|  |
| --- |
| 哈尔滨工业大学（威海） |
| 详细设计报告 |
| 房间预订模块 |
|  |
| 陈豪 |
| **2015/5/17** |

|  |
| --- |
|  |

[1. 引言 2](#_Toc419755871)

[1.1编写目的 2](#_Toc419755872)

[1.2项目背景 2](#_Toc419755873)

[2.总体设计 2](#_Toc419755874)

[2.1需求概述 2](#_Toc419755875)

[2.2软件结构 3](#_Toc419755876)

[3.程序描述 3](#_Toc419755877)

[3.1功能 3](#_Toc419755878)

[3.2性能 4](#_Toc419755879)

[3.3输入项目 4](#_Toc419755880)

[3.4输出项目 5](#_Toc419755881)

[3.5算法 6](#_Toc419755882)

[3.6程序逻辑（顺序图） 9](#_Toc419755883)

[3.7存储分配 9](#_Toc419755884)

[3.8限制条件 10](#_Toc419755885)

[3.9测试要点 10](#_Toc419755886)

# 引言

## 1.1编写目的

本文档为旅店管理系统中房间预订模块的详细设计报告，为该模块设计的基本情况分析，明确功能的具体实现，模块的具体划分与相互之间的联系，也需要为日后的测试做好具体的准备流程。

## 1.2项目背景

随着国民经济状况的提升，越来越多的人希望在假期中外出旅游，这一方面推动了全国各地旅游业的发展，另一方面也对全国各地旅馆的工作人员提出了更高的要求。为了顺应时代的趋势，也为了减轻旅店前台工作人员的负担，现在开发旅店管理系统的房间预订模块。

# 2.总体设计

## 2.1需求概述

满足客户的需求，实现房间预订的流程。主要功能包括增加预定房间，修改已预订订单的信息，删除订单，查找现有订单，显示全部订单等等。

## 2.2软件结构

显示层

Reserve

Change

Update()

Delete

Update()

Add

Update()

Update()

应用层

永久数据保持层

A

C

C

E

S

S

(

J

D

B

C

)

Update

Reserve\_function

Link()

Update()

Name

Tel

Room\_type

Room\_num

Time

End\_time

Add()

Delete()

Change()

Link()

# 3.程序描述

## 3.1功能

1.增加订单:为用户增加一个新的预定房间，要求原房间为空，且输入合理，最后将新建的订单写入数据库中。

2.删除订单:从数据库中删除订单，要求删除时有再次提示，尽可能避免用户误删信息。

3.修改订单:根据需要，将现在已存在的某张订单中的某一项或多项进行修改，最终信息要写入数据库中。

4.查询订单:从现有的订单中根据输入的已知信息得到全部有关信息。

5.订单显示:将已存在的订单全部罗列在主界面，供前台工作人员检查。

## 3.2性能

整个程序的反应时间在1-2秒左右，只要按照常理实现合理输入，便可以得到预期的结果。

## 3.3输入项目

1.在增加预定功能中，输入项目有：用户名name，联系电话tel，房间的规格room\_type，房间号room\_num，预计抵达日期的年、月、日以及预计离开日期的年、月、日。

2.在查找订单功能中，输入项目有：房间号room\_num，然后根据用户的具体需要，输入项目有：用户名name，联系电话tel，房间的规格room\_type，房间号room\_num，预计抵达日期的年、月、日以及预计离开日期的年、月、日。

3.在删除订单功能中，输入项目有：房间号room\_num。

4.在查找订单功能中，输入项目有：房间号room\_num。

5.在现实报表功能中，不需要输入信息。

## 3.4输出项目

1.在增加预定功能中，输出项目有：房间的规格room\_type，房间号room\_num。

2.在删除订单功能中，输出项目有：用户名name，联系电话tel，房间的规格room\_type，房间号room\_num，预计抵达日期的年、月、日以及预计离开日期的年、月、日。

3.在修改订单功能中，输出项目有：用户名name，联系电话tel，房间的规格room\_type，房间号room\_num，预计抵达日期的年、月、日以及预计离开日期的年、月、日。

4.在查找订单功能中，输出项目有：用户名name，联系电话tel，房间的规格room\_type，房间号room\_num，预计抵达日期的年、月、日以及预计离开日期的年、月、日。

5.在显示报表功能中，输出项目有：用户名name，联系电话tel，房间的规格room\_type，房间号room\_num，预计抵达日期的年、月、日以及预计离开日期的年、月、日。

## 3.5算法

下列代码实现数据库的读取信息：

// 定义数据库驱动程序

String DBDRIVER = "sun.jdbc.odbc.JdbcOdbcDriver";

// 定义数据库连接地址

String DBURL = "jdbc:odbc:hotel";

// 定义数据库连接对象，属于java.sql包中的接口

Connection conn = **null**;

// 定义Statement对象，用于操作数据库

Statement stmt = **null**;

// 定义一字符串变量，用于保存SQL语句

String sql = **null**;

String sql\_other =**null**;

// 定义一个结果集以存放检索的结果

ResultSet r = **null**;

// 1、加载驱动程序

**try** {

Class.*forName*(DBDRIVER);

} **catch** (Exception e) {

// 此处使用out.print是处于演示目的，在实际开发中所有的错误消息，

// 绝对不能够通过System.out.print打印，否则会存在安全问题

System.*out*.println("数据库驱动程序加载失败！！！");

}

// 2、连接数据库

**try** {

conn = DriverManager.*getConnection*(DBURL);

} **catch** (Exception e) {

System.*out*.println("数据库连接失败！！！");

}

// 3、操作数据库

// 通过Connection对象实例化Statement对象

**try** {

stmt = conn.createStatement();

// 为sql变量赋值

// 插入语句

sql = "select \* from reserve";

r = stmt.executeQuery(sql);

**while** (r.next()) {

name.add(r.getString(2));

tel.add(r.getString(3));

room\_type.add(r.getString(4));

room\_num.add(r.getString(5));

yu\_nian.add(r.getString(6));

yu\_yue.add(r.getString(7));

yu\_ri.add(r.getString(8));

end\_nian.add(r.getString(9));

end\_yue.add(r.getString(10));

end\_ri.add(r.getString(11));

}

} **catch** (Exception e) {

System.*out*.println("操作数据库失败！！！");

}

// 4、关闭数据库

**try** {

// 关闭操作

stmt.close();

// 关闭连接

conn.close();

} **catch** (Exception e) {

System.*out*.println("数据库关闭失败！！！");

}

增加新信息到数据库的代码应为：

stmt = conn.createStatement();

// 为sql变量赋值

// 插入语句

//sql = "select \* from reserve";

sql = "insert into reserve values('"+ID+"','"+Name+"','"+Tel+"','"+RoomType+"','"+RoomNum+"','"+Yuding\_nian+"','"+Yuding\_yue+"','"+Yuding\_ri+"','"+baoliu\_nian+"','"+baoliu\_yue+"','"+baoliu\_ri+"')";

stmt.executeUpdate(sql);

sql = "update roominfo set room\_state = '2' Where room\_id = '"+RoomNum+"'";

stmt.executeUpdate(sql);

删除数据库的信息的代码应为：

sql = "update roominfo set room\_state = '0' Where room\_id = '"+RoomNum+"'";

stmt.executeUpdate(sql);

sql = "delete \* from reserve where room\_id = '"+RoomNum+"'";

stmt.executeUpdate(sql);

获得与客房类型相对应的房间号的代码为：

预定规格.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent arg0) {

s1 = 预定规格.getItemAt(预定规格.getSelectedIndex()).toString();

**if**(s1 == "豪华双人间")

{

房间编号.removeAllItems();

**for**(**int** i=0;i<HHS.size();i++)

{

房间编号.addItem(HHS.get(i));

}

}

**else** **if**(s1 == "单人间")

{

房间编号.removeAllItems();

**for**(**int** i=0;i<DR.size();i++)

{

房间编号.addItem(DR.get(i));

}

}

**else** **if**(s1 == "双人间")

{

房间编号.removeAllItems();

**for**(**int** i=0;i<SR.size();i++)

{

房间编号.addItem(SR.get(i));

}

}

}

});

## 3.6程序逻辑（顺序图）

****

## 3.7存储分配

//用来存储房间号

**final** ArrayList <String>room\_num = **new** ArrayList<String>();

//用来存储客户姓名

**final** ArrayList <String>name = **new** ArrayList<String>();

//用来存储客户的联系电话

**final** ArrayList <String>tel = **new** ArrayList<String>();

//用来存储房间类型

**final** ArrayList <String>room\_type = **new** ArrayList<String>();

//用来存储预计抵达时间的年份

**final** ArrayList <String>yu\_nian = **new** ArrayList<String>();

//用来存储预计抵达时间的月份

**final** ArrayList <String>yu\_yue = **new** ArrayList<String>();

//用来存储预计抵达时间的日期

**final** ArrayList <String>yu\_ri = **new** ArrayList<String>();

//用来存储预计离开时间的年份

**final** ArrayList <String>end\_nian = **new** ArrayList<String>();

//用来存储预计离开时间的月份

**final** ArrayList <String>end\_yue = **new** ArrayList<String>();

//用来存储预计离开时间的日期

**final** ArrayList <String>end\_ri = **new** ArrayList<String>();

## 3.8限制条件

该程序主要是面对酒店的前台工作人员设计，所以一切行为都必须经过工作人员这一步骤，所以不方便用户的直接使用。另外，只要是使用本程序的人员都会有管理员权限，不利于数据的安全。

## 3.9测试要点

1.当输入空数据时，是否会写入数据库。

2.日期的输入界面，是否会出现不存在的日期，例如2月存在29日或30日等。

3.删除具体数据时，是否会改变另一个表中的相应状态。

4.是否实现了数据的封装，如数据库中的信息，会不会从外部直接改变等。