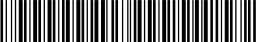


000-0-WI-0043	Phiên bản 6	Trang 1/11	
---------------	-------------	------------	---

I. Mục đích:

Tài liệu này mô tả quy chuẩn về cách đặt tên function, class, variable...khi lập trình trong bảng ngôn ngữ Java, VB6, VB.NET, C#.NET, SQL.

II. Áp dụng:

Áp dụng trong tất cả các phần được phát triển bởi FOV's software engineer

III. Định nghĩa thuật ngữ:

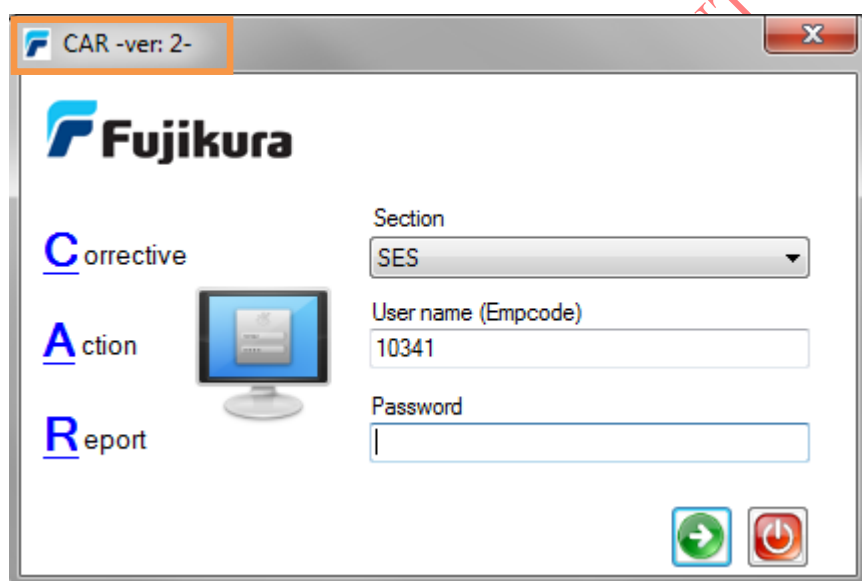
N/A

IV. Nội Dung**1.1. Quy định chung**

+ Logo của FOV phải hiển thị trên software

<\\fovsevr2\FileServer\SES\02.SourceCode\02.Image>

+ Tên software và version của software phải thể hiện trên software



+ Tất cả các câu truy vấn lấy dữ liệu từ SQL Server phải sử dụng store procedure.

+ Tên mỗi project theo format: *Prefix (prj) + danh từ hay cụm danh từ có nghĩa mô tả dự án*

Ex: prjLoss, prjLabelFixing

1.2. Quy định viết chú thích(Comment)

Confidential

FOV's Property, Do not take out without FOV's BOM Approval

Prepared by: Võ Duy Vũ Date: 22-Mar-23	Checked by: Ngô Nhật Nguyên Date: DMS date
Original: Nguyễn Trường Giang Date: 19-Mar-15	Approved by: Danh Nhu Date: DMS date



- Trên mỗi function đều phải ghi chú thích rõ ràng theo format như sau

Ví dụ :

+ VB6/VB.net/Java

```
''' <summary>
''' write down the description of the function
''' </summary>
''' <date></date>
''' <create_by></create_by>
''' <modify_by></modify_by>
''' <remarks></remarks>
```

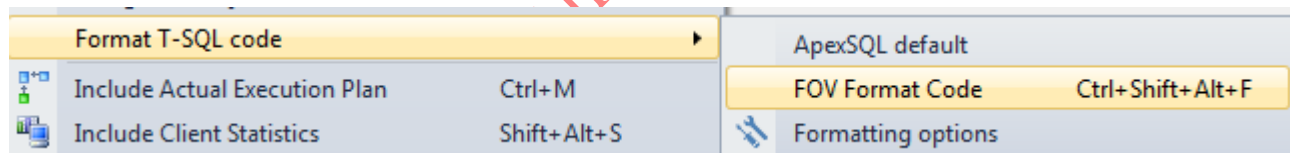
+ SQL

```
/*
*****
Created by: giangt
Created Date: 06/05/2015
Updated by:
Updated Date:
Description:
```

```
*****
*/
```

Lưu ý: Khi viết store procedure hay functions trong SQL thì sử dụng chức năng format của ApexSQL tool để format lại dòng lệnh cho chuẩn

Cài đặt tại đường dẫn \\fov\fileserver\Software\02.Software\01.Soft For Program\SQL Tool\2014



- Nếu cần thiết thì phải thêm chú thích ở từng dòng lệnh

Ví dụ

```
Dim sLabel as string ' your comment is here
```

1.3. Quy định xử lý lỗi (Error Handling)

- + Tất cả các hàm xử lý đều phải xử dụng Try...Catch hoặc On error. Nếu cần thiết thì phải ghi ra log file

Ví dụ:

+ VB6

```
Private sub example()
    On Error GoTo errExample
    ' your code is here
errExample:
    MsgBox "Error " & Err.number & ": " & Err.Description
    & " in " & Err.Source, vbExclamation, App.Title
End Sub
```

+ VB.net

```
Public sub example()
    Try
        \\ your code is here
```



```

        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End Sub

```

+ Java

```

Public void example()
{
    Try
        \\ your code is here
    Catch(Exception ex)

    End Try
}

```

+ SQL

```

Begin Transaction t1
// your code is here
If @@Error <>0
    Begin
        Rollback transaction t1
        return
    end
Commit transaction t1

Begin Try
    //your code is here
End try
Begin Catch
    //your code is here
End Catch

```

1.4. Quy định đặt tên class/module

- + Tên class/module nên được bắt đầu bằng từ hay 1 cụm từ có ý nghĩa mô tả đặc tính của class/module. Sử dụng danh từ cho các đối tượng trừu tượng hay động từ cho chức năng của class/module đó
- + Không sử dụng khoảng trắng (Space) giữa các từ
- + Viết hoa mỗi chữ cái đầu của mỗi từ
- + Thêm tiền tố : **cls** (đối với class) hay **mdl** (đối với module) vào đầu tên

Ví dụ:

```

clsCusomters.cs, clsCustomer.vb, clsCustomer.cls
mdlSendMessage.bas


```

1.5. Quy định đặt tên biến (parameter)

- + Tên biến nên là 1 danh từ hay 1 cụm danh từ có nghĩa
- + Không sử dụng khoảng trắng
- + Chữ cái đầu tiên nên viết thường, những chữ cái đầu mỗi từ sau đó viết hoa
- + Những chữ cái đầu sử dụng theo chuẩn dưới :

FUJIKURA FIBER OPTICS VIETNAM LTD.

CODING CONVENTION STANDARD

000-0-WI-0043	Phiên bản 6	Trang 4/11	
---------------	-------------	------------	---

Prefix	Type	Example
i	Integer or Long	iWordCount
f	Float (Single or Double)	fRadius
s	String	sFirstName
b	Boolean	bQuiet
c	Currency	cCurrentBal
d	Date	dStartTime
frm	Form	frmCustInput
txt	TextBox	txtLastName
lst	ListBox	lstServiceTypes
lvw	Listview	lvwData
grv	Gridview	grvData
dtg	Datagrid	dtgData
cmd/btn	Command Button	cmdCancel/btnCancel
cbo	ComboBox	cboCategory
pic	PictureBox	picLogo
chk	CheckBox	chkPrimaryAddress
opt	Option Button	optGenderFemale
tmr	Timer	tmrElapsed
lbl	Label	lblCopyright
tbr	Toolbar	tbrEditing
sbr	Statusbar	sbrLoad
pbr	Processbar	pbrComplete
tab	Tab	tabReport
dlx	Dialogbox	dlxFile
ctl	Control (if type is not known)	ctlSortableList
col	Collection	colFormFields
obj	Object (if type is not known)	objParent
var	Variant (if type is not known)	varNextField
dataset	dataset	dsResult
datareader	datareader	drResult
recordset	recordset	rsResult

*những kiểu dữ liệu hay control ở trên chưa đầy đủ nên cần phải được cập nhật.

+ **Đặt tên cho mảng** : Prefix (a) + kiểu dữ liệu (liệt kê ở bảng trên) + tên của mảng (viết hoa chữ cái đầu của mỗi từ)


Ex: afScore()

+ **Đặt tên cho biến toàn cục**: Prefix (g) + kiểu dữ liệu (liệt kê ở bảng trên) + tên của mảng (viết hoa chữ cái đầu của mỗi từ)

Ex: gsUsername as string

+ **Đặt tên hằng số** : viết hoa tất cả các từ , mỗi từ cách nhau bằng gạch dưới

Ex: CURRENCY = "VND" ; MACHINE_NAME = 'vese9482'

000-0-WI-0043	Phiên bản 6	Trang 5/11	
---------------	-------------	------------	---

+ **Đặt tên interface:** *Prefix (I) + Sử dụng danh từ hay cụm danh từ có nghĩa (viết hoa chữ cái đầu của mỗi từ)*

Ex: `IComponent; IFormatTable`

+ **Đặt tên namespace:** *FOV.projectname.packagename.namespacename*

Ex: `FOV.LabelFixing.GlobalFunction.Connection`

1.6. Quy định đặt tên hàm, thủ tục (function/procedure/sub)

+ Hàm nào trả về kết quả là function, hàm nào trả về void/sub là procedure.

+ Tên hàm nên bắt đầu bằng động từ hay cụm động từ mô tả chức năng của hàm đó.

+ Nếu là function thì thêm *prefix (f_)* vào đầu, nếu là procedure thì thêm *prefix (p_)*

+ Không sử dụng khoảng trắng (space), từ đầu tiên viết thường, những chữ cái bắt đầu của mỗi từ sau đó viết hoa

Ví dụ:

```
f_openDataFile()
p_sendEmail()
```

+ Với những hàm có chức năng trả về giá trị là Boolean (true or false) thì nên bắt đầu bằng `f_isXXX()`, `f_hasXXX()` hay `f_canXXX()`

Ví dụ:

```
Connection.f_isAvailable()
Router.f_hasPendingRequest()
```

+ Với những hàm chuyển đổi kiểu nên bắt đầu bằng `f_toXXX()`

Ví dụ:

```
f_toString()
```

1.7. Quy định đặt tên table, store procedure, functions trong SQL

+ Tên table đặt theo dạng sau:

Prefix (tbl) + softcode + danh từ hay cụm danh từ có nghĩa

Ex: `tblLFX_ProLabel`

+ **Tên store procedure đặt theo dạng sau :**

Prefix (p_) + softcode + động từ (viết thường) + danh từ hay cụm danh từ có nghĩa (viết hoa chữ cái đứng đầu)

Ko dùng tiền tố “sp” vì sẽ trùng với những store procedure của system


Ex: `p_HR_getEmpCode`

+ **Tên function đặt theo dạng sau:**

Prefix (f_) + softcode + động từ (viết thường) + danh từ hay cụm danh từ có nghĩa (viết hoa chữ cái đứng đầu)

Ex: `f_PL_getPackingList()`

+ **Tên trigger đặt theo dạng sau:**

000-0-WI-0043	Phiên bản 6	Trang 6/11	
---------------	-------------	------------	---

Prefix (trig_) + softcode(nếu có) + động từ (viết thường) + danh từ hay cụm danh từ có nghĩa(viết hoa chữ cái đứng đầu)

Ex: `trig_checkUserValid()`

+ **Tên package đặt theo dạng sau:**

Prefix (pack_) + danh từ hay cụm danh từ có nghĩa(viết thường chữ cái đứng đầu)

Ex: `pack_myCommonFunc()`

1.8. Quy định về bố cục

+ Dùng dấu ngoặc kép

Ex:

```
if (a = b & c = d) //bad
if ((a=b) & (c=d)) //good
```

+ Độ dài các ký tự trên 1 dòng

Ex:

```
//bad
Function CollectStats(sFileName As String,iStartLine As Long,iEndLine As Long)
As Long
//good
Function CollectStats(sFileName As String, _
    iStartLine As Long, _
    iEndLine As Long) _
As Long

//bad
longName1 = longName2 * (longName3 + longName4
    - longName5) + 4 * longname6;


//good
longName1 = longName2 * (longName3 + longName4 - longName5)
    + 4 * longname6;
```

1.9. Quy định đặt mã phiên bản

+ Đặt mã phiên bản theo nguyên tắc sử dụng 2 số cho mỗi phiên bản phần mềm.

4.2
MAJOR Minor

+ Nguyên tắc thay đổi mã phiên bản dựa theo các tiêu chí từ yêu cầu của người dùng và tham chiếu theo bảng dưới để quyết định.

000-0-WI-0043	Phiên bản 6	Trang 7/11	
---------------	-------------	------------	---

<i>Scope/Phạm vi</i>	MAJOR	<i>Minor</i>
<i>Add/Change Functionality / Thay đổi hoặc bổ sung tính năng</i>		<i>x</i>
<i>Change 4M characteristic / Ít nhất 1M trong 4M thay đổi thuộc tính</i>		<i>x</i>
<i>UI convenient / Thay đổi các tiện ích</i>		<i>x</i>
<i>New software / Phần mềm mới</i>	<i>x</i>	
<i>Change Data Structure / Thay đổi cấu trúc lưu trữ</i>	<i>x</i>	
<i>Change Procedure control / Thay đổi thủ tục (workflow) quản lý</i>	<i>x</i>	
<i>Software Error / Lỗi phần mềm cần khắc phục</i>	<i>x</i>	

+ Áp dụng cho tất cả các phần mềm có nhận RQ từ khách hàng kể từ sau ngày 01-Apr-2023.

1.10. Quy định về logging

- Tất cả các function chứa các dữ liệu bao gồm attribute / para (must be filled) mặc định được lưu login khi xử lý các input / output từ nội bộ phần mềm và các phần mềm liên kết.
- Logging được yêu cầu trước và sau xử lý khi thực hiện truy vấn từ server hoặc từ resource khác như api, lib ...

Ví dụ:

Request to SQL:

P_LFX_KeepStock_CheckInfor_SO 'LLT0074','FPO200117261-3', '275779'

Response from SQL

Result: NG, message "Not found PO"

- Audit trail: yêu cầu giám sát tối thiểu hoạt động của người dùng, tức cần nhận diện họ đã làm gì, request gì, nhập liệu gì và xử lý cá event nào kết quả ra sao.
- Logging được tổ chức theo định dạng file có thể truy xuất được, có format tùy chọn nhưng nên theo khuyến nghị:

Tên file: chia theo ngày định dạng *YYYYMMDD.extension*

Nội dung: break lines by lines, timestamp, content

1.11. Quy định liên quan xử lý bất đồng bộ

Khi thực thi các tác vụ UI (giao diện người dùng) và Batch (background running), yêu cầu nghiêm ngặt việc kiểm soát đồng bộ dữ liệu sau mỗi event từ chương trình hoặc từ người dùng. Chỉ được thông tin đến giao diện người dùng khi hoàn tất tác vụ, các chức năng phía sau nếu có phải được hidden hoặc *editable = false* hoặc *enable = false*. Có nhiều hướng kiểm soát như khuyến nghị sử dụng *control loading* hoặc *progressbar* để tương tác với người dùng.



1.12. Quy định về xử lý nested/ sub query trong quy vấn SQL.


- Trong các mệnh đề điều kiện, tuyệt đối không sử dụng các chuỗi query lồng nhau nhiều cấp (nested / sub query) như select from (select from (select from) where not in (select from) where in (select from)
- Chỉ được sử dụng linkserver object để truy vấn các dữ liệu phức tạp được tạo ra từ các sp có sẵn.

1.13. Một số quy tắc chung cho các nút / Some general rules for buttons

- Button là phương tiện tương tác phổ biến nhất của các trang và hộp thoại có thể thực hiện được. Chúng ta sẽ cố gắng tìm ra ý nghĩa phù hợp và vị trí đặt chúng tốt hơn và thứ tự của chúng để giảm thiểu các sự cố có thể xảy ra khi thêm nhiều màn hình.
- Dưới đây là chia thành 2 nhóm chính chức năng chính và 13 nhóm nhỏ có ý nghĩa tương tự

FUJIKURA FIBER OPTICS VIETNAM LTD.

CODING CONVENTION STANDARD


000-0-WI-0043	Phiên bản 6	Trang 9/11	
---------------	-------------	------------	---

No	Label	Group	VN Description
1	Create New Save drafft Edit Copy Capture Measure CheckOut	Actionable	Tạo mới một đối tượng dữ liệu hoặc các thuộc tính của đối tượng, cho phép chỉnh sửa và dữ liệu chưa hiệu lực.
2	Save Add Clone Book CheckIn		Lưu dữ liệu một đối tượng hoặc các thuộc tính của đối tượng. Nếu không yêu cầu phê duyệt, dữ liệu sẽ có hiệu lực chính thức.
3	Submit Post Release Save & Release Save & Send		Khóa dữ liệu của đối tượng hoặc các thuộc tính của đối tượng để thực thi hành động gửi đi phê duyệt
4	Approval Accept Sync Data		Nhóm nút chức năng phê duyệt dữ liệu của đối tượng, dữ liệu được ban hành chính thức và hiệu lực.
5	Return Recall		Nhóm nút chức năng phê duyệt yêu cầu bổ sung dữ liệu của đối tượng, không làm thay đổi đối tượng và dữ liệu của đối tượng không có hiệu lực.
6	Reject Decline		Nhóm nút chức năng phê duyệt từ chối dữ liệu của đối tượng, không làm thay đổi đối tượng và dữ liệu của đối tượng không có hiệu lực.
7	Delete Remove Remove All		Nhóm nút dùng để xóa / hủy một hoặc nhiều dòng dữ liệu làm thay đổi cấu trúc dữ liệu của đối tượng
8	Search Refresh View Select Choose Export		Nhóm nút dùng để truy xuất dữ liệu không làm thay đổi dữ liệu của đối tượng
9	OK	Dialogs	Đóng hộp thoại thông báo
10	Cancel		Bỏ qua hộp thoại thông báo
11	Yes		Xác nhận đồng ý từ hộp thoại thông báo
12	No		Xác nhận không đồng ý từ hộp thoại thông báo
13	Close		Đóng hộp thoại thông báo

- Cách thức tương tác giữa các Button và dữ liệu được mô tả trong bảng dưới

FUJIKURA FIBER OPTICS VIETNAM LTD.


CODING CONVENTION STANDARD

000-0-WI-0043	Phiên bản 6	Trang 10/11	
---------------	-------------	-------------	---

No	Label	Group	Data processing						
			Local data				Global data (Link metadata to related resources)		
			Temporary storage (Unsave)	Commit	Rollback	Not thing	Commit	Rollback	Not thing
1	Create New Save draft Edit Copy Capture Measure CheckOut	Actionable	●	○	○	○	○	○	○
2	Save Add Clone Book CheckIn		●	●	○	○	●	○	○
3	Submit Post Release Save & Release Save & Send		○	●	○	○	●	○	○
4	Approval Accept Sync Data		○	●	○	○	●	○	○
5	Return Recall		○	○	○	●	○		●
6	Reject Decline		○	○	○	●	○	○	●
7	Delete Remove Remove All		○	●	○	○	●	○	○
8	Search Refresh View Select Choose Export		○	○	○	○	○	○	○
9	OK	Dialogs	○	○	○	○	○	○	○
10	Cancel		○	○	○	○	○	○	○
11	Yes		○	○	○	○	○	○	○
12	No		○	○	○	○	○	○	○
13	Close		○	○	○	○	○	○	○

FUJIKURA FIBER OPTICS VIETNAM LTD.

CODING CONVENTION STANDARD

000-0-WI-0043	Phiên bản 6	Trang 11/11	
---------------	-------------	-------------	---

REVISION HISTORY

Date	Person	Version	Old content description	New content		Reason of change
				Description	Change requester	
19-Mar-14	Nguyễn Trường Giang	01				New
06-Jun-15	Nguyễn Trường Giang	02		1.2 Quy tắc viết chú thích 1.7 Quy định đặt tên table trong SQL		
01-Mar-23	Võ Duy Vũ	03		1.9 Quy định đặt mã phiên bản	Danh Nhu	
22-Mar-23	Võ Duy Vũ	04		Thay đổi Header và Footer của tài liệu.	Danh Nhu	
25-Jul-24	Ngô Nhật Nguyên	05		1.10 Quy định về Logging 1.11 Quy định liên quan xử lý bất đồng bộ 1.12 Quy định về sub query trong truy vấn SQL	Danh Nhu	
25-Sep-24	Ngô Nhật Nguyên	06		1.13 Một số quy tắc chung khi thiết kế các nút	Danh Nhu	Standardize button