

---

**configit**

**unknown**

**Apr 19, 2022**



## TABLE OF CONTENTS:

<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>How does it work?</b>	<b>5</b>
<b>3</b>	<b>Usage</b>	<b>7</b>
3.1	init . . . . .	7
3.2	sync . . . . .	7
3.3	update . . . . .	7
3.4	backup . . . . .	7
3.5	include . . . . .	8
3.6	exclude . . . . .	8
3.7	other . . . . .	8
3.8	Optional Arguments . . . . .	8
3.9	Git commands supported by configit . . . . .	8
3.9.1	How to install configit . . . . .	9
3.9.1.1	Clone repository . . . . .	9
3.9.1.2	Install dependencies . . . . .	9
3.9.1.3	Make alias . . . . .	9
3.9.1.3.1	fish . . . . .	9
3.9.1.3.2	bash . . . . .	9



Confgit is a Git overhead for version control of your config files. The main difference between confgit and any other config file version system is it's simplicity. It makes version control and migration of config files safe and easy.



**INSTALLATION**





## **HOW DOES IT WORK?**

With configit you do not have to learn anything new. You only need to set up a directory where configit will copy all files you register. In result, you have all your config files centralized in one directory where you can edit and maintain your config files with git.



## **USAGE**

### **3.1 init**

Initialize a git repository for your config files in current directory and generates config file in ~/.config/configit.yml if you do not specify other location using --config argument.

```
configit init
```

```
configit init --config /alternative/location/of/config/file/configit.yml
```

### **3.2 sync**

Writes content of complementary files of registered files to their origins.

```
configit sync
```

### **3.3 update**

Writes content of origins of registered files to their's complementary files.

```
configit update
```

### **3.4 backup**

Creates a zip file with backup of all files in configit repository.

```
configit backup
```

You can specify the name of the backup file.

```
configit backup my_backup_monday.zip
```

If the name of the backup does not end with .zip it will be automatically added.

## 3.5 include

Registers a file or a directory into a confgit watch list.

```
confgit include nvim.init
```

Including directories will register all its files recursively.

```
confgit include ~/.config/
```

## 3.6 exclude

Excludes a file or directory from the registered files.

```
confgit exclude zoom.conf
```

Excluding directories will exclude all its files recursively.

```
confgit exclude .config/rofi/
```

## 3.7 other

Every other command will be called as git argument in directory with registered files.

```
confgit pull -> git pull
```

## 3.8 Optional Arguments

- `-h, --help` - show this help message and exit
- `-c $config_path, --config $config_path` - load alternative config
- `--debug` - show additional information for debugging

## 3.9 Git commands supported by confgit

clone, add, mv, restore, rm, sparse-checkout, bisect, diff, grep, log, show, status, branch, commit, merge, rebase, reset, switch, tag, fetch, pull, push

## 3.9.1 How to install configit

### 3.9.1.1 Clone repository

cd into a directory where you want to install configit.

```
git clone https://github.com/yagarea/configit/
```

### 3.9.1.2 Install dependencies

```
cd configit  
pip install -r requirements.txt
```

### 3.9.1.3 Make alias

Making an alias makes usage much easier. I highly recommend it.

#### 3.9.1.3.1 fish

```
function configit  
    python /absolute/path/to/configit/main.py $argv  
end  
  
funcsave configit
```

#### 3.9.1.3.2 bash

Add the following line to your ~/.bashrc file:

```
alias configit="python absolute/path/to/main.py"
```

Then run `source ~/.bashrc` to load the new alias.