

Zend_Service_Digg - Luke Crouch & Duo Zheng

<ac:macro ac:name="note"><ac:parameter ac:name="title">Under Construction</ac:parameter><ac:rich-text-body>
<p>NOTICE

This proposal is being revised and is not ready for review!</p></ac:rich-text-body></ac:macro>

<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_Digg Component Proposal

Proposed Component Name	Zend_Service_Digg
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Service_Digg
Proposers	Luke Crouch Duo Zheng Alexander Veremyev (Zend Liaison)
Revision	1.1 - 21 October 2007: Updated from community comments. (wiki revision: 28)

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_Digg is a ZF implementation of popular digg API services.

2. References

- [Digg API docs](#)

3. Component Requirements, Constraints, and Acceptance Criteria

- This component **will** implement the non-draft [ListStories endpoints of the Digg API](#). (Add other endpoints later?)
- This component **will** support all query string parameters.
- This component **will** use an HTTP Accept header to specify json-serialized responses from Digg.
- This component **will not** implement any other endpoints until future releases.

4. Dependencies on Other Framework Components

- Zend_Http_Client
- Zend_Json

5. Theory of Operation

Instantiate an object of Zend_Service_Digg with any valid absolute URI in the constructor, which will be url-encoded and used as the mandatory appkey request parameter. Call getStories() to get an array of Zend_Service_Digg_Story objects, call getStory() to get an individual Zend_Service_Digg_Story object. getStories() has optional container, topic, and grouping (i.e., popular, upcoming, top, hot) parameters.

6. Milestones / Tasks

Describe some intermediate state of this component in terms of design notes, additional material added to this page, and / code. Note any significant dependencies here, such as, "Milestone #3 can not be completed until feature Foo has been added to ZF component XYZ." Milestones will be required for acceptance of future proposals. They are not hard, and many times you will only need to think of the first three below.

- Milestone 1: Working prototype checked into the incubator supporting use case #1
 - Milestone 2: Unit tests exist, work, and are checked into SVN.
 - Milestone 3: Initial documentation exists.
 - Milestone 4: Add implementation for [ListEvents endpoints](#).
- If a milestone is already done, begin the description with "[DONE]", like this:
- Milestone #: [DONE] Unit tests ...

7. Class Index

- Zend_Service_Digg
- Zend_Service_Digg_DataElement
- Zend_Service_Digg_Exception

8. Use Cases

9. Class Skeletons

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

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