

---

# 旅行服务管理系统 概要设计

---

	人员	日期
拟制	李双利	2018-05-19
评审人	•	yyyy-mm-dd
批准	•	yyyy-mm-dd
签发	•	yyyy-mm-dd

## 摘 要

本文是关于一套旅行服务管理系统的设计文档内容，包括旅游服务管理系统的任务概述、总体设计、接口设计、数据结构设计、数据库设计、界面设计、出错处理设计和维护设计等详细描述。在于为开发和维护人员提供参考手册。本文档的预期读者为系统设计人员、软件开发人员、客户方的系统设计人员和项目评审人员。

**关键词：**软件工程    旅行服务    USTC    设计说明    管理系统    Web 开发

表 1    缩略词清单

缩略语	英文全名	中文解释
DB	DataBase	数据库
WTF	WTForms	一组提供表单 API 接口的库
TSMS	Travel Service Management System	旅行服务管理系统
DBA	Database administration	数据库管理员
UI	User Interface	用户接口
WSGI	Web Server Gateway Interface	Web 服务器网关接口

# 目 录

摘要 .....	
第 1 章 引言 .....	7
1.1 编写目的 .....	7
1.2 项目背景 .....	7
1.3 术语 .....	8
第 2 章 任务概述 .....	9
2.1 目标 .....	9
2.2 开发与运行环境 .....	9
2.2.1 开发环境的配置 .....	9
2.2.2 测试环境的配置 .....	10
2.2.3 运行环境的配置 .....	10
2.3 需求概述 .....	10
2.3.1 R.INTF.CALC.001 酒店预定 .....	11
2.3.2 R.INTF.CALC.002 火车票预订 .....	12
2.3.3 R.INTF.CALC.003 机票预订 .....	13
2.3.4 R.INTF.CALC.004 门票预订 .....	14
2.3.5 R.INTF.CALC.005 广告推送 .....	15
2.3.6 R.INTF.CALC.006 用户评价 .....	16
2.3.7 R.INTF.CALC.007 优惠券 .....	17
2.3.8 R.INTF.CALC.008 活动推送 .....	18
2.3.9 R.INTF.CALC.009 客服服务 .....	19
2.3.10 R.INTF.CALC.010 用户登录 .....	20
2.3.11 R.INTF.CALC.011 注册用户/修改个人信息 .....	20
2.4 条件与限制 .....	22
2.4.1 开发平台与工具 .....	22
2.4.2 软件开发生命周期模型 .....	22
2.4.3 法律 .....	22
2.4.4 技术 .....	22
2.4.5 经费 .....	22

第 3 章 总体设计	23
3.1 软件描述	23
3.1.1 前台主要功能	23
3.1.2 后台主要功能	23
3.2 处理流程	24
3.2.1 总体流程	24
3.2.2 系统基本流程	24
3.2.3 客户端基本流程	25
3.2.4 服务器端基本流程	27
3.2.5 功能 1——注册和登录验证具体流程	28
3.2.6 功能 2——用户预订/领取具体流程	28
3.2.7 功能 3——用户提交评论具体流程	28
3.2.8 功能 4——管理员发布信息具体流程	28
3.2.9 功能 5——客户服务具体流程	28
3.2.10 功能 6——个性化推荐系统具体流程	29
3.3 功能结构设计	29
3.3.1 整体结构	29
3.3.2 用户端结构	29
3.3.3 服务器端结构	31
3.3.4 后台数据库维护模块结构	31
3.4 功能需求与程序代码的关系	33
第 4 章 接口设计	34
4.1 外部接口	34
4.1.1 支付宝接口	34
4.1.2 携程接口	34
4.1.3 用户接口	34
4.2 内部接口	34
4.2.1 酒店/火车票/航班/景点模块与服务器逻辑控制模块	35
4.2.2 客户服务模块与服务器逻辑控制模块	35
4.2.3 服务器逻辑控制模块与数据库维护模块	35
4.2.4 推荐系统模块与数据库维护模块	35
第 5 章 数据结构设计	36
5.1 逻辑结构设计	36
5.1.1 用户管理系统数据结构设计	36

5.1.2 客户端数据结构 .....	36
5.1.3 服务器端数据结构 .....	38
5.2 物理结构设计 .....	39
第 6 章 数据库设计 .....	40
6.1 数据库环境说明 .....	40
6.2 数据库的命名规则 .....	40
6.3 逻辑设计 .....	41
6.4 物理设计 .....	42
6.4.1 数据库产品 .....	42
6.4.2 实体属性、类型、精度 .....	42
6.5 安全性设计 .....	47
6.6 数据库管理与维护说明 .....	47
第 7 章 界面设计 .....	48
7.1 注册界面 .....	48
7.2 登录界面 .....	49
7.3 菜单界面 .....	50
7.4 主页界面 .....	51
7.5 航班预订界面 .....	52
7.6 火车票预订界面 .....	53
7.7 酒店预订界面 .....	54
7.8 景点预订界面 .....	55
7.9 优惠券领取界面 .....	56
7.10 个人信息界面 .....	57
7.11 预订信息界面 .....	58
7.12 编辑信息界面 .....	59
第 8 章 出错处理设计 .....	60
8.1 数据库出错处理 .....	60
8.2 模块失效处理 .....	60
8.2.1 酒店/航班/火车票/景点模块 .....	60
8.2.2 登录/注册模块 .....	61
8.2.3 优惠券模块 .....	61
8.2.4 活动推送模块 .....	61

8.2.5 客服模块.....	61
8.2.6 后台数据库维护模块.....	61
8.2.7 后台逻辑控制模块.....	61
8.2.8 推荐系统模块.....	62
 第 9 章 安全保密设计.....	 63
9.0.1 保密性.....	63
9.0.2 加密传输.....	63
 第 10 章 维护设计.....	 64
10.0.1 备份.....	64
10.0.2 维护.....	64

## 图目录

3.1 系统基本流程 .....	24
3.2 系统基本流程 .....	25
3.3 客户端基本流程 .....	26
3.4 服务器端基本流程 .....	27
3.5 整体结构 .....	29
3.6 用户端结构 .....	30
3.7 服务器端结构 .....	31
3.8 后台数据库维护模块结构 .....	32
6.1 逻辑设计图 .....	41
7.1 注册界面 .....	48
7.2 登录界面 .....	49
7.3 菜单界面 .....	50
7.4 主页界面 .....	51
7.5 航班预订界面界面 .....	52
7.6 火车票预订界面 .....	53
7.7 酒店预订界面 .....	54
7.8 景点预订界面 .....	55
7.9 优惠券领取界面 .....	56
7.10 个人信息界面 .....	57
7.11 火车票预订界面 .....	58
7.12 编辑信息界面 .....	59

## 表目录

1 缩略词清单 .....	
1.1 术语表 .....	8
2.1 开发环境的配置 .....	9
2.2 测试环境的配置 .....	10
2.3 运行环境的配置 .....	10
3.1 功能需求与程序代码的关系表 .....	33
6.1 活动数据表 Ads 设计 .....	42
6.2 用户数据表 Users 设计 .....	42
6.3 优惠券数据表 Coupons 设计 .....	43
6.4 用户评论数据表 Marks 设计 .....	43
6.5 航班数据表 Flights 设计 .....	44
6.6 火车票数据表 Trains 设计 .....	45
6.7 酒店数据表 Hotels 设计 .....	45
6.8 景点数据表 Attractions 设计 .....	46
6.9 用户领取优惠券数据表 MyCoupons 设计 .....	46
6.10 用户预订据表 Reservations 设计 .....	47



## 第1章 引言

### 1.1 编写目的

在本项目的前一阶段，也就是需求分析阶段，已经将系统用户对本系统的需求做了详细的阐述，这些用户需求已经在上一阶段中对不同用户所提出的不同功能，实现的各种效果做了调研工作，并在需求规格说明书中得到详尽得叙述及阐明。

本阶段已在系统的需求分析的基础上，对即时聊天工具做概要设计。主要解决了实现该系统需求的程序模块设计问题。包括如何把该系统划分成若干个模块、决定各个模块之间的接口、模块之间传递的信息，以及数据结构、模块结构的设计等。在以下的概要设计报告中将对在本阶段中对系统所做的所有概要设计进行详细的说明，在设计过程中起到了提纲挈领的作用。

在下一阶段的详细设计中，程序设计员可参考此概要设计报告，在概要设计即时聊天工具所做的模块结构设计的基础上，对系统进行详细设计。在以后的软件测试以及软件维护阶段也可参考此说明书，以便于了解在概要设计过程中所完成的各模块设计结构，或在修改时找出在本阶段设计的不足或错误。

### 1.2 项目背景

随着社会的不断发展和人们对于生活质量的要求日益提高，旅游出行几乎成为每个人经常性的活动，而且随着互联网的移动网络的飞速发展，人们越来越热衷于从线上进行预订门票、车票，安排出游活动，这一高效简便的选择大大节约了人们的时间，同时有了更多更好的选择。通过互联网在线选择旅行的各项需求，用户不用去到旅行社就能获得优质的在线旅行服务，世界各地景点的门票、乘坐的航班甚至各个酒店的详细信息、优惠券信息都可以实时展现给用户，线上的旅行服务管理系统能带给用户全方位的选择和优质的旅行服务体验，因此本项目基于这一出发点，从用户日常进行旅游的需求出发，开发一个提供高效服务和方便管理维护的旅行服务管理系统。

## 1.3 术语

[列出本文档中所用到的专门术语的定义和外文缩写的原词组]

表 1.1 术语表

缩写、术语	解释
SQL	SQL 指结构化查询语言, 使我们有能力访问数据库
Bootstrap	Bootstrap 是基于 HTML、CSS、JavaScript 的一种前端框架
gunicorn	一种 wsgi 容器, 只支持在 Unix 系统上运行, 来源于 Ruby 的 unicorn 项目

## 第 2 章 任务概述

本系统的目标是实现一个旅行服务管理系统，包括 web 客户端（包含管理员）、服务器端两个部分。

基于 web 的客户端面向旅行用户，为用户提供注册登录，机票、火车票、酒店以及门票预订服务，优惠券、活动推送服务以及客服解答服务。

### 2.1 目标

实现旅行服务管理系统，实现需求规格说明书中所描述的新用户注册和用户登录功能，支持在线预订门票、酒店、火车票和机票功能，用户评价及评分功能，优惠券和活动推荐功能以及客服功能，并且保证系统的健壮性和数据安全。

### 2.2 开发与运行环境

#### 2.2.1 开发环境的配置

表 2.1 开发环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version $\geq 4.10$ ) GNU gcc (version $\geq 6.3.1$ ) python (version $\geq 2.7$ ) PyCharm 2017.3.3	Linux (kernel version $\geq 3.10$ ) GNU gcc (version $\geq 5.4$ ) python (version $\geq 2.7$ ) PyCharm 2016.3.3
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

## 2.2.2 测试环境的配置

表 2.2 测试环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version $\geq 4.10$ ) Python (version $\geq 2.7$ )	Linux (kernel version $\geq 3.10$ ) Python (version $\geq 2.7$ )
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	采用 MySQL 数据库	采用 MySQL 数据库

## 2.2.3 运行环境的配置

表 2.3 运行环境的配置

类别	标准配置	最低配置
计算机硬件	基于 x86 结构的 CPU 主频 $\geq 2.4\text{GHz}$ 内存 $\geq 8\text{G}$ 硬盘 $\geq 200\text{G}$	基于 x86 结构的 CPU 主频 $\geq 1.6\text{GHz}$ 内存 $\geq 512\text{M}$ 硬盘 $\geq 2\text{G}$
计算机软件	Linux (kernel version $\geq 4.10$ ) Chrome 66.0.3359.139	Linux (kernel version $\geq 3.10$ ) IE 8 以上
网络通信	至少要有一块可用网卡 能运行 IP 协议栈即可	至少要有一块可用网卡 能运行 IP 协议栈即可
其他	支持 bootstrap 框架	支持 bootstrap 框架

## 2.3 需求概述

在需求分析报告的基础上稍作修改和扩展，功能需求包括：

## 2.3.1 R.INTF.CALC.001 酒店预定

### 2.3.1.1 介绍

通过搜索酒店名称得到相关信息，并且支持预订功能。输入酒店不存在将作为无效输入，在软件中将显示酒店不存在这一信息。不存在错误输入

### 2.3.1.2 输入

输入来源：用户

数量：2，一个预订选项按钮/一个搜索字符串

度量单位：次数

时间要求：无

包含精度和容忍度的有效输入范围：1-20 个字符

### 2.3.1.3 处理

输入数据的有效性检测：字符及字符串长度

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 数据库查询-> 结果显示

对异常情况的回应：通信失败则输出查询失败的信息

用于把系统输入转换到相应输出的任何方法：数据库查询，映射

对输出数据的有效性检测：存在性以及数据长度

### 2.3.1.4 输出

输出目的地：前端

数量：多个，酒店名称，价格及其他详细信息

度量单位：价格/RMB

时序：接收数据-> 前端整合渲染

包含精确度和容忍度的有效输出范围：整数，范围：>0

对非法值的处理：筛选并返回说明

错误消息：错误提示信息

## 2.3.2 R.INTF.CALC.002 火车票预订

### 2.3.2.1 介绍

通过输入目的地以及时间搜索得到相关火车票信息，并且支持预订功能。输入目的地不存在将作为无效输入，在软件中将显示无相关火车票这一信息。不存在错误输入

### 2.3.2.2 输入

输入来源：用户

数量：3，目的地出发地/时间/预订按钮

度量单位：次数

时间要求：无

包含精度和容忍度的有效输入范围：时间用年月日表示，yyyy-mm-dd 的格式，均为正整数，地点为 1—20 个字符的字符串

### 2.3.2.3 处理

输入数据的有效性检测：检查时间格式是否正确，字符串长度

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 数据库查询-> 结果显示

对异常情况的回应：通信失败则输出查询失败的信息

用于把系统输入转换到相应输出的任何方法：数据库查询，映射

对输出数据的有效性检测：存在性以及数据长度

### 2.3.2.4 输出

输出目的地：前端

数量：多个，出发地-> 目的地/时间/价格/座位数等信息

度量单位：价格/RMB，个数等

时序：接收数据-> 前端整合渲染

包含精确度和容忍度的有效输出范围：整数，范围：>0

对非法值的处理：筛选并返回说明

错误消息：错误提示信息

### 2.3.3 R.INTF.CALC.003 机票预订

#### 2.3.3.1 介绍

通过输入目的地以及时间搜索得到相关机票信息，并且支持预订功能。输入目的地不存在将作为无效输入，在软件中将显示无相关机票这一信息。不存在错误输入

#### 2.3.3.2 输入

输入来源：用户

数量：3，目的地出发地/时间/预订按钮

度量单位：次数

时间要求：无

包含精度和容忍度的有效输入范围：时间用年月日表示，yyyy-mm-dd 的格式，均为正整数，地点为 1—20 个字符的字符串

#### 2.3.3.3 处理

输入数据的有效性检测：检查时间格式是否正确，字符串长度

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 数据库查询-> 结果显示

对异常情况的回应：通信失败则输出查询失败的信息

用于把系统输入转换到相应输出的任何方法：数据库查询，映射

对输出数据的有效性检测：存在性以及数据长度

#### 2.3.3.4 输出

输出目的地：前端

数量：多个，出发地-> 目的地/时间/价格/舱位等信息

度量单位：价格/RMB，个数等

时序：接收数据-> 前端整合渲染

包含精确度和容忍度的有效输出范围：整数，范围：>0

对非法值的处理：筛选并返回说明

错误消息：错误提示信息

## 2.3.4 R.INTF.CALC.004 门票预订

### 2.3.4.1 介绍

通过输入景点名称可以搜索得到相关门票信息，并且支持预订功能。输入景点不存在将作为无效输入，在软件中将显示景点不存在这一信息。不存在错误输入

### 2.3.4.2 输入

输入来源：用户

数量：2，景点位置字符串/预订按钮点击

度量单位：次数

时间要求：无

包含精度和容忍度的有效输入范围：1-20 字符

### 2.3.4.3 处理

输入数据的有效性检测：字符串长度

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 数据库查询-> 结果显示

对异常情况的回应：通信失败则输出查询失败的信息

用于把系统输入转换到相应输出的任何方法：数据库查询，映射

对输出数据的有效性检测：存在性以及数据长度

### 2.3.4.4 输出

输出目的地：前端

数量：多个，景点、价格、门票类型等

度量单位：价格/RMB

时序：接收数据-> 前端整合渲染

包含精确度和容忍度的有效输出范围：一位浮点数，范围：无

对非法值的处理：筛选并返回说明

错误消息：错误提示信息



## 2.3.5 R.INTF.CALC.005 广告推送

### 2.3.5.1 介绍

用户使用时，在网页端显示广告，不需要输入信息。管理员可以发布广告推送。

### 2.3.5.2 输入

输入来源：管理员

数量：一条记录，包含时间/位置/结束时间/价格

度量单位：价格/RMB

时间要求：应该在当前日期之后

包含精度和容忍度的有效输入范围：时间格式为 yyyy-mm-dd，价格范围：>1

### 2.3.5.3 处理

输入数据的有效性检测：字符串长度，日期格式

操作的确切次序，包括各事件的时序：管理员输入数据-> 提交-> 后端获得数据-> 录入数据库供查询

对异常情况的回应：通信失败则输出提交失败的信息

用于把系统输入转换到相应输出的任何方法：数据库查询，映射

对输出数据的有效性检测：数据库完整性检测

### 2.3.5.4 输出

输出目的地：服务器

数量：多个，与输入一致

度量单位：与输入一致

时序：生成数据库语句-> 存入数据库

包含精确度和容忍度的有效输出范围：筛选并返回说明

对非法值的处理：错误提示信息

## 2.3.6 R.INTF.CALC.006 用户评价

### 2.3.6.1 介绍

用户通过该系统可以对景点或者酒店等其它内容进行评价，输入为用户的评价，非法条件为超过字数限制，对于非法输入将显示错误信息

### 2.3.6.2 输入

输入来源：用户

数量：2，评分/评价内容

度量单位：无

时间要求：在使用预订的一定期限内

包含精度和容忍度的有效输入范围：每段评价不超过 100 个字符，评分为 0.0-5.0 的一位浮点数

### 2.3.6.3 处理

输入数据的有效性检测：检测输入信息的字符个数，评分检测

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 录入数据库

对异常情况的回应：通信失败则输出评价失败的信息

用于把系统输入转换到相应输出的任何方法：直接输出

对输出数据的有效性检测：数据库完整性检测

### 2.3.6.4 输出

输出目的地：服务器

数量：2

度量单位：无

时序：生成数据库语句-> 存入数据库

包含精确度和容忍度的有效输出范围：输出小于 100 个字符，评分为

对非法值的处理：反馈错误提示信息，输出长度超过限制的信息等

## 2.3.7 R.INTF.CALC.007 优惠券

### 2.3.7.1 介绍

用户使用时，在网页端显示优惠券，用户可以领取。管理员可以在系统中发布优惠券。

### 2.3.7.2 输入

输入来源：用户/管理员

数量：1

度量单位：张

时间要求：用户在优惠券有效日期内提交领取

包含精度和容忍度的有效输入范围：提交记录的输入日期按照 yyyy-mm-dd 格式，折扣为 1-99 的整型数。

### 2.3.7.3 处理

输入数据的有效性检测：检测时间输入合法性，折扣输入合法。

操作的确切次序，包括各事件的时序：用户/管理员输入数据-> 提交-> 后端获得数据-> 录入数据库

对异常情况的回应：通信失败则输出领取/发布优惠券失败的信息

用于把系统输入转换到相应输出的任何方法：直接输出

对输出数据的有效性检测：数据库完整性检测，用户是否已经领取过该优惠券

### 2.3.7.4 输出

输出目的地：服务器

数量：1

度量单位：张

时序：生成数据库语句-> 存入/更新数据库

包含精确度和容忍度的有效输出范围：无

对非法值的处理：反馈错误提示信息

## 2.3.8 R.INTF.CALC.008 活动推送

### 2.3.8.1 介绍

在网页端显示活动，不需要输入信息。管理员可以在系统中发布/编辑新活动。

### 2.3.8.2 输入

输入来源：管理员

数量：1

度量单位：项

时间要求：在活动截止日期前发布

包含精度和容忍度的有效输入范围：时间格式，按照 yyyy-mm-dd hh:mm:ss，  
价格范围：>0

### 2.3.8.3 处理

输入数据的有效性检测：检测时间输入合法性，价格输入合法。

操作的确切次序，包括各事件的时序：管理员输入数据-> 提交-> 后端获得数据-> 录入数据库

对异常情况的回应：通信失败则输出发布/更新新活动失败的信息

用于把系统输入转换到相应输出的任何方法：直接输出

对输出数据的有效性检测：数据库完整性检测

### 2.3.8.4 输出

输出目的地：服务器

数量：1

度量单位：项

时序：生成数据库语句-> 存入/更新数据库

包含精确度和容忍度的有效输出范围：无

对非法值的处理：反馈错误提示信息

## 2.3.9 R.INTF.CALC.009 客服服务

### 2.3.9.1 介绍

通过系统能够与客服进行即时通讯，输入同样用字数限制，对于非法输入将显示错误信息

### 2.3.9.2 输入

输入来源：用户/客服

数量：不限

度量单位：消息条数

时间要求：延迟小于 0.1s

包含精度和容忍度的有效输入范围：每段消息不超过 100 个字符

### 2.3.9.3 处理

输入数据的有效性检测：输入信息的字符个数

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 客服回复信息-> 信息显示 (反之亦然)

对异常情况的回应：通信失败则输出信息未成功发送

用于把系统输入转换到相应输出的任何方法：无

对输出数据的有效性检测：检测是否含有非法字符

### 2.3.9.4 输出

输出目的地：前端

数量：不限

度量单位：消息条数

时序：延迟小于 0.1s

包含精确度和容忍度的有效输出范围：应该小于显示框的容纳字符数

对非法值的处理：反馈错误提示信息

## 2.3.10 R.INTF.CALC.010 用户登录

### 2.3.10.1 介绍

通过输入用户名与密码登录系统，错误输入为用户不存在或者密码错误，针对相应错误系统会分别提示错误信息

### 2.3.10.2 输入

输入来源：用户

数量：2，用户名和密码

度量单位：位

时间要求：无

包含精度和容忍度的有效输入范围：无

### 2.3.10.3 处理

输入数据的有效性检测：密码是否正确

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 数据库查询并且验证-> 若成功则显示登陆

对异常情况的回应：通信失败则输出信息未成功提交，查询不匹配则输出相关信息

用于把系统输入转换到相应输出的任何方法：数据库查询

对输出数据的有效性检测：无

### 2.3.10.4 输出

前端返回登录后的首页

## 2.3.11 R.INTF.CALC.011 注册用户/修改个人信息

### 2.3.11.1 介绍

通过输入用户名与密码进行注册，错误输入为用户名已经注册，系统会提示相应的错误信息。用户在登录状态下可以提交信息修改。

### 2.3.11.2 输入

输入来源：用户

数量：多个，用户名和密码及其他信息

度量单位：位

时间要求：无

包含精度和容忍度的有效输入范围：用户名用英文字符表示，密码仅包含数字和字母

### 2.3.11.3 处理

输入数据的有效性检测：用户名是否重复

操作的确切次序，包括各事件的时序：用户输入数据-> 提交-> 后端获得数据-> 数据库查询是否重名-> 若成功则注册用户即插入信息到数据库

对异常情况的回应：通信失败则输出信息未成功提交，查询到重名则输出相关信息

用于把系统输入转换到相应输出的任何方法：数据库查询

对输出数据的有效性检测：无

### 2.3.11.4 输出

输出目的地：服务器

数量：1

度量单位：记录

时序：生成数据库语句-> 存入/更新数据库

包含精确度和容忍度的有效输出范围：无

对非法值的处理：反馈错误信息

## 2.4 条件与限制

### 2.4.1 开发平台与工具

本项目选用 Windows10 系统作为主要的开发平台，同时选择 win64 下的 Pycharm 作为开发 IDE，基于 python 的 flask 框架开发，用阿里云的 Ubuntu16.04lts 作为服务器后台。

### 2.4.2 软件开发生命周期模型

采用瀑布模型作为软件生命周期模型，因为瀑布模型适用于需求比较固定的情形，并且实行起来较为简单。

### 2.4.3 法律

本管理系统所提供的所有航班、门票信息均来自于互联网上的正规网站发布，保证真实合法，景点门票和旅行社活动的相关信息来源于国内知名旅游网站，在管理系统真正应用时会和相关的部门人员取得合作关系，保证门票/航班等代理预订有效且在法律上允许。

### 2.4.4 技术

本系统主要是基于后端的 flask 框架进行实现，flask 是一个轻量级的 web 后端框架，结合 mysql5.7 实现后台数据库管理，在流量不是巨大的情况下保证可行，但由于后台开发技术知识有限和开发时间短，所以难免有 bug 疏漏和性能上的劣势。同时由于团队中缺少精通前端开发的人员，所以对于 web 前端采用的是 Bootstrap 框架，对于前端并没有做深入的美化，因此本项目发布在 github 上开源以供进一步完善的参考。

### 2.4.5 经费

开发过程无经费，采用的服务器主机为阿里云的学生服务器，所以带宽和存储等配置相对较差。



## 第 3 章 总体设计

### 3.1 软件描述

系统包括前台和后台两个部分。

#### 3.1.1 前台主要功能

发送请求并接收服务器的响应，对返回结果进行渲染展示。

对于用户而言，为用户在 web 端提供一套完整的旅行服务，从用户出行的航班、火车票预订到酒店预订，景点推荐和门票预订，用户在登录帐号后可以和服务器建立通信关系，用户通过前端的展示进行选择或者输入提交，产生发送对应的请求与动作，接收服务器端的响应。

对于管理员而言，在登录到前端的管理系统后，可以看活动、门票、航班和用户等所有的信息，web 前端可以方便管理员高效管理，管理员可以通过每一种类（如火车票/航班等）的新建选项发布新的信息，或者通过编辑修改信息。

#### 3.1.2 后台主要功能

接收用户/管理员在 web 端发送的动作和请求，处理并响应。

服务器在接收到用户请求时，会在服务器的数据库中进行查询操作，并且进行返回查询结果或者反馈提示信息，对于用户提交的数据（比如预订、编辑个人信息），也会存入到数据库中。

服务器在接收到管理员的提交时，会在数据库中插入数据记录进行更新。

## 3.2 处理流程

### 3.2.1 总体流程

说明：该流程图为旅行管理系统的宏观展示，用户和管理员为该系统的两个活动角色，基于旅行服务管理系统，用户从系统中获取旅行相关的各种信息，而管理员负责维护该系统，并且管理用户，为用户提供全方位的优质服务。

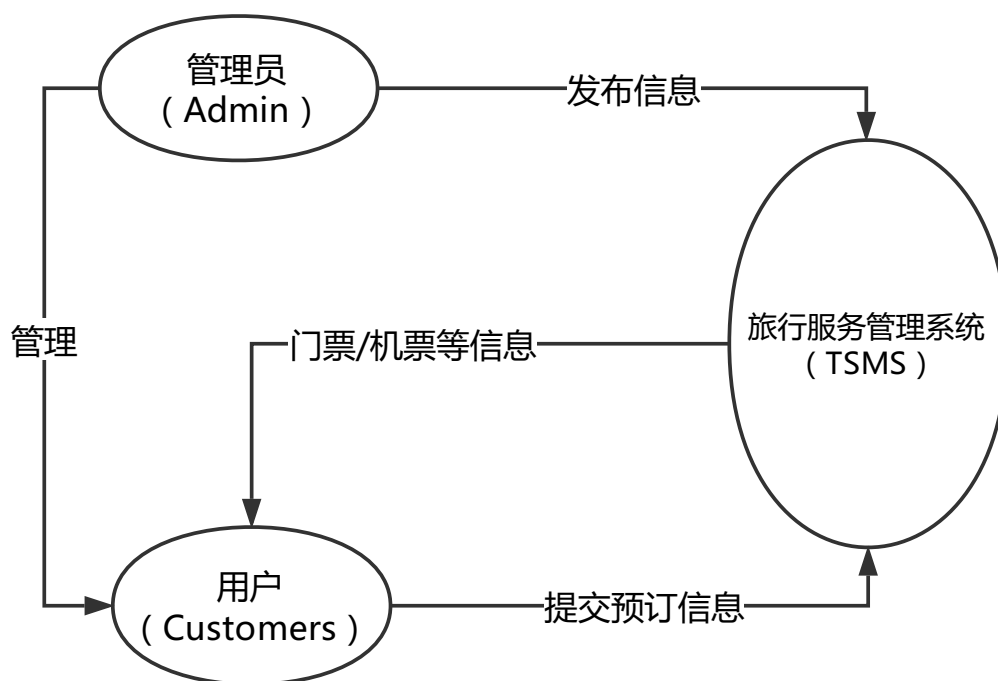


图 3.1 系统基本流程

### 3.2.2 系统基本流程

说明：该流程图为旅行管理系统的整个工作流程，其中蓝色字体所示是以用户需求为例的请求——响应流程，橙色字体所示是以管理员发布信息为例的发布——更新流程。（实际中用户也会提交更新请求，管理员也会有查询请求）

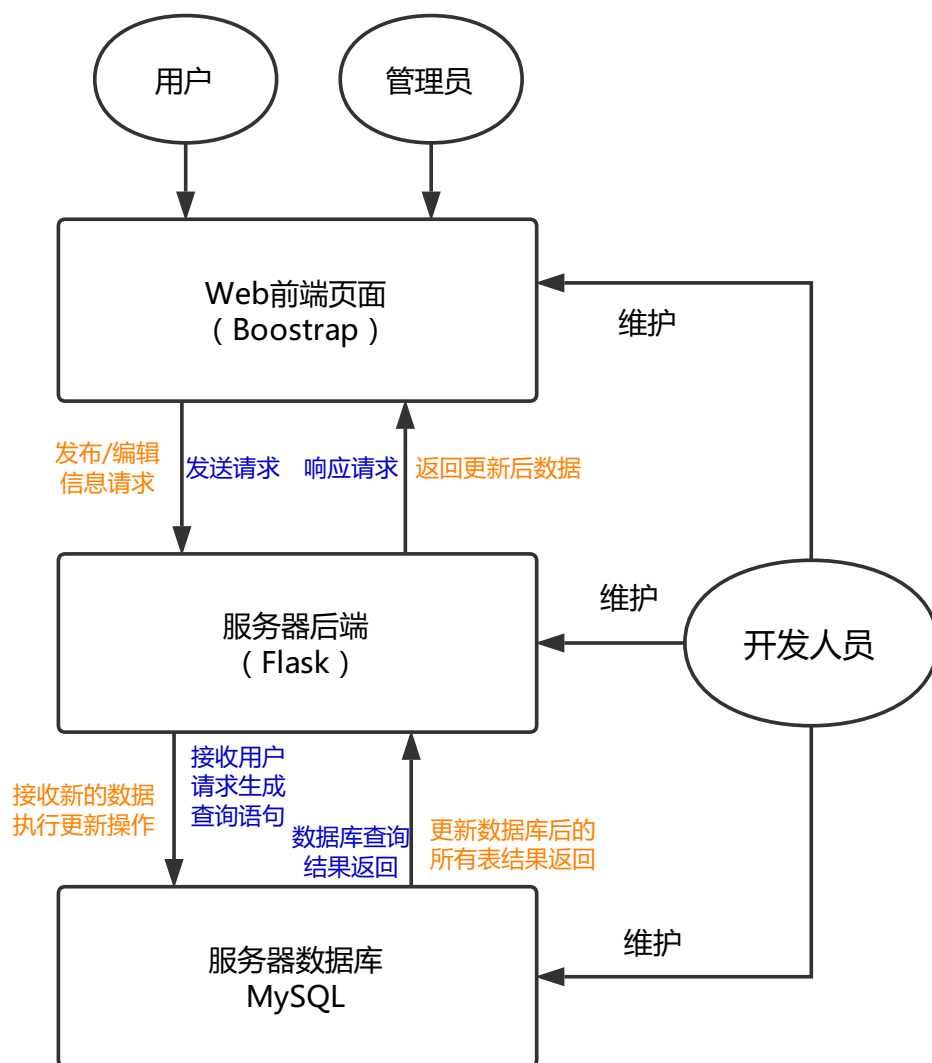


图 3.2 系统基本流程

### 3.2.3 客户端基本流程

说明：用户或者管理员在浏览器中输入用户名和密码进行验证登录（如果没有用户名需要先注册），登录后的 web 前端界面首页显示为广告活动的推送，前端为管理员（登录 id 验证为 1）提供了广告活动、航班、火车票、酒店、景点门票以及优惠券的发布和编辑选项，为用户提供了航班、火车票、酒店和景点门票的预订选项，同时可以领取优惠券享受相应的折扣，用户也可以与客服在线即时沟通，用户的预订行为会作为输入提供给推荐算法为用户进行个性化的推荐。

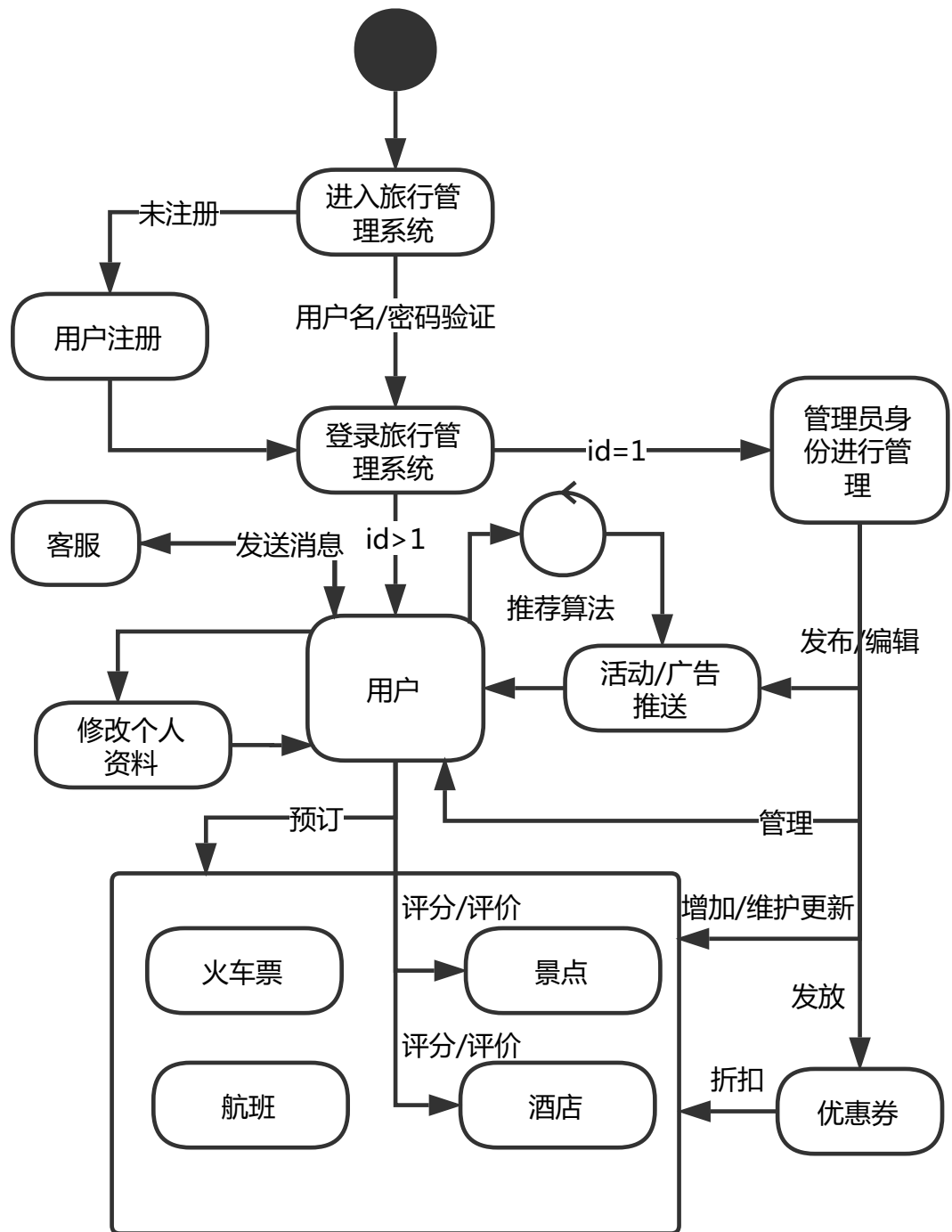


图 3.3 客户端基本流程

### 3.2.4 服务器端基本流程

说明：服务器端的 nginx 通过 80 端口与 web 客户端进行通信，在收到航班、火车票、酒店和景点的查询请求时反向代理把请求转发给 WSGI 服务器 gunicorn，后者接收到请求后直接在 flask 后端的逻辑控制模块生成 sql 查询语句连接到数据库，完成查询操作后按照原路返回到 80 端口。更新/提交请求也是类似的处理过程，在下图都有说明。优惠券和活动无需查询操作，直接返回到客户端。用户消息转发到 WSGI 服务器后把消息发送给客服，并接受客服消息转发给客户。

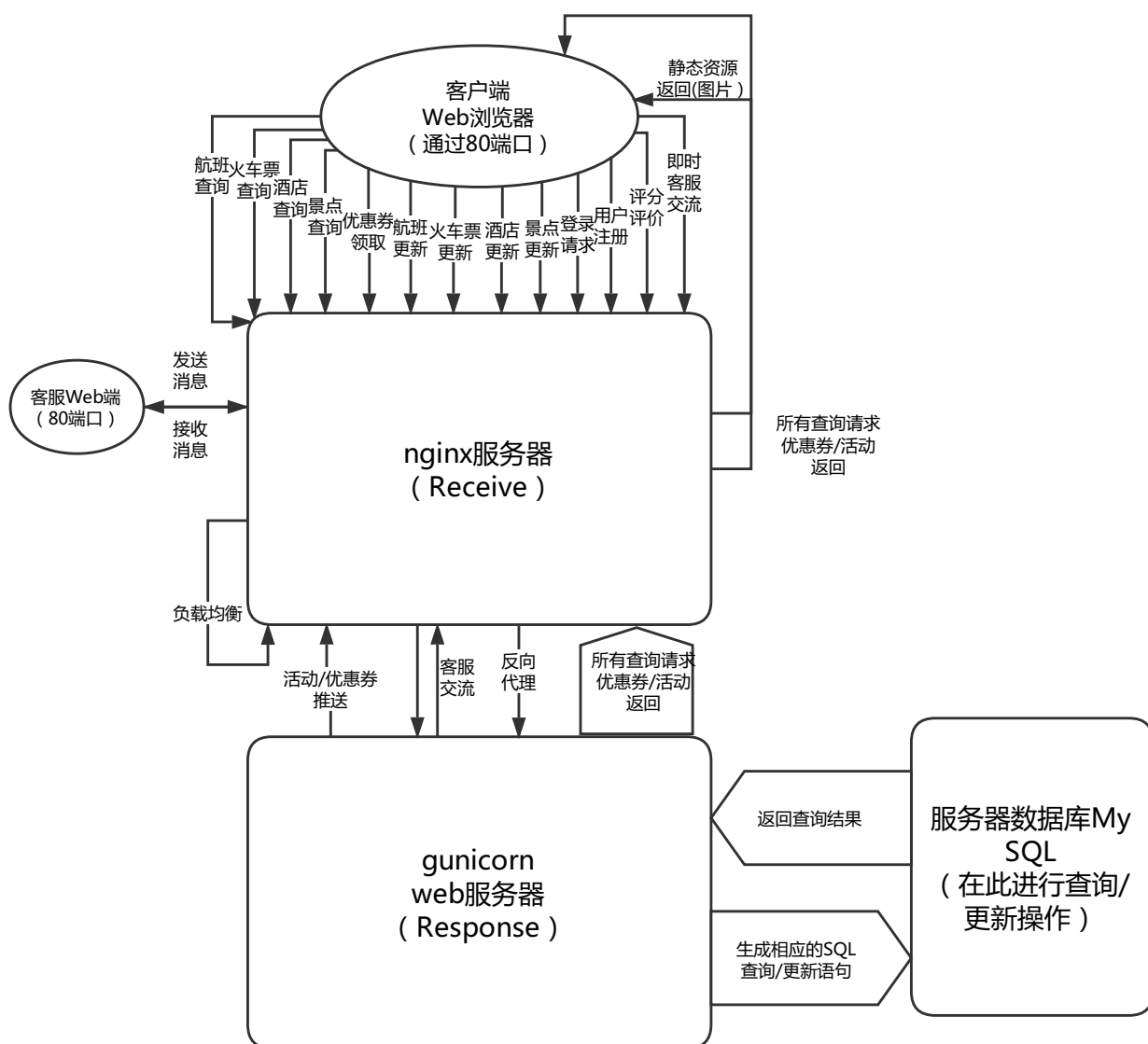


图 3.4 服务器端基本流程

### 3.2.5 功能 1——注册和登录验证具体流程

用户如果没有登录系统，在 web 前端的上方菜单栏会有注册和登录两个选项。注册时用户在注册界面填写注册信息，提交后生成表单作为 Post 请求发送给服务器，服务器接收到注册提交请求后，首先根据输入用户名在后端数据库中查询，如果查询返回不为空，说明该用户名已被注册，返回给用户提示信息“用户名已经存在”，之后会检测输入的两次密码是否一致，确实无误后将用户信息插入到数据库完成注册。用户登录时提交 post 请求，服务器端查询数据库判断该用户名是否存在以及密码是否正确，通过后用户登录，创建 session 对话。

### 3.2.6 功能 2——用户预订/领取具体流程

已经登录的用户（id>1）可以从 web 前端看到所有的航班/火车票/酒店/景点信息，同时种类的每一个项目都有预订选项，用户点击预订后提交请求，服务器端接收数据并更新数据库，然后返回给用户当前预订的数量，并且增加了取消预订的选项。优惠券的领取流程与上述类似，当某个优惠券用户未领取时可以点击领取按钮提交请求，服务器将优惠券信息更新到数据库。

### 3.2.7 功能 3——用户提交评论具体流程

在使用优惠券之后可以发表对酒店/景点的评分，用户可以提交一个 1-5 分的浮点数评分和小于 100 个字符的评论，生成表单并提交 post 请求，服务器端接收数据并且将评论信息插入到数据库。

### 3.2.8 功能 4——管理员发布信息具体流程

在管理员登录后（id=1）可以从 web 前端对所有类别的预订项目进行管理，在每一个类别（航班/机票/火车票/门票）下都可以新建或者编辑条目，提交后服务器端接收 post 请求并且将相应的记录更新到数据库以供用户查询。同时管理员亦可以在网页的首页发布新的活动，以及发放新的优惠券，提交到服务器的流程类似。

### 3.2.9 功能 5——客户服务具体流程

在用户登录之后在个人的菜单栏有客服这一选项，用户可以在线输入信息并且发送，服务器接收消息数据并且转发给相关客服人员，客服回复之后类似的通过服务器发送回用户的前端，从而实现即时通讯，这一过程无需经过数据库。

### 3.2.10 功能 6——个性化推荐系统具体流程

对于已经登录的用户，在服务器端推送活动时，会查询到用户相关的历史预订信息以及评论，从而可以判断用户的旅行喜好和个性化特点，同时也会进行用户间的相似度匹配，如果与某个用户相似度很高，那么另一个用户的相关记录可以作为依据将活动推送给该用户，整个流程为将用户个人数据集作为输入交给推荐算法进行处理，并结合后台数据库，从而生成个性化的推荐结果。

## 3.3 功能结构设计

### 3.3.1 整体结构

旅行服务管理系统的整体结构为 Web 前端——服务器——Mysql 数据库架构，这三个子系统可以相邻的进行通信，三个层次的开发架构便于管理和维护，每个子系统的结构会分别详细说明。

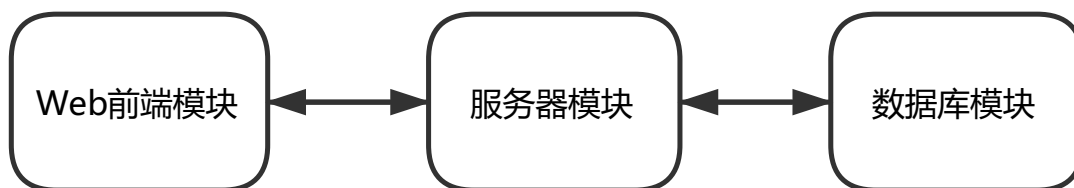


图 3.5 整体结构

### 3.3.2 用户端结构

在 web 前端的设计中，需要展现给用户的是一个美观且结构化的界面，所以在用户端的结构设计中，对每一个种类的模块进行具体划分，每一个模块都在前端网页的顶部菜单栏中找到对应的入口选项，并且为下图所示的模块都独立开发操作和展示界面，便于用户进行选择 and 提交。

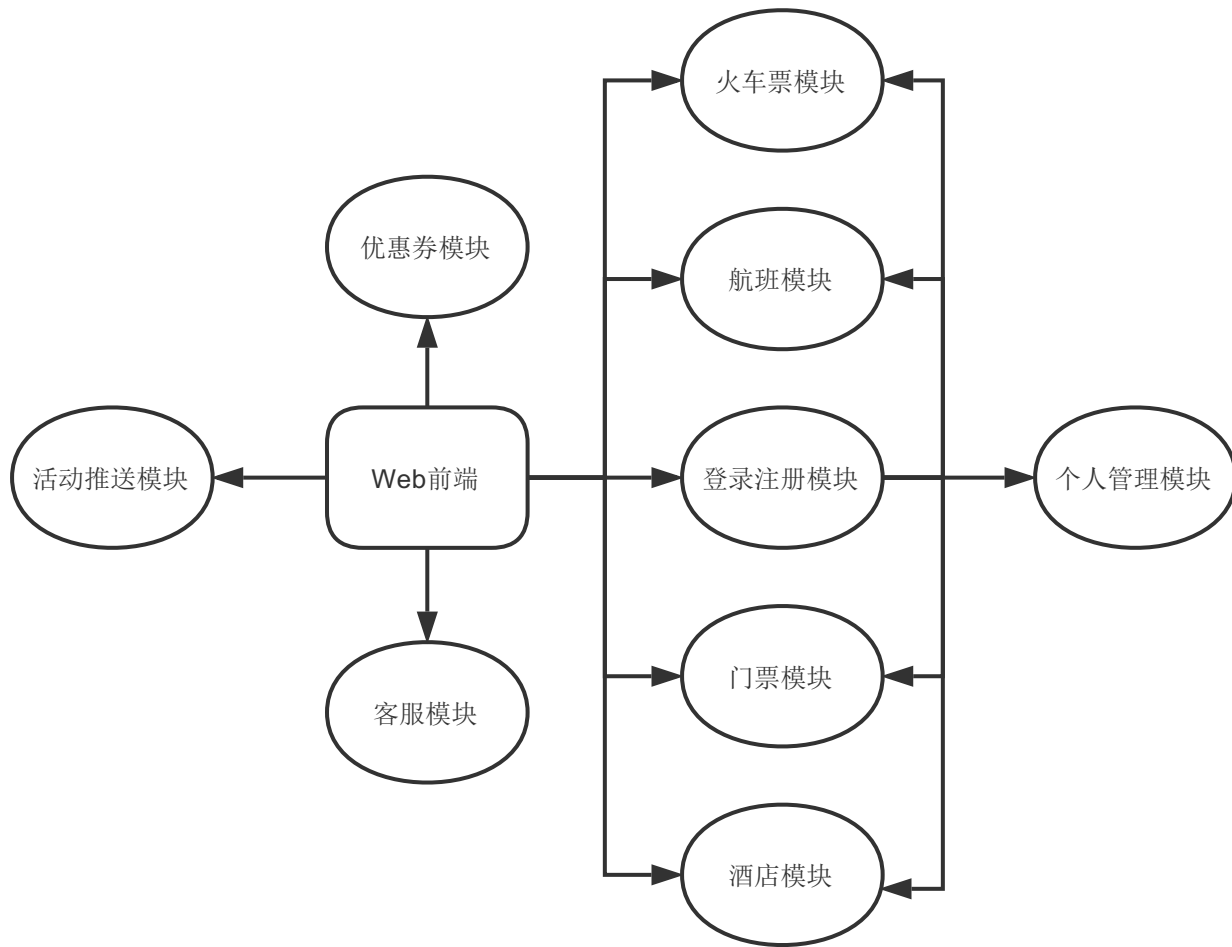


图 3.6 用户端结构



### 3.3.3 服务器端结构

服务器作为整个管理系统的核心，其分别设有前端交互的接口模块和与后端数据库进行通信的模块，同时整个系统的逻辑控制都是在服务器中进行，并且针对用户特定需求增加了个性化的推荐模块，通过有效的推荐算法为用户提供更加优质的推荐。

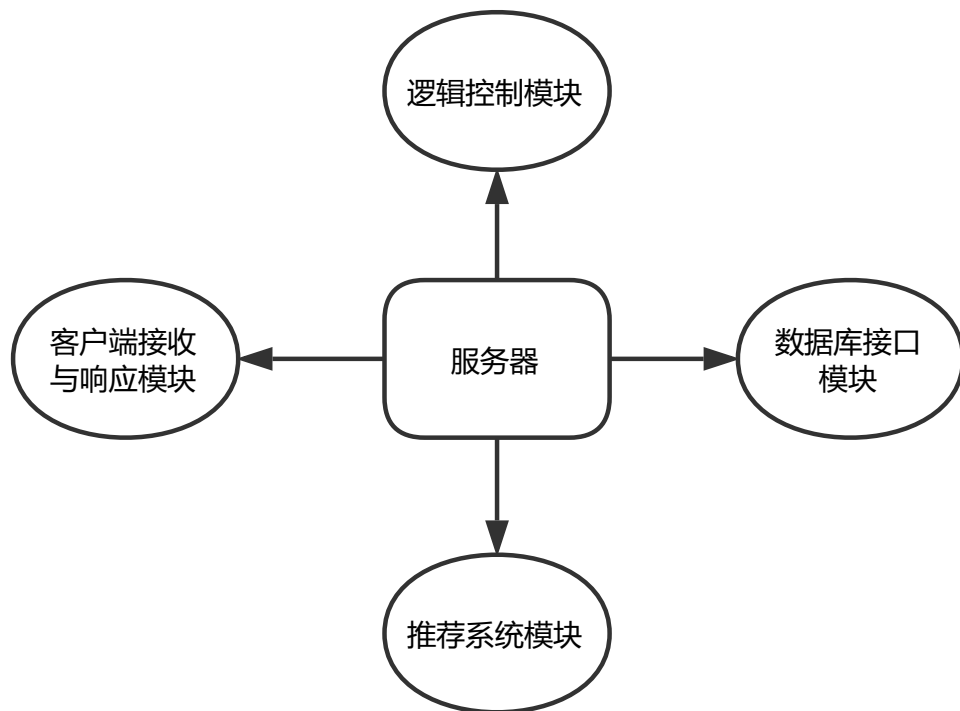


图 3.7 服务器端结构

### 3.3.4 后台数据库维护模块结构

数据库模块需要接收来自于服务器的查询和更新请求，在数据库的设计中针对用户需求设有如下图所示的多个表结构，层次清晰便于进行数据库的维护管理。

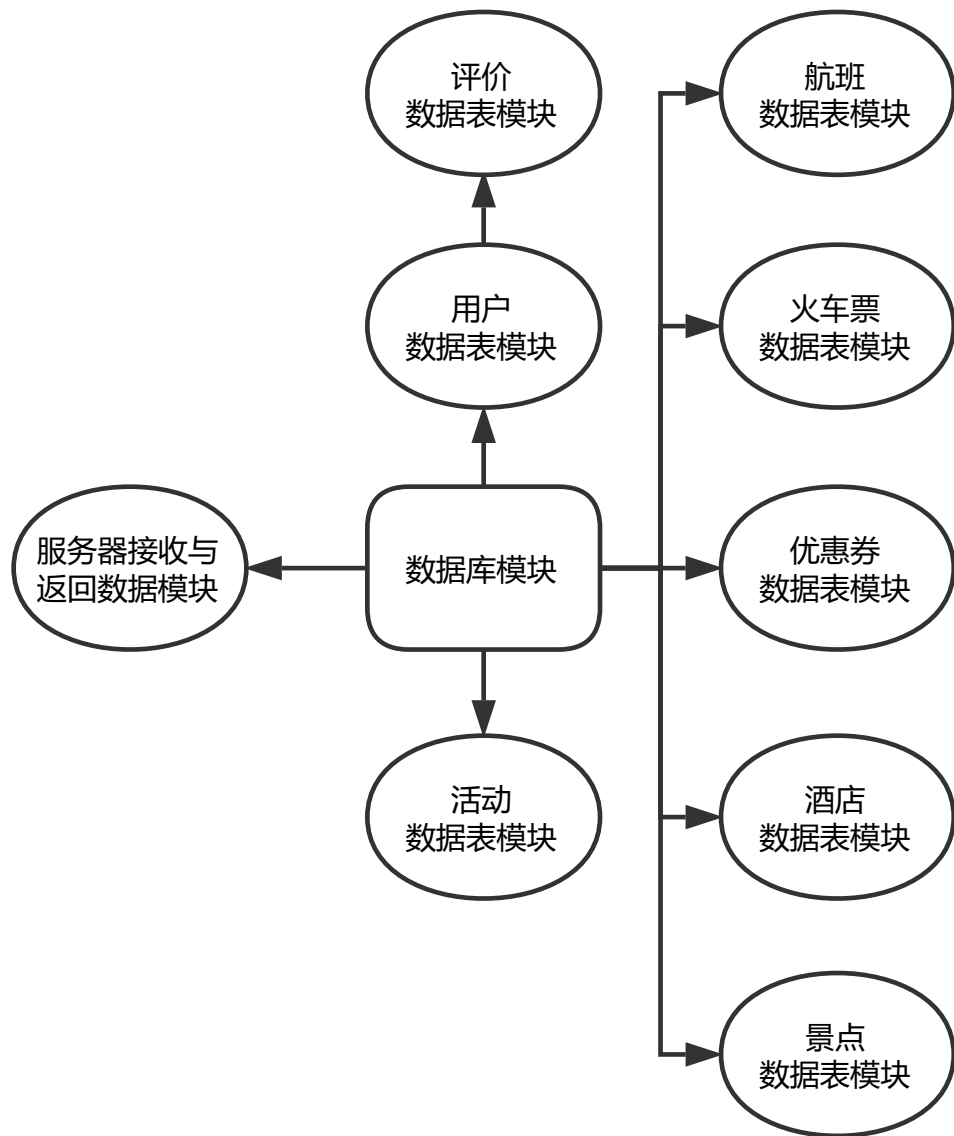


图 3.8 后台数据库维护模块结构

### 3.4 功能需求与程序代码的关系

[此处指的是不同的需求分配到哪些模块去实现。可按不同的端拆分此表]

表 3.1 功能需求与程序代码的关系表

•	酒店模块	火车票模块	航班模块	景点模块	登录注册模块	优惠券模块
酒店预订	Y	•	•	•	Y	Y
火车票预订	•	Y	•	•	Y	Y
机票预订	•	•	Y	•	Y	Y
门票预订	•	•	•	Y	Y	Y
广告/活动推送	•	•	•	•	Y	•
优惠券	•	•	•	•	Y	Y
用户登录	•	•	•	•	Y	•
注册/修改信息	•	•	•	•	Y	•

•	活动推送模块	客服模块	推荐系统	数据库模块	NULL	NULL
酒店预订	•	•	Y	Y	•	•
火车票预订	•	•	Y	Y	•	•
机票预订	•	•	Y	Y	•	•
门票预订	•	•	Y	Y	•	•
广告/活动推送	Y	•	Y	Y	•	•
用户评价	•	Y	•	Y	•	•
优惠券	•	•	•	Y	•	•
客户服务	•	Y	•	•	•	•
用户登录	•	•	•	Y	•	•
注册/修改信息	•	•	•	Y	•	•

注：各项功能需求的实现与各个程序模块的分配关系

## 第 4 章 接口设计

### 4.1 外部接口

在本项目中管理系统用到的外部接口有支付宝接口、携程（或者其他旅游网站，以携程为例）网站的接口和用户接口，下面做详述。其中用户创建预订后在线支付需要用到支付宝接口，大多数的酒店等信息等来自于携程网站的外部接口，用户在使用该系统时提供了方便用户操作的用户接口。

#### 4.1.1 支付宝接口

下单并支付交易接口：`alipay.trade.page.pay`。请求参数有商户订单号、产品码和订单总金额等参数，调用该 api 可以用于完成 PC 场景下单并支付。

统一收单交易退款接口：`alipay.trade.refund`。当交易发生之后一段时间内，由于买家或者卖家的原因需要退款时，卖家可以通过退款接口将支付款退还给买家，支付宝将在收到退款请求并且验证成功之后，按照退款规则将支付款按原路退到买家帐号上。请求参数有订单号、支付宝交易号等参数。

#### 4.1.2 携程接口

酒店查询

接口地址：`http://api2.juheapi.com/xiecheng/hotel/search`

返回格式：`json`

请求方式：`post(application/json; utf-8)`

#### 4.1.3 用户接口

用户接口即为展现给用户的交互界面，用户通过 web 界面的提交表单或者预订选项与管理系统进行交互。

### 4.2 内部接口

内部模块/系统之间的交互的接口。

#### 4.2.1 酒店/火车票/航班/景点模块与服务器逻辑控制模块

用户登录后可以进行酒店/火车票/航班/景点门票的查询请求，服务器端的逻辑模块提供了接收查询请求的接口，在收到提查询所有数据或者是关键词查询的请求后，服务器的逻辑控制模块会针对提交请求的类型进行相应处理，检测操作合法性以及确定相关的数据库模块，生成 sql 查询语句。同时，用户也可以提交预订请求，管理员登录后可以进行酒店/火车票/航班/景点/优惠券/活动推送模块的新建和更新操作，提交请求到服务器逻辑控制模块，同样的生成 sql 语句对数据库进行操作。

#### 4.2.2 客户服务模块与服务器逻辑控制模块

用户在登录状态下可以通过网页端的客服选项发起一个对话请求，服务器逻辑控制模块为客户服务模块提供了接收用户消息请求的接口，收到用户消息后服务器可以进行调度并且将用户的消息转发给某个客服，同样的客服的消息也可以通过服务器逻辑控制模块的接口转发回建立会话的客户。

#### 4.2.3 服务器逻辑控制模块与数据库维护模块

服务器在接收到客户端的请求之后，还要与数据库进行通信。服务器逻辑控制模块通过数据库维护模块提供的查询和更新接口可以高效的接收来自于服务器的相应请求，并且数据库完成操作后将数据结果返回给服务器。

#### 4.2.4 推荐系统模块与数据库维护模块

由于该旅游管理系统增加了推荐系统模块，所以必须提供一个为之输入数据集的接口，而推荐系统模块和数据库维护模块之间的数据接口就可以实现这一过程，对于特定的用户，推荐系统需要获知该用户的历史行为数据，并且要结合当前数据库中可供推送的所有数据和其他用户的数据，然后推荐系统经过算法分析和学习之后将最可能符合用户喜欢的一些信息推送给用户。

## 第 5 章 数据结构设计

### 5.1 逻辑结构设计

#### 5.1.1 用户管理系统数据结构设计

讲述本系统内需要什么数据结构。这指的是程序运行过程中维护的数据结构。只是举个例子，此处应和 3.3 一致。

#### 5.1.2 客户端数据结构

##### 1. 火车票模块

(a) 维护的数据结构：火车班次 (String)，座位类型 (String)，价格 (Integer)，座位数 (Integer)，出发时间和到达时间 (DateTime)，出发城市和目的城市 (String)，数据缓冲区，通信计时，校验串。

(b) 说明：该模块的请求发送和接收响应往往都是以表单的形式，根据用户实际在查看和预订火车票的需求维护以上所述的数据类型，并且为了维护数据结构，将其设为一个表单类 (继承来 FlaskForm)。当大量数据就进行通信传输时，需要缓冲区类型来存储图片等等数据，据结构通信计时器查看是否发生请求超时。

##### 2. 航班模块

(a) 维护的数据结构：航班号 (String)，舱位 (String)，价格 (Integer)，座位数 (Integer)，起飞时间和到达时间 (DateTime)，出发城市和目的城市 (String)，数据缓冲区，通信计时，校验串。

(b) 说明：该模块的请求发送和接收响应往往都是以表单的形式，根据用户实际在查看和预订航班的需求维护以上所述的数据类型，并且为了维护数据结构，将其设为一个表单类 (继承来 FlaskForm)。当大量数据就进行通信传输时，需要缓冲区类型来存储图片等等数据，据结构通信计时器查看是否发生请求超时。

### 3. 登录注册模块

(a) 维护的数据结构：用户名 (String)，密码 (String)，姓名 (String)，年龄 (Integer)，性别 (String)，手机号 (String)，地址 (String)，通信计时，校验串。

(b) 说明：用户在登录时需要用户名 (String) 和密码 (String)，注册时需要提交的表单包含以上所有的数据类型。

### 4. 景点模块

(a) 维护的数据结构：景点名 (String)，位置 (String)，景点特色 (String)，门票类型 (String)，价格 (Integer)，使用期限 (DateTime)，通信计时，校验串，数据缓冲区。

(b) 用户在预订门票时需要浏览一个景点的详细信息，所以景点的位置和景点的特色在前端用 String 数据类型表示维护，具体的其他信息价格为整型、使用期间为 DateTime 类型。

### 5. 酒店模块

(a) 维护的数据结构：酒店名 (String)，位置 (String)，房间类型 (String)，价格 (String)，房间数 (Integer)，使用期限 (DateTime)，通信计时，校验串，数据缓冲区。

(b) 将以上数据类型集合封装为一个表单类，具体如模块所述。

### 6. 优惠券模块

(a) 维护的数据结构：折扣商家 (String)，类型 (String)，折扣 (Integer)，优惠券数量 (String)，使用期限 (DateTime)，通信计时，校验串，数据缓冲区。

(b) 将以上数据类型集合封装为一个表单类，具体如模块所述。

### 7. 活动推送模块

(a) 维护的数据结构：活动名 (String)，内容信息 (String)，活动价格 (Integer)，结束时间 (DateTime)，通信计时，校验串，数据缓冲区。

(b) 说明：活动推送内容展现在网页端的首页，在前端需要维护表述一个活动的数据类型如上。

## 8. 客服模块

(a) 维护的数据结构：请求/接收消息 (Message)，数据缓冲区，通信计时，校验串

(b) 说明：客服模块中用户发送和接收消息的过程为一个即时通信过程，将用户发送/接收的消息封装后进行传输，并且需要动态的缓冲区存储可能到达的数据包，通信计时器和校验串为了保证通信的正常进行和数据包的完整性。

### 5.1.3 服务器端数据结构

(1) 数据库操作：

- 操作表对象；
- 操作类型
- 操作值
- 操作语句
- 发生时间
- 是否成功

(2) 表对象信息：

- 表名
- 表记录数
- 表最后一次操作
- 表可用性

(3) 服务器接收请求：

- 请求来源
- 目标 URL
- 请求类型



- 请求资源
- 请求源代码
- 发生时间
- 是否已回应

(4) 服务器回应:

- 回应目标 URL
- 回应来源
- 回应信息体
- 回应对应请求
- 回应时间
- 是否已送达

(5) 推荐模块:

- 用户名
- 预订列表
- 活动列表
- 相似度

## 5.2 物理结构设计

无

## 第 6 章 数据库设计

### 6.1 数据库环境说明

本系统的数据系统采用 MySQL 数据库系统。

### 6.2 数据库的命名规则

1. 名称的命名方式选用头字母大写（头分法）
2. 允许单词缩写，很长的单词可以采取前几个字母作为代表  
比如广告活动 Advertisements 缩写为 Ads
3. 属性名基本采用驼峰命名法的方式  
比如用户名“user name”表示为 userName
4. 表名是复数，首字符大写
5. 字段不带类型前缀

## 6.3 逻辑设计

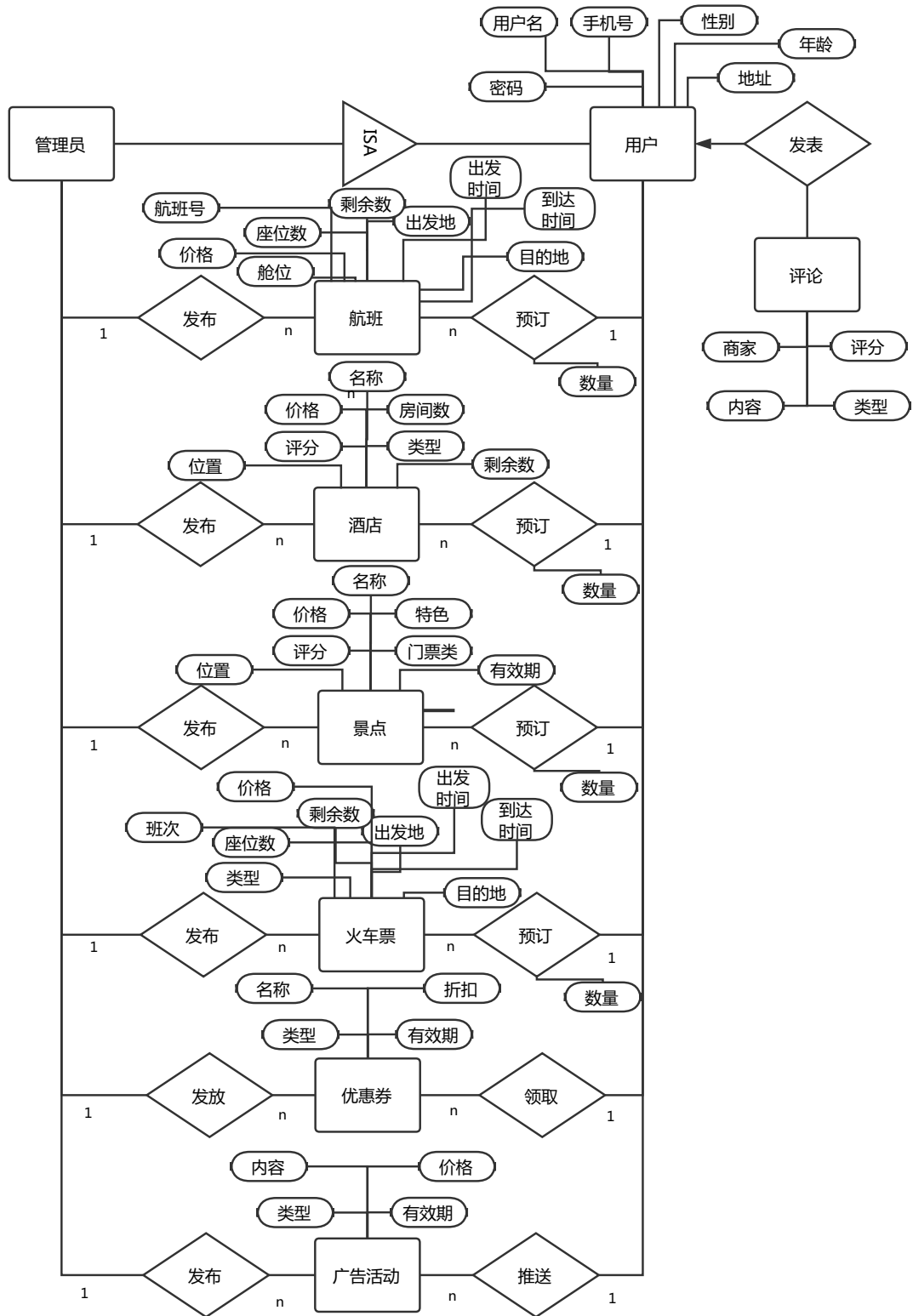


图 6.1 逻辑设计图

## 6.4 物理设计

### 6.4.1 数据库产品

瑞典 MySQL AB 公司的 MySQL 数据库，非分布式。

### 6.4.2 实体属性、类型、精度

#### 6.4.2.1 活动数据表设计

表 6.1 活动数据表 Ads 设计

字段名	类型	大小	说明	备注
id	integer	64	活动的唯一标识符	主键
adName	char	256	活动的名称	
adContent	char	1024	活动的内容	
adPrice	integer	64	活动的价格	
endTime	dateTime	256	活动的截止时间	

注：活动数据表 Ads 设计

#### 6.4.2.2 客户数据表设计

表 6.2 用户数据表 Users 设计

字段名	类型	大小	说明	备注
id	integer	64	用户的唯一标识符	主键
userName	char	128	用户的用户名	Unique 唯一
passWd	char	256	用户的登录密码	
custName	char	128	用户的姓名	
custSex	char	32	用户的性别	
custAge	integer	64	用户的年龄	大于 0
custPhone	char	128	用户的手机号	
custAddr	char	512	用户的地址	

注：用户数据表 Users 设计

## 6.4.2.3 优惠券数据表设计

表 6.3 优惠券数据表 Coupons 设计

字段名	类型	大小	说明	备注
id	integer	64	优惠券的唯一标识符	主键
couponName	char	256	优惠券的名称	
couponType	char	64	优惠券类型 (比如酒店/航班等)	
couponRate	integer	64	优惠券的折扣	范围在 1 99
endTime	dateTime	256	优惠券使用的截止时间	

注：优惠券数据表 Coupons 设计

## 6.4.2.4 用户评论数据表设计

表 6.4 用户评论数据表 Marks 设计

字段名	类型	大小	说明	备注
id	integer	64	评论的唯一标识符	主键
userName	char	128	评论者的用户名	外码，来自 Customers 表
markScore	float	64	用户评分	
markContent	char	1024	评论内容	
markName	char	256	被评论的商家名	
markType	char	128	评论类型	

注：用户评论数据表 Marks 设计

## 6.4.2.5 航班数据表设计

表 6.5 航班数据表 Flights 设计

字段名	类型	大小	说明	备注
id	Integer	64	飞机票的唯一标识符	主键
flightId	char	128	航班号	
seatType	char	64	舱位	
price	integer	64	飞机票价格	大于 0
seatNum	integer	64	座位数目	大于 0
avaiNum	integer	64	剩余座位数	$\geq 0$
fromCity	char	256	出发城市	
arivCity	char	256	到达城市	
fromTime	datetime	256	起飞时间	
arivTime	datetime	256	到达时间	

注：航班数据表 Flights 设计

## 6.4.2.6 火车票数据表设计

表 6.6 火车票数据表 Trains 设计

字段名	类型	大小	说明	备注
id	Integer	64	火车票的唯一标识符	主键
trainId	char	128	列车班次	
seatType	char	64	座位类型	
price	integer	64	火车票价格	大于 0
seatNum	integer	64	座位数目	大于 0
avaiNum	integer	64	剩余座位数	$\geq 0$
fromCity	char	256	出发城市	
arivCity	char	256	到达城市	
fromTime	datetime	256	出发时间	
arivTime	datetime	256	到达时间	

注：火车票数据表 Flights 设计

## 6.4.2.7 酒店数据表设计

表 6.7 酒店数据表 Hotels 设计

字段名	类型	大小	说明	备注
id	Integer	64	酒店预订记录的唯一标识符	主键
hotelName	char	256	酒店名称	
hotelLoca	char	512	酒店位置	
hotelScore	float	64	酒店评分	
roomType	char	64	房间类型	
price	integer	64	价格	大于 0
roomNum	integer	64	房间数目	大于 0
avaiNum	integer	64	剩余房间数	$\geq 0$

注：酒店数据表 Hotels 设计

## 6.4.2.8 景点数据表设计

表 6.8 景点数据表 Attractions 设计

字段名	类型	大小	说明	备注
id	Integer	64	景点门票的唯一标识符	主键
attrName	char	256	景点名称	
attrLoca	char	512	景点位置	
attrScore	float	64	景点评分	范围 0 5
features	char	1024	景点特色	
ticType	char	64	门票类型	
price	integer	64	价格	大于 0
endTime	datetime	256	门票有效期限	

注：景点数据表 Attractions 设计

## 6.4.2.9 用户领取优惠券数据表设计

表 6.9 用户领取优惠券数据表 MyCoupons 设计

字段名	类型	大小	说明	备注
id	integer	64	用户所领取优惠券的唯一标识符	主键
custId	integer	64	用户标识符	外码，来自 Customers 表
CouponId	integer	64	优惠券标识符	外码，来自表 Coupons 表

注：用户领取优惠券数据表 MyCoupons 设计



#### 6.4.2.10 用户预订数据表设计

表 6.10 用户预订据表 Reservations 设计

字段名	类型	大小	说明	备注
id	integer	64	用户预订单号的唯一标识符	主键
custId	integer	64	用户标识符	外码，来自 Customers 表
resvType	char	64	预订类型 (航班/酒店等)	
resvId	integer	64	预订对象的标识符	
resvNum	integer	64	预订的数量	

注：用户预订据表 Reservations 设计

### 6.5 安全性设计

备份和容灾设计。

设置一份与生产环境一致的容灾节点, 利用磁盘介质在容灾节点保留一份生产系统每天的原始数据

在应用层面上, 本地节点使用 Cluster Server 实现主机高可用性, 防止主机故障.

在数据层面山, 在本地先形成一套主机系统和业务数据的磁盘备份, 并每隔 8 小时在后台将本地备份数据复制到远程容灾节点 (周期复制), 异地节点恢复主节点数据, 以实现主备节点的数据同步。

主节点和灾备节点在硬盘上至少保持 6 个月内的系统历史数据。

### 6.6 数据库管理与维护说明

对于数据库的维护, 随时对数据库中的信息加以调试和保存备份。同样需要个工作人员进行系统的分析和用户的反馈, 对系统进行升级以及功能的完善。同时保证系统安全有序的运行。

## 第 7 章 界面设计

在本章节中将展示 web 前端界面，来源于 demo 设计，采用扁平化设计风格，pc 和手机端均可通过网页登录系统，此处选用手机端的界面，pc 端界面类似。

### 7.1 注册界面



图 7.1 注册界面

## 7.2 登录界面



图 7.2 登录界面

在登录界面，用户输入用户名和密码然后点击登录后提交到服务器后台进行验证。

## 7.3 菜单界面



图 7.3 菜单界面

此处展示的为整个 web 前端界面的顶部菜单栏，在 pc 端全部显示，在手机端点击右边的展开选项后即可看到所有的菜单选项进行选择。

## 7.4 主页界面



图 7.4 主页界面

主页界面向用户推送广告活动。

## 7.5 航班预订界面



图 7.5 航班预订界面界面

在航班预订界面，用户可以浏览到所有的航班信息，同时有一个预订按钮，用户点击后即可预订数+1，并且即时显示预订数量，同时也可以取消预订。

## 7.6 火车票预订界面



图 7.6 火车票预订界面

在火车票预订界面，用户可以浏览到所有的火车票信息，同时有一个预订按钮，用户点击后即可预订数+1，并且即时显示预订数量，同时也可以取消预订。

## 7.7 酒店预订界面

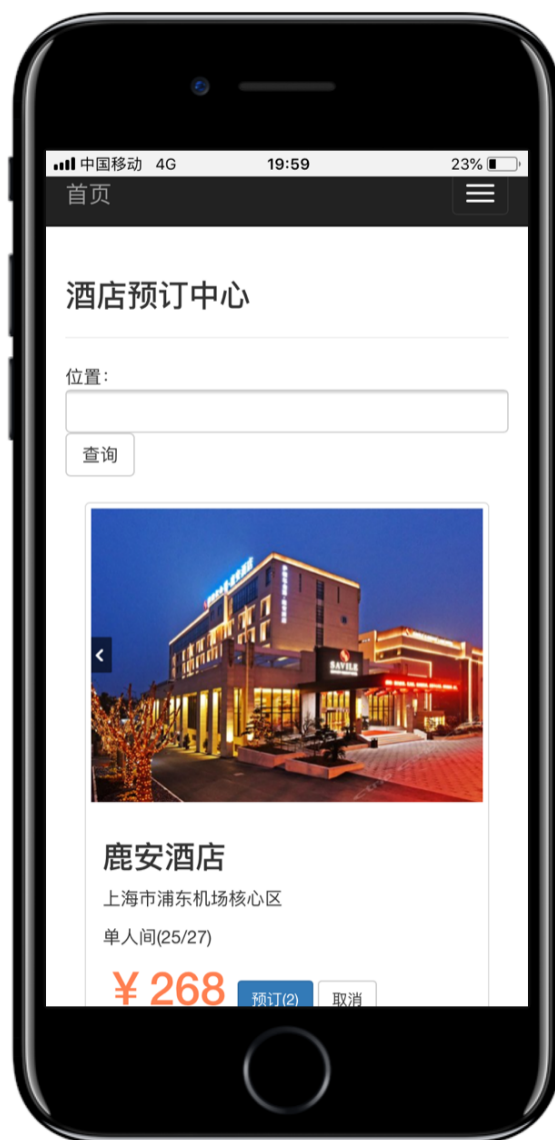


图 7.7 酒店预订界面

在酒店预订界面，用户可以浏览到所有的酒店信息，同时有一个预订按钮，用户点击后即可预订数+1，并且即时显示预订数量，同时也可以取消预订。



## 7.8 景点预订界面



图 7.8 景点预订界面

在景点门票预订界面，用户可以浏览到所有的景点信息，同时有一个预订按钮，用户点击后即可预订数+1，并且即时显示预订数量，同时也可以取消预订。

## 7.9 优惠券领取界面



图 7.9 优惠券领取界面

在优惠券领取界面，用户可以浏览到所有的优惠券信息，同时有一个领取按钮，用户点击后即可领取该优惠券，并且即时显示“已领取”，无法重复领取。

## 7.10 个人信息界面



图 7.10 个人信息界面

在此界面用户可以看到个人所有的信息，并且可以进行注销或者编辑个人信息。

## 7.11 预订信息界面



图 7.11 火车票预订界面

在预订信息界面，用户可以看到自己的全部预订信息，包括每一项的预订数量以及时间等详细说明。

## 7.12 编辑信息界面



图 7.12 编辑信息界面

当管理员登录时，会提供每一个模块类别的编辑信息，管理员可以增加新的项目或者进行编辑，此处仅仅展示其中一个编辑界面，航班/火车票等模块的编辑界面类似。

## 第 8 章 出错处理设计

### 8.1 数据库出错处理

多重备份时，应采取何种策略，先利用哪一份备份；系统是否暂停服务等。

当保存用户信息的数据库出错时，暂停整个用户系统的服务，涉及到用户信息的所有操作向客户端返回报错响。

当航班/火车票等信息数据库发生错误时，暂停整个系统的服务，对用户发送的所有查询, 预订, 更新请求返回报错响应。

当广告活动数据库发生错误时，对于一切推送给用户的信息返回报错反馈。

当预订信息数据库发生错误时，对于一切更新预订的请求返回报错响应, 此时客户端应停止运行并且在 web 端首页提示错误信息，等待后台修复。

在数据库发生错误时, 首先尝试使用日志恢复到正常状态; 如果不行, 尝试恢复本地保存的备份数据; 如果本地备份数据也已损坏, 尝试恢复容灾节点的备份数据. 如果以上操作均失败, 则服务器端自行恢复数据库已变的不可能, 向客户发出服务已失效的公告. 对于用户信息数据库发生的错误, 考虑以下解决方案: 在已经告知用户服务器端数据库已毁坏的情况下, 由客户端向服务器端发送保存在本地的相关数据, 以最大程度的恢复数据。

### 8.2 模块失效处理

是否整个系统暂停服务，还是维持最小服务状态、如何尽快恢复服务还是删库跑路等。

#### 8.2.1 酒店/航班/火车票/景点模块

当提供相关展示信息的酒店等模块失效时，暂停用户的查询请求和预订请求，其他模块正常工作，当确定模块失效是广泛存在时, 系统检测后交给维护人员进行尽快修复，并且尽量由客服人员在客服模块对所有用户说明相关的情况。

### 8.2.2 登录/注册模块

当登录/注册模块失效时, 暂停除首页广告推送之外的所有系统服务, 此时用户无法登录系统或者注册新用户, 在 web 端的首页发出维护公告, 只能等待开发维护人员修复系统之后才能正常运转。

### 8.2.3 优惠券模块

优惠券模块失效时, 暂停用户的预订请求, 因为此时优惠券失效可能导致用户预订时无法正常使用优惠券, 其他模块正常工作。首先由本地确定失效原因, 如果确定失效原因由代码内部的错误引起, 则不断尝试向服务器发送错误信息。当确定模块失效是广泛存在时。由开发者对 bug 进行修复。

### 8.2.4 活动推送模块

活动推送模块失效时, 暂停 web 端首页的推送服务, 同时暂停服务器端的推荐系统模块, 由开发人员检测是 web 端的代码出现问题还是后台的推荐系统算法存在 bug 导致, 查明问题后尽快修复并且推送给用户。

### 8.2.5 客服模块

当客服模块失效时, 暂停客服模块的工作, 其他模块正常工作, 认定非网络通信的原因之后, 由后台维护人员尽快检测 bug 并进行修复。

### 8.2.6 后台数据库维护模块

当后台数据库维护模块失效时, 实行上一节介绍的数据库出错处理操作, 恢复数据库数据. 如果后台数据库维护模块的失效不是由数据错误引起的, 则尽快对代码 bug 进行修复。

后台数据库维护模块失效时, 一切发往服务器端的请求都将收到一个保存响应, 同时暂停服务器端上其他模块的工作。

### 8.2.7 后台逻辑控制模块

后台逻辑控制模块失效时, 暂停所有接收 web 端请求的工作, 尽快对代码 bug 进行修复。

### 8.2.8 推荐系统模块

当推荐系统模块失效时，确定是否来自于数据库的问题，并且检测推荐算法，尽快对代码 `bug` 进行修复。



## 第 9 章 安全保密设计

可能的内容包括保密性、是否采取加密传输、密钥如何分发和管理等。

### 9.0.1 保密性

对设计用户信息的传输进行保密, 包括:

- 用户登录
- 用户注销
- 预订信息
- 用户预订单号
- 密码修改

### 9.0.2 加密传输

对于除广告活动推送外所有本地与服务器端的数据交换, 采用 https 进行加密传输。

## 第 10 章 维护设计

可能的内容包括数据库的日常备份、压缩、维护等。

### 10.0.1 备份

对以下数据库进行完全备份:

- 用户信息数据表 Users
- 用户评论数据表 Marks
- 航班数据表 Flights
- 火车票数据表 Trains
- 酒店数据表 Hotels
- 景点数据表 Attractions
- 用户预订数据表 Reservations
- 用户优惠券数据表 MyCoupons

同时,对收到的以下请求和结果进行备份:

- 预订操作
- 优惠券领取操作
- 用户信息更改操作

### 10.0.2 维护

每天固定的时间段内进行以下项目的检验:

- 服务器端网络的可达性
- 用户登录连接的速度
- 数据库内容的正确性 (检验和)