openbiz-cubi
Rapid PHP Application Development Platform

Search projects

Project Home   **Downloads**   **Wiki**   **Issues**   **Source**   Export to GitHub

Search Current pages ▼ for   Search

CubiRestService
Add Restful Service on Cubi
Phase-Design

Introduction

Quoted from wikipedia "A RESTful web API (also called a RESTful web service) is a web API implemented using HTTP and REST principles. It is a collection of resources, with four defined aspects:

- the base URI for the web API, such as http://example.com/resources/
- the Internet media type of the data supported by the web API. This is often JSON but can be any other valid Internet media type provided that it is a valid hypertext standard.
- the set of operations supported by the web API using HTTP methods (e.g., GET, PUT, POST, or DELETE).
- The API must be hypertext driven."

Cubi implements restful service on top of Slim framework http://www.slimframework.com/.

Add Restful Service

Add Restful service in Cubi is pretty simple. Here we use contact module as example.

- In contact module, add RestService.xml in modules/contact/websvc. Specify its Class attribute, say ContactRestService.
- Add ContactRestService.php under the same folder and extend ContactRestServicefrom Cubi base RestService implemented under websvc/lib/RestService.php
- In your class, set the resourceDOMap. This map links the resource name in the uri to dataobject which is the real resource client wants access.

```
class ContactRestServiceextends RestService
{
    protected $resourceDOMap = array('contacts'=>'contact.do.ContactDO');
}
```

You are free to implement your own logic by overriding parent class's get, post, put, delete methods.

- Now you can access the restful api with uri like
  - http://host/cubi/rest.php/contact/contacts/1?format=xml
  - http://host/cubi/rest.php/contact/contacts/1?format=json
  - http://host/cubi/rest.php/contact/contacts/q?format=json&page=1&rows=10&sort=fieldname&sorder=asc
  - http://host/cubi/rest.php/contact/contacts/q?format=json&firstname=michael
- For other methods POST, PUT, DELETE, you can write test code with curl or other http client class. The contact module includes Restful service testing code at /modules/contact/websvc/ContactRestTest.php

General Web Service

Besides Restful web service, Cubi allows developers to build general web service in easy way.

First, create a service xml. Define implementation class, public methods and their access right. Leave access as blank if the method is open to everyone.

```
<?xml version="1.0" standalone="no"?>
<PluginService Name="userService" Class="userService" MessageFile="user/message/login.msg">
    <PublicMethod Name="login" Access=""/>
    <PublicMethod Name="register" Access=""/>
</PluginService>
```

Second, create implementation service class by extending WebsvcService

```
    /**
     * login action
     * @input string username
     * @input string password
     * on success return ('redirect'=>url), on error return ('errors'=>(field1=>errormsg1,field2=>errormsg2))
     * @return array.
     */
<?php
require_once MODULE_PATH.'/websvc/lib/WebsvcService.php';
class userService extends  WebsvcService {
    public function login($username=null, $password=null) {
        return ...
    }
```

```
    }
```

Now the web service url will be like

- http://localhost/cubing/cubi/ws.php/user/userService/?method=login&format=json&argsJson={'username':'michael','password':'jones'}
- or http://localhost/cubing/cubi/ws.php/user/userService/ with post method=login&format=json&argsJson= {'username':'michael','password':'jones'}
- or http://localhost/cubing/cubi/ws.php/user/userService/?method=login&format=json&arg_username=michael&ard_password=jones

Format can be either json or xml which is the default.

To do

Enable web service compliance with JSON-RPC http://en.wikipedia.org/wiki/JSON-RPC

► Sign in to add a comment