Package ‘teamr’

October 14, 2022

Title  Send Formatted Messages, Images and Objects to Microsoft 'Teams'

Version 0.0.1

URL  https://github.com/wwwjk366/teamr

BugReports https://github.com/wwwjk366/teamr/issues

Description  Package of wrapper functions using R6 class to send requests to Microsoft 'Teams' <https://products.office.com/en-us/microsoft-teams/group-chat-software> through webhooks.

When you need to share information or data from R to 'Teams', rather than copying/pasting, you can use this package to send well-formatted output from multiple R objects.

License  MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports httr, jsonlite, R6

RoxygenNote 6.1.1

NeedsCompilation no

Author Michael Yan [aut, cre](<https://orcid.org/0000-0002-3845-0934>)

Maintainer Michael Yan <tanbingy@gmail.com>

Repository CRAN

Date/Publication 2019-07-18 16:40:02 UTC

R topics documented:

  action_card .............................................................. 2
  card_section ............................................................ 3
  connector_card ......................................................... 4

Index  6
**Description**

`action_card` creates a potentialAction object which can be added to `connector_card`.

**Usage**

```r
# ac <- action_card$new(type = "ActionCard", name = "Add comment")
```

**Format**

An `R6Class` generator object

**Details**

Assemble `action_card` objects into a `action_card` structure and use the methods to append, modify or send your card to the webhook specified.

**Methods**

- `action_card$new(type, name)` Creates a new `action_card`.
- `add_text_inputs(id, title, is_multiline)` Add `TextInput` to your `action_card`.
- `add_date_inputs(id, title)` Add `DateInput` to your `action_card`.
- `add_mchoice_inputs(id, title, is_multi_select)` Add `MultichoiceInput` to your `action_card`.
- `dump()` Dump current payload

**Properties**

- `payload` R list of payloads (will be parsed into json)

**Examples**

```r
## Not run:
library(teamr)

cc <- connector_card$new(hookurl = "https://outlook.office.com/webhook/..."
cc$text("Of on affixed civilly moments promise explain")
pa <- action_card$new(type = "ActionCard", name = "Add comment")
pa$add_actions()
pa$add_text_inputs(id = "comment", title = "Add comment for this task", is_multi_line = TRUE)
cc$add_potential_action(pa)
cc$send()

## End(Not run)
```
Description

card_section creates an Section object which can be added to connector_card.

Usage

# sec <- card_section$new()

Format

An R6Class generator object

Details

Assemble card_section objects into a action_card structure and use the methods to append, modify or send your card to the webhook specified.

Methods

card_section$new() Creates a new card_section.
text(sec_text) Change the text field of the section.
title(sec_title) Change the title field of the section.
activity_title(sec_activity_title) Change the title field of the section.
activity_sub_title(sec_activity_subtitle) Change the subtitle field of the section.
activity_image(sec_activity_image) Change the image field of the section.
activity_text(sec_activitiy_text) Change the text field of the section.
add_fact(fname, fvalue) Add fact object to section.
add_link_button(btext, burl) Add button with links.
add_image(sec_image, sec_title) Change webhook address.
print() Print out the current payload.
dump() Dump current payload.

Properties

payload  R list of payloads (will be parsed into json)
Examples

```r
## Not run:
library(teamr)

cc <- connector_card$new(hookurl = "https://outlook.office.com/webhook/...")
cc$text("Of on affixed civilly moments promise explain")
sec$text("2018-19 Finals MVP")
sec$activity_sub_title("Kawhi Leonard")
sec$add_fact("Position", "Forward")
sec$title("Player Info")
sec$add_image(sec_image = "https://....jpg", sec_title = "so funny")
sec$activity_image("https://......jpg")
sec$activity_title("Activity Title")
sec$activity_sub_title("Activity Subtitle")
sec$activity_text("Activity text")
cc$add_section(new_section = sec)
cc$print()
cc$send()

## End(Not run)
```

### connector_card

Create a connector_card Structure With teamr

**Description**

connector_card is at the very heart of the teamr package.

**Usage**

```
# cc <- connector_card$new("https://outlook.office.com.....")
```

**Format**

An R6Class generator object

**Details**

Assemble connector_card objects into a connector_card structure and use the methods to append, modify or send your card to the webhook specified.

**Methods**

connector_card$new(hookurl, proxies, http_timeout) Creates a new connector_card.
text(mtext) Change the text field of the card
title(mtitle) Change the title field of the card
summary(msummary) Change the summary field of the card.
color(mcolor) Change the default theme color.
add_link_button(btext, burl) Add button with links.
newhook(nurl) Change webhook address.
print() Print out the current hookurl and payload
send() Send connector_card to specified Microsoft Teams incoming webhook URL.

Properties

hookurl  Microsoft Teams incoming webhooks url.
payload  R list of payloads(will be parsed into json)
proxies  Proxy objects from use_proxy
http_timeout  Timeout of the HTTP request. Default to 3 seconds.

Examples

## Not run:
library(teamr)

cc <- connector_card$new(hookurl = "https://outlook.office.com/webhook/...")
cc$text("Of on affixed civilly moments promise explain")
cc$title("This is a title")
cc$summary("This is a summary")
cc$add_link_button("This is the button Text", "https://github.com/wwwjk366/teamr")
cc$color("#008000")
cc$print()
cc$send()

## End(Not run)
Index

* datasets
  action_card, 2
  card_section, 3
  connector_card, 4

action_card, 2

card_section, 3

connector_card, 4

POST, 2, 3, 5

R6Class, 2–4

timeout, 5

use_proxy, 5