

# 农药残留检测能力验证结果上报系统的研究与实现<sup>①</sup>

王跃华, 胡梅, 徐亚平, 肖洁

(农业部环境保护科研监测所, 天津 300191)

**摘要:** 农药残留检测能力验证(以下简称: 能力验证)是国家对检测机构实验室资质认定有效性的后续监督检查工作. 针对能力验证结果的管理现状, 采用 ASP.NET + ACCESS 三层架构设计技术实现了一个简易、实用的能力验证结果上报系统. 重点分析了该系统涉及的各类功能模块, 以及开发过程中应用的几项关键技术. 通过实践, 证明使用该上报系统, 显著提高了能力验证工作的效率.

**关键词:** ASP.NET; ACCESS; 能力验证

## Realization of Pesticide Residue Detection Proficiency Testing Report System

WANG Yue-Hua, HU Mei, XU Ya-Ping, XIAO Jie

(Institute of Agro-Environmental Protection, Ministry of Agriculture, Tianjin 300191, China)

**Abstract:** Pesticide residue detection proficiency testing program (hereinafter referred to as proficiency testing) is the follow-up supervision and inspection work carried out to maintain the effectiveness of laboratory detection certification of national mechanism. According to the present situation of the management of proficiency testing, a simple and practical report system of proficiency testing results program is achieved using ASP.NET + ACCESS three layer frame design technology. Sorts of function module involved the system, and the several key skills applied in the development process are analyzed in the essay. The practice has been proved that the report system can improve the efficiency of proficiency testing work.

**Key words:** ASP.NET; ACCESS; proficiency testing

能力验证作为国家对检测机构实验室质量控制的一种重要方法, 其能力验证结果上报关系到一个检测机构实验室的资质认定, 因此能力验证结果上报系统的及时性、准确性、简便性将是系统设计人员必须要解决的问题<sup>[1]</sup>.

B/S 三层结构设计工作方式具有收集信息快、保存数据完整、操作简便快捷等特点. 由于, ASP.NET+ACCESS 编制的动态网页具有可管理、高安全、易部署、扩展性强等特点, 因此本文以数据收集网站设计为基础, 提出一种基于该技术的能力验证结果上报系统设计方案, 从而为今后遇到类似工作提供参考.

## 1 系统设计框架和开发环境

### 1.1 系统框架模型

系统采用 B/S 三层系统架构, 三层结构分别为:

表示层、业务逻辑层和数据访问层<sup>[2]</sup>, 如图 1 所示.

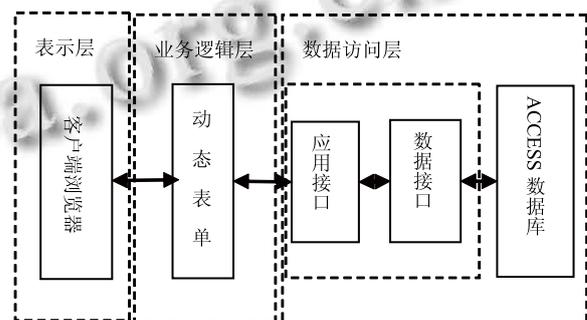


图 1 系统框架模型

表示层通过浏览器与用户进行信息交换, 接收用户的输入并将服务器端传来的数据呈现在客户浏览器上. 业务逻辑层负责接收浏览器传来的请求信息并将请求传给数据访问层, 同时将请求处理结果发给客户

<sup>①</sup> 基金项目: 中央级公益性科研院所基本科研业务费专项(2013-szjj-wyh-09)

收稿时间: 2013-03-13; 收到修改稿时间: 2013-04-09

端浏览器. 数据访问层当中应用接口和数据接口, 负责动态表单与数据库的连接通道的建立, 最终通过 ADO.NET 操控 ACCESS 数据库为业务逻辑数据层提供数据服务, 完成存储能力验证结果、返回数据检索结果等动作<sup>[3]</sup>.

### 1.2 系统开发环境

开发环境是指在基本硬件和数字软件的基础上, 为支持系统软件和应用软件的工程化开发和维护而使用的一组软件. 本系统以 Windows7+IIS6.0+ ASP.NET +ACCESS 搭建开发环境. 另外, 还需要安装 Visual Studio 2010 和 Dreamweaver 8 作为代码编写工具和网页辅助设计软件. 其中 Visual Studio 主要负责系统的代码编写、调试、编译等工作, Dreamweaver 则负责系统界面设计与布局<sup>[4]</sup>.

## 2 系统设计

### 2.1 功能模块设计

通过对能力验证结果上报系统工作流程及需求的分析, 确定了系统主要包括高级管理和普通管理两大类. 高级管理者的功能主要是系统资源初始化、数据整理、数据检索等, 系统设计应做到操作简便, 安全可靠. 普通管理者的功能则以能力验证结果上报为主, 系统设计应以及时性、准确性、简便性为前提, 同时兼顾界面美观友好、系统稳定可靠, 最终达到提高工作效率的目的.

满足上述需求的系统主要包括以下几个模块:

#### 2.1.1 用户登录模块

主要完成系统登录, 根据用户类型, 登录不同控制界面.

#### 2.1.2 系统初始化模块

完成系统各类初始化信息的添加、修改和删除功能, 其包括用户管理、检测方法管理和检测仪器管理等.

#### 2.1.3 数据上报模块

完成能力验证结果数据的上报、编辑、删除和数据分析功能, 其包括检测数据上报、检出率运行、回收率运行、检测仪器选择和检测方法选择等.

#### 2.1.4 数据检索模块

完成上报数据检索查询功能, 可以通过检测方法、检测仪器、项目编号和完成时间等检索信息.

#### 2.1.5 数据管理模块

完成上报数据的备份和导出功能, 高级管理者可

以对一次或多次能力验证结果的数据库进行数据库备份, 并根据需要导出 EXCEL 格式的文件, 该模块使系统的数据库的信息安全得到了有效的保证.

系统结构, 如图 2 所示.

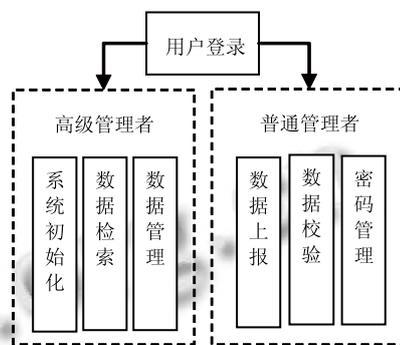


图 2 系统结构图

### 2.2 数据库设计

在对系统需求分析的基础上, 根据数据库设计原则, 重点考虑了以下设计因素<sup>[5]</sup>.

- (1) 结构层次与布局的合理性;
- (2) 数据的结构化、标准化、规范化是否符合国际标准;
- (3) 数据的一致性、安全性.

根据以上原则, 建立了“CcDataSb”数据库, 其中包括用户表、检测方法表、检测仪器表、液相及质谱类表、有机磷类表和有机氯及拟除虫菊酯类表, 下面仅对用户表、检测方法表、检测仪器表、液相及质谱类表加以说明, 如表 1-表 4 所示.

表 1 用户表

字段名	数据类型	数据长度	标示
YH_id	Int	6	Y
YH_name	Character	56	N
YH_password	Character	16	N
YH_type	Character	32	N

表 2 检测方法表

字段名	数据类型	数据长度	标示
JCFF_id	Int	6	Y
JCFF_name	Character	56	N

表 3 检测仪器表

字段名	数据类型	数据长度	标示
JCYQ_id	Int	6	Y
JCYQ_name	Character	56	N

表 4 液相及质谱类表

字段名	数据类型	数据长度	标示
YXZP_id	Int	6	Y
YH_name	Character	56	N
YXZP_time	Datetime	8	N
YXZP_jcff_1	Character	56	N
YXZP_jcyq_1	Character	56	N
YXZP_data_1_1	Float	10	N
YXZP_data_1_2	Float	10	N
YXZP_data_1_3	Float	10	N
YXZP_relevanceRatio_1	Float	10	N
YXZP_recoverRate_1	Float	10	N
YXZP_jcff_...	Character	56	N
YXZP_jcyq_...	Character	56	N
YXZP_data_..._1	Float	10	N
YXZP_data_..._2	Float	10	N
YXZP_data_..._3	Float	10	N
YXZP_relevanceRatio_...	Float	10	N
YXZP_recoverRate_...	Float	10	N
YXZP_jcff_18	Character	56	N
YXZP_jcyq_18	Character	56	N
YXZP_data_18_1	Float	10	N
YXZP_data_18_2	Float	10	N
YXZP_data_18_3	Float	10	N
YXZP_relevanceRatio_18	Float	10	N
YXZP_recoverRate_18	Float	10	N

### 3 关键技术的研究与实现

#### 3.1 数据链接实现

为了增加代码的重用性和系统部署的简易性,使用 Web.Config 文件配额数据库连接,链接字符串采用相对地址,因此不需要修改任何程序代码,就可以快速完成系统部署<sup>[6,7]</sup>. 主要描述如下:

```
<connectionStrings>
  <add name="datasb"
connectionString="Provider=Microsoft.ACE.OLEDB.12.0;Data Source=|DataDirectory|\CcDataSb.accdb"/>
  <!--建立对 ACCESS 数据库的连接并选择要作用的数据库 -->
</connectionStrings>
```

#### 3.2 系统初始化模块的实现

系统初始化模块是整个系统运行的基础部分,主要实现对用户类别、检测方法和检测仪器的信息管理。

用户类别完成高级和普通管理者不同权限身份的登录。根据数据统计分析需要,普通管理者帐号命名时应以样品编号为准,如样品编号为“水稻-001”,其在系统初始化时新建普通管理者,帐号名也应是“水稻-001”,以保证数据上报的一致性。检测方法和检测仪器的主要是将能力验证所用到的检测方法和检测仪器信息填入系统。根据不同能力验证项目的不同,系统所用到的检测方法和检测仪器都有所变化,因此在每次使用系统时,都应首先做好系统初始化工作。

#### 3.3 信息上报模块的实现

信息上报是整能力验证系统中最重要功能,它是系统的核心,它可以实现结果数据的上报、编辑、删除和数据分析的功能。

以下以结果数据上报为例介绍模块实现过程。

结果上报可分为以下 5 个步骤:

(1)用户登录;(2)检测类选择;(3)数据填报;(4)数据校验;(5)数据确认。

在 5 个步骤中如何顺利完成数据填报是关键所在,其部分代码如下:

```
.....
// 普通用户数据填报
{string sql = string.Format("insert into
YeXiangType([UserName], [YP_Id], [TbTime], [JcFf_1],
[JcYq_1], [Data_1_1], [Data_1_2], [Data_1_3], [Jcx_1],
[Hsl_1], ...) values (" + Session["user"] + "," +
Session["user"] + "," + this.Texttime.Text + "," +
this.JCFF_1.Text + "," + this.JCYQ_1.Text + "," +
this.data_1_1.Text + "," + this.data_1_2.Text + "," +
this.data_1_3.Text + "," + this.relevanceRatio_1.Text +
"," + this.recoverRate_1.Text + "," + "...");
// 读取普通用户登录帐号
string jssql = string.Format("SELECT *
FROM [YeXiangType] WHERE ([YP_Id] =" +
Session["user"] + ")");
OleDbConnection con
= new OleDbConnection(ConfigurationManager.
ConnectionStrings["datasb"].ConnectionString);
OleDbCommand com = con.CreateCommand();
OleDbCommand jscom =
con.CreateCommand();连接数据库
com.CommandText = sql;
```

```

jscom.CommandText = jssql;
OleDbDataAdapter myDataAdapter = new
OleDbDataAdapter();
myDataAdapter.SelectCommand = new
OleDbCommand(jssql, con);
OleDbCommandBuilder custCB = new
OleDbCommandBuilder(myDataAdapter);
con.Open();打开数据库
DataSet custDS = new DataSet();
myDataAdapter.Fill(custDS);
int result = 0;
if (this.Texttime.Text == "")
{ this.Response.Write("<script type='text/
javascript' language='javascript'>alert('检测时间不能为
空')</script>"); }
else
{
//判断编号是否重复
if (custDS.Tables[0].Rows.Count == 0)
{
result = com.ExecuteNonQuery();
}
else
{ this.Response.Write("<script
type='text/javascript' language='javascript'>alert('样品编
号不能重复!')</script>"); }
con.Close();
if (result == 1)
{
this.Response.Write("<script
type='text/javascript' language='javascript'>alert('数据插
入成功, 请查看数据')</script>");
}
}
}
}

```

### 3.4 系统安全设计

在 Web 中的 Session 指的是用户在浏览某个网站时, 从进入网站到浏览器关闭所经过的这段时间, 它提供了一种在所有的页面中保存信息的方法。即提供了在 ASP.NET 脚本中定义全局变量的方法, 使得这个全局变量在同一个 Session 中对于所有的 ASP.NET

脚本都有效。因此在系统登录时, 为防止非法用户绕过登录界面进入系统各信息模块, 本系统使用了 Session 来实现用户身份的验证功能<sup>[8,9]</sup>。当用户输入正确的用户类别、用户名和密码并确认后才可登录相应权限的管理页面, 若验证失败则会弹出“用户名或密码错误!”提示窗。

用户登录处理程序 login.aspx.cs 关键代码:

```

.....
{string sql = string.Format("SELECT * FROM
[loginUser] WHERE [UserName] =" +
this.username.Text + " AND [Pwd] =" +this.pwd.Text
+" AND [nameType] =" +this.nameType.Text + "");
OleDbConnection con = new
OleDbConnection(ConfigurationManager.ConnectionStrings[
"databs"].ConnectionString);
OleDbCommand com =
con.CreateCommand();
com.CommandText = sql;
con.Open();打开数据库
try
{
//判断从登录页面读取的用户类型、用户名和密码
与数据库中的是否一致
OleDbDataReader result =
com.ExecuteReader();
if (result.Read() == true)
{
//所有的页面存储的用户名和用户类型
Session["user"] = this.username.Text;
Session["usertype"] =
this.nameType.Text;
//根据用户类型进入不同管理界面
if (this.nameType.Text == "普通用户")
{
Response.Redirect("index.aspx");'普通用户登录界面
}
else
{ Response.Redirect("indexGLY.aspx"); } 管理员登录
界面
}
//登录失败弹出提示窗

```

```

else
    {
        this.Response.Write("<script
type='text/javascript' language='javascript'>alert('用户或
密码错误!)</script>");
    }
finally
{
    con.Close();
}
}

```

除了在登录中储存会话变量,还必须为其他模块中设置一段注册判断代码,将非法的编辑页面访问行为全部转到“错误界面”中去,从而可以拦截非法登录信息管理系统的行为。

.....

```

{ string a = (string)Session["usertype"];
string b = (string)Session["user"];
if (a != "" or b != "")
{ Response.Redirect("error.htm"); }
}

```

#### 4 结语

本文论述了基于 ASP.NET+ ACCESS 技术的农药残留检测能力验证结果上报系统的设计与实现。本

系统结构设计合理,经测试,系统运行稳定,并实现了原定的及时、准确、简便、可靠的设计目标,同时,为今后实现其他类型数据结果上报预留了改进空间,具有良好的扩展性。

#### 参考文献

- 1 王璐,刘潇威,罗铭,彭祎.农产品中农药残留检测能力验证现状与展望.农产品质量与安全,2012,4:41-43.
- 2 张伟罡.基于 ASP.NET 技术的学校网络办公系统的设计与实现.计算机应用与软件,2012,29(11):243-247.
- 3 史建江,李世银,黄兴,顾军.基于 ASP.NET 的信息管理系统设计与实现.微计算机信息,2008,24(2-3):32-33,54.
- 4 王凤玲.基于 PHP+MYSQL 的新闻发布系统的研究与实现.计算机应用与软件,2012,29(2):234-236.
- 5 万敏,蒋智强.Access 数据库中操作查询探析.电脑知识与技术,2012,8(32):7621-7643.
- 6 夏阳,张强,陈小林.基于 ASP.Net 的电子商务网站开发与设计.计算机工程与设计,2004,25(11):2027-2029.
- 7 洪石丹等.ASP.NET 范例开发大全.北京:清华大学出版社,2010:5-7.
- 8 赵强,张红忠.基于 ASP.NET 的网站系统安全性设计与实现.计算机应用,2008,28:271-273,279.
- 9 卢旭,程良伦.ASP 和 ASP.NET 共享 Session 状态研究.计算机应用与软件,2009,26(6):54-56.

(上接第 24 页)

- generation stateful packet filter.Sys Admin,2001,10(12):8-16.
- 6 王一平,韦卫.网络安全框架 Netfilter 在 Linux 中的实现.计算机工程与设计,2006,27(3):439-442.
- 7 章文嵩.可伸缩网络服务的研究与实现[博士学位论文].长沙:国防科技大学,2000.
- 8 苏命峰.三种 LVS 负载均衡模式及性能研究.自动化与信

- 息工程,2012,32(6):17-21.
- 9 Simpson W. RFC 1853,IP in IP Tunneling. Network Working Group,1995.
- 10 IPVS FULLNAT and SYNPROXY. [http://kb.linuxvirtualserver.org/wiki/IPVS\\_FULLNAT\\_and\\_SYNPROXY](http://kb.linuxvirtualserver.org/wiki/IPVS_FULLNAT_and_SYNPROXY).
- 11 IxLoad.[http://www.ixiacom.cn/products/applications/ix\\_load](http://www.ixiacom.cn/products/applications/ix_load).