How to use variables in templates

Text variables

To use variables in templates you need to use the notation *|VARIABLE_NAME|* . In this way, the content between the delimiters *| and |* will be replaced by whatever value that variable has. Here are some examples of variables you can use:

Tag variables

`*|TAG_NAME|*` contains the tag name
`*|TAG_DESCRIPTION|*` contains the tag description

App variables

`*|APP_NAME|*` contains the name of the App where the visited tag belongs to. `*|APP_DESCRIPTION|*` contains the App description where the visited tag belongs to.

User variables

`*|USER_NAME|*` contains the name of the logged in user `*|USER_EMAIL|*` contains the e-mail of the logged-in user

Asset variables

Since in TicTAP a tag can be "stuck" on an asset, the asset information can be accessed through asset variables.

`*|ASSET_NAME|*` contains the asset name. Every asset always has a name `*|ASSET_EXTERNAL_REFERENCE|*` contains an external reference of the asset. This is typically used when the asset has an ID other than TicTAP's and it is useful to store it with integration purposes.

Assets may also have a set of custom fields that can be configured depending the the Team you're working. When you need to show that field information for your asset, it is necessary to prepend "ASSET" to the custom field shown in SCREAMING_CASE :

`*|ASSET_MY_CUSTOM_FIELD|*`

Custom fields variables

Custom fields can be configured to be used in Tags, Apps and Assets (or Entities). Depending on the origin of that custom field, you need to add the prefix that indicates the location where that custom field resides.

For instance, we can configure a custom field, called "My custom field", which can be referenced:

`*|TAG_MY_CUSTOM_FIELD|*` `*|APP_MY_CUSTOM_FIELD|*` `*|ASSET_MY_CUSTOM_FIELD|*` , depending on whether the field has been placed as part of a TAG, an App or an ASSET.

Fallback : default value

It is possible to specify a default value if the VARIABLE we are referring to has no value.

So, for example, you can specify:

```
*|TAG_VARIABLE||TAG_FALLBACK_VARIABLE|*
```

Which means: when the TAG_VARIABLE has no value, then use the value of TAG_FALLBACK_VARIABLE

Maintenance plans variables

You can refer to a specific planning using it's `planning-slug`

```
*|asset.plannings.planning-slug.next-date:date('d/m/Y')|*
*|asset.plannings.planning-slug.last-date:date('d/m/Y')|*
*|asset.plannings.planning-slug.status|*
```

Or you can just access a planning by the order in the list of available plannings for this asset. For instance, if you want to access the first planning for this asset, just do:

```
*|asset.plannings[0].next-date:date('d/m/Y')|*
*|asset.plannings[0].last-date:date('d/m/Y')|*
*|asset.plannings[0].status|*
```

Filters

You can also use filters on your variables to transform the value of a variable. These are the currently supported filters:

url

The url filter allows to extract a URL for a file/image present in a variable

```
*|TAG_FOTO:url('page_card')|*
```

The optional parameter specifies the "format" of that media, which refers to the "dimensions" of the given image. Currently available there are: "page_card" and "page_thumbnail"

localize

Uses the "current locale" to conditionally show a text.

```
*|FILTER:localize({"en":"English","pt":"Portuguese"})|*
```

date

A filter of type "date" allows you to format a variable containing a date in different ways:

```
*|TAG_DATE:date('Y')|*   *|ASSET_LAST_MAINTENANCE_DATE:date('Y')|*
*|ASSET_LAST_MAINTENANCE_DATE:date('m-Y')|*
```

You can find all available formats in Date Formats

Functions

IF / ELSE

You can use the IF() function to control the information displayed based on a variable

Here are some examples of the use of the IF :

```
{{ IF(*|ASSET_META|*, "A2 was true", "A2 was false" ) }}
```

```
{{ IF(*|ASSET_META|*, *|ASSET_META|*,"A2 was false") }}
```

```
{{ IF(*|ASSET_META|*, *|ASSET_META|*, *|ASSET_ANOTHER_META|*) }}
```

```
{{ IF(*|ASSET_META|* = "foo", "<div>Your HTML HERE!!</div>" ) }}
```

ASSET_URL

You can use ASSET_URL function to get a direct link to an asset, even if it does not have a tag sticked to it. It requires the id of the asset as only argument:

```
{{ASSET_URL(*|asset.id|*)}}
```