**NBA软件测试报告**

目录

[1.1 编写目的 1](#_Toc105057964)

[1.2 背景 1](#_Toc105057965)

[1.3 测试范围 1](#_Toc105057966)

[1.4 测试人员 1](#_Toc105057967)

[1.5 测试阶段 1](#_Toc105057968)

[2. 测试摘要 1](#_Toc105057969)

[2.1 测试类型 2](#_Toc105057970)

[2.1.1 数据和数据库完整性测试 2](#_Toc105057971)

[2.1.2 功能测试 2](#_Toc105057972)

[2.2 测试用例 3](#_Toc105057973)

[3. 测试结论与建议 4](#_Toc105057974)

1引言

## 编写目的

通过编写软件测试报告主要达成以下几个目的：

通过对测试结果的分析，得到对软件质量的评价。

分析测试的过程，用到的资源，信息，为以后制定测试计划提供参考

评估测试执行和测试计划是否符合

分析软件存在的缺陷，为修复和预防bug提供建议

## 背景

NBA软件通过爬虫抓取平台网站数据，为NBA爱好者提供球员，比赛等信息。主要功能包括三部分：球员信息展示功能，球队信息展示功能，比赛信息展示。

## 测试范围

单元测试：对各个模块的源代码的进行测试，保证各模块基本功能都能够正确地实现。

集成测试：将模块进行组合测试，保证所有功能都能够正确地实现。

系统测试：对软件进行功能测试，对重点的模块进行性能测试，并结合可能的用户测试。

## 测试人员

|  |  |  |
| --- | --- | --- |
| 角色 | 人员 | 职责 |
| 测试经理 | 陆鑫良 | 进行管理监督  提供测试报告 |
| 测试员 | 蔡书林、王永锦 | 生成测试计划  执行测试工作 |

## 测试阶段

|  |  |  |  |
| --- | --- | --- | --- |
| 测试里程碑任务 | 负责人员 | 开始时间 | 结束时间 |
| 制定测试计划 | 陆鑫良、蔡书林、王永锦 | 4月5日 | 4月10日 |
| 测试球员功能模块 | 王永锦、蔡书林 | 4月21日 | 4月26日 |
| 测试球队功能模块 | 陆鑫良、王永锦 | 5月5日 | 5月10日 |
| 测试比赛功能模块 | 蔡书林、陆鑫良 | 5月19日 | 5月24日 |
| 项目评估测试 | 蔡书林、王永锦、陆鑫良 | 5月25日 | 5月31日 |

# 测试摘要

NBA软件测试从2022年4月5日开始到2022年5月31日结束，测试功能点30个，测试共发现 12个bug。目前未处理的bug有２个，主要是关于数据的爬取，一些网站有反爬的机制。

## 测试类型

### 数据和数据库完整性测试

|  |  |
| --- | --- |
| 测试目标： | 确保数据库访问方法和进程正常运行，数据不会遭到损坏。 |
| 方法： | •调用各个数据库访问方法和进程，并在其中填充有效的和无效的数据或对数据的请求。  • 检查数据库，确保数据已按预期的方式填充，并且所有数据库事件都按正常方式出现；或者检查所返回的数据，确保为正当的理由检索到了正确的数据] |
| 完成标准： | 所有的数据库访问方法和进程都按照设计的方式运行，数据没有遭到损坏。 |
| 需考虑的特殊事项： | •测试可能需要 DBMS 开发环境或驱动程序以便在数据库中直接输入或修改数据。  •进程应该以手工方式调用。  •应使用小型或最小的数据库（其中的记录数很有限）来 使所有无法接受的事件具有更大的可见性。] |

### 功能测试

|  |  |
| --- | --- |
| 测试目标： | 确保测试对象的功能正常，其中包括导航、数据输入、处理和检索等 |
| 方法： | 利用有效的和无效的数据来执行各个用例、用例流或功能，以核实以下内容：  •在使用有效数据时得到预期的结果。  •在使用无效数据时显示相应的错误消息或警告消息。  •各业务规则都得到了正确的应用。] |
| 完成标准： | •所计划的测试已全部执行。  •所发现的缺陷已全部解决。] |
| 需考虑的特殊事项： | 确定或说明那些将对功能测试的实施和执行造成影响的事项或因素（内部的或外部的 |

## 测试情况

|  |  |  |  |
| --- | --- | --- | --- |
| 测试类 | 测试方法 | 测试用例描述 | 测试结果 |
| MatchDaoTest | testGetMatchInfoByGameid() | 通过比赛id获取比赛信息 | 成功 |
| MatchDaoTest | testGetSectionScoreByGameid() | 通过比赛id获取分数 | 成功 |
| MatchDaoTest | testGetMatchInfoByFilter() | 通过filter获取比赛信息 | 成功 |
| MatchDaoTest | testGetRegularMatchInfoBySeason() | 通过赛季获取常规赛信息 | 成功 |
| MatchDaoTest | testGetPlayOffMatchInfoBySeason() | 通过赛季获取已经完赛的赛季信息 | 成功 |
| MatchDaoTest | testGetMatchInfoByDate() | 通过日期获取比赛信息 | 成功 |
| MatchDaoTest | testGetMatchPlayerAdvancedByGameIdTeam() | 通过球队id获取比赛信息 | 成功 |
| MatchDaoTest | testGetMatchPlayerBasicByPlayer() | 通过球员获取比赛信息 | 失败 |
| PlayerDaoTest | testGetNameList() | 获取球员姓名列表 | 成功 |
| PlayerDaoTest | testGetPlayerInfoByNameInitial() | 通过姓名获取球员信息 | 成功 |
| PlayerDaoTest | testGetAllPlayerInfo() | 获得所有球员信息 | 成功 |
| PlayerDaoTest | testGetPlayerPerGameByFilter() | 通过filter获取每场比赛的球员信息 | 成功 |
| PlayerDaoTest | testGetPlayerTotalBySeason() | 获得指定赛季的所有球员信息 | 成功 |
| PlayerDaoTest | testGetPlayerPerGameBySeason() | 获得每个赛季中指定比赛的球员信息 | 成功 |
| PlayerDaoTest | testGetSeasonHotPlayer() | 获取指定赛季的热门球员的信息 | 成功 |
| PlayerDaoTest | testGetTeamPlayerBySeason() | 获取指定赛季的球队的球员按信息 | 成功 |
| PlayerDaoTest | testGetPlayerSalaryByName() | 通过球员姓名获取该球员的薪水 | 失败 |
| PlayerDaoTest | testGetTeamByPlayerName() | 通过球员姓名获取球队信息 | 成功 |
| TeamDaoTest | testGetAllTeamInfo() | 获取所有球队的信息 | 成功 |
| TeamDaoTest | testGetTeamInfoByAbbr() | 通过球队缩写获取球队信息 | 失败 |
| TeamDaoTest | testGetTeamTotalBySeason() | 获取指定赛季的球队信息 | 成功 |
| TeamDaoTest | testGetTeamPerGameBySeason() | 获取指定赛季的球队的每场比赛的信息 | 成功 |
| TeamDaoTest | testGetTeamOppTotalBySeason() | 指定赛季和球队获取该球队的对手 | 成功 |
| TeamDaoTest | testGetTeamOppPerGameBySeason() | 获取指定球队在该赛季的所有对手 | 成功 |
| TeamDaoTest | testGetTeamTotalByFilter() | 通过filter获取球队信息 | 成功 |
| TeamDaoTest | testGetSeasonHotTeam() | 获取赛季热门球队的信息 | 成功 |
| LiveServiceTest | testGetAllLiveList() | 获取直播列表 | 失败 |
| LiveServiceTest | testCheckMatchStart() | 获取开始的比赛 | 成功 |
| LiveServiceTest | testGetMsg() | 获取直播信息 | 成功 |
| LiveServiceTest | testGetHistoryList() | 获取历史信息 | 成功 |

# 测试结论与建议

NBA软件三个模块的功能基本实现，通过爬取网站球员、球队以及比赛数据，进行数据清洗、整合、分析，再利用可视化工具进行展示，从不同维度直观的表现了各个赛季的比赛水准。但数据量偏小，使得数据的展现力略显不足，数据的处理流程不够规范，导致许多脏数据未能清洗干净，使得结果不够客观。