软件工程系列课程教学辅助网站

软件概要设计说明



|  |  |
| --- | --- |
| 编制： | G17 |
| 日期： | 2018-1-7 |
| 版本： | v0.1.0 |

**PRD-G17**

组长：蒋家俊

组员：李捷、朱秉、厉佩强、周盛

日期：2018.1.7

**文档修订记录**

| **版本** | **修订日期** | **修订人** | **修订说明** | **修订状态** | **审批日期** | **审核人** | **批准人** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0.1.0 | 2018-1-3 | 厉佩强 |  | S |  |  |  |
| 0.1.0.1 | 2018-1-7 | G17 |  | A |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

修订状态：S--首次编写，A--增加，M--修改，D--删除；

日期格式：YYYY-MM-DD。

目录

[1 引言 4](#_Toc503103333)

[1.1 编写目的 4](#_Toc503103334)

[1.2 背景 4](#_Toc503103335)

[1.3 读者对象 4](#_Toc503103336)

[1.4 参考资料 4](#_Toc503103337)

[1.5 参考文档 4](#_Toc503103338)

[1.6 范围 4](#_Toc503103339)

[2 总体设计 5](#_Toc503103340)

[2.1 功能描述 5](#_Toc503103341)

[2.1.1 运行环境 5](#_Toc503103342)

[2.2 基本设计概念和处理流程 5](#_Toc503103343)

[2.2.1 设计思想 5](#_Toc503103344)

[2.2.2 关键技术与算法 7](#_Toc503103345)

[2.2.3 关键数据结构 7](#_Toc503103346)

[2.3 人工处理过程 7](#_Toc503103347)

[2.4 尚未解决的问题 7](#_Toc503103348)

[3 接口设计 8](#_Toc503103349)

[3.1 用户接口 8](#_Toc503103350)

[3.2 外部接口 9](#_Toc503103351)

[3.2.1 硬件接口 9](#_Toc503103352)

[3.2.2 软件接口 9](#_Toc503103353)

[3.2.3 通信接口 9](#_Toc503103354)

[3.3 内部接口 9](#_Toc503103355)

[4 运行设计 10](#_Toc503103356)

[4.1 运行模块组合 10](#_Toc503103357)

[4.2 运行控制 10](#_Toc503103358)

[4.3 运行时间 10](#_Toc503103359)

[5 系统数据结构设计 11](#_Toc503103360)

[5.1 说明 11](#_Toc503103361)

[5.2 逻辑设计结构要点 11](#_Toc503103362)

[5.3 物理结构设计要点 12](#_Toc503103363)

[5.4 数据结构与程序关系 13](#_Toc503103364)

[6 系统出错处理设计 14](#_Toc503103365)

[6.1 出错信息 14](#_Toc503103366)

[6.2 补救措施 14](#_Toc503103367)

[7 系统维护设计 16](#_Toc503103368)

# 引言

## 编写目的

本文档旨在说清楚“总体实现方案”，确定软件系统的总体布局，各个子模块的功能和模块间的关系，与外部系统的关系。从全局的角度说一下 总体结构、功能、处理流程、有哪些模块、模块间的关系，总体说明外部用户、软、硬件接口。分析每个模块做什么，怎么做。以便于接下来的详细设计以及程序编写。

## 背景

项目名称：软件工程系列教学辅助网站

任务提出者：杨枨、候宏仑

开发者：蒋家俊，李捷，厉佩强，周盛，朱秉

用户：软件工程专业师生以及其他对软件工程感兴趣的用户

运行网络：校园网

## 读者对象

项目经理：蒋家俊

开发人员：李捷，厉佩强，周盛，朱秉

相关测试人员

## 参考资料

软件项目管理-------机械工业出版社

软件概要设计说明书(GB8567—88)

## 参考文档

PRD-G17-需求规格说明书1.0

## 范围

该软件工程系列课程教学辅助网站系统主要有，课程信息发布，课程资源下载以及课程交流互动。其中课程信息包括，教师介绍，课程公告，课程资料以及课程链接，课课程资源下载包括，课程资料下载，答疑记录下载，课程交流包括课程答疑，课程论坛以及相应的课程信息搜索，

# 总体设计

## 功能描述

系统主要有课程模块、论坛模块、系统功能、管理员模块四大模块构成。

### 运行环境

#### 软件环境

服务端：操作系统选用Red Hat Enterprise Linux 数据库选用SQL server2005Web。客户端：操作系统选用Windows Xp/Vista/7/8/10、Linux、Mac OS等 数据库选用SQL server2005。

#### 硬件环境

服务器选用2个CPU，16核，内存6G，硬盘1T（杨枨老师提供）。

#### 开发环境

操作系统选用Windows Xp/Vista/7/8/10、Linux、Mac OS等，数据库选用SQL server2005，使用java语言和HTML5语言对软件予以实现。

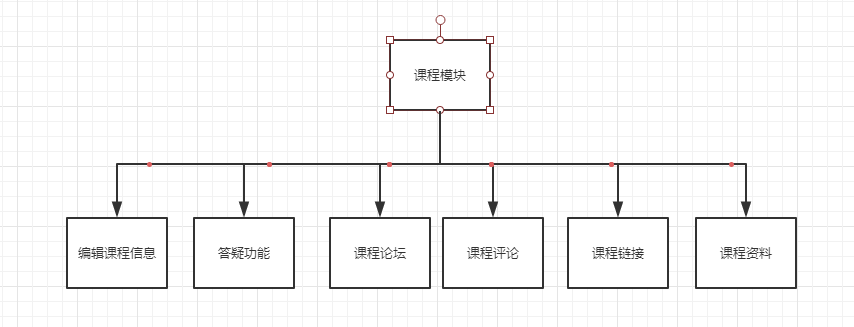
## 基本设计概念和处理流程

### 设计思想

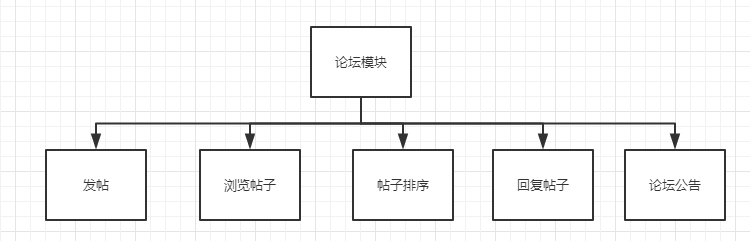
#### 系统构思

系统主要有课程模块、论坛模块、系统功能、管理员模块四大模块构成。

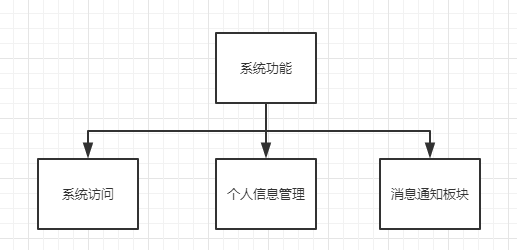
课程模块：



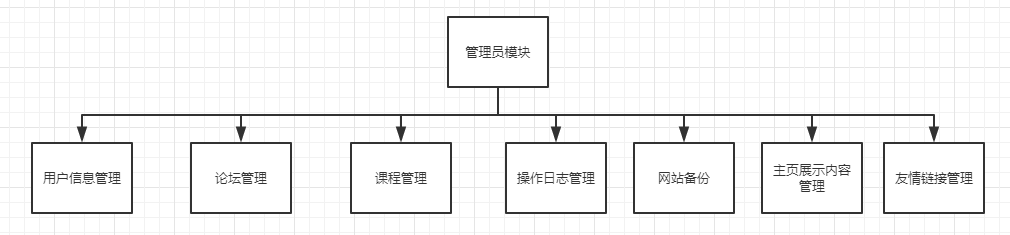
论坛模块：



系统功能：



管理员模块：



### 关键技术与算法

本系统采用典型的J2EE三层结构，分为表现层、中间层（即业务逻辑层）和数据服务层。三层体系将业务规则、数据访问及合法性校验等工作放在中间层处理。客户端不直接与数据库交互，而是通过组件与中间层建立连接，再由中间层与数据库交互。

#### JSP

本系统中，表现层采用传统的JSP技术。

#### Struts

在本系统中，Struts负责流程控制。Struts 是一种J2EE MVC实现方式，它的核心是一个MVC风格的控制器。 Struts 控制器搭起了Model 和View之间的桥梁。

#### Spring

在本系统中，Spring负责业务流转。Spring将Web层、Service层、DAO层及PO无缝整合。它的作用贯穿了整个中间层。

#### Ibatis

在本系统中，Ibatis负责数据库操作。主要通过连接池等实现数据库操作的一致性和正确性，防止垃圾数据的产生而危害到整个数据库，它还能系统的管理那些与直接操作数据库的文件，这样当更改数据库时，将不需要重新开发整个系统，只需修改接口即可，很大程度的提高了软件的重用性。

### 关键数据结构

本系统的关键数据结构为数据库，与数据库相关的E-R图等将在第五部分列出。

## 人工处理过程

## 尚未解决的问题

无

# 接口设计

## 用户接口



## 外部接口

### 硬件接口

服务器硬件接口

|  |  |
| --- | --- |
| **项目** | **接口信息** |
| 主频 | HPZ800 |
| 磁盘 | SCSI接口、转速10000转/秒以上 |
| 网络 | 浙江大学城市学院校园网 |
| 备份 | 数据备份采用RAID5 |

客户端硬件接口

|  |  |
| --- | --- |
| **项目** | **接口信息** |
| 主频 | 当前主流配置即可 |
| 显卡 | 分辨率1024\*768以上 |
| 网络 | 浙江大学城市学院校园网 |

### 软件接口

服务器软件接口

|  |  |
| --- | --- |
| **项目** | **接口信息** |
| 操作系统 | Red Hat Enterprise Linux |
| 服务器软件 | Apache5.5 |
| 数据库 | MySql5 |

客户端软件接口

|  |  |
| --- | --- |
| **项目** | **接口信息** |
| 操作系统 | Windows XP/Vista/7/8/10、Linux、Mac OS等 |
| 浏览器 | Internet Explore、Firefox、Chrome等 |

### 通信接口

|  |  |
| --- | --- |
| **项目** | **接口信息** |
| 网络环境 | 浙江大学城市学院校园网 |

## 内部接口

内部接口方面，接口传递的信息将是以数据结构封装了的数据，以参数传递或返回值的形式在各模块间传输。各模块之间采用函数调用、参数传递、返回值的方式进行信息传递。

# 运行设计

## 运行模块组合

各个模块之间相对独立，程序的可移植性好。各个模块之间主要以传递数据项的引用来实现模块之间的合作和数据共享。

客户端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，读入并输入，然后调用网络传输模块，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，将信息返回客户端。

## 运行控制

只要符合操作说明书，用户可以自由控制，不再限定用户输入，异常程序内部进行处理，给出相应的提示信息。

## 运行时间

系统运行时间：

7\*12h，每天12：00到24：00

# 系统数据结构设计

## 说明

对于程序开发底层的数据结构，表结构进行定义，便于日后开发工作高效快捷的进行。

## 逻辑设计结构要点

1. 用户信息表

用户（用户编号，用户账号，用户真实姓名，密码，头像，电话，邮箱，注册时间，收藏编号，用户类型）

1. 用户收藏表

用户收藏（编号，用户编号，收藏时间，收藏的帖子编号）

1. 管理员信息表

管理员（管理员编号，账号，密码，Email，注册时间）

1. 评论表

评论表（编号，用户账号，帖子编号，评论内容）

## 物理结构设计要点

用户信息表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | user\_id | 用户编号 | int | 11 | 1 | 否 | 是 | 自动递增 |
| 2 | username | 用户账号 | varchar | 5-30 | NILL | 否 |  |  |
| 3 | ralname | 用户姓名 | varchar | 1-20 | NILL | 否 |  |  |
| 4 | user\_pwd | 密码 | varchar | 6-16 | 123456 | 否 |  |  |
| 5 | user\_picname | 头像 | varchar | 32 | NILL |  |  |  |
| 6 | phone | 电话 | varchar | 11 | NILL | 否 |  |  |
| 7 | email | 邮箱 | Varchar | 5-30 | NILL | 否 |  |  |
| 8 | regtime | 注册时间 | Datetime |  | 2018-01-01 00:00 |  |  |  |
| 9 | collect\_id | 收藏编号 | Int | 11 | 1 |  |  | 自增 |
| 10 | Select\_identity | 用户类型 | Varchar | 32 | NILL | 否 |  |  |

评论信息表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | Comment\_id | 评论编号 | Int | 11 | 1 | 否 | 是 | 自动递增 |
| 2 | comment\_userid | 用户编号 | int | 11 |  | 否 |  |  |
| 3 | comment\_object | 评论的帖子编号 | int | 11 |  | 否 |  |  |
| 4 | comment\_content | 评论内容 | text | 1-140 |  |  |  |  |
| 5 | comment\_time | 评论时间 | datetime |  |  | 否 |  |  |

收藏表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | collect\_id | 收藏编号 | int | 11 | 1 | 否 | 是 | 自动递增 |
| 2 | user\_id | 用户编号 | int | 11 | 1 | 否 |  |  |
| 3 | collect\_time | 收藏时间 | Datetime |  |  |  |  |  |
| 4 | collect\_goodsid | 收藏的帖子编号 | int | 11 |  |  |  |  |

管理员信息表

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 字段含义 | 类型 | 长度 | 默认值 | 允许空 | 主键 | 说明 |
| 1 | Admin\_id | 管理员编号 | int | 11 | 1 | 否 | 是 | 自动递增 |
| 2 | Admin\_username | 管理员名称 | varchar | 32 | admin | 否 |  |  |
| 3 | Admin\_pwd | 密码 | varchar | 6-16 | 123456 |  |  |  |
| 4 | Admin\_email | 管理员邮箱 | varchar | 5-30 | NULL |  |  |  |

## 数据结构与程序关系

数据结构为关系型数据库，所以在程序中可以用标准的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。

# 系统出错处理设计

## 出错信息

在软件的开发过程中，需要对软件中存在的漏洞立即进行处理，即使一个很小的漏洞，也会造成整个软件开发系统的垃圾资源无限增多，最终可能会导致系统的瘫痪。因此在系统设计中，对拟开发系统要注意数据的约束性。为了进行数据统一收集和规范数据统一格式方面管理者的处理，针对存在的安全隐患，本项目组进行了统一约束规范

|  |  |
| --- | --- |
| **错误名称** | **解决方法** |
| 用户名密码不正确 | 提示输入不正确，检查信息是否正确，并返回。 |
| 密码不符合规范 | 提示密码必须为6-16字符型 |
| 两次输入密码不正确 | 提示两次的密码必须一致 |
| 用户名没有输入 | 提示用户名不能为空 |
| 身份中没有输入 | 提示请输入身份证号 |
| 未输入姓名 | 提示请输入真实姓名 |
| 验证码不正确 | 提示请输入正确的验证码 |

## 补救措施

所有的系统，不管采用哪种开发方式，都不可避免的存在弱点。所以不但在信息管理系统这一级而且在计算中心这一级(如果适用，也包括远程设备)都要审核并提出设计到安全性的问题。通过识别系统的弱点来减少侵犯安全性的危险，以及采取必要的预防措施来提高安全水平，这是用户和信息服务管理部门可做得到的。

1）后备技术

由于现代计算机科学技术的快速发展，以往的数据备份已经不能满足用户和管理者对现有系统的要求，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，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。

2）降效技术

通常情况下系统的管理员都不是专业的计算机人员，对计算机专业的知识了解相对较少。在面对系统瘫痪等错误时，都不知道如何去解决问题。如今，随着计算机科学技术的飞速发展，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，当然备用的服务器从价格上考虑，可能不如主服务器，也可能会降低工作效率，但是从数据的完整性和安全出发，可以让维修人员有更多时间进行系统的维护。

3）恢复及再启动技术

由于采取的是低端服务器，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。

# 系统维护设计

详情见《系统维护设计》文档