

单元测试-测试报告

组名：Khaos 项目名称：百讯汇 测试人：符永顺，邓荃文，谭烁 测试时间： 2010-12-25 到2011-1-3 审核人： 邓荃文 审核时间： 2011-1-5 测试报告版本：ver1.3

法国jhk hg

人 的方式

目录

[一.测试概要 3](#_Toc281839359)

[1.概述及测试目的 3](#_Toc281839360)

[2.测试项目版本 3](#_Toc281839361)

[3.测试软硬件环境 3](#_Toc281839362)

[二.测试内容 3](#_Toc281839363)

[1.测试类型 3](#_Toc281839364)

[2.测试方式 4](#_Toc281839365)

[3.覆盖率分析 4](#_Toc281839366)

[三.测试结果 5](#_Toc281839367)

[1.测试用例一. 5](#_Toc281839368)

[1.1用例一描述 5](#_Toc281839369)

[1.2用例一预期结果 5](#_Toc281839370)

[1.3用例一测试实际结果 6](#_Toc281839371)

[1.4结果分析 6](#_Toc281839372)

[2.测试用例二. 6](#_Toc281839373)

[2.1用例二描述 6](#_Toc281839374)

[2.2用例二预期结果 6](#_Toc281839375)

[2.3用例二测试实际结果 7](#_Toc281839376)

[2.4结果分析 7](#_Toc281839377)

[3.测试用例三 7](#_Toc281839379)

[3.1用例三描述 7](#_Toc281839380)

[3.2用例三预期结果 7](#_Toc281839381)

[3.3用例三测试实际结果 8](#_Toc281839382)

[3.4结果分析 8](#_Toc281839383)

[4. 结果中的Bug概览 8](#_Toc281839384)

[4.1测试发现的Bug 8](#_Toc281839385)

[4.2 测试Bug的总结及分析 9](#_Toc281839386)

[4.3 针对Bug的改进建议 9](#_Toc281839387)

[4.4 Bug最后的处理状态 9](#_Toc281839388)

[5. 单元测试总结 9](#_Toc281839389)

[四.特殊需求及说明 10](#_Toc281839390)

[1.提交的运行脚本说明 10](#_Toc281839391)

[2. 脚本运行方式 10](#_Toc281839392)

[3.特殊说明 11](#_Toc281839393)

[五.附件 11](#_Toc281839394)

[1.Wiki测试要求 11](#_Toc281839395)

[2.涉及到的文件 11](#_Toc281839396)

# 一.测试概要

## 1.概述及测试目的

本测试主要为项目的单元测试。即将程序的每个独立单元与其他部分隔离开的情况下进行测试。主要对MVC架构的项目进行单元测试，包括model部分和controller部分

## 2.测试项目版本

|  |  |  |
| --- | --- | --- |
| 条目 | 名称 | 版本 |
| 数据库 | database\_v1.0.sql | 1.0 |
| 项目代码 | MyCake\_ver1.5.5.rar | 1.55 |
| 用到的框架代码 | Cakephp & SimpleTest | 1.2.5 & 1.0.1 |

## 3.测试软硬件环境

|  |  |  |
| --- | --- | --- |
| 环境 | 网络服务器 | 客户端 |
| 硬件配置 | CPU ： Intel(R)  Celeron(R) CPU  AMD Athlon™ 64 X2  Memory： 2G | CPU ： Intel(R)  Celeron(R) CPU  AMD Athlon™ 64 X2  Memory： 2G |
| 软件配置 | Apache 2.2.0  MySQL  PHP5.3 | OS：Window s 7  浏览器：firefox3.6 |

# 二.测试内容

## 1.测试类型

对MVC架构的项目进行单元测试，包括model部分和controller部分。

## 2.测试方式

由于本程序是在Cakephp框架上进行编写的，而针对Cakephp有一个专门的单元测试工具，那就是simpleTest. 可先介绍下simpTest：

*CakePHP 整合了一个综合性的测试框架，该框架是SimpleTest的一个扩展。 但SimpleTest并没有包含在CakePHP的安装包中，于是首先，我们需要从*[*http://simpletest.sourceforge.net/*](http://simpletest.sourceforge.net/)*下载它。*

测试代码的时候，可以使用测试模板(fixture) 去生成临时的数据表并加载测试用的数据。使用测试模板的好处在于不会对现行数据有任何地干扰。CakePHP会自行使用我们在app/config/database.php中定义的名为$test的数据库配置进行数据库连接。

CakePHP会按照下述的顺序来执行基于测试模板的测试：

1. *根据测试模板的定义创建数据表*
2. *如果模板中有定义数据，则生成表数据。*
3. *执行测试*
4. *清空测试用表*
5. *从数据库中删除测试用表*

也就是说，在cakephp中配置好simpleTest后，接下来的工作就是对框架生成的测试模板和测试文件（.test.php）进行自己所需要的修改和添加。而测试模板相当于一个数据库，创建测试模板需要决定两件事：表怎么建立(也就是说表中有哪些字段)，以及测试表中应该放些什么数据。这部分在\app\tests\fixtures文件夹中。

创建好测试模板后，就是最重要的单元测试主体部分了。

测试文件有关的一些规定或者说基准：

1. *测试用的PHP文件放在app/tests/cases/[some\_folder]下。*
2. *这些文件都以.test.php结尾而不是.php.*
3. *测试类都应该继承自CakeTestCase或者CakeWebTestCase。*
4. *任何包含测试的方法(比如包含一个断言assertion)都以test开头，比如testedit()。*

由于本项目的程序基于MVC架构，且只存在model、control、view。所以只测试 model和control两个部分。

创建好测试用例后，可以通过访问**http://your.cake.domain/cake\_folder/test.php**(取决于你的具体设置，一般情况下，为<http://localhost/MyCake/test.php> )，点击测试用例及特定文件来执行测试。

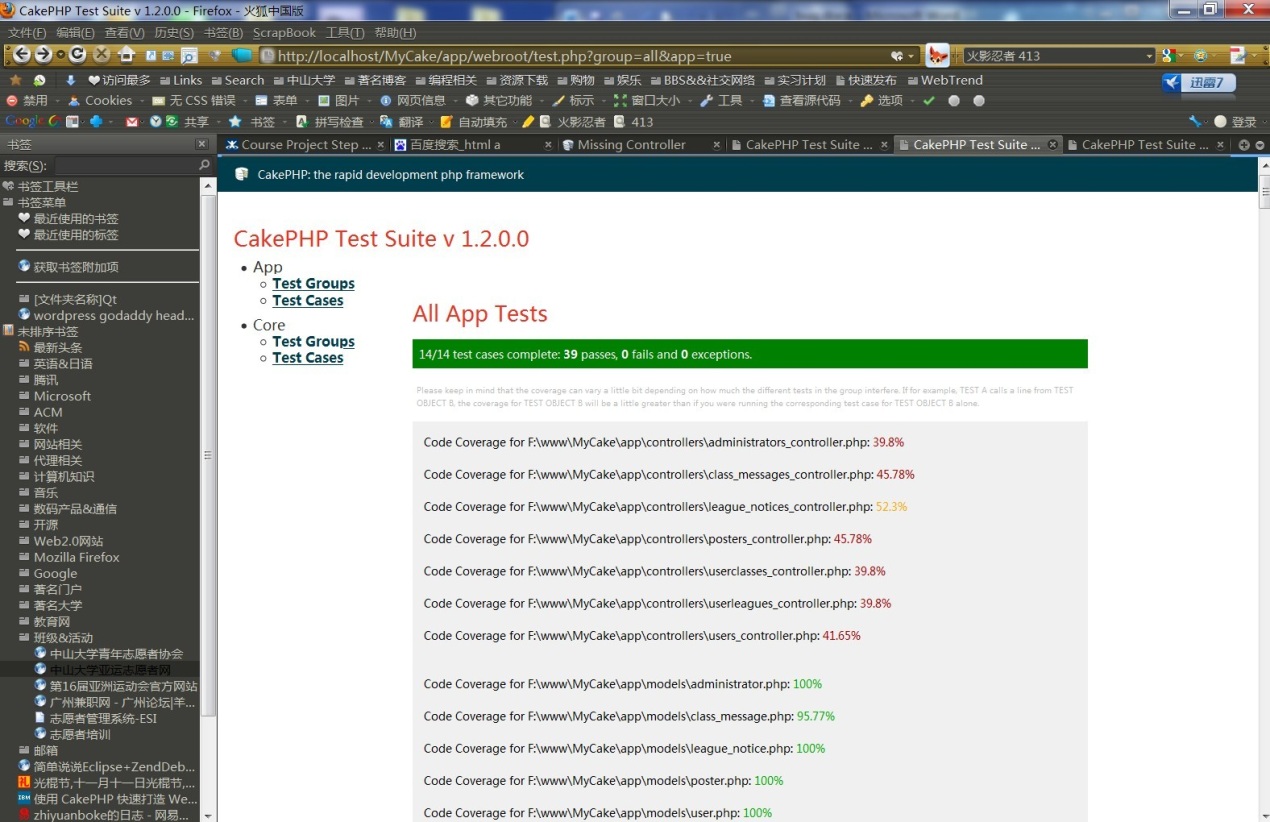
## 3.覆盖率分析

本测试采用基于SimpleTest的Cakephp的Test Suite1.2.0框架，对基于MVC架构的源程序进行了系统的测试，在单元测试方面，共对7个Controller和7个Model进行了测试，分为单个文件的Test Cases测试和对整个网络应用的Test Groups--AllTests测试两部分。

*代碼覆蓋率的两种评判方式：分别是line coverage/statement coverage和 path coverage*

因为我们这里是对行数进行分析，计算行覆盖率，所以而覆盖率分析则采用Xdebug工具。并且CakePHP的Test Suite也对Xdebug提供了良好的支持。

在得到的结果中，共14个 test Cases的用例，有50几个断言，其中有44个pass，6个失败，后来经过修改，舍去一些有问题的部分（主要是测试代码的问题，不是源代码），得到39个全部pass，而在代码覆盖率中，经测试发现，单个test case访问及对代码的覆盖有限，仅10%不到，但总体测试中基本都达到40%要求（部分接近40%），部分对于Model的测试由于涉及到对数据库的存取比较多而全面，所以几乎达到了100%对相应model的访问。



# 三.测试结果

## 1.测试用例一.

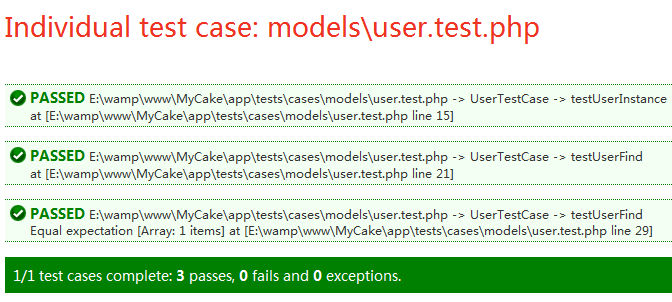
### 1.1用例一描述

检验登录的用户与数据库中的用户是否一致。

### 1.2用例一预期结果

如果输入用户正确，那结果应该通过，否则不通过。

### 1.3用例一测试实际结果



### 1.4结果分析

结果符合预期

## 2.测试用例二.

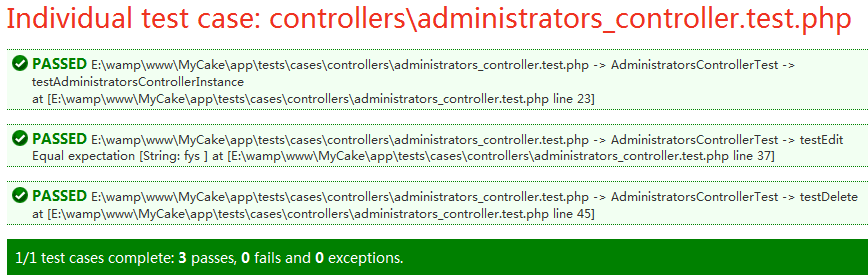
### 2.1用例二描述

检验管理员编辑页面的输入是否正确存进数据库

### 2.2用例二预期结果

都可以正确存进数据库表

### 2.3用例二测试实际结果



### 2.4结果分析

与预期相符。

## 3.测试用例三

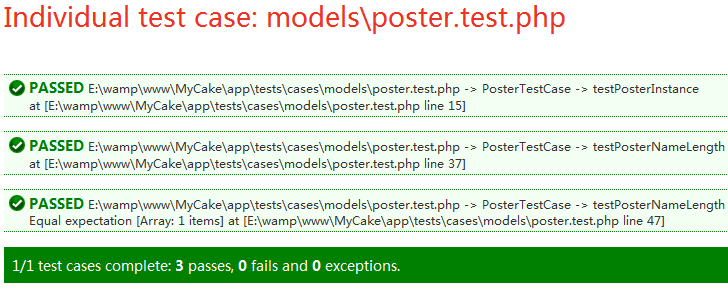
### 3.1用例三描述

检验社团的海报墙是否实例化

### 3.2用例三预期结果

可以实例化

### 3.3用例三测试实际结果



### 3.4结果分析

符合预期结果

注：以上几个单元测试只是部分示例，详细单元测试请看测试代码。

## 4. 结果中的Bug概览

### 4.1测试发现的Bug

1. 字符长度限制问题，如poster的imaName的字符长度，当实际字符长度超过限制长度时，得出错误结果，但主页面并没有提示。
2. 点击海报墙后，转向的页面有问题。并没有转向相应的社团页面。
3. 判断错误，应该以user\_leagueid判断而不是id
4. 社团无法发新通知。
   1. 可以删除，编辑，但成功之后无法正常显示。
5. 添加数据代码重复后，出现的找不到页面的错误
6. 数据库设计未考虑系统管理员角色，导致用系统管理员进行操作的时候出现找
7. 不到页面错误
8. 权限控制异常
9. etc.

### 4.2 测试Bug的总结及分析

在所有的Bug中，测试发现的bug 主要集中在代码语义有误 和不同类之间的关系不正确等几个方面，属于一般性的缺陷。前台编码方面的bug，占到了全部bug的将近一半。

除此之外，还有权限 bug，主要表现：

1. *具有相应按钮操作的权限，页面无相应按钮，无法执行该功能*
2. *无相应按钮操作权限，页面有相应按钮，点击按钮能出现权限异常错误*
3. *有相应按钮操作权限，有相应按钮，执行该功能出现权限异常错误*

### 4.3 针对Bug的改进建议

这些Bug的问题，一是来源于前台的编码，另一方面是类的设计。对于编码问题，根据错误所在地方，进行纠错即可。而类设计问题，则可能需要修改类之间的依赖等关系，

### 4.4 Bug最后的处理状态

|  |  |  |  |
| --- | --- | --- | --- |
| BUG编号 | 描述 | 后期是否解决 | 如果没有，说明原因 |
| 1 | 字符长度限制 | 是 |  |
| 2 | 海报墙转向错误 | 是 |  |
| 3 | 社团无法发布通知 | 是 |  |
| 4 | 数据重复异常 | 是 |  |
| 5 | 管理员角色界面 | 否 | 未完全实现 |
| 6 | 权限控制异常 | 是 |  |
| 7 | etc |  |  |

## 5. 单元测试总结

此次测试严格按照项目计划和测试计划执行，按时完成了测试计划规定的测试对象的测试。针对测试计划规定的测试策略，在测试执行中都有体现，在测试执行过程中，依据测试计划和测试用例，对系统大部分功能进行了测试。

由于对Cakephp框架和SimpleTest测试框架都不是太熟悉，且这方面的资料实在是少之又少。在整个单元测试过程中，基本上都是摸着石头过河。但经过小组成员的努力，至少现在能实现对程序中model和controller里的大部分功能的单元测试。

本次测试尝试用了回归测试，并使用黑盒测试和白盒测试相结合的方式，但由于临近期末，精力实在有限，就没有详细的完成。

总的来说，真的还是很努力地去根据wiki的要求去做了，尽管我们在这个过程中还要面对各种课程的大作业以及期末考试的复习，但还是尽量挤出时间去研究simpleTest，然后根据我们项目中model和controller里的函数去给test里面的fixture和case添加测试代码。而写测试代码的过程可以说又是一次项目的设计与实现，在这个过程中，我们又进一步熟悉和改进我们已有的程序。可以说，整个测试过程，收获很多。

# 四.特殊需求及说明

## 1.提交的运行脚本说明

提交的是php文件，在确保程序MYCake配置正确之后(项目配置详见帮助文件《MyCake\_AllInfo\_UserManual.pdf》,可在我们的Github上的~Project\_Docs中找到)，运行脚本的配置比较简单。

1. 将Xdebug放入php的ext目录
2. 修改php.ini，添加

*[xdebug]*

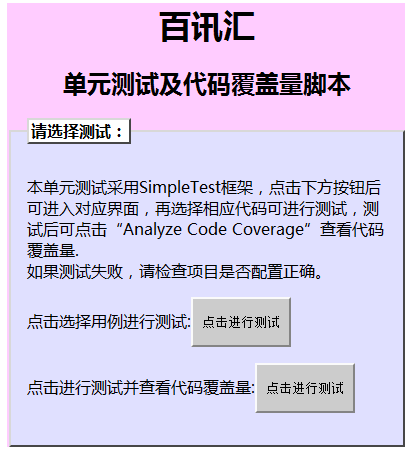
*zend\_extension = php的目录/php5.3.0/ext/php\_xdebug.dll*

到末尾，保存并重启wampserver等网络服务器

1. 将测试脚本UnitTest&CodeCoverage.php放入网站根目录（同目录下有我们的应用MyCake）
2. 浏览器访问我们的localhost/UnitTest&CodeCoverage.php即可

## 2. 脚本运行方式

按照上述说明配置之后，访问url，按照网页提示，点击相应按钮即可。两个按钮分别对应TestCases测试和总体测试并查看代码覆盖率。

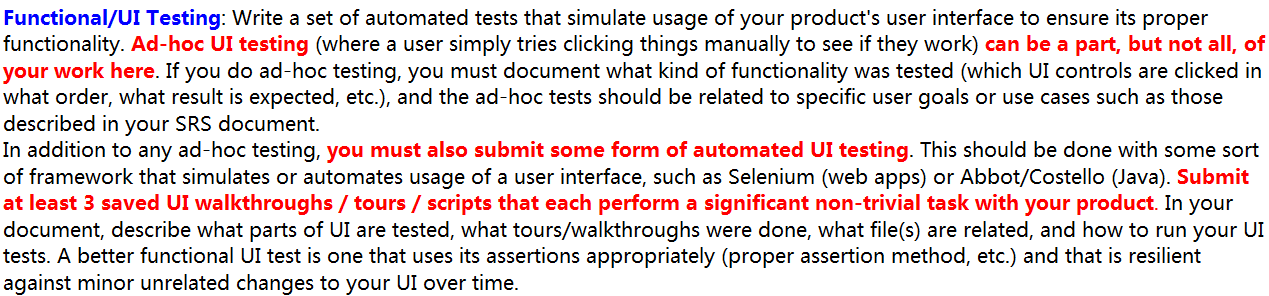


## 3.特殊说明

无

# 五.附件

## 1.Wiki测试要求



## 2.涉及到的文件

a.测试报告 《UnitTestReport\_BaiXunHui\_AllInfo\_ByKhaos》

b.录制的脚本

UnitTest&CodeCoverage.php

c.需要的测试工具Xdebug

d.Simpletest框架已经融合进我们的项目，不用另行配置。

e.测试结果文件夹，将结果网页另存为两个Html。