

# システムの”見える化”支援 サンプル設計書

～ VB.NET ～

**DCR**

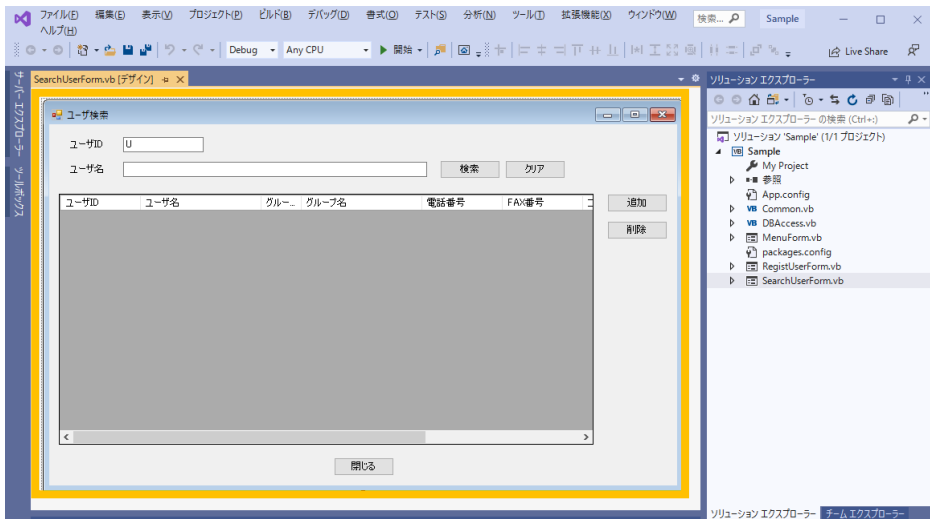
株式会社第一コンピュータリソース

システムの“見える化”支援では、VB.NETから以下を生成いたします。

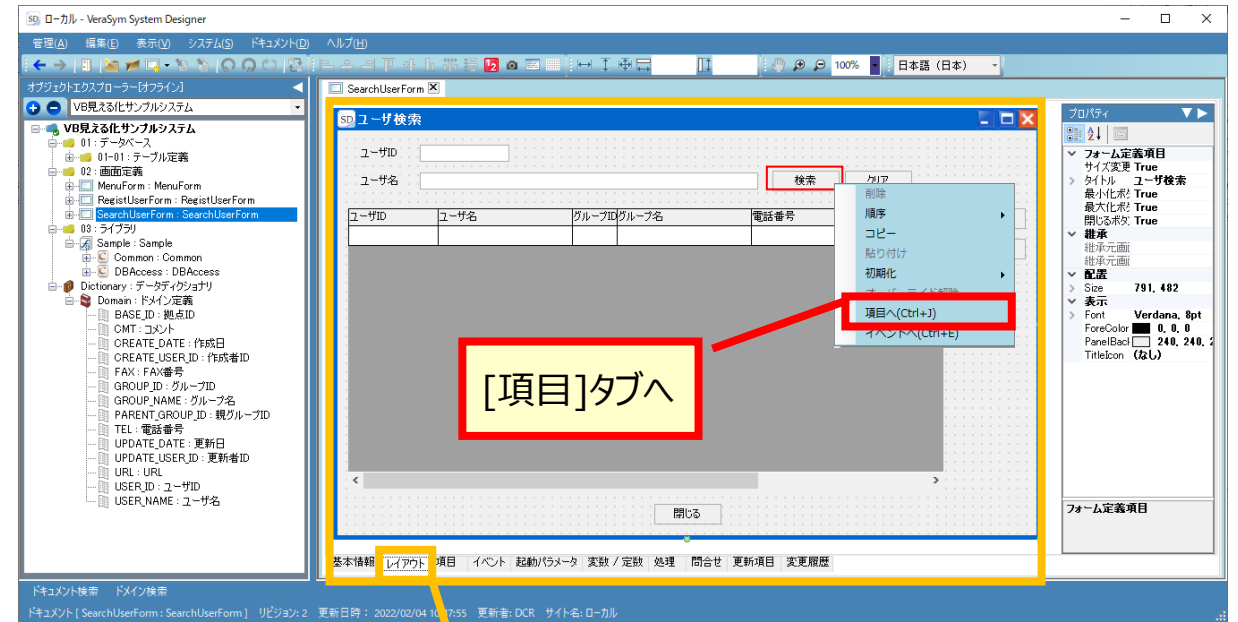
<VB.NET>		<VSSDドキュメント>
レイアウト	⇒	画面定義「レイアウトタブ」
項目	⇒	画面定義「項目タブ」
イベント	⇒	画面定義「イベントタブ」
処理	⇒	画面定義「処理タブ」
問合せ	⇒	画面定義「問合せタブ」

# レイアウト

## VB.NET



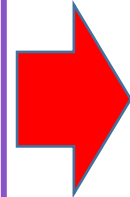
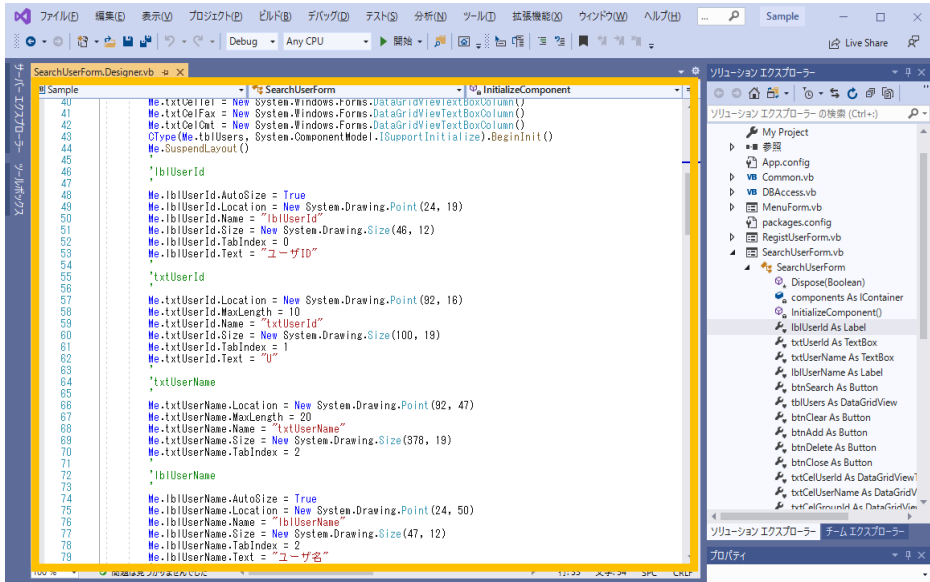
## VSSD



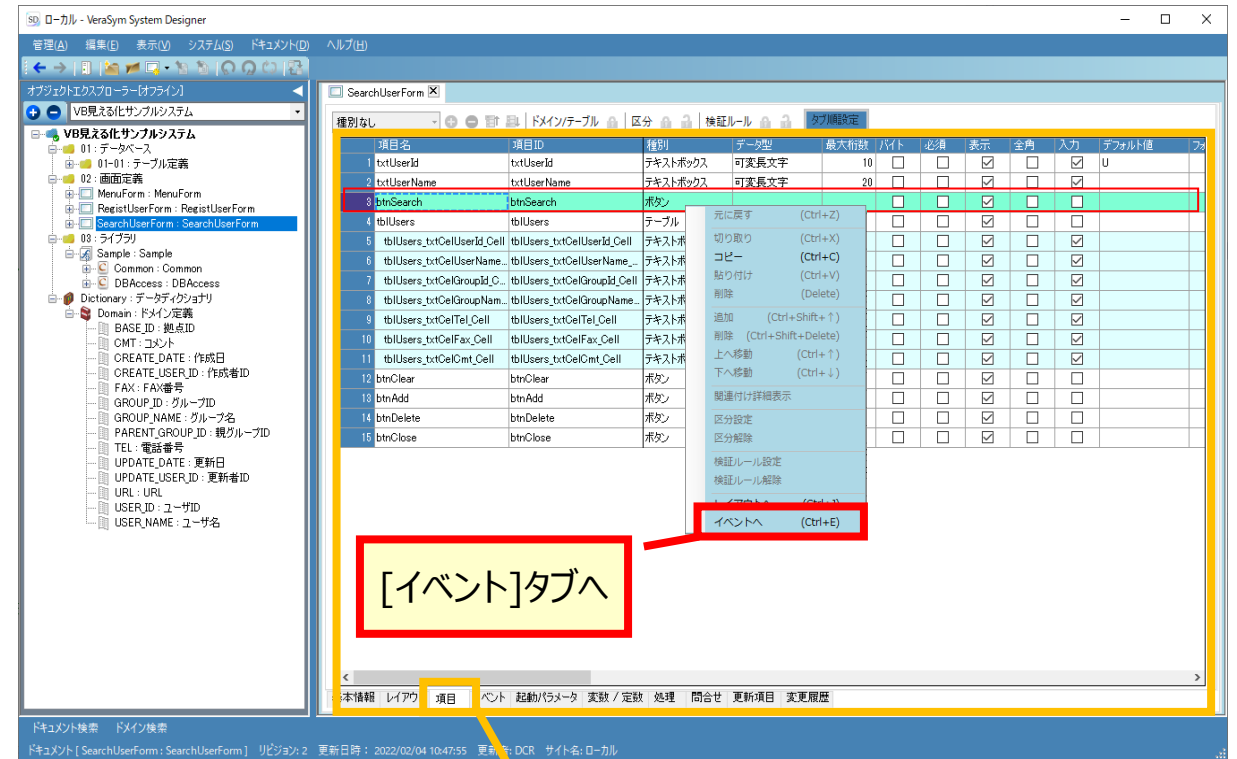
Designerファイルの情報から画面レイアウトを取り込みます。  
VSSDに無いコントロールについては、イメージなど代替のコントロールで表現します。

# 項目

## VB.NET



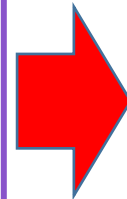
## VSSD



# イベント

## VB.NET

```
24 InitializeComponent()  
25  
26 ' InitializeComponent() 呼び出しの後で初期化を追加します。  
27  
28 End Sub  
29  
30 *** <summary>  
31 検索ボタン押下  
32 *** </summary>  
33 *** <param name="sender"></param>  
34 *** <param name="e"></param>  
35 Private Sub btnSearch_Click(sender As Object, e As EventArgs) Handles btnSearch.Click  
36  
37     ユーザ検索処理を呼び出す  
38     Call SearchUser()  
39  
40 End Sub  
41  
42 *** <summary>  
43 ユーザ検索処理  
44 *** </summary>  
45 Private Sub SearchUser()  
46  
47     変換宣言  
48     Dim dt As DataTable  
49     Dim sql As New StringBuilder  
50     Dim jdx As Integer = 0  
51  
52     入力チェック  
53     If txtUserId.Text = "" Then  
54         ShowErrorMessage("必須入力です。")  
55         txtUserId.Focus()  
56     Exit Sub  
57     End If
```



## VSSD

イベント	処理
btnSearch ボタン クリック	1. SearchUserを実行する
btnClear ボタン クリック	1. <項目>txtUserIdをクリアする 2. <項目>txtUserNameをクリアする 3. <項目>tblUsersのレコレクションを空にする
btnAdd ボタン クリック	1. AddRowを実行する
btnDelete ボタン クリック	1. DeleteUserを実行する
btnClose ボタン クリック	1. Closeを実行する

[処理]タブへ

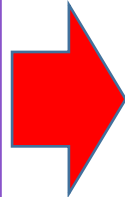
基本情報 | レアウト | 項目 | イベント | 自動リサイズ | 実数 / 定義 | 処理 | 問合せ | 更新項目 | 変更履歴

[イベント]タブでは画面やコントロールのイベントを定義できます。文章中では画面項目や処理などにリンクが設定されます。

# 処理

## VB.NET

```
41 *** Summary ***
42 *** ユーザ検索処理 ***
43 *** /summary ***
44 3 箇の巻頭
45 Private Sub SearchUser()
46
47     ' 変数宣言
48     Dim dt As DataTable
49     Dim sql As New StringBuilder
50     Dim idx As Integer = 0
51
52     ' 入力チェック
53     If txtUserId.Text = "" Then
54         ShowErrorMessage("必須入力です。")
55         txtUserId.Focus()
56         Exit Sub
57     End If
58
59     ' 検索処理
60     Try
61         ' 一覧クリア
62         tblUsers.Rows.Clear()
63
64         ' SQL文字列・パラメータ生成
65         sql.Append("SELECT ")
66         sql.Append(" T1.USER_ID ")
67         sql.Append(" T1.USER_NAME ")
68         sql.Append(" T1.GROUP_ID ")
69         sql.Append(" T2.GROUP_NAME ")
70         sql.Append(" T1.TEL ")
71         sql.Append(" T1.FAX ")
72         sql.Append(" T1.ONT ")
73         sql.Append(" FROM M_USER T1 ")
74         sql.Append(" LEFT OUTER JOIN M_GROUP T2 ")
75         sql.Append(" ON T1.GROUP_ID = T2.GROUP_ID ")
76         sql.Append(" WHERE USER_ID LIKE ")
77         sql.Append(DBAccess.GetSqlStringValue(txtUserId.Text))
78         sql.Append(" || '%' ")
79
80         ' ユーザ名が入力されている場合
81         If txtUserName.Text <> "" Then
82             sql.Append(" AND USER_NAME LIKE '%' || ")
83             sql.Append(DBAccess.GetSqlStringValue(txtUserName.Text))
84             sql.Append(" || '%' ")
85         End If
86
87         ' DBから情報取得
88         dt = GetData(sql.ToString())
89
90         ' 検索結果が0件なら終了
91         If dt.Rows.Count = 0 Then
92             ShowErrorMessage("対象データがありません。")
93             Exit Sub
94         End If
95
96         ' 画面にセット
97         For Each row As DataRow In dt.Rows
98             idx = tblUsers.Rows.Count
99             tblUsers.Rows.Add()
100             tblUsers.Rows(idx).Cells(TBLUSERS_COL_USERID).Value = row.Item("USER_ID")
101             tblUsers.Rows(idx).Cells(TBLUSERS_COL_USERNAME).Value = row.Item("USER_NAME")
102             tblUsers.Rows(idx).Cells(TBLUSERS_COL_GROUPID).Value = row.Item("GROUP_ID")
103             tblUsers.Rows(idx).Cells(TBLUSERS_COL_GROUPNAME).Value = row.Item("GROUP_NAME")
104             tblUsers.Rows(idx).Cells(TBLUSERS_COL_TEL).Value = row.Item("TEL")
105             tblUsers.Rows(idx).Cells(TBLUSERS_COL_FAX).Value = row.Item("FAX")
106             tblUsers.Rows(idx).Cells(TBLUSERS_COL_ONT).Value = row.Item("ONT")
107         Next
108
109     Catch ex As Exception
110         ' 例外処理
111         ShowErrorMessage("例外が発生しました。" + Environment.NewLine + ex.Message)
112     End Try
113     Exit Sub
114 End Try
115 End Sub
```



## VSSD

項目名	項目ID	種別	データ型	クラス	記列	デフォルト値	テーブル	テーブル属性
uid	uid	変数						
sql	sql	変数						
idx	idx	変数	数値					

処理詳細

1. <処理:変数>idxに0をセットする
2. <項目>txtUserIdのテキストと""が等しい場合
  - 2.1. <Common>ShowErrorMessageを実行する  
<Common>ShowErrorMessage(message) ← "必須入力です。"
  - 2.2. <項目>txtUserIdにフォーカスをセットする
  - 2.3. 処理を終了する
3. 以下の処理を実行する
  - 3.1. <項目>tblUsersの行コレクションを空にする
  - 3.2. <項目>txtUserNameのテキストと""が等しい場合
  - 3.3. <処理:変数>dtにSelect SearchUserの1件の戻り値をセットする
  - 3.4. <処理:変数>dtのテーブルの行コレクションの行数と0が等しい場合
    - 3.4.1. <Common>ShowErrorMessageを実行する  
<Common>ShowErrorMessage(message) ← "対象データがありません。"

[問合せ]タブへ

[処理]タブではメソッド単位の処理を記述できます。  
文章中では画面項目や処理などにリンクが設定されます。

# 問合せ

## VB.NET

```
41 *** Summary  
42 *** ユーザー検索処理  
43 *** Summary  
44 *** 関の参照  
45 Private Sub SearchUser()  
46  
47 ' 変数宣言  
48 Dim dt As DataTable  
49 Dim sql As New StringBuilder  
50 Dim idx As Integer = 0  
51  
52 ' 入力チェック  
53 If txtUserId.Text = "" Then  
54 ShowErrorMessage("必須入力です。")  
55 txtUserId.Focus()  
56 Exit Sub  
57 End If  
58  
59 ' 検索処理  
60 Try  
61 ' 一覧クリア  
62 tblUsers.Rows.Clear()  
63  
64 ' SQL文字列・パラメータ生成  
65 sql.Append("SELECT ")  
66 sql.Append(" T1.USER_ID ")  
67 sql.Append(" T1.USER_NAME ")  
68 sql.Append(" T1.GROUP_ID ")  
69 sql.Append(" T2.GROUP_NAME ")  
70 sql.Append(" T1.TEL ")  
71 sql.Append(" T1.FAX ")  
72 sql.Append(" T1.CMT ")  
73 sql.Append(" FROM M_USER T1 ")  
74 sql.Append(" LEFT OUTER JOIN M_GROUP T2 ")  
75 sql.Append(" ON T1.GROUP_ID = T2.GROUP_ID ")  
76 sql.Append(" WHERE USER_ID LIKE ")  
77 sql.Append(DBAccess.GetSqlStringValue(txtUserId.Text))  
78 sql.Append(" || '%' ")  
79  
80 ' ユーザ名が入力されている場合  
81 If txtUserName.Text <> "" Then  
82 sql.Append(" AND USER_NAME LIKE '%' || ")  
83 sql.Append(DBAccess.GetSqlStringValue(txtUserName.Text))  
84 sql.Append(" || '%' ")  
85 End If  
86  
87 ' DBから情報取得  
88 dt = GetData(sql.ToString)  
89  
90 ' 検索結果が0件なら終了  
91 If dt.Rows.Count = 0 Then  
92 ShowErrorMessage("対象データがありません。")  
93 Exit Sub  
94 End If  
95  
96 ' 画面にセット  
97 For Each row As DataRow In dt.Rows  
98 idx = tblUsers.Rows.Count  
99 tblUsers.Rows.Add()  
100 tblUsers.Rows(idx).Cells(TBLUSERS_COL_USERID).Value = row.Item("USER_ID")  
101 tblUsers.Rows(idx).Cells(TBLUSERS_COL_USERNAME).Value = row.Item("USER_NAME")  
102 tblUsers.Rows(idx).Cells(TBLUSERS_COL_GROUPID).Value = row.Item("GROUP_ID")  
103 tblUsers.Rows(idx).Cells(TBLUSERS_COL_GROUPNAME).Value = row.Item("GROUP_NAME")  
104 tblUsers.Rows(idx).Cells(TBLUSERS_COL_TEL).Value = row.Item("TEL")  
105 tblUsers.Rows(idx).Cells(TBLUSERS_COL_FAX).Value = row.Item("FAX")  
106 tblUsers.Rows(idx).Cells(TBLUSERS_COL_CMT).Value = row.Item("CMT")  
107 Next  
108  
109 Catch ex As Exception  
110  
111 ' 例外処理  
112 ShowErrorMessage("例外が発生しました。" + Environment.NewLine + ex.Message)  
113 Exit Sub  
114 End Try  
115  
116 End Sub
```



## VSSD

The screenshot shows the VSSD interface for defining a query. The '問合せ定義' (Query Definition) tab is active, showing a table with columns 'SelectSearchUser', 'SelectSearchUser', and 'SearchUser'. Below the table, the '問合せ定義' (Query Definition) section is expanded, showing the '問合せ定義' (Query Definition) tab. The '問合せ定義' (Query Definition) section includes a '問合せ定義' (Query Definition) table with columns '問合せ定義' (Query Definition), 'パラメータタイプ' (Parameter Type), and '結果タイプ' (Result Type). The '問合せ定義' (Query Definition) table is defined as follows:

問合せ定義	パラメータタイプ	結果タイプ
検索項目		
T1.ユーザID		
T1.ユーザ名		
T1.グループID		
T2.グループ名		
T1.電話番号		
T1.FAX番号		
T1.コメント		

The '問合せ定義' (Query Definition) section also includes a 'テーブル' (Table) section with the following definition:

```
UPDATE_DATE: 更新日  
UPDATE_USER_ID: 更新者ID  
LEFT OUTER JOIN グループマスタ T2  
ON T1.グループID = T2.グループID  
URL: URL  
USER_ID: ユーザID  
USER_NAME: ユーザ名
```

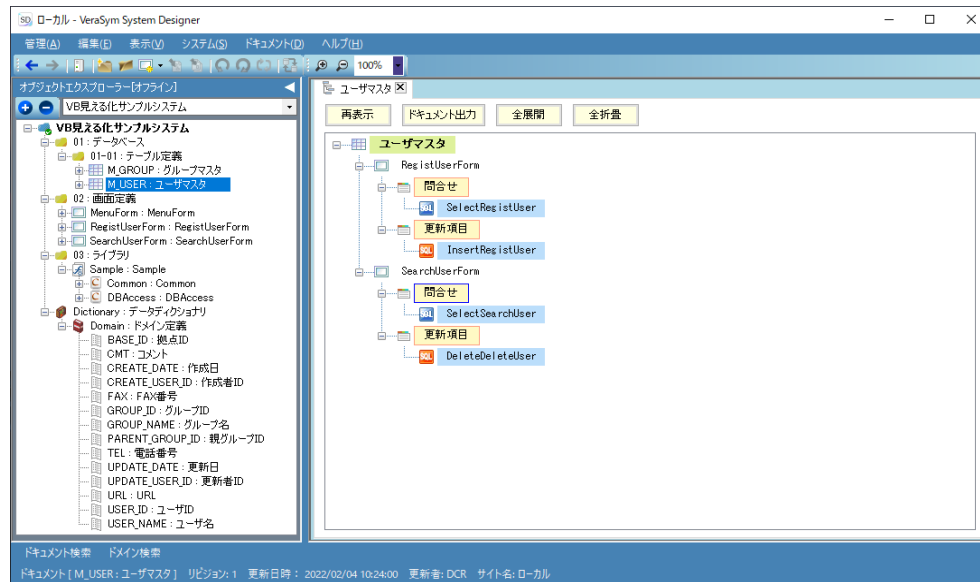
The '問合せ定義' (Query Definition) section also includes a '条件' (Condition) section with the following definition:

```
T1.ユーザID LIKE DBAccess.GetSqlStringValue (txtUserId.Text)  
[条件開始 txtUserName.Text IsNot Empty ]  
AND T1.ユーザ名 LIKE '%' ||  
DBAccess.GetSqlStringValue (txtUserName.Text)  
[条件終了]
```

[問合せ]タブでは、テーブル定義やビュー定義ドキュメントの定義を使用して、DB情報の問合せを定義できます。  
同様に、[更新項目]タブではDB情報の追加・更新・削除を定義できます。  
こうすることのメリットは「影響調査」シートをご確認ください。

# 影響調査

## オブジェクト関連図



テーブルや画面などのオブジェクトが他のオブジェクトから参照されているのか、また他のオブジェクトを参照しているのかをツリー構成で表現します。  
仕様変更時の影響範囲確認の精度向上、工数削減が実現します。

## CRUD表

ドキュメントID	ドキュメント名	区分	テーブルID/ビューID	テーブル名/ビュー名	作成 (C)	参照 (R)	更新 (U)	削除 (D)
7	SearchUserForm	SearchUserForm	M_USER	ユーザーマスタ	○	○	○	○
8	SearchUserForm	SearchUserForm	M_GROUP	グループマスタ	○	○	○	○
9	RegistUserForm	RegistUserForm	M_GROUP	グループマスタ	○	○	○	○
10	RegistUserForm	RegistUserForm	M_USER	ユーザーマスタ	○	○	○	○

画面や帳票、クラス定義の問合せ定義・更新項目定義で使用するテーブル・ビュー定義の情報を基に自動生成可能です。



製品紹介、デモ、サンプル設計書ご希望はこちらまで



**052-204-1377**



**vssd-info@dcr.co.jp**



**<https://vssd.dcr.co.jp/>**

「コウシタイ」という心の想いを  
“システムという形にする”



**DCR**

