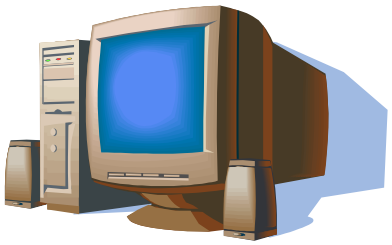
iNautix technologies India Private Limited

"STATEMENTS DATABASE"



- ***A PROJECT REPORT***

(Intended for the Development and Software Quality Assurance teams at iNautix Technologies Chennai- Tidel Park, and Program Management personnel at Pershing US)

by

MOHIDEEN ABDUL KHADER
Reg No. 8011781

G.S.SATISH KUMAR
Reg No. 8011806

DEPARTMENT OF COMPUTER SCIENCE
&
ENGINEERING

ST.JOSEPH'S COLLEGE OF ENGINEERING
CHENNAI – 600119

ACKNOWLEDGEMENT

We express our sincere gratitude to Mr.Veera Raghavan , Mr. Ilayaraja, & Ms. Lakshmi Sudha of iNautix for their guidance throughout our project.

We are very grateful to the staff and members of the Department of Computer Science and Engineering, St.Joseph's College of Engineering, who directly or indirectly helped us in carrying out my project.

We also sincerely thank our parents and friends for the continuous support and encouragement throughout my project.

Above all, We thank almighty God, without whose blessings we would never have been able to complete the work successfully.

ABSTRACT

This project provides functional specifications for development of Statements DB for STF reporting by Program Office. This guide is intended for the Development and Software Quality Assurance teams at iNautix Technologies Chennai and Program Management personnel at Pershing US, responsible for the development of Statements DB.

Presently the Program Office is generating is Statements Measurements Report for the Statements Task Force headed by PTG CIO. The objective of this project is to streamline data collection for this report.

The project effort is targeted to provide the Statements Group with a web-based interface to enter the statements related metrics and allow the Program Office team to extract this information to generate reports. The project design allows PO team members to add new measurement category as required and allows various users to enter the statements related information.

To provide a web-based interface the project set up a web page for facilitating data entry and storage of this information in SQL DB. The functional description will be divided into two sections:

Data Input Level 1- Users should be able to define categories for each of the groups

Data Input level 2- Users should be able to enter information for the defined categories using the web interface.

Although the project is intended for the full functionality of the company there are certain assumptions made in this initial β version of the project.

- No Access Control needs to be controlled
- A total of 6 groups will be defined and will remain fixed. The categories however can be created for each group.

This project is implemented in typical 3-tier architecture with Windows 2000 IIS, ASP(Front end), Visual Basic COM components(Middle Tier) and SQL Server Database (Data Tier).

TABLE OF CONTENTS

1. ABOUT iNAUTIX

iNautix was established in Chennai in 2000 as a development center to provide IT solutions for the financial services industry. The facility is located in Tidel Park, sprawling across 118,000 sq.ft and built to International specifications. The application domain of the company is “BROKERAGE” which deals with, stock, share market, clearing, online banking, wire transfer.

Continuous training and development, both academic and role-based, create a resource pool of talent completely focused on delighting the customer. Direct vendor certifications from Microsoft, Cisco, Oracle, Lotus, IBM, and Sun Microsystems add to the technical depth. A hands-on leadership and metrics-based management give every individual a personal goal to pursue within the framework of the corporate focus.

iNautix is customer-centric and future-driven. The effort is to maintain an independent think tank even as it supports the strategic goals of every customer.

iNautix is an affiliate of Pershing LLC and has a clear vision…"To be the leading provider of mission-critical technology services and infrastructure to institutions seeking to leverage their core business with advanced e-commerce solutions."

2.SOFTWARE REQUIREMENTS

WINDOWS 2000

IIS or PWS To run Asp

SQL Server for SQL Database

IE 5.0

ODBC Tool

3. FUNCTIONAL SPECIFICATION OF THE PROJECT

3.1 About this guide

This guide provides functional specifications for development of Statements DB for STF reporting Program Office.

3.2 Audience

This guide is intended for the Development and Software Quality Assurance teams at iNautix Technologies Chennai and Program Management personnel at Pershing US, responsible for the development of Statements DB.

4. OBJECTIVE OF THE PROJECT

Presently the Program Office is generating is Statements Measurements Report for the Statements Task Force headed by PTG CIO. The objective of this project is to streamline data collection for this report.

5. DETAILED FUNCTIONAL BUSINESS DECRIPTION

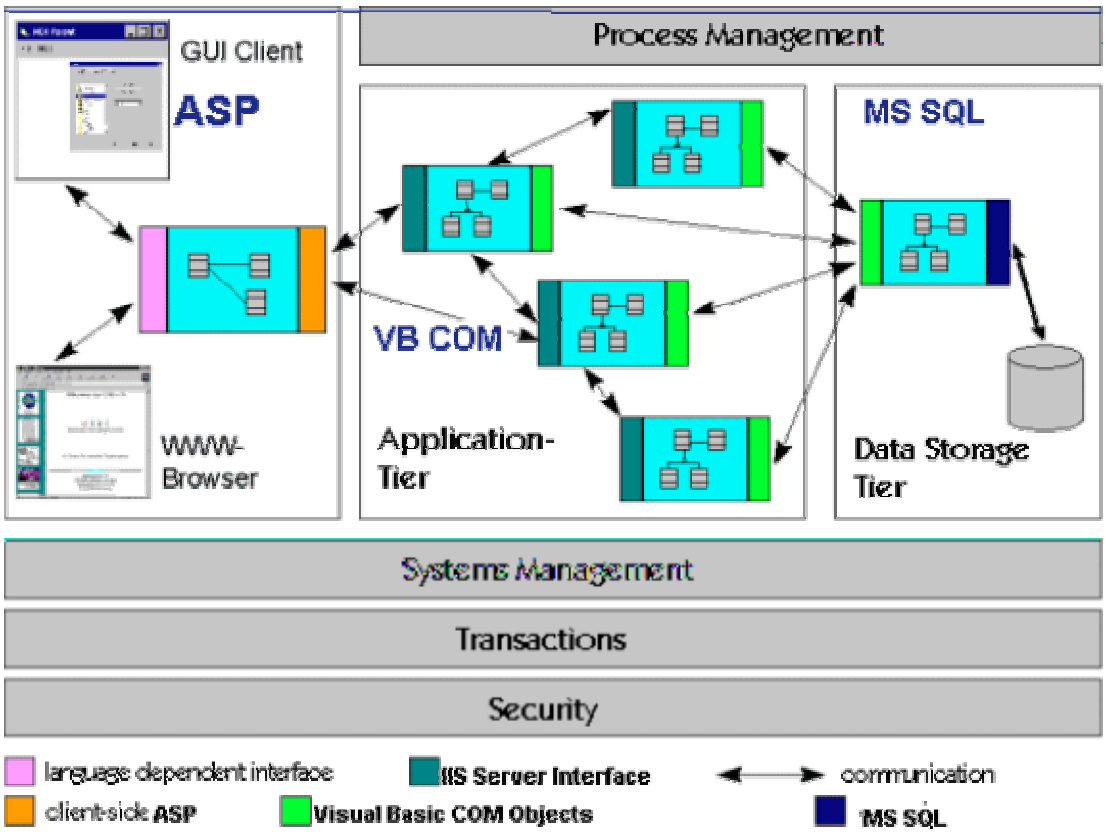
The project effort is targeted to provide the Statements Group with a web-based interface to enter the statements related metrics and allow the Program Office team to extract this information to generate reports. The requirements concern the following main major functions

Allow PO team members to add new measurement category as required
Allow various users to enter the statements related information

The project will require setting up a web page for facilitating data entry and storage of this information in SQL DB. The functional description will be divided into two sections:

Data Input Level 1- Users should be able to define categories for each of the groups
Data Input level 2- Users should be able to enter information for the defined categories using the web interface.

6. 3-TIER APPROACH TO THE PROJECT



This Project is implemented in a typical 3-tier architecture conforming to the international standard with

ASP - Front end

Visual Basic COM Object - Middle Tier

SQL Server Database - Data Tier

Which provides easy facilitation for future enhancements and any modification as suggested by the consignment.

7. **FRONT END DESIGN**

In the front end design we have categorized the users into two levels as level1 and level2.

Level 1 –Category:

1. Each Category will be defined as one data set to be captured.
2. The PO team should be able to define new categories as required.
3. Each Category would have to be assigned to a group on setup. Once created , the category will appear on the group page for which it is setup.
4. No category can be created.

The following groups will be available and fixed:

1. CICS
2. DPG
3. Mt. Prospect
4. QA
5. SCG
6. Others

This category definition screen displays the list of currently defined categories displaying the following details

Measurement No

Category Name

Group

CATEGORY

Group 

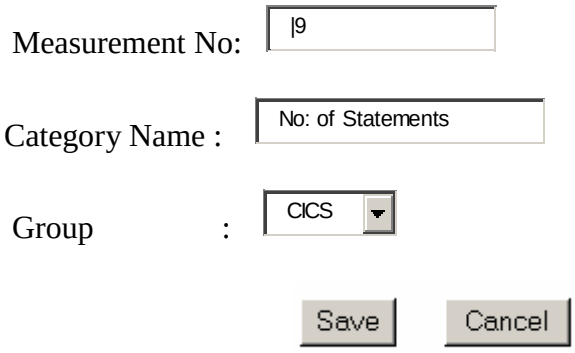
Measurement No | Category Name | Groups

2	No: of Statements Processed	CICS
2212	31313 Geetret	CICS
1231	HGHXAS	CICS
1113	Satish kumar	CICS
121	Seventeen	CICS
4123	Veera	CICS
5634	no of test cycles	CICS
575	kamal test	CICS
5685	test caseCICS	CICS
66	Total accounts processed	stmts DPG
535	no of projects task	SCG
442	no of project completed in DPG	QA

Add

Measurement No is displayed as a link clicking on which the category definition for the selected category should be launched.

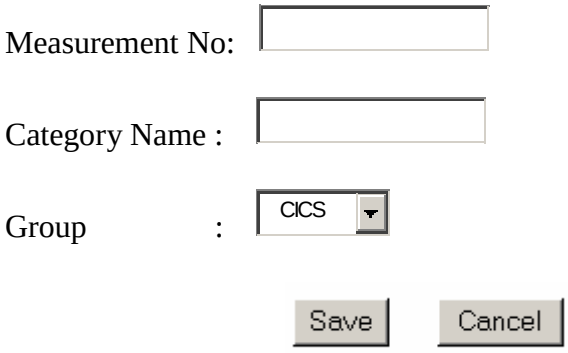
EDIT CATEGORY



Clicking on the Add Button provides an interface to define a new category

---

ADD CATEGORY



Level 2- Measurement Data: This is the data entry level of the measurement data.

1. The data captured for all categories is integer.
2. A save tab is provided at the bottom of the screen

Once the data is saved for some categories on a group page and the page is pulled up again to enter data for the others the existing data should be populated and should be editable.

MEASUREMENT DATA

Group : CICS

Data for CICS

No: of Statements Processed		323232 gertert	
HGHXAS		Satish Kumar	
Satish		terte	
wages		No of defects	
Test		No of Test cycles	
Test Category		Kamal Test	

8. MIDDLE TIER DESIGN

In this Project the middle tier used is the Visual Basic COM components created using DLL files. A single class file is created which contains functions for

- To View all records based on the group name.
- To view a particular record based on the category id.
- To insert details of Level 1 users.
- To edit the existing group details.
- To insert details of Level2 users.
- To view Level2 details based on the category id.
- To view all the details of both Level1 and Level2 as an integrated details.

Certain constraints are placed while creating the Middle Tier using DLL files :

1. Measurement Number should be unique among all the groups.
2. Category Name should be unique within the same groups i.e. different groups may have same category name.
3. Category id is not known to the user.
4. No access control needs to be defined.
5. A total of 6 groups will be defined and will remain fixed. The categories however can be created.

VB CLASS FILE(*used to create DLL file)*

This level uses a class file dll.cls with functions:

- view
- viewcategory
- nsert
- edit
- l2insert
- l2view
- Viewall

DESCRIPTION

Function view:

This function is used to view the main form.
The main form displays the details included in the first table nvm1.
When the user selects a group, the other details are automatically displayed.

Function viewcategory:

This function is used to view a particular record in the group based on the category name selected by the user.

Function nsert:

This function gets the measurement number(Measurement_no) and gets the other details like Categoryname(Category__name)and group name (Group_Name).

The details are then eneterd to the database when the Save button is pressed by the user in the browser..

Function edit:

This function allows any changes to be made with the existing document and updates the same on clicking Save button.

Function l2insert:

This Function checks whether measurement no is unique among all the groups and category name is unique within the group if so the particular value for each of the category is entered into the corresponding group's category name.

Function l2view:

This function displays level2 details when the corresponding group name as selected by the user is found if not returns an error message.

Function ViewAll:

This function displays both the group details such as Measurement no, Group name , Category name, and also the values for each category.

9. DATA TIER DESIGN

This project uses MS SQL Server Database as the Data Tier or Back end. This project basically involves 2 tables

- nvm1
- nvm2

- **nvm 1** has the following attributes :

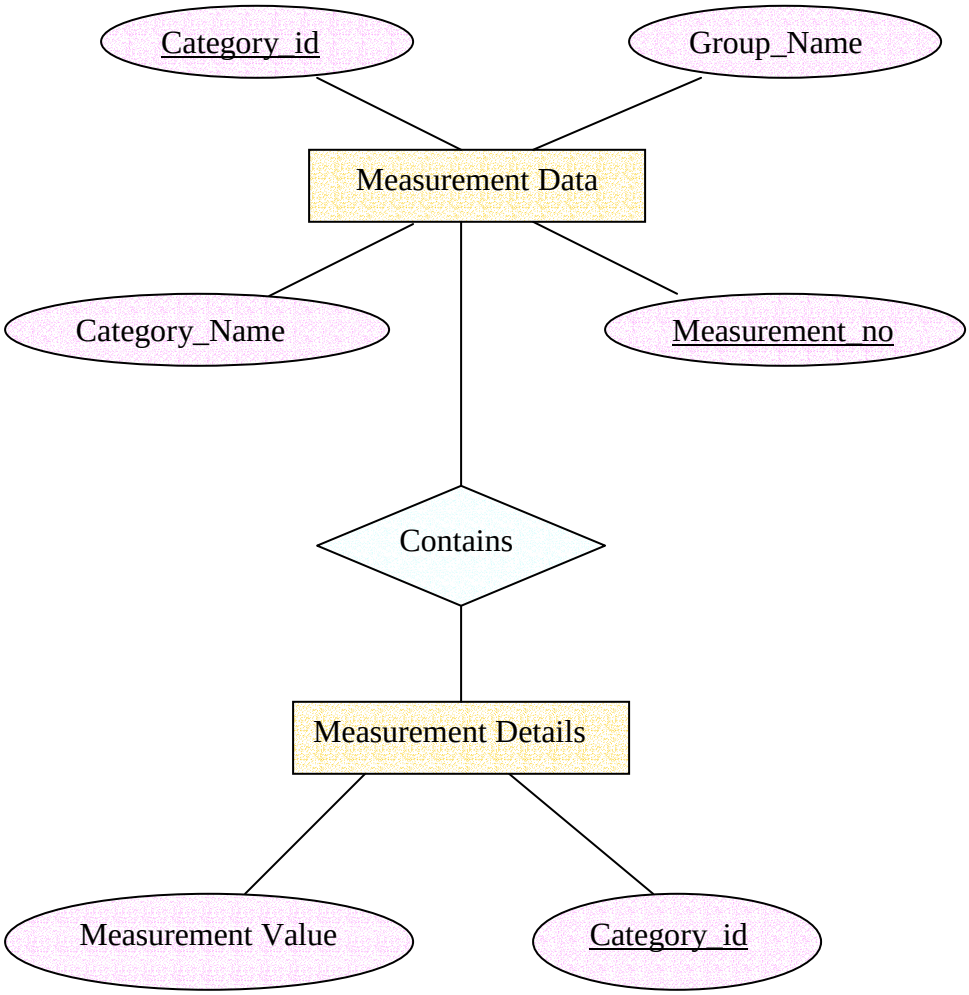
  Category_id as primary key,
  Group_Name
  Measurement no
  Category_name

- **nvm 2** has the following attributes:

  Category_id as foreign key and with nvm 1 as references
  Measurement_ value

While retrieving the Level1 and Level2 details both the tables are Naturally joined(*Outer join)* and queried using the category_id. So for each Category name of a particular group in table nvm1 we get the corresponding measurement value from the table nvm2.

ER DIAGRAM FOR THE DATABASE DESIGNED



10. DEPLOYMENT PROCEDURE

The below files are to be deployed in the root directory of the PWS i.e in the inetpub/wwwroot

1. ***ASP FILES***

- Main.asp - Home Page
- Category.asp - loaded by Main.asp
- EditCategory.asp -Edit page for the Categories
- EditCategorySubmit.asp -Submit form for the EditCategory.asp
- Level2.asp -level2 details
- Level2sub.asp -Submit form for Level2.asp
- ViewAll.asp - To view the entire details of both the Tables(level1 and level2)
- Gif file -Link files for Main page

2.DATABASE FILES

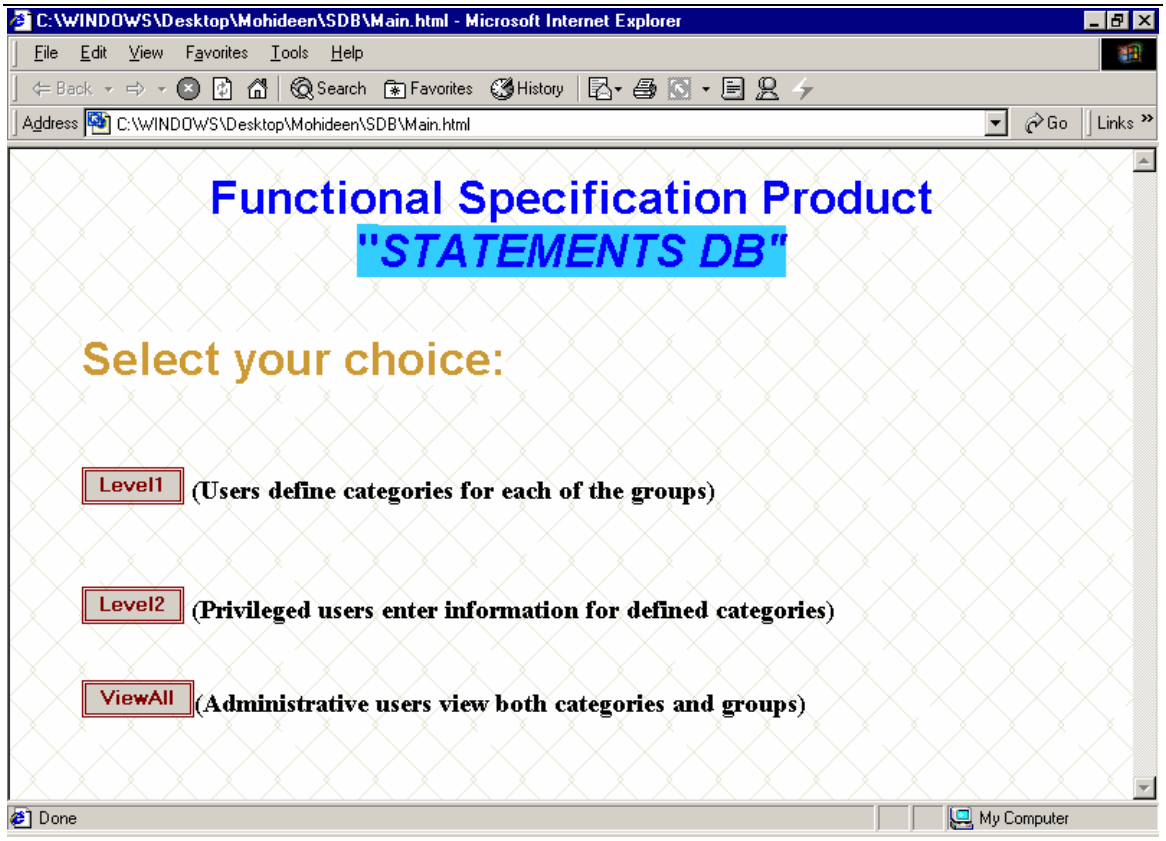
- Nvmproj.vbp -VB project file for the DLL
- NvmClass.cls -VB class file
- Nvmproj.dll -DLL file

Create the DLL file from Nvmproj.vbp for the NvmClass.cls class file.

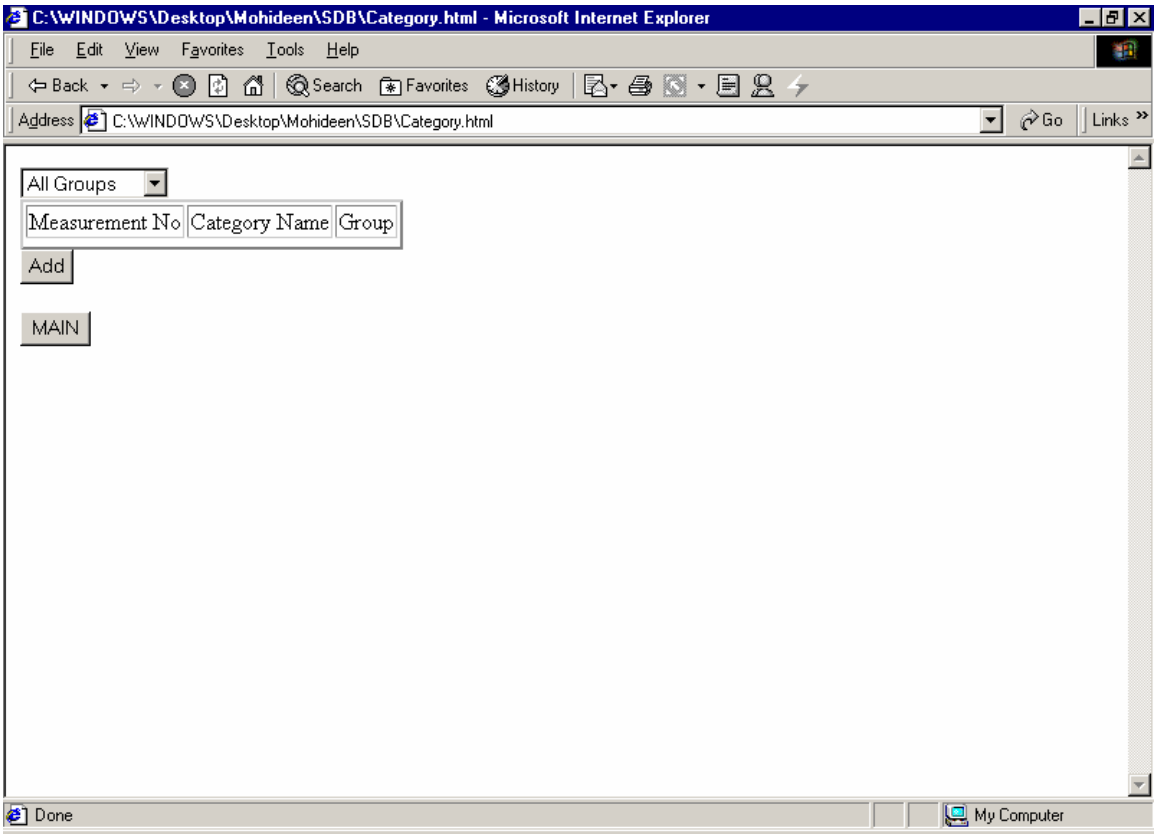
11. SCREEN SHOTS

NOTE : ALL THE SCREEN SHOTS PROVIDED HERE ARE TAKEN WITHOUT PROVIDING ACTUAL DETAILS OF THE COMPANY TO MAINTAIN THE CONFIDENTIAL DETAILS OF THE BROKERAGE COMPANY.

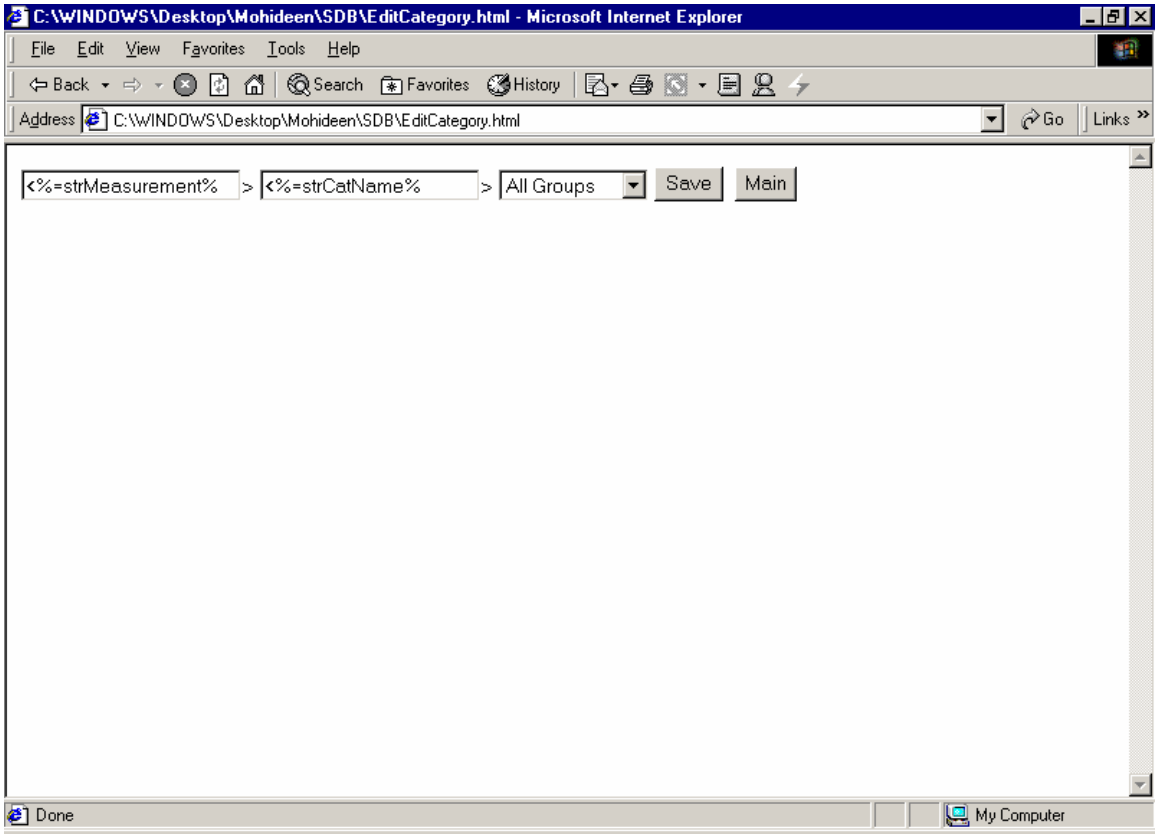
11.1 Main Screen – To select the various options available



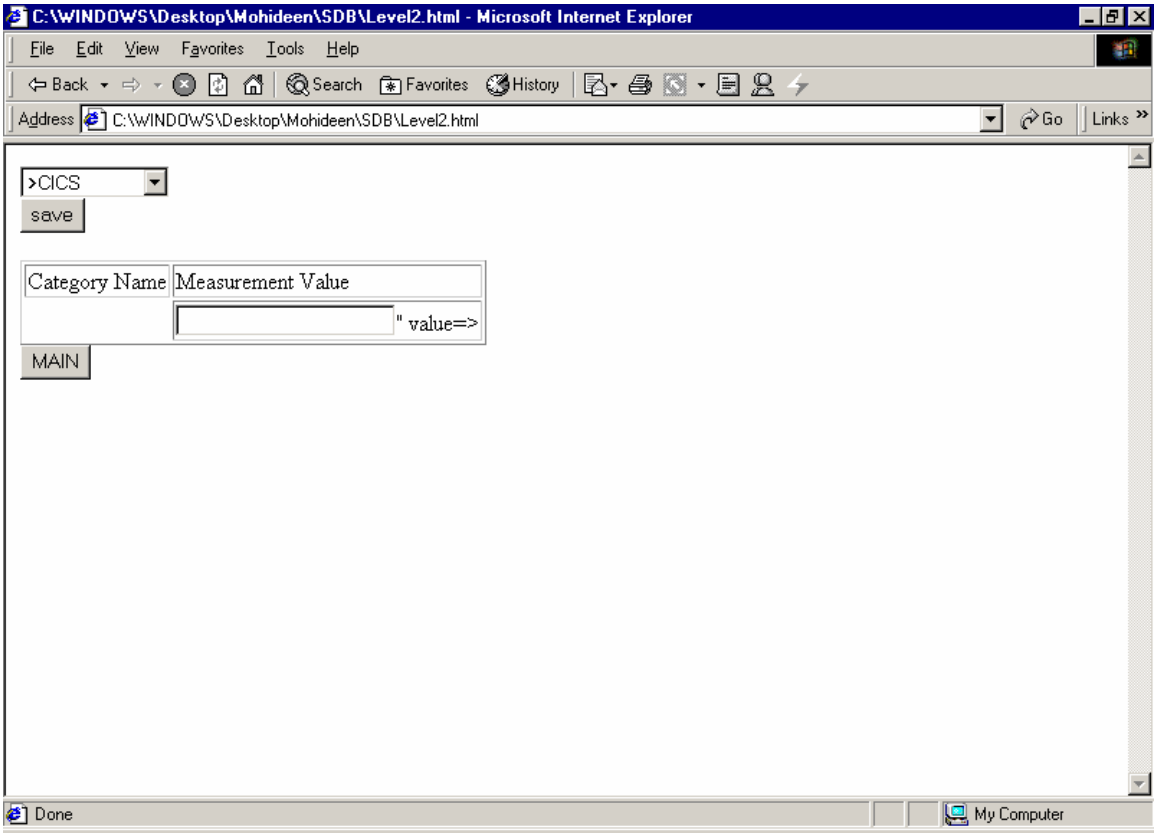
11.2 Category page- to enter the category names for various groups



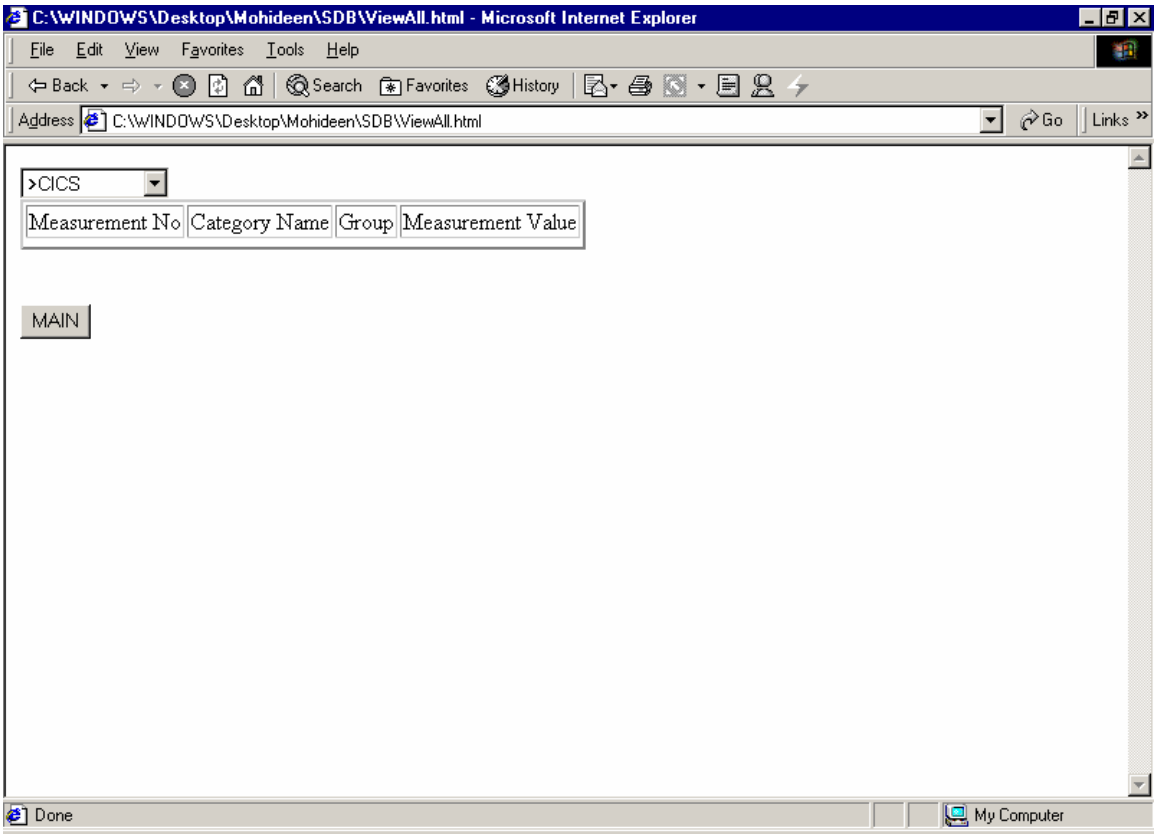
11.3 Edit Category- to edit the existing category of a group



11.4 Level2 page- to enter the measurement value for each category name



11.5 View all page- to view the details of both level1 and level2



12 SAMPLE CODE:

12.1 ASP CODE:

12.1.1 main.asp

```
<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft FrontPage 5.0">
<meta name="Microsoft Theme" content="construc 011">
</HEAD>
<BODY background="urbtextb.gif" bgcolor="#FFFFFF" text="#000000"
link="#990033" vlink="#666666" alink="#CC9933">

<P align=center><strong><font face="Arial" size="6" color="#0000FF">Functional
Specification Product <span style="background-color:
#33CCFF">&amp;quot;</span><i><span style="background-color:
#33CCFF">STATEMENTS
DB&amp;quot;</span></i></font></strong></P>

<P align=left>&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp;&amp;nbsp; <strong>
<span style="background-color: #FFFFFF">
<font face="Arial" size="6" color="#CC9933">Select your
choice:</font></span></strong></P>

<P align=left>&amp;nbsp;</P>
<blockquote>
<h1>
<font size="4">
<INPUT type=button name=level1 value=Level1
onclick="document.location.href='Category.asp'" style="font-weight: bold; color:
#800000; border-style: double; border-width: 3">
(Users define categories for each of the groups)</font></h1>
<p>
&amp;nbsp;</p>
```

```
<h1>
<font size="4">
<INPUT type=button name=level2 value=Level2
onclick="document.location.href='Level2.asp'" style="font-weight: bold; color:
#800000; border: 3px double #800000">
(Privileged users enter information for defined categories)</font></h1>
<h1>
<font size="4">
<br>
<INPUT type=button name=ViewAll value=ViewAll
onclick="document.location.href='ViewAll.asp'" style="color: #800000; font-
weight: bold; border-style: double; border-width: 3">(Administrative
users view both categories and groups) </font>
</h1>
</blockquote>
</BODY>
</HTML>
```

12.1.2 Category.asp

```
<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>

<%
dim objSDB, rsCategories,strGroup
strGroup = Request.Form("Group")
set objSDB=Server.CreateObject("Nvmproj.nvmclass")
set rsCategories = objSDB.view(cstr(strGroup), strErr)
if strErr <> "" then
        Response.Write strErr
    end if


    %>

    <form method="post" action="Category.asp">
    <select name="Group" onchange="document.forms[0].submit()">
        <option value="">All Groups </option>
        <option value="CICS" <%if strGroup = "CICS" then Response.Write
    "selected"%> >CICS </option>
        <option value="DPG" <%if strGroup = "DPG" then Response.Write
    "selected"%>>DPG </option>
        <option value="Mt.Prospect" <%if strGroup = "Mt.Prospect" then
    Response.Write "selected"%>>Mt.Prospect </option>
        <option value="SCG" <%if strGroup = "SCG" then Response.Write
    "selected"%>>SCG </option>
        <option value="Others" <%if strGroup = "Others" then Response.Write
    "selected"%>>Others </option>
    </select>
    <Table border=2>
```

```
    <tr>
        <td> Measurement No </td>
        <td> Category Name </td>
        <td> Group </td>
    </tr>
    <% if not (rsCategories is nothing) then
    do while not rsCategories.eof%>
    <tr>
        <td><a href='<%Response.Write "EditCategory.asp?cat_id=" &
rsCategories("category_id")%>'> <%=rsCategories("measurement_no")%>
</a></td>
        <td><%=rsCategories("category_name")%> </td>
        <td><%=rsCategories("Group_name")%> </td>
    </tr>

    <%rsCategories.movenext
    loop
    end if%>

</table>
<Input type=button name=Add value=Add
onclick="document.location.href='AddCategory.asp'">
<br>
<br>
<Input type=button name=MAIN value=MAIN
onclick="document.location.href='Main.asp'">
</form>
<%set objSDB = nothing%>
</BODY>
</HTML>
```

12.1.3 AddCategory.asp

```
<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>

<%
Dim strCategoryName, strGroup, strMeasurementNo, objadd

   strCategoryName = Request.Form("CategoryName")
    strGroup = Request.Form("Group")
    strMeasurementNo = Request.Form("MeasurementNo")

if (strMeasurementNo = 0)  then

else
set objadd = Server.CreateObject("Nvmproj.nvmclass")
fool        =        objadd.nsert(cint(strMeasurementNo),        cstr(strGroup),
cstr(strCategoryName), strErr)
end if

%>

<Form method="post" action="AddCategory.asp" id=form1 name=form1>
    MEASUREMENT NO
    <input type=text name="MeasurementNo" value=<%=strMeasurement%>>
    CATEGORY NAME
    <input type=text name="CategoryName" value=<%=strCatName%>>
            <select name ="Group">

    <option value="CICS" <%if strGroup = "CICS" then Response.Write
"selected"%> >CICS </option>
    <option value="DPG" <%if strGroup = "DPG" then Response.Write
"selected"%>>DPG </option>
```

```
<option value="Mt.Prospect" <%if strGroup = "Mt.Prospect" then Response.Write "selected"%>>Mt.Prospect </option>
<option value="SCG" <%if strGroup = "SCG" then Response.Write "selected"%>>SCG </option>
<option value="Others" <%if strGroup = "Others" then Response.Write "selected"%>>Others </option>
</select>
        <input type="submit" value="Save" name = "Save">

<Input type=button name=Main value=Main onclick="document.location.href='Main.asp'">

</Form>
</BODY>
</HTML>
```

12.1.4 EditCategory.asp

```
<%@ Transaction=required Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<%
Dim rsCategory, objSample,CatId
CatId = Request.QueryString("Cat_id")
set objSample = Server.CreateObject("Nvmproj.nvmclass")
set rsCategory = objSample.viewcategory(cint(CatId), strErr)
%>
<Form method="post" action="EditcategorySubmit.asp">
   <input type=text name="MeasurementNo" value=<%=strMeasurement%>>
   <input type=text name="CategoryName" value=<%=strCatName%>>
         <select name ="Group">
         <option value="">All Groups </option>
   <option value="CICS" <%if strGroup = "CICS" then Response.Write
"selected"%> >CICS </option>
   <option value="DPG" <%if strGroup = "DPG" then Response.Write
"selected"%>>DPG </option>
   <option value="Mt.Prospect" <%if strGroup = "Mt.Prospect" then
Response.Write "selected"%>>Mt.Prospect </option>
   <option value="SCG" <%if strGroup = "SCG" then Response.Write
"selected"%>>SCG </option>
   <option value="Others" <%if strGroup = "Others" then Response.Write
"selected"%>>Others </option>
   </select>
   <input type="submit" value="Save" name = "Save">
   <input type="hidden" name="hidCatId" value="<%=CatId%>">
<Input type=button name=Main value=Main
onclick="document.location.href='Main.asp'">
</Form>
</BODY>
</HTML>
```

12.1.5 EditCategorySubmi.asp

```
<%@ Language=VBScript %>
<%Response.Buffer = true%>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>

<%
Dim strCategoryName, strGroup, strMeasurementNo, objSdb
    strCategoryName = Request.Form("CategoryName")
    strGroup = Request.Form("Group")
    strMeasurementNo = Request.Form("MeasurementNo")
    strCatId = Request.Form("hidCatId")

    Response.Write strCategoryName & strGroup & strMeasurementNo &
strCatId
    set objSdb = Server.CreateObject("Nvmproj.nvmclass")
    blnSaved = objSdb.edit(cint(strCatId), cint(strMeasurementNo),
cstr(strGroup), cstr(strCategoryName), strErr)


    if (not blnSaved) then
            Response.Write StrErr
    Else
            Response.Redirect "editCategory.asp?cat_id=" & strCatId
    end if


%>

</BODY>
</HTML>
```

12.1.6 Level2.asp

```
<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>
<%
dim objSDB, rsCategories,strGroup
set objSDB=Server.CreateObject("Nvmproj.nvmclass")
strSave = Request.Form("Save")
strGroup = Request.Form("Group")
set rsCategories = objSDB.view(cstr(strGroup), strErr)
if strErr <> "" then
    Response.Write strErr
end if
if strSave="Save" then
    for each objItem in Request.Form
            if (InStr(1,objItem.name,"MeasurementValue")>0) then
                    Response.Write objItem
                            end if
            next
    end if
'z= rsCategories ("category_id")
%>
<form method="post" action="level2.asp" id=form1 name=form1>
<select name="Group" onchange="document.forms[0].submit()">
   <option value="CICS" <%if strGroup = "CICS"then Response.Write
"selected"%> >CICS </option>
   <option value="DPG" <%if strGroup = "DPG" then Response.Write
"selected"%>>DPG </option>
   <option value="Mt.Prospect" <%if strGroup = "Mt.Prospect" then
Response.Write "selected"%>>Mt.Prospect </option>
   <option value="SCG" <%if strGroup = "SCG" then Response.Write
"selected"%>>SCG </option>
```

```
   <option value="Others" <%if strGroup = "Others" then Response.Write
"selected"%>>Others </option>
</select>
<Table border=1>
   <tr>

         <td> Category Name </td>
         <td> Measurement Value </td>
   </tr>
   <% if not (rsCategories is nothing) then
   do while not rsCategories.eof%>
   <tr>

<td><%=rsCategories("category_name")%> </td>
<td>          <input          type=text          name="<%="MeasurementValue"
&rsCategories("category_id") %>" value=<%=strmeasurementvalue%>> </td>
</tr>

   <%rsCategories.movenext
   loop
   end if
   %>
   <input          type=button          name=Save          value          =save
onclick="document.forms[0].action='level2sub.asp';document.forms[0].submit()
">
<br>
<br>

</table>
<Input          type=button          name=MAIN          value=MAIN
onclick="document.location.href='Main.asp'">
</form>

</BODY>
</HTML>
```

12.1.7 Level2sub.asp

```
<%@ Language=VBScript %>
<%Response.Buffer = true%>

<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>

<%dim objSDB, rsCategories,strGroup
set objSDB=Server.CreateObject("Nvmproj.nvmclass")
     a = 1
    for each objItem in Request.Form
            if a = 1 then
             a = 2
              else
                    b = replace(objItem,"MeasurementValue","")
                    c = Request.Form(objItem)
                    rsbool = objSDB.l2insert(cint(b), cint(c))
            end if
    next

if rsbool then
 Response.Redirect "http://localhost/sdb/level2.asp"
else
response.write "jhj"
end if
%>

</form>

</BODY>
</HTML>
```

12.1.8 ViewAll.asp

```
<%@ Language=VBScript %>
<HTML>
<HEAD>
<META NAME="GENERATOR" Content="Microsoft Visual Studio 6.0">
</HEAD>
<BODY>

<%

dim objSDB, rsCategories,strGroup
strGroup = Request.Form("Group")
set objSDB=Server.CreateObject("Nvmproj.nvmclass")
set rsCategories = objSDB.ViewAll(cstr(strGroup))
%>

<form method="post" action="ViewAll.asp" id=form1 name=form1>
<select name="Group" onchange="document.forms[0].submit()">

   <option value="CICS" <%if strGroup = "CICS" then Response.Write
"selected"%> >CICS </option>
   <option value="DPG" <%if strGroup = "DPG" then Response.Write
"selected"%>>DPG </option>
   <option value="Mt.Prospect" <%if strGroup = "Mt.Prospect" then
Response.Write "selected"%>>Mt.Prospect </option>
   <option value="SCG" <%if strGroup = "SCG" then Response.Write
"selected"%>>SCG </option>
   <option value="Others" <%if strGroup = "Others" then Response.Write
"selected"%>>Others </option>
</select>
<Table border=2>
   <tr>
         <td> Measurement No </td>
         <td> Category Name </td>
         <td> Group </td>
         <td> Measurement Value </td>
```

```
    </tr>
    <% if not (rsCategories is nothing) then
    do while not rsCategories.eof%>
    <tr>
            <td><%=rsCategories("measurement_no")%> </td>
            <td><%=rsCategories("category_name")%> </td>
            <td><%=rsCategories("Group_name")%> </td>
            <td><%=rsCategories("Measurement_value")%> </td>
    </tr>

    <%rsCategories.movenext
    loop
    end if%>

</table>
<br>
<br>
<Input type=button name=MAIN value=MAIN
onclick="document.location.href='Main.asp'">
</form>
<%set objSDB = nothing%>
</BODY>
</HTML>
```

12.2 Visual Basic COM Components – NVMProj Modules

```
Dim ad As New ADODB.Connection
Dim rs As New ADODB.Recordset
'Dim rsss As New Recordset
```

```
Public Function view(gp As String, ByRef Strerr As Variant) As Recordset
On Error GoTo l1
ad.ConnectionString = "Provider=SQLOLEDB.1;Password=intern;Persist
Security Info=True;User ID=internuser;Initial Catalog=internship;Data
Source=inatpdrml21s"
ad.Open
If gp <> "" Then
   rs.Open "select * from nvm1 where Group_Name = '" & gp & "'", ad
Else
   rs.Open "select * from nvm1", ad
End If
If rs.EOF = False Then
Set view = rs
Else
Strerr = "no records found"
End If
'rs.Close
Exit Function
l1:
Strerr = Err.Description
End Function
```

```
Public Function viewcategory(catid As Integer, ByRef Strerr As Variant) As
Recordset
On Error GoTo l1
 ad.ConnectionString = "Provider=SQLOLEDB.1;Password=intern;Persist
Security Info=True;User ID=internuser;Initial Catalog=internship;Data
Source=inatpdrml21s"
```

```
ad.close
rs.Open "select * from nvm1 where category_id = " & catid, ad

If rs.EOF = False Then
   Set viewcategory = rs
Else
   Strerr = "no records found"
End If
'rs.Close
Exit Function
l1:
Strerr = Err.Description
End Function
```

```
Public Function nsert(Msno As Integer, Grpnam As String, Category_Name As String, ByRef Strerr As Variant) As Boolean
On Error GoTo l1
   ad.ConnectionString = "Provider=SQLOLEDB.1;Password=intern;Persist Security Info=True;User ID=internuser;Initial Catalog=internship;Data Source=inatpdrml21s"
   ad.Open
   rs.Open "select Category_id from nvm1 where Measurement_No= " & Msno, ad
   If rs.EOF Or rs.BOF Then
      rs.Close
      rs.Open "select Category_id from nvm1 where Group_Name ='" & Grpnam & " 'and Category_Name='" & Category_Name & "'"
      If rs.EOF Or rs.BOF Then
         ad.Execute "insert into nvm1 values (" & Msno & ",'" & Grpnam & "','" & Category_Name & "')"
         nsert = True
      Else
         Strerr = "cat_name already exists"
         nsert = False
         GoTo l2
```

```
      End If
   Else
      Strerr = "The Measurement No. already exists,please enter a new no."
      nsert = False
      'rs.Close
   End If
l2:

   Exit Function
l1:
   Strerr = Err.Description
   nsert = False
End Function
'Public Function del(id As Integer) As String
'On Error GoTo l1
'ad.ConnectionString      =      "Provider=SQLOLEDB.1;Password=intern;Persist
Security    Info=True;User    ID=internuser;Initial    Catalog=internship;Data
Source=inatpdrml21s"
'ad.Open
'rs.Open "select * from nvm1", ad
'ad.Execute "delete from nvm1 where Category_id =" & id
'del = " delete success"
''rs.Close
'Exit Function
'l1:
' Err.Description
'End Function
```

```
Public Function edit(id As Integer, Msno As Integer, Grpnam As String, Catnm
As String, ByRef Strerr As Variant) As Boolean
On Error GoTo l1
ad.ConnectionString      =      "Provider=SQLOLEDB.1;Password=intern;Persist
Security    Info=True;User    ID=internuser;Initial    Catalog=internship;Data
Source=inatpdrml21s"
ad.Open
```

```
rs.Open "select * from nvm1 where category_id=" & id, ad
 If Not rs.EOF Or Not rs.BOF Then
      rs.Close
      rs.Open "select Category_id from nvm1 where Measurement_No= " &
Msno & " and category_id != " & id, ad
      If rs.EOF Or rs.BOF Then
         rs.Close
         rs.Open "select Category_id from nvm1 where Group_Name ='" &
Grpnam & " 'and Category_Name='" & Category_Name & "' and category_id !="
& id, ad
         If rs.EOF Or rs.BOF Then
            ad.Execute "update nvm1 set Category_Name='" & Catnm & "'where
Category_id = " & id
            ad.Execute "update nvm1 set Group_Name='" & Grpnam & "'where
Category_id = " & id
            ad.Execute "update nvm1 set Measurement_No=" & Msno & "where
Category_id = " & id
            edit = True
         Else
            Strerr = "cat_name already exists"
            edit = False
            GoTo l2
         End If
       Else
         Strerr = "The Measurement No. already exists,please enter a new no."
         edit = False
      End If
   Else
      Strerr = "Id does not exist"
      edit = False
      'rs.Close
   End If
l2:
  Exit Function
l1:
  Strerr = Err.Description
```

```
End Function
```

```
Public Function l2insert(catid As Integer, measval As Integer) As Boolean
On Error GoTo l1
   ad.ConnectionString     =     "Provider=SQLOLEDB.1;Password=intern;Persist
Security    Info=True;User    ID=internuser;Initial    Catalog=internship;Data
Source=inatpdrml21s"
   ad.Open

    'rs.Open '"select * from nvm2 where Category_Id = " & catid, ad
     ad.Execute "insert into nvm2 values (" & catid & "," & measval & ")"
          l2insert = True

   Exit Function
l1:
       l2insert = False
End Function
```

```
Public Function l2view(catid As Integer) As Recordset

 On Error GoTo l1
ad.ConnectionString      =      "Provider=SQLOLEDB.1;Password=intern;Persist
Security    Info=True;User    ID=internuser;Initial    Catalog=internship;Data
Source=inatpdrml21s"
ad.Open
If gp <> "" Then
   rs.Open "select * from nvm1 where Group_Name = '" & gp & "'", ad
Else
   rs.Open "select * from nvm1", ad
End If
If rs.EOF = False Then
Set view = rs
Else
Strerr = "no records found"
```

```
End If
'rs.Close
Exit Function
l1:
Strerr = Err.Description
End Function
```

```
Public Function ViewAll(gp As String) As Recordset

ad.ConnectionString = "Provider=SQLOLEDB.1;Password=intern;Persist Security Info=True;User ID=internuser;Initial Catalog=internship;Data Source=inatpdrml21s"
ad.Open
rs.Open " select n1.Category_Id , n1.Measurement_No , n1.Group_Name , n1.Category_Name , n2.Measurement_value from nvm1 n1 , nvm2 n2 where n1.Category_Id = n2.Category_Id and n1.Group_Name='" & gp & "'", ad
Set ViewAll = rs
End Function
```

REFERENCE:

1. www.inautix.com
2. www.w3schools.com
3. www.learnasp.com
4. www.microsoft.com/vb.com
5. www.pershing.com
6. Using HTML4 by Lee Anne Phillips QUE Publications
7. Fundamentals of Database Systems by ElMasri & Navathe Addison Wesley Publications