

Package ‘badgcreatr’

January 7, 2019

Title Create Badges for 'Travis', 'Repostatus' 'Codecov.io' Etc in Github Readme

Version 0.2.0

Description Tired of copy and pasting almost identical markdown for badges in every new R-package that you create, on Github or other code-sharing sites? This package allows you to easily paste badges. If you want to, it will also search your DESCRIPTION file and extract the package name, license, R-version, and current projectversion and transform that into badges. It will also search for a ``.travis.yml`` file and create a ```Travis``` badge, if you use "Codecov.io" to check your code coverage after a "Travis" build this package will also build a "Codecov.io"-badge. All the badges can be placed individually or can be placed below the top "YAML" content of your ```RMarkdown file``` (Readme.Rmd) or ```README.md``` file. Currently creates badges for Projectstatus (```Repostatus.org```), license Travis Build Status, Codecov, Minimal R version, CRAN status, CRAN downloads, Github stars and forks, Package rank, rdocumentation, current version of your package and last change of ```README.Rmd```.

Depends R (>= 3.2.4)

License GPL-3

Encoding UTF-8

LazyData true

URL <https://github.com/RMHogervorst/badgcreatr>,
<https://rmhogervorst.nl/badgcreatr>

BugReports <https://github.com/RMHogervorst/badgcreatr/issues>

Suggests testthat, rmarkdown, knitr

RoxygenNote 6.1.1

Imports git2r

VignetteBuilder knitr

NeedsCompilation no

Author Roel M. Hogervorst [cre, aut]

Maintainer Roel M. Hogervorst <hogervorst.rm@gmail.com>

Repository CRAN

Date/Publication 2019-01-07 19:10:15 UTC

R topics documented:

badgcreatr	2
badgepaste	3
badgeplacer	3
badge_codecov	4
badge_cran	5
badge_cran_ago	5
badge_cran_date	6
badge_cran_downloads	7
badge_cran_version_ago	7
badge_cran_version_release	8
badge_github_fork	9
badge_github_star	9
badge_last_change	10
badge_last_change_static	11
badge_license	11
badge_lifecycle	12
badge_minimal_r_version	13
badge_packageversion	13
badge_projectstatus	14
badge_rank	15
badge_rdocumentation	16
badge_thanks_md	17
badge_travis	17
dynamic_badges_minimal	18
findbadges	19
github_credentials_helper	19
licbadgebuilder	20
minimal_badges	20
search_git	21
Index	22

badgcreatr

badgcreatr

Description

Create badges in your readme file on top of a github repository. This package creates badges for a wide range of functions.

Details

You can place a fixed set of functions in your readme, add badges only or add interactive versions with embedded R-code that changes when you re-knit your badges.

badgepaste	<i>return markdown with image and link.</i>
------------	---

Description

for internal use. used within licensebuilder

Usage

```
badgepaste(imagelink, referlink, name = "license")
```

Arguments

imagelink	link to image file
referlink	link to where to send to on click
name	what to call the button

Value

markdown

badgeplacer	<i>Place badges inside a readme.Rmd file</i>
-------------	--

Description

This is the main function to add badges to your readme.

Usage

```
badgeplacer(location = ".", status = "active", githubaccount = NULL,
  githubrepo = NULL, branch = NULL, name = "README.Rmd")
```

Arguments

location	folder where readme.rmd resides
status	one of: "concept", "wip", "suspended", "abandoned", "active", "inactive", or "unsupported"
githubaccount	your githubname
githubrepo	your githubrepositoryname
branch	master, develop, etc.
name	which file to place badges in, defaults to README.Rmd

Value

readme file with added badges

badge_codecov	<i>CodeCoverage</i> ' ' Adds a code cov badge
---------------	---

Description

Adds codecov badge

Usage

```
badge_codecov(ghaccount = NULL, ghrepo = NULL, branch = NULL,
  location = ".")
```

Arguments

ghaccount	Github account name
ghrepo	Github repository name
branch	branch name such as master, develop etc
location	defaults to current location

Value

markdown

See Also

Other badges: [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_codecov(ghaccount = "johntest", ghrepo = "yourreponame", branch = "master")
```

badge_cran	<i>Add a badge for CRAN</i>
------------	-----------------------------

Description

Shows the version number of the package on CRAN, or “not published” if the package is not published on CRAN. See <https://r-pkg.org/services#badges> One of the metacran badges.

Usage

```
badge_cran(packagename)
```

Arguments

packagename the name of your package

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_cran("dplyr")
```

badge_cran_ago	<i>CRAN time ago released</i>
----------------	-------------------------------

Description

CRAN time ago released

Usage

```
badge_cran_ago(packagename)
```

Arguments

packagename the name of your package

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_cran_ago("dplyr")
```

badge_cran_date	<i>CRAN release date of current version.</i>
-----------------	--

Description

CRAN release date of current version.

Usage

```
badge_cran_date(packagename)
```

Arguments

packagename the name of your package

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_cran_date("dplyr")
```

badge_cran_downloads *Add a badge for downloads from CRAN*

Description

Display the number of downloads from CRAN. Use last-week, last-day, or grand-total, defaults to monthly.

Usage

```
badge_cran_downloads(packagename, period = NULL)
```

Arguments

packagename	name of the package
period	defaults to month, other options are last-week, last-day, grand-total

Value

markdown

Examples

```
badge_cran_downloads("dplyr", period = "last-week")
```

badge_cran_version_ago
CRAN version and release in time

Description

See <https://r-pkg.org/services#badges> One of the metacran badges.

Usage

```
badge_cran_version_ago(packagename)
```

Arguments

packagename	the name of your package
-------------	--------------------------

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_cran_version_ago("dplyr")
```

```
badge_cran_version_release  
      CRAN version and date of release
```

Description

CRAN version and date of release

Usage

```
badge_cran_version_release(packagename)
```

Arguments

packagename the name of your package

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_cran_version_release("dplyr")
```

badge_github_fork *Add a Github fork badge*

Description

Add a Github fork badge

Usage

```
badge_github_fork(ghaccount = NULL, ghrepo = NULL, location = ".")
```

Arguments

ghaccount	Github account name
ghrepo	Github repository name
location	defaults to current location

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

badge_github_star *Add a Github star badge*

Description

Add a Github star badge

Usage

```
badge_github_star(ghaccount = NULL, ghrepo = NULL, branch = NULL,  
location = ".")
```

Arguments

ghaccount	Github account name
ghrepo	Github repository name
branch	branch name such as master, develop etc
location	defaults to current location

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

badge_last_change	<i>Creates last-change badge</i>
-------------------	----------------------------------

Description

Will add a badge containing the current date that changes on every reknitting. This is a simple pasting of r-code.

Usage

```
badge_last_change(location = ".")
```

Arguments

location defaults to working directory

Value

Rmarkdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

badge_last_change_static
Creates last-change badge static

Description

Will add current day to the repo. This function will not change when you reknit the readme. If you want that you will have to use the linklast_change_badge function.

Usage

```
badge_last_change_static(date = NULL)
```

Arguments

date if NULL/empty current date. otherwise use yyyy-mm-dd format

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

badge_license *Create a licensebadge*

Description

when 'license= NULL' this function will look in your DESCRIPTION file and search for the "license:" part. If it matches GPL or MIT a custom badge will be created. If it does not match, a general badge will be created with the name of the license in grey.

Usage

```
badge_license(license = NULL, location = ".")
```

Arguments

license License, for example GPL-3, MIT, etc. Alternatively, leaving it empty will search.
location defaults to current directory

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

badge_lifecycle	<i>Add a lifecycle badge</i>
-----------------	------------------------------

Description

Lifecycle denotations were introduced by the tidyverse team in 2018. Because project status was not a perfect fit. I think it is a real professional attitude, the badges communicate to your users what they can expect.

Usage

```
badge_lifecycle(lifecycle = "experimental")
```

Arguments

lifecycle one of the lifecycle issues

What do the different values mean?

Maturing - The API of a maturing package has been roughed out, but finer details likely to change. Once released to CRAN, we will strive to maintain backward compatibility, but the package needs wider usage in order to get more feedback and find the optimal API. **Stable** - In a stable package, we are largely happy with the API, and major changes are unlikely. This means that the API will generally evolve by adding new functions and new arguments; we will avoid removing arguments or changing the meaning of existing arguments. **Retired** - A retired package is no longer under active development, and a known better alternative is available. We will only make the necessary changes to ensure that retired packages remain on CRAN. No new features will be added, and only the most critical of bugs will be fixed. **Archived** - The development of an archived package is complete, and it has been archived on CRAN and on GitHub. **Dormant** - A dormant package is not completed, but is not currently under active development. We plan to return to it in the future. **Questioning** - We are no longer convinced that a questioning package is the optimal approach, but we don't yet know what a better approach is.

Source

www.tidyverse.org/lifecycle

badge_minimal_r_version
Display the minimal R version

Description

Display the minimal R version

Usage

```
badge_minimal_r_version(chunk = TRUE)
```

Arguments

chunk places a r chunk in the readme

Value

markdown
Rmarkdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_minimal_r_version()
```

badge_packageversion *Place a badge with the version of your package.*

Description

Place a badge with the version of your package which is automatically read from your description file.

Usage

```
badge_packageversion(chunk = TRUE)
```

Arguments

chunk this argument places a RMarkdown chunk

Value

Rmarkdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_packageversion()
```

badge_projectstatus *Add project status badge*

Description

Project-status is based on the repo-status project on <https://www.repostatus.org/>. A project-status badge gives a clear indication of the current state of a project. This helps answer questions about development (whether or not further development is planned) and support (whether bugfixes and user assistance will be given).

Usage

```
badge_projectstatus(status = "concept")
```

Arguments

status one of concept, wip, suspended, abandoned, active, inactive, unsupported, or moved

Value

markdown

Project Statuses

The following list is a literal copy of repostatus.org

- **Concept** – Minimal or no implementation has been done yet, or the repository is only intended to be a limited example, demo, or proof-of-concept.
- **WIP** – Initial development is in progress, but there has not yet been a stable, usable release suitable for the public.
- **Suspended** – Initial development has started, but there has not yet been a stable, usable release; work has been stopped for the time being but the author(s) intend on resuming work.
- **Abandoned** – Initial development has started, but there has not yet been a stable, usable release; the project has been abandoned and the author(s) do not intend on continuing development.
- **Active** – The project has reached a stable, usable state and is being actively developed.
- **Inactive** – The project has reached a stable, usable state but is no longer being actively developed; support/maintenance will be provided as time allows.
- **Unsupported** – The project has reached a stable, usable state but the author(s) have ceased all work on it. A new maintainer may be desired.
- **Moved** - The project has been moved to a new location, and the version at that location should be considered authoritative. This status should be accompanied by a new URL.

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_projectstatus("unsupported")
```

badge_rank	<i>R</i> Packages.io ranking
------------	------------------------------

Description

If this is something you care about, you can add the current rank of your package to your readme.

Usage

```
badge_rank(packagename)
```

Arguments

packagename the name of your package

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rdocumentation](#), [badge_thanks_md](#), [badge_travis](#)

badge_rdocumentation *R documentation badge*

Description

Add a documentation badge, from DataCamp.

Usage

```
badge_rdocumentation(packagename)
```

Arguments

packagename the name of your package

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_thanks_md](#), [badge_travis](#)

Examples

```
badge_rdocumentation("dplyr")
```

badge_thanks_md	<i>Add thanks badge</i>
-----------------	-------------------------

Description

Add a thanks badge and THANKS.md file to your project.

Usage

```
badge_thanks_md(add_file = TRUE)
```

Arguments

add_file Should the badge add the THANKS.md file to your project?

Value

markdown

Source

<https://github.com/paulmolluzzo/thanks-md>

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_travis](#)

badge_travis	<i>Travisbadge creates travis badge.</i>
--------------	--

Description

Travisbadge creates travis badge.

Usage

```
badge_travis(ghaccount = NULL, ghrepo = NULL, branch = NULL,  
             location = ".")
```

Arguments

ghaccount	Github account name
ghrepo	Github repository name
branch	branch name such as master, develop etc
location	defaults to current location

Value

markdown

See Also

Other badges: [badge_codecov](#), [badge_cran_ago](#), [badge_cran_date](#), [badge_cran_version_ago](#), [badge_cran_version_release](#), [badge_cran](#), [badge_github_fork](#), [badge_github_star](#), [badge_last_change_static](#), [badge_last_change](#), [badge_license](#), [badge_minimal_r_version](#), [badge_packageversion](#), [badge_projectstatus](#), [badge_rank](#), [badge_rdocumentation](#), [badge_thanks_md](#)

Examples

```
badge_travis(ghaccount = "johntest", ghrepo = "yourreponame", branch = "master")
```

dynamic_badges_minimal

Add dynamic content to readme

Description

This function returns markdown that you must place in the readme. This is dynamic content. This function returns a RMarkdown piece that does some calculation and a set of badges are placed underneath the code part.

Usage

```
dynamic_badges_minimal(status = "concept", license = NULL,
  last_change = TRUE, minimal_r_version = TRUE, travisfile = NULL,
  codecov = NULL, location = ".")
```

Arguments

status	one of concept, wip, suspended, abandoned, active, inactive or unsupported
license	one of GPL-3, GPL-2, MIT, or CC0.
last_change	adding a last change badge? TRUE or FALSE
minimal_r_version	adding minimal r version badge? TRUE or FALSE
travisfile	add a travis statusbadge
codecov	add a codecov badge
location	defaults to current location

Value

text to put into rmd file

See Also

Other scenarios: [minimal_badges](#)

Examples

```
## Not run:
dynamic_badges_minimal(status = "active", last_change = FALSE, minimal_r_version = FALSE)

## End(Not run)
```

findbadges

What badges are already found in the README.Rmd document

Description

What badges are already found in the README.Rmd document

Usage

```
findbadges(location = ".", name = "README.Rmd")
```

Arguments

location where should we search for README.Rmd?
name which file to place badges in defaults to README.Rmd

Value

list of locations inside of readme

github_credentials_helper

Returns github accountname, repo name and current branch.

Description

Returns github accountname, repo name and current branch.

Usage

```
github_credentials_helper(ghaccount = NULL, ghrepo = NULL,
  branch = NULL, location = ".")
```

Arguments

ghaccount	Github account name
ghrepo	Github repository name
branch	branch name such as master, develop etc
location	defaults to current location

licbadgebuilder	<i>Add license badge</i>
-----------------	--------------------------

Description

Add license badge

Usage

```
licbadgebuilder(licensetype)
```

Arguments

licensetype	one of GPL-3, GPL-2, MIT, or CC0.
-------------	-----------------------------------

Value

markdown

Examples

```
badgecreatr:::licbadgebuilder("GPL-3")
```

minimal_badges	<i>Add a minimum of badges to your project</i>
----------------	--

Description

Returns markdown to manually place in your readme.Rmd file. This is a function you might want to use in the early stages of your project. This function returns badges linkprojectstatusbadge, link(licensebadge), and link(last_change_badge_static).

Usage

```
minimal_badges(status = "concept", license = NULL, date = NULL)
```

Arguments

status one of concept, wip, suspended, abandoned, active, inactive or unsupported
license one of GPL-3, GPL-2, MIT, or CC0.
date a date of your choosing, defaults to current date.

Value

text to put into rmd

See Also

Other scenarios: [dynamic_badges_minimal](#)

Examples

```
minimal_badges("abandoned", "GPL-3")
```

search_git *Return a list of remotes associated with this repo*

Description

a function that checks the remote branches and returns their names.

Usage

```
search_git(location = ".")
```

Arguments

location defaults to current location

Index

badge_codecov, [4](#), [5](#), [6](#), [8–18](#)
badge_cran, [4](#), [5](#), [6](#), [8–18](#)
badge_cran_ago, [4](#), [5](#), [5](#), [6](#), [8–18](#)
badge_cran_date, [4–6](#), [6](#), [8–18](#)
badge_cran_downloads, [7](#)
badge_cran_version_ago, [4–6](#), [7](#), [8–18](#)
badge_cran_version_release, [4–6](#), [8](#), [8](#),
[9–18](#)
badge_github_fork, [4–6](#), [8](#), [9](#), [10–18](#)
badge_github_star, [4–6](#), [8](#), [9](#), [9](#), [10–18](#)
badge_last_change, [4–6](#), [8–10](#), [10](#), [11–18](#)
badge_last_change_static, [4–6](#), [8–10](#), [11](#),
[12–18](#)
badge_license, [4–6](#), [8–11](#), [11](#), [13–18](#)
badge_lifecycle, [12](#)
badge_minimal_r_version, [4–6](#), [8–12](#), [13](#),
[14–18](#)
badge_packageversion, [4–6](#), [8–13](#), [13](#),
[15–18](#)
badge_projectstatus, [4–6](#), [8–14](#), [14](#), [16–18](#)
badge_rank, [4–6](#), [8–15](#), [15](#), [16–18](#)
badge_rdocumentation, [4–6](#), [8–16](#), [16](#), [17](#),
[18](#)
badge_thanks_md, [4–6](#), [8–16](#), [17](#), [18](#)
badge_travis, [4–6](#), [8–17](#), [17](#)
badgcreatr, [2](#)
badgcreatr-package (badgcreatr), [2](#)
badgpaste, [3](#)
badgplacer, [3](#)

dynamic_badges_minimal, [18](#), [21](#)

findbadges, [19](#)

github_credentials_helper, [19](#)

licbadgbuilder, [20](#)

minimal_badges, [19](#), [20](#)

search_git, [21](#)