

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Screen 3](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any edge or corner cases in the UX.](#)

[Describe any libraries you'll be using and share your reasoning for including them.](#)

[Describe how you will implement Google Play Services or other external services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: 写代码、测试、修复、写代码、测试、修复.....](#)

悦动音符 (Pleasant Note)

Description

“悦动音符”是一款简洁好用的在线音乐播放应用。
您将在这里看到最热门的榜单，听到最热门的歌曲。
在悦动的音符中触碰灵魂，享受震撼！

Intended User

以年轻人（20岁-30岁）为主的群体。

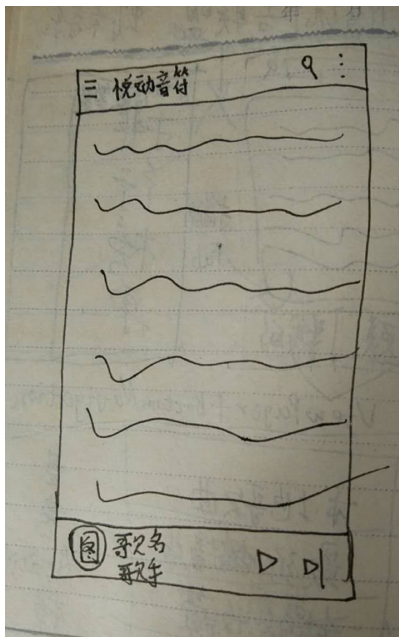
Features

- 播放在线或者本地音乐
- 根据歌名或者歌手名搜索线上歌曲

- 收藏与下载歌曲
- 获取“最近播放”歌曲列表
- 排行榜导航
- 小部件播放/暂停/切换歌曲
- 显示位置与头像等信息
- 插入广告
- 后台更新数据

User Interface Mocks

Screen 1



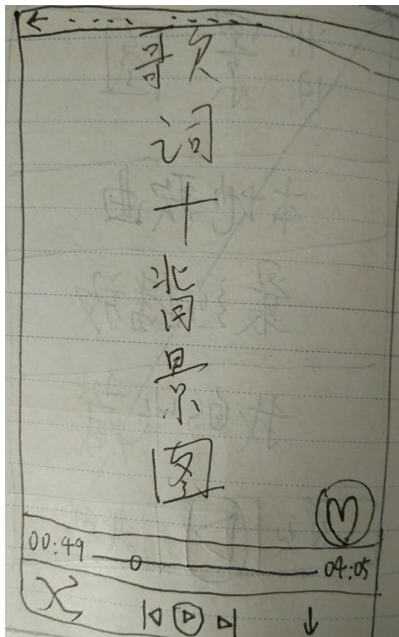
首页默认展示热门歌曲列表。

利用导航抽屉进行排行榜导航，抽屉头部显示当前位置和头像等信息，用户可以自行上传头像。

顶部搜索歌曲（将SearchView嵌入菜单）。

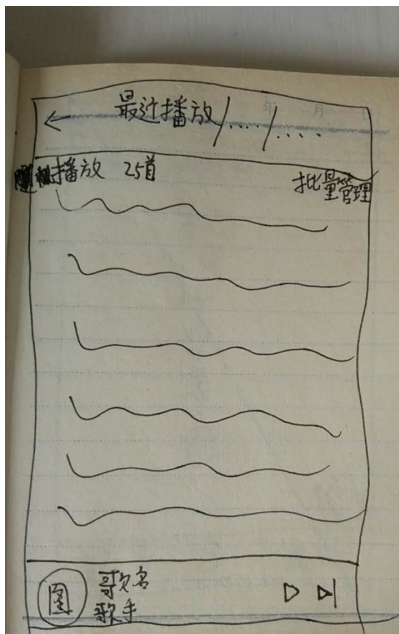
菜单项包含：搜索、本地歌曲、最近播放、我的收藏。

Screen 2



“播放”页面。

Screen 3



“最近播放”、“本地歌曲”以及“我的收藏”共用同一个活动。点击“批量管理”可以进行批量下载或删除。

Key Considerations

How will your app handle data persistence?

应用将使用 Content Provider + CursorLoader 处理数据本地化。

Describe any edge or corner cases in the UX.

没有WIFI的情况下听在线歌曲将消耗较多流量，网络条件会影响在线听歌的体验。

Describe any libraries you'll be using and share your reasoning for including them.

使用Glide加载图片，使用CircleImageView显示圆框图片，使用Gson解析Json，使用Volley处理短时联网任务，使用高德定位SDK处理定位任务，使用Android-Job后台更新数据，通过腾讯广告联盟插入广告，通过阿里云云市场的相关API获取音乐数据，全局使用Data Binding优化View与View Controller.

Describe how you will implement Google Play Services or other external services.

使用高德定位SDK进行定位并显示位置信息，使用腾讯广告联盟相关API插入广告。

Next Steps: Required Tasks

Task 1: Project Setup

- 设置目标 Android SDK Version 与最低 Android SDK Version
- 添加相关的依赖库
- 配置 Data Binding 与其它 SDK
- 配置相关的API

Task 2: Implement UI for Each Activity and Fragment

- 在MainActivity中生成首页UI
- 在PlayActivity中生成播放UI

- 在MusicListActivity中生成音乐列表UI

Task 3: 写代码、测试、修复、写代码、测试、修复.....

结合实际情况与个人经验，从MainActivity及其布局开始，一步步向前推进。
先重点进攻核心功能，最后再进行全面地细节化处理。