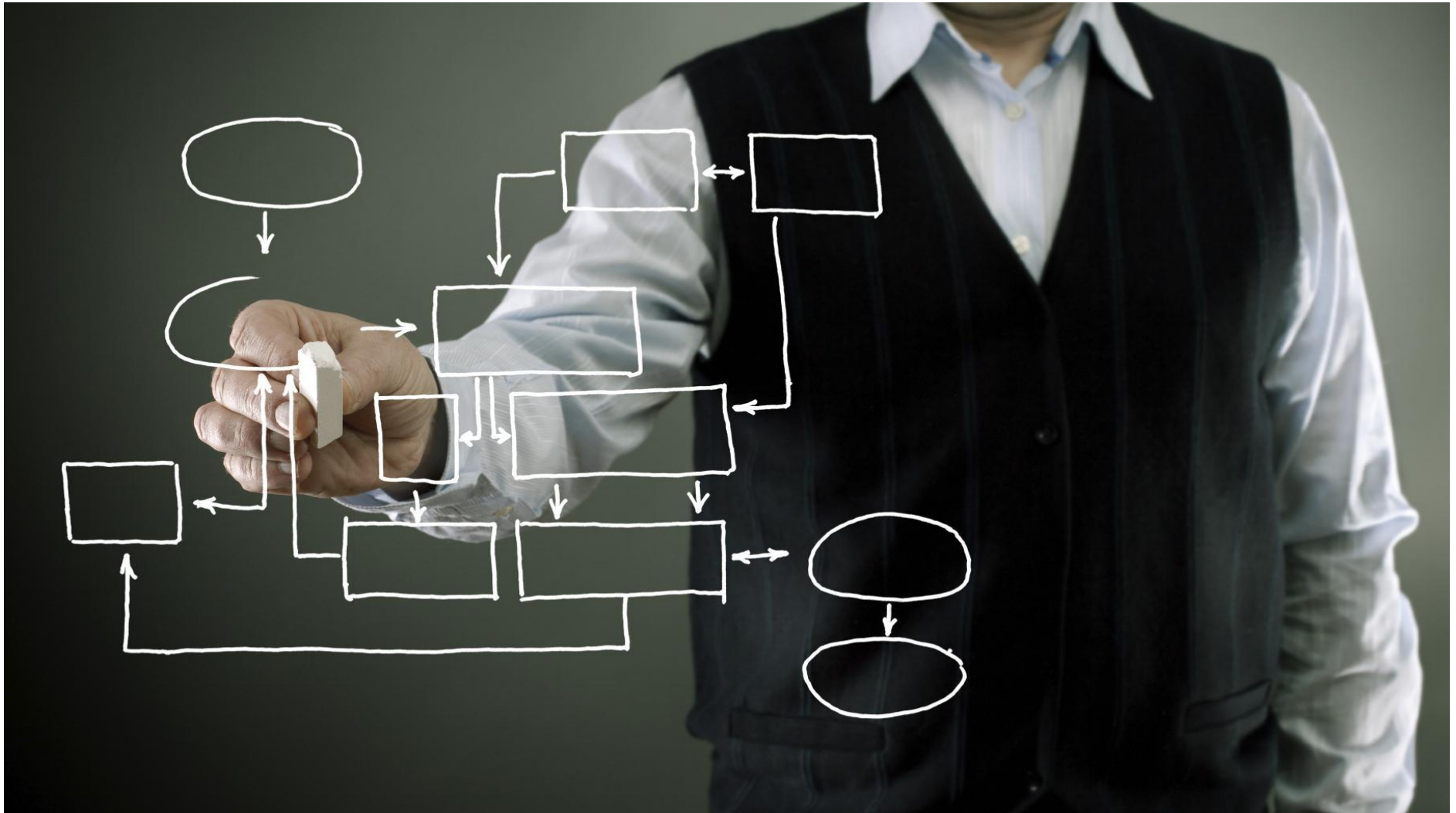


# Business Process Engineering

---

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.

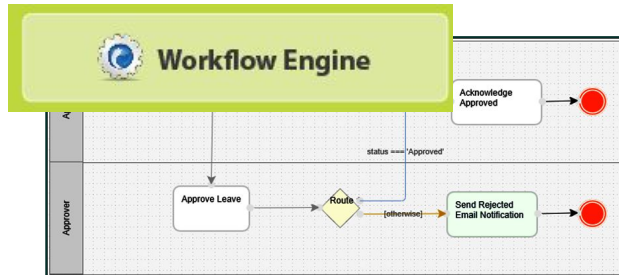
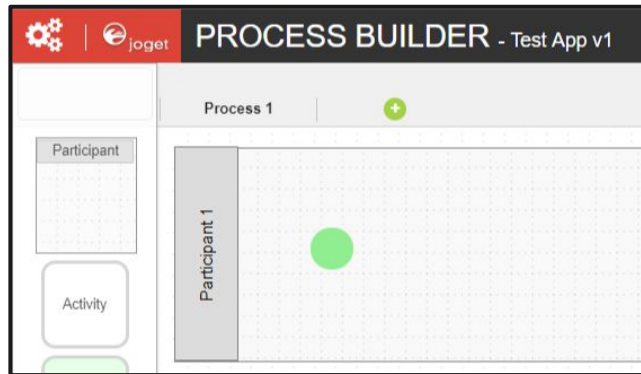
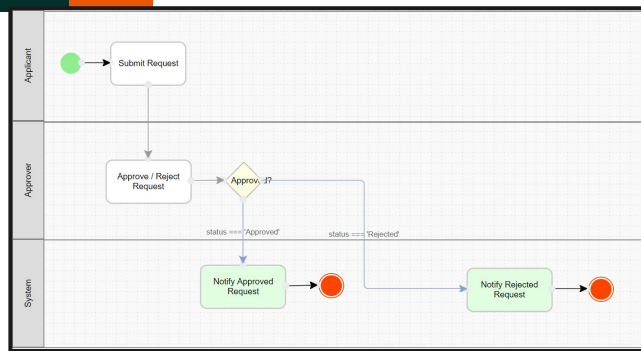




# Chapter 2

## Designing your first Process with Process Builder

# Business Process Automation



Business Process Design



Process Design



Process



### Apps

Progressive Web Apps  
Various Form Factors



Customer Relationship



Employee Services Portal



Expenses Claims App



Internal Service Request App



### Development

Simple, Visual,  
Drag and Drop.

**Form Builder**

**List Builder**

**Userview Builder**

**Process Builder**

**User Management**

- Organizations
- Groups
- Users

**Monitoring**

- Processes
- Application Performance
- Logging & Audit

**System**

- Workflow Engine
- Security and Directory
- Plugin Management

**Integration API**

- API JSON
- API Java
- API JavaScript

**Plugin Architecture**

**Add-on Builders**



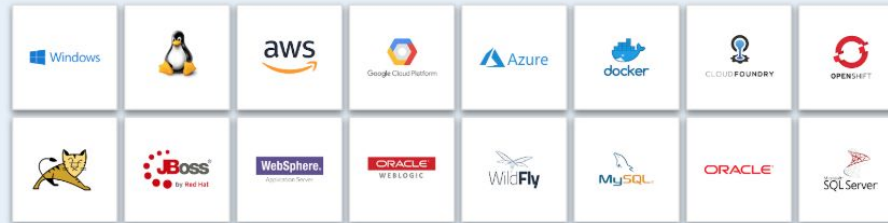
### Platform

Open, Flexible and  
Extensible.



### Infrastructure

On-Premise, Containers,  
Private / Public / Hybrid  
Cloud. App Server,  
DB, OS Independent.



Icons by [icons8.com](https://icons8.com)



### Apps

Progressive Web Apps  
Various Form Factors



Customer Relationship



Employee Services Portal



Expenses Claims App



Internal Service Request App



### Development

Simple, Visual,  
Drag and Drop.

**Joget DX**

Form Builder

List Builder

Userview Builder

Process Builder

Add-on Builders

**User Management**

- Organizations
- Groups
- Users

**Monitoring**

- Processes
- Application Performance
- Logging & Audit

**System**

- Workflow Engine
- Security and Directory
- Plugin Management

Plugin Architecture

Integration API

- API JSON
- API Java
- API JavaScript



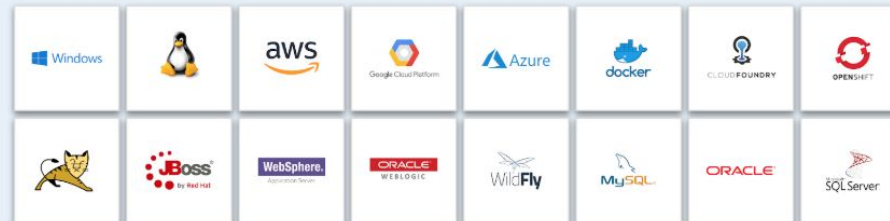
### Platform

Open, Flexible and  
Extensible.



### Infrastructure

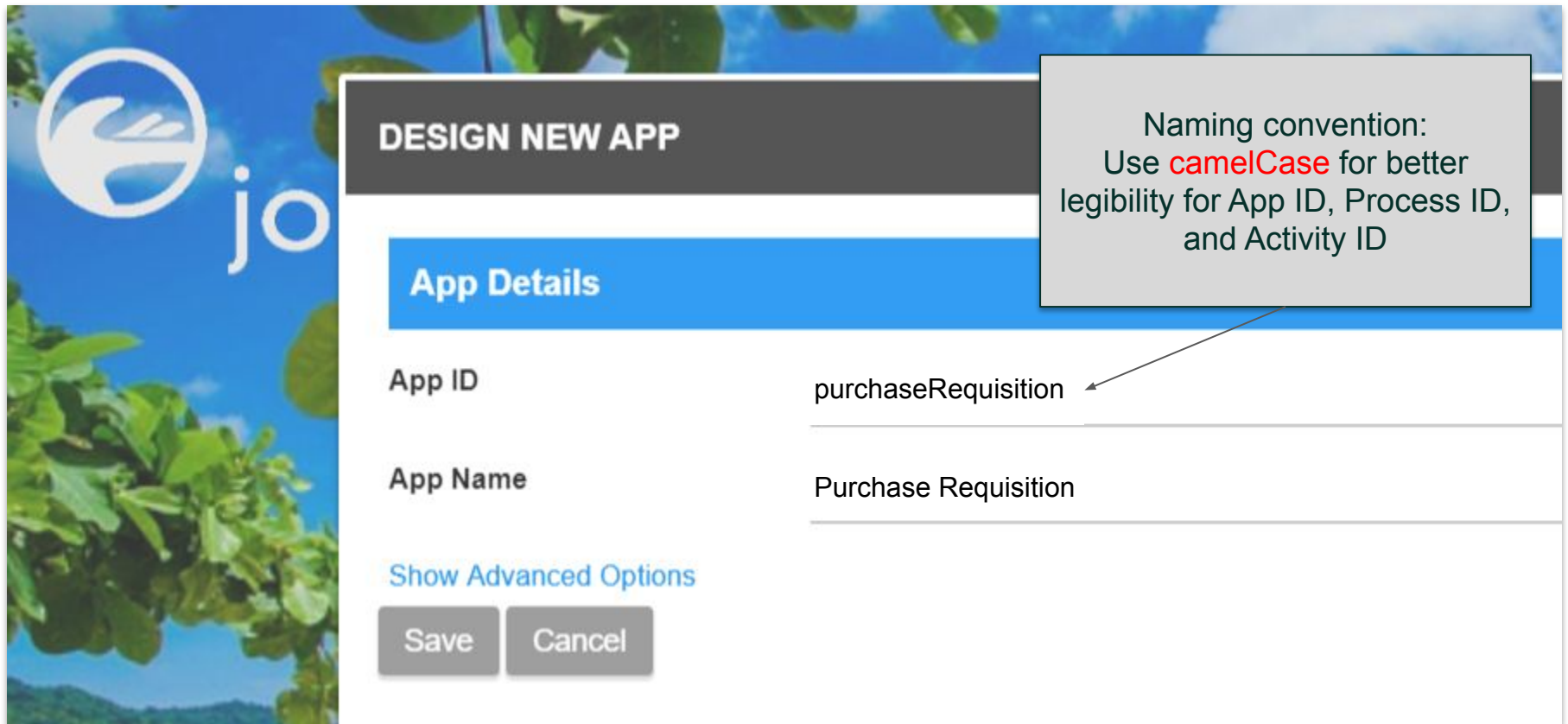
On-Premise, Containers,  
Private / Public / Hybrid  
Cloud. App Server,  
DB, OS Independent.



Icons by [icons8.com](https://icons8.com)

# Create Your First App

- Design Apps -> Design New App



**DESIGN NEW APP**

**App Details**

App ID: purchaseRequisition

App Name: Purchase Requisition

[Show Advanced Options](#)

Naming convention:  
Use **camelCase** for better legibility for App ID, Process ID, and Activity ID

# 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

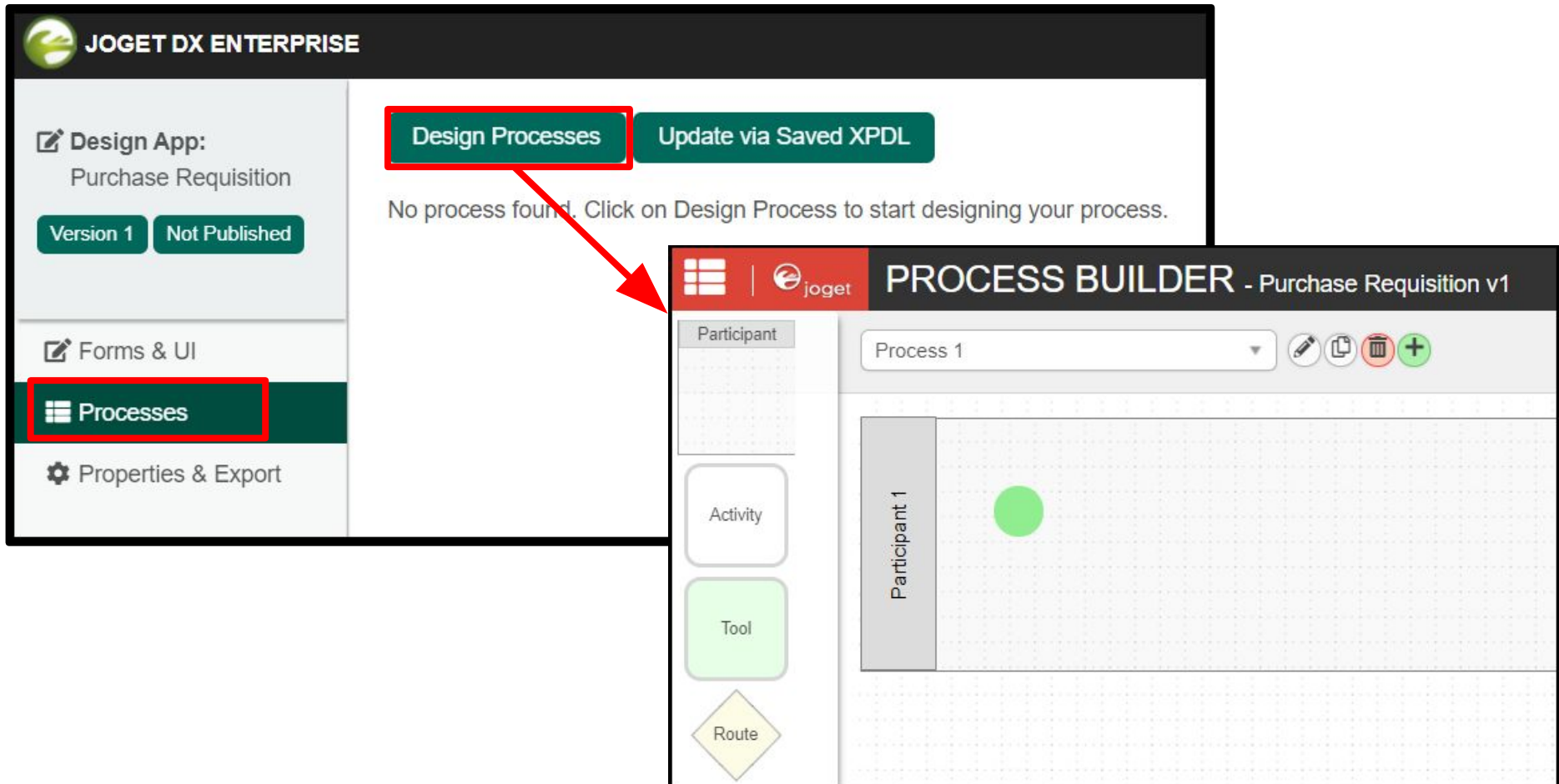
---

- 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**

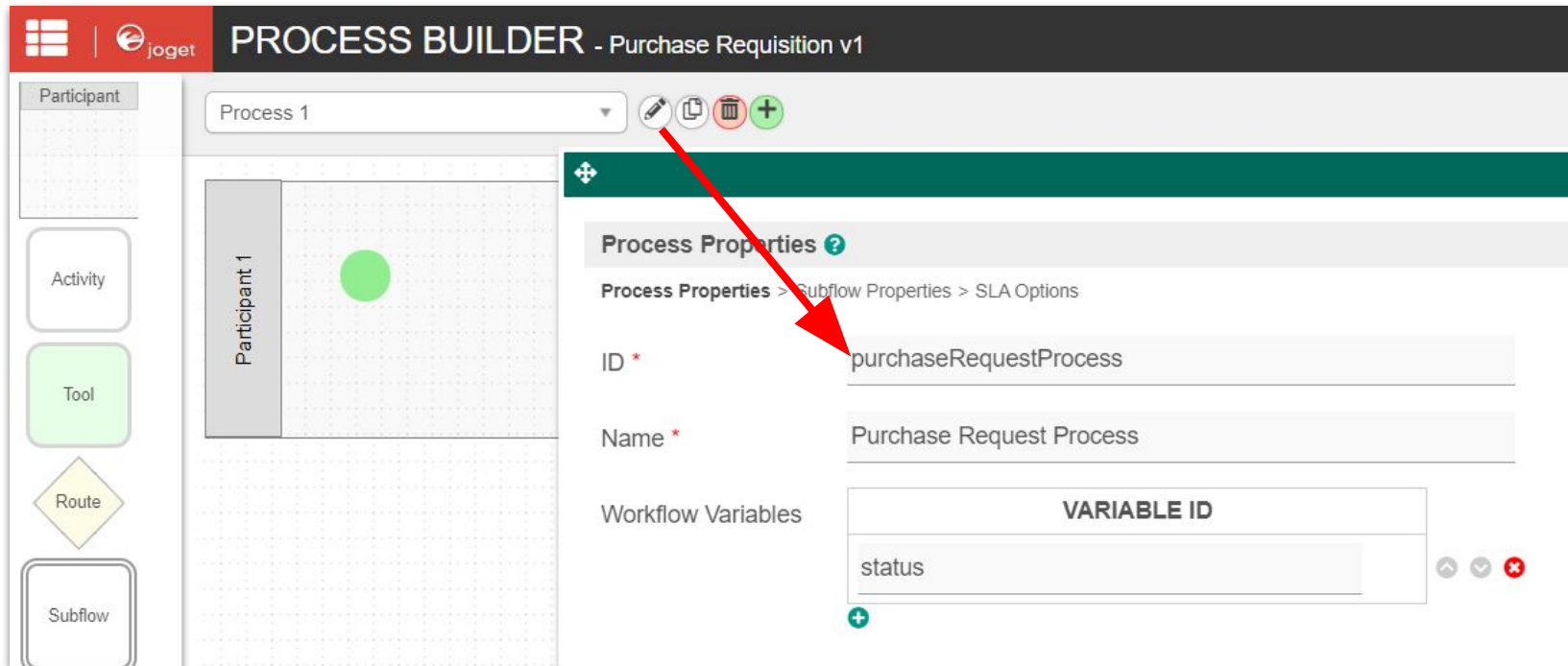


The screenshot displays the Joget DX Enterprise interface. On the left, a sidebar contains navigation options: 'Design App: Purchase Requisition' (with 'Version 1' and 'Not Published' buttons), 'Forms & UI', 'Processes' (highlighted with a red box), and 'Properties & Export'. The main area shows a 'Design Processes' button (also highlighted with a red box) and an 'Update via Saved XPD' button. Below these buttons, a message states: 'No process found. Click on Design Process to start designing your process.' A red arrow points from the 'Design Processes' button to the 'Process Builder' window. The 'Process Builder' window is titled 'PROCESS BUILDER - Purchase Requisition v1' and features a 'Participant' dropdown menu set to 'Process 1'. The main workspace shows a vertical bar labeled 'Participant 1' and a green circle on a grid background. A toolbar on the left of the workspace includes icons for 'Participant', 'Activity', 'Tool', and 'Route'.



# 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



The screenshot shows the Joget Process Builder interface for a process titled "Purchase Requisition v1". The main workspace displays a process diagram with a participant named "Participant 1" and a green circle representing an activity. A red arrow points from the "Process Properties" panel to the "ID" field.

**Process Properties**

Process Properties > Subflow Properties > SLA Options

ID \* purchaseRequestProcess

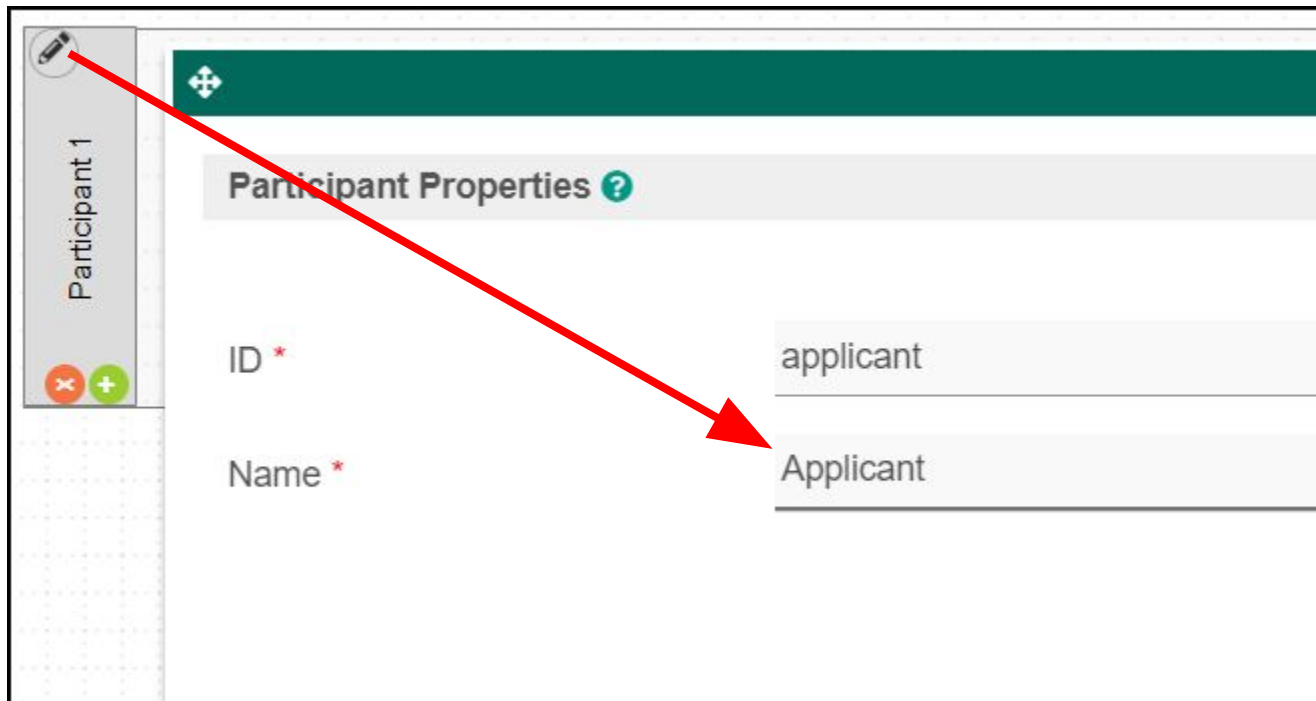
Name \* Purchase Request Process

Workflow Variables

VARIABLE ID
status

# Defining Your Participant

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

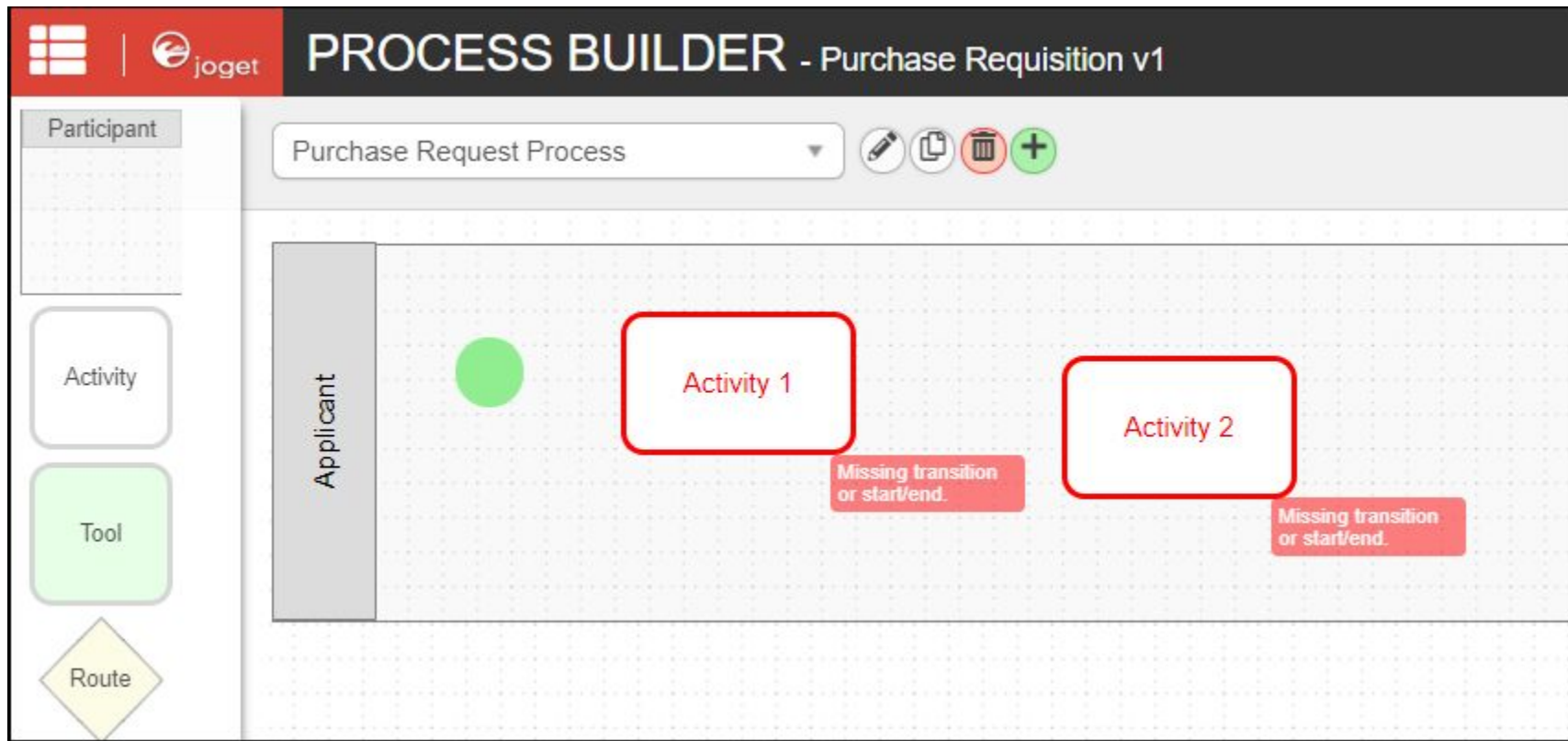


The screenshot shows the 'Participant Properties' dialog box in Joget. The dialog has a title bar with a plus sign icon. Below the title bar, there is a section titled 'Participant Properties' with a help icon. The 'ID' field is labeled 'ID \*' and contains the text 'applicant'. The 'Name' field is labeled 'Name \*' and contains the text 'Applicant'. A red arrow points from the 'Participant 1' label in the left sidebar to the 'Name' field.

ID *	applicant
Name *	Applicant

# Smart Guide

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



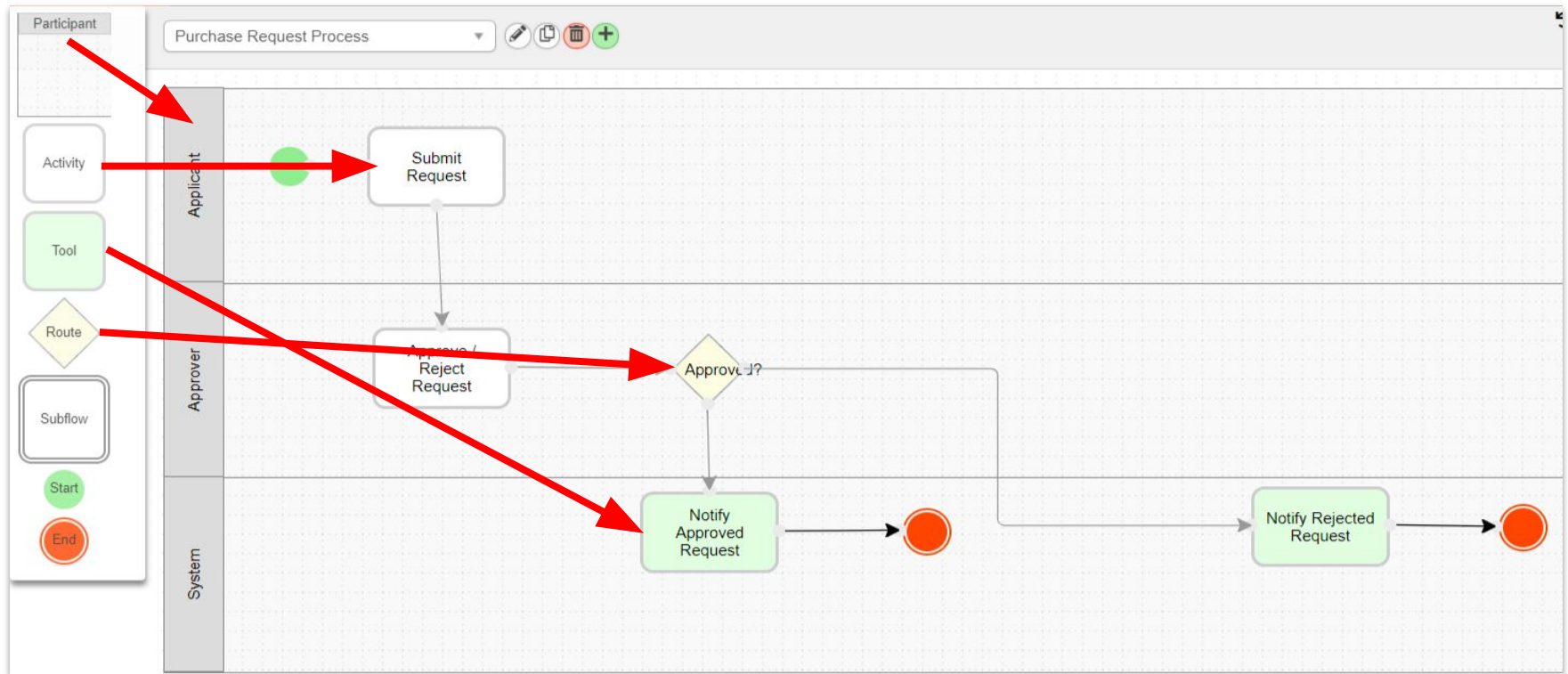
# About Smart Guide

---

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

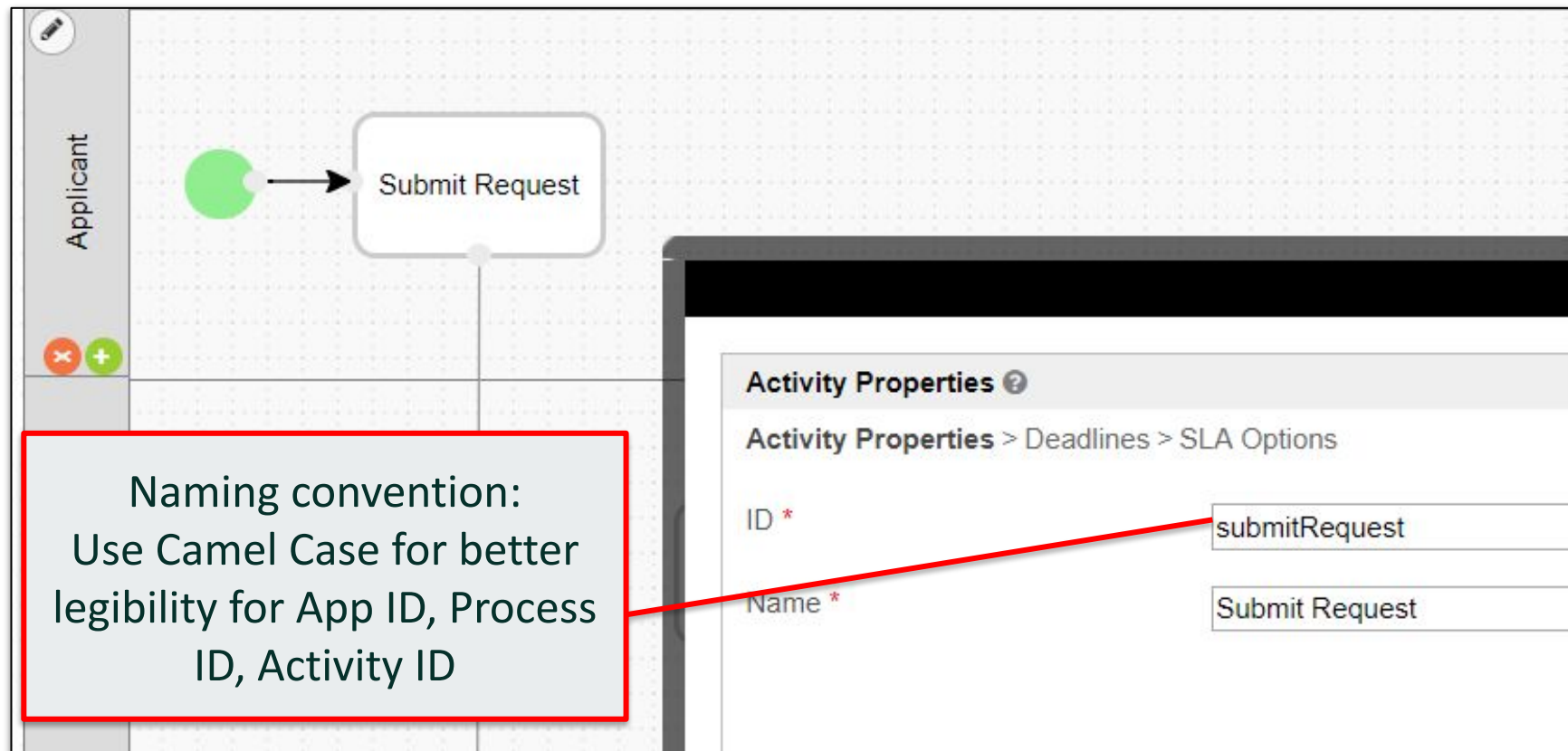
# Insert Activities and Transitions

- 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).



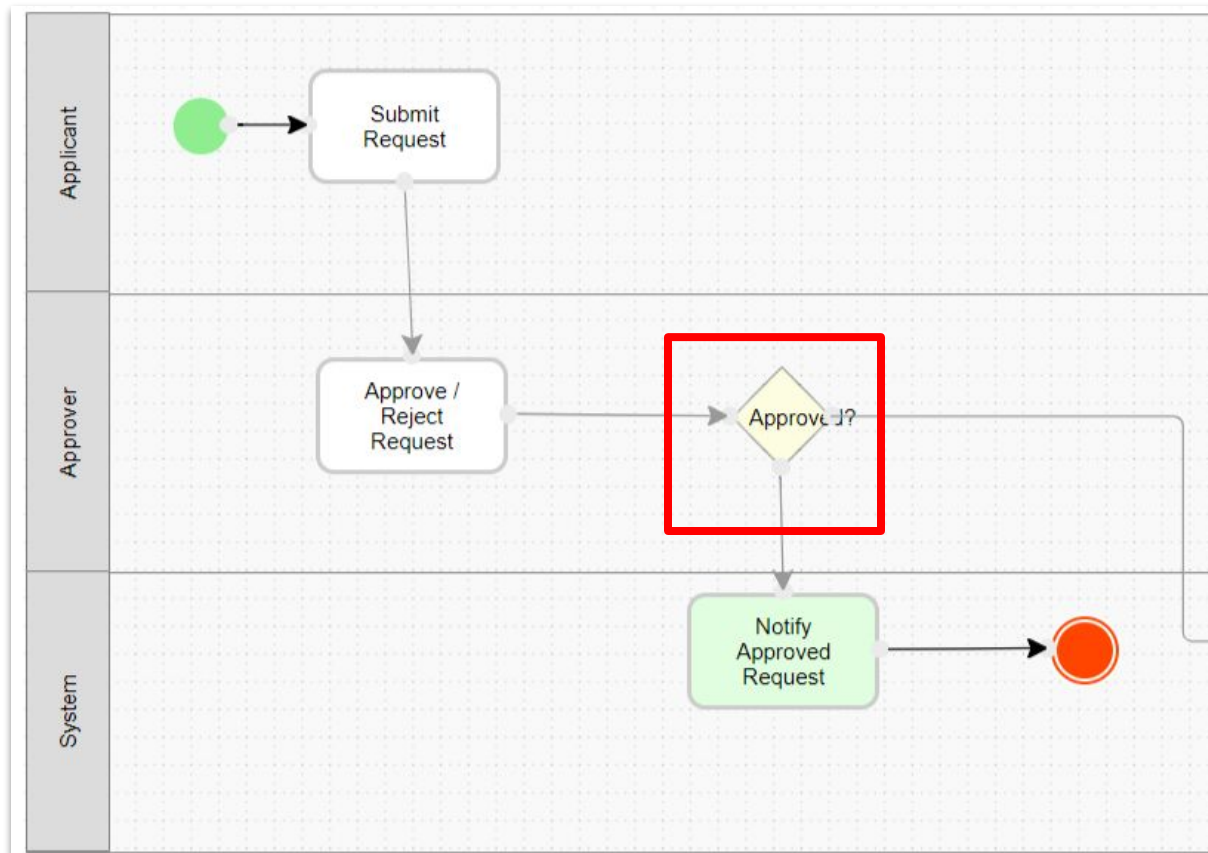
The screenshot shows a workflow editor with an activity node labeled "Submit Request". A red box highlights a note about naming conventions. To the right, the "Activity Properties" panel is open, showing the "ID" field set to "submitRequest" and the "Name" field set to "Submit Request". A red arrow points from the note to the "ID" field.

**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

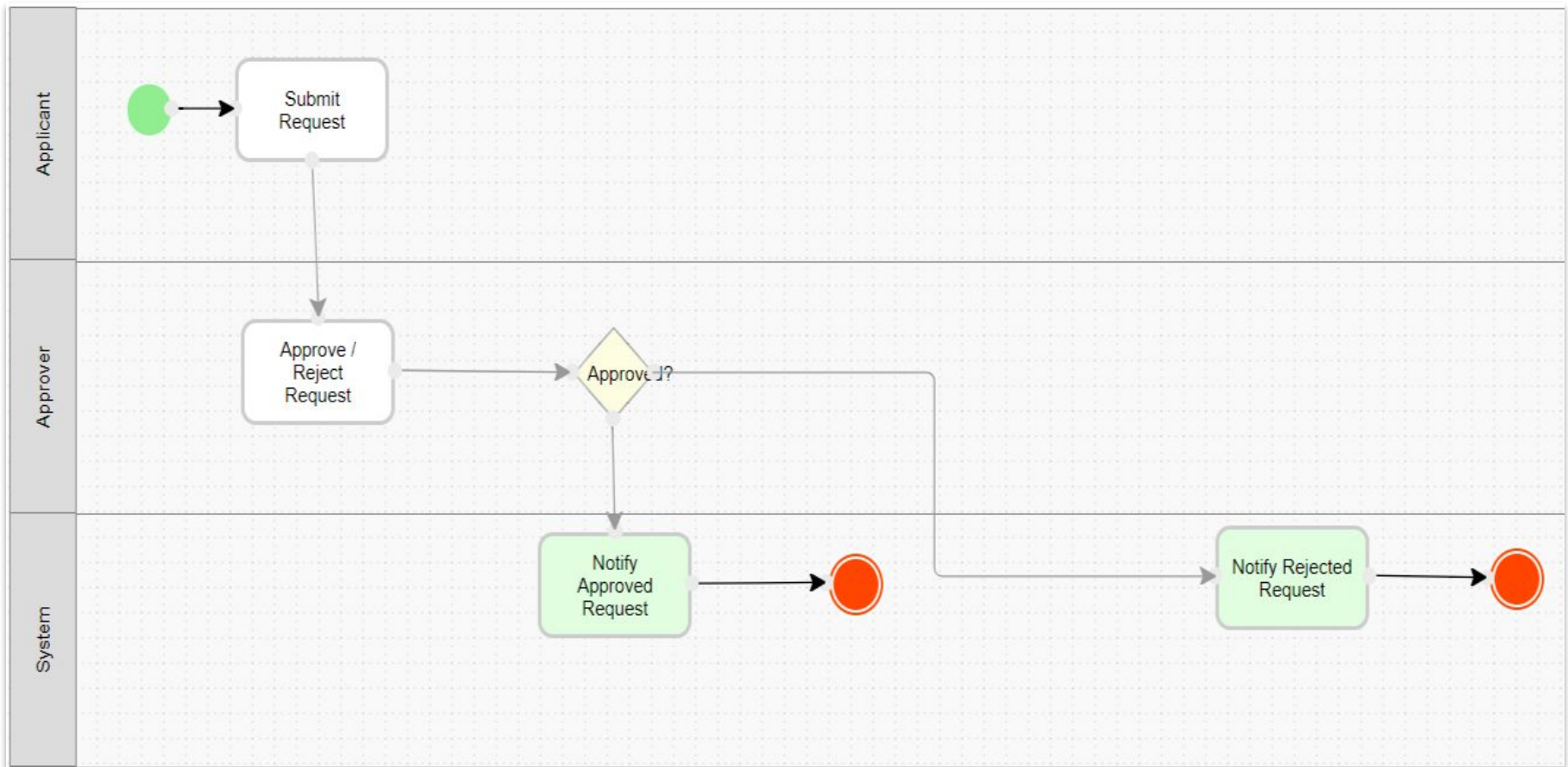
# Implementing the Business Rule

- 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.

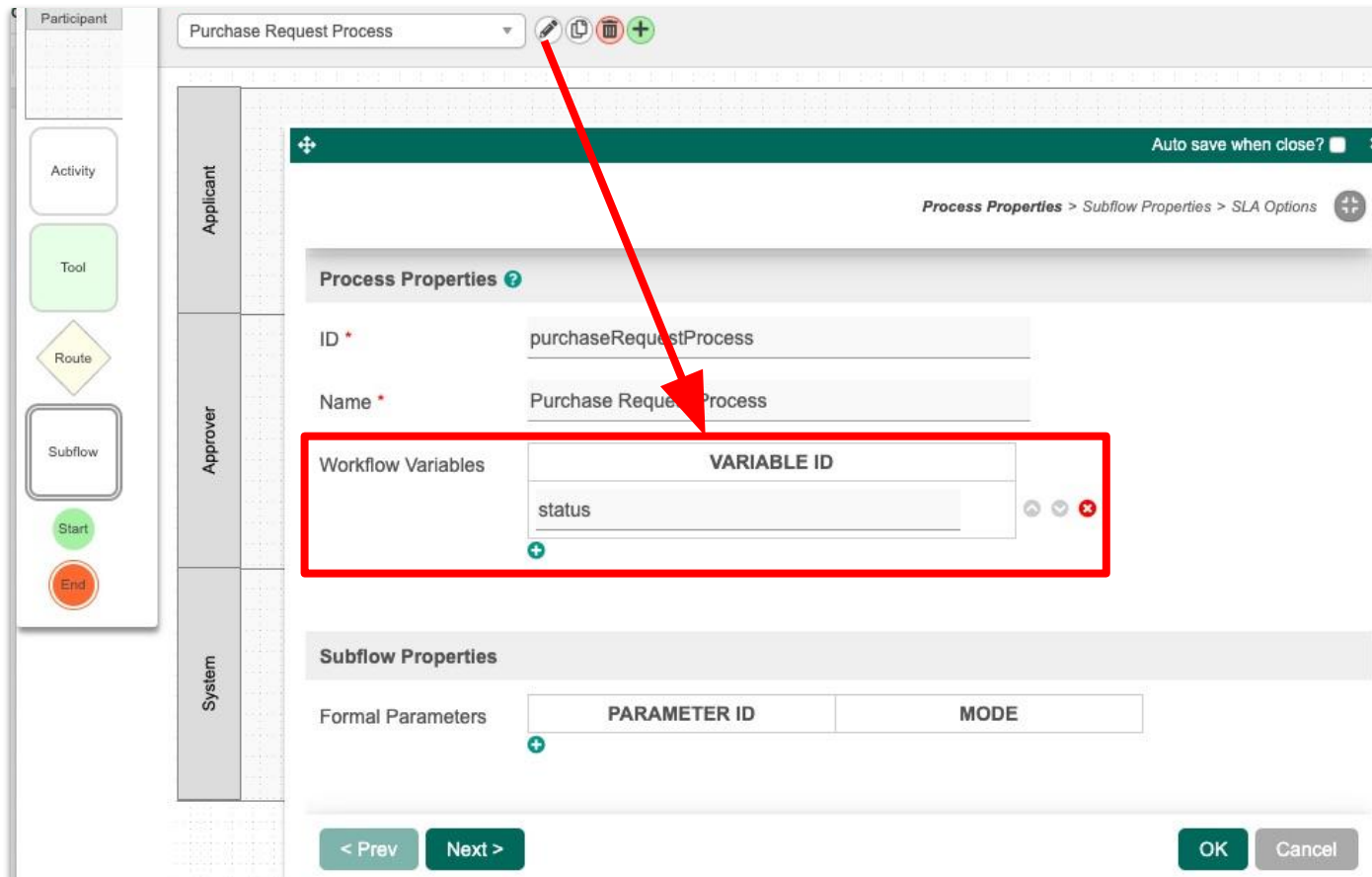
The screenshot displays the Joget BPMN editor interface. On the left, a 'Transition Properties' dialog box is open, featuring a title bar with 'Auto save when close?' and a close button. The dialog contains the following fields:

- Transition Properties** (Section Header)
- Name**: An empty text input field.
- Style**: Two radio button options:  Straight and  Orthogonal. A red arrow points from the 'Straight' option to the transition point in the diagram.
- Type**: A dropdown menu set to 'Normal'.
- Buttons**: 'OK' and 'Cancel' buttons at the bottom.

The main workspace shows a BPMN diagram on a grid background. It includes a yellow diamond-shaped decision node labeled 'Approv ?'. Two outgoing flows lead to rounded rectangular activity nodes: 'Notify Approved Request' (green) and 'Notify Rejected Request' (red). A red error message box with the text 'Missing transition or start/end.' is positioned near the transition point between the decision node and the 'Notify Rejected Request' activity. A red circle with a white center is located on the flow leading to 'Notify Approved Request'.

# Declaring the Workflow Variable

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



The screenshot shows the 'Purchase Request Process' configuration window. The 'Process Properties' section is expanded, showing the following details:

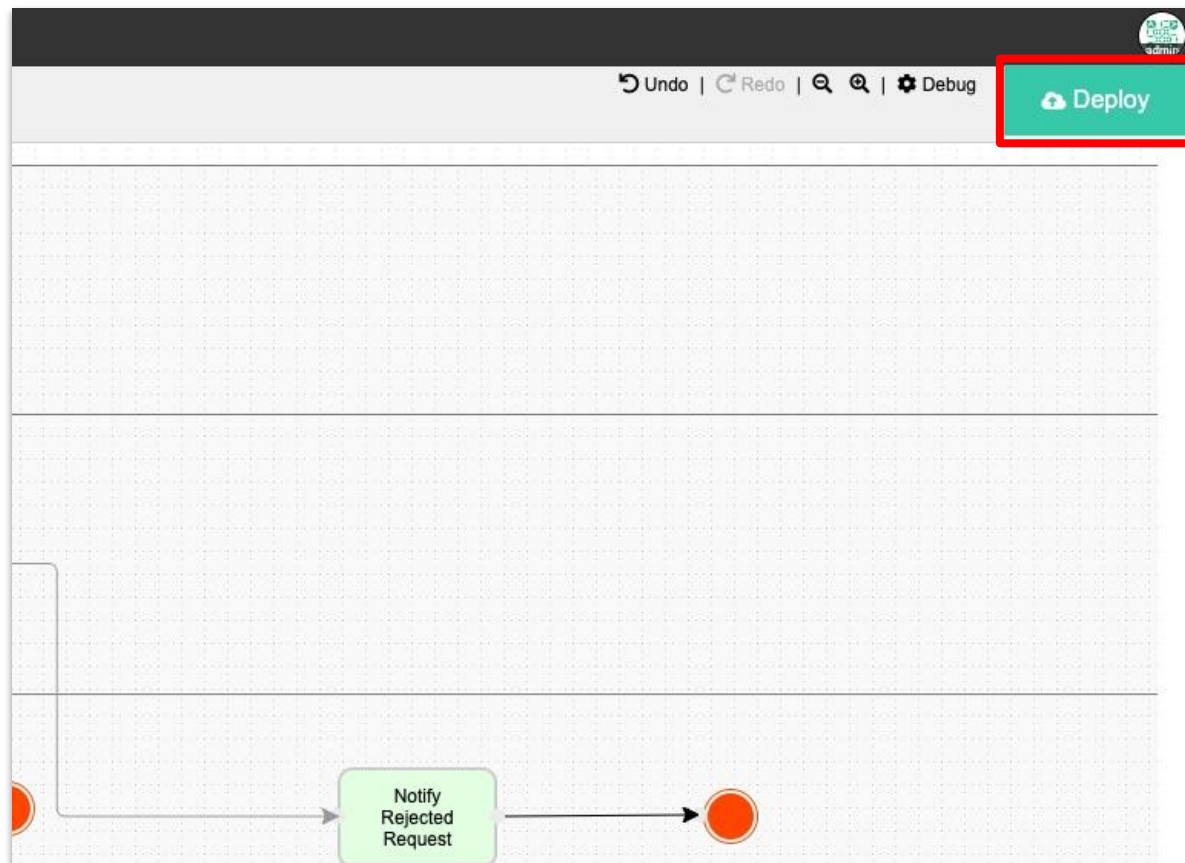
- ID \***: purchaseRequestProcess
- Name \***: Purchase Request Process
- Workflow Variables**: A table with the following content:
 

VARIABLE ID	MODE
status	

The 'Subflow Properties' section is also visible, showing a table for 'Formal Parameters' with columns 'PARAMETER ID' and 'MODE'.

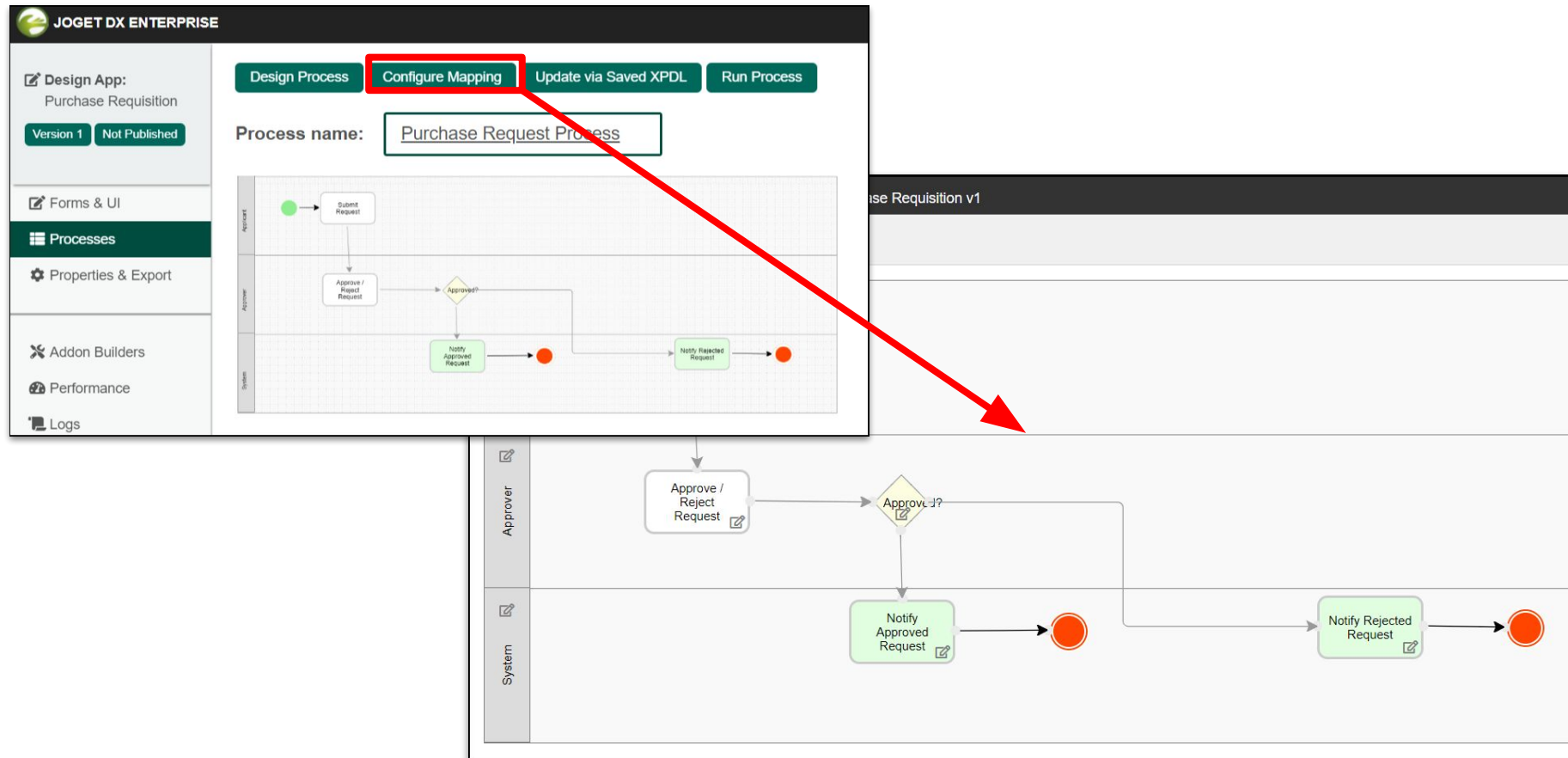
# Deploying Your Process Design

- 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.

Joget
PROCESS MAPPER - Purchase Requisition v1

Purchase Request Process

Applicant	Submit Request
Approver	Approve / Reject Request
System	Notify Approved Request

Approved?

MAP ROUTES TO PLUGINS - APPROVED? (ROUTE1)

PLUGIN NAME	PLUGIN DESCRIPTION	PLUGIN VERSION
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 ↺ ↻ Displaying 1 to 3 of 3 items

45

# Implementing the Business Rules

- Choose Simple Rules Decision.

### MAP ROUTES TO PLUGINS - APPROVED? (ROUTE1)

Search

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

15   Page 1 of 1   Displaying 1 to 3 of 3 it

### PLUGIN CONFIGURATION - APPROVED? (ROUTE1)

Configure Simple Rules Decision

Rules \*

**IF**

And ▾

status Equal To

**THEN**

Transition ▾ transition3 (Notify Approved Request) ▾

[+ Add Rule](#)

**ELSE THEN**

Transition ▾ 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)

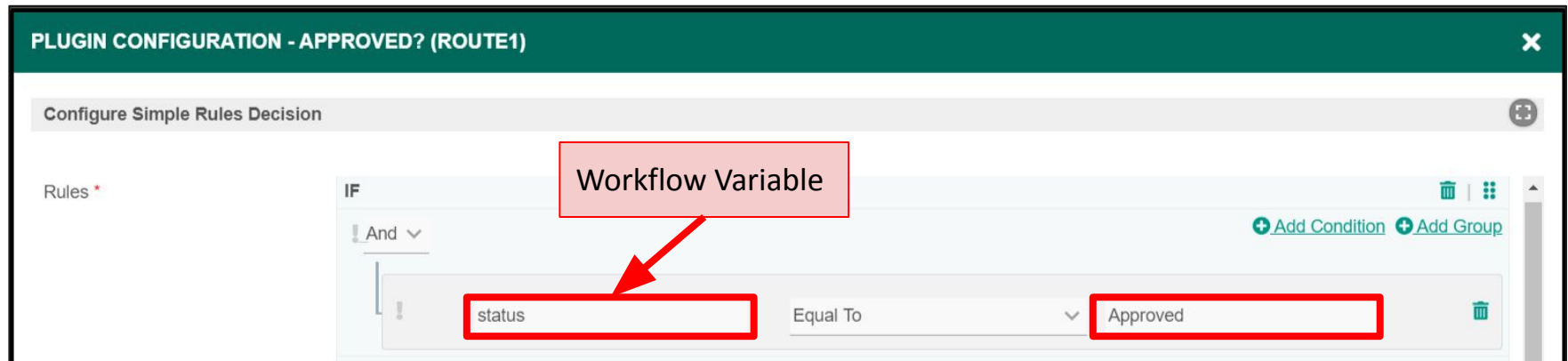
**ELSE THEN**

Transition transition4 (Notify Rejected Request)

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.

PLUGIN CONFIGURATION - APPROVED? (ROUTE2)

Configure Simple Rules Decision

Rules \*

IF

And

#form.purchase\_requests.appro

Equal To

Approved

Hash Variable

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

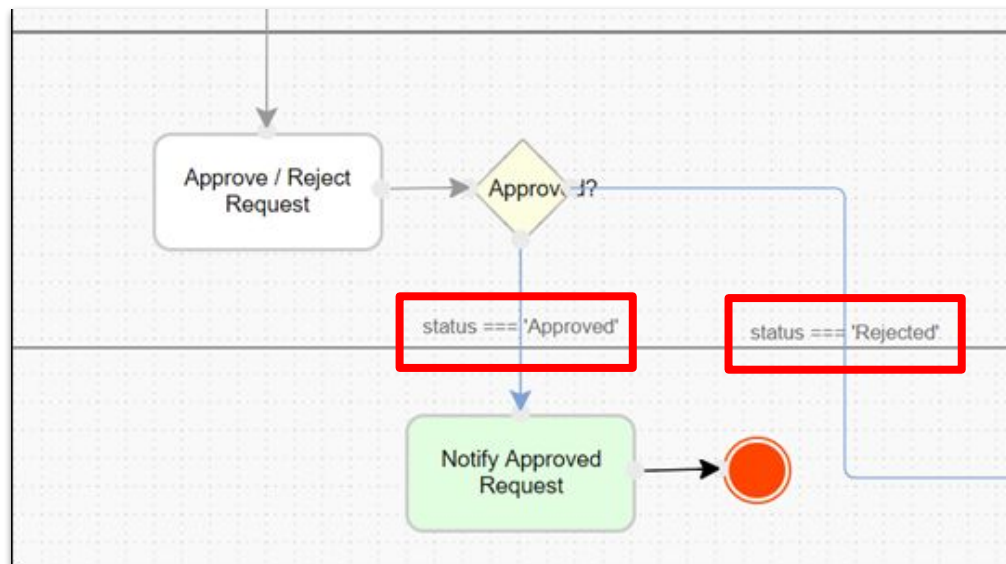
# If and otherwise...

---

- 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.



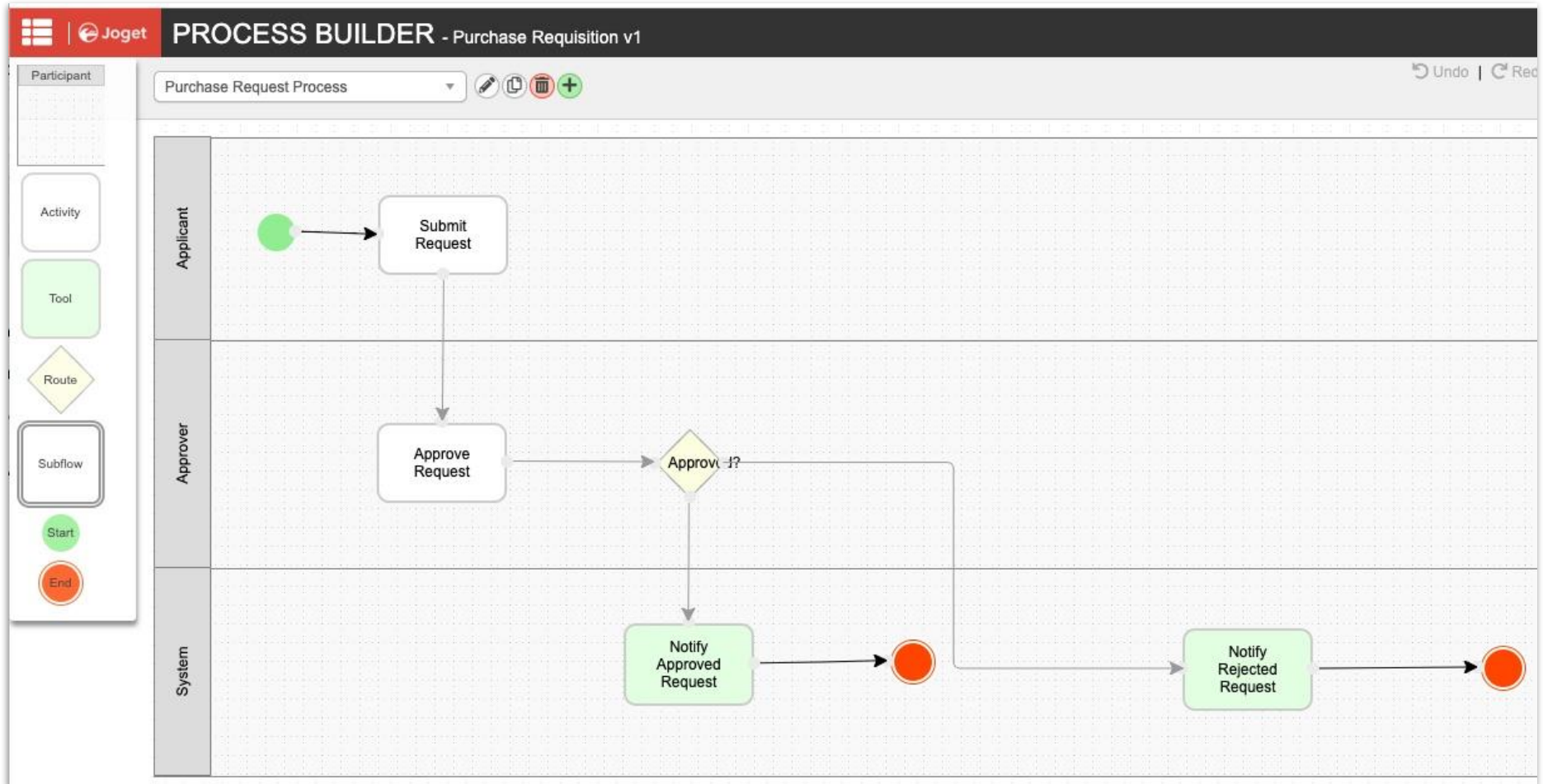
# Advantages of using Process Mapper

---

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

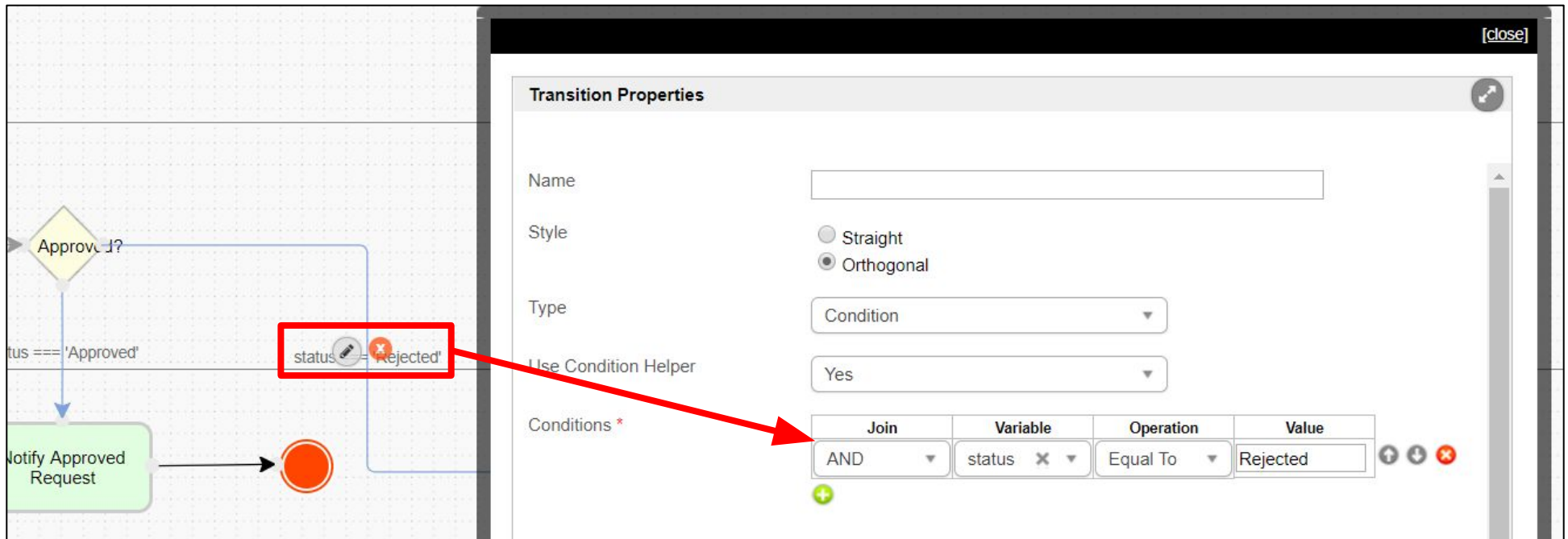
# Alternative Business Rule Implementation

- App Design > Processes > Design Process.



# Alternative - Defining the Transition Condition

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



**Transition Properties** [close]

Name:

Style:  Straight  Orthogonal

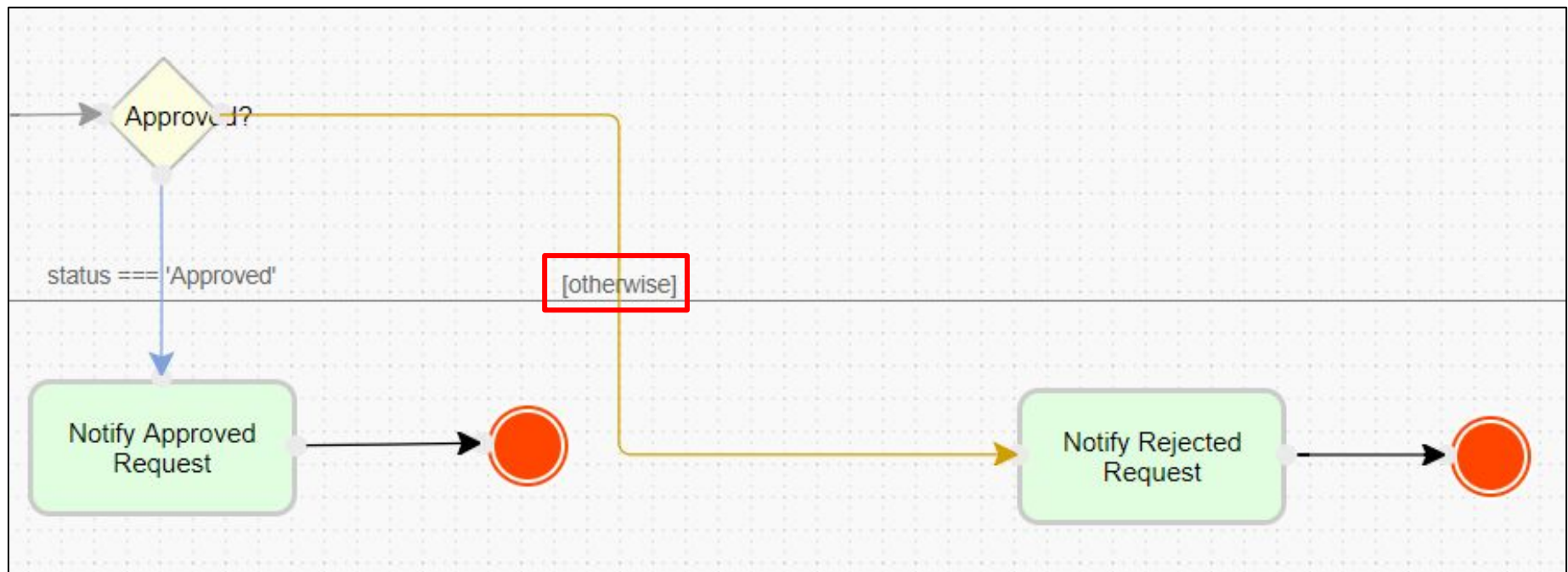
Type:

Use Condition Helper:

Join	Variable	Operation	Value
AND	status	Equal To	Rejected

# 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

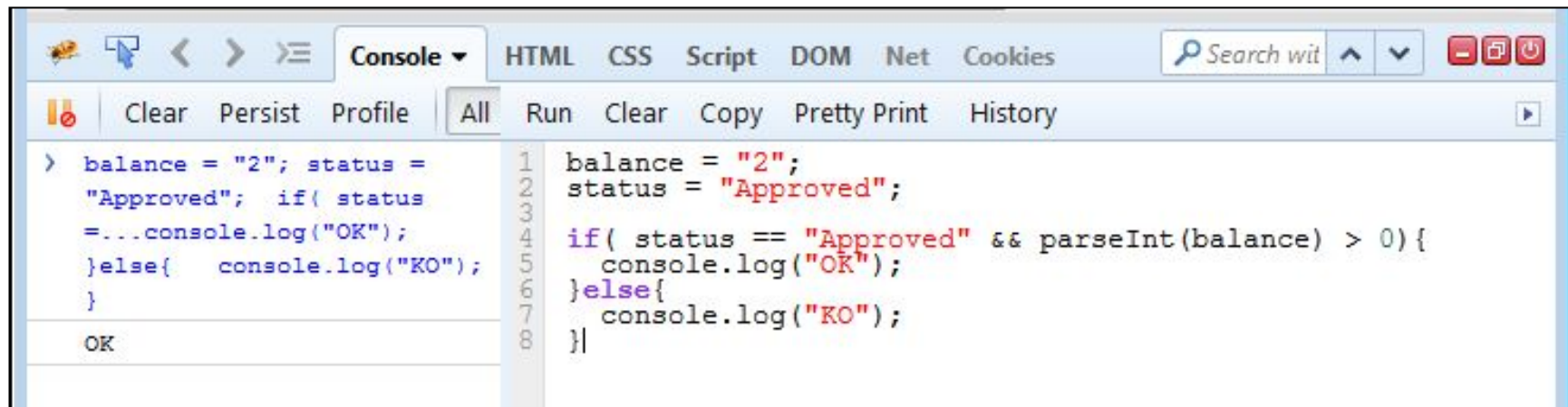
---

- 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 a browser's developer console with the 'Console' tab selected. The console displays the following JavaScript code snippet:

```
1 balance = "2";
2 status = "Approved";
3
4 if( status == "Approved" && parseInt(balance) > 0){
5     console.log("OK");
6 }else{
7     console.log("KO");
8 }
```

The execution result shown in the console is "OK".

# 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)

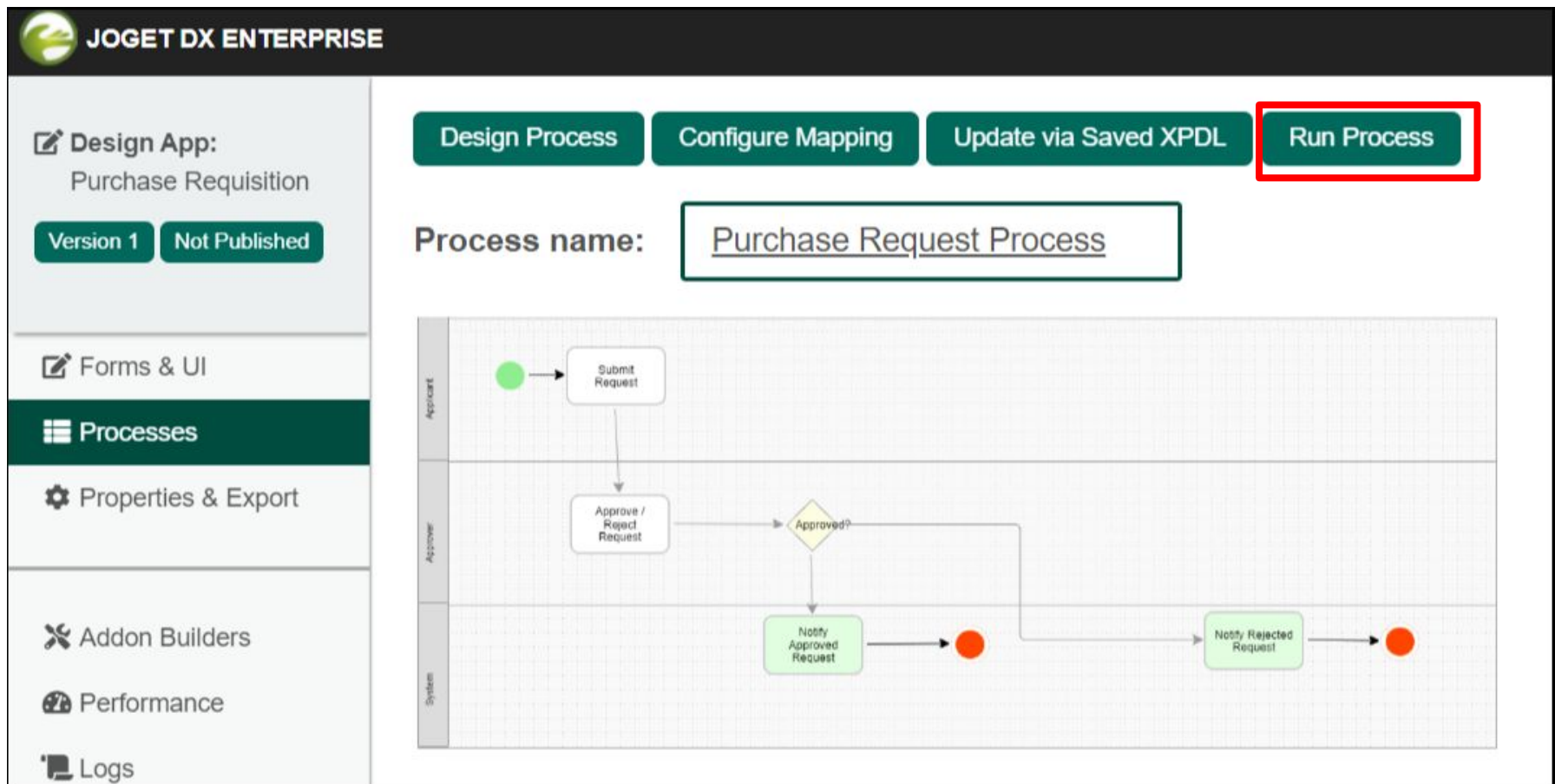
# How to Test?

---

- 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

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

Process name: Purchase Request Process

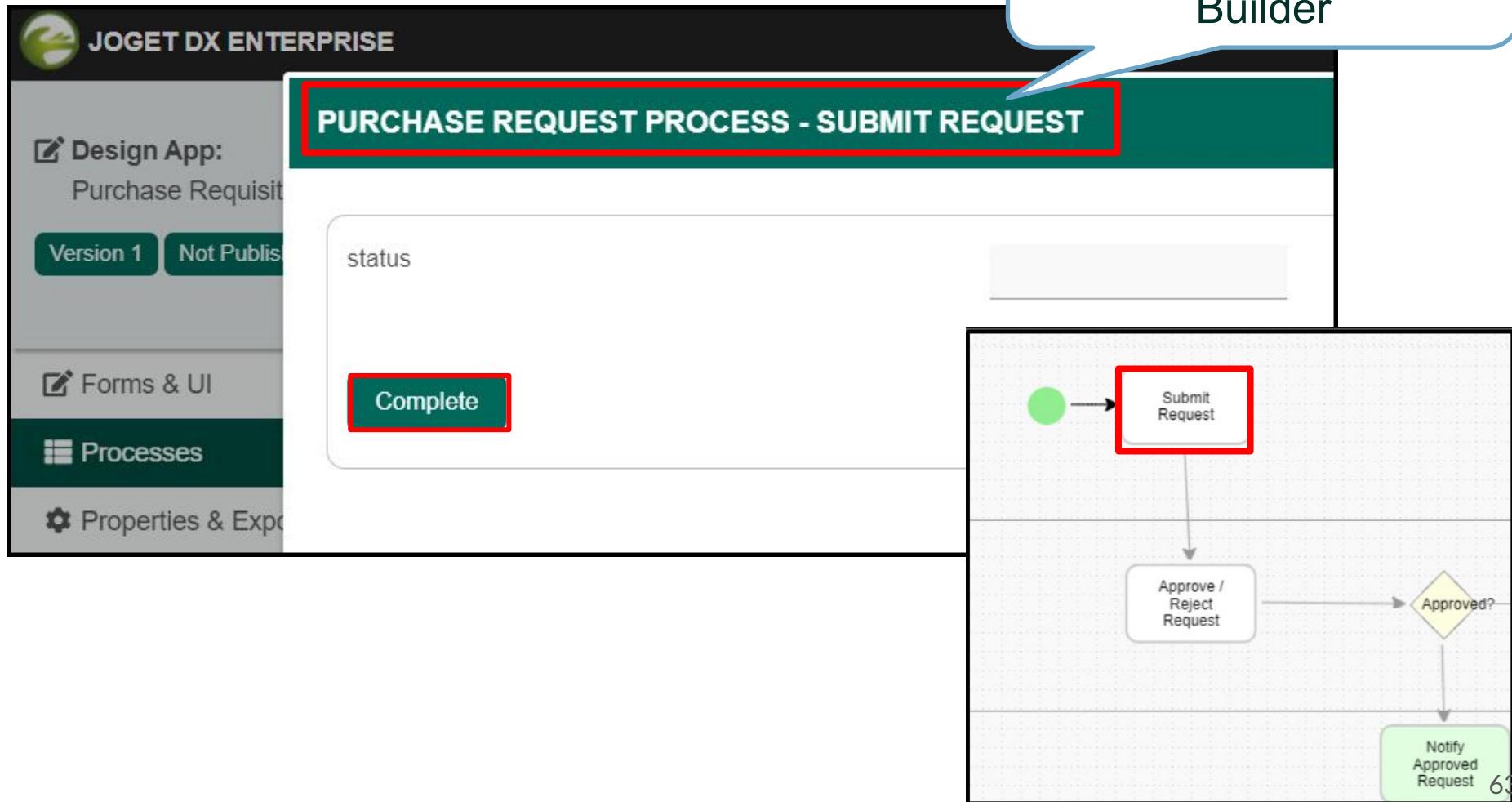
```

    graph TD
      subgraph Applicant
        Start(( )) --> Submit[Submit Request]
      end
      subgraph Approver
        Submit --> Approve[Approve / Reject Request]
        Approve --> Approved{Approved?}
      end
      subgraph System
        Approved --> NotifyApproved[Notify Approved Request]
        Approved --> NotifyRejected[Notify Rejected Request]
        NotifyApproved --> End1(( ))
        NotifyRejected --> End2(( ))
      end
  
```

# Verify the Process

- Complete the assignment to proceed.

HINT: Activity name is configured in Process Builder

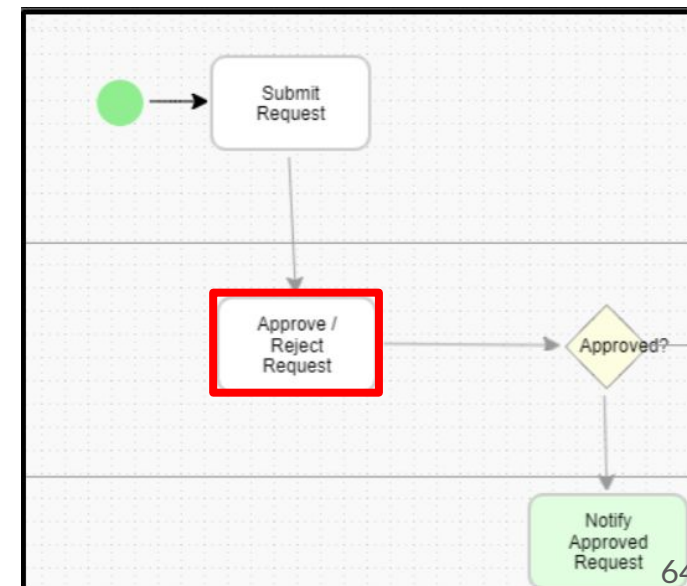
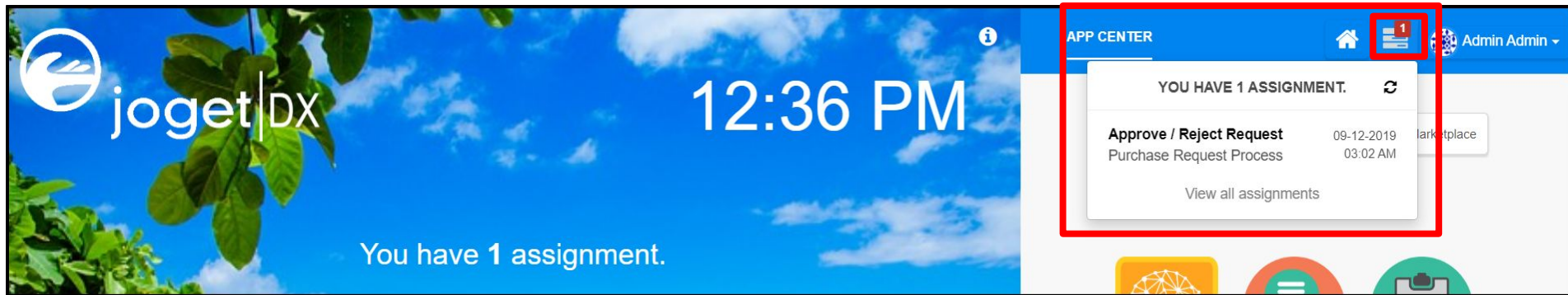


The screenshot displays the Joget DX Enterprise interface for the 'PURCHASE REQUEST PROCESS - SUBMIT REQUEST' activity. The activity name is highlighted in a red box. The 'status' field is set to 'Complete', also highlighted in a red box. A callout box points to the activity name with the hint: 'HINT: Activity name is configured in Process Builder'. An inset diagram shows the process flow: Start -> Submit Request -> Approve / Reject Request -> Approved? -> Notify Approved Request.

```
graph TD; Start(( )) --> SubmitRequest[Submit Request]; SubmitRequest --> ApproveReject[Approve / Reject Request]; ApproveReject --> Approved{Approved?}; Approved --> NotifyApproved[Notify Approved Request];
```

# Accessing the Assignments

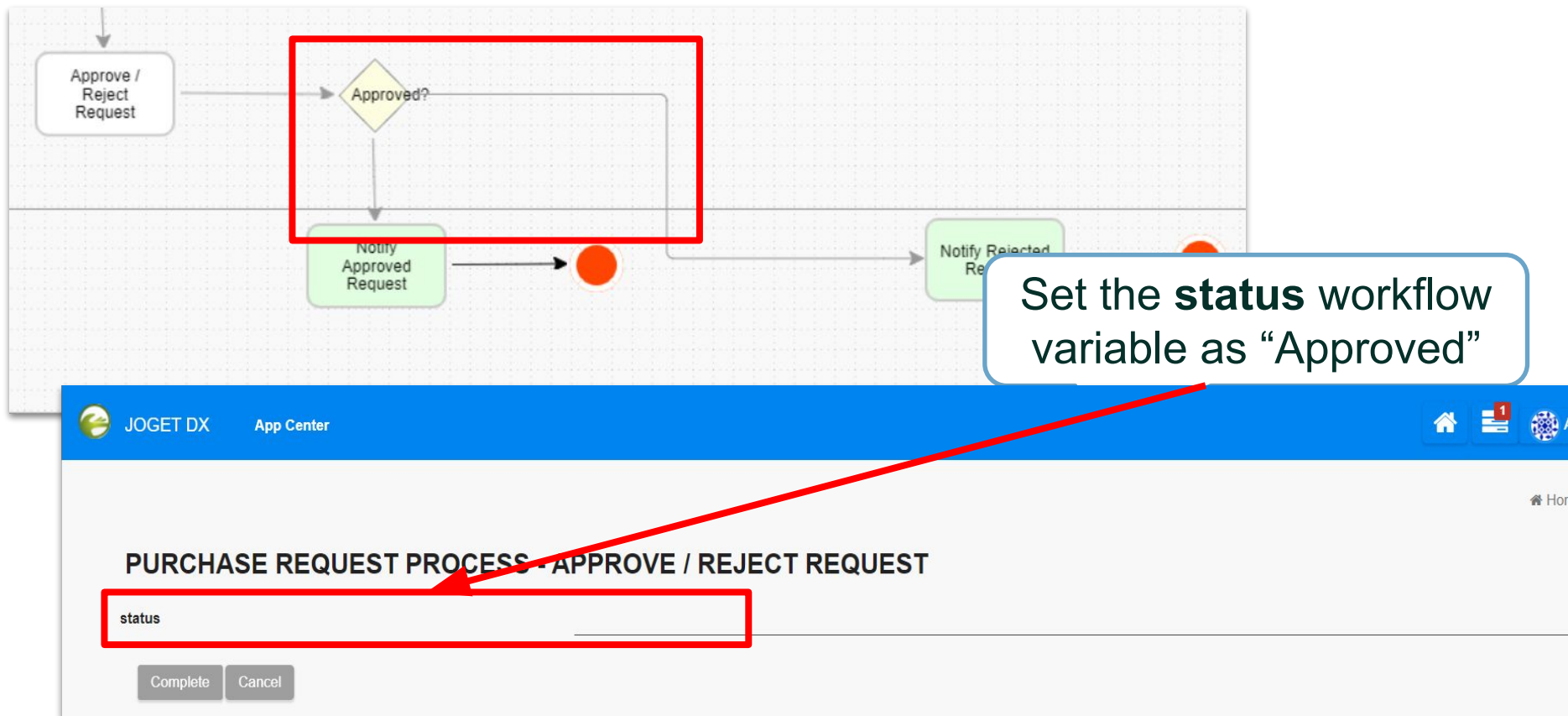
- App Center -> Universal Inbox





# Determining the Flow Using Route

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



Approve / Reject Request

Approved?

Notify Approved Request

Notify Rejected Request

Set the **status** workflow variable as "Approved"

JOGET DX App Center

PURCHASE REQUEST PROCESS - APPROVE / REJECT REQUEST

status

Complete Cancel

# 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?

---

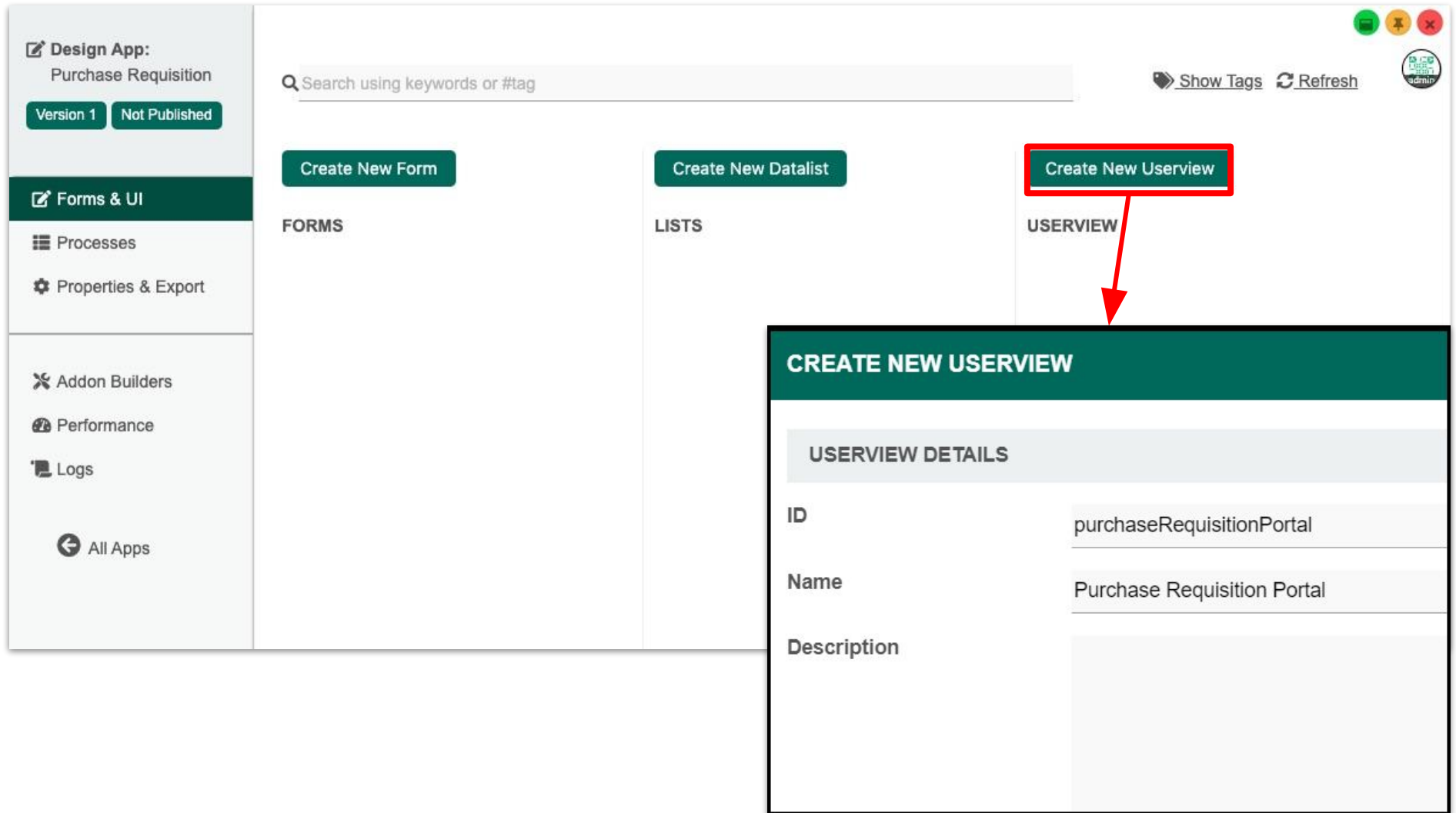
- There are 2 ways in general.
  - Make use of the “**Run Process**” button in App Design > Processes (only for admin)
  - Make use of **Usserview** (for all users)

# 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 displays the Joget Designer interface for creating a new userview. The left sidebar shows the 'Design App: Purchase Requisition' with 'Version 1' and 'Not Published' buttons. The 'Forms & UI' section is active, with options for 'Processes' and 'Properties & Export'. The main workspace is divided into three columns: 'FORMS', 'LISTS', and 'USERVIEW'. The 'Create New Userview' button in the 'USERVIEW' column is highlighted with a red box, and a red arrow points to a modal window titled 'CREATE NEW USERVIEW'.

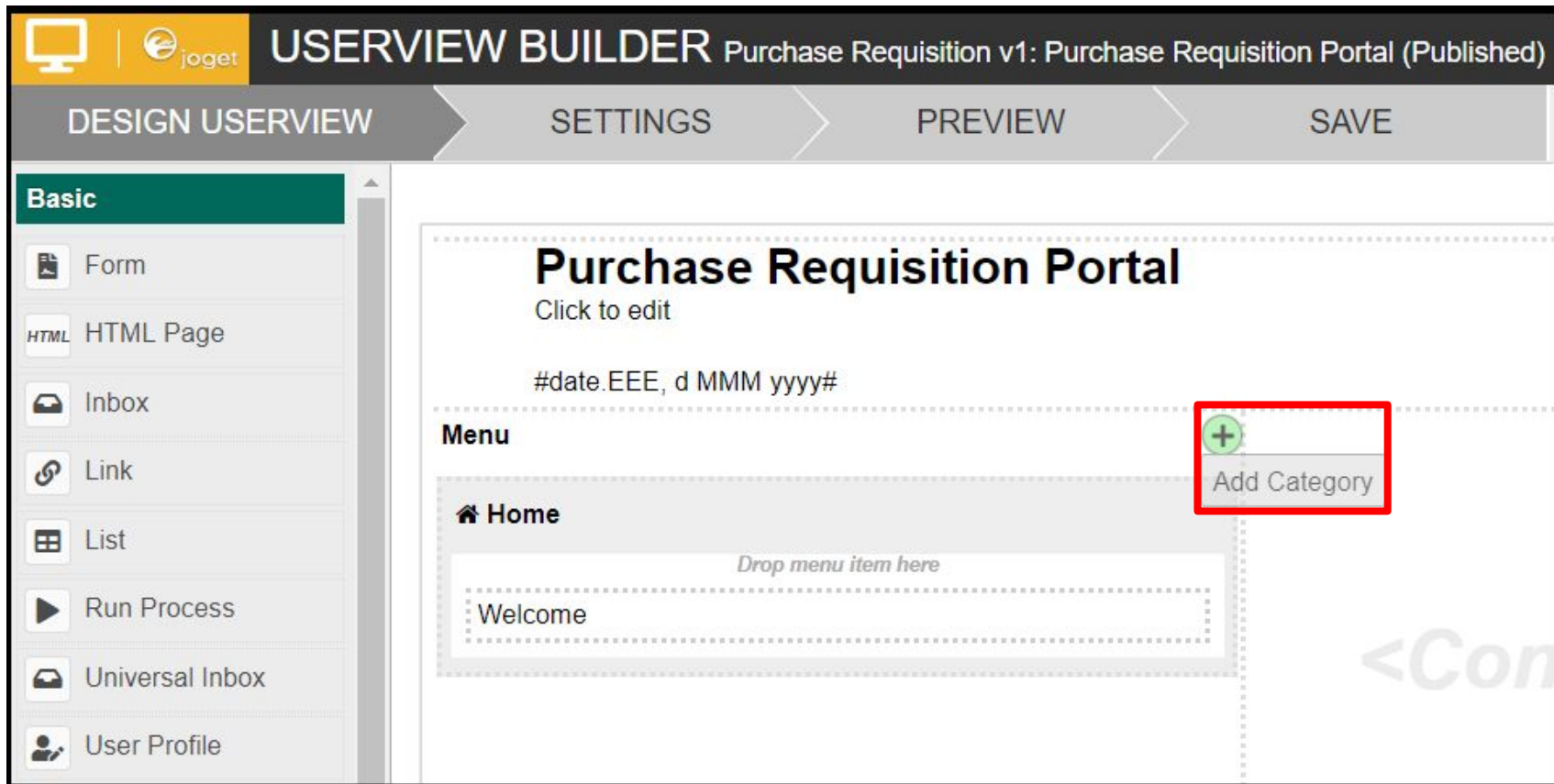
**CREATE NEW USERVIEW**

**USERVIEW DETAILS**

ID	purchaseRequisitionPortal
Name	Purchase Requisition Portal
Description	

# Userview Category

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



**USERVIEW BUILDER** Purchase Requisition v1: Purchase Requisition Portal (Published)

DESIGN USERVIEW    SETTINGS    PREVIEW    SAVE

**Basic**

- Form
- HTML Page
- Inbox
- Link
- List
- Run Process
- Universal Inbox
- User Profile

**Purchase Requisition Portal**  
Click to edit

#date.EEE, d MMM yyyy#

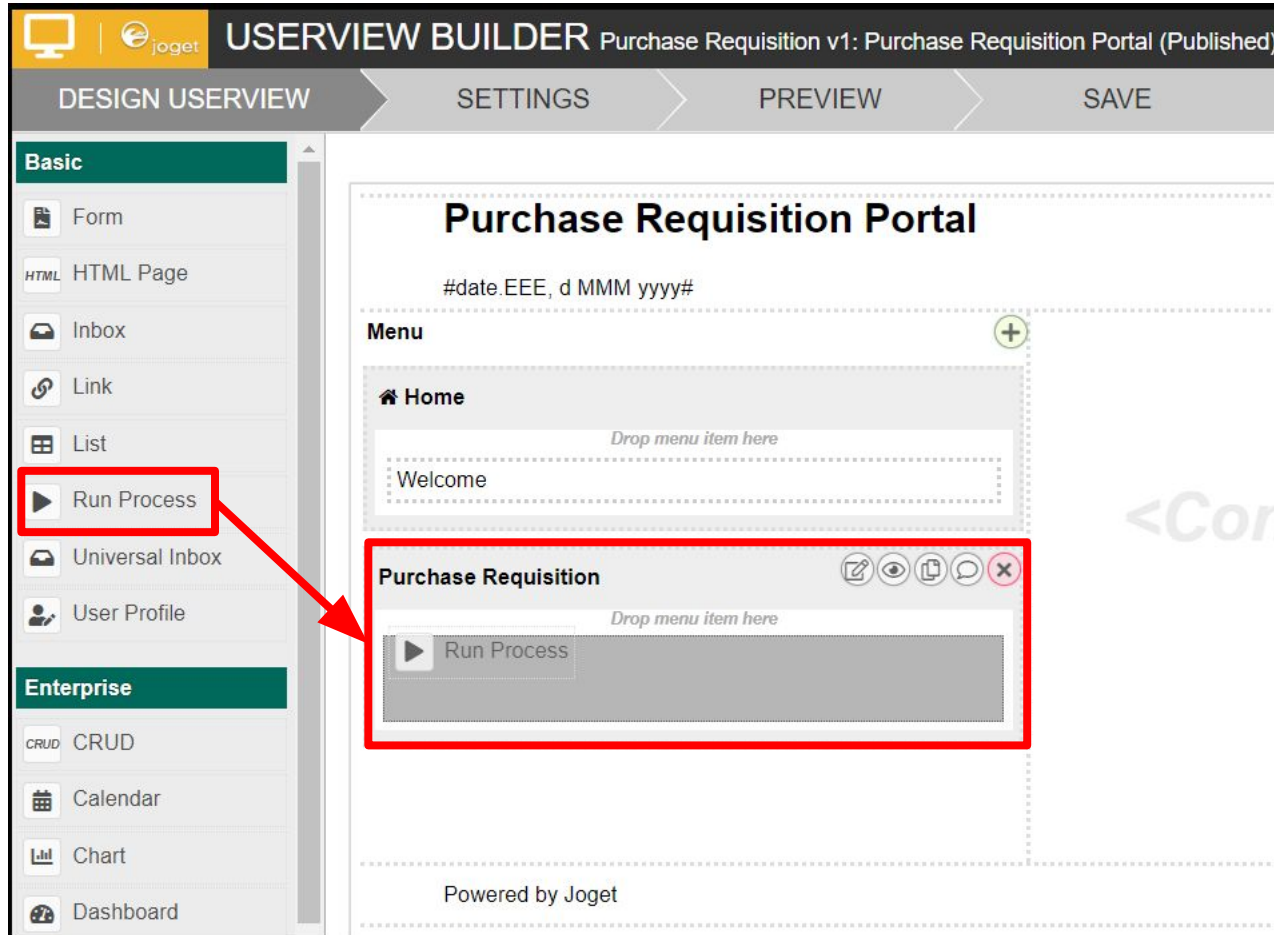
**Menu**

- Home
- Drop menu item here
- Welcome

+  
Add Category

# Add Run Process Into Userview

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

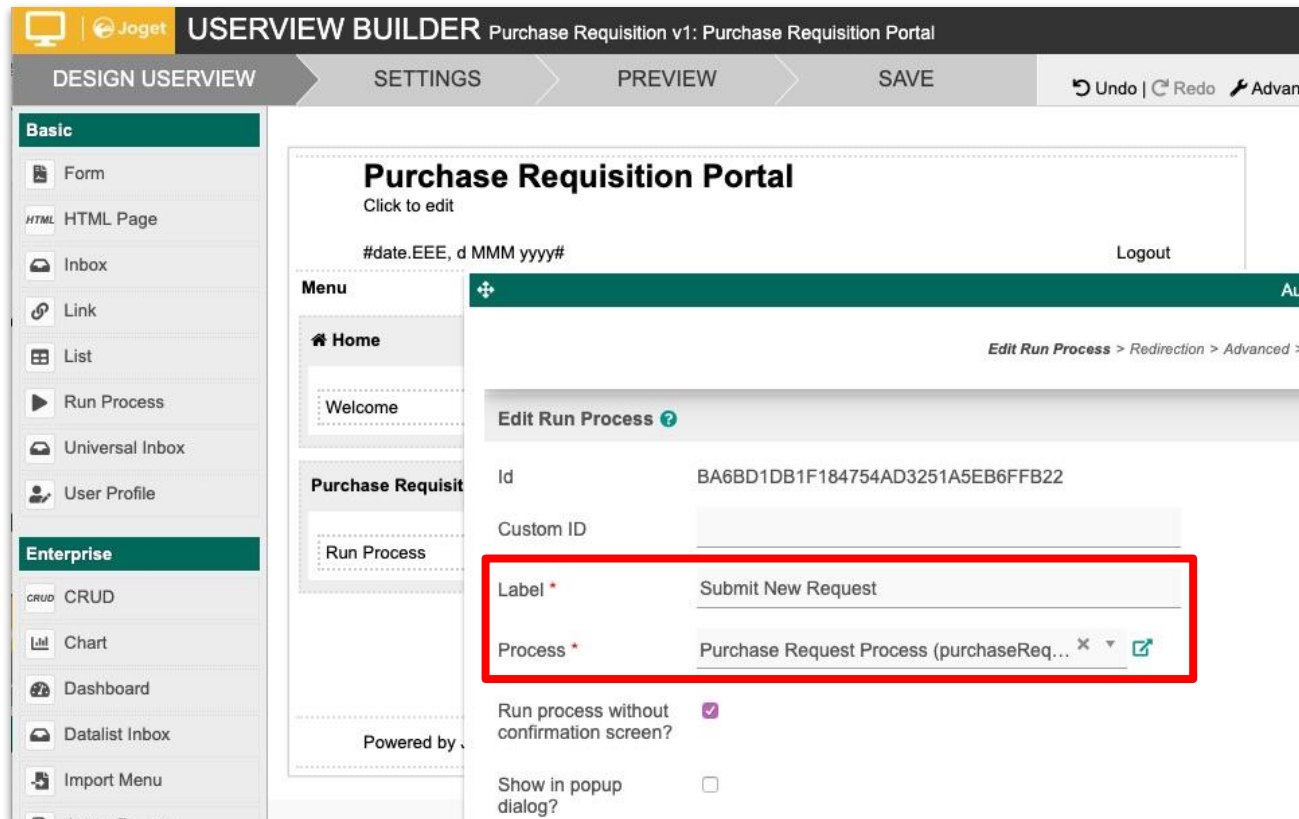


The screenshot displays the 'USERVIEW BUILDER' interface for a 'Purchase Requisition v1: Purchase Requisition Portal (Published)'. The interface includes a top navigation bar with 'DESIGN USERVIEW', 'SETTINGS', 'PREVIEW', and 'SAVE' tabs. On the left, a sidebar lists various widgets under 'Basic' and 'Enterprise' categories. The 'Run Process' widget is highlighted with a red box, and a red arrow points to its placement within a 'Purchase Requisition' menu item, which is also highlighted with a red box. The main workspace shows a 'Purchase Requisition Portal' with a date placeholder, a 'Menu' section containing 'Home' and 'Welcome' items, and a 'Purchase Requisition' menu item containing the 'Run Process' widget. The footer indicates 'Powered by Joget'.



# Integrate Run Process Into Userview

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



The screenshot displays the 'USERVIEW BUILDER' interface for a 'Purchase Requisition v1: Purchase Requisition Portal'. The interface includes a top navigation bar with 'DESIGN USERVIEW', 'SETTINGS', 'PREVIEW', and 'SAVE' tabs, along with 'Undo', 'Redo', and 'Advance' actions. A left sidebar contains a 'Basic' section with options like 'Form', 'HTML Page', 'Inbox', 'Link', 'List', 'Run Process', 'Universal Inbox', and 'User Profile', and an 'Enterprise' section with 'CRUD', 'Chart', 'Dashboard', 'Datalist Inbox', and 'Import Menu'. The main workspace shows a 'Purchase Requisition Portal' configuration. A 'Menu' section is visible on the left, with 'Home' and 'Purchase Requisition' items. The 'Purchase Requisition' item is expanded to show a 'Run Process' configuration. This configuration is highlighted with a red box and includes the following fields:

- Id:** BA6BD1DB1F184754AD3251A5EB6FFB22
- Custom ID:** (empty field)
- Label \*:** Submit New Request
- Process \*:** Purchase Request Process (purchaseReq...)
- Run process without confirmation screen?:**
- Show in popup dialog?:**

# Add Inbox Into Userview

---

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

# Publish the App

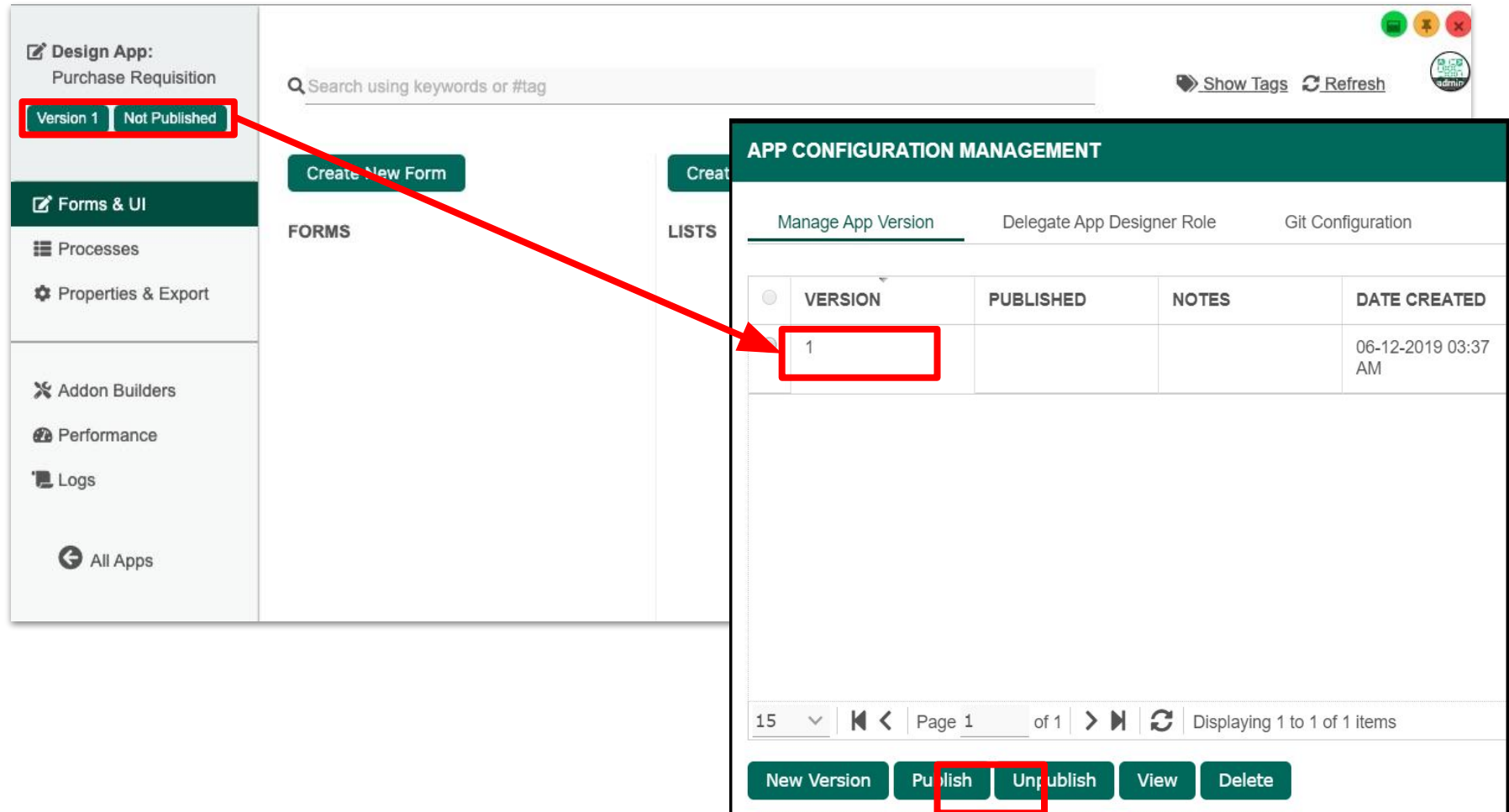
---

- 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.



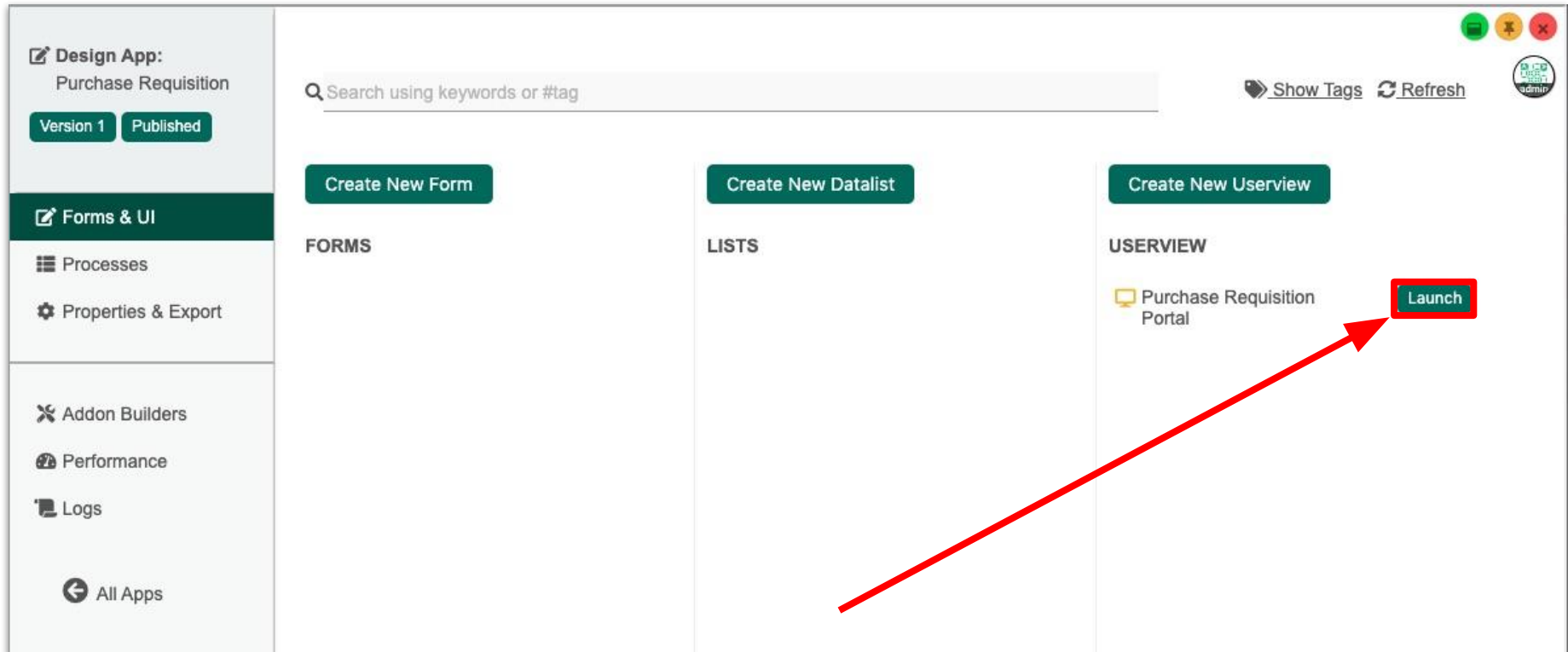
The screenshot shows the Joget application interface. On the left, the 'Design App' sidebar for 'Purchase Requisition' is visible, with 'Version 1 | Not Published' highlighted in a red box. A red arrow points from this box to a red box around the number '1' in the 'APP CONFIGURATION MANAGEMENT' table. The table has the following structure:

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

At the bottom of the table, the 'Publish' button is highlighted in a red box. The interface also includes a search bar, 'Show Tags' and 'Refresh' buttons, and a navigation bar with 'New Version', 'Publish', 'Unpublish', 'View', and 'Delete' buttons.

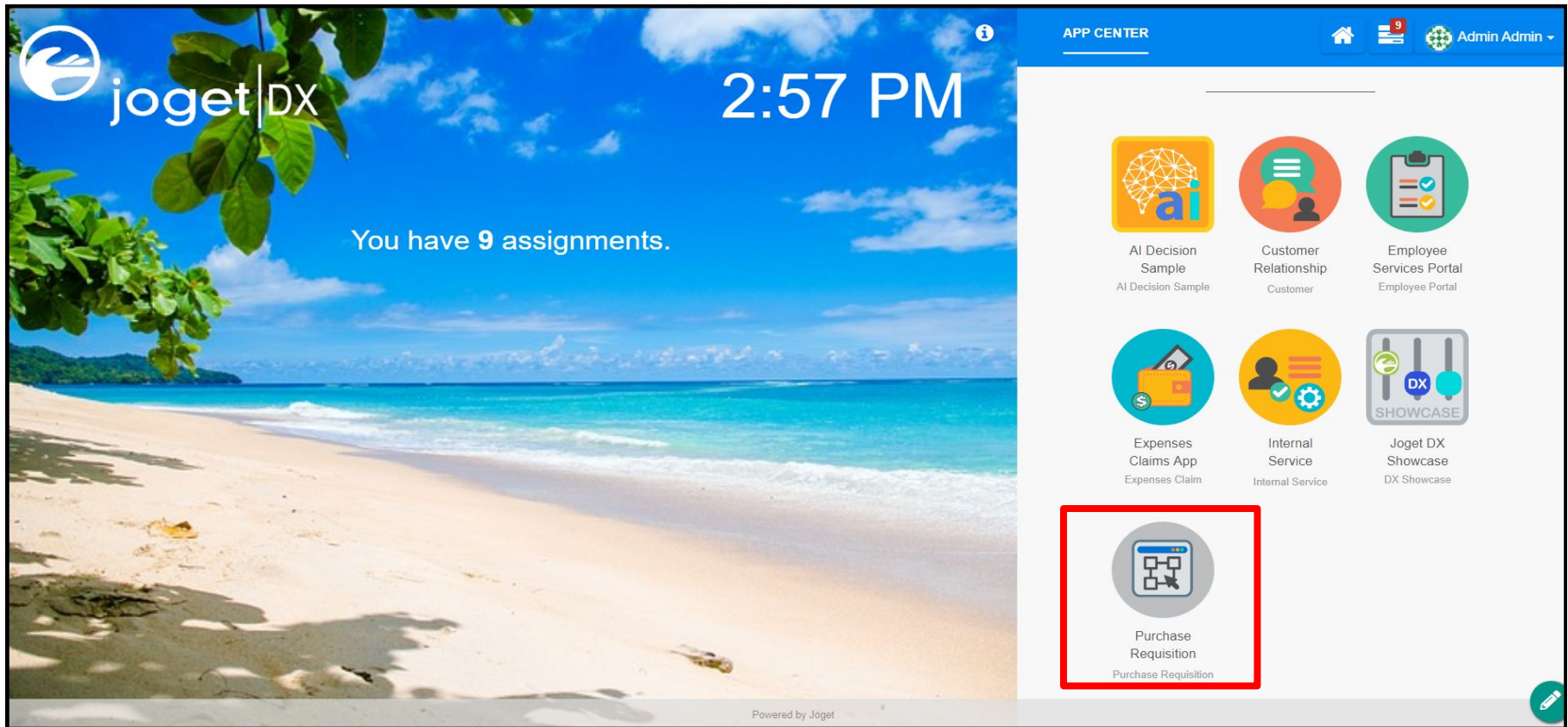
# Launching the Published App

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



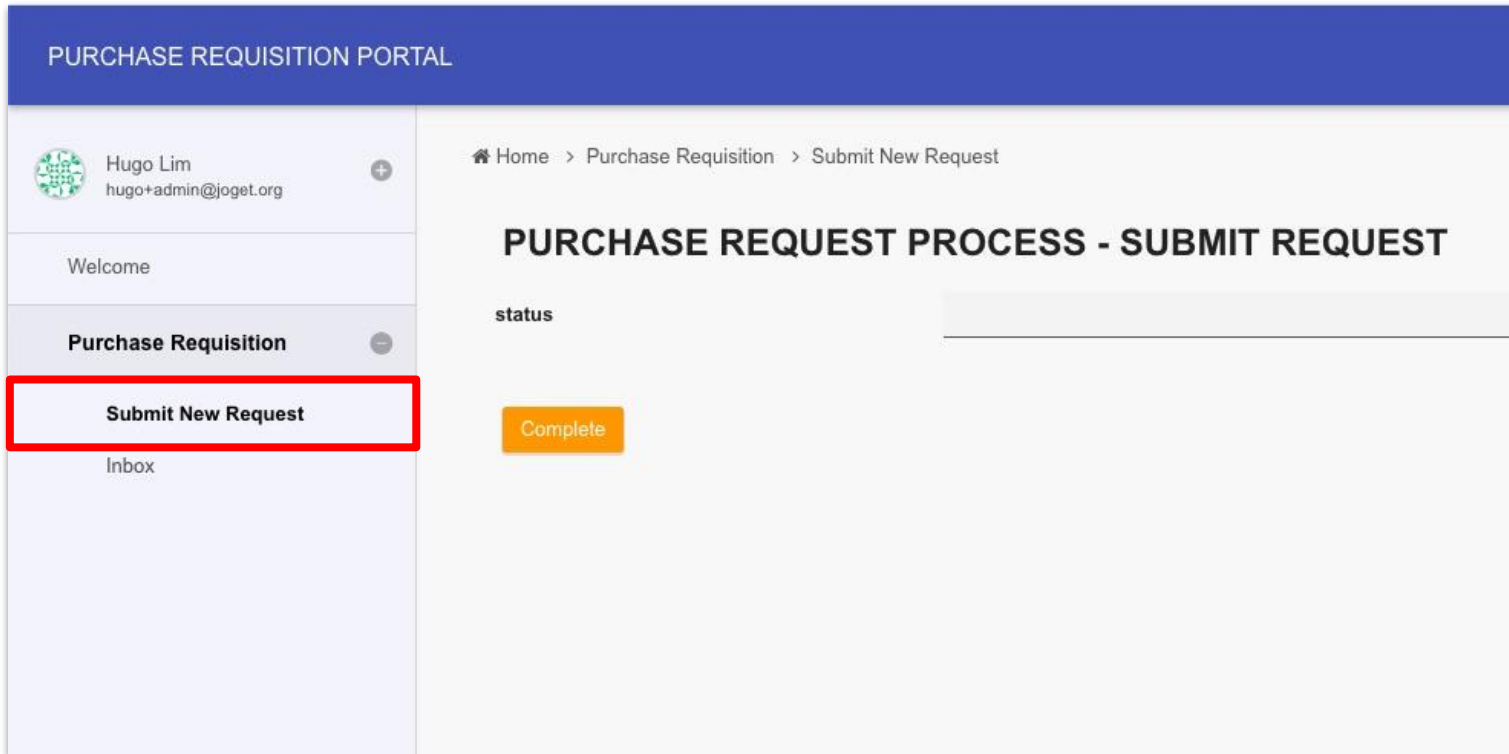
# Launching the Published App

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



# Run Through the Process

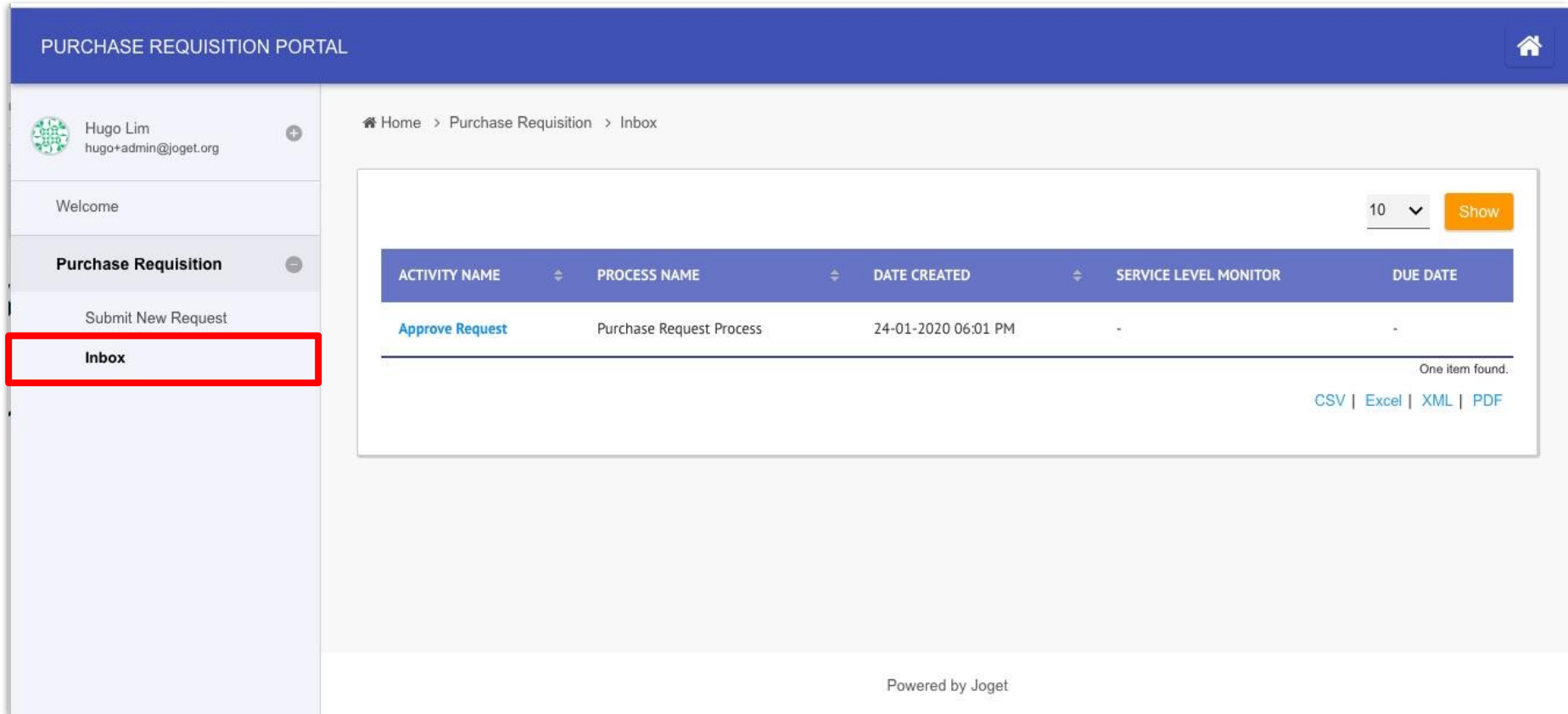
- With the Userview created, run through the entire **Purchase Request Process** again by using the userview.



The screenshot displays the 'PURCHASE REQUISITION PORTAL' interface. The left sidebar contains a user profile for Hugo Lim (hugo+admin@joget.org), a 'Welcome' message, a 'Purchase Requisition' section with a minus sign, and an 'Inbox' section. The 'Submit New Request' button is highlighted with a red border. The main content area shows a breadcrumb trail: Home > Purchase Requisition > Submit New Request. Below this, the title 'PURCHASE REQUEST PROCESS - SUBMIT REQUEST' is displayed. A 'status' label is followed by a grey input field. At the bottom, there is an orange 'Complete' button.

# Run through the Process

- Access subsequent assignment through the Inbox menu.



The screenshot displays the 'PURCHASE REQUISITION PORTAL' interface. On the left sidebar, the 'Inbox' menu item is highlighted with a red border. The main content area shows a breadcrumb trail: 'Home > Purchase Requisition > Inbox'. Below this, there is a table with one row of data. The table headers are: 'ACTIVITY NAME', 'PROCESS NAME', 'DATE CREATED', 'SERVICE LEVEL MONITOR', and 'DUE DATE'. The data row contains: 'Approve Request', 'Purchase Request Process', '24-01-2020 06:01 PM', '-', and '-'. A 'Show' button is visible in the top right of the table area, and a 'Submit New Request' button is in the sidebar. The footer of the page reads 'Powered by Joget'.

ACTIVITY NAME	PROCESS NAME	DATE CREATED	SERVICE LEVEL MONITOR	DUE DATE
<a href="#">Approve Request</a>	Purchase Request Process	24-01-2020 06:01 PM	-	-



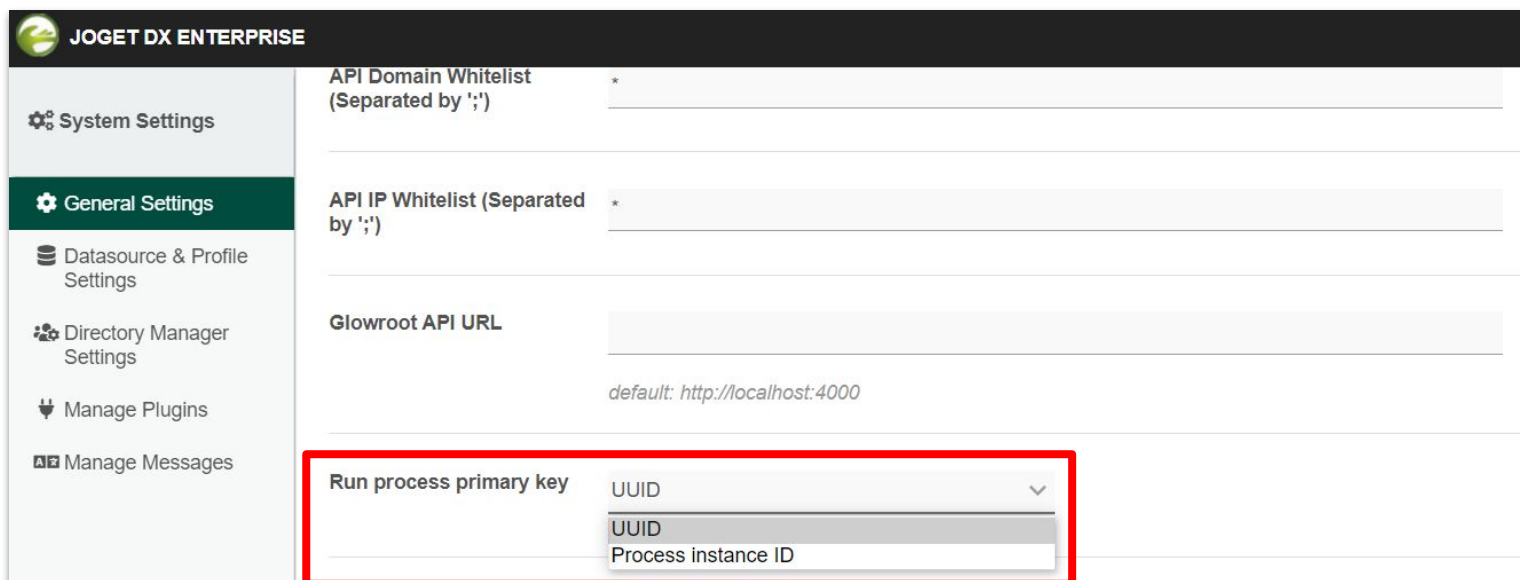
# Good To Know

---

- 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.



The screenshot shows the 'JOGET DX ENTERPRISE' Admin Bar. On the left is a navigation menu with 'General Settings' selected. The main content area shows various system settings. At the bottom, the 'Run process primary key' dropdown menu is open, with 'Process Instance ID' selected. The dropdown options are:

- UUID
- Process Instance ID

# 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?

ALL participants must be mapped, **except** "System"

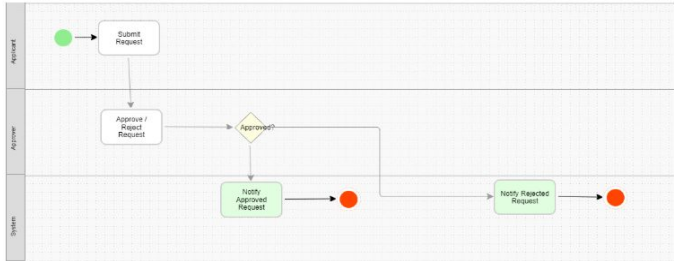
**JOGET DX ENTERPRISE**

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:



Show Additional Info

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

Search:

Applicant ID : applicant	Add/Edit Mapping
Approver ID : approver	Add/Edit Mapping
System ID : system	Add/Edit Mapping
Process Start White List ID : processStartWhiteList	Add/Edit Mapping

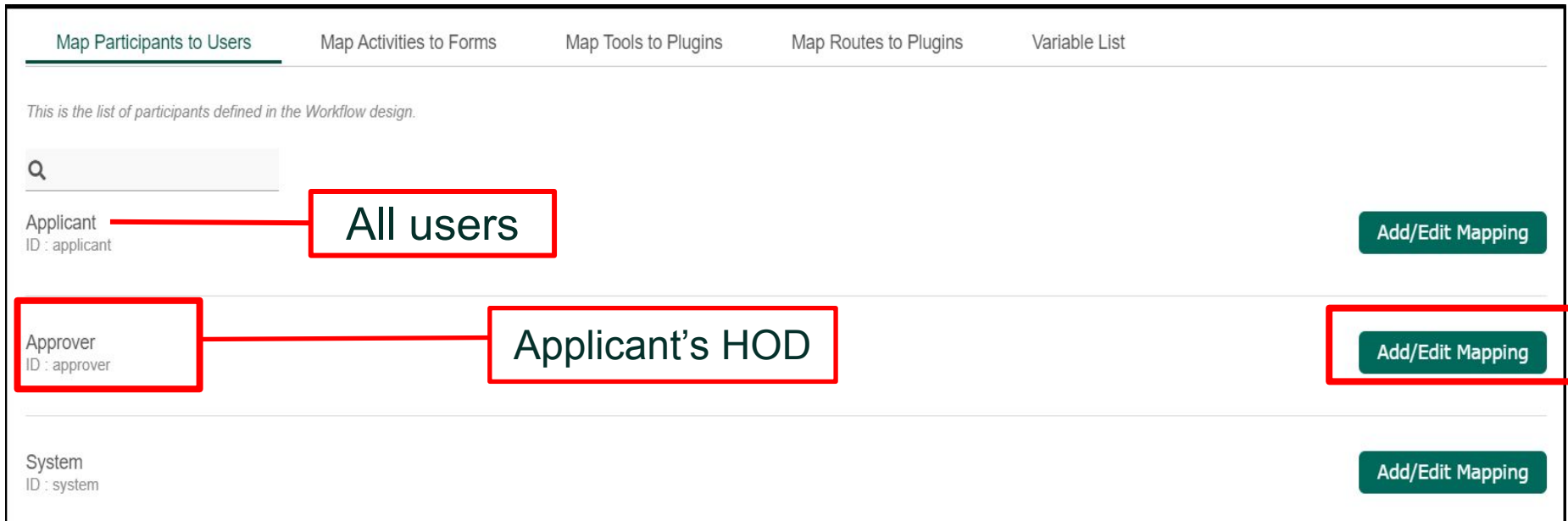
Type	Role
Value	Admin

Remove Mapping



# Participant Mapping

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



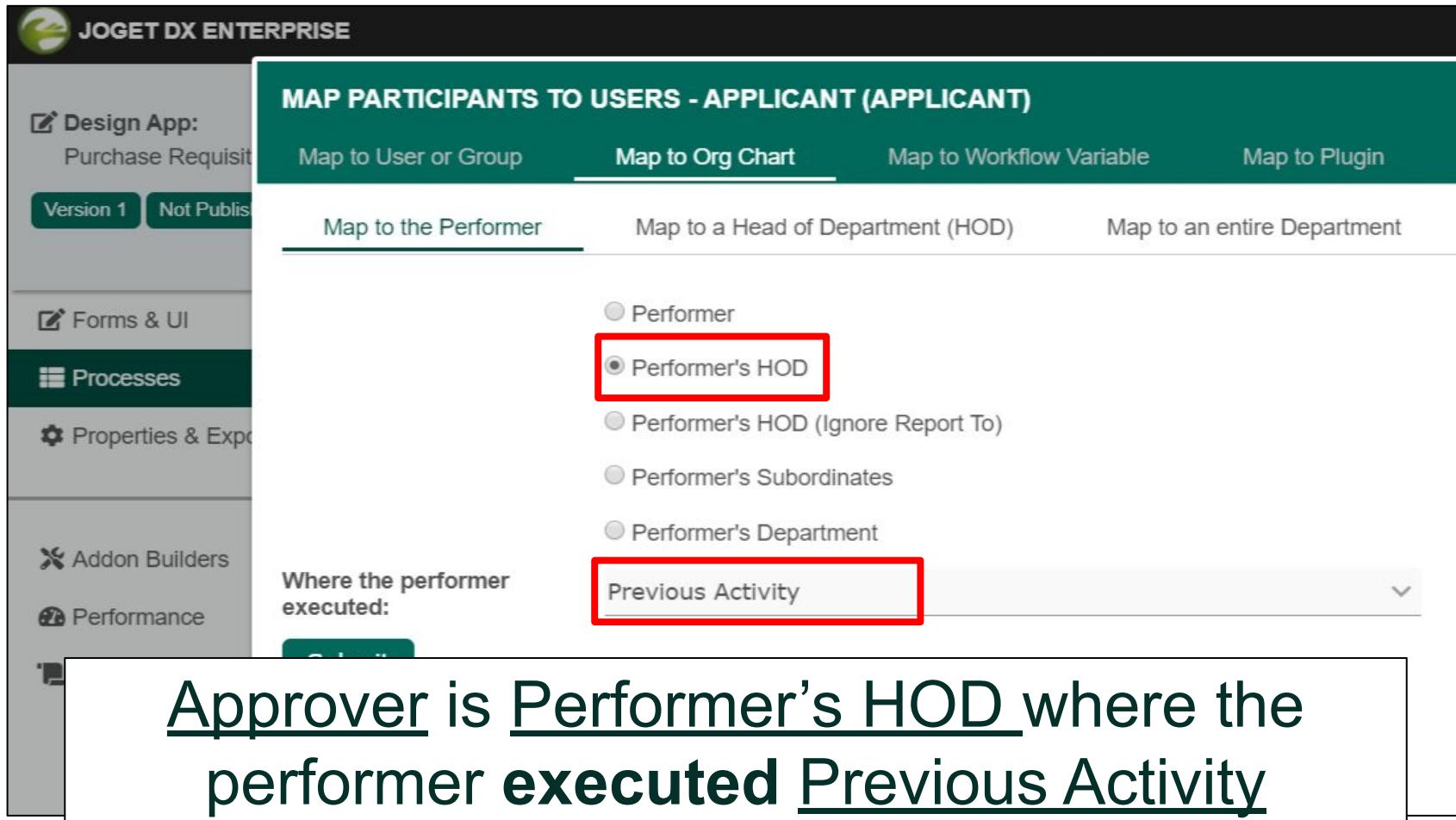
Map Participants to Users    Map Activities to Forms    Map Tools to Plugins    Map Routes to Plugins    Variable List

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

Search:

Applicant ID : applicant	All users	Add/Edit Mapping
Approver ID : approver	Applicant's HOD	Add/Edit Mapping
System ID : system		Add/Edit Mapping

# Approver: Performer's HOD



**JOGET DX ENTERPRISE**

**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.*

Q

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

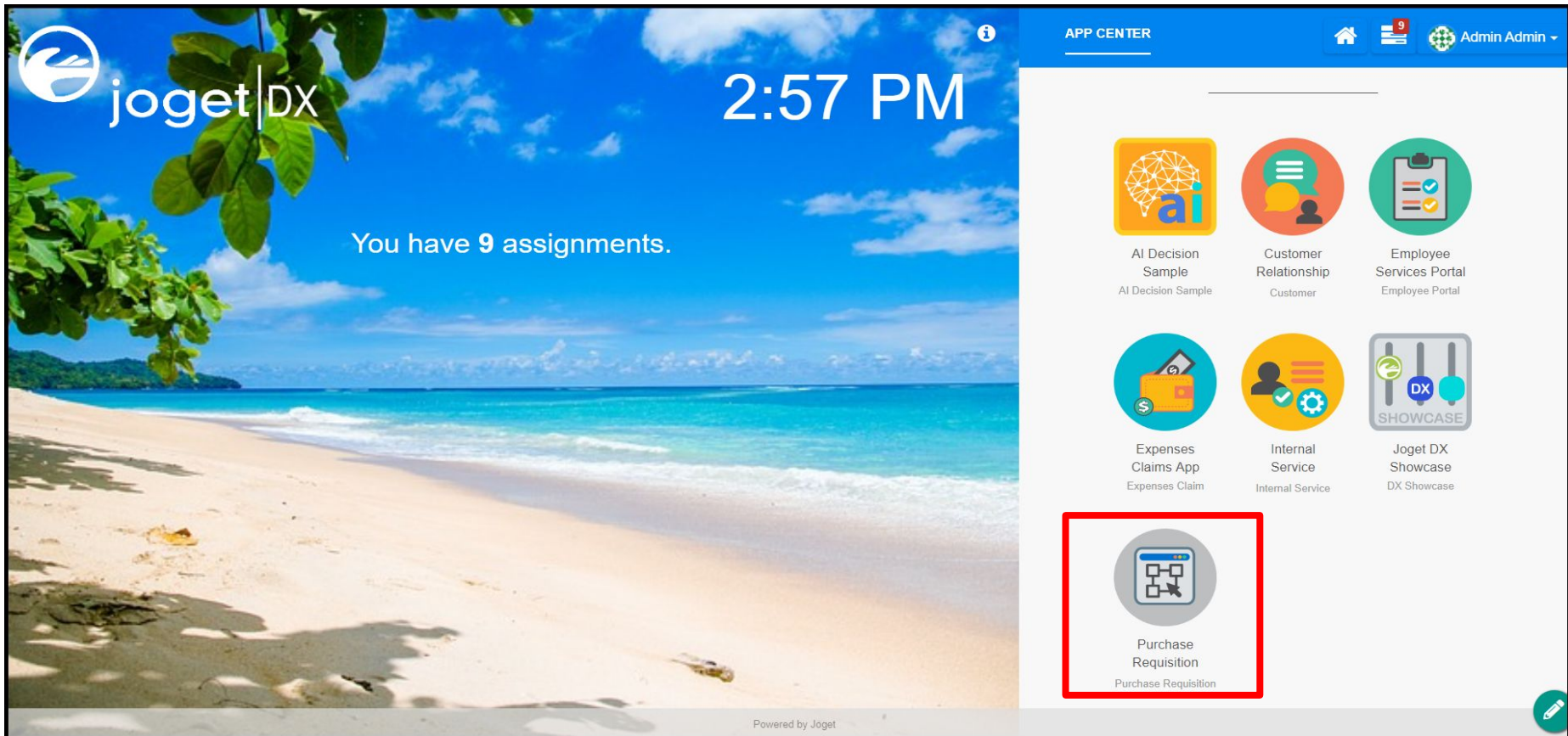
# Test Run the Process

---

- 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.

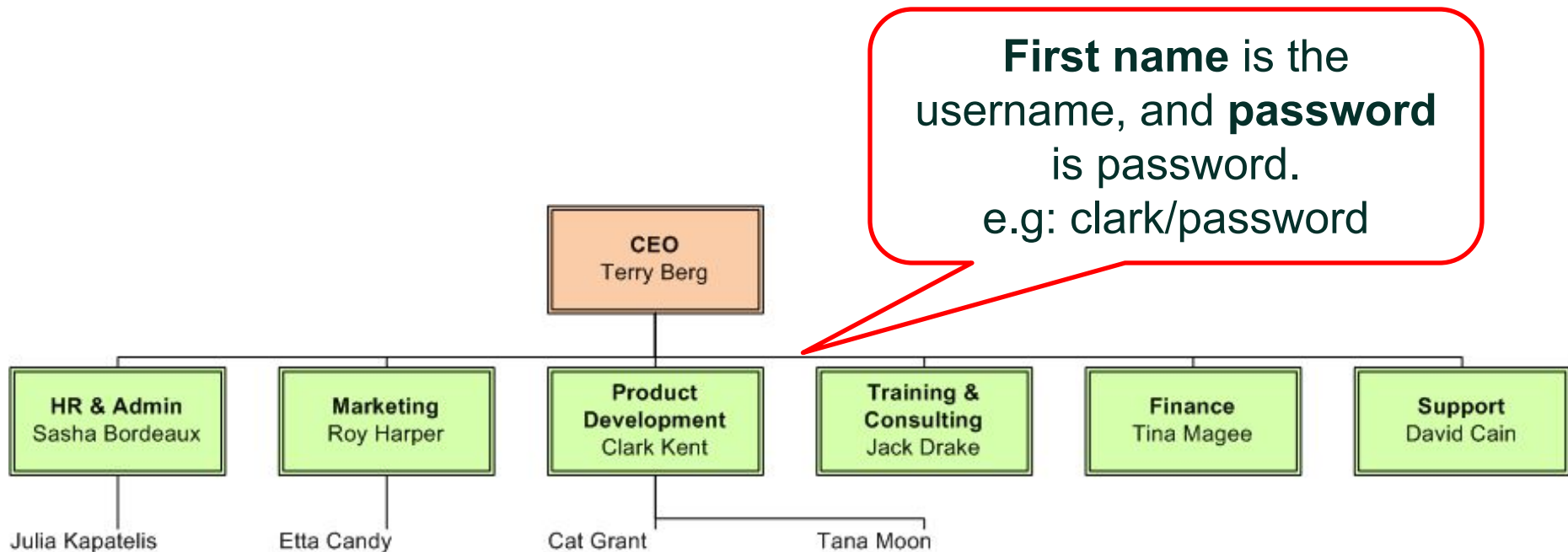


The screenshot displays the Joget DX App Center interface. On the left, a banner features the Joget DX logo, a clock showing 2:57 PM, and the text "You have 9 assignments." The right side is the "APP CENTER" dashboard, which lists several applications:

- AI Decision Sample
- Customer Relationship
- Employee Services Portal
- Expenses Claims App
- Internal Service
- Joget DX Showcase
- Purchase Requisition** (highlighted with a red box)

At the bottom of the dashboard, there is a "Powered by Joget" logo and a small green icon in the bottom right corner.

# Who Is Cat Grant's HOD?



First name is the username, and **password** is password.  
e.g: clark/password

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

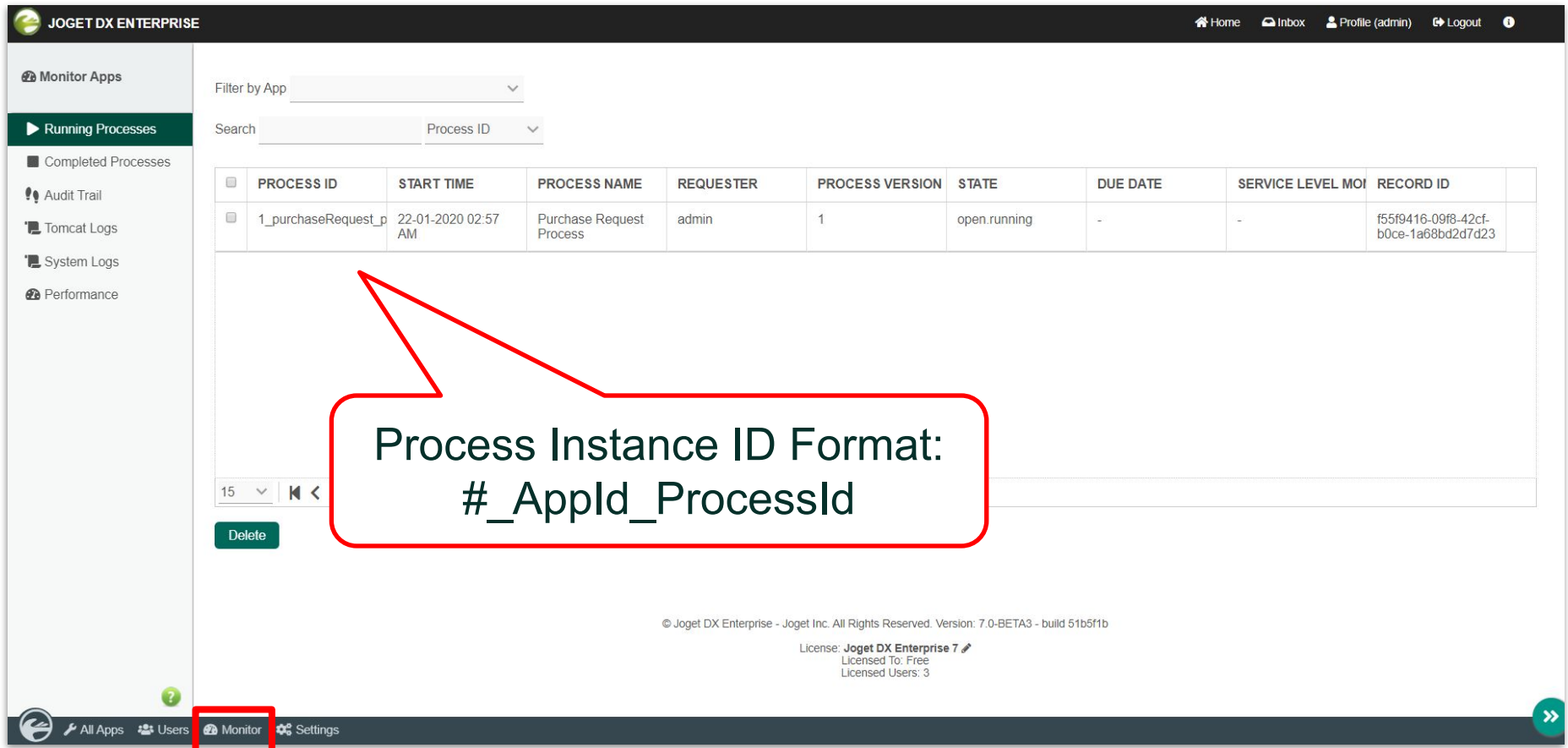
# Monitoring Processes

---

- 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.



JOGET DX ENTERPRISE

Home | Inbox | Profile (admin) | Logout

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

15 | < >

Delete

Process Instance ID Format:  
#\_AppId\_ProcessId

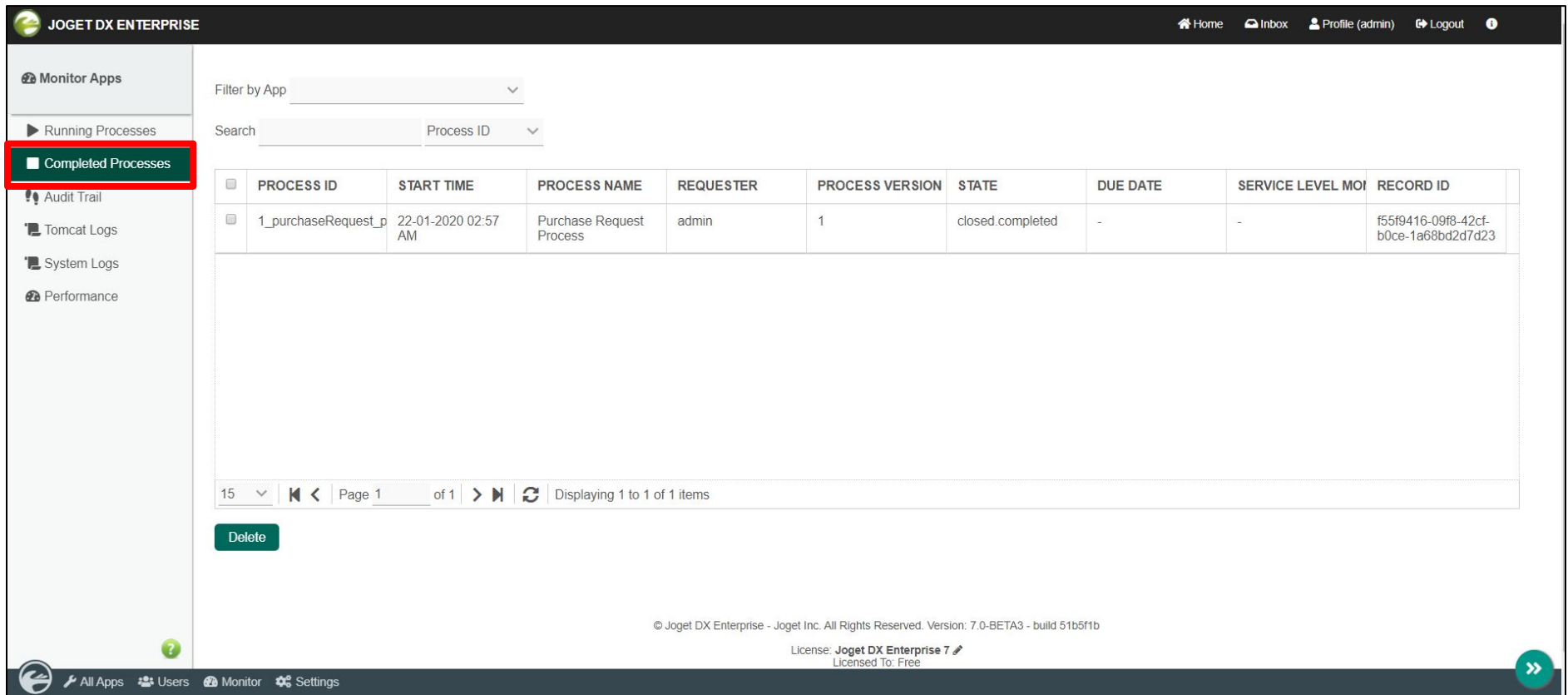
© Joget DX Enterprise - Joget Inc. All Rights Reserved. Version: 7.0-BETA3 - build 51b5f1b

License: Joget DX Enterprise 7  
Licensed To: Free  
Licensed Users: 3

All Apps | Users | Monitor | Settings

# Monitor Completed Processes

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



JOGET DX ENTERPRISE

Home | Inbox | Profile (admin) | Logout

Monitor Apps

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

Filter by App

Search  Process ID

<input type="checkbox"/>	PROCESS ID	START TIME	PROCESS NAME	REQUESTER	PROCESS VERSION	STATE	DUE DATE	SERVICE LEVEL MOI	RECORD ID
<input type="checkbox"/>	1_purchaseRequest_p	22-01-2020 02:57 AM	Purchase Request Process	admin	1	closed.completed	-	-	f55f9416-09f8-42cf-b0ce-1a68bd2d7d23

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

Delete

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

# 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
- ▶ Running Processes
- Completed Processes
- Audit Trail
- Tomcat Logs
- System Logs
- Performance

View Graph
Abort Instance
Remove Instance
Re-evaluate

**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

Time From

ACTIVITY

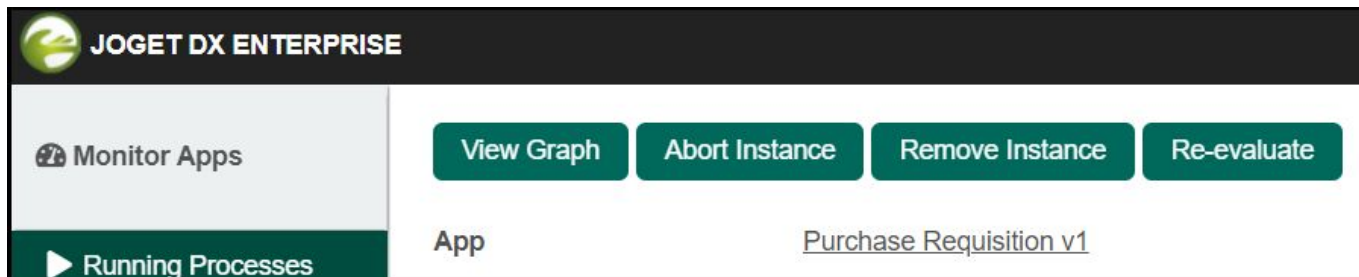
ACTIVITY ID	ACTIVITY NAME	STATE	CREATE TIME	SERVICE LEVEL MOI
913_172_purchaseRe	Approve / Reject Request	open.not_running.not	10-12-2019 01:23 AM	-
912_172_purchaseRe	Submit Request	closed.completed	10-12-2019 01:23 AM	-

Activity Instance ID format:  
#\_ProcessInstanceId\_ActivityId


Which activities are completed?  
Which activity is active now?

# 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: **ACTIVITY LIST**

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

Date limit:

Due date:

Delay:

Finish time:

From Date Created:

**VARIABLE LIST**

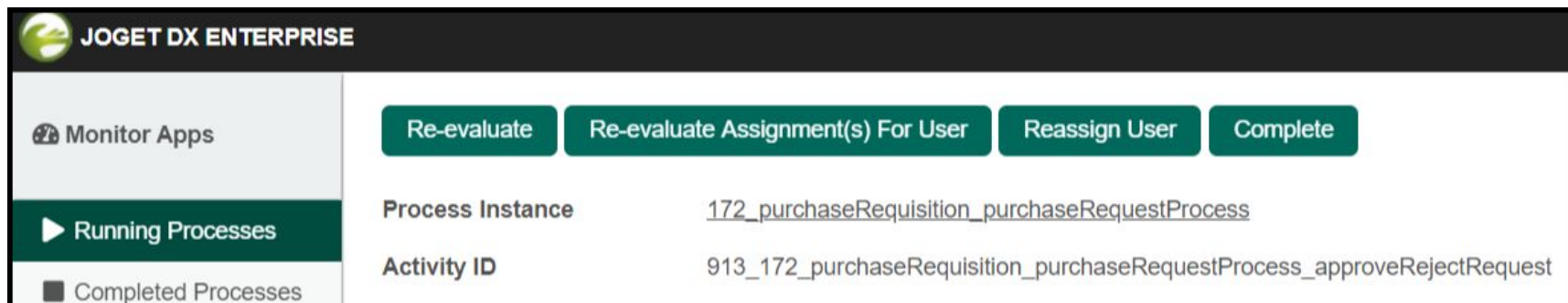
status:  Set

Workflow variable value



# 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 interface. At the top left, there is a logo and the text "JOGET DX ENTERPRISE". Below this, there is a navigation menu with three items: "Monitor Apps", "Running Processes" (which is highlighted in green), and "Completed Processes". To the right of the navigation menu, there are four green buttons: "Re-evaluate", "Re-evaluate Assignment(s) For User", "Reassign User", and "Complete". Below the buttons, there is a table with two rows of information:

Process Instance	172_purchaseRequisition_purchaseRequestProcess
Activity ID	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](https://twitter.com/jogetworkflow)
- [facebook.com/jogetworkflow](https://facebook.com/jogetworkflow)
- [youtube.com/jogetworkflow](https://youtube.com/jogetworkflow)