outsystems

Governing App Development Outside of IT Playbook

Part I - Getting Started

Governing App Development Outside of IT - Playbook
Part I - Getting Started | Version 1.0

Introduction 5

What is Citizen Development (and why it's not enough)? 6

Who should read this playbook? . . 6

Playbook Structure 7

Roles Involved. 8

Maturity Model Levels 9

Getting Started 9

Setting up for Success 9

Transforming your Business. 9

Governance Framework Overview . . . 10

1. Setup 10

2. Software Development Life Cycle (SDLC) 11

3. Monitoring and Event Management 12

4. Continuous Delivery 12

Applying the maturity levels to the Governance Framework 13

Getting Started 14

1. Setup 14

1.1. Infrastructure Configuration 15

Basic Infrastructure 16

Shared Infrastructure 16

Dedicated Pipeline. 17

Dedicated Infrastructure 17

Processes 18

Setup a new OutSystems Infrastructure 18

Setup a new pipeline 18

Setup infrastructure to use Workflow Builder. 18

1.2. Selection and Onboarding. 18

Who are my Business Developers?. 18

How to choose your candidate apps? 19

Specific requirements for simple workflow applications . . 19

What does the selection process look like? 20

How to onboard these Business Developers?. 20

1.3. Platform User Management 21

Defining the roles of Business Developers. 21

Creating users for business developers and assigning roles . . . 22

1.4. Foundational & Shared Services 23

End-user Authentication 23

Identify authentication method. 23

Define end-user classification rules 24

Configure authentication method. 24

Workflow Builder 24

2. Software Development Lifecycle 25

2.1. Application Development 25

Using Workflow Builder 26

Using Service Studio. 27

2.2. Release & Deployment Management 28

Using Workflow Builder 28

Using Service Studio. 29

2.3. User & Access Management. 31

Using Workflow Builder 32

2.4. Incident Management 33

Problem Management 34

2.5. Quality Assurance 35

Testing 35

Technical debt management 36

3. Monitoring and Event Management. 37

4. Continuous Delivery 38

Training and Certification 38

Guided Paths and Courses. 38

Training Planner 38

Closing 39

Introduction

> Software is eating the world.
>
> Marc Andreesen

This quote is even more true today than when it was penned in the Wall Street Journal by Mr. Andreesen back in 2011. Today, software has been integrated into every aspect of our society, adding increasing pressure on IT organizations to digitize every touch point of their customer and employee journeys, and fast. Between this growing demand and keeping the lights on, IT backlogs continue to grow. All too regularly IT leaders push off valuable digital business initiatives simply because of a continued lack of resources.

Forrester estimates the US economy alone will **suffer a gap of 500,000 developers by 2024.**

How to Harness Citizen Developers to Expand Your AD&D Capacity,
Forrester Research, April 19, 2017.

Enter IT-governed, business-led application development. The time has come to introduce business experts, analysts, process owners, and other line-of-business employees into your development lifecycle.

The premise and promise are simple. Champion business innovation where it actually starts, on the business side. Empower business experts to improve their team's productivity and deliver more value to the organization while freeing up valuable IT resources.

Gartner estimates that by 2023 the number of active **citizen developers at large enterprises will be at least four times the number of professional developers.**

The Future of Apps Must Include Citizen Development,
Gartner Research, April 4, 2019.

Seems simple, right? But this isn't a totally new concept. It's safe to say that, in most organizations, employees outside of IT have used the various software tools at their disposal to create digital business solutions, outside the control and scrutiny of IT. This is oftentimes referred to as **Citizen Development or Shadow IT**.

What is Citizen Development? (and why it's not enough)

Back in 2012, Garner analysts declared "We're all developers now" in reference to the growing line-of-business development movement and coined this definition:

> A user who creates business applications for consumption by others, using development and runtime environments sanctioned by IT.
>
> Gartner Definition Glossary, Gartner Research.

The challenge and focus of this playbook lies in the last three words, "sanctioned by IT." With the advent of solely no-code platforms, the line between citizen development and shadow IT has become increasingly blurred. Today, many software tools targeted at so-called citizen developers include little to no IT governance. It's not enough for IT to simply give their blessing for the line-of-business to start building their own apps. They must actively lead and manage the entire process.

This siloed citizen development approach also ignores how modern teams work. Rarely do business developers build applications from start to finish with no assistance and maintain the app themselves through its entire life. Instead, most organizations employ cross-functional teams that include both business experts and professional developers creating and evolving apps in collaboration.

While the idea of giving full autonomy to business people to solve their own unique productivity challenges looks compelling on the surface, a strategy that's not centered around IT governance and that ignores how modern teams are structured is destined to fail. Traditional citizen development or shadow IT leads to applications that are prone to data breaches, quickly orphaned when an employee leaves a company, so poorly constructed that IT must totally rewrite them when they become mission critical, or all of the above.

Who should read this playbook?

We created this playbook to help organizations establish a IT-governed, line-of-business application development strategy using OutSystems. You should read this playbook if:

- You belong to an IT department that wants to embrace line-of-business app development but in a governed way
- You belong to an IT department with a long app backlog that keeps saying "no" or "not right now" to these incoming business requests
- You want to empower your business people to take part in application development without needing to have professional development skills

Or you might feel strongly about all of the above. In any case, **this playbook will guide you through the process of setting up a successful, governed line-of-business development strategy from the start.**

Playbook Structure

This playbook has the following structure:

Roles Involved

Which are the roles that will be needed for you to have governance and full control over your app portfolio.

Maturity Model Levels

In order to help you on the path to go, we'll need to understand where you are, focusing on the right processes at the right time.

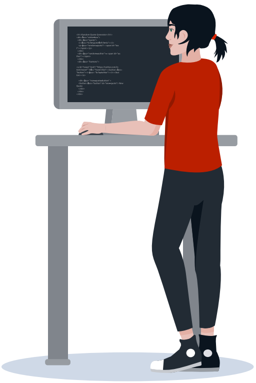
Governance Framework Overview

What do you need to do on each step of the way to assure Governance throughout your joint journey with business experts. The required roles, the process and the tool for each step.

Roles Involved

These are the roles to get you started:

In IT



IT Manager / Factory Admin

The IT manager within your company or the administrator of your OutSystems Factory. Your Factory Admin can also be the IT SPOC.

IT SPOC

Someone from IT that will be the single point of contact (SPOC) to the business user regarding this initial maturity level. If a business user needs to contact someone, this is their go-to person.

In Business

Business Developer

Someone in the line-of-business that will develop an application. Don't worry, later we'll help you find out the best person.

LoB Manager

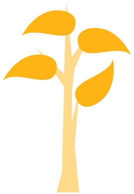
Sponsorship and support from your line of business (LoB) and departments are important so you can be surrounded by people seeking the same impact.

Co-workers

Some workers that can help on a given application testing.

Maturity Model Levels

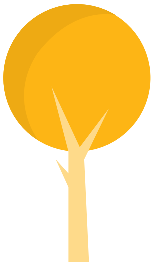
In order to help you achieve your final destination, we need to understand where you are now, so we can guide you throughout this journey. We consider 3 maturity levels:



Getting Started

- You probably have some sort of Shadow IT in your organization but with little to no governance. You have selected OutSystems to enable your organization to deliver digital solutions extending your development capacity.

- In this level, you will pilot this new strategy and start identifying the people in your organization and defining the processes that will lead to the successful adoption of governed line-of-business development.



Setting up for Success

- In this level, you have already proven that governed line-of-business development can have success in your organization. You have a few applications delivered by business developers with help and governance from IT and these apps have recurrent usage across the organization and are delivering value to the business.

- Your next step is to prepare for scale, extending the usage of line-of-business development and setting up the processes and IT structure that will support this strategy across the company, giving more autonomy to business developers while keeping the governance and control from IT.

Transforming your Business

In this level, you have established a successful governed line-of-business development strategy that can expand across the entire organization and across regions. The focus of this maturity level is on iterating the model to ensure full autonomy and governance, measuring outcomes and collecting feedback for continuous improvement.

Governance Framework Overview

Our suggested Governance Framework consists of four stages, and in each stage, there are different processes. These four stages are:

Setup

Before you can start your developments you'll need to set things up. The following processes will guide you, in the presented order, in creating the necessary conditions to have business people creating applications using OutSystems.

1. Infrastructure Configuration

OutSystems Platform Infrastructure Configuration recommendations and guidelines. According to each one of the maturity levels, different configurations will be proposed in order to increase governance over the line-of-business development practice.

2. Selection & Onboarding

The process that defines business developer safe zones, selection and suitability of applications eligible to line-of-business development and how to onboard business developers.

3. Platform User Management

How to configure business developers permissions and roles in the platform. Which assets and resources are they allowed to use and change.

4. Foundational & Shared Services

Definition of Foundation Data, Shared Components, and Services available for line-of-business development that complement what is provided out-of-the-box by the platform. This typically includes end-user authentication mechanisms, live style guides, widgets, integrations, and data services.

5. Communication Plan

How to communicate to the organization about the line-of-business developments strategy. Define a communication plan (formal and informal) about the line-of-business development Community, Software Development Life Cycle, Achievements, and Events, including in this communication the company goals for this strategy.

6. Application Portfolio Management

Framework for managing business-developed software applications and software-based services. Provides an inventory of the company's business-developed applications and metrics to illustrate the business benefits of each application.

Software Development Life Cycle (SDLC)

This stage covers the processes that support the development of business-developed apps. Some of these processes are foundational and need to be supported in the very early steps of line-of-business development some others are recurrent, but all will need to be iterated during this maturity level and evolved in the following maturity levels.

1. Application Development

Defines the process and guidelines on how a new application should be created. Establish what questions should be answered by a business developer after the ideation phase.

2. Release & Deployment Management

Defines the process responsible for planning, scheduling and controlling the build, testing and deploying Releases. It defines, depending on the line-of-business development maturity level (and IT control level), who is responsible for each action of the Release Life Cycle.

3. User & Access Management

Establishes the process of identifying which roles and users/groups the application will have. Guides the business developer on building the permission matrix of the application and assigning the role to users or groups.

4. Demand Management

Defines the methodology used to forecast, plan for and manage the demand for data, services, and utilities required for business-developed apps. This will be provided by IT.

5. Incident Management

Defines the IT activities to identify, analyze, and correct cross-cut incidents on the line-of-business development environment to prevent a future recurrence.

6. Problem Management

Defines the process and activities responsible for managing the life cycle of all problems. Its main goal is to prevent problems and their resulting incidents from happening.

7. Quality Assurance

Defines the process of quality assurance of the business-developed applications accordingly to the maturity level.

8. Change Management

Defines standardized methods and procedures used for efficient and prompt handling of all changes related to business-developed applications in order to minimize the number and impact of any related incidents. It also includes how the company should communicate the changes made.

Monitoring and Event Management

This stage will cover processes related to exceptions or events.

1. Troubleshooting

Defines how the business-developed application troubleshooting can be done

2. Auditing

Defines the processes for two different streams inside the auditing domain:

- How the business-developed application auditing can be done by the IT team on the technical perspective
- How the Business Owners can track the performance of the business process

3. Application Support and Maintenance

Defines the process of how the support is provided to business-developed applications and who provides this support. It should be clear to the users how to report the issue and its life cycle.

4. Event Handling

The process that defines a set of preventive events automatically triggered by the platform and who and how they are handled by the organization. Depending on the event's criticality, a set of manual and automatic actions should be triggered accordingly.

Continuous Delivery

Continuous delivery is the principle of building applications that can be released safely to production at any time, on-demand. The following processes are mostly addressed in higher levels of maturity in our Line-of-Business Development Maturity Model.

1. Training and Certification

Organizes training content in different categories to create awareness among business developers regarding application development best practices. Defines a Certification plan and the possibility of a Gamification strategy.

2. Culture

Developing a culture around the success of Line-of-Business Development initiatives is crucial. Culture is a key enabler for the growth or scaling of Line-of-Business Development in an established organization.

3. Line-of-business Development IT Team

As the company gets more mature, the number of business developers will grow, as well as their demands, for example, the needed APIs. Therefore, IT will need to answer that demand by creating a team that will also be responsible for helping on platform setup updates, software development life cycle, monitoring and control, and continuous delivery.

Applying the maturity levels to the Governance Framework

Depending on your company's maturity regarding line-of-business development, there are processes that apply and others that don't. Following the Agile principles, if you are in maturity 1 - "Getting Started", there are some processes that you should be operationalizing (the ones from maturity 1), others that you should be designing (the ones from the next maturity - 2, in this case) and others that you can just leave on hold (maturity 3).

Part I of the Governing App Development Outside of IT Playbook will cover the Governance Framework stages and processes according to the first maturity level, "Getting Started".

		Maturity Levels		
Stage	Process	I **Getting Started**	II Setting up for Success	III Transforming your Business
1. Setup	1.1. Infrastructure Configuration	●		
	1.2. Selection & Onboarding	●		
	1.3. Platform User Management	●		
	1.4. Foundational & Shared Services	●		
	1.5. Communication Plan			●
	1.6. Application Portfolio Management			●
2. Software Development Lifecycle	2.1. Application Development	●		
	2.2. Release & Deployment Management	●		
	2.3. User & Access Management	●		
	2.4. Demand Management		●	
	2.5. Incident Management	●		
	2.6. Problem Management	●		
	2.7. Quality Assurance		●	
	2.8. Change Management		●	
3. Monitoring & Event Management	3.1. Troubleshooting		●	
	3.2. Auditing		●	
	3.3. Application Support and Maintenance		●	
	3.4. Event Handling			●
4. Continuous Delivery	4.1. Training and Certification	●		
	4.2. Culture			●
	4.3. Line-of-business Development IT Team			●

Getting Started

1. Setup

Let’s start by detailing the processes in the Setup stage of the Governance Framework.

Most of the processes in this stage are detailed and operationalized in this maturity level, exceptions made for the Communication Plan and Application Portfolio Management which are addressed in Part III of this playbook.

1.1. Infrastructure Configuration

In this chapter, we will cover the possible infrastructure configurations that support the line-of-business development strategy, including the pros and cons of each one. We have identified four use cases or situations and, for each one, we propose a recommendation of the most suitable infrastructure configuration. This is a general recommendation that serves as a guideline for your decision but you should also consider your specific requirements, especially if you already have an OutSystems license and infrastructure.

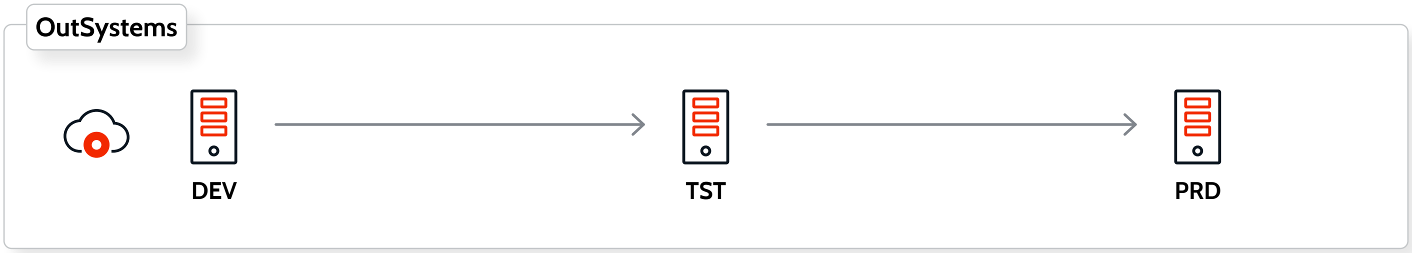
These recommended configurations have their own specificities and therefore their pros and cons that we describe below.

Use Case	Recommended Configurations	Choose this option if you...	Avoid this option if you...
Business-developed apps **only**	**Basic Infrastucture**	• Have no existing OutSystems infrastructure • Would like to set up a new infrastructure with minimum effort to start line-of-business development	• Want your infrastructure to run on-premises • Are also interested in building IT apps besides your business-developed apps
Non-mission critical IT apps and business-developed apps	**Shared Infrastructure**	• Would like to leverage an already existing OutSystems infrastructure • Want to avoid additional infrastructure provisioning costs	• Don't want your IT and business apps to share the same runtime resources (including database storage) and compete for deployment cycles
Mission-critical IT apps and **shared**[1] development between IT and business-developed apps	**Dedicated Pipeline**	• Want to ensure full-stack isolation between business users and IT apps running in Production	• Are not willing to upgrade to an Enterprise edition
Mission-critical IT apps and **full life cycle isolation** between IT and business-developed apps	**Dedicated Infrastructure**	• Want to ensure complete life cycle independence between business and IT apps	• Don't want to set up a separate OutSystems infrastructure • Want to manage the life cycle of your OutSystems applications in a single infrastructure

[1] In this scenario, IT and business apps share the same development environment

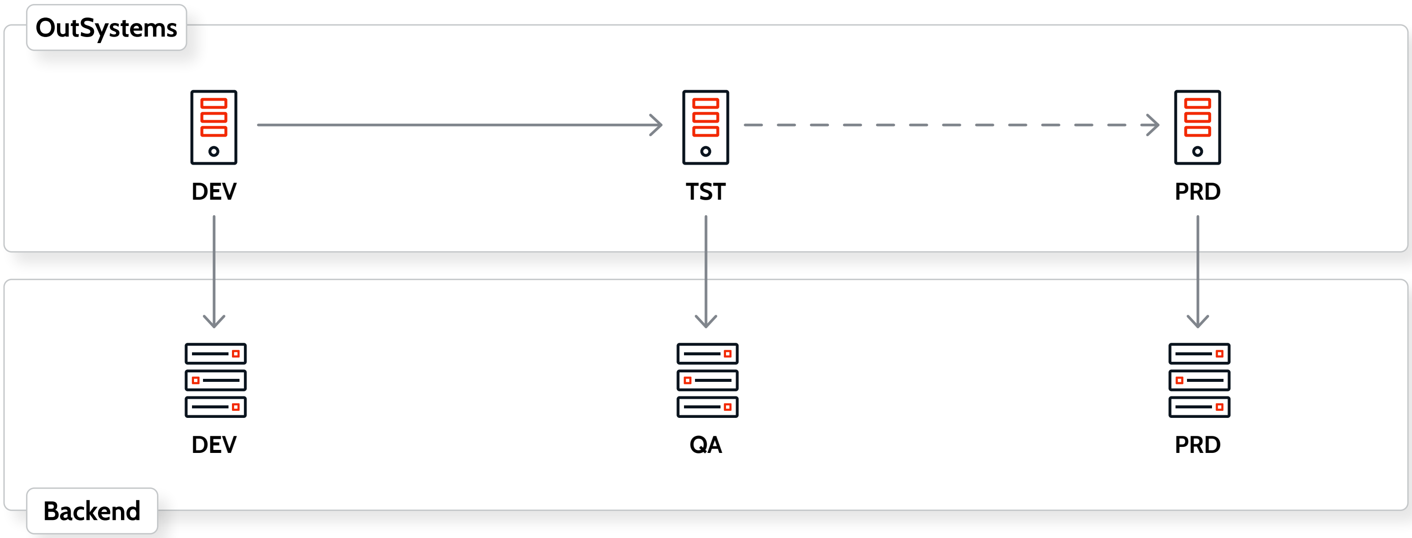
Basic Infrastructure

- Accelerate initial setup process by leveraging an OutSystems PaaS deployment
    - Requires Basic edition
    - No integrations with backend systems
- Suitable for launching a line-of-business development motion with OutSystems
- Not available for self-managed installations



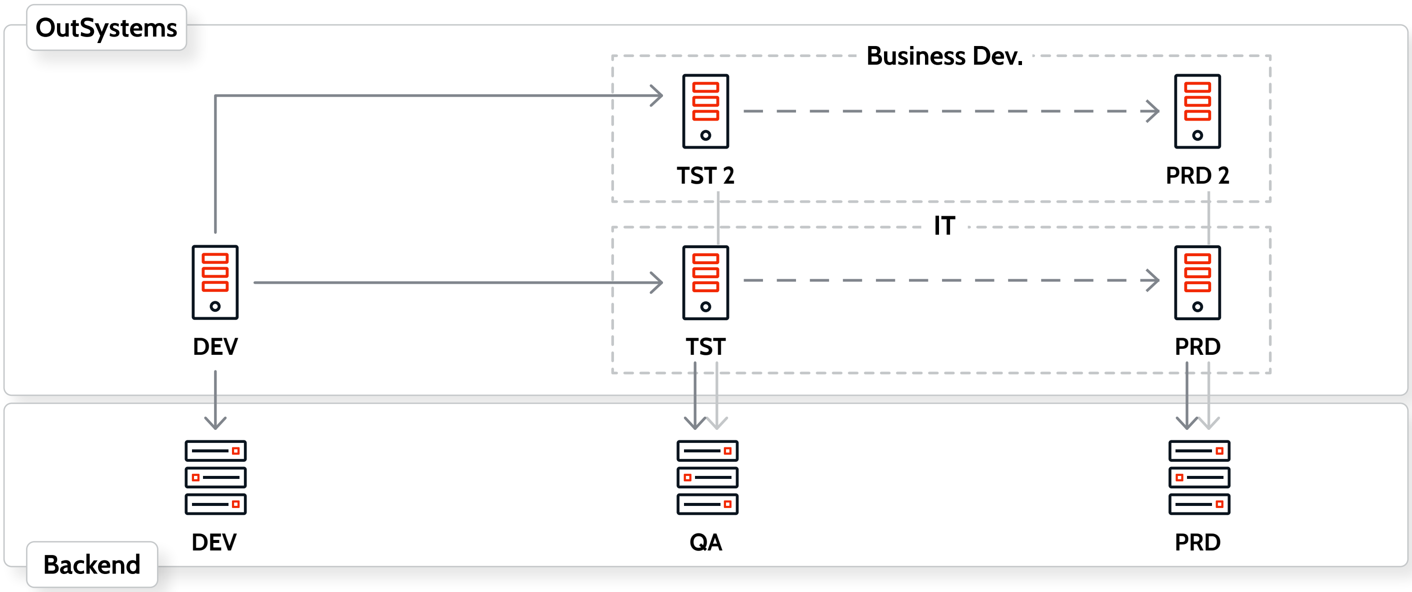
Shared Infrastructure

- Leverage already existing OutSystems infrastructure (self-managed or OutSystems PaaS) for IT apps
    - Requires Basic (PaaS only) or Standard editions
- There's no runtime or database segregation between business and IT apps



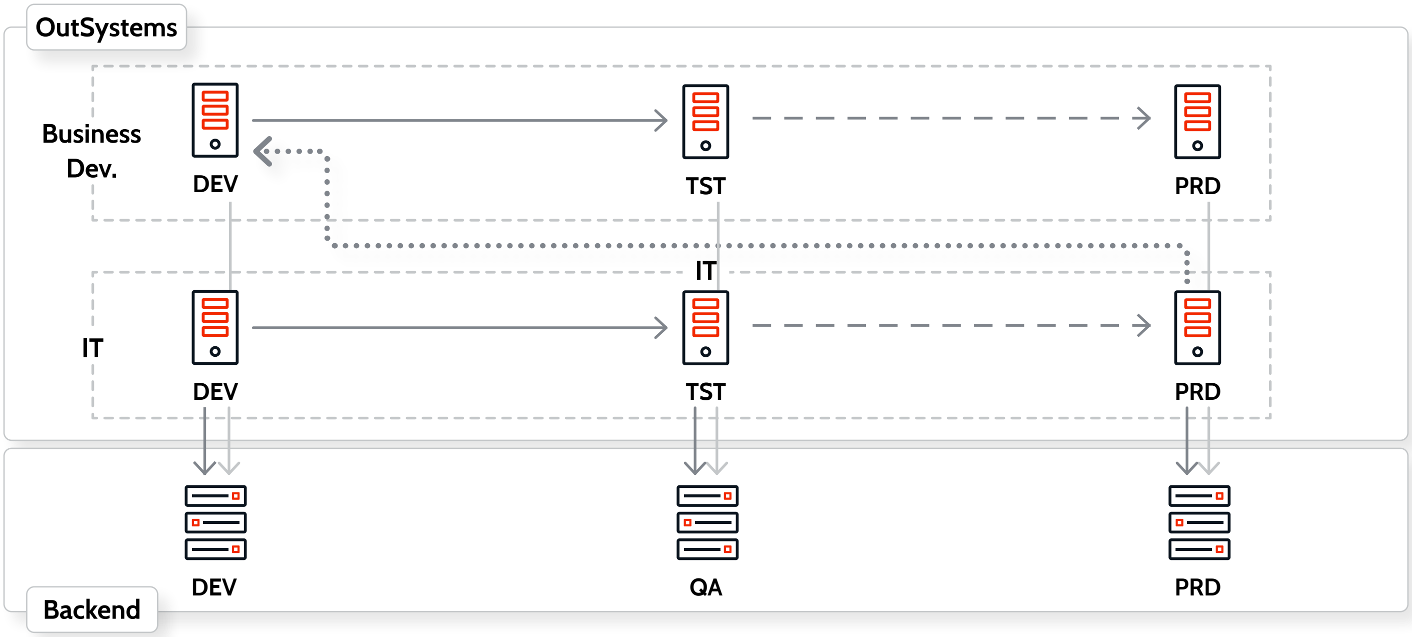
Dedicated Pipeline

- Dedicated pipeline for business-developed applications
    - Requires Enterprise edition
- Ensures full-stack isolation between line-of-business development and IT apps running in Production
- TST environment provisioning is optional
    - Possible usages include "beta-testing" step, IT validation process, etc.



Dedicated Infrastructure

- Dedicated infrastructure for building and running business-developed applications
- Ensures total release cycle independence between line-of-business development and IT apps
    - Shared components are distributed in production-ready status



Processes

Setup a new OutSystems infrastructure

This is the case if you do not already have an OutSystems infrastructure where you want to run your line-of-business development activities, e.g. a Basic Infrastructure or a Dedicated Infrastructure.
In case you have chosen a Basic edition infrastructure, OutSystems will set it for you. This is the fastest way to get going if you are starting with OutSystems.
In case you have chosen to set up a new dedicated infrastructure you will need to contact your OutSystem account manager to provision this license, either for self-managed or for OutSystems PaaS.
More information can be found here.

Setup a new Pipeline

This scenario requires an Enterprise edition license. If you do not have an Enterprise license or require an additional pipeline please contact your OutSystems account manager.

Setup infrastructure to use Workflow Builder

Please refer to the documentation on how to setup Workflow Builder here.

1.2. Selection and Onboarding

In this chapter, we will guide you on how to select potential business developer candidates on your organization and how to set up a process that onboards them into the program you define.

Who are my Business Developers?

I'm sure everyone knows that one person that, although didn't have specific training that gives them development skills, is extremely proactive and motivated to learn to develop applications in any technology. Probably there's at least one person that matches this description in any organization, but this is not the only profile you should be looking for.

Business developer candidates are people that have several of the following characteristics:

- They are working in business units, outside IT
- They need to address their business or productivity challenge
- They know their business domain and have ideas on how to improve it
- They are digitally literate and creative users
- They don't know how to code and are not familiar with an app development process
- They are motivated to solve their challenge with the means they have at their disposals such as office suite tools or many times other tools they source through searching the internet
- Eventually, they have asked IT a few/many times to help solve their challenges, but haven't had the luck to get enough priority in IT backlog

How to choose your candidate apps?

Now that you know better how to spot and select business developers in your organization it is very important that you are able to understand if their needs are suitable for your governed line-of-business development strategy.

Not all of these needs are suitable for line-of-business development. Some might actually represent mission-critical business processes, have a requirement for regulatory compliance, or require complex integration requirements. In all of these cases, you should prioritize the development of the solution that addresses these business needs in the IT backlog.

So the question that arises here is: What is the type of applications that are safe to be addressed by business developers?

The answer to this question is provided by a suitability test. We provide an example of a possible suitability test that you can run with your candidates. This suitability test will allow understanding both if the candidate business developer and the candidate business-developed app are a good fit for your governed line-of-business development strategy.

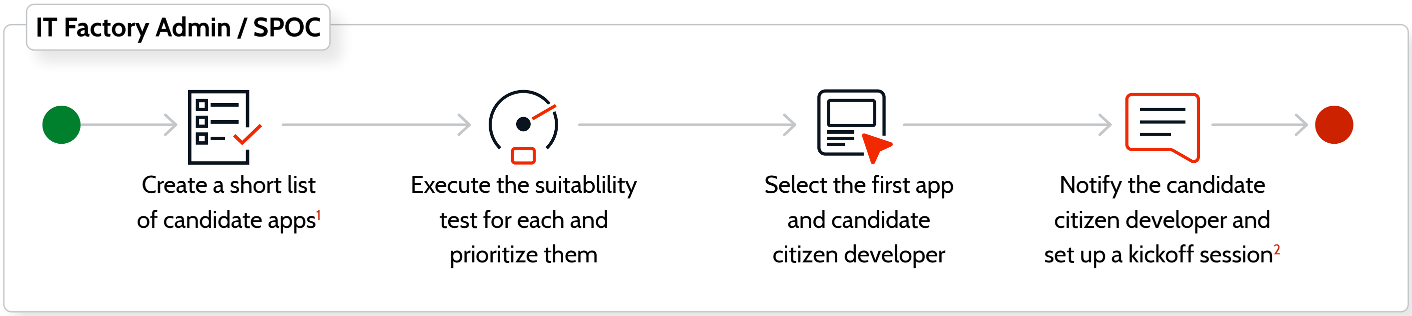
As you get started on this line-of-business development strategy it is very important that you celebrate your early successes. This will fuel the organization to continue developing this strategy and expand the adoption of line-of-business development. Therefore, make sure that the first candidate apps that are selected can actually be launched to the organization and that that milestone can be celebrated.

Specific requirements for simple workflow applications

If the candidate app is to map a business process or workflow, you should make sure that this is addressing a simple workflow to start with line-of-business development. But what do we mean with a simple workflow app?
The simple ones, internal productivity apps, are typically business processes triggered by a form submission, with a mix of simple human decisions with simple business rules.
The workflows are typically one to 2 steps of approval. Many times they represent approval processes that are being addressed in mailboxes and spreadsheets.
The typical process and case operations are the assignments, reassignment of a task/activity, the approval or rejection of a specific task, the delegation, and eventually a task escalation.
Some examples of these include expense approval, employee onboarding, employee performance, vacation approval, asset management, timesheet management, travel request, etc.
Business developers can implement these simple workflows with Workflow Builder.

What does the selection process look like?

In the “Getting Started” maturity level it is really important that you select a few initiatives that are more likely to be successful. Next, we propose a possible configuration for this initial proving maturity level.


IT Factory Admin / SPOC
Create a short list of candidate apps[1]
Execute the suitablility test for each and prioritize them
Select the first app and candidate citizen developer
Notify the candidate citizen developer and set up a kickoff session[2]

[1] Follow the guidelines in the “How to choose your candidate app” section.

[2] At this stage communications can be ahdoc, e.g. IT SPOC sends and email to candidate.

How to onboard these Business Developers?

In the “Getting Started” maturity level the processes to onboard business developers are not deeply structured and this is actually a time to start defining those and iterate. Some of the content used for this onboarding process will be discussed in other sections of this playbook and will have different requirements in other maturity levels.

Our recommendation is that a kickoff meeting is scheduled between the IT SPOC and the business developer with the following context:

- **Goals** - brief the business developer about the new initiative the organization is developing. Explain the goals and the important role the business developer will take in this initiative
- **Use cases** - define the use cases that are subject to the business developers participation and define the boundaries that require IT intervention
- **What is the platform** - introduction to the OutSystems platform and specific configuration prepared by IT, e.g. if Workflow Builder is the only tool they will use it should be introduced to the business developer
- **How it will work** - business developers do not have context about the typical software development life cycle, most of the time they won’t even have to. Still, some rules of engagement need to be defined from the start, e.g. how to get support, how to move applications into the production environment. Most of these processes are described in other sections of this playbook.
- **Communication channels** - it is important to establish a communication channel between the business developer and the IT SPOC. Our recommendation is to create a direct channel that reduces the time from question to feedback or resolution, for instance using direct email or a Slack channel. It is also important to define recurrent sessions to collect feedback from the business developer to improve on the processes defined.

1.3. Platform User Management

The definition of the privileges for business developers is another important process that IT, as the usual gatekeepers within organizations, should take into consideration. This process is clearly defined into two main operations:

Defining the roles of Business Developers

The definition of the business developer roles should take into account some considerations:

- Business developers will need to create applications, either through line-of-business development tools like Workflow Builder or through Service Studio or OutSystems consoles.
- Depending on the selected infrastructure configuration, business developers might share the same development environment as IT developers, therefore it is important to clearly define the boundaries of which shared services and components business developers can reuse.

Our recommendation includes the definition of specific roles for business developers and to allow reuse of share components made available by IT. We also recommend the definition of different teams for IT developments, shared components, and for business developers. In the latter case, you can also define different teams of business developers, e.g. belonging to different business units.

This recommendation is reflected in the table below.

TEAM	ROLE	DEV	TST	PRD
Default Role - Assigned when creating new IT user accounts				
N/A	**Basic Access**	**Access** Add Dependencies to Systems	No Access	No Access
Team Roles - Assigned whithin the scope of a Lifetime team				
Line-of-business Development Team	**Business Developer**	**Change & Deploy** Create Applications	No Access	No Access
Shared Components	**Reuse**	**Monitor & Add Dependencies**	No Access	No Access
Other Teams	**Reuse**	**Monitor & Add Dependencies**	No Access	No Access

The Shared Components team should include the very basic components required to create applications which include platform System Components like the Charts or OutSystems UI and basilar components created by IT in the context of the OutSystems Factory such as live style guides or generic widgets and components.

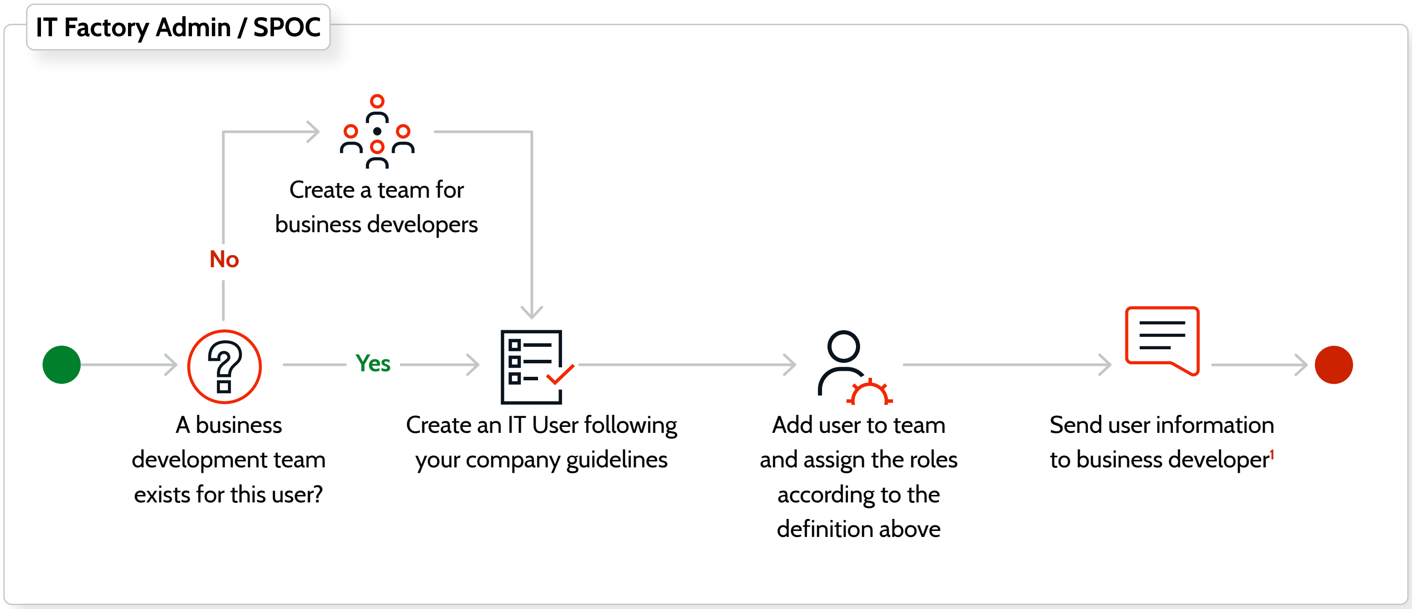
Additionally, you should create other teams responsible for the Core Business services and grant the same Reuse role on those teams according to the desired outcomes of line-of-business development. For example, you could have the HR Services team who is responsible for creating the shared services that give access to Employee data.

If you intend to use Workflow Builder, make sure you go through the documentation on how to set up the users governance model.

More information about setting up IT permission can be found here.

Creating users for business developers and assigning roles

Once you have selected your business developers, following the process described in the Selection and Onboarding section, follow the recommended process to create their users and assign the roles.


IT Factory Admin / SPOC
A business development team exists for this user?
No
Create a team for business developers
Yes
Create an IT User following your company guidelines
Add user to team and assign the roles according to the definition above
Send user information to business developer[1]

[1]- This includes username, password, workflow builder URL, development environment URL when applicable.

1.4. Foundation and Shared Services

Foundational and shared services are the support that IT provides to enable business developers to build their applications following the company guidelines and with governance in terms of access to systems and data.

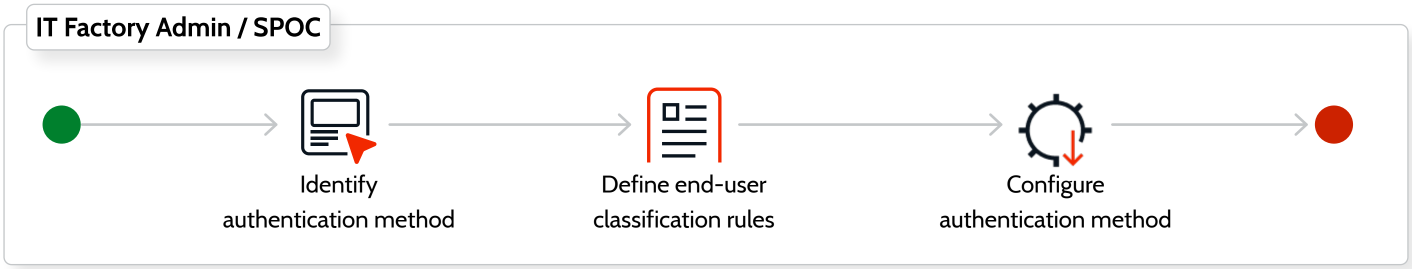
In the “Getting Started” maturity level it is important to establish some of these foundational services that will provide the necessary governance from IT.
We recommend defining and setting up the End-user Authentication mechanism.

End-user Authentication

Regarding end-user authentication there are several options that you should take into consideration and adapt to the guidelines of your organization.

If you have an existing OutSystems infrastructure most likely you have already set up the authentication method. In this case, you are good to go! Either if your business developers will be using Workflow Builder or creating applications using Service Studio, those applications will already use this authentication method you have set up.

If this is not the case, this process should look as follows:



IT Factory Admin / SPOC
Identify authentication method
Define end-user classification rules
Configure authentication method

After you have configured the end-user authentication method, later on you will need to create users and grant privileges to access the applications created by business developers just as you have to do for IT developed applications. We will cover this in the User & Access Management section of this document.

Identify authentication method

OutSystems supports out-of-the-box a wide range of authentication methods that include internal authentication, Active Directory, LDAP, Azure AD,
Okta, or SAML 2.0.

The default authentication flow in the application templates and on applications created with Workflow Builder will use this configuration, thus ensuring that your apps follow the sanctioned authentication method for end-users.

You can find more information about the supported authentication methods here.

Define end-user classification rules

End-user classification rules is a configuration aligned with OutSystems licensing model. OutSystems distinguishes between two different types of users: internal and external.

More information about end-user classification rules can be found here.

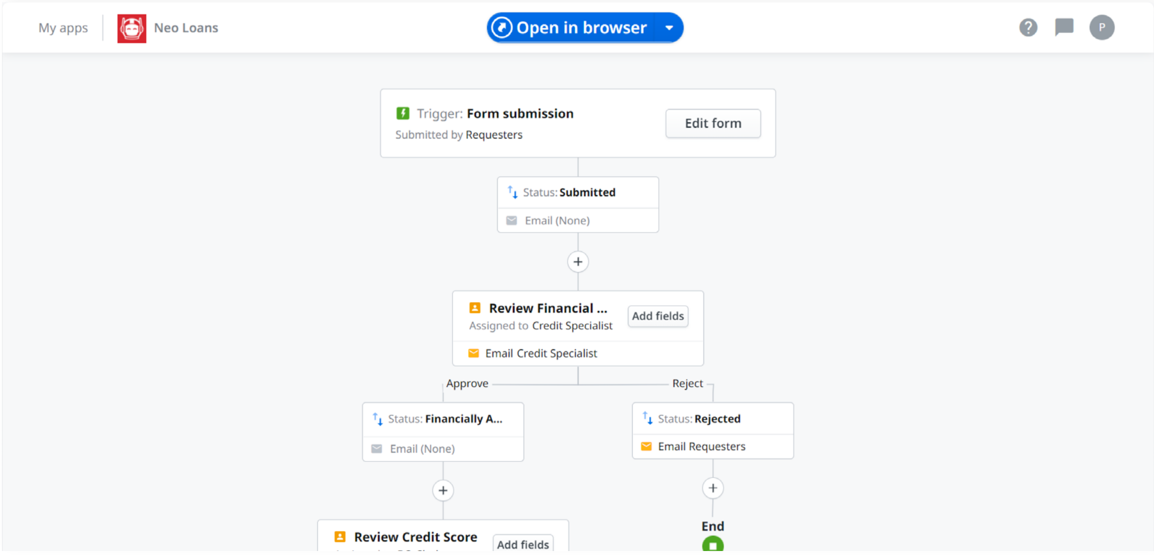
Configure authentication method

Now that you have identified the authentication method and defined end-user classification rules you can proceed to configure the authentication method in the platform.

Along with the shared link for our documentation regarding end-user authentication, we also provide a training course that you can find here.

Workflow Builder

In order to use Workflow Builder to develop simple workflow applications, you should make sure you have installed its dependencies. You will be guided during the setup process to ensure the dependencies are in place and you can find more information about this here.



2. Software Development Lifecycle

Now that the Setup stage is covered for this maturity level, the fun part begins.

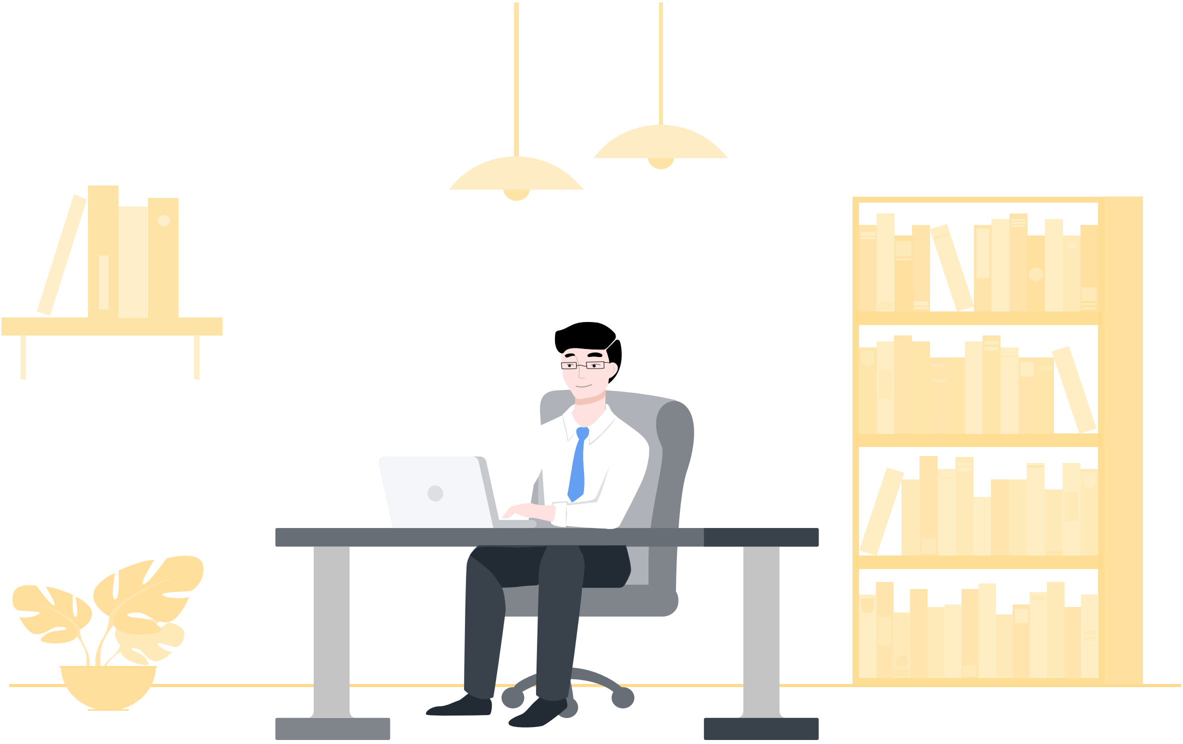
This section will cover the processes that happen during the development of applications by your business developers and that
you should consider and define.

Within this maturity level the processes for demand management, quality assurance, and change management are not operationalized but should start being designed. These processes will be addressed in full in Part II of this playbook.

Although we expect quality assurance activities to be mostly ad-hoc and consider this process does not require formal instructions during this initial maturity level, we provide an introductory guidance within Part I on how to adapt quality assurance to the reality of line-of-business development, with recommendations on testing and technical debt management.

2.1. Application Development

Depending on your strategy and use cases, line-of-business development will be developed in two different ways, with Workflow Builder or using Service Studio.



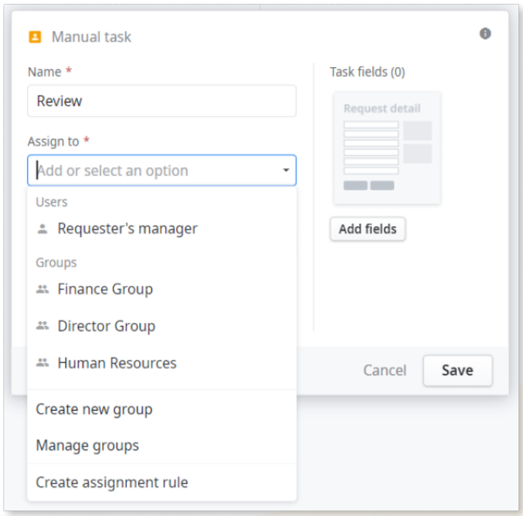
Using Workflow Builder

Workflow Builder is specifically designed for business developers and should require little to no training at all.

Applications created in Workflow Builder will require involvement from IT SPOC for User & Access Management and for Release & Deployment Management.

Business developers will login to Workflow Builder (http://workflowbuilder.outsystems.com) with the credentials and information you have provided them following the process defined in the Platform User Management section of this playbook.

In Workflow Builder, business developers will create applications that address simple workflow applications. This process basically includes the definition of a set of information present in a request form and the process that will handle the request submission. In the definition of this workflow, business developers will create groups of people that will participate in it.



Manual task

Name *: Review

Assign to *: Add or select an option

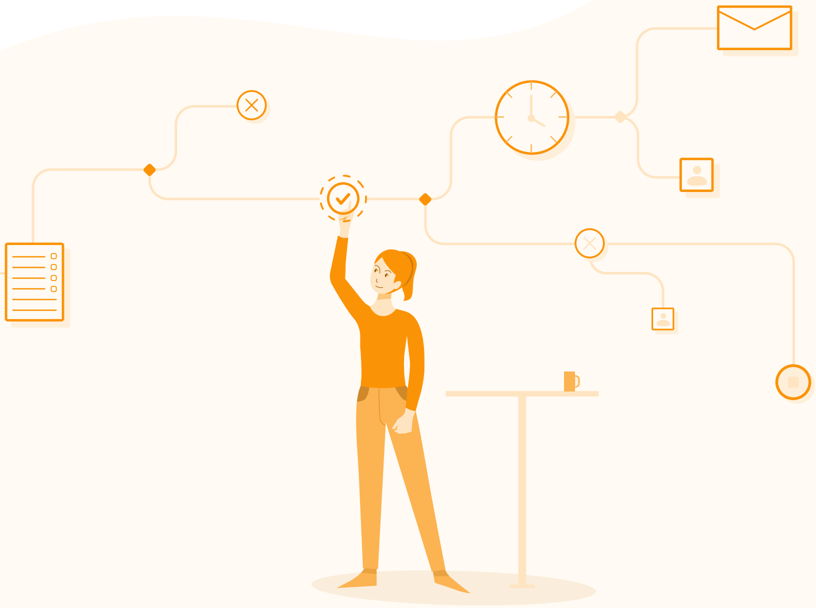
Task fields (0)

Once they have defined the form contents and the workflow, the business developer will publish the application into the environment you have configured for using Workflow Builder.

Workflow Builder will generate an OutSystems application, with proper architecture and following all the best practices. When published in the target environment, the application configurations are bootstrapped, including the creation of these groups and the creation of application roles, that we will detail in the User & Access Management section.

Once the application is published the business developer will be able to test it by using sample users created by Workflow Builder for this purpose. These users are only created and available in the development environment.

When the business developer has finished his changes and tests it is time to promote the application to the production environment. This will be covered in the Release & Deployment Management section further ahead.

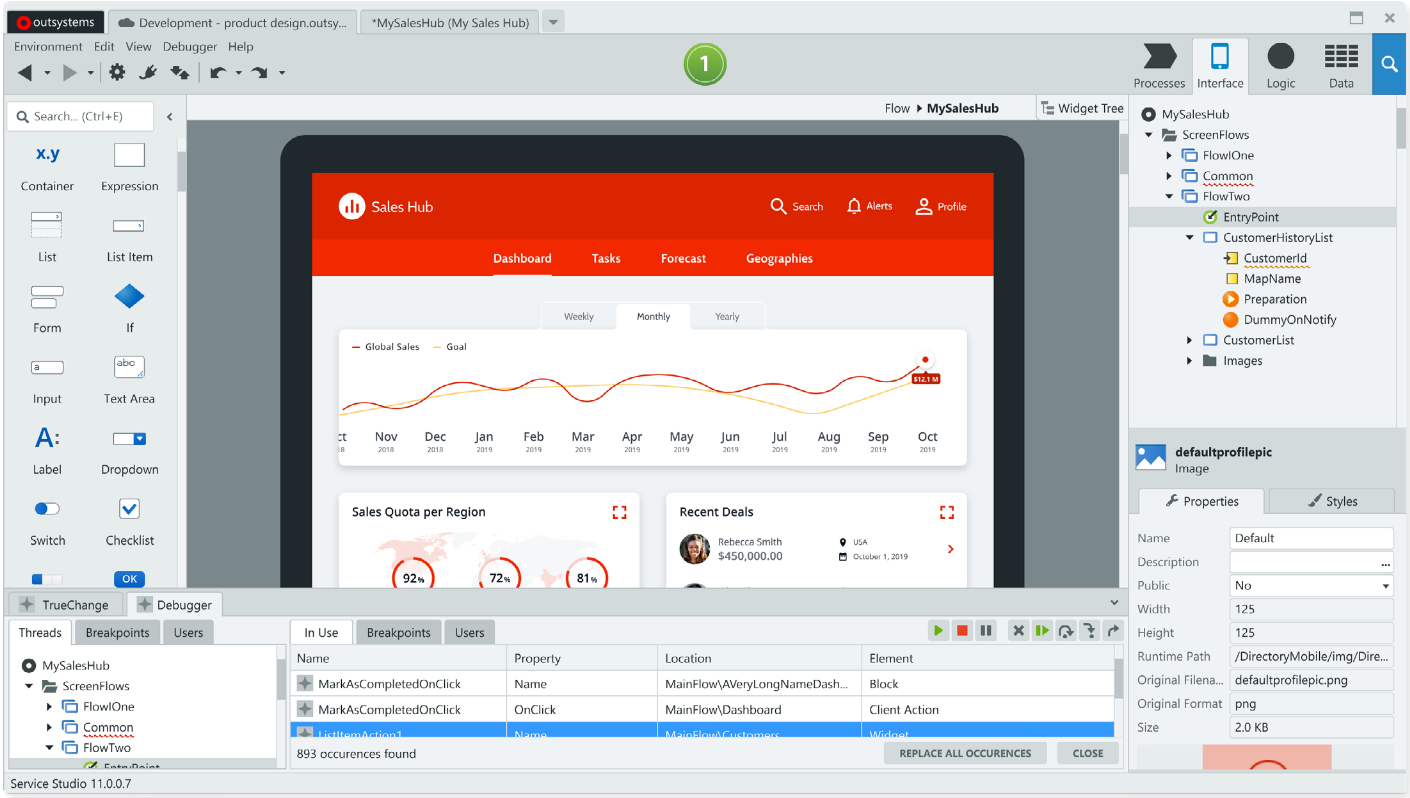


Using Service Studio

Before your business developers can start developing applications using Service Studio you will need to provide guidance on the training they should take. This information will be covered in the Training and Certification section of this playbook.

Developing applications in Service Studio shouldn't be much different for a business developer compared to a low-code IT developer, given the governance model defined by IT regarding what shared services and components business developers can reuse. Still, business developers will probably have limited skills for using extensibility, e.g. javascript, C#, etc., and will rely much more on the assets provided by IT.

Although the development part might be somewhat similar the complete SDLC will for sure be different for line-of-business development. Some of these differences are addressed in the following sections of this playbook and in Part II and III that will cover other maturity levels.



Name	Default
Description	
Public	No
Width	125
Height	125
Runtime Path	/DirectoryMobile/img/Dire...
Original Filena...	defaultprofilepic.png
Original Format	png
Size	2.0 KB

Name	Property	Location	Element
MarkAsCompletedOnClick	Name	MainFlow\AVeryLongNameDash...	Block
MarkAsCompletedOnClick	OnClick	MainFlow\Dashboard	Client Action
ListItemAction1	Name	MainFlow\Customers	Widget

893 occurences found

2.2. Release & Deployment Management

Once the business apps are developed and tested by the business developers they will need to be pushed into a production environment. This will be the case whether your business developers are building these apps in Workflow Builder or using Service Studio. Still, the process will have different configurations according to the development tool that is used.

Our recommendation is that you define this process carefully to ensure full governance from IT and that you explain how this process works to the business developers.

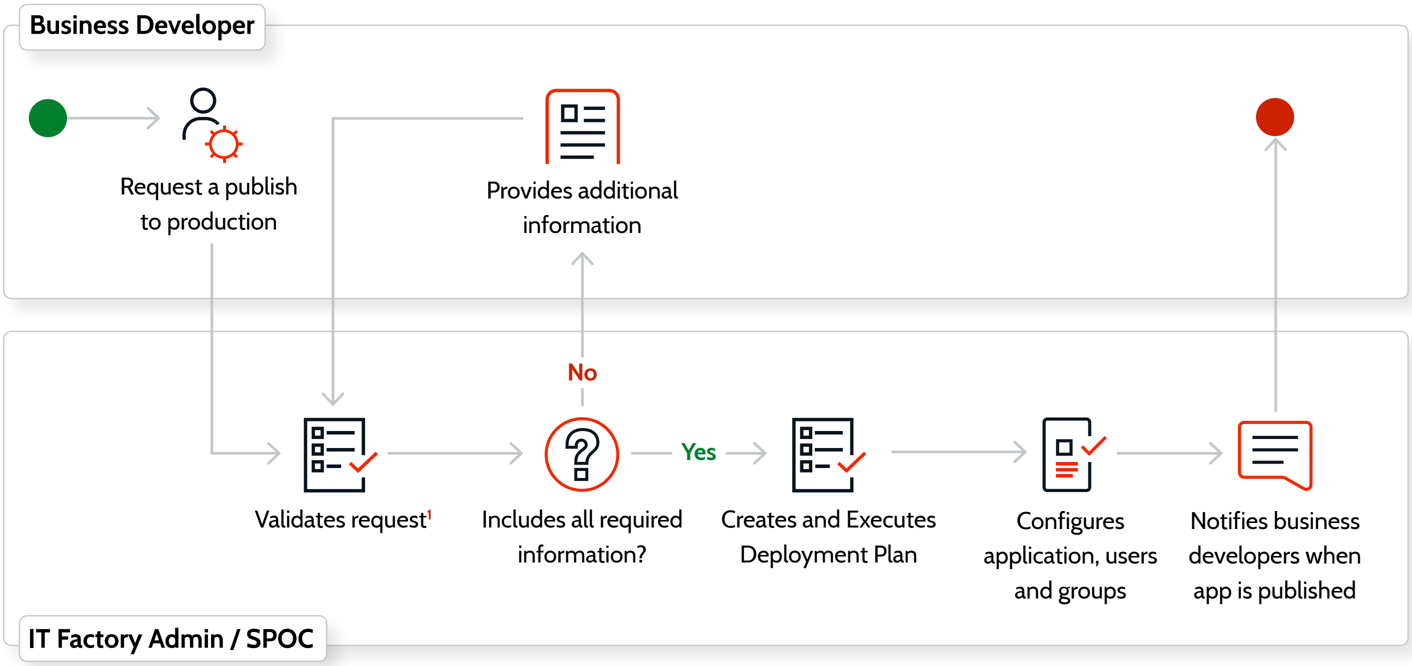
Regardless of the application development option the process starts with a request from the business developer to IT to deploy the application to production. You should consider how these requests are handled as there are several options for doing so:

- An existing helpdesk ticketing system
- An email sent to IT SPOC
- A new application built with Workflow Builder

More information about how to deploy applications can be found here.

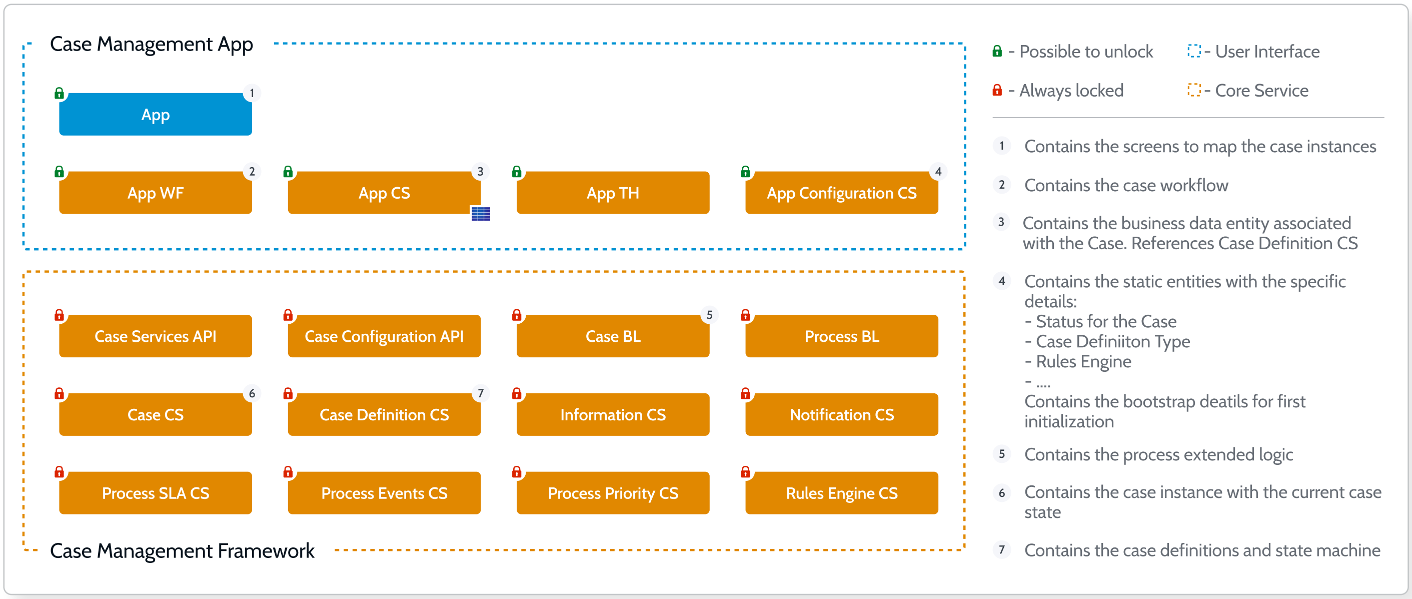
Using Workflow Builder

A typical process for deploying applications built with Workflow Builder into production would look like this.


Business Developer
Request a publish to production
Provides additional information
No
Yes
Validates request[1]
Includes all required information?
Creates and Executes Deployment Plan
Configures application, users and groups
Notifies business developers when app is published
IT Factory Admin / SPOC

[1]- Request should include name of the application and name of the (new) groups.

Workflow Builder will generate an OutSystems application with the architecture shown below which is dependent on the Case Management Framework. Therefore when deploying this application into another staging environment of your infrastructure this dependency will be validated by Lifetime.



🔒 - Possible to unlock

🔒 - Always locked

⬚ - User Interface

⬚ - Core Service

1 Contains the screens to map the case instances

2 Contains the case workflow

3 Contains the business data entity associated with the Case. References Case Definition CS

4 Contains the static entities with the specific details:
- Status for the Case
- Case Definiiton Type
- Rules Engine
- ....
Contains the bootstrap deatils for first initialization

5 Contains the process extended logic

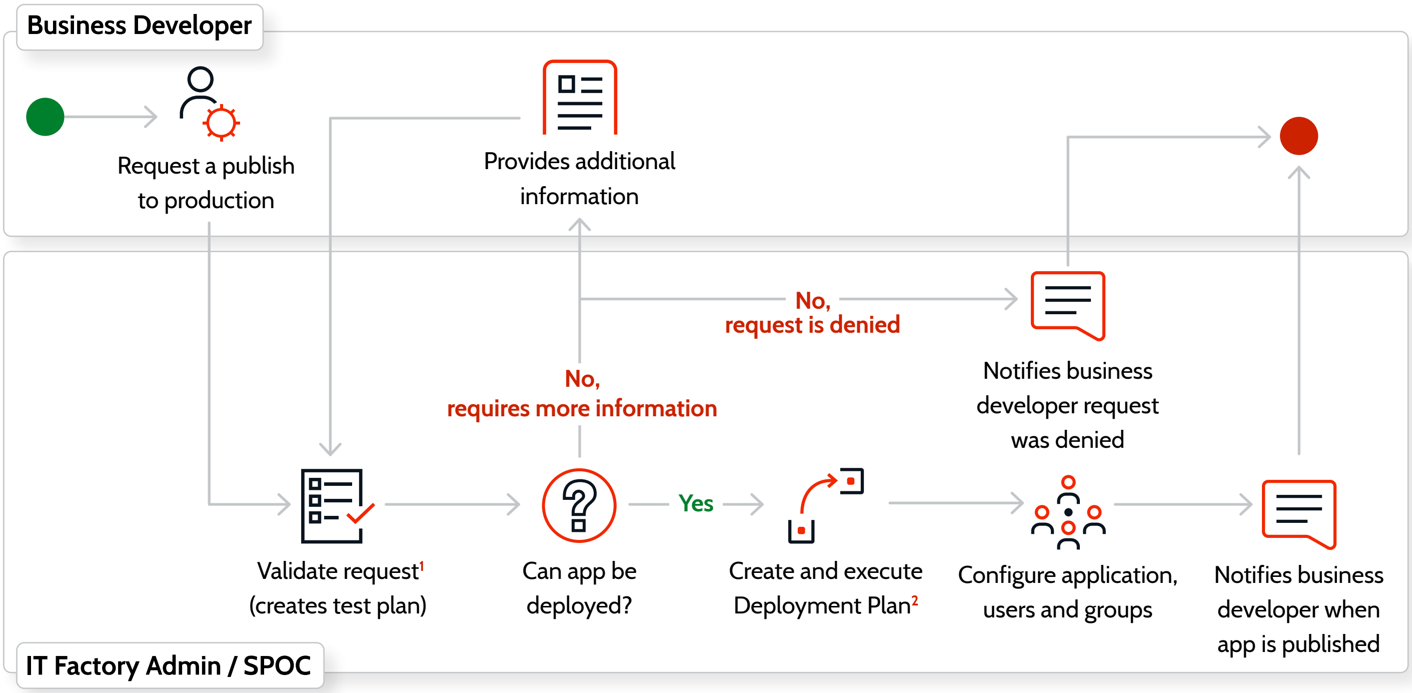
6 Contains the case instance with the current case state

7 Contains the case definitions and state machine

All the configurations of the application are bootstrapped on publish, meaning that in this scenario no additional configurations are required except for those related with User & Access Management.

Using Service Studio

A typical process for deploying applications built with Service Studio into production would look like this.


Business Developer
Request a publish to production
Provides additional information
No, request is denied
Notifies business developer request was denied
No, requires more information
Validate request[1] (creates test plan)
Can app be deployed?
Yes
Create and execute Deployment Plan[2]
Configure application, users and groups
Notifies business developer when app is published
IT Factory Admin / SPOC

1- In this case the validation includes the creation of a deployment plan and validation of dependencies inLifetime.

2- In some scenarios, a more traditional deployment pipeline is required, i.e. the app will first go through a QA environment and only then pushed to production.

If your business developers are creating applications using Service Studio, they have much more flexibility in terms of the outcomes of these applications.

It is important that you perform an assessment of the application before deploying it into production to prevent unexpected behaviours. This validation process should include a code review assessment that we will describe in more detail in the Quality Assurance section.

Part of this validation should also include the creation of a test deployment plan in Lifetime to ensure that all the dependencies are validated and that the application can be deployed in the target environment. In some cases, for which you should define a framework and thresholds, it may happen that this validation deems the application as not ready to be deployed. In that case you should inform the business developer and provide the reasons and guidelines to prevent future rejection of these requests. Possible reasons include:

- Application dependencies that are not yet deployed into the target staging environment - usually a shared service provided by IT that should have a clear release date communicated to the business developer

- Quality issues (described in more detail in the Quality Assurance section)

For applications developed using Service Studio you may want to consider a more traditional deployment pipeline. In that scenario the request follows the same process described above, but the deployment of the application is firstly executed into a QA environment - where some testing and user acceptance tasks are performed - and then the process is repeated until the final deployment is done into production.

Once the application is deployed, i.e. the deployment plan is executed, you might have configurations that you need to apply. Learn how to configure the application after deployment here.

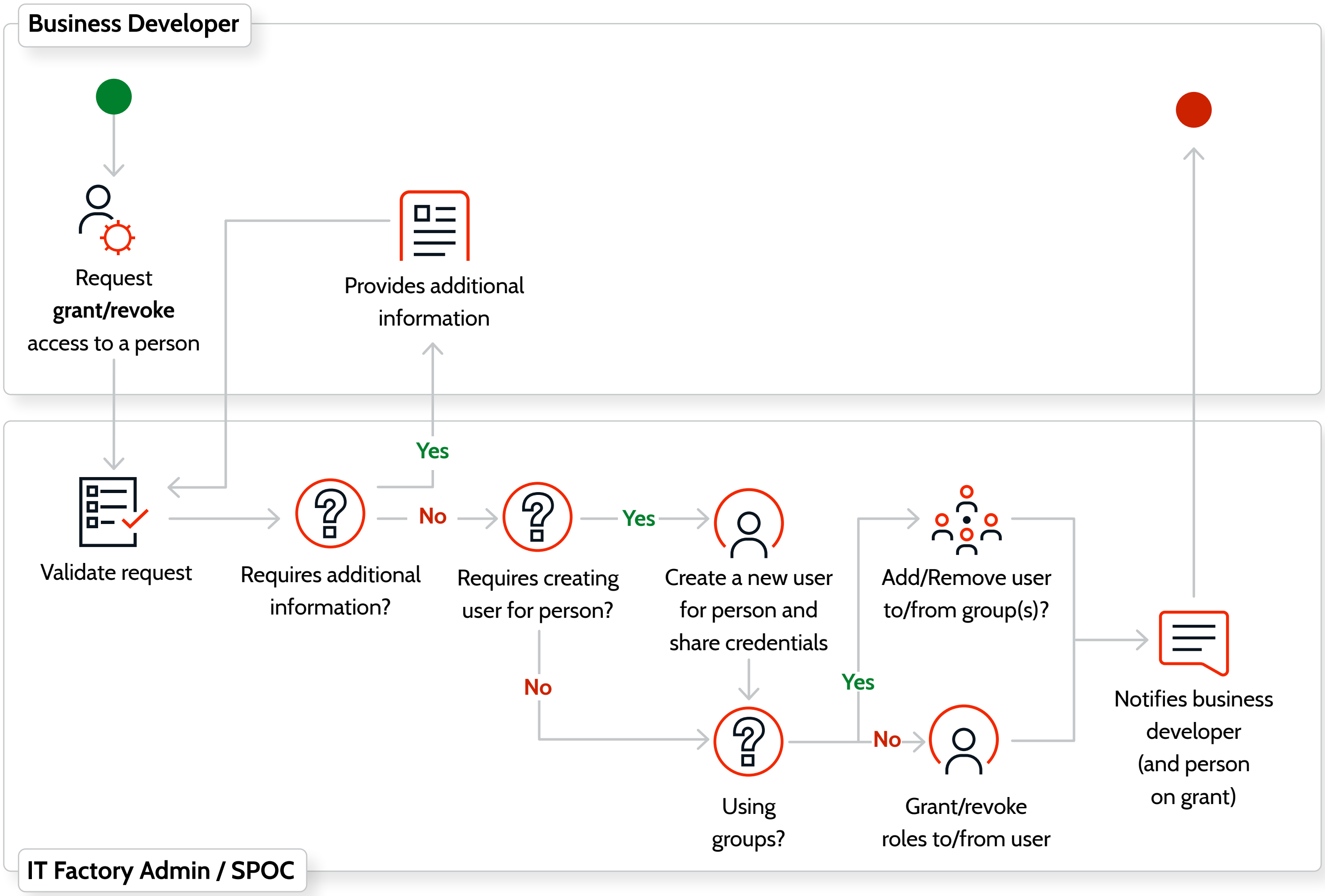
Optionally, you can configure any site property value during deployment. This is a feature that is in technical preview and that you will need to activate. Learn more about this feature here.

Additionally, once the application is deployed into the production environment, you will have to configure its users, roles, and groups, according to the application requirements.

2.3. User & Access Management

Once the application is deployed into production IT will have to manage its users and access privileges. This is a process that will happen several times throughout the existence of the application, and again a process that needs to be defined and communicated to the business developers.

A typical process would look similar to this one:


Business Developer
Request **grant/revoke** access to a person
Provides additional information
Yes
Validate request
Requires additional information?
No
Requires creating user for person?
Yes
Create a new user for person and share credentials
No
Using groups?
Yes
Add/Remove user to/from group(s)?
No
Grant/revoke roles to/from user
Notifies business developer (and person on grant)
IT Factory Admin / SPOC

Again, you should consider how these requests are handled as there are several options for doing so:

- An existing helpdesk ticketing system
- An email sent to IT SPOC
- A new application built with Workflow Builder

More information about how to create end users can be found here.

More information about how to grant and revoke end users roles can be found here.

More information about how to manage end users and organize roles using groups can be found here.

Using Workflow Builder

Workflow Builder will generate for each app a set of Groups, Roles, and Sample Users (in development environment only) that you should use for User & Access Management. The roles and groups generated are associated with each application and have the following structure:

ROLES
<AppName> Requester
<AppName> Operator
<AppName> Administrator

GROUPS	ROLES IN GROUP	SAMPLE USERS IN GROUP*
WB | <AppName> | Requesters	<AppName> Requester	Requester
WB | <AppName> | <GroupName>	<AppName> Operator	<GroupName>
WB | <AppName> | Administrators	<AppName> Administrator	Administrator

Making it clearer with an example, assuming we create an IT Expense Approval app with a validation from the Procurement and Finance groups, the roles, groups, and sample users created are:

ROLES
IT Expense Approval Requester
IT Expense Approval Operator
IT Expense Approval Administrator

GROUPS	ROLES IN GROUP	SAMPLE USERS IN GROUP*
WB | <IT Expense Approval> | Requesters	IT Expense Approval Requester	Requester
WB | <IT Expense Approval> | Procurement	IT Expense Approval Operator	Procurement
WB | <IT Expense Approval> | Finance	IT Expense Approval Operator	Finance
WB | <IT Expense Approval> | Administrators	IT Expense Approval Administrator	Administrators

More information about this can be found in Workflow Builder documentation.

*Created in the development environment only.

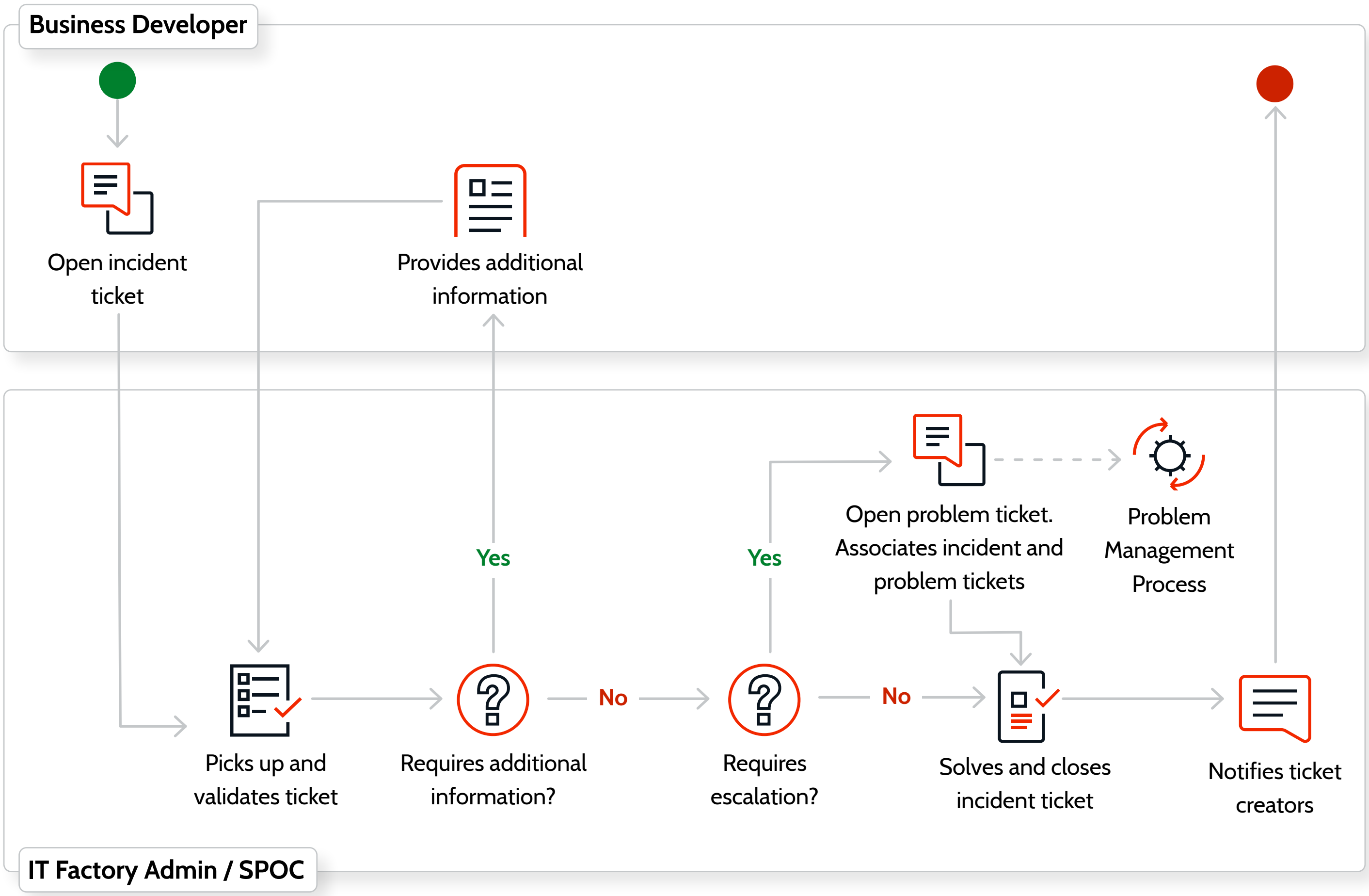
2.4. Incident Management

It is natural that during the participation of business developers in the creation of applications some questions, incidents, unexpected platform or app behaviours can occur. This is covered in the Incident Management process. This is a process that will happen several times throughout the existence of the application, and again a process that needs to be defined and communicated to the business developers.

Again, you should consider how these requests are handled as there are several options for doing so:

- An existing helpdesk ticketing system
- An email sent to IT SPOC
- A new application built with Workflow Builder

Below you will find a possible definition of this process:



Business Developer
Open incident ticket
Provides additional information
IT Factory Admin / SPOC
Picks up and validates ticket
Requires additional information?
Yes
No
Requires escalation?
Yes
No
Open problem ticket. Associates incident and problem tickets
Problem Management Process
Solves and closes incident ticket
Notifies ticket creators

In some cases, the issues reported by your business developers will require a deeper analysis of the issue and solution implementation which may lead to the creation of a problem ticket.

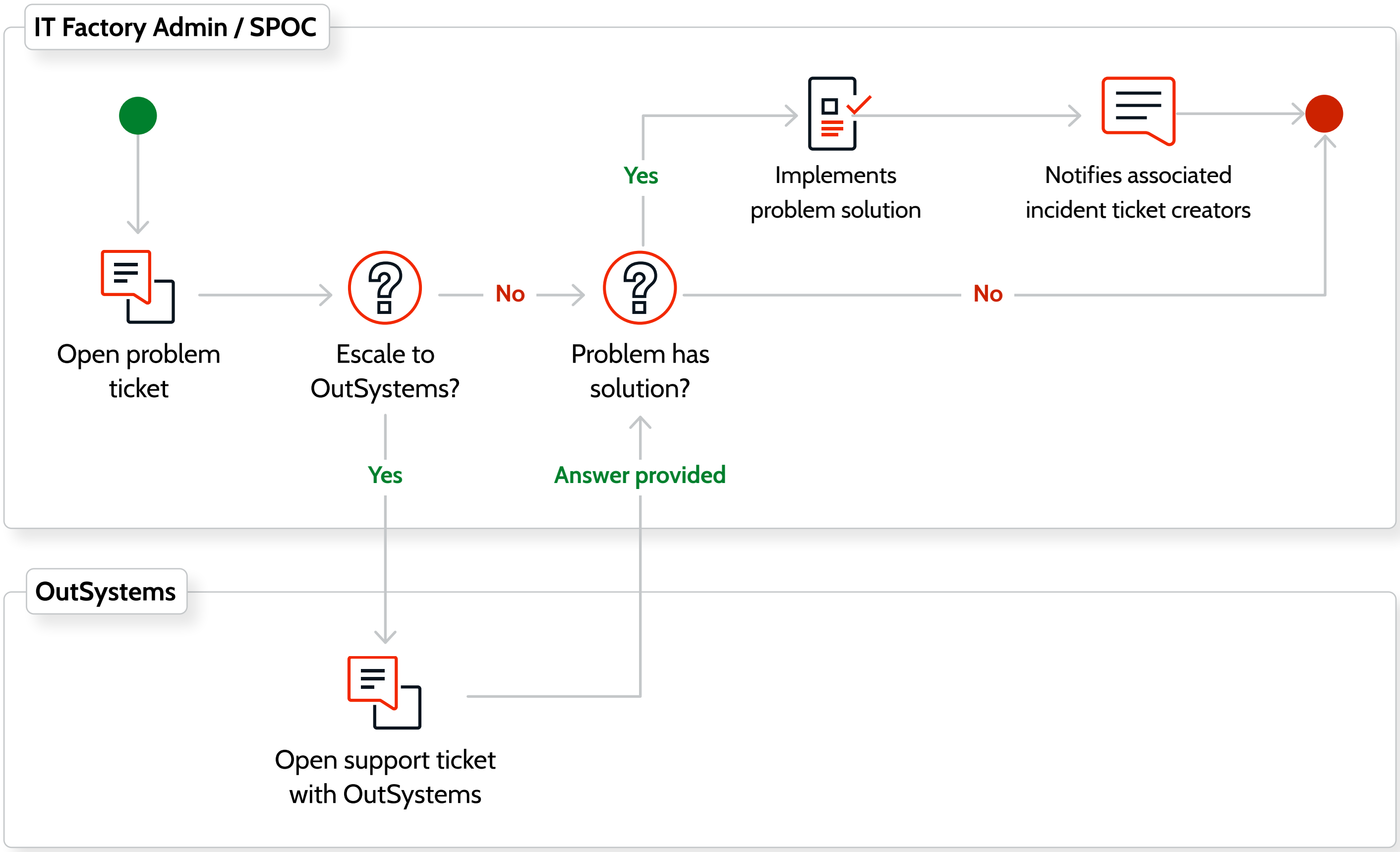
Problem Management

In some cases the open incidents can lead to the creation of a problem ticket. One of those situations is related with an issue that needs to be escalated to OutSystems due to one of the following reasons:

- Problem related with the usage of the OutSystems Platform and any of its components
- Problem related with the platform infrastructure (for OutSystems cloud customers)
- Problem related with the usage of Workflow Builder
- Problem related with an application generated by Workflow Builder
- Problem related with the usage of Case Management Framework

This is mostly an IT process that is usually not exposed to business developers except for the communication of its outcomes, e.g. notifying associated incident ticket creators when a problem is solved. Still, we recommend that you define or revisit this process in your organization.

A possible representation of this process would look similar to the following:

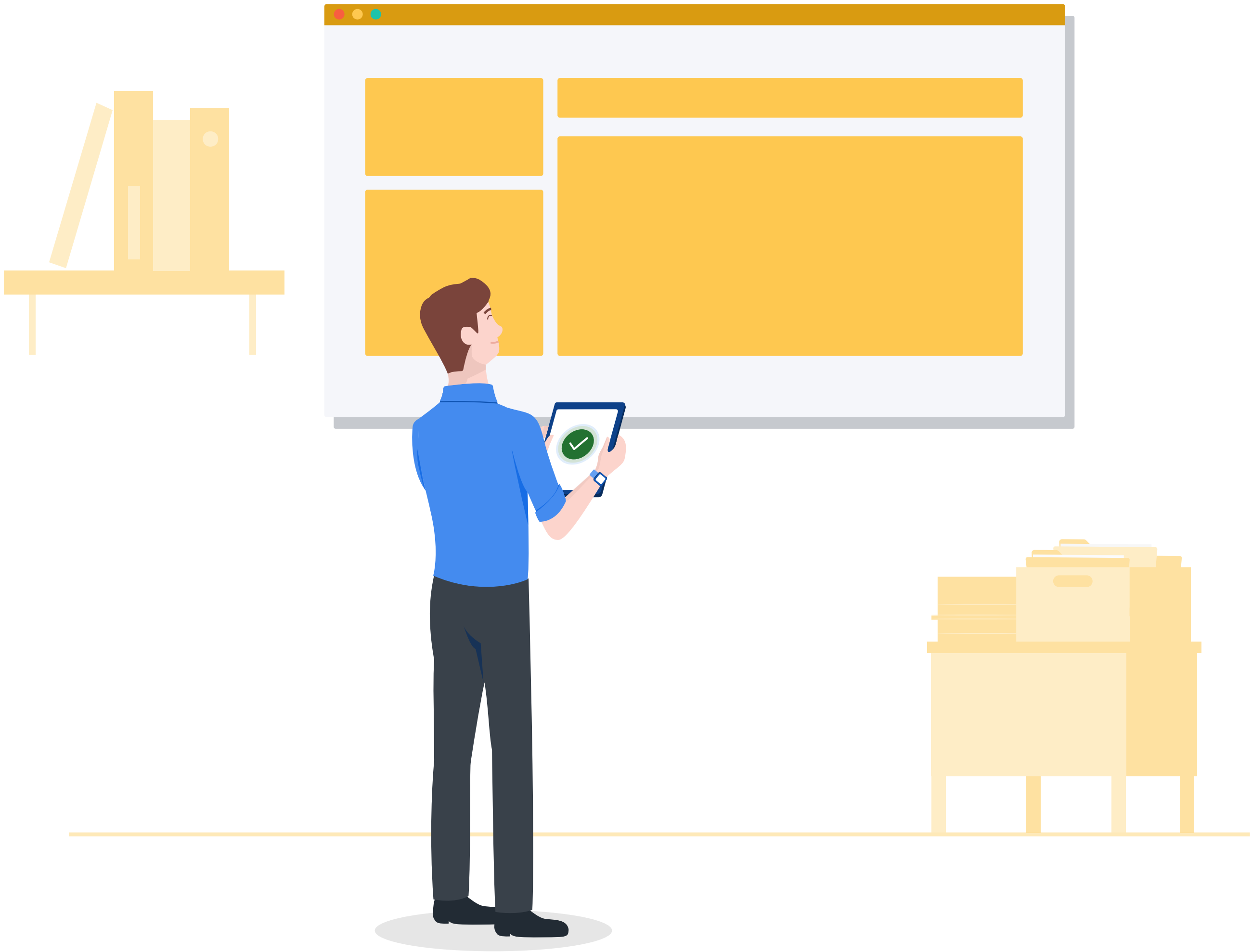


2.5. Quality Assurance

As part of the normal process of software development, quality assurance is definitely something you should take into consideration. Still, you should adapt it to the reality of line-of-business development. Asking business developers to follow a typical quality assurance process that IT follows is most probably an overkill given the skills of these business developers, their motivations, and the criticality of their applications.

Testing

We recommend that you find the right balance and assume a simplified quality assurance process for the business developer applications with just manual testing before deploying to production. Depending on how you set up your Release & Deployment Management process these tests can be executed in the development environment or a quality assurance environment. These tests are typically executed by the business developer himself/herself although they can include more end-users of the application, covering different roles.

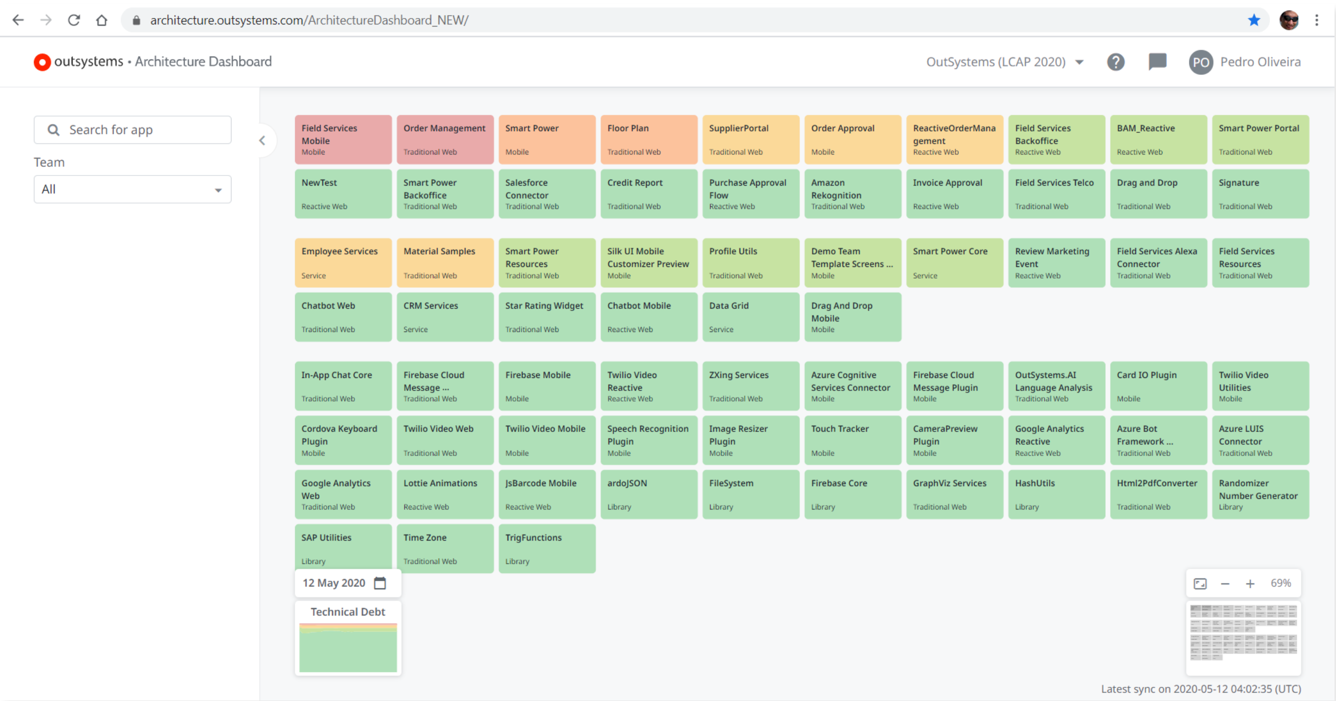


Technical debt management

Managing the technical debt is very important considering that some of the business developer applications will eventually become more mission critical and require ownership transfer to IT. This is why it is important to consider managing the technical debt of business developer applications since their inception, to avoid a future rewrite of those applications.

If you are using Workflow Builder you should know that the applications it generates have good architecture that follows development best practices and promotes reuse, maintainability and performance. The level of technical debt within these applications is very low, and as they transition to IT's ownership, IT developers can quickly understand their implementation and evolve them to satisfy new business requirements.

In the case that your business developers are creating their applications using Service Studio, they have the responsibility to create an application with good architecture and that follows development best practices. This is something that these business developers learn in their OutSystems training but it is certainly something that needs to be practiced every time they are developing.


architecture.outsystems.com/ArchitectureDashboard_NEW/
outsystems • Architecture Dashboard
OutSystems (LCAP 2020)
PO Pedro Oliveira
Search for app
Team
All

Field Services Mobile Mobile	Order Management Traditional Web	Smart Power Mobile	Floor Plan Traditional Web	SupplierPortal Traditional Web	Order Approval Mobile	ReactiveOrderManagement Reactive Web	Field Services Backoffice Reactive Web	BAM_Reactive Reactive Web	Smart Power Portal Traditional Web
NewTest Reactive Web	Smart Power Backoffice Traditional Web	Salesforce Connector Traditional Web	Credit Report Traditional Web	Purchase Approval Flow Reactive Web	Amazon Rekognition Traditional Web	Invoice Approval Reactive Web	Field Services Telco Traditional Web	Drag and Drop Traditional Web	Signature Traditional Web

Employee Services Service	Material Samples Traditional Web	Smart Power Resources Traditional Web	Silk UI Mobile Customizer Preview Mobile	Profile Utils Traditional Web	Demo Team Template Screens ... Mobile	Smart Power Core Service	Review Marketing Event Reactive Web	Field Services Alexa Connector Traditional Web	Field Services Resources Traditional Web
Chatbot Web Traditional Web	CRM Services Service	Star Rating Widget Traditional Web	Chatbot Mobile Reactive Web	Data Grid Service	Drag And Drop Mobile Mobile				

In-App Chat Core Traditional Web	Firebase Cloud Message ... Traditional Web	Firebase Mobile Mobile	Twilio Video Reactive Reactive Web	ZXing Services Traditional Web	Azure Cognitive Services Connector Mobile	Firebase Cloud Message Plugin Mobile	OutSystems.AI Language Analysis Traditional Web	Card IO Plugin Mobile	Twilio Video Utilities Mobile
Cordova Keyboard Plugin Mobile	Twilio Video Web Traditional Web	Twilio Video Mobile Mobile	Speech Recognition Plugin Mobile	Image Resizer Plugin Mobile	Touch Tracker Mobile	CameraPreview Plugin Mobile	Google Analytics Reactive Reactive Web	Azure Bot Framework ... Traditional Web	Azure LUIS Connector Traditional Web
Google Analytics Web Traditional Web	Lottie Animations Reactive Web	JsBarcode Mobile Reactive Web	ardoJSON Library	FileSystem Library	Firebase Core Library	GraphViz Services Traditional Web	HashUtils Library	Html2PdfConverter Traditional Web	Randomizer Number Generator Library
SAP Utilities Library	Time Zone Traditional Web	TrigFunctions Library							

Latest sync on 2020-05-12 04:02:35 (UTC)

To allow them to continuously monitor their developments we recommend that you include using Architecture Dashboard to monitor technical debt in two main processes:

- For business developers, as their technical debt trusted advisor, giving them the visibility of best practices they are not following and focusing on the enablement and learning of the business developers.
- For IT SPOC as an aid for the code review assessments we recommend to execute as part of the Release & Deployment Management process.

3. Monitoring and Event Management

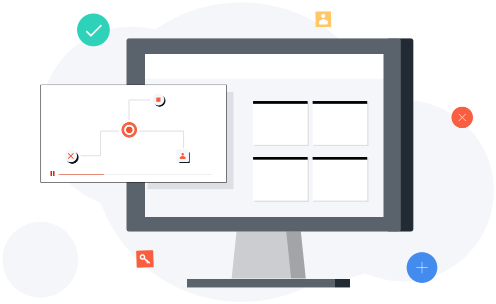
In the "Getting Started" maturity level, the monitoring and event management processes are very ad hoc and therefore we do not provide any formal recommendations for this stage. Still, during this maturity level, the processes for troubleshooting, auditing, and application support and maintenance should start being designed. These processes will be addressed in Part II of this playbook.

4. Continuous Delivery

In the "Getting Started" maturity level, the continuous delivery processes are very ad hoc, exceptions made for Training and Certification. This process will have a first iteration to support the training of business developers that will create applications using the OutSystems platform and whose use cases aren't completely covered with Workflow Builder. This process and the remaining ones on this stage will be addressed again in Part II of this playbook.

Training and Certification

OutSystems provides a full suite of courses and guided paths for enabling people to become OutSystems developers. Although much of this content is targeted to developers, some business developer profiles will be able to follow the contents and learn how to build applications using the OutSystems platform and in particular Service Studio.



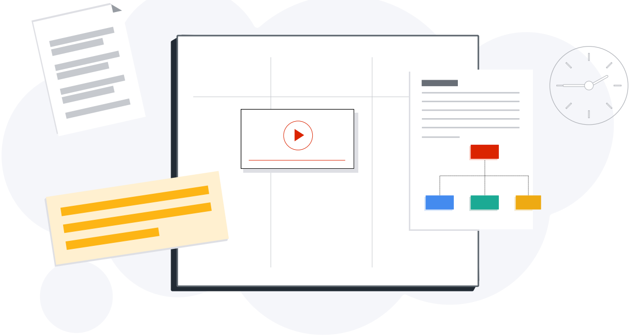
Guided Paths and Courses

We recommend the following courses and guided paths:

- Getting Started with OutSystems - an introductory course to the OutSystems platform. It provides an overview of the OutSystems Platform and an introduction to the Service Studio development environment.
- Becoming a Reactive Web Developer - an introduction to building web applications using OutSystems.
- Becoming a Mobile Developer - an introduction to building mobile applications that run on all platforms and even offline, using OutSystems.

Training Planner

Because different applications have different needs and not everyone's the same, the Training Planner allows your business developers to get a personalized training experience specifically tailored for the application that they are creating.



Closing

We hope that you have found this playbook useful. The purpose of it is to guide you through this journey and help you establish a successful and scalable motion to expand development outside of IT.

Still, this is a first release and a Part I of this playbook, and therefore we certainly expect to get a lot of feedback from you that will help us improve it.
Scan this QR code to provide feedback.

outsystems

www.outsystems.com