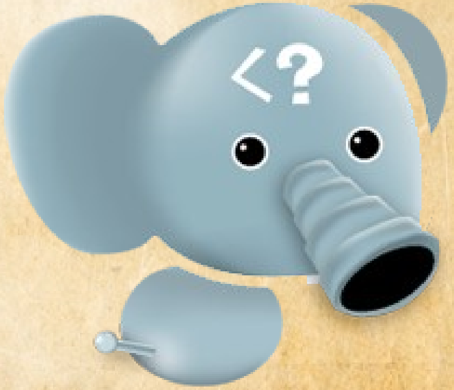
CodeIgniter

RESTful API Design & Implementation with CodeIgniter PHP Framework

2012 PHP Conference



Who Am I

Bo-Yi Wu
@appleboy
http://blog.wu-boy.com

任職於瑞昱半導體 RealTek(IC Design House)

- TV 多媒體部門

- Sencha Touch 2, Backbone.js, CodeIgniter, Node.js, MongoDB, MySQL, Twitter Bootstrap, Twitter Hogan ...

Who Am I

- Open Source Contributions (github: appleboy)
  - CodeIgniter-Native-Session
  - CodeIgniter-i18n
  - CodeIgniter-Template
  - CodeIgniter-Nexmo-Message
  - CodeIgniter-TextMagic-API

My Focus

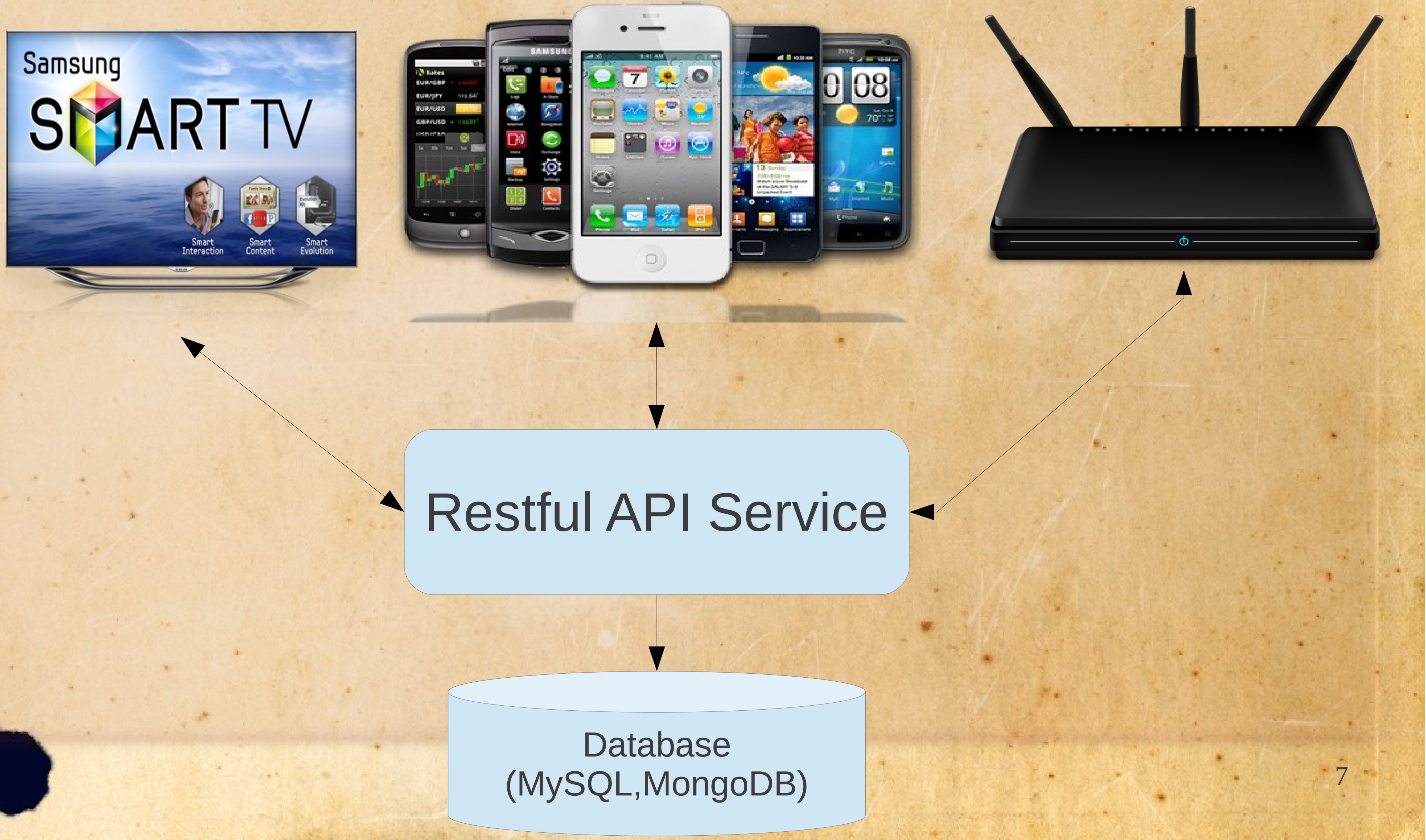
- CodeIgbiter 3.0.x develop branch
  - Support Native Session
  - Support HMVC
  - Support Sparks Package Management System
- Laravel develop branch
- Javascript (Node.js, Socket.io, Express, Backbone.js)

Outline

- Restful API Basic
- API Design
- Implementing API with CodeIgniter
- Verify your API

Restful API Basic

Why Use Restful?



Restful API Service

Database
(MySQL,MongoDB)

What is REST? Http Method

- POST
- GET
- PUT
- DELETE
- OPTIONS

Define in RFC 2616

並非所有的瀏覽器都支援

PUT, DELETE

http://api.jquery.com/jQuery.ajax/

如何解決未支援的瀏覽器

```
<input type='hidden' name='type' value='PUT'>
```

將PUT,DELETE一併寫成POST API

REST Response Format?

- JSON*
- XML
- Array
- Html
- CSV

現在皆以 JSON 為主

格式簡單 , 相容性高 , 閱讀方便

Javascript Object Notation

```
{
        key1: 'value1',
        key2: 20121103
        key3: [1,2,3]
}
```

JSON in Javascript is familiar

```
var object = {
    key1: 'value1',
    key2: 20121103
    key3: [1,2,3]
}
```

JSON in PHP (encode)

PHP

```
<?php
echo json_encode(array(
    'key' => 'value'
));
```

Outputs

```
{key: 'value'}
```

JSON in PHP (decode)

PHP

```
<?php
$json_data = '{key: value}';
echo json_decode({
    'key' => 'value'
});
```

Outputs

```
array(
        'key' => 'value'
);
```

你不可不知的 JSON 基本介紹

http://goo.gl/Wvhwb

API Design

良好的 API 設計

- Simple 簡單
- Intuitive 直觀的
- Stable 穩定
- Well Document 線上文件

Using Facebook API

Fucking Stable and Document

請務必撰寫 API 線上文件

大家每天在花在討論的時間太長

Debug 時間變少

良好的文件減少人與人溝通成本

團隊合作

Http Method RFC 2616

CRUD

- Create
- Read
- Update
- Delete

Method

- POST
- GET
- PUT
- DELETE

API URL Define

/API/Module/Method

Format 1: Topic Module

- /API/Topic/Add
- /API/Topic/Update
- /API/Topic/Delete
- /API/Topic/List

Format 2: Topic Module

- /API/Topic/Add
- /API/Topic/Update/1234
- /API/Topic/Delete/1234
- /API/Topic/List/sort/asc

個人偏好格式 1

不用記住多種不同 API 格式

API Response Format

請勿常常修改 format

（除非你想黑掉）

Example Create API

Input

```
var object = {
    title: 'value1',
    type: 'value2',
    user_id: '1000'
};
```

Output

```
{
    title: 'value1',
    type: 'value2',
    user_id: '1000',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Add

Example Create API

Input

```
var object = {
    title: 'value1',
    type: 'value2'
};
```

Output

```
{
    title: 'value1',
    type: 'value2',
    user_id: '1000',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Add/1000

Example Update API

Input

```
var object = {
    id: '1000',
    title: 'value1',
    type: 'value2'
};
```

Output

```
{
    id: '1000',
    title: 'value1',
    type: 'value2',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Update

Example Update API

Input

```
var object = {
    title: 'value1',
    type: 'value2'
};
```

Output

```
{
    id: '1000',
    title: 'value1',
    type: 'value2',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Update/1000

Example Delete API (single)

Input

```
var object = {
    id: 1000
};
```

Output

```
{
    id: '1000',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Delete

Example Delete API (multiple)

Input

```
var object = {
    id: [1000, 1001]
};
```

Output

```
{
    id: '1000',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Delete

Example Delete API

Input

```
var object = {

};
```

Output

```
{
    id: '1000',
    success_text: 'ok'
}
```

http://site.com/API/Topic/Delete/1000

Example Read API (Single)

Input

```
var object = {
    id: 1000
};
```

Output

```
{
    id: '1000',
    success_text: 'ok',
    item: {
        title: 'Kate Upton'
    }
}
```

http://site.com/API/Topic/List

Example Search API (Multiple)

Input

```
var object = {
    q: 'Kate Upton'
};
```

Output

```
{
    id: '1000',
    success_text: 'ok',
    items: [
        {title: 'I am kate'},
        {title: 'I am Upton'}
    ]
}
```

http://site.com/API/Topic/List

Kate

Upton

多虧了Youtube API讓我在上班時增加了很多動力

How to handle versioning?

內部 API 大改版

Old: http://site.com/v1/API/Topic/Add
New: http://site.com/v2/API/Topic/Add

利用URI Routing功能
Framework or mod_rewrite

http://site.com/API/Topic/Add

=

http://site.com/v1/API/Topic/Add

API Implementation

不用自己造輪子

Phil Sturgeon’s CodeIgniter REST Server

http://github.com/philsturgeon/codeigniter-restserver

Requirements

- PHP 5.2 or greater
- CodeIgniter 2.1.x to 3.0-dev

How to install?

Installation

- Drag and drop the following files into your application's directories
  - application/libraries/Format.php
  - application/libraries/REST_Controller.php
  - application/config/rest.php

Setup the config

- $config['rest_default_format'] = 'json';
- $config['rest_enable_keys'] = false;
- $config['rest_enable_logging'] = false;
- $config['rest_enable_limits'] = false;
- $config['rest_ajax_only'] = false;

Include REST Controller

```php
require(APPPATH.'/libraries/REST_Controller.php');
```

Handling Requests

```
class Topic extends REST_Controller
{
    public function index_get() {}
    public function index_post() {}
    public function index_update() {}
    public function index_delete() {}
}
```

CRUD Requests

```
class Topic extends REST_Controller
{
    public function list_get() {}
    public function add_post() {}
    public function update_update() {}
    public function delete_delete() {}
}
```

Accessing parameters is also easy

Parameters

- GET
  - `$this->get('blah');`
- POST
  - `$this->post('blah');`
- UPDATE
  - `$this->update('blah');`
- DELETE
  - `$this->delete('blah');`

Create API

Input

```
var object = {
    title: 'Kate Upton',
    text: 'Beautiful girl'
};
```

Output

```
{
    id: '1000',
    success_text: 'ok',
}
```

http://site.com/API/Topic/Add

Create API (POST)

```
public function Add_post()
{
    if (!$this->post('title')) {
        $this->response(array('error' => 'Title is required'), 404);
    }

    $output = $this->lib_topic->insert($data);

    if ($output) {
        $this->response($output, 200);
    } else {
        $this->response(array('error' => 'Insert error'), 404);
    }
}
```

Update API

Input

```
var object = {
    id: 1000,
    title: 'Kate Upton',
    text: 'Beautiful girl'
};
```

Output

```
{
    id: '1000',
    success_text: 'ok',
}
```

http://site.com/API/Topic/Update

Update API (PUT)

```
public function Update_put()
{
    if (!$this->update('id')) {
        $this->response(array('error' => 'ID is required'), 404);
    }

    $output = $this->lib_topic->update($this->update('id'), $data);

    if ($output) {
        $this->response($output, 200);
    } else {
        $this->response(array('error' => 'Insert error'), 404);
    }
}
```

Delete API

Input

```
var object = {
    id: 1000
};
```

Output

```
{
    id: '1000',
    success_text: 'ok',
}
```

http://site.com/API/Topic/Delete

Delete API (DELETE)

```
public function Delete_delete()
{
    if (!$this->delete('id')) {
        $this->response(array('error' => 'ID is required'), 404);
    }

    $output = $this->lib_topic->delete($this->delete('id'));

    if ($output) {
        $this->response($output, 200);
    } else {
        $this->response(array('error' => 'Insert error'), 404);
    }
}
```

Read API (GET)

Input

```
var object = {
    id: 1000,
    type: [1, 2]
};
```

Output

```
{
    id: '1000',
    success_text: 'ok',
    item: {
        title: 'Kate Upton'
    }
}
```

http://site.com/API/Topic/List

Read API (GET)

```
public function List_get()
{
    if (!$this->get('id') or ) {
        $this->response(array('error' => 'ID is required'), 404);
    }

    $output = $this->lib_topic->list($this->get('id'), $this->get('type'));

    if ($output) {
        $this->response($output, 200);
    } else {
        $this->response(array('error' => 'Insert error'), 404);
    }
}
```

目錄結構

Folder

```
application
    controllers/
        api/
         topic.php
         user.php
         acl.php
system
index.php
```

Routing (config/routes.php)

Default URL http://site.com/api/topic/Add

```
$route['API/Topic/(:any)'] = 'api/topic/$1';
$route['API/User/(:any)'] = 'api/user/$1';
$route['API/Acl/(:any)'] = 'api/acl/$1';
```

New URL http://site.com/API/Topic/Add

Verify Your API

✓

一樣不需要自己造輪子

Phil Sturgeon’s
CodeIgniter REST Client

https://github.com/philsturgeon/codeigniter-restclient

Requirements

Requirements

- PHP 5.1+
- CodeIgniter 2.0.0+
- CURL
- CodeIgniter Curl library: http://getsparks.org/packages/curl/show

Load Rest Client Library

Load Library

```
// Load the rest client spark
$this->load->spark('restclient/2.1.0');

// Load the library
$this->load->library('rest');
```

Setup API Server

```
// Run some setup
$this->rest->initial('xxxxxx');

// twitter server
$this->load->initial('http://twitter.com');
```

Parameter

```
// set api path
$api = '/API/Topic/Add';

// set api data
$data = array(
  'title' => 'I am Kate Upton',
  'type' => 'girl'
);
```

Test it

```
// GET API
$this->rest->get($api, $data);
// POST API
$this->rest->post($api, $data);
// UPDATE API
$this->rest->update($api, $data);
// DELETE API
$this->rest->delete($api, $data);
```

$this->rest->debug();

Rest Client Library debug mode

以上是CodeIgniter PHP Framework

Implement another Framework?

Laravel PHP Framework?

```
public $restful = true;
```

```
class Home_Controller extends Base_Controller
{
    public $restful = true;

    public function get_index()
    {
        //
    }

    public function post_index()
    {
        //
    }

}
```

More Introduction to Laravel Framework

14:20 – 14:50

用 Laravel Framework 打造現代化網站應用程式

大澤木小鐵

RESTful API 就講到這裡

如果有任何問題

可以上 CodeIgniter 論壇

http://www.codeigniter.org.tw/forum/

謝謝大家及工作團隊