5 Key Steps to Managing Software Roadmaps

VOLUME #3

MISSION CONTROL

Contents

MISSION CONTROL

Introduction

Established in 2010, Aprika Business Solutions started as a Salesforce consulting practice, which evolved into software development building high quality solutions for the AppExchange.

Mission Control is built natively on the Salesforce Platform, and has grown into one of the top Project Management and Professional Services Automation (PSA) solutions supporting businesses to elevate workflow and maximize efficiencies and profitability in over 25 countries across the globe.

One of our key specialities is being able to support other software product management and development businesses as this is at the core of what we do. Each day we are immersed in the product development lifecycle for our own software so we intrinsically understand the process and roadmap to make software development a success.

We've put together this eBook based on our own experience in this space alongside the benefits of utilizing a quality project management solution to support with the execution and key features to look for along the way.

Colin Johnson
CEO & Co-Founder
Aprika Business Solutions

@aprika_colin

MISSION
CONTROL
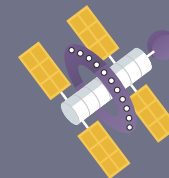
Strategic Review

🔖 Overview

When it comes to product development, particularly software development it can be easy to forget the all important strategic review side. You get lost in the excitement about new feature ideas and focusing on current feedback without stepping back and reminding yourself about the original goals and vision for the product.

The main elements to consider when reviewing the strategic direction for your software or product includes:

Goals

We release three versions of our software a year but we always make a point at the start of each year to remind ourselves about our strategic goals for the wider business as well as the product itself. It is important that you stay on track with what you ultimately want to achieve as a business and not get side-tracked with development ideas that don't add true value.

Industry

Keeping an eye on your respective industry and developments is a smart move before entering into another cycle of development. Sometimes there can be big industry changes that impact your customers behaviour and potential risks and threats that pop up through new technologies or Government policy that need to be considered and reviewed.

Competitors

Having a fresh look at what your competitors are doing in the space can be vital for development success. You want to ensure you are staying ahead of the development process and even if you have similar new features or requirements you want to include it's a great research project to look at how you would improve on what is already out there.

Once you've completed your review of the landscape you are in a good position to get into the actual lifecycle of the process.
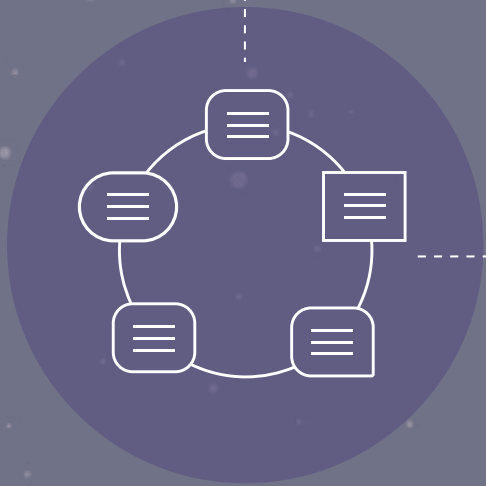
The Five Steps
of The Lifecycle

The Five Steps of The Lifecycle

Firstly, we want to stress the emphasis on the process here being continuous and an opportunity to keep evolving and whilst getting a new release of your software or product out of the door is a great achievement it doesn't end there. For us by the time of the release day we are already into the thick of the development process for our next iteration.

Within the next sections of the eBook we will deep dive into the following process steps, which can also align with any successful product management strategy utilizing a project management software.

ONBOARDING

COMMUNICATING

LISTENING AND IDEATION

DEVELOPMENT

PLANNING
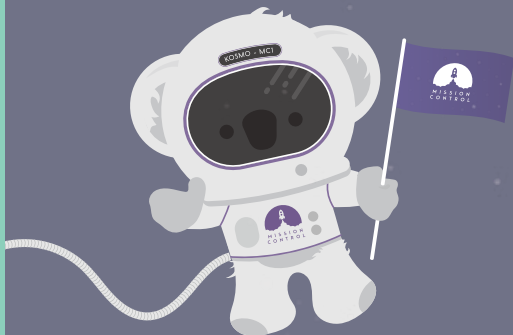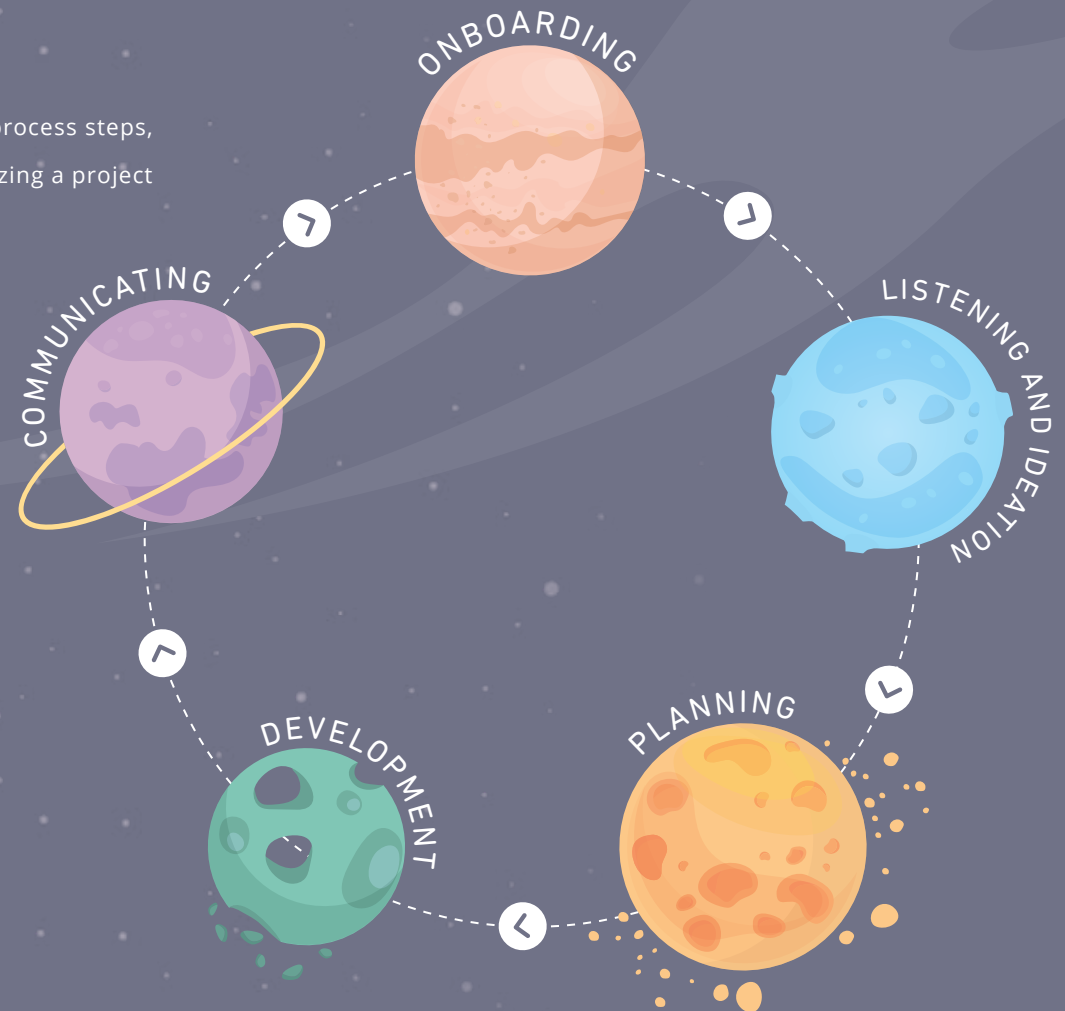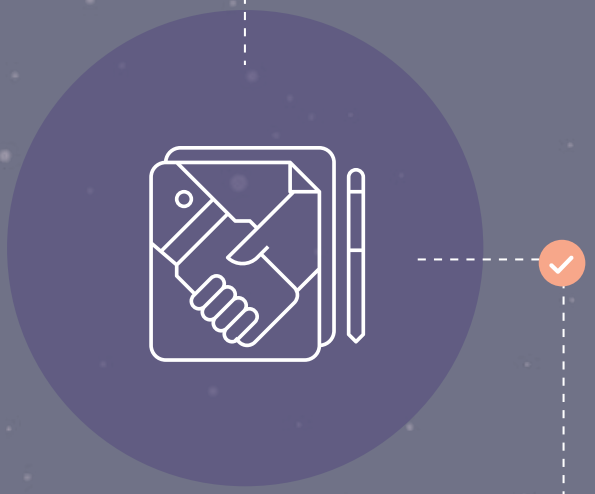
Onboarding

Onboarding

Whilst not technically part of a software development ongoing lifecycle a successful onboarding process will position you well to connect your Product Managers with your customers. This is a key step to ensure your product development roadmap takes into account what your customers need from the very beginning.

For vendors offering a Software-as-a-Service (SaaS) solution, it's widely agreed to maximize customer success and minimize churn, you need to get your customers engaged within the first 90 days of securing the new software. Often, the person that installs the software is the 'System Administrator', who may not be an end-user of the product.

We make sure we add all the users of our software to our initial onboarding campaign so we can communicate and support them directly. It is an important step to ensure all your customers and users are getting the maximum benefits of the updated software and they don't just receive the update via administration where key messaging could get lost along the way.
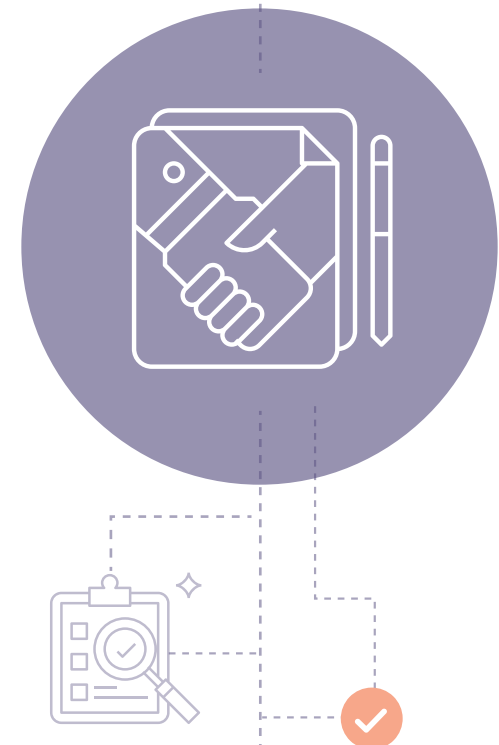
🔖 Onboarding

To enhance and maximize the user experience, during the first 35-40 days of the new software, they receive a series of emails covering various topics such as deeper product knowledge of key features, support channels, and various enhancement details. This is beneficial to the process by allowing you to communicate important information directly to end-users and opening up collaboration opportunities via community portals from the start of your relationship.

The more you can tailor this information and support the end-user with relevant insights around the improvements that impact them directly the better the experience will be for your customers. We also ensure maximum opportunity for people within the business to share their knowledge with colleagues and drive them to join our Community Forum.

The benefits of doing this are two-fold in the long run:

1. We now have access to communicate our major releases directly to our end-users rather than relying on the information being filtered through administration.

2. We also have them in the Community Forum where we can collaborate with them on patch upgrades, feature ideas and get them connected with the global user base.

Listening & Ideation

🔖 Listening & Ideation

You know you're building a successful product when your customers contribute ideas to your product roadmap. Customer-Centric Project Management is a key practice in ensuring that the end-user is always considered within the entire process of development. The customer becomes a key stakeholder in the project and by engaging them throughout the lifecycle it can lead to better outcomes.

Whilst a large portion of our roadmap comes from where we want to take the product, we constantly encourage our customers to provide feedback. Day to day users of the solution are an invaluable source for capturing new ideas and improvements. The added bonus is this strengthens relationships and engagement with our customers in the long term.

Within an Agile framework the process for collating feedback and customer ideas doesn't always have to happen within this stage of the lifecycle, the entire ethos of Agile is to be iterative and constantly improve which means input is required through every Sprint of the development process.
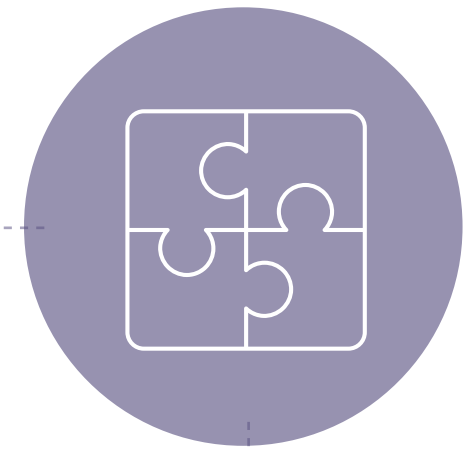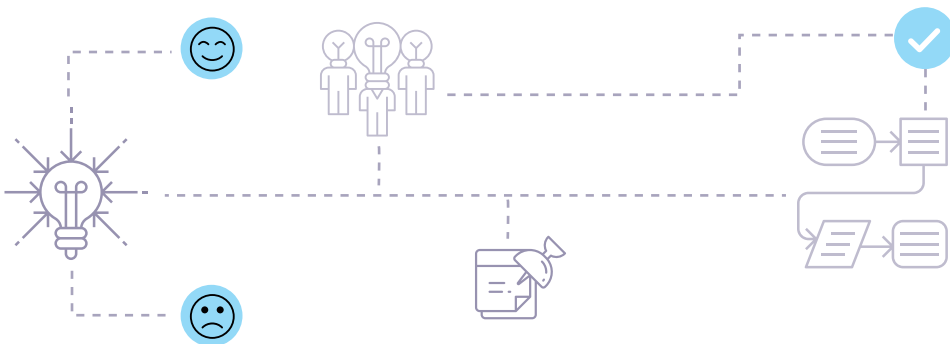
How to Capture Customer Feedback

Waterfall methods are more traditional where the customer ideas are collated at the beginning of the project but would not be considered throughout. This makes it more of a rigid approach to development but equally can be beneficial for certain types of software development, particularly where there is less opportunity to adapt during the process due to restricted development features.

Regardless of the way you gather customer sentiment and feedback this becomes a tool for more than just enhancing the features of your software it can also lead to customer retention. Customers that feel valued and included within the process are more likely to stay with the software, knowing they can continually influence its ongoing development. Your customers quickly realize that not only do they have a voice, it's being heard.
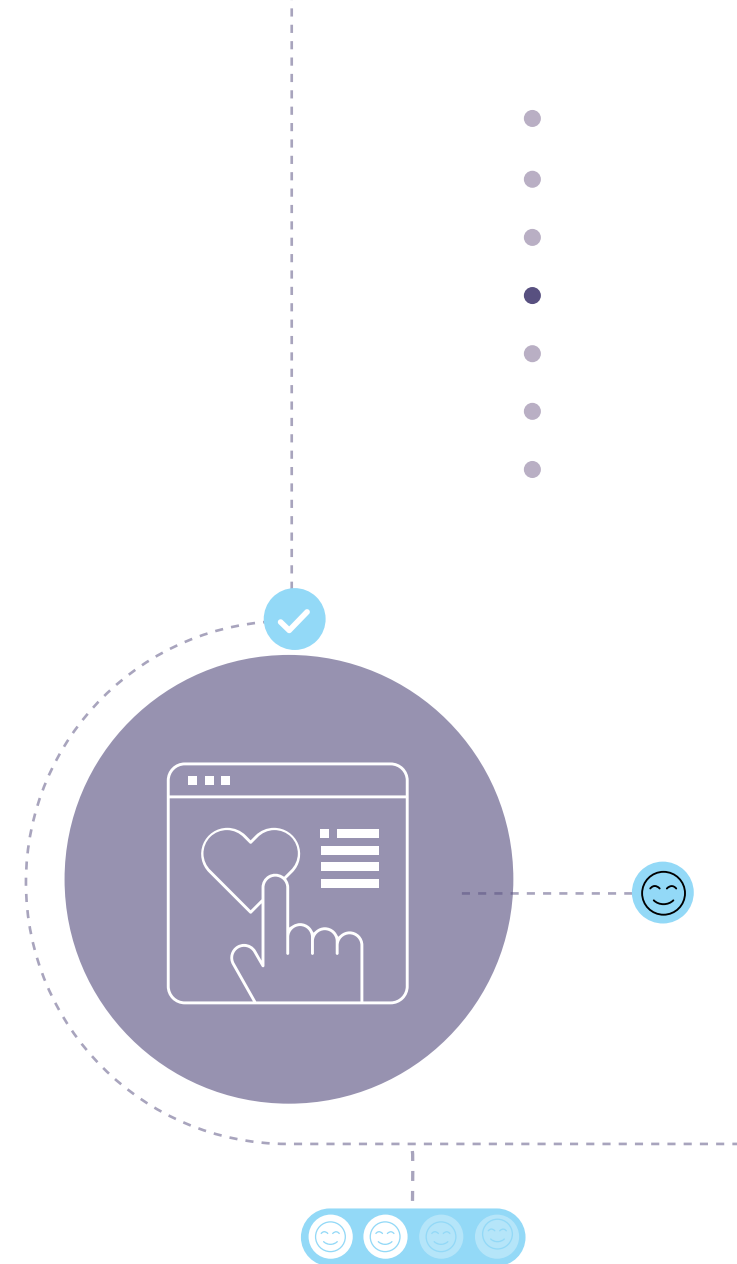
How to Capture Customer Feedback

Whilst there are many other channels to capture customer ideas, we've found the following to be invaluable:

Reviewing support cases for feature requests:

One of the key things we do is to go back and review all our support cases since the last iteration of the software and look at the challenges our customers have and also the ideas they put forward. By identifying the common challenges it can directly feed into enhancements for the product by aiming to solve the common issues our customers may be experiencing.

Holding regular customer success 'health checks'

We conduct these feedback sessions with our customers periodically, it is important that we check in on their utilization of the product and make sure they are maximizing their use and its benefit. Whilst conducting a health check we always look to respond to direct feedback and ideas that the customers may have as they take us through how they are using our software.

How to Capture Customer Feedback

## Net Promoter Score (NPS) survey feedback

Every 90 days we conduct a more formalised survey in order for our customers to have the opportunity to really provide considered feedback without the pressure of speaking to us directly. We are always so keen to hear what they think and then integrate this back into our product idea lifecycle.

Customer summits

We hold these virtual or sometimes face-to-face summits with our customers in order to strengthen relationships but also get some dedicated time with our customers to talk more freely about their businesses and needs. It's at these summits that we present ideas we have for the future but also gain valuable insight from them at the same time.
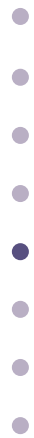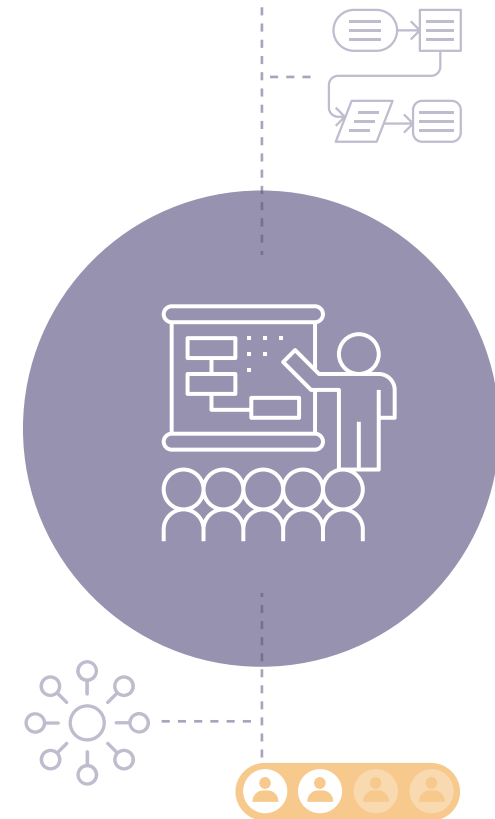
Planning

Planning

Having a structured, regular major release cycle allows your customers to become familiar with how you approach your product releases. We align with the Salesforce app release strategy of '3 major releases per year' which works well for us as there is enough time for the latest iteration to be fully embedded with our customers prior to the next feature update. Generally, we find with this timeline customers are really ready for the next release and are excited to see what is in store!

Other software development projects may not have the flexibility of regular releases, particularly if it is a complicated consumer facing application with lots of moving parts. Generally most software users would expect at the very minimum an annual update to feel secure that the software is being kept up to date. Apple will generally release its major iOS software update around September every year, although sometimes it has smaller critical updates throughout the period too. This is what customers rely on, that they can expect a fresh iteration of the software at regular intervals.

When it comes to planning for the update development it is vital you have a strong project management system that will support you through this. As you might expect, we use Mission Control, for all our product and project management, however the fundamentals of planning should apply to any solution that you've chosen.

We have a 'Program' for the overall roadmap for the product. Any new idea or feature request gets created as a 'Requirement' on our Program Planner. This is where we look at planning out all the ideas that have come from within the team and also our customer feedback. Within each Requirement, we build out detailed User Stories, and related to each User Story is a set of 'Checklist Items' containing the testing scenarios that the feature will be reviewed against.

🔖 Planning

For example, we might take a piece of feedback from a customer that says they want the option of changing the colours on the Gantt Chart to align more with their internal branding structure.

We then create a requirement 'Gantt chart updates' within the Program Planner.

From here we would build out User Stories, which are typically written in a manner that focuses on a particular persona, what they want to do, why they want to do it and what successful achievement of the feature will look like:

As a Project Manager

I want to Be able to clearly visualize my Gantt chart tasks based on the brand colours of the business

So that I can Easily see which areas of the business are across each Milestone and Actions

**Acceptance criteria** Ability to colour code the Gantt Chart

Planning

With each of these User Stories we detail out the various Checklist Items that will enable this feature to be brought to life and what exactly we need to do to ensure this happens.

The work we put into documenting the Requirements and User Stories pays off later, as it forms the basis of our development brief and testing checklists. In turn, it then feeds the release notes, so we gain efficiencies by not having to create the same content in several different places.

Each of these User Stories are also automatically built into our Story Board feature. This allows you to build up a journey / map of the evolution of a particular feature, as well as organizing the priority of the stories within a particular Requirement.

The Story Board is brilliant for people who like to see visually where each User Story is currently sitting within the process and how it relates to the entire development. This is particularly useful for developers utilizing Agile practices for their projects as it provides a clear process for development.

As the list of Requirements builds up, we start to assign them to individual 'Projects' within the Program. Each Project being a release of new features. This provides us with the visualization of what is going to be released, and when - ultimately providing us with a formal Product Roadmap.
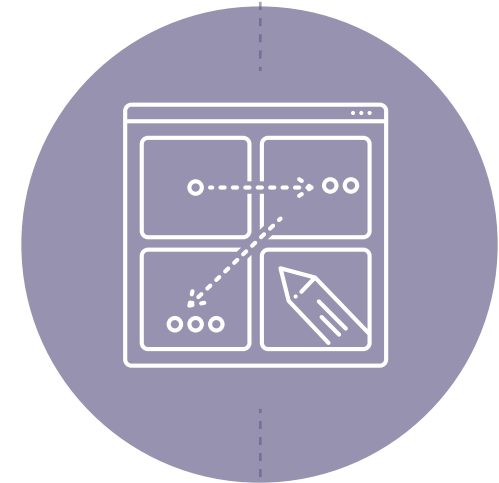
Developing

🔖 Developing

Individuals within the development team can also flag potential issues or roadblocks directly within the Action when it occurs highlighting to the team that potentially more support is needed to move this Action along. The Project Manager can keep track of these within the Issue Log and they can also be addressed during daily catch ups.

What are the key measures of success for project progress?

Transparency
Ensuring everyone in the team can visibly see what is occurring across all the Actions. Within smaller Scrum teams this keeps everyone on track and gives the opportunity for resourcing changes easily.

Flexibility
Being able to be nimble with Action progress and also dealing with issues on the go so that they don't become larger roadblocks to progression.

Communication
Enabling teams to stay in close contact either via daily catch ups but also from the Chatter feed that means notes against Actions are visible to everyone in real time.

To ensure efficiencies, time spent on Actions is also tracked in real time by individuals enabling us to evaluate how we're tracking versus our original estimates. Project Managers can keep a close watch on over-servicing of time allocated and implement further efficiencies on the go.

Testing Phase

Once development has occurred the next critical element is testing and this is where you can go back to your User Stories to set testing scenarios that will determine that the User Story has been developed successfully against the acceptance criteria. Providing that safeguard for all the different elements that need to be met for that overall requirement.

Plus, for each Checklist Item, you can record whether the test was successful or not. If not, Issues can be raised so the development team can review & resolve quickly. It's a process driven pathway to ensure every element of the detail within the development is taken into account and dealt with effectively.

At Mission Control we enable a solid process for every test we are going to do to ensure that we capture any potential Issue once the test has taken place, which then feeds back into our development cycle so that nothing is missed. If a test scenario fails, the Issue gets added to the Issues Log and the development team is alerted to work and resolve the problem. Once completed it is entered back into the testing process again until it is successful.

You now enter the phase where you can close the loop around the development process by ensuring you have incorporated all the various testing scenarios for each Requirement and User Story.
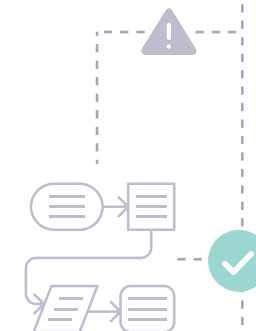
Developing

As we focus in on a particular Project, we start to identify which Sprint each of the Requirements will be delivered within, and allocate resources and effort to each Requirement. This process allows us to prioritize the requirements based on the ease of development, for example enhancements to an existing feature are generally more straight forward than an entire new feature build.

We do all of this within Mission Control, using the Project Builder that allows us to view the list of Requirements and convert them into Actions to build up a plan that can be scheduled and resourced. Aligning resourcing based on this stage is key to success and for those implementing Agile frameworks through Scrum we would encourage teams to work across the various Requirements shifting resource depending on capacity and workflow. The aim with Agile is to ensure the entire team reaches the end of the Sprint together and by utilizing all available resources it works well.

Throughout development, keeping track of progress is also important. With Agile frameworks the use of Kanban is popular so that Actions can be visualized easily at daily catch ups and moved along the progress levels accordingly. This allows everyone in the team to be across potential roadblocks and Actions that are not progressing as quickly and resourcing can be shifted accordingly.
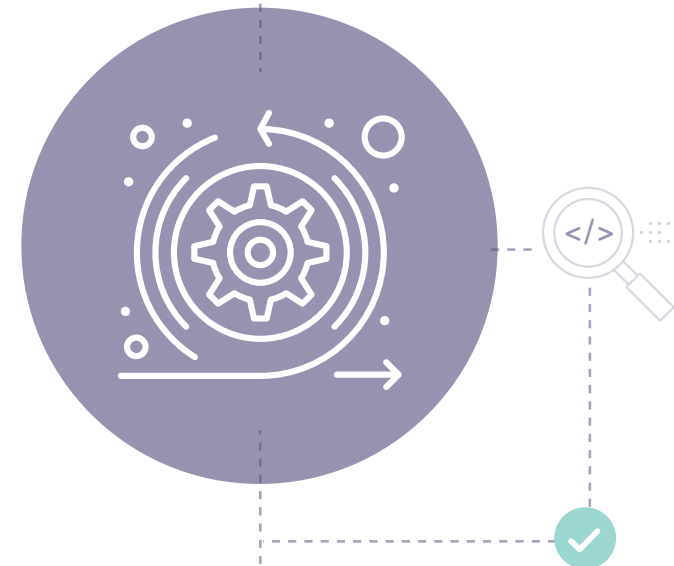
Communicating

🔖 Communicating

As much as we'd all love to think our customers are as excited as we are to deliver and utilize our new features, most still need support in fully understanding how to leverage the new functionality we're delivering.

To ensure our customers get as much information as possible, in the format that they consume the best, we put together various resources for each major release, including:
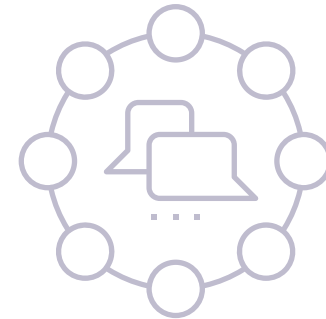
Release Notes

This is our most detailed content around the features, explaining all the new enhancements and how to make the most of them. We encourage all our users to download and refer to them regularly as most of the questions we get asked about the new features are answered in this document.

'What's New' Webinars

We recognize the importance of providing visual mediums for our customer to see the new features in action. These are extremely popular and many customers rely on the webinar to take them through all the latest updates.

Communicating

User Guides

This resource forms part of our support material for any new customers to the product. We want to make sure these guides are updated regularly taking into account the updated features and enhancements.

Knowledge Base Articles

This is an entire section of our website where we have broken down the User Guides into 'bite sized' chunks which allows for easier search-ability when users are looking for some specific information or support.
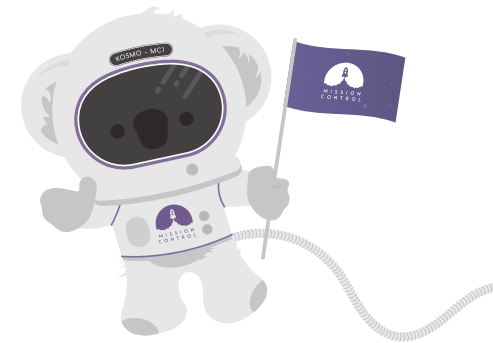
Blog Articles

It's important to showcase the features in a more meaningful way which is why we also provide a range of thought leadership blog posts that go into the usability of the features in various scenarios. It's important that customers can relate to the new updates with regards to their own particular industry and job function, which is why this content is so important.

🔖 Communicating

We then communicate all of this to our customer base, and thanks to the robust onboarding process, this includes the day-to-day end users, not just the System Administrators.

Communication and feedback is fundamentally important with any product management process, whether it's internally throughout the development or externally to engage customers. We find that this is the core reason we have been able to create such a robust process around our software lifecycles.