Logo

Description automatically generated

**RPA Best Coding Practice & Quality Standards**

March 16th, 2022

Version nr. 1

Contents

[1. Definitions 3](#_Toc98145315)

[2. Library Structure 3](#_Toc98145316)

[2.1. Objects 3](#_Toc98145317)

[2.2. Process 4](#_Toc98145318)

[3. Logging 5](#_Toc98145319)

[3.1. Business Objects 5](#_Toc98145320)

[3.2. Process 5](#_Toc98145321)

[4. Process 5](#_Toc98145322)

[4.1. General Standards 5](#_Toc98145323)

[4.2. Main Page 9](#_Toc98145324)

[4.3. Initialize page 9](#_Toc98145325)

[4.5. Item Processing (after Get Next Item prior marking item completion/exception) 10](#_Toc98145326)

[4.6. Status and Tags 10](#_Toc98145327)

[4.7. Reporting 11](#_Toc98145328)

[5. Sub-Process 11](#_Toc98145329)

[6. Business Objects 11](#_Toc98145330)

[6.1. General Standards 11](#_Toc98145331)

[6.2. Application Modeler 14](#_Toc98145332)

[6.2.1. Chromium Elements (Chrome/Edge) 14](#_Toc98145333)

[6.2.2. Naming conventions 15](#_Toc98145334)

[6.3. Attach/Activate 15](#_Toc98145335)

[6.4. Wait Stages 15](#_Toc98145336)

[6.5. Surface Automation 17](#_Toc98145337)

[7. Dynamic Objects 17](#_Toc98145338)

[8. Wrapper Objects 18](#_Toc98145339)

[9. System Settings 18](#_Toc98145340)

[9.1. Credentials 18](#_Toc98145341)

[9.2. Environmental Variables 18](#_Toc98145342)

[10. Testing and Quality Assurance 18](#_Toc98145343)

[10.1. Change Requests 18](#_Toc98145344)

[10.2. Maintenance 18](#_Toc98145345)

[11. Custom Code Stages 18](#_Toc98145346)

# **Definitions**

1. **Main Process/Process – Blue Prism Process, that contains all the main process logic**
2. Sub-Processes – Process that is created, to be called from another process
3. Splitted Process – Processes, that can be related to the same project, but can work independently on each other.
4. Objects – Blue Prism object, created for specific application window, with small actions for every field in the application, build in a way, that it would make sure, that robot is in correct window, and that actions that robot is about to perform are completed successfully. Purpose of these is to create multiple small actions, that could be used in different processes, using the same application windows.
5. Actions – Pages in objects
6. Wrapper Object – Blue Prism object, that has references to other objects and can be built with business logic. Purpose of these is to create actions that contains multiple different steps for examples Make a refund, create a case in salesforce, download report.
7. Sensitive information – Names, addresses, phone numbers, email address and other information, that could lead to a specific person.

# **Library Structure**

## Objects

* All objects should be placed under Objects > RPA BPM modelers > Application Name
* If Application Folder, contains old objects (not based on our standards), new folder Reusable should be created, and all new objects stored under it:

Graphical user interface, text

Description automatically generated

* Under main Folder, there should be object with general utilities, and folder, that would represent other windows, that either is main window, or windows, that could be reached from multiple places:

Text

Description automatically generated with medium confidence Text

Description automatically generated

* + Structure of library: should be done by a tree view principle. Folder, represents a main page, sub folders, can be names of child pages that could be reached from current page, or collection of pages, that could be reached from there.  
    A picture containing application

    Description automatically generatedA picture containing timeline

    Description automatically generated

## Process

* All Processes should be placed under Processes > RPA BMP modelers.
* Process Itself should be placed under the folder with the name of the project



Text

Description automatically generated

# **Logging**

## Business Objects

* All stages – Log Errors Only

## Process

* Decision stages/Choice stages – Enabled, if not in loop and help to determine, which path robot worked on, (stages for retries are not mandatory to be logged)
* Logging Item Key – Calculation Stage with enabled logging just to log current item key **must** be in every process.
* Calculation/Multi Calculation stages – can be used to log general details, that would help to debug process. MUST NOT contain any sensitive information. Such calculation stages should start with **Log** , so that LazyTool would be able to identify such stages (ex. Log – Item Key/Log – Adjust Country Name)
* All other stages should be set to Log Errors only.

# **Process**

## General Standards

* All Calculations should be done in Calculation stage, inputs stages must not contain any kind of data manipulation.
* **System Exceptions:**
  + Message: *Application Name: Exception Reason + any additional required/necessary information*
  + Stage Name: *SE: Exception Reason*
* **Business Exceptions**
  + Message: Application Name and/or Process Step: Exception Reason + any additional required/necessary information or message agreed with business.
  + Stage Name: BE: Exception Reason
* Credentials should be named as following: “LIVE –Application Name – ROBOT – Robot Account ID”. **Process Name should be written in Credential’s description.**
* **Make sure, credentials are retrieved only once, and not with every application launch.**
* **All Data Items and Collections must be in Blocks, block name should represent type of data items (Inputs, Outputs, Retry, Credentials, etc…)**
* All Global Data Items, except for Environmental variables, should start with G. (ex. G – Item Data)
* It’s not mandatory to set Environmental variables as global data items.
* Name in Blue Prism must correspond to field "Process Name" of the epic in JIRA, process name should make sense. If Process Name in JIRA contains “OP” with number, please inform your Product Owner to correct it.

Graphical user interface, text, application

Description automatically generated

* Only Business and System Exceptions can be used (avoid System Excepion and other spelling mistakes)Table, timeline

  Description automatically generated with medium confidence
* Avoid Sleep stages in process (Object: Utility – General, Action: Sleep), except for download/kill application actions to make sure that action was performed successfully
  + If sleep stage is used in the process, decision stage with retries should be in place
* Do not use Utility – Environment object, action Start Process, to launch applications, use specific objects instead.
* No infinite loops (Page inceptions) (Retry loops without count)
* No unused Data Items/Collections/Pages and Stages left in Process.
* No Errors



* Excel SHOW can stay as not connected stage, to ease debugging in Production.

Diagram

Description automatically generated

* Avoid nested Retry Logics (when parent and child pages has retries), except for individual retries

**Example** of how **NOT** to do: Parent page, will do retry of 3 times, if pages SFDC Locate Client of SFDC Create Case fails, plus SFDC Create Case has retry logic in page itself, and will retry 3 times if Create Case stage fails, therefore, if Create Case fails, in general there will be 9 retries to create a case (3x3)

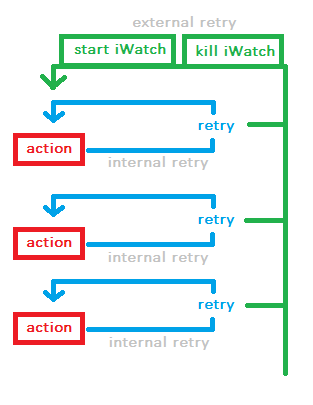
Timeline

Description automatically generated Diagram

Description automatically generated

**Nested individual action retries:**   
*(For iWatch and applications with similar problems)*

iWatch can randomly stop responding, and there are no pre-/post-conditions that could be checked to train the robot to handle such situations. The only solution is to create individual retry loops for every single individual iWatch action, which would retry 3+ times without killing the application. And after the retry loop fails to recover the happy path – bubble-up to higher-level retry loop that restarts iWatch.

Diagram

Description automatically generated

* All processes must be built in a multi-bot architecture, to enable flexible resource assignation. (Environment locks should be in place)
* Global Data items should be in a single page. Page should be named “Global Data”
* It is recommended to avoid using global variables, unless the target data is "travelling" through many pages (e.g., 3+), making the usage of global variables more valuable (especially from a maintenance standpoint) than using multiple inputs/outputs across pages.
* Prior Launching, we must make sure that Chrome is closed.
  + Recommendation to use RPA – Utility – Google Chrome (Wrapper Object).
    - Recommended Flags:  
      - Kill Chrome: True  
      - Safe Kill: False  
      - Clear Cache: True
* If any part, was build not according to our standards, and workaround was used, approval to use a workaround should be received from Senior by email.
  + Email needs to contain:
    - Exact page, and stages, that uses workaround
    - Clear explanation why workaround needed to be used.
    - Reply from Senior, stating that it is approved.
  + Email should be added to Jira, Process QA ticket attachments.
  + Approval must be received and added to ticket prior QA.

## Main Page

* + BB Exception Handling WO (Object: BB Exception, Action: Exception Handling)
    - Should be always after mark item as exception.
    - Exception verification and notification must be handled through BB Exception VBO
  + Garbage Collector should always be after marking item as Completed/Exception
  + Restart VM WO must be used – **Utility – RPA CoE – VM Restart (Wrapper Object)**
    - for one-off process runs: at the end of each session
    - for continuous processes: every 24 hours (controlled by environmental variable for later modification if needed)
  + At the end of the process, all applications must be closed.
  + Use WO for Screenshots after Exceptions
  + Once marking item as completed, reset consecutive exception count.
  + Before Get Work, Start Up page to launch and log-in to all applications used by the process to confirm if all accesses are in place.
  + Contains a safe stop option after every case/item (IsStopRequested()). This will help safely stop robot session without turning it off mid-work.

## Initialize page

* + Font smoothing to be disabled
  + If box is used -> Box Maintainer
  + Windows Username retrieved and stored in global variable
  + Ensure that Chrome popups are enabled
  + Excel templates loaded (i.e., matrix etc.)
  + Set up screen resolution (recommended: 1920x1080)
  + If process related files are stored on networked drives, copy files locally each time (to ensure that latest version is being used). For read only purposes.
  1. Get Work
  + Could be coded into main process or splitted processes, but not in any circumstances in Subprocess.
  + Pending Items check - Get Work page should start with check if pending items exists.
  + (Recommended) Loading queue items should be done for entire collection by uploading whole collection and not looping through it.
  + (Recommended) Check if "Item is in Queue" to avoid duplicates.
  + Environment locking – check if no other robots are working with this part already.
  + Release of Environment lock – it must be ensured that Environment lock will be always released. This is required in case of termination/exception during get work part.

A picture containing diagram

Description automatically generated

* + Get work part, must **not** contain validation for every single item, to check if it is business exception or not, if such logic is needed, process must be split to multiple parts.

## Item Processing (after Get Next Item prior marking item completion/exception)

* + (Recommended) Start and Close applications with every new item.

## Status and Tags

* **Tags** **should be used** **for** Queue Items **grouping** purposes **only**.
* **(Mandatory) No unique information in tags**
* **Avoid leaving tags on items. (Report part to be adjusted, instead of adding tags, remove those after items is being reported)**  
  Graphical user interface, application, table, Excel

  Description automatically generated
* **Don't tag items with unique item identifiers.**  
  Table

  Description automatically generated
* **Don't use tags for item progress logging.**  
  A picture containing application

  Description automatically generated
* **Exception/Completion marking must be set up to as follows:** 
  + - Completed
    - Partially Completed
    - Business Exception
    - System Exception
    - Internal Exception

…always should be for finished items.

Table

Description automatically generated with medium confidence

## Reporting

* If needed, NotReported tag, should be added when creating a new item, and removed, when item is reported.
* Environment lock – it must be ensured that Environment will be always released. This is required in case of termination/exception during process.

# Sub-Process

**Should not be used because of -potential memory leaks and other issues.**

# Business Objects

## General Standards

* Object name:
  + Application name – Window Name

Text

Description automatically generated

* + Application Name – Path how to get to that window

Text

Description automatically generated

* Objects must not contain any errors 
* Object corresponds to 1 Application window (Pop Ups in Web Applications with Basic actions) can be treated as same window.

|  |  |
| --- | --- |
| **Separate Object should be created** | **Can be in the same object** |
| Graphical user interface, application  Description automatically generated  Graphical user interface, text, application, email  Description automatically generated | Graphical user interface  Description automatically generated  Graphical user interface, application  Description automatically generated  Graphical user interface, text  Description automatically generated |

* If Pop up in application is being opened in a new window or tab, separate Object should be created for it.
* All Data Items and Collections must be in Blocks, block name should represent type of data items (Inputs, Outputs, Retry, Credentials, etc.)
* No initial values left in Data Items (except for Flag type data items)
* Raw input/output of data – no data manipulation on Object level (Except for Trim functions.)
* No Business Logic, no Business Exceptions
* Except for Attach page, business object pages must not refer to each other
* A Business object must never call another application or application BO than the one for which it was built
* Pause After Each Step in Navigation stages when multiple steps are used in single stage.
* There can be only exception throwing, but not exception handling (Recovery stage not used) on the object level
* There should be no global visibility and transfer of data between object pages (except Global Timeouts)
* The only data validation permitted is to check if action was completed successfully (correct input/selection etc.)
* Basic Actions stored in object - General Utilities (Launch/Login/Maximize/Close) (Application – General Utilities)
* No Credential Extraction at object level.
* Actions should be small and straight forward, performing one action in that window. (No Dynamic Actions, that do not have Waiting stages at the end.)

|  |
| --- |
| Valid Actions |
|  |
| Invalid Actions |
|  |

* Navigate stages should always contain Activate or Focus actions on UIA elements as a first step
* Application URL link - stored in environment variable or passed as input, if application has multiple environments (UAT/Production)
* If any part, was built not according to our standards, and workaround was used, approval to use a workaround should be received from Senior by email.
  + Email needs to contain:
    - Exact page, and stages, that uses workaround
    - Clear explanation why workaround needed to be used.
    - Reply from Senior, stating that it is approved.
  + Email should be added to Jira, Process QA ticket attachments.
  + Approval must be received and added to ticket prior QA.

## Application Modeler

* Make sure no unnecessary attributes are selected (avoid attributes like element count, ancestor count, empty fields.)
* Root element name should be the same as Object name, or just application name.

Graphical user interface, text, application

Description automatically generated

* Match Index must be used and set to 1 when available.
* Application Modeler must not contain unused elements.

## Chromium Elements (Chrome/Edge)

* No hardcoded HTML paths – hardcoded, Blue Prism generated HTML paths are not stable, and they can change on smallest changes in website, XPath expressions should be used instead.

|  |  |
| --- | --- |
| Good | Bad |
|  |  |

* If XPath is used, at least one more, web attribute must be selected (Web Element Type) is most convenient one. This is BP Bug, as if only XPath will be used, element will appear as visible all the time, even if it is not on the screen.

|  |  |
| --- | --- |
| Bad | Good |
| Graphical user interface, text, application  Description automatically generated |  |

* Avoid Web attributes, when attribute includes numbers in them, as it might change on next load.

|  |  |
| --- | --- |
| Bad | Good |
|  |  |

## Naming conventions

* Elements: Element Type – Element Name (Spy Mode).

|  |  |
| --- | --- |
| **Element Type Examples** | **Spy Mode Naming Examples** |
| Table  Button  Label – (Headers, or field name, that we are about to read value from)  Input  Dropdown  Window  Pop up  Checkbox  Container  Value  List | Browser mode – (Browser)  Win32 mode – (Win32)  UI Automation mode – (UIA)  Region mode – (Region)  Active Accessibility mode – (AA)  Mainframe mode – (Mainframe) |

## Attach/Activate

* Each object must contain Attach page which includes Attach and Activate Application actions
  + For mainframe applications, to activate application, use Object RPA COE – Toolbox, action Foreground Process (Input, process name, without .exe)
* The Attach page reference should be called at the beginning of every other page

## Wait Stages

* Wait Stages must be dynamic (checking for elements to appear), with minimum of 60s timeout
* Wait Stage timeouts must end in System Exception

|  |  |
| --- | --- |
| Valid Actions | Invalid Actions |
|  |  |

* Wait stage for every element that bot is going to interact with (Except or elements used to Activate/Focus/Launch/Attach)
* Wait stage to verify every screen change (after every Navigate stage, except when Select Item or Global Send Keys to inputs information is used)

|  |  |
| --- | --- |
| Valid Actions | Invalid Actions |
|  |  |

## Surface Automation

* Surface Automation relies on pixels to identify elements, therefore before spying regions it is important to ensure that:

1. The VM’s resolution is set to default resolution defined by standards (or as defined by process needs)
2. Font-smoothing is disabled.
3. The application is Maximized.
4. Zoom in Chrome settings must be set to 100%

* Spied regions (parents) must be set to:

1. Location Method: “Image”
2. Position: “Anywhere”
3. Color Tolerance: “70” (or higher, if needed; 70 is the minimal baseline)
4. Grayscale: True

* Spied regions (children) should be used only when the region has no unique pixel-based identifying traits and must be located relative to a nearby parent region:

1. Location Method: Coordinates
2. Position: Relative
3. Relative Parent: [name of other region with location method “Image”]

# **Dynamic Objects**

* Only for Mainframe applications.
* Objects/Process should be the way, that Sleep stages would not be used at the process level, prior or after dynamic objects.
* Proper evaluation, if robot is on the right window should be done either in object or process level.

# **Wrapper Objects**

* Links to already created objects
* Do not contain elements itself, as it links only to the other objects (this is to prevent cases, when some elements changes in application, and there are processes that uses simple objects and these wrapper objects, so that we would not need to change attributes in multiple objects)
* Can contain business logic.

# **System Settings**

## Credentials

* If application is using same credentials, as Windows, do not create separate credentials for it, use LIVE - WINDOWS PASSWORD – {Robot}

## Environmental Variables

* Process Name – Variable Name

# **Testing and Quality Assurance**

## Change Requests

* Working on new Change Request
  + Create copy, of original Process.
  + Do all changes, related to change request on the copy.
  + When all the changes are done, copy updated version, to original processes (do not change names, just copy pages/stages, to avoid changing ID’s).
  + Delete the copy, that you created.
* Check if Initialize page meets our standards
* VM Restart WO/Garbage Collector/BB Exception handling must be in place.

## Maintenance

* With every new release, Logging should be updated up to our standards.

# **Custom Code Stages**

* Development, and testing, should be done on copy of the object.
* Adding to main object, should be done in a meeting with one of the seniors.
* https://confluence.corpprod.awswuintranet.net/display/BP/%28draft%29+Custom+Code+Stages%3A+Best+Practices