

**内存数据库系统设计与实现文档**

|  |
| --- |
| Boying——博影票务系统 |

指导教师*:* **袁时金**

朱明志 (1854116)

石稼晟 (1851632)

谭梓煊 (1853434)

李翠琪 (1751022)

沈钰姣 (1853972)

In-Memory Database

Tongji University

School of Software Engineering

# 目录

[Boying——博影票务系统 I](#_Toc61821837)

[目录 I](#_Toc61821838)

[引言 1](#_Toc61821839)

[1.1 编写目的 1](#_Toc61821840)

[1.2 目标读者 1](#_Toc61821841)

[1.3 背景与依据 2](#_Toc61821842)

[概要设计 2](#_Toc61821843)

[1.4 系统总体架构设计 2](#_Toc61821844)

[1.5 系统软件结构设计 3](#_Toc61821845)

[1.6 接口设计 3](#_Toc61821846)

[1.6.1 类之间的接口 3](#_Toc61821847)

[1.6.2 模块之间的接口 9](#_Toc61821848)

[1.6.3 本系统和其他系统的接口 14](#_Toc61821849)

[1.7 界面设计 14](#_Toc61821850)

[1.7.1 前台演出购票 14](#_Toc61821851)

[1.7.2 后台管理 17](#_Toc61821852)

[1.8 数据库设计 21](#_Toc61821853)

[1.8.1 实体及其属性的介绍 21](#_Toc61821854)

[1.8.2 表结构 22](#_Toc61821855)

[1.8.3 数据库关系图 26](#_Toc61821856)

[1.9 系统出错处理设计 27](#_Toc61821857)

[1.9.1 出错信息 27](#_Toc61821858)

[系统软件结构 27](#_Toc61821859)

[类设计 27](#_Toc61821860)

[1.10 用户模块 27](#_Toc61821861)

[1.10.1 模块描述 28](#_Toc61821862)

[1.10.2 功能 29](#_Toc61821863)

[1.11 演出模块 38](#_Toc61821864)

[1.11.1 模块描述 39](#_Toc61821865)

[1.11.2 功能 40](#_Toc61821866)

[1.12 订单模块 46](#_Toc61821867)

[1.12.1 模块描述 47](#_Toc61821868)

[1.12.2 功能 47](#_Toc61821869)

[1.13 管理员管理模块 54](#_Toc61821870)

[1.13.1 模块描述 55](#_Toc61821871)

[1.13.2 功能 55](#_Toc61821872)

[附录A 图目录 58](#_Toc61821873)

# 引言

## 编写目的

《详细设计书》是项目设计的第二个阶段，这个阶段旨在将软件系统功能进一步细化，将原本复杂的功能拆分成不同的模块，并逐步开发健壮的系统结构，使软件系统需求能够很好的转化为未来设计，并进一步提高软件各方面性能。本文档依据高内聚低耦合的原则，将结构分解为不同的模块，各模块分别负责自己的功能，且模块与模块之间，模块与环境之间的调用关系尽可能简单。同时，在进行概要设计分析的过程中，发现并解决了系统中各个模块之间功能重合、冗余和功能短缺的部分，为详细设计和代码编写提供了保障。

概要设计解决了软件系统总体结构设计的问题，包括整个软件系统的结构、模块划分、模块功能和模块间的联系等。详细设计则要解决如何实现各个模块的内部功能，即模块设计。具体的说，模块设计就是要详细描述各模块中包含的类的处理逻辑及其提供接口的参数格式、请求方式。但这并不等同于系统实现阶段用具体的语言编码，它只是对实现细节作精确的描述，这样编码阶段就可以将详细设计中对功能实现的描述，直接翻译、转化为用某种程序设计语言书写的程序。详细设计的详细程度应达到可以编写程序的水平。

## 目标读者

本文档的目标读者为本软件的开发人员和其他相关人员。

## 背景与依据

本项目是一款网上演出购票平台，用户可以登录后查看上线的各大演出，并下单购买自己喜欢的演出门票，也可以管理自己已经购买的票务详情。

# 概要设计

## 系统总体架构设计

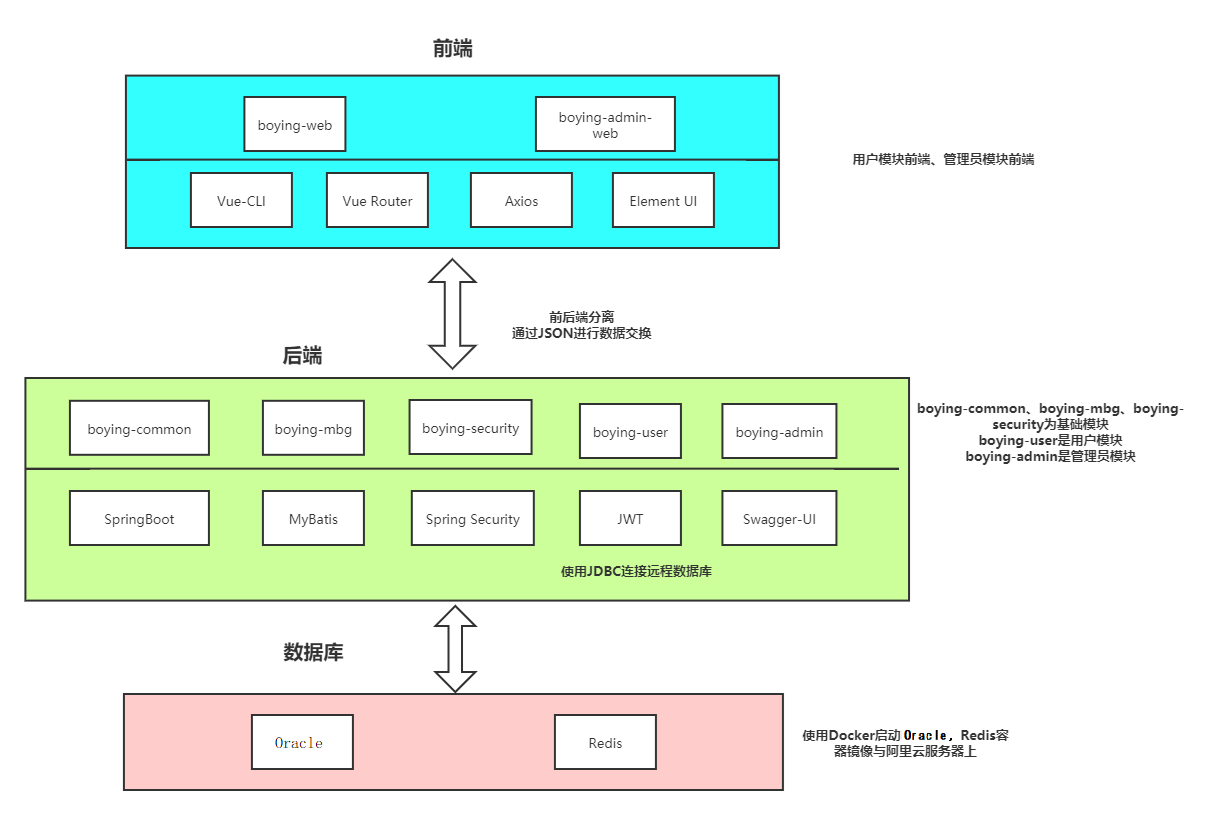


图 2.1系统总体架构设计

## 系统软件结构设计

图片包含 图形用户界面

描述已自动生成

图 2.2系统软件结构设计图

## 接口设计

### 类之间的接口

#### 获取演出目录

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **获取演出目录** |
| **URL** | /categories |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| CommonResult<List<BoyingCategory>> | string | 返回未被管理员删除的演出目录 |

#### 获取某个演出的详情

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **获取某个演出的详情** |
| **URL** | /show/detail/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **Id** | True | string | 想要查看详情的演出ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| CommonResult<BoyingShow> | string | 通过show\_id获取并返回该演出的详细信息 |

#### 搜索筛选演出

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **演出的综合搜索筛选** |
| **URL** | /show/search |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| keyword | False | string | 用户想要搜索的演出关键词 |
| city | False | String | 用户想要搜索的演出城市 |
| categoryId | False | Int | 用户想要搜索的演出类别 |
| pageNum | False | Int | 分页数 |
| pageSize | False | Int | 每页条数 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| CommonResult<CommonPage<BoyingShow>> | string | 返回符合要求的演出并按权重降序排列 |

#### 获取某场演唱会的所有座次

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **获取某场演唱会的所有座次** |
| **URL** | /seats |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| showId | True | string | 演出ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| CommonResult<List<BoyingSeat>> | string | 通过show\_id获取并返回该演出的所有座次 |

#### 用户下单

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户下单** |
| **URL** | /order/add |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| showId | True | string | 演出ID |
| showSeatIds | True | string | 座次ID |

#### 显示订单

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **显示订单** |
| **URL** | /order/list |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 用户ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| list | string | 订单列表 |
| pageNum | Int | 分页数 |
| pageSize | int | 每页条数 |

#### 获取订单详情

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **查看订单详情** |
| **URL** | /order/details/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 订单ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Data | string | 返回订单和票的信息 |

#### 删除订单

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **删除订单** |
| **URL** | /order/delete/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 订单ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果 |

#### 取消订单

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **取消订单** |
| **URL** | /order/cancel/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 订单ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果 |

### 模块之间的接口

#### 用户注册

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户注册** |
| **URL** | /register |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **username** | true | string | 用户注册的账号名称 |
| **password** | true | string | 用户的密码 |
| telephone | true | string | 用户的电话号码 |
| **authCode** | true | string | 验证码 |
| icon | False | string | 用户可以上传头像 |

#### 账号密码登陆

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户用账号密码登陆** |
| **URL** | /usernameLogin |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **username** | true | string | 用户输入的账号名称 |
| **password** | true |  | 用户输入的密码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 将用户输入的密码加密后与数据库查到的user密码对比（密码错误则报错给前端）,在服务器内存中存储了该用户信息，返回token给前端. |

#### 手机号密码登陆

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户用账号密码登陆** |
| **URL** | /telephoneLogin |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| telephone | true | string | 用户输入的手机号 |
| **password** | true |  | 用户输入的密码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 将用户输入的密码加密后与数据库查到的user密码对比（密码错误则报错给前端）,在服务器内存中存储了该用户信息，返回token给前端. |

#### 手机号验证码登陆

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户用账号密码登陆** |
| **URL** | /authCodeLogin |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| telephone | true | string | 用户输入的手机号 |
| authCode | true |  | 用户输入的验证码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 在服务器内存中存储了该用户信息，返回token给前端. |

#### 更新个人信息

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **更新个人信息** |
| **URL** | /user/update |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| age | true | string | 用户年龄 |
| Email | True | String | 用户邮箱 |
| Gender | true | string | 用户性别 |
| identityNumber | true | string | 用户的身份证号 |
| Realname | true | string | 用户真实姓名 |
| icon | True | string | 用户头像 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果状态。 |

#### 管理员登陆

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **管理员登陆** |
| **URL** | /login |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| username | true | string | 用户输入的用户名 |
| password | true | string | 用户输入的密码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 返回token给前端. |
| **tokenHead** | Bearer | 本地内存存储、生成token |

### 本系统和其他系统的接口

## 界面设计

用原型模型的设计方法对前台演出购票、后台管理两个前端网页进行了设计如下：

### 前台演出购票

界面原型设计：

1. 主页



图 2.3原型设计——主页

1. 搜索页面



图 2.4原型设计——搜索页

1. 订单页面

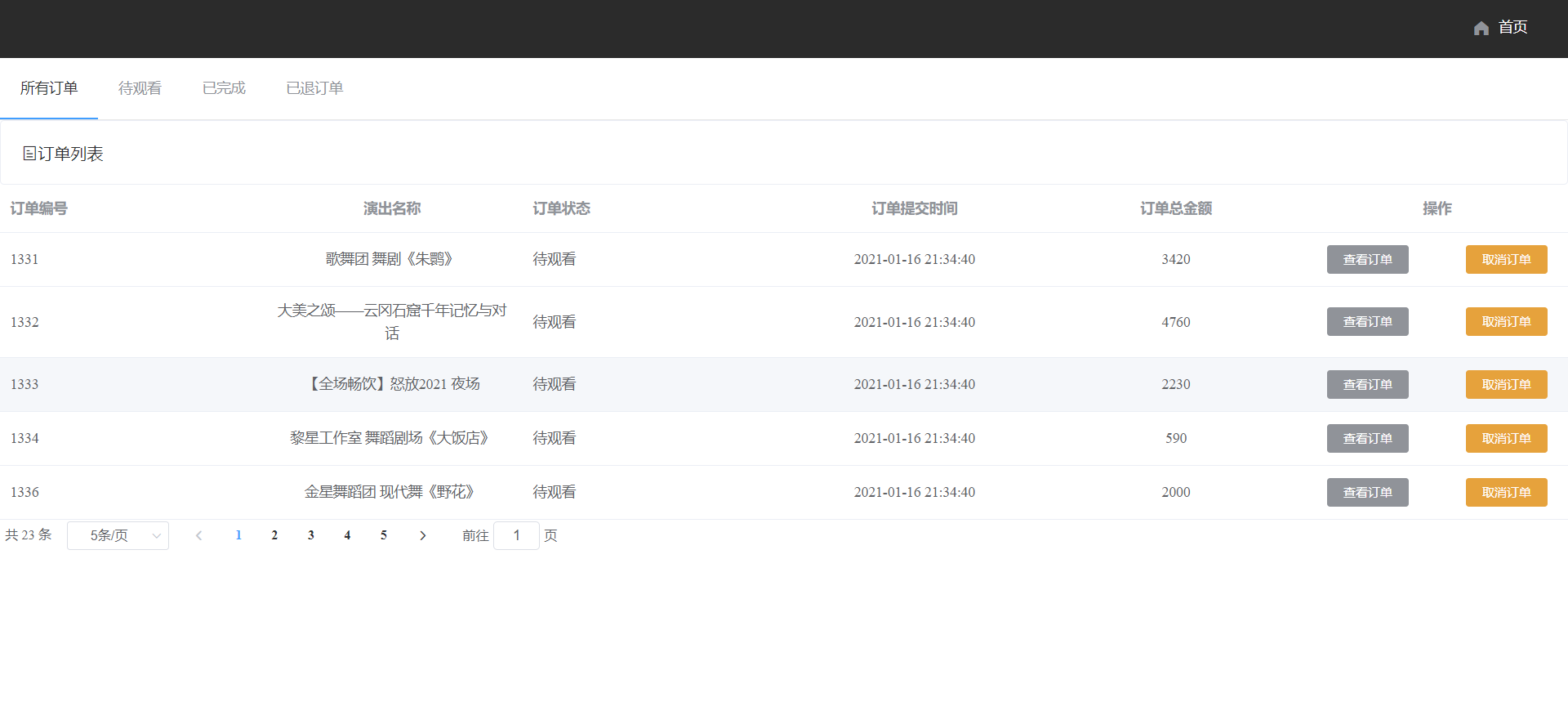


图 2.5原型设计——订单页

1. 用户信息页面

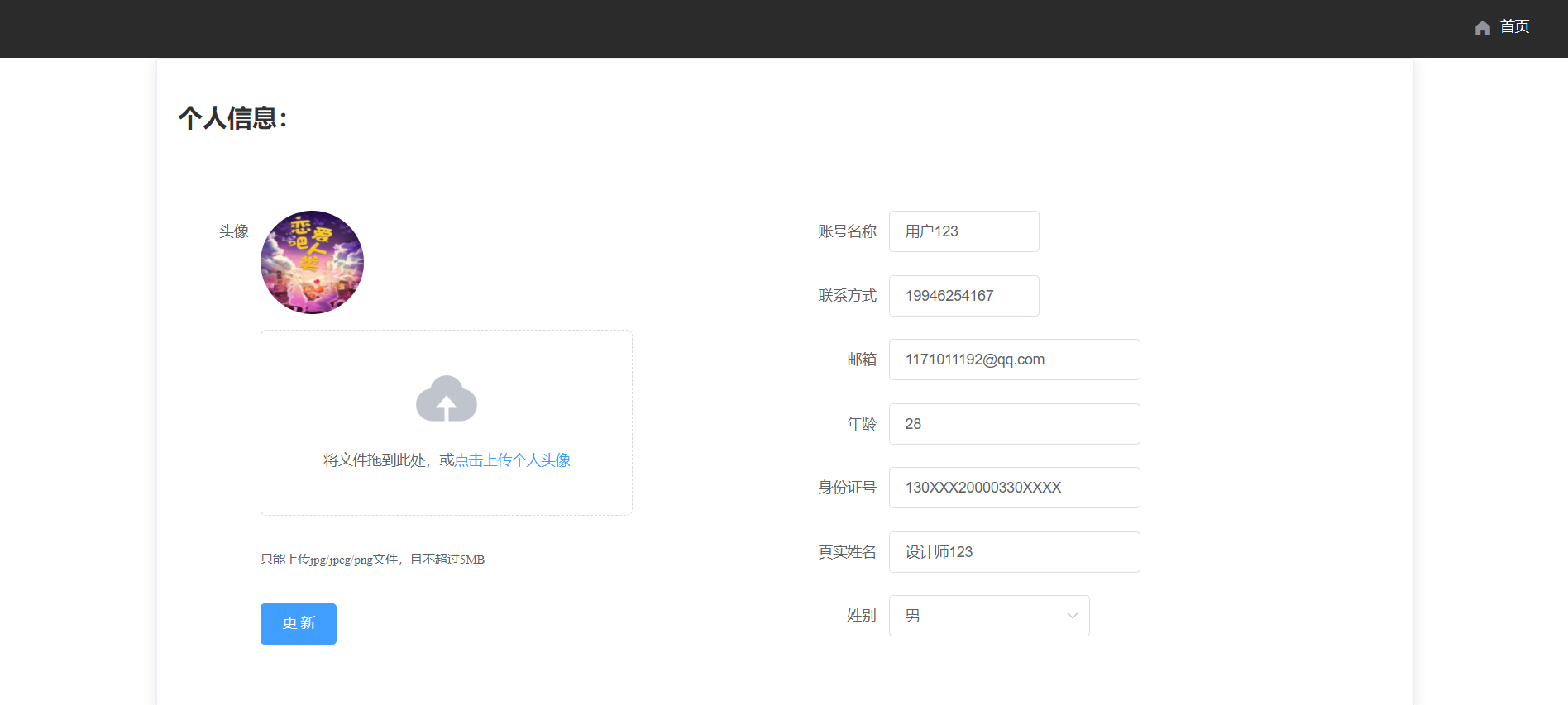


图 2.6原型设计——用户信息页

1. 演出详情页面

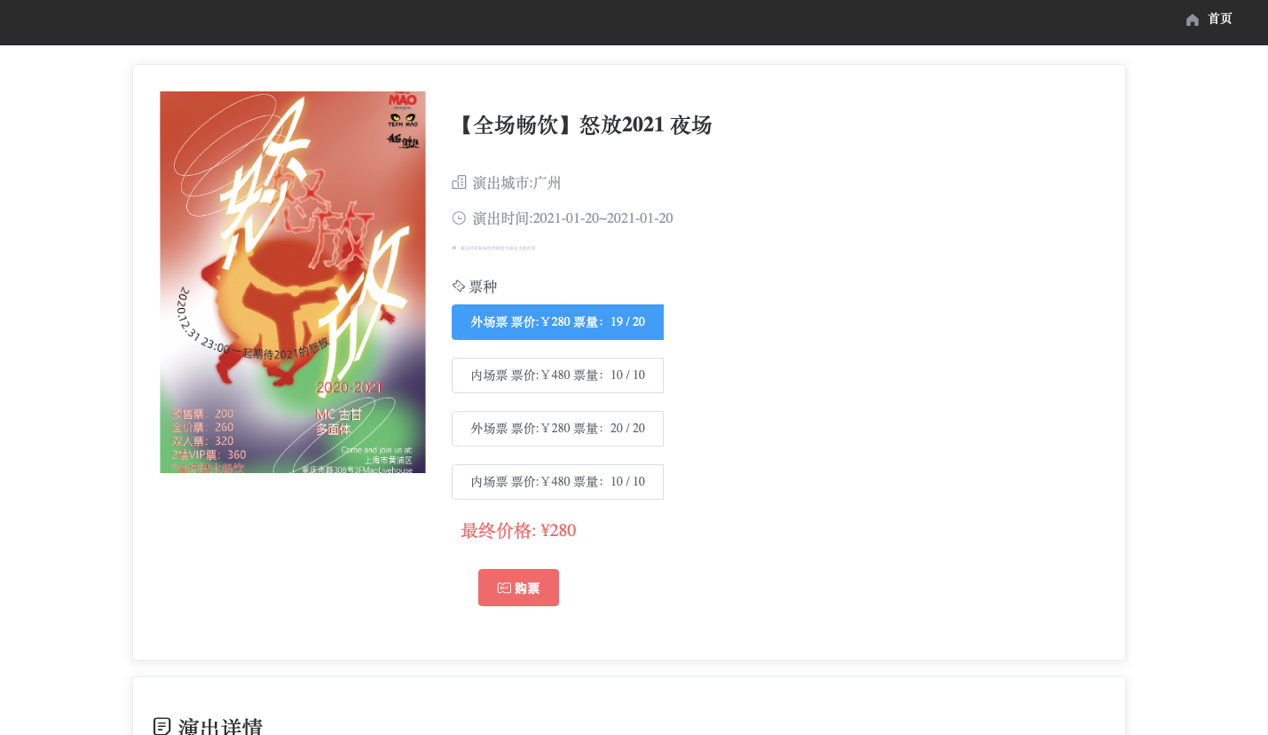


图 2.7原型设计——演出详情页

模块导航建模：

图形用户界面

描述已自动生成

图 2.8前台演出购票模块导航建模

### 后台管理

界面原型设计：

1. 主页

图表

中度可信度描述已自动生成

图 2.9原型设计——后台主页

1. 演出列表

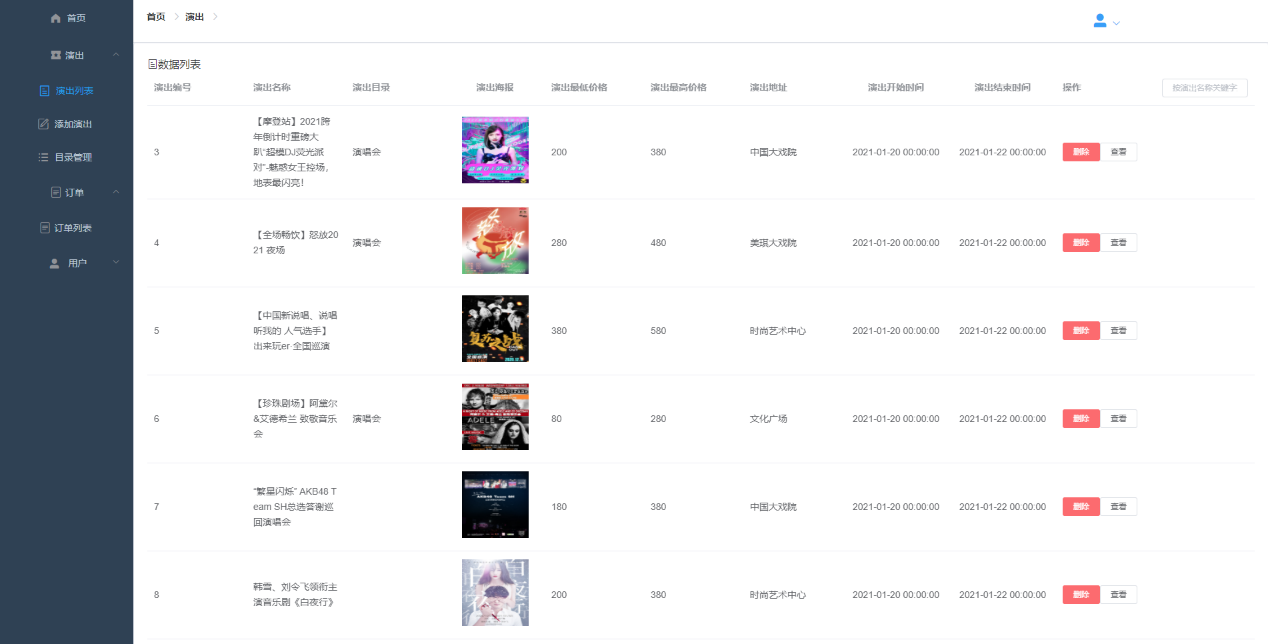


图 2.10原型设计——后台演出列表

1. 添加演出

图形用户界面, 应用程序

描述已自动生成

图 2.11原型设计——后台添加演出

1. 目录管理

图形用户界面, 应用程序, Teams

描述已自动生成

图 2.12原型设计——后台目录管理

1. 订单管理

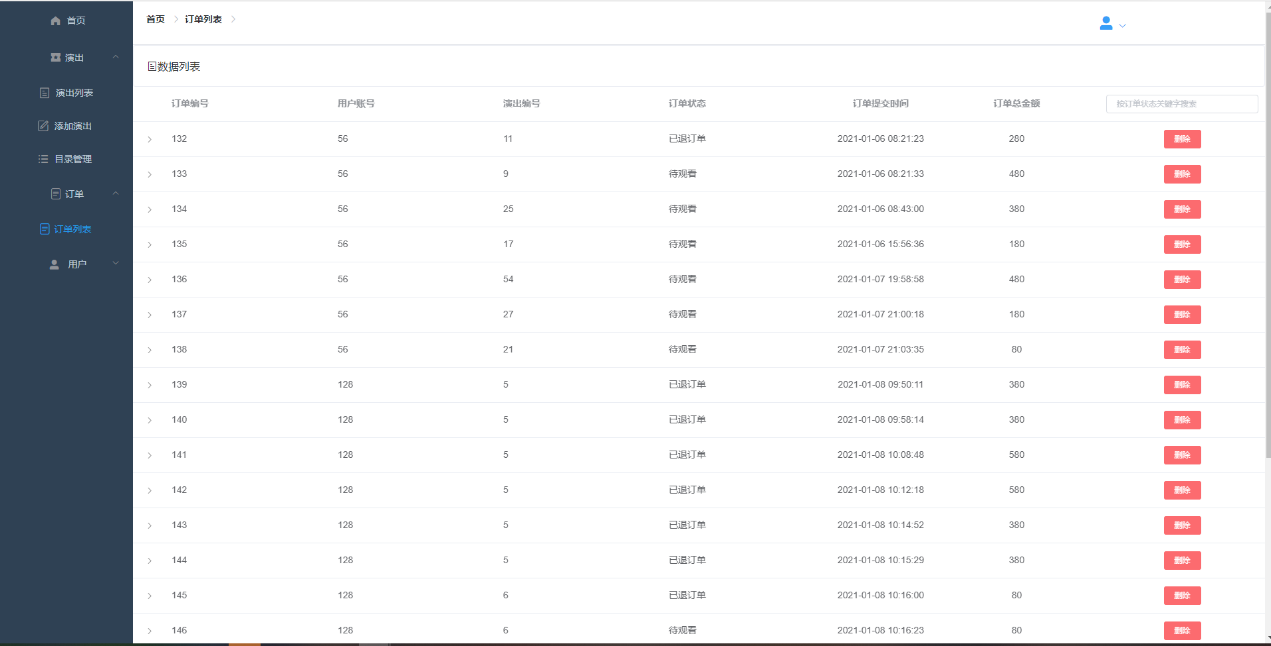


图 2.13原型设计——后台订单管理

1. 用户管理

图形用户界面, 应用程序

描述已自动生成

图 2.14原型设计——后台用户管理

模块导航建模：

图示

描述已自动生成

图 2.15后台管理模块导航建模

## 数据库设计

### 实体及其属性的介绍

<entity>用户：用来记录用户的账号以及个人信息，用户在注册账号时生成一个实例属性包括：用户ID、用户名、密码、手机号、邮箱、头像、真实姓名、身份证号、性别、年龄、创建时间、账号状态。主码为用户ID。

<entity>管理员：管理员与用户类似，其拥有用户拥有的所有属性与权限，但又独立于用户。管理员拥有整个系统的最高权限，以达到对系统进行管理的目的，包括用户管理，订单管理，演出管理，座次管理、报表管理等。属性包括：管理员ID、用户名、密码、头像、上一次登录时间。主码为：管理员ID。

<entity>目录：用来记录演出的分类信息，管理员可以动态的增加目录，管理目录、设置目录显示优先级。属性包括：目录ID、目录名称、目录优先级、目录图标、目录描述、目录删除状态。主码为：目录ID。

<entity>演出：用来记录所有的演出信息。只能由管理员进行发布与管理，普通用户只有对演出的查看功能。属性包括：演出ID、所属的目录ID、演出名称、演出海报、演出详细描述、演出所在城市、该演出展示的优先级、具体演出地址、演出开始日期、演出结束日期。主码为：演出ID。

<entity>演出座次：用来记录所有演出的演出座次等级，每一场演出都会有几个座次，例如A，B，C三等座。不同座次属于同一个演出，但是他们的容量与价格都是不一样的。属性包括：座次等级ID、所属演出Id、所属级别（所属哪个级别,例如"外场票", "内场票"等）、演出场次容量、该级别座位的定价、该级别座位的库存。主码为：演出座次ID。

<entity>订单：订单是在用户对某个演出下单所产生的，一个用户可以有很多个订单，但是每个订单只能对应于某个演出。其中每个订单可以包含很多张票信息，即包含不同的座次。属性包括：订单ID、所属用户Id、所属演出Id、订单状态（待观看，已完成,已退订单）、订单时间、订单支付方式、该订单对用户是否可见（即用户是否删除了该订单）、订单总金额、订单所包含票的总数、删除状态（是否被管理员删除）。主码为：订单ID。

<entity>票：票是对于某个演出座次的信息，一个订单可能包含多张票，但是一张票只能对应于某个订单，一张票也只能对应于某个具体座次。属性包括：票ID、所属订单Id、所属座次Id、二维码图片（供观影人验证入场） 主码为：票ID

### 表结构

**BOYING\_USER**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
|  | USERNAME | NVARCHAR2(255) NOT NULL | 用户名 |
|  | PASSWORD | NVARCHAR2(255) NOT NULL | 密码 |
|  | PHONE | NVARCHAR2(255) NOT NULL | 联系电话 |
|  | EMAIL | NVARCHAR2(255) NOT NULL | 邮箱 |
|  | ICON | NVARCHAR2(255) NOT NULL | 个人头像 |
|  | REAL\_NAME | NVARCHAR2(255) NOT NULL | 用户真实姓名 |
|  | IDENTITY\_NUMBER | NVARCHAR2(255) NOT NULL | 用户身份证号 |
|  | AGE | NUMBER(9) NOT NULL | 年龄 |
|  | CREATE\_TIME | DATE NOT NULL | 创建时间 |
|  | GENDER | NUMBER(9) DEFAULT 0 NOT NULL | 用户性别 1->男,0->女 |
|  | ADMIN\_DELETE | NUMBER(9) DEFAULT 0 NOT NULL | 管理员是否删除了该用户,默认为0，即未删除状态 |

**BOYING\_ADMIN**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
|  | USERNAME | NVARCHAR2(255) NOT NULL | 用户名 |
|  | PASSWORD | NVARCHAR2(255) NOT NULL | 密码 |
|  | ICON | NVARCHAR2(255) NOT NULL | 头像 |
|  | LAST\_TIME | DATE NOT NULL | 最后登录时间 |

**BOYING\_CATEGORY**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
|  | NAME | NVARCHAR2(255) NOT NULL | 目录名称 |
|  | ICON | NVARCHAR2(255) NOT NULL | 该目录的图标 |
|  | DESCRIPTION | NVARCHAR2(255) NOT NULL | 目录描述 |
|  | WEIGHT | NUMBER(9) NOT NULL | 用于排序,0则不显示 |
|  | ADMIN\_DELETE | NUMBER(9) DEFAULT 0 NOT NULL | 管理员是否删除了该目录,默认为0，即未删除状态 |

**BOYING\_SHOW**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
| **FK** | CATEGORY\_ID | NUMBER(9) NOT NULL | 所属的目录 |
|  | NAME | NVARCHAR2(255) NOT NULL |  |
|  | POSTER | NVARCHAR2(255) NOT NULL | 该演唱会的海报信息 |
|  | DETAILS | NVARCHAR2(255) NOT NULL | 存储该演唱会等的图文信息 |
|  | CITY | NVARCHAR2(255) NOT NULL | 演出所在城市 |
|  | ADDRESS | NVARCHAR2(255) NOT NULL | 具体演出地址 |
|  | START\_TIME | DATE NOT NULL | 演出开始时间 |
|  | END\_TIME | DATE NOT NULL | 演出结束时间 |
|  | WEIGHT | NUMBER(9) NOT NULL | 该演出展示的优先基本,0为不展示 |
|  | MIN\_PRICE | FLOAT NOT NULL | 该演出的最低价 |
|  | MAX\_PRICE | FLOAT NOT NULL | 该演出的最高价 |

**BOYING\_SEAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
| **FK** | SHOW\_ID | NUMBER(9) NOT NULL | 所属演出ID |
|  | NAME | NVARCHAR2(255) NOT NULL | 所属哪个级别,例如"A等票“，”B等票“ |
|  | PRICE | FLOAT NOT NULL | 该级别座位的定价 |
|  | CAPACITY | NUMBER(9) NOT NULL | 该级别座位的容量 |
|  | STOCK | NUMBER(9) NOT NULL | 库存 |

**BOYING\_ORDER**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
| **FK** | USER\_ID | NUMBER(9) NOT NULL | 所属用户ID |
| **FK** | SHOW\_ID | NUMBER(9) NOT NULL | 所属演出ID |
|  | STATUS | NUMBER(9) NOT NULL | 待观看，已完成，已取消(1，2，3) |
|  | TIME | DATE NOT NULL | 订单支付时间 |
|  | USER\_DELETE | NUMBER(9) DEFAULT 0 NOT NULL | 用户删除 |
|  | ADMIN\_DELETE | NUMBER(9) DEFAULT 0 NOT NULL | 管理员删除 |
|  | TICKET\_COUNT | NUMBER(9) NOT NULL | 票的总数 |
|  | MONEY | FLOAT NOT NULL | 订单总价格 |
|  | PAYMENT | NVARCHAR2(255) NOT NULL | 订单支付方式（支付宝，微信) |

**BOYING\_TICKET**

|  |  |  |  |
| --- | --- | --- | --- |
| **备注** | **字段名** | **数据类型** | **字段说明** |
| **PK** | ID | NUMBER(9) NOT NULL |  |
| **FK** | ORDER\_ID | NUMBER(9) NOT NULL | 所属订单ID |
| **FK** | SEAT\_ID | NUMBER(9) NOT NULL | 所属座次ID |
|  | QR\_CODE\_URL | NVARCHAR2(255) NOT NULL | 二维码图片,供观影人验证入场 |

### 数据库关系图

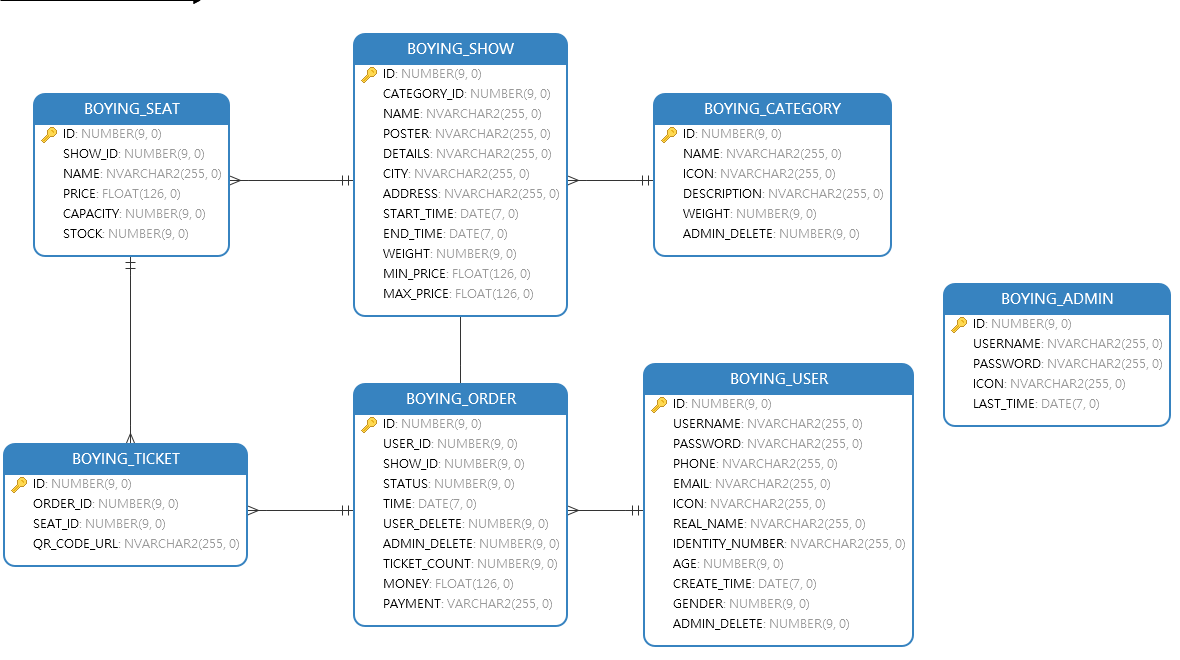


图 2.16数据库关系图

## 系统出错处理设计

### 出错信息

SUCCESS(200, "操作成功"),

FAILED(500, "操作失败"),

VALIDATE\_FAILED(404, "参数检验失败"),

UNAUTHORIZED(401, "暂未登录或token已经过期"),

FORBIDDEN(403, "没有相关权限");

# 系统软件结构

图片包含 图形用户界面

描述已自动生成

图 3.1系统软件结构图

# 类设计

## 用户模块

用户类图：

文本

描述已自动生成

图 4.1用户类图

属性：用户ID，账号，手机号，密码，真实姓名，身份证号，邮箱，年龄，性别，头像，创建时间，状态。

方法：账号密码登陆、手机号密码登陆、手机号验证码登陆、注册、生成验证码、跟新个人信息。

说明：用户的基本信息，以及一系列基本活动。被已登陆用户、未登陆用户与管理员用户继承，是最基础的角色类。用户的所有属性都是持久的，它存储了注册用户的基本信息。

### 模块描述

用户模块包括了登陆、注册，以及用户个人信息的管理，包括头像、邮箱、年龄、身份证号、性别、真实姓名。管理个人信息的功能需要用户登陆后才能使用。

### 功能

注册：用户可以选择注册一个个人账号。

登陆：登陆后可以购票、管理订单等。

管理个人信息：用户可以修改其个人信息包括修改头像、邮箱、年龄、身份证号、性别、真实姓名。

#### 注册的详细设计

输入项：注册用户用户名、注册用户密码、注册用户手机号、注册用户验证码、注册用户头像（可选）。

输出项：注册成功，请登录的提示信息。

##### 逻辑流程

图示

描述已自动生成

图 4.2注册逻辑流程

##### 接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户注册** |
| **URL** | /user/register |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **username** | true | string | 用户注册的账号名称 |
| **password** | true | string | 用户的密码 |
| telephone | true | string | 用户的电话号码 |
| **authCode** | true | string | 验证码 |
| icon | False | string | 用户可以上传头像 |

#### 登陆的详细设计

输入项：登陆用户账号、登陆用户密码或者登陆用户手机号、登陆用户密码或者登陆用户手机号、登陆用户验证码

输出项：主页

##### 逻辑流程

图示

描述已自动生成

图 4.3登录逻辑流程

##### 接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户用账号密码登陆** |
| **URL** | /usernameLogin |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **username** | true | string | 用户输入的账号名称 |
| **password** | true |  | 用户输入的密码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 将用户输入的密码加密后与数据库查到的user密码对比（密码错误则报错给前端）,在服务器内存中存储了该用户信息，返回token给前端. |

##### 手机号密码登陆接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户用手机号密码登陆** |
| **URL** | /telephoneLogin |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| telephone | true | string | 用户输入的手机号 |
| **password** | true |  | 用户输入的密码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 将用户输入的密码加密后与数据库查到的user密码对比（密码错误则报错给前端）,在服务器内存中存储了该用户信息，返回token给前端. |

##### 手机号验证码登陆接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户用手机号验证码登陆** |
| **URL** | /authCodeLogin |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| telephone | true | string | 用户输入的手机号 |
| authCode | true |  | 用户输入的验证码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Token** | string | 在服务器内存中存储了该用户信息，返回token给前端. |

##### 限制条件与出错处理

1）登录用户的账号和密码或者手机号和密码或者手机号和验证码应当匹配

2）如果用户输入的账号和密码或者手机号和密码或者手机号和验证码不匹配，显示登录失败的提示信息

#### 管理个人信息的详细设计

输入项：用户id、用户头像、用户年龄、用户邮箱、用户性别、用户身份证号、用户真实姓名。

输出项：显示修改成功的提示信息，并刷新个人信息页面。

##### 逻辑流程

图示

描述已自动生成

图 4.4管理个人信息逻辑流程

##### 接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **更新个人信息** |
| **URL** | /user/update |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 必选 | 类型 | 说明 |
| age | true | string | 用户年龄 |
| Email | True | String | 用户邮箱 |
| Gender | true | string | 用户性别 |
| identityNumber | true | string | 用户的身份证号 |
| Realname | true | string | 用户真实姓名 |
| icon | True | string | 用户头像 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| **Status** | string | 返回结果状态。 |

##### 限制条件与出错处理

1）用户上传头像只能上传jpg/jpeg/png文件，且不超过5MB。

2）若用户上传的头像文件不符合要求则提示上传头像失败。

3）限制用户需登陆才会在导航栏出现“我的信息”

## 演出模块

演出目录类：

文本

描述已自动生成

图 4.5演出目录类

属性：演出类型ID，目录名称，权重，图标，文字描述，状态。

方法：添加演出目录、删除演出目录

说明：便于对各种各样的演出进行统一管理。

演出类：

文本

描述已自动生成

图 4.6演出类

属性：演出ID，演出名，所属类型，海报，权重，城市，具体地址，开始日期，结束日期，详情。

方法：添加演出、删除演出

说明：演出的基本信息。演出类中所有属性都是持久化的，它负责存储演出的基本信息。

座次类：

表格

描述已自动生成

图 4.7座次类

属性：演出座次ID，所属演出ID，座次名称，库存，容量，定价。

方法：管理员创建座次、获取座次列表。

说明：演出的座次信息。

票类：

表格

描述已自动生成

图 4.8票类

属性：票ID，所属订单ID，所属座次ID，二维码图片。

方法：用户/管理员获取票的列表、用户添加票

说明：演出的票信息。

### 模块描述

该模块包括了用户浏览网站时的演出相关内容，具体包括搜索演出、查看演出详情、购票。该模块的所有功能必须要用户登陆才可使用。点击演出的标题或者海报等可以查看演出详情，进入演出详情页面可以购票。

### 功能

搜索演出：用户可以通过关键词、城市、时间、类别中的一个或者多个条件搜索对应的演出进行浏览。

查看演出详情：用户点击某个演出的标题或海报等后便可查看该演出的详细内容。

购票：用户可以在演出详情页面选择座次进行购票。

#### 搜索演出的详细设计

输入项：搜索演出类别、搜索演出关键词、搜索演出城市、搜索演出时间

输出项：演出搜索结果的页面

##### 逻辑流程

图示

描述已自动生成

图 4.9搜索演出逻辑流程

##### 接口

搜索筛选演出

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **演出的综合搜索筛选** |
| **URL** | /search |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| keyword | False | string | 用户想要搜索的演出关键词 |
| city | False | String | 用户想要搜索的演出城市 |
| categoryId | False | Int | 用户想要搜索的演出类别 |
| pageNum | False | Int | 分页数 |
| pageSize | False | Int | 每页条数 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| CommonResult<CommonPage<BoyingShow>> | string | 返回符合要求的演出并按权重降序排列 |

##### 限制条件与出错处理

1）限制所有搜索条件都为模式匹配的子字符串。

2）当搜索结果为空时，搜索结果页面显示空白。

3）限制用户在搜索前应登录账号。

4）如果用户未登录则直接跳转至登陆页面。

#### 查看演出详情的详细设计

输入项：用户所选择演出id

输出项：演出详情页面

##### 逻辑流程

图示

描述已自动生成

图 4.10查看演出详情逻辑流程

##### 接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **获取某个演出的详情** |
| **URL** | /show/detail/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **Id** | True | string | 想要查看详情的演出ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| CommonResult<BoyingShow> | string | 通过show\_id获取并返回该演出的详细信息 |

##### 限制条件与出错处理

1）限制用户在查看演出详情前应登录账号。

2）如果用户未登录则直接跳转至登陆页面。

#### 购票的详细设计

输入项：用户所选择演出id，用户所选择座次

输出项：生成相应订单

##### 逻辑流程

图示

描述已自动生成

图 4.11购票逻辑流程

##### 接口

用户下单接口：

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **用户购买门票** |
| **URL** | /order/add |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| showId | True | string | 演出ID |
| showSeatIds | True | string | 座次ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果状态。 |

##### 限制条件与出错处理

1）限制用户不能重复购买同一场次的票。

2）如果用户重复购买，则显示重复购买，购票失败的提示信息

## 订单模块

订单类：

文本

描述已自动生成

图 4.12订单类

属性：订单ID，所属用户ID，所属演出ID，订单状态，提交时间，用户删除状态，管理员删除状态，总金额，票数量。

方法：添加订单、删除订单、取消订单、完成订单。

说明：用户的订单信息。

### 模块描述

订单模块包括了订单的查看与管理，包括查看所有订单、查看待观看订单、查看已完成订单、查看已退订单、删除已退订单、取消待观看订单。与订单相关的所有的功能需要用户登陆后才能使用。

### 功能

查看订单：用户可以查看所有订单、查看待观看订单、查看已完成订单、查看已退订单。

查看订单详情：用户在订单列表中选择订单并点击查看订单按钮可查看订单详情。

删除订单：用户可以删除已退订单。

取消订单：用户可以取消待观看订单。

#### 查看订单的详细设计

输入项：用户选择查看所有订单、查看待观看订单、查看已完成订单、查看已退订单。

输出项：显示用户选择的相应订单列表。

##### 逻辑流程

图示

描述已自动生成

图 4.13查看订单逻辑流程

##### 接口

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **查看所有订单** |
| **URL** | /order/list |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 用户ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| list | string | 订单列表 |
| pageNum | Int | 分页数 |
| pageSize | int | 每页条数 |

##### 限制条件与出错处理

1. 限制用户需登陆才会在导航栏出现“我的订单”
2. 当用户无订单时，订单列表页面显示空白。

#### 查看订单详情的详细设计

输入项：订单ID

输出项：订单详情页面

##### 逻辑流程

图示

描述已自动生成

图 4.14查看订单详情逻辑流程

##### 接口

获取订单详情

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **查看订单详情** |
| **URL** | /order/details/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | GET |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 订单ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Data | string | 返回订单和票的信息 |

#### 删除订单的详细设计

输入项：订单ID

输出项：不显示被删除的订单，刷新订单列表

##### 逻辑流程

图示

描述已自动生成

图 4.15删除订单逻辑流程

##### 接口

删除订单

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **删除订单** |
| **URL** | /order/delete/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 订单ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果 |

#### 取消订单的详细设计

输入项：订单ID

输出项：被取消的待观看订单变为已退订单。

##### 逻辑流程

图示

描述已自动生成

图 4.16取消订单逻辑流程

##### 接口

取消订单

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **取消订单** |
| **URL** | /order/cancel/{id} |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| Id | True | string | 订单ID |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果 |

## 管理员管理模块

管理员分析类图

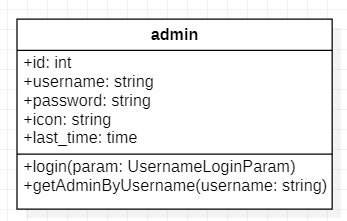


图 4.17管理员类

属性：ID、管理员名称username、密码password、图标icon、上次登陆时间last\_time

方法：管理员登陆、查询并获取管理员信息（其他继承自user类）

说明：管理员的基础信息及管理方法。对各项信息的管理接口集成在了各模块中（限制因素为用户身份），故管理员类下有登陆方法（登陆后台管理时使用）及获取管理员信息（进行管理）。

### 模块描述

管理员模块包含管理员信息属性、管理员的登陆方法及管理员获取其他管理员信息的方法。

### 功能

管理员登陆：根据账号密码登陆

获取管理员信息：根据用户名参数获取管理员信息（进行管理员管理）

#### 管理员登陆的详细设计

输入项：登陆信息（由前端页面输入账号密码后生成）

输出项：登陆认证结果

##### 逻辑流程

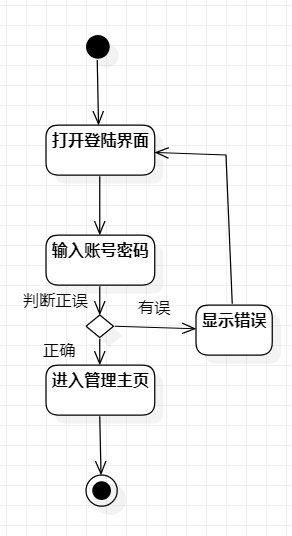


图 4.18管理员登录逻辑流程

##### 接口

管理员登陆

基本信息：

|  |  |
| --- | --- |
| **功能描述** | **取消订单** |
| **URL** | /login/login |
| **支持格式** | JSON |
| **HTTP请求方式** | POST |

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **必选** | **类型** | **说明** |
| **Username** | True | string | 管理员账号名 |
| Password | True | string | 管理员密码 |

返回字段：

|  |  |  |
| --- | --- | --- |
| **返回字段** | **字段类型** | **说明** |
| Message | string | 返回结果 |

# 附录A 图目录

[图 2.1系统总体架构设计 2](#_Toc61820900)

[图 2.2系统软件结构设计图 3](#_Toc61820901)

[图 2.3原型设计——主页 11](#_Toc61820902)

[图 2.4原型设计——搜索页 12](#_Toc61820903)

[图 2.5原型设计——订单页 12](#_Toc61820904)

[图 2.6原型设计——用户信息页 13](#_Toc61820905)

[图 2.7原型设计——演出详情页 13](#_Toc61820906)

[图 2.8前台演出购票模块导航建模 14](#_Toc61820907)

[图 2.9原型设计——后台主页 14](#_Toc61820908)

[图 2.10原型设计——后台演出列表 15](#_Toc61820909)

[图 2.11原型设计——后台添加演出 15](#_Toc61820910)

[图 2.12原型设计——后台目录管理 16](#_Toc61820911)

[图 2.13原型设计——后台订单管理 16](#_Toc61820912)

[图 2.14原型设计——后台用户管理 16](#_Toc61820913)

[图 2.15后台管理模块导航建模 17](#_Toc61820914)

[图 2.16数据库关系图 22](#_Toc61820915)

[图 3.1系统软件结构图 23](#_Toc61820916)

[图 4.1用户类图 24](#_Toc61820917)

[图 4.2注册逻辑流程 26](#_Toc61820918)

[图 4.3登录逻辑流程 28](#_Toc61820919)

[图 4.4管理个人信息逻辑流程 32](#_Toc61820920)

[图 4.5演出目录类 34](#_Toc61820921)

[图 4.6演出类 34](#_Toc61820922)

[图 4.7座次类 35](#_Toc61820923)

[图 4.8票类 35](#_Toc61820924)

[图 4.9搜索演出逻辑流程 36](#_Toc61820925)

[图 4.10查看演出详情逻辑流程 38](#_Toc61820926)

[图 4.11购票逻辑流程 40](#_Toc61820927)

[图 4.12订单类 42](#_Toc61820928)

[图 4.13查看订单逻辑流程 43](#_Toc61820929)

[图 4.14查看订单详情逻辑流程 45](#_Toc61820930)

[图 4.15删除订单逻辑流程 46](#_Toc61820931)

[图 4.16取消订单逻辑流程 48](#_Toc61820932)

[图 4.17管理员类 49](#_Toc61820933)

[图 4.18管理员登录逻辑流程 51](#_Toc61820934)