停哪儿测试方案

目录

[**1.1编写目的** 2](#_Toc61097617)

[**1.2项目背景** 3](#_Toc61097618)

[**1.4测试参考文档** 3](#_Toc61097619)

[**1.5测试提交文档** 3](#_Toc61097620)

[**1.6术语和缩略语** 4](#_Toc61097621)

[**2.1测试配置要求** 4](#_Toc61097622)

[**2.2测试方法** 4](#_Toc61097623)

[**2.3测试数据** 5](#_Toc61097624)

[**2.4测试策略** 5](#_Toc61097625)

[**2.5测试资源** 9](#_Toc61097626)

[**2.6测试阶段及范围** 9](#_Toc61097627)

[**2.7通过测试的标准** 9](#_Toc61097628)

[**3.1概述** 10](#_Toc61097629)

[**4测试用例** 10](#_Toc61097630)

[**5.1文本输入框** 10](#_Toc61097631)

[**5.2下拉列表** 11](#_Toc61097632)

[**5.3增加数据** 11](#_Toc61097633)

[**5.4修改数据** 11](#_Toc61097634)

[**5.5删除数据** 11](#_Toc61097635)

[**5.6查询数据** 12](#_Toc61097636)

[**5.7数据导入导出** 12](#_Toc61097637)

[**5.8数据接入与处理** 12](#_Toc61097638)

[**5.9其他** 12](#_Toc61097639)

**1.1编写目的**

为软件开发项目管理者、软件工程师、系统维护工程师、测试工程师提供关于项目系统整体功能和性能的测试指导，同时也是用户确定软件是否完整测试的重要依据。

**1.2项目背景**

随着汽车工业的发展以及人们生活水平的提高，驾驶员数量急速上升，私家车越来越多，不但给城市交通带来了压力，而且人们对停车位的渴求日益增加。“停车难”成为目前城市管理之中的关键问题。许多城市出现不同程度上的停车位紧缺现象。对于广大车主来说一款贴合用户需求，功能强大的智能找车位软件是目前主要需求体现。

车辆的增多势必会要求增加停车场的数量以及场内车位的数量,传统的停车场管理系统已经不能满足当前规模正在不断扩大的停车场的需求。目前，传统停车场的一些不足主要表现在:车辆在进入停车场时需要停车刷卡，如果停车数量多时会出现停车场入口堵车现象;车主进入停车场后，不能准确而快速地找到自己满意的停车位，既浪费了车主的时间也降低了停车场的运行效率;场内车辆多给管理带来不便，车辆的安全问题不能得到保障;出口计费系统还是需要人工收费并给予票据，一定程度上增加了车主的停车耗费时间。因此，提高传统停车场管理系统的自动化程度和智能化程度已成为必然趋势。

**1.3测试目标**

在用户使用软件之前，尽可能的发现软件中存在的错误和不合理之处，排除软件中潜在的错误，最终把高质量的软件系统交付给用户。系统的测试覆盖范围：功能、性能、UI、安全性、兼容性、容量

**1.4测试参考文档**

测试开发-第一阶段 –蜗牛学院

测试开发-第二阶段 –蜗牛学院

测试开发-第三阶段 –蜗牛学院

**1.5测试提交文档**

软件测试方案

软件测试用例

软件测试报告

**1.6术语和缩略语**

本文使用的术语、定义

软件测试:软件测试就是利用测试工具按照测试方案和流程对产品进行功能和性能测试，甚至根据需要编写不同的测试工具，设计和维护测试系统，对测试方案可能出现的问题进行分析和评估。

系统测试:系统测试是通过与系统的需求规格作比较，发现软件与系统需求规格不相符合或与之矛盾的地方。它将通过确认测试的软件，作为整个基于计算机系统的一个元素，与计算机硬件，外设，某些支持软件，数据和人眼等其他系统元素结合起来，在实际使用环境下，对计算机系统的测试。

功能测试:黑盒测试是基于系统需求规格，在不知系统或组件的内部结构的情况下进行的测试，通常又将黑盒测试叫做基于规格的测试，输入输出测试，功能测试或数据驱动测试，是基于用户观点出发的测试。主要是验证功能是否符合需求，包括原定功能的校验，是否有冗余功能，遗漏功能。

性能测试: 性能测试关注的是系统的整体，它和通常说的强度测试，压力测试，负载测试有密切关系。

**2.1测试配置要求**

软件：

运行系统：

服务器：

数据库：

**2.2测试方法**

系统的功能测试选用了手工测试，运用黑盒测试中的等价类划分、边界值分析、错误推测、因果图法。系统UI方面的测试包括：易用性测试、规范性测试、帮助设施测试、合理性测试、美观与协调性测试、独特性测试、快捷方式组合测试。系统的安全性、兼容性、安装与反安装、配置测试也是手工测试。单元测试采用的方法是白盒测试，功能测试采用黑盒测试。

**2.3测试数据**

测试数据主要依照《XXX》文档，参考《XXX》文档中规定的运行限制，设计测试用例，作为XX平台的测试数据。

**2.4测试策略**

**2.4.1单元测试**

首先依照系统、子系统和模块进行划分，但最终的单元必须是功能模块，或面向对象过程中的若干个类，单元测试是对功能模块进行正确性检验的测试工作，也是后续测试的基础。目的在于发现各模块内部可能存在的各种差错，因此需要从程序的内部结构出发设计测试用例，着重考虑以下五个方面：

（1） 模块接口：对所测模块的数据流进行测试；

（2） 局部数据结构：检查不正确或不一致的数据类型说明、使用尚未赋值或尚未初始化的变量、错误的初始值或缺省值。

（3） 路径：虽然不可能做到穷举测试，但要设计测试用例查找由于不正确的计算（包括算法错、表达式的符号表示不正确、运算精度不够等）、不正确的比较或不正常的控制流（包括不同数据类型量的相互比较、不适当地修改了循环变量、错误的或不可能的循环终止条件等）而导致的错误。

（4） 错误处理：检查模块有没有对于常见错误的条件设计比较完善的错误处理功能，保证其逻辑上的正确性。

（5） 边界：注意设计数据流、控制流中刚好等于、大于或小于确定的比较值的用例。

**2.4.2集成测试**

集成测试也叫组装测试或联合测试。通常，在单元测试的基础上需要将所有的模块按照设计要求组装成系统，这时需要考虑的问题如下：

（1） 把各个模块连接起来，模块接口的数据是否会丢失；

（2） 一个模块的功能是否会对另一个模块的功能产生不利的影响；

（3） 各个子功能组合起来，能否达到预期要求的父功能；

（4） 全局数据结构是否有问题；

（5） 单元模块的误差累积起来，是否会放大，从而达到不能接受的程度。我们在组装时可参考采用一次性组装方式或增值式组装方式；

**2.4.3系统测试**

系统测试目的是在于验证软件的功能和性能及其他特性是否与用户的要求一致，主要是以下类型的测试；

1. 功能测试：验证系统功能是否符合其需求规格说明书，核实系统功能上是否完整，没有冗余和遗漏的功能。功能测试详细介绍如下表：

|  |  |
| --- | --- |
| 测试范围 | 验证数据精确度、数据类型、业务功能等相关方面的正确性 |
| 测试目标 | 核实所有功能均已正常实现、即是否与需求一致 |
| 技术 | 采用黑盒测试、边界测试、等价类划分测试方法 |
| 工具和方法 | 手工测试 |
| 开始标准 | 开发阶段对应的功能完成并且测试用例设计完成 |
| 完成标准 | 测试用例通过并且高级缺陷全部修复 |
| 需考虑特殊事项 | 暂无 |

（2） 用户界面测试：测试用户界面是否具有导航性、美观性、行业或公司的规范性、是否满足设计中要求的执行功能、详细介绍如下表UI测试：

其中，Web测试通用方法可以参考《Web测试检查点总结》

|  |  |
| --- | --- |
| 测试范围 | 页面布局的校验 |
| 测试目标 | 核实各个页面（颜色、字体、提示信息、图标等）均与需求保持一致或到达可接受标准，保证用户界面的友好性、易操作性 |
| 技术 | Web测试通用方法 |
| 工具和方法 | 手工测试、目测 |
| 开始标准 | 界面开发完成 |
| 完成标准 | UI符合可接受标准，保证用户体验 |
| 需考虑特殊事项 |  |

（3） 性能测试：测试相应时间、事务处理效率和其他时间敏感的问题。性能测试介绍如下表所示：

|  |  |
| --- | --- |
| 测试范围 | 多用户长时间在线时性能方面的测试 |
| 测试目标 | 核实系统在大流量的数据与多用户操作时性能的稳定性 |
| 技术 | 手工测试、自动化测试 |
| 开始标准 | 自动化测试脚本设计并评审通过 |
| 完成标准 | 系统满足用户需求中所有的性能要求 |
| 需考虑特殊事项 |  |

（4）兼容性测试：测试软件在不同平台上使用的兼容性。兼容性测试详细介绍如下表所示：

所示：

|  |  |
| --- | --- |
| 测试范围 | 使用不同版本的浏览器、分辨率、操作系统进行测试 |
| 测试目标 | 核实系统在不同条件下的运行情况 |
| 技术 | 黑盒测试 |
| 工具与方法 | 手工测试 |
| 开始标准 | 拿到项目 |
| 完成标准 | 不同浏览器、分辨率、操作系统下系统功能正常实现 |
| 需考虑特殊事项 |  |

（5）安全性测试：测试软件系统对非法侵入的防范能力。安全测试详细介绍如下表：所示：

|  |  |
| --- | --- |
| 测试范围 | 用户、管理员的密码安全、权限 |
| 测试目标 | 用户、管理员的密码管理、应用程序级别的安全性、核实用户只能操作其所拥有的权限操作 |
| 技术 | 代码包或者非法工具 |
| 工具与方法 | 手工测试 |
| 开始标准 | 功能测试完成 |
| 完成标准 | 执行各种非法操作无安全漏洞且系统可以正常运行并使用 |
| 需考虑特殊事项 |  |

（6）配置测试：测试在不同网络、服务器、工作站的不同软硬件配置条件下，软件系统的质量，详细说明如下表所示：所示：

|  |  |
| --- | --- |
| 测试范围 | 不同网络、服务器、不同的硬件配置 |
| 测试目标 | 核实系统在不同软硬件配置条件下的质量是否达到标准 |
| 技术 | 黑盒测试 |
| 工具与方法 | 手工测试 |
| 开始标准 | 系统开发完成后 |
| 完成标准 | 达到相关标准 |
| 需考虑特殊事项 | 软硬件设备问题 |

（7）回归测试详细介绍如下表所示：

所示：

|  |  |
| --- | --- |
| 测试范围 | 所有功能、用户界面、兼容性、安全性测试 |
| 测试目标 | 根据时间执行全部或主页测试 |
| 技术 | 黑盒 |
| 工具与方法 | 手工与自动化测试 |
| 开始标准 | 软件出现新的版本 |
| 完成标准 | 95%用例通过且没有出现新的bug |
| 需考虑特殊事项 |  |

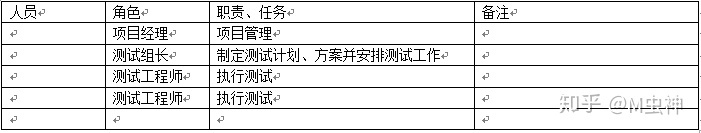
**2.4.4验收测试**

用户新增或修改内容，以及用户反馈问题确认：

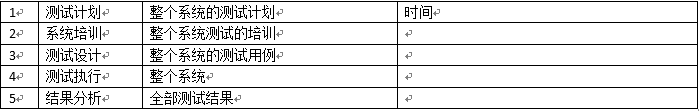
所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试项 | 测试方法 | 预计结果 | 实际结果 | 结论 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**2.5测试资源**



**2.6测试阶段及范围**



**2.7通过测试的标准**

一般有“基于测试用例” 和 “基于缺陷密度” 两种评比准则，在这里我们采用前者。

（1） 功能性测试用例通过率达到100%

（2） 非功能性测试用例通过率达到95%

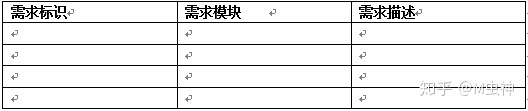
（3） 不存在高优先级的问题

备选通过方法：根据实际情况由软件开发部门的经理，项目经理和测试负责人共同讨论确定本测试阶段是否结束。（详细的系统测试通过标准可参考《系统测试各阶段准入准出规则》）

**3.1概述**

XX平台简称XX，总共有XX大功能模块分别是：XXX 。每个模块的需求模块如下表所示：

本文档描述的数据接入需求模块，需求标识及需求描述如表：



**4测试用例**

测试用例文档附件粘贴即可。（也可注明测试用例访问位置）

**5关注点**

**5.1文本输入框**

（1）检查空数据；

（2）检查过长数据（超出空间本身的长度和数据库中改字段所允许的长度）；

（3）检查特殊字符，尤其是数据库中不允许的字符，甚至回车字符、空格字符等；

（4）检查字符类型，比如应该输入数字的文本框输入英文字符；

（5）中文字符的处理；

（6）对于日期时间型数据，检查格式正确性以及时间日期的合理性。比如开始时间不能晚于结束时间等；

**5.2下拉列表**

（1）列表数据是否正确、完整；

（2）下拉列表与其他空间的联动关系；

（3）是否允许多选；

**5.3增加数据**

（1）数据个数的上限；

（2）重复数据处理，尤其是键值的重复；

（3）相关表格的更新；

（4）检查多次使用back键的情况，在有back的地方，back回到原页面，再back重复多次，看是否会出错；

**5.4修改数据**

（1）不能破坏数据库数据的关联和完整；

（2）重复数据处理，尤其是键值的重复；

（3）修改登录用户本身信息时对系统的影响；

（4）修改正在使用的数据；

（5）检查多次使用back键的情况，在有back的地方，back，回到原页面，再back，重复多次，看是否会出错。

**5.5删除数据**

（1）不能破坏数据库数据的关联和完整；

（2）删除正在使用的数据；

（3）删除登录用户本身；

**5.6查询数据**

（1）多条件组合查询的正确性；

（2）多次连续查询正确性；

**5.7数据导入导出**

（1）导入数据格式要求不应太严格，提示明确；

（2）导出数据不应乱码；

**5.8数据接入与处理**

（1）数据接入方式是否全部可用，数据是否能正确接入；

（2）数据处理方式是否全部可行；

（3）数据的动态监测是否正确无误；

**5.9其他**

（1）对网络故障的提示；

（2）同一用户多次登录；

（3）内存使用情况；

**后记**

通常来讲，每个公司有自己的文档规范以及必须遵守的行业标准规范，大体的方案架构可以按照公司内部标准，其他细节需要根据被测系统的特征来适当调整。同时如果是外包性质的项目还需要考虑到客户方的标准及交付文档规范，如相关人员与完成工作时间及范围、灰度测试环境上报、用户测试准入条件等等。另外，如果系统较为庞大 ，系统测试方案中也只体现相关整体架构 ，具体的专项测试还会以附件形式重新设计完整方案。