Markdown Manual Version 1.01

1. Functions

The translation of markdown symbols to html-tags is done in the function parseMarkdown. The function formatHtml adds <pre> tags this is needed to keep line breaks and tabs as they are otherwise removed.

1.1 General description of parseMarkdown()

The parseMarkdown functions uses the javascript function replace to identify valid markdown symbols. Regular expressions are used to identify valid sequences of markdown symbols.

1.2 Regular expressions

Regular expressions can be used to represent a series of characters, character outside of parentheses are removed by the replace function. Characters within parentheses are called the capturing group and are saved in a variable ($1).

1.2.1 Italics and bold

Expression	Description	Example
/*{1}(\S.*?\S)*{1}/g	One asterisk before and one asterisk after any characters except line-break. If there is a space after the leading asterisk or before the trailing asterisk the expression will be ignored.	*italics* *italics italics*
/_{1}(.*?\S)_{1}/g	One underscore before and one after any characters except line-break. If there is a space after the leading underscore or before the trailing underscore the expression will be ignored.	_italics_ _italics italics_
/*{2}(\S.*?\S)*{2}/g	Two asterisks before and two after any characters except line-break. If there is a space after the leading asterisk or before the trailing asterisk the expression will be ignored.	**bold** **bold bold**
/_{2}(\S.*?\S)_{2}/g	Two underscores before and two after any characters except line-break. If there is a space after the leading underscore or before the trailing underscore the expression will be ignored.	__bold__ __bold bold__
/*{3}(\S.*?\S)*{3}/g	Three asterisks before and three after any characters except line-break. If there is a space after the leading asterisk or before the trailing asterisk the expression will be ignored.	***italicsBold*** ***italics bold***
/_{3}(.*?\S)_{3}/g	Three underscores before and three after any characters except line-break. If there is a space after the leading	___italicsBold___ ___italics bold___

	underscore or before the trailing underscore the expression will be ignored.	

Figure 1.1 Full expression for italics and bold

Regex symbols	Description
*{x}	Asterisk repeated x times, the asterisk will be removed if the whole expression is valid.
_{x}	Underscore repeated x times, the underscores will be removed if the whole expression is valid.
(\S.*?\S)	This is the capturing group, characters matching this regex will be saved in a variable ($1). The capturing group start with any non-whitespace character (\S), followed by any character that is not a new-line (.*?), and the last character must be a non-whitespace character (\S).
g	Match all cases of the expression in the string instead of returning after the first case is found.

Figure 1.2 Description of parts in expression used for italics and bold

1.2.2 Headings

Expression	Description	Example
/^\#{x}\s(.*)/gm	If a line start with x hashtags followed by a whitespace character and then any characters except linebreak.	### Heading 3 ## Heading 2 # Heading 1

Figure 1.3 Full expression for headings

Regex symbols	Description
^	The expression has to be written at the start of a string.
#{x}\s	Hashtags repeated x times, followed by a whitespace character
(.*)	Any character except a newline character, these symbols are the capturing group and will be save in the $1 variable.
gm	g = match all cases of the expression in the string. m = changes the '^' symbol to match start of a line instead of start of string.

Figure 1.4 Description of parts in expression used for italics and bold

1.2.3 Lists

Expression	Description	Example
/^\s*\d*\.\s(.*)/gm	If a line start with zero or more whitespace characters followed by any number, followed by a dot, and any whitespace character, followed by any non-line break characters, the expression is valid	1. list entry 1 2. list entry 2
/^\s*\-\s(.*)/gm	If a line start with zero or more whitespace characters followed by a dash, followed by any whitespace character, followed by any non-line break characters, the expression is valid	- list entry 1 - list entry 2

Figure 1.5 Full expression for lists

Regex symbols	Description
^	The expression has to be written at the start of a string.
\s*	Any whitespace character repeated zero or more times.
\-\s	A dash followed by any whitespace character
(.*)	Any non-line break character repeated zero or more times.
gm	g = match all cases of the expression in the string. m = changes the '^' symbol to match start of a line instead of start of string.

Figure 1.6 Description of parts in expression used for lists

A list should have the tag for list type (<ul>/<ol>) before any entry and after the last entry, but the parser adds them before and after each entry.
example:
      * List entry 1 parsed as List entry 1
      * List entry 2 parsed as List entry 2
To fix this a second replace is executed after each list that removes all extra tags and the result is as below.
      * List entry 1 parsed as List entry 1
      * List entry 2 parsed as List entry 2

1.2.4 Line and codeblock

Expression	Description	Example
/^(\-{3}\n)/gm	If a line start with three dashes followed by a line break character	---
/~{3}((?:\r|\n|.)+?)\~{3}/g	Three tildes followed by any characters, ending with three tildes.	~~~ Code block ~~~ ~~~ Code block ~~~

Figure 1.7 Full expression for hr-line and code blocks

Regex symbols	Description
^	The expression has to be written at the start of a string.
\-{3}\n	Three dashes followed by a line break.
~{3}	Three tildes
(?:\r|\n|.)	Non-capturing group used to match any character (really not sure how this works).
+?	+ = Repeated one or more times. ? = lazy.
gm	g = match all cases of the expression in the string. m = changes the '^' symbol to match start of a line instead of start of string.

Figure 1.8 Description of parts in expression used for code blocks

Change list

2015-05-04 version 1.01
Removed the suggestion part.
Added description of implemented markdown.

2015-04-02 version 1.00
Created document.