Rollbar
The Hidden Costs Of Undetected Errors In Your Code

Highlights from our 2021 report "The State of Software Code Report"

88%

of developers acknowledge that bugs and errors are often still reported by users first... despite costly investments into process, talent, and tools that are supposed to prevent exactly that.

How often do errors go undetected... and how are they actually discovered?

Nearly 90% of developers admit that the error monitoring tools they use fall short.

1 in 4 developers say they detect errors by waiting until someone complains.

2/3 of developers have found out about errors through in-app reports submitted by users.

26% of developers have found out about an error by users complaining on social media.

21% of developers have heard about errors directly from their CEO... yikes!

Current error monitoring tools just aren't solving the problem of detecting hidden bugs.

Here's what developers say is missing:

- 39% of developers say they need to manually respond to errors.
- 36% say it takes too long to find the information they need to fix something.
- 31% say tech is too focused on system stability and not enough on code health.
- 23% say error aggregators too broad to narrow to be helpful.

So, what's the business impact of undetected bugs scampering around in your code base?

Revenue at risk

- 34% of developers say losing users is the biggest risk of errors in their software applications.
- 26% have actually reported losing a significant amount of users to errors.

Reputation damage

- 20% say errors damage their reputation and ability to fundraise.
- 18% say it angers investors.

Releases derailed

- 35% of respondents say their team is being held back from deploying more often.
- 32% say fixing errors slows them down and keeps them sticking to the production schedule.

Innovation suffers

- 33% nearly one-third of developers are spending up to 10 hours a week fixing bugs instead of writing code.
- 55% say they would rather see new features and functionalities if they weren't fixing bugs.

How does having to deal with escaped bugs affect developers on a personal level?

- **31% Frustrated** — Nearly one-third (31%) of developers say manually responding to errors makes them feel frustrated.
- **22% Overwhelmed** — 22% say losing time and focus to errors makes them feel overwhelmed.
- **17% Burned Out** — 17% say it makes them feel burned out.
- **12% Resentful** — 12% say it makes them resentful.
- **1:10 Ready To Quit** — Nearly 1 in 10 developers say that dealing with bugs makes them want to quit their jobs. Developer attrition comes at a high cost of sourcing, onboarding, and training new talent, making this a costly threat.

Would you rather...?

1/3 — More than 1/3 of developers said if they didn't have to spend so much time fixing code, they'd have more time for their wellbeing by...

- 67% Spending more time with their family
- 35% Exercising more
- 31% Getting a full night's sleep

In fact, 2/3 of developers would rather do an unpleasant activity than fix errors, such as:

- 26% Pay bills
- 21% Go to the dentist
- 20% Visit with in-laws

So, what can be done to make error tracking and triage less painful and time consuming for developers?

86%

of developers say they need better tools to detect and fix errors so they can iterate faster.

Here's what developers say would boost productivity and efficiency in their error monitoring tools:

- **47% See where errors originate** — Instead of spending time sifting through logs to track down the source of an error, access to location data makes tracking the error much faster.
- **34% Real-time capabilities** — One-minute snapshots and error rate charts can obscure important signals that can help developers head problems off at the pass.
- **33% Access to rich contextual information surrounding each error** — Nullam quis risus eget urna mollis ornare vel eu leo. Nullam quis risus eget urna mollis ornare vel eu leo.
- **33% Aid in quality assurance** — Testing and QA takes too long and developers want to move faster while improving accuracy before releases.
- **33% Open API access** — An open API allows development teams to build custom error monitoring features for their unique use case and business case.

Other factors holding back deployments:

- 30% say their team is too small
- 36% say testing/QA takes too long
- 38% want improved project management
- 16% say budget is too small

Rollbar

Rollbar is the continuous code improvement platform that proactively discovers, predicts, and remediates errors with real-time AI-assisted workflows. With Rollbar, developers continually improve their code and constantly innovate rather than spending time monitoring, investigating, and debugging.

Learn more at Rollbar.com

Survey Methodology
This name is survey of 950 developers and engineers was commissioned by Rollbar and conducted by Propeller Insights, an independent research firm between December 22 and 30, 2020. Propeller Insights uses quantitative and qualitative methods subject to analyze marketplace and consumer opinions. Survey respondents were nationally representative of the U.S. population for age, gender, region, and ethnicity.