**听app**

**测**

**试**

**报**

**告**

**课 程： 软件工程**

**题 目： 幻听**

**基于Flutter框架的集各大音乐平台于一体的音乐APP**

**成 员：邢海粟 31801347 组长**

**黄德煜 31801163 组员**

**章拾瑜 31801335 组员**

|  |  |  |  |
| --- | --- | --- | --- |
| **文件名** | **版本号** | **更改日期** | **更改人** |
| **SE2020-G06-测试报告** | V0.0.1 | 2020.12.24 | 黄德煜 |
| **SE2020-G06-测试报告** | V0.0.2 | 2020.12.28 | 黄德煜 |
| **SE2020-G06-测试报告** | V0.0.3 | 2020.12.30 | 邢海粟 |
| **SE2020-G06-测试报告** | V0.0.4 | 2020.12.31 | 章拾瑜 |
| **SE2020-G06-测试报告** | V0.1.0 | 2020.1.10 | 黄德煜 |

目录

[1引言 3](#_Toc62088954)

[1.1标识 3](#_Toc62088955)

[1.2系统概述 4](#_Toc62088956)

[1.3文档概述 5](#_Toc62088957)

[2引用文件 5](#_Toc62088958)

[3测试结果概述 5](#_Toc62088959)

[3.1对被测试软件的总体评估 5](#_Toc62088960)

[3.2测试环境的影响 6](#_Toc62088961)

[4详细的测试结果 6](#_Toc62088962)

[4.1 单元测试 6](#_Toc62088963)

[4.1.1单元测试+Widget测试结果 6](#_Toc62088964)

[4.1.2前端单元测试、Widget测试的问题修正记录 12](#_Toc62088965)

[4.2 SpringBoot单元测试 12](#_Toc62088966)

[4.2.1 SpringBoot单元测试结果 12](#_Toc62088967)

[4.3集成测试 16](#_Toc62088968)

[4.3.1 黑盒集成测试 16](#_Toc62088969)

[4.3.2 白盒集成测试用例 24](#_Toc62088970)

[4.3.3 后端集成测试 30](#_Toc62088971)

[4.3.4集成测试问题修正记录 34](#_Toc62088972)

[5系统测试 34](#_Toc62088973)

[5.1 CPU占用率 35](#_Toc62088974)

[35](#_Toc62088975)

[5.2 Graphics画面每秒传输帧数（FPS） 35](#_Toc62088976)

[5.3 Merrory内存占用（单位KiB） 35](#_Toc62088977)

[5.4 NetWork网络请求（单位byte） 35](#_Toc62088978)

[5.5 性能测试 36](#_Toc62088979)

[6测试活动总结 36](#_Toc62088980)

[7注解 36](#_Toc62088981)

[附录 36](#_Toc62088982)

# 1引言

## 1.1标识

本可行性分析报告仅适用于SE2020-G06小组开发的Flutter实现一个集各大音乐平台API于一体的音乐APP。

缩略词语：

TBD

版本号格式：

代码版本格式：主版本号.次版本号.修订号，版本号递增规则如下：

1. 主版本号：当你做了不兼容的 API 修改，

2. 次版本号：当你做了向下兼容的功能性新增，

3. 修订号：当你做了向下兼容的问题修正。

报告（PPT）版本格式：正式版本号.评审版本号.自行迭代号，版本号递增规则如下：

1. 正式版本号：正式评审过后的正式版本，一般为1.0.0，提交日期为评审当周的周日，

2. 评审版本号：经过预评审或先行组评审后，根据其评审内容修改的版本。3. 自行迭代号：根据上课内容或小组会议后，对上一版本内容的修正

## 1.2系统概述

系统：

该文档适用的系统为Android 4.4+/IOS8.0+的手机内核

用途：

软件用途为用Flutter框架实现一个集各大音乐平台API于一体的音乐播放器APP。

项目开发历史：

2020年9月29日至2020年10月4日：小组正式成立并完成课程预备作业。

2020年10月5日至2020年10月15日：课程正式选题阶段。

2020年10月16日至2020年10月22日：项目计划初步制定阶段。

2020年10月23日至2020年10月29日：项目计划修订阶段。

2020年10月30日至2020年11月5日：可行性分析阶段

2020年11月6日至2020年11月19日：软件需求分析阶段

2020年11月20日至2020年12月3日：软件设计阶段

2020年12月4日至2020年12月31日：翻转课堂+代码实现+测试报告

需方：

软件工程导论指导老师杨枨教授；

用户：

开发前期测试人员、软件工程导论课程评审人员、所有对使用音乐播放APP有需求的用户；

开发方:

SE2020-G06小组，组长邢海粟，组员章拾瑜、黄德煜；

支持机构：

SE2020课程组

其他相关文档：

课程作业选题报告

项目开发计划报告

可行性分析报告

需求说明报告

总体设计报告

系统说明文档

测试计划

数据库设计结果

详细设计报告

系统测试报告

项目总结报告

## 1.3文档概述

该软件设计说明对基于Flutter实现一个集各大音乐API为一体的音乐播放APP《幻听》软件做了针对系统的测试报告

本说明文档具备一定的保密性与私密性要求，出于原则与法律规定，不公开，不盗用，不贩卖用户的个人信息与数据。

# 2引用文件

1. GB/T-8567-2006标准的软件工程项目开发计划书模板
2. 软件测试报告实例

https://max.book118.com/html/2018/0616/172956833.shtm

1. 张海藩,牟永敏.软件工程导论[M].清华大学出版社:北京,2013:1-31.
2. 暴力突破，flutter自动化测试

https://blog.csdn.net/u010289802/article/details/109491031?utm\_medium=distribute.pc\_relevant.none-task-blog-baidujs\_baidulandingword-6&spm=1001.2101.3001.4242

# 3测试结果概述

## 3.1对被测试软件的总体评估

本条应：

a.根据本报告中所展示的测试结果，提供对该软件的总体评估；

b.标识在测试中检测到的任何遗留的缺陷、限制或约束。可用问题/变更报告提供缺陷信息；

c.对每一遗留缺陷、限制或约束，应描述：

1)对软件和系统性能的影响，包括未得到满足的需求的标识；

2)为了更正它，将对软件和系统设计产生的影响；

3)推荐的更正方案/方法。

TBD

## 3.2测试环境的影响

测试环境大多位于电脑上，通过模拟器控制和电脑上的程序来保证测试的可用性，对于实际的实机场景其实存在一定的区别，需要针对实机场景的黑盒测试进行多层次多方面的比较和检查才可以尽可能保证找出存在的错误，对人力成本的影响颇大。

# 4详细的测试结果

## 4.1 单元测试

### 4.1.1单元测试+Widget测试结果

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **测试项** | **标识符** | **测试环境** | **测试内容** | **预期结果** | **测试结果** | **问题分析** | **设计人员** | **测试人员** |
| 页面底部播放条PlayWidget | 1.1 | Flutter\_test | 未播放歌曲 | 所有页面底部播放条PlayWidget，默认图片logo.png，播放按钮Image，播放列表Image | 通过 | 无 | 章拾瑜 | 邢海粟 |
| 页面底部播放条PlayWidget | 1.2 | Flutter\_test | 播放歌曲时 | 所有页面底部播放条PlayWidget，歌曲图片curSong.picUrl，暂停按钮Image，播放列表Image | 通过 | 无 | 章拾瑜 | 邢海粟 |
| 页面底部播放条PlayWidget | 1.3 | Flutter\_test | 播放歌曲时，触发暂停 | 暂停按钮Image变为播放按钮Image，执行歌曲pause()事件 | 通过 | 无 | 章拾瑜 | 邢海粟 |
| 页面底部播放条PlayWidget | 1.4 | Flutter\_test | 暂停播放歌曲时 | 所有页面底部播放条PlayWidget，歌曲图片curSong.picUrl，播放按钮Image，播放列表Image | 通过 | 无 | 章拾瑜 | 邢海粟 |
| 页面底部播放条PlayWidget | 1.5 | Flutter\_test | 暂停播放歌曲时，重新播放 | 所有页面底部播放条PlayWidget，歌曲图片curSong.picUrl，播放按钮Image，播放列表Image | 通过 | 无 | 章拾瑜 | 邢海粟 |
| 网易云通用网络请求 | 2.1 | Flutter\_test | Future<Response> \_get | 状态码不为2xx，但是仍然返回了数据，Dio 抛出 DioError 的，需要我们自己捕获来处理。如果返回了正常的数据，那我们还是返回回去，如果不是正常的数据，则直接抛出 Future.error(0)。 | 通过 | 无 | 邢海粟 | 黄德煜 |
| QQ音乐通用网络请求 | 2.2 | Flutter\_test | Future<Response> \_get2 | 状态码不为2xx，但是仍然返回了数据，Dio 抛出 DioError 的，需要我们自己捕获来处理。如果返回了正常的数据，那我们还是返回回去，如果不是正常的数据，则直接抛出 Future.error(0)。 | 通过 | 无 | 邢海粟 | 黄德煜 |
| 音乐专辑数据获取 | 3.1 | Flutter\_test | List<Albums> \_albums | 获取专辑中对应的编号，名称，简介，资源，是否付费，时长，歌手，作曲，出版公司，出版时间等信息 | 通过 | 无 | 邢海粟 | 黄德煜 |
| 网易云音乐推荐数据获取 | 3.2 | Flutter\_test | DailySongsData({num code, List<Recommend> recommend}) | 获取网易云每日推荐的歌曲的信息如名称，简介，资源，付费情况，名次，歌手等 | 通过 | 无 | 邢海粟 | 黄德煜 |
| 获取QQ音乐热词搜索数据 | 3.3 | Flutter\_test | HotSearchDataQQ | 获取QQ音乐搜索的热词对应的信息如位置，图标，记录，内容等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 获取网易云热词搜索数据 | 3.4 | Flutter\_test | class HotSearchData | 获取网易云搜索的热词对应的信息如位置，图标，记录，内容等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 获取网易云歌单数据 | 3.5 | Flutter\_test | List<Playlist> playlist | 获取网易云歌单信息，如名称，歌曲数量，歌曲资源，评论等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 获取网易云排行榜数据 | 3.6 | Flutter\_test | class TopList | 获取网易云排行榜信息如编号，下载数，收藏数，创建时间，图片，资源等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 获取网易云歌曲评论数据 | 3.7 | Flutter\_test | class SongCommentData | 获取网易云音乐评论信息如点赞数，置顶，热评，回复，内容等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 获取网易云歌曲数据 | 3.8 | Flutter\_test | Song(this.id, {this.name, this.artists, this.picUrl}) | 获取网易云音乐数据如编号，名称，资源，歌手等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 圆形图像RoundWidget | 4.1 | Flutter\_test | 是否存在图像widget | 输出widget不报错 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 专辑列表AlbumWidget | 4.2 | Flutter\_test | 是否存在专辑widget以及专辑数据列表 | 输出widget不报错和专辑图片icon\_album.png以及专辑名字AlbumName和专辑信息AlbumInfo | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 轮播广告BannerWidget | 4.3 | Flutter\_test | 是否存在广告widget以及广告信息 | 输出widget不报错和广告数量Indicator，广告容器CustomBanner还有广告效果PageView | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 底部导航栏FootTabWidget | 4.4 | Flutter\_test | 是否存在底部导航栏widget | 输出widget不报错 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 加载界面LoadWidget | 4.5 | Flutter\_test | 是否存在加载界面widget | 输出widget不报错 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 搜索历史记录HistoryWidget | 4.6 | Flutter\_test | 是否存在历史记录界面widget以及搜索界面元素 | 输出widget不报错，搜索信息HEmptyView，清除文本Text，记录数量Count | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 歌曲列表界面Widget | 4.7 | Flutter\_test | 是否存在歌曲列表界面widget以及歌曲列表信息 | 输出widget不报错，歌曲名称MuxicName，作者Artist，Mv等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 网络报错界面Widget | 4.8 | Flutter\_test | 是否存在网络报错界面widget | 输出widget不报错 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 歌单界面Widget | 4.9 | Flutter\_test | 是否存在歌单界面widget | 输出widget不报错，图片Image，文字Info，点击事件Click | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 搜索界面Widget | 4.1 | Flutter\_test | 是否存在搜索界面widget | 输出widget不报错，存在搜索按钮Search，历史记录History，热点事件hot等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 搜索歌曲结果Widget | 4.11 | Flutter\_test | 是否存在搜索歌曲界面widget | 输出widget不报错，歌曲名称MuxicName，信息Info，资源url | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 搜索歌手结果Widget | 4.12 | Flutter\_test | 是否存在搜索歌手界面widget | 输出widget不报错，歌手名称Username和描述Description | 通过 | 无 | 黄德煜 | 章拾瑜 |
| 搜索mv结果Widget | 4.13 | Flutter\_test | 是否存在搜索mv界面widget | 输出widget不报错，资源url，名称Name，信息Info，创作人Creator，类型Type等 | 通过 | 无 | 黄德煜 | 章拾瑜 |
| UserService 登录 | 5.1 | Junit | 用户名 username ：admin 用户密码 password：admin | 1. 函数返回用户类的用户名为admin 2. 函数返回类的密码为admin | 通过 | 无 | 邢海粟 | 黄德煜 |
| UserService 登录 | 5.2 | Junit | 用户名 username ：admin 用户密码 password：123 | 1.函数返回用户类为Null | 通过 | 无 | 邢海粟 | 黄德煜 |
| UserService 注册 | 5.3 | Junit | 用户名 username ：user002 用户密码 password：123456 | 1.返回插入成功行数1 | 通过 | 无 | 邢海粟 | 黄德煜 |
| UserService 确认用户注册 | 5.4 | Junit | 用户名 username ：user001 | 1.返回查询到符合条件的行数1 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongListService 增加歌单 | 6.1 | Junit | 用户Id userId ：1 歌单名：songListName:测试歌单2 歌单描述：这是一个测试歌单 | 1.数据库能够成功添加这一条记录 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongListService 检测歌单 | 6.2 | Junit | 用户Id userId ：1 歌单名：songListName:测试歌单1 | 1.从数据库返回歌单类的用户Id为1 2.从数据库返回歌单类的歌单名为测试歌单1 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongListService 检测歌单 | 6.3 | Junit | 用户Id userId ：2 歌单名：songListName:测试歌单1 | 1.从数据库返回歌单类为Null | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongListService 删除指定歌单 | 6.4 | Junit | 用户Id userId ：1 歌单名：songListName:测试歌单2 | 1.数据库能够删除该条记录 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongListService修改歌单 | 6.5 | Junit | 用户Id userId ：1 歌单名：songListName:测试歌单2 歌单描述：songListProfile：修改测试  歌单图片：songListAvatar：图片网络地址 | 1.数据库能够成功修改这一条记录 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongService 歌单增加歌曲 | 7.1 | Junit | 歌单Id songListId ：4 歌曲Id：songId:1234 所属平台：platform：网易云 | 1.数据库能够成功添加这一条记录 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongService 检测歌曲 | 7.2 | Junit | 歌单Id songListId ：4 歌曲Id：songId:1234 所属平台：platform：网易云 | 1.从数据库返回歌曲类的歌单Id为1 2.从数据库返回歌曲类的歌曲Id为1234 | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongService 检测歌曲 | 7.3 | Junit | 歌单Id songListId ：1 歌曲Id：songId:1234 所属平台：platform：网易云 | 1.从数据库返回歌曲类为Null | 通过 | 无 | 邢海粟 | 黄德煜 |
| SongService 删除指定歌曲 | 7.4 | Junit | 歌单Id songListId ：1 歌曲Id：songId:1234 | 1.数据库能够删除该条记录 | 通过 | 无 | 邢海粟 | 黄德煜 |

### 4.1.2前端单元测试、Widget测试的问题修正记录

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例标识符 | 错误或问题描述 | 错误或问题状态 | 测试人员 |
| 3.1 | 在获取专辑列表时，有时会存在按钮消失或付费歌曲不显示播放按钮的bug | 已解决 | 黄德煜 |
| 4.12 | 在搜索部分特殊歌手时，如果该歌手没有任何mv，作为综合搜索的结果会报错 | 已解决 | 章拾瑜 |
| 7.4 | 如果歌曲已经被删除过，会出现bug | 已解决 | 黄德煜 |

## 4.2 SpringBoot单元测试

### 4.2.1 SpringBoot单元测试结果

|  |  |  |
| --- | --- | --- |
| 测试模块：UserService 登录 | | |
| 标识符：UserServiceTest Login | | |
| 测试项：用户登录模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户名 username ：admin  用户密码 password：admin | 1. 函数返回用户类的用户名为admin 2. 函数返回类的密码为admin | 1.函数返回用户类的用户名为admin  2.函数返回类的密码为admin |

|  |  |  |
| --- | --- | --- |
| 测试模块：UserService 登录 | | |
| 标识符：UserServiceTest Login | | |
| 测试项：用户登录模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户名 username ：admin  用户密码 password：123 | 1.函数返回用户类为Null | 1. 函数返回用户类为Null |

|  |  |  |
| --- | --- | --- |
| 测试模块：UserService 注册 | | |
| 标识符：UserServiceTest Register | | |
| 测试项：用户注册模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户名 username ：user002  用户密码 password：123456 | 1.返回插入成功行数1 | 1.返回值为1 |

|  |  |  |
| --- | --- | --- |
| 测试模块：UserService 确认用户注册 | | |
| 标识符：UserServiceTest CheckReg | | |
| 测试项：用户确认已注册模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户名 username ：user001 | 1.返回查询到符合条件的行数1 | 1.返回值为1 |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongListService 增加歌单 | | |
| 标识符：SongListService addList | | |
| 测试项：添加歌单模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户Id userId ：1  歌单名：songListName:测试歌单2  歌单描述：这是一个测试歌单 | 1.数据库能够成功添加这一条记录 | 1.数据库成功添加这一记录 |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongListService 检测歌单 | | |
| 标识符：SongListService checkList | | |
| 测试项：检测歌单模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户Id userId ：1  歌单名：songListName:测试歌单1 | 1.从数据库返回歌单类的用户Id为1  2.从数据库返回歌单类的歌单名为测试歌单1 | 1.返回歌单类的用户Id为1  2.返回歌单类的歌单名为测试歌单1 |
| 测试模块：SongListService 检测歌单 | | |
| 标识符：SongListService checkList | | |
| 测试项：检测歌单模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户Id userId ：2  歌单名：songListName:测试歌单1 | 1.从数据库返回歌单类为Null | 1.从数据库返回歌单类为Null |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongListService 删除指定歌单 | | |
| 标识符：SongListService removeList | | |
| 测试项：删除歌单模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户Id userId ：1  歌单名：songListName:测试歌单2 | 1.数据库能够删除该条记录 | 1.数据库删除该条记录成功 |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongListService修改歌单 | | |
| 标识符：SongListService modifyList | | |
| 测试项：修改歌单模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 用户Id userId ：1  歌单名：songListName:测试歌单2  歌单描述：songListProfile：修改测试  歌单图片：songListAvatar：图片网络地址 | 1.数据库能够成功修改这一条记录 | 1.数据库成功修改这一记录 |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongService 歌单增加歌曲 | | |
| 标识符：SongService addSong | | |
| 测试项：添加歌曲模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 歌单Id songListId ：4  歌曲Id：songId:1234  所属平台：platform：网易云 | 1.数据库能够成功添加这一条记录 | 1.数据库成功添加这一记录 |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongService 检测歌曲 | | |
| 标识符：SongService checkSong | | |
| 测试项：检测歌曲模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 歌单Id songListId ：4  歌曲Id：songId:1234  所属平台：platform：网易云 | 1.从数据库返回歌曲类的歌单Id为1  2.从数据库返回歌曲类的歌曲Id为1234 | 1.从数据库返回歌曲类的歌单Id为1  2.从数据库返回歌曲类的歌曲Id为1234 |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongService 检测歌曲 | | |
| 标识符：SongService checkSong | | |
| 测试项：检测歌曲模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试输入数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 歌单Id songListId ：1  歌曲Id：songId:1234  所属平台：platform：网易云 | 1.从数据库返回歌曲类为Null | 1.从数据库返回歌曲类为Null |

|  |  |  |
| --- | --- | --- |
| 测试模块：SongService 删除指定歌曲 | | |
| 标识符：SongService removeSong | | |
| 测试项：删除歌单模块的后端与数据库操作过程 | | |
| 测试环境：Junit | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过Junit测试数据 | | |
| 输入数据 | 期望输出 | 实际结果 |
| 歌单Id songListId ：1  歌曲Id：songId:1234 | 1.数据库能够删除该条记录 | 1.数据库删除该条记录成功 |

## 4.3集成测试

### 4.3.1 黑盒集成测试

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 登录注册模块 | | |
| 标识符 | 1.0 | | |
| 测试项 | 登录注册功能 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 1. 打开程序，进入登录界面 2. 点击注册 3. 输入账号密码 4. 注册成功后返回绑定界面登录 5. 登录成功 | | |
| 输入数据 | 输入账号密码（注册）  不输入账号（注册）  不输入密码（注册）  输入正确的账号密码（登录）  不输入账号（登录）  不输入密码（登录）  输入错误的账号密码（登录） | | |
| 预期输出 | 注册并返回成功数值1  提示请输入账号或者密码  提示请输入账号或者密码  登录并获取对应的账号信息  提示请输入账号或者密码  提示请输入账号或者密码  提示登录失败，请检查账号密码 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 邢海粟 | 设计日期 | 2020.12.15 |
| 测试人员 | 黄德煜 | 测试日期 | 2020.12.21 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 外部api绑定模块 | | |
| 标识符 | 2.0 | | |
| 测试项 | 登录注册功能+外部账号绑定功能 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 打开程序，进入登录界面  登录成功  绑定外部api账号  输入对应的账号密码  绑定成功 | | |
| 输入数据 | 输入正确的账号密码（登录）  输入正确的账号密码（绑定api）  不输入账号（绑定api）  不输入密码（绑定api）  输入错误的账号密码（登录） | | |
| 预期输出 | 登陆成功  绑定api成功，获取对应数据  提示请输入账号或者密码  提示请输入账号或者密码  提示登录失败，请检查账号密码 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 邢海粟 | 设计日期 | 2020.12.15 |
| 测试人员 | 黄德煜 | 测试日期 | 2020.12.21 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 推荐功能模块 | | |
| 标识符 | 3.0 | | |
| 测试项 | 推荐歌单+每日排行榜 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 打开程序，绑定外部账号  进入发现界面  点击推荐歌单，获取信息  退回发现界面  点击排行榜，获取信息 | | |
| 输入数据 | 手势点击 | | |
| 预期输出 | 推荐歌单列表  推荐歌曲信息  排行榜列表  排行榜歌曲信息 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 推荐功能模块 | | |
| 标识符 | 3.1 | | |
| 测试项 | 推荐歌单+每日排行榜 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 打开程序，不绑定外部账号  进入发现界面  点击推荐歌单，获取信息  退回发现界面  点击排行榜，获取信息 | | |
| 输入数据 | 手势点击 | | |
| 预期输出 | 推荐歌单列表  推荐歌曲信息  排行榜列表  排行榜歌曲信息 | | |
| 实际输出 | 可以获得排行榜列表  无法加载出推荐歌曲列表并报错 | | |
| 测试结果 | 未通过 | | |
| 问题描述 | 报错：  \*\*\* DioError \*\*\*:  uri: http://118.24.63.15:1020/recommend/resource  DioError [DioErrorType.RESPONSE]: Http status error [301] | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 推荐功能模块 | | |
| 标识符 | 3.2 | | |
| 测试项 | 推荐歌单+每日排行榜 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 打开程序，绑定外部账号  进入发现界面  点击推荐歌单，获取信息  退回发现界面  点击排行榜，获取信息 | | |
| 输入数据 | 手势点击 | | |
| 预期输出 | 推荐歌单列表  推荐歌曲信息  排行榜列表  排行榜歌曲信息 | | |
| 实际输出 | TestFailed，正常显示前20首歌曲，超过20个歌曲出现报错 | | |
| 测试结果 | 未通过 | | |
| 问题描述 | 报错：RangeError (index): Invalid value: Not in inclusive range 0..19: 20 | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 歌单功能模块 | | |
| 标识符 | 4.0 | | |
| 测试项 | 外部歌单导入+本地歌曲导入+歌单管理 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 绑定好外部api  自动获取外部歌单  点击本地歌曲获取本地音乐资源  通过歌单界面对歌单进行管理 | | |
| 输入数据 | 正确的账号和密码  手势点击 | | |
| 预期输出 | 外部歌单信息  本地歌曲信息  增删改本地歌单 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 搜索功能模块 | | |
| 标识符 | 5.0 | | |
| 测试项 | 搜索历史记录+搜索热词+搜索信息 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 进入搜索界面  随机搜索  进入搜索结果页  返回  查看搜索历史记录  进行热词搜索  进入搜索结果页 | | |
| 输入数据 | 搜索信息  手势点击  热词数据 | | |
| 预期输出 | 搜索综合排序  搜索歌曲排序  搜索歌手排序  搜索视频排序  搜索历史记录 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 搜索功能模块 | | |
| 标识符 | 5.1 | | |
| 测试项 | 搜索历史记录+搜索热词+搜索“林俊杰” | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 进入搜索界面  随机搜索  进入搜索结果页  返回  查看搜索历史记录  进行热词搜索  进入搜索结果页 | | |
| 输入数据 | 搜索信息  手势点击  热词数据 | | |
| 预期输出 | 搜索综合排序  搜索歌曲排序  搜索歌手排序  搜索视频排序  搜索历史记录 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 搜索功能模块 | | |
| 标识符 | 5.2 | | |
| 测试项 | 搜索历史记录+搜索热词+搜索“周杰伦” | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑 | | |
| 操作步骤 | 进入搜索界面  随机搜索  进入搜索结果页  返回  查看搜索历史记录  进行热词搜索  进入搜索结果页 | | |
| 输入数据 | 搜索信息  手势点击  热词数据 | | |
| 预期输出 | 搜索综合排序  搜索歌曲排序  搜索歌手排序  搜索视频排序  搜索历史记录 | | |
| 实际输出 | 无法正常得到“周杰伦”综合、单曲、专辑、歌手、歌单、用户、视频搜索的结果 | | |
| 测试结果 | 未通过 | | |
| 问题描述 | 报错：  The method 'map' was called on null.  Receiver: null  Tried calling: map<SearchVideoWidget>(Closure: (Videos) => SearchVideoWidget) | | |
| 设计人员 | 黄德煜 | 设计日期 | 2020.12.15 |
| 测试人员 | 章拾瑜 | 测试日期 | 2020.12.22 |

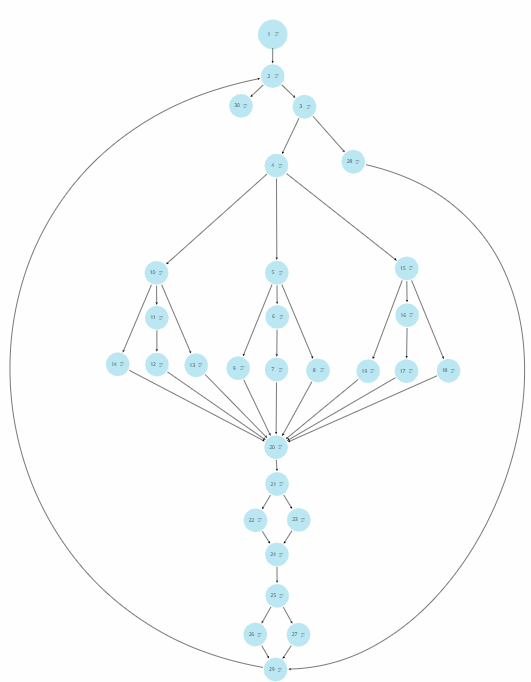
|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 播放器功能模块 | | |
| 标识符 | 6.0 | | |
| 测试项 | 播放器功能+歌词获取+音乐源获取+音乐详情+底部导航栏运行+音乐评论 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑，播放列表为有版权的音乐 | | |
| 操作步骤 | 进入播放界面  播放歌曲  暂停歌曲  快进后退歌曲  切歌  点击显示歌词  查看音乐评论  返回  查看音乐详情  退出  查看底部导航栏 | | |
| 输入数据 | 手势点击 | | |
| 预期输出 | 歌曲音乐源  歌曲封面  歌词  歌曲信息  播放器  歌曲评论  底部导航音乐播放器 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 通过 | | |
| 问题描述 | 无 | | |
| 设计人员 | 章拾瑜 | 设计日期 | 2020.12.16 |
| 测试人员 | 邢海粟 | 测试日期 | 2020.12.23 |

|  |  |  |  |
| --- | --- | --- | --- |
| 测试模块 | 播放器功能模块 | | |
| 标识符 | 6.1 | | |
| 测试项 | 播放器功能+歌词获取+音乐源获取+音乐详情+底部导航栏运行+音乐评论 | | |
| 测试环境 | 手机实际运行以及电脑Android/IOS模拟器 | | |
| 前置条件 | 能够正常使用的手机/电脑，播放列表为有版权的音乐 | | |
| 操作步骤 | 进入播放界面  播放歌曲  暂停歌曲  快进后退歌曲  切歌  点击显示歌词  查看音乐评论  返回  查看音乐详情  退出  查看底部导航栏 | | |
| 输入数据 | 手势点击 | | |
| 预期输出 | 歌曲音乐源  歌曲封面  歌词  歌曲信息  播放器  歌曲评论  底部导航音乐播放器 | | |
| 实际输出 | 如预期输出 | | |
| 测试结果 | 未通过 | | |
| 问题描述 | 歌曲无法播放，歌曲播放源url访问显示null | | |
| 设计人员 | 章拾瑜 | 设计日期 | 2020.12.16 |
| 测试人员 | 邢海粟 | 测试日期 | 2020.12.23 |

|  |  |  |
| --- | --- | --- |
| 测试模块：用户登录模块 | | |
| 标识符：User Login | | |
| 测试项：用户登录模块的URL测试 | | |
| 测试环境：postman | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过postman访问url | | |
| 输入数据 | 期望输出 | 实际结果 |
| 通过get方法提交  用户名 username ：admin  用户密码 password：admin | URL正常访问，并返回相关数据 | URL正常访问，并返回相关数据 |

### 4.3.2 白盒集成测试用例

基本路径覆盖图



PDL描述：

输入当前播放器的状态，是播放/暂停，还是下一首歌或上一首歌，还有播放歌曲模式：0对应顺序播放，1对应随机播放，2对应循环播放

Play = 0 or 1

Next = 0 or 1

Prev = 0 or 1

CurMode = 0 or 1 or 2

提前准备好音乐列表，音乐资源，音乐图片，音乐歌词以及音乐评论的数据

url\_music = ${music1}

url\_music\_png = ${music\_png}

url\_music\_comment = ${music\_comment}

url\_music\_lyric = ${ music\_lyric }

1:ClickToPlayPage()

Success = 0

2:DO WHILE GetUrlMusic(url\_music)

Success = 1

3: IF GetUrlMusicPage(url\_music\_png)

4: THEN printf(“已加载音乐资源”)

5: IF CurMode == 0

6: IF Play == 1

7: THEN PlayMusic(url\_music)

Printf(“音乐播放成功”)

ENDIF

8: ESIF Next == 1

THEN url\_music = ${music2}

Printf(“音乐切换下一首成功”)

Play = 1

Next = 0

ENDIF

9: ESIF Prev == 1

THEN url\_music = ${music0}

Printf(“音乐切换上一首成功”)

Play = 1

Prev = 0

ENDIF

10: ESIF CurMode == 1

11: IF Play == 1

12: THEN PlayMusic(url\_music)

Printf(“音乐播放成功”)

ENDIF

13: ESIF Next == 1

THEN url\_music = ${music2 }

Printf(“音乐随机下一首成功”)

Play = 1

Next = 0

ENDIF

14: ESIF Prev == 1

THEN url\_music = ${music1}

Printf(“音乐随机上一首成功”)

Play = 1

Prev = 0

ENDIF

15: ESIF CurMode == 2

16: IF Play == 1

17: THEN PlayMusic(url\_music)

Printf(“音乐播放成功”)

ENDIF

18: ESIF Next == 1

THEN url\_music = ${music1}

Printf(“音乐单曲循环成功”)

ENDIF

19: ESIF Prev == 1

THEN url\_music = ${music1}

Printf(“音乐单曲循环成功”)

ENDIF

20: ENDIF

21: IF GetLyric(url\_music\_lyric)

22: THEN Printf(“获取歌词成功”)

23: ELSE Printf(“获取歌词失败”)

24： ENDIF

25: IF GetComment(url\_music\_comment)

26: THEN Printf(“获取评论成功”)

ENDIF

27: ELSE Printf(“获取评论失败”)

ENDIF

28: ELSE printf(“加载音乐资源成功，但找不到对应图片”)

29:ENDDO

30:IF Success == 0

THEN printf(“加载音乐资源失败”)

ENDIF

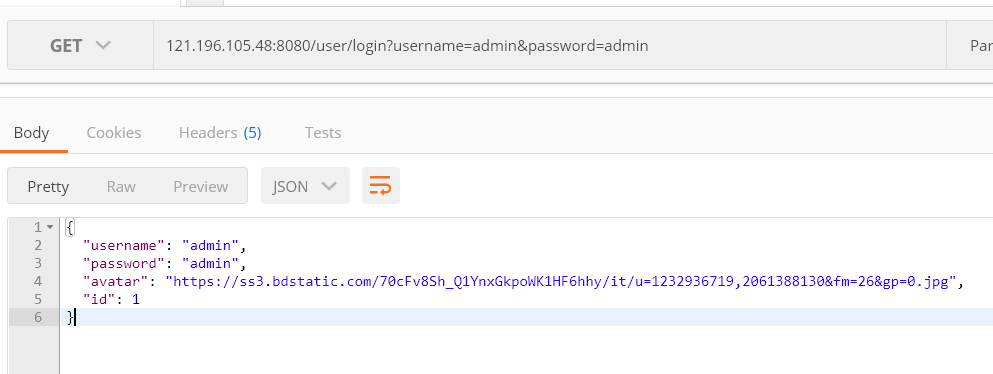
|  |  |  |  |
| --- | --- | --- | --- |
|  | 输入数据 | 测试路径 | 预期输出 |
| 测试用例1 | Play = 1,Next = 0,Prev = 0, CurMode = 0,  url\_music = ${SondId1},  url\_music\_png = ${PicUrl1},  url\_music\_comment = ${SondId1},  url\_music\_lyric = ${SondId1} | P(1,2,3,4,5,6,7,20,21,22,24,25,26,29) | 已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |
| 测试用例2 | Play = 1,Next = 0,Prev = 0, CurMode = 0,  url\_music = null,  url\_music\_png = ${PicUrl1},  url\_music\_comment = ${SondId1},  url\_music\_lyric = ${SondId1} | P(1,2,30) | 加载音乐资源失败 |
| 测试用例3 | Play = 1,Next = 0,Prev = 0,CurMode = 0,url\_music = ${SondId1}  url\_music\_png = null  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,28,29) | 加载音乐资源成功，但找不到对应图片 |
| 测试用例4 | Play = 1,Next = 0,Prev = 0, CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = null  url\_music\_lyric = ${SondId1} | P(1,2,3,4,5,6,7,20,21,22,24,25,27,29) | 已加载音乐资源  音乐播放成功  获取歌词成功  获取评论失败 |
| 测试用例5 | Play = 1,Next = 0,Prev = 0,CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = null | P(1,2,3,4,5,6,7,20,21,23,24,25,26,29) | 已加载音乐资源  音乐播放成功  获取歌词失败  获取评论成功 |
| 测试用例6 | Play = 1,Next = 0,Prev = 0, CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = null  url\_music\_lyric = null | P(1,2,3,4,5,6,7,20,21,23,24,25,26,29) | 已加载音乐资源  音乐播放成功  获取歌词失败  获取评论失败 |
| 测试用例7 | Play = 0,Next = 1,Prev = 0, CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,5,8,20,21,22,24,25,26,29) | 已加载音乐资源  音乐切换下一首成功  获取歌词成功  获取评论成功 |
| 测试用例8 | Play = 0,Next = 0,Prev = 1, CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,5,9,20,21,22,24,25,26,29) | 已加载音乐资源  音乐切换上一首成功  获取歌词成功  获取评论成功 |
| 测试用例9 | Play = 1,Next = 0,Prev = 0, CurMode = 1,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,10,11,12,20,21,22,24,25,26,29) | 已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |
| 测试用例10 | Play = 0,Next = 1,Prev = 0, CurMode = 1,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,10,13,20,21,22,24,25,26,29) | 已加载音乐资源  音乐随机下一首成功  获取歌词成功  获取评论成功 |
| 测试用例11 | Play = 0,Next = 0,Prev = 1, CurMode = 1,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,10,14,20,21,22,24,25,26,29) | 已加载音乐资源  音乐随机上一首成功  获取歌词成功  获取评论成功 |
| 测试用例12 | Play = 1,Next = 0,Prev = 0, CurMode = 2,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,15,16,17,20,21,22,24,25,26,29) | 已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |
| 测试用例13 | Play = 0,Next = 1,Prev = 0, CurMode = 2,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,15,18,20,21,22,24,25,26,29) | 已加载音乐资源  音乐单曲循环成功  获取歌词成功  获取评论成功 |
| 测试用例14 | Play = 0,Next = 0,Prev = 1, CurMode = 2,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,15,19,20,21,22,24,25,26,29) | 已加载音乐资源  音乐单曲循环成功  获取歌词成功  获取评论成功 |
| 测试用例15 | Play = 0,Next = 1,Prev = 0, CurMode = 1,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,10,13,20,21,22,24,25,26,29,2,3,4,10,11,12,20,21,22,24,25,26,29) | 已加载音乐资源  音乐随机下一首成功  获取歌词成功  获取评论成功  已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |
| 测试用例16 | Play = 0,Next = 0,Prev = 1, CurMode = 1,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,10,14,20,21,22,24,25,26,29,2,……) | 已加载音乐资源  音乐随机上一首成功  获取歌词成功  获取评论成功  已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |
| 测试用例17 | Play = 0,Next = 1,Prev = 0, CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,5,9,20,21,22,24,25,26,29,2,3,4,5,6,7,20,21,22,24,25,26,29) | 已加载音乐资源  音乐下一首成功  获取歌词成功  获取评论成功  已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |
| 测试用例18 | Play = 0,Next = 0,Prev = 1, CurMode = 0,url\_music = ${SondId1}  url\_music\_png = ${PicUrl1}  url\_music\_comment = ${SondId1}  url\_music\_lyric = ${SondId1} | P(1,2,3,4,5,9,20,21,22,24,25,26,29,2,……) | 已加载音乐资源  音乐上一首成功  获取歌词成功  获取评论成功  已加载音乐资源  音乐播放成功  获取歌词成功  获取评论成功 |

环形复杂度为：V（G） = E（边） – N（节点） + 2 = 38 – 30 + 2 = 10

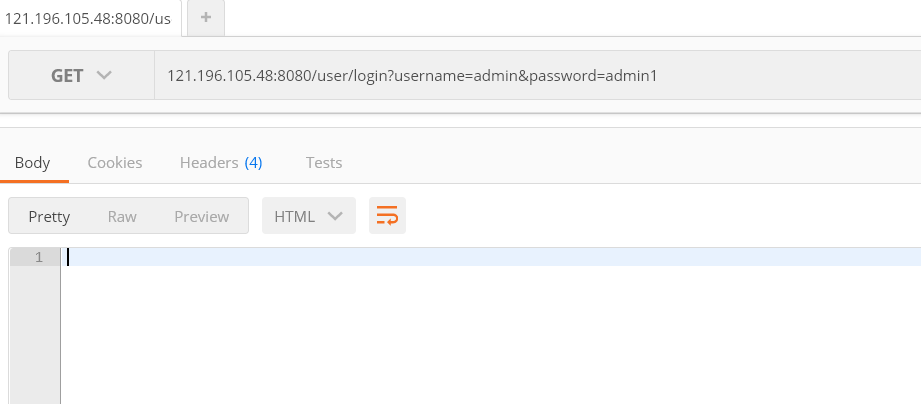
### 4.3.3 后端集成测试

**postman后端集成测试**

用户名密码存在

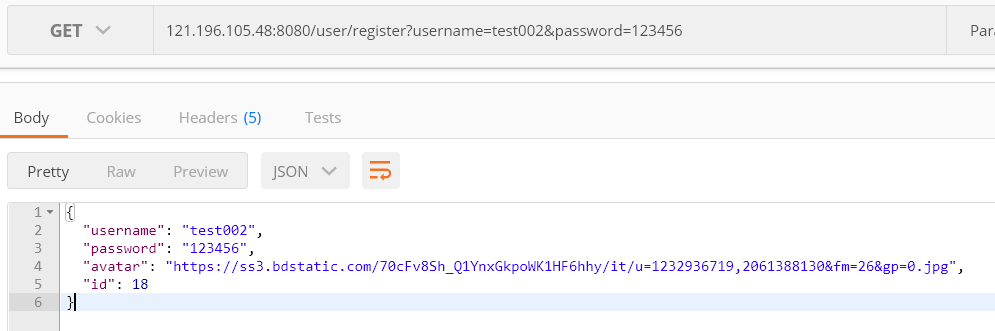


用户名或密码错误

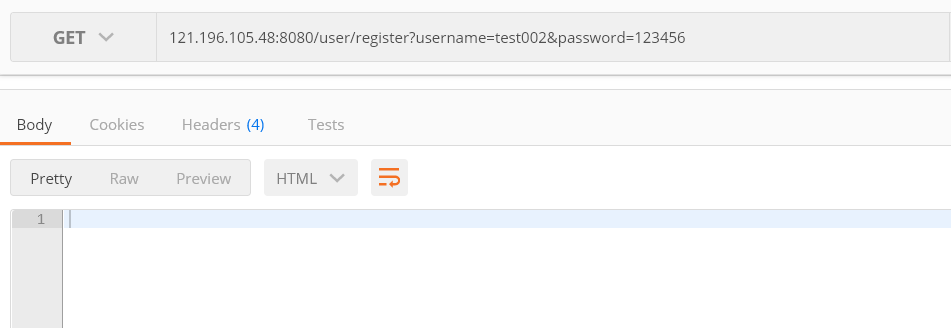


|  |  |  |
| --- | --- | --- |
| 测试模块：用户注册模块 | | |
| 标识符：User Register | | |
| 测试项：用户注册模块的URL测试 | | |
| 测试环境：postman | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过postman访问url | | |
| 输入数据 | 期望输出 | 实际结果 |
| 通过get方法提交  用户名 username ：admin  用户密码 password：admin | URL正常访问，并返回相关数据 | URL正常访问，并返回相关数据 |

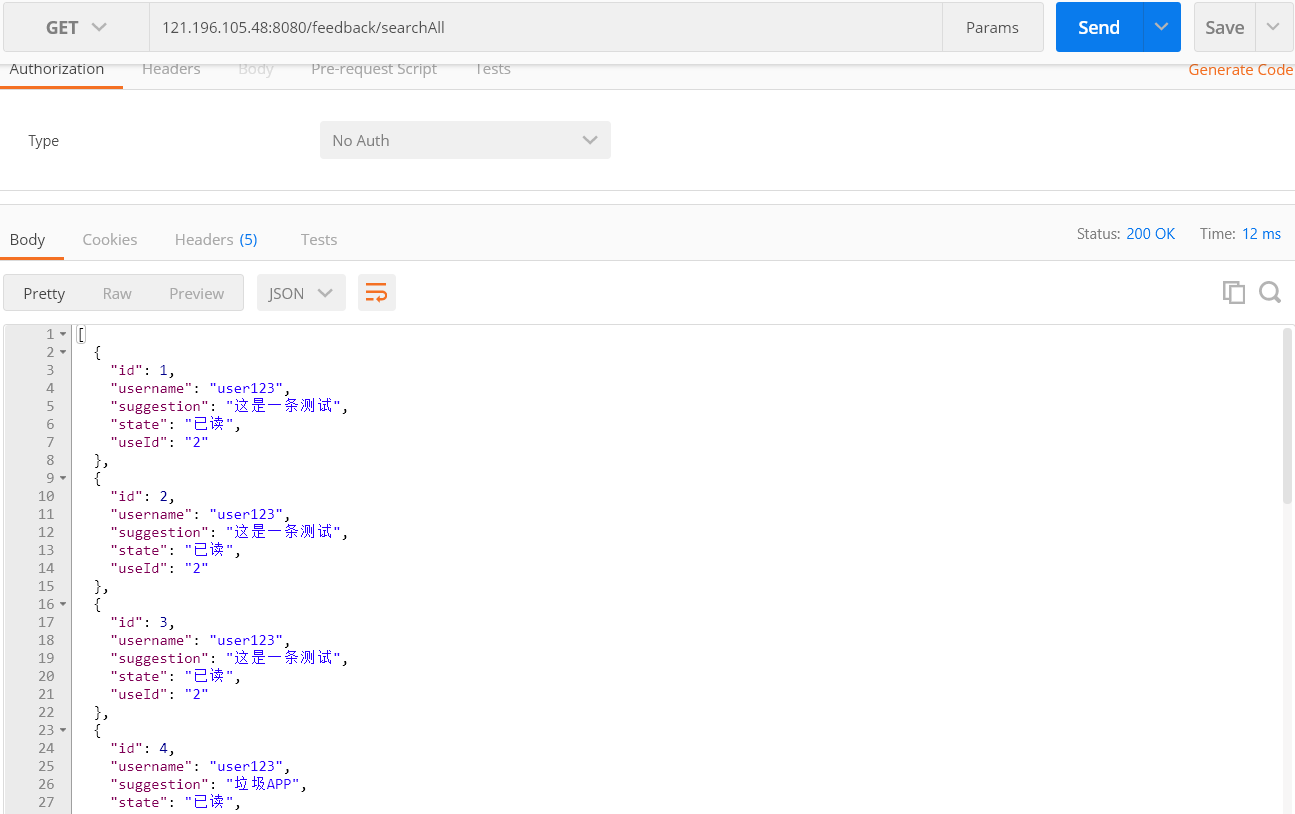
用户名未注册



用户名已注册

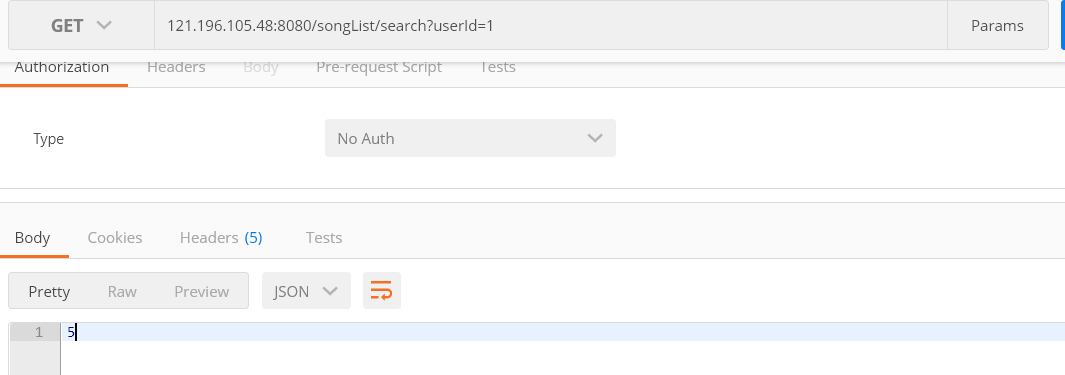


|  |  |  |
| --- | --- | --- |
| 测试模块：用户反馈模块 | | |
| 标识符：User Feedback | | |
| 测试项：用户反馈模块的URL测试 | | |
| 测试环境：postman | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过postman访问url | | |
| 输入数据 | 期望输出 | 实际结果 |
| 通过get方法提交数据 | URL正常访问，并返回相关数据 | URL正常访问，并返回相关数据 |

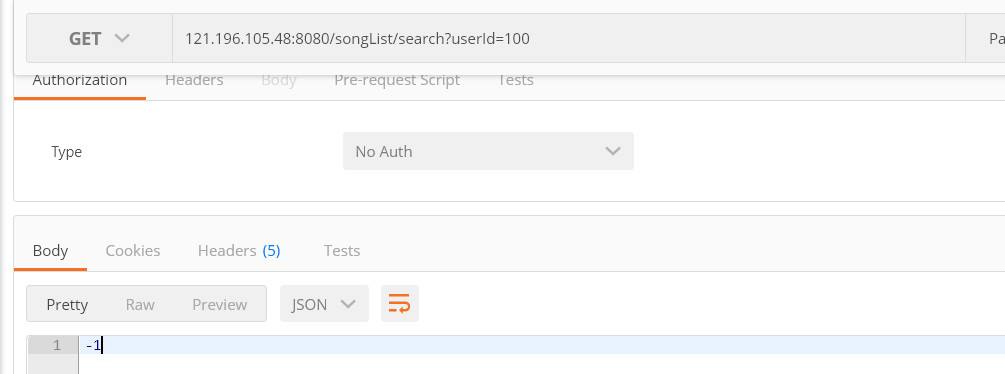


|  |  |  |
| --- | --- | --- |
| 测试模块：歌单模块 | | |
| 标识符：Song List | | |
| 测试项：歌单模块的URL测试 | | |
| 测试环境：postman | | |
| 前置条件：数据库等其他运行环境正常 | | |
| 操作步骤：  通过postman访问url | | |
| 输入数据 | 期望输出 | 实际结果 |
| 通过get方法提交数据 | URL正常访问，并返回相关数据 | URL正常访问，并返回相关数据 |

userId存在



userId不存在



### 4.3.4集成测试问题修正记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例标识符 | 错误或问题描述 | 问题原因 | 解决方案 | 测试人员 |
| 3.1 | 未绑定网易云账户时，点击每日推荐页面TestFailed，无法加载出歌曲列表并报错 | 访问推荐页面出错，DioError。每日推荐基于网易云平台用户大数据搭建，需要提供用户登录后的cookie | 提供点击重试的按钮，并告知用户需要绑定网易云账户 | 章拾瑜 |
| 3.2 | 推荐模块的功能中的每日推荐只能显示前20个 | 推荐歌单接口返回值提供两种歌曲数组[trackIds]、[track]，其中trackIds只提供前20首歌曲的相关数据trackIds，Flutter列表构造器基于trackIds数据构造，在构造超过20首歌曲时无法正常获取数据导致报错 | 将Flutter列表构造器改为基于track数据构造，获取接口中的所有歌曲ID数据track，通过track数据中的ID遍历歌曲接口/song/detail?id=，获取所有歌曲数据后重新传值 | 章拾瑜 |
| 5.2 | 无法正常得到“周杰伦”综合、单曲、专辑、歌手、歌单、用户、视频搜索的结果 | 在搜索部分内容时，搜索结果SearchVideoWidget单元返回空值，排查问题发现搜索接口中有部分搜索内容无法正常得到“视频”的搜索结果（可能是版权问题） | 暂时取消视频搜索的结果 | 章拾瑜 |
| 6.1 | 播放没有版权的歌曲时，歌曲无法播放，歌曲播放源url访问显示null | 没有版权的音乐播放源为null | 播放前检测是否音乐播放源是否返回null，若返回null，则提示无法播放，并自动跳到下一首/上一首可以播放的音乐 | 邢海粟 |

# 5系统测试

工具为**firebase**

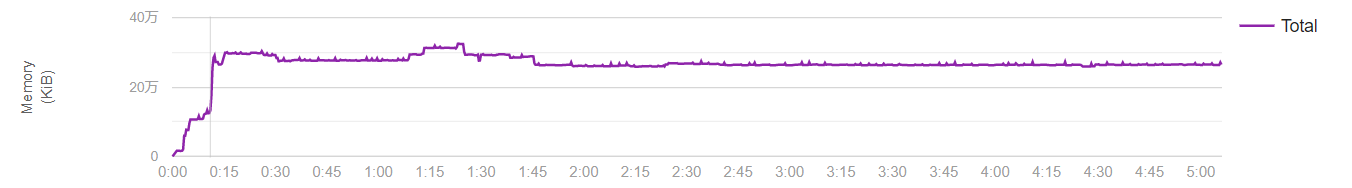
## 5.1 CPU占用率

## 

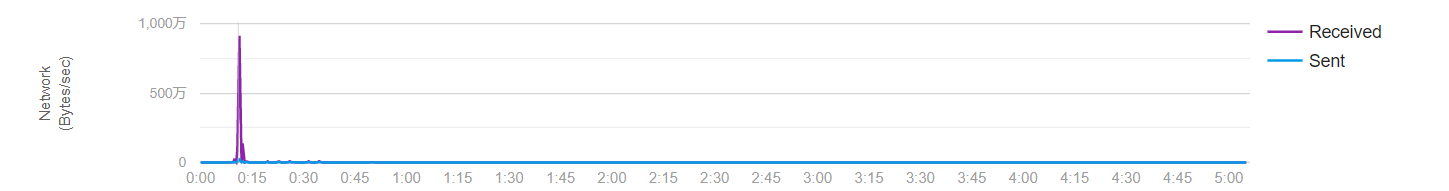
## 5.2 Graphics画面每秒传输帧数（FPS）



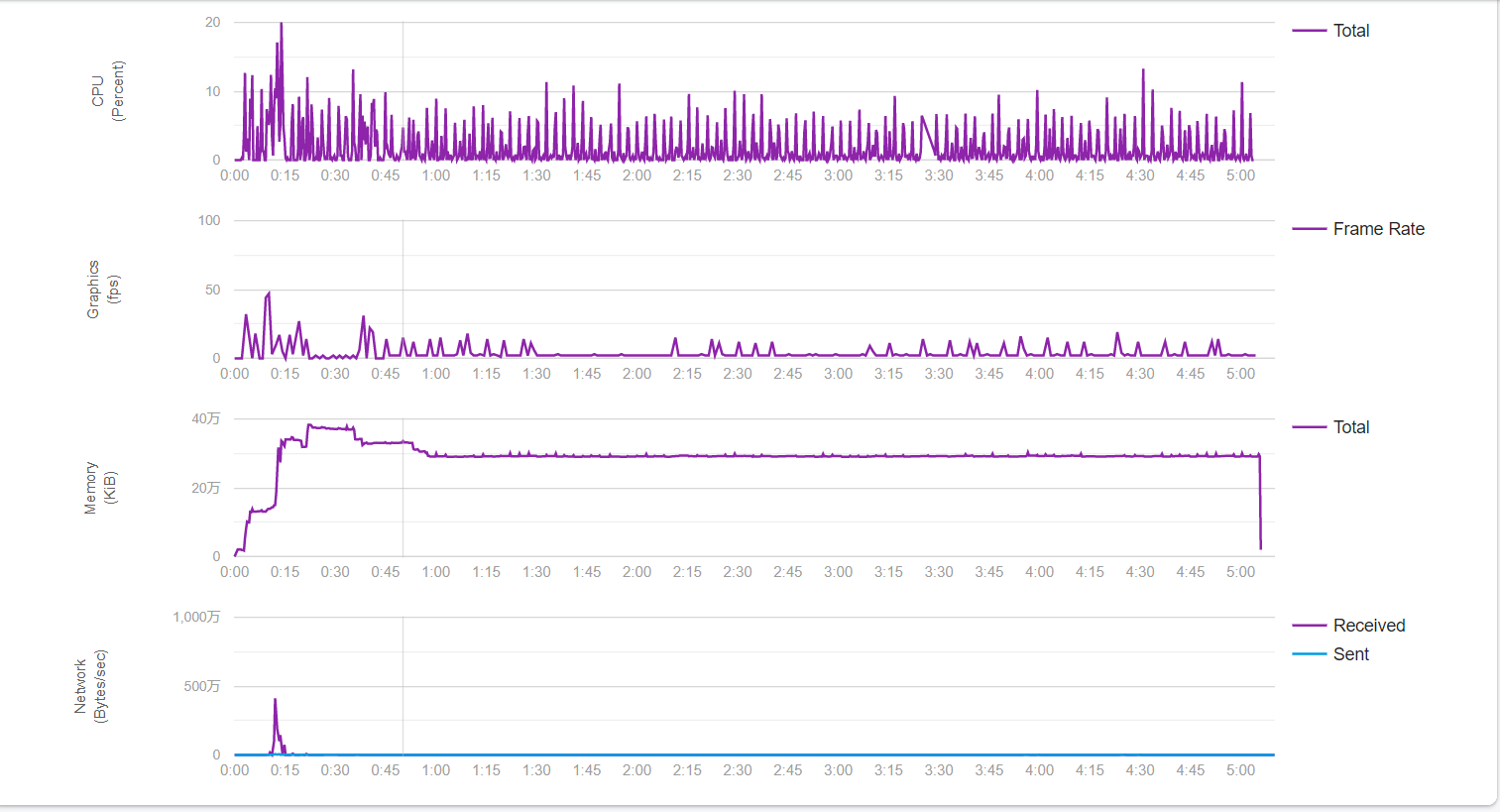
## 5.3 Merrory内存占用（单位KiB）



## 5.4 NetWork网络请求（单位byte）



## 5.5 性能测试





# 6测试活动总结

人力消耗：团队3人

资源成本消耗：能跑手机模拟器的电脑x3，15天测试周期时间

# 7注解

本章应包含有助于理解本文档的一般信息(例如背景信息、词汇表、原理)。本章应包含为理解本文档需要的术语和定义，所有缩略语和它们在文档中的含义的字母序列表。

# 附录

附录可用来提供那些为便于文档维护而单独出版的信息(例如图表、分类数据)。为便于处理，附录可单独装订成册。附录应按字母顺序(A,B等)编排。