



**JISHOU　UNIVERSITY**

**软件学院课程设计**

|  |  |
| --- | --- |
| **题 目：** | 项目经费管理系统 |
|  |  |
| **作 者：** | 付国苑 |
| **学 号：** | 2019401306 |
| **课程名称：** | Java程序设计基础 |
| **专业年级：** | 19级软件五班 |
| **指导老师：** |  |
| **完成时间：** | 2020年1月8日 |

**吉首大学软件学院制**

**目 录**

[项目经费管理系统设计与实现 1](#_Toc61083236)

[1 项目概述 1](#_Toc61083237)

[1.1 项目描述。 1](#_Toc61083238)

[1.2 项目需求 1](#_Toc61083239)

[2 系统设计与实现 4](#_Toc61083240)

[2.1 实现系统功能所采覆盖知识点 4](#_Toc61083241)

[2.2 界面与类介绍 4](#_Toc61083242)

[2 系统测试 10](#_Toc61083243)

[4 项目特色 14](#_Toc61083244)

# 项目经费管理系统设计与实现

# 1 项目概述

## 项目描述。

近年来，随着国家对科技创新工作的重视，每一个公司、单位都运行着若干个项目，它们可以分为科技类项目、教育类项目、经济类项目等多种类型。所有的项目都会涉及到相关的资产调用，人员参与，费用产生，工作计划，效益分析等。我设计的这一套系统，可以让用户在创建好自己的账号密码之后，直观清晰的了解当前公司、单位中已有的项目数量、所有项目的薪资总和、禁飞最高的项目以及经费最低的项目，还可以申请以及结算自己负责的项目。

## 1.2 项目需求

### 1.2.1 功能需求

本系统可以为用户提供相应项目经费管理的服务。用户可以在登录帐号之后直观清晰的了解当前公司、单位中已有的项目数量、所有项目的薪资总和、禁飞最高的项目以及经费最低的项目，还可以申请以及结算自己负责的项目。

本项目至少完成以下相关的功能或模块：

1）注册新用户：为新用户注册符合要求的账号密码，并保存至数据库文件中，下次登录可以直接使用该账号密码；

2）用户登录：输入数据库文件已保存的账号密码，匹配成功则可以进入该系统的主功能界面；

3）修改用户信息：可以根据用户输入符合要求的数据来更换原本的账号密码，并保存至数据库中。

4）添加项目：可以申请项目负责人为自己的项目，项目申请成功会在主界面中更新显示出来。

5）删除项目：可以结算项目负责人为自己的项目，项目结算成功会在主界面中更新显示出来。

6）查看所有项目信息：可以显示出已有的项目数量、所有项目的薪资总和、禁飞最高的项目以及经费最低的项目。并且会根据用户的操作实时更新。

### 1.2.2 用例分析

下面使用总体用例图可以更加清晰的表达整个系统功能，总体用例图如图1-1所示。

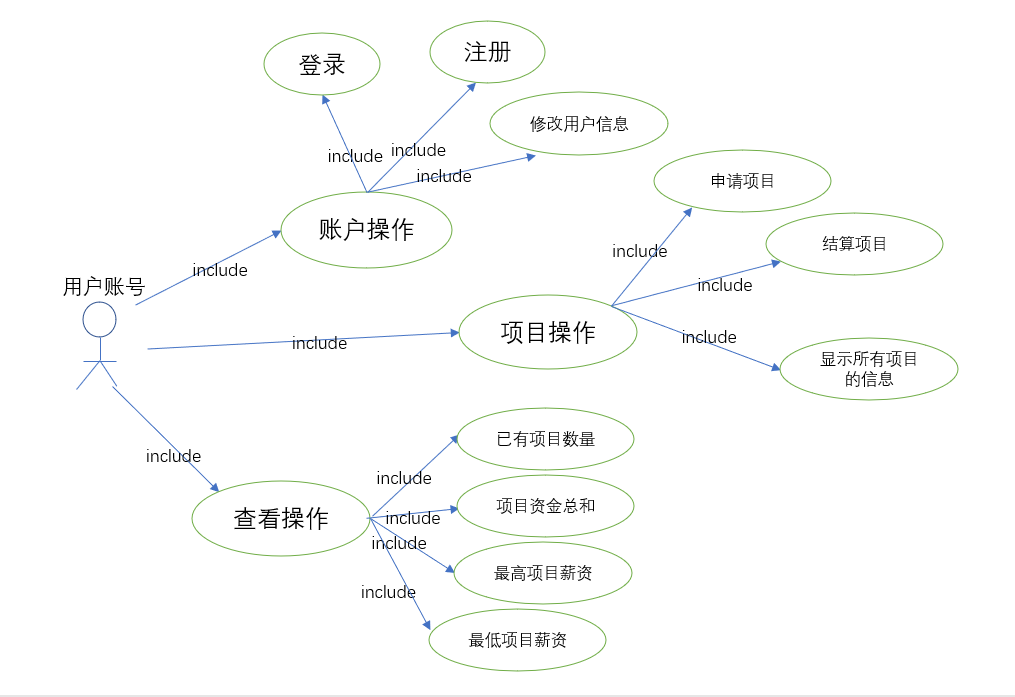


图1-1 总体用例图

### 1.2.3 系统流程图

本管理系统实现包括了登录以及注册账号，并且可以修改用户的信息；查看已有的项目数量、所有项目的薪资总和、禁飞最高的项目以及经费最低的项目；还可以申请以及结算自己负责的项目。

在界面启动时，首先需要用户的输入账号密码，若用户没有账号，则可以点击注册按钮进入注册界面，填写相关信息注册后，再行登录。登录后将跳转到主界面。

主界面直接分为三大模块。

一：账号操作：进入主界面，会显示当前用户的账号名称，还包含了修改用户信息按钮可以点击打开修改用户信息界面。主界面会实时显示当前用户的账号名称；

二：项目管理操作：进入主界面之后，会在左边的表格中列出所有项目的信息，还包含了申请项目和结算项目的按钮，再点击后可以分别打开申请项目以及结算项目的窗口。主界面也会及时更新显示所有项目的信息；

三：查看操作：进入主界面案之后，会有查看项目信息的按钮，再点击后可以打开查看项目信息界面；

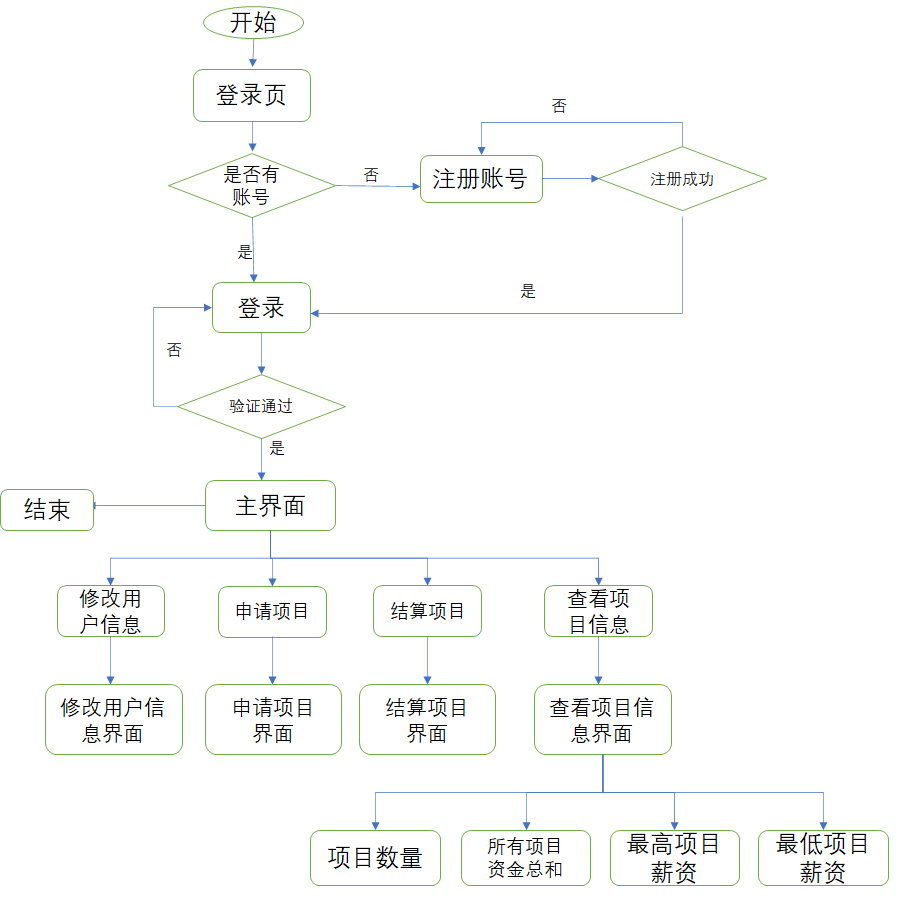


图1-2 系统流程图

### 1.2.3 数据需求

服务端采用MySQL数据库。根据实体设计，共二张数据表。数据库实体的E-R图如图1-3所示：

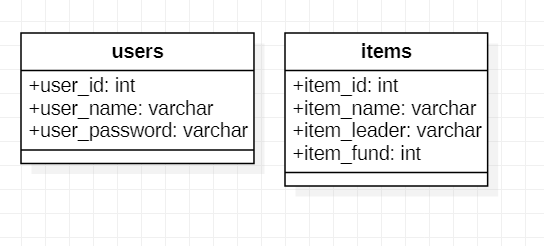


图1-3 服务端数据库实体E-R图

表设计如下：

表1-1 用户表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 说明 | 类型 | 长度 | 可否为空 | 主键 |
| 01 | user\_id | 编号 | int | 默认 | 否 | 是 |
| 02 | user\_name | 用户账号 | varchar | 20 | 否 | 否 |
| 03 | user\_password | 用户密码 | varchar | 20 | 否 | 否 |

表1-2 项目表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 说明 | 类型 | 长度 | 可否为空 | 主键 |
| 01 | item\_id | 项目编号 | int | 默认 | 否 | 是 |
| 02 | item\_name | 项目名称 | varchar | 20 | 否 | 否 |
| 03 | item\_leader | 项目负责人 | varchar | 20 | 否 | 否 |
| 04 | item\_fund | 项目资金 | Int | 默认 | 否 | 否 |

# 2 系统设计与实现

## 2.1 实现系统功能所采覆盖知识点

本系统总共实现了 个模块，各模块功能实现所采用技术见表2-1所示。

表2-1 系统功能及所采用技术一览表

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **功能点** | **基本知识点** | **扩展知识点** |
| 1 | 注册 | 1. JavaFx图形用户界面  2数据库:使用JDBC实现数据的增、删、改、查、排序等操作; | 1.泛型 |
| 2 | 登录 | 2.Java反射机制 |
| 3 | 申请、结算项目 | 3.配置文件的使用 |
| 4 | 修改用户信息 | 4.集合 |
| 5 | 查看项目信息 | 5.MVC设计模式 |
| 6 | 显示所有项目信息列表 | 6.排序功能 |
| 7.Junit单元测试 |

## 2.2 界面与类介绍

**2.2.1 界面设计与实现**

读者信息管理实现主要是对读者信息进行相关设置与维护操作。具体的实现界面如图2-1、图2-2所示。



图2-1登录界面

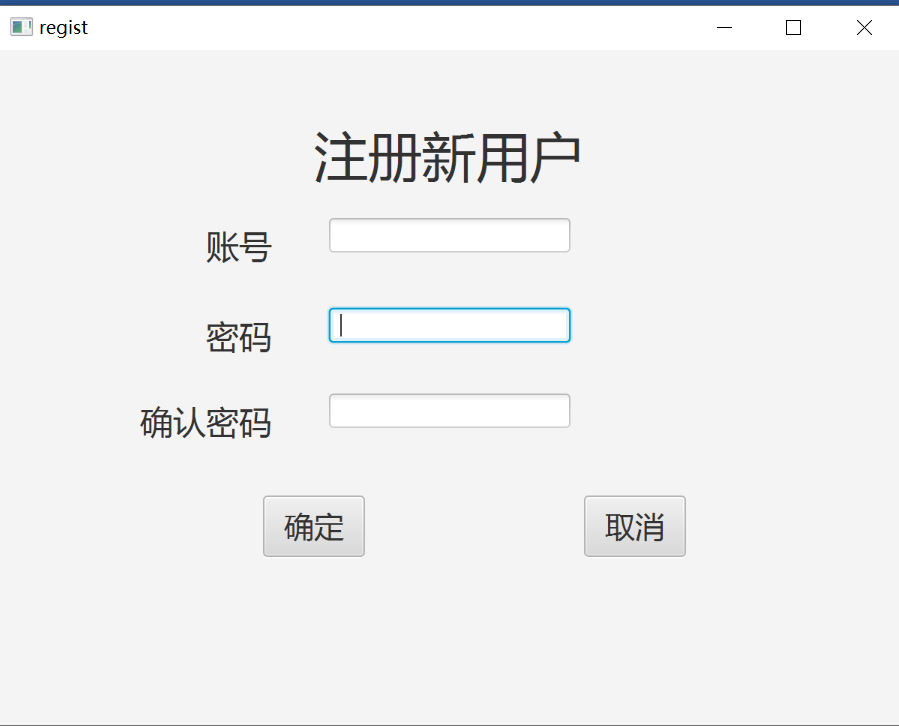


图2-2 注册界面



图2-3 主功能界面



图2-4 更改用户信息界面



图2-5 项目申请界面



图2-6 项目结算界面



图2-7 查看项目信息界面

**2.2.2 类设计与实现**

类和方法如表2-1所示。

表2-1 读者信息管理模块类和方法列表

|  |  |
| --- | --- |
| 类名称 | 说明 |
| addItem | 添加项目界面控制类 |
| changeUserInfoController | 修改用户信息界面控制类 |
| checkItemController | 查看项目信息界面控制类 |
| endItem | 结算项目界面控制类 |
| loginController | 登陆界面控制类 |
| MainPageController | 主功能界面控制类 |
| registController | 登陆界面控制类 |
| ItemServices | 项目操作类 |
| userServices | 用户操作类 |
| ItemDao | 项目实现类 |
| userDao | 用户实现类 |
| DataBaseUtil | 数据库操作工具类 |
|  |  |
| DataBaseUtil中的方法 | |
| public static Connection getConnection() | 获取数据库连接 |
| public static void CloseConnection() | 关闭数据库连接 |
| public static void synthesize() | 对数据库文件的增删改操作 |
| public static <T> List<T> select() | 数据库查询操作 |

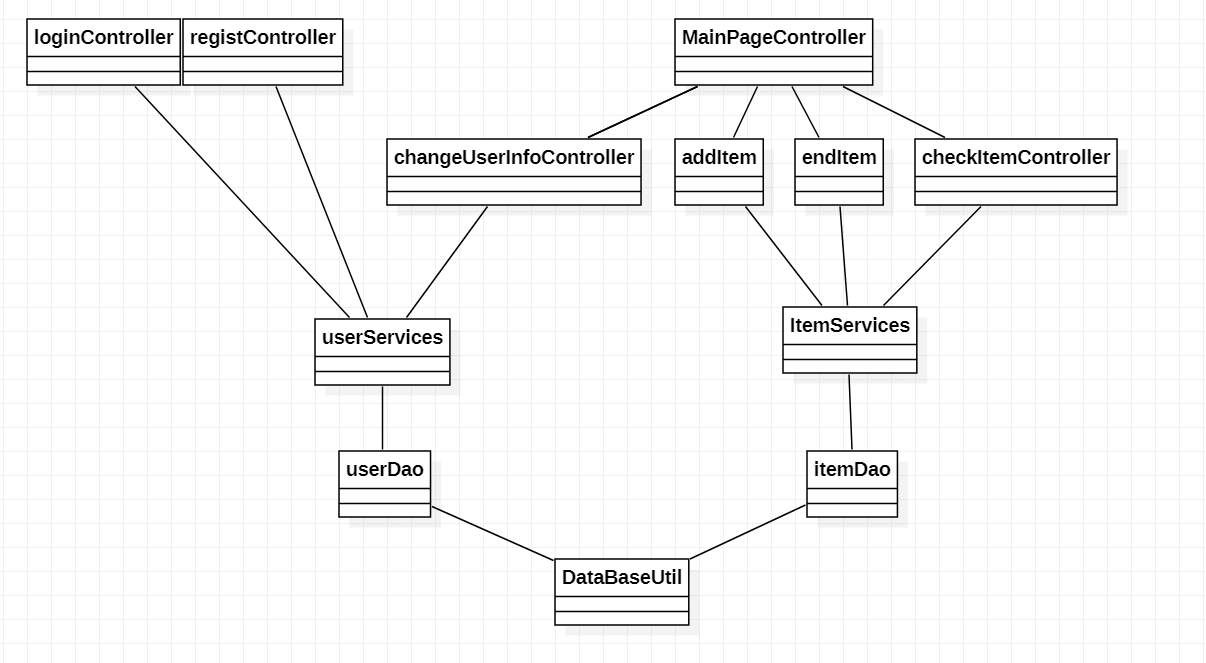


图2-8所有类的关系图

# 系统测试

1. public class addItemTest {
2. @Test
3. public void test01(){
4. loginController.localUserName = "test";
5. String itemLeader = "test";
6. //String itemLeader = "test1";
7. String itemName = "项目管理系统";
8. String itemFund = "1000";
9. //String itemFund = "";
10. boolean result = ItemServices.judgeItem(itemLeader,itemName,itemFund);
11. if(result){
12. if(ItemServices.addItem(itemLeader,itemName,itemFund)){
13. System.out.println("成功");
14. }else{
15. System.out.println("失败");
16. }
17. }else{
18. System.out.println("失败");
19. }
20. }
21. public class ChangeUserInfoTest {
22. @Test
23. public void test01(){
24. String username = "fgy";
25. String password = "123456";
26. //String password = "";
27. boolean result = userServices.judgeUsernamePassword(username,password);
28. if(result){
29. //修改成功
30. userDao.updateAccount(username,password);
31. System.out.println("成功");
32. }else{
33. System.out.println("失败");
34. }
35. }
36. }
37. }
38. public class Default\_Database {
39. @Test
40. public void setting\_items() {
41. for (int i = 0; i < 50; i++) {
42. StringBuilder itemName = new StringBuilder();
43. StringBuilder itemLeader = new StringBuilder();
44. StringBuilder itemFund = new StringBuilder();
45. for (int j = 0; j < 3; j++) {
46. itemName.append("赵钱孙李周吴郑王冯陈诸卫蒋沈韩杨朱秦尤许何吕施张孔曹严华金魏陶姜戚谢邹喻柏水窦章云苏潘葛奚范彭郎鲁韦昌马苗费凤花方俞任袁柳酆鲍史唐费廉岑薛雷贺倪汤滕殷罗毕郝邬安常乐于时傅皮齐康伍余元顾孟平黄下和穆萧尹姚邵何" .charAt((int) (Math.random()\*100)));
47. }
48. for (int j = 0; j < 4; j++) {
49. itemLeader.append("赵钱孙李周吴郑王冯陈诸卫蒋沈韩杨朱秦尤许何吕施张孔曹严华金魏陶姜戚谢邹喻柏水窦章云苏潘葛奚范彭郎鲁费韦昌马苗凤花方俞任袁柳酆鲍史唐费廉岑薛雷贺倪汤滕殷罗毕郝邬安常乐于时傅皮齐康伍余元顾孟平黄下和穆萧尹姚邵何" .charAt((int) (Math.random()\*100)));
50. }
51. itemLeader.append("项目");
52. itemFund.append((int)( Math.random()\*10000));
53. ItemDao.addItem(itemLeader.toString(),itemName.toString(),Integer.parseInt(itemFund.toString()));
54. System.out.println(1);
55. }
56. }
57. }
58. public class endItemTest {
59. @Test
60. public void test01(){
61. loginController.localUserName = "test";
62. String itemLeader = "test";
63. //String itemLeader = "test1";
64. String itemName = "100";
65. //String itemName = "1";
66. boolean result = ItemServices.judgeItem(itemLeader,itemName);
67. if(result){
68. System.out.println("成功");
69. }else{
70. System.out.println("失败");
71. }
72. }
73. }
74. public class ItemDemo {
75. @Test
76. public void test01(){
77. Connection connection = DataBaseUtil.getConnection();
78. String sql ="select item\_id as id,item\_name as itemName,item\_leader as itemLeader,item\_fund as itemFund from items";
79. List list = DataBaseUtil.select(Item.class, connection , sql);
80. for (int i = 0; i < list.size(); i++) {
81. System.out.println(list.get(i).toString());
82. }
83. }
84. }
85. public class TestDemo {
86. @Test
87. public void test01(){
88. Connection connection = DataBaseUtil.getConnection();
89. System.out.println(connection);
90. }
91. }

# 4 项目特色

至少写3条以上自己认为有特色地方，可以从功能或者技术方面描述。

1. 使用了MVC设计模式：M是指业务模型，V是指用户界面，C则是控制器，使用MVC的目的是将M和V的实现[代码](https://baike.baidu.com/item/%E4%BB%A3%E7%A0%81/86048)分离，从而使同一个程序可以使用不同的表现形式。其中，View的定义比较清晰，就是用户界面。
2. 使用了单元测试类：测试项目中重点实现的方法。
3. 给数据库中添加了大量的数据用于测试，尽量提高系统的安全性，减少因为极端情况导致的系统故障
4. 使用了配置文件，降低代码冗余。

