<聚票网>

软件架构文档

版本 <1.2>

[注：用方括号括起来并以蓝色斜体（样式=InfoBlue）显示的文本，它们用于向作者提供指导，在发布此文档之前应该将其删除。按此样式输入的段落将被自动设置为普通样式（样式=Body Text）。]

[要定制 Microsoft Word 中的自动字段（选中时显示灰色背景），请选择 File>Properties，然后将 Title、Subject 和 Company 等字段替换为此文档的相应信息。关闭该对话框后，通过选择 Edit>Select All（或 Ctrl-A）并按 F9，或只是在字段上单击并按 F9，可以在整个文档中更新自动字段。对于页眉和页脚，这一操作必须单独进行。按 Alt-F9，将在显示字段名称和字段内容之间切换。有关字段处理的详细信息，请参见 Word 帮助。]

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <14/7/2018> | <1.0> | <完成了用例图、逻辑视图、数据视图、部分部署视图> | <顾一辉、原帅> |
| <26/7/2018> | <1.1> | <完成了逻辑视图对包的描述> | <茅悦田> |
| <26/7/2018> | <1.2> | <完成了实现视图，对数据库PDM进行改进> | <顾一辉> |
|  |  |  |  |

目录

1. 简介 2

1.1 目的 2

1.2 参考资料 2

2. 用例视图 2

3. 逻辑视图 2

3.1 概述 2

3.2 在构架方面具有重要意义的设计包 2

4. 进程视图 2

5. 部署视图 2

6. 实现视图 2

7. 数据视图（可选） 2

8. 核心算法设计（可选） 2

软件架构文档 （简化版）

# 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。

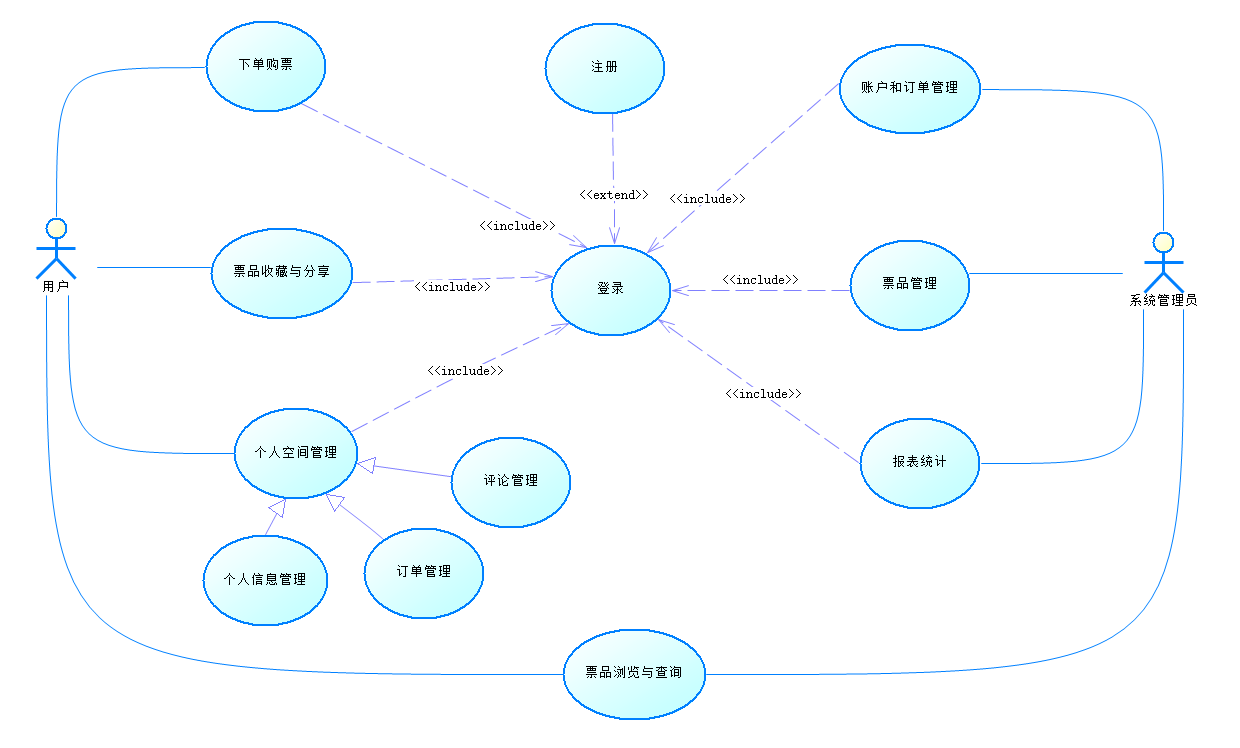
[本节确定此**软件构架文档**在整个项目文档中的作用或目的，并对此文档的结构进行简要说明。应确定此文档的特定读者，并指出他们应该如何使用此文档。]

## 参考资料

[本小节应完整地列出此**软件构架文档**中其他部分所引用的所有文档。每个文档应标有标题、报告号（如果适用）、日期和出版单位。列出可从中获取这些参考资料的来源。这些信息可以通过引用附录或其他文档来提供。]

# 用例视图

[本节列出用例模型中的一些用例或场景，这些用例或场景应体现最终系统中重要的、核心的功能；或是在构架方面涉及范围很广（使用了许多构架元素）；或强调或阐明了构架的某一具体的细微之处。]



# 逻辑视图

[本节说明设计模型在构架方面具有重要意义的部分，例如设计模型被分解为多个子系统和包。而每个重要的包又被分解为多个类和类实用程序。您应该介绍那些在构架方面具有重要意义的类，并说明它们的职责，以及几项非常重要的关系、操作和属性。]

## 概述

[本节按照设计模型中包的层次结构来说明设计模型的整体分解情况。]



## 在构架方面具有重要意义的设计包

[对于每个重要的包，都用一个小节来加以说明，其中应包括该包的名称、简要说明以及显示该包中所有重要的类和包的图。

对于该包中的每个重要类，应包括其名称、简要说明，还可选择包括对其部分主要职责、操作和属性的说明。]

### 管理系统UI

主类为Admin Space，其余为提供详细功能的子组件。

主要功能包括用户管理、票品管理、退款审核、销量统计。

### 个人空间UI

主类为Info Space，其余为提供详细功能的子组件。

功能分为三大类：订单管理、动态管理、账号管理。其中，订单管理包括购物车、优惠券、当前及历史订单，当前订单支持退款申请；动态管理分为收藏和评论两部分，展示自己收藏的票品和自己发表过的评论；账号设置包括基本信息管理、密码管理和收货地址管理，可以预先添加地址及收货信息，在购买时快速使用。

### 票品操作UI

主类为Good Detail Page，其余为票品选择、详情、评论、购买的实现组件。

票品选择包括日期、时间、票档、数量的选择，同一页面上还包括详情介绍和相关评论的展示；在选择确定后，进入分步购买页面，由Buy Step组件实现。

### 通用构件

该包主要包括了一些可复用组件和通用浏览页面。

可复用组件包括Header Menu、Footer、Result List等。其中，Header Menu集成了搜索、登录、注册等功能；Result List则负责将后台取到的数据以列表形式进行展示，用于多个票品的展示。

通用浏览页面包括主页、分类搜索、网站相关、帮助等页面。其中，主页提供了根据不同票务类型进行快速搜索的导航栏以及推荐功能；分类搜索支持多条件搜索，如：城市、类型、时间等。

**Controller:**

### 管理业务

包括了管理员相关的controller。

包括了管理员管理用户controller、管理员管理票品controller、管理员审核退款申请controller。

### 个人空间

和用户数据表相关的controller

包括用户对在个人空间中所能看到的信息的操作（个人信息管理、修改密码、收货地址管理controller），还有登录注册等需要访问用户数据库的controller

### 票品展示

包含了浏览界面和票品详情的controller

主要功能是在后端联系show、ticket等数据表组合成前端所需的数据，再发回前端，还包括从评论数据库中拿取各种票品对应的评论并发回前端

### 订单管理

包括下单、查看、退款等相关controller

查询订单主要包括在后端组合show、ticket、user、sendingAddr等多张表的数据并组成前端所需要的数据格式发回，退款、下单等操作则结合前端订单所需数据发挥后端，创建订单数据或改变数据库中已有订单的状态，管理员管理退款申请操作会在改变orders表中订单的状态，并在refund表中记录下对退款申请的处理记录。

**Service:**

### 登录

包括初次登录、每次请求的登录状态验证。

### 购物车

进行购物车的管理，包括添加、删除票品和购物车的展示。

### 用户信息

进行用户数据的显示和操作，包括管理员的维护（编辑、删除、禁用）和普通用户的注册（验证、添加）和管理（修改个人信息、修改密码、修改收货地址）。

### 评论

分为我的评论和票品评论，分别根据user id和show id获取评论并展现。另外还包括评论的发表。

### 票品

进行票品相关业务的处理，其中包括：分享、票品按条件搜索（包括相关评论数）、我的收藏（根据user id取数据）、票品具体信息和管理。

### 销量

进行销量和报表的统计。

### 订单

覆盖订单的创建流程，包括优惠券的展示和使用，以及订单创建完成后的展示。

### 退款

实现退款的业务逻辑，包括退款的申请和审核。

### DAO

每个表对应一个Repository，进行数据的直接操作。

### Domain

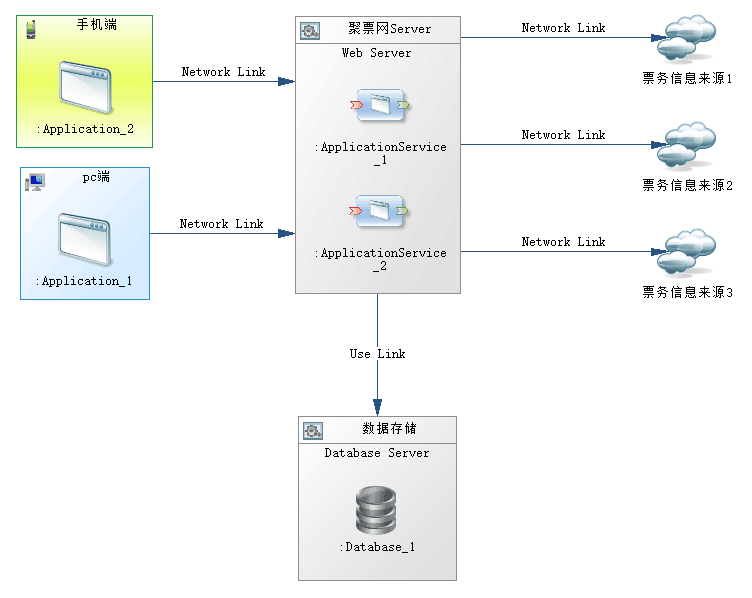
包含所有实体类。

# 进程视图

[本节说明将系统分解为轻量级进程（单个控制线程）和重量级进程（成组的轻量级进程）的情况。本节的内容按照各个通信或交互的进程组来进行组织。说明进程之间的主要通信模式，例如消息传递、中断和会合。]

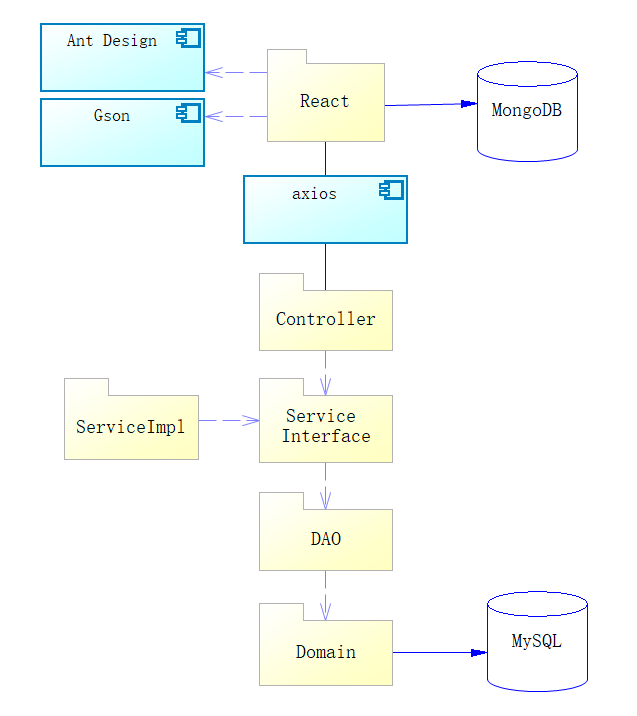
# 部署视图

[本节说明用来部署和运行该软件的一种或多种物理网络（硬件）配置。对于每种配置，它至少应该指出执行该软件的物理节点（计算机、CPU）及其互连情况（总线连接、LAN 连接、点到点连接等）。另外还要包括**进程视图**中的各进程到物理节点的映射。]



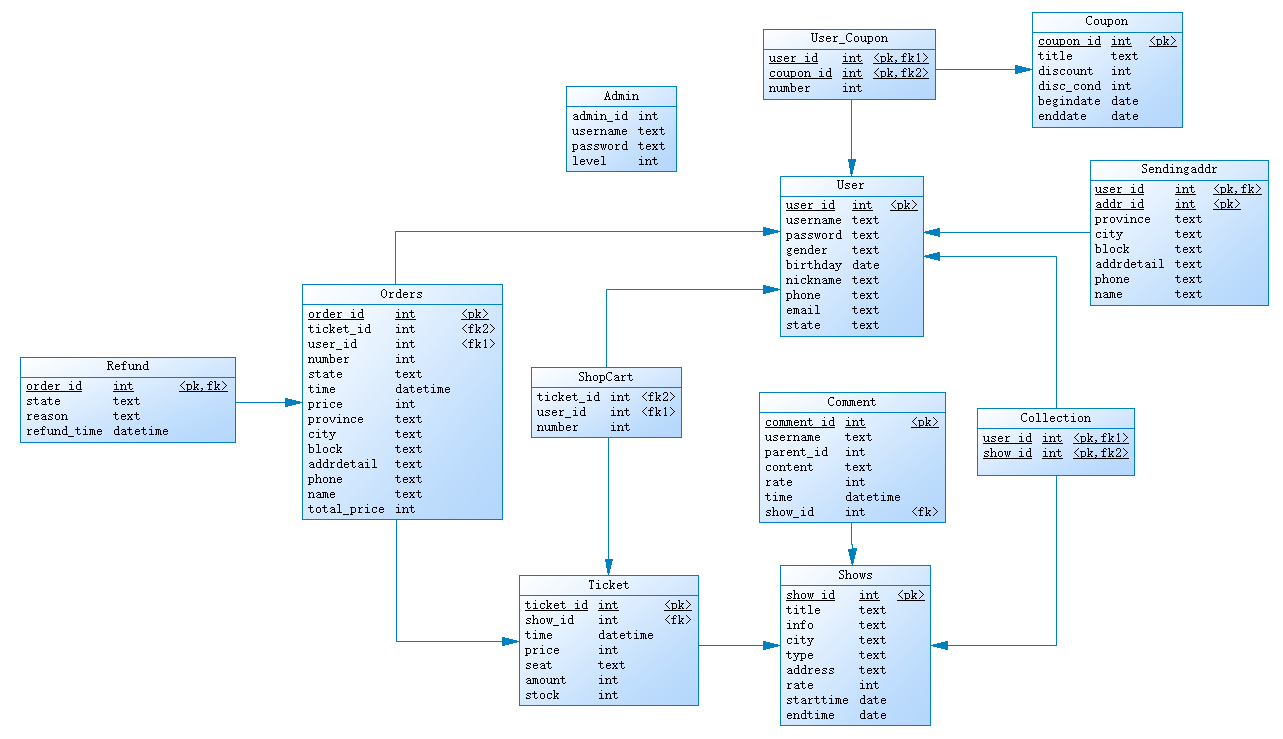
# 实现视图

[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。]



# 数据视图（可选）

[从永久性数据存储方面来对系统进行说明。如果几乎或根本没有永久性数据，或者设计模型与数据模型之间的转换并不重要，那么本节就为可选。]



# 核心算法设计（可选）

[对系统中的核心算法进行设计。如果没有什么重要的算法，那么本节就为可选。]