iText DITO 2.2

Release Preview Guide

November 2021

Disclaimer - This Release Preview Guide contains highlights that are expected to be included in the next minor release of iText DITO. It is not meant to be complete, nor does it guarantee that all of the listed items will be fully implemented.

Introduction

We are excited to announce the public release of iText DITO 2.2, the latest version of our data-driven and template-based, collaborative PDF generator. We have continued to improve upon the changes iText DITO 2.1 brought this past July by further integrating the iText DITO Manager component with the Editor.

A big update for the 2.2 release of iText DITO is that it now allows users to natively deploy it to a Kubernetes environment. Additionally with the release of iText DITO 2.2 we have changed our containers to be run as a non-root user, allowing you to set users instead of running containers as root.

Contents

iText DITO Updates

Set User

With the release of iText DITO 2.2 we have changed our containers to be run as a non-root user, allowing you to set users instead of running containers as root. In case there are issues with accessing folders created by containers in previous versions, we have added in a how-to guide to help users upgrade from pre 2.2.0 versions for the Editor Docker container. You can use this link to access the guide!

Kubernetes

iText DITO 2.2 now has native support for iText DITO deployment on Kubernetes, allowing you to deploy the DITO Template Manager, Editor & SDK application to a Kubernetes environment. This means you can now easily deploy iText DITO on Kubernetes-enabled environments from cloud providers such as Azure, AWS, and Google Cloud, as well as local deployment. All deployments, services, persistent volume claims and config maps can be specified in a single deployment file to simplify its management.

Updated Licensing Mechanism

Previously iText 7 and iText DITO had separate licensing mechanisms but with the release of iText DITO 2.2, it now uses the same licensing mechanism which we introduced with the release of iText 7.2. The licenses for the new versions of iText 7 and iText DITO will now be using a JSON structure rather than the XML structure that was previously used. For additional information you can reach out to your Sales Rep. or our Sales team regarding your complementary license update.

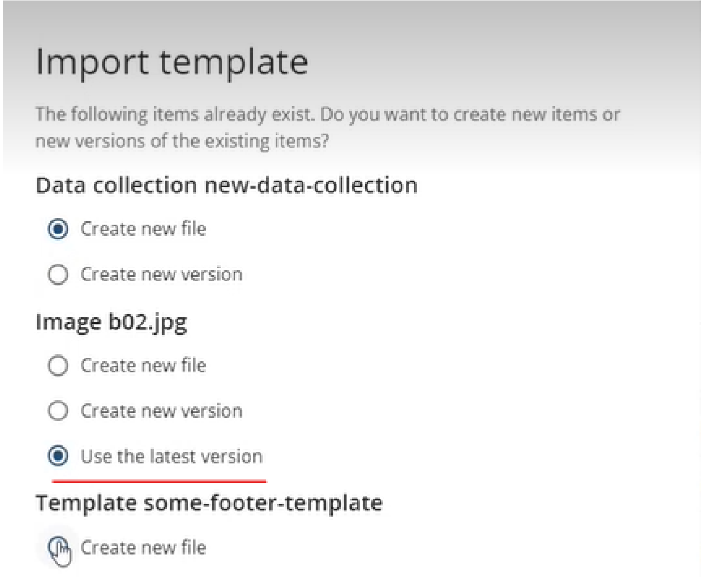
iText DITO Manager

Sub-Accounts Management

iText DITO 2.2 supports multi tenancy account management, allowing you to create workspaces that will have a child relationship to the workspace from which they were created. This allows companies with different divisions or branches to manage their own data collections, templates, and assets independently from one another.

Template Import

We have improved duplicate entity detection when importing templates. In cases where two templates were originally tied to the same data collection, images, or templates, users can use the latest versions of these files.



Import template

The following items already exist. Do you want to create new items or new versions of the existing items?

Data collection new-data-collection

● Create new file

○ Create new version

Image b02.jpg

○ Create new file

○ Create new version

● Use the latest version

Template some-footer-template

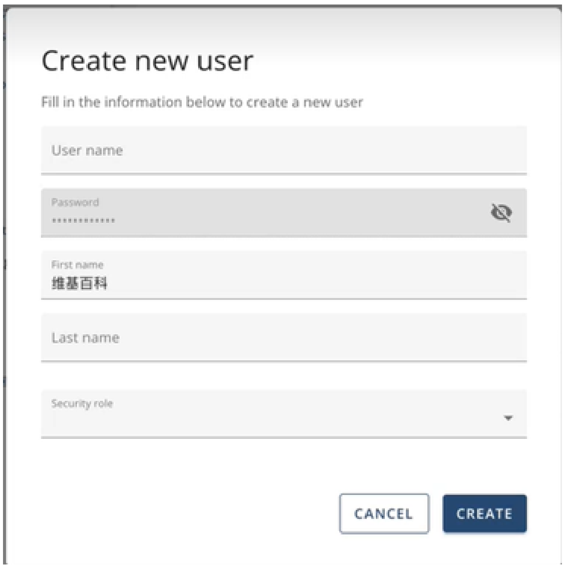
○ Create new file

Normalizations

We are also introducing a normalization report detailing repairs done on the Manager side to fix templates with incorrect structure. A normalization report containing information about the versions being replaced is shown to the user once the templates have been imported. Previously in the Manager component we did not show warnings on template import. However, we have now implemented a warning notification for when the SDK normalizes the templates and amends it during the import procedure to fix templates which are a bit broken or contain some legacy logic.

User Creation

We have improved the rules for user creation to support different languages, diacritics, dashes and double first or last name in the First and Last name fields of 'Create user/ Edit user profile and Edit user name'. We have also added warning icons in the instances table in the case they are unreachable.


Create new user

Fill in the information below to create a new user

User name

Password
••••••••••••

First name
维基百科

Last name

Security role

Customized Request Timeouts

We have made request timeouts customizable with the values being integers denoting timeout in milliseconds. We have also improved the messages on request timeouts to now show a specific message stating that the request timed out, rather than the generic error message that was previously displayed.

Legacy Template Import & Export

We have improved embedded stylesheet names handling during Legacy import by stripping all unsupported characters before using names from an embedded stylesheet.
The notification on legacy import failure has also been improved so that it includes the error message from a failed request in a failed import notification rather than a generic wrong format message. We have also removed the possibility of legacy export without dependencies on the UI level for the frontend component but not the backend for debugging purposes.

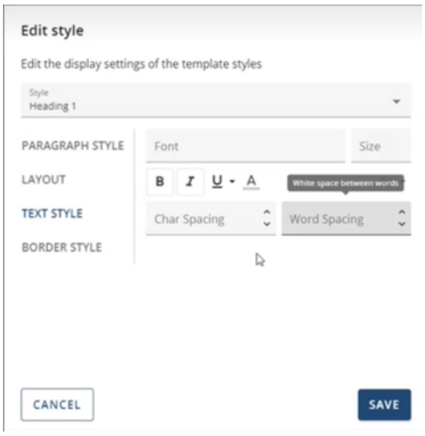
iText DITO Editor

New UI

There have also been improvements to the Editor component with the release of iText DITO 2.2. Previously the Editor used the UI from version 2.0 by default making users manually switch to the new UI. However, we have now updated it so the new UI is used by default though users can still switch to the old UI if they wish by adding “redesigned=false” to the address.

Improvements to the Editor Interface

We have improved the FormCheckbox component in frontend components. We are able to use it for the following element properties: Insert page break, Disabled, Barcode (extend code set, generate checksum), ToC (show page numbers, insert hyperlinks) and Border collapse property for tables.
There have also been efforts made to redesign controls for text styling in the Editor by integrating the font, font size and toggle buttons into the Edit Style Modal.



Edit style

Edit the display settings of the template styles

Style: Heading 1

The Anchor ID has also been simplified from the previous version that was in iText DITO 2.1. We have removed the drop-down option for the Anchor ID and now allow users to type static variables into the Anchor ID as well as choose which data to apply by clicking on the horizontal ellipsis.

Page Counter

The new Page Counter mode is compatible with PDF/UA and allows users to manually fix the template. This option allows you to add borders, text alignment and other element properties to the page counter. See example below of a page counter surrounded by borders:

For page counters that have a “width” CSS property specified, we have introduced an additional UI setting in the Editor that allows you to make the page counter static in its width. Additionally, we have implemented an “Enable performance-efficient mode” checkbox and when enabled, the field width is displayed, and the width is absolute. If the checkbox is checked, the frontend will add “width: X” and “display: inline-block” for those page counter elements, while removing those CSS properties when it’s unchecked. There will also be improved performance when there are no page counters present in a document with the use of a different page counter mechanism.

iText DITO SDK/API

Java and REST APIs

Before the release of iText DITO 2.2 we only accepted a couple of HTML tags when inserting rich text from data. Now we have also whitelisted <br/> and <ul> / <ol> and <li>.

We are also introducing a new evaluation mode for page counters which trades precision for speed in cases where accuracy isn’t needed.

There has been a “reconnect” button implemented that will try to reconnect to the instance which lost connection, as well the option to replace that broken instance with another one.

Docker Containers

With the use of a Kotlin script for starting up the Docker SDK, snakeyaml would have to be downloaded from the internet which is not possible for users that have isolated internet settings. However, we have now implemented a solution to support startups in internet isolated environments. We have also made the possibility to map log directory to external volume optional, giving users the ability to avoid specification of writable directories. This makes for an easier and more efficient logging experience with Docker.