# TSTimer项目

## 集成测试用例v1.0

目录

[集成测试用例v1.0 1](#_Toc518920549)

[1. 引言 2](#_Toc518920550)

1.1 介绍 2

1.2 目的 2

[1.3 范围 2](#_Toc518920551)

[2. 测试用例 3](#_Toc518920552)

2.1 用户注册 3

2.2 用户登录 4

[2.3 退出登录 5](#_Toc518920551)

2.4 打卡日历 6

2.5 查看计划表 7

[2.6 增加计划 9](#_Toc518920551)

2.7 删除计划 10

2.8 帮助与反馈 11

2.9 番茄时钟 12

## 引言

### 集成测试用例是一份描述具体测试步骤的文档，为集成测试而编制的一组包括测试的输入参数、条件及配置、预期的输出结果等的文档，以便测试模块之间数据接口是否满足某个特定需求或集成后的功能是否满足要求。

### *1.1 介绍*

**集成测试，也叫组装测试或联合测试。在单元测试的基础上，将所有模块按照设计要求（如根据结构图）组装成为子系统或系统，进行集成测试。例如：正在测试CunsomerData，它依赖于LINQ to SQL数据库连接，在集成测试中，确保能正常调动所有相关类的方法。测试方法一般选用黑盒测试和白盒测试相结合。**

### *1.2 目的*

### 集成测试是将已测试过的模块按照概要设计要求组装起来进行测试，主要目标是检查模块间接口是否正确，检查各个模块之间的通信和相互调用是否符合需求，属于灰盒测试。编写集成测试用例的目的：整理测试思路，防止在做集成测试的时候漫无目的，或出现思维盲区导致未测试到所有情况；记录测试结果。

### *1.3 范围*

### 主要测试模块接口之间数据传输是否正确、模块集成后的功能是否实现、模块接口功能与设计需求是否一致。

## 测试用例

**测试环境：**

**虚拟机——windows 7、Mac os**

**真机——荣耀手机：Mediatek MT6739 CPU 1.5GHz**

**小米手机：骁龙625 CPU 2.0GHz**

### *2.1 用户注册*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 01 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户点击主页左上角图标  3.用户点击登录按钮  4.用户在登录界面点击注册按钮  5.用户依次填写用户名、密码、确认密码、电话，在验证码一栏点击获取验证码，输入电话号码收到的验证码  6.用户点击注册按钮 | | | |
| 预期输出 | 显示注册成功，回到应用程序主页 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.2 用户登录*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 02 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户点击主页左上角图标  3.用户点击登录按钮  4.用户在登录界面输入用户名和密码  5.用户点击登录按钮  忘记密码，点击对应按钮  1.用户填写注册电话  2.用户点击获取验证码  3.用户输入注册电话收到的验证码  4.用户点击确认按钮  5.用户在新界面输入新密码、确认密码  6.用户点击确认按钮 | | | |
| 预期输出 | 成功登录后跳转到应用程序主页 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.3 退出登录*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 03 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并已经成功登录 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户点击主页左上角图标  3.用户点击退出登录按钮 | | | |
| 预期输出 | 成功退出登录后返回应用程序主页 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.4 打卡日历*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 04 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2 用户点击主页左上角图标  3.用户点击我的统计数据  4.用户查看各个项目及对应番茄树  5.用户点击具体项目，查看以日期分布的番茄树 | | | |
| 预期输出 | 显示以日期为横轴的具体项目的番茄数 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.5 查看计划表*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 05 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2 用户点击主页左上角图标  3.用户点击我的计划表  查看计划外事件  1.用户进入应用程序主页  2 用户点击主页左上角图标  3.用户点击计划外事件  4.点击添加计划外事件  在番茄时钟工作页面也可点击计划外事件进行查看和添加 | | | |
| 预期输出 | 显示已添加的计划事件；显示计划外事件 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.6 增加计划*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 06 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户点击主页右下角的“+”按钮  3.用户在跳出的增加日程框中，点击设置按钮  4.用户填写活动事项并设置番茄时间、番茄树和休息时间  5.用户点击添加按钮 | | | |
| 预期输出 | 跳转回应用程序主页，主页已建事项中显示刚添加的事项 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.7 删除计划*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 07 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件且软件内有已建事项 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户查看已建事项  3.用户长按已建事项的最右侧，事项右侧显示删除按钮  4.用户点击删除按钮 | | | |
| 预期输出 | 刚操作的删除事项消失在应用程序主页已建事项中 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.8 帮助与反馈*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 08 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户点击右上角“？”按钮  3.用户查看帮助界面，可点击热点问题解答基本困惑  4.用户点击下方的用户反馈  5.用户填写问题和意见及联系电话，可上传图片  6.点击提交按钮 | | | |
| 预期输出 | 显示提交成功并跳转回帮助页面 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |

### *2.9 番茄时钟*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例序号 | 09 | 版本号 | 0.1 | |
| 测试环境 | 相同 | | | |
| 测试目的 | 验证android端TSTimer功能是否符合要求 | | | |
| 前提条件 | 一部已经安装了TSTimer的测试手机并打开软件 | | | |
| 测试步骤 | 1.用户进入应用程序主页  2.用户点击已建事项  3.用户在时钟界面点击时钟图标，变为番茄即开始计时  4.倒计时完成后，用户点击计时结束  在计时结束前，用户可点击计划外事件按钮进行查看和添加；用户点击界面左上角，弹出确定退出番茄时钟对话框。若点击取消按钮，继续计时；若点击确定按钮，弹出“专注失败，失去番茄”，单击回到应用程序主页 | | | |
| 预期输出 | 计时结束后跳转回应用程序主页 | | | |
| 实际输出 | 与预期输出相同： （）是 （ ）否 | | | |
| 结论 | 用例测试是否通过：（）是 （ ）否  第一次回归测试是否通过：（）是 （ ）否  第二次回归测试是否通过：（）是 （ ）否 | | | |
| 测试用例设计人 | 石小曼 | 设计日期 | | 2018.7.12 |
| 测试用例执行人 |  | 测试日期 | |  |
| 第一次回归测试执行人员 |  | 第一次回归测试日期 | |  |
| 第二次回归测试执行人员 |  | 第二次回归测试日期 | |  |
| 开发人员修改意见 | （）同意修改 （）不必修改 （）暂时不修改 | | | |
| 问题修改摘要及日期 | 第一次回归测试： | | | |
| 第二次回归测试： | | | |
| 修改人 |  | | | |