二、概要设计说明书

[1．引言 2](#_Toc506972944)

[1.1编写目的 2](#_Toc506972945)

[1.2项目背景 2](#_Toc506972946)

[1.3定义 2](#_Toc506972947)

[1.4参考资料 2](#_Toc506972948)

[2．任务概述 2](#_Toc506972949)

[2.1目标 2](#_Toc506972950)

[2.2运行环境 3](#_Toc506972951)

[2.3需求概述 3](#_Toc506972952)

[2.4条件与限制 3](#_Toc506972953)

[3．总体设计 3](#_Toc506972954)

[3.1处理流程 3](#_Toc506972955)

[3.2总体结构和模块外部设计 7](#_Toc506972956)

[3.3功能分配 9](#_Toc506972957)

[4．接口设计 9](#_Toc506972958)

[4.1外部接口 9](#_Toc506972959)

[4.2内部接口 12](#_Toc506972960)

[5．数据结构设计](#_Toc506972961) 13

[5.1逻辑结构设计 13](#_Toc506972962)

[5.2物理结构设计 13](#_Toc506972963)

[5.3数据结构与程序的关系 14](#_Toc506972964)

[6．运行设计 15](#_Toc506972965)

[6.1运行模块的组合 15](#_Toc506972966)

[6.2运行控制 15](#_Toc506972967)

[6.3运行时间 15](#_Toc506972968)

[7．出错处理设计 15](#_Toc506972969)

[7.1出错输出信息 15](#_Toc506972970)

[7.2出错处理对策 15](#_Toc506972971)

[8.安全保密设计](#_Toc506972972) 16

[9.维护设计 17](#_Toc506972973)

## 1.1编写目的

为了更好的管理粉丝会员信息，进行会员积分消费，设计了这个系统。

本需求分析说明书的主要读者为（1）项目经理、设计人员、开发人员和测试人员，为明确软件需求、安排项目规划与进度、组织软件开发与测试，规范化本系统的编写；（2）用户，提出功能性需求、非功能性需求。 各个部门可重点阅读与本部门相关的内容。了解是否满足用户需求，以期调整和完善该管理系统。

## 1.2项目背景

随着现在社会发展，既是社会发展使然，更是人之常情所至，所以明星现象越来越严重，而追星的年龄层比较低，缺少形式化的管理，造成混乱，顺应时代发展，需要一个专业的系统来管理明星的粉丝，共管理人员进行使用参考。

## 1.3定义

## 本文档中没用到的专门术语的定义和缩写词的原文。

## 1.4参考资料

1. 软件工程——原理，方法与应用
2. 面向对象分析与设计（第三版）电子工业出版社
3. java办公自动化项目精解

# 2．任务概述

## 2.1目标

**2.1.1系统开发的意图**

为了更好的管理粉丝会员信息，进行会员积分消费，顺应管理需求并为高层管理人员提供大量丰富的数据，设计了本系统。

**2.1.2应用目标**

能够切实简化粉丝团内部管理，提高工作效率简化工作流程。

**2.1.3作用范围**

粉丝部落

## 2.2运行环境

操作系统：Microsoft Windows 2000 Advanced Server

支持环境：IIS 5.0

数 据 库：Microsoft SQL Server 2000

## 2.3条件与限制

完成本项目应具备的条件：熟练掌握java语言以及hibernate进行数据库操作，熟练掌握在myeclipse中使用jframe插件设计图形用户界面。

硬件配置要求：系统未做说明

证件号必须为身份证号；

# 3．总体设计

## 3.1处理流程

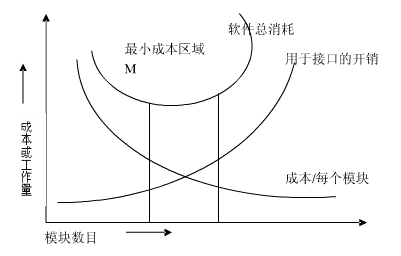
因为本系统设计的主要用户是粉丝，所以这里画出粉丝的业务流程图。



## 3.2总体结构和模块外部设计

**3.2.1软件系统结构**

模块是软件结构的基础，软件结构的好坏完全由模块的属性体现出来，把软件模块化地目的是为了降低软件复杂性，使软件设计、测试、调试、维护等工作变得简易，但随着模块数目的增加，通过接口连接这些模块的工作量也随之增加。软件设计的工作量如图所示。在考虑模块化的时候，应尽量使模块数量接近于M值，使得研发工作量最小，避免模块化不足或者超量。



软件结构图如下：

系统

修改密码

会员

注册

退出粉丝团

查询会员

会员消费

会员积分

修改信息

## 软件领域模型图



**3.2.2模块类图**

**3.2.2.1登录**



类图说明：

Getmembernumber

功能：得到相应管理员的密码

Getpassword

功能：根据会员证件号码得到会员的信息

Membernumber：会员的证件号码

**3.2.2.1注册**



类图说明：

Setmembermumber

功能：设置证件号码

Setmemberpassword

功能：设置登录密码

Setmembername

功能：设置会员姓名

Setmemberhonenumber

设置会员手机号码

Seyvipnum

设置会员号码

Setmembercost

设置当前积分

Save

保存信息进入数据库



类图说明：

Setmembermumber

功能：设置证件号码

Setmemberpassword

功能：设置登录密码

Setmembername

功能：设置会员姓名

Setmemberhonenumber

设置会员手机号码

Seyvipnum

设置会员号码

Setmembercost

设置当前积分

Save

保存信息进入数据库

Delete

删除原来的信息



Setmemberassword

修改密码

Delete

修改原来信息

Save

保存新密码

Findbyid

通过membernumber找到数据库中相应的对象



Findbyid

通过membernumber查找到数据库中相应的对象

Getmembercost

得到原来的会员积分

Setmembercost

得到新的会员积分



Delete

删除所有的信息

## 3.3功能分配

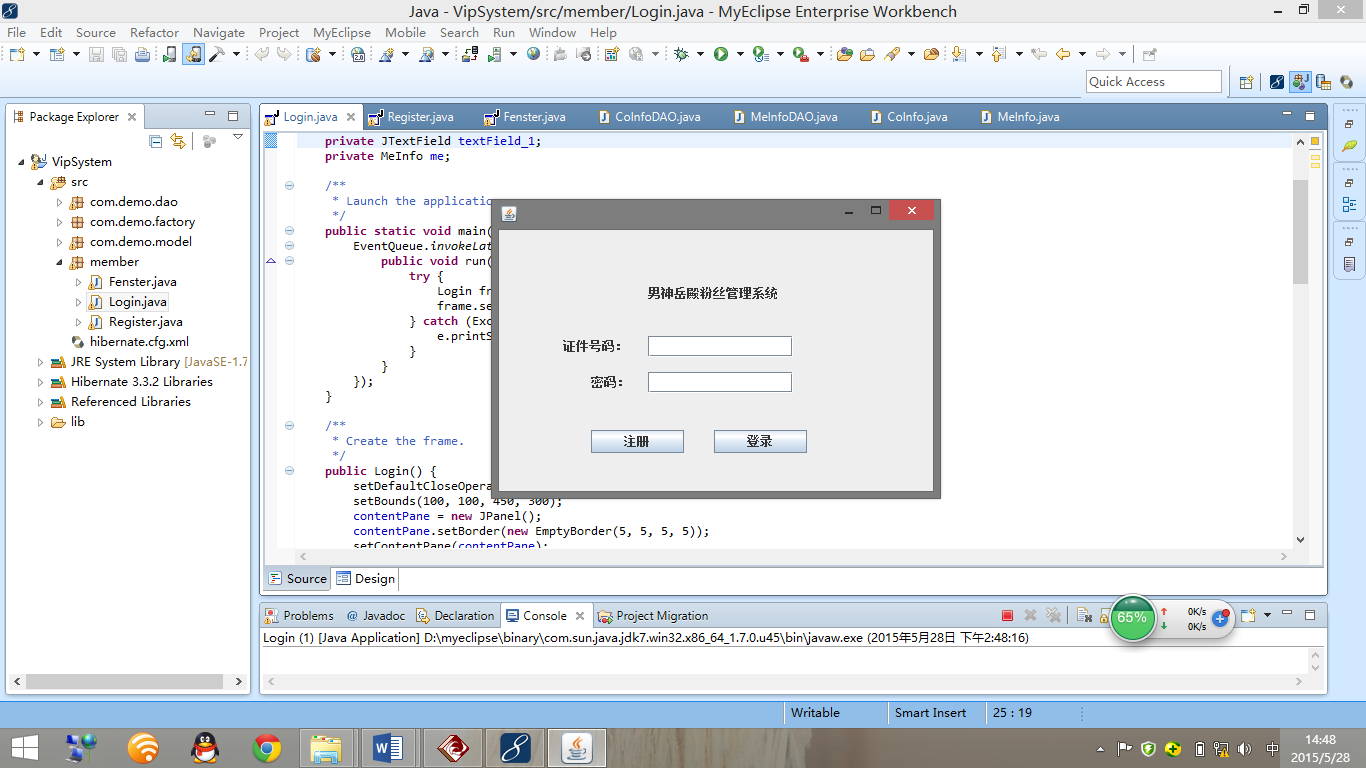
|  |  |
| --- | --- |
| 模块名称 | 功能分析 |
| 登陆 | 用户登陆 |
| 注册 | 注册新用户 |
| 修改信息 | 修改原先输入的信息 |
| 修改密码 | 修改原来的密码 |
| 粉丝消费 | 购买专辑与海报进行消费 |
| 退出部落 | 删除自己的信息，退出粉丝团 |

# 4．接口设计

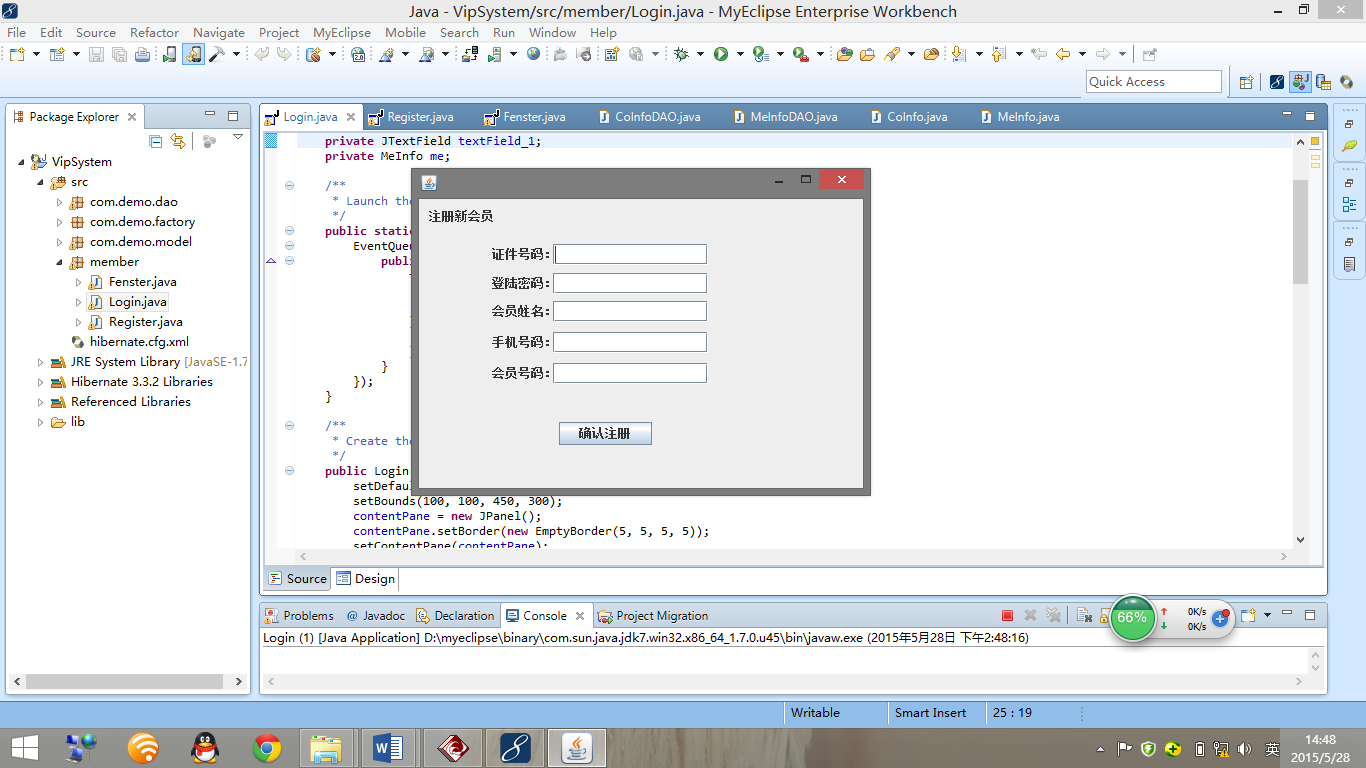
## 4.1外部接口

**4.1.1用户界面**

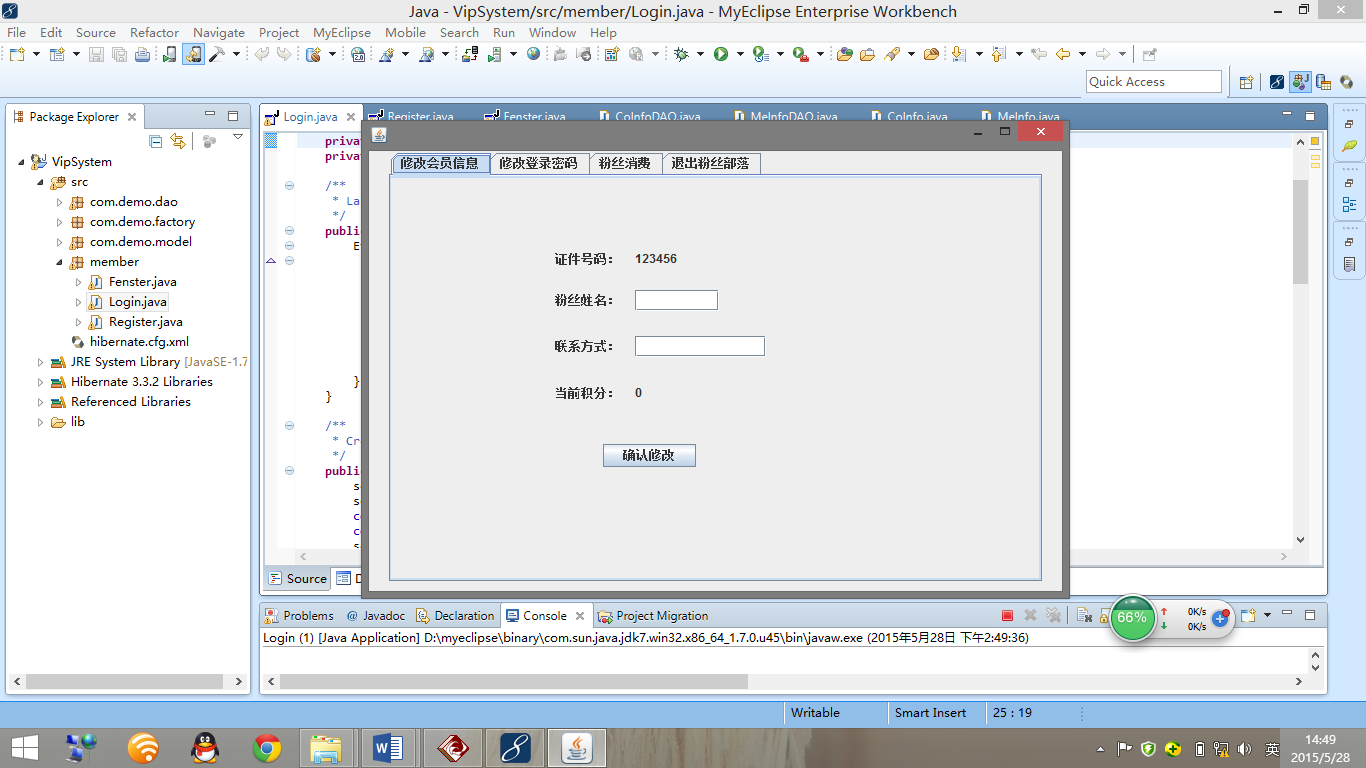
**登录界面**



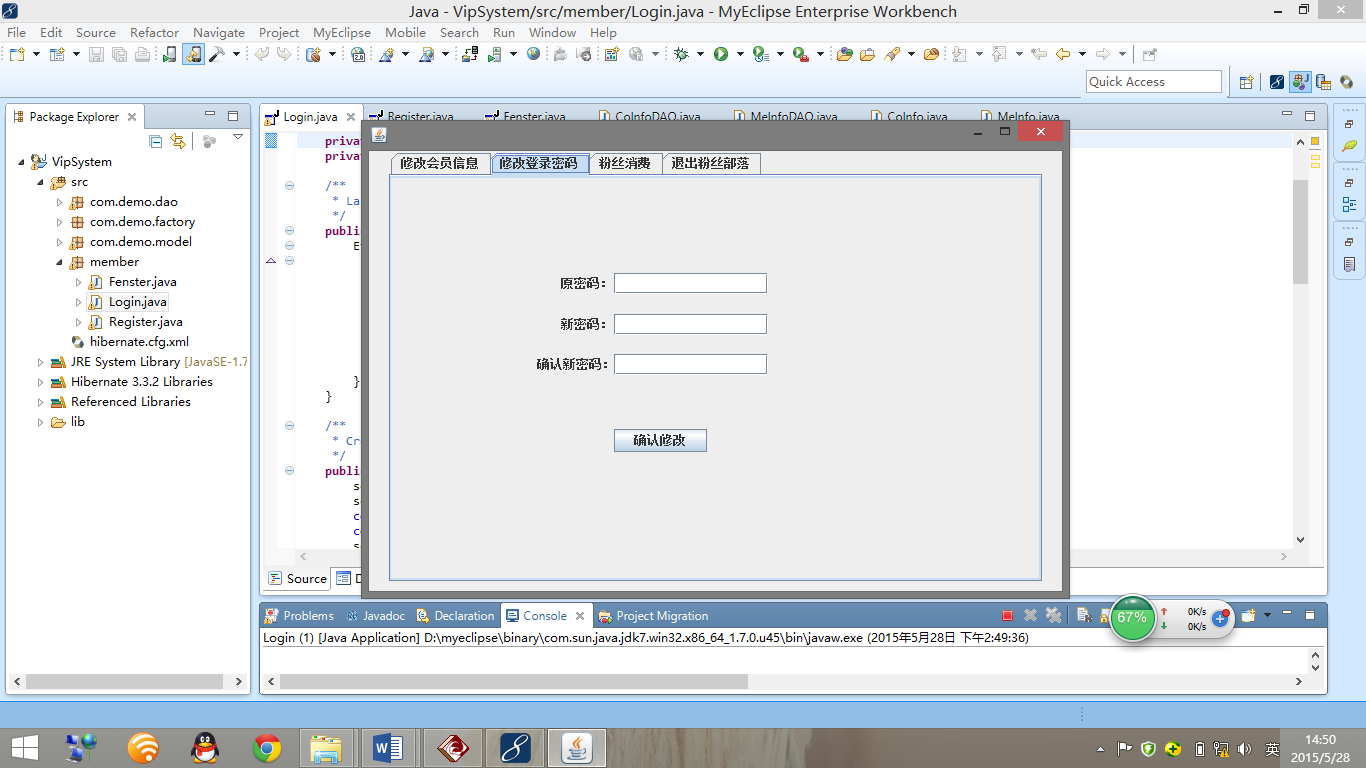
注册界面



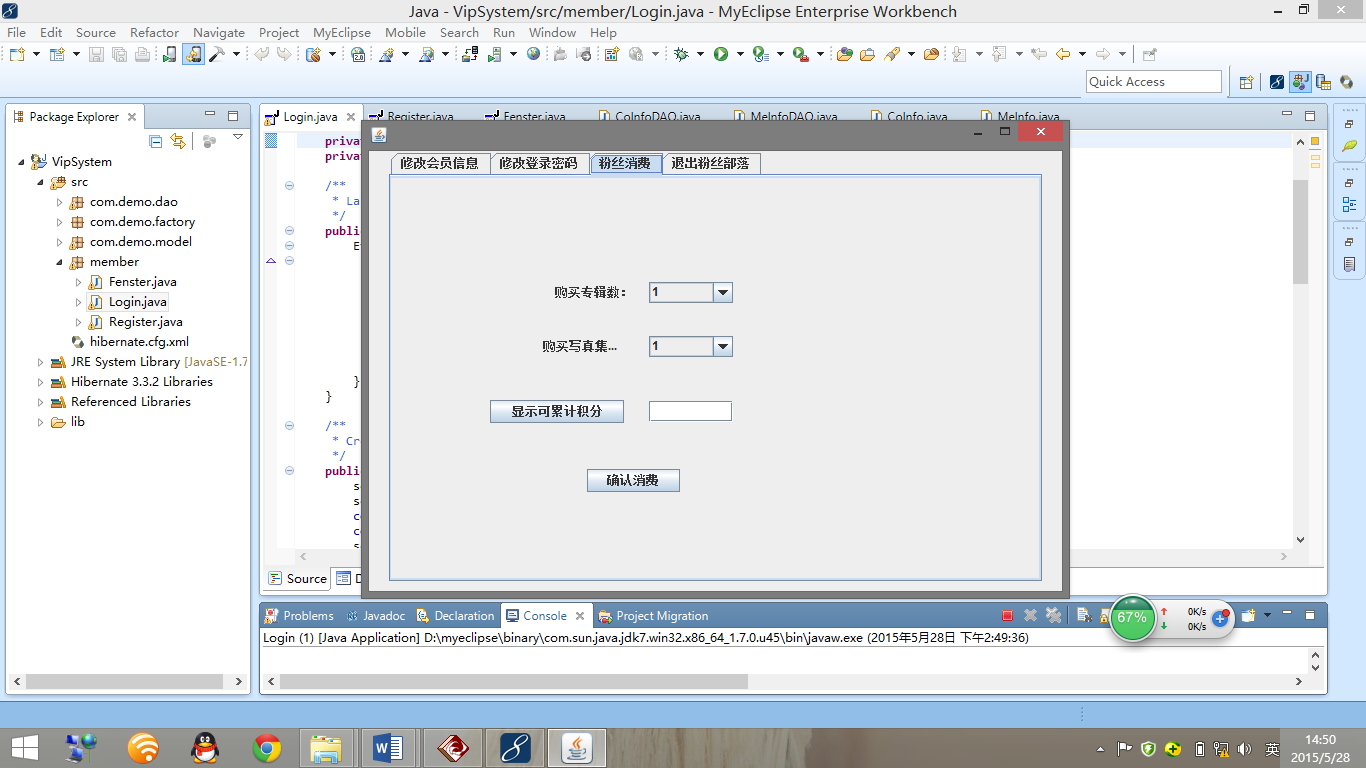
主界面——修改会员信息



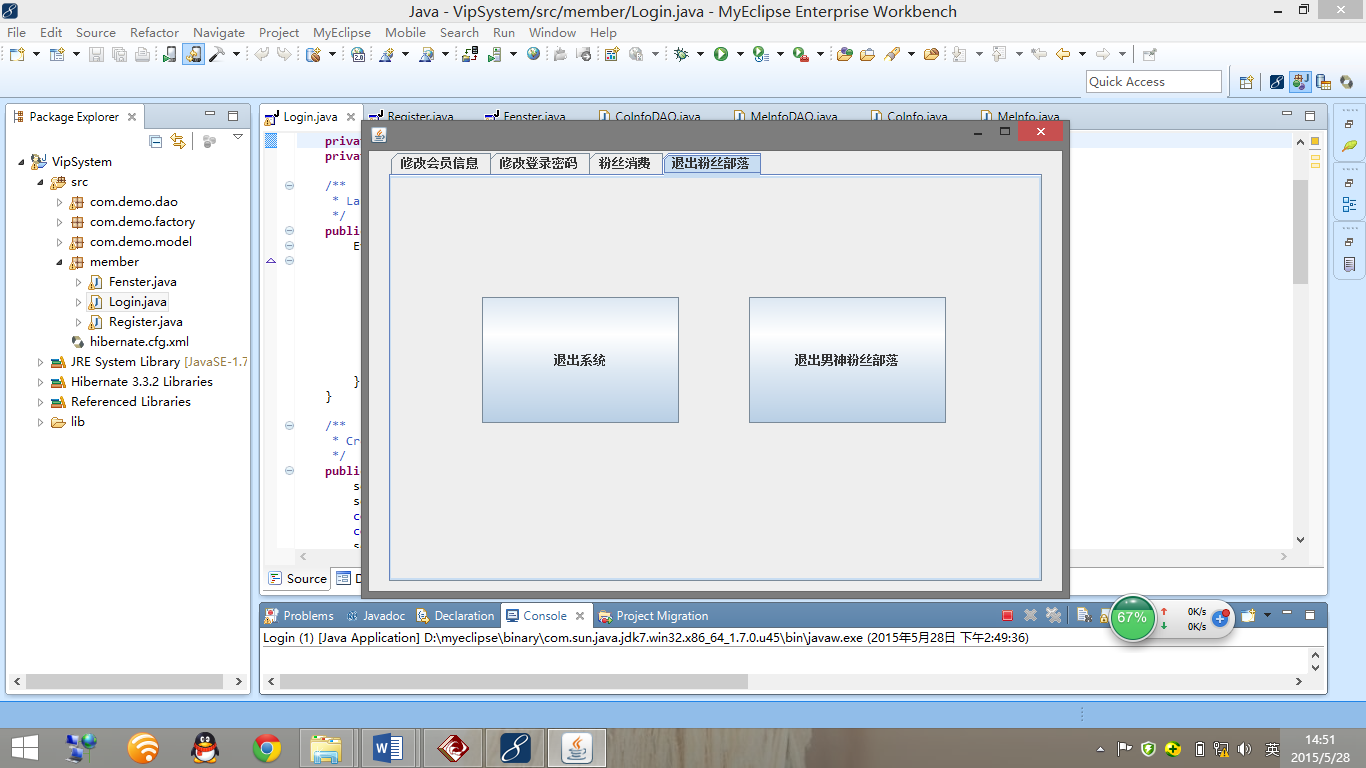
主界面——修改登录密码



主界面——粉丝消费



主界面——退出粉丝部落



**4.1.2软件接口**

软件运行在windows操作系统之上，系统分为三层结构：表示层、应用层、数据存储层，采用MVC架构，软件在后台与数据库进行交互。

java+swing打包jar转成exe并动态绑定jre再打包成安装文件方式适用于windows平台，使用java+swing框架生成jar发布文件适用于任何操作系统平台，需要java虚拟机支持。

**4.1.3硬件接口**

无需特殊的硬件接口。

## 4.2内部接口

各模块之间采用函数调用、参数传递、返回值的方式进行信息传递。具体参数结构将在下面的数据结构中说明。接口传递的信息是以数据结构封装好的数据，以参数和返回值的形式在各个模块之间传递。本系统各个模块之间直接的数据传递比较少，大多数的数据传递是通过底层数据库表之间的关联关系完成。

# 5．数据结构设计

## 5.1逻辑结构设计

**5.1.1数据库表设计**

粉丝信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 含义 | 是否可以为空 | 备注 |
| memberNumber | Varchar | 会员证件号码 | 否 | 主键 |
| memberPassword | Varchar | 登陆密码 | 否 |  |
| memberName | Varchar | 会员姓名 | 否 |  |
| memberPhoneNumber | Varchar | 会员手机号码 | 否 |  |
| memberCost | Int | 会员积分 | 是 |  |
| vipNum | Varchar | 会员号码 | 否 |  |
|  |  |  |  |  |

粉丝消费信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 含义 | 是否可以为空 | 备注 |
| memberNumber | Varchar | 会员证件号码 | 否 | 主键 |
| vipPay | Int | 消费金额 | 是 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**5.1.2E-R图**

**5.1.2.1实体E-R图**





## 5.2物理结构设计

物理结构设计主要是设计数据在模块中的表现形式。这里的物理结构主要由数据库以及hibernate来设计，数据在模块中以对象的方式来标识。批量存储或读取则利用链表存储封装好的数据对象。

## 5.3数据结构与程序的关系

数据结构通过hibernate数据库语言HQL生成。接口之间传递的数据是以数据结构封装好了的数据，以参数传递或者返回值的形式在各个模块之间传递。

# 6．运行设计

## 6.1运行模块的组合

具体软件的运行模块组合为程序多窗口的运行环境，各个模块在软件运行过程中能较好的交换信息，处理数据。 程序在有输入或者触发时间需要对数据进行操作的时候，启动数据模块，通过各个模块之间的调用，读入并对输入的数据或者对需要的数据进行格式化。然后数据模块会对数据进行处理，判断结果，将数据在显示器上动态显示出来（实时刷新）。

## 6.2运行控制

运行控制将严格按照各模块之间的函数调用来执行。各事务中心模块中需要对运行控制进行正确的判断，选择正确的运行控制路径。本软件内部操作采用WINDOWS操作界面，主窗体采用左右键菜单和按钮方式只需单击即可打开所需的操作。软件界面相对比较简洁易懂，基本能够实现用户的数据处理要求。

## 6.3运行时间

在软件的需求分析中所做的时间特性要求，必须对有效操作做出较快的反应，保证良好的用户体验。

# 7．出错处理设计

## 7.1出错输出信息

|  |  |  |
| --- | --- | --- |
| 出错情况 | 系统提示 | 处理方法 |
| 用户不能进行录入操作 | 用户权限不够 | 注册高级帐号 |
| 输入数据不能写入数据表  主键重复 | 您所录入的信息关键字重复请重新命名关键词 | 重新命名关键词 |
| 查询信息数据库中不存在 | 没有查到您要查找的信息 | 重新查询 |
| 程序打开没有数据表 | 没有找到数据表 | 检查SQLServer是否已经连接 |

## 7.2出错处理对策

说明故障出现后可能采取的变通措施，包括：

1.         后备技术：暂不考虑数据库文件备份；

2.         降效技术：暂不考虑降效技术；

3.         恢复及再启动技术：该软件对断电输入的数据而且没有储存，没有恢复功能，需重新录入。

# 8.安全保密设计

程序设定用户组权限，针对不同用户组所拥有的权限不同。当用户使用系统时,系统会审核用户的合法性,并根据不同的用户组的使用权限使进行不同的权限操作，所以本系统相对安全。

# 9.维护设计

系统比较小，维护相对简单。没有外加维护模块。