**NBA信息查询平台**

**单元测试用例文档**

**（迭代一适用）**

**（V1.0）**

**南京大学软件学院**

**羽见青柠工作组**

金翠黄涵倩王宁(79)严顺宽

**2015年3月14日**

# 更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| **修改人员** | **日期** | **变更原因** | **版本号** |
| 王宁 | 2015.3.14 | 编制文档初稿 | 1.0 |

# 1.单元测试计划概述

本文档为在NBA信息查询平台进行单元测试时提供的有关任务安排、方法、资源和进度方面的指导，目标是从代码中发现缺陷，以备后续移除缺陷参考，提高软件质量，最终达到用户需求。

本文档的读者主要是开发工程师。执行人员有4人，将执行3个工作日。

参考文献：NBA信息查询平台需求规格说明文档、NBA信息查询平台项目设计文档。

# 2.单元测试的范围

单元测试必须能够展示出NBA信息查询平台中类的每个方法都能正常工作。单元测试的概念是将一个单元和该系统的其余部分隔离开来独立测试，因此可能需要创建测试驱动（test driver）代码。要求测试驱动代码用JUnit编写，当被测单元被提交准备完成项目构建时，相关的测试驱动代码必须已经被提交到团队的代码库中。在第三部分将给出隔离测试代码的方法指导。

注意：JUnit不能进行GUI测试。

# 3.单元测试的策略

1. 对方法（method）的单元测试

* 检查对所有调用对象的使用
* 验证对所有数据结构的处理
* 验证对所有文件的处理
* 验证控制流的不变性
* 检查所有循环的正常终止
* 检查所有循环的异常终止
* 验证所有错误条件的处理
* 检查数据库操作的正确性

1. 对类（class）的单元测试
2. 结合方法的执行，推荐对类的测试

* 选择最常见的执行序列
* 需要包括可能导致缺陷的序列
* 手工计算预期的属性值

1. 集中对每个属性进行单元测试

* 初始化，然后执行会影响属性的方法序列

1. 验证每个类的不变性

* 验证默认初始值时的不变性
* 执行方法序列
* 验证不变性仍然正确

# 4.测试用例套件

## 4.1 用例1 查询球员信息

### 表1-1测试用例套件对需求的覆盖情况

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 测试用例套件1 | 测试用例套件2 | 测试用例套件3 |
| Player.input | TUS1 | TUS2 | TUS3 |
| Player.selectAttribute | TUS1 |  | TUS3 |
| Player.refresh | TUS1 | TUS2 | TUS3 |
| Player.sort | TUS1 |  |  |
| Player.sort.ascending | TUS1 |  |  |
| Player.sort.descending | TUS1 |  |  |
| Player.search |  | TUS2 |  |
| Player.search.null |  | TUS2 |  |
| Player.search.invalid |  | TUS2 |  |
| Player.showDetail | TUS1 | TUS2 | TUS3 |
| Player.showInfoList | TUS1 |  |  |
| Player.showCardList | TUS1 |  |  |
| Player.filter |  |  | TUS3 |
| Player.skip.title | TUS1 |  |  |
| Player.skip.detail |  | TUS2 |  |

### 表1-2 TUS1的测试用例

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | 输入 | | | | | | | | | | | 预期输出 |
| 排序条件 | 升降序 | | 搜索内容 | | | 筛选 | | 显示模式 | 跳转 | |
| TUS1-1 | 条件A | | 升序 | |  |  | | 卡片 | | |  | 按条件A升序排列球员信息，外观为卡片 |
| TUS1-2 | 条件A | | 降序 | |  |  | | 卡片 | | |  | 按条件A降序排列球员信息，外观为卡片 |
| TUS1-3 | 条件A | | 升序 | |  |  | | 列表 | | |  | 按条件A升序排列球员信息，外观为列表 |
| TUS1-4 | 条件B | | 升序 | |  |  | | 卡片 | | |  | 按条件B升序排列球员信息，外观为卡片 |
| TUS1-5 | 条件B | | 升序 | |  |  | | 卡片 | | | 标题栏跳转 | 跳转到对应界面 |

### 表1-3 TUS2的测试用例

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | 输入 | | | | | | 预期输出 |
| 排序条件 | 升降序 | 搜索内容 | 筛选 | 显示模式 | 跳转 |
| TUS2-1 |  |  | 合法 |  |  | 详细信息页 | 显示搜索结果，跳转到对应球队信息页 |
| TUS2-2 |  |  | 不合法 |  |  |  | 提示输入不合法 |
| TUS2-3 |  |  | 空白 |  |  |  | 提示输入为空 |

### 表1-4 TUS3的测试用例

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID | 输入 | | | | | | | 预期输出 |
| 排序条件 | 升降序 | 搜索内容 | 筛选 | | 显示模式 | 跳转 |
| TUS3-1 | 条件A |  |  | 是 | 卡片 | |  | 按条件A筛选出前50名球员，外观为卡片 |
| TUS3-2 | 条件B |  |  | 是 | 卡片 | |  | 按条件B筛选出前50名球员，外观为卡片 |

## 4.2 用例2 查询球队信息

### 表2-1测试用例套件对需求的覆盖情况

|  |  |  |
| --- | --- | --- |
| 编号 | 测试用例套件1 | 测试用例套件2 |
| Team.input | TUS1 | TUS2 |
| Team.selectAttribute | TUS1 |  |
| Team.refresh | TUS1 | TUS2 |
| Team.sort | TUS1 |  |
| Team.sort.ascending | TUS1 |  |
| Team.sort.descending | TUS1 |  |
| Team.search |  | TUS2 |
| Team.search.null |  | TUS2 |
| Team.search.invalid |  | TUS2 |
| Team.showDetail | TUS1 | TUS2 |
| Team.showInfoList | TUS1 |  |
| Team.showCardList | TUS1 |  |
| Team.skip.title | TUS1 |  |
| Team.skip.detail |  | TUS2 |

### 表2-2 TUS1的测试用例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | 输入 | | | | | 预期输出 |
| 排序条件 | 升降序 | 搜索内容 | 显示模式 | 跳转 |
| TUS1-1 | 条件A | 升序 |  | 卡片 |  | 按条件A升序排列球队，外观为卡片 |
| TUS1-2 | 条件A | 降序 |  | 卡片 |  | 按条件A降序排列球队，外观为卡片 |
| TUS1-3 | 条件B | 升序 |  | 列表 |  | 按条件B升序排列球队，外观为列表 |
| TUS1-4 | 条件B | 升序 |  | 列表 | 标题栏跳转 | 按条件B升序排列球队，外观为列表，标题栏跳转到指定页面 |

### 表2-3 TUS2的测试用例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID | 输入 | | | | | 预期输出 |
| 排序条件 | 升降序 | 搜索内容 | 显示模式 | 跳转 |
| TUS2-1 |  |  | 合法 |  | 详细信息页 | 显示搜索结果，跳转到对应球员信息页 |
| TUS2-2 |  |  | 不合法 |  |  | 提示搜索内容不合法 |
| TUS2-3 |  |  | 空白 |  |  | 提示搜索内容为空 |