Zend_Service_Longurl - Philippus Baalman

<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_Service_Longurl Component Proposal

Proposed Component Name	Zend_Service_Longurl
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Service_Longurl
Proposers	Philippus Baalman
Zend Liaison	TBD
Revision	1.0 - 19 November 2010: Initial Draft. (wiki revision: 9)

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
9. Class Skeletons

1. Overview

Zend_Service_Longurl is a component that implements the Longurl API, version 2.0.

2. References

- Longurl API

3. Component Requirements, Constraints, and Acceptance Criteria

- This component will implement an interface with the Longurl API. "Avoid phishing, malware, and viruses by examining short URLs before visiting them. Find out where links really take you."

4. Dependencies on Other Framework Components

- Zend_Rest_Client
- Zend_Service_Exception

5. Theory of Operation

The component implements all features in the Longurl API.

6. Milestones / Tasks

- Milestone 1: Implementation - [DONE] - Attached
- Milestone 2: Working prototype checked into the incubator
- Milestone 3: Unit tests exist, work, and are checked into SVN.
- Milestone 4: Initial documentation exists.

Name	Size	Creator	Creation Date	Comment
ZIP Archive Zend_Service_Longurl.zip	2 kB	Philippus Baalman	Dec 07, 2010 04:20	This is the working implementation

7. Class Index

- Zend_Service_Longurl
- Zend_Service_Longurl_Exception

8. Use Cases

UC-01

Retrieve the list of known URL shortening services in serialized php format.

```
$longurlservice = new Zend_Service_Longurl();
$longurlservice->setResponseType('php');
$response = $longurlservice->services();
```

UC-02

Retrieve the long url, include the HTML title and internet media type of the destination URL in the response in XML format.

```
$longurlservice = new Zend_Service_Longurl();
$response = $longurlservice->expandUrl(array('url' => 'http://is.gd/w', 'title' =>
true, 'content-type' => true, 'format' => 'xml'));
```

9. Class Skeletons

```
class Zend_Service_Longurl extends Zend_Rest_Client
{
    /**
     * Response Type
     * @var String
     */
    protected $_responseType = 'xml';

    /**
     * Response Format Types
     * @var array
     */
    protected $_responseTypes = array(
        'json',
     'php',
        'xml'
    );

    /**
     * Uri Component
     *
     * @var Zend_Uri_Http
     */
    protected $_uri;

    /**
     * Constructor
     *
     * @param  string $returnType
     * @return void
     */
    public function __construct($responseType = 'xml')
    {
    }

    /**
     * set responseType
     *
     * @param string $responseType
     * @throws Zend_Service_Longurl_Exception
     * @return Zend_Service_Longurl
     */
    public function setResponseType($responseType = 'xml')
    {
    }
```

```
    /**
     * Retrieve responseType
     *
     * @return string
     */
    public function getResponseType()
    {
    }

    /**
     * Get the available services
     *
     * @param array $params
     * @throws Zend_Http_Client_Exception
     * @return mixed
     */
    public function services(array $params = array())
    {
    }

    /**
     * Expand an url
     *
     * @param array $params
     * @throws Zend_Http_Client_Exception
     * @return mixed
     */
    public function expandUrl(array $params = array())
    {
    }
}

class Zend_Service_Longurl_Exception extends Zend_Service_Exception
```

```
    {
    }

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