



华中农业大学

HUAZHONG AGRICULTURAL UNIVERSITY

《HZAU 国际影城》

软件详细设计说明书

2020-06

## 版本变更历史

版本	提交日期	主要编制人	审核人	版本说明
V1.0	2020/06/02	彭士杰	徐浩春	初始版本
V2.0	2020/06/10	彭士杰	徐浩春	最终版本，对 1.0 进行了完善

## 任务分工

彭士杰	2018317210209	组长，主要负责编写目的、背景、定义、参考资料、需求概述、文档编制，说明书各部分改进完善等
常家乐	2018317210201	主要负责性能部分，模块基本信息、模块设计及模块接口、模块处理逻辑、算法部分负责用户电影评论模块、管理员订单查询模块、管理员影片推荐模块等
徐浩春	2018317210219	主要负责软件结构，说明书审核等，模块基本信息、模块设计及模块接口、模块处理逻辑、算法部分负责退订票模块、订单管理模块等
陶 威	2018317210207	主要负责模块基本信息、模块设计及模块接口、模块处理逻辑、算法部分的管理员票房数据管理&结算管理模块等
王南松	2018317210208	主要负责测试计划部分，模块基本信息、模块设计及模块接口、模块处理逻辑、算法部分负责用户浏览影片模块、管理员影片录入模块、管理员排片模块等
李铜平	2018317210206	主要负责模块设计改进完善，模块基本信息、模块设计及模块接口、模块处理逻辑、算法部分负责用户个人信息管理模块、用户会员办理模块、管理员会员管理模块等

# 目录

1 引言 .....	1
1.1 编写目的 .....	1
1.2 项目背景 .....	1
1.3 定义 .....	1
1.4 参考资料 .....	2
2 总体设计 .....	3
2.1 需求概述 .....	3
2.2 软件结构 .....	4
3 模块描述 .....	5
3.1 模块基本信息 .....	5
3.2 模块设计及模块接口 .....	8
3.3 模块处理逻辑 .....	23
3.4 算法 .....	39
3.6 性能 .....	40
3.7 测试计划 .....	42

# 1 引言

## 1.1 编写目的

本文档描述 HZAU 国际影城项目每个模块的细节设计，包括模块的接口、调用关系、处理过程和算法，以及模块测试方案等。

本模块的主要读者为软件设计人员、模块开发人员、管理人员、测试人员

## 1.2 项目背景

用 户	:	全校学生和教师
项 目 名 称	:	HZAU 国际影城
项 目 的 提 出 者	:	华中农业大学 HZAU 国际影城学习小组
开 发 单 位	:	华中农业大学信息学院
项 目 实 施 单 位	:	华中农业大学 HZAU 国际影城
与 其 他 系 统 的 关 系	:	本系统与国家授权票务数据平台存在数据交换

## 1.3 定义

售票	票	:	影院对观众观影的销售行为，电影票为收费凭证。
退票	票	:	因为某种原因取消观看电影而引起的退还票款行为。
影片	片	:	供观众当场观看的内容。
场次	次	:	一部影片在一个影厅的一次完整放映过程。影片所对应的放映时间，也称“计划”。在场次结束后，不能再做补登以外的任何票务操作。
影厅	厅	:	电影放映的银幕位置，和“场次”共同确定影片放映的具体地点。
影片编码	码	:	采用 12 位编码，由国家广播电影电视总局提供并在政府网站发布，具有唯一性。
售票原则	则	:	系统根据影院座位数，实行一人一票原则。
计数原则	则	:	对观影人次进行计数，每部电影每一名观众计一人次，一名观众看三部电影计三人次，以此类推。
统计数据上报		:	将票务数据以营业日为统计单位依照国家标准规定的格式传送到授权管理机构电影票务数据平台的行为。
原始数据上报		:	将售票原始数据依照国家标准规定的格式传送到授权管理机构电影票务数据平台的行为。

## 1.4 参考资料

- [1] 《关于转发<电影院票务系统（软件）管理实施细则>的通知》  
(2014 年 7 月 21 日 影字[2014]407 号)  
<http://dy.chinasarft.gov.cn/html/www/article/2014/01493ff2d427339b402881a7470edaf0.html>
- [2] 窦万峰. 软件工程方法与实践[M]. 北京: 机械工业出版社, 2009
- [3] HZAU 国际影城软件需求规格说明书.V2.0
- [4] HZAU 国际影城软件概要设计说明书.V1.0

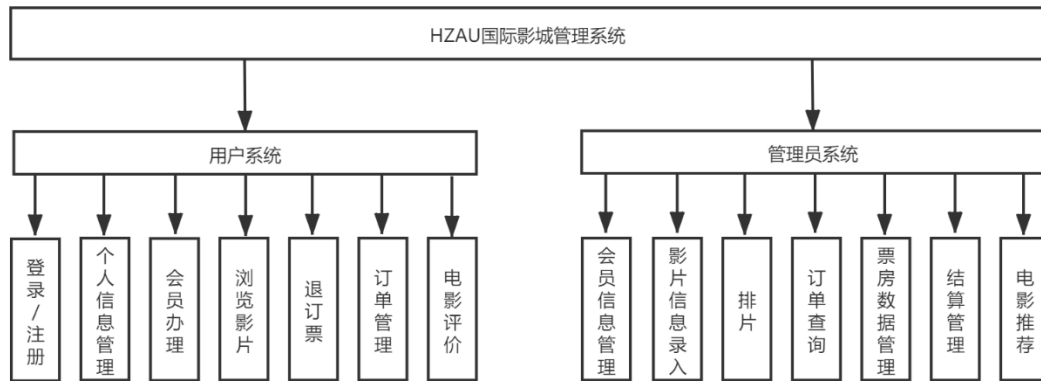
## 2 总体设计

### 2.1 需求概述

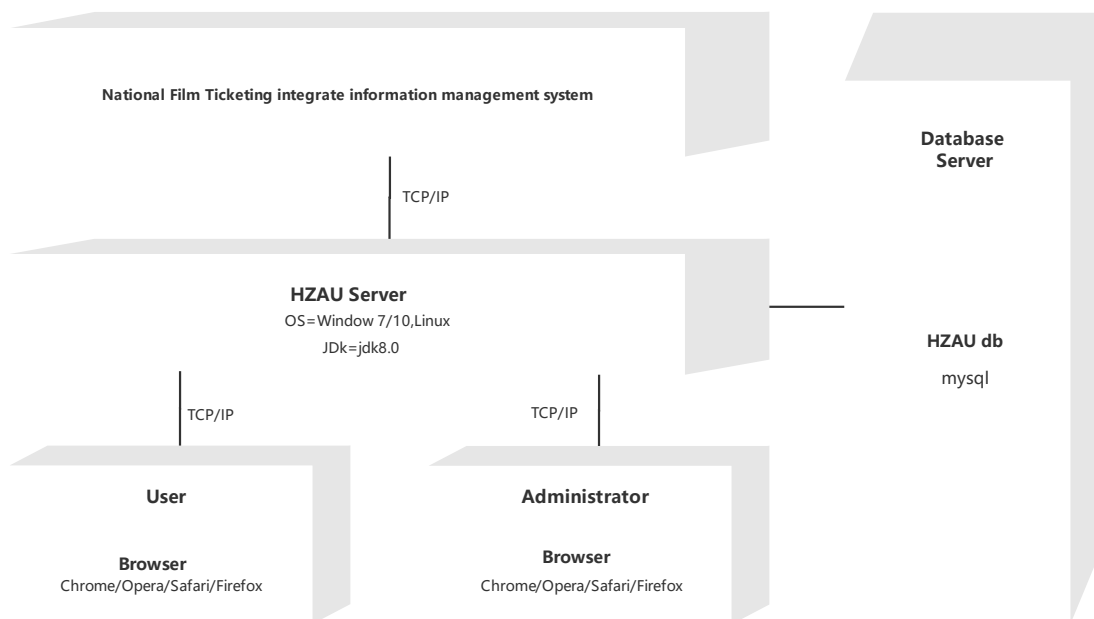
- **场次编排：**管理人员可以对电影进行排档，也可以将已下架的电影从场次表里删除。
- **影片录入：**管理员可以编辑每部电影的具体信息，包括上映时间、票价等。
- **订单管理：**用户可以浏览影院在映影片，并选择合适场次，进行图形化订票，且每次能最多可订四张票，已订电影票可以进行线上退票，系统保留购买记录并标识已退票，在十分钟内将操作上报票务数据平台，。
- **管理账户：**新用户可利用本系统进行注册，已有账号可凭密码登入。管理员凭‘员工卡’并输入员工号和密码才能进入管理系统进行管理。
- **会员管理：**对影院会员的办理、退出进行管理，购买会员显示交易记录。
- **结算管理：**对影院的日、月销售额进行统计，管理员可登录系统查看报表。
- **票房统计：**系统自动生成每部电影的当日票房数据，并在每天 12 点前将前一天票房数据上报国家电影总局。
- **售票情况统计：**对影院的影片上座率、实时售票情况进行统计。
- **电影推荐：**根据预约和票房数据，每天选择一部影片显示在用户界面上方。

## 2.2 软件结构

软件系统结构，



物理结构



## 3 模块描述

### 3.1 模块基本信息

**模块名：**登录/注册

**设计者：**李铜平

**模块描述：**处理用户登录请求，获取用户输入并与后台数据库中对应用户的密码进行比对，再判断是否相应请求。若登录时输入账号不存在，弹出注册请求，用户填写完数据之后录入数据库进行存档。

**模块名：**个人信息管理

**设计者：**李铜平

**模块描述：**用户可以在 用户端—>个人中心 点击头像进行个人信息的修改，为方便起见，本软件允许用户的昵称重复，而选用用户注册时系统生成的唯一用户 id 进行区分。

**模块名：**会员办理

**设计者：**李铜平

**模块描述：**用户可以在 用户端—>个人中心—>财富一览—>我的 VIP 进行会员办理，支持会员的退订(每月 10 号结算，之后的退订请求顺延到下月)。

**模块名：**浏览影片

**设计者：**王南松

**模块描述：**用户可以在 用户端—>首页 浏览影城上映的影片，并可通过上方搜索栏进行影片的查找。



**模块名：订退票**

**设计者：**徐浩春

**模块描述：**用户可点击影片进行订票操作，支持可视化在线选座功能，并将可供用户选择的座位标注出来，用户选定之后，提示用户进入支付界面，支付完成后向后台发送订票请求，后台数据库生成订单，并发送到用户账户。

**模块名：订单管理**

**设计者：**徐浩春

**模块描述：**用户可在 用户端—>个人中心—>财富一览—>订单查询 浏览历史订单，并可在此界面对生效前的订单进行退订操作。

**模块名：电影评价**

**设计者：**常家乐

**模块描述：**用户可对已看电影发表评论。

## **管理员系统**

**模块名：会员信息管理**

**设计者：**李铜平

**模块描述：**管理员可对影城的会员账户进行管理，包括会员账号的查询，优惠内容的变更等。

**模块名：影片信息录入**

**设计者：**王南松

**模块描述：**影片上映前，由管理人员进行影片信息的录入，包括影片名称、票价、影片编码(从国家票务数据平台获取)等，录入完毕之后数据库进行存档，进入排片操作。

**模块名：排片**

**设计者：**王南松

**模块描述：**管理人员可根据影城的影厅数量及利用情况对影片进行时间、场地上的编排，确保同任意两部电影的放映时间、场地不重叠。排片结束后，将信息反馈到用户端，用户即可在首页浏览影片。

**模块名：**订单查询

**设计者：**常家乐

**模块描述：**主要用于后续的销售情况核对，便于查找未生效订单等。

**模块名：**票房数据管理&结算管理

**设计者：**陶威

**模块描述：**每部电影下架后，后台数据库自动生成报表，并上报国家票务数据平台。

同时，管理人员可据此对该片的票房及上映期间的每日销售情况等进行可视化分析。主要统计每部影片的日销售情况及上座率。

**模块名：**影片推荐

**设计者：**常家乐

**模块描述：**管理人员可根据每部影片的结算管理数据，定期对用户端首页上方的电影推荐栏进行变更，选取影院当前热映影片进行显示。

## 3.2 模块设计及模块接口

模块名称	管理员会员信息管理 ( AdminVipService )
功能	看图
AdminVipService.overdueVip	public List<User> overdueVip(Date time) 查询再time这天vip将要过期的用户
AdminVipService.extendVip	public void extendVip(User user, Date time) 确认user的vip延长天数
AdminVipService.deleteBenefit	public void deleteBenefit(Integer benefitId) 删除benefitId的benefit
AdminVipService.insertBenefit	public void insertBenefit(String benefit) 新增benefit
AdminVipService.updateBenefit	public void updateBenefit(String newBenefit,Integer benefitId) 更新benefitId的内容
AdminVipService.selectBenefit	public String selectBenefit(User user) 查询user的benefit

需要的服务
-------

UserDao.selectVip	public List<User> selectVip(Date time) 查找在Date将要过期的vip
UserDao.updateVip	public void updateVip(String userName, Date time) 延长vip
UserBenefitDao.selectUserBenefit	public List<String> selectUserBenefit(String userName) 查找user的所有有效benefit
BenefitDao.selectBenefit	public List<Benefit> selectBenefit() 查找所有有效的优惠
BenefitDao.insertBenefit	public Benefit insertBenefit(String benefit) 新增benefit
BenefitDao.updateBenefit	public Benefit updateBenefit(Integer benefitId,String newBenefit) 修改benefit
Benefit UserDao.updateUserBenefit	public void updateUserBenefit(Integer benefitId,String newBenefit) 修改user的benefit
BenefitDao.deleteBenefit	public void deleteBenefit(Integer benefitId) 删除benefit
Benefit UserDao.deleteAllUserBenefit	public void deleteAllUserBenefit(Integer benefitId) 删除所有user的benefit
Benefit UserDao.deleteUserBenefit	public void deleteUserBenefit(String userName, Integer benefitId) 删除user的benefit

模板名称	会员办理 ( UserVipService )
模板功能	办理user的vip服务
UserVipService.checkVip	public Date checkVip(User user)
	查看vip剩余天数
UserVipService.extendVip	public void extendVip(User user,Double money)
	根据交的钱延长VIP时长

需要的服务
-------

UserDao.selectVipStart	public Date selectVipStart(String userName) 查找user vip开始时间
UserDao.selectVipEnd	public void selectVipEnd(String userName) 查找vip结束时间
UserDao.updateVip	public void updateVipd(Date time) 延长vip时间

模块名称	用户模块 ( UserService )
功能描述	用户在客户端登陆注册账户
UserService.userLogin	public String userLogin(User user)
	根据userName去找账户是否存在，然后比对password是否正确，账户不存在返回账户无效，密码不同返回密码错误成功则返回登陆成功
UserService.register	public void register(User user)
	根据userName去查找是否存在，没有则新加一条账户信息

需要的服务 ( 需接口 )
---------------

UserDao.inputUser(User user)	public void inputUser(User user) 根据user中的信息创建账户
UserDao.selectUserByName(String userName)	public User selectUserByName(String userName) 根据userName查找账户

模块名称	用户订单管理 ( UserOrderService )
模块功能	订单管理
UserOrderService.selectCompleteOrder	public List<Order> selectCompleteOrder(User user,Date startTime,Date endTime) 查询user一段时间内已完成的订单
UserOrderService.selectIncompleteOrder	public List<Order> selectIncompleteOrder(User user) 查询user未完成订单
UserOrderService.selectOutcome	public Double selectOutcome(User user,Date startTime,Date endTime) 查询用户在一段时间的支出
UserOrderService.payOrder	public void payOrder(User user, Double money, Order order) 支付订单

需要的服务
-------

UserOrderDao.selectCompleteOrder	public List<Order> selectCompleteOrder(User user,Date startTime,Date endTime) 查询user一段时间内已完成的订单
UserOrderDao.selectIncompleteOrder	public List<Order> selectIncompleteOrder(User user) 查询user未完成订单
UserOrderDao.selectOutcome	public Double selectOutcome(User user,Date startTime,Date endTime) 查询用户在一段时间的支出
UserOrderDao.insertOrder	public void insertOrder(User user, Double money, Order order) 支付订单

模块名称	用户信息 ( UserInformationService )
功能概述	对用户的基础信息进行修改
UserInformationService.changePassword	public String changePassword(User user,String newPassword)
	根据userName查找到User, 否则返回查无此账户, 当新密码符合规定时, 修改密码, 否则返回失败
UserInformationService.changePhone	public String public changePhone(User user,String phone)
	根据userName查找到User, 否则返回查无此账户, 当新电话符合规定时, 修改电话, 否则返回失败
UserInformationService.changeEmail	public String changeEmail(User user,String email)
	根据userName查找到User, 否则返回查无此账户, 当新邮箱符合规定时, 修改邮箱, 否则返回失败

需要的服务

UserDao.selectUserByName(String userName)	public User getAccountByName(String userName) 根据userName查找账户
UserDao.updatePassword	public void updatePassword(String userName,String newPassword) 根据找到的user更改密码
UserDao.updatePhone	public void updatePhone(String userName,String newPhone) 根据找到的user更改电话
UserDao.updateEmail	public void updateEmail(String userName,String newEmail) 根据找到的user更改邮箱



模块名称	影评 ( FilmReviewService )
模块功能	用户对电影发表评价
FilmReviewService.publishComments	public void publish Comments(User user,Movie movie, String comment)
	user对movie发表comment
FilmReviewService.changeComments	public void changeComments ( User user , Movie movie,String newComment )
	user更改对movie的comment

需要的功能

FilmReviewDao.insertComment	public void insrtComment(String userName,String movieName,String comment) 新增影评
FilmReviewDao.selectComment	public String selectComment(String userName,String movieName) 查找影评
FilmReviewDao.updateComment	public void updateComment(String userName,String movieName,String newComment) 更改影评
FilmReviewDao.deleteComment	public void deleteComment(String userName,String movieName) 删除影评

模块名称	影片推荐 ( FilmRecommendationService )
模块功能	见图
FilmRecommendationService.selectMovies	public List < Movie > selectMovies(Integer amount) 推荐电影

需要的服务

MovieDao.selectBoxOffice	public List < Movie > selectMovie(Integer amount) 根据票房选择前amount的电影
FilmRecommendationDao.updateMovie	public void updateMovie(List < Movie > movies) 更新推荐电影

模块名称	浏览影片 ( BrowseMoviesService )
模块功能	用户可以在用户端- -> 首页 浏览影城上映的影片，并可通过上方搜索栏进行影片的查找。
BrowseMoviesService.selectMovieByTime	public List<Movie> selectMovieByTime(String startTime, Sting endTime)
	根据时间区域选电影
BrowseMoviesService.selectMovieByName	public List<Movie> selectMovieByName(Sting name)
	根据名字选电影
BrowseMoviesService.selectMovieByPrice	public List<Movie> selectMovieByPrice(Double highestPrice, Double lowestPrice)
	根据价格区域选电影

需要的服务

BrowseMoviesDao.selectMovieByName	public List<Movie> selectMovieByName(Sting name) 根据名字选电影
BrowseMoviesDao.selectMovieByPrice	public List<Movie> selectMovieByPrice(Double highestPrice, Double lowestPrice) 根据价格选电影
BrowseMoviesDao.selectMovieByTime	public List<Movie> selectMovieByTime(String startTime, Sting endTime) 根据时间选电影

模块名称	管理员订单 ( AdminOrderService )
功能	见图 ( 订单查询、票房数据管理、结算管理 )
AdminOrderService.selectIncome	public Double selectIncome(Date startTime, Date endTime) 查找一定时间内的电影院的收入
AdminOrderService.selectMovieIncome	public Double selectMovieIncome(String movieName) 查找movie的收入情况
AdminOrderService.selectOccupancyRate	public Double selectOccupancyRate(String movieName) 查找movie的上座率

需要的服务

AdminOrderDao.selectOrder	public Double selectOrder(Date startTime, Date endTime) 查找一段时间的销售金额
AdminOrderDao.selectMovieIncome	public Double selectMovieIncome(String movieName) 查找movie收入
AdminOrderDao.selectMovieOrder	public Integer selectMovieOrder(String movieName) 查找movie销售票数
AdminOrderDao.selectMovieHall	public Integer selectMovieHall(String movieName) 查找movie放映厅号

模块名称	订票 (TicketService)
功能描述	用户浏览电影时进行订票
提供服务(接口描述)	<p>TicketService</p> <p>Public List&lt;Movie&gt; selectMovie(String filmname); 查找电影</p> <p>TicketService.</p> <p>Public void selectSeat(Integer columnIndex, Integer rowIndex); 显示该影厅的座位情况</p> <p>TicketService.</p> <p>Public void getSeat(List&lt;Seat&gt; seats); 线上选座并锁定座位</p> <p>TicketService</p> <p>Public Order insertTicket(Ticket ticket); 生成订单</p> <p>TicketService.</p> <p>Public void purchase(Integer orderId, Double money); 付款</p> <p>TicketService</p> <p>Public void setStatus(Integer orderId, Integer status); 变更订单状态</p> <p>TicketService</p> <p>Public void updateSeat(); 更新座位数据</p>

需要服务(方法 调用)	<p>TicketDao</p> <p>public Date selectTime(Timestamp time); 获取订单创建时间</p> <p>TicketDao</p> <p>public void setSeats(List&lt;Seat&gt; seats); 锁定座位</p> <p>TicketDao</p> <p>public Integer selectOrderId(Integer orderId); 获取订单号</p> <p>UserServiceDao</p> <p>public String selectUserId(User user); 获取用户账号</p> <p>TicketDao</p> <p>public Integer selectFilmId(Movie movie); 获取电影 id</p> <p>TicketDao</p> <p>public void updateState(Integer state); 设置订单状态</p> <p>TicketDao</p> <p>public Double selectActualPay(Integer orderId); 获取交易金额</p>
依赖关系	<p>接口放在 src.Service 中，其具体实现放在 src.Impl 中，</p> <p>TicketServiceImpl 实现&lt;interface&gt;TicketService，</p> <p>TicketServiceImpl 依赖于 src.vo.Ticket;</p> <p>src.Ticket 依赖于 src.vo.Seat。</p>

模块名称	退票 (ReturnService)
功能描述	用户可在订单中心进行退票
提供服务(接口描述)	ReturnTicketService public void send(); 发送退票请求 ReturnTicketService public void refund(Integer orderId); 退款 ReturnTicketService public void setStatus(Integer orderId, Integer status); 变更订单状态 ReturnTicketService public void updateSeat(); 更新座位数据
需要服务(方法调用)	TicketDao Public Integer selectOrderId(String username,String movieName); 获取订单号 TicketDao Public Double selectActualPay(Integer orderId); 获取交易金额 TicketDao Public void deleteTicket (Integer orderId); 删除订单 TicketDao ReturnServiceImpl.deleteFromList(Integer orderId) 反馈至订单中心

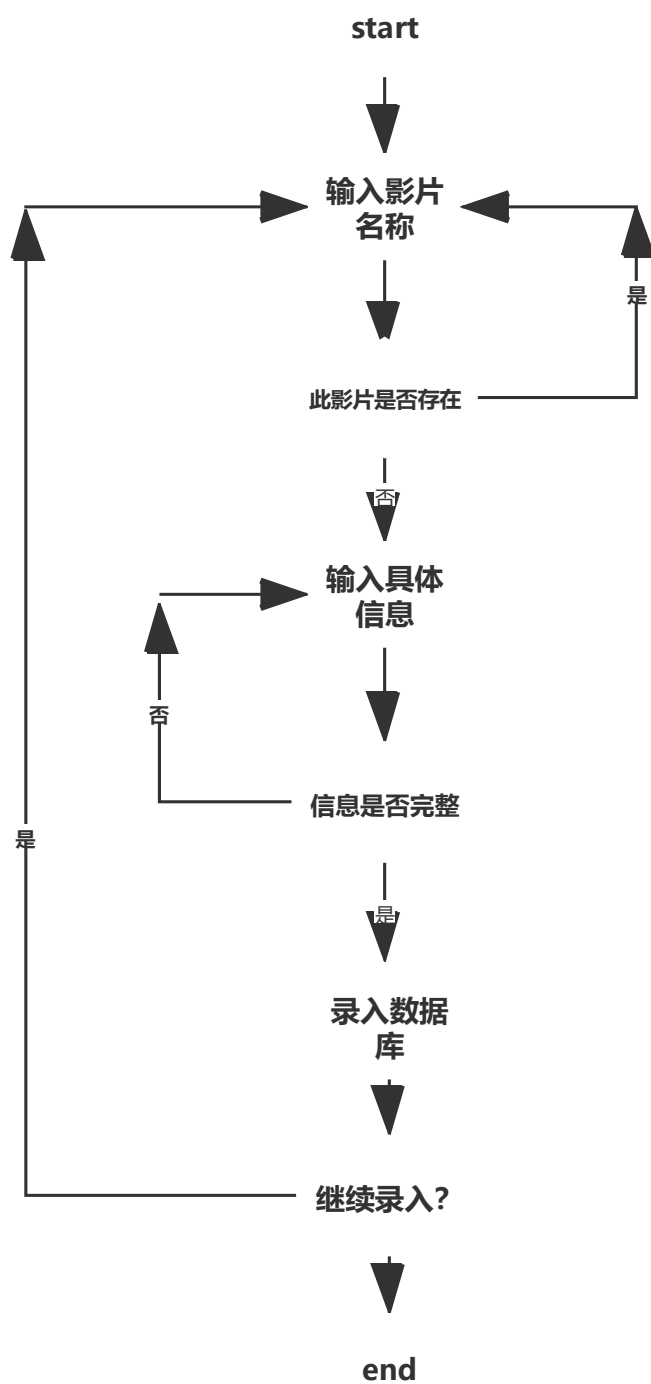
模块名称	排片 (SortService)
功能描述	<p>管理人员可根据影城的影厅数量及利用情况对影片进行时间、场地上的编排,确保同任意两部电影的放映时间、场地不重叠。排片结束后, 将信息反馈到用户端, 用户即可在首页浏览</p>
方法成员	<p>public void forward(Integer movieId, Date time)时间前移</p> <p>public void back(Integer movieId, Date time)时间后移</p> <p>public void studio(Integer movieId, Integer movieHallId)安排影厅</p> <p>public List&lt;Movie&gt; display()排片显示</p> <p>需要的服务</p> <p>MovieDao</p> <p>public void updateMovieTime(Integer movieId, Date time)</p> <p>影片时间更改</p> <p>public List&lt;Movie&gt; selectMovie(String movieName)查找影片信息</p> <p>public void updateMovieHall(Integer movieId, Integer movieHallId)安排影厅</p>



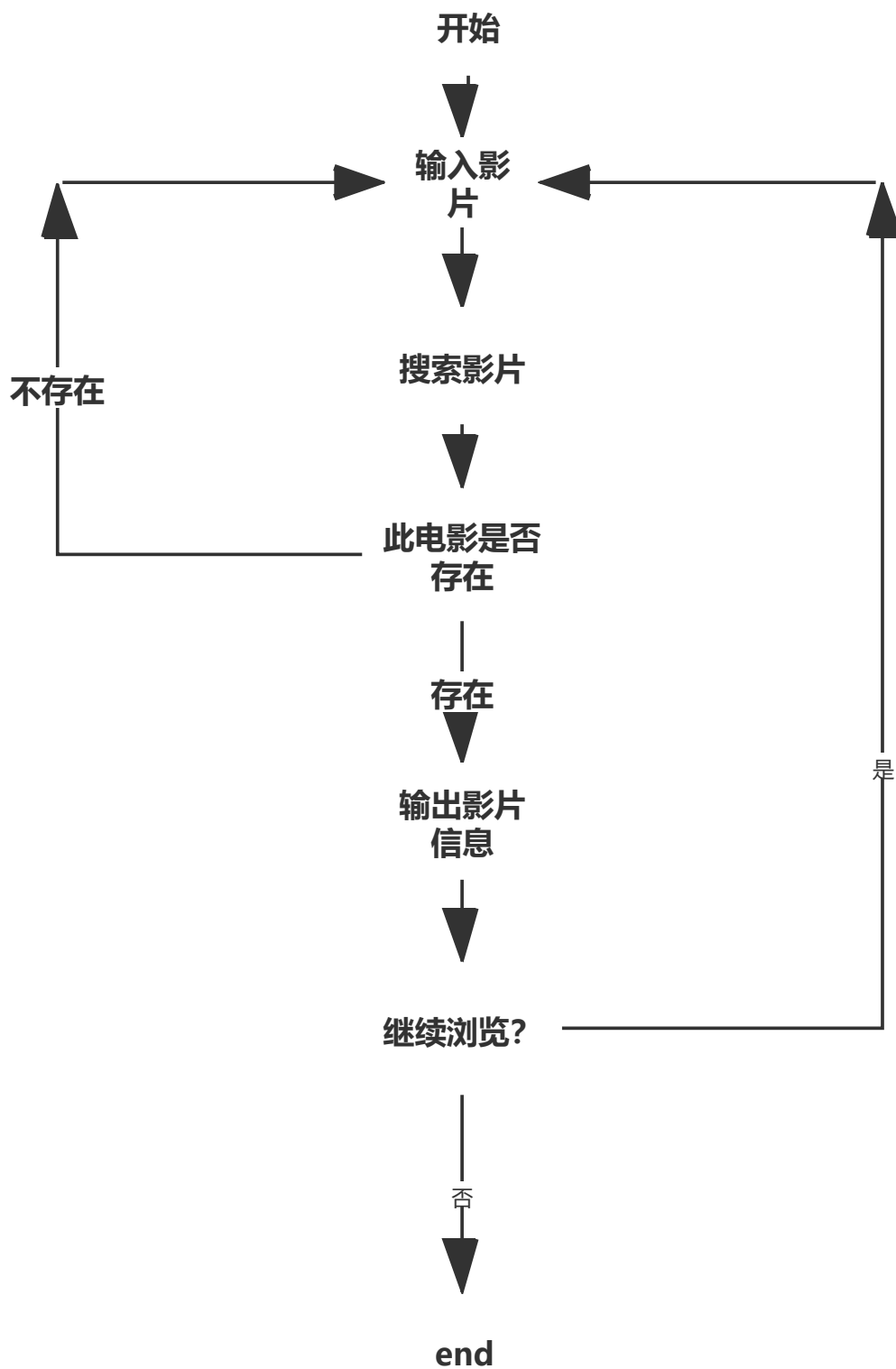
模块名称	影片录入 (MovieInputService)
功能描述	影片上映前,由管理人员进行影片信息的录入,包括影片名称、票价、影片编码(从国家票务数据平台获取)等, 录入完毕之后数据库进行存档, 进入排片操作。
方法成员	<p>public void movieInput(Movie movie)录入电影</p> <p>public void store()数据存档</p> <p>需要的服务</p> <p>MovieDao.insertMovie(Movie movie)</p> <p>录入电影</p>

### 3.3 模块处理逻辑

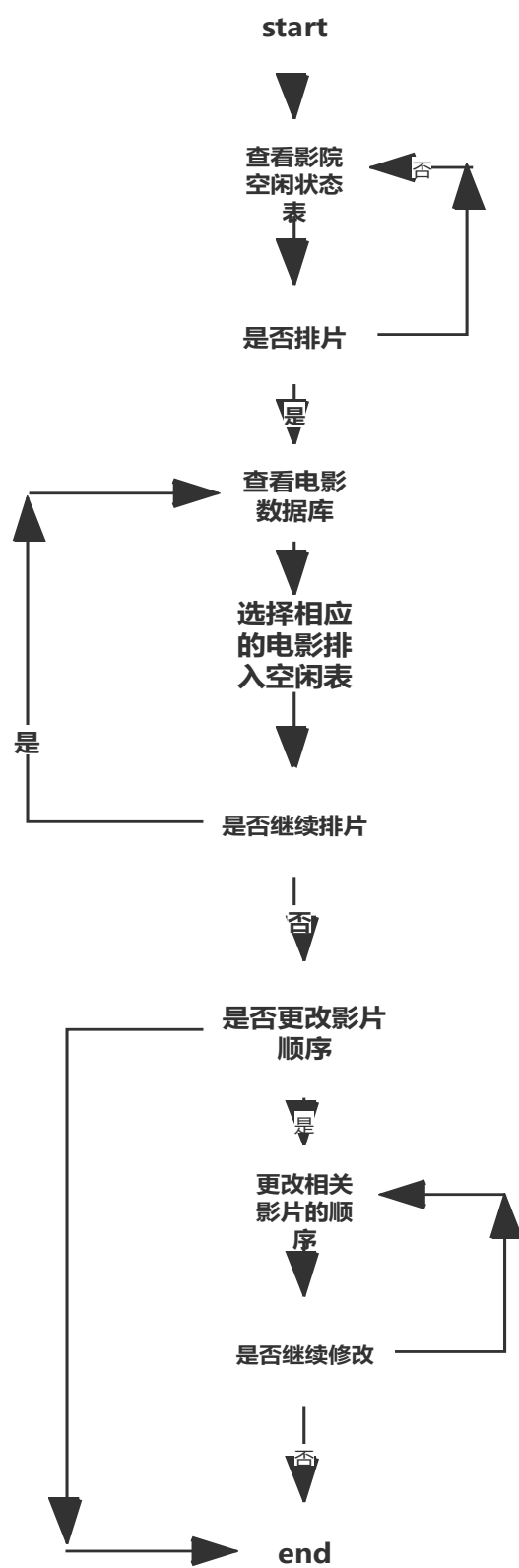
浏览影片：



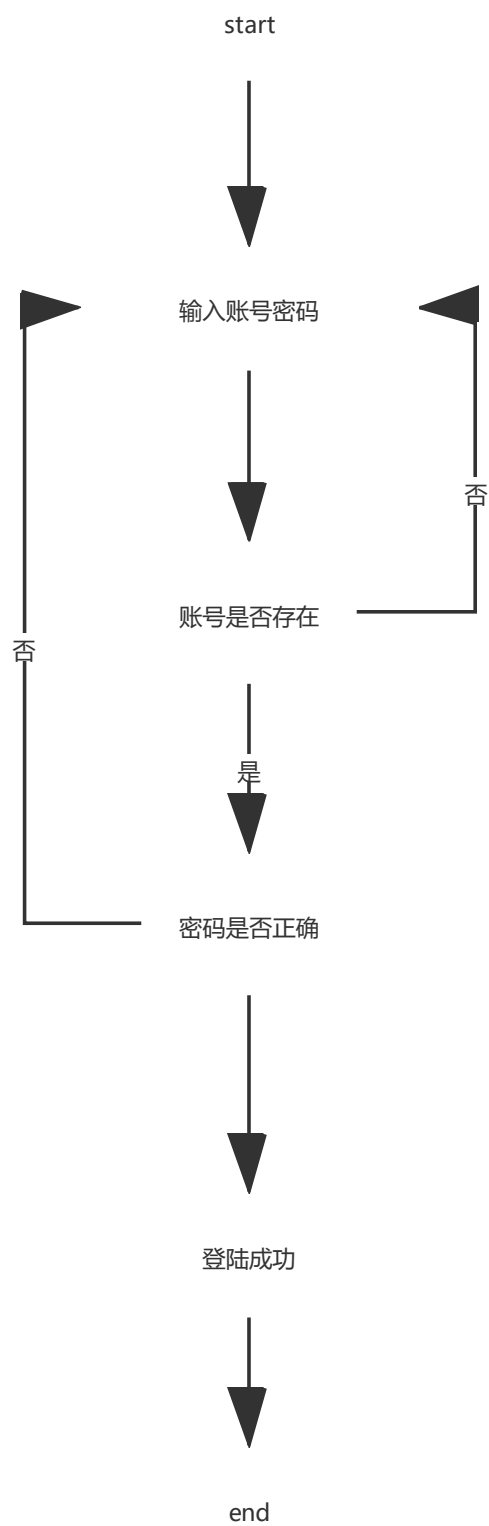
浏览影片：



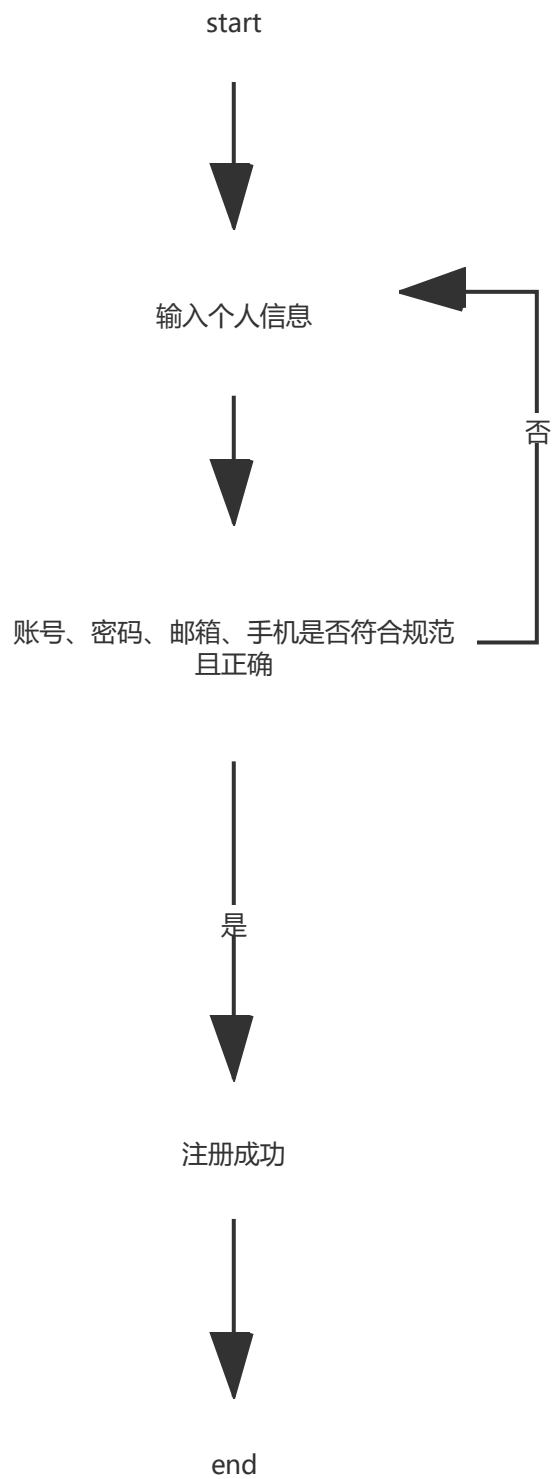
排片:



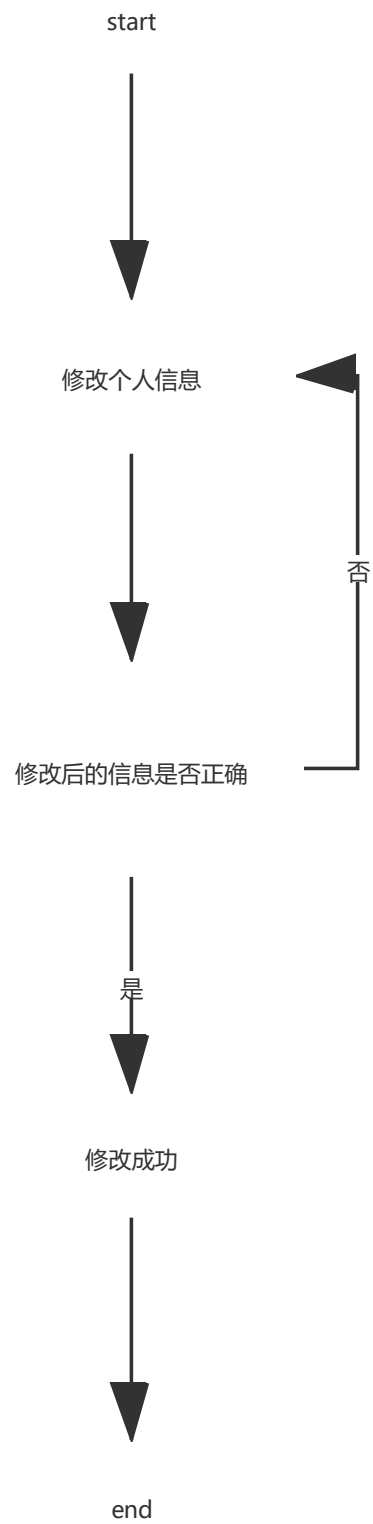
登录:



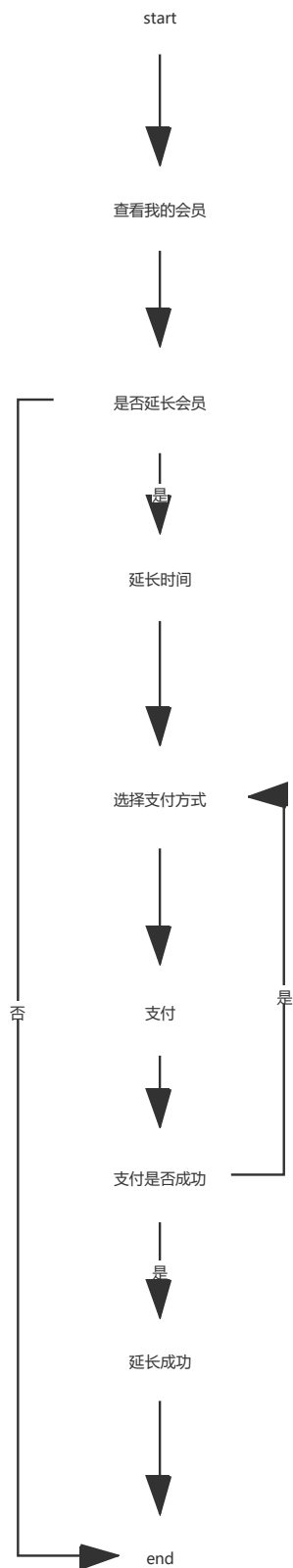
注册:



个人信息管理：

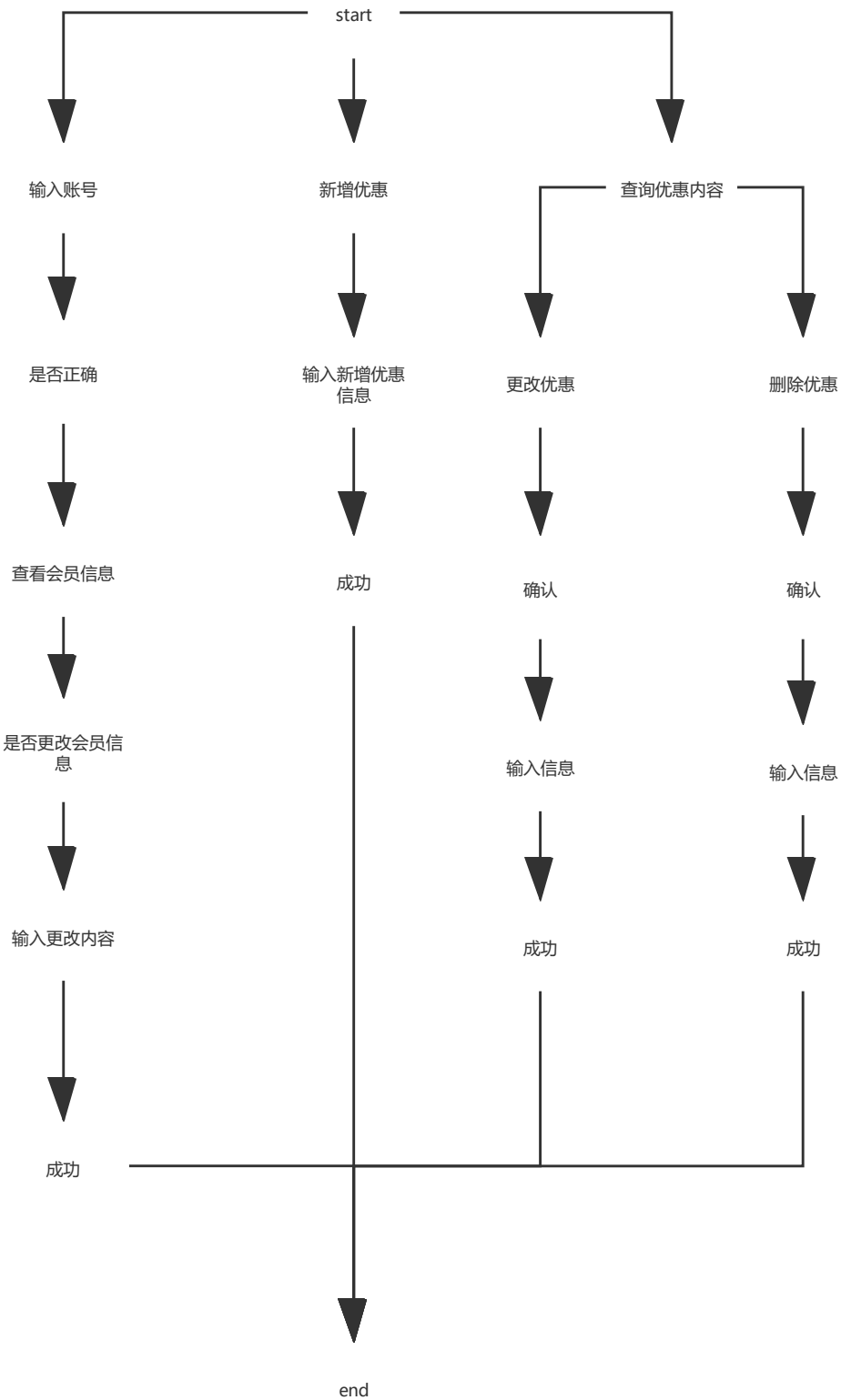


## 会员办理

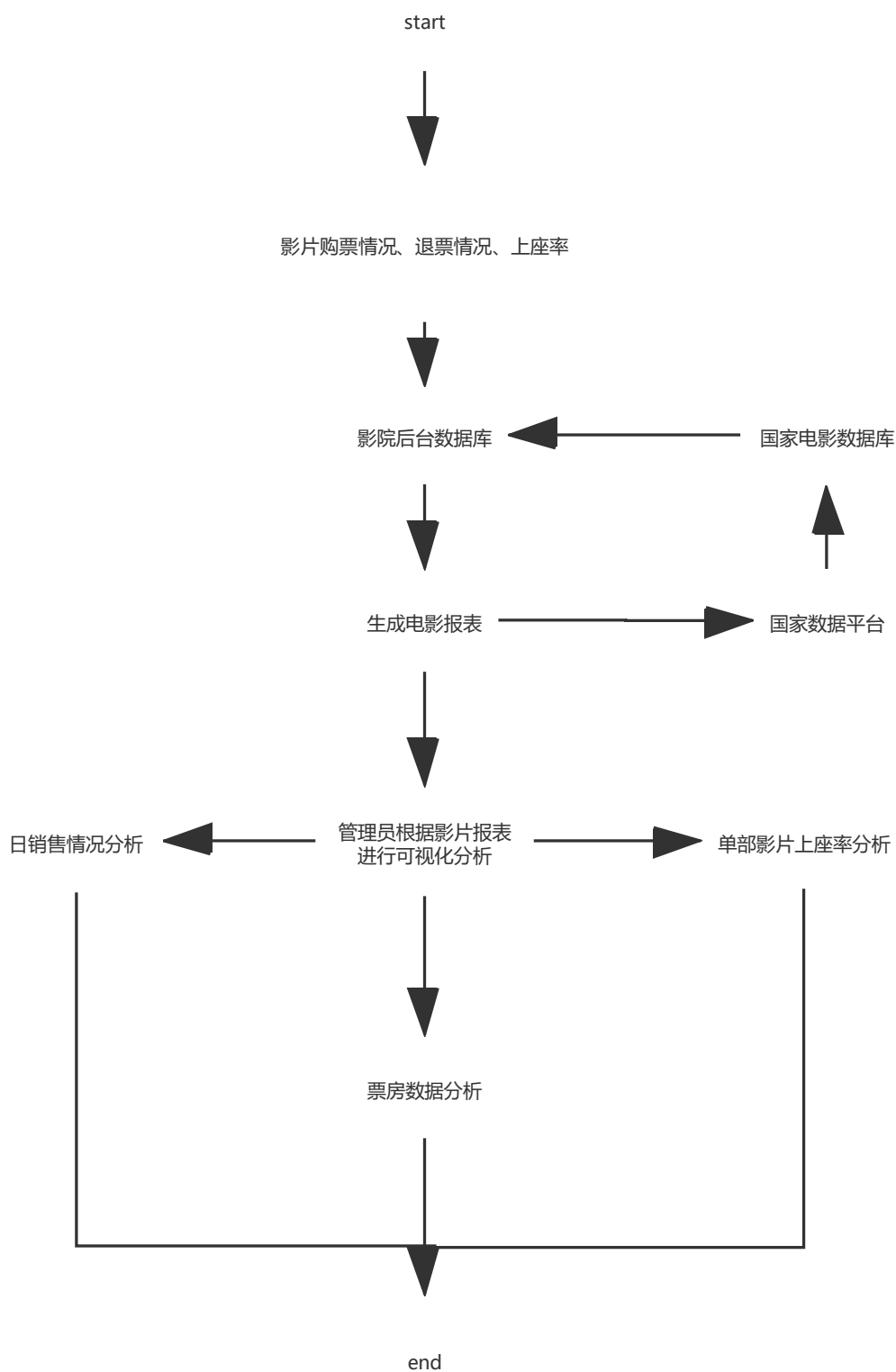




会员信息管理：

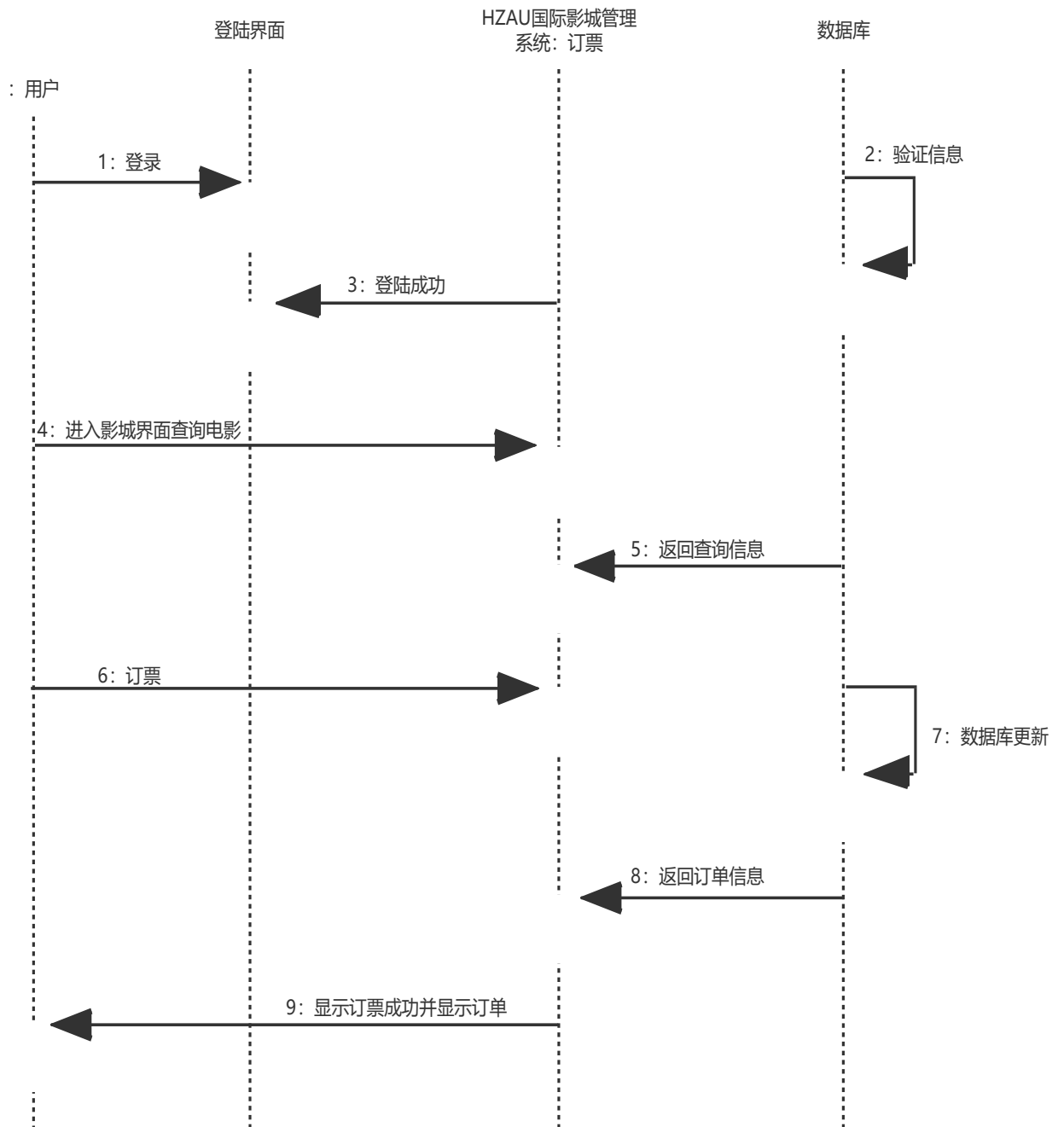


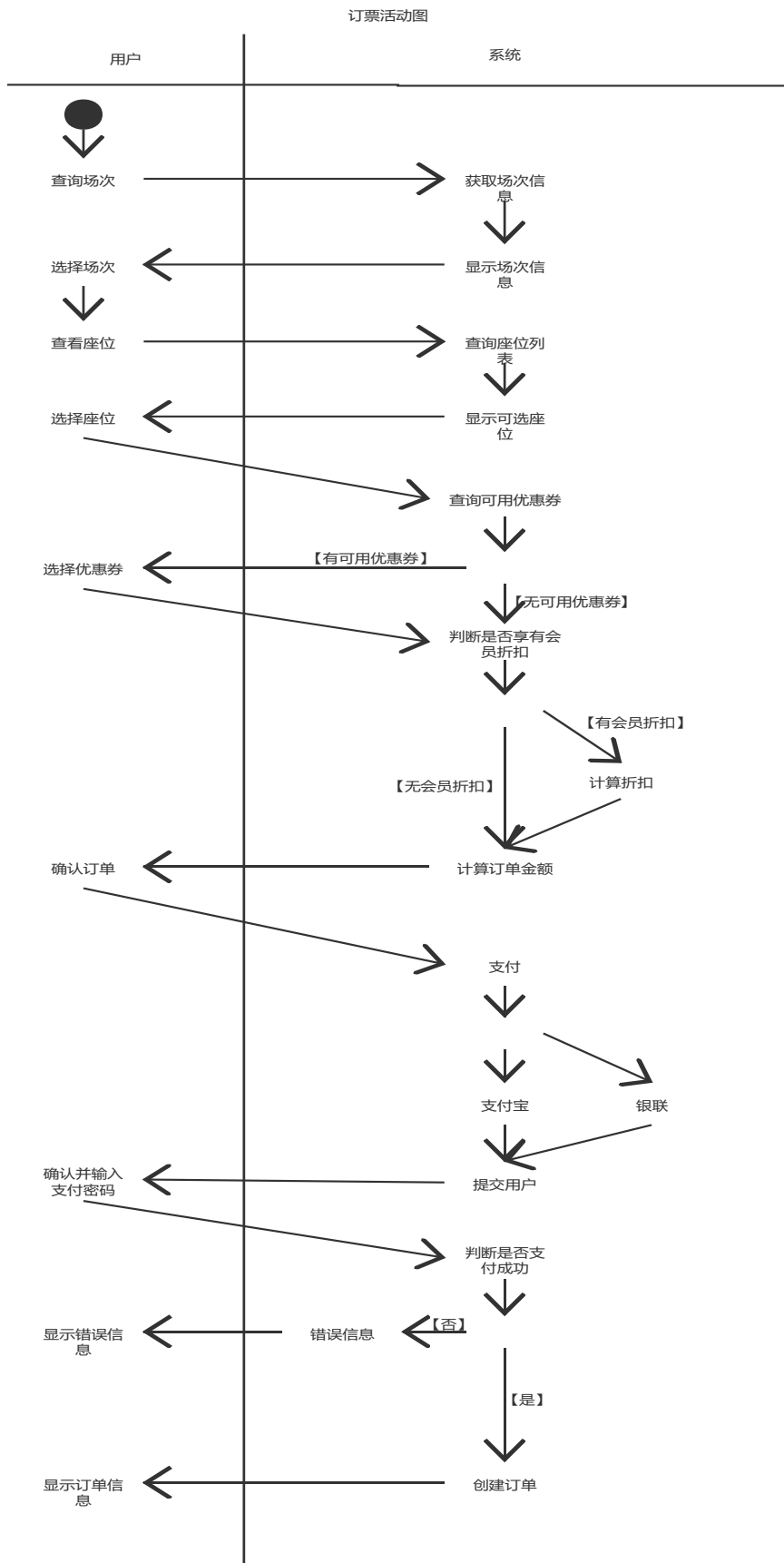
票房数据管理&结算管理:



## 订票:

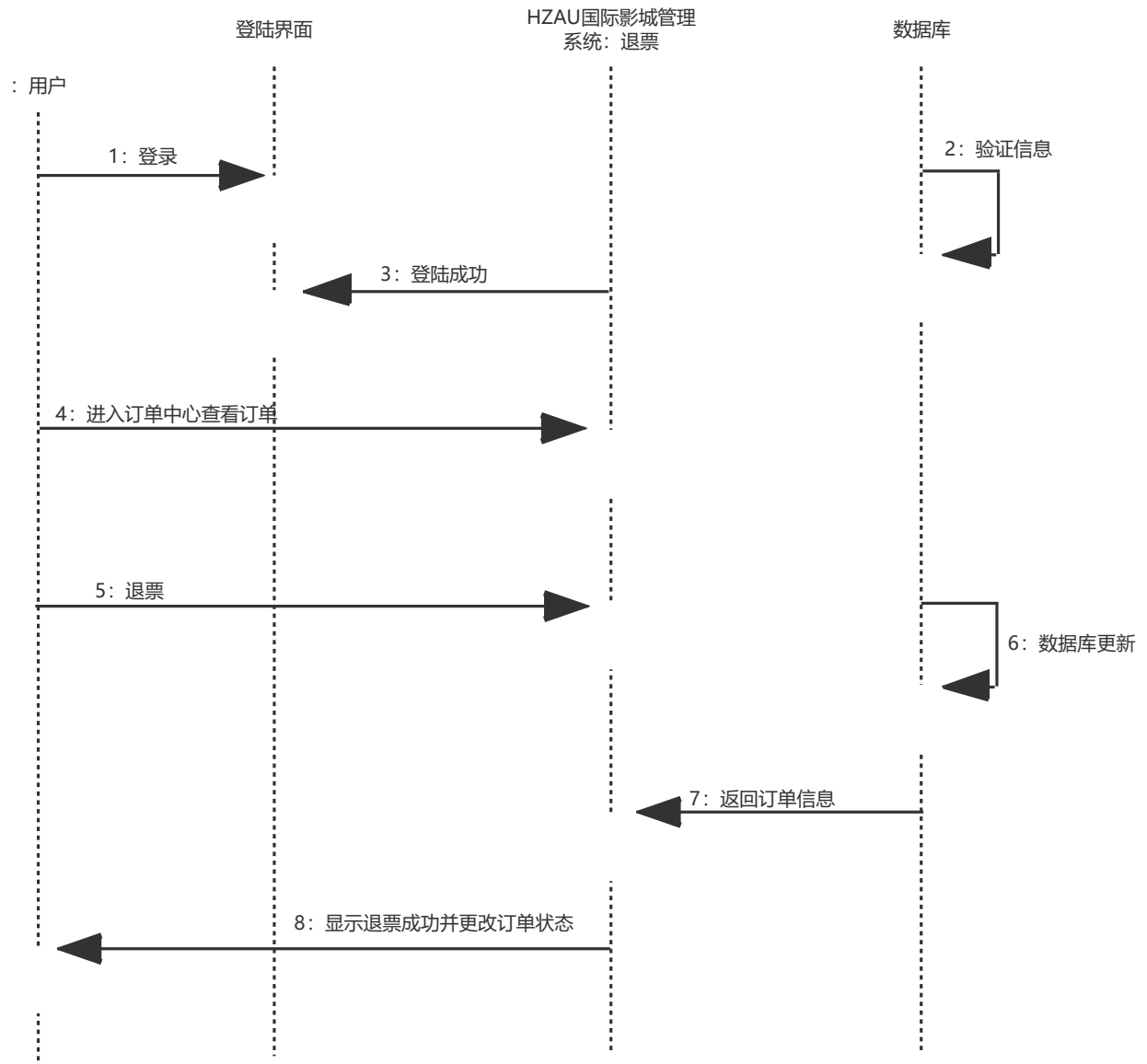
订票流程图

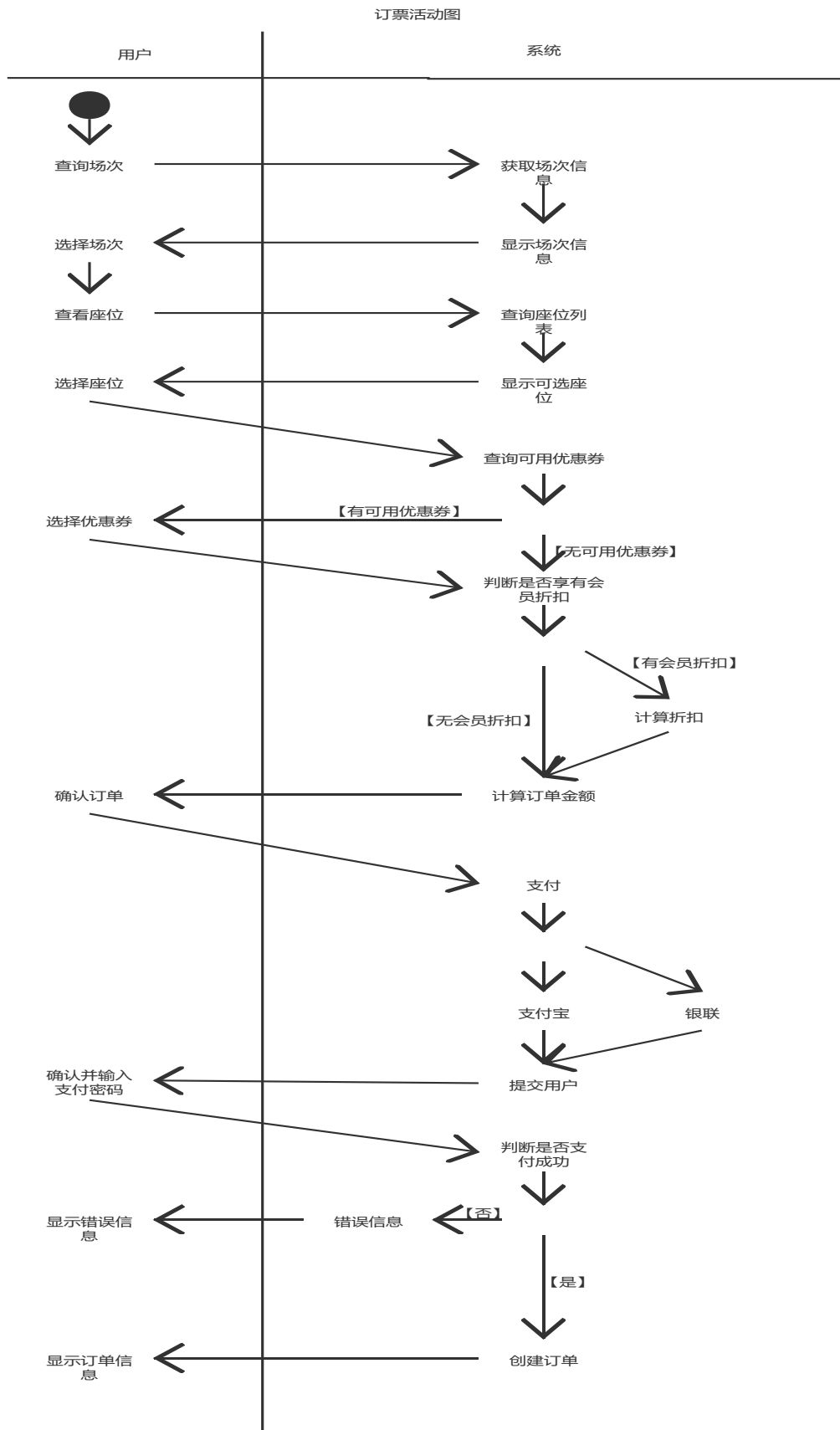




## 退票:

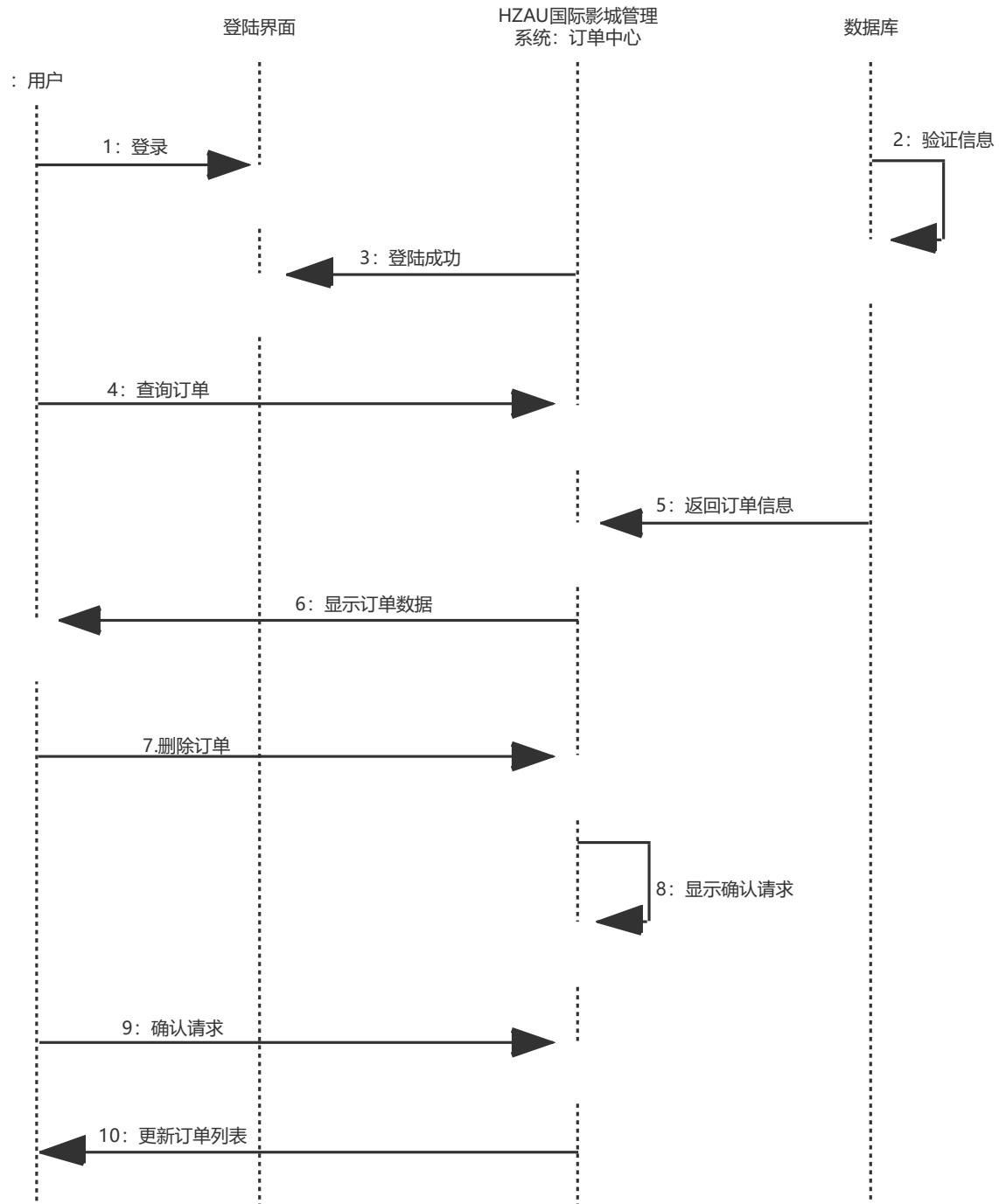
退票流程图



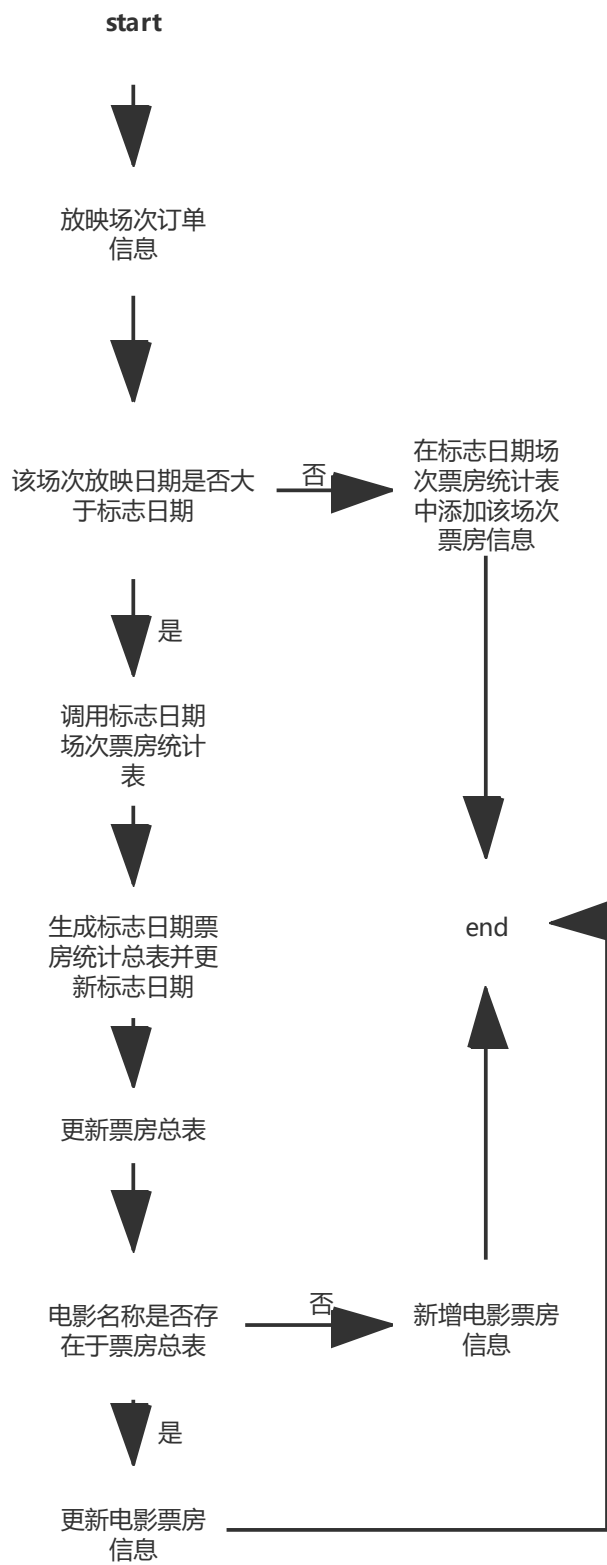


## 订单管理：

订单管理流程图

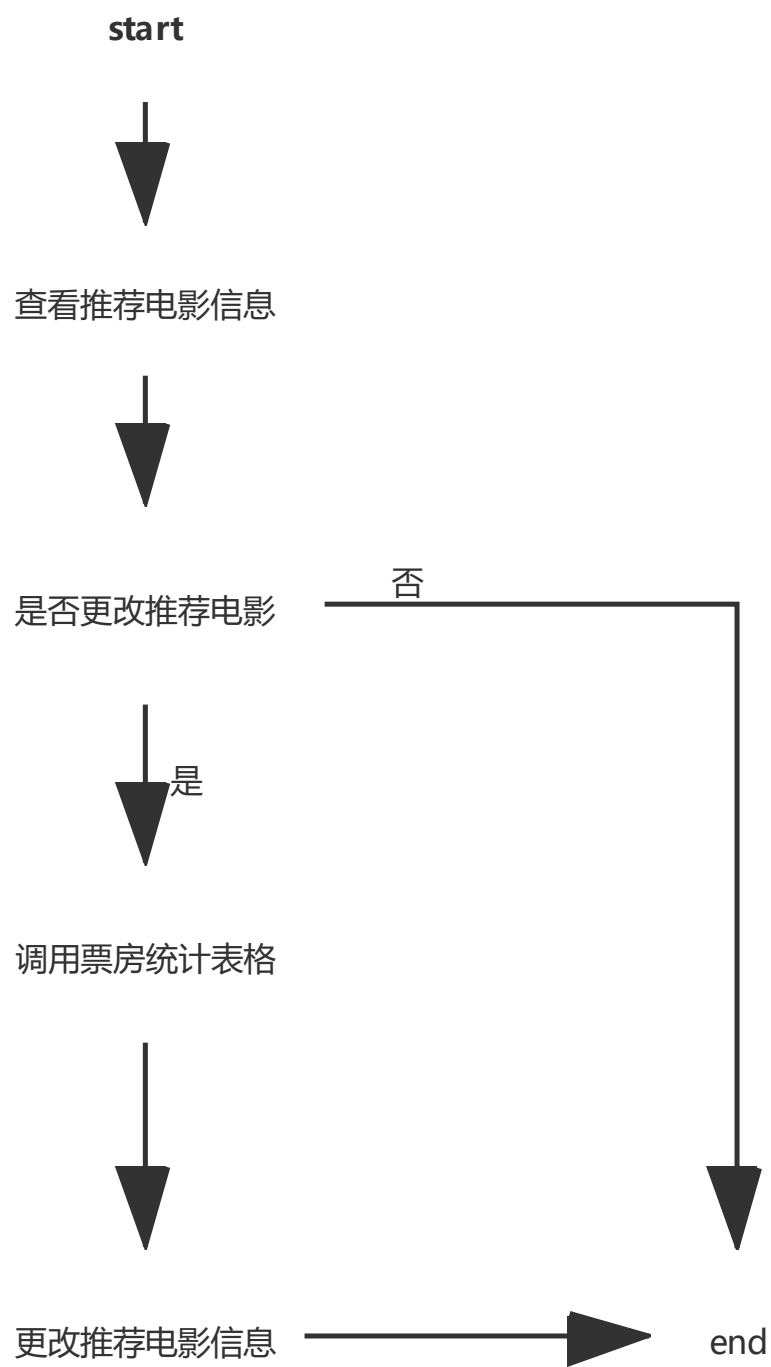


票房统计:





电影推荐：



## 3.4 算法

### 浏览影片：

- 1.用相应的控件列出所有可以观看的影片信息
- 2.利用 sql 语句筛选出符合条件的影片，显示出来，并将搜索记录存入数据库

### 影片录入：

将影片的相关信息输入进数据库，缺少某项信息则标记 no，信息完全则标记 yes，录入结果存入数据库，反馈给管理员，进行后续操作

### 排片：

将空闲时间和空闲的影厅数据列表，将相关影片填入，利用 sql 语句对排片列表进行增删改查，将下映影片标记 0

### 订单管理：

- 1) 用户点击订单中心后，后台读取操作信息，从数据库调取该用户的订单，以列表形式显示在用户界面；
- 2) 用户可选择目标订单，未失效订单或在退票允许时间范围内的订单显示退票功能，失效订单则显示删除功能；
- 3) 退票操作，见下文；
- 4) 删除，选定目标订单并点击删除后，后台会发送确认指令，用户确认后，后台读取并记录本次删除订单的订单号，下一次刷新订单列表时则对应订单不予显示。

### 订票：

- 1) 用户选订电影之后，后台通过电影 id 从数据库中调取该电影的场次信息，并将对应影厅的目前座位锁定情况  
以图形界面的方式反馈至用户端；
- 2) 用户选定座位后，座位信息会暂存在临时数组中，用户支付成功后，后台利用 SQL 语句向数据库对应表中写入  
座位信息，并更新该影厅的座位锁定情况；
- 3) 用户支付成功后，数据库利用后台传输的信息（用户 id，影片 id 等）自动在订单表中创建新的订单；
- 4) 后台读取数据库中新订单的信息，反馈至用户端。

### 退票：

- 1) 用户点击退票后，后台会读取当前时间，并与订单中电影放映时间比对，若时间超出一定范围则不予退票；
- 2) 退票条件成立的情况下，后台会读取订单号，暂存于数组中，并向用户提交确认请求，用户需输入账户密码确认身份，之后后台利用保存的订单号，检索数据库；
- 3) 查询到对应订单后，更改订单状态，读取订单金额，进行退款操作，完成后将退款结果反馈至客户端。

**票房数据管理&结算管理:**

- 1.利用 sql 语句查询电影的购票情况、退票情况、上座率
- 2.后台数据库利用 sql 语句将购票情况、退票情况、上座率制成新的表格, 管理员依次进行各方面的数据分析
- 3.利用 sql 语句, 向国家数据平台提供数据, 主要是票房数据和影片报表
- 4.利用 sql 语句, 从国家数据平台获取电影数据库详细情况

**票房统计:**

票房数据是根据每个场次的数据进行统计, 调用数据库查看订单信息, 得到订单场次与票价, 每场次总票房等于该场次订单票价之和, 进而可以得出当日总票房(当日所有订单票价之和), 每电影总票房(当日每个电影的订单票价之和), 每场次总票房(单场次订单票价之和), 场次最高票房(某场次最高票房), 日最高电影票房(某电影订单票价之和)等等, 还可以根据订单信息得出场次上座率(某场次入座总人数/该放映室座位数)。

**电影推荐:**

在票房统计时, 将电影票房、场次票房、场次入座率进行降序排列, 方便管理人员查看电影热度情况进行电影推荐。

## 3.6 性能

**排片模块:**

精度:场次放映时间精确至分钟, 放映室编号精确至整数。

**影片录入模块:**

精度:影片时长精确至秒, 上映时间精确至日, 上映地区精确至国家。

**订单管理模块:**

精度:交易时间与影片时长精确至秒, 电影放映时间精确至分钟, 交易金额精确至分。

**用户模块:**

精度:电影放映时间精确至分钟, 账户余额精确至分。

**退订票模块:**

精度:下单时间与退订票时间精确至秒, 交易金额精确至分。

**电影评价模块:**

精度:评价时间精确至日。

**会员信息管理模块:**

精度:会员剩余时长精确至分钟, 会员价格精确至分。

**结算管理模块:**

精度:金额数据精确至分, 时间数据精确至日。

**票房数据管理模块:**

精度:票房金额数据精确至分, 时间数据精确至日。

**影片推荐模块:**

精度:影片推荐更新时间精确至日。

所有模块时间特性要求:响应时间、处理时间等都为毫秒。

所有模块灵活性需求:OS=Windows7/10,Linux;Browser:Chrome/Safari/Firefox/Opera。

### 3.7 测试计划

测试用例编号	输入数据	覆盖情况	预期输出结果
1 排片模块	2月1日9厅15:00 《诛仙》	有效类	2月1日9厅15:00 《诛仙》
2 排片模块	13月32日99厅 25:00《诛仙仙》	时间地点名称无效类	无效
3 影片录入	《诛仙》2018年5月6日导演张三编剧李四出版方赵五等	有效类	录入成功
4 登录注册	账号123456 密码kkk666	有效类	登录成功/注册成功
5 登录注册	账号=123456 密码666666	账号中含有非法字符	登录/注册失败
6 个人信息管理	小张/159258/0706	修改名称/密码/生日	修改成功
7 个人信息管理	=小张/-15847/#5611	含有非法字符	修改失败
8 会员办理	123456	合法账号	办理成功
9 会员办理	+545214	非法账号	办理失败
10 浏览影片	战狼	有效类	战狼影片相关信息
11 浏览影片	战狼狼	无效类	不存在此电影
12 退票	2018/0106/9/15:00	时间/影厅/场次	退票成功
13 退票	2088/0106/9/15:00	无效时间	退票失败
14 订票	2018/0106/9/15:00	时间/影厅/场次	订票成功
15 订票	2088/0106/9/15:00	无效时间	订票失败
16 订单管理	查看 01/02 订单	有效类	15:00《战狼》
17 订单管理	查看 13/35 订单	无效时间	无效
18 电影评价	《战狼》: 666	有效评价	《战狼》: 666
19 电影评价	《诛仙》: sb	含有敏感词汇	无效评价
20 会员信息管理	删除 123456	有效类	删除此会员
21 会员信息管理	删除 213456	此会员不存在	删除无效
22 会员信息管理	添加 122456	有效类	添加成功
23 会员信息管理	添加 128456	已存在	添加失败
24 订单查询	02/01	有效类	123456: 15:00《战狼》

25 订单查询	13/35	无效类	查询无效
26 票房数据查询	01/03/战狼	时间/影片名称	12156 亿
27 票房数据查询	13/65/战狼狼	无效时间/无效名称	查询失败
28 数据结算查询	08/01	有效类	1.236 万
29 数据结算查询	13/36	无效类	查询失败
30 电影推荐	01/08 《战狼》《战狼 2》	有效时间/名称	《战狼》《战狼 2》
31 电影推荐	13/65 《战狼狼》《战狼狼 2》	无效时间/名称	无效推荐