

KMS specifications

version 0.1

scrockman@gmail.com

XML structure

NOTE: the tags between *(* and *)* are optional.

<kmsRoot version>

Stores the XML document.

version

Stores the version of the XML structure.

Used by the engine to identify the version of the XML structure.

<texts defaultLanguage>

Stores the texts for all the services

defaultLanguage

Stores the default language used if no one has been specified via code.

Used by the engine to know the default language if no one has been specified via code.

<text id>

Stores a text.

id

Stores the unique identifier of the text.

Used by the engine to identify the text.

<value language value>

Stores the value of the text.

language

Stores the language of the text.

Used by the engine to identify the language of the text.

value

Stores the value of the text.

Used by the engine to identify the value of the text.

<service s>

Stores the collection of the services

<service id (name) (icon) (description) (version) (author) (enabled)>

Stores the definition of a service.

id

Stores the unique identifier of the service.

Used by the engine to identify the service.

(name)

Stores the name of the service.

Used by the interface to show to the user the name of the service.

(icon)

Stores the icon of the service.

Used by the interface to show to the user the icon of the service.

(description)

Stores the description of the service.

Used by the interface to show to the user the description of the service.

(version)

Stores the version of the service.

Used by the update manager to evaluate if a service should be updated.

(author)

Stores the name of the author of the service.

Used by the interface to show to the user the name of the author of the service.

(enabled)

Stores the enable status of the service

Used by the engine to know if a service is currently active.

<text s>
See below.

<inputs>
Stores the collection of the inputs.

<input id (content) (type) (regex) (name) (defaultValue)>
Stores the definition of an input.

id
Stores the unique identifier of the input.
Used by the engine to identify uniquely the input.

(content)
Stores the content of the input ('string', 'password', 'text', 'collection', 'captcha').
Used by the engine to identify the content of the input.

(type)
Stores the type of the input ('constant', 'variable', 'contextual').
Used by the engine to identify the type of the input.

(regex)
Stores the regular expression of the input.
Used by the engine to allow the insertion of a value that matches the specified pattern.

(name)
Stores the name of the input.
Used by the interface to show to the user the name of the input.

(defaultValue)
Stores the predefined value of the input.
Used by the interface to show to the user the input.

<outputs>
Stores the collection of the outputs.

<output id content source>
Stores the definition of an output.

id
Stores the unique identifier of the output.
Used by the engine to identify uniquely the output.

content
Stores the content of the input ('string', 'text', 'html', 'image').
Used by the engine to identify the content shown by the output.

source
Stores the source of the output.
Used by the engine to identify the data of the output.

<actions>
Stores the collection of the actions

<action id type>
Stores the definition of an action.

id
Stores the unique identifier of the action.
Used by the engine to identify the action.

type (= \$type)
Stores the type of the action ('HttpRequest', 'checkHttpRequest', 'askCaptchaCode').
Used by the engine to identify the type of the action.

<\$type id>
Stores the type of the action.
Used by the engine to perform the action of the specified type.

<sequences>
Stores the sequences.

<sequence id>
Stores a sequence.

id
Stores the unique identified of the sequence.
Used by the engine to identify the sequence.

<usedInputs>

Stores the collection of the used inputs.

<usedInput input>

Stores an used input.

input

Stores the input anchor.

Used by the engine to identify the used input.

<usedActions>

Stores the collection of the used actions.

<usedAction action>

Stores an used action.

action

Stores the action anchor.

Used by the engine to identify the used action.

Action types

<action type="httpRequest">

<httpRequest id>

Does the http request.

id

Stores the unique identified of the http request.

Used by the engine to identify the http request.

The body is the http request.

<action type="checkHttpRequest">

<checkHttpRequest httpRequest (containsRegex) (onTrueOutput) (onFalseOutput) (onTrue) (onFalse)>

Checks the http request.

httpRequest

Stores the data of the http request.

Used by the engine to catch the data of the http request.

containsRegex

Stores the regular expression that will be evaluated.

Used by the engine to evaluate the specified regular expression.

onTrueOutput

Stores the output shown if the containsRegex is true.

Used by the engine to identify the output.

onFalseOutput

Stores the output shown if the containsRegex is false.

Used by the engine to identify the output.

onTrue

Stores the internal action performed if the containsRegex is true ('continue', 'break').

Used by the engine to identify the internal action.

onFalse

Stores the internal action performed if the containsRegex is false ('continue', 'break').

Used by the engine to identify the internal action.

<action type="askCaptchaCode">

<askCaptchaCode httpRequest input>

Asks to the user to insert the captcha code using input from the data caught by httpRequest.

httpRequest

Stores the data of the captcha image.

Used by the engine to catch the data of the captcha image.

input

Stores the input used by the captcha.

Used by the engine to identify the input used by the captcha.