ACCELQ

Right tool for Low-Code Apps

Ensure the success of your Low-Code initiatives with the right Test Automation tool

20 MINS

PUBLISHED ON 29.8.2022

WWW.ACCELQ.COM

Executive Summary

A low-code initiative can be a transformative strategy for IT teams, especially in the face of ever-changing customer demands. But if it's not meticulously planned, a low-code project risks becoming a high-risk, high-cost endeavor. Several factors affect a successful rollout and sustenance of a low-code initiative. A well thought out test automation strategy is one of the key factors. Apart from the overall Quality Assurance strategy, identifying the right automation tool is extremely critical. And this can be daunting, with many factors to take into account and numerous vendors to consider. Nevertheless, when done correctly, the results are positive across the board.

Introduction

Organizations have invariably relied on in-house and outsourced development for decades. While these approaches have their benefits, they are not without drawbacks. In-house development excels at custom tailoring of business-fit solutions but is marred by high costs, slow turnaround time, and limited technical expertise. The skill gap, in particular, is magnified in small- and medium-sized businesses, for whom hiring someone with exceptional tech expertise is often impossible or comes at a prohibitively high cost. McKinsey reports that more than 87% of the organizations surveyed are currently troubled with the skills gap as compared to only 6% that don't see it as a problem, at least for the next ten years.

Outsourcing, on the other hand, provides a high level of efficiency but has issues with quality, workmanship, and timelines. Perhaps the biggest problem is the inherent lack of process-oriented thinking, which many organizations have already realized. More or less, it comes to technical and managerial silos, hinting yet again at talent gaps – a problem that is arguably more acute than ever and perhaps accurately testified by the recent Korn Ferry report. The consulting giant warns of $8.5 trillion in lost business by 2030 due to the same.

So, in an utterly overcrowded market, low-code initiatives seem to be the strategic driver of choice for many. They put business users and citizen developers in charge of their projects, giving them the freedom to define requirements, solve issues, and deploy new applications with minimal oversight. But low code isn't a magic pill against the talent problem. It's more of a powerful mechanism to foster fast-paced user-driven innovation and development and create unprecedented business agility and responsiveness.

The Soaring Popularity of Low-Code Initiatives

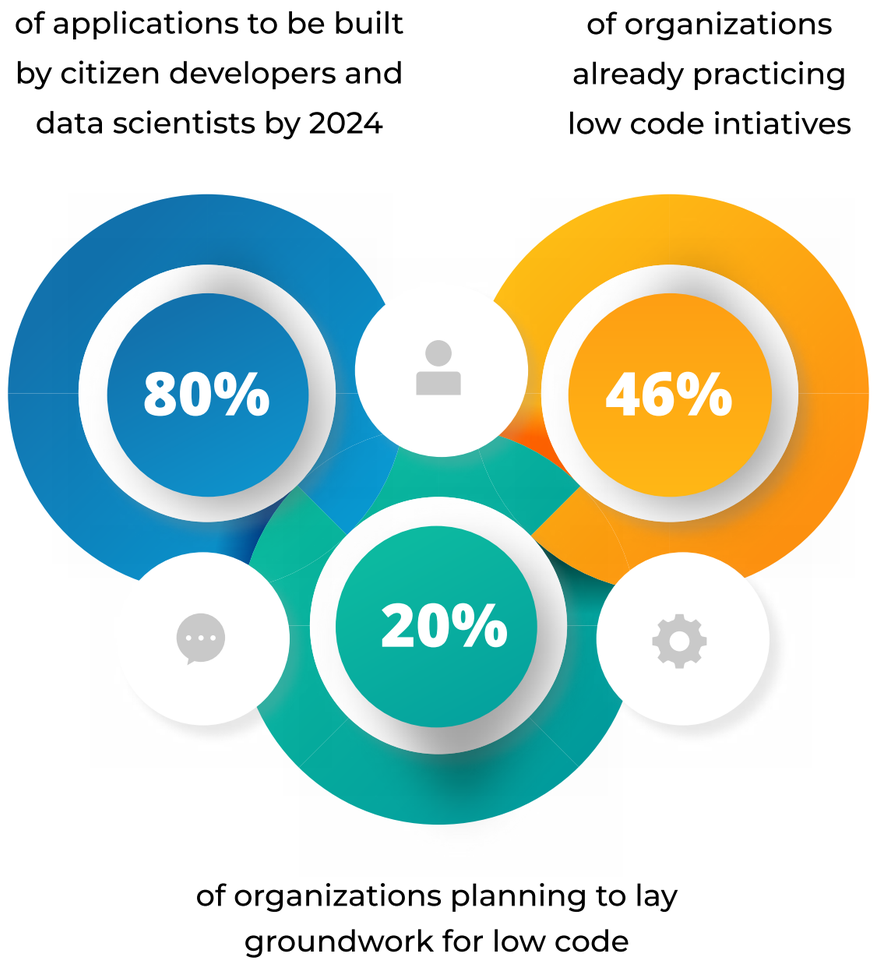
The slick, easily understandable interfaces and layouts facilitated by low-code work wonders, empowering users with the confidence to learn quickly and create, test, and deploy solutions on the fly. The result is rapid, iterative production tailor-made for business needs, delivering timely, value-based applications at a fraction of the traditional cost.

Statista estimates the low-code platform (LCAP) market to be worth $65 billion by 2027, up from $13 billion in 2020. This is in line with Gartner's interpretation of the market surge, with the research giant touting more than 2/3rd of the new applications to be born out of LCAPs by 2025. That's a considerable growth rate, given that only 1/4th of the applications existing in 2020 followed the pattern.

Benefits Galore With Low-Code

Lower Entry Barrier

Gartner predicts that business technologists, including citizen developers and data scientists, will build and test 80% of all new enterprise applications by 2024. This inclination toward citizen development initiatives points to unparalleled democratization of the application lifecycle. Even before the Covid-19 pandemic fueled the citizen development revolution, the advent of low-code applications was instrumental in lowering the barrier to entry for small- and medium-scale enterprises. In fact, in 2019, 46% of the organizations were already practicing low-code initiatives, while another 20% were planning to lay the groundwork. At present, low-code has leveled the playing field for tech-first businesses to enter the non-tech market (financial services, retail, etc.) with greater ease.



Decreased Costs

Since low code helps achieve business-fit applications through rapid prototyping, the long-term cost of development and testing is far less. Consider this; the development time for low-code applications is reduced by 50-90%, paving the way for addressing project backlogs, which otherwise incur billions of dollars in losses. Another way to look at cost reduction is from the automation and point-and-click convenience gained from mere drag-and-drop functions. This is an unbeatable advantage for businesses with a limited pool of resources because they don't need to aggressively invest in hiring amidst the talent crunch

Better Integration & Change Management

Often, legacy applications drive a culture of complacency and stagnation that greatly inhibits change. The resistance to change and the lack of a business-fit solution for modern use cases is the proverbial double-edged sword for businesses. Low-code initiatives, however, thrive on rapid technological change, thereby allowing for seamless integration with the new generation of applications. Favorably, they can integrate with legacy systems just as well, thus mitigating the risk of undertaking a costly rewrite, at least until the legacy system becomes obsolete and the business decides to terminate it.

With such benefits in tow, it's no surprise that low-code initiatives are becoming an essential component of enterprise architecture plans. They are on the path to significantly contributing to IDC's prediction of 500 million digital apps hitting the market by 2023.

Use Cases of Low-Code Platforms

Considering that "low-code" is in itself a development practice and not solely an intervention, the use cases aren't necessarily restricted to a particular vertical. However, there's always a process that highlights the potential of low code, regardless of the industry it's used in. Upon granular exploration of this process, it's possible to arrive at a conclusive understanding of the ideal use cases.

Driving Automation

Perhaps the most crucial aspect of the business is the ability to automate processes, leading to efficiency, effectiveness, productivity, and profitability improvements. Favorably, low-code versions can creep up the process mapping, which may contain numerous customizations, thereby narrowing down the automation paths that are most cost-effective and time-efficient. In turn, this may give birth to new business models and orchestrate other changes that would otherwise have been unthinkable due to high initial costs.

Tackling Microservices Complexity

More than 75% of companies with 1000+ employees attest to using microservices. However, the monolith-breaking microservices aren't always a boon. They're riddled with complexities that can paralyze the development process because there's a lot of code to review and maintain. As such, singling out and identifying the complexities, either outdated or just not entirely business-fit, is daunting. Low-code applications' reliance on common code, backend tool configurations, and a simplified UI make it easier for business technologists and seasoned developers to clean up, refactor, and maintain their microservices.

Supporting IoT Networks

IoT spending will reach $1.1 trillion in 2023, meaning IoT deployments will become a significant part of investment portfolios. However, harboring massive IoT networks entails maintaining enormous data lakes, which is inherently challenging. The problem is worsened when this data exists in silos, i.e., non-linked and non-interoperable sources. To that end, LCAPs are strategic fits for enterprises that need to tie the dots between their distributed IoT networks, for they consolidate data in a central repository and simplify the ways to integrate them.

Powering Data Analytics

About 80-90% of data available for analytics is unstructured, with disparate formats and varying degrees of sophistication. Low-code platforms can help parse this data, identify datasets of interest, and provide a unified view for the analyzer — through intuitive visualizations. Altogether, these initiatives work to optimize the end-to-end analysis workflow. In the process, they open up greater opportunities for the democratization of analytics and data discovery, thereby making their way into mainstream businesses with a customer-facing focus.

Enhancing Web-Based & Mobile Portals

Many self-service solutions are born of the desire to ease the user's mental effort and expedite instant solutions. Such consumer-facing applications, albeit minimalist in scope, present challenges for enterprises with a high volume of transactions, such as those within eCommerce. In this regard, low-code initiatives could be targeted at standardizing content management, supporting headless, and reducing the need for non-differentiated commercial-off-the-shelf (COTS) tools.

Why Low-Code Applications Need Test Automation

Since low-code stacks are about rapid and comfortable development, they must be paired with equally rapid testing. A big reason for this is the relatively fast release cadence of low-code applications compared to traditional enterprise applications, especially in light of agile, DevOps, and lean development methodologies.

However, the need for speed is just one of the reasons why low-code applications need test automation. The problem might run deeper than that, as low-code applications are expected to deliver a variety of use cases, with each being tailored to specific business requirements. Hence, a lot of iteration and customization is involved (albeit point-and-click), which presents the need for continuous improvement and evolution. The following are some of the challenges that surface in that regard:

- **LCAP-Related Issues**: At times, the modules furnished by the LCAP vendor can contain bugs and errors, thereby jeopardizing the system's functionalities. Worse, the entire stack might have to be tweaked if business-critical modules are affected. As such, low-code deployments need to be accompanied by a comprehensive automated test strategy that guarantees that the systems work as interpreted and advertised.

- **Performance Metrics:** Performance issues are the cautionary tale of every digital application, irrespective of the technology stack on which it's based. But with low-code initiatives predominantly targeted at rapidly bringing out minimum viable products (MVPs), performance issues can be particularly vexing owing to the low priority lent to performance optimization.

- **Vendor-Specific Constraints:** Maybe the vendor didn't configure the LCAP for a particular use case; however, the developers proceeded based on the assumption that it would. This might lead to an array of issues scratching off, which is somehow difficult to predict and address in time.

- **Citizen Development Limitations:** No doubt, citizen development initiatives suffice the business requirements well on the business and financial fronts. However, they leave much to be desired when it comes to professional quality checks.

- **Overlooked Integration Intricacies:** Businesses can have a wide range of front-end and back-end integration needs, specific to the application's use case. Since the LCAP may not have touched on all combination possibilities in real-world environment, it'd be difficult to predict and overcome the nitty-gritty integrations issues.

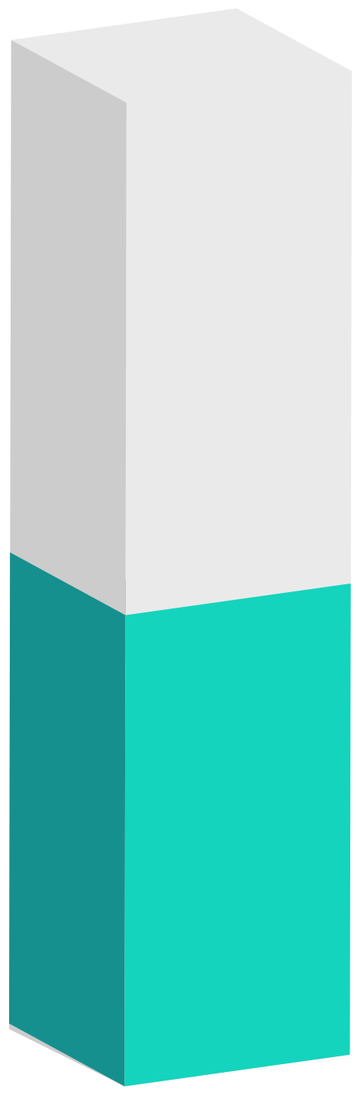
- **Compatibility Issues:** When a customer employs more than one LCAP solution to interact with business systems, they can expect a slew of compatibility issues. Besides, a series of interlaced releases across the LCAPs can further complicate the problem, for it's hard to predict the interdependencies and how they'll reflect on compatibility

All-in-all, low-code applications advocate for profound quality assurance (QA) involvement, which again is particularly crucial given the risks associated with their rapid release cadences and professional oversight. Of course, manual testing is simply not scalable or efficient enough to deliver as desired, meaning automated test suits need to be commissioned.

Nuances of Low-Code Application Testing

The traditional programming-based automation is inherently plagued with manually creating rather sophisticated test scripts. Unfavorably, while pursuing this approach, "programming-related" expectations are targeted at testers. They are expected to get their head around coding, albeit on a minimal scale, to run automation and make the needed adjustments to the test scripts.

But that isn't how test automation should be viewed. In fact, it's just the opposite of what the business needs. This way, introducing automation into testing increases resource needs and time, thereby affecting the bottom line. It's not repeatable and reusable and creates a gulf between what the business needs and what it gets. In addition, further test automation is bound to expose gaps in the current test coverage, impeding the progress of projects. This explains why 45.9% of the testing specialists keep "programming skills" at the top of their considerations while assessing the test automation tool for its effectiveness.



45.9%

of the testing specialists keep "programming skills" at the top of their considerations

This explains why low code application testing is highly admirable in the context of rapid software development and delivery. It can help lessen the development times, cut the release cycles, and boost the enterprise's agility quotient.

How To Evaluate a Test Automation Tool for Low-Code Application Testing?

About 442 low-code/no-code platform vendors suffice the modern-day market. So, it wouldn't be an overstatement to say that it's quite a challenge to choose a tool that is optimal for the enterprise's testing needs.

The good thing is that not all tools are created equal, for some might offer better functionalities than others, despite similar pricing schemes. No doubt, the tool's functionalities need to be evaluated at length but do remember that the product's ease of use and the vendor's support services are equally important. In other words, a myriad of factors must be weighed in.

Building on that, the following five-point checklist will prove helpful to the testing and QA specialists in ascertaining if a given automation tool is a good fit for their low-code application testing:

1 Is the tool truly Low-Code from the bottom up?

Often, test automation tools are marketed as low-code platforms, even though they cannot support and advance workflows beyond a certain point. Such shallow abstraction is a recipe for trouble and certainly a heavier burden on the development team and QA cadre.

2 Is the tool scalable and flexible?

What if you need to extend the core capabilities of the tool to accommodate the requirements of emerging use cases? What if you have already undertaken a few upgrades, but with the next upgrade, you need new sets of functionalities altogether? Assessing how (and if) the tool easily scales and caters to evolving needs should be a priority.

3 Is the tool centered on automation?

It's highly likely for test automation tools to actually not exhibit core automation capabilities and, instead, rely on integrations with other frameworks, APIs, libraries, and tools for support. Such a solution must be avoided at all costs, for it will only add to the development and maintenance overhead.

4 Is the tool supportive of external integration?

Besides keeping the core automation capabilities intact, the tool must also support a wide range of integration options with external LCAPs and modules. This capability is one of the primary reasons why most enterprises incline toward test automation in the first place because it allows them to tightly integrate and orchestrate data flows and functionalities across enterprise-wide systems.

5 Is the tool adaptable to change?

Because the tool will be tasked to complement the agile and DevOps strategies, it must be able to adapt to change easily. As such, it must exhibit a robust framework that will enable the teams to proceed with change management amidst the typical uncertainty that comes with disparate release cycles.

In addition to evaluating the test automation tool per this checklist, there's a nuanced way to assess whether a given product is technically worth the investment. We proceed by categorizing the capabilities of the tool into four buckets:

(1) Technical & Functional, (2) Accessibility, (3) Adoption, and (4) ROI.

Technical & Functional

- Automated test design
- Data-driven testing
- Service & API automation
- Test asset management
- Reusability & modularity

Accessibility

- Ease of Use
- Functional level test execution & analysis
- Centralized analytics & insights
- Ability to set up governance
- Flexible language

Adoption

- Ease of adoption
- Free from vendor locks
- No hidden costs
- Easy training and user onboarding
- Support

ROI

- Speed to value
- Manageable total cost of ownership (TCO)
- Seamless change management
- Low maintenance

Making a Difference with ACCELQ Live

Low-code application testing strategies need to focus on reusability, maintainability, and scalability, in addition to the regular testing metrics. That's exactly how ACCELQ Live has been making the most of codeless test automation. Following are a few reasons why it's doing so, in good measure:

More Power in the Hands of Business Users

Business user enablement is one of the most crucial aspects of low-code applications. Strictly speaking, the business users must be involved in every facet of SDLC, from conceptualization to deployment. ACCELQ manifests this by ensuring that business technologists are the driving force behind test automation of low code applications.

Speed and Reusability - the Keys to Success

When it comes to speed, numerous factors are involved, beginning with the scope of coverage to the test asset reusability and change management to vendor release alignment. ACCELQ Live makes a huge play on all of these by ensuring test assets befit the business-specific environments, change management is automated and perpetual, and every project simultaneously adheres to multiple SaaS vendor releases.

Innovation Support

End-to-end automation is the flag-bearer of innovation because it lends time and resources to "thinking out of the box." Apart from that, a low-code application testing tool must facilitate extendable capabilities to collaborate and innovate with software vendors and other partners involved. ACCELQ Live does exactly so and, thus, challenges the premise of disparate tool vendors solely leading the innovation race.

Scalability - the Holy Grail

When codeless test automation rests on one or two purpose-built scenarios, the overall scalability is bound to be compromised for the entire low-code application testing strategy. ACCELQ approaches this challenge by extending generic yet specific use cases - thanks to multi-cloud automation and multi-channel support.

ROI - the Ultimate Test

Business user enablement is one of the most crucial aspects of low-code applications. Strictly speaking, the business users must be involved in every facet of SDLC, from conceptualization to deployment. ACCELQ manifests this by ensuring that business technologists are the driving force behind test automation of low code applications.

Business user enablement is one of the most crucial aspects of low-code applications. Strictly speaking, the business users must be involved in every facet of SDLC, from conceptualization to deployment. ACCELQ manifests this by ensuring that business technologists are the driving force behind test automation of low code applications.

Conclusion

Low-code initiatives have been gaining momentum for a few years now, and rightfully so. Most of them yield swift, sustained, and actionable ROI. But without having an in-depth understanding of the responsible characteristics of low-code application testing, it's difficult to drive the right strategy. To put it in simpler terms, the general lack of understanding of low-code application testing is the biggest deterrent for organizations to embracing this promising approach. Favorably, the success of these testing strategies can be realized with the right test automation tool in place and a clear blueprint of the testing approach.
Altogether, to get your product up to par with the market, ensure it has the necessary functionalities and resources to support the low-code testing initiatives. It is arguably a lucrative investment for the enterprise and will ensure the business's long-term health.

The tips mentioned above should help you select the best low-code test automation platform and optimize its functionalities to deliver on a sound business case for your organization.

References

Gartner. (2021). Gartner says cloud will be the centerpiece of new digital experiences. www.gartner.com. Retrieved from https://www.gartner.com/en/newsroom/press-releases/2021-11-10-gartner-says-cloud-will-be-the-centerpiece-of-new-digital-experiences

Gartner. (2021). Gartner Says the Majority of Technology Products and Services Will Be Built by Professionals Outside of IT by 2024. Retrieved from https://www.gartner.com/en/newsroom/press-releases/2021-06-10-gartner-says-the-majority-of-technology-products-and-services-will-be-built-by-professionals-outside-of-it-by-2024

Gartner. (2019). The Importance of Citizen Development and Citizen IT. Retrieved from https://blogs.gartner.com/jason-wong/importance-citizen-development-citizen/

Korn Ferry. (2021). The $8.5 trillion talent shortag e. www.kornferry.com. Retrieved from https://www.kornferry.com/insights/this-week-in-leadership/talent-crunch-future-of-work#

IDC. (2019). IDC FutureScape: Worldwide IT Industry 2020 Predictions. www.idc.com. Retrieved from https://www.idc.com/research/viewtoc.jsp?containerId=US45599219

McKinsey & Company. (2020). Beyond hiring: How companies are reskilling to address talent gaps. https://www.mckinsey.com/. Retrieved from https://www.mckinsey.com/~/media/McKinsey/Business%20Functions/Organization/Our%20Insights/Beyond%20hiring%20How%20companies%20are%20reskilling%20to%20address%20talent%20gaps/Beyond-hiring-How-companies-are-reskilling.pdf

MIT Sloan. (2021). Tapping the power of unstructured data. mitsloan.mit.edu. Retrieved from https://mitsloan.mit.edu/ideas-made-to-matter/tapping-power-unstructured-data

Red Hat. (2018). Research intelligent process automation analyst paper - Red Hat. www.redhat.com. Retrieved from https://www.redhat.com/cms/managed-files/mi-451-research-intelligent-process-automation-analyst-paper-f11434-201802.pdf

Simform. (2022). State of Test Automation Survey 2022: Statistics that'll surprise you! www.simform.com. Retrieved from https://www.simform.com/blog/state-of-test-automation-report/#section1

Spreadsheet Web. (2021). How many low-code/no-code platforms are out there? www.spreadsheetweb.com. Retrieved from https://www.spreadsheetweb.com/how-many-low-code-no-code-platforms-are-out-there/

Statista. (2022). Low-code development platform market size worldwide 2025. www.statista.com. Retrieved from https://www.statista.com/statistics/1226179/low-code-development-platform-market-revenue-global/

Statista. (2022). Do you utilize microservices within your organization?. www.statista.com. Retrieved from https://www.statista.com/statistics/1236823/microservices-usage-per-organization-size/

Statista. (2022). Internet of Things (IoT) - statistics & facts. www.statista.com. Retrieved from https://www.statista.com/topics/2637/internet-of-things/#dossierKeyfigures

ACCELQ

ACCELQ is the only cloud-based Continuous Test Automation and Test Management platform that seamlessly automates API and web testing without writing a single line of code. IT teams of all sizes use ACCELQ to accelerate their testing by automating critical aspects of lifecycle like testdesign, planning, test generation and execution.

ACCELQ customers typically save over 70% of the cost involved in the change and maintenance efforts in testing - addressing one of the major pain points in the industry. ACCELQ makes this possible with an AIpowered core that brings self-healing and resilience as a critical dimension to test development capability

WWW.ACCELQ.COM

San Francisco | Dallas | Sydney