

**摘要** 培训机构建立线上培训管理系统，在如今这个几乎人人使用互联网的情况下已经是培训机构的常态了。线上管理系统能极大的改善因为人工原因导致的报名出错，时间地点受限，传统处理方式跟不上节奏等问题。

本系统采用 ASP.NET MVC5+Castle 架构进行系统设计，可扩充性与灵活性较好，使用 Layui 前端框架与 ASP.NET MVC5 进行表示层构建，Castle 进行数据访问层搭建，数据库则使用了 SQL Server2008。本文从需求分析，系统设计，系统开发，测试等软件开发过程进行描述。系统的主要功能有学员报名选班，课表查询，统计分析；教师课前考勤，反馈学习情况，登记学生成绩，学生查看出勤信息，查看课表与教师反馈等功能。改善了因人工处理导致的效率不足与容易出现错误以及时间地点受限等问题。

**关键词：** 学生管理 培训机构 MVC Castle

### 1 绪论

现在正处在一个崭新的时代，网络与各种系统在方方面面支撑着我们的生活，工作，学习。计算机行业的兴起，令我们无时无刻都能享受到各式各样的系统给我们生活学习带来的便捷。本少儿培训管理系统，能提高培训机构的学员选课效率，教师上课考勤效率，以及数据统计的准确性，高效性。本文将结合少儿培训机构的实际需求开发“少儿培训管理系统”。

#### 1.1 业务背景

在这个家长们望子成龙，不希望孩子输在起跑线的时代，出现了越来越多的培训机构，由于报名的人数越来越多，导致报名效率低下，数据的统计与管理也愈发繁忙和冗杂，使得工作量大幅增加。利用互联网，可以轻松解决报名的时间，地理问题，客户可以在各种时间，任意地点进行培训班的报名，极大的改善了客户的使用体验。且学生的考勤，信息管理等工作无需额外安排人手，管理员与教师后台操作即可，可减缓报名工作的繁忙，能提高整体的工作效率与质量。

本文的“少儿培训机构管理系统”以重庆市某艺术培训机构的真实项目为原型，在保留核心业务基础上，将项目的业务功能进行了删减，在保留核心业务的基础上进行提炼，形成为一个结构完整，流程清晰的软件项目。

#### 1.2 开发目的及意义

本系统的开发目的是帮助培训机构，制作一款能实际帮助到培训机构与学生和教师的培训管理系统，使其能够令学生的报名操作更加高效，教师展开教学管理工作更加便捷。从而使得培训机构的工作效率大大提高，同时节省不必要的人工成本，减少了出错的可能性。

---

少儿培训管理系统是一个能够给培训机构提供信息以及各种数据管理的实用工具，可以为其带来极大的便利，节省许多时间和人工成本。

### 1.3 论文结构

本文主要结构为：第一章：绪论描述了少儿培训管理系统的业务背景以及项目开发的目的是与意义，同时分析了本文的结构。第二章：开发相关技术及工具

描述了本系统开发使用的相关技术以及使用到的开发工具。主要介绍了 ASP.NET MVC, Entity Framework 等技术。

第三章：系统需求分析内容包括系统的分析，以及业务主流程、系统业务分析、用例图、用例描述以及非功能性需求的分析。

第四章：系统分析与设计描述了少儿培训管理系统具体的功能设计，UI 设计，数据库设计，以及架构设计。

第五章：项目实施

描述了项目具体的实现内容以及步骤。包括开发环境（Visual Studio 2019 等），三层架构的实现，主要功能的实现等。

第六章：测试介绍了测试目的，测试方法，测试用例等内容。对项目功能进行了黑盒测试。

第七章：总结总结了项目的开发经验以及自我收获。

## 2 开发相关技术及工具

本系统采用 ASP.NET MVC5、Castle、Layui 等技术，以及 Visual Studio 2019 等开发工具来进行实现。

### 2.1 主要开发技术

#### 2.1.1 ASP.NET MVC

ASP.NET 是较成熟的技术，MVC 则是 ASP.NET 的一种编程模式，它解耦了用户界面即视图层 (View: 显示数据)、模型层 (Model: 应用程序核心)、控制器层 (Controller: 处理输入)，这是一种把系统架构进行分离的框架模式，便于异步并行开发，不同的开发者可只关注自己的模块。

选择 ASP.NET 技术是因为该技术比较成熟，考虑到本项目，用户数量相对较多，有较大的访问量，良好的稳定性也是选择该技术的原因之一。ASP.NET 在程序部署方面也很便捷。丰富而强大的类库也能让程序的开发速度提升不少，功能也更强大且扩展性强，稳定性高。

选择 MVC 技术则是因为他的耦合性低，重用性高，进行修改时无需重新编译模型，可直接修改视图层，提高了程序开发速度

#### 2.1.2 Castle

---

Castle 是针对 .NET 平台下的一个开源项目，从数据访问框架 ORM 到依赖注入容器，再到 WEB 层的 MVC 框架基本包括了整个开发过程中的所有东西。

本系统当中，选择该技术是因为该技术能使项目开发更加迅速，只需要较少的代码，就可以完成以前需要很多行代码才能完成的数据库交互。加之本项目体量较小，使用更加方便的 Castle 能更快的完成工作。

### 2.1.3 Layui

Layui 是一款采用自身模块规范编写的前端 UI 框架，无需使用各种复杂的前端工具，即便是非前端程序员也可以轻松上手。同时 Layui 可兼容市面上绝大部分浏览器，对快速开发界面有很大的帮助。

在本系统当中，选择 layui 技术是因为该技术能使项目开发更加迅速，LayUi 作为一款国产开源的前端 UI，简单易上手，并且 UI 简洁美观。

## 2.2 开发工具

本系统的开发，采用的是微软的 Microsoft Visual Studio 2019 来进行源代码的编写，数据库则是采用的 SQL Server 2008 R2。

### (1) Microsoft Visual Studio 2019

VS2019 是微软推出的 VS 编程开发软件，它拥有强大的源码编辑器功能，以及庞大扩展插件库，可以帮助运行以及编辑 C/C++、C#、PHP 等各种编程语言。

### (2) SQL Server 2008 R2

2008 年微软推出了这个版本的数据库。SQL Server 的性能方面是相当强大的，稳定性方面也很可靠。该版本的数据库完全可以满足要求，所以在能够提高稳定性的情况下，选择了比较成熟且经典的版本。

## 2.3 本章小结

本章主要介绍了本系统实现的开发技术和开发工具。其中主要介绍了 ASP.NET MVC 实现技术这个重点，以及 EF 与 layui 两个技术。其中的 MVC 是 ASP.NET 三种开发模式 (Web Pages、MVC、Web Forms) 的一种，它最大的特点就是分为了 Model (模型) 包含了业务数据与业务逻辑，是程序的主体部分，View (视图) 呈现给用户的部分，是用户与程序交互的地方，Controller (控制器) 处理用户输入的数据。最后简要介绍了开发环境和数据库环境以及设计工具等信息，使用的开发工具为 Microsoft Visual Studio 2019，使用的数据库是 SQL Server 2008 R2。

## 第 3 章 系统需求分析

系统需求分析是问题定义的最后一个阶段，基本任务是回答“系统需要做什么”这个问题。需求分析的结果是系统开发的基础，直接关系到系统最后的质量。

### 3.1 业务需求

家长们都渴望自己的孩子赢在起跑线上，培训机构一时间成为了热门行业，但大浪涌入的客户导致信息处理成为了机构的一大工作难题。为了帮助培训机构提高工作效率，服务水平，作为帮助培训机构管理信息，处理数据的在线培训管理系统，其任务包含教务人员可进行基础信息管理与维护，如学生信息，教师信息，班级信息，课程信息，以及各种数据统计，教师可进行学生上课签到，学生课堂学习情况反馈，学生能在线进行选课，查看学习情况与教师反馈信息。

### 3.2 业务流程

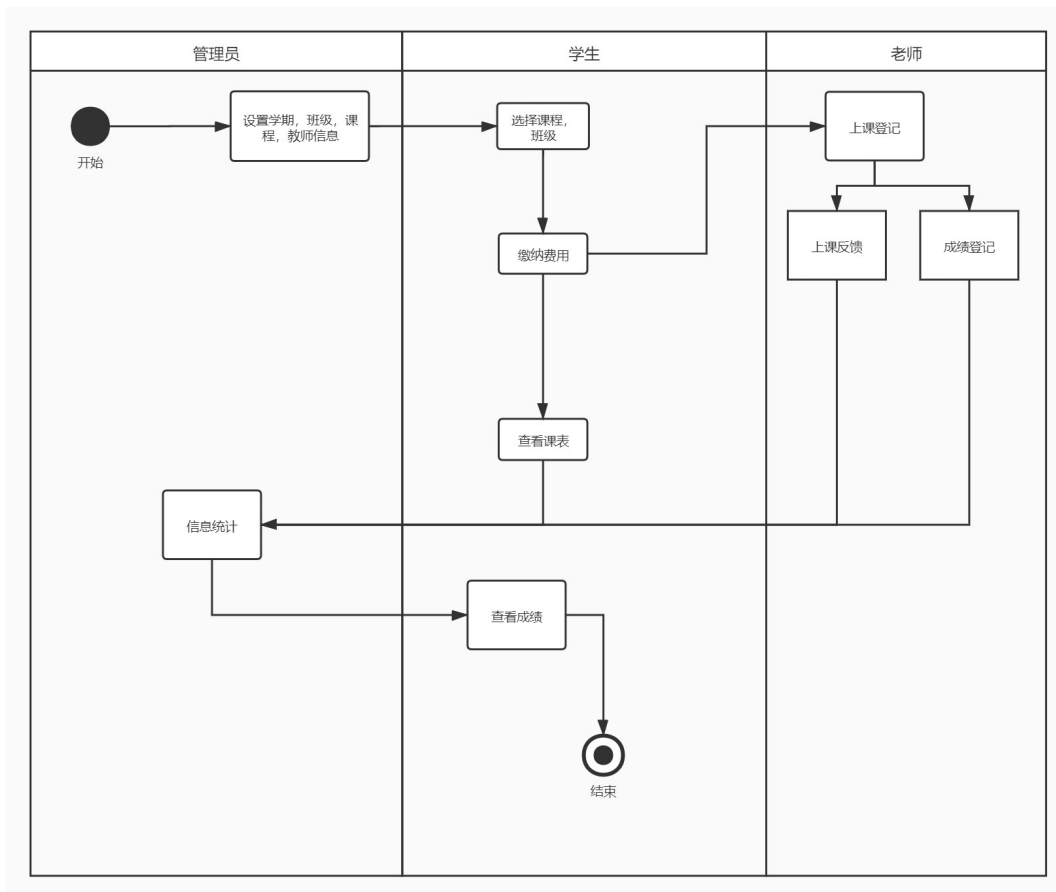


图 1 业务流程图

### 3.3 业务角色分析

本系统主要应用于少儿培训管理系统的报名管理工作。其中涉及三种角色，分别是管理员，教师与学生。

#### 3.3.1 管理员业务

---

(1) 信息管理管理学期：每学期开始，由管理员录入学期名称、设置学期时间管理课程：维护以前存在的课程，增加新课程管理班级、教师、学员：删除无效信息或成员。

(2) 班级管理

需直接设置好班级的上课课程、时间，任课教师，系统不做后续排课。

(3) 发布通知可发布系统内用户都可见的公告，能支持图片发布，指导过期时间

(4) 信息统计能实现对学生收费统计、学生出勤统计数据整合，统计结果用表格和图表显示。

统计学期内各个班级的学分收费收入及总收入。每个班学生的出勤情况，按正常、迟到、早退方式进行汇总。

### 3.3.2 教师业务

(1) 查看课表：查看自己对应的课表。

(2) 上课登记：教师在每次上课前对班上的每名学员做出勤登记，对迟到、缺课的学生做状态标记。每次课后，教师要给每名学员做课后信息反馈。

(3) 成绩登记：如部分课程需要考试，在考试结束后，教师需登记学生成绩。

### 3.3.3 学生业务

(1) 报名选班：报名时学员选择学习的班级。

(2) 查看课表：查看自己对应的课表。

(3) 查看信息：查看每次课程的出勤信息，课后评价与成绩。

## 3.4 系统用例分析

在本系统中，根据业务分析得出系统中，共涉及三个角色，分别为管理员，教师，学生，主要涉及到学生报名，教师考勤签到，管理员数据统计等功能，下面以这三个方面进行用例描述，整体见图2所示

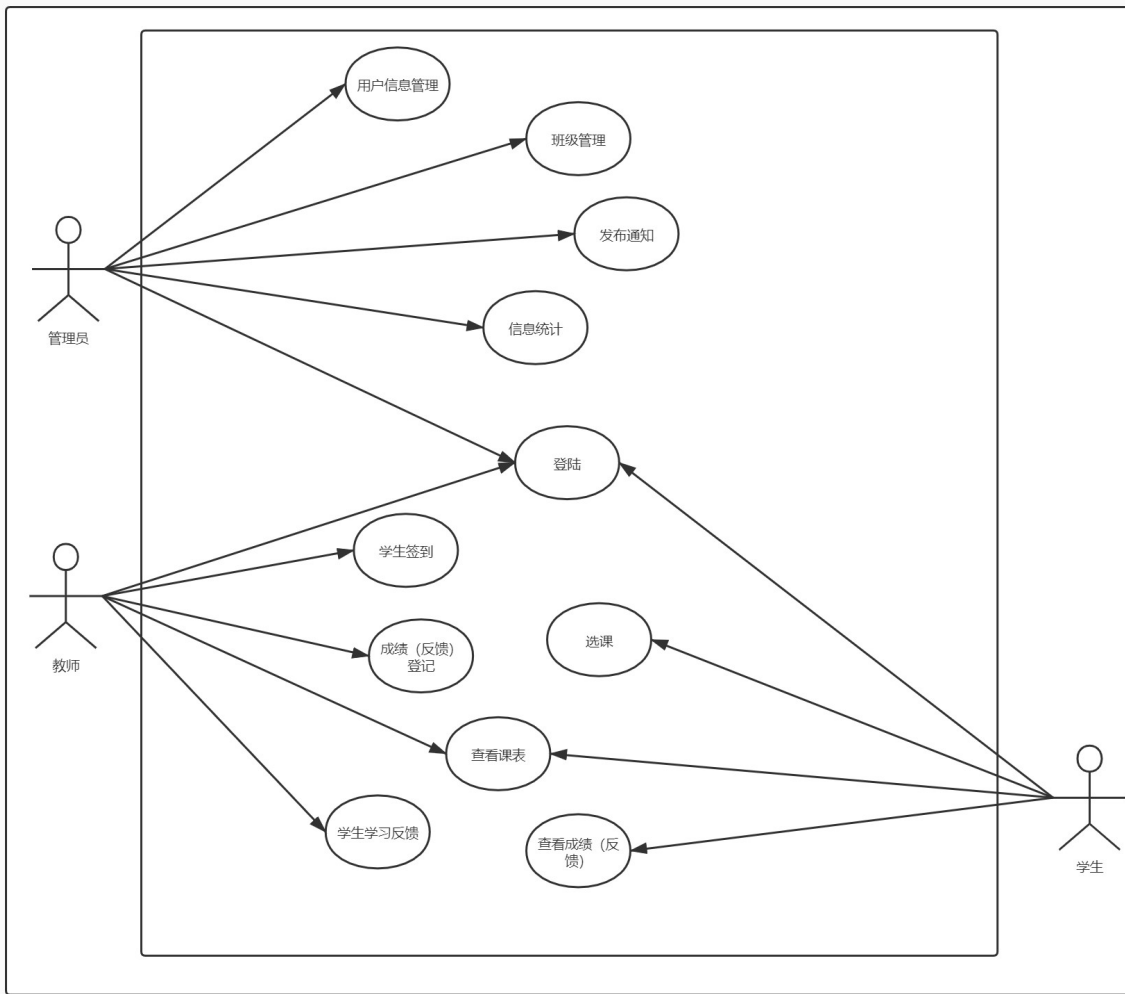


图 2 系统整体用例图 3.4.1 学生选课用例

学生选课是学生选择自己的课程，用例描述如表 1 所示。

表 1 学生选课用例

参与者	学生
基本流	1. 学生用户进入选课主题界面 2. 学生用户在主题页面选择科目和班级。 3. 学生用户提交学生的报名信息，系统对报名信息进行验证。 4. 系统返回报名结果并保存。
替代	1a. 验证失败：系统提示验证失败消息。 2a. 保存成功：跳转回到学生端主页面。 2b. 保存失败：提示保存信息错误，页面停留在选课页面。
前置条件	学生用户已登录系统。
后置条件	选择后的课程状态为已选课，系统进入学生端主页面。
特殊需求	无

### 3.4.2 教师考勤签到用例

教师考勤签到是教师开始上课后对上课学生进行考勤签到，用例描述如表 2 所示：表 2 教师考勤签到

参与者	学生教师
基本流	1. 教师用户进入上课主题界面。 2. 教师用户在上课主题界面勾选学生上课签到状态。 3. 教师用户提交学生上课签到状态信息，系统对学生上课签到状态信息进行验证。 4. 系统返回上课签到结果并保存。
替代	1a. 验证失败：系统提示验证失败信息。 2a. 保存成功：跳转页面至上课主题页面。 2b. 保存失败：提示保存信息错误，页面停留在上课签到页面。
前置条件	教师用户已登录系统，在课程教学页面点击开始上课
后置条件	完成签到后学生上课更改为状态已上课，系统进入结束课程页面
特殊需求	无

3.4.1 管理员对用户数据统计用例管理员数据统计时管理员对学生上课信息进行统计，用例描述如表 3 所示：表 3 管理员数据统计

参与者	管理员
基本流	<ol style="list-style-type: none"> <li>1. 管理员用户进入数据管理页面。</li> <li>2. 管理员用户在数据管理页面点击用户管理查看数据统计。</li> <li>3. 管理员用户点击修改按钮，系统弹出数据修改页面。</li> <li>4. 管理员用户对需要修改的数据进行修改。</li> <li>5. 管理员用户提交修改后的数据，系统对修改后的数据惊喜验证。</li> <li>6. 系统返回数据修改的结果并保存。</li> </ol>
替代	<ol style="list-style-type: none"> <li>1a. 验证失败：系统提示验证失败信息。</li> <li>2a. 保存成功：跳转至管理员用户主题页面。</li> <li>2b. 保存失败：提示保存失败信息，页面停留在数据管理页面。</li> </ol>
前置条件	管理员用户已登录系统
后置条件	完成数据修改保存后系统保存修改的数据，回到管理员用户主题页面。
特殊需求	无

### 3.5 非功能性需求

#### (1) 软件性能需求

少儿培训管理系统要求系统页面要求简洁明了。在用户使用时，访问速度不应过慢，需保证支持50人以上同时并发使用，同时预留足够的空间以共后期其他功能加入，保证系统具备可拓展性。

正确性：系统在运行过程中无数据计算错误，无过程错误。

可靠性：系统支持长时间运行，不会因系统功能计算复杂而发生错误。

易用性：系统的功能使用操作简单，没有复杂的逻辑设定。

可扩展性：提供良好的系统接口，支持后续功能的开发扩展。

清晰性：系统的功能结果及名称清晰，避免用户误解。

#### (2) 软件兼容性需求

系统软件要求为B/S架构，支持台式电脑终端和笔记本都能够自适应运行，通过火狐、谷歌、IE10+、360浏览器等主流浏览器上均能兼容运行。

### 3.6 本章小结

本章首先根据分析得出了系统的主要针对对象及系统的参与者、次要参与者等信息，其中包括管理员、教师、学生等必要参与者。通过对用例的分析，得出了系统的各个功能需求，如管理员信息管理，学生选课等。在本章的最后，还提出了非功能性需求，包括软件性能需求、软件兼容性需求。

## 4 系统分析与设计

系统分析与设计就是对研究的对象进行功能，界面，数据库等方面的内容进行分析，并做出准确的定位与设计，为之后的实现打下坚实的基础



在本次的系统分析和设计工作中，主要从系统功能设计、数据库设计、UI 设计及架构设计四个方面进行了详细阐述。

#### 4.1 系统功能设计

系统功能设计就是对整个系统进行描述，阐述出有哪些功能包括系统设计约束等，基本建设原则包括使用的技术不老旧，系统的功能实用不花哨，系统的结构合理，产品风格主流，成本控制在合理范围，维护量尽可能低等，以此来规划系统的整体架构。

根据需求分析阶段的业务流程及用例分析，得出系统的用例清单，如表 4 所示。

表 4 系统的用例清单

角色名称	用例名称	用例场景描述
管理员端	用户管理	管理员管理用户信息
	学期信息管理	管理员添加学期时间，包含班级等信息
	班级管理	管理员添加上课课程，时间，任课教师
	课程管理	管理员设置课程的编号，名称，教师，学期，上课时间，地点，人数上限，激活情况，以及课程详情
	发布公告	管理员发布系统公告
	修改公告	管理员修改系统公告
教师端	开始课程	教师开始上课
	学生签到	教师记录学生出勤情况
	结束课程	教师结束课程并给学生学习情况打分
	课程反馈	教师反馈学生上课情况
	查看课表	查看与自己相关的课表
	课程报名	学生进行课程报名
	查看课表	查看与自己相关的课表
学生端	查看反馈	学生查看课程反馈信息
	查看成绩	学生查看课程成绩

管理员端有基础信息管理与公告管理功能，基础信息管理功能包括用户管理，学期信息管理，班级管理，课程管理；公告管理功能包括发布与修改公告。

教师端有课程教学与课表查看功能，课程教学功能包括开始课程，学生签到，结束课程，以及课程反馈。

学生端有报名选课和教学反馈信息查看功能，反馈信息查看功能包括查看课表，查看教师反馈信息，查看成绩功能。

以下以系统中的学生选课，教师上课签到，管理员数据管理功能为例进行功能设计描述

##### 4.1.1 学生选课功能设计

管理员发布课程，学生再在系统中选择自己要学习的课程；发布的课程包含课程名称，课程时间，课程教师，课程费用信息。完成本功能涉及到的类及属性抽象如下：

(1) 管理员类：用户名、姓名、密码。

用户名：10 字符以内。

姓名：10 字符以内。

密码：20 字符以内，由数字或字母组成，使用 MD5 加密。

(2) 课程类：课程学院，课程名称，课程班级，课程教师，课程费用。

课程学院：20 字符以内。课程名称：20 字符以内。课程班级：10 字符以内。课程教师：10 字符以内。

课程费用：20 字符以内。

(3) 学生类：姓名，性别，学号，生日，学院，添加日期，密码。

姓名：20 字符以内。性别：10 字符以内。学号：20 字符以内。

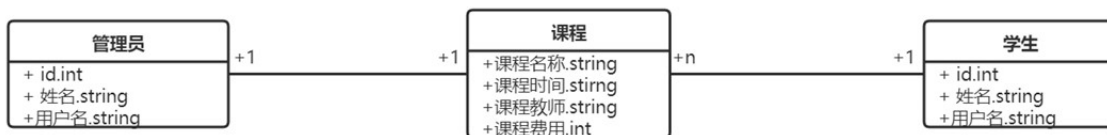
生日：具体日期。

学院：20 字符以内。

添加日期：具体日期。

密码：20 字符以内，由数字或字母组成，使用 MD5 加密。

以上三个对象的关系为：管理员可管理多门课程，管理员类与课程类之间的关系是多对多的关系；学生可选择多门课程，对象间关系为多对多。由上述分析可得场景领域模型如下图。



#### 4.1.2 教师签到功能设计

教师开始上课，对学生进行上课签到，学生上课状态记录在签到信息中，签到信息包含班级，课程名称，学生姓名，签到信息。完成本功能涉及到的对象及属性抽象为：

(1) 教师：用户名，姓名，密码。

用户名：20 字符以内。

姓名：20 字符以内。

密码：20 字符以内，由数字或字母组成，使用 MD5 加密。

(2) 签到信息：班级，课程名称，学生姓名，签到信息。

班级：20 字符以内。

签到信息：5 字符以内。

课程名称：20 字符以内。

学生姓名：20 字符以内。

(3) 学生：姓名，性别，学号，生日，学院，添加日期，密码。

姓名：20 字符以内。性别：10 字符以内。学号：20 字符以内。

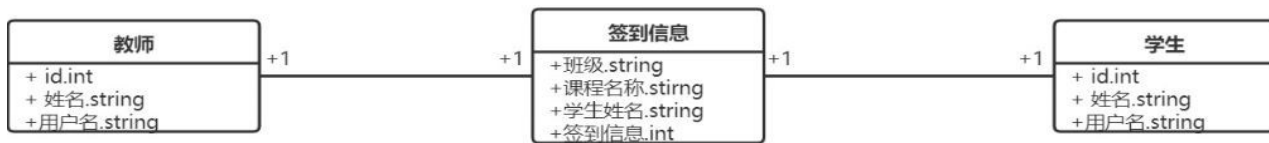
生日：具体日期。

学院：20 字符以内。

添加日期：具体日期。

密码：20 字符以内，由数字或字母组成，使用 MD5 加密。

以上三个对象的关系为：教师每次进行一门课程签到，对象间关系为一对一；学生进行一门课程被签到，对象间关系为一对一。由上述分析可得场景领域模型如下图。



4. 1. 3 管理员统计用户数据功能设计管理员对统计的管理的用户数据进行查看，修改，删除等操作。

完成本功能涉及到的对象及属性抽象为：

(1) 管理员：用户名，姓名，密码。

用户名：10 字符以内。

姓名：10 字符以内。

密码：20 字符以内，由数字或字母组成，使用 MD5 加密。

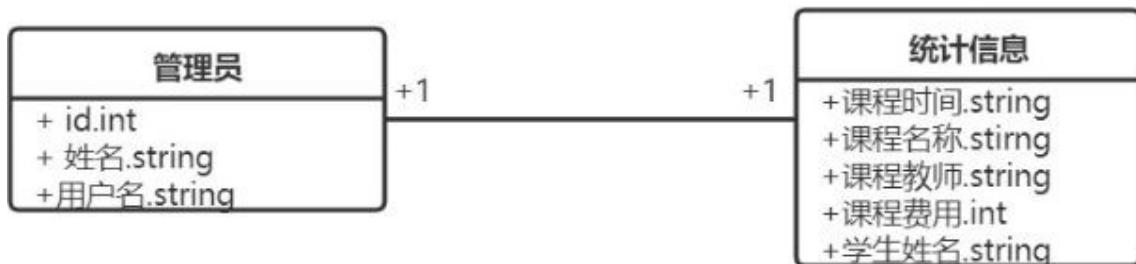
(2) 统计信息：课程名称，课程时间，课程教师，课程费用，学生姓名。

课程名称：20 字符以内。

课程时间：具体日期。

课程教师：10 字符以内。课程费用：10 字符以内。学生姓名：10 字符以内。

以上两个对象的关系为：管理员管理多门课程的统计信息，对象之间关系为多对多。由上述分析可得场景领域模型如下图。



输入到数据库中的数据要有审核和约束机制。

## 4.2 数据库设计

### 4.2.1 数据库设计原则

- (1) 一致性原则数据来源需进行及系统的设计与分析，协调好数据来源，保证数据的一致性。
- (2) 完整性原则

数据库的完整性是指数据的正确性和相容性。要防止合法用户使用数据库时向数据库加入不合语义的数据。对（3）范式要求第一范式，确保每列的原子性，要求每列（或者每个属性值）都是不可再分的最小数据单元；第二范式，要求表中的每列都和主键相关，即要求实体的唯一性；第三范式，确保每列都和主键列直接相关，而不是间接相关。

(4) 可伸缩性与可拓展性原则数据库设计要考虑今后的移植与拓展，需要具备较好的扩展性，伸缩性，同时也可以适度冗余。1

### 4.2.2 数据库模型设计

根据域模型设计的结果，依据 ORM 映射思想，从模型中抽象出数据表对象，将领域模型的对象作为数据表，对象中的属性映射为字段。

本系统经过分析，共设计数据表 9 张，分别是学期信息表（Semester\_Info）学院信息表（College\_Info）管理员信息表（Admin\_Info）班级信息表（Class\_Info）课程信息表（Course\_Info）教师信息表（Teacher\_Info）签到信息表（Sign\_Info）学生信息表（Student\_Info）选课信息表（Choice\_Info）。

### 4.2.3 数据库表

1. 学期信息表（Semester\_Info），用于储存学期的基本信息。。

表名	Semester_Info			备注
列名	数据类型（精度范围）	空/非空	约束条件	
Id	Int	非空	主键	学期编号
Name	Varchar（10）	非空		学期名称
BeginTime	Date	非空		学期开始时间

2. 学院信息表（college\_Info），用于储存学院的基本信息。

表名	college_Info			备注
列名	数据类型（精度范围）	空/非空	约束条件	
Id	Int	非空	主键	学院编号
Name	Varchar（10）	非空		学院名称

3. 管理员信息表（Admin\_Info），用于储存管理员的基本信息，主键字段为 Id。

表名	Admin_Info			备注
列名	数据类型（精度范围）	空/非空	约束条件	
Id	Int	非空	主键	编号
Name	Varchar（10）	非空		姓名
Pwd	Varchar（20）	非空		密码

4. 班级信息表（Class\_Info），用于储存班级的基本信息，主键字段为 Id。

表名	Class_Info			备注
列名	数据类型（精度范围）	空/非空	约束条件	
Id	Int	非空	主键	编号
Name	Varchar（20）	非空		名称
XueyuanId	int	非空		学院Id
XueyuanName	Varchar（20）	非空		学院名称
Teacher	Varchar（20）	非空		教师
Num	int	非空		人数

5. 课程信息表（Course\_Info），用于储存课程的基本信息，主键字段为 Id。

表名	Course_Info			备注
列名	数据类型（精度范围）	空/非空	约束条件	
Id	Int	非空	主键	编号
Name	Varchar（10）	非空		名称
XueyuanId	Varchar（20）	非空		学院Id
XueyuanName	Varchar（20）	非空		学院名称
ClassName	Varchar（20）	非空		所属班级
Teacher	Varchar（20）	非空		任课教师

6. 教师信息表（Teacher\_Info），用于储存教师的基本信息，主键字段为 Id。

表名	Teacher_Info			备注
列名	数据类型（精度范围）	空/非空	约束条件	
Id	Int	非空	主键	编号
Name	Varchar（10）	非空		名称
Sex	Varchar（5）	非空		性别
XueyuanName	Varchar（20）	非空		学院名称
ClassName	Varchar（20）	非空		所属班级
Addtime	Date	非空		添加时间
Pwd	Varchar（20）	非空		密码

7. 签到信息表（Sign\_Info），用于储存成绩的基本信息，主键字段为 Id。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/808102076026006076>