

Package ‘AzureAppInsights’

October 1, 2021

Type Package

Title Include Azure Application Insights in Shiny Apps

Version 0.3.0

Description Imports Azure Application Insights for web pages into Shiny apps via Microsoft's JavaScript snippet.
Allows app developers to submit page tracking and submit events.

License MIT + file LICENSE

BugReports <https://github.com/stefanedwards/AzureAppInsights/issues>

Depends R (>= 4.0.0)

Imports shiny (>= 1.5.0), rlang (>= 0.4.11), assertthat (>= 0.2.0),
jsonlite (>= 1.7.2), lubridate (>= 1.7)

Encoding UTF-8

Suggests testthat, here

RoxygenNote 7.1.1

Collate '0aux.R' 'AzureAppInsights-package.R' 'cfg.R' 'demo.R'
'include_snippet.R' 'tracking.R'

NeedsCompilation no

Author Stefan McKinnon Høj-Edwards [aut, cre],
Kamstrup A/S [cph]

Maintainer Stefan McKinnon Høj-Edwards <smhe@kamstrup.dk>

Repository CRAN

Date/Publication 2021-10-01 11:10:02 UTC

R topics documented:

AzureAppInsights-package	2
config	2
is_instrumentation_key	3
startAzureAppInsights	4
trackEvent	5

Index	7
--------------	----------

AzureAppInsights-package

Azure Application Insights for web pages

Description

Add Azure Application Insights tracking to a Shiny App. *Requires an active Azure subscription and Application Insights instrumentation key!* Based on <https://docs.microsoft.com/en-us/azure/azure-monitor/app/javascript> / <https://github.com/microsoft/ApplicationInsights-JS>.

Details

Documentation in this page will be limited, as most is explained on the main page.

Supports so far only pageViews (automatically sent), autoTrackPageVisitTime (when configured with [config](#)), customEvents (see [trackEvent](#)).

Author(s)

Stefan McKinnon Høj-Edwards <smhe@kamstrup.dk>

See Also

Useful links:

- Report bugs at <https://github.com/stefanedwards/AzureAppInsights/issues>

config

Configure Azure Application Insights

Description

Ensures an instrumentationKey/connectionString and appId is provided.

Usage

```
config(  
  appId,  
  instrumentationKey,  
  connectionString,  
  autoTrackPageVisitTime = TRUE,  
  ...  
)
```

Arguments

appId	String for identifying your app, if you use same Application Insights for multiple apps.
instrumentationKey, connectionString	connectionString is preferred for newer accounts. Must contain both InstrumentationKey and IngestionEndpoint.
autoTrackPageVisitTime	Submits how long time a user spent on the *previous* page (see website for more information).
...	Additional options, as given in https://docs.microsoft.com/en-us/azure/azure-monitor/app/javascript#configuration . No checks performed here.

Details

See <https://docs.microsoft.com/en-us/azure/azure-monitor/app/javascript#configuration> for explanation of options.

If jsonlite is playing tricks on the arguments given, wrap the value with I. E.g. if you want to force an atomic vector of length 1 to be parsed as an array, use I(3.14).

Value

List.

is_instrumentation_key

Check if string matches pattern for an instrumentation key.

Description

Check if string matches pattern for an instrumentation key.

Usage

```
is_instrumentation_key(x)
```

Arguments

x A string containing nothing else but an instrumentation key.

Value

Logical value.

startAzureAppInsights *Include and run Azure Application Insights for web pages*

Description

Include the JS snippet in your ui-function with `includeAzureAppInsights` and start the tracking with `startAzureAppInsights` in your server-function.

Usage

```
startAzureAppInsights(
  session,
  cfg,
  instance.name = "appInsights",
  ld = 0,
  useXhr = TRUE,
  crossOrigin = "anonymous",
  onInit = NULL,
  heartbeat = 3e+05,
  extras = list(),
  include.ip = FALSE,
  cookie.user = FALSE
)

includeAzureAppInsights()
```

Arguments

<code>session</code>	The session object passed to function given to <code>shinyServer</code> .
<code>cfg</code>	List-object from config .
<code>instance.name</code>	Global JavaScript Instance name defaults to "appInsights" when not supplied. <i>NOT</i> the app's name. Used for accessing the instance from other JavaScript routines.
<code>ld</code>	Defines the load delay (in ms) before attempting to load the sdk. -1 = block page load and add to head. (default) = 0ms load after timeout,
<code>useXhr</code>	Logical, use XHR instead of fetch to report failures (if available).
<code>crossOrigin</code>	When supplied this will add the provided value as the cross origin attribute on the script tag.
<code>onInit</code>	Once the application insights instance has loaded and initialized this callback function will be called with 1 argument – the sdk instance
<code>heartbeat</code>	Integer, how often should the heartbeat beat – or set to FALSE to disable.
<code>extras</code>	(Named) list of values to add to any tracking.
<code>include.ip</code>	Logical, adds ip to all tracking's <code>customDimension</code> . See note.
<code>cookie.user</code>	Logical, when TRUE sets a cookie with a random string and submits this along with any tracking with the key <code>userid</code> .

Value

Methods sends data to client's browser; returns the sent list, invisibly.

Tracking users' ip-address

Generally, Azure's Application Insight does not collect the users' ip-address, due to it being somewhat sensitive data ([link](#)).

`startAzureAppInsights` however has the argument 'include.ip' which, when set to TRUE, will add the entry ip to all trackings. The tracked ip-address is taken from `session$request$REMOTE_ADDR`, which is an un-documented feature and may or may not be the users ip-address.

References

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/javascript> and <https://github.com/microsoft/ApplicationInsights-JS> and <https://docs.microsoft.com/en-us/azure/azure-monitor/app/ip-collection?tabs=net>

trackEvent	<i>Sends an event or set of metrics to Application Insights</i>
------------	---

Description

Use `trackEvent` for tracking a single event together with any extra properties.

Use `trackMetric` to track a summary of some measured metrics.

Usage

```
trackEvent(session, name, properties)
```

```
trackMetric(session, name, metrics, properties)
```

Arguments

session	The session object passed to function given to shinyServer.
name	Name of the event.
properties	List of properties to track. appId and any extras given in <code>startAzureAppInsights</code> is automatically inserted.
metrics	Numeric vector of values to calculate summary on. Non-finite values are removed.

Value

Method sends data to client's browser; returns the sent list, invisibly.

Tracking Metrics

Individual measured values are not sent to Application Insights. Instead, summaries of the values (mean, range, average, standard deviation) are sent. *Note:* Standard deviation doesn't quite work yet.

Before calculating summaries, non-finite values are removed (see [is.finite](#)). If there are no values in `metrics`, nothing is sent.

Index

AzureAppInsights
 (AzureAppInsights-package), 2
AzureAppInsights-package, 2

config, 2, 2, 4

includeAzureAppInsights
 (startAzureAppInsights), 4
is.finite, 6
is_instrumentation_key, 3

startAzureAppInsights, 4, 5

trackEvent, 2, 5
trackMetric (trackEvent), 5