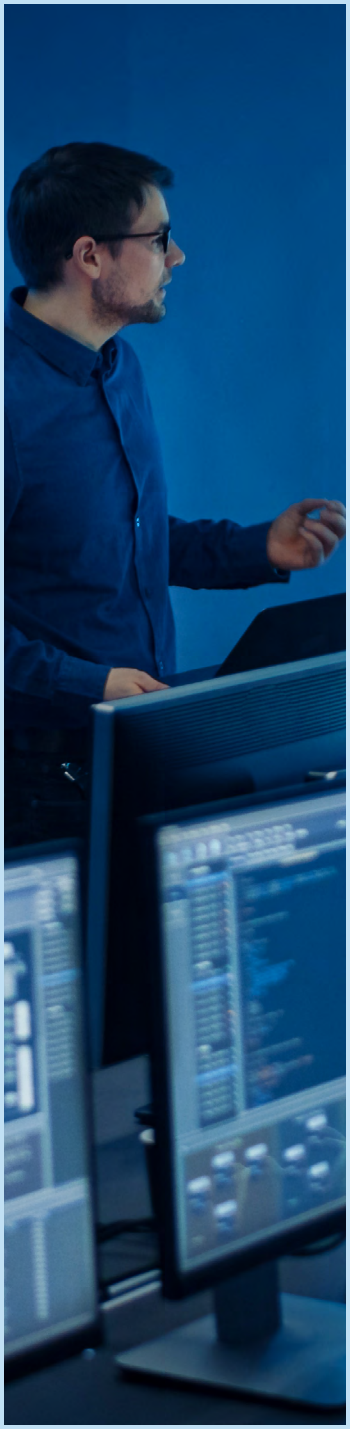
VIEW POINT



DE-RISK THE ADOPTION OF LOW CODE WITH FIT-TO-PURPOSE TESTING

Abstract

The year 2020 taught enterprises, and particularly the IT industry, the importance of building resilient applications and ecosystems. As we look to 2021, there are some disruptive trends such as the low code or no-code development methodology that are opening up new opportunities in software development and testing. A well-defined testing strategy for low code is needed to address risks and reduce maintenance costs. This paper explains how to validate low code applications using a 'fit-to-purpose' test strategy. It focuses on certain types of testing, specific to a low code development methodology. It also provides a differentiated approach from the traditional 'test it all' approaches that limit ROI and speed.

Infosys®
Navigate your next

Introduction

Low code application platforms (LCAPs) have simple drag-and-drop features, allowing even users lacking coding knowledge to develop and deploy different and complex types of applications faster than before.

Organizations have realized that low code application platforms offer faster time-to-market and access to the citizen developer community that increases their return on investment (ROI) compared to traditional application development methodologies (ADMs). This quantum leap in ROI can help businesses strengthen their financial position, particularly after the economic slowdown over the past few years. Industry experts predict that enterprises will adopt low code application platforms for more than 60% of software development activity over the next few years.

Risks in applications developed with low code platforms

Organizations can succeed in reaping higher ROI with lower risk of failures by adopting best-in-class quality engineering and industry-standard verification solutions. But first, they must be aware of the risks involved in application ecosystems developed using low-code methodologies.

The purpose of low code platforms is to enable collaboration and accelerate the speed of application delivery. They empower application developers with code templates, pre-built engineering features and reusable components to develop and deliver scalable apps. Low code development platforms assist developers of differing experience levels in creating applications for web and mobile using drag-and-drop components and model-driven logic through a graphical user interface (GUI).

Low code platforms are the fastest tool for application development. However, they have challenges related to extensibility, performance and maintainability. A closer examination of these challenges highlights how they differ from applications built using traditional application development methodologies.

- Low code applications meet basic requirements; however, any addition or customization to the architecture or design of components will take twice as much time to implement. The challenge here is in understanding **black box implementation**, its exposed interfaces and underlying integrations to extend the existing functionality to meet new requirements.
- Low-code platforms demonstrate good performance parameters on datasets that are average or low size. But there is no guarantee these will work for large datasets. The challenge lies in analyzing **performance glitches** and real-time production failures due to large datasets. This often leads to a bloated code base. Making changes to improve performance might result in a complete overhaul at design level, which defeats the purpose of using a low-code platform.
- While there are varied project spaces provided by platforms, reusing a piece across region-specific releases by enabling a subset functionality from the superset piece of code is challenging. Requirements like maintaining **multi-tenancy**, handling deployment and configuring business logic is typically open for users to explore, posing higher risk of failure.
- The proprietary code, design and methodology used in low-code platforms bind users to them. If code must be moved to other platforms or technologies, it requires significant effort to start from scratch. The ROI from low-code platforms used in frequently changing business initiatives is very low due to the **lack of support when migrating code** from the proprietary platform.
- **Lower control of data and code** coupled with the freedom to develop apps may lead to compliance issues and poor governance. Citizen developers may not be aware of compliance and regulations and, thus, may not adhere to the IT, business and data governance requirements of the organization. This may result in the deployment of feature-rich applications that have complex underlying code, do not meet compliance requirements and expose data due to poor business logic.
- The deployment architecture of apps developed on low-code platforms has fewer options. Any tweaks made by users to customize the app can lead to **possible failures in deployment** due to incompatible integration with external components/environments.

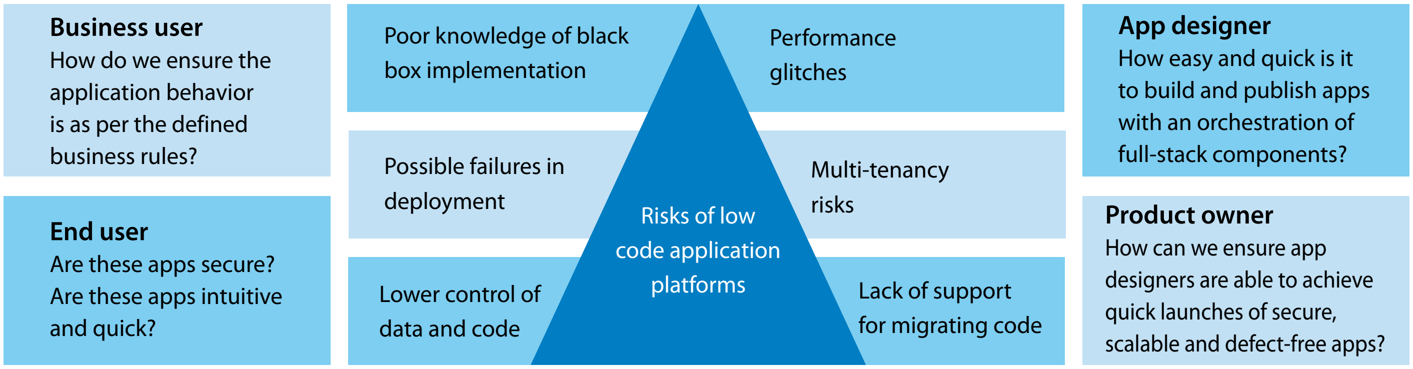

Business user
How do we ensure the application behavior is as per the defined business rules?
End user
Are these apps secure? Are these apps intuitive and quick?
Poor knowledge of black box implementation
Performance glitches
Possible failures in deployment
Multi-tenancy risks
Risks of low code application platforms
Lower control of data and code
Lack of support for migrating code
App designer
How easy and quick is it to build and publish apps with an orchestration of full-stack components?
Product owner
How can we ensure app designers are able to achieve quick launches of secure, scalable and defect-free apps?

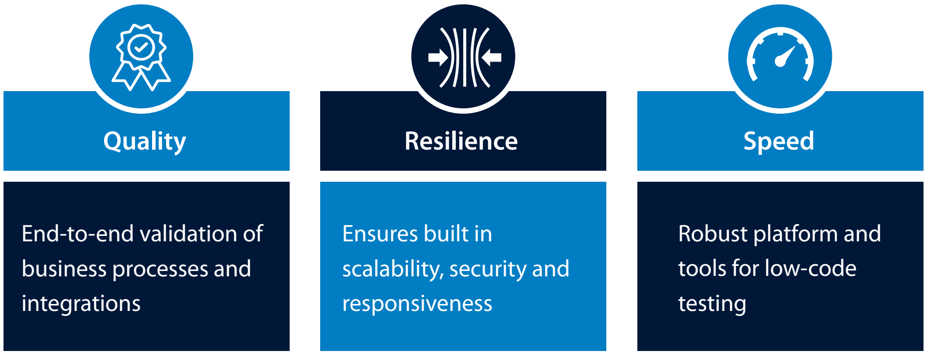
Figure 1. Risks in low code applications

Fit-to-purpose QA strategy

The right approach to validating digital services can help organizations that have started on their low-code application development journeys address risk and reap value. Validation brings business and IT together to collaborate and deliver scalable, secure and user-friendly applications. But it calls for a good understanding of low code development methodology and the ability to strategize testing of multi-experience applications. In this context, a fit-to-purpose quality assurance (QA) strategy will give organizations an undisputed competitive advantage, helping them gain a larger market share with reduced risk of failure.

A specific focus on QA and the introduction of an explicit verification activity in the low code application lifecycle may seem like an extension of the short release timeline promised by the low code development methodology. However, this could result in a diminished ability to rapidly capture the market due to low-quality production releases. The right validation approach is needed to avoid financial losses arising from breaches in security, data standards and domain-specific regulatory compliances. Organizations that aim to get maximum ROI with quick and flawless application releases will need a fit-to-purpose test strategy for their low code application development journey.

Key elements of a fit-to-purpose test strategy



Quality

End-to-end validation of business processes and integrations

Resilience

Ensures built in scalability, security and responsiveness

Speed

Robust platform and tools for low-code testing

Figure 2: Key Elements of low- code test strategy

- **Focus on quality:** It is necessary to validate the business rules implemented in low-code applications for functional correctness as well as security compliance and scalability. This makes low-code QA different from traditional functional testing
- **Stay future-ready with resilient strategies by:**
  - Focusing QA on extrapolated business needs and an expanded user base based on the organization’s roadmap
  - Adapting to technology evolution by leveraging testing methodologies and tools catering to the new-age application development landscape
  - Using AI/ML to build a live quality engineering practice that is sentient to the agile ecosystem
  - Responding to customer feedback by imbibing accessibility and UX test strategies
- **Gain market share with speed:** Hyper automation can be achieved through reusable test assets and orchestrating test automation across QA lifecycle stages with multi-facet test automation. This will ensure testing cycles are much faster compared to the traditional functional test automation.

Having understood the challenges in low code applications and the key elements of an intelligent test strategy, it is clear that low code applications cannot hit the market with low levels of QA.

Low code does not mean low QA

The fit-to-purpose test strategy detailed above clearly indicates that low code does not mean low quality assurance. Organizations that make the best use of impactful QA strategies achieve better quality at a faster pace. This will create the right impact in enhancing brand value and customer experience while saving maintenance cost. Traditional QA approaches focus on enhancing the quality of the applications. With low code, the need of the hour is speed without compromising quality.

The ideal QA partner for low code application development must have a unique test automation approach focusing on specific application areas that can upscale customer experience. They should also leverage test automation to help clients go-live with resilient strategies. With multiple demands on budgets from various important initiatives, the right QA partner can be critical in terms of managing costs.

Infosys approach to low code application validation

As a leader in quality engineering solutions, Infosys has designed a fit-for-purpose quality assurance strategy that enables testers to perform engineering tasks. This is achieved through hyper-automation of testing activities combined with end-to-end process automation, thereby helping organizations evolve into live quality engineering enterprises. Some immediate benefits of this approach are 35% cost savings, faster quality feedback and up to three times reduction in execution cycle time.

Case study

For a leading financial services company, Infosys ensured rapid deployment to production of low code apps. We also performed functional validation covering nearly 400 end-to-end test cases for 15 applications across web, mobile and tablet user channels, POS terminals, API middleware (including PDF comparison), and virtual printing of POS refund receipts. This approach to low code validation has helped the client automate nearly 85% regression testing for omni-channel test suites and reduce cost of regular fixes by 45-50% owing to optimal multi-faceted test automation.

Conclusion

Low code platforms can transform the application development market through greater encapsulation of code. These need the right validation and testing approach to eliminate impediments and mitigate risk. A best-in-class, proven, fit-to-purpose test strategy can drive greater adoption of low code applications while ensuring they are swift, secure, scalable, and user-friendly.

About the authors

Sandhya Rao, Senior Project Manager, Infosys

Harleen Bedi, Senior Industry Principal, Infosys

For more information, contact askus@infosys.com

Infosys®
Navigate your next