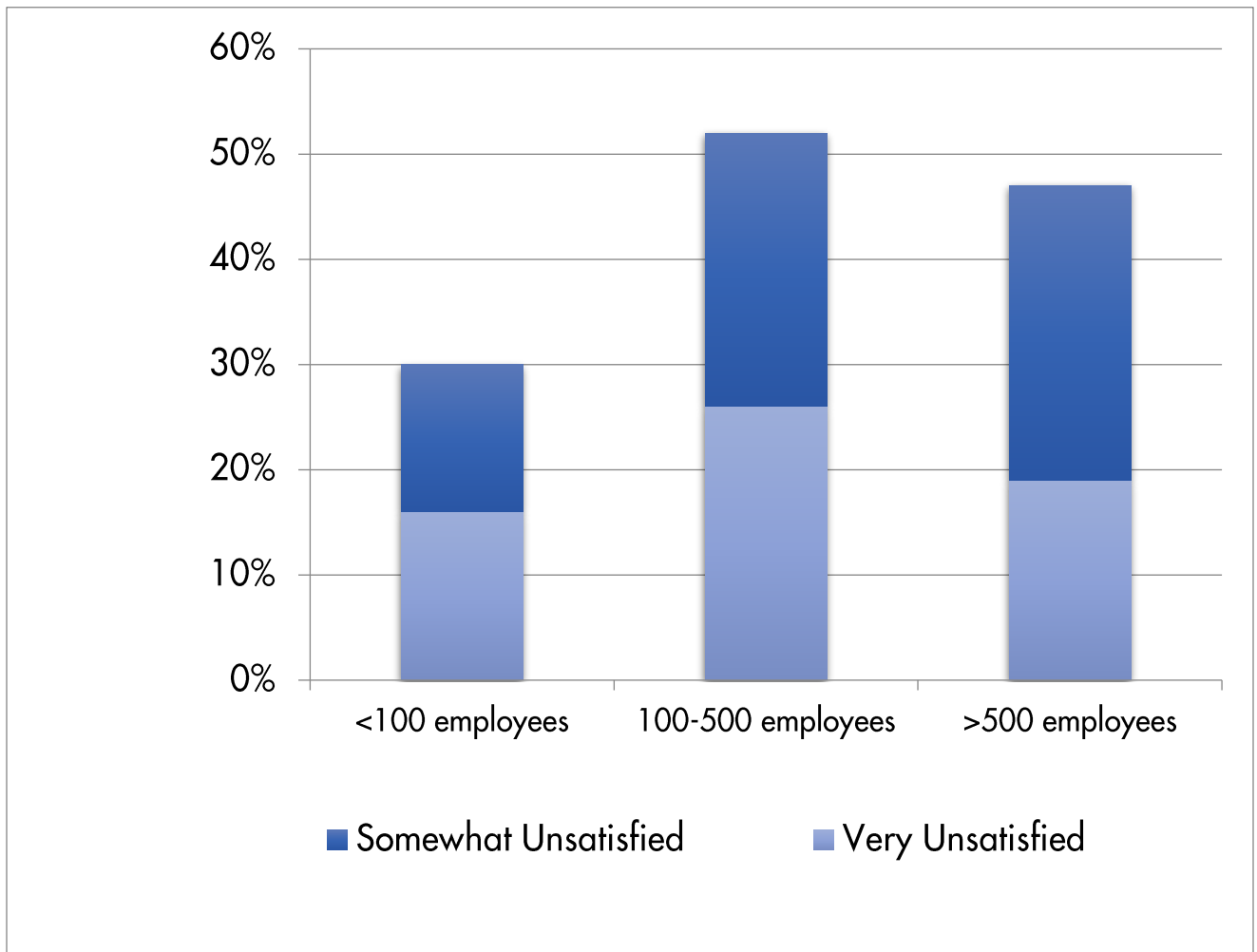
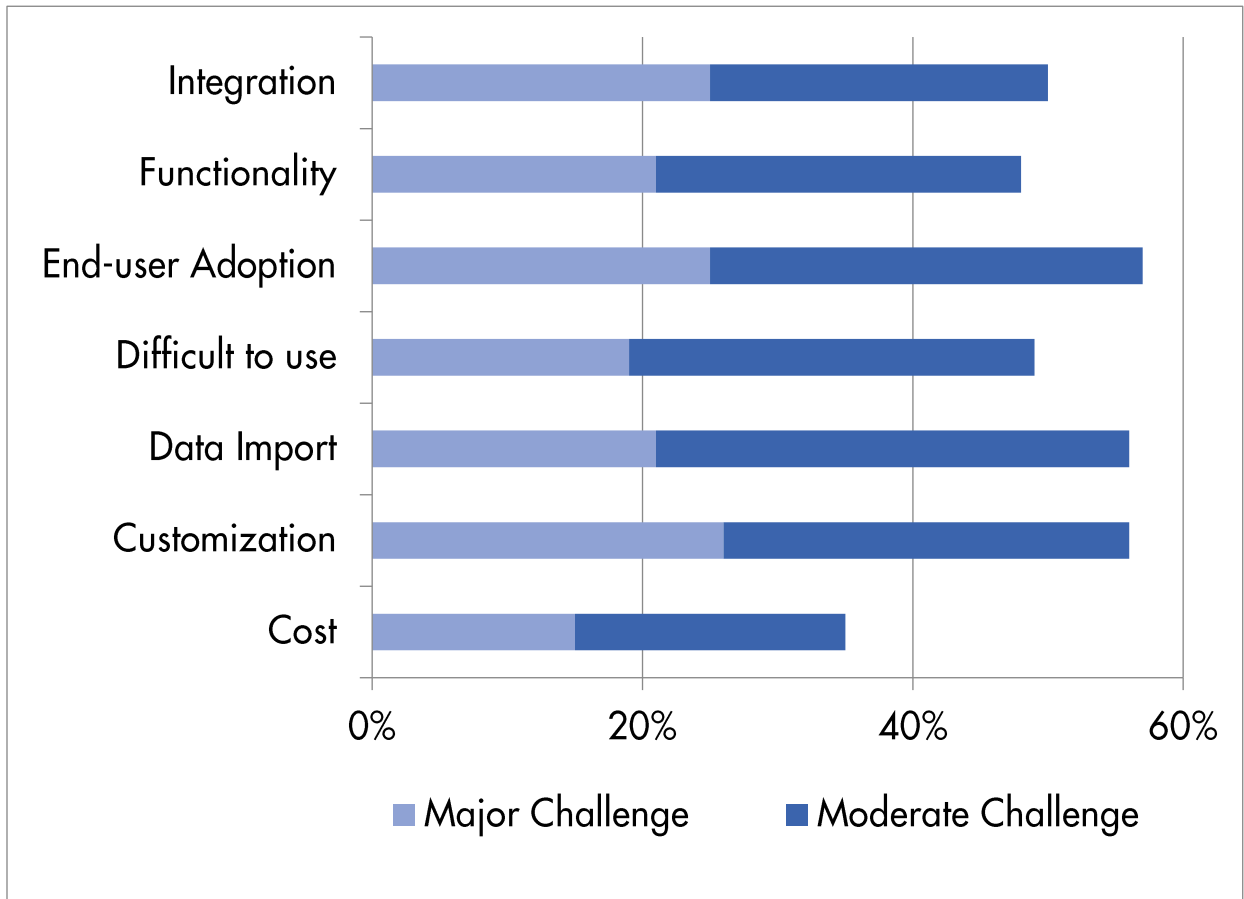


Top Reasons Software Implementations Fail

In a survey conducted by SoftwareAdvice.com in 2014 (<http://www.softwareadvice.com/crm/userview/report-2014/>), it was discovered that nearly half of all businesses were dissatisfied with their CRMs that they had purchased. And the business type that was most dissatisfied were mid-sized businesses. This is because small businesses have very little demand from their CRM solutions, and large businesses have high demands but have the budget to be able to customize a solution to the level needed for their purposes.



The survey then went further to determine the main reasons why businesses were dissatisfied with their CRM solutions. The results were as follows:



The challenges facing businesses as they look to purchase software solutions for their business can be broken down into three main groups:

1. Misalignment between the capabilities of the software and the needs of a business.
2. Problems with the management of the software as any internal manager doesn't have full access to the software and most SaaS vendors don't provide enough management to support a business's needs.
3. The cost of the software or of the unforeseen costs of additional software purchases and the labor it takes to manage the software.

Taking each one of these items on its own, we'll address first, how to identify the true capabilities needed from a software solution, next how to properly scope a project to understand the amount of management it will take to implement and manage a new software solution, and lastly how to determine whether the cost of the software solution is justified for the amount of gain expected from it.

To understand the needs of your business and how to identify which software can meet those needs, focus on Benefits rather than Features.

As with any purchase of a business, there needs to be an analysis of what are the reasons for it. These needs are often defined only at the highest level before investigating solutions. As an example, the statement, "We need to purchase a CRM" is a very broad statement. The more granular you can get with your definitions of scope, the more easily able you'll be to ensure that a software can meet those needs. The mistake that most businesses make at this point, is they don't think about what benefits they are looking to achieve from the software, but immediately start putting together a feature list. So, rather than saying, "We need a CRM that integrates with email," you should first be saying, "We need a CRM that helps our staff email more effectively." By focusing on benefit statements rather than feature statements, you can better differentiate between different SaaS vendors, and determine which solution seems to be the best at facilitating employee email, rather than simply checking boxes for each vendor. Furthermore, by focusing on benefit statements, you can rate a vendor, and better understand **how** they are achieving this benefit, rather than simply being told by a sales person that it is something that their software does. This has the added benefit of lowering the risk that the sales person is misrepresenting the actual capabilities of the system. It's much easier for a salesperson to answer feature related questions like, "Does your software do this?" inaccurately than it is for them to answer questions like, "How will your software benefit us in this specific area?"

To determine what the management of the software will look like, focus on clear Statements of Work and Service Level Agreements.

Most businesses learn the hard way that most SaaS vendors behave like house contractors, in that even when they can complete a project, once they have received payment, their incentive to complete tasks on schedule all but disappears. On the other side, many SaaS vendors have legitimate complaints around project bloat and unrealistic client expectations. To limit the problems with projects, have the vendor draw up a very clear statement of work that clearly outlines all of the deliverables you are needing. These should include sample screenshots, reports, metrics, other systems with which they will interface, and other items that are you needing. This is best drawn up by the vendor as that ensures they understand your needs and how their software will solve those needs. The other part that needs to be clearly outlined is what the day-to-day management of the system will look like, how often new features will be posted to the system, and if there is the availability for ongoing customization of the system. One thing to be aware of, is when vendors often promise that you will always be upgraded to the newest version of the system, this is not always a good thing. Often, major version changes aren't as stable, they change the user interface in such ways that can require additional training, and they sometimes aren't backwards compatible to older data, causing issues. Usually try and find a vendor that will give you advanced notice of such upgrades, and give detail of what the modifications are.

Properly determine the entire cost to benefit of a project by clearly defining objectives to be achieved through a new software purchase.

One of the biggest mistakes that is made when purchasing a new software solution is the benefit analysis where you determine the amount of additional revenue or revenue saved that can reasonably be anticipated through the purchase. It is not enough to believe that you need a new tool: it must be determined before you create a requirements list. Often to do this effectively a study must be done in advance to determine the area of greatest need. As an example: if you are looking at purchasing a CRM to assist with the sales cycle of new customers, you need to first determine how efficient and effective your sales people currently are. If, compared to industry averages, they are spending 20% longer to

achieve a successful close due to lack of scheduling tools, or they are losing 20% more qualified leads due to lack of nurturing, then you can then calculate what possible new revenue can be expected through purchasing a CRM, or look at alternative tools that might achieve similar results to a full CRM purchase that could come at a lower cost.

Once you have completed your cost analysis to determine the amount that can be saved and/or increased, you'll want to use at most 50% of that budget towards the total amount allocated for the project. The next thing that needs to be determined is if you are anticipating bringing on additional staff, reassigning current staff, or if the complete project can be outsourced to an outside vendor. One of the biggest costs of a software purchase is the additional staff it takes to own that software in the long run. To use one common example, a company that purchases a CRM might look at the cost of the initial purchase and see it as \$75,000 per year. But they are most likely also going to need to hire an in-house admin for that CRM, which balloons the annual cost of the CRM to more than double the price if you include the salary plus fringe. So, if a different software vendor is offering to have an outsourced admin with a strong service level agreement for \$125,000 versus a different vendor that is selling a solution that requires internal management, make sure you are fully understanding the cost of taking that in-house.

Conclusion

By properly understanding the requirements your business has for a software solution, understanding who and how the software will be managed, and understanding the costs and anticipated benefits from a software solution, you can ensure that you will be satisfied with your purchase now and in the years to come.