# Package 'REDCapR'

December 3, 2013

2000 not 3, 2013
Title Interaction between R and REDCap
<b>Description</b> Encapsulates functions to streamline calls from R
Version 0.1-5
Date 2013-12-03
Author Will Beasley, David Bard, Thomas Wilson
Maintainer 'Will Beasley' < wibeasley@hotmail.com>
<pre>URL http://ouhsc.edu/bbmc/</pre>
<b>Depends</b> R(>= 2.14.0),stats
Imports methods,plyr,RCurl
Suggests devtools,ggplot2,knitr,scales,testit,testthat,xtable
License GPL-2
LazyData TRUE
BuildVignettes TRUE
VignetteBuilder knitr
Collate 'REDCapR-package.R' 'redcap_read.R' 'redcap_write.R' 'validate_for_write.R'
R topics documented:
REDCapR
Index 5
REDCapR REDCapR
Description

REDCapR

2 redcap\_read

radca	p_read
reuca	p_i eau

Read records from a REDCap project.

## **Description**

This function uses REDCap's API to select and return data.

#### Usage

```
redcap_read(redcap_uri, token, records = NULL,
  records_collapsed = NULL, fields = NULL,
  fields_collapsed = NULL, verbose = TRUE,
  cert_location = NULL)
```

#### **Arguments**

redcap\_uri The URI of the REDCap project. Required.

token The user-specific string that serves as the password for a project. Required.

records An array, where each element corresponds to the ID of a desired record. Op-

tional.

records\_collapsed

A single string, where the desired ID values are separated by commas. Optional.

fields An array, where each element corresponds a desired project field. Optional.

fields\_collapsed

A single string, where the desired field names are separated by commas. Op-

tional.

verbose A boolean value indicating if messages should be printed to the R console during

the operation. Optional.

SSL verification. If the value is missing or NULL, the server's identity will be verified using a recent CA bundle from the cURL website. See the details below.

Optional.

#### **Details**

I like how PyCap creates a 'project' object with methods that read and write from REDCap. However this isn't a style that R clients typically use. I like the logic that it's associated with a particular REDCap project that shouldn't change between calls. As a compromise, I think I'll wrap the uri, token, and cert location into a single S4 object that's passed to these methods. It will make these calls take less space.

The 'REDCapR' package includes a recent version of the Bundle of CA Root Certificates from the official cURL site. This version is used by default, unless the 'cert\_location' parameter is given another location.

## Value

Currently, a list is returned with the following elements,

1. data: an R data.frame of the desired records and columns.

validate\_for\_write 3

- 2. raw\_csv: the text of comma separated values returned by REDCap through RCurl.
- records\_collapsed: the desired records IDs, collapsed into a single string, separated by commas.
- fields\_collapsed: the desired field names, collapsed into a single string, separated by commas.
- 5. elapsed\_seconds: the duration of the function.
- 6. status\_message: a boolean value indicating if the operation was apparently successful.

#### Author(s)

Will Beasley

#### References

The official documentation can be found on the 'API Examples' page on the REDCap wiki (https://iwg.devguard.com/trac/redcap/wiki/ApiExamples). A user account is required.

The official cURL site discusses the process of using SSL to verify the server being connected to.

## **Examples**

```
## Not run:
library(REDCapR) #Load the package into the current R session.
uri <- "https://miechvprojects.ouhsc.edu/redcap/api/"</pre>
token <- "9446D2E3FAA71ABB815A2336E4692AF3"
#Return all records and all variables.
ds_all_rows_all_fields <- redcap_read(redcap_uri=uri, token=token)$data</pre>
#Return only records with IDs of 1 and 3
desired_records_v1 <- c(1, 3)</pre>
ds\_some\_rows\_v1 <- redcap\_read(
   redcap_uri=uri,
   token=token,
   records=desired_records_v1
)$data
#Return only the fields recordid, first_name, and age
desired_fields_v1 <- c("recordid", "first_name", "age")</pre>
ds_some_fields_v1 <- redcap_read(</pre>
   redcap_uri=uri,
   token=token,
   fields=desired_fields_v1
)$data
## End(Not run)
```

validate\_for\_write

*Inspect a* data. frame to anticipate problems before writing to a RED-Cap project.

### **Description**

This set of functions inspect a data. frame to anticipate problems before writing with REDCap's API.

4 validate\_for\_write

#### Usage

```
validate_for_write( df )
validate_no_logical( df )
validate_no_uppercase( df )
```

#### **Arguments**

df

The data.frame containing the dataset used to update the REDCap project. Required.

#### **Details**

All functions listed in the Usage section above inspect a specific aspect of the dataset. The validate\_for\_read() function executes all these individual validation checks. It allows the client to check everything with one call.

#### Value

A data.frame, where each potential violation is a row. The two columns are:

- 1. field\_name: The name of the data.frame that might cause problems during the upload.
- 2. field\_index: The position of the field. (For example, a value of '1' indicates the first column, while a '3' indicates the third column.)
- 3. concern: A description of the problem potentially caused by the field.
- 4. suggestion: A potential solution to the concern.

## Author(s)

Will Beasley

## **Examples**

```
df <- data.frame(
  recordid = 1:4,
  flag_logical = c(TRUE, TRUE, FALSE, TRUE),
  flag_Uppercase = c(4, 6, 8, 2)
)
validate_for_write(df = df)</pre>
```

## **Index**

```
redcap_read, 2
REDCapR, 1
REDCapR-package (REDCapR), 1

validate_for_write, 3
validate_no_logical
          (validate_for_write), 3
validate_no_uppercase
          (validate_for_write), 3
```