**食途**

**测试计划书**

**版本1.0**

# 食途项目计划书说明

| 项目 | 明细 |
| --- | --- |
| 文档标题 | 食途测试计划 |
| 整理人 | 闫博文 |
| 创建日期 | 20190517 |
| 最后修改日期 | 20190617 |

目录

1 概述

[1.1 编写目的 1](#_Toc11696028)

[1.2 项目背景 1](#_Toc11696029)

[1.3 项目质量目标 1](#_Toc11696030)

[1.4 测试提交成果 1](#_Toc11696031)

[1.5 预期读者 1](#_Toc11696032)

[1.6 参考资料 1](#_Toc11696033)

[2 测试环境 2](#_Toc11696034)

[2.1 系统架构 2](#_Toc11696035)

[2.2 软硬件环境要求 2](#_Toc11696036)

[2.2.1 测试环境 2](#_Toc11696037)

[2.2.2 网络 2](#_Toc11696038)

[3 测试规划 2](#_Toc11696039)

[3.1 测试策略 2](#_Toc11696040)

[3.2 人员、角色及职责 3](#_Toc11696041)

[4 测试策略 3](#_Toc11696042)

[4.1 系统框测试 3](#_Toc11696043)

[4.2 业务流程测试 3](#_Toc11696044)

[4.3 功能点测试 4](#_Toc11696045)

[4.4 UI界面测试 4](#_Toc11696046)

[4.5 性能测试 4](#_Toc11696047)

[4.6 兼容性测试 5](#_Toc11696048)

[4.7 安全测试 5](#_Toc11696049)

[5 测试进度安排 5](#_Toc11696050)

# 概述

## 编写目的

本测试计划书主要是是系统测试中的功能测试，对食途app中的功能进行测试，测试食途app中体现的功能是否正确实现，界面设计是否达到理想效果。本次测试大致需要2周时间，测试报告递交时间为第2周，预计整体软件测试结束时间为第8周。

## 项目背景

人们每天都需要吃饭，同时餐饮行业发展也比较火热。人们存在旺盛的提高生活水平改善生活的需求，而餐饮市场急需拓展销售渠道。如今电子商务的成熟以及人民群众早已熟悉网购，可以通过构建服务广大人民群体的电子商务产品来搭建这一桥梁，实现服务人民、繁荣市场的有益局面。

## 项目质量目标

从用户的角度出发，通过软件测试暴露软件中隐藏的错误和缺陷，以考虑是否可以接受该产品；

从软件开发者的角度来说，验证食途不存在错误，已正确的实现了用户的要求，确立人们对软件质量的信心。

## 测试提交成果

在测试阶段结束后，可提交的文档有：

《食途测试计划.doc》

《食途测试用例.doc》等

食途app是为广大群众提供的一个欣赏美食，制作美食，评价美食以及推荐美食的信息共享的娱乐平台。此次测试重点进行功能、界面、易用性、兼容性及性能测试。

本阶段测试重点针对移动设备客户端进行测试。主要测试范围如下：

食途功能：登录、视频播放、内容更新、商品更新、评论社区正常评论；

## 预期读者

测试执行人员，项目评审人员以及可能涉及到的客户。

## 参考资料

参考资料包括：

《食途需求规格说明书》

《食途原型界面》

# 测试环境

## 系统架构

CS（Client/Server）：客户端----服务器结构。C/S结构在技术上很成熟，它的主要特点是交互性强、具有安全的存取模式、网络通信量低、响应速度快、利于处理大量数据。因为客户端要负责绝大多数的业务逻辑和UI展示，又称为胖客户端。它充分利用两端硬件，将任务分配到Client 和Server两端，降低了系统的通讯开销。C/S结构的软件需要针对不同的操作系统系统开发不同版本的软件，加之产品的更新换代十分快，已经很难适应百台电脑以上局域网用户同时使用。C/S 架构是一种典型的两层架构，其客户端包含一个或多个在用户的电脑上运行的程序，而服务器端有两种，一种是数据库服务器端，客户端通过数据库连接访问服务器端的数据；另一种是Socket服务器端，服务器端的程序通过[Socket](http://blog.csdn.net/fightfaith/article/details/50055581" \t "_blank)与客户端的程序通信。

BS（Browser/Server）：浏览器----服务器结构，是目前应用系统的发展方向。BS是伴随着Internet技术的兴起，对C/S架构的改进，为了区别于传统的C/S 模式，特意称为B/S模式。在这种结构下，通过W3浏览器来进入工作界面，极少部分事务逻辑在前端（Browser）实现，主要事务逻辑在服务器端（Server）实现，形成[三层（3-tier）结构](http://blog.csdn.net/fightfaith/article/details/49529957" \t "_blank)。这样使得客户端电脑负荷大大简化（因此被称为瘦客户端），减轻了系统维护、升级的支出成本，降低了用户的总体成本（TCO）。

## 软硬件环境要求

### 测试环境

软件环境：客户端：Andriod（手机），JDK+Eclipes（pc）

中间件：JDK+Eclipes

硬件环境：测试服务器：尽量模拟真实运行环境

客户端：单元测试阶段使用PC的手机模拟程序，系统测试阶段使用安卓手机

### 网络

Wifi，4G，3G（应用接口对于网络异常的处理等）

# 测试规划

## 测试策略

* **阶段测试策略**：

具体测试分为：

1. 自测报告验证阶段：根据程序员提交的自测报告，验证自测报告的正确性，获得版本提交正确率，如果不符合，不启动测试工作
2. 需求验证阶段：根据测试计划中的需求描述，验证需求是否符合，提交是否符合需求实现类型的bug，获得需求符合度
3. 流程测试阶段：重点在于验证软件流程符合需求的要求
4. 模块功能测试阶段：重在在于验证软件的各项功能在正确的操作下是否正确
5. 即兴测试阶段：重点在于设计各种的用例，发现软件在使用中的各种错误
6. 回归测试阶段：重点在于验证bug修改的正确性，及关联的bug
7. 确认测试阶段：对于公司的某些项目，配合实施部门对软件产品进行确认测试测试工具

本次测试主要采用黑盒测试方法，并对其进行功能测试、易用性测试、界面测试。

* 提供的教务角色名为admin，密码为admin。
* 所使用数据库为postgresql，并配置相关测试环境。
* 所使用的缺陷管理工具为bugfree，并配置相关测试环境。

## 人员、角色及职责

|  |  |  |  |
| --- | --- | --- | --- |
| **阶段** | **区间测试任务** | **阶段完成标志** | **涉及人员** |
| 测试计划  （9月20日） | 阅读需求，编写测试计划（生成V1.0），分派测试任务。 | 提交《移动办公中间件测试计划》并通过评审。 | 王清顺 |
| 设计阶段  （9月20日—10月7日） | 熟悉系统需求并设计测试用例；测试中同步细化、更新用例。 | 提交《移动办公中间件测试用例设计》。 | 测试组成员 |
| 执行功能测试  （10月15日—\*月\*\*日） | 客户端功能测试，参照开发进度开展；  同步进行客户端与中间件交互测试；  同步进行BVT及回归测试。 | 达到通过准则 | 测试组成员 |
| 测试总结报告编写  （\*月\*\*日） |  | 提交《移动办公中间件测试总结报告》。 | 王清顺 |

# 测试策略

## 系统框测试

将经过集成测试的软件，作为计算机系统的一部分，与系统中其他部分结合起来，在实际运行环境下进行一系列严格有效的测试，以发现软件潜在的问题，保证系统的正常运行。

## 业务流程测试

　在业务系统的测试过程中，必须充分考虑基础和业务两项数据，其中基础数据主要采取客户所提供的真实数据，而业务数据则必须充分符合业务开展的基本流程。通常来说，所有典型业务操作便是业务流程流程可以采取场景方式进行书写。在针对一个业务流程进行设计时，在设计业务用例时，需要理清系统的业务流程，可以采用相关的辅助手段理清业务，例如画总体业务流程图以及分业务流程图等测试执行

## 功能点测试

|  |  |  |  |
| --- | --- | --- | --- |
| 功能 | 描述 | 模块 | 是否通过 |
| 登录相关功能 | **有效性原则：** | | |
| 1. 按回车键，自动切换至下一个文本框； | 系统登录页面 | 是 |
| 1. 登录口令输入后，按回车键或者点击1次登录按钮，可进行登录操作； | 系统登录页面 | 是 |
| 1. 点击重新登录进入登录界面时，自动定位到登录名的输入框； | 系统登录页面 | 是 |
| 1. 用户自己更改密码：默认显示当前登录名； | 系统登录页面 | 是 |
| …… | …… | …… |  |

## UI界面测试

界面测试的目的是检查用户界面的风格是否满足客户的要求, 是否通用，文字是否正常，页面是否美观，文字，图片是否结合完美，操作是否友好等等。常见的界面测试中的bug包括：

1. 重叠（Overlap）
2. 截断（Truncated）
3. 文字自动换行（Text Wrapping）
4. 缺失快捷键（lost shortcut）
5. 重复的快捷键（Duplicate shortcut）
6. 不合适的快捷键(Improper Shortkey)
7. 丢失文本(Miss text)
8. 没有对齐(missaligned)
9. Tab键顺序不准确(Tab Order)

## 性能测试

涉及到的性能测试主要有以下几点：

1. 并发数：包括系统用户数、在线用户数、并发用户数
2. 响应时间
3. 每秒事务通过数
4. 点击率
5. 吞吐量
6. 资源利用率等

## 兼容性测试

兼容性测试目前我关注的包括web兼容性测试和APP兼容性测试；

兼容测试包括：

　　（1）浏览器兼容测试：测试程序在不同浏览器上是否可以正常运行，功能能否正常使用；

　　（2）屏幕尺寸和分辨率兼容测试：测试程序在不同分辨率下能否正常显示；

　　（3）操作系统兼容测试：测试程序在不同的操作系统下面能否正常运行，功能能否正常使用，显示是否正确等；

　　（4）不同设备型号兼容测试：针对于APP，现在移动设备型号五花八门，主要测试APP在主流设备上能否正常运行，会不会出现崩溃的现象。

## 安全测试

主要考虑以下几个方面问题：

　用户认证安全的测试要考虑问题：

　　（1）明确区分系统中不同用户权限

　　（2）  系统中会不会出现用户冲突

　　（3）  系统会不会因用户的权限的改变造成混乱

　 （4）  用户登陆密码是否是可见、可复制

　　（5）是否可以通过绝对途径登陆系统

　　（6） 用户推出系统后是否删除了所有鉴权标记，是否可以使用后退键而不通过输入口令进入系统

　　系统网络安全的测试要考虑问题

　　（1）  测试采取的防护措施是否正确装配好，有关系统的补丁是否打上

　　（2）  模拟非授权攻击，看防护系统是否坚固

　 （3） 采用成熟的网络漏洞检查工具检查系统相关漏洞

　 （4）  采用各种木马检查工具检查系统木马情况

　　（5） 采用各种防外挂工具检查系统各组程序的客外挂漏洞

　　数据库安全考虑问题：

（1）系统数据是否机密

（2）系统数据的完整性

　　（3）系统数据可管理性

　　（4）  系统数据的独立性

　 （5）   系统数据可备份和恢复能力

# 测试进度安排

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **测试模块** | **计划开始时间** | **计划用时** | **实际开始时间** | **实际用时** | **测试人** |
| 登录功能 | 2019-1-6 | 0.5天 | 2019-1-6 | 0.5天 | 陈自力 |
| 注册功能 | 2019-1-6 | 0.5天 | 2019-1-6 | 0.5天 | 闫博文 |
| 社区帖子展示 | 2019-1-11 | 0.2天 | 2019-1-11 | 0.2天 | 陈自力 |
| 评论帖子 | 2019-1-11 | 0.2天 | 2019-1-11 | 0.2天 | 陈自力 |
| 好友信息展示功能 | 2019-1-11 | 0.2天 | 2019-1-11 | 0.2天 | 陈自力 |
| 与好友聊天功能 | 2019-1-12 | 0.5天 | 2019-1-12 | 0.5天 | 闫博文 |
| 个人信息修改功能 | 2019-1-12 | 0.2天 | 2019-1-12 | 0.2天 | 陈自力 |
| 发帖功能 | 2019-1-12 | 0.3天 | 2019-1-12 | 0.3天 | 陈自力 |
| 我的帖子显示功能 | 2019-1-13 | 0.2天 | 2019-1-13 | 0.2天 | 陈自力 |
| 我的资料 | 2019-1-13 | 0.2天 | 2019-1-16 | 0.2天 | 闫博文 |
| 首页管理 | 2019-1-25 | 1天 | 2019-1-25 | 1天 | 陈自力 |
| 管理员管理 | 2019-1-26 | 1天 | 2019-1-26 | 1天 | 陈自力 |
| 帖子管理 | 2019-1-27 | 1天 | 2019-1-27 | 1天 | 陈自力 |
| 文件管理 | 2019-1-28 | 1天 | 2019-1-28 | 1天 | 陈自力 |
| 关键字管理 | 2019-1-30 | 1天 | 2019-1-30 | 1天 | 陈自力 |
| 用户管理 | 2019-2-10 | 1天 | 2019-2-10 | 1天 | 陈自力 |
| 回归测试时间 | 开始时间 | 结束时间 | 实际开始时间 | 结束时间 | 用时 |
|  | 2019-2-26 | 2019-3-1 | 2019-2-26 | 2019-3-1 | 5天 |