HCLSoftware

HCL Volt MX

5 Secrets to Citizen Development Success

Empowering your teams in harmony with IT and with the right governance

Table of Contents

03 | Introduction

04 | Empowering citizen developers, the fast way

05 | Citizen development success, demystified

Five secrets to citizen development success

06 | #1 Provide the right tools for the right team

07 | #2 Provide a safe, governed environment

08 | #3 Enrich the development toolset

09 | #4 Create a low-code center of excellence

10 | #5 Facilitate collaboration between citizen and professional developers

12 | Lufthansa: A case study in user empowerment

13 | Empower your citizen developers with Volt MX

Introduction

To maximize the impact of digital transformation, organizations should empower all employees to innovate and solve business problems. Relying solely on professional developers and IT to build applications might limit a company's potential to stand out in the digital economy. Many innovations originate with business users, the "citizen developers", who know the challenges of their day-to-day tasks and can identify opportunities for improvement.

Still, many companies struggle with just how to systematically engage and empower their citizen developers. And those that already enable citizen developers through their low-code platform investments still face limitations or haven't achieved the return on investment (ROI) they were looking for.

One of the barriers is that most low-code platforms in the market remain too complex for non-technical users. They are typically watered-down versions of professional development environments which require steep learning curves, discouraging widespread adoption. Equally, some low-code platforms impose limitations that are not immediately apparent when business users go outside of IT to secure a solution.

Governance is also a concern. Business leaders worry about compliance issues when giving non-IT users access to enterprise data. Certain types of data are subject to regulatory and business controls and require higher levels of security and handling. But with the right low-code platform in place, organizations can provide citizen developers with safe, governed environments to accelerate innovation and address business challenges.

Empowering citizen developers, ***the fast way***

IT alone cannot drive a modern organization's digital transformation strategy. IT departments have enough on their plates (and in their backlogs) with day-to-day tasks, corporate application initiatives, technology updates, and security responsibilities. Besides, innovation thrives on diversity. The best ideas for digitalization or streamlining of business processes come from those with expertise in their specific business domain.

When organizations turn these business domain experts into citizen developers, they have much to gain by unleashing otherwise untapped potential. As such, organizations can increase employee productivity, foster collaboration among teams, and accelerate time to market with solutions that boost corporate agility and competitiveness.

Empowering citizen developers produces a culture of innovation — a can-do attitude among employees tackling problems and executing ideas to make tasks more efficient and enhance customer offerings. Employees become more engaged with the organization's goals by taking ownership of their work processes. They can turn ideas into hundreds, even thousands, of new applications or processes that solve problems. Digital transformation becomes a cross-organizational collaboration, rather than a high-level process dictated by IT and the C-suite.

A low-code development platform is key to making all this happen. With the right platform, the necessary IT governance can occur in a controlled way without stifling collaboration and ensuring precious professional developer resources are used only when needed.

> *A **citizen developer** is an employee who creates application capabilities for consumption by themselves or others, using tools that are not actively forbidden by IT or business units. All citizen developers are business technologists. There is no required designation of proficiency or time allocation for citizen developers.*
>
> *— Gartner*

Citizen development success, *demystified*

Even when an organization envisions the potential benefits of citizen development, leaders may be reluctant for several reasons.

Some fret about too many low-code and no-code tools adding to their technical inventory. Most are skeptical about the extensive training and support needed for their citizen developers for the more complex tools. Concerns over the risk of shadow IT and lack of governance are common.

Common citizen development deterrents:

- Steep learning curve
- Concerns about building and sustaining a new culture of innovation
- Aversion to adding more tools and complexity to app dev
- Risk of shadow IT and lack of governance
- Potential for technical debt
- Security, data privacy, and compliance concerns

However, it is possible to implement citizen development in a safe, governed way by following these five best practices.

#1 Provide the right tools for the right team

#2 Provide a safe, governed environment

#3 Enrich the development toolset

#4 Create a low-code center of excellence

#5 Facilitate collaboration between citizen and professional developers

#1 Provide the *right tools* for the *right team*

Citizen development starts with providing employees with the capabilities they need. To maximize productivity and efficiency, the tools and platforms citizen developers use should require little to no training.

Anything that demands extensive learning is bound to discourage user involvement. Business users don't want to be mired in a complicated onboarding process. The on-ramp has to be easy, enabling them to get started without IT assistance to eliminate repetitive tasks. They want to be able to start automating or digitizing their business that can be measured in hours and days and not in weeks and months.

Aesthetics also make a difference. The interface should be well-designed and attractive to encourage use, especially in a browser-based, self-service environment where employees can easily access data and services to design, deploy, and manage applications. And to further simplify things, development experience should be web-based, hence, no desktop downloads should be required.

The application and workflow-building process should be straightforward. Web forms should make it easy to collect, store, and access data. Creating workflows for approvals should be intuitive. In addition, access to data sources, such as an employee directory to incorporate in workflow stages, also should be straightforward. The same goes for enabling users to view and analyze data — the tasks should be simple and quick.

Lastly, the platform should provide a simple transition process for data from users' existing tools and platforms, such as Excel spreadsheets, email, and SharePoint. Importing the data into your new platform should be simple, requiring only a click or two.

#2 Provide a *safe, governed environment*

Citizen development should occur in a safe, governed environment to avoid running afoul of regulatory and corporate compliance or creating cybersecurity vulnerabilities. As such, companies need to make important decisions upfront on who can use the development environment and how to authenticate the users.

Even if an organization wants to empower all users for citizen development, first it may want to test the waters to ensure the environment works as expected. Conducting a pilot in a single department, such as marketing, can help determine how to establish processes and policies for citizen developers. The company may want to store data collected by user-built applications in databases with encryption for data at rest, and limit access to the database for certain users.

Businesses also need to make decisions on what types of files and sizes can be stored, and what domains are allowed for resources such as images. Capacity limitations should be set to avoid performance degradation that can occur by exceeding limitations.

Guardrails should be in place for the whole environment to minimize risk by, for instance, limiting the possibility of malicious code to enter the environment. This may include whitelisting the types of files that can be attached and limiting database access to prevent unwanted and unsafe changes.

Implementing controls to ensure governance and safety doesn't mean preventing citizen developers from making their own decisions. The platform should allow flexibility to allow citizen developers to be in control of their own apps such as deciding co-authors of the app, who within the organization is authorized to use the application once deployed, and what authorized users can see in the app and do with it.

#3 Enrich the *development toolset*

For best results, IT should play a role in setting up the tools to ensure simple, but controlled, access to sanctioned:

- *data and connectors*
- *custom widgets, templates, and themes*

Merely providing citizen developers with out-of-the-box tools is not likely to produce the best applications users can build. Keeping in mind the typical citizen developer has limited technical skills, IT should make it simple to access the tools and services that users need for their applications. It should also ensure that these tools adhere to the organizational requirements, looks, and use cases.

Data sources such as an official customer list or a catalog of product names and part numbers should be easy to access and navigate — these sources are created specifically with the citizen developer in mind. When tools are integrated, users become more productive and effective. More importantly, this ensures proper access and governance of data used by citizen developer applications.

Widgets enable consistency in look and behavior when introducing new functionality. They enrich the citizen developers' toolset by providing shortcuts to complete projects in conformance with organizational preferences.

Templates are another type of shortcut. Users can access integrations and widgets and place them in a template so they can instantiate applications with a click or two. Templates have built-in custom logic, making them ready for use by non-technical employees.

Custom Themes function much the same way as templates. The time and effort of creating a look for an application is replaced by pre-approved corporate themes that meet a company's branding and functional requirements.

#4 Create a low-code *center of excellence*

When users feel supported, they are more likely to ask questions and make suggestions. And this leads to a more innovative, dynamic citizen development culture that delivers better outcomes.

It starts at the top. When sanctioned and encouraged by leadership, the citizen development culture is more likely to thrive.

One of the decisions business leaders can make to show support is to appoint an IT sponsor to answer questions and help novice developers when they need assistance in completing tasks.

To encourage participation and collaboration, organizations should consider holding meetings at regular intervals — weekly, monthly, or quarterly — to discuss ideas, address questions, and come up with solutions to any problems that surface. As an alternative or addition, it also may help to create a support forum and knowledge base to share ideas, apps, and questions.

To further advance the cause of citizen development, companies can encourage the formation of user groups and post tutorials online for on-demand use by employees. Whenever possible, businesses should encourage participation from tool and platform vendors in forums and meetings to help users with questions and problems.

#5 Facilitate collaboration between *citizen* and *professional developers*

Effective citizen development requires a dedicated environment. Most platforms currently available use the same integrated development environment (IDE) for both citizen and professional developers.

Typically, the version targeted at citizen developers is watered down from the professional environment, ostensibly simplifying the environment with limited features for non-technical users. This approach falls short of truly unleashing the creativity that users can tap when given a rich, full-featured environment to develop their applications as the on-ramp to learn the tool is so high.

A more effective approach involves deploying a platform designed for both citizen and professional developers so they can collaborate as needed. At the same time, each group ought to have its own full-featured development environment designed specifically for their technical skill levels with appropriate education materials.

The implementation of a common platform with environments for both personas delivers the best of both worlds. Citizen developers get an easy on-ramp to a browser-based IDE to build their applications in a controlled, governed setting where they can make decisions without violating IT policies. Meanwhile, professional developers get a rich client-based IDE tied into DevOps, GitHub, and other standing DevOps best practices and tooling with full control of key aspects such as form factor, integration, and security.

For instance, a citizen developer may design an application for their department that may be leveraged by other business units. Let's say marketing creates an application that organizes customer details in a way that benefits other departments, such as services and sales. To make it useful to a wider group of users, a citizen developer may need assistance from a professional developer to achieve this.

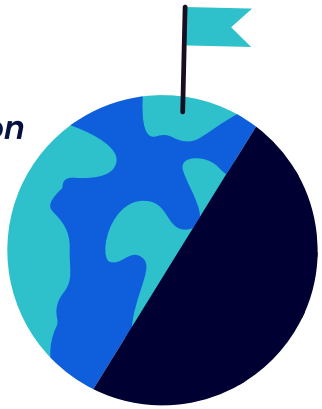
Then, a professional developer would take a user-created component and add it to an existing sales application, forming a super app. The professional developer could also extend the app onto iPad devices to give field workers access to it offline.

> *A **super app** is an application that provides end users with a set of core features plus access to independently created mini-apps. The super app is built as a platform to deliver a mini-apps ecosystem that users can choose from to activate for consistent and personalized app experiences.*

This fusion approach to development can significantly boost productivity, allowing professional developers to take over and extend the application without rebuilding it because it resides on the same platform. Besides boosting efficiency, it reduces potential technical debt.

By 2027, 50%+ of the global population will be daily active users of multiple super apps

— Gartner



Lufthansa: A case study in ***user empowerment***

Lufthansa is one of Europe's largest airlines, with 105,000 employees worldwide and 2022 profits of more than $1 billion. While undergoing digital transformation, the airline realized it needed to automate its processes because it remained too reliant on inefficient analog and paper-based tasks across the organization.

IT was getting inundated with requests for workflow automation from employees in just about every department, including technical maintenance services, catering and flight operations. The list of requests to eliminate repetitive tasks kept growing. IT couldn't keep up with it all. So Lufthansa decided to implement a low-code app development strategy to empower business users, i.e., their citizen developers, to create and manage their own process automation.

After trying other vendors, the airline decided on HCLSoftware's low-code application development platform, making it available to all employees without restrictions. The move produced quick results, as citizen developers went to work on creating more than 2,000 applications, some of which are business critical.

2,000+ apps *created*

Build times: *up to* ***70% faster***

"Within months, the system had taken off with dozens of applications springing up across the enterprise."

—Ralf Schliepat, senior consultant at Lufthansa

Examples include the Net Correction App, which transformed how Lufthansa's air freight unit accepts and invoices for deliveries, and the Personalized Name Badge App, which provides flight attendant name badges showing which languages they speak.

With applications such as these, Lufthansa created a more efficient workplace by unleashing the creativity and problem-solving talents of its employees. Without writing a single line of code, employees can build applications up to 70% faster, bringing agility and efficiency to the organization.

Empower your citizen developers with **Volt MX**

As the Lufthansa experience demonstrates, empowering businesses to tap into their employees' creativity and problem-solving skills can transform an organization to better compete in a dynamic market. Over-reliance on IT to solve every automation and digitization challenge only goes so far; companies need to harness the strength of their most important asset — the employees — to come up with the best solutions.

From business to IT, HCLSoftware's Volt MX provides a unified development experience for both citizen and professional developers to collaborate and build engaging experiences — all with IT oversight. Volt MX is the industry's first platform to provide persona-specific developer experiences on a single platform with transparent pricing and no hidden costs.

Volt MX supports the continuum of developers in the following ways:

Citizen development: Volt MX provides business users of all skillsets to create situational apps in a safe, governed environment. Citizen developers can build and deploy apps in minutes, without coding.

Professional development: Volt MX combines the speed and ease of low-code development with enterprise backend services such as integration and authentication to maximize developer productivity. Highlights include support for multiexperience apps on one code base, super and micro app development, and workflow and testing automation.

Business and IT collaboration: Volt MX provides a single platform for seamless collaboration between teams, allowing IT to easily extend apps built by business users while avoiding rewrites and technical debt at the start of the project.

Ready to take your organization to the next level?

Learn more

Try out Volt MX for free