

RPA Analyst Course

Written Guide

Contents

Contents	1
Intent	2
Feedback	2
The Process Definition Document	3
Overview	3
The PDD	3
Title Page	3
The Revision History section	4
The Restricted Distribution section	4
The Introduction section	5
The Overview section	5
Impacted Business Areas	6
The Process Diagram Section	6
Process Detail Section	7
The Exceptions section	8
A Final thought	9
Verification and UAT Plans	10
Overview	10
The Verification Test Plan	10
UAT Test Plan	10
Commonalities with each Plan	11
Introduction and Testing	11
Testing Scope	12
Test Requirements	12
Acceptance Criteria and Action Plan	12
Final Comments	13
Operational Impacts on the Organisation	14
Overview	14
The OID	14

Intent

This document is designed to augment your online learning experience and provide you with the opportunity to review and revise the content that has been discussed during the individual lessons of the course

Feedback

Please contact your WithYouWithMe RPA instructor if you have feedback on this document or any of your WYWM courseware.

The Process Definition Document

Overview

The PDD is the main deliverable of the Define phase and is required by the RPA development team to design the complete end-to-end automation solution.

If you recall the lesson on the Roles and Responsibilities of an RPA Analyst, we did a deep dive into the Define phase and steps that an RPA Analyst must take to complete this document via scheduled workshops and walkthroughs.

In this lesson we will cover the PDD section-by-section and provide you with industry best practices when capturing and building this document. This is information that can be applied directly to industry when you're working as an RPA analyst.

This lesson will focus on the technical aspects of the PDD and will also help with your capstone assignment at the end of this course.

A template of the PDD is provided in addition to this lesson and will also be needed to complete your end of course project.

An example of a PDD that's been used on a project in industry is also attached to this lesson. This is a good example of an industry standard PDD but also highlights that PDDs will differ slightly from project to project, dependent on the organisation's requirements

The PDD

As discussed previously in this course, the Functional Requirements Questionnaire (FRQ) is the document that supports the development of the PDD.

Even though these are two different documents, we are finding more and more that organisations are incorporating the FRQ into the PDD. This of course is dependent on the individual organisations and is not a one size fits all approach, but it's worth noting that this practice has become more prevalent.

That said, for the purposes of this course, we'll keep the documents separate for the ease of discussion and simplicity.

As the PDD is used to capture all the information of the manual process, you can apply this document and the practices of capturing all the information agnostically to all processes regardless of automation software vendor of choice.

Title Page

The title page of the PDD, shown in the slides, provides the unique information of the Client/Customer and the business process name.

The Business process name is just that, it is the manual process that we're capturing the information from and each business process we automate needs its own PDD.

Version control is also important and should be adopted from the very first draft that you create.

The Revision History section

The reversion history section of the PDD is quite simple as you can see in the slides, but it's worth noting the difference between revision history and version control.

The revision history focuses on the process itself and provides the business unit and RPA Team the ability to update the PDD as the process changes after the initial deployment of the automation. This allows all key stakeholders to understand when, why, who and what was changed in each revision.

The Restricted Distribution section

The restricted distribution part of the PDD is focused on a number of areas that relate to the distribution procedures required within the administrative process of this document.

Document Version Control: This provides greater detail of when, what, and who made changes to the PDD. It should provide as much detail as possible of each version change so that if this document had to be handed to another Analyst, then the new Analyst would understand what stage the document was at.

Contributors: Generally, there would be one author of this document. One single RPA Analyst who spends all the time with the process SME to define the manual process. However, there are often times where the process is complex in nature, requiring a team effort, or the original RPA analyst has been reassigned to another process due to a change in priorities. Thus, it is essential to document all contributors to the PDD for tracking, convenience and governance purposes.

Source Documents: At this stage of the delivery methodology and depending on the organisation you work for, the two documents that support the PDD are the IPA and FRQ. Ultimately, the FRQ will end up being the authority as many organisations use the FRQ as the primary source document when defining the process requirements.

Document Sign-off: This is an important part of the PDD. It is where we track who actioned the final stages of the approval process. Focusing on the responsibilities column, this is where we specify what action the reviewer has in the sign off. Some people are provided the PDD for information while others will need to review or approve by signing off.

The Review must be actioned by the Process SME with the final sign off coming from the business unit manager or process owner. Remember to always have an email trail that shows the approval from the review and sign-off stages. This should stay with the PDD at all times.

Document Classifications: This part is very dependent on the organisation or business unit. Some processes are highly sensitive and require specific confidentiality clauses included to ensure security policies are in place. This is particularly applicable in commercially sensitive business, federal government and defence projects.

The Introduction section

The introduction section of the PDD is very generic and will remain largely unchanged irrespective of the process.

This introduction provides a high level overview of the purposes and requirements of the PDD.

Most organisations will tailor this section to suit their requirements in accordance with their own governance policies. However, most times, this will be very minor changes to wording and content.

The Overview section

The PDD Overview looks at two main sections, the Manual Process description and the Target Systems.

The Manual Process Description is a brief overview of the manual process only, and should provide the following detail:

- The purposes of the process
- Source information
- Where the process starts
- Any specific triggers that start the process
- Main steps in the process at a very high level, and
- Where the process ends

This section of the PDD does not need to be more than half a page. It is a brief overview of the end-to-end manual process. We'll go into much more detail later in the PDD.

Target Systems: As shown in the example, a target system is an IT or enterprise application that the process you're defining needs to interact with or access. If the manual process requires only Outlook and Excel, then you would only need to document those two target systems. You will need to document why they are required as it aids the solution designers in understanding the required permissions or even how to structure the solution to be most efficient.

It is essential that all target systems are identified as the bot or digital worker may require further access permission that it does not already have, which will be required for the Solution Design Document.

A good way to consider all target systems is to think of all systems that are involved in the process. Walk through the manual process and write down each IT system or application that is involved as a good starting point.

Impacted Business Areas

This section of the PDD should focus on the business units that will be directly impacted by the process becoming automated. There may be only one business unit impacted or there could be several dependent on the complexity of the process.

This should be no more than a page in length and should contain at least the following points:

- Affected Departments or Business Units and why they are affected.
- Additional training requirements for these business units
- What happens if the process is not actioned in the allocated time, i.e. what are the Service Level Agreements that the business unit is required to maintain.
- What role or position within the business unit will be required to monitor the process.

While these points are key, this is not an exhaustive list and should be tailored specifically to the organisation's requirements and the process you're dealing with.

A good way to think about this is to think about inputs and outputs to your process. An example for inputs would be if a process starts with a spreadsheet in a folder, which team created the spreadsheet? Which team would you speak to if the data in the spreadsheet was incorrect?

Similarly, to find the impacted business units downstream of your process, which team picks up the outputs of your process? Which team is impacted if your process results in errors?

The Process Diagram Section

This section of the PDD is where the RPA Analyst will be required to map the flow of the manual process. In industry, this section requires the Analyst to employ the BPMN standards for business flow.

The BPMN is A Business Process Modelling and Notation Standard and as of 2020, this is BPMN 2.0. A simple search on the web will provide you with a plethora of information and software solutions to produce nicely formatted process diagrams in accordance with the BPMN 2.0 standard.

We recommend that you use MS Visio or Google's own Draw.io Application. If you're already working in an organisation, then you should have access to MS Visio otherwise, Draw.io is free to use and very easy to work with.

For the purpose of this section and this course, you only need an application that makes it easy for you to produce a process diagram. There is no requirement for you to meet BPMN standards.

Basic Symbology

Each Stage of the process diagram will require specific and common symbology. Here we will provide you with basic examples of what is required.

Start stage used to begin the process

End Stages used to end the process flow

Action stage used for steps that require an action e.g. open outlook, download file, copy data etc.

Decision stage used for true / false checks. I.e. Is a condition true or false? Effectively, we are looking for a yes or no answer to progress to the next stage. Is MS Outlook accessible? Is the Managers Report attached to the email? Is there more data in the excel spreadsheet to process? These decision stages will always be questions that need a yes or no or true or false response.

The three main symbols you will need to know to complete the PDD capstone project at the end of this course are displayed in the slides. However, there may be others that you need to include and this will be dependent on the organisation.

Layout

The example of the layout shown in the slides is considered industry standard. The process flow should mimic the example shown here. The swim lanes are the target applications such as MS Excel and Outlook. Each action that occurs for each target application should be presented in the swimlane aligned to that application.

Depending on the complexity of the process and flow from application to application, this diagram may be presented in portrait or landscape. A simple process flow that goes from one application to another and does not return back to the previous application, is best presented in portrait OR down the page.

However, if a process is complex in nature and has a number of steps that move the process flow from one application to another, then it's best practice to use landscape page layout to present the process flow.

Process Detail Section

The Process detail section of the PDD is a detailed, step-by-step explanation of each step of the process diagram in the previous section.

It is a comprehensive overview of each step in the process down to the keystroke level. This section should use images, numbering, arrows and shapes as well as detailed explanations about each part of the stage.

The example shown here is what is expected in the Process Details section of the PDD. This should be the longest section of the PDD. A basic rule is this section can never have enough information or detail. More is always better and will help the solution developer and save time in the build phase of delivery.

Images for each step need to be large and across the entire page and they should be of a high quality and clear.

It's important to highlight key information such as fields, buttons, selections and other areas of importance with red boxes, arrows and numbers for extra clarity. Boxes typically indicate a button, field or menu selection and arrows connect or draw attention to that box. Numbers are used to indicate the chronological steps within the stage and will sit beside the shape or at the end of the arrow identifying which step happens next.

This will ensure that there is no ambiguity in what to press, what to input and in what order this happens, so the solution designer has a clear understanding when they're creating the solution.

The reason we need to go to the keystroke level is that the bot you build is great at following exact instructions, but it's really bad at implied instructions. As an example, writing "Open Google Chrome" is not sufficient as there are several ways you can do this task. A better way to write this task would be:

- Open Start Menu (add image)
- Click in Search bar (add image)
- Type "Chrome"
- Wait to identify if "Google Chrome" application is highlighted (add image)
- Click Google Chrome icon
- Wait to identify if Google Chrome launched
- Maximise Google Chrome window (add image)

The Exceptions section

Exceptions are errors that could occur during the step-by-step process. This is where you ask yourself, "What could go wrong?"

When it comes to the business process that we're looking to automate, exceptions are categorised in two types, business exceptions and system exceptions.

As we're mentioned earlier, depending on the organisation, consultancy firm, head of RPA etc. the inclusion of exceptions in the PDD may or may not be required.

It is essential however that the RPA Analyst understands what they are and how they are adapted to the automation solution.

Business Exception

A Business Exception is essentially any error that may be encountered throughout the execution of the process that is caused by human error; i.e. not following the business rules or procedures. This could be as simple as someone incorrectly structuring an excel spreadsheet and not following the proper naming conventions. It could be as simple as someone not attaching a file to an email when submitting a daily report or incomplete/missing required data in a spreadsheet.

Simply, a business exception is one that is caused by human error or incorrect application of business rules within a process.

System Exception

A system exception is essentially any error encountered that's caused by systems or applications that you need to interact with. Think of the target systems. A system exception may be an email application is inaccessible or doesn't launch, meaning that you have attempted to open the application but it fails to appear and thus, produces an error window. Or, you try to access a folder on the network and it doesn't open or has even disappeared. It could be as simple as the application took too long to load or the information flow was significantly delayed due to the volume of traffic and users on the system.

It is not the responsibility of the RPA Analyst to solve these exceptions but they need to understand what these exceptions are by asking the right questions of the process SME at each step of the process. The Analyst should be asking the 'what if' questions at each stage of the process or even ask the process SME, 'what are the main issues you regularly encounter at this stage of the process?'

Remember, the process SME will fix regular exceptions without it playing out as a significant event in their mind, so they may miss this when explaining the manual process to you.

A Final thought

As an RPA Analyst, it is important to understand why a solutions architect or senior developer may need to know about these exceptions.

The reason is that automation software is not as smart as you would think. You need to tell it how to do every step and how to deal with both business and system exceptions. This is important because we need to tell the bot exactly what to do when it encounters each type of error.

If done correctly, the bot should encounter the exception, identify it as one it has been trained to deal with, take the appropriate steps and continue. If this is not done correctly, the bot will stop or break. In automation, we call these critical steps "Exception Handling". Thus, when you automate a process, best practice is to never retry a business exception, record it as an exception, refer it to a human and then move on to the next item waiting to be processed.

As an example of a business exception: if the bot opens a spreadsheet and the columns are not arranged as the bot is expecting, you may want the bot to stop what it is doing and send an email to the process owner saying the process has stopped and for what reason. If this hasn't been programmed into the automation, then it's possible the bot would mis-identify the columns and collect the wrong data, or simply stop and sit there until someone found it stalled.

With a system exception, an automated process logic may be to repeat the system exception a number of times before terminating the process. As an example, the IT department may be upgrading Microsoft Office in the middle of the night. But our bots can work 24/7, so it's possible that our automation attempts to open Microsoft Word for a process and because MS Office is down, Word doesn't open. This is not a business exception because the bot followed the process and did everything right, but it was the system that caused the error. So you may train the automation to identify Word not opening as a system exception and hence try 2 more times to open it and it still doesn't work, terminate the process and inform the human.

For now, it is just important for you as the analyst to document known exceptions, not to develop the solution to handle the exceptions. Handling exceptions is the job of the RPA developer.

Verification and UAT Plans

Overview

In this lesson we will cover the purpose of each plan and briefly review the UAT Testing Document.

Firstly, there are different approaches to testing when it comes to automation and this is normally influenced by the automation vendors, delivery partners and the organisation itself. However, these approaches to testing will generally have the same acceptance criteria that the automated solution must meet before moving to the next phase in the delivery methodology.

As an RPA Analyst you will be required to liaise with the Process Owner and Process SME to collect the information required to create these testing documents. As we discussed in the Delivery Methodology earlier in this course, while the automated solution is in the build phase, the RPA Analyst along with the process SME will be building out a range of scenarios to test the bot's robustness.

The Verification Test Plan

The Verification Test Plan is a document that provides an outline of how testing will be performed during the Verification Phase of the process configuration.

The verification phase forms part of our configuration (or the build phase in our Process Development Life Cycle), and therefore the automated solution will still be incomplete. The purpose of configuration is twofold:

- Firstly it allows new screens to be configured which can only be reached using live data (i.e.. the "Account Closed" screen can only be seen after closing an account).
- Secondly, it allows a large variety of business scenarios to be tested so that confidence can be established in the solution prior to the User Acceptance Testing (UAT).

Because scenarios are being verified (and in some cases configured) for the very first time, the solution is not considered "ready" – the verification phase is the means by which our solution will arrive at this 'ready' state.

UAT Test Plan

The User Acceptance Test Plan is a document that provides:

- A minimal set of acceptance criteria for the solution to be considered ready for approval and sign-off
- A practical framework outlining the method and scope of the solution testing

The content should be primarily produced by the Process SME with guidance from the RPA Project team where required, specifically, the RPA Analyst.

Acceptance testing takes place against the backdrop of a largely complete automated solution. A previously agreed set of scenarios will have been thoroughly tested as part of the verification phase, meaning that the main goals of Acceptance Testing will be to:

- Verify that the solution functions in the production environment
- Ensure that the solution performs as expected, in terms of
 - Reliability
 - Ability to handle volume
 - Consistency and accuracy across a large data set
 - Speed, where targets were agreed at design time
- At this point we also need to Cover off any remaining scenarios not processed during the verification phase

Acceptance Testing takes place in the Acceptance Test (Pre-Production) Environment, where available. If a pre-production environment is not available the Live (or Production) Environment can be used, but this comes with increased risks to business continuity in the event the bot fails its testing. This is also influenced by the access requirements to some of the enterprise applications.

Commonalities with each Plan

Both the Verification and UAT documents cover many of the same requirements and criteria to undertake process testing.

For the purpose of this lesson, we'll focus on the UAT Test Plan. In some organisations, the UAT is the only testing plan adopted and can be adapted for verification testing. This would come down to the organisation's internal delivery methodology and governance.

In this part of the lesson, we will focus on the following areas of the Test Plan that include:

- Testing Scope
- Test Requirements
- Acceptance Criteria, and
- The Action Plan.

By stepping through the Test Plan sections, we can learn about the steps and considerations for this phase.

Introduction and Testing

The Introduction and Testing Overview are both general in nature and like the PDD, don't require any changes. These sections are more focused on providing the business process owner and SME an understanding of the test plan.

Like with the PDD, the Test Plan Documents follow the same version control and revision history conventions. Thus we will not cover that here in this lesson.

Testing Scope

As detailed here in this document, the Testing Scope covers sections on Untested Scenarios, Business Scenarios and Controlled failure Scenarios.

Untested Scenarios are simply scenarios that could not be tested for a number of reasons.

When rare scenarios could not be tested during the Verification Phase, a decision needs to be made as to whether to allow these scenarios to run, or whether to mark them as exceptions. An example might be any exception where the whole system is down and the bot gets a “System Unavailable” error message. Obviously it is difficult to safely replicate a scenario where the whole system is down so a decision would need to be made on how to handle this in the UAT test plan.

Business Scenarios should be focused on specific functions that occur throughout the process. These could be as simple as opening and closing applications, uploading data from a spreadsheet or sending group emails out to internal staff. There is no limit to scenarios documented here. It will all depend on the specific requirements of the Business Unit.

Each heading contains a list of scenarios mandated by the business unit. Each one is to be tested thoroughly according to the notes provided. The outcome will be verified by the business unit manager of process SME.

Controlled Failure Scenarios are used to test known exceptions that will cause failure of the process or possible termination. As with the business scenarios, they should be developed and verified by the business unit manager or process SME.

Test Requirements

In this part of the Test Plan, we look at what testing requirements are needed to conduct the UAT. This section looks at:

- What case data will be used and how it will be selected,
- Who will be the Subject Matter Expert, normally this is the Process SME,
- Additional contact details of people associated with the testing,
- Are there any restriction or policies that need to be adhered to during testing,
- Is there a requirement to set up and modify virtual environments?
- The method and timing that may cause unforeseen complications, and
- Any other additional requirements not captured in the previous sections.

Acceptance Criteria and Action Plan

In this part of the lesson, we'll look at the last two sections of the Test Plan; the Acceptance Criteria and the Action Plan.

When we think of the Acceptance Criteria, we think of criteria that should be agreed and defined by the business unit manager and process SME.

This will usually include:

- Providing evidence of the bot's successful operation against the test scenarios we developed earlier.
- There should be evidence to demonstrate reliable operation for a continuous period days
- There should be a system-exception rate of 5% or lower for all cases run. As business exceptions represent designed and documented behaviour – i.e. correct operation by the automated process – these exceptions do not contribute towards the system exception total.
- There should be criteria that provide metrics for acceptable performance against speed-based performance targets, process volume limitation, triggers etc.

The Action Plan should detail any items, issues or further requirements that need actioning prior to acceptance.

This could be simple modifications or inclusions to the process to enhance robustness.

There should always be someone tasked with the action item as well as the deadline for completion.

Final Comments

Remember that as an RPA Analyst, it is important to understand the requirements of this plan so that you can work with the business unit or Process SME to assist in completing the required documents.

Remember, each organisation will have their own policies and procedures for Acceptance Testing, so ensure you understand them before before completing these documents

Operational Impacts on the Organisation

Overview

This lesson will briefly cover the operational impact Document (OID) and its requirement within the Delivery Methodology.

This document is used to describe in detail the impact of RPA on the Operation of the business, following the delivery of our automation solution.

This document should be completed once the PDD is complete and the initial solution design has been finalised. Depending on the organisation, the OID is usually completed for each automated process solution and can be included in the Implementation plan and operational handbook.

The OID

The intent of the Operational Impact Document is to make the organisation aware of the operational impacts of the bot prior to its deployment. It also annotates what the automation solution needs from the business.

The OID covers:

- A Brief Process Overview - Which can be taken from the PDD once completed.
- Resource Requirements - These may include Operational and IT Resourcing requirements
- Data Storage location and details such as email accounts, access to network drives, credentials etc.
- Exceptions and referrals. Specifically, Who, What, How and when will exceptions be handled.
- Assisted Automation Protocol, which focuses on how the process will run in the production environment. For example, Will it require specific triggers etc.
- Finally, we have management information. This should contain all the required information and reporting required by management, storage locations, schedule etc.

The purpose of the OID is to introduce the business unit to the automated solution and to provide an overview of the bot's work habits. By this we mean we that we describe the digital worker in business terminology, and provide a summary of the assumptions and preconditions for the process, so the business process owner and the RPA delivery team are on the same page and can be ready to integrate the digital worker's outputs into a wider process. As we've discussed, these outputs may include alerts, exceptions, data handling, etc.

Even though the OID does not fall within the scope of the Define Phase, it is very useful for the RPA Analyst to understand the intent of this document and to better understand how to prepare specific questions when engaging the business unit,

A template of the OID is provided after this lesson and you should take the time to review this before moving on in this lesson.