
vsuite Documentation

Release v0.4.0

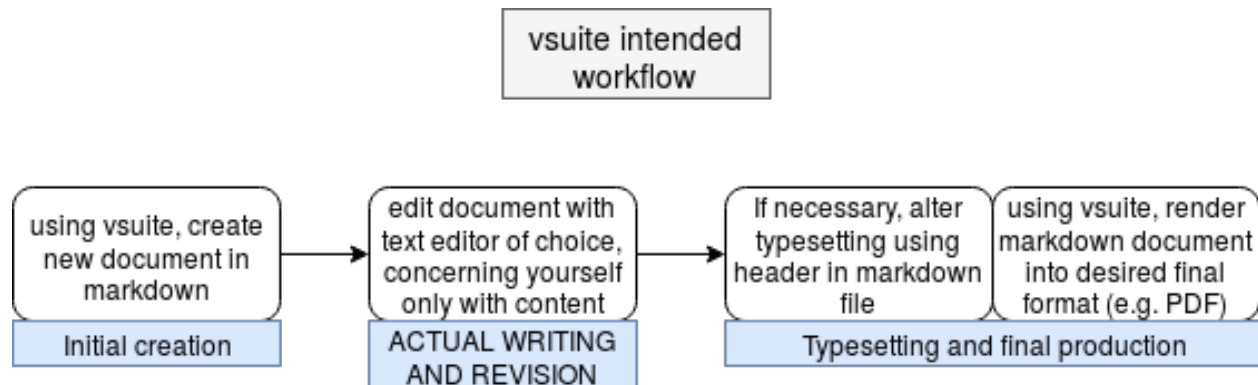
Jesse Bulson-Lewis

Feb 18, 2018

Contents:

1	Installation	3
1.1	Required software	3
1.2	vsuite	3
2	Quick Start	5
2.1	Initialize a project with <code>vs init</code>	5
2.2	Create a new markdown document with <code>vs new</code>	5
2.3	Render your document with <code>vs make</code>	6
3	Code documentation	7
3.1	User module	7
3.2	Project module	8
3.3	Asset module	9
	Python Module Index	11

vsuite is a project management suite for Linux OSes aimed at writers who want tools that help—rather than hinder—them in their writing. At its core, it is a wrapper around various technologies that makes it easy to do your writing in [markdown](#) files that can easily be turned into finished final documents. Writing in markdown allows the writer, once she is up-to-speed on its usage, to separate the conceptually-distinct tasks of writing and typesetting in a manner largely inspired by [this essay](#). Take a look at the following workflow diagram, [this demo](#), and the [quick start guide](#) to see if you might benefit from using this software.



Note: vsuite is considered to be in early alpha, and as such should not be considered reliable yet. With that said, since it is a wrapper and never puts itself in charge of deleting or overwriting any user data, the risk of using it should be very minimal. Furthermore, it should go without saying that you should always backup any data that matters.

For the time being, vsuite is only distributed through its git repositories, and so should be installed with pip after installing a few software dependencies.

1.1 Required software

- `pandoc` (for rendering markdown to other formats)
- `pandoc-citeproc` (for citations in markdown)
- `git` (for optional versioning)
- `make` (for simplifying rendering of markdown)
- `pip3` (for installing vsuite)

1.1.1 Ubuntu 16.04

```
sudo apt install pandoc pandoc-citeproc git make python3-pip
```

1.1.2 Fedora 27

```
sudo dnf install pandoc pandoc-citeproc git make python3-pip
```

1.2 vsuite

Install vsuite as a Python package using pip3 pointed at its git repository:

```
pip3 install --user git+<URL of this repo>
# For example, using the GitHub repo
pip3 install --user git+https://github.com/jessebl/vsuite
```

The program will place files in `~/.local/share/vsuite` and store its config in `~/.config/vsuite`. The requisite files should be placed when the program is run, but this has not yet been tested to any rigorous degree. Pip also creates a vsuite executable named `vs` at `~/.local/bin/vs`, but if that location is not a part of your `PATH`, you will need to [manually add it](#).

You might want to check out [this demo](#) for an example of the steps in this quick start.

2.1 Initialize a project with `vs init`

vsuite uses the concept of a project directory, in which you have various vsuite docs (which are simple pandoc mark-down files) accompanied by a hidden `.vsuite` directory that holds accompanying files like [CSL files](#) and document templates. To get started using it, you need to initialize a directory as a vsuite directory:

```
vs init
```

This effectively creates an empty bibliography file, initializes a git repository, and creates the `.vsuite` directory which includes a project config file.

2.2 Create a new markdown document with `vs new`

Finally, you can get started with actually creating markdown files using vsuite:

```
vs new <document title>
```

This will create a file `<document title>.md`, after dropping or modifying spaces or some special characters (since GNU make really struggles with these things...). This file is generated from a template by vsuite and includes a YAML header that specifies fields for pandoc. This file is the one that you are meant to edit and do your work in. Tuning your text editor for use with markdown will be greatly helpful in this, since the whole point of this writing paradigm is to leave you, the writer, with more time doing actual writing. (For example, see [this vim configuration file](#).)

2.3 Render your document with `vs make`

When you're ready to turn your markdown source into files for use by others:

```
vs make <project name>.<file extension of desired format>

# E.g generate a PDF of your document file "best_document.md"

vs make best_document.pdf
```

This uses GNU make along with a makefile in `.vsuite` to freshly generate the specified file unless it has been updated more recently than the source markdown file. Hence, you can always make sure that you have up-to-date documentation with `vs make`. The currently available formats are:

- pdf
- odt
- docx

3.1 User module

class `vsuite.user.User`

Represent a single user's vsuite installation

Track and manage vsuite's data files, global config, and more

get_fullname ()

Get user's full name from /etc/passwd

Returns user's full name

Return type str

get_global_config ()

Get user's global vsuite config

Get existing config if it exists Get and save newly-generated config if it doesn't

Returns user's global configuration

Return type configparser.ConfigParser

global_init ()

Load global config after creating it if it doesn't exist

init_global_config ()

Initialize global config

Create new config, refusing to overwrite existing one

Returns user's global configuration

Return type configparser.ConfigParser

init_global_data_dir ()

Copy project files to data dir

read_global_config()
Get existing user config

3.2 Project module

class vsuite.project.**Project** (*path=False*)
Represent a project directory

The present working directory is considered to be the vsuite project root, unless one of its parent directories is a project, in which case that directory is considered to be.

Initializing sets attributes about where various directories and project resources would be, if they exist.

check_for_project()
Verify that current directory is a project

Raises (*FileExistsError*) – If not in a project directory

copy_asset (*src_asset, dest_asset*)
Copy asset files from one asset to another

Parameters

- **src_asset** (*vsuite.asset.Asset*) – asset to be copied
- **dest_asset** (*vsuite.asset.Asset*) – asset to receive files

create_bibliography()
Create bibliography if it doesn't exist

Bibliography checked and created with *bibliography* value from project settings

create_doc (*title, template_opt=None*)
Create new document with title name from template

Parameters

- **title** (*str*) – Title of document, used as basis for filename
- **template_opt** (*str*) – Document template to override default template set in project settings

Raises (*FileExistsError*) – If file with same name already exists

create_project_dir()
Create *.vsuite* directory if it doesn't exist

get_project_dir (*cursor_dir='/home/docs/checkouts/readthedocs.org/user_builds/vsuite/checkouts/stable/docs/source', path=False*)

Absolute path to consider as project directory

Use present working directory if no parent directory is a project directory.

Parameters **cursor_dir** (*str*) – directory to check for project

Returns absolute path

Return type str

get_relpaths()
Get paths of project resources relative to *pwd*

Note: Mostly legacy method, preserved only using all assets' relative paths in `self.create_doc()`

Returns relative paths of project paths (e.g. the project's `csl_dir`)

Return type dict

get_template (*config*, *template_opt*)

Get template object to use for new document

Parameters **config** (*ConfigParser*) – config of project

Returns template to use

Return type `jinja2.environment.Template`

git_init ()

Initialize git repository in `project_path`

init ()

Initialize project directory

Reinitialize vsuite for the user, and initialize the present working directory as a project directory. This includes creating the `.vsuite` directory, creating an empty bibliography file, and initializing a git repo.

init_inherit ()

Initialize project directory

Reinitialize vsuite for the user, and initialize the present working directory as a project directory, inheriting assets and settings from the parents project

init_project_config ()

Copy user config into project

make (*output*)

Use make and pandoc to generate outputs

Use makefile from `.vsuite` directory, which leverages pandoc to generate requested outputs

Parameters **output** (*str*) – name of argument to pass to make

3.3 Asset module

In the process of transitioning to the concept of an “asset” for use within the project module.

class `vsuite.asset.Asset` (*name*, *relpath*, *file_extension*, *project_path*)

Represent a category of vsuite assets

Parameters

- **relpath** (*str*) – path to asset directory relative to `.vsuite`
- **file_extension** (*str*) – file extension of assets
- **project_path** (*str*) – path to project

abspath ()

Get absolute path to asset

Parameters **project_path** (*str*) – absolute path to project

Returns absolute path to asset

Return type str

abspaths ()

Paths of available assets

Get absolute paths of asset files in first level of asset directory

Returns file paths

Return type tuple

files ()

Available asset files

Get asset filenames in first level of asset directory

Returns asset filenames

Return type tuple

print_files ()

Print asset files, newline delimited file names

relpath_pwd ()

Get path to asset relative to current directory

Parameters **project_dir** (*str*) – absolute path to project directory

Returns relative path to asset

Return type str

V

`vsuite.asset`, 9
`vsuite.project`, 8
`vsuite.user`, 7

A

abspath() (vsuite.asset.Asset method), 9
abspaths() (vsuite.asset.Asset method), 10
Asset (class in vsuite.asset), 9

C

check_for_project() (vsuite.project.Project method), 8
copy_asset() (vsuite.project.Project method), 8
create_bibliography() (vsuite.project.Project method), 8
create_doc() (vsuite.project.Project method), 8
create_project_dir() (vsuite.project.Project method), 8

F

files() (vsuite.asset.Asset method), 10

G

get_fullname() (vsuite.user.User method), 7
get_global_config() (vsuite.user.User method), 7
get_project_dir() (vsuite.project.Project method), 8
get_relpaths() (vsuite.project.Project method), 8
get_template() (vsuite.project.Project method), 9
git_init() (vsuite.project.Project method), 9
global_init() (vsuite.user.User method), 7

I

init() (vsuite.project.Project method), 9
init_global_config() (vsuite.user.User method), 7
init_global_data_dir() (vsuite.user.User method), 7
init_inherit() (vsuite.project.Project method), 9
init_project_config() (vsuite.project.Project method), 9

M

make() (vsuite.project.Project method), 9

P

print_files() (vsuite.asset.Asset method), 10
Project (class in vsuite.project), 8

R

read_global_config() (vsuite.user.User method), 7
relpath_pwd() (vsuite.asset.Asset method), 10

U

User (class in vsuite.user), 7

V

vsuite.asset (module), 9
vsuite.project (module), 8
vsuite.user (module), 7