Feedback

This page documents the method used starting with Tiki16 to provide feedback to users on the page, for example to indicate an action was successfully completed, to give a warning, a note, or indicate an error has occurred. Previously a variety of methods have been used, including use of ErrorReportLib for some errors. Starting with Tiki16, the Feedback class is used (at lib/core/Feedback.php ) to centralize and standardize the coding used for providing feedback to the user.

Table of contents

- Server side
  - Adding Feedback
    - Feedback types
    - Feedback options
    - Display options
    - Feedback through ajax
  - Retrieving and displaying feedback (non-ajax)
- Client side
- Areas where Feedback has not been implemented

Server side

Adding Feedback

Typically, all that a developer needs to do is add a command in the PHP file where the feedback is generated. For example:

```php
Feedback::success(tr('Your action was successful.'));
```

The feedback will then be automatically be displayed at the top of the page when displayed.

Feedback types

Below or the standard feedback types available along with default settings:

<table>
<thead>
<tr>
<th>Type</th>
<th>Title</th>
<th>Icon</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>Error</td>
<td>☢</td>
</tr>
<tr>
<td>note</td>
<td>Note</td>
<td>☢</td>
</tr>
<tr>
<td>success</td>
<td>Success</td>
<td>✔</td>
</tr>
<tr>
<td>warning</td>
<td>Warning</td>
<td>⚠</td>
</tr>
</tbody>
</table>

Each one of these is a function (e.g., `Feedback::error($feedback)`). The generic function `add` can also be used if customization is needed. The `{remarksbox}` smarty function is used to add the alert formatting, icons and colors.
Feedback options

The above functions require at least a string message as a parameter, but an array can also be used to provide multiple messages and other parameters. If an array is used, the message or messages must be in the array key `mes`. Other parameters that can be set in the array are `type`, `title`, and `icon` (each of these being the array key that must be used).

The `tpl` option can be set to specify a Smarty template other than the default one (at `templates/feedback/default.tpl`). The value used for this array key needs to be the same as the name of the template file in the feedback directory. For example, `tpl => 'pref'`, will cause `templates/feedback/pref.tpl` to be used.

Display options

A second parameter with the value `session` can also be added if the feedback needs to be added through the PHP global `$SESSION` variable instead of through a Smarty template variable. This may be necessary if the page is redirected from the file where the feedback originated.

Feedback through ajax

Use the `send_headers` function right after adding feedback to send feedback using ajax. For example:

```php
Feedback::warning(tr('Your input may not be valid'), 'session');
Feedback::send_headers();
```

Retrieving and displaying feedback (non-ajax)

This part works automatically as follows:

- The base `layout_view.tpl` templates used to display tiki pages contain the `{feedback}` Smarty function.
- This function does the following:
  - Retrieves the feedback from the `$SESSION['tikifeedback']` PHP global variable or the Smarty `{$tikifeedback}` template variable.
  - Assigns or reassigns the feedback to the Smarty `{$tikifeedback}` variable.
    - Note that there is no need for a developer to set the `{$tikifeedback}` Smarty variable - this is done automatically in this step.
  - Fetches the feedback display template for display.

Client side

A javascript `feedback` function (in `lib/jquery_tiki/tiki-jquery.js`) can be used if a pure client-side method is needed. It has the same parameters as described above for the server side method, plus a `modal` parameter that sets whether the feedback is sent to a popup modal or is sent to the `div#tikifeedback` element that is on every page.

Areas where Feedback has not been implemented

Below are areas where this Feedback could possibly be used but where it has not yet been implemented, usually because of additional complexities.

- `$SESSION[tiki_importer_errors]`
- $_SESSION[tiki_importer_feedback]
- $_SESSION[tiki_importer_log]
- footer.tpl
- module errors
- messu-broadcast.php
- tiki-accounting*.php
- tiki-share.php
- $page_badchars display in a few templates (eg, tiki-copypage.tpl and tiki-renametpage.tpl)