

# Package ‘labstepR’

April 1, 2019

**Title** An R Package for Working with the Labstep API

**Version** 0.0.1

**Description** Convenient wrapper functions around the Labstep API. Document your experiments, attach files, comments and more.

**Depends** R (>= 3.1.0)

**Imports** http

**License** MIT

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1.9000

**NeedsCompilation** no

**Author** Barney Walker [aut, cre]

**Maintainer** Barney Walker <barney@labstep.com>

## R topics documented:

addComment . . . . .	2
attachFile . . . . .	2
attachProtocol . . . . .	3
createExperiment . . . . .	4
createProject . . . . .	4
createProtocol . . . . .	5
createResource . . . . .	5
getExperiment . . . . .	6
getExperiments . . . . .	7
getProject . . . . .	7
getProtocol . . . . .	8
getProtocols . . . . .	8
getResource . . . . .	9
getResources . . . . .	9
login . . . . .	10

<b>Index</b>	<b>11</b>
--------------	-----------

---

addComment	<i>Adds a comment to a thread on Labstep</i>
------------	--

---

### Description

This function allows you to comment on different content types

### Arguments

user	A labstep user object. Must contain an 'api\_key' field. Returned from 'login' commande
object	The labstep object you want to comment on. Can be 'protocol', 'experiment' or 'resource'
body	The body of the comment in plain text.

### Value

Returns a 'comment' object

### Examples

```
user <- login("demo@labstep.com", "demopassword")
expName = 'My First Experiment'
expDesc = 'An experiment testing the labstepR library'
experiment <- createExperiment(user, expName, expDesc)
commentBody = 'An experiment testing the labstepR library'
comment <- addComment(user, experiment, commentBody)
print(comment)
```

---

attachFile	<i>Attaches a file to a thread on Labstep</i>
------------	---

---

### Description

This function allows you to attach a file to different content types

### Usage

```
attachFile(user, object, filepath, caption)
```

### Arguments

user	A labstep user object. Must contain an 'api\_key' field. Returned from 'login' command
object	The labstep object you want to comment on. Can be 'protocol', 'experiment' or 'resource'
filepath	The path to ther file to attach.
caption	A caption describing your file (optional)

**Value**

Returns a 'comment' object with the file attached

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
expName = 'My First Experiment'
expDesc = 'An experiment testing the labstepR library'
experiment <- createExperiment(user, expName, expDesc)
filepath = 'mydata.RData'
save.image(filepath)
caption = 'The R workspace after running this script'
file <- attachFile(user, experiment, filepath, caption)
print(file)
```

---

attachProtocol

---

*Attach a Protocol to an Experiment on Labstep*


---

**Description**

This function allows you to attach a Protocol to an existing Experiment.

**Usage**

```
attachProtocol(user, experiment, protocol)
```

**Arguments**

user	A labstep user object. Must contain an 'api_key' field. Returned from 'login' command
experiment	The experiment you want to attach the protocol to. Must have at least an 'id' field.
protocol	The protocol you want to attach. Must have at least an 'id' field

**Value**

Returns an 'experiment' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
expName = 'My First Experiment'
expDesc = 'An experiment testing the labstepR library'
experiment <- createExperiment(user, expName, expDesc)
protocol <- createProtocol(user, 'My First Protocol')
experiment <- attachProtocol(user, experiment, protocol)
print(experiment)
```

---

createExperiment	Create an Experiment on Labstep
------------------	---------------------------------

---

**Description**

This function allows you to create an experiment card.

**Usage**

```
createExperiment(user, name, description = NULL, protocolIds = NULL)
```

**Arguments**

user	A labstep user object. Must contain an 'api\_key' field. Returned from 'login' command
name	The name you want to give your experiment.
description	Give your experiment a description to help you remember what it was about.
protocolIds	List of ids for protocols to attach to the experiment.

**Value**

Returns an 'experiment' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
expName = 'My First Experiment'
expDesc = 'An experiment testing the labstepR library'
experiment <- createExperiment(user, expName, expDesc)
print(experiment)
```

---

createProject	Create a Project on Labstep
---------------	-----------------------------

---

**Description**

This function allows you to create a Labstep Project.

**Usage**

```
createProject(user, name)
```

**Arguments**

user	A labstep user object. Must contain an 'api\_key' field. Returned from 'login' command
name	The name you want to give your Project.

**Value**

Returns an 'project' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
project <- createProject(user, 'My First Project')
print(project)
```

---

createProtocol	<i>Create a Protocol on Labstep</i>
----------------	-------------------------------------

---

**Description**

This function allows you to create a protocol.

**Usage**

```
createProtocol(user, name)
```

**Arguments**

user	A labstep user object. Must contain an 'api_key' field. Returned from 'login' command
name	The name you want to give your protocol.

**Value**

Returns an 'protocol' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
protocol <- createProtocol(user, 'My First Protocol')
print(protocol)
```

---

createResource	<i>Create a Resource on Labstep</i>
----------------	-------------------------------------

---

**Description**

This function allows you to create a resource on Labstep.

**Usage**

```
createResource(user, name)
```

**Arguments**

user	A labstep user object. Must contain an 'api\_key' field. Returned from 'login' command.
name	The name you want to give your resource

**Value**

Returns a 'resource' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
resource <- createResource(user, 'My First Resource')
print(resource)
```

---

getExperiment

*Get Experiment*

---

**Description**

This function allows you to retrieve metadata for a single experiment

**Usage**

```
getExperiment(user, id)
```

**Arguments**

user	A labstep user object. Must contain an 'api\_key' field. Returned from 'login' command
id	ID of the experiment to retrieve

**Value**

Returns an 'experiment' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
experiment <- getExperiment(user, 123)
print(experiment)
```

---

getExperiments	<i>Get Experiments</i>
----------------	------------------------

---

**Description**

This function allows you to search your Experiments on Labstep.

**Usage**

```
getExperiments(user, count = 100)
```

**Arguments**

user	A labstep user object. Must contain an 'api_key' field. Returned from 'login' command
count	Number of results to return. Defaults to 100.

**Value**

Returns a list of 'experiment' objects

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
experiments <- getExperiments(user)
print(experiments)
```

---

getProject	<i>Get Project</i>
------------	--------------------

---

**Description**

This function allows you to retrieve metadata for a single project

**Usage**

```
getProject(user, id)
```

**Arguments**

user	A labstep user object. Must contain an 'api_key' field. Returned from 'login' command
id	ID of the project to retrieve

**Value**

Returns a 'project' object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
project <- getProject(user, 123)
print(project)
```

---

`getProtocol`*Get Protocol*

---

**Description**

This function allows you to retrieve metadata for a single protocol

**Usage**

```
getProtocol(user, id)
```

**Arguments**

<code>user</code>	A labstep user object. Must contain an ‘api_key’ field. Returned from ‘login’ command
<code>id</code>	ID of the protocol to retrieve

**Value**

Returns a ‘protocol’ object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
protocol <- getProtocol(user, 123)
print(protocol)
```

---

`getProtocols`*Get Protocols*

---

**Description**

This function allows you to search your Protocol library on Labstep.

**Usage**

```
getProtocols(user, count = 100)
```

**Arguments**

<code>user</code>	A labstep user object. Must contain an ‘api_key’ field. Returned from ‘login’ command
<code>count</code>	Number of results to return. Defaults to 100.



**Value**

Returns a list of ‘protocol’ objects

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
protocols <- getProtocols(user)
print(protocols)
```

---

getResource

*Get Resource*

---

**Description**

This function allows you to retrieve metadata for a single resource

**Usage**

```
getResource(user, id)
```

**Arguments**

user	A labstep user object. Must contain an ‘api_key’ field. Returned from ‘login’ command
id	ID of the resource to retrieve

**Value**

Returns a ‘resource’ object

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
resource <- getResource(user, 123)
print(resource)
```

---

getResources

*Get Resources*

---

**Description**

This function allows you to search your Resource library on Labstep.

**Usage**

```
getResources(user, count = 100)
```

**Arguments**

user	A labstep user object. Must contain an 'api_key' field. Returned from 'login' command
count	Number of results to return. Defaults to 100.

**Value**

Returns a list of 'resource' objects

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
resources <- getResources(user)
print(resources)
```

---

login	<i>Authenticate a Labstep user</i>
-------	------------------------------------

---

**Description**

This function allows you to retrieve the user details for your Labstep account. It also returns your unique API key which you need to use the other Labstep API functions.

**Usage**

```
login(username, password)
```

**Arguments**

username	Your Labstep username (usually the email address you signed up with)
password	Your Labstep password

**Value**

Returns a 'user' object.

**Examples**

```
user <- login("demo@labstep.com", "demopassword")
print(user)
```

# Index

\*Topic **apikey**

login, [10](#)

\*Topic **login**

login, [10](#)

addComment, [2](#)

attachFile, [2](#)

attachProtocol, [3](#)

createExperiment, [4](#)

createProject, [4](#)

createProtocol, [5](#)

createResource, [5](#)

getExperiment, [6](#)

getExperiments, [7](#)

getProject, [7](#)

getProtocol, [8](#)

getProtocols, [8](#)

getResource, [9](#)

getResources, [9](#)

login, [10](#)