



教师：王晓东

评分：

Java 应用与开发课程

音乐播放器项目设计说明书

姓 名	任浩辰	刘阳
学 号	18180021016	18020031052
专 业	教育技术学	计算机科学与技术
日 期	2019-12-19	

中国海洋大学

信息科学与工程学院计算机系

目录

目录.....	I
1. 系统概述.....	- 1 -
2. 总体设计.....	- 4 -
3. 模块设计.....	- 5 -
3.1. 模块命名规则.....	- 5 -
3.2. 模块汇总	- 5 -
3.3. 子系统 A 的模块设计	- 6 -
子系统 B 的模块设计	- 9 -
4. 数据库设计	- 26 -
4.1. 数据库环境说明.....	- 26 -
4.2. 逻辑设计	- 26 -
4.3. 物理设计	- 28 -
5. 用户界面设计	- 26 -
5.1. 界面的关系图和工作流程图.....	- 30 -
6. 附录：用户手册.....	- 33 -

1. 系统概述

本应用是基于 Java Swing、JavaFX 开发的音乐播放器，功能要点总结如下：

1. 本地：

- 新建本地歌单，可以自行命名
- 添加、删除、播放本地歌曲
- 上传歌单
- 删除本地歌单

2. 推荐歌单

- 查看其他用户上传的歌单，包含歌单名，创建者，歌单图片等信息
- 查看歌单内的歌曲

3. 下载歌曲

- 下载其他用户上传的歌单内的歌曲

4. 播放音乐

- 进度条实时显示播放进度
- 拖动进度条 实现断点播放
- 播放/停止/上一首/下一首按钮 实现歌曲播放状态切换
- 顺序播放/单曲循环/随机播放 三种播放模式切换
- 音量控制栏 可以调整音量大小

2. 需求分析（需求规格说明）

1) 新建歌单

点击新建歌单按钮，在弹出的输入框中输入自定义歌单名，即可创建歌单

2) 添加歌曲

在歌单内点击添加歌曲按钮，在弹出的文件选择框中多选/单选想要添加的歌曲，即可将歌曲添加到歌单中

3) 分享歌单

在歌单内点击分享歌单按钮，将歌单分享到服务器

4) 删除歌单

在歌单内点击删除歌单按钮，删除此歌单

5) 更换歌单封面

在歌单内点击修改封面，更换歌单的封面

6) 播放歌曲：

a) 点击歌曲即可播放

b) 音乐断点播放：拖拽进度条，将其滑动至相应位置时，歌曲进度跳转到当前进度

c) 点击播放/暂停按钮，可以使当前音乐开始播放/暂停播放

d) 点击上一首/下一首按钮，实现歌曲切换

e) 点击切换播放模式 顺序播放/单曲循环/随机播放

f) 拖动滑动条，设置音量大小

7) 删除歌曲

歌单中的歌曲->单击右键，点击删除按钮即可从歌单中删除歌曲

8) 别人都在听——浏览其他用户上传的歌单

- a) 点击 别人都在听 栏目，显示其他用户上传的所有歌单，及其歌单图片，歌单名，用户名
- b) 点击歌单，进入此歌单界面，可以查看歌单中的所有歌曲
- c) 右键单击歌曲，点击下载按钮实现下载

9) 下载

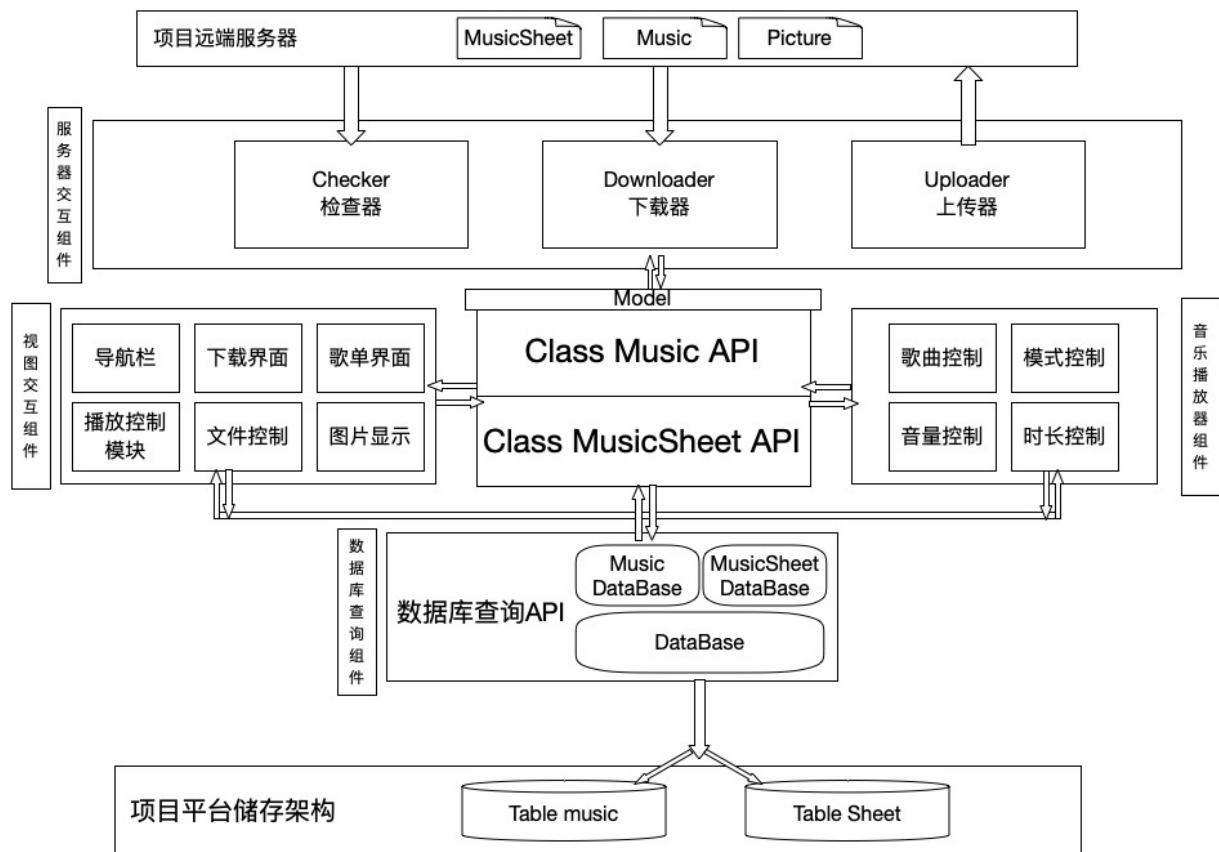
- a) 没有歌曲下载时，提示空白页



- b) 下载歌曲：提示 正在下载/已完成 两种下载状态

3. 总体架构设计

项目主要分为模型 API、视图交互组件、音乐播放组件、服务器交互组件以及数据库查询组件，配合轻量级 SQLite 数据库储存本地歌单和歌曲的基本信息，远端服务器进行上传歌单歌曲和下载歌曲。



4. 模块设计

4.1. 模块命名规则

4.2. 模块汇总

子系统- awt 界面控件	
模块名称	功能简述
滑动条 MSlider	可拖拽歌曲播放进度条、音量大小调节滑动条
MSliderUI	重写 JSlider 的 UI，有美化作用
MScrollPane	封装的滑动面板
MScrollPaneUI	重写 JScrollPaneUI，美化作用
各类 Panel	如：显示歌曲信息，显示推荐歌单信息，左侧栏栏目按钮等
子系统- db 数据库操作类	
模块名称	功能简述
DataBase	数据库连接
MusicDataBase	Music 类的数据库操作
MusicSheetDataBase	MusicSheet 类的数据库操作
子系统- model 基本类	
模块名称	功能简述
Music	歌曲类
MusicSheet	歌单类
Player	音乐播放控制器
子系统- util 联网工具类	
模块名称	功能简述

Checker	获取推荐歌单及相应内容
Downloader	下载音乐文件，并获取下载是否完成状态
MD5Utils	解析 md5 字符串
Uploader	上传歌单到服务器
子系统- view 主界面	
模块名称	功能简述
CenterDownloadView	下载界面
CenterListView	歌单界面
CenterOthersView	别人都在听 歌单列表
MainView	主界面窗口
SouthView	底部播放控制按钮、进度条等
WestView	左侧按钮/栏目

4.3. 子系统- awt 界面控件的模块设计

4.3.1. 模块

模块名称	滑动条 MSlider
功能描述	可拖拽歌曲播放进度条
接口 与属性	继承自 JSlider 实现 MouseListener 接口 int currentProgress//记录当前播放进度 int MAX_PROGRESS//进度最大值 Timer timer//计时器，实现进度自动增加

	<p>接口 mouseReleased : 鼠标松开后，歌曲进度跳转到当前位置</p> <p>public void startPlaying();//timer 开始计时，滑动条开始滑动</p> <p>public void animate();//初始化计时器</p> <p>public void setMax();//设置当前播放时长</p> <p>public void setCurrentProgressToZero();//将播放进度设置为 0</p>
数据结构 与算法	<p>点击歌曲后调用 startPlaying 方法，从播放控制器 player 获取当前歌曲进度，并赋值给 currentProgress，timer 每隔 100 毫秒如此读一次数据，并将进度条进度实时更新</p>
补充说明	<p>该类利用 MSliderUI 重写了 JSliderUI，实现美化效果</p>

模块名称	MSliderUI
功能描述	重写 SliderUI，起美化作用
接口与属性	继承自 BasicSliderUI，重写父类的 paintThumb,paintTrack 方法
数据结构 算法	<p>通过 drawImage 方法，将选定的 thumb 图片把原生 JSlider 的 thumb 图片替换掉；paintTrack 将滑块划过的地方变成红色</p>
补充说明	

模块名称	MScrollPane
功能描述	继承自 JScrollPane，滑动面板用于显示多首歌曲、歌单等

《音乐播放器》

接口与属性	重写父类 ui，美化作用
数据结构 算法	使用 setUI 方法，改变 Ui
补充说明	

模块名称	MScrollPaneUI
功能描述	重写 MScrollPaneUI，起美化作用
接口与属性	继承自 BasicScrollBarUI 类，重写父类的方法，其中包括： configureScrollBarColors,getPreferredSize,paintTrack,paintThumb,createIncreaseButton,createDecreaseButton
数据结构 算法	将原生的 ScrollPane 进行大量重写，重新绘制滑块路径，重新绘制滑块图像和按钮图像，改变整体的 UI
补充说明	原生 JScrollPane 的样式与软件 UI 整体风格相差太大，故进行重写

模块名称	DownloadPanel,SongPanel 等
功能描述	仿照安卓 app 中，Recyclerview 的 item 写法，简化代码
接口与属性	继承自 JPanel，向 panel 中添加自己想要显示的组件 实现 MouseListener 接口，重写点击事件
数据结构 算法	自己封装的各类 panel，与 music,musicsheet 相对应，可以用来显示信息

补充说明	简化代码
------	------

子系统- db 数据库操作类设计

4.3.2. 模块

模块名称	DataBase
功能描述	连接数据库
接口 与属性	<p>Connection connection;//连接数据库</p> <p>Statement statement;//SQL 语句执行器</p> <p>public void close();//结束程序时调用</p> <p>private static void connectDataBase();连接数据库</p> <p>private static void makePreparations();创建 music 和 musicsheet 表</p>
数据结构 与算法	
补充说明	

模块名称	MusicDataBase
功能描述	对 music 类在数据库中的操作的封装
接口 与属性	<p>public static ArrayList<Music> getMusicsFromMusicSheet</p> <p>(Music sheet);//返回歌曲列表，获取歌单中的所有歌曲</p>

《音乐播放器》

	<pre> public static boolean deleteMusicFromMusicSheet(Music music);//从歌单中删除歌曲，删除成功返回 true，失败则返回 false public static boolean insertMusicIntoMusicSheet(Music music,MusicSheet musicSheet);//向歌单中添加歌曲，添加成功返回 true，失败则返回 false public static void musicTablePreparation();//数据库新建音乐表 </pre>
数据结构 与算法	
补充说明	

模块名称	MusicSheetDataBase
功能描述	对 musicsheet 类在数据库中的操作的封装
接口 与属性	<pre> public static MusicSheet getMusicSheet(int id);//根据指定 id 获取 歌单 public static ArrayList<MusicSheet> getAllMusicSheets ();//获取 所有歌单 public static boolean deleteMusicSheet(MusicSheet musicSheet);//删除歌单，删除成功返回 true，失败则返回 false public static boolean insertMusicSheet(MusicSheet musicSheet);//创建歌单，创建成功返回 true，失败则返回 false public static void musicSheetTablePreparation();//数据库中新建 </pre>

	<p>歌单的表</p> <pre>public static boolean updateMusicSheetPicture(MusicSheet musicSheet,String picturePath);//更新歌单封面，更新成功返回true，失败则返回 false</pre>
数据结构 与算法	
补充说明	

子系统- model 基本类设计

4.3.3. 模块

模块名称	Music
功能描述	歌曲类
接口 与属性	<p>//主属性</p> <p>String name;</p> <p>int sheetId;</p> <p>String uuid;</p> <p>String path;</p> <p>//其他属性</p> <p>MusicSheet sheet;//属于的歌单</p> <p>File file;//文件</p>

	//检测属性 boolean isLoading;//正在加载状态 boolean canPlay;//是否可以播放 //前端显示属性 String singer;//歌手名 String album;//专辑名 double duration;//时长 boolean isDownloading;//是否正在下载 public static ArrayList<Music> newMusic(File[] files);//新建歌曲 public void loadMusic();//加载音乐 private String generateMD5();//生成歌曲的 uuid //各类属性的 get set 方法略
数据结构 与算法	
补充说明	

模块名称	MusicSheet
功能描述	歌单类
接口 与属性	int id;//歌单 id String name;//歌单名

	String dateCreated;//创建时间 String creator;//创建者 String creatorId;//创建者 id String picture;//歌单图片 String uuid; ArrayList<Music> musicArray;//歌单中的音乐 public static MusicSheet getDownloadedMusicSheet();//获取下载页歌单 public static ArrayList<MusicSheet> getAllMusicSheet();//获取所有歌单 public static MusicSheet getAllMusicSheet(int id);//根据 id 获取歌单 public static ArrayList<Music> getMusicFromMusicSheet ();//获取当前歌单的所有歌曲 public void saveMusicSheetInDatabase();//保存当前歌单到数据库 public void deleteMusicSheet();//删除当前歌单 public void deleteMusicFromMusicSheet(int index);//删除当前歌单中指定索引的歌曲 addMusicIntoMusicSheet(Music music);//向当前歌单中添加歌曲 public void addMusicIntoMusicSheet(Music[] musics) ; //添加多首歌曲
--	--

《音乐播放器》

	<pre>public void uploadMusicSheet() ; //上传当前歌单到服务器 public void updateMusicSheetPicture(File file) ; //更新歌单封面</pre>
数据结构 与算法	
补充说明	

模块名称	Player
功能描述	音乐播放控制器
接口 与属性	<pre>private Media media = null; private MediaPlayer mediaPlayer = null; private double volume = 0.7; // default 0.7 0.0 (静音) ~ 1.0 (最大音量) private double duration = Duration.seconds(0).toSeconds();//歌曲时长 private enum Mode {ORDER, RANDOM, SINGLE} //三种播放模式 private Mode mode = Mode.ORDER; //默认播放模式 private MusicSheet musicSheet = null; //当前歌单 private int index = 0; //当前歌曲索引</pre>


```
public static Player getInstance() ; //获取 player

public void selectMusicSheetAndMusic(MusicSheet sheet, int
index) ; //选择歌曲、歌单进行播放

public void selectMusic(int index) ; //选择歌曲进行播放

/**
 * 获取当前选择歌单中的所有歌曲
 * @return 歌曲列表
 */

public ArrayList<Music> getCurrentMusicList() ;

/**
 * 获取当前的歌单
 * @return 歌单 MusicSheet
 */

public MusicSheet getCurrentMusicSheet() ;

/**
 * 获取当前播放的歌曲
 * @return 歌曲
 */

public Music getCurrentMusic() ;

/**
 * 获取当前正在播放歌曲在歌单中的索引位置
 * @return 索引
```

```
*/  
  
public int getIndex() ;  
  
/**  
  
 * 从头播放当前歌曲  
  
*/  
  
public void replay() ;  
  
/**  
  
 * 暂停音乐  
  
*/  
  
public void pause() ;  
  
/**  
  
 * 清空当前播放器  
  
*/  
  
public void dispose() ;  
  
/**  
  
 * 当前是否在播放音乐  
  
 * @return true    Playing  
  
 *           false   No selected music  
  
 *           Stop  
  
*/  
  
public boolean isPlaying() ;  
  
/**
```

```
* 获取当前歌曲 music 的时间 (秒)

* @return 秒数 (s)

*/

public double getCurrentTime() ;

/**

* 获取当前歌曲 music 在总时间的比例

* @return 比例数 0.0 ~ 1.0

*/

public double getCurrentScaleOfDuration() ;

/**

* 当前歌曲跳到指定的秒数 (s)

* 若 当前歌曲 是播放状态, 跳跃后会继续播放

* 若 当前歌曲 是暂停状态, 跳跃后依然暂停

* @param second 要跳的秒数 (s)

*/

public void jumpTo(int second) ;

/**

* 当前歌曲跳到指定的比例

* @param scale 比例数 0.0 ~ 1.0

*/

public void jumpToTheScaleOfDuration(double scale) ;

/**
```

	<pre> * 切换到下一首歌曲 */ public void next() ; Get set 方法略 </pre>
数据结构 与算法	
补充说明	使用单例模式，单例模式会阻止其他对象实例化其自己的单例对象的副本，从而确保所有对象都访问唯一实例。

子系统- util 联网工具类设计

4.3.4. 模块

模块名称	Checker
功能描述	获取网络歌单
接口 与属性	<pre> Static String <i>MUSIC_SHEET_RUL</i> ; //获取歌单的 url private static ArrayList<MusicSheet> <i>arrayList</i> = null; //获取到的歌单 public static ArrayList<MusicSheet> getMusicSheetList() ; //获取歌单 public String getSheetResponseJson() ; //获取返回的 json 数据 </pre>
数据结构 与算法	

补充说明	
------	--

模块名称	Downloader
功能描述	下载歌曲
接口 与属性	<pre> /** * 下载歌曲到 MusicDownload 文件夹 * @param music 歌曲 Music */ public static void downloadMusic(Music music) ; /** * 下载歌曲到 MusicDownload 文件夹 * @param musicSheet 歌单 MusicSheet */ public static void downloadMusicSheetPicture(MusicSheet sheet) ; /** * 下载 * @param url URL * @param destFileDir 文件夹 * @param listener Listener */ </pre>

	public static void download(final String url, final String destFileDir, final Downloader.OnDownloadListener listener) ;
数据结构 与算法	
补充说明	

模块名称	MD5Utils
功能描述	生成 md5 的工具类
接口 与属性	<pre> /** * MD5 加密 * @param origin 字符 * @param charset 编码 * @return MD5 string */ public static String MD5Encode(String origin, String charset); /** * * @param file 文件 </pre>

	<pre> * @return MD5 string */ public static String MD5Encode(File file); </pre>
数据结构 与算法	
补充说明	

模块名称	Uploader
功能描述	上传歌单的工具类
接口 与属性	<pre> /** * 上传歌单以及歌单中歌曲 * @param sheet 歌单 MusicSheet */ public static void uploadMusicSheet(MusicSheet sheet) ; </pre>
数据结构 与算法	
补充说明	

子系统- view 主界面设计

4.3.5. 模块

模块名称	CenterDownloadView
------	--------------------

功能描述	中央，下载界面
接口 与属性	<pre>private int WIDTH = 1200;//宽度</pre> <pre>private int HEIGHT = 820;//高度</pre> <pre>public static boolean <i>isDownloading</i> = false;//是否正在下载</pre> <pre>public static ArrayList<Music> <i>music</i> = new ArrayList<>();//下载</pre> <p>的音乐列表</p> <pre>private static ArrayList<DownloadPanel> <i>downloadPanels</i> =</pre> <pre>new ArrayList<>();//歌曲的一条 item 面板</pre> <pre>private void initBlankView(); //如果没有下载音乐，显示空白页，</pre> <p>并给用户提示</p> <pre>private void initDownloadView(); //初始化界面</pre> <pre>public static void setStatus(Music music); //改变 Music 的下载状</pre> <p>态</p>
数据结构 与算法	继承自 JPanel
补充说明	设计了空白页提示图，在没有下载音乐时给用户提示，提升用户体验

模块名称	CenterListView
功能描述	歌单界面
接口	private JLabel lb_list_pic;//歌单图片

与属性	<pre> private JPanel mPanel = new JPanel();//根布局 panel private int WIDTH = 1200;//宽度 private int HEIGHT = 820;//高度 private String listName;//歌单名 private String userName;//用户名 private JButton btn_addMusic = new JButton("添加歌曲");//添加歌曲按钮 private JButton btn_delete = new JButton("删除歌单");//删除歌单按钮 private JButton btn_upload = new JButton("分享歌单");//分享歌单按钮 private boolean isMyList = false;//是否是本地歌单 private MusicSheet musicSheet;//显示的歌单数据 private void initBtn() ; //初始化按钮，给按钮添加点击事件 private static void showFileAddDialog(Component parent,MusicSheet musicSheet) ; //显示添加歌曲文件的 dialog </pre>
数据结构与算法	继承自 JPanel
补充说明	

模块名称	SouthView
------	-----------

功能描述	播放控制条
接口 与属性	<pre> static JButton btn_play = new JButton();//播放/暂停按钮 static JButton btn_next = new JButton();//下一首按钮 static JButton btn_front = new JButton();//上一首按钮 public static MSlider mSlider = new MSlider();//播放进度条 static boolean isPlaying = false;//是否正在播放 private JLabel btn_playStyle = new JLabel();//播放模式按钮 private JLabel btn_vol;//音量按钮 private static int playStyle = 0;//0 为列表循环，1 为单曲循环，2 为 随机播放 public static boolean haveMusic = false;//是否有歌曲 /** * 改变状态有/无音乐 * @param haveMusic */ public static void changeStatus(boolean haveMusic) ; /** * 点击音乐时执行 startPlay */ public static void startPlay() ; /** * 初始化组件 </pre>

《音乐播放器》

	<pre>*/ private void initComponents() ; public static void showPopupMenu(Component invoker, int x, int y) ; //弹出音量控制的滑动条</pre>
数据结构 与算法	
补充说明	继承自 JPanel

模块名称	WestView
功能描述	左侧栏目
接口 与属性	<pre>int WIDTH = 250;//宽度 ArrayList<MLabel> mLabels = new ArrayList<>();//按钮 ArrayList<MusicSheet> musicSheets = new ArrayList<>();//本地 歌单列表</pre>
数据结构 与算法	
补充说明	继承自 JPanel

5. 数据库设计

5.1. 数据库环境说明

数据库：SQLite

连接数据库 jar 包：sqlite-jdbc-3.27.2.jar

设计工具：OmniGraffle

数据库工具：DataGrip、Navicat

编程工具：IntelliJ IDEA

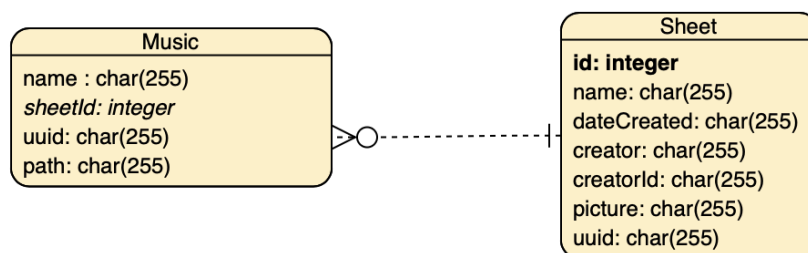
测试连接工具：IntelliJ IDEA 内置 DataBase 工具，配合 sqlite-jdbc.jar

本项目数据库文件名：Music.db













本项目数据库所含表名：music、sheet

本项目数据库编码格式：UTF-8










5.2. 逻辑设计



Music

 name	String
 sheet	MusicSheet
 file	File
 duration	double
 loading	boolean
 album	String
 uuid	String
 singer	String
 sheetId	int
 path	String
 canPlay	boolean
 downloading	boolean

MusicSheet

 name	String
 id	int
 picture	String
 uuid	String
 musicsFromMusicSheet	ArrayList<Music>
 creatorId	String
 musicArray	ArrayList<Music>
 dateCreated	String
 creator	String

Powered by yFiles

Powered by yFiles

数据库 music 表与 Music 类属性对应关系

name → Music.name

sheetId → Music.sheet → MusicSheet.id

uuid → Music.uuid

path → Music.path

数据库 sheet 表与 MusicSheet 类属性对应关系

id → MusicSheet.id

name → MusicSheet.name

dateCreated → MusicSheet.dateCreated

creator → MusicSheet.creator

creatorId → MusicSheet.creatorId

picture → MusicSheet.picture

uuid → MusicSheet.uuid

5.3. 物理设计

5.3.1. 表汇总

表名	功能说明
music	存储歌曲信息的表单
sheet	存储歌单信息的表单

5.3.2. 表 music

表名	music		
列名	数据类型（精确范围）	空/非空	约束条件
name	CHAR(255)	空	Music.name
sheetId	INT	空	Music.sheet
uuid	CHAR(255)	空	Music.uuid
path	CHAR(255)	空	Music.path
补充说明	<ul style="list-style-type: none">● name 歌曲名● sheetId 歌曲所属歌单 ID（外键）● uuid 歌曲的 MD5 值● path 歌曲所在的文件路径		

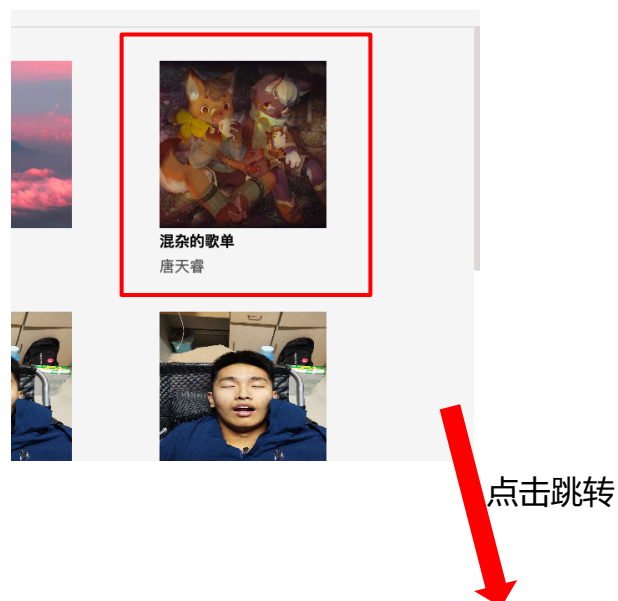
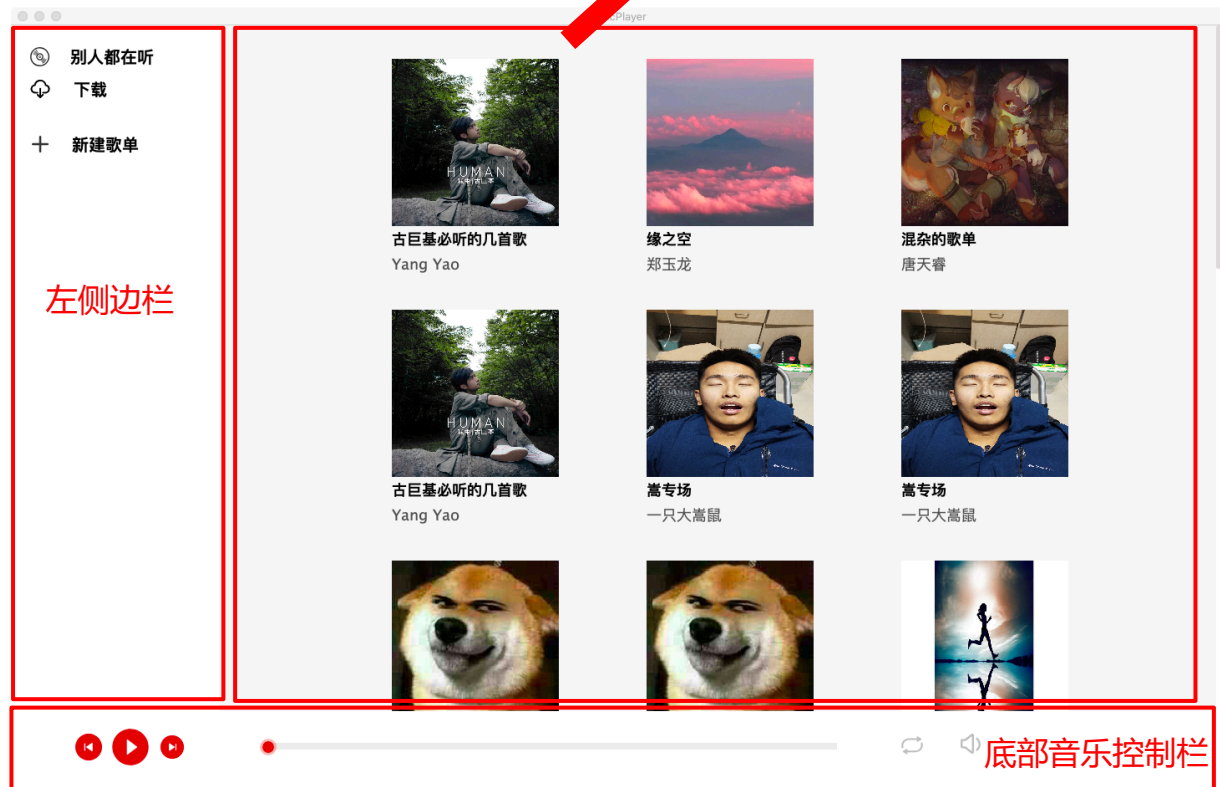
5.3.3. 表 sheet

表名	sheet		
列名	数据类型（精确范围）	空/非空	约束条件
id	INTEGER PRIMARY KEY	非空	自增主键
name	CHAR(255)	空	MusicSheet.name
dateCreated	TIMESTAMP	空	MusicSheet.dateCreated
creator	CHAR(255)	空	MusicSheet.creator
creatorId	CHAR(255)	空	MusicSheet.creatorId
picture	CHAR(255)	空	MusicSheet.picture
uuid	CHAR(255)	空	MusicSheet.uuid
补充说明	<ul style="list-style-type: none"> ● id 歌单的自增 ID ● name 歌单名 ● dateCreated 歌单的创建时间 ● creator 歌单的创建者名字 ● creatorId 歌单的创建者 ID ● picture 歌单封面的图片的文件路径 ● uuid 歌单的 MD5 值 		

6. 用户界面设计

6.1. 界面的关系图和工作流程图

主界面



《音乐播放器》

点击跳转到下载界面



《音乐播放器》

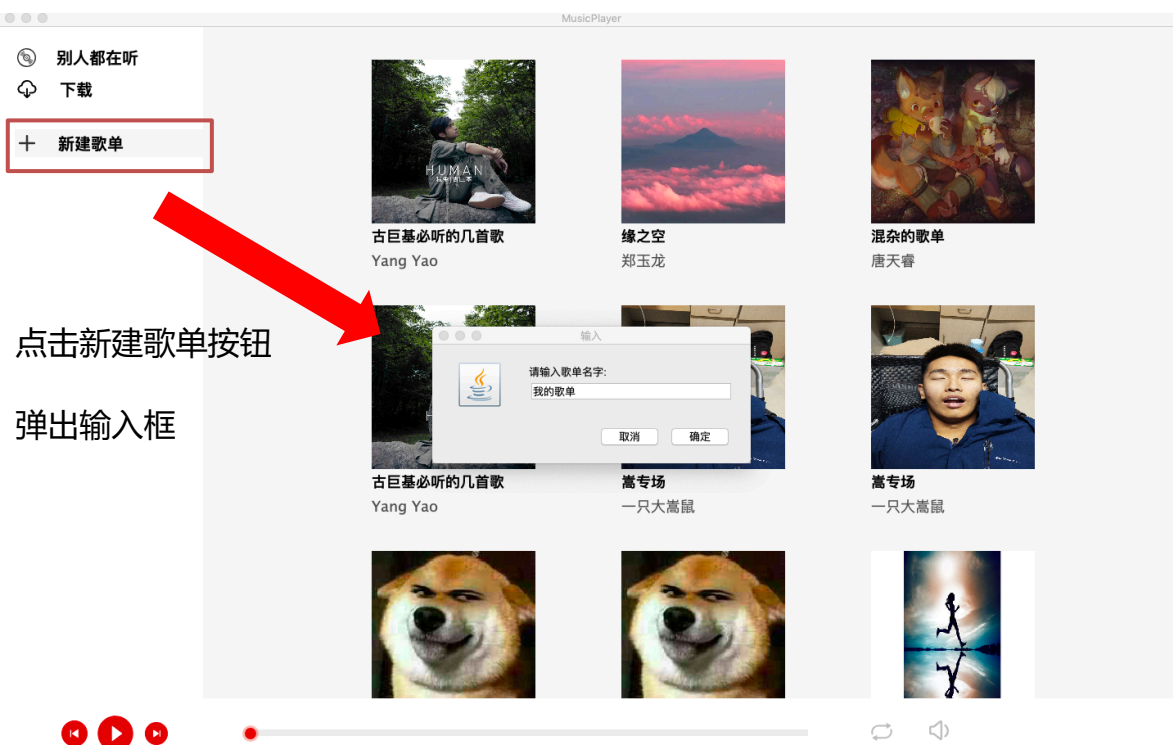
别人都在听
下载
+ 新建歌单
HelloWorld

下载管理

许嵩-弹指一挥间.mp3

已完成

下载页 (有数据时)



本地创建的歌单



我的歌单

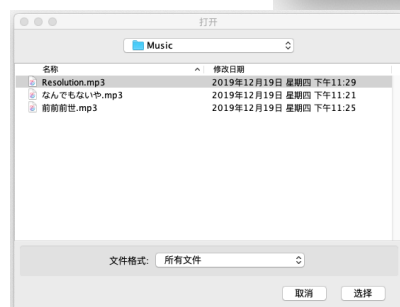
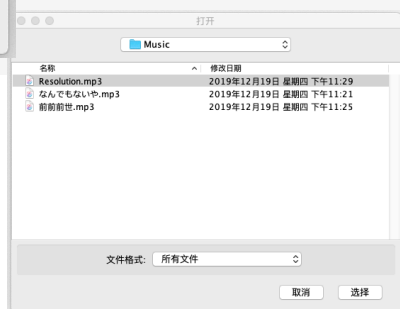
创建者: 粉色龙类物种

分享歌单

添加歌曲

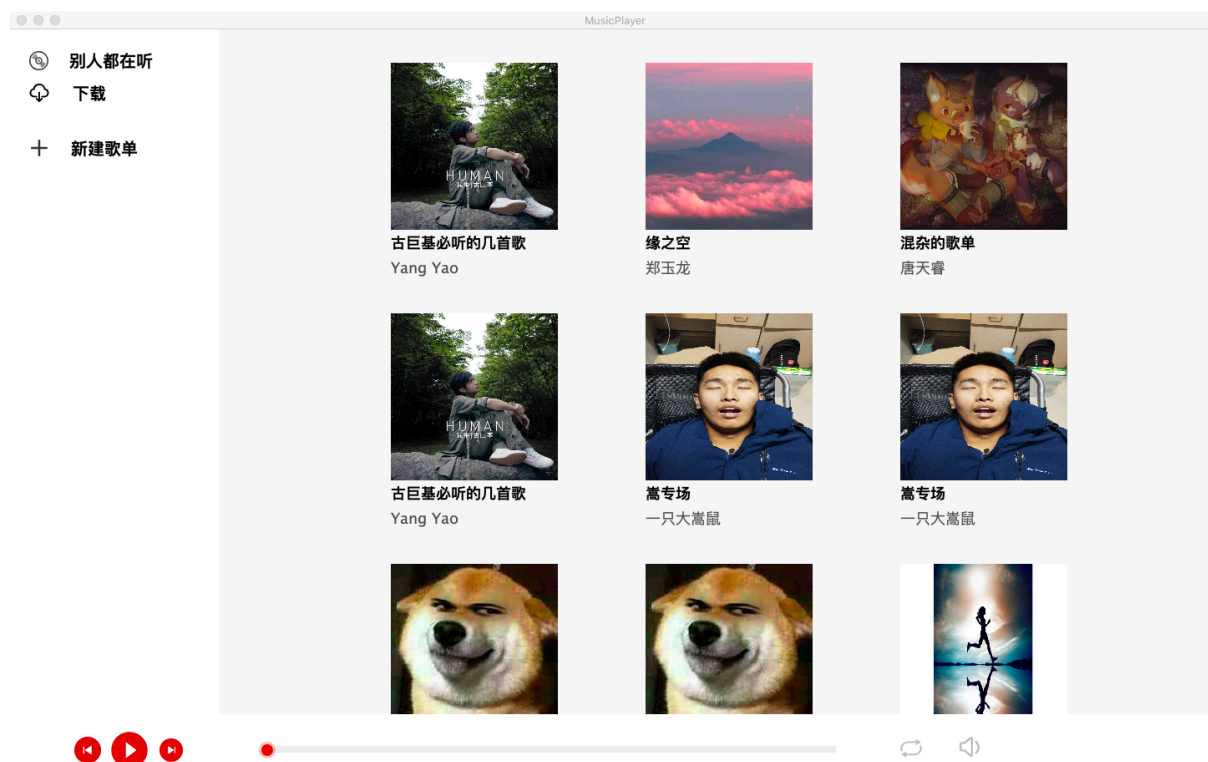
修改封面

删除歌单



7. 附录：用户手册

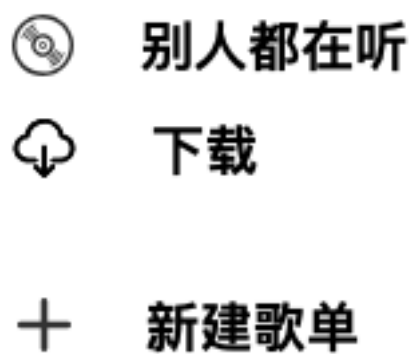
1. 主界面



主界面显示左侧导航栏、下方的播放模块以及中部的网络歌单显示区。

打开音乐播放器默认进入别人都在听模块，显示网络歌单的相关信息（仅获取网络歌单的 TOP20 ）。

2. 导航栏



导航栏显示两个固定模块别人都在听和下载。

导航栏选择查看模块的详细信息。

下部分为用户的本地歌单列表，并可以创建歌单。

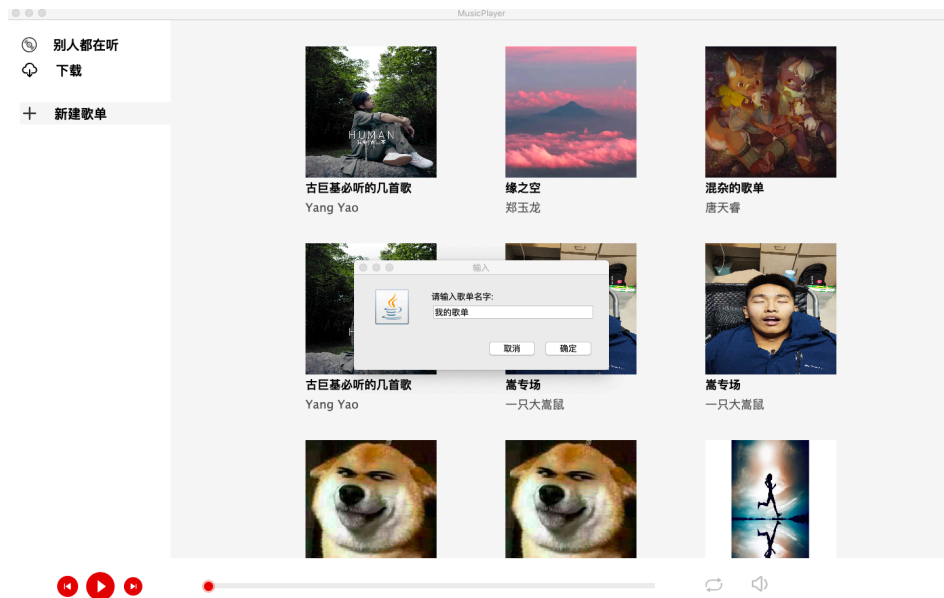
3. 创建歌单

点击新建歌单



会弹出对话框，提示输入歌单名字（默认名字为我的歌单）。

《音乐播放器》



在此输入歌单名 (如 HelloWorld) , 点击确认。



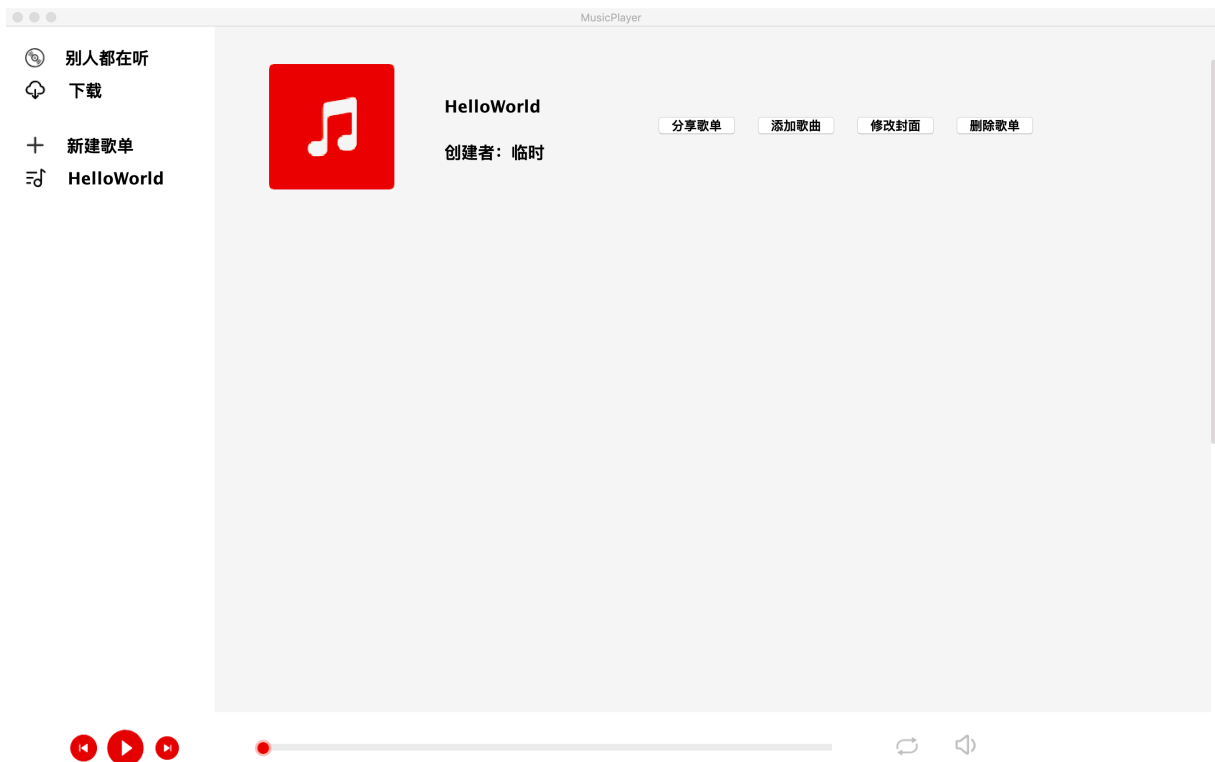
左侧导航栏就会出现刚刚创建的名为 HelloWorld 的歌单。



可以创建多个歌单。



4. 歌单界面



歌单界面显示左侧导航栏、下部的播放模块以及歌单模块。

在歌单模块的上部显示歌单的名字以及创建者，并有四个按钮分享歌单、添加歌曲、修改封面和删除歌单。

5. 删除歌单

点击删除歌单按钮。



弹出对话框，点击是。



歌单就会被移除。



6. 添加歌曲

添加歌曲到特定的歌单。

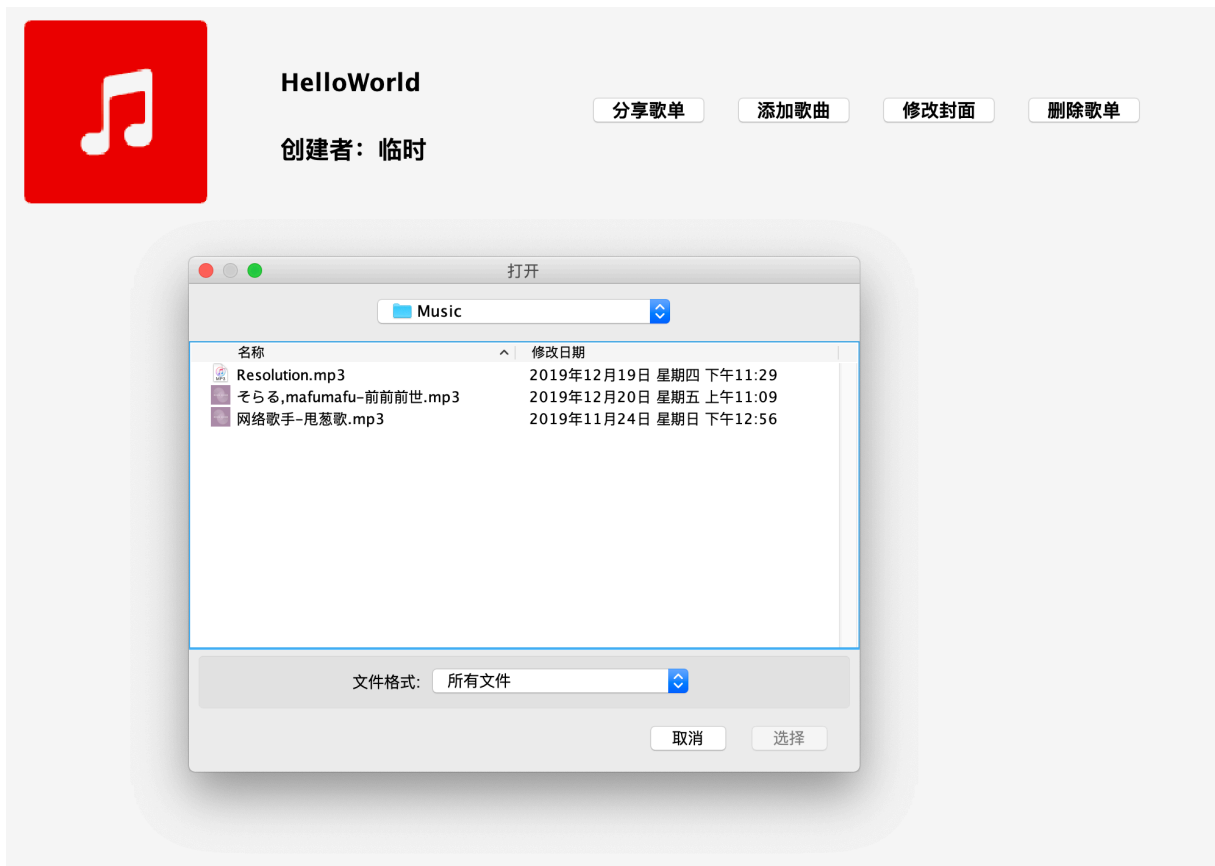
选择指定的歌单（如我们刚才创建的 HelloWorld）。

点击歌单模块上部的添加歌曲按钮。

《音乐播放器》



弹出选择框，找到想要添加的歌曲所在的目录，选择歌曲并点击选择。



歌曲会顺利添加到歌单中，并显示歌曲的相关信息歌曲名、歌手、专辑名和歌曲的时长（信息从歌曲 mp3 文件中读取）。



可以向歌单中添加多个歌曲。



在歌曲 mp3 内信息不全时会显示空（如上图中，有些歌曲的专辑名为空）。

7. 删除歌曲

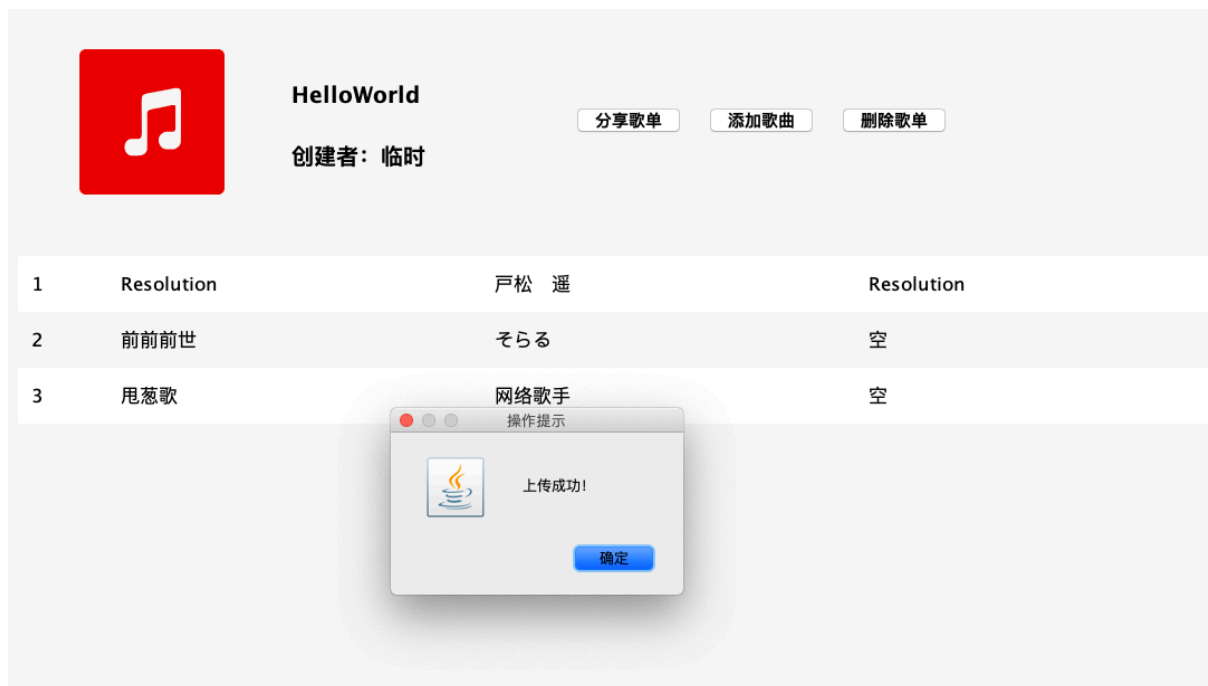


将光标放到要删除的歌曲，点击右键会出现删除选项，点击删除，歌曲就会从歌单中移除。



8. 分享歌单

点击分享歌单按钮，会将歌单上传到远端服务器，并携带歌曲文件以及歌单的相关信息。



提示上传成功，在服务器的歌单中就会有显示。

9. 下载界面



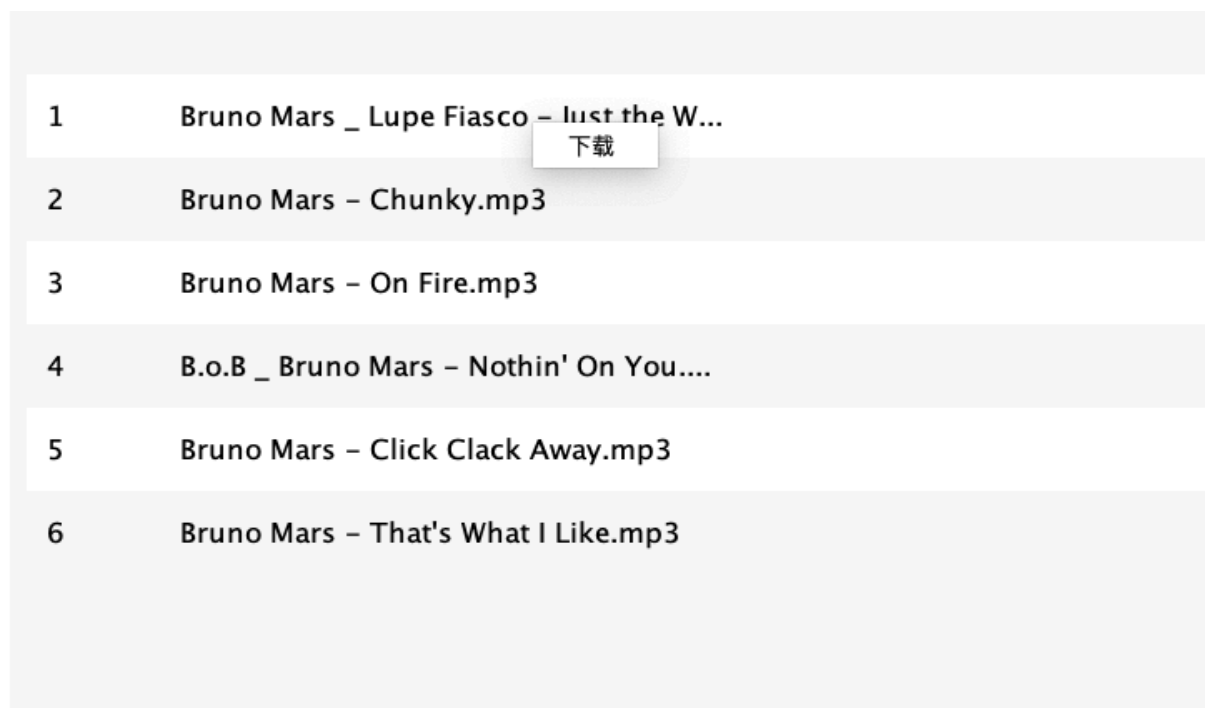
下载界面显示左侧导航栏、下方的播放模块和中部的下载管理模块。

可以从网上下载歌曲。

10. 下载歌曲

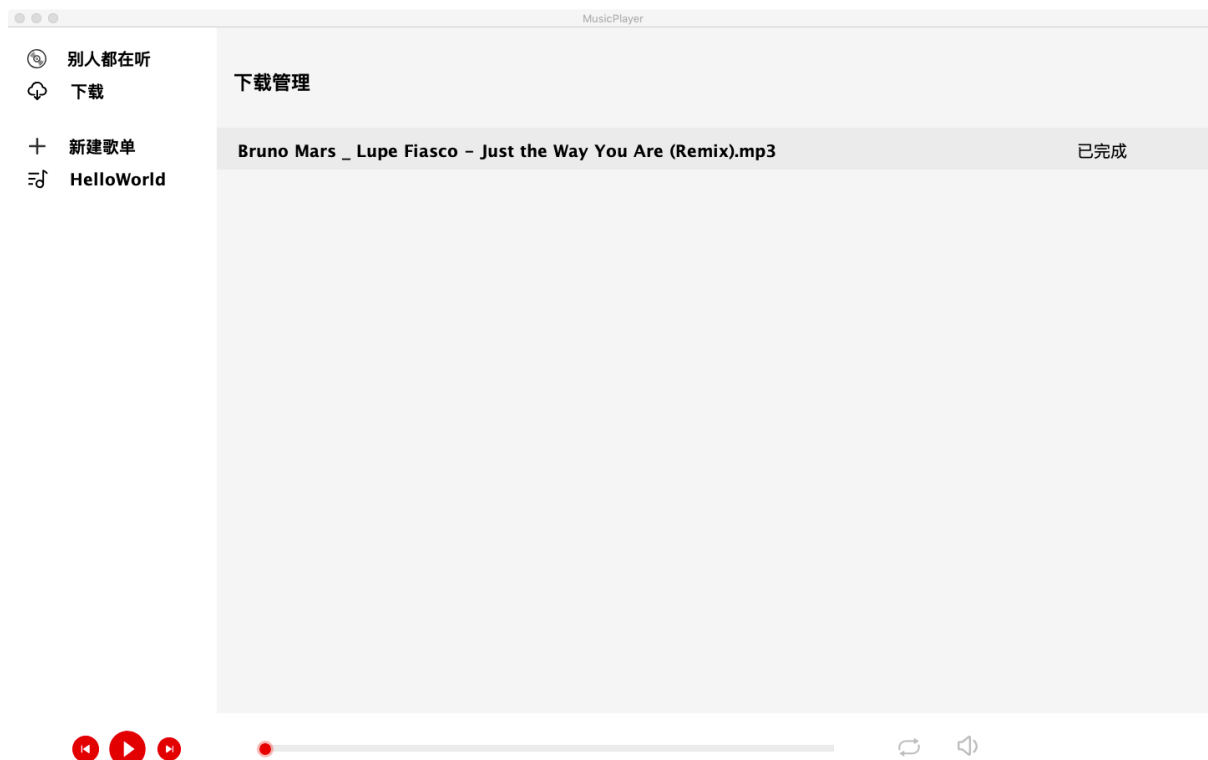
选择一个网络歌单。

右键要下载的歌曲，点击下载，即可下载歌曲。



在下载界面会进行下载进度的显示，以及显示下载过的歌曲。

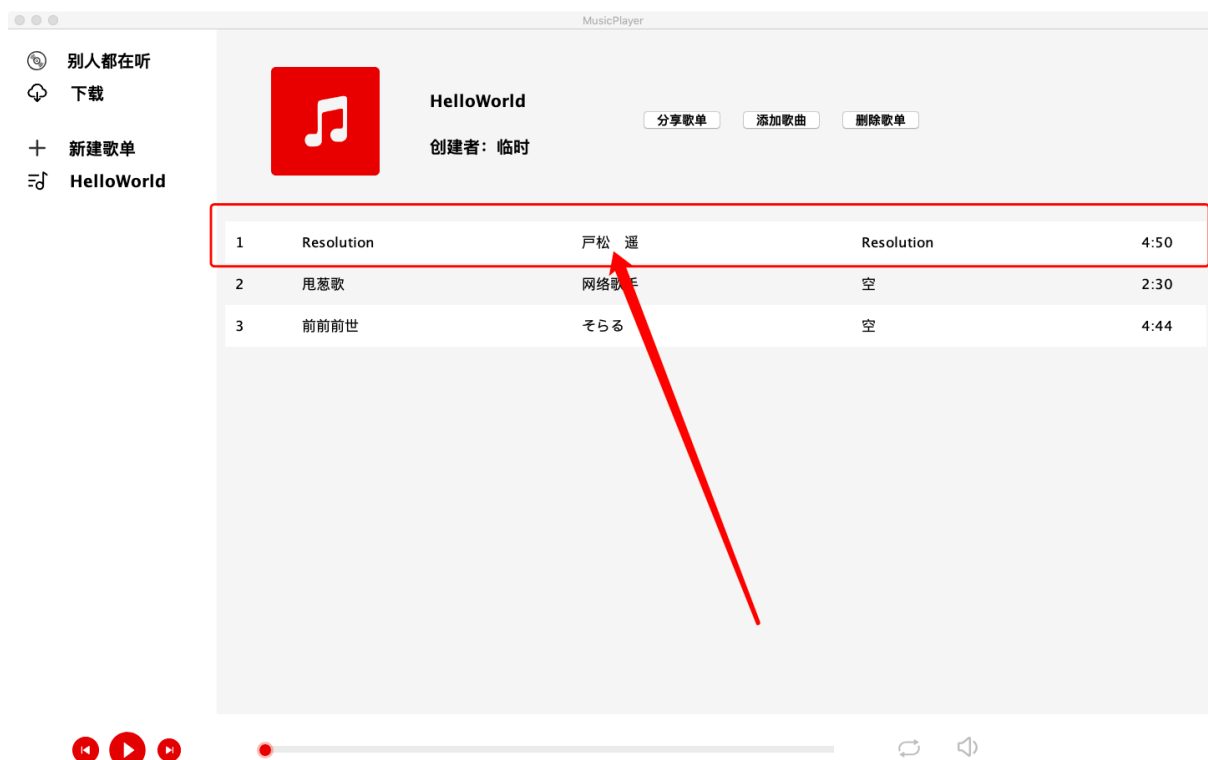
《音乐播放器》



可以对已下载的歌曲进行播放。

11. 播放歌曲

双击本地歌单中已添加的歌曲，进行播放。

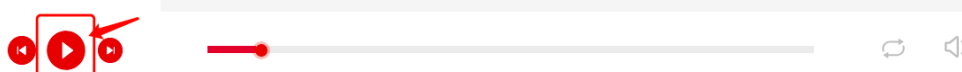


下方的滑动条会跟随歌曲进行移动。



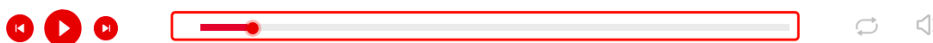
12. 暂停/播放

点击下图所示按钮可以停止播放当前歌曲或者继续播放当前歌曲。



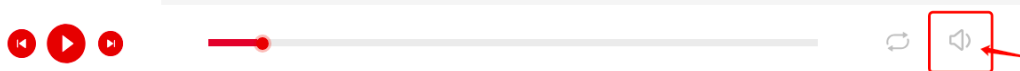
13. 选择指定播放位置

拖动滑动条到指定位置，歌曲就会跳到指定位置进行播放。

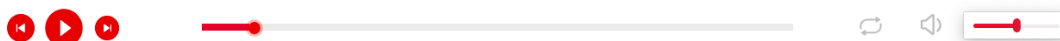


14. 调节音量

点击喇叭图标。



会出现音量调节滑动条，向左调节减小音量，向右调节增大音量。



15. 选择播放模式

下图所示位置，改变当前的播放模式。



以上为顺序播放、单曲循环和随机播放。

顺序播放为按照当前歌单中歌曲的顺序进行播放。

单曲循环为一直播放当前歌曲。

随机播放为随机选取当前歌单中歌曲进行播放。

16. 上一首/下一首



点击左侧按钮为播放上一首歌曲。

点击右侧按钮为播放下一首歌曲。