

# User HomePage Plugin

— plugin — description: Creates users' private namespace and/or public page and redirects them to their own private namespace on login. The pages are base on customizable templates. The current version added the public page feature, a helper and many ACL settings to cover common or uncommon usage. author : Simon Delage email : [simon.geekitude@gmail.com](mailto:simon.geekitude@gmail.com) type : Action lastupdate : 2015-05-12 compatible : Ponder Stibbons,Hrun depends : conflicts : similar : tags : user, redirect, userpage, private, namespace, public, homepages

downloadurl: <https://github.com/geekitude/dokuwiki-plugin-userhomepage/archive/master.zip>

bugtracker : <https://github.com/geekitude/dokuwiki-plugin-userhomepage/issues> sourcerepo : <https://github.com/geekitude/dokuwiki-plugin-userhomepage/>

screenshot\_img : <http://www.geekitude.fr/externe/dokuwiki/userhomepage/screenshot.jpg>

---

## Installation

Install the plugin using the [Plugin Manager](#) and the download URL above, which points to latest version of the plugin. Refer to [Plugins](#) on how to install plugins manually.



**Important:** 2015-02-01's version introduced a crucial change in ACL process to reduce the amount of IO operations on *acl.auth.php* file and hopefully get rid of a bug that corrupted it from time to time. Please update !

## Upgrading from version 3.0.4




Even if keeping backward compatibility has been a major concern while developing this new version, it is strongly recommended to backup your Dokuwiki (at least conf folder and data related to users' public pages or private namespaces).


Some options were kept only for backward compatibility but pay extra care to all settings concerning private namespace to make sure you get the same file organisation than with previous version. A new option [Create Private Namespace] which is disabled by default will prevent any page creation just after the installation of the new version. **Before enabling that option**, backup your ACL file (*conf/acl.auth.php*) and then remove ACL rules linked to the old version of the plugin. Note that the old template file is no longer bundled so it can't be overwritten by the new package



If you prefer to stick to old 3.0.4 version, [here's a package](#) with only a few fixes to enable saving settings (which didn't work in the most recent package available) and a minor change on ACL to actually protect the created user namespaces.

## Settings

- **create\_private\_ns** : enable creation of users' private namespaces [default: disabled (to make sure no private namespace is created before all settings are set the right way)]
- **use\_name\_string** : use user's full name instead of his login for the name of his private namespace (ie: user:simon\_delage instead of user:simon) [default: disabled]
- **use\_start\_page** use same name as Dokuwiki setting start for the start page of private namespaces (otherwise the same string as the namespace name will be used) [default: enabled]
- **users\_namespace** : the namespace under wich users' private namespaces will be created [default: user]
- **group\_by\_name** : group private namespaces in alphabetecal sub-namespaces users\_namespace
- **edit\_before\_create** : on the creation of his private namespace, a user will get the oportunity to edit the start page right away unless a public page is created too (as users will be redirected to their private namespace start page on login they will not forget to edit it some day while they might forget about the public page so it has precedence) [default : disabled]
- **acl\_all\_private** : ACL for @ALL group on others' private namespaces [default : none]
- **acl\_user\_private** : ACL for @user group on others' private namespaces [default : none]
- **create\_public\_page** : enable creation of users' public pages [default: disabled (to make sure no public page is created before all settings are set the right way)]
- **public\_pages\_ns** : namespace under wich users' public pages will be created (  if you decide to change this setting, you will have to manually manage interwiki user link used with showuseras setting<sup>1)</sup>) [default : user]
- **acl\_all\_public** : ACL for @ALL group on others' public pages [default : read]
- **acl\_user\_public** : ACL for @user group on others' public pages [default : none]
- **templates\_path** : while a path inside <dokuwiki\_data>/pages will make templates editable from within Dokuwiki, a path like lib/plugins/userhomepage will prevent any access to them (wich can be seen as an advantage, depending on your vision) and might be usefull to centralize them in a Dokuwiki farm setting
- **templatepath** : only kept for backward compatibility, this setting should point to the template used in previous version and wich will be used as new private template source **if it exists** (since the file isn't bundled anymore, **leaving this setting to it's default value with a fresh install of the plugin will do no harm and have no effect**). [default : lib/plugins/userhomepage/\_template.txt or another path as set in previous version]
- **acl\_all\_templates** : ACL for @ALL group on plugin's templates if they are inside <dokuwiki\_data>/pages [default : read]
- **acl\_all\_templates** : ACL for @user group on plugin's templates if they are inside <dokuwiki\_data>/pages [default : read]
- **no\_acl** : absolutely no ACL rule will be automatically set (ACL options above will be ignored), including any protection on templates. Rules allready set will have to be manually removed. Only enable this option if you know what you're doing and have a specific need not covered by the options above. [default : disabled]

Right after saving settings, depending on selected options, you will get a success (or error  ) message about the creation of the templates:



## Templates

The plugin uses two templates (`userhomepage_private.txt` and `userhomepage_public.txt`) that **will automatically be created when needed** (ie: enabling `create_private_ns` or `create_public_page`) in the path set in `templates_path` setting (from localized source files found in `<dokuwiki_plugins>/userhomepage/lang/<language_code>` folder). They contain basic informations usefull for users.



Don't edit the localized source files as they will be overwritten by plugin updates

Templates can use any [standard template replacement pattern](#) (`@NAME@`, `@USER@`, ...) and the following specific ones:

- `@TARGETPRIVATEPAGE@` : ID of user's private namespace start page
- `@TARGETPRIVATENS@` : ID of user's private namespace
- `@TARGETPUBLICPAGE@` : ID of user's public page
- `@TARGETPUBLICNS@` : ID of the namespace containing all users' public pages

Once the templates are set, here's what will happen when a user logs in:



### Get more out of templates

- The [Include](#) plugin can be very usefull to include the wiki global start page in private namespace start page template so that the users, while redirected on login to their private page, will still have access to the global wiki start page...
- [Pagequery](#) can also add some magic to Userhomepage templates but be carefull because the requests can make the page quite slow to load :
  - `{{pagequery>.*;filter=creator:@NAME@,!ns:user;sort=ns,title}}` will show user's creations
  - `{{pagequery>.*;filter=contrib:@NAME@,!ns:user;sort=ns,title}}` will show user's contributions

## About ACL

When possible, the plugin uses wildcards to keep [ACL](#) rules as compact as possible.

Here is how they look like for a very short number of users with all settings at their default values:



Note that rules for `@user` group only apply if they're different from the equivalent setting for `@ALL` group.

There will be less ACL rules if the private namespaces are not stored in the same namespace as private page as this enables ACL wildcards on public pages:



But much more if you enable `use_name_string` or `group_by_name` options as they break ACL wildcards on private namespaces. Moreover, in that case, unlike for public pages, acl for each private namespace will only apply when the given user logs in (wich can be a problem if the pages allready exist).

## About Redirection

When a user logs in and didn't requested a specific page (or requested the Wiki start page), he will be redirected to his private namespace start page if it exists (works if Wiki start page is localized).




In facts, a timestamp is stored in PHP \$\_SESSION variable on log in (under the name uhptimestamp) and any request to get Wiki start page within 2 seconds after log in will be redirected (this is to ensures that the redirection works even if user logged in through "Remember Me" process). Let me know if the delay should be customizable in settings.

## Helper

To use the helper, use the following code:

```
<?php
$userhomepage = plugin_load('helper','userhomepage');
if ($userhomepage) echo $userhomepage-><helper_function_call>;
?>
```

Where you replace the string `<helper_function_call>` by one of the following:

- **getPrivateID()** returns the ID of user's private namespace start page (ie: *user:simon:start*)
- **getPublicID()** returns the ID of user's public page (ie: *user:simon*)
- **getPrivateLink()** returns a simple text link named "Private Space"
- **getPrivateLink("any string")** returns a simple text link named "any string"
- **getPrivateLink("loggedinas")** returns an `<li>` element like the "Logged in as" element of Dokuwiki template with a specific style 
- **getPublicLink(), getPublicLink("any string")** or **getPublicLink("loggedinas")** returns the equivalent link to public page (with another image for last one: )
- **getComplexLoggedInAs()** returns an `<li>` element like the "Logged in as" element of Dokuwiki template with a specific style and links to both private namespace and public page (if either doesn't exist, will default to relevant more simple link above or to standard string if neither exists)  (added on 2014/09/19, improved on 2014/09/20, hopefully fixed on 2015/02/03 if neither page exist)
- **getAnyPublicLink("any user login")** returns an interwiki-like link (colored according to page existence) to provided user's public page (main purpose is to replace showuseras interwiki link in case public pages are not stored in :user namespace) (added on 2014/02/23)

## Change Log

- [Added setting to replace png icons with svg](#) (2022/08/29 07:24)
- [Removed option to use user full name for public page ID](#) (2022/01/12 07:43)
- [Remove obsolete test](#) (2021/12/23 08:58)
- [Added setting to use full name for public page id](#) (2021/12/21 09:47)
- [Add helper function to replace editor/locker id with a link](#) (2021/12/20 07:59)

## Known Bugs and Issues

Please refer to the [GitHub issue tracker](#) for reporting issues.

## Discussion

Please use [this forum thread](#) or mails.

1)

by creating a `<dokuwiki_conf>/interwiki.local.conf` file to redirect that link to the correct namespace with something like `<chosen_ns> :<chosen_ns>:{NAME}`

From:

<http://wiki.geekitude.fr/> - **Geekitude**

Permanent link:

[http://wiki.geekitude.fr/en/info/doku/userhomepage\\_official](http://wiki.geekitude.fr/en/info/doku/userhomepage_official)

Last update: **2018/05/03 11:10**

