# 服务场地预约APP

# 测试计划书

目录

[服务场地预约APP测试计划书 1](#_Toc6255163)

[一、简介 3](#_Toc6255164)

[1.1 目的 3](#_Toc6255165)

[1.2 背景 3](#_Toc6255166)

[1.3 范围 3](#_Toc6255167)

[二、测试参考文档和测试提交文档 4](#_Toc6255168)

[2.1 测试参考文档 4](#_Toc6255169)

[2.2 测试提交文档 4](#_Toc6255170)

[三、测试资源 4](#_Toc6255171)

[3.1 测试环境 4](#_Toc6255172)

[3.2 硬件资源 4](#_Toc6255173)

[3.3 软件资源 4](#_Toc6255174)

[四、要测试的功能以及不测试的功能 4](#_Toc6255175)

[五、测试策略 5](#_Toc6255176)

[5.1功能测试 5](#_Toc6255177)

[5.2 阶段测试策略 5](#_Toc6255178)

[5.3 Bug管理策略 5](#_Toc6255179)

[六、延迟测试标准及重新启动标准 6](#_Toc6255180)

[七、项目通过或失败准则 6](#_Toc6255181)

## 一、简介

### 1.1 目的

1.确定服务场地预约APP的信息和应测试的模块

2.列出测试需求

3.确定所需资源，对测试工作量进行估计

4.列出测试项目的可交付元素

### 1.2 背景

服务场地预约APP，适用于各大店铺的预约。由于目前许多店铺存在预约难，高峰期排队时间长，等待时间长等不方便因素，因此需要预约APP来协助店家和顾客进行沟通管理。最终的目的是使市民的生活更加方便快捷，商铺的管理更加方便高效。

### 1.3 范围

#### 1.3.1 总体描述

由于各种活动的相互影响和制约，系统的设计完成中可能存在某些错误。测试主要是对系统进行功能性以及安全性检查，及时发现程序功能逻辑以及安全方面的错误。提高产品的正确性和可靠性。

具体结合到系统操作，应该测试各个功能点的操作是否能正常交互完成

业务规则是否符合逻辑，准确。

事务准确性：确保事务正确完成，确保被取消的事务正确回滚

系统可靠性：是否存在较大的安全隐患，故障处理能力以及数据恢复能力

#### 1.3.2 详细描述

将测试主要分为：

完善文档：完善测试计划书，提前做好计划，使测试工作能够正常运行

框架整理：将selenium框架进行整理完善，能够有效投入功能测试中，显著提高功能性测试效率。

测试用例设计：合理设计测试用例，覆盖面广，缺陷定位度高，冗余度少。

功能测试实施：结合selenium框架以及测试用例，进行功能测试。

#### 1.3.3 相关风险

本次测试过程中，可能出现的风险如下：

1. 人员经验以及对APP的熟悉度
2. 对实际测试的执行情况
3. 人员调整等不可抗力因素导致测试未按期结束

规避：定期催促，加强员工培训。在可接受范围内推迟进度计划，提高工作效率。

## 二、测试参考文档和测试提交文档

### 2.1 测试参考文档

需求说明书，概要设计文档

### 2.2 测试提交文档

测试计划书，测试报告

## 三、测试资源

### 3.1 测试环境

Android studio

测试工具：testNG,selenium

### 3.2 硬件资源

CPU主频3.3GHz之上

内存4G及以上

硬盘256G以上

### 3.3 软件资源

Android studio

## 四、要测试的功能以及不测试的功能

要测试的功能：用户模块的注册登录、首页、浏览店铺、搜索店铺、预约及取消预约场地、个人中心、查看场地信息；商家模块的注册入驻、场地管理、预约处理、商家中心；管理员模块中的商家审核、广告管理、相似店铺推荐管理、分析并管理数据

不测试的功能：压力测试，集成测试

## 五、测试策略

### 5.1功能测试

目标：确保软件需求说明书中要求的各个功能模块全部按需求实现

测试方法和技术：按照测试需求、通过准则、测试用例，采用黑盒测试技术，核实：

1. 在使用合法数据时得到正确的
2. 在使用非法数据时显示相应的错误信息和警告信息
3. 各个功能模块的功能都得到了正确的应用

完成标准：1.计划的测试已全部执行

2.发现的缺陷修复率达到通过准则要求

3.不能实现的功能测试需求项开发组给出了合理的说明或做了需求变更

4.所做修改已达到需求所定义的

### 5.2 阶段测试策略

1、单元测试阶段：由开发人员针对个人负责的单元或模块进行单元测试。通过本阶段后进行下一阶段。

2、冒烟测试阶段：验证系统第一版本的关键功能，目的验证该系统版本是否可用，是否能进行具体功能细测，若出现过多限制后续测试的阻塞级别bug，则需要请开发组发布新版本。

3、初测/细测阶段：重点验证软件功能是否满足需求，该阶段由测试人员完成。测试成员对功能点进行独立测试，并编写简易的测试用例。同时，在该用例执行中记录该用例执行的状态（通过 / 未执行 / bug ID）

### 5.3 Bug管理策略

#### 5.3.1 错误优先级

立刻级别：必须立即修改；

紧急级别：立即修改（最长时间不得超过2天）；

高级级别：在投入正式运行前修改；

普通级别：在投入正式运行前大部分需修改，未修改的缺陷需进行讨论；

低级别：可延期修改或不修改。

#### 5.3.2 严重程度分类

|  |  |
| --- | --- |
| **严重程度** | **现象** |
| 严重 | ①   用户需求未实现（影响到用户完成业务）；  ②   用户需求实现错误（影响到用户完成业务）；  ③   导致被测软件响应明显很慢（假死）、死机、非法退出、崩溃；  ④   导致后台数据受损或丢失 |
| 中等 | ①用户需求实现错误（不影响用户完成业务、用户使用不频繁)；  ② 用户操作过程中系统出现异常报错，但不影响系统功能的使用。  ③用户使用不频繁的功能，响应时间超出忍耐限度；  ④ UI上存在错误引导用户的信息。 |
| 轻微 | ①   UI控件不符合界面规范。  ②   影响UI友好性 |

## 六、延迟测试标准及重新启动标准

延迟测试标准：测试人员未能及时到位以及因技术问题导致测试工作的进度影响

重新启动标准：确保测试结束前人员的稳定性，以及确保解决技术性问题之后

## 七、项目通过或失败准则

测试用例全部执行并通过，覆盖95%以上，缺陷修复90%以上