Everything you need to know about custom software

BEFORE DIVING IN

bitbean

1 Introduction

Are you serious about custom software for your business?

Or are you just starting to wonder if it's something to consider for your operations? Either way, you've come to the right source. This eBook will give you the information you need to help you make the right decisions about what you need to improve your business by exploring the various software options. Ultimately, you'll find the reasons why you may need custom software (or not). You'll discover how to accomplish your objectives through software and how to justify it.

We cover what software is, what it does, and how you can make it work for you. In addition, this guide will look at the different types of custom software uses and how each of these can benefit your business. You'll learn how these options will meet your goals in the short and long term.

If you have a new product in mind that you want to develop for your business, you can explore the advantages and disadvantages of using custom software to build it. We also provide some tips to help you start building your project including some of the basics. You'll review off-the- shelf considerations versus custom software development to make a better decision on your product needs. In addition, we include how to build around existing platforms and incorporate existing options like PaaS, SaaS, CRMs, and others.

Do you have an IT team?

We’ll give you some ideas to better communicate your objectives and get them aligned to your goals. And if you don’t already have an IT team, we’ll look at whether you should consider one. We also discuss why you should think about putting a Product Manager in place with increased responsibilities for greater success.

Because you have lots of choices available regarding new technologies, we review some of the most promising. By looking at these options, you’ll learn how they may best meet your needs including the various processes, design methods, and latest tools.

Finally, we discuss whether you should consider outsourcing your project and how to weigh the costs versus the benefits. You’ll also get advice on how to select a software development company. Additionally, you’ll learn what you need to prepare for a more rewarding consultation.

We hope this guide will help you become an expert in custom software development as well as become more knowledgeable about your own business and software needs. If you feel confident that you have the information you need to decide what options are best for you, we know you will be better prepared to make the right choices for your business.

2

What is custom software?

> Anything one man can imagine, other men can make real.
>
> - JULES VERNE

Software defined

Before we tackle the definition of custom software, we need to establish what software development is, what it does, and how it does it. To some, the word software itself may seem elusive.

Hardware, on the other hand, is a concrete term, easily defined. For example, hardware refers to the physical components or tangible aspects of technology, particularly a computing device. This includes computers, motherboards, wires, fans, monitors, printers, etc. Whereas, software relates to the written instructions that tell the hardware (e.g. computers) what to do. Software includes operating systems, application programs, related data, utilities, games, etc.

Software relates to the written instructions that tell the hardware (i.e. computers) what to do

That said, the focus of this eBook is on software processes and development as opposed to products. However, according to the article Software Process Definition: A Reuse-based Approach, in the Journal of Universal Computer Science, many researchers argue that software processes are also products. The fact is that software development has for decades taken advantage of "reuse techniques" including "software components, architectures, and product lines have been successfully applied in several contexts to develop software products…”

So, defining software processes may be difficult. As the article confirms, “Software process definition is not a simple task; it demands experience and involves knowledge from several aspects of software engineering.” The techniques that are used by engineers and end results of their efforts are key. The knowledge used to develop/reuse software techniques in products, for example, can also be applied to software processes.

Characteristics of custom software

Software differs from physical goods as it has no physical limitation. According to, Defining Software Ecosystems: A Survey of Software Platforms and Business Network Governance, “the main limitations are conceptual, social and economical.”

When deciding on custom software, there are various factors to consider including the “needs and characteristics of the organization or project, techniques and methods to use, adherence to standards and reference models, business constraints (schedule, costs, etc.), among others.” With this in mind, it becomes clear, that as a business leader, your participation in deciding these factors is crucial. Therefore, as you may suspect, these considerations require the skills of a professional who can make sense of all these factors and may require help from “experienced consultants.”

What are the variations?

But the best way to define software is that it solves problems. In the book, Software Engineering: Theory and Practice, software development requires analysis and synthesis. In other words, it’s the breaking down of a big, complex problem into smaller pieces (analysis) and building or composing software from “smaller building blocks” (synthesis). So, what are the various types of software?

1 Formal procedure

Software can be a method or “formal procedure” to accomplish a goal. For example, document management software is written to organize & manage files.

2 Tool or instrument

Software can also be a tool or instrument design to do something better as in automation. These tools or programs are also used to create, maintain, support, and debug other programs and applications.

3 Combined tools & techniques

Many times, software represents a combination of “tools and techniques to create a product.” An example of this may be project management software, which integrates online systems to work and collaborate on projects.

4 Product build

Finally, software is an approach to build a product. For example, the agile approach is widely used for its adaptability and speed to develop web-based software development and applications due to regular modifications.

The bottom line is that software processes help to solve problems and improve all facets of business. Knowing what needs improvement in your business can be the starting point to decide what custom software options you should consider:

1 What can your business do better?

2 How can you increase productivity?

3 How can custom software help you grow?

Did you know?

Software can also be categorized by its function as it relates to performing specific tasks. For example, productivity software can include spreadsheets or word processors. There is also graphic software (CAD/CAM), and presentation software, among others. In addition, industry-specific software is written and used in banking, real estate, retail, and other industries.

3

Does my company really need custom software?

“

> Truly great custom software has the ability to shape the future.
>
> - EPHRAIM ARNSTEIN

Your business goals

Your business goals are set in motion by important choices that dictate the path and vehicle you'll need to reach your objectives. And although your goals may seem obvious (profitability, productivity, growth, etc.), knowing the best way possible to reach them may be more challenging. The rapid advances in technology don't make these decisions any easier. But what is clear, is that whatever path you do choose, the need for flexibility and adaptability are necessary ingredients to see you through both now and into the future.

Custom software adjusts to the growth of your business while meeting objectives along the way

It is specifically designed to align to your business' uniqueness, which is what makes you stand out from the competition in the first place. So, if how you run your business is what makes you competitive, here are some strong arguments in favor of choosing custom software to hit your targets.

Benefits of custom software

Understanding how custom software works within the 'big picture' of your organization can help you determine its true value.

Custom software is driven by your business' specific needs. It is developed to deliver results by doing exactly what you want it to do and by solving problems inherent in your business model. In some cases, not implementing custom software can cause additional problems for your business.

Accuracy and efficiency are critical for all businesses to avoid errors, maintain regulations, etc. For example, healthcare must maintain data accuracy. In that industry, the issues and workflows are unique particularly within clinical research. According to Clinical Research Information Systems, "despite the proliferation of commercial software, special problems still arise that only custom software development can solve ..."

Just what you need

You won't be forced to take on added features/functions that you have no use for, making it user-friendly for employees and customers.

Meets your business goals

Customized software is built specifically for your business and is industry-specific so you won't find yourself settling for something that's not quite what you're looking for. And because many businesses are bound by regulations, custom software addresses these needs without sacrificing performance or productivity.

Works with you

Custom software works with your business processes and existing systems, no matter how unique they may be.

No updates necessary

You won't have to pay for frequent updates that don't add anything of value and can't be used on your platform.

Easy + simple configuration

Because it's built for your business, you will save time and money not having to deal with a complicated configuration.

Bridges other systems

Designed to automate some processes and make others more efficient, custom software ensures successful results.

More secure

Necessary security measures are built- in from the onset of your project for the projection and safety of your systems, data and content.

More scalable

Custom software can be expanded and modified so it has the ability to grow with your business.

Longer lifespan

Tends to have a longer lifespan than off- the-shelf counterparts and yields towards cost savings in the long run.

Current processes

With custom software, you won't have to change how you currently do business to suit a ready-made software solution.

Instead, you'll improve your current processes and enhance productivity. Off-the-shelf software products can limit your preferred features, reduce flexibility, and make integration with other systems difficult. Custom software will accommodate your processes regardless of any uniqueness or complex logic that a business may pose.

Integration

Integrating with your current systems and processes is a major plus in the custom software solutions column.

There is no limit to the ability of custom solutions to work with existing systems due to its flexibility of design and potential for easy modifications.

In fact, many times custom solutions will intentionally connect and integrate with other ready-made solutions to fully realize their potential and personalize their features and functions.

Business partners

Most companies find it necessary to work alongside or within the systems of suppliers, vendors, or partner businesses. If, for example, you provide shipping options for your customers, **the integration of preferred shippers platforms to instantly estimate costs will help boost productivity, increase inside sales, and improve conversions on your online website.**

Future growth

As companies grow, challenges can arise to complicate current systems that are not flexible or easily modified. For example, Customer Relationship Management (CRM) products may not easily grow with a company's needs. A Software Advice survey of 304 CRM software users points to the problems experienced by growing companies using such products. Their issues came to light when product "options such as custom fields and modules are not enough to satisfy the needs of their specific business or industry." Custom software, on the other hand, addresses these needs by design.

The multiple benefits that custom software offers combine to ensure your solution grows with you.

No business opens its doors with a predetermined cap or limit for how much business they want to do. The real issues lie in how much business they are capable of doing. And because every company leader looks forward to profitability and growth, implementing custom software makes sense on all fronts.

Every company leader looks forward to profibility and growth

4

Can I build my new product with custom software?

> Software is a great combination between artistry and engineering.

- BILL GATES

Your unique business

Before considering software, it's important for you to establish what makes your business stand out. What are your specialties? How are you different from your competitors? What are your products or services that make you unique?

Understanding what sets you apart from other businesses is key to your success.

This is especially true for small to medium businesses. It's also true for startups who have yet to establish their brand. In fact, according to Startup Nation, "Understanding what makes you unique (should be) the number 1 priority for every person and every business."

This makes sense when you realize that customers today go out of their way to investigate and research companies to compare offerings. It's important for them to know what makes you special in order to justify their selection.

What about your products? Do they have to be unique? According to Inc.com, "Innovative and dedicated companies will continue to elbow their way" into markets. But at the same time, when you think your product has merit, just remember "that you don't have to change the world or upend markets. You have to add value to customers."

Advantages of building your own custom software product

Every business uses software. From emails to project management, software is part of a company's everyday fabric. But if you're thinking of custom software solutions to build, enhance, or expand a product, you'll revel in what it can do for you.

- Custom software meets your specific business requirements. Your software will grow along with your business and be maintained if it needs to be. It coincides with your long-term business strategies as you won't have to worry about a commercial software company discontinuing their product or your business outgrowing it. In addition, you own the software and the contained data.

- Because custom software is tailored to work within your current technology ecosystem, you won't find yourself having to make numerous adjustments to make it work. It is designed to accommodate your current hardware and software.

- When securely built, external threats to your custom software will be greatly reduced due to its uniqueness making it more difficult to hack. It's much easier for hackers to attack commonly used software with known vulnerabilities.

- If you’re looking for custom software to work with other software packages, you won’t be disappointed. The opposite is true for commercial software options and are more likely to cause errors, reducing productivity.

- When your business changes, your custom software will change with it. Making adjustments are much easier and can be implemented quickly and simply.

If you decide to go with custom software, whether improving an existing product or creating a new one, there is a fundamental process that it must go through. Engineers will need a prototype that can be tested to determine its stability and usability. In addition, it will need to be verified that it performs according to key performance metrics and void of bugs.

The basics of building custom software

Custom software is more expensive than ready-made software products but is well worth the investment. You will need to commit to the time, energy, and resources to ensure its success. But before you actually start the building process, you need to design and develop a prototype to validate its usability and purpose. All parties, including the product manager/owner, tech lead, business analyst, and UX/UI designers should be involved in the process. So, if you've made up your mind and are ready to commit to custom software, here are some basic steps to start the build.

1

Establish product goals

First and foremost, establish your product's goals and objectives and how it will fit into your overall business strategy. How will this product increase productivity? Will this software improve business processes? Does it have the potential to increase revenue? What is the expected ROI?

2

Define an MVP

Start with an MVP (Minimal Viable Product) build. This will give you the ability to begin with a product that's void of costly features but ready to use or take to market. You'll reduce investment risks and allow your product to gain momentum before enhancing its design.

3

Set objectives

Set specific objectives for your MVP and set reasonable expectations and estimates for its progress and completion. What is the product supposed to accomplish? What is the best way to achieve results? What is the budget, schedule, technical and engineering needs, overall size, etc.? Why is solving this problem with custom software development important to the business?

4

Design

Begin the design process by assembling a skilled team. An ideal team is made up of a product manager/owner, tech lead, business analyst, and UX/UI designers. Team members with both technical and soft (communication/ collaboration) skills will prove to be the best contributors.

5

Research & analysis

Do research and analysis to uncover competitors and any existing products. Also, pinpoint potential problems, look for technical constraints, and review any industry standards and legalities.

6

Look at the risks

Determine the product's inherent risks and all the requirements. Will people actually use your product? Is your product user friendly? How long will it take to build? Do you have the right skills and technology in place? Will the product work with other aspects of your business?

7

Tools

Make sure you have the right tools needed for the design phase like whiteboards, wireframing and prototyping tools for visual and interactive user flow mockups, etc.

8

Brainstorm

Let the brainstorming process begin to vet out as many ideas as you can. An effective brainstorming team consists of 5 members. And the more diverse the team the better.

9

Create a prototype

Create a prototype that is interactive, usable, and is ready for testing. A good prototype will validate your concept so don't take any shortcuts. It's a real-life model that represents your final product so make it count.

10

Test, improve and repeat

The chances of your prototype hitting the mark the first time is remote. You should expect the need for improvements and additional testing before finalizing your product.

Enjoy the benefits of your custom software build! If done correctly, you will differentiate your business from competitors with added value your employees and customers will appreciate.

Custom software is more expensive than ready-made software products but is well worth the investment.

5

When should I consider off-the-shelf software?

> We cannot solve our problems with the same thinking we used when we created them.

- ALBERT EINSTEIN

When should I consider off-the-shelf software?

Choosing between custom software versus commercial/off- the-shelf software can be difficult in some cases. Your business model (whether traditional or digital) and uniqueness of business processes may dictate what will work best for you. There may be times when off-the-shelf software makes the most sense. For example, most smaller businesses may only need accounting software. But not always. In fact, if you stop and think about it, custom software can realize a new idea with innovation. Or, it can improve the way you do business, especially if your unique processes can't be adequately addressed with off-the- shelf products.

Evaluate what path your business will take... and how your strategy will get you there

Look at your business model and evaluate what path your business will take, your objectives, and how your strategy will get you there. If technology/software is part of the plan to innovate or differentiate your product, consider custom software and budget accordingly. Otherwise, buying off-the- shelf for your business may be the way to go. Choose wisely after all considerations have been evaluated.

Building around existing platforms

The need for software solutions become especially apparent when there is a need to solve a problem. For example, if you're already in business, you may be looking for a better way for salespeople to communicate with customers. An innovative solution may be to develop an app that would allow 24/7 communication between customers and account managers. Is this a good reason to jump into a custom software project?

Since this is on the wish list of many companies, it may likely already exist. Why? Chances are something already exists as a white-label solution developed by a software company.

> **A white-label solution is a generic, non-exclusive option that was created for a specific client. It's essentially sold and rebranded under another company name.**

Many times, these solutions are available and can be easily replicated and shared. And they will likely save you time and money, bringing your solution to your customers faster. But it doesn't really stop there. You should take a step back to see how such an addition or change will affect the rest of your business processes.

If your solution is an off- the-shelf product, the real question is if this solution integrates well with your existing systems. Or, will it cause more problems than it's worth. If you find that you'll need to do extensive adjustments to make it work the way that you envisioned, need to increase user licenses, etc., you may consider the long- term investments of this option before making the purchase.

Ways to incorporate existing options (Saas, PaaS)

But there are ways to effectively incorporate other software options into your existing systems. For example, solutions such as Software as a Service (SaaS) that solve a problem so you won't have to reinvent the wheel can be cost effective. This solution is managed at one central location, hosted on a server, and is accessed from the internet. Users of SaaS are not typically responsible for hardware of software updates.

Also known as on-demand, web-based, or hosted software, Salesforce is one of the best examples of a SaaS. Salesforce is a customer relationship management solution used by companies to manage prospects, leads, and customer information.

There are numerous cloud-based solutions in this category including common other ones like Google Apps, GoToMeeting, Cisco WebEx, Slack, Zendesk, and Microsoft Office 365. Such an option works well for small businesses who want to quickly set up shop online, have short term projects that need online collaboration, or need apps to be accessible on the web and mobile devices.

Platform as a Software (PaaS) is another option to consider. PaaS provides a platform for the purpose of software creation. Delivered over the web, the platform allows for designing and creating applications (middleware) that are built into PaaS with special software components. It lets developers focus on building the software and not have to worry about things like operating systems, software updates, storage, or infrastructure. These types of applications are scalable and highly available due to their cloud characteristics.

PaaS is cost effective and allows for the creation of customized applications and greatly reduces coding. Some of the more common examples include AWS Elastic Beanstalk, Windows Azure, Google App Engine, Force.com, Apache Stratos, OpenShift, Heroku, and Apprenda.

More affordable

If you're the owner of a small start-up business with limited funds, custom software may not be an option. In addition, you may not have an IT team in place. Off-the-shelf options like presentation, graphic, accounting, and other software offerings may make sense—at least until your business is proven successful and starts to expand.

On the other hand, if your daily business operations will rely heavily on technology and digital transactions, you may decide to increase your budget in this area. For example, if you don't require a small fleet of trucks due to your drop/ship model, you may look at increasing the technology budget based on the role it will play in your operations. Will online shipping and digital customer service (live chat) options drive your revenue? Are you a new mattress manufacturer who's developing innovative mattress designs and containers that require custom software?

If you decide you're not ready to make the custom software investment, you should determine what you will need to initially get by. Then once your business is thriving, you may need to turn to custom software to digitally transform your business, improve processes, adopt automation, etc.

Benefits vs. limitations

Off-the-shelf software solutions can save you time and money. They work well for small startups and small businesses who don't rely on unique processes. Off-the-shelf software can help you get up and running in no time with low, upfront costs. You won't be responsible for updates or its maintenance. So, if the problems you face are typical for your type of business, the off- the-shelf options may work well for your needs.

But there are limitations to this software option. First and foremost, off-the-shelf software may not suit all your business needs. It may be difficult to modify. And it can present compatibility issues. The low, upfront cost advantage may not be worth the long-term costs (license renewals, additional users, etc.) you might face with this option. But more importantly, it's available to your competitors so they can offer the same services!

Other considerations

Whether you choose custom or off-the-shelf software, the decision that you make can have long- lasting effects on your business. It's important to determine specifically what your software will address and how it will make a difference to your bottom line. If your plan is to make improvements or to create a new revenue stream, the right software choice can make all the difference.

6

IT team considerations

“

Great things in business are never done by one person. They're done by a team of people.

- STEVE JOBS

IT team considerations

The positive energy of teamwork is a powerful force in business. And this force has been around for a long time. From protecting tribes to landing on the moon, challenges are no match for a group of diverse and skilled individuals looking to make a difference.

IT teams are your first offense when innovation and automation are called into play.

Technical teams are critical to the creation of software, particularly custom software. Your ideal IT team is made up of diverse individuals with varied skills and personalities. Made up of engineers, designers, and developers, IT teams are your first offense when innovation and automation are called into play. But they are also key to solving short term and long-term business problems. Technology and those who answer the call of software needs can make things happen in today's competitive business world. They develop solutions to increase productivity, revenue, efficiency, and so much more.

How to relate your concept to the IT team

If you have an in-house IT team, you know how important time and effort is to every project. So, it's important to convey a project's objectives with clarity. But what if you're not a techie? Do you need to know coding to get your point across?

You really don't need to know IT shop talk to discuss a project with your team. As a business leader, it's more important to clearly define what the product is supposed to do. What is the desired end result?

Before any time is spent on the project, establish its purpose. For example, if your project involves creating a mobile app to complement your online store, what are you expecting to achieve? Will this app make it easier for customers to buy from you? What value are you hoping to provide and how do you expect to generate revenue from it? Focusing on what really matters in the end will help your team establish the best route to get there. They may look at staying consistent with some of the most popular mobile apps for businesses like yours. Or, if the apps out there are not ideal to your business needs, they may start from scratch. In any event, the idea is to send a clear message of your project's intentions and purpose.

Communicating with your technical team should not be any different than dealing with your other teams. Follow the best ways to communicate as a leader to keep the team motivated to move in the right direction. Be sure to ask the right questions and encourage feedback. You'll learn a lot from the answers you get.

Communicating with your technical team should not be any different than dealing with your other teams.

Getting IT onboard with your goals

At this point, it's assumed that your team is well versed on your company's vision and values. If not, that should be your starting point. They should be connected to your company’s mission and understand how they play a role in that goal. Once that’s taken care of, link your project to your company’s vision to ensure the team has a grasp on the guidelines associated with it and how it fits into the overall scheme of things.

If your project is enormous in scope, you may need to break it down into smaller parts to make it more doable. Take into consideration the other projects the team is working on. You’ll want to give your team every chance for success so make expectations realistic.

Encourage brainstorming and be sure to listen to their feedback during the design process. Allow ample time for developing a prototype and user testing. You will likely need to go back to the drawing board a few times before getting it right. Finally, support the team with the right tools and a supportive environment.

Product manager (do you need one?)

Unless you plan on leading the product team yourself, you should hire a product manager (PM).

This is especially true if your project is complex and you’re inexperienced in this area. Your team may find itself getting behind in projects and not sure which to prioritize. Or, worse yet, maybe the team is not realizing the vision behind the build.

Finding the right person to be a project manager can be challenging. Your relationship with this person will be key to successful project outcomes. That said, look first at your current team for someone who might fit the bill. On the other hand, just because someone is an awesome designer or engineer doesn’t mean they will be a successful PM.

When you’ve decided to hire a PM, whether internally or externally, look for positive leadership qualities. Make sure they are a good communicator and more importantly, that they are a problem solver.

Product manager responsibilities

Once you’ve decided that you need a PM, define what their role and responsibilities will entail. The role is a critical one in the product build.
Your PM should oversee the build process and successfully manage a cross-functional team. It is primarily an organizational role that sets the strategy with a roadmap for how to get there. A good PM will communicate the goals and objectives for the product from beginning to end.

As a decision maker, the PM will need to convey why the decision is the best option. Your PM should skillfully prioritize the team’s goals and balance the needs of the team and the customer. Your ideal PM is tuned into the short and long-term vision for each build.

If you hire someone externally, ask for referrals from people you trust. You should also post a job opening on job boards who cater to technology hires. Just make sure you define the job description accurately to be sure you find the best fit. Check out other job postings for one that matches the role and responsibilities you feel are appropriate.

Your PM should oversee the build process and successfully manage a cross-functional team.

7

Technical Considerations

> “The art challenges the technology, and the technology inspires the art.

- JOHN LASSETER

Technical considerations

Before the process of software development begins, decisions will need to be made on development methodologies, design methods, and technology tools you plan to use. What you decide will have an impact on your team's workflow and on your project. That said, no one methodology or tool will guarantee success. Consider factors like project requirements, project size, team size, timeframe, you or your client's availability, and complexity of the deliverables.

No one methodology or tool will guarantee success

In this section we will take a look at some of the choices you will need to consider (if you have an internal team) with overviews of some of the key methodologies, design processes, and tools to consider.

Waterfall vs. agile methodologies

Choosing a methodology will dictate the road you take towards developing your software. And regardless of which method you choose, you'll get the same results. So, while Agile and Waterfall methodologies are very different ways to go, your end product/project will be essentially the same.

Waterfall

So, how do you decide which of these methodologies will work for your project?

Look at their key differences. For example, the Waterfall methodology is considered the more traditional of the two. It takes a slower approach and requires careful planning of each step along the way. Originally developed in 1970 by Dr. Winston Royce to build large, risky software systems, the Waterfall method helped to ensure that complex projects were completed successfully.

With the Waterfall method, each member of the team is on the same stage of development. The six stages of this process include planning, analysis, design, implementation, verification, and maintenance. Each stage of the Waterfall method must be completed before you can move on to the next.

Pros

→ Straightforward planning and designing due to clear objectives

→ You can easily measure progress at the start and end of each stage

→ Easier to manage with fixed deliverables

→ Well-documented processes and results

→ May save time and money.

Cons

- More vulnerable to mistakes, incurring additional time and costs
- Changes are difficult as you unravel work done in previous stages
- Problems take longer to discover
- Usually one massive product is delivered at the end
- Susceptible to more assumptions made along the way due to distance from client

Agile

Agile, on the other hand, is more about speed, trial and error, and iteration and experimentation.

Although seen as a modern method, the application of iterative and incremental development dates back to the 1950's. It relies more on customer involvement and focuses on a speedier delivery as quick steps are checked off as the project progresses.

This type of project management involves a highly collaborative team and is more about breaking up the project into smaller pieces (tasks). With smaller work segments, you can count on more adaptations based on frequent reviews. It allows for greater flexibility.

The Agile process accommodates changes in the development process as problems are discovered quickly. Overall it is a simpler method and can vary on the number of steps. Basically, it involves planning, execution, launch, learn, and repeat.

Pros

- → Less risky as your output is reviewed as you go along
- → Quicker feedback and regular communication with customers
- → Greater adaptability to changes
- → Promotes teamwork and improves decision making process
- → Output quality is improved with each iteration as you learn from mistakes

Cons

- → Deadlines could be an issue with faster delivery
- → All team members need to be equally dedicated to the project
- → Team members must be cross skilled, highly proficient, and self-disciplined
- → Sometimes documentation may be ignored and lost in the process

No code vs. low code

Without getting too technical here, the discussion of low code versus no code is an important one to software development. Both platforms are used to develop software applications without the need for written code. This means that users can design and build applications without writing code.

Instead, they use a visual approach (visual integrated development environment or IDE) that makes this method accessible to more people.

Developers choose functional components from an on-screen library to drag and drop them into a visual workflow, and frequently offered in a PaaS form factor. This allows “citizen developers” to connect functionality and create a mobile or web app. So, what’s the difference between the two?

No-code tools

No-code application development tools are suitable for small applications and ideal for solving basic business functions. Considered too simplistic for complex uses, they are better suited for single department use. Great for the citizen developers/nonprofessionals, these tools are sometimes too limited and don’t provide scalable options. The integration tools that can be used with the no-code platform are usually proprietary and have limited capabilities.

Low-code tools

On the other hand, low-code platforms, provide the ability to quickly build and deploy high-quality web and mobile apps. In fact, you can create multiple apps in the fraction of the time it takes to hand code. They streamline app maintenance and updates and integrates with your current IT infrastructure. Considered as a general-purpose solution, low-code platforms work on all major mobile platforms. Full-featured platforms automatically generate, compile, and deploy necessary code.

UX and UI design

UX (user experience) design and UI (user interface) design work closely together in creating products. That said, the two design methods are very different. While UX is technical/analytical, UI is more about aesthetics. Each plays an important role in the look, feel, and functionality of your product design. Here's how:

UX design

An important goal for UX design is to understand the user. It is a collaborative process that focuses on understanding how users process information. By empathizing with users and identifying with their processes, UX design can improve the interactions between your customers and your company.

To do this, a good UX team will take steps to zero in on what motivates your customers to engage with webpages, mobile apps, etc. Therefore, UX design is responsible for research, development, content, prototyping, and testing. The steps include collecting ideas, validating assumptions, and gathering and reviewing feedback. If done correctly, the result is improved usability, ease of use, and increased user engagement.

In essence, good UX design creates a seamless user experience that engages them without having to think about it. Bad UX design frustrates users by lacking intuitiveness, which can cause a user path that includes more steps, more clicks, etc. The Interaction Design Foundation recommends understanding what your users need. Other tips include labeling links and not over-using animation in the design, among others.

UI design

UI design, although focused on visual aspects, is still very important to the design process. It may take a backseat to UX but still plays an important role in attracting and motivating users to engage in your content. UI is responsible for the visualization of your brand's assets on your product's interface to enhance the user experience. Think of it as a guiding light for users as they interact with your product, webpage, mobile app, etc.

But getting the look and feel right involves more than just a pretty cover. UI designers have their share of customer analysis, research (design), branding and graphic, user guides and storylines. UI designers can also be responsible for prototyping, animation, screen size adaptations, and working with developers on implementation.

UI design gives credibility to a brand with a look and feel that can be trusted. Looks, after all, are important to users and can affect whether they stick around or choose another company to deal with. For example, if your website looks bad, regardless how good the UX design may be, a user may leave the site before ever clicking on anything. UI delivers a strong visual presentation that leads users with guides and directives. It's accountable to how users react and interact with your content.

If you're confused about the similarities and overlap between UX and UI design, join the club. They both share a goal in knowing and engaging users.

Technology tools

The evolution of technology has progressed rapidly over the past few decades.

And more recently, it has become a moving target, making it difficult to keep up. But due to this ever-evolving characteristic of technology, satisfying business software needs for today is not enough. Your business software needs to also suit your future needs.

There's no question that business and technology are linked. But not all of what technology trends have to offer are going to have a direct impact on how you do business. For example, robot dexterity or miniature microscopes in a capsule may not have relevance to your business, products, or services—or maybe it does.

As a business leader, making decisions on digital technology has a lot to do with where you are in business. For example, a start-up business tends to look for the latest technologies to match their goals of being the newest biz on the block. On the other hand, if you are a company that has been in business for years, you may already have some great technology already in place.

Here are some considerations for your software development project:

Multiplatform user interface

Tools like Google's Flutter and Microsoft's Xamarin save developers time by eliminating the need to rewrite apps to accommodate multiple platforms like Android, iOS, and Windows.

AI-enabled applications

Developers can use simple line of code to incorporate AI into apps with toolkits like Microsoft Cognitive Services.

Cloud Services

Besides cost savings and easy deployments, the on-demand availability of cloud services continues to expand. Provided by cloud computing providers, services include software, storage, databases, networking, analytics, intelligence, and more.

Integrated development environment (IDE)

Tools like Linx, a low-code IDE and server, allow for faster development with drag and drop functionality and services.

Blockchain technology

Due to privacy, transparency, and security, blockchain technology is expected to grow with more secure and accurate applications in healthcare data identity management, contracts, supply chain management, etc.

Progressive web apps

Websites that look and feel like mobile apps with all the capabilities and features without actually downloading an app for a seamless user experience.

Endless possibilites

Whatever path you choose to acquire your custom software, it’s important to stay flexible. If your in-house team is developing the software, realize that there could be a time when they will need to switch gears and change their path. If your method of choice is Agile, change will happen frequently during the process. In any event, consider your team’s skills and any possible training needs for best results.

Of course, you can also consider outsourcing your projects. This may be your best option if the project is expected to make too many demands on your current in-house resources.

8

Should I outsource my project?

> Custom software development relies on a deep commitment to the product and deeper passion for success.
>
> \- EPHRAIM ARNSTEIN

Should I outsource my project?

There are multiple factors to consider if you're thinking about outsourcing your custom software development project. Consider the other projects that you currently outsource to other companies in different areas of your business. When did you decide to leave such work to the experts and what reasons brought you to that decision?

Outsourcing is expected to grow 8.4% in 2019 in the US

For some small businesses, custom software may not always be an option, whether done in house or outsourced. But when the budget and time allows for it, companies may need to outsource for lack of a skilled, in-house team. Many companies are expected to turn to this solution in the future. In fact, according to Gartner, outsourcing is expected to grow 8.4% in 2019 in the U.S. alone to $421 billion in enterprise software.

How do you decide when you need help

You might start out by thinking about your core business. Ask yourself the reason you are in business. Are you trying to start your own software development company? Or, is your goal to improve the operations of the business you're currently running?

Once you've determined why you are in business and what you are trying to achieve with your new custom software, think about your business strengths. For example, is software development something that you're already familiar with? How proficient are you at leading a software development team?

If you don't have a software development team in place, will you be able to handle the hiring? How many people will you need on your team? What roles will each team member handle? What skills, experience, and background will be needed for each role?

Building an in-house software team is like building a new division inside your company. Do you really have the time? Is this what you are trying to achieve?

Weighing costs and benefits

In-house team

Hiring and putting together an in-house software development team can be a very costly proposition.

Cost considerations

It begins with a long hiring process that will require finding the right mix of team members (developers, engineers, designers, etc.) each with the right skills. The salaries and benefits for these talented and skilled workers are not cheap and can vary by geographic area. The more complicated the project, the more likely you will need specialists who would command a greater salary. Just look at what other software development employees make at companies that are similar in size and industry. Costs also increase if you are faced with a need for increased employee training and expanded office space.

Time is money

Your newly hired developers and engineers will then need to assemble the rest of the team members and establish their workflow. Of course, this means that there could be a delay in the process until all team members are hired and ready to begin work on the project. And don't forget, this is a regular monthly expense, whether they're spending all their time on your project or doing other company-related tasks. Once the project is finished, the costs will likely continue as you will probably keep most of your team on for maintenance or other IT-related work.

Project size and deadlines

How big is your project? What size team will you need to get the job done? The size of your in-house team will be based on the scope, details, deadlines, and amount of work your project entails. The larger the project, the larger the team. If you're deadlines are tight, you may need to hire more people on to get the job done.

Can you continue to support the team?

Having such a skilled, talented team can be a huge asset to your business. The question is what plans you have for keeping them motivated and engaged in the long term. Will the project quickly scale your company and be able to support the team before and during development, and beyond?

Outsourcing

Outsourcing your project, on the other hand, is an investment that is specific to what you are trying to build. The costs are directly related to your project. Every dollar spent is tied to every effort and moment that is dedicated to your software development solution.

Start project sooner

When outsourced, your project can begin sooner as a dedicated team will be quickly assembled from individuals who know their way around software development projects. This outsourced team will be made up of talented professionals who will save time because of their experience working on custom builds. Many times, these teams of specialists have worked and collaborated on other projects that require the same skill sets. They have a tried and true process in place that ensures successful outcomes sooner rather than later.

Reduce error

With an in-house team, there is a risk of error that could affect your other business processes. On the other hand, an outsourced team is guaranteed to produce the results you're looking for. You won't have to turn elsewhere when the project looks as though it's going in the wrong direction.

In addition, it's important that an existing IT team stay focused on adding value to your current business. This will help keep your current processes running smoothly to help ensure productivity and efficiency.

Security

If your in-house team is not well-versed in security, you may be faced with security breaches that put your company's sensitive information at risk. With outsourcing you will be protected against breaches as coding and processes of development are secure.

Stay focused

But maybe the greatest benefit to outsourcing, is that you can stay devoted to your core business and not derail any company priorities. Do what you do best to keep your business on track without being distracted with major diversions that are best accomplished by trained professionals.

When outsourced, your project can begin sooner as a dedicated team will be quickly assembled from individuals who know their way around software development projects.

9

Finding help for your project

“

> To turn really interesting ideas and fledgling technologies into a company that can continue to innovate for years, requires a lot of discipline.

- STEVE JOBS

Finding help for your project

Once you've realized that your project is more than you can handle, you'll need to find help. The reasons for looking externally for help can vary but are usually due to project complexity and scope, time limitations, lack of IT team and/or specific skills, lack of resources, etc. And although it may sometimes be difficult to do, asking for help is one of the best ways to avoid costly errors.

How do I look for help?

The short answer to looking for help on your custom software project is to start by doing what you would normally do when looking for any vendor. Ask for referrals. Check online reviews. Read up on the subject. Do online company research.

Start by thinking about your software development company as a partner.

But there are a few things to consider before your active search begins. Depending on the scope and size of your project, you should start by thinking about your selected software development company more as a partner. The partner you choose should be able to fully understand and share your vision. Doing this may narrow down some of the potential options.

For example, are you looking for a full-service development company that can take your project from beginning (idea) to end (final product)? Or are you looking for improving an existing product?

Do you prefer working with an onshore or offshore company? You may want to select an onshore software development company as they offer a better command of UX design, provide better communication, and are able to meet you in person. Onshore is a good choice particularly for fast-growing businesses who are looking for high-quality solutions for a new product. The downside is the cost due to higher regional rates.

Offshore companies, on the other hand, may be the cheaper option but may require an already well-defined project with specifics. Using an offshore company may pose problems such as language barrier issues, cultural differences that interfere with UX design, and having to deal with different time zones.

What do I need to prepare

That said, it's important to know what you want to achieve with your custom software, both now and in the future. Create a spreadsheet that defines your product’s goals and your expectations. Have the answers for questions that will affect the estimates that you will receive from vying software development companies. What is the scope of your product? Do you have an overall strategy?

What is your vision for what this product will do for the user?What is your target timeline? Will you need designs? Are there any legal requirements that must be incorporated into your product? What about your current technology, systems, and software?

What is your vision for what this product will do for the user?

Knowing your needs and putting together a list of must-haves will help you find a software development company who can meet those specific needs. Make a list of required elements, desired features, and wish list. That way, you can be sure that your product will be designed to accommodate growth, improvements, and future modifications.

How to select a software company

Choosing the right software development company **for your project is critical to its success.**

What should you consider to be sure you uncover the best company for your project? You may take Entrepreneur's advice and start with defining the geographic area for your company search. You may want to explore some local companies and then expand outward.

Look for companies who have done similar work in the same industry. If you've identified technology that can build your project, look for companies who have expertise in that area. What methodology does the company use to develop the software? Is the project rolled out in phases or all at once?

Once you feel that you've found the right company, ask for referrals. Check for references, testimonials, and online reviews. How long have they been in business?

When you've selected a list of possible software development companies to work with, research their experience and past projects. Do they have the resources to handle the size and scope of your project? What does their design and build process look like? What about their maintenance policy?

Are there hidden costs that may surface later?

Of course, ultimately you will narrow your list down to companies who better accommodate your budget and timeline. Cost may be the biggest factor in your final selection. Just be sure that you look at why one company charges more. What are they offering that another company is not including in the quote? Are there hidden costs or expenses that may surface later with the cheaper alternative? Does one company offer more service incentives than another?

How Bitbean can help

Whatever your next project entails, Bitbean software development company can help you achieve success. With experience in several industries with a multitude of solutions, we have the expertise to make your vision a reality.

For more information on how Bitbean can help, visit www.bitbean.com bitbean