

Introduction to Low-Code application development

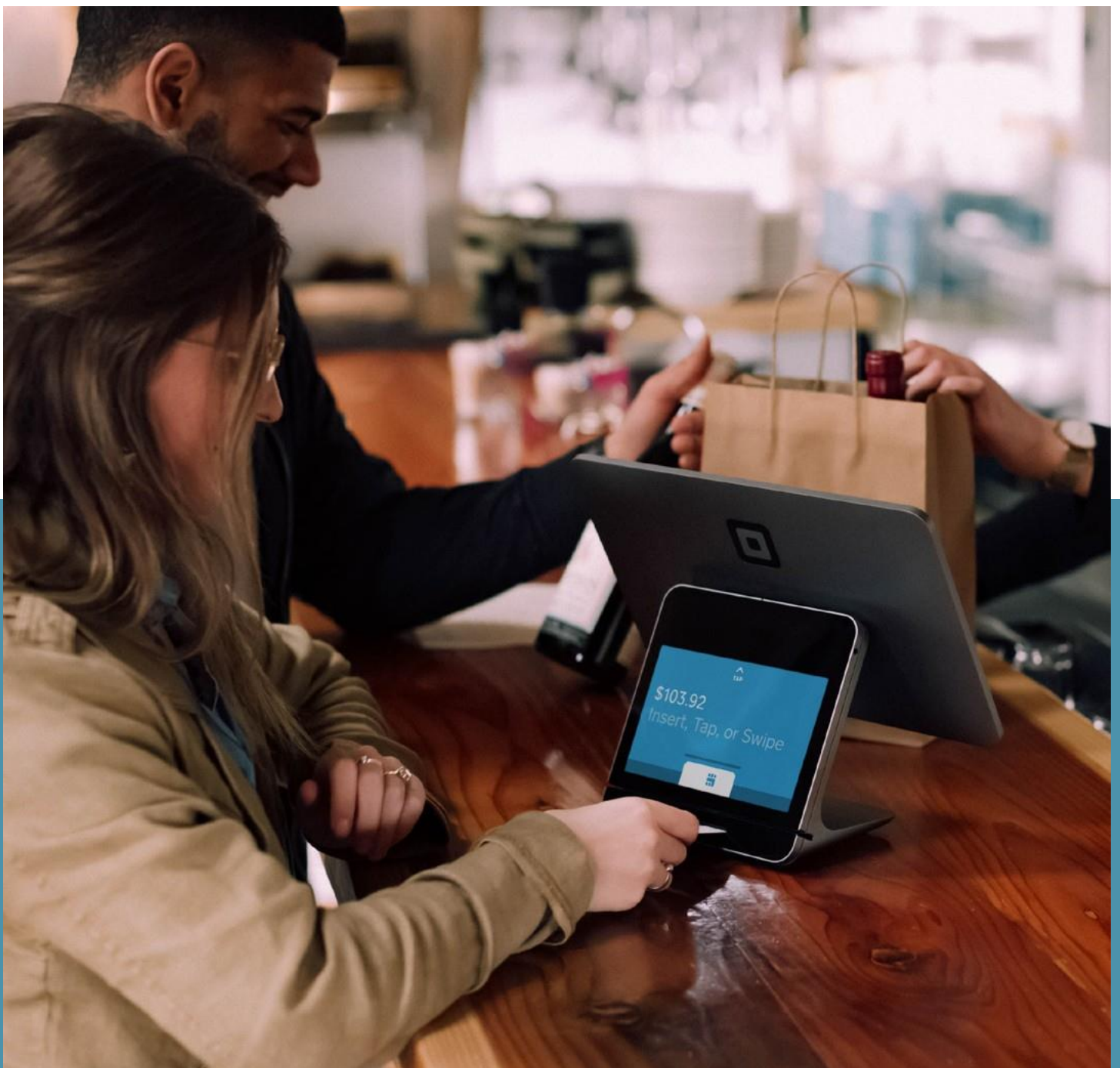


Table of Contents

1. What is low-code development technology?	2
2. Making a case for LCAPs	3
3. How does low-code development work?	4
4. Benefits of using a low-code platform	7
5. Low-code vs traditional development	8
6. Does your business need low code?	11
7. What can you build with a low-code platform?	13
8. The secret to low-code success	15
9. Low-code success in action	16
10. Fast track your company's digital transformation	25
11. About us	27

“

Low-code platforms make software development 10 times faster than traditional methods.

FORRESTER®

”



What is low-code development technology?

Low-code is an approach to software development that requires little to no coding to build applications and processes. A low-code application platform (LCAP) uses easy-to-use visual tools to build user interfaces and drag-and-drop features to add business logic into an application, instead of having to write extensive lines of code.



Making a case for LCAPs

Organizations today spend huge sums on complex legacy applications to solve key business problems. These commercial apps are used to address the business requirements of the various departments at an enterprise—from sales, all the way to accounting and HR.

Whether your business is a multinational enterprise or a five-person operation working out of a basement, you probably rely on one of two approaches when setting up your business software infrastructure—purchase packaged applications or maintain an in-house development team.

Packaged software is far from bulletproof. What happens when you encounter a deal or a roadblock that simply can't be addressed by readymade software? Furthermore, the traditional one-size-fits-all approach of off-the shelf applications leaves organizations with many gaps in operations.

However, the alternative is an expensive and time-consuming customization effort for your in-house development team.

Enter low-code development, to make the app-creation process easier. The low-code premise is simple: you just need to get someone that understands your business to create the app you need in the low-code environment. You won't have to concern yourself with things like packaged apps and outsourcing, and the dev team's time isn't wasted on limited-use apps.

How does low-code development work?

Low-code is a visual development approach to application development that uses the concept of abstraction. Low-code platforms offer intuitive visual builders, ready-to-use code snippets, form and report templates, and built-in connectors that speed up application development and extend the ability to innovate to a broader set of people.

Low-code platforms all have a few things in common:

Abstraction: Low-code platforms relieve nontechnical users from having to write code (while still supporting professional developers) by abstracting the tedious back-end plumbing and infrastructure tasks required in application development.

Intuitive visual builder: The visual development environment of low-code platforms lets users create complete applications using a drag-and-drop interface. Visual modeling interfaces enable non-developers to grasp application design quickly by allowing them to see how their application looks as they build it.

Instant development and deployment: Any good low-code platform comes with standard components like prebuilt templates for form and report creation, and ready-to-use code snippets. These out-of-the-box templates allow IT teams to develop and deploy applications faster, and with fewer errors. LCAPs also eliminate the need for creating frameworks, linking databases, and other tasks that are typically included in traditional development. to see how their application looks as they build it.

Simplified pre-built integrations: Low-code platforms allow for easy coupling with external and existing systems and new technologies, with a wide array of out-of-the-box APIs and other smart integration tools. tasks required in application development.



“

By 2024, low-code application development will be responsible for more than 65% of application development activity.

Gartner, Inc. “Low-Code Development Technologies Evaluation Guide” February 26, 2019

”



As an extension,

LCAPs:

Benefits of using a low-code platform

On a low-code platform, IT and business teams work together to build high-quality applications and accelerate time to market.

Drive agile development - Low-code platforms let you iterate apps, and release them as soon as functionality is built. Full-stack development (both client and server-side) bridges gaps between traditional SDLC and modern DevOps. Businesses can accelerate time to value by rapidly creating and launching applications, then enhancing and expanding them over time. The ability to carry out quick changes make agile transformation much easier. Application reworks are considerably reduced, and the business dodges cost overrun to a large extent.

Align business with IT - Low-code application development platforms greatly benefit enterprises by breaking down silos between IT and business analysts. Business users can bring innovative ideas to life by just dragging-and-dropping readymade elements or code snippets. These proof-of-concept apps simplify the life of programmers who can use them as a starting point when building enterprise software.

Eliminate backlogs and reduce shadow IT - Businesses can streamline, automate, and harmonize manual internal processes so that developers aren't overburdened and can focus on higher-value tasks. Minimizing overwhelming coding needs gives IT relief from the pressure of their often times unconquerable app backlog. This eliminates the need for businesses to employ risky third-party apps outside their certified IT infrastructure.

Reduce reliance on legacy apps - Outdated legacy applications, although vital to business processes, drain efficiency, and keep your IT resources in a constant state of updates and fixes. This means you spend more time on maintenance than innovation. Revive your business ecosystem using low-code platforms, with modern frameworks that can build consumer-grade applications.

Boosts security - Built-in security features, like audit logs, user-access control, and automated threat assessments, keep low-code platforms safe and reliable. Any new vulnerability identified is fixed through periodic updates. And new apps automatically inherit the latest security protection, ensuring complete safety of data. Plus, the platform provides all the necessary security framework certifications a company may need.

Low-code vs traditional development

Low-code platforms simplify and streamline the entire app rollout process, offering several advantages over traditional development technologies. Let's take a look at how they compare:

Traditional programming

Low-code platforms

Skill set/ expertise requirements

Niche, project-specific skills. Needs programmers with expertise in programming languages for web (Javascript, Python, CSS, PHP), iOS (Swift), and Android (Java, C++).

Little to no coding needed. An intuitive drag-and-drop interface makes it easy for both users and developers to create applications.

Cost of development (total cost of ownership)

Expensive. Custom-made software requires upfront investment for infrastructure setup, developer fees, continuous maintenance, and more.

Economical. License-based pricing allows you to pay as you grow and save on the massive amount needed for traditional development and infrastructure.

Multidevice deployment of apps

Single-OS focused. The specific programming language, environment, and device-focus must be selected before the project even begins. Businesses will need the capacity to develop for various different operating systems, environments, and devices. Applications need to be developed separately for each platform, and code can't be shared between Android and iOS.

Builds multiplatform-optimized apps. Cross-platform app builder helps to build native mobile applications that can be used on iOS and Android devices with multiplatform access. The same application works for mobile, the web, on-premise, and in the cloud.

Traditional programming

Low-code platforms

Speed to market

Looser timelines. Traditional software development can take months, with multiple teams building applications from scratch through extensive coding and multiple revisions of prototypes. And they offer no guarantees, and often go over schedule.

Exponentially faster. Ready-to-use code snippets and templates allow you to accelerate application development by 10x. Building off an existing framework and configuration is significantly faster than writing, debugging, and compiling code, and end user functionality can be created rapidly.

Security and privacy

Demands application-level security initiatives, which need to be manually configured by the developer. In traditional coding architecture, the code behind each form or module will need to be edited to build in the necessary defenses. If deployed into production, vulnerabilities would have to be found ASAP and plugged to prevent access to exposed data.

Safe and reliable, with built-in security features, like audit logs, user-access control, and automated threat assessments. Any new vulnerability identified is fixed in the next update. In turn, all the applications running on the platform automatically inherit the latest security protection, ensuring complete safety of data. Provides all the necessary security framework certifications in place, with proven experience in large-scale initiatives.

Enterprise application integration

Limited integration capabilities, without considerable scripting and testing of compatible code to integrate with new external applications. A significant challenge for enterprises with legacy systems that are vital to their business operations.

Fast integrations and complete control due to a responsive development environment. Easy coupling with external and existing systems, with the help of a wide array of out-of-the-box APIs and smart integration tools. Seamless cross-platform workflows to safeguard speed and efficiency.

Traditional programming

Low-code platforms

Application updates and maintenance

Requires dedicated labor to closely manage revisions and implement updates. Difficult to integrate the latest technologies across platforms (web and mobile) once the application is deployed.

Handled by the hosting company, low-code tools periodically roll out the latest updates and features to all applications built on them, without the need for a team or third-party vendor.

Scalability/ revisions

Requires substantial engineering efforts. Scaling after deployment demands a collaborative involvement from cross-functional teams—development, testing, production—to build enterprise-scale applications, with developers bridging gaps between business and IT.

Handled by the hosting company, low-code tools periodically roll out the latest updates and features to all applications built on them, without the need for a team or third-party vendor.

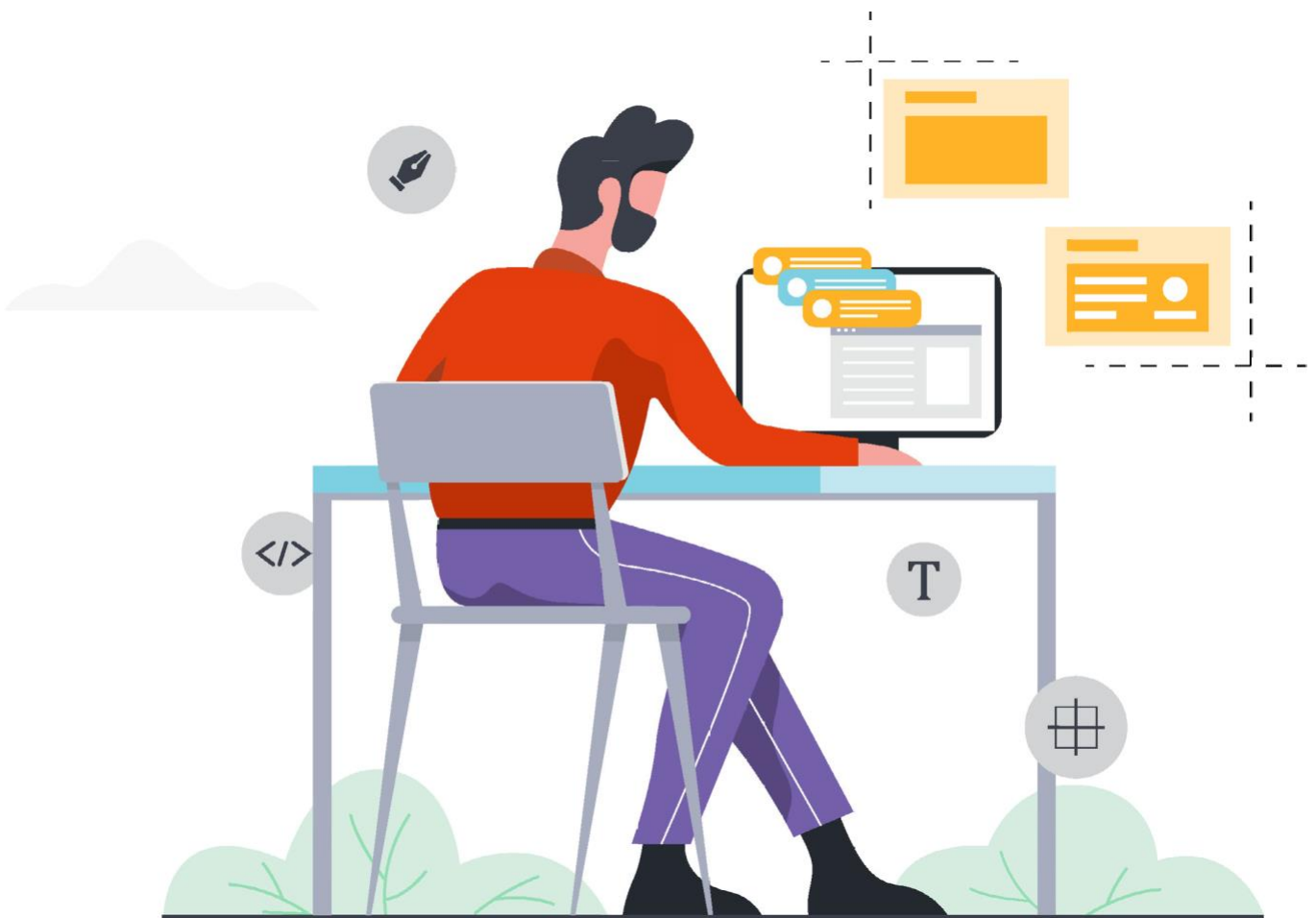
Support

Needs developers with expertise in programming languages for web (Javascript, Python, CSS, PHP), iOS (Swift), and Android (Java, C++).

Lifelong support is provided at a fraction of the cost, through AMC for the platform.

Does your business need low code?

Some signs that your enterprise should invest in low-code development:

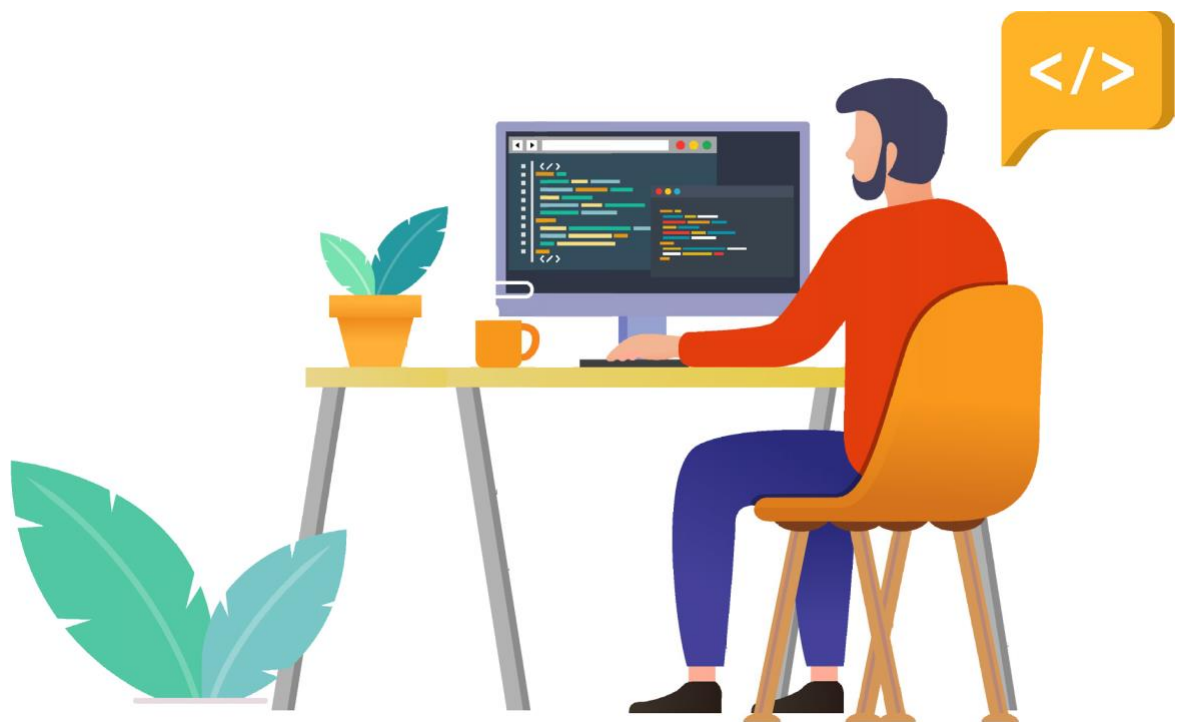


- Your organization struggles to operate at a digital speed. You're a growing business looking to capitalize on new opportunities. However, you're not able to execute on your innovative ideas quickly, and struggle with sluggish app development.
- Your business is slammed with too many projects, and your overwhelmed IT team cannot focus on innovation due to an endless app backlog.

- You work with outdated legacy systems that you installed years ago. They're constantly in a state of updates and fixes, and simply can't keep up with ever-changing customer expectations.
- Your business team has ideas that take advantage of ongoing market trends and needs to quickly craft workable applications, but lack the technical expertise.
- You think app roll-out would be much easier if your business stakeholders could create prototypes of applications as they envision them, to share with IT teams.
- Your business uses a variety of off-the-shelf solutions to get projects off the ground. Shadow IT threatens internal compliance and security protocols.
- Separate teams of programmers are required to build for each platform, as the technicalities of each one are different. This adds to your expenses and prolongs app deployment.
- Your company has budget constraints, and cannot afford skilled or professional developers, or expensive tech solutions.
- UX is currently on the backburner and your applications fall short of user expectations.
- Your competitors are already using low-code platforms.

What can you build with a low-code platform?

Low-code platforms simplify and streamline the entire app rollout process, offering several advantages over traditional development technologies. Let's take a look at how they compare:



Customer engagement applications

Build enterprise-grade applications that enable your customers, and partners, to better communicate and perform business transactions. Apps that promote personalized customer engagements help to improve satisfaction, retention, and revenue.

- Customer service management
- Appointment booking
- Point of sale

Line of business applications

Design applications that automate processes to increase the efficiency of your business operations. These apps provide several benefits to a business, including reduced costs and reduced risk of error through automation.

- Operations management
- Supply chain management
- Accounting and financing

Workplace applications

For your workplace, define workflows and build processes for tasks of any complexity, and automate operations across several departments.

- Asset management
- Approval process management
- Task management
- Customer relationship management

