Vahana Cloud

Introduction to Low-Code Development

Vahana Cloud

01 What is Low-Code Development?

Low-code development platforms are visual software development environments that enable both enterprise developers and citizen developers to connect application components using drag and drop functionality, create mobile or web applications, and more, all with minimal lines of code.

As a result, drag-and-drop functionality and pre-configured components are used on low-code and no-code platforms instead of conventional programming languages like JavaScript. This not only makes the process of creating applications easier and more accessible for novice developers, but it also vastly speeds up the process.

Even those without prior software development knowledge may easily build applications that are dependable, platform-independent, cloud-ready, and efficient with little to no coding.

Low-code/no-code platforms stem from earlier rapid application development (RAD) tools such as Excel, Lotus Notes and Microsoft Access that likewise put some development-like capabilities into the hands of business users (i.e., non-IT professionals).

However, those tools required users to thoroughly understand the business apps and their development environments in order to build capabilities. In contrast, with low-code and no-code options' drag-and-drop features, users need either minimal or no knowledge of the tools or development in general.

02 What does Low-Code Development look like?

Here's how a typical low-code platform looks like:

- **Front-end**
An interface for visually defining the UIs, workflows, and data models of your application.

- **Middleware**
Connect with external systems or convert your legacy APIs to modern APIs and manage API Orchestration

- **Deployment & Monitoring**
Tools that are automated for testing, staging, and maintaining the application as well as developing, debugging, and deploying it.

03 What problems does Low-Code solve?

IT companies are under increasing pressure to provide cutting-edge solutions. Only a limited number of top-tier businesses, however, can use traditional development to match the market's desire for speed.

Most businesses are constantly expected to perform more with fewer resources while also being burdened by massive backlogs and having difficulty finding skilled employees. The epidemic also taught us how important it is for businesses to be flexible in order to respond to changing and unanticipated needs.

Companies of any size that employ low-code can boost developer productivity and speed because it significantly decreases the complexity of software development. It increases the value of developers by letting agile teams utilize their knowledge of how to build and maintain high-quality online and mobile applications while stretching their technological boundaries.

With low-code, a UI/UX designer can do front-end development and a back-end developer can try prototyping a consumer app.

Simply said, low-code is a method for coders to complete more work. Low-code allows them to spend less time on tedious labor and more time building and inventing. Even while it's entertaining to experiment with a cutting-edge NoSQL data store or learn the newest faddish JavaScript framework, your competition is already in front of customers with an MVP while you're wasting time debugging unfamiliar code.

Vahana Cloud

04 Why is Low-Code development the future?

Low-code technology is being embraced by almost all industries. Gartner predicts that by 2024, at least four low-code development tools will be utilized by 75% of major businesses.

The global low-code platform market is forecasted to generate a revenue of $187.0 billion by 2030, increasing from $10.3 billion in 2019, and is expected to reach 31.1% CAGR in the forecast period (2020-2030).

According to Gartner, 65% of application development will be low code by 2024.

According to Gartner research, on average, 41% of employees outside of IT – or business technologists – customize or build data or technology solutions. By the end of 2025, 50% of all new low-code clients, according to Gartner, will come from business buyers outside the IT groups.

A survey by No-Code Census in 2020 showed a 4.6X productivity gain over traditional programming.

According to Gartner, the market demand for app development services will grow at least 5X faster than IT capacity to deliver them. No-code is here to help software developers fill this wide IT gap, enabling business users to develop solutions on their own.

It is also predicted that the citizen integrator tools are forecasted to reach mainstream adoption in the next 2 to 5 years.

05 Benefits of Low-Code development

Here are some of the most significant benefits of using a low-code development platform:

- **Faster time to market**

Low-code platforms allow IT teams to build applications quickly and easily. It reduces the delivery time of new applications and the manpower

- **Lower Costs**

Low code development platforms allow seamless systems integration and cross-platform workflow automation reducing costs for cross-platform development, integration, and testing.

- **Rapid Prototyping**

Low-Code Development platforms allow faster development of quick and dirty prototypes, using Rapid Application Development to create super-fast software reducing functionality glitches and the time taken for testing .

- **Cloud Hosting**

Low Code applications are agile – you can make full-stack applications in days and make changes to them in a matter of minutes. You can develop world-class cloud applications with shoestring budgets.

- **Cross Platform capabilities**

With Low Code, you can create single-click cross-platform deployments. You can also make changes that reflect across platforms simultaneously, rather than having to code them separately in each platform.

- **Legacy Integration**

Low code helps integrate legacy systems built on traditional methods of coding with newer technologies in a quicker, cheaper, and cleaner way.

06 Low-Code vs No-Code Development

Very often, the terms Low-Code and No-Code platforms are used interchangeably, but they have significant differences.

Looking at the fundamentals, Low-Code development requires little to no coding to build applications and processes. Where as, No-code platform uses a visual development interface to enable non-technical users to build applications without any previous coding experience.

Below are the differences of Low-Code & No-Code:

	Low-Code	No-Code
Users	Both IT users and business users	Citizen Developers and non-IT business teams with no coding knowledge
Coding needed	Low coding needed	No coding needed
Customization	Total customization available	Pre-built templates can be customized
Usage	Can be used for both tactical as well as business –critical apps	Used for making custom and simple apps which are not mission-critical
Level of Integration	Applications can be integrated with little coding to other business applications	Standalone applications
Cost effectiveness	Cost-effective for a company with a team of existing developers	Cost-effective for a company with a backlogged IT team and high requirements

07 Challenges of Low-Code Development

Despite the rapid growth of Low-Code platforms, several significant challenges remain, as listed below:

- Although low-code platforms facilitate the development of application software, they do require some coding, and only a small percentage of proficient developers are Low Code experts.
- In some scenarios visual development just doesn't cut it for complex use cases. Using a visual language like Business Process Modeling Notation (BPMN) to build business logic becomes so complex and unwieldy, that it would be much faster and easier to just write a simple piece of code to solve the problem.
- A common concern is that you cannot get readable or usable code out of a low-code platform, making it very hard to move your app to a different platform which creates vendor lock-in risks.
- Integration is a point of concern. Sometimes it's just not possible to integrate with some systems due to limitation in your low-code platform.

Vahana Cloud

08 How should one choose a Low-Code platform?

While choosing a low-code platform, you must search beyond the development speed and make sure that the platform employs standard technology.

- **Identify and evaluate multiple use cases:**
It's important to consider multiple app dev needs and use cases when researching and testing low-code platforms. Most important, discover what the platform cannot do or cannot do easily and gain a sense of its scope, strengths, and weaknesses.

- **Investigate and prioritize integration requirements:**
Applications need to integrate with enterprise systems, APIs, cloud and data center databases, and third-party data sources. Just about all platforms offer APIs, but what you can do with them, how well they perform, and how vendors support development teams vary considerably.

- **Review hosting, Devops, and governance options:**
Reviewing Devops options is another important consideration. Check whether the platform meets business and technical needs, not whether they conform to tools and processes engineered for coding and software development. Lastly, review the platform's citizen development governance options.

- **Understand compliance and security requirements:**
Understand how the low-code platform handles role-based administration, data masking, and other security considerations.

09 Low-Code technology enabling Banking Digital Transformation

The traditional banking industry is seeing a change like never seen before. More than 70% of key banking functions such as account opening, credit card applications, and loan processing are moving to digital.

Low code can assist banks with the massive digital transformation that they need today in the following ways-

- **Automate Application approvals:**

Custom-built banking low code solutions could help ease these processes by moving them to a digital platform and enabling cloud-based document upload and verification. For example, when Kotak Mahindra bank opted for a low-code solution to automate their customer onboarding and bring it online.

- **Modernize Existing Infrastructure :**

Banks are expected to provide a smooth, fast, and cloud-ready user interface being burdened with tighter regulatory compliances. This leads to IT teams making applications in haste that need to be re-engineered and refactored. Low Code platforms help to avoid technical debt because these applications use cloud-native and microservices architectures, coupled with custom integration to legacy applications and a reusable UI.

- **Preventing Financial Crime:**

Financial crime is on the rise across the world. According to a PwC survey, 46% of organizations have experienced it in the last two years. As IT teams rush to create new-age financial services applications and frequent updates to existing applications can cause breaks and vulnerabilities**. Low code banking** platforms help maintain legacy financial crime protection logic while building modernized applications, solving both problems simultaneously.

Vahana Cloud

Follow
Vahana Cloud
for more insights