**《人机交互技术》设计报告**

|  |  |
| --- | --- |
| **专    业：** | **软件工程** |
| **学    号：** | **201905555204** |
| **姓    名：** | **章航滔** |

2021年12月25日

目 录

[一、 概述 3](#_Toc13442)

[1.1 系统设计的目标 3](#_Toc6530)

[1.2 系统设计的意义 3](#_Toc6360)

[1.3 与法律、环境、经济及可持续性的相关性 3](#_Toc3145)

[二、 需求分析 4](#_Toc10317)

[2.1 用户分析 4](#_Toc20777)

[2.2 用户设计流程 5](#_Toc6)

[2.3 任务分析 6](#_Toc22156)

[三、 设计 7](#_Toc19284)

[3.1 界面设计原则 7](#_Toc24286)

[3.2 设计模型 9](#_Toc32203)

[3.3 系统描述 9](#_Toc18135)

[3.4 使用的交互技术 10](#_Toc8938)

[四、 界面实现 12](#_Toc12023)

[五、 可用性评估 16](#_Toc6190)

[5.1 评估指标体系 16](#_Toc10717)

[5.2 评估技术 16](#_Toc27780)

[5.3 评估结果分析 16](#_Toc24166)

[六、 总结 16](#_Toc14447)

[参考文献 18](#_Toc17334)

1. **概述**
   1. 系统设计的目标

本系统是基于Java的音乐播放器设计，运用了JetBrains IntelliJ IDEA Ultimate作为开发软件,采用JAVAFX作为GUI框架。

**本系统设计的目标主要有如下：**

1.界面简洁追求精美,UI完善且交互性高  
2.支持本地播放的音乐格式：.mp3 文件、.wav 文件、.aac 文件  
3.支持解析歌词（.lrc文件）并以动画形式滚动展示。

4.支持解析歌词文件（缩略图、专辑、时长等）  
5.支持频谱图动效，频谱图转换为动态光晕显示。  
6.具有自动扫描文件夹的音乐列表，支持拖动添加歌曲，歌词文件(mpc,lrc均支持)  
7.支持歌单列表控制，支持自定义快捷键，以键盘驱动对音量与进度进行调节

8.达到动画效果流畅，用户体验良好的效果

9.用户个人喜好设置，添加喜欢的音乐列表

* 1. 系统设计的意义

音乐是生活的必需的，它能够缓解人们焦躁的心情，调节现代化生活的节奏，让人们能够舒缓身心。一个好的音乐播放器，能够实现个性化的功能，它能够根据个人的喜好来推荐歌曲。网易云、QQ音乐播放器正是实现了此功能，它的个性歌曲推荐就是体现了人性化的特点。听音乐已经成为当今社会，人们必不可少的一项娱乐活动。根据调查所得信息，我发现最近电脑用户对较大的音乐播放器应用的喜爱度明显下滑，而相对的那些小巧的音乐播放器反而被越来越多的用户下载使用。

随着人们物质生活水平的提高，人们越来越注重精神生活的满足，热衷于在消费中寻求快乐和娱乐体验。音乐作为人类娱乐生活的重要组成元素，一直以来都备受关注。而人类进入工业社会以来，将音乐播放与工业产品结合发展出了一系列的音乐播放产品，并随着技术的革新和消费者需求的变化而不断更新，为人类的娱乐生活提供了时尚便利的道具。

特别是市场呈现红海，满足“喜好者”的需要是很常见的一些功能，音乐播放器、音乐曲库、音乐下载、音乐铃声制作等。这些基本功能的技术门槛并不高，于是，成败的关键问题就变成了软件的用户体验感是否达到要求。

因此，设计一款用户体验度高，功能完善，界面优化与高交互性的音乐播放器具有重大的价值与现实意义。

1.3 与法律、环境、经济及可持续性的相关性

经济及可持续性的分析要从中国音乐播放器行业的发展现状、竞争格局及市场供需形势进行具体分析，从行业的政策环境、经济环境、社会环境及技术环境等方面分析行业面临的机遇及挑战，了解重点音乐公司的经营现状及发展格局，作概述如下：

随着移动端、PC端应用的井喷式上升，音乐市场的蛋糕的瓜分愈加激烈，从百度指数上面的数据表明，酷狗音乐的用户多集中于20-29岁和30-39岁年龄段，自然这个年龄段的更加接近互联网而且用户粘度更高。性别上男士以69%远大于女性的31%。另一方面在百度指数的需求图谱上可以看出，作为酷狗音乐搜索的热度中像音乐盛典等词汇上升，这也体现手机音乐app线下的体验开始上升，随着音乐类综艺节目的如火如荼，“中国好声音”、“我是歌手”等节目的音乐资源也是用户刚需的一部分。作为手机音乐市场上的大佬，酷狗音乐的市场份额一直保持着领先，14年占据了24.3%的市场，紧随的有QQ音乐、酷我音乐和天天动听。虽然酷狗音乐在手机市场上的份额第一，但伴随手机音乐用户的个性化差异以及增值产业的创新，如何更好的完善自己手机市场的生态圈以及如何利用“平台化”来增加用户流量，对于各大音乐软件来说，都还有很远的路要走。

随着计算机应用的普及，越来越多的开源音乐播放器软件面向大家开放，音乐作为人类娱乐生活的重要组成元素，一直以来都备受关注。而人类进入工业社会以来，将音乐播放与工业产品结合发展出了一系列的音乐播放产品，并随着技术的革新和消费者需求的变化而不断更新，为人类的娱乐生活提供了时尚便利的道具，众所周知音乐是生活的必需的，它能够缓解人们焦躁的心情，调节现代化生活的节奏，让人们能够舒缓身心。一个好的音乐播放器，能够实现个性化的功能，它能够根据个人的喜好来推荐歌曲。网易云、QQ音乐播放器正是实现了此功能，它的个性歌曲推荐就是体现了人性化的特点，听音乐已经成为当今经济社会中人们必不可少的一项娱乐活动。设计一款用户体验度高，功能完善，界面优化与高交互性的音乐播放器具有重大的价值与现实意义。

在法律可行性方面，本次开发的音乐播放器所有使用的包和软件都选用了正版，对转载的内容也添加了引用等标记，并且不存在侵权等行为，所有技术资料都由提出方保管。

1. **需求分析**

2.1 用户分析

首先，我们对使用软件的群体进行了分析，可划分为5类人，但因为社会环境、音乐素养和音乐市场的差异，各类人群在不同国家的占比有些差异，这也就导致同一类型的音乐播放软件，在国内外的境遇不同。

|  |  |  |  |
| --- | --- | --- | --- |
| 大类 | 说明 | 国际市场 | 中国大陆市场 |
| 喜好者 | 喜好音乐,把音乐作为自己生活和工作的背景和陪伴,倾向于音乐播放器的功能 | 大众需求 | 大众需求 |
| 痴迷者 | 痴迷于某一类或几类细分类型音乐、或是某个歌星乐队,追求音乐播放器的极致音乐品质。 | 小众需求 | 小众需求,但未被 很好的满足 |
| 娱乐者 | 将音乐作为娱乐方式,比如卡拉OK、各类音乐游戏，热衷于播放器的效果与界面设计 | 大众需求 | 大众需求,但未被 很好的满足 |
| 玩票者 | 业余爱好者中的专业音乐人,讲究专业水准，对歌曲添加、收藏、导出等相关便捷程度要求高 | 大众需求 | 人数很少,还未到 达小众需求 |
| 专业者 | 专业音乐工作者，需要查看歌曲的频谱等内容 | \ | \ |

* 分析了上述使用群体后，可以从音乐播放器的使用者的角度，概括出用户的需求主要有以下7类：
* **播放歌曲**
* **导入歌曲**
* **歌词展示**
* **歌曲收藏**
* **歌曲删除**
* **频谱展示**
* **模式切换**

2.2 用户设计流程

* **本音乐播放器的用户使用流程可以总结如下：**

1.展示出本地的音乐文件，并显示该文件的相关信息，例如歌曲名、演唱者、长度、所属唱片、歌曲的封面图片；

2.音乐可以按照演唱者、专辑名称、男女歌手、歌曲类型等信息进行分类，进入每个分类后，按照字母顺序展现歌曲列表；

3.删除选中的本地音乐；

4.播放本地音乐，在播放的过程中，可以停止播放、回复播放、播放下一首、播放上一首、拖动音乐进度条到音乐任意时间点播放；

5.播放时，能显示当前播放音乐的进度，指示播放的时间；

6.播放时，能实时的显示当前歌词；

7.能添加选定的歌曲到播放列表，让播放器播放列表中的歌曲，播放的顺序可以设置顺序播放、单曲循环播放、随机播放等等；

8.音乐播放器等界面退出以后，音乐仍然能在后台播放；再次打开播放器，能显示当前播放的实时信息，例如播放进度条；

9.记录每首音乐的播放历史，下一次播放音乐就从以前播放到的地方继续播放；

10.可以连接到网络，播放网络上的音乐；

11.下载网络上的音乐；

12.让用户标注喜欢的音乐，允许用户给音乐写评论；

* **音乐播放器的用户设计流程可以总结如下：**

1.用户在MusicListActivity通过长按，开始选择多首音乐，组成一个播放列表；

2.这个播放列表被传递给MusicService，MusicService操作 Playl istcont entProvider 清空原有的播放列表，然后将这份新的列表存 储到PlayListContentProvider 当中；

3.用户点击MusicListActivity 上的播放按钮时，MusicService开始从播放列表获取第一首曲子开始播放；

4.播放的过程中，MusicService将当前播放的进度实时更新到PlayListcontentProvider中；

5.当播放的音乐有变化（播放完成、切换歌曲、播放进度每秒的变化)，MusicService都将通知给MusicL istAct ivity，让其能够同 步的改变界面显示；

6.播放过程中，MusicListActivity 能通过调用MusicService提供的接口控制音乐的暂停、继续、播放上一首、下一首；

7.如果用户单独点击音乐列表中的音乐，该音乐将被添加到播放列表中的第一位，并开始播放；

8.当播放到以前播放过的歌曲，将从它曾经停止播放的位置继续播放，如果之前该乐曲被播放完毕，这次播放则从头开始播放；

* **确定了音乐播放器的用户设计流程后，还需要给出实现这些功能的一些基本假设：**

1.设备上可被播放的音频文件很多很多，可能是通话录音，可能是语音记事本，为了简便操作，我们将认为音乐文件都放在包含了 music关键字的路径当中；

2.设备上已经准备好自带封面的音乐。比如说一首MP3格式的音乐，它所包含的内容并不只是音乐本省，还包括了很多元信息，例如歌曲的作者，歌曲的所属专辑，甚至歌曲的专辑封面图片等很多丰富的信息。只不过我们通常只注意到了MP3格式的音乐本身的音 乐内容和文件名字，没有直观的看到其它这些信息。元信息是可选的内容，所以有的音乐文件也没有把对应的信息填充到里面去。 （如果没有封面图片，会让应用显示默认的封面图片）

2.3 任务分析

结合需求分析来看，整个设计包含了简单的菜单栏，根据用户需求设置了三个简单的实现基本操作的菜单，分别是：文件，播放控制，播放列表；根据具体的操作每个菜单还包含了相应的菜单项， 文件：打开，循环，退出；播放控制：播放，暂停，停止；播放列表则没有菜单项，该菜单就是用来显示或者查看相应的歌曲目录的。

该音乐播放器设置了两个窗口，运行程序时打开的初始化欢迎界面(加载动画)，用户可以根据自己的需求进行菜单栏里面的基本操作和相应的设置，当用户添加或者打开音乐播放文件的时候，程序将会显示出第二个音乐文件播放窗口，与此同时，第一个初始化窗口将会自动隐藏，在音乐播放窗口里面用户可以进行相关的音乐播放的控制操作。

当用户进入音乐播放窗口时，该窗口有几个快捷控制的控键：时间追踪轴：可以实现播放的前进、后退功能，根据用户需求调节音乐的播放进度等；音量控制：设置静音，打开扬声器或者调节音量大小；播放列表：查看当前播放音乐的相关属性。

**(1)功能任务要求:** 读取用户的 Mp3等音乐文件，加载到播放列表，用户可以点击播放可以选择播放模式，设置，搜索列表歌曲等功能，界面清新，UI效果良好，列任务表如下：

|  |  |
| --- | --- |
| 对象 | 总体功能与任务 |
| 音乐播放器 | Javafx播放器具有:"歌曲列表","专辑列表","艺术家列表","播 放主界面","搜索按钮","暂停/播放","设置","上一首","下 一首"等 |
| 使用者（用户） | 1.通过用户界面可导入音乐,将歌曲按歌曲,专辑,艺术家,三种方式排列。 2.点击音乐播放 3.点击暂停/播放键,暂停或播放音乐 4.点击上一首/下一首可以切换歌曲 5.点击搜索按钮搜索本地歌曲 6.点击设置,设置正在播放,动画,系统动画,艺术家风格显示, 选择主题,起始页,拔出暂停等 7.点击关于可以了解版本信息等 8.点击帮助反馈,弹出邮件可以编辑自己需要反馈的意见 9.点击播放主界面按钮,进入播放的主界面 |

**(2). 性能任务要求:** 用户层面的完整，方便，快捷，具体要求如下：

根据系统要求无响应时间为5秒，所以就有如下性能要求：

(1)当要求歌曲播放时，程序响应时间最长不能超过5秒

(2)当要求歌曲暂停时，程序响应时间最长不能超过5秒

(3)当要求歌曲停止时，程序响应时间最长不能超过5秒

(4)当要求歌曲上/下一首时，程序响应时间最长不能超过5秒

(5)当要求进行清单列表时，程序响应时间最长不能超过5秒

运行环境需求:

操作系统：基于windows x64的操作系统

支持环境：JAVAFX环境

开发环境：Jetbrains IDEA 2020.2.3

**(3). 输出任务要求:** 播放的音乐正确完整，无卡顿与音频信息缺漏现象。

1. **设计**

3.1 界面设计原则

本系统从人机交互界面设计的五大原则入手，围绕此为基础开展了用户界面的设计。具体阐述如下：

* **原则一：美学完整性**

美学完整性并不是用来衡量一个软件的界面有多好看，而是用来衡量软件的界面与功能是否匹配。例如，对于一个应用来说，会用比较微妙的修饰元素和背景来体现生产性的任务，对于突出的任务则会使用标准的控件和操作行为。这样的软件会传达一个清晰和统一的信息给用户，让用户懂得APP软件的目的。但要是软件在生产性任务上使用了异想天开的元素，用户就会被这些相互矛盾的信号所困扰。

同样的，在设计的音乐播放器的软件里，用户希望一个漂亮的界面来提供更多的乐趣从而鼓励他们继续使用软件。所以，进入主界面之后，就可以看到它整体的设计风格，以暗黑色调为主，大胆创新使用了黑胶唱片的播放器的形式，让整个界面看起来十分温和而富有科技感，从而让用户能更加沉浸在音乐世界里。

* **原则二：一致性**

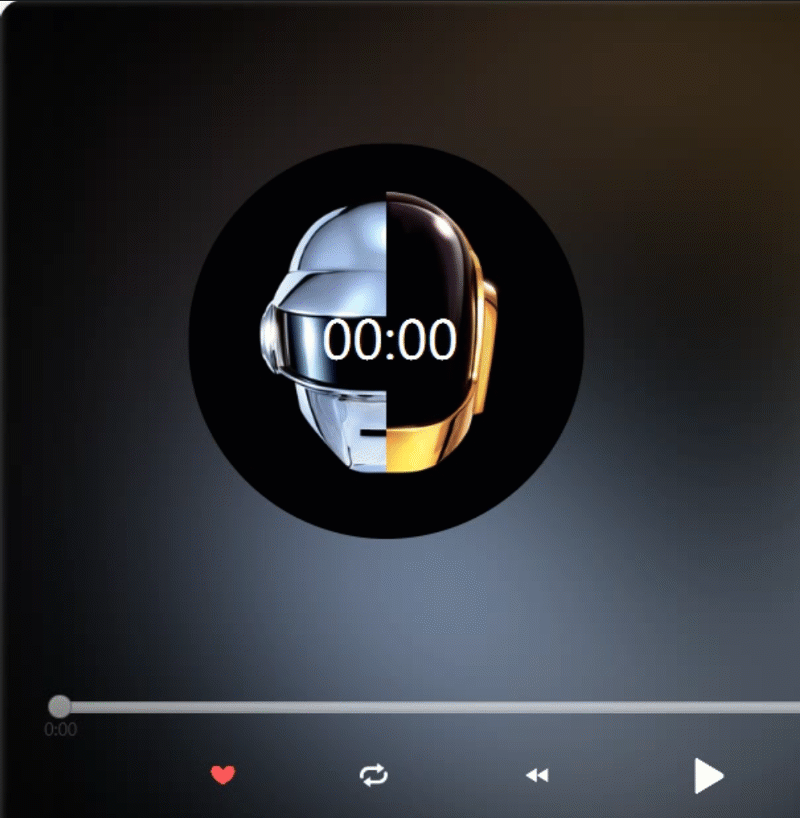
软件界面的一致性允许用户利用我们自身的知识和技能完成软件的过渡。当然，保持界面的一致性，并不是让我们盲目的复制其他的软件，因此在一致性方面，整体的UI都十分简洁明了，用矩形，三角形，圆形等简单的图案构成。图标设计都是圆润型的，基本没出现有尖图标的设计，UI设计具有一致性。

**原则三：直接操作**

当用户操作屏幕上的对象而不是单独的控件进行操作时，用户更能理解自己的任务和操作的结果。而且普遍的用户更喜欢直接的操作行为，基于这个特性，设计的音乐播放器中特别应用了快捷键操作，用户可以运用UP,DOWN,LEFT,RIGHT等键盘方向键轻松地对音量或者歌曲的进度直接进行调节。

**原则四：隐喻**

当虚拟的物品和操作行为是对现实生活中的物品和操作进行模仿的时候，玩家能够很快的掌握APP软件的规则和操作。一个简单的软件隐喻是文件夹：在现实生活中，人们把物品放入文件夹中，所以在计算机中，用户能够很快的理解这个操作的意思，很快的理解将文档等文件放入文件夹的意思。  
  最合适的隐喻应该是不受现实生活的限制。例如，用户可以将很多内容放入文件夹中，包括音乐、文档、视频等，但现实生活中显然无法做到。所以，这个原则在播放器中的应用就在于：  
   A. 音乐播放的倒退控制  
   B. 歌曲列表栏的拖拽操作，如图所示：



* **原则五：用户控制**

在音乐播放器的软件设计中，是用来控制操作行为的，尽管软件能够建议用户做哪些操作或者警告用户有危险，但是这显然对于应用本身是个错误，因为这会将玩家推得更远。成功的软件应该在一味的提示和关键时刻做提示之间进行平衡。当对操作行为和控制感到熟悉和可预见的时候，用户将会明显感觉到自己对于软件的控制。操作行为越是简单直接，玩家就能更容易的理解和记住。所以，对于每一个按键，采用了最简单直观的图像进行表达，如图所示：

a.顺序播放 b.随机播放 c.循环播放

3.2 设计模型

本基于音乐播放器系统采用的设计模型基于JAVAFX，从背景图(由原图+高斯模型+毛玻璃化处理得到)开始布局，如图所示：



具体有遵循了以下原则：

1）界面要具有一致性

2）常用操作要有快捷方式

3) 提供必要的错误处理功能

4) 提供信息反馈

5) 允许操作可逆

6) 设计良好的联机帮助

7) 合理划分并高效地使用显示屏幕

3.3 系统描述

本系统是采用JavaFX组件完成的一款小巧、界面精美的本地音乐播放器，在本地音乐模块主要采用目前java最主流的音频标记库Jaudiotagger，可解析MP3文件头信息，提取音乐封面，以及其他信息等等，支持的音频文件主要为 MP3,WAV。可读取音频文件头信息里的歌曲名，专辑、歌手名等元信息。采用了一系列UI提高人机交互使用感受，支持窗口区域化拖动、支持（拖拽）添加本地音乐、歌曲以及删除它们、具有歌词lrc文件自动扫描解析、MP3文件自动扫描解析(获取mp3文件中的缩略图，歌手，专辑等信息)、三种播放模式选择(列表循环，单曲循环，随机播放)、歌词动画滚动、歌单列表控制、频谱图展示、频谱图转换为动态光晕显示、歌曲缩略图显示、自定义背景等功能。

3.4 使用的交互技术

本系统用到的人机交互技术主要有以下四种，分别是定位、定值、选择、几何约束

**(1)定位：**确定平面或空间的一个点的坐标。

**(2)定值：**用于设置物体的旋转角度，缩放比例等。

**(3)选择：**在某个选择集中选出一个元素，通过注视，指点或接触一个对象，使对象成为后续行为的焦点。

**(4)几何约束：**用于对图形的方向，对齐方式等进行规定和校准。

接下来我将从具体的功能举例来分析介绍四种交互技术在本系统中的应用：

**一、导航**

交互方式：遵循JAVAFX桌面应用软件与音乐播放器的导航规则，主菜单放置在最下方，方便切换，层级明确。



**二、音乐图标的动效**

交互方式：播放器图标设计成音阶，跳动为播放，静止则暂停，并基本固定在每个页面顶部栏

的右侧

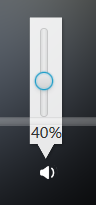
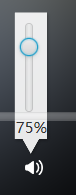
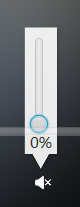
**三、搜索栏**

交互方式：

1、占据顶部栏的大部分位置配合搜索的文字，一目了然；

2、点开后跳到搜索界面，展示的内容层级分明。

**四、小图标的友好性**

如上图所示，在设计音量按键的时候，不仅仅采用了弹出式按键设计，而且进一步为了用户更直观的感受到音量的大小变化以及当前音量大小，下方的图标也会和当前音量绑定，呈现3种不同形态的变化，通过两个全满和半满的状态很直接友好的显示出了当前的音量状态，加上旁边的数字说明，哪怕一开始没有完全注意图标的差异的人，也会通过数量的对应发现图标的差异有种超出预期的感受。



特别是在鼠标悬停于按钮上后，为了加强交互效果，按键会呈现出颜色的变化或者灰度深浅的变化。

**五、音乐标记**

后台会标记收听的最后一个音乐，当用户退出程序后重新回到该歌曲，可以继续收听之前没听完的音乐。

**六、分享音乐**

用户分享时选择音乐，音乐没有让用户从曲库中寻找，而是提供了搜索栏和最近播放的5首歌，这是充分考虑了用户场景做的决定。当用户正在听某首歌觉得很棒，那他会立刻分享，所以最近播放里会有他想分享的歌。还有一种情况是用户想分享此时的心情，所以需要找一首配乐，这时候用户一般心里已经有比较明确的目标歌曲，所以只要通过搜索功能就能快速实现目标。可以发现搜索栏有一些历史搜索记录，而这些记录与首页搜索栏的记录如出一辙，用户可以很方便的根据自己的历史记录搜索到自己想要的歌曲。

**七、管理歌单**

对歌单的管理包括创建、删除和排序，网易云音乐将这些操作放在了页面左上角“更多”中。作为操作键，对要执行操作的对象-歌单距离却很远，是不利于用户操作的，降低了产品的可用性。

所以本系统采取的歌单列表是一个可以是发现音乐栏内的音乐，也可以是本地音乐，还可以是搜索结果后的，如果要加载本地音乐，点击本地音乐栏右边的文件夹按钮，打开本地音乐文件夹，一般这个文件夹都是相对于程序所在路径，在LocalMusic/Muisc文件夹内放音乐文件重新点击本地音乐栏就能加载本地音乐，如果需要歌词，请在LocalMusic/Lrc文件夹下放.lrc文件，而且下载当前音乐会把音乐文件下载到LocalMusic文件夹内，lrc和音乐文件会放到对应的文件夹。表格右键会弹出菜单，比如你喜欢当前播放的专辑封面，可以点击菜单栏复制到剪切板。

**八.****拖拽添加歌曲：**

本系统对本地歌曲播放功能做了优化,用户可以轻便的使用鼠标拖拽功能将电脑里的歌曲、歌词文件轻松地加入播放器，此外，也可以将唱片图片以选择右键弹出菜单复制链接或者直接拖拽图片出来，鼠标松开即可将图片保存到本地。

在JavaFX中关于拖放操作，实现拖拽添加歌曲需要设置以下两种事件监听器：

* setOnDragDetected(new EventHandler());

当用户从一个Node上进行拖动的时候，会检测到拖动操作，将会执行这个EventHandler。

* setOnDragEntered(new EventHandler());

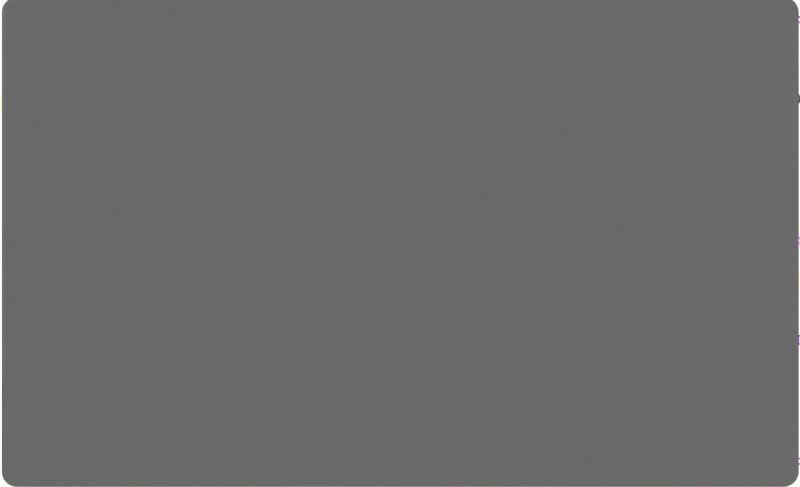
当用户拖动到目标控件的时候，会执行这个事件回调，拖动并松手的时候，执行Drag完成操作，此时我们就可以将歌曲文件复制到目标路径，完成添加，如下图所示：



1. **界面实现**

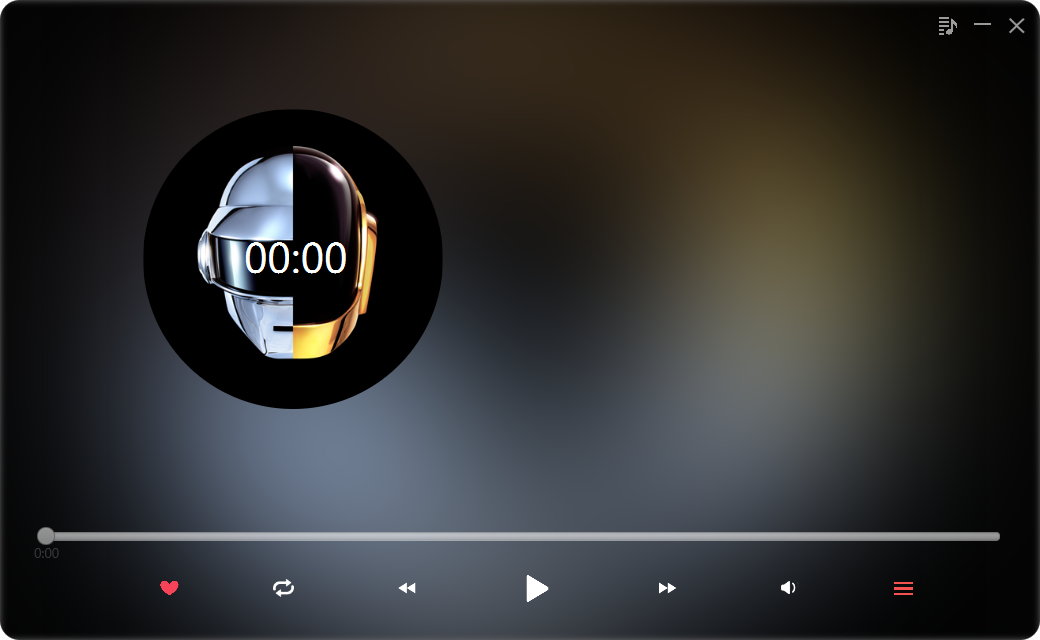
**4.1 界面加载动画**

页面加载的时间除了与处理请求的时间有关外，还会受到网络状态、以及用户电脑性能等因素的影响，所以在设计客户端软件时，加载动画是一个较好的选择，在播放动画的同时可以实现资源预加载,加载内部存储的音乐文件，从而较好地**优化用户体验**。此外，动画设计的主色调是淡色系，又搭配了不同的彩色。红黑让人感觉热情，而白色又让人感受到纯洁。整体给人的感受就是温暖而富有现代感的一个界面。如下图所示：

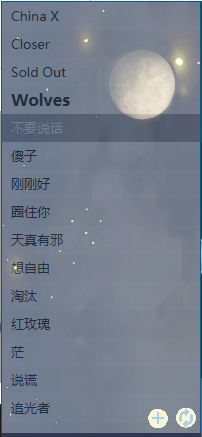
****

**4.2 主引导界面**

进入主界面之后，就可以看到它整体的设计风格，以暗黑色调为主，大胆创新使用了黑胶唱片的播放器的形式。而且整体十分简洁明了，用矩形，三角形，圆形等简单的图案构成。图标设计都是圆润型的，基本没出现有尖图标的设计，让整个界面看起来十分温和而富有科技感，从而让用户能更加沉浸在音乐世界里。如下图所示：

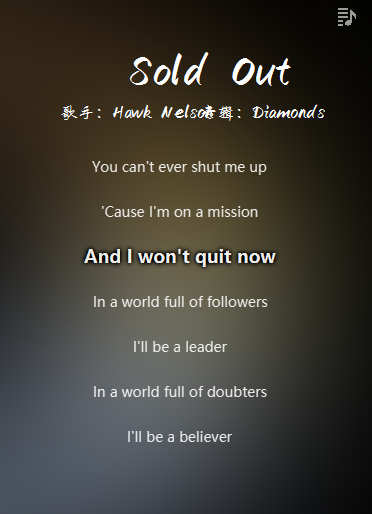


**4.3 歌曲选择界面**



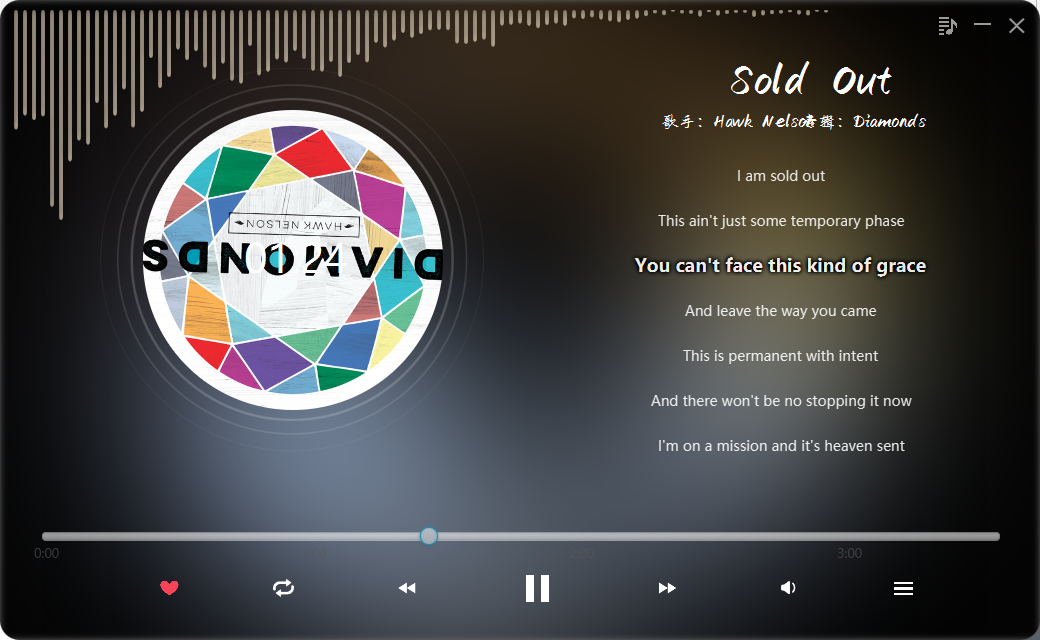
**4.4 歌词播放界面**

歌词是基于tableView的实现，首先读取lrc文件加载歌曲的歌词在List中展示，接着用个定时器每隔0.1秒就自动滚动一次，歌词是同步的，比较当前的时间和播放器的时间的大小来判断是否滚动到下一行，为了使滚动过程更加流畅，加入了动画的效果：将一次滚动分别重复的三次滑动进行执行，每次滑动将滑动1/3的滚动距离，滑动之间的时间控制在0.05s左右。当到达了下一行歌词时，便将该行歌词进行加粗，加阴影后放大展示，如下图所示：



**4.5 频谱展示元素**

跳跃的频谱是具有美感的，尤其是当音乐频率分量规律变化的时候，频谱作为音乐可视化的一种，本界面利用了128个Rectangle元素将音乐的每一个分量独立展示出来，数值越大的分量就具有越宽的长度，同时在每一个监听器的周期内对当前绘制的频谱进行刷新，来达到动态跳跃的效果，这样把频谱的画面美感做到极致，给用户传递画面和音乐的同步。



**4.5 唱片信息元素**

在播放界面可以看到的设计有很多，第一视觉中心是位于核心区域的唱片设计，音乐播放的时候唱片会随音乐匀速旋转(javafx中的Rotation类)，并且在外围设置了动态光晕随着音乐声音律动，这为用户营造了一股浓烈的音乐氛围，提升了使用体验与交互性，如下图所示：



**4.6引导栏元素**

用户引导的设计，作品追求了视觉风格、语言表达与产品定位、产品理念相一致，可以清楚的看到在这个界面，最下面陈列着播放的各种选择，停止播放，下一首，上一首，播放列表，播放顺序等排列得清清楚楚，用户一点就能达到自己的目的，十分方便。然后贴心的爱心功能让用户对歌曲的收藏只需要轻轻一点，爱心变红就把自己心爱的歌曲收藏在了歌单里。页面还隐藏着一些其他功能，像是列表按钮会显示出歌曲列表，将黑胶唱片向左滑向右滑可以直接切歌等功能。



**4.7 进度条Slider元素**

位于底部的进度条能随时间播放显示进度，也展示了歌曲时长等信息，能让用户拖动滑块到指定进度，同时歌词元素也会自动跟进跳转。



1. **可用性评估**

5.1 评估指标体系

1.**系统状态的可见性**：系统应该在合理的时间内做出适当的反馈，始终让用户了解正在发生的事情。

2.**系统与现实世界的匹配**：系统应使用用户的语言，用户熟悉的词语和概念，而不是系统导向的专业术语。遵循现实世界的惯例，使信息以自然和合乎逻辑的顺序出现。

3.**用户控制和自由**：用户有时会误操作，要提供任何时候都能从当前状态跳出来的出口，保证能够及时取消或者再运行执行过的操作（支持撤消和重做）。

4.**一致性和标准化**：不应让用户怀疑不同的词语、情况或行为是否意味着同一件事。保证用户在同样的操作下得到相同的结果。

5.**预防错误**：提前预防错误的发生，这种防患于未然的设计要比适当的错误提示更胜一筹。消除容易出错的条件或检查它们，并在用户采取行动之前让用户再次确认是否进行该操作。

6.**识别而不是回忆**： 通过使对象，动作和选项等可视化，从而最大限度地减少用户的认知负担，使用户无需回忆，一看就懂。尽量不要让用户从当前对话切换到别的对话时还必须记住某些信息，系统的使用说明应该是可见的，或者适当时可以轻易地检索。

7.**灵活性和效率**：加速器功能（初次接触的用户看不到该功能）通常可以提升专家用户的操作效率，从而使系统能够迎合无经验和有经验的用户，允许用户能够单独调整会频繁使用的操作。

8.**审美和极简主义设计**：对话不应包含无关或极少需要的信息，对话中的每条附加信息都会与关键信息形成竞争，并降低其相对可见度。

9.**帮助用户识别，诊断和从错误中恢复**：错误消息应以简单的语言表示，精确地表明问题，并建设性地提出解决方案。

10.**帮助和文档：**即使系统在没有帮助文档的情况下也可以使用良好，但还是有必要提供帮助和文档。这样的信息应该易于搜索，针对用户要执行任务列出具体步骤。

5.2 评估技术

分析法：启发式评估

实验法：一对一用户测试

5.3 评估结果分析

90%的测试者的评价满意度较高，反应了测试任务设计是比较合理的，以简单任务开始，以简单任务结束，使得用户在任务完成时有成就感。10%的测试者的满意度的原因为：软件不支持联网下载歌曲。

由于这款程序本身就是一个简易的中型应用程序，从开发到后期的完善，维护都不需要投入太多的人力、物力，因此实现起来比较容易。对电脑的性能没有过高的要求，系统利用率的占用也较低，目前支持播放电脑内已下载的音乐，还不能联网下载网上的歌曲。作品后期会不断完善、更新，使其更加的人性化。

1. **总结**

6.1 创新点

本系统是采用JavaFX组件完成的一款小巧、界面精美的本地音乐播放器，从用户体验五要素分析了网易云音乐、QQ音乐、酷狗音乐等主流音乐软件进行推陈出新式的学习，在用户体验设计在五个层次都做得较为优秀，方方面面都在为用户着想。当然也支持播放本地的音乐，在本地音乐模块主要采用目前java最主流的音频标记库Jaudiotagger，可解析MP3文件头信息，比如提取内嵌到音频文件内的音乐封面，也可以将音乐封面写入到音频文件内，支持的音频文件主要为MP3，当然也支持wav，包括嵌入和链接图像支持。可读取音频文件头信息里的歌曲名，专辑、歌手名等元信息，此外使用了javafx 第三方ui库jfoenix，toasterfx。其余功能：碟片旋转，歌词滚动特效，歌词同步，最小化精简模式，桌面歌词，切换歌单，下载音乐，最小化系统托盘，少量Material Design风格（依赖jfoenix完成）等。

此外，还采用了一系列UI提高人机交互使用感受，支持窗口区域化拖动、支持（拖拽）添加本地音乐、歌曲以及删除它们、具有歌词lrc文件自动扫描解析、MP3文件自动扫描解析(获取mp3文件中的缩略图，歌手，专辑等信息)、三种播放模式选择(列表循环，单曲循环，随机播放)、歌词动画滚动、歌单列表控制、频谱图展示、频谱图转换为动态光晕显示、歌曲缩略图显示、自定义背景等。

6.2 不足之处

在做写程序实现功能的过程中，界面和基本功能的设计需要较强的逻辑思维，一不小心就容易出错，在开发完成后我明白了在具体的功能设计和实现的过程中要尽可能用最简单的语句实现基本的播放器功能, 所以我自认为的不足之处存在于：我觉得可以设计更多的菜单及菜单项目从而实现更多的功能，如：删除，保存等，还可以实现更多的背景效果或者更多可选择的皮肤，还可以对按钮添加音效效果，这样感觉整个音乐播放器更具有灵活性，媒体感更强烈一些，界面会更加丰富和生动一些。

通过自己做课程设计，自己设计界面，功能和设计功能的代码实现，将理论知识运用到实际操作中，加深了对理论知识的理解和记忆，在做程序设计的过程中，对很多代码的多次操作和重复使用，使得对很多基本的代码和相关的操作能够牢固的记，但发现自己在代码整洁方法还有待提到，书写的代码体系较为混乱，后续开发会消耗较多的时间。

此外，在设计过程中一直追求UI和动画，没有太注意内存优化和模块化大多数对象与监听器采用了复用。 而用java编写的图形界面，而且加了不少动画，内存多多少少不会像C语言那样低，不过在代码编写方式上，已经很注意对象的创建，大量的对象都是复用的，监听器对象也是复用的。经过测试，jdk15为例：最小内存129mb，最大内存达到了380mb，jdk8内存消耗更多，不以这个作为参照。（Jdk15.0.1）or Jdk 1.8\_201（需要更换成jfoenix8.0.8版本），由于使用了动画会出现大量的内存碎片，在运行时的内存占用率一直较高，这也是后续有待优化之处。

**参考文献**

**[1] 郑蕾."一种高效率的软件开发方法——以用户为中心的软件开发方法（UCD）" (2006).**

**[2] 耿凌艳. “浅析人机交互界面“显性”版式设计” (2011).**

**[3] Paul Krill. JavaFX 14 enhances API, mobile support[J]. InfoWorld.com,2020:**

**[4] 刘海芹 and 尹星云.“基于ActionScript 3.0音乐播放器的设计与实现” (2011).**

**[5] 叶佳骏 et al. “基于“云”技术的音乐播放器设计” (2012).**

**[6] Jim Clarke et al. “Javafx: Developing Rich Internet Applications” (2009).**

**[7] Kishori Sharan. “Understanding Concurrency in JavaFX” (2015).**

**[8] 余湉 and 付强. “面向用户体验的软件界面设计——以Pro/Engineer界面为例” (2015).**

**[9] 张亮. “从Windows XP登录界面说起——谈人机界面设计中的模式现象” (2005).**