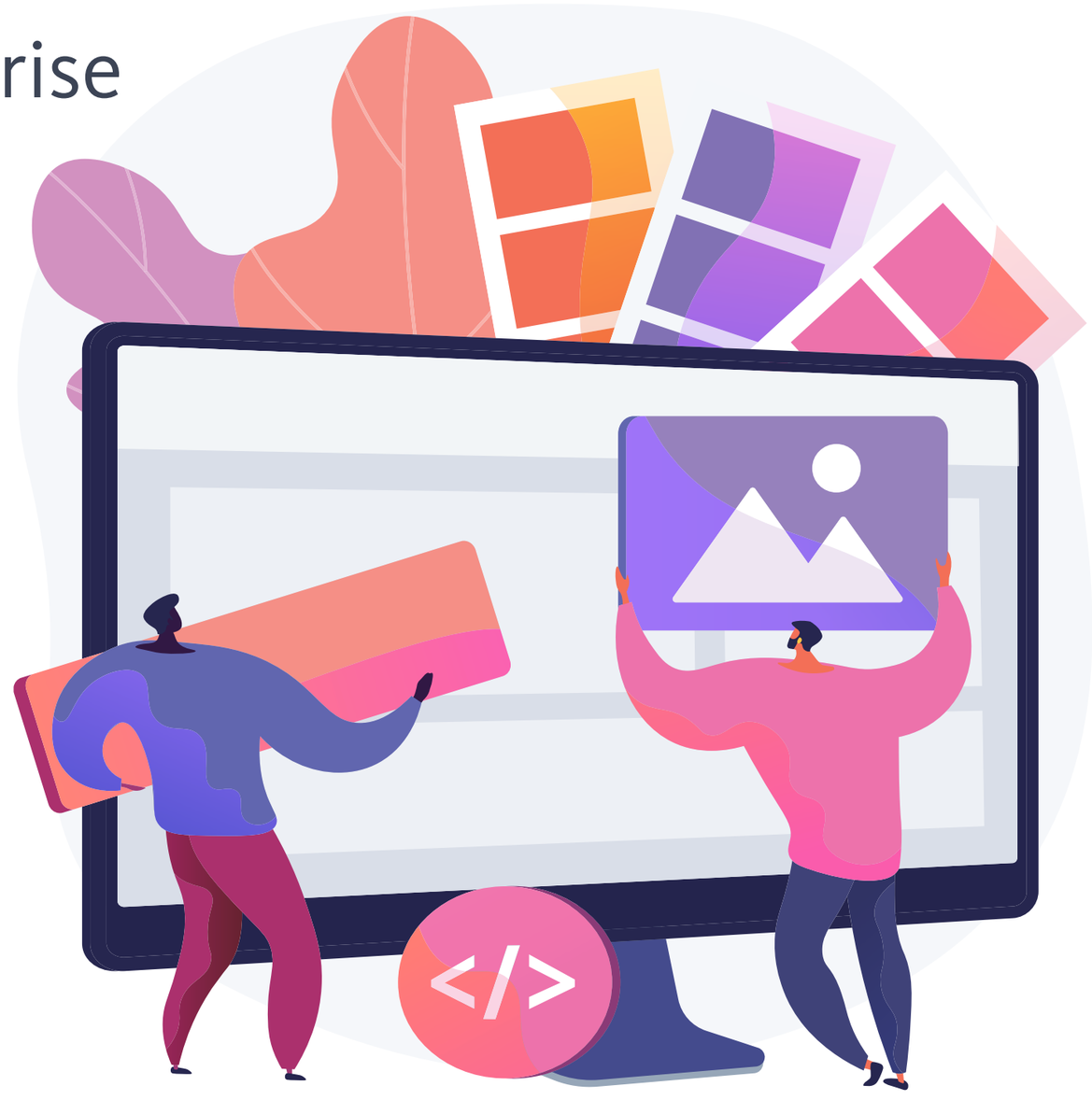
Top Ten Characteristics of Enterprise Low-Code/No-Code



Learn Why Most Low-Code/No-Code Solutions Are Not Enterprise-Ready

agilepoint

Table of Contents

Introduction

When leading IT analysts claimed that low-code technology was the second major IT development of the 21st century, second only to cloud computing, it took the IT industry by storm. As early as 2006, companies were highlighting how they successfully accelerated business growth through expedited mergers and acquisitions (M&A) activities by adopting enterprise low-code/no-code solutions.

According to Gartner, more than half of medium to large companies will adopt a low-code applications platform by 2023. Although IT executives recognize that adopting low-code could mean a competitive advantage in today's digital economy and worry about the cost of falling behind, many doubt that the technology is ready for enterprise. How can you ensure that you properly evaluate the myriad options available today and select the right enterprise low-code/no-code solutions to achieve the same success?

In this e-book, we will examine the top 10 characteristics of enterprise low-code/no-code platform solutions to help you make the right decision.

Why Most Low-Code/No-Code Solutions Are Not Enterprise-Ready

In order to work toward this digitization goal, businesses are taking a serious look at the myriad low-code/no-code solutions that dozens to hundreds of vendors are selling today. But the need for low-code/no-code technology did not emerge in the market until around 2016 when analysts started advocating for these solutions. Therefore, most low-code/no-code solutions available today are simply not enterprise-ready because many vendors did not have the time to develop an innovative solution that could empower both IT and citizen developers.

It can take upwards of a decade to create a general-purpose, domain-agnostic, enterprise-ready low-code/no-code solution. This time is needed to ensure the solution is equipped with process automation capability and mechanisms to effectively manage extensibility. In addition, time is required to enable comprehensive governance of the developed applications.

That said, most of low-code/no-code solutions available today deliver only part of the solution, most notably focusing on delivering only faster and easier development. As a result, end up adding to the problem they were striving to solve.

The State of No-Code Offerings

At one end, the vast majority of offerings today can be categorized as no-code, self-contained, task-specific technologies that are quickly designed and marketed for specific use cases. These offerings reach for the low-hanging fruit, providing quick-to-market, unsophisticated tools that address a single business issue. These may include things like a web-based user interface for a spreadsheet, an e-form application used to interact with a back-end database, or a mobile or document management application for a specific use case.

These no-code products could empower citizen developers, e.g. technically-savvy business analysts or subject matters experts to solve a short-term, immediate issue. However, when running out of the built-in functionality, they typically do not offer support for managed extensibility. This means they may not adapt to the evolving complexity of the business issue they were designed to address, or cannot adequately adapt to new use cases.

These point-oriented, no-code solutions are also not designed to easily integrate into an enterprise system to address mission critical, cross-functional needs. After adopting multiple point-oriented, no-code solutions, companies may face maintenance challenges due to ‘silos of no-code.’ Perhaps the most critical issue is that they are not easily governed by IT. As a result, they will inevitably become part of the technology sprawl that is so common in companies today.

No-Code

Task-specific, point solutions

- Lead to ‘silos of no-code’ and sprawl
- No or minimal support for process management and automation
- Lacking support for managed extensibility by IT
- Not easily governed by IT, leading to security and compliance challenges

The State of Low-Code Offerings

On the other end of the spectrum are low-code solutions designed mainly to improve the developer productivity. In general, there are two types of low-code offerings.

One type of the low-code offerings is the traditional Business Process Management (BPM) solutions that aim to enable IT to create end-to-end automation applications. Although these established BPM solutions provide the sense of 'low-code' through process modeling, enablement from process modeling to automation typically still requires complex development by IT. For example, many such BPM solutions are designed to work with a centralized API gateway-based integration solution that requires IT involvement, not suitable for citizen developers.

The more popular type of low-code offerings are those that are typically designed to increase the productivity of centralized, traditional IT development. Many, however, generate only incremental gains, shaving off a few hours here and there for the developer staff. Moreover, many of the low-code tools leverage the decades-old, code-generation techniques that only deliver the perception of 'low-code.' In reality, these solutions will still lead to the accumulation of technical debt characterized by the underlying, rigid code. Low-code solutions of this nature cannot empower citizen developers in a self-sufficient manner to perform continuous improvement.

Unless low-code technology can enable citizen developers to solve their own problems and deliver innovative solutions in the trenches, on the fly, and on a continual basis, it will fall short of the potential that low-code solutions can deliver to the enterprise. In the end, these solutions often fail to deliver more than incremental development productivity for the IT staff. They fall short of empowering decentralized development and widespread innovation by citizen developers, which is needed to deliver sustained transformation.

Low-Code
Traditional BPM Solutions

- IT-centric BPM solutions pitched as 'low-code'
- Complex development required to create end-to-end automation
- Not suitable for citizen developer empowerment
- Challenge in delivering automation at scale

Low-Code
Developer Productivity Solutions

- Productivity improvement for traditional development with 'code generation'
- Code-based 'technical debt' leads to maintenance & migration challenges
- Not suitable for process automation & continuous improvement
- Challenge in delivering automation at scale for transformation
- Not suitable for citizen developer empowerment

Layered Low-Code/No-Code Offerings

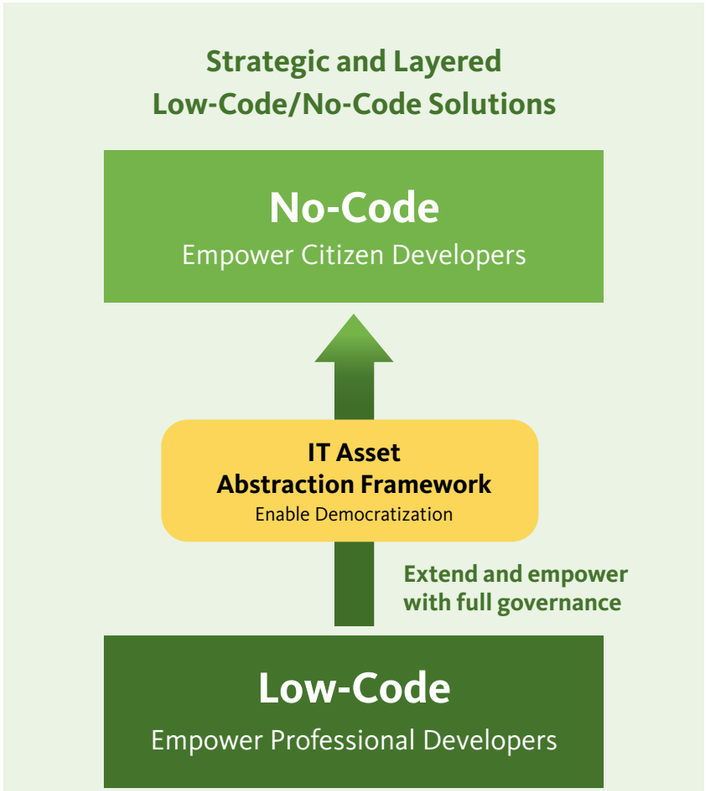
The key to delivering the agility and automation at scale that today's businesses need lies in eliminating the long cycle times and complexity of custom coding requirements. Low-code/no-code solutions must step in and shift the capability of creating highly adaptive business applications and processes as well as making minor and mundane process improvements to the citizen developer while allowing the IT department to maintain complete governance for security, compliance and manageability.

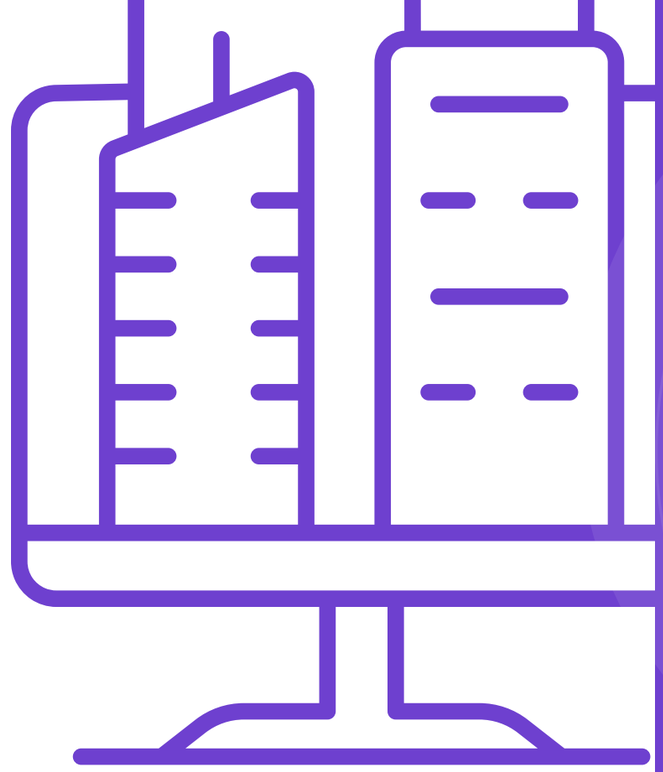
An enterprise low-code/no-code solution must be built upon a layered architecture and inherently enabled by an enterprise-grade digital process automation framework. This enables simplicity for development-time while ensuring the robustness and scalability of the developed applications and processes.

It also provides agility for run-time by enabling adaptable executions to respond to changing requirements without incurring repetitive development cycles.

This architecture allows citizen developers to innovate and resolve problems in their daily jobs using drag-and-drop, no-code tools. At the same time, it ensures that IT can extend the library of no-code functions and maintain complete governance over the solutions.

This combination promises to completely transform people and processes, making companies that adopt this technology to truly deliver sustained benefits that low-code/no-code technology has to offer and be more agile than ever. Solutions that can deliver both critical components are truly ready for enterprise.





TOP 10

Enterprise-Ready Characteristics

According to Gartner, more than half of medium to large companies will adopt a low-code application platform by 2023. If you're one of those businesses, be sure you understand the top ten characteristics you need for a truly enterprise-ready solution before making your selection. Doing so can mean avoiding constant restarts as well as high migration costs.

01. An enterprise-ready low-code/no-code solution must be general-purpose to address a wide-range of use cases.

The low-code/no-code solutions you select must be capable of addressing a wide range of business use cases. This eliminates the need to adopt multiple task-specific no-code solutions that result in 'silos of no-code,' technology sprawl and compliance risks when developed applications cannot be effectively governed by IT. The same applies to low-code solutions that may speed up code-based development, but not be suitable for creating process automation applications that need continuous improvement without additional coding.

We can see a similar set of events that occurred with cloud computing. Cloud computing started out with a large number of task-specific or domain-specific Software-as-a-Service (SaaS) solutions. As many businesses adopted multiple SaaS solutions, they realized that they had simply moved on-premise silos to the cloud. This created a market demand for general-purpose Platform-as-a-Service (PaaS) cloud solutions that could address a broader range of use cases.

During the pandemic, many businesses needed to support remote working. For instance, various business units of a regional bank adopted multiple no-code solutions to speed up digitalization. These included creating web forms to access data stored in spreadsheets and shared file servers, setting up simple approval workflows, and creating mobile apps. IT soon realized that they could not effectively govern the access control policy of data for one solution nor meet audit and compliance requirements for another. Furthermore, integration and customization of applications created with these no-code solutions typically was only possible through custom scripting codes directly for each application. That practice would lead to sprawl as it cannot be systematically tracked and managed. An enterprise-grade general-purpose low-code/no-code solution can help eliminate silos and avoid such unmanageable sprawl.

02. An enterprise-ready low-code/no-code solution must empower both citizen developers and IT.

The solution you select must have a layered low-code/no-code architecture.

The no-code layer should be designed to progressively empower citizen developers. They must be able to develop their own applications solve immediate problems, implement new ideas quickly, and make their jobs more efficient. If citizen developers see a way to improve the company's existing processes or operations, they should be able to drag-and-drop and visually configure that improvement from a set of out-of-the-box, no-code activities.

At the same time, IT must be able to leverage the low-code capability to rapidly develop and create new visually configurable no-code activities to extend the out-of-the-box activity library thereby further empowering citizen developers.

For example, a remote customer service representative of a financial services provider sees that customers often call multiple times to inquire about fund transfer status. This occurs because the existing system cannot provide in-progress visibility while the request is being processed. Since this has been a recurring request from different customers, that service representative wants to be able to add such visibility in the online self-serve message center.

Instead of burdening IT to retrieve the fund transfer status information on a recurring basis, only to then be manually emailed to the service representative, and then to the customer, they need a more automated solution. IT can instead create a no-code activity to retrieve status from the system, which allows the service representative to easily incorporate the information into self-serve services, adding this visibility for selected customers.

03. The ability to create and maintain process automation at scale must be present in enterprise solutions.

Leading analysts such as Gartner and Forrester have stated that achieving automation at scale is a key to digital business transformation. In order to be enterprise-ready, low-code/no-code solutions must be able to marry low-code technology with process automation. This union allows a business to deliver automation at scale, touching everything from specific workflows at the departmental level and all the way up to automation and management processes that span the entire organization.

For example, a global high-tech conglomerate decides to move from SharePoint on-premises to SharePoint Online in the Office 365 cloud. To do so, the organization must rebuild more than 5,000 SharePoint on-prem workflows created over 10 years. The company quickly establishes a citizen development program, leverages a low-code/no-code platform, and creates nearly 2,000 digital process automation applications in 24 months to replace 5000+legacy workflows. They're now more agile and productive as a result.

In another example, once the pandemic necessitated the rapid shift to remote work, a global manufacturer saw a surge in requests sent by its suppliers and contractors to the shared email inbox to inquire about the status of POs, invoices, statements of work, etc. The company's IT was able to leverage a low-code/no-code solution and quickly create an end-to-end digital process automation application that connected people and multiple systems including SAP, Salesforce, and document management systems to retrieve the needed information with proper approval and access control policy to improve the response.

04. Your solution must not only address process needs but also extend to day-to-day business applications and mobile apps.

While a strength of low-code/no-code platforms can reside in creating process-based development tools, an enterprise-ready solution should also allow citizen developers to create and improve business applications related to tools they use every day.

Business professionals must be empowered to improve and correct forms, databases, spreadsheets, analytics reports and dashboards, and mobile applications.

For example, technically-savvy business developers at a financial service company created hundreds of Excel-based and Microsoft Access Database applications to address the lack of IT bandwidth. Unfortunately, these applications could not be properly governed by IT and caused security and compliance risks. An enterprise low-code/no-code solution could enable citizen developers to build enterprise-grade, no-code applications as replacements while ensuring they could be fully governed by IT.

05. In order to ease user adoption, low-code/no-code solutions should be able to embed into existing systems.

When new tools appear on existing user interfaces or cloud services, they are much less intimidating than rolling out a completely new system or program. Low-code/no-code solutions should have the flexibility to seamlessly integrate into, and appear on the user interface of existing systems. If users are already familiar with the interface, they will more readily accept and adopt the low-code/no-code solution. At the same time, this will reduce re-training and support costs, which both lower total cost of ownership.

For example, if a low-code/no-code solution's task management interface could appear in the portal or as a toolbar in familiar existing applications, it would greatly facilitate adoption.

Take a professional service firm that uses Salesforce as its CRM and wants to improve its customer on-boarding experience. This firm can leverage an enterprise low-code/no-code platform to rapidly develop a customer on-boarding process application. The on-boarding automation is triggered as soon as the status of an opportunity in Salesforce is marked as 'closed won,' and new customer data is sent simultaneously to multiple departments. The legal department begins preparing contracts in Microsoft Teams and SharePoint, the finance department sets up new ERP accounts, and the service delivery team schedules work order details in Jira project tracking system and Office 365. To promote the adoption of the customer on-boarding process and collaboration across these departments, a low-code/no-code solution must make its task management and progress reporting functions visible in tools such as Salesforce, Teams, SharePoint, Office 365, and ERP.

The Strategic Digital Transformation Democratization Platform

One platform to simplify, accelerate, and sustain your digital transformation

Rapid Digitalization at Scale → **Continuous Optimization at Scale**

agilepoint

Data-Centric Applications
(Spreadsheet • Mobile • Database • Form)

Process-Centric Applications
(Workflow • BPM)

Well Governed Personal Empowerment

Enhance spreadsheet apps with centralized data governance.

Multi-Channel Experience

Deliver web, mobile and embeddable experience.

Workflow – Widespread Automation

Enable workflow collaborative automation at scale that can be easily re-purposed cross platforms.

BPM - End-to-End Automation at scale

Connect people, systems & data, and integrate with 70+ systems.

06. Low-code/no-code enterprise solutions should enable the creation of future-proof applications that can adapt to ever-changing business needs.

In the past, a software or application that met a business need and operated on a company's current system was good enough. Today, the pace of business and its needs are constantly changing, and new technology is always being introduced. At the same time, mass-personalization such as the desire to deliver individualized customer journey experiences has never been more important to drive competitive advantage in today's digital economy. An enterprise low-code/no-code solution must enable created applications to be easily portable, which means minimal future migration costs without a complete rebuild. They should also have the ability to dynamically adapt for variations of the same use case without the need to create and maintain multiple versions of the same application.

Enterprise low-code/no-code solutions must be agile enough to not only absorb those changes but help facilitate them. This is where the low-code/no-code solutions that depend on code-generation techniques fall short.

The ability to create applications that can adapt to evolving variations of the same use case also opens of the door to enable AI-driven automation applications.

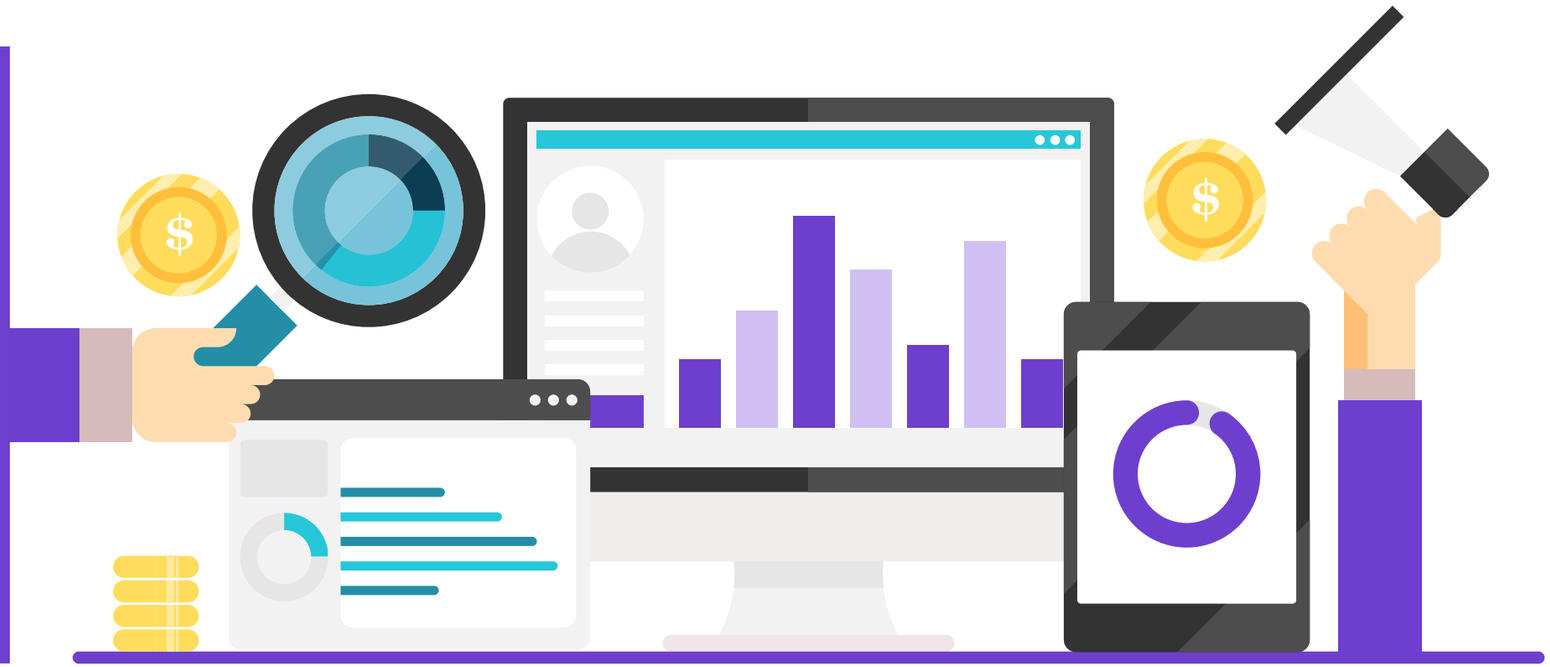
For example, if a business chooses to migrate from one software-as-a-service (SaaS) vendor to another, the company may face significant challenges if many processes and business applications are integrated into the old SaaS through calling their APIs by coding. Let's look at the previously cited global high-tech conglomerate that created 2,000 digital process automation applications. One of the key criteria of an enterprise low-code/no-code solution is to ensure the created applications are future-proof. The new applications can now transparently run on either SharePoint on-premises version or SharePoint Online version without change. When an M&A transaction occurs, the acquired subsidiary that uses a different system, such as Google Workspace (G Suite), should be able to take advantage of the 2,000 digital process automation applications with a simple re-configuration, avoiding costly refactoring or rebuilding. This same adaptability can be applied to working with multiple CRMs, ERPs, etc.

07. Low-code/no-code solutions must remain under IT governance.

One of the biggest drawbacks with point-oriented, no-code solutions is that they often sit outside of IT governance. That can lead to support, security and compliance issues for the entire organization. In many cases, technology sprawl is a result of too many non-integrated solutions that consume IT resources.

Instead, IT must have complete governance from the design through the deployment of any low-code/no-code solutions. This will allow IT to ensure the company is compliant, data is secure and technology solutions are manageable.

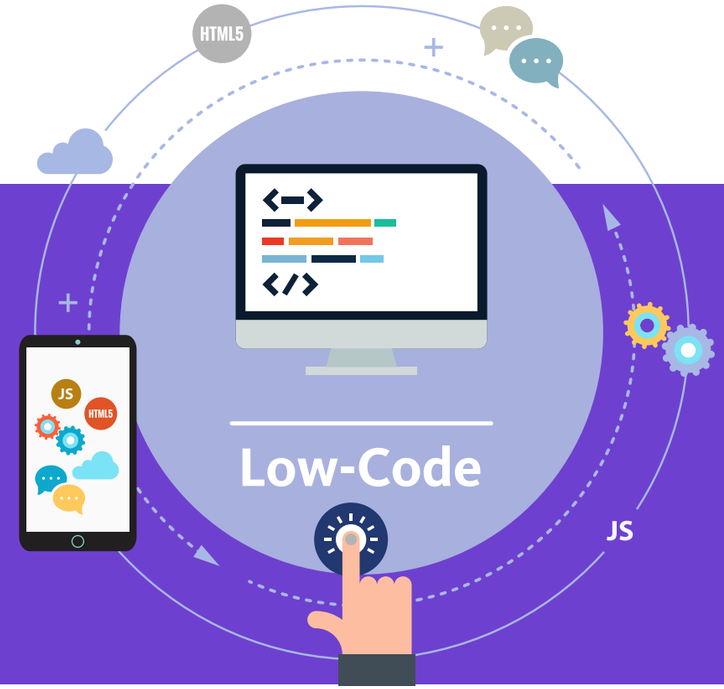
For example, a pharmaceutical company uses a no-code point solution so that its business users can web-enable their Excel spreadsheets. However, the solution does not accommodate IT's role-based access control policy, nor maintain a proper audit trail, which leads to violating both the company's security policy and regulatory compliance system.



08. Low-code/no-code enterprise solutions should have built-in integration capabilities and enable self-serve integration for citizen developers.

Digital business transformation demands that businesses scale their ability to design and deploy an increasing number of mission-critical, end-to-end business processes. These must cut across functional areas and connect people, data, and systems. Therefore, enterprise low-code/no-code solutions must have built-in capabilities that can enable citizen developers to perform self-serve integration to the most popular enterprise systems and cloud services under full governance by IT. For example, IT should be able to create data connection access tokens that can be granted to citizen developers based on permissions.

At the same time, enterprise low-code/no-code solutions should still be able to support the traditional centralized integration management by IT using an API gateway solution.



Let's look at a healthcare service provider that adopted an established BPM solution that was repackaged as a low-code offering. The provider wanted to empower its business process analysts to create more end-to-end business automations. Unfortunately, the repackaged BPM solution required that integration be done through an API gateway, which required IT involvement and increased time to value. An enterprise low-code/no-code solution must support integration through permission-based connectors that enable self-serve integration for citizen developers.

09. Low-code/no-code enterprise solutions must enable creating applications and processes that have no dependency on the underlying infrastructure and future changes.

As companies look to adopt low-code/no-code solutions to accelerate digital transformation, one common mistake is to create applications and processes that have direct code-based dependency on the underlying infrastructure. This infrastructure may subject to change over time, and such dependencies will greatly diminish the adaptability of the solution, and lead to costly migrations, massive refactoring or complete rebuilding requirements. This is especially relevant when it comes to accelerating integration of business operations in an M&A scenario.

An enterprise low-code/no-code solution must enable the creation of applications and processes that can be layered on top of the existing infrastructure and freed from the dependency on the underlying infrastructure. This allows the applications and processes to be leveraged as shared services among the parent company and subsidiaries, which will accelerate the cross-entity business integration. More importantly, this approach will shield the user from the on-going changes or consolidations of the underlying infrastructure.

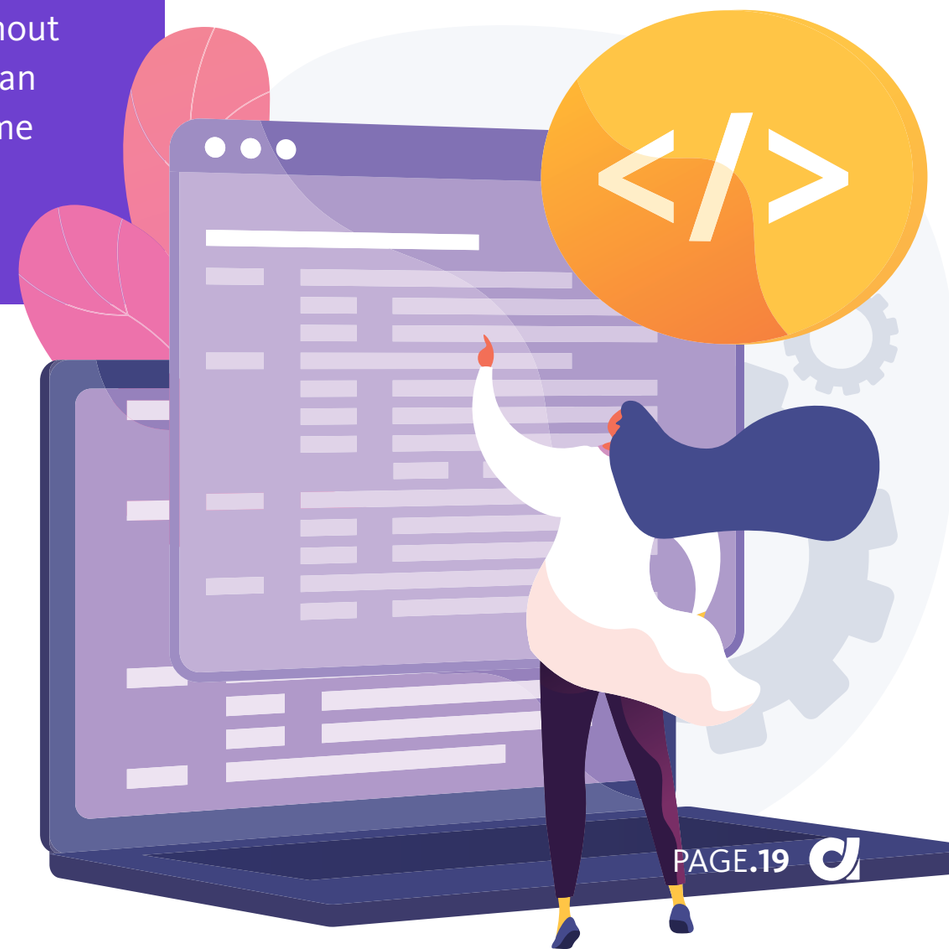
For example, a multi-national, high-tech manufacturer decides to accelerate its growth through a global M&A strategy. It adopts an enterprise, layered low-code/no-code solution. IT leverages the low-code development framework to build reusable, no-code components as effective connections to the underlying systems of various subsidiaries by encapsulating the ERP system and other functional applications. Business analysts would then create adaptive process-enabled applications as shared services and orchestrate these among subsidiaries and the parent company. This accelerates business integration while complying with regulatory requirements.

The approach allows the manufacturer to take a best-of-the-breed approach by leveraging a combination of underlying systems available from various subsidiaries. At the same time, the approach also enables the subsidiaries to undo customizations over the years to their ERP, PLM, CRM, etc., and restore them back to the native state to simplify maintenance, upgrade, and consolidation.

10. Low-code/no-code enterprise solutions must enable businesses to balance empowering more individuals to develop applications with achieving application portfolio rationalization.

While businesses may have different objectives when adopting low-code/no-code solutions, one common aspiration is to accelerate application development. Enterprise low-code/no-code solutions should also facilitate creating adaptable applications and processes that can consolidate and streamline a business' application portfolio.

For example, take the same multi-national, high-tech manufacturer who accelerated its global growth strategy through adding more than 50 global subsidiaries. The manufacturer uses an enterprise low-code/no-code solution to create adaptable business applications that can be consumed as a service. Each application can adapt to changing requirements of its global subsidiaries without the need to create and maintain more than 50 subsidiary-specific versions of the same application.



Conclusion

Aside from the top ten characteristics discussed above, there are additional characteristics that are also important to consider. For example, enterprise low-code/no-code solutions should support deployment for both on-premises and in-cloud, and AI-driven dynamic automation to enable mass-individualization without coding changes.

However, keeping these top ten characteristics of enterprise low-code/no-code in mind is critical before selecting a solution for your business. Relying on no-code point solutions may only add to your long-term technology sprawl problem, and choosing the wrong low-code package may result in only incremental gains for your IT development staff and delayed strategic benefit of widespread innovation and transformation through citizen development.

Cloud computing took years to transition from task-specific SaaS solutions to more general-purpose PaaS solutions, which addressed the ‘silos in the cloud’ challenge. Let’s learn from this lesson today. Select the enterprise low-code/no-code solution that avoids constant restarts as well as high migration costs to gain a sustained competitive advantage.

Be sure to seek a general-purpose and layered low-code/no-code solution that offers you a strategic platform to democratize digital transformation through sustained empowerment for your citizen developers while at the same time enabling managed extensibility and maintaining complete governance through your IT staff. This combination will give your company the ultimate competitive advantage the low-code/no-code advancement has to offer and the agility to be completely transformative moving into the future.

To learn more about how AgilePoint can be your strategic digital transformation democratization platform, visit **www.agilepoint.com**.

FORRESTER®
WAVE LEADER 2019
Digital Process Automation For Wide Deployments

Learn More | Request Trial | Request Demo | Download Report

© 2021 AgilePoint, Inc.