
目錄

Introduction	1.1
BaseController	1.2
BaseModel	1.3
表单配置	1.4

SwooleDistributedWeb v0.1.1

SwooleDistributed v1.6升级为v1.7，增加单元测试模块

部分bug修复

SwooleDistributedWeb v0.1.0

swoole 分布式全栈框架 web开发增强版，基于SwooleDistributed，增加了web开发所需要的常见功能，100%支持SwooleDistributed原有功能，目前SwooleDistributed版本是v1.6。

文档

1. SwooleDistributed 文档 <https://tmtbe.gitbooks.io/swooledistributed/content/>
2. Swoole 文档 <http://wiki.swoole.com/>

特性

1. 优秀的框架（MVC）设计,丰富的支持极大加快开发速度
2. 通过开启不同端口同时支持TCP和HTTP，WebSocket，同一逻辑处理不同协议
3. 全异步支持，无需手动处理连接池，异步redis,异步mysql，mysql语法构建器，支持异步mysql事务,异步httpclient，效率出众
4. 协程模式全支持，异步redis，异步mysql，异步httpclient，异步task，全部都提供了协程模式，业务代码摆脱处处回调的困扰（不是swoole2.0，php7同样支持）
5. 支持协程嵌套，支持协程内异常处理（和正常逻辑用法一样）
6. 额外提供了protobuf完整RPC实例，轻松使用protobuf
7. 天然分布式的支持，一台机器不够零配置，零代码修改完成高效分布式系统的搭建

以上都是流弊SwooleDistributed的特性，SwooleDistributedWeb特性如下：

- i. 封装常用的web功能，支持cookie、session、cache、form、validate、filter、验证码、文件上传、分页等
- ii. 自动生成表单，自动生成表单前台验证js，后台自动验证表单；只需在model中配置即可使用
- iii. 数据库操作封装增删改查，更加方便编写
- iv. session/cache 支持redis（默认）、mysql、文件（分布式不建议使用）
- v. 使用plates模板引擎
- vi. 可businessConfig.php中指定域名、默认控制器、默认方法、文件上传目录、上传大小；内置uc、ckeditor

安装须知

- 1. php 7.0 5.6的用户需要自己修改源码，将php7的部分语法重写。强烈推荐php7.X搭配最新的swoole1.X系列。
- 2. 强烈建议使用最新版的swoole，请通过github下载编译。最新版修复了很多php7下的bug
- 3. 需要redis支持，安装hredis扩展 swoole编译时选择异步redis选项
- 4. 需要composer支持，安装composer，运行composer install安装依赖
- 5. 如需集群自行搭建LVS

运行

php start_swoole_server.php start

控制器基类

- 属性

```
/**
 * 缓存
 * @var ICache
 */
protected $cache;
/**
 * session handler
 * @var ICache
 */
public $session_handler;
/**
 * 控制器名
 * @var string
 */
public $controller_name;
/**
 * 方法名
 * @var string
 */
public $method_name;
```

- initialization

```
/**
 * 不能有yield
 * 初始化每次执行方法之前都会执行initialization
 * 初始化需要返回true/false，返回true表示代码继续往下执行，返回false表示
中断执行，可能需要页面跳转
 */
public function initialization()
{
    return true;
}
```

- form表单生成、验证

```
//表单名字，对应model中的方法
$form_name = 'regForm';
//表单提交
if ($this->isPost()){
    //表单验证
    $error = '';
    $data_arr = [];
    $is_pass = yield $this->checkForm($this->model, $form_name,
$this->post(), $error, $data_arr);
    if ($is_pass !== true){
        $this->showmessage($error);
        return ;
    }
}
//生成表单
$form = $this->getForm($this->model, $form_name);
$this->view('User/reg', ['form'=>$form]);
```

- request、get、post

```
/**
 * 获取参数值，先get后post
 * @param string $index 变量名
 * @param mixed $default 默认值
 * @param bool $xss_clean
 * @return string/array
 */
protected function request($index='', $default='', $xss_clean = true)
```

- model

```
/**
 * 获取一个model
 * @param string $model_name
 * @return BaseModel
 */
protected function model($model_name)
```

- view

```
/**
 * 加载视图
 * @param string $template
 * @param array $data
 */
protected function view($template, $data=[])
```

- 获取cookie

```
/**
 * 获取cookie
 * @param string $key
 * @return string
 */
protected function getCookie($key)
```

- 设置cookie

```
/**
 * 设置cookie
 * @param string $key
 * @param string $value
 * @param number $expire    多长时间过期，单位秒
 * @param string $path
 * @param string $domain
 * @param string $secure
 * @param string $httponly
 */
protected function setCookie($key, $value, $expire = 0, $path =
 '/', $domain = '', $secure = false, $httponly = false)
```

- 跳转

```
/**
 * 跳转
 * @param string $uri
 * @param $params 参数
 */
protected function redirect($uri, $params=[])
```

- 输出到浏览器

```
/**
 * 输出到浏览器
 * @param mixed $data
 */
protected function output($data='', $is_ajax=false)
```

- ajaxOutput

```
protected function ajaxOutput($data)
```

- 是否post提交

```
protected function isPost()
```

- 是否get提交

```
protected function isGet()
```

- 是否ajax提交

```
protected function isAjax()
```

- 保存session

```
/**
 * 保存session
 * @param string $key
 * @param string $value
 * @param int $ttl session过期时间
 */
protected function setSession($key, $value, $ttl=0)
```

- 获取session

```
/**
 * 获取session
 * 需要加yield
 * @param string $key
 */
protected function getSession($key)
```

- 获取php session_id

```
protected function getPhpseSSID()
```

- 删除session


```
protected function delSession($key)
```

- 自动生成表单

```
/**
 * 自动生成表单
 * @param BaseModel $modelObj
 * @param string $form_name
 * @param array $data 默认值
 */
protected function getForm(BaseModel $modelObj, $form_name, $data=[])
```

- 表单验证

```
/**
 * 表单验证
 * 开启form_secret（默认开启）时，需要使用yield
 *
 * @param BaseModel $modelObj 模型
 * @param string $form_name 模型中的表单名字
 * @param array $input 表单输入项
 * @param string $error
 * @param array $data_arr
 * @return boolean|Generator
 */
protected function checkForm(BaseModel $modelObj, $form_name, $input, &$error, &$data_arr=[])
```

- 获取表单中字段的值

```
/**
 * 获取表单中字段的值
 * @param BaseModel $modelObj 模型
 * @param string $form_name 模型中的表单名字
 * @param array $input 表单输入项
 */
protected function getFormValue(BaseModel $modelObj, $form_name,
    $input)
```

- 获取ip

```
protected function ip()
```

- 分页函数

```
/**
 * 分页函数
 *
 * @param $num 信息总数
 * @param $curr_page 当前分页
 * @param $perpage 每页显示数
 * @param $urlrule URL规则
 * @param $array 需要传递的数组，用于增加额外的方法
 * @return 分页
 */
protected function pages($num, $curr_page, $perpage = 20, $urlrule = '', $array = array(), $setpages = 5)
```

- 获取当前页面的url，包含query_string

```
protected function get_url()
```

- 输出提示信息

```
protected function showmessage($msg)
```

- 引入uc

```
protected function uc()
```

- 获取referer

```
protected function referer()
```

model基类

- 属性

```
/**
 * 表前缀
 */
public $tbl_prefix = 'sdw_';
/**
 * 表名，不需要表前缀
 */
public $tbl_name = '';

/**
 * 设置Form Secret防止，非当前页面提交数据
 */
public $_form_secret = true;

/**
 * 表单配置项
 *
 */
public $_form = [];
```

- 完整表名

```
public function table()
```

- 获取某一列的值

```

/**
 * 获取某一列的值，
 * 需要加yield
 * @param array $contidions_arr 查询条件
 * @param string $fields      sql列名
 * @param string $order_column
 * @param string $order_by
 * @param string $group
 * @return string|null
 */
public function getColumn($contidions_arr, $field, $order_column
    = '', $order_by='DESC', $group = '')

```

- 查询一条结果

```

/**
 * 查询一条结果
 * 需要加yield
 * @param array $contidions_arr 查询条件
 * @param string $fields      sql列名
 * @param string $return_result true:返回查询结果；false:返回coroutine，需要额外获取结果
 * @param string $order_column
 * @param string $order_by
 * @param string $group
 * @return array|\Generator
 */
public function getOne($contidions_arr, $fields='*', $return_result=true, $order_column = '', $order_by='DESC', $group = '')

```

- 查询 多条结果

```

/**
 * 查询
 * 返回多条结果
 * 需要加yield
 * @param array $conditions_arr      查询条件
 * @param string $fields      sql列名
 * @param bool $return_result      true:返回查询结果; false:返回coroutine, 需要额外获取结果
 * @param int $page
 * @param int $pagesize
 * @param string $order
 * @param string $group
 * @param int $bind_id
 * @return array|\Generator
 */
public function select($conditions_arr, $fields='', $return_result=true, $page=1, $pagesize=0, $order = '', $group = '', $bind_id=null)

```

- 执行sql

```

/**
 * 执行sql
 * 需要yield
 * @param string $sql
 * @param string $return_result
 * @param number $page
 * @param number $pagesize
 * @param int $bind_id
 */
public function query($sql, $return_result=true, $page=1, $pagesize=0, $bind_id=null)

```

- 插入

```
/**
 * 插入
 * 需要yield
 * @param array $data_arr    要插入的数据
 * @return insert_id
 */
public function insert($data_arr)
```

- 更新

```
/**
 * 更新
 * 需要yield
 * @param array $data_arr
 * @param array $conditions_arr
 * @return bool
 */
public function update($data_arr, $conditions_arr)
```

- 删除

```
/**
 * 删除
 * 需要yield
 * @param array $conditions_arr
 * @return bool
 */
public function delete($conditions_arr)
```

- 统计数量

```
/**
 * 统计数量
 * 需要加yield
 * @param array $conditions_arr 查询条件
 * @param string $return_result
 * @return int|Generator
 */
public function count($conditions_arr, $return_result=true)
```

- 表单配置

参考下一章节

表单配置

例子

```
public $_form = [  
    'username' => [                                //注册使用  
        'label' => '用户名',  
        'tips' => '',  
        'type' => 'input',  
        'value' => '',  
        'placeholder' => '请输入用户名',  
        'class' => 'input w1',  
        'validates' => [  
            'required'=>'请输入用户名',  
            'minlen' => '6|用户名最少6位',  
            'maxlen' => '32|用户名最多32位',  
            'ctype' => 'common|请输入正确的用户名',  
            'ajax' =>[  
                'type' => 'post',  
                'url' => 'ajax/checkUsername',  
                'data' => '',  
                'buttons' => '$("#dosubmit")',  
                'onError' => '用户名已注册',  
                'onWait' => '验证中...',  
            ],  
        ],  
    ],  
    'password' => [  
        'label' => '密码',  
        'tips' => '建议密码由字母、数字组合、至少6位',  
        'type' => 'password',  
        'placeholder' => '',  
        'class' => 'input w1',  
        'validates' => [  
            'required'=>'请输入密码',  
            'ctype' => 'password|密码格式不正确',  
            'minlen' => '6|密码格式不正确',  
            'maxlen' => '32|密码格式不正确',  
        ],  
    ],  
]
```

```

    ]
  ],
  'repassword' => [
    'label' => '确认密码',
    'tips' => '请再次输入密码',
    'type' => 'password',
    'placeholder' => '',
    'class' => 'input w1',
    'validates' => [
      'required'=>'请输入确认密码',
      'minlen' => '6|确认密码格式不正确',
      'equalo' => 'password|两次密码不一致,请重输',
    ]
  ],
  'checkcode' => [
    'label' => '验证码',
    'tips' => '',
    'type' => 'checkcode',
    'value' => '',
    'placeholder' => '',
    'class' => 'input w2',
    'validates' => [
      'required' => '请输入验证码',
      'ajax' =>[
        'type' => 'post',
        'url' => 'ajax/checkcode',
        'data' => '',
        'buttons' => '$("#dosubmit")',
        'onError' => '验证码不正确',
        //          'onWait' => '验证中...',
      ],
    ]
  ],
  // 用户资料
  'realname' => [
    'label' => '姓名',
    'tips' => '您的姓名',
    'type' => 'input',
    'maxlength' => 10,
    'value' => '',
  ]

```

```

        'placeholder' => '',
        'class' => 'input w1',
        'validates' => [
            'required'=>'请输入姓名',
            'minlen' => '2|最少2位',
            'maxlen' => '10|最多10位',
            'ctype' => 'chinese|仅限中文',
        ]
    ],
    'thumb' => [
        'label' => '个人头像',
        'tips' => '尺寸200X200像素，能代表个人形象，请勿使用低俗图片、二
        维码、其他机构企业等涉嫌误导用户的头像',
        'type' => 'upload',
        'upload' => [
            'type' => 'image',    //上传文件类型，image：图片，
            'width' => '200',    //上传图片宽度，超过的部分会截取
            'height' => '200',    //上传图片高度，超过的部分会截取
            'succ_js' => '',//上传成功后执行的js，回调参数是data
        ],
        'validates' => [
            'required'=>'请上传个人头像',
            'ctype' => 'url|请上传个人头像',
        ]
    ],
    'city_select' => [ //select所在容器的id
        'label' => '所在地',
        'type' => 'cascade_select',
        'jsonName' => 'name',
        'jsonValue' => 'val',
        'selects' => [
            'province' => [
                'data-first-title' => '选择省',
                'data-first-value' => '0',
                'data-required'=>'true',
                'data-url' => 'ajax/linkage',
                'data-value' => '17',
                'class' => 'prov',
                'data-query-name' => 'linkage_id', //ajax参数名
                'validates' => [

```

```

        'required'=>'请选择省份',
        'ctype' => 'int|仅限数字',
    ]
],
'city' => [
    'data-first-title' => '选择地市',
    'data-first-value' => '0',
    'data-url' => 'ajax/linkage',
    'data-required'=>'true',
    'data-value' => '0',
    'class' => 'city',
    'data-query-name' => 'linkage_id', //ajax参数名
    'validates' => [
        'required'=>'请选择地市',
        'ctype' => 'int|仅限数字',
    ]
],
'county' => [
    'data-first-title' => '选择区县',
    'data-first-value' => '0',
    'data-url' => 'ajax/linkage',
    'data-empty-style'=>"none",
    'data-required'=>'true',
    'data-value' => '0',
    'class' => 'dist',
    'data-query-name' => 'linkage_id', //ajax参数名
    'validates' => [
        'ctype' => 'int|仅限数字',
    ]
],
],
],
'pics' => [
    'label' => '多图上传',
    'tips' => '',
    'type' => 'multi_upload',
    'upload' => [
        'type' => 'image',    //上传文件类型，image：图片，
        'max_nums' => 5,      //最多上传几张图片，默认一张
        'width' => '0',      //上传图片宽度，超过的部分会截取
    ]
]

```

```

        'height' => '0',    //上传图片高度，超过的部分会截取
        'succ_js' => '',    //上传成功后执行的js，回调参数是data
    ],
    'validates' => [
        'ctype' => 'url|请上传照片',
    ]
],
];

```

- 表单项使用

```

/**
 * 注册表单项
 */
public function regForm() {
    return ['username_reg', 'password', 'repassword', 'checkcode'];
}

/**
 * 登录表单
 * @return string[]
 */
public function loginForm() {
    return ['username', 'password'];
}

/**
 * 个人资料
 * @return string[]
 */
public function profileForm() {
    return ['realname', 'email', 'thumb', 'city_select', 'pics'];
};
}

```

表单项配置说明

- 配置项的格式

```
key=>[                                //key一般是表中的字段，cascade_select除外
  'label' => '',                      //
  'tips' => '',
  'type' => 'input',                  //表单类型，详见form types
  'value' => '',                      //默认值
  'placeholder' => '',
  'class' => '',                      //表单样式
  'validates' => [],                 //验证规则，详见validates
  ....                               //也可以定义其他，会自动加到表单中
]
```

- 表单类型form types

input文本框

text多行文本

password密码框

checkcode验证码

radio单选框

```
'option' => [
  '1' => '男',
  '2' => '女',
]
```

checkbox复选框

upload单文件上传，需要额外定义

```
'upload' => [
  'type' => 'image',    //上传文件类型，image：图片，
  'width' => '200',     //上传图片宽度，超过的部分会截取
  'height' => '200',   //上传图片高度，超过的部分会截取
  'succ_js' => '',      //上传成功后执行的js，回调参数是data
],
```

multi_upload多文件上传，比单文件上传多个max_nums

```
'upload' => [
  'type' => 'image',    //上传文件类型，image：图片，
  'max_nums' => 5,      //最多上传几张图片，默认一张
  'width' => '0',       //上传图片宽度，超过的部分会截取
  'height' => '0',     //上传图片高度，超过的部分会截取
  'succ_js' => '',      //上传成功后执行的js，回调参数是data
],
```

cascade_select级联下拉，这个定义最复杂

```
'jsonName' => 'name',
'jsonValue' => 'val',
'selects' => [
  'province' => [
    'data-first-title' => '选择省',
    'data-first-value' => '0',
    'data-required'=>'true',
    'data-url' => 'ajax/linkage',
    'data-value' => '17',
    'class' => 'prov',
    'data-query-name' => 'linkage_id', //ajax参数名
    'validates' => [
      'required'=>'请选择省份',
      'ctype' => 'int|仅限数字',
    ]
  ],
],
]
```

- 验证规则**validates**，可以同时填写多个规则

required：非空

```
'required'=>'不能为空'
```

minlen：最少长度，中英文一样都是1个

```
'minlen' => '2|最少2位'
```

maxlen：最大长度，中英文一样都是1个

```
'maxlen' => '10|最多10位'
```

equalo：检查对象相等的情况

```
'equalo' => 'password|两次密码不一致,请重输'
```

regx：正则

```
'regx' => '/\d+/'|仅限数字'
```

ctype：内置正则，email，mobile、domain、chinese、english等具体参见
[app/Helpers/Libs/Validate.php](#)

```
'ctype' => 'chinese|仅限中文'
```

ajax


```
'ajax' =>[
'type' => 'post',
'url' => 'ajax/checkcode',
'data' => '',
'buttons' => '$("#dosubmit")',
'onError' => '验证码不正确',
'onWait' => '验证中...',
]
```