appian

The State of Low-Code
for Developers

Contents

To know low-code is to love low-code.

Use of low-code has exploded in recent years and continues to expand at a remarkable pace. IDC predicts that the global population of low-code developers will have a compound annual growth rate of 40.4% between 2021 to 2025.[1] That's over three times the rate for the general developer population. For developers, this is an incredibly important shift, marking the beginning of a new era in software engineering.

To examine the impact of low-code on developers' careers, Appian commissioned an extensive study of software developers with a wide range of backgrounds and experience. The survey revealed that the respondents who incorporate low-code into their toolkits have higher job satisfaction, a more positive outlook toward future earnings, and better career prospects when compared to respondents who exclusively use traditional high-code approaches.

The survey also revealed initial skepticism around low-code. Sixty-four percent of developers surveyed said they were skeptical of low-code before learning it. But once they learn low-code, that skepticism often gives way to more positive outlooks on low-code's ability to deliver powerful, innovative projects on time.

Low-code is becoming an increasingly common, must-have tool for developers. Gartner® predicts that "by 2025, 70% of new applications developed by enterprises will use low-code or no-code technologies, up from less than 25% in 2020."[2]

This report breaks down why low-code users are more optimistic about their careers, why they're more satisfied with their jobs and the projects they work on, and how low-code resolves some of the key issues tied to a traditional high-code approach.

The study.

Fielded by Renegade LLC via Propeller Insights on behalf of Appian, this self-administered survey was taken by 403 software developers across the United States. Those surveyed worked in a wide range of industries, with the largest proportion being from tech, financial services, healthcare, or retail. The pool of respondents included currently employed developers representing a mix of seniority levels and company sizes.

1. "Understanding the Significance of the Low-Code Developer Forecast," IDC (March 2021): https://www.idc.com/getdoc.jsp?containerId=US47546621

2. "Gartner Risks and Opportunity Index: Low-Code Platforms," Fabrizio Biscotti, Paul Vincent, Jason Wong, Laurie Wurster (June 2021): https://www.gartner.com/en/documents/4003142/risk-and-opportunity-index-low-code-application-platforms. GARTNER is a registered trademark and service mark of Gartner, Inc. and/or its affiliates in the U.S. and internationally and is used herein with permission. All rights reserved.

The global population of low-code developers will grow at 3x the rate of the general developer population over the next 4 years.[1]
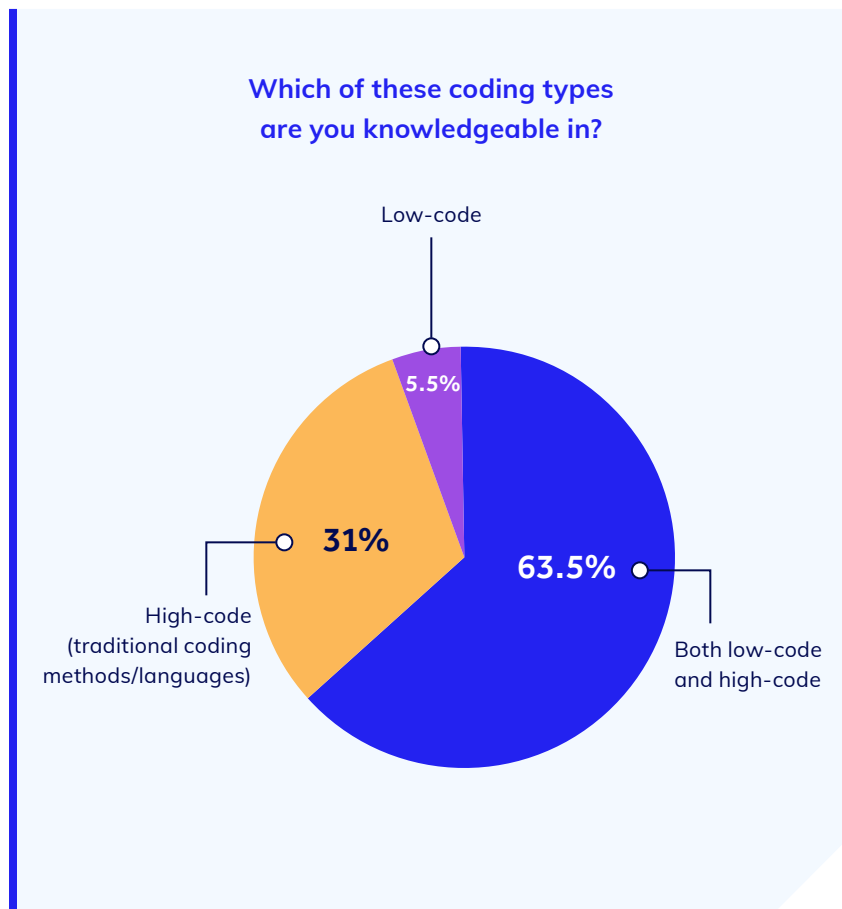
- IDC

The goal of this study was to compare the experiences of developers who exclusively use traditional high-code methods with those who use low-code as part of their jobs. Of the respondents, 31% exclusively use traditional coding methods, while 69% have at least some low-code skills (and most of that group use a mix of both low-code and high-code tools).

Key findings:

- Low-code users are, on average, more satisfied with their jobs than high-code-only users (42% vs. 31% report "very satisfied").

- More low-code users report base salaries of $100,000+ than high-code-only developers (72% vs. 64%).*

- A majority of both low-code users and high-code users believe low-code enables faster delivery of projects (79% of all respondents).

- 64% of low-code users were skeptical of low-code before learning it.

Which of these coding types are you knowledgeable in?

Low-code

5.5%

31%

High-code
(traditional coding
methods/languages)

63.5%

Both low-code
and high-code

69% of developers surveyed use low-code, and most of them use a mix of both low-code and high-code to deliver projects.

*The majority of developers surveyed had at least 6 years of experience.

Low-code's presence in software development.

Low-code adoption continues to expand as a valuable option for developers. More and more, developers are finding that embracing low-code as an addition to their high-code skills helps them deliver powerful software faster.

Low-code has become an important tool for many developers. It's often used as a supplement to high-code approaches and sometimes as the primary driver of development efforts. Those who adopt low-code use it most effectively as a way to augment their existing high-code skillsets. These tools offer flexibility and help automate and/or streamline large portions of typical workflows while still allowing room for high-code development where necessary.

Low-code and developer satisfaction.

When comparing the two groups—those who use low-code in some capacity, and those who exclusively use high-code—the low-code users reported higher job satisfaction and higher satisfaction with the projects they work on, and they were more likely to report working on innovative and mission-critical projects.

The consistency with which low-code users report higher levels of satisfaction is noteworthy: 87% of developers with low-code skills enjoy the work they do with low-code, likely because it is easier, faster, and simpler.

Low-code users tend to be more satisfied with their current jobs than those who only use traditional coding methods. Developers who use low-code report working on more innovative and important projects when compared to developers who exclusively use high-code. Since low-code accelerates tedious development, it frees developers up to focus on more interesting work.

> Low-code developers are, on average, more satisfied with their jobs and happier with the types of projects they get to work on compared to high-code-only developers.

Survey Question	Low-Code Users	High-Code-Only Users
Are you satisfied with your current job?	42% are very satisfied.	31% are very satisfied.
How satisfied are you with the type of programming projects you typically work on?	44% are very satisfied.	31% are very satisfied.
Do you work on innovative projects?	75% said yes.	64% said yes.
Do you work on mission-critical projects?	60% said yes.	40% said yes.

Many benefits of low-code tools contribute to higher levels of developer satisfaction. Speed is a core promise of low-code that simplifies not only the delivery of a project, but production and maintenance as well. Reusable low-code components, simplified process modeling, and integrated bot deployment also improve organizational automation efforts. Furthermore, those pre-built components, UI templates, and clear data aggregation greatly reduce the risks of human error and help prevent and resolve buggy code. These features allow developers to work quickly and effectively, freeing up considerable time to focus on more innovative or important projects—thus making for more enjoyable, rewarding jobs.

Career development and earning potential.

Developers of all types—both low-code and high-code users—recognize the positive impacts low-code has on careers. Seventy-one percent of all developers surveyed believe that low-code will help them achieve their career goals faster. Furthermore, 82% of all developers believe using low-code will increase their earning power.

Somewhat surprisingly, developers who only use traditional coding methods still recognize the career value of low-code: an impressive 64% of high-code-only developers believe that being able to use low-code would increase their earning potential.

This indicates a clear shift in the industry as low-code becomes more common. Developers who don't currently use low-code see the writing on the wall, while current low-code users actively enjoy the benefits low-code offers.

Even high-code-only users see low-code's benefits. Roughly two out of three high-code-only developers (64%) believe the ability to use low-code tools would increase their earning potential.

Survey Question	Low-Code Users	High-Code-Only Users
Will low-code help you achieve career goals faster?	84% said yes.	45% said yes.
Is low-code certification important for career development?	83% said yes.	47% said yes.
Will being able to use low-code tools increase your earning potential?	89% said yes.	64% said yes.

Embracing low-code also leads to real compensation benefits, and the majority of respondents recognize this. Eighty-one percent of all developers polled believe that knowing low-code will increase their earning potential, with 86% of those developers believing their earnings will increase by 10–20%.

Survey Question	Low-Code Users	High-Code-Only Users
Do you have a base salary of over $100,000?*	72% said yes.	64% said yes.
Compared to a year ago, has your salary increased 5–10%?	37% said yes.	26% said yes.
Compared to a year ago, has your salary increased 11–20%?	25% said yes.	14% said yes.

Such widespread acknowledgement of low-code's importance in growing a career tells a clear story. Across all portions of the developer world, a majority of people—including those who don't yet use low-code—believe that low-code is an important factor in earning potential and career progression moving forward.

*The majority of developers surveyed had at least 6 years of experience.

Low-code users report higher base salaries than high-code-only developers (72% vs. 64% make over $100,000). This is especially impressive considering the low-code respondents have less tenure than high-code-only respondents (35% of high-code-only developers have over 16 years of experience, compared to 11% of low-code developers).*

Agility without compromise.

Once a developer has low-code in their skillset, any early skepticism tends to give way to some very clear benefits, and a common sentiment among low-code users becomes clear: to know low-code is to love low-code. The functionality and development agility that low-code provides are making it a crucial part of any development career.

Speed.

One of the most commonly mentioned benefits of low-code is speed. Seventy-nine percent of all respondents believe low-code enables faster delivery of projects. Even when looking at responses from only the high-code users, it's notable that roughly two-thirds of them (68%) said low-code leads to faster projects. Perhaps these developers are seeing their colleagues use low-code to get faster results. After all, 67% of low-code users say they typically complete projects on time. Only 58% of high-code-only users say the same.

While development teams across the board are no strangers to tough timelines, low-code users are able to stay on track more often. Another ringing endorsement: 66% of all surveyed believe low-code helps developers complete projects 10x faster—a stunning increase in efficiency.

> Low-code, in a lot of ways, can make things you commonly do much faster and much easier to understand. But it also doesn't mean what low-code enables is simple; it can still tackle the most challenging problems at enterprise scale.

—James, Senior Product Strategy Software Engineer

66% of all developers surveyed believe using low-code helps complete projects 10x faster.

" **"**

With [a low-code platform], there is a nice compromise of still being able to code and set up a general project structure, while being able to take advantage of the sheer speed and ease of use to rapidly prototype, develop, and iterate on changes much quicker than I may be able to develop with traditional programming languages.

—Amit, Project Leader

Innovation and power.

The incredible speed of low-code, however, doesn't come at the cost of functionality. When incorporating low-code tools into their approaches, developers say they are able to build out powerful applications. In fact, 83% of developers with low-code skills say they're able to deliver state-of-the-art solutions. It may seem counterintuitive that low-code can spur innovation, but this likely stems from a common misconception about low-code: that low-code can only handle basic development projects. While it is true that some basic low-code tools are best for simple jobs, powerful low-code platforms can also handle complex, mission-critical development. By supplementing traditional coding skills, low-code can supercharge developers' abilities. Powerful low-code solutions help creative coders eliminate mundane portions of their work—such as routine maintenance, fixing bad code, and inefficient workflow processes around app development. This grants coders more time to spend on creative, highly cognitive work and innovative projects.

Many of the developers surveyed validated the concerns around those tedious tasks. Forty-six percent of all respondents believe they are spending too much time on coding tasks that contribute little value, and 56% believe they spend too much time on repetitive coding tasks. Low-code alleviates these concerns by handling much of the rote, repetitive work for developers. Additional features like visual app designers, collaborative development tools, and seamless data integration also empower fast, effective development, further reducing time spent on the aforementioned tasks. This powerful functionality is likely the source of low-code user confidence.

83% of low-code developers say they're able to deliver state-of-the-art solutions.

" **"**

[With low-code], it's easy to deliver on what you promise and make clients happy. I feel very fortunate that I can be proud of the work I do, and that I can see the value provided to the customers.

—Dan, Project Leader

Misplaced skepticism and the rewards of low-code.

Given the consistent correlations between low-code and better job satisfaction, improved career outlooks, and optimism about project quality, it's noteworthy that 64% of developers admitted to being skeptical about low-code before learning how to use it. That skepticism is tied largely to lack of experience with low-code and a hesitancy to learn a new tool.

Low-code skepticism may also be rooted in something of an "old school" mindset that is resistant to significant changes. Thirty-five percent of high-code-only developers have been programming for 16 years or more, whereas only 11% of developers with low-code skills share that history. The average age of high-code-only developers also skews slightly higher than the average age of low-code developers.

Resistance to change and new technologies has long been a thorn in every business's side. "Tech adoption at an organizational level is often slow or even nonexistent," notes a Harvard Business Review article. "This keeps old legacy systems alive and hinders an organization from achieving its full potential efficiently."[3]

Indeed, one developer in the study even noted their main gripe with low-code was rooted in a feeling of "cheating." After all, low-code removes much of the leg work that traditional developers have spent years learning how to address. It may feel too easy when, in reality, it serves as an efficiency booster. All that frustrating, tedious leg work is still being done, only by a tool that eliminates the need for a human to waste work hours. That time can then be spent on more rewarding work—a truly crucial benefit many low-code users are recognizing in real time.

64% of developers admitted to being skeptical about low-code before they learned how to use it.

3.   "Why Do Your Employees Resist New Tech?" Frank-Jürgen Richter & Gunjan Sinha, Harvard Business Review (2020): https://hbr.org/2020/08/why-do-your-employees-resist-new-tech

Given the generally positive sentiments people express toward low-code tools once they've adopted them, it is likely that developer attitudes will continue to shift more toward low-code acceptance and adoption in the future. Those who take the time to learn low-code will reap the rewards: more innovative projects, vastly improved efficiency, a stronger career trajectory, and overall happiness in their roles.

Low-code is the future.

The Economist noted in a recent report that "IT departments are being urgently called upon to fast-track transformation." As software markets grow increasingly competitive, quick turnaround for development is no longer optional. The average development firm faces a 3–12 month backlog of planned IT projects.[4]

This need for quick turnaround means low-code is here to stay. Forward-looking developers are quickly adopting low-code, honing their skills, and seeing how it can open up a world of career opportunities while enhancing their day-to-day work. Even developers who work exclusively in high-code largely see how low-code can benefit them moving forward, improving speed to build and time to value without compromising on functionality. Changing long-held habits is always daunting, but it's clear low-code is worth it.

About Renegade.

Renegade LLC is a NYC-based boutique specializing in B2B brand strategy and market research.

About Propeller Insights.

Propeller Insights is a full-service market research firm dedicated to taking any business to new heights through quantitative and qualitative measurement.

Explore low-code, build your own apps, and learn using hands-on tutorials with the Appian Community Edition.

Create your free account

4. "IT's changing mandate in an age of disruption," Economist Intelligence Unit (2021): https://www.appian.com/resources/resource-center/whitepapers/economist-survey-its-changing-mandate.html

appian

Appian is the unified platform for change. We accelerate customers' businesses by discovering, designing, and automating their most important processes. The Appian Low-Code Platform combines the key capabilities needed to get work done faster, Process Mining + Workflow + Automation, in a unified low-code platform. Appian is open, enterprise-grade, and trusted by industry leaders. For more information, visit www.appian.com.