

.NET程序设计课程设计

实验报告

**2021/2022(1)**



实验题目 高校成绩管理系统

学生姓名 吕锐

学生学号 201906061619

学生班级 软件工程1902

任课教师 王松

提交日期 2022/6/4

**计算机科学与技术学院**

# 需求分析

计算机已经深入到日常工作和生活的方方面面，成为我们学习和工作的得力助手，比如文字处理、信息管理、辅助设计、图形图像处理、教育培训以及游戏娱乐等。随着越来越多的应用软件出现，人们对它的要求也越来越高；虽然现在世界上的各种软件层出不穷，但它们依然不能满足用户的各种特殊需要，所以人们仍是不得不开发适合特殊需求的软件。高校成绩管理系统记录了一个大学生成绩的系统,它的出现使得查询、更新、插入简单化，高效化，成本也随之大大减少。使用计算机对成绩信息的管理,具有手工管理所 无法比拟的优点：信息存储及时，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,也是高校成绩正规化管理的重要途径。

对于某校管理学生成绩的工作人员，根据实际工作需要，提出了以下数据和业务处理需求：

1、学校设置了各专业，在专业下开设班级，每个班级包含若干学生，学生信息至少需要包含学号、姓名、性别、年龄、生源所在地、已修学分总数等数据项；另外，需要有地区信息，用于统计某一地区的学生数；

2、课程信息表至少需包含课程编号、课程名称、任课教师、开课学期、学时、考试或考查、学分等数据项，课程根据班级开设。

3、教师信息至少需要包含教师编号、姓名、性别、年龄、职称、联系电话等数据项；

4、学生成绩至少需要学号，学期，课程名称，成绩，任课老师等数据项；

5、需要实现以下业务处理和查询功能：

学生成绩按每学年成绩统计

学生成绩名次排定

每门课程平均成绩统计

学生所学课程及学分统计

对每个学生输入成绩的时候，自动生成学生的已修学分总数

学生成绩查询

教师任课查询

班级课程开设查询

# 程序设计

## 技术选型

在前端，我们使用Vue CLI工具一体化配置、构建、发布项目，使用Vue作为MVVM框架，利用Naïve UI的组件库来加速和美化前端页面的设计。除此之外，我们还使用组件化的形式，给予代码高可重用性并统一页面风格。

在后端，我们使用ASP.NET作为我们的后端服务框架，使用LINQ方便数据库的查询并自动化数据库对应类的生成，节省工作时间。

## 系统总体设计

数据库

ORM

ASP

主体逻辑

Web前端

图示, 示意图

描述已自动生成

## 系统功能设计

1. 学生信息管理

根据班级添加、删除、修改学生信息

1. 教师信息管理

添加、删除、修改教师信息

1. 班级信息管理

根据专业添加、删除、修改班级信息

1. 课程信息管理

添加、删除、修改课程信息

1. 学生成绩录入

根据课程，为课程下所有学生添加、修改成绩

1. 班级排课管理

可以为班级添加、删除课程

1. 教师任课管理

可以为课程分配、移除任课教师

1. 学生成绩查询

可以根据专业班级查询特定学生的所有课程成绩

1. 学生成绩名次排定

可以筛选班级来选择学生，并对选择的学生按照成绩排名

1. 课程平均成绩统计

显示所有课程的平均成绩

1. 学生生源地人数统计

显示各个生源地的学生人数

## 类设计

用于表示班级信息，有班级编号，姓名，专业编号。

public partial class Class

{

public Class()

{

Students = new HashSet<Student>();

}

public string Id { get; set; }

public string Name { get; set; }

public string MajorId { get; set; }

[JsonIgnore]

public virtual Major MajorNavigation { get; set; }

[JsonIgnore]

public virtual ICollection<Student> Students { get; set; }

}

用于表示班级排课信息，有班级编号和课程编号。

public partial class ClassCourse

{

public string ClassId { get; set; }

public string CourseId { get; set; }

[JsonIgnore]

public virtual Class ClassNavigation { get; set; }

[JsonIgnore]

public virtual Course CourseNavigation { get; set; }

}

用于表示课程信息，有课程编号，课程名称，开设学期，学时，学分，考试方式。

public partial class Course

{

public string Id { get; set; }

public string Name { get; set; }

public string Semester { get; set; }

public int Hours { get; set; }

public int Credit { get; set; }

public string ExamMethod { get; set; }

}

用于表示课程的平均成绩，有课程编号，课程名称，任课教师，开设学期，学时，学分，考试方式和平均成绩。

public class CourseAverageScore

{

public string Id { get; set; }

public string Name { get; set; }

public List<string> Teacher { get; set; }

public string Semester { get; set; }

public int Hours { get; set; }

public int Credit { get; set; }

public string ExamMethod { get; set; }

public double? AverageScore { get; set; }

}

用于表示完整的课程信息，有课程编号，课程名称，任课教师编号，任课教师姓名，开设学期，学时，学分，考试方式。

public class CourseFull

{

public string Id { get; set; }

public string Name { get; set; }

public List<string> TeacherId { get; set; }

public List<string> TeacherName { get; set; }

public string Semester { get; set; }

public int Hours { get; set; }

public int Credit { get; set; }

public string ExamMethod { get; set; }

}

用于表示专业信息，有专业编号和专业名称。

public partial class Major

{

public Major()

{

Classes = new HashSet<Class>();

}

public string Id { get; set; }

public string Name { get; set; }

[JsonIgnore]

public virtual ICollection<Class> Classes { get; set; }

}

用于表示某课程的学生成绩信息，有学生编号，学生姓名和成绩。

public class StudentCourseScore

{

public string Id { get; set; }

public string Name { get; set; }

public int? Score { get; set; }

}

用于表示生源地人数西信息，有生源地和生源地人数。

public class SourcePlaceCount

{

public string SourcePlace { get; set; }

public int Count { get; set; }

}

用于表示学生信息，有用户编号，生源地，班级编号和已修学分。

public partial class Student

{

public string UserId { get; set; }

public string SourcePlace { get; set; }

public string ClassId { get; set; }

public int? Credit { get; set; }

[JsonIgnore]

public virtual Class ClassNavigation { get; set; }

[JsonIgnore]

public virtual User UserNavigation { get; set; }

}

用于表示学生某一课程的成绩信息，有学生编号，课程编号和成绩信息。

public partial class StudentCourse

{

public string StudentId { get; set; }

public string CourseId { get; set; }

public int? Score { get; set; }

[JsonIgnore]

public virtual Course CourseNavigation { get; set; }

[JsonIgnore]

public virtual Student StudentNavigation { get; set; }

}

用于表示某学生的课程成绩信息，有课程编号，课程名称，任课教师名称，开设学期，学时，学分，考试方式，成绩。

public class StudentCourseFull

{

public string Id { get; set; }

public string Name { get; set; }

public List<string> Teacher { get; set; }

public string Semester { get; set; }

public int Hours { get; set; }

public int Credit { get; set; }

public string ExamMethod { get; set; }

public int? Score { get; set; }

}

用于表示学生完整信息，有学生编号，学生姓名，年龄，性别，班级名称，班级编号，专业名称，专业编号，已修学分和生源地。

public partial class StudentFull

{

public string StudentId { get; set; }

public string Name { get; set; }

public int Age { get; set; }

public string Gender { get; set; }

public string Class { get; set; }

public string ClassId { get; set; }

public string Major { get; set; }

public string MajorId { get; set; }

public int? Credit { get; set; }

public string SourcePlace { get; set; }

}

用于表示学生的平均成绩完整信息，与学生完整信息的区别在于没有已修学分而是改成了课程平均成绩。

public class StudentGradeFull

{

public string StudentId { get; set; }

public string Name { get; set; }

public int Age { get; set; }

public string Gender { get; set; }

public string Class { get; set; }

public string ClassId { get; set; }

public string Major { get; set; }

public string MajorId { get; set; }

public double Grade { get; set; }

public string SourcePlace { get; set; }

}

表示教师信息，有用户编号，职称和电话。

public partial class Teacher

{

public string UserId { get; set; }

public string Title { get; set; }

public string Telephone { get; set; }

[JsonIgnore]

public virtual User UserNavigation { get; set; }

}

用于表示教师的任课信息，有教师编号和课程编号。

public partial class TeacherCourse

{

public string TeacherId { get; set; }

public string CourseId { get; set; }

[JsonIgnore]

public virtual Course CourseNavigation { get; set; }

[JsonIgnore]

public virtual Teacher TeacherNavigation { get; set; }

}

用于表示教师的完整信息，有教师编号，教师姓名，年龄，性别，职称和电话号码。

public class TeacherFull

{

public string TeacherId { get; set; }

public string Name { get; set; }

public int Age { get; set; }

public string Gender { get; set; }

public string Title { get; set; }

public string Telephone { get; set; }

}

用于表示用户信息，有用户编号，用户姓名，年龄，性别，密码和角色。

public partial class User

{

public string Id { get; set; }

public string Name { get; set; }

public int Age { get; set; }

public string Gender { get; set; }

public string Password { get; set; }

public string Role { get; set; }

[JsonIgnore]

public virtual Student Student { get; set; }

[JsonIgnore]

public virtual Teacher Teacher { get; set; }

}

# 程序开发

## 程序运行截图

1. 学生信息管理

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

1. 教师信息管理

图形用户界面, 应用程序

描述已自动生成

1. 班级信息管理



1. 课程信息管理

电脑屏幕截图

描述已自动生成

1. 学生成绩录入

图形用户界面, 应用程序, Teams

描述已自动生成

1. 班级排课管理

图形用户界面, 文本, 应用程序

描述已自动生成

1. 教师任课管理

图形用户界面, 应用程序, Teams

描述已自动生成

1. 学生成绩查询

图形用户界面, 文本, 应用程序

描述已自动生成

1. 学生成绩名次排定

电脑屏幕截图

描述已自动生成

1. 课程平均成绩统计

图形用户界面, 应用程序

描述已自动生成

1. 学生生源地人数统计

图形用户界面, 应用程序

描述已自动生成

# 收获与总结

在本次实验中，我们充分利用学习过的.NET知识和我们学过的前端、数据库等知识融合，让.NET赋能高校成绩管理系统，打造完善的一流管理系统抓手。

通过这次.NET的大型实验，我深刻的明白到：课本知识与实践能力相结合的重要性。会读程序的人并不f一定会编程序。要想把一门专业课程学好，必须增强自己的动手实践能力，而不是一天到晚只知道看书，那种行为只不过是“纸上谈兵”。看再多的书都不如自己亲手试一试。俗话说的好：会打仗的士兵才是好士兵。再者，课本上的知识不一定是完全准确的，只有自己动手进行试验过才知道。实践是检验真理的唯一标准，这话不假。