

重庆交通大学信息科学与工程学院

课 程 设 计 报 告

专 业： 计算机科学与技术

班 级： 2014 级 1 班

学 号： 631406010109

姓 名： 郭 文 浩

课 程 名 称： 数据库及应用课程设计

实验室(中心)： 软件与信息技术实验室

指 导 教 师： 王 家 伟

实验完成时间： 2017 年 1 月 5 日

此页必须保留

重庆交通学院计算机与信息学院课程设计任务书

课 程	数据库及应用课程设 计	班级	计算机科学与技术 2014.1		指导教师	王家伟		
题 目	个人财务管理系统		完成时间	2016 年 12 月 12 日至 2017 年 1 月 7 日				
内 容 主 要	题目具体内容见附件							
设计 报告 要求	1、选定题目的需求分析； 2、分析选定题目的软件功能结构； 3、选定题目的数据库设计（概念模型、逻辑模型和物理模型）； 4、软件代码设计； 5、总结； 6、根据以上要求完成课程设计报告（报告除代码以外，至少 2000 字）。 每个题目要求代码以附件的方式打印出来。							
版面 要求	1. 题目用黑体三号，段后距 18 磅（或 1 行），居中对齐； 2. 标题用黑体四号，段前、段后距 6 磅（或 0.3 行）； 3. 正文用小四号宋体，行距为 1.25 倍行距； 4. 标题按“一”、“(一)”、“1”、“(1)”顺序编号。							
指导 时间 安排	星期	一	二	三	四	五	六	日
	周次							
	14	7-8				5-6		
	15	7-8				5-6		
	16	7-8				5-6		
	17	7-8				5-6		
18	7-8				5-6			
附加要 求	1、题目可以从给定的题目中选择，也可以自选。 2、每个题目可以独立也可以分组完成，但每个小组的人数不能超过 3 人。 3、设计工具自选，但每个题目必须用高级语言实现。 4、每位同学都必须有独立的设计报告且相同的内容不能超过 20%。 5、上交设计内容时，包含设计报告（打印件）和设计成果的内容。 6、最终成绩评定：见评分标准。 7、于 18 周星期五交设计内容。							
指导 地点	软件与信息技术实验室							

此页必须保留

评分标准及成绩

评分细则		评分	
作品演示（40 分）			
设计报告（40 分）			
平时考勤（20 分）			
总成绩			
教师签名		日期	

（1）设计成果评分细则

设计成评分采用答辩方式

序号	评分细则	评分
1	工作量（10 分）	
2	设计难度（10 分）	
3	界面友好性（10 分）	
3	操作方便性（20 分）	
4	功能正确性（40 分）	
5	功能丰富性（10 分）	

（2）设计报告评定细则

序号	评分细则	评分
1	问题描述（5 分）	
2	设计思想（10 分）	
3	具体设计的正确性（70 分）	
4	组内是否前后一致（5 分）	
5	其他（10 分）	

此页必须保留

实 验 要 求

一 总体要求

1 排版要求

- (1). 题目用黑体三号，段后距 18 磅（或 1 行），居中对齐；
- (2). 标题用黑体四号，段前、段后距 6 磅（或 0.3 行）；
- (3). 正文用小四号宋体，行距为 1.25 倍行距；
- (4). 报告必须生成目录，目录必须带页码。

2 设计报告需要打印，装订，1 份/人。

3 学习委员收集每位同学做的成果，包含数据库的备份、程序源代码和运行配置说明，并将每位同学的成果拷贝到软件与通信实验室的教师机上。

二 成绩评定要求

主要由作品演示、设计报告和平时成绩三个部分组成。

- 1 作品演示 40 分，本次作品演示考核以组为单位分组考核，考核分组根据所选题目进行分组；每个小组必须做出本小组所做内容的 ppt，由组长演示整体内容，每个成员演示自己做得内容。
- 2 设计报告 40 分；
- 3 平时考勤 20 分。

目 录

一 绪论	5
(一) 系统简介	5
(二) 设计目的	5
(三) 设计内容	5
二 需求分析	6
(一) 需求分析说明	6
(二) 功能需求分析	6
(三) 总体设计	6
(四) 模块详细设计	8
三 数据库设计	9
(一) 概念模型的设计	9
(二) 逻辑模型的设计	10
(三) 转换成的物理模型	11
(四) PowerDesigner 转换为的 sql 的部分语句	12
(五) 插入部分语句的部分截图	12
四 软件代码设计	13
(一) 注册登录模块设计	13
(二) 个人信息模块的设计	14
(三) 收支模块设计	14
(四) 账户安全模块	15
五 心得体会	16

个人财务管理系统

一 绪论

（一）系统简介

一个有经济头脑的人，在日常生活和投资中，都会对自己的支出和开销做到心里有数，以便于制定出自己的一套理财计划，达到合理的经济统筹，有理财习惯的人可以从以往的统计中看到自己主要消费方向，看出自己的消费是否合理，由此制定出一套合理的经济计划方案，从而可以不断的长进自己的经济头脑，对自己的财富积累，合理投资有着不可或缺的帮助。

从查阅资料来看，一些比较发达国家人的生活中对理财有着很严格的计划，因此国外的理财系统总体来说比较普遍，做的也比较完善。国内的理财系统主要偏重于办公财务软件，而个人理财方面的比较少，因此需要开发相关的系统软件。

本个人财务管理系统的优点是：通过系统的数据录入，可以将数据保存在相应的数据库里面，而数据库的设计方式，可以使录入的数据更加有效，更有结构化，合理。

对数据的统计有比较完善的设计，对各种输入错误的情况有着比较合理的考虑，通过代码提示用户输入完整，合理的数据，以便于查找，统计。

本文将介绍基于 C/S 模式的个人理财系统的设计和开发。因此，文章将涵盖以下内容：介绍贯穿项目设计，开发始终的 GUI+JDBC 技术；介绍系统的详细设计以及具体实现的过程。

（二）设计目的

具体对本系统来说，个人财务管理系统数据库设计目的主要有以下 3 个：

- 1、为用户提供更便捷的操作方式，可以将用户的信息做长期的存储；
- 2、对用户的收入、支出进行统计，让用户实时掌握自己的经济情况；
- 3、对用户的数据安全要做保护，绝对不能出现信息泄露的现象。

（三）设计内容

此次课程设计的主要内容就是对个人财务管理系统建立一个数据库模型，并作出一个真实存在的系统。包括：

- 1、数据库系统需求分析；
- 2、数据库概念模型设计；
- 3、数据库逻辑模型实现；
- 4、数据库物理模型实现。
- 5、系统存储数据的实现。
- 6、系统维护数据的实现。

二 需求分析

（一）需求分析说明

进行整个数据库设计必须准确了解与分析用户需求（包括数据和处理）。需求分析是整个设计过程的基础，是最困难、最耗费时间的一步。作为地基的需求分析是否做得充分与准确，决定了在其上构建数据库大厦的速度和质量。需求分析的任务是通过详细调查现实世界要处理的对象（组织、部门、企业等），充分了解原系统（手工系统或计算机系统）工作概况，明确用户的各种需求，然后在此基础上确定新系统的功能。

需求分析调查的重点是“数据”和“处理”，通过调查、收集和分析，获得用户对数据库的如下需求：

1、信息需求。指用户需要从数据库中获得信息的内容与性质。由信息要求可以导出数据要求，即在数据库中需要存储哪些数据。

2、处理要求。指用户需要完成什么处理功能。明确用户对数据有什么样的处理要求，从而确定数据之间的相互关系。

3、安全性与完整性要求。

（二）功能需求分析

1、服务器目前由于安全原因，先以本机作为服务器，数据不对外，存在本机中可以对数据做一定的保护。

2、账户安全功能。这部分一定要加密，对用户的密码加密，然后整个系统如果没有账户密码无法访问，从而实现数据的隐蔽性，加密算法暂时采用为当前流行的 MD5 加密，虽然这个算法已经被公开了，不过还是能起到一定的加密作用，总之最重要的一点，不能有明文数据流、明文存储。

3、个人信息管理。这部分不是本系统的重点，只要能够存储一定的信息就好。反过来想一想，如果我是用户，看到一个系统，如果他要我很详细的信息，那我一定不会去用，不管他的系统多牛逼，我都会担心自己的信息会泄露。

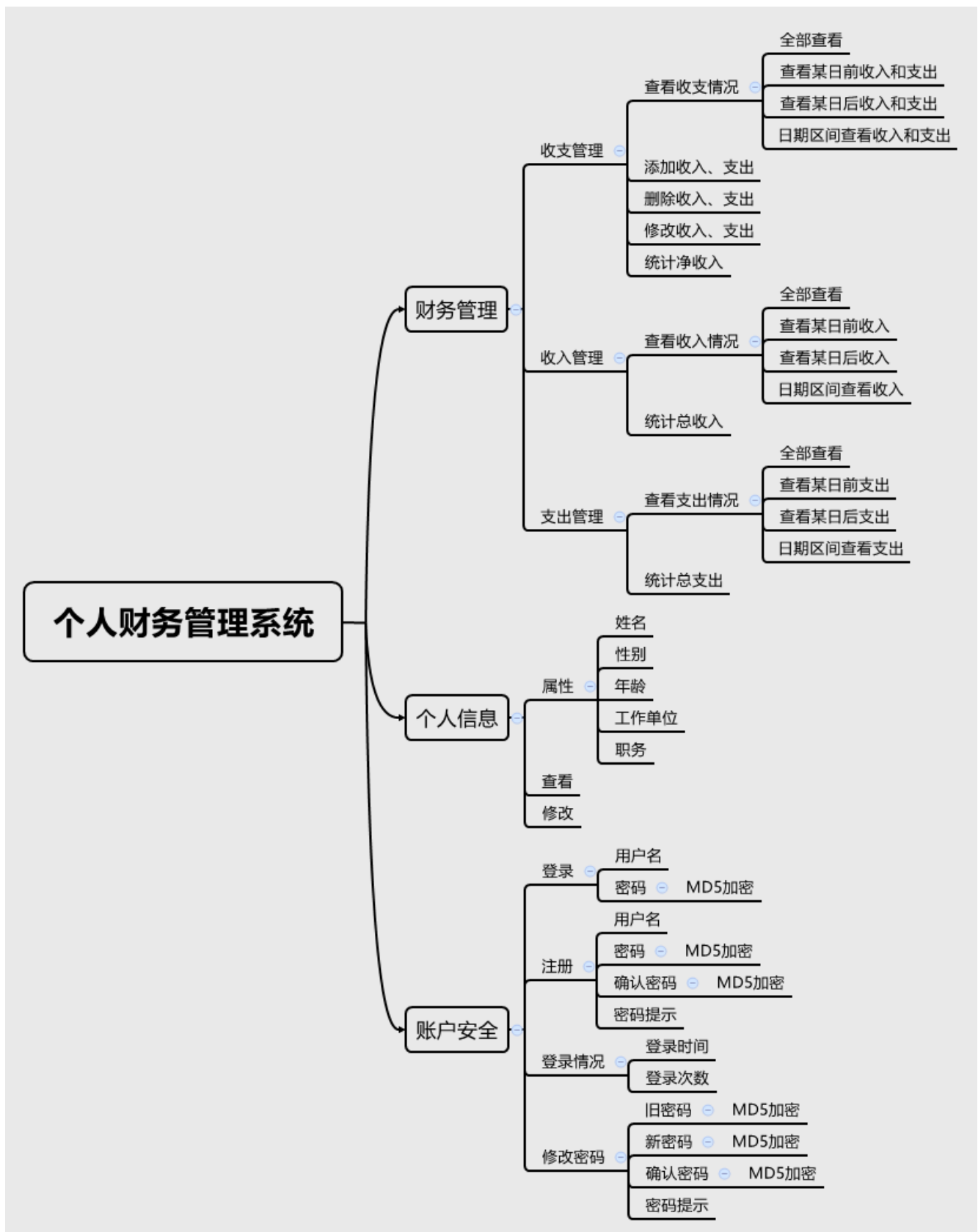
4、财务管理。此部分为本系统的重点，要实现对收入、支出的记录，并且可以对净收入、总收入、总支出进行统计，而且还要实现时间区间的查询、统计，以及对收支信息的修改、删除等等操作。

（三）总体设计

从系统的功能上大体划分为三大模块：财务模块、安全模块、个人信息模块。

财务模块应该包含收入、支出、收支三个小模块，小模块下要实现信息的查询修改，对金额的统计等等。安全模块主要涉及登录、注册、登录记录以及密码的修改。个人信息模块就是简单的添加修改就好。

根据分析，个人财务管理系统的功能图：

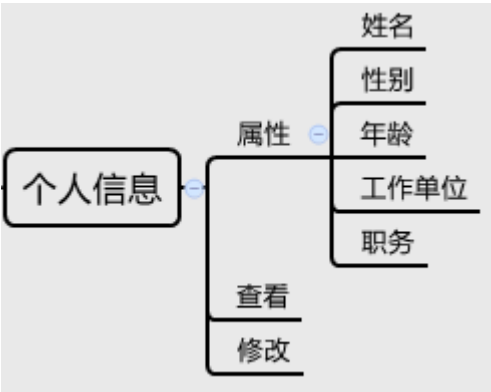


（四）模块详细设计

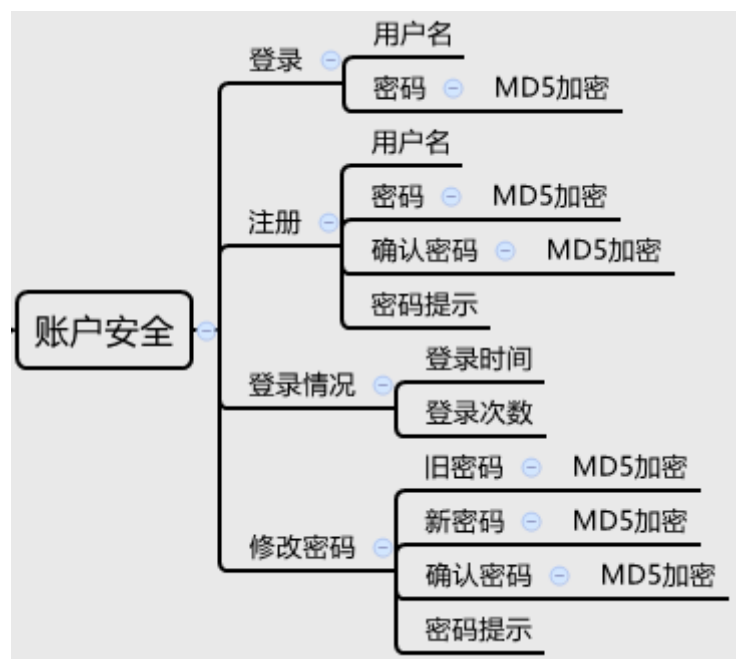
1、财务模块，包含收支管理、收入管理、支出管理。如图：



2、个人信息模块，包含有个人信息的属性、查看功能、修改功能。属性包括姓名、性别、年龄、工作单位、职位。



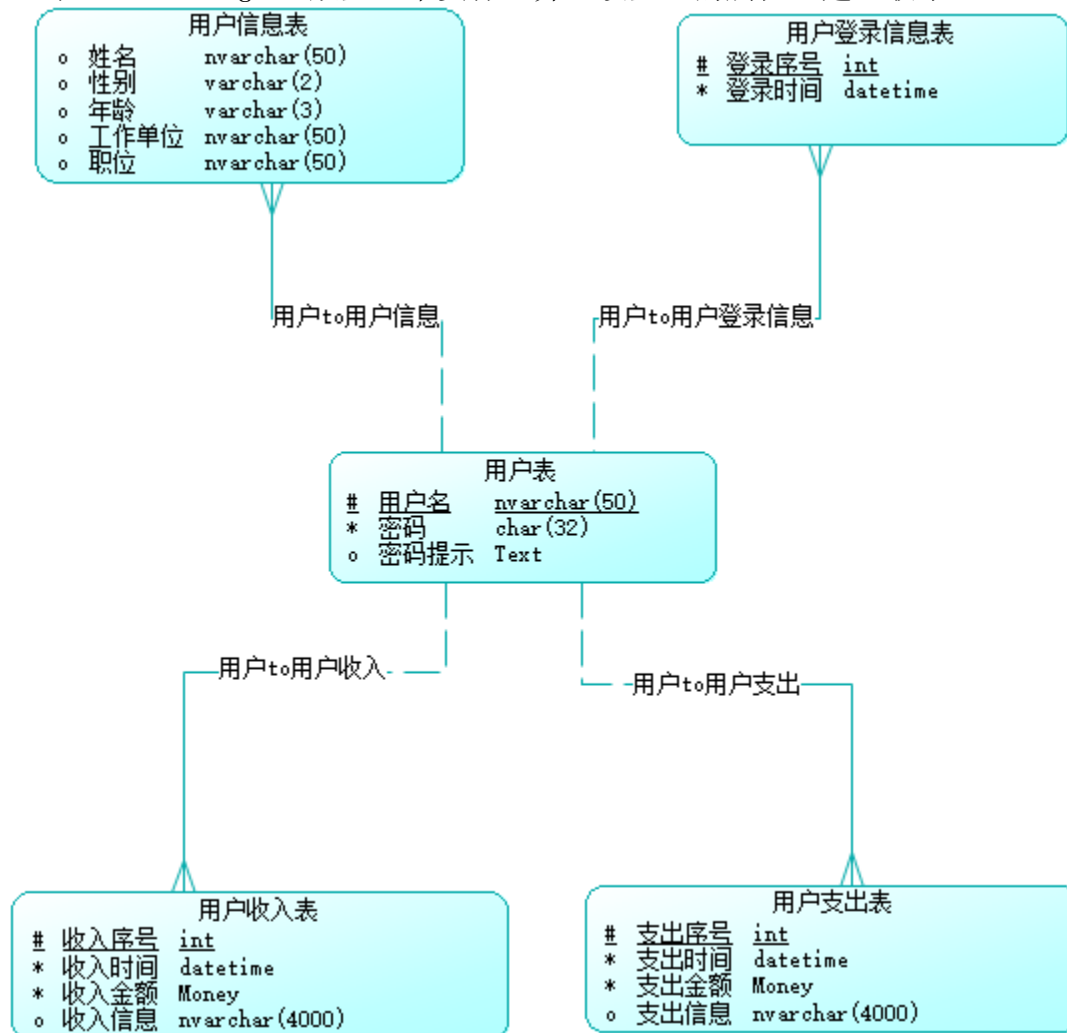
3、账户安全，包含登录、注册、查看登录情况、修改密码等功能，其中涉及到密码的均用 MD5 加密后，以密文传输、存储。



三 数据库设计

(一) 概念模型的设计

在 PowerDesigner 放置 5 个实体，并且设置它的属性，建立联系。



（二）逻辑模型的设计

1、实体集的转换规则一个实体集转换为关系模型中的一个关系，实体的属性就是关系的属性，实体的码就是关系的码，关系的结构是关系模式。

2、实体集间联系的转换规则

(1) 1:1 联系的转换方法

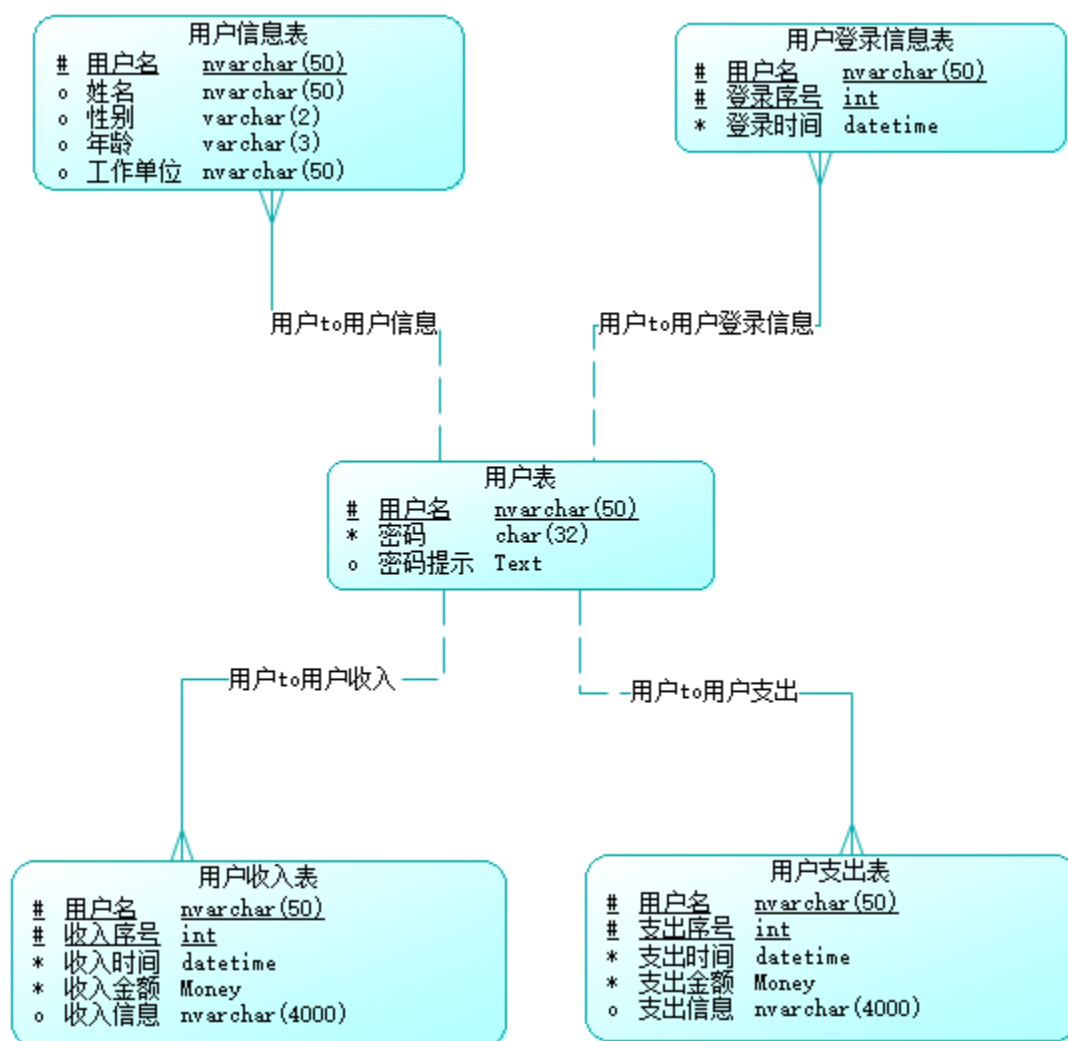
将 1:1 联系转换为一个独立的关系：与该联系相连的各实体的码以及联系本身的属性均转换为关系的属性，且每个实体的码均是该关系的候选码。

(2) 1:n 联系的转换方法

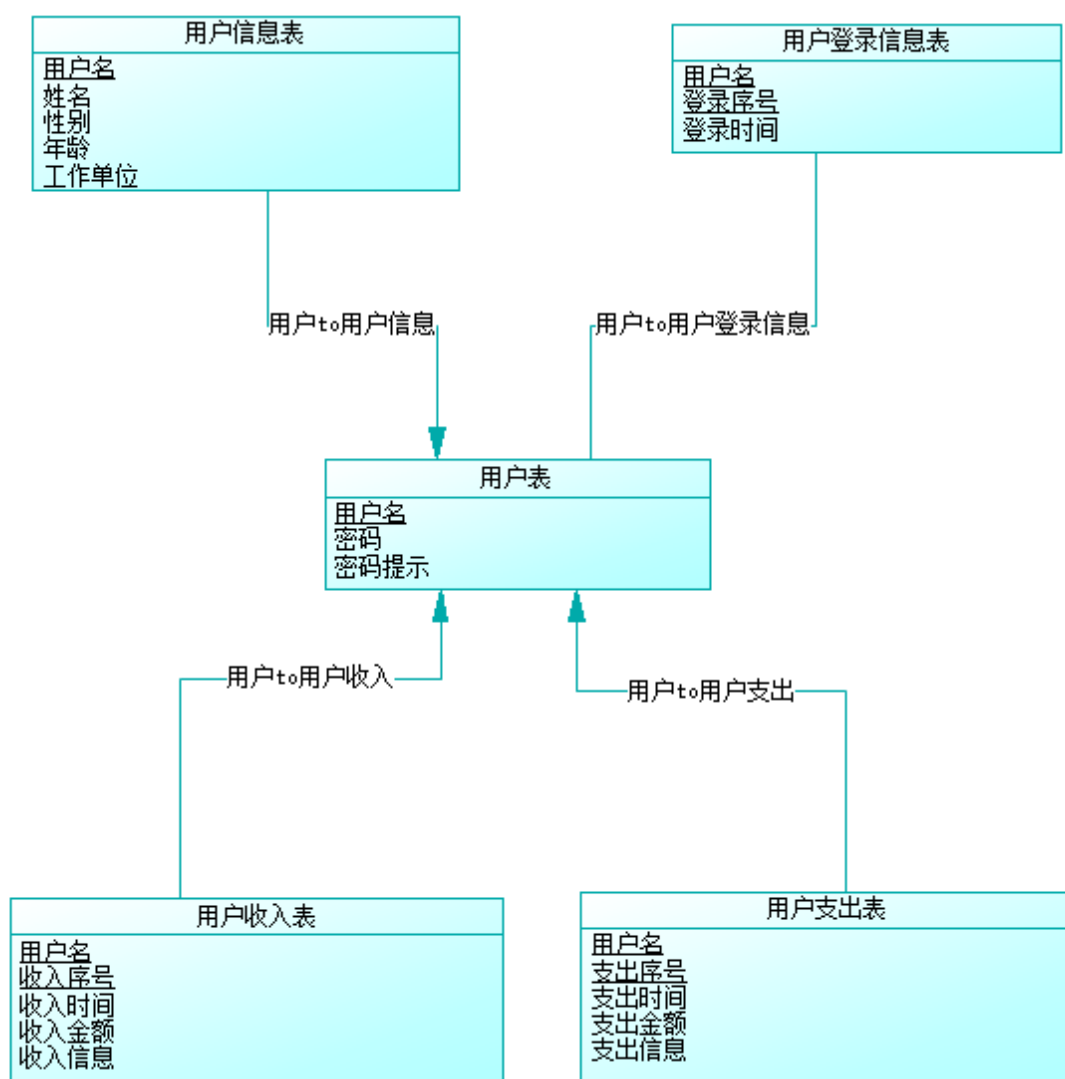
1:n 联系的转换方法是：将一方的主码作为多方的外码。

(3) m:n 联系的转换方法

在向关系模型转换时，一个 m:n 联系转换为一个关系。转换方法为：与该联系相连的各实体集的码以及联系本身的属性均转换为关系的属性，新关系的码为两个相连实体码的组合（该码为多属性构成的组合码）。



（三）转换成的物理模型



(四) PowerDesigner 转换为的 sql 的部分语句

```
PersonalMoneyDB....MoneyDB (sa (52))

/*=====*/
/* Table: T_User */
/*=====*/
create table T_User (
    userName          nvarchar(50)          not null,
    userPwd            char(32)              not null,
    userPwdTiShi       text                  null,
    constraint PK_T_USER primary key nonclustered (userName)
)
go

/*=====*/
/* Table: T_UserInfo */
/*=====*/
create table T_UserInfo (
    userName          nvarchar(50)          not null,
    xm                nvarchar(50)          null,
    xb                varchar(2)            null,
    nl                varchar(3)            null,
    gzdw              nvarchar(50)          null,
    zw                nvarchar(50)          null,
    constraint PK_T_USERINFO primary key nonclustered (userName)
)
go

/*=====*/
/* Table: T_UserLogin */
/*=====*/
```

已连接。 (1/1) GUOWENHAO

(五) 插入部分语句的部分截图

```
SQLTest.sql - GUO...MoneyDB (sa (53)) PersonalMoneyDB....MoneyDB (sa (52))

select * from T_User where userName='郭大神' and userPwd='?????'
select * from T_User
delete from T_User where userName='qqq'
select loginIndex,loginTime from T_UserLogin where userName='郭大神'
select loginIndex,loginTime from T_UserLogin where userName='郭大神'and logi
select loginIndex,loginTime from T_UserLogin where userName='郭大神'and logi
select loginIndex,loginTime from T_UserLogin where userName='郭大神'and logi
select loginIndex,loginTime from T_UserLogin where userName='郭大神'and logi
select loginIndex,loginTime from T_UserLogin where userName='郭大神'and logi
update T_User set userPwd='???' ,userPwdTiShi='111' where userName='郭大神'
select * from T_ZhiChu
select * from T_ShouRu

select ShouRuTime,ShouRuMoney,ShouRuInfo from T_ShouRu where userName='郭大神'
union select ZhiChuTime,ZhiChuMoney=-ZhiChuMoney,ZhiChuInfo from T_ZhiChu
order by ShouRuTime

insert into T_ShouRu values ('郭大神','2016-12-29 13:11:46',999.99,'测试增加收

select ShouRuIndex,ShouRuTime,ShouRuMoney,ShouRuInfo from T_ShouRu where us
union select ZhiChuIndex,ZhiChuTime,ZhiChuMoney=-ZhiChuMoney,ZhiChuInfo
order by ShouRuTime
delete from T_ShouRu where ShouRuIndex=46
delete from T_ZhiChu where ZhiChuIndex=19

update T_ShouRu set ShouRuTime='111',ShouRuMoney=111,ShouRuInfo='111' where
select xm,xb,nl,gzdw,zw from T_UserInfo
```

已连接。 (1/1) GUOWENHAO

四 软件代码设计

（一）注册登录模块设计

1、登录模块

在登陆时，有两种方式，以用户名查询密码，比较密码是否相同，目前市面上基本是采取这种方法，但我觉得这种方法有漏洞，如果是以用户名查询密码，再返回来比较密码的话，这样子如果通过抓包，抓取到了数据流，就很有可能造成密码泄露，如果密码泄露了，后果将十分严重。所以我才用了另一种方法，用户名和密码一起上传查询数据库，看是否有对应的用户，因为用户名唯一，所以用户名对应的密码也唯一，查询出的用户也唯一，当且仅当用户名和密码对应时，登陆成功。登陆成功后，上传当前时间，记录一下。关键代码如下：

```
dbConn = help.GetConnection();
dbState = dbConn.createStatement();
sql = "select * from T_User where userName='" + USERNAME + "' and
userPwd='" + PASSWORD + "'";
dbRs = dbState.executeQuery(sql);
if (dbRs.next()) {
    .....;
    .....;
    .....;
    return true;
} else {
    sql = "select userPwdTiShi from T_User where userName='" + USERNAME
+ "'";
    dbRs = dbState.executeQuery(sql);
    if (dbRs.next()) {
        lable_passworldTiShi.setText("密码提示:" + dbRs.getString(1));
    } else {
        lable_passworldTiShi.setText("密码提示:");
    }
    .....;
    .....;
    .....;
    return false;
}
```

2、注册模块

这部分就是简单的用户注册，注册时需要用户名和密码，当注册条件满足时（用户名不重复，密码和确认密码相同），对密码加密，然后将信息传至数据库。完成注册后自动返回登录页面，用户名自动变成刚才注册的用户名。

3、数据安全。

此部分主要为密码的加密，在（四）-4 中详细讲解。

（二）个人信息模块的设计

1、添加个人信息

此模块主要为第一次登陆的用户提供，用户第一次登陆后，可以添加个人信息，当然用户也有权利不填。

2、修改个人信息

对添加的个人信息可以修改，为了防止误点，加一个信息锁定，每次要修改时，要先解锁才可以开启编辑。

3、细节处理

细节处理主要处理年龄的范围（0-100），性别（男/女）等等。

（三）收支模块设计

1、收支记录

收支记录中要统计净收入哪天之前的，哪天之后的，哪个时间段的，还要包括添加、删除、修改。添加功能要提醒是添加收入还是支出。删除是要提醒一下，是否删除这几行，不能直接删除。修改时一次只能选定一行修改。代码如下：

```
/******添加******/
Object[] options = { "收入记录", "支出记录", "取消" };
int m = JOptionPane.showOptionDialog(null, "请问您要添加收入记录还是支出记录: ", "提示", JOptionPane.YES_NO_OPTION,
JOptionPane.QUESTION_MESSAGE, null, options, options[0]);
if (m == 0) {
    AddShouRu asr = new AddShouRu();
    asr.setVisible(true);
    MainFrame.mf.setVisible(false);
} else if (m == 1) {
    AddZhiChu azc = new AddZhiChu();
    azc.setVisible(true);
    MainFrame.mf.setVisible(false);
} else {
    // 什么都不做
}

/******删除******/
int n = JOptionPane.showConfirmDialog(null, "是否要删除选中的这 " +
selectRows.length + " 行?", "提示",
JOptionPane.YES_NO_OPTION);
if (n == 0) {
    // 删除操作
}
```

```

/*****修改*****/
if (selectRows.length <= 0) {
    JOptionPane.showMessageDialog(null, "请选择要修改的行!", "消息",
        JOptionPane.INFORMATION_MESSAGE);
} else if (selectRows.length > 1) {
    JOptionPane.showMessageDialog(null, "每次只能修改 1 行!", "消息",
        JOptionPane.INFORMATION_MESSAGE);
} else {
    int n = JOptionPane.showConfirmDialog(null, "是否要修改选中的这
        1 行?", "提示", JOptionPane.YES_NO_OPTION);
    if (n == 0) {
        // 修改操作
    }
}

```

2、收入详情

收入详情可以查询哪天之前的，哪天之后的，哪个时间段的收入情况，统计收入总额。

3、支出详情同收入详情。

(四) 账户安全模块

1、登录记录

每当用户登陆成功后，会自动上传一个登录时间，存储在数据库中。这里主要是为了提醒用户，让用户清楚地看到登录情况，及时发现自己有没有异常登录，若有异常登录，可以及时更改密码。保护账号的安全。

2、修改密码

在旧密码正确的情况下修改密码，然后根据用户名来重新上传一个密码，同样要加密。

3、密码提示

一个小功能，在用户登录一次失败后，提示一个小短句，用户之间注册时填写的，可以提醒用户，减少忘记密码的可能。

4、加密

MD5 加密：

```

public static String getMD5(String str) {
    try {
        MessageDigest md5 = MessageDigest.getInstance("MD5");
        md5.update(str.getBytes());
        String s = new BigInteger(1, md5.digest()).toString(16);
        return s;
    } catch (Exception e) {
        System.out.println("MD5加密出现错误\n"+e.toString());
        return "";
    }
}

```


五 心得体会

此次开发数据模型很简单，所以在数据库层面没有花费太多时间，基本上时间都用在了系统的规划中，个人财务系统给我的感觉是我不光要做好财务部分，对安全的处理也要极为严格，一个安全的系统才是可以真正被大众接受的。

这次开发充分感受到了 java 的强大，利用一个游标，可以很方便的获取数据库中的值，查询修改起来十分方便，尽管现在只掌握了很少的一部分技术，但已经被 java 在 jdbc 方面的强大所折服，慢慢来，让自己对 jdbc 使用的更加顺手、熟练。

目前在 java 和 c#中都开发了连接数据库的系统，这两个平台给我的感觉其实并没有太大的差别，可以说机制其实是一样的，但是用起来还是觉得 java 更加便于操作，尽管在 GUI 界面上开发效率和 c#相比极低，但是在真正的数据库方面的操作上还是很有优势的。而且，关键的一点是以后的开发趋势必然是以 B/S 取代 C/S，所以 GUI 的短板我认为不是很关键。

这次开发时间方面处理的还是挺不错的，尽管开发期间都是考试，但是每次都可以抓住零碎的时间，借助开发日志，每次只完成一个小点，慢慢的系统就开发完了。总结一点，在项目开发上一定要做好足够的估计，在时间安排上要结合实际，借助实际开发日志，尽量把时间安排得合理些。