# Training Academy

## Error Handling and Automation Lifecycle

reveal group

## Course Path

Introduction to Intelligent Automation and UiPath Overview

User Interface Automation and Project Notebook

Microsoft Office Automation

Automation Bootcamp and Intelligent Automaton Demo

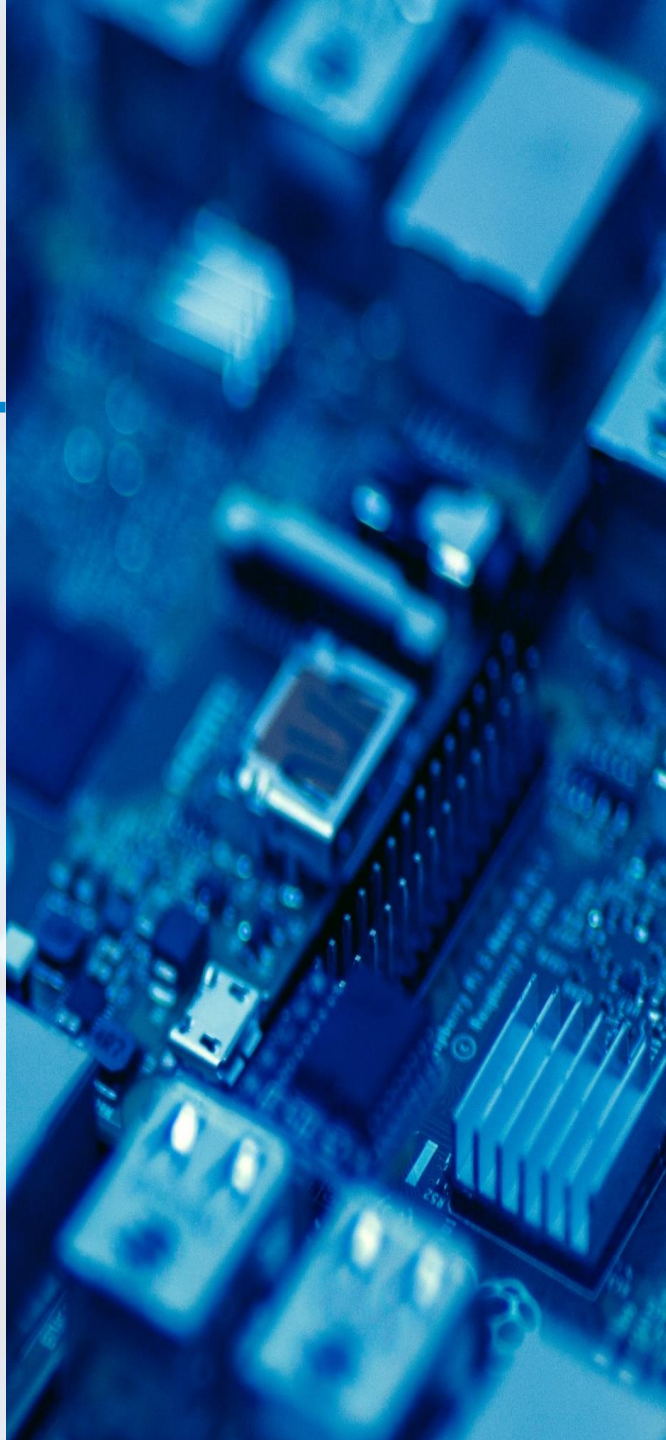
StudioX and Planning Your Automation

Decisions, Iterations and Scenarios with StudioX

**Error Handling and Automation Lifecycle**

# Agenda

1. Recap of Module 5
2. Error Handling
3. Running StudioX Automation
4. Citizen Delivery Lifecycle
5. Intelligent Document Processing
6. Other RPA Platforms: Blue Prism, Automation Anywhere, Power Automate
7. Spotlight: Operate



# Topic 1

## Recap of Module 5

**As a business user, you most likely work a lot with Excel as it is a versatile tool that allows you to organize your data!**

## What is Excel Automation?

Excel automation refers to task automation related to your Excel files. StudioX allows you to build automation projects involving various types of data from an Excel file through a number of dedicated activities.

## How do I use it?

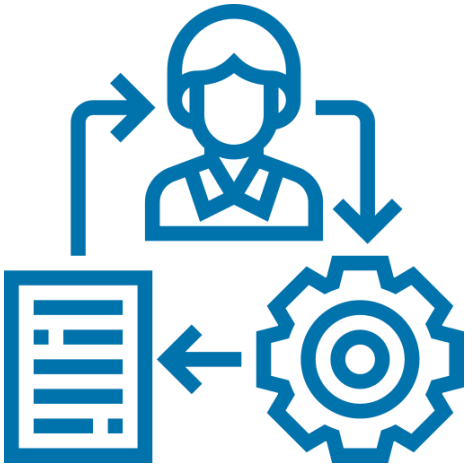
When creating a new project that involves Excel, you have to make use of the Use Excel File resource. Inside it, an Excel file must be specified so that all the actions added afterwards can access the data within that file. When configuring actions, you can select sheets, tables, named ranges, and named cells from that file directly from StudioX, or, if needed, indicate them in Excel.

**As a business or regular user you need to keep your data organized in such a manner that whenever you need something, it has to be in the right place. Sorting through your files and folders is always a time-consuming endeavor!**

## Working with Files and Folders

When we talk about keeping things in order, organizing and sorting through files and folders has always been something that's needed in both business and personal life. Actions like creating, copying or renaming files and folders which are part of our everyday tasks can now be automated through a series of simple steps so that you don't have to invest large amounts of time into doing this anymore.

File automation refers to the automation of tasks the user performs with files and folders on the computer, such as creating, copying, or renaming files and folders.



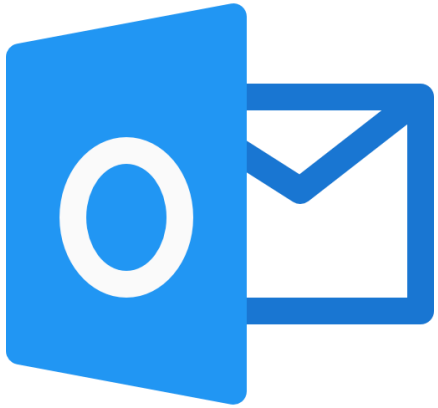
### How do I use it?

In the File activity menu, you will find similar actions for both files and folders. You will not need a resource to apply file actions, unless you use, for example, information from an Excel file. In this case, you will need an Excel resource.

**As a business user, you are sending or receiving email messages, working with specific emailing groups, or downloading certain attachments on a daily basis!**

## What is Outlook Automation?

Outlook automation refers to task automation related to your email account. You can automate minor tasks involving your email account that have a regular cadence or those that are part of larger, more complex automation projects.



## How do I use it?

When creating a new project that involves Outlook, you have to make use of the Use Outlook Account resource.

As the automation project progresses, you can, of course, combine different resources to achieve your email automation goal. For example, using information from an Excel File to create an emailing list, or to just add context to the email's subject. Just don't forget that for values to travel from one resource to another they need to be nested together.

## As a business user, you most likely work a lot with Word Templates and Word Documents

### What is Word Automation?

Word automation refers to task automation related to your MS Word application. StudioX allows you to build automation projects involving a limited set actions in Word such as extracting text, using word templates, and inserting data tables.

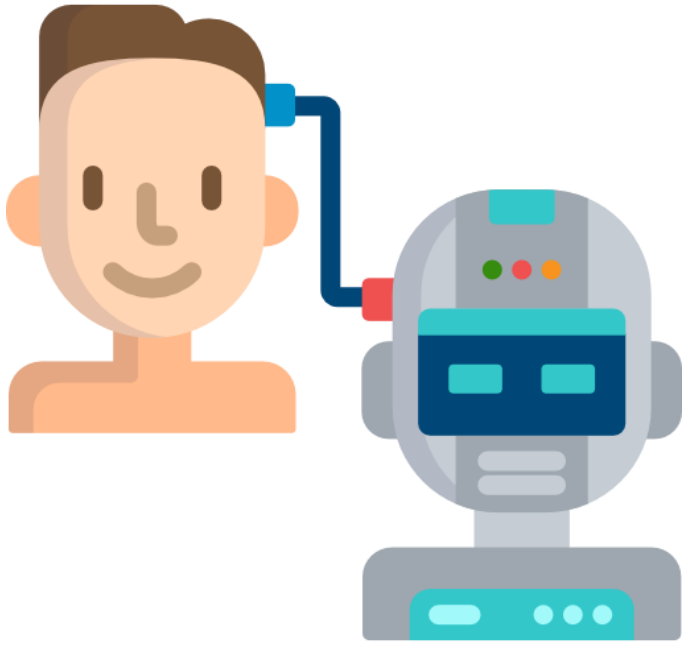
### How do I use it?

When creating a new project that involves MS Word, you have to make use of the Use Word File resource. Inside it, an MS Word file may be specified so that all the actions added afterwards can access and use that Word document. When configuring actions, find and replace text, add pictures, and add data tables for that Word document directly from StudioX.

## Discussion:

- How would you start your Automation Journey?

- What ideas can we try to jump start Automation discussions in organizations new to Automation?



# Topic 2

## Error Handling

## As you continue your automation journey, you will soon find out that handling errors plays an important role in your automation journey!

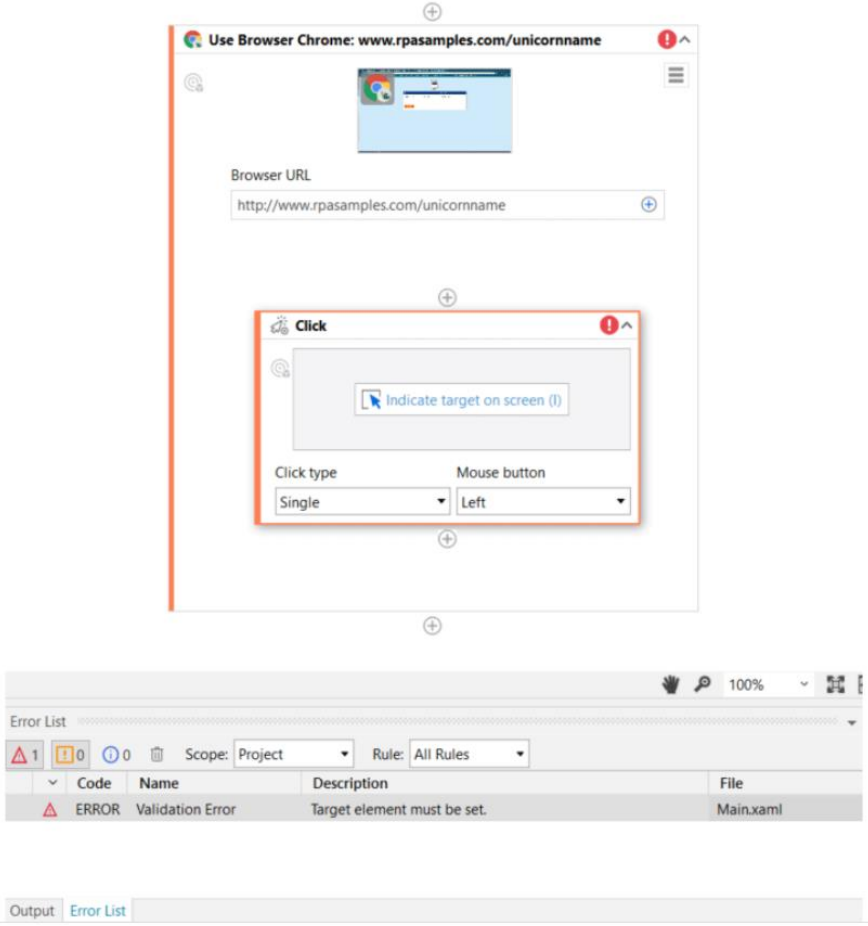
Learning to automate your tasks can be a frustrating endeavor because you are destined to encounter errors along the way. What will make you successful isn't avoiding errors—no one can avoid them. Instead, you should understand that errors are part of the process and know how to find the solution to each while learning something new from them.

### How to start?

StudioX comes with two tools that will help you anticipate some errors. These are called **Validate** and **Analyze** and are both found under the **Analyze** dropdown.

# Validate

The Validate tool will find all the activities with validation errors. These are usually incomplete activities. You can easily spot them as having the red icon in the upper-right corner. You cannot run or publish a project until all the validation errors are solved.



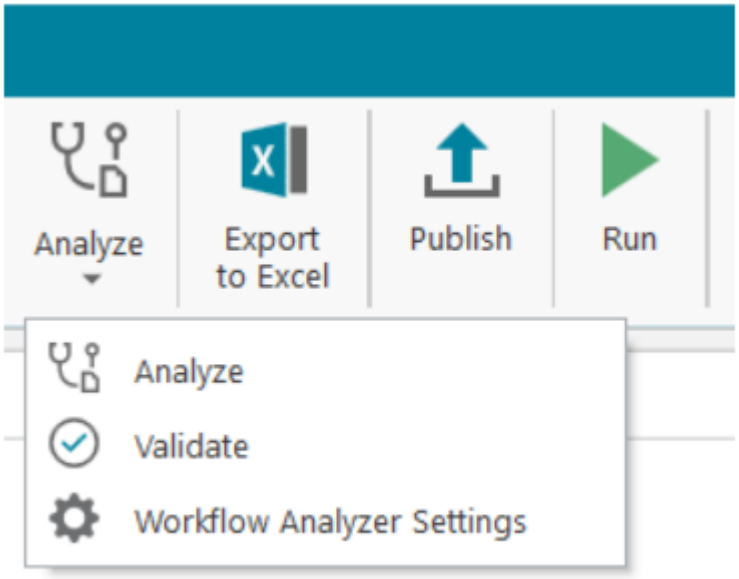

As you can notice in the screenshot on the left, both the resource and the action have a red icon, even though the resource has all the information provided.

This informs you of the resources you have to pay attention to.

Inside the Error List panel, you have information about the error. It says that the "target element must be set". Double-clicking on it will take you to the activity with validation errors, and in this example, you can see that the target for the Click action was not specified.

## Analyze

By the time you reached this course, you may have wondered what are some best practices when building task automation. How can you build robots in a manner that ensures reliability and quality? While there is indeed a set of best practices, we recommend you follow (which you will discover later in the course), StudioX also comes with a set of built-in best practices and analyzes the project against them.



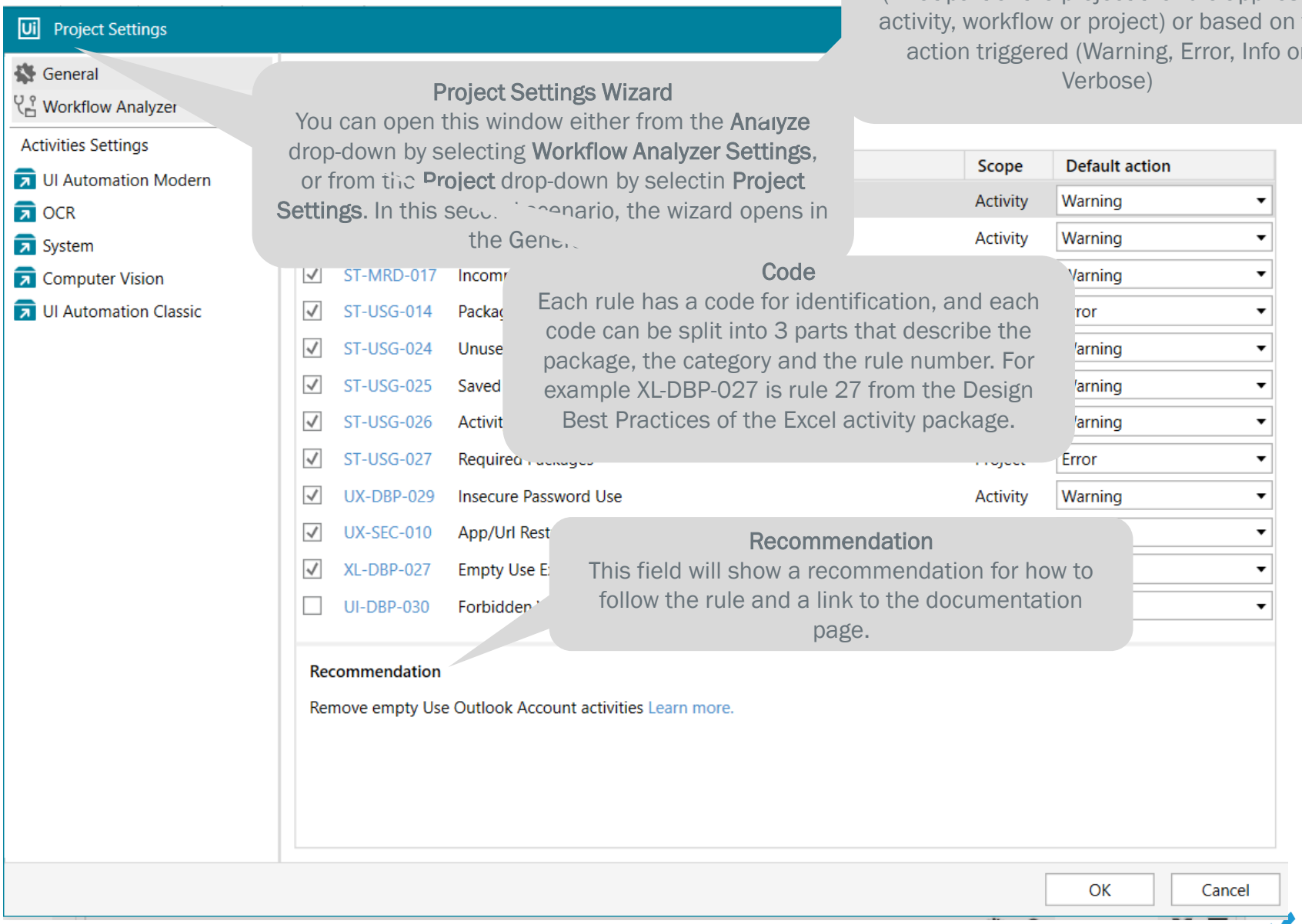

You can access the Workflow Analyzer options by clicking the Analyze drop-down menu from the Design ribbon tab.

Here you can find:

- The **Analyze** option, that checks the workflow against a set of predefined rules;
- The **Workflow Analyzer Settings** option, that allows you to manage, enable, or disable rules.

StudioX comes with a set of built-in rules, enabled by default, that check the activities, the workflow, and the whole project. To examine them, select the **Workflow Analyzer Settings** option.

# Workflow Analyzer Settings Window




**Project Settings Wizard**
You can open this window either from the **Analyze** drop-down by selecting **Workflow Analyzer Settings**, or from the **Project** drop-down by selectin **Project Settings**. In this sec… scenario, the wizard opens in the Gene…

**Scope and Action Filters**
You can filter the rules based on their scope (what part of the project the rule applies to: activity, workflow or project) or based on the action triggered (Warning, Error, Info or Verbose)

**Code**
Each rule has a code for identification, and each code can be split into 3 parts that describe the package, the category and the rule number. For example XL-DBP-027 is rule 27 from the Design Best Practices of the Excel activity package.

**Recommendation**
This field will show a recommendation for how to follow the rule and a link to the documentation page.

## Typical Rules

1. Some of these rules, like "**Activity Restrictions**", "**Package Restrictions**" and "**App/Url Restrictions**" are managed by your organization and they enforce limitations on the activities, websites, apps, and packages you're using in the automation.

2. "**Empty Use Excel File**" and "**Empty Use Outlook Account**" check if you provided an action to use with the resource. Similarly, "**Incomplete IF**" checks if at least one branch (the Then branch is recommended) contains activities.

3. There are also rules that look at the values used. If you **Save For Later Use** and don't use the value, the Analyzer will throw a warning. You will also get a warning if you use the saved For Later Use value in an activity that is placed before the activity that saves that value.

4. Because StudioX projects can also be open in Studio, there is a rule that checks for Studio best practices, "**Undefined Output Properties**". This rule checks whether output properties for certain activities are declared and used. In Studio, variables are activity outputs.

## Bad Unicorn Name Project

When triggered, the Workflow Analyzer uses the configured rule set to check the project and then logs the found errors in the Error List and Output panels, in accordance with the rule action (Error, Warning, Info or Verbose).

An important thing to keep in mind is that before analyzing the project, StudioX will first validate the project. Until all validation errors are resolved, you cannot analyze the file.

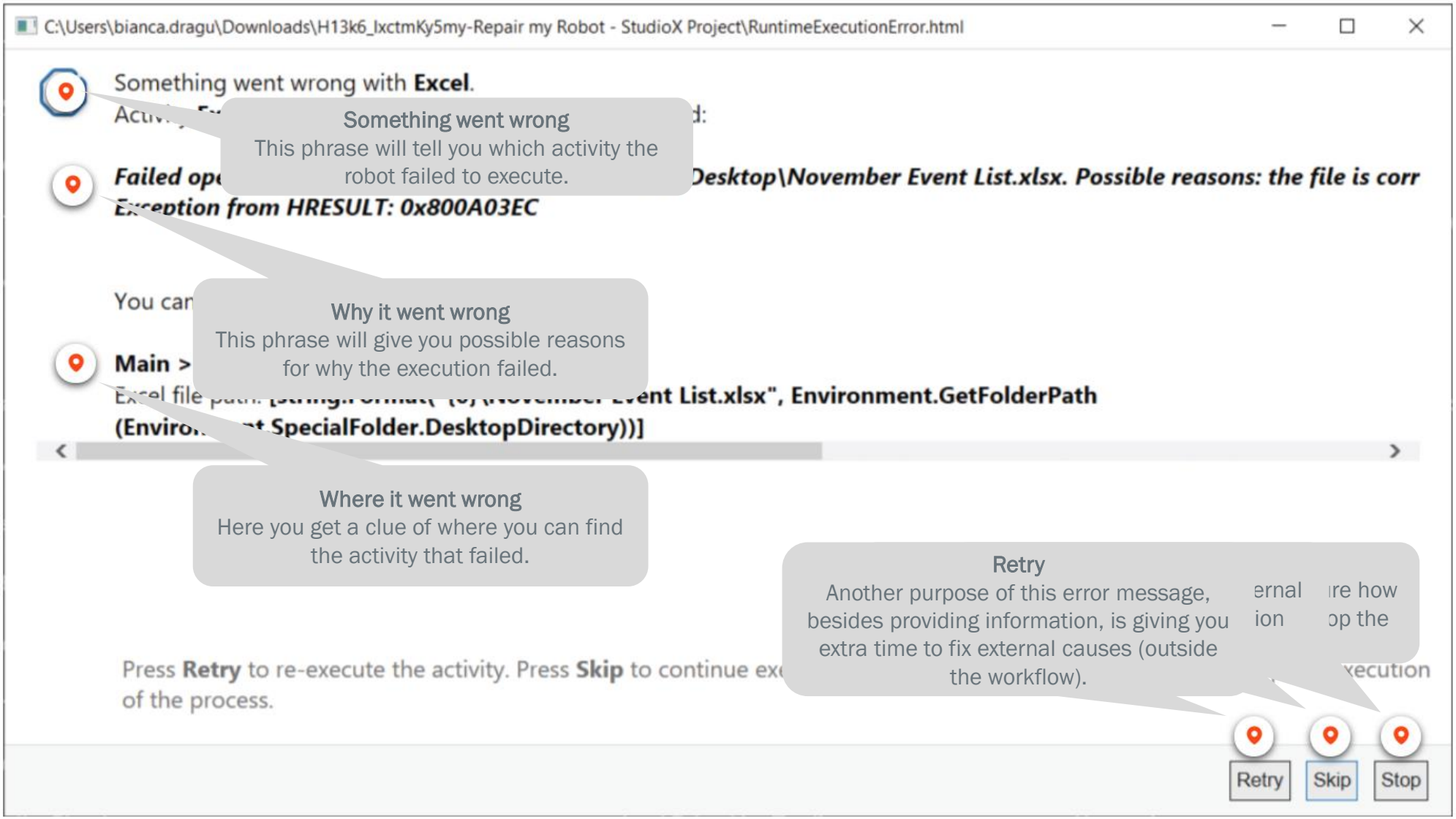
Let's have a look at a badly built "Find your Unicorn Name" project. It's supposed to save the value For Later Use and display it in a Message Box.

**Step 1:** First, download the .zip file, extract the documents, and open the project.

**Step 2:** Answer the following questions:

1) Start with Analyzing the workflow. What happens in the Error List panel?
2) Now double-click on the error. What activity has thrown the error?
3) With that error solved, it's time to Analyze the workflow once more. What happens in the Error List panel?
4) If you select Analyze again, the Error List should be empty. Now run the project! Did it work?

# How do I read an error message?

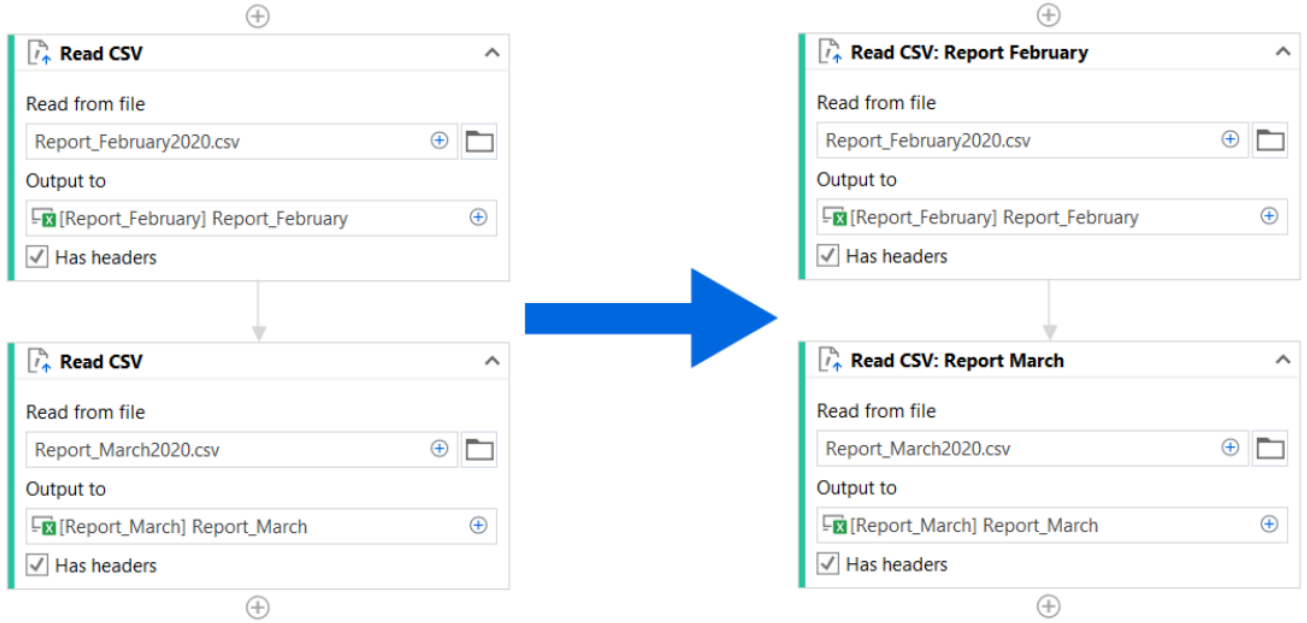


# Best Practices

1. Start with creating the **Robot Path** for your task. Breaking it down into a sequence of steps will help you identify easier the actions and the resources you need to use.

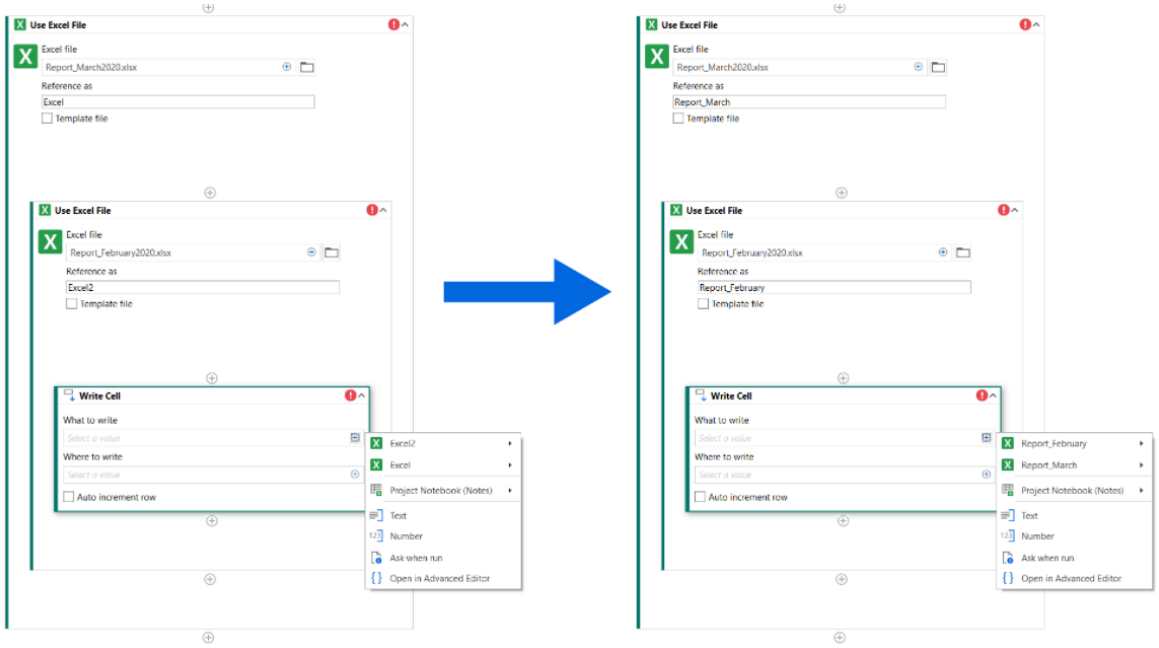
2. **Rename Activities** in order to have more specific names for them or do not use the same name for multiple activities. This will also help you identify easier the problematic activity in an Error Message.

   i.e.: One 'Read CSV' action can be **renamed** in order to specify the file that should be read. This way, when using the output for later use it will be easier to identify the content.

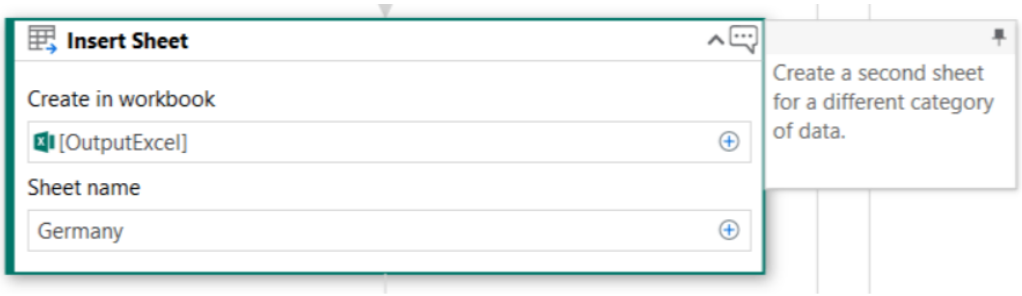


# Best Practices (continued 1)

3 For all the Activities that have the option of ‘Reference as’ (such as Use Excel File resource, Use Outlook Account resource, etc.) use **suggestive naming.**




4 Use **Annotations** for Activities in order to add more information for documentation purposes. This is especially helpful when sharing a project with another person.



## Best Practices (continued 2)

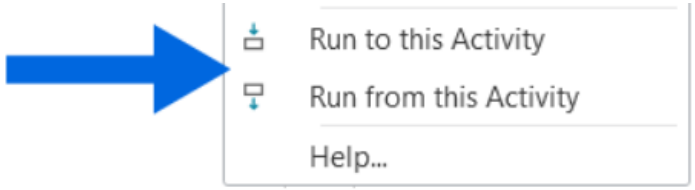
5. **Close all Excel applications** before running a project.

6. Use the local **Project Notebook** for any Data Manipulation, but do not save any values in it.

7. You should constantly check for the **input data** to be correct. At the beginning of the process, and also in the development phase.

8. Use "**Message Box" Activities** when creating a Task Automation before and after each important step/block of your automation. (i.e. write a certain message such as "Started reading file").

9. **As you build your project, test small parts of it often**. There are actions that do not overwrite, for example, **Create Pivot Table**, which means that a second run might result in an error. Make use of the **Run to this Activity**, to run the project from the start and stop before a specific activity, or **Run from this Activity** to run the project starting with a specific activity. You can find these options in the Context Menu (right-click on the activity).



## How to Troubleshoot your Robot Exercise

**Step 1:** Open the project and click Run

**Step 2:** Review the error message and try to solve the issue with use Excel file resource

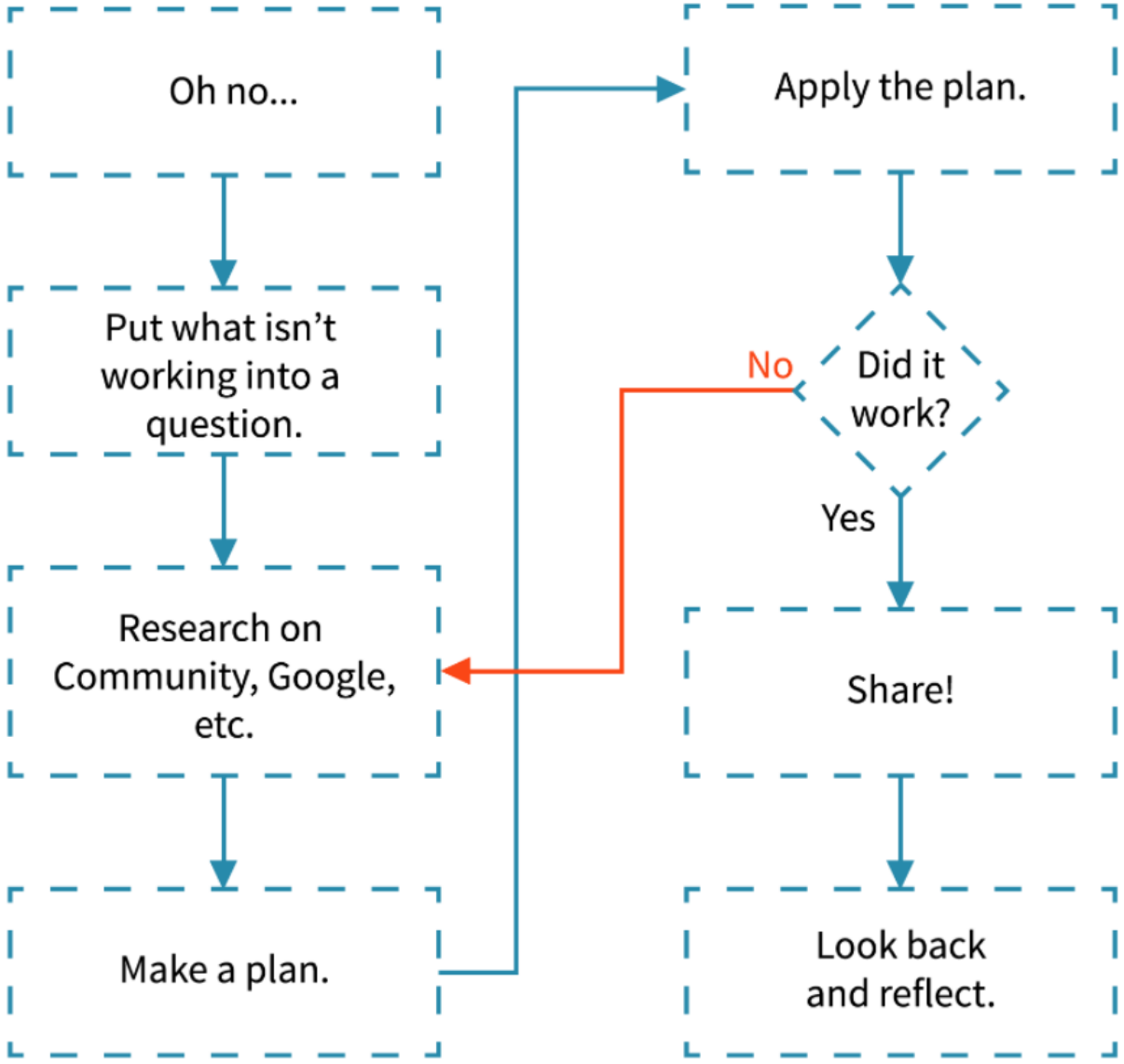
**Step 3:** Review both excel files to see if they were updated with the Unicorn names

**Step 4:** Look at the filter within the For Each File in Folder activity and try to solve the issue

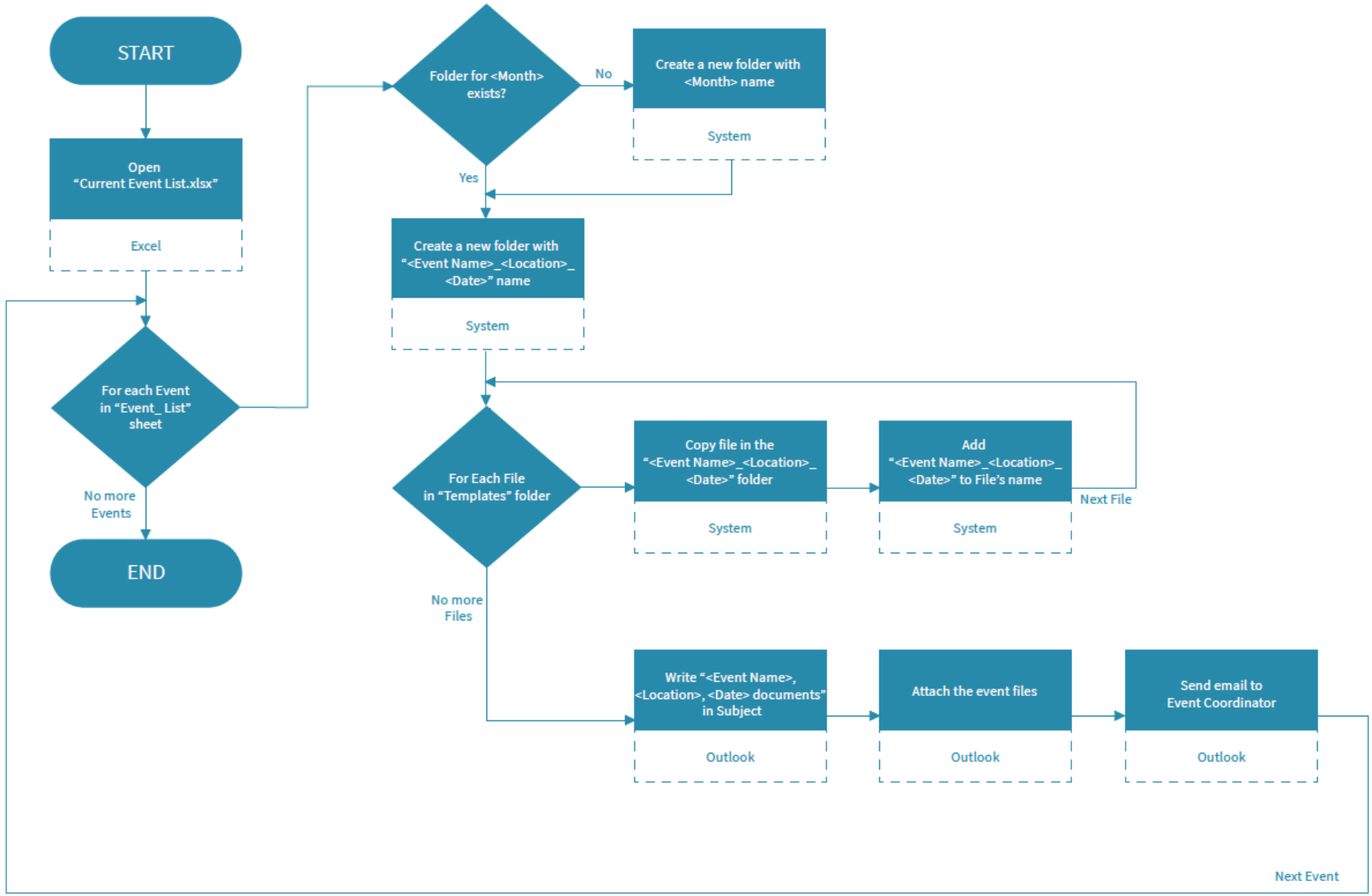
**Step 5:** Review the files again and see if they were updated

**Step 6:** Try to solve the issue by looking at the use For Each Excel Row properties on how to deal with empty rows

## How to search for solutions



# RobotPath: Event Manager



## Practice: Fixing the Event Manager Process

**Step 1:** Open the Current Event List Excel file.

**Step 2:** Second, you need to check if in the Events folder from My Documents you already have folders created for each month from the Excel file. If not, you need to create them.

**Step 3:** Third, inside each month's folder, you need to create a folder for each event taking place.

**Step 4:** After creating the event's folder, you need to create the event's documents by copying the templates and renaming them.

**Step 5:** Lastly, after the documents are created, you need to send them to the email address assigned to each event and include the relevant information.

Above are the steps to perform the automation. Check the files below and see if it works:

- The Robot Path, that documents the automation
- Working Documents
- The zip folder with the StudioX project

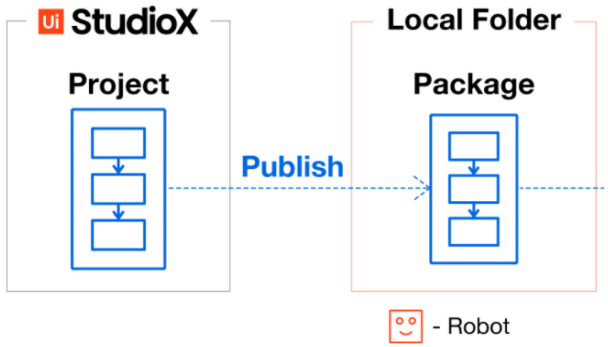
# Topic 3

## Publishing and Running StudioX Automation

# How do I share my automation?

Once you've completed and tested your automation, there might be team members that find this useful. The way to do this is Publishing your automation for everyone to use.

Remember to Validate, Analyze your automation to fix any errors. Otherwise you won't be able to Publish.



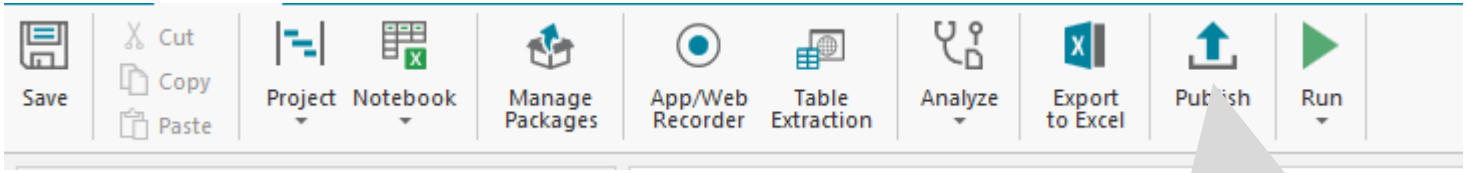

## How to start?

After completing your automation, you click the Publish button at the top.

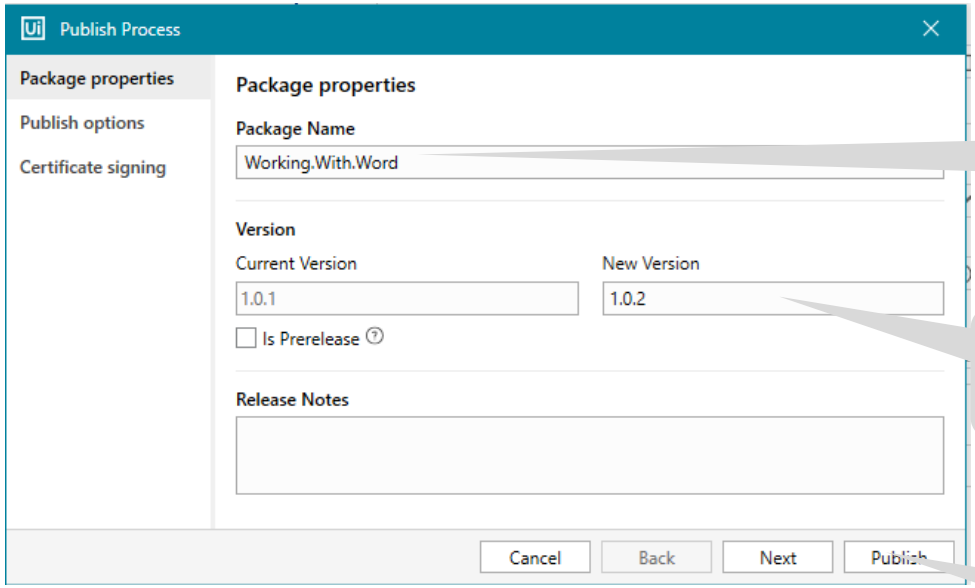

Follow the steps to confirm the upload to Orchestrator

Someone with access to Orchestrator will make it available afterwards

# Publishing Processes (Part 1)




Click Publish



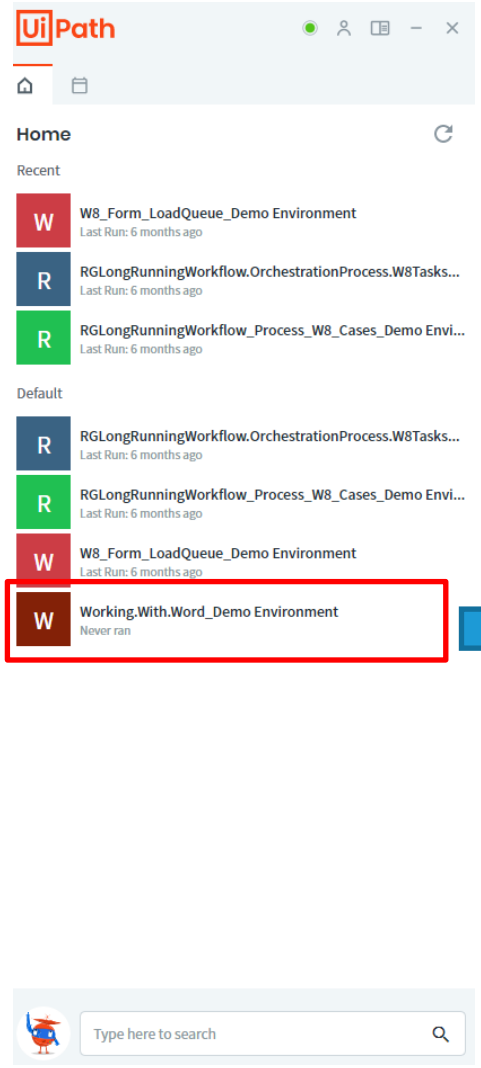

Insert the Package Name

Insert Version Number

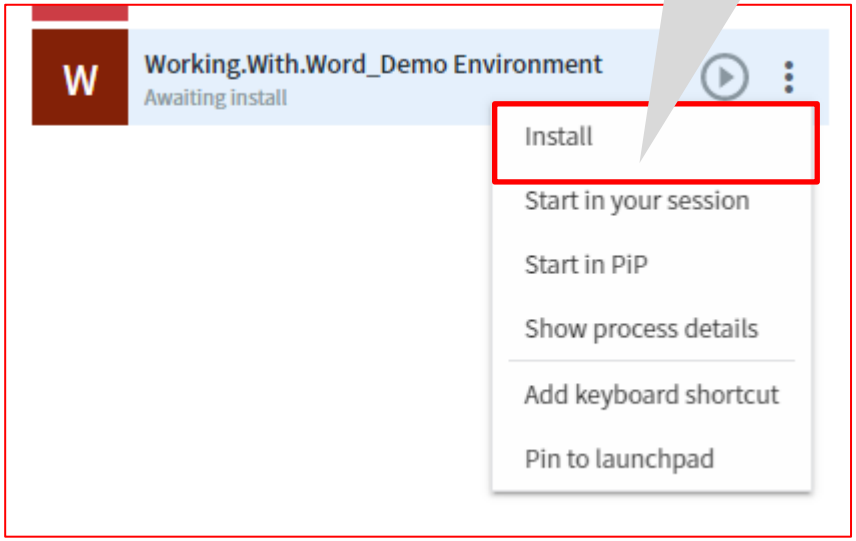
Click Publish

*Assistant (Default) in Publish Option will publish to your local drive*

# Publishing Processes (Part 3)



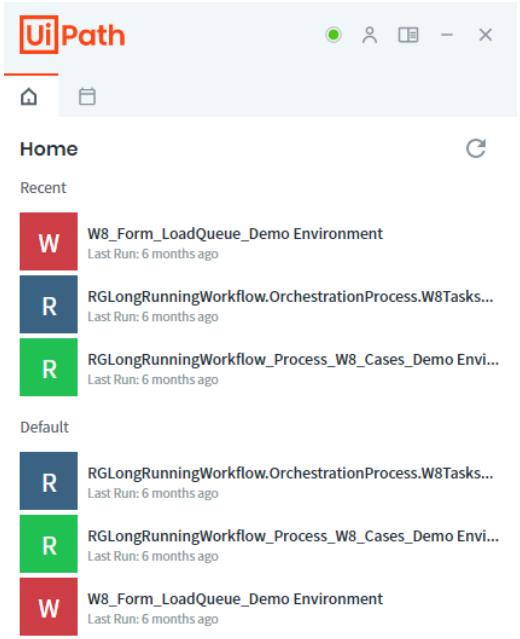

Click Install, to have it available to Run



# How do I run my automation?

You will notice we have been opening StudioX and running it from there. That's pretty tedious to keep doing every day.

After publishing your automation, becomes available to run through the Robot Assistant that can be found in your Windows tray. This will make it easier to just pop up this menu and run which ever automation you wish.




## How to start?

Go to the Window's tray and select UiPath Assistant.

Click on the new Process and make sure it is installed.

Click the Icon to run.

# Topic 4

## Citizen Delivery Lifecycle

# Recap: Standard RPA Delivery methodology (Module 4)

## Delivery Framework

**Deliver**

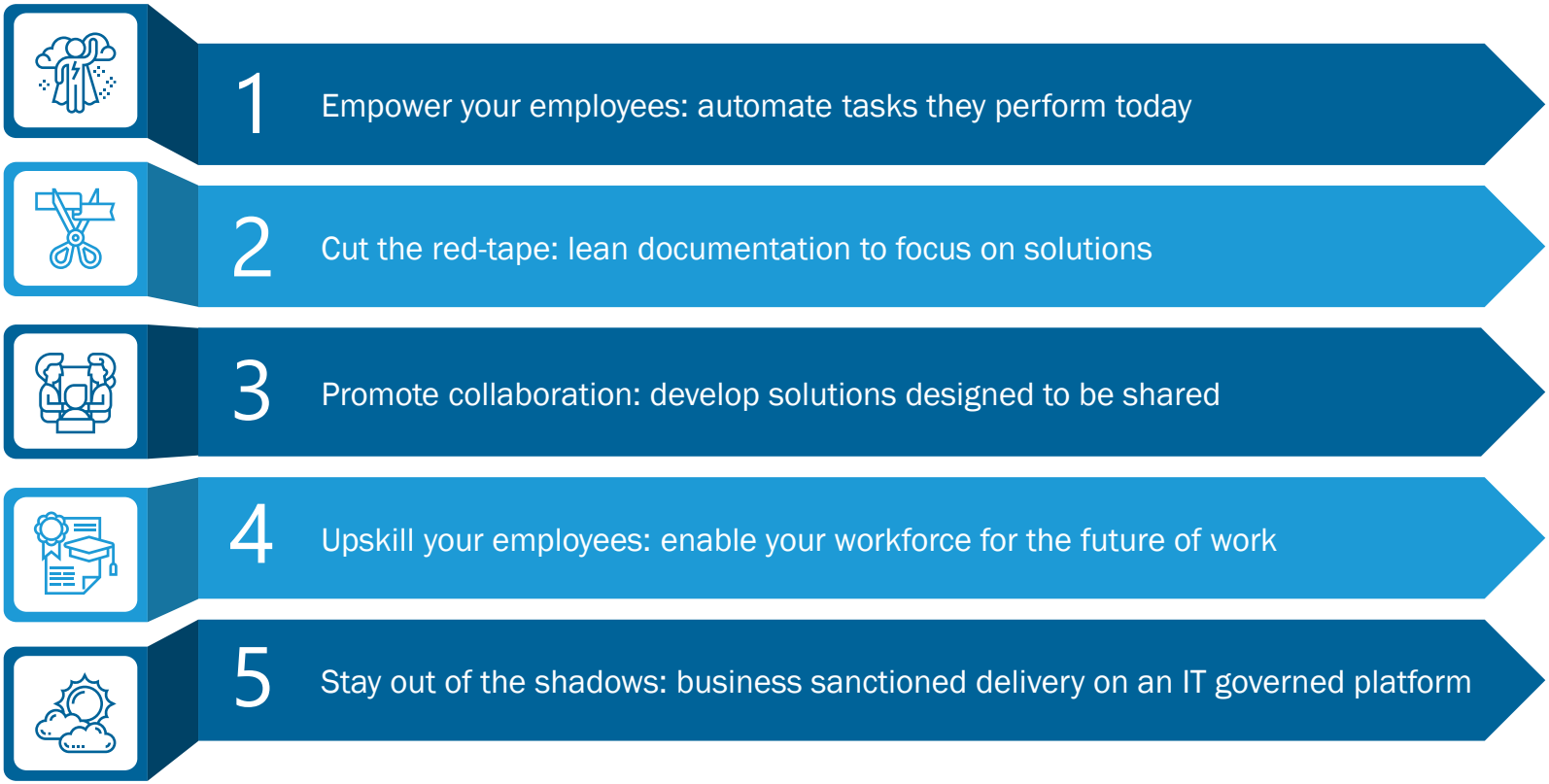
| 1 Define ~1 week | 2 Design ~1 week | 3 Build & Test* ~2-4 weeks | 4 UAT ~1-2 weeks | 5 Deploy ~1-2 weeks |
|---|---|---|---|---|
| • Conduct deep dives to develop a key-stroke level process map of the current state process<br>• Identify variation in the process (business exceptions) and how to address them<br>• Establish baseline metrics and agree on performance targets<br>• Engage and raise system access requests<br>• Confirm project plan in RoboManager | • Outline how the process detailed in the PDD will be automated following Solution Design Principles<br>• Map required automation components and process flow and identify opportunities for code reuse in the solution<br>• Align with the business on the roles and responsibilities once the automation is in place<br>• Ensure application owner engagement | • Develop the automated solution as per the Development Best Practice<br>• Conduct iterative configuration and testing of each component<br>• Conduct Verification Testing of the automated solution in the Build environment | • Conduct User Acceptance Testing (UAT) of the automated solution per UAT Plan<br>• Increase volumes to the expected daily workload under the supervision of the business | • Release the automated process into the Run Environment<br>• Increase the volumes to the expected daily workload under the supervision of the business<br>• Hand the process over to the RPA Operations Team<br>• Hypercare period to ensure a smooth transition to BAU |

**Typically 6 – 10 weeks**

**Modelers adopt agile-like approach of iterative build and test when configuring the automation*

# Citizen Developer Framework

## Enables the five key principles of Citizen Development

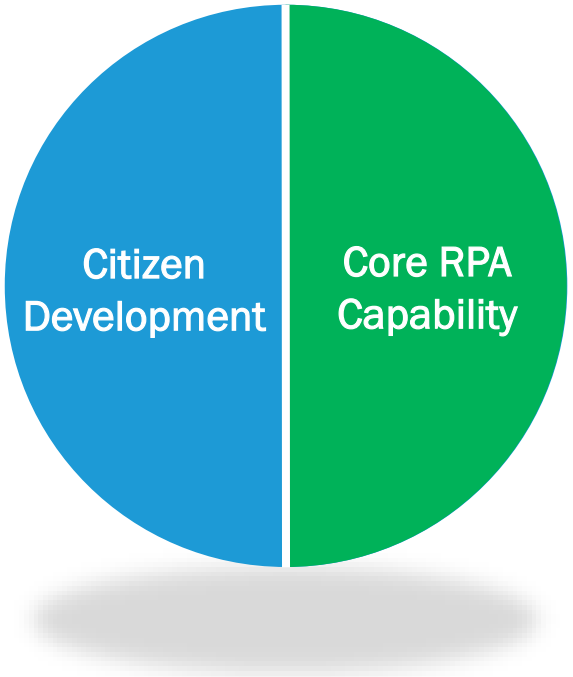


## The most successful Intelligent Automation programs utilize both citizen-led and a dedicated development team to promote a cultural shift towards operational excellence through automation

Citizen Development vs. Core RPA Capability

### Citizen Developer Lifecycle

- Task based automation
- Automation built by the person who performs the process
- Manually triggered and runs on the desktop
- Linear sequence of actions with manual intervention
- Task performed in 1-2 applications

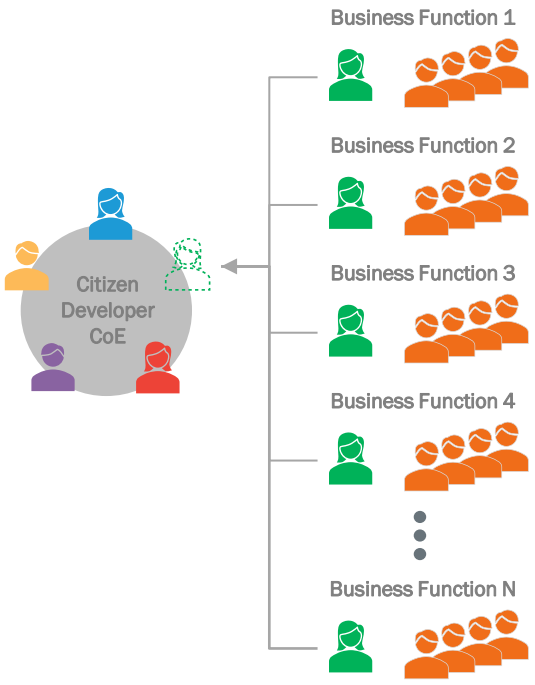



### Standard RPA Delivery methodology

- End-to-end process automation
- Automation built by a dedicated delivery team
- Event driven or scheduled and runs without monitoring
- Handles multiple decisions points and referral handling
- Process utilizes multiple applications

# An effective Citizen Development model requires a centralized governance structure which enables each business function to create and share automated solutions

## Roles and Responsibilities




**Automation Lead**

- Oversees the Citizen Developer CoE and capability
- Facilitate growth of the Citizen Developer program across all business functions
- Report on RPA outcomes to key stakeholders

**Project Coordinator**

- Works with the CoE and Functional Champions to schedule required meetings
- Captures and communicates risk and issues
- Facilitates status reporting

**Tech. Platform Representative**

- Maintenance and monitoring of RPA hardware and software
- Deployment and go-live support
- Ensure logging and archiving policy control and testing

**Design Authority**

- Approves the automation opportunities being developed (technical perspective)
- Review the future-state automation designs
- Reviews completed automations prior to sharing

**Functional Champion**

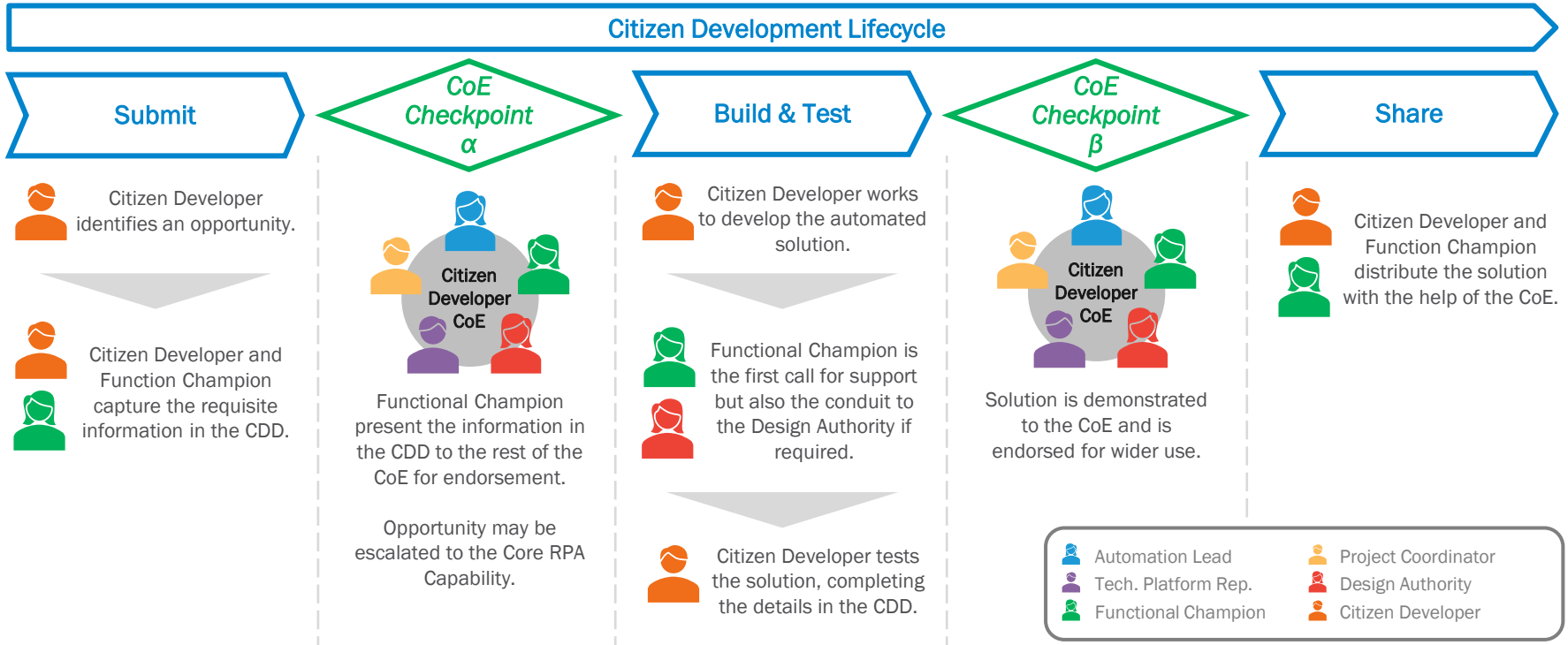
- Approves the automation opportunities being developed (business perspective)
- Manages the communications for sharing automated solutions with the business function

**Citizen Developer**

- Development of self-identified automation opportunities
- Produce the required documentation including testing and deployment plans
- Shares automated solutions with the business function

# The automation of each opportunity should follow the defined Citizen Development Lifecycle including touchpoints with the Functional Champions and CoE checkpoints

## Citizen Developer Lifecycle



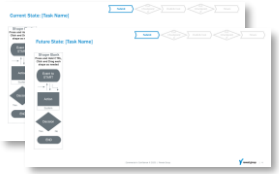
# The Citizen Delivery Document (CDD) is a single PowerPoint document and is used for all stages of the lifecycle from information gathering to distributing a completed solution

## Citizen Delivery Document

**Citizen Development Lifecycle**

**Submit**

Task details, business metrics and data structure



Current state and future state mapping

**CoE Checkpoint α**

Value calculation, implementation effort and checkpoint outcomes

**Build & Test**

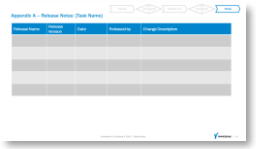
Testing approach and outcomes, scope definitions

**CoE Checkpoint β**

Development outcomes and CoE endorsement

**Share**

Automation details, how-to guide and support information



Appendix: release notes

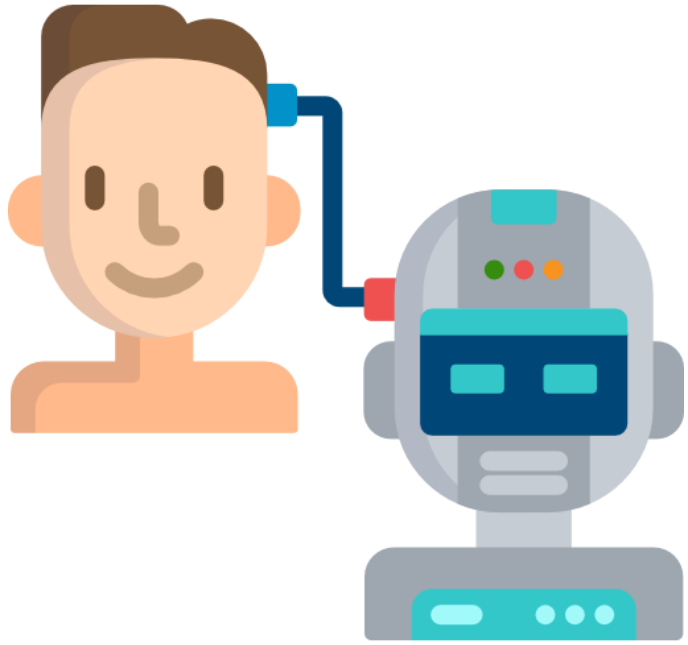
Functional Champion — **Owner** of the CDD

Citizen Developer — **Preparer** of the CDD

## Discussion:

- What challenges do you see with Citizen Development?

- How can we mitigate these Challenges?

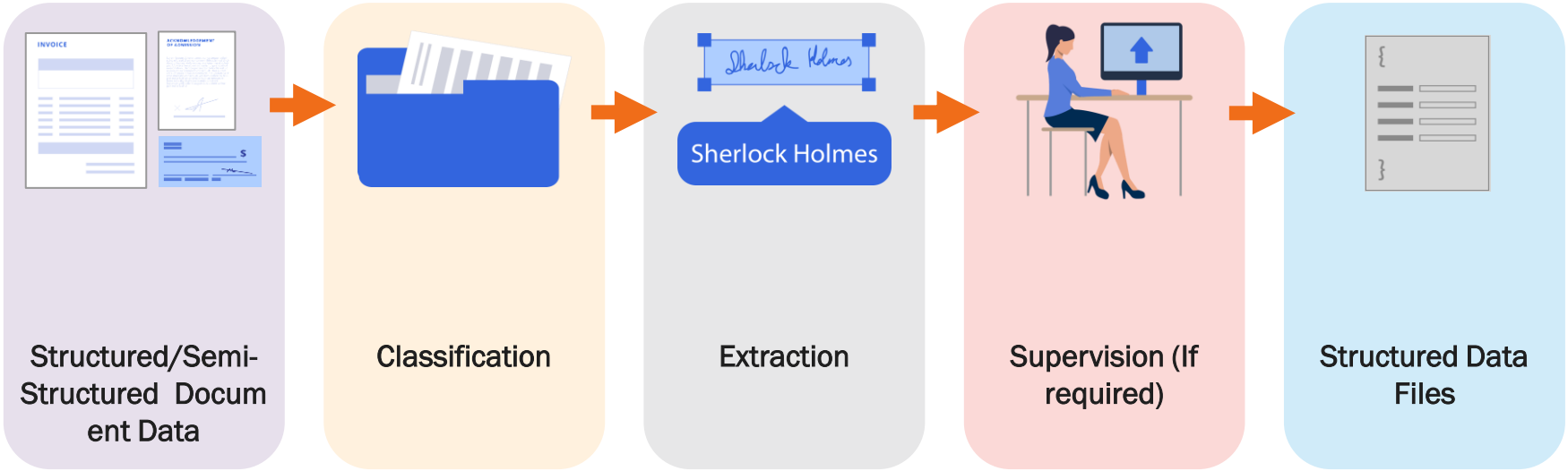


# Topic 5

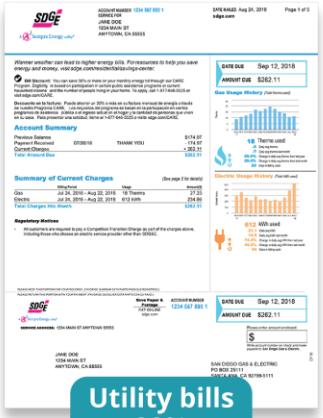
## Demo: Intelligent Document Processing

# Intelligent document processing tools use Machine Learning to classify and extract data from structured, semi-structured, or unstructured data types

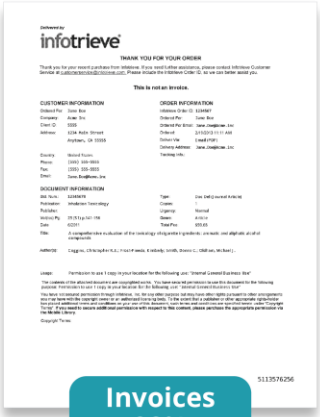
Emerging AI software enables greater accuracy and automation than previously possible



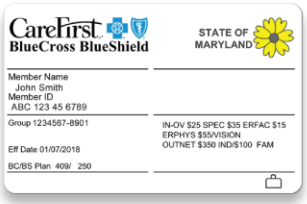
# Successful end-to-end automations rely on IDP tools to provide highly accurate data extraction

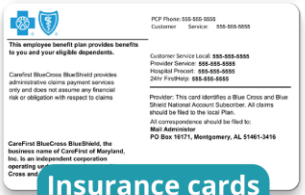


Utility bills
98%

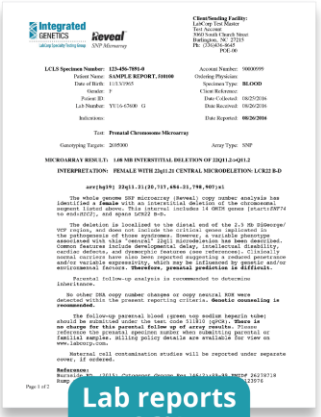


Invoices
98%



Insurance cards
99%



Lab reports
96%


Pay stubs
98%


Building permits
96%

# Topic 6

## Other RPA Platforms: Blue Prism, Automation Anywhere, Power Automate

# RPA Platform Comparison – Features

| | Power Automate + softomotive | blueprism | UiPath | Automation Anywhere |
|---|---|---|---|---|
| **Why do clients choose this product?** | • Seamless integration with Microsoft Office/Dynamics 365 suite<br>• RDA and RPA capability<br>• Citizen Developer enablement<br>• Great Microsoft community support | • Efficient development at scale<br>• Ability to deploy bots rapidly with minimal IT infrastructure cost when scaling<br>• Centralized control system provides greater management of virtual workforce and scheduling | • Product road map, vision and strategy<br>• Ease of system integration & Remote Desktop strength<br>• Capability to automate processes with multiple systems easily<br>• Attended automation | • Easy to use out of the box<br>• Good balance between ease of use and features<br>• Financial/Organizational viability<br>• Great technical, customer and account support for implementations |
| **Leading when compared to Competitors** | • Security and compliance<br>• Low-code desktop automation solution (with Softomotive)<br>• Attended automation<br>• Native Integration with Microsoft 365 and Dynamics 365 | • Implementation<br>• Security and compliance<br>• Reusability<br>• Robot management<br>• Scalability<br>• Lower technical skill requirements | • Process development<br>• Development tools<br>• Data extraction<br>• System & workflow integration<br>• Product strategy and roadmap<br>• Attended automation | • Technology roadmap<br>• Support provided via customer success managers, technical support and account executives<br>• Implementation<br>• Ease of deployment (Bot Lifecycle Management) |
| **Lacking when compared to Competitors** | • Scalability<br>• Implementation<br>• Debugging capability | • Attended automation<br>• Recording capability<br>• Product road map | • Required skills and ease of use<br>• Complex license model<br>• More expensive than competitors | • Required skills for more advanced automations and ease of use<br>• Complex role permissions<br>• Debugging capability |

# Topic 1

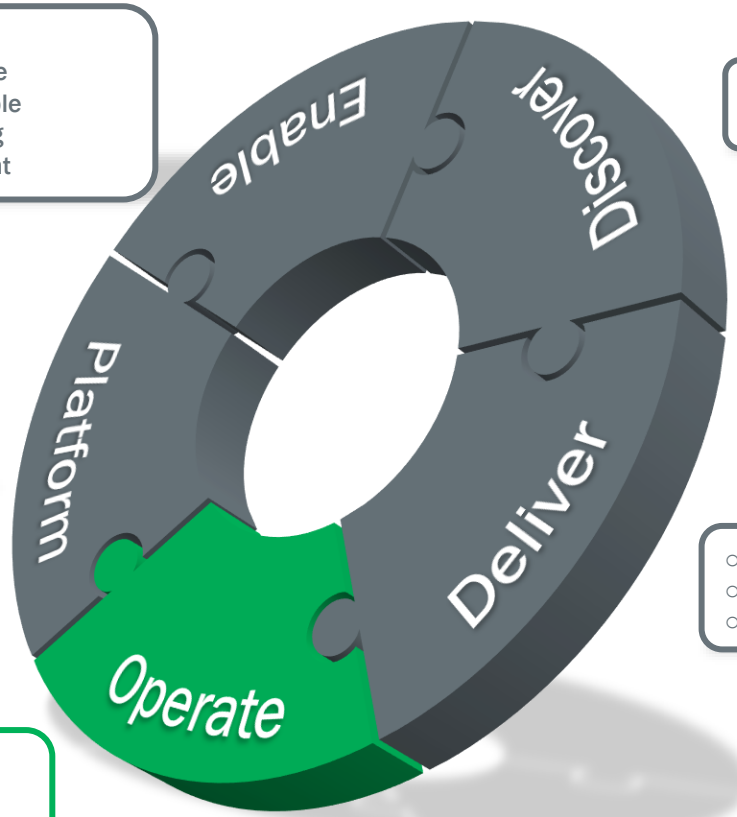
## Operate

# Blueprint for Scale (BfS) covers five capability components required for a successful and scalable automation program

## Blueprint for Scale

- Vision & Strategy
- Program Governance
- Organization & People
- Capability & Training
- Change Management

- Intake Approach
- Opportunity Assessment




- Application Management
- Technical Infrastructure
- Platform Management

- Delivery Methodology
- Technical Best Practice
- Testing Approach

- Business as Usual
- Support Model
- Permissions & Credential Management