Hex后端开发文档3.0

## 功能详细设计

#### 功能基本描述

本项目后端应用Flask框架，接受前端不同的ajax请求，从数据库读取所需内容，在后台进行处理以后向前端返回json格式的数据以完成前后端的基本交互。

主要实现如下功能：

1. 爬取网易云音乐热门歌单的歌曲信息，并将数据存入sqlite数据库中。当前端发来请求时从数据库去除所需 的数据，以json的格式返回给前端进行显示。
2. 对应歌曲‘版权信息’属性为“无”的条目，模拟在qq音乐和酷狗音乐上进行搜索，将爬取到的结果更新到数据库中，返回歌曲的真实url给前端。
3. 对于前端的注册登录请求，将前端传递来的用户名密码写入数据库；对于“添加歌单”功能，将传递的歌单信息写入用户数据库。

#### 功能实现流程

1. 使用python 的requests库进行网易云音乐及其对应歌曲页面框架的网页源代码获取，对得到的源码进行分析获取歌单和歌曲信息，保存到数据库中。
2. 搭建轻量级python web框架Flask 测试运行，Flask 是一个 Python 实现的 Web 开发微框架，其 WSGI 工具箱采用 Werkzeug ，模板引擎则使用 Jinja2 。Flask使用 BSD 授权。本项目将在该框架下对爬取的数据进行处理并与前端进行交互。
3. 编写后台数据处理的函数，主要包括：
4. 歌曲信息返回函数：从数据库中获取已经爬取好的对应歌单的歌曲条目信息，打包成json格式的字符串，返回给前端。
5. 模拟搜索函数：对于没有版权信息为“false”的歌曲，对前端发起的搜索请求应用selenium方法进行模拟搜索，将爬取到的结果打包成json格式的字符串返回给前端
6. 播放跳转函数：当前端发送“播放”请求时，返回当前歌曲的url；

规定所有数据传输格式为json格式的字符串。

1. 编写ajax请求和响应的函数。提供给前端合适的接口，确定请求命令的具体格式。

#### 流程基本模型

#### 传输数据规范

前后端的数据格式均为json格式。主要包括如下信息：

1. 歌单信息：歌单名称 歌单类型
2. 歌曲信息：歌曲名 歌手 专辑名 歌曲url 版权信息
3. 用户信息：用户名 密码