Tickets网站

J2EE大作业文档

151250200 软件学院 郑皓铭

2018

目录

[一、 数据库设计 2](#_Toc511037980)

[1. ER图 2](#_Toc511037981)

[2. 数据库表 3](#_Toc511037982)

[二、 架构设计 8](#_Toc511037983)

[1. 工程的项目结构 8](#_Toc511037984)

[2. 后端框架 9](#_Toc511037985)

[3. 前端框架 9](#_Toc511037986)

[三、 类设计 9](#_Toc511037987)

[1. 类与职责 9](#_Toc511037988)

[2. 前端页面与功能 10](#_Toc511037989)

[四、其他 11](#_Toc511037990)

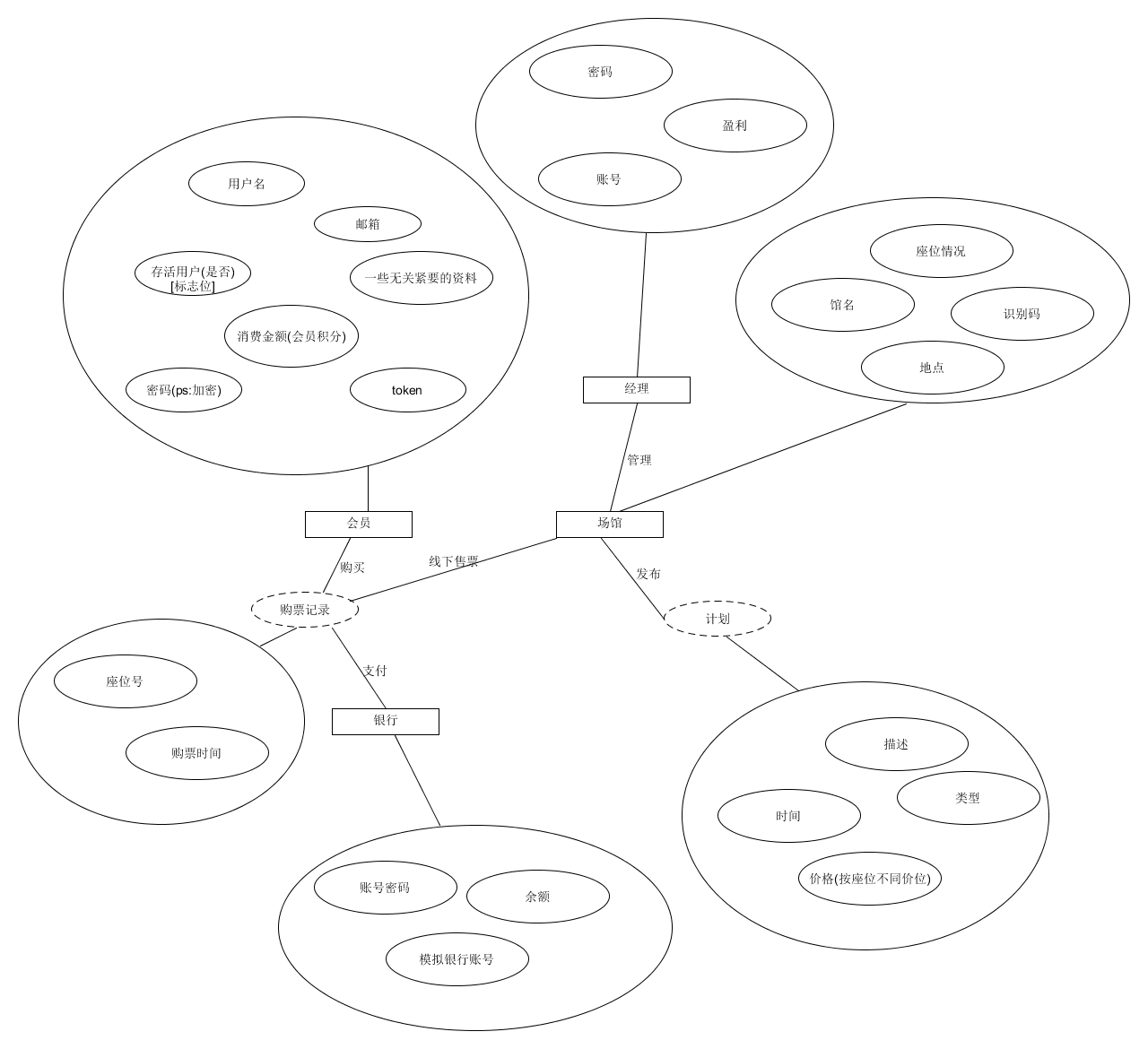
[1. 开发环境 11](#_Toc511037991)

[2. 开发心得体会 11](#_Toc511037992)

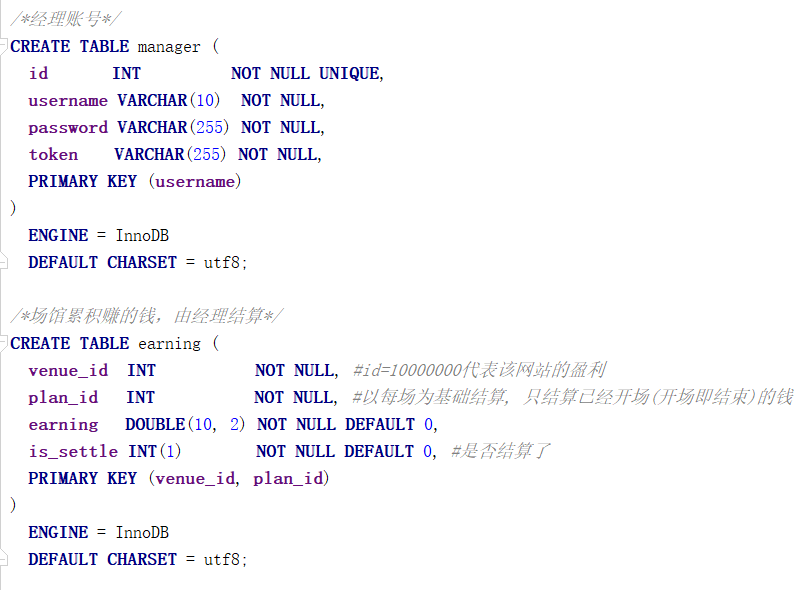
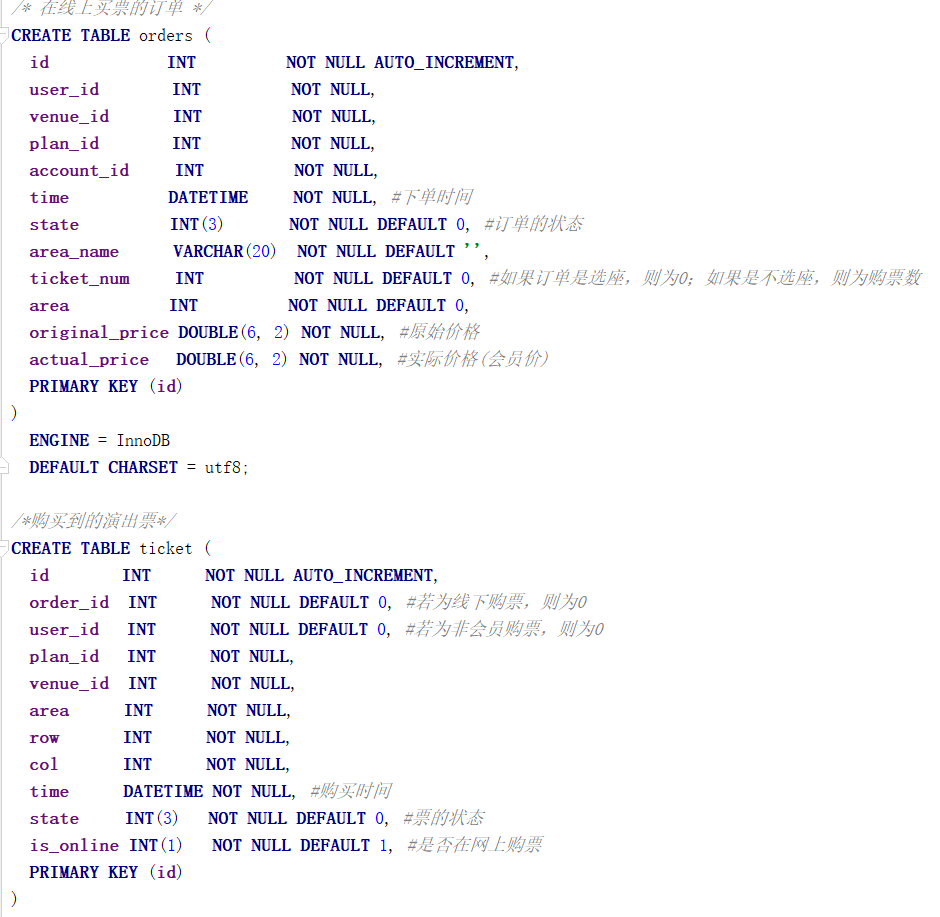
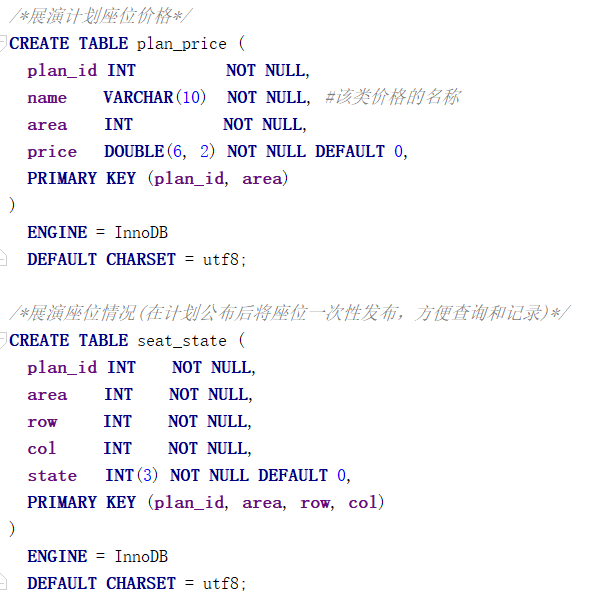
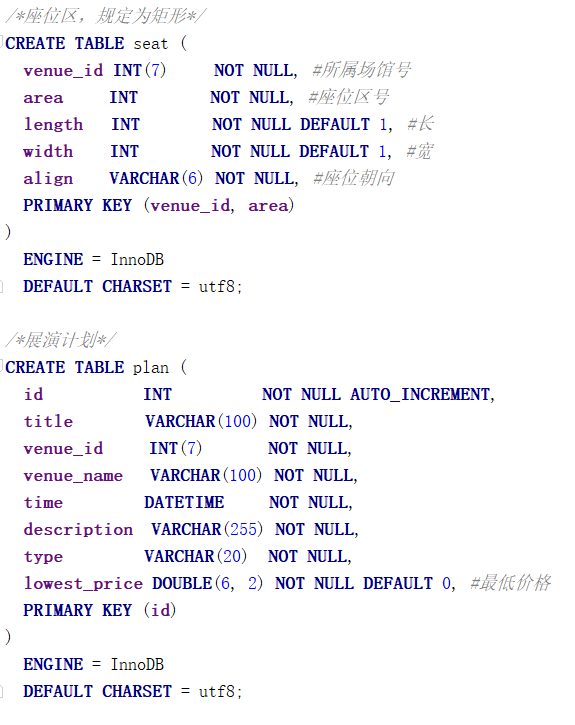
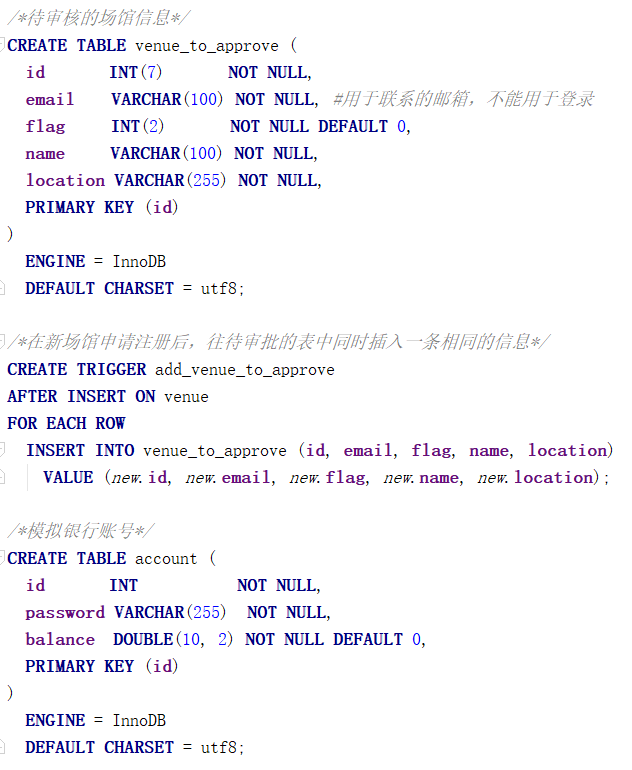
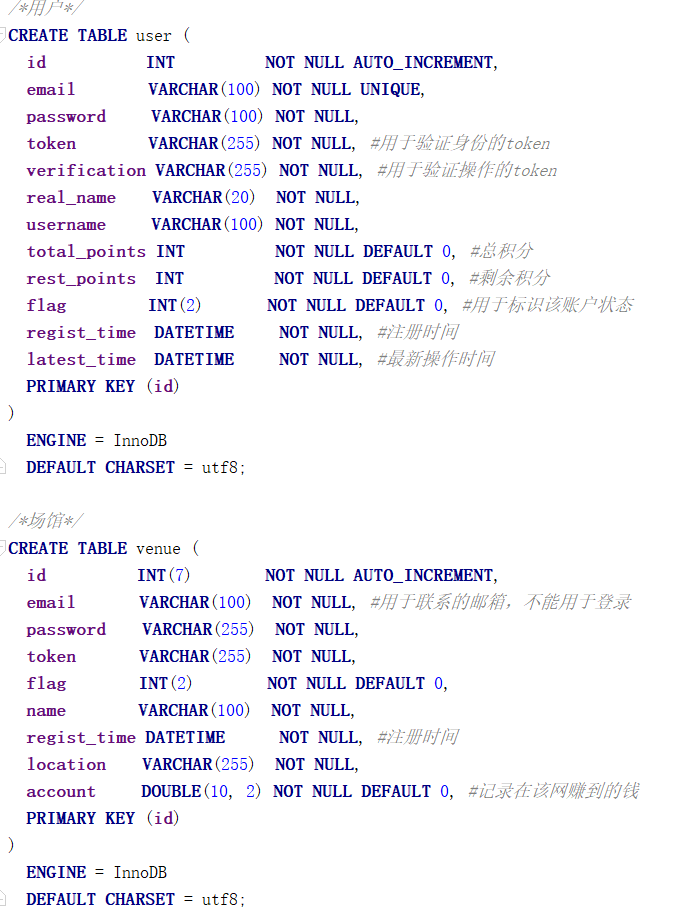
[3. 补充 11](#_Toc511037993)

# 数据库设计

## ER图

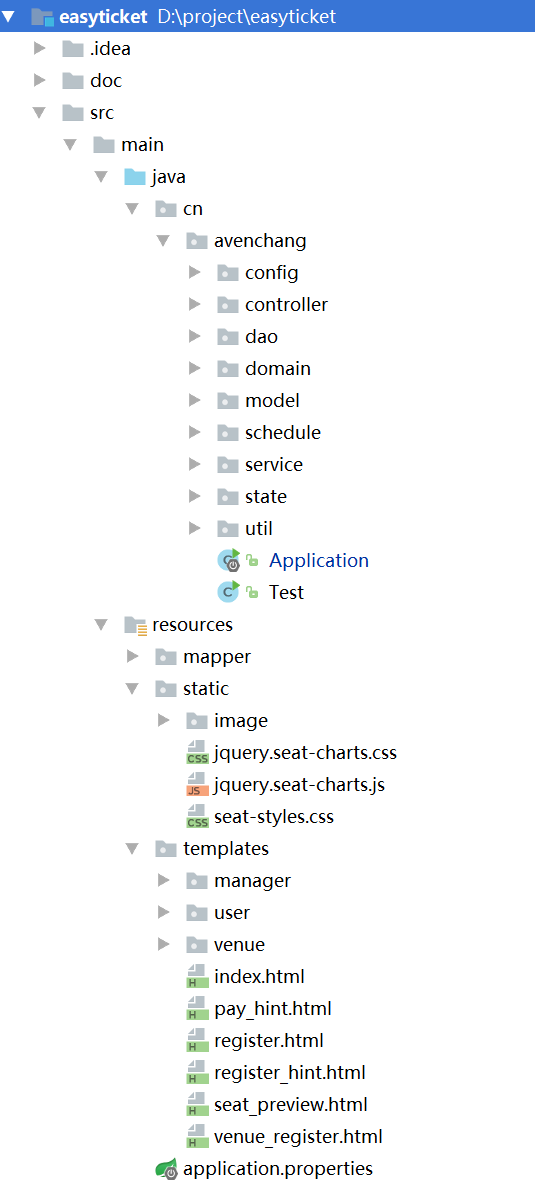


## 数据库表



# 架构设计

## 工程的项目结构



## 后端框架

SpringBoot+MyBatis

## 前端框架

Bootstrap

# 类设计

## 类与职责

|  |  |  |
| --- | --- | --- |
| 包名 | 类名 | 职责 |
| config | Constant | 项目用到的常亮定义 |
| DefaultConfig | Application配置文件中自定义的赋值以及一些规则定义 |
| EmailConfig | Application配置文件中定义的与发邮件相关的赋值 |
| SecurityConfig | 安全相关的过滤器 |
| TokenVerifyInterceptor | 检验当前角色是否能访问该页面，以及检测用户的token是否有效 |
| controller | LoginController | 与登录相关的控制器 |
| MainController | 主控制器，主要用于分发操作 |
| ManagerController | 与经理操作相关的控制器 |
| PaymentController | 与支付相关的控制器 |
| UserController | 与会员操作相关的控制器 |
| VenueController | 与场馆操作相关的控制器 |
| dao | [table]Dao | 与[table]表有关的数据库操作 |
| domain | [table] | [table]中的属性与mysql中[table]的列相关联 |
| model | LiveSeatMap | 实时座位表 |
| OrderInfo | 会员查看订单时所需要的相关信息 |
| PlanInfo | 会员和场馆查看计划时所需要的信息 |
| Profit | 优惠券 |
| ResultMessage | 逻辑层返回给控制层的结果封装 |
| SeatInfo | 座位信息 |
| UserDiscount | 会员的折扣优惠 |
| UserInfo | 会员查看个人信息时所需的信息 |
| ValidInfo | 登录成功后返回的验证信息 |
| VenueEarning | 场馆某计划赚取的盈利 |
| VenueInfo | 某场馆查看自己场馆资料所需的信息 |
| VenueStatistic | 该场馆的统计信息 |
| VenueUpdate | 场馆信息修改前后的值 |
| WebStatistic | 网站的统计信息 |
| schedule | AllocateTicketScheduler | 用于定期配票的Scheduler |
| service | EmailService | 与邮件操作相关的Service |
| LoginService | 与登录操作相关的Service |
| ManagerService | 与经理操作相关的Service |
| UserMangeService | 与会员操作相关的Service |
| VenueManageService | 与场馆操作相关的Service |
| PaymentService | 与支付操作相关的Service |
| PlanService | 与计划操作相关的Service |
| TicketService | 与票操作相关的Service |
| SeatService | 与座位操作相关的Service |
| state | OrderState | 订单状态的Enum |
| Role | 所有角色 |
| SeatState | 座位状态的Enum |
| TicketState | 票务状态的Enum |
| UserState | 会员账号状态的Enum |
| VenueState | 场馆账号状态的Enum |
| util | MD5 | 加密工具 |
| PointsUtil | 用于对会员积分计算的工具 |
| SeatUtil | 将座位对象转换成网页展示座位所需信息的工具 |
| TimeUtil | 与时间有关操作的工具 |
| TicketSynchronizer | 所有的购票操作都需要线程安全 |
|  | Application | 启动类 |
|  | Test | 用于测试一些操作 |

## 前端页面与功能

|  |  |  |
| --- | --- | --- |
| 包名 | 页面名 | 功能 |
| Manager | Approve\_list | 审批场馆的注册和修改 |
| Settle\_list | 计算场馆的盈利 |
| Web\_statistic | 网站的相关统计 |
| Venue | Check\_in | 检票入场 |
| Choose\_seat | 线下售票选座 |
| My\_plan | 该场馆发布的计划 |
| New\_plan | 发布一个新的计划 |
| Venue\_info | 该场馆的相关信息 |
| Venue\_statistic | 该场馆的相关统计 |
| User | Choose\_seat | 用户选座购票 |
| My\_info | 该用户的相关资料 |
| My\_order | 该用户的订单 |
| Order\_unpaid | 某未支付订单的详细信息 |
| Pay | 支付页面 |
| Plan\_detail | 某计划的详细信息 |
| Plan\_list | 所有可购票的计划 |
| 无角色 | Index | 首页，也为登录页面 |
| Pay\_hint | 支付后的反馈页面 |
| Register | 用户注册页面 |
| Venue\_register | 场馆注册页面 |
| Register\_hint | 注册后的反馈页面 |
| Seat\_preview | 场馆注册时预览座位设置的页面 |

# 四、其他

## 1. 开发环境

数据库: Mysql

服务器: Tomcat

## 2. 开发心得体会

这应该是第一次个人的全栈开发，以往都是写后端，没有真正接触过前端以及控制器的东西。这次作业让我有机会尝试到前端的开发。这是第一次真正意义上的使用MVC开发项目，也让我对MVC有了更深一步的了解。虽然对前端只是接触了一点，但是对控制器认识更加深入，以后开发就知道前端需要什么，我应该传什么样的数据到前端。

同时对springboot的框架学习了不少，也新学习了持久化框架Mybatis，这是一次不错的开发体验。

## 补充

关于项目的有关文档都放在了doc目录下，如果对该文档还有不清楚的地方，可以进入该目录进行更详细的查看。