



The Software Success Scorecard

See how your design, development, and deployment stack up.
Get clear on what you need to do to make sure your project succeeds.

LAUNCHWARE

**MANY SOFTWARE
PROJECTS FAIL.**

**Yours deserves
to succeed.**

When your software projects succeed, everyone wins.

You and your team look like heroes. Your company grows. Your users' lives get better because of your work.

At LaunchWare, we've discovered making that happen isn't happenstance. It's predictable. From start to finish, you can literally engineer success. (Pun totally intended, by the way.)

The better you design, develop, and deploy your projects, the more likely you are to win.

Do those things well? The sky's the limit.

If you don't? Your project could very well crash and burn.

Getting more clarity on how your project is going doesn't need to take a bunch of consulting time, though. Actually, all you've got to do is answer a few simple questions.

That's what this Software Success Scorecard is for.

In the pages below, you'll find 21 simple Yes or No questions. It should take you like two minutes to answer them – maybe even less.

From there, you'll be able to take your scores and see where you and your team are at. Hopefully your results will give you lots to celebrate...and give you clarity on where to improve to better ensure success.

Best,



Dan Pickett

LaunchWare Founder



Design

The vast majority of companies don't go through the steps we ask about here. As a result, what they build gets built in a vacuum. To maximize user adoption, you don't want to do that.

Have you explicitly identified the audience you're serving with your software?

Yes

No

Have you clearly documented the problem you are solving for that audience?

Yes

No

Have you validated that this is a problem worth solving for them?

Yes

No

Have you had 5 or more conversations with members of that audience to confirm your assumptions?

Yes

No

Have you established 2 or more hypotheses about how your solution will solve their problem?

Yes

No

Have you constructed wireframes or mockups that illustrate how you will solve their problem?

Yes

No

Have you shown those wireframes or mockups to members of the audience you're serving?

Yes

No

Total

Yes

No



Develop

How you develop your application matters. When you don't use a clear and proven development process, it costs you. The more efficient this is, though, the more time and money you'll save.

Is your current development team meeting established deadlines?

Yes

No

Are you releasing versions of your software at least once every 2-4 weeks?

Yes

No

Are you writing automated tests as you go?

Yes

No

Are you inventorying development work to be done in a user story format?

Yes

No

Do your user stories include explicit acceptance criteria to help developers understand what "done" looks like?

Yes

No

Note: If you're not yet writing user stories, you should select "No."

Are you using an issue tracker to monitor progress?

Yes

No

Are you conducting regular demos of working software?

Yes

No

Total

Yes

No



Deploy

You don't have to lose sleep every time you deploy. With the right systems in place, you can be confident your team will spot any issues quickly and will proactively solve them for your users.

Are you practicing continuous integration?

Yes

No

Are you practicing continuous deployment?

Yes

No

Do you have a somewhat automated mechanism for releasing changes?

Yes

No

Do you have an exception tracking and reporting solution?

Yes

No

Do you have visibility into how users are engaging with your software?

Yes

No

Do you have mechanisms in place to facilitate feedback from your users?

Yes

No

Do you have alerting systems in place in case there are problems with your production infrastructure?

Yes

No

Total

Yes

No

**YOUR RESULTS**

Design Stage

These results correspond to your answers on page 3

Yes No

If you had 1-2 yeses, your project's success is in danger.

While you didn't mean to, you likely started on your project without a clear map or destination in mind. Most days, you can feel it, and it stresses you out. You know what you want to build, but you probably don't have a deep knowledge of your audience or the problems they're facing. Without validation for your ideas, you and your team could end up building something that doesn't meet the users' needs. Additionally, developers are likely spending extra time iterating requirements, not shipping features. Unexpected delays are frequent, the cost is running up, and your team spends lots of time reworking features. The sooner you implement well-designed, user-centric plans and processes, the better chance you have to succeed.

If you had 3-5 yeses, the success of your project is at risk.

You've done a decent job mapping out your project, but to ensure success, there's more work to do. More likely than not, some of your deadlines have already become moving targets. You're probably also not getting regular feedback or validation as you're building your product. Without those practices firmly in place, it's hard to know if you're building a product users will love. If you are struggling to course correct quickly and offer regular feedback on progress, your design process is likely to blame. You're at risk of needing to rework features later and could even be building a 'black box' that will be tough to scale. Still, you can tell success is attainable, but you'd like to make it feel like more of a sure thing and get there without a bunch of bumps in the road.

If you had 6-7 yeses, congrats! You have set yourself up to succeed.

By doing the necessary work in your design stage, you've got a competitive advantage over most of the market. You take the time to validate your ideas and talk with your audience. As a result, you're in touch with who your users are, what problems you're solving for them, and if what you're building actually solves those problems. Building wireframes or mockups is an automatic part of your process before building new features or products. That takes some time on the front end, but it's saving you tons of time as you develop your application. Better yet, the insights you get from your audience before you send things to production unlocks ideas for features that make what you're building better to use and has people raving about what you're doing.

**YOUR RESULTS**

Develop Stage

These results correspond to your answers on page 4

Yes No

If you had 1-2 yeses, you need to get the development of your application back on track.

Without swift and immediate action, delays (and costs) are going to keep piling up. You're probably releasing new versions of your software much less frequently than you'd like. You also probably aren't doing regular tests as often as you need to. Your code is buggier than it ought to be and mistakes often slip through. Without dramatically improving how you lay out what you need to do and testing things as they get built, your code will keep getting messier. But with some focused work, you can still make developing your application feel like a well-oiled machine.

If you had 3-5 yeses, you'd be wise to change some things up.

You've got some good pieces in place, but you can feel that gaps are there. You're making progress and your team usually is on the same page. Still, your development process isn't as bulletproof as you'd like it to be. More likely than not, as time goes by, it's getting tougher to get clarity on what 'done' looks like. When you do ship new features, existing ones can sometimes break. Deadlines also can tend to slip, holding you back from working on other strategic initiatives. You're almost there — but it's time to make some smart moves to shore development up to better ensure the success of your project.

If you had 6-7 yeses, great job! Your development is a well-oiled machine.

It feels great when development runs smoothly. You've put in the hard work to make that happen. You're meeting deadlines and shipping new features on a regular schedule every 2-4 weeks. Throughout your sprints, you know exactly how projects are progressing and overcome any roadblocks quickly. Everyone knows who's responsible for what and what "done" looks like. Your user stories are fine-tuned, keep everyone focused on the right results, and stops costs from ballooning. You do demos at regular intervals, which not only help you catch bugs early but also give you ideas to make things better. With your development humming along, you and your team are having fun, and you're set up to succeed.

**YOUR RESULTS**

Deploy Stage

These results correspond to your answers on page 5

Yes No

If you had 1-2 yeses you are seeing big problems in production.

Worse, your users could be having difficulty, and you're none the wiser! Not integrating or deploying continuously is slowing things down for your team and is leading to unexpected issues. And if you're releasing changes manually or you're in the dark about errors and user behavior, you're especially at risk of having to roll back your releases. You can still streamline your deployment and get crystal-clear clarity about the state of your software, but you've got to make some big moves to improve your processes and team.

If you had 3-5 yeses, improving your deployment will boost efficiency and control costs.

You've got a decent handle on deployment, but you have areas that can be polished and improved. Deploying still feels risky sometimes, and you don't always know how launching new features will go. If you're in this stage, implementing or improving upon any continuous integration and scriptable deployment practices will remove any sense of unpredictability. Additionally, getting better insights into user behavior will be a game-changer. With a little work, you can make your deployment strategy and processes robust, efficient, and stress-free.

If you had 6-7 yeses, fantastic! You've got a great deployment process.

Your smooth deployment is impressive. Each part of the process is well thought out, and your team knows what to do to execute each release with precision. As a result, your users experience minimal disruption and downtime, if at all. Because of your strong processes here, you avoid production problems and the need for roll-backs. You can spot any problems or issues almost immediately and also monitor your systems to spot opportunities for future improvements. Best yet, since you actively engage your users for feedback, you have created a user-centric culture that will set you up for continued, scaleable success.

Whether this assessment shows you're struggling in an area or doing quite well overall, LaunchWare can help you turbocharge your project.

At LaunchWare, we partner with technology leaders who are eager for their projects to succeed. We're much more than code monkeys or a feature factory. Our team helps companies **design, develop, and deploy** great software.

When you work with us, we'll help you make sure:

- your software solves the right problems
- your team is staffed with the right people
- you using the right processes
- you build the right products

How do we do that? It starts with a simple, 30-minute conversation. Sure, we offer things from staff augmentation and custom development, to code audits and project planning. But each project is unique. We want to make sure you're set up to take the best next step for you and your project. If you're ready to build what you're envisioning a lot more efficiently — and with a lot less stress — let's talk and figure out how to turbocharge your project.

[Book my free 30-min Launch Call](#)