软件测试计划

--基于微信开发者工具开发的飞翔的小鸟游戏



课 程： 软件工程导论

名 称： Flappy Bird

人 员: G17小组

**7.3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。

**版本记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 修订日期 | 版本/状态 | 修订人 | 备注 |
| 1 | 2020/12/08 | 0.1 | 陈骁 | 初稿 |

目录

[1引言 4](#_Toc58351776)

[1.1标识 4](#_Toc58351777)

[1.2系统概述 4](#_Toc58351778)

[1.3文档概述 5](#_Toc58351779)

[1.4与其他计划的关系 5](#_Toc58351780)

[1.5基线 5](#_Toc58351781)

[2引用文件 5](#_Toc58351782)

[3软件测试环境 6](#_Toc58351783)

[3.1测试现场 6](#_Toc58351784)

[3.2软件项 6](#_Toc58351785)

[3.3硬件及固件项 6](#_Toc58351786)

[3.4其他材料 7](#_Toc58351787)

[3.5所有权种类、需方权利与许可证 7](#_Toc58351788)

[3.6安装、测试与控制 7](#_Toc58351789)

[3.7参与组织 7](#_Toc58351790)

[3.8人员 7](#_Toc58351791)

[3.9定向计划 8](#_Toc58351792)

[3.10要执行的测试 8](#_Toc58351793)

[4计划 8](#_Toc58351794)

[4.1总体设计 8](#_Toc58351795)

[4.2计划执行的测试 10](#_Toc58351796)

[4.3测试用例 11](#_Toc58351797)

[5测试进度表 11](#_Toc58351798)

[1)分配给测试主要部分的时间和现场测试的时间， 11](#_Toc58351799)

[3)测试所需的数据库/数据文件值、输入值和其他操作数据的集合； 11](#_Toc58351800)

[6需求的可追踪性 11](#_Toc58351801)

[7评价 11](#_Toc58351802)

[7.1评价准则 11](#_Toc58351803)

[7.2数据处理 12](#_Toc58351804)

[7.3结论 12](#_Toc58351805)

[8注解 12](#_Toc58351806)

[附录 14](#_Toc58351807)

**软件测试计划的正文的格式如下：**

## 1引言

本章应分成以下几条。

### 1.1标识

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [ ] 正式发布  [√] 正在修改 | 文件标识： | SE2020-G17-软件测试计划(STP) |
| 当前版本： | 0.1 |
| 作者： | 陈骁 |
| 编写日期： | 2020-12-08 |

### 1.2系统概述

用途:

本产品是一款主要面向20-30岁有许多碎片时间的游戏爱好者的微信小程序，玩家可以在游戏中搜集拥有不同属性和技能的角色以及相应的皮肤，同时还可以体会到不同难度的游戏内容和地图。

一般特性:

基于微信小程序的休闲类的角色搜集手游

开发、运行和维护的历史:

开发历史:

|  |  |
| --- | --- |
| 时间 | 内容 |
| 2020/10/1—2020/10/15 | 项目开发方向及形式确定 |
| 2020/10/19—2020/10/27 | 项目计划书 |
| 2020/10/28—2020/11/6 | 可行分析报告 |
| 2020/11/8—2020/11/29 | 软件需求说明书 |
| 2020/11/29— | 用户手册 |
| 2020/11/29— | 软件(结构)设计说明 |
| 2020/11/29— | 数据库(顶层)设计说明 |
| 2020/11/29— | 系统设计(结构设计)说明 |

暂无运行和维护的历史

投资方:G17组全体成员

需方:杨枨老师

用户: 20-30岁有许多碎片时间的游戏爱好者

开发方:G17组全体成员

### 1.3文档概述

本文档主要用于对本项目的软件设计说明，明确开发时应注意的要点，以及作为设计的说明书供需方和开发人员参考。

本文档在开发期间由SE2020-G17组全体成员编写，仅供SE2020-G17小组使用，不应进行商业性传播或为其他人员使用。

该软件设计说明文档对《基于微信开发者工具开发的飞翔的小鸟游戏》软件做了全面细致的设计说明文档。

本书的预期读者为评审组成员，项目组成员，当然用户和关键用户组

### 1.4与其他计划的关系

(若有)本条应描述本计划和有关的项目管理计划之间的关系。

### 1.5基线

项目计划书 -基于微信开发者工具开发的飞翔的小鸟游戏1.2

可行性报告-基于微信开发者工具开发的飞翔的小鸟游戏1.2

软件需求说明-基于微信开发者工具开发的飞翔的小鸟游戏1.0

项目设计-用户手册0.2

项目设计-软件(结构)设计说明0.4

项目设计-数据库(顶层)设计说明0.2

项目设计-系统设计(结构设计)说明0.3

## 2引用文件

本章应列出本文档引用的所有文档的编号、标题、修订版本和日期。本章也应标识不能通过正常的供货渠道获得的所有文档的来源。

[1]GB+T-8567-2006计算机软件文档编制规范 11 - 软件测试计划(STP)

[2]SE2020-G17-项目介绍正式版1.0.docx

[3]SE2020-G17-项目计划书1.2.docx

[4]SE2020-G17-可行性分析1.2.docx

[5]SE2020-G17-软件需求说明1.0.docx

[6]SE2020-G17-数据字典.docx

[7]SE2020-G17-用户手册0.2

[8]SE2020-G17-软件(结构)设计说明0.4

[9]SE2020-G17-数据库(顶层)设计说明0.2

[10]SE2020-G17-系统设计(结构设计)说明0.3

[11] 张海藩编著 《软件工程导论》（第六版） 清华大学出版社

## 3软件测试环境

本章应分条描述每一预计的测试现场的软件测试环境。可以引用软件开发计划(SDP)中所描述的资源。

### 3.1测试现场

各小组的寝室以及制定的网络人群对象，通过在固定时间进行测试。同时理四的教室，寝室也可以成为测试现场。

### 3.2软件项

**操作系统**：Windows 10 操作系统，Android10.0操作系统，IOS11.0操作系统

**编译程序**：IDEA 2020-02；微信开发者工具

**通信软件**：微信； QQ

**相关应用软件**：uniapp；powerdesigner;PhotoShop;墨刀； office系列

**数据库**：微信自带腾讯云数据库

**输入文件**：已准备好的各种测试文档或文件

**代码检查程序**：编译程序的自动检测

**测试数据产生器**： WeTest

**测试控制软件**：； WeTest

**其他专用测试软件**：

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16.00 GB (3300 MHz)；固态硬盘130G；移动硬盘1T；显卡为NVIDIA GeForce GTX 1060；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;一台苹果手机

**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。

**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;一台苹果手机。

**测试事件记录仪**：两台安卓系统手机;一台苹果手机。

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G17小组成员共同开发，所有权属于G17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G17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G17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，其中两位成员有web开发和制作的经验，一位同学对java较为熟悉，也有一位成员对前端较为熟悉。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。测试日期暂定为第5章所示时间，但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等。

### 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

主要包括游戏逻辑测试，队伍约局发布与接收测试，账号内容修改的测试，账号退出与注销的测试，动态发布与接受测试，历史约局信息界面统计信息的测试，信誉分统计测试、软件点击或滑动等操作是否有效的测试，交互是否友好等测试，响应时间测试，其他用户查看的测试等。

## 4计划

本章应描述计划测试的总范围并分条标识，并且描述本STP适用的每个测试。

### 4.1总体设计

#### 4.1.1测试集

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、Beta测试、UAT测试等级别。

#### 4.1.2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、β、第三方

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

本条应描述要执行的测试的类型或类别(例如，定时测试、错误输入测试、最大容量测试)。

#### 4.1.3一般测试条件

在测试中，通常需要给定一定的测试数据样例，比如在并发上最多可以有250位玩家同时在玩游戏

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。

本条应描述运用于所有测试或一组测试的条件，例如：“每个测试应包括额定值、最大值和最小值；”“每个x类型的测试都应使用真实数据(live data)；”“应度量每个CSCI执行的规模与时间。”并对要执行的测试程度和对所选测试程度的原理的陈述。测试程度应表示为某个已定义总量(如离散操作条件或值样本的数量)的百分比或其他抽样方法。也应包括再测试/回归测试所遵循的方法。

#### 4.1.4测试过程

**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

### 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

本章应包含或引用指导实施本计划中所标识测试的进度表。包括：

a.描述测试被安排的现场和指导测试的时间框架的列表或图表。

b.每个测试现场的进度表，(若适用)它可按时间顺序描述以下所列活动与事件，根据需要可附上支持性的叙述。

1)分配给测试主要部分的时间和现场测试的时间，

2)现场测试前，用于建立软件测试环境和其他设备、进行系统调试、定向培训和熟悉工作所需的时间；

3)测试所需的数据库/数据文件值、输入值和其他操作数据的集合；

4)实施测试,包括计划的重测试；

5)软件测试报告(STR)的准备、评审和批准。

## 6需求的可追踪性

本章应包括：

a.从本计划所标识的每个测试到它所涉及的CSCI需求和(若适用)软件系统需求的可追踪性(此可追踪性亦可在4.2.x.y中提供，而在此引用)。

b.从本测试计划所覆盖的每个CSCI需求和(若适用)软件系统需求到针对它的测试的可追踪性。这种可追踪性应覆盖所有适用的软件需求规格说明(SRS)和相关接口需求规格说明(IRS)中的CSCI需求，对于软件系统，还应覆盖所有适用的系统/子系统规格说明(SSS)及相关系统级IRS中的系统需求。

## 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，要有典型用户和当然用户以及由开发经验的用户参与；

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。由于本产品最主要的技术为数据库技术，因此，在bug分析中应该结合数据库的特性对bug进行分析。

同时，对于多项数据处理中，应采用数学方法和模型(V模型,W模型,X模型)对数据进行统一分析，

### 7.3结论

（TBD）

## 8注解

**1、软件开发背景**

本产品是一款主要面向20-30岁有许多碎片时间的游戏爱好者的微信小程序，玩家可以在游戏中搜集拥有不同属性和技能的角色以及相应的皮肤，同时还可以体会到不同难度的游戏内容和地图。

**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。不仅观念朱输入输出的正确性。同时也关注程序内部的情况。

## 附录

附录可用来提供那些为便于文档维护而单独出版的信息(例如图表、分类数据)。为便于处理，附录可单独装订成册。附录应按字母顺序(A, B等)编排。