
Collection of repoze.who friendly form plugins

Release 1.0.8

Gustavo Narea

February 23, 2012

CONTENTS

1	How to install	3
2	Available form plugins	5
2.1	FriendlyFormPlugin examples	6
3	Support and development	9
3.1	Releases	9
	Python Module Index	11
	Index	13

Author Gustavo Narea.

Latest release 1.0.8

Module author: Gustavo Narea <me@gustavonarea.net>

Overview

repoze.who-friendlyform is a `repoze.who` plugin which provides a collection of developer-friendly form plugins, although for the time being such a collection has only one item.

HOW TO INSTALL

The minimum requirement is `repoze.who`, and you can install both with `easy_install`:

```
easy_install repoze.who-friendlyform
```


AVAILABLE FORM PLUGINS

```
class repoze.who.plugins.friendlyform.FriendlyFormPlugin(login_form_url,
                                                         lo-
                                                         gin_handler_path,
                                                         post_login_url,
                                                         lo-
                                                         gout_handler_path,
                                                         post_logout_url,
                                                         re-
                                                         mem-
                                                         berer_name,
                                                         lo-
                                                         gin_counter_name=None,
                                                         charset='iso-
                                                         8859-
                                                         1')
```

RedirectingFormPlugin-like form plugin with more features.

It is like `RedirectingFormPlugin`, but provides us with the following features:

- Users are not challenged on logout, unless the referrer URL is a private one (but that's up to the application).
- Developers may define post-login and/or post-logout pages.
- In the login URL, the amount of failed logins is available in the environ. It's also increased by one on every login try. This counter will allow developers not using a post-login page to handle logins that fail/succeed.

You should keep in mind that if you're using a post-login or a post-logout page, that page will receive the referrer URL as a query string variable whose name is "came_from".

Forms can be submitted with any encoding (non-ASCII credentials are supported) and ISO-8859-1 (aka "Latin-1") is the default one.

```
__init__(login_form_url, login_handler_path, post_login_url, lo-
         gout_handler_path, post_logout_url, rememberer_name, lo-
         gin_counter_name=None, charset='iso-8859-1')
```

Parameters

- **login_form_url** (*str*) – The URL/path where the login form is located.
- **login_handler_path** (*str*) – The URL/path where the login form is submitted to (where it is processed by this plugin).
- **post_login_url** (*str*) – The URL/path where the user should be redirected to after login (even if wrong credentials were provided).
- **logout_handler_path** (*str*) – The URL/path where the user is logged out.
- **post_logout_url** (*str*) – The URL/path where the user should be redirected to after logout.
- **rememberer_name** (*str*) – The name of the repoze.who identifier which acts as rememberer.
- **login_counter_name** (*str*) – The name of the query string variable which will represent the login counter.
- **charset** (*str*) – The character encoding to be assumed when the user agent does not submit the form with an explicit charset.

The login counter variable's name will be set to `__logins` if `login_counter_name` equals `None`. Changed in version 1.0.1: Added the `charset` argument.

2.1 FriendlyFormPlugin examples

When you use this plugin, you would implement a controller action for logins, like this one:

You have to adapt this function to the way things work in your framework:

```
def login(request):
    login_counter = request.environ['repoze.who.logins']
    if login_counter > 0:
        display_message("Wrong credentials", status="error")
        came_from = request.params.get("came_from") or "/"
    return render("login.html", login_counter=login_counter, came_from=came_from)
```

Where the “login.html” template is defined as:

```
<!-- Adapt this code to your templating engine -->
<form action="/login_handler?came_from={{ came_from }}&__logins={{ login_cou
    method="POST">
    <label>Username: <input type="text" name="login"/></label>
    <label>Password: <input type="password" name="password"/></label>
    <input type="submit" value="Login"/>
</form>
```

2.1.1 Post-login action

A controller action for post-logins could look like:

You have to adapt this function to the way things work in your framework:

```
def welcome_back(request):
    identity = request.environ.get("repoze.who.identity")
    came_from = request.params.get('came_from', '') or "/"

    if identity:
        # Login succeeded
        userid = identity['repoze.who.userid']
        display_message('Welcome back, %s!' % userid, status="success")
        destination = came_from
    else:
        # Login failed
        login_counter = request.environ['repoze.who.logins'] + 1
        destination = "/login?came_from=%s&__logins=%s" % (came_from, login_counter)

    return Redirect(destination)
```

2.1.2 Post-logout action

A controller action for post-logouts could look like:

You have to adapt this function to the way things work in your framework:

```
def see_you_later(request):
    display_message("We hope to see you soon!", status="success")
    came_from = request.params.get('came_from', '') or "/"
    return Redirect(came_from)
```


SUPPORT AND DEVELOPMENT

The preferred place to ask questions is the [Repoze mailing list](#) or the [#repoze](#) IRC channel. Bugs reports and feature requests should be sent to [the issue tracker of the Repoze project](#).

The development mainline is available at the following Subversion repository:

<http://svn.repoze.org/whoplugins/whofriendlyforms/trunk/>

3.1 Releases

3.1.1 repoze.who-friendlyform releases

This document describes the releases of `repoze.who.plugins.friendlyform`.

Version 1.0.6 (2010-04-28)

- Restricted the plugin to identify and challenge to browser requests only. So it won't come into play if the client is a Subversion client, for example.

Version 1.0.5 (2010-04-22)

- Made the plugin return `str` instead of `unicode` when the request is made using ASCII.
- Added sample code to the documentation on how to use `FriendlyFormPlugin`, as suggested by Ian Stevens.

Version 1.0.1 (2010-01-29)

Added the ability to set another default character encoding for the credentials in `FriendlyFormPlugin`.

Although ISO-8859-1 (aka "Latin-1") is the official default charset according to the HTTP 1.1 specification (the one to be used when no encoding is explicitly mentioned in the request), browsers (e.g., Firefox) don't set it even when they use other encodings like UTF-8.

So, if you use a charset other than Latin-1, you should let the plugin know.

This has been reported by Christoph Zwerschke.

Version 1.0 Final (2010-01-28)

Christoph Zwerschke made the `FriendlyFormPlugin` support non-ASCII credentials. Thank you Christoph!

It made no sense to do a release candidate because this has been the only bug over the last year. Hence the “final” release.

Version 1.0b3 (2009-03-02)

- Specified the required version of `repoze.who`, otherwise the buggy `setuptools` won't install it.

Version 1.0b2 (2009-02-20)

- Forced the login counter name in the query string to be `'__logins'` even when `login_counter_name` is passed as `None` to `repoze.who.plugins.friendlyform.FriendlyFormPlugin.__init__()`. The previous behavior was causing some weird problems on TG2 applications.

Version 1.0b1 (2009-02-17)

This is the first release of **repoze.who-friendlyform** as an independent project. The initial form plugin, `repoze.who.plugins.friendlyform.FriendlyFormPlugin`, has been moved from `repoze.what.plugins.quickstart.FriendlyRedirectingFormPlugin`.

This new version of `FriendlyRedirectingFormPlugin` doesn't extend `RedirectingFormPlugin` anymore. Instead, the relevant bits from the `RedirectingFormPlugin` have been copied over, as recommended by Chris McDonough.

This new version of `FriendlyRedirectingFormPlugin` behaves exactly as the original one.

PYTHON MODULE INDEX

r

`repoze.who.plugins.friendlyform,`

[1](#)

INDEX

Symbols

`__init__()` (re-
poze.who.plugins.friendlyform.FriendlyFormPlugin
method), [5](#)

F

FriendlyFormPlugin (class in re-
poze.who.plugins.friendlyform),
[5](#)

R

repoze.who.plugins.friendlyform (module), [1](#)