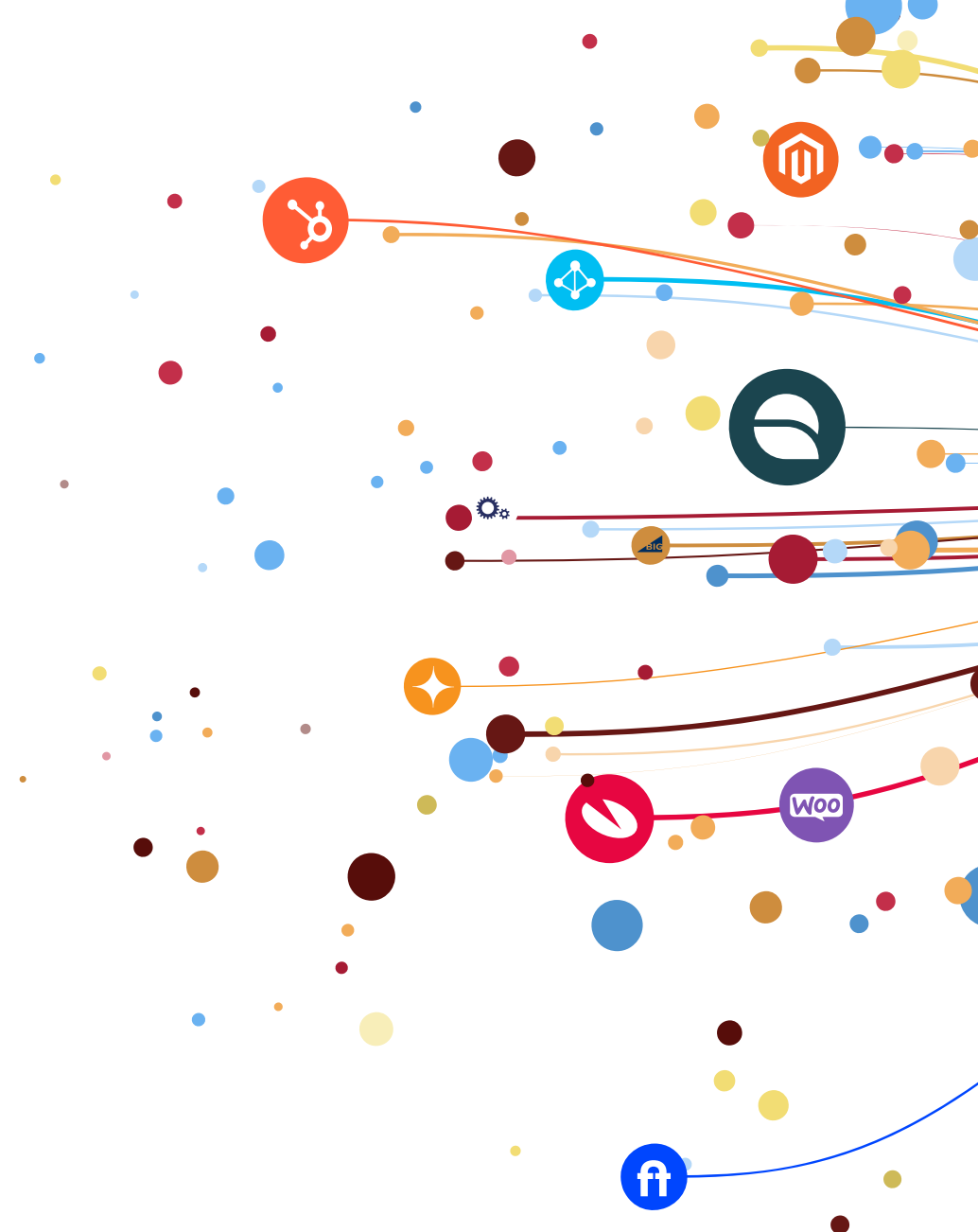


The ConnectMyApps guide to

BEST PRACTICE FOR A SUCCESSFUL INTEGRATION PROJECT

CONTENTS

Foreword.....	3
Getting started internally	4
Preparing the integration specification	6
Selecting an integration partner.....	8
Development.....	10
User Acceptance Testing (UAT)	11
Deployment / Go-live.....	13
Checklist.....	14
More information	15



FOREWORD

Since 2014 we have delivered many thousands of integrations for over a thousand different clients. While no two integrations are the same, our experience has shown that if not done right, even a modest integration project can become complex and cause frustration for all parties - most importantly, the client.

Fortunately, there are some simple steps common to most integration projects that will help make the process as smooth as possible. This guide gives practical advice for making an integration project successful for people who do not necessarily have deep project management or technical experience.

If you would like to discuss your integration requirements with our team of experts, please contact us. In the meantime, we hope you find our **Best Practice for a Successful Integration Project** guide useful.

Nathan Bray
Co-founder & CMO
ConnectMyApps



GETTING STARTED INTERNALLY

Think business process first, technology second. Because in many ways, integrations that transfer data from one part of your business to another, are the backbone of your business process.

You probably have specific pains you want to solve, such as reducing time on data entry, more reliable records, more secure data transfer, or more timely information. However, focus on solving the business problem, and an experienced integration partner will help you find the best technical solution to achieve that.



Define the business case internally

Getting the mandate and budget for an integration project requires a business case, for example reducing the time spent on manual data entry and correcting incomplete or inaccurate records; making a manual process more resilient if a key employee is not available; having the right data in the right place at the right time; or better data security. How will this integration improve your operations, and the monetary value.



Set realistic timeframes and milestones

This is especially important if you are implementing other systems in parallel, such as a new ERP or HR system. Your team will be required to work on multiple fronts, and it is easy to become overloaded. Allow yourself and your vendor sufficient time to properly scope, develop, test, and implement the integration. Many integrations can be done in stages, so agree milestones with the partner in advance.



Set a realistic budget

Moderately sized integrations typically do require consulting, development, and testing. Your integration partner should be able to give you a ballpark estimate of the hours needed if they have expertise in the field, and then a fixed price once the scope is agreed.



Assign one project owner internally

If an integration involves multiple departments in your organization, there can be many competing voices and opinions, which sometimes slow the process to the point of inertia. Assign a project owner with an overview of all the different requirements and the authority to make a final decision to help keep up momentum on your project.



Give yourself time to use a new application in practice

Tricky one as often the SW vendor will want us to make integrations from the start, and the client insists that having a perfect process from day one is essential. But we know from experience that often it takes time to understand how users will use an application, and things change. Therefore, if possible, allow yourself some time bed down an app, and only do the absolutely critical integrations from day one. We can always build new integrations later.

PREPARING THE INTEGRATION SPECIFICATION

This is the most important step for you. The specification, or scope, is like the blueprint of a house, and will largely determine if the “house” is stable or needs frequent repairs.

A partner with deep integration expertise can help by asking the right questions and identifying issues they foresee in the business logic or technology in advance. But ultimately, if you have agreed a scope of work and the integrator delivers to that, it is your responsibility, so a thorough specification is a good investment of time.



Involve management from all relevant functions

If your integration is crossing departments, for example connecting a CRM app used by the sales team to an ERP app used by the finance team, the leaders in the teams should be involved at the early stages to define what the integration should do at a high level.



Include people working daily with the systems

Once the high-level objectives are agreed, involve the people who work daily with the applications. They are often the subject matter experts and have valuable input into the daily work processes in their roles, and what will make their days more productive.



Be detailed in your specification

This is key. An experienced integration partner will not expect you to know the exact API endpoints needed, for example, as they will usually talk through your requirements at an application level. However, specifying the fields in an application to transfer is a great start. It does not need to be 100% perfect on the first draft, as this will be discussed in the scoping process with your integration partner, but the less ambiguity the better. If you are not sure where to start, we have an easy field mapping template to help.



Do not use integration to perform business logic

It can be tempting to try to “add” missing functionality in an app into an integration. Our advice is to keep the business logic in the specialist applications themselves, which is what they excel at. In some cases, a small amount of logic may be needed, but the less the better.



Align your ambitions with your capacity

We are often asked to build up to 10 different integrations across multiple applications simultaneously. This is technically possible for us, but our advice is to approach it in stages and prioritize the key integrations first. This is especially so if the integration is part of implementing a new system, which itself consumes a lot of internal resources.

SELECTING AN INTEGRATION PARTNER

Integration is a partnership, and you will almost certainly be involved throughout the process. Your organization may have its own vendor selection policy, including data security, financial stability and similar. This section only covers criteria related to a vendors' ability to deliver integrations robustly and at good value for money.

If the integration is part of a larger software investment, your software vendor or implementation partner may have an integration partner they recommend. If not, below are some criteria to consider when evaluating a potential integration partner.



They understand the business process you want to enable

An understanding of the business process allows your integration partner to propose solutions and anticipate issues you may not have considered. Almost any developer can code a basic integration, but an integration partner that views this as a business problem to be solved with a technical solution will do a better job than one who sees it purely as a technical solution for a technical problem.



They have experience from similar integration projects

While they do not necessarily need to have been the exact same apps or workflows, an understanding of the application domain and business processes involved will give them insights and know the most important considerations for this type of integration. Similarly, an integration partner that has worked on similar size projects as yours is important, whether that is very large, very small or somewhere in between.



They provide ongoing support and maintenance

This is key. Developing an integration is only part of the job. Supporting and maintaining an integration for years is the difficult part. Requirements often change over time; APIs fail or get updated; and bugs can appear. Even small changes can become expensive, depending on your choice of integration partner. Ensure the vendor is clear on support and maintenance costs, and that they provide a documented Service Level Agreement.



They do not deliver “black box” integrations

Traditionally many integrations were “black boxes” that did not give the client visibility into how it worked. This can cause problems if you want to switch applications at some point, or the developer responsible for making the integration leaves their job, whereby it can be very difficult to understand how to update or maintain the integration.



They provide a clear Statement of Work (SoW)

This is key. The SoW is the base from which your integration is built, including which data will be mapped to which fields; the frequency the integration will run; and include information

required for GDPR compliance, such as the data being processed. A vague SoW from your integration partner will likely lead to problems later.



They do not use taxi-meter pricing

Traditional integration providers usually have a consultancy business model and are incentivized to bill as many hours and prolong projects as much as possible. However, once the Statement of Work is signed, there should not be additional costs unless you have out-of-scope requests.



Build robust rather than intricate

The most important quality in an integration is reliability, ensuring the right data gets to the right place at the right time. Building intricate workflows that try to do many tasks in one can make maintenance more difficult and means that a small issue in one part of the workflow can impact other parts. We recommend building workflows to perform one specific task each for simplicity and supportability.



Give your vendor guidance on your acceptance criteria

Set the expectations for what you will test before going live with the integration and how long that process will take as early as possible to avoid misunderstandings at the end of the project.

DEVELOPMENT

If you have completed the first three stages, the development stage is mostly out of your hands and with your integration partner. However, there are some steps you can take to help your partner ensure the development goes smoothly as possible.



Assign a dedicated internal technical resource

Your integration partner may have technical questions while development is underway. We have experienced that a relatively straightforward question can go unanswered for a week or more, sometimes blocking the rest of the development. Having a technical contact is helpful to provide answers promptly and avoid unnecessary delays.



Invest in a small Proof of Concept (PoC)

Not all APIs and applications are equally mature. In some cases, a small PoC can give you confidence that your requirements can be met, without investing in a full-scale integration project that ultimately does not meet your needs. This will typically be agreed during the specification phase.



Agree project updates if required

If an integration project is complex or expected to take more than a couple of weeks, scheduled project management meetings with your integration partner can help keep the project running smoothly and give you better control.



Allow for iterations

We believe in prototyping and developing in iterations where possible, ensuring we are on the right track before going too far, and giving the client the opportunity to make minor adjustments in flight. If your integration partner has a similar approach, ensure you have resources available to conduct “smoke tests” at each iteration.

USER ACCEPTANCE TESTING (UAT)

Despite being close to the finish line, UAT is a step that clients often overlook during planning and can delay the go-live of an integration that was, until then, progressing on schedule.

Yet a thorough UAT is a vital step that can save you time, money, and frustration. It does not necessarily require a lot of resources if well planned, but helps avoid going live with an integration, only to realize that changes are needed after it is in production, which can be costly, not to mention inconvenient.



Define your UAT plan

This can be as simple as making a list of business processes that should be tested in which order; guidelines on which data will be used for testing; the expected results; and who is responsible for conducting and signing off on UAT. Most of these questions would ideally have been considered in the specification phase.



Ensure testing resources are available

Depending on the integration and the sophistication of your internal procedures, you may do different layers of acceptance testing, from the functional to the technical. Your integration partner should give you advance notice of when they expect the integration to be ready for UAT, so you can schedule the necessary testing resources. A client not having scheduled test resources is one of the most common bottlenecks we face, and can delay their go-live date.



Prepare test cases in advance

If your integration partner has provided a clear Statement of Work and you have defined the acceptance criteria in advance, it should be straightforward to test the integrations against those requirements. Depending on how business critical and complex the integration is, it can be sensible to also prepare edge cases to test against. If you are unsure how to approach this, ask your integration partner for advice.



Clear feedback procedures for system users and testers

There may be several different users involved with testing the integration at a technical and application user level. Therefore, create a clear procedure that consolidates all feedback so the project owner can review, and where necessary, escalate changes to the integration partner in a structured manner.



DEPLOYMENT / GO-LIVE

Congratulations, your integration is ready to go live and you've taken a step to a more efficient business! There are just a couple more steps to ensure the project is successfully completed.



Monitor the first live runs closely

Despite all the testing, keep a close eye on the data being transferred over the first few production runs. Ensure the data is transferring to the correct fields in the application and reconcile the number of records transferred. For example, if your Ecommerce shop shows 100 orders that day, check that 100 orders have been transferred to your ERP and 100 new contacts have been created in your CRM.



Clear feedback procedures for system users

As with the UAT step described previously, create a clear procedure that consolidates all feedback from system users so the project owner can review, and where necessary, escalate changes to the integration partner in a structured manner.



CHECKLIST

PHASE	TASK	STATUS
GETTING STARTED	Integration business case defined	
	Budget availability confirmed	
	Approximate timeframe and milestones set	
	Internal project owner assigned	
SPECIFICATION	High level internal agreement on how integration will work	
	Key application users consulted on requirements	
	First draft technical specification prepared	
	Keep business logic in the applications, not integrations	
	Reality check: internal resources aligned with ambitions	
VENDOR SELECTION	Understands the business process	
	Has experience from similar integration projects	
	Provide ongoing support and maintenance with clear pricing	
	Provide visibility into our integrations, not a black box solution	
	Acceptable TCO including support and maintenance with SLA	
	Provides a clear and detailed SoW	
	UAT criteria communicated to vendor	
DEVELOPMENT	Evaluate benefits of a small Proof of Concept	
	Technical resource assigned and available if needed	
	Resources available for iteration testing if needed	
	Agree schedule for project updates from vendor if needed	
UAT	Define UAT plan	
	Testing resources scheduled and available	
	Test cases ready in advance	
	Feedback routines from system testers to project owner agreed	
DEPLOYMENT / GO-LIVE	Procedure for monitoring initial production runs in place	
	Feedback routines from system users to project owner agreed	
	Go-live celebration planned!	

MORE INFORMATION

ConnectMyApps Canvas is a low-code integration platform that makes building and maintaining integrations simple and efficient using a drag and drop visual interface, all at market-leading prices.

Either build integrations yourself, use our integration experts, or ask your trusted IT partner. Please contact us to find out more.

info@connectmyapps.com

About ConnectMyApps

ConnectMyApps was founded in 2014 with the vision to simplify integration. Today, we have over a 1,000 clients, helping them be more efficient by connecting the applications they rely on such as ERP, CRM, HR, Ecommerce, POS and more. Headquartered in Oslo and with offices in Copenhagen, we work with leading clients across Europe including McDonalds, Maersk Drilling, Vinmonopolet, Odeon, DHL, Stepstone and Orange. ConnectMyApps is part of the Amesto Group.