TagThis User Manual

http://oinam.com/wordpress/plugins/tag-this/

For Version 0.9.1
Manual Version: 0.1.3

Abstract

TagThis is a wordpress plugin that allows the community to appropriately tag your posts. It does so by adding a small textbox below a post. The most popular tags are then added automatically to your post, and helps organize your blog. It helps you tag your old posts, or get a large database of relevant tags, and better interaction with the community.

Contents

1   Credits:

TagThis was an idea conceived by Brajeshwar (brajeshwar -at- brajeshwar -dot- com) and written in whole by Anirudh(anirudh -at- anirudhsanjeev -dot- org), and released under GPLv3 License. Small parts of the code are used from Lester Chan's wp-ratings plugin(under provision of GPLv2), and it uses the OneClick framework, under LGPL2 license. The project is hosted on Google Code, and the latest version of the code can be taken from SVN at http://wp-tagthis.googlecode.com. The manual is published using Latex.

2   Requirements:

TagThis requires Wordpress 2.3 or higher. There is no possibility of it working with any previous versions. Support for older versions with Ultimate Tag Warrior may be released, but wordpress 2.3 is required.

3   Installation:

Upload the tagthis directory to your wp-content/plugins/ directory, and activate it from the plugin menu.

4   Enabling your theme to use TagThis!:

To use TagThis!, you need to manually edit your theme, which takes under five minutes. The reason this can't be done automatically as wordpress cannot guess where your loops are. The inherent flexibility and freedom it provides prevents wordpress from deciding where to put TagThis!

You'll typically need to open up three files using notepad/wordpad/gedit/emacs/or your favorite text editor. You'll need to insert one line:

```
if(function_exists('wp_tagthis')){wp_tagthis();}
```

4.1   index.php: For your front page

We suggest putting your TagThis widget after every post in the loop. look for something like this:

```
    the_content('Continue Reading');
```

Change it to:

```
    the_content('Continue Reading');
    if(function_exists('wp_tagthis')){wp_tagthis();}
```

Essentially, calling the PHP function

```
"wp_tagthis()"
```

will display the widget for the current displayed post.

4.2   single.php: for a single post

We suggest putting your TagThis widget at the end of the post look for something like this:

```
the_content('Continue Reading');
```

Change it to:

```
the_content('Continue Reading');
if(function_exists('wp_tagthis')){wp_tagthis();}
```

Essentially, calling the PHP function

```
"wp_tagthis()"
```

will display the widget for the current displayed post.

4.3   page.php: for a single page

We suggest putting your TagThis widget at the end of the page look for something like this:

```
the_content('Continue Reading');
```

Change it to:

```
the_content('Continue Reading');
if(function_exists('wp_tagthis')){wp_tagthis();}
```

Essentially, calling the PHP function

```
"wp_tagthis()"
```

will display the widget for the current displayed post.

4.4   Getting it customized by a developer

You can get the TagThis! widget customized to your theme, and as per your liking by the developer of the plugin itself. This way you can get it working without touching the code, and supporting development of the plugin in the process. Visit http://wpstudio.org/tagthis/ for more details.

5 TagThis! Explained:

5.1 Basics:

It works this way.

1. Visitor read article

2. Visitor tags article with a relevant tag

3. This gets registered in an internal database

4. If quite a few people are tagging the same article with the same tag, then it must mean it's relevant.

Thus, visitor's tag gets added as a real wordpress tag, and works with tag clouds, organization, searching, SEO and all other benefits of tagging your article with relevant tags.

5.2 TagThis Management Page:

On your WordPress admin panel, there'll be a new top level menu called "TagThis". Clicking on it brings up the management page. This page is self explanatory showing who(which IP) tagged which post with what tag. It also shows the tags detected as spam, and allows you to moderate the tags, and flag some as spam.

5.3 On TagThis and spam:

TagThis follows three steps to prevent spam whatsoever:

1. The preliminary PHP/AJAX security prevents people from brute forcing several hundred tags at once on your TagThis install.

2. Even if someone tries to add a tag several times, more people will have to tag it the same way, and you can decide how many. (try tagging something yourself several times and see how it works)

3. Even if a spammer gets several of his/her buddies to tag a post with the same word, you can have a blacklist of words and IPs, so it'll detect them as spam before they even try to touch the database.

5.4 On TagThis! upgrades:

TagThis! upgrades itself. If you want it to, it downloads the latest zip file of itself, and unzips itself, giving you the latest and greatest in community tagging. Kindof bears a haunting similarity to OneClick for wordpress (prolly cause the same bloke wrote it).

6 TagThis Options:

Access the options page when the tagthis section is enabled, in the admin dashboard of your wordpress blog.

6.1 Number of tags for frontpaging:

This is sort of like digg. The number of people that must tag a post for it to be added to the official wordpress database. Suggested values are 2-5, though you might want to keep it higher if you run a high traffic blog.

6.2 Secret Word:

This isn't a password, just a string used for encrypting the TagThis key. Just don't tell any hackers this string and set it to anything under 5 characters.

6.3 Display TagThis Option:

This enables or displays the widget, textbox, ajax, and all. Highly recommended as the plugin is useless with this off.

6.4 Display Tags of the Post:

The TagThis! widget can display the current wordpress tags for your post. If your theme doesn't display the tags, you might want to use this. If it's being displayed elsewhere, you can disable this.

6.5 Manually Moderate:

For the clinically paranoid, this requires you to moderate every tag that your readers supply. Not recommended.

6.6 Custom CSS:

If you want the widget to follow different CSS rules, edit this box.

6.7 Spam Filter:

You can access the spam blacklist. Add one word for each line for the word blacklist. Spam matching is case insensitive and follows the SQL "Like" rule. So if "abc" is blacklisted, "fabcd" is also marked as spam. Also URLs are marked as spam (so people won't even try). You can also add IPs to manual blacklisting. If you mark a tag as spam, it's automatically added to the spam filter. If extra strength spam filter is enabled, you will also blacklist the IP along with the term.

6.8   Tips and Tricks:

Separating tags by commas will add multiple tags to the same post. Encouraging users to tag your posts will build stronger bond between the author and community, as the users feel like they own a part of the content (emotionally, not legally)

7   Support:

Please use the support forums (http://forum.oinam.com/viewforum.php?id=43) for help, and to report bugs, go to http://code.google.com/p/wp-tagthis/issues/list and open up an issue. Bugs can also be sent to anirudh -at- anirudhsanjeev -dot- org. Localization project will begin after the first stable release. The plugin has been written 99% with GNU GetText, so feel free to translate.

8   About:

Brajeshwar is the head/grand master of Oinam. [this description is a stub] Anirudh, with Odin on his side, is a 20 year old physics major at IIT. Metalhead. Loves coffee and intelligent conversation. Runs, sortof plays the bass, and writes code like poetic swordplay.