



## Contenido

On Time Framework.....	1
Propiedades.....	2
Errores.....	2
Clase Grupos (Grp).....	4
Método Instalador.....	4
Métodos Básicos.....	5
Métodos Avanzados.....	7
Clase Principal (Core).....	7
Métodos Básicos.....	7
Métodos Avanzado.....	10
Funciones.....	12
Básicas.....	12
Contenido.....	14
Lectura/Escritura del contenido.....	16
Validación.....	17
Depuración.....	17
Derechos de Autor y Licencia.....	17



## On Time Framework

On Time, es la plataforma de desarrollo (framework) de Mundo Genesys diseñada especialmente para aprovechar las ventajas técnicas y económicas que se obtiene de utilizar las nubes, la versión actual es la 3.47 y se esta libreando la 4.0, su uso es totalmente intuitivo y no se requieren conocimientos técnicos para operarlo, no requiere servidor de base de datos para almacenar, es robusto y flexible.

Tecnológicamente esta totalmente separado el backend del frontend, el backend esta totalmente diseñado en php, nuestro frontend esta diseñado de forma dinámica, la resultante para el usuario final es html y java script, técnicamente es posible que ante el usuario que todo el framework trabaje unicamente con la pagina index.php, por seguridad dicha tecnología es reservada y en la demostraciones del código unicamente se muestran fragmentos inconexos.

Su estructura de almacenamiento se basa en módulos, contenedores, contenido, registro y dato, es multinivel y jerárquico y se controla totalmente por el backend.

Su estrategia de despliegue se basa en Widgets y Gadget, conceptualmente son lo mismo solo se diferencias por la amplitud, un widget es una solución a un problema especifico, si se agrupan varios Widgets lo llamamos gadget.

Los módulos: de respaldo, respaldo+, doble servicio, encriptacion, encriptacion+, fullcripto, webservice y diseño no están incluidos en esta versión gratuita, de la misma forma gadget s diseñados por Mundo Genesys y terceros tampoco son incluidos.

Por fines demostrativos todos los módulos que se incluyen están diseñados para trabajar en localhost sin ip publica o conexión a internet , y asume que se encuentra instalado apache y php 7.4, sobre un servidor ubuntu 20.04, se han escogido estas versiones no porque sea limitativo a ellas sino porque son totalmente gratuitos y por su ciclo de desarrollo garantizan soporte duran el menos 5 años.

Este manual es incremental y se publica anexo a cada clase, siendo la que se publica la primera en el mismo (no se incluyen los códigos previos debes de descargarlos manualmente).

**PRECAUCIÓN: NO ALTERES DE FORMA MANUAL LA ESTRUCTURA O CONTENIDO, LOS RESULTADOS SON IMPREVISIBLES.**

## Propiedades

Nombre	Desde	Uso
conected	Core	Tiene el estado de la conexión si lo estas o eres un usuario anónimo
content	Core	Incluye la lista del contenido del contenedor
err	Core	Código del ultimo error presentado, si la ultima llamada no presento error tiene el valor "0"
errtext	Core	Arreglos de las descripción de los errores, clasificado por su código
features	Core	Incluye los módulos instalados en el contenedor
id	Core	Id del usuario conectado, si no esta conectado ninguno tiene el valor "Anonimus"
name	Core	El nombre completo del usuario
nick	Core	Su nombre corto, apodo o como lo conocen socialmente
retval	Core	Variable interna
safety	Core	Indica todo los módulos a los que tiene acceso el usuario, tanto la suya propia que es la prioritaria como la asignada por medio de grupos
user	Core	Arreglo que obtiene la información publica del usuario, lo que quiere compartir con todos
userp	Core	Arreglo que obtiene la información privada del usuario, lo que no quiere compartir con todos

## Errores

Nombre	Desde	Significado
C0010M001	Core	Fallo la lectura de contenido



Nombre	Desde	Significado
C0010M002	Core	Fallo la creación de contenido
C0010M003	Core	Fallo la grabación de contenido
C0010M004	Core	Contenedor faltante
C0010M005	Core	Falta contenido básico del sistema, sistema corrupto
C0010M006	Core	Error de acceso
C0010M007	Core	Registro ya existe
C0010M008	Core	Registro no existe
C0010M009	Core	Registro no disponible
C0010M010	Core	Registro no visible
C0010M011	Core	No hay usuario Conectado
C0010M012	Core	No estas autorizado
C0010M013	Core	Id, no se puede cambiar
C0010M014	Core	Contenedor desconocido
C0010M015	Core	Modulo solo disponible en la versión de pago
C0010M016	Core	Error de tipo de datos
C0010M017	Core	Contenedor existe



Nombre	Desde	Significado
C0010M018	Core	Valor Invalido
C0010M019	Core	Modulo no instalado
C0010M020	Core	Modulo instalado
C0010M021	Core	Hay un usuario Conectado
C0010M022	Core	Registro no esta activo
C0010M023	Core	Registro no autorizado
C0010M024	Core	Estatus desconocido
C0010M025	Core	No es un estatus valido
C0010M026	Core	Dato invalido

## Clase Grupos (Grp)

El objetivo de los grupos es simplificar la seguridad en el framerok por lo tanto es la mejor practica, de esa manera al eliminar a un usuario de un grupo automáticamente pierde los derechos, esta clase esta dividida en 3 secciones, la de instalación solo debe de ser invocada en una ocasión, la básica siempre de estar presente, y la avanzada solo para los administradores, adicionalmente se incluye el archivo DemoGrp.php y el archivo OntimeInstallerGrp.php, al ejecutarse colococa los archivos en las posiciones requeridas para su uso y luego se elimina

### Método Instalación

Nombre	Parámetro Tipo:Contenido	Uso
InstallGrp	<b>Entrada:</b> Null <b>Salida:</b> Logico	Crea contenedores y contenido necesario para la operación del modulo de grupos



## Métodos Básicos

Nombre	Parámetro Tipo:Contenido	Uso
FtrShwGrp	Entrada: Modulo Salida: Arreglo	Muestra los modulos a los que tiene acceso el grupo
GrpAddFtr	Entrada: Grupo Entrada: Modulo Entrada: Nivel Salida: Lógico	Al grupo le asigna el acceso a un modulo con el nivel especifico, valida el código de error y valor de retorno para determinar el resultado.
GrpAddPbl	Entrada: Grupo Entrada: Clave Entrada: Valor Salida: Lógico	Al grupo se le añade el valor asociado a la clave en la información publica, valida el código de error y valor de retorno para determinar el resultado
GrpAddPrv	Entrada: Grupo Entrada: Clave Entrada: Valor Salida: Lógico	Al grupo se le añade el valor asociado a la clave en la información privada, , valida el código de error y valor de retorno para determinar el resultado
GrpAddUsr	Entrada: Grupo Entrada: Usuario Entrada: Nivel Salida: Lógico	Al grupo le agrega el usuario con el nivel indicado, valida el código de error y valor de retorno para determinar el resultado
GrpChgFtr	Entrada: Grupo Entrada: Modulo Entrada: Nivel Salida: Lógico	Si el grupo tiene acceso al modulo cambia su nivel de acceso, valida el código de error y valor de retorno para determinar el resultado
GrpChgPbl	Entrada: Grupo Entrada: Clave Entrada: Valor Salida: Lógico	Si en la información publica del grupa esta almacenada la claves modifica su valor asociado, valida el código de error y valor de retorno para determinar el resultado

Nombre	Parámetro Tipo:Contenido	Uso
GrpChgPrv	Entrada: Grupo Entrada: Clave Entrada: Valor Salida: Lógico	Si en la información privado del grupa esta almacenada la claves modifica su valor asociado, valida el código de error y valor de retorno para determinar el resultado
GrpChgUsr	Entrada: Grupo Entrada: Usuario Entrada: Nivel Salida: Lógico	Si el usuario tiene acceso al grupo cambia su nivel de acceso, valida el código de error y valor de retorno para determinar el resultado
GrpDltFtr	Entrada: Grupo Entrada: Modulo Entrada: Nivel Salida: Lógico	Elimina la relación entre el grupo y el modulo, valida el código de error y valor de retorno para determinar el resultado
GrpDltPbl	Entrada: Grupo Entrada: Clave Salida: Lógico	De la información publica el grupo elimina la clave y su valor, valida el código de error y valor de retorno para determinar el resultado asociado
GrpDltPrv	Entrada: Grupo Entrada: Clave Salida: Lógico	De la información Privada el grupo elimina la clave y su valor, valida el código de error y valor de retorno para determinar el resultado asociado
GrpDltUsr	Entrada: Grupo Entrada: Usuario Salida: Lógico	Elimina la relación entre el grupo y el usurio, valida el código de error y valor de retorno para determinar el resultado
GrpShwAll	Entrada: Null Salida: Arreglo	Muestra todos los grupos
GrpShwFtr	Entrada: Grupo Salida: Arreglo	Muestra todos los módulos del grupo
GrpShwPbl	Entrada: Grupo Salida: Arreglo	Muestra la información publica del grupo
GrpShwPrv	Entrada: Grupo	Muestra la información privado del grupo



Nombre	Parámetro Tipo:Contenido	Uso
	Salida: Arreglo	
GrpShwUsr	Entrada: Clave Salida: Arreglo	Informa la lista de usuarios de un grupo, para poder verla debes de ser partes de el
MyGrps	Entrada: Null Salida: Arreglo	Devuelve un arreglo, con todos los grupos de los que forma parte el usuario activo
MySafety	Entrada: Null Salida: Arreglo	Devuelve la módulos y nivel de acceso a los que tiene acceso el usuario conectado, lo que es asignado de forma personal es prioritario sobre lo definido por grupo

## Métodos Avanzados

Nombre	Parámetro Tipo:Contenido	Uso
CreateGroup	<b>Entrada:</b> Clave <b>Entrada:</b> Nombre <b>Entrada:</b> Nick <b>Salida:</b> Lógico	Crea un grupo con el código indicado, le asigna el nombre y el nick para identificación humana, el valor de retorno indica si es exitoso
DeleteGroup	<b>Entrada:</b> Clave <b>Salida:</b> Lógico	Elimina el grupo indicado por su código, el valor de retorno indica si es exitoso

## Clase Principal (Core)

Esta es la clase principal, en ella se manejan usuarios, seguridad y errores, para que este lista para trabajar coloca todos los archivos en el lugar donde desees llamarlo y ejecuta desde tu navegador el archivo "OnTimeInstaller.php" se distribuirán el contenido si todo es exitoso se eliminara el archivo, el archivo "DemoCore.php" te mostrara un caso practico del uso de la clase, la clase esta dividida en "para todos" (CoreB) y "para administradores" (CoreA), la básica debe de usarse siempre, la avanzada solo si quien este conectado sea un administrador.

## Métodos Básicos

Nombre	Parámetro Tipo:Contenido	Uso
Check	<b>Entrada:</b> None	Valida la existencia del contenido mínimo necesario para que el sistema opere, en caso



Nombre	Parámetro Tipo:Contenido	Uso
	<b>Salida;</b> Lógico	de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
Connect	<b>Entrada:</b> Usuario <b>Entrada:</b> Clave de acceso <b>Salida:</b> Lógico	Se emplea para abrir el sistema para un usuario en particular, si la conexión es exitosa carga su seguridad e información, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
DiscConnect	<b>Entrada:</b> None <b>Salida;</b> Lógico	Borra los datos del usuario y queda como usuario “Anonimus”
FrcPssChg	<b>Entrada:</b> Usuario <b>Entrada:</b> Clave de acceso actual <b>Entrada:</b> Nueva clave de acceso <b>Salida:</b> Lógico	Si un administrador cambio el estatus a “force”, clave de acceso debe de ser cambiada lo cual la volverá a activar en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
FrtShwAll	<b>Entrada:</b> None <b>Salida;</b> Arreglo	Muestra la lista de todos los módulos instalados, si no existe devuelve un arreglo vacío
FtrShwPrv	<b>Entrada:</b> Modulo <b>Salida:</b> Información	Muestra la información publica de un modulo, si no existe la información o el modulo regresa un arreglo vacío
FtrShwPbl	<b>Entrada:</b> Modulo <b>Salida:</b> Arreglo	Muestra la información publica de un modulo, si no existe la información o el modulo regresa un arreglo vacío
MyAddPbl	<b>Entrada:</b> Clave <b>Entrada:</b> Contenido <b>Salida:</b> Lógico	Agrega el contenido a la información publica del usuario conectado bajo el rubro de clave, si la información publica no existe la crea, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
MyAddPrv	<b>Entrada:</b> Clave <b>Entrada:</b> Contenido <b>Salida:</b> Lógico	Agrega el contenido a la información privada del usuario conectado bajo el rubro de clave, si la información privada no existe la crea, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
MyChgInf	<b>Entrada:</b> Nombre <b>Entrada:</b> Nick <b>Salida:</b> Lógico	Al usuario conectado, le asigna el nombre de usuario y el nick, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo



Nombre	Parámetro Tipo:Contenido	Uso
MyChgPbl	<b>Entrada:</b> Clave <b>Entrada:</b> Contenido <b>Salida:</b> Lógico	En la información publica del usuario, modifica el contenido almacenado bajo el rubro de clave, verificar el código de error para identificar el motivo
MyChgPrv	<b>Entrada:</b> Clave <b>Entrada:</b> Contenido <b>Salida:</b> Lógico	En la información privada del usuario, modifica el contenido almacenado bajo el rubro de clave, verificar el código de error para identificar el motivo
MyDltPbl	<b>Entrada:</b> Clave <b>Salida:</b> Lógico	En la información privada del usuario, modifica el contenido almacenado bajo el rubro de clave, verificar el código de error para identificar el motivo
MyDltPrv	<b>Entrada:</b> Clave <b>Salida:</b> Lógico	De la información privada elimina el contenido relacionado con la clave, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
MyPssChg	<b>Entrada:</b> Clave de acceso actual <b>Entrada:</b> Clave de acceso nueva <b>Salida:</b> Lógico	Del usuario actualmente conectado cambia la clave de acceso si la clave suministrada es correcta y la cuenta se encuentra activa, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
PssChk	<b>Entrada:</b> Usuario <b>Entrada:</b> Clave de acceso <b>Salida:</b> Lógico	Valida si la combinación de clave del usuario y clave de acceso del usuario son correctas y el usuario esta activo en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo
UserDlt	<b>Entrada:</b> Usuario <b>Salida:</b> Lógico	Elimina todo rastro del usuario, tanto físico como lógico, esta acción no puede revertirse, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo, verificar el código de error para identificar si hay error.
UserShwPbl	<b>Entrada:</b> Usuario <b>Salida:</b> Arreglo	Busca si existe información publica de un usuario, si no existe el usuario o su información publica devuelve un arreglo vacío, verificar el código de error para identificar si hay error.
UsrShwAll	<b>Entrada:</b> None <b>Salida:</b> Arreglo	Devuelve la lista de todos los usuarios, en caso de obtener una lista en blanco verificar el código de error para identificar el motivo.
UsrShwNln	<b>Entrada:</b> None <b>Salida:</b> Arreglo	Devuelve la lista de usuarios activos, en caso de obtener una lista en blanco verificar el código de error para identificar el motivo.



## Métodos Avanzado

Nombre	Parámetro Tipo:Contenido	Uso
ErrAdd	<b>Entrada:</b> Clave <b>Entrada:</b> Descripción <b>Salida:</b> Lógico	Agrega la clave y descripción del error, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
ErrChg	<b>Entrada:</b> Clave <b>Entrada:</b> Descripción <b>Salida:</b> Lógico	Modifica la descripción asociada a la clave del error, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
ErrDlt	<b>Entrada:</b> Clave <b>Salida:</b> Lógico	Elimina la descripción asociada a la clave del error, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
FtrAddPbl	<b>Entrada:</b> Modulo <b>Entrada:</b> Clave <b>Entrada:</b> Descripción <b>Salida:</b> Lógico	Si el modulo existe y el usuario tiene permiso en el mismo, añade una nueva descripción relacionada a la clave en el contenido publico, el valor de retorno contiene el resultado, en caso de ser falso validad el código de error
FtrAddPrv	<b>Entrada:</b> Modulo <b>Entrada:</b> Clave <b>Entrada:</b> Descripción <b>Salida:</b> Lógico	Si el modulo existe y el usuario tiene permiso en el mismo, añade una nueva descripción relacionada a la clave en el contenido privado, el valor de retorno contiene el resultado, en caso de ser falso validad el código de error
FtrChgPbl	<b>Entrada:</b> Modulo <b>Entrada:</b> Clave <b>Entrada:</b> Descripción <b>Salida:</b> Lógico	Si el modulo existe y el usuario tiene permiso en el mismo, cambia la descripción relacionada a la clave en el contenido publico, el valor de retorno contiene el resultado, en caso de ser falso validad el código de error
FtrChgPrv	<b>Entrada:</b> Modulo <b>Entrada:</b> Clave <b>Entrada:</b> Descripción <b>Salida:</b> Lógico	Si el modulo existe y el usuario tiene permiso en el mismo, cambia la descripción relacionada a la clave en el contenido privado, el valor de retorno contiene el resultado, en caso de ser falso validad el código de error
FtrDltPbl	<b>Entrada:</b> Modulo <b>Entrada:</b> Clave <b>Salida:</b> Lógico	Si el modulo existe y el usuario tiene permiso en el mismo, elimina la descripción relacionada a la clave en el contenido publico, el valor de retorno contiene el resultado, en caso de ser falso validad el código de error



Nombre	Parámetro Tipo:Contenido	Uso
FtrDltPrv	<b>Entrada:</b> Modulo <b>Entrada:</b> Clave <b>Salida:</b> Lógico	Si el modulo existe y el usuario tiene permiso en el mismo, elimina la descripción relacionada a la clave en el contenido privado, el valor de retorno contiene el resultado, en caso de ser falso validad el código de error
FtrShwUsr	<b>Entrada:</b> Modulo <b>Salida:</b> Arreglo	Devuelve los usuarios y niveles asociados, a la clave de modulo, en caso de obtener un arreglo vacío, verificar el código de error para identificar el motivo
UserAdd	<b>Entrada:</b> Clave <b>Entrada:</b> Clave de acceso <b>Entrada:</b> Estatus <b>Entrada:</b> Nombre (opcional) <b>Entrada:</b> Nick (opcional) <b>Salida:</b> Lógico	Agrega un usuario al sistema, y le asigna el estatus inicial, en caso de que no se suministre nombre y/o nick se les asigna el valor 'None', en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UserChgStt	<b>Entrada:</b> Clave <b>Salida:</b> Valor	Devuelve el valor del estatus del usuario, en caso de ser un valor vacío, verificar el código de error para identificar el motivo.
UserChkStt	<b>Entrada:</b> Clave <b>Entrada:</b> Valor <b>Salida:</b> Lógico	Modifica el valor del estatus del usuario al nuevo valor, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UserChkStt	<b>Entrada:</b> Clave <b>Entrada:</b> Valor <b>Salida:</b> Lógico	Modifica el valor del estatus del usuario al nuevo valor, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UsrAddFtr	<b>Entrada:</b> Clave <b>Entrada:</b> Modulo <b>Entrada:</b> Nivel <b>Salida:</b> Lógico	Agrega al usuario clave acceso al modulo con el nivel, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UsrAddMn	<b>Entrada:</b> Clave <b>Entrada:</b> Nivel <b>Salida:</b> Lógico	Agrega al usuario clave acceso al modulo interno con el nivel, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UsrChgFtr	<b>Entrada:</b> Clave <b>Entrada:</b> Modulo <b>Entrada:</b> Nivel	Modifica el nivel de acceso al usuario clave acceso del modulo, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.

Nombre	Parámetro Tipo:Contenido	Uso
	<b>Salida:</b> Lógico	
UsrChgMn	<b>Entrada:</b> Clave <b>Entrada:</b> Nivel <b>Salida:</b> Lógico	Modifica el nivel de acceso al usuario clave acceso del modulo interno, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UsrDltFtr	<b>Entrada:</b> Clave <b>Entrada:</b> Modulo <b>Entrada:</b> Nivel <b>Salida:</b> Lógico	Elimina al usuario clave del modulo, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UsrDltMn	<b>Entrada:</b> Clave <b>Salida:</b> Lógico	Elimina al usuario clave del modulo interno, en caso de que el valor devuelto sea falso, verificar el código de error para identificar el motivo.
UsrShwFtr	<b>Entrada:</b> Clave <b>Salida:</b> Arreglo	Devuelve los módulos y niveles asociados, a la clave de usuario, en caso de obtener un arreglo vacío, verificar el código de error para identificar el motivo
UsrShwMn	<b>Entrada:</b> None <b>Salida:</b> Arreglo	Devuelve los usuarios que tienen derechos a realizar modificaciones en el modulo interno así como su nivel de acceso, en caso de obtener un arreglo vacío, verificar el código de error para identificar el motivo

## Funciones

### Básicas

Nombre	Parámetro Tipo:Contenido	Desde	Uso
not_exist	<b>Entrada:</b> Contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico	Core	Ademas de validar la no existencia del contenido en el contenedor, también almacena el resultado en el error y en el valor de retorno
not_value	<b>Entrada:</b> Dato <b>Entrada:</b> Valor <b>Entrada:</b> Error	Core	Valida que el dato y el valor sean diferentes si son idénticos registra el error y el valor de retorno queda como falso



Nombre	Parámetro Tipo:Contenido	Desde	Uso
	<b>Salida:</b> Lógico		
ot_can	<b>Entrada:</b> Nivel <b>Entrada:</b> Modulo <b>Salida:</b> Lógico	Core	Valida que usuario actualmente conectado, en el modulo tiene un nivel menor o igual a nivel requerido, el nivel 0 es administrador, el 1 es puede eliminar, el 2 puede añadir y el 3 puede cambiar, no solo regresa el resultado lo almacena en el error y valor de retorno
ot_clean	<b>Entrada:</b> Cadena <b>Salida:</b> Limpia	Core	A la cadera le elimina caracteres ocultos para dejar una cadena limpia que se pueda procesar mas fácilmente
ot_connect	<b>Entrada:</b> Lógico (opcional) <b>Salida:</b> Lógico	Core	Si el parámetro de entrada es cierto o se omite, valida si existe un usuario conectado, ,si es falso valida que nadie esta conectado, en ambos casos devuelve el resultado
ot_create	<b>Entrada:</b> Contenedor <b>Entrada:</b> En (opcional <b>Salida:</b> Lógico	Core	Esta función crea un contenedor, en la raíz o dentro del otro, en caso de presentarse error lo almacena en la variable y el valor de retorno
ot_destroy	<b>Entrada:</b> Contenedor <b>Salida:</b> Null	Core	Esta función es recursiva y elimina uno a uno contenido y contenedor
ot_exist	<b>Entrada:</b> Contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico	Core	Ademas que valida la existencia del contenido en el contenedor, también almacena el resultado en el error y en el valor de retorno
ot_getin	<b>Entrada:</b> Contenedor <b>Salida:</b> Arreglo	Core	Devuelve el arreglo de los contenedores dentro del contenedor, nunca considera la posibilidad de error, si no existe el contenedor o esta vacío, devuelve un arreglo vacio
ot_maxvalue	<b>Entrada:</b> Base <b>Entrada:</b> Valor <b>Entrada:</b> Error <b>Salida:</b> Lógico	Gpr	Basándose en la base y el valor valida si es menor o igual, en caso contrario presenta el error, el resultado se almacena en el valor de retorno y el error
ot_name	<b>Entrada:</b> Contenedor <b>Salida:</b> Nombre	Core	De un contenedor en particular si existe devuelve el nombre y el nick del mismo, si no existe devuelve la cadena vacía

Nombre	Parámetro Tipo:Contenido	Desde	Uso
ot_now	<b>Entrada:</b> Formato (opcional) <b>Salida:</b> Cadena	Core	Devuelve en la cadena, la fecha y/o hora del servidor, en el formato especificado
ot_qexist	<b>Entrada:</b> Contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico	Core	Regresa si existe o no el contenido dentro del contenedor, solo regresa un valor lógico no almacena un error o un valor de retorno r
ot_RandomString	<b>Entrada:</b> Largo <b>Salida:</b> Cadena	Core	Devuelve una cadena alfanumérica aleatoria, de largo en caracteres, las cadenas aleatorias siempre comienzan en X
ot_remove	<b>Entrada:</b> Contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico	Core	Elimina el contenido dentro del contenedor en caso de que exista, si el contenido es un contenedor elimina todo el contenido, en caso de error lo almacena en su variable y en el valor de retorno
ot_value	<b>Entrada:</b> Dato <b>Entrada:</b> Valor <b>Entrada:</b> Error <b>Salida:</b> Lógico	Core	Valida que el dato y el valor sean idénticos, en caso contrario, registra el error, y el valor de retorno queda en falso

## Contenido

Nombre	Parámetro Tipo:Contenido	Desde	Uso
ot_add	<b>Entrada:</b> Clave <b>Entrada:</b> Valor <b>Entrada:</b> Contenido (arreglo) <b>Entrada:</b> Nombre del contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Arreglo	Core	Busca la clave en el contenido recibido, en caso que no exista lo agrega asociandolo al valor, inmediatamente lo almacena usando el nombre del contenido, dentro de un contenedor que puede omitirse, si existe la clave coloca el error que aplica y falso en el valor de retorno, el resultado es el contenido original adicionando el nuevo si es que procede
ot_addchangein	<b>Entrada:</b> Clave	Core	Abre el contenido localizado en el contenedor, busca la clave si no





Nombre	Parámetro Tipo:Contenido	Desde	Uso
	<b>Entrada:</b> Valor <b>Entrada:</b> Nombre del contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico		existe agrega la clave asociándola al valor, si existe modifica el valor relacionado, si el contenido no existe lo crea, en todo caso devuelve el arreglo con la nueva información
ot_addin	<b>Entrada:</b> Clave <b>Entrada:</b> Valor <b>Entrada:</b> Nombre del contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Logico	Core	Abre el contenido localizado en el contenedor, busca la clave, si no existe la agrega asociada al valor, si el contenido no existe lo crea, si la clave ya existe, mueve el valor del error y del valor del retorno, en ambos caos devuelve como resultado el arreglo
ot_array	<b>Entrada:</b> Arreglo <b>Entrada:</b> Nombre del contenido <b>Entrada:</b> Sobrescribir <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico	Core	Si sobrescribir es falso, abre el contenido dentro del contenedor, agrega el arreglo al a lectura si la clave existe modifica el valor, si no existe contenido previo o tiene la opción de sobre escribir el contenido es es el valor del arreglo, unicamente maneja un error si el Arreglo que debe de suministrar no es tal
ot_change	<b>Entrada:</b> Clave <b>Entrada:</b> Valor <b>Entrada:</b> Contenido (arreglo) <b>Entrada:</b> Nombre del Contenido <b>Entrada:</b> Contenedor (Opcional) <b>Salida:</b> Arreglo	Core	Busca la clave en el contenido, si existe modifica su valor asociado, si no existe mueve el error y el valor de retorno a falso, lo almacena inmediatamente en el contenedor con el nombre del contenido, el valor de regreso es el arreglo modificado
ot_changein	<b>Entrada:</b> Clave	Core	Si existe abre el contenido dentro del contenedor, dentro del el busca la



Nombre	Parámetro Tipo:Contenido	Desde	Uso
	<b>Entrada:</b> Valor <b>Entrada:</b> Nombre del Contenido <b>Entrada:</b> Contenedor (Opcional) <b>Salida:</b> lógico		clave y en caso afirmativo le asigna el valor, grabándolo inmediatamente, si estas condiciones no se cumple graba el error y modifica el valor de retorno, devuelve un arreglo con la información.
ot_delete	<b>Entrada:</b> Clave <b>Entrada:</b> Contenido (arreglo) <b>Entrada:</b> Nombre del Contenido <b>Entrada:</b> Contenedor (Opcional) <b>Salida:</b> Arreglo	Core	Busca la clave en el contenido y si existe la elimina del contenido, almacenándolo con el nombre indicado dentro del contenido, regresa el arreglo resultante, si se presenta un error lo almacena tanto en su variable como en la de retorno
ot_deletein	<b>Entrada:</b> Clave <b>Entrada:</b> Nombre del Contenido <b>Entrada:</b> Contenedor (Opcional) <b>Salida:</b> lógico	Core	Abre el contenido en el contenedor, busca la clave y la elimina, almacenado el nuevo contenido, si se presento un error este se coloca en su propiedad igual que el valor de retorno

## Lectura/Escritura del contenido

Nombre	Parámetro Tipo:Contenido	Desde	Uso
ot_read	<b>Entrada:</b> Contenido <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Arreglo	Core	Trata de leer el contenido en el contenedor, en caso de que no exista reconoce el error, también se refleja en el valor de retorno, se asume que el contenido es JSON.
ot_readif	<b>Entrada:</b> Contenido <b>Entrada:</b> Contenedor (opcional)	Core	Trata de leer el contenido en el contenedor, en caso de no existir, no marca error lo genera en blanco se asume que el contenido es JSON.



Nombre	Parámetro Tipo:Contenido	Desde	Uso
	<b>Salida:</b> Arreglo		
ot_write	<b>Entrada:</b> Contenido <b>Entrada:</b> Datos <b>Entrada:</b> Contenedor (opcional) <b>Salida:</b> Lógico	Core	Trata de escribir los datos en el contenido dentro del contenedor

## Validación

Nombre	Parámetro Tipo:Contenido	Desde	Uso
ot_record	No aplica	Core	No aplica
ot_valid	No aplica	Core	No aplica

## Depuración

Nombre	Parámetro Tipo:Contenido	Desde	Uso
ot_error	<b>Entrada:</b> Valor (opcional) <b>Salida:</b> Texto	Core	En caso de que en la ultima llamada no presente error, devuelve como texto el valor, si se presento error busca la definición del mismo y la devuelve si la clave de error no esta registra el texto es Undefined Error
ot_log	No Aplica	Core	No Aplica
ot_show	<b>Entrada:</b> Datos <b>Entrada:</b> Nivel (opcional) <b>Salida:</b> Null	Core	Despliega a pantalla los datos indicando clave y valor, taladra los niveles que se le indique si son otros arreglos



[marioc@mundogenesys.com](mailto:marioc@mundogenesys.com) [mario.carrocera@hotmail.com](mailto:mario.carrocera@hotmail.com)

## Derechos de Autor y Licencia

\*\*\*\*\*

Diseño y programación por Mario Carrocera Menéndez

copyright @ 2003-2021 Mario Carrocera Menéndez

copyright @ 2019-2021 Mundo Genesys de IT (On Time)

Licencia para las versiones anteriores a la 4.0 publicadas como open source en los términos de:

Attribution-NonCommercial-ShareAlike 4.0 International

<https://creativecommons.org/licenses/by-nc-sa/4.0/>

Limite de Responsabilidad:

Mario Carrocera Menéndez y/o Mundo Genesys de IT, no son responsables legal o moralmente de daños físicos o lógicos causados por el uso del software, tampoco se comprometen a realizar ninguna modificación al mismo, se suministra de forma gratuita en su estado actual y su venta esta prohibida.

Pongo a tu disposición mis mail de contacto, sin embargo, el recibir una sugerencia incluya código fuente o no, no me comprometo de ninguna forma a incluirlo y/o genera compromisos comerciales o legales con quien me envíe un mail,

Si deseas desarrollos especiales para ti o tu empresa, también te ofrezco mis servicios como desarrollador independiente, siéntete con la total confianza de contactarme

Email:

[marioc@mundogenesys.com](mailto:marioc@mundogenesys.com)



[mario.carrocera@hotmail.com](mailto:mario.carrocera@hotmail.com)

[marcartools@gmail.com](mailto:marcartools@gmail.com)

Fecha de Ultima Modificación:

2 / 1 / 2021

Release: 3.47

\*\*\*\*\*

On Time  
Mario Carrocera@2003-2021  
marioc@mundogenesys.com