**分类号：TP311.5 U D C：D10621-408-(2017)2511-0**

**密 级：公 开 编 号：2014081052**

**成都信息工程大学**

**学位论文**

**在线博客系统的设计与实现**

|  |  |
| --- | --- |
| **论文作者姓名：** | **孔浩** |
| **申请学位专业：** | **软件工程** |
| **申请学位类别：** | **工学学士** |
| **指导教师姓名（职称）：** | **郑文斌（副教授）** |
| **论文提交日期：** | **2018年05月22日** |

**在线博客系统的设计与实现**

**摘 要**

正所谓学习使人进步，作为一名大学生，这点体现得更为重要。现在是一个互联网飞速发展的时代，通过互联网学习已经变成一个主流的方式。然而互联网资源多种多样，来源不一，所以我们经常会遇到不知如何充分利用网络资源的问题，而且，有效保存网络资源也是一个不可忽视的问题。现在，多种关于笔记管理的软件相继推出，并越来越受到人们的青睐，如：有道云笔记等。然而这些系统只为用户提供了笔记的网络保存，却无法从网络上添加有效的网络资源，所以，开发一个能下载网络资源并可有效使用的系统越来越有必要。

本文详细阐述了基于C#的网络爬虫的笔记管理系统的设计与实现。首先简明扼要地介绍了系统相关的基础知识和总体设计，然后详细阐述了基于.NET平台的笔记归档、笔记展示与收录、笔记数据获取、用户登录四个主要模块的设计与实现。最后对系统单元测试和模块测试做了较详细的介绍。经测试，系统功能达到要求、性能良好，界面美观友好易用，具有较强的可移植性和系统运行稳定等优点，达到了预期的设计目标。

**关键词：**笔记管理系统；网络爬虫；数据源；C#；WinForm

**Design and Implementation of Notebook Management System Based on C # and Web Reptile**

**Abstract**

It is the so-called learning to make progress. As a college student, which is more important. Now is an era of rapid development of the Internet, through the Internet to learn has become a mainstream way. However, a variety of internet resources, different sources, so we often encounter I do not know how to make full use of network resources, and effective preservation of network resources is also a problem that can’t be ignored. Now, a variety of notebook management software have been introduced, and more and more people of all ages, such as: Youdao Cloud Notes and so on. However, these systems only provide users with a note of the network to save, but can’t add effective network resources from the network. So the development of a system is more and more necessary, which can meet a network resources can be downloaded and effective use.

This paper elaborates the design and implementation of the note management system based on C# and web crawler. First of all, it shows the basic knowledge and overall design of the system, and then introduces the design and implementation of the four main modules based on the .NET platform, includes the notes display and collection, the notes data acquisition and the user login. Finally, the system unit test is elaborated in detail. After testing, the system functions meets the requirements, and the system has a good performance, beautiful interfaces and easy to use, with strong portability and system stability, etc., and achieves the desired design goals.

**Key words:** Notebook management system; Web Crawler; Data Source; WinForm

**目 录**

论文总页数：37页

[1 引言 1](#_Toc484710111)

[1.1 课题背景 1](#_Toc484710112)

[1.2 课题的研究意义 1](#_Toc484710113)

[1.3 国内外研究现状分析 1](#_Toc484710114)

[1.4 课题的研究路线 1](#_Toc484710115)

[2 理论基础与支撑技术 2](#_Toc484710116)

[2.1 人机交互设计简介 2](#_Toc484710117)

[2.2 WinForm技术简介 2](#_Toc484710118)

[2.3 Microsoft Visual Studio 2010简介 2](#_Toc484710119)

[2.4 XML技术简介 3](#_Toc484710120)

[2.5 HTML技术简介 4](#_Toc484710121)

[3 系统模块划分与设计 4](#_Toc484710122)

[3.1 系统模块划分 4](#_Toc484710123)

[3.2 系统总体结构设计与分析 6](#_Toc484710124)

[3.3 类结构设计 8](#_Toc484710125)

[3.4 XML结构组成设计 11](#_Toc484710126)

[3.5 界面设计 12](#_Toc484710127)

[3.6 系统功能及流程设计 14](#_Toc484710128)

[3.6.1 笔记数据获取 14](#_Toc484710129)

[3.6.2 笔记展示与管理 15](#_Toc484710130)

[3.6.3 笔记归档 16](#_Toc484710131)

S[3.6.4 用户登录 17](#_Toc484710132)

[3.7 系统运行环境 17](#_Toc484710133)

[3.8 系统技术难点及关键技术 18](#_Toc484710134)

[3.9 系统设计优点 19](#_Toc484710135)

[4 系统的实现 19](#_Toc484710136)

[4.1 HtmlHelper类的实现 19](#_Toc484710137)

[4.2 笔记数据获取模块的实现 22](#_Toc484710138)

[4.3 笔记展示与管理模块的实现 24](#_Toc484710139)

[4.4 笔记归档模块的实现 26](#_Toc484710140)

[4.5 用户登录模块的实现 27](#_Toc484710141)

[5 系统运行及测试结果 29](#_Toc484710142)

[5.1 测试方案 29](#_Toc484710143)

[5.2 测试运行平台 29](#_Toc484710144)

[5.3 系统单元测试 30](#_Toc484710145)

[5.3.1 单元测试结果及分析 30](#_Toc484710146)

[5.3.2 系统测试结果分析及说明 32](#_Toc484710147)

[5.4 系统模块测试 32](#_Toc484710148)

[5.5 测试结果结论 33](#_Toc484710149)

[结 论 34](#_Toc484710150)

[参考文献 35](#_Toc484710151)

[致 谢 36](#_Toc484710152)

[声 明 37](#_Toc484710153)

# 1 引言

## 1.1 课题背景

在这个互联网无处不在的时代，几乎人人或多或少都会涉及到互联网。互联网总的说来是一个能为人们带来便利的东西，人们可以从互联网上获取到自己想要的资料。然而，这些网络资源往往是分散到不同的网站上，并不能从一个地方集中查找，所以很多时候为人们查找和学习带来了很多不便。正所谓学习使人进步，假如我们拥有了一套能便于管理自己想要资源的系统，那将大大提高我们的学习效率。

## 1.2 课题的研究意义

本系统旨在构建一个功能完善、性能良好、界面美观、操作友好并方便用户查阅管理笔记的操作平台。主要包含用户从互联网下载相关的网页资源、编辑并添加私有笔记、对网页资源的展示、对网页资源的修改、对网页资源的分类、上传和下载个人数据包，以便用户对私有数据的管理及查阅。实现在此平台下，用户大大提高自己的学习效率目的。[13]

## 1.3 国内外研究现状分析

目前，国内出现了两款主流的与笔记管理相关的软件，分别是：印象笔记、有道云笔记。而这两款软件都很好的诠释了笔记的有效管理，能很好地做到在笔记的异地上传与下载。然而，这两款软件的重点都是在笔记的管理上，从而忽视了从网络上获取笔记资源并可离线使用的功能。而基于C#的网络爬虫的笔记管理系统能很好的克服这个缺点，从而实现学习资源查找的便利化管理，同时侧重于对笔记资源的获取并在本地进行有效的分类管理与使用。恰恰弥补了印象笔记与有道云笔记的不足。本系统的笔记来源主要分为网上下载与用户编辑，前者通过用户提供的网页链接获取到对应的HTML代码并保存到本地文件使用，后者则是将用户编辑内容保存到文件并使用。这样，就可以达到网络资源和用户资源长久保存并使用的目的。

## 课题的研究路线

* 对课题范围业务知识点进行调研。在此基础上，完成功能性需求和非功能性需求，最后做成《需求分析说明书》。
* 完成系统开发平台和框架选择，划分系统层次，划定系统边界，明确系统包层次结构。将成果物以文档记录说明。
* 程序编码，使用.Net4.0平台和C# 编程语言编写，使用WinForm 技术实现系统开发。
* 系统测试，分别对系统进行单元测试和模块测试。

# 2 理论基础与支撑技术

## 2.1 人机交互设计简介

人机界面是系统和用户之间进行交互和信息交换的媒介，它实现信息的内部形式与人类可以接受形式之间的转换[1]。

其主要特点：

* 系统界面直观简单，作为人机交互界面，开始设计开发时就考虑到软件系统界面设计需要遵循相同，直观，功能突出这三个原则。使用户在一个简单直观的界面下，灵活运用软件功能。
* 一致性。保持一致的几个界面元素是语言，布局和设计。界面的一致性使用户能够更好地了解如何操作，从而提高效率。
* 反馈提供。界面始终保持与用户沟通。
* 易于使用的原则。符合用户习惯是方便使用的第一原则。其他包括实现目标功能的最小操作数，鼠标移动距离最短等。

## 2.2 WinForm技术简介

Visual C#是Windows环境下的应用程序开发工具，其特点是可视化编程、事件驱动和交互式[2]。而WnForm是C#基础上一种主流的开发技术，是.Net开发平台中的一部分，全称为Window Form（Windows窗体）。.Net为用它开发的应用程序提供了大量的类库，这些类库支持RAD（快速应用程序开发），这些类库都被封装在System.Windows.Forms这个命名空间中，在这个命名空间中定义了许多类，在基于WinForm的程序开发时，通过继承和扩展这些类来绘制出各种各样的用户界面[3]。

其主要特点：

* 功能强大。Windows窗体可用于设计窗体和可视化控件，以创建丰富的基于Windows的应用程序。
* 操作方便。新的数据提供程序管理：数据提供程序管理提供易于连接OLEDB和ODBC数据源的数据控件，包括Microsoft SQL Server、Microsoft Access、Jet、DB2以及Oracle等数据库。
* 使用安全。Windows窗体充分利用公共语言运行库的安全特性。这就意味着，一切都可以通过Windows窗体来实现，包括在浏览器中运行的不可信控件和用户硬盘上安装的完全可信的应用程序。

## 2.3 Microsoft Visual Studio 2010简介

Visual Studio是微软公司推出的开发环境，是目前最流行的Windows平台应用程序开发环境。Visual Studio 2010版本于2010年4月12日上市，其集成开发环境（IDE）的界面被重新设计和组织，变得更加简单明了。Visual Studio 2010同时带来了 NET Framework 4.0、Microsoft Visual Studio 2010 CTP( Community Technology Preview--CTP)，并且支持开发面向Windows 7的应用程序。除了Microsoft SQL Server，它还支持 IBM DB2和Oracle数据库。

其主要特点：

* 代码管理工具比平台更重要。对于大型的开发项目，由于现有规模和类型的业务已经非常复杂，再加上多年的遗留应用，开发平台最合适的选择是拥有强大的代码管理能力。Visual Studio 2010的工具功能不仅功能越来越强大，而且还具有其他产品的优势。 更值得注意的是，Visual Studio Team System的首次推出与软件开发生命周期管理平台的开发工具完全集成，并且集成了敏捷方法和CMMI规范。Visual Studio团队系统除了软件开发过程更加严格，软件开发的质量和时间也更容易控制。
* 灵活高效的全新IDE。对软件开发来说IDE是一项了不起的发明，它可以很大地减低软件开发的门槛。微软从1998年发布了VC++6.0以来，VS的IDE就成为了软件开发工具的重要标杆，许多的开发工具都会模仿VS的IDE。微软能战胜其竞争对少，主要是因为创建了高效率的IDE。在Microsoft Visual Studio 2010中，微软通过WPF技术重新打造了VS的编辑器。新打造的编辑器可以实现很多2010版本前的VS的IDE无法想象的功能。如：代码无级缩放，文档地图，多窗口更新，代码自动生成，这些新的IDE特性都极大地提高了开发效率。
* 快速开发。Visual Studio 2010提供了一个扩展，可帮助开发团队改进协作。包括帮助数据库专业人员和平面设计师加入开发过程的工具。 例如，.NET Framework提供了一个用于解决常见编程任务的构建块（预构建软件），使其能够以极好的最终用户体验快速构建应用程序。 此外，Visual Studio和.NET Framework的组合减少了对通用管道代码的需求，减少了开发时间并使开发人员能够专注于业务问题。

## 2.4 XML技术简介

XML是eXtensible Markup Language 的缩写，是由万维网联盟定义的一种语言，称为可扩展标记语言。所谓可扩展性是指XML允许用户按照XML规则自定义标记。XML文件是由标记及其所标记的内容构成的文本文件，与HTML文件不同的是，这些标记可自由定义，其目的是使得XML文件能够很好地体现数据的结构和含义。W3C推出XML的主要目的是使得Internet网络上的数据相互交流更方便，让文件的内容更加通俗易懂[4]。

XML技术现在已广泛应用于数据转换、科学计算、图形图像、无线通信、Web开发等技术领域。尤其是Internet的迅猛发展，不仅成就了Java，同时也成就了XML。XML已经成为Web开发中不可或缺的技术之一[5]。

## 2.5 HTML技术简介

HTML语言是一种标记语言，是网页设计的基础。它的词汇由尖括号包围的单词组成，如<body>。这些具有尖括号的单词通常配对。 HTML文档的最大特点是整个文档的内容用尖括号标记来结构化。每对尖括号在HTML中也称为元素。例如，被<body> 被尖括号包围的部分被称为body元素。 元素也可以嵌套在其他元素中。HTML文档是结构化文档，也因为HTML文档由嵌套的元素组成。网页用HTML语言编写，由网页浏览器下载并翻译成我们看到的彩色页面。大多数网页允许浏览器查看其HTML源代码，也可以下载到本地。手写HTML是繁琐的工作，开发网站有很多工具，如Dreamweavercs6，HTML编译器等[6]。

# 3 系统模块划分与设计

## 3.1 系统模块划分

网络爬虫笔记管理系统是基于WinForm的用户下载添加并管理与查阅笔记资源的管理系统平台。主要包含笔记下载与添加、笔记归档、笔记展示、笔记管理、用户数据包上传下载。系统可以对用户从网站上下载的网页资源和用户自己添加的笔记进行集中管理，可对这些笔记进行浏览、搜索、收藏、修改、删除等，还可进行登录，把登录用户私有数据包进行上传和下载。

总体功能划分为：笔记数据获取系统，笔记展示与管理系统，笔记归档系统，用户登录系统四大功能模块。

功能划分如图1所示。



图1 网络爬虫笔记管理系统功能图

图1展示的对网络爬虫笔记管理系统简要的模块划分图。

1. 笔记数据获取模块功能描述

用户通过该模块进行笔记数据源的添加，添加的笔记数据为其他模块的使用奠定基础。

表1 笔记数据获取模块

|  |  |
| --- | --- |
| 功能编号 | F\_1 |
| 功能名 | 笔记数据获取模块 |
| 使用角色 | 用户 |
| 运行条件 | 无 |
| 功能描述 | 用户可以通过输入网址下载网页内容和直接编辑并保存笔记两种方式来添加笔记数据，输入网址下载方式将获取网页内容及其对应的图片文件和样式文件，编辑并保存方式添加数据源则是将用户自己编辑的内容保存为HTML文件形式。 |
| 其他说明 | 输入网址下载方式由于对其对应网页所对应的样式文件大小未知，所以可能出现展示效果不佳的情况。 |

1. 笔记展示与管理模块功能描述

用户可通过该模块将添加的笔记进行显示，以及对笔记的删除、修改、关键字匹配搜索等。

表2 笔记展示与管理模块

|  |  |
| --- | --- |
| 功能编号 | F\_2 |
| 功能名 | 笔记展示与管理模块 |
| 使用角色 | 用户 |
| 运行条件 | 软件中笔记数据源不为空 |
| 功能描述 | 用户可对获取的笔记数据进行展示、修改保存、删除操作，还可通过键入关键字与笔记内容进行匹配，在匹配的结果中可在显示时对匹配的关键字进行高亮显示 |
| 其他说明 | 无 |

1. 笔记归档模块功能描述

用户可通过该模块进行笔记的分类处理，并且，用户可为系统添加不同的分类。

表3 笔记归档模块功能

|  |  |
| --- | --- |
| 功能编号 | F\_3 |
| 功能名 | 笔记归档模块 |
| 使用角色 | 用户 |
| 运行条件 | 软件中笔记数据源不为空 |
| 功能描述 | 用户可以对获取到的笔记数据进行归档，新增加的笔记最开始是未归档的，用户可把未归档笔记添加到相应的分组。用户还可对分组进行添加、重命名、删除。已经归档的笔记还可添加到书签中。 |
| 其他说明 | 无 |

1. 用户登录模块功能描述

用户通过该模块进行登录，登录后用户可进行数据包上传、下载，以及数据源的切换使用。

表4 用户登录模块功能

|  |  |
| --- | --- |
| 功能编号 | F\_4 |
| 功能名 | 用户登录模块 |
| 使用角色 | 用户 |
| 运行条件 | 系统无用户正在登录 |
| 功能描述 | 用户可以登录自己的账号，无账号可以注册。登录了自己的账号后，可以从服务端下载以前版本的数据包，然后用户可以在公用数据源和私有数据源进行切换使用。用户登录后还可上传现在用户对应在本地的数据源到服务器。 |
| 其他说明 | 用户下载的数据包将覆盖现有用户私有的数据包 |

## 3.2 系统总体结构设计与分析

系统的设计使用三层架构的设计方式，系统的各子模块之间的功能是相互独立的[7]，并且模块与模块之间耦合度低。由于各个子模块之间的耦合度低，所以在需要修改一个子模块时，往往能把对其他模块的影响降到最低，从而提高了开发效率，系统的灵活性也大大提高，从而减低了系统的开发风险[8]。

系统的层次结构划分如下：

1. 数据访问层

数据访问层是连接系统数据和业务逻辑的一个结构层，其主要作用就是将系统对数据的操作都封装在一块，通过业务逻辑层的调用来统一操作数据。本系统的数据访问层主要为笔记数据和分组设计了相应的类，如图2所示。

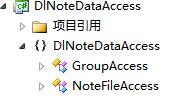


图2 数据访问层类视图

笔记信息在数据访问层对应的类为NoteFileAccess，该类主要封装了对笔记对应的XML文件的增删改查等操作。该类通过Note实体类来与XML文件进行数据转换，如：对笔记信息的添加就需要为对应的操作函数提供一个Note对象，然后通过转换，存储到对应的XML文件中。

分组信息在数据访问层对应的类为GroupAccess，该类主要封装了对分组对应的XML文件的增删改查等操作。该类通过Group实体类来与XML文件进行数据转换。如：对分组信息的更新操作就需要为对应的更新函数提供Group对象，然后通过数据转换，更新XML文件中的内容。

1. 业务逻辑层

业务逻辑层是在数据访问层和用户界面层之间的一个结构层，其主要作用就是根据自身系统的业务逻辑来调用数据访问层，进行符合业务逻辑的数据操作。和数据访问层相对应，业务逻辑层也设计了笔记信息和分组信息相关的业务处理类，同时为提高系统运行效率，还在此处设计了一个业务数据缓存类。如图3所示。

图3 业务逻辑层类视图

与笔记信息相对应业务逻辑层的类为NoteBiz，该类与笔记相关的业务息息相关。通过该类可以获取已分组笔记、未分组笔记、已添加标签的笔记及其他的一些对笔记的信息的批量删除或添加操作，同时，在添加或删除笔记信息时，会对应删除或添加该条信息对应的笔记文件。

与分组信息相对应的业务逻辑层的类为GroupBiz，该类与NoteBiz类似，主要根据分组信息相关业务需求对GroupAccess类中的功能进行拼装，以此满足不同业务逻辑需求。

CacheBiz则是基于前两个类来封装的，由于系统操作数据频繁，而且数据量较大，所以设置缓存是有必要的，CacheBiz是一个静态类，该类为整个系统缓存笔记信息和分组信息。只有在数据更改时才相应地从文件中读取数据，更新缓存。这样大大降低了系统访问文件的次数，从而提高了系统效率。

1. 用户界面层

用户界面层是系统的最上层，主要作用是对从业务逻辑层获取的信息进行合理的展示，同时，通过用户的操作命令来进行数据的增删改。该层主要提供了一个主体框架以及镶嵌到其内部的五个子框架。即与之对应的有一个主窗体及五个用户控件。

户界面的呈现主要是根据主界面展示，子控件嵌套的方式来设计的。主界面通过合理的布局来控制子控件在主界面上镶嵌的位置和大小。主界面起展示作用，而子控件则为用户提供交互的功能。主界面框架布局如图4所示。



图4 主界面框架布局图

## 3.3 类结构设计

1. Note类

笔记信息对象对应的类为Note，该类是一个实体类，其属性与笔记信息属性一一对应，其类视图如图5所示，类相关描述见表4。

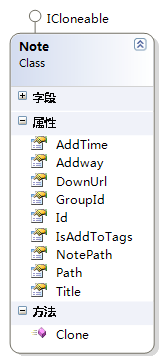


图5 Note类视图

表4是对Note类相关属性及方法的描述，该类对应的实体与笔记详细信息相对应。

表4 Note表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 作用 |
| AddTime | 属性 | 笔记添加时间 |
| AddWay | 属性 | 笔记添加方式（网站下载、文件添加、未知） |
| DownUrl | 属性 | 笔记下载网址（只有AddWay为网站下载时才有） |
| GroupId | 属性 | 笔记所在分组Id(未分组则为空) |
| Id | 属性 | 笔记对象唯一标识 |
| IsAddToTags | 属性 | 笔记是否添加标签 |
| NotePath | 属性 | 笔记文件对应绝对路径 |
| Path | 属性 | 笔记文件对应相对路径 |
| Title | 属性 | 笔记标题 |
| Clone | 方法 | 复制笔记对象 |

1. Group类

分组信息对应的类为Group，该类是一个实体类，其属性与分组信息属性一一对应，其类视图如图6所示，类相关描述见表5。



图6 Group类视图

表5示对Group类相关属性与方法的简要描述，该类对应的实体与分组信息相对应。

表5 Group表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 作用 |
| Id | 属性 | 分组对象唯一标识 |
| Name | 属性 | 分组名称 |
| ParentId | 属性 | 所在分组对应Id |
| Clone | 方法 | 复制分组对象 |

1. Tree<T>类

由于分组信息存储在XML文件中的形式为列表结构，而不是树形结构，所以，在读取分组信息列表后，需要将其转换成树形结构。而Tree<T>类就能达到这样的功能。Tree<T>的设计与树类似，设计为泛型，可适用于不同类型的集合。其类视图如图7所示，类相关描述见表6。

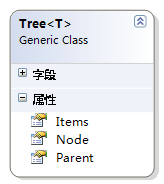


图7 Tree<T>类视图

表6作用是对Tree<T>泛型类相关属性与方法进行简要描述，该类的作用是将Group类集合树形化，同时，该类还是和其他类型的列表数据。

表6 Tree<T>表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 作用 |
| Itmes | 属性 | 子节点集合 |
| Node | 属性 | 该节点对应的对象 |
| Parent | 属性 | 父节点 |

1. NewForm类

NewForm类是一个以Form为基类，然后通过GDI+绘制以及其他事件函数的配合，得到的一个外形美观的窗体。类视图如图8所示，类相关描述见表7。该类的设计思路大致为：继承Form类->隐藏窗体边框，只剩下视图区->在视图区绘制边框、背景、标题栏、标题栏文本等->为最大化、最小化、关闭图标编写功能代码->通过计算鼠标点击区域在鼠标相关事件中实现窗体拖拽、窗体大小更改等功能。

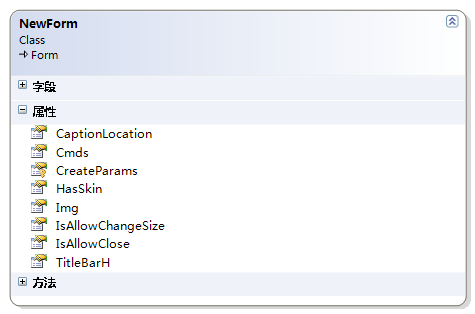


图8 NewForm类视图

表7是对NewForm类相关属性与方法的描述，该类是系统所有窗体的基类，通过继承Form类来实现的。

表7 NewForm 表

|  |  |  |
| --- | --- | --- |
| 名称 | 类型 | 作用 |
| CpationLocation | 属性 | 设定标题栏中标题位置（左中右） |
| Cmds | 属性 | 窗体右上角图标及其命令集合 |
| CreateParams | 属性 | 窗体各种参数集合 |
| HasSkin | 属性 | 是否显示皮肤更换图标 |
| Img | 属性 | 窗体图标 |
| IsAllowChangeSize | 属性 | 是否允许改变窗体大小 |
| IsAllowClose | 属性 | 是否显示关闭按钮图标 |
| TitleBarH | 属性 | 标题栏高度 |

## 3.4 XML结构组成设计

本系统未采用数据库作为数据源，而是采用XML文件作为数据源来存储相关数据。这样，在系统通过网络存储与下载数据时就很方便，数据可移植性强。本系统主要涉及两种信息的存储，一种就是笔记信息，一种则是分组信息。

笔记信息对应的XML文件名为NoteCfg.xml，其数据结构如下：

|  |  |
| --- | --- |
| |  | | --- | | <?xml version="1.0" encoding="utf-8"?>  <ArrayOfNote>  <Note>  <Id>185dd4b5-7ccd-41db-8f09-bcb4568f491c</Id>  <Title>C#设计模式(2)</Title>  <GroupId>4f74ce57-a57d-44b2-b8de-ca031bb0456d</GroupId>  <AddTime>2017-04-04T10:16:28.0165055+08:00</AddTime>  <Addway>未知</Addway>  <DownUrl>http://www.baidu.com</DownUrl>  <IsAddToTags>true</IsAddToTags>  </Note>  </ArrayOfNote> | |

该XML文件内容是由List<Note>对象经过XML序列化获得，每个Note节点对应一个Note对象，他们通过Id来区分。该XML文件存储了所有的笔记信息，通过笔记信息可以获取到笔记相关属性。其子节点作用见表8。

表8 NoteCfg.xml文件结构表

|  |  |
| --- | --- |
| 节点名称 | 描述 |
| Id | 用作Note节点的唯一标识，通过Id可以获取到给笔记信息对应资源存放的位置。 |
| Title | 笔记标题 |
| GroupId | 所在分组对应的分组Id（未分组则为空） |
| AddTime | 笔记添加的时间 |
| AddWay | 笔记添加的方式（网站下载、手动添加、未知） |
| DownUrl | 下载笔记对应的网址（下载方式为网站下载时有用） |
| IsAddToTag | 是否为该笔记设定标签 |

分组信息对应的XML文件名为GroupCfg.xml，其数据结构如下：

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <Groups>  <Group Id="ed2baf95-88ca-4b8a-a4d6-e9b5756c11cd" ParentId="" Name="爬虫笔记本" />  <Group Id="4f74ce57-a57d-44b2-b8de-ca031bb0456d" ParentId="ed2baf95-88ca-4b8a-a4d6-e9b5756c11cd" Name="C#" />  <Group Id="65e54cc7-ae68-434c-af9e-feb16ce7e7fa" ParentId="ed2baf95-88ca-4b8a-a4d6-e9b5756c11cd" Name="毕业设计论文" />  </Groups> |

该XML中每一个Group节点都对应一个分组信息，也就对应着一个Group对象。他们通过Id属性来区分，然后通过ParentId属性来确定其父节点。其属性作用见表9。

表9 GroupCfg.xml文件结构表

|  |  |
| --- | --- |
| 属性名称 | 描述 |
| Id | 用作Group节点的唯一标识 |
| ParentId | 通过该属性寻找其位于哪个分组下 |
| Name | 分组名称 |

## 3.5 界面设计

本系统界面布局合理，颜色搭配得当，而且用户使用方便简单，带给用户很好的用户体验。其主要设计思路如下。

1. 基于业务特点的考虑

本系统是一个笔记的管理与展示的系统，所以，系统应该围绕“管理”、“展示”这两个词来布局。而展示需要占据系统大部分界面，而且，用户使用本系统的最终目的是查看笔记内容，所以系统应该从主界面上划分很大一块区域作为笔记展示的区域，而剩下的区域就可用来作为笔记的管理区域。所以系统采用了左边笔记管理，右边笔记展示的布局来设计整个界面。左边的笔记管理区域可以通过页的切换来决定使用哪种笔记相关功能。

1. 基于用户个人喜好的考虑

由于每个用户视觉与喜好差异，用户使用系统时，对系统主题颜色喜好不一。所以该系统专门为用户设计了主题选择的功能，用户可以根据自己需求，选择系统主题，如图9所示。系统主题配色多种多样，灵活多变，基本能满足所有用户的视觉体验。

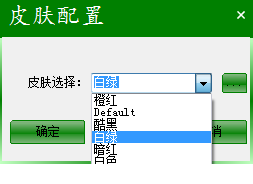


图9 皮肤配置图

1. 基于人机交互方便的考虑

由于该系统主要是用于用户学习的系统，笔记管理只是为了方便学习，最终用户更多时间是用户笔记的展示与学习上，所以，在用户长时间重点关注笔记展示区域时，隐藏笔记管理区域来扩大笔记展示区域是一个不错的选择。所以系统采用了用户手动隐藏笔记管理区域的设计方案，如图10所示，从而减少了系统在用户使用过程中带给用户的影响。



图10 隐藏笔记管理界面图

## 3.6 系统功能及流程设计

### 3.6.1 笔记数据获取

**（1）功能概述**

笔记数据获取系统可以通过输入网址下载网页内容和直接编辑并保存笔记两种方式来添加笔记数据，输入网址下载方式将获取网页内容及其对应的图片文件和样式文件，编辑并保存方式添加数据源则是将用户自己编辑的内容保存为HTML文件形式。

**（2）业务流程**

笔记获取的业务流程如图11所示。

****

图11 笔记数据获取业务流程图

图11时对笔记数据获取流程的展示，首先，用户进入系统。然后可以选择是否登录，无论是否登录都可以添加笔记数据。用户可以选择通过输入有效网址和手动添加两种方式来添加笔记数据。前者若输入的网址无效，将下载失败。后者则是将用户输入的内容保存下来，作为笔记资源。成功添加后，系统相应界面将同步更新。

### 3.6.2 笔记展示与管理

**（1）功能概述**

用户可以对获取的笔记数据进行展示、修改保存、删除操作，还可以通过键入关键字与笔记内容进行匹配，在匹配的结果中可在显示时对匹配的关键字进行高亮显示。

**（2）业务流程**

笔记展示与管理业务流程如图12所示。

****

图12 笔记展示与管理业务流程图

图12是对笔记展示与管理模块的流程的展示，首先用户进入系统，可以选择是否登录，由于该模块是离线模块，所以无论是否登录都能使用。在系统中，用户可以进行笔记展示，笔记关键字匹配搜索，以及笔记的删除和笔记的修改。

### 3.6.3 笔记归档

**（1）功能概述**

用户可以对获取到的笔记数据进行归档，新增加的笔记最开始是未归档的，用户可把未归档笔记添加到相应的分组。用户还可对分组进行添加、重命名、删除。已经归档的笔记还可添加到书签中。

**（2）业务流程**

笔记归档的业务流程如图13所示。

****

图13 笔记归档业务流程图

图13是对笔记归档模块流程的展示，该模块与上面两个模块类似，都是离线模块。所以用户可以登录使用，也可不登录使用。用户进入系统后，可以从未归档笔记列表中选择笔记添加到相应的分组，进行笔记归档。归档后的笔记还可以添加到标签列表。同时，用户还可以对分组进行添加，删除，重命名。

### 3.6.4 用户登录

**（1）功能概述**

用户可以登录自己的账号，无账号可以注册。登录了自己的账号后，可以从服务端下载以前版本的数据包，然后用户可以在公用数据源和私有数据源进行切换使用。用户登录后还可上传用户在本地的数据源到服务器。

**（2）业务流程**

用户登录的业务流程如图14所示。

****

图14 用户登录业务流程图

图14是对用户登录模块的流程展示，这个模块是一个在线使用的模块。用户通过键入账号密码进行登录，如果账号密码不匹配，则登录失败。登录成功后，用户可以进行数据包上传，数据包下载，以及数据源切换。

## 3.7 系统运行环境

**（1）运行环境**

* 硬件平台：CPU 1000MHz及以上，运行内存1GB及以上，硬盘16GB及以上。
* 操作系统：Windows7、Windows8、Windows10
* 其他有关组件：.NET framework 4.0、IIS、Ftp
* 与该软件共存的应用程序：无

**（2）运行模块组合**

* 爬虫笔记管理系统总共分为四个模块，其中笔记获取模块、笔记归档模块和笔记展示与管理模块在系统打开时就已经随着系统一起加载进来，用户在系统加载成功后边可以随时使用这三个模块，而用户登录模块是供用户选择使用的模块，只用用户登录成功后才能使用。
* 用户可以根据自己的需求选择不同的模块来使用，在无法连接网络的情况下，用户只能选择使用本地数据源。
* 系统分为在线使用与离线使用两种使用状态，在线功能为单独的一个模块，只有在用户登录成功后方可使用。登录成功后，就能把从服务器下载下来的数据包作为数据源来展示与使用，同时还能将存在客户端的个人数据包上传到服务器，供以后使用，这样用户就能在离线状态时使用不同的数据源。

**（3）运行控制**

* 在系统界面是，可以使用鼠标单击或者双击来触发事件进行相应操作。
* 在系统界面上，某些地方也可以通过点击回车键来代替鼠标点击作用。
* 系统模块的运行是根据系统界面上的菜单和按钮来控制的，除“用户登录”模块外的其他三个模块之间的数据同步是根据模块数据更新函数的注册和在模块外回调注册函数来实现的。登录模块的数据同步是通过WebService来请求数据，通过应答结果来同步。

## 3.8 系统技术难点及关键技术

为了更高效更方便地使用网络爬虫笔记系统。经过多次分析与假设，发现要达到这个目的，必须完美地克服以下两个技术难点：数据源的有效提取和保存、增强界面皮肤自适应性。

本系统笔记数据来源主要来自两个方面：第一，用户直接从输入有效网站，下载该网址对应的网络资源。第二，用户直接通过编辑保存来添加数据。第二种方法添加数据的方式对于本系统来讲，可控性高，能高效加载和有效管理。而第二种方法由于网络资源多种多样，所以可控性相对较差。为了提高方式一所对应数据源加载效率，系统采用异步下载并处理数据源的方式。首先从网络上抓取对应的HTML代码，然后从代码中提取对应的图片链接和样式链接，再将获取到的连接进行下载，然后重名得到的图片和样式文件资源，再将HTML代码中的对应的链接部分一一替换，保存至HTML文件中。由于Javascript文件在加载时特别慢，而且大多数网站Javascript代码臃肿，对于本系统而言相对无用，所以会将HTML中的Javascript链接全部屏蔽掉，从而保证了数据源的有效提取和保存。

由于每个用户个人视觉习惯不同，所以需要给用户提供不同视觉体验，供用户自行选择。为了达到这个效果，专门为本系统设定了一套控件。这套控件是用GDI+绘制，基于WinForm，其中有窗体、按钮、列表、复选框、单选框、进度条、菜单等常用控件，而且这套控件都有一套专门的配色管理，可以通过它来为控件配置不同的皮肤。本系统有了这套控件为基础，用户就能根据自己的个人习惯来选择不同的自己喜欢的配色，用户还可自己配置颜色来达到自己的个人需求。

## 3.9 系统设计优点

* 该系统使用了WinForm框架来开发，所以对于开发人员来说，这能大大提高系统的开发速度。该系统使用架构为典型的三层架构，所以能在此基础上进行灵活的开发，可以达到开发思路清晰，可控性高的特点。而且，在需要修改系统某块时简单快捷，而不用牵一发动全身。
* 本系统的开发语言是C#，所以代码简单易懂，可扩展性强。
* 该系统由于是基于文件的操作，所以没有选择数据库来存储数据。而是使用XML结构来存储数据，从而让系统的数据存储地方不用分散到文件和数据库，更方便处理，在系统移植时也会有很高的兼容性。

# 4 系统的实现

## 4.1 HtmlHelper类的实现

HtmlHelper类主要用于处理HTML代码相关操作，如：通过URL下载HTML代码，获取HTML代码中相关网络链接，下载链接中对应资源等。HtmlHelper所有方法如表10所示。

表 10 HtmlHelper类方法表

|  |  |
| --- | --- |
| 方法名 | 作用 |
| GetHtmlByUrl | 通过链接下载HTML代码 |
| GetRightHtml | 将HTML代码进行链接提取、链接资源下载、HTML对应链接重名操作，最后得到处理后的HTML代码 |
| GetTitleByHtml | 得到HTML代码中对应<Title>标签中的内容 |
| ReplaceImgUrl | 替换HTML代码中的图片链接为下载图片后重新更改的名字。 |
| CallImgMatchEvaluator | 用正则表达式替换对HTML代码中对应图片链接时Replace函数需要回调的内容。 |
| ReplaceLinkFileUrl | 下载<link>标签中对应链接部分对应的资源，同时用下载资源重新更换的名字替换对应链接部分。 |
| CallFileMatchEvaluator | 用正则表达式匹配HTML中对应的<link>标签中的链接部分，同时用下载的文件名字替换对应链接。 |
| DownLoadImg | 通过URL下载图片 |
| DownLoadFile | 通过链接下载文件，如css文件和js文件等 |
| UrlChange | 由于HTML代码中的链接有些是相对路径，有些是绝对路径，如果是相对路径，则需要根据HTML对应URL来将相对路径转换为绝对路径，然后才能下载相应的资源。 |

接下来详细说明ReplaceImgUrl和DownLoadImg方法。

ReplaceImgUrl代码如下：

|  |
| --- |
| /// <summary>  /// 替换图片链接  /// </summary>  /// <param name="html">html代码</param>  /// <param name="url">html代码对应的连接</param>  /// <param name="path">图片存放的地方</param>  /// <returns>返回替换后的HTML代码</returns>  public string ReplaceImgUrl(string html, string url,string path)  {  if (string.IsNullOrEmpty(html)) //检测HTML文本是否为空  return html;  Regex imgReg = new Regex("<\\s\*img\\s+[^>]\*src\\s\*=\\s\*[\"|']{1}([^\"']+)[\"|']{1}", RegexOptions.IgnoreCase); //匹配图片链接的正则表达式  Html = imgReg.Replace(html, new MatchEvaluator(CallImgMatchEvaluator)); //替换图片链接  foreach (KeyValuePair<string, string> p in imgs)  {  DownLoadImg(p, url, path); //下载图片  }  imgs.Clear(); //清除图片集合  return html;  } |

本段代码是函数ReplaceImgUrl的内容，本函数传入三个参数，分别为：需要操作的HTML文本，该代码对应的链接，连接中需要下载的资源应该存储的路径。首先，检测传入的HTML文本是否为空，如果为空则返回该HTML文本。如果不为空，则编写需要匹配图片链接的正则表达式，同时设定参数为匹配时忽略大小写，然后通过正则表达式匹配符合条件的图像链接，同时在匹配时将图片链接替换成随机生成的Guid字符串，然后将链接和对应的Guid字符串组成一个键值对添加到imgs字典中。最后循环遍历imgs字典中的键值对，下载图片，并保存到对应地方，然后清除imgs字典内容，返回处理后的HTML文本。

DownLoadImg方法代码如下：

|  |
| --- |
| /// <summary>  /// 下载图片  // </summary>  /// <param name="imgUrl">图片链接</param>  /// <param name="url">图片所在HTML文本对应的链接</param>  /// <param name="path">图片保存的路径</param>  void DownLoadImg(KeyValuePair<string, string> imgUrl,string url,string path)  {  if (imgUrl.Key == null||imgUrl.Value==null)  return;  try  {  string newUrl = UrlChange(url, imgUrl.Value); //获取图片链接的绝对路径  HttpWebRequest request = WebRequest.Create(newUrl) as HttpWebRequest;  HttpWebResponse response = request.GetResponse() as HttpWebResponse;  Stream stream = response.GetResponseStream();  Image img = Image.FromStream(stream);  img.Save(path+"\\"+imgUrl.Key); //保存图片  }  catch  {  return;  }  } |

本函数在上面提到的ReplaceImgUrl函数中被调用，该函数传入三个参数，分别为：以图片链接为键下载后的图片新名字为值的键值对，图片所在HTML文本对应链接，下载图片后存储的路径。其编写思路如下：首先判断键值对内容是否为空，如果为空，则结束。如果不为空，则在try区域内继续，通过为UrlChange函数提供HTML对应的链接和图片链接，返回图片链接对应的绝对路径。然后通过传入图片绝对路径链接，来获取一个HttpWebRequest对象，通过该对象发送请求，得到一个HttpWebResponse对象，将该对象中对应的流保存为图片，并存放在传入的路径位置。如果这个过程发生异常，则将会从try区域抛出异常，并在catch区域被捕获。

## 4.2 笔记数据获取模块的实现

笔记数据获取模块主要功能是获取数据源，为其他模块做支撑。其获取数据的方式主要有两种：一种是用户自己编辑保存笔记来添加数据笔记。另一种则是用户输入有效的URL来从网上下载资源保存到本地。这两种方式都是以Note类作为实体类。

对于第一种添加方式，系统采用的是为用户提供一个编辑模板，在模板中，用户可以添加自己想要的文字，以及为文字排版，还能添加链接和图片，最终将编辑的内容保存在HTML文件中。系统可通过模板接口获取到用户编辑的内容，然后让用户提供名称，然后初始化一个Note实例，将实例信息写入到笔记列表文件，同时将HTML文件存到相应位置。

对于第二种添加方式，系统采用的是先下载HTML内容，然后从HTML代码中提取CSS文件链接和图片链接，然后将文件图片一一下载到本地并重新命名，然后再一一替换掉原来链接的名称，最后将HTML代码保存到HTML文件中。然后初始化一个Note实例，将实例信息写入到笔记文件列表，同时将HTML文件存到相应位置。

**主要功能函数**

笔记数据获取模块主要功能函数的功能与操作方式如表11所示。

表11 笔记数据获取模块的主要功能函数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **函数名** | **参数** | **返回类型及返回** | **功能描述** |
| 1 | GetRightHtml(string url, string path) | url:下载所需链接  path:保存路径 | Note，下载成功返回初始化的Note实例 | 通过输入链接获得被处理后的HTML代码 |
| 2 | ReplaceLinkFileUrl(string html,string url, string path) | html:需要替换的HTML代码  url:与该HTML代码对应的链接，用于将HTML代码中的链接相对路径转换为绝对路径，然后下载  path：下载链接CSS文件需要存储的地方 | string，下载并替换成功后返回的HTML代码 | 通过搜索HTML中的CSS链接，下载链接中对应的CSS文件重命名，将名字与原链接对应的地方替换。 |
| 3 | DownLoadImg(KeyValuePair<string, string> imgUrl,string url,string path) | imgUrl：图片链接  url:对应HTML代码的链接  path:下载图片需要存储的地方 | void | 通过链接下载图片，并保存 |
| 4 | UrlChange(string url, string subUrl) | url:HTML对应的链接  subUrl:HTML代码中的相对路径链接 | string，返回相对路径转换为绝对路径后的链接 | 通过提供完整链接和对应HTML代码下的相对路径链接，最终转换为绝对路径的链接 |
| 5 | EditHtmlFrm(Note note) | note:需要编辑的Note实例 | void | 通过传入Note实例对象，为编辑的内容提供保存路径 |

笔记数据获取模块部分关键性代码如下：

|  |
| --- |
| if (noteBiz.IsContainNote(note)) // 检测笔记信息中是否包含对应添加的笔记信息  return;  string title; // 字符变量，用户存储下载的笔记资源的标题  HtmlHelper htmlHelper = new HtmlHelper(); // 创建一个新的HtmlHelper对象  bool isDownSuccess = htmlHelper.GetRightHtml(strs[0],strs[1]+note.Id,out title); //下载  if (!isDownSuccess) // 是否下载成功  {  if (File.Exists(note.NotePath)) // 如果下载不成功，则删除下载的内容  File.Delete(note.NotePath);  }  else  {  note.Title = title; // 下载成功后为note对象的Title属性赋值  if (!noteBiz.AddNote(note) && File.Exists(note.NotePath)) // 添加笔记信息到文件  {  File.Delete(note.NotePath); // 若添加失败，则删除对应笔记文件资源  }  } |

该部分代码的作用是通过网址下载对应的笔记资源。noteBiz为操作笔记信息文件的相关对象，代码首先检测note笔记对象是否在笔记文件中是否已经存在，如果存在，则不做下载操作。如果不存在，则创建一个HtmlHelper对象，通过调用该对象的GetRightHtml函数，来下载对应的笔记资源，如果下载成功，则返回对应的Title值，同时，将title值赋给note对象，再将该对象保存至文件，如果保存失败，则删除下载的笔记资源。如果下载不成功，则检测是否存在对应的下载资源，若存在，则删除。

## 4.3 笔记展示与管理模块的实现

笔记展示与管理模块是完成对笔记数据源的有效展示与管理的模块。系统在从笔记数据获取模块获取到数据之后，需要通过该模块将得到的笔记展示处理。用户可以根据自己的需求来展示相应的笔记。用户还可根据键入关键字来从笔记中匹配符合条件的笔记，大大提高学习效率。同时，用户还能通过对笔记的修改和删除来达到管理笔记数据源的目的。

该模块中对笔记展示的部分是直接从XML文件中读取相应的笔记信息，并实例化为一个个Note对象，然后将对象加载到列表中，由于每个Note对象中包含了对应笔记的存放路径，所以在展示时可以直接读取相应路径下的HTML文件，展示在WebBrowser控件中。关键字匹配问题是通过键入关键字，然后从每个笔记对应的HTML代码中进行正则表达式匹配，将匹配的内容两端加上<span style="background:gold"></span>标签，把关键在进行高亮标记，把匹配的笔记添加到一个列表中，供用户查看。删除笔记则是通过每个Note对象对应的Id，通过调用NoteBiz类下的DeleteNoteById(string Id)函数来删除XML文件中记录和相应位置的文件。修改笔记则是将笔记加载到一个可编辑的WebBrowser中，编辑后保存即可。模块对应代表性界面如图15所示。



图 15 笔记展示界面

**主要功能函数**

笔记展示与管理模块主要功能函数的功能与操作方式如表12所示。

表12 笔记展示与管理模块的主要功能函数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 函数名 | 参数 | 返回类型及返回值 | 功能描述 |
| 1 | SetUrlForWBrowser(string url) | url:需要展示出来笔记的位置或链接 | void | 通过为主界面上的WebBrowser控件设置链接来加载笔记 |
| 2 | GetReplacedHtml(string path, string key) | path:需要替换关键字的HTML文件路路径  key:需要替换的关键字 | string:匹配关键字后返回的HTML代码 | 通过读取提供路径的HTML代码，然后将代码中关键字两端加上<span style="background:gold"></span>标签，进行高亮显示 |
| 3 | IsContainKey(string file, string key) | file:需要检查的文件路径  key:需要匹配的关键字 | bool:返回HTML文件中是否含有key关键字 | 测试对应HTML文件中是否含有需要匹配的关键字 |
| 4 | DeleteNoteById(string Id) | id:需要删除的笔记对应的Id | bool：是否删除成功 | 删除对应Id的笔记记录和相应的笔记文件 |

笔记展示与管理模块部分关键性代码如下：

|  |
| --- |
| if (string.IsNullOrEmpty(filePath)||!File.Exists(filePath))  return null;  lock (lockObj) //锁定，使得数据访问同步  {  string txt = File.ReadAllText(filePath); //读取XML文件内容  return txt;  } |

该部分对应的代码的作用是从NoteCfg.xml文件中读取并返回对应的内容。filePath表示对应的文件的路径。首先代码检测对应文件是否存在，如果不存在，则返回null，如果存在则用让lock代码块中的代码获取lockObj锁，使该块代码只能在一个线程中调用。Lock中的代码块作用则是读取对应文件中的文件，然后返回该文本，最后释放该锁，供其他地方调用。同时使用lockObj对象锁定的地方还有写入NoteCfgXml文件的地方，这样就能保证系统不同时读写该文件，保证数据正确性。

## 4.4 笔记归档模块的实现

笔记归档是方便用户对笔记进行分类而设计的。用户可以自定义添加分组，然后在分组内添加笔记，这样就能根据用户不同的需要，将笔记归类到不同的分组。

笔记归档模块的实现分为两部分，第一，分组的管理。第二，笔记的分类。对于分组管理的实现，系统将分组信息从XML文件中读取出来，初始化成一个个Group对象，然后通过Group对象的ParentId属性与其余Group对象的Id属性匹配，通过Tree<T>泛型类，将读取到的Group列表转换为树形结构，最后显示到TreeView控件中，然后通过TreeView控件的右键菜单进行添加、删除、重命名。笔记的分类的实现则是从一个未分类笔记列表中，通过右键菜单，将笔记直接添加到用户指定的分组。对于已经分组的笔记，用户还能将它添加到标签列表中。模块代表性界面如图16所示。

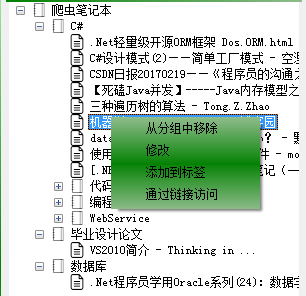


图 16 笔记归档界面

**主要功能函数**

笔记归档模块主要功能函数的功能与操作方式如表13所示。

表13 笔记归档模块的主要功能函数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 函数名 | 参数 | 返回类型及返回值 | 功能描述 |
| 1 | Tree<Group> GetGroupsToTree() | void | Tree<Group>，返回分组信息的树形结构 | 从XML文件中读取出分组信息，转换为树形结构 |
| 2 | LoadTree() | void | void | 加载分组信息到TreeView控件，在界面上展示出来 |
| 3 | AddGroup(Group group) | group:需要添加的分组对象 | bool，添加成功返回True | 添加分组 |
| 4 | RemoveGroup(string Id) | Id:需要删除的分组Id | bool，删除成功返回True | 将对应Id的分组删除 |
| 5 | addTag\_Click(object sender, EventArgs e) | sender：右键菜单项  e：对应的事件对象 | void | 将已经分组的笔记设定为标签 |

笔记归档模块部分关键性代码如下：

|  |
| --- |
| if (CurNote != null) //判断笔记是否为空  {  CurNote.GroupId = groupID; //为笔记分组赋值  bool flag = noteBiz.UpdateNote(CurNote); //更新笔记到文件  if (flag)  {  DlCom.CallMethod("UICatelog", "LoadTree", null); //更新界面  DlCom.CallMethod("UIDownLoad", "InitDownLoadList"); return flag;  }  }  return false; |

该部分代码的作用是，添加文件笔记到某个分组。CurNote为当前需要归档的笔记对象，groupID为用户选择的选择分组对应的Id。首先判定需要归档的笔记对象是否为空，若为空则返回false，若不为空，则通过调用noteBiz对象的UpdateNote函数来更新NoteCfg.xml文件中对应的笔记信息。如果更新成功，则刷新界面上的目录列表和未分类列表。

## 4.5 用户登录模块的实现

用户登录模块是为用户在线使用系统而设计。系统可以分为在线使用和离线使用，在线使用时用户需要通过账号密码来通过身份验证，只有验证成功后才能使用服务端的数据源。

用户登录模块是以服务端启用WebService和Ftp服务器为基础，用户通过调用WebService的功能函数，传送账号和密码到服务端进行验证，如果验证身份成功，则返回一个随机的Guid字符串，并将该字符串在服务端保存，用户需要其他操作时，携带上该字符串方可继续，否则无权操作。如果身份验证失败，则返回null或0。用户登录成功后的数据源上传和下载是通过Ftp服务器进行的，上传时将本地用户私有数据源压缩成一个压缩包，然后上传到服务器。下载时则是将压缩包下载到本地，然后解压到相应位置即可使用。

**主要功能函数**

用户登录模块主要功能函数的功能与操作方式如表14所示。

表14 用户登录模块的主要功能函数

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **函数名** | **参数** | **返回类型及返回值** | **功能描述** |
| 1 | IsLoginSuccess(string id, string pwd) | id:用户登录账号  pwd：用户登录密码 | string:登录成功返回随机生成的Guid字符串；账号密码错误返回null；已有用户登录返回”0” | 验证用户登录账号密码是否正确 |
| 2 | UploadFtp(string filePath, string newName) | filePath:需要上传文件的路径  newName:上传后新的名称 | int，上传成功返回0，上传失败返回-2 | 上传数据包到服务器 |
| 3 | IsContainDirectory(string dicName) | dicName:需要检测的目录名称 | bool, 存在返回True,  不存在返回False | 检测Ftp服务器上是否已存在该目录 |
| 4 | DownloadFtp(string filePath, string fileName) | filePath：下载到本地保存的路径  fileName:需要下载的文件 | Int,下载成功返回0，下载失败返回-2 | 从Ftp服务器下载数据包到本地 |

用户登录模块对应的部分关键性代码如下：

|  |
| --- |
| if(id == null||pwd == null) //判断账号密码是否为空  return null;  string path = this.Server.MapPath("");  path = path + "/bin/userInfo/userInfo.xml";  string xml = File.ReadAllText(path); //从文件中读取账号信息  XmlDocument doc = new XmlDocument();  doc.LoadXml(xml);  XmlNode userNode = doc.SelectSingleNode(string.Format("//user[@id='{0}' and @pwd='{1}']", id, pwd)); //匹配查找账号密码  if(userNode!=null)  {  string key = Guid.NewGuid().ToString();  if (!UserKeys.ContainsKey(id))  {  UserKeys.Add(id, key);  return key;  }  else  return key;  }  else  {  return null;  } |

该段代码的作用是检测用户输入的用户名和密码是否匹配。首先检测用户名和密码是不是都不为空，如果都不为空，则从笔记userInfo.xml文件中读取用户账号密码相关信息文本，并将文本内容加载到XmlDocument对象中。然后通过XPath来找出与对应账号密码相同的XML节点，如果为找到，则返回null，如果找到，则创建一个键值对，键为输入的账号，值为返回的的随机生成的一个Guid字符串，然后将该键值对添加到列表中，并返回该Guid字符串。表示用户登录成功，用户下次操作则可带上该字符串进行操作即可。

# 5 系统运行及测试结果

## 5.1 测试方案

在基于C#的网络爬虫笔记管理系统的测试任务中，主要采用的是黑盒测试。

黑盒测试也称功能测试，它是通过测试来检测每个功能是否可正常使用[9]。 在测试中，程序作为一个无法打开的黑盒子，没有考虑到程序的内部结构和内部特性的情况，在程序界面中进行测试，它只检查程序的功能是否能符合要求地规范正常使用，程序是否可以正确接收输入数据并产生正确的输出信息[10]。 黑盒测试侧重于程序的外部结构，无论内部逻辑结构如何，主要用于软件界面和软件功能的测试[11]。

理论上，黑盒测试只使用穷尽的输入测试，所有可能的输入作为测试用例来考虑，以便检测程序中的所有错误。实际上，有无数的测试用例，人们不仅要测试所有的合法输入，还可以对那些不合法的用户进行测试。以这种方式，全面的测试是不可能的，所以我们必须进行有针对性的测试，通过开发测试用例来指导测试的实施，以确保软件测试覆盖面大。黑盒测试必须能够量化才能真正保证软件的质量，而测试用例是测试具体量化的一种方法。具体的黑盒测试案例设计方法包括等价分类法，边界值分析法，误差估计法，因果法，决策表驱动法，正交检验设计法，功能图法，场景法等。

## 5.2 测试运行平台

根据系统的需求，测试计算机的配置如表15所示。

表15 测试计算机配置

|  |  |  |
| --- | --- | --- |
| 硬件配置 | 计算机类型 | 华硕笔记本电脑 |
| CPU | Intel Core i5 |
| 内存 | DDR3 4G |
| 硬盘 | 2BFB4DCB (500GB） |
| 软件配置 | 计算机操作系统 | Windows 7 旗舰版64位操作系统 |

## 5.3 系统单元测试

单元测试，是指最小测试单元中的软件可以进行检查和验证。对于单元测试单元的意思，一般来说，根据实际情况来确定其具体含义，如C语言是指单位的功能，C#指的是一个类，图形化软件可以参考一个窗口或菜单等。单元测试是软件开发过程中测试活动的最低水平，软件的独立单元将与程序的其余部分隔离测试。

### 5.3.1 单元测试结果及分析

网络爬虫笔记管理系统单元测试结果如表16所示。

表16 网络爬虫笔记管理系统单元测试结果及分析

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **功能模块** | **功能** | **操作步骤** | **实际结果** | **状态** |
| 1 | 笔记数据获取 | 下载笔记 | 1. 打开网络爬虫笔记管理软件； 2. 点击“下载”Tab页标签； 3. 向输入框中输入网址； 4. 点击“下载”按钮。 | 当用户输入网址无效时，提示“下载失败”，当输入有效网址时，提示“下载成功” | 通过 |
| 2 | 笔记数据获取 | 添加笔记 | 1、打开网络爬虫笔记管理软件；  2、点击“下载”Tab页标签；  3、在弹出的编辑框中编辑内容并保存为“Test” | 下载列表中出现标题为“Test”的笔记，添加成功 | 通过 |
| 3 | 笔记展示与管理 | 归档笔记展示 | 1. 打开网络爬虫笔记管理软件； 2. 点击“目录”Tab页标签； 3. 从目录中选择一个笔记节点双击。 | 对应的笔记内容展示在界面上 | 通过 |
| 4 | 笔记展示与管理 | 笔记修改 | 1、打开网络爬虫笔记管理软件；  2、点击“目录”Tab标签；  3、选中一个笔记，右键单击，选择“修改”；  4、在修改窗口中编辑并保存。 | 再次打开被修改的笔记，笔记已经被修改，修改成功 | 通过 |
| 5 | 笔记展示与管理 | 笔记删除 | 1、打开网络爬虫笔记管理软件；  2、点击“目录”Tab标签；  3、选中一个笔记，右键单击，选择“删除”；  4、点击提示框的“是”按钮 | 提示“删除成功”，目录中被选中删除的笔记已经不存在，删除成功 | 通过 |
| 6 | 笔记展示与管理 | 笔记添加到标签列表 | 1. 打开网爬虫笔记管理系统软件； 2. 点击“目录”Tab标签； 3. 选中一个笔记，右键单击，选择“设定标签”。 | 弹出提示“已经添加到书签”，添加成功 | 通过 |
| 7 | 笔记归档 | 笔记添加到分组 | 1、打开网爬虫笔记管理系统软件；  2、点击“下载”Tab标签；  3、选中一个笔记，右键单击，选择“添加到分组”；  4、选中某个分组，点击“确定”按钮 | 提示“添加成功”，在“目录”Tab页中，可以在刚刚选择分组中找到新添加的笔记 | 通过 |
| 8 | 笔记归档 | 添加分组 | 1、打开网爬虫笔记管理系统软件；  2、点击“目录”Tab标签；  3、选中一个分组，右键单击，选择“添加分组”；  4、为添加的分组命名为“Test” | 在“目录”Tab页中，被选择分组下新添加了一个名为“Test”的分组，当再次添加相同名字的分组时，则会提示“已经存在相同名字的分组” | 通过 |
| 9 | 笔记归档 | 删除分组 | 1、打开网爬虫笔记管理系统软件；  2、点击“目录”Tab标签；  3、选中一个分组，右键单击，选择“删除分组”；  4、在提示框中点击“是”按钮 | 弹出提示“删除成功”，被选中的分组消失，删除分组成功 | 通过 |
| 10 | 笔记归档 | 重命名分组 | 1、打开网爬虫笔记管理系统软件；  2、点击“目录”Tab标签；  3、选中一个分组，右键单击，选择“重命名”；  4、在编辑框中输入“TestA” | 被选中分组名称变成“TestA”，重命名分组成功 | 通过 |
| 11 | 用户登录 | 登录 | 1、打开网络爬虫笔记管理系统软件；  2、从工具栏中点击“登录”图标；  3、弹出窗口中输入账号密码，点击登录 | 当账号密码匹配时，主界面的工具栏部分提示“当前登录用户：xxx”。当账号密码不匹配时，提示“账号密码不匹配” | 通过 |
| 12 | 用户登录 | 注销 | 1、打开网络爬虫笔记管理系统软件；  2、从工具栏中点击“登录”图标；  3、弹出窗口中输入正确的账号密码，点击登录；  4、登录成功后点击工具栏中的“注销”图标，点击弹出框中的“是”按钮。 | 工具栏中提示“当前登录用户：无”，注销成功 | 通过 |
| 13 | 用户登录 | 数据上传 | 1、成功登录用户；  2、点击工具栏中的“上传数据包”图标，点击提示框中的“是”按钮。 | 弹出提示“上传成功”，上传数据包成功 | 通过 |
| 14 | 用户登录 | 数据下载 | 1、成功登录用户；  2、点击工具栏中的“下载数据包”图标；  3、在“文件浏览”窗口中选择一个压缩包，点击右键，选择“下载”，点击提示框中的“是”按钮。 | 弹出提示“下载成功”，下载数据成功 | 通过 |

### 5.3.2 系统测试结果分析及说明

网络爬虫笔记管理系统测试结果及分析如表17所示。

表17 网络爬虫笔记管理系统系统测试结果及分析

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **功能** | **预期结果** | **实际结果** | **状态** |
| 1 | 下载数据包 | 数据包被成功下载到本地，解压并替换了原来的数据 | 数据包被下载后，切换到用户私有数据源，使用的是下载的数据包。 | 通过 |
| 2 | 系统安全性测试 | 系统能安全运行，保证用户之间数据独立 | 用户能独立使用自己的数据 | 通过 |

## 5.4 系统模块测试

软件测试主要分为黑盒测试和白盒测试[12]，黑盒测试是按功能说明进行操作，而白盒测试则是根据软件内部逻辑进行操作测试，这里我们选取两个比较重要的模块进行黑盒测试的展示：

表18 笔记数据获取模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 系统 | 网络爬虫的笔记管理系统 | 版本 |  |
| 功能模块名 | 笔记数据获取模块 | 对应开发人员 | 卢勇 |
| 用例编号 | TestCase1 | 用例设计者 | 卢勇 |
| 测试人员 | 卢勇 | 测试日期 | 2017.05.15 |
| 功能特性 | 下载笔记资源和手动添加笔记资源 | | |
| 测试目的 | 验证笔记资源能否下载成功，能否手动添加成功 | | |
| 前置条件 | 进入笔记管理系统 | | |
| 测试步骤 | 1、点击左侧导航栏中的“下载”页面  2、点击“添加”按钮  3、在弹出的添加窗体中编辑并保存相应的笔记内容，然后命名为Test  4、在输入框中输入有效网址，点击“下载”按钮 | | |
| 期望结果 | 1、系统弹出添加成功提示，点击下载后出现下载成功提示  2、下载列表中新增了两条记录 | | |
| 实际结果 | 与期望结果一致 | | |
| 结论 | 笔记数据可成功手动添加，也可成功通过链接下载。 | | |

表19 笔记展示与管理模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 系统 | 网络爬虫的笔记管理系统 | 版本 |  |
| 功能模块名 | 笔记展示与管理模块 | 对应开发人员 | 卢勇 |
| 用例编号 | TestCase2 | 用例设计者 | 卢勇 |
| 测试人员 | 卢勇 | 测试日期 | 2017.05.15 |
| 功能特性 | 展示笔记，修改笔记，删除笔记 | | |
| 测试目的 | 验证笔记能否展示成功，修改成功，删除成功 | | |
| 前置条件 | 进入笔记管理系统 | | |
| 测试步骤 | 1、点击左侧导航栏中的“下载”页面  2、从下载列表中双击某个笔记  3、点击右键，从菜单中点击“修改”  4、点击右键，从菜单中点击“删除” | | |
| 期望结果 | 1、右侧主界面中显示笔记内容  2、笔记内容被成功修改  3、点击删除后，笔记被成功删除 | | |
| 实际结果 | 与期望结果一致 | | |
| 结论 | 笔记可成功展示，笔记可成功修改，笔记可成功删除 | | |

## 5.5 测试结果结论

经过测试，系统的笔记数据获取、笔记展示与管理、笔记归档、用户登录四大模块及其所属的子功能模块都达到了预期功能，并精准地完成了数据的处理和及时的用户反馈。

综上，基于C#的网络爬虫的笔记系统达到了预期的设计目标。

# **结 论**

从2016年12月开始，作者就开始着手为“基于C#的网络爬虫管理系统”做准备，从刚开始的毫无头绪到现在的开发细节烂熟于心，作者也是付出了很多努力。在经过这几个月的不断思考与摸索，作者一次又一次地完善业务，优化细节，在一次又一次的否定与肯定中，最终得到了这个完整的系统。现在，该软件系统的开发设计与论文编写基本完成，从这个毕业设计中，作者学到了很多很多，收获了宝贵的经验和知识。

本文主要对系统的模块划分、结构设计、系统实现、系统测试等几个方面进行了相对详细的描述。详细介绍了系统的设计到系统的实现，最后到系统的完整性测试。对网络爬虫管理系统进行了一个比较详细分析。

在完成该论文后，学到了很多很多东西。本系统主要用C#语言进行开发，在经过多次思考与修改后，对该门语言有了一个更深层次的认识。这次的开发拓展了许多的知识面。特别是在遇到问题的时候，这也对作者的分析解决问题能力有了一个不小的提升。

爬虫笔记管理系统的最初设计的原因正是因为作者没有一个方便的学习平台，也曾多次想开发出这样一个平台，但由于时间原因和其他的一些不可抗因素迟迟没能开始。刚好趁着这次毕业设计，本着方便大众的心态，作者选择了做这个题目。虽然这个系统并不是一个特别复杂的系统，但是，它对.Net开发基础有很高的要求，只有有了较扎实的基础，才能一步一步完成这个系统。这次的毕业设计对作者而言是证明大学结束，但它同样也标志着作者的工作历程刚刚开始，相信这次的开发经历会带给作者很大的帮助，也希望这个系统能帮助到更多的人。

# [**参考文献**](file:///C:\Users\wzlsyy\Desktop\毕业设计参考资料（To%20王梓懿）\AppData\Roaming\Microsoft\Word\论文格式摸板(2007).doc)

[1] 李方园.人机界面设计与应用[M].北京：化学工业出版社，2008

[2] 汪维华，汪维清，胡章平.C#.NET程序设计实用教程[M].北京：清华大学出版社，2012

[3] 武汉厚薄信息技术有限公司，WinForm技术应用[M].北京：清华大学出版社，2014

[4] 耿祥义，张跃平.XML基础教程（第2版）[M].北京：清华大学出版社，2012

[5] 张朝明.XML开发经典应用[M].北京：电子工业出版社，2008

[6] 王石磊.HTML5开发从入门到精通[M].北京：机械工业出版社，2016

[7] 穆拉德·沙巴纳·奥萨拉赫.法）软件架构[M].北京:机械工业出版社，2016

[8] 李伟，吴庆海.软件架构的艺术[M].北京：电子工业出版社，2009

[9] 陈能技，黄志国.软件测试技术大全[M].北京：人民邮电出版社，2016

[10]朱少民，全程软件测试[M].北京：电子工业出版社，2014

[11] 基特(Kit,E.)，软件测试过程改进[M].北京：机械工业出版社，2003

[12] 陈能技，郭柏雅.性能测试诊断分析与优化[M].北京：电子工业出版社，2012

[13] 黄瀚琳.笔记心灵解码学[M].北京：中国经济出版社，2009

# 致 谢

该论文是在成都信息工程大学软件工程学院李代伟老师的悉心指导下完成的。李代伟老师是一位见多识广、拥有丰富经验的教师，具有大量的软件设计知识和丰富的项目开发经验，在整个系统的开发和论文撰写的过程中，李老师对我进行了悉心的指导，也为我提出了许多建设性意见，同时李老师时刻监督着我在整个毕业设计期间，有效地保证了我的毕业设计进度。每当我遇到问题时，李老师都会为我答疑解惑，勉励我大胆创新，在这几个月中，我在李老师的带领下学习到了许多解决问题的方法，也收获了许多宝贵的经验。在此，我向我的指导老师表示最真诚的感谢！

最后，我的毕业设计论文也算到了结尾，非常感谢我的指导老师对我的悉心教导，也感谢学院的苦心栽培，同时感谢学校几年来为我提供了这么好的学习环境和学习条件，希望毕业以后还能有机会为学校作出自己的贡献。

作者简介：

姓 名： 卢勇 性别： 男

出生年月： 1995.05 民族： 汉

E-mail: 411316917@qq.com

# 声 明

本论文的工作是 2016年12月至2017年5月在成都信息工程大学软件工程学院完成的。文中除了特别加以标注地方外，不包含他人已经发表或撰写过的研究成果，也不包含为获得成都信息工程大学或其他教学机构的学位或证书而使用过的材料。

关于学位论文使用权和研究成果知识产权的说明：

本人完全了解成都信息工程大学有关保管使用学位论文的规定，其中包括：

（1）学校有权保管并向有关部门递交学位论文的原件与复印件。

（2）学校可以采用影印、缩印或其他复制方式保存学位论文。

（3）学校可以学术交流为目的复制、赠送和交换学位论文。

（4）学校可允许学位论文被查阅或借阅。

（5）学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

除非另有科研合同和其他法律文书的制约，本论文的科研成果属于成都信息工程大学。

特此声明！

作者签名：

2017年06月 日