There's no code like the low code

The growing need for mobile and web applications is driving development teams away from hand-coding to a new breed of application platforms. They are moving to platforms that accelerate app delivery across a multi-channel environment and that includes mobile and web applications.

© olly - Fotolia.com

According to Forrester Research, hand-coding is too slow to develop and to deliver many of the applications that companies today need to win, serve and retain customers. It says businesses are turning to new 'low-code' application platforms that accelerate app delivery by dramatically reducing the amount of hand-coding required.

OutSystems South Africa director Craig Terblanche says companies are investing in these software projects principally to enrich customer interactions. "However, they also drive innovations in customer service, brand connection, products and business models."

As software projects, customer-facing applications challenge all the norms of enterprise software development and solution architecture. Delivery speed is the biggest challenge, but in formulating strategies for customer-facing applications, application development and delivery (AD&D), leaders tackle some of the challenges.

Keeping up with customers

Terblanche points to the age of the customer. "Everything runs as quickly as customers and markets - and software delivery must keep up. To make software delivery as nimble as their customers' imaginations, application delivery leaders must slash project schedules to weeks for initial delivery and days for updates and fixes. Doing so requires completely different software structures and project methods."

Create software that is immediately intuitive. Consumer mobile apps reset the definition of intuitive software to mean, "No training required to be immediately useful." All customer-facing applications - even those used by employees - must meet this standard.

Transition application delivery methods must fit customer engagement models. Systems of engagement extend back-end applications and systems of record to make it easier for customers to engage with a company.

Many teams are applying Agile methods to coding frameworks like Spring Framework and .NET to deliver systems of engagement applications. However, making frequent updates and module deployments requires continuous integration methods that go beyond Agile.

"The trick is to find platforms that fit customer-engagement realities. Many development teams are re-evaluating the role Java, .NET, and other coding platforms should play in their customer-facing systems."

Time is of an essence

The control that coding provides comes at the cost of time. Application development platforms that rely on faster development techniques - visual development, automatic configuration and deployment, or user interface transcoding - can dramatically speed delivery compared with hand coding.

"We call these products low-code platforms and define them as platforms that enable rapid application delivery with a minimum of hand-coding and quick setup and deployment, for systems of engagement."

If hand-coding can't keep pace with the speed of change that customers and customer-facing employees demand, what are the alternatives? Some enterprises cope by outsourcing the challenge and some look to packaged applications and others to specialised middleware. None of these alternatives addresses the problem head-on. How can application delivery teams speed up their useful output?

"A promising approach is to use low-code platforms such as OutSystems, an emerging category that supports rapid assembly of customer-facing applications, requiring minimal hand-coding and enabling productive new development practices."

Low-code platforms speed up development by allowing development teams to eliminate barriers to customer participation in projects, as well as handoffs between phases of projects.

He says slash the hand-coding needed to deliver applications. "Low-code platforms minimise hand-coding and speed up delivery by providing visual tools for quick definition and assembly of user experiences and forms, rapid build out of multistage workflows, and easily configured data models that eliminate common data integration headaches."

Less skilled developers needed

Low-code platforms allow companies to succeed with less-skilled developers. However, some hand-coding is still required to deploy the application to production - but far less hand-coding than previously.

Business leaders trying to move rapidly on new ideas to boost revenue and improve competitiveness often are bogged down by rigid and siloed development approaches. Low-code platforms allow business leaders to experiment with new product and service ideas by merging requirements, design, development and deployment into a single platform.

This "sandbox" approach allows one- or two-person teams to compose new apps and quickly gain feedback from customers, employees and partners.

App can cross channels

Customers using the low-code application platforms are primarily focused on apps for the web channel. However, these platforms support responsive design and mobile-ready functionality that makes it as easy as pushing a button to extend the app to work across other channels, including tablets and smartphones.

Low-code platforms provide a unified and centralised environment for configuration management, role-based access, authentication and repository control of apps and configuration components. All key players in software delivery - enterprise architects, security experts, IT operations and business experts - can add their input to the configuration and management of the platform.

"Low-code platforms are a converging category, not a new one. Most vendors of these products established their technologies with internal-facing applications. Customer-facing applications are quickly becoming a popular new use case for these platforms because these types of applications demand rapid delivery and evolution," Terblanche concludes.

Customer-facing applications pose the most difficult challenges. AD&D teams urgently need to change to address these project needs, as well as to readjust the balance in their application delivery strategies between innovation and integrity, and between hand-coding and low-code tools.

For more, visit: https://www.bizcommunity.com