

 corporación maresa	Instructivo de Trabajo	Revisión: 01
	Ref: Procedimiento de Desarrollo de Aplicaciones	
IT - POMPN58-03	Nomenclatura de Desarrollo de Aplicaciones	Página 1 de 11
Elaborado por: Patricio Soria	Cargo: Jefe de Proyectos y Mantenimiento de Aplicaciones	FIRMA:
Aprobado por: Juan Carlos Játiva	Cargo: Gerente de Tecnología y Procesos	FIRMA:

1. INFORMACION GENERAL

1.1 Historia de Revisiones y Causas

# DE REVISIÓN	CAUSA DE LA REVISIÓN	AUTOR	FECHA DE IMPLEMENTACIÓN
01	LANZAMIENTO INICIAL	Patricio Soria	21 – MAR - 2018

1.2 Documentos de Referencia

CÓDIGO DEL DOCUMENTO	NOMBRE DEL DOCUMENTO
PO - MPN - 58	<i>Desarrollo de Aplicaciones</i>

2. PROPOSITO

Unificar la metodología y el lenguaje utilizado en la construcción de productos de software que se despliegan en la plataforma de Corporación Maresa. Mejorar la calidad de los desarrollos y la mantenibilidad de estos, dando cumplimiento a los principios y fundamentos de la Estrategia de TI, como la innovación, confianza y seguridad.

3. ALCANCE

Los lineamientos aplican para el desarrollo de todos los proyectos de software que se estén ejecutando en la entidad. Deben ser acogidos por el Área de Tecnologías y Sistemas de Información y se debe reflejar en cada producto puesto en producción, ya sean nuevos desarrollos o ajustes de sistemas existentes.

Aplica a todas las UENs de Corporación Maresa, excepto (Mareauto).

4. DEFINICIONES Y ABREVIATURAS

CONCEPTO	DEFINICIÓN
<i>Assembly</i>	<ul style="list-style-type: none"> Uno a más archivos que son la salida de la compilación de un programa. Fuente de esta información.
<i>AssemblyInfo</i>	<ul style="list-style-type: none"> Clase que provee atributos para obtener información sobre el programa, tal como número de versión, descripción, etc.
<i>Bases de datos</i>	<ul style="list-style-type: none"> Conjunto de datos o archivos que tiene una estructura en común, la cual está organizada de tal forma que el ordenador pueda fácilmente encontrar la información
<i>CamelCase</i>	<ul style="list-style-type: none"> Estilo de escritura que se aplica a frases o palabras compuestas y se asemejan a las jorobas de un camello
<i>C#</i>	<ul style="list-style-type: none"> Lenguaje de programación orientado a objetos, desarrollado y estandarizado por Microsoft, forma parte de la plataforma .NET.
<i>Excepción</i>	<ul style="list-style-type: none"> Indicación de un problema en la ejecución de un programa desarrollado.
<i>Indentación</i>	<ul style="list-style-type: none"> Mover un bloque de texto hacia la derecha con espacios o tabulaciones para distinguirlo del texto adyacente
<i>keyword internal</i>	<ul style="list-style-type: none"> Palabra clave que referencia a un modificador de acceso a los tipos y miembros de una clase que pertenecen a un mismo conjunto de Librerías y clases
<i>kebab-case</i>	<ul style="list-style-type: none"> Nomenclatura que consiste en escribir todas las palabras en letras minúsculas separadas con un guion
<i>Loop</i>	<ul style="list-style-type: none"> Bucle de programación
<i>Namespace</i>	<ul style="list-style-type: none"> Sistema de organización lógico que proporciona una forma de presentación de los elementos del programa que son expuestos hacia otros
<i>Namespace System</i>	<ul style="list-style-type: none"> Sistema de organización lógico que contiene clases base que definen tipos de datos de uso común, eventos, controladores de eventos, interfaces, atributos y excepciones de procesamiento. Fuente de esta información

5. RESPONSABILIDAD

QUIEN	RESPONSABILIDADES Y AUTORIDADES
<i>Jefe de Proyectos y Mantenimiento de</i>	<ul style="list-style-type: none"> Definir Nomenclatura y lineamientos técnicos a utilizar para desarrollo de aplicaciones.

QUIEN	RESPONSABILIDADES Y AUTORIDADES
<i>Aplicaciones</i>	
<i>Analista Programador</i>	<ul style="list-style-type: none"> Validar que las nomenclaturas se encuentren implementadas en los desarrollos realizados y/o en los desarrollos entregados por los proveedores

6 PROCEDIMIENTO

6.1 Prefijos y Subfijos

Es obligatorio utilizarlos para denominar cada tipo de elemento identificado como un artefacto de desarrollo, su función es identificar diferentes tipos de objetos, datos, entradas o salidas. Se puede aplicar para:

- Directorios y estructura de archivos.
- Archivos
- Clases, Interfaces, Servicios, Formularios, Apis, Enumeraciones, Etc.
- Proyectos, Soluciones, Espacios de Nombres.

6.1.1 Reglas de generación.

Los prefijos y sufijos deben contener exactamente 3 letras.

Si la palabra tiene al menos 3 consonantes, se escogen las primeras 3 consonantes, ejemplo:

Desarrollo: dsr

- Metodología: mtd
- Trabajo: trb

Si la primera palabra tiene menos de 3 consonantes y empieza por una vocal, se escoge la primera vocal y las siguientes consonantes o vocales.

- Anual: anl
- Opera: opr
- Único: unc

Si la palabra tiene menos de 3 consonantes y empieza por una consonante, se escogen las consonantes y vocales en el orden que aparezcan.

- Tema: tem
- Cueva: cuv
- Trio: tri

Ejemplos de prefijos:

- fnt indicando que se trata de una función

- int, dbl, bol, indicando el tipo de datos de una variable.
- txt, cmb, ddl, opt, gvw, indicando el tipo de componente
- TBL, BDS, VWS, indicando el tipo de objeto de Base de Datos.

Ejemplos de sufijos

- lst, obj, int, str indicando que una el resultado de una función es una lista de objetos o la instancia de un objeto.

Estilo de escritura o sintaxis de los prefijos o sufijos.

Los prefijos y sufijos se toman como una nueva palabra adicionada al elemento que se quiere describir y por lo tanto deben respetar la sintaxis del elemento.

6.2 Nomenclatura Base de Datos.

6.2.1 Bases de Datos, nombres y rutas de archivos.

El nombre de la base de datos debe venir precedido del prefijo DB (Base de datos) y seguido por el carácter (_) underline.

PREFIJO	EJEMPLO
DB_	[Servidor].[Schema].[DB_CATALOGOS]
	[Servidor].[Schema].[DB_ADMINVEH]

6.2.2 Tablas y vistas.

El nombre de la tabla debe venir precedido del prefijo TB (Tabla) y seguido por el carácter (_) underline.

Los nombres de las tablas deben reflejar de manera precisa su contenido y función. Se recomienda la notación Pascal_Snake_Case: No se deben utilizar espacios en el nombre de archivo.

Los nombres de las vistas siguen las mismas convenciones que los nombres de las tablas. Se recomienda utilizar el prefijo "VW".

PREFIJO	EJEMPLO
TB_	[Servidor].[Schema].[Base_Datos].[TB_Cursos]
	[Servidor].[Schema].[Base_Datos].[TB_Pagos]

6.2.3 Campos.

Las siguientes recomendaciones aplican a todos los campos.

El nombre de los campos deben venir precedido del prefijo PK o FK, seguido por el carácter (_) underline.

Cada nombre de campo debe ser único dentro de su tabla correspondiente.

Cada nombre de campo debe derivarse del identificador utilizado durante el proceso de análisis de negocio.

No se deben utilizar palabras reservadas como nombres de campos.

Campos	PREFIJO	EJEMPLO
Campos propios	[servidor].[sln_Plan_Compras].[DB_Plan_Compras].[TB_Pagos]	PGS_Descripcion
	[servidor].[sln_Plan_Compras].[DB_Plan_Compras].[TB_Contabilidad]	CTB_Valor_Moneda
	[servidor].[sln_Plan_Compras].[DB_Plan_Compras].[TB_Personas]	PSN_Observaciones

6.3 Nomenclatura de programación.

6.3.1 Definición de variables

Comience el nombre de la variable con una letra.

La primera letra de cada palabra en mayúscula.

Debe describir claramente el propósito de la variable.

Usar un prefijo que indique el tipo de dato.

Usar comentarios que describan el ámbito en el que se encuentran definidas las variables.

La nomenclatura que se debe usar es camelCase

TIPO DE DATO	PREFIJO	EJEMPLO
Bool, Booleano (boolean)	bol/bln	bolEsMenor
Byte	byt	bytEdad
(fecha-hora) date-time	dtm/dat	datFechaInicial
doble (double)	dbl	dblValorDescuento
Int, Integer (entero)	int	intEdad
Long (entero largo)	lng	lngTotalEmpleados
Char (caracteres unicode)	chr	chrNombre
String (cadena de caracteres)	str	strNombre
Float	flt	fltTotalDescuento
Decimal	dcm	dcmToneladas

6.3.2 Definición de Funciones

Usar el prefijo: fnc

Usar el sufijo: tipo de dato que retorna, deben ser tan largos como sea necesario para describir su funcionalidad, la Primera letra del nombre en Mayúscula. Nomenclatura camelCase.

Ej: *fntNombreFuncionInt, fntCalcularSumaDbl, fntResolverNombreStr*

Si la función no retorna nada, no se usará sufijo.

Ej: *fncNombreFuncion, fncNotificarUsuario*

6.3.3 Definición de Procedimientos

Usar el prefijo Prc, Primera letra del nombre en mayúscula. Incluir además de la acción, la entidad que será devuelta: Verbo-sustantivo (CalcularDesperdicio()), nomenclatura camelCase.

Ej: *PrcCalcularDesperdicio*

6.3.4 Definición de Clases

Usar el prefijo Cls. La primera letra de cada palabra en mayúscula. Nomenclatura PascalCase.

Ej: *ClsCuentaBancaria*

6.3.5 Definición de Objetos

Utilizar prefijos para definir los objetos:

Prefijo	Tipo y Significado	Ejemplos
lbl	Label	lblNombre
lnk	LinkLabel	lnkNombre
btn	Button	btnNombre
txt	TextBox	txtNombre
mnm	MainMenu	mnmNombre

chk	CheckBox	chkNombre
rdn	RadioButton	rdnNombre
gbx	GroupBox	gbxNombre
pct	PictureBox	pctNombre
pnl	Panel	pnlNombre
dtg	DataGrid	dtgNombre
lbn	ListBox	lbnNombre
cbx	ComboBox	cbxNombre
lsn	ListView	lsnNombre
tvw	TreeView	tvwNombre
gvw	GridView	gvwNombre
tbc	TabControl	tbcNombre
prg	ProgressBar	prgNombre
rtb	RichTextBox	rtbNombre
img	ImageList	imgNombre
stb	StatusBar	stbNombre
dtp	DateTimePicker	dtpNombre
ddl	DropDownList	ddlNombre
hdf	HiddenField	hdfNombre
cld	Calendar	cldNombre
rep	Repeater	repNombre
rfv	RequiredFieldValidator	rfvNombre
smg	ScriptManager	smgNombre
upl	UpdatePanel	uplNombre
ups	UpdateProgress	upsNombre
crv	CrystalReportViewer	crvNombre

Tabla resumen de nombramientos:

	DESCRIPCIÓN	PREFIJO	EJEMPLO
VARIABLES	Bool (boolean)	bol	bolEsMenor
Byte	byt	bytEdad	
Date (date-time)	dtm	dtmFechaInicial	
Double (double)	dbl	dblValorDescuento	
Integer (entero)	int	intEdad	
Long (entero largo)	lng	lngTotalEmpleados	
Char (caracteres)	chr	chrNombre	
String (cadena)	str	strNombre	
Float	flt	fltTotalDescuento	
Decimal	dec	decToneladas	
FUNCIONES	Sin retorno	fun	fntNombre
Retorna Int	fun	funNombreInt	
	Retorna Objeto	fun	fntNombreObj
	Retorna Double	fun	fntNombreDbf
	Retorna String	fun	fntNombreStr

	DESCRIPCIÓN	PREFIJO	EJEMPLO
PROCEDIMIENTOS	Procedimientos	prc	prcNombreProcedimiento
CLASES	Clases	cls	clsNombreClase
OBJETOS	Label	lbl	lblNombre
LinkLabel	lnk	lnkNombre	
Button	btn	btnNombre	
TextBox	txt	txtNombre	
MainMenu	mnm	mnmNombre	
CheckBox	chk	chkNombre	
RadioButton	rdn	rdnNombre	
GroupBox	grp	grpNombre	
PictureBox	pct	pctNombre	
Panel	pnl	pnlNombre	
DataGrid	dtg	dtgNombre	
ListBox	lbn	lbnNombre	
ComboBox	cbx	cbxNombre	
ListView	lsn	lsnNombre	
TreeView	tvn	tvnNombre	
GridView	gvn	gvnNombre	
TabControl	tbc	tbcNombre	
ProgressBar	prg	prgNombre	
RichTextBox	rtb	rtbNombre	
ImageList	img	imgNombre	
StatusBar	stb	stbNombre	
DateTimePicker	dtp	dtpNombre	
FORMULARIOS	WEB	frm	frmNombreFormulario
MAESTRA	mtr	mtrNombreMaestra	
REPORTES	Reportes	rpt	rptNombreReporte
WEBSERVICES	WebServices	wsv	wsvNombreWebService
SOLUCIONES	Soluciones	sln	slnNombreSolución
PROYECTOS	Proyectos	prj	prjNombreProyecto
ESTILOS	Estilos	sty	styNombreEstilo

IT - POMPN58-03	Nomenclatura de Desarrollo de Aplicaciones	Página 8 de 11
------------------------	---	-----------------------

6.4 Accesibilidad y usabilidad

Siguiendo los Lineamientos y metodologías de Usabilidad para Corporación Maresa en línea. Se debe garantizar el acceso universal a la Web, independientemente del tipo de hardware, software, infraestructura de red, idioma, cultura, localización geográfica y capacidades de los usuarios para lo cual se debe tener en cuenta las siguientes recomendaciones:

6.4.1 Diseño Centrado en el usuario

En términos generales, el diseño centrado en el usuario es una filosofía y proceso de diseño en el que las necesidades, los deseos y las limitaciones del usuario final de una interfaz o documento toman una atención y relevancia considerable en cada nivel del proceso de diseño.

6.4.2 Directrices de usabilidad web

Las directrices más relevantes, ayudan a tener un portal, sitio o aplicación orientada al usuario con información presentada de forma clara y concisa.

Objetivos del Portal Web: Las necesidades concretas de la organización y lo que esta espera conseguir a través del sitio web se ven plasmadas en los objetivos. Serán estos, en armonía con las necesidades de los usuarios, los que guiarán todo el proceso de conceptualización, diseño e implementación del portal.

Navegación global consistente. Proporcione una navegación global visible y consistente a lo largo del sitio web.

URL limpios. El URL es el la abreviatura de (Uniform Resource Locator) Localizador Uniforme de Recursos, es el nombre técnico con el que se conoce a la comúnmente llamada —direcciónll.

Un esquema de URL limpios tiene las siguientes características:

Corresponde a la jerarquía del sitio

No incluye caracteres especiales como \$, &, ?, = entre otros.

6.5 Diseño de Interfaz del usuario

Ubicación del logotipo: Ubique el logotipo de la entidad en el mismo lugar en todas las páginas y vincúlelo con la página de inicio.

Interfaces en movimiento. Evite diseñar interfaces en movimiento. Las posibilidades de movimiento de elementos de las interfaces, a partir de tecnologías como Flash, animaron a algunos diseñadores a proponer páginas

IT - POMPN58-03	Nomenclatura de Desarrollo de Aplicaciones	Página 9 de 11
-----------------	--	----------------

web más llamativas visualmente, por la misma novedad que supone la diferencia. Sin embargo, este recurso conlleva dificultades, particularmente en la facilidad de uso y en la accesibilidad. El siguiente es un claro ejemplo, a través de un sitio de venta de repuestos para computadores.

Contraste en brillo y color: Verifique que el texto y las imágenes de texto tengan suficiente contraste de brillo y color con el fondo.

Información transmitida a través de color: Debido a que un porcentaje de la población (8% de hombres y 1% de mujeres) tienen una percepción visual diferente, llamado daltonismo, el diseño de las páginas web debe asegurar que cuando se transmita información a través del color, se provean mecanismos alternativos para que toda la población entienda.

Esta medida beneficiaría no solo a la población con daltonismo, sino también a personas que utilicen pantallas con deficiencia cromática, monocromáticas o en condiciones donde la percepción de luz pueda verse alterada.

Justificación del texto: Evite la alineación justificada del texto de prosa al margen izquierdo y derecho a la vez. El diseño tipográfico que contempla la alineación justificada (margen derecho e izquierdo al mismo tiempo), es una práctica habitual en la industria de los impresos (libros, revistas, folletos).

Ancho del cuerpo de texto. Utilice un ancho promedio entre 60 y 80 cpl (caracteres por línea) para el cuerpo de texto. Un renglón corto en cpl implica que el ojo tiene que realizar constantemente saltos de línea, que lo lleva rápidamente a cansarse. De otra parte, para un renglón demasiado largo la vista deba realizar un recorrido muy largo, produciendo de igual forma fatiga visual.

Para los sitios web del Estado Ecuatoriano, se ha seleccionado un tamaño medio, a partir de la investigación de García y Short (2002) que sugiere el renglón ideal entre 60-80 cpl.

Desplazamiento horizontal: La resolución mínima para monitores de computador que debe tenerse en cuenta en la actualidad para los sitios del Estado Ecuatoriano, es de 1024 x 768 pixeles. Sobre esta medida, el diseñador debe probar su diseño para que en ningún navegador aparezca la barra de desplazamiento horizontal.

Por ello se sugiere una medida de 960 pixeles de ancho, que encaja perfectamente en la resolución establecida en esta directriz y que no generará barra de desplazamiento horizontal.

Hojas de estilo para diferentes formatos: Asigne estilos para lectura en pantalla e impresión en papel.

Independencia de navegador: Aunque Internet Explorer sea actualmente el navegador más utilizado, esta tendencia cada día está a la baja, con la aparición de excelentes alternativas libres, como Firefox, Opera, Chrome y

IT - POMPN58-03	Nomenclatura de Desarrollo de Aplicaciones	Página 10 de 11
-----------------	--	-----------------

Safari, navegadores que cada día demuestran más rapidez en su desempeño y, sobre todo, más respeto por los estándares web.

Calidad del código: Verifique que el código HTML y CSS de su sitio cumpla estándares. Codificar de acuerdo con los estándares es en principio una forma de garantizar que el sitio web se verá adecuadamente en la mayoría de navegadores y dispositivos. Escribir código válido no solo es un indicador de calidad, sino una clara manifestación de que quien diseña el sitio hace un esfuerzo por generar un sitio interoperable y accesible.

Para verificar que el código se escribe de acuerdo con el estándar emitido por la W3C, existen diversas herramientas de validación.

Diseño de interacción

Campos obligatorios: Distinga claramente los campos obligatorios de los opcionales. Uno de los elementos que puede afectar la experiencia de usuario es una mala señalización de los campos de formulario. El usuario se ha habituado a encontrar una convención para aquellos campos que son requeridos, por eso en ocasiones los campos sin señalización pueden ser interpretados como opcionales.

Asociación de etiquetas y campos: Asocie claramente las etiquetas con los campos de formulario. Para que el usuario tenga una interacción fluida con los formularios se hace necesario que conozca los datos que el sistema espera que ingrese en cada campo. Un adecuado rotulado de los campos de formulario, permite una lectura rápida y un ingreso ágil de la información.

Validación dinámica de datos: Proporcione una validación dinámica de datos, antes de que el usuario envíe un formulario. Los lenguajes de programación ejecutados del lado del cliente pueden beneficiar mucho la experiencia de los usuarios al diligenciar formularios. Proporcionar una validación dinámica en línea antes de enviar el formulario completo hace más sencillo el flujo de trabajo y evita al usuario errores innecesarios.

Error de página no encontrada: Diseñe e implemente una página específica para el error de página no encontrada (404). Que no exista ningún tipo de error cuándo los usuarios interactúan con un sitio web es algo difícil de garantizar. Un simple enlace mal formulado, una URL que cambia, un dato ingresado de manera incorrecta e incluso una acción equivocada del usuario pueden desencadenar en un error.

Un error común es el de página no encontrada, conocido a veces por su código (404). Es un error que aparece cuándo el usuario intenta ingresar a una página del sitio que no existe o ha sido eliminada.

Ejemplos en los campos de formulario: Proporcione ejemplos en los campos de formulario que sean de difícil comprensión. Los ejemplos son una forma sencilla y clara de orientar al usuario. El visitante entiende claramente la forma

IT - POMPN58-03	Nomenclatura de Desarrollo de Aplicaciones	Página 11 de 11
------------------------	---	------------------------

de diligenciar un formulario a través de ejemplos, especialmente en aquellos campos que requieren algún formato especial.

6.6 Requisitos Técnicos

- Norma Técnica ISO/IEC 23270:2006 especifica la forma y establece la interpretación de los programas escritos en el lenguaje de programación C#.
- Norma Técnica Ecuatoriana NTE 5854, Accesibilidad a Páginas Web.
- C# Language Specification ECMA INTERNATIONAL Standard ECMA-334 4th edición – Junio 2006 ver: <https://www.iso.org/obp/ui/es/#iso:std:iso-iec:23270:ed-2:v1:en>