**软件测试计划**

——基于微型小程序的快递代拿小程序



项目名称： 东风代拿小程序

专业班级： 软件工程1903

小组成员：林敏杰、陈沿良、林舒逸

指导教师： 杨枨老师

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [ ] 正式发布  [√] 正在修改 | 文件标识： | SE2020-G014-软件测试计划(STP) |
| 当前版本： | 0.1 |
| 作者： | 林敏杰、陈沿良、林舒逸 |
| 完成日期： | 2021-11-14 |

软件测试计划(STP)

软件测试计划(STP)

说明：

1.《软件测试计划》（STP)描述对计算机软件配置项CSCI,系统或子系统进行合格性测试的计划安排。内容包括进行测试的环境、测试工作的标识及测试工作的时间安排等。

2.通常每个项目只有一个STP,使得需方能够对合格性测试计划的充分性作出评估。

**注**：本文档参考GB+T-8567-2006计算机软件文档编制规范

**版本记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 修订日期 | 版本/状态 | 编写人 | 审核人 | 发布日期 | 备注 |
| 01 | 2021-11-14 | 0.1 | 陈沿良 | 林敏杰 | 2020-11-23 | 报告的初始版本制作 |
| 02 |  |  |  |  |  |  |
| 03 |  |  |  |  |  |  |
| 04 |  |  |  |  |  |  |
| 05 |  |  |  |  |  |  |
| 06 |  |  |  |  |  |  |

**目录**

[1引言 5](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351883)

[1.1标识 5](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351884)

[1.2系统概述 5](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351885)

[1.3文档概述 5](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351886)

[1.4与其他计划的关系 6](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351887)

[1.5基线 6](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351888)

[2引用文件 6](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351889)

[3软件测试环境 7](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351890)

[3.1测试现场 7](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351891)

[3.2软件项 7](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351892)

[3.3硬件及固件项 7](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351893)

[3.4其他材料 8](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351894)

[3.5所有权种类、需方权利与许可证 8](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351895)

[3.6安装、测试与控制 8](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351896)

[3.7参与组织 8](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351897)

[3.8人员 9](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351898)

[3.9定向计划 9](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351899)

[3.10要执行的测试 9](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351900)

[4计划 9](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351901)

[4.1总体设计 9](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351902)

[4.1.1测试目标 9](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351903)

[4.1.2测试级 10](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351904)

[4.1.3测试类别 10](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351905)

[4.1.4一般测试条件 11](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351906)

[4.1.5测试过程 11](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351907)

[4.1.5.1单元测试 12](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351908)

[4.1.5.2集成测试 12](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351909)

[4.1.5.3白盒测试技术 13](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351910)

[4.1.5.4黑盒测试技术 14](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351911)

[4.1.5.5 调试 14](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351912)

[4.1.6数据记录、归约和分析 14](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351913)

[4.2计划执行的测试 15](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351914)

[4.2.1登录、注册与注销 15](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351915)

[4.2.2 个人日记和个人动态 16](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351916)

[4.2.3 动态共享、消息中心统计 17](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351917)

[4.3测试用例 18](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351918)

[5测试进度表 18](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351919)

[6需求的可追踪性 18](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351920)

[7评价 19](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351921)

[7.1评价准则 19](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351922)

[7.2数据处理 19](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351923)

[7.3结论 19](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351924)

[8注解 20](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351925)

[附录 21](file:///C:\\Users\\hp\\Desktop\\SE2020-G13-软件测试计划(STP)V0.1.5.doc" \l "_Toc59351926)

**1引言**

**1.1标识**

标题：《软件测试计划》

简称：STP

版本号：0.1.5

说明：本文档支持Office、WPS等word文档阅读软件

**1.2系统概述**

1. **软件概述**

项目由G014小组在软件工程课程下提出，根据杨枨老师提出的建议，我们确定要求：包含用户实名注册、用户发接单、双向评分、仲裁等功能，争取做到精简界面，为用户带来良好的应用体验。

**1.3文档概述**

本文档的内容包括需求、合格性规定、需求可追踪性等，其中对需求模块进行详细分析。

本文档的使用除了项目开发团队的内部使用外，还需提交用户和客户组织负责人审查批准。

本文档的使用应遵守国家先关法律法规，未经允许不得对外公开，需要有一定的保密性和私密性。

**1.4与其他计划的关系**

本文档属于项目管理计划的一部分，是对软件测试的一个计划与安排。

**1.5基线**

基线是软件生存期各个开发阶段的工作成果。

重要的检查点是里程碑，重要的需要客户确认的里程碑是基线。

主要基线如下表所示：

SE2021-G014-正式选题1.0.docx

SE2021-G014-项目计划书1.0.docx

SE2021-G014-可行性分析(研究)报告(FAR)0.1.docx

SE2021-G014-软件需求说明(SRS)0.1.docx

**2引用文件**

[1]《GB T-8567-2006计算机软件文档编制规范》

[2]《SE2020-G014-项目计划1.0》

[3]《SE2020-G014-可行性分析1.0》

**3软件测试环境**

**3.1测试现场**

本软件的测试现场为小组各开发成员的寝室、求真楼自习室、理四一楼大厅或者北校图书馆。以上涉及的测试现场，外界干扰均较少，能满足正常的软件测试。

**3.2软件项**

**操作系统**：Windows 10 操作系统

**编译程序**：

**通信软件**：微信；钉钉；QQ

**相关应用软件**：Android studio；Flutter

**数据库**：MySQL 8.0；Navicat 11.1；

**输入文件**：已准备好的各种测试文档或文件

**代码检查程序**：

**测试数据产生器**：DataFactory；JMeter；Generatedata；DBMonster

**测试控制软件**：；SilkTest；AutoRunner；JMeter；LoadRunner；postman；buglliza

**其他专用测试软件**：TestLink；TestLodge；TestRail；Gemini；PractiTest；TestCenter

所有与软件项有关的测试，软件项目开发方均遵循严格的保密措施，对于其他可能的保密性与私密性问题，也将严格遵守相关法律法规，妥善处理。

**3.3硬件及固件项**

**计算机硬件**：CPU为(英特尔)Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz(2201 MHz)；内存为8.00 GB (2667 MHz)；固态硬盘130G；移动硬盘1T；显卡为NVIDIA GeForce GTX 1050 Ti；声卡为Realtek High Definition Audio；网卡为Realtek 8821CE Wireless LAN 802.11ac PCI-E NIC。

**接口设备**：三台个人Windows10操作系统的电脑；两台安卓系统手机，一台iOS系统手机。

**通信设备**：三台个人Windows10操作系统的电脑；两台安卓系统手机，一台iOS系统手机；路由器；中国移动固定基站等。

**测试数据归约设备**：三台个人Windows10操作系统的电脑

**仪器设备**：三台个人Windows10操作系统的电脑；两台安卓系统手机，一台iOS系统手机。

**测试消息生成器**：DataFactory；JMeter；Generatedata；DBMonster

**测试计时设备**：两台安卓系统手机，一台iOS系统手机。

**测试事件记录仪**：两台安卓系统手机，一台iOS系统手机。

所有与硬件及固件项有关的测试，软件项目开发方均遵循严格的保密措施，对于其他可能的保密性与私密性问题，也将严格遵守相关法律法规，妥善处理。

**3.4其他材料**

数据字典的设计清单、输入的测试样例清单、输出的样本结果清单、项目的提出和介绍、软件项目计划、软件可行性分析报告、软件需求分析报告、软件总体设计报告、软件详细设计报告、用户使用手册、软件结构说明。

所有可能涉及的其他材料，软件项目开发方均遵循严格的保密措施，对于其他可能的保密性与私密性问题，也将严格遵守相关法律法规，妥善处理。

**3.5所有权种类、需方权利与许可证**

由于本软件是软件工程课程实践项目，是由G014小组成员共同开发，所有权属于G014小组并获得需方权利，且使用的软件、代码或程序等是基于免费、开源或开放等，且本软件在开发时直至目前暂时不涉及许可证。

**3.6安装、测试与控制**

a.获取和开发软件测试环境中的每个元素；

b.使用前，安装与测试软件测试环境中的每项；

c.控制与维护软件测试环境中的每项。

**3.7参与组织**

本软件测试的主要组织人员和开发成员均为G014小组成员，必要时可以有普通测试用户的加入。

**3.8人员**

在测试现场，主要的测试人员为3人即G014小组成员，必要时可以有一位普通用户的加入。主要测试人员的类型为软件开发成员，由于是第一次开发软件，成员的测试技术相对而言较为薄弱。测试日期预定于2021年12月1日到2021年12月14日，预计在总的一个测试规定时间完成测试。在成员之间交接时，当前测试人员需要把自己已经做好的工作复述给下一位测试人员，并告知下一位测试人员接下来需要做什么，尤其是一些注意事项，以便测试计划的正常进行。

**3.9定向计划**

测试前或测试期间的用到的技术或者技巧需要测试人员相互之间交流与学习，无统一集中的定向培训。对于可能参加的普通用户，开发人员应当对其进行简单解释，并给与软件的用户使用手册。

**3.10要执行的测试**

TBD

**4计划**

**4.1总体设计**

**4.1.1测试目标**

1、测试是为了发现程序中的错误而执行程序的过程。

2、好的测试方案是极可能发现迄今为止尚未发现的错误的测试方案。

3、成功的测试是发现了至今为止尚未发现的错误的测试。

**4.1.2测试级**

主要有以下四大测试：

1. 单元测试：在软件测试中，单元就是组成软件最小的底层代码结构，一般就是类、函数、组件。目的是检验软件基本组成单位的正确性。测试内容：模块接口测试、局部数据结构测试、路径测试、错误处理测试、边界测试。
2. 集成测试（Integration Test）：将不同的单元模块组合到一起，然后验证它们之间沟通的“桥梁”是否能正常工作。目的是检查软件单位之间的接口是否正确。测试内容：模块之间数据传输、模块之间功能冲突、模块组装功能正确性、全局数据结构、单模块缺陷对系统的影响。
3. 系统测试：将软件系统看成是一个系统的测试。包括对功能、性能以及软件所运行的软硬件环境进行测试。系统测试的目的是对最终软件系统进行全面的测试，确保最终软件系统满足产品需求并且遵循系统设计。测试内容：功能、界面、可靠性、易用性、性能、兼容性、安全性等。
4. 验收测试:是部署软件之前的最后一个测试操作。它是技术测试的最后一个阶段，也称为交付测试。目的是确保软件准备就绪，按照项目、任务、约定的验收依据文档，向需方展示该软件系统满足原始需求。测试内容：功能、界面、可靠性、易用性、性能、兼容性、安全性等。

此外，还有Alpha测试和Beta测试等级别。其中，Alpha测试由用户在开发者的场所进行,并且在开发者对用户的“指导”下进行测试。开发者负责记录发现的错误和使用中遇到的问题,是在受控的环境中进行的。Beta测试由软件的最终用户们在一个或多个客户场所进行。与Alpha测试不同,开发者通常不在Beta测试的现场， Beta测试是软件在开发者不能控制的环境中的“真实”应用。用户记录在Beta测试过程中遇到的一切问题(真实的或想象的),并且定期把这些问题报告给开发者。接收到在Beta测试期间报告的问题之后,开发者对软件产品进行必要的修改,并准备向全体客户发布最终的软件产品。

**4.1.3测试类别**

1、按开发阶段：单元测试、集成测试、系统测试、验收测试

2、按测试实施组织：α测试、β测试、第三方测试

3、按测试执行方式：静态测试、动态测试

4、按是否查看代码：黑盒测试、白盒测试、灰盒测试

5、按是否手工执行划分：手工测试、自动化测试

6、按测试对象划分：性能测试、安全测试、兼容性测试、文档测试、易用性测试（用户体验测试）、业务测试、界面测试、安装测试

**4.1.4一般测试条件**

在测试中，通常需要给定一定的测试数据样例，例如在注册的测试中，需要输入手机号，众所周知，手机号尾数一般为13位数字，因此可以设计成三组数据，分别是小于13位的，等于13位的和大于13位的，这里面既包含了限定的值，也包含了非限定的值。同样，例如在个人动态的发布上，设置了图片和字数的限制，图片最大值为3张，最小值为0；文本的字数限定最大值为1000个字，最小值为0，这是因为对相关存储库设定了大小的存储限制。此外，在消息的测试上，可以采用并发的模式进行确认是否可以正常接收和发送。

在上述基本的测试基础之上，也应当更合理的设置测试数据，遵循一些测试/回归测试所遵循的办法等。

**4.1.5测试过程**

在软件测试的基础上，一般情况下主要的测试步骤有如下几点：

1.模块测试：在设计好的软件系统中,每个模块完成一个清晰定义的子功能，而且这个子功能和同级其他模块的功能之间没有相互依赖关系。把每个模块作为一个单独的实体来测试，而且通常比较容易设计检验模块正确性的测试方案。模块测试的目的是保证每个模块作为一个单元能正确运行，在这个测试步骤中所发现的往往是编码和详细设计的错误。

2.子系统测试：子系统测试是把经过单元测试的模块放在一起形成一个子系统来测试。模块相互间的协调和通信是这个测试过程中的主要问题，着重测试模块的接口。

3.系统测试：系统测试是把经过测试的子系统装配成一个完整的系统来测试。在这个过程中不仅应该发现设计和编码的错误，还应该验证系统确实能提供需求说明书中指定的功能，而且系统的动态特性也符合预定要求。

4.验收测试：验收测试把软件系统作为单个实体进行测试，测试内容与系统测试基本类似,但是它是在用户积极参与下进行的，而且可能主要使用实际数据(系统将来要处理的信息)进行测试。验收测试的目的是验证系统确实能够满足用户的需要。

5.平行运行：同时运行新开发出来的系统和将被它取代的旧系统，以便比较新旧两个系统的处理结果。这样做的具体目的有如下几点。(1)可以在准生产环境中运行新系统而又不冒风险。(2)用户能有一段熟悉新系统的时间。(3)可以验证用户指南和使用手册之类的文档。(4)能够以准生产模式对新系统进行全负荷测试，可以用测试结果验证性能指标。

**4.1.5.1单元测试**

测试重点有如下几点：

1.模块接口：首先应该对通过模块接口的数据流进行测试，如果数据不能正确地进出，所有其他测试都是不切实际的。在对模块接口进行测试时主要检查下述几个方面：参数的数目、次序、属性或单位系统与变元是否一致；是否修改了只作输人用的变元；全局变量的定义和用法在各个模块中是否一致。

2.局部数据结构：对于模块来说，局部数据结构是常见的错误来源。应该仔细设计测试方案，以便发现局部数据说明、初始化、默认值等方面的错误。

3.重要的执行通路：应该设计测试方案用来发现由于错误的计算、不正确的比较或不适当的控制流而造成的错误。

4.出错处理通路：需要着重测试一些可能发生的错误：①对错误的描述是难以理解的；②记下的错误与实际遇到的错误不同；③在对错误进行处理之前，错误条件已经引起系统干预；④对错误的处理不正确；⑤描述错误的信息不足以帮助确定造成错误的位置。

5.边界条件：边界测试是单元测试中最后的也可能是最重要的测试，需要确定各种边界值的测试方案。

除此之外还有代码审查和计算机测试等。

**4.1.5.2集成测试**

集成测试是测试和组装软件的系统化技术，例如，子系统测试即是在把模块按照设计要求组装起来的同时进行测试，主要目标是发现与接口有关的问题(系统测试与此类似)。例如，数据穿过接口时可能丢失；一个模块对另一个模块可能由于疏忽而造成有害影响；把子功能组合起来可能不产生预期的主功能；个别看来是可以接受的误差可能积累到不能接受的程度；全程数据结构可能有问题等。

由模块组装成程序时有两种方法。一种方法是先分别测试每个模块，再把所有模块按设计要求放在起结合成所要的程序，这种方法称为非渐增式测试方法;另一种方法是把下一个要测试的模块同已经测试好的那些模块结合起来进行测试，测试完以后再把下一个应该测试的模块结合进来测试。这种每次增加一个模块的方法称为渐增式测试，这种方法实际上同时完成单元测试和集成测试。

非渐增式测试一下子把所有模块放在一起,并把庞大的程序作为一个整体来测试,测试者面对的情况十分复杂。测试时会遇到许许多多的错误，改正错误更是极端困难，因为在庞大的程序中想要诊断定位一个错误是非常困难的。而且一旦改正一个错误之后，马上又会遇到新的错误。渐增式测试与“一步到位”的非渐增式测试相反，它把程序划分成小段来构造和测试，在这个过程中比较容易定位和改正错误；对接口可以进行更彻底的测试；可以使用系统化的测试方法。因此，目前在进行集成测试时普遍采用渐增式测试方法。

当使用渐增方式把模块结合到程序中去时，有自顶向下和自底向上两种集成策略。

**4.1.5.3白盒测试技术**

白盒测试技术中的典型技术有逻辑覆盖和控制结构测试。

关于逻辑结构有如下一些覆盖标准：

1. 语句覆盖：为了暴露程序中的错误，选择足够多的数据，至少每个测试语句应该执行一次。
2. 判定覆盖：不仅每个测试语句必须至少执行一次，而且每个判定的每种可能的结果都应该至少执行一次。
3. 条件覆盖：不仅每个语句至少执行一次，而且使判定表达式中的每个条件都取到各种可能的结果。
4. 判定/条件覆盖：选取足够多的测试数据，使得判定表达式中的每个条件都取到各种可能的值，而且每个判定表达式也都能取到各种可能的结果。
5. 条件组合覆盖：要求选取足够多的测试数据，使得每个判定表达式中的条件各种可能组合都至少出现一次。
6. 点覆盖：满足点覆盖的标准要求选取足够多的测试数据，使得程序执行的路径至少经过流图的每个结点一次。
7. 边覆盖：满足边覆盖的测试标准，要求选取足够多的测试数据，使得程序执行路径至少经过流图中的每一条边一次。
8. 路径覆盖：选取足够多的测试数据，使得程序的每条可能测试路径都至少执行一次。

关于白盒测试技术的控制结构测试：一般有基本路径测试、条件测试和循环测试。

**4.1.5.4黑盒测试技术**

黑盒测试着重测试软件功能。一般包括等价划分、边界值分析和错误推测。

黑盒测试技术力图发现以下类型错误：

1. 功能不正确或者遗漏功能；
2. 界面错误；
3. 数据结构错误或者外部数据库访问错误；
4. 性能错误；
5. 初始化和终止错误。

同时，黑盒测试技术也应当考虑下述问题：

1. 怎样测试功能的有效性?
2. 哪些类型的输人可构成好测试用例?
3. 系统是否对特定的输人值特别敏感?
4. 怎样划定数据类的边界?
5. 系统能够承受什么样的数据率和数据量?
6. 数据的特定组合将对系统运行产生什么影响?

应用黑盒测试技术，能够设计出满足下述标准的测试用例集：

1.所设计出的测试用例能够减少为达到合理测试所需要设计的测试用例的总数。

2．所设计出的测试用例能够告诉人们，是否存在某些类型的错误，而不是仅仅指出与特定测试相关的错误是否存在。

**4.1.5.5 调试**

一般情况下，软件的测试调试包括蛮干法、回溯法、原因排除法。

**4.1.6数据记录、归约和分析**

在经过测试过程中，通常都会对流程中涉及的数据进行一个记录，在测试完毕后都会对反馈得到的数据进行一个简单的分析。对于测试使用的记录应当一一记录和备份，对测试得到的结果也应当一一记录。在数据归约上，通过属性选择和数据采样两种方式，分别针对原始数据集中的属性和记录进行归约分析。

在数据归约上，有如下几类：①特征归约，从原有的特征中删除不重要或不相关的特征，或者通过对特征进行重组来减少特征的个数，其原则是在保留、甚至提高原有判别能力的同时减少特征向量的维度。一般包括搜索过程、评估过程、分类过程三个步骤。通过更少的数据提高挖掘效率。②样本归约，从数据集中选出一个有代表性的样本的子集。子集大小的确定要考虑计算成本、存储要求、估计量的精度以及其它一些与算法和数据特性有关的因素。③特征值归约，它是特征值离散化技术，将连续型特征的值离散化，使之成为少量的区间，每个区间映射到一个离散符号。这种技术的好处在于简化了数据描述，并易于理解数据和最终的挖掘结果。常见的可以采用线性回归和多元回归等。

在结果分析上，应当尊重实际得到的实验结果，实事求是，不可造假，分析数据可以采用多个方法进行横向或者纵向的一个对比。这样得出的结果才是最准确可靠的。同时在整一个工程中，基本都是人工参与操作。

**4.2计划执行的测试**

**TBD**

**4.3测试用例**

a.测试用例的名称和标识；

b.简要说明本测试用例涉及的测试项和特性；

c.输入说明，规定执行本测试用例所需的各个输入，规定所有合适的数据库、文件、终端信息、内存常驻区域和由系统传送的值，规定各输入间所需的所有关系(如时序关系等)；

d.输出说明，规定测试项的所有输出和特性(如：响应时间)，提供各个输出或特性的正确值；

e.环境要求，见本文档第3章。

（TBD）

**5测试进度表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **测试现场** | **测试时间** | **测试项目** | **测试人员** | **审核人员** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**6需求的可追踪性**

对软件测试要解决的问题进行详细的分析,弄清楚参与软件测试活动的相关人员对软件测试活动和交付物的要求，包括需要输入什么数据，要得到什么结果，最后应输出什么等。

对测试需求分析，要知道以下几点：可以根据软件开发需求说明书逐条列出软件开发需求，并判断其可测试性，形成可测试的描述并界定出测试范围。根据质量标准，逐条制定质量需求，即测试通过标准。分析测试执行时需要实施的测试类型，建立测试需求跟踪矩阵，并输入测试需求管理系统，对测试需求实施严格有效的管理。

**7评价**

**7.1评价准则**

1.测试应基于用户需求；

2.做好软件测试计划是做好软件测试工作的关键；

3.应尽早的开始软件测试并不断的进行软件测试；

4.测试前必须明确定义好产品的质量标准；

5.避免测试自己的软件，要有基本用户参与；

6.应充分注意测试中的集群现象；

7.必须检查每个实际输出结果；

8.穷举测试是不可能的；

9.测试设计决定了测试的有效性和效率；

10.注意保留测试设计和说明文档，并注意测试设计的可重用性。

**7.2数据处理**

应当实事求是，每一个得到的数据都应有其实际的意义，应当充分挖掘。根据得到的bug，应当理解其背后可能存在的问题。必要时，应充分发挥数学头脑的思维，运用数学大方法进行分析，例如最常见的统计、回归分析等。

**7.3结论**

在当前预定的软件测试环境与软件测试人员的条件下，以及伴随着个人的知识水平和成本（主要是资金和时间问题），软件测试计划的任务完成难度较大，可能存在一定的风险。所以，在实际情况下，需要结合一定的进度和情况来进行测试，测试功能可能存在不完善的一个情况，如果时间不允许，测试人员将只对必要的组成部分进行一个测试和完善。

**8注解**

**1、软件开发背景**

项目由G014小组在软件工程课程下提出，根据杨枨老师提出的建议，我们确定要求：包含用户实名注册、用户发接单、双向评分、仲裁等功能，争取做到精简界面，为用户带来良好的应用体验。

**2、软件测试相关补充**

在软件实现中包括编码和测试两个阶段。按照传统的软件工程方法学，编码是在对软件进行了总体设计和详细设计之后进行的，它只不过是把软件设计的结果翻译成用某种程序设计语言书写的程序，因此，程序的质量基本上取决于设计的质量。程序内部的良好文档资料，有规律的数据说明格式，简单清晰的语句构造和输人输出格式等，都对提高程序的可读性有很大作用，也在相当大的程度上改进了程序的可维护性。

目前软件测试仍然是保证软件可靠性的主要手段。测试阶段的根本任务是发现并改正软件中的错误，软件测试是软件开发过程中最艰巨最繁重的任务，大型软件的测试应该分阶段地进行，通常至少分为单元测试、集成测试和验收测试3个基本阶段。设计测试方案是测试阶段的关键技术问题，基本目标是选用最少量的高效测试数据，做到尽可能完善的测试，从而尽可能多地发现软件中的问题。

白盒测式和黑盒测试是软件测试的两类基本方法,这两类方法各有所长，相互补充。通常，在测试过程的早期阶段主要使用白盒方法，而在测试过程的后期阶段主要使用黑盒方法。为了设计出有效的测试方案。软件工程师应该深人理解并坚持运用关于软件测试的基本准则。设计白盒测试方案的技术主要有，逻辑覆盖和控制结构测试；设计黑盒测试方案的技术主要有，等价划分、边界值分析和错误推测。在测试过程中发现的软件错误必须及时改正，这就是调试的任务。为了改正错误，首先必须确定错误的准确位置，这是调试过程中最困难的工作，需要审慎周密的思考和推理。为了改正错误往往需要修正原来的设计，必须通盘考虑统筹兼顾，而不能“头疼医头、脚疼医脚”,应该尽量避免在调试过程中引进新错误。

测试和调试是软件测试阶段中的两个关系非常密切的过程，它们往往交替进行。程序中潜藏的错误的数目，直接决定了软件的可靠性。通过测试可以估算出程序中剩余的错误数。根据测试和调试过程中已经发现和改正的错误数，可以估算软件的平均无故障时间；反之，根据要求达到的软件平均无故障时间，可以估算出应该改正的错误数，从而能够判断测试阶段何时可以结束。

**3、相关术语解释**

**单元测试**：对软件组成进行的测试。其目的是检验软件基本组成单位的正确性。测试对象是软件设计的最小单元：模块，又称为模块测试。

**集成测试**：也成联合测试，联调、组装测试，将程序模块采用适当的集成策略组装起来。

**系统测试**：将软件系统看成一个系统测试。包括对功能、性能以及软件所运行的硬软件环境进行测试。时间大部分在系统测试执行阶段，包括了回归测试和冒烟测试。

**验收测试**：部署软件之前的最后一个测试操作，它是技术测试室的最后一个阶段，也叫做交付测试，验收测试的目的是保证软件的准备就绪，按照项目合同、任务书、双方约定的验收依据文档，向软件的购买者展示该软件的原始的需求。

**α测试**：主要是由一个用户在开发环境进行的测试，也可以是公司内部的用户在模拟实际操作环境下进行的测试。主要的目的是：评价软件产品的FLURPS（即功能、局域化、可使用性、可靠性、性能和支持）。

**β测试**：由软件的最终的用户们在一个或者多个客户场所进行的测试。

**第三方测试**：介于开发方和用户之间的组织测试。

**静态测试**：静态测试值的是不运行程序本身，仅通过分析和检查源程序的语法、结构、过程、接口来检查程序的正确性。对需求规格说明书、软件设计说明书、流程图分析、符号执行来进行找错。

**动态测试**：指的就是运行被测的程序。检查运行结果与预期结果的差异，并分析运行效率、正确性和健壮性的等性能，这种方法主要是由三部分进行组成的：测试用例、执行程序、分析程序运行输出的结果。

**手工测试**：是由人一个一个的输入测试用例，然后观察结果、和机器测试相对应，属于比较原始，大事需要一个一个步骤进行测试。

自动化测试：在预设条件下运行系统或应用程序，评估运行结果、预先条件应该包括正常的条件和异常条件。简单的说自动化测试是把人为驱动的测试行为转化为机器执行的一种过程。

黑盒测试（Black-box-Testing）：黑盒测试也称为功能测试，测试中把被测的软件当成一个黑盒子，不关心盒子的内部结构是什么，指关心软件的输入数据和输出数据。

白盒测试（White-box-Testing）：白盒测试又称结构测试，透明盒测试、逻辑驱动测试或基于代码的测试。白盒值的是打开的盒子，去研究里面的源代码和程序结果。

灰盒测试（White-box-Testing）：灰盒测试是介于白盒测试与黑盒测试之间的一种测试，主要用于集成测试阶段。不仅关注输入输出的正确性。同时也关注程序内部的情况。

**附录**

**参考资料：**

[1]《GB T-8567-2006计算机软件文档编制规范》