Pipeline Admin Manual

Installation, Admin, & Design

by Team Digital

v1.0

Published by: Digital Services

Last updated: 2022-10-31

Created: 2022-10-26

Language: English (United Kingdom)

Created by: User Name

Contents

Publication Structure and Layout Design

Note: Admin rights are needed to edit the parts described below. The admin area is located at /admin/ after your installation top level domain. Ask support for access if you cannot access this area.

Parts	*Description*	*Comments*
Fidus Writer		
Document (Fidus Document)		
Template (Document Template)		
Element types (Template parts)		
Document Style (CSS Styling)		
Citation Styles	These use Citation Style Language (CSL) to configure citations.	
Export Filters	These are filters for each format export. Only DOCX and ODT are exposed to admins. Other exporters are core application parts and have to be custom-made.	
Book (Fidus Book)		
Book Style (CSS Styling)		
Plugins	These add functionality - like the Book or Open Journals System plugins.	
Output (Formats)	Website; Paginated Web; PDF; Print-on-Demand (PDF); e-Book	Note: Other file formats are available for outputting.
GitHub and GitLab		
Repo template	This is a template for the repository. It contains Docsify resources, and JSON configuration to link other formats.	
Vivliostyle		
Paginated Media CSS	Vivliostyle is run from inside Fidus Writer, as a stand-alone app, and locally.	

Fidus Writer

Fidus Writer Document

Fidus Document - These can be considered to be similar to DOCX word processing document. Fidus Documents are JSON files and can be exported and imported between versions of Fidus Writer only.

Templates (Document Templates)

Templates (Document Templates) - These are the parts of a document. Generally templates do not need to be created as the default template 'Standard Article' works for most publication.

The templates have three main functions:

- Adding custom document parts 'Element types': headers, rich text, separators, etc.
- Assigning CSS Classes to the 'Element types' for styling purposes
- Document settings: citation style, language, owner, etc

Template document parts 'Element types' are as follows. These parts have setting such as editable / not editable - so that fixed text can be added to documents.

- Heading
- Namelist (persons)
- Richtext
- Table
- Table of Contents (headers from the document)
- Separator

Creating and editing templates: Templates can be made in the Templates section or Duplicated in the Admin area (Template creation in the Admin area does not work at present Oct '22).

A list of templates can be found in the Admin area.

Important: It is usually better to create a template by duplicating a template in the admin area as then all the associated 'styles' and 'export' filters are automatically created. If you use the Template area to create a new template these will all have to be manually created and added.

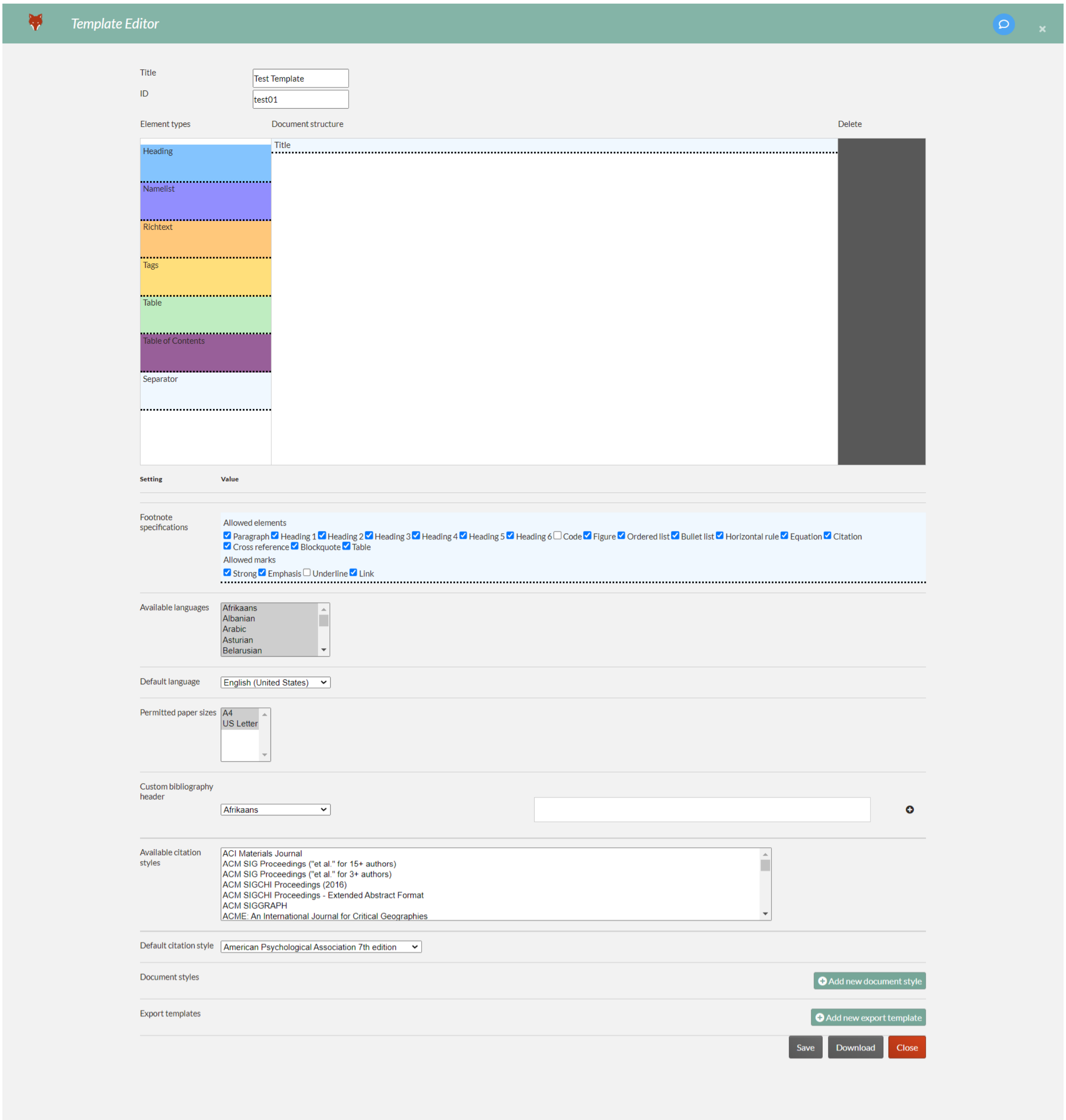
Note: Template IDs must be unique. If you find your templates not working check for mistaken duplicate IDs as this will stop templates working. Template ID should have no spaces in the name and only be letters and numbers. The same applies for all IDs.

Note when creating or editing templates - drag 'Element types' to the right dark area to delete them.

Ownership of templates and access rights, for groups as an example: If a template is assigned to a user it is private and only accessible to documents the user makes and shares. If not assigned a user it is available to all users in the Template area.

Changing a Fidus Document template after document creation: This can only be done by admins in the admin area. Note: if field are removed and no longer exist - users will be give a red notice on their document asking if they want to copy existing content as a field ''Element type' has been removed.

To change a document template - navigate to the Admin area > Documents > edit document > then at the bottom the document template can be changed.

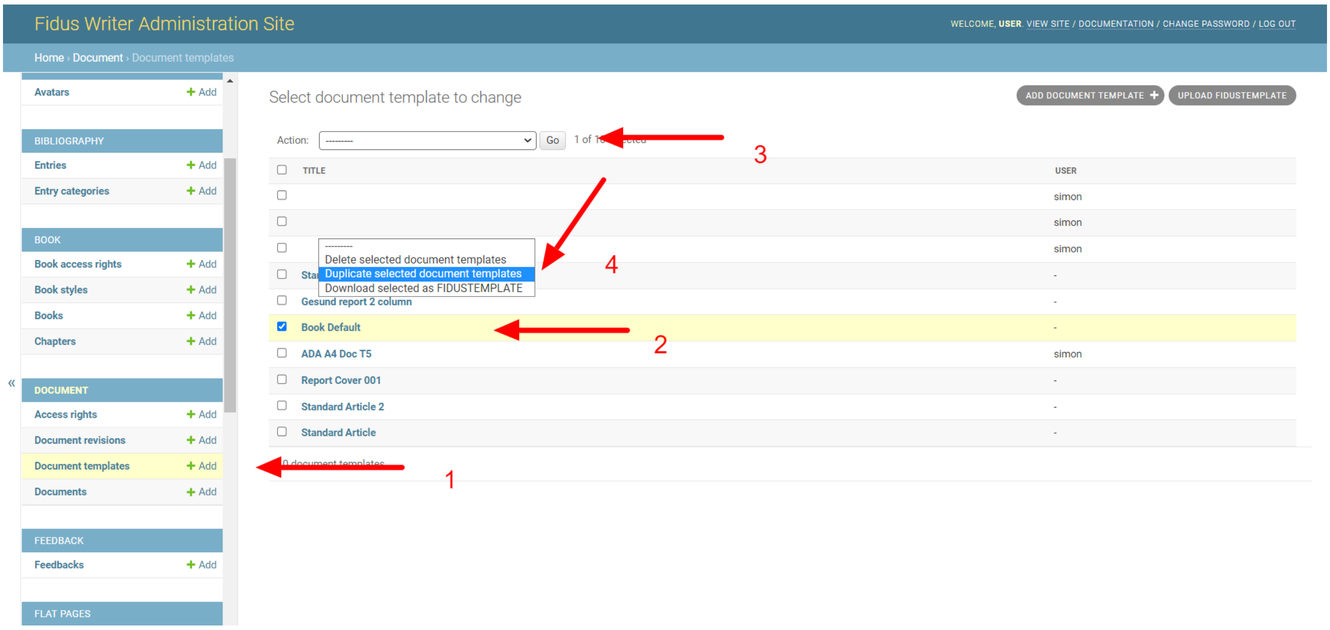


Title: Test Template

ID: test01

Setting	Value
Footnote specifications	Allowed elements ☑ Paragraph ☑ Heading 1 ☑ Heading 2 ☑ Heading 3 ☑ Heading 4 ☑ Heading 5 ☑ Heading 6 □ Code ☑ Figure ☑ Ordered list ☑ Bullet list ☑ Horizontal rule ☑ Equation ☑ Citation ☑ Cross reference ☑ Blockquote ☑ Table Allowed marks ☑ Strong ☑ Emphasis □ Underline ☑ Link
Available languages	Afrikaans Albanian Arabic Asturian Belarusian
Default language	English (United States)
Permitted paper sizes	A4 US Letter
Custom bibliography header	Afrikaans
Available citation styles	ACI Materials Journal ACM SIG Proceedings ("et al." for 15+ authors) ACM SIG Proceedings ("et al." for 3+ authors) ACM SIGCHI Proceedings (2016) ACM SIGCHI Proceedings - Extended Abstract Format ACM SIGGRAPH ACME: An International Journal for Critical Geographies
Default citation style	American Psychological Association 7th edition
Document styles	
Export templates	

Template Editor (Fidus Document Template Editor)

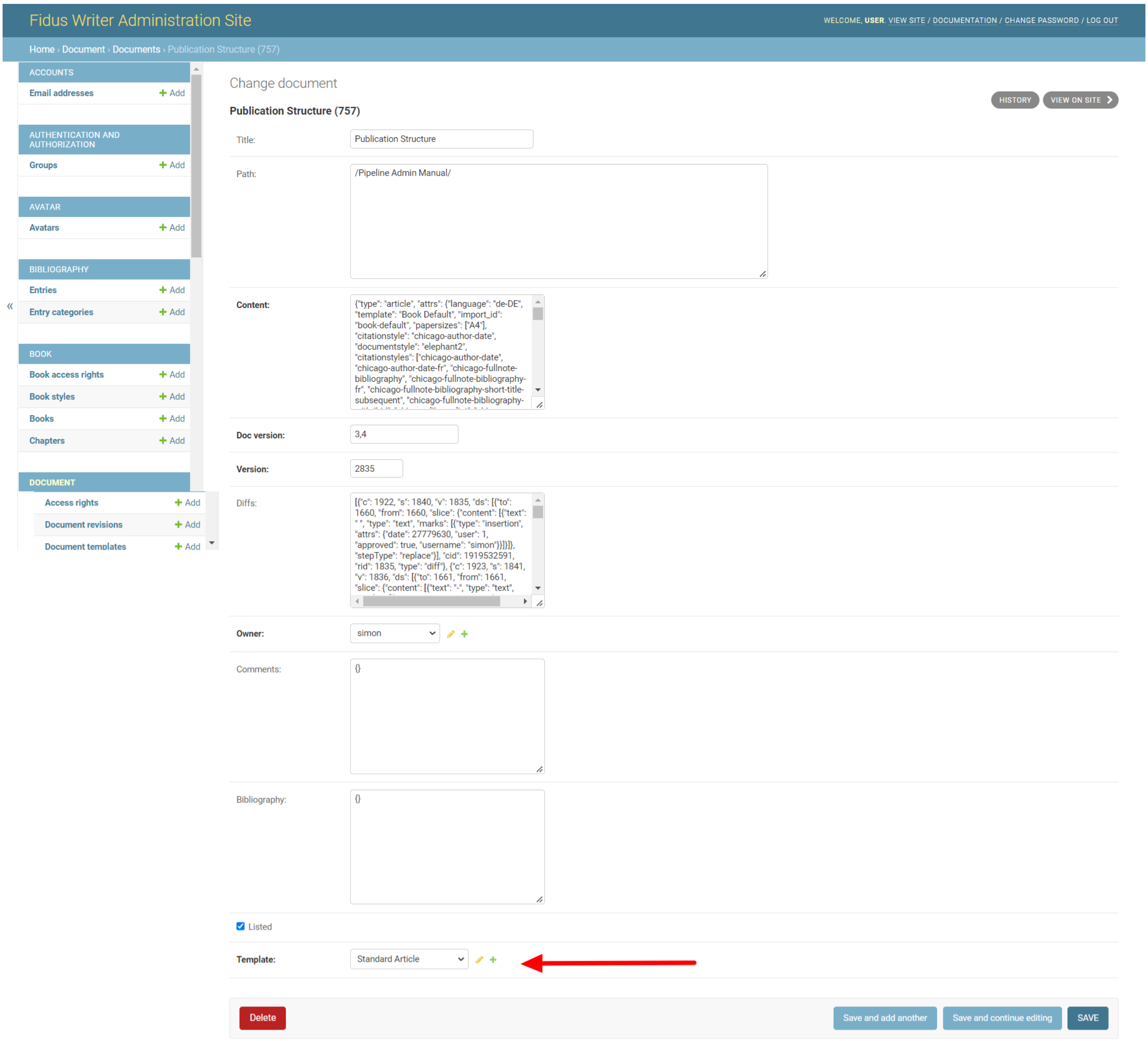


Select document template to change

☐ TITLE	USER
☐	simon
☐	simon
☐	simon
☐ Sta	-
☐ Gesund report 2 column	-
☑ Book Default	-
☐ ADA A4 Doc T5	simon
☐ Report Cover 001	-
☐ Standard Article 2	-
☐ Standard Article	-

0 document templates

Template list and steps for duplication.



Change document

Publication Structure (757)

Title: Publication Structure

Path: /Pipeline Admin Manual/

Content:

{"type": "article", "attrs": {"language": "de-DE", "template": "Book Default", "import_id": "book-default", "papersizes": ["A4"], "citationstyle": "chicago-author-date", "documentstyle": "elephant2", "citationstyles": ["chicago-author-date", "chicago-author-date-fr", "chicago-fullnote-bibliography", "chicago-fullnote-bibliography-fr", "chicago-fullnote-bibliography-short-title-subsequent", "chicago-fullnote-bibliography-

Doc version: 3,4

Version: 2835

Diffs:

[{"c": 1922, "s": 1840, "v": 1835, "ds": [{"to": 1660, "from": 1660, "slice": {"content": [{"text": " ", "type": "text", "marks": [{"type": "insertion", "attrs": {"date": 27779630, "user": 1, "approved": true, "username": "simon"}}]}]}, "stepType": "replace"}], "cid": 1919532591, "rid": 1835, "type": "diff"}, {"c": 1923, "s": 1841, "v": 1836, "ds": [{"to": 1661, "from": 1661, "slice": {"content": [{"text": "-", "type": "text",

Owner: simon

Comments: {}

Bibliography: {}

☑ Listed

Template: Standard Article

Change the document template

Styles

Styles allow for automated outputting with no manual intervention needed for as many publications as needed.

Document styles and book styles are made the same way, just stored in different places.

The main CSS Styles do not cover all output formats. Below is a list of which output formats are controlled by which style - there are three style types: CSS Styles, Docsify CSS Style, and ODT/DOCX Styles.

In the documentation we mainly cover the CSS Styling used in Fidus Writer:

- CSS Styles: Output formats - PDF, Paginated Web, e-book outputs - made with Fidus styles for document and book. Note, book styles override document styles on outputting.
- Docsify CSS Style: Output formats - Website output - this is a combination of the Fidus Book Style and Docsify style found in the GitHub Template Repo.
- ODT/DOCX Styles: Output formats - DOCX and ODT - styles need to be made using the DOCX and ODT markup languages - the existing export templates can be downloaded for reference.
- LaTeX and JATS are not covered here.

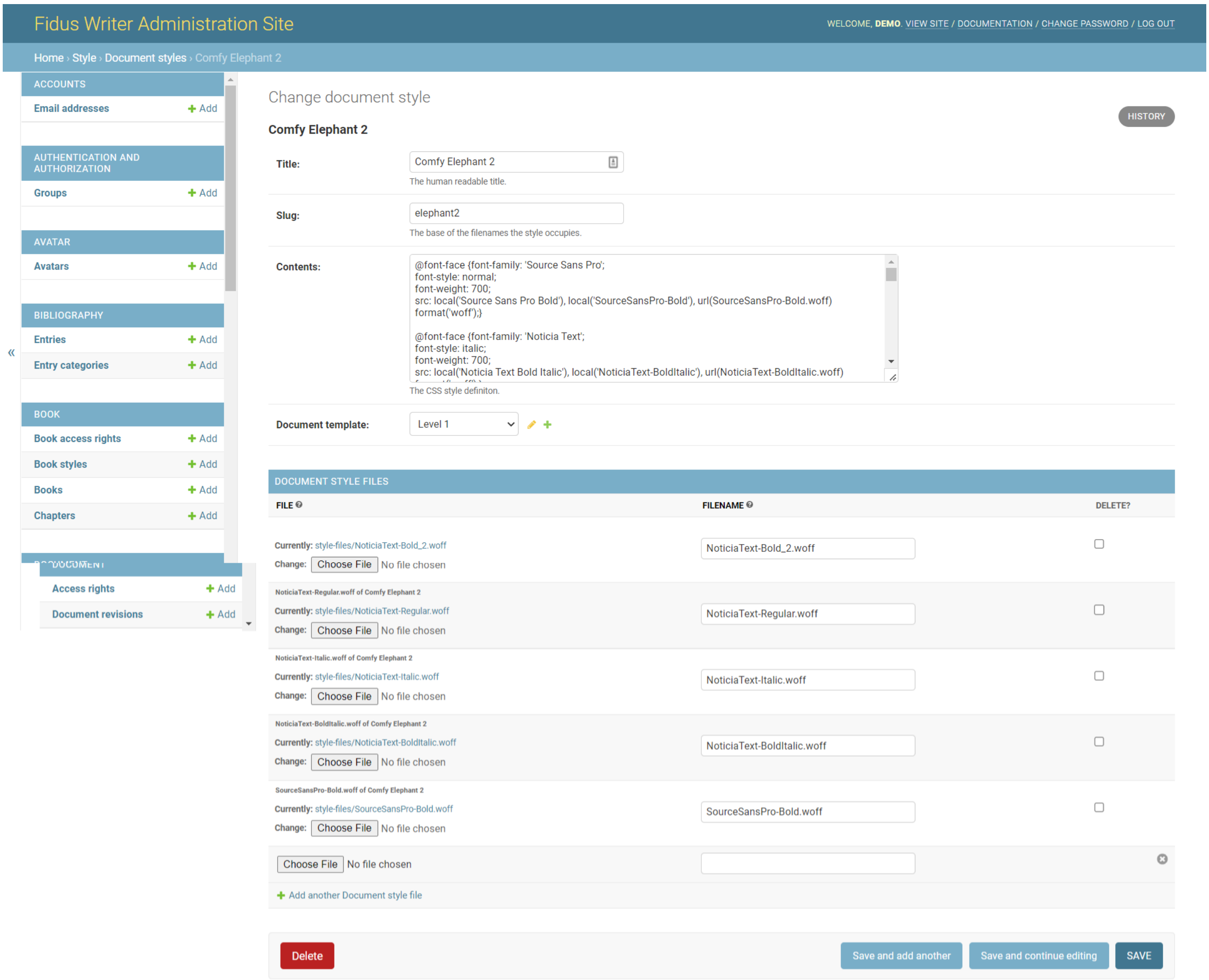
How to make a style

Styles are loaded into Fidus Writer in the /admin/ area:

- Documents - /admin/style/documentstyle/
- Books - /admin/book/bookstyle/

Styles are made of the following:

1. CSS - pasted into the style editing area
2. Fonts - these need to be uploaded and named
3. Assets - images that will be used in the style as branding or static content. Do not upload edited content images here.



Fidus Writer Administration Site

WELCOME, **DEMO**. VIEW SITE / DOCUMENTATION / CHANGE PASSWORD / LOG OUT

Home › Style › Document styles › Comfy Elephant 2

ACCOUNTS
- Email addresses + Add

AUTHENTICATION AND AUTHORIZATION
- Groups + Add

AVATAR
- Avatars + Add

BIBLIOGRAPHY
- Entries + Add
- Entry categories + Add

BOOK
- Book access rights + Add
- Book styles + Add
- Books + Add
- Chapters + Add

DOCUMENT
- Access rights + Add
- Document revisions + Add

Change document style

HISTORY

Comfy Elephant 2

Title: Comfy Elephant 2
The human readable title.

Slug: elephant2
The base of the filenames the style occupies.

Contents:

```
@font-face {font-family: 'Source Sans Pro';
font-style: normal;
font-weight: 700;
src: local('Source Sans Pro Bold'), local('SourceSansPro-Bold'), url(SourceSansPro-Bold.woff)
format('woff');}

@font-face {font-family: 'Noticia Text';
font-style: italic;
font-weight: 700;
src: local('Noticia Text Bold Italic'), local('NoticiaText-BoldItalic'), url(NoticiaText-BoldItalic.woff)
```

The CSS style definiton.

Document template: Level 1

DOCUMENT STYLE FILES

FILE	FILENAME	DELETE?
Currently: style-files/NoticiaText-Bold_2.woff **Change:** Choose File No file chosen	NoticiaText-Bold_2.woff	□
NoticiaText-Regular.woff of Comfy Elephant 2 **Currently:** style-files/NoticiaText-Regular.woff **Change:** Choose File No file chosen	NoticiaText-Regular.woff	□
NoticiaText-Italic.woff of Comfy Elephant 2 **Currently:** style-files/NoticiaText-Italic.woff **Change:** Choose File No file chosen	NoticiaText-Italic.woff	□
NoticiaText-BoldItalic.woff of Comfy Elephant 2 **Currently:** style-files/NoticiaText-BoldItalic.woff **Change:** Choose File No file chosen	NoticiaText-BoldItalic.woff	□
SourceSansPro-Bold.woff of Comfy Elephant 2 **Currently:** style-files/SourceSansPro-Bold.woff **Change:** Choose File No file chosen	SourceSansPro-Bold.woff	□
Choose File No file chosen		

+ Add another Document style file

Style editing. This is for a document, but it is the same editing process for a Book Style.

You can preview styles directly from Fidus for documents and books.

Note: Multilingual publications. CSS can contain content parts when the system cannot deliver content. These parts will usually be thing like branding or labels. Examples are page label 'Page 4/26' as opposed to 'Blatt or Seite - 4/26'. In these exceptions style copies per language will need to be made.

Margins and recto/verso issues - Layouts for targetted at documents or book styles. It is important to consider the difference between the needs of the two layouts. As an example in books there is a need to have different recto and verso left margins because of gutter being bound, also running headers, and chapter starts, as well as designated recto/verso front matter conventions.

CSS styling resources

- Vivliostyle - This is the platform used in the pipeline for PDF and Paginated Web.
- CSS Print Rocks - Community resource comparing Paged Media CSS platforms.
- Docsify styling - Docsify is a static site generator that we use to make the GitHub/Lab Pages websites.

Document styles

Document styles need to be associated to a single document template in the style editing area. If you want to use a style on another document template a copy needs to me made.

Note when you duplicate a Template it duplicates its associated style (it is more efficient to use the duplication process).

Book styles

Book style involve using folios, page numbers, sections etc. - Vivliostyle and CSS Print Rocks should be consulted for details. Also see the existing resources that have been created for reference.

Covers - Covers can be made in several ways in the pipeline as well as multi-format posing the difficulty that each formats cover can be different.

Report001 Style Covers - The cover for this style is made in the book CSS style. The e-book cover is only added later and is specific for the EPUB e-book export as technically it requires an image file.

The Report001 style has a partner style Report002 with branding included, you can switch between the two to see the difference.

The cover design in this style aims to make use of the book information fields and place them over the CSS generated artwork, this way the style can be used for a publication series with no manual intervention.

Report002 - The style contain branding assets, but apart from that is the same as Report001.

See full CSS linked here:

https://github.com/mrchristian/guide-en/blob/main/uhtml/css/report001.css#L118

- this creates the first page and the background

```
@page :first {

size: 210mm 297mm;

background-image: linear-gradient(#2062af,#20215E 80%, #fff 10%);
background-size: 210mm 297mm;

display: flex;

margin-top: 30mm;

color: #001668 ;

page-break-after: always;

@bottom-left { content: normal; }

@bottom-center { content: normal; }

@bottom-right { content: normal; }

@top-left { content: normal; }

@top-right { content: normal;}

@top-center { content: normal;}

}
```

Then the following writes the content into the first page:

https://github.com/mrchristian/guide-en/blob/main/uhtml/css/report001.css#L355

```
div.titlepage {
```

Note: Images are now added to the style assets and will have local paths /media/image.png etc.

And not external link like:

```
/*background-image:
url("https://raw.githubusercontent.com/mrchristian/quick-
start/05c5b3dd2bdbdfd785bb158752e3bd3c7b415c66/assets/logo-gfm.png");
```

The content comes from the UHTML file where the book information has been written out. Normally this would appear in page 2 if the CSS were not controlling the info.

https://github.com/mrchristian/guide-en/blob/main/uhtml/index.html#L94

Book styles - Website

This is where Docsify is used.

Customizations are made on the Git repo at this path /blob/main/assets/css/theme-custom.css

Here is an example:

https://github.com/mrchristian/guide-en/blob/main/assets/css/theme-custom.css

Docsify can be run locally for theme development.

Note: If you are making a multilingual publication for some small parts like 'labels' in the table of contents menu you might need to make a template per language as parts are hardcoded, for example 'Contents'.

GitHub and GitLab

Repo Templates should be made and used. These contain the parts that Fidus does not make and then Fidus writes its content into the template repo, merging the two parts to create your output publication.

See example template:

https://github.com/TIBHannover/ADA-Book-Template

As noted there may be a need to have language specific Repo Templates.

Viviliostyle

Vivliostyle is used for Paginate Web and PDF generation using Paginated CSS to W3C standards.

See all instructions on the Vivliostyle website.

Vivliostyle can be run locally for theme development.

Install and Admin

Note, all software used is open source and compliant with OSI open sources licences.

Installation

Fidus Writer

Fidus Writer can be installed in a variety of locations: secure cloud, public service cloud, dedicated machines, in a local office network, on a Raspberry Pi (micro-computer), or on a local personal computer.

Download and install instructions are on Fidus Writer GitHub Wiki:

- https://github.com/fiduswriter/fiduswriter/wiki

It is recommended to use the Ubuntu Snap installation process as it offers a way to get constant updates.

- Ubuntu Snap install
- Snap configuration is here

Configuring for Books

To create Books additional plugins are needed, all plugins can be found here on GitHub:

- All plugins

Plugins needed for books:

- fiduswriter-books
- fiduswriter-gitrepo-export
- fiduswriter-citation-api-import (optional)

Tips

- An SMTP account is needed for transactional emails.
- Up the file upload limit to 20 MB in case of needing to upload large Fidus Writer JSON file.

- ADMIN area: Fidus Writer has a admin area which is not visible to regular users in the interface - visit /admin/ here all admin functions can be carried out.
- Roles: Fidus Writer has three roles: Admin, Staff, Users
- Flat pages - pages for data privacy, imprint can be found here /admin/flatpages/flatpage/
- Connect to GitHub and GitLab - add details here /admin/socialaccount/socialapp/
- GDPR - Fidus Writer is GDPR-compliant. As default users have to give permission to create account and accept document invites. If you are running Fidus Writer for use in a private office where all users have agreed to GDPR and privacy conditions then these conditions can be relaxed.
- User registration - this can be set as open, or by invite only. Also, via social accounts, OAuth, can be used. With others added if special plugins are written.
- Social accounts - these can be added for authentication.
- Groups - Groups are covered in the Quick Start Guide.

GitLab Community Edition (CE)

Download and instructions are here: https://gitlab.com/rluna-gitlab/gitlab-ce

Admin

Fidus Writer

Most admin functions can be found in /admin/

Fidus writer has three types of user: users, staff, and admin. Admin user rights can be given to publication managers.

User registration has three settings: open, social logins using a third party authentication, and invite only.

Groups and ACLs for content currently (Nov 2022) happens through the ownership of files: templates, documents, and books by single users. Files can be reassigned ownership in the admin area. In 2023 full groups and sharing features will be added.

Note: Fidus loads a lot of the application over into the browser and so force browser refresh is needed to update many processes.

Documents can worked on offline and on reconnect Fidus will attempt to resolve what the last edit was and update the documents on the server.

Git - If a user is granted access to a book with write permissions, and write permissions to the Git repo they can output to the repo.

GitLab Community Edition CE

See GitLab CE instructions - https://gitlab.com/rluna-gitlab/gitlab-ce

Vivliostyles

Vivliosytle has full documentation online - https://vivliostyle.org/documents/

Docsify

Docsify has detailed instructions online - https://docsify.js.org/#/

About the Admin Manual

Admin Manual – A Publishing Pipeline

English version

Pre-release v0.1

Date: 2022

Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0).

Description

The Admin Manual is intended as a short intorduction to installing the pipeline and administrative functions, as well as covering the content structure and how to create styles in Fidus Writer.

The pipeline connects Fidus Writer an online word processor with GitLab or GitHub for versioned storage and uses Vivliosyle for CSS Typesetting. The pipeline project is based on open source work of ADA Semantic Publishing Pipeline at the Open Science Lab, TIB (Hannover).

All software used with the exception of GitHub is open source.

Contributors

Author(s): Simon Worthington - ORCID 0000-0002-8579-9717

Technical credits

Layout design style

Publication layout design style 'Report 001' is based on CSS Template from **Interpunct** – *full stack graphic design,* Interpunct.dev. GNU General Public License (GPLv3).

Fonts

All fonts are Open Licence Fonts.

Headers - Fira Sans Condensed. These fonts are licensed under the Open Font License. This project is led by Carrois, a type foundry based in Berlin. To contribute, see github.com/mozilla/Fira.

Body - Fira Sans. These fonts are licensed under the Open Font License. This project is led by Carrois, a type foundry based in Berlin. To contribute, see github.com/mozilla/Fira

Logo - Source Sans Pro. These fonts are licensed under the Open Font License. Source® Sans Pro, Adobe's first open source typeface family, was designed by Paul D. Hunt.