Package: RCognito (via r-universe)

June 1, 2026

Type Package

Title A Simplified Interface for 'Amazon Cognito'

Version 0.2.0

Depends R (>= 4.1.0)

Maintainer Sanjaya J Shetty <shettysanjaya01@gmail.com>

BugReports https://github.com/SanjayShetty01/RCognito/issues

Description Simplifies integration with 'Amazon Cognito' (<https://aws.amazon.com/cognito/>) for R developers, enabling easy management of user authentication, registration, and password flows.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Imports paws.security.identity

Suggests mockery, testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://sanjayshetty01.r-universe.dev

Date/Publication 2026-05-31 14:23:24 UTC

RemoteUrl https://github.com/sanjayshetty01/rcognito

RemoteRef HEAD

RemoteSha 4a96ffaa961460ce50f148b5609e92c46537320b

Contents

change_password 2
confirm_forgot_password 2
confirm_sign_up_user 3
forgot_password 3

get_user_list_from_cognito 4
is_valid_email 4
is_valid_token 5
sign_in 5
sign_up_user 6

Index **7**

`change_password` *Change User Password*

Description

Change User Password

Usage

```
change_password(old_password, new_password, token)
```

Arguments

`old_password`	A character string representing the user's current password.
`new_password`	A character string representing the user's new password.
`token`	A character string representing the user's access token.

Value

A message indicating the success of the password change, or an error if it fails.

`confirm_forgot_password` *Confirm Forgot Password*

Description

This function confirms the forgot password process by setting a new password for a user.

Usage

```
confirm_forgot_password(client_id, username, confirmation_code, new_password)
```

Arguments

`client_id`	A character string representing the app client ID.
`username`	A character string for the user's username.
`confirmation_code`	A character string for the confirmation code sent to the user.
`new_password`	A character string for the user's new password.

Value

Nothing (displays a message on success or raises an error if it fails).

---

confirm_sign_up_user *Confirm User Sign Up*

Description

This function confirms the sign-up process for a user in AWS Cognito.

Usage

```
confirm_sign_up_user(client_id, userpool, username, verification_code)
```

Arguments

client_id	A character string representing the app client ID.
userpool	A character string representing the user pool ID.
username	A character string for the user's username.
verification_code	A character string for the verification code sent to the user.

Value

Nothing (displays a message on success or raises an error if it fails).

---

forgot_password *Initiate Forgot Password Request*

Description

Initiate Forgot Password Request

Usage

```
forgot_password(client_id, username)
```

Arguments

client_id	A character string representing the app client ID.
username	A character string for the user's username.

Value

Nothing (displays a message on success or raises an error if it fails).

get_user_list_from_cognito *Get User List from Cognito*

Description

Retrieve the list of usernames and email addresses from Cognito.

Usage

```
get_user_list_from_cognito(cognito_client, userpool)
```

Arguments

cognito_client	The Cognito client.
userpool	A character string representing the user pool ID.

Value

A list containing usernames and email addresses.

is_valid_email *Check if an email address is valid.*

Description

This function validates the format of an email address.

Usage

```
is_valid_email(email)
```

Arguments

email	The email address to be validated.

Value

TRUE if the email is valid, otherwise an error is thrown.

is_valid_token — *Check if an access token is valid.*

Description

This function validates the format of an access token.

Usage

```
is_valid_token(token)
```

Arguments

token	The access token to be validated.

Value

TRUE if the token is valid, otherwise an error is thrown.

sign_in — *Sign in to AWS Cognito*

Description

Sign in to AWS Cognito

Usage

```
sign_in(client_id, username, password, authflow = "USER_PASSWORD_AUTH")
```

Arguments

client_id	A character string representing the app client ID.
username	A character string for the user's username.
password	A character string for the user's password.
authflow	A character string specifying the authentication flow (default: "USER_PASSWORD_AUTH").

Value

A list with user information and tokens if successful; throws an error otherwise.

`sign_up_user` *Sign up a new user to AWS Cognito*

Description

Sign up a new user to AWS Cognito

Usage

```
sign_up_user(client_id, email, username, password, ...)
```

Arguments

`client_id`	A character string representing the app client ID.
`email`	A character string for the user's email address.
`username`	A character string for the user's username.
`password`	A character string for the user's password.
`...`	Additional named user attributes for Cognito (e.g., 'phone_number').

Value

A list with sign-up confirmation details if successful; throws an error otherwise.

Index

change_password, 2
confirm_forgot_password, 2
confirm_sign_up_user, 3

forgot_password, 3

get_user_list_from_cognito, 4

is_valid_email, 4
is_valid_token, 5

sign_in, 5
sign_up_user, 6