# 目 录

1 引言 ............................................................................................................................................... 3

1.1 编写目的 ............................................................................................................................. 3

1.2 背景 ..................................................................................................................................... 3

1.3 定义 ..................................................................................................................................... 3

1.4 参考资料 ............................................................................................................................. 3

2 任务概述 ................................................................................................................................ 3

2.1 目标 ..................................................................................................................................... 3

2.2 运行环境 ............................................................................................................................. 4

2.3 需求概述 ............................................................................................................................. 4

3 总体设计 ........................................................................................................................................ 5

3.1 基本设计概念和处理流程 ................................................................................................. 5

3.2 结构 ..................................................................................................................................... 8

3.3 功能分配 ............................................................................................................................. 9

4 接口设计 ...................................................................................................................................... 11

4.1 用户接口 ........................................................................................................................... 11

4.2 外部接口 ........................................................................................................................... 11

4.3 内部接口 ........................................................................................................................... 11

5 运行设计 ...................................................................................................................................... 11

5.1 运行模块组合 ................................................................................................................... 11

5.2 运行控制 ........................................................................................................................... 12

5.3 运行时间 ........................................................................................................................... 12

6 系统数据结构设计 ...................................................................................................................... 12

7 系统出错处理设计 ...................................................................................................................... 12

7.1 出错信息 ........................................................................................................................... 12

7.2 补救措施 ........................................................................................................................... 13

## 1 引言

1.1 编写目的

为明确 HUST 学分制教务管理系统的系统结构、安排项目规划与进度、制定详细测试

计划、组织软件开发与测试，特撰写本文档。

本文档供项目经理、设计人员、开发人员参考。

1.2 背景

开发软件名称： HUST 学分制教务管理系统

项目任务提出者：

项目开发者： ZOWL 团队

用户：大学学生，教务管理人员

实现软件单位： ZOWL 团队

项目与其他软件，系统的关系：

该系统需要在学生成绩管理系统， 学生选课系统等已有系统直接复制数据， 所以在定义数据时要与已有系统相兼容。

本系统采用 B/S，服务器端采用 Microsoft Server 2003 为操作系统的工作站， 是采用 MicrosoftSQL Server 2005 的为开发软件的数据库服务程序。

1.3 定义

SQL ：结构化查询语言，关系数据库查询的标准语言；

SQL SERVER 2005 ：微软公司生产的 SQL SERVER 数据库管理系统；

1.4 参考资料

## 2 任务概述

2.1 目标

因大量用户需要使用本系统的前台，所以决定采用 B/S 体系来设计本系统。

教务管理人员登录本系统后， 系统列出多有可供使用的功能模块， 主要有八大模块： 学

籍管理、 注册管理、 教师信息管理、 学生信息管理、 成绩管理、 课程管理、 选课与排课管理、教务通知管理。教务管理人员工作时可以选择相应的功能模块进行快捷方便的教务管理。

学生随时随地可以利用互联网，来进行查询个人基本信息、成绩查询、注册状态查询、

课表下载，还可以查看教务人员发布的教务通知，方便地了解成绩、课程、通知等信息。

教师用户同样可以随时随地进行进行教务通知查看、同时进行成绩录入工作。

HUST 学分制教务管理系统 概要设计文档 编号 :hust\_gaiyaosheji

本系统基于 UML 建模， 概要设计应输出用例图， 主要用例的顺序图， 系统的主要类图。

2.2 运行环境

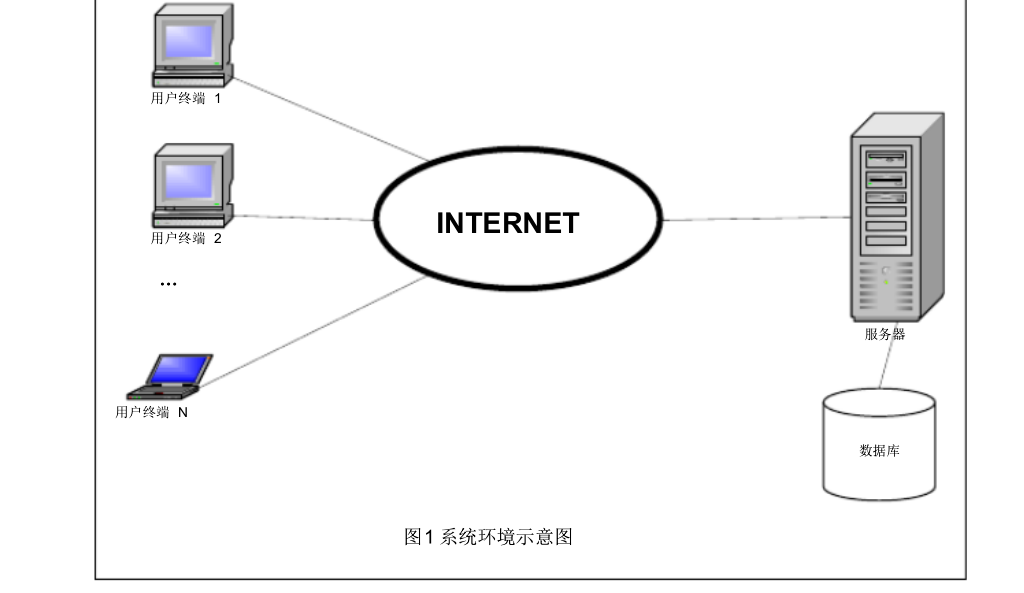
硬件及软件环境：

1 台服务器： PIII1.2G 双 CPU ，SCSI 双硬盘镜像， 1G 以上内存， Windows NT ，

Apache+JSP+SQLSERVER 2005 服务器；

客户机为普通支持浏览器的 PC 机；

系统运行环境示意图如下：



2.3 需求概述

旨在开发一个统一的网上教务信息管理平台，利用信息化手段辅助教学管理，达到提升

我校教学管理信息化水平的目的。 教务管理人员能够更方便地进行教务管理工作， 包括学籍

管理，注册管理，教师信息管理，学生信息管理，成绩管理，课程管理，选课与排课管理，

以及教务通知管理等； 学生用户能够利用该系统进行成绩查询， 个人选课操作， 查看教务通知，课表下载等；教师则能够进行成绩录入，查看教务通知等。

HUST 学分制成绩管理系统主要需求列表如下：

学籍管理功能。包括转专业管理、停学、休学、复学、退学、延长学习年限。

在进行学籍管理各子功能的操作时， 操作界面应统一、 直观。 其中转专业时应包括的信

息为（学号、转专业记录号、原转出院系、原转出专业、原转出班级、现转入院系、现转入

专业），而停、休、复、退学则内容一致（学号、姓名、院系、专业、停学时间、停学原因、

HUST 学分制教务管理系统 概要设计文档 编号 :hust\_gaiyaosheji,医院意见、班主任意见、院系意见、备注） 。

注册管理功能 。包括个人注册查询与注册操作、班级批量注册查询与操作。

个人注册管理时，只显示个人的注册状态及相关信息；而班级注册管理时则显示班级各

成员的注册信息列表，能够支持统一处理。

教师信息管理。包括教师信息查询、修改、添加和删除。

学生信息管理。包括学生信息查询、修改、添加和删除。

成绩管理。包括成绩录入、成绩报表（个人学期、学年、综合成绩报表；班级学年加权、

综合加权成绩表、学分统计表；年级加权成绩一览表） 。

各种成绩报表的格式参见《需求规范说明书》 。

课程管理。包括课程信息查询、修改、添加和删除。

选课与排课管理。包括班级批量选课、个人选课；课表导入（上传） 、课表导出（下载） 。

不需要提供排课功能，只需要实现课表文件的导入与导出。即实现上传与下载功能。

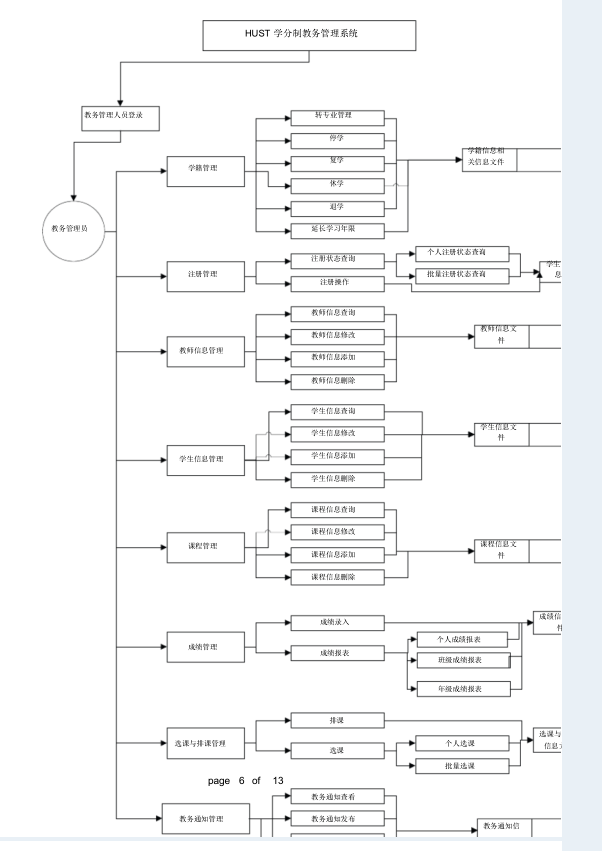
教务通知管理。包括通知查询、修改、添加和删除。

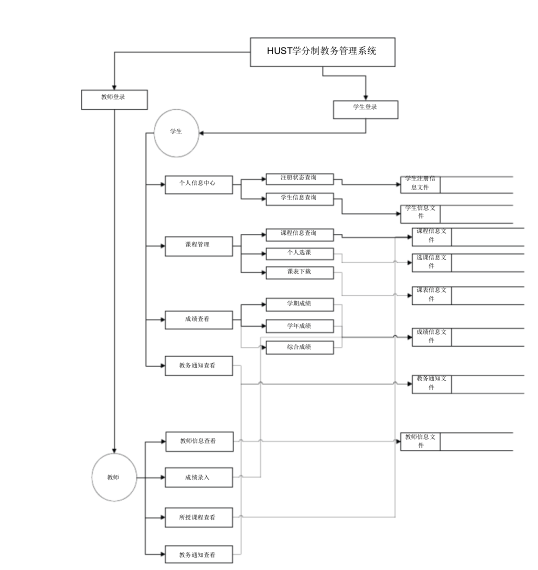
教务管理人员进行通知修改、添加和删除。教师和学生进行查看。

## 3 总体设计

3.1 基本设计概念和处理流程

针对本系统的教务管理人员、 学生、教师三种不同的角色， 系统的处理流程如下图所示：



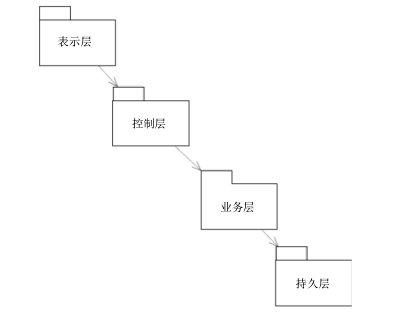


3.2 结构

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 3 所示：



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

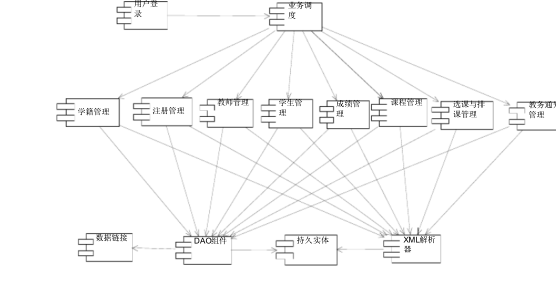
执行建立在模型组件基础上的业务逻辑，模型组件是和应用程序关联的。一旦 Action 类处

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。

教务管理员拥有所有功能权限，学生和教师所有的功能均为教务管理员功能的子集。

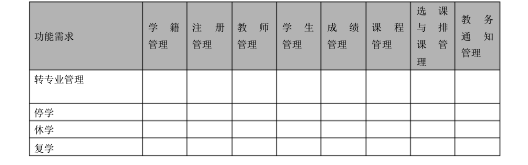
故系统只对管理员的角度进行设计，学生和教师模块可以直接复用相关模块完成其功能。

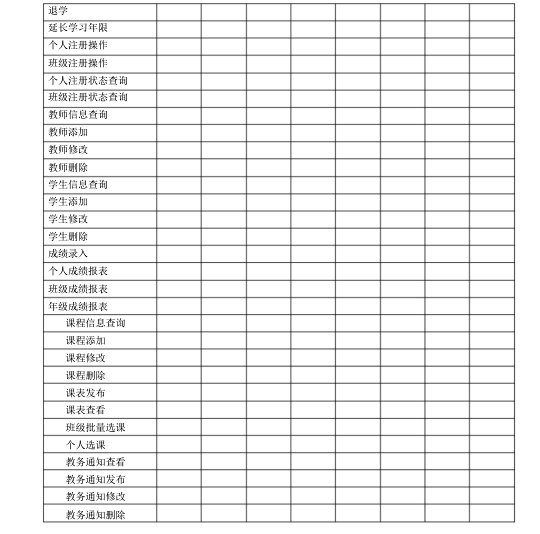
本系统组件图如图 4 所示：



3.3 功能分配

本系统中各功能需求与程序模块（组件）之间的关系如下图所示：





## 4 接口设计

4.1 用户接口

在用户界面部分，根据需求分析的结果，用户需要一个用户友善界面。在界面设计上，应

做到简单明了，易于操作，并且要注意到界面的布局，应突出的显示重要以及出错信息。可

以使用 Dreamweaver 网页设计工具直接拖曳出美观、简介、友善的用户接口。其中针对教

务管理人员的界面要做到操作简单， 易于管理。 在设计上采用逐级下拉式菜单方式， 但菜单层数最多不能超过 4 层。同时，运行出错时应以标准形式给出出错提示。总的来说，系统的用户界面应作到可靠性、简单性、易学习和使用。

4.2 外部接口

服务器端程序可 DAO （数据访问对象）应用程序编程接口（ API），对 SQL SERVER

数据库进行所有的访问。

服务器程序上可使用 SQL SERVER 的对数据库的备分命令，以做到对数据的保存。

在网络软件接口方面， 使用一种无差错的传输协议， 采用滑动窗口方式对数据进行网络

传输及接收。

4.3 内部接口

教师信息管理模块。对其他模块提供对课程的查询接口。

学生信息管理模块。对其他模块提供对学生信息查询的接口，包括按学号查询和按姓名查

询和按班级查询三种。

课程管理模块。对其他模块提供对课程信息查询的接口，包括按课程编号查询、按课程名

查询等。

## 5 运行设计

5.1 运行模块组合

客户机程序在有输入时启动接收数据模块， 通过各模块之间的调用， 读入并对输入进行

格式化。 在接收数据模块得到充分的数据时， 将调用网络传输模块， 将数据通过网络送到服务器， 并等待接收服务器返回的信息。 接收到返回信息后随即调用数据输出模块， 对信息进

HUST 学分制教务管理系统 概要设计文档 编号 :hust\_gaiyaosheji

行处理，产生相应的输出。

服务器程序的接收网络数据模块必须始终处于活动状态。 接收到数据后， 调用数据处理

/查询模块对数据库进行访问，完成后调用网络发送模块，将信息返回客户机。

5.2 运行控制

运行控制将严格按照各模块间函数调用关系来实现。在各事务中心模块中，需对运行控

制进行正确的判断，选择正确的运行控制路径。

在网络传方面，客户机在发送数据后，将等待服务器的确认收到信号，收到后，再次等

待服务器发送回答数据， 然后对数据进行确认。 服务器在接到数据后发送确认信号， 在对数据处理、访问数据库后，将返回信息送回客户机，并等待确认。

5.3 运行时间

在软体的需求分析中， 对运行时间的要求为必须对作出的操作有较快的反应。 网络硬件

对运行时间有最大的影响， 当网络负载量大时， 对操作反应将受到很大的影响。 所以将采用高速 ATM 网络，实现客户机与服务器之间的连接，以减少网络传输上的开销。其次是服务器的性能，这将影响对数据库访问时间即操作时间的长短， 影响加大客户机操作的等待时间，

所以必须使用高性能的服务器。硬件对本系统的速度影响将会大于软件的影响。

## 6 系统数据结构设计

系统数据结构设计在《数据库设计说明书》进行描述。

## 7 系统出错处理设计

7.1 出错信息



7.2 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， 就算系统崩溃用户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。