

Package: certemail (via r-universe)

October 1, 2026

Title A Certe R Package for Automated Mailing

Version 1.8.0

Description A Certe R Package for working remotely with emails, using Microsoft 365. This package is part of the 'certedata' universe.

URL <https://certe-medical-epidemiology.github.io/certemail>,
<https://github.com/certe-medical-epidemiology/certemail>

Depends R (>= 4.1.0)

Imports certeprojects (>= 1.14.0), certestyle, crayon (>= 1.4.1),
blastula (>= 0.3.2), dplyr (>= 1.0.0), htmltools (>= 0.5.1),
httr (>= 1.4.0), Microsoft365R (>= 2.4.1), xml2 (>= 1.3.0),
yaml (>= 2.2.0)

Suggests certetoolbox, commonmark (>= 1.7.0), testthat (>= 2.0.0)

License GPL-2

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/testthat/edition 2

Config/pak/sysreqs cmake make libicu-dev libuv1-dev libxml2-dev libssl-dev libx11-dev zlib1g-dev

Repository <https://certe-medical-epidemiology.r-universe.dev>

Date/Publication 2026-04-04 08:35:41 UTC

RemoteUrl <https://github.com/certe-medical-epidemiology/certemail>

RemoteRef HEAD

RemoteSha 041da1fae5f9aa0dd85a1eac5570c99272968832

Contents

account_properties	2
calendar	3
download_mail_attachment	4
mail	6

Index**10**

account_properties	<i>Work with Outlook 365</i>
--------------------	------------------------------

Description

After connecting using `connect_outlook()`, the `get_*`() functions retrieve properties from this connection.

Usage

```
get_name(account = connect_outlook())

get_department(account = connect_outlook())

get_address(account = connect_outlook())

get_phone_numbers(account = connect_outlook())

get_mail_address(account = connect_outlook())

get_inbox_name(account = connect_outlook())

get_drafts_name(account = connect_outlook())

get_certe_name(user = Sys.info()["user"], account = connect_outlook())

get_job_title(user = Sys.info()["user"], account = connect_outlook())

get_certe_name_and_job_title(
  user = Sys.info()["user"],
  account = connect_outlook()
)

get_certe_location(account = connect_outlook())

get_certe_signature(account = connect_outlook(), plain = FALSE)
```

Arguments

account	the Microsoft 365 Outlook account object, e.g. as returned by <code>connect_outlook()</code>
user	Certe user ID, to look up the <code>secret</code> <code>user.{user}.fullname</code>
plain	a <code>logical</code> to indicate whether textual formatting should not be applied

Details

The `get_certe_name()` and `get_certe_name_and_job_title()` functions first look for the `secret` `user.{user}.fullname` and falls back to `get_name()` if it is not available.

calendar

Connect to Calendar in Outlook 365

Description

After connecting using `connect_outlook()`, the `get_*`() functions retrieve properties from this connection.

Usage

```
calendar_get_events(  
  start_date = Sys.Date(),  
  end_date = start_date,  
  account = connect_outlook()  
)  
  
calendar_unavailable_users(  
  start_date = Sys.Date(),  
  end_date = start_date,  
  account = connect_outlook()  
)
```

Arguments

<code>start_date, end_date</code>	date range
<code>account</code>	the Microsoft 365 Outlook account object, e.g. as returned by <code>connect_outlook()</code>

Details

`calendar_get_events()` retrieves a `data.frame` with all appointments from the shared calendar set in `account`.

`calendar_unavailable_users()` returns a vector of appointment subjects that are shown as "free" within `start_date` and `end_date`.

download_mail_attachment

Download Email Attachments Using Microsoft 365

Description

This uses the Microsoft365R package to download email attachments via Microsoft 365. Connection will be made using [connect_outlook\(\)](#).

Usage

```
download_mail_attachment(
  path = getwd(),
  filename = "mail_{name}_{datetime}",
  search = "hasattachment:yes",
  search_subject = NULL,
  search_from = NULL,
  search_when = NULL,
  search_attachment = NULL,
  folder = get_inbox_name(account = account),
  n = 5,
  sort = "received desc",
  overwrite = TRUE,
  skip_inline = TRUE,
  account = connect_outlook(),
  interactive_mode = interactive()
)
```

Arguments

path	location to save attachment(s) to
filename	new filename for the attachments, use NULL to not alter the filename. For multiple files, the files will be appended with a number. The following texts can be used for replacement (invalid filename characters will be replaced with an underscore): • "{date}", the received date of the email in format "yyyymmdd" • "{time}", the received time of the email in format "HHMMSS" • "{datetime}" and "{date_time}", the received date and time of the email in format "yyyymmdd_HHMMSS" • "{name}", the name of the sender of the email • "{address}", the email address of the sender of the email • "{original}", the original filename of the attachment
search	an ODATA filter, ignores sort and defaults to search only mails with attachments
search_subject	a character , equal to search = "subject: (search_subject)", case-insensitive

search_from	a character , equal to search = "from:(search_from)", case-insensitive
search_when	a Date vector of size 1 or 2, equal to search = "received:date1..date2", see <i>Examples</i>
search_attachment	a character to use a regular expression for attachment file names
folder	email folder name to search in, defaults to Inbox of the current user by calling get_inbox_name()
n	maximum number of emails to search
sort	initial sorting
overwrite	a logical to indicate whether existing local files should be overwritten
skip_inline	a logical to indicate whether inline attachments such as meta images must be skipped
account	a Microsoft 365 account to use for searching the mails. This has to be an object as returned by connect_outlook() or Microsoft365R::get_business_outlook() .
interactive_mode	a logical to indicate interactive mode. In non-interactive mode, all attachments within the search criteria will be downloaded.

Details

search_* arguments will be matched as 'AND'.

search_from can contain any sender name or email address. If search_when has a length over 2, the first and last value will be taken.

Refer to [this Microsoft Support page](#) for all filtering options using the search argument.

See Also

[mail\(\)](#)

Examples

```
## Not run:
download_mail_attachment(search_when = "2021-10-28")

if (require("certetoolbox")) {
  download_mail_attachment(search_from = " groningen",
                           search_when = last_week(),
                           filename = " groningen_{date_time}")

  download_mail_attachment(search_from = "drenthe",
                           search_when = yesterday(),
                           n = 1)
}

## End(Not run)
```

mail

Send Emails Using Microsoft 365

Description

These functions use the Microsoft365R R package to send emails via Microsoft 365. They require an Outlook Business account.

Usage

```
mail(
  body,
  subject = "",
  to = NULL,
  cc = read_secret("mail.auto_cc"),
  bcc = read_secret("mail.auto_bcc"),
  reply_to = NULL,
  attachment = NULL,
  header = FALSE,
  footer = FALSE,
  background = certestyle::colourpicker("certeblauw6"),
  send = TRUE,
  markdown = TRUE,
  signature = get_certe_signature(account = account),
  automated_notice = !interactive(),
  save_location = read_secret("mail.export_path"),
  sent_subfolder = read_secret("mail.sent_subfolder"),
  expect = NULL,
  account = connect_outlook(),
  account_teams = connect_teams(),
  identifier = NULL,
  ...
)

mail_plain(
  body,
  subject = "",
  to = NULL,
  cc = read_secret("mail.auto_cc"),
  bcc = read_secret("mail.auto_bcc"),
  reply_to = read_secret("mail.auto_reply_to"),
  attachment = NULL,
  send = TRUE,
  ...
)

mail_image(image_path, width = NULL, ...)
```

```
mail_on_error(expr, to = read_secret("mail.error_to"), ...)
```

```
mail_is_sent(project_number, date = Sys.Date(), account = connect_outlook())
```

Arguments

body	body of email, allows markdown if markdown = TRUE
subject	subject of email
to	field 'to', can be character vector
cc	field 'CC', can be character vector
bcc	field 'BCC', can be character vector
reply_to	field 'reply-to'
attachment	character (vector) of file location(s), or (a list of) data.frames . In case of file locations, these can also be SharePoint locations in the certeprojects::teams_projects_channel() .
header, footer	extra text for header or footer, allows markdown if markdown = TRUE. See blastula::blocks() to build blocks for these sections.
background	background of the surrounding area in the email. Use "", NULL or FALSE to remove background.
send	directly send email, FALSE will show the email in the Viewer pane and will ask whether the email should be saved to the Drafts folder of the current Microsoft 365 user.
markdown	treat body, header and footer as markdown
signature	text to print as email signature, or NULL to omit it, defaults to get_certe_signature()
automated_notice	a logical to print a notice that the mail was sent automatically (default is TRUE if not in interactive() mode)
save_location	location to save email object to, which consists of all email details and can be printed in the R console
sent_subfolder	mail folder within Sent Items in the Microsoft 365 account, to store the mail if !interactive()
expect	expression which should return TRUE prior to sending the email
account	a Microsoft 365 account to use for sending the mail. This has to be an object as returned by connect_outlook() or Microsoft365R::get_business_outlook() . Using account = FALSE is equal to setting send = FALSE.
account_teams	a Microsoft 365 account to use for retrieving MS Teams contents. This has to be an object as returned by certeprojects::connect_teams() or Microsoft365R::get_team() .
identifier	a mail identifier to be printed at the bottom of the email. Defaults to project_identifier() . Use FALSE to not print an identifier.
...	arguments for mail()
image_path	path of image
width	required width of image, must be in CSS style such as "200px" or "100%"

expr expression to test, an email will be sent if this expression returns an error
project_number Number of a project. Will be used to check the grey identifier in the email.
date A date, defaults to today. Will be evaluated in `as.Date()`. Can also be of length 2 for a date range.

Details

`mail_on_error()` can be used for automated scripts.

`mail_plain()` sends a plain email, without markdown support and with no signature.

Use `mail_is_sent()` to check whether a project email was sent on a certain date from any Sent Items (sub)folder. The function will search for the grey identifier in the email body, which is formatted as `[-yymmdd][0-9]+[-project_number][^0-9a-z]`. It returns TRUE if any email was found, and FALSE otherwise. If TRUE, the `name` will contain the date(s) and time(s) of the sent email.

See Also

`download_mail_attachment()`

Examples

```

mail("test123", "test456", to = "mail@domain.com", account = NULL)

mail_plain("test123", "test456", to = "mail@domain.com", account = NULL)

mail(mail_image(image_path = system.file("test.jpg", package = "certemail")),
      "test456", to = "mail@domain.com", account = NULL)

## Not run:

# attachments can be local files, SharePoint files, or data.frame objects
mail("test123", "test456", to = "mail@domain.com", account = NULL,
      attachment = c("C:/existing_file.pdf",
                     "Testproject - p123/sharepoint_file.pdf"))

# alternatively, use certprojects::project_get_file() to easily add a SharePoint file:
mail("test123", "test456", to = "mail@domain.com", account = NULL,
      attachment = project_get_file("Analyse.*test.pdf", 123))

## End(Not run)

# data.frames as body will be transformed with certstyle::plain_html_table()
mail(body = mtcars[1:5, ],
      subject = "Check these cars!",
      to = "somebody@domain.org",
      account = FALSE)

# but better is to add as an attachment - they will become Excel files
mail(body = "Hello there",

```

```
subject = "Check these cars!",
to = "somebody@domain.org",
attachment = mtcars[1:5, ],
account = FALSE)

# use list() to add multiple data sets / attachments
mail(body = "Hello there",
      subject = "Check these cars and flowers!",
      to = "somebody@domain.org",
      attachment = list(mtcars[1:5, ],
                        iris),
      account = FALSE)
```

Index

account_properties, 2
as.Date(), 8

blastula::blocks(), 7

calendar, 3
calendar_get_events(calendar), 3
calendar_get_events(), 3
calendar_unavailable_users(calendar), 3
calendar_unavailable_users(), 3
certeprojects::connect_teams(), 7
certeprojects::teams_projects_channel(),
7
character, 4, 5
connect_outlook(), 2–5, 7

data.frame, 3, 7
Date, 5
download_mail_attachment, 4
download_mail_attachment(), 8

get_address(account_properties), 2
get_certe_location
(account_properties), 2
get_certe_name(account_properties), 2
get_certe_name(), 3
get_certe_name_and_job_title
(account_properties), 2
get_certe_name_and_job_title(), 3
get_certe_signature
(account_properties), 2
get_certe_signature(), 7
get_department(account_properties), 2
get_drafts_name(account_properties), 2
get_inbox_name(account_properties), 2
get_inbox_name(), 5
get_job_title(account_properties), 2
get_mail_address(account_properties), 2
get_name(account_properties), 2
get_name(), 3

get_phone_numbers(account_properties),
2

interactive(), 7

list, 7
logical, 2, 5, 7

mail, 6
mail(), 5, 7
mail_image(mail), 6
mail_is_sent(mail), 6
mail_is_sent(), 8
mail_on_error(mail), 6
mail_on_error(), 8
mail_plain(mail), 6
mail_plain(), 8
Microsoft365R::get_business_outlook(),
5, 7
Microsoft365R::get_team(), 7

name, 8

project_identifier(), 7

secret, 2, 3