**1.1课题背景**

近年来人民群众对住房的刚性需求一直处在上升趋势，政府现在又提倡保障人们群众的刚性住房需求，所以房地产行业还是很有前景的。房地产管理部门面临的工作量日益繁多, 房屋管理中的庞大数据量及更新速度快同传统的手工办理、纸制档案管理模式之间的矛盾已经越来越突出。如何有效地管理庞大复杂的房地产信息资源, 科学地规范房地产的管理工作, 已经成为房地产管理部门提高自身管理以及服务水平的关键所在。如何让房屋变成公司的流动资金，是一个公司竞争中的重中之重。在现在网络飞速发展的背景下，设计房屋销售管理系统来促进销售也是一个很好的手段。当前形势下，仅仅靠原始的手工管理或简单的单机管理是不够的，商品部门面对大量的信息，无法有效率地将其中的重要部分提取出来，并做出相应的判断和处理。公司管理者的决策只能依据报表数据，在浪费大量人力、物力的同时无法做到实时监控，难以保证数据的准确性和及时性。因此，先进的管理思想在商业中实现就成为了一个可望而不可及的目标。公司集团非常急需一套既有先进管理思想又适合国内大小房地产企业的房屋销售管理系统，作为实现目标和提高现有水平的一种重要手段。房屋销售管理系统是一个房屋销售公司不可缺少的部分,它的执行对于公司的决策者和管理者来说是很有帮助的, 随着科学技术的不断提高,计算机科学日渐成熟,其强大的功能已为人们深刻认识,它已进入人类社会的各个领域并发挥着越来越重要的作用。

**1.2编写目的**

通过该实训使学生巩固和深化所学知识，加深对面向对象概念、Java面向对象程序设计思想方法的理解和掌握；并能综合运用所学知识，初步应用软件工程方法进行软件开发；增强学生的动手能力，使学生能够体验到小型项目开发的全过程，培养学生自学及解决问题的能力，增强学生的团队合作及沟通能力。

本文设计的是一个房屋管理系统，具体是通过XAMPP/MySQL知识，文件流和Java语言的程序设计完成的，房屋信息的添加、删除和编辑操作，在管理方面，用户登录后可以更改自己的密码，达到安全的效果，也可以增加和删除用户信息，设置密码。

**2.1问题的提出**

房屋销售在国内外都是一个已经发展了很久的行业，关于房屋销售的各种系统也是多样化的，系统的人性化设计是加强了客户和销售者之间联系，使用户在购买房屋之前就和销售方有了沟通，对自己中意的房屋也很了解。由于我国经济的发展迅速，带动了房地产业的进步，对系统的设计也是向多功能方面发展。"公正、公平、合理"的商业管理理念和商业管理水平的提高，使社会对房屋销售管理系统有了更高的需求；同时由于个人电脑的普及，数据库技术、客户/服务器技术，特别是Internet/Intranet技术的发展，使得第三代房屋销售管理系统的出现成为必然。

面对激烈的竞争和愈发挑剔的购房者，房地产发展商也在不断的寻找能够加强自身竞争力的手段。这时一个比较完善高效的房屋销售系统就越发显得重要了。

**2.2系统的功能分析**

1. 建立顾客类和管理员类，实现对用户个人信息的完善级及修改。
2. 建立房屋类，实现对房屋的编号，地址、面积、代价、房主备注等信息的描述以及查询查看所有房屋。
3. 建立验证码的功能

**2.3需要完成的功能**

（1）能够实现对房屋信息的添加、删除和编辑操作。

（2）对基础设施进行设置。

（3）可以与内部人员进行信息交互。

（4）可以对房屋的成本和报价信息进行编辑。

（5）在管理方面，用户登录后可以更改自己的密码，达到安全的效果，也可以增加和删除用户信息，设置密码。系统还可以提供一定的附加功能来方便用户。

**2.4功能分类**

表2.4.1用户注册、登录、密码修改功能分析

|  |  |
| --- | --- |
| 实现功能 | 用户注册、登录 |
| 对象 | 员工，顾客，管理员 |
| 功能描述 | a.个人通过输入其身份证、姓名、验证码自定义用户名和密码等完善个人信息进行注册并登录；  b.管理员登录后，可以对已注册的员工，顾客，房屋进行管理；  c.员工，顾客，和管理员在登录后，可以对自己的登录密码和个人信息进行修改。 |

**三，设计报告**

本章内容概括了：

①将登录可视化的设计。

②用于定义类的房屋、员工、顾客、用户、房屋卖、员工员的设计。

③用于存储房屋的数据管理、员工管理、顾客管理、用户操作。

④用于创建添加房屋、员工、顾客、值班人员、图书管理、登录、修改用户信息的界面和功能的实体设计。

**3.1 概要设计**

本房屋系统主要三个部分，第一个部分为程序启动类，通过名为Start的类来实现。第二个部分为员工用户房屋管理系统，主要包括查询房屋信息管理，编辑房屋，用户，房屋信息，卖个房屋，修改密码等功能，分别建立Employee、User、Costumer、Product等类来实现这些功能；第三个部分为管理员系统，主要包括管理用户、个人信息，房屋管理，分别建立ViewEmployeeActivity、ViewProductActivity、LoginActivity, 等类来实现这些功能。第三个部分为界面设计系统。

**3.2 详细设计**

**3.2.1系统框架**

项目分为activity, attr, sql三个包，具体项目架构的设计如下图所示

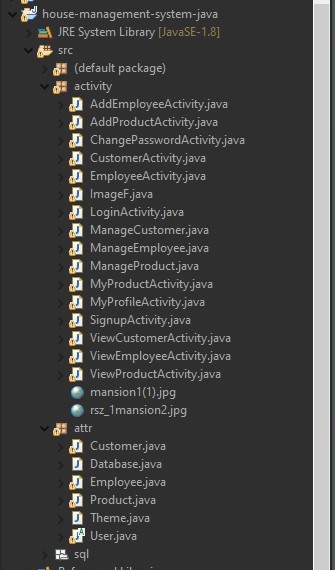


图3.2.1.1构架包

1. 在sql包中创建房屋销售管理系统中：

表3.2.1.1

|  |  |
| --- | --- |
| 文件名 | 说明 |
| f1.sql | XAMPP/MySql 数据库 |

1. 在activity包中创建房屋销售管理系统中的用于定义实体类：

表3.2.1.2

|  |  |
| --- | --- |
| 文件名 | 说明 |
| AddEmployeeActivity.java | 员工添加类模型 |
| AddProductActivity.java | 房屋添加类模型 |
| ChangePassordActivity.java | 修改员工密码类模型 |
| CustomerActivity.java | 查看顾客买信息类模型 |
| EmployeeActivity.java | 查看员工类卖类模型 |
| ImageF.java | 管理房屋图片类模型 |
| LoginActivity.java | 登录类模型 |
| ManageCustomer.java | 管理顾客信息类模型 |
| ManageEmployee.java | 管理员工信息类模型 |
| ManageProduct.java | 管理房屋信息类模型 |
| MyProductActivity.java | 房屋信息类模型 |
| MyProfileActivity.java | 用户信息类模型 |
| SignupActivity.java | 登录类模型 |
| ViewCustomerActivity.java | 演示顾客的卖类模型 |
| ViewEmployeeActivity.java | 演示员工的买类模型 |
| ViewProductActivity.java | 演示房屋的基本信息类模型 |

3．在library.management.file包中创建房屋销售管理系统中的用于保存数据的类：

表3.2.1.3

|  |  |
| --- | --- |
| 文件名 | 说明 |
| Customer.java | 顾客基本类模型 |
| Database.java | 数据库基本类模型 |
| Employee.java | 员工类模型 |
| Product.java | 房屋类模型 |
| Theme.java | 房屋演示标准操作类模型 |
| User.java | 用户操作类模型 |

3.1.2功能图

管理员：

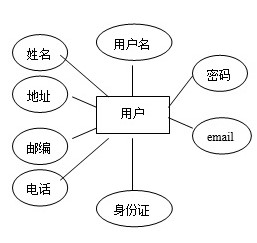


图3.1.2.1管理员

管理员增添： 用户信息修改与删除：

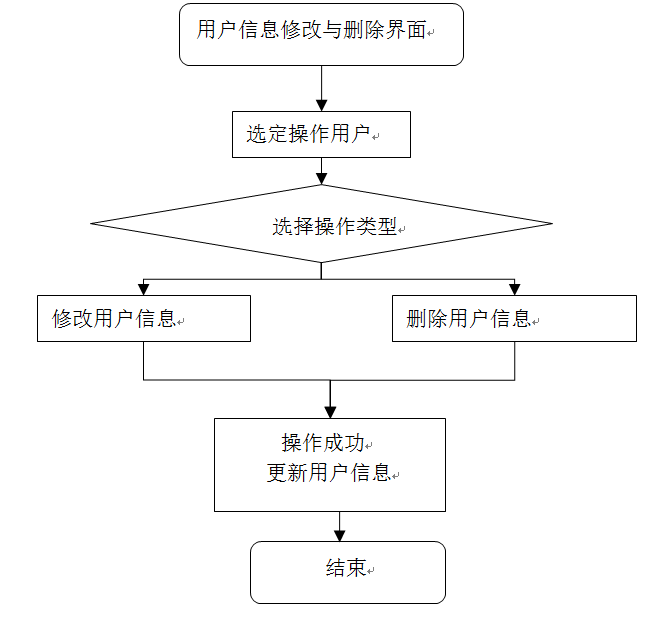
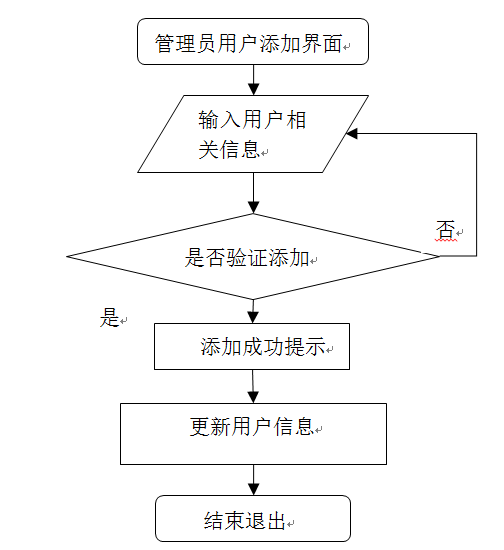


图3.1.2.3 管理员添加 图3.1.2.4用户信息修改与删除

**3.2关键代码与界面**

1.启动类：设置一个启动类Start【public class Start】，将登陆可视化【LoginActivity la = new LoginActivity();

la.setVisible(true);】。

1. 用户注册

当用户登录到系统首页 SignupActivity.java，如果要使用房屋卖功能，那么系统就会通过判断status中的用户帐号是否为空来对使用者身份进行验证判断，如果帐号为空，说明用户没有登录，属于一般浏览用户，所以不能进行买房屋功能，系统就会显示“您没有登录”消息，点击确定跳转到SignupActivity.java注册页面去。

在用户在注册页面填好个人信息后，用户点击注册按钮后，系统首先进行连接数据库操作，使用XAMPP/MySql连接到本地数据库，然后使用Sql声明在打开的数据库中查询，先判断用户注册的用户名已经被别人先注册了，如果这个用户名已经被别人注册了，则显示“注册失败：该用户名已经存在”，这里是用User对象的Write方法来显示。如果这个帐号没有被注册，那么进行insert into插入操作。注册成功后直接跳转到登录页面LoginActivity.java。

3 . 用户登录和管理员登录

用户是通过Login.java页面来登录的。在登录时将用户帐号（userID）和密码（password）输入到各自相应的TEXT控件中去，点击提交后系统把刚才获取的userID和password赋值给新定义的2个字符串 id 和pwd，然后使用SqlConnection连接本地数据库。使用SqlDataReader进行读操作，如果输入的帐号和密码是与数据库中的值是一致的话，那么Status中的值就等于该帐号，登录成功，跳转到首页。如果用户名不存在或输入错误，那么使用Response对象的Write方法显示出提示信息，同时Status中的值为空。在首页中判断用户是否登录就是看Status里的值。如果为空，那么用户为一般浏览用户。如果为注册用户的帐号，那么就可进行注册用户的相应操作。如果其中的值为管理员名，那么该用户为管理员，可以点击管理按钮进入管理页面。管理员的登录和注册用户登录类似，只是实现是在Login.java页面中进行的。页面如图3-2-1所示。

图3-2-1 Login界面

搜索功能实现

用户在User.java中实现搜索操作。使用搜索功能时使用ID， Name， 2个字段联合来查询数据库，然后进行数据库的查询select \*from …where…操作，使用SqlDataAdapter对象来填充DataSet对象，使用SelectCommand来查询记录，最后通过DataGrid将查询的信息显示出来。

部分代码为：

public void fetch{) {

String query = "SELECT `userId`, `employeeName`, `phoneNumber`, `role`, `salary` FROM `employee` WHERE userId='"+this.userId+"';";

Connection con = null;

Statement st = null;

ResultSet rs = null;

System.***out***.println(query);

try {

Class.*forName*("com.mysql.jdbc.Driver");

System.***out***.println("driver loaded");

con = DriverManager.*getConnection*(Database.***HOST\_URI***, Database.***USER***, Database.***PASSWORD***);

System.***out***.println("connection done");//connection with database established

st = con.createStatement();//create statement

System.***out***.println("statement created");

rs = st.executeQuery(query);//getting result

System.***out***.println("results received");

boolean flag = false;

while(rs.next()) {

this.employeeName = rs.getString("employeeName");

this.phoneNumber = rs.getString("phoneNumber");

this.role = rs.getString("role");

this.salary = rs.getDouble("salary");

}

}

catch(Exception ex) {

System.***out***.println("Exception : " +ex.getMessage());

}

finally {

try {

if(rs!=null)

rs.close();

if(st!=null)

st.close();

if(con!=null)

con.close();

}

catch(Exception ex) {}

}

}

* + 1. 房屋信息模块

只要是登录到人都可以查看所有的房屋信息，首先进入的是一个显示全部房屋信息的页面ViewProduct.java（如图3-2-2所示），在这个页面中，可以根据自己的需要点击按钮来查看某个房屋的具体信息（如图3-2-2所示）那就需要使用到显示详细房屋信息的页面Product.java。



图3-2-2 房屋界面

在显示所有的房屋信息时，首先对表House进行查询，使用SqlDataAdapter对象中的SelectCommand属性来选择所查询的值，将具体的房屋信息在Product.java的DataList中都显示出来。如图3-2-3所示



图3-2-3 房屋界面

* + 1. 员工模块

员工管理部分由多个小的模块组成的。员工员登录后首先进入管理截面ViewEmployeeActivity.java（如图3-2-4所示）。



图3-2-4 员工界面

房屋信息管理中，管理员可以对房屋信息进行录入，并可以对已录入系统的房屋信息做出修改，分别在页面ViewEmployeeActivity.java和Employee.java 中进行。在添加时，首先把管理员输入的值赋值给相应的字符串，然后输入数据库插入操作语句。当添加成功后，使用Response对象的Write方法把添加成功的信息显示出来

当添加成功消息显示的同时，代码将页面中的所有要填的项都置为空，这样方便下次添加进行添加输入操作，节省时间。

其他的员工操作的步骤同房屋添加类似。在使用员工功能时，管理员可以对所有注册的用户进行管理，可以在ViewEmployeeActivity.java中对用户账号进行筛选，，通过对DataGrid的属性设置中删除冻结按钮，可以将该用户账号删除或冻结。通告信息管理中，管理员可以在管理界面中，对当前要发送的信息给予录入，如果要对已经发送了的信息编辑，那么直接点击操作按钮跳转到相应的页面就可以实现了。点击确定后我们就可以对数据库写入数据，添加成功后代码就将所有需要填的项置空。管理员要想添加通告信息也在ViewEmployeeActivity.java页面下进行。在预订单的管理中，管理员要对新近产生的预订单作出该有的反映，这里只有一个删除操作，和房屋信息的删除步骤一样。

添加员工的界面

在添加新员工时，用户可以通过生成密码功能生成一个8位数字

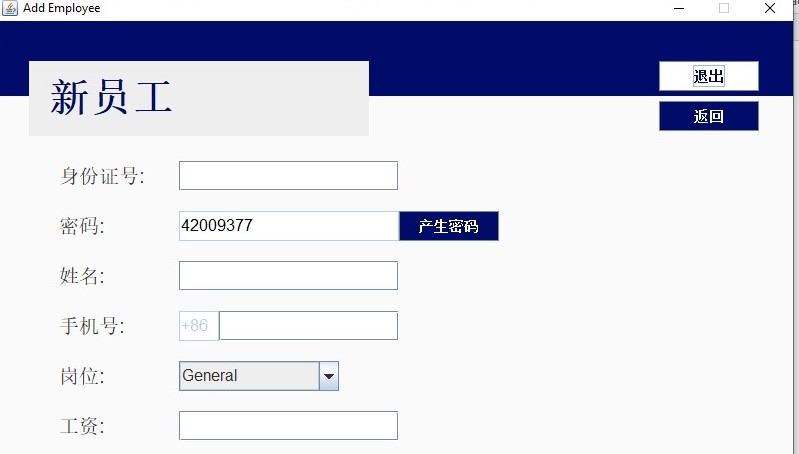


图3-2-5 员工添加界面