

Package: covr2gh (via r-universe)

May 18, 2026

Title Test Coverage Summary on 'GitHub'

Version 0.0.0.9039

Description Calculate test coverage data with 'covr', extract insights, and post them as comments on 'GitHub' pull requests.

License MIT + file LICENSE

URL <https://github.com/dragosmg/covr2gh>,
<https://dragosmg.github.io/covr2gh/>

BugReports <https://github.com/dragosmg/covr2gh/issues>

Depends R (>= 4.2.0)

Imports cli, covr, dplyr, fs, gh, glue, knitr, purrr, rlang (>= 1.2.0), rprojroot, stringr, tibble, tidyr, usethis

Suggests quarto, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr, quarto

Config/Needs/build moodymudskipper/devtag

Config/Needs/website rmarkdown, quarto

Config/roxygen2/version 8.0.0

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE, roclets = c("`collate", "`rd",
`namespace", `devtag::dev_roclet`))

Config/pak/sysreqs cmake git make libgit2-dev libicu-dev libuv1-dev libssl-dev libx11-dev

Repository <https://dragosmg.r-universe.dev>

Date/Publication 2026-05-18 20:22:40 UTC

RemoteUrl <https://github.com/dragosmg/covr2gh>

RemoteRef HEAD

RemoteSha 6b0d74189665fdb75cbb34f435da8f5a6144ef3d

Contents

compose_comment	2
generate_badge	3
post_comment	4
use_covr2gh_action	5
use_covr2gh_badge	6
Index	7

compose_comment	<i>Compose a coverage comment</i>
-----------------	-----------------------------------

Description

Compose a coverage comment

Usage

```
compose_comment(
  head_coverage,
  base_coverage,
  repo,
  pr_number,
  diff_cov_target = NULL
)
```

Arguments

`head_coverage` (coverage) active / current branch (HEAD) coverage. The output of `covr::package_coverage()` on the head branch.

`base_coverage` (coverage) base / target branch coverage (coverage for the branch merging into). The output of `covr::package_coverage()` on the base branch.

`repo` (character) the repository name in the GitHub format ("OWNER/REPO").

`pr_number` (integer) the PR number

`diff_cov_target` (numeric) minimum accepted diff coverage. Defaults to NULL which is then interpreted as overall base branch coverage.

Value

a character scalar with the content of the GitHub comment

Examples

```
## Not run:
coverage_head <- covr::package_coverage()
system("git checkout main")
coverage_main <- covr::package_coverage()

compose_comment(
  head_coverage = coverage_head,
  base_coverage = coverage_main,
  repo = "<owner>/<repo>",
  pr_number = 3
)

## End(Not run)
```

generate_badge	<i>Generate a coverage badge</i>
----------------	----------------------------------

Description

Generate a coverage badge

Usage

```
generate_badge(value)
```

Arguments

value (a numeric scalar) percentage coverage.

Value

a SVG string invisibly

Examples

```
generate_badge(5)
generate_badge(48)
```

post_comment	<i>Post comment</i>
--------------	---------------------

Description

post_comment() first checks if a "known" covr2gh comment exists on the target pull request. If it does, then it updates it, if it doesn't, then a new comment is posted.

Usage

```
post_comment(body, repo, pr_number, update = TRUE)
```

Arguments

body	(character scalar) the content of the body of the message.
repo	(character) the repository name in the GitHub format ("OWNER/REPO").
pr_number	(integer) the PR number
update	(logical) update an existing comment or post a new one. Defaults to TRUE.

Details

Users can also choose to always post a new comment (this always deletes the previous one).

Value

a gh_response object containing the API response

Examples

```
## Not run:  
post_comment(  
  "this is amazing",  
  repo = "<owner>/<repo>",  
  pr_number = 3  
)  
  
## End(Not run)
```

use_covr2gh_action *Set up a covr2gh GitHub Action workflow*

Description

Sets up a **GitHub Actions** workflow that calculates and reports test coverage:

- uses `covr::package_coverage()`.
- on a *pull request*:
 - calculates test coverage for head.
 - builds a badge (for use in the commit message).
 - initialises or switches to the `covr2gh-storage` branch.
 - switches to base branch and calculates test coverage.
 - compares coverage for head and base and posts a comment with the main findings.
 - uploads base coverage and badge as artefacts.
 - posts a workflow **summary**.
- on a *push to main*:
 - calculates test coverage for head and creates a badge.
 - switches to or creates the `covr2gh-storage` branch.
 - commits badge (referenced in the README).

Usage

```
use_covr2gh_action(badge = TRUE)
```

Arguments

`badge` (logical) should a badge be added to README? Defaults to TRUE.

Details

`use_covr2gh_action()` wraps `usethis::use_github_action()`.

Examples

```
## Not run:  
use_covr2gh_action()  
  
## End(Not run)
```

use_covr2gh_badge *Add a coverage README badge*

Description

Inserts the markdown text for a coverage badge in the usethis block (between `<!-- badges: start -->` and `<!-- badges: end -->`). The src URL for the badge SVG points to a storage branch in the repo(covr2gh-storage). The link out points to the actions workflow page.

Usage

```
use_covr2gh_badge()
```

Examples

```
## Not run:  
use_covr2gh_badge()  
  
## End(Not run)
```

Index

`compose_comment`, [2](#)
`covr::package_coverage()`, [2](#), [5](#)
`generate_badge`, [3](#)
`post_comment`, [4](#)
`use_covr2gh_action`, [5](#)
`use_covr2gh_badge`, [6](#)
`usethis::use_github_action()`, [5](#)