

# Package: pkgdev (via r-universe)

November 20, 2024

**Type** Package

**Title** Helpers to Develop a Package using GitHub Actions

**Version** 0.1.0.9060

**Description** A small set of functions that takes advantage of GitHub Actions for making your life easier as a R package developer. This package is primarily intended for personal use, however feel free to use it (at your own risk :)).

**License** MIT + file LICENSE

**URL** <https://dieghernan.github.io/pkgdev/>,  
<https://github.com/dieghernan/pkgdev>

**BugReports** <https://github.com/dieghernan/pkgdev/issues>

**Depends** R (>= 3.6.0)

**Imports** cffr (>= 0.2.0), cli (>= 3.4.0), codemetar (>= 0.3.3), cpp11, devtools (>= 2.4.1), jsonld, knitr (>= 1.33), lintr (>= 2.0.1), rcmdcheck, resmush, roxygen2 (>= 7.1.0), roxygen2md, stringr, styler (>= 1.5.1), urlchecker (>= 1.0.0), usethis (>= 2.2.2), withr

**Suggests** covr (>= 3.5.1), gitdevr, goodpractice (>= 1.0.2), magick, pkgdown, rcmdcheck (>= 1.3.3), remotes, rmarkdown (>= 2.9), sessioninfo, tidyverse, V8 (>= 4.0.0)

**VignetteBuilder** knitr

**Remotes** dieghernan/gitdevr

**Config/Needs/website** dieghernan/gitdevr, cranlogs, tidyverse, jsonlite, lubridate, reactable, Polychrome, pkgsearch, htmltools

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Config/pak/sysreqs** libfontconfig1-dev libfreetype6-dev libfribidi-dev  
 git make libharfbuzz-dev libgit2-dev libicu-dev libjpeg-dev  
 libpng-dev libtiff-dev libxml2-dev libssl-dev libnode-dev  
 libx11-dev zlib1g-dev

**Repository** https://dieghernan.r-universe.dev

**RemoteUrl** https://github.com/dieghernan/pkgdev

**RemoteRef** HEAD

**RemoteSha** 0d72709db06f04d504cba5828f6b8a0a13316ae4

## Contents

add_global_gitignore . . . . .	2
check_rd_titles . . . . .	3
gha_check_full . . . . .	3
gha_pkgdown_branch . . . . .	4
gha_update_docs . . . . .	5
precompute_vignette . . . . .	6
update_docs . . . . .	8

<b>Index</b>	<b>10</b>
--------------	-----------

---

add_global_gitignore	<i>Add .gitignore to package</i>
----------------------	----------------------------------

---

### Description

Adds a global .gitignore file to the package. File based on default .gitignore provided by GitHub.

### Usage

```
add_global_gitignore(pkg = ".")
```

### Arguments

pkg Path to a (subdirectory of an) R package.

### Value

Invisible, and writes a global .gitignore file

### See Also

[usethis::use\\_git\\_ignore\(\)](#), [usethis::git\\_vaccinate\(\)](#)

**Examples**

```
## Not run:  
add_global_gitignore()  
  
## End(Not run)
```

---

check_rd_titles	<i>Check the titles of a Rd file</i>
-----------------	--------------------------------------

---

**Description**

Check if title of Rd files ends in a period.

**Usage**

```
check_rd_titles(pkg = ".")
```

**Arguments**

pkg                    Path to a (subdirectory of an) R package.

**Value**

A data frame with the results.

**Examples**

```
## Not run:  
  
check_rd_titles()  
  
## End(Not run)
```

---

gha_check_full	<i>Create a GitHub action that checks regularly your package</i>
----------------	--

---

**Description**

The GitHub action created would run R CMD check on your package. It uses a wide range of platforms, can be reduced by commenting or deleting platforms on the matrix config.

**Usage**

```
gha_check_full(pkg = ".", overwrite = TRUE, cron_expr = "30 08 1 * *")
```

## Arguments

pkg	Path to a (subdirectory of an) R package.
overwrite	Overwrite the action if it was already present.
cron_expr	Valid cron expression. By default, the first day of the month at 08:30 AM. See <b>Details</b> .

## Details

Use [crontab.guru](https://crontab.guru) to check and create your own cron tag.

## Value

A GitHub Action on <pkg>/ .github/workflows.

## Source

[r-lib/actions](#)

## See Also

[usethis::use\\_github\\_action\\_check\\_full\(\)](#)

## Examples

```
## Not run:  
gha_check_full(cron_expr = "57 16 12 * *")  
  
## End(Not run)
```

---

gha\_pkgdown\_branch      *Create a GitHub action that builds a pkgdown site*

---

## Description

The GitHub action created would deploy a pkgdown site of your package on the gh-pages branch.

## Usage

```
gha_pkgdown_branch(  
  pkg = ".",  
  overwrite = TRUE,  
  platform = "macOS",  
  version = "latest"  
)
```

## Arguments

pkg	Path to a (subdirectory of an) R package.
overwrite	Overwrite the action if it was already present.
platform	Platform to use for deploying the package. See <b>Details</b>
version	Version of the platform. See <b>Details</b> .

## Details

Check <https://github.com/actions/runner-images> to see the available options.

## Value

A GitHub Action on <pkg>/.github/workflows.

## Source

[r-lib/actions](#)

## See Also

[gha\\_check\\_full\(\)](#)

## Examples

```
## Not run:  
# With Ubuntu 20.04  
gha_pkgdown_branch(platform = "ubuntu", version = "20.04")  
  
## End(Not run)
```

---

gha\_update\_docs

*Create a GitHub action that documents and checks your package*

---

## Description

The GitHub action created would document your package (see [update\\_docs\(\)](#)), would check it and would deploy the package on a gh-pages branch.

## Usage

```
gha_update_docs(  
  pkg = ".",  
  overwrite = TRUE,  
  platform = "macOS",  
  version = "latest"  
)
```

### Arguments

pkg	Path to a (subdirectory of an) R package.
overwrite	Overwrite the action if it was already present.
platform	Platform to use for deploying the package. See <b>Details</b>
version	Version of the platform. See <b>Details</b> .

### Details

Check <https://github.com/actions/runner-images> to see the available options.

### Value

A GitHub Action on `.github/workflows`.

### Source

[r-lib/actions](https://github.com/r-lib/actions)

### See Also

[update\\_docs\(\)](#), [gha\\_pkgdown\\_branch\(\)](#)

### Examples

```
## Not run:  
# With Ubuntu 20.04  
gha_update_docs(platform = "ubuntu", version = "20.04")  
  
## End(Not run)
```

---

precompute\_vignette *Precompute vignettes*

---

### Description

Precompute vignettes from CRAN, based on <https://ropensci.org/blog/2019/12/08/precompute-vignettes/>.

### Usage

```
precompute_vignette(  
  source,  
  pkg = ".",  
  figure_ext = ".png",  
  create_r_file = FALSE  
)  
  
precompute_vignette_all(dir = "vignettes", pkg = ".", ...)
```

## Arguments

source	Name without path of the .Rmd.orig file (e.g. "some_name.Rmd.orig").
pkg	Path to a (subdirectory of an) R package.
figure_ext	Extension of the figures plotted on the vignette. See <b>Details</b> .
create_r_file	Whether to create an additional R file with the code of the vignette.
dir	Path to directory where the "Rmd.orig" files are stored.
...	Parameters passed to <code>precompute_vignette()</code> .

## Details

The function would search for the desired precomputed vignette on the `"/vignettes/"` folder and for plots on the root `"/"` folder.

### Important:

On your .Rmd.orig file make sure you have set at least the following lines if you are producing plots:

```
knitr::opts_chunk$set(  
  ... ,  
  fig.path = "./",  
  ... ,  
)
```

## Value

A precomputed vignette

## Source

<https://ropensci.org/blog/2019/12/08/precompute-vignettes/>

## Examples

```
## Not run:  
  
precompute_vignette(source = "precompute.Rmd.orig")  
  
## End(Not run)
```

---

 update\_docs

 Document your package
 

---

## Description

Run routine checks on the package:

- Clean DESCRIPTION with `usethis::use_tidy_description()`
- Compress the data of the paths `"/data"` and `"/R"` using `tools::resaveRdaFiles()`.
- Style code with `styler::style_pkg()`
- Check urls with `urlchecker::url_check()`
- Roxygenise with `roxygen2::roxygenise()`
- Precompute vignettes if present (see <https://ropensci.org/blog/2019/12/08/precompute-vignettes/>)
- Rebuild README.Rmd (if present) with `devtools::build_readme()`
- Optimize images with `resmush::resmush_dir()`
- Write codemeta.json with `codemeta::write_codemeta()`
- Write CITATION.cff with `cffr::cff_write()`

## Usage

```
update_docs(
  pkg = ".",
  url_update = TRUE,
  create_codemeta = TRUE,
  create_cff = TRUE,
  build_readme = TRUE,
  verbose = TRUE,
  precompute = TRUE,
  opt_imgs = TRUE,
  opt_dir = c("pkgdown/favicon", "man/figures", "vignettes"),
  opt_ext = "png$|jpg$",
  opt_overwrite = TRUE,
  ...
)
```

## Arguments

<code>pkg</code>	Path to a (subdirectory of an) R package.
<code>url_update</code>	Logical, do you want to update urls with <code>urlchecker::url_update()</code> ?
<code>create_codemeta</code>	Logical, do you want to create a codemeta file with <code>codemeta::write_codemeta()</code> ?
<code>create_cff</code>	Logical, do you want to create a CITATION.cff file with <code>cffr::cff_write()</code> ?
<code>build_readme</code>	Logical, build README.Rmd with <code>devtools::build_readme()</code>



verbose	Display informative messages on the console
precompute	Logical, detect and precompute vignettes? See also <a href="#">precompute_vignette()</a> .
opt_imgs	Logical. Optimize images with <a href="#">resmush::resmush_dir()</a> ?
opt_dir, opt_ext, opt_overwrite	See dir, ext and overwrite in <a href="#">resmush::resmush_dir()</a> .
...	Additional arguments to functions

### Details

This function tries to update and clean the package following a mix of best practices (e.g. checking the urls, roxygenise and rebuilding the README) and some other discretionary practices I like to have in a package, as the tidyverse approach for the DESCRIPTION file and overall code and the use of codemeta.json.

### Value

invisible, or some messages on verbose = TRUE.

### See Also

[usethis::use\\_tidy\\_description\(\)](#), [styler::style\\_pkg\(\)](#), [urlchecker::url\\_check\(\)](#), [roxygen2::roxygenise\(\)](#), [devtools::build\\_readme\(\)](#), [codemeta::write\\_codemeta\(\)](#), [tools::resaveRdaFiles\(\)](#)

### Examples

```
## Not run:  
  
update_docs()  
  
## End(Not run)
```

# Index

`add_global_gitignore`, 2

`cffr::cff_write()`, 8

`check_rd_titles`, 3

`codemeta::write_codemeta()`, 8, 9

`devtools::build_readme()`, 8, 9

`gha_check_full`, 3

`gha_check_full()`, 5

`gha_pkgdown_branch`, 4

`gha_pkgdown_branch()`, 6

`gha_update_docs`, 5

`precompute_vignette`, 6

`precompute_vignette()`, 7, 9

`precompute_vignette_all`  
(`precompute_vignette`), 6

`resmush::resmush_dir()`, 8, 9

`roxygen2::roxygenise()`, 8, 9

`styler::style_pkg()`, 8, 9

`tools::resaveRdaFiles()`, 8, 9

`update_docs`, 8

`update_docs()`, 5, 6

`urlchecker::url_check()`, 8, 9

`urlchecker::url_update()`, 8

`usethis::git_vaccinate()`, 2

`usethis::use_git_ignore()`, 2

`usethis::use_github_action_check_full()`,  
4

`usethis::use_tidy_description()`, 8, 9