**《教务管理系统》**

**概要设计说明书**

**班级：信1604-2**

**姓名：张孟婕**

**学号：20163795**

**目录**

**[1.总体设计](#_Toc13012_WPSOffice_Level1)** **[3](#_Toc13012_WPSOffice_Level1)**

[1.1需求规定](#_Toc24613_WPSOffice_Level2) [3](#_Toc24613_WPSOffice_Level2)

[1.2运行环境](#_Toc31732_WPSOffice_Level2) [3](#_Toc31732_WPSOffice_Level2)

[1.3基本设计概念和处理流程](#_Toc28003_WPSOffice_Level2) [3](#_Toc28003_WPSOffice_Level2)

[1.4系统体系结构](#_Toc22323_WPSOffice_Level2) [3](#_Toc22323_WPSOffice_Level2)

**[2.用户接口](#_Toc24613_WPSOffice_Level1)** **[4](#_Toc24613_WPSOffice_Level1)**

**[3.运行设计](#_Toc31732_WPSOffice_Level1)** **[5](#_Toc31732_WPSOffice_Level1)**

[3.1运行模块组合](#_Toc31065_WPSOffice_Level2) [5](#_Toc31065_WPSOffice_Level2)

[3.2运行控制](#_Toc26196_WPSOffice_Level2) [5](#_Toc26196_WPSOffice_Level2)

[3.3运行时间](#_Toc3726_WPSOffice_Level2) [5](#_Toc3726_WPSOffice_Level2)

**[4.系统数据结构](#_Toc28003_WPSOffice_Level1)** **[5](#_Toc28003_WPSOffice_Level1)**

[4.1逻辑结构设计](#_Toc19465_WPSOffice_Level2) [5](#_Toc19465_WPSOffice_Level2)

[4.2物理结构设计](#_Toc9745_WPSOffice_Level2) [6](#_Toc9745_WPSOffice_Level2)

**[5.系统出错处理设计](#_Toc22323_WPSOffice_Level1)** **[6](#_Toc22323_WPSOffice_Level1)**

[5.1出错信息](#_Toc453_WPSOffice_Level2) [6](#_Toc453_WPSOffice_Level2)

[5.2补救措施](#_Toc3506_WPSOffice_Level2) [6](#_Toc3506_WPSOffice_Level2)

## 1.总体设计

### 1.1需求规定

教务管理系统可分为学生信息管理系统和教师管理信息系统，系统开发的整体任务是实现学校教师和学生信息管理的系统化、规范化、自动化和智能化，从而达到提高学校管理效率的目的。

本阶段目的在于明确系统的数据结构和软件结构，此外总体设计还将给出内部软件和外部系统部件之间的接口定义，各个软件模块的功能说明，数据结构的细节以及具体的装配要求。

### 1.2运行环境

软件基本运行环境为Windows10环境。

### 1.3基本设计概念和处理流程

概要说明书的目的在于明确系统的数据结构和软件结构，设计外部软件和内部软件的接口，说明各个软件模块的功能说明，数据结构的细节等。系统的总体处理流程如图1-1所示：

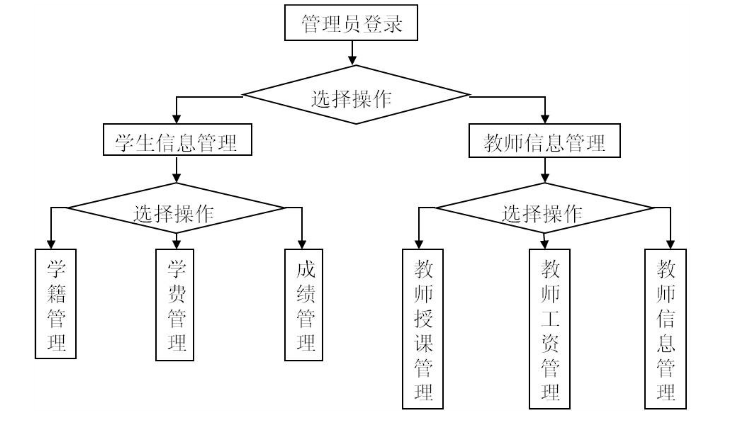


图1-1系统的总体处理流程

### 1.4系统体系结构

用一览表及框图的形式说明本系统的系统元素（各层模块、子程序、公用程序等）的划分，扼要说明每个系统元素的标识符和功能，分层次地给出各元素之间的控制与被控制关系。

本系统的体系架构如图1-2所示：

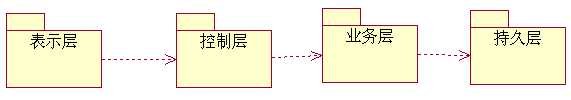
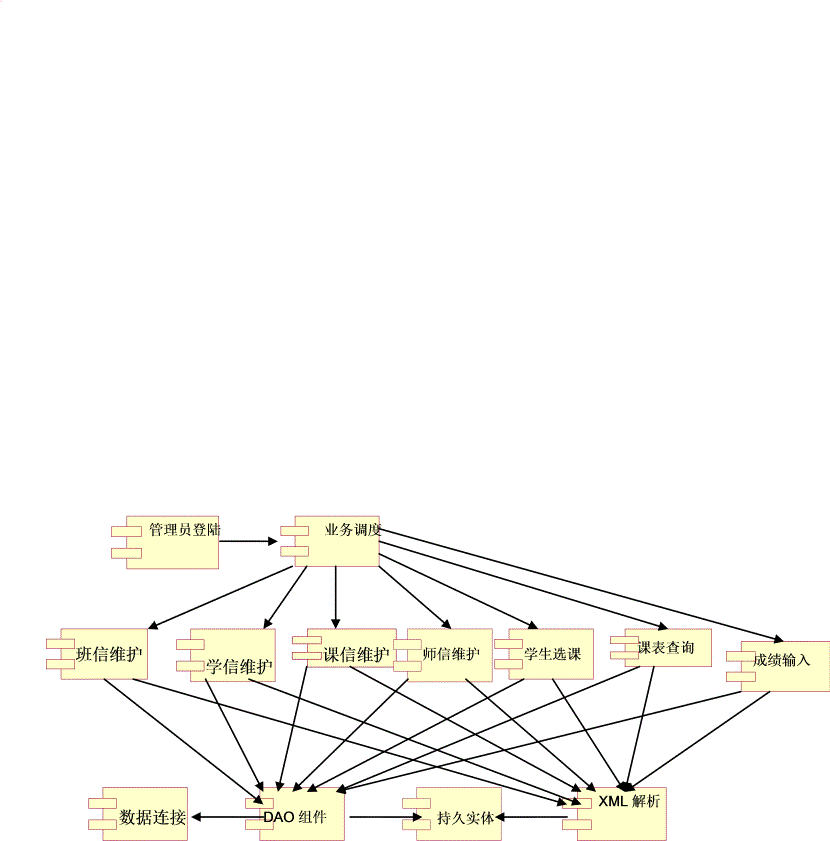


图1-2系统体系架构

本系统体系结构大致可以定义为：客户机层上的表示层主要是通过Struts框架实现的，由显示视图产生一个请求。请求被ActionServlet（控制器）接收，它在struts-config.xml文件中寻找请求的URI，找到对应的Action类后，Action类执行相应的业务逻辑。Action类执行建立在模型组件基础上的业务逻辑，模型组件是和应用程序关联的。一旦Action类处理完业务逻辑，它把控制权返回给ActionServlet，Action类提供一个键值作为返回的一部分，它指明了处理的结果。ActionServlet使用这个键值来决定在什么视图中显示Action的类处理结果。当ActionServlet把Action类的处理结果传送到指定的视图中，请求的过程也就完成了。中间业务层是通过Spring框架实现的，首先建立一个BaseAction,它继承了Action类，而其他定义的Action都要继承这个BaseAction。这个BaseAction需要导入AppContext工具类，这个AppContext需要导入Spring中org.springframework.context.support.\*；这样一个继承BaseAction的Action，就可以getXXXService()的方法得到某一个service的实例-----服务定位器的设计模式。持久（PO）层是由hibernate架构实现的，它包括关于整体数据库的hibernate.cfg.xml文件、每个表的JavaBean类和每个表的hbm.xml文件，通过Spring集成模板HibernateTemplate提供DAO来使用PO。在Spring的配置文件（applicationContext.xml）中配置sessionFactory的bean来管理hibernate。

本系统组件图如图1-3所示：



## 图1-3系统组件图

## 2.用户接口

|  |  |  |
| --- | --- | --- |
| 用户接口 | 语法结构 | 软件回答信息 |
| 教务系统管理员登录 | 以英文和汉字开头，不超过6个字符 | 进入主界面 |
| 信息管理相关操作 |  | 进行相应的操作 |

## 3.运行设计

### 3.1运行模块组合

具体软件的运行模块组合为程序多窗口的运行环境，各个模块在软件运行过程中能较好的交换信息，处理数据。

### 3.2运行控制

软件运行时有较友好的界面，基本能够实现用户的数据处理要求。

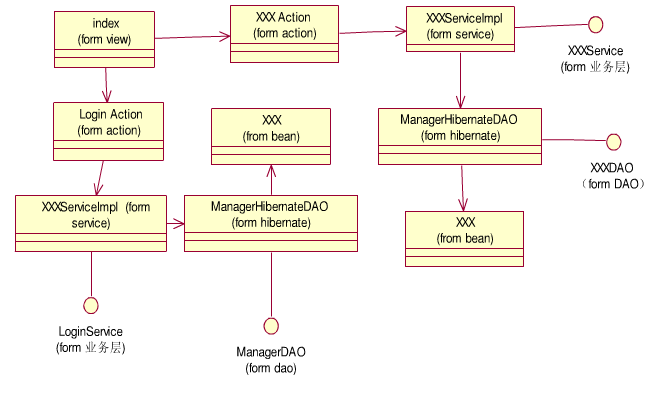
### 3.3运行时间

系统的运行时间基本可以达到要求。

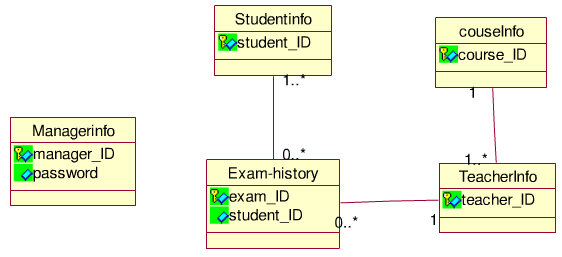
## 4.系统数据结构

### 4.1逻辑结构设计

根据系统需求，把系统分为登录模块、学生管理模块和教师管理模块等。系统的“粗粒度”的概要设计类图：



系统的数据库表关系图（ER图）：



### 4.2物理结构设计

系统的物理结构具体由数据库来设计与生成，故此处略。

## 5.系统出错处理设计

### 5.1出错信息

|  |  |  |
| --- | --- | --- |
| 错误登录信息 | 系统提示错误登录信息 | 系统返回登录界面 |
| 输入错误的数据 | 系统提示数据错误信息 | 提示重新输入数据 |
| 打印不成功 | 系统提示打印不成功信息 | 进行重新打印操作 |

### 5.2补救措施

由于数据在数据库中已经有备份，故在系统出错后可以依靠数据库的恢复功能，并且依靠日志文件使系统再启动，就算系统崩溃数据也不会丢失或遭到破坏。但有可能占用更多的数据存储空间，权衡措施由用户来决定。