悠行

目录

[1 引言 2](#_Toc13000634)

[1.1编写的目的 2](#_Toc13000635)

[1.2 项目背景 2](#_Toc13000636)

[1.3定义 2](#_Toc13000637)

[2 任务概述 3](#_Toc13000638)

[2.1 目标 3](#_Toc13000639)

[2.2 运行环境 3](#_Toc13000640)

[2.3限制描述 3](#_Toc13000641)

[3总体设计 4](#_Toc13000642)

[3.1基本设计概述和处理流程 4](#_Toc13000643)

[3.2系统结构设计和模块外部设计 4](#_Toc13000644)

[4接口设计 5](#_Toc13000645)

[4.1外部接口设计 5](#_Toc13000646)

[4.2内部接口设计 5](#_Toc13000647)

[5数据结构设计 5](#_Toc13000648)

[5.1逻辑结构设计 5](#_Toc13000649)

[5.2物理结构设计 6](#_Toc13000650)

[5.2.1用户信息表（user） 6](#_Toc13000651)

[5.2.2景点信息表（attraion） 7](#_Toc13000652)

[5.2.3评价信息表（comment） 8](#_Toc13000653)

[5.2.4图片信息表（image） 8](#_Toc13000654)

[5.2.5标签信息表（label） 8](#_Toc13000655)

[5.2.6 聊天信息表（chat） 9](#_Toc13000656)

[5.2.7论坛信息表（forum） 9](#_Toc13000657)

[5.2.8论坛回复表（forum answer） 10](#_Toc13000658)

[5.2.9攻略信息表（strategy） 10](#_Toc13000659)

[5.2.10评价信息表（assess） 11](#_Toc13000660)

[5.2.11旅游团信息表（society） 11](#_Toc13000661)

[5.2.12组团信息表（group） 12](#_Toc13000662)

[5.3数据结构与程序关系设计 12](#_Toc13000663)

[6运行设计 12](#_Toc13000664)

[6.1运行模块的组合 12](#_Toc13000665)

[6.2运行控制 13](#_Toc13000666)

[6.3运行时间 13](#_Toc13000667)

[7出错设计 13](#_Toc13000668)

[7.1出错输出信息 13](#_Toc13000669)

[7.2出错补救措施 13](#_Toc13000670)

[8维护设计 14](#_Toc13000671)

# 1 引言

## 1.1编写的目的

由前面的需求分析，得出了系统的基本需求和基本用例图。要实现整个系统，需要对用户的需求进行设计。概要设计主要是利用比较抽象的语言对整个需求进行概括，确定对系统的物理配置，确定整个系统的处理流程和系统的数据结构，接口设计，实现对系统的初步设计。预期读者为客户代表及程序设计人员。

## 1.2 项目背景

随着景区不断通过大数据、互联网打造智慧景区，使线上线下信息化体验不断融合，度假品质不断提高。在线旅游平台不断对景区进行融合打造、使在线度假更加便利。预计2019在线度假市场仍将保持高速增长。而机票交易份额则会不断下滑。住宿交易相对稳定。

当前移动旅游的主要形式为线上线下一体化解决方案。包括了以携程为代表的在线旅游；以马蜂窝为代表的用户生产内容社区；以腾讯的“一部手机游云南”为代表的头部企业，以及众多类似于定制游、个性游产品的创新型服务。过去一年中，携程、美团点评通过资本层面并购等方式，分别确立了各自在旅游预订与生活服务市场的优势地位；用户生产内容攻略社区尽管在2018年出现了诸如评论造假等负面事件，但仍通过内容变现商业模式的回归，重新成为在线旅游资本寒冬中为数不多的亮色；而以腾讯“一机游”模式为代表的目的地一体化平台，虽具争议但也对以目的地整体进行信息化建设和运营提供了新的模式。线上旅游项目具有很大的市场前景。

## 1.3定义

MySQL 8.0：数据库管理软件

DBMS：数据库管理系统

Windows 10：运行环境

Java：软件开发语言

Tomcat：服务器

# 2 任务概述

## 2.1 目标

悠行旅游在线系统的各个功能是围绕着为使用者推荐最合乎他们自己心意的旅游地点。该系统主要是根据去过该景点，并且在我们的网站上留下的评论以及网络上一些别的平台上的对该景点的评论。采用一定的方式排序出景点中的最优选择，或者最优选择组合，推荐给用户。后台管理员则是负责对这些景点的信息修改，对用户的管理。

## 2.2 运行环境

服务器端：

软件环境是操作系统：Windows 10 数据库：MySQL 8.0

硬件环境是内存256M以上；

客户端：

软件环境是IE6.0及以上

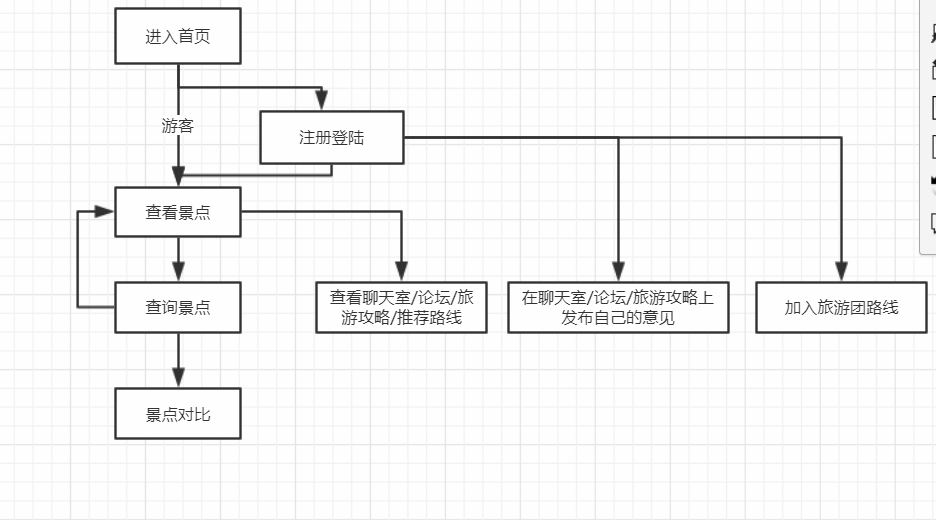
硬件环境是能支持软件的最低配置。

## 2.3限制描述

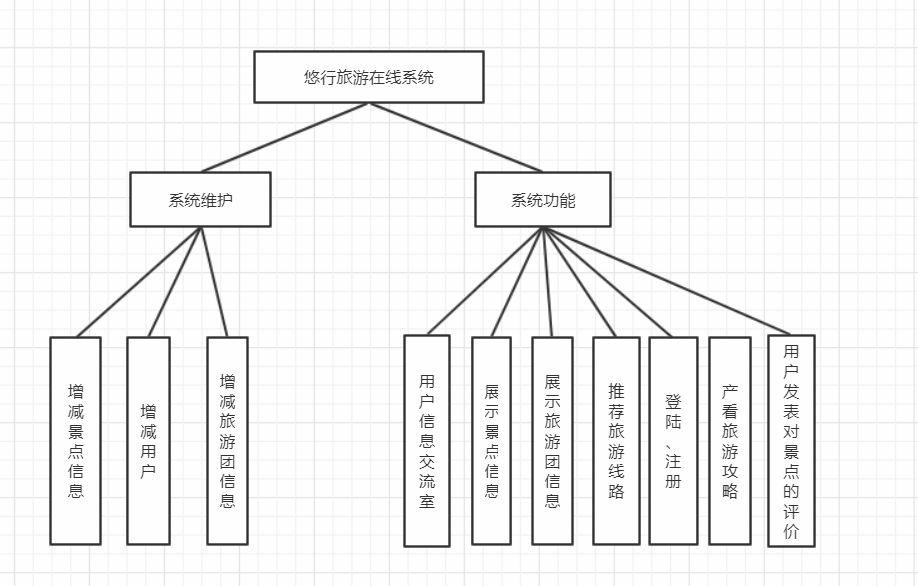
由于网站的信息中的对于景点的展示信息较多，以及数据读取速率上的限制，需要版本相对较高的浏览器。

# 3总体设计

## 3.1基本设计概述和处理流程



## 3.2系统结构设计和模块外部设计



# 4接口设计

## 4.1外部接口设计

无硬件接口；

软件接口是通过JDBC对MySQL数据库进行连接。

## 4.2内部接口设计

通过JSP+Servlet+JavaBean，结合xml语言实现系统各模块功能的结合及实现。

# 5数据结构设计

## 5.1逻辑结构设计

整个系统各实体可以用以下关系来表示（其中主键放在前部用英文粗体,外键放在尾部用英文斜体表示）

1. 用户信息表( **user\_id,** 用户名,密码,区分标签,用户性别,用户电话,用户电子邮件,用户地址,头像)
2. 景点信息表(**attract\_ID**, 景点名字, 地理位置,电话, 邮箱, 等级,门票价格, 点赞数,浏览量,推荐游玩时间, *image\_ID)*
3. 评价信息表(**comment\_ID**,支持与否,内容,*user\_id*,*attract\_id*)
4. 图片信息表(**image\_ID**,图片位置,*attract\_ID*)·
5. 标签信息表(**lable\_ID,**标签信息,*attract\_ID*)
6. 聊天信息表(**char\_ID**,信息,时间, *attract\_ID,user\_ID*,)
7. 论坛信息表(**forum\_ID,**话题,内容,*user\_ID*)
8. 论坛回复表(**answer\_ID**,时间,回复信息,*forum\_ID,user\_ID*,)
9. 攻略信息表**(strategy\_ID**,主题,内容,图片存放位置,*attract\_ID*)
10. 评价信息表(**assessment\_ID**,*attract\_ID,user\_ID*,评分,内容)
11. 旅游社团信息表**(society\_ID**,,名字,时间,描述,*user\_ID)*
12. 组团游信息表(**group\_ID** , *society\_ID, user\_ID*)

## 5.2物理结构设计

### 5.2.1用户信息表（user）

用于记录用户的基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| user\_ID | varchar(64) | 否 | 主键,创建时赋予 |
| 用户名 | varchar(255) | 否 | 用户名称（如：2018年《信号与系统》试卷）,创建时赋值 |
| 密码 | datetime | 否 | 用户创建时赋值 |
| 区分标签 | int | 否 | 区分管理员与普通用户的标签 |
| 用户性别 | varchar(1) | 是 | 男或女 |
| 用户电话 | varchar(11) | 是 | 11长度的纯数字 |
| 用户邮箱 | varchar(255) | 是 | String |
| 用户地址 | varchar(255) | 是 | String |
| 头像 | varchar(255) | 是 | 头像图片的存放地址 |

//备注:下面的考虑一下，要用的话就去掉表12(group)，表11的负责人(user\_id)改成参加人和负责人，不用这样就当没有

|  |  |  |  |
| --- | --- | --- | --- |
| Society\_ID | varchar(64) | 是 | 有的话表示是某个团的负责人 |

### 5.2.2景点信息表（attraion）

景点信息表,用于存放景点相关信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| attract\_ID | varchar(64) | 否 | 主键,创建时添加 |
| 景点名称 | varchar(255) | 否 | 名称（如: 等于多少） |
| 地理位置 | varchar(255) | 否 | 创建时添加,不可修改 |
| 电话 | varchar(11) | 是 | 11长度纯数字 |
| 邮箱 | varchar(255) | 是 |  |
| 景点等级 | int | 是 | 0-5,0表示未评级 |
| 门票价格 | Int | 是 | Null即为0 |
| 点赞数 | Int | 是 | Null即为0 |
| 浏览量 | Int | 是 | Null即为0 |
| 推荐游玩时间 | varchar(255) | 是 | 包括start(开始时间),end(结束时间) |
| image\_ID | varchar(64) | 是 | 外键,图片信息表的主键,摸了  Null即为无图, |

### 5.2.3评价信息表（comment）

评价信息表,用于得到用户评价的相关信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| attract\_ID | varchar(64) | 否 | 主键 |
| 支持与否 | Int | 否 | 0-1 |
| 内容 | varchar(255) | 是 |  |
| user\_ID | varchar(64) | 否 | 发起评价的人 |

### 5.2.4图片信息表（image）

图片信息表,用于记录景点图片。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| image\_id | varchar(64) | 否 | 主键 |
| 图片存储位置 | varchar(255) | 否 | 为空你还留着这条信息干嘛,方便优化？ |
| attract\_ID | varchar(64) | 否 | 景点信息表的主键, 表示是哪个景点图片 |

### 5.2.5标签信息表（label）

评卷信息表,用于记录用户考试成绩。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| label\_ID | varchar(64) | 否 | 主键 |
| 内容 | varchar(255) | 否 | 见上面 |
| attract\_ID | Varchar(64) | 否 | 外键,景点信息表的主键,表示针对哪个景点 |

### 5.2.6 聊天信息表（chat）

用户信息表,用于记录聊天基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| chat\_ID | varchar(64) | 否 | 主键 |
| 内容 | varchar(64) | 否 |  |
| 时间 | Date | 否 | 或者换成string存放时间 |
| user\_ID | varchar(64) | 否 | 外键。用户信息表的主键,表示来自哪个用户 |
| attract\_ID | varchar(64) | 否 | 外键,景点信息表的主键,表示针对(来自)那个景点 |

### 5.2.7论坛信息表（forum）

论坛信息表,用于记录论坛基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| forum\_ID | varchar(64) | 否 | 主键 |
| 话题 | varchar(255) | 否 |  |
| 内容 | varchar(255) | 否 |  |
| user\_ID | varchar(255) | 否 | 外键,用户信息表的主键,表示来自那个用户 |

### 5.2.8论坛回复表（forum answer）

论坛信息表,用于记录论坛基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| answer\_ID | varchar(64) | 否 | 主键 |
| 内容 | varchar(255) | 否 |  |
| 时间 | Date | 否 | 可以用其他方式存放时间,比如String |
| user\_ID | varchar(64) | 否 | 外键,用户信息表的主键,表示来自那个用户 |
| forum\_ID | varchar(64) | 否 | 外键,论坛信息表的主键,表示来自那个论坛 |

### 5.2.9攻略信息表（strategy）

攻略信息表,用于记录攻略基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| strategy\_ID | varchar(64) | 否 | 主键 |
| 内容 | varchar(255) | 是 |  |
| 主题 | Varchar(255) | 否 |  |
| user\_ID | varchar(64) | 否 | 外键,用户信息表的主键,表示来自那个用户 |
| 图片存放地址 | varchar(64) | 是 |  |

### 5.2.10评价信息表（assess）

攻略信息表,用于记录攻略基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| assess\_ID | varchar(64) | 否 | 主键 |
| 内容 | varchar(255) | 是 |  |
| 评分 | Int | 否 | 0-5,0表示0分 |
| user\_ID | varchar(64) | 否 | 外键,用户信息表的主键,表示来自那个用户 |
| attract\_ID | varchar(64) | 否 | 外键,景点信息表的外键,表示针对那个景点打分 |

### 5.2.11旅游团信息表（society）

攻略信息表,用于记录攻略基本信息。

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| society\_ID | varchar(64) | 否 | 主键,去参加的旅游团 |
| 旅游团名称 | varchar(255) | 是 | 无名社团 |
| 介绍 | Varchar(255) | 是 |  |
| user\_ID | varchar(64) | 否 | 外键,用户信息表的主键,表示团的负责人 |

### 5.2.12组团信息表（group）

浪费，2个外键完全可以做主键，这样实际上就是1张表2个键，直接合并多好

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否允许空置 | 描述 |
| group\_ID | varchar(64) | 否 | 主键 |
| society\_ID | varchar(64) | 否 | 外键，旅游团的主键，摸了 |
| user\_ID | varchar(64) | 否 | 主键，同时也是外键,用户信息表的主键,表示团的参加者 |

## 5.3数据结构与程序关系设计

数据结构为关系型数据库，所以，在程序中可以通过标准的sql语句与数据结构进行交互，交互过程中采用通用的数据访问接口，为了保持良好的程序架构，对数据库访问采用dao设计模式实现，提高维护性及扩展性。

# 6运行设计

## 6.1运行模块的组合

本系统是由多个jsp页面及servlet构成，通过页面跳转来实现各个模块之间不同功能的链接和组合。各模块之间相互独立，属于高内聚低耦合，可以执行好。各模块之间主要通过jsp的内置对象以及动作指令标记实现模块之间的合作和数据共享。

## 6.2运行控制

操作按照操作说明以及流程图来进行，用户根据自身登陆之后的权限进行操作。对输入无特别限制，系统自行处理异常并给出提示信息。

## 6.3运行时间

由于是从数据库中调取文件，可能速度没有本地那么快，但基本无影响。

# 7出错设计

## 7.1出错输出信息

本程序多处采用了异常处理的机制, 当遇到异常时不但能及时的处理, 保证程序的安全性和稳定性, 而且各种出错信息能通过页面形式, 及时告诉用户出错的原因及解决的办法, 使用户以后能够减少错误的发生。 例如：当输入的账号类型，或者出现长度不正确就会弹出相应的提示界面。

## 7.2出错补救措施

数据库链接错误：

主要是因为数据库设置出问题，或者SQL server出现异常导致。只要取消本次操作，联系管理员对数据库进行排查。

输入错误：

用户在没注意输入账户要求是出现的错误，主要是通过页面提示解决的。

其他操作错误：

对于各种不正当操作可能发生的错误,我们主要是通过刷新页面或重新登录系统来解决问题。

# 8维护设计

软件的维护主要包括, 数据库的维护和软件功能的维护。

对于数据库的维护, 本系统主要由管理员对数据库基本结构进行管理维护。

对于系统功能方面的维护,由于我们采用的是模块化的设计方法, 每个模块 (页面) 之间相互独立性较高, 这样对系统的维护带来了很大的方便, 对于单独功能的修正只需要修改一个页面就行了。 对于功能的添加,只要再添加页面选项的内容即可。