

**软 件 学 院**

**实践环节报告**

**课程名称： 软件工程实践二**

**课题名称：**  **网易云音乐网页版**

**专 业： 软件工程**

**班 级：** **卓越161**

**学 号：** **201619140430**

**学生姓名：** **张亚光**

**指导教师：**  **郭丽**

**2019年 3 月 28 日**

**中原工学院软件学院**

**软件工程实践二设计任务书**

|  |  |  |
| --- | --- | --- |
| **姓 名** | 张亚光 | 软件工程 专业 卓越161 班 |
| **题 目** | 网易云音乐网页版 | |
| **设**  **计**  **任**  **务** | 开发一个网页版的音乐播放平台。主要功能有：  1. 歌手发布专辑，上传音乐。  2. 用户搜索音乐，播放音乐，创建歌单，收录音乐到歌单。  3. 用户之间的评论、私信。  4．对用户进行音乐的个性化推荐。  开发工具：IDEA，WebStorm，Mysql。 | |
| **时**  **间**  **进**  **度** | 第17周（2019.02.25~2019.03.03）：需求分析，功能设计。  第18周（2019.03.04~2019.03.10）：界面设计，编码实现。  第19周（2019.03.11~2019.03.17）：功能测试，系统。 | |
| **原 主**  **始 要**  **资 参**  **料 考**  **与 文**  **献** | [01]汪云飞.Java EE开发的颠覆者Spring Boot实战 [M]:北京.电子工业出版社. 2016.3  [02]Craig Walls.Spring Boot实战 [M]:北京.人民邮电出版社.2016.9  [03]翟永超.Spring Cloud 微服务实战 [M]:北京.电子工业出版社.2017.5 | |

**指导教师签字： 年 月 日**

网易云音乐网页版

摘 要

开发一个网页版的音乐播放平台。人们对着音乐的需求很大，提供一个搜索音乐，用户评论互动，找到更多喜欢相同音乐的朋友，以及根据用户的爱好进行音乐的个性化推荐。在如今，新的音乐层出不穷，而每个人希望听到更多自己喜欢听的音乐。在众多的音乐中对用户推荐更符合用户喜好的音乐。仿照网易云音乐网页版。

关键词：个性化推荐；互动；评论。

目录

[第一章 项目分析 1](#_Toc6169788)

[1.1 项目概览 1](#_Toc6169789)

[1.2 技术分析 1](#_Toc6169790)

[第二章 系统设计 2](#_Toc6169791)

[2.1 系统功能设计 2](#_Toc6169792)

[2.2 系统用例图 3](#_Toc6169793)

[第三章 用例规约 4](#_Toc6169794)

[第四章 数据库设计 16](#_Toc6169795)

[4.1 数据表结构 16](#_Toc6169796)

[4.2 ER图 22](#_Toc6169797)

[第五章 结束语 23](#_Toc6169798)

# 项目分析

## 项目概览

一个网页版的音乐播放器。主要实现两方面的功能，一类是歌手，歌手可以在该平台发布原创专辑，上传歌曲，对歌曲，专辑的一些简介。歌手也可以将歌曲分享到动态，发动态，以增大歌曲流行度，扩大个人影响力。另一类是用户，用户可以搜索音乐，播放音乐，下载音乐。关注喜欢的歌手和用户，还可以按照自己的喜好创建歌单，将喜欢的音乐进行分类，可以将喜欢的音乐或者歌单分享给关注的人，对喜欢的音乐发表自己的感悟，评论说出自己的想法，找到更多喜欢的相同歌曲的朋友。还有提供对用户进行个性化推荐，根据用户听歌的偏好，特点，综合考虑分析出用户在音乐方面的一个画像，向用户进行精准的歌曲推荐。仿照网易云音乐的歌单，歌曲，个性化推荐等功能。

## 技术分析

使用前后端分离的开发方法，使前后端各司其职，功能划分清晰明了，也使项目易于维护。

前端使用H5，Vue，ElementUI，axios，npm，webpack等技术。H5是代码更加具有语义化，使用H5新增支持的标签功能，简化前端构建;使用Vue渐进式框架，借助其双向绑定的特点，减少前端js对DOM的操作，提升性能，简化开发。ElementUI快速构建精美的页面，使整体页面具有良好的一致性。axios对后端进行请求数据，npm，webpack对前端工程进行依赖添加，打包构建。

后端使用SSM框架，即Spring+SpringMVC+MyBatis，使用mysql进行数据存储，使用FastDFS存储音乐文件。

# 系统设计

## 系统功能设计

系统主要分歌手和用户两类，歌手主要在该平台发布专辑，上传音乐。用户可以搜索，播放音乐。详细功能如下。

歌手功能：

1. 登录注册
2. 发布专辑
3. 查看专辑
4. 上传歌曲
5. 查看作品

用户功能：

1. 搜索、播放、下载音乐
2. 创建、修改、查看、收藏、删除歌单。添加音乐到歌单
3. 分享音乐、歌单、专辑到动态
4. 关注歌手，用户，对歌手用户进行私信
5. 对音乐、歌单、专辑、动态进行评论

## 系统用例图

系统用例一共有十五个用例，其中部分是用户和歌手的相同用例。主要用例分为音乐，歌单和评论。系统用例图如下图2.1所示。

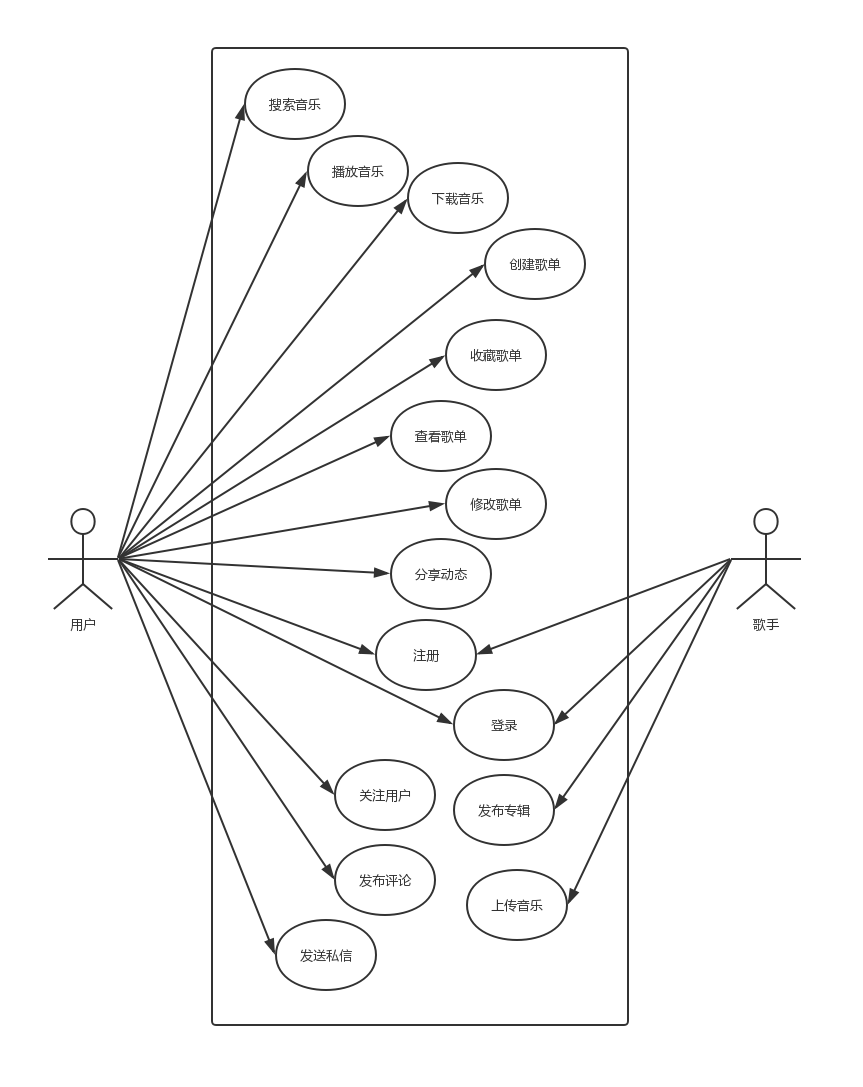


图2.1 系统用例图

# 用例规约

用例规约一共十五个，多数为歌手和用户共同的用例，故共同用例不再区分主体。

1. 登录

用户输入正确的账号和密码进行登录系统。

表3.1 登录用例

|  |  |
| --- | --- |
| 用例名称 | 登录 |
| 用例ID | UC-1 |
| 角色 | 用户/歌手 |
| 用例说明 | 用户通过邮箱进登录 |
| 前置条件 | 打开系统到登录页，输入正确的邮箱和密码。 |
| 基本事件流 | 用户点击登录，请求后台响应，登录成功。 |
| 异常事件流 | 未注册，或者密码错误，则提示用户相应信息 |
| 后置条件 | 登录成功，给予用户提示并转到首页。 |

1. 注册

用户通过邮箱进行注册，输入正确的邮箱和通过密码校验。

表3.2 注册用例

|  |  |
| --- | --- |
| 用例名称 | 注册 |
| 用例ID | UC-2 |
| 角色 | 用户/歌手 |
| 用例说明 | 用户通过邮箱进行注册。 |
| 前置条件 | 打开系统到注册页面，输入正确的邮箱，符合条件的密码。 |
| 基本事件流 | 点击注册，请求后台响应，注册成功。 |
| 异常事件流 | 邮箱格式错误，密码错误，提示用户 |
| 后置条件 | 注册成功，提示用户并跳转到登录界面。 |

1. 搜索音乐

用户可以对自己喜欢的音乐直接进行搜索，可以按照歌曲名称，歌手名称进行搜索，根据搜索的类型不同展示不同的信息。例如按照歌曲搜索，则展示相应歌曲的名称，歌手，所属专辑和时长等信息，若按照歌手名称搜索，则显示歌手的头像和昵称。如图3-1所示。

表3.3 搜索音乐

|  |  |
| --- | --- |
| 用例名称 | 搜索音乐 |
| 用例ID | UC-3 |
| 角色 | 用户/歌手 |
| 用例说明 | 根据关键字对音乐进行搜索 |
| 前置条件 | 打开系统搜索框，输入关键字 |
| 基本事件流 | 点击搜索，根据关键字请求后台音乐信息。 |
| 异常事件流 | 暂无相关音乐，提示数据为空。 |
| 后置条件 | 展示相应的音乐。 |



图3-1 搜索音乐

1. 播放音乐

音乐平台的基本功能，播放音乐，以及相关的上一曲，下一曲，暂停等。如图3-2所示。

表3.4 播放音乐

|  |  |
| --- | --- |
| 用例名称 | 播放音乐 |
| 用例ID | UC-4 |
| 角色 | 用户/歌手 |
| 用例说明 | 点击音乐的播放按钮，进行播放音乐。 |
| 前置条件 | 进入歌曲列表界面。 |
| 基本事件流 | 点击播放按钮，请求音乐资源。 |
| 异常事件流 | 若网络不佳则对用户进行提示。 |
| 后置条件 | 播放音乐 |



图3-2 播放音乐

1. 下载音乐

用户如果想要离线，则可以选择下载音乐。点击下载按钮之后，弹出下载界面，让用户选择下载路径。

表3.5 下载音乐

|  |  |
| --- | --- |
| 用例名称 | 下载音乐 |
| 用例ID | UC-5 |
| 角色 | 用户/歌手 |
| 用例说明 | 点击歌曲列表的下载按钮，下载音乐。 |
| 前置条件 | 进入音乐列表界面 |
| 基本事件流 | 点击相应音乐下载按钮，弹出下载界面，选择瞎子啊路径，点击确认，开始下载音乐。 |
| 异常事件流 | 点击下载之后未响应，用户重试。 |
| 后置条件 | 成功下载音乐 |

1. 创建歌单

用户可以对自己喜欢的一些歌曲进行归类，创建属于自己的歌单。可以分享歌单给别人。如图3-3所示。

表3.6 创建歌单

|  |  |
| --- | --- |
| 用例名称 | 创建歌单 |
| 用例ID | UC-6 |
| 角色 | 用户/歌手 |
| 用例说明 | 用户对自己喜欢的音乐进行归类，创建歌单。 |
| 前置条件 | 用户已经登录，到个人页。 |
| 基本事件流 | 点击创建歌单按钮，创建歌单。 |
| 异常事件流 | 歌单名称重复，创建失败，提示用户。 |
| 后置条件 | 用户歌单列表有新创建的歌单。 |



图3-3 创建歌单

1. 收藏歌单

如果用户在浏览时，看到比较喜欢的歌单，可以对歌单进行收藏。方便下次找到。点击歌单的封面或者下面的文字，进入歌单详情页，点击收藏按钮进行收藏。

表3.7 收藏歌单

|  |  |
| --- | --- |
| 用例名称 | 收藏歌单 |
| 用例ID | UC-7 |
| 角色 | 用户/歌手 |
| 用例说明 | 对喜欢的歌单进行收藏。 |
| 前置条件 | 用户已登录，歌单存在。 |
| 基本事件流 | 点击歌单的收藏按钮，对歌单进行收藏。 |
| 异常事件流 | 若歌单被已收藏则收藏按钮无法点击。 |
| 后置条件 | 可以在收藏歌单列表找到收藏的歌单。 |

1. 查看歌单

用户不仅可以对音乐进行搜索，还可以对歌单进行搜索查看，展示歌单的名称，图片，创建人，收藏数，标签，介绍，歌曲列表，歌曲名称，时长，歌手，所属专辑。如图3-4所示。

表3.8 查看歌单

|  |  |
| --- | --- |
| 用例名称 | 查看歌单 |
| 用例ID | UC-8 |
| 角色 | 用户/歌手 |
| 用例说明 | 查看歌单的具体信息，收录的歌曲。 |
| 前置条件 | 暂无 |
| 基本事件流 | 点击歌单。 |
| 异常事件流 | 请求资源失败，显示暂无信息。 |
| 后置条件 | 展示歌单的具体信息以及收录歌曲。 |

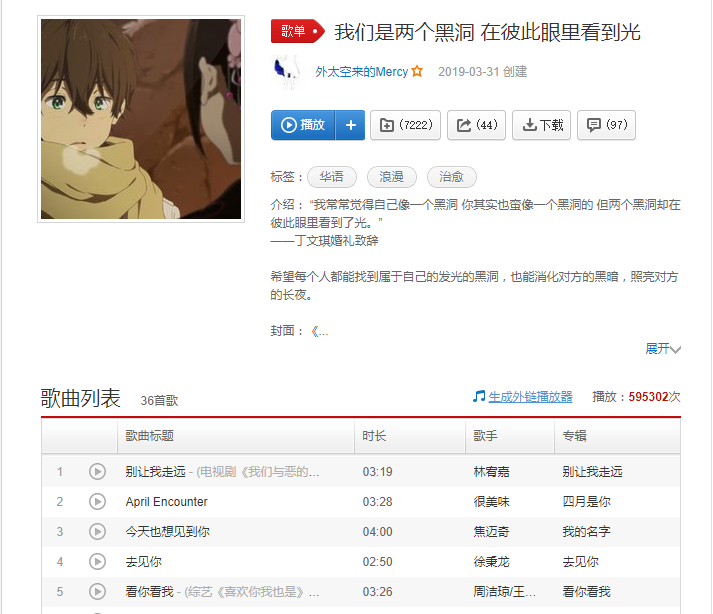


图 3-4 查看歌单

1. 修改歌单

用户可以对自己创建的歌单进行修改，可以更换歌单封面，修改对歌单的描述，添加或者删除里面收录的歌曲。如图3-5所示。

表3.9 修改歌单

|  |  |
| --- | --- |
| 用例名称 | 修改歌单 |
| 用例ID | UC-9 |
| 角色 | 用户/歌手 |
| 用例说明 | 修改歌单的相关信息。 |
| 前置条件 | 用户已登录，修改用户自己创建的歌单。 |
| 基本事件流 | 点击歌单的编辑按钮，修改信息，确认修改。 |
| 异常事件流 | 若添加歌曲已存在提示用户歌曲已存在。 |
| 后置条件 | 修改成功，可以看到修改后的歌单信息。 |



图 3-5修改歌单

1. 分享动态

用户对于喜欢的音乐和歌单等，可以分享到动态，分享的时候可以添加简短的文字说明。如图3-6所示。

表3.10 分享动态

|  |  |
| --- | --- |
| 用例名称 | 分享动态 |
| 用例ID | UC-10 |
| 角色 | 用户/歌手 |
| 用例说明 | 分享音乐，专辑到动态。 |
| 前置条件 | 用户已登录 |
| 基本事件流 | 点击音乐，专辑的分享按钮。 |
| 异常事件流 | 网络原因导致发送失败，保留用户内容，以便重试。 |
| 后置条件 | 在动态中可以看到分享的动态。 |



图 3-6 分享动态

1. 关注用户

用户对于喜欢的歌手或者用户可以进行关注，关注之后可以收到更多关于关注的人的消息，动态优先展示关注的人的动态。如果关注的歌手发布了新的音乐或者专辑，可以接收到歌手发来的通知。此时只有歌手同时关联了普通用户身份之后才能进行关注，动态，私信等都是以歌手普通用户身份进行。如图3-7所示。

表3.11 关注用户

|  |  |
| --- | --- |
| 用例名称 | 关注用户 |
| 用例ID | UC-11 |
| 角色 | 用户/歌手 |
| 用例说明 | 对喜欢的歌手/用户进行关注。 |
| 前置条件 | 歌手/用户已经登录。 |
| 基本事件流 | 找到歌手/用户的个人页，点击关注按钮。 |
| 异常事件流 | 歌手若无个人用户则无法关注。 |
| 后置条件 | 可以在自己的关注列表里看到关注的用户/歌手。 |



图 3-7 关注用户

1. 发送私信

用户可以对其他用户发送私信，分享歌曲给他人。

表3.12 发送私信

|  |  |
| --- | --- |
| 用例名称 | 发送私信 |
| 用例ID | UC-12 |
| 角色 | 用户/歌手 |
| 用例说明 | 用户发送私信给用户。 |
| 前置条件 | 用户已登录，接收方允许收到私信。 |
| 基本事件流 | 点击关注的人的私信按钮，编辑私信内容，点击发送。 |
| 异常事件流 | 如果是歌手，歌手要有个人用户账号，否则无法发送。 |
| 后置条件 | 接收方收到私信。 |



图 3-8 发送私信

1. 发布评论

用户可以对喜欢的音乐，专辑，歌单以及动态进行评论。如图3-9所示。

表3.13 发布评论

|  |  |
| --- | --- |
| 用例名称 | 发布评论 |
| 用例ID | UC-13 |
| 角色 | 用户/歌手 |
| 用例说明 | 对音乐专辑等进行评论。 |
| 前置条件 | 用户已登录。 |
| 基本事件流 | 点击评论按钮，编写评论，发布评论。 |
| 异常事件流 | 暂无 |
| 后置条件 | 在评论去看到自己的评论。 |



图 3-9 发布评论

1. 发布专辑

歌手可以发布专辑，发布时编辑专辑的相关信息，上传里面收录的歌曲。

表3.14 发布专辑

|  |  |
| --- | --- |
| 用例名称 | 发布专辑 |
| 用例ID | UC-14 |
| 角色 | 歌手 |
| 用例说明 | 歌手发布新的专辑 |
| 前置条件 | 歌手已登录 |
| 基本事件流 | 点击发布专辑按钮，编写专辑信息，上传音乐到专辑。 |
| 异常事件流 | 暂无 |
| 后置条件 | 可以看到歌手的专辑。 |

1. 上传歌曲

歌手上传音乐到自己的专辑。

表3.15 上传歌曲

|  |  |
| --- | --- |
| 用例名称 | 上传歌曲 |
| 用例ID | UC-15 |
| 角色 | 歌手 |
| 用例说明 | 歌手上传音乐到专辑。 |
| 前置条件 | 歌手已登录，已创建专辑。 |
| 基本事件流 | 点击上传歌曲按钮，上传歌曲到专辑。 |
| 异常事件流 | 暂无 |
| 后置条件 | 看到歌手的专辑内有 |

# 数据库设计

## 数据表结构

数据库表经过系统分析和整理，一共有十张表。各个表的结构如下表所示。

表4.1 用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 用户唯一标识 |  | false |
| nick\_name | varchar | 50 | 用户昵称 |  | false |
| email | varchar | 50 | 用户邮箱 |  | false |
| password | varchar | 50 | 用户密码 |  | false |
| image | varchar | 2083 | 头像url | "" | true |
| gender | int | 4 | 用户性别 | 2 | true |
| tel | long |  | 用户手机号码 | 0 | true |
| address | varchar | 50 | 用户地址 | "" | true |
| discription | varchar | 50 | 用户描述，简介 | "" | true |
| register\_time | date |  | 注册时间 | 当天 | false |
| notification | json |  | 允许通知类型 |  | false |
| birthday | date |  | 用户生日 | 0-0-0 | true |
| moment\_number | int | 4 | 动态记录数 |  |  |
| follow\_number | int | 4 | 关注人数 |  |  |
| follower\_number | int | 4 | 粉丝人数 |  |  |

表4.2 歌手表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 歌手唯一标识 |  | false | |
| nick\_name | varchar | 50 | 歌手昵称 |  | false | |
| email | varchar | 50 | 歌手邮箱 |  | false | |

续表4.2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| password | varchar | 50 | 歌手密码 |  | false |
| image | varchar | 2083 | 头像url | "" | true |
| gender | int | 4 | 歌手性别 | 2 | true |
| tel | long |  | 歌手手机号码 | 0 | true |
| address | varchar | 50 | 歌手地址 | "" | true |
| discription | varchar | 50 | 描述，简介 | "" | true |
| introduction | varchar |  | 歌手的介绍 | "" | true |
| register\_time | date |  | 注册时间 | 当天 | false |
| notification | json |  | 允许通知类型 |  | false |
| user\_id | long |  | 歌手普通用户id |  | true |
| birthday | date |  | 歌手生日 | 0-0-0 | true |
| moment\_number | int | 4 | 动态记录数 | 0 | false |
| follow\_number | int | 4 | 关注人数 | 0 | false |
| follower\_number | int | 4 | 粉丝人数 | 0 | false |
| tag | json |  | 歌手标签 |  | true |

表4.3 关注表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| user\_id | long |  | 用户id |  | false |
| follow\_id | long |  | 用户关注的用户id |  | false |
| follow\_nick\_name | varchar | 50 | 关注用户的昵称 | "" | false |
| follow\_image | varchar | 2083 | 关注用户的头像 | "" | false |

表4.4 歌曲表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| url | varchar | 2083 | 歌曲url |  | false |
| name | varchar |  | 歌曲名称 |  | false |
| belong\_album\_id | long |  | 歌曲所属专辑id |  | false |
| belong\_album\_name | varchar | 50 | 歌曲所属专辑名称 |  | false |

表4.5 专辑表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| title | varchar | 50 | 专辑名称 | "" | false |
| description | varchar | 50 | 专辑简介描述 | "" | true |
| singer\_id | long | 50 | 歌手id |  | false |
| singer\_name | varchar | 50 | 歌手名称 | "" | false |

表4.6 歌单表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| title | varchar | 50 | 歌单名称 | "" | false |
| description | varchar | 3000 | 歌单描述 |  | true |
| image | varchar | 2083 | 歌单图片 |  | false |
| creater\_id | long |  | 创建者id |  | false |
| creater\_nick\_name | varchar | 50 | 创建者昵称 | "" | false |
| create\_time | date |  | 创建时间 | 当天 | false |
| collection\_number | long |  | 收藏数 | 0 | false |

续表4.6

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| collection\_music | json |  | 收录的歌曲["id":123, "title":"歌曲名称","singer":"GEM"}] |  | false |
| tag | json |  | 歌单标签["流行", "轻音乐"] |  | true |

表4.7 评论表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| type | int | 4 | 对什么类型进行的评论，是歌曲，歌单还是专辑等 |  | false |
| content | varchar | 500 | 评论的内容 |  | false |
| create\_time | timestamp |  | 评论时间 | current\_timestamp | false |
| thumb\_number | int | 4 | 点赞数 | 0 | false |
| record\_id | long |  | 对歌曲，歌单，专辑等的id |  | false |
| commentator\_id | long |  | 评论者id |  | false |
| commentator\_cick\_name | varchar | 50 | 评论者的昵称 |  | false |
| commentator\_image | varchar | 2083 | 评论者头像 |  | false |

表4.8 动态表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| creater\_id | long |  | 创建者id |  | false |
| creater\_nick\_name | varchar | 50 | 创建者昵称 |  | false |
| creater\_image |  | 2083 | 创建者头像 |  | false |
| content | json |  | 内容 |  | false |
| publish\_time | timestamp |  | 发布时间 | current\_timestamp | false |

表4.9 用户收藏表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| user\_id | long |  | 用户id |  | false |
| sheet\_id | long |  | 歌单id |  | false |
| sheet\_title | varchar | 50 | 歌单名称 | "" | false |
| sheet\_image | varchar | 2083 | 歌单图片 | "" | false |

表4.10 私信表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 备注 | 默认值 | 允许为空 |
| id | long |  | 唯一标识 |  | false |
| sender\_id | long |  | 发送者id |  | false |
| sender\_nick\_name | varchar | 50 | 发送者昵称 |  | false |
| sender\_image | varchar | 2083 | 发送者头像 |  | false |
| receiver\_id | long |  | 接受者id |  | false |
| receiver\_nick\_name | varchar | 50 | 接收者昵称 |  | false |
| receiver\_iamge | varchar | 50 | 接收者头像 |  | false |
| content | json |  | 发送消息内容 | "" | false |
| sending\_time | timestamp |  | 发送时间 | current\_timestamp | false |

## ER图

系统各个实体之间的关系如下图所示。

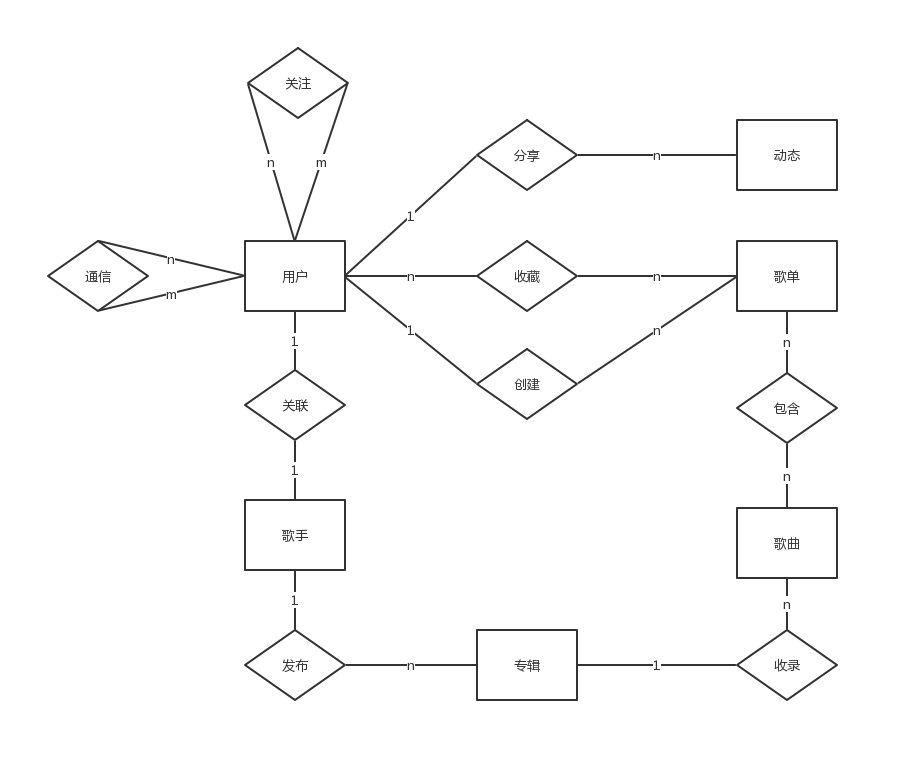


图4.1 ER图

# 结束语

通过这次的课题，学到了很多东西，知道了设计对于工程的重要性，一个设计的好坏直接影响了后面代码的编写难易程度，以及后面的维护成本。这次的完成情况不是很好，还有很多地方可以做的更好，可以进行优化。还有一项功能完成之后一定要进行完整的测试，不能把程序开发的错误呈现给用户。这样用户的体验是非常不好的。以后要更加细心，认真地对待，提高代码的质量以及项目的稳定性。感谢郭老师认真的指导，以后会更努力的。

指导教师评语

成绩：

指导教师签名：

年 月 日