**总体设计：**

系统架构

微信端（用户管理员应用）

PC端（书库管理）

JAVA数据接口

PHP后台程序

MYSQL

数据库

ES搜索引擎

**接口设计：**

管理员登录接口

管理员退出接口

上架、下架图书接口

确认还书接口

添加管理员、删除管理员接口

添加书库、删除书库接口

管理员接口

添加分类、删除分类接口

管理员权限控制接口

授权借书接口

用户注册接口

用户信息修改接口

搜索书籍接口

确认借阅接口

收藏图书接口

还书提醒接口

添加书车接口

推荐阅读接口

在线预订接口

相关书籍接口

图书详情接口

新书推荐接口

分类导航接口

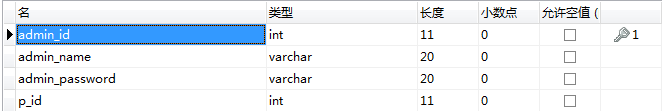
用户接口

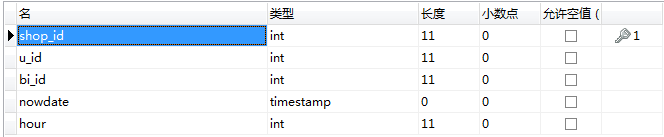
历史记录接口

预约书籍接口

数据结构设计：

admin管理员表

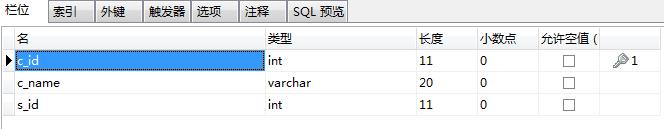


book\_cart书车表

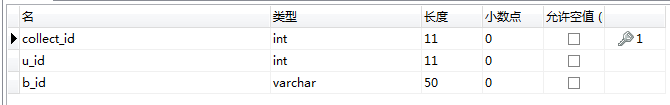
book\_item书项表

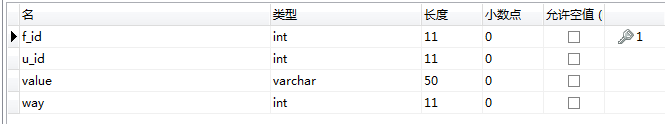


category分类表

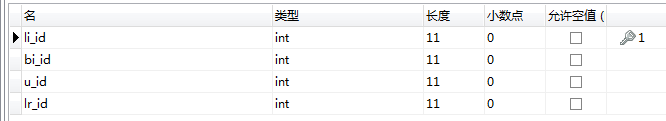


collection收藏表

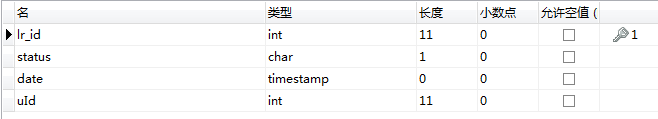


find\_record查找历史表

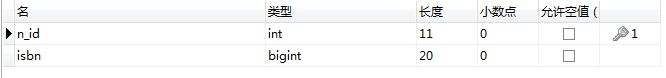
lend\_item借书记录对应关系表



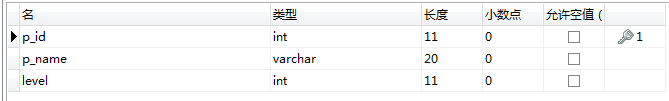
lend\_record借书记录表



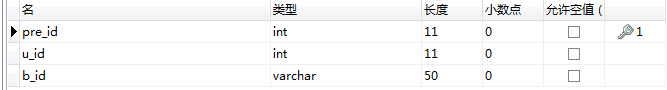
new\_book新书推荐表



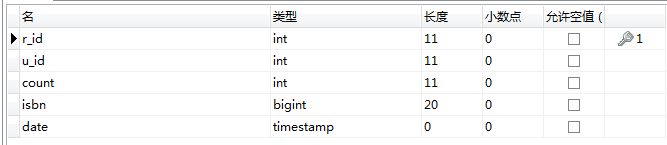
power管理员权限对应表

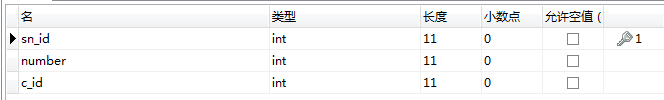


pre\_book预定书籍表

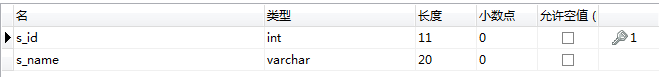


record详情查询记录表



serial\_num序列号表

stack书库表



user用户信息表



**运行设计:**

程序选用linux服务器作为硬件服务器系统，linux具有良好的稳定性，安全性，以及高性能。

采用nginx服务器做请求代理。当用户量和访问量增大时，可以用多台服务器，在服务器上放上相同的程序，用nginx做负载均衡。使得程序具有一定的高并发性。

将PHP程序放在apache服务器上。将java程序放在tomcat服务器中。Nginx将请求转发到apache或者tomcat。

为保证搜索引擎的性能，将elasticsearch搜索引擎放在单独的服务器上，暂时采用一个分片的形式，如果访问量增加，可以采用多个分片。形成分布式搜索引擎，提供高效的访问和一定的高并发性能。

**出错处理设计：**

1. 出错信息：

（1）第一类为预设出错信息，类似密码错误等。

（2）第二类为程序bug，未知信息。

2、出错处理：

第一类出错信息将给前端返回对应的消息值；第二类返回未知错误。出错后数据回滚，不会对系统造成影响。同时出错信息打印到日志文件，供修复bug查看。