

TYPO3 Core - Feature #90133

Extbase: configurable 404 handling for missing action parameters

2020-01-16 13:48 - Christian EBI

Status:	New	Start date:	2020-01-16
Priority:	Should have	Due date:	
Assignee:		% Done:	0%
Category:	Extbase	Estimated time:	0.00 hour
Target version:		Complexity:	
PHP Version:		Sprint Focus:	
Tags:			

Description

There are already two typoscript settings for determining the behaviour if an action can't be resolved:

```
mvc.callDefaultActionIfActionCantBeResolved = ...
mvc.throwPageNotFoundExceptionIfActionCantBeResolved = ...
```

These are taken into account if a called action is not allowed by the plugin. (for more info see <https://docs.typo3.org/m/typo3/book-extbasefluid/master/en-us/b-ExtbaseReference/Index.html>)

It also happens all the time that an extbase action is called without all the required parameters. This will throw an uncaught `RequiredArgumentMissingException` which usually looks like this:

```
#1298012500: Required argument "foo" is not set for Cyberhouse\Theme\Controller\BarController->list.
```

There are many cases where you usually want to handle such a case as an 404 error. To achieve that, you would need to override the default behaviour of the extbase controller, like override the `processRequest()`, catching the exception and doing the 404 handling in there.

For easier management I propose two new typoscript similar to the ones described above. This makes it possible to either globally configure 404 handling for missing parameters (config.tx_extbase namespace) or per extension/plugin via typoscript.

Example configuration:

```
mvc.callDefaultActionIfRequiredArgumentsAreMissing = 0
mvc.throwPageNotFoundExceptionIfRequiredArgumentsAreMissing = 1
```

History

#1 - 2020-03-10 17:27 - Stephan Bauer

Yes please