

One dumpster fire at a time, please!



Did we just say dumpster fire? Why, yes, we did just say dumpster fire. If you raised your eyebrow at this moment wondering what precisely a dumpster fire is:

**Dumpster fire** (*dʌmpstər ˈfaɪər*) –

A recurring daily emergency that never gets fixed. Something stupid keeps happening.

Oh, now you know what we're talking about! If you identify empathetically with this term, as in been there – done that (or still doing that), so do we. Our rockstars here at Lighthouse Technologies have some tips to become the fire extinguisher instead of the fire starter.

**Pro Tip:** [Contact us for a free 1hr consultation](#). -Sometimes the answers are quick and easy.

## Identify and reduce technical debt

If you are feeling a sense of déjà vu, it's for a good reason. We have mentioned technical debt a few times recently, as it's often a key contributor to someone's hair catching fire in a project.

How familiar are you with the [technical debt](#) left behind as you complete each sprint? If you are skipping this critical step, STOP! Your technical debt can grow with each subsequent sprint and build upon one another. Before you close the book on a sprint, you want to identify all remaining bugs and missed requirements. Once identified, focus on clearing it out.

That means go through it all (each time). Prioritize what to address now and what is acceptable to allow to continue, but not without a plan on solving it. Finally, DO IT. Don't talk about 'one day,' don't file it away in your imaginary to-do list, and don't sweep it underneath the carpet. Otherwise, you face the consequence of the dumpster fires that will soon haunt every sprint from this point forward.

## Performance and stability always come first

When it comes to sales and product, resist the pressure to press on and ignore instability problems and bugs. If the software isn't stable and not performing well, then you've unimpressed your customers before they even got a chance to see what's new and improved. Your customers WILL abandon before they can be dazzled.

The same goes for new functionality, for the rules still apply. What good is it to have a new feature if the legacy functionality becomes unstable? Do you see the point?

It's okay to push your roadmap forward into the future to ensure your system is stable and performing as expected. With a proper accounting of your technical debt, you can present this change to your client and senior leadership team without incident.

At-risk clients will appreciate your approach. The clients not at risk won't care and probably won't even notice.

**Reliable performance and stability = less technical debt = fewer dumpster fires**

## Business requirement details are critical

Business requirements are another crucial factor that either ignites or extinguishes dumpster fires. The reason is that robust requirements can often prevent technical debt from occurring.

When left vague, way too much is left to interpretation with no guarantees that everyone speaks the same language. A strong business analyst knows this and understands the impact potential.

If you fear that your business analyst lacks wisdom and experience to capture the details you need, or worse yet, you don't have one at all, and you're capturing requirements on the fly (as in a less than methodical way), you have another resource. Have testers review your business requirements before you add them to a sprint, while there's still time to make adjustments.

[Lighthouse Technologies](#) can review your requirements before that sprint and bring gaps and questions to light ahead of time. It's amazing what just one or two Lighthouse testers can bring to the team along the line of foresight. You will notice that more functionality completes consistently, with fewer defects at the end of the sprint.

Even better, **we see that this approach saves companies more money than the cost of the additional testers** because whole teams aren't redirected towards emergencies and mid-sprint changes.

You'll next realize the added benefit of watching validation and production releases soar as a result of better requirements. Robust business requirements enable your testers to write better test cases to those requirements, which creates superior traceability.

**Better requirements = better test cases = better reliability = less technical debt = fewer dumpster fires**

## Priorities set in stone

If you're saying, "yeah, right," stick with me a little longer. If you feel like your priorities are always a moving target, that is a telltale sign that your business requirements have significant gaps. Before you hop on that merry-go-round of the priority struggle again, circle back a little farther to your business requirements before you launch the next sprint.

Set priorities that won't always change by making sure they are nested in complete and robust business requirements.

**Another pro-tip:** If a dollar value is associated with each detail or task and it's prioritized, it will be highly unlikely to change what you do daily. The result you get to enjoy is that projects don't change as much, the scope is no longer a moving target, and everyone has a better understanding of the finish line.

**Better requirements = static priorities = better test cases = less technical debt = fewer dumpster fires**

## Let's bring it all together

Are you starting to see the clouds part for that ray of sunshine? Imagine your smile as the sun shines on your face because everything comes together as it should. We have a cluster of blogs all in the family of "hair on fire," and you'll see technical debt and business requirements mentioned each time. At the same time, these topics are often glossed over in real life.

2020 is knocking at your door right now. What positive changes can you make to enhance your efficiency, productivity, and effectiveness? Lighthouse Technologies is at your service to help you get pointed True North with the help of the [Lighthouse True North Assessment](#). We are that good at what we do because we practice what we preach on our own teams. We bring our best practices to every client, leaving behind new skills and ideas to improve efficiency going forward.

How is your project lineup looking for next year? We'd love to talk about how Lighthouse brings value to your development projects with testing and beyond. [Contact us](#), and let's get started!