

遼寧大學

# Linux 操作系统 实验报告

题    目：\_\_\_\_\_基于 LAMP 环境的音乐网站搭建\_\_\_\_\_

院    系：\_\_\_\_\_信息学院\_\_\_\_\_

专    业：\_\_\_\_\_计算机科学与技术\_\_\_\_\_

姓    名：\_\_\_\_\_刘欣月\_\_\_\_\_

指导教师：\_\_\_\_\_纪文迪\_\_\_\_\_

完成日期：\_\_\_\_\_2020 年 6 月 29 日\_\_\_\_\_

# 基于 LAMP 架构的音乐网站搭建

## 1.设计背景

当下社会，互联网和网络技术的迅猛发展，使得人们对网络的需求日益提高，通常在生活中我们需要使用各式各样的软件来下载音乐不仅不方便管理，而且占用了储存空间。例如下载的歌曲不喜欢听的还要删除，十分不便。但如果有一个网络音乐库那么情况将大不一样。作为一个数字音乐交互网站，人们可以通过它方便、快捷、安全地实现音乐搜索查找，实时试听下载收藏，互动交流等，从而为人与人之间搭建一个自由、自主和安全的音乐世界随性畅聊的局域网。

## 2.系统分析与设计

网站的主要目的是为用户提供一个内容全面，操作便捷简易的在线音乐试听交流网站。

### 2.1 系统功能描述

该选课系统所具有的功能：

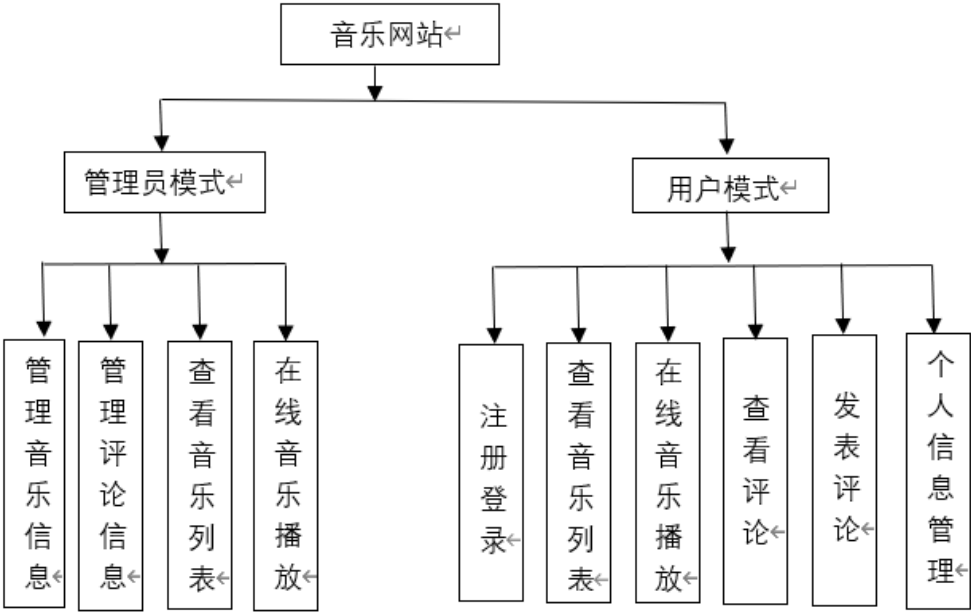
用户模块：

- 1．用户可以注册，并登录；
- 2．用户可以查看最新音乐列表；
- 3．用户可以使用在线音乐播放功能；
- 4．用户可以查看并发表评论；
- 5．用户可以查看并修改个人信息；
  - ①.头像修改
  - ②.个人信息修改
  - ③.密码修改

管理员模块：

1. 管理员登陆后可以查看、添加、删除音乐及信息
2. 管理员可以管理评论信息
3. 管理员可以查看最新音乐列表，以便检查音乐信息
4. 管理员可以使用在线音乐播放功能，以便检查音乐文件

## 2.2 系统功能流程



## 3. 网站实现

### 3.0 开发工具

HTML, CSS, JavaScript 实现页面展示与交互，PHP 实现数据处理与业务逻辑，MySQL 数据库存储资源相关信息。

### 3.1 环境搭建

#### 3.1.1 Apache 安装

##### 1. 解压安装 Apache 需要的所有软件

```
# tar -xf httpd-2.4.41.tar.gz
```

```
# tar -xf apr-1.6.5.tar.gz
```

```
# tar -xf apr-util-1.6.1.tar.gz
```

```
# tar -xf pcre-8.43.tar.gz
```

```
# tar -xf zlib-1.2.11.tar.xz
```

## 2. Apache 默认需要依赖 pcre 软件，但系统预安装的 pcre 无法使用，更换适合版本

```
# cd /root/lamp/pcre-8.43
```

```
# ./configure
```

```
# make
```

```
# make install
```

## 3. 安装 zlib

```
# cd /root/lamp/zlib-1.2.11
```

```
# ./configure
```

```
# make
```

```
# make install
```

## 4. Apache 的加密传输模块 mod\_ssl，需要安装此软件产生

```
# apt-get openssl-devel
```

```
#install openssl-devel
```

## 5. httpd 软件安装

```
# cd /lamp/httpd-2.4.41
```

```
# ./configure \
```

```
--prefix=/usr/local/apache2 \
```

```
--sysconfdir=/usr/local/apache2/etc \
```

```
--with-included-apr \
```

```
--enable-so \
```

```
--enable-deflate=shared \
```

```
--enable-expires=shared \
```

```
--enable-rewrite=shared \
```

```
--enable-ssl
```

```
# make
```

```
# make install
```

## 6. 启动 Apache 测试

```
# /usr/local/apache2/bin/apachectl start
```

```
# ps aux | grep httpd
```

使用进程查看命令确认 Apache 是否启动，是否产生进程

```
# netstat -tlun
```

验证: It works!

### 3.1.2 安装 MySQL

#### 1. 安装 ncurses

```
# apt-get ncurses-devel
```

```
#install ncurses-devel
```

```
# cd /root/lamp/ncurses-6.1
```

```
# ./configure --with-shared --without-debug --without-ada --enable-overwrite
```

```
# make
```

```
# make install
```

#### 2. 安装 MySQL

```
# tar -xf mysql-8.0.17.tar.gz //初次安装版本 后来更换为 5.7.3
```

```
#useradd -r -s /sbin/nologin mysql
```

为 MySQL 软件创建运行用户, 创建为系统用户, 并限制此用户登录操作系统

```
#cd /root/lamp/mysql-8.0.17/
```

```
#cmake3 \
```

```
-DDOWNLOAD_BOOST=1
```

```
-DWITH_BOOST=/boot/ \
```

```
-DCMAKE_INSTALL_PREFIX=/usr/local/mysql \
```

```
-DCMAKE_C_COMPILER=/opt/rh/devtoolset-8/root/usr/bin/gcc
```

```
DCMAKE_CXX_COMPILER=/opt/rh/devtoolset-8/root/usr/bin/g++ \
```

```
-DMYSQL_UNIX_ADDR=/tmp/mysqld.sock \
```

```
-DDEFAULT_CHARSET=utf8 \
```

```
-DDEFAULT_COLLATION=utf8_general_ci \
```

```
-DWITH_EXTRA_CHARSETS:STRING=utf8,gbk \
```

```
-DWITH_MYISAM_STORAGE_ENGINE=1 \
```

```
-DWITH_INNOBASE_STORAGE_ENGINE=1 \
```

```
-DWITH_MEMORY_STORAGE_ENGINE=1 \
```

```
-DWITH_READLINE=1 \  
-DFORCE_INSOURCE_BUILD=1 \  
-DENABLED_LOCAL_INFILE=1 \  
-DMYSQL_DATADIR=/data/mysql \  
-DMYSQL_USER=mysql \  
-DMYSQL_TCP_PORT=3306
```

```
# make
```

```
# make install
```

### 3.MySQL 调整相应配置文件和参数

a. 修改 MySQL 目录的用户归属

```
#chown -R mysql:mysql /usr/local/mysql
```

b. 初始化数据库

```
#mkdir -p /data/mysql/
```

```
#chown -R mysql:mysql /data/mysql/
```

```
#/usr/local/mysql/bin/mysqld --initialize --user=mysql \
```

```
--basedir=/usr/local/mysql --datadir=/data/mysql
```

c. 修改配置文件，并初始化授权表

```
#vi /etc/my.cnf
```

```
[mysqld]
```

```
datadir=/data/mysql
```

```
basedir=/user/local/mysql
```

```
socket=/tmp/mysqld.sock
```

```
# Disabling symbolic-links is recommended to prevent assorted security risks
```

```
symbolic-links=0
```

```
# Settings user and group are ignored when systemd is used.
```

```
# If you need to run mysqld under a different user or group,
```

```
# customize your systemd unit file for mariadb according to the
```

```
# instructions in http://fedoraproject.org/wiki/Systemd
```

```
[mysqld_safe]
```

```
log-error=/var/log/mariadb/mariadb.log
```

```
pid-file=/var/run/mariadb/mariadb.pid
```

```
# include all files from the config directory
```

```
!includedir /etc/my.cnf.d
```

d. 创建并授权[mysqld\_safe]的两个选项目录和文件

```
# mkdir /var/log/mariadb
```

```
# touch /var/log/mariadb/mariadb.log
```

```
# chown -R mysql:mysql /var/log/mariadb/
```

```
# mkdir /var/run/mariadb/
```

```
# chown -R mysql:mysql /var/run/mariadb/
```

e. 启动 MySQL 服务

用原本源代码的方式去使用和启动 mysql

```
/usr/local/mysql/bin/mysqld_safe --user=mysql &
```

f. 登录 MySQL

```
# /usr/local/mysql/bin/mysql -u root -p
```

进入之后必须先更改密码才能操作

```
mysql> alter user root@'localhost' identified by "123456";
```

```
mysql> show databases;
```

```
mysql> use test;
```

```
mysql> show tables;
```

```
mysql> exit
```

g. 关闭 MySQL 服务

```
# /usr/local/mysql/bin/mysqladmin -uroot -p shutdown
```

Enter password:123456

### 3. 1. 3 安装 PHP

#### 1. 安装 libxml2

```
# tar -xf libxml2-2.9.9.tar.gz
```

```
# cd /root/lamp/libxml2-2.9.9
```

```
# ./configure --prefix=/usr/local/libxml2/
```

```
# make
```

```
# make install
```

## 2. 安装 libmcrypt

```
# tar -xf libmcrypt-2.5.8.tar.gz
```

```
# cd /root/lamp/libmcrypt-2.5.8
```

```
# ./configure --prefix=/usr/local/libmcrypt/
```

```
# make
```

```
# make install
```

安装 libltdl

```
# cd /root/lamp/libmcrypt-2.5.8/libltdl
```

```
# ./configure --enable-ltdl-install
```

```
# make
```

```
# make install
```

## 3. 安装 mhash

```
# tar -xf mhash-0.9.9.9.tar.gz
```

```
# cd /root/lamp/mhash-0.9.9.9
```

```
# ./configure
```

```
# make
```

```
# make install
```

## 4. 安装 mcrypt

```
# tar -xf mcrypt-2.6.8.tar.gz
```

```
# cd /lamp/mcrypt-2.6.8
```

```
# export LD_LIBRARY_PATH=/usr/local/libmcrypt/lib:/usr/local/lib
```

```
# ./configure --with-libmcrypt-prefix=/usr/local/libmcrypt
```

```
# make
```

```
# make install
```

## 5. 安装 libpng

```
# tar -xf libpng-1.6.7.tar.gz
```

```
# cd /root/lamp/libpng-1.6.7
```

```
# ./configure --prefix=/usr/local/libpng
```



```
# make
```

```
# make install
```

## 6. 安装 jpeg6

```
#tar -xf jpegsrc.v9c.tar.gz
```

```
# cd /root/lamp/jpeg-9c/
```

```
# ./configure --prefix=/usr/local/jpeg9/ --enable-shared --enable-static
```

```
# make
```

```
# make install
```

--enable-shared 与--enable-static 参数分别为建立共享库和静态库使用的 libtool

## 7. 安装 freetype

```
# tar -xf freetype-2.10.0.tar.gz
```

```
# cd /root/lamp/freetype-2.10.0
```

```
# ./configure --prefix=/usr/local/freetype/
```

```
# make
```

```
# make install
```

## 8. 安装 PHP

```
# tar -xf php-7.3.10.tar.gz
```

```
# cd /root/lamp/php-7.3.10
```

```
# ./configure \
```

```
--prefix=/usr/local/php/ \
```

```
--with-config-file-path=/usr/local/php/etc/ \
```

#指定配置文件目录

```
--with-apxs2=/usr/local/apache2/bin/apxs \
```

#指定 apache 动态模块位置

```
--with-libxml-dir=/usr/local/libxml2/ \
```

#指定 libxml 位置

```
--with-jpeg-dir=/usr/local/jpeg9/ \
```

#指定 jpeg 位置

```
--with-png-dir=/usr/local/libpng/ \
```

#指定 libpng 位置

```
--with-freetype-dir=/usr/local/freetype/ \
```

#指定 freetype 位

```
--with-mcrypt=/usr/local/libmcrypt/ \
```

#指定 libmcrypt 位置

```
--with-gd \
```

#启用 gd 库

```
--enable-soap \
```

#支持 soap 服务(api)

```
--enable-mbstring=all \
```

#支持多字节, 字符串

```
--enable-sockets \ #支持套接字
--with-pdo-mysql=/usr/local/mysql \ #启用 mysql 的 pdo 模块支持
--with-pdo-mysql=mysqlnd \ #Mysql Native Driver
--with-gd --without-pear #不安装 pear(Composer 代替)
# make&& make install
```

PHP 安装后需要调整相应配置文件和参数才能正常运行。

## 9. 生成 php 配置文件

```
# mkdir /usr/local/php/etc
# cp /root/lamp/php-7.3.10/php.ini-production /usr/local/php/etc/php.ini
```

## 10.修改 Apache 配置文件,使其识别.php 文件, 并能通过 php 模块调用 php 进行页面解析

```
# vim /usr/local/apache2/etc/httpd.conf
AddType application/x-httpd-php .php
```

## 11.重启 Apache 服务

```
# /usr/local/apache2/bin/apachectl stop
# /usr/local/apache2/bin/apachectl start
```

## 12.测试 apache 和 php 连通性

```
# vim /usr/local/apache2/htdocs/test.php
<?php phpinfo(); >
```

通过浏览器输入地址访问: [http://Apache 服务器地址/test.php](http://Apache服务器地址/test.php)

## 3.1.4 安装 phpMyAdmin

```
# unzip phpMyAdmin-5.0.0-alpha1-all-languages.zip
# cp -a /root/lamp/phpMyAdmin-5.0.0-alpha1-all-languages
/usr/local/apache2/htdocs/phpmyadmin
# cd /usr/local/apache2/htdocs/phpmyadmin
# cp -a config.sample.inc.php config.inc.php
# vim config.inc.php
$cfg['Servers'][$i]['auth_type'] = 'cookie';
```

```
$cfg['Servers'][$i]['auth_type'] = 'http';
```

#设置 auth\_type 为 HTTP 身份认证模式

通过浏览器输入地址访问: http://Apache 服务器地址/phpmyadmin/index.php

## 3.2 出现的问题及解决

### 1. Apache 第一次安装失败,

**解决方案:** 源码包 2.4 版本中默认没有集成 apr 的依赖包,所以需要提前解决依赖问题

```
# cp -a /root/lamp/apr-1.6.5 /root/lamp/httpd-2.4.41/src/lib/apr
```

```
# cp -a /root/lamp/apr-util-1.6.1 /root/lamp/httpd-2.4.41/src/lib/apr-util
```

编译 apr 时出错, 缺少 expat 库

```
#apt-get expat-devel
```

```
#install expat-devel
```

**2.启动服务提示: httpd: Could not reliably determine the server's fully qualified domain name, using localhost.localdomain. Set the 'ServerName' directive globally to suppress this message**

**解决方案:** 修改配置文件

打开主配置文件 httpd.conf

```
ServerName localhost:80
```

### 3.安装 PHP Configure: error: freetype.h not found

**解决方案:** 补充依赖包

```
install freetype-devel
```

### 4.管理员模式登录网站上传音乐文件失败, 原因显示上传文件过大

**解决方案:** 修改 PHP 配置文件:

```
max_execution_time = 600    //每个 PHP 页面运行的最大时间
max_input_time = 600       //每个 PHP 页面接收数据所需的最大时间
memory_limit = 32m         //每个 PHP 页面所吃掉的最大内存
file_uploads = on          //允许通过 HTTP 上传文件
upload_tmp_dir = /tmp       //文件上传至服务器上存储临时文件的地方
upload_max_filesize = 32m   //允许上传文件大小的最大值
post_max_size = 32m        //通过表单 POST 给 PHP 的所能接收的最大值
```

注: lamp 环境搭建与 4 月, 除上面的问题之外, 还遇到很多问题, 但时间久了, 只能记住上面几个了。

## 3.5 运行结果

### 3.5.0 网站地址：

[www.christinalau.top](http://www.christinalau.top) (之前域名备案出了问题，后来重新备案，预计 7 月 15 日之后域名备案能完成，在这之前您可使用 88 端口查看网站效果)

```
C:\Users\xyl55>ping www.christinalau.top

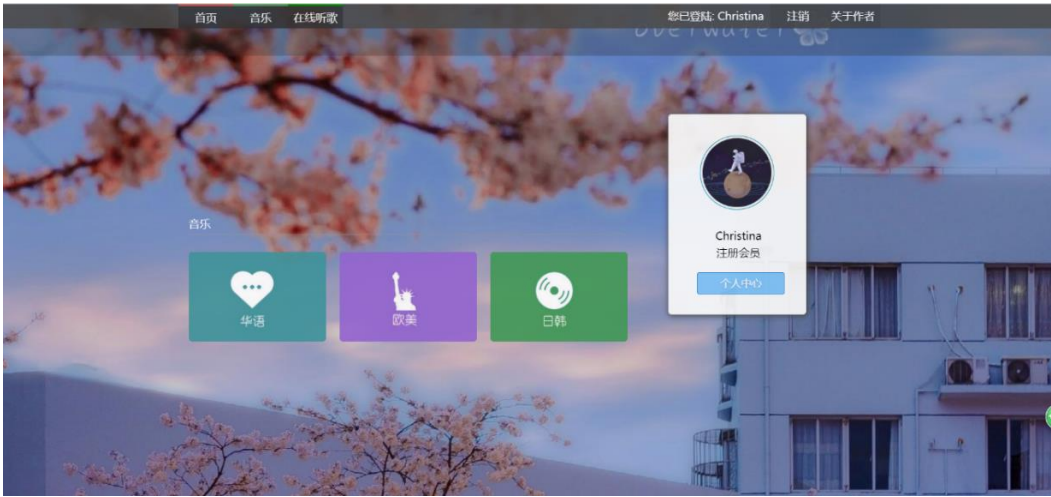
正在 Ping www.christinalau.top [39.106.122.158] 具有 32 字节的数据:
来自 39.106.122.158 的回复: 字节=32 时间=34ms TTL=50
来自 39.106.122.158 的回复: 字节=32 时间=34ms TTL=50
来自 39.106.122.158 的回复: 字节=32 时间=30ms TTL=50
来自 39.106.122.158 的回复: 字节=32 时间=32ms TTL=50

39.106.122.158 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 30ms, 最长 = 34ms, 平均 = 32ms
```

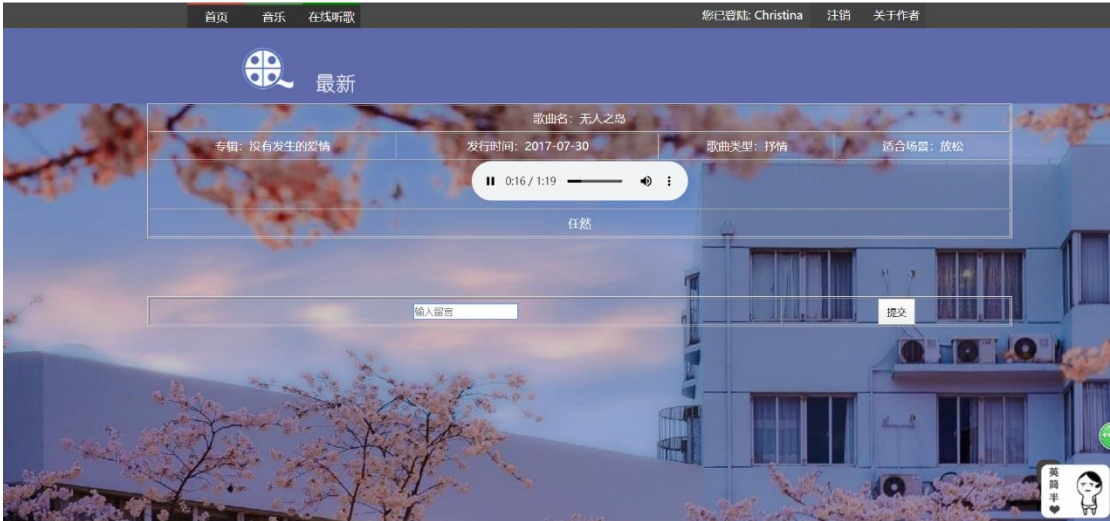
### 3.5.1 用户登录与注册界面：



### 3.5.2 首页



### 3.5.3 在线音乐播放界面



### 3.5.4 评论页面



[首页](#)

[音乐](#)

[在线听歌](#)

您已登录: Christina 注销 关于作者

 最新

歌曲名: 无人之岛

专辑: 没有发生的爱情

发行时间: 2017-07-30

歌曲类型: 抒情

适合场景: 放松

0:03 / 1:19





任然

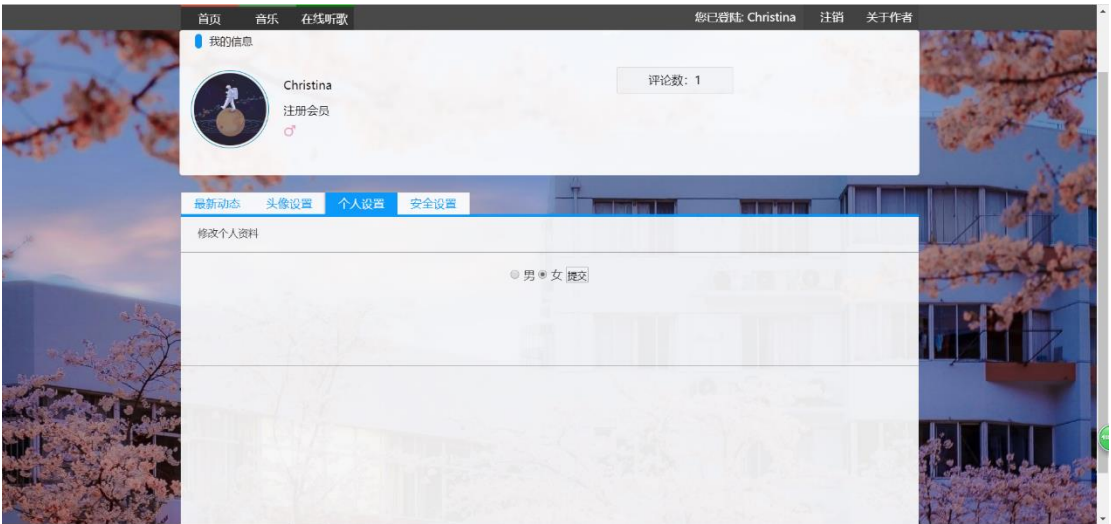
Christina

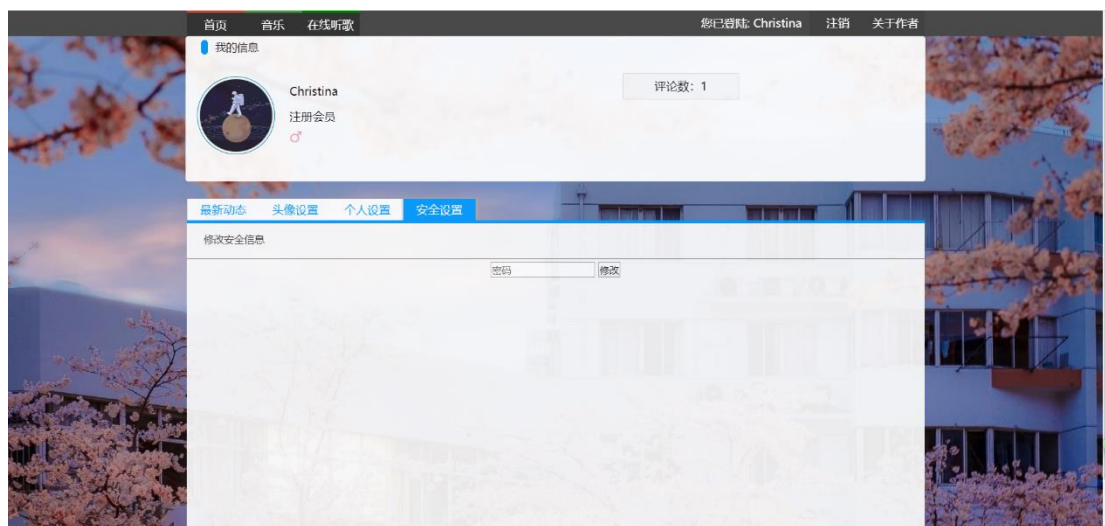
2020-06-29 09:05:17

hahaha

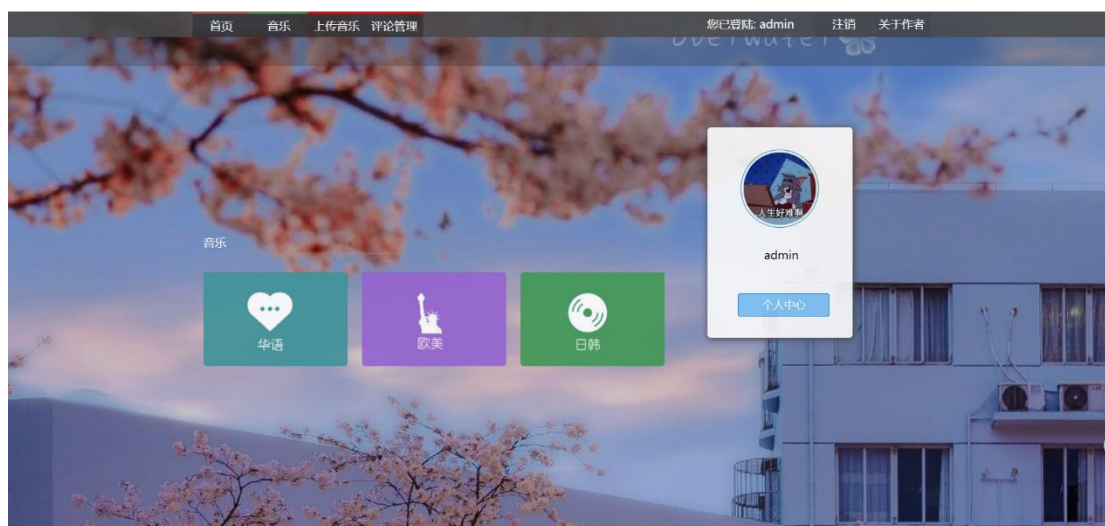
提交

3.5.5 个人信息管理界面：





### 3.5.6 管理员登陆界面：






### 3.5.7 管理员上传音乐界面：

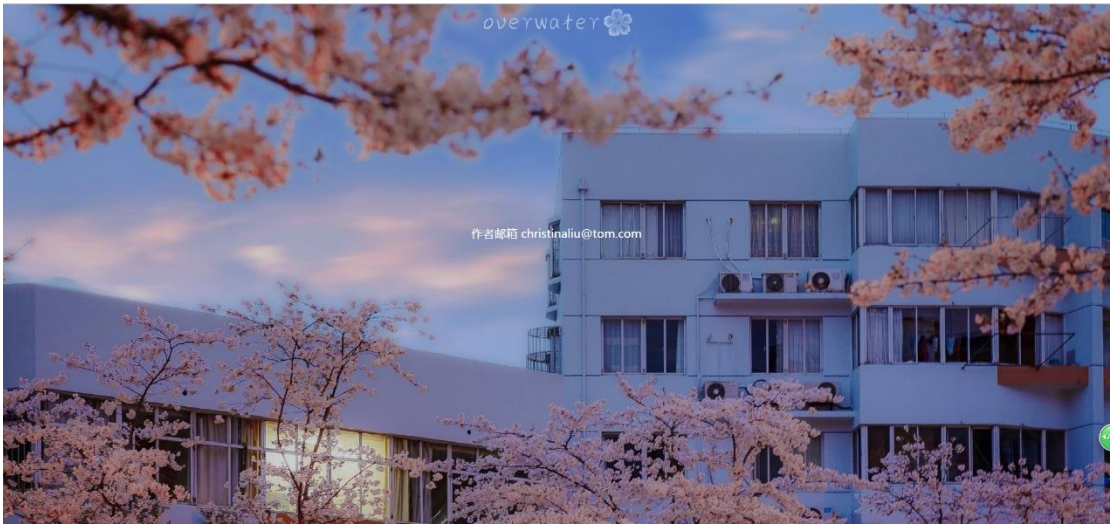




3.5.8 管理员管理评论界面：

首页 音乐 上传音乐 评论管理			您已登陆: admin 注销 关于作者		
123456	 	2020-06-07 01:50:10			
hahaha	 	2020-06-29 09:05:17			
1234567867654321	 	2020-06-07 08:27:52			
1111111111	 	2020-06-07 08:28:05			
2222222	 	2020-06-07 08:29:13			
33333333333	 	2020-06-07 08:29:15			
44444444	 	2020-06-07 08:29:18			
1	 	2020-06-29 06:50:37			
5555555	 	2020-06-07 08:30:51			
1111	 	2020-06-28 09:04:17			

3.5.9 作者信息界面：



## 4.实验总结

为了真正地去接触学习 Linux 系统，在服务器搭建环境时并没有图简单省时去使用宝塔面板之类的工具，而是原生搭建了 LAMP 环境。由于是初次接触，搭建过程中遇到了非常多的问题(两个总结：坎坷)，每遇到一个问题，都会仔细思考，独立解决不了就去网上搜索（安装搭载是遇到那么多的问题，百分之九十都可以从网上的找到问题所在，不得不感慨，网络为我的学习提供了极大的便利!!!），查阅书籍，虽然过程没有十分顺利，但遇到并解决这些问题却更加深了对 Linux 系统的了解。

网站建设采用的是原生写法，用的是本学期数据库应用开发技术课程中所学的知识，后续将优化界面，增加曲库数目等。

初次真正地使用 Linux 系统，感受最深的是当缺少某个软件时，使用自带的包管理工具可以很快地找到相应的安装包安装，可以根据自己的需求和喜好随意搭配各种环境。技术水平还很基础，也许这点感受对于 Linux 系统来讲，非常浅显，未来将会继续学习，提升技术。

多动手多思考，小刘同学要加油！