

unqork

Platform Release Notes

Version: 7.10.0

Release Highlights

Introducing UDesigner

Starting with the 7.10 release, UDesigner will become the default user interface for Unqork users. UDesigner is Unqork's new core Integrated Development Environment (IDE) that supports all existing and new Unqork functionalities. Building off of the same user interface you're used to, UDesigner introduces a suite of enhancements designed to revolutionize your development experience including:

- **Easier Cross-Collaboration:** Enhanced collaboration and communication tools ensuring everyone—not just developers—can contribute effectively, leading to faster problem-solving and reduced development time.
- **Seamless Navigation:** Improved wayfinding capabilities that help users get information faster, and boost productivity, making development smoother and allowing for a quicker time to market.
- **Integrated Application Lifecycle Management:** Streamlined processes and tools embedded in UDesigner enable effortless collaboration, facilitate easier updates, and confident testing throughout the entire Software Development Lifecycle (SDLC).

To learn more about the new UDesigner IDE, click [here](#). To receive a free training session on UDesigner, click [here](#) and register for a session.

Introducing Vega

Creators can now also start building next-generation applications with UDesigner on the new Vega runtime. Rest assured, your existing applications will continue to function seamlessly, ensuring continuity while you explore the advanced features and benefits of Vega. This transformative runtime includes:

- **Performance and Stability:** Architected from the ground up with performance and stability as its core objectives, Vega offers new layers of scalability for both business and IT.
- **Powerful New Features:** Vega unlocks a suite of game-changing features designed to elevate user experience and enable the creation of more sophisticated applications. These features include Embedded UI, Vega Tables, and Operations Builder.
- **Extensibility and Modularity:** Build, reuse, and iterate faster with this highly modular runtime that supports the creation of complex, yet maintainable applications.

To learn more about the new Vega runtime, click [here](#). To receive a free training session on Vega, click [here](#) and register for a session.

Introducing Application Versioning

Application Versioning is an important process in software development. When multiple team members contribute to the creation of an application, versioning ensures team members are building, testing, and viewing the correct application version and not creating unintended changes for other team members. Application Versioning also lets you easily add enhancements, test functionality, troubleshoot issues, and roll back an application to a working version. When enabled, you can publish unique, uneditable versions of your application for stability, create branches to work in parallel, and merge changes across branches.

Important Considerations

1. Application Versioning for the 7.10.0 release only supports module-type applications. It does not currently support workflow-type applications.
2. Once an application is converted to a versioned application, it cannot be reverted back into an unversioned application.
3. Library modules or dependencies from unversioned applications are not supported with versioned applications. Any shared modules must be added into another versioned application and reshared if needed in your application.
4. Application versioning can be enabled on an application-by-application basis.

To learn more Application Versioning and the enhancements to expect in upcoming releases, click [here](#).

New UDesigner and Vega Documentation

Due to the large amount of Unqork documentation on the [Documentation Hub](#), article updates are ongoing, and will be available as features are released. As new Vega features are developed, you can find all Vega documentation [here](#). To support Unqork's Classic Designer and our new UDesigner IDE, updated documentation will include both versions of the IDE. Supported documentation includes new Classic Designer and UDesigner tabs that allow you to view the appropriate documentation without having to navigate to a new page.


Introduction to the Module Builder

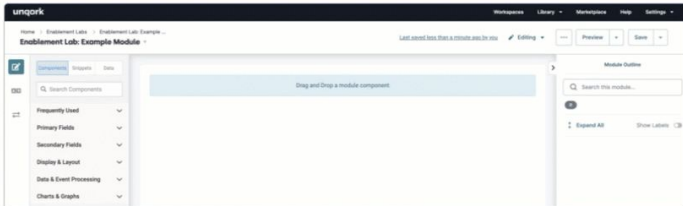
Classic Designer UDesigner

Classic Designer

Overview

The **Module Builder** is one of the core functions of the Unqork Designer Platform. With **modules** built in the Module Builder, **Creators** develop **applications** that businesses use to facilitate their products. These applications can be as simple as a questionnaire for **end-users** to fill out or complex interfaces that can manipulate databases and display dynamic information in a dashboard.

 **TIP** To learn more about what modules are, view our [What is a Module?](#) article.



(click image to expand)

Enhancements



Enhancements

Administration

- **Environment Administration:** Adjusted the minimum **Inactivity Timeout** value to **5 minutes**. Previously, Creators could configure this value to **1 minute**, which made it difficult for end-users to sign in.

APIs

- **enable-partitioned-cookies:** Google Chrome has phased out third-party cookies for 1% of users and will continue more broadly in 2024. Unqork has added the `enable-partitioned-cookies` feature flag to allow you to opt into third-party cookies before Google Chrome phases them out. If your configuration loads Unqork inside an iframe or pop-up window, this feature flag could impact the ability to load those windows.
- **Get Data Model Records:** This API can be used to return all available Data Model records. Previously, if no Data Model records existed, a value of `null` was returned. If no Data Model records exist, the call now returns a value of `0`.

Bug Fixes

Bug Fixes

APIs

- **Get Data Model Records:** Fixed an issue where the `countOnly` parameter did not return a Boolean value and resulted in an error.

Components

- **Date Input component:** Fixed an issue when an end-user entered a non-numeric value into the component in Express View, and the component accepted it without returning a validation error.
- **Email component:** Fixed an issue where after updating and saving an Email component, the updates are not reflected in Express View.



Bug Fixes (Continued)

Module Builder

- **Dynamic Refresh:** After updating an imported module and returning to the importing module, a warning displays to notify Creators that a change occurred to the imported module. This message lets you refresh the changes or view them. Fixed an issue where the refresh icon did not display.

Workflow Builder

- **Signal node:** Fixed an issue where workflows would not use the correct signal message when navigating in a loop back to a previous task.
- **Workflow Version History:** When saving a workflow version, Creators have the ability to enter a comment of changes that were made to this workflow version. Then, Creators can use the Workflow Version History to restore a workflow. Fixed an issue where saving a workflow version with a long comment disables the ellipsis button in Workflow Version History, making it impossible to restore the version.