Implementation Methodology

The ultimate guide
to successfully implement
Odoo projects.

odoo

Implementation Methodology

The ultimate guide to successfully implement Odoo projects.

Table of Contents

Introduction 5

1. Key Concepts 6

2. What is a Successful Project? 10

3. Roles 16
Odoo: Project Leader 17
Odoo: Project Director 18
Odoo: App Expert 20
Odoo: Developer 20
Customer: Single Point of Contact (SPoC) 20
Customer: Extra Roles 22

4. Implementation Phases 24
ROI Analysis 26
Project Kick-off 29
Implementation 34
Go-Live 38
Second Deployment 39

5. Implementation Challenges 40
How to encourage users to embrace Odoo 41
How to deal with resistant people 42
How to keep things simple 44
How to manage customer expectations 44
How to write a good specification 46
Avoid importing data history 46
Avoid writing documentation for your customer 48
How to deal with customers' specific demands 50
Why should you minimize custom development? 56
How to deal with internal politics 59

How to deal with different people's dynamics 61
Why should young Project Leaders feel confident? 62

6. Quiz 64

7. Measure your progress 72
8. Extra References 74

Introduction

As Project Leaders, our job is amazing: **we have the opportunity to improve people's lives at work** by automating boring tasks and making companies more productive. Rare are the solutions that have such an impact on the people using it.

But implementing a management software is **as difficult as it is impactful.** Odoo connects all departments, which means a lot of changes and a lot of users relying on you to improve their workflows.

It's hard to be a great Project Leader... very hard. More than 50% of proprietary ERP implementations fail and only 18% of SME's have deployed an integrated management software, because it's too complex and too expensive for them. But the constant failures to deliver are actually our opportunity to thrive.

By making implementation projects smooth, predictable and affordable, Odoo is transforming the market and in doing so fulfills a huge demand. Over the past 5 years, more than 95% of our implementations have been successful, which stands in stark contrast to other solution providers.

To reach to that point, we had a critical look at our approach and at the role of our 80 Project Leaders. We fine-tuned our methodology, analyzed how the top performers behave and realized what makes some projects more successful than others. **This guide summarizes our best practices** and all the tricks we've learned.

01

Key Concepts

Responsibilities

- As Project Leaders, our top priority is to ensure the project stays on schedule, and on budget.
- The customer defines their business' needs (*why* and *what*) and the way to satisfy these is defined by the product through us (*how*).
- Odoo challenges customer demand to ensure the benefit is worth the cost.

Keeping It Simple

- Fewer meetings, less paperwork, faster decisions.
- Limit the number of stakeholders to accelerate decision cycle.
- Limit custom development to the minimum necessary.
- Working on site is inefficient to get things done, but efficient for change management and training. Go on site only when needed.

People

- Project Leaders must be problem solvers, as well as product & business experts.
- Avoid intermediaries who are not decision makers.
- Train the key-users early on in the project. They should be confident in the product and committed to the project.

Project Managers

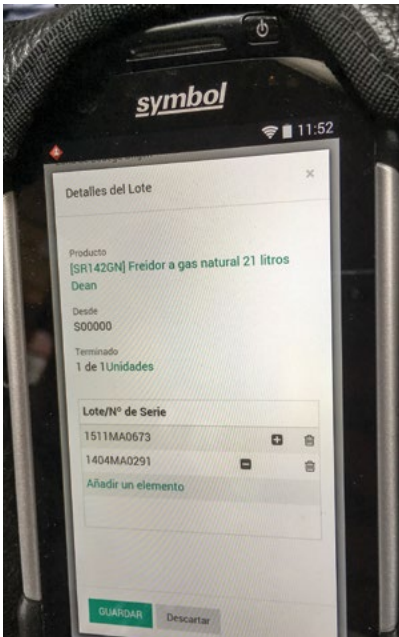
- The key success factor of any implementation is the Project Manager (aka Project Leader at Odoo[1]).
- Recruit & train the best talent. Retain only the top performers.
- Even the best Project Leaders miss critical details. To limit risk, Odoo Experts should challenge their work at critical steps in the project.

But never forget:

Common sense always prevails over any rule!

1 See chapter 3. Role definition

"Go-Live" at Industrial Taylor: Michaël, Project Leader with warehouse operators.



02

What is a Successful Project?

As a Project Leader, it's difficult to find the right balance between satisfying the customer, accepting more change requests, keeping the budget low, sticking strictly to agreements, spending more or less time in analysis, aligning with the project schedule, etc.

The key priority for a successful project is to **onboard users** in Odoo, and to do this **on time** and **within budget**. When a project fails, it's always because it took too much time or cost too much.

Timing and budget are the key elements to structure your methodology. The rest is secondary:

- developing custom features is not a priority.
- customer satisfaction during the implementation is not a priority.
- early service upsell is not important.

Developing specific features does not help the project

Custom development always creates additional costs and drags out the implementation project, sometimes to the point of putting the project at risk[2]. In addition, custom development leads to technical debt that the customer will have to pay for within the coming years in the form of additional maintenance and upgrade costs.

Each customization may seem simple and affordable. **But the complexity of a project grows exponentially as the number of**

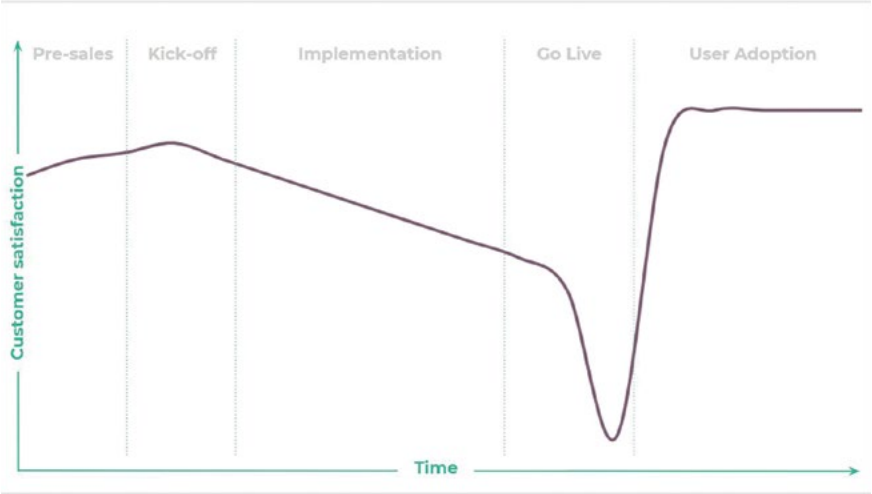
2 See article on https://news.ycombinator.com/item?id=17541092

customizations increase, not linearly

A project is successful if it's delivered on time and on budget. Developing custom features for the customer's specific needs doesn't make the project successful, but it is sometimes necessary in order to support the customer's core business.

Customer satisfaction is not a useful KPI

The customers satisfaction is not a good measure of the success of a project. First, it evolves constantly during the different phases of a project. Second, each person working for the customer might have different expectations, for example: a key-user wants additional features, but the CEO wants the project on time and on budget.



Customer satisfaction evolution throughout the project.

Focusing on customer satisfaction distracts the Project Leader from the project's main objectives. **We definitely prefer our customer to be temporarily dissatisfied** (because they had a harsh discussion about a complex decision) **than missing the implementation deadline.** Dissatisfaction is inherent in any project.

Even though customer satisfaction is not a goal during the implementation, it's still a good way to evaluate the motivation of key-users.

Therefore, we use periodical customer ratings to know which customers require more attention (and not to assess the quality of a Project Leader).

Selling extra services before "Go-Live" is not a priority

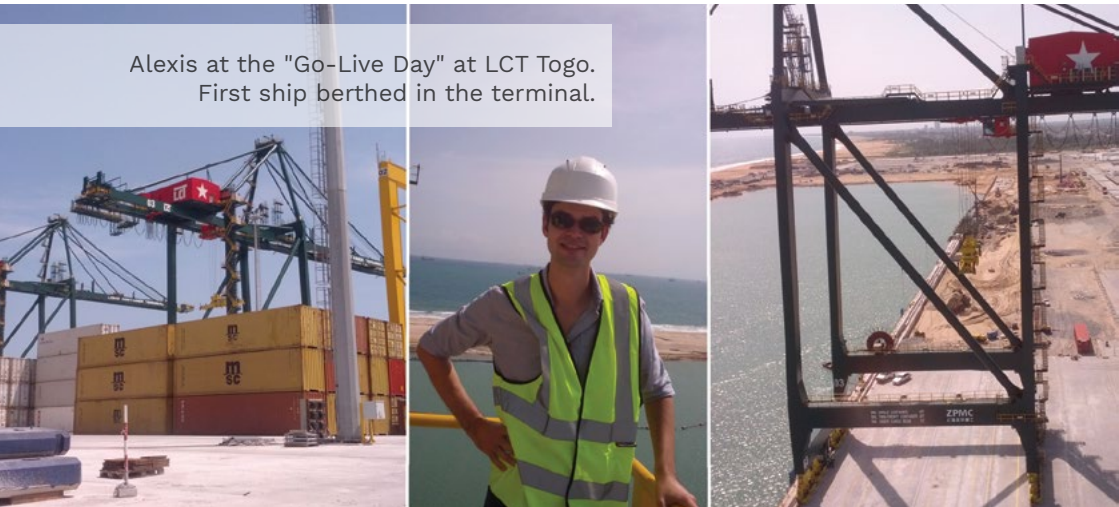
Service companies want to bill the customer as much as possible. It's their core business after all! Large service companies even have complex methodologies that lead to needing more services, like large analysis phases in the name of limiting the risk on a project.

We believe that selling more should never be an initial objective. Our company's growth should be the result of a quality service or of happy customers (ideally both). We actually think it's better to deploy customers in fewer days of work. Not only does it reduce the risk of project failure, but it also makes us more competitive on the market.

Having a good pace throughout the project is a huge competitive advantage to acquire new customers. And, as you build your customer base, it becomes very **easy to sell extra services to existing customers:**

- It's 7x easier to upsell existing users rather than acquiring a new customer.
- You can always divide the project into phases and sell non-mandatory features after the "Go-Live". This way, you will never need to squeeze your margins because the budget is consumed.

Long story short; **to achieve sustainable growth, focus on the project's success.** If you have successful projects, customers will buy more services later on. Every time you upsell before the "Go-Live", you weaken customer's trust; they might think twice before buying extra services in the future.



Alexis at the "Go-Live Day" at LCT Togo.
First ship berthed in the terminal.

A few years ago, I interviewed 15 customers to gather feedback about our implementation methodology and services. A customer told me this: “During the first 3 months, I didn’t like working with Frédéric [the Project Leader]. He was constantly questioning everything I asked for, to the point that I had the feeling I was wasting my time. It was a bit frustrating.

But, later on, I understood it was for the good of the project. He often found better solutions than what I asked for during the implementation. Now, even though we are in production, everytime I have a business process decision to make, I call him first to get his advice.”

This story perfectly illustrates our approach: by prioritizing the success of the project over the short term customer satisfaction, we actually make customers happy in the long term. Frédéric could have agreed to develop every custom feature the customer had asked for to make him happy at first, but the project would have cost more, been delayed, and we would have been at the risk of losing the customer.

- Fabien, Odoo Founder

03

Roles

Traditional ERP vendors define different roles for analyzing business flows: Project Managers, Junior Business Analysts, Senior Business Analysts, Testers, Trainers, etc. But too many cooks spoil the broth!

Making the right decision always involve a trade-off between specific business needs and existing product features. If you split the role of business analyst and product expert, you might make inefficient decisions.

Odoo, as a product, is much easier to use than traditional ERPs. This allows for one-single person to know both the business and the product - something competitors can't do.

Odoo: Project Leader

The Project Leaders are the main decision makers of the project. However, Project Leaders wear many hats - at the same time, they play the role of project manager, business analyst and product expert.

As project manager, **we lead the project** by:

- defining the project plan and following it up
- focusing on the main objectives
- onboarding the SPoC (Single Point of Contact) on the project
- using the right resources and anticipating the risks

As business analyst and product expert, **we keep things simple** by:

- deciding how to implement specific needs
- challenging the customer’s demands and managing their expectations

- configuring Odoo
- migrating the required data
- writing the specifications (if any development is required)

The Odoo Project Leader has to be considered as *THE* key point of contact by the customer during their implementation.

Odoo: Project Director

On larger projects or highly political environment, a Project Director is assigned in addition to the Project Leader. While the Project Leader focuses on the implementation, the Project Director helps present the project and manage executives' expectations, with a higher view on the project.

Their role is to keep decision makers informed and committed to the project by:

- reporting project progress to the steering committee
- tracking the efficiency of the project
- offering solutions to fix inefficiencies on how the project is handled (on both sides)

As opposed to the Project Leader, the Project Director does not work full time on a project, but overseas it from start to finish. On smaller projects, this role is usually done by the Project Leader directly.

For a large publicly traded company we had a mission to deploy a full scope ERP for 3000+ users, in the middle of a complex merger between two companies.

We started by following their way of managing the project. Being an experienced service company they wanted to teach us how to do things. But after a few months, the project started to slip.

I suggested a new approach to the steering committee, one which aligned more closely with our methodology. We changed the mechanics in place to do it the Odoo way:

- Working using a SPoC and giving a weekly demo (only one person decides, no more committee).
- Challenging every request to see if it can be dropped or done in another way (stick to the standard environment as much as possible).
- Saying *No!* to non-rational time consuming requests.
- Bypassing most of the project team members to get the decision makers involved directly (avoid wasting time in validation cycles).

At first, the customer was frustrated (after all we, a young team, challenged the way a big and experienced company was managing projects), but as the project moved forward, the executives were very happy and we met the deadlines!

- Grégory, Project Director, Odoo BE

Odoo: App Expert

For key apps (finance, inventory, marketing, manufacturing, website), the most knowledgeable person of the app plays the role of an Odoo App Expert. They have developed a deep functional knowledge in industry-specific Odoo features and have acquired solid business experience.

The App Experts are not part of the project. They do peer-reviews, working across all projects of the company. They usually get involved in GAP analyses to perform peer reviews. They especially do this for more complex projects if the level of complexity requires it.

Odoo: Developer

Not all projects require developers. Most small companies (<50 users) use Odoo out-of-the-box and do not require custom development. They'll be involved if, and only if, the business requires development.

Customer: Single Point of Contact (SPoC)

To make the implementation as fast, simple and affordable as possible, we also need to have a strong ally on the customer's side. To do so, the Odoo Project Leader will need an equivalent profile in front of them.

As project manager, the **SPoC works closely with the Odoo Project Leader** by:

- following up with the project
- being an ambassador who convince the end-users (Change Management)
- making sure that the project plan aligns with the company's agenda and constraints

Acting as "super key-user", the **SPoC has a 360° understanding** of the project requirements by:

- gathering and assessing the project requirements
- training the end-users with the support of the Project Leader (there is no better trainer than a colleague who knows your internal processes)
- becoming an internal Odoo expert and ensuring the first level of support for their colleagues

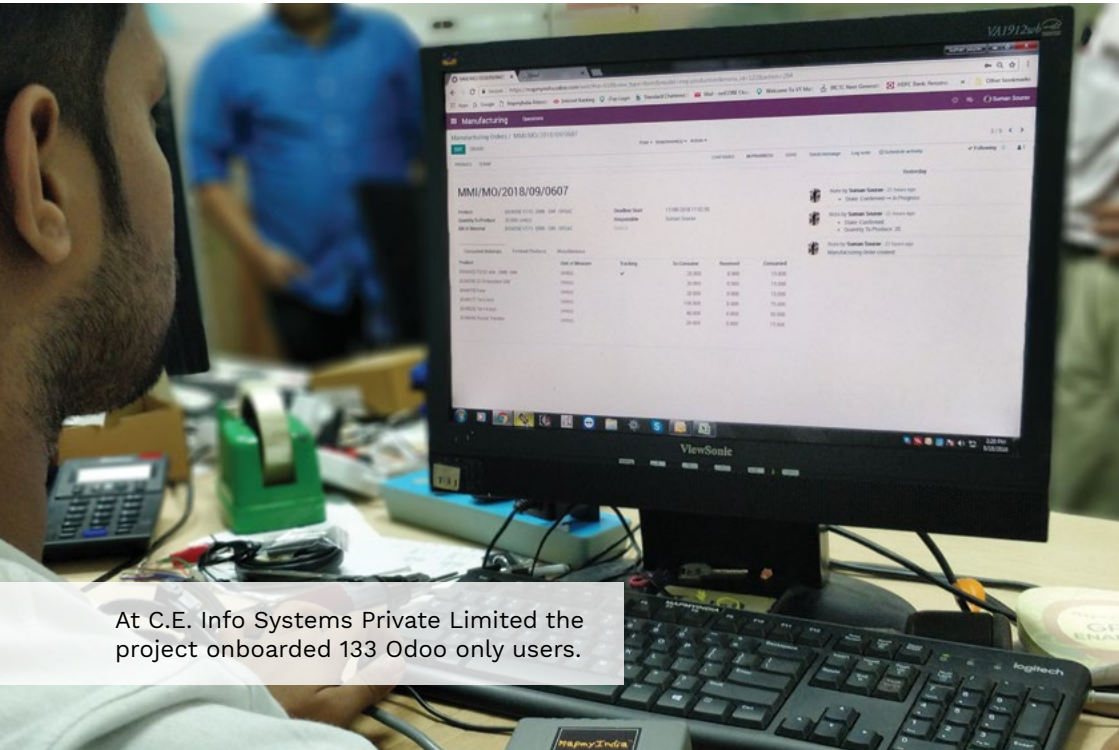
In sharing the responsibility of the project's success with the Project Leader, we expect the SPoC to get involved in every step of the project. Therefore, we need the SPoC to:

- be available for the project
- have the authority to make decisions

Customer: Extra Roles

On large projects, extra roles might be defined:

- **Steering committee:** a committee (composed of customer's decision makers and Odoo's Project Director) that decides on the project's priorities, methodologies, and tracks the success of the project
- **Key-users**: in addition to the SPoC, the key-users act as experts in their specific domain and will help the SPoC to define the requirements. They also test and validate deliverables
- **Sponsors:** usually the CEO or CFO, who pays for the project and who has high level objectives. They're usually part of the Steering Committee as well



At C.E. Info Systems Private Limited the project onboarded 133 Odoo only users.

Two years ago, I started two projects with two manufacturing companies with similar flows and owned by the same person. At the beginning of the project, we had two SPoC: the first was the operational manager of one of the companies, and the second was the CEO of the group.

The first implementation went very well. In a few months, we went full scope into production. It was due to a good collaboration with the SPoC. On the contrary, the second implementation was very difficult to manage due to the CEO's (acting as SPoC) unavailability.

We decided to change the SPoC, but the CEO didn't trust this new person. Every decision had to be validated by the CEO, which was adding days to the process. Discussions with the new SPoC were good, but he had no authority. The project was a nightmare and it took months to implement the first phase.

After the first production launch, we decided to change the SPoC again. The person responsible for the implementation in the first company took over the responsibility to implement the second company for the next phases. The CEO trusted the decisions he would make and no validation was needed. Things started to move forward much faster. Just by improving the decision making process, we increased efficiency.

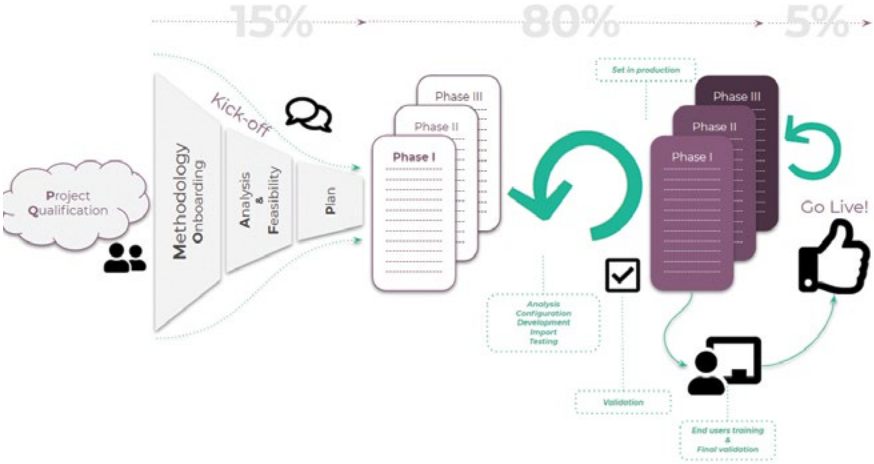
- Benjamin, Project Leader, Odoo BE

04

Implementation Phases

The phases of an implementation project and their relative durations are:

Phase	Time	Goals
ROI Analysis	10%	ROI analysis, phasing & budget.
Kick-Off	5%	Align stakeholders on methodology + standard training.
Implementation	80%	Series of cycles: analysis, development, validation, key-user training.
Go-Live	5%[3]	End-user training, bug fixes.
Second deployment	variable	Broaden scope or add custom features.



Classic Project Journey

3 The Go-Live might require more time in larger projects (from 10% to 15%) due to heavier change management.

ROI Analysis

On large projects, the ROI Analysis is sold before the customer commits on the whole project and budget. According to the size of the project, it can take from 3 days to 50 days. On very small projects (<4 months), the ROI analysis is not a distinct phase, but performed during the Kick-Off phase[4].

The ROI analysis allows the customers to:

- get a precise plan and budget
- assess the Return on Investment (ROI): benefits for the cost
- review their specifications according to the software
- clear their doubts about the feasibility of the project
- assess the Project Leader

The Project Leader delivers:

- an analysis of the savings and benefits for customer (Returns)
- A budget and an implementation plan (Investments)
- a mapping between business needs & product features
- a proof of concept (POC): demos of key business flows

The phases of the ROI Analysis:

1. Meet stakeholders, define the objectives, motivations, and risks in the *ROI Kick-off* (Appendix A) mindmap.
2. Meet with key-users per department using the *ROI Key-user Interview* (Appendix B) mindmap:
    - Understand the current situation

4 We use the templates provided in chapter "8 Extra Reference" only for actual ROI Analysis (larger projects)

- Spot waste of times & pain points (potential savings)
- Define what should be done

3. Document the ROI Analysis & phasing.
4. Peer-review it by Odoo Experts and developers to challenge suggested solutions.
5. Present the results to the customer by using the *ROI Closing presentation* (Appendix D) and make a product demo.

Analysis Tips

During interviews:

- **Be a salesperson**, as the project is not sold yet! At this stage, your goal is to reassure people and motivate them: do demos of key features.
- After reassuring key-users on the project, **assess how people spend their time** (X% on this, Y% on that); a key element to assess potential returns.
- It's always good to **observe how they currently work**: ask for demos of their current software and get a copy of each document they use. The current situation is more important than their future goals, as it defines the minimal scope to cover. If you perfectly understand how they work today, you'll be able to better challenge their requirements and spot their inefficiencies.
- **Identify key-users' pain points.** Use the ROI Analysis template to get ideas of most common pain points per department.
- **Find solutions for each problem**, try to stick to standard solutions as much as possible. It's not required to do everything

key-users ask for, their demands should be challenged.

- **Never present different options to the customer**: as Project Leader, you have to suggest the best one and make the decisions yourself. The customer's role is to challenge what you offer, not to decide what to do.
- **Avoid having the customer decide whether a feature is "necessary" or "optional"** or everything will be mandatory. Make the decision for them by classifying items as "optional", when you think they should be excluded from the scope. Customer's role is to challenge your proposition.

After the interviews:

- A **peer-review** is organized with an Odoo Expert who is external to the project. They are not influenced by the customer and can easily make harsh decisions and provide criticism.
- The goal of the peer-review is to
  1. assess if custom development is really needed and if so, to prioritize it in order to **reduce the budget and planning period** and
  2. check if you didn't miss key pain points according to the industry.
- All necessary **development should be divided into two categories:**
  1. Development that is absolutely necessary before going into production (i.e. you cannot operate the business without it).
  2. Development that can be rolled out in the second deployment phase, after the project goes live (i.e. you can operate the business without it, but it's not optimal).

At the end of the mission:

- **Summarize your analysis** (functional and business coverage, resources required on both sides, planning and risks).
- Organize **specific demos** to reassure the stakeholders and point out the benefits they will get from using Odoo.

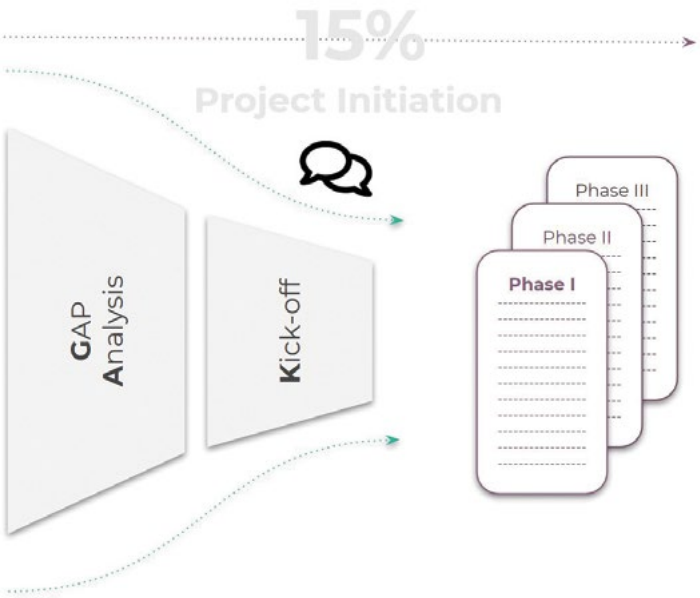
Project Kick-Off

We need to get people on board. That's what the Project Kick-Off is about. Generate buy-in within our customer's company, manage expectations, make them agree with our approach and, above all, build a solid plan!

You should know how critical this step is. The whole project's success depends on the way you will carry it out. That's why you will spend at least 10% of the project on it.

The goals of the kick-off are to:

- onboard the SPoC on the methodology, and align visions
- do a quick ROI Analysis to validate the project's feasibility (if not done yet)
- finalize the project plan
- onboard the SPoC and make sure they invest time in learning Odoo



I once was assigned a project, "Electronics123". The message from the salesperson went along these lines: "This customer ABSOLUTELY NEEDS his Warehouse, Manufacturing, Purchase, Sales Management and Website/eCommerce up and running in 2 weeks. His Netsuite contract ends then and he will be left without a system."
I had only 12 calendar days to migrate his full ERP into production.

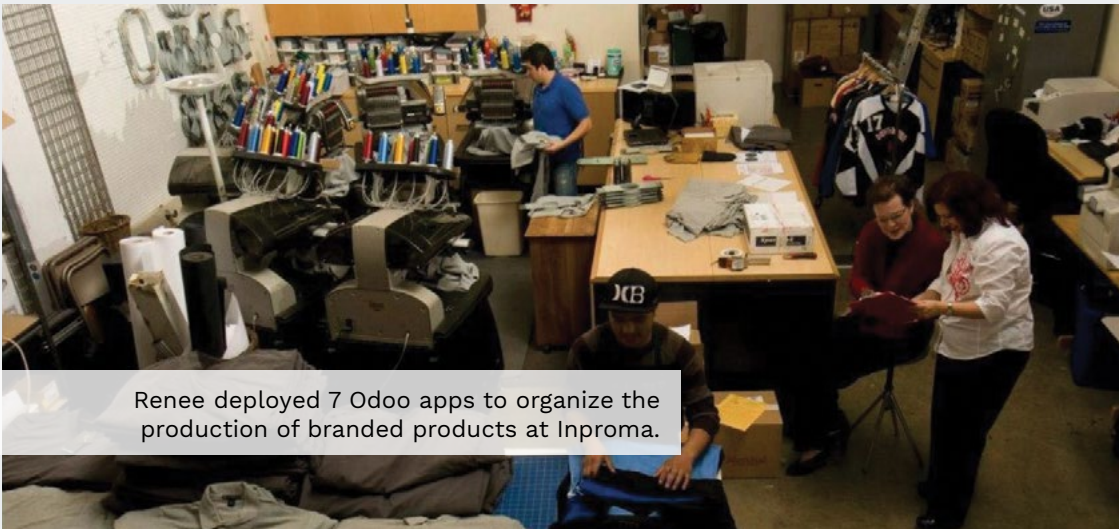
Here is what I told Johan, the CEO, during the kick-off meeting: "First, the project is impossible. We will fail. We usually need 2 weeks per app. But if there is only one tiny little chance to make it work, we have to do this: 1)

We go full standard, 2 You do what I say and you don't ask questions, since I won't have time to explain every single decision I make". He agreed.

We worked night and day together for the next 9 days. He explained his business processes, and I made all the decisions myself while I was configuring the system. The company went into production 9 days later during the night, on all apps. One of the best projects and customers I ever had.

I can't stress enough the importance of the kick-off meeting. This 'impossible' project has been made possible only because the expectations were set right during the kick-off. Also keep in mind that project managers should not be afraid to make decisions on behalf of the customer, it makes the process much easier. The customer's role should not be to decide what to do, but to approve what you suggest.

- Laurence, Project Leader, Odoo SF



Renee deployed 7 Odoo apps to organize the production of branded products at Inproma.

Kick-Off Tips

- **Tackle issues directly**: if you think the planning is too short, negotiate a delay and ask to push the deadlines asap. Similarly, if you detect misunderstanding about feasibility, mindset or features, discuss these ASAP, rather than delaying the conversation. Inexperienced Project Managers tend to avoid difficult discussions, which is a common mistake.
- **Check the customer's involvement:** ensure the right people are involved on the customer's side. Ensure they have enough time and knowledge to fulfill their duties.
- **Invest time in training the SPoC:** introduce them to the eLearning platform, the online documentation, and train them on key business flows. They won't be able to perform their duties if they don't become an expert in the product themselves.
- **Managing customer expectations:** this is a key skill of any Project Leader. Don't set deadlines that are too short, don't promise complex features, don't say the change will be easy, don't say yes to everything. In short, if you promise less, you can over deliver.

I have 2 stories illustrating the importance of followng these rules.

Case 1: Failed implementation
My customer knew exactly what he wanted. Instead of challenging them from the beginning, I accepted everything because he could afford it. Big mistake.

Today, the maintenance costs have started to burden the customer. He keeps asking for additional development. Hey doesn’t understand why I started saying no and now he's not open to alternative (standard) solutions.

As a result, the project has been delayed for several months and I have to admit that the level of satisfaction is not good.

Case 2: Successful implementation
I began the project the other way around by setting the right expectations. I explained to the customer that I will say no to any development requests unless the need is a justified "must have" and there is no appropriate workaround.

Doing this changed everything. The customer is really open to my suggestions and we saved 100 hours of development by using standard workarounds.

- Audric, Project Leader, Odoo BE

Implementation

No matter the level of complexity, the project must constantly move forward.
Keeping a steady pace is a key success factor. The best way to maintain a high level of involvement is to keep the SPoC engaged in everything.

After the Kick-Off phase, the designed solution has been demonstrated to key-users. It's now time to make it come to life!

Within each phase, the project team works in short cycles in order to deliver functionalities every week. The solution is shaped progressively throughout the phase and validated by the Project Leader and the SPoC. The configuration, data import and specific development is handled in parallel by the Project Leader and the developer, if required.

Configuration

The Project Leader configures the software themselves, including customization with the Studio app, but no custom development. Once the apps are configured, the Project Leader involves the SPoC and key-users through a series of training sessions in order to validate the setup.

Data import

Depending on the volume and complexity of data to import this task is handled either by the Project Leader or a developer. Following the Project Leader's instructions, the SPoC and the key-users gather the data and prepare the file to import.

The data migration from the current software to Odoo can generate delays and requires making the right decisions:

- **Don’t delay the production launch due to data quality:** Importing the cleanest data possible is optimal, but not at the cost of delaying the project. So, if your customer didn’t clean it on time and was already using their data in this state, don’t delay a production launch to clean it. Some cleaning can be done directly in Odoo in post-production.
- **Import master data, but avoid the full history** (if possible): it takes a lot of time and money for very low ROI in the long run.

Specific development

The Project Leader is responsible for the success of the project. Therefore, they are also responsible for deciding if custom development (which risks delaying the project) is worth it or not. It’s never too late to question if a specific development is a must have. Remember: the more you cut the amount of development, the better.

At this stage, the Project Leader approves what should be developed; usually that which is necessary to operate the business, not the things that are simply "nice-to-have" (you can operate the business without them, but it’s not ideal).

The Project Leader writes the specifications, including the scenarios to be tested, and the SPoC attests the conformity with the business requirements. Then, the developer takes over the task and completes it. They are also responsible for automated tests.

The Project Leader tests the new features and makes sure they integrate with other features or apps. Once the development is validated they train the SPoC and key-users.

The SPoC also has the responsibility to test and validate the development. If issues are detected, they inform the Project Leader who processes the feedback with the developer to fix the bugs and make the necessary changes.

Validation & End-Users Training

Once all the requirements of a phase have been completed, the SPoC is responsible for all the final tests and gives the green light to go live.

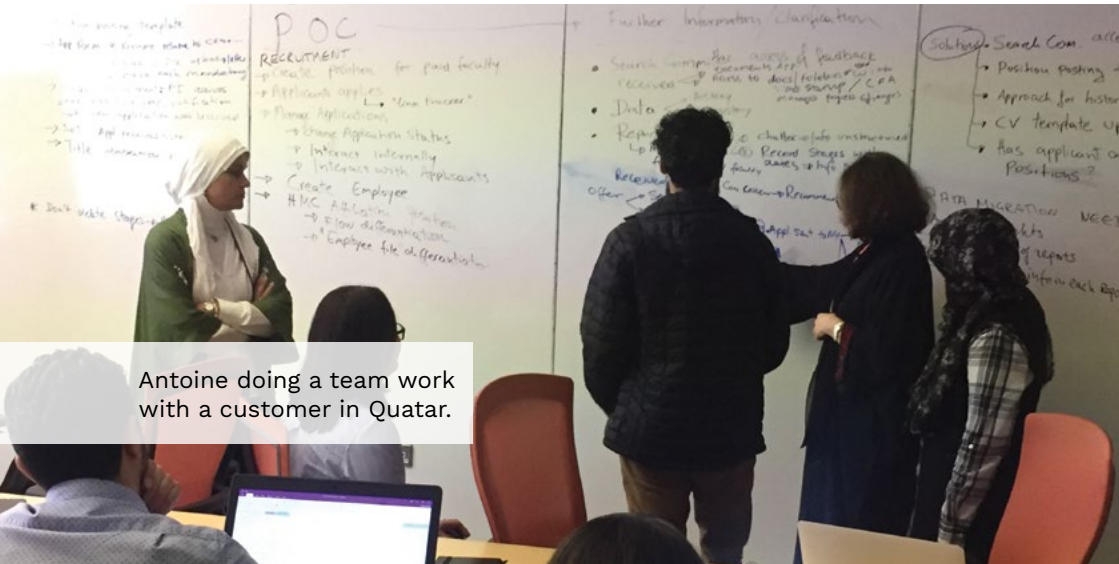
As internal Odoo experts, the SPoC and/or the key-users organize and train all the end-users.

Similarly, writing the user manual is the responsibility of the customer as good documentation should match customer's internal processes and terminology. Additionally, having the customer write the user manual is a good way to ensure they have fully tested the software in "standard practice" before going into production (we should never say something like this. Nothing is ever "too expensive", but instead the cost does not warrant the ROI).

Implementation Tips

- **Ask the SPoC to do the business flow themselves.** They will learn faster. You can guide them, but they must drive the mouse and keyboard. It changes everything for training efficiency and their involvement. You will also quickly sense if they don't understand some key concepts.

- **Transform your project plan into a series of quick wins:** To keep your customer involved in the project, deliver regularly. If users start to use the system, even if it's on a small scope, their knowledge of the system will improve quickly.
- **Keep challenging your customer:** It's not because we have a list of requirements that your customer will stop having new ideas. Generally, you don't accept changes in an ongoing cycle except if the deadline and budget are not impacted. If a change has to be implemented, it will be completed in a later cycle. If the change impacts a requirement to be completed in a later phase/cycle, accept it only if another requirement is dropped to compensate.
- **Don't do something you are not convinced of:** the salesperson's promise can be rediscussed. A contract is less important than the project's success. You can always convince a CEO to not implement an idea (even if it was in the initial contract).
- **Conduct face to face meetings.** It's a good way to unlock complex situations: fear of change, need for reassurance or lack of involvement.



Antoine doing a team work with a customer in Quatar.

Go-Live

When it's time to turn on the switch you may be faced with unexpected issues. there might be unexpected issues. Typically this is due to one or both of the following:

- **The database is not fully tested:** do your best to ensure the key-users have fully tested all business flows.
- **The users are not well trained:** if the training was completed months ago, they might have forgotten. If they did not practice themselves during the training they might have missed critical steps.

Go-Live Tips

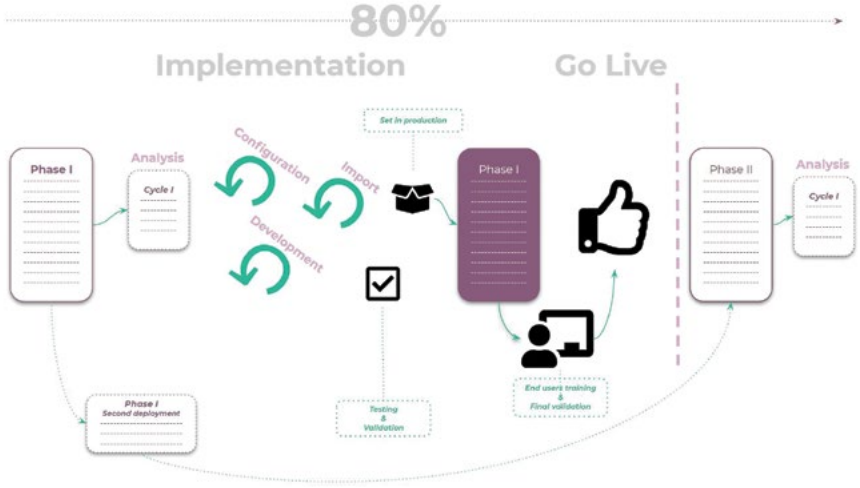
- **A training is not a conference.** Encourage the SPoC to have key-users perform the flow themselves, with their guidance.
- **Key-users are not professional testers.** Quality testing is hard, so they are likely to miss issues. Cross check risky parts with them.
- **React quickly.** It's ok to have issues, if you fix them quickly.
- **Avoid pushing back the Go-Live date.** Although it may feel like the best choice at the time, a lot of things can change in 2 months: people can loos motivation, new change requests will appear, data import may need to be done again, etc. Pushing back deadlines exposes the project to extra-risks and costs. It's usually better to Go-Live fast, even if it's not perfect.

- **Be on-site the first days** of the deployment if there is resistance to change amongst the users. You will coach them.
- **After a few days, check if they really went live.** Sometimes they continue using their old software: habits are difficult to change!

Second Deployment

A month after the initial deployment, the Project Leader reviews the list of the remaining development that were not launched during Phase 1 (i.e. development scheduled at a subsequent phase: you can operate the business without it, but it's not ideal).

With the users' feedback, the prioritization of specific development will usually change (typically we notice that 50% of development was not necessary and new development has been requested).



05

Implementation Challenges

How to encourage users to embrace Odoo

Humans are deeply resistant to change. There is no small change and it is never just a small detail. From the newest employee to the founder, change is a big deal!

To implement a new management software consists of not only replacing a tool used by most employees, but also creating buy-in from the end-users in order to achieve the company's vision. It's definitely a big deal. As you will never use chopsticks to peel a carrot, employees in the company need to be convinced that the new management software is the best peeler ever!

To do this, the Project Leader has these resources:

- **The product itself:** once users are convinced by the features and benefits, the overall acceptance increases. Train unconvinced users on key features that they will benefit from.
- **The SPoC support and the key-users:** their role in the company and positivity about the project in front of the employees will ease the change management.
- **The project sponsors** (i.e.: CEO) supporting the project.

The most successful projects generally benefit from large acceptance from end-users, making the onboarding in Odoo smoother.

How to deal with resistant people

A common mistake is to put aside those who are not convinced (afterall, we all prefer working with people who agree with what we are saying). If someone is not convinced, do the opposite: invest time explaining to them the why/how, and “sell” them the solution with a training.

A change is always perceived as a cost and a risk for the impacted users. And a cost/risk is always worth taking if the benefit you get is much higher. While trying to convince people, don’t say it’s not risky, or not a big deal. **A change is a big deal.**
Instead, you have to **“sell” the benefits** of using the new solution. Organize demos of the product, check how we can solve their pain points, and explain to them why we work that way. If they can see the benefit of the solution, they will accept the risk.

This subcontractor does machine work on windmill parts.
The production floor uses Odoo to manage their machining operations.

During an implementation with Casual Cushions, the accountant was the most resistant person in the company. The production teamwas happy about moving away from the previous system and were extremely optimistic about the project. Months later, the accountant kept challenging me in every training and discussion - we went over each workflow to exhaustion (bank reconciliation was probably the worst one). On the other hand, the production team was liking every training, and they never really had many questions.

Finally, when we went live, everything changed. The accountant accepted the change - and since she put a lot of thought into Odoo and its features, she was ready for it. She knew where to go, what to do, and all her corner cases and possible entries were covered.

On the other side, the production team had a much harder time. They kept forgetting the training, and they presented me with a lot of corner cases in production that were never addressed before. They worked overnight to get caught up in production and were frustrated.

I talked to the accountant last week, and she mentioned how much better the bank reconciliation in Odoo is and that she is happy to be moving forward. When a customer is inquisitive and willing to test and discuss the process (essentially involved in the project), the project is much more likely to be a success.

- Mateus, Project Leader - Odoo SF

How to keep things simple

Customers have a tendency to make things more complex than they are.

- **Avoid giving several options to the customer** and letting them choose what they prefer. Instead, propose the best solutions, and don't show the other options unless the customer is not satisfied with your proposition.
- **Avoid delaying decisions**, force people to decide, even if they are unsure. Prevent them from saying, "we will ask key-users what they think" and "I have to ask a manager about their opinion".

Even if you think you simplified everything, we can always do better:

- **Involve an app expert at critical phases** (e.g. GAP Analysis) for a peer-review. As they're not involved in the project, they can easily make tough decisions and provide you with another point of view.

How to manage customer expectations

A few years ago, the CEO of a prospect asked me to meet before signing the contract. She told me "this project is life or death for my company, please tell me everything will go smoothly". I replied something like this: "No. Such a project is really difficult. We'll have a lot of issues.

But at the end, your company will be better. And I need you, as a CEO to support the project when your teams complain, to get over these difficulties."

I didn't hear back from this customer until two years later when the CEO phoned me. The project was seriously delayed and the key-users lost trust in the product. One of the first things the CEO told me was: "This project is a nightmare, we are 12 months late, and I don't see the end. But I did what you asked me for: I always supported the project, I never criticised your product in front of the team, and always motivated key-users who wanted to drop the project. But, now, we are reaching the limit, I need you to do something for me..."

So, these two minutes with the CEO before signing the contract have actually saved the project. If the CEO would have taken the key-users side, the project would have been aborted months ago. Because she took her role to support the project seriously, the project has been saved, and deployed in production two months after.

- Fabien - Odoo's founder, about the first customer who purchased the Point of Sale app.

This story perfectly illustrates the **power of managing customer expectations**. If Fabien would have said "don't worry, the project is easy", the CEO would have lost trust in the project and probably followed the key-users advice a long time ago when the tension escalated seriously.

How to write a good specification

A good specification is short, visual and structured as follows:

1. Business need: The use case (what) and its justification explaining why the customer specifically needs that feature (max 2 or 3 paragraphs).
2. Functional specification: The suggested solution using Odoo (how) illustrated, if possible, by a series of screenshots, or mockups with notes.
3. Technical hints: Things the developer will have to take care of.

(See Appendix E).

Avoid importing data history

In addition to master data, customers often ask to import the full data history, like quotations and sales going back 7 years. This takes a lot of time, and it will eat a large part of the budget. Because it adds risks to the project's success, we should do it only when it’s really justified.

Ask the following questions to check if it’s really necessary:

- Is it possible to keep that information in the old software, or an export file?
- How often will your customer consult that information? For which purpose?
- What strategic impact might it have in 2, 3 or 4 years?

Just like any other request, as long as the customer can't convince you, the import request should be refused, or delayed until after the Go-Live.

A few months ago, I implemented Odoo Accounting for Ibbeo Cosmetiques, a pharmaceutical group of three companies in France. During the Kick-Off, my SPoC told me that taking over all historic data from Sage was a must. She needed it all in Odoo to check it when necessary. I explained to her that importing historical data put the project's schedule at risk, and that it would take days of consultancy for very little added value.

I made her a deal: we start the accounting for all three companies in only three weeks, with very little effort from her side. If we do it my way, we will have several hours left on the service pack to use in the future. And if, later on, she decides it was still necessary to import the historical data into Odoo, we could do it in a second phase. She agreed.

One week later we had our first call and I imported opening balances and master data. Finally, after the training, the project went live for all three companies after just 2,5 weeks, with 9 hours left on the service pack.

A month ago, she sent me an email in order to thank me for the quick startup and that Odoo was so much more user friendly than the previous ERP she was working

with. She also told me that in the last 3 months she did not need to check a previous accounting entry once and that she was in contact with her sales advisor to add more modules and take over all her activities in Odoo.

As we get more accounting firms joining Odoo, I often have requests to import accounting entries from the past 5 years. Everytime I use this project as an example and let them decide. Either to deploy the project in only 2 weeks with little effort and few hours of consultancy or to deploy in 2 months and pay 4 times more for something they will use once a year.

- Wynand, Project Leader, Odoo BE

Avoid writing documentation for your customer

When implementing Odoo, you might be driven to create specific documentation to ease the end-users' onboarding. Even if it seems to be a good idea, you will realize that the value you can bring isn't worth the time invested. As a Project Leader, you should focus on tasks that only you can deliver.

In other words, Project Leaders should not waste time on repeating explanations already given throughout the project. The customer should be responsible for building their own documentation, based on their own business cases and terminology.

In addition, the SPoC is the person who has the widest business knowledge among all the Odoo implementation stakeholders.

Moreover, letting the client write their own documentation guarantees that the configured Odoo workflows are properly understood and handled by the SPoC. This eases the change management process as the end-users have direct access to a reliable source of knowledge in their own company.

Of course, most standard flows are already covered by existing documentation, Odoo shares all knowledge online[5]. Small projects usually do not require specific documentation.

5 https://www.odoo.com/slides

Odoo is implemented in Lab Society, a factory of laboratory equipments.

How to deal with customers' specific demands

When dealing with customers, remember that there is a difference between the stakeholders' objectives, and the key-user's needs. Most **decision makers' priorities are time and budget**, while key-users will mostly ask for specific features. As these objectives contradict each other, you'll have to decide: who do you want to please?

You should always do what you think is best for the project; this means challenging what key-users request, to the point of refusing to do it if you think it's not worthwhile for the project. Use the following tactics to deal with custom development requests:

1. Is it really necessary?
2. Is it worth supporting the cost (of developing and maintaining it)?
3. Is the gain significant enough?
4. Can we serve the same objective with a different approach?

If you come to the conclusion that developing a specific feature is not worthwhile, you should try harder to convince the customer. There are different tactics for this: explain the "Why", phase it after the "Go-Live" date, escalate to a manager (while not ideal, it's sometimes necessary).

Tactic 1: Is it really necessary?

Let's say you have a request for the following custom development:

> *The customer has a website developed with Magento Commerce and does not want to change their website since they already invested a lot in it. But they would like Odoo to be completely integrated with their Magento website (including products, coupons, follow-ups on abandoned carts, etc.)*

The best way to assess if it's necessary is to check if **the customer already has this feature in their old software**, or not. If the customer records all orders manually in their old software, they can do it the same way with Odoo. We would then recommend first going into production without developing an integration with Magento, use Odoo for a few months, and then decide if it's worth it. Remember, customer's priorities change when they go into production.

In terms of change management, it's easier to roll out business process changes progressively, rather than everything at once. With the modularity of Odoo, it makes sense to deploy in several phases: first, with what the customer absolutely needs to operate the business, and only after moving to the second phase, other features to improve efficiencies.

A key-user asked me to reproduce a complex report that they built every week in Excel. According to the user, this report was very important for the CEO. However, I challenged the user with the aim to know which KPI was relevant for the CEO.

It appeared that the CEO only checked a few amounts, which were the balance of some analytic accounts. Instead of going forward with development, I showed the CEO how to check these balances directly in Odoo, and they were happy about it.

Challenging the customer not only reduces custom development and risks of delay, it also brings a lot of business value.

- Cédric, Project Leader, Odoo BE

Tactic 2: Is it worth supporting the cost?

You need to evaluate the benefit: how many man-days per month will the customer save because of this feature? Often, the customer only accounts for how much time they spend on this kind of task currently, and how much they think they will save in the future. In reality, they will still have to record all data necessary for the computation, deal with exceptions manually, etc. Example: Even if the customer develops a Magento connector, they will still have to solve all conflicts, record price discounts in both software solutions, deal with inventory conflicts

as the synchronization is not real-time, etc.
Then, you need to evaluate the ongoing costs. Often, the customer only sees the "one-shot development cost". In reality, you can multiply this cost by 2 or 3 for the future maintenance and upgrades over the next 5 years.

Note that using a community module allows you to save time in the initial development, but you'll still have the testing, maintenance, project delays and upgrade to account for in the cost. A community module is a technical debt too.

Tactic 3: Is the gain significant enough?

Let's say you have a request for the following custom development:

> *When we confirm a sale order for a construction project, we want to create a series of tasks based on a set of rules that depend on the products, customers and locations.*

When you have a request for customization, you should first question the volume: “How many construction projects do you win per month?” Let's say the customer wins 10 of these projects per month. It probably takes 10 minutes to create and update tasks by duplicating and updating project templates.

Is it worth launching a complex development to save under 2 hours of work per month? Surely not, this feature will cost around 10 days of development, plus 25% of this every year.

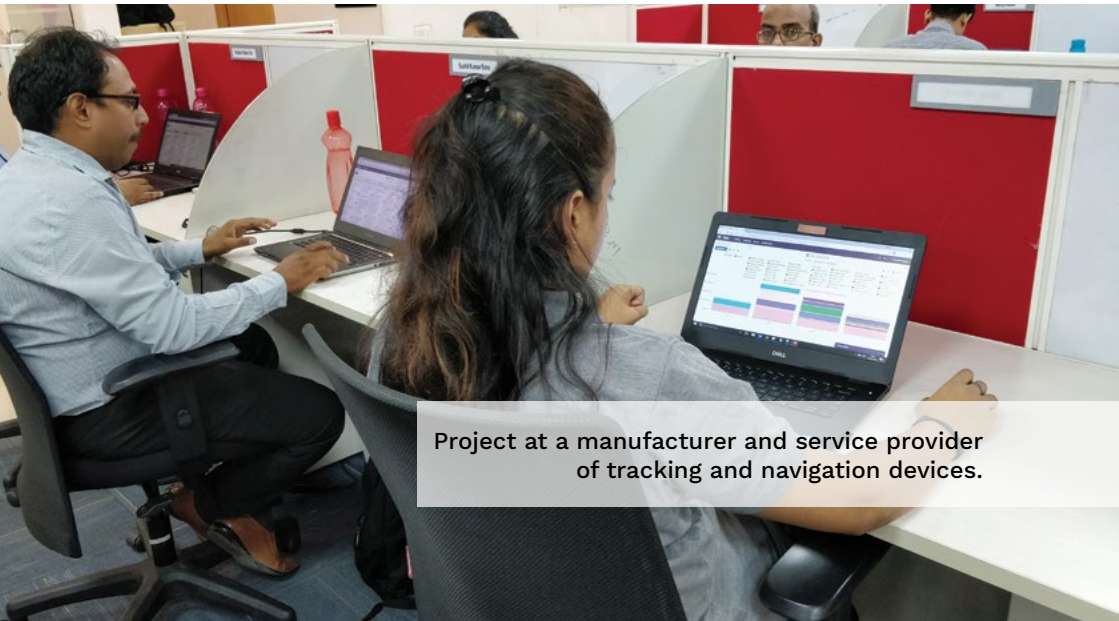
Tactic 4: Can we serve the same objective, with a different approach?

Let's say you have the following customer request:

> *I want to synchronize our Outlook calendar with Odoo CRM.*

Odoo has a connector with Google Calendar in standard, but not with Outlook. But developing and maintaining a connector might cost a lot of money. However, there are plenty of services that synchronize Google Calendar with Outlook (such as IFTTT). So, a solution would be to subscribe and setup such a service for every employee.

It's not perfect as you will have to modify your setup every time you recruit a new employee. But the solution is ready in 2 hours, and it will only take 10 minutes per new employee. It's still much less than a new development, if the company has less than 100 employees.



Project at a manufacturer and service provider of tracking and navigation devices.

A couple of years ago I did an implementation for Bioulvax, a Belgian wine producer. The SPoC for the project was passionate, but he was convinced that his way of working was the most efficient and that everyone in the company should work like him.

We didn’t do a good job since we compromised and accepted some of his requests. We had date-fields that were almost never used, selection-fields that filled up screens for no good reason, menu items that were not user friendly, etc. After a couple of months with the system, we noticed that a lot of employees just did not use it, they were reverting back to their old software without telling the management.
When we realized that, we suggest the following: "let's revert back to how the system works in full standard and let's talk to the team to see what they want to get out of the system". In other words, we asked people what they wanted, instead of forcing a specific way of working. We organized demos fully based on the standard with the objectives of the team in mind.

They could not believe how easy it was to use the system and, when going live again, the entire team adopted Odoo and dropped their old habits. I learned to always challenge the SPoC's requirements, and keep focusing on the business objectives. Often, customers know their business, but not how to implement it.

- Cédric, Project Leader, Odoo BE & SF

Why should you minimize custom development?

For customers, a custom development creates additional cost and timeline delays, sometimes to the point of putting the project at risk. In addition, custom development leads to technical debt that the customer will have to pay for within the coming years in the form of maintenance and upgrade costs. Such a technical debt costs around 25% of the development cost EVERY YEAR (~17% in maintenance + ~8% in upgrades).

Each development may seem simple and affordable. But, as already explained, the complexity of a project grows exponentially as the number of customizations increase, not linearly. Customer's probably want to solve what they didn't like in their previous software, but wouldn't it be better to standardize their business processes and implement the project two times faster and half the price?

For implementation service companies, custom development usually comes at a high cost, for low customer value. This is the typical scenario: you estimated 10 days to develop a feature; the customer thinks it's too much for such a basic feature so you charge for only 8 days, but in the end, you spent 12 days on it. We later discover issues/changes needed because you had to do it in a hurry and the customer won't pay for the extra day since it's your fault. What should have been a 35% margin service is now a 6% loss on service!

To grow, it's easier to focus on valuable services with better margins, and decreasing the risk of non-billable hours. Such services include: project management, business analysis, customizations without development, change management, and training.

Service companies who build a mindset on reducing custom development, sooner or later, will be more competitive than the others. Companies who better manage customers' expectations will have lower project implementation costs. Of course, custom development is sometimes necessary to run a business. But, from our experience, the majority of customer requests are either not worth the cost and not relevant once they start to use Odoo, or they can get by without it as it’s not part of their core usage.

Mecatis is an engineering company specialized in conception, production and maintenance of manufacturing machines. They started using Odoo community in 2013 and were stuck with a very old version. In 2017, they contacted us to upgrade to Odoo 11. We discovered 4 custom modules, and 55 community modules on their database.
They spent tens of thousands of EUR in development and maintenance, which is not normal for a company of 10 employees. That was a huge technical debt, and the cost to upgrade these modules would have been astronomical.

Instead of upgrading these modules, we challenged every single one: some modules could be replaced by existing features in Odoo 11, some features were not used, and for the rest, the customer did not even know what these modules did. So, instead of upgrading their custom modules, we trained them on the standard, and helped them change the way they worked. In only 100 hours of service, we moved them to Odoo Online, without a single development. Their monthly cost has been reduced by a 10x factor.

I am sure that during the implementation, Mecatis, or their partner, thought they needed these community modules. But in the end, it turns out that it was more beneficial for a small company to stick to what is standard in order to avoid being stuck in old versions. After the move to Odoo Online, the customer was so happy that we won a good sponsor; they subscribed as partners to resell Odoo Online to their own customers, delegating the service to our teams.

- Fabien, Odoo's founder

How to deal with internal politics

When something goes wrong, people try to find someone to hold responsible and come up with excuses to defend themselves. This often happens when several service companies are involved in the project. In the case of a failure, they will claim they are not at fault and say it was the other's responsibility.

Internal politics is a game where everyone loses. The best way to deal with this is to avoid playing the game. When a project fails, it's usually everyone's fault. So, when shit happens, don't waste your time arguing about who is responsible. Don't waste your time building a defense either.

Instead, focus on moving forward and solving the issue (whether the issue is yours or not; we don't care who is responsible, we care about the project's success).

Minerex is a company that buys different building materials from the USA, then imports and distributes them within Mexico. Their offices are all based in Mexico. Before 2016, they were using SAP to manage their business.

The owners of Minirex were living in the US (Jacksonville, FL). They decided to implement Odoo, but they were not involved in the company's operations. They were not aware of the different processes or business documents used by the company, etc. The company was effectively

operated by the employees in the Mexico office.
And this was a recipe for failure. Why?

The decision to implement Odoo was made by the owners, without having the buy-in from anyone from the Mexico office. So, since the Kick-Off call, it was evident that none of the employees in Mexico were looking forward to implementing Odoo. Since no one had asked their opinion about it, they saw Odoo as something that was being imposed onto them (i.e. "the owners are just trying to save money, dumping this erp software onto us"). Throughout the implementation, they were very resistant to change: everything was slow moving, implementation was low priority for them, etc.

Initially, we were not aware of this situation. As soon as we figured it out, we had a conversation with the owners. Both the owners and the consultant travelled to Mexico in order to gain buy-in from the Mexico office. We showed them what Odoo was capable of, and how it would be able to handle their processes more efficiently than their previous SAP system. It wasn't until that moment that the implementation started really moving forward.

In conclusion, make sure that the key-users are on board before jumping into an implementation. In the end, it will be these key-users that will use Odoo, and collaborate with you during the implementation process.

- Miquel, Head of Project Leaders, Odoo San Francisco

How to deal with different people's dynamics

Managing many projects at once is not easy and having to adapt your speech to every single person working with you makes it impossible. However, it sometimes helps to detect different personality profiles:

“Do it now” gets right to the point.
Your SPoC will generally not invest enough time in their learning process and might go too fast for the end-users to be correctly onboarded in Odoo. Your actions:

- Make sure that the SPoC knows Odoo properly (triple check).
- Make sure that the SPoC communicates enough internally about the project (check that they take the time to prepare the speech they are going to give to the end-users).
- Make sure the resistant people get involved in the project (greater involvement in the key-users selection).
- Get involved in the final-users training (train in tandem).

“Do it right” respects the rules down to the smallest detail.
Your SPoC might be resistant to any change “because we’ve always worked like that” and question your legitimacy in suggesting a new approach. Your actions:

- Skillfully challenge your customer’s requirements (focus more on the added value than on the fact that it’s standard).
- Involve App Experts faster to strengthen the legitimacy of your suggestions.

“Do it harmoniously” has a good overview of their business and expects the same in their project. Your SPoC might want to have full control over all situations (from the smallest details to the big picture). Your actions:

- Make sure key-users attend courses on https://odoo.com/learn.
- Make sure that they become knowledgeable in Odoo (extra training if required).

“Do it together” is highly flexible, solution oriented. Your SPoC will have 1000 ideas per minute and change their mind often. Your action:

- Make the rules crystal clear: The SPoC expresses the business needs (what & why) and you make decisions on how they should use Odoo (how).

Why should young Project Leaders feel confident?

We have seen too many young Project Leaders thinking they are not "good enough" in front of experienced people and thus they don’t stand up for what they think.

Experience is overrated. Business decisions should always be supported by common sense, not (only) by past experiences. You don’t need to be experienced to be rational. Sometimes, experience is a liability more than a proof of knowledge: people do things “by experience” (habits), and not because it’s the most rational approach according to the context.

For that reason, young Project Leaders should not be afraid to defend their point of view, and challenge experienced people. This will force experienced people to explain why they think a certain way. If they are rational, their experience will convince you. If not, feel free to defend your point of view.

I have hired more than 60 Project Leaders over the last few years and most of them were fresh graduates. These juniors are open minded, motivated, full of energy and want to prove themselves.

Our project implementation methodology combined with the constant evolution of the product make our “newbies” learning curve almost vertical in their first months as Project Leader. Rather than experience, good onboarding gives them the right tools when convincing customers.
The crucial message I pass on to my newbies is always the same: no matter the knowledge and experience, the most important skill to succeed as a Project Leader is having a strong mindset.

-Catherine, Head of Project Leaders, Odoo Belgium

06
Quiz

Use Case 1

During the implementation of a 9-month project, a key-user requests a change that will save them 4 hours of work every week. The user tells you that it's the main reason why they want to change their software. Unfortunately, the feature is not standard, and it is estimated to take 2 weeks of extra development.

What will you do?

- **A** If the customer is ready to pay; it's ok to develop what will satisfy their need.
- **B** Try to convince the customer to avoid the development, but ultimately accept if they really want it.
- **C** Add this feature to the backlog of development to do after the Go-Live.

Answer: C. Since the user is not using such a feature in their current software, they can live without it for a few extra months. Avoid an extra delay to the project: an extra 2 weeks doesn't seem a lot, but you never know how many decisions like this one you'll have to face during the course of the project.

Use Case 2

The Project Manager of a company with 20 employees wants to add an additional validation step on employees' expenses: any expense higher than 500€ should have a second approval by the CFO. You estimate 2 days of extra development.

What will you do?

- **A** Add the new validation steps in order to fit with the company's constraints.
- **B** Define an internal policy (ask your manager and CFO for >500€ expenses) and ask employees to send an email to both of them.
- **C** Refuse to consider it as an acceptable need.

Answer B: Small companies often change the way they operate. Thus, it's usually better to define a policy, rather than developing a custom feature. All you have to do is to send the procedure by email to employees: ask your manager and your CFO by sending an email. As opposed to a rigid development, policies can be adjusted easily and still allow common sense to prevail (e.g. if your manager is on holiday, you can only ask the CFO to validate).

Use case 3

During a remote weekly delivery validation session, you are demoing your work (configuration and minimal customizations) on the customer's billing cycle. All of a sudden, the CFO reveals himself in the audience by strongly indicating that a mass invoicing validation feature is missing and that (overall) your demo is a fiasco. The key-user for this area turns silent and refuses to react to your inquiries, in spite of the fact that the latter actively participated in all previous sessions. How do you get disentangled?

A You apologize, shut down your computer & go home... tomorrow is a new day.

B You continue repeating that everything has been done according to the key-user.

C You apologize, agree with the CFO and promise to get it fixed.

D You remind them of the basics by referring to the GAP analysis.

Answer D: Recall the GAP analysis by sharing it on your screen. You pinpoint that project success is defined by the responsible ownership of the key-user for that area as explicitly agreed with all stakeholders at project start. You are willing to address the CFO together with the key-user to circumscribe his worries in a separate session, noting that (if assessed appropriately) the need will be put in the project parking lot for everyone's post Go-Live requests. Escalation might be appropriate since an alternative could be dropping existing backlog content to be replaced by their needs.

Use Case 4

For a project status meeting, the customer invited stakeholders of each of the 7 departments. They have 10 representatives in the meeting. How many project leaders of Odoo should attend this meeting?

A 1 or 2, more would be a waste of time.

B 4, to feel 'serious' compared to the 10 brought forth by the customer.

C 7, one for each department to integrate.

D 10, as much as the customer.

Answer A: We have to be the most efficient possible. If you have experienced project managers, one person is enough. However, it's good to invite new business analysts to the meeting for them to learn. Or, if the project manager is not comfortable on a topic, he can ask the help of an expert.

Use Case 5

Imagine the following situation:
Before the Go-Live, you have a meeting with the CEO. You have had a lot of issues during the project, they are not confident about your solution anymore, and they are scared about the Go-Live. The CEO is thinking about pushing back the Go-Live another 6 months. They meet with you and say "My company can't afford more issues. To accept the Go-Live, I need you to tell me everything will be smooth."

What will you answer:

- **A** Pushing back the deadline is actually a good idea to reduce risks.
- **B** Don't worry, everything is under-control; we tested everything.
- **C** A go-live is always difficult, but we will fix issues quickly.

Answer C: A go live is always difficult: shit will happen, even if we push the deadline by 6 months. That's normal. And I need you to support the project when the team complains. On our side, we will fix issues quickly as they rise. Pushing back the deadline by 6 months will increase the cost of the project, and put the project's success at risk (so many things can change in 6 months). It's ok to be honest and upfront about the challenge ahead. If the CEO sees that you know what you are doing, and are transparent in your approach, they will trust you.

07

Measure your Progress

Measure your progress

Here are milestones that most of Odoo's Project Leaders reach as they move forward in their career. Use them to assess yourself.

NOVICES

- ☐ Have a project on time & on budget **1 Point**
  Deploy a customer in production within the initial timing and budget.

- ☐ Receive a gift from a satisfied customer **1 Point**
  Have a customer so satisfied that they offer you a gift.

- ☐ Deploy min 4 apps, in 2 weeks per app **1 Point**
  Example: Sales, Purchase, Inventory, CRM, Invoicing in less than 10 weeks.

- ☐ Deliver a project at 80% of initial budget **1 Point**
  Deploy a customer in production in 80% less hours/days than the initial quote.

- ☐ Deploy a company of 25 users, alone **1 Point**
  Put in production with no help from another expert.

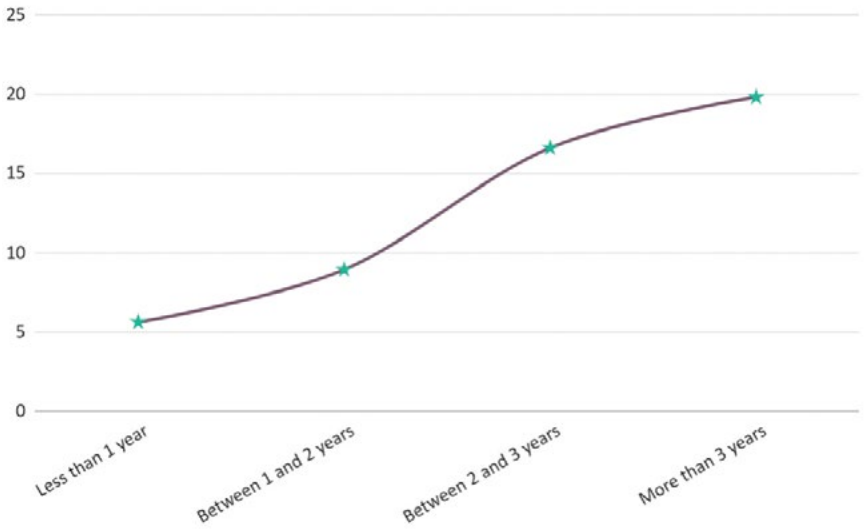
EXPERIENCED

☐	Pass Odoo's certification with 70%+	**2 Points**
☐	Go-Live within 70% of the initial budget Deploy in production in less than 70% of the initial time estimated.	**2 Points**
☐	Success in 3 different industries Have implemented a project successfully in 3 different industries.	**2 Points**
☐	Migration from a traditional ERP in 2 months Move from Netsuite, Dynamics, SAP to Odoo is less than 2 months before the Go-Live.	**2 Points**
☐	Have 3 projects in a row on budget Deploy 3 projects in production, without an extra success pack.	**2 Points**

EXPERTS

- ☐ Implement 500 users in production **3 Points**
- ☐ Save a customer from bankruptcy **3 Points**
  Have a customer who tells you that you saved their company from bankruptcy.
- ☐ Have 10 projects on budget **3 Points**
- ☐ Did a migration of a traditional ERP in 4 weeks. **3 Points**
  Migrate a traditional ERP (Netsuite, MS. Dynamics) to Odoo in less than 4 weeks.

Your Results



Points / Years of Experience @Odoo

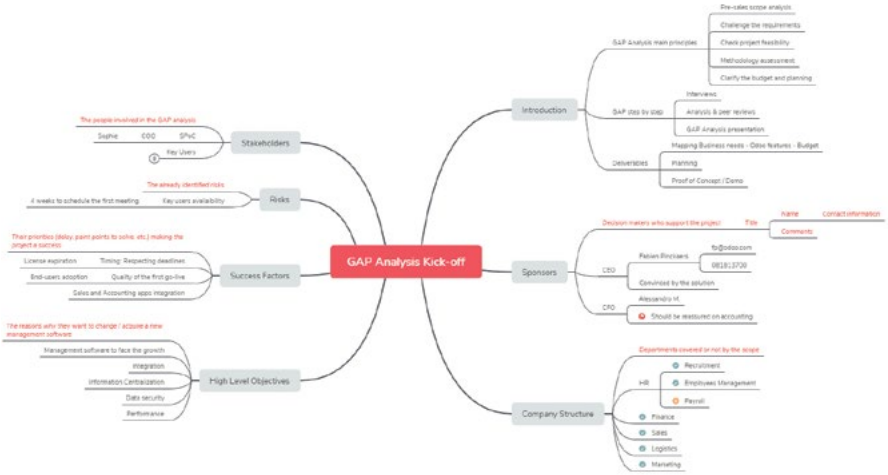
08.
Extra References

Appendices

Appendix A: ROI - Kick-Off

Use **XMind**[1] to take notes during the ROI Kick-off meeting with the SPoC and the decision makers. With this template, start from the top right element “Introduction” and move clockwise during the interview (the left nodes are for notes taken during discussions).

Download the template: ROI Kick-off:
https://www.odoo.com/r/roi_kickoff



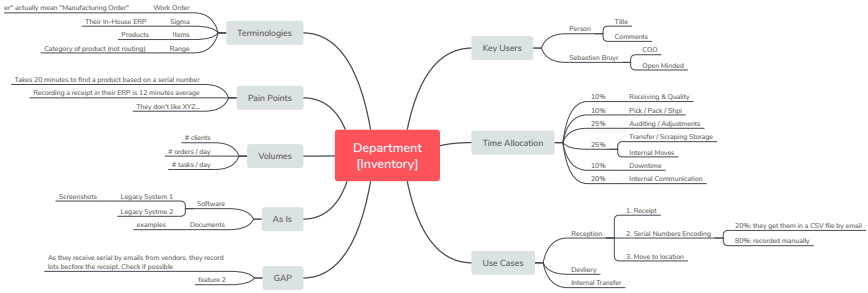
ROI Kick-off Template

1 https://www.xmind.net/

Appendix B: ROI - Key-users Interview

Use **XMind**[2] to take notes during meetings with the key-users. With this template, start from the top right element “People” and move clockwise during the interview (the left nodes are for notes taken during discussions).

Download the template: ROI Key-user Interview:
https://www.odoo.com/r/roi_key_user_intw



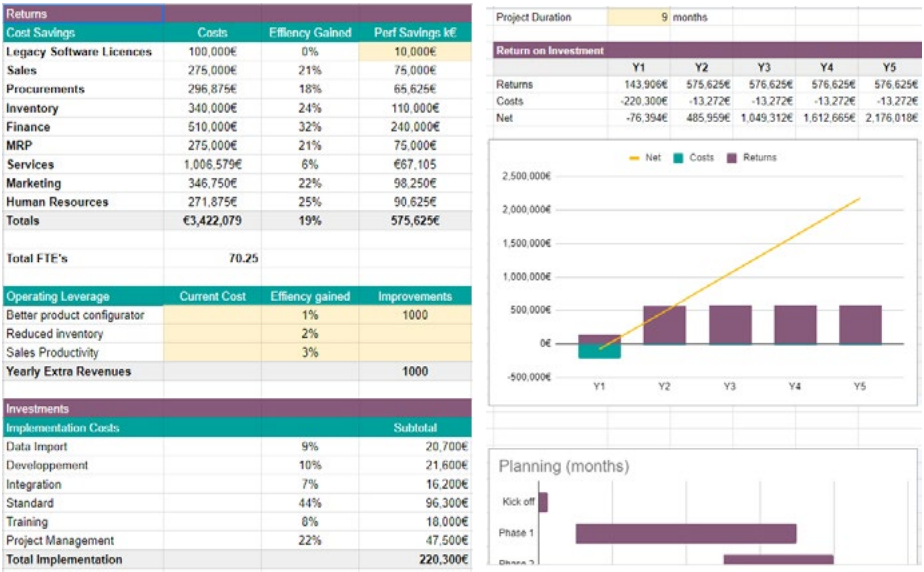
ROI Key-user Interview Template

2 https://www.xmind.net/

Appendix C: GAP - Analysis Tool

Use a Google Spreadsheet to document your GAP Analysis.

Download the template: ROI Analysis Tool:
https://www.odoo.com/r/roi_analysis



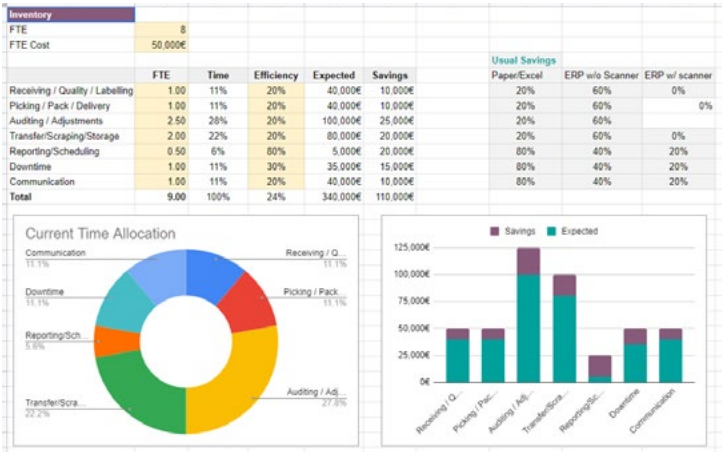
Returns			
Cost Savings	Costs	Effiency Gained	Perf Savings k€
Legacy Software Licences	100,000€	0%	10,000€
Sales	275,000€	21%	75,000€
Procurements	296,875€	18%	65,625€
Inventory	340,000€	24%	110,000€
Finance	510,000€	32%	240,000€
MRP	275,000€	21%	75,000€
Services	1,006,579€	6%	€67,105
Marketing	346,750€	22%	98,250€
Human Resources	271,875€	25%	90,625€
Totals	€3,422,079	19%	575,625€
Total FTE's	70.25		

Operating Leverage	Current Cost	Effiency gained	Improvements
Better product configurator		1%	1000
Reduced inventory		2%	
Sales Productivity		3%	
Yearly Extra Revenues			1000

Investments		
Implementation Costs		Subtotal
Data Import	9%	20,700€
Developpement	10%	21,600€
Integration	7%	16,200€
Standard	44%	96,300€
Training	8%	18,000€
Project Management	22%	47,500€
Total Implementation		220,300€

Project Duration	9	months			
Return on Investment					
	Y1	Y2	Y3	Y4	Y5
Returns	143,906€	575,625€	576,625€	576,625€	576,625€
Costs	-220,300€	-13,272€	-13,272€	-13,272€	-13,272€
Net	-76,394€	485,959€	1,049,312€	1,612,665€	2,176,018€

ROI Analysis - Overview



Inventory								
FTE	8							
FTE Cost	50,000€							
						Usual Savings		
	FTE	Time	Efficiency	Expected	Savings	Paper/Excel	ERP w/o Scanner	ERP w/ scanner
Receiving / Quality / Labelling	1.00	11%	20%	40,000€	10,000€	20%	60%	0%
Picking / Pack / Delivery	1.00	11%	20%	40,000€	10,000€	20%	60%	0%
Auditing / Adjustments	2.50	28%	20%	100,000€	25,000€	20%	60%	
Transfer/Scraping/Storage	2.00	22%	20%	80,000€	20,000€	20%	60%	0%
Reporting/Scheduling	0.50	6%	80%	5,000€	20,000€	80%	40%	20%
Downtime	1.00	11%	30%	35,000€	15,000€	80%	40%	20%
Communication	1.00	11%	20%	40,000€	10,000€	80%	40%	20%
Total	9.00	100%	24%	340,000€	110,000€			

ROI Analysis - Department Returns

To modify Accordingly to the project				
Apps / Departments	Phase	Ressources	# Resources	Kick-off #weeks
Sales	Phase - 1	Project Management	1	2
Human Resources	Phase - 2	Business Analysis	3	
Department 3	Phase - 3	Development	4	
Department 4				
Department 5				

ROI Analysis Template - Tab Investments

Appendix D: ROI - Closing Presentation

Use Google Presentation to document your ROI Analysis. Link all your graphs and tables with the GAP Analysis Tool for an automated update of the information.

When it comes to submitting the results of your analysis to your customer, it's important to give a clear and clean overview of the coming project to the decision makers.

Download the template: GAP Closing Presentation:
https://www.odoo.com/r/gap_closing

Appendix E: Specification Example

How to write a good specification: https://www.odoo.com/r/Spec_example

Extra References

- Odoo Blog Article: "*The key to Implementation Projects: Manage Customers Expectations.*" by Fabien Pinckaers[1].

1 https://www.odoo.com/r/blog_implementation


odoo