目录

[1导言 2](#_Toc11850461)

[1.1 目的 2](#_Toc11850462)

[1.2 范围 2](#_Toc11850463)

[1.3 编写说明 2](#_Toc11850464)

[1.4 术语定义 2](#_Toc11850465)

[1.5 引用标准 2](#_Toc11850466)

[1.6 参考资料 2](#_Toc11850467)

[1.7 版本更新信息 2](#_Toc11850468)

[2系统定义 2](#_Toc11850469)

[2.1 项目来源及背景 2](#_Toc11850470)

[2.2 用户的特点 2](#_Toc11850471)

[2.3 项目目标 2](#_Toc11850472)

[3应用环境 3](#_Toc11850473)

[3.1 系统运行的网络环境 3](#_Toc11850474)

[3.2 系统运行的硬件环境 3](#_Toc11850475)

[4功能规格 3](#_Toc11850476)

[4.1 角色定义 3](#_Toc11850477)

[4.1.1 旅客 3](#_Toc11850478)

[4.1.2航空公司管理员 3](#_Toc11850479)

[4.1.3 柜员 4](#_Toc11850480)

[4.2 系统主用例图 4](#_Toc11850481)

[4.3 客户端子系统 4](#_Toc11850482)

[4.3.1 注册 5](#_Toc11850483)

[4.3.2 用户登录 5](#_Toc11850484)

[4.3.3 查询航班 5](#_Toc11850485)

[4.3.4 管理订单 5](#_Toc11850486)

[4.3.5 管理用户信息 5](#_Toc11850487)

[4.4 管理端子系统 5](#_Toc11850488)

[4.4.1管理员登录 6](#_Toc11850489)

[4.4.2管理用户信息 6](#_Toc11850490)

[4.4.3管理航班信息 6](#_Toc11850491)

[4.4.4办理登机牌 6](#_Toc11850492)

[4.4.5行李托运 6](#_Toc11850493)

[5性能需求 6](#_Toc11850494)

[5.1 界面需求 6](#_Toc11850495)

[5.2 响应时间需求 6](#_Toc11850496)

[5.3 可靠性需求 6](#_Toc11850497)

[5.4 开放性需求 6](#_Toc11850498)

[5.5 可扩展性需求 6](#_Toc11850499)

[5.6 系统安全性需求 6](#_Toc11850500)

[6产品提交 6](#_Toc11850501)

[7实现约束 6](#_Toc11850502)

# 1导言

## 1.1 目的

## 1.2 范围

## 1.3 编写说明

## 1.4 术语定义

## 1.5 引用标准

## 1.6 参考资料

## 1.7 版本更新信息

# 2系统定义

## 2.1 项目来源及背景

## 2.2 用户的特点

## 2.3 项目目标

本项目设定的目标如下：

· 系统能够提供友好的用户界面，使操作人员的工作量最大限度的减少；

· 系统具有良好的运行效率，能够达到提高生产率的目的；

· 系统应有良好的可扩充性，可以容易地加入其他系统的应用；

· 平台的设计具有一定的超前性，灵活性，能够适应企业生产配置的变化；

· 通过这个项目可以锻炼队伍，提高团队的开发能力和项目管理能力。

# 3应用环境

## 3.1 系统运行的网络环境

## 3.2 系统运行的硬件环境

# 4功能规格

我们采用面向对象分析作为主要的系统建模方法，使用UML（Unified Modeling Language）作为建模语言。UML为建模活动提供了从不同角度观察和展示系统的各种特征的方法。在UML中，从任何一个角度对系统所作的抽象都可能需要几种模型来描述，而这些来自不同角度的模型图最终组成了系统的映像。

用例描述角色（用户、外部系统以及系统处理）是如何与系统交互来完成工作的。用例模型提供了一个非常重要的方式来界定系统边界以及定义系统功能，同时，该模型将来可以派生出动态对象模型。

设计用例时，我们遵循下列步骤：

1）识别出系统的角色。角色可以是用户、外部系统，甚至是外部处理，通过某种途径与系统交互。重要的是着重从系统外部执行者的角度来描述系统需要提供哪些功能，并指明这些功能的执行者（角色）是谁。尽可能地确保所有角色都被完全识别出来。

2）描述主要的用例。可以采取不断地问自已“这个角色究竟想过系统做什么？”来准确地描述用例。

3）重新审视每个用例，为它们下个详尽的定义。

## 4.1 角色定义

角色或者执行者指与系统产生交互的外部用户或者外部系统。

### 4.1.1 旅客

旅客是指在这个系统中通过客户端进行订票等操作的人员。这个角色主要参与客户端的查询航班，管理订单，查询修改自己的信息等功能。

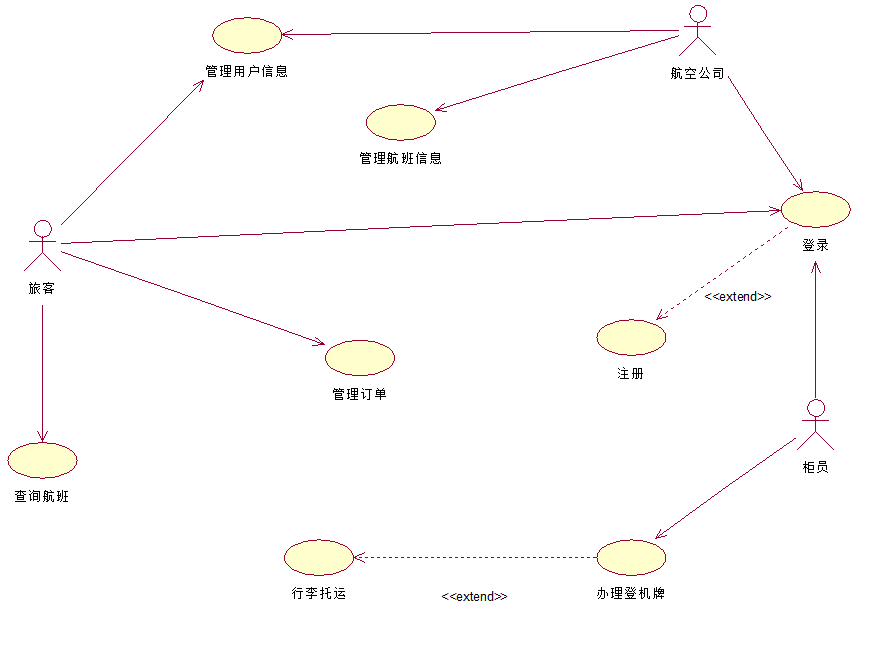
### 4.1.2航空公司管理员

航空公司管理员是管理端的用户，主要负责管理用户信息，编排航班，修改航班信息，取消航班等功能。

### 4.1.3 柜员

柜员是管理端的用户，主要负责办理登机牌，行李托运等功能。

## 4.2 系统主用例图



## 4.3 客户端子系统

旅客通过订票系统的网站查询航班信息，登录后进行预订机票，管理订单，管理自己的信息，这是客户端子系统的功能。在客户端用户可以通过出发城市，到达城市，往返时间查询航班信息，对自己的身份信息进行修改，修改和删除常用旅客信息，查看自己的历史订单，当前订单，在一定的时间范围内取消订单。

图

客户端管理的这些用例描述如下：

**F-R-1：注册**

用户登录前，若没有账户，需要注册，填写用户信息并提交。

**F-L-1：用户登录**

用户执行管理订单，管理用户信息的操作前，必须经过登录界面。输入自己的用户用和密码，校验通过后进入对应界面。

**F-C-1：查询航班**

旅客进入到查询航班的界面，输入出发城市，到达城市，选择出发日期，往返日期，点击搜索按钮，系统会返回相应的搜索结果列表。

**F-C-2：管理订单**

包括取消订单，查看订单，修改订单/\*旅客信息，座位信息，改签\*/，生成订单/\*包括选座\*/，付款。所有的订单操作必须登录后进行。

**F-C-3：管理用户信息**

包括修改自身账户信息，修改常用旅客信息。所有用户信息的操作必须登录后进行。

### 4.3.1 注册

### 4.3.2 用户登录

### 4.3.3 查询航班

### 4.3.4 管理订单

### 4.3.5 管理用户信息

## 4.4 管理端子系统

**F-L-1：管理员登录**

登录管理负责所有的管理端的登录，管理端的人员要登录到管理端必须经过登录界面，输入自己的用户名和密码，然后系统判断这个用户的权限信息，不同的登录人可能具有不同的权限，系统根据不同的权限实现不同的功能。

**F-M-1：管理用户信息**

管理员登录到系统后，根据权限的不同，可以执行查询用户信息，修改部分用户信息/\*，删除用户信息的操作\*/。

**F-M-2：管理航班信息**

管理员登录到系统后，可以对航班进行安排，包括增加航班，查询航班信息，修改航班信息，删除航班/\*包括邮件通知用户\*/。

**F-M-3：办理登机牌**

柜员登录到系统后，可以执行办理登机牌的操作。

**F-M-4：行李托运**

柜员登录到系统后，办理完成登机牌之后，可以执行行李托运的操作。

### 4.4.1管理员登录

### 4.4.2管理用户信息

### 4.4.3管理航班信息

### 4.4.4办理登机牌

### 4.4.5行李托运

# 5性能需求

根据用户对本系统的要求，确定系统在响应时间、可靠性、安全性等方面有较高的必能要求。

## 5.1 界面需求

## 5.2 响应时间需求

## 5.3 可靠性需求

## 5.4 开放性需求

## 5.5 可扩展性需求

## 5.6 系统安全性需求

# 6产品提交

# 7实现约束