|  |  |
| --- | --- |
| 231 | 123 |
| **PINGDINGSHAN UNIVERSITY** |

**《.net开发实训》**

|  |
| --- |
| **成 绩** |
|  |

**课程设计报告**

**题 目: 个人财务管理系统**

**院(系):** 计算机学院（软件学院）

**专业年级:**  计算机科学与技术2016级

**姓 名:** 李迎波

**学 号:** 161210240

**任课教师:** 李永明

**2018年06月21日**

目 录

[1 绪论 1](#_Toc518407587)

[1.1 课题背景及意义 1](#_Toc518407588)

[1.2 论文主要工作和结构安排 2](#_Toc518407589)

[2 需求分析 2](#_Toc518407590)

[2.1 业务流程描述 3](#_Toc518407591)

[2.2 系统功能需求分析 3](#_Toc518407592)

[2.3 系统性能需求分析 5](#_Toc518407593)

[2.4 本章小结 6](#_Toc518407594)

[3 系统总体设计 7](#_Toc518407595)

[3.1 系统功能结构图 7](#_Toc518407596)

[3.2 系统主要界面设计 8](#_Toc518407597)

[3.2.1 注册界面设计 8](#_Toc518407598)

[3.2.2 主界面设计 9](#_Toc518407599)

[3.2.3 账户管理界面设计 10](#_Toc518407600)

[3.3 数据库设计 12](#_Toc518407601)

[3.4 本章小结 14](#_Toc518407602)

[4 结束语 15](#_Toc518407603)

[4.1 总结 15](#_Toc518407604)

[4.2 展望 15](#_Toc518407605)

[附 录 16](#_Toc518407606)

[**致 谢** 19](#_Toc518407608)

1 绪论

该部分主要介绍当前个人财务管理系统的背景及主要内容。基于数据库的财务管理系统能够方便快捷地管理个人财务，轻松理财。本章主要描述个人财务管理系统的开发背景、项目研究的目的和意义以及项目的主要特点，最后介绍了论文的结构。

## 1.1 课题背景及意义

当今社会越来越朝着信息化数字化发展，计算机的应用领域使我们的生活越来越方便和快捷，可以说我们每个人的生活都离不开计算机的帮助了。越来越多的个人应用软件成为人们重要的助手，也在潜移默化的改变着我们的生活。

随着现代科学技术的迅猛发展，计算机技术已经渗透到各个领域，成为各个行业必不可少的工具，特别是网络技术的推广和信息高速公路的建立，使 IT 产业在市场竞争中越发显示出其独特的优势，步入数字化时代，有巨大的数据信息等待着加工处理和传输，这使得对应用软的开发和利用显得尤为迫切。

心理学家马斯洛的需求理论告诉我们，人类的需求是有层级之分的：在安全无虞的前提下，追求温饱；当基本的生活条件获得满足之後，则要求得到社会的尊重；并进一步追求人生的最终目标自我实现。而要依层级满足这些需求，必须建立在不虞匮乏的财务条件之上。因此，你必须认识理财的重要，订定一套适合自己的理财计划，来达成自己的生活目标。理财须要了解自己的财务情况，再进行投资等规划，此时就需要财务管理软件用于记录和管理个人财务情况。个人财务管理软件是帮助实现个人财务管理的计算机应用程序。理财须要了解自己的财务情况，再进行投资等规划，此时就需要财务管理软件用于记录和管理个人财务情况。现在常见的个人财务管理软件可以完成收支记帐、借贷管理、统计分析、帐户管理等功能，帮助用户能更好的管理个人财务！操作简单，报表分析功能能让用户随时可以了解自己的财务状况。我们经常看到企业的一些财务报表，这些报表主要是关注企业的资产负债表和损益表。简答的说，前者是反映这个企业有多少钱，是存量；后者，反映了企业在一定时间能赚到多少钱，是流量。其实，对于个人或者家庭的财务情况，也是关注这两个方面。关注自己存了多少钱，赚了多少钱，花了多少钱，借出多少等等[1]。

现在人本都追求高质量，高水平的生活，设计此系统可以免去一些繁琐的手工记录，而且节省时间和资源，为人们创造更多的价值和效益。个人财务管理软件是帮助实现个人财务管理的计算机应用程序。理财须要了解自己的财务情况，再进行投资等规划，此时就需要财务管理软件用于记录和管理个人财务情况。理财须要了解自己的财务情况，再进行投资等规划，此时就需要财务管理软件用于记录和管理个人财务情况。此系统的主要功能有如下：

1.实现用户注册。

2.实现验证用户名和密码的正确性再登入，以及修改密码和退出系统。

3.对个人的银行帐号信息，可进行查询信息、增加、删除和修改信息的功能。

4.对个人借贷信息，可进行查询信息、增加、删除和修改信息的功能。

## 1.2 论文主要工作和结构安排

1．本文主要讲解个人财务管理系统的设计及实现，管理系统针对用户要求用计算机，对自己的财务收入、支出以及相关的各种信息进行记录、修改、添加、查询而设计的一种现代化软件。论文主要讲述个人财务管理系统的开发背景、项目研究的目的和意义以及项目的主要特点

1. 本文总共由四个部分组成，各个部分的内容如下：

（1）绪论，对系统的背景、研究目的和项目特点进行介绍，然后做了详细的工作安排，明确工作的方向和目标，最后介绍论文的结构。

（2）需求分析，首先描述了共享仓库系统的业务操作流程，然后对系统的功能需求进行分析，最后介绍了系统的性能需求。

（3）系统总体设计，开头介绍了本系统的功能结构设计并且画出其功能模块图，再对系统的功能设计和界面设计进行详细的介绍，其中主要的界面设计有三个，分别是：注册界面、账户管理界面和个人财务管理系统主界面，最后根据用户的需求，设计出合理的数据库。

（4）结束语，对整个的论文工作做出总结，对系统的前景进行展望。

2 需求分析

个人财务管理系统是每一个个人或家庭管理财务收入和支出的不可缺少的一个管理信息系统,它的内容对于个人来说是至关重要的,所以个人财务管理系统应该能够为每一个用户提供充足的信息和快捷的查询手段，大的方便个人用户的合理理财。

现代科学技术的迅猛发展，计算机技术已经渗透到各个领域，成为各个行业必不可少的工具，特别是网络技术的推广和信息高速公路的建立，使 IT 产业在市场竞争中越发显示出其独特的优势，步入数字化时代，有巨大的数据信息等待着加工处理和传输，这使得对应用软的开发和利用显得尤为迫切。

本章从系统的业务操作流程、系统的设计目标、功能需求和性能需求进行分析。

## 2.1 业务流程描述

基于.NET的个人财务管理系统从功能上讲，主要系统管理、系统维护、实用工具、存储账户管理、借还钱管理和收支等模块组成。用户注册账号，注册成功后登录系统，用户可以在系统中修改在自己的密码，对个人账户管理、借还钱记录和收支记录等进行增删查改，记录并管理财务。

## 2.2 系统功能需求分析

开发的财务管理系统只有用户角色，各用户之间互不相通，提高了用户账户信息的安全性。个人用户所具备的功能如下。

1.注册功能。用户输入用户名，密码，真实姓名，头像等相关信息进行注册，注册成功后即可登录。

2.登录功能。如果用户名和密码填写正确就可以正常的进入财务管理系统，否则系统会提示用户名或者密码填写错误。

3.系统管理功能。用户登录之后可以切换用户注销用户。

4.系统维护功能。如何想更换密码可以在此模块下修改密码。

5.实用工具功能。当用户需要使用到计算器等工具可以点击此模块下的计算器工具。

6.存储账户管理功能。用户可以在此页面对个人银行账户信息进行增删查改操作。

7.借还钱管理功能。当用户借用、或借出钱财时，可以在此模块进行记录。

8.收支管理功能。用户钱财可能会有其他方式流入、流出，可以记录在此模块。

用户的用例图如图2-1所示：

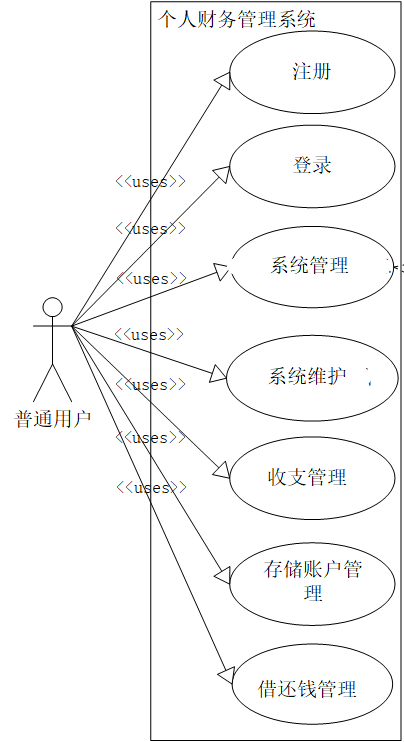


图2-1 用户用例图

其中用户主要功能用例描述如表2-1、表2-2所示：

存储账户管理功能用例主要是介绍用户在财务管理系统中对账户管理功能如何操作，进行借入、借出的增删查改。

表2-1 用户存储账户管理用例描述

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **用例编号** | 1 | | **用例名称** | 存储账户管理 | **参与者** | 用户 |
| **用例概述** | | 用户对个人存储账户进行增删改查 | | | | |
| **前置条件** | | 用户成功进入财务管理系统 | | | | |
| **成功保证** | | 数据库更改成功，财务管理系统能够正常运行 | | | | |
| **事件流** | | 1．进入财务管理系统  2．系统跳转到主界面  3．在菜单栏中选择存储账户管理，即可对账户进行操作 | | | | |

借还钱管理功能用例主要是介绍用户在财务管理系统对借还钱功能中如何操作，进行借入、借出的增删查改。

表2-2 借还钱管理功能用例描述

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **用例编号** | 2 | | **用例名称** | 借还钱管理 | **参与者** | 用户 |
| **用例概述** | | 对用户借还钱进行操作 | | | | |
| **前置条件** | | 用户进入财务管理系统首页 | | | | |
| **成功保证** | | 数据库更改成功，财务管理系统能够正常运行 | | | | |
| **事件流** | | 1．打开系统首页  2．系统根据数据库信息，保证合法用户登录系统  3．系统跳转到主界面  4．用户根据菜单栏，找到借还钱管理按钮  5．用户在借还钱管理界面进行操作 | | | | |

## 2.3 系统性能需求分析

为了保证个人财务管理系统在运行过程中，不会发生故障，并且用户能够在系统中进行基本的操作，共享仓库系统需要达到以下三个性能的需求：

1．快速性

快速性指的是用户在使用系统时，系统的响应速度，本系统采用复合窗口形式，既优化了界面，又改善系统的交互功能，使用户在最快的时间内获得所需的信息。

3.可用性

可用性指的是系统符合大多数用户的需求，能给用户带来舒适的体验效果，在用户不了解系统功能的情况下，也能进行操作，并对用户的失误操作，给予恰当的提示。

4.保密性

保密性指的是系统中所有的信息，都不会对外界公布，当用户在登录系统时，系统需要验证用户的合法性，只有合法的用户才能登录系统，并且用户之前相互独立，这样才能保护用户的隐私，让用户更放心的使用本系统。

## 2.4 本章小结

本章对开发的个人财务管理系统进行了需求分析，首先描述了系统的业务流程，然后对用户主要功能作出了详细的分析，并在最后对系统的性能需求作了研究。

3 系统总体设计

总体设计阶段的意义在于将系统所具有的功能划分成不同的模块，了解每个模块的用途，并用有效的代码将每个模块的功能具体实现出来。

## 3.1 系统功能结构图

本次开发的个人账目管理系统由系统管理、系统维护、数据管理、存储账户管理、借还钱管理和收支等模块组成。

系统功能模块结构图如图3-1所示

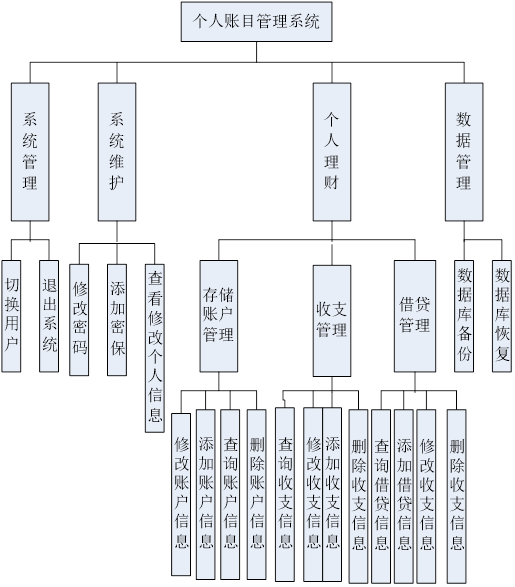


图3-1系统功能结构模块图

## 3.2 系统主要界面设计

界面设计的目的在于使用户更加简单、直观的使用系统。系统由三个主要的界面组成，分别是注册界面、主界面和账户管理界面。下面对这三个界面的页面布局进行详细的介绍。

### 3.2.1 注册界面设计

设计思想：管理员进入注册页面之后根据提示输入相应的用户名和密码之后，系统将数据提交到数据库进行匹配，如果不存在相同的用户名，那么提示注册成功。

注册界面设计图如图3-2所示，注册界面控件说明表如表3-2所示：



图3-2 注册界面

表3-1 注册界面控件说明表

|  |  |
| --- | --- |
| **界面部分** | **控件说明** |
| **头像** | 使用pictureBox控件 |
| **文本输入框** | 用textbox作为用户ID、姓名、密码、确认密码的文本输入框，密码和确认密码的PasswordChar属性为\* |
| **注册按钮** | 用button作为注册、重置、取消按钮 |

该界面特色功能有上传头像、关键字提示、密码字符隐藏等，其中上传头像思路是将头像转化为二进制文件存入数据库中，再通风查询语句取出，显示在pictureBox控件上，其实现代码如下：

private byte[] GetImageBytes(Image image)

{

MemoryStream mstream = new MemoryStream();

image.Save(mstream, System.Drawing.Imaging.ImageFormat.Jpeg);

byte[] byteData = new Byte[mstream.Length];

mstream.Position = 0;

mstream.Read(byteData, 0, byteData.Length);

mstream.Close();

return byteData;

}

private void button2\_Click(object sender, EventArgs e)

{

var open = new OpenFileDialog { Filter = "常见图片|\*.jpg;\*.gif;\*.png;\*.bmp|全部文件|\*.\*" };

if (open.ShowDialog() == DialogResult.OK)

{

pictureBox2.Image = Image.FromFile(open.FileName);

}

}

### 3.2.2 主界面设计

设计思想：主界面作为用户操作的主要界面，一定要让用户感到界面舒适、操作简单。基于此，使用美丽的背景图片、显示头像、音乐播放、实时显示用户等功能使用户可以听着音乐进行操作。

主界面设计图如图3-3所示，主界面控件说明表如表3-2所示：



图3-3 主界面

表3-2 主界面控件说明表

|  |  |
| --- | --- |
| **界面部分** | **控件说明** |
| **头像** | 使用pictureBox控件 |
| **菜单栏** | 使用menuStrip控件，标签显示菜单栏中每个列表的信息 |
| **音乐播放** | 使用axWindowsMediaPlayer控件，添加本地音频文件 |

该界面特色功能有菜单栏、音乐播放、复合窗体、调用系统计算器等，复合窗体思路是使用panel控件将另一个窗体嵌套在主界面中，其实现代码如下：

private void 收支管理ToolStripMenuItem\_Click\_1(object sender, EventArgs e)

{

this.Text = "主界面-[借贷管理]";

panel1.Controls.Clear();

Borrow gmFrm = new Borrow();

gmFrm.FormBorderStyle = FormBorderStyle.None;

gmFrm.TopLevel = false;

this.panel1.Controls.Add(gmFrm);

gmFrm.Show();

}

private void 普通计算器ToolStripMenuItem\_Click(object sender, EventArgs e)

{

Process.Start(Application.StartupPath + Path.DirectorySeparatorChar + "Cal" + Path.DirectorySeparatorChar + "WinCal.exe");

Process.start("calc.exe");

System.Diagnostics.ProcessStartInfo Info = new System.Diagnostics.ProcessStartInfo();

Info.FileName = "calc.exe ";

System.Diagnostics.Process Proc = System.Diagnostics.Process.Start(Info);

}

### 3.2.3 账户管理界面设计

设计思想：账户管理界面应具有账户信息的增删改查功能，在实现这些功能的同时，一定要优化用户体验。使用图片菜单栏是操作更方便，查询优化。

账户管理界面设计图如图3-4所示，账户管理界面控件说明表如表3-3所示：

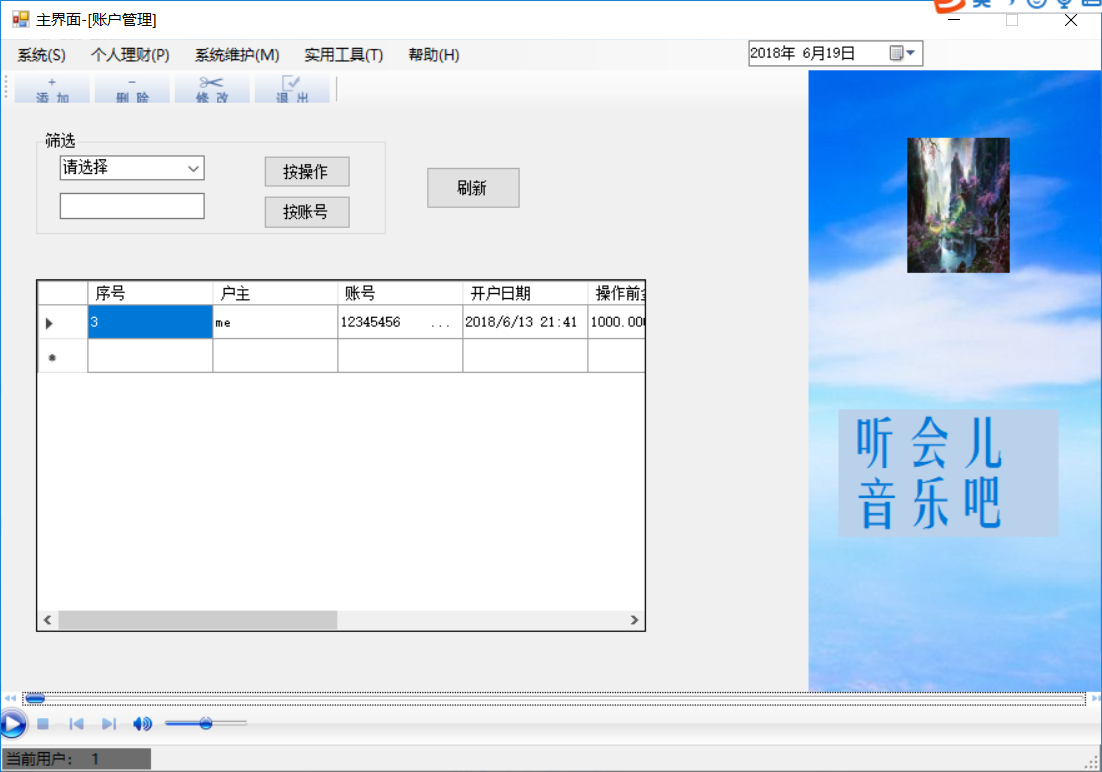


图3-4 账户管理界面

表3-3账户管理界面控件说明表

|  |  |
| --- | --- |
| **界面部分** | **控件说明** |
| **菜单** | 使用toolStrip控件，并设button控件为图片 |
| **数据显示** | 使用DataGridViewRow 控件连接数据库，并显示 |
| **用户显示** | 使用toolStripStatusLabel控件，实时显示当前账户 |
| **头像** | 使用pictureBox控件 |
| **音乐播放** | 使用axWindowsMediaPlayer控件，添加本地音频文件 |

该界面特色功能有图片菜单栏、固定显示用户名等，图片菜单栏设计思路是使用过toolStrip控件并设图片为button控件显示面板。

## 3.3 数据库设计

报表分析功能能让用户随时可以了解自己的财务状况。我们经常看到企业的一些财务报表，这些报表主要是关注企业的资产负债表和损益表。简答的说，前者是反映这个企业有多少钱，是存量；后者，反映了企业在一定时间能赚到多少钱，是流量。

数据库概念结构设计是专业人员将用户的实际需求进行统计和整理之后，形成一个独立的概念模型。

系统数据库名为MyMoney

1.用户信息表(UserInformantion);

2.存储账户信息表(CaInformation);

3.借出信息表(借出信息);

4.借入信息表(借入信息)

数据表设计是数据库设计的核心，各个数据表相互联系起来构成了数据库。财务管理系统数据库中的关系表如表3-4至3-7所示。

用户信息表主要用于保存用户的ID、用户名、密码、头像等信息

表3-4 用户信息表(UserInformantion)的数据结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 描述 | 长度 | 允许空 |
| UserId | char | 用户名 | 6 |  |
| UserName | char | 姓名 | 20 | √ |
| UserPwd | char | 密码 | 6 | √ |
| Images | image | 头像 | 16 | √ |
| PwdIn | Char(10) | 密保 | 10 | √ |

账户信息表主要用于保存账户的户主、账号、开户时间、操作前金额、操作、余额、是否使用、说明等信息。

表3-5 账户信息表(CaInformation)的数据结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 描述 | 长度 | 允许空 |
| 序号 | int | 序号(主键) | 4 |  |
| 户主 | nchar | 户主 | 10 |  |
| 帐号 | nchar | 帐号（唯一） | 20 |  |
| 开户日期 | datetime | 开户日期 | 8 | √ |
| 开户银行 | nchar | 开户银行 | 10 | √ |
| 币种 | nchar | 币种 | 10 | √ |
| 操作前余额 | Money | 操作前余额 | 8 | √ |
| 操作 | Char | 操作 | 10 | √ |
| 操作时间 | Datetime | 操作时间 | 8 | √ |
| 余额 | Money | 操作后的余额 | 8 | √ |
| userId | Char | 用户名 | 6 | √ |
| 是否使用 | Char | 帐号是否有效 | 4 | √ |
| 说明 | Char | 帐号信息说明 | 50 | √ |

借出信息表主要用于保存用户借出的金额、借出时间、借款人、还款人、收款日期、是否回收等信息。

表3-6 借出信息表(借出信息)的数据结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 描述 | 长度 | 允许空 |
| 序号 | int | 序号（主键） | 4 |  |
| 借出金额 | Money | 借出金额 | 8 | √ |
| 借出时间 | Datetime | 借出时间 | 8 | √ |
| 借款人 | Char | 借款人 | 50 | √ |
| 利息 | Float | 利息 | 8 | √ |
| 还款人 | Char | 还款人 | 20 | √ |
| 收款日期 | Datetime | 收款日期 | 8 | √ |
| 是否收回 | char | 是否收回 | 4 | √ |
| userId | Char | 用户名 | 6 |  |
| 说明 | Char | 借出信息说明 | 20 | √ |

借出信息表主要用于保存用户借入的金额、借出时间、借款人、还款人、还款日期、是否还款、说明等信息。

表3-7 借入信息表(借入信息)的数据结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 描述 | 长度 | 允许空 |
| 序号 | int | 序号（主键） | 4 |  |
| 借入金额 | Money | 借入金额 | 8 | √ |
| 借入日期 | Datetime | 借入时间 | 8 | √ |
| 贷款人 | Char | 贷款人 | 50 | √ |
| 利息 | Float | 利息 | 8 | √ |
| 还款人 | Char | 还款人 | 20 | √ |
| 是否还款 | char | 是否还款 | 4 | √ |
| userId | Char | 用户名 | 6 |  |
| 说明 | Char | 借入信息说明 | 20 | √ |

数据表之间的视图关系（如图3-5所示）

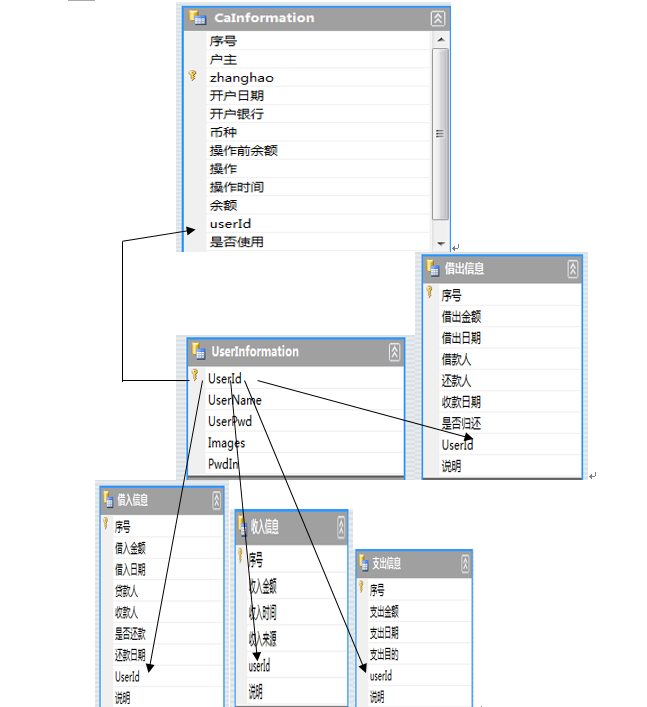


图3-5 表之间的视图关系

## 3.4 本章小结

本章首先对个人财务管理系统的功能作出整体的设计，并确定系统的功能结构图，然后根据功能结构图，确定每个界面的具体功能，最后对系统的数据库进行详细的设计。

4 结束语

## 4.1 总结

本系统作为.NET 的大作业，根据自己平时的记账信息编写，还有一些不完善的，还没有考虑到的地方。

本系统的使用还需用户有一定的操作能力，会进行手动的存储账户、借入借出、收支等的数据信息，根据所提供的功能进行查询和修改

## 4.2 展望

通过本次的课程设计，让我学会了把书本上的知识用到了实际中来。虽然在这一周中一直沉浸在课程设计中，有的问题一直到了最后才被解决，虽然现在还留有一个问题，但是我认为这未必就不是好事，它的存在激发了我继续深入学习c#开发程序的热情，我会更加努力的去探索，去深入的剖析。这样能锻炼我的意志，磨练我的耐心，失败是成功之母，这话一点都没错，没有失败就没有成功。让我没有失去信心的是关怀我们的老师，当我们有了问题和疑问，老师就很耐心的给予讲解，让我们有了一个良好的学习氛围。当然同学之间的互相帮助也是不容护忽视的，同时也证实了那句话“团结就是力量”。

还有让我学会了利用c#的windows程序可以做出这样的小系统，以前只会用c#编写控制台程序，现在让我懂得了如何利用c#来编写窗体代码，我很开心。以后会做得更好

回顾此次课程设计感觉收获很大，不仅巩固了以前所学的知识，而且学到了很多书本上学不到的东西。通过这次课程设计使我们懂得了理论和实际相结合的重要性，只有把理论与实际相结合，从理论中得出结论，才能真正为社会服务。

附 录

上传头像的核心代码：

private void btn\_upload\_Click(object sender, EventArgs e)

{

var open = new OpenFileDialog { Filter = "常见图片|\*.jpg;\*.gif;\*.png;\*.bmp|全部文件|\*.\*" };

if (open.ShowDialog() == DialogResult.OK)

{

pictureBox1.Image = Image.FromFile(open.FileName);

insert(pictureBox1.Image);

}

}

private byte[] GetImageBytes(Image image)

{

MemoryStream mstream = new MemoryStream();

image.Save(mstream, System.Drawing.Imaging.ImageFormat.Jpeg);

byte[] byteData = new Byte[mstream.Length];

mstream.Position = 0;

mstream.Read(byteData, 0, byteData.Length);

mstream.Close();

return byteData;

}

void insert(Image image)

{

byte[] imageBytes = GetImageBytes(image);

string connStr = @"Data Source=DESKTOP-KDKK51O\SQLEXPRESS;Initial catalog=student;integrated Security=True";

using (SqlConnection conn = new SqlConnection(connStr))

{

string sql = "Insert Into src(id,username,password,images) Values('"+textBox1.Text+"','"+textBox2.Text+"','"+textBox3.Text+"',@ImgData) ";

using (SqlCommand cmd = new SqlCommand(sql))

{

SqlParameter param = new SqlParameter("ImgData", SqlDbType.VarBinary, imageBytes.Length);

param.Value = imageBytes;

cmd.Parameters.Add(param);

cmd.Connection = conn;

conn.Open();

int i = cmd.ExecuteNonQuery();

}

}

}

使用panel实现组合窗体及登录代码：

private void 账户管理ToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Text = "主界面-[账户管理]";

panel1.Controls.Clear();

AccountManage gmFrm = new AccountManage();

gmFrm.FormBorderStyle = FormBorderStyle.None;

gmFrm.TopLevel = false;

this.panel1.Controls.Add(gmFrm);

gmFrm.Show();

}

private void btn\_ok\_Click(object sender, EventArgs e)

{

SqlConnection conn = new SqlConnection(constr);

conn.Open();

SqlDataAdapter sda = new SqlDataAdapter();

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "select \* from src where id='" + textBox1.Text + "'";

cmd.Connection = conn;

SqlDataReader reader = cmd.ExecuteReader();

while (reader.Read())

{

string username = reader["id"].ToString();

string password = reader["password"].ToString();

name = username;

pwd = password;

if (this.textBox1.Text.Trim() == username.Trim() && password.Trim() == this.textBox2.Text.Trim())

{

Form\_Main f1 = new Form\_Main();

this.Hide();

f1.Show();

}

}

}

连接数据库代码：

private void Form\_Main\_Load(object sender, EventArgs e)

{

OpenFileDialog ofd = new OpenFileDialog();

ofd.Multiselect = true;

ofd.InitialDirectory = @"C:\Users\kubo\Documents\Visual Studio 2012\Projects\WindowsFormsApplication3\WindowsFormsApplication3\胡夏 - 同桌的你.mp3";

this.toolStripStatusLabel2.Text = Form\_login.name;

string constr = @"Data Source=DESKTOP-KDKK51O\SQLEXPRESS;Initial catalog=student;integrated Security=True";

SqlConnection conn = new SqlConnection(constr);

conn.Open();//打开连接

//SqlDataAdapter sda = new SqlDataAdapter();

SqlCommand cmd = new SqlCommand();

cmd.CommandText = "select \* from src where id='" + Form\_login.name + "'";

cmd.Connection = conn;

SqlDataReader reader = cmd.ExecuteReader();

byte[] imagebytes = null;

while (reader.Read())

{

imagebytes = (byte[])reader["images"];

}

MemoryStream ms = new MemoryStream(imagebytes);

Bitmap bmpt = new Bitmap(ms);

pictureBox2.Image = bmpt;

}

参考文献

[1] 王东明等．Visual C# .NET程序设计与应用开发．北京：清华大学出版社，2008.

[2] 李春葆等.C#程序设计教程（第3版）.北京：大学出版社，2015.

问 题

1、如何使用 panel控件？

答案：通过panel控件，在主界面构成复合窗体，使界面更加直观、简介。使用panel实现组合窗体及登录代码：

private void 账户管理ToolStripMenuItem\_Click(object sender, EventArgs e)

{

this.Text = "主界面-[账户管理]";

panel1.Controls.Clear();

AccountManage gmFrm = new AccountManage();

gmFrm.FormBorderStyle = FormBorderStyle.None;

gmFrm.TopLevel = false;

this.panel1.Controls.Add(gmFrm);

gmFrm.Show();

}

2、如何上传头像？

答案: 添加上传头像功能，把头像信息以二进制形式存入数据库中，并根据用户ID信息动态显示在主界面。上传头像的核心代码：

private void btn\_upload\_Click(object sender, EventArgs e)

{

var open = new OpenFileDialog { Filter = "常见图片|\*.jpg;\*.gif;\*.png;\*.bmp|全部文件|\*.\*" };

if (open.ShowDialog() == DialogResult.OK)

{

pictureBox1.Image = Image.FromFile(open.FileName);

//insert(pictureBox1.Image);

}

}

3、音乐播放器如何实现？

答案：通过在工具连添加WindowsMediaPlayer控件，然后在面板中添加此控件，再添加本地音频文件即可。

**致 谢**

从课题选取、方案论证到具体设计和调试，无不凝聚着心血和汗水，在本学期的学习和生活期间，也始终感受着老师的精心指导和无私的关怀，我受益匪浅。

程序设计是在李永明老师的精心的指导下，我进行了大胆的理论与实践的结合。通过这次设计，我的动手能力得到了很大的提高，使我学到了怎样去查找翻阅有关理论资料和技术手册，怎样把理论知识与实践相结合，使我对做一个基本的软件具备了最基础的知识，肯定会对今后的工作产生深远的影响。

在此，我表示衷心的感谢和崇高的敬意！