“睡吧”旅店系统

**详细设计报告**

**版本：1.0**

目录

[1引言](#_Toc23694_WPSOffice_Level1) [3](#_Toc23694_WPSOffice_Level1)

[1.1 标识](#_Toc3374_WPSOffice_Level1) [3](#_Toc3374_WPSOffice_Level1)

[1.2系统概述](#_Toc4872_WPSOffice_Level1) [3](#_Toc4872_WPSOffice_Level1)

[1.3 文档概述](#_Toc3464_WPSOffice_Level1) [4](#_Toc3464_WPSOffice_Level1)

[1.4功能需求](#_Toc18126_WPSOffice_Level1) [4](#_Toc18126_WPSOffice_Level1)

[1.5 基线](#_Toc31198_WPSOffice_Level1) [4](#_Toc31198_WPSOffice_Level1)

[2引用文件](#_Toc30840_WPSOffice_Level1) [5](#_Toc30840_WPSOffice_Level1)

[1.中华人民共和国国家标准GB T-8567-2006](#_Toc22527_WPSOffice_Level1) [5](#_Toc22527_WPSOffice_Level1)

[3详细系统设计](#_Toc32601_WPSOffice_Level1) [5](#_Toc32601_WPSOffice_Level1)

[3.1账户信息管理](#_Toc28585_WPSOffice_Level1) [5](#_Toc28585_WPSOffice_Level1)

[3.2客房信息管理](#_Toc781_WPSOffice_Level1) [5](#_Toc781_WPSOffice_Level1)

[3.3绩效及收益管理](#_Toc5530_WPSOffice_Level1) [6](#_Toc5530_WPSOffice_Level1)

[4数据描述](#_Toc19129_WPSOffice_Level1) [6](#_Toc19129_WPSOffice_Level1)

[1）房间信息表](#_Toc11316_WPSOffice_Level1) [6](#_Toc11316_WPSOffice_Level1)

[2）客户信息表](#_Toc28590_WPSOffice_Level1) [6](#_Toc28590_WPSOffice_Level1)

[3） 员工绩效表](#_Toc28214_WPSOffice_Level1) [7](#_Toc28214_WPSOffice_Level1)

[4）员工信息表](#_Toc21928_WPSOffice_Level1) [7](#_Toc21928_WPSOffice_Level1)

[5程序设计说明](#_Toc31840_WPSOffice_Level1) [7](#_Toc31840_WPSOffice_Level1)

[5.1程序描述](#_Toc14122_WPSOffice_Level1) [9](#_Toc14122_WPSOffice_Level1)

[5.2 功能](#_Toc22736_WPSOffice_Level1) [13](#_Toc22736_WPSOffice_Level1)

[5.3性能](#_Toc27527_WPSOffice_Level1) [13](#_Toc27527_WPSOffice_Level1)

[5.4输入项](#_Toc28441_WPSOffice_Level1) [13](#_Toc28441_WPSOffice_Level1)

[5.5输出项](#_Toc2356_WPSOffice_Level1) [14](#_Toc2356_WPSOffice_Level1)

[5.6算法](#_Toc1184_WPSOffice_Level1) [14](#_Toc1184_WPSOffice_Level1)

[5.7流程逻辑](#_Toc19600_WPSOffice_Level1) [14](#_Toc19600_WPSOffice_Level1)

[6接口](#_Toc16345_WPSOffice_Level1) [15](#_Toc16345_WPSOffice_Level1)

[7注释设计](#_Toc22437_WPSOffice_Level1) [15](#_Toc22437_WPSOffice_Level1)

[a． 加在模块首部的注释；](#_Toc633_WPSOffice_Level1) [15](#_Toc633_WPSOffice_Level1)

[b． 加在各分枝点处的注释；](#_Toc5574_WPSOffice_Level1) [15](#_Toc5574_WPSOffice_Level1)

[c． 对各变量的功能、范围、缺省条件等所加的注释；](#_Toc8500_WPSOffice_Level1) [15](#_Toc8500_WPSOffice_Level1)

[d． 对使用的逻辑所加的注释等等。](#_Toc27862_WPSOffice_Level1) [15](#_Toc27862_WPSOffice_Level1)

[8限制条件](#_Toc18014_WPSOffice_Level1) [16](#_Toc18014_WPSOffice_Level1)

[9测试计划](#_Toc12784_WPSOffice_Level1) [16](#_Toc12784_WPSOffice_Level1)

[9.1 测试目的](#_Toc15978_WPSOffice_Level1) [16](#_Toc15978_WPSOffice_Level1)

[9.2 测试方法](#_Toc17498_WPSOffice_Level1) [16](#_Toc17498_WPSOffice_Level1)

[9.3 测试记录与运行结果](#_Toc9639_WPSOffice_Level1) [16](#_Toc9639_WPSOffice_Level1)

[10尚未解决的问题](#_Toc2263_WPSOffice_Level1) [19](#_Toc2263_WPSOffice_Level1)

[11注解](#_Toc26798_WPSOffice_Level1) [19](#_Toc26798_WPSOffice_Level1)

[11.1定义](#_Toc4320_WPSOffice_Level1) [19](#_Toc4320_WPSOffice_Level1)

[11.2参考资料](#_Toc21844_WPSOffice_Level1) [19](#_Toc21844_WPSOffice_Level1)

[12总结](#_Toc21579_WPSOffice_Level1) [19](#_Toc21579_WPSOffice_Level1)

**1引言**

众所周知，现在的宾馆行业竞争异常激烈，怎样吸引顾客，提高宾馆满员率以及管理水平是十分重要的任务，而借助于当前蓬勃发展的IT工具来进行宾馆的管理既方便又快捷，大大提高了办事效率。而且在数据保存和数据交换等方面也是手工操作所无法比拟的。虽然信息化不是宾馆走向成功的关键因素，但是它可以帮助那些真正影响成败的要素发挥更大的作用！

* 1. **标识**

本系统只适用于windows操作系统下安装及使用。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文件状态：  [ ]草稿  [ √ ]正式发布  [ ]正在修改 | 报告编号 | “睡吧”旅店系统详细设计报告  Hotelmanage-003 | | |
| 当前版本 | V1.0 | | |
| 编写人 | 刘琛 | 编写日期 | 2019-7-4 |
| 审批人 | 03小组 | 审批日期 | 2012-7-4 |
| 保密级别 | …… | | |

**1.2系统概述**

随着我国经济的迅速发展，人们的生活水平有了显著的提高，假日经济和旅游经济已成为人们消费的热点。各地也把旅游业当作本地经济发展的重要经济支柱之一，从而促进了宾馆酒店业的快速发展。同时，随着宾馆酒店越来越多，人们的要求越来越高，宾馆酒店业的竞争也愈来愈激烈。

如何在激烈的竞争中生从发展，是每一个宾馆酒店必须面临的问题。对宾馆的经营状况起决定作用的是宾馆的服务管理水平。如何利用先进的管理手段、提高宾馆的管理水平，以成为宾馆酒店业发展的当务之急。面对信息时代的机遇和挑战，利用科技手段提高企业管理水平是企业管理无疑的一条行之有效的途径。现在的宾馆行业已经发生拉巨大变化，已不仅是提供给人们一个吃住行的简单场所，所以，将计算机管理引入宾馆的管理体系中,无疑对宾馆的服务水平更上一层楼提供了一种有利的支持。

虽然计算机管理并不是宾馆管理走向成功的关键元素，但它可以最大限度地发挥准确、快捷、高效等作用，对宾馆的服务管理提供强有力的支持。因此，采用全新的计算机网络和宾馆管理系统，已成为提高宾馆的管理效率、改善服务的水准的重要手段之一。在某种意义上，宾馆客房管理方面的信息话已成为现代化宾馆的重要标志。

建立一套功能完善的管理信息系统，既能满足业务人员日常处理的需要，增强企业经营全过程的数字化管理水平；又能满足管理人员决策分析的需要，提高公司管理层对公司经营反馈信息的响应速度。从而大幅度提高工作效率，提高实施管理的准确性、科学性，使担负管理的工作人员从繁杂的手工劳作中解脱出来。

面对酒店行业的高速发展和酒店行业信息化发展的过程中出现的各种情况，酒店客房管理系统在实施后，应能达到以下目标：

(1)实现多点操作的信息共享，相互之间的信息传递准确、快捷和顺畅。

(2)服务管理信息化，可随时掌握客人住宿、客房状态、客人挂帐等情况。

(3)系统界面友好美观，操作简单易行，查询灵活方便，数据存储安全。

(4)客户档案、挂帐信息、预警系统的结合，可对往来可户进行住宿监控，防止坏帐发生。

(5)通过酒店客房管理系统的实施，可逐步提高宾馆的管理水平，提升员工的素质。

(6)系统维护方便可靠、有较高的安全性，满足实用性、先进性的要求。

**1.3 文档概述**

本系统程序用于管理客户和旅馆工作人员，安排客户入住及人员工作调度。本系统包括账户信息管理子系统，客房信息管理子系统，绩效及收益管理子系统。该系统包括如下单元：首页，客户功能页面，前台功能页面，经理功能页面，管理员功能页面

**1.4功能需求**

a) 客户账户功能：客户注册、登录、修改密码功能，其中注册时所填客户信息包括用户名、密码、身份证号、邮箱、电话。

b) 查看剩余客房功能：前台和客户可以查询当前剩余客房数目及类型。

c) 预定房间功能：客户可以通过该功能预定宾馆客房

d)营业额查询功能：管理员和经理可以查询当前营业额，可按月检索、以图表显示。

e) 管理人员信息管理功能：对前台和经理信息的增删改查、管理员重置前台，经理密码。

f) 客房信息管理功能：客房信息管理（客房号、客房类型、房间位置、面积、

床位、价格）增删改查。

g）管理人员账户功能：前台，经理，管理员登录、修改密码。

h）打卡功能：前台、经理上下班打卡。

i）前台登记功能：换房登记（记录客户和房间信息）、退房登记（记录客户和房间信息，包括是否有破损，对服务态度及体验进行打分）、入住登记（记录客户和房间信息）、预订登记（记录客户和房间信息，渠道：电话，网络）。

j) 用户账号管理功能：前台可以对客户信息进行增删改查、经理对前台和保洁人员信息查看。

k) 打卡查询功能：管理员查询前台和经理的打卡记录。

l) 智能QA功能：客户可以通过智能QA功能查询宾馆相关问题。

m) 评分查看功能：经理查看客户评分（按操作员或时间检索）。

n) 打分功能：对员工的绩效打分。

o) 确认功能：前台确认保洁人员打扫完毕客房。

**1.5 基线**

1. “睡吧”旅店系统可行性分析报告

2.“睡吧”旅店系统概要设计说明书

3. “睡吧”旅店系统详细设计报告

4. “睡吧”旅店系统需求分析报告

5. “睡吧”旅店系统软件测试计划

**2引用文件**

1.中华人民共和国国家标准GB T-8567-2006

**3详细系统设计**

本系统包括账户信息管理子系统，客房信息管理子系统，绩效及收益管理子系统。如图2-1所示

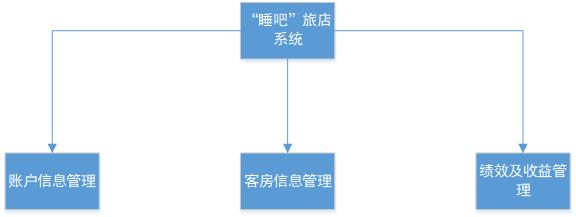


图2－1 “睡吧”旅店系统功能图

**3.1账户信息管理**

客户账户功能：客户注册、登录、修改密码功能，其中注册时所填客户信息包括用户名、密码、身份证号、邮箱、电话。

用户账号管理功能：前台可以对客户信息进行增删改查、经理对前台和保洁人员信息查看。

管理人员信息管理功能：对前台和经理信息的增删改查、管理员重置前台，经理密码。

管理人员账户功能：前台，经理，管理员登录、修改密码。

打卡功能：前台、经理上下班打卡。

**3.2客房信息管理**

前台登记功能：换房登记（记录客户和房间信息）、退房登记（记录客户和房间信息，包括是否有破损，对服务态度及体验进行打分）、入住登记（记录客户和房间信息）、预订登记（记录客户和房间信息，渠道：电话，网络）。

客房信息管理功能：客房信息管理（客房号、客房类型、房间位置、面积、床位、价格）增删改查。

查看剩余客房功能：前台和客户可以查询当前剩余客房数目及类型。

预定房间功能：客户可以通过该功能预定宾馆客房.

智能QA功能：客户可以通过智能QA功能查询宾馆相关问题。

**3.3绩效及收益管理**

打卡查询功能：管理员查询前台和经理的打卡记录。

评分查看功能：经理查看客户评分（按操作员或时间检索）。

打分功能：对员工的绩效打分。

营业额查询功能：管理员和经理可以查询当前营业额，可按月检索、以图表显示。

**4数据描述**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 管理员账号 | | | | | |
| 字段名 | 数据类型 | 默认值 | 允许空值 | 自动递增 | 备注 |
| Username | Varchar(20) |  | 否 | 是 | 唯一 |
| password | Varchar(20) |  | 否 |  |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 员工每日绩效 | | | | | | | | | | |
| 字段名 | 数据类型 | | | 默认值 | | 允许空值 | | 自动递增 | 备注 | |
| judge\_no | int | | |  | | 否 | | 是 | 唯一 | |
| employee\_id | int | | |  | | 否 | |  |  | |
| perfermance | double | | |  | | 是 | |  | 表现得分 | |
| time | varchar(20) | | |  | | 否 | |  | 评分日期 | |
| 工作人员上班记录 | | | | | | | | | | |
| 字段名 | | 数据类型 | 默认值 | | 允许空值 | | 自动递增 | | | 备注 |
| record\_no | | int |  | | 否 | | 是 | | | 唯一 |
| Staff\_no | | Int |  | | 否 | |  | | |  |
| start\_time | | Varchar(20) |  | | 否 | |  | | | 上班时间 |
| end\_time | | Varchar(20) |  | | 否 | |  | | | 下班时间 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 预订记录 | | | | | |
| 字段名 | 数据类型 | 默认值 | 允许空值 | 自动递增 | 备注 |
| reserve\_no | Int(20) |  | 否 | 是 | 唯一 |
| client\_no | Int(20) |  | 否 |  | 客户id |
| room\_no | Int(20) |  | 否 |  | 房间id |
| orderarrivedate | date |  | 否 |  | 预订到达 |
| orderleavedate | date |  | 否 |  | 预订离开 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 房间表 | | | | | |
| 字段名 | 数据类型 | 默认值 | 允许非空 | 自动递增 | 备注 |
| room\_no | int(3) |  | 否 | 是 | 唯一 |
| room\_id | varchar(20) |  | 否 |  | 房间号，可以反映出房间的位置 |
| room\_type | Varchar(20) |  | 否 |  | 存储了房间的类型，目前定的类型有doubleroom.singleroom,bigbedroom |
| room\_state | Varchar(20) |  | 否 |  | 第一位指明了房间的状态：1，checkedin 已入住 2，unavialiable 正在打扫 3.avialiable 打扫完毕，可入住 |
| room\_price | Double(20) |  | 否 |  | 房间价格 |
| room\_location | Varchar(20) |  | 否 |  | 房间楼层 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 客户信息表 | | | | | |
| 字段名 | 数据类型 | 默认值 | 允许非空 | 自动递增 | 备注 |
| Client\_no | Int(20) |  | 否 | 是 | 唯一 |
| username | varchar(20) |  | 否 |  | 唯一 |
| password | varchar(20) |  | 否 |  |  |
| Idcard | char(18) |  | 否 |  | 唯一 |
| Email | varchar(100) |  | 否 |  |  |
| Telephone | char(11) |  | 否 |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 入住登记表 | | | | | |
| 字段名 | 数据类型 | 默认值 | 允许非空 | 自动递增 | 备注 |
| checkin\_no | Int(20) |  | 否 | 是 |  |
| Room\_no | Varchar(20) |  | 否 |  | 房间号 |
| Client\_no | Varchar(20) |  | 否 |  | 身份证 |
| leavedate | Date |  | 否 |  | 离店日期 |
| Arrivedate | Date |  | 否 |  | 到店日期 |
| isdamaged | Varchar(20) |  | 否 |  | 是否损坏  Yes or no |
| Exp\_score | Varchar(20) |  | 是 |  | 住宿评分 |
| Ser\_score | Varchar(20) |  | 是 |  | 服务评分 |
| cost | Double(20,2) |  | 否 |  | 消费合计 |
| Operator\_id | Varchar(20) |  | 否 |  | 操作员no |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 换房登记 | | | | | |
| 字段名 | 数据类型 | 默认值 | 允许非空 | 自动递增 | 备注 |
| Change\_no | Int(20) |  | 否 | 是 | 唯一 |
| Oldroomid | varchar(20) |  | 否 |  | 旧房间号码 |
| newroomid | varchar(20) |  | 否 |  | 新房间号码 |
| Client\_no | Varchar(20) |  | 否 |  | 客户编号 |

**5程序设计说明**

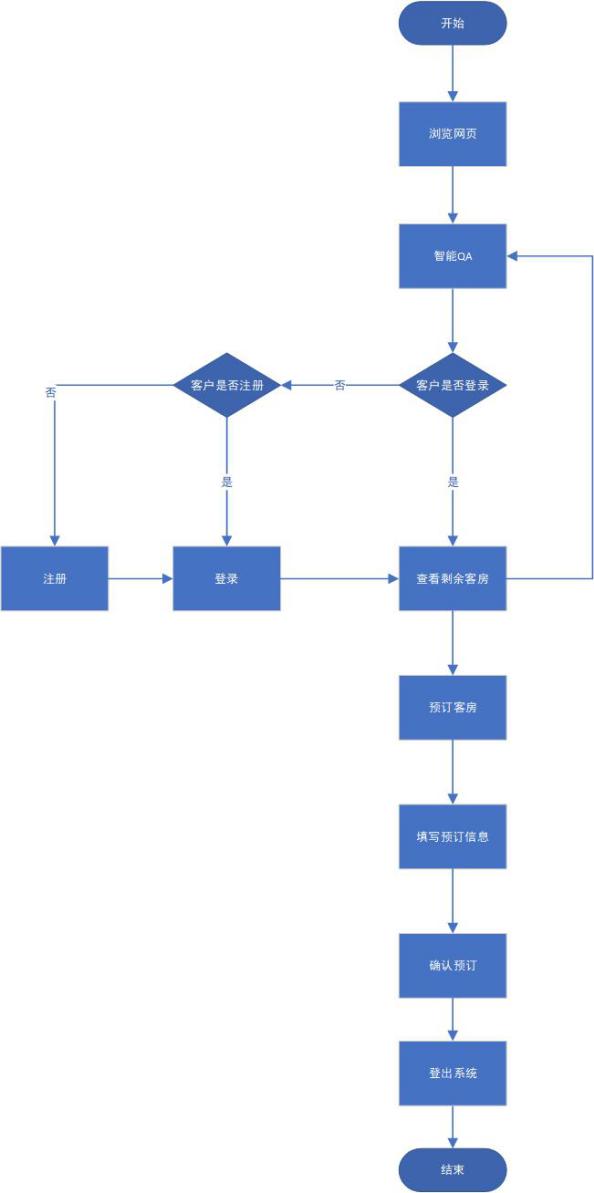
本系统程序用于管理客户和旅馆工作人员，安排客户入住及人员工作调度。本系统包括账户信息管理子系统，客房信息管理子系统，绩效及收益管理子系统。该系统包括如下单元：首页，客户功能页面，前台功能页面，经理功能页面，管理员功能页面

程序描述

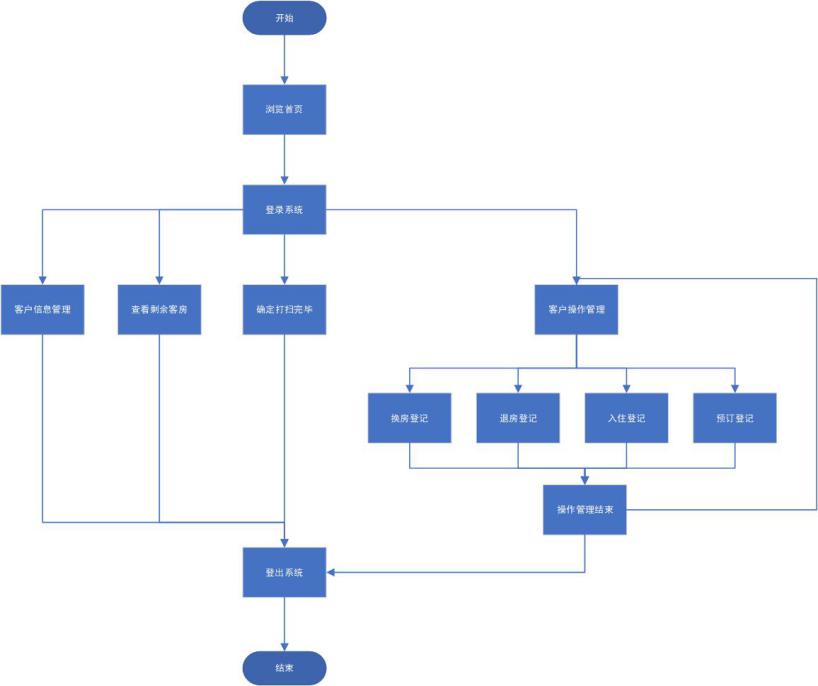
客户通过浏览剩余客房信息，进行预订客房操作；前台人员进行客户，客房管理操作；经理对员工绩效评分，确认收益。

流程图示：

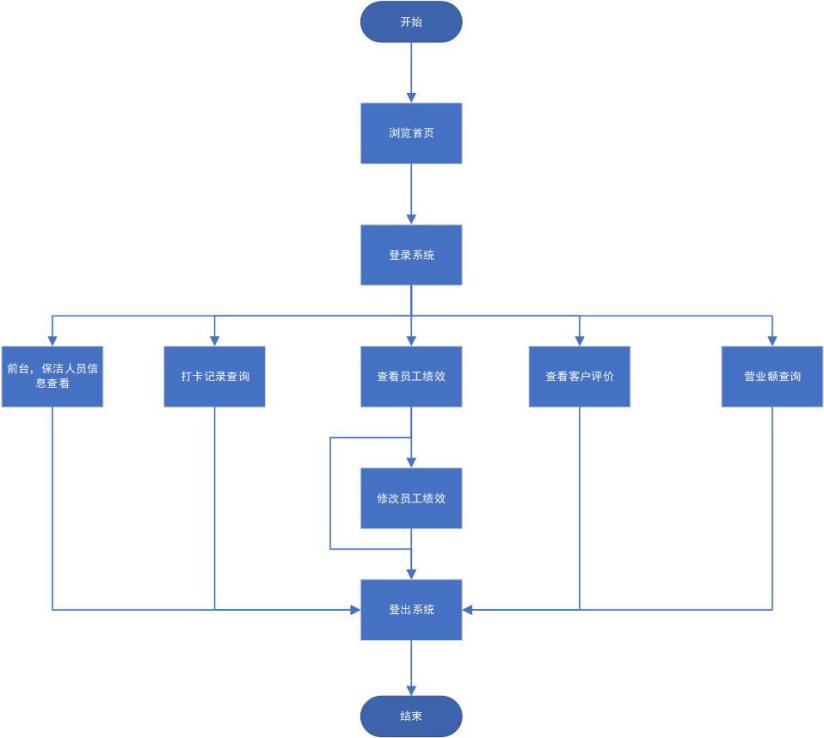
客户：



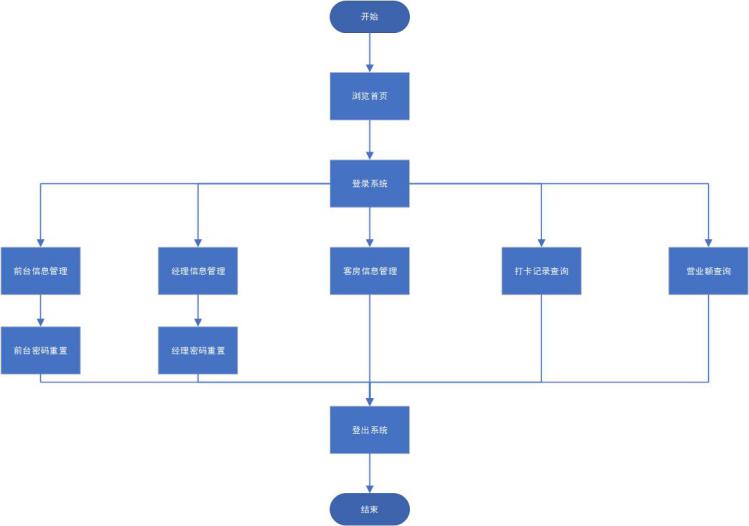
前台：



经理：



管理员：



**5.1程序描述**

（1）连接数据库模块代码

package DAO;  
  
/\*\*  
 \*  
 \*/  
  
  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Statement;  
  
/\*\*  
 \* @author bfs  
 \*  
 \*/  
public class DBHelper {  
  
    public String url = "jdbc:mysql://localhost:3306/hotel";  
    public String username = "root";  
    public String password = "123456";  
    public static DBHelper instance = null;  
  
    // 通过静态代码块注册数据库驱动，保证注册只执行一次  
    static {  
        try {  
            Class.forName("com.mysql.jdbc.Driver").newInstance();  
        } catch (InstantiationException | IllegalAccessException  
                | ClassNotFoundException e) {  
            e.printStackTrace();  
        }  
    }  
  
    private DBHelper() {  
    }  
  
    /\*\*  
     \* 获得类实例  
     \*  
     \* @return  
     \*/  
    public static DBHelper getInstance() {  
        // 给类加锁,防止线程并发  
        synchronized (DBHelper.class) {  
            if (instance == null) {  
                instance = new DBHelper();  
            }  
        }  
        return instance;  
    }  
  
    /\*\*  
     \* 获得连接  
     \*  
     \* @return  
     \* @throws SQLException  
     \*/  
    public Connection getConnection() throws SQLException {  
        return DriverManager.getConnection(url, username, password);  
    }  
  
    /\*\*  
     \* 关闭连接  
     \*  
     \* @param conn  
     \* @param st  
     \* @param rs  
     \*/  
    public static void closeConnection(Connection conn, Statement st, ResultSet rs) {  
        try {  
            if (rs != null) {  
                rs.close();  
            }  
        } catch (SQLException e) {  
            e.printStackTrace();  
        } finally {  
            try {  
                if (st != null) {  
                    st.close();  
                }  
            } catch (SQLException e) {  
                e.printStackTrace();  
            } finally {  
                try {  
                    if (conn != null) {  
                        conn.close();  
                    }  
                } catch (SQLException e) {  
                    e.printStackTrace();  
                }  
            }  
        }  
    }  
}

（2）用户登录界面代码

using System;

using System.Configuration;

using System.Data;

using System.Linq;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.HtmlControls;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Xml.Linq;

using System.Data.SqlClient;

using System.Collections;

public partial class \_Default : System.Web.UI.Page

{

protected void Page\_Load(object sender， EventArgs e)

{

}

protected void ImageButton1\_Click(object sender， ImageClickEventArgs e)

{

String sqlStr = "select \* from UserInfo where UserName='"+UserName.Text+"' and UserPwd='"+UserPwd.Text+"'";

DataSet ds = DB.reDs(sqlStr);

if (ds.Tables[0].Rows.Count != 0)

{

Session["UserID"] = ds.Tables[0].Rows[0][0].ToString();

Session["UserName"] = UserName.Text;

Response.Redirect("GPSContent/home\_page.aspx");

}

else

Response.Write("<script language=javascript>alert('登录失败！');</script>");

}

protected void ImageButton2\_Click(object sender， ImageClickEventArgs e)

{

Response.Redirect("ZhuCe.aspx");

}

}

运行结果如下图2-6所示。





图2-6 “睡吧”旅店系统登录界面图

**5.2 功能**

初始化模块：

* + 功能描述：系统初始时，由操作系统调用，经过身份验证，进入不同的模块。
  + 输入项：用户名：任意长度字符串或数字；密码：任意长度字符串或数字。
  + 输出项：有效用户和无效用户。

**5.3性能**

说明对该程序的全部性能要求，包括对精度、灵活性和时间特性的要求。

**5.4输入项**

给出对每一个输入项的特性，包括名称、标识、数据的类型和格式、数据值的有效范围、输入的方式。数量和频度、输入媒体、输入数据的来源和安全保密条件等等。表1给出了“睡吧”旅店系统登录界面输入项表。

表1“睡吧”旅店系统登录输入项表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 输入方式 | 输入媒介 | 输入来源 |
| Content | Content | Varchar | 用户输入 | 键盘 | TextBox1 |
| Title | Title | Varchar | 用户输入 | 键盘 | TextBox2 |

**5.5输出项**

给出对每一个输出项的特性，包括名称、标识、数据的类型和格式，数据值的有效范围，输出的形式、数量和频度，输出媒体、对输出图形及符号的说明、安全保密条件等等。表2给出了“睡吧”旅店系统登录界面输出项表。

表2“睡吧”旅店系统登录输出项表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 标识 | 数据类型 | 输出方式 | 输出媒介 |
| Content | Content | Varchar | 写入数据库 | 数据库 |
| Title | Title | Varchar | 写入数据库 | 数据库 |
| UserName | UserName | Varchar | 写入数据库 | 数据库 |
| Time | Time | DateTime | 写入数据库 | 数据库 |

**5.6算法**

详细说明本程序所选用的算法，具体的计算公式和计算步骤。

**5.7流程逻辑**

“睡吧”旅店系统的流程逻辑图如图5－4所示。

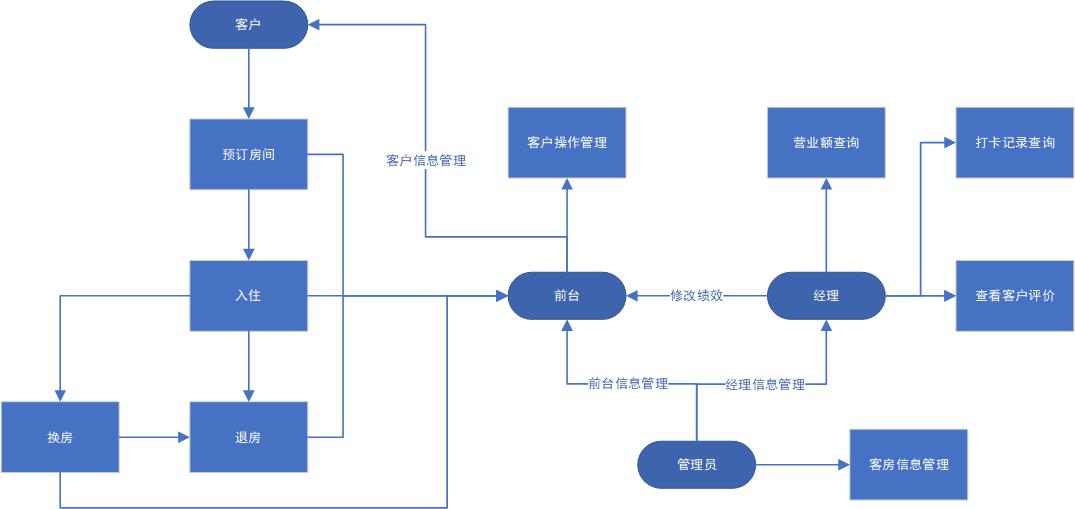


图5－4 睡吧”旅店系统的流程逻辑图

**6接口**

**BaseDAO{**

**private void getConnection()**

**public ResultSet executeQuery(String query, List<Object> params)**

**public int executeUpdate(String query, List<Object> params)**

**public void close()**

**}**

**CheckinDAO{**

**public int addCheckin(Checkin checkin)**

**public int updateLeaveDate(Checkin checkin, int duration)**

**public int addCheckout(Checkin checkout)**

**public int updateCost(Checkin checkout)**

**public int updateRoomState\_arrive(String room\_no)**

**public int updateRoomState\_leave(String room\_no)**

**public int updateRoomState\_clean(String room\_no)**

**public ResultSet SearchScoreByOpr(String operator\_name)**

**public ResultSet SearchReserveById(String id,String arrive\_date)**

**public int ChangeRoom(String or\_room\_id,String to\_room\_id,String client\_no)**

**public int updateChangeRoom(String to\_room\_no, String client\_no)**

**public ResultSet searchReserveByNo(int room\_no)**

**public ResultSet searchRoomByType(String roomType)**

**public Room SearchRoomByNo(int room\_no) throws SQLException**

**public List<Room> searchRoomAvailable(String roomType,String arr\_dateStr,String lea\_dateStr)**

**public boolean is\_solt(Date arr\_a, Date lea\_a, Date arr\_b, Date lea\_b)**

**public Date strToDate(String str)**

**public boolean compare\_date(Date date1, Date date2)**

**}**

**ClientDAO{**

**public ResultSet showClient()**

**public ResultSet searchClientById(String id)**

**public int searchClientByEmail(String id)**

**public int addClient(Client client)**

**public static void deleteClient(String email)**

**public static void changePassword(String email)**

**public int searchClientByEmailTrue(String email)**

**}**

**DBHelper{**

**public static DBHelper getInstance()**

**public Connection getConnection() throws SQLException**

**public static void closeConnection(Connection conn, Statement st, ResultSet rs)**

**}**

**EmployeeDao{**

**public static List<Employee> ShowEmployee()**

**public static int addEmployee(String username,String password,String email,String telephone,String position)**

**public static boolean deleteEmployee(int employee\_no)**

**public static boolean updateEmployee(int employee\_no,String username,String password,String email,String telephone,String position)**

**public static boolean resetPassword(int employee\_no)**

**public static List<Employee> DisplayEmployee()**

**}**

**IncomeDAO{**

**public static List<Income> ShowIncome()**

**public static double doubleRoomIncome(String year,String month)**

**public static double singleRoomIncome(String year,String month)**

**public static double bigBedRoomIncome(String year,String month)**

**}**

**LoginDAO{**

**public static boolean existEmployee(String email)**

**public static boolean existmanager(String email)**

**public static String getPasswrod(String email)**

**public static String getEPasswrod(String email)**

**public static String getName(String email)**

**public int addClient(Client client)**

**public static String getEName(String email)**

**public static String getEno(String email)**

**}**

**QueryDAO{**

**public ArrayList<Integer> getRoomRemain(String queryDate)**

**public boolean checkCheckin(String queryDate, String roomNo)**

**public boolean checkReserve(String queryDate, String roomNo)**

**}**

**ReserveDAO{**

**public ResultSet researchRe()**

**public ResultSet searchRoomByType(String roomType)**

**public ResultSet searchReserveByNo(int room\_no)**

**public int reserveRoom(Reserve reserve)**

**}**

**RoomDAO{**

**public static String addroom(String id, String price, String type, String location)**

**public static String editroom(String id, String price, String type, String location, String no)**

**public static String deleteroom(String no)**

**public static String verpassword(String id,String password)**

**}**

**WorkRecordDAO{**

**public static List<WorkRecord> getrecords(String staff\_no)**

**public static List<WorkRecord> getrecords2(String year, String month, String day)**

**public static String getstaffno(String username)**

**public static String getemployeename(String staff\_no)**

**public static int addStartrecord(String staff\_no)**

**public static int changeEndrecord(String staff\_no)**

**}**

**7注释设计**

说明准备在本程序中安排的注释，如：

1. 加在模块首部的注释；
2. 加在各分枝点处的注释；
3. 对各变量的功能、范围、缺省条件等所加的注释；
4. 对使用的逻辑所加的注释等等。

**8限制条件**

说明本程序运行中所受到的限制条件。

**9测试计划**

**9.1 测试目的**

在系统开发过程中存在许多复杂的问题，为了保证系统在投入运行中；保证软件的质量和运行的安全行，测试过程是必不可少的，同时也是关键的。测试是为了发现软件在运行过程中所出现的问题，并在出现问题时能够及时的做出相应的修改。测试的目标是保证系统在投入运行中的安全性和稳定性。

**9.2 测试方法**

本系统采用黑盒测试，测试方法是测试过程中至关重要的环节，采用的测试方法不同所产生的效果和所用的测试经费也是不同的。一个好的测试方法能够提高测试的效率，缩短测试的时间，也能够减少测试的经费。

**9.3 测试记录与运行结果**

**9.3.1经理模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 功能模块 | 操作 | 预期结果 | 判定 |
| 登陆系统 | 输入账号密码，点击登陆 | 输入正确账号密码登陆成功，输入错误的账号密码，无法登陆。 | 通过 |
| 修改密码 | 点击修改密码，输入新密码 | 重新登陆，输入新密码，登陆成功，输入旧密码登陆失败 |  |
| 打卡上下班 | 登陆账号，点击上班打卡，点击下班打卡 | 数据库中，该经理工作天数增加一天 |  |
| 营业额查询 | 经理登陆账号，点击营业额查询 | 按照月，日显示营业额 |  |
| 对员工绩效打分 | 登陆账号，点击绩效打分，对员工绩效打分 | 数据库中，更新员工的绩效数据 |  |
| 查看客户评分 | 登陆系统，点击客户评分 | 显示所有客户的评分结果 |  |

**9.3.2前台模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 登陆系统 | 通过管理员提供的账号密码登陆系统 | 输入正确账号密码可以成功登陆，输入错误的账号密码无法成功登陆 |  |
| 修改密码 | 登陆系统后，点击修改密码，重新设置个人密码 | 修改密码成功后，用新的账号密码可以成功登陆，输入旧密码无法登陆 |  |
| 客户信息增删改查 | 登陆系统后，可以注册一个客户，删除一个客户，修改客户信息 | 注册客户，数据库中将增加一个客户，删除客户，数据库中将减少一个客户，修改信息后，客户信息被更新 |  |
| 查看剩余客房 | 点击查看剩余客房 | 显示剩余房间 |  |
| 打卡上下班 | 登陆账号，点击上班打卡，点击下班打卡 | 数据库中，该前台工作天数增加一天 |  |
| 入住登记 | 输入客户信息，点击入住 | 该房间状态被修改为已入住，可以查看预定时间 |  |
| 换房登记 | 点击换房登记，输入原房间，新房间 | 数据库中新房间状态被改为已入住 |  |
| 退房登记 | 点击退房登记，输入离开时间 | 数据库中，增添退房时间，花费等数据 |  |
| 预定登记 | 点击预定登记，输入电话等客户信息 | 数据库中添加预定信息 |  |
| 确定打扫完毕 | 点击确定打扫完毕 | 该房间状态被重置为可预订 |  |

**9.3.3管理员模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 功能模块 | 操作 | 预期结果 | 判定 |
| 登陆系统 | 通过数据库中账号密码，登陆系统 | 输入正确账号密码登陆成功，输入错误的账号密码，无法登陆。 | 通过 |
| 修改密码 | 点击修改密码，输入新密码 | 重新登陆，输入新密码，登陆成功，输入旧密码登陆失败 |  |
| 注册新经理，新前台 | 点击注册经理，前台，输入账号密码，点击注册 | 数据库中，成功添加新经理和前台，新经理和前台都可以成功登陆 |  |
| 员工信息修改 | 管理员登陆系统，点击员工信息，增删改员工信息 | 数据库中，修改信息被更新 |  |
| 房间信息修改 | 管理员登陆系统，点击修改房间信息，可以增加，删除，修改房间信息 | 数据库中，房间信息被修改更新 |  |
| 打卡记录查询 | 登陆系统，点击打卡记录查询 | 显示所有员工的打卡记录 |  |
| 营业额查询 | 管理员登陆账号，点击营业额查询 | 按照月，日显示营业额 |  |

**9.3.4客户模块**

|  |  |  |  |
| --- | --- | --- | --- |
| 注册账号 | 点击注册账号，输入账号，密码，点击注册成功 | 注册成功的账号可以进行登陆 |  |
| 登陆系统 | 通过数据库中账号密码，登陆系统 | 输入正确账号密码登陆成功，输入错误的账号密码，无法登陆。 | 通过 |
| 修改密码 | 点击修改密码，输入新密码 | 重新登陆，输入新密码，登陆成功，输入旧密码登陆失败 |  |
| 查看剩余房间 | 客户点击查询房间 | 显示剩余的房间类型，房间数，价格等房间信息 |  |
| 预定房间 | 客户登录系统后，可以点击预定房间，输入入住时间等个人信息 | 预定成功，数据库中增加该房间的预定信息 |  |
| 申请房间打扫 | 客户登录系统，点击申请房间打扫 | 前台可以从数据库中看到该申请 |  |
| 评分 | 登陆系统，可以对旅店进行评分 | 数据库中更新评分结果 |  |

**10尚未解决的问题**

暂未在本程序的设计中尚未解决而设计者认为在软件完成之前应解决的问题。

**11注解**

**11.1定义**

Hotelmanage：本“睡吧”旅店管理系统的简称。

**11.2参考资料**

[1] 萨师煊,王珊 .数据库系统概论.北京：高等教育出版社，1998年.

[2] 张海藩.软件工程导论（第三版）.北京：清华大学出版社，1998年.

[3] 刘志铭,高春燕 .数据库开发实例解析VB.北京：机械工业出版社，1999年

[4] 龚沛曾,陆慰民.VB程序设计简明教程（第二版）.北京：高等教育出版社，2003年.

[5] 张国有.软件工程上机指导书.太原：太原科技大学印刷厂，2007年.

[6] 白尚旺.PowerDesigner数据库建模技术.西安：西安电子科技大学出版社，1999年.

**12总结**

旅店管理系统需要的微机数量和规模都不必太大，但在系统设计时应考虑到酒店的发展需求，在选择硬件设备、服务器操作系统、数据库时都考虑到能够逐步的增加和扩展。