

ACTION PLAN: SOFTWARE ROLLOUT TO A SINGLE TEAM

For a successful rollout, you need to set tasks and assign clear roles, responsibilities and deadlines for each task.

Filling in this action plan will help to make sure you've considered what you need so you're good to go on launch day, when introducing new software to a specialised team in your business.

How to plan – align your rollout plans with your goals for the software

Set a launch date

Consider the date you want, or need, the software to go live.

When you're clear on your launch date(s), plan backwards from there.

Allow enough time for tasks such as getting staff trained, taking into account the number of staff and the level of training needed, and having a test run of the software to check for issues or bugs.

Action: write down here how much time you expect each task to take.

You might find it helpful to run through the rest of the action plan and fill in this section last.

How to plan – align your rollout plans with your goals for the software *(continued)*

Use the skills you've got in-house

Make the most of the expertise and skills within your business to help with planning.

Do you have a tech-savvy team member who can provide technical support?

You'll need good communication; is there a team or person in the business who can help with this?

Can someone in HR help you schedule training and identify if anyone needs extra support to complete it?

Your rollout may need several people - one within each team - to help oversee the rollout and ensure successful adoption.

Action: make a list of who can help with these tasks.

As above, this is something you might want to come back to later when you've completed the rest of the action plan.

Consider what technical support you'll need

Do you have someone in-house or are you getting external help?

This support could be from your software supplier or an independent consultant. Find out when they'll be able to provide specific services, such as training.

Factor these timings into your planning. Communicate your timings for the rollout and make sure everyone knows what's expected of them, and their deadlines.

Action: write down where your support will come from, and delivery timings that relate to your launch date.

Maintenance and bug fixing

Who will address any issues with the software on launch day and afterwards? Will a tech-savvy member of your team take on this responsibility? Or will you use your IT team, the software provider, or other external support?

Action: consider who's the best person/people to be responsible for which tasks.

Consider how software will be accessed

Think about the people who will be using the software regularly, but also make sure you identify anyone who may use it on an occasional basis.

How will they access the software? Do they have the hardware they need to use the software? If not, how long will it take to get this in place?

Action: make a list of the people who'll need access, hardware and whether you need to order extra kit.

How to plan – align your rollout plans with your goals for the software *(continued)*

How will your new software affect other teams?

Think about how adopting new software for one team might affect another.

If you're adopting new Digital Accounting software, for example, will invoices continue to be passed to the person who pays them as it usually does? Will staff who won't be using the software be affected indirectly? For example, they may be required to submit their expenses in a different way.

Action: list how the rollout may affect each team and their work, including any indirect impacts.

Be aware of the potential risks

Listing the areas of risk will help ensure your back-up plans are robust. For example, if your internet connection goes down on launch day, could you use another network at short notice?

What is your contingency plan to ensure business as usual across different teams?

If you've introduced new technology before, consider the lessons learned.

Action: write down the issues that may crop up and how they could be addressed.

What's the worst-case scenario plan?

If, on go-live day, the software doesn't work as it should, have your old system on standby, just in case.

Once live, if the system goes down, how will your team report it? Who will they report it to? Will you arrange for automatic updates to be sent out when there are issues with the new system?

Action: if this is necessary, write down here who will make this decision and when.

Communication is key – involve your team throughout and listen to feedback

Draw up a timeline

Create a timeline, set milestones and check-in points for each stage of the rollout.

How and when will training take place? Will there be a test run of the software ahead of the launch? If so, when will this be? For example, if you're introducing eCommerce software, send a dummy email to a team in your business to make sure the technology works as it should before you email your customers from launch day.

What do you need to tell your team, or teams, and when do you need to tell them?

For example, will you introduce a rota to make sure any incoming queries from other departments are dealt with efficiently?

Consider if you need to tell your customers about the change too, and when this should happen.

Action: list each milestone that you'll communicate to the business here.

Communicate rollout plans

How will you communicate the rollout?

For example, will a kick-off meeting with someone overseeing the team who'll use the new software then be followed up with a team-wide meeting?

Will you this be followed by regular diary dates for everyone in the business to meet and share ideas and feedback?

Action: list everyone you need to communicate with, when that needs to happen and how you'll do it.

Gather feedback

How will you gather feedback? Will it be on a continuous basis throughout the rollout via emails, for example? Or at specific points during the process, in meetings?

Get your representatives to gather feedback from colleagues in their teams; this could be concerns, issues or suggestions.

For example, who will answer technical questions, or pass on suggestions for improvements? Who will address concerns within the team?

As we highlighted in our guide to [making sure software works for your team](#), getting everyone on board comes from having regular conversations and not just giving instructions.

Action: write down how you will gather feedback and how often.

Communication is key – involve your team throughout and listen to feedback *(continued)*

Make the whole business aware of the change and how it might affect them

Although your software is only being rolled out across one specialised team, other departments might still be affected.

For example, if you're adopting new Digital Accounting software, will staff across your business be required to submit their expenses in a different way?

Also, make sure the wider business is aware that some standard services offered by the team adopting the software may be affected in the short-term during rollout.

Write down your communication plans, making sure you've included everyone in the business.

Action: identify who the change will affect and how, and when you need to communicate this throughout the business.

Keeping your customers informed

Consider customer experience and whether anything will be impacted due to your rollout.

Could the ability to place orders and pay, speak to the team, or access your website be affected at any time? If so, write down what might be affected and how you might communicate this to your customers.

Action: consider how customers might be affected, and how you might communicate this to them. For example, via an email.

Make training a priority – it's vital to smooth integration and successful adoption

Decide who will carry out training

Decide who will organise and manage training. Will it be delivered by someone in-house, with one person nominated as a training champion? Would this be easy to extend across multiple teams? Will you need external support?

Make the most of all the information your software supplier provides with the product; many provide training videos and tutorials on how to use the technology.

Action: outline your plans here and list the training resources you're considering.

Make training a priority – it's vital to smooth integration and successful adoption *(continued)*

Will you need to offer different levels of training to different people?

Think about who will use the software, and how. This will help shape your training plan.

Will you offer different levels of training depending on individual team members' skillsets, and how they will need to use the new software? Allow extra time to provide additional training for some members of the team if it's required.

Will basic training be offered to other teams your software rollout might affect?

Action: if you intend to offer different levels, write down here what these will be.

Will training be part of inductions?

Have you got new staff starting soon after launch date that will use the software? If so, write this down here. How and when will they be trained on using it?

You might want to take a long-term approach. For example, do you want to make training part of inductions for specific new joiners to the company? If yes, write it down here to action at a later date.

Action: decide how you can support new staff to adopt the software.

Enroll 'software champions'

Natural software champions are those who embrace the challenge of getting to grips with new technology.

Involve them throughout, including testing the software, helping to train colleagues and identifying a small group who can pilot the software ahead of it going live.

Action: Choose your software champions.

Keep testing the software ahead of launch

Incorporate all feedback

Use feedback from your ambassadors and software champions to identify what bugs and issues need to be fixed as the rollout progresses.

Consider setting up a dedicated document that logs each fix that's needed, how this will be fixed and who will complete the work.

Action: decide how to log all feedback and action relevant suggestions to improve how the software is implemented.

Keep testing the software ahead of launch *(continued)*

Decide what fixes are pre and post-launch jobs

As part of testing, decide which issues have to be fixed before launch.

You might find bugs that can be fixed after launch, but make sure you know how much they'll impact your business operations.

Action: write down what fixes are needed, plus a deadline for each one.

Next steps, post launch – consider what success looks like

Gather feedback

Continue to gather feedback to assess how the software is bedding in.

Action: write down how and when you will gather feedback. For example, two weeks after launch, then monthly catchups.

Does the software achieve the goals you set?

Look back at the goals you set yourself at the start of this action plan to help measure success.

[Monitoring how the new software is being used](#) will help you establish whether you're getting value for money.

You can check if these goals are being met by talking to the new users, and looking at the impact your new software is having on business operations.

Action: list which goals have been met and where improvements could be made.

Once you've filled in this action plan, you've laid the groundwork for a well-thought-out rollout plan.

Find out more about [reviewing software post-launch](#).