

Brian Fleming

Low-Code Buyer's Guide

How to assess vendors, understand licensing models and decide which low-code platform is right for you.



Low-Code Buyer's Guide

There's no doubt about it – low-code software development is the secret to a successful and sustainable digital transformation. That's why Gartner predicts that the use of low-code and no-code technologies will rise threefold by 2025, accounting for over 70% of new applications developed by businesses.¹

Many established enterprises currently face a growing backlog of required business apps and integrations, especially as they make the move to more agile multi-cloud environments. Low-code and no-code platforms offer promising solutions to meeting those demands. However, with hundreds of vendors to choose from, along with numerous licensing models, and radically differing and complex business requirements, navigating the low-code marketplace is far from easy.

In this buyer's guide to low-code development platforms (LCDPs), we'll explore the challenges and opportunities of low-code development and provide practical advice to help you evaluate and integrate low-code solutions into your digital transformation strategy.

Table of contents

1	Chapter 1: Democratizing Software Development With Low-Code
3	Chapter 2: Key Capabilities to Consider When Choosing an LCDP
5	Chapter 3: Addressing the Challenges of Low-Code Development
6	Chapter 4: Understanding Low-Code Licensing Models
7	Chapter 5: Defining Your Business Use Cases and Their Complexity
9	Chapter 6: Evaluating and Negotiating With Low-Code Vendors
10	Chapter 7: Top 10 Low-Code Platforms
20	Conclusion
21	Sources

Chapter 1: Democratizing Software Development With Low-Code

Software development is undeniably one of the world's most valuable industries to the point it has become a key driver of innovation across every sector. However, it's also an industry that has long presented very high barriers to entry. Professional software developers acquire their skills over many years, and there simply aren't enough to go around to meet today's demands. This, in turn, presents barriers to innovation and makes it harder for businesses to release the applications, user experiences, and other innovations that their customers expect.

By democratizing software development, and thus reducing the barriers for people seeking to enter the industry, organizations and the economy as a whole can unlock further opportunities for innovation. Low-code and no-code development makes that possible by abstracting away one of the biggest complexities of software development – i.e. programming.

That's not to say software engineers and other professionals will suddenly become obsolete. What it does mean, however, is that they will in turn be empowered to take on more complex challenges, while routine software development operations, such as business app creation and maintenance, can be tackled by a far broader range of people.

Most importantly, low-code enables citizen developers and business technologists with little to no programming experience to be actively involved in development. According to Gartner, nearly two thirds of organizations now have, or plan to implement, citizen developer initiatives.² The driving force behind this need is the constantly evolving demand for digital transformation. Business users need agile solutions to their problems, while IT teams face growing backlogs that they simply can't handle by themselves. In other words, low-code is fast becoming a key enabler of digital transformation.

The democratization of software development is also born of the fact that, as is often said, every company is now a technology company. Software plays an essential role in the day-to-day routines of practically all knowledge workers, regardless of which department or industry they work in. To that end, it just makes sense to adopt a low-code and open-source approach to software development and maintenance.

Ultimately, low-code presents an opportunity for businesses to democratize, customize, and automate a fundamental domain of digital transformation – that being software development. By choosing the right low-code development platform (LCDP), businesses can also reduce unnecessary complexity and, in doing so, cut down on backlogs in the IT department. Another challenge that low-code aims to address is the need to optimally distribute IT budgets which, all too often, end up being disproportionately invested in maintaining outdated legacy systems.

Going forward, these challenges are only likely to intensify, so it makes sense now more than ever to invest in the right LCDP.

Why Invest in a LCDP?

- ✓ Your organization faces a growing need to deploy multiple apps and workflows
- ✓ There is a lack of off-the-shelf solutions to accommodate your specific use cases
- ✓ You seek to achieve a closer alignment between business and technology
- ✓ You need to reduce software complexity and development cycle times
- ✓ You need to deliver customized, albeit standardized, user experiences

Chapter 2: Key Capabilities to Consider When Choosing an LCDP

There are now hundreds of low-code vendors and platforms to choose from, and many more integrate low-code functionality into their broader software suites. This can make it difficult to choose the right solution, especially given the fact that there are so many possible use cases. We'll explore these in more depth in Chapter 5.

High code vs. low-code vs. no code

When choosing a low-code solution, one of the most important considerations is the level of abstraction it provides – in other words the degree to which the platform hides the underlying complexity of software development. There's no industry-standard definition of precisely what constitutes a low-code platform, which is why it's better to view it as a spectrum.

On one side of this spectrum, we have traditional programming, which, in comparison to low-code, might be referred to as high code. High code relies on professional developers to write, deploy, and maintain code. While high code is essential for creating underlying infrastructure and application frameworks, it relies heavily on professional developer resources.

On the other side is no code, which puts business users directly in the driver's seat by entirely removing the need for programming. Instead, they rely on things like user-friendly forms and other simple input methods that don't require any specialized skills. However, this also means that no-code tools are limited in scope and can't help bring about large-scale innovations.

Low-code sits around the middle of the spectrum, which also makes it by far the broadest area. However, the main value of low-code is that it promotes closer collaboration between business users and IT, something which has become extremely important in breaking down technology siloes and optimizing business-critical processes.

Abstraction levels in low-code platforms

Software developers also talk about abstraction levels in terms of capabilities, behavior, and users. Capabilities refer to the technical capabilities of a software application, while behavior focuses on how the application executes those capabilities, and the use level covers the actual way the end user interacts with the application.

Different LCDPs address these abstraction levels in different ways. For example, one LCDP might only offer a limited framework, thus providing only a limited range of capabilities. This is often the case with low-code solutions tied to specific software suites, such as how Salesforce Lightning is heavily tied to customer relationship management (CRM). Similarly, the behavior and use levels are restricted by the underlying capabilities of the platform. No-code platforms, such as Betty Blocks, by contrast, fully control the software stack.

A true LCDP should allow professional developers to code in custom functions across these abstraction layers, but they will often still be limited by the capabilities of the underlying source code. However, these limitations can be mitigated by using an open-source platform that gives you complete access to and control over the platform's entire code. We'll take a deeper look into why open source is a natural fit for low-code in Chapter 4.



Chapter 3: Addressing the Challenges of Low-Code Development

As with any innovation, the adoption of low-code development introduces both challenges and opportunities. Choosing the right solution is vital to mitigating those challenges and capitalizing on the opportunities.

One of the most common criticisms of low-code development is that it lacks ability to address complex business use cases. However, this is something of an outdated reputational problem that stems from the time when LCDPs were not nearly as sophisticated as they are today. In fact, while low-code will never completely replace high code in every use case, comprehensive LCDPs can still accommodate a broad range of use cases. Open-source alternatives can go even further by allowing professional developers to change and adapt the underlying source code to their specific needs as well. To truly empower creative freedom in your organization, you need an LCDP that democratizes development and ensures that everyone can put in their time in a manner that suits their areas of expertise.

Another common criticism of low-code is that it can lack alignment with industry standards, potentially resulting in the rise of shadow IT and reduced control over vital factors like security and regulatory compliance. Once again, however, this is an outmoded perception. Low-code has already been around for long enough and is more than developed enough as an industry that it has become the new standard in business application development. However, to ensure you obtain maximum value from an LCDP, it's important to choose a solution that's standards-orientated and, in doing so, allows you to seamlessly integrate the third-party services and APIs that you require. Moreover, many modern LCDPs are built around the common business process model and notation (BPMN) standard, which uses graphical notations that business users, rather than just professional developers, can understand.

The other challenge that businesses need to address when choosing an LCDP is the potential risk of vendor lock-in. In the case of low-code platforms, this may involve becoming dependent on vendor-supplied customization, professional services, or licensed vendor technology. This remains true of a lot of low-code vendors, but there are exceptions. Opting instead for a fully open-source LCDP entirely alleviates the risk of vendor lock-in, while an optional partnership with a support vendor still ensures you get the expertise and support you need when you need it. That way, you can enjoy the benefits of contracted consultancy and support services without being beholden to a particular vendor.

Chapter 4: Understanding Low-Code Licensing Models

As with choosing any software solution to empower digital transformation in your business, it's essential to have a clear understanding of licensing models and how they work. On one hand, low-code brings performance and agility to software development, but if the vendor only offers inflexible licensing models, those benefits can quickly end up being offset.

Most LCDPs are priced on a recurring basis, allowing you to use them for the duration of your subscription. In a typical software-as-a-service (SaaS) model, you'll usually either pay for the number of end users or the number of applications you want to release. While this does offer flexibility in the form of an on-demand service offering, it can still get costly. Furthermore, these licensing models usually pertain to closed-source software. This can be a deal breaker if you want to build software applications that you want to sell on to your own customers, either by way of an individual product or as part of a broader range of products and services. No-code platforms are typically even more restrictive when it comes to software licensing, since they tend to be heavily reliant on the vendor's underlying infrastructure.

Fortunately, while most LCDPs don't give you access to the underlying source code, they may still grant you ownership of the code generated by the platform. This does, in theory at least, mean that you get to keep any apps you create in your low-code environment and do whatever you want with them, even if you cancel your subscription.

One of the reasons why open source is a natural fit for low-code is that it allows businesses to retain full ownership and control over their code, including even the source code that powers the LCDP itself. With an open-source licensing model, you can also reduce your total cost of ownership (TCO) and ensure complete sovereignty over your data. This makes it open-source low-code solutions the obvious choice in any use cases that involve confidential or regulated data.

It's important to recognize, however, that open-source licenses aren't always what they seem to be. There are, after all, dozens of open-source licenses, all of which offer varying degrees of freedom over areas like linking, distribution, modification, patent grants, sub-licensing, and trademark grants. The open-source LCDP Corteza, for example, uses an Apache 2.0 license, which gives you the freedom to use, modify, distribute, and sell your apps in any way you want under your own license and trademark.

Chapter 5: Defining Your Business Use Cases and Their Complexity

Low-code can and should become a driving force of digital transformation in your organization, so it's important to look carefully for a solution that can accommodate your long-term needs. This is just as important in the case of free and open-source software too, since making any major change to your software stack is also a considerable investment in time, training, and business performance.

The first step is to clearly define your business goals before evaluating your current processes. Equipped with this information, your team will be able to develop a number of use cases for an LCDP and home in on the more precise requirements.

Common Use Cases For Low-Code Development For Business Applications and Processes

Fast Time-to-Market

One of the fundamental benefits of low-code is simplicity, so business apps that require a quick time-to-market are among the most common use cases.

Seamless Integration

Many businesses adopt low-code to improve their current infrastructure, in which case one of your use cases will be seamless integration of existing apps and processes.

Self-Service

Low-code can help bridge knowledge gaps in your organization, so if your IT team are overburdened with a large backlog, self-service functionality will be a key use case.

Automation

Pre-built and ready-to-use templates prevent you from having to start from scratch, in which case the automation of routine business workflows will be a use case.

Digital Sovereignty

Certain business apps and workloads are built to handle highly sensitive and regulated data, in which case achieving digital sovereignty will be a use case.

While most LCDPs should accommodate the first four use cases, the fifth is much more of a grey area, since software vendors often have the last word when it comes to control, security, and ownership of the platform. If achieving greater control and sovereignty over your apps and data is a key priority, then you're almost certainly going to want an open-source LCDP.

Use cases also vary by complexity, and different low-code platforms can support varying levels of complexity when it comes to software development.

Criteria Defining Complexity In Software Development

Integrations with other IT systems

Choosing a standards-orientated and API-centric platform can help make integration much easier.

The overall number of end users

If you're building apps for a large and diverse user base, you'll need a scalable and, ideally, cloud-native solution.

The number of business rules in your workflows

Again, a scalable and agile solution that can support lengthy and complex workflows will be vital here.

Data operations & complex calculations

You may need to choose a platform which caters specifically to the needs of data scientists and analysts.

Security & regulatory requirements

If an application is to handle sensitive data, such as financial information, the security requirements involved will increase complexity.

Chapter 6: Evaluating and Negotiating With Low-Code Vendors

Now that you have a clear picture of your requirements, it's time to evaluate the various market offerings. By now, you should have a shortlist of potential low-code vendors whose offerings broadly align with your business use cases. The next step is to narrow down the list further by evaluating the industry expertise of each vendor. For example, if you're a financial institution, it makes sense to choose a vendor that understands the industry and the unique challenges it faces. Moreover, many low-code vendors have partnerships with third parties such as systems integrators, which can also prove valuable in certain use cases.

Since investing in any platform that will come to play a major part in your digital transformation is a big decision, you'll also want to check reviews from both professional analysts and peers. Leading advisory firms like Gartner and Forrester provide independent research into low-code and other platforms, making them important resources during the evaluation stage.

By this point, you'll likely have narrowed down your list of potential vendors even further. Now is the time to request a demo from the vendors remaining on your list. Some vendors provide simple self-service tools that offer a guided trial to give you a more hands-on experience with the platform before making a commitment. As you're working to get a better understanding of the product, you should also consider requesting a specific demonstration that aligns with your particular use case. Then, finally, you can start negotiating with the vendor regarding pricing plans, implementation models, and the levels of support required.

What about open-source low-code vendors?

By far the most flexible LCDPs are those that are completely open source. However, people often make the mistake of thinking of open-source software purely because it's free to use and distribute. While that's true, there are many good reasons to take out a support contract, ideally from a vendor that's a primary contributor to the platform's development.

Although one of the main benefits of adopting low-code in your digital transformation initiatives is the opportunity to become self-sufficient, it's important to have realistic expectations. Many smaller businesses simply don't have the internal resources required to maintain their entire software stacks by themselves, and even if they do, it might not be cost-effective.

There are hundreds of low-code vendors to choose from. Below is a quick overview of the top ten.

1. Corteza

CORTEZA

Corteza, a Docker-based digital work and low-code development platform for web applications, is almost unique in the sector on account of being 100% free and open source. The platform, which was created by and is fully supported and maintained by Planet Crust, relies on open processes, standards, and licenses, which means it can cater to the needs of all organizations. Impressively, its Integration Gateway lets you connect Corteza with any data source for unparalleled control and flexibility, while its liberal Apache 2.0 license allows for complete digital sovereignty in a way that other LCDPs cannot.

Key Features

- 100% open-source and free forever
- 100% API centric and extensible
- BPMN 2.0 compatible workflows
- Import & export complete app configurations
- Full ownership of all software created

Who Should Use Corteza?

Corteza is an ideal choice for ISVs owing to its open-source nature, but ISVs can also partner with Planet Crust, the primary developer and maintainer of the project, to outsource their operational risk and get the developer support and hosting services they need to innovate rapidly.

Pricing Structure

As a fully open-source project, Corteza is available to install on your own server for free. However, Planet Crust offers several levels of support for Corteza and works closely with its customers to establish and execute its roadmap. For vendors seeking a cloud-based solution, Planet Crust offers managed, cloud-hosted pricing plans starting at \$3.74 per user per month (depending on the number of users) for enterprise subscribers or else can help vendors build their own clouds on Corteza.

2. Appian



Appian combines intelligent automation with low-code software development to accelerate the development of enterprise applications and workflows. Being one of the biggest players in the low-code industry, it is the platform of choice for dozens of global enterprises. It allows users to connect any data sources while meeting the demands of information security and regulatory compliance.

Key Features

- Supports mobile app development
- Integrates with IoT devices
- Includes prebuilt UI frameworks
- Extensible via integration SDK
- One-click deployment

Who Should Use Appian?

ISVs can partner with Appian to source new business opportunities and develop new services and applications for their customers. There are partner programs tailored to the needs of ISVs, MSPs, and OEMs.

Pricing Structure

The Appian Community Edition is available for free and has a 15-user limit. The fully featured Application package costs \$90 per user per month for a minimum number of 100 users. For the Platform or Unlimited package, you will need to contact Appian for a custom quote.

3. Mendix



Mendix supports application development for every type of device, and it is one of the pioneers of AI-powered software project management. It also provides offline working capabilities and can be installed either on-premises or in a private or public cloud. Intelligent automation is one of the platform's key selling points, and it can be adapted to automate a raft of routine business processes.

Key Features

- Build-in collaboration tools
- Extensible with APIs or pre-built connectors
- Curated library of 400+ building blocks
- AI-powered development assistant
- Cloud-native scaling

Who Should Use Mendix?

The Mendix ISV partner program provides an intuitive three-step process by which software vendors showcase their concepts, transform those concepts into solutions using Mendix, and then, if approved, sell and support them on the Mendix marketplace.

Pricing Structure

Mendix pricing starts at €50 per month for up to 5 users and one app. The Standard package costs €800 per month or €2000 for unlimited app development.

4. Microsoft Power Apps



Microsoft Power Apps makes it easy to build professional business apps with drag-and-drop simplicity and pre-built machine learning and AI components. Being deeply integrated with the tech giant's broader technology ecosystem and able to connect to hundreds of external data sources, developers can easily extend functionality. Users can also leverage the Azure cloud platform to scale their apps to the needs of their enterprises.

Key Features

- Develop web-based apps
- Create and connect AI models
- Native integration with SharePoint and Dynamics 365
- Build & certify customer connectors
- List your solutions on Microsoft AppSource

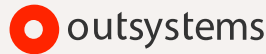
Who Should Use Microsoft Power Apps?

Power Apps is one of the more obvious choices for any ISV that already works primarily with Microsoft's products, especially those who use the Azure cloud computing platform. Microsoft offers a partner program that allows ISVs to sell their apps on the AppSource marketplace.

Pricing Structure

Microsoft Power Apps plans start at \$5 per user, per app, per month. For unlimited apps, the rate increases to \$20 per month. Pay-as-you-go subscriptions are also available, though they require an Azure subscription.

5. OutSystems



OutSystems leverages visual, model-driven software development and AI-powered assistants to accelerate the entire application life-cycle. It is a cloud-native platform with a strong focus on responsive app design for mobile users. The platform also incorporates in-app feedback to allow users to share their experiences and suggest improvements. Thus, OutSystems works especially well in the context of change management.

Key Features

- AI-assisted visual development tools
- Strong focus on mobile app development
- Enterprise-grade security and compliance
- In-app feedback and collaboration
- Scalable microservices architecture

Who Should Use OutSystems?

OutSystems can help modernize the way technology firms build apps and re-platform existing systems. The vendor's ISV and MSP partner programs facilitate flexible product delivery and incorporate turnkey marketing plans and comprehensive support.

Pricing Structure

Customers can launch their first app for free for up to 100 end users. The Standard plan starts at \$1,500 per month and allows customers to build and run multiple apps for their employees, customers, or partners.

6. Visual LANSA



Visual LANSA provides a powerful integrated development environment for creating desktop, web, and mobile applications. What makes Visual LANSA unique in the low-code market is how it combines low-code with traditional application development in the same IDE. This lets developers tackle more complex challenges and integrations while also taking advantage of the shorter cycle times of low-code for everyday development.

Key Features

- Create desktop, web, and mobile apps
- Ability to write code within the IDE
- Runs on IBM i, Windows, and Linux
- Flexible cloud-based or on-premises deployments
- Active repository engine for defining business rules

Who Should Use Visual LANSA?

Combining both low-code and traditional software development, Visual LANSA is a one-of-a-kind solution tailored to the needs of IT managers, ISVs, and full-stack developers. ISVs are free to build and sell solutions anywhere they want when using the LANSA framework.

Pricing Structure

Visual LANSA offers a 90-day free trial for non-commercial use. Paid subscription plans start at \$999 per month for one developer and two development environments and \$4,999 for the Standard plan. Tailored Premium plans are also available upon request.

7. Zoho Creator



Although best known for its CRM software, Zoho's technology ecosystem has expanded to incorporate the Zoho Creator development platform. The platform features visual workflows, integrated analytics and reporting, and connectors for more than 600 third-party tools. Every app built on the platform comes with native iOS and Android counterparts for multi-platform accessibility.

Key Features

- Import data with AI-assisted migration tools
- Model app structure and data flows
- Cross-functional data analytics
- Readymade integrations for 600+ apps
- Create custom UI components and Java libraries

Who Should Use Zoho Creator?

The Zoho Creator partner program is constantly expanding, with new ISVs joining up to create and sell customized apps to their own customers. The platform is best suited to consultancy firms and ISVs that specialize in deploying and advancing Zoho-based technology environments.

Pricing Structure

The Professional edition costs €25 per month, billed annually, and allows for the creation of up to five apps. The Ultimate edition costs €400 per month and supports unlimited apps and up to 10 users.

8. Salesforce Lightning



Salesforce Lightning is the low-code platform underpinning the industry-leading Salesforce CRM. The huge and constantly growing library of apps, connectors, and other tools, available through the AppExchange marketplace, is one of its major selling points. The Lightning App Builder facilitates quick and simple app creation and customization by way of a drag-and-drop interface and numerous reusable components – including those created by third parties.

Key Features

- Create personalized per-user experiences
- Automate workflows across connected systems
- Connect external data sources
- Secure and compliant testing environments
- Customize pages with drag-and-drop actions

Who Should Use Salesforce Lightning?

The Salesforce partner program lets ISVs tap into one of the largest online marketplaces for enterprise software apps. It supports the entire process, including development and design, selling and marketing, and customer service and support.

Pricing Structure

The Platform Starter edition costs \$25 per user per month, while Platform Plus costs \$100 per user per month and provides access to 110 custom objects and the Lightning Console.

9. Budibase



Budibase is an open-source low-code platform designed for building internal business apps. It features customizable templates for building forms and interfaces for a broad range of use cases. Budibase also supports all common data sources, and it may be self-hosted on your own servers or using Budibase's own paid hosting service.

Key Features

- Create responsive apps across all devices
- Use built-in automations for repeatable tasks
- Connect external database sources
- Adapt the user interface based on app conditions
- Choose from multiple hosting options

Who Should Use Budibase?

Budibase is best suited for building internal business apps. Being an open-source option, it's 100% customizable, making it a good choice for any organization that wants to retain complete control over their apps and the platform used to build them.

Pricing Structure

Budibase is completely free to use if you opt for the open-source self-managed tier. There's also the Cloud Free tier, which allows you to host the platform in Budibase, while additional services and support are priced on a per-client basis.

10. Wavemaker



Wavemaker is a low-code platform designed to accelerate app development and digital transformation in enterprises. While the Wavemaker platform itself is a licensed product, it is built on an open-standards tech stack that allows users to export the code used to build their apps.

Key Features

- Create user interfaces with customizable prefabs
- Build custom data models
- Deploy on-premises or in the cloud
- Control application behavior based on user roles
- Build responsive dashboards, login pages, and more

Who Should Use Budibase?

Wavemaker is ideal for larger businesses and enterprises seeking to implement a low-code solution that carries no risk of vendor lock-in. Its high-end security features, such as role-based access controls and authentication, also make it suitable for use cases that are subject to strict security and regulatory demands, such as banking.

Pricing Structure

Wavemaker offers a free trial, but the company does not publish any pricing information. Since they primarily target enterprise customers, which tend to have highly variable requirements, their pricing plans are quotation-based.

By closely following the advice in this guide, you'll be far better able to align your requirements with the right market offering. However, it's important to clearly define your goals, use cases, feature needs, and product ownership requirements before you even start looking for vendors, as doing so will save a lot of time.

Remember that no vendor knows your business as well as you do, so it's important not to leave it entirely up to potential providers to provide you a list of features and functions that you supposedly need.

On a final note, be sure to get your broader team involved in the selection process, not least because the main selling point of low-code solutions is how it enables more people to get involved in software development. As such, it stands to reason that choosing an LCDP should be a collaborative effort.

*Planet Crust helps independent software vendors, enterprises, and public sector organizations power and optimize their business processes at a fraction of the cost of traditional providers. Our team is the primary contributor to the Corteza digital work solution, a low-code platform that facilitates the development of scalable cloud apps and enterprise-grade CRM systems. We also offer consulting, customization, and support services to help you get the most out of the platform. **Get started for free** today.*



Sources

1 <https://www.gartner.com/en/newsroom/press-releases/2021-11-10-gartner-says-cloud-will-be-the-centerpiece-of-new-digital-experiences>

2 <https://www.gartner.com/en/documents/3970067>