

Improve Software Quality by Building Digital Immunity

By Joachim Herschmann, Senior Director Analyst

Improve Software Quality by Building Digital Immunity

Published 22 January 2021 - ID G00735246 - 12 min read

By Analysts [Joachim Herschmann](#)

Initiatives: [Applications and Software Engineering Leaders](#)

Traditional, testing-focused software quality approaches fail to deliver innovation quickly and lack the agility to quickly respond to defects. Software engineering leaders must take three steps to move beyond their approach to software quality and build digital immunity.

Overview

Key Findings

- Software engineering leaders who limit their development strategy to just building according to specification set their teams up for failure, with no validation that a product actually provides value to the customer.
- Software engineering leaders struggle to plan for all eventualities of how modern, highly distributed software systems may fail, resulting in poor ability to quickly remediate software defects and avoid impact on users.
- Software testing that requires human interaction doesn't prevent bugs from escaping into production or fixes them, leading to additional toil and waste for application teams.

Recommendations

Software engineering leaders responsible for software quality improvement should:

- Create a vision for digital immunity by optimizing delivery teams on customer value and customer experience (CX).
- Build digital immunity by adopting the five key practices that prevent and efficiently rectify negative customer impacts.
- Replace inefficient testing practices by eliminating waste in development and investing in autonomous testing.

Strategic Planning Assumption

By 2025, organizations who invest in building digital immunity will increase customer satisfaction by decreasing downtime by 80%.

Introduction

Cloud, social and immersive computing scenarios have raised end-user expectations for application quality and delivery considerably. “Quality” now entails user experience ^{1,2} quality of service, availability, performance security and even business impact. In fact, Gartner’s 2020 Digital Business Platform survey found that over three quarters of respondents said their team had responsibility for generating revenue.

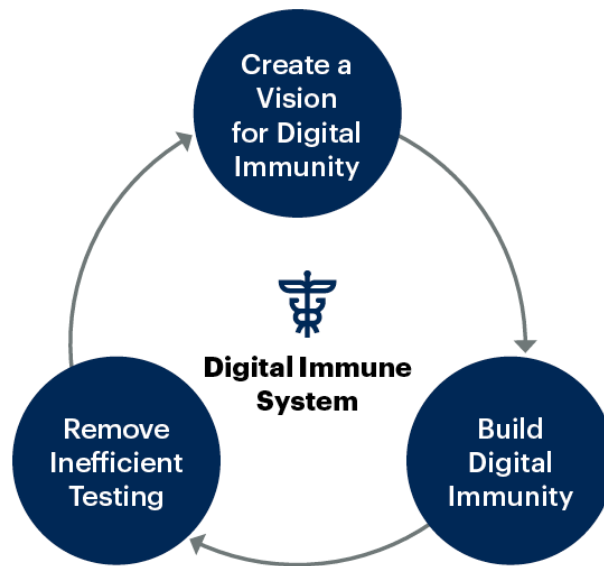
Many application and software engineering leaders are inadequately equipped to fulfill these expectations because their teams rely on antiquated development and testing approaches. In these cases, teams lack the skills to build robust and resilient applications, deal with unexpected failures, and deliver value faster than they create technical debt.

As a consequence, they cannot handle the vastly accelerated rate of change or the complexity of systems and fail to meet customers’ expectations. This exposes organizations to operational and business risks when applications and services are severely compromised or stop working altogether.

Software engineering leaders are looking for new practices and approaches that their teams can adopt to mitigate these risks and deliver at a high level of business impact. The digital immunity concept provides such a roadmap. Start by creating the vision, building digital immunity and removing inefficient testing (see Figure 1).

Figure 1: Digital Immune System

Digital Immune System



Source: Gartner
735246_C

Gartner

Analysis

Create a Vision for Digital Immunity

The fact that software has been tested creates a false sense of security, leading to the idea that “we have followed the process, so things must be okay. And if they aren’t, well, we can’t be blamed because we did what we were asked to do.” That is the wrong mindset.

Following the process, building to the product’s specification and running tests to verify, won’t guarantee that the product will function, let alone be loved, used often or fulfill all its desired outcomes. These processes do almost nothing to validate that it is of value to the customer.

Instead, software engineering leaders must focus on what constitutes a compelling user experience (UX). Don’t just ask “Does it work?” but “How could it work better?” and “How could it be more resilient?” This requires a mindset of innovation and a shift toward building quality into the product (see [Innovation Insight for Continuous Quality](#)) and plan for the reality that defects will slip through into production software.

Here is an example of a vision statement for building digital immunity:

We are fostering a quality culture and ensuring that teams are creating a superior UX that will not be compromised by defects or system failures. We do this by building systems that embrace failure as a natural occurrence even if we don't know what the failure might be or when it may occur. We are continuously improving our skills of minimizing the impact of a failure occurrence, with a focus to keep the overall system health intact.

A powerful vision statement such as this creates a frame of reference for defining the implementation strategy for building digital immunity. It helps to align the organization and initiate actions to implement the vision such as:

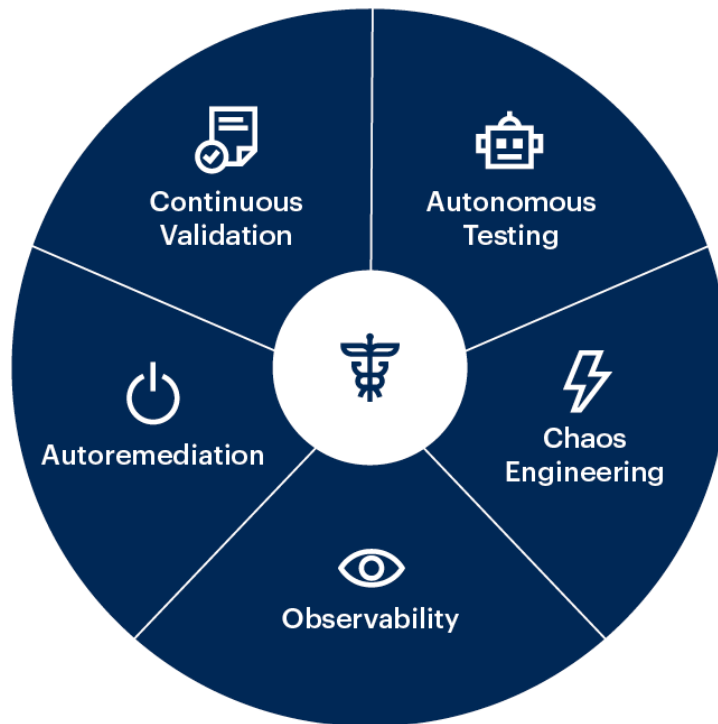
- Moving the organization from the traditional application- or project-centric model of quality to a holistic quality approach by adopting an ecosystem-centric view of quality.
- Infusing quality in every step, from the inception of an idea through to operations, by building links between CX, multiexperience, UX and employee experience (see [Build Links Between Customer Experience, Multiexperience, User Experience and Employee Experience](#)).
- Allocating ownership and appointing staff with the required skills needed for building resilient applications, by identifying the required roles, technologies and practices.

Build Digital Immunity

To develop digital immunity, software engineering leaders must empower their teams to adopt the following practices (see Figure 2).

Figure 2: Elements of the Digital Immune System

Elements of the Digital Immune System



Source: Gartner
735246_C

Gartner

Autonomous Testing

Autonomous testing comprises artificial intelligence (AI)- and machine learning (ML)-based technologies and practices to make software testing activities independent from human intervention. It continuously improves testing outcomes by learning from the collected data from performed activities. It extends traditional test automation beyond the automated execution of test cases to include fully automated planning, creation, maintenance and analysis of tests (see [Innovation Insight for Autonomous Testing](#)). Example vendors include Diffblue, ProdPerfect, TestModeller and Testsigma.

Autonomous testing is the integrated automation of all testing activities related to requirements quality, design quality, code quality, release quality and operational resilience. It extends beyond automation by providing orchestration, as well as independence and autonomy for those activities.

Chaos Engineering

Chaos engineering is the use of experimental and potentially destructive failure or fault injection testing to uncover vulnerabilities and weaknesses within a complex system (see [Innovation Insight for Chaos Engineering](#)). Much like attacking the immune system with a controlled injection of an attenuated virus, we can use chaos engineering to train an organization to deal with bugs and system failures.

Chaos-engineering practices are relatively new, but are an important part of the arsenal of high-performing teams. Gartner’s Achieve Business Agility With Automation, Continuous Quality and DevOps survey found that 18% of participants were using or planning to use chaos engineering. Example vendors include ChaosIQ, Chaos Monkey, Gremlin and Litmus.

Software engineering leaders must enable their teams to start with chaos engineering in preproduction environments so they can safely master the practice in a nonintrusive and test-first manner. The lessons learned can then be applied to normal operations and production hardening (see [How to Safely Begin Chaos Engineering to Improve Reliability](#)).

Autoremediation

Chaos engineering helps us to learn more about the effects of software bugs and the potential failure points of a system. A consequential next step is to build context-sensitive monitoring capabilities and automated remediation functions directly into an application. Such a system not only monitors itself, but also corrects issues automatically when it detects them and returns to a normal working state without requiring the involvement of operations staff (“self-healing”).

Table 1 lists common scenarios for remediation actions that address common issues in today’s environments.

Table 1: Autoremediation Scenarios

Defect or Issue	Remediation Action
Bad configuration change	Revert to previous configuration
Full or slow disk	Clean up
Application crash	Restart
End-user impact	Reverse blue/green
Low resources	Scale up or scale out

Source: Gartner (January 2021)

While these scenarios have been known for years by many teams, few have invested in completely automating them. Scripts for these scenarios may exist, but they are executed manually when a problem is detected.

Software engineering leaders must encourage their teams to fully automate the entire process from the moment an anomaly is detected to the point when the issue is resolved, so that no human intervention is required. Value-add is easily found if systems remain usable during impacting events and human intervention is not required.

Observability

Observability is the characteristic of software and systems that allows them to be “seen” and allows questions about their behavior to be answered (see [Innovation Insight for Observability](#)). To fully realize the promise of modern development methodologies, applications must be built with “observability-driven development.”

The key for observability tools is to spot the anomaly that is relevant to the problem at hand, and then to link other bits of information from log files/metrics that are likely to be related. By surfacing correlated information in context, the operator can more quickly isolate the potential root cause of problems. Example vendors include AppDynamics, Dynatrace, New Relic and Splunk.

Software engineering leaders must work with their infrastructure & operations (I&O) peers to enable observability by agreeing on utilizing emerging open standards for collection, such as OpenTelemetry and OpenMetrics. They can help to increase application uptime by training their teams to design observability directly into the application and its supporting infrastructure.

Continuous Validation

Autoremediation is a necessary element for building a highly resilient system. But it is still reactive in the sense that it triggers only when an anomaly is detected. This raises two questions: “When is an anomaly detected?” and “Is it affecting a user?”

In some scenarios, a user may identify that there seems to be a bug in the system. But there are countless other scenarios where an error may not be identified as an issue until a month-end or quarter-end reconciliation is performed. By this time, system impairment or data corruption could be very widespread.

Continuous validation involves building industrial-strength services that continuously monitor the integrity of data and systems in a live environment. The key point is to actively search for data inconsistencies and anomalous system behavior, not to wait for an unfortunate user to detect them or it (see [Avoid Digital Quality Disasters With Continuous Autonomous Validation and Verification](#)).

When a bug (dormant or active), a data inconsistency or an anomalous event is detected, it should not only trigger a corresponding incident response. It should also activate the autoremediation procedures discussed earlier to prevent the effects of the bug spreading further, to restore service and to mitigate the impact on the user. The point is that the issue has been discovered much earlier as part of a *proactive*, continuous validation mechanism, so that the “blast radius” of the error is much smaller.

Replace Inefficient Testing Practices

Faster delivery of customer value through a continuous flow of software deployed in production environments is at the core of DevOps. Increasing velocity requires DevOps teams to identify and remove the greatest constraint (see [6 Steps to Increase DevOps Release Velocity by Removing Constraints](#)).

DevOps teams can often identify the greatest constraint by simply asking team members “Who are you always waiting on?” In many organizations, the answer is that testing is the greatest constraint, often because of a high ratio of manual testing. Software schedules for major applications are about 25% longer than they should be, due to poor-quality, expanding testing intervals. ³

Evaluate the current state of manual tests and determine the most effective approach for transitioning these testing assets to increasingly autonomous levels of automated testing. Note that the existing structure of a manual test may or may not be suited for automation. However, relevant components of existing manual tests (such as input values, expected results, navigational path and test data) can be reused as input for autonomous testing.

Evidence

Gartner has had more than 1,000 client interactions (inquiries and one-on-one meetings) discussing testing best practices, initiatives, successes and failures, from 2019 to present.

Gartner’s 2020 Building Digital Platforms Survey was conducted to provide guidance on how to build a digital initiative. The research was conducted online during May and June 2020 among 206 respondents working for organizations in North America and Western Europe with at least \$1 billion in annual revenue. The respondents’ organizations are active in a range of industry sectors, as detailed in Table 2.

Table 2: Industry Demographics of Survey Respondents

Enlarged table in Appendix

Industry	Percent
Manufacturing and natural resources	20%
Banking/investment services	20%
Healthcare providers	17%
Retail	14%
Communications	9%
Insurance	8%
Transportation	6%
Utilities	3%
Services	2%
Media	1%
Q. What is your organization's primary industry classification?	

Source: Gartner 2020 Building Digital Platforms Survey

Respondents' organizations were required to be working on digital business efforts or planning to do so. "Digital business" is defined as involving the Internet of Things (IoT), delivery of public APIs, private/B2B APIs or a combination thereof.

Quotas were set to ensure a majority of respondents had a fully implemented digital business initiative.

Respondents were required to have a job title of director or more senior, and to be involved in either digital business, data analytics, IoT- or API-based platforms for partners. In respect to digital business initiatives, respondents were also required to have a role in either defining technology requirements, investigating or evaluating service providers, or making final decisions.

The results of this study do not represent global findings or the market as a whole, but reflect the sentiment of the respondents and companies surveyed.

Gartner's 2020 Achieve Business Agility With Automation, Continuous Quality and DevOps survey was conducted online from June through August 2020, among 205 respondents working for service providers, cloud providers and end-user organizations in North America and Western Europe that have deployed or are using DevOps.

Qualified organizations had at least \$500 million in annual revenue and were required to primarily operate in the banking and financial services, government, insurance, healthcare and retail industries.

Respondents were required to work in their organization's IT function, have a job title less senior than C-level, and be two or more layers away from the most senior executive in their organization.

The respondent's role had to be primarily focused on application development, infrastructure and operations or business intelligence and information management. In these focus areas, they were also required to perform relevant roles/activities.

Results of this study do not represent global findings or the market as a whole, but reflect the sentiments of the respondents and companies surveyed.

1. M. Hassenzahl and N. Tractinsky, "User Experience – A Research Agenda," Taylor & Francis Group, March – April, 2006.
2. J.J. Garrett, "The Elements of User Experience," Pearson Education, December 2010.
3. C. Jones. "Wastage: the Impact of Poor Quality on Software Economics," version 8:0. 3, September 2017.

Acronym Key and Glossary Terms

AI	artificial intelligence
CX	customer experience
IoT	Internet of Things
ML	machine learning
UX	user experience

Recommended by the Author

[Build Links Between Customer Experience, Multiexperience, User Experience and Employee Experience](#)

[Innovation Insight for Continuous Quality](#)

[Innovation Insight for Autonomous Testing](#)

[Innovation Insight for Chaos Engineering](#)

[Innovation Insight for Observability](#)

[Use AIOps for a Data-Driven Approach to Improve Insights From IT Operations Monitoring Tools](#)

[Maverick* Research: Software Testing and the Illusion of Exterminating Bugs](#)

[Infographic: Artificial Intelligence Use-Case Prism for Software Development and Testing](#)

Recommended For You

[Applications and Software Engineering Leaders Primer for 2021](#)

[Hype Cycle for Software as a Service, 2020](#)

[Executive Summary Video: Understanding Increased MIPS After a Mainframe Modernization](#)

[What to Expect When You're Expecting Digital Twins](#)

[What Should I Do To Ensure Digital Twin Success?](#)

© 2021 Gartner, Inc. and/or its affiliates. All rights reserved. Gartner is a registered trademark of Gartner, Inc. and its affiliates. This publication may not be reproduced or distributed in any form without Gartner's prior written permission. It consists of the opinions of Gartner's research organization, which should not be construed as statements of fact. While the information contained in this publication has been obtained from sources believed to be reliable, Gartner disclaims all warranties as to the accuracy, completeness or adequacy of such information. Although Gartner research may address legal and financial issues, Gartner does not provide legal or investment advice and its research should not be construed or used as such. Your access and use of this publication are governed by [Gartner's Usage Policy](#). Gartner prides itself on its reputation for independence and objectivity. Its research is produced independently by its research organization without input or influence from any third party. For further information, see "[Guiding Principles on Independence and Objectivity](#)."

Table 1: Autoremediation Scenarios

Defect or Issue	Remediation Action
Bad configuration change	Revert to previous configuration
Full or slow disk	Clean up
Application crash	Restart
End-user impact	Reverse blue/green
Low resources	Scale up or scale out

Source: Gartner (January 2021)

Table 2: Industry Demographics of Survey Respondents

Industry	Percent
Manufacturing and natural resources	20%
Banking/investment services	20%
Healthcare providers	17%
Retail	14%
Communications	9%
Insurance	8%
Transportation	6%
Utilities	3%
Services	2%
Media	1%
Q. What is your organization's primary industry classification?	

Source: Gartner 2020 Building Digital Platforms Survey

Become a Client

Get access to this level of insight all year long — plus contextualized support for your strategic priorities — by becoming a client.

gartner.com/en/become-a-client

U.S.: 1 800 213 4848

International: +44 (0) 3331 306 809

About Gartner

Gartner is the world's leading research and advisory company and a member of the S&P 500. We equip business leaders with indispensable insights, advice and tools to achieve their mission-critical priorities today and build the successful organizations of tomorrow.

Our unmatched combination of expert-led, practitioner-sourced and data-driven research steers clients toward the right decisions on the issues that matter most. We are a trusted advisor and an objective resource for more than 14,000 enterprises in more than 100 countries — across all major functions, in every industry and enterprise size.

To learn more about how we help decision makers fuel the future of business, visit gartner.com.