<电影订票>

软件架构文档

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2023/3/26 | 1.0 | 初始版本 | 方嘉彤、曾永诚 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 参考资料 4

2. 用例视图 4

3. 逻辑视图 4

3.1 概述 4

3.2 在构架方面具有重要意义的设计包 4

4. 进程视图 4

5. 部署视图 4

6. 实现视图 5

7. 技术视图 5

8. 数据视图（可选） 5

9. 核心算法设计（可选） 5

10. 质量属性的设计 5

软件架构文档

# 简介

## 目的

本文档将从构架方面对系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面作出的重要决策。

## 参考资料

1. 电影订票系统前景文档，1.0版，6组
2. 电影订票系统第一次迭代计划，1.0版，6组
3. 电影订票系统细化迭代计划，1.0版，6组
4. 电影订票系统用例规范，1.0版，6组
5. 电影订票系统编程规范，1.0版，6组
6. 电影订票系统概念建模，1.0版，6组

# 用例视图

本章是对软件架构的用例视图的描述。由于电影购票系统的用例数量太多，因此本章只选了部分与

架构设计相关的用例。对于其余的用例，可参考软件用例规范。

选取的用例包括：

* 选择电影
* 购票选座
* 填写影评
* 支付
* 审核
* 生成票据
* 影评更新

主要用例：

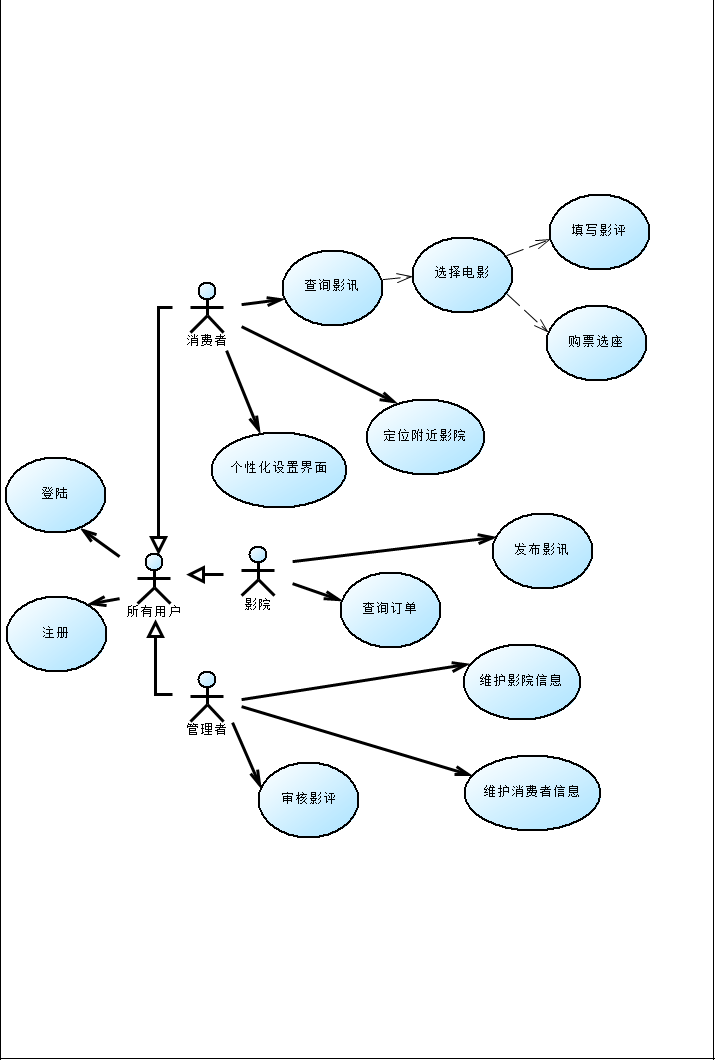


图1 电影购票系统用例视图

### **选择电影**

简要说明：此用例允许用户从系统检索将要上映的所有电影列表中选择一部电影，并选择执行相应的功能（购票选座、填写影评）。该用例的主要参与者是用户。

### **购票选座**

简要说明：此用例允许用户选择购票方式、座位、确认支付。该用例的主要参与者是用户。

### **填写影评**

简要说明：此用例允许用户打开评论区界面、检索所有电影评价、填写评价并提交待审核。该用例的主要参与者是用户。

### **支付**

简要说明：此用例允许用户选择系统支持的支付方式完成支付。该用例的主要参与者是用户。

### **审核**

简要说明： 此用例允许管理员审核用户新填写影评，并决定是否通过审核。该用例的主要参与者是系统的管理员。

### **生成票据**

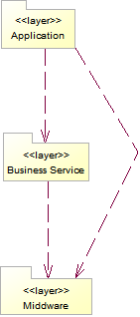
简要说明：系统按照票据模板自动生成电子票据。该用例的主要参与者是系统。

### **影评更新**

简要说明：系统将已通过的影评加入对应电影的评论区。该用例的主要参与者是系统。

# 逻辑视图

## 概述



电影购、订票系统的逻辑视图主要由三层组成，分别是Application层、Business Service层和Middle层。

Application层主要由响应各种用户界面请求的动作类组成，它会调用Business Service层中的函数进行业务逻辑处理，同时根据结果显示不同的界面给用户。

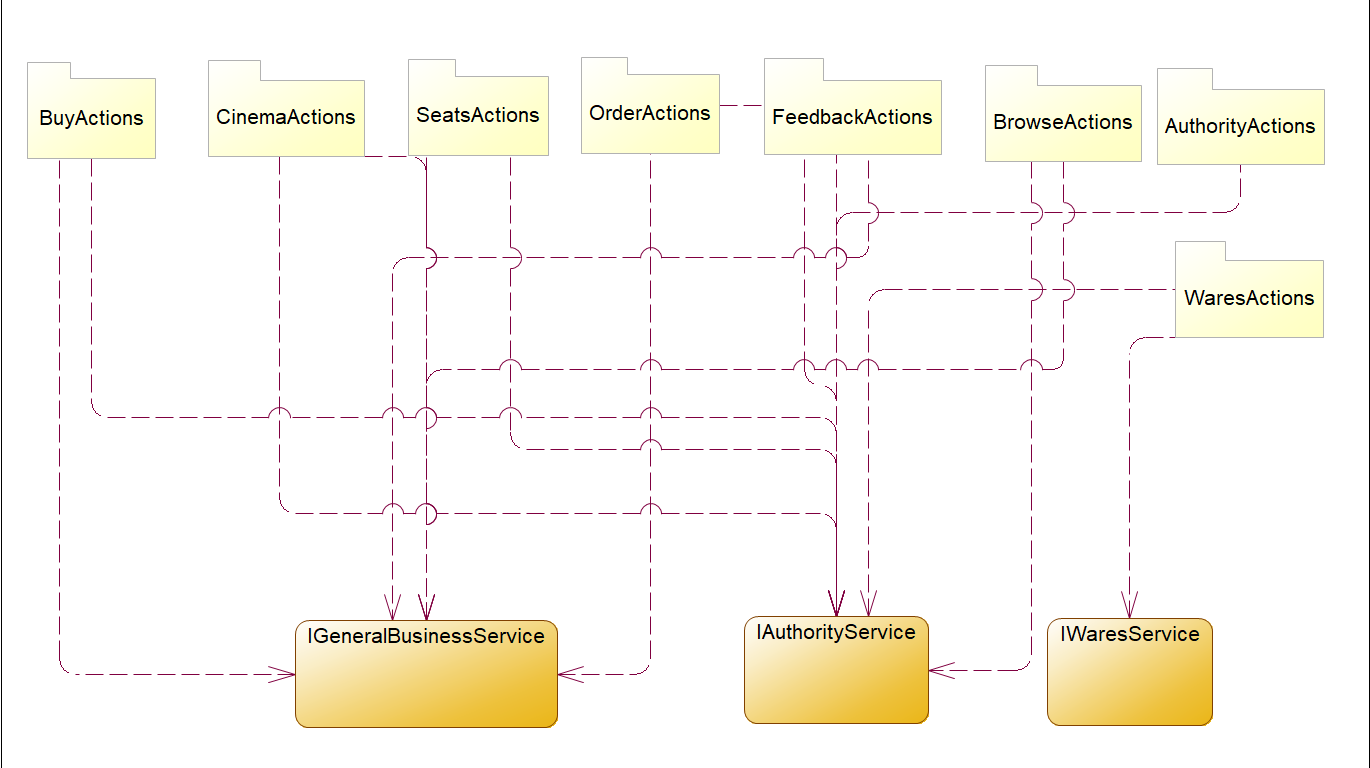
Business Service层主要完成实际的业务逻辑，同时包括与数据库的表对应的实体类，以及访问数据库的DAO类。

Middle层为SSH2框架的函数库。

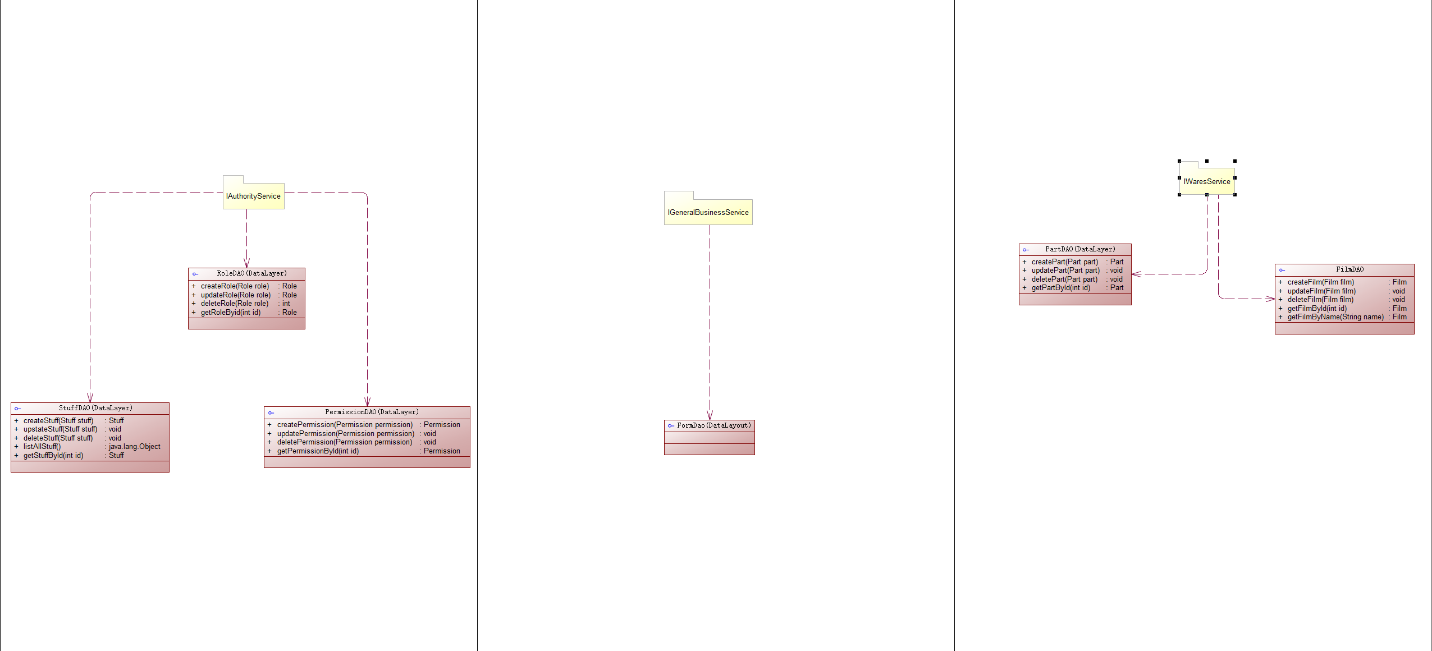
## 在构架方面具有重要意义的设计包

### Application 层

Application层主要由8个子包组成，这8个子包分别是BuyActions包、FeedbackActions包、CinemaActions包、SeatsActions包、OrderActions包、BrowseActions包、AuthorityActions包和WaresActions包。

****

### Business Service 层

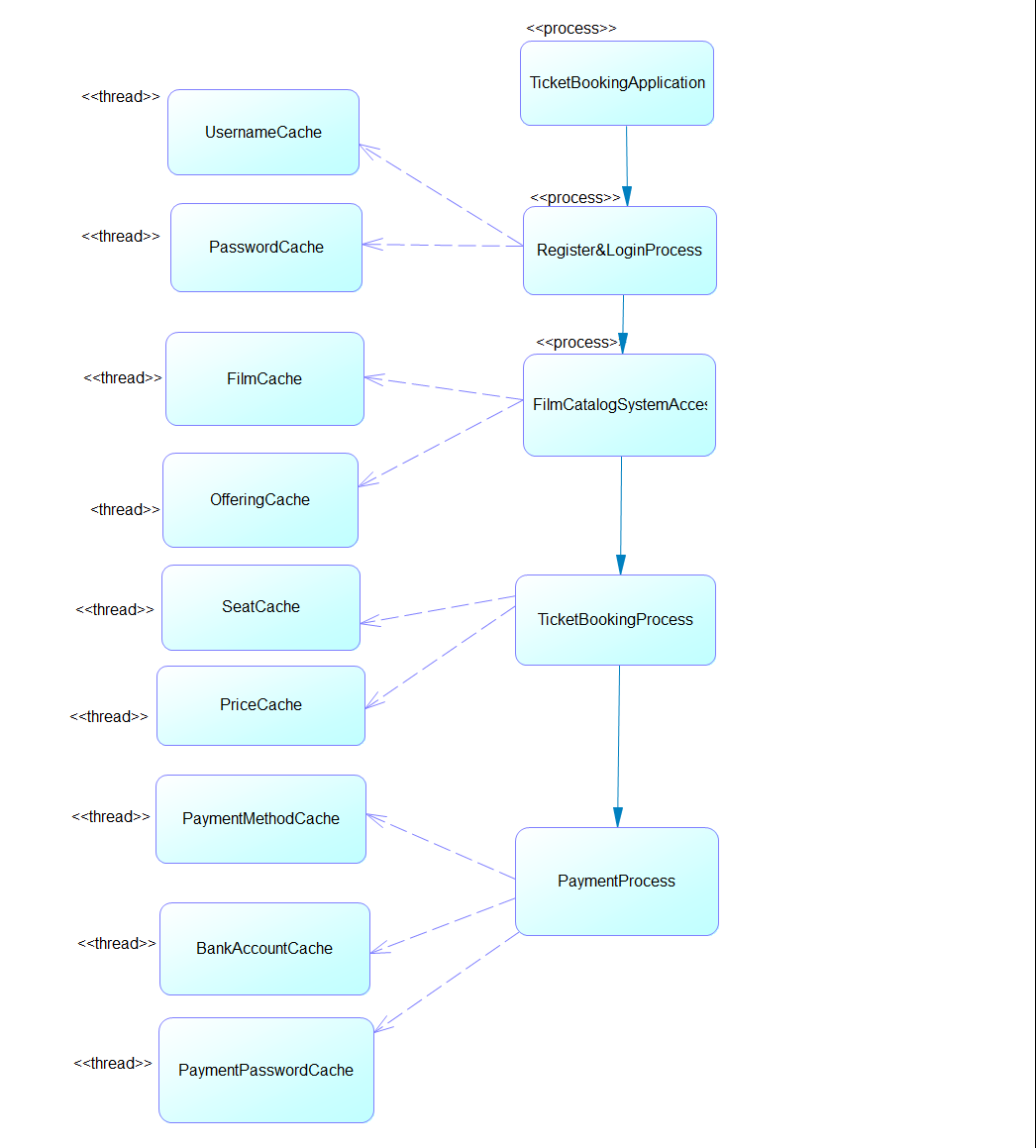
****

Service层由Service包构成。Service包主要由三个包组成，IAuthorityService包负责处理用户的身份认证以及角色和权限管理，IGeneralBusinessService包负责填写订单、订票购票以及信息反馈过程中的表单处理，IWaresService包负责电影信息、影院信息的管理以及座位信息。

### Middleware 层

****

# 进程视图



## FilmCatalogSystemAccess

此过程管理对电影目录系统的访问。它可以由注册账户的多个用户共享。这允许缓存最近检索到的电影、影院和订单，以提高性能。FilmCache,OfferingCache中的独立线程用于从遗留系统异步检索项目。

## Register&LoginProcess

对于目前正在注册或者登录账户的每个用户，都有一个这个过程的例子。

## TicketBookingProcess

TicketBookingProcess处理的是购票过程中对座位和价格的操作，包括SeatCache和PriceCache。

## PaymentProcess

PaymentProcess主要处理的是支付订单过程中选择支付方式、输入支付密码、选择银行卡的操作，包括PaymentMethodCache,BankAccountCache,PaymentPasswordCache。

# 部署视图 采用了SSH的架构，数据库选择mysql，server选择tomcat，定位服务选择高德地图web service； 移动用户发送http请求到服务器，服务器响应并发送请求到mysql数据库或web service。

# 实现视图

[本节说明实现模型的整体结构、软件分解为实现模型中的层和子系统的情况，以及所有在构架方面具有重要意义的构件。]

详见3.逻辑试图。

# 技术视图

[技术栈的选型，包括编程语言、开发工具、框架、数据库、中间件等。]

1. 环境：jdk 1.8+
2. 开发工具：idea、vscode、android studio
3. 硬件环境：windows 10，android系统6.0以上版本

4.框架：用户端：安卓原生开发；  
 服务端：SpringMVC

5.数据库：mysql 8.0

# 数据视图（可选）

暂无。

# 核心算法设计（可选）

暂无。

# 质量属性的设计

1.支持100个并发用户，响应时间不超过3秒；  
2.支持500个电影的信息储存；  
3.支持1000个场次的信息储存。