Zend_Microformat_Hcard - Pádraic Brady

<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_Microformat_Hcard Component Proposal

Proposed Component Name	Zend_Microformat_Hcard
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Microformat_Hcard
Proposers	Pádraic Brady Matthew Weier O'Phinney (Zend Liaison)
Revision	1.0 - Awaiting Community Comment (wiki revision: 5)

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

hCard is a simple microformat for representing people, groups, companies and even places. It is quite simply a publishable representation of a vCard for embedding in HTML/XHTML. If you need an example, check the source HTML for your Flickr profile.

The purpose of Zend_Microformat_Hcard is to both to collect and generate hCard data. You may collect the data from any URL, or HTML/XHTML source code, and generate hCard output using inputted data.

2. References

- Microformats.org hCard Wiki
- About Microformats

3. Component Requirements, Constraints, and Acceptance Criteria

- This component **will** parse hCard data from any given URL.
- This component **will** parse hCard data from any given HTML/XHTML source.
- This component **will** generate hCard output for embedding in HTML based on user input.
- This component **will** present and accept data using an iterable and countable OO API.

4. Dependencies on Other Framework Components

- Zend_Microformat
- Zend_Microformat_Exception

5. Theory of Operation

The operation of this component is covered in the related Zend_Microformat proposal. In short, when supplied with a URL or HTML source, this component will parse out hCard data into a results object for use. It will also generate similar embeddable output.

Although this component can be used directly for specific purposes, or for implementing custom hCard forms, the majority of use cases work perfectly fine using the API supplied with Zend_Microformat.

6. Milestones / Tasks

- Milestone 1: Assemble some initial use cases to explore desired API
- Milestone 2: Commence proof-of-concept coding (will discard after)
- Milestone 3: Pending review, complete initial development using TDD
- Milestone 4: Complete acceptance testing, verify unit test coverage
- Milestone 5: Let's assume documentation is written during development

7. Class Index

- Zend_Microformat_Hcard
- Zend_Microformat_Hcard_Entity
- Zend_Microformat_Hcard_Result
- Zend_Microformat_Hcard_Exception

8. Use Cases

UC-01

Fetch hCard data using default Zend_Microformat API

```
$mf = new Zend_Microformat('http://www.example.com');
$hcards = $mf->find('hcard');
foreach($hcards as $hcard) {
    echo $hcard->nickname, ' at ', $hcard->url;
}
```

UC-02

Fetch hCard data using hCard class directly

```
$mfcard = new Zend_Microformat_Hcard;
$hcards = $mfcard->find('http://www.example.com');
foreach($hcards as $hcard) {
    echo $hcard->nickname, ' at ', $hcard->url;
}
```

UC-03

Generating hCard embeddable output

```
$mfcard = new Zend_Microformat_Hcard;
$hcard = $mfcard->new();
$hcard->fn = 'someone';
$hcard->n->given-name = 'someone'
$hcard->n->nickname = 'someone';
$hcard->n->url = 'http://www.example.com';
$embeddable = $hcard->generate();
echo $embeddable; // html/xhtml to embed in a page
```

9. Class Skeletons

Please see the above use cases. Based on these, source code will be developed following Test-Driven Development and therefore class skeletons will not be presented in this proposal prematurely. The TDD process will implement Zend_Microformat specifically to enable the use cases presented above.

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