PLATFORMATION

Whitepaper

How to get from idea to business app in 10 days

Contents

Introduction

From working with our customers, we discovered that the majority of a workforce is spending huge amounts of time on labor-intensive manual processes that you can easily automate. Low-code platforms like ServiceNow are extremely suitable to solve such problems. Yet, these opportunities often remain unexploited because people believe that building good apps requires highly skilled developers and months of lead time. For many use cases, this is simply not true.

Building a great business app fast requires a couple of must-have ingredients though:

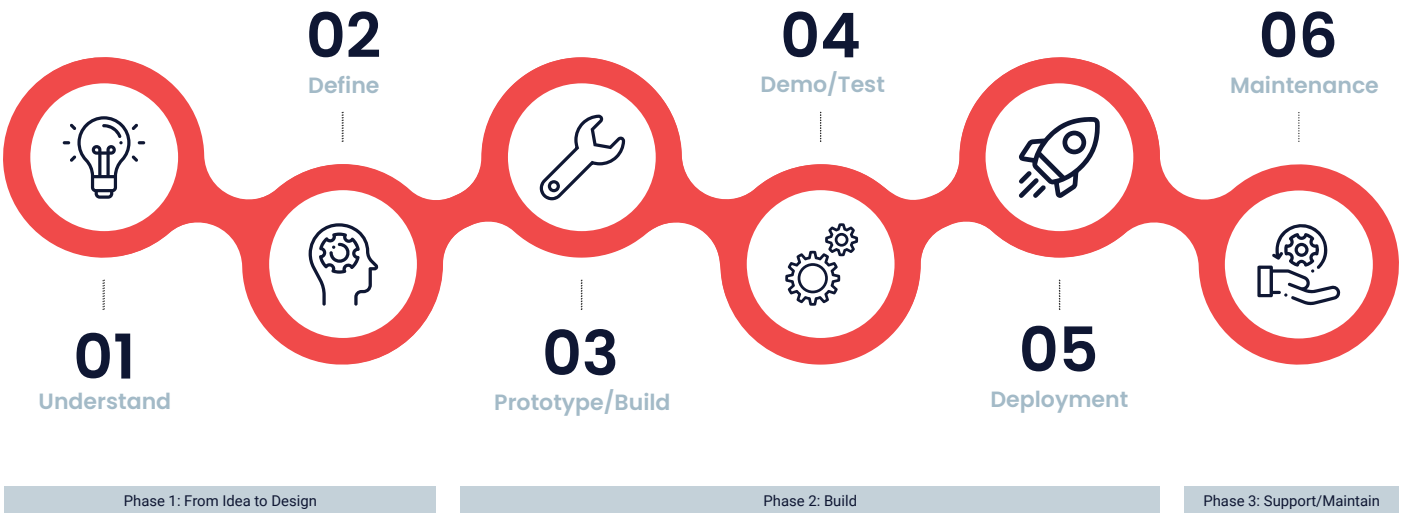
- A good understanding of the problem to solve
- Engaged users who are willing and able to test and provide feedback
- A platform which allows you to rapidly prototype applications (e.g. ServiceNow)
- An organizational culture that stimulates experiments

The approach outlined below is based on proven Design Thinking principles, optimized through years of designing and building business applications.

Meanwhile, at the office...

Abel from marketing complains his team is always super busy, but is asked what's keeping them so busy? He has no direct answer to that. He has to check his personal mail, the marketing mailbox, multiple overviews on the knowledge base, an Excel that was sent by his manager with strategic priority themes, and then he has his personal To Do list in yet a different app. Abel has no overview of what's going on.

Abel
Marketing Manager



Phase 1: From idea to design (inspired by Design Thinking)

3 days

Getting a good understanding of the situation is essential for creating a good design. What is the problem we are solving? Who are we helping by solving this problem? What are we trying to achieve with the solution we are aiming to build? This requires close collaboration with the people who are going to use the application.

Understanding the situation

The first step is to talk with these people to understand what they are doing, why they are doing it and in which context and process they are doing it. Capturing a summary of this information in a simple format (e.g. 1-3 PowerPoint slides) helps to identify the essence, and allows to validate with the user whether you've understood them correctly.

This document then describes the as-is or situation. In Design Thinking, this step is called Emphasize. While talking with the users, the problems at hand (Complications) and the desired end state (To-Be) will already become clear, which is the focus of the next step.

Meanwhile, at the office...

Over lunch, Abel has shared his frustrations with Julia, a new colleague who is still learning about ServiceNow. He starts thinking: what if we would have a tool that provides us with a single task overview including task assignment feature? Abel schedules a meeting with Julia to bounce off his idea and ask her what kind of tasks the marketing team is executing, what takes up most of their time, and which frustrations they have.

Abel
Marketing Manager

Defining the problem

Now, we're going to capture the problem or use case in a Problem/Value Statement. This is also referred to as the Complication. In Design Thinking, this is the Define step.

Want to get right down to the essence? Ask yourself **Who, What, Where** and most importantly **Why?** This will help you take the user perspective.

Note down your Problem/Value Statement in one or two PowerPoint slides.

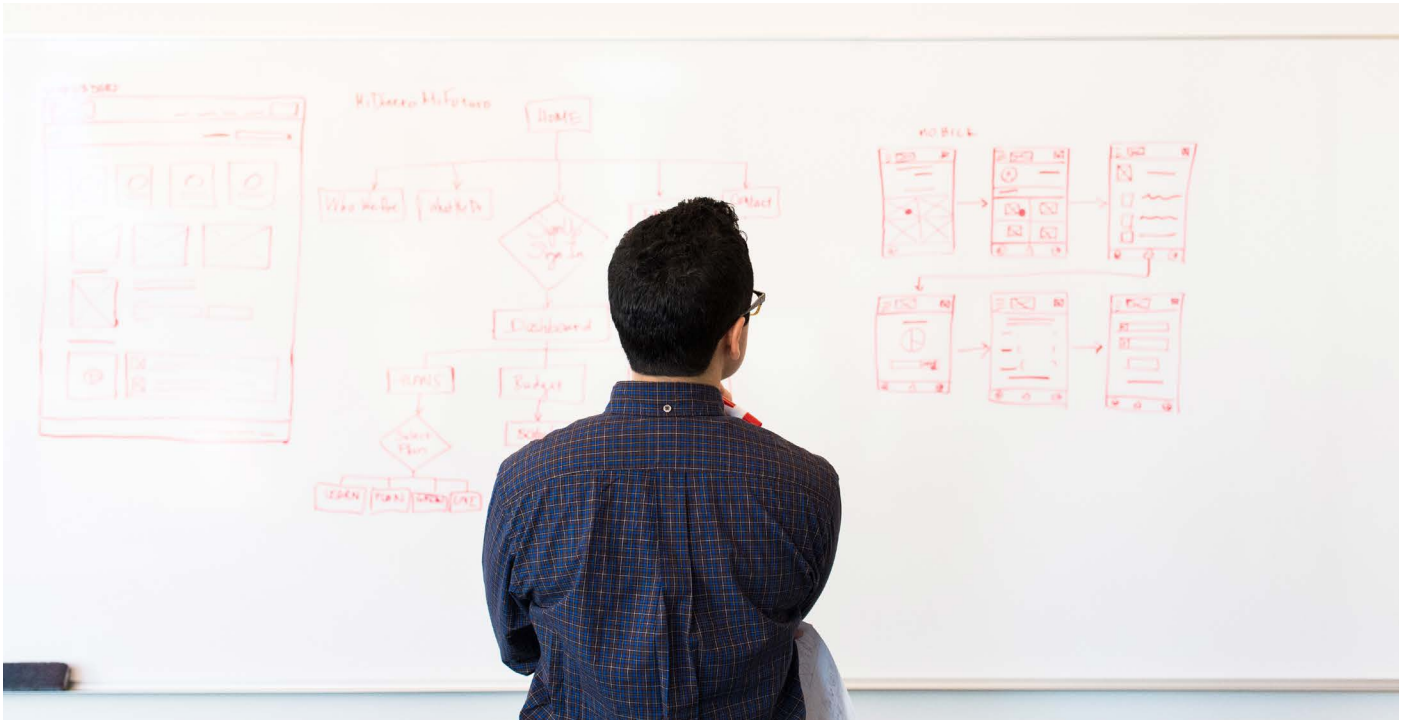
Next, we're going to break down the problem in smaller chunks. We suggest to define these sub-problems or use cases in Epics or maybe User Stories. For complex problems, a useful method could be Story Mapping. Engage with the users to prioritize this list of Epics and User Stories, and start defining the scope for the Minimum Viable Product (MVP), where the most important guiding principle should be: less is more!

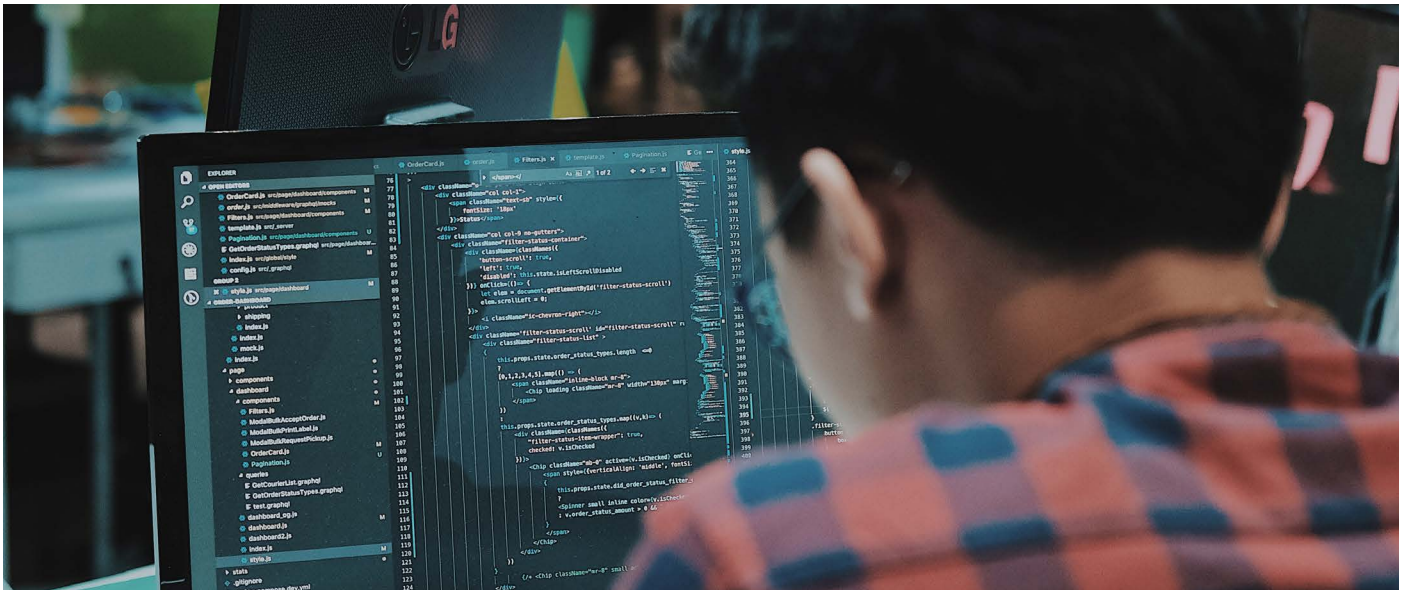
Meanwhile, at the office...

Abel summarizes all that Julia has told him in a simple flow chart with individual activities and arrows that indicate the relations. He also creates a list of the problems that Julia mentioned, and the repetitive activities of gathering information, and creating overviews. He presents the summary in the Marketing team meeting and gets some immediate feedback and suggestions.

Taking the feedback into consideration, they prioritize the list of problems according to team impact. Their biggest problem is that support requests are coming in unstructured. In many cases, essential information is missing too, which they need to collect later. Abel points out that he wants to add the flow chart to the marketing knowledge base which they use to document all incoming support requests.

Abel
Marketing Manager





Phase 2: Build (prototype + test)

5 days

On a low-code rapid development platform like e.g. ServiceNow, building a prototype is typically already building the core of the actual application. A prototype allows for quick user validation and a fit for purpose check.

Building the prototype

Three components are required to be able to present a first working version:

1. **Data model -** The tables to store the data
2. **User Interface (UI) -** Determines how the users can interact with the data
3. **Logic -** The automations, (business) logic and workflows that bring the app to life

You can create above components quite easily using a low-code platform, as these are often already pre-built. Particularly for a prototype, but in general as a rule of thumb, we recommend to work with the available standard components.

Start with building the backbone with some basics first—only the tables and a couple of fields. Add the UI components, and present a preliminary solution to get early feedback. This will lead to a better solution, and prevent time wasted on superfluous features.

Meanwhile, at the office...

Based on her understanding of what the app needs to do, Julia uses ServiceNow App Engine Studio to quickly generate a new application. For the first version, she uses a Case Management template to quickly generate the tables, UI and notifications which are used to manage all unstructured requests. In addition, Julia creates a new Catalog where she adds an item to "Request Help from Marketing". The next day, she also adds the two most frequently requested items as Catalog Items in ServiceNow.

Julia
Marketing Professional

Testing

Whenever a working version is ready, involve the users to test the new functionality. When executed correctly, testing actually serves multiple purposes:

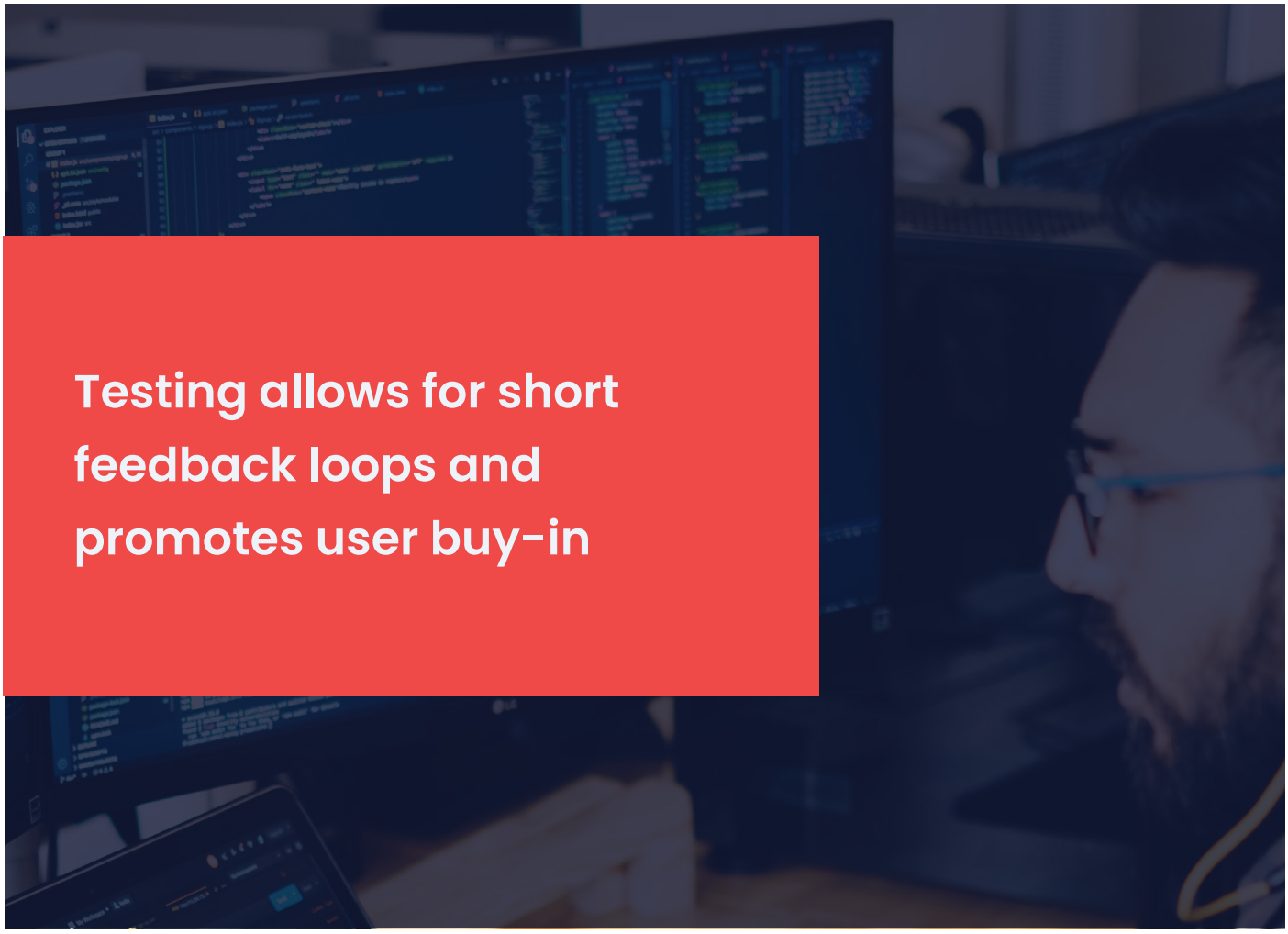
- **Training -** Letting users discover the application will learn them how to use it.
- **Functional validation -** Users can provide early feedback on whether the app is fit for purpose.
- **Technical validation -** Testing will help identifying bugs and check if the app is fit for use.
- **Inspire -** Users can spot and propose potential improvements.

Guiding users in the testing phase is crucial to avoid users getting lost or frustrated. Start with a demo, to showcase the functionality and guide your users through the use cases. Next, provide the steps to test the happy flow, and explain they are free to click around and explore potential issues outside the happy flow.

Meanwhile, at the office...

Proud of what she has build so far, Julia organizes a short demo with Abel and one of her teammates. She shows how all employees can now create standard requests, how the marketing team can assign all incoming work, and how all the work shows up in a single overview. Abel is excited, and notices how easy it is to make these small changes. "I could probably even do this myself!" Julia's teammate points out that the security still needs to be set up, after which they enforce a security model.

Julia
Marketing Professional



Testing allows for short feedback loops and promotes user buy-in

Make sure testers can easily report on any findings (e.g. test results, enhancements/defects, etc.).
Once you have the test results in, it is important to properly classify and prioritize the results. Possible classifications can be Defects (bugs), Enhancements or Instruction/Documentation requirements. Important for prioritization is to check their impact and then determine whether they can be solved at a later stage or not. A good prioritization model is the MoSCoW method. Any must-have requirements should be implemented before moving on.

Going back and forth between building/prototyping and testing is typically good practice. It allows for short feedback loops, promotes user buy-in and user adoption.

Meanwhile, at the office...

Julia has implemented the security rules, and has moved the app to the UAT (User Acceptance Testing) environment. While Abel and his team are trying the app out in the test environment, the ServiceNow platform owner takes care of the technical implementation. Abel indicates he would like to add some notifications, but he's also eager to start using it so he agrees to add those later.

Julia
Marketing Professional

Deployment

Once it is time for the deployment of the application, Change Control or Change Management processes are usually in place to avoid issues, outages and security and compliance breaches. Yet, with an app validated, tested and ready to deploy, nothing is more frustrating than having to wait weeks before being able to use it. By applying the DevOps practice of CI/CD and automating the Change Control process, you can move at the Speed of DevOps whilst staying in control.

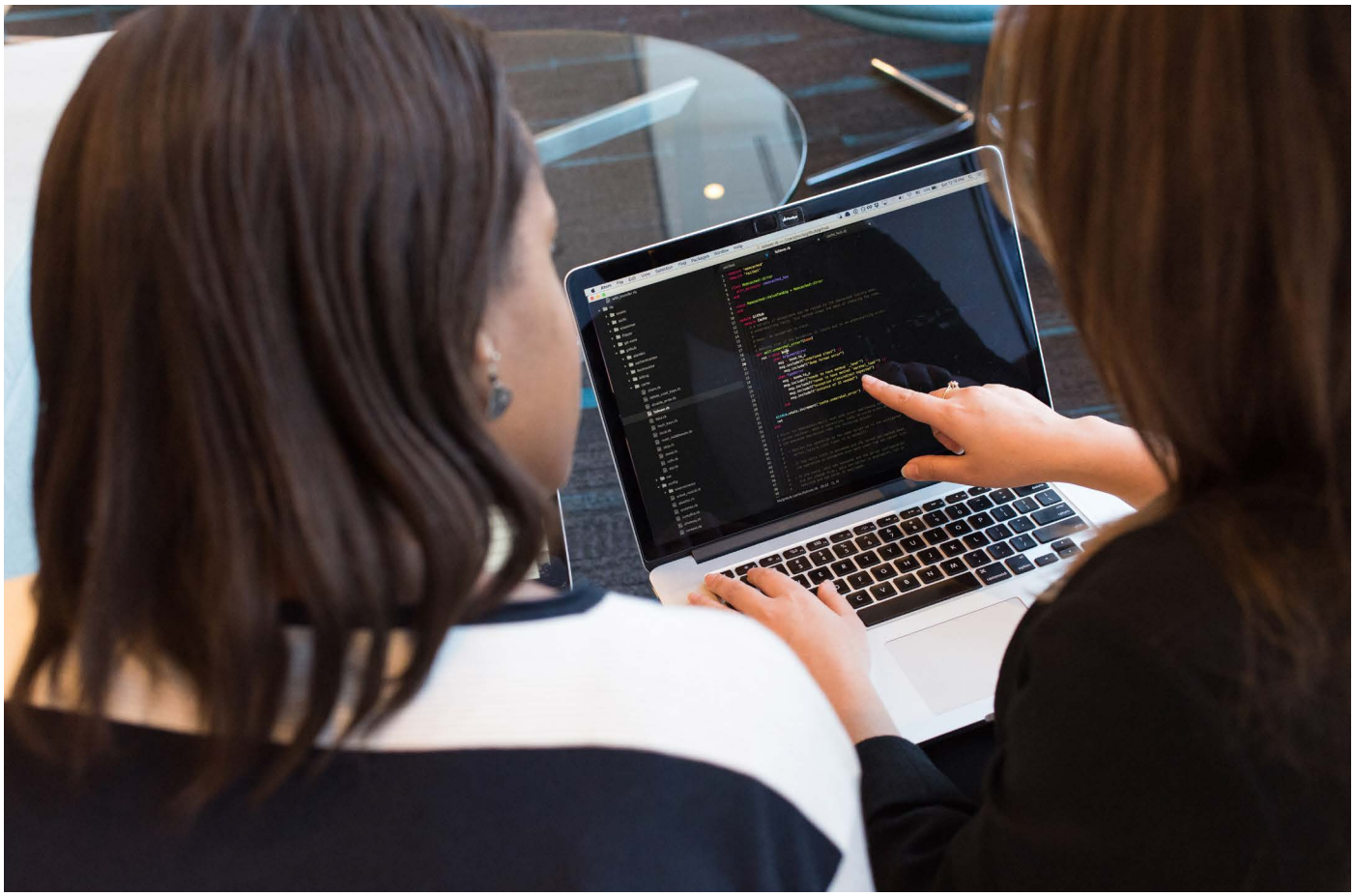
Safeguard the quality of your application, prevent human errors, and get buy-in from the Change and Compliance Officers by building Quality Control steps into the deployment pipeline.

Modern platforms like ServiceNow offer these capabilities and allow you to automatically run Instance Scans and Automated Tests as part of your deployments. Moreover, some of these quality checks can be configured to run during development, shortening the feedback loop even further.

Meanwhile, at the office...

After consulting with the Platform Owner and Change Manager, Julia agrees with Abel that the app can be deployed without further complications. It is an isolated application which isn't business critical. They register the new app in the ServiceNow CMDB and a change ticket is raised to deploy the app. For now, the deployment is done manually, but they agree to explore automated deployment for this application.

Julia
Marketing Professional



Phase 3: Support / Maintain

2 days

Ensuring proper support and maintenance is critical for the success of any type of implementation. An appropriate support and maintenance structure could range from identifying a single person to contact in case of issues to setting up an entire support organization.

To identify what type of structure is most applicable, consider the following questions:

- Who will be using the application and how large is this user group?
- What is the business impact if the app were to be unavailable for some time?
- Will the app undergo changes or approvals in the near future?
- Who is sufficiently familiar with the app to solve any problems, should they occur?
- How is support and maintenance organized for similar apps in your organization? Does this work well?

Considering these questions can help clarify the picture of what an appropriate support and maintenance organization should look like. While it is good to start asking these questions in an early stage, one should avoid making a full governance plan a prerequisite for getting started, as this will kill the efforts to add value quickly.

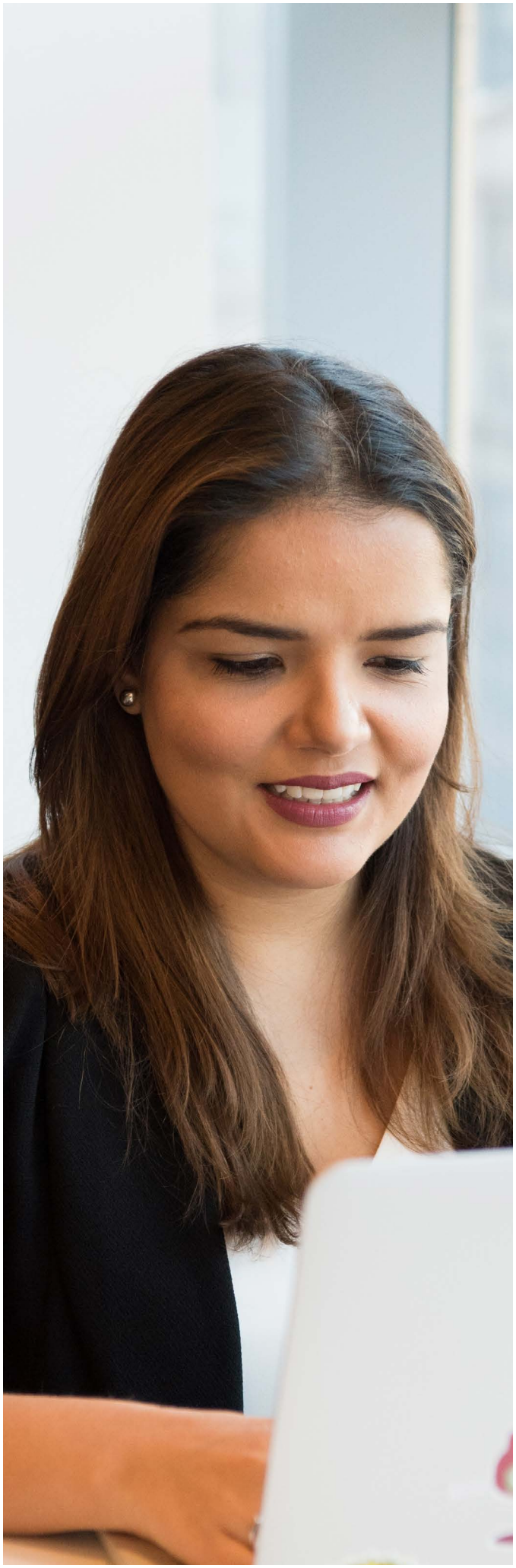
Meanwhile, at the office...

Abel is really happy with the new Marketing Services app, as it provides a much better overview of what they are working on. They're saving loads of time now that they don't have to chase requestors for missing additional info anymore. Abel now has a long list with additional catalog items he wants to add and small improvements he wants to make. Since Julia currently has a lot of other projects on her plate, Julia suggests that Dave, who is the tech-savvy one in the Marketing team, could in fact help in expanding the app. Dave, who suggested most of the improvements anyways, agrees and after completing a few of the online training exercises, he now does the development and support/maintenance.

Abel
Marketing Manager

Key takeaways

1. Make sure you take the time to thoroughly understand the user and his problem. This is not a one-off action, but a process of continuous involvement.
2. Start small! Scope a Minimum Viable Product that can be delivered fast. This allows for quick feedback and reduces possible waste.
3. Use a modern low-code platform like ServiceNow App Engine to boost your development to keep your applications Reliable, Scalable and Secure.



About Plat4mation

Plat4mation is a global pure-play ServiceNow partner that makes work flow. Our goal is to boost workforce productivity and happiness by implementing solutions that provide great experiences and ensure work flows intelligently across and between organizations. Furthermore, we future-proof our solutions by enabling our customers to continually measure and improve performance.

Besides offering world-class ServiceNow Implementation and Managed Services, we provide thought leadership to strategically guide our customers to take full advantage of all ServiceNow capabilities. Our award-winning ServiceNow Application Development Services ensure we can transform any business need into a powerful digital business application.

Author

Guus Hutschemaekers
guus.hutschemaekers@plat4mation.com

Contact

Plat4mation BV
Arthur van Schendelstraat 650
3511 MJ Utrecht, The Netherlands

+31 30 760 26 70
info@plat4mation.com
www.plat4mation.com

2021 GLOBAL PARTNER AWARD WINNER now
App Development