# MovieReview

# 单元测试报告文档

# v1.0

南京大学软件学院

Inf1Nity 团队

帖佳韵 宋奎熹 薛恺丰 杨思佳

2017.04.12

# 变更记录

| 修改人员 | 日期 | 变更原因 | 版本号 |
|---|---|---|---|
| 杨思佳 | 2017.4.12 | 最初框架 | V1.0 |

# 目录

# 1. 撰写单元测试报告目的

1.1 整理和汇总单元测试结果，形成正式的文档

1.2 为团队内部对单元的评审提供依据

1.3 将该文档纳入软件产品配置管理库

# 2. 单元测试的范围和工具

单元测试使用工具 JUnit 完成了对 GitMining 中主要类的方法的测试.

# 3. 单元测试报告描述

## 3.1. Unit Test Case L1

被测单元: MovieBLService

设计人员: 杨思佳

设计日期: 2017–04–07

测试人员: 杨思佳

测试日期: 2017–04–12

测试用例表

| 测试用例 | 测试用例描述 | 输入数据 | 预期输出 | 实际输出 | Pass/Fail | 备注 |
|---|---|---|---|---|---|---|
| Movie BLService–01 | testFind MoviesB yKeywor dInPage | keyword = ""Snow White" | pageVO.list.size() = 2, pageVO.totalPage = 1 | pageVO.list.size() = 2, pageVO.totalPage = 1 | Pass | |
| Movie BLService–02 | testFind MoviesB yTagInPa ge | tag = Action, MovieSortType. DATE_ASC, page = 2 | pageVO.list.size() = 10, pageVO.totalPage = 10 | pageVO.list.size() = 10, pageVO.totalPage = 10 | Pass | |
| Movie BLService–03 | testFind MovieSt atisticsV OByMovi eId | id = "B005K23S20 | amountOfReviewFromAmazon = 516, amountOfReviewFromImdb = 38, averageScore = 9.018, variance = 5.036, firstReviewDate = "2010–07–27" lastReviewDate = "2016–12–21" | amountOfReviewFromAmazon = 516, amountOfReviewFromImdb = 38, averageScore = 9.018, variance = 5.036, firstReviewDate = "2010–07–27" lastReviewDate = "2016–12–21" | Pass | |
| Movie BLService–04 | testFind Reviews ByMovieI dInPage | id = "B00005JO1X", ReviewSortType .DATE_ASC, page = 2 | pageVO.list.size() = 10, pageVO.totalPage = 61 | pageVO.list.size() = 10, pageVO.totalPage = 61 | Pass | |

| 测试用例 | 测试用例描述 | 输入数据 | 预期输出 | 实际输出 | Pass/Fail | 备注 |
|---|---|---|---|---|---|---|
| MovieBLService–05 | testFindMovieGenre | | assertEquals(movieGenreVO.amounts.size(), movieGenreVO.tags.size()) | TRUE | Pass | |
| MovieBLService–06 | testFindRelationBetweenScoreAndReviewAmount | tag = Action | assertEquals(scoreAndReviewAmountVO.scores.size(), scoreAndReviewAmountVO.reviewAmounts.size()) | TRUE | Pass | |
| MovieBLService–07 | testfindScoreDateByYear(byMonth, ByDay同理) | id = "B005K23S20", startYear = "2011", endYear = "2012 | assertEquals(scoreDateVOActual.dates.size(), scoreDateVOActual.scores.size()) | TRUE | Pass | |
| MovieBLService–08 | testFindScoreDistributionByMovieIdFromIMDB(FromAmazon同理) | id = "B005K23S20" | new ScoreDistributionVO(38, {3, 5, 2, 1, 2, 2, 2, 6, 3, 9}) | new ScoreDistributionVO(38, {3, 5, 2, 1, 2, 2, 2, 6, 3, 9}) | Pass | |

| 测试用例 | 测试用例描述 | 输入数据 | 预期输出 | 实际输出 | Pass/Fail | 备注 |
|---|---|---|---|---|---|---|
| MovieBLService–09 | testBoxPlot | id = "B00000F168" | maxScore = 5, minScore = 0, quartiles = {5.0,5.0,5.0,5.0,5.0} | maxScore = 5, minScore = 0, quartiles = {5.0,5.0,5.0,5.0,5.0} | Pass | |
| MovieBLService–10 | testFindYearCountByMovieIdFromAmazon(MonthCount, DayCount 同理) | id = "B00005JO1X" startYear = "2005", endYear = "2006" | reviewCountVOs.length=6, keys={2005,2006},reviewCountVO[0]={96, 1,7,14,19,55}, reviewCountVO[1]={299, 20,15,29,38,127} | reviewCountVOs.length=6, keys={2005,2006},reviewCountVO[0]={96, 1,7,14,19,55}, reviewCountVO[1]={299, 20,15,29,38,127} | Pass | |
| MovieBLService–11 | testFindYearCountByMovieIdFromImdb | id = "B005K23S20", "startYear = "2011", endYear = "2012" | reviewCountVOs.length=11, keys={2011,2012},reviewCountVO[0]={12, 0, 1, 1, 0, 0, 1, 1, 1, 2, 5}, reviewCountVO[1]={5, 0, 1, 0, 1, 2, 0, 0, 1, 0, 0} | reviewCountVOs.length=11, keys={2011,2012},reviewCountVO[0]={12, 0, 1, 1, 0, 0, 1, 1, 1, 2, 5}, reviewCountVO[1]={5, 0, 1, 0, 1, 2, 0, 0, 1, 0, 0} | Pass | |

## 3.2. Unit Test Case L2

被测单元: MovieDataService

设计人员: 薛恺丰

设计日期: 2016–04–07

测试人员: 薛恺丰

测试日期: 2017–04–12

测试用例表

| 测试用例 | 测试用例描述 | 输入数据 | 预期输出 | 实际输出 | Pass/ Fail | 备注 |
|---|---|---|---|---|---|---|
| DataService–01 | testFindReviewsByMovieId | id = "B002LII6KA" | List<ReviewPO> 包含607条 reviewPO | List<ReviewPO> 包含607条 reviewPO | Pass | |
| DataService–02 | testFindReviewsByUserId | id = "A11YJS79DZD7D9" | List<ReviewPO> 包含13条 reviewPO | List<ReviewPO> 包含13条 reviewPO | Pass | |
| DataService–03 | testFindMovieByMovieId | id = "B000I5XDV1" | new MoviePO("B000ZLFALS"," Across the Universe [Blu–ray]") | new MoviePO("B000ZLFALS"," Across the Universe [Blu–ray]") | Pass | |

| | | | | | | |
|---|---|---|---|---|---|---|
| DataService–04 | testFindReviewsByMovieId2 | id = "B00005JO1X" | List<ReviewPO><br>包含610条 reviewPO | List<ReviewPO><br>包含610条 reviewPO | Pass | |
| DataService–05 | testFindReviewsByMovieId3 | id = "B00" | List<ReviewPO><br>包含0条 reviewPO | List<ReviewPO><br>包含0条 reviewPO | Pass | |
| DataService–06 | testFindReviewsByMovieId4 | id = "B000ZLFALS" | List<ReviewPO><br>包含1111条 reviewPO | List<ReviewPO><br>包含1111条 reviewPO | Pass | |
| DataService–07 | testFindReviewsByUserId2 | id = "A11YJS79DZD7D9" | Page<ReviewPO><br>包含1条 reviewPO | Page<ReviewPO><br>包含1条 reviewPO | Pass | |
| DataService–08 | testFindMovieByMovieId2 | id = "B000ZLFALS11" | moviePO<br>其Id为–1<br>（不存在） | moviePO<br>其Id为–1<br>（不存在） | Pass | |
| DataService–09 | testFindWordByMovieId | id = "B000ZLFALS" | wordPO<br>包含10个word | wordPO<br>包含10个word | Pass | |
| DataService–10 | testFindWordByMovieId2 | id = "2" | null | null | Pass | |
| DataService–11 | testFindWordByUserId | id = "A11YJS79DZD7D9" | wordPO<br>包含10个word | wordPO<br>包含10个word | Pass | |

| | | | | | | |
|---|---|---|---|---|---|---|
| DataService–12 | testFindWordByUserId2 | id = "2" | null | null | Pass | |
| DataService–13 | testFindReviewsByMovieIdFromAmazon | id = "B00005JO1X" ReviewSortType.DATE_ASC page = 0 | Page<ReviewPO> 包含10条 reviewPO | Page<ReviewPO> 包含10条 reviewPO | Pass | |
| DataService–14 | testFindReviewsByMovieIdFromAmazon2 | id = "B00005JO1X" ReviewSortType.DATE_ASC page = 19 | Page<ReviewPO> 包含10条 reviewPO | Page<ReviewPO> 包含10条 reviewPO | Pass | |
| DataService–15 | testFindMoviesByTag1 | EnumSet.of(MovieGenre.All), MovieSortType.DATE_ASC, page = 1 | Page<MoviePO> 包含10条 MoviePO | Page<MoviePO> 包含10条 MoviePO | Pass | |
| DataService–16 | testFindMoviesByTag2 | EnumSet.of(MovieGenre.Short), MovieSortType.DATE_ASC, page = 1 | Page<MoviePO> 包含10条 MoviePO | Page<MoviePO> 包含10条 MoviePO | Pass | |

| | | | | | | |
|---|---|---|---|---|---|---|
| DataService–17 | testFindMoviesByName | name = "a"<br>page = 1 | Page<MoviePO><br>包含10条<br>MoviePO | Page<MoviePO><br>包含10条<br>MoviePO | Pass | |
| DataService–18 | testFindMovieGenre | | MovieGenrePO.amounts.size=22 | MovieGenrePO.amounts.size=22 | Pass | |
| DataService–19 | testCheckNetwork | | TRUE | TRUE | Pass | |
| DataService–20 | testFindRelationBetweenScoreAndReviewAmount | EnumSet.of(MovieGenre.Action) | scoreAndReviewAmountPO<br>包含42条记录 | scoreAndReviewAmountPO<br>包含42条记录 | Pass | |

# 3.3 Unit Test Case L3

被测单元: UserBLService

设计人员: 杨思佳

设计日期: 2017–04–07

测试人员: 杨思佳

测试日期: 2017–04–12

测试用例表

| 测试用例 | 测试用例描述 | 输入数据 | 预期输出 | 实际输出 | Pass/Fail | 备注 |
|---|---|---|---|---|---|---|
| UserBLService–01 | testGetReviewWords | id = "ABSX5TGEGRH76" | new ReviewWordsVO({"0–20","20–40","40–60","60–80","80–100","100–120","120–140","140–160","160–180","180–200","200+"},{0,0,7,0,0,0,0,0,0,0,0}) | new ReviewWordsVO({"0–20","20–40","40–60","60–80","80–100","100–120","120–140","140–160","160–180","180–200","200+"},{0,0,7,0,0,0,0,0,0,0,0}) | Pass | |
| UserBLService–02 | testFindYearCountByMovieId | id = "ANGH6401L9Y0T" | new ReviewCountVO[]<br><br>包含6个ReviewCountVO,分布表示全部、评分为1、2、3、4、5的评论数量在时间上的分布 | new ReviewCountVO[]<br><br>包含6个ReviewCountVO,分布表示全部、评分为1、2、3、4、5的评论数量在时间上的分布 | Pass | |
| UserBLService–03 | testFindMonthCountByMovieId | id = "ANGH6401L9Y0T" | new ReviewCountVO[]<br><br>包含6个ReviewCountVO,分布表示全部、评分为1、2、3、4、5的评论数量在时间上的分布 | new ReviewCountVO[]<br><br>包含6个ReviewCountVO,分布表示全部、评分为1、2、3、4、5的评论数量在时间上的分布 | Pass | |

| UserBLService–04 | testFindDayCountByMovieId | id = "ANGH6401L9Y0T" | new ReviewCountVO[]<br><br>包含6个ReviewCountVO,分布表示全部、评分为1、2、3、4、5的评论数量在时间上的分布 | new ReviewCountVO[]<br><br>包含6个ReviewCountVO,分布表示全部、评分为1、2、3、4、5的评论数量在时间上的分布 | Pass | |
|---|---|---|---|---|---|---|
| UserBLService–05 | testGetUserVO | id = "A2582KMXLI2P06 " | new UserVO("A2582KMXLI2P06", "B. E Jackson 6", 6, "2011–05–25", "2011–06–01") | new UserVO("A2582KMXLI2P06", "B. E Jackson 6", 6, "2011–05–25", "2011–06–01") | Pass | |
| UserBLService–06 | testFindWordByMovieId | id = "A11YJS79DZD7D9" | wordVO.getTopWords().size() = 10 | wordVO.getTopWords().size() = 10 | Pass | |
| UserBLService–07 | testFindWordByMovieId2 | id = "2" | wordVO = null | wordVO = null | Pass | |
| UserBLService–08 | testFindReviewsByUserIdInPage | id = "A2HBJS5V7RS1SM",ReviewSortType.DATE_ASC, page = 0 | pageVO.list.size() = 1, pageVO.totalPage = 1 | pageVO.list.size() = 1, pageVO.totalPage = 1 | Pass | |

# 4. 结论

单元测试达到了 100%的语句覆盖率，100%的分支覆盖率。

单元测试达到了 100%的语句覆盖率，100%的分支覆盖率。