# 公车管理服务平台测试报告

目录

[公车管理服务平台测试报告 1](#_Toc426931251)

[1. 引言 3](#_Toc426931252)

[1.1 编写目的 3](#_Toc426931253)

[1.2 项目背景 3](#_Toc426931254)

[1.3 术语解释 3](#_Toc426931255)

[1.4 参考资料 4](#_Toc426931256)

[2. 测试概要 4](#_Toc426931257)

[2.1 系统简介 4](#_Toc426931258)

[2.2 测试计划描述 4](#_Toc426931259)

[2.2.1 单元测试 4](#_Toc426931260)

[2.2.2 功能测试 5](#_Toc426931261)

[2.2.3 压力测试 5](#_Toc426931262)

[2.3 测试环境 5](#_Toc426931263)

[2.3.1 单机测试环境 5](#_Toc426931264)

[2.3.2 集群测试环境 7](#_Toc426931265)

[3. 测试结果 8](#_Toc426931266)

[3.1 测试执行情况 8](#_Toc426931267)

[3.2 单元测试报告 8](#_Toc426931268)

[3.2.1 测试用例覆盖率 9](#_Toc426931269)

[3.2.2 测试用例测试结果 9](#_Toc426931270)

[3.3 系统功能测试报告 11](#_Toc426931271)

[3.4 系统性能测试报告 15](#_Toc426931272)

[3.5 不间断运行测试报告 16](#_Toc426931273)

[3.6 可靠性测试报告 17](#_Toc426931274)

[3.7 集群配置测试报告 18](#_Toc426931275)

[3.8 安全性测试报告 18](#_Toc426931276)

[3.9 可维护性测试报告 19](#_Toc426931277)

[4. 测试结论 19](#_Toc426931278)

[4.1 功能性 19](#_Toc426931279)

[4.2 可靠性 19](#_Toc426931280)

[4.3 兼容性 20](#_Toc426931281)

[4.4 安全性 20](#_Toc426931282)

# 引言

## 编写目的

本测试报告为公车管理服务平台的测试报告，目的在于对系统开发和实施后的结果进行测试以及测试结果分析，发现系统中存在的问题，描述系统是否符合项目需求说明书中规定的功能和性能要求。

预期参考人员包括用户、测试人员、开发人员、项目管理者、其他质量管理人员和需要阅读本报告的高层领导。

## 项目背景

## 术语解释

系统测试：按照需求规格说明对系统整体功能进行的测试。

功能测试：测试软件各个功能模块是否完成，功能是否正常，逻辑是否正确。

系统测试分析：对测试的结果进行分析，形成报告，便于交流和保存。

严重bug：出现以下缺陷，测试定义为严重bug

1. 系统无响应，处于死机状态，需要其他人工修复系统才可复原。
2. 点击某个菜单后出现“The page cannot be displayed”或者返回异常错误。
3. 进行某个操作（增加、修改、删除等）后，出现“The page cannot be displayed”或者返回异常错误
4. 当对必填字段进行校验时，未输入必输字段，出现“The page cannot be displayed” 或者返回异常错误
5. 系统定义不能重复的字段输入重复数据后，出现“The page cannot be displayed”或者返回异常错误

模块：系统的基本组成部分，模块是一个可独立运行的系统，内部的代码高度耦合，一般采用WebAPI方式与其他模块通信。

数据连接层：即系统与数据库连接的中间层次

服务层：对数据连接层进行进一步封装的层次

角色：一组具有相同权限的用户集合。

权限：执行某个功能，或操作某个模块、某个子系统的钥匙。

用户：操作系统的人。

## 参考资料

1. GB/T 8566—2001 《信息技术 软件生存期过程》(原计算机软件开发规范)
2. GB/T 8567—1988 《计算机软件产品开发文件编制指南》
3. GB/T 11457—1995 《软件工程术语》
4. GB/T 12504—1990 《计算机软件质量保证计划规范》
5. GB/T 12505—1990 《计算机软件配置管理计划规范》

# 测试概要

## 系统简介

公车管理服务平台是对企事业单位公车进行管理的多租户云平台。

## 测试计划描述

系统测试包括系统的易用性、可靠性、安全性、可维护性进行测试，整个系统集成后提供服务的能力，还包括系统服务性能测试、疲劳测试（不间断运行）。

本测试主要分为以下几个方面；

### 单元测试

单元测试使用junit单元测试工具运行程序中编写的单元测试。主要测试包括模型层和控制层的逻辑测试，如数据的查询，添加，修改，删除等。

### 功能测试

功能测试即对照项目需求文档，逐一测试系统中功能是否完成，逻辑是否正确。当该功能确认完成，逻辑正确时即表示功能通过测试。

### 性能测试

性能测试主反应系统反应时间，CPU使用率，占用内存大小，系统反应速度等硬性指标。

### 连接数测试

连接数测试主要测试系统服务器同时可以支持多少个用户使用。

### 压力测试

压力测试分为两个方面：1. Obd在线测试；即测试系统是否能处理同时在线10000辆车的情况，是否会出现数据处理阻塞的情况，数据更新是否及时。2. 系统在连接数较高时是否能及时处理用户请求，服务器负载情况如何。

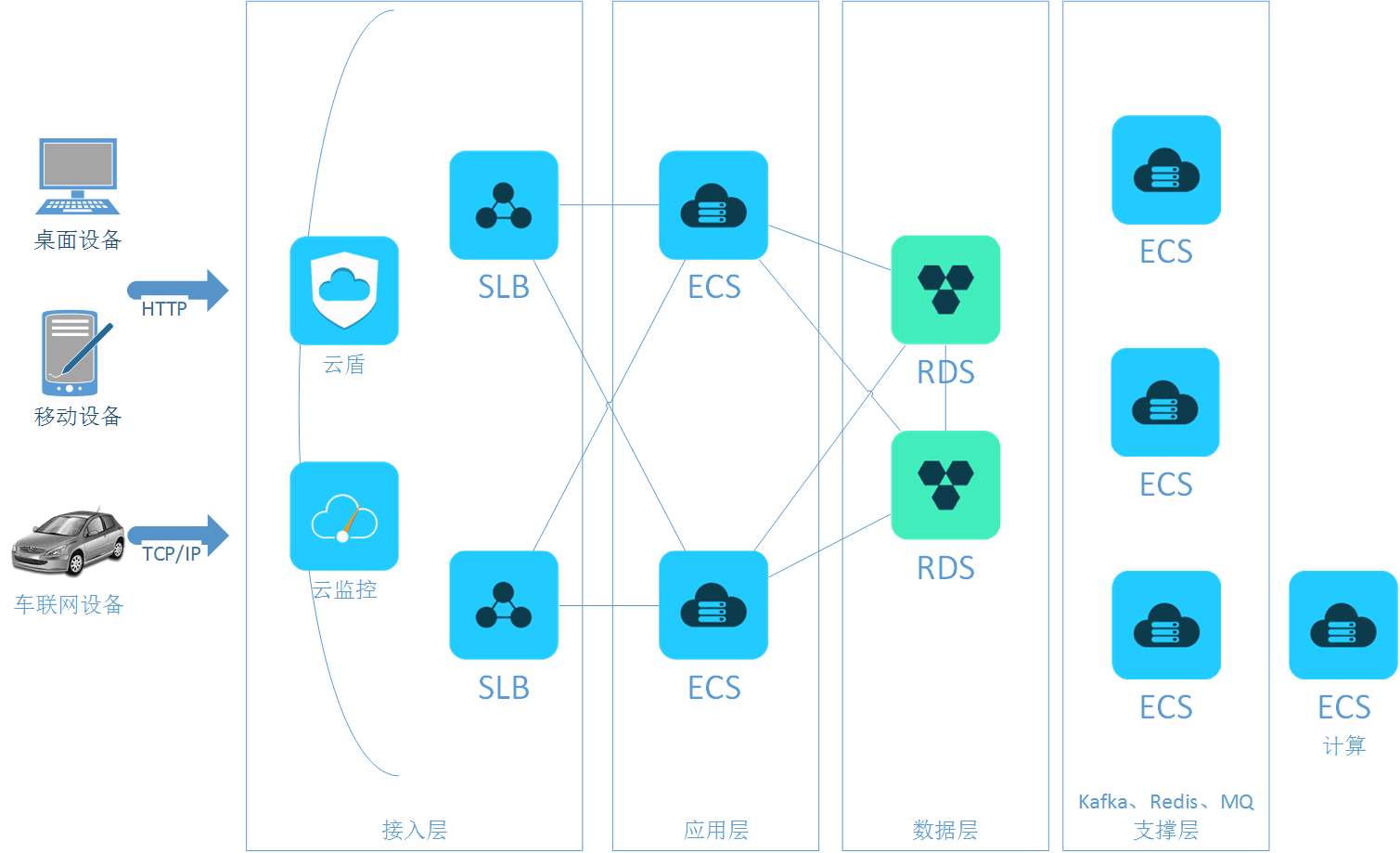
## 测试环境

单机测试环境用于测试系统资源使用情况，方便地从数值上了解系统对资源的占用情况，使用效率等信息。区别于集群测试，集群测试主要测试多个集群、多个系统之间的协作、通讯情况。

### 单机测试环境

|  |  |  |  |
| --- | --- | --- | --- |
| 条目名称 | 类别名称 | 环境属性 | 环境数值 |
| 服务器 | 硬件系统 | OS 名称 | Debian 8 |
| OS 版本 | 3.16.0-4-amd64 |
| OS 制造商 | Debian |
| 系统类型 | x64-based PC |
| 处理器 | Intel(R) Xeon(R) CPU E7- 4807 @ 1.87GHz |
| BIOS 版本 | VMware |
| 系统区域设置 | 中文(中国) |
| 服务器 | 硬件系统 | 时区 | (UTC+08:00)北京，重庆，香港特别行政区，乌鲁木齐 |
| 物理内存总量 | 16GB |
| 系统软件 | 缓存服务程序 | Redis 2.8.19 , 64 bit |
| JAVA虚拟机 | java version 1.7.0\_79, 32bit |
| Web服务器 | apache-tomcat-7.0.62, 32bit |
| 数据库 | MySQL5.5 32bit |
| 队列服务程序 | Kafka 2.11, 32bit |
| 客户端 | 硬件系统 | OS 名称 | Microsoft Windows 8 |
| 系统类型 | x64-based PC |
| 处理器 | Intel64 2494Mhz |
| 物理内存总量 | 3,987 MB |
| 系统软件 | JAVA虚拟机 | java version 1.7.0\_79, 32bit |
| 浏览器 | Chrome 44.0.2403.125 |
| 浏览器兼容性 | IETester |
| 移动端 | Android | 制造商 | ASUS |
| 型号 | Nexus 7 |
| 系统版本 | Android 5.1.1 |
| 内核版本 | 3.4.0-gfe3bd61 [android-build@vped12.mtv.corp.google.com#1](mailto:android-build@vped12.mtv.corp.google.com#1) Wed May 27 22:29:37 UTC 2015mtv.corp.google.com#1 |
| iOS | 制造商 | Apple Inc. |
| 型号 | iPhone 6p |
| 系统版本 | 8.4 |
| 运营商 | 中国移动（CMCC） |

### 集群测试环境



利用阿里云平台进行负载均衡和集群配置测试。集群架构图如上图所示。包含两台ECS，即业务服务器，两台服务器均安装了网页服务、业务服务等多个模块。阿里RDS数据库服务及其他相关的支持业务。

阿里云部署与功能测试采用不同的硬件架构，使用了多个配置相同的ECS实例以及SLB和RDS服务。具体配置如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 条目名称 | 类别名称 | 环境属性 | 环境数值 |
| ECS  服务器 | 硬件系统 | OS 名称 | CentOS |
| OS 版本 | 6.5 x64 |
| OS 制造商 | CentOS |
| 系统类型 | x64-based PC |
| 系统区域设置 | 中文(中国) |
| 硬件系统 | 时区 | (UTC+08:00)北京，重庆，香港特别行政区，乌鲁木齐 |
| 物理内存总量 | 4GB |
| 系统软件 | 缓存服务程序 | Redis 2.8.19 , 64 bit |
| JAVA虚拟机 | java version 1.7.0\_79, 32bit |
| Web服务器 | apache-tomcat-7.0.62, 32bit |
| 数据库 | MySQL5.5 32bit |
| 队列服务程序 | Kafka 2.11, 32bit |
| RDS | 数据库 | 数据库类型 | MySQL |
| 版本 | 5.5 |
| 客户端 | 硬件系统 | OS 名称 | Microsoft Windows 8 |
| 系统类型 | x64-based PC |
| 处理器 | Intel64 2494Mhz |
| 物理内存总量 | 3,987 MB |
| 系统软件 | JAVA虚拟机 | java version 1.7.0\_79, 32bit |
| 浏览器 | Chrome 44.0.2403.125 |
| 浏览器兼容性 | IETester |

# 测试结果

## 测试执行情况

系统测试时间：2015/08—2015/08

## 单元测试报告

针对系统中制定的接口（参考《公车管理服务平台接口文档》）和数据连接层，共编写测试用例。

### 测试用例覆盖率

此次测试，所有测试用例都是在中文界面下执行，未在英文界面下执行，测试不包括英文界面下的测试，也不包括正对英文翻译的测试。 此次测试，部分页面需求描述无明确的定义，对输入限制无详细定义，无明确的测试依据，在测试过程中，测试是根据输入字段含义，测试人员理解，以及和项目经理，开发人员沟通获得测试依据，无法保证测试依据的正确性和完整性，因此，没有进行完整的，正确的无效数据的测试。下面为此次测试测试用例覆盖率分析图：

其中，性能和负责测试未写测试用例，采用独立的实际测试方式测试。

### 测试用例测试结果

通过运行测试用例，检查整改系统的运行结果。测试结果如下所示：

|  |  |  |
| --- | --- | --- |
| 测试用例 | 测试过程描述 | 测试结果 |
| BUControllerTest | 测试组织管理相关接口的可用性，包括增删改查，管理架构，获取组织架构树等功能 | 通过 |
| DriverControllerTest | 测试驾驶员相关接口可用性以及该接口处理部分特殊输入输出的健壮性。包括增删改查的测试。 | 通过 |
| OAuthLoginTest | 测试系统登录系统的安全性，检查权限认证的可用性 | 通过 |
| VehicleControllerTest | 测试车辆管理相关接口。包括增删改查的测试。 | 通过 |
| VehicleOrderControllerTest | 测试订单管理相关接口的可用性，包括订单建立，订单流转，订单修改，订单查询等相关功能 | 通过 |
| UserOrderDAOTest | 用户订单在数据连接层的测试，保证数据与数据库连接的可靠性。 | 通过 |
| OrderFlowServiceTest | 订单流程管理服务测试，包括订单流程的获取，订单流程的修改等 | 通过 |
| OrdersServiceTest | 订单服务测试，即在服务层对订单进行测试，保证订单服务可用 | 通过 |
| VehicleServiceTest | 车辆管理在服务层的测试，与车辆管理接口测试对应。 | 通过 |
| BusinessUnitServiceTest | 部门管理在服务层的测试，与部门管理接口对应 | 通过 |
| FlowControllerTest | 订单流程接口测试，对各部门订单流程的修改和添加。 | 通过 |
| OBDControllerTest | Obd设备相关接口测试，包括设备的注册，绑定，状态修改等。 | 通过 |
| OrgControllerTest | 组织架构管理接口测试。包括增删改查的测试。 | 通过 |
| PushControllerTest | 移动端推送接口的测试，通过该测试确认移动端是否能收到消息 | 通过 |
| ReminderControllerTest | 提醒接口测试，测试提醒系统是否工作正常，相关数据是否能入库。 | 通过 |
| StatisticControllerTest | 统计接口测试。测试统计接口是否可用，是否有逻辑错误。 | 通过 |
| SystemControllerTest | 系统接口测试。测试该接口是否工作正常 | 通过 |
| UserControllerTest | 用户接口测试。测试用户的登录，用户信息的获取，用户信息的修改，添加删除等相关操作。 | 通过 |
| VehicleBrandsControllerTest | 车辆型号接口测试。测试是否能正常或取得到车辆的型号类型列表。 | 通过 |
| VehicleManagerControllerTest | 车辆管理接口测试。测试车辆的注册，修改，相关信息的录入接口是否正常，可用。 | 通过 |
| VehicleOrderControllerTest | 车辆订单接口测试。测试车辆订单是否正常。 | 通过 |
| OrderFlowDAOTest | 订单流程在数据连接层的测试，测试订单流程是否符合要求，流转是否正常。出现异常是否能正确处理。 | 通过 |
| OrgDAOTest | 组织架构管理在数据连接层的测试，测试有关组织架构管理的数据是否能写入数据库，是否能正常从数据库读取。 | 通过 |
| UserOrderDAOTest | 用户订单在数据连接层的测试，测试数据是否能写入数据库，是否能正常从数据库读取。 | 通过 |
| BusinessUnitServiceTest | 组织架构在数据连接层的测试，测试数据是否能写入数据库，是否能正常从数据库读取。 | 通过 |
| OrderFlowServiceTest | 订单流转在数据连接层的测试，测试数据是否能写入数据库，是否能正常从数据库读取。 | 通过 |
| OrdersServiceTest | 订单在服务层的测试，测试订单的查询，读取是否正常。 | 通过 |
| PushServiceTest | 推送服务在服务层的测试，确定推送服务在服务层可正常工作。 | 通过 |
| VehicleServiceTest | 车辆管理在服务层的测试，以确定车辆管理的相关逻辑正确 | 通过 |
| ConverterTest | 坐标转换测试，测试百度坐标和GPS坐标之间的互转是否正确。 | 通过 |
| UUIDGeneratorTest | 测试UUID获取是否正确 | 通过 |
| CheShouYeApiTest | 测试第三方违章提供商车首页的服务是否正常可用，编写的接口是否正确 | 通过 |
| SaasApiControllerTest | 测试第三方信息提供服务接口编写是否正确，是否符合需要的数据格式 | 通过 |

## 系统功能测试报告

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1** | **调度管理** | | | **测试结果** |
| 1 | 审批流程 | 订单流程 | 用户申请用车、领导审批、车管员派车的流程管控 | 通过 |
| 2 | 订单信息 | 审批中涉及的其他相关订单功能 | 通过 |
| 3 | 轨迹回放 | 订单运行过程记录，对记录的及时搜索回放 | 通过 |
| 4 | 监控管理 | 车辆运行监控 | 记录车辆运行过程中的各项指标，包括位置、速度、油量 | 通过 |
| 5 | 时空异常 | 实时监控车辆异常用车状态：所处位置，使用时段 | 通过 |
| 6 | 非调度用车 | 车辆不在调度使用中时发出提醒 | 通过 |
| 7 | 事故提醒 | 车辆运行中发生事故及时提醒 | 通过 |
| 8 | 费用异常 | 车辆使用过程中产生的各项税费异常 | 通过 |
| 9 | 其他提醒 | 年审到期、保险、年限等定时提醒 | 通过 |
| **2** | **车辆管理** | | |  |
| 1 | 基本管理 | 车辆注册 | 车辆基本信息的录入 | 通过 |
| 2 | 车辆修改 | 车辆各项参数的变动 | 通过 |
| 3 | 处置记录 | 车辆的各项处置记录 | 通过 |
| 4 | 行驶过程费用 | 油耗记录 | 记录车辆使用过程中的油耗 | 通过 |
| 5 | 通行费 | 记录使用过程中各项通行费用 | 通过 |
| 6 | 交通罚款 | 记录交通罚款 | 通过 |
| 7 | 固定费用 | 年审费用 | 年审记录、年审缴费等管理 | 通过 |
| 8 | 保险费用 | 保险时间、年限等管理 | 通过 |
| 9 | 突发事件 | 突发事件上报 | 事件编号、名称、位置、时间、类型、描述； 路况巡查人员发现并上报，上报后的突发事件在地图上展示，上报人员和科长可见 | 通过 |
| 10 | 突发事件处理 | 录入处理结果，关闭地图显示 | 通过 |
| 11 | 突发事件统计查询 | 对突发事件查询，多样化展示（列表，图形） | 通过 |
| 12 | 应急救援 | 救援事件上报 | 由车辆管理员上报救援事件至相关领导 | 通过 |
| 13 | 救援事件处理 | 录入处理结果，关闭地图显示 | 通过 |
| **3** | **地图管理** | | |  |
| 1 | 地图数据处理 | OBD数据 | 监控设施、数据处理、硬件接口等。将数据放入数据库备查。 | 通过 |
| 2 | OBU数据 | 高速公路收费站，收费信息，收费车辆人员等管理。根据OBU数据查找套牌车 | 通过 |
| 3 | 地图功能 | 基础功能 | 放大、缩小、鹰眼、全图等 | 通过 |
| 4 | 图例：地图上各类图标表示内容 | 通过 |
| 5 | 测距：任意多个点的距离 | 通过 |
| 6 | 区域管理 | 在地图上画出车辆的运行范围。根据所制定的轨迹或范围对车辆实施监控并提醒 | 通过 |
| 7 | 监控管理 | 显示监控状态，车辆实时参数 调阅情报板显示内容； | 通过 |
| 9 | 地图发布 | 发布地图信息 | | 通过 |
| **4** | **报表管理** | | |  |
| 1 | 管理报表 | 车辆统计 | 车辆数据统计，数量，运行时间，油耗等 | 通过 |
| 2 | 人员统计 | 根据人员进行数据统计，个人的驾驶行为，费用等 | 通过 |
| 3 | 行程统计 | 对行程进行统计，行程各项参数。根据行程进行整体税费统计 | 通过 |
| 4 | 费用统计 | 整体费用统计，各项费用的分析，图表展示 | 通过 |
| 5 | 异常用车统计 | 异常用车的各项参数统计 | 通过 |
| 6 | 绩效考核统计 | 针对驾驶员的驾驶行为进行绩效考核统计 | 通过 |
| 7 | 运营统计 | 运营过程中的其他各项参数统计 | 通过 |
| 8 | 维修管理 | 修理信息 | 管理车辆修理信息单 | 通过 |
| 9 | 信息查询 | 对车辆修理记录的查询，管理 | 通过 |
| 10 | 配件管理 | 车辆修理时所用的配件，价格等进行记录管理 | 通过 |
| 11 | 报表展示 | web展示 | 在web页面展示格式正确的报表 | 通过 |
| 12 |  | 报表导出 | 根据需求导出pdf或其他格式的文件 | 通过 |
| **5** | **第三方接口** | | |  |
| 1 | OBU数据接口 | OBU数据调用 | 调用OBU数据接口，对数据进行处理 | 通过 |
| 2 | OBU数据管理 | 录入OBU数据，根据OBU数据进行消息提醒。需要时查询所有OBU相关数据 | 通过 |
| 3 | OBD数据 | OBD数据调用 | 接收养护科报修的小修工程任务单 | 通过 |
| 4 | OBD管理 | OBD硬件设备的管理 | 通过 |
| 5 | 路况信息 | 路况信息接口 | 路况信息输入接口 | 通过 |
| **6** | **监控分中心** | | |  |
| 1 | 信息推送管理 | 内容展示 | 有历史记录并能使用 | 通过 |
| 2 | 内容生成 | 可按类型储存内容并能修改内容 | 通过 |
| 3 | 内容发布 | 可预存、定时发布(含通信协议) | 通过 |
| 4 | 发布记录 | 向移动端推送信息 | 通过 |
| 5 | 推送日志记录 | 记录每日推送的日志 | 通过 |
| 6 | 推送管理 | 管理不同用户推送类型 | 通过 |
| **7** | **系统管理** | | |  |
| 1 | 系统管理 | 用户登录 | | 通过 |
| 2 | 修改密码 | | 通过 |
| 3 | 用户权限 | 用户管理、角色管理、权限分配、操作权限 | 通过 |
| 4 | 参数管理 | 系统运行参数、基础代码维护管理 | 通过 |
| 5 | 系统日志 | 操作日志、登录日志 | 通过 |
| **8** | **Web端开发** | | |  |
| 1 | Web页面开发 | 用车申请 | 用户申请用车的流程，填入各项信息并能通过地图选择出行地点 | 通过 |
| 2 | 申请管理 | 待办任务、已办任务、全部任务的列表，查看，管理等接口 | 通过 |
| 3 | 系统设置 | 系统设置，个人信息，提醒等设置修改 | 通过 |
| 4 | 上/下车 | 司机对乘客上下车的管理 | 通过 |
| 5 | 我的任务 | 司机的任务列表，待完成和已完成列表的查看 | 通过 |
| 6 | 信息提交 | 提交行驶过程中的税费、罚款、事故等各项信息 | 通过 |
| 7 | 消息提醒 | 接收来自平台的各方面消息 | 通过 |
| 8 | 事件提醒 | 接收来自平台的各方面消息 | 通过 |
| 9 | 费用管理 | 查看和修改所有车辆的费用情况 | 通过 |
| 10 | 司机信息 | 查看和管理所有司机的情况 | 通过 |
| 11 | 人车匹配 | 司机和人的信息匹配 | 通过 |
| 12 | 车辆基本信息 | 管理所有车辆的基本信息 | 通过 |
| 13 | 车辆处置 | 车辆到期处置的记录，管理 | 通过 |
| 14 | 维修保养 | 保养维修车辆记录 | 通过 |
| 15 | 车辆管理 | 车辆的各项信息管理 | 通过 |
| 16 | 报表查看 | 查看报表管理中的各项报表 | 通过 |
| **9** | **Android移动端开发** | | |  |
| 1 | 使用者APP | 用车申请 | 用户申请用车的流程，填入各项信息并能通过地图选择出行地点 | 通过 |
| 2 | 我的申请 | 待办任务、已办任务、全部任务的列表，查看，管理等接口 | 通过 |
| 3 | 系统设置 | 系统设置，个人信息，提醒等设置修改 | 通过 |
| 4 | 领导审批 | 领导对下属使用者有用车审批的功能，可以批复或查看用户申请 | 通过 |
| 5 | 司机APP | 上/下车 | 司机对乘客上下车的管理 | 通过 |
| 6 | 我的任务 | 司机的任务列表，待完成和已完成列表的查看 | 通过 |
| 7 | 我的车 | 所驾驶车辆的车况检查查看 | 通过 |
| 8 | 查看维修保养记录 | 通过 |
| 9 | 信息提交 | 提交行驶过程中的税费、罚款、事故等各项信息 | 通过 |
| 10 | 消息提醒 | 接收来自平台的各方面消息 | 通过 |
| 11 | 系统设置 | 系统设置 | 通过 |
| 12 | 车管员APP | 事件提醒 | 接收来自平台的各方面消息 | 通过 |
| 13 | 费用管理 | 查看和修改所有车辆的费用情况 | 通过 |
| 14 | 司机信息 | 查看和管理所有司机的情况 | 通过 |
| 15 | 人车匹配 | 司机和人的信息匹配 | 通过 |
| 16 | 车辆基本信息 | 管理所有车辆的基本信息 | 通过 |
| 17 | 车辆处置 | 处置车辆 | 通过 |
| 18 | 维修保养 | 保养维修车辆 | 通过 |
| 19 | 车辆管理 | 车辆的各项信息管理 | 通过 |
| 20 | 适配 | Android版本适配 | 适配主要的Android平台 | 通过 |
| **10** | **IOS移动端开发** | | |  |
| 1 | 使用者APP | 用车申请 | 用户申请用车的流程，填入各项信息并能通过地图选择出行地点 | 通过 |
| 2 | 我的申请 | 待办任务、已办任务、全部任务的列表，查看，管理等接口 | 通过 |
| 3 | 系统设置 | 系统设置，个人信息，提醒等设置修改 | 通过 |
| 4 | 司机APP | 上/下车 | 司机对乘客上下车的管理 | 通过 |
| 5 | 我的任务 | 司机的任务列表，待完成和已完成列表的查看 | 通过 |
| 6 | 我的车 | 所驾驶车辆的车况检查查看 | 通过 |
| 7 | 查看维修保养记录 | 通过 |
| 8 | 信息提交 | 提交行驶过程中的税费、罚款、事故等各项信息 | 通过 |
| 9 | 消息提醒 | 接收来自平台的各方面消息 | 通过 |
| 10 | 系统设置 | 系统设置 | 通过 |
| 11 | 车管员APP | 事件提醒 | 接收来自平台的各方面消息 | 通过 |
| 12 | 费用管理 | 查看和修改所有车辆的费用情况 | 通过 |
| 13 | 司机信息 | 查看和管理所有司机的情况 | 通过 |
| 14 | 人车匹配 | 司机和人的信息匹配 | 通过 |
| 15 | 车辆基本信息 | 管理所有车辆的基本信息 | 通过 |
| 16 | 车辆处置 | 处置车辆 | 通过 |
| 17 | 维修保养 | 保养维修车辆 | 通过 |
| 18 | 车辆管理 | 车辆的各项信息管理 | 通过 |

## 系统性能测试报告

|  |  |  |
| --- | --- | --- |
| 制表日期：2015/08/01 | |  |
| 测试内容： | 测试服务器 CPU、内存和磁盘 的峰值使用情况。主要测试在obd设备达到1000时系统的负载情况。 | |
| 测试步骤： | (1)启动Redis服务  (2)启动Kafka服务  (3)启动OBD Server服务  (4)启动tomcat服务  (5)安装并启动主业务服务  (6)利用obd server测试程序模拟10000辆车发送消息的情况  (7)同时启动10000个客户端模拟用户实时请求  (8)记录24小时内系统资源使用情况 | |
| 测试结果： | 内存使用率  http://220.178.67.250/cacti/graph_image.php?action=view&local_graph_id=1&rra_id=1  CPU使用率  http://220.178.67.250/cacti/graph_image.php?action=view&local_graph_id=16&rra_id=1  磁盘使用率  http://220.178.67.250/cacti/graph_image.php?action=view&local_graph_id=8&rra_id=1 | |
| 测试结果 | 系统运行平稳，对kafka收到的消息能及时处理，未出现明显的滞后情况。对系统资源的使用未随时间产生较大变化。 | |

## 不间断运行测试报告

|  |  |  |
| --- | --- | --- |
| 制表日期：2015/08/01 | |  |
| 测试内容： | 对比服务器程序连续运行24小时前后，CPU、内存和磁盘的使用情况。 | |
| 测试步骤： | (1)启动Redis服务  (2)启动Kafka服务  (3)启动OBD Server服务和2000个客户端连接  (4)记录当前CPU和内存的使用情况  (5)连续运行24小时后，记录CPU和内存的使用情况  (6)检查日志内容是否正常 | |
| 测试结果： | |  |  |  |  | | --- | --- | --- | --- | |  | CPU（%） | 内存（M） | 磁盘（%） | | 启动时 | 4.3 | 232 | 10.2 | | 运行24小时后 | 4.1 | 285 | 10.3 | | |
| 运行24小时后检查日志没有错误信息输出。日志输出正常。 | |

## 可靠性测试报告

|  |  |  |
| --- | --- | --- |
| 测试用例 | 测试过程描述 | 测试结果 |
| 成熟性 | 公车管理服务平台使用系统资源达到规定的极限时，系统不会崩溃、不会异常退出也没丢失数据。 | 系统在达到极限时拒绝新的客户端接入 |
| 公车管理服务平台命令行参数错误，系统不会崩溃也不丢失数据。 | 系统遇到错误启动参数时，会输出警告信息并退出。 |
| 系统部分模块错误后，不会引起整个系统的服务失败。 | 系统部分模块错误会造成部分服务不可用，不会对整体系统运行造成影响。 |
| 容错性 | 遇到网络异常，系统不会崩溃。 | 记录网络崩溃的时间，并断开客户端连接。 |
| 遇到系统错误后，当问题修复科自主恢复 | 当出现网络问题、硬件问题等系统问题时，当系统修复后程序可自主恢复服务。 |
| 客户端连接异常，系统不会崩溃。 | 记录该客户端出现异常的时间和具体原因，并关闭该连接。 |
| 系统不会出现内存溢出漏洞 | 当系统运行足够长的时间，不会出现内存占用过高的情况，不会有内存溢出。 |
| 易恢复性 | 系统运行失效后，应能较快重建系统 | 重新启动服务程序。 |
| 数据校验机制 | 遇到错误时，系统会提示用户并做处理 | 用户使用时若出现错误，系统都能正确返回相关提示，引导用户修复错误。 |
| 遇到非法请求，系统不会崩溃。 | 当有非法用户尝试破解系统时，系统不会崩溃。不会影响用户使用。 |

## 集群配置测试报告

|  |  |  |
| --- | --- | --- |
| 测试用例 | 测试过程描述 | 测试结果 |
| 网页端集群测试 | 测试网页端是否可以利用SLB作集群访问，是否能保持用户session。   1. 开启其中一台ECS，关闭另一台。 2. 打开网页服务vmp.witgo.cn 3. 登录系统 4. 关闭当前ECS，开启另一台ECS 5. 刷新网页 | 网页刷新后系统仍然保持登录状态，两台ECS的session达到共享状态。 |
| Redis集群测试 | 1. 在支撑层的ECS上分别安装redis服务。 2. 配置redis为集群 3. 在主节点查看redis集群状况 4. 在主节点存入键值对 5. 在从节点读取键值，对比是否与存入的一致 | 可以正常看到主节点下的多个从节点已正常连接。  键值对对比正常，表明Redis集群完成测试 |
| Kafka集群测试 | 1. 在某个ECS上安装并启动zookeeper 2. 启动多个kafka实例，并配置连接到同一个zookeeper 3. 查看zookeeper是否有多个kafka实例连接 4. 在每个ECS实例上启动kafkaServer 5. 查看kafkaServer是否正常处理数据并存入数据库 | Kafka集群搭建成功，队列消息被分散到多个ECS实例上。 |
| 系统功能测试 | 1. 启动服务器相关服务 2. 逐项测试系统功能是否可用，逻辑是否错误 | 测试结果与3.2节相同，相关服务使用正常。 |

## 安全性测试报告

|  |  |  |
| --- | --- | --- |
| 测试用例 | 测试过程描述 | 测试结果 |
| 安全性测试 | (1) 使用未登录用户访问需要登录的接口，检查是否会允许访问  (2) 使用用户角色直接访问车管员URL，查看是否允许操作  (3) 使用构造的URL访问接口，查看是否有数据库注入漏洞 | 使用未登录用户无法访问需要登录的接口，权限认证结果正确。  使用用户角色可访问车管员页面，但无法看到相应信息，无法进行需要权限的操作。  未发现sql注入漏洞。 |

## 可维护性测试报告

|  |  |  |
| --- | --- | --- |
| 测试用例 | 测试过程描述 | 测试结果 |
| 安装配置 | (1) 产品安装  (2) 初始化参数设置  (3) 使用 | 系统提供安装维护手册方便使用  《公车管理服务平台安装部署指南》 |
| 日志维护 | 业务操作记录都能自动记录到日志  日志内容要包括：时间、模块、日志类型和日志具体信息。 | 系统各个模块记录日志均包括时间、模块、日志类型和日志具体信息。 |
| 日志内容中包含的所有信息都记录正确。例如：在执行操作时，没有操作成功，而日志中的操作结果却显示成功。 | 系统日志能正确显示操作结果。 |

# 测试结论

## 功能性

系统实现了项目需求书中所规定的相关功能，实现了1）用户管理，组织架构管理，车辆管理 2）运营管理，包括终端设备管理，系统管理 3）调度管理，建立用车申请流程，对车辆运行过程中的相关参数进行监控，在地图上显示轨迹回放 4）监管功能，包括统计报表，有效区域设定，路线设定 5）事件提醒，实时监控车辆运行状态，返回相应警告或信息 6）费用管理，费用异常监控，通行费、保险费、年审、罚款记录 7）人车管理，司机信息，车辆基本信息，车辆处置。8）扩展功能，系统设计可伸缩，可扩展，允许服务器集群搭建。的功能。

## 可靠性

系统的不间断运行测试表明该系统控制严密，基本能应对正常使用中出现的各种情况，在系统出现不可抗力失效的情况下可快速恢复，丢失数据的可能性较低。整个系统的可靠性较高。

系统运行24小时后资源消耗未出现较大波动，登录用户数量对资源的消耗影响较小，系统对资源的利用效率高，未出现内存溢出的情况，表明系统在长期运行时仍然有较好的稳定性，在集群的情况下与单机测试结果相同，系统的可靠性较好。

## 兼容性

系统在单机测试和集群测试时采用不同的操作系统和架构，公车管理服务平台在Debian系统和CentOS系统下均工作正常，未出现兼容性问题。

其他操作系统未做进一步测试。

## 安全性

系统未出现常见的漏洞，权限设计合理且有效。暂未发现明显的安全性问题。