Low-Code and No-Code Platforms

Low-code and no-code platforms let people build business applications by arranging ready-made blocks instead of writing every line of code. The term low-code was popularised by the analyst firm Forrester around 2014, and analysts such as Gartner now track the market. These platforms promise faster delivery and help business teams solve their own problems. They also bring new risks if nobody manages them. Samer, an operations officer, uses them to improve processes in his company.

1. What They Are

Low-code platforms use visual tools, templates and connectors, and developers add code only where it is needed. No-code platforms go further and let non-programmers build solutions entirely through menus, forms and drag-and-drop screens. People who build this way without a technical role are often called citizen developers. Both approaches hide the technical details, such as servers and databases, behind simple controls. The result is a working application, often in days or weeks instead of months.

Example: A teacher builds a trip sign-up form by dragging fields into place with an online form builder. Using a no-code tool, she needs no programmer, and the form collects the replies in a table.

At work: Samer uses a low-code platform to build a leave request app. Working visually, he drags forms and approval steps into place, and a developer adds one small script to connect payroll.

2. Typical Business Uses

Typical uses are internal tools that follow clear steps. Examples include request and approval forms, simple dashboards, data collection apps and automated workflows between systems. They work best for problems that are specific to one team and need to change often. They are less suitable for very large systems with heavy performance or security demands. Choosing the right use is the first step to success.

Example: A small shop owner tracks orders in a shared app made from templates instead of paper notes. Applying a typical business use, simple tracking, the owner sees every order's status at a glance.

At work: Samer's warehouse team needs a damage-report form with photos and a manager approval. Applying a typical business use, he builds the request-and-approval form in two days, and reports stop getting lost in email.

This document is a property of Asaseeyat Academy - © Copy Right Protected.

3. Governance Risks

The same ease of building can create problems. When many people create apps without oversight, the company may end up with hidden tools that the IT department does not know about, often called shadow IT. Apps can expose sensitive data if permissions are set badly, and they may stop working when their builder leaves. Some platforms also make it hard to move to another supplier, which is called lock-in. Spotting these risks early keeps the benefits.

Example: A club member builds a sign-up sheet that shows everyone's phone numbers to all visitors. Spotting this governance risk, the club limits access to the organisers only.

At work: A sales assistant builds a customer list app and shares it widely, and Samer notices that it holds personal data. Spotting this governance risk, he asks IT to review the permissions before more people use it.

4. Govern Wisely

Good governance does not stop people from building, it guides them. Companies set simple rules, such as an approved list of platforms, a clear owner for every app and a review of apps that use sensitive data. They train builders in basics like data protection and testing. They also keep a register of apps and plan what happens when an owner leaves. With these guardrails, speed and control can live together.

Example: A school lets teachers create class apps only on one approved platform, and each app has a named teacher as owner. Applying governance guardrails, the school can find and fix any app that causes trouble.

At work: Samer's company keeps a register of all apps, each with an owner and a data check. Applying governance guardrails, the register shows that an old app has no owner, so Samer assigns one before it breaks.

Conclusion

In summary, low-code and no-code platforms help people build business applications faster with visual tools. They work well for team-level tools such as forms, approvals and dashboards. Without oversight they can create hidden risks, such as exposed data and apps with no owner. Simple governance lets companies keep the speed and control the risks. Samer's leave app shows that a small, well-managed tool can save real time.

Sources

- Low-code development platform: overview of the concept, history and risks – https://en.wikipedia.org/wiki/Low-code_development_platform
- Forrester: The Forrester Wave, Low-Code Development Platforms (2016), an analyst report on this market – https://www.forrester.com/report/The+Forrester+Wave+LowCode+Development+Platforms+Q2+2016/-/E-RES117623
- Shadow IT: overview of hidden technology used without IT approval – https://en.wikipedia.org/wiki/Shadow_IT
- Forrester and Gartner are names of their owners. Our wording is our own explanation.

This document is a property of Asaseeyat Academy - © Copy Right Protected.