

unqork



# A Guide to the Enterprise No-Code Journey

What to expect as you integrate no-code into your organization.

## Contents

Introduction	3
Common Myths About No-Code	4
Progression of the No-Code Journey	6
Why Unqork?	12

**Businesses are under relentless pressure to digitally transform and enhance their entire organization**—it’s the getting there that is fraught with challenges, e.g., addressing rising expectations of users, navigating complex and increasingly sprawling digital ecosystems, and competing for scarce IT talent just to name a few. And this is why many of today’s leading enterprises are turning to no-code.

No-code is a new class of cloud-based development platform that replaces editable codebases with a completely visual UI. No-code platforms come “out-of-the-box” (or, out of the virtual SaaS box) with all the toolsets and industry-specific elements necessary to build and manage a robust application (e.g., front end UI, workflow, rules engine, analytics, visualizations, integrations, and maintenance). Since they’re all components of the same unified platform, everything works together in instant harmony. This means companies are freed to focus their efforts addressing business challenges, rather than technical ones, resulting in a greatly accelerated time to market.

No-code empowers organizations to expand the scope of who is doing the development. In a no-code platform, users (or “[Creators](#)” as we refer to them at Unqork) build applications by drag-and-dropping configurable elements that represent both user-facing features and back-end application logic. This allows companies to open up their workforce to include less-experienced developers in addition to those who have spent years building enterprise applications.

These technologies are already being used by global-leading enterprises across a wide variety of industries to rapidly develop—and effectively maintain—digital solutions that [automate complex back-office processes](#), deliver powerful [new front-end functionality](#), and can be quickly adapted [around disruptions of any scale](#).

No-code isn’t a new language or some tool that injects iterative improvements—it’s an entirely new development paradigm. In our experience, many IT decision-makers have questions about the process and ramifications of adopting no-code into their organization. In this eBook, we will attempt to address common misconceptions and provide a high-level overview of the enterprise no-code journey.

# Common Myths About No-Code



## “We have to replace all of our developers”

Just because you know how to build an application with no-code, doesn't mean you necessarily know how to build a value-additive application with high-quality user experience. In our experience, the best no-code Creators are those who already have a foundation in enterprise software development.

No-code should be viewed as a means to amplify the productivity of veteran engineers while empowering newer programmers to build beyond their experience.

Don't view no-code simply as a replacement for experienced coders. Rather, it should be seen as freedom to use development resources more judiciously—lean on your experienced programmers to build amazing new enterprise-scale applications quickly, while entrusting newer engineers with maintenance and non-critical updates.



## “It's only for simple projects”

This is certainly true of the first generation of no-code platforms, but today, it's highly dependent on your project and chosen no-code vendor. While no-code may not be the best choice for literally everything your company needs (for example, a space shuttle operating system), today's enterprise no-code platforms can help you efficiently build sophisticated customer- and internal-facing applications.



## “It’s hard to learn”

While modern programming languages can take several months to learn and years to master, most Creators are up-and-running on no-code in a matter of weeks. In the case of Unqork, we can provide detailed virtual instruction and structured classes (we hope to return to in-person on-site instruction as soon as circumstances allow). We also provide detailed documentation for all your needs via [Unqork Academy](#).



## “It’s not flexible”

Just about anything that can be done with code can be accomplished via repeatable configurable components—and much faster. The Unqork platform has been used to address business use cases from across multiple industries. Indeed, we’ve proven that [Unqork is Turing Complete](#), meaning it can actually be used to build a version of itself.



## “We will lose control of our digital infrastructure”

There is a common misconception that no-code leads to “the wild west” where all users will just be implementing software in their department however they seem fit. This couldn’t be farther from the truth. Building with no-code requires a deliberate and organized approach, and won’t necessarily be accessible to self-styled “citizen developers.” Indeed, Unqork comes with centralized gatekeeping functionality such as [RBAC](#) and our new [Workspaces](#) feature-set—these ensure that centralized administrators can oversee and control the company’s development function.

Furthermore, many “shadow IT” projects are often born out of desperation—business units who can no longer afford to wait for their needs to be addressed by an overwhelmed IT team. No-code accelerates development processes, so IT resources are able to work on a greater number of projects and move at the speed of business.

# Progression of the No-Code Journey

## PHASE ONE

### Demonstrating Value

**Owner:** No-code provider or SI

**Goals:** Gaining first-hand experience with no-code development

**Workforce:** Most work is handled by the provider or SI with constant communication with an internal IT champion and core team

**Number of projects:** 1 or 2 simple projects

**Scope:** Medium complexity

**Focus:** Learning

**Engagement with no-code provider or SI:** Very involved

## PHASE TWO

### First Steps

**Owner:** IT

**Goals:** Establish IT's no-code abilities

**Workforce:** A small "tiger team" within IT of 1-to-5 resources

**Number of projects:** 1-to-5 simple projects

**Scope:** Low-risk, not-mission critical

**Focus:** Technology

**Engagement with no-code provider or SI:** Heavy-to-medium

## PHASE THREE

### Scaling Up

**Owner:** Business and IT

**Goals:** Establish the capability to rapidly develop sophisticated applications based on the needs of the business

**Workforce:** 5-15 resources, both business and IT

**Number of projects:** 5-to-10

**Scope:** Simple-to-highly-sophisticated including mission-critical solutions

**Focus:** Business

**Engagement with no-code provider or SI:** Medium-to-limited

## PHASE FOUR

### Transformation

**Owner:** C-level sponsor

**Goals:** Establish the organization-wide capability to rapidly adjust digital infrastructure to the needs of the business

**Workforce:** Numerous IT and business resources organized into CoEs for different projects

**Number of projects:** 10+ projects per-team per-year

**Scope:** Simple-to-highly sophisticated including mission-critical solutions. Rapid prototypes.

**Focus:** Transformation

**Engagement with no-code provider or SI:** Little-to-none

## PHASE ZERO

### The Before Time

This is the phase where organizations decide to tap into the power of no-code to help them address long-standing challenges with their development function such as overwhelmed IT teams, a proliferation shadow IT, and long development times leading to missed revenue and opportunities. No-code represents a new paradigm, which provides increased flexibility for both the IT workforce—and the organization as a whole.

If you are currently in this phase, you may want to check out our [Field Guide](#) or [The aPaaS Buyer's Guide](#).

## PHASE ONE

### Demonstrating Value

This initial phase is heavily dependent on the no-code provider or SI. An internal no-code champion on the IT team will help them identify a business challenge requiring a digital solution of medium-complexity. The majority of the development work will take place from the external team(s). This initial phase is where organizations can determine if implementing no-code will allow them to address the needs and existing challenges of the organization.

#### RESOURCES

The internal no-code champion key IT members who will be in regular communication with the external teams.

#### NORTH STAR

The point of this phase is education and, if necessary, securing executive buy-in with a viable value-additive solution.



## PHASE TWO

### First Steps

The purpose of Phase 2 is to achieve comparatively small “wins” within the IT team. The chief activities of this phase are to get a select number of IT resources fluent in no-code and have them build “bite-sized,” non-mission-critical projects. These products should be relatively self-contained and require limited dependencies and integrations outside of the core team.

#### RESOURCES

Phase 2 is led by the internal no-code champion and powered by a small “tiger team” within the IT department. In the case of Unqork, an experienced code-based developer can get up and running after a three-week bootcamp, and have access to 24/7 access to documentation and support for crucial issues. At this point, resources from the no-code provider or SI are still a crucial part of the hybrid development team.

#### NORTH STAR

By the end of Phase 2, the team has ideally completed at least one low-to-medium complexity project. The completed project should showcase value for the company through the acceleration of the development process as compared to a traditional code-based approach. The key KPI is the amplification of engineering resources.





## PHASE THREE

### Scaling-Up

Now that internal resources have some no-code experience under their belt, organizations move on to establishing a larger no-code team that can take on an increasing number of projects of greater sophistication. The goal of this phase is to build out the capability to rapidly develop and iterate no-code-based applications of increasing complexity.

#### RESOURCES

During this phase, organizations will develop their first Center of Excellence (CoE) featuring between 5 and 15 business and IT resources depending on the complexity and scope of the project (see accompanying CoE sidebar). At this point, the provider or SI is still engaged in the partnership, but less from a direct development standpoint and more to ensure an organizations' Creators are developing the necessary skills and a strong foundation in best practices for architecture and security.

#### NORTH STAR

By the conclusion of Phase 3, the organization should have successfully implemented between 10 and 15 advanced solutions across a wide variety of use cases and business areas. At this point, all no-code-built applications should demonstrate a discernable value-add through increased operational efficiencies, amplified productivity, or an enhanced UX (measured through an improved NPS). The company should have the ability to define clear goals and KPIs for each application, which can be used to determine success or a need for further iteration.

## What does a no-code CoE model look like?

Each project has teams that vary based on complexity and scope. A team made up of the following roles-types helps to create a balanced team that is able to plan, build, test, and iterate. Example CoE:



**Solutions Architect:** This is the technical lead who is in charge of quality and build and oversees the project from a technical standpoint. They have a technical background but understand the importance of business goals.



**Engagement Manager/Product Owner:** This is the lead in charge of ensuring that the business's needs are being addressed and the resulting outcomes. This person has project management knowledge, Scrum or Agile, and has experience with cloud development.



**Configuration Resources:** These resources may go by many names (developers, engineers, programmers—we use the term “Creators”). These are the workers engaging with the no-code platform who will configure and work with the platform to create a final product.



**Quality Assurance:** This role is charged with ensuring that the product meets the needs of the business and end-user. They test all team builds and establish an iterative testing plan. They should have experience with manual and automated testing.



**Designer:** This role designs as part of the planning phase and finalizes it via usability testing.



**UX Engineer:** It may not be crucial to have this role on the payroll (at least not for non-mission-critical applications), but we have found that applications that emphasize a high-quality UX are the ones who have the most success.

There are various CoE models, and organizations will find the ones that work for them. Once this organizational core is established, companies can move in faster cycles with increasing agility.

## **PHASE FOUR**

### **Transformation**

In this final phase, the entire organization will have developed the operational and technical muscle to rapidly address evolving business challenges with a sophisticated digital response using no-code. The company should also have established a development function that will allow them to efficiently maintain and upgrade existing solutions across the organization.

#### **RESOURCES**

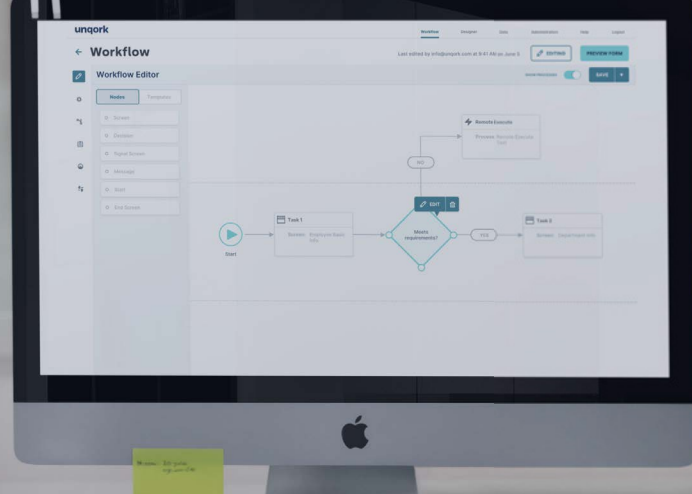
At this point, no-code will have become an integral part of the organization with a C-level sponsor overseeing it. The no-code provider or SI may provide some guidance or resources, but engagement should diminish over a course of several months.

This is also the point when companies should be able to take advantage of a new IT workforce paradigm that will allow them to apply resources more judiciously. Experienced IT professionals can be completely freed from high-volume (but not necessarily high-value) development tasks and can spend all their time building sophisticated value-additive solutions, while newer no-code literate resources—or even no-code-trained business users—can address small iterative changes and updates.

#### **NORTH STAR**

Moving forward from this phase, the IT development function should be able to keep up with the speed of business. Various teams throughout the company can freely—and rapidly—translate the needs of the business into custom digital solutions and establish a competitive advantage over those who don't. No-code-trained business users (with proper permissions) can even rapidly build functional prototypes, which can be further accelerated by existing IT resources.

# Unqork: The First Enterprise No-Code Application Platform



The leaders of tomorrow will be the companies that can digitize their processes most thoroughly and adapt their infrastructure most rapidly around shifting business challenges. With no-code, firms are empowered to build scalable, secure, complex, compliant, custom applications with unprecedented speed and flexibility.

That's why many of the most innovative players are partnering with Unqork, the first enterprise no-code development platform specifically designed for the world's most complex and regulated industries. Our platform represents an entirely new paradigm that optimizes every aspect of enterprise development through:



**A unified SaaS platform:** Unqork is a completely unified SaaS platform, which means it provides all the components and capabilities related to crucial areas like **compliance** (up-to-date regulatory and enterprise rules engines for FATCA, CRS, UK CDOT, Dodd-Frank, EMIR, and MiFID II, etc.), **security** (native encryption both in transit and rest, custom RBAC capabilities, and crowd-sourced penetration tests), and **application management** (SDLC governance, application versioning, and module management)<sup>1</sup>.



**A visual UI:** Applications are built via an intuitive, visual User Interface (UI) featuring drag-and-drop components representing user-facing elements, backend processes, data transformations, third-party integrations, and a growing library of industry-specific templates.



**Enterprise-grade standards:** While there are several business-area-specific or consumer-level no-code systems on the market, Unqork is the only no-code platform designed specifically to build complex, scalable, enterprise-ready applications, which is why it's already being used by some of the world's leading organizations.

<sup>1</sup>While Unqork is a SaaS platform, our customers operate in single-tenant environments, which means there is never a mixing of client data between Unqork customers. Unqork is cloud-agnostic, so customers can avoid cloud vendor lock-in and deploy applications in the cloud of their choice.

Unqork allows enterprises to shift all their focus to addressing business challenges instead of technical ones. The platform takes on the “heavy lifting” and frees organizations to invest their resources building operational efficiencies and perfecting the client experience. This streamlined approach helps organizations achieve:

- **Accelerated speed-to-market:** No-code automates many high-volume development tasks so new applications can be built and deployed much faster. In many cases, applications that would take months or years to reach the market can be built in a matter of weeks, or even days.
- **The elimination of legacy code:** Code becomes legacy nearly instantly. With no-code, organizations only need to be concerned with building business logic, even if there is a technical change, the platform handles all that on the backend.
- **Ease of updates and maintenance:** Large enterprises can spend up to 75% of total IT budget maintaining existing systems. One of the reasons is the complexity of making a change in one area requires changes throughout the process. A no-code platform automates many of these cascading tasks and therefore reduces the complexity of making changes.
- **Business agility:** Whether it is a pandemic or disruptions of a smaller scale, no-code can provide organizations with a way to address events quickly.

Curious about how no-code can be applied within your organization? Get in touch to [schedule a demonstration](#) from one of our no-code experts.

---

# unqork

## Enterprise application development, reimagined

Unqork is a no-code application platform that helps large enterprises build complex custom software faster, with higher quality, and lower costs than conventional approaches.

