第二届全国高校云计算应用创新大赛

作品综合设计报告

题　　目： 命题二：旅游比价决策系统

队　　名： MagicCloud

参赛队员： 张清恒、唐士杰、丁胜杰、郑燏辉

指导老师： 毛莺池

参赛学校： 河海大学

目录

一、绪论

1.1 背景

旅游本是享受，然而传统旅游的一成不变的模式，千篇一律的线路，成为游客集中抱怨的焦点，市场需要创新的、更适合中国人的旅游产品，颠覆传统旅游便成为一种必然。随着人们出游意识的不断成熟和旅游市场的完善，旅行社的角色定位也必须发生变化，从“提供产品”向“提供服务”转化，旅游进入后旅行社时代，游客的自我意识将越来越多地受到尊重和满足。“这是旅游市场成熟的一种表现”，游客想去玩什么地方，完全可以从旅游情报、网络获取包括景点、交通、饮食、住宿等各方面的充分资讯，现在兴起的自驾游不断满足现代人的旅游需求。正式进入旅游+互联网的时代，旅游体验会更好，旅游方式也会改变。

随着社会发展，人们更加重视精神方面的消费，而旅游因其陶冶性情、放松心情、增长见识的优势而备受人们的青睐。伴随着对旅游的需求的增长，各种旅游网站如雨后春笋一般迅速出现，然而作为一名普通用户，更看重的是价格是否优惠、内容是否实际以及旅游产品的性价比。因此，可以根据用户的个性化需求提供特定网站的特定产品（包括路线、价格、酒店、交通方式等）的旅游比价系统的出现就有了其重要性和必要性。

1.2 项目内容

该项目的主要目标是设计实现一个为旅游热爱者提供各旅游网站比价服务的系统。该平台将支持用户的在线比价，为用户带

来方便；将爬虫技术结合云存储技术，利用爬虫广泛收集旅游网站的信息，利用云存储技术存储关键信息；在数据安全的基础上为用户提供准确、全面的旅游信息；专注旅游信息的提供，为广大旅游爱好者们能相对经济、舒适地出行提供有力保障。

二、系统介绍

2.1 系统用户特点

本系统主要面向有出行旅游需求的用户。用户使用该系统进行不同旅游网站之间、不同旅游线路之间的比价与决策，对信息的准确性要求较高，同时信息及时更新也至关重要。对于部分用户而言，对于路线需求较多，使得系统须具备较快的运行速度。

2.2 系统功能性需求

2.2.1 旅游网站信息爬取

从给定的各个旅游网站爬取信息。在爬取过程中，爬虫将会不断发掘新的 URL，经过页面下载、URL 发现、URL 去重、目标页面信息存储几个步骤，完成一个完整的爬取过程。

2.2.2 信息存储

存储的信息主要包括：爬取获得的目标网页的全部信息、经过分析和提取之后得到的有价值信息、用户与管理员信息、目标旅游网站的首页地址信息等。

该部分采用关系型数据库（MySQL）与非关系型数据库（HBase）

相结合的方式，优势互补，提高系统运行效率。

2.2.3 关键信息抽取

从已爬取的网页内容中，提取出有价值的旅游信息，已爬取的网页以 json 格式存储在 HBase 数据库中，需要先从数据库中读取数据,然后由 json 格式转换成 html 形式的“page”,再利用 jsoup、Xpath、正则表达式等解析工具进行页面解析和提取工作。

2.2.4 比价决策

针对已经提取出来的目标信息进行比价决策，其中“比价”的方案不唯一，用户可以自行定制比价内容，系统默认按照价格高低进行决策。

2.2.4.1 旅游线路定制

对于用户定制路线而言，用户需要从系统给出的附加选项中，选择和自己相关的选项，系统根据用户提交的定制信息，针对该用户生成比价决策方案，进行比价与决策。

2.2.4.2 个性化推荐

系统可向用户推荐旅游路线，推荐的依据是用户之前选择的城市和各类旅游产品价格的高低，默认价格低者优先推荐，管理员可以设定默认值；此外，系统在用户尚无目的的情况下，会向用户推荐当前热门旅游信息，以供用户选择。

2.2.5 系统管理

该部分功能主要面向系统管理员和维护人员。系统将会自行监控自身运行情况，主要有：分布式集群结点的运行情况、系统访问量、比价策略更改、增删待爬取旅游网站列表、设置系统爬取频率和爬取时间、处理用户留言和请求等等。

2.3 系统非功能性需求

2.3.1 高效性

满足系统高效访问需求；爬取、存储、抽取等环节高效完成；满足不同用户、不同功能之间并发的需求。

2.3.2 稳定性

系统性能稳定、可靠运行，有较好的检错能力，对于单点故障，能够迅速有效解决，并且保证不丢失重要数据。

2.3.3 易用性

系统的用户图形界面友好，人性化，方便用户快速获取想要的旅游信息；系统管理界面能直观反映系统当前运行情况。

2.3.4 可扩展性

各业务系统具有良好的扩充能力，提供今后扩充系统功能、规模的接口，且系统功能扩充时不影响原系统的功能。

2.4 特色

2.4.1 智能化爬取

智能化爬取是指给定任意一个旅游网站的首页 URL，根据该首页 URL，系统会自动生成该网站的爬取规则，根据生成的专用爬取规则，进行目标页面的信息爬取。该特色满足设计原则中的开闭原则，使得系统在爬取模块具有较高的可扩展性。

2.4.2 用户可定制的比价决策机制

系统为用户“量体裁衣”，针对不同用户的不同需求，制定不同的比价决策机制，大大提高了系统的灵活性；同时，系统对用户制定的比价策略进行预处理，以保证策略的合理性。

2.4.3 个性化推荐

对于尚无目的的用户，系统提供个性化推荐服务，推荐主要以当前用户的过往记录和当前旅游相关热门信息为依据；同时，系统定期推出专题特色推荐。

2.4.4 可视化管理

系统管理功能为管理员以及系统维护人员，提供可视化的管理平台，可以实时监控集群的运行，设定包括爬取频率、爬取时间在内的配置信息；同时，系统若出现故障，管理员可第一时间获知并协助系统制定解决方案。

2.4.5 友好的用户体验

用户操作以简捷快速为原则，简化用户操作，方便用户使用；同时，系统用户界面以简约时尚为设计风格，友好的、美观的界面设计使得用户更容易接受和使用该系统。

2.5 创新点

2.5.1 轻量级的分布式框架

系统基于 zookeeper 实现 Master-Worker 主从式分布式框架，该框架定向完成爬取和处理等重要功能，并根据系统性能需求，借助 Redis 高速缓存的优势进行 URL 去重等操作。该框架轻巧方便，同时兼顾系统性能，也具有较高的可扩展性。

2.5.2 合理的任务分配机制

在自主实现主从分布式框架的基础上，对任务分配进行优化，实现集群负载均衡。任务分配优化的主要体现在 zookeeper 管理节点方面，系统借助 zookeeper 对各个节点的监控机制，实时监控任务执行状态，平衡各个节点之间的工作量，合理使用系统资源。

2.5.3 智能化生成爬取规则

对于一般爬虫而言，爬取规则是固定的，即爬去规则提前人为输入，此类爬虫仅对某一确定对象有效。鉴于本系统的需求特点，

需要面对多种多样的旅游网站，这意味着爬取规则需要不断更改。智能化生成爬去规则，对于某一确定对象（即某一确定旅游网站）而言是唯一确定的，通过对以网站首页为主的页面进行分析，获取相关爬取规则信息，然后生成针对该网站的爬去规则，进行爬取。

三、系统设计

3.1 总体架构

MagicCloud 旅游比价决策系统分为三大模块，分别为爬虫集群模块、数据存储模块和网站模块。系统总体框架如图 3-1 所示：

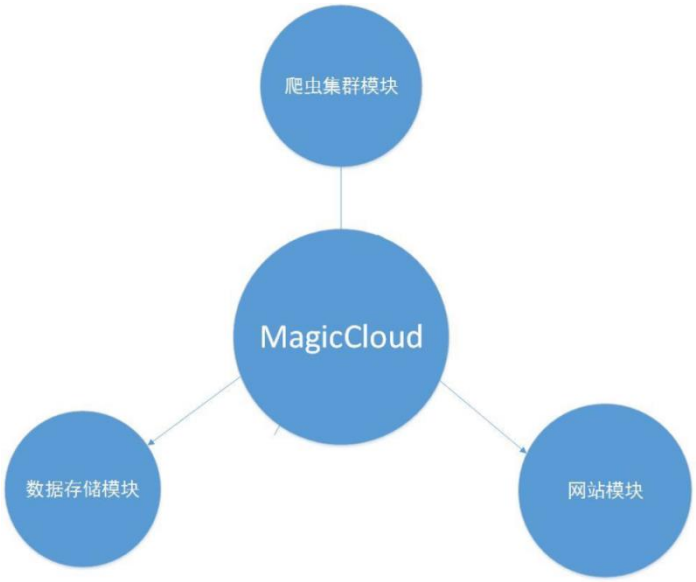


图 3-1 系统模块划分图

爬虫集群模块：主要负责分布式信息爬取，接收来自网站模块的爬取请求信息，并向数据存储模块（HBase）存入经过初步处理的信息。

数据存储模块：包括 HBase 存储子模块和 MySQl 存储子模块，该模块主要封装与数据库有关的所有功能，如连接、增删改查等。

网站模块：包括业务逻辑子模块和前端展示子模块，其中业务逻辑子模块以实现比价决策功能为主。

3.2 技术方案

本系统采用基于 SOA 的分布式应用框架和 B/S 结构，系统从服务器到客户端分为分布式爬取层、数据存储层、数据访问层、业务逻辑层、前端展示层（浏览器）。技术架构如图 3-2 所示:

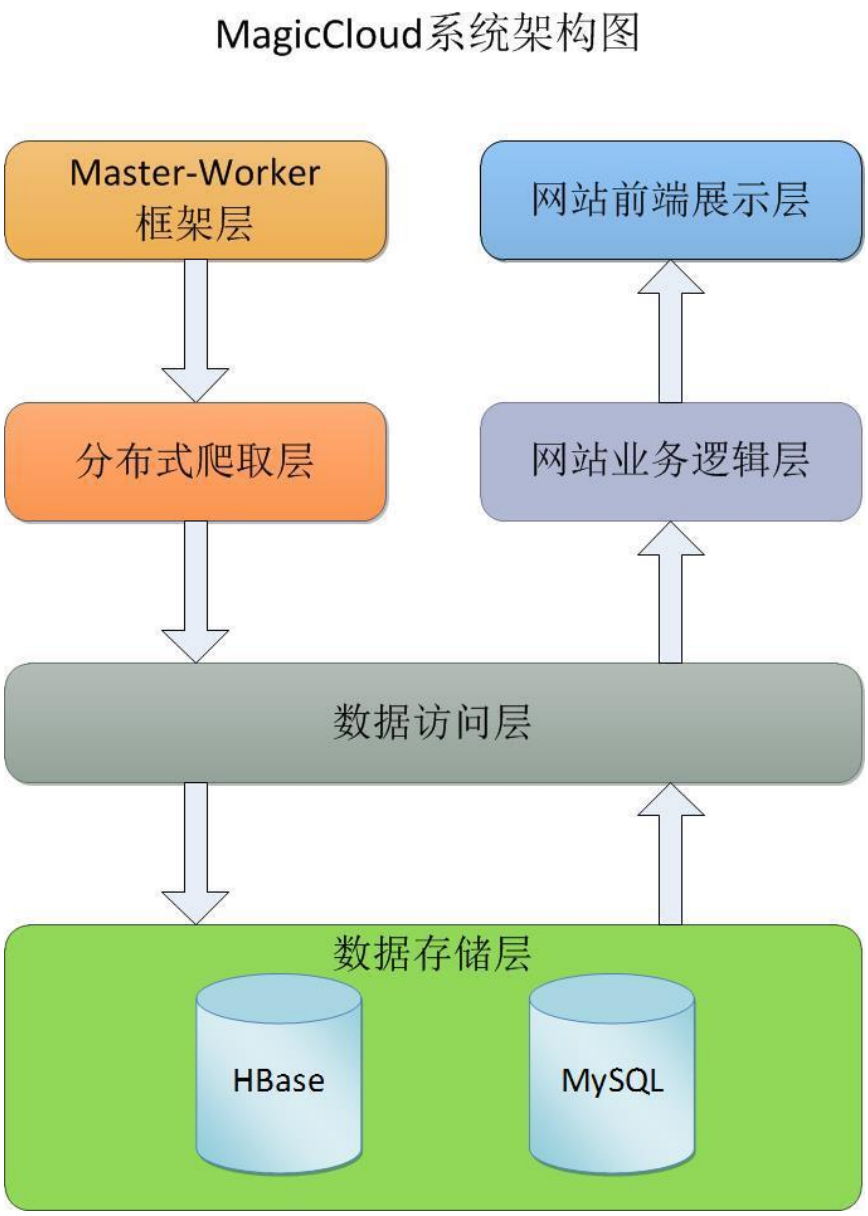


图 3-2 系统架构图

为了提高旅游比价决策系统的灵活性、可重用性、高可靠性及使用的方便性，整个系统分层实现，分布式爬取层和网站业务逻辑层的开发是整个系统的核心。其中业务逻辑层是实现数据操作及事务管理的，主要技术特征有：

（1）基于 SOA 的理念和技术，采用了 REST 架构风格发布服务，便于接入到用户体验平台和系统管理平台中。

（2）符合实际业务需求，通过处理用户不同的请求信息，系统进行用户定制或推荐路线的查找、比价、决策、收藏等操作，同时管理员也可以通过管理平台进行相关业务处理，方便快捷。

（3）减少业务处理层和服务提供层的耦合度，提高系统安全性，维护方便，具备数据容灾功能。

3.3 核心技术设计

3.3.1Master-Worker 框架

本系统的分布式爬取部分采用的架构基于 ZooKeeper 的 Master-Worker 主从工作机制。

在具体实现上，使用了一台云主机作 Master 节点（Leader），三台云主机作 Worker 节点（Followers），其中系统初始化时，Master 节点由人为指定。当 Master 节点故障时，会触发选举，将从三个 Worker 节点中选出新的 Master 节点，这也符合此框架的可靠性要求：参与选举的节点个数须为奇数。

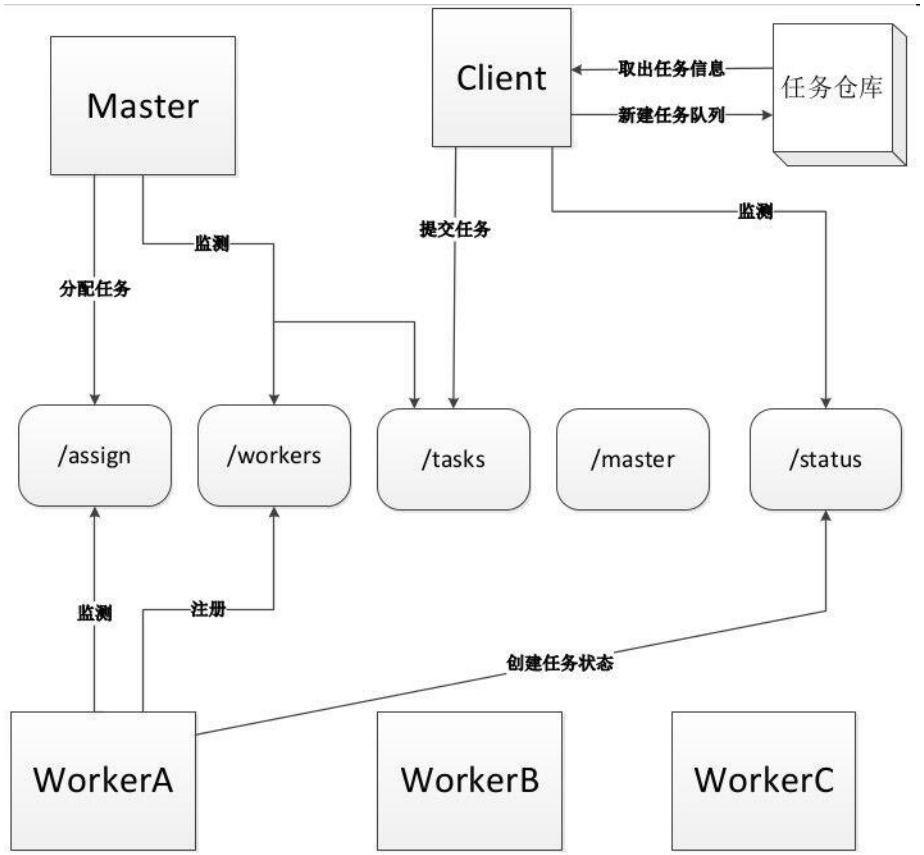


图 3-3 分布式爬虫框架图

3.3.2 爬取规则

本项目的需求中提到，系统应该可以实现由管理员进行控制，即由管理员手动添加待爬网站的网址，然后系统同样可以准确爬取新网站的内容。然而根据团队成员的观察，各旅游网站各目标页面的组织（即 URL 的命名规则）没有统一规则，若针对具体的全部旅游网站提取爬取规则，则这些规则不具备通用性，实际上不能实现上述需求，因此本团队决定尝试设计“万能”爬取规则，思路如下：

爬取规则基于一个前提，即认为在所有旅游网站的首页以及导航栏中提供的主题页面中给出的推荐列表均链接向对比价系统有

意义的页面，因此提取上述页面中推荐列表链接向的 URL，将全部这些 URL 按照“/”进行切割，构建一棵 URL 树，如 http://www.tuniu.com/tours/210055569 ，认为***.tuniu.com(观察发现各大旅游网站的这一部分都会改变)为根节点，然后 www.tuniu.com/tour 为一级节点，最后 www.tuniu.com/tours/210055569 为叶子节点，找出所有子节点大于等于 5 的节点，即认为这个节点是目标页面的父节点，将其作为其中一个爬取规则，遍历过全部 URL 之后即得到该旅游网站的爬取规则，由此可以实现对任意网站都能有效爬取到目标信息。

3.3.3 比价决策算法

使用余弦相似度算法，提取每个旅游产品的一系列特征组成一个 N 维的向量（价格，起点，终点，交通工具，天数），可以认为这是一个空间中的 5 维向量，每个旅游产品都具有这样一个向量，此时根据余弦定理，计算出这两个向量之间的夹角，若为 0°，则认为两个向量重合，若为 90°，则认为没有关系，若为 180° 则认为刚好相反，因此根据余弦定理的公式：

$$\cos\theta = \frac{\sum_{i=1}^{n}(A_i \times B_i)}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} = \frac{A \cdot B}{|A| \times |B|}$$

得到的值介于 0 到 1 之间，越接近 1 则认为相似度越高。当用户

浏览过本网站，产生浏览记录之后，则可以根据余弦向量法计算用户浏览过的产品和数据库中产品的相似度，将符合要求的产品（认为余弦相似度大于等于 0.95 位符合要求）定向推荐给特定用户。

而这个向量的取值问题，可以这样设定：

价格：直接使用价格数字；

起点和终点：我们使用全国城市的邮编，将城市转换成编号；

交通工具：同理，将所有交通工具编号；

天数：直接使用产品信息中的天数。

四、功能实现

4.1 旅游网站信息爬取

4.1.1 功能说明

信息爬取集群基于 ZooKeeper 的 Master-Worker 主从架构，一台云主机作 Master，三台作 Worker：

1.Master 分配任务到/assign 节点。

2./assign 监测各个 Worker，注册到/workers 节点（该节点由 Master 监测）。

3.Client 提交任务到/tasks 节点（由 Master 监测）。

4.已注册到/workers 的 Worker 去/tasks 中接受任务，同时创建任务状态到/status 节点（由 Client 监测，用于判断任务状态）。

之后每个 worker 启动一个 spider，开启具体爬取流程。

4.1.2 处理流程

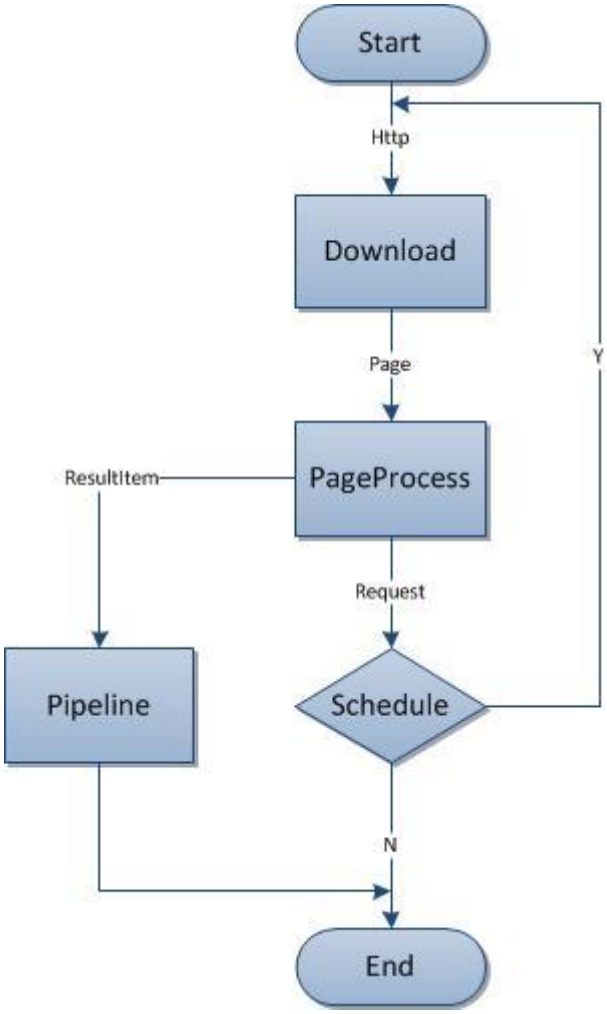


图 4-1 爬取流程图

1.Spider 启动后，爬取任务开始；

2.Spider 调用 Download,把某个 page 的全部内容获取并下载；

3.将 page 传入 PageProcess，PageProcess 将处理两个过程：将当前页面的内容封装成 ResultItem，交给 Pipeline 处理；将当前页面中的所有新的 URL 请求交给 Schedule 处理；

4.Schedule 获取到新的 URL 请求后，进行去重工作，将尚未爬取的 URL 请求提交给 Spider，进行新一轮爬取；

5.Pipeline 对 ResultItem 进行格式处理,并存入指定数据库。

4.1.3 关键实现技术描述

1、HttpClient：向客户端发送 Http 请求，接口基于 Http 协议实现，开发效率较高，具有一定的健壮性。

2、redis：是一个开源的使用 ANSI C 语言编写、支持网络、可基于内存亦可持久化的日志型、Key-Value 数据库，并提供多种语言的 API，本项目中主要用来存放待爬取 URL 列表和已爬取 URL 列表，巧妙利用其特点进行 URL 爬取前的去重工作。

4.2 关键信息抽取

4.2.1 功能说明

若在爬取页面信息的同时进行信息抽取工作，则整个系统的效率会完全受限于爬虫的运行效率，因此系统设计分为两个过程，首先将整个页面爬取得到，然后进行第二步，从中抽取信息，本功能即实现第二步，抽取信息。

4.2.2 处理流程

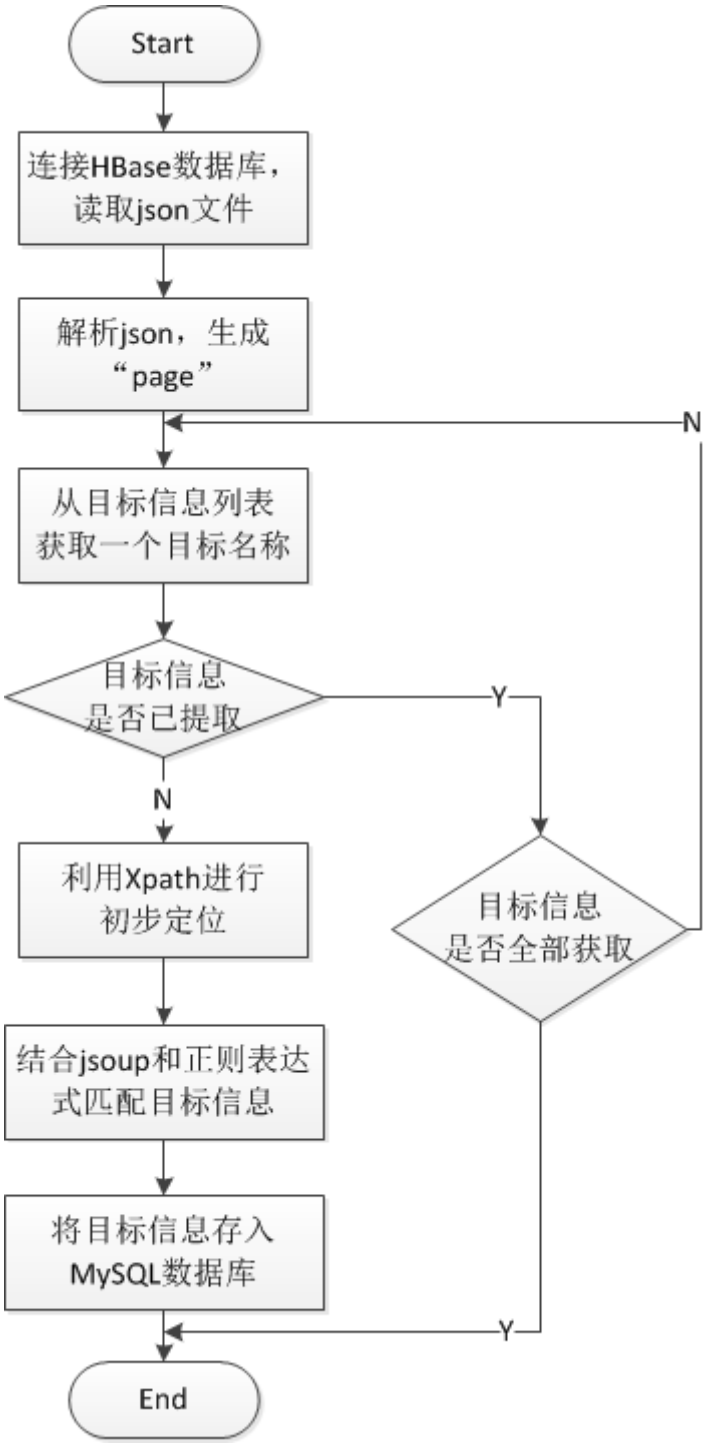


图 4-2 抽取流程图

1. 连接 HBase 数据库，并从中读取之前已经存为 json 格式的页面数据，此处读取时，以出发地点和目标地点为读取依据；

2. 从已设定的目标信息欲提取列表中，获取一个目标信息的名称；

3.检查该目标信息是否已被提取；

4.若该目标信息未被获取，先利用 Xpath 进行标签定位；

5.结合 Jsoup 和正则表达式，具体匹配该目标信息；

6.若该目标信息已经被获取，检查是否所有目标信息都已被获取；

7.将最终提取的具体目标信息，存入 MySQL 数据库。

4.2.3 关键实现技术描述

(1)json：是一种轻量级的数据交换格式，可以将 对象中表示的一组数据转换为字符串，然后就可以在函数之间轻松地传递这个字符串，或者在异步应用程序中将字符串从 Web 客户机传递给服务器端程序，本项目中主要用作中间形态的数据格式，与 Hbase 数据库交互。

(2)Xpath：即为 XML 路径语言，是一种用来确定 XML（标准通用标记语言的子集）文档中某部分位置的语言，XPath 基于 XML 的树状结构，提供在数据结构树中找寻节点的能力，项目利用 Xpath 进行初步定位，为后续操作做准备。

(3)Jsoup:一款 Java 的 HTML 解析器，可直接解析某个 URL 地址、HTML 文本内容，提供了一套非常省力的 API，可通过 DOM、CSS 以及类似于 jQuery 的操作方法来取出和操作数据，项目中主要用来做解析工作。

(4)正则表达式：使用单个字符串来描述、匹配一系列符合某个句法规则的字符串，在 Xpath 初步定位的情况下，使用正则表达式来完成具体的匹配和筛选。

4.3 比价决策

4.3.1 用户定制

4.3.1.1 功能说明

该功能为整个系统的核心功能，根据用户在前端页面键入或鼠标点击给出的筛选条件，从数据库中提取相应信息并予以显示。

4.3.1.2 处理流程

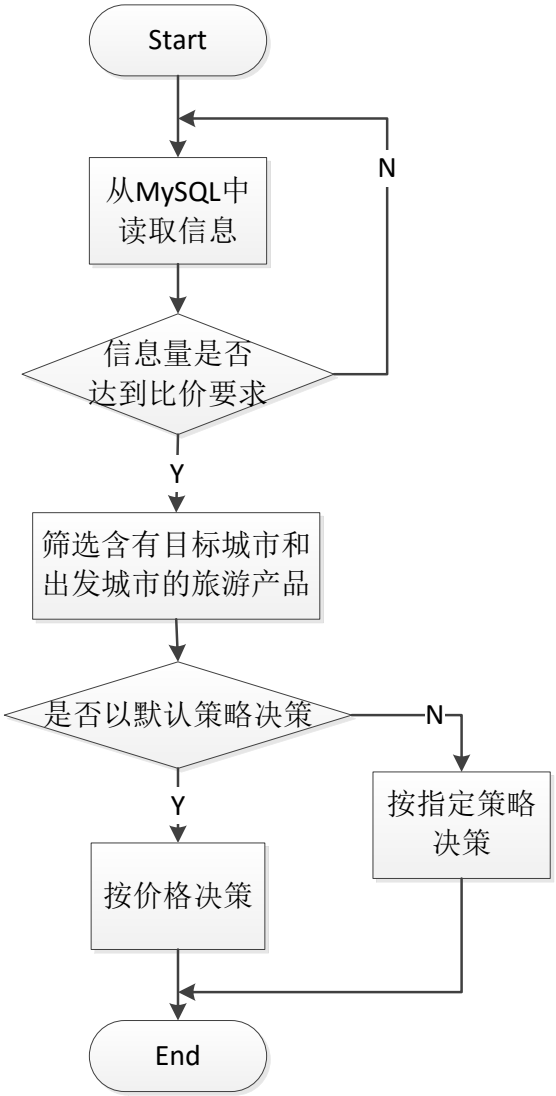


图 4-3 比价决策流程图

(1)从 MySQL 中读取已经提取完毕的旅游信息，读取时，为保证没有信息遗失，只要与目标信息有相似或相同的内容，就认为可以进行比较；

(2)系统默认有同时进行比较的信息数量，用户可自行设置，当达到该数量时，方可触发比价决策；

(3)筛选含有目标城市和出发城市的旅游产品，也可以按照其他特殊信息（如旅游天数、中间城市等）进行筛选；

(4)若以默认策略比价决策，系统将按照目标旅游产品的价格进行比价决策；

(5)若用户有具体的要求，则按照用户定制的策略进行比价决策，并最终将决策结果反馈给用户。

4.3.1.3 关键实现技术描述

比价决策算法：该部分内容从实现角度上看，关键技术为比价决策算法；系统默认算法为价格优先，意味着价格是权值最大的因素；用户自行定制旅游线路时，系统将会获取用户选择的附加选项和选择的先后顺序，按照先后顺序赋予不同权值，提交时顺序靠前者权值大。

4.3.2 个性推荐

4.3.2.1 功能说明

该功能致力于提高用户体验，根据用户以往的查询记录推测用户喜好，根据设定的算法计算出用户最有可能喜欢的旅游产品、线路、

价位、酒店等信息，通过前端页面推送给用户，实现信息的个性化推荐。

4.3.2.2 处理流程

（1）用户使用本系统，一段时间后积累一定量的浏览记录信息；

（2）系统按照预定算法分析得到的用户浏览记录，得到特定用户的特定喜好（用户可能感兴趣的信息）；

（3）根据分析结果，定向推送特定信息给特定用户，实现个性化的推荐。

4.3.2.3 关键实现技术描述

实现本功能的关键在于推荐算法，根据目前已有的成熟推荐算法（如余弦向量法）并根据系统具体情况进行一定的修正和改进，以适应具体应用场景。

4.4 系统管理

4.4.1 功能说明

根据项目需求，系统投入运行后，管理员应能控制整个系统运行的频率，以及对供系统参考的旅游网站站点进行增删改查，本功能即实现上述目标。

4.4.2 处理流程

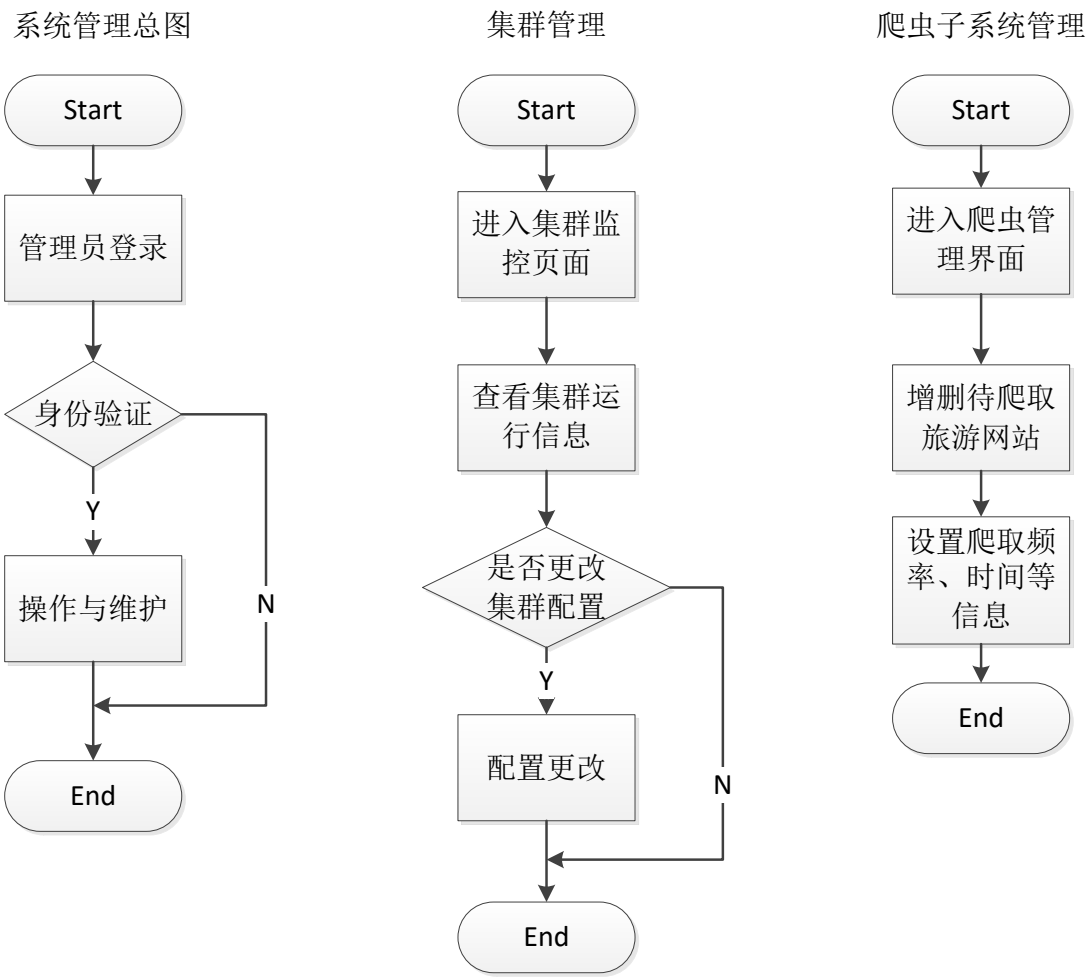


图 4-4 系统管理流程图

（1）系统管理总流程：

1）管理员登录，并验证身份；

2）管理员选择是否进行管理和维护。

（2）集群管理子流程：

1）进入集群监控界面；

2）查看当前集群运行情况；

3）跟据当前运行情况，选择是否更改配置信息；

4）具体更改信息。

（3）爬虫管理子流程：

1）进入爬虫管理界面；

2）增删待爬取旅游网站首页；

3）设置爬虫爬取的频率、时间等信息。

4.4.3 关键实现技术描述

（1）AdminClient:集群接口，用于提供集群运行的详细信息。

（2）Bootstrap：前端采用的是 Bootstrap，来自 Twitter，是目前很受欢迎的前端框架。Bootstrap 是基于 HTML、CSS、JAVASCRIPT 的，它简洁灵活，使得 Web 开发更加快捷。Bootstrap 提供了优雅的 HTML 和 CSS 规范，它即是由动态 CSS 语言 Less 写成。国内一些移动开发者较为熟悉的框架，如 WeX5 前端开源框架等，也是基于 Bootstrap 源码进行性能优化而来。其具有相当简洁的 css 样式，组件与插件封装。

（3）Structs2: 后端采用 Struts2 框架，Struts2 是在 struts 1 和 WebWork 的技术基础上进行了合并的全新的 MVC 框架。其全新的 Struts 2 的体系结构与 Struts 1 的体系结构差别巨大。Struts 2 以 WebWork 为核心，采用拦截器的机制来处理用户的请求，使用非侵入式设计，使得业务逻辑控制器能够与 ServletAPI 完全脱离开，使用它可以大大减轻后端开发的工作量，使软件开发人员能更加专注于业务逻辑的设计。

五、数据库设计

5.1 外部设计

5.1.1 标识符和状态

数据库软件：Hbase 0.98

MySQL 5.0

数据库运行环境：Ubuntu14.04

5.1.2 使用它的系统

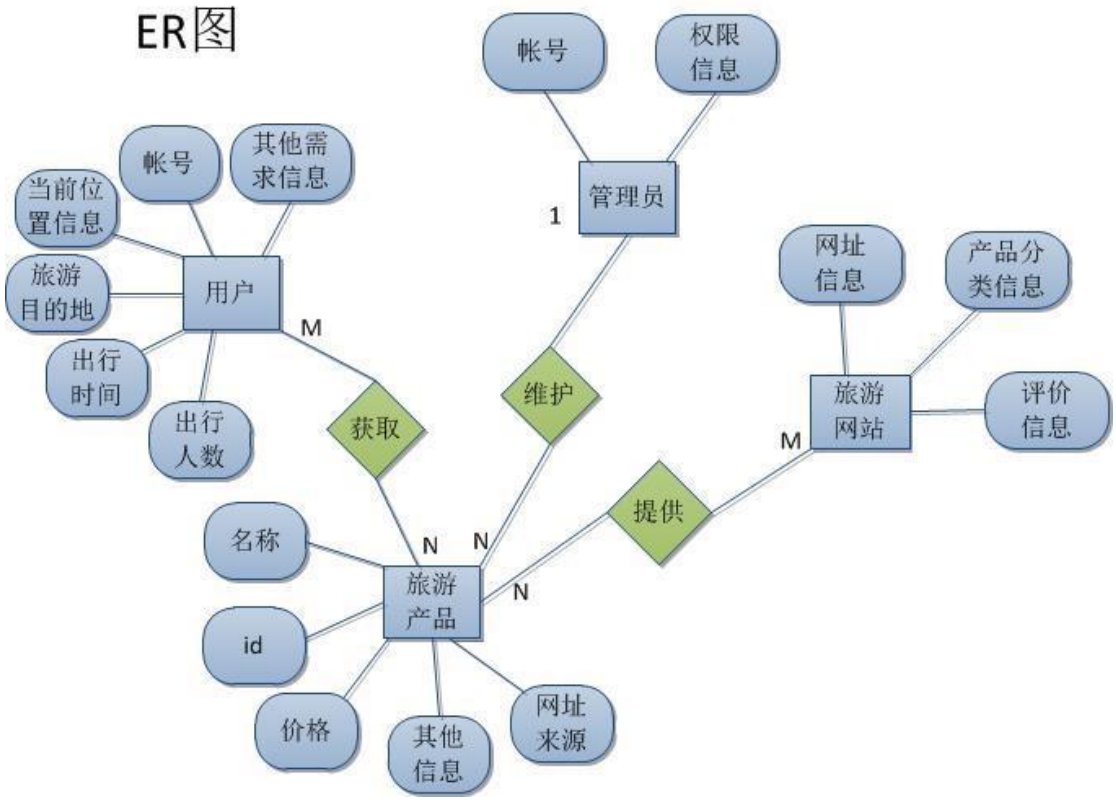
本数据库服务于 MagicCloud 团队的 MagicCloud 系统。

5.1.3 命名约定

本数据库命名均已 mc_ 开头，再加上表内容的英文单词或英文短语缩写，这样可以统一命名，并且见名知意。

5.2 概念结构设计

5.2.1 实体—关系图



5.2.2 数据库表设计

表 5-1 数据库表

表名	中文表名	数据库类型	功能描述
mc_user	用户信息表	MySQL	存储用户信息
mc_resources	线路资源表	MySQL	存储线路信息
mc_history	浏览与收藏表	MySQL	存储浏览记录
mc_recommend_resources	推荐资源表	MySQL	存储首页推荐资源信息

mc_article	旅游文章表	MySQL	存储文章信息
mc_resource_page	旅游路线爬取资源表	HBase	存储爬取的目标页面的信息

5.3 逻辑结构设计

5.3.1 数据库表设计

5.3.1.1 用户信息表 mc_user 设计

用户信息表用于存储用户的注册信息和统计信息，其中，对于用户登录密码要进行 MD5 加密存储。

字段名	字段类型	字段解释	示例
uid	int	主键;用户 ID	1001001
username	varchar	用户名	magic
password	varchar	用户密码	BFE94C76028861FD
telephone	varchar	联系方式	13845678901
email	varchar	电子邮件	123@hhu.edu.cn
authority	int	权限：1 代表管理员	1

5.3.1.2 路线资源表 mc_resources 设计

该表用于存储系统处理后的旅游线路资源，其原数据来自于 mc_resource_page，其中 picture 字段存储图片的相对路径。

字段名	字段类型	字段解释	示例
rid	int	主键；路线 ID	1001001
name	varchar	路线名称	<厦门 4 日自由行>双人立减 400，充足自由，好评过千
start_city	varchar	出发城市	南京
end_city	varchar	目的城市	厦门
days	int	出行天数	4
picture	varchar	图片相对路径	/resouces/pic/nj-xm-000001
grade	double	来自该网站的评分	0.94
price	int	价格	783
source_web	varchar	来源网站	途牛
url	varchar	线路 url	http://nj.tuniu.com/tours/5019858

5.3.1.3 推荐资源表 mc_recommend_resources 设计

该表用于存储系统首页的动态推荐的线路信息，系统会定期更新当前推荐，该表与 mc_resources 表关联。

字段名	字段类型	字段解释	示例
rrid	int	主键；推荐资源 ID	3000001
rid	int	详细资源 ID	1001001
category1_1	varchar	一级推荐目录 1	国内游览
category2_1	varchar	二级推荐目录 1	江南姿色
category1_2	varchar	一级推荐目录 2	为您推荐
category2_2	varchar	二级推荐目录 2	--

5.3.1.4 浏览与收藏表 mc_history 设计

该表用于存储用户的浏览记录和收藏信息，用字段 collection_arr 存储用户收藏的旅游产品的 rid，用字段 browse_arr 存储用户的浏览记录。该表与表 mc_user 和表 mc_resources 有关联关系。

字段名	字段类型	字段解释	示例
hid	int	主键；浏览信息记录 ID	1000001
uid	int	用户 ID	1001001
browse_arr	varchar	浏览记录数组	3000001,3000002,3000003
collection_arr	varchar	收藏线路数组	3000003,3000005,3000010

5.3.1.5 旅游文章表 mc_article 设计

该表用于存储旅游文章信息。

字段名	字段类型	字段解释	示例
aid	int	主键；文章 ID	4000001
title	varchar	文章题目	带着妈妈，步行厦门
source_web	varchar	来源站点	途牛
author	varchar	作者	樱之鹅
city	varchar	相关城市	厦门
picture	varchar	图片相对路径	/resouces/pic/wz/xm-000001
url	varchar	文章链接	http://www.tuniu.com/guide/d-xiamen-414/youji/

5.3.1.6 旅游路线爬取资源表 mc_resource_page 设计

该表为 HBase 类型，用于存储爬取的目标页面的完整信息。

主键/列族	字段名	字段解释	示例
rowkey	key	主键，线路名称	nj_bj_tuniu_0001
timestamp	--	时间戳	1459168288
var	start_city	线路出发城市	南京
	end_city	线路目的城市	北京
	page	线路完整页面	<html>.....</html>

六、业务应用系统设计

6.1 业务总体架构

MagicCloud旅游比价决策系统由旅游比价决策平台和系统管理平台两部分组成。其结构如图 6-1 所示：

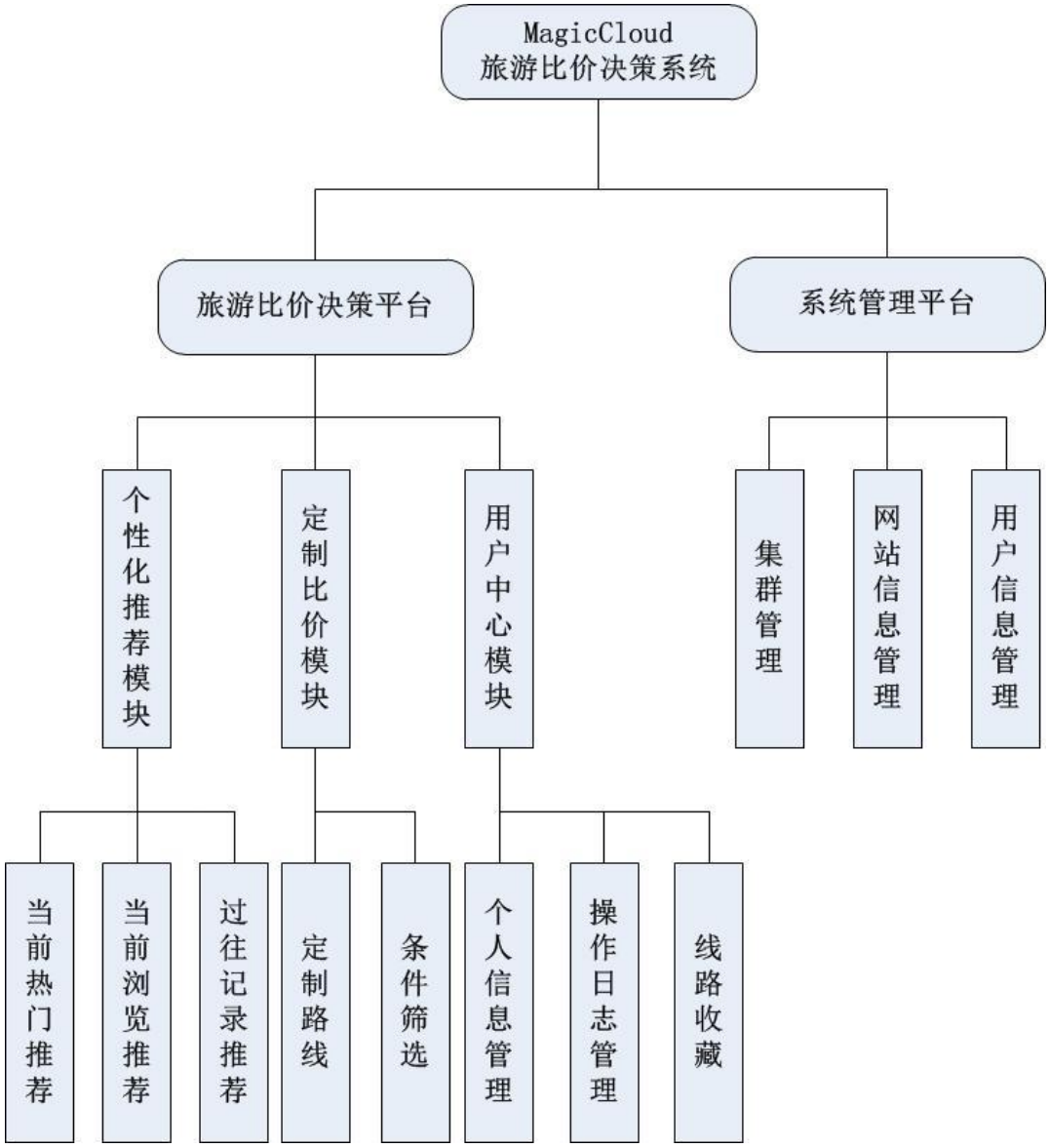


图 6-1 业务应用系统总体结构

6.1.1 旅游比价决策平台

旅游比价决策平台，可实现旅游线路的定制与推荐，包括当前热

门旅游产品的推荐、基于用户浏览信息产生的推荐、基于用户过往记录的推荐和用户个性化定制。同时，用户可以查看或修改个人信息、操作日志、收藏信息。

6.1.2 系统管理平台

系统管理平台，可实现对系统性能、集群运行情况的检测，管理爬取信息，其中包括爬取网站的增加与删除、爬取频率的设置等。该模块需要管理员权限才能访问。

6.2 业务功能模块详细设计

6.2.1 旅游比价决策子系统

6.2.1.1 系统用例

旅游比价决策子系统的用户，可分为系统管理员和普通用户，其用例图如图 6-2 所示：

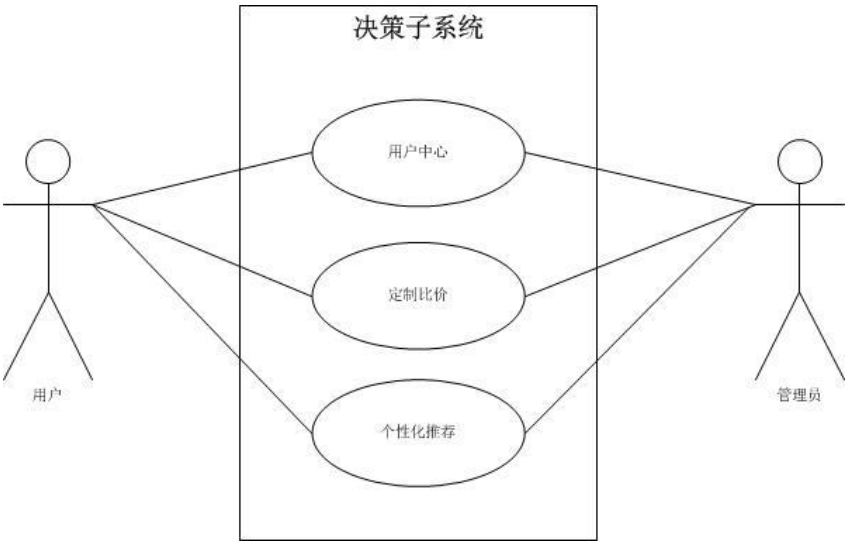


图 6-2 旅游比价决策子系统用例图

6.2.1.2 功能详细设计

1、定制比价模块

功能描述：根据用户的要求，定向为用户提供比价服务。

用户输入：出发地、目的地等定制信息。

系统输出：根据用户的定制信息得到的比价结果。

相关数据库表：旅游产品信息表。

定制比价模块顺序图如图 6-3 所示：

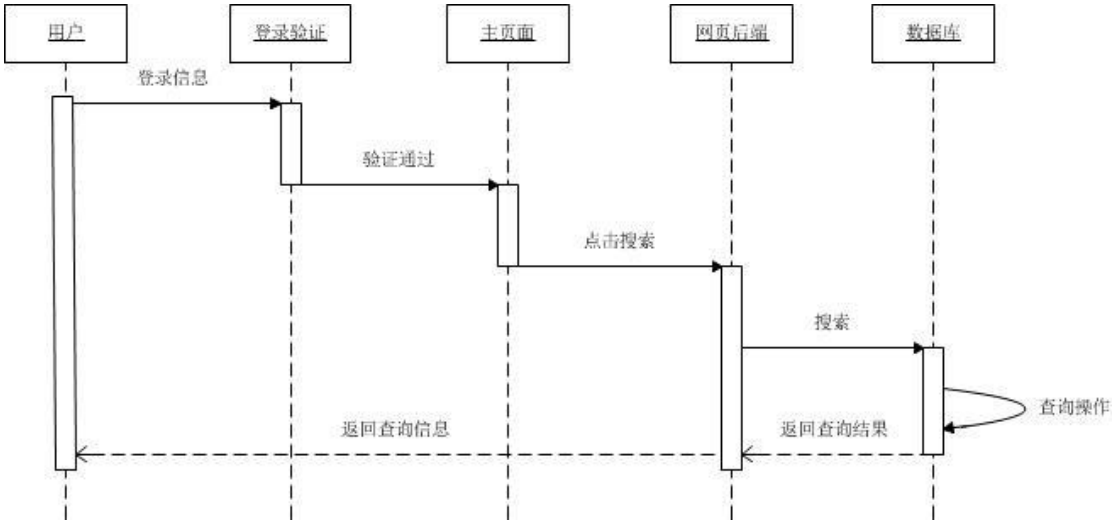


图 6-3 定制比价模块顺序图

2、个性化推荐模块

功能描述：根据用户的浏览记录，推荐用户可能感兴趣的产品。

用户输入：历次访问留下的浏览记录。

系统输出：根据浏览记录得到的个性化推荐结果。

相关数据库表：旅游产品信息表。

个性化推荐模块顺序图如图 6-4 所示：

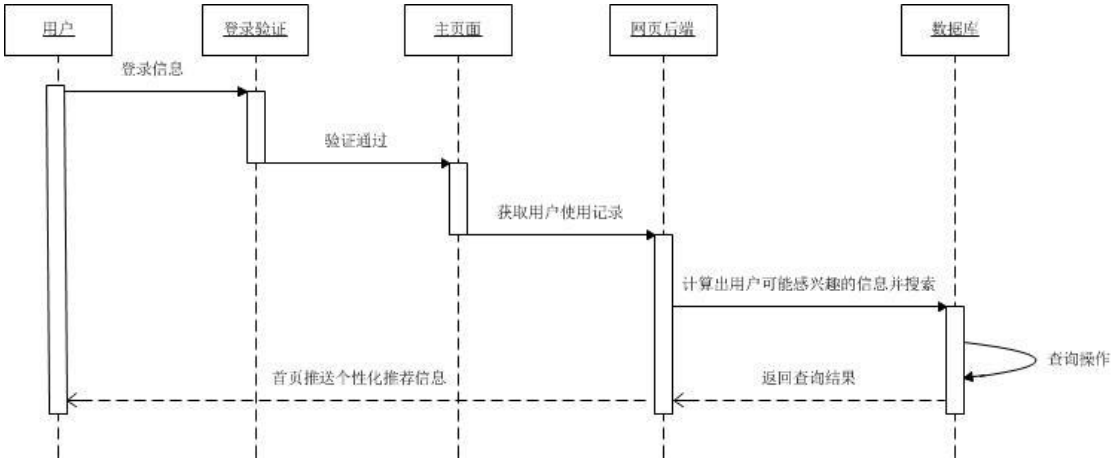


图 6-4 个性化推荐模块顺序图

3、用户中心模块

功能描述：用户查看、修改个人资料、查看个人日志等。

用户输入：修改的个人资料，查看的日志页码。

系统输出：返回用户对个人资料的修改结果，返回查询的日志。

相关数据库表：用户表、日志表。

用户中心模块顺序图如图 6-5 所示：

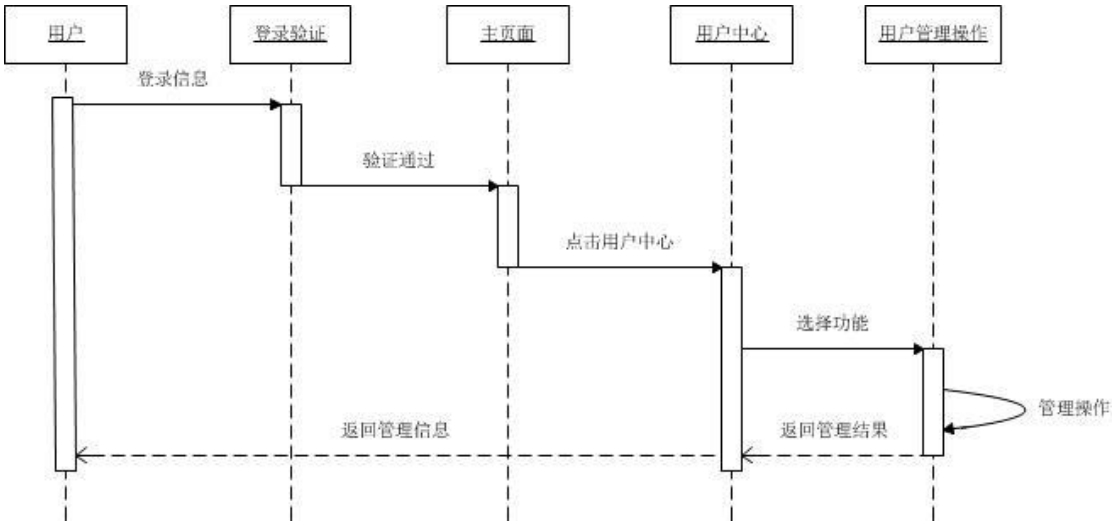


图 6-5 用户中心模块顺序图

6.2.2 系统管理子系统

6.2.2.1 系统用例

系统管理子系统的用户仅为系统管理员，其用例图如图 6-6 所示：

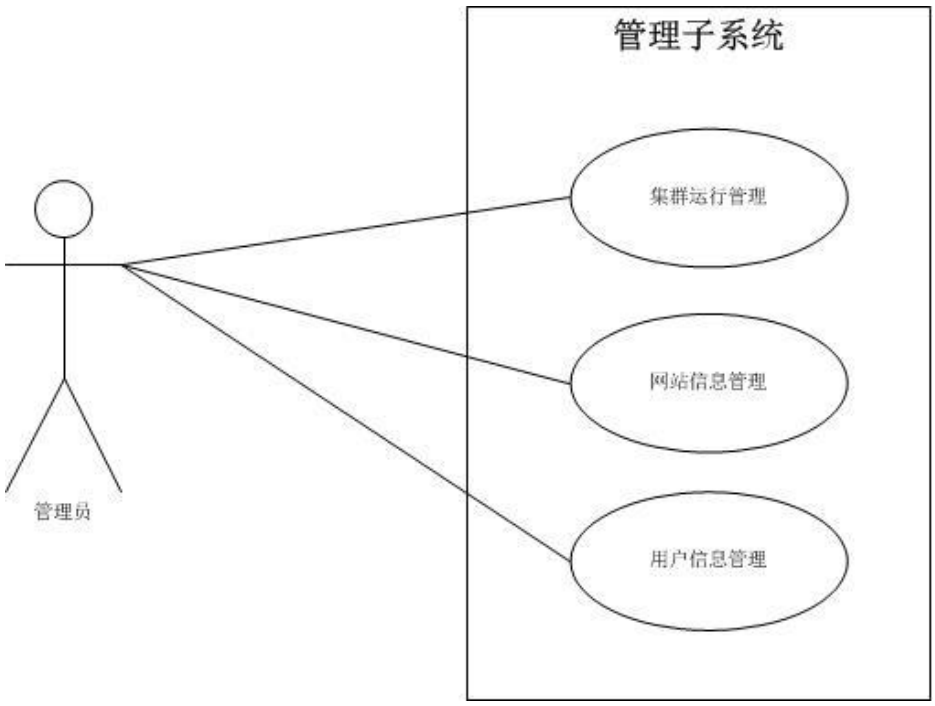


图 6-6 系统管理子系统用例图

6.2.2.2 功能详细设计

1、集群管理模块

功能描述：管理员查看集群运行情况、设置集群运行频率等。

用户输入：点击相应按钮。

系统输出：当前集群运行情况。

相关数据库表：日志表。

集群管理模块顺序图如图 6-7 所示：

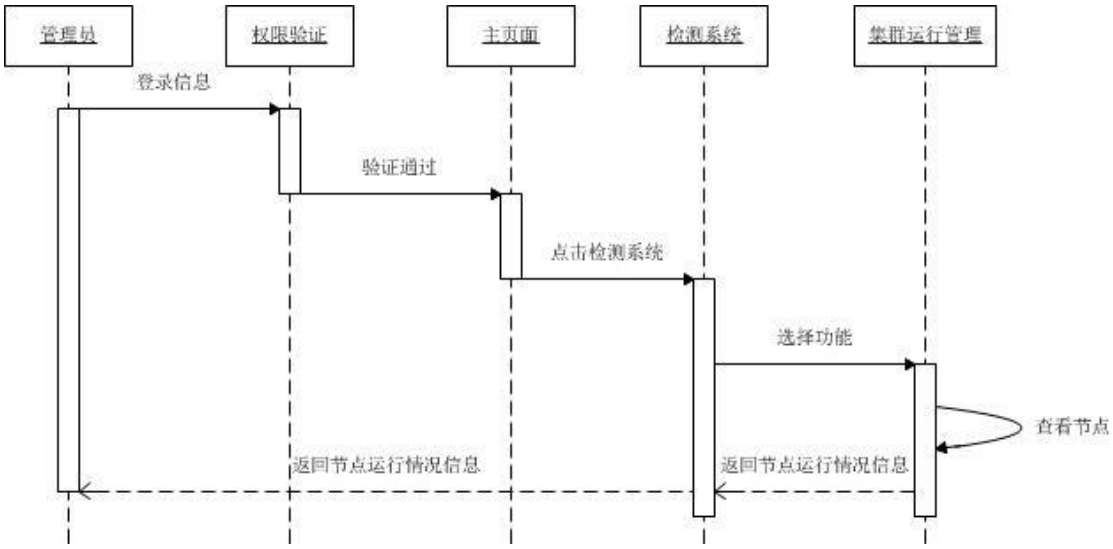


图 6-7 集群管理模块顺序图

2、网站信息管理模块

功能描述：管理员设置本系统的目标网站。

用户输入：希望系统爬取的旅游网站 URL。

系统输出：系统给出的反馈信息，同时集群开始爬取该网站信息。

相关数据库表：待爬网站信息表。

网站信息管理模块顺序图如图 6-8 所示：

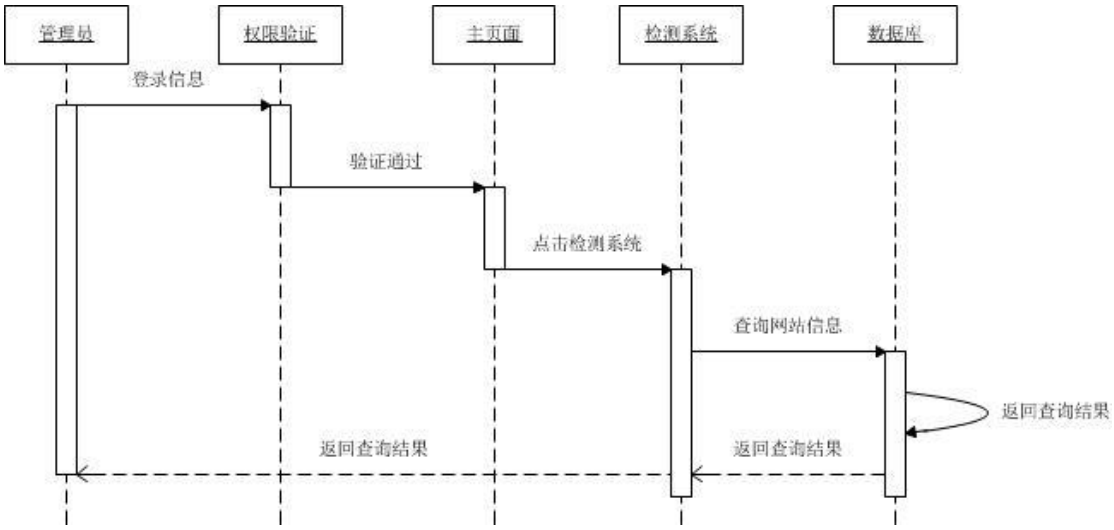


图 6-8 网站信息管理模块顺序图

3、用户信息管理模块

功能描述：管理员管理本系统的普通用户信息。

用户输入：点击相应按钮。

系统输出：全部普通用户信息列表。

相关数据库表：用户信息表。

用户信息管理模块顺序图如图 6-9 所示：

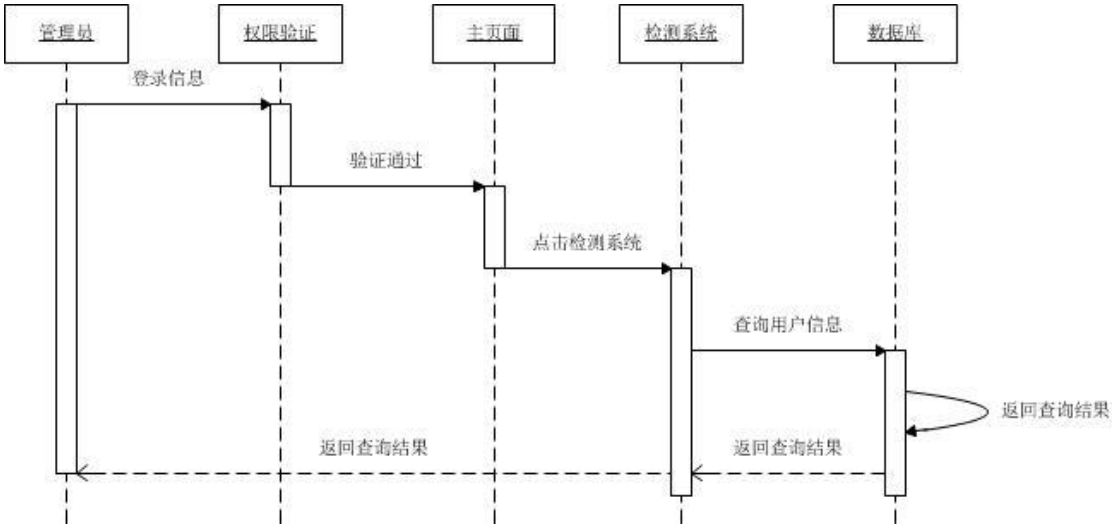


图 6-9 用户信息管理模块顺序图

6.3 界面详细设计

6.3.1 旅游比价决策子系统

系统界面如图 6-10 所示：

图 6-10 首页部分截图

6.3.2 系统管理子系统

管理平台界面如图 6-11、6-12 所示：

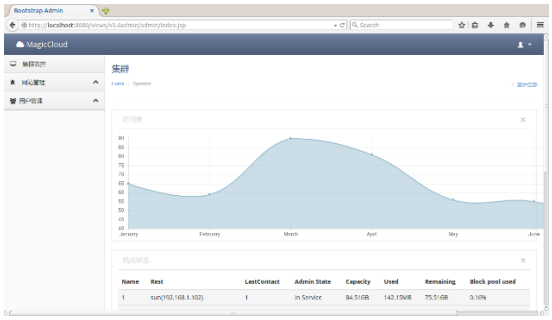


图 6-11 管理平台界面 1

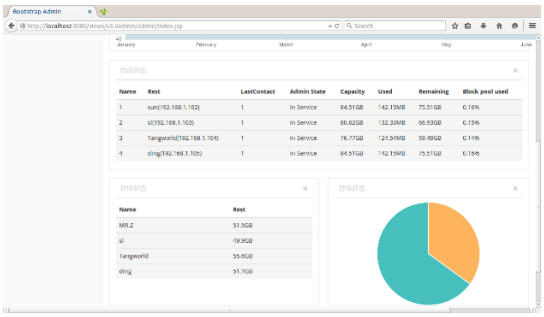


图 6-12 管理平台界面 2

七、系统测试

7.1 测试环境

7.1.1 集群运行结点

配置	Master 性能参数	Worker 性能参数
处理器 CPU	1 核	1 核
主存	1024 MB	1024 MB
操作系统	Ubuntu 14.04 64 位	Ubuntu 14.04 64 位
带宽	1Mbps（峰值）	1Mbps（峰值）
云环境	Hadoop 2.4.0	Hadoop 2.4.0
数据库	Hbase 0.98.16.1 + MySQL 5.5.14	Hbase 0.98.16.1 + MySQL 5.5.14

7.1.2 服务器端

配置	性能参数
处理器 CPU	1 核
主存	1024 MB
操作系统	Ubuntu 14.04 64 位
带宽	1Mbps（峰值）
服务器环境	Nginx + Tomcat + Apache
云环境	Hadoop 2.4.0
数据库	Hbase 0.98.16.1 + MySQL 5.5.14

7.1.3 客户端

客户端只需要一个浏览器即可。为了更好的用户体验，在测试时使用了 Firefox 浏览器。

7.2 主要功能测试

7.2.1 单元测试

7.2.1.1 注册单元

功能	输入	预期输出	实际输出
注册	账号/密码： null/null wer/123	①用户名或密码不能为空！②注册成功	一致

7.2.1.2 登录单元

功能	输入	预期输出	实际输出
登录	账号/密码： wer/123 null/null 123/123	①登录成功②用户名或密码不能为空！③密码错误	一致

7.2.1.3 主页搜索单元

功能	输入	预期输出	实际输出
搜索本地至某地的旅游产品	目的地	本地到目的地的旅游产品信息	一致

7.2.1.4 搜索页搜索单元

功能	输入	预期输出	实际输出
搜索某地到某地的旅游产品	出发地/目的地	出发地到目的地的旅游产品信息	一致

7.2.1.5 查看网站信息单元

功能	输入	预期输出	实际输出
查看网站信息	点击“更新”按钮	全部待爬取网站信息：网站名/网址	一致

7.2.1.6 添加网站单元

功能	输入	预期输出	实际输出
添加待爬网站	网站名/网址：途牛/www.tuniu.com null/null	①点击“更新”按钮后显示新添加的网站信息②点击“更新”按钮后显示的内容没有变化	一致

7.2.1.7 删除网站单元

功能	输入	预期输出	实际输出
删除待爬网站	点击“删除”按钮	被删除的网站信息立刻消失	一致

7.2.1.8 查看用户信息单元

功能	输入	预期输出	实际输出
查看用户信息	点击“更新”按钮	在页面内显示全部用户信息（密码不予显示）	一致

7.2.1.9 赋予管理员权限单元

功能	输入	预期输出	实际输出
赋予某普通用户管理员的权限	点击“赋予管理员权限”按钮	该用户信息中“角色”一栏由“普通用户”变为“管理员”	一致

7.2.1.10 设置爬取频率单元

功能	输入	预期输出	实际输出
设置爬取频率	选择频率值并点击“提交”按钮	爬虫启动频率成为管理员设置的值	一致

7.2.1.11 上次爬取单元

功能	输入	预期输出	实际输出
查看上次爬取信息	点击“上次爬取”选项	页面显示上次爬取的相关信息	一致

7.2.1.12 集群运行单元

功能	输入	预期输出	实际输出
查看集群运行情况	点击“集群运行情况”选项	页面展示集群运行情况信息，包括节点信息，流量监控等信息	一致

7.2.1.13 查看日志单元

功能	输入	预期输出	实际输出
查看日志	点击日志查看	显示近期操作日志	一致

7.2.1.14 推荐单元

功能	输入	预期输出	实际输出
推荐旅游产品	页面展示,用户点击对应区域	链接到推荐信息对应的网站	一致

7.2.2 集成测试

7.2.2.1 用户管理模块

功能	输入	输出	实际输出
注册	账号/密码： null/null wer/123	①用户名或密码不能为空！②注册成功	一致
登录	账号/密码： wer/123 null/null 123/123	①登录成功②用户名或密码不能为空！③密码错误	一致

7.2.2.2 核心服务模块

功能	输入	输出	实际输出
搜索本地至某地的旅游产品	目的地	本地到目的地的旅游产品信息	一致
搜索某地到某地的旅游产品	出发地/目的地	出发地到目的地的旅游产品信息	一致
推荐旅游产品	页面展示，用户点击对应区域	链接到推荐信息对应的网站	一致

7.2.2.3 管理员模块

功能	输入	输出	实际输出
查看网站信息	点击“更新”按钮	全部待爬取网站信息：网站名/网址	一致
添加待爬网站	网站名/网址：途牛/www.tuniu.com null/null	①点击“更新”按钮后显示新添加的网站信息②点击“更新”按钮后显示的内容没有变化	一致
删除待爬网站	点击“删除”按钮	被删除的网站信息立刻消失	一致
查看用户信息	点击“更新”按钮	在页面内显示全部用户信息（密码不予显示）	一致
赋予某普通用户管理员的权限	点击“赋予管理员权限”按钮	该用户信息中“角色”一栏由“普通用户”变为“管理员”	一致
设置爬取频率	选择频率值并点击“提交”按钮	爬虫启动频率成为管理员设置的值	一致
查看上次爬取信息	点击“上次爬取”选项	页面显示上次爬取的相关信息	一致
查看集群运行情况	点击“集群运行情况”选项	页面展示集群运行情况信息，包括节点信息，流量监控等信息	一致

7.2.2.4 用户中心模块

功能	输入	输出	实际输出

查看日志	点击日志查看	显示近期操作日志	一致

7.2.3 确认测试

7.2.3.1 首页浏览

图 7-1 首页

7.2.3.2 注册

点击主页右上方的“注册”按钮，如图 7-2 所示：

图 7-2 注册

然后，输入注册需要的用户名、密码等必要信息，输入完后，点击“注册”按钮。

7.2.3.3 登录

点击主页右上方的“登录”按钮，如图 7-3 所示：

图 7-2 登录

然后输入用户名和密码，例如：用户名输入 zhangsan，密码为 *********，点击“立刻登录”。登录成功后，如图 7-3 所示：

图 7-3 登录成功

7.2.3.4 首页推荐

在首页可以看到五个方面的推荐内容。“为您推荐”是综合推荐，

如图 7-4 所示；“国内游览”是国内旅游资源推荐，如图 7-5、7-6 所示；“世界环游”是国外旅游资源推荐，如图 7-7、7-8 所示；“身边美景”是基于用户当前定位的推荐，如图 7-9 所示；“旅游灵感”是旅游文章的推荐，如图 7-10 所示。

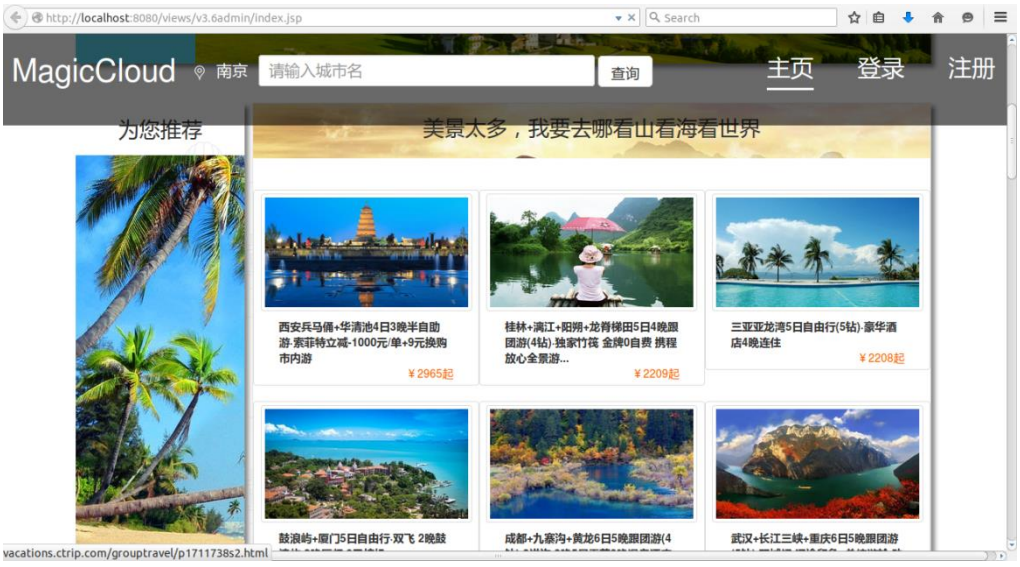


图 7-4 为您推荐

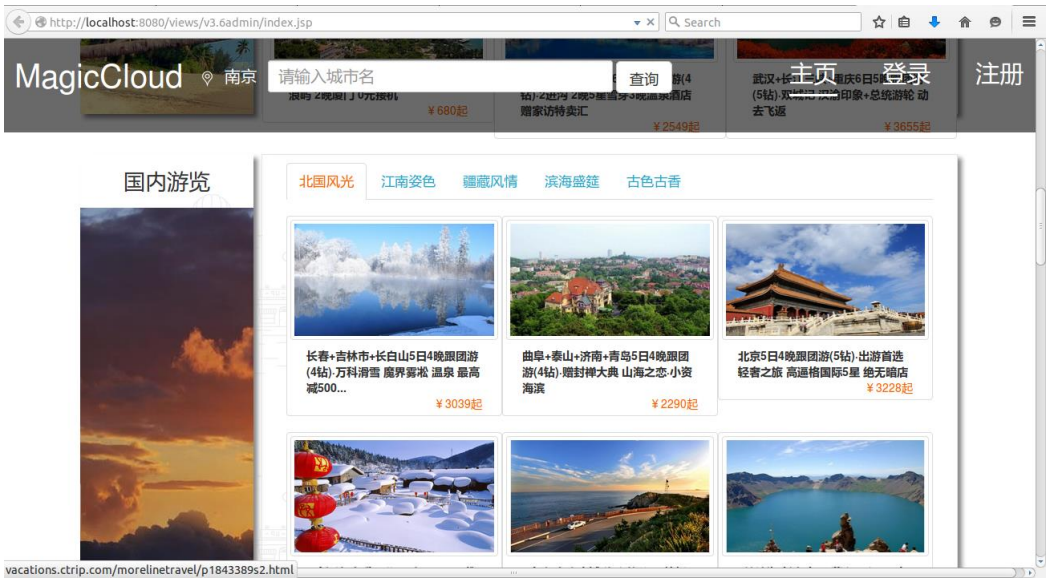


图 7-5 国内游览 1

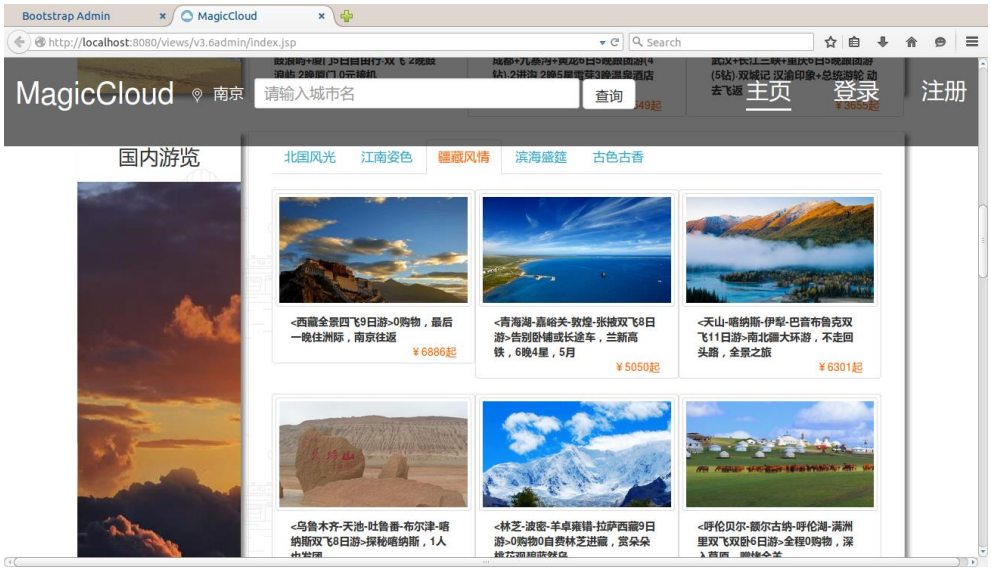


图 7-6 国内游览 2

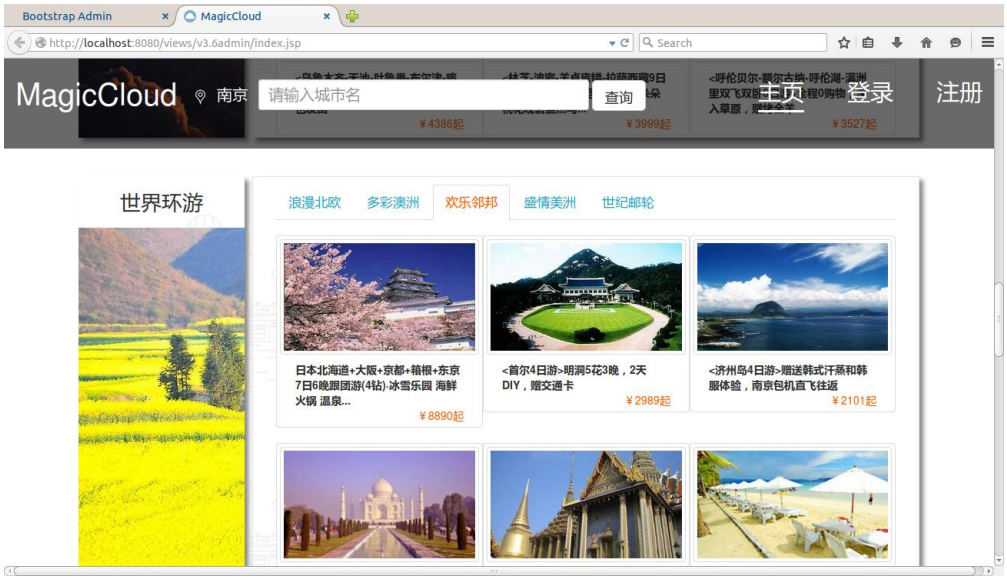


图 7-7 世界环游 1

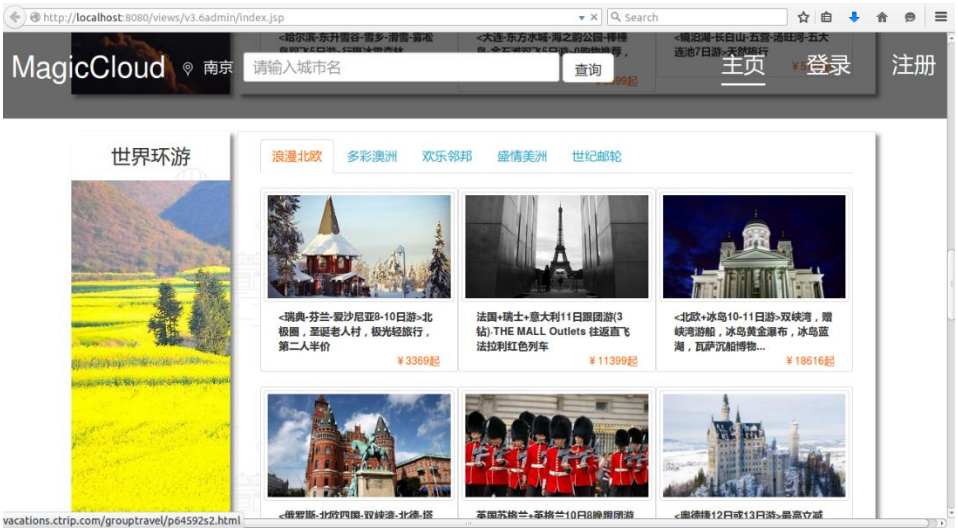


图 7-8 世界环游 2

图 7-9 身边美景

图 7-10 旅途感悟

7.2.3.5 线路产品搜索

在首页选择出发城市和到达城市，点击“查询”按钮，如图 7-11 所示，即可到达所有结果页面，如图 7-12 所示：

图 7-11 搜索 1

图 7-12 搜索 2

7.2.3.6 用户个人中心

用户可以查看个人信息，如图 7-13 所示；可以修改个人相关信息，如图 7-14 所示；可修改用户密码，如图 7-15 所示；可以查看浏览记录，如图 7-16 所示。

图 7-13 查看个人信息

图 7-14 修改个人信息

图 7-15 修改用户密码

图 7-16 查看浏览记录

八、深入分析与创新需求

8.1 深入分析

8.1.1 安全性探讨

本系统的数据库中会保存用户的用户名和密码，因此，若数据库中数据被非法查询的话，用户的账户信息就会被非法掌握，这就会造成安全隐患。为了提高安全性，系统可以使用 MD5 码进行加密，数据库中存储的是密码的 MD5 码，同时因为 MD5 码的不可能被反计算的特性，即使数据库被非法查询，得到的也只是一串 MD5 码，不可能威胁到用户的账户安全。而我们的系统进行用户验证的时候，只要计算用户输入的密码的 MD5 码，判断与数据库中存储的 MD5 是否匹配，若匹配则认为密码正确，否则认为密码错误。通过这种方式提高了系统的安全性。

8.2 创新需求与解决方案

8.2.1 基于地图的旅游景点介绍

本系统实现了最基本的旅游产品比价功能，为用户提供个性化的推荐服务。在此基础上，我们设想可以实现基于地图的景点介绍功能。

例如用户希望出行、旅游，但是又没有具体的目的地，这时候我们可以提供一张全国甚至全球的地图，在所有具有知名旅游景点的城市设置标记，用户在地图上任意点击这些标记，点击之

后就会弹出一个对该城市知名景点的简介，可以包括图片和他人的评论，这些信息作为用户选择旅游产品的参考信息，这样使得用户的旅游体验更好，帮助用户更快、更便捷地找到适合自己的旅游产品。

该功能的实现可以使用百度地图，百度地图的官方提供了一整套功能非常完备和强大的 API，借助这组 API，可以实现地图的展示以及覆盖物的添加等一系列功能，支持基于地图的旅游景点介绍功能的实现。基本用户操作如下：用户点击打开地图展示页面，在浏览的同时点击地图上的各个标记点，查看相关信息。

九、总结

本平台完成了预定的功能性需求和非功能性需求，完成了命题中的全部需求，开发了一个完善的旅游推荐系统，成功完成了所有任务。

在平台的设计和实现过程中，我们充分尊重用户的需求以及设身处地地为用户着想，为了简化用户的操作和降低使用难度，为用户提供了人性化的操作界面，便捷的操作接口。同时在系统的性能上做了较深的研究，充分保证了系统性能优秀。

如果继续做下一步的研究，可以考虑在现有系统的基础上探寻更多提高性能的方法，追寻系统的进一步优化，同时思考更多、更好、更适合的推荐方式，为广大用户提供一个更加个性化、更加友好并且更加智能化的旅游比价推荐系统，将该系统持续发展下去，利用当今发展得如火如荼的信息化的技术为用户们提供舒适、贴心、便捷的服务。