刘钰 151250101

简单文档说明

一、数据库设计 3

1、ER图 3

2、数据库表（表名，各列名及数据类型，外键关系） 4

二、架构设计 9

1、工程的项目结构截图 9

2、是否使用框架，如Spring+Hibernate 架构 10

3、前端页面是否使用框架，如bootstrap框架 10

三、类设计 10

1、各包的类：名称及职责 10

2、前端的各页面：名称及功能 12

四、其他 13

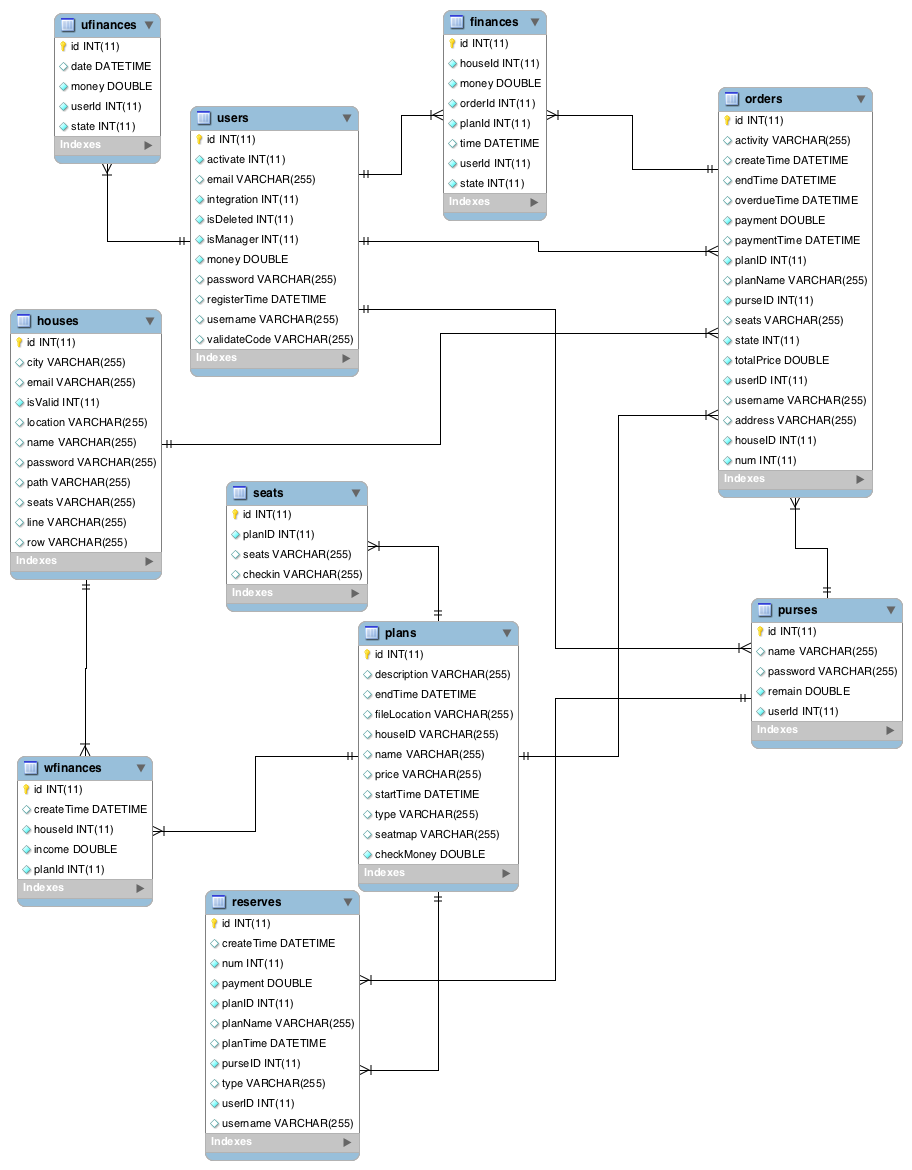
1、开发环境（数据库，服务器等） 13

2、开发心得体会 13

3、其他需要补充的 13

# 一、数据库设计

## 1、ER图



## 2、数据库表（表名，各列名及数据类型，外键关系）

users表（用户信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| activate | int(11) |  |
| email | varchar(255) |  |
| integration | int(11) |  |
| isDeleted | int(11) |  |
| isManager | int(11) |  |
| money | double |  |
| password | varchar(255) |  |
| registerTime | datetime |  |
| username | varchar(255) |  |
| validateCode | varchar(255) |  |

purses表（用户支付账户）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| name | varchar(255) |  |
| password | varchar(255) |  |
| remain | double |  |
| userId | int(11) | users表的外键 |

houses表（场馆信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| city | varchar(255) |  |
| email | varchar(255) |  |
| isValid | int(11) |  |
| location | varchar(255) |  |
| name | varchar(255) |  |
| password | varchar(255) |  |
| path | varchar(255) |  |
| seats | varchar(255) |  |
| line | varchar(255) |  |
| row | varchar(255) |  |

plans表（计划信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| description | varchar(255) |  |
| endTime | datetime |  |
| fileLocation | varchar(255) |  |
| houseID | varchar(255) | houses表的外键 |
| name | varchar(255) |  |
| price | varchar(255) |  |
| startTime | datetime |  |
| type | varchar(255) |  |
| seatmap | varchar(255) |  |
| checkMoney | double |  |

seats表（座位信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| planID | int(11) |  |
| seats | varchar(255) |  |
| checkin | varchar(255) |  |

orders表（订单信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| activity | varchar(255) |  |
| createTime | datetime |  |
| endTime | datetime |  |
| overdueTime | datetime |  |
| payment | double |  |
| paymentTime | datetime |  |
| planID | int(11) | plans表的外键 |
| planName | varchar(255) |  |
| purseID | int(11) | purses表的外键 |
| seats | varchar(255) |  |
| totalPrice | double |  |
| userID | int(11) | users表的外键 |
| usename | varchar(255) |  |
| address | varchar(255) |  |
| houseID | int(11) | houses表的外键 |
| num | int(11) |  |

reserves表（预订信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| createTime | datetime |  |
| endTime | datetime |  |
| overdueTime | datetime |  |
| payment | double |  |
| paymentTime | datetime |  |
| planID | int(11) | plans表的外键 |
| planName | varchar(255) |  |
| purseID | int(11) | purses表的外键 |
| seats | varchar(255) |  |
| totalPrice | double |  |
| userID | int(11) | users表的外键 |
| usename | varchar(255) |  |
| num | int(11) |  |

finances表（场馆收入财务信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| houseId | int(11) | houses表的外键 |
| money | double |  |
| orderId | int(11) | orders表的外键 |
| planId | int(11) | plans表的外键 |
| time | datetime |  |
| userId | int(11) | users表的外键 |
| state | int(11) |  |

wfinances表（网站收入财务信息）

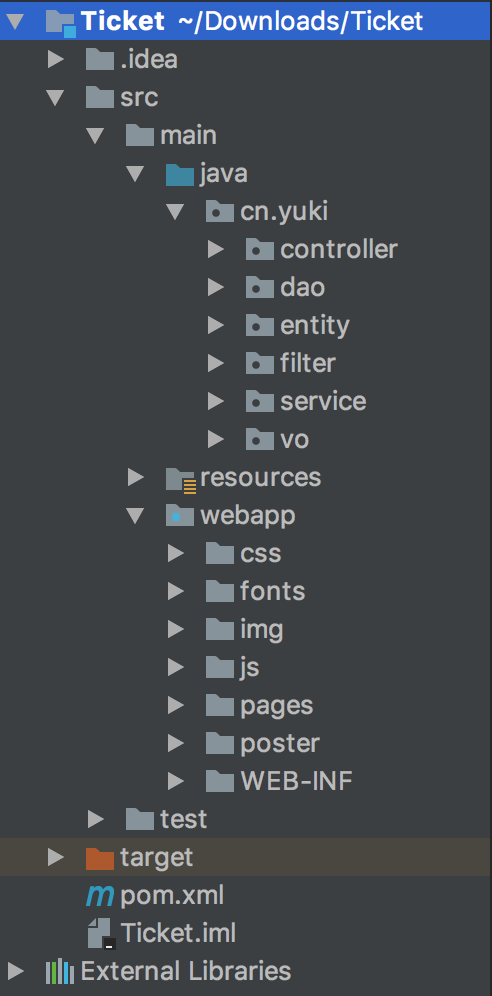
|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| createTime | datetime |  |
| houseId | int(11) | houses表的外键 |
| income | double |  |
| planId | int(11) | plans表的外键 |

ufinances表（用户支出财务信息）

|  |  |  |
| --- | --- | --- |
| 列名 | 数据类型 | 外键关系 |
| id | int(11) |  |
| date | datetime |  |
| money | double |  |
| userId | int(11) | users表的外键 |
| state | int(11) |  |

# 二、架构设计

## 1、工程的项目结构截图



说明:

|  |  |
| --- | --- |
| 包名 | 说明 |
| controller | 控制器层 |
| entity | 数据模型层 |
| dao | 数据访问层 |
| service | 业务逻辑层 |
| filter | servlet过滤器 |
| vo | 提供给界面显示的封装数据对象包 |

## 2、是否使用框架，如Spring+Hibernate 架构

使用SpringMVC+Hibernate 架构

该系统使用的架构是MVC架构，分为Controller、View（JSP页面，对应webapp/pages目录）、Model（对应entity包）。除了这三个包以外，还增加了Service包和Dao包，Service包用于进行业务逻辑的处理，简化Controller的业务逻辑，Dao包用于和数据库以及Model的交互。

## 3、前端页面是否使用框架，如bootstrap框架

使用bootstrap框架

# 三、类设计

## 1、各包的类：名称及职责

controller包

|  |  |
| --- | --- |
| 名称 | 职责 |
| HouseController | 分发处理场馆发出的前端请求 |
| ManagerController | 分发处理经理发出的前端请求 |
| UserController | 分发处理用户发出的前端请求 |
| MainController | 分发处理127.0.0.1:8080这个请求 |

dao包

|  |  |
| --- | --- |
| 名称 | 职责 |
| HouseDAO | 场馆信息的增改查 |
| HouseFinanceDAO | 场馆财务信息的增查 |
| OrderDAO | 订单信息的增删改查 |
| PlanDAO | 计划信息的增删改查 |
| PurseDAO | 用户支付账户信息的改查 |
| ReserveDAO | 座位预订信息的增查 |
| SeatDAO | 计划的座位信息的增改查 |
| UserDAO | 用户信息的增删改查 |
| UserFinanceDAO | 用户财务信息的增查 |
| WebsiteFinanceDAO | 网站财务信息的增删改查 |

注：dao.daoImpl包中是具体实现

entity包

|  |  |
| --- | --- |
| 名称 | 职责 |
| House | 场馆信息的实体类 |
| HouseFinance | 场馆财务信息的实体类 |
| Order | 订单信息的实体类 |
| OrderDelayItem | 用于处理3分钟后订单失效的包装类 |
| Purse | 用户支付账户信息的实体类 |
| Reserve | 座位预订信息的实体类 |
| Seat | 计划的座位信息的实体类 |
| User | 用户信息的实体类 |
| UserFinance | 用户财务信息的实体类 |
| VIP | 用户会员信息的实体类 |
| VIPList | 从配置文件中读取积分与等级的对应关系 |
| WebsiteFinance | 网站财务信息的实体类 |

filter包

|  |  |
| --- | --- |
| 名称 | 职责 |
| CharFormatFilter | 设置jsp页面utf-8格式 |

service包

|  |  |
| --- | --- |
| 名称 | 职责 |
| HouseService | 场馆相关的业务逻辑 |
| ManagerService | 经理相关的业务逻辑 |
| OrderService | 单独的线程，负责订单相关的业务逻辑 |
| PlanService | 计划相关的业务逻辑 |
| UserService | 用户相关的业务逻辑 |

注：service.serviceImpl包中是具体实现

其中多出的DeliverOrderQtz是执行Quartz插件的定时任务方法的类

service.util包

|  |  |
| --- | --- |
| 名称 | 职责 |
| DateUtil | 日期相关的工具类 |
| MailUtil | 发送邮件相关的工具类 |
| MD5 | MD5加密的工具类 |
| RandomNumber | 产生随机数的工具类 |
| Result | enum值，负责处理返回值 |

vo包

|  |  |
| --- | --- |
| 名称 | 职责 |
| HouseFinanceRatioVO | 界面显示场馆财务的比例信息 |
| HouseFinanceVO | 界面显示场馆的财务信息 |
| PlanFinanceVO | 界面显示计划的财务信息 |
| PlanVO | 界面显示计划信息 |
| UserFinanceVO | 界面显示用户财务信息 |

## 2、前端的各页面：名称及功能

house目录下

|  |  |
| --- | --- |
| 名称 | 功能 |
| apply.jsp | 场馆申请 |
| checkin.jsp | 场馆检票 |
| house.jsp | 场馆登录 |
| housefinance.jsp | 场馆财务信息 |
| househeader.jsp | 场馆导航栏，重用在其他页面上 |
| houseindex.jsp | 场馆信息 |
| houseorder.jsp | 场馆订单 |
| houseplandetail.jsp | 场馆计划详细信息，可以进行会员或非会员购票 |
| houseplanlist.jsp | 场馆计划列表 |
| planmake.jsp | 场馆制定计划 |

manager目录下

|  |  |
| --- | --- |
| 名称 | 功能 |
| check.jsp | 支付结算 |
| fiinfo.jsp | 网站财务信息 |
| houseinfo.jsp | 审批场馆 |

user目录下

|  |  |
| --- | --- |
| 名称 | 功能 |
| header.jsp | 用户导航栏，重用在其他页面上 |
| index.jsp | 主界面，含有各种计划的分类 |
| islogin.jsp | 判断是否已经登录 |
| login.jsp | 用户登录 |
| order.jsp | 用户订单 |
| plandetail.jsp | 计划详细，可以进行在线选座或不选座购票 |
| planlist.jsp | 计划列表 |
| profile.jsp | 个人信息 |
| register.jsp | 注册 |
| seat.jsp | 在线选座的详细信息 |

# 四、其他

## 1、开发环境（数据库，服务器等）

数据库采用了Mysql

服务器采用了tomcat

## 2、开发心得体会

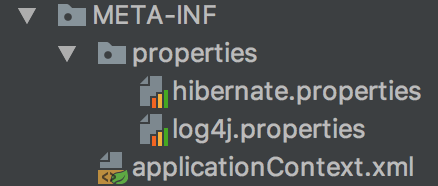
已整理成博客

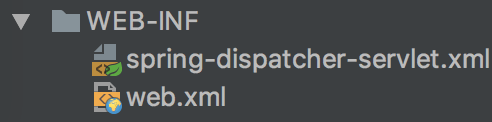
https://sweetstreet.github.io/blog/J2EE-homework-Tickets/

## 3、其他需要补充的

该项目是一个Maven项目，方便管理依赖包以及构建项目。

项目中的配置文件如下所示：

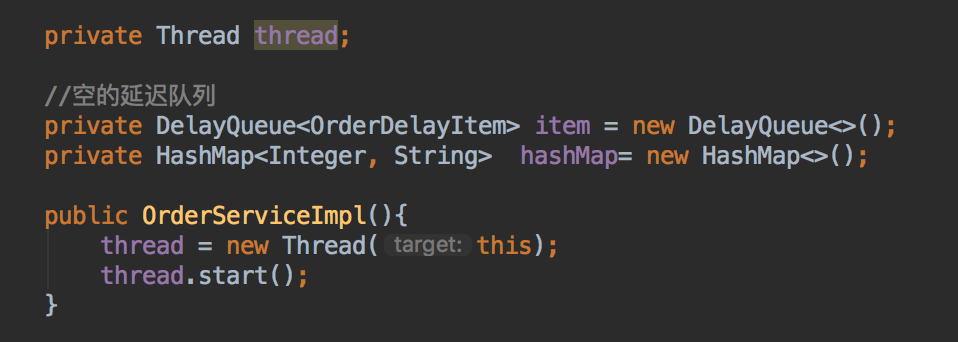




web.xml 是Java Web项目的配置文件

mvc-dispatcher-servlet.xml是Spring MVC的配置文件

15分钟自动取消订单，采用了DelayQueue，开了一个单独的线程负责处理这个



演出开始前2周自动售票，采用了Quartz插件，每天0点定时检查数据库，如果有演出符合要求，就将数据库中预订信息转成相应的订单

