

# 《软件工程》课程项目 概要设计说明书

项目名称： 针对学生的个性化  
评价系统

组 号： 第八组

组 长： 张睿诚

组 员： 周杨洋、杨振宇、刘冠群、  
姜鑫宇、龚雨珂、陈煜斌、  
陈若愚、曹灿

完成时间： 2020/11/23

# 软件需求规格说明书

---

## 软件需求规格说明书

### 1 引言

- 1.1 编写目的
- 1.2 背景
- 1.3 定义
- 1.4 参考资料

### 2 总体设计

- 2.1 需求规定
- 2.2 运行环境
- 2.3 基本设计概念和处理流程
- 2.4 结构
- 2.5 功能需求与程序关系
- 2.6 人工处理过程
- 2.7 尚未解决的问题

### 3 接口设计

- 3.1 用户接口
- 3.2 外部接口
- 3.3 内部接口
  - 3.3.1 算法部分
  - 3.3.2 Web应用部分

### 4 运行设计

- 4.1 运行模块组合
  - 4.1.1 成绩预测功能:
  - 4.1.2 个性化评价功能:
- 4.2 运行控制
  - 4.2.1 成绩预测功能运行控制
  - 4.2.2 个性化评价功能运行控制
- 4.3 运行时间

### 5 系统数据结构设计

- 5.1 逻辑结构设计要点
- 5.2 物理结构设计要点
- 5.3 数据结构与程序的关系

### 6 系统出错处理设计

- 6.1 出错信息
  - 6.1.1 输入错误
  - 6.1.2 Web服务器错误处理设计
- 6.2 补救措施
  - 6.2.1 后备技术
  - 6.2.2 降效技术
  - 6.2.3 恢复及再启动技术
- 6.3 系统维护设计

## 1 引言

---

### 1.1 编写目的

---

编写此报告旨在说明对程序系统的设计考虑,包括程序系统的基本处理流程、程序系统的组织结构、模块划分、功能分配、接口设计、运行设计、安全设计、数据结构设计和出错处理设计等。为程序的详细设计提供基础,同时也是对后续所有技术工作的约束、规范和指南。

本报告预期读者广泛，主要针对小组中的程序系统设计、开发人员和相关的测试人员。

## 1.2 背景

---

待开发软件系统为：针对学生的个性化评价系统。此项目为北京大学软件与微电子学院软件工程课程的课程项目，开发者为软件工程第八小组成员组成，主要用户为软件工程老师、助教和选修了软件工程课程的学生。

大规模管理的班级内，学生的知识基础、学习能力会明显分层。因此对不同的学生提出不同的建议和个性化评价，找到其最适应的学习路线，是提高效率的最佳方式。然而因材施教的教育方式对教师提出了极高的要求，不仅需大量关注学生的学习情况，还需针对每个学生制定个性化方案。个性化评价系统将代替老师完成这些大量冗杂的工作，自动分析学生的学习状态并作出个性化评价。此系统在还提供了根据学生日常表现预测学生期末成绩的功能，旨在督促和激励学生的阶段性学习。

## 1.3 定义

---

SRS:Software Requirement Specification(软件需求说明)

DFD:Data Flow Diagram(数据流图)

DD: Data Dictionary(数据字典)

WSGI:Web Server Gateway Interface(Web服务器网关接口)

wsgiref:python内置的一个WSGI服务器的参考实现模块

ORM:Object Relational Mapping(对象关系映射)，用于实现面向对象编程语言里不同类型系统的数据之间的转换。

MySQL:一个RDBMS(Relational Database Management System)关系型数据库管理系统。

MTV: Model (模型: 负责业务对象和数据库的ORM) ,Template (负责如何把页面展示给用户html) ,View (负责业务逻辑, 并在适当时候调用Model和Template) 。

## 1.4 参考资料

---

《针对学生的个性化评价系统需求分析说明书》

# 2 总体设计

---

## 2.1 需求规定

---

输入项目：学生学习情况和学生基本情况的录入；师登录信息、学生登录信息及管理员登录信息的录入。

输出项目：输入信息经由相应处理程序处理后向存储结构、下一处理程序或用户的输出；对于教师的个性化评价信息、成绩预测信息和学生信息的输出；对于学生的个性化评价信息和成绩预测信息的输出；对于管理员的版本信息、更新信息、允许更新信息和学生个人更新信息的输出。

处理项目：对于学习信息、基本信息、分类信息、成绩预测信息、个性评价信息、成绩预测报告及个性化评价信息的生成；对于个性化评价查询信息、成绩预测查询信息、学生信息查询信息及个人信息更新信息的生成；对于版本更新信息、学生信息更新信息、个人更新审核信息、版本信息、更新信息及允许更新信息的生成；对于相关存储结构中数据的导入及调用。

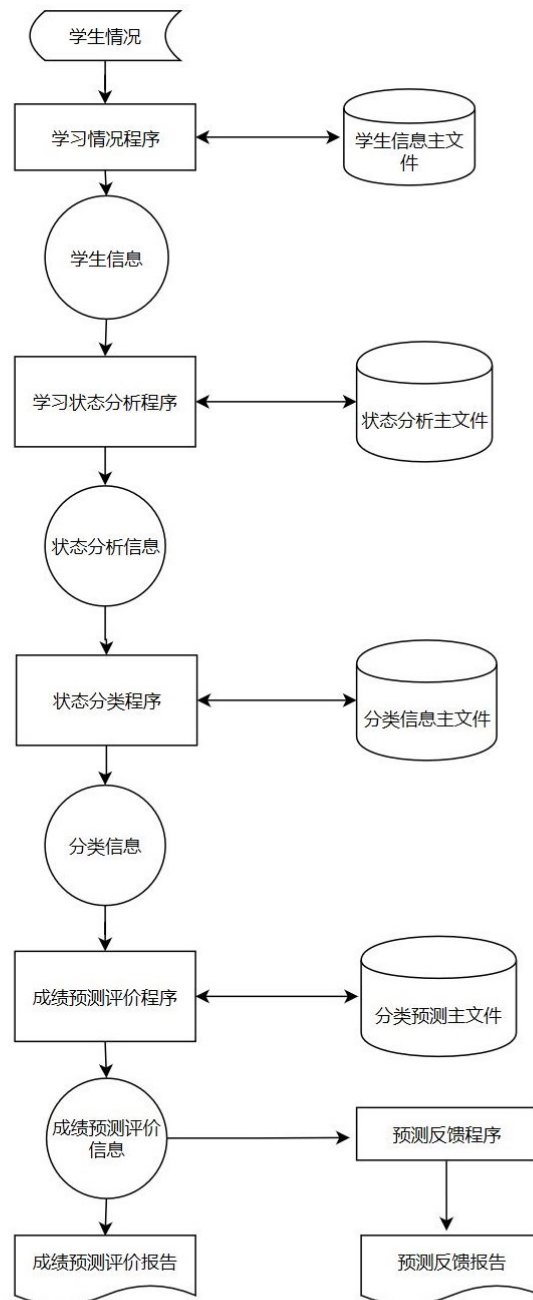
有能力支持100个并发用户；平均响应时间小于1秒；最大响应时间小于5秒。

## 2.2 运行环境

操作系统：Windows 7、OS X El Capitan、Android Oreo及以上。

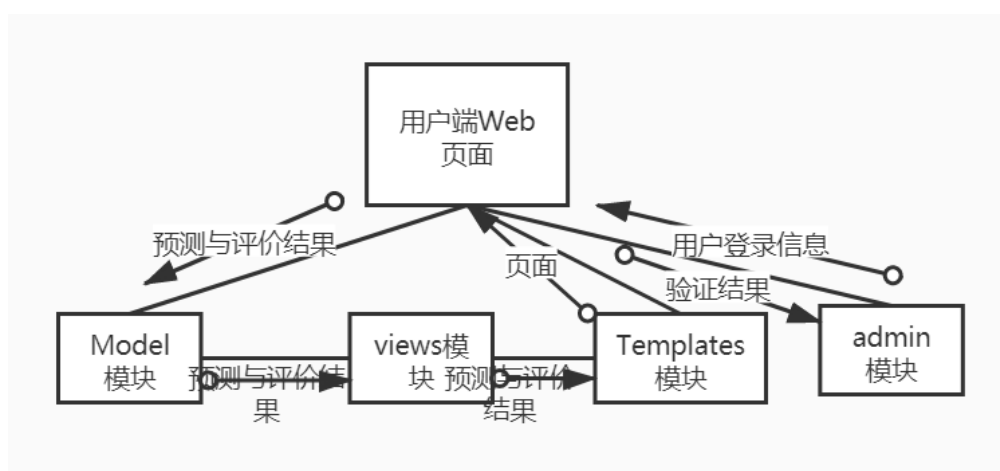
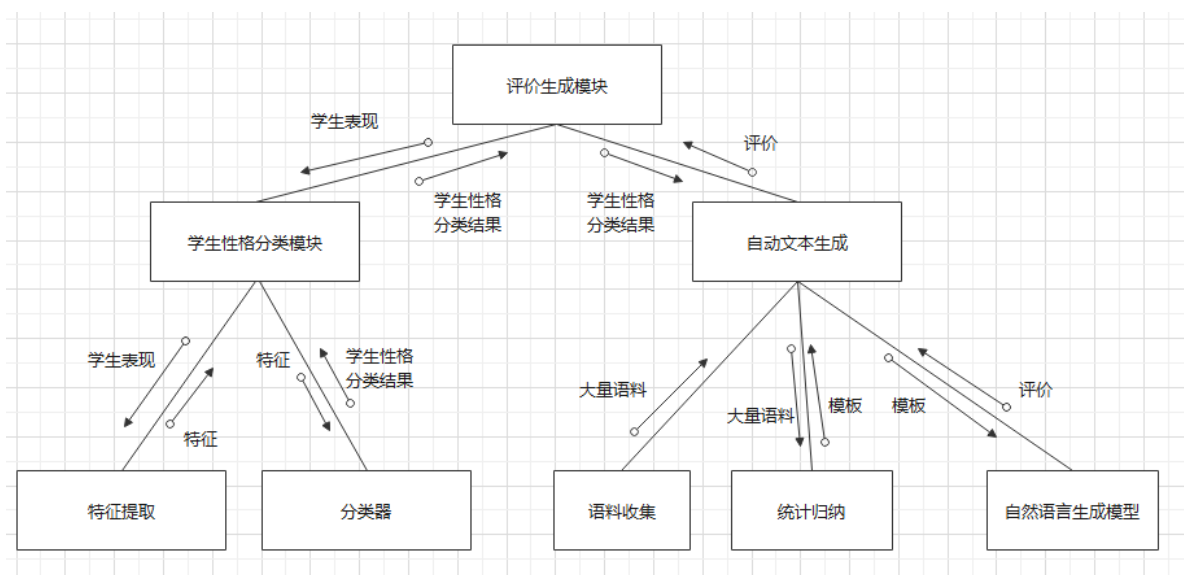
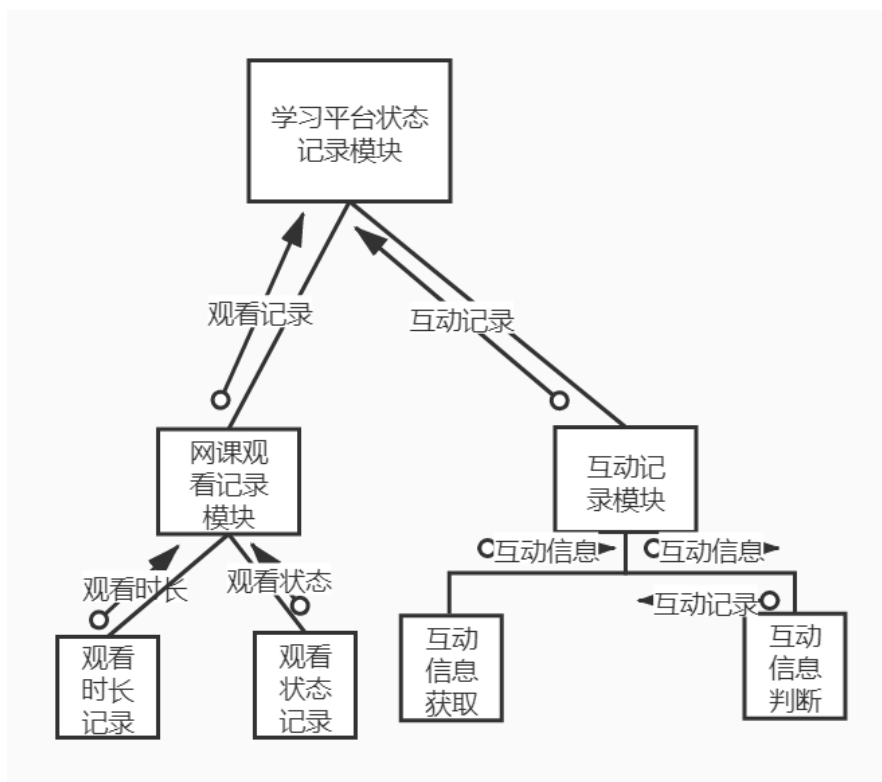
浏览器：Chrome、Firefox、Edge、Safari等主流浏览器。

## 2.3 基本设计概念和流程



## 2.4 结构





## 2.5 功能需求与程序关系

模块功能	成绩预测模块	个性化评价模块	用户端Web	事件记录模块	课堂测验模块	网课观看记录模块	互动记录模块	信息处理模块
成绩预测	√							
个性化评价		√						
用户登录			√					
结果展示			√					
事件捕获				√				
事件分析				√				
答题					√			
批改					√			
观看时长记录						√		
观看状态记录						√		
互动信息获取							√	
互动信息判断							√	
信息处理								√

## 2.6 人工处理过程

1. 课堂状态记录和学习平台状态记录等信息需要人工录入；
2. 用户登录需要用户本人操作。

## 2.7 尚未解决的问题

由于人工录入数据导致的无法实时更新的问题

## 3 接口设计

### 3.1 用户接口

用户接口	语法结构	软件回答信息
管理员登陆	以英文和汉字开头，不超过6个字符	进入后台管理界面
教师登录	以英文和汉字开头，不超过6个字符	进入主页面
学生登陆	以英文和汉字开头，不超过6个字符	进入主页面

### 3.2 外部接口

#### (1) 软件接口

web应用程序与web服务器程序之间的接口格式遵循WSGI(Web Server Gateway Interface)规范，以达到web应用程序与web服务器程序间的解耦。应用程序采用Django开发框架，服务器程序使用wsgiref模块进行开发。

其中，服务器程序使用Python提供的pymysql模块进行对数据库的连接和访问。

网络软件接口使用tcp传输协议，采用滑动窗口方式对数据进行网络传输及接收。

应用程序使用Django ORM，通过使用描述对象和数据库之间的映射的元数据，将程序中的对象自动持久化到数据库中

#### (2) 硬件接口

在输入方面，对于键盘、鼠标的输入，可用Python的标准输入/输出，对用户输入进行处理。

在输出方面，对于打印机的连接和使用，也可用Python的标准输入/输出对其进行处理。

### 3.3 内部接口

#### 3.3.1 算法部分

##### (1) 成绩预测模块

使用pymysql连接MySQL数据库服务器，获取原始数据，提取特征值，使用机器学习模型建模处理，将预测出的学生成绩存入MySQL数据库中。

##### (2) 个性化评价模块

使用pymysql连接MySQL数据库服务器，获取原始数据，提取特征值，使用机器学习模型建模处理，将学生的个性化评价指标存入MySQL数据库中。

#### 3.3.2 Web应用部分

框架：Django；设计模式：MTV

Model：数据存储层，处理所有数据相关的业务，和数据库进行交互，并提供数据的增删改查；

Template：模板层（也叫表现层）具体来处理页面的显示；

View：业务逻辑层，处理具体的业务逻辑，它的作用是连通Model层和Template。

Web应用部分的具体接口如下：



- (1) Model模块：数据库模型
- (2) Templates模块：前端css js html文件
- (3) admin模块：调用Django自带的后台管理系统，管理后台数据以及用户注册、登陆验证
- (4) views模块：从数据库提取数据，并传送给templates模块来进行前端展示

## 4 运行设计

---

本章描述对系统施加不同边界运行控制时所引起的各种不同的运行模块组合，包括运行模块组合、运行控制和运行时间。

### 4.1 运行模块组合

---

#### 4.1.1成绩预测功能：

- 成绩预测模块、用户端Web页面
- 机器学习预测成绩模块、信息处理模块
- 课堂状态记录模块、学习平台状态记录模块

#### 4.1.2个性化评价功能：

- 个性化评价模块、用户Web页面
- 评价生成模块、信息处理模块
- 课堂状态记录模块、学习平台状态记录模块

### 4.2 运行控制

---

#### 4.2.1成绩预测功能运行控制

- 管理员需要登陆用户名、验证用户名口令，经过系统核对以后才能进行相关操作。
- 教师需要登陆用户名、验证用户名口令，经过系统核对以后才能进行相关操作。
- 管理员登陆后需要对系统或者操作人员进行设定，确定登陆系统人员的身份。

#### 4.2.2个性化评价功能运行控制

- 管理员需要登陆用户名、验证用户名口令，经过系统核对以后才能进行相关操作。
- 教师需要登陆用户名、验证用户名口令，经过系统核对以后才能进行相关操作。

### 4.3 运行时间

---

系统的运行时间主要用在对学生学习信息数据库的访问方面，由于本身数据库容量较小，因此时间开销成本很小。

## 5 系统数据结构设计

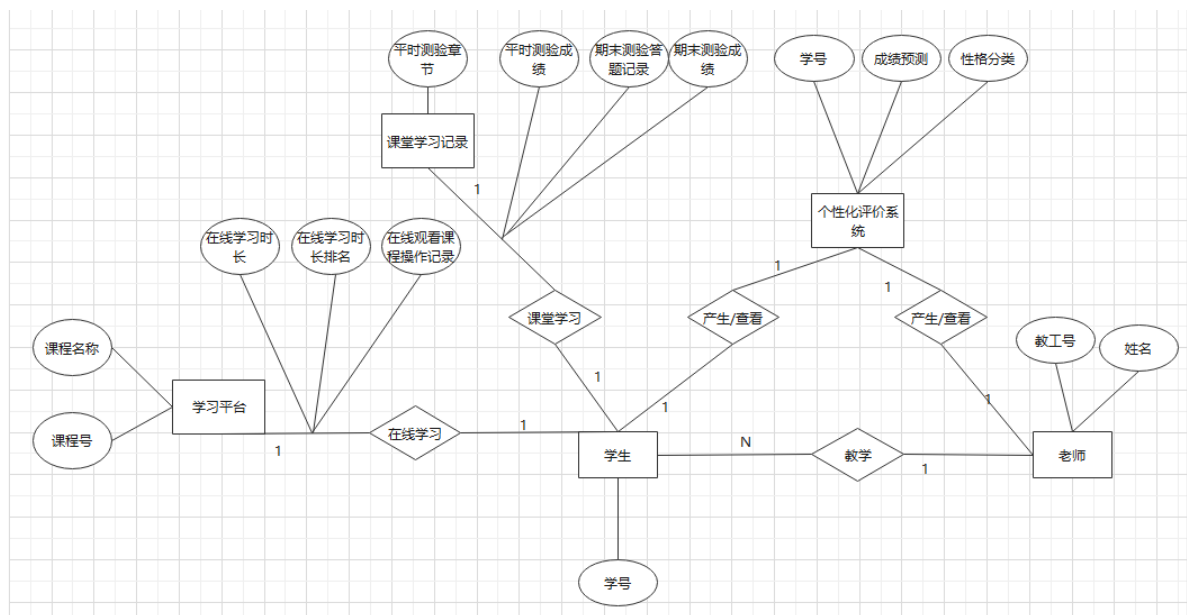
---

本节内容主要根据软件产品需求规格说明书和软件产品数据字典建立系统的逻辑模型。它是系统需求与物理实现的中间结构。

### 5.1 逻辑结构设计要点

---

E-R图



学生情况	学生情况=学生账户+平时测验成绩+学习时长+更新信息
学生信息	学生信息=学生账户学号+编号+学习时长
学习信息	学习信息=平时测验成绩+学习时长
状态分析信息	状态分析信息=在线课堂参与度+学习投入度
分类信息	分类信息=学生类别
预测、评价信息	预测、评价信息=平时表现评价+期末成绩预测
预测、反馈信息	预测、反馈信息=平时表现评价+期末成绩预测
平时测验成绩	平时测验成绩=0{数字}100
成绩预测评价报告	成绩预测评价报告=平时测验成绩+期末成绩预测
预测、反馈信息	预测、反馈信息=预测与实际的差值+相差的百分比
预测反馈报告	预测反馈报告=预测与实际的差值+相差的百分比+产生误差的原因
学习时长	学习时长=0{数字}课程总时长
在线课堂参与度	在线课堂参与度=0{数字}1
学习投入度	学习投入度=0{数字}1
学生类别	学生类别=[常缺席 表现较差 表现较好 表现优异]
期末成绩预测	期末成绩预测=0{数字}100

## 5.2 物理结构设计要点

本系统内所使用的每个数据均以数据库表的形式存储在SQL数据库中。

## 5.3 数据结构与程序的关系

数据结构为关系型数据库，所以在程序中可以用标准SQL语句与数据结构进行交互，交互过程中采用通用的数据反问接口。为了保持良好的程序架构，对数据库访问采用DAO设计模式实现，提高维护性和扩张性。

	[程序1]	[程序2]	[.....]	[程序m]
[数据结构1]	√			
[数据结构2]	√	√		
⋮				
[数据结构n]		√		√

具体的例如创建学生信息表的程序用SQL表示就是：

```

1 CREATE TABLE [dbo].[DeptInfo](
2 [bmID] [int] IDENTITY (1,1) NOT NULL,
3 [bmMC] [varchar](10) NULL,
4 [bmNO] [varchar](30) NULL,
5 [DepFatherCode] [varchar](30) NULL,
6 CONSTRAINT [PK_DEPTINFO] PRIMARY KEY CLUSTERED.

```

## 6 系统出错处理设计

### 6.1 出错信息

#### 6.1.1 输入错误

错误名称	解决方法
1.用户名和密码出错	提示输入不正确，检查信息是否正确，并返回。
2.验证码输入错误	登陆失败，请重新输入。
3.注册密码低于6位	提示低于6位信息，并重新输入

#### 6.1.2 Web服务器错误处理设计

错误名称	解决方法	补救
1.不合法的Url请求	提示错误，返回登录界面	mvc的请求错误控制
2.程序出错抛出异常	输入Log日志中的错误信息	判断错误信息，返回登录界面
3.数据访问量的过大	返回请求失败信息	针对需求，重新选择服务器
4.登陆调用Web服务器方法	登录页面，要求用户登录	mvc拦截所有请求，判断用户是否登录，必须登录才能进行其他操作。
5.服务器无法访问	返回请求失败信息	重启服务器，尝试重连服务器
6.数据更新过程访问结果不准确	返回提示数据处理中，请稍后查询	中断查询请求，数据更新完毕后接受查询
7.数据丢失	返回请求失败信息	定期备份数据，出错时及时恢复备份，并且依靠日志文件使系统再启动

## 6.2 补救措施

---

### 6.2.1 后备技术

现在的技术一般是进行系统数据库的实时备份和备份服务器。当数据库瘫痪时，通过快速进行系统库的切换，这样可以有效的减少数据的丢失。当用户量大时（目前此系统还不需考虑），也可以使用备用服务器，两个服务器进行同步数据存储，如果一台服务器出错时，可以转换到另一台进行相同的操作，让用户不需要因瘫痪而面临网页打不开、数据丢失等相关问题。

### 6.2.2 降效技术

如今为了让系统用户能够自行解决运行过程中遇到的问题，大多采取小型服务器作为后备技术。这项技术在现有的开发系统中使用很广泛，但备用服务器从价格和工资效率的考虑上都不如主服务器，但从数据的完整性和完全出发，可以让维修人员有更多时间进行系统的维护。

### 6.2.3 恢复及再启动技术

采用低端的服务器，同时进行数据库的实时备份。这样在系统维护好完成后，可以很快的进行数据同步，同时也可以很快的进行系统的运行。

## 6.3 系统维护设计

---

由于此系统较小，维护工具比较简单，仅依靠数据库的一些基本维护，故没有外加维护模块。