
目录

1.	绪论	1
2.	需求分析	2
2.1	应用背景	2
2.2	开发学生成绩管理的必要性	2
2.3	需求分析目标	3
2.4	系统设计概要	3
2.5	系统可行性分析	4
2.6	系统的目的和意义	4
2.7	系统的业务流程及具体功能	5
2.8	数据流程	7
2.9	系统的数据字典:	9
3.	概念结构设计	13
4.	逻辑结构设计	16
5.	物理结构设计	18
6.	数据库实施	19
7.	数据库运行和调试	24
8.	总结	25

1. 绪论

学生成绩管理系统是整个学校工作系统的重要组成部分，其内容对于学校管理者和学生至关重要，所以学生成绩管理系统应该为用户提供所需要的信息以及方便的管理平台。现代成绩查询系统理论已经不拘泥于传统的理论基础，不强调对教学活动的严密控制，放弃了繁琐的设计模式，开始注重教学设计的关系性和灵活性，并且提供人性化的服务，这些优点能够极大的提高学生成绩查询的效率，从而更加有利于学生的管理和提高学生的主动性。

每个学校都需要在学期末进行期末考试成绩的统计分析和统计工作，而这些工作都必须在考试结束后近一个星期的时间内完成。大量的成绩数据的统计分析和统计工作如果只靠人工来完成，费时费力，还容易出错。随着计算机技术的飞速发展，计算机在日常管理应用中迅速普及，利用计算机进行学生成绩管理势在必行。因此需开发出一个能满足教务处进行成绩的录入，统计，查询等需求的、功能完善、安全可靠、迅速简便的学生成绩管理信息系统。

本系统立足于教务处的角度,面向全校师生以及课程,对全学校的学生和成绩做统一管理，以实现管理的科学化，系统化，自动化，最大限度的为老师和同学们提供方便，节约人力、物力等各方面资源。彻底杜绝以前那种老师工作量大，工作步骤繁琐，学生有问题不知道去哪里解决的现象，而且消灭由于信息冗余而造成的数据不一致和数据查询困难的问题。

2. 需求分析

2.1 应用背景

当今时代是飞速发展的信息时代，在各行各业中离不开信息处理，这正是计算机被广泛应用于信息管理系统的环境。计算机的最大好处在于利用它能够进行信息管理。使用计算机进行信息控制，不仅提高了工作效率，而且大大的提高了其安全性。尤其对于复杂的信息管理，计算机能够充分发挥它的优越性。管理信息系统是进行信息的采集、存储、加工、维护和使用的系统，它是随着管理科学和技术科学的发展而形成的。学生成绩管理系统是一个教育单位不可缺少的部分，它的内容对于学校的决策者和管理者来说都至关重要，所以学生成绩管理系统应该能够为用户提供充足的信息和快捷的查询手段，对学生来说可以轻松的查阅自己在校的成绩以及信息等。但是一直以来人们都是靠传统人工的方式管理学生成绩，这种管理方式存在着许多缺点，如：效率低、保密性差，另外时间一长，将产生大量的文件和数据，这对于查找、更新和维护都带来了不少的困难。

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。作为计算机应用的一部分，使用计算机对学生成绩信息进行管理，有着手工管理所无法比拟的优点.例如：检索迅速、查找方便、可靠性高、存储量大、保密性好、寿命长、成本低等。这些优点能够极大地提高学生成绩管理的效率，也是企业的科学化、正规化管理，与世界接轨的重要条件。因此，开发这样的一套成绩管理系统成为很有必要的事情。

学生成绩管理系统提供了强大的学生成绩管理功能，方便系统管理员对学生成绩等等信息的添加、修改、删除、查询、报表生成等操作，同时一样的方便学生对自己各科成绩查询，学习的交流。

2.2 开发学生成绩管理的必要性

在如今这个信息时代，一个高效的学生成绩管理系统可以储存历届的学生成

绩档案，不需要当来那个的人力，只需要几名专门录入人员即可操作系统，节省了大量人力，可以迅速查到所需的信息，高效，安全，学生可以方便快速的查看自己的成绩。

开发一个学生成绩管理系统，采用计算机对学生成绩进行管理，进一步提高了办学效益和现代化水平。为广大教师和学生提高工作效率，为实现学生成绩信息管理工作流程的系统化、规范化和自动化。

学生成绩管理系统是高校各种信息管理系统中的一种，传统的手工管理方式需要提取日常教学活动中相关记录，易发生数据统计错误、丢失数据、教师劳动强度高且效率低等问题。网络、计算机的普及很大的解决了这些问题，计算机连入网络后，各类数据在网络上快速流通，实现了数据共享，避免重复劳动，教学管理方式规范化，提高了管理效率和高校办学水平。学生成绩管理系统不但满足了高校的管理者、教师及学生对该方面的需求，极大地减少了管理成本、节约了用户时间，而且完善了学生成绩管理方面的数据存储、数据统计、信息检索、信息分析等功能。为了给高校管理学生成绩提供更优良的技术支持，提高教学质量和水平，所以需要研究和开发新的学生管理系统。

2.3 需求分析目标

1. 教师能够实现对整个学生信息的添加、修改、删除、查询等操作，并在此基础上生成符合教师要求的报表。
2. 学生可以在得到相应授权口令的情况下对自己的信息进行查询，便于学生间的相互交流和学习。
3. 教师能够在一定的权限内了解所有学生的基本信息情况。
4. 学生可以在自己的权限内对任课教师、课程编号及课程学分情况进行查询。

2.4 系统设计概要

为了适应现代社会人们高度强烈的时间观念，学生成绩管理系统为教学办公带来了极大的方便。该学生成绩管理系统是以 JAVA 编程+MYSQL 语言编写，

其系统功能通过编程后生成相应的安装程序包，可在任何单机上安装运行。系统管理员、教师、学生只需通过简单的操作，就可以了解本系统软件的基本工作原理。用户只需进行输入一些简单的汉字、数字，或用鼠标点击即可达到自己想要的目标。

2.5 系统可行性分析

可行性分析指的是设计者在开发系统之前，需要分析系统是否具有开发价值，开发的系统具有哪些实用性能。本文仅从技术方面、经济方面和组织方面进行系统可行性分析。

1. 技术可行性

设计者可以根据技术可行性研究选择现有的技术,用其解决在系统设计与实现中遇到的技术问题。天津工业大学学生成绩管理系统选择架构，以 Java 为开发平台，应用面向对象的开发方法，并使用 MySQL 数据库存储数据。这些主流技术及软件，符合本系统的要求。因此，开发本系统在技术上完全可行。

2. 经济可行性

经济可行性是指系统开发过程中可以使用的资源的可能性(资源包括人力资源、自然资源和资金条件)。天津工业大学学生成绩管理系统所需软硬件设备，设计者都可以免费获取，人工成本可忽略不计，开发成本大大减少。本系统开发完成后投入使用，不仅减少纸张的使用，同样降低了管理成本。在经济方面，开发系统期间，不会出现较大问题。

3. 组织可行性

组织可行性是指系统开发中，系统实现与用户需求的匹配度，现有组织制度的完善度。本系统能够实现学生成绩管理中业务流程的信息化、规范化，提高教务人员的准确率和工作效率、管理者管理力度、学生自主学习性。上述这些和用户所提需求基本一致。因此,系统在组织上完全可行。

2.6 系统的目的和意义

实现了学生个人信息查询、修改，教师个人信息查询、修改，学生成绩录

入，学生个人信息录入等基本功能。实现了学生成绩管理的规范化，减少了教学人员的工作量。本系统界面简洁，操作简单，稳定性高，存储量大等优点，能便于学生成绩管理和学生发展。

2.7 系统的业务流程及具体功能

通过对用户需求和系统设计思想的分析，可以得出该学生成绩管理系统大致可以分为几个模块：

1. 管理员的业务流程

首先管理员进入系统需要通过身份验证，验证通过后，才能进入学生个人信息管理、教师信息管理、专业管理、班级管理、学生管理、成绩管理、成绩统计等功能界面，能够进行系统维护、信息更新操作。进行信息管理的查、改、删、增的操作时，将查询到的信息从数据库中提取出来，将操作后的信息写入数据库进行存储或将信息从数据库中删除。

2. 教师的业务流程

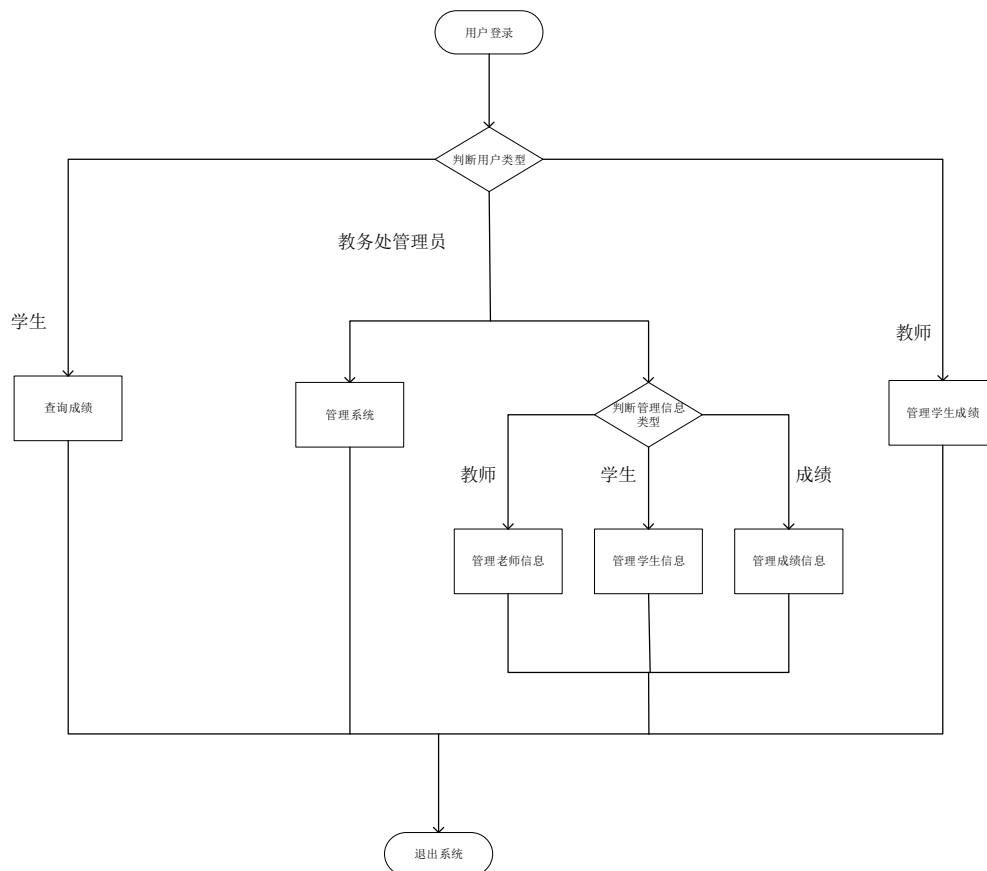
与管理者的认证流程相同，教师也需要通过身份验证，验证通过后，才能进入学生成绩信息管理、教师个人信息管理和课程信息管理等功能界面。教师进行学生信息管理和课程信息管理的查、增、改、删的操作时，将查询到的信息从数据库中提取出来，将修改后的信息写入数据库。进行学生成绩信息管理的查、增、改、删的操作时，将需要登记的学生成绩写入系统中进行存储，将查询到的学生成绩从数据库中提取出来，将改后的信息写入数据库表中进行存储。

3. 学生的业务流程

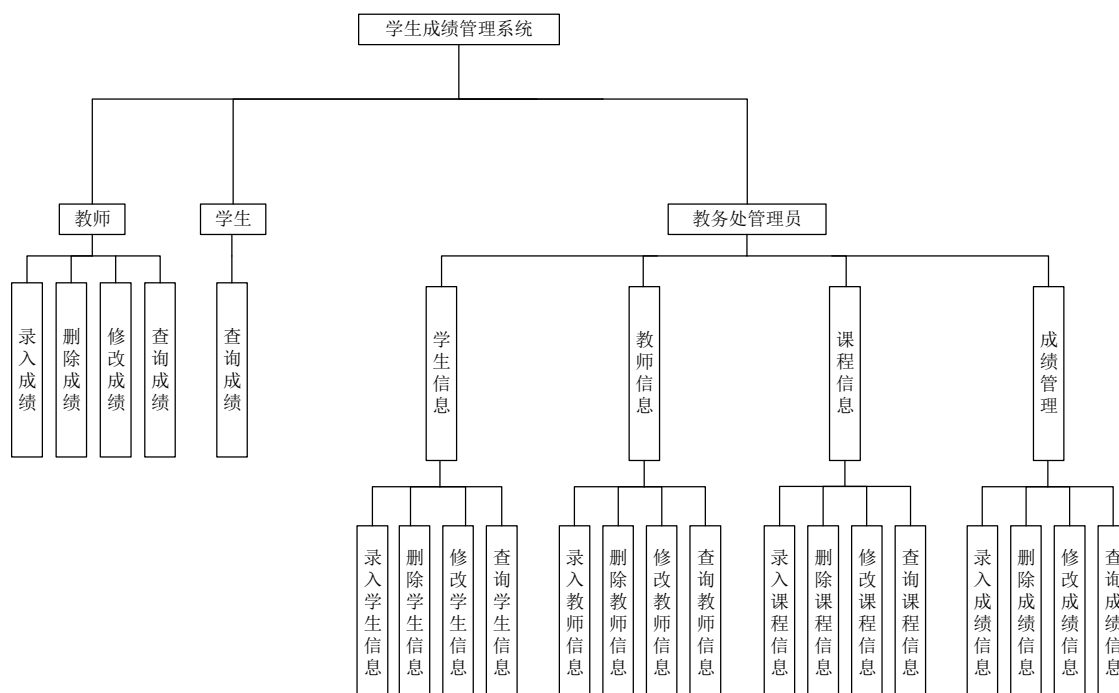
与管理者的认证流程相同，学生也需要通过身份验证，验证通过后，才能进入学生个人信息，学生成绩查询等功能界面。进行学生成绩查询的查询操作时，将需要查询的成绩信息从数据库中提取出来。

本系统的业务流程图和功能结构图如下图所示：

2.7.1 业务流程图



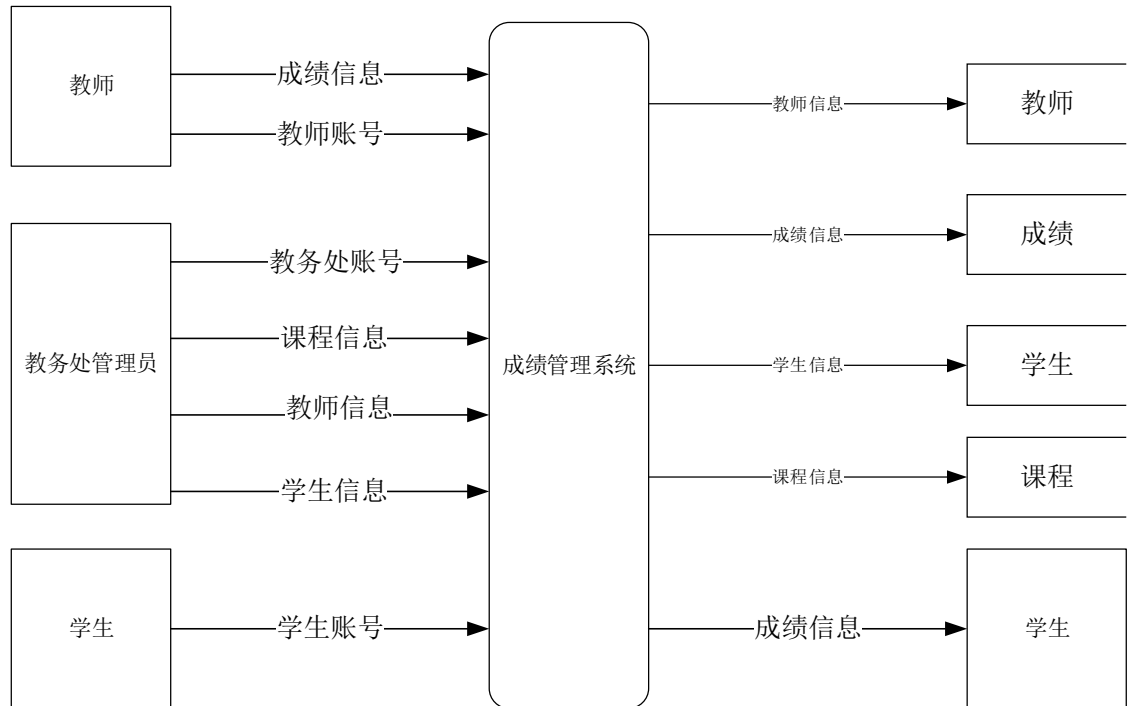
2.7.2 功能结构图



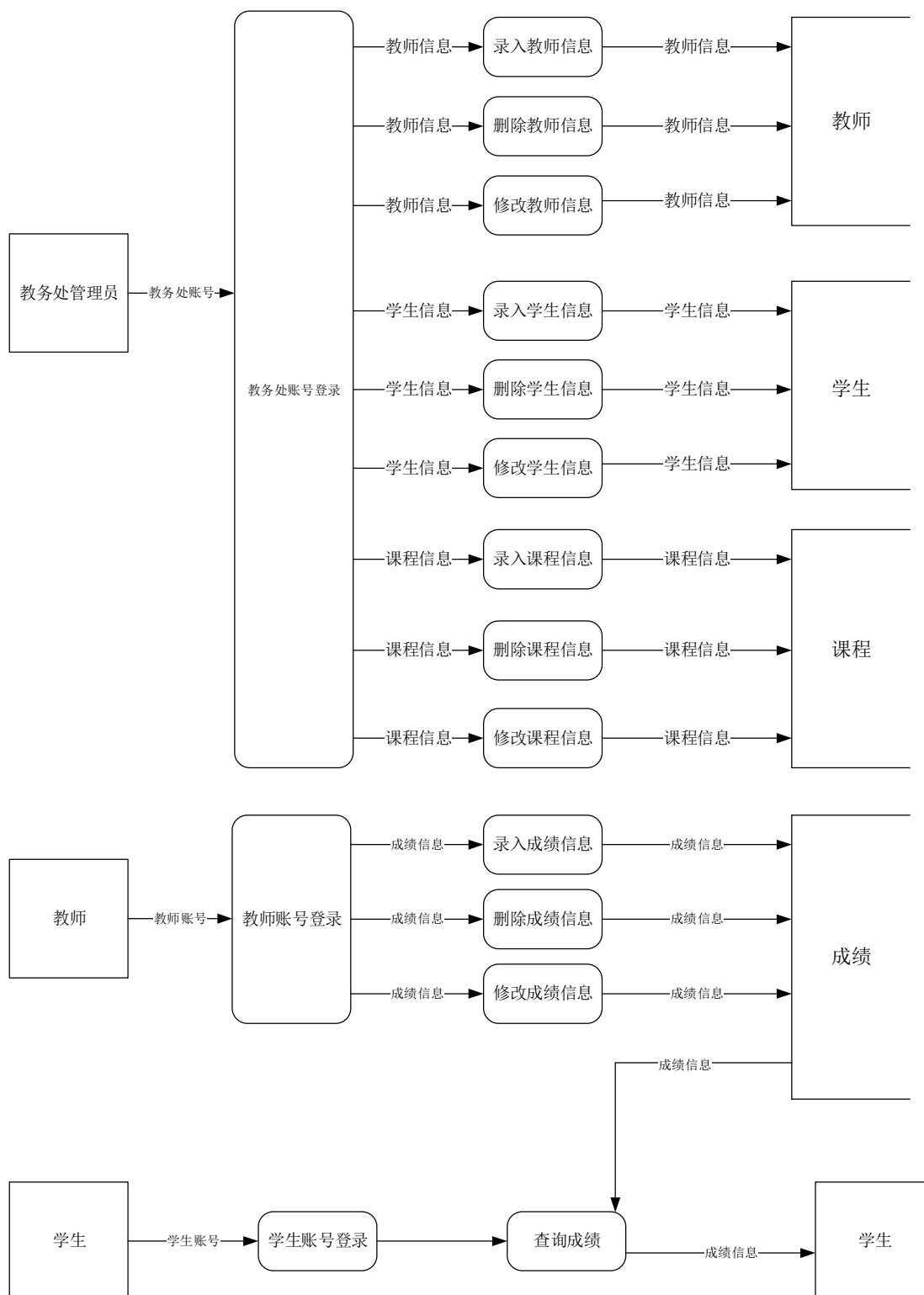
2.8 数据流程

2.8.1 数据流程图

2.8.1.1 顶层：



2.8.1.2 (2) 第一层:



2.9 系统的数据字典：

数据从程序内部分离出来,形成独立的外部数据文件。静态数据文件与动态数据文件相互独立,依靠特定程序分别进行参照或存取。静态数据由外部数据文件存放方式转为数据库存放方式,将以前程序中静态数据对动态数据加工的某些控制关系抽象为静态表与动态表之间的关系。

数据字典：

2.9.1 数据项

数据项名	数据项含义	别名	数据类型	取值范围	取值含义
Id	教务处管理员 ID	教务处管理员 ID	char(15)	0~9999999999999999	
Password	教务处管理员密码	教务处管理员密码	char(30)	0~999999...	
Sno	学生学号	学生学号	char(8)	0~99999999	
Sname	学生姓名	学生姓名	char(10)	0~9999999999	
Sex	学生性别	学生性别	char(2)	0~2	
Sdate	学生出生日期	学生出生日期	date	1970/1/1~2999/12/31	
Sdept	学生系部	学生系部	char(15)	0~9999999999999999	
Tno	教师编号	教师编号	char(6)	0~999999	

Tname	教师姓名	教师姓名	character(10)	0~9999999999	
Sex	教师性别	教师性别	character(2)	0~2	
Tel	教师电话	教师电话	character(11)	0~999999999999	
Cno	课程号	课程号	character(4)	0~9999	
Cname	课程名	课程名	character(15)	0~99999999999999	
Credit	课程学分	课程学分	int	0~5	
Term	学期	学期	character(8)	第一~八学期	
Grade	课程成绩	课程成绩	int	0~100	

2.9.2 数据结构

数据结构名	含义说明	组成
教务处管理员	是学生成绩管理子系统的主体数据结构, 定义了一个课程, 教师, 学生, 成绩的有关信息	教务处 ID, 教务处密码
课程	是学生成绩管理子系统的主体数据结构, 定义了一个课程的有关信息	课程号, 课程名, 课程学分, 学期
教师	是学生成绩管理子系统的主体数据结构, 定义了一个教师的有关信息	教师编号, 教师姓名, 教师性别, 教师电话

学 生	是学生成绩管理子系统的主体数据结构, 定义了一个学生的有关信息	学号, 姓名, 性 别, 学生出生日期, 系 部
成 绩	是学生成绩管理子系统的主体数据结构, 定义了一个成绩的有关信息	学生学号, 课程 号, 课程学分, 课程成 绩
课 程表	是课程子系统的主体数据	课程号, 课程名, 课程学分, 教师编号, 学期
教 师信息	是教师子系统的主体数据	教师编号,教师姓名, 教师性别,教师电话
学 生信息	是学生子系统的主体数据	学号, 姓名, 性 别, 出生日期, 系部
成 绩信息	是成绩子系统的主体数据	学生学号, 课程 号, 课程成绩

2.9.3 数据流

数 据流名	说 明	数据 流来源	数据 流去向	组成	
课 程表		教务 处提供	课程 表存储	课程号, 课程名, 课程学分, 教师编号, 学期	
教 师信息		教务 处提供	教师 信息存储	教师编号,教师姓名,教师性别, 教师电话	
学 生信息		教务 处提供	学生 信息存储	学生学号, 学生姓名, 学生性 别, 学生出生日期, 系部	
成 绩信息		教师 提供	成绩 信息存储	学生学号, 课程号, 课程成绩	

2.9.4 数据存储

数据 存储名	说 明	流入 的数据流	流出的数据流	组成
-----------	--------	------------	--------	----

课程		课程表		课程号, 课程名, 课程学分
教师		教师信息		教师编号,教师姓名,教师性别,教师年龄
学生		学生信息		学生学号, 学生姓名, 学生性别, 学生出生日期, 系部
成绩		成绩信息	成绩信息	学生学号, 课程号, 课程学分, 课程成绩
教务处管理员			课程表, 教师信息, 学生信息, 成绩信息	教务处 ID, 教务处密码

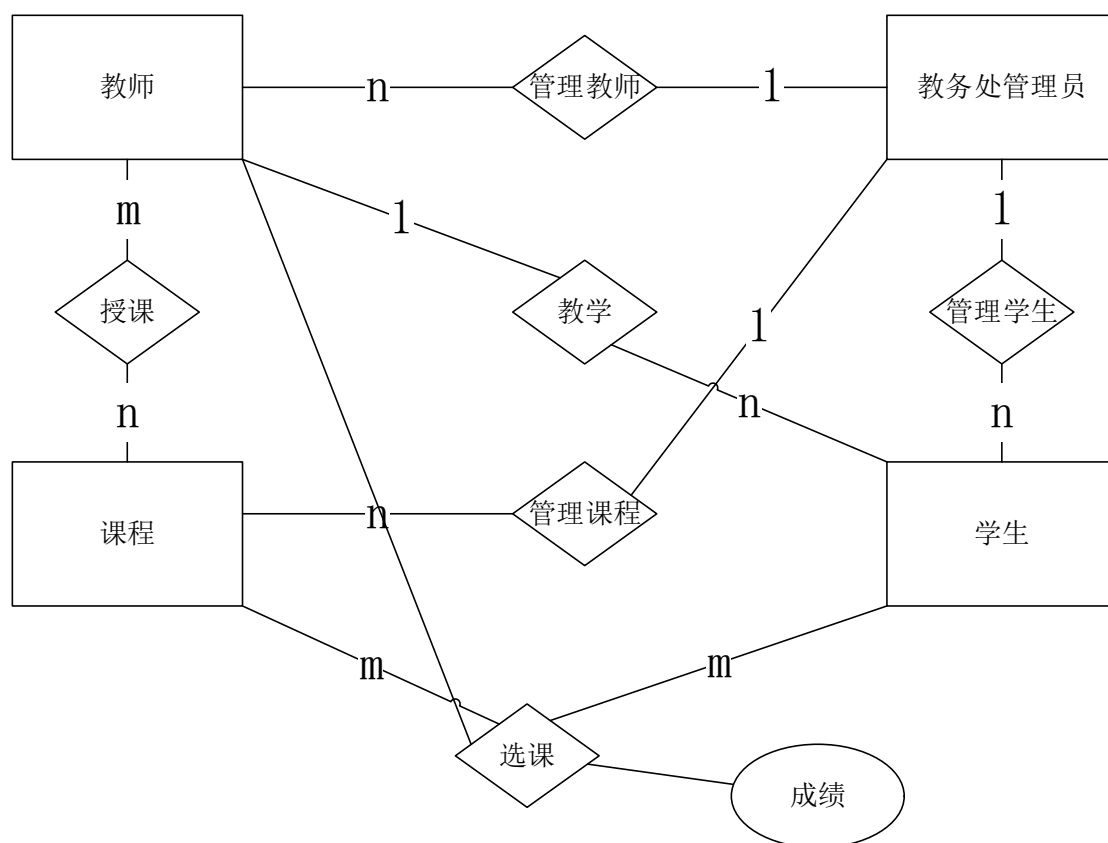
2.9.5 处理过程

处理过程名	说明	输入数据流	输出数据流	处理
课程导入		课程表		
教师信息导入		教师信息		
学生信息导入		学生信息		
成绩信息导入		成绩信息	成绩信息	
教务处管理员导入信息		课程表, 教师信息, 学生信息, 成绩信息		

3. 概念结构设计

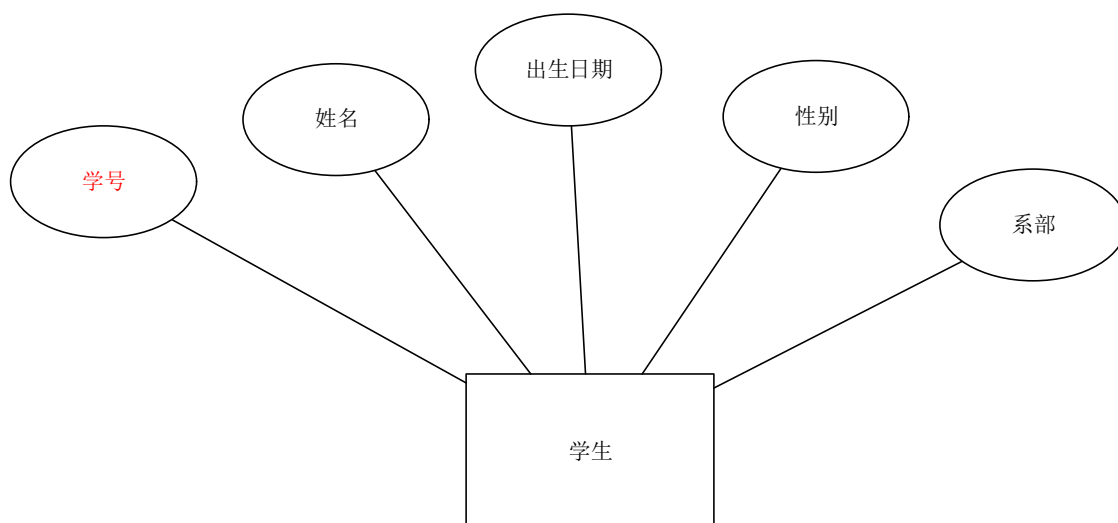
本系统的总 E-R 图如下图所示：

3.1 总 E-R 图

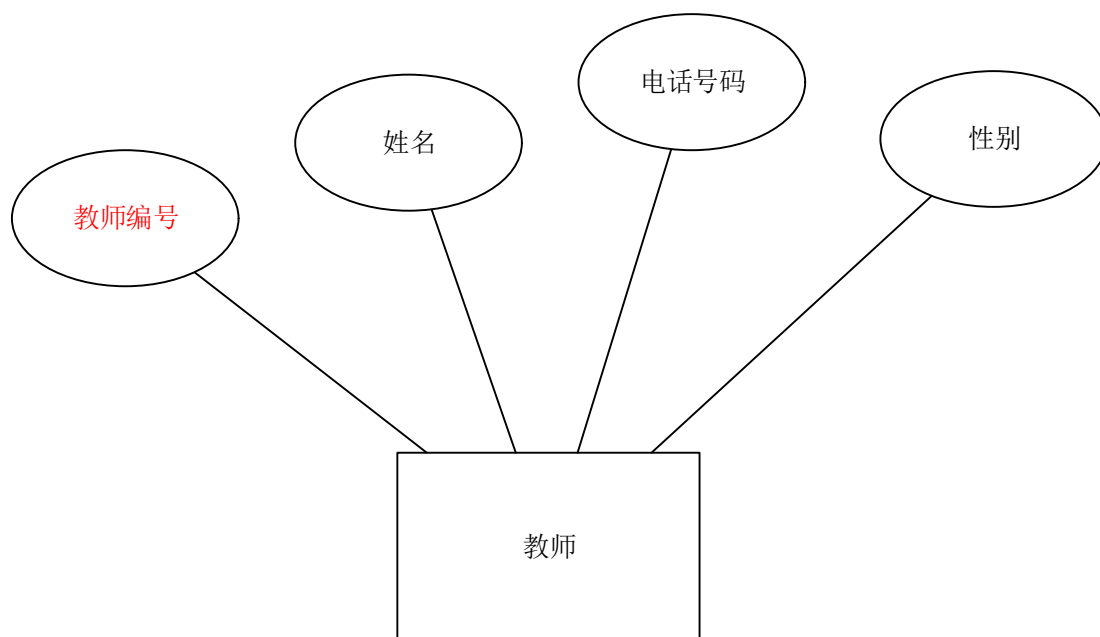


分 E-R 图如下图所示：（红色为主码）

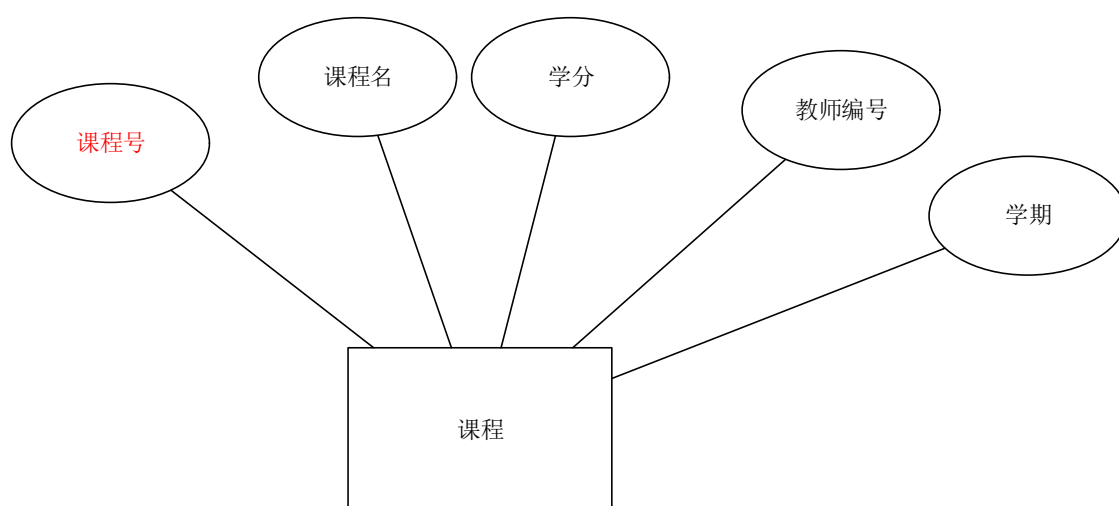
3.2 学生 E-R 图：



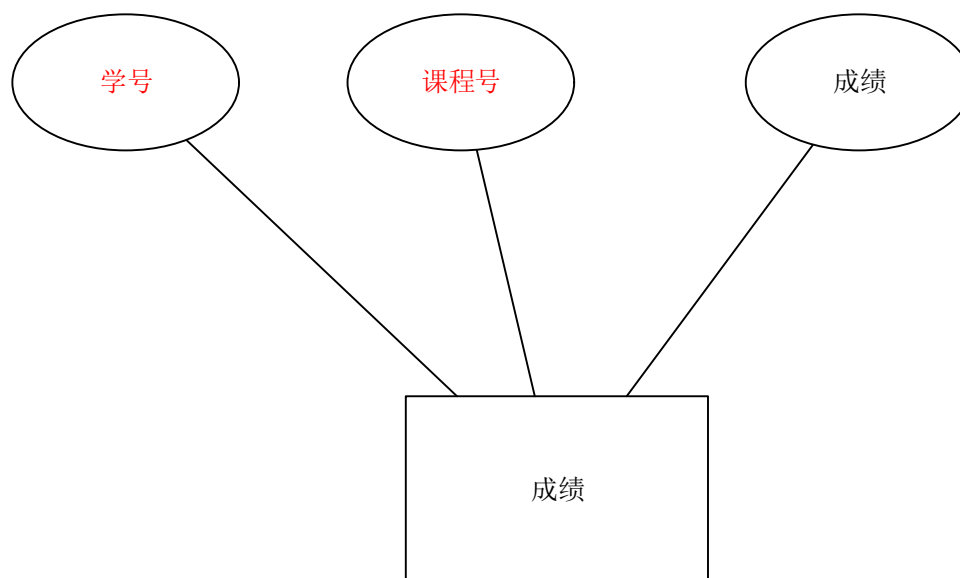
3.3 教师 E-R 图:



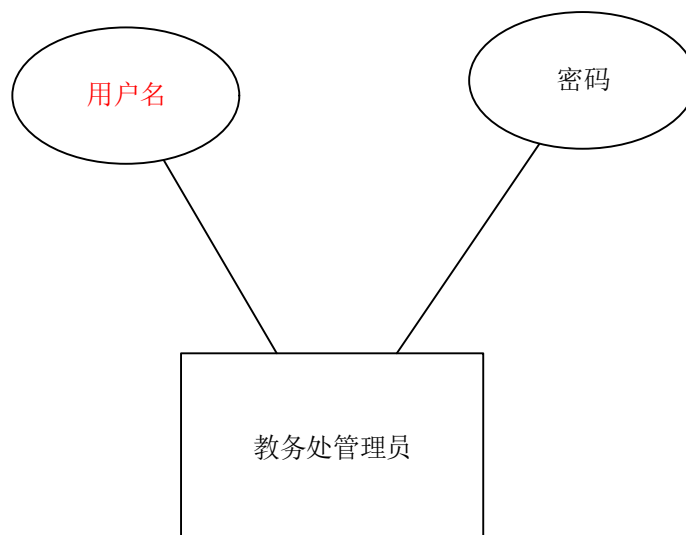
3.4 课程 E-R 图:



3.5 成绩 E-R 图:



3.6 教务处管理员 E-R 图:



4. 逻辑结构设计

把系统的 E-R 图转换成数据库模式如下（红色为主码）：

4.1 学生表：

列名	数据类型	说明
Sno	char(8)	学生学号
Sname	char(10)	学生姓名
Sex	char(2)	学生性别
Sdate	date	学生出生 日期
Sdept	char(15)	学生系部

4.2 教师表：

列名	数据类型	说明
Tno	char(6)	教师编号
Tname	char(10)	教师姓名
Sex	char(2)	教师性别
Tel	char(11)	教师电话

4.3 课程表：

列名	数据类型	说明
Cno	char(4)	课程号
Cname	char(15)	课程名
Credit	int	课程学分
Term	char(8)	学期
Grade	int	课程成绩

4.4 成绩表：

列名	数据类型	说明
Sno	Char (8)	学生学号
Cno	Char (4)	课程号

Grade	Int	学生成绩
-------	-----	------

4.5 教务处管理员表:

列名	数据类型	说明
Id	char(15)	教务处管 理员 ID
Password	char(30)	教务处管 理员密码

4.6 本系统的关系模式

学生（学号、姓名、性别、出生日期、系部）

教师（教师编号、姓名、性别、联系电话）

课程（课程号、课程名、教师编号、学分、学期）

选课（学号、课程号、成绩）

教务处（用户名、密码）

5. 物理结构设计

数据库在物理设备上的存储结构与存取方法称为数据库的物理结构，它依赖于选定的数据库管理系统。为一个给定的逻辑数据模型选取一个最适合的应用要求的物理结构的过程，就是数据库的物理设计。

数据库的物理设计通常分为两步：

（1）确定数据库的物理结构，在关系数据库中主要指存取方法和存储结构。

（2）对物理结构进行评价，评价的重点是时间和空间的效率。

6. 数据库实施

本系统前端开发工具我们选择 java ， 后台数据库采用 MySQL 。本系统是通过 java 代码进行连接的。

MYSQL 代码如下：

创建数据库

Create database sams;

Use sams;

创建表

Create table admin(

Id char(15) unique not null primary key,

Password char(30)

);

Create table s(

Sno char(8) unique not null primary key,

Sname char(10),

Sex char(2),

Sdate date,

Sdept char(15)

);

Create table t(

Tno char(6) unique not null primary key,

Tname char(10),

Sex char(2),

Tel char(11)

);

```
Create table c(  
Cno char(4) unique not null primary key,  
Cname char(15),  
Credit int,  
Term char(8)  
);
```

```
Create table sc(  
Sno char(8),  
Cno char(4),  
Tno char(6),  
Grade int,  
Foreign key (sno) references s(sno),  
Foreign key (cno) references c(cno),  
Foreign key (tno) references t(tno)  
);
```

插入数据

```
Insert into admin values('101','123456');
```

```
Insert into s values
```

```
(' 20200001' , ' ' , ' 男' , ' 2000/01/01 ' , ' 计算机' ),  
( ' 20200002' , ' ' , ' 男' , ' 2001/01/01 ' , ' 计算机' ),  
( ' 20200003' , ' ' , ' 男' , ' 2001/02/01 ' , ' 机械' ),  
( ' 20200004' , ' ' , ' 男' , ' 2000/10/01 ' , ' 机械' ),  
( ' 20200005' , ' ' , ' 女' , ' 2002/01/01 ' , ' 经管' ),
```

```
( ' 20200006 ' , '      ' 女 ' , ' 2001/01/01 ' , ' 艺术 ' ),  
( ' 20200007 ' , '      ' , ' 男 ' , ' 1999/01/01 ' , ' 体育 ' ),  
( ' 20200008 ' , '      ' , ' 男 ' , ' 2000/06/01 ' , ' 体育 ' );
```

Insert into t values

```
( ' 100001 ' , ' ██████████ ' , ' 女 ' , ' 17809520001 ' ),  
( ' 100002 ' , ' ██████████ ' 女 ' , ' 17809520002 ' ),  
( ' 100003 ' , ' ██████████ ' , ' 男 ' , ' 17809520003 ' ),  
( ' 100004 ' , ' ██████████ ' 女 ' , ' 17809520004 ' );
```

Insert into c values

```
( ' 2001 ' , ' 数据库 ' , 3 , ' 第二学期 ' ),  
( ' 2002 ' , ' JAVA ' , 3 , ' 第二学期 ' ),  
( ' 2003 ' , ' 离散 ' , 3 , ' 第二学期 ' ),  
( ' 2004 ' , ' C语言 ' , 2 , ' 第一学期 ' );
```

Insert into sc values

```
('20200001','2001','100001',90),  
( '20200002','2001','100001',90),  
( '20200003','2001','100001',90),  
( '20200004','2002','100003',90),  
( '20200005','2003','100004',90),  
( '20200006','2004','100002',90),  
( '20200007','2001','100001',90),  
( '20200008','2002','100003',90);
```

创建视图和索引

```
Create index i_sname on s(sname);
Create index i_cname on c(cname);
Create index i_tname on t(tname);
Create view v_samesdept_grade as (
select * from sc where sc.sno in(
select sno from s where sdept = ' 计算机' )
);
```

创建触发器

```
Delimiter @@
Create trigger t_delete_s before delete on s for each row
Begin
Declare oldsno char(8);
Set oldsno = old.sno;
Delete from sc where sc.sno = oldsno;
End @@
Delimiter ;
```

代码截图：

```
1899 x 1043 MySQL Command Line Client - Unicode
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 22
Server version: 3.0.22 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> Create database sams;
Query OK, 1 row affected (0.01 sec)

mysql> Use sams;
Database changed

mysql>
mysql> Create table admin(
  -> Sno char(15) unique not null primary key,
  -> Password char(30)
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql>
mysql> Create table s(
  -> Sno char(8) unique not null primary key,
  -> Sname char(10),
  -> Sex char(2),
  -> Sdate date,
  -> Sdept char(15)
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql>
mysql> Create table t(
  -> Tno char(6) unique not null primary key,
  -> Tname char(10),
  -> Sex char(2),
  -> Tel char(11)
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql>
mysql> Create table c(
  -> Cno char(4) unique not null primary key,
  -> Cname char(15),
  -> Credit int,
  -> Term char(8)
  -> );
Query OK, 0 rows affected (0.02 sec)

mysql>
mysql> Create table sc(
  -> Sno char(8),
  -> Cno char(4),
  -> Tno char(6),
  -> Grade int,
  -> Foreign key (sno) references s(sno),
  -> Foreign key (cno) references c(cno),
  -> Foreign key (tno) references t(tno)
  -> );
Query OK, 0 rows affected (0.04 sec)

mysql>
mysql> Insert into admin values('101','123456');
Query OK, 1 row affected (0.00 sec)

mysql> Insert into s values ('20200001','男','2000/01/01','计算机'),
  -> ('20200002','男','2001/01/01','计算机'),
  -> ('20200003','男','2001/02/01','机械'),
  -> ('20200004','男','2000/10/01','机械'),
  -> ('20200005','女','2002/01/01','经济'),
  -> ('20200006','女','2001/01/01','艺术'),
  -> ('20200007','男','1999/01/01','体育'),
  -> ('20200008','男','2000/06/01','体育');
Query OK, 8 rows affected (0.01 sec)
Records: 8 Duplicates: 0 Warnings: 0

mysql>
mysql> Insert into t values('100001','女','17809520001'),
  -> ('100002','女','17809520002'),
  -> ('100003','男','17809520003'),
  -> ('100004','女','17809520004');
Query OK, 4 rows affected (0.00 sec)
Records: 4 Duplicates: 0 Warnings: 0

mysql>
mysql> Insert into c values('2001','数据库',3,'第二学期'),
  -> ('2002','JAVA',3,'第二学期'),
  -> ('2003','离散',3,'第二学期'),
  -> ('2004','C语言',2,'第二学期');
Query OK, 4 rows affected (0.01 sec)
Records: 4 Duplicates: 0 Warnings: 0

mysql>
mysql> Insert into sc values('20200001','2001','100001',90),
  -> ('20200002','2001','100001',90),
  -> ('20200003','2001','100001',90),
  -> ('20200004','2002','100003',90),
  -> ('20200005','2003','100004',90),
  -> ('20200006','2004','100002',90),
  -> ('20200007','2001','100001',90),
  -> ('20200008','2002','100003',90);
Query OK, 8 rows affected (0.00 sec)
Records: 8 Duplicates: 0 Warnings: 0

mysql>
mysql> Create index i_sname on s(sname);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> Create index i_cname on c(cname);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> Create index i_tname on t(tname);
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> Create view v_samedept_grade as (
  -> select * from sc where sc.sno in(
  -> select sno from s where sdept='计算机');
Query OK, 0 rows affected (0.01 sec)

mysql>
mysql> Delimiter @@
mysql> Create trigger t_delete_s before delete on s for each row
  -> Begin
  -> Declare oldano char(8);
  -> Set oldano=old.sno;
  -> Delete from sc where sc.sno=oldano;
  -> End @@
Query OK, 0 rows affected (0.02 sec)

mysql> Delimiter ;
mysql>
mysql>
```

7. 数据库运行和调试

软件维护是软件生命周期的最后一个阶段，它处于系统投入生产性运行以后的日期中，为了纠正错误或满足新的需要而进行修改软件的过程，即是软件维护也是为软件的升级提供详细的资料。系统维护的目的是保证管理信息系统正常而可靠的运行，并能使系统不断得到改善和提高以充分发挥作用。

1. 学生管理

包括入住新的学生，以及对学生信息进行修改、查询，删除。

2. 课程管理

包括开设新的课程，以及对课程信息进行修改、查询，删除。

3. 教师管理

包括聘请新的教师，以及对教师信息进行修改、查询，删除。

4. 退出系统

退出该学生成绩管理信息系统。

8. 总结

在这么短的时间内，我们跟随者老师的步伐，完成了数据库综合实训。在整个过程中，我体会到了团结合作的重要性，也不能忽视分工协作的重要性。一开始，我们都是自己做自己的，直到后来，老师说要一起做，我们才开始商量着解决问题的方案，有不会的也一起上去询问老师，老师也很耐心地解答。

在实训中，我学会了 Visio 2013 的使用以及如花画项目流程图和功能图，还学会了数据库设计的 ER 图和数据流程图，制作数据字典等，也对 MYSQL 更加熟练。也学会了如何跟团队进行合作，如何交流，划分任务等。希望在以后能有更多的时间来完善优化这个系统，达到最优的状态。

在这里，我要感谢老师的教诲，同学的帮助，组员的支持！让我在这次实训中受益匪浅！