How to Develop More and Better Software and Not Die Trying

Gisela Bertelli | SVP Global Sales

GeneXus™

The recent acceleration of digitalization and technology modernization among businesses has aggravated a longstanding problem: software applications have grown more and more complex. The number of developers and engineers has not risen enough to keep pace with the complexity.

Unfortunately, the efforts to address the talent gap - like outsourcing or government promotion of software careers - have fallen short. So I want to talk about the emerging value of Low-Code platforms to address the imbalance.

The Rise of Complexity

Let's start by understanding what **growth in complexity** involves. Those of us who have been in the software development field for some years know that the skills required in the '80s and '90s consisted of mastering two or three technologies, and that the IT area defined the technology to be used.

Even then, there was a shortage of qualified developers. With the emergence of the web world, software engineering teams needed knowledge of new programming languages, architectures, and the acquisition of other skills such as interface design, security, etc.

Complexity has grown exponentially. In recent years, the availability of new platforms such as mobile devices has added to it, adding new languages, technologies for different devices, interface designs, and user experiences to the list of skills developers needed. In addition, end-users now had the power to choose the technology to be used and, consequently, the technology areas had to grow in order to support the various options on the market (Android, iOS, etc.).

This exponential growth has accelerated due to the need to develop multi-experience applications, including the use of mobile apps (online and offline), chatbots, AR, wearables, sensors, cloud, AI, etc., etc., etc.

Complexity has also increased in response to factors beyond the user experience. Applications need to integrate with different data sources, as well as with countless services and technologies on the back-end. Computer security has become a differentiator in the information age.

And software development is not confined to coding but also testing the developed software and deploying the solution in an agile process (DevOps).

In summary, it's no longer possible for a single person to master all this knowledge. More experts are required to complete the development of a solution.

The development complexity curve trends upward.

The Demand Curve: What Businesses Need From Their Applications

The evolution of the second variable reflects the **increasing needs** of a company's different business areas. Several years ago, probably the biggest challenge of businesses linked to the IT area was related to selecting a platform and developing an ERP, HR, CRM system and/or some other one.

Those choices have had massive consequences in recent months. The pandemic has accelerated the notion that all areas of a company must be redesigned with the use of technology, and even reinvent the business model in order to survive.

Users are definitely the ones who define their needs and, in many cases, determine the technologies in which they want to run their daily applications. In addition to this, users define ever-changing requirements at the pace of reality and want their applications to adopt the new requests and be flexible in the face of frequent changes in a highly uncertain context. It is obvious that software development has become critical in all organizations. Every company, regardless of their industry, has transformed into small "software houses."

Finally, an analysis of the **increasing number of programmers**[1] all over the world reflects a modest acceleration in the last few years. However, despite the various efforts made by governments and IT industry associations to promote careers linked to programming, the gap is still enormous.

Other strategies have faced headwinds as well. Many companies have met some of their needs by outsourcing software development to countries like India. However, today the acceleration of production timelines and the need for flexibility make this strategy difficult to sustain. In-house software development teams are likely to be the norm in the coming years.

In the face of this reality, what options are there?

The Solution For New Solutions

Probably a clear path is to move forward in the **automation** of software development. If we put it in perspective, in recent years most industries have been able to achieve automation, to a greater or lesser extent. However, software development is still the same as it was originally, where lines of code have to be written by hand to represent reality. Years have passed since the emergence of the software development industry, a new century has started, and yet we continue to develop software in the same way.

Is it possible to use robots and/or artificial intelligence to build software in order to meet the multiple needs of users efficiently and effectively? Are the so-called "Low-Code Platforms" a valid option?

I am convinced that they are, and I encourage the software industry to reflect on this problem and how to make it possible to develop, maintain, and evolve software automatically, in order to achieve innovation, but also in the way those solutions are built.

[(1)]Forrester estimates that by 2024 there will be a shortfall of 500,000 software developers in the US. View report

GeneXus™

MONTEVIDEO - URUGUAY	Av. Italia 6201- Edif. Los Pinos, P1	(598) 2601 2082
CIUDAD DE MÉXICO - MÉXICO	Hegel N° 221, Piso 2, Polanco V Secc.	(52) 55 5255 4733
MIAMI - USA	8950 SW 74th Ct, Suite 1406	(1) 201 603 2022
SÃO PAULO - BRASIL	Rua Samuel Morse 120 Conj. 141	(55) 11 4858 0300
TOKYO - JAPAN	2-27-3, Nishi-Gotanda	(81) 3 6303 9381
	Shinagawa-ku, Tokyo, 141-0031	(81) 3 6303 9980