



FIELD SERVICE SOFTWARE

Beyond Build vs. Buy

Are you tempted to build your own field service software?

It can be an attractive option, with the opportunity to customize to your exact needs, full control over security and system updates, and guaranteed integration with the tools and processes you already use.

But building your own enterprise software also comes with its downsides. It's significantly more expensive up-front than buying software, and your time to deploy is a lot longer. It can also come with some costs you may not have thought about.

With that in mind, here are some **pros** and **cons** of both buying and building your field service software.





Build Your Own Software


Why It's Attractive

Building your own field service solution has several advantages:



It's highly customized

If you build your own software from scratch, you can tailor its functions, features, and performance to your exact needs. This is especially attractive if you have complex or unique use cases that might not be covered by a one-size-fits-all purchased solution.




You own the software roadmap

Customizing your software to your unique work environment means you can bake integrations with other software and platforms into the system. Building that compatibility in from the start means you don't have to find new solutions for other parts of your workflow or add extra silos to your process.



It can integrate better with your current tools.

Customizing your software to your unique work environment means you can bake integrations with other software and platforms into the system. Building that compatibility in from the start means you don't have to find new solutions for other parts of your workflow or add extra silos to your process.



You have better control

You can respond much more quickly with things like bug fixes and feature upgrades as your users need them. You've also got better control over versions and release schedules. And because you design and monitor the solution yourself, you have the intimate knowledge of the code necessary for resolving problems quickly.



You (might) save money

If your solution is relatively simple and you can hire an inexpensive vendor to build it, you may be able to achieve a better ROI than with a commercial solution. Before looking just at the price tag, though, you need to think about the additional time and resources involved in building, maintaining, troubleshooting, and improving the tool – which can all eat up your initial savings.

The Downside

While the above advantages are attractive, designing, building, and monitoring custom software also comes with a significant set of challenges. Building a field service solution is often a more complicated, time-consuming, and resource-diverting option.

Here are a few of the disadvantages you may run into:



It has a large up-front time investment

From hiring skilled developers (if they're not already on your team), to designing your software, to assigning project managers and planning deployment, building your own software takes up a lot of time. Established software companies can spend millions of dollars and take years to build a minimum viable product, let alone a stable and feature-complete one for reliable internal use.



You're on the hook for ongoing maintenance costs

When you use a SaaS field service platform, your monthly or per-user fee covers the ongoing maintenance, troubleshooting, and innovation for the software. If you build your own solution, you absorb all of those responsibilities and the costs that go along with them.



Integration can be complicated

Field service software can be massively difficult to integrate into existing systems, even when you're coding it yourself. And once you've managed to build the APIs and integrate your systems, managing the integrations (especially as the other software gets updated) is an ongoing challenge and drain on staff and resources.



You'll have to keep up with growing and changing organizational needs.

While your custom field service platform may address all of your current business requirements, it may not be well positioned to grow your business or scale with it effectively.



It's rarely funded for an entire product lifecycle

Many custom-built solutions end up rarely updated, used until they're obsolete, and don't get funded for an entire lifecycle. In addition to not integrating best-in-class technology, you end up wasting time, efforts, and costs on software that only works well for a limited time



You have to hire additional expertise

Software architects, engineers, analysts, project managers, and other subject matter experts don't come cheap – but all are crucial to a smooth rollout. If your company doesn't have the bandwidth to hire all of these experts in-house, you'll have to outsource some of these roles. This reintroduces vendor management, cutting out one of the main benefits of building your own software.



It has an opportunity cost

Vendors have insight into how other businesses are handling similar problems in their industry. Without this insight, you're limiting your ability to innovate and grow both your software platform and your overall business. And by focusing on all of the tasks involved with building the solution, you're taking time away from other business goals.





Buy a Software Solution

Why It's Attractive

The clearest benefit of buying software is outsourcing all of the costs of building and maintaining the tool to third-parties.

This singular advantage leads to massive savings in capital, time, and resources.

Buying software has several other benefits:



It saves you money

The most direct savings come in the form of vendor pricing structures. They're far simpler and more transparent than trying to calculate your total cost of planning, designing, building, rolling out, and managing your in-house solution.



It's easier and cheaper to scale up

Instead of having to try to grow your software yourself to meet changing needs, you can just pay for additional users or an upgraded tier of service. Growing becomes much simpler without sacrificing speed, service, or core values.



You'll have better security and reliability

Increasing competition demands reliable and secure software. Consumers are also getting more concerned about their private information and how it's being used. Software vendors get strict security certifications – and because their ability to demonstrate uptime can make or break their business, their solutions deliver secure, reliable system access.



It provides more advanced and expensive features.

The ongoing development of a purchased solution is spread out among all of the vendor's customers. That gives them the flexibility to introduced more advanced features that would be too expensive to develop for an in-house build.



You don't have to provide your own support

Established solutions have dedicated and vibrant user communities and good support resources. This gives users the opportunity to connect with peers from other companies, learn how others use the tool, expand their network, and troubleshoot individual problems faster. Taking this pressure off your IT team lets them focus on other, more pressing concerns.



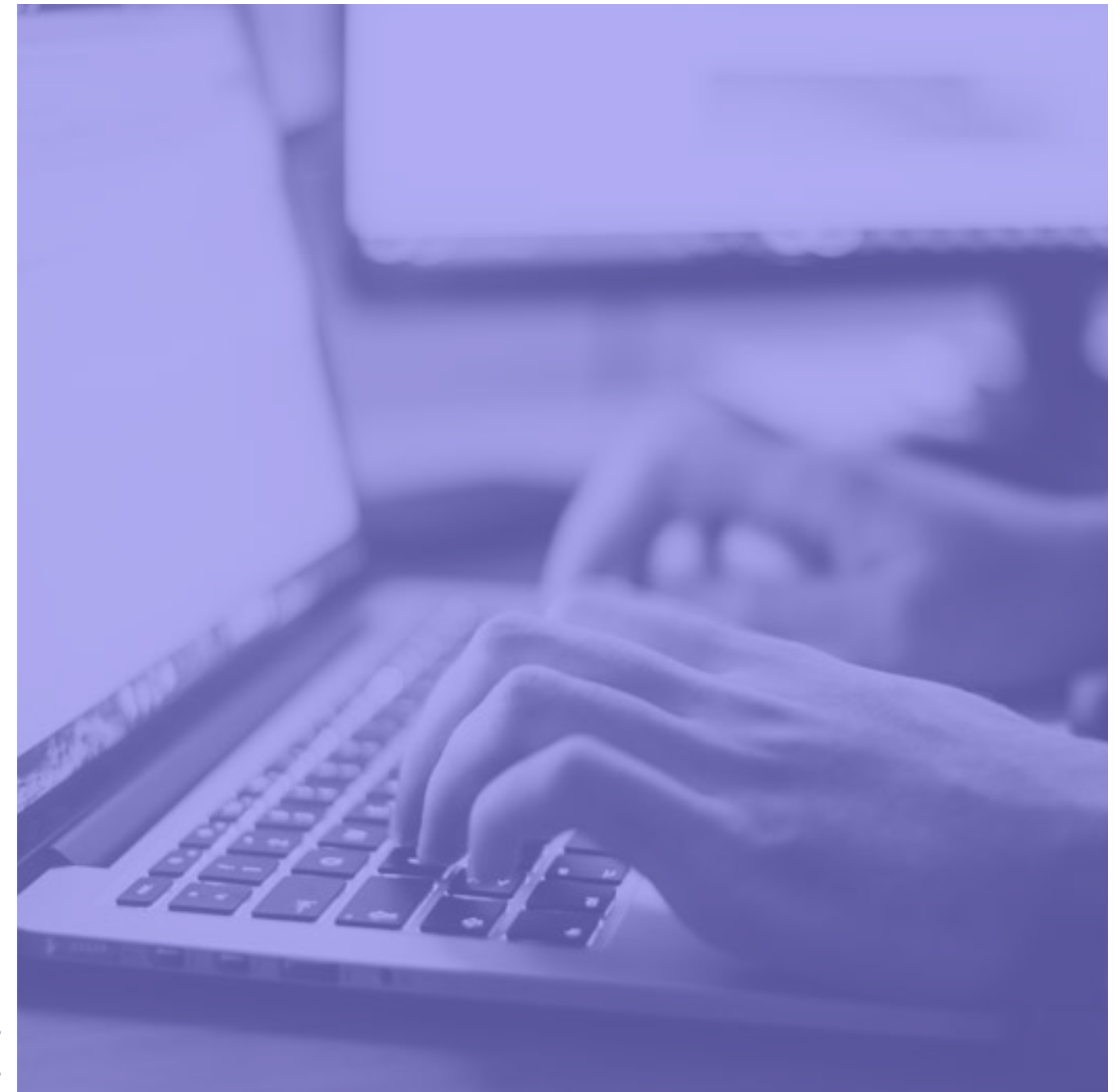
Integration and compatibility are built-in

Vendors know their customers have a wide array of existing tools and systems, and constantly work to make their platforms compatible with more of these other pieces of software.



You can focus on larger business goals

Letting the software company handle engineering, testing, monitoring, upgrades, and maintenance lets your internal staff use their creativity to meet organizational challenges.



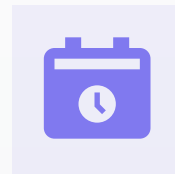
The Challenges

The disadvantages of buying software mostly align with the advantages of building:



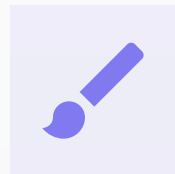
You have less control over bugs and upgrades

However, most third-party vendors do have feedback mechanisms that encourage customers to guide product direction.



You have less control over timing

Waiting for an external dev team to fix bugs, release patches, and roll these changes out on their own schedule introduces a level of uncertainty and inconvenience for your internal teams.



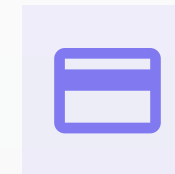
You don't have control over the UI framework

Third-party vendors have their own user interfaces that your team will have to adapt to, rather than having an interface customized for them.



You won't have as many custom-build integrations

But remember that vendors are constantly working to expand their integrations – and some field service solutions are specifically built for integration.



You're trading up-front for ongoing costs

Third-party software comes with fees like annual or monthly licensing fees. This still may work out to be a better option when you weigh it against the total cost of planning, building, and maintaining your own mobile workforce management tool.

Beyond Build vs Buy

With all of those pros and cons to weight for both of the above solutions, you may be back at square one in your decision-making process. It would be nice to find a solution that incorporates all of the pros without being weighed down by all of the cons.

The good news is, that solution exists. It provides easy, low-code/no-code configuration that lets you customize it to your needs without hiring an expensive developer team. Using drag-and-drop, you can choose the features your team needs and leave out the ones they don't. You get the enterprise-grade security infrastructure, stability, and scalability of a prebuilt software solution, while still getting a tool that can be customized to your changing business workflows.

Want to learn how to toss out the build vs. buy dilemma and get the best of both worlds?



Check out Zinier today.

hello@zinier.com | www.zinier.com