**BBS论坛系统的设计与实现**

**DESIGN AND IMPLEMENTATION OF BBS FORUM SYSTEM**

**专 业： 计算机科学与技术（信息处理）**

**姓 名： 戴勇军**

**指 导 教 师：**

**申请学位级别： 学 士**

**论文提交日期： 2017年 6月 10日**

**学位授予单位： 天津科技大学**

**摘 要**

随着互联网的快速且越来越进一步发展，网络的信息已经涉及了生活的各个层面。在这样的网络大环境下，BBS论坛系统不仅给网络访问者提供了一个交流的平台，同时也提供了一个讨论各种问题的平台，可以让不同地区，不同文化，不同身份的人群在一个开放的平台上进行思想的交流和传递。随着BBS论坛的流行，网上一些大的门户网站也相继建立了自己的论坛，方便用户的交流，加快传递网络信息，同时使得门户网站更受欢迎。在一些公司的内部网络上同样也建立了公司论坛，方便员工讨论问题和交流技术，加快了信息的流通。

BBS论坛系统设计参考了网上比较盛行的网上社区论坛，在原有的框架下，进行设计分析，加入一些自己设计的元素。通过现实的需求分析，创建了一些概念设计模型，添加一些符合论坛的数据，设计出符合系统的数据库解决方案。本系统基本符合了软件工程开发流程，从前期架构到后期开发，完成了系统功能的实现。BBS论坛系统开发使用了现在比较盛行的开发技术ASP.NET和C#语言组合技术，在微软系统上的Microsoft Visual Studio 2015平台上做系统的开发，后台的数据库使用Microsoft SQL Server 2012作为数据库管理工具，系统服务器使用Windows系统上自带的IIS服务器。BBS论坛系统包含了游客权限，一般用户权限和管理员权限。游客只能进行帖子浏览，注册、登录后可以成为一般用户进行帖子的回复，收藏，发贴等行为。管理员在后台可以对用户信息、帖子、板块进行增、删、改等操作。通过BBS论坛系统的运行，实现了用户的交流和信息的传递。

关键词：BBS论坛系统； C#； ASP.NET； SQL server； 三层架构技术

**ABSTRACT**

Due to the in-depth development of the Internet, network information covers all aspects of life. In such a network environment, BBS forum system provides majority of users a platform to exchange information and a plateform to discuss problems, you can let different regions, different cultures, different identities of the crowd in an open platform for ideological communication and transmission. With the popular BBS forum, some large online portal also have established their own forums, user-friendly communication, speed up the transmission of information, but also increased the popularity of the site. In some companies on the internal network also established a company forum to facilitate staff to discuss issues and exchange of technology to speed up the flow of information.

BBS Forum system design reference to the online popular online community forum, in the original framework, the design analysis, adding some of their own design elements. Through the reality of the needs of analysis, to create a number of conceptual design model, add some of the data in line with the forum, designed to meet the system of database solutions. The system is basically match software engineering development process, from the early structure to the late development, the completion of the system function. BBS Forum system development using the more popular development technology ASP.NET and C # language combination technology, Microsoft system on the Microsoft Visual Studio 2015 platform to do the system development, the background of the database using Microsoft SQL Server 2012 as a database management tool, The system server uses the IIS server that comes with the Windows system. The BBS Forum system includes visitor rights, general user rights, and administrator privileges. Visitors can only post browsing, registration, login can become a general user post reply, collection, posting and other acts. Administrators in the background can be user information, posts, plates to increase, delete, change and other operations. Through the BBS forum system to run, to achieve the user's communication and information transmission.

**Key Words:** BBS forum system； C#； ASP.NET； SQL Server； Three-Tier Architecture Technology

**目 录**

[第一章 绪论 1](#_Toc483426082)

[第一节 课题背景 1](#_Toc483426083)

[第二节 开发目的和意义 1](#_Toc483426084)

[第二章 系统开发环境与相关技术 1](#_Toc483426085)

[第一节 系统开发环境 1](#_Toc483426086)

[第二节 相关技术介绍 3](#_Toc483426087)

[第三章 系统分析 6](#_Toc483426088)

[第一节 可行性分析 6](#_Toc483426089)

[第二节 需求分析 6](#_Toc483426090)

[第三节 任务概述 7](#_Toc483426091)

[第四节 系统功能描述 7](#_Toc483426092)

[第四节 用例图及用例描述 9](#_Toc483426093)

[第四章 系统总体设计 21](#_Toc483426094)

[第一节 系统功能模块设计 21](#_Toc483426095)

[第二节 系统数据库设计 22](#_Toc483426096)

[第五章 系统详细设计 23](#_Toc483426097)

[第一节 系统运行平台设置 23](#_Toc483426098)

[第二节 网站主页设计 24](#_Toc483426099)

[第三节 用户管理模块 26](#_Toc483426100)

[第四节 帖子管理模块 32](#_Toc483426101)

[第五节 论坛管理模块 38](#_Toc483426102)

[第六节 后台界面设计 41](#_Toc483426103)

[结 论 42](#_Toc483426104)

[参考文献 43](#_Toc483426105)

[致谢 43](#_Toc483426106)

# 第一章 绪论

## 第一节 课题背景

上个世纪70年代初的美国计算机得以发展，其中出现了BBS这样的社区。BBS刚开始的目的仅仅是为了方便网络用户的交流，方便通讯和留下信息，给加入BBS中的成员之间提供这么一个平台。在那个时候网络还不太发达，网络上没有像现在这样充斥着形形色色的网站，人们获取信息特别的困难，网络等待着人们去开发。BBS刚开始的功能包括获取新闻信息，获取各种交易的信息，发布信息，发布信息并获得回答。现在BBS前三项的功能已经逐步的独立出来，出现了现在的新闻专题网站，电子交易网站，博客类网站。到了上个世纪90年代，BBS发展成为论坛形式的网站，由于网络的快速发展，网络人群增长迅速，BBS论坛受到广大网友的喜爱，网上也出现了一些大受欢迎的论坛网站，BBS论坛也开展了一系列的功能，包括网上在线支持，网上在线服务等。

## 第二节 开发目的和意义

BBS论坛对于社会、商业、学校来说，依旧是个不可或缺的一部分，社会上的人群可以在BBS论坛上获得交流和技术支持，商业网站也建立了自己的论坛，调高论坛的活跃度，增加网站的人气，学校论坛成为学生平时交流，讨论问题的主要场所。

开发BBS论坛系统是为了能够给大家的生活提供更多的方便并且有益于大家之间沟通跟互相之间获取更多的信息。BBS论坛系统采用了Asp.Net相关的知识和SqlServer2012来设计开发。本论坛系统发布在网络上，以网络作为交流平台，前台设计了方便用户登录论坛的服务，同时也可以让任何网络用户进行用户注册。只有登录在论坛上的用户可以进行浏览查看帖子、发表新帖、回复相关帖子内容等操作，还可以查询搜索帖子。总而言之，开发本论坛网站实现了对计算机的软硬件的解决方案进行论证，对于应用领域的相关例证进行了调查与分析，并且BBS论坛系统参考了各种相关的资料，实现了数据库相关编程的实践。BBS论坛系统基本上满足了大家对于网上内容获取的需求，通过BBS论坛系统进行沟通和学习。

# 第二章 系统开发环境与相关技术

## 第一节 系统开发环境

BBS论坛系统的设计是基于.NET的实现，以Visual Studio 2015为开发平台、C# 为开发语言，结合了SQL Server 2012数据库。

一、 开发平台——Visual Studio 2015

Visual Studio（VS）其集成开发环境非常强大且完整，具有强大的可视化用户界面设计功能，简单的操作便可以生成一个漂亮界面。可用于生成ASP.NET Web应用程序、XML WebServices、桌面应用程序和移动应用程序，功能十分强大。在该平台的基础之上，还提供了辅助工具—MSDN，通过查询MSDN，用户可以查到自己想查到的任何东西，包括C#类、属性、方法、语句以及一些常用的例子和功能点。VS除了内置的数据库可以操作之外，它还支持 IBM DB2和Oracle数据库。VS支持C#（以及托管和非托管C++、VB和其他一些语言），把.NET 功能集成到代码中。

二、 开发语言——C#

C# 是建立在.NET Framework上的安全、可靠的程序，C#本身则是一种面向对象的简洁、类型安全的语言，用户在编程或者其他操作的时候可以选择.NET Framework的版本，从过去的1.0，2.0到现在的3.0和4.0，该环境一直在进化。 针对C#开发语言来说，您可以使用 C# 来创建传统的 Windows 客户端应用程序、XML Web services、客户端/服务器应用程序等等。C# 语法简单易学，适合程序员进行开发。 C# 的很多语句让使用过 C、C++ 或 Java 的人都感到熟悉。 了解上述任何一种语言的开发人员都可以在相对较短的时间内高效开始使用C#进行开发，但是真的想完全理解C#还是需要很长时间的开发经验。 C# 对C++进行了良好的封装，让我们的开发人员只要通过调用就能解决很多复杂的问题，并提供了很多强大的功能，例如可为 null 的值类型、枚举、委托和进行内存的直接存取，这些都是 Java 所不具备的。 C# 支持泛型方法和类型，从而提供了更出色的类型安全和性能。语言集成查询 (LINQ) 表达式使强类型查询成为了一流的语言构造[3]。

三、 数据库——SQL Server 2012

Microsoft SQL Server 2012是微软提供的存储大数据量的软件，该软件具有完整的企业级技术与工具且功能严谨，帮助您以最低的成本获得最有价值的信息。用户可以充分享受高性能，高可用性，高安全性以及用更多的高效管理与开发工具，利用自服务的商业智能实现更为广泛深入的商业洞察[4]。

四、 三层架构设计思想

三层架构，顾名思义是通过三层类库来实现，具体内容为用户界面层（UI），逻辑判断层（BLL）以及数据连接层（DAL），各层各司其职，互不干扰。

所谓三层体系结构，是加入了一个夹在客户端与数据库之间的“中间层”-逻辑访问层，也叫组件层。这里所说的三层，不是单纯的指物理上的放置三台机器，也不仅仅有B/S应用才是三层体系结构，三层是指一台机器上的逻辑上的三层。

三层体系的应用程序将程序的思路逻辑以及合法性校验等工作放到了中间层处理。通常情况下，客户端与数据库不能直接进行交流，而是通过逻辑访问层建立通话，再经由逻辑访问层与数据库进行交互。各层之间独立运行，互不干扰。其中，数据访问层：主要是对数据进行交互，通过与数据库相连接达到对数据库增删改查的目的。业务逻辑层：主要是针对出现的具体问题的逻辑操作，也可以理解成对数据层的操作。表示层：主要表示WEB方式，表现形式为用户界面即.aspx。也可以表示成WINFORM方式。如果逻辑层相当强大和完善，那么无论用户界面层怎么修改或者做其他操作，逻辑层都能很好地提供服务。

## 第二节 相关技术介绍

一、 NET技术

1.1发展背景：

在2000年第二季时，微软正式推动.NET策略，ASP+也顺理成章的改名为ASP.NET，经过四年的开发，第一个版本的ASP.NET在2002年1月5日亮相(和.NET Framework 1.0)，Scott Guthrie也成为ASP.NET的产品经理（到现在已经开发了数个微软产品，像ASP.NET AJAX和Microsoft Silverlight）。目前最新版本的 ASP.NET 4.0 以及 .NET Framework 4.0 仍在开发中。

1.2 ASP.NET开发工具：

ASP.net构架是可以用Microsoft（R）公司最新的产品 [Visual Studio](http://baike.baidu.com/view/28727.htm).net开发环境进行开发，WYSIWYG（What You See Is What You Get所见即为所得）的编辑。这些仅是ASP.net强大化软件支持的一小部分。

ASP.NET的特点——强大性和适应性。因为ASP.net是基于通用语言的编译运行的程序，所以它的强大性和适应性，可以使它运行在Web应用软件开发者的几乎全部的平台上（笔者到现在为止只知道它只能用在Windows 2000/2003 Server/VISTA/7上）。通用语言的基本库，消息机制，数据接口的处理都能无缝的整合到ASP.net的Web应用中。ASP.net同时也是language-independent语言独立化的，所以，你可以选择一种最适合你的语言来编写你的程序，或者把你的程序用很多种语言来写，现在已经支持的有C#（C++和Java的结合体），VB，Jscript , C++、F++。将来，这样的多种程序语言协同工作的能力保护您现在的基于COM+开发的程序，能够完整的移植向ASP.net。ASP.NET一般分为两种开发语言，VB.NET和C#，C#相对比较常用，因为是.NET独有的语言，VB.NET则为以前VB程序设计，适合于以前VB程序员，如果新接触.NET，没有其他开发语言经验，建议直接学习C#即可。

1.3 ASP.NET处理架构

ASP.NET 运行的架构分为几个阶段： 在 IIS 与 Web 服务器中的消息流动阶段。 在 ASP.NET 网页中的消息分派。 在 ASP.NET 网页中的消息处理。

1.4 ASP.NET 内部的 HTTP 处理器有：

ISAPIRuntime：由 aspnet\_isapi.dll 调用，初始化 HttpWorkerRequest 对象（会由IIS的版本决定要初始化的版本）。 HttpRuntime：提供要求队列 (Request Queue)、调用 HttpWorkerRequest 中的 ProcessRequest() 方法，以及后续的处理工作。 HttpWorkerRequest：产生 HttpApplication、HttpRequest、HttpResponse 等基础对象的 HTTP 要求对象，并将要求转送到要处理的对象（并调用它的 ProcessRequest() 方法）。 IHttpHandler 与 IHttpAsyncHandler：负责处理 HTTP 要求的单元，由 ProcessRequest() 来分派与运行要求。 ASP.NET网页中的事件程序 当 HttpWorkerRequest 调用 ASP.NET 网页 (System.Web.UI 命名空间的 Page 类) 的 Page.ProcessRequest() 方法时，它会依序的引发 Page 内的各个事件，并同时调用在 Page 中所有控件的相关事件，其引发顺序为：

PreInit 事件：运行预先初始化的工作，在ASP.NET 2.0中，若要动态调整主版页面 (Master Page)、布景主题 (Theme) 时，要在这个事件中调整。 Init 事件：运行初始化工作。 InitCompleted 事件：在完成初始化工作后引发。 Preload 事件：运行预先加载的工作。 Load 事件：运行加载的工作，大多数的网页都拥有 Page\_Load 事件处理程序，用户控件 (user control) 中也有 Page\_Load 事件例程，都会在此时调用。 控件的 PostBack 变更通知：当网页侦测到是 PostBack 要求时，会引发 PostBack 消息通知的事件。 控件的 PostBack 相关事件：当网页侦测到是 PostBack 要求时，会引发 PostBack 消息指定的控件的事件。 LoadCompleted 事件：运行加载完成后的工作。 PreRender 事件：处理在产生 HTML 结果前的工作。 SaveStateCompleted 事件：处理页面状态 (ViewState 与 ControlState) 储存完成后的事件。 Render 事件：处理产生 HTML 的工作。 Unload 事件：处理退出网页处理时的工作。 如果 HttpWorkerRequest 调用的是实现 IHttpHandler 界面的 HTTP 处理程序 时，它只会调用 IHttpHandler.ProcessRequest() 方法，由它来处理程序的输出，不像 Page.ProcessRequest() 会处理事件顺序，因此 HTTP Handler 很适合轻量级的数据处理，像是输出文件数据流或是图片数据流等。

1.5 事件模型

ASP.NET 的原始设计构想，就是要让开发人员能够像 VB 开发工具那样，可以使用事件驱动式程序开发模式 (Event-Driven Programming Model) 的方法来开发网页与应用程序，若要以ASP技术来做到这件事的话，用必须要使用大量的辅助信息，像是查询字符串或是窗体字段数据来识别与判断对象的来源、事件流向以及调用的函数等等，需要撰写的代码量相当的多，但 ASP.NET 很巧妙利用窗体字段和JavaScript脚本把事件的传递模型隐藏起来了。

ASP.NET 的事件模型是由以及数个 Hidden Field 组合而成，基于 HTTP 模型的限制，所有的网页程序在运行结果输出到用户端后，程序就会退出运行，为了维护在 ASP.NET 网页与控件的状态数据，因此在输出 ASP.NET 控件时，ASP.NET 会将部份状态数据储存到网页的 Hidden Field 中，这类型的状态数据称为 ViewState（ID 为 \_\_VIEWSTATE），在服务器端即会被解译出状态与事件数据。在大多数的内置 Web 控件中都有使用到这个机制，因此在使用大量 ASP.NET Web 控件的网页中，会有许多的 ViewState 会存放在网页中并随着 HTTP 数据流输出到用户端，ViewState 在输出时，会被加密为一组乱码字符串，其金钥值定义在计算机中，并且每一个对象都会被序列化 (serialize) 成字符串（因此若是自定义对象要放到 ViewState 时，则应要让它支持序列化），再输出到 \_\_VIEWSTATE 字段中，在每次的网页来回时都会被传输，较大的 ViewState 会让网页大小膨胀，不利于快速的网络传输，不过 ASP.NET 本身有提供将 ViewState 关闭的功能，因此如果控件不需要状态保存时，可将它关闭以减少输出的大小。

为确保控件的事件能够确实被引发，让事件驱动能够被运行，因此控件事件引发命令时需要的参数，是交由 JavaScript 脚本在用户端引发时，填入另一个 Hidden Field（ID 为 \_\_EVENTTARGET 以及 \_\_EVENTARGUMENT），并且引发窗体的送出指示 (submit)，传送到服务端后，服务端的 HttpApplication 中的工具函数会解析 \_\_EVENTTARGET 和 \_\_EVENTARGUMENT 字段中的信息，并且交由控件所实现的 RaisePostBackEvent() 来引发事件，并由 .NET Framework 内部的事件处理器制接手处理（调用控件设置的事件处理程序）。

状态管理状态管理 (state management) 在Web应用程序中，一向是很重要的课题，良好的状态管理可以帮助开发人员发展出具有状态持续能力的应用程序（像是工作流程型应用程序或是电子商务应用程序），但状态管理功能会视应用程序的部署状态以及信息的共用程度来选择，在 ASP.NET 中，分为服务器端状态管理以及用户端状态管理，用户端状态管理为ViewState以及Cookies，服务端状态管理则是Session与Application对象。它们的差异点在于：

ViewState 是加密的数据流，和 HTML 一起输出到用户端。 Cookies 是加密（也可不加密）的小型数据，和 HTML 不同，它可以高速缓存在用户端浏览器中。 Session 是服务器端的状态保存机制，每个用户端均有独立的空间（以浏览器运行个体来赋与唯一的SessionID值）。 Application 是服务器端的状态保存机制，但应用程序所有的用户端共用同一份状态数据。

1.6 应用程序层级对象

Application 对象会在应用程序的 Application\_OnStart 事件中初始化，并使用名称来识别数据（它是一个 NameObjectCollectionBase 集合的实现品），它会储存在应用程序的范围内，所有的连接（用户）都可以使用，属于共用型的储存体，适合储存所有用户都可使用的数据，在多人使用的情况下，可以适当的使用 Lock/Unlock 的机制来确保应用程序状态的更新。

Application.Lock();Application["PageRequestCount"]=((int)Application["PageRequestCount"])+1;Application.UnLock();

二、 SQL Server技术

2.1基本信息

SQL Server 是一个关系数据库管理系统。它最初是由Microsoft、 Sybase 和Ashton-Tate三家公司共同开发的，于1988 年推出了第一个OS/2 版本。在Windows NT 推出后，Microsoft与Sybase 在SQL Server 的开发上就分道扬镳了，Microsoft 将SQL Server 移植到Windows NT系统上，专注于开发推广SQL Server 的Windows NT 版本。Sybase 则较专注于SQL Server在UNIX 操作系统上的应用。本系统主要采用SQL Server 2005。

SQL Server 2005 是一个全面的数据库平台，使用集成的商业智能 (BI) 工具提供了企业级的数据管理。SQL Server 2005 数据库引擎为关系型数据和结构化数据提供了更安全可靠的存储功能，使您可以构建和管理用于业务的高可用和高性能的数据应用程序。

SQL Server 2005 数据引擎是本企业数据管理解决方案的核心。此外 SQL Server 2005 结合了分析、报表、集成和通知功能。这使您的企业可以构建和部署经济有效的 BI 解决方案，帮助您的团队通过记分卡、Dashboard、Web services 和移动设备将数据应用推向业务的各个领域。

与 Microsoft Visual Studio、Microsoft Office System 以及新的开发工具包（包括 Business Intelligence Development Studio）的紧密集成使 SQL Server 2005 与众不同。无论您是开发人员、数据库管理员、信息工作者还是决策者，SQL Server 2005 都可以为您提供创新的解决方案，帮助您从数据中更多地获益。

# 第三章 系统分析

在开始开发系统之前需要对系统进行分析，完成系统设计前期工作。在这个阶段主要是定义一个系统目标是什么，可行性分析报告，系统能够实现哪些功能，系统预期的成本等。下面对系统可行性，系统需求，系统功能描述等部分进行进一步的说明。

## 第一节 可行性分析

可行性研究的意义在于能否完成一个项目的设计和功能实现，能够花费越小的成本，能够在最短的时间里实现。为了确定本系统开发具有可行性，主要进行了四个方面的分析：

一、经济可行性

在开发一个系统过程中，需要确定一个项目中的开发人员数量，使用的机器，办公地点等资源的安排，以及系统开发周期的计划。本系统对于开发者来说没有任何的限制，只需要一部电脑即可，可以在任何地点，进行系统的开发。本系统开发周期不长，一至两个月即可完成开发工作。进过分析，本系统在经济可行性上是通过的。

二、技术可行性

技术的可行性要考虑到本次开发技术是否过于老旧，不适用于现在大部分机器，分别从软件环境和硬件环境对本次开发进行分析。

本次系统开发使用的硬件，一般电脑即可满足要求，越高的配置能够提高系统的运行速度，当然是好的。软件方面，BBS论坛系统使用ASP.NET技术+C#语言对系统功能进行编写，软件开发平台使用MicrosoftVisualStudio2015，后台处理数据库使用SQLServer2012作为论坛数据库管理软件。经过分析，本系统在技术可行性上是通过的。

三、时机可行性

时机可行性要考虑本次开发的系统是否能够使广大用户所接受。从目前的网络上分析，网上论坛已经成为大部分网络访问者的驻足处，通过论坛发表看法，浏览和回复帖子已经成为一种趋势。而且越来越多基于web的应用出现网上，因此，经过分析，本系统在时机可行性上是通过的。

四、管理可行性

管理可信性要考虑到本次开发系统是否有管理人员进行管理系统，系统管理功能是否科学，是否能够提供完善的管理方式。本次开发的BBS论坛系统提供了后台管理员，管理员通过后台管理操作系统，所以系统管理功能是科学的，可行的，完善的。

通过上面的可行性分析，系统不管是在经济上，技术上，时机上，还是管理上都是可行的，所以本系统具备了开发的可行性。

## 第二节 需求分析

BBS论坛系统主要作用是方便网上用户进入论坛进行交流和讨论，可以是生活，或工作，或技术方面的，但是整个系统需要对论坛的帖子浏览和发表帖子进行限制，只有这样才能更好的管理论坛，论坛的管理功能显得极其重要。BBS论坛系统从用户的角度，将论坛访问者划分为不同权限，从系统的角度，可以分为多个管理模块。本系统在分析和设计下具有如下几个特点：

一、系统具有极好的交互性，友好的界面

二、丰富的功能，简单的操作，极具扩张性，极高的稳定性

三、系统能够进行不同平台的移植，没有系统的限制

四、论坛的可维护性高，方便操作

五、论坛运行环境稳定，用户信息安全。

## 第三节 任务概述

前面对BBS论坛系统进行了需求分析，BBS论坛系统将论坛访问者视为三种，分别是游客，一般权限用户和管理员。游客访问BBS论坛系统能够进入到帖子详细页面查看帖子信息，选择用户，查看用户信息。任何游客都可以通过注册成为BBS论坛用户，用户登录后可以选择论坛上的帖子进行查看帖子详细信息，回复帖子，发表新帖等操作。管理员登录后台管理界面可以对后台数据库进行操作，可增加、删除、修改论坛用户，对帖子进行审阅，修改、删除帖子，可以增加、修改、删除板块，可以任命用户为版主等操作。

## 第四节 系统功能描述

为了方便BBS论坛系统功能部分的设计与实现，从前期的可行性分析到系统需求分析，了解功能需求，从系统用户角度对系统功能进行如下的详细分析：

一、游客用户功能描述

游客进入BBS论坛后，可以注册成为论坛会员，未注册的游客可以查看页面上各个帖子，选择帖子进入到详细页面查看帖子内容，也可进入各个板块查看浏览帖子。游客可以查询贴子。点击论坛用户后，游客可以查看论坛其他用户成员信息。



图3-1 游客用户功能模块

二、一般权限用户功能描述

一般权限用户进入BBS论坛系统，登录论坛，搜索查询帖子信息，浏览帖子详细信息，进入各个板块发表帖子，查看帖子详细页面，回复帖子。进入留言页面，浏览留言信息，发表看法。查看个人详细信息，添加好友，查看好友留言，删除留言。

一般权限用户被任命为版主后可对板块内帖子进行管理，可以增加、删除、修改帖子内容。



图3-2 一般权限用户功能模块

三、管理员功能模块

管理员可以直接进入后台管理界面，或者登录BBS论坛系统，通过主页登录点击个人资料进入后台管理界面。管理员除了能够进行一般权限用户的功能外，还能够进行用户管理，增加、删除修改用户；板块管理，添加、删除修改板块；帖子管理，增加、修改删除帖子；回复管理，增加、修改删除回复的操作。



图3-3 管理员功能模块

本次利用了层次图表示的方法来说明各个模块在系统中的相对应之关系。层次图使用树形结构的方式构建系统功能，每一层的方框图都体现了系统中各个功能模块，层次机构的上下层体现了系统功能的上下级关系。树形结构的顶层是一个单独的矩形框，它代表完整的数据结构，下面各层矩形框代表各个数据的子集，最底层的各个矩形框代表组成这个数据的实际数据元素（不能再分的元素）。

系统功能结构图见图3-4。



图3-4 系统功能结构图

## 第四节 用例图及用例描述

用例图是从软件需求分析到最终实现的第一步。用例图通过用户角度描述，形象反映了系统的主要功能，用例图反映了相关用户与用户对系统的期待，在进行需求分析时，开发人员与用户通过多次沟通协调商定共同完成，这样能更好的描述系统的所应该具有的基本功能以及用户需要为系统提供的服务，这样方便了系统用户更容易的理解系统元素的作用，以方便软件开发人员实现这些系统元素。通过与用户共同商定完成，能够将需求不明确的元素造成的项目失败的几率降到了最小，提高了整个系统项目的成功率。用例图主要由系统、用例、参与者和关系构成的。BBS论坛系统的各用户用例图如下图所示。

1：游客用例

浏览帖子

游客

查看用户信息

搜索查询帖子

注册

查看留言

图3-4 游客用户用例图

游客用例描述：

1.1注册

|  |
| --- |
| 用例名称：注册 |
| 用例标识号：1.1 |
| 参与者：游客 |
| 简要说明：  游客注册成为BBS论坛会员 |
| 前置条件：  游客访问BBS论坛系统 |
| 基本事件流：   1. 游客登录BBS论坛系统，点击“注册”按钮 2. 系统出现用户注册界面，游客填写信息 3. 填写完毕后，确认无误，点击注册 4. 用例终止 |
| 其他事件流：  1.游客注册用户名，点击“检察用户名”，可查询用户名是否被注册使用  2.填写用户信息页面，用户点击“重置”按钮，可重置信息 |
| 异常事件流：   1. 注册不成功，提示错误，游客确认 2. 返回注册界面，游客重新填写相关信息 |
| 后置条件：  游客注册成功 |
| 注视：无 |

1.2浏览帖子

|  |
| --- |
| 用例名称：浏览帖子 |
| 用例标识号：1.2 |
| 参与者：游客 |
| 简要说明：  游客可以浏览BBS论坛系统各个板块帖子 |
| 前置条件：  游客访问BBS论坛系统 |
| 基本事件流：   1. 游客登录BBS论坛系统，选择板块 2. 系统跳转到板块页面，游客点击帖子标题 3. 系统跳转到帖子详细信息页面 4. 游客浏览帖子信息 5. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，游客确认 2. 返回BBS论坛系统页面 |
| 后置条件：无 |
| 注视：无 |

1.3搜索查询帖子

|  |
| --- |
| 用例名称：搜索查询帖子 |
| 用例标识号：1.3 |
| 参与者：游客 |
| 简要说明：  游客可以在BBS论坛系统中搜索帖子，查找相关帖子 |
| 前置条件：  游客访问BBS论坛系统 |
| 基本事件流：   1. 游客登录BBS论坛系统，点击“帖子搜索” 2. 系统跳转到搜索帖子界面，游客输入帖子关键字，点击“搜索”按钮 3. 页面出现帖子信息 4. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，游客确认 2. 返回BBS论坛系统页面 |
| 后置条件：无 |
| 注视：无 |

1.4查看用户信息

|  |
| --- |
| 用例名称：查看用户信息 |
| 用例标识号：1.4 |
| 参与者：游客 |
| 简要说明：  游客访问BBS论坛系统，点击用户，查看用户信息 |
| 前置条件：  游客访问BBS论坛系统 |
| 基本事件流：   1. 游客进入BBS论坛系统，点击用户名 2. 系统跳转到用户信息界面 3. 游客浏览用户信息 4. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，游客确认 2. 返回BBS论坛系统页面 |
| 后置条件：无 |
| 注视：无 |

2.一般权限用户用例

浏览帖子

查看用户信息

搜索查询帖子

发表帖子

回复帖子

查看留言

一般权限用户

版主

修改帖子

删除帖子

发表留言

修改用户信息

注册

登录

添加好友

图3-5 一般权限用户用例图

一般权限用户用例描述：

2.1登录

|  |
| --- |
| 用例名称：登录 |
| 用例标识号：2.1 |
| 参与者：一般权限用户，版主 |
| 简要说明：  用户访问BBS论坛系统，登录论坛享受更多功能 |
| 前置条件：  用户访问BBS论坛系统 |
| 基本事件流：   1. 用户进入BBS论坛系统，点击“登录”按钮，发出登录请求 2. 系统跳转到用户登录界面 3. 用户填写用户信息，点击“登录”按钮 4. 系统检验用户登录信息 5. 检验成功，跳转到BBS论坛系统主页；检验失败，系统提示登录失败 6. 用例终止 |
| 其他事件流： |
| 异常事件流：   1. 提示错误信息，用户确认 2. 返回BBS论坛系统页面 |
| 后置条件：用户登录成功 |
| 注视：无 |

2.2发表帖子

|  |
| --- |
| 用例名称：发表帖子 |
| 用例标识号：2.2 |
| 参与者：一般权限用户，版主 |
| 简要说明：  用户访问BBS论坛系统，发表帖子 |
| 前置条件：  用户已登录BBS论坛系统 |
| 基本事件流：   1. 用户进入BBS论坛系统，点击“发新帖”图标 2. 系统跳转到用户发贴页面，包括主题，分类和内容 3. 用户填写帖子内容，点击“提交”按钮，完成帖子发布 4. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，用户确认 2. 返回BBS论坛系统页面 |
| 后置条件：用户帖子发表成功 |
| 注视：无 |

2.3回复帖子

|  |
| --- |
| 用例名称：回复帖子 |
| 用例标识号：2.3 |
| 参与者：一般权限用户，版主 |
| 简要说明：  用户访问BBS论坛系统，回复板块帖子 |
| 前置条件：  用户已登录BBS论坛系统 |
| 基本事件流：   1. 用户进入BBS论坛系统，点击帖子主题名 2. 系统跳转到帖子详细信息页面，页面包括帖子内容，其他用户回复内容，文本框，提交按钮，取消按钮 3. 用户填写编辑回复内容，点击“提交”按钮，完成帖子回复 4. 用例终止 |
| 其他事件流：用户编辑回复内容，点击“取消”按钮，取消回复，回到帖子详细页面 |
| 异常事件流：   1. 提示错误信息，用户确认 2. 返回BBS论坛系统页面 |
| 后置条件：用户帖子发表成功 |
| 注视：无 |

2.4查看留言

|  |
| --- |
| 用例名称：查看留言 |
| 用例标识号：2.4 |
| 参与者：一般权限用户，版主 |
| 简要说明：  用户访问BBS论坛系统，查看BBS论坛系统留言 |
| 前置条件：  用户已登录BBS论坛系统 |
| 基本事件流：   1. 用户进入BBS论坛系统，点击“留言”按钮 2. 系统跳转到系统留言页面，页面包括留言信息内容，编辑留言文本框，留言按钮 3. 用户查看留言内容 4. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，用户确认 2. 返回BBS论坛系统页面 |
| 后置条件：无 |
| 注视：无 |

2.5发表留言

|  |
| --- |
| 用例名称：发表留言 |
| 用例标识号：2.5 |
| 参与者：一般权限用户，版主 |
| 简要说明：  用户访问BBS论坛系统，发表留言 |
| 前置条件：  用户已登录BBS论坛系统 |
| 基本事件流：   1. 用户进入BBS论坛系统，点击“留言”按钮 2. 系统跳转到系统留言页面，页面包括留言信息内容，编辑留言文本框，留言按钮 3. 用户编辑留言内容，点击“留言”，完成留言发表 4. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，用户确认 2. 返回BBS论坛系统页面 |
| 后置条件：留言发表成功 |
| 注视：无 |

3.管理员用户用例

管理员

用户登录

用户注册

查看留言

发表留言

版主

帖子管理

回复管理

用户管理

板块管理

浏览帖子

搜索查询帖子

发表帖子

修改帖子

删除帖子

回复帖子

修改回复

删除用户

修改用户信息

添加好友

删除板块

增加板块

删除回复

查看用户信息

修改板块

图3-6 管理员用户用例图

管理员用户用例描述：

3.1修改帖子

|  |
| --- |
| 用例名称：修改帖子 |
| 用例标识号：3.1 |
| 参与者：管理员，版主 |
| 简要说明：  管理员登录BBS论坛系统，修改帖子内容，该帖子内容最终更新在帖子列表上 |
| 前置条件：  用户已登录BBS论坛系统或进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击发贴管理下“修改删除帖子” 2. 系统跳转到帖子编辑页面，选择要修改的那一项帖子，点击编辑 3. 管理员修改编辑帖子内容，点击更新，完成帖子修改 4. 用例终止 |
| 其他事件流：进入帖子编辑页面，点击编辑，完成帖子编辑，点击取消，取消帖子修改 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：帖子修改成功 |
| 注视：无 |

3.2删除帖子

|  |
| --- |
| 用例名称：删除帖子 |
| 用例标识号：3.2 |
| 参与者：管理员，版主 |
| 简要说明：  管理员登录BBS论坛系统，删除帖子 |
| 前置条件：  用户已登录BBS论坛系统或进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击发贴管理下“修改删除帖子” 2. 系统跳转到帖子编辑页面，选择要删除的那一项帖子，点击删除 3. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：帖子删除成功 |
| 注视：无 |

3.3修改回复

|  |
| --- |
| 用例名称：修改回复 |
| 用例标识号：3.3 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，修改回复内容，最终回复信息更新在回复列表中 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击回复管理下“修改删除回复” 2. 系统跳转到回复编辑页面，选择要修改的那一项回复，点击编辑 3. 管理员修改编辑回复内容，点击更新，完成回复修改 4. 用例终止 |
| 其他事件流：进入回复编辑页面，点击编辑，完成回复编辑，点击取消，取消回复修改 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：回复修改成功 |
| 注视：无 |

3.4删除回复

|  |
| --- |
| 用例名称：删除回复 |
| 用例标识号：3.4 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，删除回复 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击回复管理下“修改删除回复” 2. 系统跳转到回复编辑页面，选择要删除的那一项回复，点击删除 3. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：回复删除成功 |
| 注视：无 |

3.5修改用户信息

|  |
| --- |
| 用例名称：修改用户信息 |
| 用例标识号：3.5 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，修改用户信息，用户信息最终更新在用户列表上 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击用户管理下“修改删除用户” 2. 系统跳转到用户编辑页面，选择要修改的那一项用户，点击编辑 3. 管理员修改编辑帖子内容，点击更新，完成用户信息修改 4. 用例终止 |
| 其他事件流：进入用户信息编辑页面，点击编辑，完成用户信息编辑，点击取消，取消用户信息修改 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：用户信息修改成功 |
| 注视：无 |

3.6删除用户

|  |
| --- |
| 用例名称：删除用户 |
| 用例标识号：3.6 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，删除用户 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击回复管理下“修改删除回复” 2. 系统跳转到回复编辑页面，选择要删除的那一项用户，点击删除 3. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：用户删除成功 |
| 注视：无 |

3.7增加板块

|  |
| --- |
| 用例名称：增加板块 |
| 用例标识号：3.7 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，增加板块，板块内容更新在板块列表中 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击板块管理下“增加板块” 2. 系统跳转到板块编辑页面，点击添加 3. 管理员修改编辑板块内容，点击插入，完成板块添加 4. 用例终止 |
| 其他事件流：进入板块信息编辑页面，点击增加，完成板块信息编辑，点击取消，取消板块增加 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：板块增加成功 |
| 注视：无 |

3.8修改板块

|  |
| --- |
| 用例名称：修改板块 |
| 用例标识号：3.8 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，修改板块内容，板块内容更新在板块列表中 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击板块管理下“修改删除板块” 2. 系统跳转到板块编辑页面，点击修改 3. 管理员修改编辑板块内容，点击修改，完成板块内容修改 4. 用例终止 |
| 其他事件流：进入板块信息编辑页面，点击修改，完成板块信息编辑，点击取消，取消板块修改 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：板块信息修改成功 |
| 注视：无 |

3.9删除板块

|  |
| --- |
| 用例名称：删除板块 |
| 用例标识号：3.9 |
| 参与者：管理员 |
| 简要说明：  管理员登录BBS论坛系统，删除板块，板块内容更新在板块列表中 |
| 前置条件：  管理员已经进入后台管理界面 |
| 基本事件流：   1. 用户进入BBS论坛系统后台管理界面，点击板块管理下“修改删除板块” 2. 系统跳转到板块编辑页面，选择需要删除的板块，点击删除 3. 用例终止 |
| 其他事件流：无 |
| 异常事件流：   1. 提示错误信息，管理员确认 2. 返回BBS论坛系统页面 |
| 后置条件：板块删除成功 |
| 注视：无 |

# 第四章 系统总体设计

## 第一节 系统功能模块设计

BBS论坛系统主要包括三个大的功能模块：用户管理模块，帖子管理模块，论坛管理模块。下面对三个功能模块进行说明：

1. 用户管理模块

（一）用户注册 游客进入到BBS论坛系统注册页面，编辑填写用户信息，注册成为论坛会员，可以享受论坛会员权利。

（二）用户登录 用户进入到BBS论坛系统登录页面，输入用户名和密码等待系统验证，验证成功，进入到BBS论坛系统主页面。验证错误，系统提示用户名或密码不正确，用户确认，返回登录界面。

（三）用户信息管理

1.查看用户信息 BBS论坛系统访问者可以点击论坛会员名称，进入用户信息页面，查看该用户信息。

2.修改用户信息 论坛注册用户登录BBS论坛系统，点击进入个人信息页面，编辑修改个人信息，提交修改后的内容，后台更新个人信息。

3.删除用户 BBS论坛系统管理员登录后台管理界面，选择管理菜单下用户管理，修改删除用户，点击删除用户后，系统更新用户列表。

4.添加好友 用户进入BBS论坛系统，点击论坛会员名称，进入用户信息界面，点击添加好友，加入个人信息中，进入个人信息页面，查看好友列表中好友。

1. 帖子管理模块
2. 浏览帖子详细信息 BBS论坛系统访问者浏览系统主页，浏览最新帖子，进入BBS论坛板块中查看各个板块中的帖子，选择帖子，进入帖子详细信息页面，浏览帖子详细信息。
3. 发表帖子 BBS论坛系统会员用户登录论坛，进入发表新帖页面，编辑帖子内容，包括主题，内容，板块。提交帖子，系统审核后更新到帖子列表中。
4. 回复帖子 BBS论坛系统会员用户登录论坛，进入帖子详细信息页面，浏览帖子，在帖子最底部，编辑回复信息，提交回复内容，系统更新回复内容。
5. 查询贴子 BBS论坛系统访问者点击帖子搜索，进入帖子搜索页面，输入帖子关键词，点击搜索，系统进行帖子查询，反馈帖子信息到搜索页面，访问者可以帖子浏览。
6. 修改帖子 系统管理员登录BBS论坛系统后台管理页面，选择管理菜单里的发帖管理，点击修改删除帖子，进入帖子编辑页面，编辑帖子内容，提交帖子内容，后台更新更新帖子信息，反馈到系统页面。
7. 删除帖子 系统管理员登录BBS论坛系统后台管理页面，选择管理菜单里的发帖管理，点击修改删除帖子，进入帖子编辑页面，选择一项或几项进行删除，后台更新帖子内容，反馈到系统页面。
8. 论坛管理模块
9. 板块管理
10. 增加板块 系统管理员登录BBS论坛系统后台管理页面，选择管理菜单里的板块管理，点击增加板块，点击添加进入板块内容编辑页面，编辑板块内容，提交板块信息，后台更新板块列表，反馈到系统页面。
11. 修改板块 BBS论坛系统管理员登录后台管理页面，选择管理菜单里的板块管理，点击修改删除板块，选择板进行修改，修改后提交系统，系统更新板块内容，反馈到系统页面上。
12. 删除板块 BBS论坛管理员登录后台管理页面，选择管理菜单里的板块管理，点击修改删除板块，选择一项或几项进行删除，系统更新板块内容，反馈到系统页面上。
13. 留言管理
14. 查看留言 BBS论坛系统访问者进入留言页面，查看用户留言
15. 发表留言 BBS论坛系统用户登录论坛，进入留言页面，编辑留言内容，提交留言，系统更新留言信息，反馈到留言页面。

## 第二节 系统数据库设计

BBS论坛系统数据库为BBS\_DBA，共包含下列表：板块内容表（dbo.BBS\_Class），回复内容表（dbo.BBS\_Reply），帖子内容表（dbo.BBS\_Topic），用户信息表（dbo.BBS\_User），留言信息表（dbo.Message）。以下对各个表结构进行说明：

1. 板块内容表（dbo.BBS\_Class）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 是否为主键 | 备注 |
| 1 | ClassID | int | 是 | 不可以为Null，不是唯一标识，记录板块号 |
| 2 | ClassName | nchar | 否 | 可以为Null，记录板块名称 |
| 3 | TopicCount | int | 否 | 可以为Null，记录帖子发表数 |
| 4 | ClassUrl | nvarchar | 否 | 可以为Null，记录板块图片地址 |

表4-1 板块内容表

1. 回复内容表（dbo.BBS\_Reply）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 是否为主键 | 备注 |
| 1 | ReplyID | int | 是 | 不可以为Null，是唯一标识，记录帖子回复号 |
| 2 | TopicID | int | 否 | 可以为Null，记录帖子号 |
| 3 | ReContentery | nvarchar | 否 | 可以为Null，记录回复内容 |
| 4 | UserID | int | 否 | 副键，可以为Null，记录用户ID号 |
| 5 | CreatedOn | datetime | 否 | 可以为Null，记录回复创建时间（随系统时间） |
| 6 | ReplyCount | int | 否 | 可以为Null，记录帖子回复个数 |

1. 表4-2 回复内容表
2. 帖子内容表（dbo.BBS\_Topic）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 是否为主键 | 备注 |
| 1 | TopicID | int | 是 | 不可以为Null，是唯一标识，记录帖子号 |
| 2 | ClassID | int | 否 | 副键，可以为Null，记录板块号 |
| 3 | Title | nvarchar | 否 | 可以为Null，记录帖子标题内容 |
| 4 | UserID | int | 否 | 副键，可以为Null，记录用户ID号 |
| 5 | Contentery | nvarchar | 否 | 可以为Null，记录帖子内容 |
| 6 | ReplyCount | int | 否 | 可以为Null，记录帖子回复个数 |
| 7 | IsClose | char | 否 | 可以为Null，记录帖子当前状态 |
| 8 | CreatedOn | datetime | 否 | 可以为Null，记录帖子创建时间（随系统时间） |
| 9 | RepliedOn | datetime | 否 | 可以为Null，记录帖子回复时间（随系统时间） |

表4-3 帖子内容表

1. 用户信息表（dbo.BBS\_User）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 是否为主键 | 备注 |
| 1 | UserID | int | 是 | 不可以为Null，是唯一标识，记录用户ID号 |
| 2 | UserName | nvarchar | 否 | 可以为Null，记录用户名称 |
| 3 | NickName | nvarchar | 否 | 可以为Null，记录用户身份，昵称 |
| 4 | Password | nvarchar | 否 | 可以为Null，记录用户密码 |
| 5 | Email | nvarchar | 否 | 可以为Null，记录用户邮箱地址 |
| 6 | HeadUrl | nvarchar | 否 | 可以为Null，记录用户头像 |
| 7 | TopicCount | int | 否 | 可以为Null，记录发表帖子数 |
| 8 | ReplyCount | int | 否 | 可以为Null，记录回复帖子数 |
| 9 | Enabled | bit | 否 | 可以为Null，记录用户状态 |
| 10 | CreatedOn | datetime | 否 | 可以为Null，记录用户创建时间（随系统时间） |
| 11 | Sex | nvarchar | 否 | 可以为Null，记录用户性别 |

表4-4 用户信息表

1. 留言信息表（dbo.Message）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 是否为主键 | 备注 |
| 1 | MessageID | int | 是 | 不可以为Null，是唯一标识，记录留言号 |
| 2 | MessageContent | nvarchar | 否 | 可以为Null，记录留言信息 |
| 3 | UserID | int | 否 | 可以为Null，记录用户ID号 |
| 4 | CreateOn | datetime | 否 | 可以为Null，记录留言创建时间（随系统时间） |

表4-5 留言信息表

# 第五章 系统详细设计

## 第一节 系统运行平台设置

系统运行平台对于开发BBS论坛系统是至关重要的，稳定的系统软件环境和硬件环境，能够给系统开发提供一个足够的支撑，保证系统的正常开发和后续维护，减少系统成本的支出，提高系统的运行效率。随着技术的不断发展，选择合适的软件环境和硬件环境成为开发系统的稳定性的首要因素。

1. 系统硬件环境设置

计算机的基本硬件需满足这样的条件：拥有Pentium III800或以后发版的系统处理器，内存满足2G或以上，机器硬盘固定空间至少在20G或以上。

如果需要更快的处理速度，可以配置更高的硬件配件，但需要考虑硬件的兼容性，不能乱加硬件，防止系统经常性崩溃。

1. 系统软件环境设置

BBS论坛系统选择在Windows7操作系统上进行编程，可选择Windows XP或以上系统进行编程。系统在Microsoft Visual Studio 2015 集成开发环境上进行开发，采用的asp.net技术，选择C#语言编写内部功能，系统数据库使用Microsoft SQL Sever 2005数据库服务系统。系统使用Web服务器：Internet 信息服务（IIS）管理器发布管理系统。系统选择浏览器为IE10.0或FireFox53.0或以上。网站服务器运行环境：Microsoft .NET Framework SDK v4.0。计算机系统选择在分辨率为1024\*768像素上进行开发。

## 第二节 网站主页设计

网站主页作为一个系统的门户，其页面设计的友好性，决定能否提高用户的使用率，同时本BBS论坛系统在主页面不使用大幅度图片，因为图片越大越清晰，读取速度越慢，不利于系统的执行效率，本系统使用部分小图片，减少了系统反应时间，给用户提供友好的系统反馈。

BBS论坛系统主页包含四部分：页面顶部导航部分，中间最新帖子和板块区部分，底部说明部分。

BBS论坛系统主页导航部分：



图5-1 主页导航

该部分使用了asp.net 母板页技术，使用母板页技术的好处是当系统有多个页面时，而每个页面只有部分内容不同，其他内容相同，就可以将相同部分的内容放进母板页，下次创建新的页面时，就可以绑定相应的母板页。使用母板页很好的实现了页面设计的模块化，而且实现了代码重用性。

该部分内容包括首页链接，个人资料链接，帖子搜索链接，发帖链接，板块链接，用户登录链接，注册链接，以及留言链接。其中除了板块链接是绑定到数据库的，其他链接都是通过页面控件设计的，包括Label，LinkButton，ImageButton，Button等控件。

板块导航绑定主要通过Repeater控件实现，具体代码如下：

<asp:Repeater ID="Repeater2" runat="server" DataSourceID="ObjectDataSource0">

<ItemTemplate><ahref="ClassContent.aspx?classid=<%#Eval("Classid")%>"><%#Eval("Classname") %></a></ItemTemplate> </asp:Repeater>

BBS论坛系统主页最新帖子：

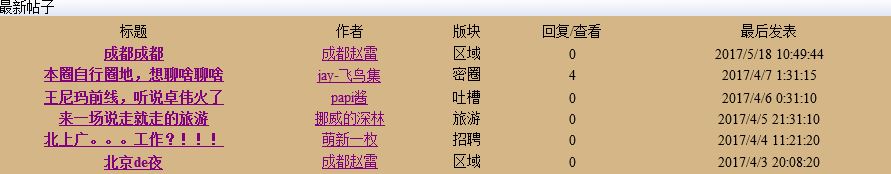


图5-2 主页最新帖子

主页最新帖子内容也是通过Repeater控件实现，具体代码如下：



图5-3 主页最新帖子代码

BBS论坛系统主页板块区：



图5-4 主页板块区

主页板块区罗列了整个系统的板块，游客可以选择性的进入其中一个板块。其实现代码如下：

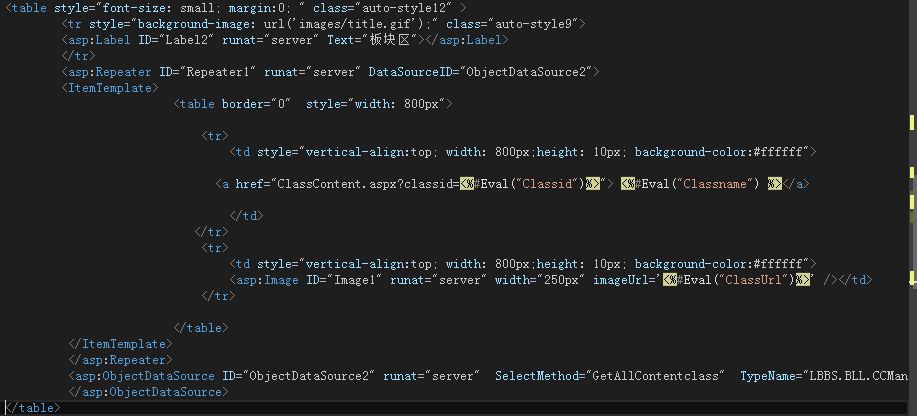


图5-4 主页板块区代码

BBS论坛系统主页底部说明部分：

C:\Users\dai\Desktop\主页底部.JPG

图5-5 主页底部说明

这一部分作了简短说明和留言链接。

## 第三节 用户管理模块

通过需求分析，对用户管理模块进行了三个部分划分，包括用户注册，用户登录，用户信息管理。

1. 用户注册

访问者进入BBS论坛系统主页后，点击注册，进入用户注册页面，该页面填写信息包含用户名，密码，重复密码，用户头像，E-mail，性别。画面效果如下：

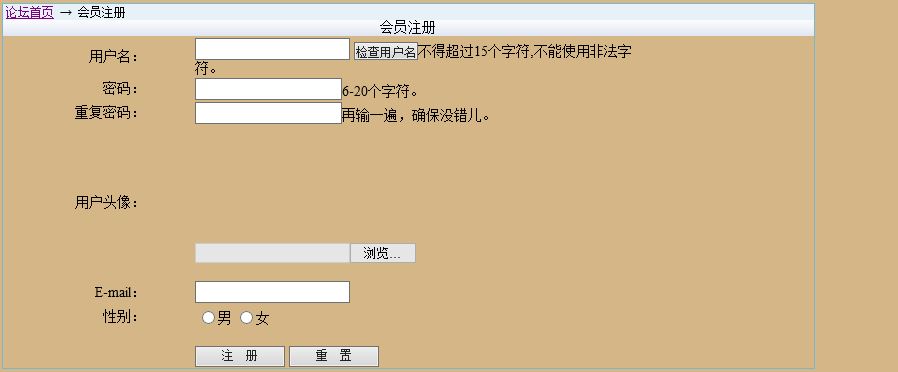


图5-6 用户注册页面

注册页面整体效果简单但信息具体，给用户提供了良好的体验。用户输入用户名后，点击文本框后面的检查用户名按钮，系统会进行用户名搜索，如果检验出用户名已经被使用了，提示用户名已存在。反之，提示该用户名可以正常使用。

检查用户名实现代码如下：

protected void Button3\_Click(object sender, EventArgs e)

{

string user\_name = name.Text;

string str = "select \* from BBS\_User where UserName='" + user\_name + "'";

OleDbConnection connection = new OleDbConnection("provider=sqloledb;data source=cocodai;initial catalog=bbs\_dba;user id=sa;password=Password00");

connection.Open();

OleDbCommand cmd = new OleDbCommand(str, connection);

OleDbDataReader dr = cmd.ExecuteReader();

if (user\_name == "")

{

Label2.Text = "×";

Label3.Text = "用户名不得为空!";

return;

}

if (dr.Read())

{

Label2.Text = "×";

Label3.Text = "该用户名已存在!";

return;

}

if (user\_name.Length > 15)

{

Label2.Text = "×";

Label3.Text = "请按要求填写用户名!";

return;

}

Label2.Text = "√";

Label3.Text = "该用户名可正常使用!";

}

上传用户头像使用的是FileUpload控件进行头像图片的上传，并且为控件设置

Onpropertychange事件，具体功能实现如下：

添加js函数部分：

<script type="text/javascript">

function show(upimg)

{

var dd=document.getElementById("divview");

dd.filters.item("DXImageTransform.Microsoft.AlphaImageLoader").src = upimg;

dd.style.width = 166;

dd.style.height = 190;

dd.filters.item("DXImageTransform.Microsoft.AlphaImageLoader").sizingMethod = 'scale';

}

</script>

添加一个Image控件，一个FileUpload控件，具体代码如下：

<div id="divview" style="filter : progid:DXImageTransform.Microsoft.AlphaImageLoader(sizingMethod=image);" class="auto-style1">

<!--用于显示图片--> <asp:Image ID="Image1" src="#" runat="server" Width="100" Height="100" alt="预览" />

</div>

<asp:FileUpload ID="FileUpload1" runat="server" onpropertychange="show(this.value)"/>

1. 用户登录

BBS论坛系统用户进入论坛主页，点击登录按钮后，用户进入到论坛登录画面。本系统登录单独在一个页面上，用户需填写好用户名和密码后点击登录方可以用户的身份访问论坛。

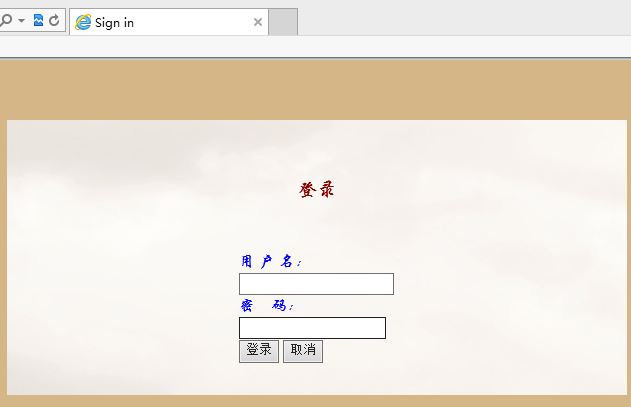


图5-7 用户登录页面

用户填写用户名和密码后提交系统检证，验证成功，跳转到系统主页，系统导航栏表示用户名，用户继续进行论坛的访问，并行使更多权力。验证失败，系统跳出用户登录失败提示，并重新登录。用户也可点取消不进行登录，并返回论坛主页。

1. 用户信息管理

用户信息管理下包括：查看个人信息，修改个人信息，删除用户操作。根据论坛的权限管理，所有论坛访问者均可查看论坛发帖者的个人信息，论坛注册用户可以修改个人信息，系统管理员可以进行用户删除操作。

查看个人信息方式：未注册用户（即游客）进入BBS论坛系统，浏览最新帖子板块，点击一位用户，进入用户信息页面，即可查看用户信息。会员用户登录BBS论坛后点击导航部分个人资料，即可进入个人信息页面，查看个人信息。

修改个人信息：仅限BBS论坛注册会员用户登录论坛后，点击导航部分个人资料进入个人信息页面。个人信息页面如下：



图5-7 个人信息页面

进入到个人信息页面后，点击修改，可以编辑修改个人信息，提交后交由系统检验，修改成功，后台更新用户信息表。

删除用户：系统管理员登录后台管理页面，选择用户管理，选择删除修改用户，实现页面如下：



图5-8 用户管理

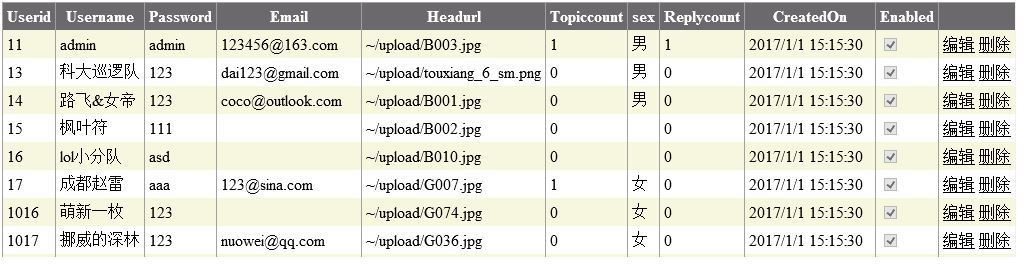


图5-9 删除修改用户

管理员可以对用户信息进行修改编辑，并提交。删除用户后用户信息将被从系统用户列表中移除。页面控件部分代码如下：

<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"

BackColor="White" BorderColor="#DEDFDE" BorderStyle="None" BorderWidth="1px"

CellPadding="4" DataSourceID="ObjectDataSource1" ForeColor="Black"

GridLines="Vertical" DataKeyNames="Userid">

<FooterStyle BackColor="#CCCC99" />

<RowStyle BackColor="#F7F7DE" />

<Columns>

<asp:BoundField DataField="Userid" HeaderText="Userid"

SortExpression="Userid" />

<asp:BoundField DataField="Username" HeaderText="Username"

SortExpression="Username" />

<asp:BoundField DataField="Password" HeaderText="Password"

SortExpression="Password" />

<asp:BoundField DataField="Email" HeaderText="Email" SortExpression="Email" />

<asp:BoundField DataField="Headurl" HeaderText="Headurl"

SortExpression="Headurl" />

<asp:BoundField DataField="Topiccount" HeaderText="Topiccount"

SortExpression="Topiccount" />

<asp:BoundField DataField="sex" HeaderText="sex" SortExpression="sex" />

<asp:BoundField DataField="Replycount" HeaderText="Replycount"

SortExpression="Replycount" />

<asp:BoundField DataField="CreatedOn" HeaderText="CreatedOn"

SortExpression="CreatedOn" />

<asp:CheckBoxField DataField="Enabled" HeaderText="Enabled"

SortExpression="Enabled" />

<asp:CommandField ShowDeleteButton="True" ShowEditButton="True" />

</Columns>

<PagerStyle BackColor="#F7F7DE" ForeColor="Black" HorizontalAlign="Right" />

<SelectedRowStyle BackColor="#CE5D5A" Font-Bold="True" ForeColor="White" />

<HeaderStyle BackColor="#6B696B" Font-Bold="True" ForeColor="White" />

<AlternatingRowStyle BackColor="White" />

</asp:GridView>

<asp:ObjectDataSource ID="ObjectDataSource1" runat="server"

DataObjectTypeName="LBBS.Model.User" DeleteMethod="DeleteUser"

SelectMethod="GetAllUser" TypeName="LBBS.BLL.UserManager"

UpdateMethod="ModifyUser"></asp:ObjectDataSource>

页面通过控件绑定到数据库后，调用业务逻辑层LBBS.BLL的UserManager类DeleteUser（）方法，GetAllUser（）方法，ModifyUser（）方法，业务逻辑层Bll调用数据访问层Dal的UserService类的相应方法对数据库进行删除，查询，修改操作。

业务逻辑层相关代码如下：

public static IList<User> GetAllUser()

{

return UserService.GetAllUser();

}

public static int ModifyUser(User user)

{

return UserService.MoifyUser(user);

}

public static int DeleteUser(User user)

{

return UserService.DeleteUser(user);

}

数据访问层调用数据部分代码如下：

public static IList<User> GetAllUser()

{

string sqlAll = "select \* from BBS\_User";

return GetUserBySql(sqlAll);

}

public static int MoifyUser(User user)

{

string sql = "update BBS\_User set UserName=@UserName,Email=@Email,HeadUrl=@HeadUrl,CreatedOn=@CreatedOn,Sex=@Sex,Topiccount=@Topiccount,Replycount=@Replycount where UserID=@UserID";

SqlParameter[] para = new SqlParameter[]{

new SqlParameter ("@UserName",user.Username ),

new SqlParameter ("@Password",user.Password ),

new SqlParameter ("@Email",user.Email),

new SqlParameter ("@HeadUrl",user.Headurl ),

new SqlParameter ("@UserID",user.Userid ),

new SqlParameter ("@CreatedOn",user.CreatedOn),

new SqlParameter ("@Topiccount",user.Topiccount),

new SqlParameter ("@Replycount",user.Replycount),

new SqlParameter ("@Sex",user.sex )};

return DBHelper.ExecuteCommand(sql, para);

}

public static int DeleteUser(User user)

{

string sql = "delete BBS\_User where UserID=@UserID";

SqlParameter[] para = new SqlParameter[]

{

new SqlParameter("@UserID",user.Userid)

};

return DBHelper.ExecuteCommand(sql, para);

}

private static IList<User> GetUserBySql(string safesql)

{

List<User> list = new List<User>();

DataTable table = DBHelper.GetDataSet(safesql);

foreach (DataRow row in table.Rows)

{

User user = new User();

user.Userid = (int)row["Userid"];

user.Username = (string)row["Username"];

user.Password = (String)row["Password"];

user.Email = (string)row["Email"];

user.Topiccount = (int)row["Topiccount"];

user.Replycount = (int)row["Replycount"];

user.sex = (string)row["sex"];

user.CreatedOn = (DateTime)row["CreatedOn"];

user.Headurl = (string)row["Headurl"];

user.Enabled = (bool)row["Enabled"];

list.Add(user);

}

return list;

}

## 第四节 帖子管理模块

BBS论坛系统帖子管理模块是系统主要功能部分，做好帖子管理，能够有效提升用户体验，增加用户流动性，提高论坛的访问量。本系统模仿了一般BBS论坛系统的帖子管理功能，查找方便，用户体验高。其中包括搜索帖子，浏览帖子详细信息，发表新帖，回复帖子，修改回复内容，删除回复，修改帖子信息，删除帖子。

1. 查询搜索帖子

BBS论坛访问者进入论坛主页后，点击系统导航部分帖子搜索 ，进入帖子搜索画面。

帖子搜索画面如下：

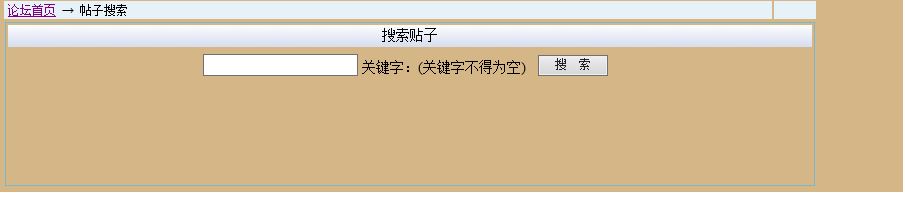


图5-10 搜索帖子画面

搜索查询帖子画面包括返回论坛首页链接，搜索文本框，搜索按钮，提醒关键字。论坛访问者进入到本页面后，在搜索文本框里输入关于搜索帖子的关键词，点击搜索按钮提交系统处理，系统进行帖子查询，并将帖子标题信息反馈到系统页面。

搜索查询帖子部分代码如下：

public void bind(string title)

{

if (leave\_name.Text == "")

{

return;

}//如果搜索文本框里面不输入帖子关键词并搜索，系统返回帖子信息为空

PagedDataSource pdsList = new PagedDataSource();

pdsList.DataSource = ContentManager.GetContentByTitle(title);//数据掉用业务逻辑层ContentManager类的GetContentByTitle（）方法，进而调用数据访问层相应方法对数据库数据进行操作。

pdsList.AllowPaging = true;

GridView1.DataSource = pdsList;

GridView1.DataBind();

if (this.GridView1.Rows.Count == 0)

{

Page.ClientScript.RegisterStartupScript(this.GetType(), "", "alert('关于该关键字的主题不存在!');", true);

return;

}//如果数据库操作返回的个数为空的话，系统跳出报错窗口，提示该关键字的主题不存在。

}

1. 浏览帖子详细信息

论坛访问者（包括游客和用户）进入BBS论坛后，在主页会看到最新帖子板块，该板块通过绑定数据，记录了帖子列表中最新发表的前六项帖子，访问者可以选择其中一项帖子进入其详细信息页面，查看帖子内容，及整个帖子回复内容；论坛访问者也可以在论坛主页下方选择一项感兴趣的板块，点击板块名后进入板块，板块列出了本板块所属所有帖子，可以选择其中一项帖子进入帖子详细信息页面，查看帖子内容，以及整个帖子回复内容。

板块下帖子内容如下：



图5-10 板块帖子内容画面

帖子详细信息画面：

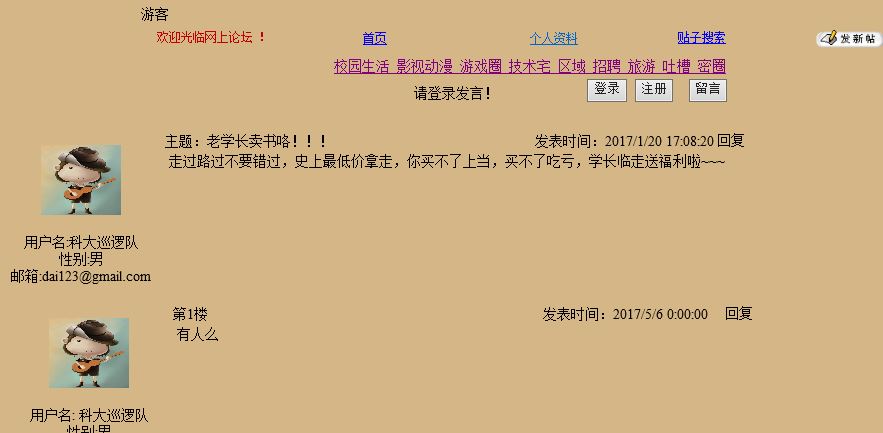


图5-11 帖子详细信息画面

论坛访问者在板块页面下可以看到绑定的板块名，帖子信息。其部分实现绑定的控件代码如下：

<asp:ObjectDataSource ID="ObjectDataSource0" runat="server" SelectMethod="GetContentByClassID1" TypeName="LBBS.BLL.CCManager">

<SelectParameters>

<asp:QueryStringParameter DefaultValue="1" Name="classid" QueryStringField="classid" Type="Int32" />

</SelectParameters></asp:ObjectDataSource>

通过Repeater控件，ObjectDdataSource控件，调用了业务逻辑层CCManager类，进而调用数据访问层CCService类，实现对板块帖子的查询，并将帖子信息反馈到系统页面。

1. 发表新帖

BBS论坛系统用户进入系统主页面后，点击发新帖图标，进入发帖页面，如果是注册登录用户，系统会显示用户名等用户信息。用户未登录情况下，发帖页面不显示用户信息。发帖页面包括主题编辑文本框，内容编辑文本框，选择分类（帖子所属板块）下拉列表，具体画面如下所示：

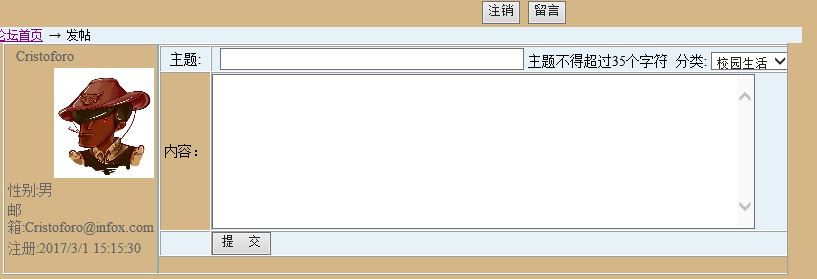


图5-12 发新帖画面

用户编辑内容，点击提交，交由系统处理后，如果内容符合要求，则系统更新数据库用户帖子列表，并反馈到系统页面。用户如果输入不正确的内容，系统则会提示发表内容不符合要求，重新发表。

页面首次加载获取用户登录的信息：

protected void Page\_Load(object sender, EventArgs e)

{

int userid = Convert.ToInt32(Session["UserID"]);

if (!IsPostBack)

{

ViewState["UserID"] = Request.QueryString["UserID"];

}

}

通过控件绑定业务逻辑层方法，调用数据访问层，并将用户数据记录到发帖页面，部分控件代码如下：

<asp:ObjectDataSource ID="ObjectDataSource1" runat="server"

SelectMethod="GetUserById1" TypeName="LBBS.BLL.UserManager">

<SelectParameters>

<asp:SessionParameter DefaultValue="1" Name="userid" SessionField="userid"

Type="Int32" />

</SelectParameters>

</asp:ObjectDataSource>

为提交按钮注册事件，在用户点击提交按钮后，判断用户是否登录，未登录用户发表新帖，系统会跳出提示，提醒用户未登录。发表帖子的标题和内容是不能为空的，否则系统会跳出提示。页面内容填写完毕后，点击提交，系统将调用业务逻辑层ContentManager类的addContent（）方法，进一步通过数据访问层ContentService类的addContent（）将帖子内容增加到系统数据库。判断帖子是否发表成功部分代码如下所示：

if (ContentManager.addContent(content) > 0)

{

this.Page.ClientScript.RegisterStartupScript(this.GetType(), "", "<script>alert('增加成功！');</script>");

return;

}

else

{

this.Page.ClientScript.RegisterStartupScript(this.GetType(), "", "<script>alert('增加失败！');</script>");

return;

}

1. 回复帖子

BBS论坛系统用户登录论坛后，选择任意一个帖子进行访问，进入帖子详细页面后，将页面拉到最底部，会出现一个回复文本框，一个回复按钮，一个取消按钮。详细画面如下：



图5-13 帖子回复画面

用户编辑回复内容后，点击回复，提交系统，系统首先验证用户是否已经登录，如果用户未登录，系统提示用户进行登录，并将回复编辑框设置为不可用。成功提交内容后，系统将调用业务逻辑层ReManage类的addReply（）方法，进而调用数据访问层ReService类的addReply（）方法，将回复内容插入到系统数据库，更新回复内容列表，并将新添加的回复内容反馈到系统页面。判断回复内容是否发表成功部分代码如下：

if (ReManager.addReply(contentreply) > 0)

{

this.Page.ClientScript.RegisterStartupScript(this.GetType(), "", "<script>alert('回复成功！')</script>");

Response.Redirect("TopicContent.aspx?topicid=" + topicid + "");

return;

}

else

{

this.Page.ClientScript.RegisterStartupScript(this.GetType(), "", "<script>alert('回复失败！');</script>");

return;

}

1. 修改删除帖子信息

BBS论坛系统管理员登录后台管理界面，选择发帖管理，选择修改删除帖子，功能实现页面如下所示：



图5-14 发帖管理画面

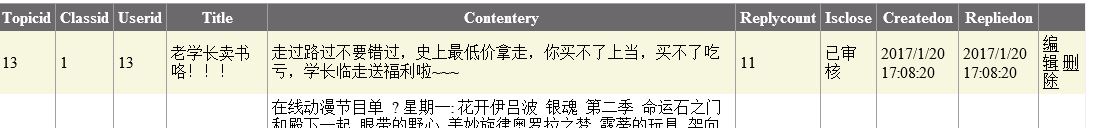


图5-15 修改删除帖子内容画面

帖子修改页面通过GridView控件绑定数据源，CommandField控件设置编辑和删除按钮。管理员选择其中一项帖子，选编辑或删除对帖子进行操作。前台部分控件代码如下：

<asp:CommandField ShowDeleteButton="True" ShowEditButton="True" />

<asp:ObjectDataSource ID="ObjectDataSource1" runat="server"

DataObjectTypeName="LBBS.Model.Content"

SelectMethod="GetAllContent" TypeName="LBBS.BLL.ContentManager"

UpdateMethod="ModifyContent" DeleteMethod="DeleteContent">

<UpdateParameters>

<asp:Parameter Name="title" Type="String" />

<asp:Parameter Name="contententry" Type="String" />

<asp:Parameter Name="classid" Type="Int32" />

<asp:Parameter Name="topicid" Type="Int32" />

<asp:Parameter Name="replycount" Type="Int32" />

<asp:Parameter Name="isclose" Type="String" />

<asp:Parameter Name="createon" Type="DateTime" />

<asp:Parameter Name="replyon" Type="DateTime" />

</UpdateParameters>

</asp:ObjectDataSource>

帖子修改页面通过控件，获取业务逻辑层ContentManager类的方法GetAllContent（）方法，ModifyContent（）方法，DeleteContent（）方法，进而处理数据访问层相应方法，对数据库进行查找，修改，删除操作。系统完成数据库的操作后，更新数据库帖子列表，并将帖子信息反馈到系统页面。

1. 修改删除回复内容

BBS论坛系统管理员登录后台管理界面，选择回复管理，选择修改删除回复，功能实现页面如下所示：



图5-16 回复管理画面



图5-17 修改删除回复内容画面

回复修改页面通过控件，获取业务逻辑层ReManager类的方法GetAllReply（）方法，ModifyContentReply（）方法，DeleteContentReply（）方法，进而处理数据访问层相应方法，对数据库进行查找，修改，删除操作。系统完成数据库的操作后，更新数据库回复内容列表，并将回复内容信息反馈到系统页面。

## 第五节 论坛管理模块

BBS论坛系统论坛管理模块包括板块管理和留言管理。本系统并不会限制板块的数量，提供了板块的增加，板块的修改，板块的删除操作。留言管理提供了用户在系统留言页面查看留言和发表留言。

1. 板块管理

板块管理通过FormView控件将板块的增加，修改，删除操作放在一个页面上操作。

增加板块：管理员登录后台管理界面，选择板块管理，选择增肌板块，功能实现画面如下所示：



图5-18 板块管理画面



图5-19 增加板块画面



图5-20 增加板块内容画面

实现板块增加的控件代码如下所示：

<InsertItemTemplate>

版块编号:

<asp:TextBox ID="ClassidTextBox" runat="server" Text='<%# Bind("Classid") %>' />

<br />

版块名:

<asp:TextBox ID="ClassnameTextBox" runat="server" Text='<%# Bind("Classname") %>' />

<br />

<br />

版块图片:

<asp:TextBox ID="ClassurlTextBox" runat="server" Text='<%# Bind("ClassUrl") %>' />

<br />

<asp:LinkButton ID="InsertButton" runat="server" CausesValidation="True" CommandName="Insert" ForeColor="#660066" Text="插入" onclick="InsertButton\_Click1" />

&nbsp;<asp:LinkButton ID="InsertCancelButton" runat="server" CausesValidation="False" CommandName="Cancel" ForeColor="#660066" Text="取消" />

</InsertItemTemplate>

添加板块页面通过控件，获取业务逻辑层CCManager类的方法GetContentclassById（）方法，AddContentclass（）方法，进而处理数据访问层相应方法，对数据库进行查找，插入操作。系统完成数据库的操作后，更新数据库板块内容列表，并将板块信息反馈到系统页面。

修改删除板块：管理员登录后台管理界面，选择板块管理，选择修改删除板块，功能实现画面如下所示：



图5-21 板块管理画面



图5-22 删除板块画面

修改删除板块页面通过控件，获取业务逻辑层CCManager类的方法GetContentclassById（）方法，ModifyContentclass（）方法，DeleteContentclass（）方法，进而处理数据访问层相应方法，对数据库进行查找，修改，删除操作。系统完成数据库的操作后，更新数据库板块内容列表，并将板块信息反馈到系统页面。

1. 留言管理

BBS论坛系统提供了一个留言页面，可以使论坛用户查看留言和发表留言。留言界面如下所示：

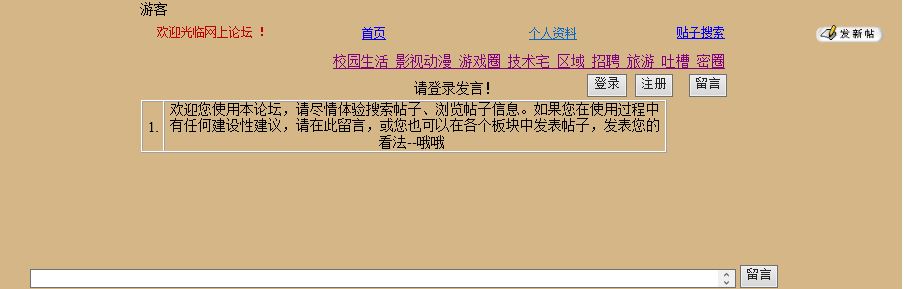


图5-23 留言画面

留言页面使用了母板页，上面部分是系统导航部分，下方绑定数据库里面留言表里面的内容，一个留言文本框，一个留言按钮。用户登录主页，点击留言按钮，进入留言页面，编辑留言信息（留言信息不能为空），提交系统处理后，系统更新数据库留言列表，并将留言信息反馈到留言页面。

留言管理页面使用Repeater控件绑定数据源，ObjectDataSource控件对系统数据库进行处理，相关控件代码如下所示：

<asp:Repeater ID="Repeater1" runat="server" DataSourceID="ObjectDataSource1" >

<ItemTemplate>

<table style="TABLE-LAYOUT: fixed" border="1" cellspacing="0" cellpadding="0" width="200">

<tr>

<td style="align-content:flex-start" width="20">

<%#Eval("MessageID") %>.

</td>

<td style="WORD-WRAP: break-word" width="500">

<%#Eval("MessageContent") %>

</td>

</tr>

</table>

<br /> <br /><br /><br /><br /><br /><br />

</ItemTemplate>

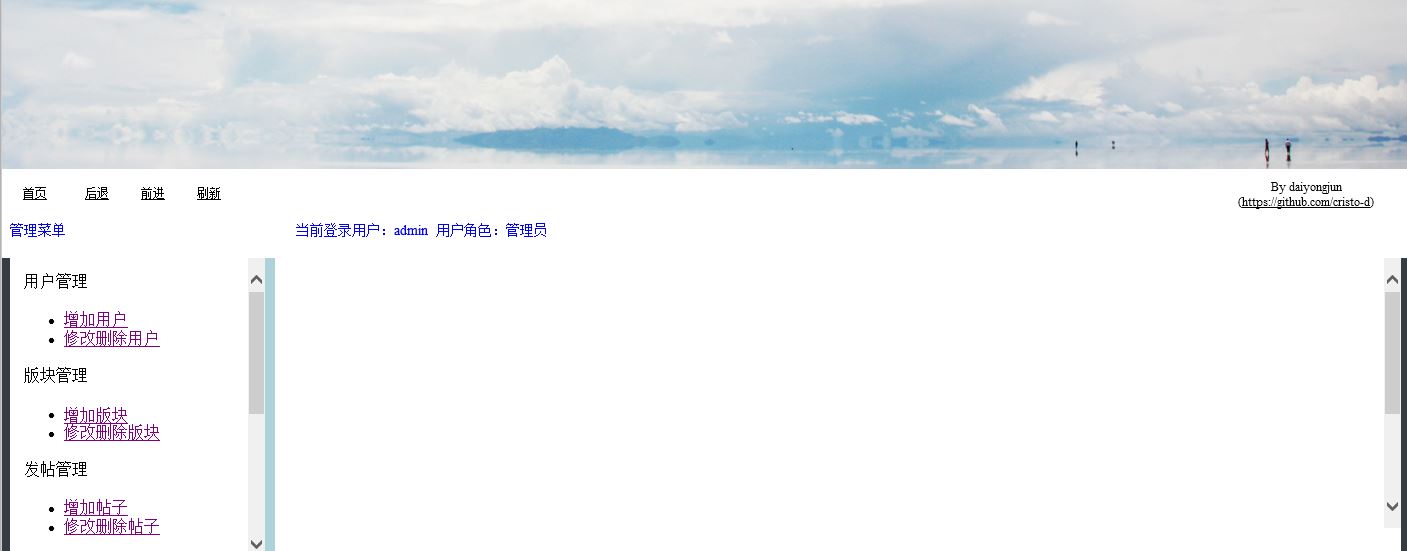
</asp:Repeater>

<asp:ObjectDataSource ID="ObjectDataSource1" runat="server" SelectMethod="GetMessageById" TypeName="LBBS.BLL.MessageManager">

</asp:ObjectDataSource>

## 第六节 后台界面设计

后台界面使用iframe元素创建了三个内联框架，分别是：topframe，leftframe和rightframe，通过将三个框架有序的组合起来形成后台管理界面。实现画面如下所示：



其中页面顶部是在topframe框架下设计的，包括网站log，首页标签，后退标签，前进标签，刷新标签，说明文字。点击首页标签能够rightframe框架下跳出系统主页，点击后退时调用“javascript:history.go(-1);”能够在当前状态下回到上一个状态，点击前进时调用“javascript:history.go(1);”能够回到下一个状态。点击刷新调用“javascript:window.parent.location.reload();”能够刷新页面内容，回到初始状态。

左边页面是在leftframe框架下设计的，包括网站的用户管理，发帖管理，板块管理，回复管理。管理员选择其中一项进行网站的管理。

页面右边是一个空白框架rightframe，主要功能是在不影响后台页面其他内容的情况下，将管理员的所有操作内容都在这个框架下显示出来，方便管理员进行系统内容的更新。

# 结 论

1. 关于系统的不足与优点

这次开发的BBS论坛系统能够完成用户的注册与登录，浏览帖子的信息，发表和回复帖子信息，搜索帖子，查看个人信息等功能，虽然很简单，但简洁而不失韵味。系统使用了母板页的功能，能够很方便的开发网站，管理员对于网站也能够很方便的管理，不同权限的论坛访问者也能够使用网站的不同功能。

BBS论坛简洁的设计也反映了部分界面的不完善，用户登录页面设计的过于简单了，只有用户名和密码的检证，如果加入一些其他验证信息，验证码，验证图片之类的，不仅能够美化登录页面，同时还能提高用户的安全性。

1. 关于系统的扩展开发

在开发本次的BBS论坛系统时，有很多方面未考虑，一些论坛功能也没能实现。比方说，在设置论坛板块版主时，没有使版主能够行使相应板块的帖子管理，还不够具体。另外，系统没有实现用户添加好友的功能，给帖子点赞的功能，同时通过帖子访问量，将帖子置顶的功能。很地方都是参考网上论坛设计后，按自己的理解去设计开发的，可能存在问题。希望以后学习到更多的之后，弥补论坛的不足，将论坛完善一下。

1. 关于自己开发本系统后的感想与体悟

在这次开发BBS论坛系统的过程中，我遇到了很多问题，最终都通过询问老师和同学，或是通过网上获取帮助，解决了开发中遇到的难点。这次开发通过软件工程的设计思想，整理了一遍系统开发的完整流程。开发BBS论坛系统是通过三层架构的思想设计的，开发过程中感受到了三层架构了方便。同时感受到前期准备的重要性。在遇到问题的同时，不能钻牛角尖，可以查看其他相关的资料，换个可行的方法再试一遍。

# 参考文献

[1]张李义，孟健，陈为思.网站开发与管理.高等教育出版社.2008.03

[2]丛书编委会.ASP动态网站开发案例教程.中国电力出版社.2011.11.1

[3]前沿文化，何琛.网页设计与网站建设完全学习手册.科学出版社.2011.7.1

[4]吕双.房大伟.刘云峰.ASP.NET学习手册.电子工业出版社.2011.3.1

[5]汪洋..NET应用架构设计原则、模式与实践.机械工业出版社.2012.1

[6]常倬林.从零开始学ASP.NET.电子工业出版社.2011.2

[7]邹欣，刘起芳.DIV+CSS商业案例与网页布局.中国铁道出版社.2010.9

[8]Rebecca M.Riordan.Head First Ajax.O’Reilly Media,Inc,USA.2008.9.2

[9]Scott Millett.Professional ASP.NET Design Patterns.Wrox.2011.11

[10]曾毅.SQL Server数据库技术大全.清华大学出版社.2009.5

[11]覃征，虞凡，王志敏.程序设计方法与优化.西安交通大学出版社.2004.1

[12]王英英，张少军，刘增杰.SQL Server2012从零开始学.清华大学出版社.2012.10

[13]帕特里克.ADO.NET 4 从入门到精通.清华大学出版社.2012.1

[14] 王海明，韩瑞霞.国内BBS研究现状评述[J].兰州石化职业技术学院学报.2004(04)

[15]Marco Bellinaso.ASP.Net MVC 1.0 Website Programming:Problem-Design-Solution.Wrox.2009.6.1

# 致谢

首先感谢在大学四年的生活中给我提供细心指导的各位老师，谢谢大学四年里无私奉献的导员，也感谢四年同窗的大学校友和一块生活四年的舍友，在你们的热心的鼓励、善意的批评和无私的帮助下，我也一步步从青涩走向成熟。

本次毕业设计是在满春雷老师的指导下从初步定题，到制定计划，再到完成设计，最终编码实现系统功能。感谢满老师孜孜不倦的帮我们省定系统设计，是满老师组织我们一起讨论毕业设计的方案，在刚开始定题目的时候，满老师给我们提供了很多指导性的帮助，听取了老师的意见，我查阅了相关知识之后，才最终确定了题目，并且对于BBS论坛系统的设计有了一个初步清晰的认识，然后确定了系统功能的框架，完成此次毕业设计的实现。同时也感谢和我一组的其他成员们，在我做这次毕业设计期间，给我了很多帮助和鼓励。

感谢我的父母，在我失败时给我支持和鼓励，在我决定报考计算机专业时，是他们给了我鼓励，让我坚持了自己的选择，虽然学习的过程很辛苦，但依然坚持的走下去。感谢他们在我的生活和学习中无私奉献了他们的关爱。

再次对在此次毕业设计中帮助和鼓励我的人表达感谢，同时也对过去和将来对我提供帮助和产生影响的人表达感谢。