Разработка плагинов

Написать плагин DokuWiki очень просто. Перед тем, как написать свой собственный, не забудьте поискать похожие плагины среди уже написанных and read through рекомендации по безопасности для авторов плагинов ..

DokuWiki имеет гибкий API-интерфейс плагина, который позволяет разработчикам расширять его функциональность, будь то дополнительные режимы синтаксиса , пользовательские режимы действий или новые форматы экспорта и многое другое. В зависимости от назначения плагина, он может состоять из одного или нескольких компонентов следующих типов плагинов.

Типы плагинов

На данный момент DokuWiki предлагает семь различных типов плагинов.

- Syntax Plugins extend DokuWiki's basic syntax.
- Action Plugins can be used to extend or replace many aspects of DokuWiki's core operations, from saving wikipages to adding new action modes.
- Admin Plugins can provide administration functionality for DokuWiki - these plugins are accessible to superusers and managers via the Admin button.
- Helper Plugins can be used to provide functionality to many other plugins, so each plugin doesn't have to re-implement a certain function over and over again.
- Renderer Plugins allow to create new export modes and to replace the standard DokuWiki XHTML renderer.
- Remote Plugins allow to add webservices to your plugin.
- Auth Plugins add additional authentication backend to the DokuWiki, which could be activated via configuration manager.
- CLI Plugins add command line tools to your plugin

Plugin types are not mutually exclusive, i.e. a plugin can also be assigned to several plugin types.

Examples: A simple plugin like color plugin may only consist of a single file, `syntax.php`. More advanced plugins consist of several components, for example the structured data plugin have the syntax handling split in «data entry» and «table view» components, an action component for the special edit button attached to every data entry and an admin interface for database maintenance.

Sources

There are a number of sources especially for plugin development:

- The plugin file structure gives overview of the naming convention and structure, and where you add JavaScript and CSS stylesheets, relevant for all plugin types
- There are some common plugin functions, available in all plugins types. Adds localisation, configuration and interspection.
- These security guidelines give some deeper understanding of common issues and presents

some protection measures.
- Some general plugin programming tips are available, mostly good practices and pointers to handsome tools in DokuWiki.

Plugin Name

A valid plugin name:

- Should only contain the characters a-z and 0-9.
- Underscore is NOT allowed as:
  - This is used to separate `<plugin name>` from `<component name>`.
  - Using underscore will also give a popularity rating of zero.
- If the same name is used by two different plugins
  - they are mutually excluding and inherently incompatible,
  - furthermore only one of them can have a plugin homepage on dokuwiki.org.

Plugin Wizard

A Wizard to create the basic skeleton for a DokuWiki plugin can be found at http://pluginwizard.dokuwiki.org/ and is recommended to be used for starting the development. Alternatively use the command line dev Plugin.

If you later need to extend your plugin refer to the plugin file structure on how files are arranged in a plugin.

Publishing a Plugin on dokuwiki.org

If you created a plugin, please share it with the community. Just create a page named after your plugin in the plugin namespace. E.g. if your pluginfolder is named `sample` create a page `plugin:sample` here in the wiki.

The page should contain all needed documentation on how to install and use the plugin and give users a pretty good idea what it does before installing the plugin. Adding screenshots might be a good idea as well. When creating the new page, a namespace template will prefill sections and tips for you.

At the top of the plugin page a few metadata fields have to be filled. A description of each field can be found on Repository Plugin page.

Users will see that an update is available in the extension manager. For this to work properly it is necessary that the date “Last updated on” at the Plugin wiki page equals the date in the file plugin.info.txt in the source tarball/zipfile. If this is not the case, available updates will not be shown or the “Update” signal will persist despite the plugin being up-to-date!

Uploads are not allowed on dokuwiki.org, so you need to host your plugin files somewhere else. We recommend to manage your source with a Revision Control System like git. If you do, it's easiest to

use a public repository host like GitHub, GitLab or Bitbucket which offers also a bug tracker for your repository.

Plugin developers can check the bad extension page for differences in their github repository checkin, the plugin.info.txt and the respective page at dokuwiki.org.

Inspirations

Existing Plugins

Already more than 1400 plugins are written for DokuWiki. To learn how things are working, you might study some existing plugins that have some similarity with your ideas. At codesearch.dokuwiki.org you can find all them in a browsable format.

Event List

Another way is to search the event list for a suitable event that may be used to implement the desired plugin function. The plugins that use this event can then be found using the code search mentioned above.

At the bottom of each event page, where the details of the event are listed, there is a link to a pre-filled code search. It provides specific examples of the function and application of the respective event.