

Low-Code vs Traditional Coding, how do they differ?

Low Code

Traditional Coding



Skills required

Simple drag-and-drop environment, it can be used effectively by both software engineers and citizen developers.

Specialised software engineers, in high demand and expensive to hire.



Cost

The platforms **only require a fee for access to make applications**, which significantly reduces labour, maintenance and infrastructure costs compared to traditional means.

Applications must be built **from scratch**, which costs much more than **licensing** a low-code platform.



Speed

Low-code cloud-native applications can be developed **more than 10 times faster** than the traditional method.

Creating **platforms from scratch** is a **lengthy** and time-consuming **process**, taking years to complete.



Deployment

It allows you to create applications for **specific business needs**, one at a time, regardless of size.

Large-scale application development and long-term investment are necessary for even the smallest business process applications.



Outreach

Development sprints can be significantly reduced, resulting in **faster delivery and deployment**.

Development cycles are much slower, which means that deployment time is much longer compared to lowcode.

