

# HOW LONG DOES IT TAKE TO AUTOMATE A BUSINESS PROCESS?



If you consider automating some of your business processes, you took a wise decision! You are already paving the way to your digital transformation! One that should help reduce operational cost, increase margin profits and improve customer satisfaction levels. However, you may face some difficulties finding the answers of this question:

## How long does it take to automate a business process?

In this white paper, I will walk you through the process of estimating the right expected duration of a process workflow automation. Enjoy reading.



**Peter Pavlov**

RPA Chief

Solution Architect



## **01** Measure the Complexity

How many applications are involved?

How many steps does the process consist of?

## **02** Additional Factors

Partial or end-to-end process automation?

User access management

Test preparations

## **03** Conclusion

Small-sized processes

Medium-sized processes

Large-sized processes

I am pretty sure you have already asked yourself how long it will take to automate your business processes and achieve the expected business outcomes. Keep reading to find the answer and grasp a lot more useful information. Without further ado, let's start with it.

The usual answer you would get to such a question is "it depends". There are numerous factors that your process automation depends on. Having a clear understanding of each of them helps you estimate any upcoming request to automate a given business process with ease and confidence. Always keep in mind that your initial idea will evolve over time and new requirements and/or improvements will start popping up. That should not scare you off, but rather encourage you to invest more of your precious time and efforts.

It is virtually impossible to put a price tag on something that is just an idea. For this purpose, people tend to assign different attributes to their respective subjects of interest, that would describe them in the most extensive way, which can be then used as a base for comparison. Considering as many factors as possible to describe your process for automation will allow you to easily identify the amount of time required to bring your idea of eliminating repetitive tasks to life.

Let's start with the most important one - the complexity of the business process.

## Measure the Complexity

You are right to think that the most important aspect of estimating how long it will take to automate a given business process is to measure its complexity. Although it might seem a pretty straight forward task, it isn't until you define the proper "dimensions" for measurement.

### RPA BENEFITS

1. Reduce operational costs
2. Ensure accuracy
3. Enhance productivity
4. Improve customer satisfaction
5. Accelerate employee satisfaction
6. Increase revenue

## RPA METHODOLOGY

1. RPA 1o1 Sessions
2. Build Proof-of-concept (PoC)
3. Due Diligence
4. Plan The Project
5. RPA Project Delivery
6. Animate The Bot
7. Ongoing Support

I like to classify the processes for automation based on their complexity into three distinctive categories. While the definition and naming for each of them tends to slightly differ from client to client, we generally recognize Small, Medium and Large processes. There are two main “dimensions” which are taken into consideration when classifying any given business process - number of applications and number of steps.

### How many applications are involved?

When it comes to Robotic Process Automation (RPA), “Application” has a much broader meaning than usual. Email access or usage of network resources (shared folder) as well as any web or desktop application fall into the RPA’s definition of application. Let me explain.

You can consider the applications as points of focus which your robot (the outcome of your automation) is switching between when executing automation steps throughout the run of your business process. Since the bot is mimicking user behavior and interaction with applications’ interface, it will have to focus on a single application at any given time.

Here is an example that will help you better visualize the idea. Let’s imagine we want to automate a business process that consists of the following steps:

1. Process is triggered by receiving an email
2. Read the email and download its attachments to a shared folder
3. Log in to a CRM system from where you have to read some client related data
4. Open a web-based application to input and read some data

5. Open an ERP system and enter some data which depends on data coming from step 4
6. Write the result summary to a file on shared drive
7. Send an email back to the sender

The diagram below shows a simplified example of your automation steps and the respective dependencies between them. This type of diagram is also called a **Gantt Chart**. It not only gives you a clear picture of what comes after what, but also what your bot must keep focus on at any given point in time.

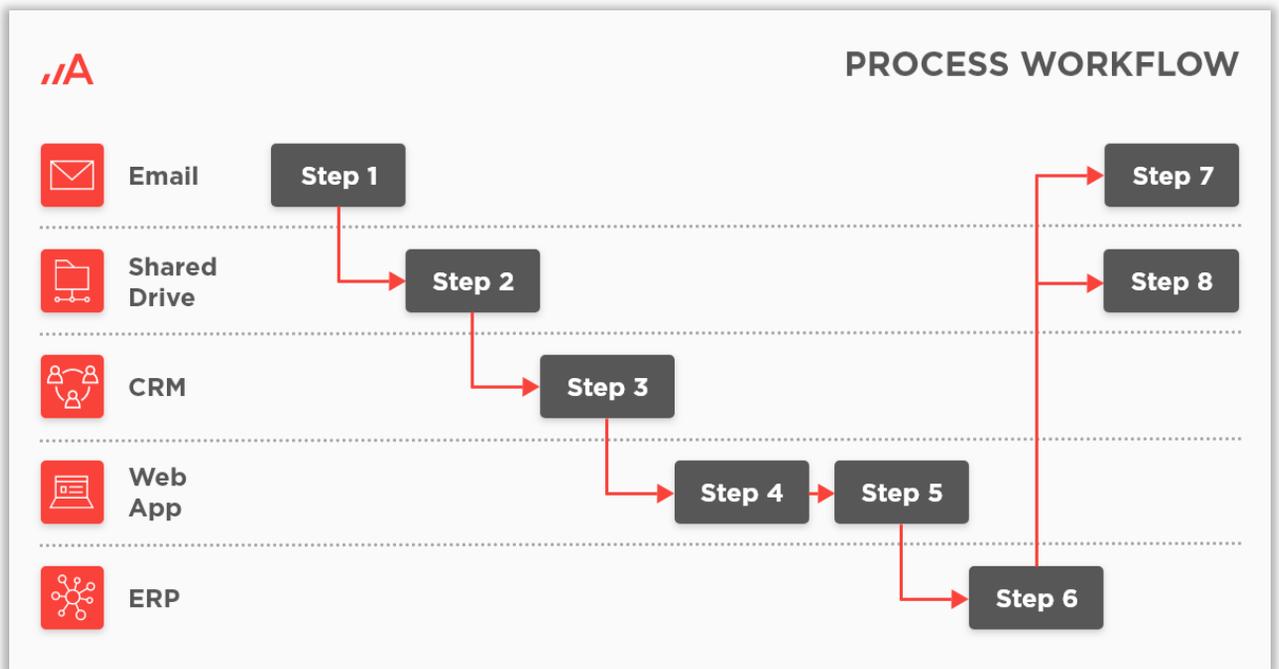


Illustration: sample process workflow

One might argue that “Email” for example is NOT an application, especially if you have a direct access to MS Exchange server and you don’t have to interact with any given mail client like MS Outlook or Thunderbird. But when working with emails, more often than not, you will have to implement some rules to validate one or more of its attributes.

What you would normally do is to apply a “black” or “white” list of senders that are allowed or forbidden to send emails which are of interest to your process. Then, you would probably want emails that contain only specific keywords in their respective subject lines. Finally, you would probably want only those emails that contain attachments, and especially such that are of a specific file type like Excel or Word documents for example.

One might argue that “Email” for example is NOT an application, especially if you have a direct access to MS Exchange server and you don’t have to interact with any given mail client like MS Outlook or Thunderbird. But when working with emails, more often than not, you will have to implement some rules to validate one or more of its attributes.

What you would normally do is to apply a “black” or “white” list of senders that are allowed or forbidden to send emails which are of interest to your process. Then, you would probably want emails that contain only specific keywords in their respective subject lines. Finally, you would probably want only those emails that contain attachments, and especially such that are of a specific file type like Excel or Word documents for example.

Having said that, I personally consider “Email” or any other focus point, which is a part of your process automation, that requires implementing some business logic, to be a separate “application”. As a general rule of thumb, if you have more than one non-trivial automation steps at a given point of focus, then you count it as an application.

**Once you have cleared out how many applications will be automated, it becomes effortless to classify your business process.**

Small ones are usually utilizing one or two applications, mainly involving emails and excel or a web app.

The medium ones are using three to four applications often adding an ERP system to the apps I mentioned for small sized processes.

The “heavy lifters” are the ones that have at least five applications named in their Gantt charts. These are very complex processes and their automation requires a lot of components to fit in perfectly in order to deliver a flawless outcome (Bot).

Earlier, I mentioned that there are a couple of dimensions you should consider when estimating the complexity of your business process nominated for automation. Let’s turn our attention to the second one.

## **How many steps does the process consist of?**

It should come as no surprise to you that there is no clear definition of what a “step” is. That is because sometimes it is extremely hard to identify the existence of a step required for the technical implementation. Business users describe the process in a way they understand it. In 99% of the cases, it differs greatly from what the technically engaged people need in order to automate the business process. Ask a business user something about the app they are using, and they will explain to you what they see on a daily basis – buttons and clicks, or simply put “interface interaction”.

Fair enough, I would say, but **RPA developers think about APIs and “backdoors” to your data.** It often takes a few meetings accompanied by a couple of visual presentations (pptx) before the business users understand the benefits of opening the backdoor. If you’re sitting on that side of the process, don’t hesitate to give your “go” as long as these shortcuts are not against your company’s information security policy. Sometimes, such kinds of optimizations can greatly reduce the overall execution time of your automated process. It can also reduce the amount of automation steps that your process goes through, although this particular effect is recognized and becomes visible much later in the development phase. I am always advocating for a process flow that reflects both perspectives - the one of the business user as well as the one of the technical implementation.

You are right to ask: **“Why don’t you just define the complexity on a task level, instead of mixing up business with technical steps?”.**

I feel obliged to answer that assigning a particular number or alphabetic letters representing the complexity of each and every task, is not only cumbersome, but also quite subjective. The latter is something you would want to avoid at any cost because it will obscure your ability to adequately estimate the total effort needed to automate your business process. Therefore, **try to be as granular as possible when it comes to identifying all the steps.**

In my opinion, any of the below stated operations should be considered as a separate step. These points are not a full list of all possible steps but are one of the most common ones you would come across as soon as you start reviewing a candidate process for automation.

I personally notice one huge issue with step definition, and that is underestimation. Every now and then, I see in the Process Workflow diagram a step either called “validate the email”, “validate the file” or simply “if this step fails, try again 3 times”. Now that is exactly the recipe for missed deadlines. The reason I’m saying it is - **none of these is a single step and a good business analyst should recognize that straight away.** It does sound like a simple one to any business user familiar with the process, but in fact it is a collection of steps which the RPA developer would have to implement.

Let’s take the restore functionality as an example. The simplest definition of it, that comes to my mind, would be something along these lines:

“ Any piece of code that is responsible to bring your bot to the last known successfully executed automation step, without losing information about the progress made so far.

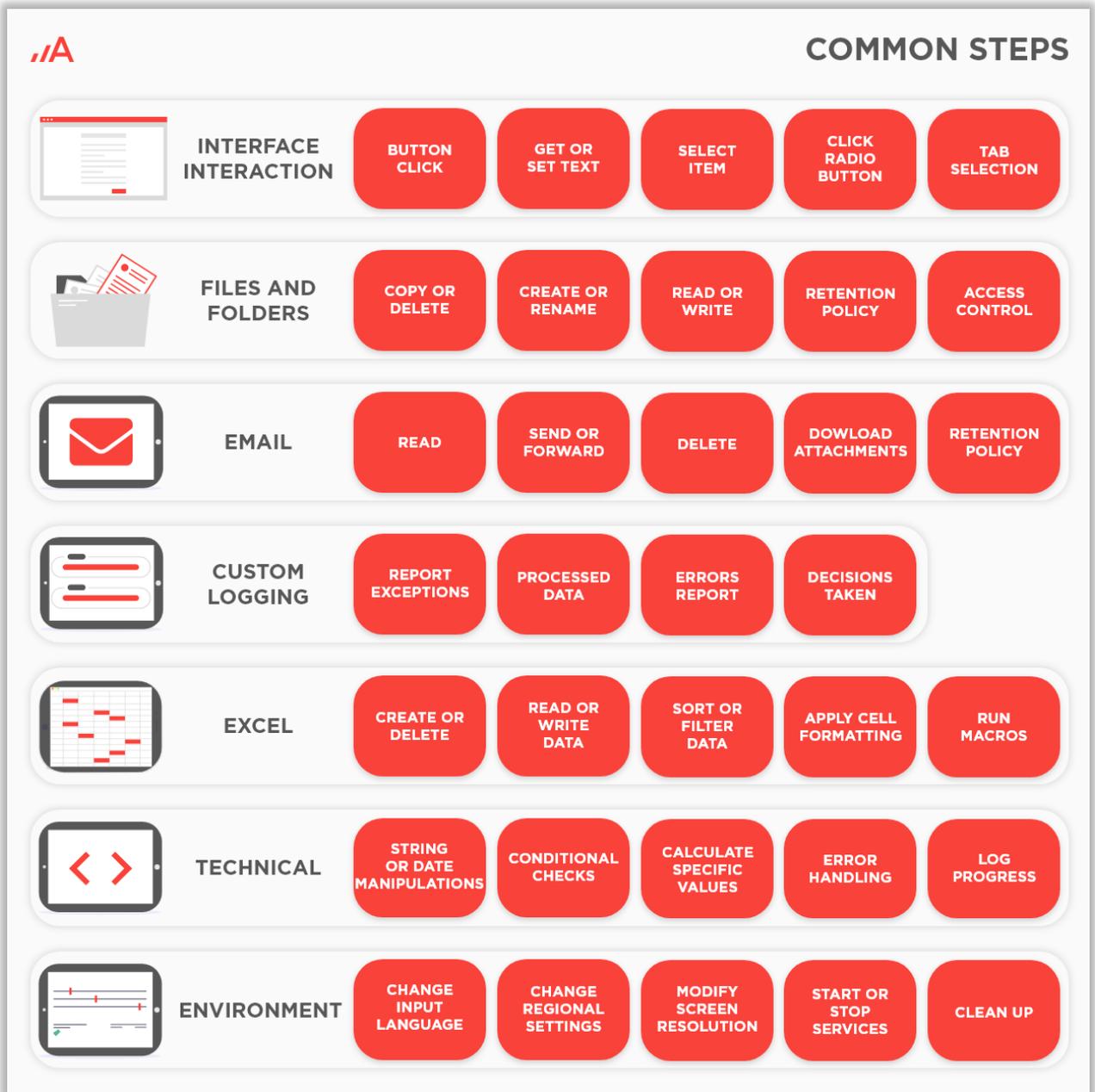


Illustration: common steps

Let's imagine we have a business process which at some point requires interaction with your ERP system. That interaction results in generating and sending of over 100 emails to a dedicated mailbox used by your bot. These emails then have to be forwarded to some external **suppliers to warn them they will be late with their delivery.** So far so good. But one of the business rules tells you that any email received in the dedicated mailbox prior to the starting time of your bot, is considered obsolete

On the other hand, your ERP system allows for only one email to be generated per day. In other words – **you can't afford your process to fail, once it started generating the emails, because a simple restart will lead to their loss.** Well, they won't be lost but someone would have to manually forward them, which makes the idea of automating the process totally meaningless.

So, what you would like to do in such cases should be to implement specific logic that will constantly keep track of how many emails are expected, how many were processed so far and what is left to be done. Should your automated email forwarding step fail for whatever reason, you must bring your bot back to the state as if it never happened. Business users do not care if there was a technical glitch or not, but they are very sensible if results are not waiting for them at the appointed time of the day.

OK, that is a great piece of information, but how many steps are hidden in this restore functionality after all? I say, stating clear rules of what exactly is meant with the restoring, will reveal all the “hidden” automation steps. My guidelines for this particular example are:

- Check the nature of the error (Is it critical or not?)
- Report to support team or log properly that you entered the restore mode
- Check how much of the total batch was process prior to the failure
- Check for leftovers and newly accumulated pieces of the batch that came during the downtime of the bot
- Attempt to repeat the failed step after some delay (ideally controlled by parameter)
- Return to normal workflow after remnants and fresh bits of the job are processed

All the above-mentioned will bring in about five to seven additional steps. Having correctly identified the number of steps within your process should allow to easily classify it. For a small sized process, we usually put 50 as a reasonable threshold, while for the medium ones we typically use 100 as an upper boundary. Large processes are considered those that are comprised of more than 100 steps. Of course, you can adjust these numbers to better fit your organization but try to keep 1:2:3 ratio respectively.

Here comes the question:

**“What if my process is classified as small in terms of applications used and large size in terms of steps involved or vice versa? What weights more?”**

As a matter of fact, this is a quite common situation. I usually refer to this table in order to classify the complexity of the process:



The diagram illustrates the classification of process complexity based on the number of apps and the number of steps. On the left, a person interacts with a smartphone displaying four numbered app icons (1, 2, 3, 4). On the right, a person is shown with a large circle containing the number 1, and two smaller empty circles, representing steps in a process. A red arrow points from the 'APPS' column down to the 'STEPS' column, indicating the relative weight of step count.

APPS		PROCESS COMPLEXITY CLASSIFICATION			
< 2	SMALL	MEDIUM	LARGE		
2 to 4	SMALL	MEDIUM	LARGE		
> 4	MEDIUM	MEDIUM	LARGE		
		< 50	50 to 100	> 100	STEPS

Illustration: process complexity classification

It becomes obvious that step count has far more weight than the number of apps when determining how hard your process would be to automate. My experience so far, shows that **in 50-60% of the cases you would have to deal with Medium size, 20-30% with Small size and 10-20% with Large sized processes.**

Now that we have thoroughly discussed how to measure the complexity of your process subject for automation, let's turn our sight towards some other points which have to be considered.

## Additional Factors

Everything said so far goes into the “common sense” category, but sadly you have to think of a few more points before drawing the bottom line and figuring out how long will it take you to automate your business process. I will name the most prominent ones without claiming that’s all there is. Let’s turn our attention towards the big picture and ask ourselves the question: “Where does my automated process fit in?”

### Partial or end-to-end process automation?

In reality, you won’t always get the chance to automate an end-to-end business process, meaning you will have to make sure your automated part of the process has to fit in tight with whatever lies before and after it. That often requires you to think of the interfaces that will be used to make this connection work smoothly.

My experience shows that it takes a high amount of time to plan and negotiate with third parties a precise interface description. That effort is just ridiculous compared to what it takes afterwards to simply implement and integrate it. Examples of this are to be found almost everywhere. **Say you have to feed data to an AI model, and consume the result it produces,** because that result is needed later in your process flow. **If your ultra-smart AI model is to be provided by a third party, make sure you provision some additional time for clearing out all the details which tech guys need in order to make the great idea works precisely as you intended.**

Neglecting this aspect will only lead to a constantly failing process which itself will generate a lot of disappointed customers or employees. Supportive employees are something you will always need on your side, because whenever they hear “automation” and “robots”, they start getting worried about their own jobs. They should feel the bots that automate various business processes within your organization as their colleagues, not as competitors. So, if you want the bot to be an integrated team member, then it will have to get the proper grants to perform its daily tasks.

## User access management

Initiating an onboarding procedure for your upcoming bot should start as soon as you pass the feasibility check and budget has been granted. The process of clearing out all the automation steps will consume some time, which can be used to get necessary access permissions in parallel. If you, by some reason, happen to postpone this task, you will have to add a few extra weeks on top of the total duration of your process automation. Onboarding time varies greatly for each organization. I have seen everything between two and eight weeks, with average being around four weeks. The sooner you start, the more flawless your development and integration will be.

Another head start you can give yourself in order to reduce the overall delivery time is to have an idea about how you’re going to test the automated business process.

## Test preparations

User Acceptance Tests (UAT) are probably the most important part of your RPA delivery phase. Reaching that point in time brings you just a step away from the real automation of your process. It means that you can’t afford wasting more time. **Be prepared with a clear Test Scenario and enough test cases to cover all possible paths of your process.** You just don’t know how many times I’ve seen developers sitting and waiting for test data to cover the appointed test cases, and I’m not talking just RPA here.

It happens all the time across virtually every single branch of the IT industry. Think in advance if you want to achieve faster results with zero idle time of your team. On average, you need no more than a week to complete the UAT phase. Any extension above this number, without a strong requirement posed by the business users is deemed as inefficient and usually decreases your ROI.

Like I already said, there are a lot more factors that could affect the duration of your process automation. I had the chance to outline just a few of them, which I personally consider to be essential for the overall timespan and should allow you to make a better conclusion of how long it will take you to complete the process of switching from manual labor to digitally assisted one.

## Conclusion

Earlier I cleared out the idea of how to measure the complexity of a business process by knowing the amount of applications and steps required to execute it. I added a few other secondary points worth mentioning. That said, we have the door opened for our conclusion.

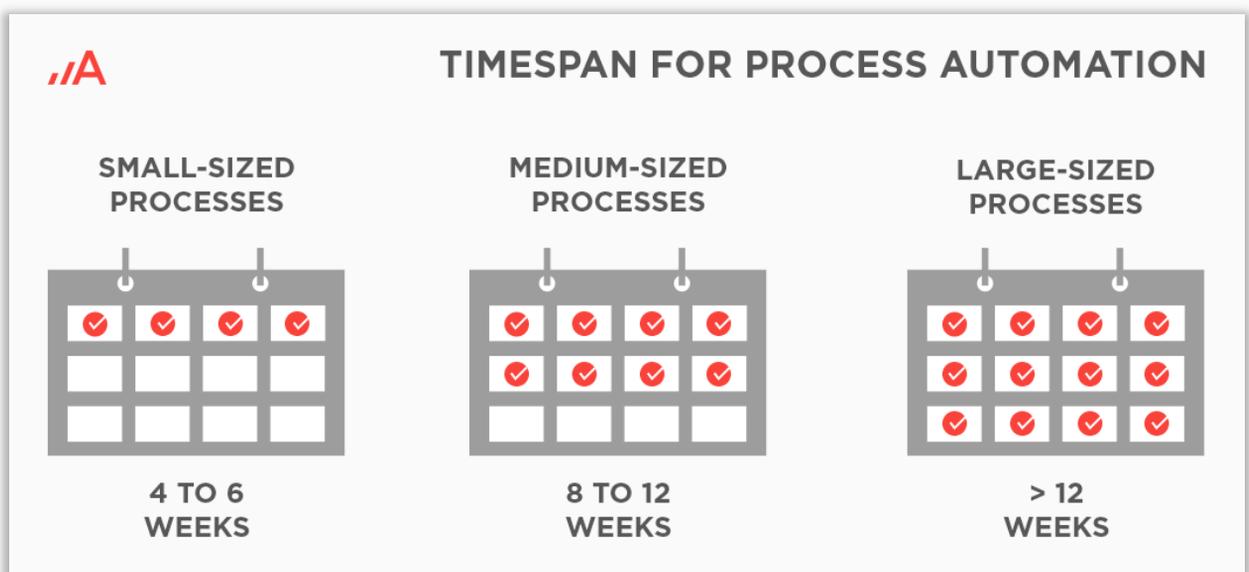


Illustration: timespan for process automation

Detailed estimations will be a subject of another post I am looking forward to creating in the near future. So, stay tuned. I will gladly share with you how to chop up the complex processes into atomic tasks. Then you will be able to precisely determine how many man hours would be needed to automate your process from the inception of an idea to having a working productive robot. However, even without this information currently in place, simply due to the lack of an actual example to work with, I can still reveal the numbers drawn from my practice.

## Small-sized processes

Small-sized processes are usually the ones that are least applicable for parallelism in terms of efforts done for business analysis (BA) and development. The ratio between hours spent on these two topics is also less than the one for medium and large size processes. Therefore, you will probably need between 4 and 6 weeks to deliver an automated solution assuming you know what you are doing.

## Medium-sized processes

Medium-sized processes are the most common ones. You will come across one of these 3 times more often than any of the other two types. Such process requires slightly more development hours relative to those spent on BA. You can expect to reach your goal in 8 to 12 weeks depending on the number of steps you have defined.

## Large-sized processes

On the upper end of the scale you have the large bots, which usually allow parallelism between BA and development. You can afford to start with implementation without clearing out all the steps required by the process and do it on the fly. This kind of processes will require at least 12 weeks until you reach the expected automation levels. I have seen a mammoth process with over 200 steps that took about half a year to be automated.

The bigger the process, the more obstacles you can expect down the road. Luckily, you can learn and avoid them with any successive process that comes next. There are no two processes that are exactly the same which in fact will always keep you alert and prepared.

**Should you have any further questions on this topic, please reach out to us.**

## ABOUT ADASTRA BULGARIA

Adastra Bulgaria is a European-based globally recognized IT services and consultancy company. We enable business digitalization through tailored solutions in the domains of Cloud, Information Management, RPA, Managed Services, Business Intelligence, and Enterprise Application Development.

Adastra's Bestshoring Delivery Model provides the best blend of onsite, nearshore and offshore engagement to get tailored End-to-End enterprise solutions delivered with speed and agility.

The geographical diversity of our delivery hubs network helps diversify operational and delivery risk, and gain greater management flexibility, while enabling our highly skilled consultants to cooperate on day-to-day matters.

## ABOUT THE AUTHOR

Peter Pavlov has been growing his expertise in the IT field for more than 13 years. During that time, he has had a hands-on experience in various sectors of the economy like Retail, Wholesale, Manufacturing and Automotive.

Peter has been specializing in Robotic Process Automation for 5 years now. Recently, he has been crafting RPA solutions that brought tremendous outcomes for leading players in Automotive, Logistics, Procurement and Finance industries.



32 Atanas Dukov Str., 1407 Sofia, Bulgaria



[marketing@adastrabg.com](mailto:marketing@adastrabg.com)



[www.adastrabg.com](http://www.adastrabg.com)

Follow us at:



**././ADASTRA**