Implementación MAAGTIC

Guía de Estándares Progress

Versión 1.0

26/05/2011

Histórico de Versiones

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 26/05/2011 | 1.0 | Generación de documento | Patricia Vázquez Montellano |
|  |  |  |  |

Tabla de Contenido

[1. Introducción 4](#_Toc294262236)

[1.1 Objetivo 4](#_Toc294262237)

[1.2 Definiciones, acrónimos y abreviaturas 4](#_Toc294262238)

[1.3 Referencias 4](#_Toc294262239)

[2. General 4](#_Toc294262240)

[2.1 Estructura de directorios 4](#_Toc294262241)

[3. Estándares de programación Progress 5](#_Toc294262242)

[3.1 Comentarios Iniciales 5](#_Toc294262243)

[3.2 Sangría 5](#_Toc294262244)

[3.3 Saltos de línea 5](#_Toc294262245)

[3.4 Inicialización de variables 5](#_Toc294262246)

[3.5 Comentarios de bloque 5](#_Toc294262247)

[3.6 Declaración de variables 5](#_Toc294262248)

[3.7 Declaración if-else 6](#_Toc294262249)

[3.8 Declaración for 6](#_Toc294262250)

[3.9 Declaración while 6](#_Toc294262251)

[3.10 Declaración do-while 6](#_Toc294262252)

[3.11 Declaración case 6](#_Toc294262253)

**Guía de Estándares Progress**

# Introducción

Este documento presenta los estándares para desarrollar una solución tecnológica luego de seguir el proceso de Desarrollo de Soluciones Tecnológicas (DST) según se indica en el MAAGTIC. Estos estándares se refieren básicamente al uso del lenguaje Progress como tal y no al uso de herramientas de desarrollo en Progress.

## Objetivo

Este documento tiene como objetivo brindar una guía de los estándares a emplear en los desarrollos de aplicaciones tecnológicas con Progress.

## Definiciones, acrónimos y abreviaturas

N/A.

## Referencias

* Anexo A. Nomenclatura para identificadores.
* Anexo B. Tipos de datos en Progress.

# General

## Estructura de directorios

Se generará un directorio que contenga toda la información del proyecto, dentro del cual se albergarán los directorios de cada uno de los módulos que lo conforman.

La estructura a utilizar es la siguiente:

|  |  |  |
| --- | --- | --- |
| \Sistema |  |  |
|  | \Versión |  |
|  |  | \adm2 |
|  |  | \pf |
|  |  | \img |
|  |  | \Subsistema |

| **Directorio** | **Descripción** |
| --- | --- |
| **\Sistema** | Es el directorio principal que engloba el contenido de un sistema |
| **\Versión** | Se genera un directorio por cada una de las versiones por año que conforman a un sistema. |
| **\adm2** | Contiene los componentes principales de Progress que permiten la operación de los sistemas desarrollados bajo esa plataforma. |
| **\pf** | Contiene los archivos que sirven para establecer la conexión a bases de datos. |
| **\img** | Contiene los archivos de imagen que utilizan diferentes programas en los sistemas. |
| **\Subsistema** | Contiene diferentes subsistemas que forman parte del sistema principal |

# Estándares de programación Progress

## Comentarios Iniciales

Todos los códigos fuente deben empezar con un encabezado que contenga: El nombre del archivo, la línea base, fecha de último cambio.

/\*

\* nombre del archivo

\*

\* fecha

\*

\* Ultimo cambio

\*

\* Nombre del programador

\*/

## Sangría

Cuatro (4) espacios deben ser usados como sangría o bien el tabulador.

## Saltos de línea

Cuando una expresión no cabe en una sola línea el salto de línea debe hacerse:

* Después de una coma
* Antes de un operador

Es necesario alinear la nueva línea con el inicio de la expresión de la línea anterior.

Si el alineado hace confuso el código puede utilizarse una sangría.

Def var v-Editar as log init no.

Def var Step as int.

Def var NomInter as char init "".

Do Step = 1 to 7:

Case Step:

when 1 then x = 5 Ner = 2.

end.

## Inicialización de variables

Tratar de inicializar las variables locales al momento en que son declaradas.

## Comentarios de bloque

Los comentarios de bloque son usados para describir archivos, métodos, estructuras de datos o algoritmos. Se usan al principio de cada archivo o antes de cada llamada a subrutina o función.

/\*Variables definidas para la obtencion de datos del SIPARH\*/

DEF VAR v\_id\_bancoPH AS INT FORMAT "999" NO-UNDO.

## Declaración de variables

Una declaración de variable por cada línea.

DEF VAR t\_cancelar AS LOG NO-UNDO.

DEF VAR t\_sigue AS INT NO-UNDO.

DEF VAR sw-rh AS CHAR FORMAT 'XX' INIT 'NO' NO-UNDO.

## Declaración if-else

if (*condition*) {

*statements*;

}

if (*condition*) {

*statements*;

} else {

*statements*;

}

if (*condition*) {

*statements*;

} else if (*condition*) {

*statements*;

} else {

*statements*;

## Declaración for

for (*initialization*; *condition*; *update*) {

*statements*;

}

## Declaración while

while (*condition*) {

*statements*;

}

## Declaración do-while

do {

statements;

} while (condition);

## Declaración case

case (condition) {

when ABC:

statements;

/\* falls through \*/

when DEF:

statements;

when XYZ:

statements;

default:

statements;

}

**Anexo A. Nomenclatura para identificadores**

Para definir un identificador es necesario seguir las siguientes recomendaciones:

* Los identificadores deben ser una o más palabras que representen el contenido de la información que almacena
* No deben contener acentos, eñes (ñ, Ñ) ni caracteres especiales, incluyendo – y \_
* En caso de que el nombre del identificador se encuentre formado por más de una palabra, debe colocarse la primera letra de cada palabra en mayúscula
* En caso de que la descripción contenga conjunciones, éstas serán omitidas
* Deben utilizarse las abreviaciones naturales de cada palabra, o en su caso, generar la abreviación partiendo de la primera letra y la primera consonante de la palabra

**Anexo B. Tipos de datos Progress**

El identificador del tipo de dato se utiliza para no perder de vista los valores que almacena una variable o regresa una función. El identificador se especifica con letras minúsculas.

A continuación se presenta la lista de los tipos de datos utilizados en Progress y su identificador.

|  |  |
| --- | --- |
| **Tipo de Dato** | **Identificador** |
| CHARACTER | c |
| COM-HANDLE | ch |
| DATE | dt |
| DECIMAL | d |
| HANDLE | h |
| INTEGER | i |
| LOGICAL | l |
| MEMPTR | m |
| RAW | rw |
| RECID | rc |
| ROWID | rd |
| WIDGET-HANDLE | wh |