



## STANDARD OPERATING PROCEDURE

<b>DOC TITLE:</b>	<b>RPA BLUE PRISM DEVELOPMENT STANDARDS</b>
<b>DOC ID:</b>	<b>SOP-8721</b>
<b>VERSION:</b>	<b>7.0</b>
<b>PROCESS CATEGORY:</b>	<b>AS DEVELOPMENT</b>

## TABLE OF CONTENTS

<b>1.</b>	<b>OBJECTIVE.....</b>	<b>5</b>
<b>2.</b>	<b>SCOPE.....</b>	<b>5</b>
<b>3.</b>	<b>DEFINITIONS.....</b>	<b>5</b>
3.1	Terms and Acronyms: .....	5
3.2	Role Descriptions .....	6
<b>4.</b>	<b>REFERENCES .....</b>	<b>6</b>
<b>5.</b>	<b>OBJECT DESIGN .....</b>	<b>6</b>
5.1	Object Studio.....	6
5.1.1	Object Design Principles .....	6
5.1.2	Action Design Principles .....	6
5.1.3	Shared Application Model .....	7
5.1.4	MS Visual Basic Objects .....	8
5.1.5	Graph API Object.....	8
5.1.6	BP Capture - Use of BP Capture.....	8
5.1.7	Wait and Timeout Stages.....	9
5.1.8	Action Independency .....	10
5.1.9	Action Inputs .....	10
5.2	Standard Actions .....	10
5.2.1	Initialize.....	10
5.2.2	Clean Up.....	11
5.2.3	Attach .....	11
5.3	Application Modeling .....	11
5.3.1	Attribute Selection Standard .....	11
<b>6.</b>	<b>PROCESS STUDIO .....</b>	<b>11</b>
6.1	Process Studio Configuration .....	11
6.2	Standard Template for Processes .....	11
6.2.1	Standard Pages .....	12
6.2.2	Standard Blocks.....	12
6.3	Standard Block Color Guide .....	13
<b>7.</b>	<b>MAXIMIZING READABILITY: STAGE ORIENTATION, SPACING, SIZE, AND COLOR .....</b>	<b>13</b>

# RPA BLUE PRISM DEVELOPMENT STANDARDS

7.1	Readability Definition.....	13
7.2	Stage Flow .....	13
7.3	Stage Spacing .....	14
7.4	Stage Size.....	14
7.5	Stage Color .....	14
7.6	Stage Font.....	14
<b>8.</b>	<b>DESCRIPTIONS AND NOTE STAGES.....</b>	<b>14</b>
<b>9.</b>	<b>EXCEPTION HANDLING .....</b>	<b>15</b>
9.1	Exception Type Standard .....	15
9.2	Tagging Exceptions in the Work Queue .....	16
9.3	Incident Creation Process .....	16
<b>10.</b>	<b>DATA MANAGEMENT AND DATA PROTECTION .....</b>	<b>16</b>
10.1	Credentials Management .....	16
10.2	Stage Logging Configuration .....	16
10.2.1	Exception Handling Stages .....	16
10.2.2	All Other Stages.....	17
10.3	Interacting with Target Application Environments .....	17
10.3.1	Developing and Testing while Interacting with Production.....	17
10.3.2	Developing and Testing While Interacting with QA.....	18
10.3.3	Developing and Testing While Interacting with DEV.....	18
10.4	BP Memory Management.....	18
10.4.1	Clean Up Work Queue.....	18
10.4.2	Empty Temporary Collection Data Items.....	19
10.4.3	Force Garbage Collection .....	20
10.5	Connecting to SharePoint.....	20
<b>11.</b>	<b>ACCESS METHODS FOR LIVE (PRODUCTION) ENVIRONMENTS .....</b>	<b>20</b>
<b>12.</b>	<b>TESTING APPROACH .....</b>	<b>21</b>
12.1	Application Model Verification.....	22
12.2	Business Simulation Testing.....	22
12.3	Formal Testing in Blue Prism QA .....	23
<b>13.</b>	<b>SECURITY CONFIGURATION STANDARD .....</b>	<b>23</b>
13.1	User Access and Security Configuration .....	23
13.1.1	Single Sign On.....	23

13.1.2	Credential Manager .....	24
13.1.3	Encryption Schemes .....	25
13.1.4	Audit – Logs.....	25
13.1.5	System Settings.....	27
13.1.6	System License .....	28
13.1.7	System Archiving .....	29
13.1.8	System Scheduler.....	29
13.1.9	System Calendar .....	30
13.1.10	Workflow and Work Queues .....	31
<b>14.</b>	<b>STAGE AND FLOW VALIDATION CHECK.....</b>	<b>32</b>
<b>15.</b>	<b>THIRD PARTY TOOLS .....</b>	<b>36</b>
15.1	Blue Prism Digital Exchange (DX) VBO.....	36
15.2	Third-Party Programming Environments and Source Code .....	37
<b>16.</b>	<b>CHROME BROWSER BEST PRACTICES.....</b>	<b>37</b>
16.1	Chrome Browser Version .....	37
16.2	Download Settings .....	37
16.3	Spy Modes .....	38
16.4	Match Optimization.....	38
16.4.1	Optimization attributes.....	39
16.4.2	Browser-based application optimization .....	39
<b>17.</b>	<b>DOCUMENT HISTORY.....</b>	<b>40</b>
17.1	Superseded Document(s).....	40
17.2	Revision History .....	40

## 1. Objective

The purpose of this Standard Operating Procedure (SOP) is to define the process and requirements for activities and deliverables associated with Blue Prism Bot development. Blue Prism is a standard technology within Johnson & Johnson for Robotic Process Automation (RPA). The coding standards defined within this document are required for any development / configuration of processes in Blue Prism.

## 2. Scope

This SOP applies to RPA Bot application coding development standards to ensure that Johnson & Johnson (J&J) Service Owners, SDLC Project Managers, Scrum Masters, IT delivery teams, IT operations teams, Service Delivery Managers, and Johnson & Johnson IT vendors are aware of the objectives, activities and tasks involved in the developing Robotics Process Automation code.

## 3. Definitions

It is assumed the audience of this SOP has an understanding of basic IT terms and acronyms.

### 3.1 Terms and Acronyms:

- **SOP** – Standard Operating Procedure
- **RPA** – Robotics Process Automation
- **SDLC** – Software Development Lifecycle
- **Blue Prism**- Blue Prism is a software platform within Johnson & Johnson used for Robotic Process Automation (RPA)
- **WI** – Work Instruction
- **Bot** – The software code that executes a process (transaction or group of transactions) that replicates a human end user
- **CMDB** – Configuration Management Database
- **CI** – Configuration Item
- **Target Application** – A business application that is accessed / interacted with by Blue Prism.

### 3.2 Role Descriptions

- **Developer** – Builds bot / configures automations for the respective business unit; communicates risks and issues to the team. Also provides peer review of configuration of the automation and adherence to published guidelines, templates, and standards.
- **SME** – Provides details on the process to be automated and the new business process using the automated steps.

### 4. References

- **Doc ID:** – SOP-8722      **Title:** – RPA Application Access Management
- **Doc ID:** – WI-8732      **Title:** – RPA Target and Source Application  
Credential Management
- **Doc ID:** – N/A      **Title:** – S-26\_STND-2 Security Standard for Audit  
Logs
- **Doc ID:** WI-8726      **Title:** – RPA Blue Prism Development Naming  
Standard

### 5. Object Design

#### 5.1 Object Studio

##### 5.1.1 Object Design Principles

The object layer should be for application logic only and actions should be small, generic and re-useable. Business objects should not contain any business logic or process rules. Objects should offer a set of simple functions that a process can orchestrate into a complex sequence.

All Applications should contain a “Basic Action” object. This will contain actions for tasks such as launching the application, logging in, closing the application, and any other actions that are not screen-specific such as ‘Go Home’.

Beyond the “Basic Action” object, most other objects should be designed at the screen-level.

##### 5.1.2 Action Design Principles

Actions that read or get data should simply read the data from the screen and pass it back to the calling process. The calling process can then apply any process specific

business logic to the data. Following this approach and keeping any business logic out of the object layer enables maximum re-use of the object layer. There is no need to have a separate action for each data item to be read, all fields can be read within a single action.

Actions other than read or write should be designed with proper use of input/output data items to maximize reuse. Actions should be granular based on a specific general task.

For example, an object may be created to download a report, e.g.

Object: JJ\_Cognos\_ReportView

Action: Download Report

1. Click File
2. Click Export
3. Input Directory
4. Input File Name
5. Input File Type
6. Click Save

In this example, the Action should have the following inputs:

- Directory
- File Name
- File Type

Designed in such a way, the action for download report may be used by any process, because the process may pass the directory, file name, and file type into the object.

Actions should contain a verb as a prefix to indicate the type of action such as:

- **Navigate** – Used to navigate to another screen, e.g. Navigate to Next Page
- **Get** – Used to extract data/content from the screen, e.g. Get Form Fields
- **Set** – Used to input data/content onto the screen, e.g. Set Form Fields
- **Submit** – Used to commit / send data to the system, e.g. Submit Form

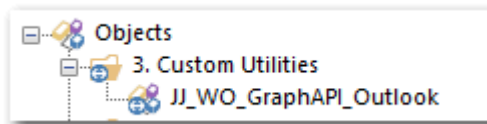
### 5.1.3 Shared Application Model

Occasionally, you may encounter applications where it is not possible to Attach to the target application, making the recommended multi-object design problematic. In these scenarios, developers should enable a shared application model on the “Basic Actions” object, so that all the elements are managed within that object and accessed by the other objects.

### 5.1.4 MS Visual Basic Objects

### 5.1.5 Graph API Object

Wrapper object is created on top of Standard Graph API object and can be located at below path in BluePrism Studio.



Standard Object for Graph API is located below path in BluePrism Studio.



Thoroughly evaluate the Graph API for performing Outlook-related actions.

- If the Graph API does not meet the requirements, consider utilizing the standard MS Outlook VBO - Standard.
- Utilize the Wrapper Object 'JJ\_WO\_GraphAPI\_Outlook' consistently.
  - Should use the actions available in Wrapper object: Authenticate, Send Email and Expiry Alert.  
*Note: New actions will be added in wrapper object based on requirement.*
  - If the required action is not available in Wrapper object, then only make the use of standard object 'Microsoft 365 - Outlook - Standard'.
  - Should not create any distinct extended versions of Graph API objects.
- Ensure that all actions are applied at the process level and are not invoked within other objects

### 5.1.6 BP Capture - Use of BP Capture

If an automation process use case is provided by a user or business team through a .json file created using BP Capture, then developer must follow these steps:

- Import the .json file into BP Capture.
- Evaluate the captured steps against the requirements defined in the FRA document.
- If everything is appropriate, rename the objects in BP Capture according to the JJT naming standard (*refer to WI-8726*).
  - Before creating a new object, ensure there is no existing object in BP Studio for the same source/target application performing the same actions.



- If an object doesn't exist, create a standard object for the source/target application for the actions that can be reused across the RPA platform.
- If an object does exist, check if it includes all the actions you need. If not, create a customized object for your specific process under Custom objects.

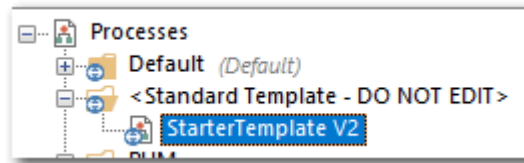
*Note: Detailed information on object creation and naming standard, please refer to WI-8726*

- Refer to the BP Capture settings required before exporting any object or process as a '.bprelease' file.

BP Capture settings: [Confluence Link](#)> SME User Guide

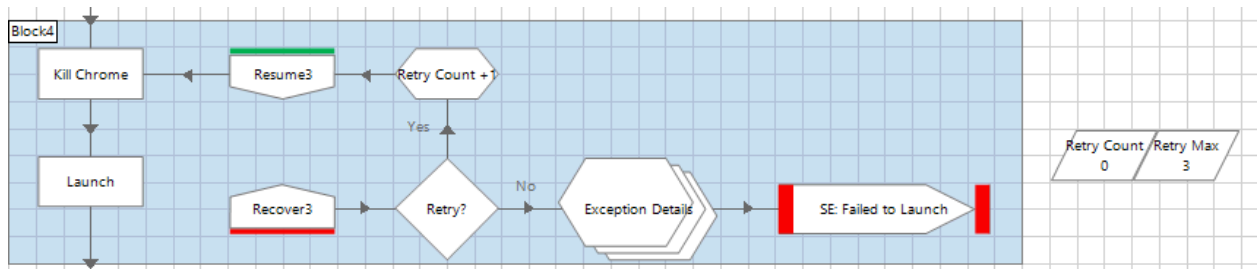
- Export the BP process and object from BP Capture in '.bprelease' format.
- Import the BP process (. bprelease) and BP object (. bprelease) files into Blue Prism Studio.
- Use this BP process skeleton (created using BP capture blank template) and design/convert it as per the process use case in StarterTemplate format.

Refer to the screenshot below to find the StarterTemplate.



### 5.1.7 Wait and Timeout Stages

- Actions should start with an Attach stage and then a Wait Stage to confirm that the process is on the correct path.
- Actions must throw an exception at timeout and not try and recover within the action
- Wait Stages must be used between read, write and send keys stages.
- Recovery logic must always reside with the process
  - Retry logic blocks are used to retry a stage if it failed the first time.
  - For example, if the bot is trying to Launch an application, but the application failed to launch, the Bot would hit the Recover stage, then check if Retry Count is less than Retry Max, which is set to 3, add 1 to Retry Count, then the Resume stage and then try again
  - The Retry Max data item is set to 3, so if the Bot fails 3 times, it will throw a System Exception and Terminate.
  - All Retry Logic blocks must be definitive.



- Global Variable should be used for the “Timeout” parameters
- Wait Stages should utilize a dynamic wait where possible

### 5.1.8 Action Independency

- Actions must not be called within another action. This includes utilities such as Microsoft (MS) Excel VBO operations, which must be used in process level and not called upon within other objects
- Actions must stand on their own and be independent of other actions
- Actions must only be linked together through the Process

### 5.1.9 Action Inputs

- Developer must not ‘hard code’ what should be Process Inputs within an Action

*For example, a validation step that checks if the value of a field is greater than 'x', should not have 'x' defined within the action. The process should define the value of 'x' and pass it to the Action as an input. This enables other process to use the action with a different 'x' value*

## 5.2 Standard Actions

Every object should have a standard set of actions below at minimum.

- Initialize – Specific details below
- Clean Up – Specific details below
- Attach – Specific details below
- Detach

### 5.2.1 Initialize

- Initialization page must be used for global variables
- Wait stages should use variables and should not be hard coded

### **5.2.2 Clean Up**

- The Clean Up page should call the Reset Global Variables page where required variables should be reset to initial value

### **5.2.3 Attach**

- Parameters for Attach page must not be hard-coded and must be passed to a Start parameter
- Flag should be maintained to check the status on connection
- Attach action may not be required if the action is using a shared application model.
- An exception must be thrown when an object is unable to Attach to the Application
- The Attach action should not be published. Other actions should call the Attach Action.

## **5.3 Application Modeling**

### **5.3.1 Attribute Selection Standard**

- Use only attributes which are needed to make the element unique
- Blank attributes must not be selected as criteria
- Priority for spying should be given to Win32, Accessibility, Region for Windows Applications
- Priority for spying should be given to HTML, Win32, AA, Region for Web Applications
- When using region mode developers must make sure that all runtime resources have the same screen resolution

## **6. Process Studio**

### **6.1 Process Studio Configuration**

- When re-using a stage (copy-paste), you should change the name of the stage, such as adding a suffix. This makes debugging easier. Example: Stage, Stage (1), Stage (2), and so on.
- Global Data items should be used only under following circumstances:
  - The data item (variable) is used across 3+ pages
  - The data item is 'read-only' and will not be changed by the process
- Global Data items should be placed on the Main Page or on a "Global Data Items" page that is dedicated to them

### **6.2 Standard Template for Processes**

Every Process should be based on the Standard Templates which are available in the Blue Prism Development environment. Any deviation from the Standard Templates are

reviewed as part of Code Review, and processes may be rejected from moving to QA/PROD if there is not a reasonable rationale for deviating from the Standard Templates.

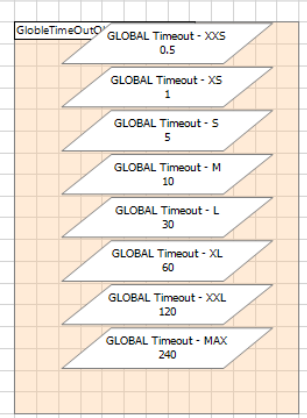
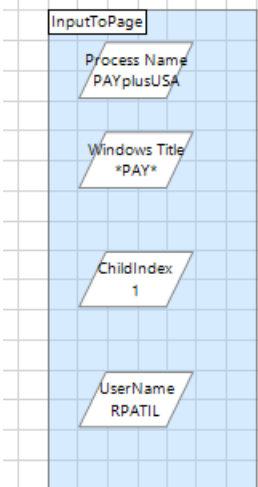
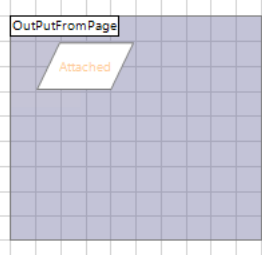
### **6.2.1 Standard Pages**

Each process should have a standard set of pages as follows:

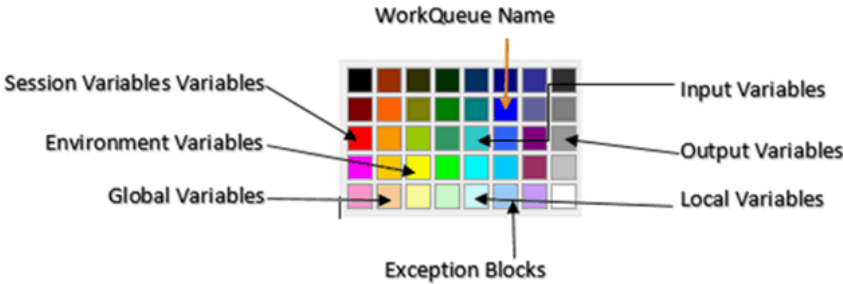
- Main Page
- Clean Up
- Start Up
- Populate Queue
- Close Down
- Mark Item Complete
- Mark Item Exception
- Global Variables
- Reset Global Variables

### **6.2.2 Standard Blocks**

Variables should be organized in different blocks based on the type e.g. Local variable, global variables, environment variables, and so on. Examples are below.

Global Variable	Input Variable	Output Variable
		

6.3 Standard Block Color Guide



7. Maximizing Readability: Stage Orientation, Spacing, Size, and Color

Developers must maximize readability.

7.1 Readability Definition

Readability is the ease with which a reader can understand the flow and logic within a process/object. In Blue Prism, the readability depends on its content (the complexity of the process/object) and its presentation (such as stage orientation, spacing, and size).

7.2 Stage Flow

Flow of stages should generally be arranged in a top-to-bottom vertical sequence, in such a way that the process/object reads from top to bottom. The exception to this is

looping/decision and other branching logic that may require various orientations to maximize readability.

### 7.3 Stage Spacing

Spacing of stages should be consistent within a process/object. This improves readability and understanding of the process. For example, stages should not have wide spacing on one page, and narrow spacing on the next page; spacing needs to be consistent and optimized for readability.

### 7.4 Stage Size

Size of stages should be consistent within a process/object. For instance, data items consistently sized as 2 cell height x 10 cell width. Again, this is to maximize readability and consistency/symmetry within a process/object.

### 7.5 Stage Color

Use of color in any stage must be purposeful and not arbitrary or excessive. For example, use of different block colors to indicate the purpose of the block, as shown in Standard Blocks Section.

### 7.6 Stage Font

Use of different fonts should be minimized as consistency in font lends to better readability. Additionally, management of fonts is highly manual in Blue Prism and applying a custom font makes the process/object more time consuming to maintain.

## 8. Descriptions and Note Stages

A purposeful and reasonably detailed description should be provided in **every** place where a description field is provided within the software. This includes but is not limited to:

- Process (including version comments)
- Object (including version comments)
- Stages
- Stage
- Work Queue
- Credential
- Release
- Release Package

Notes stages should be used to explain complex logic and be linked directly inside the flow of objects/processes. Notes should be clearly stated and able to be understood by process SMEs, not just the developer.

## **9. Exception Handling**

### **9.1 Exception Type Standard**

Developers must use only the following exception types

- Planned Business Exception
  - Naming Standard: PBE-<Exception Name>q
  - Occurs when cases/transactions cannot be processed due to some problem or business rule that makes it impossible for the case to be worked but the business has planned for an exception like this to occur, adding a human in the loop stage.
- Business Exception
  - Naming Standard: BE-<Exception Name>
  - Occurs when cases/transactions cannot be processed due to some problem or business rule that makes it impossible for the case to be worked. For instance, an exception may be used to exclude some types of transactions from processing
- System Login Exception
  - Naming Standard: IL-<Exception Name>
  - Occurs when there is a failed attempt to login to a system
- System Unavailable Exception
  - Naming Standard: SU-<Exception Name>
  - Occurs when the system is unavailable
- System Exception
  - Naming Standard: SE-<Exception Name>
- System Exception Try Once
  - Naming Standard: SETO-<Exception Name>
  - Same as system exception but should only be thrown once and not recovered.

## 9.2 Tagging Exceptions in the Work Queue

Exception items must be tagged in the Work Queue as “System Exception” or “Business Exception” depending upon the exception type. This helps in generating the Work Queue report.

## 9.3 Incident Creation Process

IRIS Incident Creation Process – This process facilitates creation of an incident whenever the automation stops unexpectedly (ends with terminated status). Upon failure, an incident will be created in IRIS which will then be assigned to the support team. For details on how to use this process please refer to User Guide Incident Creation Process.

Below please find the path for User Guide Incident Creation Process.

Robotic Process Automation SharePoint>Templates>3. Build>User Guide Incident Creation Process.pdf

Please note “IRIS Incident Creation Process” should not be modified. The Developer may utilize it in their process by supplying required input parameters and utilize the output parameter ‘Incident Number’ to notify the support team.

## 10. Data Management and Data Protection

Developer must follow these practices to conform with data protection standards.

### 10.1 Credentials Management

- Use of credential manager is required for all ID/Passwords in QA and Production environments. Use of Credential Manager must be in accordance with “RPA Target and Source Application Credential Management”.
- BOT credentials must not be saved in any data item / environmental variable, in any environment.
- Access rights must be granted in compliance with the Minimally Viable Security (MVS) “Access and Security Configuration Standards”

### 10.2 Stage Logging Configuration

#### 10.2.1 Exception Handling Stages

- Stage logging may be Enabled for Exception Handling Stages that are used to (1) Pass exception detail to the queue, or (2) Communicate exception information to business users for further action (e.g. email message).



- Exception Detail must not contain any restricted, GxP, or highly restricted data
- When logging is enabled on Actions, the checkbox "Don't log parameters on this stage" must be checked. This prevents sensitive data from being passed to the Queue.

### **10.2.2 All Other Stages**

- Stage logging must be set to Disabled or Errors Only for all other stages
- When Actions are set to Errors Only, the checkbox "Don't log parameters on this stage" must be checked. This prevents sensitive data from being passed to the Queue.

## **10.3 Interacting with Target Application Environments**

Where possible, target application owners should remediate version mismatches between environments to reduce risk, and not add complexity to development/testing processes. Applications should be assessed prior to the beginning of an automation project, and any remediation work on the target application should be conducted prior to the start of RPA development.

An assessment of applications prior to the start of development must be performed, which includes assessing the access method, differences in the environments, and the release schedule for the environments. The assessment will aid the decision on which application environments will be interacted with during development and testing.

The application assessment must be reviewed in conjunction with the Compliance Analysis, which focuses on the process to be automated. The GxP Compliance Risk and Data Risk must be considered when determining which environments will be accessed during development and testing.

While access approach for target applications may vary by application and process, Blue Prism code (release packages) must always move per the release management standard; through the Blue Prism DEV-QA-PROD flow.

### **10.3.1 Developing and Testing while Interacting with Production**

- Development and Testing may begin by interacting with the Target Application Production Environment when the process is low risk, such as not transactional and read-only and GxP Compliance Risk is 'N/A'.
- When there are differences in QA and PROD for the target application, it may be required to also model the QA environment. This would enable testing in a timely manner with no interruption to the running of the process.

### 10.3.2 Developing and Testing While Interacting with QA

- Development and Testing may begin by interacting with the Target Application QA Environment when the development activities do not interfere with the target application testing / validation activities being performed elsewhere in the environment.

### 10.3.3 Developing and Testing While Interacting with DEV

- Development and Testing for high risk processes should begin by interacting with the Target Application Development Environment. Rationale for beginning in a higher environment must be provided in the associated documentation (i.e. change control, Operation Run Book, etc.).

## 10.4 BP Memory Management

### 10.4.1 Clean Up Work Queue

- Over time, items in the Work Queue start to build up which could cause memory issues and slowness to the automation.
- In the standard Clean Up page of each Process should use the Delete Processed Items action in the Work Queues Business Object.

**Action Properties**

Name:

Description:

Business Object:

Action:

Inputs Outputs Conditions

Name	Data Type	Value
Queue Name	Text	
Date Threshold	DateTime	
Delete From All Queues	Flag	

Group: ☐ Page ☒ Data Type

☐ View All Items

Binaries: Collections, Dates, DateTimes, Flags, Images, Numbers, Passwords, Text, Times, TimeSpans

Stage logging:  ☐ Don't log parameters on this stage

Warning threshold:  Number of minutes  (0 to disable)

OK Cancel

Parameter	Direction	Data Type	Description
Queue Name	In	Text	The name of the queue to delete items from
Date Threshold	In	DateTime	The threshold date. Items completed or marked with an exception before this date will be deleted from the queue - if not given, all processed items in the specified queue will be deleted. JnJ best practice is 30 days.
Delete from all Queues	In	Flag	Indicates whether to delete items from all queues. If this flag is set to true you cannot specify a queue to delete from. This value defaults to false if nothing is entered.
Deletion Count	Out	Number	Indicates the number of items which were deleted (which may legitimately be zero).

### 10.4.2 Empty Temporary Collection Data Items

- Most often memory issues can be resolved by changing the design of your Processes or Objects so that they are more efficient, or so that they avoid loading large amounts of contiguous data into memory and then process this data in one Object or Process.
- The following activities are known to be memory-intensive operations for Collection Data items:
  - Loading large data sets into Collection
  - Filtering large Collections into other Collections
  - Passing large Collections between pages on Object or Process
  - Not emptying Collections when the Process or Object no longer requires them to exist (e.g. Remove Null Rows, Remove Empty Rows or Remove All Rows)
- To reclaim memory, the data used by an application needs to be marked as available for garbage collection. A Data Item which is storing data can be reset (the data removed) and the memory space which that data formerly occupied will be marked as available for garbage collection.
- For example, a Collection data item would have all its rows removed to clear out the existing data. This could be achieved using the internal "Collections" VBO with its Action of "Remove All Rows".

#### Action Properties

Name:
Empty Redundant Collection

Description:

Business Object
Collections


Action
Remove All Rows

Inputs
Outputs
Conditions

Name	Data Type	Value
Collection Name	Text	MyLargeCollection

### 10.4.3 Force Garbage Collection

- The concept of “garbage collection” is that the operating system will periodically (during “free time”) reclaim memory which has been designated as available to be recovered – that is, memory which is no longer being used to store information required by running applications.
- Blue Prism retains data which has been created within it until the “automate.exe” process is closed.
- One useful function relating to garbage collection is the GC.Collect() function. This requests that the operating system recover memory marked for collection as soon as possible.
- The Garbage Collection VBO is required to be called:
  - 1. After calling any sub process
  - 2. After deleting any big data collection

 Action Properties

Name:	Utility - Garbage Collector::GC Collect
Description:	
Business Object	Utility - Garbage Collector
Action	GC Collect

### 10.5 Connecting to SharePoint

- Using WebDav (connecting SharePoint location to Windows drive) is not an acceptable method to connect to SharePoint
- Process should connect to SharePoint using one of the two following methods:
  - Using the SharePoint API
  - OLEDB
  - SharePoint sync via OneDrive
  - Blue Prism DX Skills (3)

## 11. Access Methods for Live (Production) Environments

When live environment must be accessed by developer, the following approaches should be assessed and are required to be performed in this order. Regardless of the approach, the SME should always maintain oversight of actions performed in the system.

1. SME logs into the live environment using their own credentials.

- a. This can take place in person or via screenshare where SME keys in their credentials on the developer's machine or takes control of developer's desktop.
  - b. SME credentials must not be shared with the developers, and the SME must always maintain oversight of actions performed in the system.
  - c. This process is not possible with trusted SSO.
2. Developer gains temporary access to the application as an end user.
  - a. This temporary access should be minimized to only the time required to complete the work.
  - b. Effort should be taken to minimize training requirements required to access the target application.
3. Developer will use the Runtime Machine to access live system.
  - a. This temporary access must be approved for the appropriate role and assigned only for the time required to complete the work.
  - b. Gaining access to the runtime machine shall be facilitated by the platform administration team in such a way that the credential for the resource is not shared.
  - c. The activities to be performed by the developer under the Virtual Identity shall be only to those directly related to fixing/addressing issues pertaining to the live environment.

## 12. Testing Approach

The image below provides an overview of the testing approach during development and testing phases. These approaches should be considered for each project.

Phase	Develop			Test		
Activity	Application Model Verification	Unit Testing	Business Simulation	Configuration Testing	System Testing	UAT Testing
Description	Confirms the application model is valid across environments	Testing of objects and processes individually	Testing small snippets of the process on the Live System without Committing Data	Testing of the end-to-end process through the studio and control room.	Testing of the end-to-end process on live scenarios using Studio, then Control Room	End to end testing of the completed solution by the Tester and SME
Script Driven?	No	No	No	Yes	Yes	Yes
When Used?	While developing the compliance plan	While developing a user story	Before entering the formal testing phase	After deployment to Blue Prism QA	Once Configuration Testing is approved	Once System Testing is approved
When Applicable	Always	Always	When there are many environment differences	Always	Always	Always
Blue Prism Environment	DEV	DEV	DEV	QA	QA	QA
Application Environment	TEST then LIVE	TEST	TEST then LIVE	TEST	LIVE	LIVE
Blue Prism Module	Studio	Studio Control Room	Studio Control Room	Control Room	Control Room	Control Room
Runtime Resource	Developer	Developer	Developer	Tester Bot	Tester Bot	Bot
Business / SME Needed?	Yes	No	Yes	Yes	Yes	Yes
Performing Role	Developer Business SME	Developer	Developer Business SME	Tester Business SME	Tester Business SME	Tester Business SME

## 12.1 Application Model Verification

Application Model Verification will catch environment differences early in the development cycle. The verification should be executed within Blue Prism Development environment and in the presence of the process SME/ end user. Use of Blue Prism functionality is limited to only the Application Modeler. This process is only applicable during the development phase when a non-production Target Application environment will be used for development and testing.

### Steps of verification (Complete in specified order):

1. Developer validates Application Model in Application Development or QA environment (depending on which is targeted in Blue Prism DEV)
  - I. Acceptance Criteria: All elements in the Application Model can be recognized/highlighted
2. Developer and SME log into Application Production environment and validate Application Model on Live / Production Environment
  - I. This may involve processing at least one real case in the system, committing at least one real transaction
  - II. Acceptance Criteria: All elements in the Application Model can be recognized / highlighted

## 12.2 Business Simulation Testing

Business Simulation Testing allows early testing of process steps on live data / environments through the creation and running of a Component. A Component is a small process build specifically for testing a portion of the overall solution.

An example Component is a small Blue Prism process that contains multiple completed Object Studio actions linked together to read and write actions to complete a form.

Components should not include any data commit or data creation actions. They should include exception handling to retry if there is an unexpected system exception, depending on the complexity of the process.

Business Simulation Testing should be executed within Blue Prism DEV environment in the presence of the process SME / end user.

### Steps (Complete in specified order):

1. From Blue Prism DEV, developer runs the Component in Target Application Test Environment (if available) in Process Studio with Breakpoints as needed
  - Acceptance Criteria: Component runs successfully, and no creation/alteration of data is made

2. From Blue Prism DEV, developer runs the Component in Target Application Test Environment (if available) in Control Room
  - Acceptance Criteria: Component runs multiple cases successfully, exception logic operates as expected, Component returns process to a 'known' or 'expected' location so that process can continue
3. From BP DEV, developer runs Steps #1 and Steps #2 in the Application Live Environment

### 12.3 Formal Testing in Blue Prism QA

The objective of formal testing is to validate and capture evidence that the process will work as expected in production. Testing is performed in Test environment, then introduced to live (production) data in a controlled manner

**From Blue Prism QA perform (Complete in specified order):**

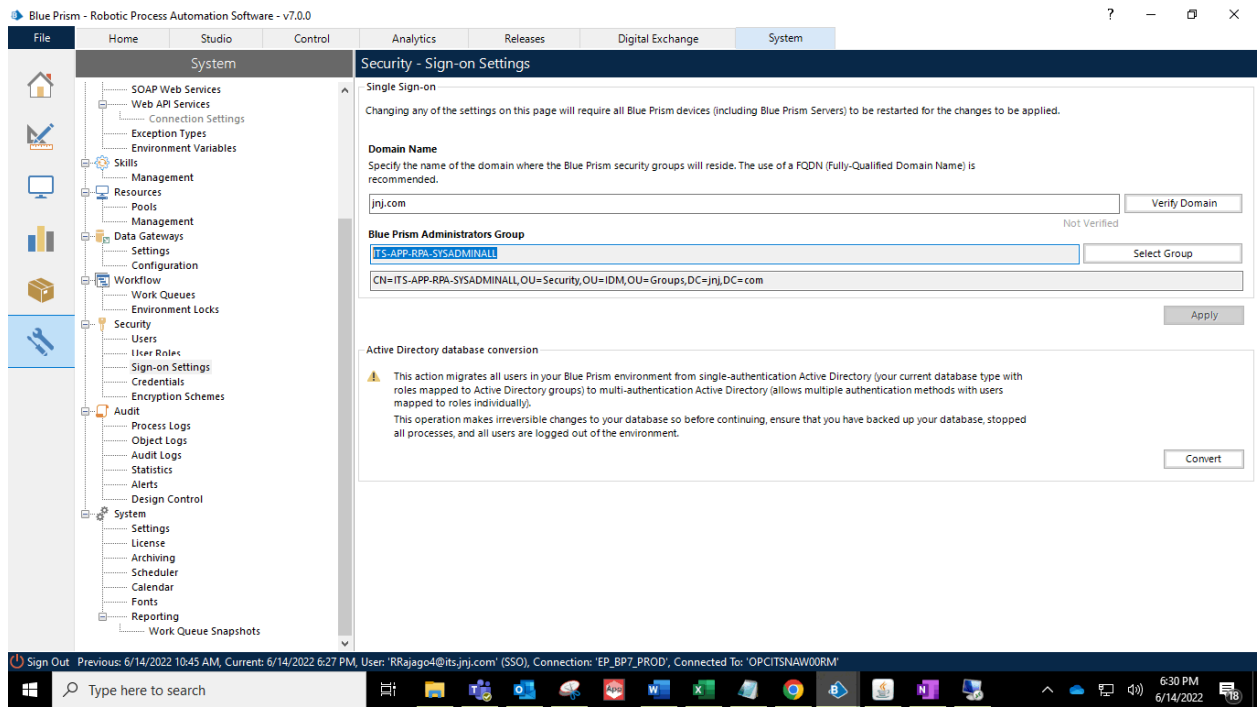
1. Configuration Testing (Test Application Environment)
  - End to end test scenarios
  - Test starts in Process Studio, then graduates to Control Room
2. Verification Testing (Live Application Environment)
  - End-to-end live scenarios
  - Test starts in Process Studio, then graduates to Control Room
3. UAT Testing (Live Application Environment)
  - Full solution including scheduling
  - Control room only
  - Limited number of cases processed at a time and reviewed by SME

## 13. Security Configuration Standard

### 13.1 User Access and Security Configuration

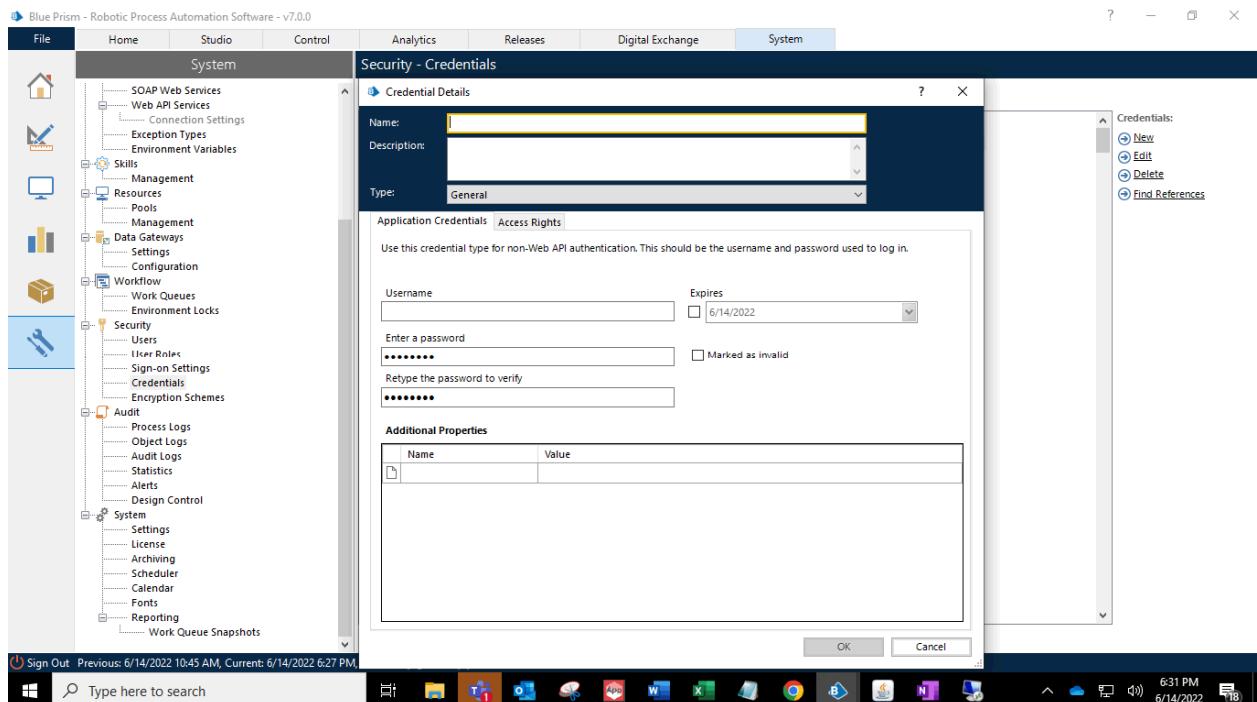
#### 13.1.1 Single Sign On

- Define Domain Name (where Blue Prism Security Groups will reside) = jnj.com
- Enterprise Blue Prism Administrators Group = ITS-APP-RPA-SYSADMINALL  
AMS Blue Prism Administrators Group = ITS-APP-AMS-RPA-SYSADMINALL



## 13.1.2 Credential Manager

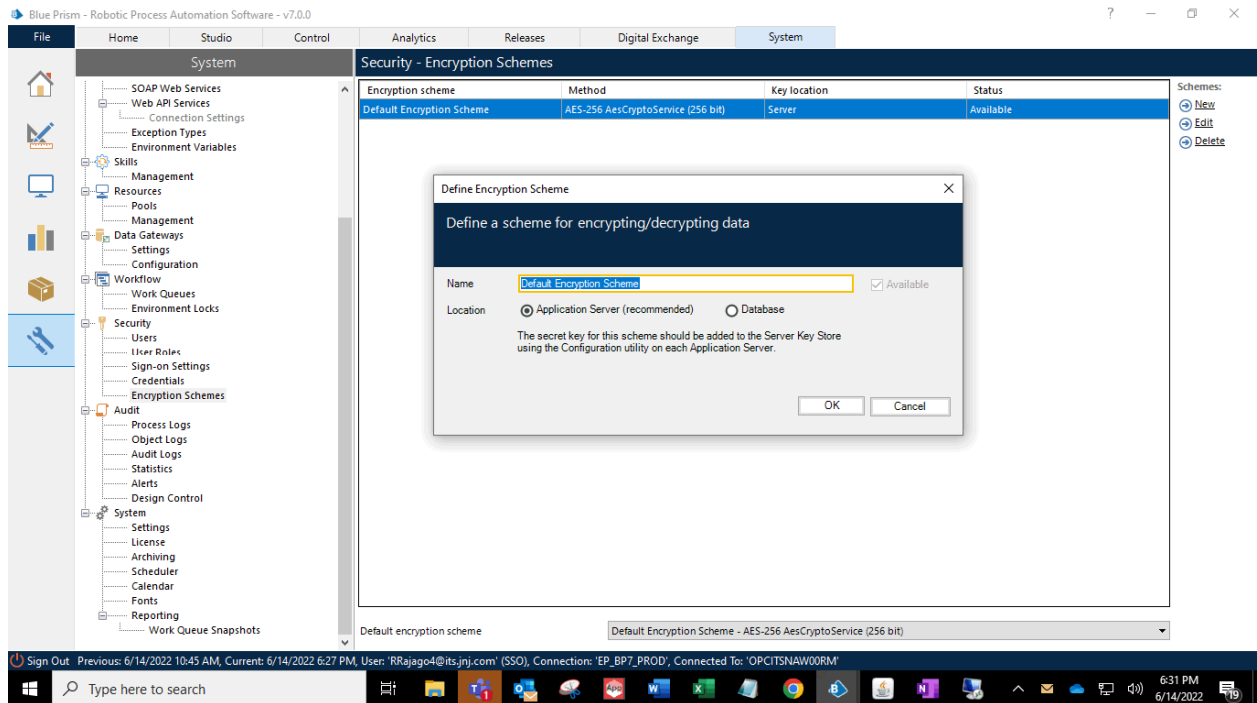
Use of credential manager is required for all ID/Passwords in QA and Production environments. Use of Credential Manager must be in accordance with “RPA Target and Source Application Credential Management”.





### 13.1.3 Encryption Schemes

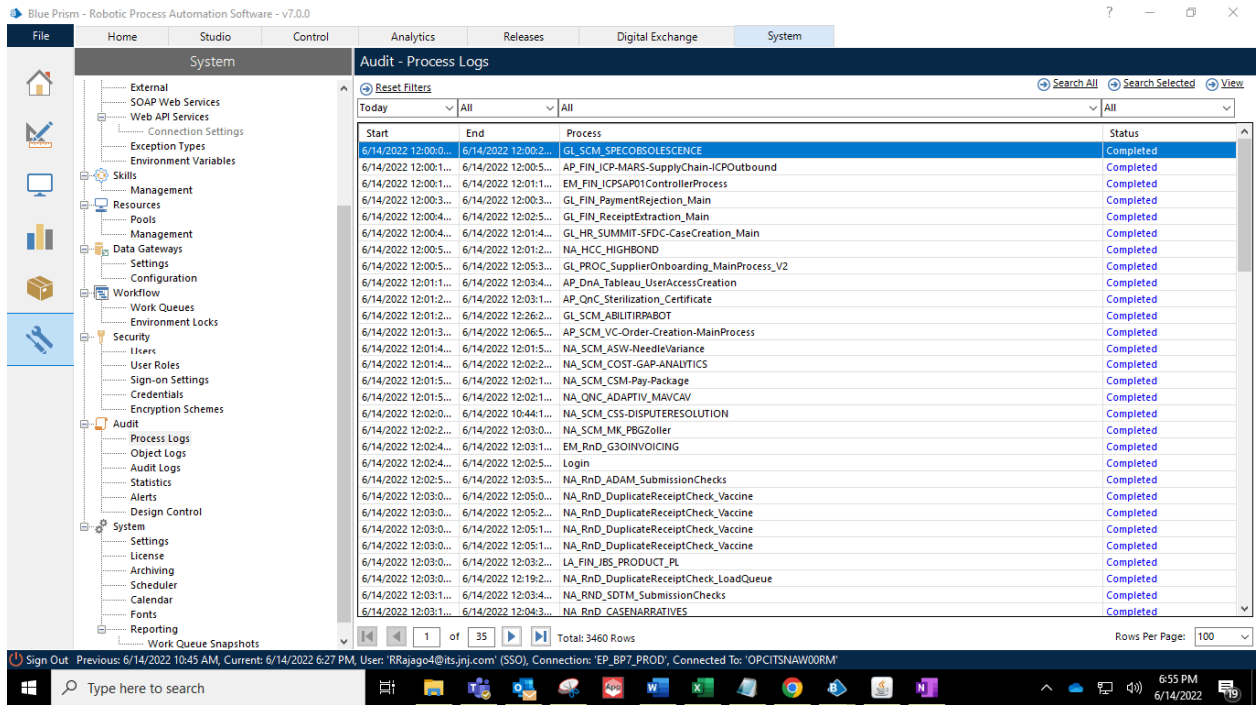
- Default encryption Scheme
- Location = on the Application Server
- The Secret key shall be added to the Server Key Store using the Configuration utility on each Application Server



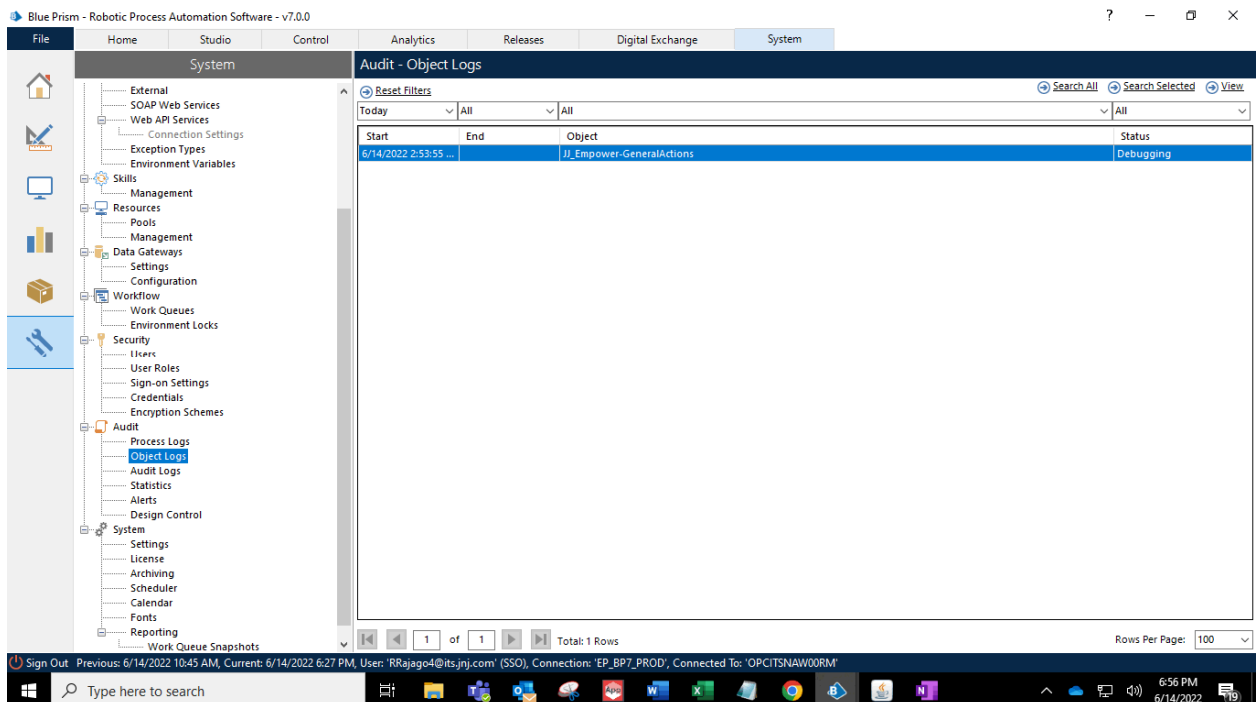
### 13.1.4 Audit – Logs

All audit logs including the following below shall be protected from deletion/modification and retained under J&J guidelines

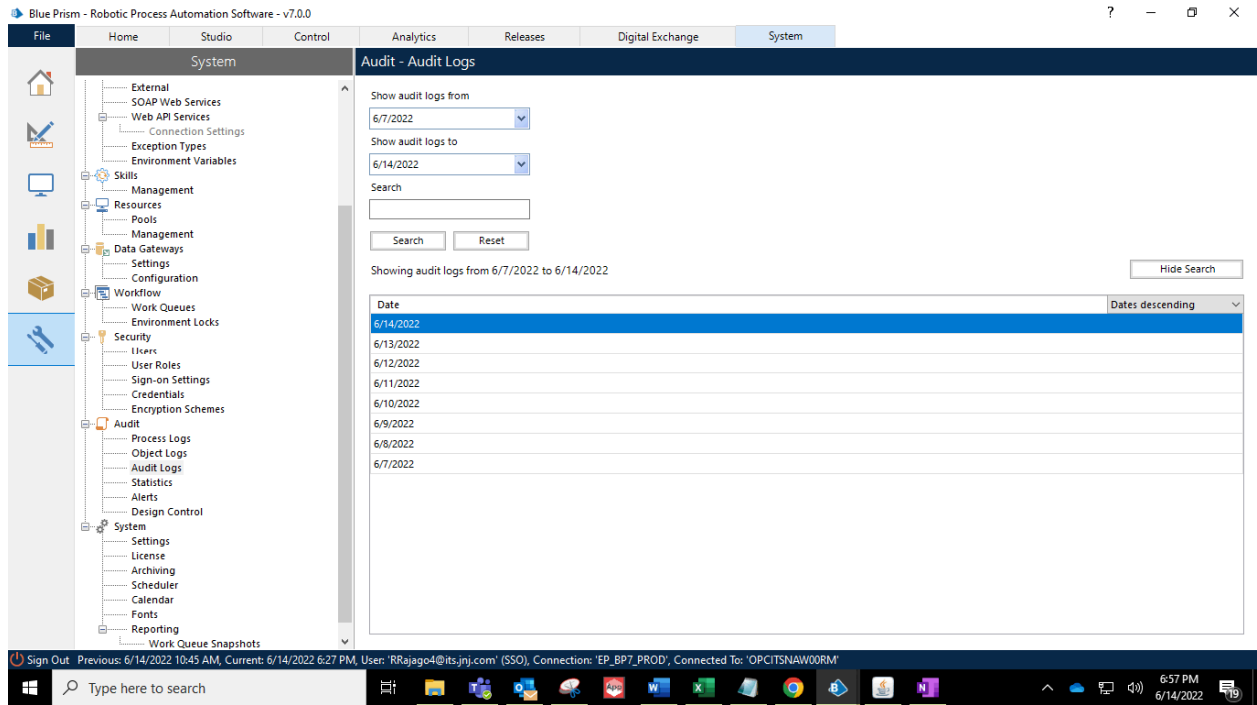
- Process Logs



- Object Logs

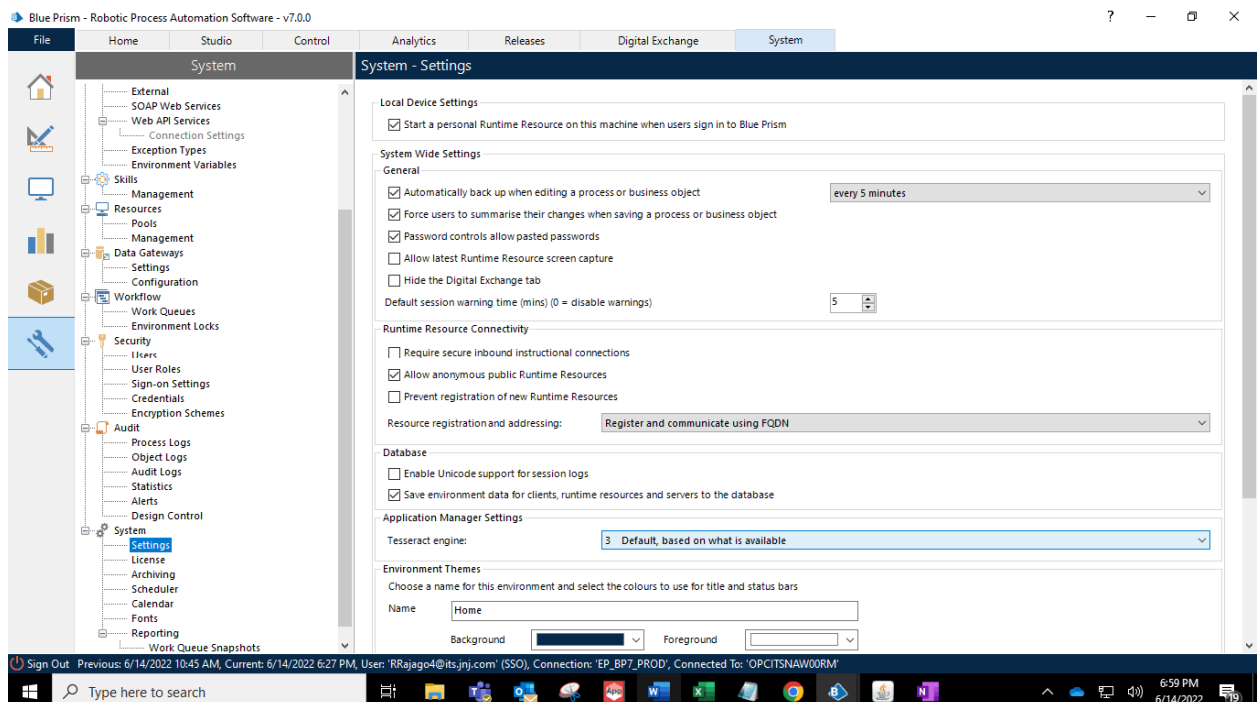


- Audit Logs



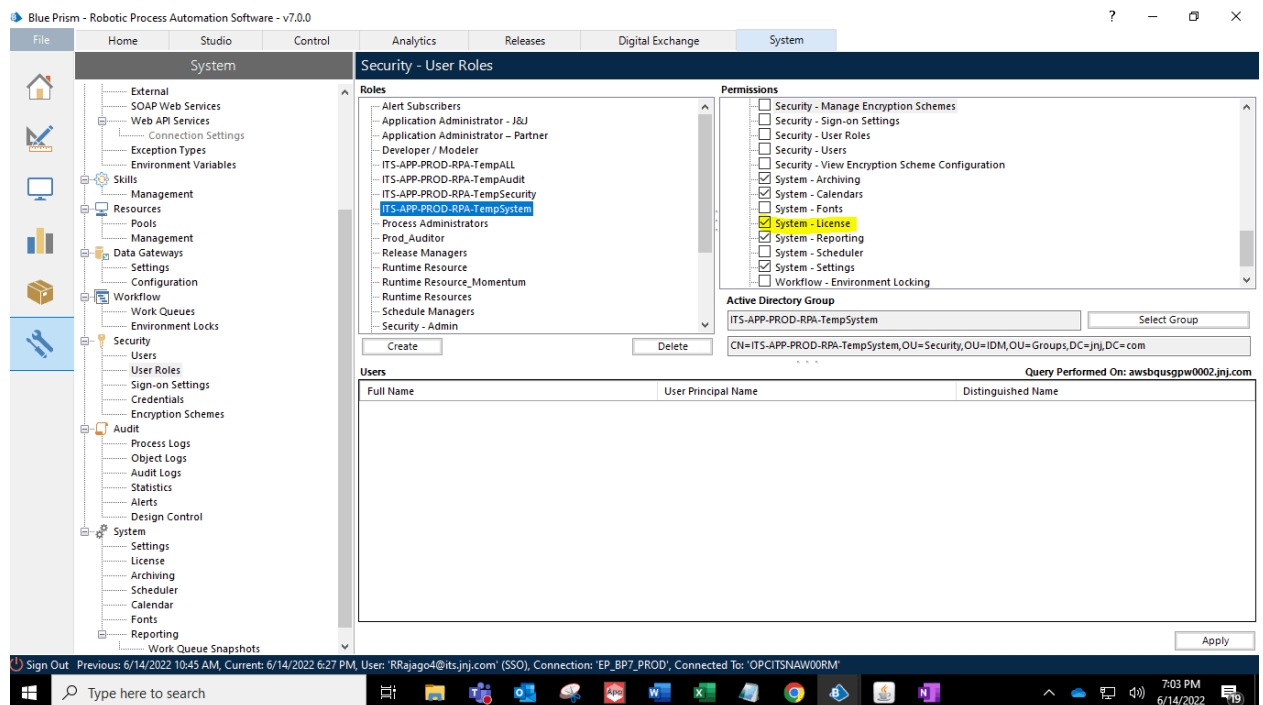
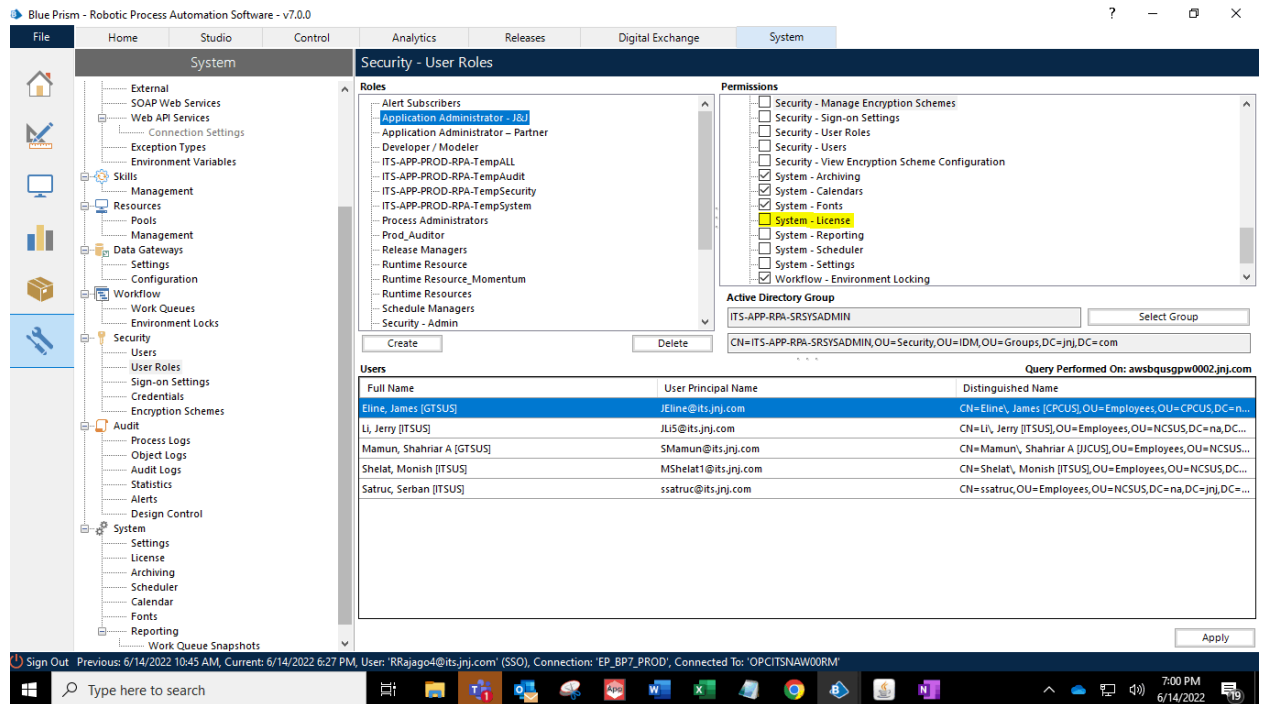
## 13.1.5 System Settings

The following settings shall be defined, tested, and follow an approved Change Management process if a change is required. They shall be defined in a formal/approved Security Configuration Standard (SCS).



### 13.1.6 System License

Administration of bot automation licenses shall be restricted by user role as critical. All modifications to the License must follow the RPA Application Access Management SOP



## 13.1.7 System Archiving

Archiving shall not be performed until further notice-

## 13.1.8 System Scheduler

Administration to modify the Scheduler shall be restricted by user role as critical.

Blue Prism - Robotic Process Automation Software - v7.0.0

System - User Roles

Roles

- ITS-APP-PROD-RPA-TempSecurity
- ITS-APP-PROD-RPA-TempSystem
- Process Administrators
- Prod\_Auditor
- Release Managers
- Runtime Resource
- Runtime Resource\_Momentum
- Runtime Resources
- Schedule Managers
- Security - Admin
- Security-Operation
- Support Engineer**
- System Administrators
- Temp Role - RPA Admin
- Testers
- Web Service Consumers

Permissions

- ☐ Object Studio
- ☒ Process Alerts
- ☒ Process Studio
- ☒ Release Manager
- ☒ Resources
- ☒ Scheduler
  - ☐ Create Schedule
  - ☐ Delete Schedule
  - ☐ Edit Schedule
  - ☐ Retire Schedule
  - ☒ System - Scheduler
  - ☒ View Schedule
- ☐ Skills

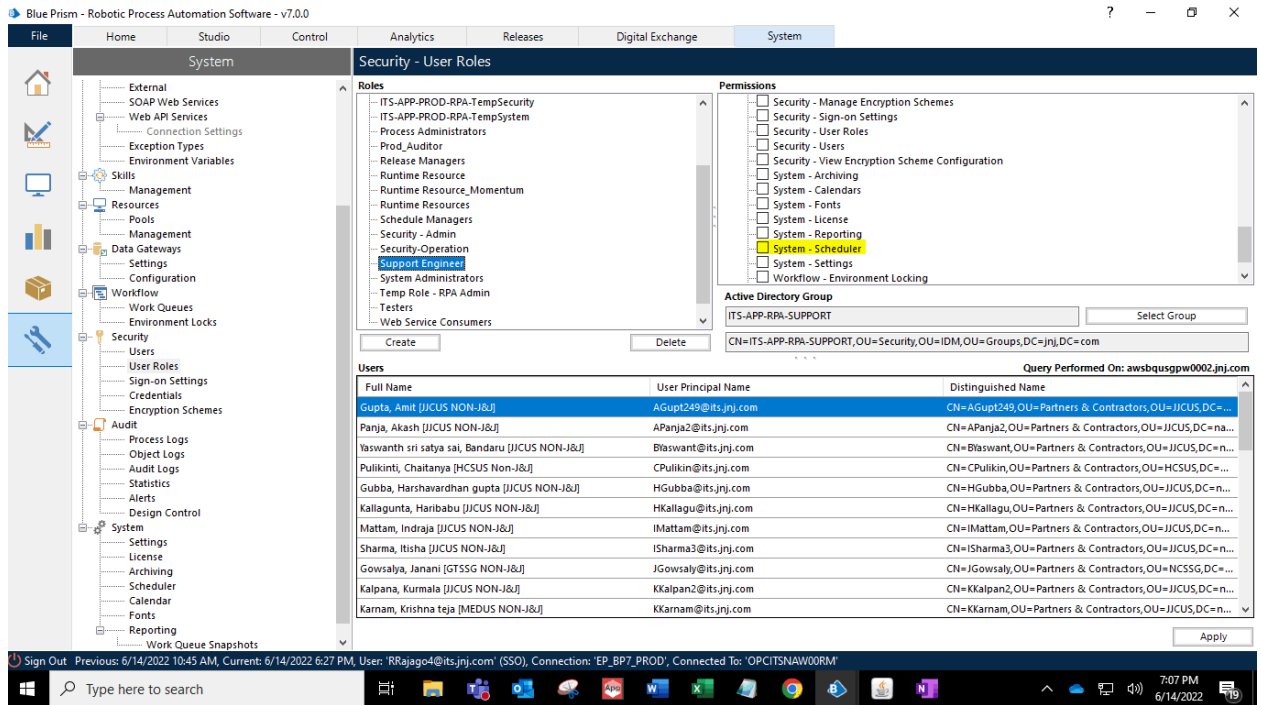
Active Directory Group

ITS-APP-RPA-SUPPORT

Query Performed On: awsbusqpw0002.jnj.com

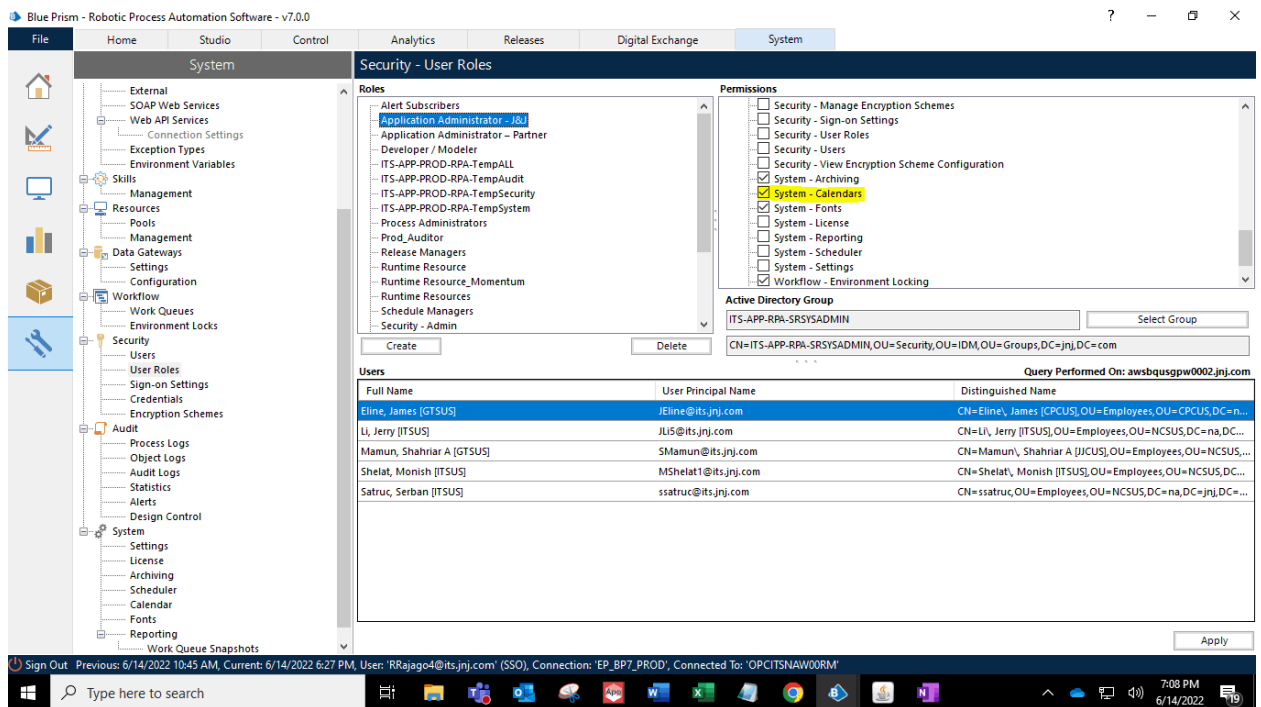
Full Name	User Principal Name	Distinguished Name
Gupta, Amit [JJCUS NON-J&J]	AGupt249@its.jnj.com	CN=AGupt249,OU=Partners & Contractors,OU=JJCUS,DC=...
Panja, Akash [JJCUS NON-J&J]	APanja2@its.jnj.com	CN=APanja2,OU=Partners & Contractors,OU=JJCUS,DC=na...
Yaswanth sri satya sai, Bandaru [JJCUS NON-J&J]	Blaswant@its.jnj.com	CN=Blaswant,OU=Partners & Contractors,OU=JJCUS,DC=n...
Pulikinti, Chaitanya [HCSUS NON-J&J]	CPulikin@its.jnj.com	CN=CPulikin,OU=Partners & Contractors,OU=HCSUS,DC=...
Gubba, Harshavardhan gupta [JJCUS NON-J&J]	HGubba@its.jnj.com	CN=HGubba,OU=Partners & Contractors,OU=JJCUS,DC=n...
Kallagunta, Haribabu [JJCUS NON-J&J]	HKallagu@its.jnj.com	CN=HKallagu,OU=Partners & Contractors,OU=JJCUS,DC=n...
Mattam, Indrāja [JJCUS NON-J&J]	IMattam@its.jnj.com	CN=IMattam,OU=Partners & Contractors,OU=JJCUS,DC=n...
Sharma, Risha [JJCUS NON-J&J]	ISharma3@its.jnj.com	CN=ISharma3,OU=Partners & Contractors,OU=JJCUS,DC=n...
Gowsalya, Janani [GTSSG NON-J&J]	JGowsaly@its.jnj.com	CN=JGowsaly,OU=Partners & Contractors,OU=NCSSG,DC=...
Kalpana, Kurnala [JJCUS NON-J&J]	KKalpan2@its.jnj.com	CN=KKalpan2,OU=Partners & Contractors,OU=JJCUS,DC=n...
Karnam, Krishna teja [MEDUS NON-J&J]	KKarnam@its.jnj.com	CN=KKarnam,OU=Partners & Contractors,OU=JJCUS,DC=n...

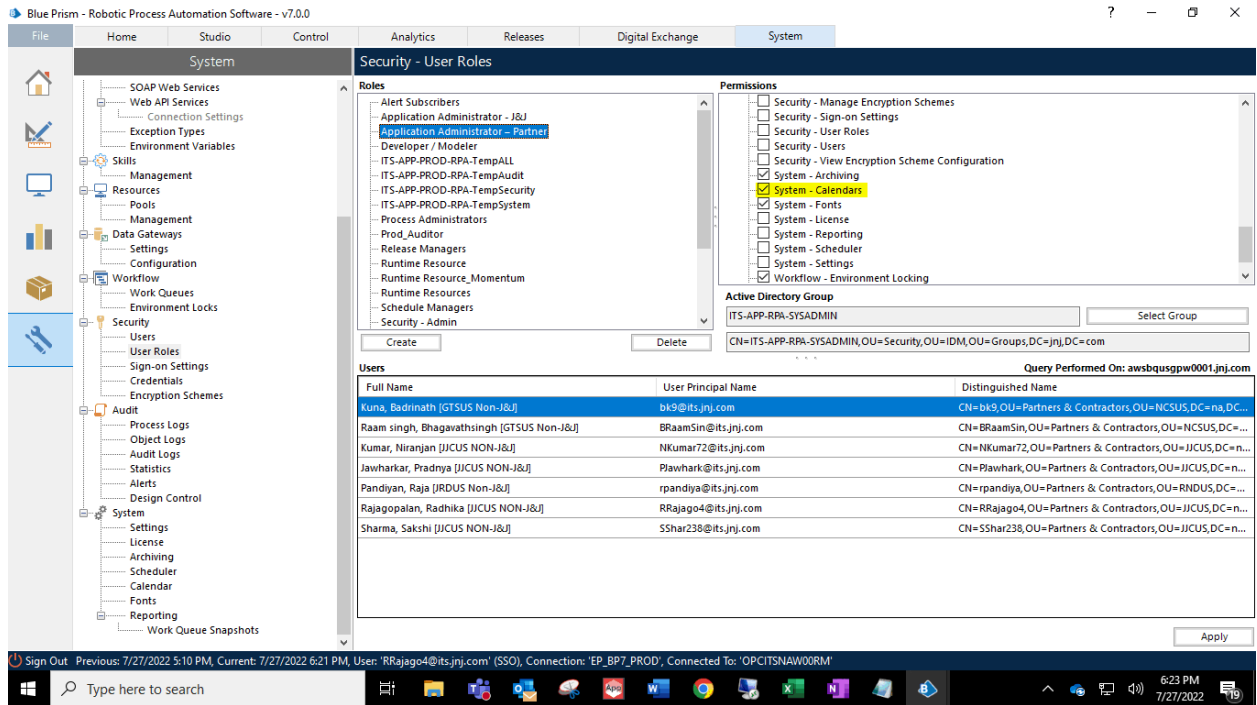
Sign Out Previous: 6/14/2022 10:45 AM, Current: 6/14/2022 6:27 PM, User: 'RRajago4@its.jnj.com' (SSO), Connection: 'EP\_BP7\_PROD', Connected To: 'OPCITSNAW00RM'



## 13.1.9 System Calendar

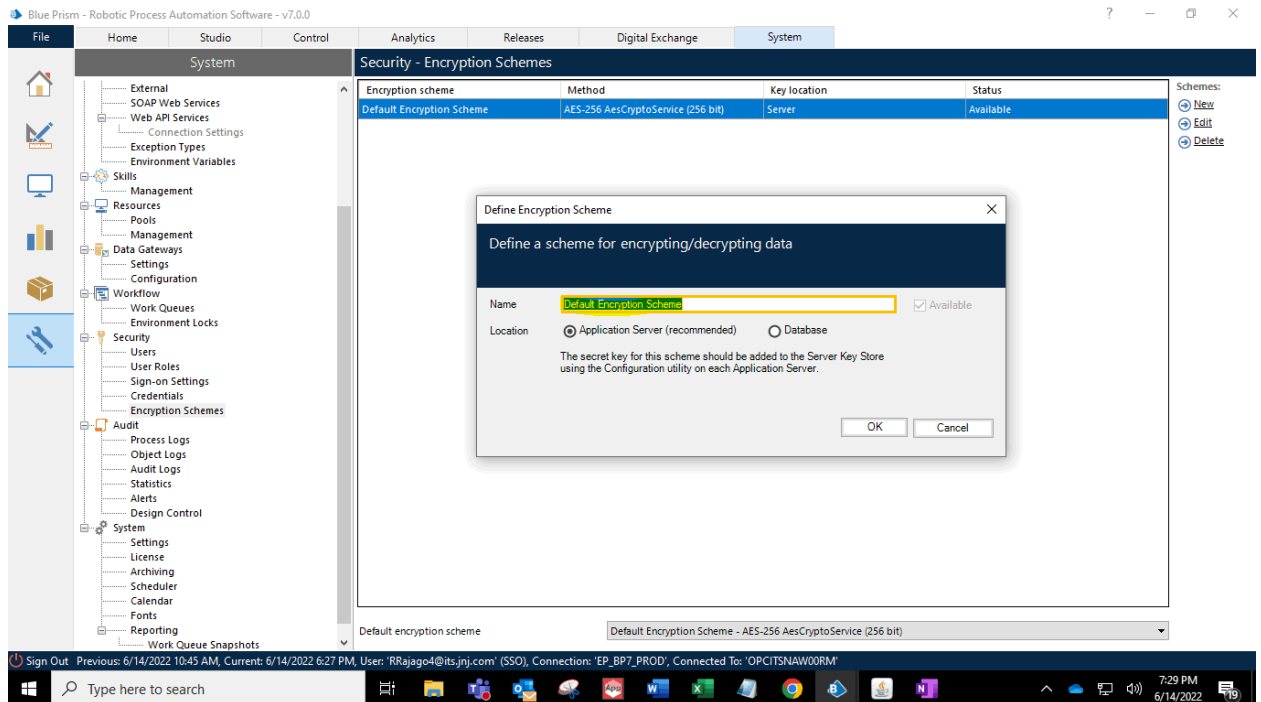
Administration to modify the calendar shall be restricted by user role as critical.





## 13.1.10 Workflow and Work Queues

All Work Queues shall be encrypted using the "Default Encryption Scheme".





## 14. Stage and Flow Validation Check

The following is the Design Control set-up for Development environment. Modifications will follow the IRIS Change Request process outlined in the RPA Change Management process.

Category	Enabled	Validation Check	Severity
Stage Validation	TRUE	No name assigned to ...	Warning
Stage Validation	TRUE	Cannot store data in an environment variable	Error
Stage Validation	TRUE	Cannot determine resulting data type of expression...	Error
Stage Validation	TRUE	Value to be stored is not compatible with destination...	Error
Stage Validation	TRUE	Field '...' does not exist in the collection...	Error
Stage Validation	TRUE	Stage to store result in is not accessible from this page...	Error
Stage Validation	TRUE	Stage to store in is not a Data or Collection stage...	Error
Stage Validation	TRUE	Stage to store result in does not exist...	Error
Stage Validation	TRUE	Stage to read value from does not exist...	Error
Stage Validation	TRUE	The chosen action in row ... is not permitted for the corresponding element type	Error
Stage Validation	TRUE	Row ... contains an action which is outdated. Please consider updating.	Warning
Stage Validation	TRUE	Internal Error: No type information found on the element in row ....	Error
Stage Validation	TRUE	There is no action selected in row ...	Error
Stage Validation	TRUE	Failed to find referenced application element in row .... Element ID is ...	Error
Stage Validation	TRUE	No application element chosen in row ...	Error
Stage Validation	TRUE	Expression evaluates to the wrong datatype	Error
Stage Validation	TRUE	Expression... is valid, but its data type cannot be resolved until runtime	Warning
Stage Validation	TRUE	Invalid expression... - ...	Error
Stage Validation	TRUE	Expression... is blank	Error
Stage Validation	TRUE	Block overlaps another block	Error
Stage Validation	TRUE	Stage is only partially within a block	Error
Stage Validation	TRUE	Page '...' is not called by any other page	Warning
Stage Validation	TRUE	Resume stage is not in a recovery section	Error
Stage Validation	TRUE	Exception stage wants to use existing exception information, but there can be no existing exception there	Error
Stage Validation	TRUE	Wrong type of parameter	Error
Stage Validation	TRUE	Bad map type for an expression-type parameter	Error
Stage Validation	TRUE	Bad map type for a stage-type parameter	Error
Stage Validation	TRUE	Collection mismatch...: ...	Error



# RPA BLUE PRISM DEVELOPMENT STANDARDS

Category	Enabled	Validation Check	Severity
Stage Validation	TRUE	Expression... uses the function ..., which is deprecated	Warning
Stage Validation	TRUE	Stage has blank name	Warning
Stage Validation	TRUE	Invalid data type match in row ....	Error
Stage Validation	TRUE	The stage referenced in row ... is out of scope.	Error
Stage Validation	TRUE	The stage referenced in row ... does not exist.	Error
Stage Validation	TRUE	The element referenced in row ... has no data type.	Error
Stage Validation	TRUE	Failed to find referenced application element in row .... Element ID is ...	Error
Stage Validation	TRUE	The stage calls a process with parameter '...' which is not defined in the stage	Warning
Stage Validation	TRUE	The Page Reference stage provides a value for the input parameter '...' which is no longer defined by the target page.	Warning
Stage Validation	TRUE	The Page Reference stage provides a mapping for the Output parameter '...' which is no longer defined by the target page.	Error
Stage Validation	TRUE	Row ... does not have a valid target element	Error
Stage Validation	TRUE	Row ... does not specify a condition	Error
Stage Validation	TRUE	Row ... contains a condition which is outdated. Please consider updating.	Warning
Stage Validation	TRUE	Row ... has no application element set	Error
Stage Validation	TRUE	Row ... does not have a condition which is appropriate to the target element	Error
Stage Validation	TRUE	Row ... has no read action defined.	Error
Stage Validation	TRUE	The chosen read action in row ... is not permitted for the corresponding element type	Error
Stage Validation	TRUE	Row ... contains an action which is outdated. Please consider updating.	Warning
Stage Validation	TRUE	Row ... has an invalid store in location - the stage '...' does not exist.	Error
Stage Validation	TRUE	Row ... has an invalid store in location - the collection stage '...' has some fields defined, whereas an empty (dynamic) collection is required.	Error
Stage Validation	TRUE	Row ... has a blank store in location	Error
Stage Validation	TRUE	Row ... has an invalid store in location - the stage '...' cannot be found....	Error
Stage Validation	TRUE	Row ... has an invalid store in location - the stage '...' is located on another page and is marked as hidden.	Error
Stage Validation	TRUE	Row ... does not refer to a valid application element	Error
Stage Validation	TRUE	Row ... supplies a blank expression to argument '...'	Error
Stage Validation	TRUE	Row ... does not supply a value to the compulsory parameter '...'	Error

# RPA BLUE PRISM DEVELOPMENT STANDARDS

Category	Enabled	Validation Check	Severity
Stage Validation	TRUE	Row ... contains an argument '...' that is not defined in the application model	Error
Stage Validation	TRUE	Missing element parameter for attribute '...' in row ...	Error
Stage Validation	TRUE	The data item '...' on page '...' conflicts with the data item '...' on page '...'	Error
Stage Validation	TRUE	Data item does not yet have a data type	Error
Stage Validation	TRUE	The Action has no Business Object (or Action) set.	Error
Stage Validation	TRUE	The Action refers to the Business Object '...' which does not seem to be installed.	Error
Stage Validation	TRUE	The referenced business object has been retired.	Error
Stage Validation	TRUE	The Action refers to the action '...' within the Business Object '...' but this Business Object does not contain an action with that name.	Error
Stage Validation	TRUE	The Page Reference stage does not refer to any page.	Error
Stage Validation	TRUE	The Page Reference stage refers to a page which does not exist.	Error
Stage Validation	TRUE	The Page Reference stage does not acknowledge the input parameter '...' defined by the target page.	Error
Stage Validation	TRUE	The Page Reference stage does not acknowledge the output parameter '...' defined by the target page.	Error
Stage Validation	TRUE	Collection field definition '...' has an invalid name	Error
Stage Validation	TRUE	Initial value field '...' has an invalid name	Error
Stage Validation	TRUE	Data item refers to a missing environment variable '...'	Error
Stage Validation	TRUE	Undefined loop type	Error
Stage Validation	TRUE	Unrecognized loop type	Error
Stage Validation	TRUE	No collection set for loop stage	Error
Stage Validation	TRUE	Cannot find collection stage '...' referenced in loop stage	Error
Stage Validation	TRUE	Loop stages references a collection which is not accessible - it is private and resides on another page	Error
Stage Validation	TRUE	The Process stage has not yet set the subprocess.	Error
Stage Validation	TRUE	The Process stage references a subprocess which has been retired.	Error
Stage Validation	TRUE	Compiler warning at line ...: ...	Error
Stage Validation	TRUE	Compiler error at line ...: ...	Error
Stage Validation	TRUE	Compiler warning at top section line ...: ...	Error
Stage Validation	TRUE	Compiler error at top section line ...: ...	Error
Stage Validation	TRUE	Expression... uses Exception function but is not in a Recovery section	Error
Stage Validation	TRUE	Referenced subprocess does not exist	Error
Stage Validation	TRUE	No 'Store In' mapping set...	Warning
Stage Validation	TRUE	Blank value supplied...	Warning

# RPA BLUE PRISM DEVELOPMENT STANDARDS

Category	Enabled	Validation Check	Severity
Stage Validation	TRUE	Data item '...' is not referenced	Advice
Stage Validation	TRUE	Collection '...' is not referenced	Advice
Stage Validation	TRUE	Application element '...' is not referenced	Advice
Stage Validation	TRUE	"Pause After Step" not specified in navigate stage	Warning
Stage Validation	TRUE	Exception Type '{0}' not previously defined within the environment	Advice
Stage Validation	TRUE	Stage has a direct or indirect reference to Processes or Objects which may contain references to items that the user does not have permission to use. Execution may fail for this user: {0}	Warning
Stage Validation	TRUE	Web API Business Object Action input collection mismatch{0}: {1}	Error
Stage Validation	TRUE	The collection stage used{0} in the Web API Business Object Action inputs has no defined fields: {1}	Warning
Flow Validation	TRUE	The Skill '{0}' is not installed or is unavailable.	Error
Flow Validation	TRUE	The data type ({1}) for data stage '{0}' does not match its associated environment variable data type ({2}).	Error
Flow Validation	TRUE	'Batch Size' input value is - {1}. {2}	Warning
Flow Validation	TRUE	False link goes to a different page	Error
Flow Validation	TRUE	Invalid link	Error
Flow Validation	TRUE	Invalid link for the False result	Error
Flow Validation	TRUE	Invalid link for the True result	Error
Flow Validation	TRUE	Invalid link from ...	Error
Flow Validation	TRUE	Link to a different page	Error
Flow Validation	TRUE	Link to a different page from ...	Error
Flow Validation	TRUE	Loop end is on a different page	Error
Flow Validation	TRUE	Loop start is on a different page	Error
Flow Validation	TRUE	Missing end stage for loop	Error
Flow Validation	TRUE	Missing group ID	Error
Flow Validation	TRUE	Missing group ID on loop start stage	Error
Flow Validation	TRUE	Missing link	Error
Flow Validation	TRUE	Missing start stage for loop	Error
Flow Validation	TRUE	No link from ...	Error
Flow Validation	TRUE	No link when the decision is False	Error
Flow Validation	TRUE	No link when the decision is True	Error
Flow Validation	TRUE	Recovery stage is linked back to main process	Error
Flow Validation	TRUE	True link goes to a different page	Error
Documentation	TRUE	Action stage has no description	Advice

Category	Enabled	Validation Check	Severity
Control			
Documentation Control	TRUE	Application element "..." has dynamic attributes but no notes	Advice
Documentation Control	TRUE	No preconditions are defined	Advice
Documentation Control	TRUE	No postconditions are defined	Advice
Documentation Control	TRUE	No description given for parameter '...'	Advice
Documentation Control	TRUE	Published action does not contain a description...	Advice

## 15. Third Party Tools

- The recommended way to include code as part of a Blue Prism Process is to use a Code stage within a Visual Business Object (VBO).
- Since Blue Prism has been built upon the Microsoft .NET framework it is recommended that Visual Basic, or C# be the preferred languages for in-app code development.
- The 2nd way to include external code snippets are to develop the functionality external to the Blue Prism environment and call the code segment via a command line interface using the Utility – Environment (VBO) and selecting the Start Process action.
- However, it should be noted that adding code to a process this way requires the host virtual environment to have the ability to interpret the submitted code.
- For example, for Python that means a python interpreter must be installed on the host environment.

### 15.1 Blue Prism Digital Exchange (DX) VBO

Third-party VBOs are available on Blue Prism's internal Digital Exchange network which may be required for the development of a Blue Prism automation. Please use the following procedure to request one be installed:

1. Submit Service Request to "Robotic Process Automation Platform Team L2" requesting a new VBO is installed on Sandbox from Digital Exchange (DX)
  - a. Provide DX URL and reasoning
2. Developer will analyze the VBO in Sandbox for practicality.
  - a. Note: You may not access a production or QA system during your analysis when connected to the Sandbox.
3. A decision is made to bring the VBO on for use in development.
4. The VBO is scanned and/or manually reviewed by the security/compliance team.
5. The CA must identify a third-party VBO is being brought in, clarify it had been scanned and/or reviewed, and it passed cleanly.

6. Approval is granted by both security/compliance and the RPA team.
7. Once approved in Sandbox, an additional Service Request will be necessary to request the VBO be installed on DEV.
8. Once on DEV the newly installed VBO can be included in future release packages for deployment to QA and/or PROD.

## **15.2 Third-Party Programming Environments and Source Code**

In some cases, non-Blue Prism development environments may be necessary to interact with a Blue Prism automation. These third-party environments perform certain tasks outside the scope of Blue Prism such as utilizing a Python script to perform a statistical analysis on a set of data collected by an automation. For these situations, please use the following procedure:

1. Determine the programming environment you plan to implement as well as any source code (including custom and open-source libraries, frameworks, SDKs, etc.).
2. The security/compliance team must scan or manually review any source code being executed.
3. The CA must identify a third-party programming environment is being brought in, clarify it had been scanned and/or reviewed, and it passed cleanly.
4. Approval is granted by both security/compliance and the RPA team.
5. Once approved, the developer can work with the platform team to install and configure the new programming environment on the necessary VI workspace(s).

## **16. Chrome Browser Best Practices**

### **16.1 Chrome Browser Version**

- Use an evaluated Chrome browser version.
- Latest Blue Prism Extension must be downloaded and enabled.
  - This extension provides connectivity between the browser and Blue Prism.
  - Avoid using chrome extension versions 2.2.0.2 and 2.2.0.4

### **16.2 Download Settings**

- In Settings>Downloads, “Ask where to save each file before downloading” should be checked off. Otherwise, any file downloaded from the browser will go directly to the users Downloads file on their machine.

## Downloads

Location C:\Users\MSulli11\Downloads	<a href="#">Change</a>
Ask where to save each file before downloading <input checked="" type="checkbox"/>	

### 16.3 Spy Modes

- When using Browser Mode to spy the elements on chrome browser and Browser mode is not getting activated, then possible reason is that chrome is not closed properly, and it's tied with previous session. To check chrome is running properly or not follow below steps:
  - Open command prompt with admin/elevated privileges.
  - Run the command **'netstat -ano|find "31924"**.
  - If it shows result with rows but chrome application is not opened that means it didn't release the port "31924" from previous session.
  - In such case close you have two options:
    - i. Close and save any files open and then reboot the machine. Reboot will help to release the port and then you will be able to use browser mode for spying the elements.
    - ii. Release the port by following these steps:
      - a. Close Blue Prism (Automate.exe process)
      - b. Again, type **netstat -ano|find "31924"** in **command prompt**.
      - c. Find the PID number for that **port** number (it should be the last number to the right)
      - d. Do one of the following:
        - i. Go to Task Manager and end process the corresponding PID number.
        - ii. Open **command prompt** and type in **taskkill /F /PID <PID FROM ABOVE>**
      - e. Reopen Blue Prism and Chrome should have the ability to be spied again.

### 16.4 Match Optimization

When an application has a complicated user interface containing many user interface elements, it can take Blue Prism a long time to find the element within the Application Model. There are a number of attributes that affect the way in which the search is performed and, in some cases, can speed up the search time.

### 16.4.1 Optimization attributes

- Ancestor count
  - The ancestor count attribute is particularly useful in a tree like object model which includes Active Accessibility and Java.
  - Within these models searches along the branches of the tree can be stopped short when we are about to descend deeper than the ancestor count.
  - Ancestor count can be used for all application types but only provides optimization in AA and Java. It should be used with care as the ancestor count may change as the model is updated.
  - The HTML ancestor count attribute is known to change across IE versions so elements using the ancestor count may have to be re-spied when IE is upgraded.
- Match index
  - The match index attribute allows the search to stop as soon as the Nth element is found.
  - This should be used carefully because usually the search would continue looking for duplicate matches and thus eliminates an ambiguous match.
  - It is also not very safe to assume that the elements will be found in the same order, and so the Nth element could potentially be a different element than the one that was expected.
- Match reverse
  - The match reverse attribute simply changes the direction of the search, this can be useful if the element is at the end of the search list.
  - The match reverse attribute is only relevant when used in combination with the match index attribute.

### 16.4.2 Browser-based application optimization

- Paths
  - Matching against the Path attribute of an Element within a browser-based application will be fast.
  - Where the path attribute is constant it is most appropriate to use this attribute for matching.
- Element ID
  - When the path attribute is not a suitable attribute to match against the next best alternative is to match on Element ID.
  - The Element ID is name that the web page developer may have added to an element. It is required to be unique and provides the ability for scripts within the web page to look up the element quickly.
- Exclude HTC components
  - Within the Application Modeller wizard there is an option to exclude HTC components. Turning this option on will improve performance.
  - The only time this option should be left turned off is when the application being interfaced with uses HTC components.
- Only interact with elements on the active browser tab

- Within the Application Modeller wizard there is an option to only interact with elements within the current tab.
- This option will provide more consistent results and is turned on by default. When this option is turned on you must switch to a different tab before interacting with it.
- Turning this option off allows interaction with elements on tabs which are not currently active, but this can sometimes produce unexpected results.

## 17. Document History

### 17.1 Superseded Document(s)

- **Doc ID:** – N/A      **Title:** – N/A      **Effective Date:** – N/A

### 17.2 Revision History

Version	DD-Mmm-YYYY	Author	Change Summary	Major/Minor Change
7.0	28-Aug-2024	Anoop Kumar	<ul style="list-style-type: none"> <li>• Added section 5.1.6 BP Capture - Use of BP Capture</li> </ul>	Major
6.0	26-Oct-2023	Anoop Kumar	<ul style="list-style-type: none"> <li>• Added section 5.1.5 for Graph API Object Usage</li> </ul>	Major
5.0	14-July-2022	Radhika R	<ul style="list-style-type: none"> <li>• Updated Section 13 for SCS with screenshots</li> <li>• Updated Section 13 for SOX requirements</li> <li>• Added Blue Prism Administrators Group For AMS in Section 13.1.1</li> <li>• Changed Design Control to a new section 14 Stage &amp; Flow Validation Check</li> </ul>	Major
4.0	12-Mar-2021	Pratibha Hiray	<ul style="list-style-type: none"> <li>• Added BP Memory</li> </ul>	Major



# RPA BLUE PRISM DEVELOPMENT STANDARDS

Version	DD-Mmm-YYYY	Author	Change Summary	Major/Minor Change
			Management <ul style="list-style-type: none"> <li>Added Chrome Browser Best Practice</li> </ul>	
3.0	04-Jun-2020	Pratibha Hiray	<ul style="list-style-type: none"> <li>Added Third Party Tools</li> <li>Added Retry Logic</li> <li>Standard Block Color Guide</li> <li>Added Incident Creation Process</li> </ul>	Major
2.0	22-Feb-2019	Dave Krause	<ul style="list-style-type: none"> <li>Added Readability Section</li> <li>Revised Object Design Section</li> <li>Added Test Strategy Section</li> <li>Added Interacting with Live Environments Section</li> <li>Updated Exception Handling Section</li> </ul>	Major
1.0	08-Aug-2018	Jarel Hawkins	New Document	Major

SOP-8721 ( Version 7.0) - EFFECTIVE - Check electronic version in eDMS before use

# UNCONTROLLED PRINT

## J&J SERVICES, INC.

### CONFIDENTIAL - USE PURSUANT TO COMPANY INSTRUCTIONS

**Title:** RPA Blue Prism Development Standards

Signed By: LI CHANGYU  
Decision: Approved  
Decision Date: 9/19/2024 11:05:40 AM  
Role: Owner  
Purpose: Approve Document  
Meaning Of Signature: As the Owner, I confirm that this document is technically correct and complete, and that the appropriate persons are reviewing/approving this document

Signed By: Hamid Omar  
Decision: Approved  
Decision Date: 9/20/2024 1:34:05 PM  
Role: Q&C Functional Approver  
Purpose: Approve Document  
Meaning Of Signature: As the Q&C functional approver, I confirm that this document is correct and complete from a quality perspective.

Signed By: Sinha Riya  
Decision: Approved  
Decision Date: 9/23/2024 10:26:05 AM  
Role: Q&C Document Approver  
Purpose: Approve Document  
Meaning Of Signature: As a Q&C approver, I confirm that this document complies with Document management process.

SOP-8721 ( Version 7.0) - EFFECTIVE - Check electronic version in eDMS before use