

A Guide to Custom Software Development



Table of Contents

<u>Introduction</u>	2
<u>Chapter 1 Time for New Software</u>	3
<u>Chapter 2 Custom vs. Boxed Software</u>	5
<u>Reasons to choose a custom solution</u>	6
<u>Reasons to choose an off-the-shelf solution</u>	7
<u>Chapter 3 Custom Software Benefits</u>	8
<u>Chapter 4 Developer Options</u>	11
<u>The benefits of outsourcing</u>	12
<u>The benefits of an in-house team</u>	13
<u>The challenges of outsourcing</u>	14
<u>The challenges of an in-house team</u>	15
<u>Chapter 5 Custom Software Risks</u>	16
<u>Chapter 6 Risk Mitigation Strategies</u>	18
<u>Conclusion</u>	20

Introduction

In today's business environment, the role of software continues to grow as companies look to establish enduring success. A software system that is misaligned to your business needs can create headaches and – even worse – missed opportunities. Throughout your search for the right software solution you'll encounter a number of industry buzzwords, different types of software, varying architectural styles, developers touting a wide range of skillsets, and many other topics native to any discussion about software solutions.

A Guide to Custom Software Development aims to make some sense of this glut of information, specifically that pertaining to custom software. What is custom software, exactly? When is a custom solution the best software option for a business? What are the risks and benefits associated with custom software? This e-book provides answers to these and other questions pertaining to custom software in an effort to supply useful and accurate information to assist you during your software decision.

Key Takeaways

1. *Determine if it's an appropriate time to look for a new software solution*
2. *Know when a custom or boxed solution best aligns with your scenario*
3. *Identify 10 benefits that custom software can provide*
4. *Weigh the challenges and benefits of outsourcing vs. an in-house team*
5. *Know what risks are common when developing custom software*
6. *Employ strategies to reduce custom software risk in your project*

Chapter 1 Time for New Software

As your business environment changes, so too do your software needs. If one or more of the following descriptions can be said about your current software system, it may be time to start thinking about a new solution.



Not mobile capable

With today's "always on" culture, if your business is not mobile capable then you'll quickly turn off your target audience. Mobile capability should not be an afterthought, as mobile-designed software affords more broad and instant access. By thinking mobile first with your solution - whether via download of a [native app](#) on your device or via [web application](#) targeting all internet browsers - your software can provide greater flexibility in reaching your target audience.

Customers can now instantly engage with your business directly from their devices, while employees can easily utilize the system and complete tasks without having to be tethered to their desks. In going mobile, both you and your customers can do business anytime, anywhere.

Lacks integration

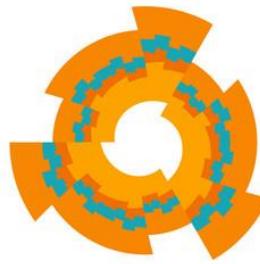
While mobile functionality boosts productivity and efficiency, having all of your core software elements working independently of one another is an inefficient use of resources. When these separate elements of your system require data entry, detailed calculations, or other essential processes to be duplicated, employee productivity is sacrificed along with the business's bottom line.

System integration enables each piece to work together harmoniously, streamlining the code to eliminate redundancies. An integrated system enables users to interact with the various elements cohesively and intuitively, creating a more efficient workflow. Additionally, maintenance costs - both short- and long-term - are reduced, allowing more time to be spent with the software in completing tasks rather than finding and fixing bugs and other issues.

Inefficient legacy system

Integration may help reduce maintenance costs for your system, but legacy software may require more than simple maintenance. A [legacy system](#) is the software backbone that enables a business to perform many of its core processes. It's common for legacy systems to be utilized for years without any significant advancement, rendering the system sub-optimal. However, due to the key role the system plays within the company, sunsetting the software may not be an option.

When legacy software is performing at a sub-optimal level, it can act more toward hindering success of your business rather than advancing it. A custom [legacy software facelift](#) and other modernization techniques can retain the core attributes of your legacy system while helping the software meet or exceed the standards of today's tools and solutions.



Complicated user experience

One such standard that should be met with your new solution is ease of use. Software designed without the user in mind is bad software. A poor user interface (UI) and/or user experience (UX) can frustrate your audience and make them abandon the software entirely.

An inadequately designed UI can pose a problem both internally and externally: your employees may find it difficult to complete their tasks, decreasing their output and efficacy; customers may have trouble navigating the software causing them to abandon their willingness to make a purchase or otherwise spend their money with your business.

A troublesome [UI impacts the UX](#) too, as it creates a burdensome and vexing experience for users. If these users happen to be customers, it's a very real possibility they will turn their attention elsewhere to find a more suitable solution to satisfy their needs.

Misaligned with current needs

The current stage of your business can impact the user experience associated with your software; in earlier stages, the number of users may not justify upgrading to a more streamlined feel. But as your company experiences success and enters a growth stage, your business needs are sure to shift, impacting the software that you'll utilize. It's common for startups to opt for a relatively simple and low-cost solution to satisfy their software requirements as they strive to become an established and known entity.

But as startup transitions to mature business, it's a near certainty that today's requirements differ drastically from when the company was first breaking in; perhaps an expanded feature set or greater scalability is necessary. There are different ways to attain these new requirements, but whether you opt for a ready-made or custom solution, an upgrade to your software system will help you keep pace with the changing market dynamics of your industry.

Compliance concerns

Regardless of the current stage your business is in, it is paramount to continue to follow industry rules and regulations. Every industry has regulations and laws in effect that must be adhered to in order to avoid significant penalties - it is imperative for your business to be compliant with any that can adversely affect the company.

Whether these regulations cover confidentiality, accurate reporting, cyber security, or some other topic pertinent to your business environment it's important to understand how they can impact the continued success of your organization.

Noncompliance is easy to fall into, but building safeguards into your software helps protect against future compliance missteps. If the software's current adherence to existing regulations is lacking or if it is at risk for those edicts on the horizon, it is critical for your software to fall in line.

Chapter 2 Custom vs. Boxed Software

Managers in today's companies are keenly aware that software is vital to ensuring smooth operations and competitive success. The software that handles business operations typically falls into two general categories: custom software and off-the-shelf software.



***Custom software** is written and tailored for the specific needs of a single business or family of businesses.*



Off-the-shelf (a.k.a. boxed, canned, or packaged) software is ready-made and often specialized for a defined industry vertical.

Reasons to choose a custom solution

You have unique requirements

Because boxed software is designed for a broad audience, it cannot realistically meet all of the specific needs of every business. Its functionality may not line up with your unique environment, potentially draining your business of efficiency or imposing undue costs and overhead on your workflow. Custom software addresses this challenge, as it's tailored to meet your particular functional objectives.



Flexibility is key

Similarly, the rigidity of boxed software may prove problematic in how it overlays business operations. There are a set number of features that can be toggled on and off, leaving you with more features than you'll ever truly use or not nearly enough to fulfill your fundamental needs. As demand, customer characteristics, and competitors change, the software may or may not adapt. Custom software, however, can be readily extended and modified in a dynamic landscape.



Specialized integration is required

One of the ways custom software can be modified in a dynamic landscape is with integration. [Integration](#) is the key to speed in corporate information systems. Not only do the in-house systems need to adequately talk to each other, but we're often tasked with connecting to customer or supplier systems to enable favored trade partner status. Commercial off-the-shelf software may not envision the need for such interconnection, or even be compatible with those other systems you have in place or are looking to add. This scenario adds complexity and overhead, ultimately requiring more work.



An essential legacy system is in place

Like integrating with multiple systems, a legacy software facelift helps your software run more efficiently. Old, inefficient software is often kept around because its processes are essential to business functions. As time goes by, attempts at modernizing infrastructure are made more complicated by legacy systems that can't be eliminated. Custom software can keep this integral, core functionality intact while applying a modernized facelift or a means to communicate with other modern systems, helping keep up with today's demands of the business.



You have a modest budget

Just as there are off-the-shelf solutions for nearly every industry, there are off-the-shelf software systems that appeal to almost every budget. Not all are inexpensive – some can range in cost from a few thousand to millions of dollars. However, a wide variety of ready-made software products do provide great value on a small budget, with a primary benefit coming in the form of reduced upfront costs. Such software is sure to satisfy the common needs of most consumers in that particular vertical.



Your competitive advantage lies elsewhere

Because prepackaged software is designed to satisfy a broad range of consumers, the product typically doesn't convey a competitive advantage for businesses. If your competitive advantage lies outside of the realm of the software – providing low cost, delicious food at a restaurant or demonstrating quality craftsmanship in residential construction, for example – then nearly any canned software would suffice. There's no need to overthink investing in software for processes that provide little or no strategic benefit. In these instances, it makes sense to have the software be a complimentary piece while the other elements of your business do the heavy lifting.



There is an immediate need

Additionally, because it is designed as a commodity for a wide range of users, off-the-shelf software can often allow near-immediate deployment and use. You don't have to face down time while the solution is being built; the process is usually as simple as: purchase, install, go. While some functionality may be sacrificed by going with an instant fix, such packaged offerings can get you up and running right away while affording some time to determine a proper long-term solution that aligns with your anticipated requirements. A boxed solution can allow you to stay in the race rather than get completely left behind.



A sufficient solution already exists

Boxed software often is highly valuable; because many industries have dedicated software for a wide audience, it's possible and even likely that most or all of your needs can be covered. Countless companies exclusively use off-the shelf software and see their businesses thrive. If yours is one such business and there is a packaged software solution already available that sufficiently meets your functionality specifications, there is little reason to spend time and money reinventing the wheel with a custom solution. Keep this simple.

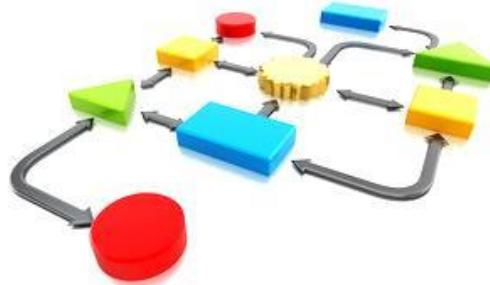


Chapter 3 Custom Software Benefits

While boxed software is a perfectly viable solution for some business settings, companies often find themselves wishing for more from their technology. Custom software provides a number of benefits to these firms.

Control development

With custom software, you can be involved in the process from start to finish, ensuring the end result aligns with your specifications. Track important milestones and provide input if changes or additions need to be implemented. You know your business better than anyone; you need software that works to your priorities. With a custom solution, you control development of the software to make sure it does just that.

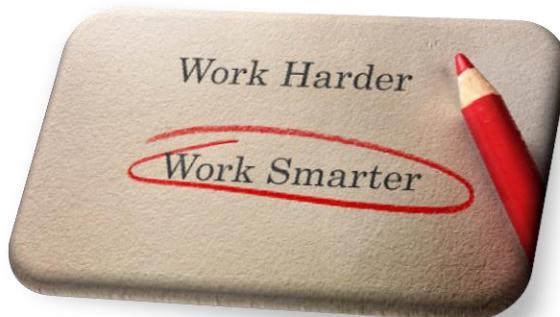


Include only necessary features

By taking control of development and being involved in the process, you don't have to worry about being burdened with unnecessary features that add complexity and cost. Decide what your software needs to do, and it can be designed to match that unique scenario. Gather input from stakeholders then demonstrate that you were listening by giving them a better experience by way of better software. Custom software eliminates needless complexity and non-intuitive distractions to include only the functionality you deem proper.

Boost productivity and efficiency

Through a focus on only the necessary features, you can ensure your custom solution is designed to match the way people work. With an intuitive user experience and simplified data entry, confusion and guesswork are removed as impediments to task completion. By streamlining and automating the various steps in a complex workflow, processes that previously took days to complete can be [reduced by as much as 50% or more](#), and the issues arising from human error can be drastically reduced.





Tie costs directly to value

With increased efficiency comes greater value. A custom software solution ties costs more directly to that value. To begin, there is no ongoing cost of seat licenses or maintenance, both of which are commonly found in boxed software products. And because a custom solution is tailored specifically to your business processes, redundancies and bottlenecks can be eliminated which ultimately saves time, money, and productivity that would otherwise be lost with the use of a less ideal system. While its upfront cost may equal or exceed a quality packaged system, a custom software solution helps lower costs in the long run.

Own the IP

As alluded to, off-the-shelf software products typically work on a licensing basis: you purchase permission to use the software but you do not own it. By contrast, it is standard practice for a custom software developer to build your solution while you retain full controllable rights to the code. The custom solution becomes an addition to your intellectual property portfolio and boosts your balance sheet. An extra benefit: you now have the opportunity to explore additional creative options such as marketing your custom software to other industry players facing the very same challenges for which you have developed a unique solution. Owning the code of the solution enables you to be more flexible in your use of the software than you otherwise might be with a boxed product.

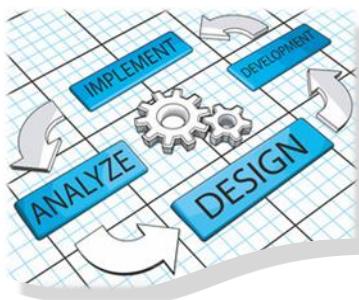


Gain a competitive advantage

Along with increased creative flexibility, a custom software solution can deliver an edge for your business by separating your organization's offerings from others within the industry. Consider the possibility of providing your customers a dedicated portal with a [professional look and feel](#) and self-serve access to data, as well as other services not available through competitors. Or turn your focus to inside your organization by providing a continuing education platform to facilitate both training of new employees and sharing industry knowledge among more experienced personnel. There are countless ways custom software can be leveraged to gain competitive advantage.

Scale the software with your business

While custom software affords greater opportunity for garnering competitive advantage, a small business generally won't have difficulty finding inexpensive software that can perform admirably. However, as the business grows it is confronted with meeting the demands of an ever changing market dynamic. Boxed software for large businesses may be overly broad and expensive, and for a small business, a non-starter. A custom software solution ensures your business doesn't get stuck in the middle; as your company and its marketplaces evolve, so too can the software, ensuring you have the best match to meet the challenge.



React more quickly

Typically with software projects - as with most any project – these challenges arise in the form of unforeseen needs and adjustments. As your business grows or pivots, needs change and the ability to adapt your technology to these demands is paramount. When such a change or improvement is necessary, custom software – by design – allows for expedited alterations. You're no longer at the mercy of the packaged software developer, hoping that their roadmap and timeline for updates align with your needs and future plans. A custom software solution lets changes and tweaks be made on your schedule.

Avoid obsolescence

Being tethered to the update and maintenance schedule of the boxed software's developer can have severe ramifications. Prepackaged software runs the risk of becoming outdated – or worse, sunsetted – due to factors that are beyond your control. When this occurs, support begins to degrade and future software updates become fewer and further between, leaving companies scrambling to address the new risks imposed upon them. By way of iterative improvement or a legacy facelift, custom software can ensure that core essential functionality is retained while new standards and trends are introduced. Functionality can always be added to help your business keep pace with the changing demands of the industry.



Receive better support

Because a custom solution is tailored to fit your environment and specifications – whether through a legacy facelift or from the ground up – a more in depth analysis of your business is necessary, affording the developer a better understanding of your business functions and requirements. This fosters collaboration. If an issue arises, you have a partner to help think holistically through to the solution. Any items that crop up are easier to recognize and intervene, thanks to the developer's familiarity with the expanded environment. Should opportunities present themselves, you're able to address and capitalize on them much faster. The developer's familiarity with your business will play a key role in your success.

Chapter 4 Developer Options

When it comes to building a custom solution, there are two primary routes: outsourcing the work or keeping development in-house. Each alternative offers its own benefits, but can also bring about significant hurdles.

Outsourcing

While it is generally associated with moving jobs out of one country and creating them in another, the concept of outsourcing is not limited to this practice, which is more accurately called off-shoring. In fact, local or onshore outsourcing is a popular method for businesses seeking to have certain processes or business functions completed externally. In the custom software industry, both off- and onshore developers are viable alternatives when pursuing a custom solution. Although reduced costs are primarily associated with outsourcing, there are additional benefits with which external custom software developers can provide your business. Despite a number of benefits, outsourcing your software development does not come without risk. On the contrary; there are some challenges that should be considered before determining if outsourcing is right for your scenario.

In-house

Opting for an in-house development team is the route to take if your business wishes to internally develop the solutions to your software needs. As a branch of the company, the development team will be fully under your management and direction. While maintaining a team of in-house software developers certainly brings benefits, there are significant challenges that can arise, much like with outsourcing development to an external firm. Review the benefits and challenges of putting together an in-house development staff to determine if it is the right path to contribute to the success of your business.



Reduced staffing and time costs

In today's business realm, outsourcing is nearly synonymous with cost reduction. By moving work outside of the country, companies can realize considerable savings – primarily in employee costs and benefits – because many foreign countries do not face the same laws and payment requirements seen in the United States. Outsourcing locally also helps reduce cost; many firms have staff levels that allow for multiple developers to work on a project in quick succession, reducing the overall length of the project from concept to final delivery compared to the time a small in-house team can complete the work.

Short-term commitment

The time being tethered to the developer is similarly shortened. Upon completion of an outsourced project, the developing firm's services are no longer necessary. While the firm may continue to provide services in the form of maintenance, the cost associated with such action is minimal compared to the potential inefficiency of retaining in-house developers. Conversely, with an in-house team, a project's conclusion often sees the retention of the staff – whether or not the employees' skills are aligned with current business needs – due to the high level of time and effort initially undertaken to build and train the in-house team.

Focus on core business

Shifting your custom solution development to a software development firm that is external to your company helps you focus on your core skill: running your business. No longer will you have to be the project manager for a technology with which you may not be at all familiar. The development firm can absorb this role, allowing you to oversee marketing, engage customers, or drum up sales – areas you know best. With an in-house team, particularly a small one, it is necessary for you to play the role of project manager during development.

Developers with wide-ranging skill sets

As finding the right employees with the desired skill-set for in-house teams takes considerable time and energy, the resulting teams are often small. Because of this small size, it is not uncommon to focus on one technology at the expense of many others. Outsourcing enables you to find developers who possess a vast library of knowledge and are trained in a number of technologies, ensuring coverage of a broad spectrum of potential client compatibility and requirements. You can then select those developers that are most appropriately aligned toward providing a solution to your business needs.

The benefits of outsourcing



The benefits of an in- house team

Employee commitment and responsibility

Employees want to know that their efforts are having an impact for the company. With an in-house team performing your software development duties, you'll likely see high levels of commitment from your employees because they will take pride in building something that can both showcase their skills and further the success of the business. Often when development is outsourced overseas, however, this commitment and responsibility to the company is lacking, leading to merely average expectations and results. Because these developers are not a part of your company, their vested interest in both the project and your company's overall outlook may be substantially lower than that of an in-house employee.

Complete business knowledge

As "in-house" implies, such a development team is indeed a part of your company. Because of their position within the company, these employees are generally exposed to the inner workings of the business. They can recommend improvements for a better work environment or suggest changes that may lead to greater market competitiveness and performance. An in-house team contributes to the company's culture and, through their work, represents the business. It's difficult to find an outsourced software developer that has a similar impact. An external firm may go to great lengths to understand your business during project development in an effort to provide the best custom solution, but completion of the project generally brings such overhead to an end.

Confidence in quality and security

While outsourced developers typically lack the internal knowledge of your business, overseas developers, specifically, may be short on practical talent as well. Many such developers attain only a theoretical knowledge base in a number of tools and technologies, lacking the skills for building solutions with these technologies in the real world and potentially exposing their clients to devastating quality issues. Additionally, many foreign laws don't afford the level of software and intellectual property protection that can be found in the United States, making theft or infringement of the code a real concern. In the United States, with in-house developers you can feel secure in the protection of your code thanks to the laws and penalties in place to safeguard such property. Likewise, you can ensure the quality of your solution by seeking and hiring staff that have had experience applying their knowledge and skills in a true business setting.

Logistics

When software development is outsourced, logistics become complicated and can prove to be an issue. Overseas developers come with the added complication of working in a completely different country and time zone. Phone calls and virtual meetings may be difficult and inconvenient for at least one party as these types of communication will often need to be conducted during traditional business off hours. Onshore developers may require substantial travel distances that consume time and other resources, or face smaller time zone differences that can still prove to be a challenge. Coordination between an outsourced developer and their client can be a difficult hurdle to overcome.

Quality of foreign developers

If distance doesn't present a problem, quality might. While they may be trained in a variety of technologies, many overseas developers lack the practical experience of applying their software acumen in a true setting. Because of this, the quality of the software solution may not be up to par. It is not uncommon for companies to outsource to an offshore software development firm only to require extensive reworking of the custom solution once the final deliverable has been received. This necessary reworking effectively negates any cost savings that may have been realized through outsourcing in addition to causing a delay in the deployment of the new software. Such a stall in utilization further erodes any savings or industry advantage that may have been afforded with an on-time project.

Cultural differences

Outsourcing the development of your software project inevitably leads to interactions between different company cultures. Offshore developers, due to limited real world experience by coders, may not achieve as high of a quality standard, leading to a breakdown in the relationship and a disappointed customer – you. Outsourcing to local companies can also result in culture clashes. Not all companies value the importance of communication, while others lack the humility to admit they don't understand a business requirement which causes them to waste time and dollars going down the wrong path. It's important to be aware of cultural differences between your business and the developer, as they can easily sink a project.



The challenges of outsourcing



The challenges of an in-house team

Cost of staff

An in-house development team can be a substantial cost for your business to maintain. Developers are in high demand, and to hire and retain these employees means paying significant salaries to ensure you employee the best in their craft. In addition to a competitive salary, many employees seek extra benefits – healthcare coverage, a 401(k) plan, personal days, paid vacation – that cause their overall compensation to increase. Such costs increase as your in-house team grows, while no such cost to you is accumulated when you opt for an outsourced software developer.

Time investments

Some companies are willing to pay high financial costs associated with retaining internal developers because finding the right candidates with whom to assemble your in-house team is not a task that is completed overnight. Seeking out these candidates takes considerable time and effort; while you work to find employees that will be a fit for both your immediate and future software needs, you are losing valuable time that could be better spent actually building the new technology. Outsourcing your custom software development allows you to engage a team that is currently staffed with professionals experienced and skilled in the technologies your require, enabling the project to commence in a fraction of the time it takes to get underway while trying to build an in-house team.

Staffing issues

Due to the prolonged timeframe of finding the right employees, new hires may be misaligned with current IT demands; business requirements fluctuate from year to year, creating peaks and valleys in staffing needs. Taking on new employees simply to meet the business demands of the current quarter may result in being over-staffed in the next. Paring this abundance back when requirements are low all too often sees yet another spike in IT demands in short order. This instability in business needs and the resultant staffing levels is a common problem businesses face, causing either a waste of resources or missed business opportunities. By outsourcing the software development for your custom solution, you can [balance IT staffing levels](#) with your current business need. As your needs change, it's easy to ramp up or scale back your software developer workforce.

Chapter 5 Custom Software Risks

Risk is inherent to every project, and custom software is no exception. The failure of a custom endeavor can generally be attributed to a few common factors that, with careful attention, can be avoided altogether.

Overly ambitious

When an idea for a new product comes together, it's common to try to expand upon it in an effort to build a grand solution that can solve many problems and satisfy multiple needs. However, such methodology can lead your new custom software solution down a rocky path. Bloated feature sets and capability requirements for the new software contribute to extended timelines and budgets, often without even knowing the true value of these add-ons to the end user. It's wise to begin development on your new software with an MVP – [minimum viable product](#) – in order to keep initial costs low while identifying the best direction for your software. Utilizing an MVP as a foundation of your new software strategy allows you to be more adaptable while safeguarding against the risks of being overly ambitious in the early stages of development.

Lack of purpose

Software is undoubtedly essential for certain business functions to run smoothly, but a custom project can be troublesome if its purpose is not clearly defined. Frequently, companies get lost in the limitless buffet of nice-to-haves that are easily added to their custom solution. Without a precise value proposition, the solution's development can spiral out of control. Providing honest and concrete answers to questions like those in the sidebar will help ensure the custom solution in which you are about to invest considerable resources truly does have a focused purpose. By continually revisiting these questions throughout development, you can help keep your project on track and positioned for success.

Poor communication

Whether you start your software project with an MVP or an expanded feature set, frequent and clear communication is critical when developing a custom solution, as even a minor breakdown can put the project at a severe risk of failure. Throughout the software's [development process](#), care should be taken to ensure you completely identify the requirements and that they are fully understood by the development team. While it's not unusual for new features and functionality to be added in later stages of development, it is necessary to share these new specifications thoroughly and expediently with the developers so that the production timeline and forecast can be altered as needed. Similarly, the developers should be available throughout the project's life cycle to answer any questions and provide you with updates and testable software. Strong two-way communication helps ensure the project is on the right track.

Determine A Purpose

Before and during development, questions like the following can help you determine a purpose for your custom software solution:

- *What problem does this custom software help solve?*
- *At what point will the problem be considered solved?*
- *What value does the solution provide – efficiency, cost savings, productivity?*
- *How will this software provide benefits both today and in the future?*

Quality Developers

By asking pointed questions to and about a software developer, you can get an educated idea of their development practices:

- *Does the development team use an agile approach to software development?*
- *Do the developers test thoroughly throughout the process?*
- *Does the firm put forth an honest effort to better understand the business and industry?*
- *Is the client kept in the loop throughout the stages of development?*

Inaccurate estimates

Similarly, poor internal communication can lead to a failure to properly determine your company's budget for a custom software solution or a severe misidentification of what the software needs to do. To safeguard against this, again make communication and transparency a priority. Be sure all stakeholders - whether they are customers or employees - are accounted for in factoring in system requirements to help lessen the likelihood of significant surprises or roadblocks occurring and impacting project development. Frequent and thorough communication between your organization and the developer leads to a more complete understanding of budget constraints, requirements, and timelines. From the developer's perspective, this more in-depth understanding enables them to provide an accurate project estimate and give a clearer view of the overall roadmap to the solution's completion.

Substandard development practices

Constant communication doesn't guarantee a quality developer, though. Just as there are [varied methods](#) to actually having your custom software solution developed, the developers of the software can differ widely in how they perform. Not every development firm follows the best practices, and in many cases this can severely hinder the success of your new custom solution. If the company doesn't have an in-depth engagement with you in an effort to understand your business, is often unavailable for support and assistance, or doesn't follow and utilize industry standards then your software is likely to suffer. Be sure to thoroughly research the developer prior to selecting them as the project's development team.

Failure to implement

Once you've found a great developer to bring your ideas to life, it's up to you to make sure the new software is put to use. Imagine paying for something you're never going to use – this scenario plays out more often than it should with new software projects. New apps are built and then abandoned without promotion or deployment. Processes are looked at but never changed, leaving employees struggling to boost productivity. The software's purpose is not fulfilled simply because the solution was built, and considerable effort needs to be taken to ensure employees understand and adopt the new software. Employee interest and support for change can diminish if they are not engaged or their input is not considered. Customers need to be aware of – and actually have a reason to use – the new solution; make them know their life is better with it than without. Without great implementation, you've failed to close the loop and the software project is a wasted endeavor.

Chapter 6 Risk Mitigation Strategies

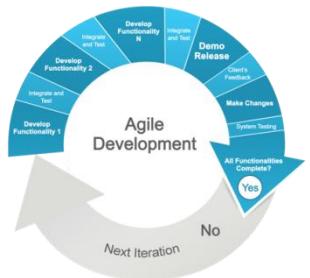
The risks brought upon by custom software have given rise to techniques and strategies that strive to mitigate the likelihood of a negative outcome. The use of such methodologies helps pave the path toward the long-term success of your custom system.

Start with an MVP



The fundamental premise behind custom software solutions is the ability to go anywhere and do anything. But you need to be strategic in managing that freedom. While mindful of utopia, it's important to get there in stages. The first stage should strive toward articulating the minimum viable product (MVP). An MVP enables the first iteration of your system to be attained with the lowest possible budget of time and financial expense while still providing adequate functionality and value to stakeholders through a pared back requirements list. Ideally, the analysis that leads to the system's definition should include all features envisioned, with those identified for inclusion for MVP to be noted. Those capabilities not earmarked for MVP must be agreed to by the stakeholder affected. This way that individual provides buy-in to the MVP and is assured their "pet" will not be forgotten, but incorporated at the proper time and circumstances in the future.

Be agile



Sometimes, this proper point in the future occurs in the middle of development. Adding more functionality mid-build or tweaking the direction of the project are common and reasonable adjustments that arise during a custom software project's development cycle. With an agile development methodology, a series of iterations are used in building the custom software solution, allowing for new pieces to be added or reworked with each iteration. Agile development - especially when paired with an MVP strategy - allows for significant flexibility for both your business and the software developer. As the demands of your industry change, you can react quickly and efficiently to incorporate new features and functionality into your custom solution for further competitive differentiation.

Address technical risks early

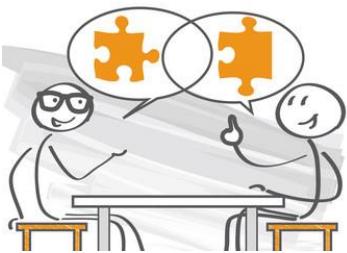


Custom software solutions, as a way to boost that competitive differentiation for the business, often require the application of new tools, techniques, and technologies. When these technologies play a role in your software project, a quality development team will take the time to research and understand each one, so that it may be seamlessly integrated into the solution. Understanding the strengths and vulnerabilities inherent in the various tools ensures a path toward getting the most out of a future-proof system. Ignoring potential technical risks can give rise to bottlenecks or blockers downstream in the development cycle. It's critical to consider the possible consequences that might be faced down the line while you assess the best path forward.



Communicate frequently

Lack of communication is often cited as a frequent cause for why a custom software project fails to deliver on expectations. As the team tasked with building your custom software solution, the development firm should be in frequent contact throughout every phase of the development process. They must fully understand your needs and requirements for the software in addition to providing feedback and updates regarding scope, timeline, and budgetary concerns. Clear communication for the duration of the endeavor's development cycle helps ensure all parties are on the same page, each forward step is methodical, and risk of any unwanted surprise is diminished.



Engage users

Engaging users is a critical element of the frequent communication philosophy. In order for the software to perform at its best, the people who depend on it need to be tapped for input. These conversations are as much about the ideal workflow as they are about software requirements. Very few employees and system users have any experience in driving software requirements discussions, so you need a development firm that's experienced enough to collaborate with users and constructively gain their input. But it doesn't end there; users need to be exposed to the work in progress at certain milestones to provide input and feedback that ensures the product is on course and on-value. The feedback from these engagements steers the direction of the subsequent iterations, continuing until the project has been completed. By engaging users, the development firm is certain to deliver a custom solution that's useful and beneficial to the target user audience.



functional testing

Test thoroughly

By involving users, developers can identify just how broken something needs to be in order to really be considered broken. All software should be tested early and often before it becomes too broken and too expensive to get back on track. As various elements and iterations of the custom solution are built, testing helps identify whether the software is performing as it should and if all of the pieces are interacting cohesively. With a frequent testing schedule, developers are able to quickly find and correct issues that would have crippled the entire project had they waited until a later point in the development cycle to run diagnostics. Such testing methods ensure the software is healthy from start to finish.



Maintain knowledge and expertise with support agreements

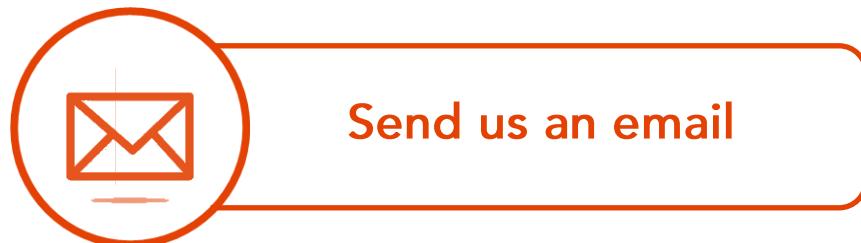
Once the custom solution completes its in-depth testing and ultimately has been delivered, the work doesn't necessarily stop for a software developer. The software needs to be future-proof. Ongoing support and enhancement agreements provide for updates, upgrades, and overall responsiveness to the changing business conditions of customers and marketplaces. Such agreements enable quick continued adaptation in response to changing technology considerations, both opportunity and threat. You need assurance that your development team will be ready when you're ready. Retaining detailed and accurate documentation of the project's specifications provides you with continuity to respond to any scenario requiring change or enhancement to your custom software system. Support agreements prevent lost sleep, wondering if you'll be ready to respond to change.

Conclusion

Organized to provide targeted insight about a range of topics associated with custom software, the six chapters of this book were developed to help give you an edge when making software decisions. The key takeaways highlighted at the beginning of this e-book, *A Guide to Custom Software Development*, have provided a solid foundation of knowledge for you to have the skills and confidence to assess your current software scenario and make a decision that is right for your business. Here, again, are those key takeaways:

1. *Determine if it's an appropriate time to look for a new software solution*
2. *Know when a custom or boxed solution best aligns with your scenario*
3. *Identify 10 benefits that custom software can provide*
4. *Weigh the challenges and benefits of outsourcing vs. an in-house team*
5. *Know what risks are common when developing custom software*
6. *Employ strategies to reduce custom software risk in your project*

Have questions that weren't answered? Eager to get started on your custom solution? Contact Gavant Software today and a solution consultant will be happy to answer any inquiries you may have, from strategy to measurable results.



A Guide to Custom Software Development

©2017

Developed and assembled by Gavant Software, Inc.

216 River Street

Troy, NY 12180

(518) 273 - 2880