Zend_Service_SecondLife - Lars Strojny

<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_SecondLife Component Proposal

Proposed Component Name	Zend_Service_SecondLife
Developer Notes	http://framework.zend.com/wiki/display/ZFDEV/Zend_Service_SecondLife
Proposers	Lars Strojny Alexander Veremyev (Zend Liaison)
Revision	0.1 - 10 March 2007: Initial submission 0.2 - 26 December: Added details (wiki revision: 24)

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_SecondLife provides a convenient way to access the Second Life API provided by Linden Labs. Second Life is a virtual reality game.
Currently the API just offers methods to create new accounts.

2. References

*Linden Labs PHP-lib (PHP4 only)
*Second Life API documentation
*Sources in the laboratory

3. Component Requirements, Constraints, and Acceptance Criteria

- Zend_Service_SecondLife **will** abstract the access to the SecondLife API
- Zend_Service_SecondLife **will** use Zend_Http_Client for HTTP communication
- Zend_Service_SecondLife **will** wrap all existant SecondLife API calls
- Zend_Service_SecondLife **will not** strictly expose the API calls in the reomte API in 1:1

- Zend_Service_SecondLife **will** normalize API calls and paramater names

4. Dependencies on Other Framework Components

- Zend_Exception

5. Theory of Operation

This component is used to call the Secondlife API. Until now only user registrations are possible. The proprietary SecondLife API protocol is similiar to XML/RPC and therefore a similiar design pattern is used (every value represented by an object).

6. Milestones / Tasks

- Milestone 1: Get feedback
- Milestone 2: Work on prototype in the laboratory
- Milestone 3: [DONE] Working prototype supporting retrieval of valid SL last names, getting error codes, verifying names and creating users.
- Milestone 4: Checkin to the incubator.
- Milestone 5: Unit tests exist, work, and are checked into SVN.
- Milestone 6: Initial documentation exists.

7. Class Index

- Zend_Service_SecondLife
- Zend_Service_SecondLife_Value_Array
- Zend_Service_SecondLife_Value_Boolean
- Zend_Service_SecondLife_Value_Collection
- Zend_Service_SecondLife_Value_Integer
- Zend_Service_SecondLife_Value_Key
- Zend_Service_SecondLife_Value_Llsd
- Zend_Service_SecondLife_Value_Map
- Zend_Service_SecondLife_Value_Abstract
- Zend_Service_SecondLife_Value_Scalar
- Zend_Service_SecondLife_Value_String
- Zend_Service_SecondLife_Exception

8. Use Cases

UC-01: Register at SecondLife

```
$client = new Zend_Service_SecondLife($firstname, $lastname, $password);
if (!in_array($_POST['name'], $client->getLastNames()) {
   $msg = "Invalid last name";
} elseif (!$client->checkName($_POST['first_name'], $_POST['name']) {
   $msg = "Name already registered";
} elsif (!$client->registerUser($_POST['first_name'], $_POST['name'], $_POST['email'],
$_POST['password'], $_POST['date_of_birth'])) {
   $msg = "Could not register account";
} else {
   $msg = "Everything worked fine";
}
```

UC-02: Use Zend_Service_SecondLife::getLastNames() to retrieve last names for select box

```
$client = new Zend_Service_SecondLife($firstname, $lastname, $password);
<form method="post">
<p>
   <label for="first_name">Firstname</label>
   <input id="first_name" type="text" name="first_name"/>
</p>

<p>
   <label for="last_name">Lastname</label>
   <select id="last_name" name="last_name">
       <? foreach ($client->getLastNames() as $name): ?>
       <option><?=$name ?></option>
       <? foreachend; ?>
   </select>
</p>

<p>
   <label for="email">Email</label>
   <form id="email" name="email" type="text"/>
</p>

<p>
   <label for="date_of_birth">Date of birth</label>
   <form id="date_of_birth" name="date_of_birth" type="text"/>
</p>

<input type="submit" value="Register"/>
</form>
```

9. Class Skeletons

See laboratory for a fully working prototype

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

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