

# 黑盒测试

——3-1 普通 OO 是 AD 而且还是 OOAD 这样的 J2EE 你喜欢吗

## 1. 简介

### 1.1 目的

本测试计划说明书旨在提高测试效率及版本质量,涵盖数据及数据库完整性测试、接口测试、集成测试、功能测试、用户界面测试、性能评测、负载测试、强度测试、容量测试、安全性和访问控制测试、故障转换和恢复测试、配置测试、安装测试等方面,用于指导测试的具体实施,以期发现系统存在的错误并进行改正。本文方的预期读者是参与本项目的开发人员。

### 1.2 背景

a.

项目的委托单位:厦门大学信息学院软件工程系

开发单位:

厦门大学信息学院 2017 级“普通 OO 是 AD 而且还是 OOAD 这样的 J2EE 你喜欢吗”团队。

主管部门:

厦门大学信息学院。

b.

该软件系统与其他系统的关系。

产品名称:

Litmall 系统。

开发背景:

Litmall 系统是一个 B2C 网络购物平台,其商品涵盖家居、服装、日用品、母婴产品等多种门类。还拥有团购优惠、组合商品优惠以及不定时发放的商品优惠券等优惠策略,能有效地刺激用户消费。平台还提供了广告平台以及专题平台等商品宣传平台。为在满足繁杂需求的同时保证平台的运行效率和易扩展性、易维护性、易管理性,我们将对 Litmall 系统进行面向对象的重新设计,为系统搭建新的运行框架。

解决问题:

为了对 Litmall 系统进行面向对象设计,我们需要设计若干个对象来满足系统的需求,并保证对象之间符合高内聚、低耦合的原则。

在目前阶段，本文档设计这些对象所包含的属性以及类之间的联系，不涉及类的方法设计。首先对原本的 Litemall 系统进行四个维度的需求分析。根据已有的前端反向分析 Litemall 系统应该具有功能，并为新需求设计功能，按照面向对象的方法，产出对应的类图、实体关系图、活动图、对象状态机图、数据流图。

在完成面向对象的需求分析之后，为提高开发效率，课程成员应进行模块分工，除此之外需要形成标准工作小组来对模块工作进行宏观调整和控制。

相关系统：

用户支付订单时需要第三方微信支付系统和银行提供的支付接口的支持。

用户在注册账号和忘记密码时需要第三方验证系统的支持。

用户的订单需要第三方物流系统的支持。

## 1.3 范围

本次测试涉及本系统的如下模块：

- (1) userInfo 用户模块（用户注册、登录、管理个人信息等；管理员新增、登录、管理个人和用户信息等；以及网关）
- (2) discount 折扣模块(团购、团购规则；优惠券，优惠券规则等)
- (3) address 地址模块（地址的增删查改等）
- (4) 数据库连接、同步、更新。

# 2 . 任务概述

## 2.1 目标

本测试计划的目标是为了确保项目在正式上线之前通过测试，达到以下目标：

1. 达到《需求规格说明书》、《概要设计说明书》中的主要功能需求、非功能需求是否已实现，业务流程是否正确；能在已有 litemall 界面的基础上实现界面对应的功能。
  2. 项目规定的操作和运行稳定
  3. 错误率和 bug 数控制在可接受的范围之内。
- 测试的最终目标是为了确保项目的质量得到保障。

## 2.2 运行环境

### 2.2.1 项目部署结构

//

### 2.2.2 测试软硬件信息

#### 服务器软件环境

资源	名称/类型
数据库服务器	Mysql
运行操作系统环境	Windows
应用服务器中间件	J2EE 应用服务器
JDK 版本	1.8

#### 服务器硬件环境

设备名称	系统配置	备注
数据库服务器	CPU: Intel(R) Xeon(R) CPU E5-2682 v4 @ 2.50GHz 内存: 2G 硬盘空间: 40G SSD 数量: 1	——
应用服务器	CPU: Intel(R) Xeon(R) CPU E5-2682 v4 @ 2.50GHz 内存: 2G 硬盘空间: 40G SSD 数量: 1	——

#### 测试机软件环境

资源	名称/类型
系统	Window 10、IOS
浏览器	Chrome、FireFox

#### 测试机硬件环境

资源	名称/类型
硬件配置——服务端	CPU: I7-6700HQ 内存: 12GB 系统: Windows 磁盘空间: 250GB
硬件配置——客户端	Iphone、Android

## 2.3 需求概述

本测试计划主要针对 Discount 服务、Address 服务、User 服务的主要功能进行测试。

Discount 服务主要功能包括：优惠券、团购、预售三块功能。其中核心功能有：用户抢优惠券、订单使用优惠券、管理员增删改查优惠券规则；用户参加团购活动、团购根据参团人数实现分级返款、管理员增删改查团购活动；用户参加预售活动、进行定金尾款支付、计算预售活动优惠等。

Address 服务主要功能包括：获取用户地址，管理员查询用户地址，用户添加修改删除查询地址等。

User 服务主要功能包括：用户、管理员、zuul 网关三块功能。其中核心功能有：登陆注册、修改用户信息、获取返点、管理员管理信息、权限控制、分发 token、转发请求等。

## 2.4 条件与限制

仅在给定的测试环境下进行，暂不考虑未列出。

在有限人力、资源、时间的前提下，保证测试可靠性，并尽可能完善测试的全面性。

# 3 . 测试策略

## 3.1 整体策略

### 3.1.1 测试调度策略标准

在开始进行测试时必须满足下列条件：

1. 提交的版本的单元测试已通过，具备可测性
2. 测试计划和测试方案的制订已完成
3. 缺陷跟踪与管理系统已搭建
4. 测试所需的资源已经到位
5. 测试组人员配置合理，测试人员的工作技能符合测试要求
6. 测试所需的软、硬件和操作系统等测试环境准备完毕

出现下面任一情况时，测试活动就可能暂停：

1. 被测系统有大量错误或严重错误或流程走不下去，继续测试没有意义
2. 测试环境遭到破坏，无法继续测试。如：测试所需的设备没有到位，测试环境被病毒感染等等
3. 性能测试：当被测的功能或模块存在严重的性能缺陷的情况下暂停测试

如果测试暂停，满足下面条件时，测试重新开始：

1. 开发组成功安装，并测试通过了产品的基本功能

### 3.1.2 测试质量评估标准

按照系统测试计划完成系统测试。

达到系统测试所规定的覆盖率的要求：

- 1) 测试用例执行覆盖率应达到 80%；
- 2) 测试需求覆盖率应达到 100%；
- 3) 系统满足需求规格说明书的要求。

在系统测试中发现的缺陷达到修改标准：

- 1) 致命和严重级缺陷修复率应达到 100%；
- 2) 一般和轻微级缺陷修复率根据实际情况达到 95%以上。

BUG 级别说明：

BUG 分 4 个严重级别：致命、严重、一般和建议。具体描述如下：

致命 BUG：

- 1) 测试执行主要功能直接导致系统死机、蓝屏、挂起或是程序非法退出；
- 2) 被测系统的主要功能点没有实现；
- 3) 主要模块/功能不满足需求或设计上的要求；
- 4) 软件的安全缺陷导致重要数据丢失或损坏，且无法恢复。

严重 BUG：

- 1) 测试执行次要功能导致系统死机、蓝屏、挂起或是程序非法退出；
- 2) 被测系统的次要功能点没有实现；
- 3) 对于主要功能的执行结果与预期结果差别较大，或是计算结果不正确；
- 4) 软件的易用性不好，导致用户可能不能正常完成软件的主要功能操作；
- 5) 主要界面有明显的错别字或描述错误。

一般 BUG：

- 1) 软件的实际执行过程与预期结果有差异，但不严重；
- 2) 非正常操作或输入导致系统出错，或执行结果不正确；
- 3) 系统运行过程中偶尔（出现概率<5%）有出错提示或导致系统运行不正常；
- 4) 软件交互性不好，对于用户可能造成难于操作、学习和理解；
- 5) 在用户经常使用的环境中，界面不美观，影响软件品质；
- 6) 界面、程序或帮助文档中文档或文字描述问题，造成用户难于理解。

建议 BUG：

- 1) 软件的实际执行过程与预期结果有较小的差异；
- 2) 软件不能处理用户可能使用的极端条件下的操作；
- 3) 界面、程序或帮助文档中文档或文字描述问题，但影响不大。

### 3.1.3 测试完成准则

主要质量属性	详细要求
正确性	能够防止脏、废数据进入数据库；从接口读取得数据正确无误。
健壮性	系统有较强的容错性，能够保证在出现非预期状况下正常运行
可靠性	系统在不断电情况下持续工作。 系统无单点故障。 系统具有动态负载均衡处理能力，保证用户享受最快的信息服务。
性能，效率	响应性能：用户可以容忍 1—2 秒的延迟。 要求为在 10000 名用户同时在线的情形下，所有页面 2 秒内需要能显示。 数据存储时间：要求数据库用户设置详细信息在线长期保存，系统数据详细信息要求在服务器中长期保存。
易用性	提供方便的系统安装程序，系统服务器安装配置方便易操作。 提供友好、方便的功能界面。 尽量减少用户输入信息量，提高数据信息共享程度，提供方便的帮助信息。
清晰性	提供足够的软件说明文档，配图表说明
安全性	保证数据访问的安全性，同时对关键数据采取访问权限限制。 保证数据的完整性、一致性和有效性。 保证用户、系统业务数据传输过程的安全性、完整性及不可抵赖性。 操作系统、数据库系统符合安全标准，提供管理、监控和故障处理等功能。 采用操作员登录身份认证机制，进入系统采用密码认证进入，建立完整的日志记录，服务器脚本进行加密，使用户无法看到网页脚本源代码，防止伪造身份人员冒用系统资源。
可扩展性	系统应有良好的横向和纵向扩展能力，可以通过提高服务器主机

	的性能提高整个系统的处理能力。 在软件方面，响应时间，更新处理时间，数据传送和转换时间，处理和解决问题时都比较迅速，满足用户需求。当用户需求，如操作方式，运行环境，结果精度，数据结构与其他软件接口等发生变化时，设计的软件要做适当调整，灵活性非常大。 各模块负载能力及整体负载能力应可平滑扩展，新功能模块的增加应不影响现有模块的运行。
兼容性	保证系统与各种硬件和操作系统具有良好的兼容性 支持手机、桌面端各主流操作系统和分辨率自适应
数据精度	
抗压性	

依据标准本次测试中测试文档的编写、测试用例的编写、具体的执行测试以及测试中各项资源的分配和估算，均以各子系统的需求文档、设计文档为标准，软件的执行以系统逻辑设计构架为依据。

## 4 . 测试项目说明

【按顺序逐个对测试项目做出说明：】

### 4.1 测试项目名称及测试内容

### 4.2 测试用例

编写测试用例，包括用例编号、输入数据、预期的输出结果、操作步骤

测试用例编号	Test01
测试项目	用户登录
输入	输入错误的密码
操作步骤	1.进入小程序 2.打开登陆页面

	3.输入正确的用户名 4.输入错误的密码 5.点击登陆
预期输出	系统显示“密码错误”

测试用例编号	Test02
测试项目	用户登录
输入	输入错误的用户名
操作步骤	1.进入小程序 2.打开登陆页面 3.输入错误的用户名 4.输入密码 5.点击登录
预期输出	系统显示“密码错误”

测试用例编号	Test03
测试项目	用户登录
输入	输入正确的密码和用户名
操作步骤	1.进入小程序 2.打开登陆页面 3.输入正确的用户名 4.输入正确的密码
预期输出	登陆成功，进入商城首页

测试用例编号	Test04
测试项目	获取用户返点数
输入	Token
操作步骤	1.传入 token 2.解析出 userid 3.根据 userid 获取用户返点数
预期输出	用户的返点数

测试用例编号	Test05
测试项目	修改个人信息
输入	用户信息
操作步骤	1.传入 token 、修改后的个人信息 2.解析出 userid 3.根据 userid 将个人信息修改
预期输出	修改后的用户信息



测试用例编号	Test06
测试项目	查找用户的地址列表
输入	Token, 页数 page, 每页数据条数 limit
操作步骤	1.传入 token 2.解析出 userid 3.根据 userid 获取当前页数的用户地址列表
预期输出	用户的地址列表

测试用例编号	Test07
测试项目	查看地址详细信息
输入	地址 id
操作步骤	1.传入 id 2.根据 id 获取用户详细信息
预期输出	用户详细信息

测试用例编号	Test08
测试项目	查看地址详细信息
输入	错误的地址 id
操作步骤	1.传入 id 2.根据 id 获取用户详细信息 3.查找不到数据
预期输出	输出此地址不存在

测试用例编号	Test09
测试项目	修改地址信息
输入	地址 id、待修改的地址信息
操作步骤	1.传入 id、地址信息 2.根据 id 修改用户的地址信息
预期输出	用户详细信息

测试用例编号	Test10
测试项目	删除地址
输入	地址 id
操作步骤	1.传入 id 2.根据 id 删除地址
预期输出	删除成功

测试用例编号	Test11
测试项目	管理员查询地址列表
输入	空的用户 id、收货人姓名、page、limit
操作步骤	1.传入 userid、收货人姓名 2.userid 为空 3.根据姓名查询地址列表

	4.返回地址列表
预期输出	返回当前页数的地址列表

测试用例编号	Test12
测试项目	管理员查询地址列表
输入	用户 id、空的收货人姓名、page、limit
操作步骤	1.传入 userid、收货人姓名 2.收获姓名为空 3.根据 userid 查询地址列表 4.返回地址列表
预期输出	返回当前页数的地址列表

测试用例编号	Test13
测试项目	管理员查询地址列表
输入	错误的用户 id、收货人姓名、page、limit
操作步骤	1.传入 userid、收货人姓名 2.根据 userid 查询地址列表 3.在上一步地址列表中根据姓名查询地址列表 4.返回地址列表
预期输出	返回当前页数的地址列表

测试用例编号	Test14
测试项目	新增一条地址
输入	Token、address 对象
操作步骤	1.传入 token 和地址对象 2.解析出 userid 3.根据 userid 新增地址 4.返回地址对象
预期输出	返回新增的地址对象

测试用例编号	Test15
测试项目	新增一条地址
输入	Token、缺少收货姓名的 address 对象
操作步骤	1.传入 token 和地址对象 2.解析出 userid 3.根据 userid 新增地址 4.提示没有收货姓名
预期输出	返回“没有收货姓名”

测试用例编号	Test16
测试项目	新增一条地址
输入	Token、缺少电话的 address 对象
操作步骤	1.传入 token 和地址对象

	2.解析出 userid 3.根据 userid 新增地址 4.提示没有收货电话
预期输出	返回“没有收货电话”

测试用例编号	Test17
测试项目	新增一条地址
输入	Token、缺少详细地址的 address 对象
操作步骤	1.传入 token 和地址对象 2.解析出 userid 3.根据 userid 新增地址 4.提示没有详细地址
预期输出	返回“没有详细地址”

#### 4.2.1 输入

【输入的数据和输入命令。】

#### 4.2.2 输出

【预期的输出数据。】

#### 4.2.3 步骤及操作

#### 4.2.4 允许偏差

【给出实测结果与预期结果之间允许偏差的范围。】

### 4.3 进度

测试活动	完成日期	工作量（人数）
测试大纲	2019/12/10	1
测试设计	2019/12/12	3
集成测试	2019/12/14	5
系统测试	2019/12/17	20
项目验收	2019/12/19	——

## 4.4 条件

【给出测试对资源的特殊要求，如设备、软件、人员等。】

### 4.4.1 人员

项目在人员分配上的设定：

角色	所推荐的最少资源（所分配的专职角色数量）	具体职责或注释
后端工程师	5	完成系统后台的交互和互动，以及数据库的建立和维护
测试工程师	3	检查软件存在的错误和问题，检测软件的稳定性，编写相应的测试规范和测试用例
产品经理	1	与客户沟通，确定需求和提出需求更改要求
备注		项目集成现有前端界面，故不需要前端工程师

### 4.4.2 设备

硬件环境：

- 1) PC 机
- 2) iphone
- 3) ipad
- 4) Macbook

### 4.4.3 软件

软件环境：

- 1) 操作系统：Windows、IOS
- 2) 数据库：MySQL
- 3) 服务器：NGINX

## 4.5 测试资料

3-1 组，需求规格说明书，2.0 版本

3-1 组，概要设计说明书，2.0 版本

3-1 组，详细设计说明书；

## 5. 评价

### 5.1 范围

### 5.2 准则

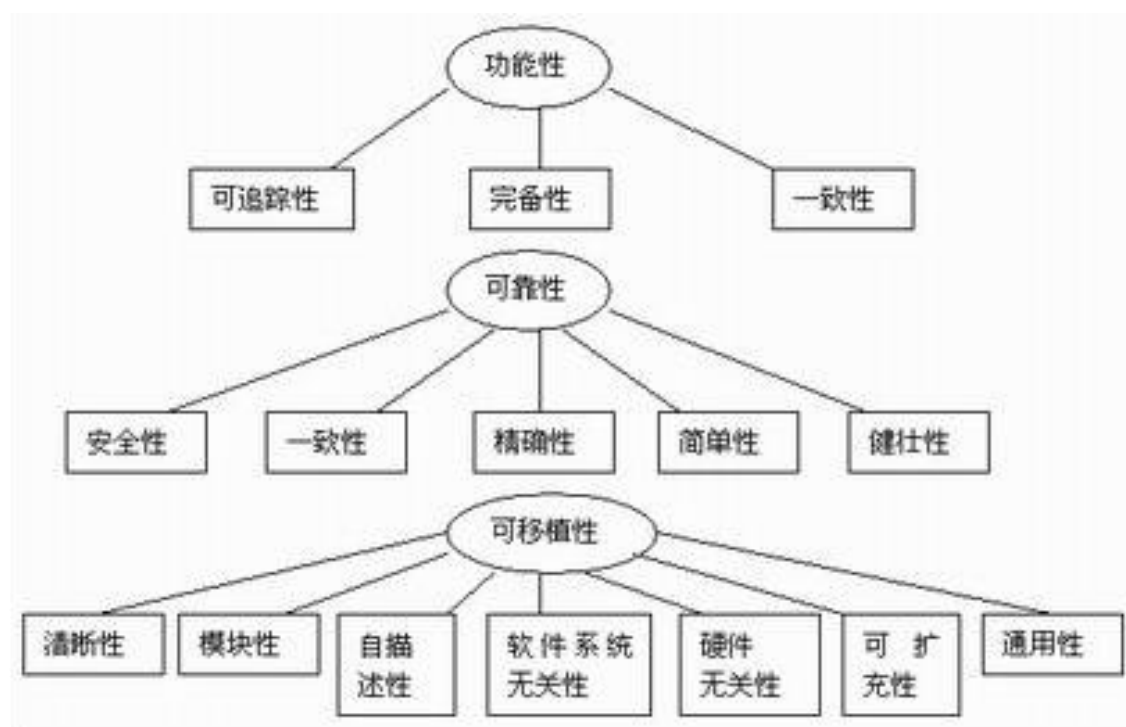
#### 5.2.1 软件质量要素

1. 功能性 :软件所实现的功能满足用户需求的程度 .功能性反映了所开发的软件满足用户称述的或蕴涵的需求的程度，即用户要求的功能是否全部实现了。
2. 可靠性 :在规定的的时间和条件下，软件所能维持其性能水平的程度。可靠性对某些软件是重要的质量要求，它除了反映软件满足用户需求正常运行的程度，且反映了在故障发生时能继续运行的程度。
3. 易使用性 :对于一个软件，用户学习、操作、准备输入和理解输出时，所做努力的程度。易使用性反映了与用户的友善性，即用户在使用本软件时是否方便。
4. 效率 :在指定的条件下，用软件实现某种功能所需的计算机资源（包括时间）的有效程度。效率反映了在完成功能要求时，有没有浪费资源，此外 " 资源"这个术语有比较广泛的含义，它包括了内存、外存的使用，通道能力及处理时间。
5. 可维修性 :在一个可运行软件中，为了满足用户需求、环境改变或软件错误发生时，进行相应修改所做的努力程度。可维修性反映了在用户需求改变或软件环境发生变更时，对软件系统进行相应修改的容易程度。一个易于维护的软件系统也是一个易理解、易测试和易修改的软件，以便纠正或增加新的功能，或允许在不同软件环境上进行操作。

6. 可移植性：从一个计算机系统或环境转移到另一个计算机系统或环境的容易程度。

## 5.2.2 评价准则

评价准则可分成 22 点。包括精确性（在计算和输出时所需精度的软件属性）；健壮性（在发生意外时，能继续执行和恢复系统的软件属性）；安全性（防止软件受到意外或蓄意的存取、使用、修改、毁坏或泄密的软件属性）；以及通信有效性、处理有效性、设备有效性、可操作性、培训性、完备性、一致性、可追踪性、可见性、硬件系统无关性、软件系统无关性、可扩充性、公用性、模块性、清晰性、自描述性、简单性、结构性、产品文件完备性。评价准则的一定组合将反映某一软件质量要素，软件质量要素与评价准则间的关系如下图：



### 5.2.3 度量

根据软件的需求分析、概要设计、详细设计、实现、组装测试、确认测试和维护与使用七个阶段，制定了针对每一个阶段的问卷表，以此实现软件开发过程的质量控制。对于企业来说，不管是定制，还是外购软件后的二次开发，了解和监控软件开发过程每一个环节的进展情况、产品水平都是至关重要的，因为软件质量的高低，很大程度上取决于用户的参与程度。

这里需要说明几点：

(1) 对于不同类型的软件，系统软件、控制软件、管理软件、CAD 软件、教育软件、网络软件及不同规模的软件，对于质量要求、评价准则、度量问题的侧重点有所不同应加以区别。比如：

应用环境特性	需要考虑的要素
生存期长	可移植性、可维护性
实时系统	可靠性、效率
要在不同的环境中使用	可移植性
在银行系统中使用	可靠性、功能性

软件质量保证和评价活动有其不同的侧重点。在需求分析、概要设计、详细设计及其实现阶段，主要评价软件需求是否完备，设计是否完全反映了需求以及编码是否简洁、清晰。而且，每一个阶段都存在一份特定的度量工作表，它由特定的度量元组成，根据度量元的得分就可逐步得到度量准则，要素的得分，并在此基础上做出评价。这一点很适用于本学期的 litemall 系统的分析与重构。

(2) 对软件质量各阶段都进行度量的根本目的是以此控制成本、进度，改善软件开发的效率和质量，但是，目前大规模的软件公司在我国并不多，大多数软件开发单位都缺乏软件质量保证与软件质量评价的专门部门，因而客户可以委托专

业机构参与帮助软件质量控制与保证。

(3) 客户选择软件供应商、开发商，需要考察甲方是否建立起自己的软件质量度量和评价数据，数据库中是否存有与用户所在行业相关的软件，是否具有相关的开发经验。

5.2.4 问题严重程度描述

问题严重程度	描述	响应时间
严重	服务器宕机	1.5 小时内修复系统
严重	数据库异常	1.5 小时内修复系统
严重	用户非法获取既定权限之外的数据访问和修改能力	1.5 小时内修复系统
错误	事务运行结果与期望不一致	2 小时内改正错误
错误	事物运行长时间无响应	2 小时内改正错误
错误	页面显示内容不完整	2 小时内改正错误
错误	页面跳转错误	2 小时内改正错误
警告	事物运行时间超过预期	2 小时内找出原因，有条件则修复
警告	在高峰期系统能正确响应但速度极其缓慢	2 小时内找出原因，有条件则修复
信息	用户体验反馈	2 小时内回复