Zend_View_Helper_Gravatar - Marcin Morawski

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

Zend Framework: Zend_View_Helper_Gravatar Component Proposal

Proposed Component Name	Zend_View_Helper_Gravatar
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_View_Helper_Gravatar
Proposers	Marcin Morawski
Zend Liaison	TBD
Revision	1.0 - 1 January 2008: Initial Draft. (wiki revision: 25)

Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
UC 01
UC 02
UC 03
UC 04
UC 05
9. Class Skeletons

1. Overview

This view helper receives avatars from Gravatar service.

2. References

- http://pl.gravatar.com/site/implement/url
- http://github.com/MarcinM/Zend_View_Helper_Gravatar

3. Component Requirements, Constraints, and Acceptance Criteria

- This component **will** return the different closing tag of img dependencies on DOCTYPE of websites.
- This component **will** check whether email adress is valid.

- This component **will not** check whether the default imgage option is valid url or is one of the options available(identicon, monsterid, wavatar or 404)

4. Dependencies on Other Framework Components

- Zend_View_Helper_HtmlElement
- Zend_View_Exception

5. Theory of Operation

6. Milestones / Tasks

Milestone 1: [DONE] Proposal finished
Milestone 2: [DONE] Proposal accepted
Milestone 3: [DONE] Working implementation
Milestone 4: [DONE] Unit tests
Milestone 5: [DONE] Documentation
Milestone 6: [DONE] Moved to core

7. Class Index

- Zend_View_Helper_Gravatar

8. Use Cases

9. Class Skeletons

```
<?php
/**
 * Zend Framework
 *
 * LICENSE
 *
 * This source file is subject to the new BSD license that is bundled
 * with this package in the file LICENSE.txt.
 * It is also available through the world-wide-web at this URL:
 * http://framework.zend.com/license/new-bsd
 * If you did not receive a copy of the license and are unable to
 * obtain it through the world-wide-web, please send an email
 * to license@zend.com so we can send you a copy immediately.
 *
```

```
 * @category   Zend
 * @package    Zend_View
 * @subpackage Helper
 * @copyright  Copyright (c) 2005-2009 Zend Technologies USA Inc. (http://www.zend.com)
 * @version    $Id: Doctype.php 16971 2009-07-22 18:05:45Z mikaelkael $
 * @license    http://framework.zend.com/license/new-bsd     New BSD License
 */

/** Zend_View_Helper_HtmlElement */
require_once 'Zend/View/Helper/HtmlElement.php';

/**
 * Helper for retrieving avatars from gravatar.com
 *
 * @package    Zend_View
 * @subpackage Helper
 * @copyright  Copyright (c) 2005-2009 Zend Technologies USA Inc. (http://www.zend.com)
 * @license    http://framework.zend.com/license/new-bsd     New BSD License
 * @link http://pl.gravatar.com/site/implement/url
 */
class Zend_View_Helper_Gravatar extends Zend_View_Helper_HtmlElement
{

    /**
     * URL to gravatar service
     */
    const GRAVATAR_URL = 'http://www.gravatar.com/avatar';
    /**
     * Secure URL to gravatar service
     */
    const GRAVATAR_URL_SECURE = 'https://secure.gravatar.com/avatar';

    /**
     * Gravatar rating
     */
    const RATING_G  = 'g';
    const RATING_PG = 'pg';
    const RATING_R  = 'r';
    const RATING_X  = 'x';

    /**
     * Default gravatar image value constants
     */
    const DEFAULT_404       = '404';
    const DEFAULT_MM        = 'mm';
    const DEFAULT_IDENTICON = 'identicon';
    const DEFAULT_MONSTERID = 'monsterid';
    const DEFAULT_WAVATAR   = 'wavatar';

    /**
     * Options
     *
     * @var array
     */
    protected $_options = array('img_size' => 80, 'default_img' => self::DEFAULT_MM,
                                'rating' => self::RATING_G, 'secure' => null);

    /**
     * Email Adress
     *
     * @var string
     */
    protected $_email;

    /**
     * Attributes for HTML image tag
     *
     * @var array
     */
    protected $_attribs;
```

```
    /**
     * Returns an avatar from gravatar's service.
     *
     * @link http://pl.gravatar.com/site/implement/url
     * @param string|null $email Email adress.
     * @param null|array $options Options
     * 'imgSize' int height of img to return
     * 'defaultImg' string img to return if email adress has not found
     * 'rating' string rating parameter for avatar
     * 'secure' bool load from the SSL or Non-SSL location
     * @param array $attribs Attributes for image tag (title, alt etc.)
     * @link http://pl.gravatar.com/site/implement/url More information about gravatar's service.
     * @return string|Zend_View_Helper_Gravatar
     */
    public function gravatar( $email = "", $options = array(), $attribs = array() )
    {

    }

    public function setOptions($options)
    {

    }

    /**
     * Get img size
     *
     * @return int The img size
     */
    public function getImgSize()
    {

    }

    /**
     * Set img size in pixels
     *
     * @param int $imgSize Size of img must be between 1 and 512
     * @return Zend_View_Helper_Gravatar
     */
    public function setImgSize($imgSize)
    {

    }

    /**
     * Get default img
     *
     * @return string
     */
    public function getDefaultImg()
    {

    }

    /**
     * Set default img
     *
     * Can be either an absolute URL to an image, or one of the DEFAULT_* constants
     *
     * @param string $defaultImg
     * @link http://pl.gravatar.com/site/implement/url More information about default image.
     * @return Zend_View_Helper_Gravatar
     */
    public function setDefaultImg($defaultImg)
    {

    }
```

```
    /**
     *  Set rating value
     *
     * Must be one of the RATING_* constants
     *
     * @param string $rating Value for rating. Allowed values are: g, px, r,x
     * @link http://pl.gravatar.com/site/implement/url More information about rating.
     * @throws Zend_View_Exception
     */
    public function setRating($rating)
    {

    }

    /**
     * Get rating value
     *
     * @return string
     */
    public function getRating()
    {

    }

    /**
     * Set email adress
     *
     * @param string $email
     * @return Zend_View_Helper_Gravatar
     */
    public function setEmail( $email )
    {

    }

    /**
     * Get email adress
     *
     * @return string
     */
    public function getEmail()
    {

    }

    /**
     * Load from an SSL or No-SSL location?
     *
     * @param bool $flag
     * @return Zend_View_Helper_Gravatar
     */
    public function setSecure($flag)
    {

    }

    /**
     * Get an SSL or a No-SSL location
     *
     * @return bool
     */
    public function getSecure()
    {

    }

    /**
     * Get attribs of image
     *
     * Warning!
```

```
     * If you set src attrib, you get it, but this value will be overwritten in protected method
_setSrcAttribForImg().
     * And finally your get other src value!
     *
     * @return array
     */
    public function getAttribs()
    {

    }

    /**
     * Set attribs for image tag
     *
     * Warning! You shouldn't set src attrib for image tag.
     * This attrib is overwritten in protected method _setSrcAttribForImg().
     * This method(_setSrcAttribForImg) is called in public method getImgTag().

     * @param array $attribs
     * @return Zend_View_Helper_Gravatar
     */
    public function setAttribs( array $attribs )
    {

    }

    /**
     * Get URL to gravatar's service.
     *
     * @return string URL
     */
    protected function _getGravatarUrl()
    {

    }

    /**
     * Get avatar url (including size, rating and default image oprions)
     *
     * @return string
     */
    protected function _getAvatarUrl()
    {

    }

    /**
     * Set src attrib for image.
     *
     * You shouldn't set a own url value!
     * It sets value, uses protected method _getAvatarUrl.
     *
     * If already exsist overwritten.
     */
    protected function _setSrcAttribForImg()
    {

    }

    /**
     * Return valid image tag
     *
     * @return string
     */
    public function getImgTag()
    {

    }

    /**
```

```
     * Return valid image tag
     *
     * @return string
     */
    public function  __toString()
    {
```

```
        }
  }
```

]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>