

**<< KeepFit健身App >>**

**软件测试计划书**



北京航空航天大学

2011-09

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| A2015-00-03-00 | 2015/1/3 | 宋丛溪 | 马元 | 对软件测试计划的说明 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目 录

[1引言 1](#_Toc439543825)

[1.1标识 1](#_Toc439543826)

[1.1编写目的 1](#_Toc439543827)

[1.2背景 1](#_Toc439543828)

[1.3定义 2](#_Toc439543829)

[1.4参考资料 2](#_Toc439543830)

[2测试计划 2](#_Toc439543831)

[2.1软件说明 2](#_Toc439543832)

[2.2测试内容 3](#_Toc439543833)

[2.3测试进度安排 10](#_Toc439543834)

[2.4测试条件 11](#_Toc439543835)

[3测试用例设计 11](#_Toc439543836)

[3.1类级划分测试（单元测试） 11](#_Toc439543837)

[3.2基于线程的测试（集成测试） 20](#_Toc439543838)

[3.3基于场景的测试（确认测试） 26](#_Toc439543839)

[3.4.压力测试（系统测试） 34](#_Toc439543840)

[4评价准则 35](#_Toc439543841)

[4.1范围 35](#_Toc439543842)

[4.2数据整理 35](#_Toc439543843)

[4.3尺度 35](#_Toc439543844)

**软件测试计划书**

# 1引言

## 1.1标识

适用系统：Android 5.0、Android 6.0

标识号：keepfit\_1.0

标题：KeepFit健身App

版本号：1.0

文档标识号：A2015-00-03-00

文档标题：软件测试计划书

版本号：1.0

发型号：1.0

## 1.1编写目的

本文档的用途在于详尽描述对KeepFit健身App的软件测试策略：描述测试计划、测试用例的设计，记录测试执行及测试结果，最后进行测试结果的评估。

文档的引言部分叙述了本文档适用的系统和软件的完整标识，介绍了软件测试前所完成的工作，列出参考资料。文档的测试计划部分首先分析了软件的功能以供后续测试计划需要，而后总结出全部测试内容的提纲，接下来根据提纲规划测试进度，最后明确测试所需具备的条件。文档的测试用例设计部分首先确定了从单元测试到集成测试到确定测试再到系统测试所用的具体方法，详细地列出了每一个测试用例的设计，包括驱动用例的描述、测试过程的描述、测试可能产生的结果集合、可能抛出的异常。文档的评价准则部分首先说明所选择的测试用例能够接查的范围及其局限性，而后整理结果数据以供评价，最后确定软件可容忍的通过尺度。

软件测试是软件质量保证的重要元素之一，很多软件开发组织将30%—40%甚至更多的项目资源用在测试上，软件测试技术和软件测试策略受到了高度的重视和广泛的应用。故编写测试文档旨在统筹规划合理的测试策略，有计划地开展测试，由小范围过渡到软件整体，及时修补小构件中的漏洞，通过开发人员及时调试，有助于提升系统的稳定性、健壮性，保证系统的质量。

本文档的预期读者是设计人员、开发人员、项目管理人员、测试人员、用户，出于私密性和保密性考虑，除了这些人以外的其他人员都无权阅读该文档。

## 1.2背景

测试计划从属于本团队开发的“KeepFit健身App”。 该项目于10月15日立项，10月15日至10月31日召开4次团队会议进行人员分工，并于10月31日提交软件开发计划书，11月30日需求分析人员完成需求分析并提交软件需求说明书，12月20日设计人员完成软件设计并提交软件设计说明。

在执行本测试计划前必须完成工程整体流程的规划，团队人员的分工，软件的需求分析，软件的设计工作，软件的开发可以与本测试同步执行，实现单个类的代码之后便可以进行单元测试，实现多个类协同的逻辑后便可以进行集成测试，确认测试和系统测试要在整个软件系统开发完成后执行。

## 1.3定义

OOA: 是确定需求或者业务的角度，按照面向对象的思想来分析业务。

OOD: 其主要作用是对OOA分析的结果作进一步的规范化整理

CRC: 阐述类、类的行为和类的责任

## 1.4参考资料

[1] KeepFit健身App项目开发计划书（1.0） 2015

[2] KeepFit健身App系统规格需求说明书（1.0） 2015

[3] KeepFit健身App项目设计说明（1.0） 2015

[4] 《软件工程》 Roger S.Pressman 机械工业出版社 2015

# 2测试计划

## 2.1软件说明

“KeepFit健身App”软件系统的功能是根据用户需求和自身条件，提供相应的健身语音视频，记录用户的健身进展并在相应时间有闹钟提醒，用户可将健身进展和成效分享到软件自带或其他社交平台。其核心功能如图1所示

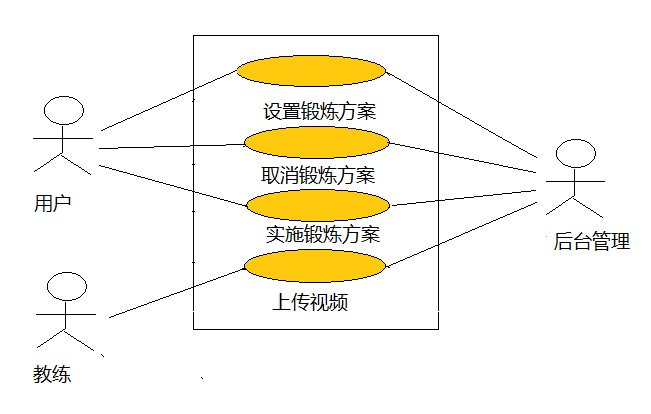


图 1

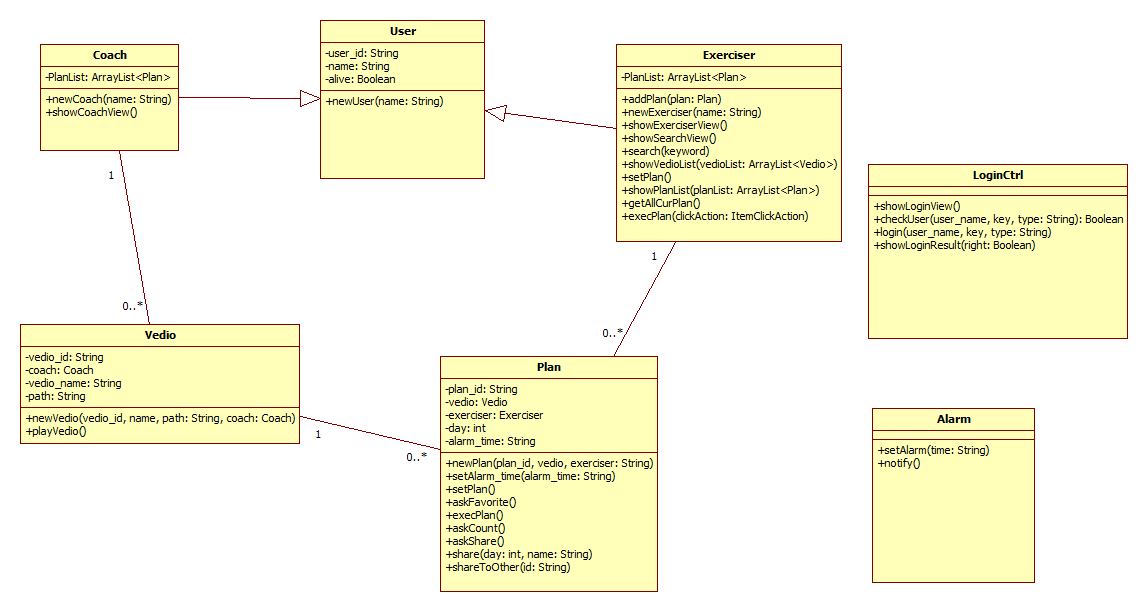
## 2.2测试内容

本系统采用面向对象的设计构建，所以测试也是基于面向对象的测试方法。

2.2.1单元测试

采用类级的划分测试，测试单个类及封装。

需要测试的类及功能如下图所示



2.2.2集成测试

采用基于线程的测试方法，每个线程单独集成和测试，并应用回归测试确保不产生副作用。

构成线程测试的类间关系如下图所示

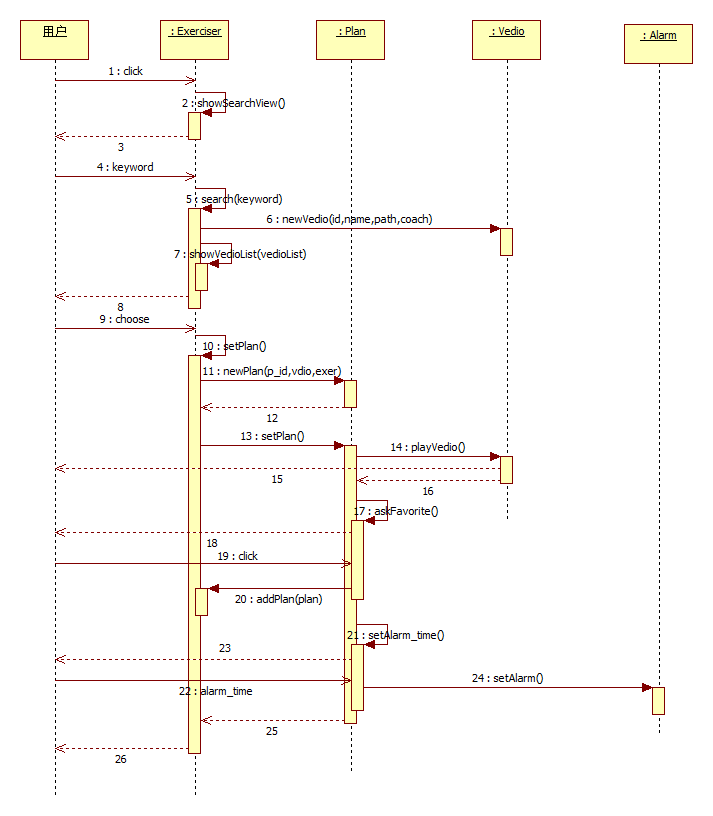


图 2设置方案序列图

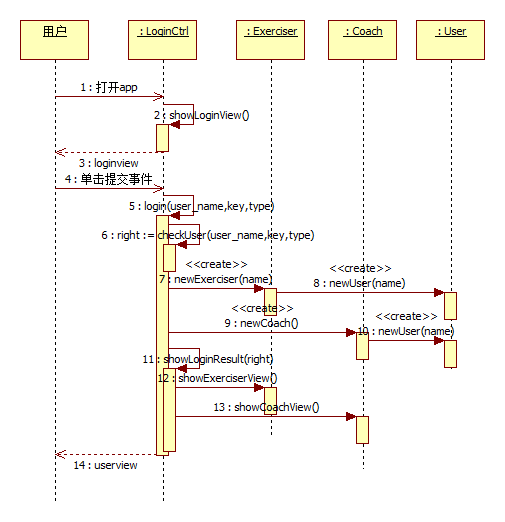


图 3登录序列图

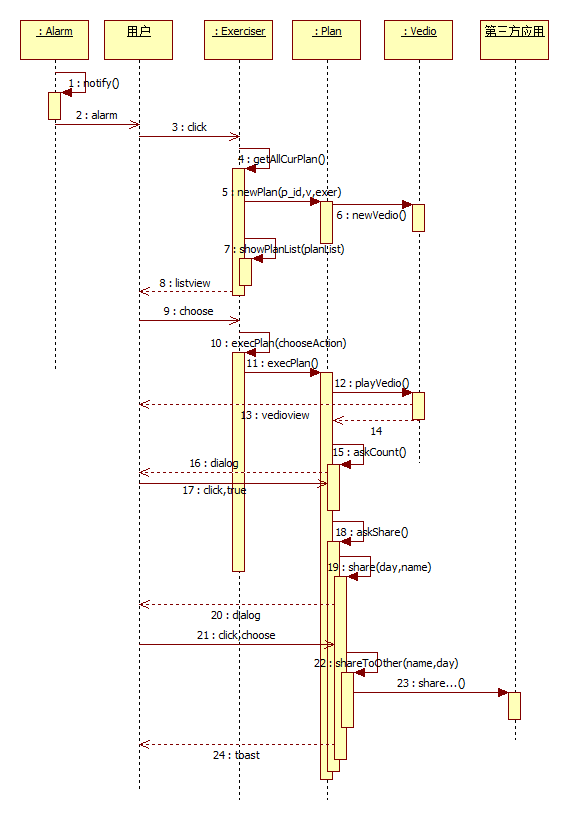


图 4执行计划序列图

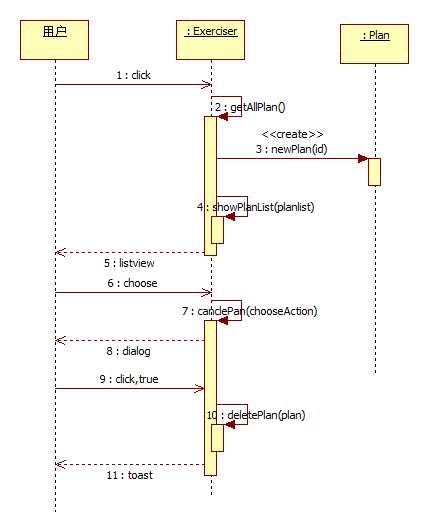


图 5取消计划序列图

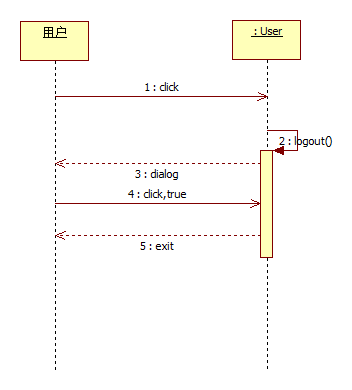


图 6登出序列图

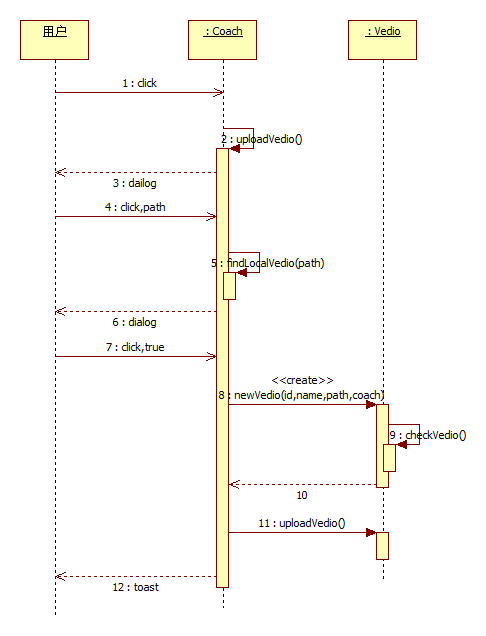


图 7上传视频序列图

2.2.3确认测试

采用基于场景的测试方法，通过用例捕获用户必须完成的任务，然后在测试时使用它们及其变体。

设计的测试用例将满足下表所有的功能需求。

表 1

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 功能名称 | 用例定义 | 参与者 |
| 1 | 登录系统 | 以用户身份登录该app | 用户 |
| 2 | 观看视频 | 用户搜索视频并观看 | 用户 |
| 3 | 收藏视频 | 用户收藏自己喜欢的视频 | 用户 |
| 4 | 设置锻炼时间 | 用户对自己收藏过的视频设置锻炼时间 | 用户 |
| 5 | 取消锻炼方案 | 用户取消自己的锻炼方案，即将相应锻炼视频从收藏中删掉，取消闹钟提醒 | 用户 |
| 6 | 发布成果 | 用户向社交平台发布自己的锻炼成果 | 用户 |
| 7 | 打卡 | 用户观看完自己收藏的一个视频后，打卡记录自己的锻炼天数 | 用户 |
| 8 | 登出系统 | 用户登出该app | 用户 |
| 9 | 登入系统 | 以教练身份登录该app | 教练 |
| 10 | 上传视频 | 教练向该app上传视频 | 教练 |
| 11 | 登出系统 | 教练登出系统 | 教练 |
| 12 | 获取视频列表 | 获得app中拥有的所有视频 | 管理者 |
| 13 | 增加收藏视频 | 在用户收藏列表中增加该视频 | 管理者 |
| 14 | 设置锻炼时间 | 为该视频设置好锻炼时间 | 管理者 |
| 15 | 获取用户信息 | 获取用户的锻炼方案、锻炼天数、收藏视频等信息 | 管理者 |
| 16 | 取消锻炼方案 | 系统将相应锻炼视频从用户收藏中删掉，取消闹钟提醒 | 管理者 |
| 17 | 发布成果 | 相关社交平台发布一条该用户的锻炼成果的信息 | 管理者 |
| 18 | 增加锻炼天数 | 用户锻炼该方案的天数增加1 | 管理者 |
| 19 | 修改视频列表 | 系统将教练上传的视频增加到视频集中供用户选择 | 管理者 |

2.2.4系统测试

进行压力测试，用非正常的数量、频率、容量执行系统。本次压力测试通过loadrunner软件生成虚拟用户进行测试。录制不同的操作脚本，生成多个vuser，构建场景并每隔固定的时间增加vuser的数量达到加压，记录响应时间、资源占用等性能数据以及出错情况。

这几种测试是螺旋式前进的。

## 2.3测试进度安排

表 2

|  |  |  |
| --- | --- | --- |
| 日期 | 前提 | 工作内容 |
| 2015-12-20  至2015-12-30 | 设计人员完成类的设计 | 根据相应类设计单元测试，根据相应类及之间的关系设计集成测试，根据软件功能设计确认测试，根据系统整体设计系统测试 |
| 2015-12-31  至2016-1-31 | 开发人员完成单个类代码的编写 | 进行单元测试，记录类的状态 |
|  | 开发人员完成具有协同关系的多个类的编写 | 进行集成测试，记录类的状态 |
| 2016-2-1  至2016-2-3 | 开发人员完成全部系统的编写 | 进行确认测试，进行系统测试，记录测试结果 |
| 2016-2-4  至2016-2-6 | 测试结果 | 将测试结果进行整理评估 |

## 2.4测试条件

有关本项任务的文件：KeepFit健身App系统规格需求说明书（1.0）、 KeepFit健身App项目设计说明（1.0）

被测试程序及其开发环境：KeepFit的安卓工程，Eclipse

测试用例构造工具：JUINT单元测试框架，LoadRunner

测试设备：安卓手机,服务器

测试人员：1人

# 3测试用例设计

## 3.1类级划分测试（单元测试）

### 3.1.1测试1-1（Exerciser类测试1）

#### 3.1.1.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.1.2操作序列

newExcerciser

#### 3.1.1.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

### 3.1.2测试1-2（Exerciser类测试2）

#### 3.1.2.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.2.2操作序列

newExerciser->showSearchView->search->showVideoList

#### 3.1.2.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

3) showSearchView没有成功显示

4) showVideoList没有显示或显示结果有误

### 3.1.3测试1-3（Exerciser类测试3）

#### 3.1.3.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.7.2操作序列

newExerciser->showSearchView->search->showVideoList->setPlan->addPlan

#### 3.1.3.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

3) showSearchView没有成功显示

4) showVideoList没有显示或显示结果有误

5)addPlan输入的plan有误无法加入planlist

6) plan没有加入到planlist的指定位置

### 3.1.4测试1-4（Exerciser类测试4）

#### 3.1.4.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.4.2操作序列

newExerciser->getAllCurPlan->showPlanList->execPlan

#### 3.1.4.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

3) showPlanList没有输出期望结果

4) execPlan执行错误

### 3.1.5测试1-5（Exerciser类测试5）

#### 3.1.5.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.5.2操作序列

newExerciser->getAllPlan->showPlanList->cancelPlan->deletePlan

#### 3.1.5.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

3) showPlanList没有输出期望结果

4) deletePlan没有成功将planlist指定项删除

5) deletePlan的plan在planlist中不存在

6）deletePlan的plan有误

### 3.1.6测试1-6（Coach类测试1）

#### 3.1.6.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.6.2操作序列

newCoach

#### 3.1.6.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

### 3.1.7测试1-7（Coach类测试2）

#### 3.1.7.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |
| Planlist | Arraylist<Plan> |

#### 3.1.7.2操作序列

newExerciser->upLoadVideo->findLocalVideo

#### 3.1.7.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

3) findLocalVideo路径输入不合法

4) 路径对应的视频上传失败

5) 视频上传后没有添加到planlist中

### 3.1.8测试1-8（User类测试1）

#### 3.1.8.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| User\_id | String |
| name | String |
| alive | Boolean |

#### 3.1.8.2操作序列

newUser->logout

#### 3.1.8.3可能发生异常

1) 输入user\_id\name不合法

2) alive没有被操作更改

3) logout后alive没有被置成false

### 3.1.9测试1-9（Video类测试1）

#### 3.1.9.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| Video\_id | Int |
| coach | Coach |
| Video\_name | String |
| path | String |

#### 3.1.9.2操作序列

newVideo

#### 3.1.9.3可能发生异常

输入video\_id\video\_name\coach\path不合法

### 3.1.10测试1-10（Video类测试2）

#### 3.1.10.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| Video\_id | Int |
| coach | Coach |
| Video\_name | String |
| path | String |

#### 3.1.10.2操作序列

newVideo->playVideo

#### 3.1.10.3可能发生异常

1)输入video\_id\video\_name\coach\path不合法

2)播放视频执行不成功

### 3.1.11测试1-11（Video类测试3）

#### 3.1.11.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| Video\_id | Int |
| coach | Coach |
| Video\_name | String |
| path | String |

#### 3.1.11.2操作序列

newVideo->checkVideo->upLoadVideo

#### 3.1.11.3可能发生异常

1)输入video\_id\video\_name\coach\path不合法

2)checkVideo没有得到预期结果

3)视频上传失败

### 3.1.12测试1-12（Plan类测试1）

#### 3.1.12.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| plan\_id | Int |
| video | Video |
| exerciser | Exerciser |
| day | int |
| Alarm\_time | String |

#### 3.1.12.2操作序列

Newplan->setPlan-> execPlan->askCount

#### 3.1.12.3可能发生异常

1)输入plan\_id\video\exerciser\day\alarm\_time不合法

2)execPlan执行失败

3) askCount没有修改day

### 3.1.13测试1-13（Plan类测试2）

#### 3.1.13.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| plan\_id | Int |
| video | Video |
| exerciser | Exerciser |
| day | int |
| Alarm\_time | String |

#### 3.1.13.2操作序列

Newplan->askFavorite->setAlarm\_time

#### 3.1.13.3可能发生异常

1)输入plan\_id\video\exerciser\day\alarm\_time不合法

2)alarm\_time没有修改

### 3.1.14测试1-14（Plan类测试3）

#### 3.1.14.1指定状态列表

|  |  |
| --- | --- |
| 状态名 | 类型 |
| plan\_id | Int |
| video | Video |
| exerciser | Exerciser |
| day | int |
| Alarm\_time | String |

#### 3.1.14.2操作序列

Newplan->askShare->share->shareToOther

#### 3.1.14.3可能发生异常

1)输入plan\_id\video\exerciser\day\alarm\_time不合法

2)share没有成功执行

## 3.2基于线程的测试（集成测试）

将相应系统的一个输入或一个时间所需要的一组类集成到一起。每个线程单独集成和测试，并应用回归测试确保不产生副作用。

### 3.2.1测试2-1（打开app）

#### 3.2.1.1涉及的类

1)LoginCtrl

#### 3.2.1.2操作序列

showLoginView

#### 3.2.1.3期望输出

Loginview

### 3.2.2测试2-2(click登录)

#### 3.2.2.1涉及的类

1)LoginCtrl

2)Exerciser

3)Coach

4)User

#### 3.2.2.2操作序列

Login->checkUser----right&&type==exerciser---->newExerciser-> newUser->showLoginResult->showExcerciserview

----right&&type==coach------->newCoach-> newUser->showLoginResult->showCoachView

----wrong->showLoginResult

#### 3.2.2.3期望输出

userview

### 3.2.3测试2-3(alarm)

#### 3.2.3.1涉及的类

1)Alarm

2)Exerciser

3)Plan

4)Video

#### 3.2.3.2操作序列

Notify->alarm->getAllCurPlan->newPlan->newVideo->showPlanList

#### 3.2.3.3期望输出

listview

### 3.2.4测试2-4(choose执行计划)

#### 3.2.4.1涉及的类

1)Alarm

2)Exerciser

3)Plan

4)Video

#### 3.2.4.2操作序列

Notify->alarm->execPlan->playVideo->askCount

#### 3.2.4.3期望输出

1）Videoview

2）dialog

### 3.2.5测试2-5(click打卡)

#### 3.2.5.1涉及的类

1)Alarm

2)Exerciser

3)Plan

4)Video

#### 3.2.5.2操作序列

Notify->alarm->execPlan->playVideo->askCount-----click true---->askShare->share

#### 3.2.5.3期望输出

1）Videoview

2）dialog

3）dialog

### 3.2.6测试2-6(click分享)

#### 3.2.6.1涉及的类

1)Alarm

2)Exerciser

3)Plan

4)Video

#### 3.2.6.2操作序列

Notify->alarm->execPlan->playVideo->askCount-----click true---->askShare->share-----click true---->shareToOther

#### 3.2.6.3期望输出

1）Videoview

2）dialog

3）dialog

4）tbast

### 3.2.7测试2-7(click取消方案)

#### 3.2.7.1涉及的类

1)Exerciser

2)Plan

#### 3.2.7.2操作序列

getAllPlan->newPlan->showPlanList

#### 3.2.7.3期望输出

listview

### 3.2.8测试2-8(choose取消方案)

#### 3.2.8.1涉及的类

1)Exerciser

2)Plan

#### 3.2.8.2操作序列

getAllPlan->newPlan->showPlanList->cancelPlan

#### 3.2.8.3期望输出

1)listview

2)dialog

### 3.2.9测试2-9(clicktrue取消方案)

#### 3.2.9.1涉及的类

1)Exerciser

2)Plan

#### 3.2.9.2操作序列

getAllPlan->newPlan->showPlanList->cancelPlan-->deletePlan

#### 3.2.9.3期望输出

1)listview

2)dialog

3)toast

### 3.2.10测试2-10(click登出)

#### 3.2.10.1涉及的类

User

#### 3.2.10.2操作序列

logout

#### 3.2.10.3期望输出

Dialog

### 3.2.11测试2-11(click登出)

#### 3.2.11.1涉及的类

User

#### 3.2.11.2操作序列

logout

#### 3.2.11.3期望输出

1)dialog

2)exit

### 3.2.12测试2-12(click上传)

#### 3.2.12.1涉及的类

1）Coach

2）Video

#### 3.2.12.2操作序列

uploadVideo

#### 3.2.12.3期望输出

dialog

### 3.2.13测试2-13(clickpath上传)

#### 3.2.13.1涉及的类

1）Coach

2）Video

#### 3.2.13.2操作序列

uploadVideo->findLocalVideo

#### 3.2.13.3期望输出

1)dialog

2)dialog

### 3.2.14测试2-14(clicktrue上传)

#### 3.2.13.1涉及的类

1）Coach

2）Video

#### 3.2.14.2操作序列

uploadVideo->findLocalVideo->newVideo->checkVideo->uploadVideo

#### 3.2.14.3期望输出

1)dialog

2)dialog

3)toast

## 3.3基于场景的测试（确认测试）

### 3.3.1测试3-1（用户登录系统）

#### 3.3.1.1测试背景

用户打开KeepFit健身app

#### 3.3.1.2事件序列

1. 系统显示登录界面
2. 用户输入用户名的密码
3. 用户选择“用户”身份登入系统
4. 系统对用户名和密码进行验证
5. 系统跳转至app使用界面

#### 3.3.1.3异常及处理

1. 如果在系统还未跳转至app使用界面时，用户选择取消，则系统停留在登录界面
2. 如果验证信息不正确，显示提示信息，页面停留在登录界面

### 3.3.2测试3-2（用户设置锻炼方案）

#### 3.3.2.1测试背景

一个合法用户已经登录到该app

#### 3.3.2.2事件序列

1. 当用户选择“设置锻炼方案”时，用例开始
2. 系统跳转至视频搜索界面
3. 用户输入要搜索的视频关键字
4. 用户点击搜索
5. 系统显示所有符合要求的视频
6. 用户选择要观看的视频
7. 用户观看完视频
8. 系统询问是否收藏该视频
9. 用户选择“是”
10. 系统跳转至锻炼时间设置界面
11. 用户设置每日锻炼时间
12. 用户点击确定
13. 系统显示设置锻炼方案成功
14. 锻炼方案被保存在系统中

#### 3.3.2.3异常及处理

1） 在第4步，如果系统中没有符合要求的视频，系统显示未找到相关视频，返回第2步。

2） 在以上各步，若用户选择“取消”，则系统跳转至app首页。

### 3.3.3测试3-3（用户实施锻炼方案）

#### 3.3.3.1测试背景

闹钟提醒用户该进行今日锻炼

#### 3.3.3.2事件序列

1. 用户选择“开始锻炼”，用例开始
2. 系统显示所有该时间用户设置过的锻炼方案
3. 用户选择要锻炼的锻炼方案
4. 用户根据视频完成今日锻炼任务
5. 系统询问用户是否打卡
6. 用户选择“是”
7. 系统显示“打卡成功”
8. 系统将今日锻炼记入天数
9. 系统询问是否发布锻炼成果
10. 用户选择“是”
11. 系统显示若干社交平台供用户选择
12. 用户选择某一款社交软件
13. 系统显示“分享成功”
14. 在该社交平台发布该用户锻炼该视频天数的信息。系统显示所有该时间用户设置过的锻炼方案

#### 3.3.3.3异常及处理

1. 在第2步，若用户没有收藏的视频，则系统提示“没有可以实施的锻炼方案”，系统返回app首页用户选择要锻炼的锻炼方案
2. 若在第6步，用户选择“取消”，则系统不将今日锻炼记入天数，返回第2步
3. 若在第9步，用户选择“取消”，则不分享该方案的锻炼成果，返回第2步。
4. 在其他各步，若用户选择“取消”，系统跳转至app首页，用例结束。

### 3.3.4测试3-4（用户取消锻炼方案）

#### 3.3.4.1测试背景

一个合法用户已经登录到该app

#### 3.3.4.2事件序列

1. 当用户选择“取消锻炼方案”时，用例开始
2. 系统显示所有用户设置过的锻炼方案
3. 用户选择要取消的锻炼方案
4. 系统询问用户是否取消方案
5. 用户选择“是”
6. 系统显示取消锻炼方案成功
7. 系统在用户收藏的视频删除该视频，并且取消与该视频相关的锻炼时间和闹钟提醒

#### 3.3.4.3异常及处理

1. 在第2步，若用户没有收藏的视频，则系统提示“没有可以删除的锻炼方案”，系统返回app首页系统显示所有用户设置过的锻炼方案
2. 在以上各步，若用户选择“取消”，则系统跳转至app首页

### 3.3.5测试3-5（用户登出系统）

#### 3.3.5.1测试背景

一个合法用户已经登录到该app

#### 3.3.5.2事件序列

1. 当用户选择“登出系统”时，用例开始
2. 系统询问用户是否登出
3. 用户选择“是”
4. 退出系统

#### 3.3.5.3异常及处理

1. 若在第3步，用户选择“否”，则留在原页面

### 3.3.5测试3-5（教练登录系统）

#### 3.3.5.1测试背景

教练打开KeepFit健身app

#### 3.3.5.2事件序列

1. 系统显示登录界面
2. 用户输入用户名的密码
3. 用户选择“教练”身份登入系统
4. 系统对用户名和密码进行验证

5） 系统跳转至app使用界面

#### 3.3.5.3异常及处理

1）如果在系统还未跳转至app使用界面时，教练选择取消，则系统停留在登录界面

2） 如果验证信息不正确，显示提示信息，页面停留在登录界面

### 3.3.6测试3-6（教练上传视频）

#### 3.3.6.1测试背景

一个合法教练已经登录到该app

#### 3.3.6.2事件序列

1. 当教练选择“上传视频”时，用例开始
2. 系统提示教练，输入要上传视频的本地路径
3. 教练输入本地路径
4. 系统找到该视频，询问教练是否确定上传
5. 教练选择“是”
6. 系统显示视频上传成功
7. 系统中增加一个可供用户选择的视频

#### 3.3.6.3异常及处理

1）在第3步，若教练输入的本地路径错误，系统提示“输入路径错误”，返回第2步。

2）若在第5步，教练选择“否”，则返回第2步。

3）若在其他各步中，教练选择“取消”，则系统返回app首页3.4系统测试

### 3.3.7测试3-7（教练登出系统）

#### 3.3.7.1测试背景

一个合法教练已经登录到该app

#### 3.3.7.2事件序列

1. 当教练选择“登出系统”时，用例开始
2. 系统询问教练是否登出
3. 教练选择“是”
4. 退出系统

#### 3.3.7.3异常及处理

1）若在第3步，教练选择“否”，则留在原页面

### 3.3.8测试3-8（修改视频列表）

#### 3.3.8.1测试背景

系统接收到一个教练上传的视频

#### 3.3.8.2事件序列

1. 检查该视频的质量、完整性、清晰度
2. 检查无误后，后台管理者获取视频列表
3. 在视频列表中添加新视频

#### 3.3.8.3异常及处理

1）若视频质量有问题，则后台管理不将视频添加至视频列表，给教练发送通知详述上传失败原因

### 3.3.9测试3-9（修改用户锻炼方案）

#### 3.3.9.1测试背景

系统显示用户选择了修改锻炼方案的行为

#### 3.3.9.2事件序列

1. 获取用户的锻炼方案列表和视频收藏列表
2. 若用户要取消锻炼方案
3. 在用户收藏的视频中删除该方案中的视频
4. 取消该方案中的锻炼时间的闹钟提醒
5. 若用户要增加锻炼方案
6. 将用户选择的视频加入用户收藏列表
7. 在用户设定的锻炼时间里设置闹钟提醒

C ) 将新生成的锻炼方案加入锻炼方案列表

#### 3.3.9.3异常及处理

1）若在第3.B步，用户未设置锻炼时间，则没有闹钟提醒，不生成相应锻炼方案

### 3.3.10测试3-10（修改用户锻炼天数）

#### 3.3.10.1测试背景

用户选择了“打卡”

#### 3.3.10.2事件序列

1. 获取用户实施该锻炼方案的天数
2. 将锻炼天数加1

### 3.3.11测试3-11（发布成果）

#### 3.3.11.1测试背景

用户选择了“发布锻炼成果”

#### 3.3.11.2事件序列

1. 获取该锻炼方案详情，包括视频名称、上传教练、用户锻炼天数
2. 将上述信息发布至用户选择的社交平台

## 3.4.压力测试（系统测试）

### 3.4.1测试准备

安卓手机，安装有loadrunner的计算机，服务器

### 3.4.2 测试流程

#### 3.4.2.1录制脚本

1） 配置测试用手机的wifi连接HTTP代理地址和端口

2） 打开loadrunner脚本编辑器，选择录制协议为HTTP协议，同时设置录制模式为代理，端口填写手机上设置的端口

3） 点击录制，打开KeepFit app 操作需要录制的业务

业务包括：用户、教练、管理者三种身份可以进行的各种操作的组合

4） 结束录制

#### 3.4.2.2创建负载测试场景

1） 打开loadrunner controller

2） 新建场景，选择手动场景

3） 更改脚本的参数生成不同的vuser

4) 添加脚本

5） 初始化vuser，在开始设置10个vuser，每30秒启动再两个

6） 配置windows资源监控器

#### 3.4.2.3 运行负载测试

1) 开始场景

2） 检查性能图，观察每个事物所用的时间、在指定时间运行的user数目

3） 观察windows资源图

4） 记录数据与异常

# 4评价准则

## 4.1范围

单元测试能够涵盖所有类，因类的操作本质隐含了一些限制，有的操作必须跟在一些操作之后执行。单元测试的操作序列是构建在这些限制的基础上的。

确认测试是根据需求分析而逐条构造，确保满足用户的所有功能需求，具有所有的行为特征。

系统测试运用了压力测试的方法可以测试系统在设定的大量用户操作的情况下的反映状况。由于测试用户数量上升的过程有时间限制，所以无法预知更大量的用户操作会给系统带来的影响。

## 4.2数据整理

因单元测试、集成测试的结果均是功能的显示，所以只需记录功能是否实现成功，测试结果采用手工方式记录。

确认测试需要记录每个测试完成所需时间，并与最大允许响应时间进行比较

系统测试需要将此时对应的vuser数量，再记录事物响应时间、吞吐率、内存占用情况，从loadrunner导入到excel中，与此时最大允许的值进行比较。特别地对于响应时间，将事物全部响应时间进行排序然后求90%数据中的最大值。

## 4.3尺度

单元测试和集成测试输出结果不容许偏离

保证测试用机状态良好，确认测试的每个操作进行时，响应时间小于3s

保证测试用机状态良好，系统测试的响应时间小于10s