Design and Development of Sports Enrollment Digital Management Information System Based on ASP.NET

Xue Ding[1, a] Honghong Yang[2, b] Guang Yang[3, c]

[1] School of Enrolment and Job-guidance Center, YunNan Normal University, Kunming 650500, China;

[2] School of the Arts, YunNan Normal University, Kunming 650500, China;

[3] Recruitment and Employment Office, Yunnan Culture and Arts Vocational and Technical College, Kunming 650111, China

[a]dingxue_1983@163.com, [b]bbyhyanghonghong@163.com, [c]634900157@qq.com

Abstract. This paper introduces the design framework, main modules and function of college admission information management system based on ASP. NET, development of this system based on ASP.NET and SQL Server technology

Keywords: ASP.NET; digitization; sports enrollment; enrollment management information system.

1. Introduction

All manuscripts must be in English, also the table and figure texts, otherwise we cannot publish your paper. Please keep a second copy of your manuscript in your office. When receiving the paper, we assume that the corresponding authors grant us the copyright to use the paper for the book or journal in question. Should authors use tables or figures from other Publications, they must ask the corresponding publishers to grant them the right to publish this material in their paper. Use *italic* for emphasizing a word or phrase. Do not use boldface typing or capital letters except for section headings (cf. remarks on section headings, below).

Do not number your paper: All manuscripts must be in English, also the table and figure texts, otherwise we cannot publish your paper. Please keep a second copy of your manuscript in your office. When receiving the paper, we assume that the corresponding authors grant us the copyright to use the paper for the book or journal in question.

With the development of computer technology, information technology and network technology, and with the information modernization of the college management, the original admissions website whose function is relatively simple become more and more unsuitable for the needs of university enrollment, and must be replaced by strong interactive, more powerful recruitment website information management system. Sports enrollment management information system mentioned in this paper mainly for the college, can improve the efficiency of the sports enrollment, make it convenient to communicate with students, and provide great convenience for students.

2. Overall framework of the system

Sports enrollment management information system includes information publishing, information inquiry, fast registration, online consulting, application data information management, data uploading, system maintenance and other functions. Based on the traditional recruitment management information system, online registration, data import and export function is realized so that it can make the enrollment information management function more perfect, forms more diverse, the enrollment and management work more scientific and convenient. Sports enrollment management information system framework as shown in figure 1:
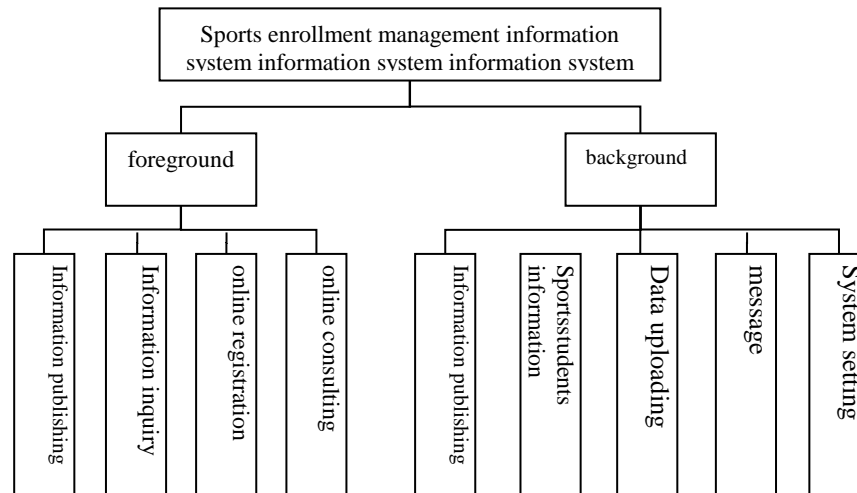
Figure 1 Framework of Sports enrollment management information system

2.1 Information display module

This module is responsible for the information to be displayed on the Webpage. According to the different sections and different page layout, the information in the database will be displayed on the Webpage.

2.2 Information inquiry module

This module mainly introduces that the foreground makes the users query information conveniently, provides college admission information query, sports scores query, and art students' scores query, and college admission inquiry. User only needs to input the name and ID number so that the related information can be found.

2.3 Online registration module

This system mainly provides online registration for the sports examination candidates to ensure the information accuracy and convenience of the examinee registration, but also to facilitate the school admission officers get the information timely, and to shorten the gap between the two groups and make the two interaction more conveniently. The main functions include that sports examination students can register in given time, query information, modify the information and print the registration form.

2.4 Online consultation module

This module is responsible to realize the direct communication between candidates or parents and school personnel The examinee can ask questions about the admissions policy and unkonwn information, while admissions staff can answer, which can improve the rate of entering the school and registration rate.

2.5 Information publishing management

This module mainly publishes various kinds of information, such as the recruitment information, recruitment plans, recruitment policy and college information etc. In the background management, the information can be added, deleted and modified the information, and it also can set up the paper type shown in different modules of foreground.

2.6 Sports students information management

This module main collects sports students' registration information and derives to an Excel file. Besides, it also can set majors, specific part of relative major and athletes' degree.

2.7 Data upload management

This module mainly answers the questions proposed by foreground users, and deletes the similar and boring messages.

2.8 Message management

This module mainly answers the questions proposed by foreground users, and deletes the similar and boring messages.

2.9 System setting management

This module mainly sets up the time of the online registration and information query, and adds and deletes administrator, and modifies the password.

3. Key technology

3.1 Public module design

Based on the above analysis and design of software design, following the principles of standardization, in order to make the program convenient, to improve code maintainability and portability, the CommonClass class is created in this website system, in which class can connect the database, delete, change and check the data. The main codes are as follows:

```
public string errstring = "";
public String constring =
System.Web.Configuration.WebConfigurationManager.ConnectionStrings["ConnectionString"].ToString();

  public DataSet Getdata(string sqlstr)
  {
    SqlConnection conn = new SqlConnection(constring);
    DataSet get = new DataSet();
    try
    {
      SqlDataAdapter adapter = new SqlDataAdapter(sqlstr, conn);
      adapter.Fill(get);
    }
    catch (Exception err)
    {
      errstring = err.Message.Trim();
    }
    finally
    {
      if (conn.State == ConnectionState.Open)
      {
        conn.Close();
      }
    }
    return get;
  }
  public Boolean Updatadata(string sqlstr)
  {
    SqlConnection conn = new SqlConnection(constring);
    SqlCommand ObjCommand = new SqlCommand();
    try
    {
      conn.Open();
      ObjCommand.Connection = conn;
      ObjCommand.CommandText = sqlstr;
      ObjCommand.CommandType = CommandType.Text;
      ObjCommand.ExecuteNonQuery();
      conn.Close();
      return true;
    }
    catch (Exception err)
    {
      errstring = err.Message.Trim();
      conn.Close();
      return false;
    }
}
```

Besides, a public CSS style CommonStyle is also built, and the main codes are as follows:

```
a:link
{
    text-decoration: none;
    color: #000000;
}
a:visited
{
    text-decoration: none;
    color: #000000;
}
a:hover
{
    text-decoration: underline;
    color: #000000;
}

a:active
{
    text-decoration: none;
    color: #000000;
}
.shenlue
{
    table-layout: fixed;
    overflow: hidden;
    word-spacing:normal;
    white-space: nowrap;
    text-decoration:none
}
```

3.2 Message reply management

This module mainly answers the questions proposed by foreground users, and deletes the similar and boring messages. Use DataList to display the message which are not replied and there is a "delete" and a "reply" button behind each message. Click on the "reply" button, the hidden text box displays; then click on the "back" button, the displayed text box is hidden. This is setVisible (bh) function when click on "reply" button, and the setVisible (bh) function codes are as follows:

```
function setVisible(bh)
{
    var style = document.getElementById('tbl1'+ bh).style.display;
    if (style == 'none')
    {
        document.getElementById('tbl1'+ bh).style.display="block"
    }
    else
    {
        document.getElementById('tbl1'+ bh).style.display='none'
    }
}
    Reply button code: <a href="#" onclick="setVisible('<%# Eval("lybh") %>')">reply</a>
```

3.3 Import Excel data

The main function of this module is to upload the recruiting information and scores to the Internet to facilitate the examinees inquiry. When uploading the information, the Excel data is imported to SQL database table directly according to the specified data format. When uploading the information, use FileUpload control to find the Excel, and when click the "import" button, we can use the import_click event to import the data into the database. Detailed codes can be seen in appendix.

Here is the correlated function of Excel operation:

```
    public DataSet ExecleDs(string filenameurl, string table)
{
    string strConn = "Provider=Microsoft.Jet.OleDb.4.0;" + "data source=" + filenameurl + ";Extended Properties='Excel 8.0;
HDR=YES; IMEX=1'";
    OleDbConnection conn = new OleDbConnection(strConn);
    conn.Open();
    DataSet ds = new DataSet();
```

```
        OleDbDataAdapter odda = new OleDbDataAdapter("select * from [Sheet1$]", conn);
        odda.Fill(ds, table);
        return ds;
    }

    protected void import_Click(object sender, EventArgs e)
    {
        if (FileUpload1.HasFile == false)//HasFile is used to check whether FileUpload has particular files       {
            Response.Write("<script>alert('lpease choose Excel file')</script> ");
            return;//when no file exists, please go back
        }
        string IsXls = System.IO.Path.GetExtension(FileUpload1.FileName).ToString().ToLower();//System.IO.Path.GetExtension get the
extension name of the file
        if (IsXls != ".xls")
        {
            Response.Write("<script>alert('only can choose Excel file')</script>");
            return;//when Excel file isn't chosen, please go back
        }
        SqlConnection conn = new SqlConnection(strConn);
        conn.Open();
        string filename = DateTime.Now.ToString("yyyymmddhhMMss") + FileUpload1.FileName;          //get Execle file name
DateTime date function
        string savePath = Server.MapPath(("~\\Uploads\\") + filename);//Server.MapPath get the relative path of virtual server
        FileUpload1.SaveAs(savePath);                    //SaveAs save the uploading file on the server
        DataSet ds = ExecleDs(savePath, filename);         //call custom setup
        DataRow[] dr = ds.Tables[0].Select();          //define a DataRow database
        int rowsnum = ds.Tables[0].Rows.Count;
        if (rowsnum == 0)
        {
            Response.Write("<script>alert('Excel is empty list,no data!')</script>");   //when Excel is empty,prompt users
        else
        {
            for (int i = 0; i < dr.Length; i++)
            {
                string ksh = dr[i]["ksh"].ToString();//date excel [name can't be changed, otherwise something will go wrong]
                string xm = dr[i]["xm"].ToString();//number name as follow
                string sfzh = dr[i]["sfzh"].ToString();
                string fs1 = dr[i]["fs1"].ToString();
                string fs2 = dr[i]["fs2"].ToString();
                string fs3 = dr[i]["fs3"].ToString();
                string fs4 = dr[i]["fs4"].ToString();
                string sqlcheck = "select count(*) from ysscj where ksh='" + ksh + "'And xm='" + xm + "'";  //check whether user exists
                SqlCommand sqlcmd = new SqlCommand(sqlcheck, conn);
                int count = Convert.ToInt32(sqlcmd.ExecuteScalar());
                if (count < 1)
                {
                    string insertstr = "insert into ysscj (ksh,xm,sfzh,fs1,fs2,fs3,fs4) values('" + ksh + "','" + xm + "','" + sfzh + "','" + fs1 + "','" +
fs2 + "','" + fs3 + "','" + fs4 + "')";
                    SqlCommand cmd = new SqlCommand(insertstr, conn);
                    try
                    {
                        cmd.ExecuteNonQuery();
                    }
                    catch (MembershipCreateUserException ex)      //catch exception
                    {
                        Response.Write("<script>alert('import contents:" + ex.Message + "')</script>");
                    }
                }
                else
                {
                    Response.Write("<script>alert('contents are repeating！ No importing ');location='admin_ysscjsc.aspx'</script></script>
");
                    continue;
                }
            }
```

```
      Response.Write("<script>alert('Excle importing succeed!');location='admin_ysscjsc.aspx'</script>");
   }

   conn.Close();
}
```

4. Conclusion

With the development of computer technology, database technology, network technology and multimedia technology, sports recruitment management information system will have more and more powerful function, which will greatly improve the convenience and efficiency of enrollment, and promote the recruitment develop more rapidly and conveniently.

Acknowledgements

References

[1] Liu Xiaoyuan. Realization of enrollment information dynamic website management system based on ASP. Computer Knowledge and Technology, 2009, 5(9): 2056-2057.

[2] Yan Hui, Long Duo. Technology study and realization of ASP.NET in college secondary website development. Journal of Jilin Business and Technology College, 2009, 25 (3): 61-66.

[3] Zhang Heng, Liao Zhifang and Liu Yanli. Network programming design course of ASP.NET. (2009 February Second Edition). Beijing: People's Posts and Telecommunications Press, 2009, 9.

[4] Pan Shiyue. Dynamic web production methods and skills analysis. Computer Knowledge and Technology, 2009, 5 (21): 5652-5653.

[5] Liu Danni. ASP.NET 2.0 (C#) University practical course Beijing: Electronic Industry Press, 2009: 184-194.

[6] Zhang Shuliang, Li Chao. ASP.NET 2.0+SQL Intensive introduction of network application system development. Beijing: Tsinghua University Press, 2007, 116-120.