成 绩:\_\_\_\_\_\_\_\_\_\_\_\_

教师签名:\_\_\_\_\_\_\_\_\_\_\_\_



**需求分析说明书**

**项目名称:** 登录注册系统

教 师: 王欣

姓 名: 黄昱桥 \_

学 号: 201710253121

班 级: 17gb计算三班

日 期: 2020年07月13日

目录

[一、前言](#_Toc7184) 1

[1.项目前言 1](#_Toc31335)

[2.编写目的 1](#_Toc26921)

[二、可行性研究 1](#_Toc9250)

[1.管理可行性 1](#_Toc31442)

[2.经济可行性 1](#_Toc11303)

[3.技术可行性 1](#_Toc5415)

[三、系统需求分析 2](#_Toc2177)

[1.系统功能概述 2](#_Toc18720)

2.系统功能需求.....................................2

3.需求功能详细介绍.................................2

[四、系统设计](#_Toc176)

1.系统功能流程图..................................3

2.系统设计的原则..................................3

3.业务流程分析....................................3

4.数据库设计与实现................................4

[五、应用程序设计 9](#_Toc8425)

[1.特定模块的详细介绍.............................9](#_Toc14574)

六、附：页面前端程序代码.......................14

1. 登录页面................................14
2. 注册页面....................................15
3. 系统主界面..................................16

七、结束语.......................................17

**一．前言**

**1.项目前言**

**现在计算机已经是非常普遍的东西了，各种网站，各种软件,各种系统，当然每个系统少不了的模块就是登陆注册,这次课设让我明白了asp.net这门编程语言的基本知识和基本方法。希望我能通过这次的课程设计学到更多的知识，学到更多有用的东西**

**2.编写目的**

**（1）通过这次课设掌握开发ASP.NET MVC程序的基础知识和基本方法。**

**（2）让我们对网络程序设计有一个全面的认识和了解。**

**（3）让我们能够独立自主的开发后台管理系统、博客或者商城等网络应用程序。**

**（4）能够掌握微软.net架构基本思想、c#.net语言的基本编程以及动态网页开发的基本技术。**

**二．可行性研究**

**1.管理可行性**

**现在的人们随着网络系统的水平不断提高，在程序设计方面积累了丰富的经验方法，并且简历了成熟的管理机制，齐全的制度。有正确的资料支持，使得现在的网络程序设计有了更好的发展。**

**根据以往程序设计中出现的问题，现在的人们对新程序的开发保持积极态度，很肯定新程序的开发，并且为新程序的开发创造了各种良好的条件，指定了有利于程序实施的管理方法。目前程序设计方面已经能满足登录注册功能的开发以及实施，所以在管理方面上是可行的。**

**2.经济可行性**

**本程序的功能如果上线之后负担的主要费用只有服务器的费用以及后期维护和修改的费用，包括开发费用，虽然有这么多费用但是根本上需要的费用根本不多。**

**本程序的功能有优美的界面、人性化的功能以及功能上线之后产生较少的费用，成本很低，所以经济上也是可行的。**

**3.技术可行性**

**在IT行业中从业的人都必须要熟悉要掌握计算机技术，具备一定的软件硬件基础，会使用从业人员的工作的相应的软件，熟悉IT产品。本程序简单易用，所以在功能上线时只要对客户进行简单的培训，本程序的功能和使用方法就能够顺利运行。所以在技术上是可行的。**

**综合以上三个方面，该功能具有很高的开发可行性，无论是从技术上或者经济上还是管理上。因此，可以设计该系统的数据流程图，建立数据字典。**

**三．系统需求分析**

**1.系统功能概述**

**本程序主要完成的功能有登录功能，登录时实现在页面显示账号密码错误或者登录成功，注册功能，注册时显示注册失败或注册成功，以及登录成功后在页面显示用户用户名功能。**

**2.系统功能需求**

**登录需求：用户进过注册之后可以进行登录若没有注册则不能进行登录。**

**注册需求：用户想要登录的话若没有账号的话需要先进行注册。**

**主页面需求：用户在登录成功之后显示用户的用户名，并显示欢迎登录某某某。**

**3.需求功能详细介绍**

**3.1客户界面部分**

**注册界面**

注册界面需要用户提供用户名和密码，用户名和密码不能为空，而且用户名不能有重复如果有重复的话在底部标签中给出相应的错误信息。

**登陆界面**

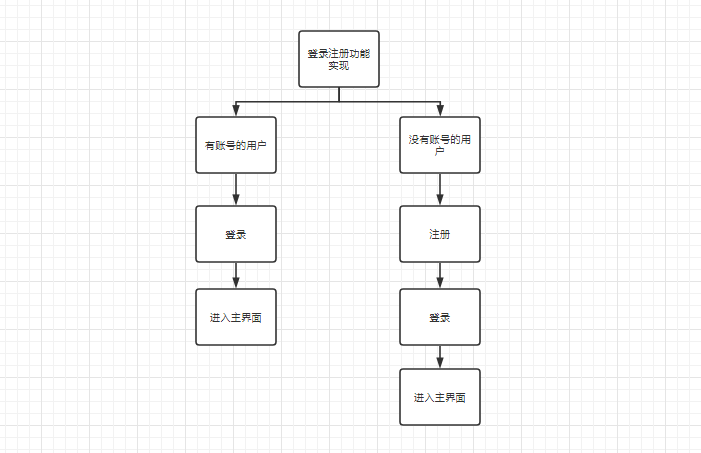
登陆界面验证用户名、密码,用户名和密码不能为空。信息不匹配通过底部标签给出对应错误信息。

**主界面**

主页面及是欢迎界面在用户登录成功之后程序会把数据存入数据库然后进入主页面并显示欢迎登录某某某。

**四．系统设计**

**1. 系统功能流程图**



**2.系统设计的原则**

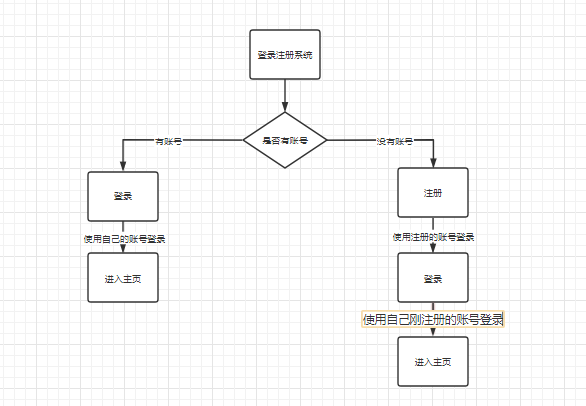
**①易用性原则：本程序面对的大部分用户是没有计算机基础的普通用户，为了更好的展示我们的程序，界面设计需要优美，功能需要简单易操作。**

**②实用性原则：设计出的系统不仅要求满足用户以及管理者的网站登录注册的正常流程还要，把程序设计的能让普通用户能随便上手。**

**3.业务流程分析**

**经过以上的系统的分析我得到了一下的系统业务流程:**

**如图所示**



**4.数据库设计与实现**

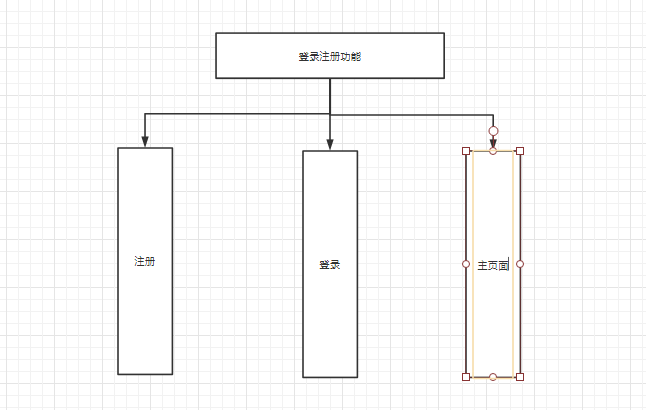
数据库是本程序的核心，系统中所有用户的信息都存在数据库里面，在所有jsp页面中通过Web.confing来连接到数据库，操作数据库中的数据，然后再后端通过查询，插入语句来实现查询和插入功能，从而实现所有功能。数据软件非常的多。本程序中采用的是SqlServer作为数据库工具。数据库是网站的核心，数据库设计的合理与否对网站的建设有着至关重要的影响。

**1）数据库目的与要求**

**包括对本程序的登录，注册，主页面等功能进行数据库设计。**

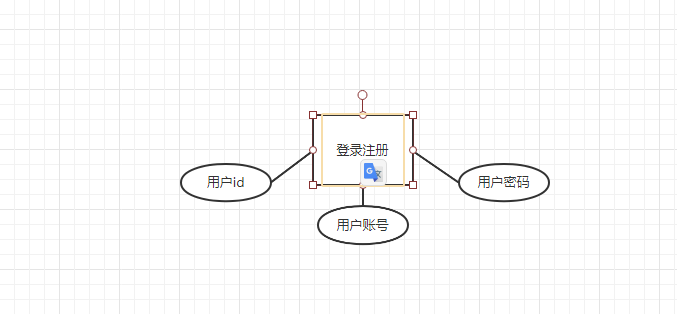
**2）实验内容和步骤**

**①需求分析**



**② 概念设计**

1. **实体及其属性**
2. **本程序E-R图：**



**③ 逻辑设计**

**用户登录表及注册表（用户账号，用户密码）主键为账号，符合三范式。**

**④ 数据库物理实现**

在经过系统的分析了数据库之后，数据库的结构以及非常清晰了，剩下的就是进入SqlServer数据库进行数据库和表的创建，首先在数据库中创建一个数据库名为cmrhyq，然后打开数据库创建一个表名为UserInfo，向表中添加两个字段，第一个是账号字段，第二个是密码字段。下面是数据的属性

1. gongyingshang表。gongyingshang表用来存储供应商信息。

UserInfo表

|  |  |
| --- | --- |
| 字段 | 属性 |
| 账号 | 文本 |
| 密码 | 文本 |

1. 数据库代码

1. **/\***
2. **Navicat Premium Data Transfer**
3. **Source Server : cmrhyq**
4. **Source Server Type : SQL Server**
5. **Source Server Version : 14003281**
6. **Source Host : bj.kevinkda.cn:1433**
7. **Source Catalog : cmr**
8. **Source Schema : dbo**
9. **Target Server Type : SQL Server**
10. **Target Server Version : 14003281**
11. **File Encoding : 65001**
12. **Date: 14/07/2020 13:53:11**
13. **\*/**
14. **-- ----------------------------**
15. **-- Table structure for UserInfo**
16. **-- ----------------------------**
17. **IF EXISTS (SELECT \* FROM sys.all\_objects WHERE object\_id = OBJECT\_ID(N'[dbo].[UserInfo]') AND type IN ('U'))**
18. **DROP TABLE [dbo].[UserInfo]**
19. **GO**
20. **CREATE TABLE [dbo].[UserInfo] (**
21. **[账号] nvarchar(255) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NOT NULL,**
22. **[密码] nvarchar(255) COLLATE SQL\_Latin1\_General\_CP1\_CI\_AS NOT NULL**
23. **)**
24. **GO**
25. **ALTER TABLE [dbo].[UserInfo] SET (LOCK\_ESCALATION = TABLE)**
26. **GO**
27. **-- ----------------------------**
28. **-- Records of UserInfo**
29. **-- ----------------------------**
30. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'123', N'123')**
31. **GO**
32. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'aaa', N'123')**
33. **GO**
34. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'abc', N'abc')**
35. **GO**
36. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'admin', N'a123')**
37. **GO**
38. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'bbb', N'123')**
39. **GO**
40. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'cc', N'')**
41. **GO**
42. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'ccc1', N'cccc')**
43. **GO**
44. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'cmr', N'123')**
45. **GO**
46. **INSERT INTO [dbo].[UserInfo] ([账号], [密码]) VALUES (N'张三', N'123')**
47. **GO**
48. **-- ----------------------------**
49. **-- Primary Key structure for table UserInfo**
50. **-- ----------------------------**
51. **ALTER TABLE [dbo].[UserInfo] ADD CONSTRAINT [PK\_\_UserInfo\_\_1E43ABC2EB730DB9] PRIMARY KEY CLUSTERED ([账号])**
52. **WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON)**
53. **ON [PRIMARY]**
54. **GO**

**五．应用程序设计**

## 1. 特定模块的详细介绍

每个前端和后端模块都有其相应的作用，以下是详细介绍

### (1) 登陆模块

输入用户的账号和密码进行登录，若没有账号密码则点击下方的注册按钮进行注册



输入正确的用户名和密码后进行登陆。



代码：

namespace WebApplication1

public partial class login : System.Web.UI.Page

{

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

{

}

public void Login()

{

String connString = System.Configuration.ConfigurationManager.ConnectionStrings["connStr"].ConnectionString;

SqlConnection conn = new SqlConnection(connString);

conn.Open();

String sql = "select \* from UserInfo where 账号=@useraccount and 密码=@userpassword";

SqlCommand comm = new SqlCommand(sql, conn);

comm.Parameters.Add("useraccount", useraccount.Text);

comm.Parameters.Add("userpassword", userpassword.Text);

SqlDataReader sqlreader = comm.ExecuteReader();

if (sqlreader.Read())

{

Response.Write("<script language=javascript>alert('登录成功！');window.location = 'show.aspx?useraccount="+useraccount.Text+"&userpassword"+userpassword.Text+"';</script>");

}

else

{

connotlogin.Text = "账号或密码错误！";

}

}

protected void Button1\_Click(object sender, EventArgs e)

{

Login();

}

}

}

### (1) 注册模块

用户若没有账号点进注册页面可以进行注册



输入想要注册的账号密码可以进行注册



代码：

using System;

using System.Collections.Generic;

using System.Data.SqlClient;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication1

{

public partial class reg : System.Web.UI.Page

{

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

{

}

public void Reg()

{

string connStr = System.Configuration.ConfigurationManager.ConnectionStrings["connStr"].ConnectionString;

using (SqlConnection conn = new SqlConnection(connStr))

{

using (SqlCommand cmd = new SqlCommand())

{

int count;

conn.Open();

cmd.Connection = conn;

cmd.CommandText = "select count(\*) from UserInfo where 账号=@useraccount";

cmd.Parameters.Add("useraccount", useraccount.Text);

count = (int)cmd.ExecuteScalar();

if (count > 0)

{

connotlogin.Text = "账号已注册！";

}

else

{

cmd.CommandText = "insert into UserInfo(账号,密码) values(@useraccount,@userpassword)";

cmd.Parameters.Add("userpassword", userpassword.Text);

if (cmd.ExecuteNonQuery() > 0)

{

Response.Write("<script language=javascript>alert('注册成功！');window.location = 'login.aspx';</script>");

}

else

{

connotlogin.Text = "注册失败！";

}

}

}

}

}

protected void Button1\_Click(object sender, EventArgs e)

{

Reg();

}

}

}

主页面模块：

用户进过登录之后进入主页面主页面显示当前用户的用户名



代码

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication1

{

public partial class show : System.Web.UI.Page

{

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

{

String useraccount = Request.QueryString["useraccount"];

account.Text = useraccount;

}

}

}

**六．附：页面前端程序代码**

1. **登陆页面**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="login.aspx.cs" Inherits="WebApplication1.login" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title></title>

<style type="text/css">

body {

background: linear-gradient(to right,#47C7AC,#7FD9AD,#C3EFAE,#F1FCB6);

}

#main {

padding-top: 10%;

}

</style>

</head>

<body>

<div id="main" align="center">

<form id="form1" runat="server">

<h1>登录</h1>

<asp:Label ID="Label1" runat="server" Text="用户名："></asp:Label>

<asp:TextBox ID="useraccount" runat="server"></asp:TextBox><br />

<asp:Label ID="Label2" runat="server" Text="密 码："></asp:Label>

<asp:TextBox ID="userpassword" runat="server" TextMode="Password"></asp:TextBox><br />

<asp:Button ID="Button1" runat="server" Text="登录" onclick="Button1\_Click" /><br />

<a href="reg.aspx">注册</a>

</form>

<asp:Label Font-Size="22px" color="red" id="connotlogin" runat="server"></asp:Label>

</div>

</body>

</html>

**2.注册页面**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="reg.aspx.cs" Inherits="WebApplication1.reg" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title></title>

<script src="reg.js"></script>

<style type="text/css">

body {

background: linear-gradient(to right,#47C7AC,#7FD9AD,#C3EFAE,#F1FCB6);

}

#form1{

padding-top:10%;

}

</style>

</head>

<body>

<form id="form1" runat="server" align="center">

<h1>注册</h1>

<asp:Label ID="Label1" runat="server" Text="用户名：" ></asp:Label>

<asp:TextBox ID="useraccount" runat="server"></asp:TextBox><br />

<asp:Label ID="Label2" runat="server" Text="密 码：" ></asp:Label>

<asp:TextBox ID="userpassword" runat="server" TextMode="Password"></asp:TextBox><br />

<asp:Button ID="Button1" runat="server" Text="注册" onclick="Button1\_Click" /><br />

<a href="login.aspx">登录</a>

</form>

<asp:Label Font-Size="22px" color="red" id="connotlogin" runat="server"></asp:Label>

</body>

</html>

**3.系统主界面**

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="show.aspx.cs" Inherits="WebApplication1.show" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title></title>

<style type="text/css">

body {

background: linear-gradient(to right,#47C7AC,#7FD9AD,#C3EFAE,#F1FCB6);

}

div{

color:red;

padding-top:10%;

}

</style>

</head>

<body>

<form id="form1" runat="server">

<div align="center">

欢迎登录:<asp:Label Font-Size="22px" color="red" id="account" runat="server" Text="用户名"></asp:Label>

</div>

</form>

</body>

</html>

**七．结束语**

在大二学期学校给我们开展了ASP.NET这门课程，这个学期我们的任课老师王欣给我们耐心的讲解各个知识点，给我们讲了很多知识，也解答了我们很多的疑惑。这次的课程设计我更深层次的理解了Asp.net这门编程语言。希望我能通过这次的课程设计学到更多的知识，学到更多有用的东西